Diss. ETH No. 25557

COMPUTATIONAL CAUSALITY AND LEARNING FROM PARTITIONED DATA

A thesis submitted to attain the degree of

DOCTOR OF SCIENCES of ETH ZURICH
(Dr. sc. ETH Zurich)

presented by
CHRISTINA HEINZE-DEML

Master of Science ETH in Statistics, ETH Zurich
born April 10, 1989
citizen of Germany

accepted on the recommendation of
Prof. Dr. Nicolai Meinshausen, examiner
Prof. Dr. Jonas Peters, co-examiner

2018

To Stefan.

Acknowledgements

First of all, I wish to thank my supervisor Nicolai Meinshausen for all the great support and helpful guidance over the course of the past four years. Your enthusiasm, patience, endless creativity, openness to explore new areas, and kindness made my doctoral studies truly enjoyable—I cannot thank you enough!

Furthermore, a very big thanks to Jonas Peters. Working with you has always been very pleasant, motivating and interesting. I have always highly valued your feedback and guidance.

Thanks to everyone at the Seminar for Statistics for providing such a friendly environment—and especially thanks to Susan, "unserer guten Fee", for everything you do. I would also like to thank the ISG at the D-MATH for all the technical support and for always trying to accommodate my technical requests.

At the beginning of 2018, I had a great time at Facebook AI research in New York. Thanks to Léon Bottou, David Lopez-Paz, Martín Arjovsky and Diane Bouchacourt for insightful discussions. Thanks, Diane, for uncountable video calls and your boundless enthusiasm for research. Your curiosity is truly inspiring.

In the fall of 2018, I spent three exciting months at DeepMind in London. Thanks to Silvia Chiappa and everyone in the team for a very interesting time, all the things I learned and the welcoming atmosphere.

A special thanks to Brian McWilliams for helpful discussions and encouragement, valuable advice for navigating the academic world, being my personal English teacher, common work and invaluable ice cream meetings. You have been a great mentor and friend.

Last but not least, I would like to thank my family for their continuous support and for always being there for me—your contribution to this work is immeasurable. Above all, I am especially grateful to Stefan—your extraordinary drive to learn and to understand motivates and inspires me every day. Thank you for your love, your encouragement and your support.

Contents

Abstract xiii

Zusammenfassung xv

I Introduction and Background 1

1 Introduction 3
1.1 Causal models 4
1.1.1 Robustness and interpretability 6
1.2 Computation 13
1.3 Learning from partitioned data 14
1.4 Outline and contributions 16
1.4.1 Publications 18

2 Causal structure learning 21
2.1 Introduction 21
2.2 The model 22
2.2.1 Interventions 23
2.2.2 Graphical representation 24
2.2.3 Factorization and truncated factorization 24
2.2.4 Counterfactuals 25
2.2.5 Assumptions 25
2.3 Methods 27
2.3.1 Target graphical objects 27
2.3.2 Ancestral and parental relationships 28
2.3.3 Considered methods 29
2.4 Empirical evaluation 34
2.4.1 Data generation 35
2.4.2 Evaluation methodology 37

2.4.3 Results 40
2.5 Discussion 48

Appendices

2.A Considered tuning parameter configurations 50
2.B Simulation settings 52

II Invariance-based Causal Learning 55

3 backShift: Learning causal cyclic graphs 57
3.1 Introduction 57
3.2 Method 60
3.2.1 Grouping of data 60
3.2.2 Notation 61
3.2.3 Assumptions 61
3.2.4 Population method 62
3.2.5 Finite-sample estimate of the connectivity matrix . . 63
3.2.6 Estimating the intervention variances 64
3.3 Identifiability 65
3.4 Numerical results 66
3.4.1 Synthetic data 66
3.4.2 Flow cytometry data 69
3.4.3 Financial time series 71
3.5 Conclusion 73

Appendices

3.A Identifiability – Proof of Theorem 3.1 74
3.B Polynomial-time algorithm 76
3.C Intervention variances and model misspecification 78
3.D Beyond covariances 79
3.E Additional figures 80

4 Invariant Causal Prediction for nonlinear models 83
4.1 Introduction 84
4.1.1 Structural causal models 84
4.1.2 Causal discovery 85
4.1.3 Invariance based causal discovery 86
4.1.4 Contribution 87
4.1.5 Fertility rate modeling 88
4.2 Nonlinear Invariant Causal Prediction 90

4.2.1 Invariance approach for causal discovery 90
4.2.2 Defining sets 96
4.2.3 Confidence bands 97
4.2.4 Average causal effects 97
4.3 Conditional independence tests 108
4.4 Simulation study 111
4.5 Discussion and future work 119

Appendices

4.A Time series bootstrap procedure 121
4.B Conditional independence tests 121
4.B.1 Kernel conditional independence test 122
4.B.2 Residual prediction test 122
4.B.3 Invariant environment prediction 124
4.B.4 Invariant target prediction 126
4.B.5 Invariant residual distribution test 127
4.B.6 Invariant conditional quantile prediction 128
4.B.7 Overview of conditional independence tests in `CondIndTests` package 129
4.C Experimental settings for numerical studies 130
4.D Additional experimental results 133
4.E Example 138

5 Conditional variance penalties and domain shift robustness 141
5.1 Introduction 142
5.2 Motivating examples 145
5.3 Related work 146
5.4 Setting 149
5.4.1 Notation 149
5.4.2 Causal graph 150
5.4.3 Data 153
5.4.4 Domain adaptation, adversarial examples and adversarial domain shifts 153
5.5 Conditional variance regularization 155
5.5.1 Invariant parameter space 155
5.5.2 CoRe estimator 156
5.5.3 Estimating expected conditional variance as a graph Laplacian 157
5.5.4 Classification example 158
5.6 Adversarial risk consistency and distributional robustness . 158

5.6.1 Adversarial risk consistency for classification and logistic loss . . . 160
5.6.2 Population limit: optimal robustness against increases of the style-noise variance . . . 162
5.7 Experiments . . . 165
5.7.1 Eyeglasses detection with small sample size . . . 165
5.7.2 Eyeglasses detection with known and unknown image quality intervention . . . 168
5.7.3 Stickmen image-based age classification with unknown movement interventions . . . 170
5.7.4 Gender classification with unknown confounding . . . 172
5.7.5 MNIST: more sample efficient data augmentation . . . 174
5.7.6 Elmer the Elephant . . . 174
5.7.7 Eyeglasses detection: unknown brightness intervention 176
5.8 Conclusion . . . 177

Appendices

5.A Proof of Theorem 5.2 . . . 179
5.B Proof of Theorem 5.3 . . . 182
5.C Network architectures . . . 183
5.D Additional experiments . . . 184
5.D.1 Eyeglasses detection: known and unknown brightness interventions . . . 184
5.D.2 Gender classification . . . 188
5.D.3 MNIST: more sample efficient data augmentation . . . 190
5.D.4 Stickmen image-based age classification . . . 190
5.D.5 Eyeglasses detection: image quality intervention . . . 192
5.D.6 Elmer the Elephant . . . 195

III Distributed Estimation **199**

6 Distributing statistical estimation using random projections **201**
6.1 Introduction . . . 201
6.2 Related work . . . 203
6.2.1 Distributed estimation . . . 203
6.2.2 Random projections for estimation and optimization 204
6.3 The Dual-Loco algorithm . . . 206
6.4 Dual-Loco approximation error . . . 209
6.5 Implementation and experiments . . . 210

6.6 Conclusions and further work 218

Appendices
6.A Supplementary results 220
6.B Proof of row summing lemma 222
6.C Supplementary material for §6.5 227

7 Preserving privacy between features in distributed estimation 229
7.1 Introduction 229
7.2 Problem setting 232
7.3 The PriDE algorithm 233
7.4 Analysis 235
  7.4.1 Distributed privacy guarantee 236
  7.4.2 Approximation error of PriDE 237
7.5 Experiments 239
  7.5.1 Simulated data 240
  7.5.2 Climate model data 242
7.6 Related work 245
7.7 Conclusions and further work 246

Appendices
7.A Proofs 248
7.B Supporting results 253
7.C Connection between $(\epsilon, \delta, \mathcal{S})$-differential privacy and regularized least-squares estimation 254
7.D Additional experimental details and results 255
  7.D.1 Breast cancer gene expression data 255
  7.D.2 Data tables 261
  7.D.3 Simulation setting 261
  7.D.4 Additional results for simulated data 263
  7.D.5 Additional results for climate model data 265
7.E Privacy preserving cross validation 268

Bibliography 271

Curriculum vitae 293

Abstract

In many areas of science and industry, the volume of data that is collected and stored has increased significantly over the past years (Marx, 2013). Statistical and machine learning approaches start to get adopted more widely and data-driven decision making is gaining traction both in industry as well as the public sector (Dieterich et al., 2016). With this widening impact, the need for methods that have certain robustness properties, are computationally efficient and can handle privacy and fairness requirements (Barocas and Selbst, 2016) is growing. Similarly, discovering cause-effect relationships between variables is an important goal in many applications, leading to a rising demand for causal inference.

The starting point of a statistical analysis (which can be part of a wider scientific study) is a training dataset that was previously collected. Often one assumes that the observations are independently and identically distributed (i.i.d.). Furthermore, future data—for which the results of the analysis should be valid—is assumed to follow the same distribution as the training data. However, in modern real-world applications of statistics and machine learning these assumptions may not always hold. In many cases the collected data is partitioned in some form. Broadly speaking, it may have been recorded in different "circumstances" or "environments", giving rise to heterogeneities in the training distribution. For instance, the data may have been gathered at various geographical locations or may result from different experiments. Another case of partitioned data can arise when data has been collected by distinct parties such as different institutions.

These heterogeneities in the training data lead to an array of challenges as many classical statistical methods may not be directly applicable. They might be too computationally demanding for the potentially large amounts of data collected by different parties. Another difficulty could arise due to privacy concerns that prevent sharing data between different data owners. Finally, future data may follow yet a slightly different distribution than the ones encountered in the different partitions of the training data. This may lead to poor generalization performance if the analysis relies on a statistical estimator without robustness properties addressing these distribution shifts. On the other hand, partitioned data can also constitute an opportunity for statistical inference. In this thesis, we study the following challenges and opportunities related to partitioned data.

– How can data collected in different environments be used for causal inference? (Chapters 3 and 4)

– How can a grouping structure in the data be leveraged to attain distributional robust estimators? (Chapter 5)

– How can a joint model be estimated when the data is held by different parties and is too large to be stored on a single computer? (Chapter 6) How can this be achieved under privacy constraints? (Chapter 7)

Zusammenfassung

In vielen Bereichen der Wissenschaft und der Industrie ist das gesammelte und gespeicherte Datenvolumen in den letzten Jahren stark gestiegen (Marx, 2013). Statistische Ansätze und solche des maschinellen Lernens werden zunehmend verwendet und Methoden zur datengetrieben Entscheidungsfindung erhalten Einzug sowohl in der Industrie als auch im öffentlichen Sektor (Dieterich u. a., 2016). Mit diesem wachsenden Einfluss steigt der Bedarf für Methoden, die bestimmte Robustheitseigenschaften haben, die rechnerisch effizient sind und welche Datenschutz- und Gerechtigkeitsanforderungen (Barocas und Selbst, 2016) erfüllen. Des Weiteren ist bei vielen Anwendungen die Entdeckung von Ursache-Wirkung-Beziehungen zwischen Variablen ein wichtiges Ziel. Dies führt zu einem vermehrten Bedarf für kausale Inferenz.

Der Ausgangspunkt einer statistischen Analyse (welche Teil einer umfangreicheren wissenschaftlichen Studie sein kann) ist ein Trainingsdatensatz, der zuvor gesammelt wurde. Oft nimmt man an, dass die Beobachtungen die gleiche Verteilung haben und unabhängig voneinander sind. Zudem wird angenommen, dass künftige Daten—für welche die Analyse gültig sein sollte—der gleichen Verteilung wie die Trainingsdaten folgen. In modernen praktischen Anwendungen der Statistik und des maschinellen Lernens sind diese Annahmen jedoch nicht immer erfüllt. In vielen Fällen sind die gesammelten Daten auf eine Weise partitioniert. Allgemein formuliert können die Daten unter verschiedenen “Umständen” oder in unterschiedlichen “Umgebungen” erhoben worden sein, die zu Heterogenitäten in den Trainingsdaten führen. Beispielsweise können die Daten an diversen geografischen Orten aufgezeichnet worden sein oder das Resultat mehrerer Experimente sein. Ein weiterer Fall von partitionierten Daten kann daraus resultieren, dass verschiedene Parteien die Daten erhoben haben, z.B. unterschiedliche Institutionen.

Diese Heterogenitäten in den Trainingsdaten stellen eine Reihe von Herausforderungen dar, weil viele klassische statistische Methoden möglicherweise nicht direkt anwendbar sind. Diese mögen rechnerisch zu ineffizient für die potentiell grossen Datenmengen sein, die von verschiedenen Parteien gesammelt wurden. Eine weitere Schwierigkeit kann durch Datenschutzanforderungen entstehen, die den Austausch von Daten zwischen verschiedenen Dateneigentümern verhindern. Zudem können weitere Komplikationen daraus resultieren, dass die Verteilung von zukünftigen Daten unter Umständen wieder etwas unterschiedlich von den Verteilungen ist, die in den verschiedenen Partitionen der Trainingsdaten vorgefunden wurden. Dies führt in der Regel zu einer schlechten Generalisierung, wenn die Analyse auf einem statistischen Schätzer beruht, welcher keine Robustheitseigenschaften in Bezug auf solche Verteilungsverschiebungen aufweist. Auf der anderen Seite können partitionierte Daten auch eine Chance für statistische Inferenz darstellen. In dieser Dissertation betrachten wir die folgenden Herausforderungen und Chancen für statistische Inferenz, wenn partitionierte Daten vorliegen.

- Wie können Daten, die in unterschiedlichen Umgebungen gesammelt wurden, für kausale Inferenz genutzt werden? (Kapitel 3 und 4)
- Wie kann eine Gruppenstruktur in den Daten ausgenutzt werden um verteilungsrobuste Schätzer zu erhalten? (Kapitel 5)
- Wie kann ein gemeinsames Modell aus Daten geschätzt werden, wenn die Daten von verschiedenen Parteien erhoben wurden und die Datenmenge zu gross ist um die Daten am selben Ort zu speichern? (Kapitel 6) Wie kann dies auch unter Datenschutzanforderungen erreicht werden? (Kapitel 7)

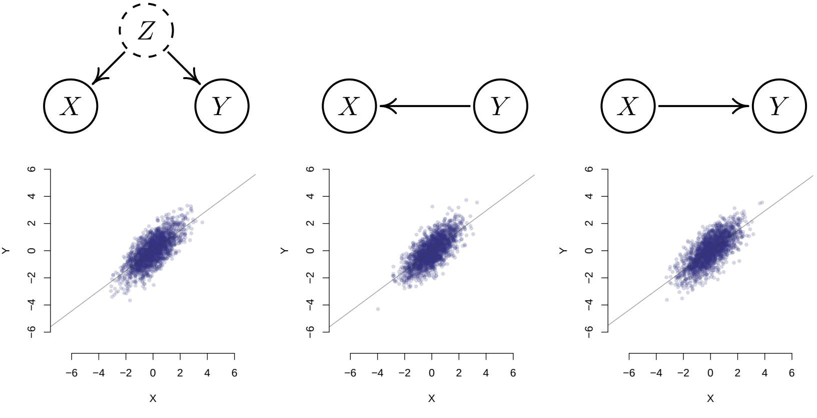
Part I.

Introduction and Background

Chapter 1.

Introduction

In recent years, we have witnessed a number of success stories in artificial intelligence, machine learning and statistics. Notably, deep neural networks (DNNs) have achieved outstanding performance on prediction tasks like visual object recognition (He et al., 2015; Krizhevsky et al., 2012; Szegedy et al., 2015) and machine translation has improved considerably when adopting neural network based systems (Wu et al., 2016). Famously, DeepMind's Alpha Go (Silver et al., 2016) beat the world champion Lee Sedol at the ancient Asian board game Go. At that time, this accomplishment was believed not to be attainable until a decade later (Knight, 2016). Also in our daily lives, statistical methods and machine learning models increase in impact as they gain traction in areas such as medicine. For instance, models based on convolutional neural networks (CNNs) have been reported to outperform the accuracy of dermatologists when classifying images of skin lesions as being benign or malignant (Esteva et al., 2017). While these are important advances, there are significant caveats and challenges that remain to be addressed. Important statistical, computational and social questions related to stability and robustness as well as privacy and fairness arise. In part II of this thesis, we propose a number of methods using causal modeling to address the challenges of stability and robustness. In part III, we focus on computational questions arising when a model needs to be estimated from data held by different parties and we also consider a setting where privacy considerations restrict the sharing of the data in its original form. In the following, we give a brief introduction to the topics and challenges addressed in this thesis.



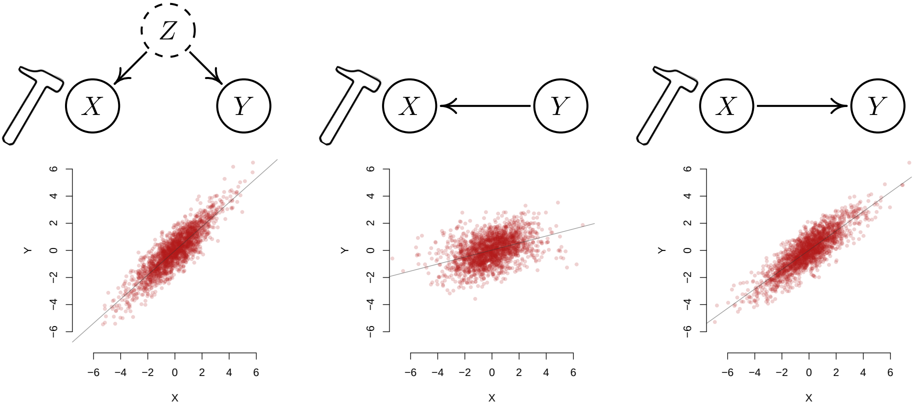
(a) (b) (c)

Figure 1.1.: Three different causal graphical models can give rise to the same observational joint distribution of X and Y.

1.1. Causal models

In this work, we propose a number of methods that aim to infer causal relations from observational data. Therefore, we briefly introduce some properties of causal models here. A more formal introduction to causal structure learning—the problem of trying to infer the underlying causal graph from observational data—and related concepts is presented in Chapter 2.

It is well-known that correlation does not imply causation. This point is illustrated in Figure 1.1 where we consider the simple case of having observed data from two random variables X and Y. The three different causal graphical models shown in panels (a)–(c) can give rise to the same observational joint distribution of X and Y. In this distribution X and Y are strongly correlated but only in panel (c), X has a direct causal effect on Y as illustrated by the arrow in the corresponding graph. X is then also called a “(causal) parent” of Y. When Y causes X, as in panel (b), X is a “(causal) child” of Y. The third possibility giving rise to the strong dependence between X and Y we consider here is shown in panel (a) where X and Y have a common cause Z. Z is called a confounder. When only observing the data shown in the scatter plots in Figure 1.1, we cannot



(a) (b) (c)

Figure 1.2.: When intervening on the predictor X, the conditional distribution of $Y|X = x$ changes in panels (a) and (b), compared to the observational case shown in Figure 1.1. The differences can be best seen when comparing the fitted regression lines. In panel (c), the conditional distribution of $Y|X = x$ remains invariant.

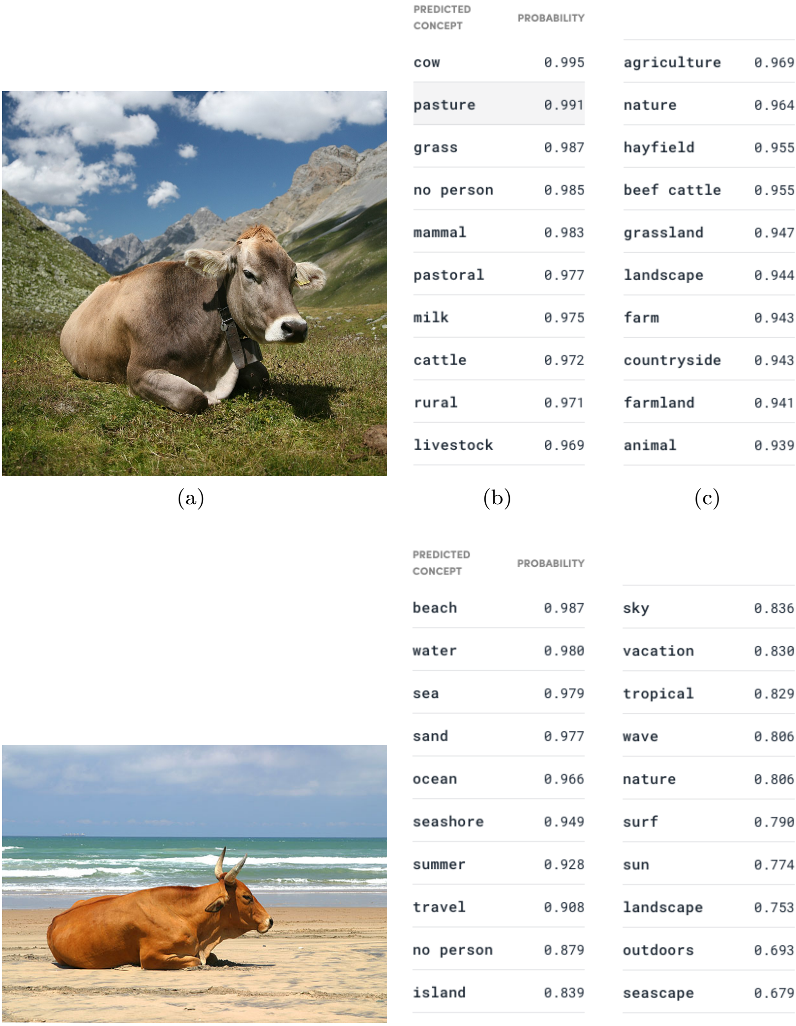
infer the underlying causal graph without making further assumptions. One possibility to identify the data generating structure arises when we have observed the system in different conditions or "environments". This is illustrated in Figure 1.2. When the distribution of X is manipulated externally in the three considered models, we observe different effects in the joint and conditional distributions. In this example, the external manipulation consists of adding i.i.d. noise to X. In case of panels (a) and (b) the conditional distribution of $Y|X = x$ changes; in case of panel (c) it stays invariant (the differences can be best seen when comparing the fitted regression lines). In other words, only the causal model, i.e. the model in which the target variable Y is predicted using its causal parent(s), continues to give valid predictions when we manipulate the distribution of the predictor X externally.

In causal inference, these external manipulations are called "interventions". Mathematically, they can be formulated in different ways as discussed in §2.2.1. Conceptually, interventions can be used to model different kinds of changes in the joint distribution of the variables of interest. The property that the causal model remains valid under interventions sets causal graphical models apart from standard probabilistic graphical models. Causal graphical models do not only describe an observational multivariate distribution; they also describe the distributions induced by arbitrary and unseen interventions on variables in the graph. Consequently, causal graphical models allow us to reason about questions like what changes if a variable is set to a particular value or what happens if the noise distribution changes in a particular way. As such, causal graphical models constitute a richer model class than purely probabilistic graphical models. Unsurprisingly, learning causal models from data is a challenging task and always relies on some underlying assumptions. We return to this point in more detail in Chapter 2.

1.1.1. Robustness and interpretability

In the following, we outline a few challenges related to stability, robustness and interpretability which can potentially be addressed using causal models and reasoning.

Statistical learning and distributional robustness As learning causal models from data is difficult, most current machine learning methods rely on exploiting statistical associations only. While those models do not make causal claims, their desiderata include properties like stability under interventions or “domain changes” in the input data distribution, and often their predictions need to be suitable for decision making. The latter presupposes that the model allows to predict what happens under changes or interventions in the system—something only a causal model can do. Learning purely based on statistical associations can lead to an array of problems and failures as the following example demonstrates. We use the API offered by Clarifai (`https://www.clarifai.com/`) to classify the two images of cows, shown in Figure 1.3(a) and Figure 1.3(d), respectively. The first image shows a cow on green pasture, an environment we would consider being its natural habitat. The second image shows a cow on a beach—where they are typically not observed. Nonetheless, in both images the cows are in the foreground of the respective image and clearly recognizable as such. Yet, in the second case the image classification system fails to output the label “cow” in the top 20 predictions. Furthermore, it does not output related labels such as “mammal”, “cattle” or “animal”. This example demonstrates that a powerful visual object recognition system is not robust to a perturbation or domain change which here consists of a



(a)

PREDICTED CONCEPT	PROBABILITY
cow	0.995
pasture	0.991
grass	0.987
no person	0.985
mammal	0.983
pastoral	0.977
milk	0.975
cattle	0.972
rural	0.971
livestock	0.969

(b)

agriculture	0.969
nature	0.964
hayfield	0.955
beef cattle	0.955
grassland	0.947
landscape	0.944
farm	0.943
countryside	0.943
farmland	0.941
animal	0.939

(c)

(d)

PREDICTED CONCEPT	PROBABILITY
beach	0.987
water	0.980
sea	0.979
sand	0.977
ocean	0.966
seashore	0.949
summer	0.928
travel	0.908
no person	0.879
island	0.839

(e)

sky	0.836
vacation	0.830
tropical	0.829
wave	0.806
nature	0.806
surf	0.790
sun	0.774
landscape	0.753
outdoors	0.693
seascape	0.679

(f)

Figure 1.3.: Panels (b) and (c) show the classifications along with their estimated probabilities, when presenting the image shown in panel (a). The label “cow” is correctly returned as the most likely one. Panels (e) and (f) show the classifications along with their estimated probabilities, when presenting the image shown in panel (d)—a cow outside of its usual habitat. Neither the label “cow” nor labels denoting related concepts appear among the top 20 predictions. Retrieved on May 1st, 2018 using `https://www.clarifai.com/demo`. The example is attributed to Pietro Perona.

background not observed (or rarely observed) during training. In other words, the image showing the cow on the beach comes from a distribution which differs from the training distribution and the system does not perform well on such examples. However, generally speaking, we would also like to achieve good performance on test distributions that differ—up to a reasonable degree—from the training distribution. Informally, this is the aim of distributional robustness. Next, we briefly discuss the relation between distributional robustness and causal models.

Causal models, predictive accuracy and distributional robustness The example shown in Figures 1.1 and 1.2 illustrates one of the defining advantages of a causal model. It is robust to interventions in the system: we can intervene on all predictors except for the target itself and the causal model remains valid. This is a strong guarantee but may come at the price of predictive accuracy. In many cases, it is advantageous for predictive performance to use other variables as predictors, too. For instance, often the mutual information between a causal child of the target and the target itself is large which can be exploited for prediction. At the same time, the set of possible perturbations and changes in the system of interest might not be arbitrary and exploiting knowledge of what sort of changes are possible might yield an interesting trade-off between the two regimes of (a) only using the causal parents as predictors and (b) allowing for an arbitrary set of input variables. One approach to address the brittleness of statistical models to domain changes is to use distributionally robust inference in favor of ordinary empirical risk minimization. More formally, let $Y \in \mathcal{Y}$ be a target of interest. Typically $\mathcal{Y} = \mathbb{R}$ for regression or $\mathcal{Y} = \{1, \ldots, K\}$ in classification with K classes. Let $X \in \mathcal{X}$ with $\mathcal{X} = \mathbb{R}^p$ be a predictor. The prediction $\hat{y}$ for y, given $X = x$, is of the form $f_\theta(x)$ for a suitable function f_θ with parameters $\theta \in \mathbb{R}^d$. For regression, $f_\theta(x) \in \mathbb{R}$, whereas for classification $f_\theta(x)$ corresponds to the conditional probability distribution of $Y|X = x$ for $Y \in \{1, \ldots, K\}$. Let ℓ be a suitable loss that maps y and $\hat{y} = f_\theta(x)$ to $\mathbb{R}^+$. Let (x_i, y_i) for $i = 1, \ldots, n$ be the sample from an unknown training distribution over $\mathcal{X} \times \mathcal{Y}$ with density $P(X, Y)$. $\hat{y}_i = f_\theta(x_i)$ denotes the prediction for y_i. In standard empirical risk minimization, the goal is to minimize the expected loss or risk

$$\operatorname{argmin}_\theta \; E_{P(X,Y)}\Big[\ell(Y, f_\theta(X))\Big]$$

which is approximated by the empirical loss as

$$\hat{\theta} = \operatorname{argmin}_{\theta} \frac{1}{n} \sum_{i=1}^{n} \Big[\ell\big(y_i, f_\theta(x_i)\big) \Big] + \gamma \cdot \operatorname{pen}(\theta).$$

The term $\operatorname{pen}(\theta)$ is a complexity penalty to prevent overfitting, for example a ridge term $\|\theta\|_2^2$ on the parameters of the model. In empirical risk minimization, the implicit assumption is that the test data follows the same distribution as the training data. Distributionally robust inference, in contrast, allows for a distribution shift between training and test data. Specifically, in distributionally robust inference the aim is to learn

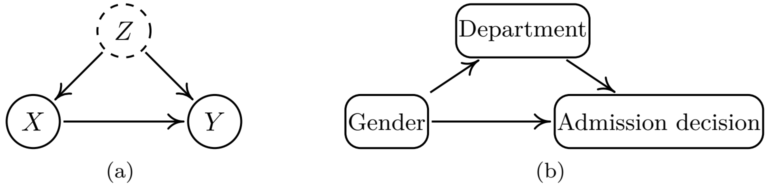
$$\operatorname{argmin}_{\theta} \sup_{F \in \mathcal{F}} E_F(\ell(Y, f_\theta(X)))$$

for a given set $\mathcal{F}$ of distributions. The set $\mathcal{F}$ is the set of distributions on which one would like the estimator to achieve a guaranteed performance bound. Using a causal framework, we can define the set $\mathcal{F}$ of distributions as being induced by interventions on variables in the causal model (Heinze-Deml and Meinshausen, 2017b; Rothenhäusler et al., 2018a). For instance, one may try to achieve robustness against a set of distributions that are generated by certain kinds of interventions on a specific subset of variables in the causal generative model. This may allow to obtain the trade-off between the two regimes mentioned above as we can model the set of interventions that can be realistically expected to occur while excluding implausible ones. This approach may improve predictive accuracy in contrast to allowing for arbitrary changes in the system.

To reiterate, in distributional robustness we are interested in achieving good performance for a set of distributions $\mathcal{F}$. In broad terms, this differs from "classical" robust statistics as follows. In robust statistics, the training data is assumed to consist of observations coming from two different distributions, P and Q, where $(1-\varepsilon)n$ observations come from P, εn observations are generated by Q and ε is assumed to be small. The distribution of interest is P, in the sense that we want to minimize the expected loss

$$\operatorname{argmin}_{\theta} E_{P(X,Y)} \Big[\ell(Y, f_\theta(X)) \Big]$$

where the expectation is with respect to P only, even though some training observations come from Q (Hampel et al., 2005).



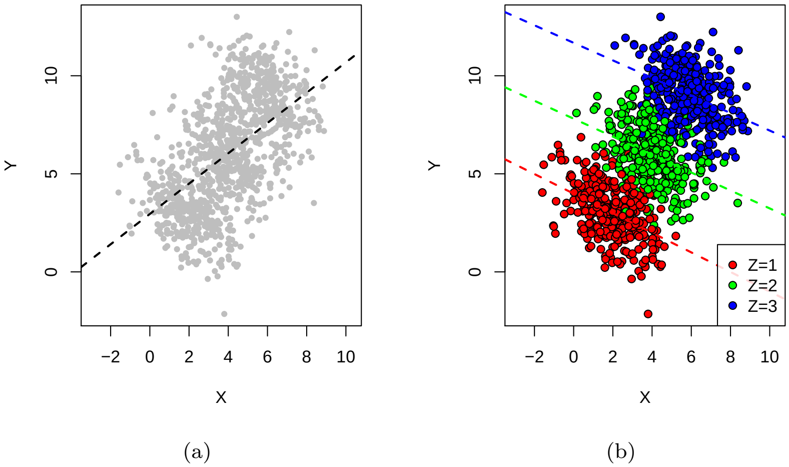
(a) (b)

Figure 1.4.: Panel (a): X and Y are both caused by the unobserved variable Z. The common cause Z is called a confounder. Panel (b): Gender influences the department choice and potentially the admission decision. To detect the existence of a gender bias, the direct effect of gender on the admission decision needs to be assessed. If we fail to condition on department choice in the analysis, the total effect of gender on admission is estimated.

Causality and observational data bias Robustness properties constitute just one attractive property of causal methods and we now turn to further important advantages of causal models compared to purely statistical approaches which relate to analyzing and understanding biases in observational data.

Traditionally, in statistics the methods for establishing causal relations rely on carefully designed randomized studies. However, in many real-life applications we do not have sufficient control over the system to perform such experiments. Problems like the following can only be solved meaningfully by causal inference: What would need to be changed in a developing country to have the fertility rate drop to Western levels? What is the phenotypical change if some genes in a cell are knocked-out? Yet, genes cannot be randomly assigned to different groups of people and factors like ‘infant mortality rate’ are highly complex and cannot be manipulated in isolation. In other cases, the required experiments would be unethical, e.g. if they expose an experimental group to dangers. This gives rise to the need for causal inference from observational data. As many of today’s training data sets were not collected through carefully designed experiments but do in fact consist of observational data, they are subject to a number of biases and heterogeneities that do not fit into the classical statistical framework of having i.i.d. data from a carefully designed randomized control trial.

To illustrate one common problem of this kind, we now look at confounding in more detail. Confounding occurs when two variables of interest X and Y are both caused by a third variable Z (cf. Figure 1.4(a)). The common



(a) (b)

Figure 1.5.: Panel (a) shows the linear regression line when not accounting for the hidden common cause Z: X and Y are positively correlated. In panel (b) we see that the slope of the linear regression fit is reversed when accounting for the hidden common cause Z.

cause Z is called a confounder. When the causal graph has this structure, it is important to condition on Z when assessing the influence of X on Y. Figure 1.5 shows a toy dataset generated from a graph where X and Y are both caused by Z and X causes Y. In Figure 1.5(a), Z is not accounted for in the regression of Y on X, indicating a positive causal effect of X on Y (presupposing that X is a parent of Y). When conditioning on Z, we observe a sign flip. Now, the causal effect of X on Y seems to be negative.

In practice, the literature on adjustment (e.g. Pearl (2009)) treats what covariates need to be conditioned on to estimate causal effects accurately. For this, it is crucial to know the true underlying causal graph structure. One famous real-world example for this are the 1973 UC Berkeley graduate admissions. When not accounting for the department an applicant applied for, the data suggests the existence of a bias against female applicants. When conditioning on the former, however, it becomes apparent that female applicants tend to apply to more competitive departments.

This explains the bias suggested when looking at the aggregated data and overall there is even small bias in favor of women (Bickel et al., 1975). Figure 1.4(b) shows one plausible causal graph for this setting. As the graph structure shows, this is not an example of confounding. For the total causal effect no adjusting is necessary. However, to detect the existence of a gender bias, the direct effect of gender on the admission decision needs to be assessed, i.e. the part of the total effect that is not mediated via other variables. Cases where conditioning as opposed to not conditioning on other variables lead to sign flips of the estimated causal effects are instances of Simpson's paradox (Simpson, 1951). For another example, see for instance Peters et al. (2017, Example 6.16). Phenomena like Simpson's paradox or confounding are even more challenging when the causal graph structure is unknown, common causes are unobserved or, as we discuss next, when the input variables lack interpretability, e.g. if they consist of measurements such as pixel intensities.

Interpretability In a classical statistical setting, the starting point is often a dataset which consists of a number of input variables having a semantic meaning. The analyst can then reason about the underlying causal graph or estimate it from the data under suitable assumptions. When working with data such as images, text or speech, the individual input variables consist of "raw perceptual data". Higher-level concepts are now latent variables to be inferred by the statistical method. While this challenge is not unique to deep learning, the paradigm of learning models "end-to-end" from raw data, using as little imposed structure as possible, has become very popular in machine learning with the surge of deep learning (Bottou, 2018). CNNs perform so-called "automatic feature extraction", meaning that their input is raw data—e.g. raw pixel values—from which the model extracts features in the various layers of the neural network. As neural networks excel at prediction tasks, one is easily drawn to believe that the representations learnt by the various network layers capture the relevant features to which we can attribute a semantic meaning. However, since the model is learnt end-to-end, it is intransparent what latent features are extracted and they cannot be accessed in isolation. While there is a growing literature on attempting to explain the meaning of features extracted by neural networks (e.g. Olah et al. (2017) and references therein), interpretability remains a challenging problem. In particular, the problem of confounding discussed above may exacerbate in case of automatic feature extraction from raw data. Even if all latent variables manifest themselves in the input data (e.g. images), it is unclear how they are used by the model and therefore, there is no possibility to adjust for the correct set of variables.

1.2. Computation

In addition to the areas outlined above, parts of this thesis study the following computational questions.

Data efficiency We start with the observation that visual object recognition is based on “training on more images than a human can see”, machine translation relies on “training on more text than a human can read” and playing Atari games requires “playing more games than any teenager can endure” (Bottou, 2018). In other words, the state-of-the-art machine learning systems referenced above require very large amounts of training data. Humans, on the other hand, are able to learn invariances from a few instances of the same object only. This observation gives rise to two questions: How can we make methods more efficient in terms of the required samples? And how can we design algorithms that scale with large datasets?

Large-scale data While many modern machine learning methods require large amounts of training data, larger and larger amounts of data are in fact collected across industry and scientific disciplines. Iterative and stochastic machine learning methods allow to handle cases where the input data does not fit into memory of one machine. However, in some applications the input data is even too large to fit into the storage of one machine. This necessitates the development of distributed machine learning methods.

Privacy Last but not least, as more data is collected and stored across industries and institutions, important questions of privacy arise. For instance in medical applications, highly personal data is collected which can be invaluable for scientific discovery. At the same time, from a privacy standpoint it may be unacceptable to share this data in undisguised form. The statistical challenge is to develop methods that guard an individual’s privacy while sacrificing as little useful information contained in the data as possible.

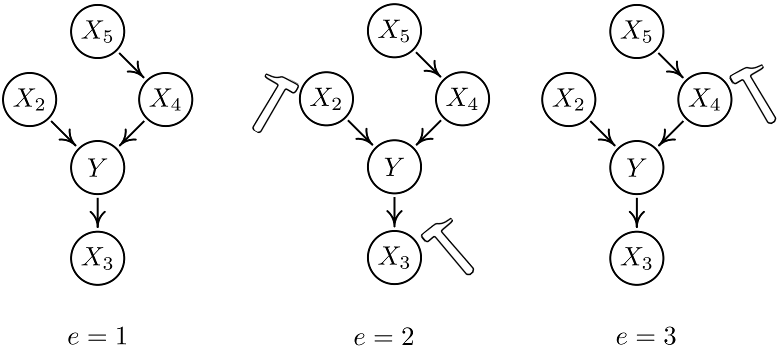


Figure 1.6.: Multiple environments: The environment is discrete and not random.

1.3. Learning from partitioned data

In statistics and machine learning, one often presupposes the existence of a sample of training observations which are independently and identically distributed. Furthermore, it is often assumed that the test data will again follow the same distribution. As we have already noted above, this classical setting of working with i.i.d. datasets does not apply in many real-world applications. Heterogeneities in the data can arise in a multitude of different ways and often the available data is partitioned in some form, yielding a joint dataset whose observations may not be independently and identically distributed. While this seems cumbersome at first, heterogeneous data can also represent an opportunity for inference as the differences between the partitions or groups can be informative. Below we outline the settings considered throughout this thesis.

Multiple environments When data has been recorded in different domains or “environments”, it is typically “horizontally-partitioned”. This means that we observe the same set of variables or the same data type, say images, in different conditions. For instance in biology, these can arise through explicit experimentation. In other settings, external changes in the system may take place, lying outside of the control of the analyst. Of course, a combination of both experimental and observational data is also possible. Furthermore, there can be time shifts in the distribution or the distribution of unobserved confounders changes between different settings

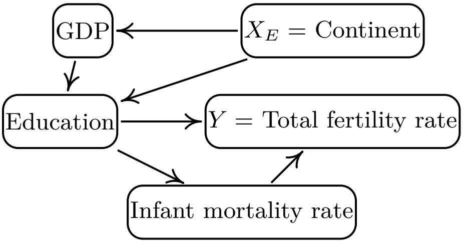


Figure 1.7.: Multiple environments: The environment arises through a node in the causal graph. The true causal graph is unknown; here we show one potential causal graph for the considered problem setting.

or environments. In many cases, the heterogeneities in the different data partitions can be modeled as interventions in the causal graph describing the system under study. For instance, as illustrated in Figure 1.6, the different environments can be discrete. In each environment, a different set of interventions occurred in the causal graph, giving rise to different distributions of the observed variables. Figure 1.7 is taken from the analysis of a real-world example studied in Chapter 4. It shows one potential causal graph for the considered problem setting and illustrates a different way to conceptualize how data from multiple environments arises. Here, we consider the problem of total fertility rate modeling where data from different countries is available. The continent the country is located on can be chosen to encode the environment (while other choices may also be suitable). We model this by treating the environment as a variable X_E that belongs to the (unknown) causal graph to be inferred. In general, this formulation allows for discrete as well as continuous environments.

Data grouped by identity Another type of partitioning consists of grouping observations by the identity of the underlying object or person. For instance, when considering images of individuals, we can group those images together that show the same person. We can then exploit information how observations within a group differ compared to observations from different groups. In other words, we can split the total variance in the data into within-group and between-group components. In some settings, it may be plausible that directions associated with a large within-group or between-group variance might be subject to distributional changes in the future. We can then penalize the estimation procedure to achieve robust-

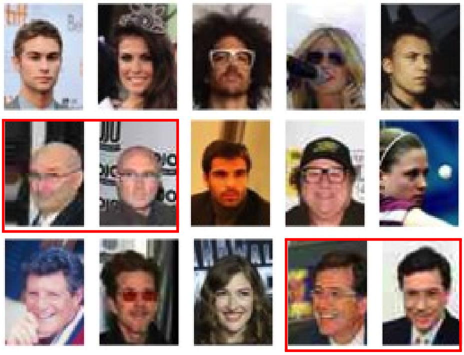


Figure 1.8.: Data grouped by identity: The data show grouped observations in two red boxes. These observations show the same person.

ness with respect to such perturbations.

Vertically-partitioned data At the beginning of §1.1.1 we noticed that complex machine learning methods require large amounts of training data. While such datasets are available to some institutions and companies today, this availability cannot be presupposed in all domains. Oftentimes data is collected by different parties and therefore partitioned in various ways. It may then be "horizontally-partitioned" or it can be "vertically-partitioned". The latter implies that different data owners hold different sets of features but have information about the same set of observations or individuals. This is illustrated in Figure 1.9. Learning a joint model is often desirable as this allows to adjust for variables held by different parties, potentially accounting for confounders. However, privacy concerns may restrict the sharing of the data in undisguised form.

1.4. Outline and contributions

Chapter 2 introduces causal concepts such as interventions and counterfactuals, and reviews recently proposed causal structure learning algorithms. We compare their underlying assumptions and perform a simulation study for an empirical evaluation.

Chapter 3 proposes BACKSHIFT, a method that learns linear causal cyclic

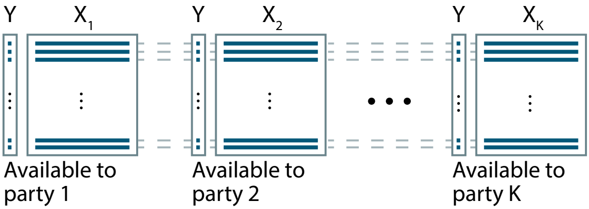


Figure 1.9.: Vertically partitioned data: Each party holds a subset of the total number of features, containing the data from the same set of individuals. Some or all parties have access to Y.

models from data collected in multiple environments, arising through shift interventions. backShift exploits invariances in second moments and allows for the presence of latent confounders.

Chapter 4 extends Invariant Causal Prediction (ICP), proposed in Peters et al. (2016), to nonlinear models. Again, we rely on data collected in different environments. We exploit invariant conditional distributions to estimate the causal parents of a target variable of interest. (Nonlinear) ICP has the guarantee of controlling the type-I-error rate α, i.e. the probability of wrongly declaring a set of variables to have a direct effect on the target is bounded by α.

Chapter 5 proposes conditional variance regularization (CoRe) to achieve distributional robustness with respect to domain shifts arising through interventions on latent "style" features. Here, we exploit observations that are grouped by the identity of the underlying object or person. CoRe penalizes directions in the feature space where the within-group variance is non-zero as the distribution of these, conditional on the target of interest, is believed to change in the future.

Chapters 6 and 7 address estimation in a distributed computation setting. Specifically, the data is vertically partitioned and the proposed methods—Dual-Loco and PriDE—make use of random projections to allow for a one-shot communication scheme between the different machines or parties. PriDE extends Dual-Loco by additionally considering privacy constraints. In this context, the notion of (ϵ, δ)-distributed differential privacy is introduced.

1.4.1. Publications

This thesis is cumulative and contains the manuscripts listed in Table 1.1. We indicate the associated chapters of this dissertation and also whether the manuscript has been published.

---

[1] Christina Heinze-Deml and Dominik Rothenhäusler are joint first authors of Rothenhäusler et al. (2015) and contributed equally. Christina Heinze-Deml wrote the implementation of the algorithm, along with the R package `backShift`, and conducted all experiments. Together with Jonas Peters and Nicolai Meinshausen, Christina Heinze-Deml wrote the main text. Dominik Rothenhäusler's main contributions were the theoretical result regarding identifiability, the methodological result that the cycle-product assumption guarantees uniqueness of the estimator, and the algorithmic result that the cycle-product restriction can be cast as a linear program.

Manuscript	used in
C. Heinze-Deml, M. H. Maathuis, and N. Meinshausen. "Causal Structure Learning". In: *Annual Review of Statistics and Its Application* 5.1 (2018), pp. 371–391	Chapter 2
D. Rothenhäusler, C. Heinze, J. Peters, and N. Meinshausen. "backShift: Learning causal cyclic graphs from unknown shift interventions". In: *Advances in Neural Information Processing Systems 28 (NIPS).* 2015, pp. 1513–1521[1]	Chapter 3
C. Heinze-Deml, J. Peters, and N. Meinshausen. "Invariant Causal Prediction for Nonlinear Models". In: *Journal of Causal Inference* 6 (2 2018)	Chapter 4
C. Heinze-Deml and N. Meinshausen. "Conditional Variance Penalties and Domain Shift Robustness". In: *arXiv preprint arXiv:1710.11469* (2017) *Submitted.*	Chapter 5
C. Heinze, B. McWilliams, and N. Meinshausen. "DUAL-LOCO: Distributing Statistical Estimation Using Random Projections". In: *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics.* PMLR, 2016, pp. 875–883	Chapter 6
C. Heinze-Deml, B. McWilliams, and N. Meinshausen. "Preserving Privacy Between Features in Distributed Estimation". In: *Stat.* 7 (1 2018)	Chapter 7

Table 1.1.: Manuscripts contained in this thesis.

Chapter 2.

Causal structure learning

Graphical models can represent a multivariate distribution in a convenient and accessible form as a graph. Causal models can be viewed as a special class of graphical models that not only represent the distribution of the observed system but also the distributions under external interventions. They hence enable predictions under hypothetical interventions, which is important for decision making. The challenging task of learning causal models from data always relies on some underlying assumptions. We discuss several recently proposed structure learning algorithms and their assumptions, and compare their empirical performance under various scenarios.

2.1. Introduction

A graphical model is a family of multivariate distributions associated with a graph, where the nodes in the graph represent random variables and the edges encode allowed conditional dependence relationships between the corresponding random variables (Lauritzen, 1996). A *causal* graphical model is a special type of graphical model, where edges are interpreted as direct causal effects. This interpretation facilitates predictions under arbitrary (unseen) interventions, and hence the estimation of causal effects e.g. Pearl (2009), Spirtes et al. (2000), and Wright (1934). This ability to make predictions under arbitrary interventions sets causal models apart from standard models. We refer to Didelez (2017) for an introductory overview of causal concepts and graphical models.[1]

---

[1] Causal inference is also possible without graphs, using for example the Neyman-Rubin potential outcome model (e.g., Rubin, 2005). Single world intervention graphs (SWIGs) (Richardson and Robins, 2013) provide a unified framework for potential

Structure learning is a model selection problem in which one estimates or learns a graph that best describes the dependence structure in a given data set (Drton and Maathuis, 2017). *Causal* structure learning is the special case where one tries to learn the causal graph or certain aspects of it, and this is what we focus on in this paper. We describe various algorithms that have been developed for this purpose under different assumptions. We then compare the algorithms in a simulation study to investigate their performances in settings where the assumptions of a particular method are met, but also in settings where they are violated.

The outline of the paper is as follows. §2.2 discusses the basic causal model and its various assumptions. §2.3 describes different target graphical objects, such as directed acyclic graphs or equivalence classes thereof, and describes algorithms that can learn them under certain assumptions. §2.4 describes the simulation set-up, the evaluation scheme, and the results. We close with a brief discussion in §2.5.

2.2. The model

We formulate the model as a structural causal model (Pearl, 2009). In particular, we consider a linear structural equation model (e.g., Wright, 1921) for a p-dimensional random variable $X = (X_1, \dots, X_p)^t$ under noise contributions $\varepsilon = (\varepsilon_1, \dots, \varepsilon_p)^t$:

$$X_j \leftarrow \sum_{k=1}^{p} \beta_{j,k} X_k + \varepsilon_j \qquad \text{for } j = 1, \dots, p, \tag{2.1}$$

or in vector notation,

$$X \leftarrow BX + \varepsilon, \tag{2.2}$$

where B is a $p \times p$ matrix with entries $B_{j,k} = \beta_{j,k}$. Thus, the distribution of X is determined by the choice of B and the distribution of ε.

This model is called *structural* since it is interpreted as the generating mechanism of X (emphasized by the assignment operator $\leftarrow$), where each structural equation is assumed to be invariant to possible changes in the other structural equations. This is also referred to as *autonomy* (Frisch, 1938; Haavelmo, 1944). This assumption is key for causality, since it

---

outcome and graphical approaches to causality.

allows the derivation of the distribution of X under external interventions. For example, a gene knockout experiment can be modeled by replacing the structural equation of the relevant gene, while keeping the other structural equations unchanged. If the gene knockout experiment has significant off-target effects (e.g., Cho et al., 2014), then this approach is problematic with respect to the autonomy assumption. A possible remedy consists of modeling the experiment as a simultaneous intervention on all genes that are directly affected by the experiment.

2.2.1. Interventions

In this paper, we consider the following two types of interventions:

(a) A do-intervention (also called “surgical" intervention): This intervention is modeled by replacing the structural equation

$$X_j \leftarrow \sum_{k=1}^{p} \beta_{j,k} X_k + \varepsilon_j \quad \text{by} \quad X_j \leftarrow Z_j,$$

where Z_j is the (either deterministic or random) value that variable X_j is forced to take in this intervention.

(b) An additive intervention (also called “shift” intervention): This intervention consists of adding additional noise, modeled by replacing the structural equation

$$X_j \leftarrow \sum_{k=1}^{p} \beta_{j,k} X_k + \varepsilon_j \quad \text{by} \quad X_j \leftarrow \sum_{k=1}^{p} \beta_{j,k} X_k + \varepsilon_j + Z_j,$$

where Z_j is the additional noise (again either deterministic or random) that is added to variable X_j. Shift interventions are standard in the econometric literature on instrumental variables with binary treatments where the additive shift of an exogenous instrument changes the probability of a binary treatment variable (Angrist et al., 1996). Shift interventions are also natural in biological settings where an inhibitor or enhancer can amplify or decrease the presence of, for example, mRNA in a cell. If the concentrations are amplified by a fixed factor, then this corresponds to an additive shift in the log-concentrations.

2.2.2. Graphical representation

We can represent the model defined in (2.1) as a directed graph G, where each variable X_k is represented by a node k, $k = 1, \ldots, p$, and there is an edge from node k to node j ($k \neq j$) if and only if $\beta_{j,k} \neq 0$. Thus, the parents $pa(j, G)$ of node j in G correspond to the random variables that appear on the right hand side of the jth structural equation. In other words, $X_{\mathrm{pa}(j,G)} := \{X_i : i \in pa(j, G)\}$ are the variables that are involved in the generating mechanism of X_j and are also called the *direct causes* of X_j (with respect to $X_1, \ldots, X_p$). In this sense, edges in G represent direct causal effects and G is also called a *causal graph*. The nonzero $\beta_{j,k}$'s can be depicted as edge weights of G, yielding a weighted graph. This weighted graph and the distribution of ε fully determine the distribution of X.

The graph G is called *acyclic* if it does not contain a cycle[2]. A directed acyclic graph is also called a DAG. A directed graph is acyclic if and only if there is an ordering of the variables, called a *causal order*, such that the matrix B in Eq. (2.2) is triangular. In terms of the causal mechanism, acyclicity means that there are no feedback loops. We refer to §2.2.5 for more details on cycles.

2.2.3. Factorization and truncated factorization

If $\varepsilon_1, \ldots, \varepsilon_p$ are jointly independent and G is a DAG, then the probability density function $f(\cdot)$ of X factorizes according to G:

$$f(x) = f(x_1, \ldots, x_p) = \prod_{i=1}^{p} f(x_i | x_{pa(i,G)}). \tag{2.3}$$

Moreover, f is then called *Markov* with respect to G. This means that for pairwise disjoint subsets A, B and S of V ($S = \emptyset$ is allowed) the following holds: if A and B are separated by S in G according to a graphical criterion called d-separation (Pearl, 2009), then X_A and X_B are conditionally independent given X_S in f.

One can model an intervention on X_j by replacing the conditional density $f(x_j | x_{\mathrm{pa}(j)})$ by its conditional density under the intervention, keeping the other terms unchanged. For example, a do-intervention on X_j yields the

---

[2] A cycle (sometimes also called directed cycle) is formed by a directed path from i to j together with the edge $j \to i$.

following factorization:

$$f(x|do(x_j)) = g(x_j) \prod_{i=1, i \neq j}^{p} f(x_i|x_{\mathrm{pa}(i)}),$$

where $g(\cdot)$ is the density of Z_j (allowed to be a point mass). When intervening on several variables simultaneously, one simply conducts such replacements for all intervention variables. The resulting factorization is known as the g-formula (Robins, 1986), the manipulated density (Spirtes et al., 2000), or the truncated factorization formula (Pearl, 2009).

2.2.4. Counterfactuals

We note that the structural causal model is often discussed in the context of counterfactual outcomes. In particular, if one assumes that ε is identical under different interventions, the model defines a joint distribution on all possible counterfactual outcomes. The problematic aspect is clearly that the realizations of the noise under different interventions can never be observed simultaneously and any statement about the joint distribution of the noise under different interventions is thus in principle unfalsifiable and untestable (Dawid, 2000). Without assuming anything on the joint noise distributions under different interventions, a causal model can equivalently be formulated via structural equations, a graphical model, or potential outcomes (Imbens, 2014; Richardson and Robins, 2013). For the causal structure learning methods discussed in this paper, no assumption on the joint noise distribution is necessary and we chose to use the structural equation framework for ease of exposition.

2.2.5. Assumptions

We will consider various assumptions for the model defined by Eq. (2.2):

Causal sufficiency. Causal sufficiency refers to the absence of hidden (or latent) variables (Spirtes et al., 2000). There are two common options for the modeling of hidden variables[3]: They can be modeled explicitly as

---

[3] In this manuscript we look at the behavior of various methods under the presence and absence of latent confounding. Throughout, we do not allow hidden selection variables, that is, unmeasured variables that determine if a unit is included in the data sample. More details on selection variables can be found in, e.g., Spirtes et al. (1999).

nodes in the the structural equations, or they can manifest themselves as a dependence between the noise terms $(\varepsilon_1, \ldots, \varepsilon_p)$, where the noise terms are assumed to be independent in the absence of latent confounding.

Causal faithfulness. We saw in §2.2.3 that the distribution of X generated from Eq. (2.2) is Markov with respect to the causal DAG, meaning that if A and B are d-separated by S in the causal DAG, then X_A and X_B are conditionally independent given X_S. The reverse implication is called causal faithfulness. Together, the causal Markov and causal faithfulness assumptions imply that d-separation relationships in the causal DAG have a one-to-one correspondence with conditional independencies in the distribution.

Acyclicity. Cycles can be used to model instantaneous feedback mechanisms. In the presence of cycles, the structural equations (2.1) are typically interpreted (implicitly) as a dynamical system. There are various assumptions that can be made about the strength of cycles in the graph[4]:

(i) Existence of a unique equilibrium solution of Eq. (2.2). Is there a unique solution X for each realization ε such that $X = BX + \varepsilon$ or, equivalently, $(I - B)X = \varepsilon$, where I is the p-dimensional identity matrix. Existence of a unique equilibrium requires that $I - B$ is invertible. In this case the equilibrium is

$$X = (I - B)^{-1}\varepsilon.$$

(ii) Convergence to a stable equilibrium. Iterating Eq. (2.2) from any starting value $X^{(0)}$ for X (and for a fixed and constant realization of the noise ε), we can form an iteration $X^{(k)} = BX^{(k-1)} + \varepsilon$ for $k \in \mathbb{N}$. The question is then whether the iterations converge to the equilibrium, that is, whether $\lim_{k\to\infty} X^{(k)} = (I - B)^{-1}\varepsilon$. This convergence requires that the spectral radius of B is smaller than 1.

(iii) Existence of a stable equilibrium under do-interventions. This requires in addition that the cycle product (the maximal product of the coefficients along all loops in the graph) is smaller than 1, see for example Rothenhäusler et al. (2015).

DAGs fulfill all three assumptions (i)-(iii) above trivially as their spectral radius and cycle product both vanish identically.

---

[4] We exclude self-loops (an edge from a node to itself), as models would be unidentifiable if self-loops were allowed (see, e.g., Rothenhäusler et al., 2015).

Gaussianity of the noise distribution. We consider both Gaussian distributions and t-distributions with various degrees of freedom.

One or several experimental settings. We consider both homogeneous data, where all observations are from the same experimental setting, and heterogeneous data, where the observations come from different experimental settings. In particular, we consider settings with unknown shift-interventions and known do-interventions.

Linearity. While the assumptions and the models have been discussed in the context of linear models, the ideas can be extended to nonlinear models and to discrete random variables to various degrees.

2.3. Methods

Since different structure learning methods output different types of graphical objects, we first discuss the various target graphical objects in §2.3.1. To conduct a comparison based on such different graphical targets, we focus on certain ancestral relationships that can be read off from all objects (see §2.3.2). The different algorithms and their assumptions are discussed in §2.3.3, and their assumptions are summarized in Table 2.1.

2.3.1. Target graphical objects

The structure learning methods that we will compare use different types of data, from purely observational data to data with clearly labelled interventions, from not allowing hidden variables and cycles to allowing both of these. As a result, the different methods learn the underlying causal graph at different levels of granularity. At the finest level of granularity, a method learns the underlying *directed graph* (DG) from Eq. (2.1). If the method assumes acyclicity (no feedback), then the target object is a *directed acyclic graph* (DAG).

Under the model of Eq. (2.2) with acyclicity, independent and multivariate Gaussian errors and i.i.d. observational data, the underlying causal DAG is generally not identifiable. Instead, one can identify the Markov equivalence class of DAGs, that is, the set of DAGs that encode the same set of d-separation relationships (Pearl, 2009). A Markov equivalence can be conveniently summarized by another graphical object, called a *completed partially directed acyclic graph* (CPDAG) (Andersson et al., 1997; Chickering, 2002a). A CPDAG can be interpreted as follows: $i \rightarrow j$ is in the CPDAG if $i \rightarrow j$ in every DAG in the Markov equivalence class, and $i \circ\!\!-\!\!\circ j$ in the CPDAG if there is a DAG with $i \rightarrow j$ and a DAG with $i \leftarrow j$ in the Markov equivalence class. Thus, edges of the type $\circ\!\!-\!\!\circ$ represent uncertainty in the edge orientation.

DAGs are not closed under marginalization. In the presence of latent variables, some algorithms therefore aim to learn a different object, called a *maximal ancestral graph* (MAG) (Richardson and Spirtes, 2002). In general, MAGs contain three types of edges: $i - j$, $i \rightarrow j$ and $i \leftrightarrow j$, but in our settings without selection variables (see footnote 3), $i - j$ does not occur. A MAG encodes conditional independencies via m-separation (Richardson and Spirtes, 2002). Every DAG with latent variables can be uniquely mapped to a MAG that encodes the same conditional independencies and the same ancestral relationships among the observed variables. Ancestral relationships can be read off from the edge marks of the edges: a tail mark $i {-}\!\!* j$ means that i is an ancestor of j in the underlying DAG, and an arrowhead $i {\leftarrow}\!\!* j$ means that i is not an ancestor of j in the underlying DAG, where $*$ represents any of the possible edge marks (again assuming no selection variables).

Several MAGs can encode the same set of conditional independence relationships. Such MAGs form a Markov equivalence class, which can be represented by a *partial ancestral graph* (PAG) (Ali et al., 2009; Richardson and Spirtes, 2002). A PAG can contain the following edges: $i \rightarrow j$, $i - j$, $i -\!\!\circ j$, $i \leftrightarrow j$, $i \circ\!\!\rightarrow j$, and $i \circ\!\!-\!\!\circ j$, but the edges $i - j$ and $i -\!\!\circ j$ do not occur in our setting without selection variables. The interpretation of the edge marks is as follows. A tail mark means that this tail mark is present in all MAGs in the Markov equivalence class, and an arrowhead means that this arrowhead is present in all MAGs in the Markov equivalence class. A circle mark represents uncertainty about the edge mark, in the sense that there is a MAG in the Markov equivalence class where this edge mark is a tail, as well as a MAG where this edge mark is an arrowhead.

2.3.2. Ancestral and parental relationships

To compare methods that output the different graphical objects discussed above, we focus on the following two basic questions for any variable X_j,

$j \in \{1, \ldots, p\}$, and the underlying causal DAG G:

(a) What are the direct causes of X_j, or equivalently, what is $\mathrm{pa}_G(j)$? The parents are important, since they completely determine the distribution of X_j. Hence, the conditional distribution $X_j | X_{\mathrm{pa}(j)}$ is constant, even under arbitrary interventions on subsets of $X_{\{1,\ldots,p\}\setminus\{j\}}$. The set of parents is unique in this respect and allows to make accurate predictions about X_j even under arbitrary interventions on all other variables. Moreover, the (possible) parents of X_j can be used to estimate (bounds on) the total causal effect of X_j on any of the other variables (Maathuis et al., 2009, 2010; Nandy et al., 2017a; Stekhoven et al., 2012).

(b) What are the causes of X_j, or equivalently, what is the set of ancestors $\mathrm{an}_G(j)$ (the set of nodes from which there is a directed path to j in G)? The ancestors are important, since any intervention on ancestors of X_j has an effect on the distribution of X_j, as long as no other do-type interventions happen along the path. Thus, if we want to manipulate the distribution of X_j, we can consider interventions on subsets of $X_{\mathrm{an}_G(j)}$.

2.3.3. Considered methods

We include at least one algorithm from each of the following five main classes of causal structure learning algorithms: constraint-based methods, score-based methods, hybrid methods, methods based on structural equation models with additional restrictions, and methods exploiting invariance properties. Limiting ourselves to algorithms with an implementation in R (R Core Team, 2017), we obtain the following selection of methods, with assumptions summarized in Table 2.1:

- Constraint-based methods: PC (Spirtes et al., 2000), rankPC (Harris and Drton, 2013), FCI (Spirtes et al., 2000), and rankFCI[5]
- Score-based methods: GES (Chickering, 2002b), rankGES (Nandy et al., 2017b), GIES (Hauser and Bühlmann, 2012), and rankGIES[6]
- Hybrid methods: MMHC (Tsamardinos et al., 2006)
- Structural equation models with additional restrictions: LiNGAM (Shimizu et al., 2006)

---

[5] rankFCI is obtained by using rank correlations in FCI, analogously to rankPC.

[6] rankGIES is obtained by using rank correlations in GIES, analogously to rankGES.

Table 2.1.: The assumptions (see §2.2.5) and output format for the different methods. (For example, PC requires acyclicity, causal faithfulness and causal sufficiency, and LiNGAM requires non-Gaussian errors.) Please note that linearity is not explicitly listed, but all versions of the algorithms based on rank-correlations allow certain types of nonlinearities. The different output formats are: DG (directed graph), DAG (directed acyclic graph), PDAG (partially directed acyclic graph), CPDAG (completed partially directed graph) and PAG (partial ancestral graph).

	(rank)PC	(rank)FCI	(rank)GES	(rank)GIES	MMHC	LiNGAM	backShift
Causal sufficiency	yes	no	yes	yes	yes	yes	no
Causal faithfulness	yes	yes	yes	yes	yes	no	no
Acyclicity	yes	yes	yes	yes	yes	yes	no
Non-Gaussian errors	no	no	no	no	no	yes	no
Unknown shift interventions	no	no	no	no	no	no	yes
Known do-interventions	no	no	no	yes	no	no	no
Output	CPDAG	PAG	CPDAG	PDAG	DAG	DAG	DG

- Exploiting invariance properties: backShift (Rothenhäusler et al., 2015)

We have not included methods for time series data, mixed data, or Bayesian methods. Other excluded methods that make use of interventional data include Cooper and Yoo (1999) and Tian and Pearl (2001) and Eaton and Murphy (2007), where the latter does not require knowledge of the precise location of interventions in a similar spirit to Rothenhäusler et al. (2015). Hyttinen et al. (2012) also makes use of intervention data to learn feedback models, assuming do-interventions, while Peters et al. (2016) permits to build a graph nodewise by estimating the parental set of each node separately.

2.3.3.1. (rank)PC and (rank)FCI

The PC algorithm (Spirtes et al., 2000) is named after its inventors Peter Spirtes and Clark Glymour. It is a constraint-based algorithm that assumes acyclicity, causal faithfulness and causal sufficiency. It conducts numerous conditional independence tests to learn about the structure of the underlying DAG. In particular, it learns the CPDAG of the underlying DAG in three steps: (i) determining the skeleton, (ii) determining the v-structures, and (iii) determining further edge orientations. The skeleton of the CPDAG is the undirected graph obtained by replacing all directed edges by undirected edges. The PC algorithm learns the skeleton by starting with a complete undirected graph. For $k = 0, 1, 2, \dots$ and adjacent nodes i and j in the current skeleton, it then tests conditional independence of X_i and X_j given X_S for all $S \subseteq \text{adj}(i) \setminus \{j\}$ with $|S| = k$, and for all $S \subseteq \text{adj}(j) \setminus \{i\}$ with $|S| = k$. The algorithm removes an edge if a conditional independence is found (that is, the null hypothesis of independence was not rejected at some level α), and stores the corresponding separating set S. Step (i) stops if the size of the conditioning set k equals the degree of the graph.

In step (ii), all edges are replaced by $\circ\!\!-\!\!\circ$, and the algorithm considers all unshielded triples, that is, all triples $i \circ\!\!-\!\!\circ j \circ\!\!-\!\!\circ k$ where i and k are not adjacent. Based on the separating set that led to the removal of $i - k$, the algorithm determines whether the triple should be oriented as a v-structure $i \to j \leftarrow k$ or not. Finally, in step (iii) some additional orientation rules are applied to orient as many of the remaining undirected edges as possible.

The PC algorithm was shown to be consistent in certain sparse high-dimensional settings (Kalisch and Bühlmann, 2007). There are various modifications of the algorithm. We use the order-independent version of Colombo and Maathuis (2014). The PC algorithm does not impose any distributional assumptions, but conditional independence tests are easiest in the binary and multivariate Gaussian settings. Harris and Drton (2013) proposed a version of the PC algorithm for certain Gaussian copula distributions. We include this algorithm in our comparison and denote it by rankPC. There is also a version of the PC algorithm that allows cycles (Richardson and Spirtes, 1999), but we did not find an R implementation of it.

The Fast Causal Inference (FCI) algorithm Spirtes et al. (2000, 1999) is a modification of the PC algorithm that drops the assumption of causal sufficiency by allowing arbitrarily many hidden variables. The output of the FCI algorithm can be interpreted as a PAG (Zhang, 2008a). The first step of the FCI algorithm is the same as step (i) of the PC algorithm, but the FCI algorithm needs to conduct additional tests to learn the correct skeleton. There are also additional orientation rules, which were shown to be complete in Zhang (2008b). Since the additional tests can slow down the algorithm considerably, faster adaptations have been developed, such as RFCI (Colombo et al., 2012) and FCI+ (Claassen et al., 2013). Colombo et al. (2012) proved high-dimensional consistency of FCI and RFCI. The idea of Harris and Drton (2013) can also be applied to FCI, leading to rankFCI.

2.3.3.2. (rank)GES and (rank)GIES

Greedy equivalence search (GES) (Chickering, 2002b) is a score-based algorithm that assumes acyclicity, causal faithfulness and causal sufficiency. Score-based algorithms use the fact that each DAG can be scored in relation to the data, typically using a penalized likelihood score. The algorithms then search for the DAG or CPDAG that yields the optimal score. Since the space of possible graphs is typically too large, greedy approaches are used. In particular, GES learns the CPDAG of the underlying causal DAG by conducting a greedy search on the space of possible CPDAGs. Its greedy search consists of a forward phase, where it conducts single edge additions that yield the maximum improvement in score, and then a backward phase, where it conducts single edge deletions. Despite the greedy search, Chickering (2002b) showed that the algorithm is consistent under some assumptions (for fixed p). Nandy et al. (2017b) showed high-dimensional consistency of GES.

Greedy interventional equivalence search (GIES) (Hauser and Bühlmann, 2012) is an adaptation of GES to settings with data from different known do-interventions. Due to the additional information from the interventions, its target graphical object is a so-called interventional Markov equivalence class, which is a sub-class of the Markov equivalence class of the underlying DAG and can be seen as a partially directed acyclic graph (PDAG).

Nandy et al. (2017b) showed a close connection between score-based and constraint-based methods for multivariate Gaussian data. As a result, the copula methods that can be used for the PC and FCI algorithms, can be transferred to the GES and GIES algorithms. We include these algorithms in our comparison, and refer to them as rankGES and rankGIES.

2.3.3.3. MMHC

Max-Min Hill Climbing (MMHC) (Tsamardinos et al., 2006) is a hybrid algorithm that assumes acyclicity, causal faithfulness, and causal sufficiency. Hybrid algorithms combine ideas from both constraint-based and score-based approaches. In particular, MMHC first learns the CPDAG skeleton using the constraint-based Max-Min Parents and Children (MMPC) algorithm and then performs a score-based hill-climbing DAG search to determine the edge orientations. Its output is a DAG. Nandy et al. (2017b) showed that the algorithm is not consistent for fixed p, due to the restricted score-based phase.

2.3.3.4. LINGAM

LiNGAM (Shimizu et al., 2006) is an acronym derived from “linear non-gaussian acyclic models” and has been designed for model (2.2) with non-Gaussian noise. It assumes acyclicity and causal sufficiency. It is based on the fact that $X = A\varepsilon$ with $A = (I - B)^{-1}$. This can be viewed as a source separation problem, where identification of the matrix B is equivalent to identification of the mixture matrix A. It was shown in Comon (1994) that whenever at most one of the components of ε is Gaussian, the mixing matrix is identifiable up to scaling and permutation of columns, via independent component analysis (ICA). This observation lies at the basis of the LiNGAM method. There are various modifications of LiNGAM , for example to allow for hidden variables (Hoyer et al., 2008) or cycles (Lacerda et al., 2008). There is also a different implementation called DirectLiNGAM (Shimizu et al., 2011) that uses a pairwise causality measure instead of ICA. Since only ICA-based LiNGAM assuming acyclicity and causal sufficiency is available in R, we include this version in our comparison.

2.3.3.5. BACKSHIFT

backShift (Rothenhäusler et al., 2015) makes use of non-i.i.d. structure in the data and unknown shift interventions on variables. Assume that the data are divided into distinct blocks $\mathcal{E}$. Let $\Gamma_e \in \mathbb{R}^{p \times p}$ be the empirical Gram matrix of the p variables in block $e \in \mathcal{E}$ of the data. In the absence of shift-interventions the expected values of Γ_e would be identical for all $e \in \mathcal{E}$. Under unknown-shift interventions the Gram matrices can change from block to block. However, for the true matrix B of causal coefficients from Eq. (2.2), it can be shown that the expected value of

$$(I - B)(\Gamma_e - \Gamma_{e'})(I - B)^t$$

is a diagonal matrix for all $e, e' \in \mathcal{E}$, even in the presence of latent confounding. BACKSHIFT estimates $I - B$ (and hence B) by a joint diagonalization of all Gram differences $\Gamma_e - \Gamma_{e'}$ for all pairs $e, e' \in \mathcal{E}$. A necessary and sufficient condition for identifiability of the causal matrix B is as follows. Let $\eta_{e,k}$ be the variance of the noise interventions at variable $k \in \{1, \dots, p\}$ in setting $e \in \mathcal{E}$. Full identifiability requires that we can find for each pair of variables (k, l) two settings $e, e' \in \mathcal{E}$ such that the product $\eta_{e,k}\eta_{e',l}$ is *not* equal to the product $\eta_{e,l}, \eta_{e',k}$. A consequence of this necessary and sufficient condition for identifiability is $|\mathcal{E}| \geq 3$, that is, we need to observe at least three different blocks of data for identifiability.

2.4. Empirical evaluation

We conducted an extensive simulation study to evaluate and compare the methods, paying particular attention to sensitivity of the methods to model violations. We are also interested in realistic boundaries (in terms of the number of variables, the sample size, and other simulation parameters) beyond which we cannot expect a reasonable reconstruction of the underlying graph.

In §2.4.1, we describe the data generating mechanism used in the simulation study. §2.4.2 discusses the framework for comparison of the considered methods, and §2.4.3 contains the results.

The code is available in the R package `CompareCausalNetworks` (Heinze-Deml and Meinshausen, 2017a) along with further documentation. All methods are called through the interface offered by the `CompareCausalNetworks` package which depends on the packages `backShift` (Heinze-Deml, 2017), `bnlearn` (Scutari, 2010) and `pcalg` (Kalisch et al., 2012) for the code of the considered methods. In particular, BACKSHIFT is in `backShift`, MMHC is in `bnlearn`, and all other considered methods are in `pcalg`.

2.4.1. Data generation

We generate data sets that differ with respect to the following characteristics: the number of observations n, the number of variables p, the expected number of edges in B, the noise distribution, the correlation of the noise terms, the type, strength and number of interventions, the signal-to-noise ratio, the presence and strength of a cycle in the graph, and possible model misspecifications in terms of nonlinearities. The function `simulateInterventions()` from the package `CompareCausalNetworks` implements the simulation scheme that we describe in more detail below.

We first generate the adjacency matrix B. Assume the variables with indices $\{1, \ldots, p\}$ are causally ordered. For each pair of nodes i and j, where i precedes j in the causal ordering, we draw a sample from $\mathrm{Bern}(p_s)$ to determine the presence of an edge from i to j. After having sampled the non-zero entries of B in this fashion, we sample their corresponding coefficients from $\mathrm{Unif}(-1, 1)$. As described below, the edge weights are later rescaled to achieve a specified signal-to-noise ratio. We exclude the possibility of $B = \mathbf{0}$, that is, we resample until B contains at least one non-zero entry.

Second, we simulate the interventions. We let n_I denote the total number of (interventional and observational) settings that are generated. Let $I \in \{0, 1\}^{n_I \times p}$ be an indicator matrix, where an entry $I_{e,k} = 1$ indicates that variable k is intervened on in setting e and a zero entry indicates that no intervention takes place. For each variable k, we first set the k-th column $I_{\cdot k} \equiv 0$ and then sample one setting e' uniformly at random and set $I_{e' k} = 1$. In other words, each variable is intervened on in exactly one setting. It is possible that there are settings where no interventions take place, corresponding to zero rows of the matrix I, representing the observational setting. Similarly, there may be settings where interventions are performed on multiple variables at once. After defining the settings, we sample (uniformly at random with replacement) what setting each data point belongs to. So for each setting we generate approximately the same number of samples. In any generated data set, the interventions are all of the same type, that is, they are either all shift or all do-interventions (with equal probability). In both cases, an intervention on a variable X_j is modeled by sampling Z_j from a t-distribution as $Z_j \sim \sigma_Z \cdot t(df_\varepsilon)$ (cf. §2.2.1). If $\sigma_Z = 0$ is sampled, it is taken to encode that no interventions should be performed. In that case, all interventional settings correspond to purely observational data.

Third, the noise terms ε are generated by first sampling from a p-dimensional zero-mean Gaussian distribution with covariance matrix Σ, where $\Sigma_{i,i} = 1$ and $\Sigma_{i,j} = \rho_\varepsilon$. The magnitude of ρ_ε models the presence and the strength of hidden variables (cf. §2.2.5). For a positive value of ρ the correlation structure corresponds to the presence of a hidden variable that affects each observed variable. To steer the signal-to-noise ratio, we set the variance of the noise terms of all nodes except for the source nodes to ω, where $0 < \omega \leq 1$. Stepping through the variables in causal order, for each variable X_j that has parents, we uniformly rescale the edge weights $\beta_{j,k}$ in the jth structural equation such that the variance of the sum $\sum_{k=1}^{p} \beta_{j,k} X_k + \varepsilon_j$ is approximately equal to one in the observational setting. In other words, the parameter ω steers what proportion of the variance stems from the noise ε_j. The signal-to-noise ratio can then be computed as $\text{SNR} = (1 - \omega)/\omega$ (in the absence of hidden variables).

Fourth, if the causal graph shall contain a cycle, we sample two nodes i and j such that adding an edge between them creates a cycle in the causal graph. We then compute the coefficient for this edge such that the cycle product is 1. Subsequently, we sample the sign of the coefficient with equal probability and set the magnitude by scaling the coefficient by w_c, where $0 < w_c < 1$.

Fifth, we transform the noise variables to obtain a t-distribution with df_ε degrees of freedom. X is then generated as $X = (I - B)^{-1}\varepsilon$ in the observational case; under a shift interventions X can be generated as $X = (I - B)^{-1}(\varepsilon + Z)$ where the coordinates of Z are only non-zero for the variables that are intervened on. Under a do-intervention on X_j, $\beta_{j,k}$ for $k = 1, \ldots, p$ are set to 0 to yield B' and ε_j is set to Z_j to yield ε'_j. We then obtain X as $X = (I - B')^{-1}\varepsilon'$.

Sixth, if nonlinearity is to be introduced, we marginally transform all variables as $X_j \leftarrow \tanh(X_j)$.

Lastly, we randomly permute the order of the variables in X before running the algorithms. Methods that are order-dependent can therefore not exploit any potential advantage stemming from a data matrix with columns ordered according to the causal ordering or a similar one.

2.4.1.1. Considered settings

We sample the simulation parameters uniformly at random from the following sets.

- Sample size $n \in \{500, 2000, 5000, 10000\}$
- Number of variables $p \in \{2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 15, 20, 50, 100\}$
- Edge density parameter $p_s \in \{0.1, 0.2, 0.3, 0.4\}$
- Number of interventions $n_I \in \{3, 4, 5\}$
- Strength of the interventions $\sigma_Z \in \{0, 0.1, 0.5, 1, 2, 3, 5, 10\}$
- Degrees of freedom of the noise distribution $df_\varepsilon \in \{2, 3, 5, 10, 20, 100\}$
- Strength of hidden variables $\rho_\varepsilon \in \{0, 0.1, 0.2, 0.5, 0.8\}$
- Proportion of variance from noise $\omega \in \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$
- Strength of cycle $w_c \in \{0.1, 0.25, 0.5, 0.75, 0.9\}$

In total, we consider 842 different configurations. For each sampled configuration, we generate 20 different causal graphs with one data set per graph. Appendix 2.B summarizes the number of simulation settings for different values of the simulation parameters.

2.4.2. Evaluation methodology

As the targets of inference differ between the considered methods, we evaluate a method's accuracy for recovering (a) parental and (b) ancestral relations (see also §2.3.2). For each of these, we look at a method's performance for predicting (i) the existence of a relation, (ii) the absence of a relation and (iii) the potential existence of a relation. We formulate these different categories as so-called queries which are further described in §2.4.2.1.

An additional challenge in comparing a diverse set of methods involves choosing the options and the proper amount of regularization that determines the sparsity of the estimated structure. We address this challenge in two ways. First, we run different configurations of each method's tuning parameters and options as detailed in the Appendix in §2.A. In the evaluation of the methods for a certain metric, we choose the method's configuration that yielded the best results under the considered metric in each simulation setting (averaged over the twenty data sets for each setting). This means that the results are optimistically biased, but we found that the ranking was largely insensitive to the tuning parameter choices. Secondly, we use a subsampling scheme (stability ranking) so that each method outputs a ranking of pairs of nodes for a given query. For instance, the first entry in this ranking for the existence of parental relations is the edge most likely to be present in the underlying DAG. Further details are given in §2.4.2.2 and §2.4.2.3.

2.4.2.1. Considered queries

For both the parental and the ancestral relations, we consider three queries. The existence of a relation is assessed by the queries `isParent` and `isAncestor`; the absence of a relation is assessed by the queries `isNoParent` and `isNoAncestor`; the potential existence of a relation is assessed by the queries `isPossibleParent` and `isPossibleAncestor`.

All queries return a connectivity matrix which we denote by A. The interpretation of the entries of A differ according to the considered query:

Parental relations

1. `isParent` This query cannot be easily answered by methods that return a PAG. For the other graphical objects, $A_{i,j} = 1$ if $i \rightarrow j$ in the estimated graph, and $A_{i,j} = 0$ otherwise.
2. `isPossibleParent` Entry $A_{i,j} = 1$ if there is an edge of type $i \mathrel{-\!\!*} j$ or $i \mathrel{\circ\!\!-\!\!*} j$ in the estimated graph. Concretely, for methods estimating DGs or DAGs $A_{i,j} = 1$ if $i \rightarrow j$ in the estimated graph, for PDAGs and CPDAGs $A_{i,j} = 1$ if $i \rightarrow j$ or $i \mathrel{\circ\!\!-\!\!\circ} j$ in the estimated graph, and for PAGs $A_{i,j} = 1$ if $i \rightarrow j$, $i \mathrel{-\!\!\circ} j$, $i - j$, $i \mathrel{\circ\!\!\rightarrow} j$, $i \mathrel{\circ\!\!-\!\!\circ} j$ or $i \mathrel{\circ\!\!-} j$ in the estimated graph. Otherwise, $A_{i,j} = 0$.
3. `isNoParent` The complement of the query `isPossibleParent`: If the latter returns the connectivity matrix A', then entry $A_{i,j} = 1$ if $A'_{i,j} = 0$ and $A_{i,j} = 0$ if $A'_{i,j} = 1$.

Ancestral relations

1. `isAncestor` Entry $A_{i,j} = 1$ if there is a path from i to j with edges of type $\mathrel{-\!\!*}$. For example, for DGs, DAGs and CPDAGs this reduces to a directed path. Otherwise, $A_{i,j} = 0$.
2. `isPossibleAncestor` Entry $A_{i,j} = 1$ if there is a path from i to j such that no edge on the path points towards i (possibly directed path), and $A_{i,j} = 0$ otherwise. In general, such a path can contain

edges of the type $i \mathrel{-\!\!*} j$ and $i \mathrel{\circ\!\!-\!\!*} j$. For DAGs and DGs this again reduces to a directed path, and for CPDAGs it is path with edges $\circ\!\!-\!\!\circ$ and $\rightarrow$.

3. `isNoAncestor` The complement of the query `isPossibleAncestor`: If the latter returns the connectivity matrix A', then entry $A_{i,j} = 1$ if $A'_{i,j} = 0$ and $A_{i,j} = 0$ if $A'_{i,j} = 1$.

2.4.2.2. Stability ranking

To obtain a ranking of pairs of nodes for a given query, we run the method under consideration on $n_{sim} = 100$ random subsamples of the data, where each subsample contains approximately $n/2$ data points. More specifically, we use the following stratified sampling scheme: In each round, we draw samples from $1/\sqrt{2} \cdot n_I$ settings, where n_I denotes the total number of (interventional and observational) settings. In each chosen setting s, we sample $1/\sqrt{2} \cdot n_s$ observations uniformly at random without replacement, where n_s denotes the number of observations in setting s. After a random permutation of the order of the variables, we run the method on each subsample and evaluate the method's output with respect to the considered query.

For each subsample k and a particular query q, we obtain the corresponding connectivity matrix A. We can then rank all pairs of nodes i, j according to the frequency $\pi_{i,j} \in [0, 1]$ of the occurrence of $A_{i,j} = 1$ across subsamples. Ties between pairs of variables can be broken with the results of the other queries—for instance, if the query is `isParent`, ties are broken with counts for `isPossibleParent`. This stability ranking scheme is implemented in the function `getRanking()` in the package `CompareCausalNetworks`. Further details about the tie breaking scheme are given in the package documentation.

2.4.2.3. Metrics

For a chosen query and cut-off value of $t \in (0, 1)$, we select all pairs (i, j) for which $\pi_{i,j} \geq t$. This leads to a true positive rate $\text{TPR}_t = |\{(i, j) : \pi_{i,j} \geq t\} \cap S|/|S|$, where $S := \{(i, j) : A_{i,j} = 1\}$ is the set of correct answers (for example the set of true direct causal effects for the query `isParent`). The corresponding false positive rate is $\text{FPR}_t = |\{(i, j) : \pi_{i,j} \geq t\} \cap S^c|/|S^c|$, with $S^c := \{(i, j) : A_{i,j} = 0\}$. The four metrics we consider are as follows.

(i) **AOC.** The standard area-under-curve (AUC) measures the area below the graph $(\mathrm{FPR}_t, \mathrm{TPR}_t) \in [0,1]^2$ as t is varied between 0 and 1. Under random guessing, the area is 0.5 in expectation and the optimal values is 1. Here, to make rates comparable, we look at the area-above-curve defined as $\mathrm{AOC} = 1 - \mathrm{AUC}$, such that low values are preferable.

(ii) **Equal-error-rate (E-ER).** The equal-error-rate measures the false-negative rate $\mathrm{FNR}_t = 1 - \mathrm{TPR}_t$ at the cutoff t where it equals the false-positive-rate FPR_t, that is, for the value $t \in (0,1)$ for which $1 - \mathrm{TPR}_t = \mathrm{FPR}_t$. The advantage over AOC is that it is a real error rate and is also identical whether we look at the missing edges or at the true edges. For random guessing, the expected value is 0.5 and does not depend on the sparsity of the graph.

(iii) **No-false-positives-error-rate (NFP-ER).** The no-false-positives-error-rate measures the false negative rate $\mathrm{FNR}_t = 1 - \mathrm{TPR}_t$ for the minimal cutoff t at which $\mathrm{FPR}_t = 0$, that is, for the largest number of selections under the constraint that not a single false positive occurs. The expected value under random guessing depends on the sparsity of the graph.

(iv) **No-false-negatives-error-rate (NFN-ER).** The no-false-negatives-detection-rate measures the false-positive rate $\mathrm{FPR}_t = 1 - \mathrm{TNR}_t$ for the maximally large cutoff t at which $\mathrm{FNR}_t = 0$, that is, for the smallest number of selections possible that not a single false negative occurs. The expected value under random guessing depends on the sparsity of the graph.

All four metrics are designed so that lower values are better.

2.4.3. Results

Below, we mostly present results for the `isAncestor` query and the metric E-ER. Results for other queries and metrics are similar in nature.

2.4.3.1. Multi-dimensional scaling

For each simulation setting and each method, we compute the equal-error-rate for the `isAncestor` query. This yields a (no. of simulation settings) × (no. of methods) matrix with E-ER values. The Euclidean distance

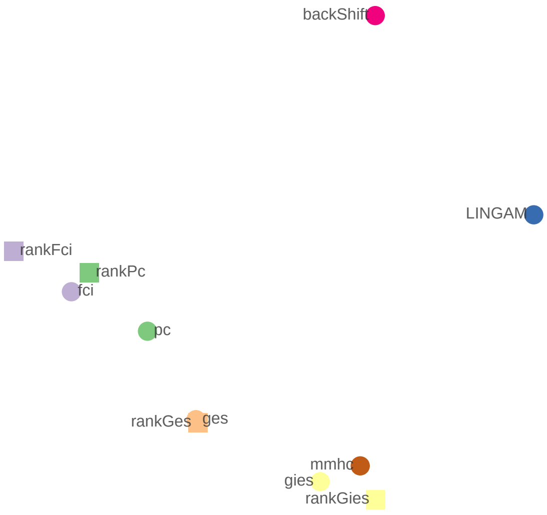


Figure 2.1.: A multi-dimensional scaling visualization of the methods considered. The distance between two methods is taken to be the Euclidean distance between the equal-error-rate of both methods across all settings for the `isAncestor` query. The MDS plot uses least-squares scaling.

between two columns in this matrix is a distance between methods. Similarly, the Euclidean distance between two rows in the matrix is a distance between simulation settings.

Figure 2.1 shows an MDS plot based on distances between the methods, using least-squares scaling. We see that the rank-based methods rankFCI, rankPC, rankGES and rankGIES are close to their counterparts FCI, PC, GES and GIES. It is somewhat unexpected that MMHC is closer to GIES and rankGIES than to PC and GES. The two methods that have the largest average distance to the other methods are LiNGAM and backShift . This is perhaps expected as these methods are of a very different nature than the other methods.

Figure 2.2 shows an MDS plot based on distances between the simula-

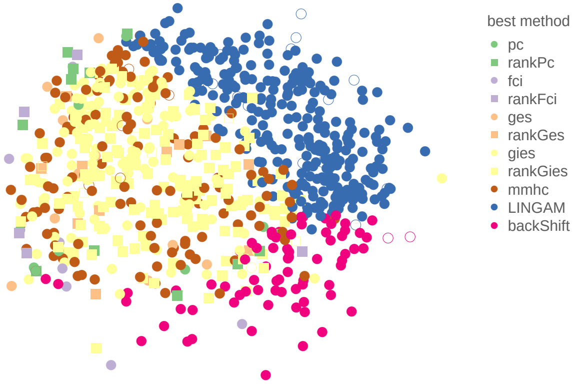


Figure 2.2.: A multi-dimensional scaling visualization of the simulation settings. The distance between two simulation settings is taken to be the correlation distance between the equal-error-rate of both simulation settings across all methods for the `isAncestor` query. Each setting is shown as a sample point with color coding for the best performing method. A filled symbol indicates that the performance metric was smaller than 0.3 and an un-filled symbol that it was above. MDS uses least-squares scaling.

tion settings, again using least-squares scaling. Thus, each point in the plot now corresponds to a simulation setting. The points are colored according to the best performing method. We see that the regions where either LiNGAM or backShift are optimal are relatively well separated, while the regions where GIES, MMHC, PC, GES, FCI or their rank-based versions are optimal, do not show a clear separation, as perhaps already expected from the previous result in Figure 2.1.

2.4.3.2. Pairwise comparisons

Next, we investigate whether there are methods that dominate the others. We compare the equal-error-rate across all different settings in Table 2.2. It is apparent that no such dominance is visible among different pairs of methods. A block-structure is visible, however, with similar groups as in

Table 2.2.: A pairwise comparison. Each column shows the percentage of settings where methods were better by a margin of at least 0.1 in the equal-error-rate compared to method in the given column. For example, LiNGAM beats PC in 14% of the settings, while PC beats LiNGAM by the given marge in 29% of the settings. There is no globally dominant algorithm and a block-structure among related algorithms is visible.

	PC	rankPC	FCI	rankFCI	GES	rankGES	GIES	rankGIES	MMHC	LiNGAM	backShift
PC	0	6	10	16	1	1	1	0	0	14	20
rankPC	0	0	9	10	1	2	0	0	0	11	17
FCI	1	9	0	5	1	1	1	0	0	11	17
rankFCI	0	1	0	0	1	1	0	0	0	10	16
GES	5	15	16	23	0	0	1	0	0	16	26
rankGES	6	15	16	24	0	0	1	0	1	16	25
GIES	18	29	26	35	10	11	0	0	2	25	35
rankGIES	26	36	34	44	17	17	4	0	1	27	38
MMHC	21	33	30	40	16	17	5	0	0	23	36
LiNGAM	29	34	34	38	27	27	19	14	14	0	31
backShift	18	23	24	29	16	16	9	5	7	13	0

Figure 2.1. One block is formed by the constraint-based methods {PC, rankPC, FCI, rankFCI}: the equal-error-rate of constraint-based methods is hardly ever substantially different. The second block is formed by the score-based approaches {GES, rankGIES} and the third given by the extensions and hybrid methods {GIES, rankGIES, MMHC}. This latter block is of interest as MMHC makes fewer assumptions about the available data and does not need to know where interventions occurred. LiNGAM and backShift , on the other hand, do not fit nicely into any block in the empirical comparison and are more orthogonal to the other algorithms in that they perform substantially better *and* substantially worse in many settings, if compared to the other approaches.

Table 2.3.: Marginal rank correlations between equal-error-rate performance (for the `isAncestor` query) on the one hand and parameters settings on the other hand (shown only if absolute value exceeds 0.1, multiplied by 100 and rounded to the nearest multiple of 5). A positive value for p indicates, for example, that the method becomes less successful with increasing p.

	PC	rankPC	FCI	rankFCI	GES	rankGES	GIES	rankGIES	MMHC	LiNGAM	backShift
n		15		10							-15
p	45	45	25	25	40	35	35	40	45	40	75
df$_\varepsilon$										15	
ρ_ε	50	60	55	60	55	55	65	50	50	35	
ω	10	10	10	10	15	10	20	15	10		20
p_s	20	15	20	15	25	25	25	30	30	15	25
do-interv			-10	-10							
n_I											
σ_Z	-35	-25	-35	-30	-35	-35	-25	-35	-30		-30
cyclic			-15	-15						35	
w_c			-15	-15						35	
nonlinear										20	

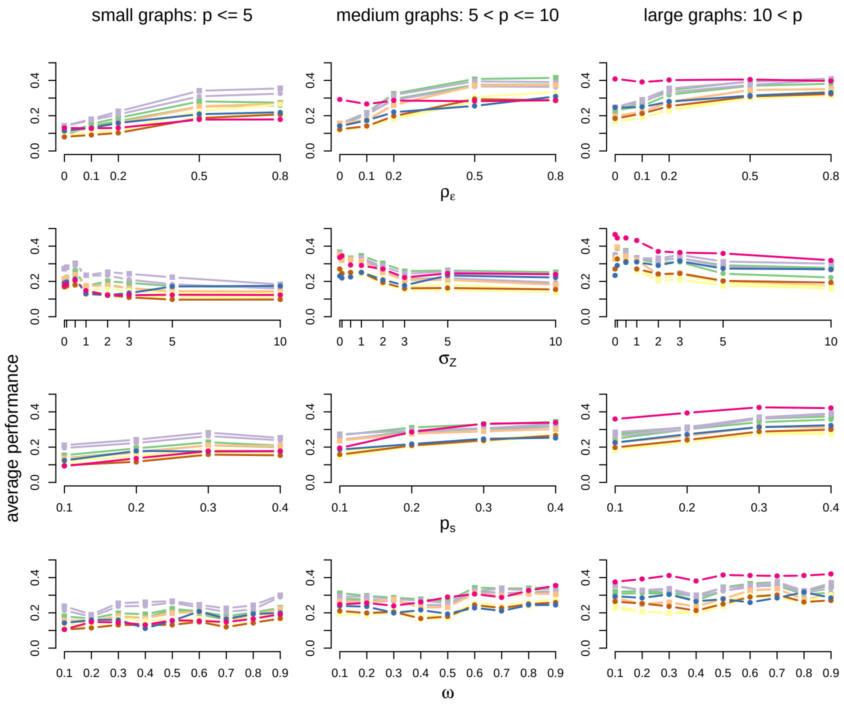


Figure 2.3.: The average equal-error-rate for the `isAncestor` query, for each method as a function for the four most important parameters (besides the number of variables p). The left column shows results for small graphs ($p \leq 5$), the middle column intermediate graphs ($5 < p \leq 10$), and the right column for large graphs ($p > 10$). The color coding is identical to previous plots.

2.4.3.3. Which causal graphs can be estimated well?

Which graphs can be estimated by some or all methods? To start answering the question, we show in Table 2.3 the rank correlation between the equal-error-rate for the `isAncestor` query and parameter settings for all methods. We see that the number of variables p and the strength of the hidden variables ρ_ε show the highest correlations. In both cases the correlation is positive, indicating that increased p or ρ_ε leads to higher equal-error-rates. Other parameters that show substantial correlations are

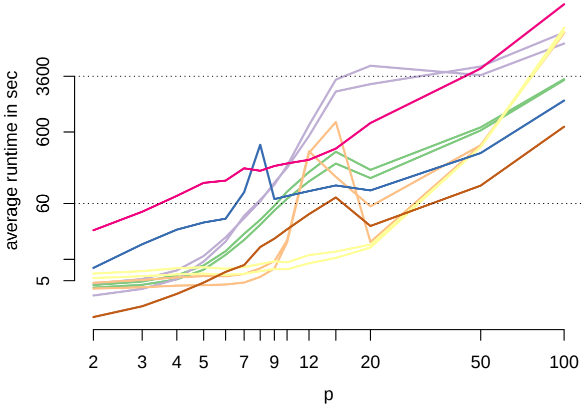


Figure 2.4.: The average runtime in seconds of each method on a logarithmic scale as a function of the number of variables p on a logarithmic scale. A minute and one hour is shown as horizontal bars. The time includes the stability ranking. A single run is faster by a factor of 100 for all methods. (A single run of backShift already includes ten subsamples.)

ω, p_s and σ_Z. For ω and p_s we again see positive correlations, indicating that large noise contributions and denser graphs are associated with higher equal-error-rates. The correlation with σ_Z is negative for all methods except for LiNGAM . While it is expected that backShift benefits from strong interventions, the benefit for for example PC and FCI is unexpected.

We note that the strong effect of ρ_ε can be explained by the fact that we created a correlation ρ_ε between all pairs of noise variables. It is not surprising that this has a larger impact than adding for example a single cycle to the graph (which only seems to substantially affect the performance of LiNGAM).

Figure 2.3 shows the average equal-error-rate for the `isAncestor` query for each method as a function of the simulation parameters ρ_ε, ω, p_s and σ_Z as identified from Table 2.3, split according to the number of variables p in the graph (small, medium-sized and large graphs). Again, we see that the size of the graph p and the strength of the hidden variables ρ_ε have the strongest effect on performance, with the exception that BACKSHIFT is not much affected by ρ_ε (but which is also perhaps less competitive in the absence of latent confounding). The strength of the interventions, the sparsity of the graph and the signal-to-noise ratio also affect the average performance but perhaps to a lesser extent.

Some other observations:

(a) The most surprising outcome is perhaps that the number of samples n has only a very weak influence on the success despite it being varied between a few hundred and twenty thousand.

(b) Sparser graphs with fewer edges are consistently easier to estimate with all methods than dense graphs.

(c) Less heavy tails in the error distribution have an adverse effect on the performance of LiNGAM only, as it makes use of higher moments. LiNGAM is also most affected when each variable undergoes a non-linear transformation.

(d) A cycle in the graph again has a detrimental effect on LiNGAM (which is likely different in the version of LiNGAM that allows for cycles (Lacerda et al., 2008)).

2.4.3.4. Bounds on performance

The outcome of the simulations show a large degree of variation. To further investigate the role of the number of variables p, we show in Figure 2.5 the bounds of the performance as a function of p for the `isAncestor` query. Specifically, for each value of p, we consider the range of the four considered metrics when varying all other parameters for each method and show the lower and upper bounds in the figure.

The upper bounds show the worst performance across all parameters while holding p constant. It can be compared to the expected value under random guessing which is 0.5 for the E-ER and AOC metrics and 1 for NFP-ER and NFN-ER.

The lower bound reveals in contrast the error rates in the best setting for a given p. The metric NFP-ER seems more difficult to keep at reasonable levels than NFN-ER, with the exception of LiNGAM which has very small values of NFP-ER in some settings up to $p \approx 20$. The NFN-ER rate is typically lower than NFP-ER as there are typically more non-ancestral pairs in the graphs (due to not connected components for example) as ancestral pairs. This is confirmed by the third row of panels in Figure 2.5 which shows the error rates for the `isNoAncestor` query. Here the roles of NFN-ER and NFP-ER are reversed due to the relative abundance of non-ancestral pairs.

2.5. Discussion

We have tried to give a contemporaneous overview of structure learning for causal models that are available in R and conducted an extensive empirical comparison. It is noteworthy that we found a clustering of methods into constraint-based, score-based, and other approaches that do not fall neatly into these categories. Methods from the same class behave empirically very similar. We also tried to quantify to what extent methods are negatively or positively affected by various parameters such as the size of the graph to learn, sparsity and strength of hidden variables. The most important parameters in our set-up are the size of the graph p and the strength of the hidden variables ρ_ε. An easily accessible interface to all methods is contributed as R package `CompareCausalNetworks`.

The results suggest that more efficient algorithms would be desirable, both from a computational and from a statistical point-of-view. As it stands, the success of the algorithms depends on both the assumptions made about the data generating process (and how accurate these assumptions are) and the specific implementation details of each algorithm. It would be worthwhile if the relative importance of these two factors could be separated better by more modular estimation methods and perhaps more work on worst-case bounds. These latter bounds would allow to quantify to what extent the empirically poor statistical scalability is inherent to the problem or a consequence of choices made in the considered algorithms.

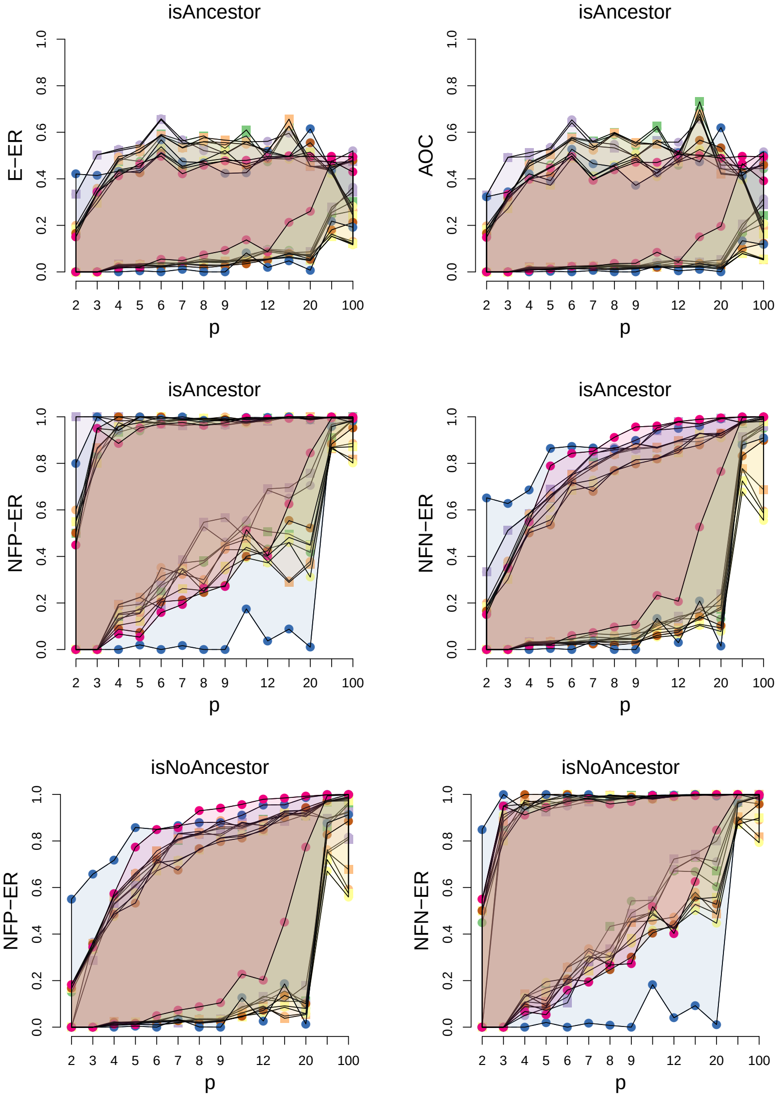


Figure 2.5.: The range of equal-error rate (E-ER) for all methods as a function of the number of variables p for the `isAncestor` query (top left). Top right shows the same for the area-above-curve (AOC), while second row shows the no-false-positives-error-rate (NFP-ER) and no-false-negatives-error-rate (NFN-ER). The last row contains the corresponding plots to the second row but for the `isNoAncestor` query.

Appendix 2.A Considered tuning parameter configurations

All methods were run through the interface offered by the `CompareCausalNetworks` package (Heinze-Deml and Meinshausen, 2017a). Below we also indicate the R packages from which the `CompareCausalNetworks` package calls the respective methods.

backShift Code available from the R package `backShift` (Heinze-Deml, 2017).

- `covariance` $\in$ {`TRUE, FALSE`}
- `ev` $\in \{0.1, 0.25, 0.5\} \cdot p$
- `threshold` $= 0.75$
- `nsim` $= 10$
- `sampleSettings` $= 1/sqrt(2)$
- `sampleObservations` $= 1/sqrt(2)$
- `nodewise = TRUE`
- `tolerance` $= 10^{-4}$

GES and rankGES Code available from the R packages `pcalg` (Kalisch et al., 2012) (GES) and `CompareCausalNetworks` (rankGES).

- `phase = 'turning'`
- `score = GaussL0penObsScore`
- $\lambda \in \{0.05 \log n, 0.5 \log n, 5 \log n\}$
- `adaptive = "none"`
- `maxDegree = integer(0)`

GIES and rankGIES Code available from the R packages `pcalg` (Kalisch et al., 2012) (GIES) and `CompareCausalNetworks` (rankGIES).

- `phase = 'turning'`
- `score = GaussL0penObsScore`

- $\lambda \in \{0.05 \log n, 0.5 \log n, 5 \log n\}$
- `adaptive = "none"`
- `maxDegree = integer(0)`

FCI and rankFCI Code available from the R packages `pcalg` (Kalisch et al., 2012) (FCI) and `CompareCausalNetworks` (rankFCI).

- `conservative = FALSE` and `maj.rule = FALSE`
- `conservative = TRUE` and `maj.rule = FALSE`
- `conservative = FALSE` and `maj.rule = TRUE`
- `alpha` $\in \{0.001, 0.01, 0.1\}$
- `indepTest = gaussCItest`
- `skel.method = "stable"`
- `m.max = Inf`
- `pdsep.max = Inf`
- `rules = rep(TRUE,10)`
- `NAdelete = TRUE`
- `doPdsep = TRUE`
- `biCC = FALSE`

MMHC Code available from the R package `bnlearn` (Scutari, 2010).

- $\lambda \in \{0.05 \log n, 0.5 \log n, 5 \log n\}$
- `alpha` $\in \{0.001, 0.01, 0.1\}$
- `whitelist = NULL`
- `blacklist = NULL`
- `test = NULL` – corresponds to correlation
- `score = NULL` – corresponds to BIC
- `B = NULL`
- `restart = 0`
- `perturb = 1`

- `max.iter = Inf`
- `optimized = TRUE`
- `strict = FALSE`

PC and Rank PC Code available from the R packages `pcalg` (Kalisch et al., 2012) (PC) and `CompareCausalNetworks` (rankPC).

- `conservative = FALSE` and `maj.rule = FALSE`
- `conservative = TRUE` and `maj.rule = FALSE`
- `conservative = FALSE` and `maj.rule = TRUE`
- `alpha` $\in \{0.001, 0.01, 0.1\}$
- `indepTest = gaussCItest`
- `NAdelete = TRUE`
- `m.max = Inf`
- `u2pd = "relaxed"`
- `skel.method = "stable"`
- `solve.confl = FALSE`

Appendix 2.B Simulation settings

The results in this work are based on 842 unique simulation settings. The tables below show for each parameter in the data generation scheme how many settings were generated for each considered value for the given parameter.

Sample size

n	500	2000	5000	10000
# of settings	231	200	217	194

Number of variables

p	2	3	4	5	6	7	8	9	10	12	15	20	50	100
# of settings	71	89	84	77	62	60	74	68	62	76	60	43	8	8

Edge density parameter

p_s	0.1	0.2	0.3	0.4
# of settings	202	226	200	214

Number of settings

n_I	3	4	5
# of settings	271	275	296

Intervention type

	shift intervention	do-intervention
# of settings	417	425

Strength of the interventions

σ_Z	0	0.1	0.5	1	2	3	5	10
# of settings	111	105	102	105	106	98	116	99

Degrees of freedom of the noise distribution

df_ε	2	3	5	10	20	100
# of settings	140	136	147	140	144	135

Strength of hidden variables

ρ_ε	0	0.1	0.2	0.5	0.8
# of settings	161	164	166	179	172

Proportion of variance from noise

ω	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
# of settings	116	85	88	110	85	91	82	91	94

Settings with cycles

	no cycles	cycles
# of settings	576	266

Strength of cycle

w_c	0	0.1	0.25	0.5	0.75	0.9
# of settings	576	56	51	50	55	54

Settings with model misspecification

	no model misspecification	model misspecification
# of settings	715	127

Part II.

Invariance-based Causal Learning

Chapter 3.

backShift: Learning causal cyclic graphs from unknown shift interventions

We propose a simple method to learn linear causal cyclic models in the presence of latent variables. The method relies on equilibrium data of the model recorded under a specific kind of interventions ("shift interventions"). The location and strength of these interventions do not have to be known and can be estimated from the data. Our method, called BACKSHIFT, only uses second moments of the data and performs simple joint matrix diagonalization, applied to differences between covariance matrices. We give a sufficient and necessary condition for identifiability of the system, which is fulfilled almost surely under some quite general assumptions if and only if there are at least three distinct experimental settings, one of which can be pure observational data. We demonstrate the performance on some simulated data and applications in flow cytometry and financial time series.

3.1. Introduction

Discovering causal effects is a fundamentally important yet very challenging task in various disciplines, from public health research and sociological studies, economics to many applications in the life sciences. There has been much progress on learning acyclic graphs in the context of structural equation models (Bollen, 1989), including methods that learn from observational data alone under a faithfulness assumption (Chickering, 2002b; Hauser and Bühlmann, 2012; Maathuis et al., 2009; Spirtes et al., 2000), exploiting non-Gaussianity of the data (Hoyer et al., 2008; Shimizu et al., 2011) or non-linearities (Mooij et al., 2011). Feedbacks are prevalent in most applications, and we are interested in the setting of Hyttinen et al. (2012), where we observe the equilibrium data of a model that is characterized by a set of linear relations

$$\mathbf{x} = \mathbf{B}\mathbf{x} + \mathbf{e}, \tag{3.1}$$

where $\mathbf{x} \in \mathbb{R}^p$ is a random vector and $\mathbf{B} \in \mathbb{R}^{p \times p}$ is the connectivity matrix with zeros on the diagonal (no self-loops). Allowing for self-loops would lead to an identifiability problem, independent of the method. See §3.B in the Appendix for more details on this setting. The graph corresponding to $\mathbf{B}$ has p nodes and an edge from node j to node i if and only if $\mathbf{B}_{i,j} \neq 0$. The error terms $\mathbf{e}$ are p-dimensional random variables with mean 0 and positive semi-definite covariance matrix $\boldsymbol{\Sigma}_{\mathbf{e}} = E(\mathbf{e}\mathbf{e}^T)$. We do not assume that $\boldsymbol{\Sigma}_{\mathbf{e}}$ is a diagonal matrix which allows the existence of latent variables.

The solutions to (3.1) can be thought of as the deterministic equilibrium solutions (conditional on the noise term) of a dynamic model governed by first-order difference equations with matrix $\mathbf{B}$ in the sense of Lauritzen and Richardson (2002). For well-defined equilibrium solutions of (3.1), we need that $\mathbf{I} - \mathbf{B}$ is invertible. Usually we also want (3.1) to converge to an equilibrium when iterating as $\mathbf{x}^{(new)} \leftarrow \mathbf{B}\mathbf{x}^{(old)} + \mathbf{e}$ or in other words $\lim_{m \to \infty} \mathbf{B}^m \equiv \mathbf{0}$. This condition is equivalent to the spectral radius of $\mathbf{B}$ being strictly smaller than one (Lacerda et al., 2008). We will make an assumption on cyclic graphs that restricts the strength of the feedback. Specifically, let a cycle of length η be given by $(m_1, \ldots, m_{\eta+1} = m_1) \in \{1, \ldots, p\}^{1+\eta}$ and $m_k \neq m_\ell$ for $1 \leq k < \ell \leq \eta$. We define the cycle-product $CP(\mathbf{B})$ of a matrix $\mathbf{B}$ to be the maximum over cycles of all lengths $1 < \eta \leq p$ of the path-products

$$CP(\mathbf{B}) := \max_{\substack{(m_1, \ldots, m_\eta, m_{\eta+1}) \text{ cycle} \\ 1 < \eta \leq p}} \prod_{1 \leq k \leq \eta} \left| \mathbf{B}_{m_{k+1}, m_k} \right|. \tag{3.2}$$

The cycle-product $CP(\mathbf{B})$ is clearly zero for acyclic graphs. We will assume the cycle-product to be strictly smaller than one for identifiability results, see Assumption (A) below. The most interesting graphs are those for which $CP(\mathbf{B}) < 1$ and for which the spectral radius of $\mathbf{B}$ is strictly smaller than one. Note that these two conditions are identical as long as the cycles in the graph do not intersect, i.e., there is no node that is part of two cycles (for example if there is at most one cycle in the graph). If cycles do intersect, we can have models for which either (i) $CP(\mathbf{B}) < 1$ but the spectral radius is larger than one or (ii) $CP(\mathbf{B}) > 1$ but the spectral radius is strictly smaller than one. Models in situation (ii) are not stable in the sense that the iterations will not converge under interventions. We can for example block all but one cycle. If this one single unblocked cycle has a cycle-product larger than 1 (and there is such a cycle in the graph if $CP(\mathbf{B}) > 1$), then the solutions of the iteration are not stable[1]. Models in situation (i) are not stable either, even in the absence of interventions. We can still in theory obtain the now instable equilibrium solutions to (3.1) as $(\mathbf{I} - \mathbf{B})^{-1}\mathbf{e}$ and the theory below applies to these instable equilibrium solutions. However, such instable equilibrium solutions are arguably of little practical interest. In summary: all interesting feedback models that are stable under interventions satisfy both $CP(\mathbf{B}) < 1$ and have a spectral radius strictly smaller than one. We will just assume $CP(\mathbf{B}) < 1$ for the following results.

It is impossible to learn the structure $\mathbf{B}$ of this model from observational data alone without making further assumptions. The LiNGAM approach has been extended in Lacerda et al. (2008) to cyclic models, exploiting a possible non-Gaussianity of the data. Using both experimental and interventional data, Scheines et al. (2010) and Hyttinen et al. (2012) could show identifiability of the connectivity matrix $\mathbf{B}$ under a learning mechanism that relies on data under so-called "surgical" or "perfect" interventions. In their framework, a variable becomes independent of all its parents if it is being intervened on and all incoming contributions are thus effectively removed under the intervention (also called do-interventions in the classical sense of Pearl (2009)). The learning mechanism makes then use of the knowledge where these "surgical" interventions occurred. Eberhardt et al. (2010) also allow for "changing" the incoming arrows for variables that are intervened on; but again, Eberhardt et al. (2010) requires the location of the interventions while we do not assume such knowledge. Peters et al. (2016) consider a target variable and allow for arbitrary interventions on all other nodes. They neither permit hidden variables nor cycles.

Here, we are interested in a setting where we have either no or just very limited knowledge about the exact location and strength of the interventions, as is often the case for data observed under different environments

---

[1] The blocking of all but one cycle can be achieved by do-interventions on appropriate variables under the following condition: for every pair of cycles in the graph, the variables in one cycle cannot be a subset of the variables in the other cycle. Otherwise the blocking could be achieved by deletion of appropriate edges.

(see the example on financial time series further below) or for biological data (Jackson et al., 2003; Kulkarni et al., 2006). These interventions have been called “fat-hand” or “uncertain” interventions (Eaton and Murphy, 2007). While Eaton and Murphy (2007) assume acyclicity and model the structure explicitly in a Bayesian setting, we assume that the data in environment j are equilibrium observations of the model

$$\mathbf{x}_j = \mathbf{B}\mathbf{x}_j + \mathbf{c}_j + \mathbf{e}_j, \tag{3.3}$$

where the random intervention shift $\mathbf{c}_j$ has a mean and covariance $\boldsymbol{\Sigma}_{\mathbf{c},j}$. The *location* of these interventions (or simply the *intervened variables*) are those components of $\mathbf{c}_j$ that are not zero with probability one. Given these locations, the interventions simply shift the variables by a value determined by $\mathbf{c}_j$; they are therefore not “surgical” but can be seen as a special case of what is called an “imperfect”, “parametric” (Eberhardt and Scheines, 2007) or “dependent” intervention (Korb et al., 2004) or “mechanism change” (Tian and Pearl, 2001). The matrix $\mathbf{B}$ and the error distribution of $\mathbf{e}_j$ are assumed to be identical in all environments. In contrast to the covariance matrix for the noise term $\mathbf{e}_j$, we *do* assume that $\boldsymbol{\Sigma}_{\mathbf{c},j}$ is a diagonal matrix, which is equivalent to demanding that interventions at different variables are uncorrelated. This is a key assumption necessary to identify the model using experimental data. Furthermore, we will discuss in §3.4.2 how a violation of the model assumption (3.3) can be detected and used to estimate the location of the interventions.

In §3.2 we show how to leverage observations under different environments with different interventional distributions to learn the structure of the connectivity matrix $\mathbf{B}$ in model (3.3). The method rests on a simple joint matrix diagonalization. We will prove necessary and sufficient conditions for identifiability in §3.3. Numerical results for simulated data and applications in flow cytometry and financial data are shown in §3.4.

3.2. Method

3.2.1. Grouping of data

Let $\mathcal{J}$ be the set of experimental conditions under which we observe equilibrium data from model (3.3). These different experimental conditions can arise in two ways: (a) a controlled experiment was conducted where the external input or the external imperfect interventions have been deliberately changed from one member of $\mathcal{J}$ to the next. An example are the flow cytometry data (Sachs et al., 2005) discussed in §3.4.2. (b) The data are recorded over time. It is assumed that the external input is changing over time but not in an explicitly controlled way. The data are grouped into consecutive blocks $j \in \mathcal{J}$ of observations, see §3.4.3 for an example.

3.2.2. Notation

Assume we have n_j observations in each setting $j \in \mathcal{J}$. Let $\mathbf{X}_j$ be the $(n_j \times p)$-matrix of observations from model (3.3). For general random variables $\mathbf{a}_j \in \mathbb{R}^p$, the population covariance matrix in setting $j \in \mathcal{J}$ is called $\mathbf{\Sigma}_{\mathbf{a},j} = \mathrm{Cov}(\mathbf{a}_j)$, where the covariance is under the setting $j \in \mathcal{J}$. Furthermore, the covariance on all settings except setting $j \in \mathcal{J}$ is defined as an average over all environments except for the j-th environment, $(|\mathcal{J}|-1)\mathbf{\Sigma}_{\mathbf{c},-j} := \sum_{j' \in \mathcal{J} \setminus \{j\}} \mathbf{\Sigma}_{\mathbf{c},j'}$. The population Gram matrix is defined as $\mathbf{G}_{\mathbf{a},j} = E(\mathbf{a}_j {\mathbf{a}_j}^T)$. Let the $(p \times p)$-dimensional $\hat{\mathbf{\Sigma}}_{\mathbf{a},j}$ be the empirical covariance matrix of the observations $\mathbf{A}_j \in \mathbb{R}^{n_j \times p}$ of variable $\mathbf{a}_j$ in setting $j \in \mathcal{J}$. More precisely, let $\tilde{\mathbf{A}}_j$ be the column-wise mean-centered version of $\mathbf{A}_j$. Then $\hat{\mathbf{\Sigma}}_{\mathbf{a},j} := (n_j - 1)^{-1} \tilde{\mathbf{A}}_j^T \tilde{\mathbf{A}}_j$. The empirical Gram matrix is denoted by $\hat{\mathbf{G}}_{\mathbf{a},j} := n_j^{-1} \mathbf{A}_j^T \mathbf{A}_j$.

3.2.3. Assumptions

The main assumptions have been stated already but we give a summary below.

(A) The data are observations of the equilibrium observations of model (3.3). The matrix $\mathbf{I} - \mathbf{B}$ is invertible and the solutions to (3.3) are thus well defined. The cycle-product (3.2) $CP(\mathbf{B})$ is strictly smaller than one. The diagonal entries of $\mathbf{B}$ are zero.

(B) The distribution of the noise $\mathbf{e}_j$ (which includes the influence of latent variables) and the connectivity matrix $\mathbf{B}$ are identical across all settings $j \in \mathcal{J}$. In each setting $j \in \mathcal{J}$, the intervention shift $\mathbf{c}_j$ and the noise $\mathbf{e}_j$ are uncorrelated.

(C) Interventions at different variables in the same setting are uncorrelated, that is $\mathbf{\Sigma}_{\mathbf{c},j}$ is an (unknown) diagonal matrix for all $j \in \mathcal{J}$.

We will discuss a stricter version of (C) in §3.D in the Appendix that allows the use of Gram matrices instead of covariance matrices. The conditions above imply that the environments are characterized by different interventions strength, as measured by the variance of the shift $\mathbf{c}$ in each setting. We aim to reconstruct both the connectivity matrix $\mathbf{B}$ from observations in different environments and also aim to reconstruct the a-priori unknown intervention strength and location in each environment. Additionally, we will show examples where we can detect violations of the model assumptions and use these to reconstruct the location of interventions.

3.2.4. Population method

The main idea is very simple. Looking at the model (3.3), we can rewrite

$$(\mathbf{I}-\mathbf{B})\mathbf{x}_j = \mathbf{c}_j + \mathbf{e}_j. \tag{3.4}$$

The population covariance of the transformed observations are then for all settings $j \in \mathcal{J}$ given by

$$(\mathbf{I}-\mathbf{B})\boldsymbol{\Sigma}_{\mathbf{x},j}(\mathbf{I}-\mathbf{B})^T = \boldsymbol{\Sigma}_{\mathbf{c},j} + \boldsymbol{\Sigma}_{\mathbf{e}}. \tag{3.5}$$

The last term $\boldsymbol{\Sigma}_{\mathbf{e}}$ is constant across all settings $j \in \mathcal{J}$ (but not necessarily diagonal as we allow hidden variables). Any change of the matrix on the left-hand side thus stems from a shift in the covariance matrix $\boldsymbol{\Sigma}_{\mathbf{c},j}$ of the interventions. Let us define the difference between the covariance of $\mathbf{c}$ and $\mathbf{x}$ in setting j as

$$\boldsymbol{\Delta\Sigma}_{\mathbf{c},j} := \boldsymbol{\Sigma}_{\mathbf{c},j} - \boldsymbol{\Sigma}_{\mathbf{c},-j}, \quad \text{and} \quad \boldsymbol{\Delta\Sigma}_{\mathbf{x},j} := \boldsymbol{\Sigma}_{\mathbf{x},j} - \boldsymbol{\Sigma}_{\mathbf{x},-j}. \tag{3.6}$$

Assumption (B) together with (3.5) implies that

$$(\mathbf{I}-\mathbf{B})\boldsymbol{\Delta\Sigma}_{\mathbf{x},j}(\mathbf{I}-\mathbf{B})^T = \boldsymbol{\Delta\Sigma}_{\mathbf{c},j} \qquad \forall j \in \mathcal{J}. \tag{3.7}$$

Using assumption (C), the random intervention shifts at different variables are uncorrelated and the right-hand side in (3.7) is thus a diagonal matrix for all $j \in \mathcal{J}$. Let $\mathcal{D} \subset \mathbb{R}^{p\times p}$ be the set of all invertible matrices. We also define a more restricted space $\mathcal{D}_{cp}$ which only includes those members of $\mathcal{D}$ that have entries all equal to one on the diagonal and have a cycle-product less than one,

$$\mathcal{D} := \Big\{\mathbf{D} \in \mathbb{R}^{p\times p} : \mathbf{D} \text{ invertible}\Big\} \tag{3.8}$$

$$\mathcal{D}_{cp} := \Big\{\mathbf{D} \in \mathbb{R}^{p\times p} : \mathbf{D} \in \mathcal{D} \text{ and } \mathrm{diag}(\mathbf{D}) \equiv 1 \text{ and } CP(\mathbf{I}-\mathbf{D}) < 1\Big\}. \tag{3.9}$$

Algorithm 1 BACKSHIFT

```
Input: $\mathbf{X}_j \; \forall j \in \mathcal{J}$
1: Compute $\widehat{\boldsymbol{\Delta\Sigma}}_{\mathbf{x},j} \; \forall j \in \mathcal{J}$
2: $\tilde{\mathbf{D}} = $ FFDIAG($\widehat{\boldsymbol{\Delta\Sigma}}_{\mathbf{x},j}$)
3: $\hat{\mathbf{D}} = $ PermuteAndScale($\tilde{\mathbf{D}}$)
4: $\hat{\mathbf{B}} = \mathbf{I} - \hat{\mathbf{D}}$
Output: $\hat{\mathbf{B}}$
```

Under Assumption (A), $\mathbf{I} - \mathbf{B} \in \mathcal{D}_{cp}$. Motivated by (3.7), we now consider the minimizer

$$\mathbf{D} = \mathrm{argmin}_{\mathbf{D}' \in \mathcal{D}_{cp}} \sum_{j \in \mathcal{J}} L(\mathbf{D}' \boldsymbol{\Delta\Sigma}_{\mathbf{x},j} \mathbf{D}'^T), \qquad \text{where } L(\mathbf{A}) := \sum_{k \neq l} \mathbf{A}_{k,l}^2 \tag{3.10}$$

is the loss L for any matrix $\mathbf{A}$ and defined as the sum of the squared off-diagonal elements. In §3.3, we present necessary and sufficient conditions on the interventions under which $\mathbf{D} = \mathbf{I} - \mathbf{B}$ is the unique minimizer of (3.10). In this case, exact joint diagonalization is possible so that $L(\mathbf{D} \boldsymbol{\Delta\Sigma}_{\mathbf{x},j} \mathbf{D}^T) = 0$ for all environments $j \in \mathcal{J}$. We discuss an alternative that replaces covariance with Gram matrices throughout in §3.D in the Appendix. We now give a finite-sample version.

3.2.5. Finite-sample estimate of the connectivity matrix

In practice, we estimate $\mathbf{B}$ by minimizing the empirical counterpart of (3.10) in two steps. First, the solution of the optimization is only constrained to matrices in $\mathcal{D}$. Subsequently, we enforce the constraint on the solution to be a member of $\mathcal{D}_{cp}$. The BACKSHIFT algorithm is presented in Algorithm 1 and we describe the important steps in more detail below.

Steps 1 & 2. First, we minimize the following empirical, less constrained variant of (3.10)

$$\tilde{\mathbf{D}} := \mathrm{argmin}_{\mathbf{D}' \in \mathcal{D}} \sum_{j \in \mathcal{J}} L(\mathbf{D}'(\widehat{\boldsymbol{\Delta\Sigma}}_{\mathbf{x},j})\mathbf{D}'^T), \tag{3.11}$$

where the population differences between covariance matrices are replaced with their empirical counterparts and the only constraint on the solution is that it is invertible, i.e. $\tilde{\mathbf{D}} \in \mathcal{D}$. For the optimization we use the joint approximate matrix diagonalization algorithm FFDiag (Ziehe et al., 2004).

Step 3. The constraint on the cycle product and the diagonal elements of $\mathbf{D}$ is enforced by (a) permuting and (b) scaling the rows of $\tilde{\mathbf{D}}$. Part (b) simply scales the rows so that the diagonal elements of the resulting matrix $\hat{\mathbf{D}}$ are all equal to one. The more challenging first step (a) consists of finding a permutation such that under this permutation the scaled matrix from part (b) will have a cycle product as small as possible (as follows from Theorem 3.3, at most one permutation can lead to a cycle product less than one). This optimization problem seems computationally challenging at first, but we show that it can be solved by a variant of the *linear assignment problem* (LAP) (see e.g. Burkard (2013)), as proven in Theorem 3.3 in the Appendix. As a last step, we check whether the cycle product of $\hat{\mathbf{D}}$ is less than one, in which case we have found the solution. Otherwise, no solution satisfying the model assumptions exists and we return a warning that the model assumptions are not met. See Appendix 3.B for more details.

Computational cost. The computational complexity of backShift is $O(|\mathcal{J}| \cdot n \cdot p^2)$ as computing the covariance matrices costs $O(|\mathcal{J}| \cdot n \cdot p^2)$, FFDiag has a computational cost of $O(|\mathcal{J}| \cdot p^2)$ and both the linear assignment problem and computing the cycle product can be solved in $O(p^3)$ time. For instance, this complexity is achieved when using the Hungarian algorithm for the linear assignment problem (see e.g. Burkard (2013)) and the cycle product can be computed with a simple dynamic programming approach.

3.2.6. Estimating the intervention variances

One additional benefit of backShift is that the location and strength of the interventions can be estimated from the data. The empirical, plug-in version of Eq. (3.7) is given by

$$(\mathbf{I} - \hat{\mathbf{B}})\widehat{\boldsymbol{\Delta\Sigma}}_{\mathbf{x},j}(\mathbf{I} - \hat{\mathbf{B}})^T = \widehat{\boldsymbol{\Delta\Sigma}}_{\mathbf{c},j} = \widehat{\boldsymbol{\Sigma}}_{\mathbf{c},j} - \widehat{\boldsymbol{\Sigma}}_{\mathbf{c},-j} \qquad \forall j \in \mathcal{J}. \qquad (3.12)$$

So the element $(\widehat{\boldsymbol{\Delta\Sigma}}_{\mathbf{c},j})_{kk}$ is an estimate for the difference between the variance of the intervention at variable k in environment j, namely $(\boldsymbol{\Sigma}_{\mathbf{c},j})_{kk}$, and the average in all other environments, $(\boldsymbol{\Sigma}_{\mathbf{c},-j})_{kk}$. From these differences we can compute the intervention variance for all environments up to an offset. By convention, we set the minimal intervention variance across all environments equal to zero. Alternatively, one can let observational data, if available, serve as a baseline against which the intervention variances are measured.

3.3. Identifiability

Let for simplicity of notation,

$$\boldsymbol{\eta}_{j,k} := (\boldsymbol{\Delta\Sigma}_{\mathbf{c},j})_{kk}$$

be the variance of the random intervention shifts $\mathbf{c}_j$ at node k in environment $j \in \mathcal{J}$ as per the definition of $\boldsymbol{\Delta\Sigma}_{\mathbf{c},j}$ in (3.6). We then have the following identifiability result (the proof is provided in Appendix 3.A).

Theorem 3.1 *Under assumptions (A), (B) and (C), the solution to* (3.10) *is unique if and only if for all $k, l \in \{1, \ldots, p\}$ there exist $j, j' \in \mathcal{J}$ such that*

$$\boldsymbol{\eta}_{j,k}\boldsymbol{\eta}_{j',l} \neq \boldsymbol{\eta}_{j,l}\boldsymbol{\eta}_{j',k} \,. \tag{3.13}$$

If none of the intervention variances $\boldsymbol{\eta}_{j,k}$ vanishes, the uniqueness condition is equivalent to demanding that the ratio between the intervention variances for two variables k, l must not stay identical across all environments, that is there exist $j, j' \in \mathcal{J}$ such that

$$\frac{\boldsymbol{\eta}_{j,k}}{\boldsymbol{\eta}_{j,l}} \neq \frac{\boldsymbol{\eta}_{j',k}}{\boldsymbol{\eta}_{j',l}}, \tag{3.14}$$

which requires that the ratio of the variance of the intervention shifts at two nodes k, l is not identical across all settings. This leads to the following corollary.

Corollary 3.2 *(i) The identifiability condition* (3.13) *cannot be satisfied if $|\mathcal{J}| = 2$ since then $\boldsymbol{\eta}_{j,k} = -\boldsymbol{\eta}_{j',k}$ for all k and $j \neq j'$. We need at least three different environments for identifiability.*

(ii) The identifiability condition (3.13) *is satisfied for all* $|\mathcal{J}| \geq 3$ *almost surely if the variances of the intervention* $\mathbf{c}_j$ *are chosen independently (over all variables and environments* $j \in \mathcal{J}$*) from a distribution that is absolutely continuous with respect to Lebesgue measure.*

Condition (ii) can be relaxed but shows that we can already achieve full identifiability with a very generic setting for three (or more) different environments.

3.4. Numerical results

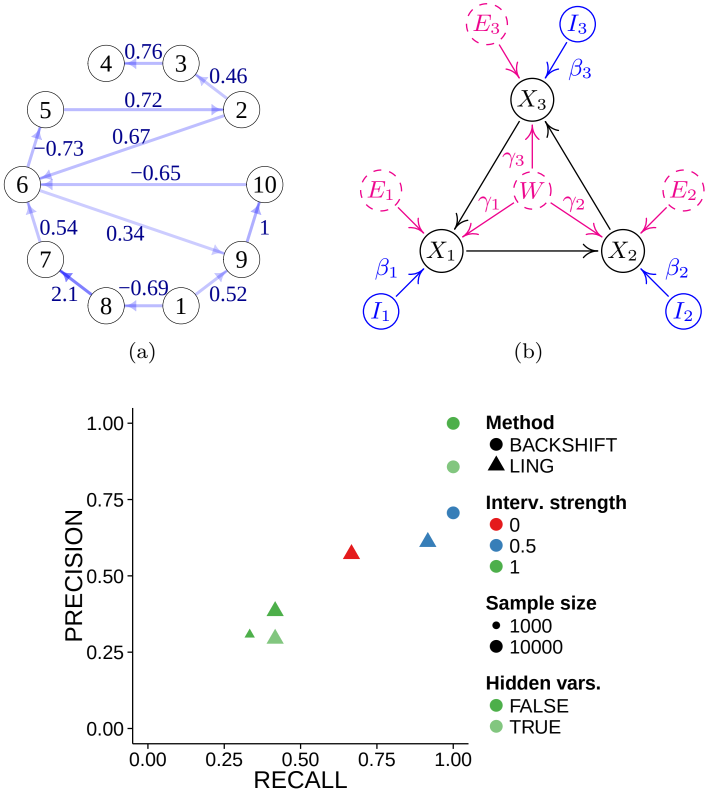
In this section, we present empirical results for both synthetic and real data sets. In addition to estimating the connectivity matrix $\mathbf{B}$, we demonstrate various ways to estimate properties of the interventions. Besides computing the point estimate for BACKSHIFT, we use *stability selection* (Meinshausen and Bühlmann, 2010) to assess the stability of retrieved edges. We attach R code with which all simulations and analyses can be reproduced[2].

3.4.1. Synthetic data

We compare the point estimate of BACKSHIFT against LING (Lacerda et al., 2008), a generalization of LINGAM to the cyclic case for purely observational data. We consider the cyclic graph shown in Figure 3.1(a) and generate data under different scenarios. The data generating mechanism is sketched in Figure 3.1(b). Specifically, we generate ten distinct environments with non-Gaussian noise. In each environment, the random intervention variable is generated as $(\mathbf{c}_j)_k = \beta_k^j I_k^j$, where $\beta_1^j, \ldots, \beta_p^j$ are drawn i.i.d. from $\text{Exp}(m_I)$ and $I_1^j, \ldots, I_p^j$ are independent standard normal random variables. The intervention shift thus acts on all observed random variables. The parameter m_I regulates the strength of the intervention. If hidden variables exist, the noise term $(\mathbf{e}_j)_k$ of variable k in environment j is equal to $\gamma_k W^j$, where the weights $\gamma_1, \ldots, \gamma_p$ are sampled once from a $\mathcal{N}(0,1)$-distribution and the random variable W^j has a $\text{Laplace}(0,1)$ distribution. If no hidden variables are present, then $(\mathbf{e}_j)_k$, $k = 1, \ldots, p$ is sampled i.i.d. $\text{Laplace}(0,1)$. In this set of experiments, we

---

[2] An R package called "`backShift`" is available from CRAN.



(a) (b) (c)

Figure 3.1.: Simulated data. (a) True network. (b) Scheme for data generation. (c) Performance metrics for the settings considered in §3.4.1. For backShift, precision and recall values for Settings 1 and 2 coincide.

consider five different settings (described below) in which the sample size n, the intervention strength m_I as well as the existence of hidden variables varies.

We allow for hidden variables in only one out of five settings as Ling assumes causal sufficiency and can thus in theory not cope with hidden variables. If no hidden variables are present, the pooled data can be interpreted as coming from a model whose error variables follow a mixture

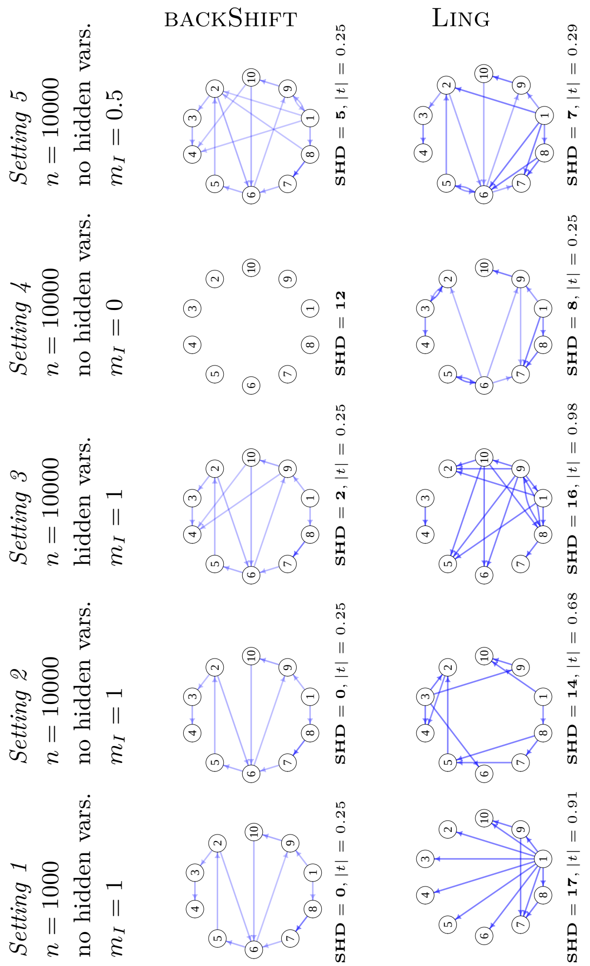


Figure 3.2.: Point estimates of backShift and Ling for synthetic data. We threshold the point estimate of backShift at $t = \pm 0.25$ to exclude those entries which are close to zero. We then threshold the estimate of Ling so that the two estimates have the same number of edges. In Setting 4, we threshold Ling at $t = \pm 0.25$ as backShift returns the empty graph. In Setting 3, it is not possible to achieve the same number of edges as all remaining coefficients in the point estimate of Ling are equal to one in absolute value. The transparency of the edges illustrates the relative magnitude of the estimated coefficients. We report the structural Hamming distance (SHD) for each graph. Precision and recall values are shown in Figure 3.1(c).

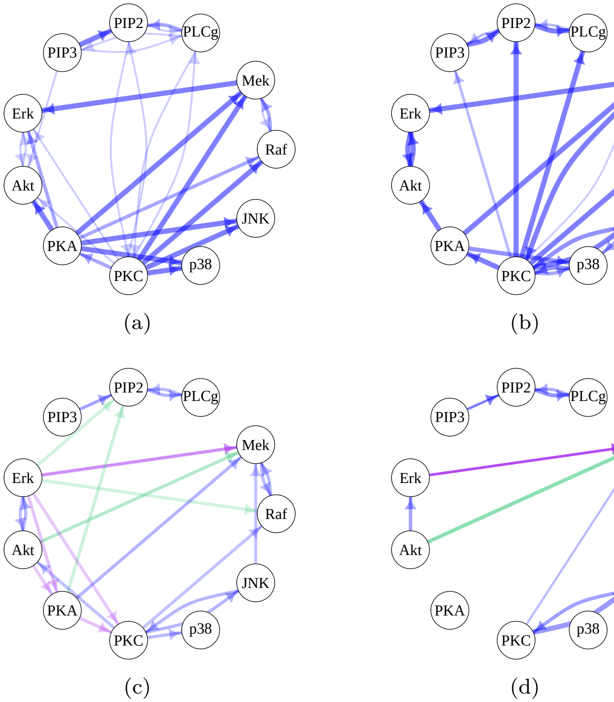
distribution. But if one of the error variables comes from the second mixture component, for example, the other error variables come from the second mixture component, too. In this sense, the data points are not independent anymore. This poses a challenge for LING which assumes an i.i.d. sample. We also cover a case (for $m_I = 0$) in which all assumptions of LING are satisfied (Scenario 4).

Figure 3.2 shows the estimated connectivity matrices for five different settings and Figure 3.1(c) shows the obtained precision and recall values. In Setting 1, $n = 1000$, $m_I = 1$ and there are no hidden variables. In Setting 2, n is increased to 10000 while the other parameters do not change. We observe that BACKSHIFT retrieves the correct adjacency matrix in both cases while LING's estimate is not very accurate. It improves slightly when increasing the sample size. In Setting 3, we do include hidden variables which violates the causal sufficiency assumption required for LING. Indeed, the estimate is worse than in Setting 2 but somewhat better than in Setting 1. BACKSHIFT retrieves two false positives in this case. Setting 4 is not feasible for BACKSHIFT as the distribution of the variables is identical in all environments (since $m_I = 0$). In Step 2 of the algorithm, FFDIAG does not converge and therefore the empty graph is returned. So the recall value is zero while precision is not defined. For LING all assumptions are satisfied and the estimate is more accurate than in the Settings 1–3. Lastly, Setting 5 shows that when increasing the intervention strength to 0.5, BACKSHIFT returns a few false positives. Its performance is then similar to LING which returns its most accurate estimate in this scenario. The stability selection results for BACKSHIFT are provided in Figure 3.5 in Appendix 3.E.

In short, these results suggest that the BACKSHIFT point estimates are close to the true graph if the interventions are sufficiently strong. Hidden variables make the estimation problem more difficult but the true graph is recovered if the strength of the intervention is increased (when increasing m_I to 1.5 in Setting 3, BACKSHIFT obtains a SHD of zero). In contrast, LING is unable to cope with hidden variables but also has worse accuracy in the absence of hidden variables under these shift interventions.

3.4.2. Flow cytometry data

The data published in Sachs et al. (2005) is an instance of a data set where the external interventions differ between the environments in $\mathcal{J}$ and might act on several compounds simultaneously (Eaton and Murphy,



(a) (b) (c) (d)

Figure 3.3.: Flow cytometry data. (a) Union of the consensus network (according to Sachs et al. (2005)), the reconstruction by Sachs et al. (2005) and the best *acyclic* reconstruction by Mooij and Heskes (2013). The edge thickness and intensity reflect in how many of these three sources that particular edge is present. (b) One of the *cyclic* reconstructions by Mooij and Heskes (2013). The edge thickness and intensity reflect the probability of selecting that particular edge in the stability selection procedure. For more details see Mooij and Heskes (2013). (c) BACKSHIFT point estimate, thresholded at ±0.35. The edge intensity reflects the relative magnitude of the coefficients and the coloring is a comparison to the union of the graphs shown in panels (a) and (b). Blue edges were also found in Mooij and Heskes (2013) and Sachs et al. (2005), purple edges are reversed and green edges were not previously found in (a) or (b). (d) BACKSHIFT stability selection result with parameters $\mathbb{E}(V) = 2$ and $\pi_{thr} = 0.75$. The edge thickness illustrates how often an edge was selected in the stability selection procedure.

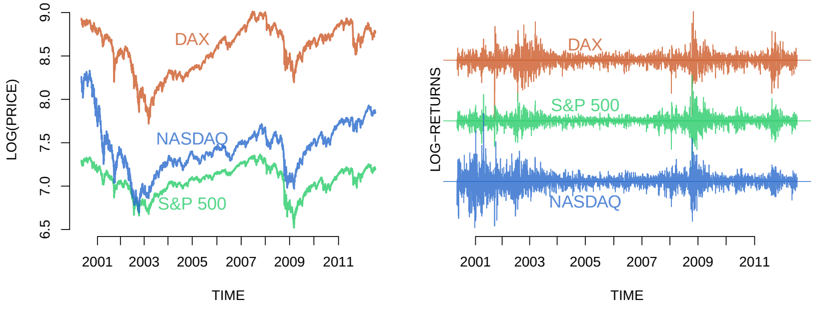
2007). There are nine different experimental conditions with each containing roughly 800 observations which correspond to measurements of the concentration of biochemical agents in single cells. The first setting corresponds to purely observational data.

In addition to the original work by Sachs et al. (2005), the data set has been described and analyzed in Eaton and Murphy (2007) and Mooij and Heskes (2013). We compare against the results of Mooij and Heskes (2013), Sachs et al. (2005) and the "well-established consensus", according to Sachs et al. (2005), shown in Figures 3.3(a) and 3.3(b). Figure 3.3(c) shows the (thresholded) BACKSHIFT point estimate. Most of the retrieved edges were also found in at least one of the previous studies. Five edges are reversed in our estimate and three edges were not discovered previously. Figure 3.3(d) shows the corresponding stability selection result with the expected number of falsely selected variables $\mathbb{E}(V) = 2$. This estimate is sparser in comparison to the other ones as it bounds the number of false discoveries. Notably, the feedback loops between PIP2 ↔ PLCg and PKC ↔ JNK were also found in Mooij and Heskes (2013).

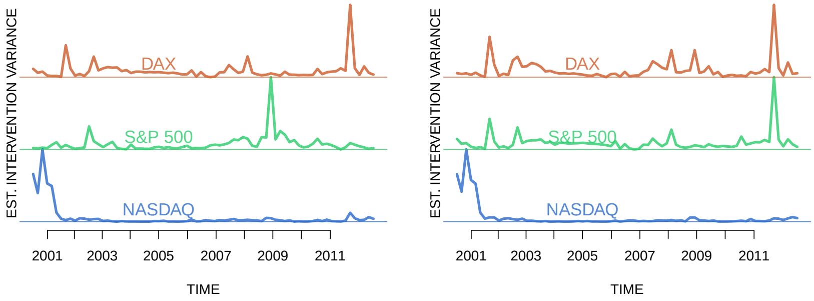
It is also noteworthy that we can check the model assumptions of shift interventions, which is important for these data as they can be thought of as changing the mechanism or activity of a biochemical agent rather than regulate the biomarker directly (Mooij and Heskes, 2013). If the shift interventions are not appropriate, we are in general not able to diagonalize the differences in the covariance matrices. Large off-diagonal elements in the estimate of the r.h.s in (3.7) indicate a mechanism change that is not just explained by a shift intervention as in (3.1). In four of the seven interventions environments with known intervention targets the largest mechanism violation happens directly at the presumed intervention target, see Appendix 3.C for details. It is worth noting again that the presumed intervention target had not been used in reconstructing the network and mechanism violations.

3.4.3. Financial time series

Finally, we present an application in financial time series where the environment is clearly changing over time. We consider daily data from three stock indices NASDAQ, S&P 500 and DAX for a period between 2000-2012 and group the data into 74 overlapping blocks of 61 consecutive days each. We take log-returns, as shown in panel (b) of Figure 3.4 and estimate the connectivity matrix, which is fully connected in this case and perhaps of



(a) Prices (logarithmic) (b) Daily log-returns



(c) backShift (d) Ling

Figure 3.4.: Financial time series with three stock indices: NASDAQ (blue; technology index), S&P 500 (green; American equities) and DAX (red; German equities). (a) Prices of the three indices between May 2000 and end of 2011 on a logarithmic scale. (b) The scaled log-returns (daily change in log-price) of the three instruments are shown. Three periods of increased volatility are visible starting with the dot-com bust on the left to the financial crisis in 2008 and the August 2011 downturn. (c) The scaled estimated intervention variance with the estimated backShift network. The three down-turns are clearly separated as originating in technology, American and European equities. (d) In contrast, the analogous Ling estimated intervention variances have a peak in American equities intervention variance during the European debt crisis in 2011.

not so much interest in itself. It allows us, however, to estimate the intervention strength at each of the indices according to (3.12), shown in panel (c). The intervention variances separate very well the origins of the three major down-turns of the markets on the period. Technology is correctly estimated by backShift to be at the epicenter of the dot-com crash in 2001 (NASDAQ as proxy), American equities during the financial crisis in 2008 (proxy is S&P 500) and European instruments (DAX as best proxy) during the August 2011 downturn.

3.5. Conclusion

We have shown that cyclic causal networks can be estimated if we obtain covariance matrices of the variables under unknown shift interventions in different environments. backShift leverages solutions to the linear assignment problem and joint matrix diagonalization and the part of the computational cost that depends on the number of variables is at worst cubic. We have shown sufficient and necessary conditions under which the network is fully identifiable, which require observations from at least three different environments. The strength and location of interventions can also be reconstructed.

Appendix 3.A Identifiability – Proof of Theorem 3.1

Proof. “if”: Let $\mathbf{D}'$ be a solution of (3.10). Let us write $\mathbf{D}'_{m\bullet}$ for the m-th row of $\mathbf{D}'$ and $\mathbf{D}_{m\bullet}$ for the m-th row of $\mathbf{D}$, $m = 1, \ldots, p$. Furthermore let us define $\mathbf{g}_m := \mathbf{D}^{-T}\mathbf{D}'_{m\bullet}$, $m = 1, \ldots, p$. We will show that at most one entry of this vector is nonzero. Note that by equation (3.7) we have $\mathbf{\Delta\Sigma}_{\mathbf{x},j} = \mathbf{D}^{-1}\mathbf{\Delta\Sigma}_{\mathbf{c},j}\mathbf{D}^{-T}$ for all $j \in \mathcal{J}$. By equation (3.7), $L(\mathbf{D}\mathbf{\Delta\Sigma}_{\mathbf{x},j}\mathbf{D}^T) = 0$. As $\mathbf{D}'$ solves equation (3.10), this implies $L(\mathbf{D}'\mathbf{\Delta\Sigma}_{\mathbf{x},j}\mathbf{D}'^T) = 0$ for all $j \in \mathcal{J}$. Hence the offdiagonal elements of $\mathbf{D}'\mathbf{\Delta\Sigma}_{\mathbf{x},j}\mathbf{D}'^T$ are zero, which implies

$$\mathbf{g}_{m'} \perp \mathbf{\Delta\Sigma}_{\mathbf{c},j}\mathbf{g}_m \text{ for all } m' \neq m \text{ and for all } j \in \mathcal{J}.$$

As the $\mathbf{g}_{m'}$ are linearly independent, this implies that for all pairs $j, j' \in \mathcal{J}$, $\mathbf{\Delta\Sigma}_{\mathbf{c},j}\mathbf{g}_m$ and $\mathbf{\Delta\Sigma}_{\mathbf{c},j'}\mathbf{g}_m$ are collinear i.e. for all (j, j') there exists a $\lambda_{j,j'} \in \mathbb{R}$ such that $\mathbf{\Delta\Sigma}_{\mathbf{c},j}\mathbf{g}_m = \lambda_{j,j'}\mathbf{\Delta\Sigma}_{\mathbf{c},j'}\mathbf{g}_m$ or $\lambda_{j,j'}\mathbf{\Delta\Sigma}_{\mathbf{c},j}\mathbf{g}_m = \mathbf{\Delta\Sigma}_{\mathbf{c},j'}\mathbf{g}_m$

Take arbitrary $k, l \in \{1, \ldots, p\}$ and choose $j, j' \in \mathcal{J}$ such that (3.13) is satisfied. By the argumentation above, there exists a $\lambda_{j,j'} \in \mathbb{R}$ such that $\mathbf{\Delta\Sigma}_{\mathbf{c},j}\mathbf{g}_m = \lambda_{j,j'}\mathbf{\Delta\Sigma}_{\mathbf{c},j'}\mathbf{g}_m$ or $\lambda_{j,j'}\mathbf{\Delta\Sigma}_{\mathbf{c},j}\mathbf{g}_m = \mathbf{\Delta\Sigma}_{\mathbf{c},j'}\mathbf{g}_m$. Without loss of generality let us assume the latter. Recall that both $\mathbf{\Delta\Sigma}_{\mathbf{c},j}$ and $\mathbf{\Delta\Sigma}_{\mathbf{c},j'}$ are diagonal matrices. Now condition (3.13) implies that the k-th or the l-th entry on the diagonal of $\lambda_{j,j'}\ \mathbf{\Delta\Sigma}_{\mathbf{c},j} - \mathbf{\Delta\Sigma}_{\mathbf{c},j'}$ is nonzero (or both). Hence, the k-th or the l-th entry of $\mathbf{g}_m$ s zero (or both). By repeating this argumentation for all k and l, at most one entry of $\mathbf{g}_m$ is nonzero. Thus, $\mathbf{D}'_{m\bullet} = \mathbf{D}^T\mathbf{g}_m = (\mathbf{g}_m^T\mathbf{D})^T$ is a multiple of one of the rows of $\mathbf{D}$.

By applying this argumentation for all $m = 1, \ldots, p$, each row of $\mathbf{D}'$ is a multiple of one of the rows of $\mathbf{D}$. As both $\mathbf{D}$ and $\mathbf{D}'$ are invertible, there exists a bijection between the rows of $\mathbf{D}'$ and $\mathbf{D}$ such that the corresponding rows are collinear. Furthermore, the diagonal of $\mathbf{D}'$ and $\mathbf{D}$ is $(1, \ldots, 1)$. Hence let us consider a bijection $\sigma : \{1, \ldots, p\} \mapsto \{1, \ldots, p\}$ such that the $\sigma(m)$-th row of $\mathbf{D}'$ is a multiple of the m-th row of $\mathbf{D}$, i.e. $\frac{1}{\mathbf{D}'_{\sigma(m),m}}\mathbf{D}'_{\sigma(m)\bullet} = \mathbf{D}_{m\bullet}$ for all $m = 1, \ldots, p$. We want to show that this bijection is the identity. First observe that, as the diagonal of $\mathbf{D}'$ and $\mathbf{D}$ is $(1, \ldots, 1)$, $\frac{1}{\mathbf{D}'_{\sigma(m),m}} = \mathbf{D}_{m,\sigma(m)}$ for all $m = 1, \ldots, p$. Now let us consider a cycle in this permutation , i.e. $m_1, \ldots, m_{\eta+1} = m_1$, $\eta > 1$, $m_\iota \neq m_\kappa$ for $1 \leq \iota < \kappa \leq \eta$ and with $\sigma(m_\iota) = m_{\iota+1}$ for $1 \leq \iota \leq \eta$. If this leads to a contradiction, we can conclude that σ is the identity. As $\mathbf{D}_{m,m} = 1$, $\mathbf{D}'_{\sigma(m),m} \neq 0$, i.e. $\mathbf{D}'_{m_{\iota+1},m_\iota} \neq 0$ for $1 \leq \iota \leq \eta$. This corresponds to a cycle in with product

$$\prod_{\iota=1,\ldots,\eta} \mathbf{D}'_{m_{\iota+1},m_\iota} = \prod_{\iota=1,\ldots,\eta} \frac{1}{\mathbf{D}_{m_\iota,m_{\iota+1}}}. \tag{3.15}$$

As $\mathbf{D}'$ is a solution of (3.10), $CP(\mathbf{I}-\mathbf{D}') < 1$, hence the product on the left hand side of equation (3.15) is in absolute value strictly smaller than 1, see (3.2). Analogously, as $\mathbf{D}_{m_\iota,m_{\iota+1}} \neq 0$ for $\iota = 1,\ldots,\eta$, the sequence $m_{\eta+1}, m_\eta, \ldots, m_1$ corresponds to a cycle with product

$$\prod_{\iota=1,\ldots,\eta} \mathbf{D}_{m_\iota,m_{\iota+1}}.$$

Using the same argumentation as for $\mathbf{D}'$, this product is in absolute value strictly smaller than 1, which contradicts (3.15). Hence such cycles of length ≥ 2 do not exist and σ is the identity. Hence, $\mathbf{D}' = \mathbf{D}$.

"only if": As above define $\mathbf{D}_{m\bullet}$ as the m-th row of $\mathbf{D}$ and let us write $\mathbf{u}_m \in \mathbb{R}^p$ for the m-th unit vector for $m = 1,\ldots,p$. Assume that (3.13) is not true, i.e. there exist $k, l \in \{1,\ldots,p\}$ such that for all $j, j' \in \mathcal{J}$,

$$(\mathbf{\Delta\Sigma}_{\mathbf{c},j})_{kk}(\mathbf{\Delta\Sigma}_{\mathbf{c},j'})_{ll} = (\mathbf{\Delta\Sigma}_{\mathbf{c},j})_{ll}(\mathbf{\Delta\Sigma}_{\mathbf{c},j'})_{kk}. \tag{3.16}$$

Without loss of generality let us fix a $j' \in \mathcal{J}$ with $(\mathbf{\Delta\Sigma}_{\mathbf{c},j'})_{kk} \neq 0$, and define $\lambda := (\mathbf{\Delta\Sigma}_{\mathbf{c},j'})_{ll}/(\mathbf{\Delta\Sigma}_{\mathbf{c},j'})_{kk}$. If such a j' does not exist, we can apply the same argumentation as below but with the k and l interchanged and $\lambda := 0$.

Note that the definition of λ does not depend on j and that by equation (3.7) we have $\mathbf{\Delta\Sigma}_{\mathbf{x},j} = \mathbf{D}^{-1}\mathbf{\Delta\Sigma}_{\mathbf{c},j}\mathbf{D}^{-T}$. Then, for $\delta \in \mathbb{R}$ we can define $\mathbf{D}'_{k\bullet} := \mathbf{D}_{k\bullet} + \delta\mathbf{D}_{l\bullet}$ and $\mathbf{D}'_{l\bullet} := \mathbf{D}_{l\bullet} - \delta\lambda\mathbf{D}_{k\bullet}$ and we obtain for all $j \in \mathcal{J}$

$$\begin{aligned}\mathbf{D}'^T_{l\bullet}\mathbf{\Delta\Sigma}_{\mathbf{x},j}\mathbf{D}'_{k\bullet} &= (\mathbf{u}_l - \delta\lambda\mathbf{u}_k)^T\mathbf{\Delta\Sigma}_{\mathbf{c},j}(\mathbf{u}_k + \delta\mathbf{u}_l)\\ &= \delta(\mathbf{\Delta\Sigma}_{\mathbf{c},j})_{ll} - \delta\lambda(\mathbf{\Delta\Sigma}_{\mathbf{c},j})_{kk}\\ &= 0.\end{aligned}$$

In the second equation we used (3.16). Furthermore, for small δ let us scale $\mathbf{D}'_{k\bullet}$ such that the k-th component of the vector is 1. Analogously, let us scale $\mathbf{D}'_{l\bullet}$ such that the l-th component of the vector is 1. Then we can define the matrix $\mathbf{D}'$ as the rows of $\mathbf{D}$ except for row k and l which are replaced by $\mathbf{D}'_{k\bullet}$ and $\mathbf{D}'_{l\bullet}$. By above reasoning, this matrix satisfies

$$\mathbf{D}'\mathbf{\Delta\Sigma}_{\mathbf{x},j}\mathbf{D}'^T \in \mathrm{Diag}(p)$$

for all $j \in \mathcal{J}$ and $\mathbf{D}'$ is invertible. Furthermore, the diagonal elements of $\mathbf{D}'$ are 1. Recall that the path-products of $\mathbf{I}-\mathbf{D}$ over cycles are in absolute value smaller than 1, see (3.2). For small δ, $\mathbf{I}-\mathbf{D}'$ is close to $\mathbf{I}-\mathbf{D}$ (in an arbitrary matrix norm) and hence the path products of $\mathbf{I}-\mathbf{D}'$ over cycles are in absolute value smaller than 1 as well. As $\mathbf{D}$ is invertible, $\mathbf{D}' \neq \mathbf{D}$. Hence the solution to (3.10) is not unique. This concludes the proof.

□

Appendix 3.B Polynomial-time algorithm

Here, we provide the necessary theoretical result to show that BACKSHIFT has a computational cost of $O(|\mathcal{J}| \cdot n \cdot p^2)$. Specifically, we show that Step 3 in Algorithm 1 can be cast in terms of the classical linear sum assignment problem, having a computational complexity of $O(p^3)$.

Theorem 3.3 *Let* $\mathbf{D} \in \mathbb{R}^{p\times p}$ *be a matrix with* $CP(\mathbf{D}) < 1$, *diag*$(\mathbf{D}) \equiv 1$ *and* $\mathbf{D}_{k,l} \neq 0$ *for* $k,l \in \{1,\ldots,p\}$. *For* $\mathbf{D}' \in \mathbb{R}^{p\times p}$ *define*

$$P(\mathbf{D}') := \prod_{k,l} |\mathbf{D}'_{k,l}|.$$

Furthermore define

$$\begin{aligned}\mathcal{D}_p := \{\mathbf{D}' : \ & \textit{There exists a permutation } \sigma \textit{ of } \{1,\ldots,p\} \textit{ such that the} \\ & \sigma(m)\textit{-th row of } \mathbf{D} \textit{ is collinear to the } m\textit{-th row of } \mathbf{D}' \\ & \textit{and diag}(\mathbf{D}') \equiv 1 \ \}.\end{aligned}$$

Then,

$$\mathbf{D} = \arg\min_{\mathbf{D}'\in\mathcal{D}_p} P(\mathbf{D}') = \arg\min_{\mathbf{D}'\in\mathcal{D}_p} \log P(\mathbf{D}').$$

Proof. Let $\mathbf{D}' \in \mathcal{D}_p$ with $\mathbf{D}' \neq \mathbf{D}$. Let us write $\mathbf{D}_{m\bullet}$ for the m-th row of $\mathbf{D}$ and analogously $\mathbf{D}'_{m\bullet}$ for the m-th row of $\mathbf{D}'$, $m = 1,\ldots,p$. Now let σ be a permutation such that the $\sigma(m)$-th row of $\mathbf{D}$ is collinear to the m-th row of $\mathbf{D}'$. As $\mathbf{D}' \neq \mathbf{D}$, we have that $\sigma \neq \mathrm{Id}$. As $\mathrm{diag}(\mathbf{D}') \equiv 1$,

$$\frac{1}{\mathbf{D}_{\sigma(m),m}}\mathbf{D}_{\sigma(m)\bullet} = \mathbf{D}'_{m\bullet}.$$

It immediately follows that

$$\left(\prod_{m=1,\ldots,p} \frac{1}{|\mathbf{D}_{\sigma(m),m}|} \right)^p P(\mathbf{D}) = P(\mathbf{D'}).$$

As $CP(\mathbf{D}) < 1$ and σ is not the identity, $\prod_{m=1,\ldots,p} |\mathbf{D}_{\sigma(m),m}| < 1$. As all elements of $\mathbf{D}$ and $\mathbf{D'}$ are nonzero, $P(\mathbf{D}) > 0$ and $P(\mathbf{D'}) > 0$. Hence, $P(\mathbf{D'}) > P(\mathbf{D})$. This concludes the proof.

□

Remark: We can define the relative loss function of moving row k to row l as

$$\ell(k,l) = -\log(|\mathbf{D}'_{k,l}|).$$

Then the linear assignment problem that minimizes this problem also yields the correct permutation for Step 3 in Algorithm 1 if it exists, i.e. the permutation σ on $\{1,\ldots,p\}$ that minimizes

$$\sum_{k=1}^{p} \ell(k,\sigma(k))$$

satisfies that $\mathbf{D}'_{m\bullet}$ is collinear to $\mathbf{D}_{\sigma(m)\bullet}$.

Remark: Allowing for self-loops would lead to an identifiability problem, independent of the method. For every model with self-loops and $CP < 1$ there is a model without self-loops and $CP \leq 1$ yielding the same observational distribution in equilibrium. The connectivity matrix without self-loops can thus be seen as a representative of a whole class of connectivity matrices that allow self-loops. Specifically, if the connectivity matrix with self-loops is $\mathbf{B}^*$, define matrix $\mathbf{T}$ by `PermuteAndScale`$(\mathbf{I}-\mathbf{B}^*) = \mathbf{T}(\mathbf{I}-\mathbf{B}^*)$, where `PermuteAndScale`() is the operation defined in Step 3 of the BACKSHIFT algorithm. Technically, `PermuteAndScale`() is only defined for matrices that are nonzero outside of the diagonal. Using similar arguments as in Theorem 3.3, `PermuteAndScale`() can be extended to arbitrary matrices with nonzero diagonal elements. To be more precise, there exists a matrix $\mathbf{T}$ such that $CP(\mathbf{T}(\mathbf{I}-\mathbf{B}^*)) \leq 1$, $\mathrm{diag}(\mathbf{T}(\mathbf{I}-\mathbf{B}^*)) \equiv 1$ and such that $\mathbf{T}$ is the product of a diagonal scaling matrix with a permutation matrix. Then define $\mathbf{B}_{new} := \mathbf{I} - \mathbf{T}(\mathbf{I}-\mathbf{B}^*)$, $\mathbf{e}_{j,new} = \mathbf{T}\mathbf{e}_j$ and $\mathbf{c}_{j,new} = \mathbf{T}\mathbf{c}_j$ for all $j \in \mathcal{J}$. As $\mathbf{T}$ is the product of a diagonal scaling matrix with a permutation matrix, assumptions (B) and (C) are still fulfilled and $\mathbf{x}_{j,new} = (\mathbf{I} - \mathbf{B}_{new})^{-1}(\mathbf{e}_{j,new} + \mathbf{c}_{j,new}) = (\mathbf{I} - \mathbf{B}^*)^{-1}(\mathbf{e}_j + \mathbf{c}_j) = \mathbf{x}_j$ for all $j \in \mathcal{J}$. This implies that the two matrices $\mathbf{B}^*$ with self-loops and $\mathbf{B}_{new}$ without self-loops (since it has zeroes on the diagonal by construction) have both $CP \leq 1$ and yield the same distribution.

Appendix 3.C Intervention variances and model misspecification

The method allows to validate and check the assumptions to some extent. This is especially important in the data of Sachs et al. (2005) as pointed out in Mooij and Heskes (2013). The interventions can mostly be thought of as not changing the concentration of a biochemical agent but rather changing the activity of the agent, for example by inhibiting the reactions in which the agent is involved (Mooij and Heskes, 2013). Under such a mechanism change, it is doubtful whether the interventions are well approximated by our model (3.3) with independent shift-interventions. We can check the assumptions by the success of the joint diagonalization procedure. Specifically, we get an empirical version of (3.7) when plugging in the estimators and can check whether all off-diagonal elements on the right hand side of (3.7) are small or vanishing. We list below results for the seven experimental intervention conditions whose target is well described in Mooij and Heskes (2013). The element on the right-hand side of (3.7) with the largest absolute value is selected. We use now the Gram instead of the covariance matrix to be also sensitive to model-violations of the additional assumption (C'), see §3.D, though the results are almost identical whether using the Gram or covariance matrix. These large off-diagonal elements indicate a violated mechanism in the sense that the model (3.3) does not fit very well, because either the interventions have not been of the assumed shift-type or the causal mechanism in which the agent is involved has changed under the intervention.

Exp.	Reagent	Intervention	largest mech. violation
3	Akt-Inhibitor	inhibits AKT activity	PLCg $\leftrightarrow$ PKA
4	G0076	inhibits **PKC** activity	**PKC** $\leftrightarrow$ PIP2
5	Psitectorigenin	inhibits **PIP2** abundance	**PIP2** $\leftrightarrow$ PKA
6	U0126	inhibits **MEK** activity	**MEK** $\leftrightarrow$ PKA
7	LY294002	changes PIP2/PIP3 mech.	PKA $\leftrightarrow$ JNK
8	PMA	activates PKC activity	MEK $\leftrightarrow$ PKA
9	β2CAMP	activates **PKA** activity	**PKA** $\leftrightarrow$ PKC

The table above lists the results for the seven experimental conditions where we know the intervention mechanism, at least approximately. The results are interesting in that the most violated mechanism (the largest entry in the off-diagonal matrix on the right-hand side of the empirical version of (3.7)) occurs in 4 of the 7 experimental conditions directly at the intervention target. In 3 of these 4 cases, the violated mechanism concerns a relation that has a large entry in the estimated connectivity matrix. This corresponds well with the model of activity interventions in Mooij and Heskes (2013). Note that we have not made use of the intervention targets in the estimation procedure. The interesting point is that we can use the model violations to estimate with some success where the interventions occurred.

Appendix 3.D Beyond covariances

For the method above, we exploit differences in the covariance of observations across different environments. We can also exploit a shift in the mean of the intervention strength $\mathbf{c}$ (and consequently in the observations $\mathbf{x}$) when strengthening the condition (C) to (C'). Specifically, we require for (C') that in each environment $j \in \mathcal{J}$ the shift in the mean $E(\mathbf{c}_j)$ equals zero for all variables except at most one variable. The variable with a non-zero shift in the mean can change from one environment to another. Note that the counterpart of (3.5) when using the Gram matrix instead of the covariance matrix reads

$$(\mathbf{I}-\mathbf{B})\mathbf{G}_{\mathbf{x},j}(\mathbf{I}-\mathbf{B})^T = \mathbf{G}_{\mathbf{c},j} + \mathbf{G}_{\mathbf{e}}. \tag{3.17}$$

Under the stronger version (C'), the difference across environments of the right-hand side in (3.17) is again a diagonal matrix and we can proceed just as above, by replacing the covariance matrices with Gram matrices throughout. If the assumption (C’) is satisfied, this allows identifiability of the graph in a wider range of settings (Theorem 3.1 can be adapted in a straightforward manner by again replacing covariances with Gram matrices) but requires the stricter condition (C’). Since in practice it is often unclear whether the stricter condition is approximately true, we work mainly with the weaker assumption (C) and exploit only shifts in the covariance matrices.

Appendix 3.E Additional figures

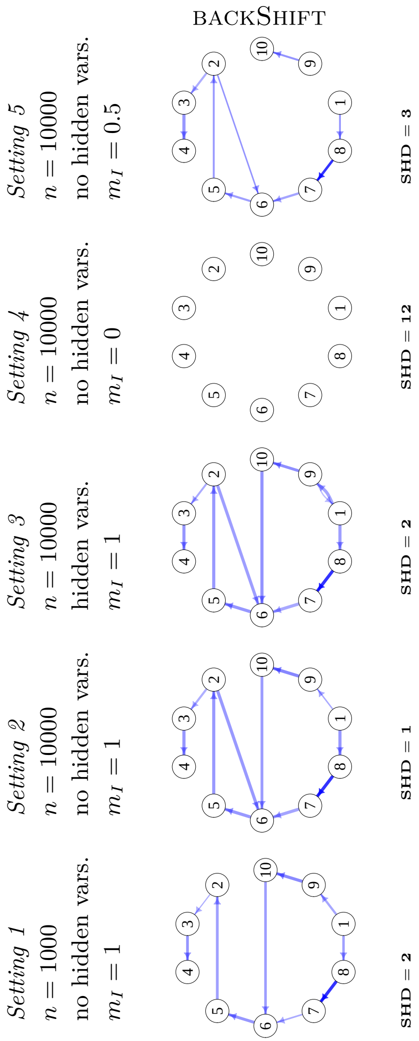


Figure 3.5.: Synthetic data. Stability selection results for backShift with parameters $\mathbb{E}(V) = 2$ and $\pi_{thr} = 0.75$. The intensity of the edges illustrates the relative magnitude of the estimated coefficients, the width shows how often an edge was selected. The edge from node 6 to node 10 is associated with the smallest coefficient in absolute value. It is retained in none of the settings in the stability selection procedure.

Chapter 4.

Invariant Causal Prediction for nonlinear models

An important problem in many domains is to predict how a system will respond to interventions. This task is inherently linked to estimating the system's underlying causal structure. To this end, Invariant Causal Prediction (ICP) (Peters et al., 2016) has been proposed which learns a causal model exploiting the invariance of causal relations using data from different environments. When considering linear models, the implementation of ICP is relatively straightforward. However, the nonlinear case is more challenging due to the difficulty of performing nonparametric tests for conditional independence.

In this work, we present and evaluate an array of methods for nonlinear and nonparametric versions of ICP for learning the causal parents of given target variables. We find that an approach which first fits a nonlinear model with data pooled over all environments and then tests for differences between the residual distributions across environments is quite robust across a large variety of simulation settings. We call this procedure "invariant residual distribution test". In general, we observe that the performance of all approaches is critically dependent on the true (unknown) causal structure and it becomes challenging to achieve high power if the parental set includes more than two variables.

As a real-world example, we consider fertility rate modeling which is central to world population projections. We explore predicting the effect of hypothetical interventions using the accepted models from nonlinear ICP. The results reaffirm the previously observed central causal role of child mortality rates.

4.1. Introduction

Invariance based causal discovery (Peters et al., 2016) relies on the observation that the conditional distribution of the target variable Y given its direct causes remains invariant if we intervene on variables other than Y. While the proposed methodology in Peters et al., 2016 focuses on linear models, we extend Invariant Causal Prediction to nonlinear settings. We first introduce the considered structural causal models in §4.1.1 and review related approaches to causal discovery in §4.1.2. The invariance approach to causal discovery from Peters et al., 2016 is briefly summarized in §4.1.3 and we outline our contribution in §4.1.4. In §4.1.5 we introduce the problem of fertility rate modeling which we consider as a real-world example throughout this work.

4.1.1. Structural causal models

Assume an underlying structural causal model (also called structural equation model) (e.g. Pearl, 2009)

$$\begin{aligned} Z_1 &\leftarrow g_1(Z_{\mathrm{pa}_1}) + \eta_1, \\ Z_2 &\leftarrow g_2(Z_{\mathrm{pa}_2}) + \eta_2, \\ &\vdots \\ Z_q &\leftarrow g_q(Z_{\mathrm{pa}_q}) + \eta_q, \end{aligned}$$

for which the functions g_k, $k = 1, \ldots, q$, as well as the parents $\mathrm{pa}_k \subseteq \{1, \ldots, q\} \setminus \{k\}$ of each variable are unknown. Here, we have used the notation $Z_S = (Z_{i_1}, \ldots, Z_{i_s})$ for any set $S = \{i_1, \ldots, i_s\} \subseteq \{1, \ldots, q\}$. We assume the corresponding directed graph to be acyclic. We further require the noise variables $\eta_1, \ldots, \eta_q$ to be jointly independent and to have zero mean, i.e. we assume that there are no hidden variables.

Due to its acyclic structure, it is apparent that such a structural causal model induces a joint distribution P over the observed random variables. Interventions on the system are usually modeled by replacing some of the structural assignments (e.g. Pearl, 2009). If one intervenes on variable Z_3, for example, and sets it to the value 5, the system again induces a distribution over $Z_1, \ldots, Z_q$, that we denote by $P(\cdot|do(Z_3 \leftarrow 5))$. It is usually different from the observational distribution P. We make no counterfactual assumptions here: we assume a new realization η is drawn from the noise distribution as soon as we make an intervention.[1]

4.1.2. Causal discovery

In causal discovery (also called structure learning) one tries to reconstruct the structural causal model or its graphical representation from its joint distribution (e.g. Chickering, 2002b; Hauser and Bühlmann, 2015; Heckerman, 1997; Pearl, 2009; Peters and Bühlmann, 2014; Peters et al., 2017; Spirtes et al., 2000).

Existing methods for causal structure learning can be categorized along a number of dimensions, such as (i) using purely observational data vs. using a combination of interventional and observational data; (ii) score-based vs. constraint-based vs. "other" methods; (iii) allowing vs. precluding the existence of hidden confounders; (iv) requiring vs. not requiring faithfulness;[2] (v) type of object that the method estimates. Moreover, different methods vary by additional assumptions they require. In the following, we give brief descriptions of the most common methods for causal structure learning[3].

The PC algorithm (Spirtes et al., 2000) uses observational data only and estimates the Markov equivalence class of the underlying graph structure, based on (conditional) independence tests under a faithfulness assumption. The presence of hidden confounders is not allowed. Based on the PC algorithm, the IDA algorithm (Maathuis et al., 2009) computes bounds on the identifiable causal effects.

The FCI algorithm is a modification of the PC algorithm. It also relies on purely observational data while it allows for hidden confounders. The output of FCI is a partial ancestral graph (PAG), i.e. it estimates the Markov

---

[1] The new realization of η under an intervention and the realization under observational data can be assumed to be independent. However, such an assumption is untestable since we can never observe realizations under different interventions simultaneously and we do not make statements or assumptions about the joint distribution of observational and interventional settings.

[2] A distribution satisfies faithfulness and the global Markov condition with respect to a graph G if the following statement holds for all disjoint sets A, B, and C of variables: A is independent of B, given C, if and only if A is d-separated (in G) from B, given C. The concept of d-separation (Pearl, 1985, 1988) is defined in Peters et al. (2017, Def. 6.1), for example.

[3] Also see Heinze-Deml et al., 2018a for a review and empirical comparison of recently proposed causal structure learning algorithms.

equivalence class of the underlying maximal ancestral graph (MAG). Faster versions, RFCI and FCI+, were proposed by Colombo et al. (2012) and Claassen et al. (2013), respectively.

The PC, FCI, RFCI and FCI+ algorithms are formulated such that they allow for an independence oracle that indicates whether a particular (conditional) independence holds in the distribution. These algorithms are typically applied in the linear Gaussian setting where testing for conditional independence reduces to testing for vanishing partial correlation.

One of the most commonly known score-based methods is greedy equivalence search (GES). Using observational data, it greedily searches over equivalence classes of directed acyclic graphs for the best scoring graph (all graphs within the equivalence class receive the same score) where the score is given by the Bayesian information criterion, for example. Thus, GES is based on an assumed parametric model such as linear Gaussian structural equations or multinomial distributions. The output of GES is the estimated Markov equivalence class of the underlying graph structure. Heckerman (1997) describe a score-based method with a Bayesian score.

Greedy interventional equivalence search (GIES) extends GES to operate on a combination of interventional and observational data. The targets of the interventions need to be known and the output of GIES is the estimated interventional Markov equivalence class. The latter is typically smaller than the Markov equivalence class obtained when using purely observational data.

Another group of methods makes restrictive assumptions which allows for obtaining full identifiability. Such assumptions include non-Gaussianity (Shimizu et al., 2006) or equal variances (Peters et al., 2013) of the errors or non-linearity of the structural equations in additive noise models (Hoyer et al., 2008; Peters and Bühlmann, 2014).

Instead of trying to infer the whole graph, we are here interested in settings, where there is a target variable Y of special interest. The goal is to infer both the parental set S^* for the target variable Y and confidence bands for the causal effects.

4.1.3. Invariance based causal discovery

This work builds on the method of Invariant Causal Prediction (ICP) (Peters et al., 2016) and extends it in several ways. The method's key observation is that the conditional distribution of the target variable Y given its direct causes remains invariant if we intervene on variables other than Y. This follows from an assumption sometimes called autonomy or modularity (Aldrich, 1989; Haavelmo, 1944; Hoover, 1990; Pearl, 2009; Schölkopf et al., 2012). In a linear setting, this implies, for example, that regressing Y on its direct causes yields the same regression coefficients in each environment, provided we have an infinite amount of data. In a nonlinear setting, this can be generalized to a conditional independence between an index variable indicating the interventional setting and Y, given X; see Eq. (4.3). The method of ICP assumes that we are given data from several environments. It searches for sets of covariates, for which the above property of invariance cannot be rejected. The method then outputs the intersection of all such sets, which can be shown to be a subset of the true set with high probability, see §4.2.1 and Algorithm 2 in Appendix 4.B for more details. Such a coverage guarantee is highly desirable, especially in causal discovery, where information about ground truth is often sparse.

In many real life scenarios, however, relationships are not linear and the above procedure can fail: The true set does not necessarily yield an invariant model and the method may lose its coverage guarantee, see Example 4.4. Furthermore, environments may not come as a categorical variable but as a continuous variable instead. In this work, we extend the concept of ICP to nonlinear settings and continuous environments. The following paragraph summarizes our contributions.

4.1.4. Contribution

Our contributions are fivefold.

Conditional independence tests. We extend the method of ICP to nonlinear settings by considering conditional independence tests. We discuss in §4.3 and in more length in Appendix 4.B several possible nonlinear and nonparametric tests for conditional independence of the type (4.3) and propose alternatives. There has been some progress towards nonparametric independence tests (Bergsma and Dassios, 2014; Blum et al., 1961; Hoeffding, 1948; Rényi, 1959; Székely et al., 2007; Zhang et al., 2011). However, in the general nonparametric case, no known non-trivial test of conditional independence has (even asymptotically) a type I error rate less than the pre-specified significance level. This stresses the importance of empirical evaluation of conditional independence tests.

Defining sets. We discuss in §4.2.2 cases of poor identifiability of the causal parents. If there are highly correlated variables in the dataset, we might get an empty estimator if we follow the approach proposed in (Peters et al., 2016). We can, however, extract more information via defining sets. The results are to some extent comparable to similar issues arising in multiple testing (Goeman and Solari, 2011). For example, if we know that the parental set of a variable Y is either $S = \{1, 3\}$ or $S = \{2, 3\}$, we know that $\{3\}$ has to be a parent of Y. Yet we also want to explore the information that one variable out of the set $\{1, 2\}$ also has to be causal for the target variable Y, even if we do not know which one out of the two.

Confidence bands for causal effects. Beyond identifying the causal parents, we can provide nonparametric or nonlinear confidence bands for the strength of the causal effects, as shown in §4.2.3.

Prediction under interventions. Using the accepted models from nonlinear ICP, we are able to forecast the average causal effect of external interventions. We will discuss this at hand of examples in §4.2.4.

Software. R (R Core Team, 2017) code for nonlinear ICP is provided in the package `nonlinearICP`. The proposed conditional independence tests are part of the package `CondIndTests`. Both packages are available from CRAN.

4.1.5. Fertility rate modeling

At the hand of the example of fertility rate modeling, we shall explore how to exploit the invariance of causal models for causal discovery in the nonlinear case.

Developing countries have a significantly higher fertility rate compared to Western countries. The fertility rate can be predicted well from covariates such as ‘infant mortality rate’ or ‘GDP per capita’. Classical prediction models, however, do not answer whether an active intervention on some of the covariates leads to a change in the fertility rate. This can only be answered by exploiting causal knowledge of the system.

Traditionally, in statistics the methods for establishing causal relations rely on carefully designed randomized studies. Often, however, such experiments cannot be performed. For instance, factors like ‘infant mortality rate’ are highly complex and cannot be changed in isolation. We may still be interested in the effect of a policy that aims at reducing the infant mortality rate but this policy cannot be randomly assigned to different groups of people within a country.

There is a large body of work that is trying to explain changes in fertility; for an interesting overview of different theories see Hirschman (1994) and the more recent Huinink et al. (2015). There is not a single established theory for changes in fertility and we should clarify in the beginning that all models we will be using will have shortcomings, especially the shortcoming that we might not have observed all relevant variables. We would nevertheless like to take the fertility data as an example to establish a methodology that allows data-driven answers; discussing potential shortfalls of the model is encouraged and could be beneficial in further phrasing the right follow-up questions and collecting perhaps more suitable data.

An interesting starting point for us was the work of Raftery et al. (1995) and very helpful discussions with co-author Adrian Raftery. That work tries to distinguish between two different explanatory models for a decline in fertility in Iran. One model argues that economic growth is mainly responsible; another argues that transmission of new ideas is the primary factor (ideation theory). What allows a distinction between these models is that massive economic growth started in 1955 whereas ideational changes occurred mostly 1967 and later. Since the fertility began to drop measurably already in 1959, the demand theory seems more plausible and the authors conclude that reduced child mortality is the key explanatory variable for the reduction in fertility (responsible for at least a quarter of the reduction).

Note the way we decide between two potential causal theories for a decline in fertility: if a causal model is valid, it has to be able to explain the decline consistently. In particular, the predictions of the model have to be valid for all time-periods, including the time of 1959 with the onset of the fertility decline. The ideation theory wrongly places the onset of fertility decline later and is thus dismissed as less plausible.

The invariance approach of Peters et al. (2016) we follow here for linear models has a similar basic idea: a causal model has to work consistently. In our case, we choose geographic location instead of time for the example and demand that a causal model has to work consistently across geographic locations or continents. We collect all potential models that show this invariance and know that *if* the underlying assumption of causal sufficiency is true and we have observed all important causal variables *then* the causal model will be in the set of retained models. Clearly, there is room for a healthy and interesting debate to what extent the causal sufficiency assumption is violated in the example. It has been argued, however, that missing variables do not allow for any invariant model, which renders the method to remain conservative (Peters et al., 2016, Prop. 5).

We establish a framework for causal discovery in nonlinear models. Incidentally, the approach also identifies reduced child mortality as one of key explanatory variables for a decline in fertility.

4.2. Nonlinear Invariant Causal Prediction

We first extend the approach of Peters et al. (2016) to nonlinear models, before discussing defining sets, nonparametric confidence bands and prediction under interventions.

4.2.1. Invariance approach for causal discovery

Peters et al., 2016 proposed an invariance approach in the context of linear models. We describe the approach here in a notationally slightly different way that will simplify statements and results in the nonlinear case and allow for more general applications. Assume that we are given a structural causal model (SCM) over variables (Y, X, E), where Y is the target variable, X the predictors and E so-called environmental variables.

Definition 4.1 (Environmental variables) *We know or assume that the variables E are neither descendants nor parents of Y in the causal DAG of (Y, X, E). If this is the case, we call E environmental variables.*

In Peters et al., 2016, the environmental variables were given and non-random. Note that the definition above treats the variables as random but we can in practice condition on the observed values of E. The definition above excludes the possibility that there is a direct causal connection between one of the variables in E and Y. We will talk in the following about the triple of random variables (Y, X, E), where the variable X of predictor variables is indexed by $X_1, \dots, X_p$. With a slight abuse of notation, we let $S^* \subseteq \{1, \dots, p\}$ be the indices of X that are causal parents pa_Y of Y. Thus, the structural equation for Y can be written as

$$Y \leftarrow f(X_{S^*}) + \varepsilon, \tag{4.1}$$

where $f : \mathbb{R}^{|S^*|} \to \mathcal{Y}$. We let $\mathcal{F}$ be the function class of f and let $\mathcal{F}_S$ be the subclass of functions that depend only on the set $S \subseteq \{1, \dots, p\}$ of variables. With this notation we have $f \in \mathcal{F}_{S^*}$.

The assumption of no direct effect of E on Y is analogous to the typical assumptions about instrumental variables (Angrist et al., 1996; Imbens, 2014). See §5 in Peters et al., 2016 for a longer discussion on the relation between environmental variables and instrumental variables. The two main distinctions between environmental and instrumental variables are as follows. First, we do not need to test for the "weakness" of instrumental/environmental variables since we do not assume that there is a causal effect from E on the variables in X. Second, the approaches are used in different contexts. With instrumental variables, we assume the graph structure to be known typically and want to estimate the strength of the causal connections, whereas the emphasis is here on both causal discovery (what are the parents of a target?) and then also inference for the strength of causal effects. With a single environmental variable, we can identify in some cases multiple causal effects whereas the number of instrumental variables needs to match or exceed the number of variables in instrumental variable regression. The instrumental variable approach, on the other hand, can correct for unobserved confounders between the parents and the target variable if their influence is linear, for example. In these cases, our approach could remain uninformative (Peters et al., 2016, Proposition 5).

Example 4.2 (Fertility data) *In this work, we analyze a data set provided by the United Nations, 2013. Here, Y, X and E correspond to the following quantities:*

(a) $Y \in \mathbb{R}$ *is the total fertility rate (TFR) in a country in a given year,*

(b) $X \in \mathbb{R}^9$ *are potential causal predictor variables for TFR:*

- *IMR – infant mortality rate*
- *Q5 – under-five mortality rate*
- *Education expenditure (% of GNI)*
- *Exports of goods and services (% of GDP)*

  - *GDP per capita (constant 2005 US\$)*
  - *GDP per capita growth (annual %)*
  - *Imports of goods and services (% of GDP)*
  - *Primary education (% female)*
  - *Urban population (% of total)*

(c) $E \in \{C_1, C_2, C_3, C_4, C_5, C_6\}$ *is the continent of the country, divided into the categories Africa, Asia, Europe, North and South America and Oceania. If viewed as a random variable (which one can argue about), the assumption is that the continent is not a descendant of the fertility rate, which seems plausible. For an environmental variable, the additional assumption is that the TFR in a country is only indirectly (that is via one of the other variables) influenced by which continent it is situated on (cf. Figure 4.1).*

Clearly, the choices above are debatable. We might for example also want to include some ideation-based variables in X (which are harder to measure, though) and also take different environmental variables E such as time instead of geographic location. We could even allow for additive effects of the environmental variable on the outcome of interest (such as a constant offset for each continent) but we do not touch this debate much more here as we are primarily interested in the methodological development.

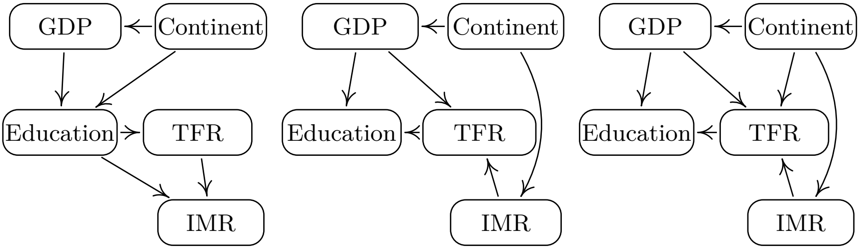


Figure 4.1.: Three candidates for a causal DAG with target total fertility rate (TFR) and four potential causal predictor variables. We would like to infer the parents of TFR in the true causal graph. We use the continent as the environment variable E. If the true DAG was one of the two graphs on the left, the environmental variable would have no direct influence on the target variable TFR and 'Continent' would be a valid environmental variable, see Definition 4.1.

The basic yet central insight underlying the invariance approach is the fact that for the true causal parental set $S^* := \mathrm{pa}_Y$ we have the following conditional independence relation under Definition 4.1 of environmental variables:

$$Y \perp\!\!\!\perp E \mid X_{S^*}. \tag{4.2}$$

This follows directly from the local Markov condition (e.g. Lauritzen, 1996). The goal is to find S^* by exploiting the above relation (4.2). Suppose we have a test for the null hypothesis

$$H_{0,S}: \qquad Y \perp\!\!\!\perp E \mid X_S. \tag{4.3}$$

It was then proposed in Peters et al., 2016 to define an estimate $\hat{S}$ for the parental set S^* by setting

$$\hat{S} := \bigcap_{S: H_{0,S} \text{ not rejected}} S. \tag{4.4}$$

Here, the intersection runs over all sets S, s.t. $E \cap S = \emptyset$. If the index set is empty, i.e. $H_{0,S}$ is rejected for all sets S, we define $\hat{S}$ to be the empty set. If we can test (4.3) with the correct type I error rate in the sense that

$$P\big(H_{0,S^*} \text{ is rejected at level } \alpha\big) \leq \alpha, \tag{4.5}$$

then we have as immediate consequence the desired statement

$$P\big(\hat{S} \subseteq S^*\big) \;\geq\; P\big(H_{0,S^*} \text{ accepted}\big) \;\geq\; 1 - \alpha.$$

This follows directly from the fact that S^* is accepted with probability at least $1 - \alpha$ since H_{0,S^*} is true; see Peters et al., 2016 for details.

In the case of linear models, the method proposed by Peters et al. (2016, Eq. (16)) considers a set S as invariant if there exist linear regression coefficients β and error variance σ which are identical across all environments. We consider the conditional independence relation in (4.3) as a generalization, even for linear relations. In the following example the regression coefficients are the same in all environments, and the residuals have the same mean and variance, but differ in higher order moments (cf. Peters et al., 2016, Eq. (3)):

Example 4.3 *Consider a discrete environmental variable E. If in $E = 1$ we have*

$$Y = 2X + N, N \perp\!\!\!\perp X,$$

and in $E = 2$

$$Y = 2X + M, M \perp\!\!\!\perp X,$$

where M *and* N *have the same mean and variance but differ in higher order moments. In this case, we would have* $E \not\!\perp\!\!\!\perp Y \mid X$*, but the hypothesis "same linear regression coefficients and error variance" cannot be rejected.*

The question remains how to test (4.3). If we assume a linear function f in the structural equation (4.1), then tests that can guarantee the level as in (4.5) are available (Peters et al., 2016). The following examples show what could go wrong if the data contain nonlinearities that are not properly taken into account.

Example 4.4 (Linear model and nonlinear data) *Consider the following SCM, in which* X_2 *and* X_3 *are direct causes of* Y*.*

$$\begin{aligned} X_1 &\leftarrow E + \eta_X \\ X_2 &\leftarrow \sqrt{3X_1 + \eta_{X_1}} \\ X_3 &\leftarrow \sqrt{2X_1 + \eta_{X_2}} \\ Y &\leftarrow X_2^2 - X_3^2 + \eta_Y \end{aligned}$$

Due to the nonlinear effect, a linear regression from Y *on* X_2 *and* X_3 *does not yield an invariant model. If we regress* Y *on* X_1*, however, we obtain invariant prediction and independent residuals. In this sense, the linear version of ICP fails but it still chooses a set of ancestors of* Y *(it can be argued that this failure is not too severe).*

Example 4.5 (Linear model and nonlinear data) *In this example, the model misspecification leads to a wrong set that includes a descendant of* Y*. Consider the following SCM*

$$\begin{aligned} X_1 &\leftarrow E + \eta_1 \\ Y &\leftarrow f(X_1) + \eta_Y \\ X_2 &\leftarrow g(X_1) + \gamma Y + \eta_2 \end{aligned}$$

with independent Gaussian error terms. Furthermore, assume that

$$\begin{aligned} \forall x \in \mathbb{R} : \qquad & f(x) = \alpha x + \beta h(x) \\ \forall x \in \mathbb{R} : \qquad & g(x) = h(x) - \gamma f(x) \\ & \beta\gamma^2 - \gamma = -\beta \mathrm{var}(\eta_2)/\mathrm{var}(\eta_Y) \end{aligned}$$

for some α, β *and* $h : \mathbb{R} \to \mathbb{R}$. *Then, in the limit of an infinite sample size, the set* $\{X_1, X_2\}$ *is the only set that, after a linear regression, yields residuals that are independent of* E. *(To see this write* $Y = f(X_1) + \eta_Y$ *as a linear function in* X_1, X_2 *and show that the covariance between the residuals and* X_2 *is zero.) Here, the functions have to be "fine-tuned" in order to make the conditional* $Y|X_1, X_2$ *linear in* X_1 *and* X_2.[4] *As an example, one may choose* $Y \leftarrow X_1 + 0.5X_1^2 + \eta_Y$ *and* $X_2 \leftarrow 0.5X_1^2 - X_1 + Y + \eta_2$ *and* η_1, η_Y, η_2 *i.i.d. with distribution* $\mathcal{N}(0, \sigma^2 = 0.5)$.

The examples show that ICP loses its coverage guarantee if we assume linear relationships for testing (4.3) while the true data generating process is nonlinear.

In the general nonlinear and nonparametric case, however, it becomes more difficult to guarantee the type I error rate when testing the conditional independence (4.3) (Shah and Peters, 2018). This in contrast to nonparametric tests for (unconditional) independence (Bergsma and Dassios, 2014; Székely et al., 2007). In a nonlinear conditional independence test setting, where we know an appropriate parametric basis expansion for the causal effect of the variables we condition on, we can of course revert back to unconditional independence testing. Apart from such special circumstances, we have to find tests that guarantee the type I error rate in (4.5) as closely as possible under a wide range of scenarios. We describe some methods that test (4.3) in §4.3 but for now let us assume that we are given such a test. We can then apply the method of nonlinear ICP (4.4) to the example of fertility data.

Example 4.6 (Fertility data) *The following sets were accepted at the level* $\alpha = 0.1$ *when using nonlinear ICP with invariant conditional quantile prediction (see Appendix 4.B for details) as a conditional independence test:*

$$\begin{aligned} S_1 &= \{\textit{Q5}\} \\ S_2 &= \{\textit{IMR, Imports of goods and services, Urban pop. (\% of total)}\} \\ S_3 &= \{\textit{IMR, Education expend. (\% of GNI), Exports of goods and services,} \\ &\qquad \textit{GDP per capita}\} \end{aligned}$$

As the intersection of $S_1, \ldots, S_3$ *is empty, we have* $\hat{S} = \emptyset$. *This motivates the concept of defining sets.*

---

[4] This example is motivated by theory that combines linear and nonlinear models with additive noise (Rothenhäusler et al., 2018b).

4.2.2. Defining sets

It is often impossible to distinguish between highly correlated variables. For example, infant mortality IMR and under-five mortality Q5 are highly correlated in the data and can often be substituted for each other. We accept sets that contain either of these variables. When taking the intersection as in (4.4), this leads to exclusion of both variables in $\hat{S}$ and potentially to an altogether empty set $\hat{S}$. We can instead ask for the defining sets (Goeman and Solari, 2011), where a defining set $\hat{D} \subseteq \{1, \ldots, p\}$ has the properties

(i) $S \cap \hat{D} \neq \emptyset$ for all S such that $H_{0,S}$ is accepted.

(ii) there exists no strictly smaller set D' with $D' \subset \hat{D}$ for which property (i) is true.

In words, we are looking for subsets $\hat{D}$, such that each accepted set S has at least one element that also appears in $\hat{D}$. If the intersection $\hat{S}$ (4.4) is non-empty, any subset of $\hat{S}$ that contains only one variable is a defining set. Defining set are especially useful, however, in cases where the intersection $\hat{S}$ *is* empty. We still know that, with high probability, at least one of the variables in the defining set $\hat{D}$ has to be a parent. Defining sets are not necessarily unique. Given a defining set $\hat{D}$, we thus know that

$$P(S^* \cap \hat{D} = \emptyset) \leq P(H_{0,S^*} \text{ rejected}) \leq \alpha.$$

That is, a) at least one of the variables in the defining set $\hat{D}$ is a parent of the target, and b) the data do not allow to resolve it on a finer scale.

Example 4.7 (Fertility data) *We obtain seven defining sets:*

$\hat{D}_1 = \{$*IMR, Q5*$\}$

$\hat{D}_2 = \{$*Q5, Education expenditure (% of GNI), Imports of goods and services*$\}$

$\hat{D}_3 = \{$*Q5, Education expenditure (% of GNI), Urban pop. (% of total)*$\}$

$\hat{D}_4 = \{$*Q5, Exports of goods and services, Imports of goods and services*$\}$

$\hat{D}_5 = \{$*Q5, Exports of goods and services, Urban pop. (% of total)*$\}$

$\hat{D}_6 = \{$*Q5, GDP per capita, Imports of goods and services*$\}$

$\hat{D}_7 = \{$*Q5, GDP per capita, Urban pop. (% of total)*$\}$

Thus the highly-correlated variables infant mortality IMR and under-five mortality Q5 indeed form one of the defining sets in this example in the *sense that we know at least one of the two is a causal parent for fertility but we cannot resolve which one it is or whether both of them are parents.*

4.2.3. Confidence bands

For a given set S, we can in general construct a $(1-\alpha)$-confidence band $\hat{\mathcal{F}}_S$ for the regression function when predicting Y with the variables X_S. Note that if f is the regression function when regressing Y on the true set of causal variables X_{S^*} and hence, then, with probability $1-\alpha$, we have

$$P(f \in \hat{\mathcal{F}}_{S^*}) \geq 1-\alpha.$$

Furthermore, from §4.2.1 we know that H_{0,S^*} is accepted with probability $1-\alpha$. We can hence construct a confidence band for the causal effects as

$$\hat{\mathcal{F}} := \bigcup_{S: H_{0,S} \text{ not rejected}} \hat{\mathcal{F}}_S. \tag{4.6}$$

Using a Bonferroni correction, we have the guarantee that

$$P\big(f \in \hat{\mathcal{F}}\big) \;\geq\; 1-2\alpha,$$

where the coverage guarantee is point-wise or uniform, depending on the coverage guarantee of the underlying estimators $\hat{\mathcal{F}}_S$ for all given $S \subseteq \{1,\ldots,p\}$.

4.2.4. Average causal effects

The confidence bands $\hat{\mathcal{F}}$ themselves can be difficult to interpret. Interpretability can be guided by looking at the average causal effect in the sense that we compare the expected response at $\tilde{x}$ and x:

$$ACE(\tilde{x}, x) \;:=\; E\big(Y \big| \mathrm{do}(X=\tilde{x})\big) - E\big(Y \big| \mathrm{do}(X=x)\big). \tag{4.7}$$

For the fertility data, this would involve a hypothetical scenario where we fix the variables to be equal to x for a country in the second term and, for the first term, we set the variables to $\tilde{x}$, which might differ from x just in one or a few coordinates. Eq. (4.7) then compares the average expected fertility between these two scenarios. Note that the expected response under a do-operation is just a function of the causal variables $S^* \subseteq \{1, \ldots, p\}$. That is—in the absence of hidden variables—we have

$$E\big(Y\big|\mathrm{do}(X = x)\big) \;=\; E\big(Y\big|\mathrm{do}(X_{S^*} = x_{S^*})\big),$$

and the latter is then equal to

$$E\big(Y\big|\mathrm{do}(X_{S^*} = x_{S^*})\big) \;=\; E\big(Y\big|X_{S^*} = x_{S^*}\big),$$

that is it does not matter whether we set the causal variables to a specific value x_{S^*} or whether they were observed in this state.

Once we have a confidence band as defined in (4.6), we can bound the average causal effect (4.7) by the interval

$$\widehat{ACE}(\tilde{x}, x) := \big[\inf_{g \in \hat{\mathcal{F}}} (g(\tilde{x}) - g(x)),\; \sup_{g \in \hat{\mathcal{F}}} (g(\tilde{x}) - g(x))\big],$$

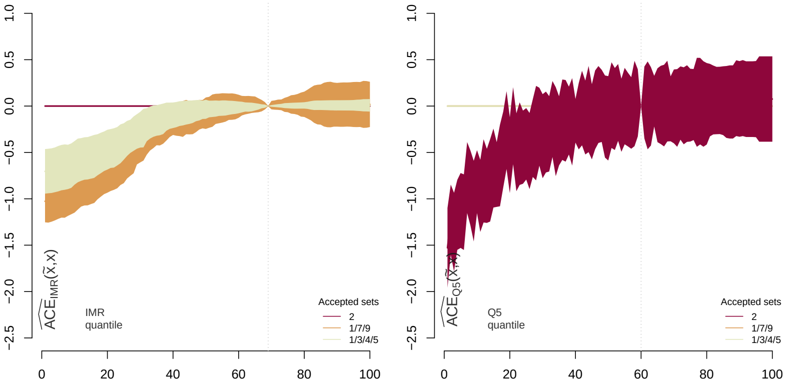
with the immediate guarantee that

$$P\big(ACE(\tilde{x}, x) \;\in\; \widehat{ACE}(\tilde{x}, x)\big) \geq 1 - 2\alpha, \tag{4.8}$$

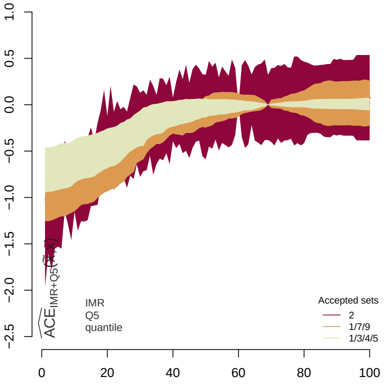
where the factor 2α is guarding, by a Bonferroni correction, against both a probability α that S^* will not be accepted—and hence $\hat{S} \subseteq S^*$ is not necessarily true—and another probability α that the confidence bands will not provide coverage for the parental set S^*.

Example (Fertility data). The confidence bands $\hat{\mathcal{F}}$, required for the computation of $\widehat{ACE}(\tilde{x}, x)$, are obtained by a time series bootstrap (Künsch, 1989) as the fertility data contain temporal dependencies. The time series bootstrap procedure is described in Appendix 4.A. We use a level of $\alpha = 0.1$ which implies a coverage guarantee of 80% as per (4.8). In the examples below, we set x to an observed data point and vary only $\tilde{x}$.

In the first example, we consider the observed covariates for Nigeria in 1993 as x. The point of comparison $\tilde{x}$ is set equal to x, except for the variables in the defining set $\hat{D}_1 = \{\text{IMR}, \text{Q5}\}$. In Figures 4.2(a) and (b), these are varied individually over their respective quantiles. The overall confidence interval $\hat{\mathcal{F}}$ consists of the union of the shown confidence intervals $\hat{\mathcal{F}}_S$. If $x = \tilde{x}$ (shown by the vertical lines), the average causal effect is zero, of course. In neither of the two scenarios shown in Figures 4.2(a) and (b), we observe consistent effects different from zero as some of the accepted



(a) $\widehat{ACE}_{\text{IMR}}(\tilde{x}, x)$ (b) $\widehat{ACE}_{\text{Q5}}(\tilde{x}, x)$



(c) $\widehat{ACE}_{\text{IMR+Q5}}(\tilde{x}, x)$

Figure 4.2.: Data for Nigeria in 1993: The union of the confidence bands $\hat{\mathcal{F}}_S$, denoted by $\hat{\mathcal{F}}$, bounds the average causal effect of varying the variables in the defining set $\hat{D}_1 = \{\text{IMR, Q5}\}$ on the target log(TFR). IMR and Q5 have been varied individually, see panels (a) and (b), as well as jointly, see panel (c), over their respective quantiles. In panels (a) and (b), we do not observe consistent effects different from zero as some of the accepted models do not contain IMR and some do not contain Q5. However, when varying the variables $\hat{D}_1 = \{\text{IMR, Q5}\}$ jointly (see panel (c)), we see that all accepted models predict an increase in expected log(TFR) as IMR and Q5 increase.

models do not contain IMR and some do not contain Q5. However, when varying the variables $\hat{D}_1 = \{\text{IMR, Q5}\}$ jointly (see Figure 4.2(c)), we see that all accepted models predict an increase in expected log(TFR) as IMR and Q5 increase.

In the second example, we compare the expected fertility rate between countries where all covariates are set to the value x, which is here chosen to be equal to the observed values of all African countries in 2013. The expected value of log-fertility under this value x of covariates is compared to the scenario where we take as $\tilde{x}$ the same value but set the values of the child-mortality variables IMR and Q5 to their respective European averages. The union of intervals in Figure 4.3 (depicted by the horizontal line segments) correspond to $\widehat{ACE}(\tilde{x}, x)$ for each country under nonlinear ICP with invariant conditional quantile prediction. The accepted models make largely coherent predictions for the effect associated with this comparison. For most countries, the difference is negative, meaning that the average expected fertility declines if the child mortality rate in a country decreases to European levels. The countries where $\widehat{ACE}_{\text{IMR+Q5}}(\tilde{x}, x)$ contains 0 typically have a child mortality rate that is close to European levels, meaning that there is no substantial difference between the two points $\tilde{x}, x$ of comparison.

For comparison, in Figure 4.4, we show the equivalent computation as in Figure 4.3 when *all* covariates are *assumed* to have a direct causal effect on the target and a Random Forest is used for estimation (Breiman, 2001). We observe that while the resulting regression bootstrap confidence intervals often overlap with $\widehat{ACE}_{\text{IMR+Q5}}(\tilde{x}, x)$, they are typically much smaller. This implies that if the regression model containing all covariates was—wrongly—used as a surrogate for the causal model, the uncertainty of the prediction would be underestimated. Furthermore, such an approach ignoring the causal structure can lead to a significant bias in the prediction of causal effects when we consider interventions on descendants of the target variable, for example.

Lastly, we consider a cross validation scheme over time to assess the coverage properties of nonlinear ICP. We leave out the data corresponding to one entire continent and run nonlinear ICP with invariant conditional quantile prediction using the data from the remaining five continents. We perform this leave-one-continent-out scheme for different values of α. For each value of α, we then compute the predicted change in the response $\log(\text{TFR})$ from $1973 - 2008$ for each country belonging to the continent

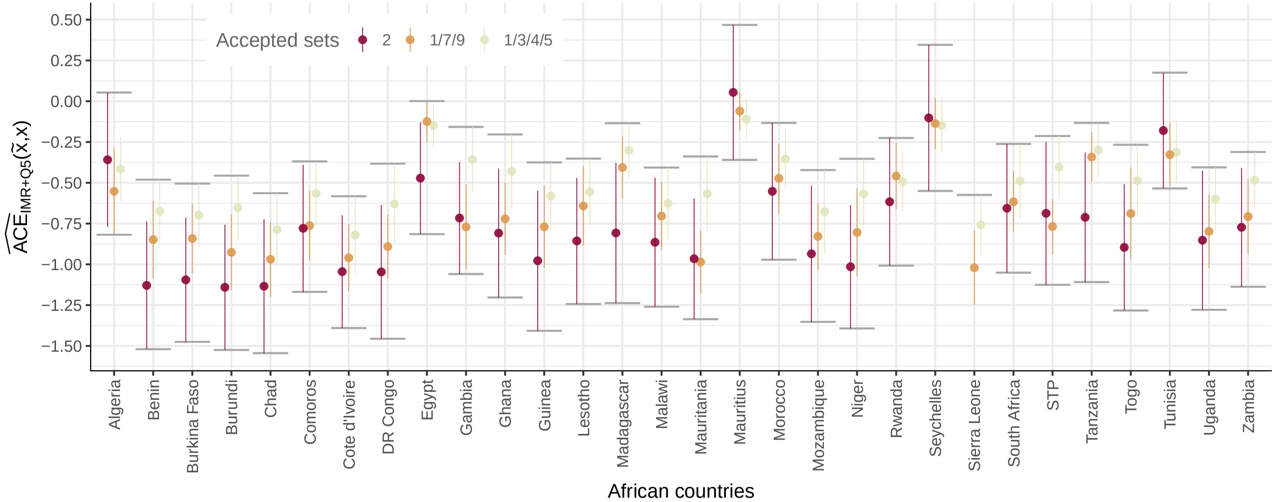


Figure 4.3.: Bounds for the average causal effect of setting the variables IMR and Q5 in the African countries in 2013 to European levels, that is $\tilde{x}$ differs from the country-specific observed values x in that the child mortality rates IMR and Q5 have been set to their respective European average. $\widehat{ACE}_{\mathrm{IMR+Q5}}(\tilde{x}, x)$ is estimated using nonlinear ICP with invariant conditional quantile prediction. The implied coverage guarantee is 80% as we chose $\alpha = 0.1$.

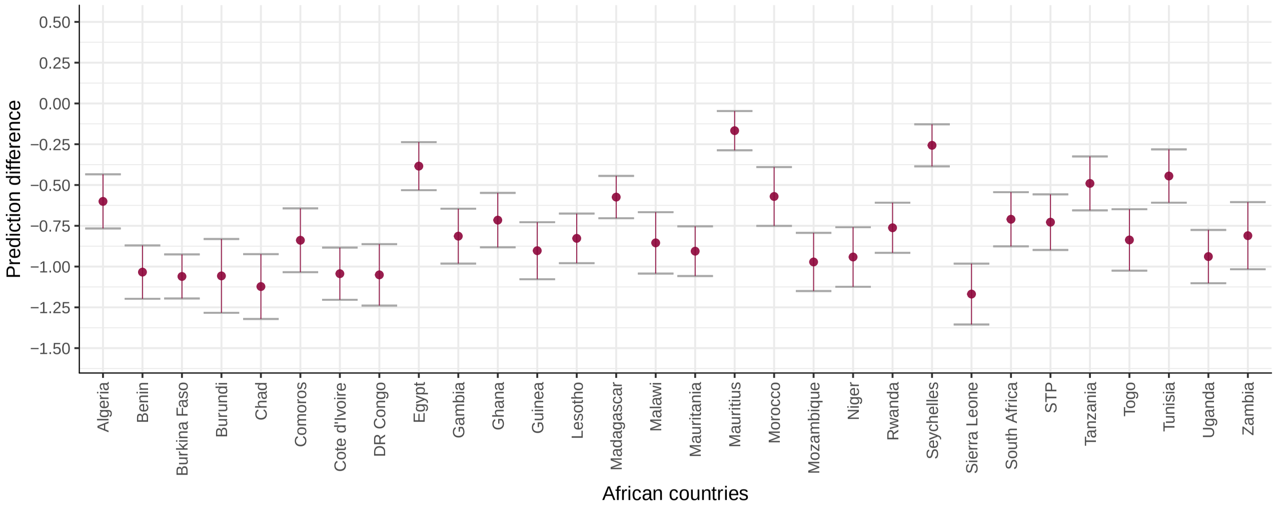


Figure 4.4.: Random Forest regression model using all covariates as input with bootstrap confidence intervals. The (non-causal) regression effect coverage is again set to 80%. We will argue below that the confidence intervals obtained by random forest are too small, see Table 4.1 and Figures 4.6—4.8.

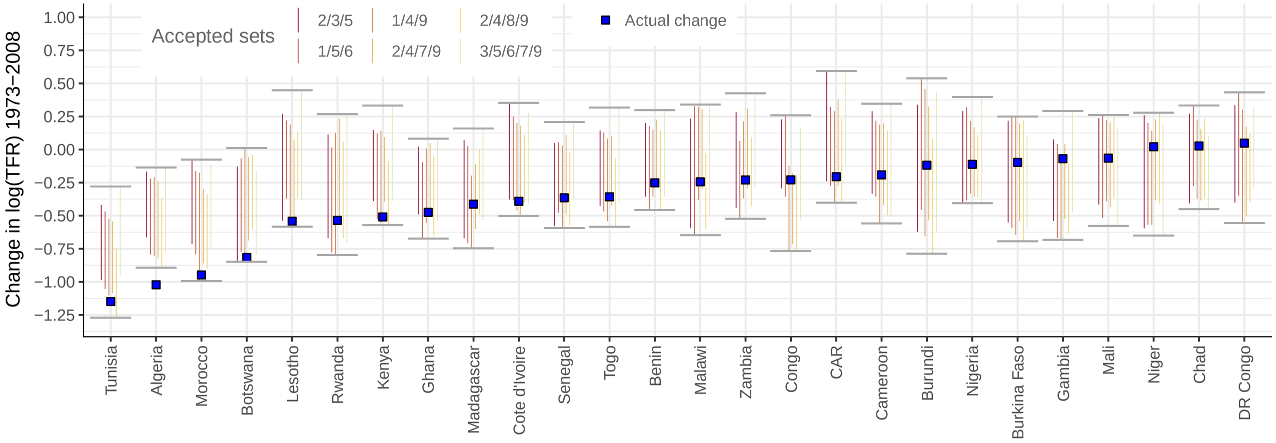


Figure 4.5.: The confidence intervals show the predicted change in log (TFR) from 1973 – 2008 for all African countries when not using their data in the nonlinear ICP estimation (using invariant conditional quantile prediction with $\alpha = 0.1$). In other words, only the data from the remaining five continents was used during training. The horizontal line segments mark the union over the accepted models' intervals for the predicted change; the blue squares show the true change in log (TFR) from 1973 – 2008. Only those countries are displayed for which the response was not missing in the data, i.e. where log (TFR) in 1973 and in 2008 were recorded. The coverage is $25/26 \approx 0.96$.

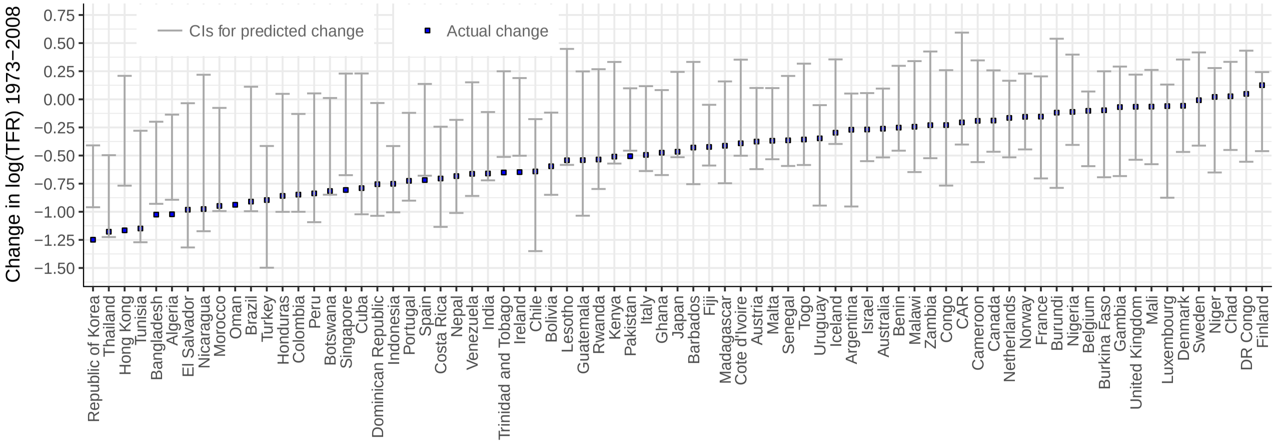


Figure 4.6.: Nonlinear ICP with invariant conditional quantile prediction. Coverage: 0.88. The confidence intervals show the predicted change in log (TFR) from 1973–2008 for all countries when not using the data of the country's continent in the estimation (with implied coverage guarantee of 80%). Only those countries are displayed for which log (TFR) in 1973 and 2008 were not missing in the data. The shown intervals are the union over the accepted models' intervals.

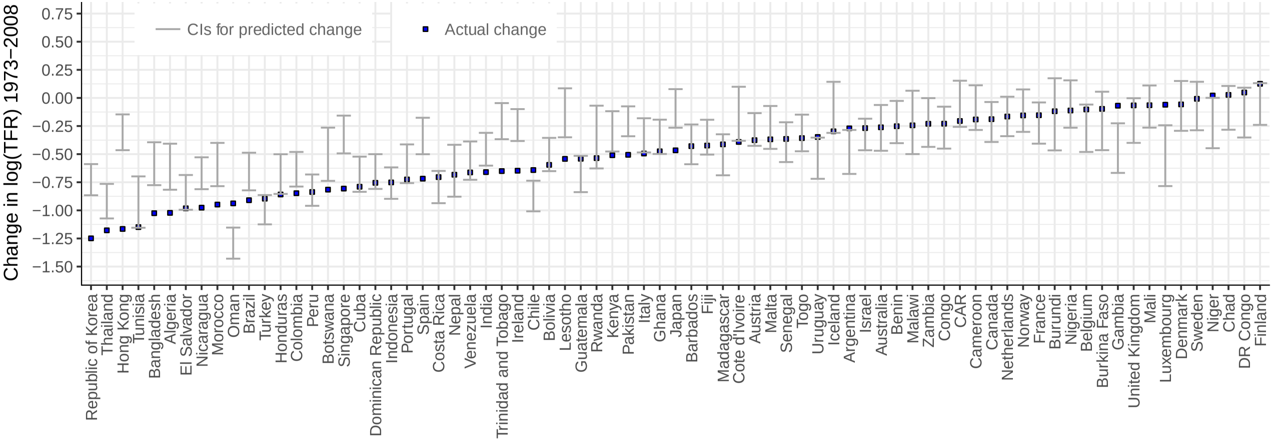


Figure 4.7.: Random Forest regression model with bootstrap confidence intervals. Coverage: 0.61. The confidence intervals show the predicted change in log (TFR) from 1973–2008 for all countries when not using the data of the country's continent in the estimation (with implied coverage guarantee of 80%). Only those countries are displayed for which log (TFR) in 1973 and 2008 were not missing in the data.

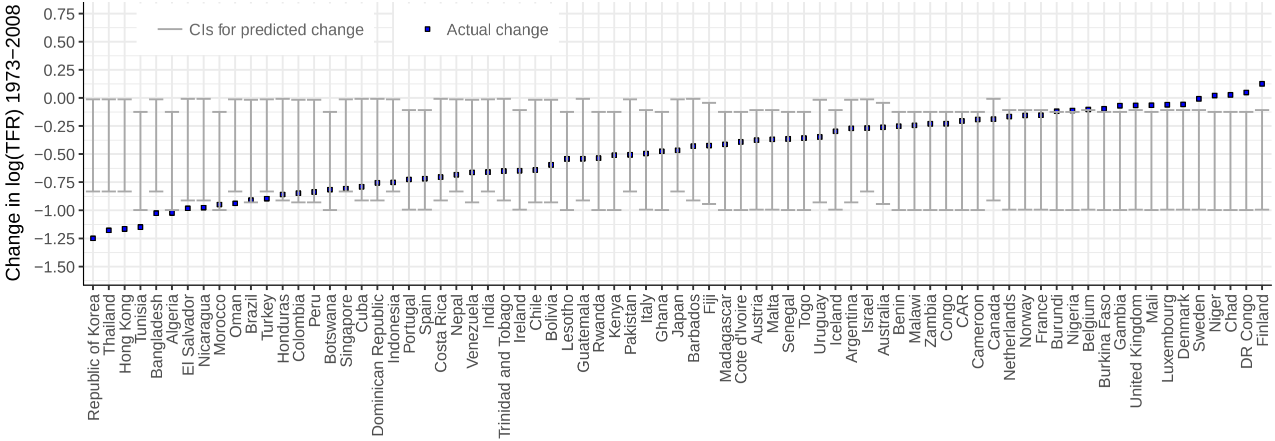


Figure 4.8.: Prediction based on mean change on continents other than country's own continent. Coverage: 0.68. The confidence intervals show the predicted change in $\log(\text{TFR})$ from 1973–2008 for all countries when not using the data of the country's continent in the estimation (with implied coverage guarantee of 80%). Only those countries are displayed for which $\log(\text{TFR})$ in 1973 and 2008 were not missing in the data.

Table 4.1.: Coverage

Coverage guarantee	0.95	0.90	0.8	0.5
Coverage with nonlinear ICP	0.99	0.95	0.88	0.58
Coverage with Random Forest	0.76	0.71	0.61	0.32
Coverage with mean change	0.95	0.88	0.68	0.36

that was left out during the estimation procedure. The predictions are obtained by using the respective accepted models.[5] We then compare the union of the associated confidence intervals with the real, observed change in $\log(\text{TFR})$. This allows us to compute the coverage statistics shown in Table 4.1. We observe that nonlinear ICP typically achieves more accurate coverage compared to (i) a Random Forest regression model including all variables and (ii) a baseline where the predicted change in $\log(\text{TFR})$ for a country is the observed mean change in $\log(\text{TFR})$ across all continents other than the country's own continent. Figures 4.5—4.8 show the confidence intervals and the observed values for all African countries (Figure 4.5) and all countries (Figures 4.6—4.8) with observed $\log(\text{TFR})$ in 1973 and 2008.

Recall that one advantage of a causal model is that, in the absence of hidden variables, it does not matter whether certain variables have been intervened on or whether they were observed in this state – the resulting prediction remains correct in any of these cases. On the contrary, the predictions of a non-causal model can become drastically incorrect under interventions. This may be one reason for the less accurate coverage statistics of the Random Forest regression model—in this example, it seems plausible that some of the predictors were subject to different external ‘interventions’ across continents and countries.

---

[5] Their number differs according to α: for a smaller α, additional models can be accepted compared to using a larger value of α. In other words, the accepted models for α_2 where $\alpha_1 < \alpha_2$ are a subset of the accepted models for α_1.

4.3. Conditional independence tests

We present and evaluate an array of methods for testing conditional independence in a nonlinear setting, many of which exploit the invariance of causal models across different environments. Here, we briefly sketch the main ideas of the considered tests, their respective assumptions and further details are provided in Appendix 4.B. All methods (A) – (F) are available in the package `CondIndTests` for the R language. Table 4.2 in Appendix 4.B.7 shows the supported methods and options. An experimental comparison of the corresponding power and type I error rates of these tests can be found in §4.4.

(A) **Kernel conditional independence test.** Use a kernel conditional independence test for $Y \perp\!\!\!\perp E \mid X_S$ (Fukumizu et al., 2008; Zhang et al., 2011). See Appendix 4.B.1 for further details.

(B) **Residual prediction test.** Perform a nonlinear regression from Y on X_S, using an appropriate basis expansion, and apply a variant of a Residual Prediction (RP) test (Shah and Bühlmann, 2018). The main idea is to scale the residuals of the regression such that the resulting test statistic is not a function of the unknown noise variance. This allows for a straight-forward test for dependence between the residuals and (E, X_S). In cases where a suitable basis expansion is unknown, random features (Rahimi and Recht, 2007; Williams and Seeger, 2001) can be used as an approximation. See Appendix 4.B.2 for further details.

(C) **Invariant environment prediction.** Predict the environment E, once with a model that uses X_S as predictors only and once with a model that uses (X_S, Y) as predictors. If the null is true and we find the optimal model in both cases, then the out-of-sample performance of both models is statistically indistinguishable. See Appendix 4.B.3 for further details.

(D) **Invariant target prediction.** Predict the target Y, once with a model that uses X_S as predictors only and once with a model that uses (X_S, E) as predictors. If the null is true and we find the optimal model in both cases, then the out-of-sample performance of both models is statistically indistinguishable. See Appendix 4.B.4 for further details.

(E) **Invariant residual distribution test.** Pool the data across all environments and predict the response Y with variables X_S. Then test whether the distribution of the residuals is identical in all environments E. See Appendix 4.B.5 for further details.

(F) **Invariant conditional quantile prediction.** Predict a $1-\beta$ quantile of the conditional distribution of Y, given X_S, by pooling the data over all environments. Then test whether the exceedance of the conditional quantiles is independent of the environment variable. Repeat for a number of quantiles and aggregate the resulting individual p-values by Bonferroni correction. See Appendix 4.B.6 for further details.

Another interesting possibility for future work would be to devise a conditional independence test based on model-based recursive partitioning (Hothorn and Zeileis, 2015; Zeileis et al., 2008).

Non-trivial, assumption-free conditional independence tests with a valid level do not exist (Shah and Peters, 2018). It is therefore not surprising that all of the above tests assume the dependence on the conditioning variable to be “simple” in one form or the other. Some of the above tests require the noise variable in (4.1) to be additive in the sense that we do not expect the respective test to have the correct level when the noise is not additive. As additive noise is also used in §4.2.3 and §4.2.4, we have written the structural equations above in an additive form.

One of the inherent difficulties with these tests is that the estimation bias when conditioning on potential parents in (4.3) can potentially lead to a more frequent rejection of a true null hypothesis than the nominal level suggests. In approaches (C) and (D), we also need to test whether the predictive accuracy is identical under both models and in approaches (E) and (F) we need to test whether univariate distributions remain invariant across environments. While these additional tests are relatively straightforward, a choice has to be made.

Discussion of power. Conditional independence testing is a statistically challenging problem. For the setting where we condition on a continuous random variable, we are not aware of any conditional independence test that holds the correct level and still has (asymptotic) power against a wide range of alternatives. Here, we want to briefly mention some power properties of the tests we have discussed above.

Invariant target prediction (D), for example, has no power to detect if the noise variance is a function of E, as shown by the following example

Example 4.8 *Assume that the distribution is entailed by the following model*

$$
\begin{aligned}
E &\leftarrow 0.2\eta_E \\
X &\leftarrow \eta_X \\
Y &\leftarrow X^2 + E \cdot \eta_Y,
\end{aligned}
$$

where $\eta_E, \eta_X, \eta_Y \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$. *Then, any regression from* Y *on* X *and* E *yields the same results as regressing* Y *on* X *only. That is,*

$$\textit{for all } x, e: \ \mathbf{E}[Y \mid X = x] = \mathbf{E}[Y \mid X = x, E = e]$$

although

$$Y \not\perp\!\!\!\perp E \mid X.$$

The invariant residual distribution test (E), in contrast, assumes homoscedasticity and might have wrong coverage if this assumption is violated. Furthermore, two different linear models do not necessarily yield different distributions of the residuals when performing a regression on the pooled data set.

Example 4.9 *Consider the following data generating process*

$$
\begin{aligned}
Y^{e=1} &\leftarrow 2X^{e=1} + N^{e=1} \\
Y^{e=2} &\leftarrow -X^{e=2} + 0.3N^{e=2},
\end{aligned}
$$

where the input variables $X^{e=1}$ *and* $X^{e=2}$ *and the noise variables* $N^{e=1}$ *and* $N^{e=2}$ *have the same distribution in each environment, respectively. Then, approach (E) will accept the null hypothesis of invariant prediction.*

It is possible to reject the null hypothesis of invariant prediction in Example 4.9 by testing whether in each environment the residuals are uncorrelated from the input.

Invariant conditional quantile prediction (F) assumes neither homoscedasticity nor does it suffer from the same issue of (D), i.e. no power against an alternative where the noise variance σ is a function of E. However, it is possible to construct examples where (F) will have no power if the noise variance is a function of both E *and* the causal variables X_{S^*}. Even then, though, the noise level would have to be carefully balanced to reduce the power to 0 with approach (F) as the exceedance probabilities of various quantiles (a function of X_{S^*}) would have to remain constant if we condition on various values of E.

4.4. Simulation study

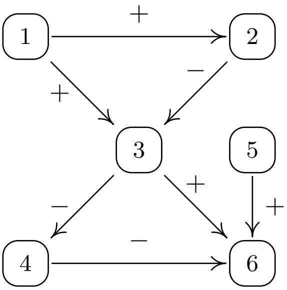


Figure 4.9.: The structure of the causal graph used in the simulations. The causal order is unknown for the simulations. All edge weights are 1 in absolute value.

For the simulations, we generate data from different nonlinear additive noise causal models and compare the performance of the proposed conditional independence tests. The structural equations are of the form $Z_k \leftarrow g_k(Z_{\mathrm{pa}_k}) + \eta_k$, where the structure of the DAG is shown in Figure 4.9 and kept constant throughout the simulations for ease of comparison. We vary the nonlinearities used, the target, the type and strength of interventions, the noise tail behavior and whether parental contributions are multiplicative or additive. The simulation settings are described in Appendix 4.C in detail.

We apply all the conditional independence tests (CITs) that we have introduced in §4.3, implemented with the following methods and tests as subroutines:

CIT	Implementation
(A)	KCI without Gaussian process estimation
(B)(i)	RP w/ Fourier random features
(B)(ii)	RP w/ Nyström random features and RBF kernel
(B)(iii)	RP w/ Nyström random features and polynomial kernel (random degree)
(B)(iv)	RP w/ provided polynomial basis (random degree)
(C)	Random forest and χ^2-test
(D)(i)	GAM with F-Test
(D)(ii)	GAM with Wilcoxon test
(D)(iii)	Random forest with F-Test
(D)(iv)	Random forest with Wilcoxon test
(E)(i)	GAM with Kolmogorov-Smirnov test
(E)(ii)	GAM with Levene's test + Wilcoxon test
(E)(iii)	Random forest with Kolmogorov-Smirnov test
(E)(iv)	Random forest with Levene's test + Wilcoxon test
(F)	Quantile regression forest with Fisher's exact test

As a disclaimer we have to note that KCI is implemented without Gaussian process estimation. The KCI results shown below might improve if the latter is added to the algorithm.

Baselines. We compare against a number of baselines. Importantly, most of these methods contain various model misspecifications when applied in the considered problem setting. Therefore, they would not be suitable in practice. However, it is instructive to study the effect of the model misspecifications on performance.

1. The method of Causal Additive Models (CAM) (Bühlmann et al., 2014) identifies graph structure based on nonlinear additive noise models (Peters and Bühlmann, 2014). Here, we apply the method in the following way. We run CAM separately in each environment and output the intersection of the causal parents that were retrieved in each environment. Note that the method's underlying assumption of Gaussian noise is violated.

2. We run the PC algorithm (Spirtes et al., 2000) in two different variants. We consider a variable to be the parent of the target if a *directed* edge between them is retrieved; we discard undirected edges. In the first variant of PC we consider, the environment variable is part of the input; conditional independence testing within the PC

algorithm is performed with KCI, for unconditional independence testing we use HSIC (Gretton et al., 2008, 2005), using the implementation from Pfister and Peters, 2017 (denoted with ‘PC(i)’ in the figures). In the second variant, we run the PC algorithm on the pooled data (ignoring the environment information), testing for zero partial correlations (denoted with ‘PC(ii)’ in the figures). Here, the model misspecification is the assumed linearity of the structural equations.

3. We compare against linear ICP (Peters and Bühlmann, 2014) where the model misspecification is the assumed linearity of the structural equations.
4. We compare against LiNGAM (Shimizu et al., 2006), run on the pooled data without taking the environment information into account. Here, the model misspecifications are the assumed linearity of the structural equations and the i.i.d. assumption which does not hold.
5. We also show the outcome of a random selection of the parents that adheres to the FWER-limit by selecting the empty set ($\hat{S} = \emptyset$) with probability $1 - \alpha$ and setting $\hat{S} = \{k\}$ for k randomly and uniformly picked from $\{1, \dots, p\} \setminus k'$ with probability α, where k' is the index of the current target variable. The random selection is guaranteed to maintain FWER at or below $1 - \alpha$.

Thus, all considered baseline models in 1. – 4. —except for ‘PC(i)‘—contain at least slight model misspecifications.

Metrics. Error rates and power are measured in the following by

(i) Type I errors are measured by the **family-wise error rate** (FWER), the probability of making one or more erroneous selections

$$P\big(\hat{S} \not\subseteq S^*\big).$$

(ii) Power is measured by the **Jaccard similarity**, the ratio between the size of the intersection and the size of the union of the estimated set $\hat{S}$ and the true set S^*. It is defined as 1 if both $S^* = \hat{S} = \emptyset$ and otherwise as

$$\frac{|\hat{S} \cap S^*|}{|\hat{S} \cup S^*|}.$$

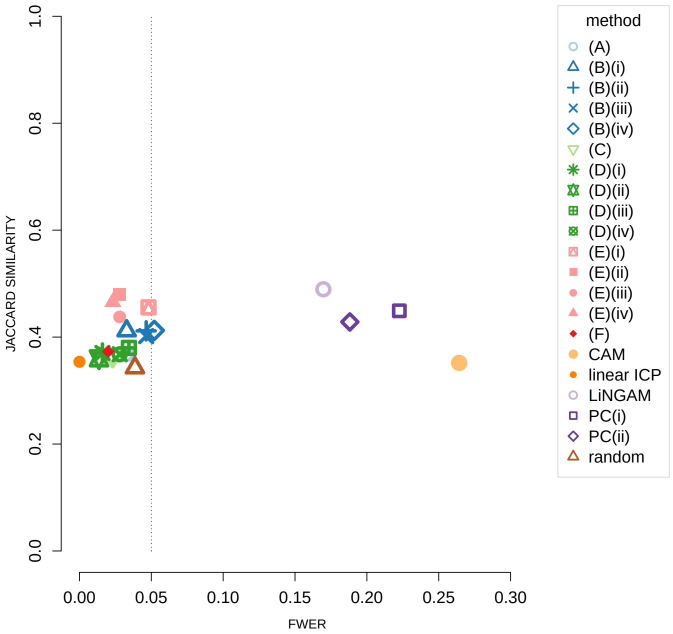


Figure 4.10.: Average Jaccard similarity (y-axis) against average FWER (x-axis), stratified according to which conditional independence test (A) – (F) or baseline method has been used. The nominal level is $\alpha = 0.05$, illustrated by the vertical dotted line. The shown results are averaged over all target variables. Since the empty set is the correct solution for target variable 1 and 5, methods that mostly return the empty set (such as random or linear ICP) perform still quite well in terms of average Jaccard similarity. Since all variables are highly predictive for the target variable Y, see Figure 4.9, classical variable selection techniques as LASSO have a FWER that lies far beyond α. Importantly, the considered baselines are not suitable for the considered problem setting due to various model misspecifications. We show their performance for comparison to illustrate the influence of these misspecifications.

The Jaccard similarity is thus between 0 and 1 and the optimal value 1 is attained if and only if $\hat{S} = S^*$.

Type-I-error rate of conditional independence tests. Figure 4.10 shows the average FWER on the x-axis (and the average Jaccard similarity on

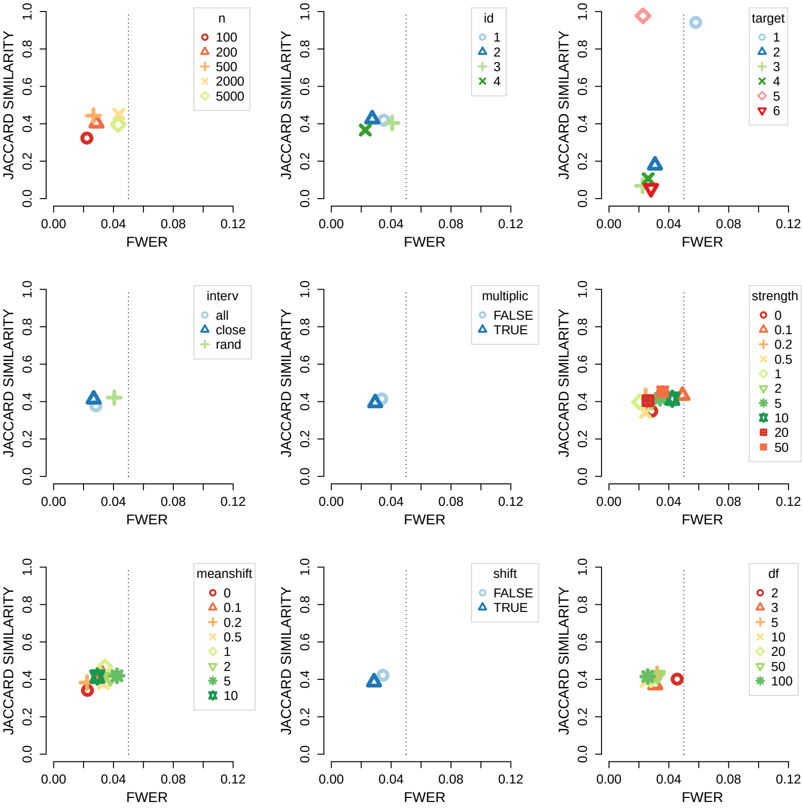
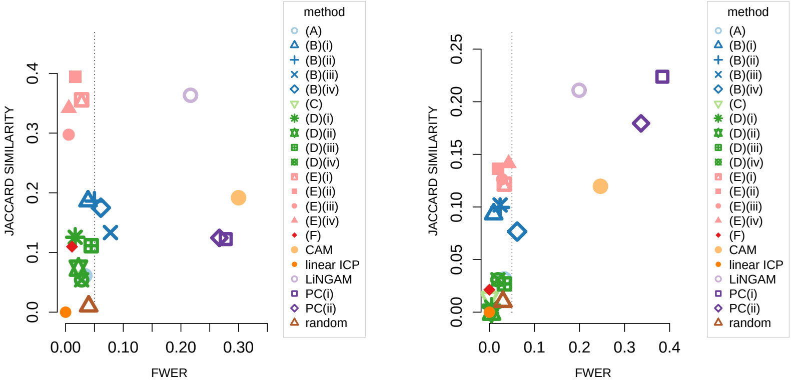
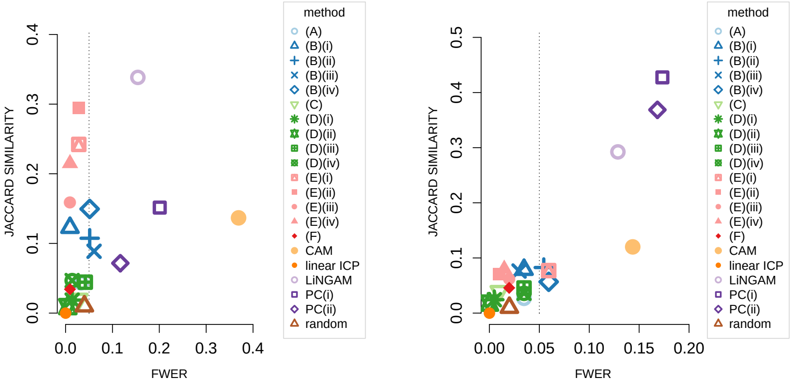


Figure 4.11.: Average Jaccard similarity over the conditional independence tests (A) – (F) (y-axis) against average FWER (x-axis), stratified according to various parameters (from top left to bottom right): sample size ‘n’, type of nonlinearity ‘id’, ‘target variable’, intervention location ‘interv’, multiplicative effects indicator ‘multiplic’, ‘strength’ of interventions, mean value of interventions ‘meanshift’, shift intervention indicator ‘shift’ and degrees of freedom for t-distributed noise ‘df’. For details, see the description in Appendix 4.C. The FWER is within the nominal level in general for all conditional independence tests. The average Jaccard similarity is mostly determined by the target variable under consideration, see top right panel.

the y-axis) for all methods. The FWER is close but below the nominal FWER rate of $\alpha = 0.05$ for all conditional independence tests, that is $P(\hat{S} \subseteq S^*) \geq 1 - \alpha$. The same holds for the baselines linear ICP and



(a) Target variable 2 (b) Target variable 3



(c) Target variable 4 (d) Target variable 6

Figure 4.12.: The identical plot to Figure 4.10 separately for target variables 2, 3, 4 and 6. For all target variables, method (E)(ii)—an invariant residual distribution test using GAM with Levene's test + Wilcoxon test—performs constantly as good or nearly as good as the optimal method among the considered tests.

random selection. Notably, the average Jaccard similarity of the random selection baseline is on average not much lower than for the other methods. The reason is mostly a large variation in average Jaccard similarity across the different target variables, as discussed further below and as will be evident from Figure 4.11 (top right plot). In fact, as can be seen from Figure 4.12, random guessing is much worse than the optimal methods on each target variable. The FWER of the remaining baselines CAM, LiNGAM, PC(i) and PC(ii) lies well above α.

A caveat of the FWER control seen in Figure 4.10 is that while the FWER is maintained at the desired level, the test H_{0,S^*} might be rejected more often than with probability α. The error control rests on the fact that H_{0,S^*} is accepted with probability higher than $1-\alpha$ (since the null is true for S^*). However, if a mistake is made and H_{0,S^*} is falsely rejected, then we might still have $\hat{S} \subseteq S^*$ because either all other sets are rejected, too, in which case $\hat{S} = \emptyset$, or because other sets (such as the empty set) are accepted and the intersection of all accepted sets is—by accident—again a subset of S^*. In other words: some mistakes might cancel each other out but overall the FWER is very close to the nominal level, even if we stratify according to sample size, target, type of nonlinearity and other parameters, as can be seen from Figure 4.11.

Power. Figures 4.10 shows on the y-axis the average Jaccard similarity for all methods. The optimal value is 1 and is attained if and only if $\hat{S} = S^*$. A value 0 corresponds to disjoint sets $\hat{S}$ and S^*. The average Jaccard similarity is around 0.4 for most methods and not clearly dependent on the type I errors of the methods. Figure 4.11 shows the average FWER and Jaccard similarities stratified according to various parameters.

One of the most important determinants of success (or the most important) is the target, that is the variable for which we would like to infer the causal parents; see top right panel in Figure 4.11. Variables 1 and 5 as targets have a relatively high average Jaccard similarity when trying to recover the parental set. These two variables have an empty parental set ($S^* = \emptyset$) and the average Jaccard similarity thus always exceeds $1-\alpha$ if the level of the procedure is maintained as then $\hat{S} = \emptyset = S^*$ with probability at least $1-\alpha$ and the Jaccard similarity is 1 if both $\hat{S}$ and S^* are empty. As testing for the true parental set corresponds to an unconditional independence test in this case, maintaining the level of the test procedure is much easier than for the other variables.

Figure 4.12 shows the same plot as Figure 4.10 for each of the more difficult target variables 2, 3, 4, and 6 separately. As can be seen from the graph in Figure 4.9 and the detailed description of the simulations in Appendix 4.C, the parents of target variable 3 are difficult to estimate as the paths $1 \to 2 \to 3$ and $1 \to 3$ cancel each other exactly in the linear setting (and approximately for nonlinear data), thus creating a non-faithful distribution. The cancellation of effects holds true if interventions occur on variable 1 and not on variable 2. A local violation of faithfulness leaves type I error rate control intact but can hurt power as many other sets besides the true S^* can get accepted, especially the empty set, thus yielding $\hat{S} = \emptyset$ when taking the intersection across all accepted sets to compute the estimate $\hat{S}$ in (4.4). Variable 4, on the other hand, has only a single parent, namely $S^* = \{3\}$, and the recovery of the single parent is much easier, with average Jaccard similarity up to a third. Variable 6 finally again has average Jaccard similarity of up to around a tenth only. It does not suffer from a local violation of faithfulness as variable 3 but the size of the parental set is now three, which again hurts the power of the procedure, as often already a subset of the true parents will be accepted and hence $\hat{S}$ in (4.4) will not be equal to S^* any longer but just a subset. For instance, when variable 5 is not intervened on in any environment it cannot be identified as a causal parent of variable 6, as it is then indistinguishable from the noise term. Similarly, in the linear setting, merely variable 3 can be identified as a parent of variable 6 if the interventions act on variables 1 and/or 2 only.

The baselines LiNGAM and PC show a larger Jaccard similarity for target variables 3, 4 (only LiNGAM), and 6 at the price of large FWER values.

In Appendix 4.D, Figures 4.13 – 4.16 show the equivalent to Figure 4.11, separately for target variables 2, 3, 4 and 6. For the sample size n, we observe that increasing it from 2000 to 5000 decreases power in case of target variables 4. This behavior can be explained by the fact that when testing S^* in Eq. (4.3), the null is rejected too often as the bias in the estimation performed as part of the conditional independence test yields deviations from the null that become significant with increasing sample size. For the nonlinearity, we find that the function $f_4(x) = \sin(2\pi x)$ is the most challenging one among the nonlinearities considered. It is associated with very low Jaccard similarity values for the target variables that do have parents. For the intervention type, it may seem surprising that ‘all’ does not yield the largest power. A possible explanation is that intervening on all variables except for the target yields more similar intervention settings—the intervention targets do not differ between environments 2 and 3, even though the strength of the interventions is different. So more heterogeneity between the intervention environments, i.e. also having different intervention targets, seems to improve performance in terms of Jaccard similarity. Lastly, we see that power is often higher for additive parental contributions than for multiplicative ones.

In summary, all tests (A) – (F) seem to maintain the desired type I error, chosen here as the family-wise error rate, while the power varies considerably. An invariant residual distribution test using GAM with Levene's test and Wilcoxon test produces results here that are constantly as good or nearly as good as the optimal methods for a range of different settings. However, it is only applicable for categorical environmental variables. For continuous environmental variables, the results suggest that the residual prediction test with random features might be a good choice.

4.5. Discussion and future work

Causal structure learning with the invariance principle was proposed Peters et al., 2016. However, the assumption of linear models in Peters et al., 2016 is unrealistic in many applications. In this work, we have shown how the framework can be extended to nonlinear and nonparametric models by using suitable nonlinear and nonparametric conditional independence tests. The properties of these conditional independence tests are critically important for the power of the resulting causal discovery procedure. We evaluated many different test empirically in the given context and highlighted approaches that seem to work robustly in different settings. In particular we find that fitting a nonlinear model with pooled data and then testing for differences between the residual distributions across environments results in desired coverage and high power if compared against a wide range of alternatives.

Our approach allowed us to model how several interventions may affect the total fertility rate of a country, using historical data about decline and rise of fertilities across different continents. In particular, we provided bounds on the average causal effect under certain (hypothetical) interventions such as a reduction in child mortality rates. We showed that the causal prediction intervals for hold-out data have better coverage than various baseline methods. The importance of infant mortality rate and under-five mortality rate on fertility rates is highlighted, reconfirming previous studies that have shown or hypothesized these factors to be important (Hirschman, 1994; Raftery et al., 1995). We stress that the results rely on causal sufficiency of the used variables, an assumption that can and should be debated for this particular example.

We also introduced the notion of ‘defining sets’ in the causal discovery context that helps in situations where the signal is weak or variables are highly correlated by returning sets of variables of which we know that at least one variable (but not necessarily all) in this set are causal for the target variable in question.

Finally, we provide software in the `R` (R Core Team, 2017) package `nonlinearICP`. A collection of the discussed conditional independence tests are part of the package `CondIndTests` and are hopefully of independent interest.

In applications where it is unclear whether the underlying models are linear or not, we suggest the following. While our proposed methods also hold the significance level if the underlying models are linear, we expect the linear version of ICP to have more power. Therefore, it is advisable to use the linear version of ICP if one has strong reasons to believe that the underlying model is indeed linear. In practice, one might first apply ICP with linear models and apply a nonlinear version if, for example, all linear models are rejected. One would then need to correct for multiple testing by a factor of 2.

Appendix 4.A Time series bootstrap procedure

In the time series bootstrap procedure used to obtain the confidence bands $\hat{\mathcal{F}}$ in §4.2.4, B bootstrap samples of the response Y are generated by first fitting the model on all data points. We then use the fitted values and residuals from this model. Each bootstrap sample is generated by resampling the residuals of this fit block- and country-wise. In more detail, we define the block-length l_b of residuals that should be sampled consecutively (we use $l_b = 3$) and we sample a number of time points $t_{s_1}, \ldots, t_{s_k}$ from which the residuals are resampled. For a country a and the first time point t_1, consider the fitted values at point t_1 and the fitted values for the $l_b - 1$ consecutive observations. We then sample a country b and add country b's residuals from time points $t_{s_1}, t_{s_1+1}, \ldots t_{s_1+l_b-1}$ to the fitted values of country a for the considered period $t_1, \ldots, t_{l_b}$. We then proceed with the next l_b consecutive fitted values for country a and add country b's residuals from observations $t_{s_2}, t_{s_2+1}, \ldots t_{s_2+l_b-1}$, until all fitted values of country a are covered. This procedure is applied to each country. Finally, to obtain the confidence intervals, we fit the model on each of the B bootstrap samples (Y^b, X), consisting of the response Y^b generated from the fitted values and the resampled residuals, and the observations X which have not been modified.

Appendix 4.B Conditional independence tests

For completeness, we first restate the generic method for Invariant Causal Prediction from Peters et al., 2016:

Algorithm 2 Generic method for Invariant Causal Prediction

```
Input: i.i.d. sample of $(Y, X, E)$, $\alpha$
for each $S \subseteq \{1, \ldots, p\}$ do
    Test whether $H_{0,S}$ holds at level $\alpha$.
end for
Set $\hat{S} := \bigcap_{S: H_{0,S} \text{ not rejected}} S$
Output: $\hat{S}$
```

The conditional independence tests discussed in this work can be used to perform the test in Step 2 of Algorithm 2. Therefore, the inputs to these tests consist of an i.i.d. sample of (Y, X_S, E) and α where X_S contains the variables corresponding to $S \subseteq \{1, \ldots, p\}$, i.e. the subset to be tested. Additionally, some test specific parameters might need to be specified. The return value of the tests is the respective test's decision about $H_{0,S}$.

For most tests, $E \in \mathbb{R}^d$ can be either discrete or continuous. As all empirical results in this work use an environment variable that is discrete and one-dimensional, the descriptions below focus on this setting. We then denote the index set of different environments with $\mathcal{E}$. We will comment on the required changes for the continuous and higher-dimensional case in the respective sections. Whenever applying the test for environmental variables $E \in \mathbb{R}^d$ with $d > 1$ is infeasible with the method, each test can be applied separately for each variable in E. The overall p-value is obtained by multiplying the minimum of the individual p-values by d, i.e. by applying a Bonferroni correction for the number of environmental variables. When applying the function `CondIndTest()` from the `R` package `CondIndTests` with a conditional independence test that does not support a multidimensional environment variable, the described Bonferroni correction is applied.

4.B.1 Kernel conditional independence test

Setting and assumptions. We use the kernel conditional independence test proposed in Zhang et al., 2011. When E is discrete, we use a delta kernel for E, and otherwise an RBF kernel. The test is also applicable when E contains more than one environmental variable as the inputs can be sets of random variables.

4.B.2 Residual prediction test

Setting and assumptions. We do not expect this test to have the correct level when the noise in Eq. (4.1) is not additive. The described procedure does not need to be modified for higher-dimensional and/or continuous environmental variables E.

We consider a version of a Residual Prediction test as proposed in Shah and Bühlmann, 2018 to determine whether $H_{0,S}$ holds at level α for a particular set of variables S. The main idea is to find a suitable basis expansion of f that allows us to regress Y on X_S by reverting back to the linear case. Given an appropriate basis expansion, the *scaled* ordinary least squares residuals can then be tested for possible remaining nonlinear dependencies between the scaled residuals and (E, X_S). The scaling ensures that the resulting test statistic is not a function of the noise variance. Under the null, the scaled residuals are expected to behave roughly like the noise term. In other words, there should be no dependence between the scaled residuals and the environmental variables and X_S, so there should be no signal left in the residuals that could be fitted by a nonparametric method like a random forest using E and X_S as predictors. This necessitates to make an assumption on the noise distribution F_ε, e.g. $\varepsilon \sim \mathcal{N}(0, 1)$.

In order to generalize the method to settings where an appropriate basis expansion of f is unknown, we look at ways to find such a suitable basis expansion automatically by using random features (Rahimi and Recht, 2007; Williams and Seeger, 2001).

Algorithm 3 Residual Prediction tests applied to nonlinear ICP

```
Input: i.i.d. sample of ($Y$, $X_S$, $E$), $\alpha$, $F_\varepsilon$, $B$, a subroutine to compute the basis functions $h_m(\cdot)$ for $m = 1, \dots, M$
Compute the non-linear transformations $h_m(X_S), m = 1, \dots, M$ and create the design matrix $\mathbf{H}_{X_S} \in \mathbb{R}^{n \times M}$ comprising these $M$ nonlinear features.
Regress $Y$ on $\mathbf{H}_{X_S}$ with ordinary least squares.
Predict (a function of) the scaled residuals with the environment variable $E$ and $X_S$.
Compute a statistic for the prediction accuracy to be used as test statistic.
for $b$ from 1 to $B$ do
    Simulate one sample of size $n$ from the assumed noise distribution $F_\varepsilon$.
    Predict (a function of) these simulated values after rescaling with the environment variable $E$ and $X_S$.
    Compute a statistic for the prediction accuracy.
end for
The $B$ simulated values for prediction accuracy yield the empirical null distribution from which the $p$-value is obtained.
Output: Decision about $H_{0,S}$
```

Step 1. The choice of $h_m(X_S), m = 1, \dots, M$ can be based on domain knowledge, e.g. when the nonlinearity in Eq. (4.1) is known to be a polynomial of a given order. If such domain knowledge is not available, the linear basis expansion can be approximated by random features, e.g. using the Nyström method or by random Fourier features. For these methods, the kernel function needs to be chosen as well as the kernel parameters and the number of random features to be generated.

Step 3. For instance, a random forest can be used for the estimation. If the residuals only differ in the second moments, predicting the expectation of the residuals is not sufficient as the predictors E have no discriminative power for this task. In such a setting, the absolute value of the residuals can be predicted to exploit the heterogeneity in the second moments across environments.

Step 4. For instance, the mean squared error can be used here.

Step 5. If the error term is non-Gaussian, the appropriate distribution can be used at this stage to accommodate non-Gaussianity of the noise.

Parameter settings used in simulations. In the simulations, we use $B = 250$ and $\varepsilon \sim \mathcal{N}(0, 1)$. In step 1, we consider the following options: (a) Fourier random features (approach (B)(i) in §4.4), (b) Nyström random features and RBF kernel ((B)(ii)), (c) Nyström random features and polynomial kernel of random degree ((B)(iii)), (d) polynomial basis of random degree ((B)(iv)). The number of random features in (c) and (d) is chosen to be $\lceil n/4 \rceil$. In step 7, we predict the mean as well as the absolute value of the residuals and aggregate the results using a Bonferroni correction.

4.B.3 Invariant environment prediction

Setting and assumptions. The described procedure does not need to be modified for continuous environmental variables E. For higher-dimensional E the test would need to be applied for each variable separately and the resulting p-values would need to be aggregated with a Bonferroni correction.

Algorithm 4 Invariant environment prediction for nonlinear ICP

```
Input: i.i.d. sample of $(Y, X_S, E)$, $\alpha$, subroutine for test in step 5.
1: Split the sample into training and test set.
2: Use the training set to train a model to predict $E$ with $(Y, X_S)$ as predictors.
3: Use the training set to train a model to predict $E$ with $X_S$ as predictors.
4: For both fits, compute the prediction accuracy on the test set.
5: Use a one-sided test at the significance level $\alpha$ to assess whether the prediction accuracy of the fit using $(Y, X_S)$ as predictors is larger than the prediction accuracy of the fit using only $X_S$ as predictors.
Output: Decision about $H_{0,S}$
```

Step 3. When a random forest is used to predict the environment variable, one can also use X_S and a permutation of Y as predictors to ensure the random forest fits are based on the same number of predictor variables. As the number of variables considered for each split in the random forest estimation procedure is a function of the total number of predictor variables, this helps to mitigate differences between the prediction accuracies that are only due to artifacts of the estimation procedure. This is especially relevant for small sets S.

Step 5. For instance, a χ^2 test can be used here. If the null is true and we find the optimal model in both cases, then the out-of-sample performance of both models is statistically indistinguishable as Y is independent of E given X_S. If the null is not true, we expect the model containing the response to perform better as Y contains additional information in this case (since Y is not independent of E given X_S).

Parameter settings used in simulations. In step 1, we use 2/3 of the data points for training and 1/3 for testing. In step 3, we use a random forest to predict the environment variable and use X_S and a permutation of Y as predictors. In step 4, we use the χ^2 test implemented in `prop.test()` (Wilson, 1927) from the `stats` package in R.

4.B.4 Invariant target prediction

Setting and assumptions. The described procedure does not need to be modified for continuous and/or higher-dimensional environmental variables E.

Algorithm 5 Invariant target prediction for nonlinear ICP

```
Input: i.i.d. sample of ($Y$, $X_S$, $E$), $\alpha$, subroutine for test in step 5.
1: Split the sample into training and test set.
2: Use the training set to train a model to predict $Y$ with $(X_S, E)$ as predictors.
3: Use the training set to train a model to predict $Y$ with $X_S$ as predictors.
4: For both fits, compute the prediction accuracy on the test set.
5: Use a one-sided test at the significance level $\alpha$ to assess whether the prediction accuracy of the fit using $(X_S, E)$ as predictors is larger than the prediction accuracy of the fit using only $X_S$ as predictors.
Output: Decision about $H_{0,S}$
```

Step 3. When a random forest is used, one can also use X_S and a permutation of E as predictors to ensure the random forest fit is based on the same number of predictor variables. As the number of variables considered for each split in the random forest estimation procedure is a function of the total number of predictor variables, this helps to mitigate differences between the prediction accuracies that are only due to artifacts of the estimation procedure. This is especially relevant for small sets S. As an alternative to using a random forest, one could use GAMs as the estimation procedure, implying the implicit assumption that the components in $f(X)$ in Eq. (4.1) are additive.

Step 5. For instance, an F-test can be used here. Another option is a Wilcoxon test using the difference between the absolute residuals. If the null is true and we find the optimal model in both cases, then the out-of-sample performance of both models is statistically indistinguishable as Y is independent of E given X_S. If the null is not true, we expect the model additionally containing E to perform better as E contains additional information in this case (since Y is not independent of E given X_S).

Parameter settings used in simulations. In step 1, we use 2/3 of the data points for training and 1/3 for testing. In step 3, to predict Y we use a GAM or a random forest. In step 5, we use an F-test or a Wilcoxon test (`wilcox.test()` from the `stats` package in R). These combinations yield approaches (D)(i) – (iv) in §4.4. When using a random forest in step 3, we use X_S and a permutation of E as predictors.

4.B.5 Invariant residual distribution test

Setting and assumptions. We do not expect this test to have the correct level when the noise in Eq. (4.1) is not additive. It is only applicable to discrete environmental variables. For higher-dimensional E the test would need to be applied for each variable separately and the resulting p-values would need to be aggregated with a Bonferroni correction.

Algorithm 6 Invariant residual distribution test for nonlinear ICP

```
Input: i.i.d. sample of ($Y$, $X_S$, $E$), $\alpha$, subroutine for test in step 4.
Pool the data from all environments and fit a model to predict $Y$ with $X_S$.
Initialize $pv \leftarrow 1$, $t \leftarrow 0$.
for each $e \in \mathcal{E}$ do
    Use a two-sample test to assess whether the residuals of samples from environment $e$ have the same distribution as the residuals of samples from environments in the index set $\mathcal{E}'$ where $\mathcal{E}' = \mathcal{E} \setminus \{e\}$, yielding the $p$-value $pv_e$.
    $t \leftarrow t + 1$
    $pv \leftarrow \min(pv, pv_e)$.
    if $|\mathcal{E}| = 2$ then
        break
    end if
end for
Apply a Bonferroni correction for the number of performed tests $t$: $pv \leftarrow t \cdot pv$.
Output: Decision about $H_{0,S}$
```

Step 1. For instance, one could use a random forest or a GAM as the estimation procedure. The latter implicitly assumes that the components in f in Eq. (4.1) are additive.

Step 4. For instance, a nonparametric test such as Kolmogorov-Smirnov can be used here. Alternatively, we can limit the test to assess equality of first and second moments by first using a Wilcoxon test for the expectation with an one-vs-all scheme as described in the algorithm. Subsequently, Levene's test for homogeneity of variance across groups can be used to test for equality of the second moments of the residual distributions. In this case, the final p-value would be twice the minimum of (a) the Bonferroni-corrected p-value from the one-vs-all Wilcoxon test and (b) the p-value from Levene's test.

Parameter settings used in simulations. In step 1, we use a GAM or a random forest. In step 4, we use both approaches described above, using (a) `ks.test()` from the `stats` package in R (Conover, 1971) and (b) `wilcox.test()` and `levene.test()` (the latter being contained in the `lawstat` package in R (Gastwirth et al., 2015; Levene, 1960)). These combinations yield approaches (E)(i) – (iv) in §4.4.

4.B.6 Invariant conditional quantile prediction

Setting and assumptions. For continuous and/or higher-dimensional environmental variables E the test described in Steps 4 – 11 which assesses whether Exceedance $\perp\!\!\!\perp E$ would need to be modified according to the structure of E.

Algorithm 7 Invariant conditional quantile prediction for nonlinear ICP

Input: i.i.d. sample of $(Y,\, X_S,\, E)$, α, set of quantiles $\mathcal{B}$, subroutine for test in step 7.

```
 1: Initialize $pv \leftarrow 1$, $t \leftarrow 0$.
 2: for each $\beta \in \mathcal{B}$ do
 3:     Predict $1-\beta$ quantile $Q_{1-\beta}(x)$ of $Y|X_S = x$.
 4:     for each $e \in \mathcal{E}$ do
 5:         Define one-vs-all environment $I = \mathbb{1}_{\{E=e\}}$
 6:         Define exceedance $E_{1-\beta} = \mathbb{1}_{\{Y > \hat{Q}_{1-\beta}(x)\}}$
 7:         Test whether $E_{1-\beta}$ is independent of $I$: $pv_{e,\beta} \leftarrow$ StatTest$(E_{1-\beta}, I, \alpha)$
 8:         $t \leftarrow t + 1$, $pv \leftarrow \min(pv, pv_{e,\beta})$
 9:         if $|\mathcal{E}| = 2$ then
10:             break
11:         end if
12:     end for
13: end for
14: Apply a Bonferroni correction for the number of performed tests $t$: $pv \leftarrow t \cdot pv$
```

Output: Decision about $H_{0,S}$

Step 3. For instance, a Quantile Regression Forest (Meinshausen, 2006) can be used here.

Step 7. For instance, Fisher's exact test can be used here by computing the 2×2 contingency table of the exceedance of the residuals for the quantile $1 - \beta$ for $I = 0$ and for $I = 1$.

Parameter settings used in simulations. In step 3, we use a quantile regression forest for $\mathcal{B} = \{0.1, 0.5, 0.9\}$. In step 7, we use `fisher.test()` from the `stats` package in R.

4.B.7 Overview of conditional independence tests in `CondIndTests` package

The described conditional independence tests are available in the R package `CondIndTests`. A wrapper function `CondIndTest()` is provided which takes the respective test as the argument `method`. The package supports the estimation procedures, subroutines and statistical tests shown in Table 4.2. The column E indicates whether the environmental variables can be discrete ('D'), continuous ('C'), or both; the column d shows the supported dimensionality of E.

As described at the beginning of Appendix 4.B, a Bonferroni correction is applied when calling the function `CondIndTest()` with a conditional independence test that does not support a multidimensional environment variable. Similarly, a Bonferroni correction is applied when the first input argument `Y` to the respective test is multidimensional and if the specified test does not support this internally.

Appendix 4.C Experimental settings for numerical studies

For each simulation, we compare the performance of all methods and conditional independence tests while choosing the following parameters randomly (but keeping them constant for one simulation): In total, there are 27478 simulations from 1240 distinct settings that are evaluated for each of the 22 considered methods.

(i) **Sample size.** Sample size 'n' is chosen randomly in the set $\{100, 200, 500, 2000, 5000\}$.

(ii) **Target variable.** We sample one of the variables in the graph in Figure 4.9 at random as a target variable (variable 'target' is chosen uniformly from $\{1, 2, \ldots, 6\}$ in other words).

(iii) **Tail behavior of the noise.** The noise η_k for $k = 1, \ldots, 6$ is sampled from a t-distribution and the degrees of freedom are chosen at random from $\mathrm{df} \in \{2, 3, 5, 10, 20, 50, 100\}$, where the latter is very close to a Gaussian distribution.

(iv) **Multiplicative or additive effects.** For each simulation setting, we determine whether $g_k(\cdot)$ has additive or multiplicative components. We sample additive components of the form $g_k(Z_{\mathrm{pa}_k}) = \sum_{j \in pa_k} f(\epsilon_{j,k} \cdot Z_j)$ (*multiplic* = FALSE) and multiplicative components of the form $g_k(Z_{\mathrm{pa}_k}) = \prod_{j \in pa_k} f(\epsilon_{j,k} \cdot Z_j)$ (*multiplic* = TRUE) with equal probability, where the signs $\epsilon_{j,k} \in \{-1, 1\}$ are as shown in Figure 4.9 along the relevant arrows.

Table 4.2.: Overview of implemented test combinations in `CondIndTests` package

CIT	R function name/METHOD	TEST	E	d
(A)	`KCI()`		D/C	≥ 1
	KCI (without GP support)	–		
(B)	`ResidualPredictionTest()`		D/C	≥ 1
	Residual prediction test with	–		
	– Nyström random features (RBF and polynomial kernel)			
	– Fourier random features			
	– fixed basis expansion			
(C)	`InvariantEnvironmentPrediction()`		D/C	1
	Random forest	χ^2 test (`prop.test()`)		
		Wilcoxon test (`wilcox.test()`)		
(D)	`InvariantTargetPrediction()`		D/C	≥ 1
	Random forest	F-Test		
	GAM	Wilcoxon test (`wilcox.test()`)		
(E)	`InvariantResidualDistributionTest()`		D	1
	Random forest	Kolmogorov-Smirnov (`ks.test()`)		
	GAM	Levene's test + Wilcoxon test (`levene.test()`, `wilcox.test()`)		
(F)	`InvariantConditionalQuantilePrediction()`		D	1
	Quantile regression forest	Fisher's exact test (`fisher.test()`)		

(v) **Shift- or do-Interventions.** The variable ‘shift’ is set with equal probability to either TRUE (shift-interventions) or FALSE (do-interventions). For do-interventions we replace the structural equation of the intervened variable $k \in \{1, \dots, q\}$ by

$$Z_k \leftarrow e_k,$$

where e_k is the randomly chosen intervention value which is sampled i.i.d for each observation as described under (vi). For shift-interventions, the value e_k is added as

$$Z_k \leftarrow g_k(Z_{\mathrm{pa}_k}) + \eta_k + e_k.$$

See for example §5 of Peters et al., 2016 for a more detailed discussion of shift interventions.

(vi) **Strength of interventions** The intervention values e_k are chosen independently for all variables from a t-distribution with ‘df’ degrees of freedom, shifted by a constant ‘meanshift’ (chosen uniformly at random in $\{0, 0.1, 0.2, 0.5, 1, 2, 5, 10\}$, and scaled by a factor ‘strength’, chosen uniformly at random in $\{0, 0.1, 0.2, 0.5, 1, 2, 5, 10\}$).

(vii) **Non-linearities.** For the functions $f = f_{\mathrm{id}}$ we consider the following four nonlinear functions, where the index ‘id’ is sampled uniformly from $\{1, 2, 3, 4\}$ and the same nonlinearity is used throughout the graph:

$$\begin{aligned} f_1(x) &= x, \\ f_2(x) &= \max\{0, x\}, \\ f_3(x) &= \mathrm{sign}(x) \cdot \sqrt{|x|}, \\ f_4(x) &= \sin(2\pi x), \end{aligned}$$

(viii) **Location of interventions.** Each sample is independently assigned into one environment in $\mathcal{E} = \{1, 2, 3\}$, where $\{1\}$ corresponds to observational data, that is all samples in environment $\{1\}$ are sampled as observational data, where samples in environments $\{2, 3\}$ are intervention data. The intervention targets and strengths for samples in environment $\{2\}$ are drawn as per the description below and kept constant for all samples in environment $\{2\}$ and then analogously for environment $\{3\}$, where intervention targets are drawn independently and identically to environment $\{2\}$. The variable ‘interv’ is set uniformly at random to one of the values {'all','rand','close'}. If it is equal to 'all', then interventions in environments $\{2, 3\}$ occur at all variables except for the target variable. If it is equal to 'rand', then interventions occur at one ancestor, chosen uniformly at random, and one descendant of the target variable, chosen again uniformly at random (the empty set is chosen in case there are no ancestors or descendants). Finally, if it is equal to 'close', interventions occur at one parent, chosen uniformly at random, and one child of the target variable, chosen again uniformly at random (and again no interventions occur if these sets are empty).

Appendix 4.D Additional experimental results

Figures 4.13 – 4.16 show the equivalent to Figure 4.11, separately for target variables 2, 3, 4 and 6. For the sample size n, we observe that increasing it from 2000 to 5000 decreases power in case of target variable 4. This behavior can be explained by the fact that when testing S^* in Eq. (4.3), the null is rejected too often as the bias in the estimation performed as part of the conditional independence test yields deviations from the null that become significant with increasing sample size. For the nonlinearity, we find that the function $f_4(x) = \sin(2\pi x)$ is the most challenging one among the nonlinearities considered. It is associated with very low Jaccard similarity values for the target variables that do have parents. For the intervention type, it may seem surprising that 'all' does not yield the largest power. A possible explanation is that intervening on all variables except for the target yields more similar intervention settings—the intervention targets do not differ between environments 2 and 3, even though the strength of the interventions is different. So more heterogeneity between the intervention environments, i.e. also having different intervention targets, seems to improve performance in terms of Jaccard similarity. Lastly, we see that power is often higher for additive parental contributions than for multiplicative ones.

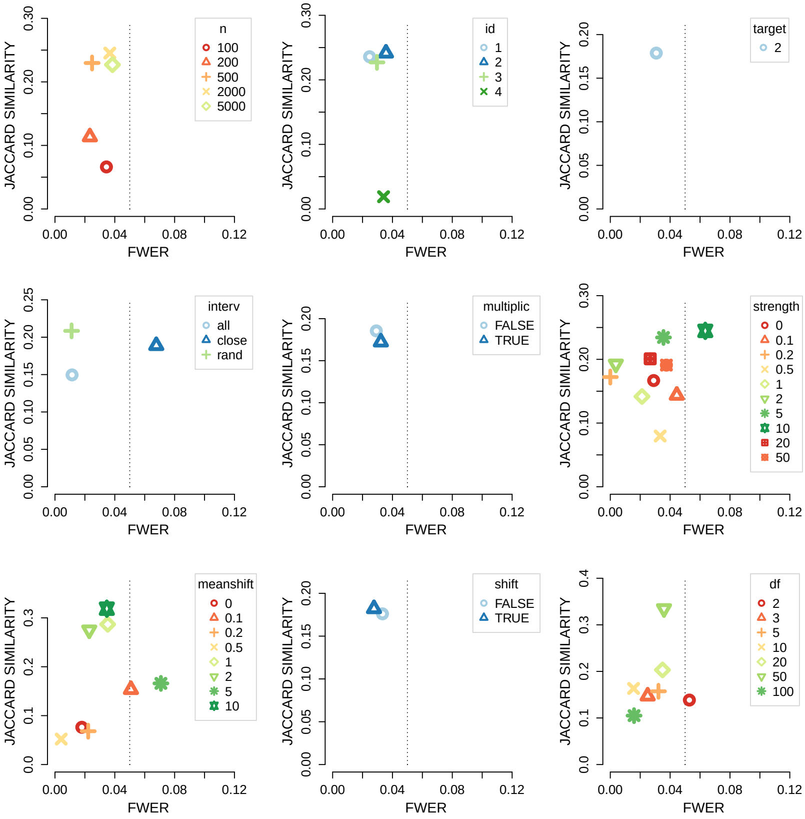


Figure 4.13.: Average Jaccard similarity over the conditional independence tests (A) – (F) (y-axis) against average FWER (x-axis) when estimating the parents of variable 2. The figure is otherwise generated analogously to Figure 4.11.

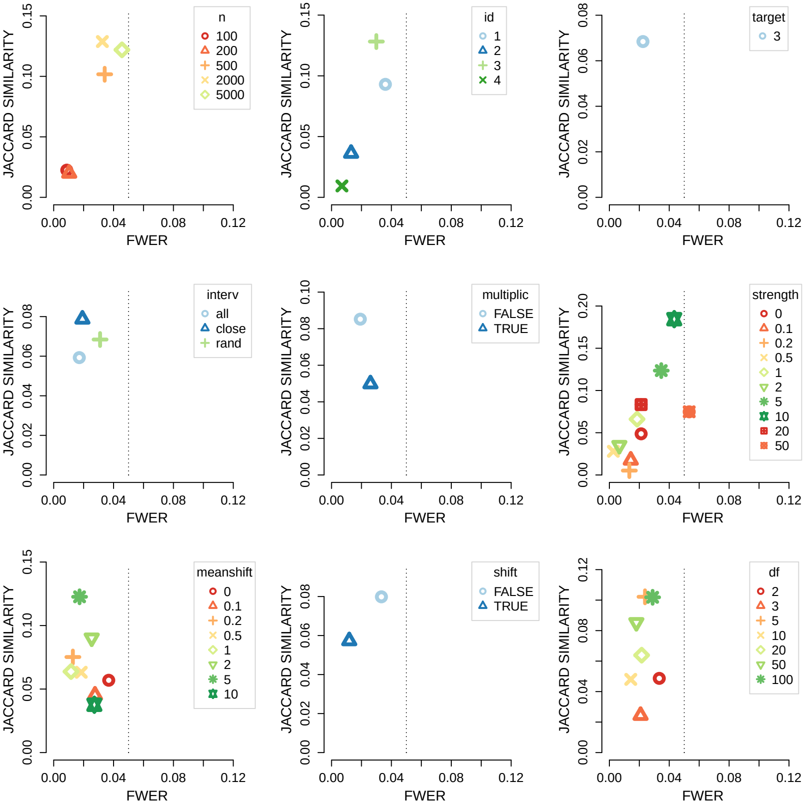


Figure 4.14.: Average Jaccard similarity over the conditional independence tests (A) – (F) (y-axis) against average FWER (x-axis) when estimating the parents of variable 3. The figure is otherwise generated analogously to Figure 4.11.

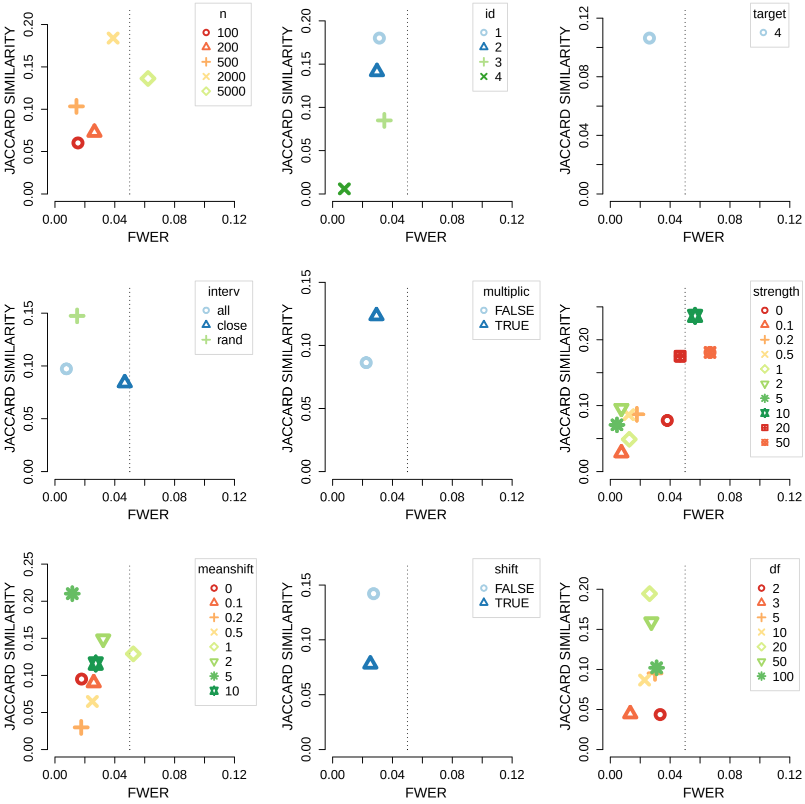


Figure 4.15.: Average Jaccard similarity over the conditional independence tests (A) – (F) (y-axis) against average FWER (x-axis) when estimating the parents of variable 4. The figure is otherwise generated analogously to Figure 4.11.

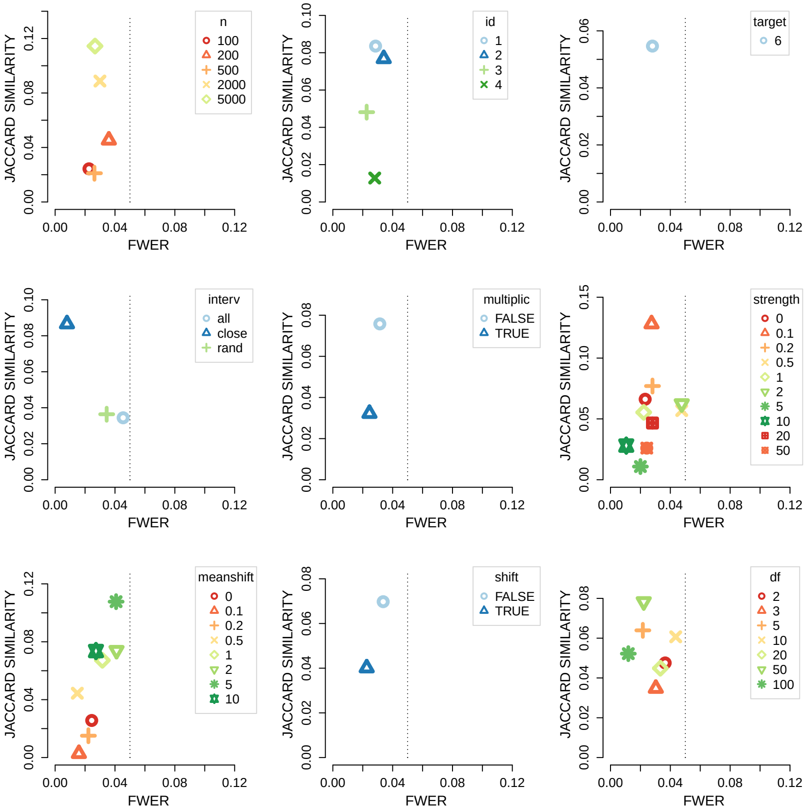


Figure 4.16.: Average Jaccard similarity over the conditional independence tests (A) – (F) (y-axis) against average FWER (x-axis) when estimating the parents of variable 6. The figure is otherwise generated analogously to Figure 4.11.

Appendix 4.E Example

Here we illustrate the methods presented in this manuscript by considering a causal DAG $X_1 \rightarrow X_2 \rightarrow X_3$. Figure 4.17 visualizes the generated data. There are six environments with shift interventions. The latter act on X_1 in two environments (green, yellow) and on X_3 in four environments (green, cyan, blue, magenta). The red environment consists of observational data. We run the proposed approaches (A) – (F) to retrieve the parents of X_2, i.e. $S^* = \{X_1\}$. Below we give an overview of which sets were accepted by the respective methods with $\alpha = 0.05$. We see that approaches (A), (B)(i)+(ii), (E)(i)-(iii) and (F) retrieve S^* correctly, while the other approaches return the empty set.

CIT	$S_0 = \{\}$	$S_1 = \{X_1\}$	$S_2 = \{X_3\}$	$S_3 = \{X_1, X_3\}$	$\hat{S}$
(A)		✓		✓	$\{X_1\}$
(B)(i)		✓		✓	$\{X_1\}$
(B)(ii)		✓		✓	$\{X_1\}$
(B)(iii)					$\{\}$
(B)(iv)					$\{\}$
(C)		✓	✓	✓	$\{\}$
(D)(i)	✓	✓		✓	$\{\}$
(D)(ii)	✓	✓	✓	✓	$\{\}$
(D)(iii)					$\{\}$
(D)(iv)	✓			✓	$\{\}$
(E)(i)		✓		✓	$\{X_1\}$
(E)(ii)		✓		✓	$\{X_1\}$
(E)(iii)		✓			$\{X_1\}$
(E)(iv)					$\{\}$
(F)		✓		✓	$\{X_1\}$

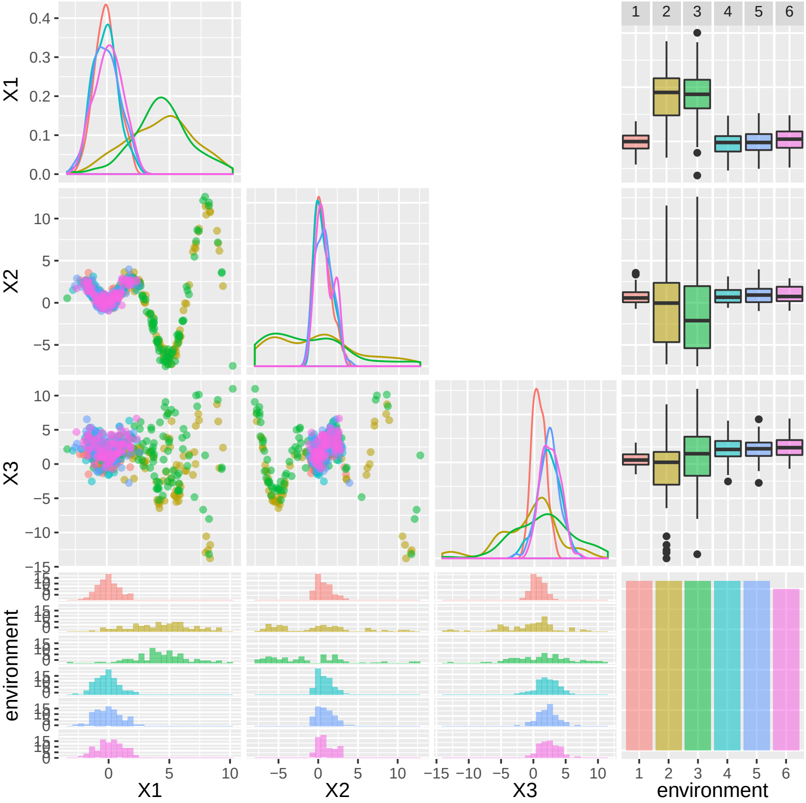


Figure 4.17.: Visualization of the sample considered in the example in Appendix 4.E.

Chapter 5.

Conditional variance penalties and domain shift robustness

When training a deep neural network for supervised image classification, one can broadly distinguish between two types of latent features of images that will drive the classification. Following the notation of Gong et al. (2016), we can divide latent features into (i) 'core' or 'conditionally invariant' features X^{core} whose distribution $X^{\text{core}}|Y$, conditional on the class Y, does not change substantially across domains and (ii) 'style' or 'orthogonal' features X^{style} whose distribution $X^{\text{style}}|Y$ can change substantially across domains. These latter orthogonal features would generally include features such as position, rotation, image quality or brightness but also more complex ones like hair color or posture for images of persons. Guarding against future adversarial domain shifts implies that the influence of the second type of style features in the prediction has to be limited. In contrast to previous work, we assume that the domain itself is not observed and hence a latent variable. Therefore, we can not directly see the distributional change of features across different domains.

We do assume, however, that we can sometimes observe a typically discrete identifier or ID variable. We know in some applications, for example, that two images show the same person, and ID then refers to the identity of the person. In data augmentation, we generate several images from the same original image, and ID then refers to the relevant original image. The method requires only a small fraction of images to have an ID variable.

The causal framework of Gong et al. (2016) is adapted by adding the identifier ID variable to the model and making domain a latent variable. We group data samples if they share the same class and identifier $(Y, \mathrm{ID}) = (y, \mathrm{id})$ and penalize the conditional variance of the prediction if we condition on (Y, ID). The regularization is equivalent to penalizing with an appropriate graph Laplacian. Using this grouping-by-ID approach is shown to protect against shifts in the distribution of the style variables for both regression and classification models. Specifically, the conditional variance penalty CoRe is shown to be equivalent to minimizing the risk under noise interventions in a regression setting and is shown to lead to adversarial risk consistency in a partially linear classification setting.

We show empirically that the CoRe penalty substantially improves performance in settings where domains changes occur in terms of image quality, brightness and color while we also look at more complex changes such as changes in movement and posture. The attractive property is that the type of domain change on future data does not need to be known a priori.

5.1. Introduction

Deep neural networks (DNNs) have achieved outstanding performance on prediction tasks like visual object and speech recognition (He et al., 2015; Krizhevsky et al., 2012; Szegedy et al., 2015). Issues can arise when the learned representations rely on dependencies that vanish in test distributions, see for example Csurka (2017), Quionero-Candela et al. (2009), and Torralba and Efros (2011) and references therein. Such domain shifts can be caused by changing conditions such as color, background or location changes. Predictive performance is then likely to degrade. The “Russian tank legend” is an example where the training data was subject to sampling biases that were not replicated in the real world. Concretely, the story relates how a machine learning system was trained to distinguish between Russian and American tanks from photos. The accuracy was very high but only due to the fact that all images of Russian tanks tended to be of bad quality while the photos of American tanks were not. The system learned to discriminate between images of different qualities but would have failed badly in practice (Emspak, 2016)[1]. For a directly equivalent example, see §5.7.2. Existing biases in datasets used for training machine learning algorithms tend to be replicated in the estimated models (Bolukbasi et al., 2016). For another example involving Google’s photo app, see Crawford (2016) and Emspak (2016). In §5.7 we show many examples where unwanted biases in the training data are picked up by the trained model. As any bias in the training data is in general used to discriminate between classes, these biases will persist in future classifications, raising also

---

[1] A different version of this story can be found in Yudkowsky, 2008.

considerations of fairness and discrimination (Barocas and Selbst, 2016).

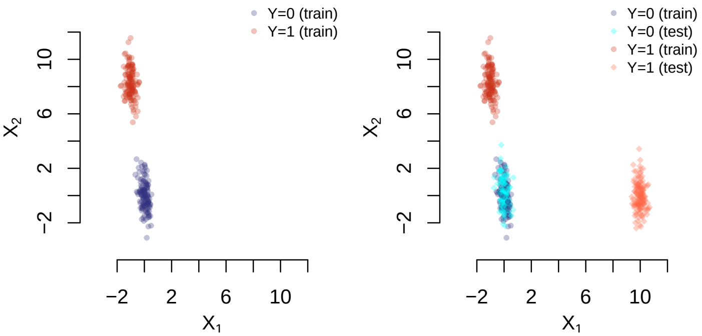
Addressing the issues outlined above, we propose COnditional variance REgularization (CoRe) to give differential weight to different latent features. Conceptually, we take a causal view of the data generating process and categorize the latent data generating factors into 'conditionally invariant' (*core*) and 'orthogonal' (*style*) features, as in Gong et al. (2016). It is desirable that a classifier uses only the *core* features as they pertain to the target of interest in a stable and coherent fashion. Basing a prediction on the *core* features alone yields a stable predictive accuracy even if the *style* features are altered. CoRe yields an estimator which is approximately invariant under changes in the conditional distribution of the style features (conditional on the class labels). Consequently, it is robust with respect to *adversarial domain shifts*, arising through arbitrarily strong interventions on the style features. CoRe relies on the fact that for certain datasets we can observe 'grouped observations' in the sense that we observe the same object under different conditions. Rather than pooling over all examples, CoRe exploits knowledge about this grouping, i.e. that a number of instances relate to the same object. By penalizing between-object variation of the prediction less than variation of the prediction for the same object, we can steer the prediction to be based more on the latent *core* features and less on the latent *style* features.

The remainder of this manuscript is structured as follows: §5.2 starts with a few motivating examples, showing simple settings where the style features change in the test distribution such that standard empirical risk minimization approaches would fail. In §5.3 we review related work, introduce notation in §5.4 and in §5.5 we formally introduce conditional variance regularization CoRe. In §5.6, CoRe is shown to be equivalent to minimizing the risk under noise interventions in a regression setting and is shown to lead to adversarial risk consistency in a partially linear classification setting. In §5.7 we evaluate the performance of CoRe in a variety of experiments.

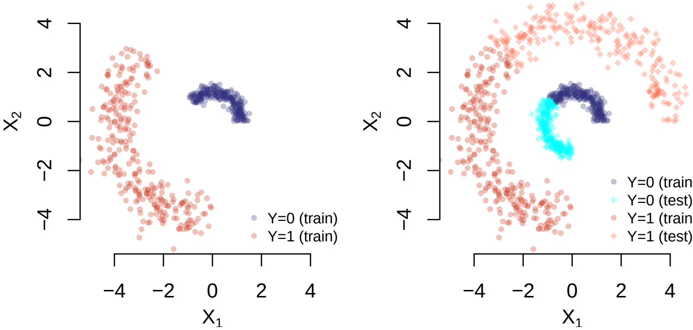
To summarize, our contributions are the following:

(i) **Causal framework.** We extend the causal framework of Gong et al. (2016) to address situations where the domain variable itself is latent.

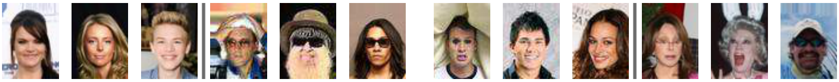
(ii) **Conditional variance penalties and distributional robustness.** We introduce conditional variance penalties, which are equivalent to a suitable graph Laplacian penalty. For classification, we



(a) Example 1, training set. (b) Example 1, test set.



(c) Example 2, training set. (d) Example 2, test set.



(e) Example 3, training set. (f) Example 3, test set.

Figure 5.1.: Three motivating examples: a linear example in (a) and (b), a nonlinear example in (c) and (d) and an example where the goal is to predict whether a person is wearing glasses in (e) and (f). The distributions are shifted in test data by style interventions where style in example (a/b) is the linear direction $(1, -0.75)$, the polar angle in example (c/d), and the image quality in example (e/f). In this latter example, a 5-layer CNN achieves 0% training error and 2% test error for images that are sampled from the same distribution as the training images (e), but a 65% error rate on images where the confounding between image quality and glasses is changed (f). See §5.7.2 for more details.

show in Theorem 5.2 that we can achieve consistency under a risk definition that allows adversarial domain changes. For regression, we show in Theorem 5.3 that estimator achieves distributional robustness against intervention distributions where the noise variance of domain-specific noise is increased. A one-to-one correspondence between the penalty parameter and the set of distributions we are protected against is shown.

(iii) **Software.** We illustrate our ideas using synthetic and real-data experiments. A TensorFlow implementation of CoRe as well as code to reproduce some of the experimental results are available at `https://github.com/christinaheinze/core`.

5.2. Motivating examples

To motivate the methodology we propose, consider the examples shown in Figure 5.1. Example 1 shows a setting where a linear decision boundary is suitable. Panel (a) shows a subsample of the training data where class 1 is associated with red points, dark blue points correspond to class 0. If we were asked to draw a decision boundary based on the training data, we would probably choose one that is approximately horizontal. The style feature here corresponds to a linear direction $(1, -0.75)^t$. Panel (b) shows a subsample of the test set where the style feature is intervened upon for class 1 observations: class 1 is associated with orange squares, cyan squares correspond to class 0. Clearly, a horizontal decision boundary would have misclassified all test points of class 1.

Example 2 shows a setting where a nonlinear decision boundary is required. Here, the *core* feature corresponds to the distance from the origin while the style feature corresponds to the angle between the x_1-axis and the vector from the origin to (x_1, x_2). Panel (c) shows a subsample of the training data and panel (d) additionally shows a subsample of the test data where the style—i.e. the distribution of the angle—is intervened upon. Clearly, a circular decision boundary yields optimal performance on both training and test set but is unlikely to be found by a standard classification algorithm when only using the training set for the estimation. We will return to these examples in §5.5.4.

Lastly, we mimic the Russian tank legend in the third example by manipulating the face images from the CelebA dataset (Liu et al., 2015): in the training set images of class “wearing glasses” are associated with a lower image quality than images of class “not wearing glasses”. Examples are shown in panel (e). In the test set, this relation is reversed, i.e. images showing persons wearing glasses are of higher quality than images of persons without glasses, with examples in panel (f). We will return to this example in §5.7.2 and show that training a convolutional neural network to distinguish between people wearing glasses or not works well on test data that are drawn from the same distribution (with error rates below 2%) but fails entirely on the shown test data, with error rates worse than 65%.

5.3. Related work

For general distributional robustness, the aim is to learn

$$\operatorname{argmin}_\theta \sup_{F \in \mathcal{F}} E_F(\ell(Y, f_\theta(X))) \tag{5.1}$$

for a given set $\mathcal{F}$ of distributions, loss ℓ, and prediction $f_\theta(x)$. The set $\mathcal{F}$ is the set of distributions on which one would like the estimator to achieve a guaranteed performance bound and the set is often taken to be of the form $\mathcal{F} = \mathcal{F}_\epsilon(P_0)$ with

$$\mathcal{F}_\epsilon(P_0) := \{\text{distributions } Q \text{ such that } D(Q, P_0) \le \epsilon\}, \tag{5.2}$$

with $\epsilon > 0$ a small constant and $D(Q, P_0)$ being, for example, a ϕ-divergence (Bagnell, 2005; Ben-Tal et al., 2013; Namkoong and Duchi, 2017) or a Wasserstein-distance (Gao et al., 2017; Shafieezadeh-Abadeh et al., 2017; Sinha et al., 2018). The distribution P_0 can be the true (but generally unknown) population distribution P from which the data were drawn or its empirical counterpart P_n. The distributionally robust targets (5.1) can often be expressed in penalized form; see Gao et al., 2017; Sinha et al., 2018; Xu et al., 2009.

In this work, we do not try to achieve robustness with respect to a set of distributions that are pre-defined by a Kullback-Leibler divergence or a Wasserstein metric as in (5.2). We try to achieve robustness against a set of distributions that are generated by interventions on latent style variables. As such the right distribution set $\mathcal{F}$ in (5.1) has to be learnt from data and we need a causal model to define the set of distributions we would like to protect ourselves against.

Similar to this work in terms of their goals are the work of Gong et al., 2016 and Domain-Adversarial Neural Networks (DANN) proposed in Ganin et al. (2016), an approach motivated by the work of Ben-David et al., 2007. The main idea of Ganin et al. (2016) is to learn a representation that contains no discriminative information about the origin of the input (source or target domain). This is achieved by an adversarial training procedure: the loss on domain classification is maximized while the loss of the target prediction task is minimized simultaneously. The data generating process assumed in Gong et al., 2016 is similar to our model, introduced in §5.4.2, where we detail the similarities and differences between the models (cf. Figure 5.2). Gong et al. (2016) identify the conditionally independent features by adjusting a transformation of the variables to minimize the squared MMD distance between distributions in different domains[2]. The fundamental difference between these very promising methods and our approach is that we use a different data basis. The domain identifier is explicitly observable in Gong et al., 2016 and Ganin et al. (2016), while it is latent in our approach. In contrast, we exploit the presence of an identifier variable ID that relates to the identity of an object (for example identifying a person). In other words, we do not assume that we have data from different domains but just different realizations of the same object under different interventions.

Causal modeling has related aims to the setting of transfer learning and guarding against adversarial domain shifts. Specifically, causal models have the defining advantage that the predictions will be valid even under arbitrarily large interventions on all predictor variables (Aldrich, 1989; Haavelmo, 1944; Magliacane et al., 2018; Pearl, 2009; Peters et al., 2016; Rojas-Carulla et al., 2018; Schölkopf et al., 2012; Yu et al., 2017; Zhang et al., 2013a, 2015a). There are two difficulties in transferring these results to the setting of adversarial domain changes in image classification. The first hurdle is that the classification task is typically anti-causal since the image we use as a predictor is a descendant of the true class of the object we are interested in rather than the other way around. The second challenge is that we do not want to guard against arbitrary interventions on any or all variables but only would like to guard against a shift of the style features. It is hence not immediately obvious how standard causal inference can be used to guard against large domain shifts. Recently, various approaches have been proposed that leverage causal motivations for deep learning or

---

[2] The distinction between 'conditionally independent' features and 'conditionally transferable' (which is the former modulo location and scale transformations) is for our purposes not relevant as we do not make a linearity assumption in general.

use deep learning for causal inference. Chalupka et al., 2014 characterize learning the visual causes for a certain target behavior and thereby model perceiving systems. Various approaches focus on cause-effect inference where the goal is to find the causal relation between two random variables, X and Y (Goudet et al., 2017; Lopez-Paz and Oquab, 2017; Lopez-Paz et al., 2017). Lopez-Paz et al., 2017 propose the Neural Causation Coefficient (NCC) to estimate the probability of X causing Y and apply it to finding the causal relations between image features. Specifically, the NCC is used to distinguish between features of objects and features of the objects' contexts. Lopez-Paz and Oquab, 2017 note the similarity between structural equation modeling and CGANs (Mirza and Osindero, 2014). One CGAN is fitted in the direction $X \to Y$ and another one is fitted for $Y \to X$. Based on a two-sample test statistic, the estimated causal direction is returned. Goudet et al., 2017 use generative neural networks for cause-effect inference, to identify v-structures and to orient the edges of a given graph skeleton. Bahadori et al., 2017 devise a regularizer that combines an ℓ_1 penalty with weights corresponding to the estimated probability of the respective feature being causal for the target. The latter estimates are obtained by causality detection networks or scores such as estimated by the NCC. Besserve et al., 2018 draw connections between GANs and causal generative models, using a group theoretic framework. Kocaoglu et al., 2018 propose causal implicit generative models to sample from conditional as well as interventional distributions, using a conditional GAN architecture (CausalGAN). The generator structure needs to inherit its neural connections from the causal graph, i.e. the causal graph structure must be known. Louizos et al., 2017 propose the use of deep latent variable models and proxy variables to estimate individual treatment effects. Kilbertus et al., 2017 exploit causal reasoning to characterize fairness considerations in machine learning. Distinguishing between the protected attribute and its proxies, they derive causal non-discrimination criteria. The resulting algorithms avoiding proxy discrimination require classifiers to be constant as a function of the proxy variables in the causal graph, thereby bearing some structural similarity to our style features. Distinguishing between core and style features can be seen as some form of disentangling factors of variation. Estimating disentangled factors of variation has gathered a lot of interested in the context of generative modeling (Bouchacourt et al., 2018; Chen et al., 2016; Higgins et al., 2017). For example, Matsuo et al., 2017 propose a "Transform Invariant Autoencoder" where the goal is to reduce the dependence of the latent representation on a specified transform of the object in the original image. Specifically, Matsuo et al. (2017)

predefine location as the style feature and the goal is to learn a latent representation that does not include X^{style}. Our approach is different as we do not predefine which features are considered style features. The style features in our approach could be location but also image quality, posture, brightness, background and contextual information or something entirely unknown. We try to learn a representation of style and core features from data by exploiting the grouping of training samples. Additionally, the approach in Matsuo et al., 2017 cannot effectively deal with a confounding situation where the distribution of the style features differs conditional on the class (this is a natural restriction for their work, however, as the class label is not even observed in the autoencoder setting). As in CoRe, Bouchacourt et al., 2018 exploit grouped observations. In a variational autoencoder framework, they aim to separate style and content—they assume that samples within a group share a common but unknown value for one of the factors of variation while the style can differ. Denton and Birodkar, 2017 propose an autoencoder framework to disentangle style and content in videos using an adversarial loss term where the grouping structure induced by clip identity is exploited. Here we try to solve a classification task directly without estimating the latent factors explicitly as in a generative framework.

5.4. Setting

We first describe the general notation used before describing the causal graph that allows us to compare the setting of adversarial domain shifts to transfer learning, domain adaptation and adversarial examples.

5.4.1. Notation

Let $Y \in \mathcal{Y}$ be a target of interest. Typically $\mathcal{Y} = \mathbb{R}$ for regression or $\mathcal{Y} = \{1, \dots, K\}$ in classification with K classes. Let $X \in \mathbb{R}^p$ be a predictor, for example the p pixels of an image. The prediction $\hat{y}$ for y, given $X = x$, is of the form $f_\theta(x)$ for a suitable function f_θ with parameters $\theta \in \mathbb{R}^d$, where the parameters θ correspond to the weights in a DNN. For regression, $f_\theta(x) \in \mathbb{R}$, whereas for classification $f_\theta(x)$ corresponds to the conditional probability distribution of $Y \in \{1, \dots, K\}$. Let ℓ be a suitable loss that maps y and $\hat{y} = f_\theta(x)$ to $\mathbb{R}^+$. A standard goal is to minimize the expected loss or risk

$$L(\theta) \;=\; E\Big[\ell(Y, f_\theta(X))\Big].$$

Let (x_i, y_i) for $i = 1, \ldots, n$ be the samples that constitute the training data and $\hat{y}_i = f_\theta(x_i)$ the prediction for y_i. The standard approach is to simply pool over all available observations, ignoring any grouping information that might be available. The pooled estimator thus treats all examples identically by summing over the empirical loss as

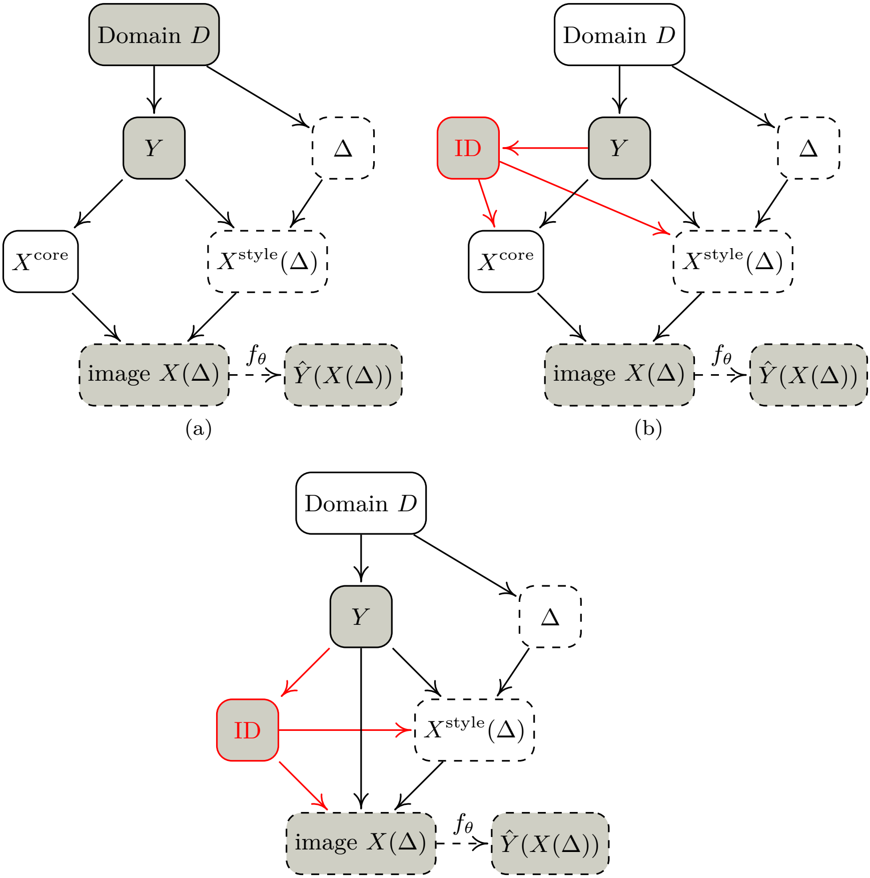
$$\hat{\theta}^{pool} \;=\; \mathrm{argmin}_\theta \; \frac{1}{n}\sum_{i=1}^{n} \Big[\ell\big(y_i, f_\theta(x_i)\big)\Big] + \gamma \cdot \mathrm{pen}(\theta), \tag{5.3}$$

where $\mathrm{pen}(\theta)$ is a complexity penalty, for example a ridge term $\|\theta\|_2^2$ on the weights of a convolutional neural network. All examples that compare to the pooled estimator will include a ridge penalty as default. Different penalties can exploit underlying geometries, such as the Laplacian regularized least squares (Belkin et al., 2006). In fact, the proposed estimator will be of this form, exploiting grouping information in the data.

5.4.2. Causal graph

The full causal structural model for all variables is shown in the panel (b) of Figure 5.2. The domain variable D is latent, in contrast to Gong et al. (2016) whose model is shown in panel (a) of Figure 5.2. We add the ID variable (identity of a person, for example), whose distribution can change conditional on Y. In Figure 5.2, $Y \rightarrow$ ID but in some settings it might be more plausible to consider ID $\rightarrow Y$. For our proposed method both options are possible since we condition on ID and Y. The ID variable is used to group observations. The variable is typically discrete and relates to the identity of the underlying object. The variable can be assumed to be latent in the setting of Gong et al. (2016).

The rest of the graph is in analogy to Gong et al. (2016). The prediction is anti-causal, that is the predictors X that we use for $\hat{Y}$ are non-ancestral to Y. In other words, the class label is causal for the image and not the other way around. The causal effect from the class label Y on the image X is mediated via two types of latent variables: the so-called *core* or 'conditionally invariant' features X^{core} and the orthogonal or *style* features X^{style}. The distinguishing factor between the two is that external interventions Δ are possible on the *style* features but not on the *core* features. If the



(a) (b) (c)

Figure 5.2.: Observed quantities are shown as shaded nodes; nodes of latent quantities are transparent. Left: data generating process for the considered model as in Gong et al. (2016), where the effect of the domain on the orthogonal features X^{style} is mediated via unobserved noise Δ. The style interventions and all its descendants are shown as nodes with dashed borders to highlight variables that are affected by style interventions. Observed variables are shaded. Middle: our setting. The domain itself is unobserved but we can now observe the (typically discrete) ID variable we use for grouping. Right: the same model as in the middle if marginalizing out over the unobserved X^{core}.

interventions Δ have different distributions in different domains, then the distribution $P(X^{\text{core}}|Y)$ is constant across domains while $P(X^{\text{style}}|Y)$ can change across domains. The style features X^{style} and Y are confounded, in other words, by the latent domain D. In contrast, the *core* or 'conditionally invariant' features satisfy $X^{\text{core}} \perp\!\!\!\perp D|Y$. The *style* variable can include point of view, image quality, resolution, rotations, color changes, body posture, movement etc. and will in general be context-dependent[3]. The style intervention variable Δ influences both the latent style X^{style}, and hence also the image X. In potential outcome notation, we let $X^{\text{style}}(\Delta = \delta)$ be the style under intervention $\Delta = \delta$ and $X(Y, \text{ID}, \Delta = \delta)$ the image for class Y, identity ID and style intervention Δ. The latter is sometimes abbreviated as $X(\Delta = \delta)$ for notational simplicity. Finally, $f_\theta(X(\Delta = \delta))$ is the prediction under the style intervention $\Delta = \delta$. For a formal justification of using a causal graph and potential outcome notation simultaneously see Richardson and Robins (2013).

To be specific, if not mentioned otherwise we will assume a causal graph as follows. For independent $\varepsilon_Y, \varepsilon_{\text{ID}}, \varepsilon_{\text{style}}$ in $\mathbb{R}, \mathbb{R}, \mathbb{R}^q$ respectively with positive density on their support and continuously differentiable functions k_y, k_{id}, and $k_{\text{style}}, k_{\text{core}}, k_x$,

$$\begin{aligned} Y &\leftarrow k_y(D, \varepsilon_Y) \\ \text{identifier ID} &\leftarrow k_{\text{id}}(Y, \varepsilon_{\text{ID}}) \\ \text{core or conditionally invariant features } X^{\text{core}} &\leftarrow k_{\text{core}}(Y, \text{ID}) \\ \text{style or orthogonal features } X^{\text{style}} &\leftarrow k_{\text{style}}(Y, \text{ID}, \varepsilon_{\text{style}}) + \Delta \\ \text{image } X &\leftarrow k_x(X^{\text{core}}, X^{\text{style}}). \end{aligned} \tag{5.4}$$

Of these, Y, X and ID are observed whereas $D, X^{\text{core}}, X^{\text{style}}, \Delta$ and the noise variables are latent. The model can be generalized by allowing further independent noise terms inside k_x and k_{core} but the model above is already fairly general and keeps notational simplicity more constrained than the fully general version.

---

[3] The type of features we regard as style and which ones we regard as core features can conceivably change depending on the circumstances—for instance, is the color "gray" an integral part of the object "elephant" or can it be changed so that a colored elephant is still considered to be an elephant?

5.4.3. Data

To summarize, we assume we have n samples $(x_i, y_i, \mathrm{id}_i)$ for $i = 1, \dots, n$, where the observations id_i with $i = 1, \dots, n$ of variable ID can also contain unobserved values. Let $m \le n$ be the number of unique realizations of (Y, ID) and let $S_1, \dots, S_m$ be a partition of $\{1, \dots, n\}$ such that, for each $j \in \{1, \dots, m\}$, the realizations (y_i, id_i) are identical[4] for all $i \in S_j$. The cardinality of S_j is denoted by $n_j := |S_j| \ge 1$. Then $n = \sum_i n_i$ is again the total number of samples and $c = n - m$, the total number of grouped observations. Typically $n_i = 1$ for most samples and occasionally $n_i \ge 2$ but one can also envisage scenarios with larger groups of the same identifier (y, id).

5.4.4. Domain adaptation, adversarial examples and adversarial domain shifts

In this work, we are interested in guarding against adversarial domain shifts. We use the causal graph to explain the related but not identical goals of domain adaptation, transfer learning and guarding against adversarial examples.

(i) **Domain adaptation and transfer learning.** Assume we have J different domains, each with a new distribution F_j for the joint distribution of (Y, Δ). The shift of F_j for different domains $j = 1, \dots, J$ causes a shift in both the distribution of X and in the conditional distribution $Y|X$. If we consider domain adaptation and transfer learning together, the goal is generally to give the best possible prediction $\hat{Y}_j(x)$ in each domain $j = 1, \dots, J$. In contrast, we do not aim to give the best possible prediction in each domain as we aim to infer a single prediction that should work as well as possible in a worst-case sense over a set of distributions generated by domain changes. Some predictive accuracy needs to be sacrificed compared to the best possible prediction in each domain.

(ii) **Standard adversarial examples.** The setting of adversarial examples in the sense of Szegedy et al. (2014) and Goodfellow et al. (2015) can also be described by the causal graph above by using $X^{\mathrm{style}}(\Delta) = \Delta$ and identifying X^{style} with pixel-by-pixel additive ef-

---

[4] Observations where the ID variable is unobserved are not grouped, that is each such observation is counted as a unique observation of (Y, ID).

fects. The magnitude of the intervention Δ is then typically assumed to be within an ϵ-ball in ℓ_q-norm around the origin, with $q = \infty$ or $q = 2$ for example. If the input dimension is large, many imperceptible changes in the coordinates of X can cause a large change in the output, leading to a misclassification of the sample. The goal is to devise a classification in this graph that minimizes the adversarial loss

$$E\Big[\max_{\Delta\in\mathbb{R}^q:\ \|\Delta\|_q\le\epsilon} \ell\Big(Y, f_\theta\big(X(\Delta)\big)\Big)\Big], \tag{5.5}$$

where $X(\Delta)$ is the image under the intervention Δ and $\hat{Y} = f_\theta(X(\Delta))$ is the estimated conditional distribution of Y, given the image under the chosen interventions. See Sinha et al. (2018) for recent work on achieving robustness to a pre-defined class of distributions.

(iii) **Adversarial domain shifts.** Here we are interested in arbitrarily strong interventions $\Delta \in \mathbb{R}^q$ on the style features X^{style}, which are not known explicitly in general. Analogously to (5.5), the adversarial loss under arbitrarily large style interventions is

$$L_{adv}(\theta) = E\Big[\max_{\Delta\in\mathbb{R}^q} \ell\Big(Y, f_\theta\big(X(\Delta)\big)\Big)\Big]. \tag{5.6}$$

In contrast to (5.5) the interventions can be arbitrarily strong but we assume that the style features X^{style} can only change certain aspects of the image, while other aspects of the image (mediated by the core features) cannot be changed. In contrast to Ganin et al., 2016, we use the term “adversarial” to refer to adversarial interventions on the style features, while the notion of “adversarial” in domain adversarial neural networks describes the training procedure. Nevertheless, the motivation of Ganin et al., 2016 is equivalent to ours—that is, to protect against shifts in the distribution(s) of test data which we characterize by distinguishing between core and style features. We also look at random interventions Δ. Each distribution of the random interventions is inducing a distribution for (X, Y). Let $\mathcal{F}$ be the set of all such induced distributions. We then try to minimize the worst-case across this distribution class, as in (5.1), with the difference to standard distributional robustness being that the set $\mathcal{F}$ takes a specific form induced by the causal graph.

The adversarial loss $L_{adv}(\theta)$ of the pooled estimator (5.3) will in general be infinite; see §5.6.1 for a concrete example. Using panel (b) in Figure 5.2, one can show that the pooled estimator will work well in terms of the adversarial loss L_{adv} if both (i) $Y \perp\!\!\!\perp X|X^{\text{core}}$ and (ii) $Y \not\perp\!\!\!\perp X^{\text{core}}|X^{\text{style}}$. The first condition (i) implies that if the estimator learns to extract X^{core} from the image X, there is no further information in X that explains Y and, therefore, the direction corresponding to X^{style} is not required for predicting Y. The second condition (ii) prevents that the relations between Y, X^{core}, and X^{style} are deterministic and ensures that X^{style} cannot replace X^{core} in the first condition. From (i) and (ii), we see that the pooled estimator will work well in terms of the adversarial loss L_{adv} if (a) the edge from X^{style} to X is absent or if (b) both the edge from D to X^{style} and the edge from Y to X^{style} are absent (cf. Figure 5.2).

5.5. Conditional variance regularization

5.5.1. Invariant parameter space

In order to minimize the adversarial loss (5.6) we have to ensure $f_\theta(x(\Delta))$ is as constant as possible as a function of the style variable Δ for all $x \in \mathbb{R}^p$. Let I be the *invariant parameter space*

$$I := \{\theta : f_\theta(x(\Delta)) \text{ is constant as function of } \Delta \text{ for all } x \in \mathbb{R}^p\}.$$

For all $\theta \in I$, the adversarial loss (5.6) is identical to the loss under no interventions at all. More precisely, let X be a shorthand notation for $X(\Delta = 0)$, the images in absence of external interventions:

$$\text{if } \theta \in I, \text{ then} \qquad E\Big[\max_{\Delta \in \mathbb{R}^q} \ell\big(Y, f_\theta\big(X(\Delta)\big)\big)\Big] \;=\; E\Big[\ell\big(Y, f_\theta\big(X\big)\big)\Big].$$

The optimal predictor in the invariant space I is

$$\theta^* \;=\; \text{argmin}_\theta \; E\Big[\ell(Y, f_\theta(X))\Big] \text{ such that } \theta \in I. \tag{5.7}$$

If f_θ is only a function of the core features X^{core}, then $\theta \in I$. The challenge is that the core features are not directly observable and we have to infer the invariant space I from data.

5.5.2. CoRe estimator

To get an approximation to the optimal invariant parameter vector (5.7), we use empirical risk minimization under an invariance constraint:

$$\hat{\theta}^{core} = \operatorname{argmin}_\theta \frac{1}{n}\sum_{i=1}^n \ell\big(y_i, f_\theta(x_i)\big) \text{ such that } \theta \in I_n, \tag{5.8}$$

where the first part is the empirical version of the expectation in (5.7). The unknown invariant parameter space I is approximated by an empirically invariant space I_n. For all structural equation models of the form (5.4), the invariant space I is constrained by the space of models that have vanishing expected conditional variance in the sense that

$$I \subseteq \{\theta : C_\theta = 0\}, \quad \text{where } C_\theta := E(\mathrm{Var}(f_\theta(X)|Y, \mathrm{ID}))$$

is the expected conditional variance of $f_\theta(X)$, given (Y, ID). As empirical approximation $I_n = I_n(\tau)$ we use

$$I_n(\tau) := \big\{\theta : \hat{C}_\theta \le \tau\big\}, \quad \text{where } \hat{C}_\theta := \hat{E}(\widehat{\mathrm{Var}}(f_\theta(X)|Y, \mathrm{ID})) \tag{5.9}$$

is an estimate of the expected variance (details below). Setting $\tau = 0$ is equivalent to demanding that the conditional variance vanishes which implies that the estimated predictions for the class labels are identical across all images that share the same identifier (y, id) while slightly larger values of τ allow for some small degree of variations. Under the right assumptions we get $I_n(\tau) \to I$ for $n \to \infty$ and $\tau \to 0$. We return to this question in §5.6.1. One can equally use the Lagrangian form of the constrained optimization in (5.8), with a penalty parameter λ instead of a constraint τ, to get

$$\hat{\theta}^{core} = \operatorname{argmin}_\theta \frac{1}{n}\sum_{i=1}^n \ell\big(y_i, f_\theta(x_i)\big) + \lambda \cdot \hat{C}_\theta. \tag{5.10}$$

We will give an explicit interpretation of this conditional variance penalty λ in §5.6.2. We can also add a standard ridge penalty in addition to the conditional variance penalty.

Before showing numerical examples, we first discuss the estimation of the expected conditional variance in §5.5.3, before returning to the simple examples of §5.2 in §5.5.4. Adversarial risk consistency in a classification setting for a partially linear version of (5.4) is shown in §5.6.1. Furthermore, we discuss the population limit of the penalized version in §5.6.2, where we show that the regularization parameter $\lambda \geq 0$ is proportional to the size of the future style interventions (or rather proportional to the magnitude of the noise on the style variables) that we want to guard against for future test data.

5.5.3. Estimating expected conditional variance as a graph Laplacian

Recall that $S_j \subseteq \{1, \ldots, n\}$ contains samples with identical realizations of (Y, ID) for $j \in \{1, \ldots, m\}$. For each $j \in \{1, \ldots, m\}$, define the arithmetic mean across all $f_\theta(x_i), i \in S_j$ as $\mu_{\theta,j}$. As estimator of the conditional variance $\hat{C}_\theta$ we use

$$\hat{C}_\theta := \frac{1}{m} \sum_{j=1}^{m} \frac{1}{|S_j|} \sum_{i \in S_j} (f_\theta(x_i) - \mu_{\theta,j})^2,$$

where the right hand side can also be interpreted as the graph Laplacian (Belkin et al., 2006) of an appropriately weighted graph that fully connects all observations $i \in S_j$ for each $j \in \{1, \ldots, m\}$. If there are no groups of samples that share the same identifier (y, id), the graph Laplacian is zero and we also define $\hat{C}_\theta$ to vanish in this case. The CoRe estimator is then identical to pooled estimation in this special case.

As an alternative to penalizing with the expected conditional variance of the predicted response, we can constrain I by looking at the expected conditional variance of the loss

$$I \subseteq \{\theta : C_\theta^\ell = 0\}, \quad \text{where } C_\theta^\ell = E(\mathrm{Var}(\ell(Y, f_\theta(X))|Y, \mathrm{ID}))$$

and get an empirical estimate as

$$I_n^\ell(\tau) := \left\{\theta : \hat{C}_\theta^\ell \leq \tau\right\}, \quad \text{where } \hat{C}_\theta^\ell = \hat{E}(\widehat{\mathrm{Var}}(\ell(Y, f_\theta(X))|Y, \mathrm{ID})). \quad (5.11)$$

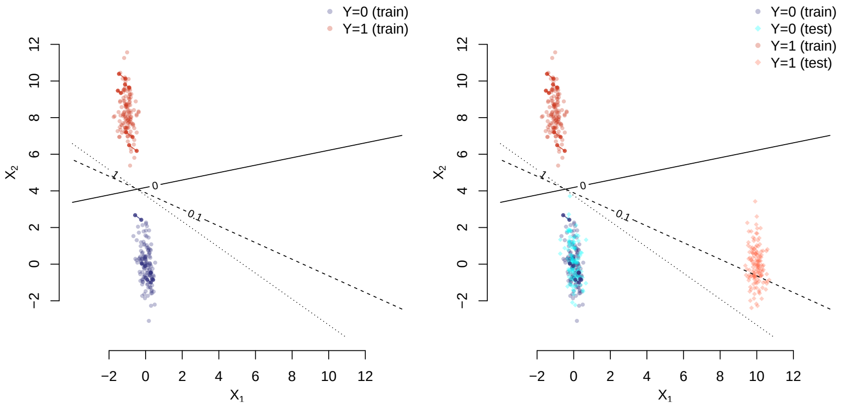
The penalty is then taking a similar form to Namkoong and Duchi (2017). A crucial difference of our approach is that we penalize with the expected *conditional* variance. That we take a conditional variance is here important as we try to achieve distributional robustness with respect to interventions on the style variables. Conditioning on ID allows to guard specifically against these interventions. An unconditional variance penalty, in contrast, can achieve robustness against a pre-defined class of distributions such as a ball of distributions defined in a Kullback-Leibler or Wasserstein metric; see for example Sinha et al., 2018 for an application in the context of adversarial examples. Some further discussion is in §5.6.2. If not mentioned otherwise we use the conditional variance of the predictions as in (5.9) as a conditional variance penalty.

5.5.4. Classification example

We revisit the first and the second example from §5.2. Figure 5.3 shows subsamples of the respective training and test sets with the estimated decision boundaries for different values of the penalty parameter λ; in both examples, $n = 20000$ and $c = 500$. Additionally, grouped examples that share the same (y, id) are visualized: two grouped observations are connected by a line or curve, respectively. In each example, there are ten such groups visualized (better visible in the nonlinear example). Panel (a) shows the linear decision boundaries for $\lambda = 0$, equivalent to the pooled estimator, and for CoRe with $\lambda \in \{.1, 1\}$. The pooled estimator misclassifies all test points of class 1 as can be seen in panel (b). In contrast, the decision boundary of the CoRe estimator with $\lambda = 1$ aligns with the direction along which the grouped observations vary, classifying the test set with almost perfect accuracy. Panels (c) and (d) show the corresponding plots for the second example for penalty values $\lambda \in \{0, 0.05, 0.1, 1\}$. While all of them yield good performance on the training set, only a value of $\lambda = 1$, which is associated with a circular decision boundary, achieves almost perfect accuracy on the test set.

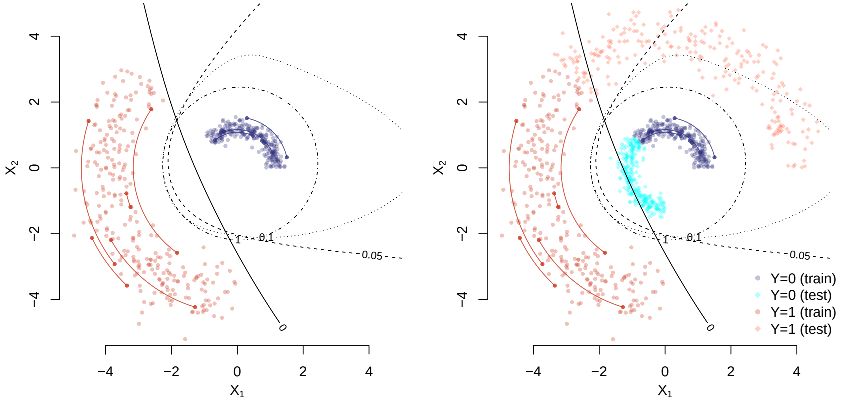
5.6. Adversarial risk consistency and distributional robustness

We show two properties of the CoRe estimator. First, adversarial risk consistency is shown for logistic models. Second, we show that the population CoRe estimator protects optimally against an increase in the variance of the noise in the style variable in a regression setting.



(a) Example 1, training set.

(b) Example 1, test set.



(c) Example 2, training set.

(d) Example 2, test set.

Figure 5.3.: The decision boundary as function of the penalty parameters λ for the examples 1 and 2 from Figure 5.1. There are ten pairs of samples visualized that share the same identifier (y, id) and these are connected by a line resp. a curve in the figures (better visible in panels (c) and (d)). The decision boundary associated with a solid line corresponds to $\lambda = 0$, the standard pooled estimator that ignores the groupings. The broken lines are decision boundaries for increasingly strong penalties, taking into account the groupings in the data. Here, we only show a subsample of the data to avoid overplotting.

5.6.1. Adversarial risk consistency for classification and logistic loss

We analyze the adversarial loss, defined in Eq. (5.6), for the pooled and the CoRe estimator in a one-layer network for binary classification (logistic regression). The proof is given in §5.A.

Assume the structural equation for the image $X \in \mathbb{R}^p$ is linear in the style features $X^{\text{style}} \in \mathbb{R}^q$ (with generally $p \gg q$) and we use logistic regression to predict the class label $Y \in \{-1, 1\}$. Let the interventions $\Delta \in \mathbb{R}^q$ act additively on the style features X^{style} (this is only for notational convenience) and let the style features X^{style} act in a linear way on the image X via a matrix $W \in \mathbb{R}^{p \times q}$ (this is an important assumption without which results are more involved). The core or 'conditionally invariant' features are $X^{\text{core}} \in \mathbb{R}^r$, where in general $r \leq p$ but this is not important for the following. For independent $\varepsilon_Y, \varepsilon_{\text{ID}}, \varepsilon_{\text{style}}, \varepsilon_X$ in $\mathbb{R}, \mathbb{R}, \mathbb{R}^q, \mathbb{R}^p$ respectively with positive density on their support and continuously differentiable functions $k_y, k_{\text{id}}, k_{\text{style}}, k_{\text{core}}, k_x$,

$$\begin{aligned} \text{class } Y &\leftarrow k_y(D, \varepsilon_Y) \\ \text{identifier ID} &\leftarrow k_{\text{id}}(Y, \varepsilon_{\text{ID}}) \\ \text{core or conditionally invariant features } X^{\text{core}} &\leftarrow k_{\text{core}}(Y, \text{ID}) \\ \text{style or orthogonal features } X^{\text{style}} &\leftarrow k_{\text{style}}(Y, \text{ID}, \varepsilon_{\text{style}}) + \Delta \\ \text{image } X &\leftarrow k_x(X^{\text{core}}, \varepsilon_X) + W X^{\text{style}}. \end{aligned} \tag{5.12}$$

Of these, Y, X and ID are observed whereas $D, X^{\text{core}}, X^{\text{style}}, \Delta$ and the noise variables are latent. The distribution of Δ can depend on the unobserved domain.

We assume a logistic regression as a prediction of Y from the image data X:

$$f_\theta(x) := \frac{\exp(x^t \theta)}{1 + \exp(x^t \theta)}.$$

Given training data with n samples, we estimate θ with $\hat{\theta}$ and use here a logistic loss $\ell_\theta(y_i, x_i) = \log(1 + \exp(-y_i(x_i^t \theta)))$ for training and testing.

We want to compare the following losses on test data

$$L(\theta) = E\Big[\ell\big(Y, f_\theta(X)\big)\Big]$$
$$L_{adv}(\theta) = E\Big[\max_{\Delta \in \mathbb{R}^q} \ell\big(Y, f_\theta(X(\Delta))\big)\Big],$$

where the X in the first loss is a shorthand notation for $X(\Delta = 0)$, that is the images in absence of interventions on the style variables. The first loss is thus a standard logistic loss in absence of adversarial interventions. The second loss is the loss under adversarial style or domain interventions as we allow arbitrarily large interventions on X^{style} here. The corresponding benchmarks are

$$L^* = \min_\theta L(\theta), \text{ and } L^*_{adv} = \min_\theta L_{adv}(\theta).$$

The formulation of Theorem 5.2 relies on the following assumptions.

Assumption 5.1 *We require the following conditions:*

- *(A1) Assume Δ is sampled from a distribution for training data in $\mathbb{R}^q$ with positive density (with respect to the Lebesgue measure) in an ϵ-ball in ℓ_2-norm around the origin for some $\epsilon > 0$.*
- *(A2) Assume the matrix W has full rank q.*
- *(A3) For a fixed number n of samples, the samples of (Y, ID, X) are drawn iid from a distribution such that the number $m \leq n$ of unique realizations of (Y, ID) is smaller than $n - q$ with probability p_n and $p_n \to 1$ for $n \to \infty$.*

The last assumption guarantees that the number $c = n - m$ of grouped examples is at least as large as the dimension of the style variables. If we have too few or no grouped examples (small c), we cannot estimate the conditional variance accurately. Under these assumptions we can prove adversarial risk consistency.

Theorem 5.2 (Adversarial risk consistency) *Under model* (5.12) *and Assumption 5.1, with probability 1 with respect to the training data, the pooled estimator* (5.3) *has infinite adversarial loss*

$$L_{adv}(\hat{\theta}^{pool}) = \infty.$$

The CoRe *estimator* (5.8) *with* $\tau = 0$ *in* (5.9) *is adversarial loss consistent, in the sense that for* $n \to \infty$,

$$L_{adv}(\hat{\theta}^{core}) \to_p L^*_{adv}.$$

A proof is given in §5.A. The respective ridge penalties in both estimators (5.3) and (5.10) are assumed to be zero for the proof, but the proof can easily be generalized to include ridge penalties that vanish sufficiently fast for large sample sizes. The Lagrangian regularizer λ is assumed to be infinite for the CoRe estimator. Again, this could be generalized to finite values if the adversarial interventions Δ are constrained to be in a region with finite ℓ_2-norm. An equivalent result can be derived for misclassification loss instead of logistic loss, where the adversarial misclassification error of the pooled estimator is then 1 while the adversarial misclassification error of the CoRe estimator will converge to the optimal adversarial value.

5.6.2. Population limit: optimal robustness against increases of the style-noise variance

We look at a partially linear version of the causal graph and least squares loss as a special case, using the marginalized version of the causal graph as in panel (c) of Figure 5.2. Let $Y \in \mathbb{R}$ be a continuous target variable, $\mathrm{ID} \in \mathbb{Z}$ an integer-valued identity variable, and $X^{\text{style}} \in \mathbb{R}^r$ the style or orthogonal features and the observed vector $X \in \mathbb{R}^p$. Let $\varepsilon_Y, \varepsilon_{\text{ID}}, \varepsilon_{\text{style}}$ be independent mean-zero random vectors in $\mathbb{R}, \mathbb{R}, \mathbb{R}^r$ respectively with positive density on their respective support, variance σ_Y^2 for ε_Y and non-singular covariance Σ_{style} for $\varepsilon_{\text{style}}$[5]. We look at the population limit of the CoRe estimator in its penalized form (5.10)

$$\theta^{core}(\lambda) \;=\; \mathrm{argmin}_{b \in \mathbb{R}^p}\, E\big[\ell\big(Y, b^t X\big)\big] + \lambda \cdot C_\theta, \tag{5.13}$$

where again $C_\theta := E(\mathrm{Var}(f_\theta(X)|Y, \mathrm{ID}))$ is the expected conditional variance and $\ell(y, z) = (y - z)^2$. We analyze the case where interventions Δ are random and follow the same distribution as the noise $\varepsilon_{\text{style}}$, just with a different scaling that can depend on the domain. Specifically, as a special case of the marginalized version of the causal graph in panel (c) of Figure 5.2, consider a partially linear version of (5.4) with a constant marginal

---

[5] We can also add an independent noise term ε_X for X but choose to omit it here to retain notational simplicity.

distribution of Y in all domains

$$\begin{aligned} Y &\leftarrow \varepsilon_Y \in \mathbb{R} \\ \text{ID} &\leftarrow k_{\text{id}}(Y, \varepsilon_{\text{ID}}) \\ X^{\text{style}} &\leftarrow k_{\text{style}}(Y, \text{ID}) + \varepsilon_{\text{style}} + \kappa \cdot \varepsilon'_{\text{style}} \\ X &\leftarrow k_x(Y, \text{ID}) + BX^{\text{style}} \end{aligned} \tag{5.14}$$

for suitable functions $k_{\text{id}} : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{Z}$, $k_{\text{style}} : \mathbb{R} \times \mathbb{Z} \mapsto \mathbb{R}^r$, $k_x : \mathbb{R} \times \mathbb{Z} \times \mathbb{R}^p \mapsto \mathbb{R}^p$ and matrix $B \in \mathbb{R}^{p \times r}$. As mentioned above, the interventions Δ are modeled as random interventions $\kappa\varepsilon'_{\text{style}}$, where $\varepsilon'_{\text{style}}$ has the same distribution as $\varepsilon_{\text{style}}$ but the two random variables are independent. The scaling $\kappa \geq 1$ is variable. In a standard setting, we might have $\kappa = 0$ for training data but we suppose that κ can increase in the future. In a new domain, for example, it might be larger. We would like to have a prediction of Y that works well even if the scaling κ of the 'style noise' increases substantially. Let E_κ denote the expectation under model (5.14) with parameter $\kappa \in \mathbb{R}^+$.

Theorem 5.3 (Distributional robustness) *Under model* (5.14)*, the population* CoRe *estimator* (5.13) *is optimal against the class of distributions generated by varying the style noise level* κ *in* $[0, \sqrt{\lambda}]$,

$$\theta^{core}(\lambda) = argmin_b \sup_{\kappa \leq \sqrt{\lambda}} E_\kappa(\ell(Y, b^t X)).$$

A proof is given in §5.B.

The CoRe estimator hence optimizes the worst case among all noise scalings of the style variable. The value of the penalty λ determines the level up to which we are protected when the noise variance increases. More precisely, a penalty $\lambda = \kappa^2$ is mimicking an increase in the variance of the noise in the style variable and allows using the current training data (with $\kappa = 0$) to optimize the loss under arbitrarily large values of the additional style variance κ^2. In this sense, there is a clear interpretation of the penalty factor λ in the CoRe estimator (5.10). Choosing $\lambda = 0$ means that we expect the variance of the style variable to remain unchanged, whereas using a strong penalty $\lambda \to \infty$ assumes that the variance of the style variable will grow very large in the future and the performance of the estimator will then not be affected even under arbitrarily strong interventions on the style variable.

A similar result is shown in Rothenhäusler et al. (2018a) who propose "anchor regression". Anchor regression penalizes the ordinary least squares objective with a quantity that relies on so-called "anchors" A which are exogeneous variables. This quantity corresponds to the change in loss under shift interventions of a given strength. Anchor regression is shown to yield optimal predictive performance under such interventions. While the theoretical results have a similar form as the estimator is shown to be distributionally robust, they do not follow as corollaries from each other. Both CoRe and anchor regression rely on the presence of "auxiliary variables"—in CoRe, we exploit the grouping information given by ID while anchor regression relies on the exogeneous anchor variable A. However, ID and A play almost orthogonal roles. In anchor regression, the aim is to achieve predictive stability if the variance term explained by A is varying. In CoRe, the aim is to retain the variance term explained by ID as we expect the variance under a *constant* ID = id to grow in the future. The interventions considered in anchor regression are shift interventions and it protects against specific distributional shifts up to a given strength. In Theorem 5.3, we consider noise interventions on the latent style variable.

While Theorem 5.3 was derived for regression under squared error loss, a similar result can be obtained for classification under (truncated) squared error loss. The (truncated) quadratic loss $\ell(Y, f_\theta(X)) = (Y - f_\theta(X))^2$ is classification-calibrated (Bartlett et al., 2003) and the truncation is even unnecessary in our case. For example, if $Y \in \{0, 1\}$, let $f_\theta(x) \in [0, 1]$ be the predicted probability of $Y = 1$, given $X = x$. Taking a first-order Taylor approximation of f_θ one can derive an analogous result to Theorem 5.3, where the approximation error of the Taylor expansion hinges on the magnitude of the future interventions and hence on the penalty level of the CoRe estimator. For loss functions other than truncated squared error loss one could make a similar argument but one would have to use the conditional variance of the loss as a penalty as in (5.11). This approach would then be similar to Namkoong and Duchi (2017), with the important difference that we work with conditional variances instead of unconditional variances. Conditioning on the ID variable is crucial in our context as we do not want to protect against general shifts in distribution but specifically against shifts in the distribution of the style variable. Conditioning on ID allows us to distinguish between the conditional variance caused by the unknown style variable (which we assume will change in the future) and the conditional variance caused by the randomness of ID (which we expect to stay constant in the future). Exploring the possibility of using the conditional variance of the loss instead of the prediction for general loss functions would be interesting follow-up work.

5.7. Experiments

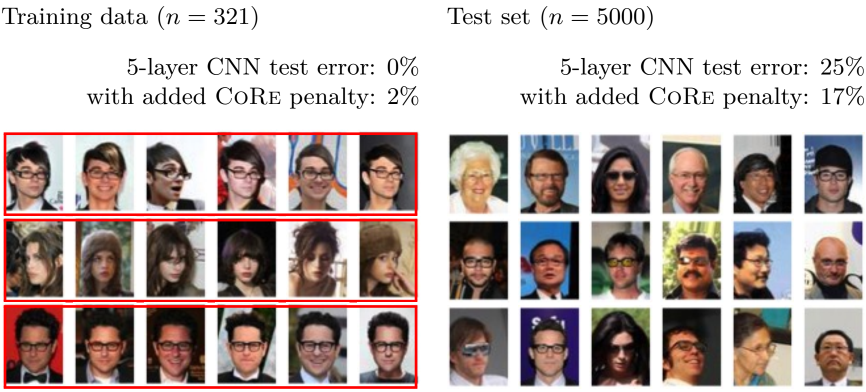
We perform an array of different experiments, showing the applicability and advantage of the conditional variance penalty for two broad settings:

1. Settings where we **do not know** what the style variables correspond to but still want to protect against a change in their distribution in the future. In the examples we show cases where the style variable ranges from brightness (§5.7.7), image quality (§5.7.2), movement (§5.7.3) and fashion (§5.7.4), which are all not known explicitly to the method. We also include genuinely unknown style variables in §5.7.1 (in the sense that they are unknown not only to the methods but also to us as we did not explicitly create the style interventions).
2. Settings where we **do know** what type of style interventions we would like to protect against. This is usually dealt with by data augmentation (adding images which are, say, rotated or shifted compared to the training data if we want to protect against rotations or translations in the test data (Schölkopf et al., 1996)). The conditional variance penalty is here exploiting that some augmented samples were generated from the same original sample and we use as ID variable the index of the original image. We show that this approach generalizes better than simply pooling the augmented data, in the sense that we need fewer augmented samples to achieve the same test error. This setting is shown in §5.7.5.

Details of the network architectures can be found in Appendix §5.C. All reported error rates are averaged over five runs of the respective method. A TensorFlow (Abadi et al., 2015) implementation of CoRe can be found at `https://github.com/christinaheinze/core`.

5.7.1. Eyeglasses detection with small sample size

We use a subsample of the CelebA dataset (Liu et al., 2015), without editing the images. We try to classify images according to whether the person in the image is wearing glasses or not. For construction of the ID variable we exploit the fact that several photos of the same person



Training data ($n = 321$)

5-layer CNN test error: 0%
with added CoRe penalty: 2%

Test set ($n = 5000$)

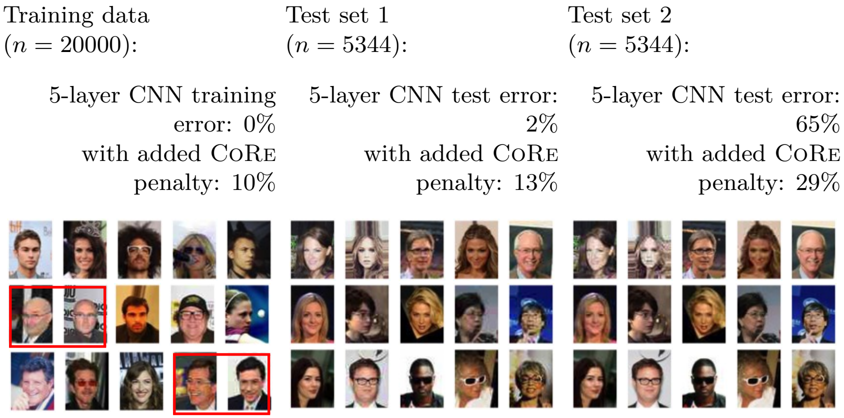
5-layer CNN test error: 25%
with added CoRe penalty: 17%

Figure 5.4.: Eyeglass detection for CelebA dataset. The goal is to predict whether a person wears glasses or not. Random samples from training and test data are shown. Groups of observations in the training data that have common (Y, ID) here correspond to pictures of the same person with either glasses on or off. These are labelled by red boxes in the training data and the conditional variance penalty is calculated across these groups of pictures.

are available and set ID to be the identifier of the person in the dataset. Figure 5.4 shows examples from both the training and the test data set. The conditional variance is estimated across groups of observations that share a common (Y, ID), which here corresponds to pictures of the same person, where all pictures show the person either with glasses (if $Y = 1$) or all pictures show the person without glasses ($Y = 0$).

The standard approach would be to pool all examples. The only additional information we exploit is that some observations can be grouped. We include $m = 10$ identities in the training set, resulting in a total sample size $n = 321$ as there are approximately 30 images of each person. If using a 5-layer convolutional neural network (details can be found in Table 5.1) and pooling all data with a standard ridge penalty, the test error on unseen images is 24.76%. Using ImageNet pre-trained features from Inception V3 does not yield lower error rates. Exploiting the group structure with the CoRe penalty (in addition to a ridge penalty) reduces the test error to 16.89%. Results are not very sensitive to the specific choice of the penalty, as discussed further in 5.D.6.

The surprising aspect here is that both training and test data are drawn from the same distribution so we would not expect a distributional shift.



Training data ($n = 20000$):

5-layer CNN training error: 0% with added CoRe penalty: 10%

Test set 1 ($n = 5344$):

5-layer CNN test error: 2% with added CoRe penalty: 13%

Test set 2 ($n = 5344$):

5-layer CNN test error: 65% with added CoRe penalty: 29%

Figure 5.5.: Eyeglass detection for CelebA dataset with image quality interventions (which are unknown to any procedure used). The JPEG compression level is lowered for $Y = 1$ (glasses) samples on training data and test set 1 and lowered for $Y = 0$ (no glasses) samples for test set 2. To the human eye, these interventions are barely visible but the CNN that uses pooled data without CoRe penalty has exploited the correlation between image quality and outcome Y to achieve a (arguably spurious) low test error of 2% on test set 1. However, if the correlation between image quality and Y breaks down, as in test set 2, the CNN that uses pooled data without a CoRe penalty has a 65% misclassification rate. The training data on the left show paired observations in two red boxes: these observations share the same label Y and show the same person ID. They are used to compute the conditional variance penalty for the CoRe estimator that does not suffer from the same degradation in performance for test set 2.

The distributional shift in this example is caused by statistical fluctuations alone (by chance the background of eyeglass wearers might, for example, be darker in the training sample than test samples, the eyeglass wearers might be more outdoors, might be more women than men etc.). The following examples are more concerned with biases that will persist even if the number of training and test samples is very large. A second difference to the subsequent examples is the grouping structure—in this example, we consider only a few identities, namely $m = 10$, with a relatively large number of associated observations ($n_i \approx 30$ for all i). In the following examples, m is much larger while n_i is typically smaller than five.

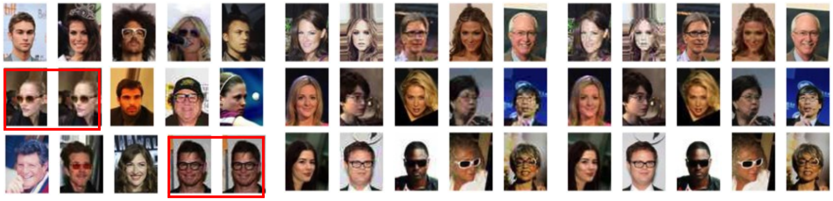
5.7.2. Eyeglasses detection with known and unknown image quality intervention

We revisit the third example from §5.2. We again use the CelebA dataset and consider the problem of classifying whether the person in the image is wearing eyeglasses. In contrast to §5.7.1 we modify the images in the following way: in the training set and in test set 1, we sample the image quality[6] for all samples $\{i : y_i = 1\}$ (all samples that show glasses) from a Gaussian distribution with mean $\mu = 30$ and standard deviation $\sigma = 10$. Samples with $y_i = 0$ (no glasses) are unmodified. In other words, if the image shows a person wearing glasses, the image quality tends to be lower. In test set 2, the quality is reduced in the same way for $y_i = 0$ samples (no glasses), while images with $y_i = 1$ are not changed. Figure 5.5 shows examples from the training set and test sets 1 and 2. This setting mimics the confounding that occurred in the Russian tank legend (cf. §5.1). For the CoRe penalty, we calculate the conditional variance across images that share the same ID if $Y = 1$, that is across images that show the same person wearing glasses on all images. Observations with $Y = 0$ (not wearing glasses) are not grouped. Two examples are shown in the red box of Figure 5.5. Here, we have $c = 5000$ grouped observations among a total sample size of $n = 20000$.

Figure 5.5 shows misclassification rates for CoRe and the pooled estimator on test sets 1 and 2. The pooled estimator (only penalized with an ℓ_2 penalty) achieves low error rates of 2% on test set 1, but suffers from a 65% misclassification error on test set 2, as now the relation between Y and the implicit X^{style} variable (image quality) has been flipped. The CoRe estimator has a larger error of 13% on test set 1 as image quality as a feature is penalized by CoRe implicitly and the signal is less strong if image quality has been removed as a dimension. However, in test set 2 the performance of the CoRe estimator is 28% and improves substantially on the 65% error of the pooled estimator. The reason is again the same: the CoRe penalty ensures that image quality is not used as a feature to the same extent as for the pooled estimator. This increases the test error slightly if the samples are generated from the same distribution as training data (as here for test set 1) but substantially improves the test error if the distribution of image quality, conditional on the class label, is changed on

---

[6] We use ImageMagick (`https://www.imagemagick.org`) to change the quality of the compression through `convert -quality q_ij input.jpg output.jpg` where $q_{i,j} \sim \mathcal{N}(30, 100)$.



Training data ($n = 20000$):

5-layer CNN training error: 0%
with added CoRe penalty: 3%

Test set 1 ($n = 5344$):

5-layer CNN test error: 2%
with added CoRe penalty: 7%

Test set 2 ($n = 5344$):

5-layer CNN test error: 65%
with added CoRe penalty: 13%

Figure 5.6.: Eyeglass detection for CelebA dataset with image quality interventions. The only difference to Figure 5.5 is in the training data where the paired images now use the same underlying image in two different JPEG compressions. The compression level is drawn from the same distribution. The CoRe penalty performs better than for the experiment in Figure 5.5 since we could explicitly control that only $X^{\text{style}} \equiv$ *image quality* varies between grouped examples. On the other hand, the performance of the pooled estimator is not changed in a noticeable way if we add augmented images as the (spurious) correlation between image quality and outcome Y still persists in the presence of the extra augmented images. Thus, the pooled estimator continues to be susceptible to image quality interventions.

test data (as here for test set 2).

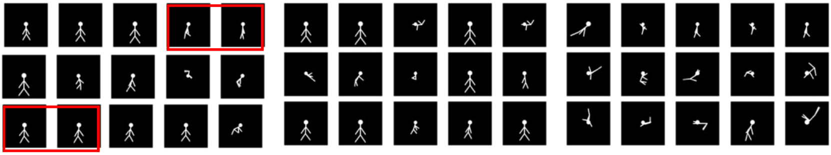
Eyeglasses detection with known image quality intervention To compare to the above results, we repeat the experiment by changing the grouped observations as follows. Above, we grouped images that had the same person ID when $Y = 1$. We refer to this scheme of grouping observations with the same (Y, ID) as 'Grouping setting 2'. Here, we use an explicit augmentation scheme and augment $c = 5000$ images with $Y = 1$ in the following way: each image is paired with a copy of itself and the image quality is adjusted as described above. In other words, the only difference between the two images is that image quality differs slightly, depending on the value that was drawn from the Gaussian distribution with mean $\mu = 30$ and standard deviation $\sigma = 10$, determining the strength of the image quality intervention. Both the original and the copy get the same value of identifier variable ID. We call this grouping scheme 'Grouping setting 1'. Compare the left panels of Figures 5.5 and 5.6 for examples.

While we used explicit changes in image quality in both above and here, we referred to grouping setting 2 as 'unknown image quality interventions' as the training sample as in the left panel of Figure 5.5 does not immediately reveal that image quality is the important style variable. In contrast, the augmented data samples (grouping setting 1) we use here differ only in their image quality for a constant (Y, ID).

Figure 5.6 shows examples and results. The pooled estimator performs more or less identical to the previous dataset. The explicit augmentation did not help as the association between image quality and whether eyeglasses are worn is not changed in the pooled data after including the augmented data samples. The misclassification error of the CoRe estimator is substantially better than the error rate of the pooled estimator. The error rate on test set 2 of 13% is also improving on the rate of 28% of the CoRe estimator in grouping setting 2. We see that using grouping setting 1 works best since we could explicitly control that only $X^{\text{style}} \equiv$ *image quality* varies between grouped examples. In grouping setting 2, different images of the same person can vary in many factors, making it more challenging to isolate image quality as the factor to be invariant against.

5.7.3. Stickmen image-based age classification with unknown movement interventions

In this example we consider synthetically generated stickmen images; see Figure 5.7 for some examples. The target of interest is $Y \in \{\text{adult}, \text{child}\}$. The core feature X^{core} is here the height of each person. The class Y is causal for height and height cannot be easily intervened on or change in different domains. Height is thus a robust predictor for differentiating between children and adults. As style feature we have here the movement of a person (distribution of angles between body, arms and legs). For the training data we created a dependence between age and the style feature 'movement', which can be thought to arise through a hidden common cause D, namely the place of observation. The data generating process is illustrated in Figure 5.17. For instance, the images of children might mostly show children playing while the images of adults typically show them in more "static" postures. The left panel of Figure 5.7 shows examples from



Training data ($n = 20000$):

5-layer CNN training error: 4%
with added CoRe penalty: 4%

Test set 1 ($n = 20000$):

5-layer CNN test error: 3%
with added CoRe penalty: 4%

Test set 2 ($n = 20000$):

5-layer CNN test error: 41%
with added CoRe penalty: 9%

Figure 5.7.: Classification into $\{\text{adult}, \text{child}\}$ based on stickmen images, where children tend to be smaller and adults taller. In training and test set 1 data, children tend to have stronger movement whereas adults tend to stand still. In test set 2 data, adults show stronger movement. The two red boxes in the panel with the training data show two out of the $c = 50$ pairs of examples over which the conditional variance is calculated. The CoRe penalty leads to a network that generalizes better for test set 2 data, where the spurious correlation between age and movement is reversed, if compared to the training data.

the training set where large movements are associated with children and small movements are associated with adults. Test set 1 follows the same distribution, as shown in the middle panel. A standard CNN will exploit this relationship between movement and the label Y of interest, whereas this is discouraged by the conditional variance penalty of CoRe. The latter is pairing images of the same person in slightly different movements as shown by the red boxes in the leftmost panel of Figure 5.7. If the learned model exploits this dependence between movement and age for predicting Y, it will fail when presented images of, say, dancing adults. The right panel of Figure 5.7 shows such examples (test set 2). The standard CNN suffers in this case from a 41% misclassification rate, as opposed to the 3% on test set 1 data. For as few as $c = 50$ paired observations, the network with an added CoRe penalty, in contrast, achieves also 4% on test set 1 data and succeeds in achieving an 9% performance on test set 2, whereas the pooled estimator fails on this dataset with a test error of 41%.

These results suggest that the learned representation of the pooled estimator uses movement as a predictor for age while CoRe does not use this feature due to the conditional variance regularization. Importantly,

Training data ($n = 17000$):	Test data 1 ($n = 4224$):	Test data 2 ($n = 1120$):
5-layer CNN training error: 0%	5-layer CNN test error: 3%	5-layer CNN test error: 44%
with added CoRe penalty: 9%	with added CoRe penalty: 8%	with added CoRe penalty: 26%
Inception v3: 2%	Inception v3: 2%	Inception v3: 42%
with added CoRe penalty: 9%	with added CoRe penalty: 8%	with added CoRe penalty: 23%

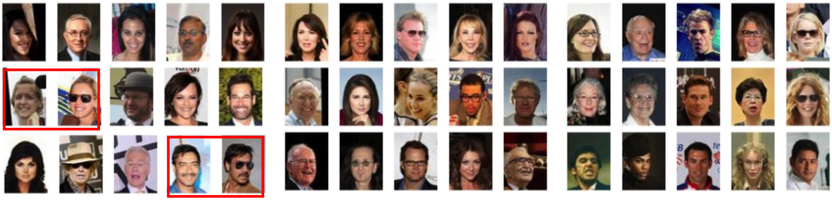


Figure 5.8.: Classification for $Y \in \{\text{woman}, \text{man}\}$. There is an unknown confounding here as men are very likely to wear glasses in training and test set 1 data, while it is women that are likely to wear glasses in test set 2. Estimators that pool all observations are making use of this confounding and hence fail for test set 2. The conditional variance penalty for the CoRe estimator is computed over groups of images of the same person (and consequently same class label), such as the images in the red box on the left. Here, $c = 500$.

including more grouped examples would not improve the performance of the pooled estimator as these would be subject to the same bias and hence also predominantly have examples of heavily moving children and "static" adults (also see Figure 5.18 which shows results for $c \in \{20, 500, 2000\}$).

5.7.4. Gender classification with unknown confounding

We work again with the CelebA dataset. This time we consider the problem of classifying whether the person in the image is male or female. We create a confounding on training and test set 1 by including mostly images of men wearing glasses and women not wearing glasses. In test set 2 the association between gender and glasses is flipped: women always wear glasses while men never wear glasses. Examples from the training and test sets 1 and 2 are shown in Figure 5.8.

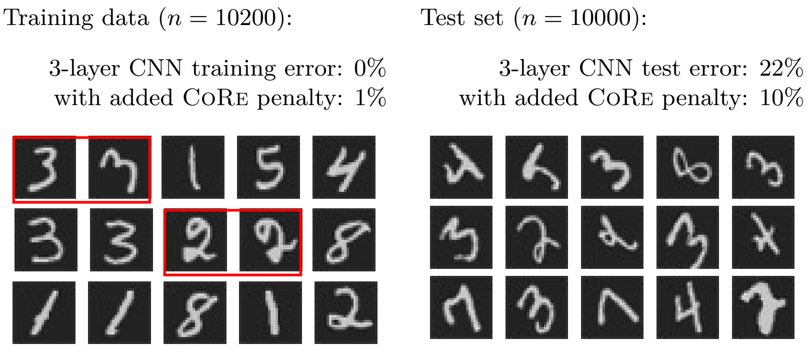

Training data ($n = 10200$):
3-layer CNN training error: 0%
with added CoRe penalty: 1%
Test set ($n = 10000$):
3-layer CNN test error: 22%
with added CoRe penalty: 10%

Figure 5.9.: Data augmentation for MNIST images. The left shows training data with a few rotated images. Evaluating on only rotated images from the test set, a standard network achieves only 22% accuracy. We can add the CoRe penalty by computing the conditional variance over images that were generated from the same original image. The test error is then lowered to 10% on the test data of rotated images.

To compute the conditional variance penalty, we use again images of the same person. The ID variable is, in other words, the identity of the person and gender Y is constant across all examples that have a constant ID. Conditioning on (Y, ID) is hence identical to conditioning on ID alone. Another difference to the other experiments is that we consider a binary style feature here.

For this example, we computed the relevant results both with a 5-layer CNN if trained end-to-end as well as for using Inception V3 pre-trained features and retraining the last softmax layer. Interestingly, the results do not change much and both models lead to misclassification error rates above 40% for test set 2 data and $c = 500$ paired examples. Adding the CoRe penalty has the desired effect in both models, as the performance is much more stable across all data sets. Additional results for different sample sizes and different numbers of paired examples can be found in Appendix §5.D.2.

5.7.5. MNIST: more sample efficient data augmentation

The goal of using CoRe in this example is to make data augmentation more efficient in terms of the required samples. In data augmentation, one creates additional samples by modifying the original inputs, e.g. by rotating, translating, or flipping the images (Schölkopf et al., 1996). In other words, additional samples are generated by interventions on style features. Using this augmented data set for training results in invariance of the estimator with respect to the transformations (style features) of interest. For CoRe we can use the grouping information that the original and the augmented samples belong to the same object. This enforces the invariance with respect to the style features more strongly compared to normal data augmentation which just pools all samples. We assess this for the style feature 'rotation' on MNIST (LeCun et al., 1998) and only include $c = 200$ augmented training examples for $m = 10000$ original samples, resulting in a total sample size of $n = 10200$. The degree of the rotations is sampled uniformly at random from $[35, 70]$. Figure 5.9 shows examples from the training set. By using CoRe the average test error on rotated examples is reduced from 22% to 10%. Very few augmented sample are thus sufficient to lead to stronger rotational invariance. The standard approach of creating augmented data and pooling all images requires, in contrast, many more samples to achieve the same effect. Additional results for $m \in \{1000, 10000\}$ and c ranging from 100 to 5000 can be found in Figure 5.16 in Appendix §5.D.3.

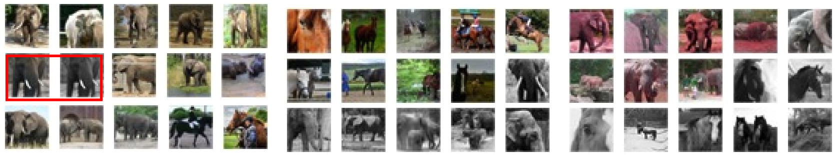
5.7.6. Elmer the Elephant

In this example, we want to assess whether invariance with respect to the style feature 'color' can be achieved. In the children's book 'Elmer the elephant'[7] one instance of a colored elephant suffices to recognize it as being an elephant, making the color 'gray' no longer an integral part of the object 'elephant'. Motivated by this process of concept formation, we would like to assess whether CoRe can exclude 'color' from its learned representation by penalizing conditional variance appropriately.

We work with the 'Animals with attributes 2' (AwA2) dataset (Xian et al., 2017) and consider classifying images of horses and elephants. We include additional examples by adding grayscale images for $c = 250$ images of elephants. These additional examples do not distinguish themselves

---

[7] `https://en.wikipedia.org/wiki/Elmer_the_Patchwork_Elephant`



Training data ($n = 1850$):

5-layer CNN training error: 0% with added CoRe penalty: 0%

Test data 1 ($n = 414$):

5-layer CNN test error: 24% with added CoRe penalty: 30%

Test data 2 ($n = 414$):

5-layer CNN test error: 52% with added CoRe penalty: 30%

Figure 5.10.: Elmer-the-Elephant dataset. The left panel shows training data with a few additional grayscale elephants. The pooled estimator learns that color is predictive for the animal class and achieves test error of 24% on test set 1 where this association is still true but suffers a misclassification error of 53% on test set 2 where this association breaks down. By adding the CoRe penalty, the test error is consistently around 30%, irrespective of the color distribution of horses and elephants.

strongly from the original training data as the elephant images are already close to grayscale images. The total training sample size is 1850.

Figure 5.10 shows examples and misclassification rates from the training set and test sets for CoRe and the pooled estimator on different test sets. Examples from these and more test sets can be found in Figure 5.21. Test set 1 contains original, colored images only. In test set 2 images of horses are in grayscale and the colorspace of elephant images is modified, effectively changing the color gray to red-brown. We observe that the pooled estimator does not perform well on test set 2 as its learned representation seems to exploit the fact that ‘gray’ is predictive for ‘elephant’ in the training set. This association is no longer valid for test set 2. In contrast, the predictive performance of CoRe is hardly affected by the changing color distributions. More details can be found in Appendix §5.D.6.

It is noteworthy that a colored elephant can be recognized as an elephant by adding a few examples of a grayscale elephant to the very lightly colored pictures of natural elephants. If we just pool over these examples, there is still a strong bias that elephants are gray. The CoRe estimator, in contrast, demands invariance of the prediction for instances of the same elephant and we can learn color invariance with a few added grayscale

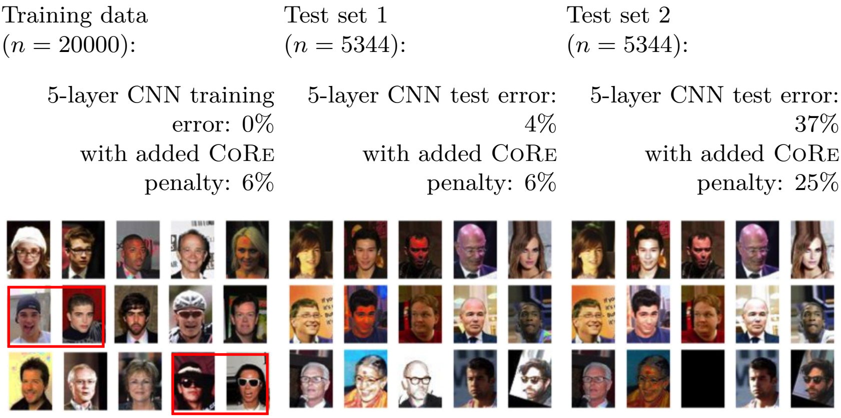

Training data
($n = 20000$):
5-layer CNN training error: 0%
with added CoRe penalty: 6%
Test set 1
($n = 5344$):
5-layer CNN test error: 4%
with added CoRe penalty: 6%
Test set 2
($n = 5344$):
5-layer CNN test error: 37%
with added CoRe penalty: 25%

Figure 5.11.: Eyeglass detection for CelebA dataset with brightness interventions (which are unknown to any procedure used). On training data and test set 1 data, images where people wear glasses tend to be brighter whereas on test set 2 images where people do not wear glasses tend to be brighter.

images.

5.7.7. Eyeglasses detection: unknown brightness intervention

As in §5.7.2 we work with the CelebA dataset and try to classify whether the person in the image is wearing eyeglasses. Here we analyze a confounded setting that could arise as follows. Say the hidden common cause D of Y and X^{style} is a binary variable and indicates whether the image was taken outdoors or indoors. If it was taken outdoors, then the person tends to wear (sun-)glasses more often and the image tends to be brighter. If the image was taken indoors, then the person tends not to wear (sun-)glasses and the image tends to be darker. In other words, the style variable X^{style} is here equivalent to brightness and the structure of the data generating process is equivalent to the one shown in Figure 5.17. Figure 5.11 shows examples from the training set and test sets. As previously, we compute the conditional variance over images of the same person, sharing the same class label (and the CoRe estimator is hence not using the knowledge that brightness is important). Two alternatives for constructing grouped observations in this setting are discussed in §5.D.1. We use $c = 2000$ and

$n = 20000$. For the brightness intervention, we sample the value for the magnitude of the brightness increase resp. decrease from an exponential distribution with mean $\beta = 20$. In the training set and test set 1, we sample the brightness value as $b_{i,j} = [100 + y_i e_{i,j}]_+$ where $e_{i,j} \sim \text{Exp}(\beta^{-1})$ and $y_i \in \{-1, 1\}$, where $y_i = 1$ indicates presence of glasses and $y_i = -1$ indicates absence.[8] For test set 2, we use instead $b_{i,j} = [100 - y_i e_{i,j}]_+$, so that the relation between brightness and glasses is flipped.

Figure 5.11 shows misclassification rates for CoRe and the pooled estimator on different test sets. Examples from all test sets can be found in Figure 5.13. First, we notice that the pooled estimator performs better than CoRe on test set 1. This can be explained by the fact that it can exploit the predictive information contained in the brightness of an image while CoRe is restricted not to do so. Second, we observe that the pooled estimator does not perform well on test set 2 as its learned representation seems to use the image's brightness as a predictor for the response which fails when the brightness distribution in the test set differs significantly from the training set. In contrast, the predictive performance of CoRe is hardly affected by the changing brightness distributions. Results for $\beta \in \{5, 10, 20\}$ and $c \in \{200, 5000\}$ can be found in Figure 5.14 in Appendix §5.D.1.

5.8. Conclusion

Distinguishing the latent features in an image into *core* and *style* features, we have proposed conditional variance regularization (CoRe) to achieve robustness with respect to arbitrarily large interventions on the style or "orthogonal" features. The main idea of the CoRe estimator is to exploit the fact that we often have instances of the same object in the training data. By demanding invariance of the classifier amongst a group of instances that relate to the same object, we can achieve invariance of the classification performance with respect to adversarial interventions on style features such as image quality, fashion type, color, or body posture. The training also works despite sampling biases in the data.

There are two main application areas:

---

[8] Specifically, we use ImageMagick (`https://www.imagemagick.org`) and modify the brightness of each image by applying the command `convert -modulate b_ij,100,100 input.jpg output.jpg` to the image.

1. If the style features are known explicitly, we can achieve the same classification performance as standard data augmentation approaches with substantially fewer augmented samples, as shown for example in §5.7.5. Additionally, the augmented images do not need to be balanced carefully for the CoRe estimator, as shown for example in §5.7.6, where adding grayscale images to a set of grayish elephants leads to invariance to color with the CoRe approach while a pooled estimator is still using color to predict the animal class with the same dataset.
2. Perhaps more interesting are settings in which it is unknown what the style features are, with examples in §5.7.1, §5.7.2, §5.7.3, §5.7.4, and §5.7.7. CoRe regularization forces predictions to be based on features that do not vary strongly between instances of the same object. We could show in the examples and in Theorems 5.2 and 5.3 that this regularization achieves distributional robustness with respect to changes in the distribution of the (unknown) style variables.

An interesting line of work would be to use larger models such as Inception or large ResNet architectures (He et al., 2016; Szegedy et al., 2015). These models have been trained to be invariant to an array of explicitly defined style features. In §5.7.4 we include results which show that using Inception V3 features does not guard against interventions on more implicit style features. We would thus like to assess what benefits CoRe can bring for training Inception-style models end-to-end, both in terms of sample efficiency and in terms of generalization performance. While we showed some examples where the necessary grouping information is available, an interesting possible future direction would be to use video data since objects display temporal constancy and the temporal information can hence be used for grouping and conditional variance regularization.

Appendix 5.A Proof of Theorem 5.2

First part. To show the first part, namely that with probability 1,

$$L_{adv}(\hat{\theta}^{pool}) = \infty,$$

we need to show that $W^t\hat{\theta}^{pool} \neq 0$ with probability 1. The reason this is sufficient is as follows: if $W^t\theta \neq 0$, then $L_{adv}(\theta) = \infty$ as we can then find a $v \in \mathbb{R}^q$ such that $\gamma := \theta^t W v \neq 0$. Setting $\Delta_\kappa = \kappa v$ for $\kappa \in \mathbb{R}$, we get $x(\Delta_\kappa)^t\theta = x(\Delta = 0)^t\theta + \kappa\gamma$. Hence $\log(1 + \exp(-y \cdot x(\Delta_\kappa)^t\theta)) \to \infty$ for either $\kappa \to \infty$ or $\kappa \to -\infty$.

To show that $W^t\hat{\theta}^{pool} \neq 0$ with probability 1, let $\hat{\theta}^*$ be the oracle estimator that is constrained to be orthogonal to the column space of W:

$$\hat{\theta}^* = \operatorname{argmin}_{\theta : W^t\theta = 0} L_n(\theta) \quad \text{with} \quad L_n(\theta) := \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f_\theta(x_i(\Delta_i))). \tag{5.15}$$

We show $W^t\hat{\theta}^{pool} \neq 0$ by contradiction. Assume hence that $W^t\hat{\theta}^{pool} = 0$. If this is indeed the case, then the constraint $W^t\theta = 0$ in (5.15) becomes non-active and we have $\hat{\theta}^{pool} = \hat{\theta}^*$. This would imply that taking the directional derivative of the training loss with respect to any $\delta \in \mathbb{R}^p$ in the column space of W should vanish at the solution $\hat{\theta}^*$. In other words, define the gradient as $g(\theta) = \nabla_\theta L_n(\theta) \in \mathbb{R}^p$. The implication is then that for all δ in the column-space of W,

$$\delta^t g(\hat{\theta}^*) = 0 \tag{5.16}$$

and we will show the latter condition is violated.

As we work with the logistic loss and $\mathcal{Y} \in \{-1, 1\}$, the loss is given by $\ell(y_i, f_\theta(x_i(\Delta_i))) = \log(1 + \exp(-y_i x_i(\Delta_i)^t\theta))$. Define $r_i(\theta) := y_i/(1 + \exp(y_i x_i(\Delta_i)^t\theta))$. For all $i = 1, \dots, n$ we have $r_i \neq 0$. Then

$$g(\hat{\theta}^*) = \frac{1}{n}\sum_{i=1}^{n} r_i(\hat{\theta}^*) x_i(\Delta_i). \tag{5.17}$$

Let $x_i(0)$ for $i = 1, \dots, n$ be training data in absence of any interventions, that is under $\Delta_i = 0$. We call these data in the following the (counterfactual) intervention-free training data. Since the interventions only have an effect on the column space of W in X, the oracle estimator $\hat{\theta}^*$ is identical under the true training data and the intervention-free training data $x(0)$. By assumption, $x_i - x_i(0) = W\Delta_i$ and (5.17) can hence also be written as

$$\delta^t g(\hat{\theta}^*) = \frac{1}{n}\sum_{i=1}^{n} r_i(\hat{\theta}^*) x_i(0)^t \delta + \frac{1}{n}\sum_{i=1}^{n} r_i(\hat{\theta}^*)\Delta_i^t W^t \delta. \tag{5.18}$$

Since δ is in the column-space of W, there exists $u \in \mathbb{R}^q$ such that $\delta = Wu$ and we can write (5.18) as

$$\delta^t g(\hat{\theta}^*) = \frac{1}{n}\sum_{i=1}^{n} r_i(\hat{\theta}^*) x_i(0)^t Wu + \frac{1}{n}\sum_{i=1}^{n} r_i(\hat{\theta}^*)\Delta_i^t W^t Wu. \tag{5.19}$$

From (A2) we have that the eigenvalues of W^tW are all positive. Also $r_i(\hat{\theta}^*)$ is not a function of the interventions Δ_i since, as above, the estimator $\hat{\theta}^*$ is identical whether trained on the original data x_i or on the intervention-free data $x_i(0)$. If we condition on everything except for the random interventions by conditioning on $(x_i(0), y_i)$ for $i = 1, \dots, n$, then the rhs of (5.19) can be written as

$$a^t u + B^t u,$$

where $a \in \mathbb{R}^q$ is fixed (again conditional on the intervention-free training data) and $B = \frac{1}{n}\sum_{i=1}^{n} r_i(\hat{\theta}^*)\Delta_i^t W^t W \in \mathbb{R}^q$ is a random vector and $B \neq -a \in \mathbb{R}^q$ with probability 1 as the interventions Δ_i are, by (A1), drawn from a continuous distribution. Hence the left hand side of (5.19) has a continuous distribution for any δ in the column-space of W, and the left hand side of (5.19) is not identically 0 with probability 1 for any given δ in the column-space of W. This shows that the implication (5.16) is incorrect with probability 1 and hence completes the proof of the first part by contradiction.

Second part. For the second part, we first show that with probability at least p_n, as defined in (A3), $\hat{\theta}^{core} = \hat{\theta}^*$ with $\hat{\theta}^*$ defined as in (5.15). Note that the invariant space for this model is the linear subspace $I = \{\theta : W^t\theta = 0\}$ and by their respective definitions,

$$\hat{\theta}^* = \text{argmin}_\theta \; \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f_\theta(x_i)) \text{ such that } \theta \in I,$$

$$\hat{\theta}^{core} = \text{argmin}_\theta \; \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f_\theta(x_i)) \text{ such that } \theta \in I_n.$$

Since we use $I_n = I_n(\tau)$ with $\tau = 0$,

$$I_n = \big\{\theta : \hat{E}(\mathrm{Var}(f_\theta(X)|Y, \mathrm{ID})) = 0\big\}.$$

This implies that for $\theta \in I_n$ it holds that $f_\theta(x_i) = f_\theta(x_{i'})$ if $i, i' \in S_j$ for some $j \in \{1, \ldots, m\}$ (recall that $(y_i, \mathrm{id}_i) = (y_{i'}, \mathrm{id}_{i'})$ if $i, i' \in S_j$ as the subsets S_j, $j = 1, \ldots, m$, collect all observations that have a unique realization of (Y, ID)). Since $f_\theta(x) = f_\theta(x')$ implies $(x - x')^t\theta = 0$, it follows that $(x_i - x_{i'})^t\theta = 0$ if $i, i' \in S_j$ for some $j \in \{1, \ldots, m\}$ and hence

$$I_n \subseteq \big\{\theta : (x_i - x_{i'})^t\theta = 0 \text{ if } i, i' \in S_j \text{ for some } j \in \{1, \ldots, m\}\big\}.$$

Since X^{style} has a linear influence on X in (5.12), $x_i - x_{i'} = W(\Delta_i - \Delta_{i'})$ if i, i' are in the same group S_j of observations for some $j \in \{1, \ldots, m\}$. Note that the number of grouped examples $n - m$ is equal to or exceeds the rank q of W with probability p_n, using (A3), and $p_n \to 1$ for $n \to \infty$. By (A2), it follows then with probability at least p_n that $I_n \subseteq \{\theta : W^t\theta = 0\} = I$. As, by definition, $I \subseteq I_n$ is always true, we have with probability p_n that $I = I_n$. Hence, with probability p_n (and $p_n \to 1$ for $n \to \infty$), $\hat{\theta}^{core} = \hat{\theta}^*$. It thus remains to be shown that

$$L_{adv}(\hat{\theta}^*) \to_p L^*_{adv}. \tag{5.20}$$

Since $\hat{\theta}^*$ is in I, we have $\ell(y, x(\Delta)) = \ell(y, x(0))$, where $x(0)$ are the previously discussed intervention-free data. Hence

$$\hat{\theta}^* = \mathrm{argmin}_\theta\ \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f_\theta(x_i(0))) \text{ such that } \theta \in I, \tag{5.21}$$

that is the estimator is unchanged if we use the data without interventions ($\Delta_i = 0$) as training data. Define the population-optimal vector as

$$\theta^* = \mathrm{argmin}_\theta\ E\big[\max_\Delta \ell(Y, f_\theta(X(\Delta)))\big] \text{ such that } \theta \in I,$$

which can for the same reason be written as

$$\theta^* = \mathrm{argmin}_\theta\ E\big[\ell(Y, f_\theta(X(\Delta = 0)))\big] \text{ such that } \theta \in I. \tag{5.22}$$

Hence (5.21) and (5.22) can be written as

$$\hat{\theta}^* = \mathrm{argmin}_{\theta:\theta\in I}\ L_n^{(0)}(\theta) \text{ where } L_n^{(0)}(\theta) := \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f_\theta(x_i(0)))$$

$$\theta^* = \mathrm{argmin}_{\theta:\theta\in I}\ L^{(0)}(\theta) \text{ where } L^{(0)}(\theta) := E[\ell(Y, f_\theta(X(\Delta = 0)))].$$

Comparing (5.21) and (5.22), by uniform convergence of $L_n^{(0)}$ to the population loss $L^{(0)}$, we have $L^{(0)}(\hat{\theta}^*) \rightarrow_p L^{(0)}(\theta^*)$. By definition of I and θ^*, we have $L_{adv}^* = L_{adv}(\theta^*) = L^{(0)}(\theta^*)$. As $\hat{\theta}^*$ is in I, we also have $L_{adv}(\hat{\theta}^*) = L^{(0)}(\hat{\theta}^*)$. Since, from above, $L^{(0)}(\hat{\theta}^*) \rightarrow_p L^{(0)}(\theta^*)$, this also implies $L_{adv}(\hat{\theta}^*) \rightarrow_p L_{adv}(\theta^*) = L_{adv}^*$. Using the previously established result that $\hat{\theta}^{core} = \hat{\theta}^*$ with probability at least p_n and $p_n \rightarrow 1$ for $n \rightarrow \infty$, this completes the proof.

Appendix 5.B Proof of Theorem 5.3

Let $\hat{Y} = \theta^t X$ be the prediction under parameter vector θ. Let E_κ be again the expectation with respect to random (Y, X) under model (5.14),

$$\begin{aligned}
Y &\leftarrow \varepsilon_Y \in \mathbb{R} \\
\mathrm{ID} &\leftarrow k_{\mathrm{id}}(Y, \varepsilon_{\mathrm{ID}}) \\
X^{\mathrm{style}} &\leftarrow k_{\mathrm{style}}(Y, \mathrm{ID}) + \varepsilon_{\mathrm{style}} + \kappa \cdot \varepsilon'_{\mathrm{style}} \\
X &\leftarrow k_x(Y, \mathrm{ID}) + B X^{\mathrm{style}}.
\end{aligned}$$

Looking at the expected squared error in a bias-variance decomposition (with the classical roles of Y and $\hat{Y}$ here reversed due to the nature of the causal graph),

$$E_\kappa\Big[(Y - \theta^t X)^2\Big] = E_\kappa\Big[(Y - \hat{Y})^2\Big] = \underbrace{E_\kappa\Big[(E_\kappa(\hat{Y}|Y) - Y)^2\Big]}_{\text{constant with respect to } \kappa} + \underbrace{E_\kappa\Big[\mathrm{Var}_\kappa(\hat{Y}|Y)\Big]}_{\text{increasing with } \kappa}. \tag{5.23}$$

The bias term in (5.23) is unaffected by a change in κ as we can write the structural equation for X for a suitable function $g(Y, \mathrm{ID}) = k_x(Y, \mathrm{ID}) + B k_{\mathrm{style}}(Y, \mathrm{ID})$ as

$$X \leftarrow g(Y, \mathrm{ID}) + B\varepsilon_{\mathrm{style}} + \kappa B \varepsilon'_{\mathrm{style}},$$

and hence, using that the expected values of $\varepsilon_{\mathrm{style}}$ and $\varepsilon'_{\mathrm{style}}$ vanish and both are here independent of Y and ID,

$$E_\kappa(\hat{Y}|Y) = E_\kappa(\theta^t X|Y) = E_\kappa(\theta^t g(Y, \mathrm{ID})|Y) = E_0(\theta^t g(Y, \mathrm{ID})|Y). \tag{5.24}$$

The variance term in (5.23) can be decomposed by the law of total variance as

$$E_\kappa\Big[\mathrm{Var}_\kappa(\hat{Y}|Y)\Big] = E_\kappa\Big[\quad \underbrace{\mathrm{Var}_\kappa(\hat{Y}|Y,\mathrm{ID})}_{\text{proportional to } (1+\kappa^2)} \quad\Big] + \mathrm{Var}_\kappa\Big[\quad \underbrace{E_\kappa(\hat{Y}|Y,\mathrm{ID})}_{\text{constant with respect to } \kappa} \quad\Big]. \tag{5.25}$$

The second term is not a function of κ, using the analogous argument as in (5.24). Using that $\varepsilon_{\text{style}}$ and $\varepsilon'_{\text{style}}$ are independent and identically distributed, the first term in (5.25) can be written as

$$\begin{aligned}
\mathrm{Var}_\kappa(\hat{Y}|Y,\mathrm{ID}) &= \mathrm{Var}_\kappa(\theta^t B\varepsilon_{\text{style}}) + \kappa^2 \mathrm{Var}_\kappa(\theta^t B\varepsilon'_{\text{style}}) \\
&= (1+\kappa^2)\mathrm{Var}_\kappa(\theta^t B\varepsilon_{\text{style}}) \\
&= (1+\kappa^2)\mathrm{Var}_{\kappa=0}(\theta^t B\varepsilon_{\text{style}}) \\
&= (1+\kappa^2)E_{\kappa=0}\big[\mathrm{Var}_{\kappa=0}(\hat{Y}|Y,\mathrm{ID})\big] \\
&= (1+\kappa^2)C_\theta.
\end{aligned}$$

The expected loss under a scaling κ of the noise is then

$$E_\kappa\Big[(Y-\theta^t X)^2\Big] = E_{\kappa=0}\Big[(Y-\theta^t X)^2\Big] + \kappa^2 \cdot C_\theta,$$

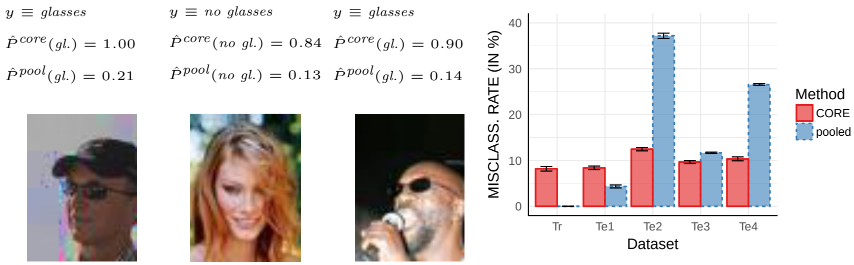
where $C_\theta = E_{\kappa=0}(\mathrm{Var}_{\kappa=0}(f_\theta(X)|Y,\mathrm{ID}))$ is the expected conditional variance under $\kappa = 0$. If we thus have training data generated under $\kappa = 0$, then the CoRe estimator with $\lambda = \kappa^2$ is optimizing the loss function under an increased style noise level, anticipating that the multiplier κ will rise potentially from the current value of 1 to higher values in different domains. In other words, the population CoRe estimator (5.13) is

$$\begin{aligned}
\theta^{core}(\lambda) &= \mathrm{argmin}_\theta\ E_{\kappa=0}\Big[(Y-\theta^t X)^2\Big] + \lambda \cdot C_\theta \\
&= \mathrm{argmin}_\theta\ E_{\kappa=\sqrt{\lambda}}\Big[(Y-\theta^t X)^2\Big] \\
&= \mathrm{argmin}_\theta\ \sup_{\kappa \le \sqrt{\lambda}} E_\kappa\Big[(Y-\theta^t X)^2\Big],
\end{aligned}$$

which completes the proof.

Appendix 5.C Network architectures

We implemented the considered models in TensorFlow (Abadi et al., 2015). The model architectures used are detailed in Table 5.1. CoRe and the



(a) Examples of misclassified observations.

(b) Misclassification rates.

Figure 5.12.: CelebA eyeglasses detection with brightness interventions, grouping setting 1. (a) Misclassified examples from the test sets. (b) Misclassification rates for $\beta = 20$ and $c = 2000$. Results for different test sets, grouping settings, $\beta \in \{5, 10, 20\}$ and $c \in \{200, 5000\}$ can be found in Figure 5.14.

pooled estimator use the same network architecture and training procedure; merely the loss function differs by the CoRe regularization term. In all experiments we use the Adam optimizer (Kingma and Ba, 2015). All experimental results are based on training the respective model five times (using the same data) to assess the variance due to the randomness in the training procedure. In each epoch of the training, the training data $x_i, i = 1, \dots, n$ are randomly shuffled, keeping the grouped observations $(x_i)_{i \in I_j}$ for $j \in \{1, \dots, m\}$ together to ensure that mini batches will contain grouped observations. In all experiments the mini batch size is set to 120. For small c this implies that not all mini batches contain grouped observations, making the optimization more challenging.

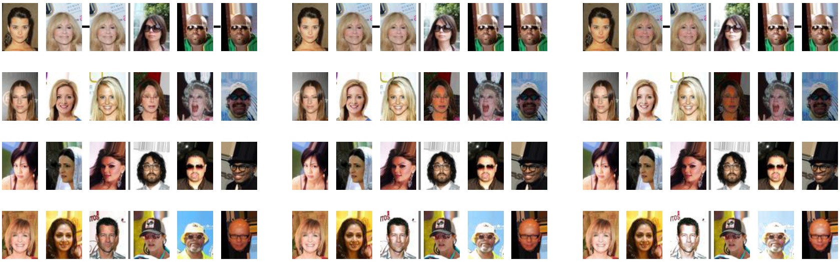
Appendix 5.D Additional experiments

5.D.1 Eyeglasses detection: known and unknown brightness interventions

Here, we show additional results for the experiment discussed in §5.7.7. Recall that we work with the CelebA dataset and consider the problem of classifying whether the person in the image is wearing eyeglasses. We discuss two alternatives for constructing different test sets and we vary the number of grouped observations in $c \in \{200, 2000, 5000\}$ as well as the strength of the brightness interventions in $\beta \in \{5, 10, 20\}$, all with sample

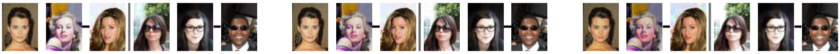
Dataset	Optimizer		Architecture
MNIST	Adam	Input	$28 \times 28 \times 1$
		CNN	Conv $5 \times 5 \times 16$, $5 \times 5 \times 32$ (same padding, strides = 2, ReLu activation), fully connected, softmax layer
Stickmen	Adam	Input	$64 \times 64 \times 1$
		CNN	Conv $5 \times 5 \times 16$, $5 \times 5 \times 32$, $5 \times 5 \times 64$, $5 \times 5 \times 128$ (same padding, strides = 2, leaky ReLu activation), fully connected, softmax layer
CelebA (all experiments using CelebA)	Adam	Input	$64 \times 48 \times 3$
		CNN	Conv $5 \times 5 \times 16$, $5 \times 5 \times 32$, $5 \times 5 \times 64$, $5 \times 5 \times 128$ (same padding, strides = 2, leaky ReLu activation), fully connected, softmax layer
AwA2	Adam	Input	$32 \times 32 \times 3$
		CNN	Conv $5 \times 5 \times 16$, $5 \times 5 \times 32$, $5 \times 5 \times 64$, $5 \times 5 \times 128$ (same padding, strides = 2, leaky ReLu activation), fully connected, softmax layer

Table 5.1.: Details of the model architectures used.



(a) Grouping setting 1, $\beta = 5$ (b) Grouping setting 1, $\beta = 10$ (c) Grouping setting 1, $\beta = 20$

(d) Grouping setting 2, $\beta = 5$ (e) Grouping setting 2, $\beta = 10$ (f) Grouping setting 2, $\beta = 20$

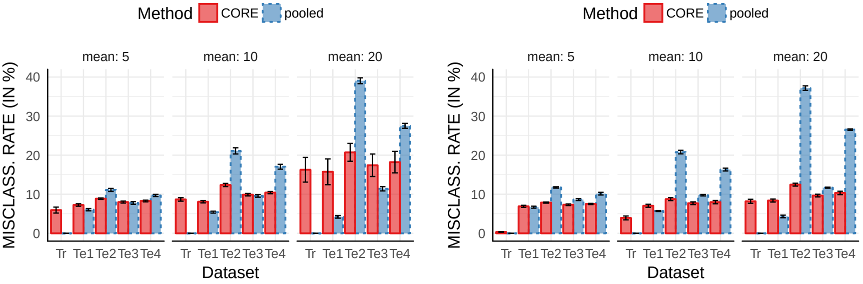


(g) Grouping setting 3, $\beta = 5$ (h) Grouping setting 3, $\beta = 10$ (i) Grouping setting 3, $\beta = 20$

Figure 5.13.: Examples from the CelebA eyeglasses detection with brightness interventions, grouping settings 1–3 with $\beta \in \{5, 10, 20\}$. In all rows, the first three images from the left have $y \equiv$ *no glasses*; the remaining three images have $y \equiv$ *glasses*. Connected images are grouped examples. In panels (a)–(c), row 1 shows examples from the training set, rows 2–4 contain examples from test sets 2–4, respectively. Panels (d)–(i) show examples from the respective training sets.

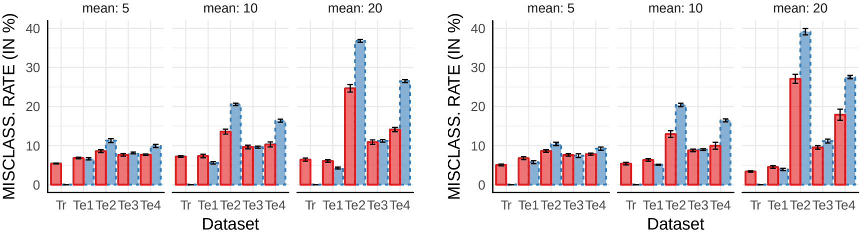
size $n = 20000$. Generation of training and test sets 1 and 2 were already described in §5.7.7. Here, we consider additionally test set 3 where all images are left unchanged (no brightness interventions at all) and in test set 4 the brightness of all images is increased.

Furthermore, we consider three different ways of grouping images. In §5.7.7 we used images of the same person to create a grouped observation by sampling a different value for the brightness intervention. We refer to this as ‘Grouping setting 2’ here. An alternative is to use the same image of the same person in different brightnesses (drawn from the same distribution) as a group over which the conditional variance is calculated. We call this ‘Grouping setting 1’ and it can be useful if we know that we want to



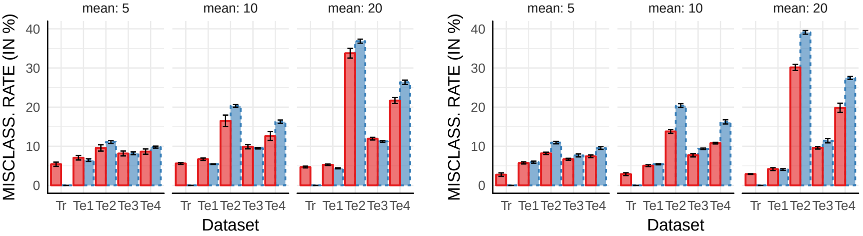
(a) Grouping setting 1, $c = 200$

(b) Grouping setting 1, $c = 2000$



(c) Grouping setting 2, $c = 2000$

(d) Grouping setting 2, $c = 5000$



(e) Grouping setting 3, $c = 2000$

(f) Grouping setting 3, $c = 5000$

Figure 5.14.: Misclassification rates for the CelebA eyeglasses detection with brightness interventions, grouping settings 1–3 with $c \in \{200, 2000, 5000\}$ and the mean of the exponential distribution $\beta \in \{5, 10, 20\}$.

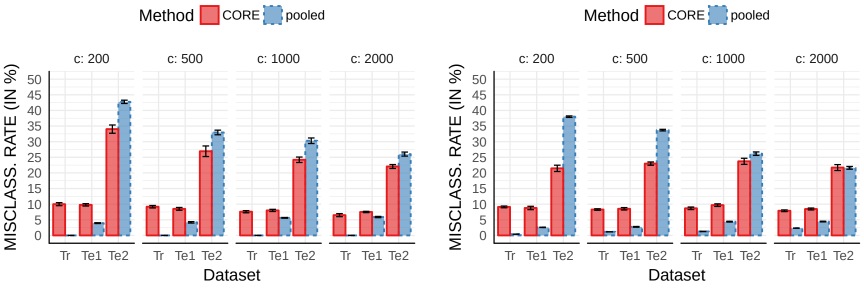
protect against brightness interventions in the future. For comparison, we also evaluate grouping with an image of a different person as a baseline (‘Grouping setting 3’). Examples from the training sets using grouping settings 1, 2 and 3 can be found in Figure 5.13.

Results for all grouping settings, $\beta \in \{5, 10, 20\}$ and $c \in \{200, 5000\}$ can be found in Figure 5.14. We see that using grouping setting 1 works best since we could explicitly control that only $X^{\text{style}} \equiv$ *brightness* varies between grouping examples. In grouping setting 2, different images of the same person can vary in many factors, making it more challenging to isolate brightness as the factor to be invariant against. Lastly, we see that if we group images of different persons (‘Grouping setting 3’), the difference between CoRe estimator and the pooled estimator becomes much smaller than in the previous settings.

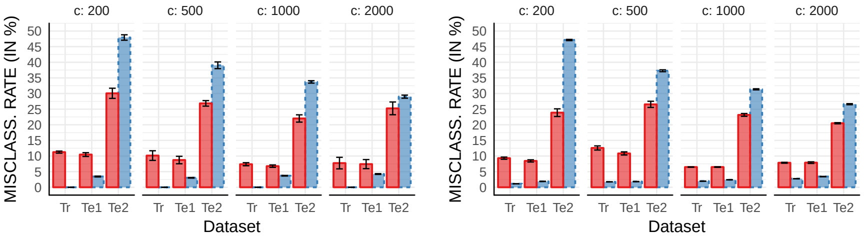
Regarding the results for grouping setting 1 in Figure 5.12, we notice that the pooled estimator performs better than CoRe on test set 1. This can be explained by the fact that it can exploit the predictive information contained in the brightness of an image while CoRe is restricted not to do so. Second, we observe that the pooled estimator does not perform well on test sets 2 and 4 as its learned representation seems to use the image’s brightness as a predictor for the response which fails when the brightness distribution in the test set differs significantly from the training set. In contrast, the predictive performance of CoRe is hardly affected by the changing brightness distributions.

5.D.2 Gender classification

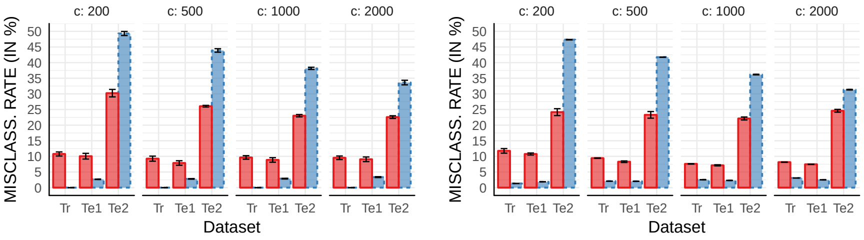
In §5.7.4 we assessed whether the results differ when (a) training a five-layer CNN (as detailed in Table 5.1) end-to-end versus (b) using Inception V3 features and merely retraining the softmax layer. Here, we show some additional results for different sample sizes and number of grouped observations. Figure 5.15 shows the results for varying numbers of n and c—in the left column for training a five-layer CNN; in the right column for using Inception V3 features. Overall, we see the same trends: As c increases, the performance difference between CoRe and the pooled estimator becomes smaller. This is due to the fact that X^{style} is binary in this example and, therefore, including grouped examples corresponds to data augmentation. Interestingly, the pooled estimator performs worse on test set 2 as n becomes larger. It thus seems to exploit X^{style} to a larger extent as n grows.



(a) $n = 5000$, 5-layer CNN (b) $n = 5000$, Inception V3

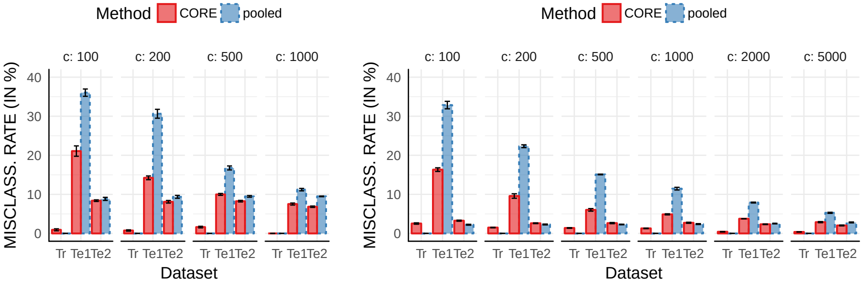


(c) $n = 10000$, 5-layer CNN (d) $n = 10000$, Inception V3



(e) $n = 17000$, 5-layer CNN (f) $n = 17000$, Inception V3

Figure 5.15.: Misclassification rates for the CelebA gender classification datasets with varying numbers for n and c. The left column shows results for training a five-layer CNN (cf. Table 5.1) end-to-end, the right column shows results for using Inception V3 features and retraining the softmax layer.



(a) $m = 1000$ (b) $m = 10000$

Figure 5.16.: Data augmentation setting: Misclassification rates for MNIST and $X^{\text{style}} \equiv$ *rotation*. In test set 1 all digits are rotated by a degree randomly sampled from $[35, 70]$. Test set 2 is the usual MNIST test set.

5.D.3 MNIST: more sample efficient data augmentation

Here, we show further results for the experiment introduced in §5.7.5. We vary the number of augmented training examples c from 100 to 5000 for $m = 10000$ and $c \in \{100, 200, 500, 1000\}$ for $m = 1000$. The degree of the rotations is sampled uniformly at random from $[35, 70]$. Figure 5.16 shows the misclassification rates. Test set 1 contains rotated digits only, test set 2 is the usual MNIST test set. We see that the misclassification rates of CoRe are always lower on test set 1, showing that it makes data augmentation more efficient. For $m = 1000$, it even turns out to be beneficial for performance on test set 2.

5.D.4 Stickmen image-based age classification

Here, we show further results for the experiment introduced in §5.7.3. Figure 5.17 illustrates the data generating process. Recall that test set 1 follows the same distribution as the training set. In test sets 2 and 3 large movements are associated with both children and adults, while the movements are heavier in test set 3 than in test set 2. Figure 5.18b shows results for different numbers of grouping examples. For $c = 20$ the misclassification rate of CoRe estimator has a large variance. For $c \in \{50, 500, 2000\}$, the CoRe estimator shows similar results. Its performance is thus not sensitive to the number of grouped examples, once there are

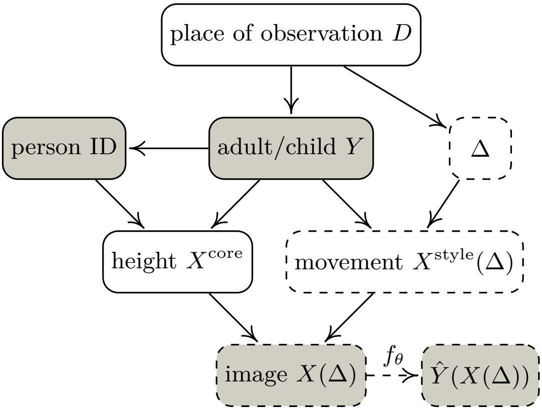
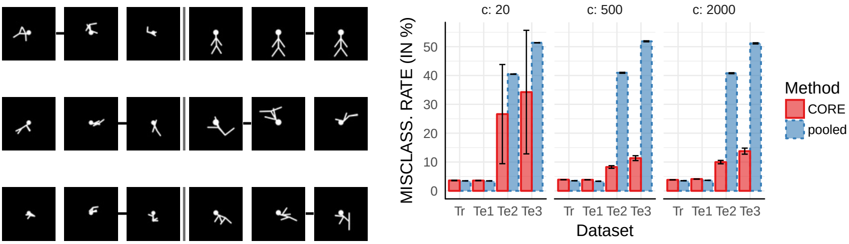


Figure 5.17.: Data generating process for the stickmen example.



(a) Examples from test sets 1–3.

(b) Misclassification rates.

Figure 5.18.: a) Examples from the stickmen test set 1 (row 1), test set 2 (row 2) and test sets 3 (row 3). In each row, the first three images from the left have $y \equiv$ *child*; the remaining three images have $y \equiv$ *adult*. Connected images are grouped examples. b) Misclassification rates for different numbers of grouped examples.

sufficiently many grouped observations in the training set. The pooled estimator fails to achieve good predictive performance on test sets 2 and 3 as it seems to use "movement" as a predictor for "age".

5.D.5 Eyeglasses detection: image quality intervention

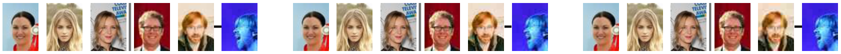
Here, we show further results for the experiments introduced in §5.7.2. Specifically, we consider interventions of different strengths by varying the mean of the quality intervention in $\mu \in \{30, 40, 50\}$. Recall that we use ImageMagick to modify the image quality. In the training set and in test set 1, we sample the image quality value as $q_{i,j} \sim \mathcal{N}(\mu, \sigma = 10)$ and apply the command `convert -quality q_ij input.jpg output.jpg` if $y_i \equiv$ *glasses*. If $y_i \equiv$ *no glasses*, the image is not modified. In test set 2, the above command is applied if $y_i \equiv$ *no glasses* while images with $y_i \equiv$ *glasses* are not changed. In test set 3 all images are left unchanged and in test set 4 the command is applied to all images, i.e. the quality of all images is reduced.

We run experiments for grouping settings 1–3 and for $c = 5000$, where the definition of the grouping settings 1–3 is identical to §5.D.1. Figure 5.19 shows examples from the respective training and test sets and Figure 5.20 shows the corresponding misclassification rates. Again, we observe that grouping setting 1 works best, followed by grouping setting 2. Interestingly, there is a large performance difference between $\mu = 40$ and $\mu = 50$ for the pooled estimator. Possibly, with $\mu = 50$ the image quality is not sufficiently predictive for the target.

(a) Grouping setting 1, $\mu = 50$

(b) Grouping setting 1, $\mu = 40$

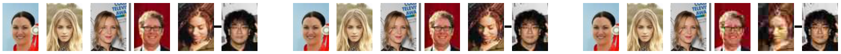
(c) Grouping setting 1, $\mu = 30$



(d) Grouping setting 2, $\mu = 50$

(e) Grouping setting 2, $\mu = 40$

(f) Grouping setting 2, $\mu = 30$

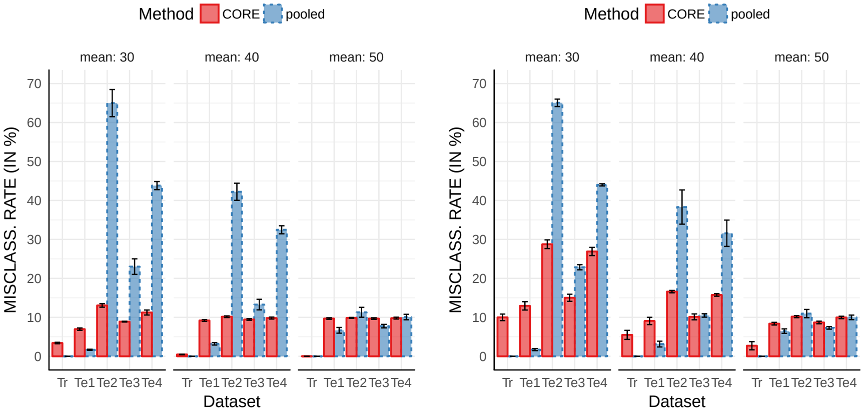


(g) Grouping setting 3, $\mu = 50$

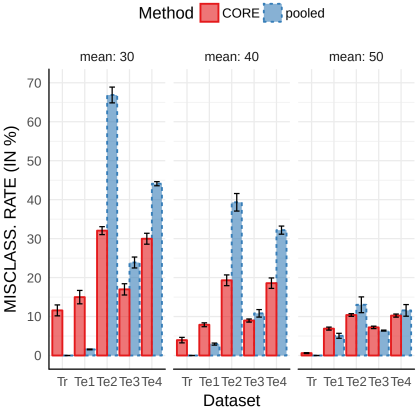
(h) Grouping setting 3, $\mu = 40$

(i) Grouping setting 3, $\mu = 30$

Figure 5.19.: Examples from the CelebA image quality datasets, grouping settings 1–3 with $\mu \in \{30, 40, 50\}$. In all rows, the first three images from the left have $y \equiv$ *no glasses*; the remaining three images have $y \equiv$ *glasses*. Connected images are grouped observations over which we calculate the conditional variance. In panels (a)–(c), row 1 shows examples from the training set, rows 2–4 contain examples from test sets 2–4, respectively. Panels (d)–(i) show examples from the respective training sets.



(a) Grouping setting 1 (b) Grouping setting 2



(c) Grouping setting 3

Figure 5.20.: Misclassification rates for the CelebA eyeglasses detection with image quality interventions, grouping settings 1–3 with $c = 5000$ and the mean of the Gaussian distribution $\mu \in \{30, 40, 50\}$.

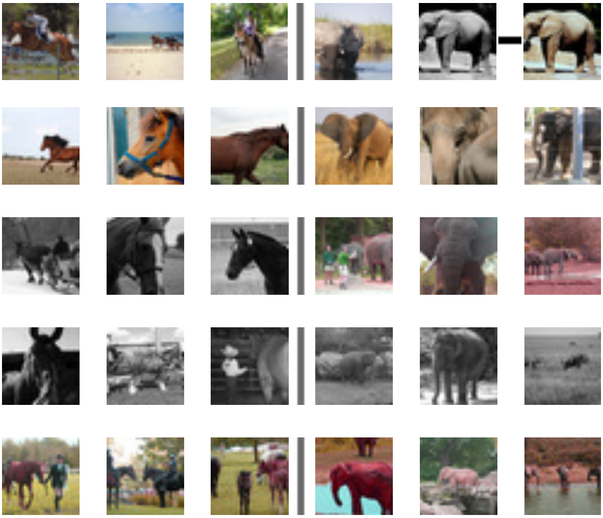


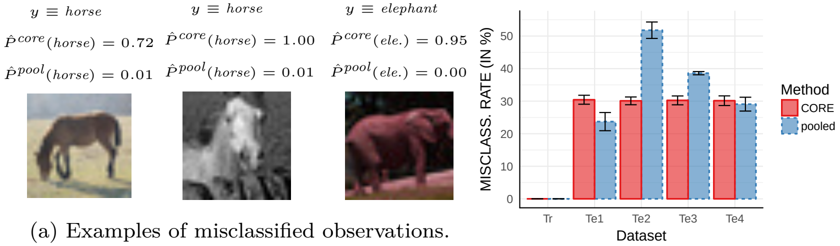
Figure 5.21.: Examples from the subsampled and augmented AwA2 dataset (Elmer-the-Elephant dataset). Row 1 shows examples from the training set, rows 2–5 show examples from test sets 1–4, respectively.

5.D.6 Elmer the Elephant

The color interventions for the experiment introduced in §5.7.6 were created as follows. In the training set, if $y_i \equiv elephant$ we apply the following ImageMagick command for the grouped examples `convert -modulate 100,0,100 input.jpg output.jpg`. Test sets 1 and 2 were already discussed in §5.7.6: in test set 1, all images are left unchanged. In test set 2, the above command is applied if $y_i \equiv$ horse. If $y_i \equiv$ elephant, we sample $c_{i,j} \sim \mathcal{N}(\mu = 20, \sigma = 1)$ and apply `convert -modulate 100,100,100-c_ij input.jpg output.jpg` to the image. Here, we consider again some more test sets than in §5.7.6. In test set 4, the latter command is applied to all images. It rotates the colors of the image, in a cyclic manner[9]. In test set 3, all images are changed to grayscale. The causal graph for the data generating process is shown in Figure 5.23. Examples from all four test sets are shown in Figure 5.21 and classification results are shown in Figure 5.22.

---

[9] For more details, see `http://www.imagemagick.org/Usage/color_mods/#color_mods`.



(a) Examples of misclassified observations.

(b) Misclassification rates.

Figure 5.22.: Elmer-the-Elephant dataset. (a) Misclassified examples from the test sets. (b) Misclassification rates on test sets 1 to 4.

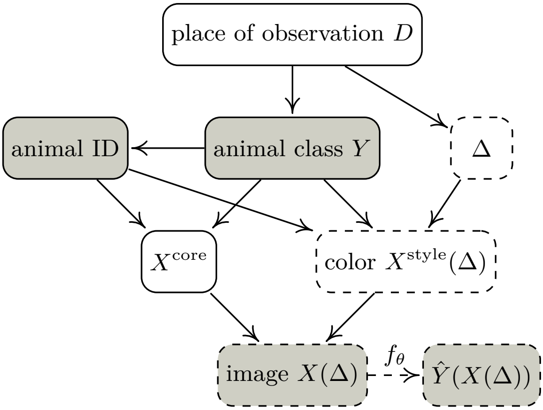
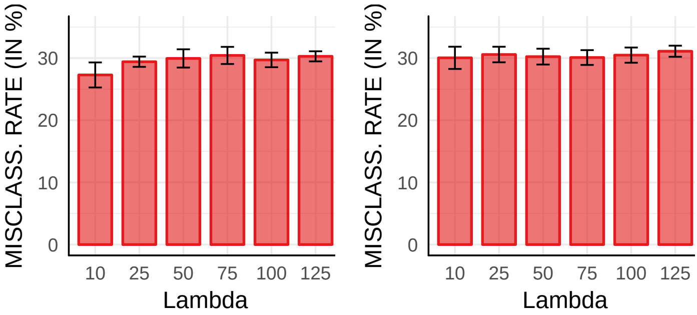
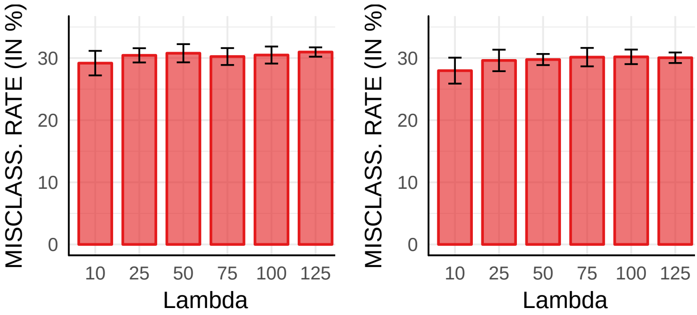


Figure 5.23.: Data generating process for the Elmer-the-Elephant example.



(a) Test set 1. (b) Test set 2.



(c) Test set 3. (d) Test set 4.

Figure 5.24.: Misclassification rates of CoRe on the subsampled and augmented AwA2 dataset (Elmer-the-Elephant dataset) as a function of the penalty λ. The outcome does not depend strongly on the chosen value.

The value of the penalty parameter λ in Eq. (5.10) is chosen depending on the expected strength of future interventions. Figure 5.24 shows the misclassification rates of CoRe on the subsampled and augmented AwA2 dataset (Elmer-the-Elephant dataset) as a function of the penalty λ. We see that performance is not very sensitive to the choice of the penalty parameter in a reasonable range.

Part III.

Distributed Estimation

Chapter 6.

DUAL-LOCO: Distributing statistical estimation using random projections

We present Dual-Loco, a communication-efficient algorithm for distributed statistical estimation. Dual-Loco assumes that the data is distributed across workers according to the features rather than the samples. It requires only a single round of communication where low-dimensional random projections are used to approximate the dependencies between features available to different workers. We show that Dual-Loco has bounded approximation error which only depends weakly on the number of workers. We compare Dual-Loco against a state-of-the-art distributed optimization method on a variety of real world datasets and show that it obtains better speedups while retaining good accuracy. In particular, Dual-Loco allows for fast cross validation as only part of the algorithm depends on the regularization parameter.

6.1. Introduction

Many statistical estimation tasks amount to solving an optimization problem of the form

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^p} J(\boldsymbol{\beta}) := \sum_{i=1}^{n} f_i(\boldsymbol{\beta}^\top \mathbf{x}_i) + \frac{\lambda}{2} \|\boldsymbol{\beta}\|^2 \tag{6.1}$$

where $\lambda > 0$ is the regularization parameter. The loss functions $f_i(\boldsymbol{\beta}^\top \mathbf{x}_i)$ depend on labels $y_i \in \mathbb{R}$ and linearly on the coefficients, $\boldsymbol{\beta}$ through a vector of covariates, $\mathbf{x}_i \in \mathbb{R}^p$. Furthermore, we assume all f_i to be convex and smooth with Lipschitz continuous gradients. Concretely, when $f_i(\boldsymbol{\beta}^\top \mathbf{x}_i) = (y_i - \boldsymbol{\beta}^\top \mathbf{x}_i)^2$, Eq. (6.1) corresponds to ridge regression; for logistic regression $f_i(\boldsymbol{\beta}^\top \mathbf{x}_i) = \log(1 + \exp(-y_i \boldsymbol{\beta}^\top \mathbf{x}_i))$.

For large-scale problems, it is no longer practical to solve even relatively simple estimation tasks such as (6.1) on a single machine. To deal with this, approaches to distributed data analysis have been proposed that take advantage of many cores or computing nodes on a cluster. A common idea which links many of these methods is stochastic optimization. Typically, each of the workers only sees a small portion of the data points and performs incremental updates to a global parameter vector. It is typically assumed that the number of data points, n, is very large compared with the number of features, p, or that the data is extremely sparse, meaning that many entries in the data matrix are zero. In such settings—which are common, but not ubiquitous in large datasets—distributed stochastic optimization algorithms perform well but may converge slowly otherwise.

A fundamentally different approach to distributed learning is for each worker to only have access to a portion of the available features. Distributing according to the features could be a preferable alternative for several reasons. Firstly, for high-dimensional data, where p is large relative to n, better scaling can be achieved. This setting is challenging, however, since most loss functions are not separable across coordinates. High-dimensional data is commonly encountered in the fields of bioinformatics, climate science and computer vision. Furthermore, for a variety of prediction tasks it is often beneficial to map input vectors into a higher dimensional feature space, e.g. using deep representation learning or considering higher-order interactions. Secondly, individual blocks of features could correspond to sensitive information (such as medical records) which should be included in the predictive model but is not allowed to be communicated in an un-disguised form due to privacy concerns.

Our contribution. In this work we introduce Dual-Loco to solve problems of the form (6.1) in the distributed setting when each worker only has access to a subset of the *features*. Dual-Loco is an extension of the Loco algorithm (Heinze et al., 2014) which was recently proposed for solving distributed ridge regression in this setting. We propose an alternative formulation where each worker instead locally solves a *dual* optimization problem. Dual-Loco has a number of practical and theoretical improvements over the original algorithm:

- DUAL-LOCO is applicable to a wider variety of smooth, convex ℓ_2 penalized loss minimization problems encompassing many widely used regression and classification loss functions, including ridge regression, logistic regression and others.
- In §6.4 we provide a more intuitive and tighter theoretical result which crucially does not depend on specific details of the ridge regression model and has weaker dependence on the number of workers, K.
- We also show that *adding* (rather than concatenating) random features allows for an efficient implementation yet retains good approximation guarantees.

In §6.5 we report experimental results with high-dimensional real world datasets corresponding to two different problem domains: climate science and computer vision. We compare DUAL-LOCO with COCOA$^+$, a recently proposed state-of-the-art algorithm for distributed dual coordinate ascent (Ma et al., 2015). Our experiments show that DUAL-LOCO demonstrates better scaling with K than COCOA$^+$ while retaining a good approximation of the optimal solution. We provide an implementation of DUAL-LOCO in Apache Spark[1]. The portability of this framework ensures that DUAL-LOCO is able to be run in a variety of distributed computing environments.

6.2. Related work

6.2.1. Distributed estimation

Recently, several asynchronous stochastic gradient descent (SGD) methods (Duchi et al., 2013a; Recht et al., 2011) have been proposed for solving problems of the form (6.1) in a parallel fashion in a multi-core, shared-memory environment and have been extended to the distributed setting. For such methods, large speedups are possible with asynchronous updates when the data is sparse. However, in some problem domains the data collected is dense with many correlated features. Furthermore, the $p \gg n$ setting can result in slow convergence. In the distributed setting, such methods can be impractical since the cost of communicating updates can dominate other computational considerations.

Jaggi et al. (2014) proposed a communication-efficient distributed dual

---

[1] `http://spark.apache.org/`

coordinate ascent algorithm (CoCoA resp. CoCoA$^+$) (Jaggi et al., 2014; Ma et al., 2015). Each worker makes multiple updates to its local dual variables before communicating the corresponding primal update. This allows for trading off communication and convergence speed. Notably they show that convergence is actually independent of the number of workers, thus CoCoA$^+$ exhibits *strong scaling* with K.

Other recent work considers solving statistical estimation tasks using a single round of communication (Liu and Ihler, 2014; Zhang et al., 2015b). However, all of these methods consider only distributing over the rows of the data where an i.i.d. assumption on the observations holds.

On the other hand, few approaches have considered distributing across the columns (features) of the data. This is a more challenging task for both estimation and optimization since the columns are typically assumed to have arbitrary dependencies and most commonly used loss functions are not separable over the features. Recently, Loco was proposed to solve ridge regression when the data is distributed across the features (Heinze et al., 2014). Loco requires a single round to communicate small matrices of randomly projected features which approximate the dependencies in the rest of the dataset (cf. Figure 6.1). Each worker then optimizes its own sub-problem independently and finally sends its portion of the solution vector back to the master where they are combined. Loco makes no assumptions about the correlation structure between features. It is therefore able to perform well in challenging settings where the features are correlated between blocks and is particularly suited when $p \gg n$. Indeed, since the relative dimensionality of local problems decreases when splitting by columns, they are easier in a statistical sense. Loco makes no assumptions about data sparsity so it is also able to obtain speedups when the data is dense.

One-shot communication schemes are beneficial as the cost of communication consists of a fixed cost and a cost that is proportional to the size of the message. Therefore, it is generally cheaper to communicate a few large objects than many small objects.

6.2.2. Random projections for estimation and optimization

Random projections are low-dimensional embeddings $\mathbf{\Pi} : \mathbb{R}^{\tau} \to \mathbb{R}^{\tau_{subs}}$ which approximately preserve an entire subspace of vectors where τ_{subs} denotes the projection dimension. They have been extensively used to

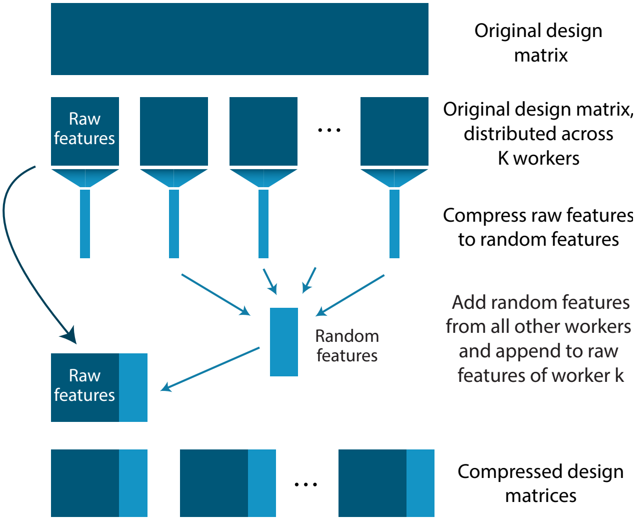

Original design matrix
Raw features
Original design matrix, distributed across K workers
Compress raw features to random features
Add random features from all other workers and append to raw features of worker k
Random features
Raw features
Compressed design matrices

Figure 6.1.: Schematic for the distributed approximation of a large data set with random projections, used by DUAL-LOCO.

construct efficient algorithms when the sample-size is large in a variety of domains such as: nearest neighbours (Ailon and Chazelle, 2009), matrix factorization (Boutsidis and Gittens, 2012), least squares (Dhillon et al., 2013; McWilliams et al., 2014) and recently in the context of optimization (Pilanci and Wainwright, 2017).

We concentrate on the Subsampled Randomized Hadamard Transform (SRHT), a structured random projection (Tropp, 2011). The SRHT consists of a projection matrix, $\mathbf{\Pi} = \sqrt{\tau/\tau_{subs}}\mathbf{DHS}$ (Boutsidis and Gittens, 2012) with $\mathbf{\Pi} \in \mathbb{R}^{\tau \times \tau_{subs}}$ and the definitions:

(i) $\mathbf{S} \in \mathbb{R}^{\tau \times \tau_{subs}}$ is a randomly chosen subsampling matrix,

(ii) $\mathbf{D} \in \mathbb{R}^{\tau \times \tau}$ is a diagonal matrix whose entries are drawn independently from $\{-1, 1\}$,

(iii) $\mathbf{H} \in \mathbb{R}^{\tau \times \tau}$ is a normalized Walsh-Hadamard matrix.

The key benefit of the SRHT is that due to its recursive definition the product between $\mathbf{\Pi}^\top$ and $\mathbf{u} \in \mathbb{R}^{\tau}$ can be computed in $O(\tau \log \tau)$ time while never constructing $\mathbf{\Pi}$ explicitly.

For moderately sized problems, random projections have been used to reduce the dimensionality of the data prior to performing regression (Kabán, 2014; Lu et al., 2013). However after projection, the solution vector is in the compressed space and so interpretability of coefficients is lost. Furthermore, the projection of the low-dimensional solution back to the original high-dimensional space—obtained by multiplying $\mathbf{\Pi}$ with the solution vector—is in fact guaranteed to be a *bad* approximation of the optimum (Zhang et al., 2012).

Dual Random Projections. Recently, Zhang et al. (2014, 2012) studied the effect of random projections on the *dual* optimization problem. For the primal problem in Eq. (6.1), defining $\mathbf{K} = \mathbf{X}\mathbf{X}^\top$, we have the corresponding dual

$$\max_{\boldsymbol{\alpha} \in \mathbb{R}^n} - \sum_{i=1}^{n} f_i^*(\alpha_i) - \frac{1}{2n\lambda} \boldsymbol{\alpha}^\top \mathbf{K} \boldsymbol{\alpha} \tag{6.2}$$

where f^* is the conjugate Fenchel dual of f and $\lambda > 0$. For example, for squared loss functions $f_i(u) = \frac{1}{2}(y_i - u)^2$, we have $f_i^*(\alpha) = \frac{1}{2}\alpha^2 + \alpha y_i$. For problems of this form, the dual variables can be directly mapped to the primal variables, such that for a vector $\boldsymbol{\alpha}^*$ which attains the maximum of (6.2), the optimal primal solution has the form $\boldsymbol{\beta}^*(\boldsymbol{\alpha}^*) = -\frac{1}{n\lambda}\mathbf{X}^\top \boldsymbol{\alpha}^*$.

Clearly, a similar dual problem to (6.2) can be defined in the projected space. Defining $\tilde{\mathbf{K}} = (\mathbf{X}\mathbf{\Pi})(\mathbf{X}\mathbf{\Pi})^\top$ we have

$$\max_{\boldsymbol{\alpha} \in \mathbb{R}^n} - \sum_{i=1}^{n} f_i^*(\alpha_i) - \frac{1}{2n\lambda} \boldsymbol{\alpha}^\top \tilde{\mathbf{K}} \boldsymbol{\alpha}. \tag{6.3}$$

Importantly, the vector of dual variables does not change dimension depending on whether the original problem (6.2) or the projected problem (6.3) is being solved. Under mild assumptions on the loss function, by mapping the solution to this new problem, $\tilde{\boldsymbol{\alpha}}$, back to the original space one obtains a vector $\tilde{\boldsymbol{\beta}}(\tilde{\boldsymbol{\alpha}}) = -\frac{1}{n\lambda}\mathbf{X}^\top \tilde{\boldsymbol{\alpha}}$, which is a *good* approximation to $\boldsymbol{\beta}^*$, the solution to the original problem (6.1) (Zhang et al., 2014, 2012).

6.3. The DUAL-LOCO algorithm

In this section we detail the DUAL-LOCO algorithm. DUAL-LOCO differs from the original LOCO algorithm in two important ways. (i) The random features from each worker are summed, rather than concatenated, to obtain a τ_{subs} dimensional approximation allowing for an efficient implementation in a large-scale distributed environment. (ii) Each worker solves a local *dual* problem similar to (6.3). This allows us to extend the theoretical guarantees to a larger class of estimation problems beyond ridge regression (§6.4).

We consider the case where p features are distributed across K different workers in non-overlapping subsets $\mathcal{P}_1, \ldots, \mathcal{P}_K$ of equal size[2], $\tau = p/K$.

Algorithm 8 DUAL-LOCO

```
Input: Data: $\mathbf{X}, Y$, # workers: $K$; Params.: $\tau_{subs}$, $\lambda$
Partition $\{1, \ldots, p\}$ into $K$ subsets of equal size $\tau$ and distribute feature vectors in $\mathbf{X}$ accordingly over $K$ workers.
for each worker $k \in \{1, \ldots K\}$ in parallel do
    Compute the SRHT projection matrix $\mathbf{\Pi}_k$.
    Send random features $\mathbf{X}_k \mathbf{\Pi}_k$ to other workers.
    Receive random features from other workers and construct $\bar{\mathbf{X}}_k$.
    $\tilde{\boldsymbol{\alpha}}_k \leftarrow$ LocalDualSolver($\bar{\mathbf{X}}_k, Y, \lambda$)
    $\widehat{\boldsymbol{\beta}}_k = -\frac{1}{n\lambda} \mathbf{X}_k^\top \tilde{\boldsymbol{\alpha}}_k$
    Send $\widehat{\boldsymbol{\beta}}_k$ to driver.
end for
Output: Solution vector: $\widehat{\boldsymbol{\beta}} = \left[\widehat{\boldsymbol{\beta}}_1, \ldots, \widehat{\boldsymbol{\beta}}_K\right]$
```

Since most loss functions of interest are not separable across coordinates, a key challenge addressed by DUAL-LOCO is to define a local minimization problem for each worker to solve *independently* and *asynchronously* while still maintaining important dependencies between features in different blocks and keeping communication overhead low. Algorithm 8 details DUAL-LOCO in full.

We can rewrite (6.1) making explicit the contribution from block k. Letting $\mathbf{X}_k \in \mathbb{R}^{n \times \tau}$ be the sub-matrix whose columns correspond to the coordinates in $\mathcal{P}_k$ (the "raw" features of block k) and $\mathbf{X}_{(-k)} \in \mathbb{R}^{n \times (p-\tau)}$ be the remaining columns of $\mathbf{X}$, we have

$$J(\boldsymbol{\beta}) = \sum_{i=1}^{n} f_i \left(\mathbf{x}_{i,k}^\top \boldsymbol{\beta}_{\text{raw}} + \mathbf{x}_{i,(-k)}^\top \boldsymbol{\beta}_{(-k)} \right) + \lambda \left(\|\boldsymbol{\beta}_{\text{raw}}\|^2 + \|\boldsymbol{\beta}_{(-k)}\|^2 \right). \quad (6.4)$$

---

[2] This is for simplicity of notation only, in general the partitions can be of different sizes.

Where $\mathbf{x}_{i,k}$ and $\mathbf{x}_{i,(-k)}$ are the rows of $\mathbf{X}_k$ and $\mathbf{X}_{(-k)}$ respectively. We replace $\mathbf{X}_{(-k)}$ in each block with a low-dimensional randomized approximation which preserves its contribution to the loss function. This procedure is described in Figure 6.1 and we describe some steps in more detail below.

In **Step 3**, each worker computes its local SRHT projection matrix $\mathbf{\Pi}_k$ (cf. §6.2.2), which is independent from the other workers' SRHT projection matrices.

In **Step 4**, the matrices of random features $\mathbf{X}_k\mathbf{\Pi}_k$ are communicated and in **Step 5**, worker k constructs the matrix

$$\bar{\mathbf{X}}_k = \left[\mathbf{X}_k, \sum_{k'=1,k'\neq k}^{K} \mathbf{X}_{k'}\mathbf{\Pi}_{k'}\right] \quad \text{i.e.} \quad \bar{\mathbf{X}}_k \in \mathbb{R}^{n\times(\tau+\tau_{subs})} \tag{6.5}$$

which is the concatenation of worker k's raw features and the *sum* of the random features from all other workers.

As we prove in Lemma 6.2, summing $\mathbb{R}^{\tau} \to \mathbb{R}^{\tau_{subs}}$-dimensional random projections from $(K-1)$ blocks is equivalent to computing the $\mathbb{R}^{(p-\tau)} \to \mathbb{R}^{\tau_{subs}}$-dimensional random projection in one go. The latter operation is impractical for very large p and not applicable when the features are distributed. Therefore, summing the random features from each worker allows the dimensionality reduction to be distributed across workers. Additionally, the summed random feature representation can be computed and combined very efficiently. We elaborate on this aspect in §6.5.

For a single worker the local, approximate primal problem is then

$$\min_{\bar{\boldsymbol{\beta}}\in\mathbb{R}^{\tau+\tau_{subs}}} J_k(\bar{\boldsymbol{\beta}}) := \sum_{i=1}^{n} f_i(\bar{\boldsymbol{\beta}}^\top \bar{\mathbf{x}}_i) + \frac{\lambda}{2}\|\bar{\boldsymbol{\beta}}\|^2 \tag{6.6}$$

where $\bar{\mathbf{x}}_i \in \mathbb{R}^{\tau+\tau_{subs}}$ is the i^{th} row of $\bar{\mathbf{X}}_k$. The corresponding dual problem for each worker in the Dual-Loco algorithm is

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^n} -\sum_{i=1}^{n} f_i^*(\alpha_i) - \frac{1}{2n\lambda}\boldsymbol{\alpha}^\top\tilde{\mathbf{K}}_k\boldsymbol{\alpha}, \quad \tilde{\mathbf{K}}_k = \bar{\mathbf{X}}_k\bar{\mathbf{X}}_k^\top. \tag{6.7}$$

The following steps in Algorithm 8 detail respectively how the solution to (6.7) and the final Dual-Loco estimates are obtained.

Step 6. `LocalDualSolver`. The `LocalDualSolver` computes the solution for (6.7), the local dual problem. The solver can be chosen to best suit the problem at hand. This will depend on the absolute size of n and $\tau + \tau_{subs}$ as well as on their ratio. For example, we could use SDCA (Shalev-Shwartz and Zhang, 2013) or Algorithm 1 from Zhang et al. (2012).

Step 7. Obtaining the global primal solution. Each worker maps its local dual solution to the primal solution corresponding only to the coordinates in $\mathcal{P}_k$. In this way, each worker returns coefficients corresponding only to its own raw features. The final primal solution vector is obtained by concatenating the K local solutions. Unlike Loco, we no longer require to discard the coefficients corresponding to the random features for each worker. Consequently, computing estimates is more efficient (especially when $p \gg n$).

6.4. Dual-Loco approximation error

In this section we bound the recovery error between the Dual-Loco solution and the solution to Eq. (6.1).

Theorem 6.1 (Dual-Loco error bound) *Consider a matrix* $\mathbf{X} \in \mathbb{R}^{n \times p}$ *with rank at most* r. *Assume that the loss* $f(\cdot)$ *is smooth with Lipschitz continuous gradients. For a subsampling dimension* $\tau_{subs} \geq c_1 pK$ *where* $0 \leq c_1 \leq 1/K^2$, *let* $\boldsymbol{\beta}^*$ *be the solution to* (6.1) *and* $\widehat{\boldsymbol{\beta}}$ *be the estimate returned by Algorithm 8. We have with probability at least* $1 - K\left(\delta + \frac{p-\tau}{e^r}\right)$

$$\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\| \quad \leq \quad \frac{\varepsilon}{1-\varepsilon}\|\boldsymbol{\beta}^*\| \quad \textit{where} \quad \varepsilon = \sqrt{\frac{c_0 \log(2r/\delta) r}{c_1 p}} < 1. \quad (6.8)$$

Proof. By Lemma 6.5 and applying a union bound we can decompose the global optimization error in terms of the error due to each worker as $\|\boldsymbol{\beta}^* - \widehat{\boldsymbol{\beta}}\| = \sqrt{\sum_{k=1}^{K} \|\boldsymbol{\beta}_k^* - \widehat{\boldsymbol{\beta}}_k\|^2} \leq \sqrt{K}\frac{\rho}{1-\rho}\|\boldsymbol{\beta}^*\|$, which holds with probability $1 - K\left(\delta + \frac{p-\tau}{e^r}\right)$. The final bound, (6.8) follows by setting $\rho = \sqrt{\frac{c_0 \log(2r/\delta) r}{\tau_{subs}}}$ and $\tau_{subs} \geq c_1 pK$ and noting that $\sqrt{K} \cdot \frac{\frac{\varepsilon}{\sqrt{K}}}{1 - \frac{\varepsilon}{\sqrt{K}}} \leq \frac{\varepsilon}{1-\varepsilon}$. □

Theorem 6.1 guarantees that the solution to Dual-Loco will be close to the optimal solution obtained by a single worker with access to all of the data. Our result relies on the data having rank $r \ll p$. In practice, this assumption is often fulfilled, in particular when the data is high dimensional. For a large enough projection dimension, the bound has only a weak dependence on K through the union bound used to determine the probability with which Eq. (6.8) holds. The error is then mainly determined by the ratio between the rank and the random projection dimension. When the rank of $\mathbf{X}$ increases for a fixed p, we need a larger projection dimension to accurately capture its spectrum. On the other hand, the failure probability increases with p and decreases with r. However, this countering effect is negligible as typically $\log(p-\tau) \ll r$.

6.5. Implementation and experiments

In this section we report on the empirical performance of Dual-Loco in two sets of experiments. The first demonstrates the performance of Dual-Loco in a large, distributed classification task. The second is an application of ℓ_2 penalized regression to a problem in climate science where accurate recovery of the coefficient estimates is of primary interest.

Cross validation. In most practical cases, the regularization parameter λ is unknown and has to be determined via v-fold cross validation (CV). The chosen algorithm is usually run entirely once for each fold and each of l values of λ, leading to a runtime that is approximately $v \cdot l$ as large as the runtime of a single run[3]. In this context, Dual-Loco has the advantage that steps 3-5 in Algorithm 8 are independent of λ. Therefore, these steps only need to be performed *once per fold*. In step 6, we then estimate $\tilde{\boldsymbol{\alpha}}_k$ for each value in the provided sequence for λ. Thus, the runtime of Dual-Loco will increase by much less than $v \cdot l$ compared to the runtime of a single run. The performance of each value for λ is then not only averaged over the random split of the training data set into v parts but also over the randomness introduced by the random projections which are computed and communicated once per fold. The procedure is provided in full detail in Algorithm 9 in Appendix 6.C.

Implementation details. We implemented Dual-Loco using the Apache Spark framework[4]. Spark is increasingly gaining traction in the research community as well as in industry due to its easy-to-use high-level API and the benefits of in-memory processing. Spark is up to 100× faster than Hadoop MapReduce. Additionally, Spark can be used in many different large-scale computing environments and the various, easily-integrated libraries for a diverse set of tasks greatly facilitate the development of applications.

---

[3] "Approximately" since the cross validation procedure also requires time for testing. For a single run we only count the time it takes to estimate the parameters.

[4] Available from: `http://christinaheinze.github.io/loco-lib/`.

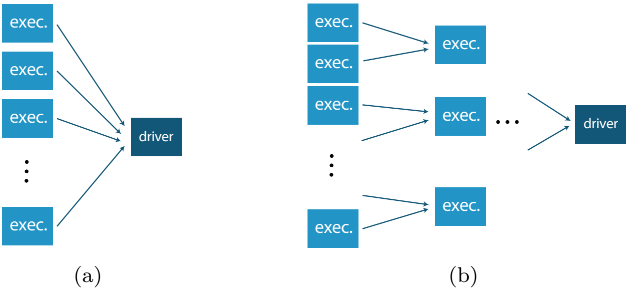


Figure 6.2.: Schematic for the aggregation of the random features in Spark. (a) When concatenating the random features naively, every worker node (exec.) sends its random features to the driver from where they are broadcasted to all workers. (b) Using the `treeReduce` scheme we can reduce the load on the driver by summing the random features from each worker node as this operation is associative and commutative. Worker k is only required to subtract its own random features locally.

When communicating and summing the random features in Spark, Dual-Loco leverages the `treeReduce` scheme as illustrated in Figure 6.2(b). Summing has the advantage that increasing the number of workers simply introduces more layers in the tree structure (Figure 6.2b) while the load on the driver remains constant and the aggregation operation also benefits from a parallel execution. Thus, when increasing K only relatively little additional communication cost is introduced which leads to speedups as demonstrated below.

In practice, we used the discrete cosine transform (DCT) provided in the FFT library jTransforms[5][6] and we ran Dual-Loco as well as CoCoA$^+$ on a high-performance cluster[7].

Competing methods. For the classification example, the loss function is the hinge loss. Although the problem is non-smooth, and therefore not covered by our theory, we still obtain good results suggesting that Theorem 6.1 can be generalized to non-smooth losses. Alternatively, for classification the smoothed hinge or logistic losses could be used. For the

---

[5] `https://sites.google.com/site/piotrwendykier/software/jtransforms`

[6] For the Hadamard transform, τ must be a power of two. For the DCT there is no restriction on τ and very similar theoretical guarantees hold.

[7] CoCoA$^+$ is also implemented in Spark with code available from `https://github.com/gingsmith/cocoa`.

regression problem we use the squared error loss and modify CoCoA$^+$ accordingly. As the `LocalDualSolver` we use SDCA (Shalev-Shwartz and Zhang, 2013).

We also compared Dual-Loco against the reference implementation of distributed loss minimization in the MLlib library in Spark using SGD and L-BFGS. However, after extensive cross-validation over regularization strength (and step size and mini-batch size in case of SGD), we observed that the variance was still very large and so we omit the MLlib implementations from the figures. A comparison between CoCoA and variants of SGD and mini-batch SDCA can be found in Jaggi et al. (2014).

Kaggle Dogs vs Cats dataset. This is a binary classification task consisting of $25,000$ images of dogs and cats[8]. We resize all images to 430×430 pixels and use Overfeat (Sermanet et al., 2014)—a pre-trained convolutional neural network—to extract $p = 200,704$ *fully dense* feature vectors from the 19^{th} layer of the network for each image. We train on $n_{train} = 20,000$ images and test on the remaining $n_{test} = 5,000$. The size of the training data is 37GB with over 4 billion non-zero elements. All results we report in the following are averaged over five repetitions and by "runtime" we refer to wall clock time.

Figure 6.3 shows the median normalized training and test prediction MSE of Dual-Loco and CoCoA$^+$ for different numbers of workers[9]. For Dual-Loco, we also vary the size of the random feature representation and choose $\tau_{subs} = \{0.005, 0.01, 0.02\} \times (p - \tau)$. The corresponding errors are labeled with Dual-Loco 0.5, Dual-Loco 1 and Dual-Loco 2. Note that combinations of K and τ_{subs} that would result in $\tau < \tau_{subs}$ cannot be used since the projection dimension τ_{subs} should be smaller than τ to achieve a dimensionality reduction (e.g. this is the case for $K = 192$ and $\tau_{subs} = 0.01 \times (p - \tau)$). We ran CoCoA$^+$ until a duality gap of 10^{-2} was attained so that the number of iterations varies for different numbers of workers[10]. Notably, for $K = 48$ more iterations were needed than in the other cases which is reflected in the very low training error in this case. The fraction of local points to be processed per round was set to 10%. We determined the regularization parameter λ via 5-fold cross validation.

While the differences in training errors between Dual-Loco and CoCoA$^+$

---

[8] `https://www.kaggle.com/c/dogs-vs-cats`

[9] In practice, this choice will depend on the available resources in addition to the size of the data set.

[10] For K ranging from 12 to 192, the number of iterations needed were $77, 207, 4338, 1966$, resp. 3199.

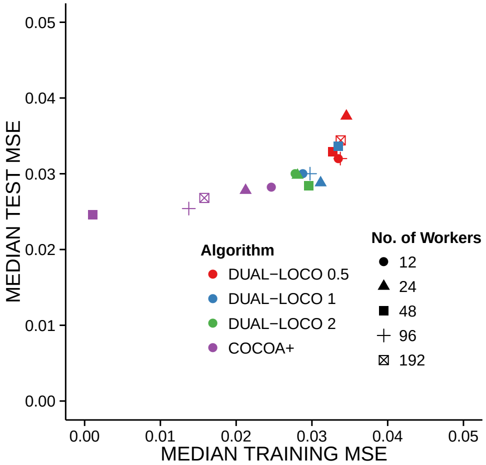


Figure 6.3.: Dogs vs Cats data: Median normalized training and test prediction MSE based on 5 repetitions.

are notable, the differences between the test errors are minor as long as the random feature representation is large enough. Choosing τ_{subs} to be only 0.5% of $p - \tau$ seems to be slightly too small for this data set. When setting τ_{subs} to be 1% of $p - \tau$ the largest difference between the test errors of DUAL-LOCO and COCOA$^+$ is 0.9%. The averaged mean squared prediction errors and their standard deviations are collected in Table 6.1 in Appendix 6.C.

Next, we would like to compare the wall clock time needed to find the regularization parameter λ via 5-fold cross validation. For COCOA$^+$, using the number of iterations needed to attain a duality gap of 10^{-2} would lead to runtimes of more than 24 hours for $K \in \{48, 96, 192\}$ when comparing $l = 20$ possible values for λ. One might argue that using a duality gap of 10^{-1} is sufficient for the cross validation runs which would speed up the model selection procedure significantly as much fewer iterations would be required. Therefore, for $K \geq 48$ we use a duality gap of 10^{-1} during cross validation and a duality gap of 10^{-2} for learning the parameters, once λ has been determined. Figure 6.4 shows the runtimes when $l = 20$ possible values for λ are compared; Figure 6.6(a) compares the runtimes when cross validation is performed over $l = 50$ values. The absolute runtime of COCOA$^+$ for a single run is smaller for $K = 12$ and $K = 24$ and larger

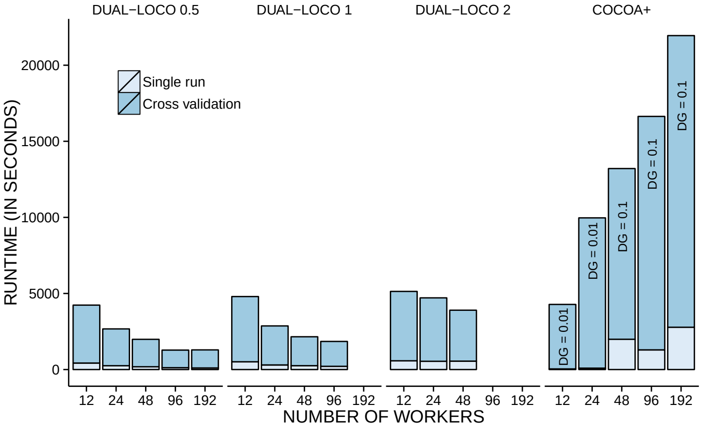
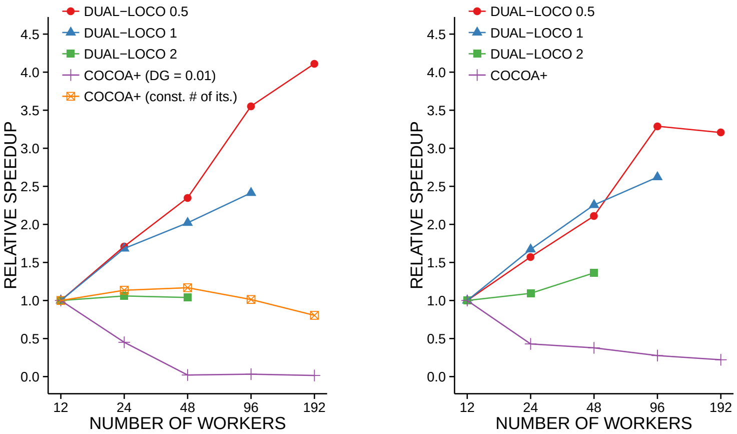


Figure 6.4.: Total wall clock time including 5-fold CV over $l = 20$ values for λ. For CoCoA^{+}, we use a duality gap (DG) of 10^{-1} for the CV runs when $K \geq 48$.

for $K \in \{48, 96, 192\}$, so using more workers increased the amount of wall clock time necessary for job completion. The total runtime, including cross validation and a single run to learn the parameters with the determined value for λ, is always smaller for Dual-Loco, except when $K = 12$ and $l = 20$.

Figures 6.5 and 6.6(b) show the relative speedup of Dual-Loco and CoCoA^{+} when increasing K. The speedup is computed by dividing the runtime for $K = 12$ by the runtime achieved for the corresponding $K = \{24, 48, 96, 192\}$. A speedup value smaller than 1 implies an *increase* in runtime. When considering a single run, we run CoCoA^{+} in two different settings: **(i)** We use the number of iterations that are needed to obtain a duality gap of 10^{-2} which varies for different number of workers[10]. Here, the speedup is smaller than 1 for all K. **(ii)** We fix the number of outer iterations to a constant number. As K increases, the number of inner iterations decreases, making it easier for CoCoA^{+} to achieve a speedup. We found that although CoCoA^{+} attains a speedup of 1.17 when increasing K from 12 to 48 (equivalent to a decrease in runtime of 14%), CoCoA^{+} suffers a 24% increase in runtime when increasing K from 12 to 192.

For Dual-Loco 0.5 and Dual-Loco 1 we observe significant speedups as K increases. As we split the design matrix by features the number of



(a) (b)

Figure 6.5.: Relative speedup for (a) a single run and (b) 5-fold CV over $l = 20$ values for λ.

observations n remains constant for different number of workers. At the same time, the dimensionality of each worker's local problem decreases with K. Together with the efficient aggregation of the random features, this leads to shorter runtimes. In case of DUAL-LOCO 2, the communication costs dominate the costs of computing the random projection and of the `LocalDualSolver`, resulting in much smaller speedups.

Although COCOA$^+$ was demonstrated to obtain speedups for low-dimensional data sets (Ma et al., 2015) it is plausible that the same performance cannot be expected on a very high-dimensional data set. This illustrates that in such a high-dimensional setting splitting the design matrix according to the columns instead of the rows is more suitable.

Climate data. This is a regression task where we demonstrate that the coefficients returned by DUAL-LOCO are interpretable. The data set contains the outcome of control simulations of the GISS global circulation model (Knutti et al., 2013; Schmidt et al., 2014) and is part of the CMIP5 climate modeling ensemble. We aim to forecast the monthly global average temperature Y in February using the air pressure measured in January. Results are very similar for other months. The $p = 10,368$ features are pressure measurements taken at 10,368 geographic grid points in January.

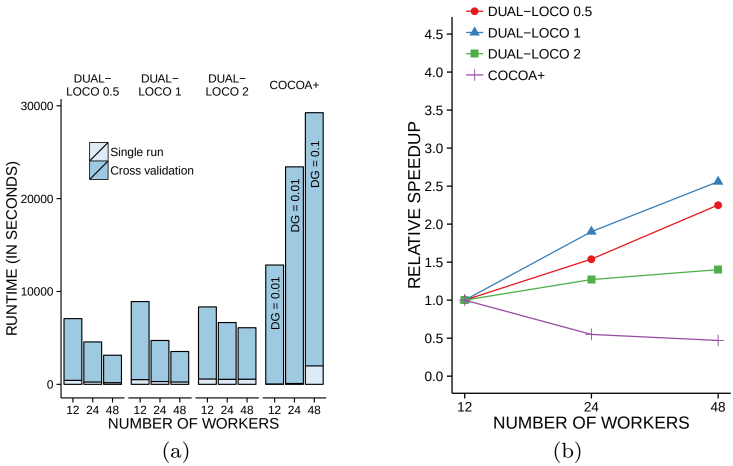
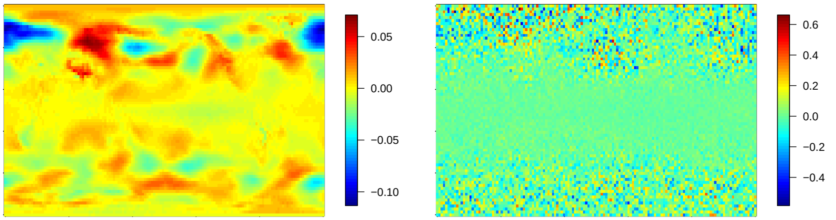


Figure 6.6.: 5-fold CV over $l = 50$ values for λ: (a) Total wall clock time and (b) relative speedup.

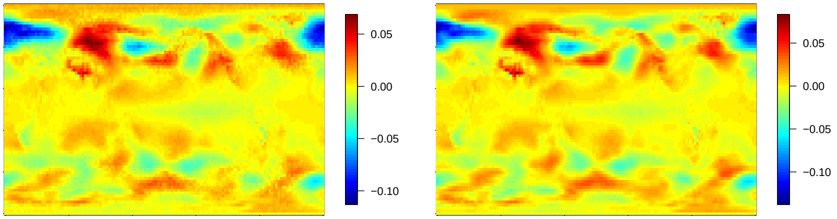
The time span of the climate simulation is 531 years and we use the results from two control simulations, yielding $n_{\text{train}} = 849$ and $n_{\text{test}} = 213$.

In Figure 6.7 we compare the coefficient estimates for four different methods. The problem is small enough to be solved on a single machine so that the full solution can be computed (using SDCA; cf. Figure 6.7(a)). This allows us to report the normalized parameter estimation mean squared error ($\text{MSE}_{\widehat{\boldsymbol{\beta}}}$) with respect to the full solution in addition to the normalized mean squared prediction error (MSE). The solution indicates that the pressure differential between Siberia (red area, top middle-left) and Europe and the North Atlantic (blue area, top left and top right) is a good predictor for the temperature anomaly. This pattern is concealed in Figure 6.7(b) which shows the result of up-projecting the coefficients estimated following a random projection of the columns. Using this scheme for prediction was introduced in Lu et al. (2013). Although the MSE is similar to the optimal solution, the recovered coefficients are not interpretable as suggested by Zhang et al. (2012). Thus, this method should only be used if prediction is the sole interest. Figure 6.7(c) shows the estimates returned by Dual-Loco which is able to recover estimates which are close to the full solution. Finally, Figure 6.7(d) shows that CoCoA$^+$ also attains accurate results.



(a) Single machine:
Full solution (MSE $= 0.72$)

(b) Single machine:
Column-wise compression
(MSE $= 0.73$, $\text{MSE}_{\widehat{\boldsymbol{\beta}}} = 21.28$)



(c) Distributed setting:
Dual-Loco 10 with $K = 4$
(MSE $= 0.72$, $\text{MSE}_{\widehat{\boldsymbol{\beta}}} = 0.02$)

(d) Distributed setting:
CoCoA$^+$ with $K = 4$
(MSE $= 0.72$, $\text{MSE}_{\widehat{\boldsymbol{\beta}}} = 0.01$)

Figure 6.7.: Climate data: The regression coefficients are shown as maps with the prime median (passing through London) corresponding to the left and right edge of the plot. The Pacific Ocean lies in the center of each map.

Considering a longer time period or adding additional model variables such as temperature, precipitation or salinity rapidly increases the dimensionality of the problem while the number of observations remains constant. Each additional variable adds $10,368$ dimensions per month of simulation. Estimating very high-dimensional linear models is a significant challenge in climate science and one where distributing the problem across features instead of observations is advantageous. The computational savings are much larger when distributing across features as $p \gg n$ and thus reducing p is associated with larger gains than when distributing across observations.

6.6. Conclusions and further work

We have presented Dual-Loco which considers the challenging and rarely studied problem of statistical estimation when data is distributed across features rather than samples. Dual-Loco generalizes Loco to a wider variety of loss functions for regression and classification. We show that the estimated coefficients are close to the optimal coefficients that could be learned by a single worker with access to the entire dataset. The resulting bound is more intuitive and tighter than previous bounds, notably with a very weak dependence on the number of workers. We have demonstrated that Dual-Loco is able to recover accurate solutions for large-scale estimation tasks whilst also achieving better scaling than a state-of-the-art competitor, CoCoA$^+$, as K increases. Additionally, we have shown that Dual-Loco allows for fast model selection using cross-validation.

The dual formulation is convenient for ℓ_2 penalized problems but other penalties are not as straightforward. Similarly, the theory only holds for smooth loss functions. However, as demonstrated empirically Dual-Loco also performs well with a non-smooth loss function.

As n grows very large, the random feature matrices may become too large to communicate efficiently even when the projection dimension is very small. For these situations, there are a few simple extensions we aim to explore in future work. One possibility is to first perform row-wise random projections (cf. Mahoney (2011)) to further reduce the communication requirement. Another option is to distribute $\mathbf{X}$ according to rows and columns.

Contrary to stochastic optimization methods, the communication of Dual-Loco is limited to a single round. For fixed n, p and τ_{subs}, the amount of communication is deterministic and can be fixed ahead of time. This can be beneficial in settings where there are additional constraints on communication (for example when different blocks of features are distributed *a priori* across different physical locations).

Clearly with additional communication, the theoretical and practical performance of Dual-Loco could be improved. For example, Zhang et al. (2012) suggest an iterative dual random projection scheme which can reduce the error in Lemma 6.5 exponentially. A related question for future research involves quantifying the amount of communication performed by Dual-Loco in terms of known minimax lower bounds (Zhang et al., 2013b).

Appendix 6.A Supplementary results

Here we introduce two lemmas. The first describes the random projection construction which we use in the distributed setting.

Lemma 6.2 (Summing random features) *Consider the singular value decomposition* $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ *where* $\mathbf{U} \in \mathbb{R}^{n\times r}$ *and* $\mathbf{V} \in \mathbb{R}^{p\times r}$ *have orthonormal columns and* $\mathbf{\Sigma} \in \mathbb{R}^{r\times r}$ *is diagonal;* $r = rank(\mathbf{X})$*. In addition to the raw features, let* $\bar{\mathbf{X}}_k \in \mathbb{R}^{n\times(\tau+\tau_{subs})}$ *contain random features which result from summing the* $K-1$ *random projections from the other workers, as defined in* (6.5)*. Furthermore, assume without loss of generality that the problem is permuted so that the raw features of worker k's problem are the first* τ *columns of* $\mathbf{X}$ *and* $\bar{\mathbf{X}}_k$*. Let* $\mathbf{\Pi}_S$ *denote the row-wise concatenation of the SRHT projection matrices* $\mathbf{\Pi}_{k'}$ *from the* $K-1$ *other workers, i.e.* $\mathbf{\Pi}_S = [\mathbf{\Pi}_{k'}^\top]^\top_{k'\in\{1,\dots,K\setminus k\}}$ *and* $\mathbf{\Pi}_S \in \mathbb{R}^{(p-\tau)\times\tau_{subs}}$*. Finally, let*

$$\Theta_S = \begin{bmatrix} \mathbf{I}_\tau & 0 \\ 0 & \mathbf{\Pi}_S \end{bmatrix} \in \mathbb{R}^{p\times(\tau+\tau_{subs})}$$

such that $\bar{\mathbf{X}}_k = \mathbf{X}\Theta_S$*.*

There exists a fixed positive constant c_0 *such that*

$$\|\mathbf{V}^\top\Theta_S\Theta_S^\top\mathbf{V} - \mathbf{V}^\top\mathbf{V}\| \leq \sqrt{\frac{c_0\log(2r/\delta)r}{\tau_{subs}}}.$$

with probability at least $1 - \left(\delta + \frac{p-\tau}{e^r}\right)$*.*

Proof. See Appendix 6.B. □

Definition 6.3 *For ease of exposition, we shall rewrite the dual problems so that we consider minimizing convex objective functions. More formally, the original problem is then given by*

$$\boldsymbol{\alpha}^* = \operatorname*{argmin}_{\boldsymbol{\alpha}\in\mathbb{R}^n} \left\{ D(\boldsymbol{\alpha}) := \sum_{i=1}^n f_i^*(\alpha_i) + \frac{1}{2n\lambda}\boldsymbol{\alpha}^\top\mathbf{K}\boldsymbol{\alpha} \right\}. \tag{6.9}$$

The problem worker k solves is described by

$$\tilde{\boldsymbol{\alpha}} = \operatorname*{argmin}_{\boldsymbol{\alpha}\in\mathbb{R}^n} \left\{ \tilde{D}_k(\boldsymbol{\alpha}) := \sum_{i=1}^n f_i^*(\alpha_i) + \frac{1}{2n\lambda}\boldsymbol{\alpha}^\top\tilde{\mathbf{K}}_k\boldsymbol{\alpha} \right\}. \tag{6.10}$$

Recall that $\tilde{\mathbf{K}}_k = \bar{\mathbf{X}}_k \bar{\mathbf{X}}_k^\top$*, where* $\bar{\mathbf{X}}_k$ *is the concatenation of the* τ raw *features and* τ_{subs} random *features for worker* k*.*

To proceed we need the following result which relates the solution of the original problem to that of the approximate problem solved by worker k.

Lemma 6.4 (Adapted from Lemma 1 in Zhang et al. (2014).) *Let* $\boldsymbol{\alpha}^*$ *and* $\tilde{\boldsymbol{\alpha}}$ *be as defined in Definition 6.3. We obtain*

$$\frac{1}{\lambda}(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*)^\top \left(\mathbf{K} - \tilde{\mathbf{K}}_k\right) \boldsymbol{\alpha}^* \geq \frac{1}{\lambda}(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*)^\top \tilde{\mathbf{K}}_k (\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*). \tag{6.11}$$

Proof. See Zhang et al. (2014). ☐

For our main result, we rely heavily on the following variant of Theorem 1 in Zhang et al. (2014) which bounds the difference between the coefficients estimated by worker k, $\widehat{\boldsymbol{\beta}}_k$ and the corresponding coordinates of the optimal solution vector $\boldsymbol{\beta}_k^*$.

Lemma 6.5 (Local optimization error. Adapted from Zhang et al. (2014).) *For* $\rho = \sqrt{\frac{c_0 \log(2r/\delta) r}{\tau_{subs}}}$ *the following holds*

$$\|\widehat{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_k^*\| \leq \frac{\rho}{1-\rho} \|\boldsymbol{\beta}^*\|$$

with probability at least $1 - \left(\delta + \frac{p-\tau}{e^r}\right)$*.*

The proof closely follows the proof of Theorem 1 in Zhang et al. (2014) which we restate here identifying the major differences.

Proof. Let the quantities $\tilde{D}_k(\boldsymbol{\alpha})$, $\tilde{\mathbf{K}}_k$, be as in Definition 6.3. For ease of notation, we shall omit the subscript k in $\tilde{D}_k(\boldsymbol{\alpha})$ and $\tilde{\mathbf{K}}_k$ in the following.

By the SVD we have $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top$. So $\mathbf{K} = \mathbf{U}\boldsymbol{\Sigma}\boldsymbol{\Sigma}\mathbf{U}^\top$ and $\tilde{\mathbf{K}} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^\top \Theta_S \Theta_S^\top \mathbf{V}\boldsymbol{\Sigma}\mathbf{U}^\top$ with Θ_S as defined in Lemma 6.2. We can make the following definitions

$$\gamma^* = \boldsymbol{\Sigma}\mathbf{U}^\top \boldsymbol{\alpha}^*, \qquad \tilde{\gamma} = \boldsymbol{\Sigma}\mathbf{U}^\top \tilde{\boldsymbol{\alpha}}.$$

Defining $\tilde{\mathbf{M}} = \mathbf{V}^\top \Theta_S \Theta_S^\top \mathbf{V}$ and plugging these into Lemma 6.4 we obtain

$$(\tilde{\gamma} - \gamma^*)^\top (\mathbf{I} - \tilde{\mathbf{M}}) \gamma^* \geq (\tilde{\gamma} - \gamma^*)^\top \tilde{\mathbf{M}} (\tilde{\gamma} - \gamma^*). \tag{6.12}$$

We now bound the spectral norm of $\mathbf{I} - \tilde{\mathbf{M}}$ using Lemma 6.2. Recall that Lemma 6.2 bounds the difference between a matrix and its approximation by a *distributed* dimensionality reduction using the SRHT.

Using the Cauchy-Schwarz inequality we have for the l.h.s. of (6.12) with $\rho = \sqrt{\frac{c_0 \log(2r/\delta) r}{\tau_{subs}}}$

$$(\tilde{\gamma} - \gamma^*)^\top \left(\mathbf{I} - \tilde{\mathbf{M}}\right) \gamma^* \leq \rho \|\gamma^*\| \|\tilde{\gamma} - \gamma^*\|.$$

For the r.h.s. of (6.12), we can write

$$\begin{aligned} &(\tilde{\gamma} - \gamma^*)^\top \tilde{\mathbf{M}} (\tilde{\gamma} - \gamma^*) \\ &= \|\tilde{\gamma} - \gamma^*\|^2 - (\tilde{\gamma} - \gamma^*)^\top \left(\mathbf{I} - \tilde{\mathbf{M}}\right) (\tilde{\gamma} - \gamma^*) \\ &\geq \|\tilde{\gamma} - \gamma^*\|^2 - \rho \|\tilde{\gamma} - \gamma^*\|^2 \\ &= (1 - \rho) \|\tilde{\gamma} - \gamma^*\|^2. \end{aligned}$$

Combining these two expressions and inequality (6.12) yields

$$\begin{aligned} (1 - \rho) \|\tilde{\gamma} - \gamma^*\|^2 &\leq \rho \|\gamma^*\| \|\tilde{\gamma} - \gamma^*\| \\ (1 - \rho) \|\tilde{\gamma} - \gamma^*\| &\leq \rho \|\gamma^*\|. \end{aligned} \tag{6.13}$$

From the definition of γ^* and $\tilde{\gamma}$ above and $\boldsymbol{\beta}^*$ and $\tilde{\boldsymbol{\beta}}$, respectively we have

$$\boldsymbol{\beta}^* = -\frac{1}{n\lambda} \mathbf{V} \gamma^*, \qquad \tilde{\boldsymbol{\beta}} = -\frac{1}{n\lambda} \mathbf{V} \tilde{\gamma}$$

so $\frac{1}{n\lambda} \|\gamma^*\| = \|\boldsymbol{\beta}^*\|$ and $\|\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\| = \frac{1}{n\lambda} \|\tilde{\gamma} - \gamma^*\|$ due to the orthonormality of $\mathbf{V}$. Plugging this into (6.13) and using the fact that $\|\boldsymbol{\beta}^* - \tilde{\boldsymbol{\beta}}\| \geq \|\boldsymbol{\beta}_k^* - \widehat{\boldsymbol{\beta}}_k\|$ we obtain the stated result. □

Appendix 6.B Proof of row summing lemma

Proof of Lemma 6.2 . Let $\mathbf{V}_k$ contain the first τ rows of $\mathbf{V}$ and let $\mathbf{V}_{(-k)}$ be the matrix containing the remaining rows. Decompose the matrix products as follows

$$\mathbf{V}^\top \mathbf{V} = \mathbf{V}_k^\top \mathbf{V}_k + \mathbf{V}_{(-k)}^\top \mathbf{V}_{(-k)}$$

and

$$\mathbf{V}^\top \Theta_S \Theta_S^\top \mathbf{V} = \mathbf{V}_k^\top \mathbf{V}_k + \tilde{\mathbf{V}}_k^\top \tilde{\mathbf{V}}_k$$

with $\tilde{\mathbf{V}}_k^\top = \mathbf{V}_{(-k)}^\top \mathbf{\Pi}_S$. Then

$$\begin{aligned}
&\|\mathbf{V}^\top \Theta_S \Theta_S^\top \mathbf{V} - \mathbf{V}^\top \mathbf{V}\| \\
&= \|\mathbf{V}_k^\top \mathbf{V}_k + \tilde{\mathbf{V}}_k^\top \tilde{\mathbf{V}}_k - \mathbf{V}_k^\top \mathbf{V}_k - \mathbf{V}_{(-k)}^\top \mathbf{V}_{(-k)}\| \\
&= \|\mathbf{V}_{(-k)}^\top \mathbf{\Pi}_S \mathbf{\Pi}_S^\top \mathbf{V}_{(-k)} - \mathbf{V}_{(-k)}^\top \mathbf{V}_{(-k)}\|.
\end{aligned}$$

Since Θ_S is an orthogonal matrix, from Lemma 3.3 in Tropp (2011) and Lemma 6.6, summing $(K-1)$ independent SRHTs from τ to τ_{subs} is equivalent to applying a single SRHT from $p-\tau$ to τ_{subs}. Therefore we can simply apply Lemma 1 of Lu et al. (2013) to the above to obtain the result. ☐

Lemma 6.6 (Summed row sampling) *Let* $\mathbf{W}$ *be an* $n \times p$ *matrix with orthonormal columns. Let* $\mathbf{W}_1, \ldots, \mathbf{W}_K$ *be a balanced, random partitioning of the rows of* $\mathbf{W}$ *where each matrix* $\mathbf{W}_k$ *has exactly* $\tau = n/K$ *rows. Define the quantity* $M := n \cdot \max_{j=1,\ldots n} \|e_j^\top \mathbf{W}\|^2$. *For a positive parameter* α, *select the subsample size*

$$l \cdot K \geq \alpha M \log(p).$$

Let $\mathbf{S}_{T_k} \in \mathbb{R}^{l \times \tau}$ *denote the operation of uniformly at random sampling a subset,* T_k *of the rows of* $\mathbf{W}_k$ *by sampling* l *coordinates from* $\{1, 2, \ldots \tau\}$ *without replacement. Now denote* $\mathbf{SW}$ *as the sum of the subsampled rows*

$$\mathbf{SW} = \sum_{k=1}^{K} \left(\mathbf{S}_{T_k} \mathbf{W}_k\right).$$

Then

$$\sqrt{\frac{(1-\delta) l \cdot K}{n}} \leq \sigma_p(\mathbf{SW})$$

and

$$\sigma_1(\mathbf{SW}) \leq \sqrt{\frac{(1+\eta) l \cdot K}{n}}$$

with failure probability at most

$$p \cdot \left[\frac{e^{-\delta}}{(1-\delta)^{1-\delta}}\right]^{\alpha \log p} + p \cdot \left[\frac{e^{\eta}}{(1+\eta)^{1+\eta}}\right]^{\alpha \log p}$$

Proof. Define $\mathbf{w}_j^\top$ as the j^{th} row of $\mathbf{W}$ and $M := n \cdot \max_j \|\mathbf{w}_j\|^2$. Suppose $K = 2$ and consider the matrix

$$\begin{aligned}\mathbf{G}_2 : &= (\mathbf{S}_1\mathbf{W}_1 + \mathbf{S}_2\mathbf{W}_2)^\top(\mathbf{S}_1\mathbf{W}_1 + \mathbf{S}_2\mathbf{W}_2) \\ &= (\mathbf{S}_1\mathbf{W}_1)^\top(\mathbf{S}_1\mathbf{W}_1) + (\mathbf{S}_2\mathbf{W}_2)^\top(\mathbf{S}_2\mathbf{W}_2) \\ &\quad + (\mathbf{S}_1\mathbf{W}_1)^\top(\mathbf{S}_2\mathbf{W}_2) + (\mathbf{S}_2\mathbf{W}_2)^\top(\mathbf{S}_1\mathbf{W}_1).\end{aligned}$$

In general, we can express $\mathbf{G} := (\mathbf{SW})^\top(\mathbf{SW})$ as

$$\mathbf{G} := \sum_{k=1}^{K} \sum_{j \in T_k} \left(\mathbf{w}_j \mathbf{w}_j^\top + \sum_{k' \neq k} \sum_{j' \in T_k'} \mathbf{w}_j \mathbf{w}_{j'}^\top \right).$$

By the orthonormality of $\mathbf{W}$, the cross terms cancel as $\mathbf{w}_j \mathbf{w}_{j'}^\top = \mathbf{0}$, yielding

$$\mathbf{G} := (\mathbf{SW})^\top (\mathbf{SW}) = \sum_{k=1}^{K} \sum_{j \in T_k} \mathbf{w}_j \mathbf{w}_j^\top.$$

We can consider $\mathbf{G}$ as a sum of $l \cdot K$ random matrices

$$\mathbf{X}_1^{(1)}, \ldots, \mathbf{X}_1^{(K)}, \ldots, \mathbf{X}_l^{(1)}, \ldots, \mathbf{X}_l^{(K)}$$

sampled uniformly at random without replacement from the family $\mathcal{X} := \left\{\mathbf{w}_i \mathbf{w}_i^\top : i = 1, \ldots, \tau \cdot K\right\}$.

To use the matrix Chernoff bound in Lemma 6.7, we require the quantities $\mu_{\min}$, $\mu_{\max}$ and B. Noticing that $\lambda_{\max}(\mathbf{w}_j \mathbf{w}_j^\top) = \|\mathbf{w}_j\|^2 \leq \frac{M}{n}$, we can set $B \leq M/n$.

Taking expectations with respect to the random partitioning ($\mathbb{E}_P$) and the subsampling within each partition ($\mathbb{E}_S$), using the fact that columns of $\mathbf{W}$ are orthonormal we obtain

$$\mathbb{E}\left[\mathbf{X}_1^{(k)}\right] = \mathbb{E}_P \mathbb{E}_S \mathbf{X}_1^{(k)} = \frac{1}{K} \frac{1}{\tau} \sum_{i=1}^{K\tau} \mathbf{w}_i \mathbf{w}_i^\top = \frac{1}{n} \mathbf{W}^\top \mathbf{W} = \frac{1}{n} \mathbf{I}$$

Recall that we take l samples in K blocks so we can define

$$\mu_{\min} = \frac{l \cdot K}{n} \qquad \text{and} \qquad \mu_{\max} = \frac{l \cdot K}{n}.$$

Plugging these values into Lemma 6.7, the lower and upper Chernoff bounds respectively yield

$$\mathbb{P}\left\{\lambda_{\min}\left(\mathbf{G}\right) \leq (1-\delta)\frac{l \cdot K}{n}\right\}$$
$$\leq p \cdot \left[\frac{e^{-\delta}}{(1-\delta)^{1-\delta}}\right]^{l \cdot K/M} \quad \text{for } \delta \in [0,1), \text{ and}$$

$$\mathbb{P}\left\{\lambda_{\max}\left(\mathbf{G}\right) \geq (1+\delta)\frac{l \cdot K}{n}\right\}$$
$$\leq p \cdot \left[\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right]^{l \cdot K/M} \quad \text{for } \delta \geq 0.$$

Noting that $\lambda_{\min}(\mathbf{G}) = \sigma_p(\mathbf{G})^2$, similarly for $\lambda_{\max}$ and using the identity for $\mathbf{G}$ above obtains the desired result. □

For ease of reference, we also restate the Matrix Chernoff bound from Tropp (2011, 2010) but defer its proof to the original papers.

Lemma 6.7 (Matrix Chernoff from Tropp (2011)) *Let $\mathcal{X}$ be a finite set of positive-semidefinite matrices with dimension p, and suppose that*

$$\max_{\mathbf{A} \in \mathcal{X}} \lambda_{\max}(\mathbf{A}) \leq B$$

Sample $\{\mathbf{A}_1, \ldots, \mathbf{A}_l\}$ uniformly at random from $\mathcal{X}$ without replacement. Compute

$$\mu_{\min} = l \cdot \lambda_{\min}(\mathbb{E}\mathbf{X}_1) \qquad \textit{and} \qquad \mu_{\max} = l \cdot \lambda_{\max}(\mathbb{E}\mathbf{X}_1)$$

Then

$$\mathbb{P}\left\{\lambda_{\min}\left(\sum_i \mathbf{A}_i\right) \leq (1-\delta)\mu_{\min}\right\}$$
$$\leq p \cdot \left[\frac{e^{-\delta}}{(1-\delta)^{1-\delta}}\right]^{\mu_{\min}/B} \quad \textit{for } \delta \in [0,1), \ \textit{and}$$

$$\mathbb{P}\left\{\lambda_{\max}\left(\sum_i \mathbf{A}_i\right) \geq (1+\delta)\mu_{\max}\right\}$$
$$\leq p \cdot \left[\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right]^{\mu_{\max}/B} \quad \textit{for } \delta \geq 0.$$

Appendix 6.C Supplementary material for §6.5

Algorithm 9 DUAL-LOCO – cross validation

```
Input: Data: $\mathbf{X}$, $Y$, no. workers: $K$, no. folds: $v$
Parameters: $\tau_{subs}$, $\lambda_1, \dots \lambda_l$
Partition $\{p\}$ into $K$ subsets of equal size $\tau$ and distribute feature vectors in $\mathbf{X}$ accordingly over $K$ workers.
Partition $\{n\}$ into $v$ folds of equal size.
for each fold $f$ do
    Communicate indices of training and test points.
    for each worker $k \in \{1, \dots K\}$ in parallel do
        Compute and send $\mathbf{X}_{k,f}^{train} \mathbf{\Pi}_{k,f}$.
        Receive random features and construct $\bar{\mathbf{X}}_{k,f}^{train}$.
        for each $\lambda_j \in \{\lambda_1, \dots \lambda_l\}$ do
            $\tilde{\boldsymbol{\alpha}}_{k,f,\lambda_j} \leftarrow \texttt{LocalDualSolver}(\bar{\mathbf{X}}_{k,f}^{train}, Y_f^{train}, \lambda_j)$
            $\widehat{\boldsymbol{\beta}}_{k,f,\lambda_j} = -\frac{1}{n\lambda_j} \mathbf{X}_{k,f}^{train\top} \tilde{\boldsymbol{\alpha}}_{k,f,\lambda_j}$
            $\hat{Y}_{k,f,\lambda_j}^{test} = \mathbf{X}_{k,f}^{test} \widehat{\boldsymbol{\beta}}_{k,f,\lambda_j}$
            Send $\hat{Y}_{k,f,\lambda_j}^{test}$ to driver.
        end for
    end for
    for each $\lambda_j \in \{\lambda_1, \dots \lambda_l\}$ do
        Compute $\hat{Y}_{f,\lambda_j}^{test} = \sum_{k=1}^{K} \hat{Y}_{k,f,\lambda_j}^{test}$.
        Compute $\mathrm{MSE}_{f,\lambda_j}^{test}$ with $\hat{Y}_{f,\lambda_j}^{test}$ and $Y_f^{test}$.
    end for
end for
for each $\lambda_j \in \{\lambda_1, \dots \lambda_l\}$ do
    Compute $\mathrm{MSE}_{\lambda_j} = \frac{1}{v} \sum_{f=1}^{v} \mathrm{MSE}_{f,\lambda_j}$.
end for
Output: Parameter $\lambda_j$ attaining smallest $\mathrm{MSE}_{\lambda_j}$
```

Algorithm	K	TEST MSE	TRAIN MSE
Dual-Loco 0.5	12	0.0343 (3.75e-03)	0.0344 (2.59e-03)
Dual-Loco 0.5	24	0.0368 (4.22e-03)	0.0344 (3.05e-03)
Dual-Loco 0.5	48	0.0328 (3.97e-03)	0.0332 (2.91e-03)
Dual-Loco 0.5	96	0.0326 (3.13e-03)	0.0340 (2.67e-03)
Dual-Loco 0.5	192	0.0345 (3.82e-03)	0.0345 (2.69e-03)
Dual-Loco 1	12	0.0310 (2.89e-03)	0.0295 (2.28e-03)
Dual-Loco 1	24	0.0303 (2.87e-03)	0.0307 (1.44e-03)
Dual-Loco 1	48	0.0328 (1.92e-03)	0.0329 (1.55e-03)
Dual-Loco 1	96	0.0299 (1.07e-03)	0.0299 (7.77e-04)
Dual-Loco 2	12	0.0291 (2.16e-03)	0.0280 (6.80e-04)
Dual-Loco 2	24	0.0306 (2.38e-03)	0.0279 (1.24e-03)
Dual-Loco 2	48	0.0285 (6.11e-04)	0.0293 (4.77e-04)
CoCoA$^+$	12	0.0282 (4.25e-18)	0.0246 (2.45e-18)
CoCoA$^+$	24	0.0278 (3.47e-18)	0.0212 (3.00e-18)
CoCoA$^+$	48	0.0246 (6.01e-18)	0.0011 (1.53e-19)
CoCoA$^+$	96	0.0254 (5.49e-18)	0.0137 (1.50e-18)
CoCoA$^+$	192	0.0268 (1.23e-17)	0.0158 (6.21e-18)

Table 6.1.: Dogs vs Cats data: Normalized training and test MSE: mean and standard deviations (based on 5 repetitions).

Chapter 7.

Preserving privacy between features in distributed estimation

Privacy is crucial in many applications of machine learning. Legal, ethical and societal issues restrict the sharing of sensitive data making it difficult to learn from datasets that are partitioned between many parties. One important instance of such a distributed setting arises when information about each record in the dataset is held by different data owners (the design matrix is "vertically-partitioned").

In this setting few approaches exist for private data sharing for the purposes of statistical estimation and the classical setup of differential privacy with a "trusted curator" preparing the data does not apply. We work with the notion of (ϵ, δ)-distributed differential privacy which extends single-party differential privacy to the distributed, vertically-partitioned case. We propose PriDE, a scalable framework for distributed estimation where each party communicates perturbed random projections of their locally held features ensuring (ϵ, δ)-distributed differential privacy is preserved. For ℓ_2-penalized supervised learning problems PriDE has bounded estimation error compared with the optimal estimates obtained without privacy constraints in the non-distributed setting. We confirm this empirically on real world and synthetic datasets.

7.1. Introduction

Data driven personalization—from user experience on the web to medicine and healthcare—relies on aggregating a large amount of potentially sensitive data relating to individuals from disparate sources in order to answer statistical queries. Understandably, from a privacy perspective it may be

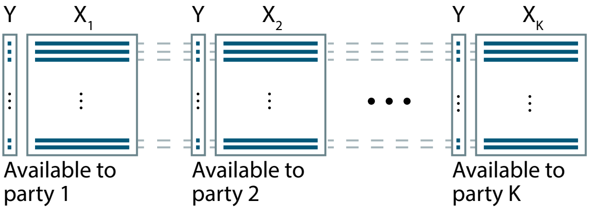


Figure 7.1.: Vertically partitioned data: Each party holds a subset of the total number of features, containing the data from the same set of individuals. Each party with access to Y can estimate $\boldsymbol{\beta}_k$.

undesirable—or even impossible—for such data to be shared in an undisguised form. For example, in healthcare and medical science applications, highly personal information is collected about individuals which can be invaluable for diagnosis, treatment and drug discovery. The use and sharing of such data is governed by relevant laws such as the Health Insurance Portability and Accountability Act (HIPAA) which typically only allow data to be shared if it has been de-identified (Sarwate et al., 2014). However, even after a dataset has been sanitized, the risk of subjects being re-identified is an ongoing concern and in many such cases privacy breaches actually occurred (El Emam et al., 2011).

Differential privacy (DP) (Dwork, 2006) constitutes a powerful theoretical framework for guaranteeing that the output of a suitable algorithm will not allow the identification of individuals in a dataset. Recently, it has been considered as a method of complying with the many regulations for sharing data in e.g. healthcare applications (Dankar and El Emam, 2013). Informally, a differentially private algorithm is one that ensures information identifying an individual cannot be learned from the output of that algorithm on two datasets which differ only by that individual. (Many definitions with subtle differences are used; we will formally state a definition for our purposes in §7.4.) In case of supervised learning, research has mainly focused on ensuring that a model estimated in the single party setting can be publicly released (Chaudhuri et al., 2011). However, in many application areas where sensitive data is held by several parties—e.g. health informatics, risk modeling and computational social science (D'Orazio et al., 2015)—estimating a model and performing statistical inference, rather than coefficient release, is often the stated goal. Therefore, an important open question concerns how sensitive data can be shared among different parties in a distributed computation framework to opti-

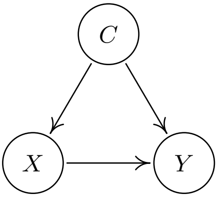


Figure 7.2.: Scenario (A): The data held by some parties is sensitive while the data held by other parties is public and not sensitive (e.g. Burkhardt et al. (2015)). It is possible to publish coefficients of the public blocks—here, corresponding to the variables X—while still accounting for confounding effects of the private blocks—here corresponding to the variables in C. The confounders in C influence both the variables in X as well as the response Y.

mize a global statistical learning objective.

Summary of contributions. In §7.2 we formally introduce the problem setting—statistical estimation where *sensitive* data is partitioned *vertically* between multiple parties—and describe some of the unique challenges in this setting. In §7.3 we propose PriDE (PRIvate Distributed Estimation), a scalable algorithm for differentially private statistical estimation when the data are partitioned vertically among multiple parties. Our key insight is that to ensure privacy, we require a small algorithmic change to the recently proposed Dual-Loco framework (Heinze et al., 2016). In §7.4, we show the following theoretical properties of PriDE:

§7.4.1 **Privacy**: PriDE preserves (ϵ, δ)-distributed differential privacy (cf. Definition 7.2).

§7.4.2 **Utility**: The estimation error of PriDE with respect to the optimal coefficients (estimated in the non-distributed setting under no privacy constraints) is bounded.

The second main contribution is an extensive evaluation of the empirical behavior of PriDE on a variety of simulated and real datasets in §7.5 and §7.D. We observe that PriDE improves upon a fully-private baseline which avoids communicating any data between parties and quickly approaches the performance of the optimal solution. Related work is discussed in §7.6.

7.2. Problem setting

In this work, we are interested in objectives of the form

$$\min_{\mathbf{b}\in\mathbb{R}^p} \left\{ J(\mathbf{b}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{b}^\top \mathbf{x}_i) + \frac{\lambda}{2}\|\mathbf{b}\|^2 \right\} \tag{7.1}$$

where $\lambda > 0$ is the regularization parameter. The loss functions $f_i(\mathbf{b}^\top \mathbf{x}_i)$ depend on a response $y_i \in \mathbb{R}$ and linearly on the coefficients, $\mathbf{b} \in \mathbb{R}^p$ through a vector of covariates, $\mathbf{x}_i \in \mathbb{R}^p$. Furthermore, we assume all f_i to be convex and smooth with Lipschitz continuous gradients. For example when $f_i(\mathbf{b}^\top \mathbf{x}_i) = (y_i - \mathbf{b}^\top \mathbf{x}_i)^2$, Eq. (7.1) corresponds to ridge regression; for logistic regression $f_i(\mathbf{b}^\top \mathbf{x}_i) = \log(1 + \exp(-y_i \mathbf{b}^\top \mathbf{x}_i))$. Let $\boldsymbol{\beta}$ denote the true underlying coefficients of interest.

In the multi-party setting where the data are vertically partitioned, each party k has some proportion of the features corresponding to all of the observations (cf. Figure 7.1). Given a design matrix $\mathbf{X} \in \mathbb{R}^{n\times p}$ whose rows are $\mathbf{x}_i^\top$, each party holds a disjoint subset of the p available features, $\mathcal{P}_1, \ldots, \mathcal{P}_K$ of size $\tau = p/K$ belonging to the same observations (in general the partitions can be of different sizes). Throughout, we assume that the columns of $\mathbf{X}$ are normalized to have mean zero and unit variance. Let $\mathbf{X}_k \in \mathbb{R}^{n\times\tau}$ be the sub-matrix whose columns correspond to the coordinates in $\mathcal{P}_k$. The set $\mathcal{P}_{-k}$ contains all coordinates not in $\mathcal{P}_k$. Each party aims to estimate $\boldsymbol{\beta}_k \in \mathbb{R}^\tau$, the portion of the true underlying parameter vector $\boldsymbol{\beta}$ corresponding to the features it holds, *while accounting for the contribution of the features held by the remaining parties.* However, due to privacy concerns the parties are not allowed to share their locally-held features. This scenario is of particular interest in healthcare and biomedicine (Li et al., 2015; Ohno-Machado, 2012; Que et al., 2012; Wu et al., 2012) but also in customer profiling and personalization.

Example scenarios. Here we briefly outline (non-exhaustively) two special cases of the general problem setting which cover a wide range of possible use cases—in particular in medical analyses—where a thorough accounting of confounding factors requires a mixture of public and private data to be aggregated.

(A) The data held by some parties is sensitive while the data held by other parties is public and not sensitive (e.g. Burkhardt et al. (2015)).

It is possible to publish coefficients of the public blocks while still accounting for possible confounding effects of the private blocks.

(**B**) The response is not known to all parties. Then coefficients are only estimated for the blocks which know the response. The remaining parties just provide their data in secure form.

A concrete example of (**A**) is described graphically in Figure 7.2. Consider the response Y being a variable measuring a cancer patient's health. Both genomic factors (contained in the set C) as well as gene expressions (in set X) have an influence on Y. In turn, genomic factors affect gene expressions. It is impossible to conduct a randomized study to estimate the effect of X on Y because gene expressions cannot be randomized. Additionally, due to its highly personal and sensitive nature, genomic data is rarely publicly available so C and X are stored separately (i.e. the full design matrix is vertically partitioned as in Burkhardt et al. (2015)). Due to the confounding links between C and X, only including gene expressions in the model can result in heavily biased estimates for the effect of X on Y (e.g. Pearl, 2009). Conducting studies that offer a holistic view on the factors influencing the response—as opposed to relying on biased estimates resulting from marginal studies—is tremendously important. However, it is an open question how to estimate the full model while providing formal privacy guarantees on the data sharing mechanism.

7.3. The PRIDE algorithm

In this section we propose PRIDE, a scalable low-communication algorithm which extends the LOCO framework (Heinze et al., 2016) for distributed estimation to the *private* setting. Key to the PRIDE algorithm is the data sharing mechanism. The schematic is given in Figure 7.3. The full procedure is presented in Algorithm 10. We explain the following steps in more detail:

In **Step 2**, we compute the random features $(\mathbf{X}_k\mathbf{\Pi}_k) \in \mathbb{R}^{n\times\tau_{subs}}$. $\mathbf{\Pi}_k \in \mathbb{R}^{\tau\times\tau_{subs}}$ is the subsampled randomized Hadamard transform (SRHT) matrix which admits fast matrix-vector products (Tropp, 2011). We then perturb this by a Gaussian random matrix $\mathbf{W}_k \in \mathbb{R}^{n\times\tau_{subs}} \sim \mathcal{N}(0,\sigma_k^2\mathbf{I})$ to get $\widehat{\mathbf{Z}}_k = \mathbf{X}_k\mathbf{\Pi}_k + \mathbf{W}_k$. The exact form of σ_k is given in Theorem 7.3.

In **Step 4**, the matrices of random features are communicated. For ease

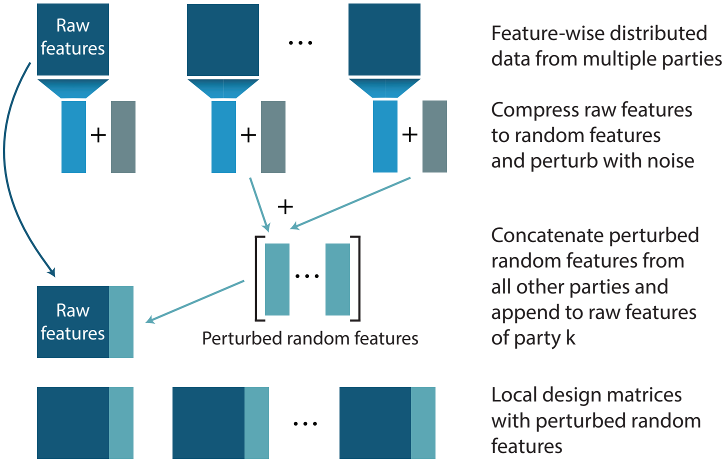

Raw features
...
+
+
+
+
...
Perturbed random features
Raw features
...
Feature-wise distributed data from multiple parties
Compress raw features to random features and perturb with noise
Concatenate perturbed random features from all other parties and append to raw features of party k
Local design matrices with perturbed random features

Figure 7.3.: PRIDE's distributed private data sharing mechanism.

of notation, let $\tau_K = (K-1)\tau_{subs}$. Party k then constructs the matrix

$$\bar{\mathbf{X}}_k \in \mathbb{R}^{n\times(\tau+\tau_K)} = \left[\mathbf{X}_k, \left[\widehat{\mathbf{Z}}_{k'}\right]_{k'\neq k}\right], \tag{7.2}$$

which is the column-wise concatenation of party k's raw features and the perturbed random features from all other parties.

In **Step 5** each party solves the following local dual optimization problem

$$\tilde{\boldsymbol{\alpha}}_k = \underset{\boldsymbol{\alpha}\in\mathbb{R}^n}{\operatorname{argmax}} - \sum_{i=1}^{n} f_i^*(\alpha_i) - \frac{1}{2n\lambda}\boldsymbol{\alpha}^\top \bar{\mathbf{X}}_k \bar{\mathbf{X}}_k^\top \boldsymbol{\alpha}, \tag{7.3}$$

where f^* is the conjugate Fenchel dual of f. For example, for squared loss functions $f_i(u) = \frac{1}{2}(y_i - u)^2$, we have $f_i^*(\alpha) = \frac{1}{2}\alpha^2 + \alpha y_i$. This is solved using e.g. SDCA (Shalev-Shwartz and Zhang, 2013).

The main difference to DUAL-LOCO is the perturbation of the random features in **Step 2**. Although a small algorithmic difference, this has important consequences for the analysis which we present in the following section.

Algorithm 10 PRIDE

```
Input: $Y$, $\mathbf{X}$ vertically-partitioned over $K$ parties, $\tau_{subs}$, $\lambda$, $\epsilon$, $\delta$
for each party $k \in \{1, \dots K\}$ in parallel do
    Compute perturbed random features
    $\widehat{\mathbf{Z}}_k = \mathbf{X}_k \mathbf{\Pi}_k + \mathbf{W}_k$.
    Communicate $\widehat{\mathbf{Z}}_k$ to all parties $k'$ where $k' \neq k$.
    Construct local design matrix $\bar{\mathbf{X}}_k$.
    $\tilde{\boldsymbol{\alpha}}_k \leftarrow$ LocalDualSolver$(\bar{\mathbf{X}}_k, Y, \lambda)$
    $\widehat{\boldsymbol{\beta}}_k = -\frac{1}{n\lambda} \mathbf{X}_k^\top \tilde{\boldsymbol{\alpha}}_k$
end for
Output: Each party $k$ obtains $\widehat{\boldsymbol{\beta}}_k$.
```

7.4. Analysis

For the discussion which follows we use the following definition of privacy which is concerned with changes in the attribute values of the observations rather than the difference in observations[1].

Definition 7.1 **($(\epsilon, \delta, \mathcal{S})$-differential privacy)** *A randomized algorithm* ALG *satisfies $(\epsilon, \delta, \mathcal{S})$-differential privacy if for all inputs $\mathbf{X}$ and $\mathbf{X}'$ differing in at most one user's one attribute value of an attribute in $\mathcal{S} \subseteq \{1, \dots, p\}$, and for all sets of possible outputs $\mathcal{D} \subseteq$ range(*ALG*)*

$$\mathbb{P}\left[\textsc{Alg}(\mathbf{X}) \in \mathcal{D}\right] \leq e^{\epsilon} \ \mathbb{P}\left[\textsc{Alg}(\mathbf{X}') \in \mathcal{D}\right] + \delta \tag{7.4}$$

where the probability is computed over the randomness of the algorithm.

When $\mathcal{S} = \{1, \dots, p\}$, $(\epsilon, \delta, \mathcal{S})$-differential privacy reduces to (ϵ, δ)-differential privacy. Informally, this states that (up to the parameters of the differential privacy guarantee) an adversary cannot infer a single attribute value for a single observation of an attribute in $\mathcal{S}$ from the output of the algorithm *despite knowing the values of all other attributes for all other observations*. In the following definition, we use Definition 7.1 to formulate differential privacy in the distributed setting. The definition is close to Definition 2.4 in Beimel et al. (2008); here, we state it in our notation and for the case when $\delta > 0$.

---

[1] Many definitions with subtle differences are used; here, we follow the definition used in Kenthapadi et al. (2013).

Definition 7.2 ((ϵ, δ)-distributed differential privacy) *A randomized algorithm* ALG *satisfies* (ϵ, δ)*-distributed differential privacy, if* ALG *satisfies* $(\epsilon, \delta, \mathcal{S})$*-differential privacy for all* $\mathcal{S} \in \{\mathcal{P}_{-k}; k = 1, \ldots, K\}$ *where* $\mathcal{P}_{-k}$ *is the set of indices corresponding to the features* non-local *to party* k*.*

A randomized algorithm ALG is $(\epsilon, 0)$-distributed differentially private if Definition 2.4 in Beimel et al. (2008) is fulfilled for $t = \max_k |\mathcal{P}_k|$. The condition in Beimel et al. (2008) is a bit stricter than ours as it requires $(\epsilon, \delta, \mathcal{S})$-differential privacy for all sets $\mathcal{S}$ with $|\mathcal{S}^c| \leq t$ and not just for $\mathcal{P}_{-k}$ with $k = 1, \ldots, K$ as we do here. We also want to allow for $\delta > 0$ with Definition 7.2.

PRIDE achieves (ϵ, δ)-distributed differential privacy by perturbing random features with Gaussian noise before communicating them. As detailed in §7.4.1, this procedure preserves differential privacy according to Definition 7.2. While perturbing the random features has an adverse effect on the accuracy of the coefficient estimates, we prove an upper bound on the coefficient estimation error in §7.4.2. The error bound shows an interesting trade-off between the desired level of privacy and the accuracy of the random feature representation.

7.4.1. Distributed privacy guarantee

Theorem 7.3 (Adapted from Kenthapadi et al. (2013).) *Let* $w_2(\mathbf{\Pi}_k)$ *denote the maximum* ℓ_2*-norm of any row in the projection matrix* $\mathbf{\Pi}_k$ *and let the range of the columns of* $\mathbf{X}_k$ *be upper bounded by* θ_k*.* $\mathcal{P}_{-k}$ *is the set of indices corresponding to the features* non-local *to party* k*. Assuming* $\delta < \frac{1}{2}$*, let the entries of party* k*'s noise matrix* $\mathbf{W}_k$ *be drawn from* $\mathcal{N}(0, \sigma_k^2 \mathbf{I})$ *with*

$$\sigma_k > \frac{w_2(\mathbf{\Pi}_k) \cdot \theta_k}{\epsilon} \sqrt{2(\ln(1/2\delta) + \epsilon)}.$$

Then PRIDE *satisfies* (ϵ, δ)*-distributed differential privacy.*

The proof follows by adapting Kenthapadi et al. (2013) to hold for (ϵ, δ)-distributed differential privacy. When $\mathbf{\Pi}_k$ is the SRHT, $w_2(\mathbf{\Pi}_k) = 1$. Theorem 7.3 guarantees that an adversary who has access to the data held by party k and knows all values of all attributes for every individual except for a single non-locally stored attribute value cannot infer that value from the perturbed random features which have been communicated to party k. This ensures PRIDE fulfills Definition 7.2. In contrast to the Laplace mechanism, the use of the Gaussian mechanism has the advantage that the required noise level is independent of the dimension of the projection matrix.

7.4.2. Approximation error of PRIDE

We now bound the coefficient approximation error between the PRIDE solution and the optimal solution to Eq. (7.1).

Assumption 7.4 *Letting r denote the rank of $\mathbf{X}$ and $\tau_K = (K-1)\tau_{subs}$, we require the following conditions to hold:*

(A1) The projection dimension is chosen such that $\tau_K \gtrsim r \log r$.
(A2) The problem is high-dimensional, i.e. $n \leq p$, and $r = n$.

Theorem 7.5 (PRIDE approximation guarantee) *Assume all f_i in Eq. (7.1) to be convex and smooth with Lipschitz continuous gradients. Under Assumption 7.4 the overall error between the optimal solution to Eq. (7.1) $\boldsymbol{\beta}^*$ and the solution returned by PRIDE $\widehat{\boldsymbol{\beta}}$ is bounded with probability at least $1 - K\zeta$ by*

$$\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\| \leq \underbrace{\frac{\sqrt{K}\rho}{(1-2\rho)}\|\boldsymbol{\beta}^*\|}_{\textbf{(i)}} + \underbrace{\frac{\sqrt{K}\rho}{(1-2\rho)}\frac{\sigma}{d_{min}}\left(2 + \frac{\sigma\tau_K + \sigma\tau_K^2}{d_{min}}\right)\|\boldsymbol{\beta}^*\|}_{\textbf{(ii)}} \tag{7.5}$$

where $\rho = C\sqrt{\frac{r\log(2r/\xi)}{\tau_K}}$, $\sigma = \max_k \sigma_k$ and $d_{min} = d_r(\mathbf{X})$, the smallest non-zero singular value of $\mathbf{X}$. C and ξ are absolute positive constants. The exact form of ζ is given in §7.A.

Proof strategy. (Full details are given in §7.A.) We require to bound the local coefficient estimation error of a single party k which can then be combined with a union bound to obtain the global approximation error. To bound the local error (Theorem 7.6), a key step is bounding the difference between the full (non-perturbed, single-party) kernel matrix $\mathbf{K}$ and the projected-and-perturbed kernel matrix $\tilde{\mathbf{K}}$ (omitting the subscript k for ease of notation) where

$$\begin{aligned}
&\mathbf{K} = \mathbf{X}\mathbf{X}^\top \qquad \text{and} \qquad \tilde{\mathbf{K}} = (\mathbf{X}\Theta + \mathbf{E})(\mathbf{X}\Theta + \mathbf{E})^\top, \\
&\Theta = \operatorname{diag}(\mathbf{I}_\tau, \mathbf{\Pi}_1, \dots, \mathbf{\Pi}_{K-1}) \in \mathbb{R}^{p \times (\tau + \tau_K)} \quad \text{and} \\
&\mathbf{E} = \begin{bmatrix} \mathbf{0}_\tau & \mathbf{W}_1 & \dots & \mathbf{W}_{K-1} \end{bmatrix} \in \mathbb{R}^{n \times (\tau + \tau_K)}.
\end{aligned}$$

When privacy is not required, $\sigma = 0$ and $\mathbf{E} = \mathbf{0}$ in which case we recover the approximation guarantee of Dual-Loco which relies on the fact that $\|\mathbf{K} - \tilde{\mathbf{K}}\| \leq \rho$ (Heinze et al., 2016). However, this bound does not hold when i.i.d. Gaussian noise is added to those entries of $\mathbf{X}\Theta$ corresponding to the random features (i.e. $\sigma > 0$ and $\mathbf{E} \neq \mathbf{0}$). Now, we require to find an upper bound on $\|\mathbf{K} - \tilde{\mathbf{K}}\|$ and the proof also requires a lower bound on $\|\tilde{\mathbf{K}}\|$. We can bound $\|\mathbf{K} - (\mathbf{X}\Theta)(\mathbf{X}\Theta)^\top\| \leq \rho$ and use Lemma 7.10 to bound the terms involving $\mathbf{E}$ with high probability. While the exact expressions are more involved, intuitively, in expectation the cross terms are zero while the diagonal elements of $\mathbf{E}\mathbf{E}^\top$ are at most $\sigma^2 \tau_K$. Finally, lower bounding $\tilde{\mathbf{K}}$ requires a different technique as the involved cross terms are not positive semidefinite. Using that terms involving $\mathbf{E}$ are centered around 0 and applying a Chernoff bound (Lemma 7.11) allows us to show $\|\tilde{\mathbf{K}}\| \geq 1 - 2\rho$. Full details are given in §7.A. □

Discussion. The bound in Theorem 7.5 consists of two terms: **(i)** The approximation error due to the (distributed) random projection representation. This decreases as the projection dimension τ_{subs} increases, providing a more accurate approximation to the non-local features. **(ii)** The error due to the perturbation necessary for guaranteeing privacy. This term is increasing in τ_{subs}—a larger dimensional random feature representation contributes more noisy dimensions which act like an additional ℓ_2-regularizer. This can be seen clearly when comparing the solutions to the dual formulation of the ridge regression objective: The optimal solution is given by $\boldsymbol{\alpha}^* = (\mathbf{K} + \lambda \mathbf{I})^{-1} Y$ while party k computes $\tilde{\boldsymbol{\alpha}} = ((\mathbf{X}\Theta + \mathbf{E})(\mathbf{X}\Theta + \mathbf{E})^\top + \lambda \mathbf{I})^{-1} Y$. The diagonal elements of $\mathbf{E}\mathbf{E}^\top$ are centered around $\sigma^2 \tau_K = \sigma^2 (K-1) \tau_{subs}$, so using a larger projection dimension τ_{subs} increases the regularizing effect (and therefore bias) induced by $\mathbf{E}\mathbf{E}^\top$ which acts in addition to the one caused by λ. In §7.C we show that for the primal formulation of the least-squares objective, the effect of $\mathbf{E}$ can be understood as an ℓ_2-regularizer which acts on the random

Table 7.1.: Data set statistics: $\max_k(\theta_k)$ is the largest bound on the range of the columns of $\mathbf{X}_k$ among all parties, $d_{\min}$ is the smallest non-zero singular value of the design matrix, $r_{\text{eff}}(\mathbf{X})$ denotes the effective rank and J_u denotes the number of principal components that capture $u\%$ of the variance in the data set.

	n_{train}	p	$\max_k(\theta_k)$	$d_{\min}$	$r_{\text{eff}}(\mathbf{X})$	J_{80}	J_{90}
SYNTHETIC	800	400	7.41	$3.7e$-6	2.03	3	5
CLIMATE	849	$10,368$	8.51	3.32	4.03	29	54
CANCER	188	$2,000$	10.92	11.57	4.53	65	107

features only. On the other hand, the bias can be decreased by increasing ϵ (decreasing σ) implying a weaker privacy guarantee.

We thus observe a trade-off between approximation quality and privacy. When a very strong privacy guarantee is required—implying a large value of σ—a smaller τ_{subs} should be chosen so that the additional regularization does not become too strong. On the other hand, if the privacy requirements are less stringent, a larger τ_{subs} together with a larger ϵ will yield better approximation quality. In general, PRIDE will be most effective when the rank of the problem is such that a relatively small projection dimension will capture most of the important structure in the data. We demonstrate the effect of this trade-off empirically in the following section. Importantly, we shall see that the induced bias that results from not communicating any data is often much larger than the bias of the PRIDE estimates.

7.5. Experiments

We present results on three datasets summarized in Table 7.1: results for simulated data and an application from climate science are presented in §7.5.1 and §7.5.2, respectively. A gene expression dataset is analyzed in §7.D.1. Table 7.1 also contains the smallest non-zero singular value of the design matrix, $d_{\min}$; the effective rank $r_{\text{eff}}(\mathbf{X}) = \operatorname{tr}\left(\mathbf{X}^\top\mathbf{X}\right)/\|\mathbf{X}\|^2$ and the largest bound on the range of the columns of $\mathbf{X}_k$ among all parties, $\max_k(\theta_k)$. Informally, the effective rank (Vershynin, 2010) is a measure of the intrinsic dimension of a matrix which captures whether the matrix lies near to a low-dimensional subspace. We compare the performance of five methods:

(a) "Semi-Naive Bayes" (NB). Here, a separate model is learned by each party independently:

$$\widehat{\boldsymbol{\beta}}_k^{\text{NB}} = \text{argmin}_{\mathbf{b}_k} \sum_{i=1}^n f_i(\mathbf{x}_k^\top \mathbf{b}_k) + \lambda \|\mathbf{b}_k\|^2.$$

Since no data is communicated, the features are kept completely private.

(b) The standard Dual-Loco algorithm (corresponding to PriDE with $\sigma_k = 0 \; \forall k$). Since the random features are not perturbed this does not guarantee privacy according to Theorem 7.3.

(c) Our proposed PriDE algorithm. We show the effect of varying the privacy parameter ϵ by varying the noise variance σ_k^2. We fix $\delta = 0.05$ as varying δ has only little effect on σ_k. As σ_k also depends on the maximal range of the columns of $\mathbf{X}_k$, we report the maximum of σ_k for $k = 1, \ldots, K$ in Table 7.3.

(d) In the non-distributed setting: Glmnet (Friedman et al., 2010) and SDCA (Shalev-Shwartz and Zhang, 2013).

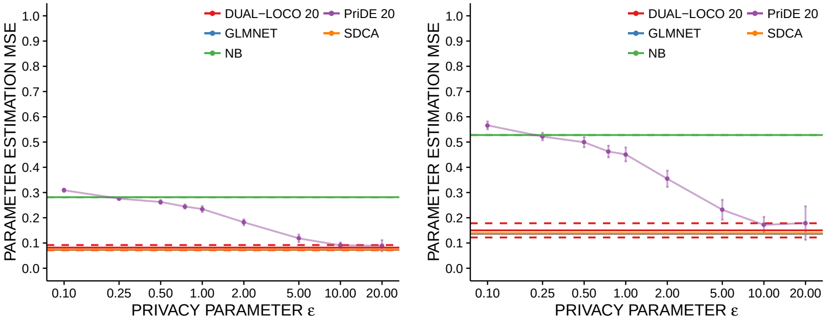
For both Dual-Loco and PriDE we show results for different values of the projection dimension τ_{subs}. The absolute dimensions are given in Table 7.2. Details on the cross validation procedure are given in §7.E.

7.5.1. Simulated data

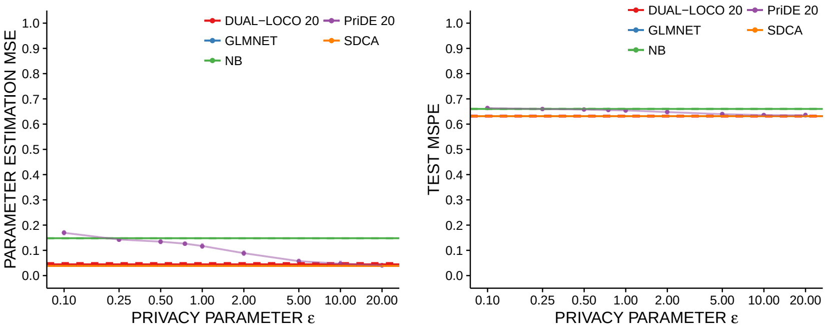
We revisit example (**A**) given in §7.2. The data are simulated according to the model in Figure 7.2. Full simulation details are given in §7.D.3[2]. We consider two blocks of features, C and X. For example, C could contain genomic data such as measurements of single nucleotide polymorphisms (SNPs). Due to its highly personal and sensitive nature, genomic data arising from techniques like SNP genotyping is rarely publicly available. The other block, X could hold gene expression data. Some of the genomic features have an effect on some of the gene expression features and both sets of features contribute to the response Y. We distribute the two blocks of features over $K = 2$ parties so that X and C are kept separately. In this experiment we aim to analyze the parameter estimation error with respect to the *true underlying coefficients* $\boldsymbol{\beta}$. Due to the dependence between C

---

[2] The data generating code is available at `https://onlinelibrary.wiley.com/doi/abs/10.1002/sta4.189`.



(a) $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|^2/\|\boldsymbol{\beta}\|^2$ (b) $\|\widehat{\boldsymbol{\beta}}_X - \boldsymbol{\beta}_X\|^2/\|\boldsymbol{\beta}_X\|^2$



(c) $\|\widehat{\boldsymbol{\beta}}_C - \boldsymbol{\beta}_C\|^2/\|\boldsymbol{\beta}_C\|^2$ (d) Test MSPE

Figure 7.4.: Simulated data: Results for projection dimension $\tau_{subs} = 0.2\tau$. Parameter estimation errors are computed w.r.t. the data generating model. Additional plots for $\tau_{subs} = \{0.05, 0.1\} \cdot \tau$ can be found in Figures 7.11–7.12 in the supplementary information.

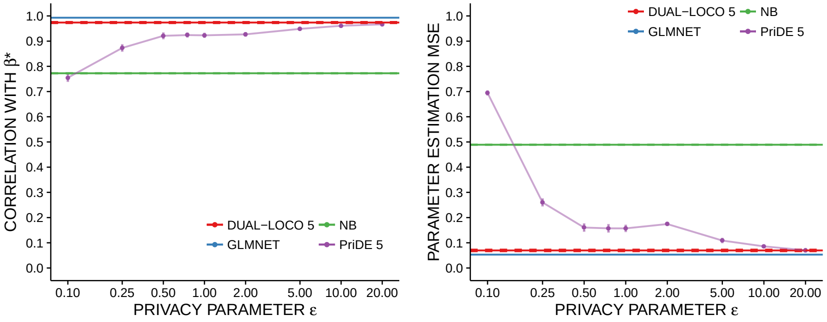
and X one cannot obtain accurate coefficient estimates for the effect of X on Y when only including X into the model. We aim to assess whether the perturbed random projections used by PriDE suffice to communicate enough information to obtain accurate estimates in this challenging estimation task.

Comparisons of normalized coefficient estimation error with respect to the data generating coefficients $\boldsymbol{\beta}$ are shown for $\tau_{subs} = 0.2\tau$ in Figure 7.4(a)-(c). There is a significant difference between the NB and Dual-Loco and SDCA solutions, particularly for block X. This performance gap is to be expected due to the confounding effect of C. It shows that in order to obtain accurate coefficient estimates in the distributed setting some degree of communication is crucial which allows to adjust for the dependencies between the features. For small ϵ (more privacy) PriDE performs similarly to NB, i.e. the incurred biases are on the same scale. As ϵ increases, PriDE approaches and eventually equals the performance of Dual-Loco and SDCA. This demonstrates that PriDE is able to approximate the true $\boldsymbol{\beta}$ accurately for sufficiently large values of ϵ. Thus PriDE allows to adjust for the confounding effects from C on X while guaranteeing $(\epsilon, \delta, \mathcal{S})$-differential privacy.

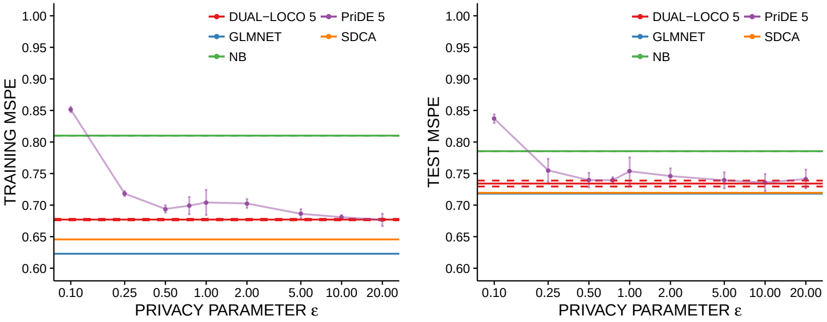
Figure 7.4d shows the normalized prediction MSE on the test set. All methods perform similarly. Due to the confounding effect of C and X, NB is unable to obtain accurate coefficient estimates but it can achieve good predictive performance in this example. This experiment also suggests that Assumption 7.4 can be weakened to settings where the effective rank of the data is low while $n > p$. Different proof techniques would be required to extend Thereom 7.5 to such cases.

7.5.2. Climate model data

Next, we present an application to a problem in climate modeling. We consider data from part of the CMIP5 climate modeling ensemble which are taken from control simulations of the GISS global circulation model (Schmidt et al., 2014). We aim to forecast the monthly global average temperature Y in February using the air pressure measured in January. The features are pressure measurements taken at $p = 10,368$ geographic grid points. The model simulates the climate for a range of 531 years and we use the output from two control simulation runs. The data set is split into training ($n_{\text{train}} = 849$) and test set ($n_{\text{test}} = 213$), and we distribute the problem across $K = 4$ parties.



(a) $\text{corr}(\widehat{\boldsymbol{\beta}}, \boldsymbol{\beta}^*)$ (b) $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|^2 / \|\boldsymbol{\beta}^*\|^2$



(c) Training MSPE (d) Test MSPE

Figure 7.5.: Climate model data: Results for projection dimension $\tau_{subs} = 0.05\tau$. The parameter estimation metrics are computed w.r.t. the optimal single-machine solution $\boldsymbol{\beta}^*$ obtained with SDCA. Additional plots for $\tau_{subs} = \{0.01, 0.1, 0.2\} \cdot \tau$ can be found in Figures 7.13–7.15.

Comparisons of correlation and estimation error, global training and test error (all normalized) are shown for $\tau_{subs} = 0.05\tau$ in Figure 7.5. Since the true coefficients $\boldsymbol{\beta}$ are unknown in this example, comparisons of correlation and estimation error are computed with respect to the empirical risk minimizer, i.e. the optimal parameters $\boldsymbol{\beta}^*$, estimated using SDCA where all the data was available, non-perturbed, on a single machine. The parameter estimation error is the quantity which is bounded in Theorem 7.5.

Importantly, there is a significant difference between the NB and Dual-Loco solutions. This performance gap shows that in the distributed setting some degree of communication is crucial for good statistical estimation and predictive accuracy for this problem as not communicating any features incurs a large bias. In Figures 7.13–7.15 we observe that increasing τ_{subs} does not cause a large change in the accuracy achieved by Dual-Loco. This suggests that the problem is nominally low rank and a small projection dimension suffices to capture the structure of the data. This is to be expected given the high degree of spatial correlation of pressure measurements and is confirmed by the estimate of the effective rank and the PCA statistics in Table 7.1. For reasonable values of ϵ, the PriDE solution quickly approaches the Dual-Loco solution for all four measures of accuracy. Importantly, PriDE achieves a test prediction error within the margin of error of the Dual-Loco prediction error.

We observe the trade-off implied by Theorem 7.5: As the projection dimension increases, the Dual-Loco approximation error decreases (i.e. term (i) in Eq. (7.5)). However, term (ii) grows with $\sigma^2(K-1)\tau_{subs}$. For very small values of ϵ this second contribution dominates so a smaller projection dimension typically yields better performance. As ϵ increases, the gain in approximation quality starts to outweigh the regularization bias incurred by increasing τ_{subs}, so that for large values of ϵ a large projection dimension performs best. In Figures 7.13–7.15, this trade-off is reflected in a slower convergence to the Dual-Loco solution for larger values of τ_{subs}.

In summary, the behavior predicted by Theorem 7.5 is confirmed empirically. The best performance of PriDE can be obtained by finding the optimum of the accuracy-privacy trade-off, respecting the problem-specific constraints on privacy. That is, by choosing a projection dimension τ_{subs} that suffices to capture the signal contained in the non-local features, so that term (i) in Eq. (7.5) is as small as possible without over-regularizing the objective and introducing a large bias from term (ii). Finding the optimal projection dimension is then a problem of model selection. We discuss the challenges of a privacy preserving cross validation scheme in §7.E.

In general, given a suitable projection dimension, PriDE can significantly improve upon the NB solution: the bias of the NB estimates is often much larger than the bias of the PriDE estimates. This suggests that the PriDE framework allows for accurate distributed statistical estimation while guaranteeing (ϵ, δ)-distributed differential privacy.

7.6. Related work

Privacy-aware learning. Ensuring differential privacy in supervised learning techniques has garnered increasing interest in recent years (Bassily et al., 2014; Chaudhuri and Monteleoni, 2009; Chaudhuri et al., 2011; Kasiviswanathan et al., 2008; Sheffet, 2017) and approaches have been proposed to solve more general convex optimization problems in a private fashion (Song et al., 2013). Many of these approaches achieve privacy by either applying noise to the coefficient vector before it is returned or perturbing the objective with noise during optimization.

Kenthapadi et al. (2013) apply a Johnson-Lindenstrauss random projection to compress the column space of the design matrix and perturb the resulting matrix with Gaussian noise. This procedure allows the compressed, perturbed data matrix to be published but forfeits the interpretability of the features as any subsequent queries must be performed in the compressed space. This approach is related to *local privacy* (Duchi et al., 2013b; Smith et al., 2017) where the algorithm only observes a disguised version of the data.

Distributed statistical estimation. Distributed estimation and optimization when the data is horizontally partitioned has been a popular topic in recent years (Jaggi et al., 2014; Zhang et al., 2015b). However, the problem of statistical estimation when the data is vertically partitioned has been less well studied since most loss functions of interest are not separable across coordinates. A key challenge addressed by Heinze et al. (2014) and Heinze et al. (2016) was to define a local minimization problem for each worker to solve *independently* while still maintaining important dependencies between features held by different parties. This is achieved by communicating low-dimensional random projections of the data held by each party which keeps communication overhead low. Although this obfuscates the data to some degree, it does not guarantee privacy.

Preserving privacy in distributed learning. McGregor et al. (2010) introduce the notion of two-party differential privacy which is generalized to an arbitrary number of parties in Beimel et al. (2008). We discuss the relation to Definitions 7.1 and 7.2 in §7.4.1.

In distributed supervised learning, private approaches have been much less studied and focus mainly on the setting where data is horizontally partitioned (Huang et al., 2015; Zhang and Zhu, 2016). Few approaches have been considered for privacy preserving learning in the distributed setting when the data is partitioned *vertically* (Mangasarian et al., 2008; Mohammed et al., 2014; Wu et al., 2012; Yu et al., 2006). However, no formal guarantees with respect to both privacy and utility are given.

7.7. Conclusions and further work

We have proposed PriDE which addresses some of the important concerns in learning from sensitive, vertically partitioned data in a principled and scalable way. PriDE preserves (ϵ, δ)-distributed differential privacy while maintaining a low approximation error with respect to the optimal, non-private, non-distributed model. To the best of our knowledge, no other methods with similar guarantees have been proposed for the considered problem setting.

PriDE only communicates perturbed low-dimensional random projections of the original features so the communication overhead is small. Since estimation is performed on a combination of raw and random features, the solution is returned in the original space preserving interpretability of the coefficients. This allows to assess a feature's impact on the response while accounting for the contribution of—possibly confounding—sensitive features held by other parties. For prediction tasks, each party can use its own local design matrix, consisting of raw and perturbed random features.

Empirically, we have shown on simulated and real-world datasets that the PriDE estimates greatly improve upon the performance of the fully-private semi-Naive Bayes model and approach (i) the true underlying coefficients, and (ii) the estimates of the non-private and non-distributed Glmnet and SDCA solvers. PriDE also performs well in areas not specifically covered by Theorem 7.5, as shown for the low-dimensional synthetic data. This suggests that our result could be generalized to when the effective rank of the problem is small.

Perturbing the random features is necessary to preserve privacy but adds bias to the solution. An open question concerns whether recent approaches to errors-in-variables regression (Loh and Wainwright, 2012) could be used to obtain an unbiased solution and perhaps improve the performance of local cross validation. For ensuring differentially private public coefficient release, existing techniques such as perturbing the coefficients, objective (Chaudhuri and Monteleoni, 2009; Chaudhuri et al., 2011), or dual variables (Zhang and Zhu, 2016) with additive heavy-tailed noise may be used.

Appendix 7.A Proofs

We first state and prove a theorem which bounds the estimation error for a single party, k. The proof of Theorem 7.5 follows straightforwardly from combining this result for all K parties and applying a union bound.

Theorem 7.6 *Assume all f_i in Eq.* (7.1) *to be convex and smooth with Lipschitz continuous gradients. Under Assumption 7.4, the local difference between the optimal solution to Eq.* (7.1) *at party k, $\boldsymbol{\beta}^*_k$, and the solution returned by* PriDE *at party k, $\widehat{\boldsymbol{\beta}}_k$, is bounded with probability $1-\zeta$ by*

$$\|\widehat{\boldsymbol{\beta}}_k - \boldsymbol{\beta}^*_k\| \leq \frac{\rho}{(1-2\rho)}\left(1+\frac{\sigma}{d_{min}}\left(2+\frac{\sigma\tau_K+\sigma\tau_K^2}{d_{min}}\right)\right)\|\boldsymbol{\beta}^*\| \tag{7.6}$$

where $\zeta = 3c_1\exp(-c_2\log r)+2\xi+2\frac{p}{e^r}+e^{-(\tau+\tau_K)/16}$, $\tau_K=(K-1)\tau_{subs}$, $\rho = C\sqrt{\frac{r\log(2r/\xi)}{\tau_K}}$, $\sigma=\max_k\sigma_k$ and $d_{min}=d_r(\mathbf{X})$, the smallest non-zero singular value of $\mathbf{X}$. C, c_1, c_2, and ξ are absolute positive constants.

Definition 7.7 *For ease of exposition, we shall rewrite the dual problems so that we consider minimizing convex objective functions. More formally, the original problem is then given by*

$$\boldsymbol{\alpha}^* = \underset{\boldsymbol{\alpha}\in\mathbb{R}^n}{\operatorname{argmin}}\left\{D(\boldsymbol{\alpha}) := \sum_{i=1}^n f_i^*(\alpha_i) + \frac{1}{2n\lambda}\boldsymbol{\alpha}^\top\mathbf{K}\boldsymbol{\alpha}\right\}. \tag{7.7}$$

The problem party k solves is described by

$$\tilde{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}\in\mathbb{R}^n}{\operatorname{argmin}}\left\{\tilde{D}_k(\boldsymbol{\alpha}) := \sum_{i=1}^n f_i^*(\alpha_i) + \frac{1}{2n\lambda}\boldsymbol{\alpha}^\top\tilde{\mathbf{K}}_k\boldsymbol{\alpha}\right\}. \tag{7.8}$$

Recall that $\tilde{\mathbf{K}}_k = \bar{\mathbf{X}}_k\bar{\mathbf{X}}_k^\top$, where $\bar{\mathbf{X}}_k$ is the concatenation of the τ raw *features and $(K-1)\tau_{subs}$ perturbed* random *features for party k as in* **Step 4** *of Algorithm 10.*

Proof of Theorem 7.6. For ease of notation, we shall omit the subscript k in $\tilde{\mathbf{K}}_k$ in the following. Defining

$$\Theta = \begin{bmatrix} \mathbf{I}_\tau & 0 & \dots & 0 \\ 0 & \boldsymbol{\Pi}_1 & 0 & \vdots \\ \vdots & \dots & \ddots & 0 \\ 0 & \dots & \dots & \boldsymbol{\Pi}_{K-1} \end{bmatrix} \in \mathbb{R}^{p\times(\tau+(K-1)\tau_{subs})} \tag{7.9}$$

and

$$\mathbf{E} = \begin{bmatrix} \mathbf{0}_\tau & \mathbf{W}_1 & \dots & \mathbf{W}_{K-1} \end{bmatrix} \in \mathbb{R}^{n \times (\tau + (K-1)\tau_{subs})}, \tag{7.10}$$

we can write the original as well as the projected and perturbed kernel matrices explicitly as

$$\mathbf{K} = \mathbf{X}\mathbf{X}^\top \qquad \text{and} \qquad \tilde{\mathbf{K}} = (\mathbf{X}\Theta + \mathbf{E})(\mathbf{X}\Theta + \mathbf{E})^\top$$

respectively. Applying Lemma 7.8, on the l.h.s. of (7.18) we have

$$\begin{aligned}&(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*)^\top (\mathbf{K} - \tilde{\mathbf{K}}) \boldsymbol{\alpha}^* = \\ &\quad (\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*)^\top \left(\mathbf{X}\mathbf{X}^\top - (\mathbf{X}\Theta)(\mathbf{X}\Theta)^\top - (\mathbf{X}\Theta\mathbf{E}^\top) - (\mathbf{X}\Theta\mathbf{E}^\top)^\top - \mathbf{E}\mathbf{E}^\top \right) \boldsymbol{\alpha}^*.\end{aligned}$$

Denoting $\mathbf{U}\mathbf{D}\mathbf{V}^\top = \mathbf{X}$, $\tilde{\gamma} = \mathbf{D}\mathbf{U}^\top \tilde{\boldsymbol{\alpha}}$ and $\gamma^* = \mathbf{D}\mathbf{U}^\top \boldsymbol{\alpha}^*$ we have

$$\begin{aligned}(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*)^\top (\mathbf{K} - \tilde{\mathbf{K}}) \boldsymbol{\alpha}^* = {} & (\tilde{\gamma} - \gamma^*)^\top \left(\mathbf{V}^\top\mathbf{V} - \mathbf{V}^\top\Theta\Theta^\top\mathbf{V} - \mathbf{V}^\top\Theta\mathbf{E}^\top\mathbf{U}\mathbf{D}^{-1} \right. \\ & \left. - \mathbf{D}^{-1}\mathbf{U}^\top\mathbf{E}\Theta^\top\mathbf{V} - \mathbf{D}^{-1}\mathbf{U}^\top\mathbf{E}\mathbf{E}^\top\mathbf{U}\mathbf{D}^{-1} \right) \gamma^*.\end{aligned}$$

By Assumption 7.4 $r = n$, so we have $\mathbf{U}\mathbf{U}^\top = \mathbf{I}_n$. Adding and subtracting $(\tilde{\gamma} - \gamma^*)^\top \mathbf{D}^{-1}\mathbf{U}^\top \left(\sigma^2 (K-1)\tau_{subs} \mathbf{I} \right) \mathbf{U}\mathbf{D}^{-1}\gamma^*$ where $\sigma^2 = \max_k(\sigma_k^2)$ yields

$$\begin{aligned}&(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*)^\top (\mathbf{K} - \tilde{\mathbf{K}}) \boldsymbol{\alpha}^* = \\ &\qquad (\tilde{\gamma} - \gamma^*)^\top \underbrace{\left(\mathbf{I}_r - \mathbf{V}^\top\Theta\Theta^\top\mathbf{V} \right)}_{\text{(I)}} \gamma^* \\ &\qquad - (\tilde{\gamma} - \gamma^*)^\top \underbrace{\left(\mathbf{V}^\top\Theta\mathbf{E}^\top\mathbf{U}\mathbf{D}^{-1} + \mathbf{D}^{-1}\mathbf{U}^\top\mathbf{E}\Theta^\top\mathbf{V} \right)}_{\text{(II)}} \gamma^* \\ &\qquad + (\tilde{\gamma} - \gamma^*)^\top \underbrace{\mathbf{D}^{-1}\mathbf{U}^\top \left(\sigma^2 (K-1)\tau_{subs}\mathbf{I} - \mathbf{E}\mathbf{E}^\top \right) \mathbf{U}\mathbf{D}^{-1}}_{\text{(III)}} \gamma^* \\ &\qquad - (\tilde{\gamma} - \gamma^*)^\top \underbrace{\mathbf{D}^{-1}\mathbf{U}^\top \left(\sigma^2 (K-1)\tau_{subs}\mathbf{I} \right) \mathbf{U}\mathbf{D}^{-1}}_{\text{(IV)}} \gamma^*.\end{aligned}$$

We now focus on bounding each of the terms **(I)**, **(II)**, **(III)** and **(IV)** in turn.

(I). This term is bounded with probability $1 - \left(\xi + \frac{p-\tau}{e^r} \right)$ by $\rho_1 = \sqrt{\frac{cr \log(2r/\xi)}{(K-1)\tau_{subs}}}$ which follows directly from applying Lemma 7.9.

(II). We aim to bound the term $\mathbf{D}^{-1}\mathbf{U}^\top\mathbf{E}\Theta^\top\mathbf{V}\gamma^*$. Since the random terms are sub-Gaussians, we will bound this term using Lemma 7.10 with $Y = (\mathbf{U}^\top\mathbf{E})^\top \in \mathbb{R}^{\tau+(K-1)\tau_{subs}\times r}$, $X = (\Theta^\top\mathbf{V}) \in \mathbb{R}^{\tau+(K-1)\tau_{subs}\times r}$ and $\mathbb{E}\left[\mathbf{U}^\top\mathbf{E}\Theta^\top\mathbf{V}\right] = \mathbf{0}$. Since the first τ rows of Y are $Y_0 = 0$, we decompose Y and the corresponding rows in X as

$$Y = \begin{bmatrix} Y_0 = \mathbf{0} \\ Y_1 \end{bmatrix} \qquad X = \begin{bmatrix} X_0 \\ X_1 \end{bmatrix}.$$

According to this decomposition we can write $Y^\top X = Y_0^\top X_0 + Y_1^\top X_1$. Clearly $Y_0^\top X_0 = 0$ so $Y^\top X$ only has $(K-1)\tau_{subs}$ non-zero summands. Now, applying Lemma 7.10, with probability $1 - c_1\exp(-c_2\log r)$

$$\begin{aligned}\|\mathbf{D}^{-1}\mathbf{U}^\top\mathbf{E}\Theta^\top\mathbf{V}\gamma^*\| &\le \frac{1}{d_{\min}}\|\mathbf{U}^\top\mathbf{E}\Theta^\top\mathbf{V}\gamma^*\| \\ &\le \frac{\sqrt{r}}{d_{\min}}\left|\left|\mathbf{U}^\top\mathbf{E}\Theta^\top\mathbf{V}\gamma^*\right|\right|_\infty \\ &\le \frac{\sigma c_0}{d_{\min}}\|\gamma^*\|\sqrt{\frac{r\log r}{(K-1)\tau_{subs}}}.\end{aligned}$$

(III). Since each entry of $\mathbf{E}$ is an independent Gaussian with variance bounded by σ^2, $\mathbb{E}\left[\mathbf{E}\mathbf{E}^\top\right] = \sigma^2(K-1)\tau_{subs}\mathbf{I}$. By Lemma 7.10 we have with probability $1 - c_1\exp(-c_2\log r)$

$$\|\mathbf{D}^{-1}\mathbf{U}^\top\left(\mathbf{E}\mathbf{E}^\top - \sigma^2(K-1)\tau_{subs}\mathbf{I}\right)\mathbf{U}\mathbf{D}^{-1}\gamma^*\| \le \frac{\sigma^2 c_0}{d_{\min}^2}\|\gamma^*\|\sqrt{r\log r(K-1)\tau_{subs}}$$

(IV).

$$\|\mathbf{D}^{-1}\mathbf{U}^\top\left(\sigma^2(K-1)\tau_{subs}\mathbf{I}\right)\mathbf{U}\mathbf{D}^{-1}\gamma^*\| \le \frac{\sigma^2(K-1)\tau_{subs}}{d_{\min}^2}\|\gamma^*\| \qquad (7.11)$$

Combining (I) – (IV) and using

$$c_0\sqrt{\frac{r\log r}{(K-1)\tau_{subs}}} \le c'\rho_1 = \rho$$

where $\rho = C\sqrt{\frac{r\log(2r/\xi)}{\tau_K}}$ and $\tau_K = (K-1)\tau_{subs}$, we have with probability $1 - \left(3c_1 \exp(-c_2 \log r) + \xi + \frac{p-\tau}{e^r}\right)$

$$(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*)^\top (\mathbf{K}-\tilde{\mathbf{K}})\boldsymbol{\alpha}^* \leq \\ \|\tilde{\gamma} - \gamma^*\|\|\gamma^*\|\rho \left(1 + 2\frac{\sigma}{d_{\min}} + \frac{\sigma^2}{d_{\min}^2}\left(\tau_K + \tau_K^2\right)\right) \quad (7.12)$$

On the r.h.s. of (7.18) we have with $a = \tilde{\gamma} - \gamma^*$

$$(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*)^\top \tilde{\mathbf{K}}(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*) = a^\top (\mathbf{V}^\top \Theta + \mathbf{D}^{-1}\mathbf{U}^\top \mathbf{E})(\mathbf{V}^\top \Theta + \mathbf{D}^{-1}\mathbf{U}^\top \mathbf{E})^\top a.$$

Denoting $\mathbf{w} = \Theta^\top \mathbf{V} a$ and $\mathbf{m} = \mathbf{E}^\top \mathbf{U}\mathbf{D}^{-1} a$ we have

$$(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*)^\top \tilde{\mathbf{K}}(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*) = (\mathbf{w}+\mathbf{m})^\top (\mathbf{w}+\mathbf{m}).$$

For convenience say $\tilde{\tau} = \tau + (K-1)\tau_{subs} = \tau + \tau_K$. Importantly, $\mathbf{m}$ is symmetric around 0, so

$$(\mathbf{w}+\mathbf{m})^\top (\mathbf{w}+\mathbf{m}) = \sum_{i=1}^{\tilde{\tau}} (w_i + m_i)^2 \geq \sum_{i=1}^{\tilde{\tau}} w_i^2 \ \cdot \mathbb{I}_{\{m_i > 0\}}. \quad (7.13)$$

The r.h.s. of this expression corresponds to randomly subsampling summands from $\mathbf{w}^\top \mathbf{w} = a^\top \mathbf{V}^\top \Theta\Theta^\top \mathbf{V} a$ where the subsampling scheme is defined by the non-zero summands stemming from the indicator function in Eq. (7.13). When only considering the non-zero summands, we can write the resulting matrix product as

$$(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*)^\top \tilde{\mathbf{K}}(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*) = (\mathbf{w}+\mathbf{m})^\top (\mathbf{w}+\mathbf{m}) \geq \\ \sum_{i=1}^{\tilde{\tau}} w_i^2 \ \cdot \mathbb{I}_{\{m_i > 0\}} = a^\top \mathbf{V}^\top \tilde{\Theta}\tilde{\Theta}^\top \mathbf{V} a$$

where $\tilde{\Theta}$ contains the columns of Θ corresponding to the non-zero summands. In other words, $\tilde{\Theta}$ corresponds to a random projection matrix that projects to a lower-dimensional space than Θ. Next, we can write

$$(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*)^\top \tilde{\mathbf{K}}(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*) \geq a^\top \mathbf{V}^\top \tilde{\Theta}\tilde{\Theta}^\top \mathbf{V} a = \|a\|^2 - \underbrace{a^\top \left(\mathbf{I} - \mathbf{V}^\top \tilde{\Theta}\tilde{\Theta}^\top \mathbf{V}\right) a}_{\text{(V)}}. \quad (7.14)$$

To lower bound the r.h.s. of this expression, we need to upper bound (V). We achieve this by first finding a lower bound on the number of non-zero summands in Eq. (7.13), i.e. on the number of columns of $\tilde{\Theta}$. Intuitively, the smaller the projection dimension realized by $\tilde{\Theta}$, the larger term (V) becomes. Using the Chernoff bound from Lemma 7.11 with $\delta = 1/2$, we can bound the probability that the number of non-zero summands lies below $\tilde{\tau}/4$ by $\exp(-\tilde{\tau}/16)$. We can then upper bound (V) using Lemma 7.9 for $\tilde{\Theta} \in \mathbb{R}^{p\times\tilde{\tau}/4}$. So with $\tilde{\tau} = \tau + \tau_K$ and $\tilde{\rho} = \sqrt{\frac{cr\log(2r/\xi)}{\tau/4+(K-1)\tau_{subs}/4}}$, we have with probability $1 - (\xi + \frac{p}{e^r} + e^{-(\tau+\tau_K)/16})$

$$
\begin{aligned}
(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*)^\top \tilde{\mathbf{K}}(\tilde{\boldsymbol{\alpha}} - \boldsymbol{\alpha}^*) &\geq \|a\|^2 - a^\top \left(\mathbf{I} - \mathbf{V}^\top\tilde{\Theta}\tilde{\Theta}^\top\mathbf{V}\right) a \\
&\geq \|a\|^2 - \tilde{\rho}\|a\|^2 \\
&\geq (1-\tilde{\rho})\|a\|^2 \\
&\geq (1-2\rho)\|a\|^2. \qquad (7.15)
\end{aligned}
$$

Plugging 7.12 and 7.15 into Lemma 7.8 we have with probability at least $1 - \left(3c_1\exp(-c_2\log r) + 2\xi + 2\frac{p}{e^r} + e^{-(\tau+\tau_K)/16}\right)$

$$
(1-2\rho)\|\tilde{\gamma} - \gamma^*\|^2 \leq \|\tilde{\gamma} - \gamma^*\|\|\gamma^*\|\rho\left(1 + \frac{\sigma}{d_{\min}}\left(2 + \frac{\sigma\tau_K + \sigma\tau_K^2}{d_{\min}}\right)\right). \qquad (7.16)
$$

Finally, with the relationship $\boldsymbol{\beta}^* = -\frac{1}{n\lambda}\mathbf{V}\gamma^*$ and $\tilde{\boldsymbol{\beta}} = -\frac{1}{n\lambda}\mathbf{V}\tilde{\gamma}$ we have $\frac{1}{n\lambda}\|\gamma^*\| = \|\boldsymbol{\beta}^*\|$ and $\|\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\| = \frac{1}{n\lambda}\|\tilde{\gamma} - \gamma^*\|$ due to the orthonormality of $\mathbf{V}$. Thus, we obtain the following error bound for the coefficients estimated by party k

$$
\|\widehat{\boldsymbol{\beta}}_k - \boldsymbol{\beta}_k^*\| \leq \|\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\| \leq \frac{\rho}{(1-2\rho)}\left(1 + \frac{\sigma}{d_{\min}}\left(2 + \frac{\sigma\tau_K + \sigma\tau_K^2}{d_{\min}}\right)\right)\|\boldsymbol{\beta}^*\| \qquad (7.17)
$$

which holds with probability at least
$1 - \left(3c_1\exp(-c_2\log r) + 2\xi + 2\frac{p}{e^r} + e^{-(\tau+\tau_K)/16}\right)$.

$\square$

Appendix 7.B Supporting results

Lemma 7.8 (Adapted from Lemma 1 from Zhang et al. (2014).) *Let* $\boldsymbol{\alpha}^*$ *and* $\tilde{\boldsymbol{\alpha}}$ *be as defined in Definition 7.7. We obtain*

$$\frac{1}{\lambda}(\tilde{\boldsymbol{\alpha}}-\boldsymbol{\alpha}^*)^\top\left(\mathbf{K}-\tilde{\mathbf{K}}_k\right)\boldsymbol{\alpha}^* \geq \frac{1}{\lambda}(\tilde{\boldsymbol{\alpha}}-\boldsymbol{\alpha}^*)^\top\tilde{\mathbf{K}}_k(\tilde{\boldsymbol{\alpha}}-\boldsymbol{\alpha}^*). \tag{7.18}$$

Proof. See Zhang et al. (2014). □

Lemma 7.9 (Concatenating random features (Lemma 3 from Heinze et al. (2014))) *Consider the singular value decomposition* $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ *where* $\mathbf{U} \in \mathbb{R}^{n\times r}$ *and* $\mathbf{V} \in \mathbb{R}^{p\times r}$ *have orthonormal columns and* $\mathbf{D} \in \mathbb{R}^{r\times r}$ *is diagonal;* $r = \mathrm{rank}(\mathbf{X})$*. In addition to the raw features, let* $\bar{\mathbf{X}}_k \in \mathbb{R}^{n\times(\tau+(K-1)\tau_{subs})}$ *contain random features which result from concatenating the* $K-1$ *random projections from the other parties. Furthermore, assume without loss of generality that the problem is permuted so that the raw features of party k's problem are the first* τ *columns of* $\mathbf{X}$ *and* $\bar{\mathbf{X}}_k$*. Finally, let*

$$\Theta_C = \begin{bmatrix} \mathbf{I}_\tau & 0 & \dots & 0 \\ 0 & \mathbf{\Pi}_1 & 0 & \vdots \\ \vdots & \dots & \ddots & 0 \\ 0 & \dots & \dots & \mathbf{\Pi}_{K-1} \end{bmatrix} \in \mathbb{R}^{p\times(\tau+(K-1)\tau_{subs})}$$

such that $\bar{\mathbf{X}}_k = \mathbf{X}\Theta_C$*.*

With probability at least $1-\left(\xi+\frac{p-\tau}{e^r}\right)$

$$\|\mathbf{V}^\top\Theta_C\Theta_C^\top\mathbf{V}-\mathbf{V}^\top\mathbf{V}\| \leq \sqrt{\frac{c\log(2r/\xi)r}{(K-1)\tau_{subs}}}.$$

Lemma 7.10 (Adapted from Lemma 14 from Loh and Wainwright (2012).) *If* $X \in \mathbb{R}^{n\times p_1}$ *and* $Y \in \mathbb{R}^{n\times p_2}$ *are zero-mean sub-Gaussian matrices with parameters* (Σ_x, σ_x^2) *and* (Σ_y, σ_y^2) *respectively. If* $n \gtrsim \log p$ *then*

$$\mathbb{P}\left(\left|\left|\left(\frac{Y^\top X}{n}-\mathbb{E}\left[Y^\top X\right]\right)\right|\right|_\infty \geq c_0\sigma_x\sigma_y\sqrt{\frac{\log p}{n}}\right) \leq c_1\exp(-c_2\log p).$$

Lemma 7.11 (Chernoff bound for sum of independent Bernoulli trials (Goemans, 2015)) *Let* $X = \sum_{i=1}^{n} X_i$*, where* $X_i = 1$ *with probability* p_i *and* $X_i = 0$ *with probability* $1 - p_i$*, and all* X_i *are independent. Let* $\mu = \mathbb{E}(X) = \sum_{i=1}^{n} p_i$*. Then*
(i) Upper Tail:

$$\mathbb{P}(X \geq (1+\delta)\mu) \leq e^{-\frac{\delta^2}{2+\delta}\mu} \text{ for all } \delta > 0;$$

(ii) Lower Tail:

$$\mathbb{P}(X \leq (1-\delta)\mu) \leq e^{-\mu\delta^2/2} \text{ for all } 0 < \delta < 1.$$

Appendix 7.C Connection between $(\epsilon, \delta, \mathcal{S})$-differential privacy and regularized least-squares estimation

In the PriDE framework, consider the unregularized local objective function for a single party k when the functions f_i are the squared error. From (7.9) and (7.10) we have (omitting the subscript k for ease of notation)

$$\|Y - \bar{\mathbf{X}}\mathbf{b}\|^2 = \|Y - (\mathbf{X}\Theta + \mathbf{E})\mathbf{b}\|^2.$$

Denoting $\tilde{\mathbf{X}}_l$, $\forall\ l \neq k$ as the concatenated random features we have

$$\|Y - (\mathbf{X}\Theta + \mathbf{E})\mathbf{b}\|^2 = \|Y - (\mathbf{X}_k\mathbf{b}_k + \sum_{l \neq k}(\tilde{\mathbf{X}}_l + \mathbf{W}_l)\mathbf{b}_l)\|^2.$$

Since all of the elements in $\mathbf{W}$ are sampled i.i.d. from an independent Gaussian with variance σ^2, let us now consider taking the expectation of this expression with respect to the randomness in $\mathbf{W}$:

$$\mathbb{E}_{\mathbf{W}}\|Y - (\mathbf{X}_k\mathbf{b}_k + \sum_{l \neq k}(\tilde{\mathbf{X}}_l + \mathbf{W}_l)\mathbf{b}_l)\|^2.$$

Due to independence, we can simply consider the univariate expectation

$$\begin{aligned}\mathbb{E}_{w\sim\mathcal{N}(0,\sigma^2)}\ (y-&(\tilde{x}_l + w)\mathbf{b}_l)^2 \\ &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{w^2}{2\pi\sigma^2}\right\} \cdot \left(y - (\tilde{x}_l + w)\mathbf{b}_l\right)^2 dw \\ &= (y - \tilde{x}_l\mathbf{b}_l)^2 + \sigma^2\mathbf{b}_l^2.\end{aligned}$$

So in $\tau + (K-1)\tau_{subs}$ dimensions we obtain a *regularized* least squares objective where the regularization is only on the $(K-1)\tau_{subs}$ *random* features

$$\|Y - (\mathbf{X}_k\mathbf{b}_k + \sum_{l\neq k}\tilde{\mathbf{X}}_l\mathbf{b}_l)\|^2 + \sigma^2\sum_{l\neq k}\mathbf{b}_l^2. \tag{7.19}$$

The strength of the regularization is governed by the variance of the perturbation.

Appendix 7.D Additional experimental details and results

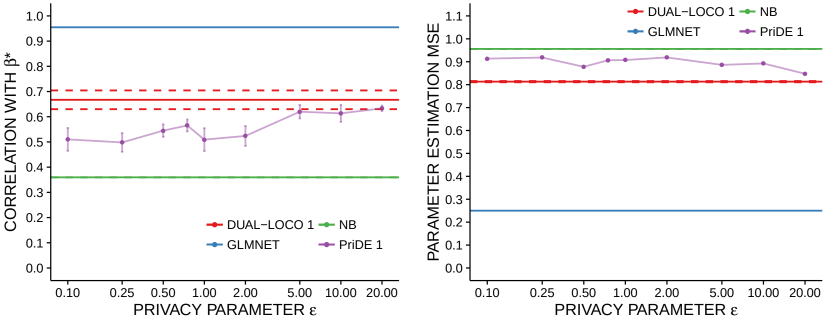
7.D.1 Breast cancer gene expression data

Finally, we show an application to a problem in clinical bioinformatics. This experiment aims to assess the performance of PriDE on a real data set from a domain where sensitive data is ubiquitous. We use the breast cancer data set GSE3494[3] (Miller et al., 2005). Our task is to predict the disease specific survival time of each patient in years where the objective can be reformulated via an Accelerated Failure Time model as a least squares objective; here with constant weights. The median follow-up of patients was 122 months. Approximately $45,000$ gene expressions are available from $n = 236$ patients. We selected genes with the largest absolute marginal correlation with the response, resulting in $p = 2,000$ distributed across $K = 4$ parties. The data set is split into training ($n_{\text{train}} = 188$) and test set ($n_{\text{test}} = 48$). In this application, accurate estimates of $\boldsymbol{\beta}$ are of primary interest to assess which genes are good predictors for survival time.

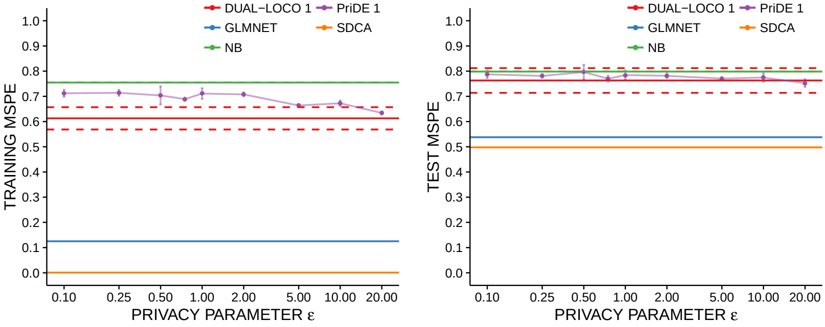
---

[3] `http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE3494`.

The first and second column in Figures 7.6–7.9 show comparisons of correlation and estimation error for $\tau_{subs} = \{0.05, 0.2\} \cdot \tau$ with respect to the SDCA solution $\boldsymbol{\beta}^*$. The third and the fourth column show the normalized training and test prediction MSE. As above, we again observe a large difference between the NB and Dual-Loco solutions which is essential for there to be some expected gain from using PriDE. We observe similar trends as in the previous experiments: as ϵ increases, PriDE improves upon NB and approaches the Dual-Loco solution. The trade-off between ϵ and τ_{subs} (implied by Theorem 7.5 and discussed above) is again apparent. The convergence to the Dual-Loco solution with increasing values of ϵ is somewhat slower than in the previous experiment. This is partly due to the gene expression data having heavier tails, resulting in a larger $\max_k (\theta_k)$. This requires a larger noise level to guarantee privacy leading to a more heavily regularized learning problem (cf. Tables 7.1 and 7.3).

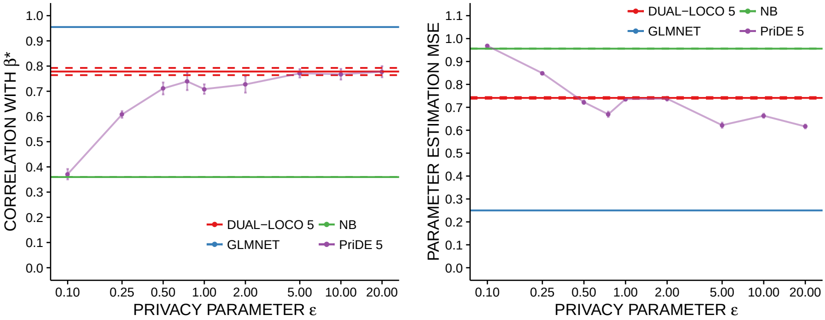


(a) $\mathrm{corr}(\widehat{\boldsymbol{\beta}}, \boldsymbol{\beta}^*)$ (b) $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|^2/\|\boldsymbol{\beta}^*\|^2$



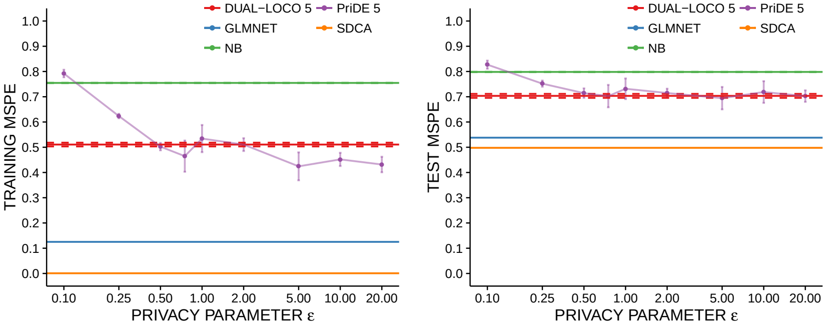
(c) Training MSPE (d) Test MSPE

Figure 7.6.: Gene expression data: Results for projection dimension $\tau_{subs} = 0.01\tau$. In comparison to larger projection dimensions (cf. Figures 7.7–Figures 7.9), a projection dimension of $\tau_{subs} = 0.01\tau$ is not sufficient to capture the signal of the non-local features accurately. This is apparent from the gap in performance between Dual-Loco and the Glmnet/SDCA estimates which are obtained without any constraints on privacy or communication. Secondly, when $\tau_{subs} = 0.01\tau$, varying ϵ only has a very small effect on the performance of PriDE: due to the small projection dimension the additional regularization introduced by the additive noise can be attenuated by choosing smaller values for λ also when ϵ is small. As τ_{subs} increases, the performance of Dual-Loco and PriDE improve as term (i) in Eq. (7.5) decreases. As predicted by Theorem 7.5, we also observe the adverse effect on the approximation accuracy induced by term (ii) in Eq. (7.5) for small values of ϵ and large values of τ_{subs}.



(a) $\mathrm{corr}(\widehat{\boldsymbol{\beta}}, \boldsymbol{\beta}^*)$

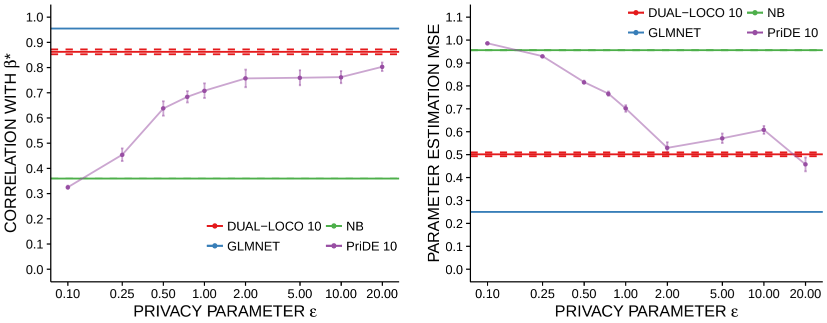
(b) $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|^2/\|\boldsymbol{\beta}^*\|^2$



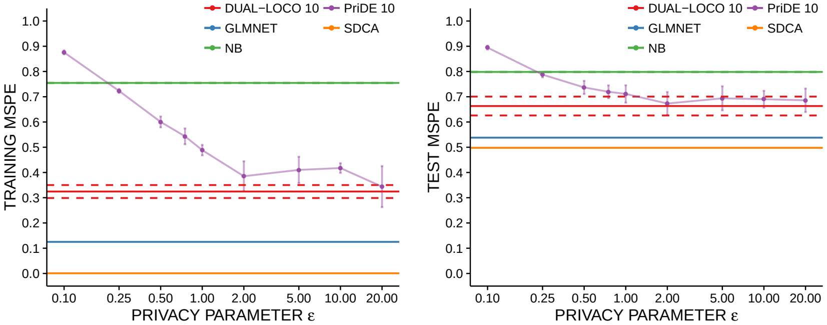
(c) Training MSPE

(d) Test MSPE

Figure 7.7.: Gene expression data: Results for projection dimension $\tau_{subs} = 0.05\tau$.

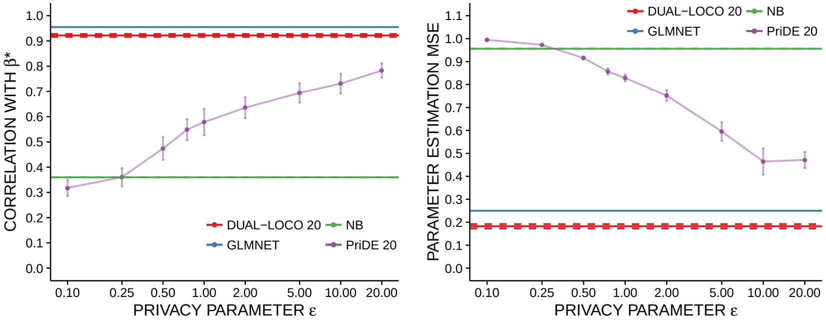


(a) $\mathrm{corr}(\widehat{\boldsymbol{\beta}}, \boldsymbol{\beta}^*)$ (b) $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|^2/\|\boldsymbol{\beta}^*\|^2$

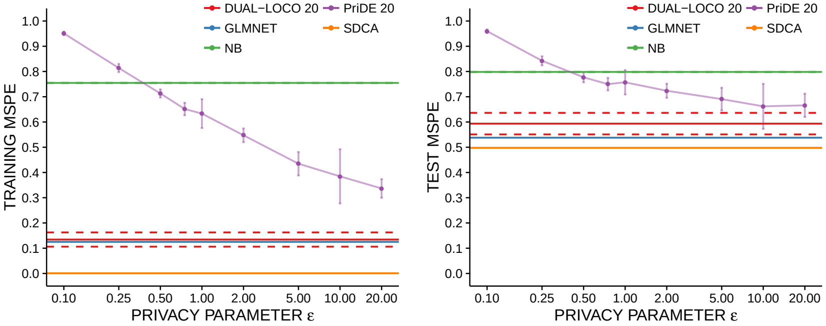


(c) Training MSPE (d) Test MSPE

Figure 7.8.: Gene expression data: Results for projection dimension $\tau_{subs} = 0.1\tau$.



(a) $\mathrm{corr}(\widehat{\boldsymbol{\beta}}, \boldsymbol{\beta}^*)$ (b) $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|^2 / \|\boldsymbol{\beta}^*\|^2$



(c) Training MSPE (d) Test MSPE

Figure 7.9.: Gene expression data: Results for projection dimension $\tau_{subs} = 0.2\tau$.

7.D.2 Data tables

Table 7.2.: Projection dimensions

	K	$\tau = p/K$	0.01τ	0.05τ	0.1τ	0.2τ
SIMULATED	2	200	2	10	20	40
CLIMATE	4	$2,592$	26	130	259	518
CANCER	4	500	5	25	50	100

Table 7.3.: Largest noise standard deviation $\max_k(\sigma_k)$ when $\delta = 0.05$

ϵ	0.1	0.25	0.5	0.75	1	2	5	10	20
SIMULATED	162.47	66.99	35.10	24.42	19.05	10.87	5.67	3.68	2.48
CLIMATE	186.59	76.93	40.30	28.04	21.88	12.48	6.51	4.22	2.84
CANCER	239.41	98.71	51.71	35.98	28.07	16.02	8.35	5.42	3.65

7.D.3 Simulation setting

We consider $K = 2$ blocks of features. One block of features could, for instance, contain genomic data, such as measurements of single nucleotide polymorphisms (SNPs). We shall denote the set of features contained in this block by C. Due to its highly personal and sensitive nature, genomic data arising from techniques like SNP genotyping is hardly ever publicly available. The other block of features could hold gene expression data. We denote this second set of features by X. Some of the genomic features have an effect on some of the gene expression features and both sets of features contribute to the response Y. This results in the structure shown in Figure 7.10. Due to the dependence between C and X one cannot obtain accurate coefficient estimates for the effect of X on Y when only including X into the model. In such settings, PRIDE allows to adjust for the

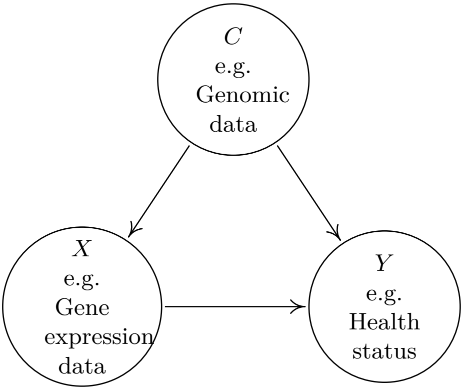
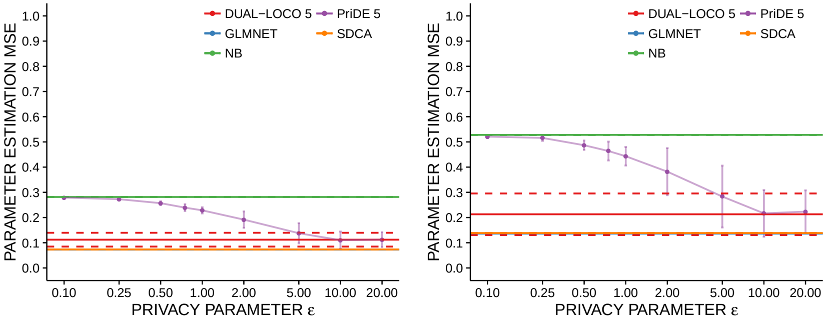


Figure 7.10.: The confounders in C influence both the variables in X as well as the response Y.

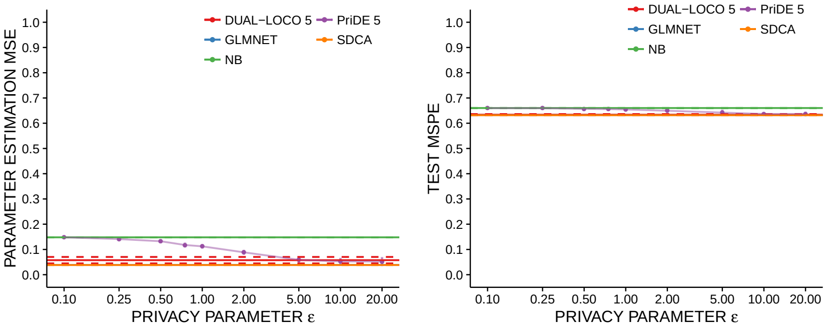
confounding effects from C on X while guaranteeing $(\epsilon, \delta, \mathcal{S})$-differential privacy.

Specifically, each blocks of features contains $\tau = 200$ features. So $p = 400$ and we choose $n = 1000$ ($n_{train} = 800$ resp. $n_{test} = 200$). In order to create an interesting correlation structure both within the blocks of features and between C and X, we consider a Gaussian random field on a 20×20 grid. So each grid point corresponds to one feature and we generate n realizations from the model. We add confounding effects from C on X by selecting 20 pairs of features from X and C at random. Denote the set of tuples by $\mathcal{M}$ and a single tuple by $m = (i_x, j_c)$ where i_x is the index of the chosen feature from X and j_c is the index of the chosen feature from C. For all tuples in $\mathcal{M}$ we set $X^{i_x} \leftarrow X^{i_x} + C^{j_c}$. Subsequently, we create the signal by aligning the coefficients $\boldsymbol{\beta}$ with the top 20 principal components of the full design matrix. Finally, the response is generated as $Y = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\eta}$. The elements of $\boldsymbol{\eta}$ are i.i.d. zero-mean Gaussian noise with a standard deviation set to achieve a signal-to-noise ratio $\text{SNR} = ||\mathbf{X}\boldsymbol{\beta}||_2^2 / ||\boldsymbol{\eta}||_2^2$ of approximately 0.75. In this simulation, a noise standard deviation of 500 yielded the desired SNR.

For all further details, we refer to the data generating code which is available at `https://onlinelibrary.wiley.com/doi/abs/10.1002/sta4.189` in the script `generate.R`.



(a) $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|^2/\|\boldsymbol{\beta}\|^2$ (b) $\|\widehat{\boldsymbol{\beta}}_X - \boldsymbol{\beta}_X\|^2/\|\boldsymbol{\beta}_X\|^2$

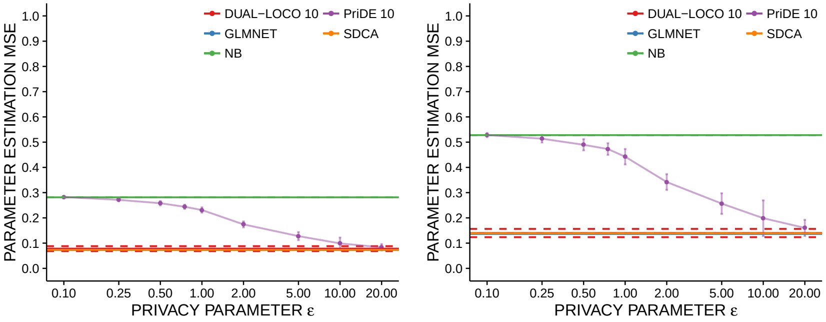


(c) $\|\widehat{\boldsymbol{\beta}}_C - \boldsymbol{\beta}_C\|^2/\|\boldsymbol{\beta}_C\|^2$ (d) Test MSPE

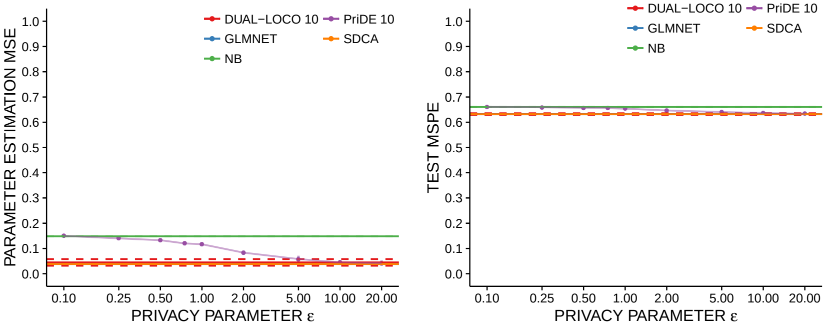
Figure 7.11.: Simulated data: Results for projection dimension $\tau_{subs} = 0.05\tau$. Normalized parameter estimation MSE w.r.t. true $\boldsymbol{\beta}$: ((a)) overall, ((b)) for block X and ((c)) for block C. ((d)) Normalized prediction MSE on test set.

7.D.4 Additional results for simulated data

In contrast to the other two experiments, Figures 7.11–7.12 show that the performance of PriDE is not as sensitive to the chosen projection dimension in case of the synthetic data set. The approximation quality is fairly similar for $\tau_{subs} = \{0.05, 0.1, 0.2\} \cdot \tau$ even though the standard errors are larger for $\tau_{subs} = 0.05\tau$. This can be explained by the small value of $d_{\min}$—here, $d_{\min}$ seems to be the quantity mostly determining the term (ii) in Eq. (7.5), so that manipulating τ_{subs} only has a very small effect on the overall bias.

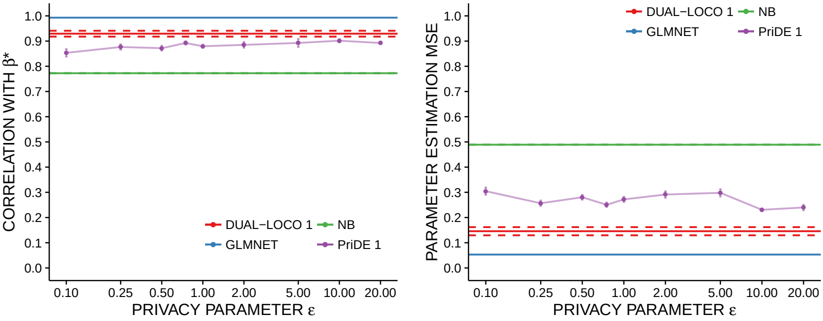


(a) $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|^2/\|\boldsymbol{\beta}\|^2$ (b) $\|\widehat{\boldsymbol{\beta}}_X - \boldsymbol{\beta}_X\|^2/\|\boldsymbol{\beta}_X\|^2$

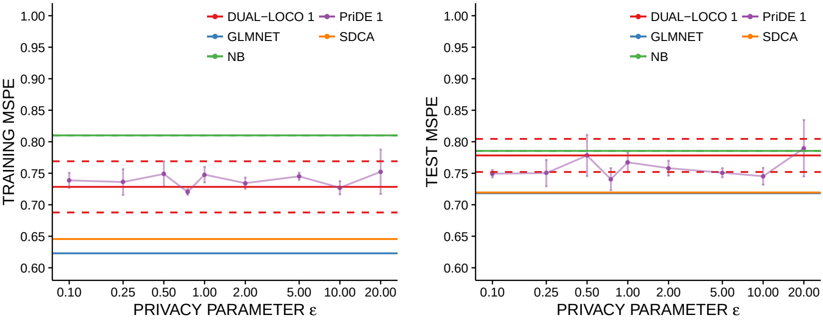


(c) $\|\widehat{\boldsymbol{\beta}}_C - \boldsymbol{\beta}_C\|^2/\|\boldsymbol{\beta}_C\|^2$ (d) Test MSPE

Figure 7.12.: Simulated data: Results for projection dimension $\tau_{subs} = 0.1\tau$. Normalized parameter estimation MSE w.r.t. true $\boldsymbol{\beta}$: ((a)) overall, ((b)) for block X and ((c)) for block C. ((d)) Normalized prediction MSE on test set.



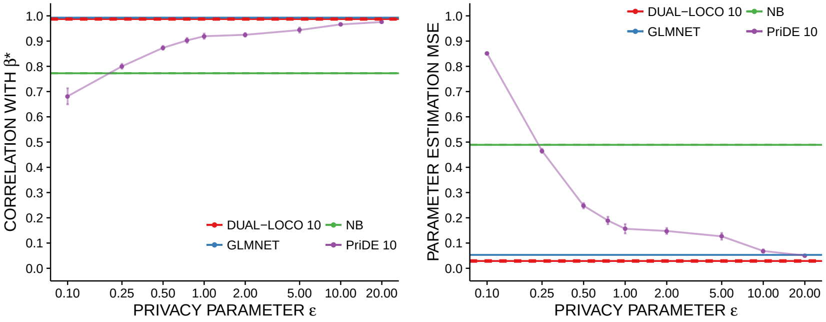
(a) $\mathrm{corr}(\widehat{\boldsymbol{\beta}}, \boldsymbol{\beta}^*)$ (b) $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|^2/\|\boldsymbol{\beta}^*\|^2$



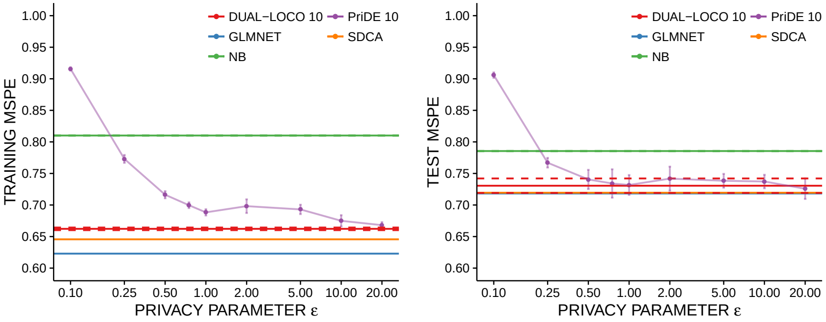
(c) Training MSPE (d) Test MSPE

Figure 7.13.: Climate model data: Results for projection dimension $\tau_{subs} = 0.01\tau$. In comparison to larger projection dimensions (cf. Figures 7.14–Figures 7.15), a projection dimension of $\tau_{subs} = 0.01\tau$ is not sufficient to capture the signal of the non-local features accurately. This is apparent from the gap in performance between Dual-Loco and the Glmnet/SDCA estimates which are obtained without any constraints on privacy or communication. Secondly, when $\tau_{subs} = 0.01\tau$, varying ϵ only has a very small effect on the performance of PriDE: due to the small projection dimension the additional regularization introduced by the additive noise can be attenuated by choosing smaller values for λ also when ϵ is small.

7.D.5 Additional results for climate model data

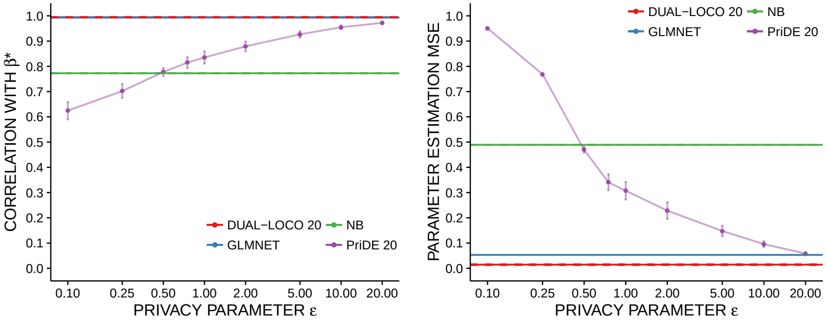


(a) $\operatorname{corr}(\widehat{\boldsymbol{\beta}}, \boldsymbol{\beta}^*)$ (b) $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|^2/\|\boldsymbol{\beta}^*\|^2$

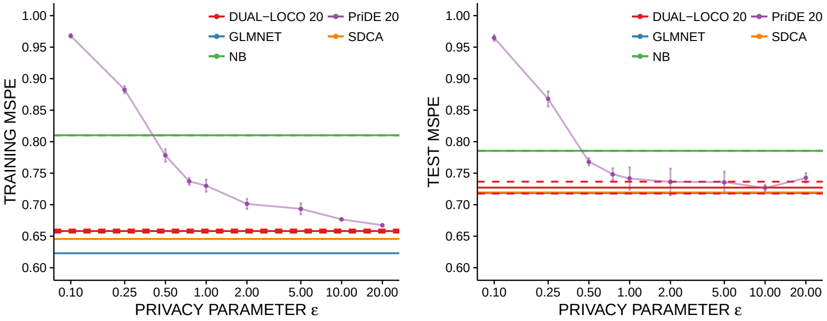


(c) Training MSPE (d) Test MSPE

Figure 7.14.: Climate model data: Results for projection dimension $\tau_{subs} = 0.1\tau$. As τ_{subs} increases, the performance of Dual-Loco and PriDE improve as term (i) in Eq. (7.5) decreases. As predicted by Theorem 7.5, we also observe the adverse effect on the approximation accuracy induced by term (ii) in Eq. (7.5) for small values of ϵ and large values of τ_{subs}.



(a) $\mathrm{corr}(\widehat{\boldsymbol{\beta}}, \boldsymbol{\beta}^*)$ (b) $\|\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\|^2 / \|\boldsymbol{\beta}^*\|^2$



(c) Training MSPE (d) Test MSPE

Figure 7.15.: Climate model data: Results for projection dimension $\tau_{subs} = 0.2\tau$. As τ_{subs} increases, the performance of Dual-Loco and PriDE improve as term (i) in Eq. (7.5) decreases. As predicted by Theorem 7.5, we also observe the adverse effect on the approximation accuracy induced by term (ii) in Eq. (7.5) for small values of ϵ and large values of τ_{subs}.

Appendix 7.E Privacy preserving cross validation

When the regularization parameter λ is given, PRIDE preserves $(\epsilon, \delta, \mathcal{S})$-differential privacy (see Theorem 7.3). Finding a suitable λ via v-fold cross validation (CV) without compromising privacy is challenging. In general, useful privacy preserving model selection procedures are an active area of research and few procedures have been proposed (Chaudhuri and Vinterbo, 2013). In Heinze et al. (2016), λ is tuned "globally", i.e. the local predictions for a particular λ are communicated, added and thus evaluated on the global objective. Alternatively, the local objectives could be targeted—in this case only the perturbed random features are communicated. Communicating predictions would compromise privacy so only local CV is feasible in a setting where privacy is critical. The optimal λ is then chosen by each party individually based on the CV performance on the local design matrix, using both the raw and the perturbed random features.

A few results concerning the selection of λ in local and global CV are given in Table 7.4 which compares the chosen value for λ using global and local cross validation on the climate dataset. For larger values of ϵ, local CV selects similar values for λ as global CV. However, for small values of ϵ ($\epsilon \leq 0.5$) the local cross validation scheme selects values for λ that are much too large. Consequently, the predictive accuracy deteriorates, making the local CV scheme infeasible for small values of ϵ. In §7.5, we tuned the regularization parameter using 5-fold global cross validation for all methods to assess the performance of PRIDE without confounding the comparison with this additional source of uncertainty.

One interesting aspect about the optimal value for λ chosen by global CV is the following trend. The smaller ϵ, the smaller a value for λ tends to be selected. This is consistent with the fact that the additive noise acts as an additional regularizer (see discussion in §7.4.2 and §7.C). As this additional regularization increases with σ^2, the chosen λ decreases, keeping the total regularization constant. However, this balancing effect is only possible as long as the additional regularization is not too large—at some point the chosen λ approaches zero and cannot be decreased further.

Table 7.4.: Cross validation results: Comparison of the chosen value for λ in local cross validation (LCV) and global cross validation (GCV) using the climate simulation data with projection dimension $\tau_{subs} = 0.05\tau$.

ϵ	0.25	0.5	0.75	1	2	5	10	20
GCV: λ	47	125	137	140	159	116	108	104
GCV: TEST MSE	0.7551	0.7396	0.7396	0.7537	0.7460	0.7393	0.7358	0.7414
LCV: λ for $k = 1$	$> 1,000,000$	$85,000$	117	85	99	153	147	84
LCV: λ for $k = 2$	$> 1,000,000$	$90,000$	132	108	110	158	157	85
LCV: λ for $k = 3$	$> 1,000,000$	$105,000$	133	141	197	164	160	104
LCV: λ for $k = 4$	$> 1,000,000$	$110,000$	174	145	237	250	161	111
LCV: TEST MSE	0.9960	0.9956	0.7576	0.7399	0.7351	0.7380	0.7332	0.7329

Bibliography

Abadi, M. et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems.* Software available from tensorflow.org. 2015. URL: `https://www.tensorflow.org/`.

Ailon, N. and B. Chazelle. “The fast Johnson-Lindenstrauss transform and approximate nearest neighbors”. In: *SIAM Journal on Computing* 39.1 (2009), pp. 302–322.

Aldrich, J. “Autonomy”. In: *Oxford Economic Papers* 41 (1989), pp. 15–34.

Ali, R. A., T. S. Richardson, and P. Spirtes. “Markov equivalence for ancestral graphs”. In: *Annals of Statistics* 37 (2009), pp. 2808–2837.

Andersson, S. A., D. Madigan, and M. D. Perlman. “A characterization of Markov equivalence classes for acyclic digraphs”. In: *Annals of Statistics* 25 (1997), pp. 505–541.

Angrist, J. D., G. W. Imbens, and D. B. Rubin. “Identification of causal effects using instrumental variables”. In: *Journal of the American Statistical Association* 91 (1996), pp. 444–455.

Bagnell, J. “Robust supervised learning”. In: *Proceedings of the national conference on artificial intelligence.* Vol. 20. 2. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999. 2005, p. 714.

Bahadori, M. T., K. Chalupka, E. Choi, R. Chen, W. F. Stewart, and J. Sun. “Causal Regularization”. In: *arXiv preprint arXiv:1702.02604* (2017).

Barocas, S. and A. D. Selbst. “Big Data’s Disparate Impact”. In: *104 California Law Review 671* (2016).

Bartlett, P., M. Jordan, and J. McAuliffe. *Convexity, classification, and risk bounds.* Tech. rep. Department of Statistics, U.C. Berkeley, 2003.

Bassily, R., A. Smith, and A. Thakurta. “Private Empirical Risk Minimization: Efficient Algorithms and Tight Error Bounds”. In: *Proceedings of the 2014 IEEE 55th Annual Symposium on Foundations of Computer Science*. FOCS ’14. Washington, DC, USA: IEEE Computer Society, 2014, pp. 464–473.

Beimel, A., K. Nissim, and E. Omri. “Distributed Private Data Analysis: Simultaneously Solving How and What”. In: *Advances in Cryptology – CRYPTO 2008*. Ed. by D. Wagner. Berlin, Heidelberg: Springer, 2008, pp. 451–468.

Belkin, M., P. Niyogi, and V. Sindhwani. “Manifold regularization: A geometric framework for learning from labeled and unlabeled examples”. In: *Journal of machine learning research* 7.Nov (2006), pp. 2399–2434.

Ben-David, S., J. Blitzer, K. Crammer, and F. Pereira. “Analysis of Representations for Domain Adaptation”. In: *Advances in Neural Information Processing Systems 19*. 2007.

Ben-Tal, A., D. Den Hertog, A. De Waegenaere, B. Melenberg, and G. Rennen. “Robust solutions of optimization problems affected by uncertain probabilities”. In: *Management Science* 59.2 (2013), pp. 341–357.

Bergsma, W. and A. Dassios. “A consistent test of independence based on a sign covariance related to Kendall’s tau”. In: *Bernoulli* 20 (2014), pp. 1006–1028.

Besserve, M., N. Shajarisales, B. Schölkopf, and D. Janzing. “Group invariance principles for causal generative models”. In: *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics (AISTATS)*. Vol. 84. Proceedings of Machine Learning Research. PMLR, 2018, pp. 557–565.

Bickel, P. J., E. A. Hammel, and J. W. O’connell. “Sex Bias in Graduate Admissions: Data from Berkeley”. In: *Science* 187 (Mar. 1975), pp. 398–404.

Blum, J. R., J. Kiefer, and M. Rosenblatt. “Distribution Free Tests of Independence Based on the Sample Distribution Function”. In: *The Annals of Mathematical Statistics* 32 (1961), pp. 485–498.

Bollen, K. A. *Structural Equations with Latent Variables*. New York, USA: John Wiley & Sons, 1989.

Bolukbasi, T., K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai. "Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings". In: *Advances in Neural Information Processing Systems 29*. 2016.

Bottou, L. *Une approche geometrique de l'apprentissage non supervise.* Talk at the Sorbonne on March 6th, 2018. Retrieved April 30th, 2018. Mar. 2018. URL: `http://video.upmc.fr/differe.php?collec=S_C_colloquium2018&video=1`.

Bouchacourt, D., R. Tomioka, and S. Nowozin. "Multi-Level Variational Autoencoder: Learning Disentangled Representations From Grouped Observations". In: *AAAI Conference on Artificial Intelligence*. 2018.

Boutsidis, C. and A. Gittens. "Improved matrix algorithms via the Subsampled Randomized Hadamard Transform". In: *arXiv preprint arXiv: 1204.0062* (2012).

Breiman, L. "Random Forests". In: *Machine Learning* 45 (2001), pp. 5–32.

Bühlmann, P., J. Peters, and J. Ernest. "CAM: Causal Additive Models, high-dimensional order search and penalized regression". In: *Annals of Statistics* 42 (2014), pp. 2526–2556.

Burkard, R. E. "Quadratic Assignment Problems". In: *Handbook of Combinatorial Optimization*. Ed. by P. M. Pardalos, D.-Z. Du, and R. L. Graham. 2nd. Springer New York, 2013, pp. 2741–2814.

Burkhardt, R. et al. "Integration of Genome-Wide SNP Data and Gene-Expression Profiles Reveals Six Novel Loci and Regulatory Mechanisms for Amino Acids and Acylcarnitines in Whole Blood". In: *PLOS Genetics* 11.9 (Sept. 2015), pp. 1–25.

Chalupka, K., P. Perona, and F. Eberhardt. "Visual Causal Feature Learning". In: *Uncertainty in Artificial Intelligence* (2014).

Chaudhuri, K. and C. Monteleoni. "Privacy-preserving logistic regression". In: *Advances in Neural Information Processing Systems 21*. Ed. by D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou. Curran Associates, Inc., 2009, pp. 289–296.

Chaudhuri, K. and S. A. Vinterbo. "A Stability-based Validation Procedure for Differentially Private Machine Learning". In: *Advances in Neural Information Processing Systems 26*. Ed. by C. J. C. Burges, L.

Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger. Curran Associates, Inc., 2013, pp. 2652–2660.

Chaudhuri, K., C. Monteleoni, and A. D. Sarwate. “Differentially private empirical risk minimization”. In: *The Journal of Machine Learning Research* 12 (2011), pp. 1069–1109.

Chen, X., Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. “InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets”. In: *Advances in Neural Information Processing Systems 29*. 2016.

Chickering, D. M. “Learning equivalence classes of Bayesian-network structures”. In: *Journal of Machine Learning Research* 2 (2002), pp. 445–498.

Chickering, D. M. “Optimal structure identification with greedy search”. In: *Journal of Machine Learning Research* 3 (2002), pp. 507–554.

Cho, S. W., S. Kim, Y. Kim, J. Kweon, H. S. Kim, S. Bae, and J.-S. Kim. “Analysis of off-target effects of CRISPR/Cas-derived RNA-guided endonucleases and nickases”. In: *Genome Research* 24 (2014), pp. 132–141.

Claassen, T., J. M. Mooij, and T. Heskes. “Learning Sparse Causal Models is not NP-hard”. In: *Proceedings of the 29th Annual Conference on Uncertainty in Artificial Intelligence (UAI)*. 2013.

Colombo, D. and M. H. Maathuis. “Order-Independent Constraint-Based Causal Structure Learning”. In: *Journal of Machine Learning Research* 15 (2014), pp. 3741–3782.

Colombo, D., M. H. Maathuis, M. Kalisch, and T. S. Richardson. “Learning high-dimensional directed acyclic graphs with latent and selection variables”. In: *Annals of Statistics* 40 (2012), pp. 294–321.

Comon, P. “Independent component analysis – a new concept?” In: *Signal Processing* 36 (1994), pp. 287–314.

Conover, W. J. *Practical nonparametric statistics*. New York: John Wiley & Sons, 1971.

Cooper, G. and C. Yoo. “Causal discovery from a mixture of experimental and observational data”. In: *Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence (UAI)*. 1999, pp. 116–125.

Crawford, K. “Artificial Intelligence’s White Guy Problem”. In: *The New York Times, June 25 2016* (2016). URL: `https://www.nytimes.com/2016/06/26/opinion/sunday/artificial-intelligences-white-guy-problem.html`.

Csurka, G. “A Comprehensive Survey on Domain Adaptation for Visual Applications”. In: *Domain Adaptation in Computer Vision Applications.* 2017, pp. 1–35.

Dankar, F. K. and K. El Emam. “Practicing Differential Privacy in Health Care: A Review”. In: *Transactions on Data Privacy* 6.1 (2013), pp. 35–67.

Dawid, A. P. “Causal inference without counterfactuals”. In: *Journal of the American Statistical Association* 95 (2000), pp. 407–424.

Denton, E. L. and V. Birodkar. “Unsupervised Learning of Disentangled Representations from Video”. In: *Advances in Neural Information Processing Systems 30.* 2017.

Dhillon, P., Y. Lu, D. P. Foster, and L. Ungar. “New Subsampling Algorithms for Fast Least Squares Regression”. In: *Advances in Neural Information Processing Systems.* 2013.

Didelez, V. “Handbook of Graphical Models”. In: To appear. Chapman & Hall/CRC, 2017. Chap. Causal Concepts and Graphical Models.

Dieterich, W., C. Mendoza, and T. Brennan. “Compas risk scales: Demonstrating accuracy equity and predictive parity”. In: *Technical Report* (2016).

D’Orazio, V., J. Honaker, and G. King. “Differential Privacy for Social Science Inference”. In: *SSRN Electronic Journal* (2015).

Drton, M. and M. H. Maathuis. “Structure Learning in Graphical Modeling”. In: *Annual Review of Statistics and Its Application* 4 (2017), pp. 365–393.

Duchi, J. C., M. I. Jordan, and H. B. McMahan. “Estimation, Optimization, and Parallelism when Data is Sparse”. In: *Advances in Neural Information Processing Systems.* 2013.

Duchi, J. C., M. I. Jordan, and M. J. Wainwright. “Local privacy and statistical minimax rates”. In: *Proceedings of the 2013 IEEE 54th Annual*

Symposium on Foundations of Computer Science. IEEE. Washington, DC, USA: IEEE Computer Society, 2013, pp. 429–438.

Dwork, C. “Differential privacy”. In: *Automata, languages and programming*. Springer, 2006.

Eaton, D. and K. P. Murphy. “Exact Bayesian structure learning from uncertain interventions”. In: *Proceedings of the 11th International Conference on Artificial Intelligence and Statistics (AISTATS)*. 2007, pp. 107–114.

Eberhardt, F. and R. Scheines. “Interventions and Causal Inference”. In: *Philosophy of Science* 74 (2007), pp. 981–995.

Eberhardt, F., P. O. Hoyer, and R. Scheines. “Combining experiments to discover linear cyclic models with latent variables”. In: *International Conference on Artificial Intelligence and Statistics (AISTATS)*. 2010, pp. 185–192.

El Emam, K., E. Jonker, L. Arbuckle, and B. Malin. “A systematic review of re-identification attacks on health data”. In: *PLOS ONE* 6.12 (2011), pp. 1–12.

Emspak, J. “How a Machine Learns Prejudice”. In: *Scientific American, December 29 2016* (2016). URL: `https://www.scientificamerican.com/article/how-a-machine-learns-prejudice/`.

Esteva, A., B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun. “Dermatologist-level classification of skin cancer with deep neural networks”. In: *Nature* 542 (2017).

Friedman, J., T. Hastie, and R. Tibshirani. “Regularization paths for generalized linear models via coordinate descent”. In: *Journal of Statistical Software* 33.1 (2010), pp. 1–22.

Frisch, R. “Autonomy of economic relations: Statistical versus theoretical relations in economic macrodynamics”. Paper given at League of Nations. Reprinted in D.F. Hendry and M.S. Morgan (1995), The Foundations of Econometric Analysis, Cambridge University Press. 1938.

Fukumizu, K., A. Gretton, X. Sun, and B. Schölkopf. “Kernel Measures of Conditional Dependence”. In: *Advances in Neural Information Processing Systems 20 (NIPS)*. MIT Press, 2008, pp. 489–496.

Ganin, Y. et al. “Domain-adversarial Training of Neural Networks”. In: *Journal of Machine Learning Research* 17.1 (2016).

Gao, R., X. Chen, and A. Kleywegt. In: *arXiv preprint arXiv:1712.06050* (2017).

Gastwirth, J. L., Y. R. Gel, W. L. W. Hui, V. Lyubchich, W. Miao, and K. Noguchi. *lawstat: Tools for Biostatistics, Public Policy, and Law.* R package version 3.0. 2015. URL: `https://CRAN.R-project.org/package=lawstat`.

Goeman, J. J. and A. Solari. “Multiple testing for exploratory research”. In: *Statistical Science* (2011), pp. 584–597.

Goemans, M. “Chernoff bounds and some applications”. In: (2015). URL: `http://math.mit.edu/~goemans/18310S15/chernoff-notes.pdf`.

Gong, M., K. Zhang, T. Liu, D. Tao, C. Glymour, and B. Schölkopf. “Domain adaptation with conditional transferable components”. In: *International Conference on Machine Learning.* 2016.

Goodfellow, I., J. Shlens, and C. Szegedy. “Explaining and Harnessing Adversarial Examples”. In: *International Conference on Learning Representations.* 2015.

Goudet, O. et al. “Learning Functional Causal Models with Generative Neural Networks”. In: *arXiv preprint arXiv:1709.05321* (2017).

Gretton, A., K. Fukumizu, C. Teo, L. Song, B. Schölkopf, and A. Smola. “A Kernel Statistical Test of Independence”. In: *Advances in Neural Information Processing Systems 20 (NIPS).* MIT Press, 2008, pp. 585–592.

Gretton, A., O. Bousquet, A. Smola, and B. Schölkopf. “Measuring Statistical Dependence with Hilbert-Schmidt Norms”. In: *Algorithmic Learning Theory.* Springer-Verlag, 2005, pp. 63–78.

Haavelmo, T. “The Probability Approach in Econometrics”. In: *Econometrica* 12 (1944), S1–S115 (supplement).

Hampel, F., E. Ronchetti, P. Rousseeuw, and W. Stahel. *Robust statistics: the approach based on influence functions.* Wiley series in probability and mathematical statistics. Probability and mathematical statistics. Wiley, 2005.

Harris, N. and M. Drton. “PC algorithm for nonparanormal graphical models”. In: *Journal of Machine Learning Research* 14 (2013), pp. 3365–3383.

Hauser, A. and P. Bühlmann. “Characterization and greedy learning of interventional Markov equivalence classes of directed acyclic graphs”. In: *Journal of Machine Learning Research* 13 (2012), pp. 2409–2464.

Hauser, A. and P. Bühlmann. “Jointly interventional and observational data: estimation of interventional Markov equivalence classes of directed acyclic graphs”. In: *Journal of the Royal Statistical Society, Series B* 77 (2015), pp. 291–318.

He, K., X. Zhang, S. Ren, and J. Sun. “Deep Residual Learning for Image Recognition”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), pp. 770–778.

He, K., X. Zhang, S. Ren, and J. Sun. “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification”. In: *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)* (2015), pp. 1026–1034.

Heckerman, D. *A Tutorial on Learning With Bayesian Networks*. Tech. rep. Microsoft Research (MSR-TR-95-06), 1997.

Heinze, C., B. McWilliams, N. Meinshausen, and G. Krummenacher. “LOCO: Distributing Ridge Regression with Random Projections”. In: *arXiv preprint arXiv:1406.3469* (2014).

Heinze, C., B. McWilliams, and N. Meinshausen. “DUAL-LOCO: Distributing Statistical Estimation Using Random Projections”. In: *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*. PMLR, 2016, pp. 875–883.

Heinze-Deml, C. *backShift: Learning Causal Cyclic Graphs from Unknown Shift Interventions*. R package version 0.1.4.1. 2017. URL: `https://github.com/christinaheinze/backShift`.

Heinze-Deml, C. and N. Meinshausen. *CompareCausalNetworks: Interface to Diverse Estimation Methods of Causal Networks*. R package version 0.1.6. 2017. URL: `https://github.com/christinaheinze/CompareCausalNetworks`.

Heinze-Deml, C. and N. Meinshausen. “Conditional Variance Penalties and Domain Shift Robustness”. In: *arXiv preprint arXiv:1710.11469* (2017).

Heinze-Deml, C., M. H. Maathuis, and N. Meinshausen. “Causal Structure Learning”. In: *Annual Review of Statistics and Its Application* 5.1 (2018), pp. 371–391.

Heinze-Deml, C., J. Peters, and N. Meinshausen. “Invariant Causal Prediction for Nonlinear Models”. In: *Journal of Causal Inference* 6 (2 2018).

Heinze-Deml, C., B. McWilliams, and N. Meinshausen. “Preserving Privacy Between Features in Distributed Estimation”. In: *Stat.* 7 (1 2018).

Higgins, I. et al. “beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework”. In: *International Conference on Learning Representations* (2017).

Hirschman, C. “Why fertility changes”. In: *Annual review of sociology* (1994), pp. 203–233.

Hoeffding, W. “A Non-Parametric Test of Independence”. In: *The Annals of Mathematical Statistics* 19 (Dec. 1948), pp. 546–557.

Hoover, K. D. “The logic of causal inference”. In: *Economics and Philosophy* 6 (1990), pp. 207–234.

Hothorn, T. and A. Zeileis. “partykit: A Modular Toolkit for Recursive Partytioning in R”. In: *Journal of Machine Learning Research* 16 (2015), pp. 3905–3909.

Hoyer, P. O., S. Shimizu, A. J. Kerminen, and M. Palviainen. “Estimation of causal effects using linear non-Gaussian causal models with hidden variables”. In: *International Journal of Approximate Reasoning* 49 (2008), pp. 362–378.

Huang, Z., S. Mitra, and N. Vaidya. “Differentially private distributed optimization”. In: *Proceedings of the 2015 International Conference on Distributed Computing and Networking*. ACM. New York, NY, USA: ACM, 2015, 4:1–4:10.

Huinink, J., M. Kohli, and J. Ehrhardt. “Explaining fertility: The potential for integrative approaches”. In: *Demographic Research* 33 (2015), p. 93.

Hyttinen, A., F. Eberhardt, and P. O. Hoyer. “Learning Linear Cyclic Causal Models with Latent Variables”. In: *Journal of Machine Learning Research* 13 (2012), pp. 3387–3439.

Imbens, G. “Instrumental Variables: An Econometrician’s Perspective”. In: *Statistical Science* 29 (2014), pp. 323–358.

Jackson, A. L. et al. “Expression profiling reveals off-target gene regulation by RNAi”. In: *Nature Biotechnology* 21 (2003), pp. 635–637.

Jaggi, M., V. Smith, M. Takác, J. Terhorst, S. Krishnan, T. Hofmann, and M. I. Jordan. “Communication-efficient distributed dual coordinate ascent”. In: *Advances in Neural Information Processing Systems*. 2014, pp. 3068–3076.

Kabán, A. “New Bounds on Compressive Linear Least Squares Regression”. In: *Artificial Intelligence and Statistics*. 2014.

Kalisch, M. and P. Bühlmann. “Estimating High-Dimensional Directed Acyclic Graphs with the PC-Algorithm”. In: *Journal of Machine Learning Research* 8 (2007), pp. 613–636.

Kalisch, M., M. Mächler, D. Colombo, M. H. Maathuis, and P. Bühlmann. “Causal inference using graphical models with the R package `pcalg`”. In: *Journal of Statistical Software* 47.11 (2012), pp. 1–26.

Kasiviswanathan, S. P., H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith. “What Can We Learn Privately?” In: *2008 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 531–540.

Kenthapadi, K., A. Korolova, I. Mironov, and N. Mishra. “Privacy via the Johnson-Lindenstrauss Transform”. In: *Journal of Privacy and Confidentiality* 5.1 (2013), pp. 39–71.

Kilbertus, N., M. Rojas Carulla, G. Parascandolo, M. Hardt, D. Janzing, and B. Schölkopf. “Avoiding Discrimination through Causal Reasoning”. In: *Advances in Neural Information Processing Systems 30* (2017), pp. 656–666.

Kingma, D. P. and J. Ba. “Adam: A Method for Stochastic Optimization.” In: *International Conference on Learning Representations (ICLR)* (2015).

Knight, W. “Artificial Intelligence’s White Guy Problem”. In: *MIT Technology Review, January 27 2016* (2016). `https://www.technologyreview.com/s/546066/googles-ai-masters-the-game-of-go-a-decade-earlier-than-expected/`.

Knutti, R., D. Masson, and A. Gettelman. “Climate model genealogy: Generation CMIP5 and how we got there”. In: *Geophysical Research Letters* 40.6 (2013), pp. 1194–1199.

Kocaoglu, M., C. Snyder, A. G. Dimakis, and S. Vishwanath. “CausalGAN: Learning Causal Implicit Generative Models with Adversarial Training”. In: *International Conference on Learning Representations* (2018).

Korb, K., L. Hope, A. Nicholson, and K. Axnick. “Varieties of causal intervention”. In: *Proceedings of the Pacific Rim Conference on AI.* 2004, pp. 322–331.

Krizhevsky, A., I. Sutskever, and G. E Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems 25.* 2012.

Kulkarni, M. M., M. Booker, S. J. Silver, A. Friedman, P. Hong, N. Perrimon, and B. Mathey-Prevot. “Evidence of off-target effects associated with long dsRNAs in Drosophila melanogaster cell-based assays”. In: *Nature Methods* 3 (2006), pp. 833–838.

Künsch, H.-R. “The jackknife and the bootstrap for general stationary observations”. In: *The Annals of Statistics* 17 (1989), pp. 1217–1241.

Lacerda, G., P. Spirtes, J. Ramsey, and P. Hoyer. “Discovering cyclic causal models by independent components analysis”. In: *Proceedings of the 24th Conference on Uncertainty in Artificial Intelligence (UAI).* 2008, pp. 366–374.

Lauritzen, S. L. and T. S. Richardson. “Chain graph models and their causal interpretations”. In: *Journal of the Royal Statistical Society, Series B* 64 (2002), pp. 321–348.

Lauritzen, S. *Graphical Models.* Oxford University Press, 1996.

LeCun, Y., L. Bottou, Y. Bengio, and P. Haffner. “Gradient-Based Learning Applied to Document Recognition”. In: *Proceedings of the IEEE* (1998).

Levene, H. “Robust tests for equality of variances”. In: *Contributions to Probability and Statistics.* Ed. by I. Olkin. Palo Alto, CA: Stanford University Press, 1960, pp. 278–292.

Li, Y., X. Jiang, S. Wang, H. Xiong, and L. Ohno-Machado. “VERTIcal Grid lOgistic regression (VERTIGO)”. In: *Journal of the American Medical Informatics Association* 23.3 (2015), pp. 570–579.

Liu, Q. and A. T. Ihler. “Distributed Estimation, Information Loss and Exponential Families”. In: *Advances in Neural Information Processing Systems*. 2014, pp. 1098–1106.

Liu, Z., P. Luo, X. Wang, and X. Tang. “Deep Learning Face Attributes in the Wild”. In: *Proceedings of International Conference on Computer Vision (ICCV)*. 2015.

Loh, P.-L. and M. J. Wainwright. “High-dimensional regression with noisy and missing data: Provable guarantees with nonconvexity”. In: *The Annals of Statistics* 40.3 (June 2012), pp. 1637–1664.

Lopez-Paz, D. and M. Oquab. “Revisiting Classifier Two-Sample Tests”. In: *International Conference on Learning Representations (ICLR)* (2017).

Lopez-Paz, D., R. Nishihara, S. Chintala, B. Schölkopf, and L. Bottou. “Discovering Causal Signals in Images”. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017)*. 2017.

Louizos, C., U. Shalit, J. M. Mooij, D. Sontag, R. Zemel, and M. Welling. “Causal Effect Inference with Deep Latent-Variable Models”. In: *Advances in Neural Information Processing Systems* (2017).

Lu, Y., P. Dhillon, D. P. Foster, and L. Ungar. “Faster Ridge Regression via the Subsampled Randomized Hadamard Transform”. In: *Advances in Neural Information Processing Systems 26*. 2013, pp. 369–377.

Ma, C., V. Smith, M. Jaggi, M. I. Jordan, P. Richtárik, and M. Takáč. “Adding vs. averaging in distributed primal-dual optimization”. In: *Proceedings of The 32nd International Conference on Machine Learning*. 2015.

Maathuis, M. H., M. Kalisch, and P. Bühlmann. “Estimating high-dimensional intervention effects from observational data”. In: *Annals of Statistics* 37 (2009), pp. 3133–3164.

Maathuis, M. H., D. Colombo, M. Kalisch, and P. Bühlmann. “Predicting causal effects in large-scale systems from observational data.” In: *Nature Methods* 7 (2010), pp. 247–248.

Magliacane, S., T. van Ommen, T. Claassen, S. Bongers, P. Versteeg, and J. M. Mooij. “Domain Adaptation by Using Causal Inference to Predict Invariant Conditional Distributions”. In: *Advances in Neural Information Processing Systems* (2018).

Mahoney, M. W. “Randomized algorithms for matrices and data”. In: *arXiv preprint arXiv:1104.5557v3* (2011).

Mangasarian, O. L., E. W. Wild, and G. M. Fung. “Privacy-preserving classification of vertically partitioned data via random kernels”. In: *ACM TKDD* 2.3 (2008), p. 12.

Marx, V. “The big challenges of big data”. In: *Nature* 498 (2013), 255–260.

Matsuo, T., H. Fukuhara, and N. Shimada. “Transform invariant autoencoder.” In: *IROS* (2017), pp. 2359–2364.

McGregor, A., I. Mironov, T. Pitassi, O. Reingold, K. Talwar, and S. Vadhan. “The Limits of Two-Party Differential Privacy”. In: *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*. Washington, DC, USA: IEEE Computer Society, 2010, pp. 81–90.

McWilliams, B., G. Krummenacher, M. Lucic, and J. M. Buhmann. “Fast and robust least squares estimation in corrupted linear models”. In: *Advances in Neural Information Processing Systems*. 2014, pp. 415–423.

Meinshausen, N. “Quantile Regression Forests”. In: *Journal of Machine Learning Research* 7 (2006), pp. 983–999.

Meinshausen, N. and P. Bühlmann. “Stability selection (with discussion)”. In: *Journal of the Royal Statistical Society, Series B* 72 (2010), pp. 417–473.

Miller, L. D., J. Smeds, J. George, and V. B. o. Vega. “An expression signature for p53 status in human breast cancer predicts mutation status, transcriptional effects, and patient survival”. In: *Proceedings of the National Academy of Sciences of the United States of America* 102.38 (2005), pp. 13550–13555.

Mirza, M. and S. Osindero. “Conditional Generative Adversarial Nets”. In: *arXiv preprint arXiv:1411.1784* (2014).

Mohammed, N., D. Alhadidi, B. C. M. Fung, and M. Debbabi. “Secure Two-Party Differentially Private Data Release for Vertically Partitioned Data”. In: *IEEE Trans. Dependable Sec. Comput.* 11.1 (2014), pp. 59–71.

Mooij, J. M. and T. Heskes. “Cyclic Causal Discovery from Continuous Equilibrium Data”. In: *Proceedings of the 29th Annual Conference on Uncertainty in Artificial Intelligence (UAI)*. 2013, pp. 431–439.

Mooij, J. M., D. Janzing, T. Heskes, and B. Schölkopf. “On Causal Discovery with Cyclic Additive Noise Models”. In: *Advances in Neural Information Processing Systems 24 (NIPS)*. 2011, pp. 639–647.

Namkoong, H. and J. Duchi. “Variance-based Regularization with Convex Objectives”. In: *Advances in Neural Information Processing Systems*. 2017, pp. 2975–2984.

Nandy, P., M. H. Maathuis, and T. S. Richardson. “Estimating the effect of joint interventions from observational data in high-dimensional settings”. In: *Annals of Statistics* 45 (2017), pp. 647–674.

Nandy, P., A. Hauser, and M. H. Maathuis. “High-dimensional consistency in score-based and hybrid structure learning”. In: *Annals of Statistics, to appear* (2017).

Ohno-Machado, L. “To Share or Not To Share: That Is Not the Question”. In: *Science Translational Medicine* 4.165 (2012).

Olah, C., A. Mordvintsev, and L. Schubert. “Feature Visualization”. In: *Distill* (2017). URL: `https://distill.pub/2017/feature-visualization`.

Pearl, J. “A constraint propagation approach to probabilistic reasoning”. In: *Proceedings of the 4th Annual Conference on Uncertainty in Artificial Intelligence (UAI)*. 1985, pp. 31–42.

Pearl, J. *Causality: Models, Reasoning, and Inference*. 2nd. New York, USA: Cambridge University Press, 2009.

Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA: Morgan Kaufmann Publishers Inc., 1988.

Peters, J. and P. Bühlmann. “Identifiability of Gaussian Structural Equation Models with Equal Error Variances”. In: *Biometrika* 101 (2014), pp. 219–228.

Peters, J., D. Janzing, and B. Schölkopf. “Causal Inference on Time Series using Structural Equation Models”. In: *Advances in Neural Information Processing Systems 26 (NIPS)* (2013), pp. 585–592.

Peters, J., P. Bühlmann, and N. Meinshausen. “Causal inference using invariant prediction: identification and confidence intervals”. In: *Journal of the Royal Statistical Society, Series B* 78 (2016), pp. 947–1012.

Peters, J., D. Janzing, and B. Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms.* Cambridge, MA, USA: MIT Press, 2017.

Pfister, N. and J. Peters. *dHSIC: Independence Testing via Hilbert Schmidt Independence Criterion.* R package version 2.0. 2017. URL: `https://CRAN.R-project.org/package=dHSIC`.

Pilanci, M. and M. J. Wainwright. "Newton Sketch: A Near Linear-Time Optimization Algorithm with Linear-Quadratic Convergence". In: *SIAM Journal on Optimization* 27.1 (2017), pp. 205–245.

Que, J., X. Jiang, and L. Ohno-Machado. "A collaborative framework for distributed privacy-preserving support vector machine learning". In: *AMIA Annual Symposium Proceedings.* 2012, pp. 1350–1359.

Quionero-Candela, J., M. Sugiyama, A. Schwaighofer, and N. D. Lawrence. *Dataset Shift in Machine Learning.* The MIT Press, 2009.

R Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing. Vienna, Austria, 2017. URL: `https://www.R-project.org/`.

Raftery, A., S. Lewis, and A. Aghajanian. "Demand or ideation? Evidence from the Iranian marital fertility decline". In: *Demography* 32.2 (1995), pp. 159–182.

Rahimi, A and B Recht. "Random features for large-scale kernel machines". In: *Advances in Neural Information Processing Systems* 20 (2007), pp. 1177–1184.

Recht, B., C. Re, S. Wright, and F. Niu. "Hogwild: A Lock-Free Approach to Parallelizing Stochastic Gradient Descent". In: *Advances in Neural Information Processing Systems 24.* Curran Associates, Inc., 2011, pp. 693–701.

Rényi, A. "On measures of dependence". In: *Acta Mathematica Academiae Scientiarum Hungarica* 10 (1959), pp. 441–451.

Richardson, T. and J. M. Robins. "Single World Intervention Graphs (SWIGs): A unification of the counterfactual and graphical approaches to causality". In: *Center for the Statistics and the Social Sciences, University of Washington Series. Working Paper 128, 30 April 2013* (2013).

Richardson, T. and P. Spirtes. “Ancestral graph Markov models”. In: *Annals of Statistics* 30 (2002), pp. 962–1030.

Richardson, T. and P. Spirtes. “Automated discovery of linear feedback models”. In: *Computation, Causation, and Discovery.* Ed. by C. Glymour and G. Cooper. MIT Press, 1999, pp. 253–304.

Robins, J. M. “A new approach to causal inference in mortality studies with a sustained exposure period – application to control of the healthy worker survivor effect”. In: *Mathematical Modelling* 7 (1986), pp. 1393 –1512.

Rojas-Carulla, M., B. Schölkopf, R. Turner, and J. Peters. “Causal Transfer in Machine Learning”. In: *To appear in Journal of Machine Learning Research.* (2018).

Rothenhäusler, D., P. Bühlmann, N. Meinshausen, and J. Peters. “Anchor regression: heterogeneous data meets causality”. In: *arXiv preprint arXiv:1801.06229* (2018).

Rothenhäusler, D., C. Heinze, J. Peters, and N. Meinshausen. “backShift: Learning causal cyclic graphs from unknown shift interventions”. In: *Advances in Neural Information Processing Systems 28 (NIPS).* 2015, pp. 1513–1521.

Rothenhäusler, D., J. Ernest, and P. Bühlmann. “Causal inference in partially linear structural equation models”. In: *The Annals of Statistics* 46 (2018), pp. 2904–2938.

Rubin, D. B. “Causal inference using potential outcomes”. In: *Journal of the American Statistical Association* 100 (2005), pp. 322–331.

Sachs, K., O. Perez, D. Pe’er, D. Lauffenburger, and G. Nolan. “Causal protein-signaling networks derived from multiparameter single-cell data”. In: *Science* 308 (2005), pp. 523–529.

Sarwate, A. D., S. M. Plis, J. A. Turner, M. R. Arbabshirani, and V. D. Calhoun. “Sharing privacy-sensitive access to neuroimaging and genetics data: a review and preliminary validation”. In: *Frontiers in neuroinformatics* 8 (2014).

Scheines, R., F. Eberhardt, and P. Hoyer. “Combining Experiments to Discover Linear Cyclic Models with Latent Variables”. In: *International Conference on Artificial Intelligence and Statistics (AISTATS).* 2010, pp. 185–192.

Schmidt, G. A. et al. “Configuration and assessment of the GISS ModelE2 contributions to the CMIP5 archive”. In: *Journal of Advances in Modeling Earth Systems* 6.1 (2014), pp. 141–184.

Schölkopf, B., D. Janzing, J. Peters, E. Sgouritsa, K. Zhang, and J. Mooij. “On causal and anticausal learning”. In: *Proceedings of the 29th International Conference on Machine Learning (ICML).* 2012, pp. 1255–1262.

Schölkopf, B., C. Burges, and V. Vapnik. “Incorporating invariances in support vector learning machines”. In: *Artificial Neural Networks — ICANN 96.* Berlin, Heidelberg: Springer Berlin Heidelberg, 1996, pp. 47–52.

Scutari, M. “Learning Bayesian Networks with the bnlearn R Package”. In: *Journal of Statistical Software* 35.3 (2010), pp. 1–22.

Sermanet, P., D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. “OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks”. In: *International Conference on Learning Representations (ICLR 2014).* CBLS, 2014.

Shafieezadeh-Abadeh, S., D. Kuhn, and P. Esfahani. “Regularization via mass transportation”. In: *arXiv preprint arXiv:1710.10016* (2017).

Shah, R. D. and J. Peters. “The Hardness of Conditional Independence Testing and the Generalised Covariance Measure”. In: *arXiv preprint arXiv:1804.07203* (2018).

Shah, R. D. and P. Bühlmann. “Goodness-of-fit tests for high dimensional linear models”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 80.1 (2018), pp. 113–135.

Shalev-Shwartz, S. and T. Zhang. “Stochastic dual coordinate ascent methods for regularized loss”. In: *The Journal of Machine Learning Research* 14.1 (2013), pp. 567–599.

Sheffet, O. “Differentially Private Ordinary Least Squares”. In: *Proceedings of the 34th International Conference on Machine Learning.* Ed. by D. Precup and Y. W. Teh. PMLR, 2017, pp. 3105–3114.

Shimizu, S., P. O. Hoyer, A. Hyvärinen, and A. J. Kerminen. “A linear non-Gaussian acyclic model for causal discovery”. In: *Journal of Machine Learning Research* 7 (2006), pp. 2003–2030.

Shimizu, S. et al. “DirectLiNGAM: A Direct Method for Learning a Linear Non-Gaussian Structural Equation Model”. In: *Journal of Machine Learning Research* 12 (2011), pp. 1225–1248.

Silver, D. et al. “Mastering the Game of Go with Deep Neural Networks and Tree Search”. In: *Nature* 529.7587 (2016), pp. 484–489.

Simpson, E. H. “The interpretation of interaction in contingency tables”. In: *Journal of the Royal Statistical Society. Series B* 13.2 (1951), pp. 238–241.

Sinha, A., H. Namkoong, and J. Duchi. “Certifiable Distributional Robustness with Principled Adversarial Training”. In: *International Conference on Learning Representations*. 2018.

Smith, A., A. Thakurta, and J. Upadhyay. “Is Interaction Necessary for Distributed Private Learning?” In: *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2017, pp. 58–77.

Song, S., K. Chaudhuri, and A. D. Sarwate. “Stochastic gradient descent with differentially private updates”. In: *GlobalSIP*. IEEE. 2013, pp. 245–248.

Spirtes, P., C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. 2nd. Cambridge, USA: MIT Press, 2000.

Spirtes, P., C. Meek, and T. Richardson. “Computation, Causation and Discovery”. In: MIT Press, 1999. Chap. An algorithm for causal inference in the presence of latent variables and selection bias, pp. 211–252.

Stekhoven, D., I. Moraes, G. Sveinbjörnsson, L. Hennig, M. Maathuis, and P. Bühlmann. “Causal Stability Ranking”. In: *Bioinformatics* 28 (2012), pp. 2819–2823.

Szegedy, C. et al. “Going Deeper with Convolutions”. In: *Computer Vision and Pattern Recognition (CVPR)*. 2015.

Szegedy, C., W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. “Intriguing properties of neural networks”. In: *International Conference on Learning Representations*. 2014.

Székely, G. J., M. L. Rizzo, and N. K. Bakirov. “Measuring and testing dependence by correlation of distances”. In: *The Annals of Statistics* 35 (2007), pp. 2769–2794.

Tian, J. and J. Pearl. “Causal discovery from changes”. In: *Proceedings of the 17th Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI)*. 2001, pp. 512–522.

Torralba, A. and A. A. Efros. “Unbiased look at dataset bias”. In: *Computer Vision and Pattern Recognition (CVPR)*. 2011.

Tropp, J. A. “Improved Analysis of the subsampled Randomized Hadamard Transform”. In: *Advances in Adaptive Data Analysis* 3.1-2 (2011), pp. 115–126.

Tropp, J. A. “User-friendly tail bounds for sums of random matrices”. In: *arXiv preprint arXiv:1004.4389v7* (2010).

Tsamardinos, I., L. E. Brown, and C. F. Aliferis. “The max-min hill-climbing Bayesian network structure learning algorithm”. In: *Machine Learning* 65 (2006), pp. 31–78.

United Nations. “World Population Prospects: The 2012 Revision”. In: *Population Division, Department of Economic and Social Affairs, United Nations, New York* (2013). URL: `"https://esa.un.org/unpd/wpp/Download/Standard/ASCII/"`.

Vershynin, R. “Introduction to the non-asymptotic analysis of random matrices”. In: *arXiv preprint arXiv:1011.3027* (Nov. 2010).

Williams, C. K. I. and M. Seeger. “Using the Nyström Method to Speed Up Kernel Machines”. In: *Advances in Neural Information Processing Systems 13*. MIT Press, 2001, pp. 682–688.

Wilson, E. B. “Probable Inference, the Law of Succession, and Statistical Inference”. In: *Journal of the American Statistical Association* 22 (1927), pp. 209–212.

Wright, D. “The Method of Path Coefficients”. In: *Annals of Mathematical Statistics* 5 (1934), pp. 161–215.

Wright, S. “Correlation and Causation”. In: *Journal of Agricultural Research* 20 (1921), pp. 557–585.

Wu, Y. et al. “Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation”. In: *arXiv preprint arXiv:1609.08144* (2016).

Wu, Y., X. Jiang, J. Kim, and L. Ohno-Machado. “Grid Binary LOgistic REgression (GLORE): building shared models without sharing data". In: *Journal of the American Medical Informatics Association* 19.5 (2012), pp. 758–764.

Xian, Y., C. H. Lampert, B. Schiele, and Z. Akata. "Zero-Shot Learning - A Comprehensive Evaluation of the Good, the Bad and the Ugly". In: *arXiv preprint arXiv:1707.00600* (2017).

Xu, H., C. Caramanis, and S. Mannor. "Robust regression and lasso". In: *Advances in Neural Information Processing Systems*. 2009, pp. 1801–1808.

Yu, H., J. Vaidya, and X. Jiang. "Privacy-preserving svm classification on vertically partitioned data". In: *Advances in Knowledge Discovery and Data Mining*. Berlin, Heidelberg: Springer, 2006, pp. 647–656.

Yu, X., T. Liu, M. Gong, K. Zhang, and D. Tao. "Transfer Learning with Label Noise". In: *arXiv preprint arXiv:1707.09724* (2017).

Yudkowsky, E. "Artificial Intelligence as a Positive and Negative Factor in Global Risk". In: *Global catastrophic risks* 1 (2008).

Zeileis, A., T. Hothorn, and K. Hornik. "Model-Based Recursive Partitioning". In: *Journal of Computational and Graphical Statistics* 17.2 (2008), pp. 492–514.

Zhang, J. "Causal reasoning with ancestral graphs". In: *Journal of Machine Learning Research* 9 (2008), pp. 1437–1474.

Zhang, J. "On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias". In: *Artificial Intelligence* 172 (2008), pp. 1873–1896.

Zhang, K., B. Schölkopf, K. Muandet, and Z. Wang. "Domain Adaptation under Target and Conditional Shift." In: *International Conference on Machine Learning*. 2013.

Zhang, K., J. Peters, D. Janzing, and B. Schölkopf. "Kernel-based Conditional Independence Test and Application in Causal Discovery". In: *Proceedings of the 27th Annual Conference on Uncertainty in Artificial Intelligence (UAI)*. 2011.

Zhang, K., M. Gong, and B. Schölkopf. "Multi-Source Domain Adaptation: A Causal View". In: *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*. 2015.

Zhang, L., M. Mahdavi, R. Jin, T. Yang, and S. Zhu. “Random Projections for Classification: A Recovery Approach”. In: *IEEE Transactions on Information Theory* 60.11 (2014), pp. 7300–7316.

Zhang, L., M. Mahdavi, R. Jin, T. Yang, and S. Zhu. “Recovering optimal solution by dual random projection”. In: *arXiv preprint arXiv:1211.3046* (2012).

Zhang, T. and Q. Zhu. “Dynamic Privacy For Distributed Machine Learning Over Network”. In: *arXiv preprint arXiv:1601.03466* (2016).

Zhang, Y., J. Duchi, M. Jordan, and M. J. Wainwright. “Information-theoretic lower bounds for distributed statistical estimation with communication constraints”. In: *Advances in Neural Information Processing Systems*. 2013, pp. 2328–2336.

Zhang, Y., J. Duchi, and M. Wainwright. “Divide and Conquer Kernel Ridge Regression: A Distributed Algorithm with Minimax Optimal Rates”. In: *Journal of Machine Learning Research* 16.1 (Jan. 2015), pp. 3299–3340.

Ziehe, A., P. Laskov, G. Nolte, and K.-R. Müller. “A Fast Algorithm for Joint Diagonalization with Non-orthogonal Transformations and its Application to Blind Source Separation”. In: *Journal of Machine Learning Research* 5 (2004), pp. 801–818.

Curriculum vitae

Name	Christina Heinze-Deml
Date of birth	April 10, 1989

Education

2014 – 2018	Doctoral studies in Statistics at ETH Zurich
2012 – 2014	Master studies in Statistics at ETH Zurich
2009 – 2012	Bachelor studies in Social Sciences at Roosevelt Academy (Utrecht University), Netherlands
2008	Abitur at Jungmannschule Eckernförde, Germany

Work experience

08/2018 – 11/2018	Internship at DeepMind, London, United Kingdom
01/2018 – 03/2018	Internship at Facebook AI Research, New York, USA
09/2014 – 11/2014	Internship at Teralytics AG, Zurich

Working papers

C. Heinze-Deml and N. Meinshausen. “Conditional Variance Penalties and Domain Shift Robustness”. In: *arXiv preprint arXiv:1710.11469* (2017)

Publications

C. Heinze-Deml, J. Peters, and N. Meinshausen. “Invariant Causal Prediction for Nonlinear Models”. In: *Journal of Causal Inference* 6 (2 2018)

C. Heinze-Deml, B. McWilliams, and N. Meinshausen. “Preserving Privacy Between Features in Distributed Estimation”. In: *Stat.* 7 (1 2018)

C. Heinze-Deml, M. H. Maathuis, and N. Meinshausen. “Causal Structure Learning”. In: *Annual Review of Statistics and Its Application* 5.1 (2018), pp. 371–391

C. Heinze, B. McWilliams, and N. Meinshausen. “DUAL-LOCO: Distributing Statistical Estimation Using Random Projections”. In: *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*. PMLR, 2016, pp. 875–883

D. Rothenhäusler, C. Heinze, J. Peters, and N. Meinshausen. “backShift: Learning causal cyclic graphs from unknown shift interventions”. In: *Advances in Neural Information Processing Systems 28 (NIPS)*. 2015, pp. 1513–1521

Talks

04/2017	Preserving Differential Privacy Between Features in Distributed Estimation, Fairness and Privacy in Machine Learning workshop, DALI 2017, Tenerife, Spain
05/2016	DUAL-LOCO: Distributing Statistical Estimation Using Random Projections, AISTATS, Cádiz, Spain
03/2016	Distributed Estimation With Random Projections, MFO workshop, Oberwolfach, Germany
09/2015	Distributed Machine Learning with Apache Spark, Zurich Machine Learning and Data Science Meetup, Zurich

Software[4]

R packages	backShift, CompareCausalNetworks, nonlinearICP, CondIndTests
TensorFlow	CoRe
Spark	LOCO$^{\text{lib}}$

Honors and awards

2015 – 2018	Associate Doctoral Fellow at Max Planck ETH Center for Learning Systems
2009 – 2014	Scholarship of the German National Merit Foundation

---

[4] Available from `https://github.com/christinaheinze/core`.

What of the future? The future of data analysis can involve great progress, the overcoming of real difficulties, and the provision of a great service to all fields of science and technology. Will it? That remains to us, to our willingness to take up the rocky road of real problems in preferences to the smooth road of unreal assumptions, arbitrary criteria, and abstract results without real attachments. Who is for the challenge?.

– Tukey, The Future of Data Analysis, 1962