

# **Applied Causal Inference Powered by ML and AI**

Victor Chernozhukov\*, Christian Hansen†, Nathan Kallus‡, Martin Spindler§, Vasilis Syrgkanis ¶

July 7, 2022

Publisher: Online

\* MIT

† Chicago Booth

‡ Cornell

§ Hamburg University

¶ Microsoft Research

© Victor Chernozhukov, Christian Hansen, Martin Spindler, Vasilis Syrgkanis

**Colophon**

This document was typeset with the help of KOMA-Script and L<sup>A</sup>T<sub>E</sub>X using the kaobook class.

The source code of this book is available at:

<https://github.com/fmarotta/kaobook>

**Publisher**

First printed in September 2022 by the Authors for Online Distribution

# Contents

<b>Contents</b>	<b>iii</b>
<b>Preface</b>	<b>2</b>
<b>CORE MATERIAL</b>	<b>4</b>
<b>1 Predictive Inference with Linear Regression in Moderately High Dimensions</b>	<b>5</b>
1.1 Foundation of Linear Regression . . . . .	6
Regression and the Best Linear Prediction Problem . . . . .	6
The Best Linear Prediction Problem in Finite Samples . . . . .	7
Properties of Sample Linear Regression . . . . .	8
Analysis of Variance (ANOVA) . . . . .	9
Overfitting: What happens when $p/n$ is not small. . . . .	10
Measuring Predictive Ability by Sample Splitting . . . . .	11
1.2 Inference about Predictive Effects or Association . . . . .	13
Understanding $\beta_1$ via "Partialling-Out" . . . . .	13
Adaptive Inference . . . . .	15
1.3 Application: Wage Equation . . . . .	17
Prediction of Wages . . . . .	18
Gender Wage Gap . . . . .	19
1.4 Notes . . . . .	22
<b>2 Causal Inference via Randomized Experiments</b>	<b>25</b>
2.1 Potential Outcomes Framework and Average Treatment Effects . .	26
Random Assignment/ Randomized Control Trials . . . . .	29
Statistical Inference with two sample means . . . . .	30
Pfizer/BioNTech Covid Vaccine RCT . . . . .	31
2.2 Pre-treatment Covariates and Heterogeneity . . . . .	33
Regression and Statistical Inference for ATEs . . . . .	35
The Role of Covariates: Improve Precision of Estimating ATE . . .	36
The Role of Covariates: Discover Heterogeneity through CATE . .	37
Reemployment Bonus RCT . . . . .	38
2.3 Drawing RCTs via Causal Diagrams . . . . .	39

2.4	The limitations of RCTs . . . . .	40
	Externalities, Stability, and Equilibrium Effects . . . . .	41
	Ethical, Practical and Generalizability Concerns . . . . .	41
2.A	Approximate Distribution of the Two Sample Means . . . . .	44
2.B	Approximate Distribution for Intercept and Slope Estimators . . . . .	45
<b>3</b>	<b>Predictive Inference via Modern High Dimensional Linear Regression</b>	<b>49</b>
3.1	Linear Regression with High-Dimensional Covariates . . . . .	50
	The Framework . . . . .	50
	Lasso . . . . .	54
3.2	Predictive Performance of Lasso and Post-Lasso . . . . .	59
3.3	A Helicopter Tour of Other Penalized Regression Methods for Prediction . . . . .	61
3.4	Choice of Regression Methods in Practice . . . . .	66
3.A	Additional Discussion and Results . . . . .	68
	Iterative Estimation of $\sigma$ . . . . .	68
	Some Lasso Heuristics via Convex Geometry* . . . . .	68
	Other Variations on Lasso . . . . .	70
	Cross-Validation . . . . .	71
<b>4</b>	<b>Statistical Inference on Predictive and Causal Effects in High Dimensional Linear Regression Models</b>	<b>75</b>
4.1	Introduction . . . . .	76
4.2	Inference with Double Lasso . . . . .	76
	Inference on One Coefficient . . . . .	76
	Application to Testing the Convergence Hypothesis . . . . .	79
	Inference on Many Coefficients . . . . .	80
	Discovering Heterogeneity in the Wage Pay Gap Analysis . . . . .	82
4.3	Why Partialling-out works: Neyman Orthogonality . . . . .	84
	Neyman Orthogonality . . . . .	84
	What happens if we don't have Neyman Orthogonality? . . . . .	86
4.4	Other Approaches that have the Neyman Orthogonality Property . . . . .	89
	Double Selection . . . . .	89
	Desparsified Lasso . . . . .	89
<b>5</b>	<b>Causal Inference via Conditional Exogeneity</b>	<b>94</b>
5.1	Introduction . . . . .	95
5.2	Potential Outcomes Framework and Ignorability . . . . .	96
	Identification by Conditioning . . . . .	96
	Conditional Ignorability via Causal Diagrams . . . . .	99
5.3	Identification Using Propensity Scores . . . . .	100
	Conditioning On Propensity Scores . . . . .	100
	Identification by Propensity Score Reweighting . . . . .	100
	Stratified RCTs . . . . .	101
	Covariate Balance Checks . . . . .	102

5.4	Average Treatment Effect for Groups and on the Treated* . . . . .	102
5.5	Connection to Linear Regression . . . . .	103
	What if the propensity score is known? . . . . .	105
5.A	Rosenbaum-Rubin's result . . . . .	106
5.B	Details of ATET . . . . .	106
<b>6</b>	<b>Causal Inference via Structural Equations and Conditional Exogeneity</b>	<b>109</b>
6.1	Structural Equation Modelling and Conditional Exogeneity . . . . .	110
	A Simple Triangular Structural Equation Model (SEM) . . . . .	110
6.2	Drawing the Model: Causal Diagrams, aka DAGs . . . . .	113
6.3	When Conditioning Can Go Wrong: Collider Bias aka Heckman Selection Bias . . . . .	115
6.4	Wage Gap Analysis and Discrimination . . . . .	117
6.A	Details of the Wage Discrimination Analysis . . . . .	123
<b>7</b>	<b>Predictive Inference via Modern High Dimensional Nonlinear Regression</b>	<b>127</b>
7.1	Introduction . . . . .	128
7.2	Regression Trees and Random Forests . . . . .	128
	Introduction to Regression Trees . . . . .	128
	Random Forests . . . . .	131
	Boosted Trees . . . . .	133
7.3	Neural Nets / Deep Learning . . . . .	134
	Basic Ideas . . . . .	134
	Deep Neural Networks . . . . .	139
7.4	Prediction Quality of Modern Nonlinear Regression Methods . . . . .	140
	Learning Guarantees of DNNs . . . . .	141
	Learning Guarantees of Trees and Forests . . . . .	143
	Trust but Verify . . . . .	146
	A Simple Case Study using Wage Data . . . . .	147
7.5	Combining Predictions - Aggregation - Ensemble Learning . . . . .	148
	Auto ML Frameworks . . . . .	149
7.6	When do Neural Networks win? . . . . .	150
7.A	Variable Importance via Permutations . . . . .	153
<b>8</b>	<b>Causal Inference via Directed Acyclical Graphs and Nonlinear Structural Equation Models</b>	<b>157</b>
8.1	Introduction . . . . .	158
8.2	From Causal Diagrams to Causal DAGs . . . . .	159
	Identification by Regression . . . . .	161
	Interventions . . . . .	162
8.3	General Acyclic SEMs, Causal DAGs and Counterfactuals . . . . .	164
	DAGs and Acyclic SEMs (ASEMs) . . . . .	164
	General definitions . . . . .	165
	Testable Restrictions by D-Separation . . . . .	168
8.4	Counterfactuals Induced by Interventions . . . . .	170

8.5	Identification by Conditioning . . . . .	172
	Main Idea . . . . .	172
	Useful Adjustment Strategies as Corollaries . . . . .	175
8.6	Falsifiability and Causal Discovery* . . . . .	179
8.A	Counterfactual Distributions via Markov Networks . . . . .	184
8.B	Causal Discovery Algorithms . . . . .	185
	PC Algorithm . . . . .	185
	FCI Algorithm . . . . .	187
<b>9</b>	<b>Statistical Inference on Predictive and Causal Effects in Modern Non-linear Regression Models</b>	<b>191</b>
9.1	Introduction . . . . .	192
9.2	DML Inference in the Partially Linear Regression Model (PLM) . . . . .	193
	Discussion of DML Construction . . . . .	197
	The Effect of Gun Ownership on Gun-Homicide Rates . . . . .	200
9.3	DML Inference in the Interactive Regression Model (IRM) . . . . .	202
	DML Inference on APEs and ATEs . . . . .	202
	DML Inference for GATEs and ATET . . . . .	205
	The effect of 401(k) Eligibility on Net Financial Assets . . . . .	206
9.4	Generic Debiased (or Double) Machine Learning . . . . .	210
	Key Ingredients . . . . .	210
	Neyman Orthogonal Scores for Regression Problems . . . . .	213
	The DML Inference Method . . . . .	214
	Properties of the general DML estimator . . . . .	216
9.5	Bias Bounds with Proxy Treatments . . . . .	221
9.6	Illustrative Neyman Orthogonality Calculations . . . . .	222
<b>10</b>	<b>Feature Engineering With Deep Learning for Causal and Predictive Inference</b>	<b>226</b>
10.1	Introduction . . . . .	227
10.2	From Principal Components to Variational Autoencoders . . . . .	228
10.3	From Auto-Encoders to General Embeddings . . . . .	231
10.4	Text Embeddings . . . . .	232
10.5	Image Embeddings . . . . .	240
10.6	Constructing Hedonic Prices Using Apparel Data . . . . .	242
<b>ADVANCED CORE MATERIAL</b>		<b>248</b>
<b>11</b>	<b>Advanced Core 1: Unobserved Confounders, Instrumental Variables, and Proxy Controls</b>	<b>249</b>
11.1	The Impossibility of Causal Inference with an Unobserved Confounder . . . . .	250
11.2	Impact of Confounders on Causal Effect Identification and Sensitivity Analysis . . . . .	251

11.3	Partially Linear IV Models . . . . .	254
	A Wage Equation with Unobserved Ability . . . . .	254
	Aggregate Market Demand . . . . .	256
	SEMs with Griliches-Chamberlain Proxy Controls . . . . .	257
11.4	Nonlinear IV Models . . . . .	259
	The LATE Model . . . . .	259
	The IV Quantile Model* . . . . .	261
11.5	Nonlinear Models with Proxy Controls* . . . . .	262
11.6	Study Problems . . . . .	264
11.7	Proofs . . . . .	266
	Latent Confounder Bias Result: Theorem 11.2.1 . . . . .	266
	Linear Proxy Model: Theorem 11.3.2. . . . .	267
<b>12</b>	<b>Advanced Core 2: Debiased ML for IV and Proxy Controls Models and Robust DML Inference under Weak Identification</b>	<b>272</b>
12.1	DML Inference in Partially Linear IV Models . . . . .	273
	The Effect of Institutions on Economic Growth . . . . .	275
12.2	DML Inference in the Interactive IV Regression Model (IRM) . . . . .	278
	DML Inference on LATE . . . . .	278
	The effect of 401(k) Participation on Net Financial Assets . . . . .	279
12.3	DML Inference with Weak Instruments . . . . .	281
	Motivation . . . . .	281
	DML Inference Robust to Weak-IV in PLMs . . . . .	283
	The Effect of Institutions on Economic Growth Revisited . . . . .	284
12.4	Generic DML Inference under Weak Identification . . . . .	286
<b>TOPICS</b>	<b>291</b>	
<b>13</b>	<b>Inference on Heterogeneous Treatment Effects</b>	<b>292</b>
13.1	Inference on CATEs and Best Linear Predictors of CATEs . . . . .	293
	Using Least Squares Methods for Learning CATEs . . . . .	295
	Using ML Methods for Learning CATEs . . . . .	297
	Application to 401(k) Example . . . . .	298
<b>APPENDIX</b>	<b>301</b>	

# List of Figures

1.1	The only known portrait of Legendre (a friendly caricature). Source: Wikipedia. The hairstyle is amazing. . . . .	6
1.2	Pythagoras of Samos invented least squares learning and analysis of variance for the case of $n = 2$ and $p \leq 2$ around 570 BC. He was therefore the first known machine learner. . . . .	10
2.1	Tozinameran (Pfizer-BioNTech Covid-19 vaccine); Image Source: Wikipedia	31
2.2	The aggregate data from the Pfizer RCT; source: FDA <a href="#">briefing</a> . . . . .	32
2.3	A Causal Diagram for RCT . . . . .	40
2.4	A Causal Diagram for the RCT Research Design . . . . .	40
3.1	Example of regression coefficients, $\beta_j = 1/j^2$ that satisfy approximate sparsity. . . . .	55
3.2	The true coefficients (black) vs. coefficients estimated by Lasso (blue) in Example 3.1.2. . . . .	57
3.3	The true coefficients (black) vs. coefficients estimated by Post-Lasso (blue) in the Example 3.1.2. . . . .	59
3.4	The Lasso penalty is best suited for approximately sparse models, and the Ridge penalty for models with small dense coefficients. The Elastic Net can be tuned to perform well with either sparse or dense coefficients. The Lava penalty is best suited for models with coefficients generated as the sum of approximately sparse coefficients and small dense coefficients.	62
4.1	<b>Top Panel:</b> Simulated distribution of the orthogonal estimator centered around the true value. <b>Bottom Panel:</b> Simulated distribution of the naive (single-selection) non-orthogonal estimator centered around the true value. . . . .	88
5.1	Source: Franz H. Messerli, "Chocolate Consumption, Cognitive Function, and Nobel Laureates", New England Journal of Medicine. 2012 . . . . .	95
5.2	A Contrived Causal Path Diagram for the Effect of Country's Wealth on Chocolate Consumption and Nobel Prize Production per capita. . . . .	95
5.3	A Causal Diagram for the Conditional Ignorability Research Design . .	99
5.4	A Causal Diagram with Conditional Ignorability . . . . .	99
6.1	A simple causal diagram representation of the TSEM for the household gasoline demand example. . . . .	113
6.2	An expanded causal diagram representation of the TSEM that shows the unobserved shocks $\epsilon_P$ and $\epsilon_Y$ as root nodes. . . . .	114
6.3	DAG with a collider representing SEM (6.3.1). . . . .	115

6.4 Our SEM predicts that this actor, A. Terminator, is (essentially) the most talented actor in Hollywood. . . . .	116
6.5 A Simple Model of Discrimination . . . . .	118
6.6 Early 20th century: The work of Sewall and Philip Wright made it possible for humans to begin to "fly" in the space of causal models. Another family of Wrights made it possible for humans to begin to fly in the air. . . . .	121
6.7 An early drawing for an airplane appears very much like an early drawing of a DAG. . . . .	121
6.8 DAG for Supply-Demand Systems in P. Wright's work in 1928 [1]. . . . .	121
7.1 Regression tree based on wage data. The bottom nodes on the tree provide prediction rules for different subsets of observations. For example, the predicted hourly wage for a college educated worker with 9.5 or more years of experience (a worker with college = 1 and exper $\geq$ 9.5) is 24 dollars. . . . .	129
7.2 Depth 1 tree in the wage example . . . . .	130
7.3 Depth 2 tree in the wage example . . . . .	130
7.5 "To prune a tree". Source: Wikipedia . . . . .	131
7.4 Depth 3 tree in the wage example. The depth of three was chosen to avoid getting headaches from looking at a more complicated tree. . . . .	131
7.6 Approximation of $g(Z) = \exp(4Z)$ by a shallow Regression Tree in the noiseless case. . . . .	132
7.7 Approximation of $g(Z) = \exp(4Z)$ by a deep Regression Tree in the noiseless case. . . . .	132
7.8 Approximation of $g(Z) = \exp(4Z)$ by a Random Forest in the noiseless case. . . . .	132
7.9 Approximation of $g(Z) = \exp(4Z)$ by Boosted Trees in the noiseless case with a sufficient number of steps $J$ . . . . .	133
7.10 The sigmoid (logit) and smoothed ReLU activation functions . . . . .	135
7.11 Standard Architecture of a Deep Neural Network as depicted in Nielsen [6]. The input is mapped nonlinearly into the first hidden layer of the neurons. The output of this first mapping is then mapped nonlinearly into the second layer. This process is then repeated $m$ times. The output of the penultimate layer is finally mapped (linearly or nonlinearly) into the output layer, which can have multiple outputs corresponding to different tasks. . . . .	139
7.12 Approximation of $g(Z) = \exp(4Z)$ by a Neural Network . . . . .	140

7.13 The structure of the predictive model in Bajari et al. (2021) [17]. The input consists of images and unstructured text data. The first step of the process creates the moderately high-dimensional numerical embeddings $I$ and $W$ for images and text data via state-of-the art deep learning methods, such as ResNet50 and BERT. The second step of the process takes as input $X = (I, W)$ and creates predictions for hedonic prices $H_t(X)$ using deep learning methods with a multi-task structure. The models of the first step are trained on tasks unrelated to predicting prices (e.g., image classification or word prediction), where embeddings are extracted as hidden layers of the neural networks. The models of the second step are trained by price prediction tasks. The multitask price prediction network creates an intermediate lower dimensional embedding $V = V(X)$ , called value embedding and then predicts the final prices in all time periods $\{H_t(V), t = 1, \dots, T\}$ . Some variations of the method include fine-tuning the embeddings produced by the first step to perform well for price prediction tasks (i.e. optimizing the embedding parameters so as to minimize price prediction loss). . . . .	151
8.1 The causal DAG equivalent to the TSEM in Example 8.2.1. . . . .	160
8.2 The causal DAG corresponding to the TSEM in Example 8.2.1 with latent root nodes erased. . . . .	160
8.3 The graph produced from Figure 8.1 by conditioning on $X = x$ . Here $X$ is a parent to both $P$ and $Y$ . After conditioning, the remaining source of variation in $P(x)$ is $\epsilon_P$ . $\epsilon_P$ is determined exogenously – as if by an experiment – which allows measurement of the causal effect $P(x) \rightarrow Y$ . . . . .	161
8.4 Causal DAG describing the counterfactual SEM induced by doing $P = p$ . . . . .	163
8.5 Causal DAG describing the counterfactual SEM induced by setting $P = p$ in the $Y$ equation in (8.2.1) (formally a SWIG). . . . .	163
8.6 LS-DAG Example . . . . .	165
8.7 The path $Y \leftarrow X \rightarrow D$ is blocked by conditioning on $X$ . . . . .	168
8.8 The path $Y \rightarrow C \leftarrow D$ is blocked, but becomes open by conditioning on $C$ . . . . .	168
8.9 Example of d-separation. . . . .	169
8.10 Example of d-separation. . . . .	169
8.11 CF LS-DAG induced by $do(D = d)$ intervention. . . . .	170
8.12 CF LS-DAG (SWIG) induced by the $fix_Y(D = d)$ intervention. . . . .	171
8.13 CF LS-DAG induced by $fix(D = d)$ intervention. . . . .	173
8.14 A DAG in Pearl's Example . . . . .	175
8.15 The DAG induced by the Fix/Swig intervention $fix(D = d)$ in Pearl's Example. . . . .	175
8.16 Reduced DAG for Pearl's Example . . . . .	178
8.17 The equivalence class for DAGs in the TSEM (Example 8.2.1). The undirected edges mean that they can be directed in any direction as long as this does not create a cycle. In empirical analysis directionality must therefore be deduced and assumed from the context. . . . .	179

8.18 The Equivalence Class for the DAG in Pearl's Example (Example 8.5.2). Only two edges can be reoriented here. . . . .	179
8.19 Uhler et. al [11]: A set of "unfaithful" distributions $\rho$ in the simple triangular Gaussian SEM/DAG: $X_1 \rightarrow X_2, (X_1, X_2) \rightarrow X_3$ . The set is parameterized in terms of the covariance of $(X_1, X_2, X_3)$ . The right panel shows the set, and the three panels shows 3 of 6 components of the set. Each of the cases corresponds to the non-generic case which would make faithfulness fail, leading to discovery of the wrong DAG structure. In finite samples, we are not able distinguish models that are close to the set of unfaithful distributions from unfaithful distributions and may also discover the wrong DAG structure. . . . .	182
8.20 Pearl's Example . . . . .	183
8.21 Illustration of PC Algorithm due to Glymour et al . . . . .	186
8.22 Illustration of FCI Algorithm due to [14]. . . . .	187
9.1 Left: Behavior of a conventional (non-orthogonal) ML estimator. Right: Behavior of the orthogonal, DML estimator. . . . .	198
9.2 Left: DML distribution without sample-splitting. Right: DML distribution with cross-fitting. . . . .	200
9.3 Witchcraft tables used by some ML hackers to tune parameters. There are no known theoretical guarantees attached to this tuning method. . . . .	200
9.4 A Possible DAG Structure for the Gun Ownership Example. Here we approximate the average causal effect $G_{j,t} \rightarrow Y_{jt}$ only if $G_{j,t} \approx D_{j,t}$ . Under additive error of $D_{j,t}$ , the target parameter $\beta$ will be attenuated relative to the true causal effect; see Section 9.5. . . . .	201
9.5 Three Causal DAGs for analysis of the 401(K) example in Which adjusting for $X$ is a valid identification strategy. The bottom figure encompasses the other two as special cases. . . . .	207
9.6 A DAG Structure where adjusting for $X$ is not sufficient. If there is no arrow from $F$ to $M$ , adjusting for $X$ is sufficient. . . . .	208
9.7 Another DAG Structure where adjusting for $X$ is not sufficient. Here the latent confounder $U$ affects all variables, so even in the absence of an arrow connecting $F$ to $M$ , causal effects cannot be determined after adjusting for $X$ . The presence of such latent confounders is always a threat to causal interpretability of any observational study. . . . .	208
10.1 Featurizing the most talented man: The original 3072-dimensional image $W$ and image $\hat{W}$ produced from 256-dimensional embedding. (As a by-product, we've just made an important causal discovery that, surprisingly, doing embedding causes one to be younger). . . . .	229
10.2 The left panel shows a linear single layer autoencoder, such as linear principal components. The right panel shows a three layer nonlinear autoencoder; the middle layers can be used as embeddings. . . . .	230

10.3 Examples of words converted to numerical features via Word2Vec. Compare embeddings for words "shirt" and "shirts" and for "luggage" and "dress". . . . .	233
10.4 ELMO Architecture. This is the ELMO network for a string of 4 words, with $L = 2$ hidden layers. Here, the softmax layer (multinomial logit) is a single function mapping each input in $\mathbb{R}^d$ to a probability distribution over the dictionary $\Sigma$ . . . . .	236
10.5 BERT Architecture . . . . .	239
10.6 The ResNet50 operates on numerical 3-dimensional arrays representing images. It first does some pre-processing by applying convolutional and pooling filters, then it applies many L-residual block mappings, producing the arrays shown in green. The penultimate layer produces a high-dimensional vector $I$ , the image embedding, which is then used to predict the image type. . . . .	241
10.7 The structure of the predictive model in Bajari et al. [5]. The input consists of images and unstructured text data. The first step of the process creates numerical embeddings $I$ and $W$ for images and text data via state of the art deep learning methods, such as ResNet50 and BERT. The second step of the process takes as its input $X = (I, W)$ and creates predictions for hedonic prices $H_t(X)$ using deep learning methods with a multi-task structure. The models of the first step are trained on tasks unrelated to predicting prices (e.g., image classification or word prediction), where embeddings are extracted as hidden layers of the neural networks. The models of the second step are trained by price prediction tasks. The multitask price prediction network creates an intermediate lower dimensional embedding $V = V(X)$ , called a value embedding, and then predicts the final prices in all time periods $\{H_t(V), t = 1, \dots, T\}$ . Some variations of the method include fine-tuning the embeddings produced by the first step to perform well for price prediction tasks (i.e. optimizing the embedding parameters so as to minimize price prediction loss). . . . .	243
11.1 $D$ causes $Y$ . . . . .	250
11.2 $D$ and $Y$ are caused by a latent factor $A$ . . . . .	250
11.3 A DAG with Latent Confounder $A$ and Instrument $Z$ . . . . .	250
11.4 A DAG with two proxies for latent confounders. . . . .	250
11.5 $X$ are observed confounders, and $A$ are unobserved confounders. . . . .	251
11.6 Sensitivity contour plots: The graph shows values of $R_{\tilde{Y} \sim \tilde{D}   \tilde{A}}^2$ and $R_{\tilde{D} \sim \tilde{A}}^2$ that give a given value of the bias $ \hat{\phi}  = .026$ . . . . .	254
11.7 An IV model with observed and unobserved confounders. . . . .	254
11.8 DAG corresponding to Figure ?? after partialling out observed confounder $X$ . . . . .	256
11.9 A DAG for aggregate demand, with the latent node $\epsilon^d$ representing the demand shock . . . . .	256
11.10 A DAG with Controls and Proxy Controls . . . . .	257
11.11 A DAG with Proxy Controls After Partialling Out . . . . .	258
11.12 LATE models. Green arrow denotes a monotone functional relation. . . . .	259

11.13IV Quantile Model. The green arrow represents a strictly monotonic effect.	261
11.14A SEM with Proxy Controls $Q$ and $S$ . Note that conditioning on $Q$ and $S$ does not block the backdoor path $Y \leftarrow A \rightarrow D$ , hence we cannot use the regression adjustment method for identification of $D \rightarrow Y$ .	262
12.1 DAG for the Effect of Quality of Institutions on Wealth.	276
12.2 Actual sampling distribution of the IV estimator in a simulation experiment vs the normal approximation of the IV Estimator using weak instrument.	282
12.3 Construction of weak IV robust confidence regions for the effect of institutions on output using DML. Values of the $C(\theta)$ statistic are shown on the vertical axis; values of $\theta$ tested on the horizontal axis. The 90% confidence region is given by the red vertical bars.	285
13.1 Inference on ATE of 401(k) Eligibility by Income Group	298
13.2 Inference on CATE of 401(k) Eligibility Conditional on Log-Income	299

## List of Tables

1.1 Descriptive statistics for sample of never married workers.	17
1.2 Assessment of predictive performance with sample $R^2$ and $MSE$ .	18
1.3 Assessment of predictive performance on a 20% validation sample.	19
1.4 Empirical means given gender for never-married workers.	20
1.5 Estimated gender wage gap for never married worker.	20
1.6 The estimated gender wage gap for never married workers with approximately 1000 controls generated as all possible two-way interactions of raw controls.	21
4.1 Estimates for the convergence coefficient	80
4.2 Estimates of Heterogeneous Predictive Effects in the CPS 2012 data	83
4.3 Simultaneous 90% Confidence Intervals for the Estimates of Heterogeneous Predictive Effects in the CPS 2012 data.	83
7.1 Prediction Performance for the Test/Validation Sample.	148
7.2 Weights of the ensemble method.	149
9.1 Estimated Effect of 401(k) Eligibility on Net Financial Assets	209
12.1 DML Estimates of the Effect of Institutions on Output	277
12.2 DML Estimates of LATE on 401(k) Participation on Net Financial Assets	280



# Notation

$:=$	assignment or definition
$\equiv$	equivalence
$\mapsto$	"maps to" in the definition of a function
$X, Y, Z$	random variables (includes vectors)
$X'$	transpose of a vector;
$x$	for noise vectors, we use $\epsilon, \epsilon_X, \epsilon_j, \dots$
$P$	value of a random variable $X$
$P_X$	probability measure
$X^1, \dots, X^n \stackrel{\text{iid}}{\sim} P_X$	probability distribution of $X$
$P_{Y X}$	an i.i.d. sample of size $n$ ; sample index is usually $i$
$p$	conditional law of $Y$ given $X$
$p_X$	density (either probability mass function or probability density function)
$p(x)$	density of $P_X$
$\int p(x)dx$	integral with respect to the base measure (Lebesgue for probability density and counting measure for pmf)
$p(y x)$	(conditional) density of $P_{Y X=x}$ evaluated at $y$
$E[X]$	expectation of $X$
$\text{Var}(X)$	variance of $X$
$\text{Cov}(X, Y)$	covariance of $X, Y$
$X \perp Y$	orthogonality of $X, Y$ , i.e. $E(XY') = 0$
$X \perp\!\!\!\perp Y$	independence of $X, Y$
$X \perp\!\!\!\perp Y   Z$	conditional independence of $X, Y$ given $Z$
$\mathbf{X} = (X_1, \dots, X_d)$	random vector of length $d$ ; dimension index is usually $j$
$M$	structural causal model
$P_{Y(x)} = P_{Y:do(X=x)}$	intervention distribution (can be indexed by $M$ )
$P_{Y(x) X} = P_{Y X:fix_Y(X=x)}$	counterfactual distribution
$G$	directed graph
$\text{pa}_G(X), \text{deg}(X), \text{an}_G(X)$	parents, descendants, and ancestors of node $X$ in graph $G$

# Preface

This book aims to provide a working introduction to the emerging fusion of modern statistical (machine learning) inference and causal inference methods, with the main focus given to the latter. The book is aimed at upper level undergraduates, master's-level students, and doctoral students focusing on applied empirical research. A sufficient background for the core material is one semester of introductory econometrics and one semester of machine learning. We hope the book is also useful to empirical researchers looking to apply modern methods in their work.

The book introduces ideas from classical structural equation models (SEMs) and their modern AI equivalent, directed acyclic graphs (DAGs) and structural causal models (SCMs). It provides tools for statistical inference that are based on Machine Learning and AI (LASSO, random forest and deep neural networks, among others) to infer causal parameters and quantify uncertainty.

The book has three main sections: Core Material, Advanced Core and Topics. The Core is made up of chapters which either focus on predictive inference or causal inference. The type of chapter alternates between predictive inference and causal inference— to show how tools developed for predictive inference can be used to answer causal inference questions.

Within sections, blocks marked with an asterisk require more substantial preparation in mathematical statistics. We recommend that the reader looking to apply machine learning methods in their work skim or pass them on their first reading and return to them at their leisure.

Short lists of references and study problems are included after each chapter to offer the reader opportunities to investigate further, and consolidate their knowledge.

We would like to also acknowledge the tremendous and exceptional help and expertise provided by John Walker, Andy Haupt, Suhas Vijaykumar, Jannis Kuck, Malte Kurz, and Sven Klassen, David Hughes, Sophie Sun, Vira Semenova with both writing and developing supporting Notebooks in R and Python. We are

also grateful to Alexander Quispe and Anzony Quispe for developing a Bookdown version of the notebooks and providing other complementary topics and great examples.

*Chernozhukov, Hansen, Spindler & Syrgkanis*

# **CORE MATERIAL**

# Predictive Inference with Linear Regression in Moderately High Dimensions

1

To infer: to deduce or conclude (information) from evidence; Google Search.

Least squares, and particularly its application to linear regression, is the most widely used statistical method. It is an intuitive tool for predictive inference and for establishing association. The method of least squares was introduced in the 1800s by L. Legendre and C.F. Gauss, who used it to fit structural models. Here we review properties of least squares estimation of linear models in moderately high-dimensional problems, focusing on its use in predictive inference and for establishing association. This treatment provides a starting point for our subsequent review of modern statistical (machine) learning methods, which will relax our assumption on dimensionality and consider nonlinear models.

1.1 Foundation of Linear Regression . . . . .	6
Regression and the Best Linear Prediction Problem . . . . .	6
The Best Linear Prediction Problem in Finite Samples . . .	7
Properties of Sample Linear Regression . . . . .	8
Analysis of Variance (ANOVA) . . . . .	9
Overfitting: What happens when $p/n$ is not small. . . . .	10
Measuring Predictive Ability by Sample Splitting . .	11
1.2 Inference about Predictive Effects or Association . . . . .	13
Understanding $\beta_1$ via "Partialling-Out" . . . . .	13
Adaptive Inference . . . . .	15
1.3 Application: Wage Equation . . . . .	17
Prediction of Wages . . . . .	18
Gender Wage Gap . . . . .	19
1.4 Notes . . . . .	22

## 1.1 Foundation of Linear Regression

### Regression and the Best Linear Prediction Problem

We consider a scalar random variable  $Y$ , an outcome of interest, and a  $p$ -vector of covariates

$$X = (X_1, \dots, X_p)'$$

We assume that a constant of 1 is included as the first component in  $X$ ; that is,  $X_1 = 1$ .

For theoretical purposes, we first consider linear regression in the population. Working in the population means that we have access to unlimited amounts of data to compute population moments – such as  $EY$ ,  $EYX$ , and  $EXX'$  – and that we can define "ideal" quantities. After defining these ideal quantities, we then turn to estimation with real data which we will take to be a sample of observations drawn from the population.

Our first goal is to construct the best linear prediction rule for  $Y$  using  $X$ . That is, the predicted value of  $Y$  given  $X$  will be of the linear form:

$$\sum_{j=1}^p \beta_j X_j = \beta' X, \text{ for } \beta = (\beta_1, \dots, \beta_p)',$$

where  $\beta$ 's are called the regression parameters or coefficients.

We define  $\beta$  as any solution to the **Best Linear Prediction (BLP) Problem**,

$$\min_{b \in \mathbb{R}^p} E(Y - b'X)^2,$$

where we minimize the Expected or Mean Squared Error (MSE) for predicting  $Y$  using the linear rule

$$b'X = \sum_{j=1}^p b_j X_j, \quad b = (b_1, \dots, b_p)'.$$

The solution to this optimization problem,  $\beta'X$ , is called the **Best Linear Predictor** of  $Y$  using  $X$ . This jargon refers to the fact that  $\beta'X$  is the best, according to Mean Squared Error, linear prediction rule for  $Y$  among all possible linear prediction rules.



**Figure 1.1:** The only known portrait of Legendre (a friendly caricature). Source: Wikipedia. The hairstyle is amazing.

We can compute an optimal  $\beta$  by solving the first order conditions for the BLP problem:

$$E(Y - \beta'X)X = 0.$$

These equations are also referred to as the Normal Equations and are obtained by setting the derivative of the objective function  $b \mapsto E(Y - b'X)^2$  with respect to  $b$  equal to zero. Thus, any optimal  $b = \beta$  satisfies the Normal Equations.

Defining the regression error as

$$\varepsilon := (Y - \beta'X),$$

we can write the Normal Equations as

$$E\varepsilon X = 0 \text{ or } \varepsilon \perp X.$$

Therefore, we obtain the simple decomposition of  $Y$ :

$$Y = \beta'X + \varepsilon, \quad E\varepsilon X = 0,$$

where  $\beta'X$  is the part of  $Y$  that can be linearly predicted or explained with  $X$ , and  $\varepsilon$  is the remaining unexplained or residual part.

## The Best Linear Prediction Problem in Finite Samples

In practice, the researcher does not have access to the entire population, but observes only a sample

$$(Y_i, X_i)_{i=1}^n = ((Y_1, X_1), \dots, (Y_n, X_n)).$$

We assume that this sample is a random sample from a distribution  $F$ , which is the distribution of  $(Y, X)$ . Formally, this condition means that the observations were obtained as realizations of independently and identically distributed copies of the random variable  $(Y, X)$ .

We construct the best in-sample linear prediction rule for  $Y$  using  $X$  analogously to the population case by replacing theoretical expected values,  $E$ , with empirical averages,  $\mathbb{E}_n$ . Specifically, given  $X$ , our predicted value of  $Y$  will be

$$\sum_{j=1}^p \hat{\beta}_j X_j = \hat{\beta}'X, \text{ for } \hat{\beta} = (\hat{\beta}_1, \dots, \hat{\beta}_p)',$$

$\mathbb{E}_n$  abbreviates the notation  $\frac{1}{n} \sum_{i=1}^n$ .  
For example,

$$\mathbb{E}_n f(Y_i, X_i) := \frac{1}{n} \sum_{i=1}^n f(Y_i, X_i).$$

where  $\hat{\beta}$  is any solution to the **Best Linear Prediction Problem in the Sample**:

$$\min_{b \in \mathbb{R}^p} \mathbb{E}_n (Y_i - b'X_i)^2.$$

That is,  $\hat{\beta}$  minimizes the Sample Mean Squared Error for predicting  $Y$  using the linear rule  $b'X$ . The  $\hat{\beta}$ 's are called the sample regression coefficients.

We can compute an optimal  $\hat{\beta}$  as any solution to the Sample Normal Equations,

$$\mathbb{E}_n X_i(Y_i - X'_i \hat{\beta}) = 0,$$

which are obtained as the first order conditions to the Best Linear Prediction Problem in the Sample. Further, defining the in-sample regression error as

$$\hat{\varepsilon}_i := (Y_i - \hat{\beta}'X_i),$$

we obtain the decomposition

$$Y_i = X'_i \hat{\beta} + \hat{\varepsilon}_i, \quad \mathbb{E}_n X_i \hat{\varepsilon}_i = 0,$$

where  $X'_i \hat{\beta}$  is the predicted or explained part of  $Y_i$ , and  $\hat{\varepsilon}_i$  is the unexplained/residual part.

## Properties of Sample Linear Regression

The best linear prediction rule in the population is  $\beta'X$ , and a key question is whether  $\hat{\beta}'X$  approximates (that is, estimates)  $\beta'X$  well.

The best linear prediction rule is also the best linear rule for predicting future values of  $Y$  given a new draw  $X$ , when new  $(Y, X)$  are sampled from distribution  $F$ . Therefore, if we can approximate the best linear prediction rule in the population, we can also approximate the best linear prediction rule for predicting outcomes given future  $X$ s sampled from distribution  $F$ .

The fundamental statistical issue is that we are trying to estimate  $p$  parameters,  $\beta_1, \dots, \beta_p$ , without imposing any assumptions on these parameters. Intuitively, to estimate each parameter well, we need many observations per parameter. This intuition suggests that  $n/p$  should be large, or, equivalently that  $p/n$  should be small, in order for estimation error to be small. The following result captures this intuition more formally.

**Theorem 1.1.1** (Approximation of BLP by OLS) *Under regularity conditions,<sup>a</sup>*

$$\begin{aligned}\sqrt{\text{E}_X(\beta'X - \hat{\beta}'X)^2} &= \sqrt{(\hat{\beta} - \beta)' \text{E}_X XX' (\hat{\beta} - \beta)} \\ &\leq \text{const}_F \cdot \sqrt{\text{E}\varepsilon^2} \sqrt{\frac{p}{n}},\end{aligned}$$

where  $\text{E}_X$  is the expectation with respect to  $X$ , the inequality holds with probability approaching 1 as  $n \rightarrow \infty$ , and  $\text{const}_F$  is a constant that depends on the distribution  $F$  of  $(Y, X)$ .

<sup>a</sup> See Notes (Section 1.4) for references.

If  $n$  is large and  $p$  is much smaller than  $n$ , for nearly all realizations of data, the sample linear regression is close to the population linear regression:

$$\sqrt{\text{E}_X(\beta'X - \hat{\beta}'X)^2} \approx 0.$$

In other words, under our requirement of  $p/n$  small, the sample BLP approximates the population BLP well.

Given indexed random variables (vectors, elements)  $A_n$  and  $B_n$  in a metric space equipped with metric  $d$ , the notation  $A_n \approx B_n$  means that the distance between  $A_n$  and  $B_n$  concentrates around 0 – formally, that  $\lim_{n \rightarrow \infty} \text{P}(d(A_n, B_n) \leq \varepsilon) \rightarrow 1$  for each  $\varepsilon > 0$ .

## Analysis of Variance (ANOVA)

ANOVA involves the decomposition of the variation of  $Y$  into explained and unexplained parts. Explained variation is a measure of the predictive performance of a model. ANOVA can be conducted both in the population and in the sample.

The main idea is to use the previous decomposition of  $Y$ ,

$$Y = \beta'X + \varepsilon, \quad \text{E}\varepsilon X = 0,$$

to decompose the variation in  $Y$  into the sum of *explained variation* and *residual variation*:

$$\text{E}Y^2 = \text{E}(\beta'X)^2 + \text{E}\varepsilon^2.$$

The quantity

$$\text{MSE}_{pop} = \text{E}\varepsilon^2$$

is the population Mean Squared Prediction Error (MSE). The ratio of the explained variation to the total variation is the population  $R^2$ :

$$R^2_{pop} := \frac{\text{E}(\beta'X)^2}{\text{E}Y^2} = 1 - \frac{\text{E}\varepsilon^2}{\text{E}Y^2} \in [0, 1].$$

That is,  $R_{pop}^2$  is the proportion of variation of  $Y$  explained by the BLP.

**Remark 1.1.1** The standard definition of  $R^2$  assumes that either we work with a centered  $Y$  or that we recenter  $Y$  such that  $EY = 0$ . (However, our definition above does not require this property).

ANOVA in the sample proceeds analogously. Using the representation

$$Y_i = \hat{\beta}' X_i + \hat{\varepsilon}_i$$

and the orthogonality condition  $\mathbb{E}_n X_i \hat{\varepsilon}_i = 0$  provided by the sample Normal Equations, we obtain the decomposition

$$\mathbb{E}_n Y_i^2 = \mathbb{E}_n (\hat{\beta}' X_i)^2 + \mathbb{E}_n \hat{\varepsilon}_i^2.$$

Thus, we can define the sample MSE,

$$\text{MSE}_{sample} = \mathbb{E}_n \hat{\varepsilon}_i^2,$$

and the sample  $R^2$ ,

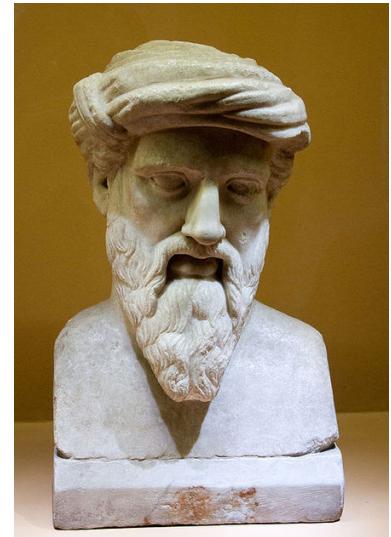
$$R_{sample}^2 := \frac{\mathbb{E}_n (\hat{\beta}' X_i)^2}{\mathbb{E}_n Y_i^2} = 1 - \frac{\mathbb{E}_n \hat{\varepsilon}_i^2}{\mathbb{E}_n Y_i^2} \in [0, 1].$$

By the law of large numbers and Theorem 1.1.1, when  $p/n$  is small, we have the following approximations:

$$\mathbb{E}_n Y_i^2 \approx EY^2, \quad \mathbb{E}_n (\hat{\beta}' X_i)^2 \approx E(\beta' X)^2, \quad \mathbb{E}_n \hat{\varepsilon}_i^2 \approx E\varepsilon^2,$$

Thus, when  $p/n$  is small and  $n$  is large, the sample fit measures are good approximations to population fit measures:

$$\text{MSE}_{sample} \approx \text{MSE}_{pop} \text{ and } R_{sample}^2 \approx R_{pop}^2.$$



**Figure 1.2:** Pythagoras of Samos invented least squares learning and analysis of variance for the case of  $n = 2$  and  $p \leq 2$  around 570 BC. He was therefore the first known machine learner.

## Overfitting: What happens when $p/n$ is not small.

When  $p/n$  is not small, the picture about predictive performance of the in-sample BLP becomes inaccurate and possibly misleading. In this setting, the in-sample linear predictor can be substantially different from the population BLP.

Consider an extreme example where  $p = n$  and all variables in  $X$  are linearly independent. In this case, we have

$$\text{MSE}_{\text{sample}} = 0 \text{ and } R_{\text{sample}}^2 = 1$$

no matter what  $\text{MSE}_{\text{pop}}$  and  $R_{\text{pop}}^2$  are. Therefore, here we have an extreme example of **overfitting**, where the in-sample predictive performance overstates the out-of-sample predictive performance of the linear model. The following example illustrates less extreme cases.

**Example 1.1.1** (Overfitting Example) Suppose  $X \sim N(0, I_p)$  and  $Y \sim N(0, 1)$  are statistically independent. It follows that the best linear predictor of  $Y$  is  $\beta'X = 0$  and the true  $R_{\text{pop}}^2$  is 0.

- ▶ If  $p = n$ , then the typical  $R_{\text{sample}}^2$  is  $1 \gg 0$ .
- ▶ If  $p = n/2$ , then the typical  $R_{\text{sample}}^2$  is about  $.5 \gg 0$ .
- ▶ If  $p = n/20$ , then the typical  $R_{\text{sample}}^2$  is about  $.05$ .

These results can be deduced by simulation or analytically.

The Linear Model Overfitting R Notebook contains the numerical experiment.

Better measures of out-of-sample predictive ability are the "adjusted"  $R^2$  and  $\text{MSE}$ :<sup>1</sup>

$$\text{MSE}_{\text{adjusted}} = \frac{n}{n-p} \mathbb{E}_n \hat{\varepsilon}_i^2, \quad R_{\text{adjusted}}^2 := 1 - \frac{n}{n-p} \frac{\mathbb{E}_n \hat{\varepsilon}_i^2}{\mathbb{E}_n Y_i^2}.$$

The adjustment by  $\frac{n}{n-p}$  corrects for overfitting and provides a more accurate assessment of predictive ability of the linear model in Example 1.1.1 and more generally under the assumption of homogeneous  $\varepsilon$ . The intuition is that models with many parameters increase the in-sample fit and potentially cause overfitting. Hence, the number of parameters is incorporated in the definition of  $\text{MSE}_{\text{adjusted}}$  and  $R_{\text{adjusted}}^2$  in an attempt to account for this phenomenon.

1: The adjustment factor  $\frac{n}{n-p}$  is derived in a homogeneous model, so that  $E[\text{MSE}_{\text{adjusted}}] = \text{MSE}_{\text{pop}}$ , see e.g. p. 8 in [1] for the derivation.

## Measuring Predictive Ability by Sample Splitting

How should we measure the predictive ability of the linear model (or other nonlinear models that we discuss) more reliably, even in cases when  $p/n$  is not small?

A general way to measure predictive performance is to perform **data splitting**. The idea can be summarized in two parts:

1. Use a random part of a dataset, called the training sample, for estimating/training the prediction rule.
2. Use the other part, called the testing sample, to evaluate the quality of the prediction rule, recording out-of-sample mean squared error and  $R^2$ .

Generally, a predictive model is trained on a sample and the real test of its predictive ability happens when “new, unseen” observations arrive. With new observations in hand, we learn how far off our predictions were then compared to the realized values. By partitioning the data set into two parts, we preserve an “unseen” set of observations on which to test our model, mimicking this process of ex-post performance assessment.<sup>2</sup>

The data splitting procedure can be described more formally as follows:

### Generic Evaluation of Prediction Rules by Sample-Splitting

1. Randomly partition the data into training and testing samples. Suppose we use  $n$  observations for training and  $m$  for testing/validation.
2. Use the training sample to compute a prediction rule  $\hat{f}(X)$ . For example,  $\hat{f}(X) = \hat{\beta}'X$  in the linear model.
3. Let  $V$  denote the indexes of the observations in the test sample. Then the out-of-sample/test mean squared error is

$$\text{MSE}_{test} = \frac{1}{m} \sum_{k \in V} (Y_k - \hat{f}(X_k))^2,$$

and the out-of-sample/test  $R^2$  is

$$R^2_{test} = 1 - \frac{\text{MSE}_{test}}{\frac{1}{m} \sum_{k \in V} Y_k^2}.$$

There is an important variation on the sample splitting procedure, called **stratified splitting** that provides guarantees that the training and test samples are similar.<sup>3</sup> In large samples, training and test samples will be similar by virtue of the laws of large numbers, but this is not guaranteed in moderate-sized samples. For more discussion, please see this blog on [Data Splitting \[2\]](#).<sup>4</sup>

2: If the "test set" is used many times to evaluate models, it becomes a "validation" set. The term "test set" is often reserved for the final evaluations of very few models.

3: For example, we can make sure that the proportions of men and women, or of college-educated and non-college-educated workers are the same in both training and test samples. These issues are important in moderate-sized samples.

4: The caret package in R provides this functionality via the “create-DataPartition” function.

## 1.2 Inference about Predictive Effects or Association

Here we examine inference on *predictive effects*, which describe how our (population best linear) predictions change if a value of a target regressor changes by a unit while the other regressors remain unchanged.

Specifically, we partition the vector of regressors  $X$  into two components:

$$X = (D, W')',$$

where  $D$  represents the "target" regressor of interest, and  $W$  represents the other regressors, sometimes called the controls. We can therefore write

$$Y = \underbrace{\beta_1 D + \beta_2' W}_{\text{predicted value}} + \varepsilon, \quad (1.2.1)$$

and ask the question:

How does the predicted value of  $Y$  change if  $D$  increases by a unit while  $W$  remains unchanged?

The answer is the predicted value of  $Y$  changes by

$$\beta_1.$$

Note that this question is purely about the properties of the predictive rule, and generally has nothing to do with causality.

**Example 1.2.1** (Gender Pay Gap) In the analysis of wages, which we will discuss later in more detail, the question can be formulated as:

- ▶ "What is the difference in predicted wages between men and women with the same job-relevant characteristics?"

Letting  $D$  represent the female indicator and  $W$  represent experience, educational, occupational, and geographic characteristics, the answer to the question is the population regression coefficient

$$\beta_1$$

corresponding to  $D$ .

### Understanding $\beta_1$ via "Partialling-Out"

"Partialling-out" is an important tool that provides conceptual understanding of the regression coefficient  $\beta_1$ .

In the *population*, we define the partialling-out operation as a procedure that takes a random variable  $V$  and creates a "residual"  $\tilde{V}$  by subtracting the part of  $V$  that is linearly predicted by  $W$ :

$$\tilde{V} = V - \gamma'_{VW} W, \quad \gamma_{VW} \in \arg \min_{\gamma} E(V - \gamma' W)^2.$$

When  $V$  is a vector, we apply the operation to each component. It can be shown that the partialling-out operation is linear in the sense that

$$Y = V + U \implies \tilde{Y} = \tilde{V} + \tilde{U}.$$

Formally, this operation is well defined on the space of random variables with finite second moments.

We apply the partialling-out operation to both sides of our regression equation  $Y = \beta_1 D + \beta'_2 W + \varepsilon$  to get

$$\tilde{Y} = \beta_1 \tilde{D} + \beta'_2 \tilde{W} + \tilde{\varepsilon},$$

which simplifies to the decomposition:

$$\tilde{Y} = \beta_1 \tilde{D} + \varepsilon, \quad E\varepsilon \tilde{D} = 0. \quad (1.2.2)$$

Decomposition (1.2.2) follows because partialling-out eliminates  $\beta'_2 W$ , since  $\tilde{W} = 0$ , and leaves  $\varepsilon$  untouched,  $\tilde{\varepsilon} = \varepsilon$ , since  $\varepsilon$  is linearly unpredictable by  $X$  and therefore by  $W$ . Moreover,  $E\varepsilon \tilde{D} = 0$  since  $\tilde{D}$  is a linear function of  $X = (D, W)'$  and  $\varepsilon$  is orthogonal to  $X$  and therefore to any linear function of  $X$ .

The decomposition (1.2.2) implies that  $E\varepsilon \tilde{D} = 0$  are the Normal Equations for the population regression of  $\tilde{Y}$  on  $\tilde{D}$ . Therefore, we just rediscovered the following result.

**Theorem 1.2.1** (Frisch-Waugh-Lovell, FWL) *The population linear regression coefficient  $\beta_1$  can be recovered from the population linear regression of  $\tilde{Y}$  on  $\tilde{D}$ :*

$$\beta_1 = \arg \min_{b_1} E(\tilde{Y} - b_1 \tilde{D})^2 = (E \tilde{D}^2)^{-1} E \tilde{D} \tilde{Y},$$

where we assume  $D$  cannot be perfectly predicted by  $W$ , i.e.  $E \tilde{D}^2 > 0$ , so  $\beta_1$  is uniquely defined.

In other words,  $\beta_1$  can be interpreted as a (univariate) linear regression coefficient in the linear regression of *residualized*  $Y$  on *residualized*  $D$ , where the residuals are defined by partialling-out the linear effect of  $W$  from  $Y$  and  $D$ .

When we work with the *sample*, we simply mimic the partialling-out operation in the population in the sample. In what follows, we assume  $p/n$  is small, so sample linear regression provides high-quality partialling-out. By the FWL Theorem applied to the sample instead of population, the sample linear regression of  $Y$  on  $D$  and  $W$  gives us the estimator  $\hat{\beta}_1$  which is identical to the estimator obtained via sample partialling-out.

It is useful to give the formula for  $\hat{\beta}_1$  in terms of sample partialling-out:

$$\hat{\beta}_1 = \arg \min_b \mathbb{E}_n (\check{Y}_i - b \check{D}_i)^2 = (\mathbb{E}_n \check{D}_i^2)^{-1} \mathbb{E}_n \check{D}_i \check{Y}_i, \quad (1.2.3)$$

where the notation  $\check{V}_i$  denotes the residual left after predicting  $V_i$  with controls  $W_i$  in the sample and we assume  $\mathbb{E}_n \check{D}_i^2 > 0$ . That is,

$$\check{V}_i = V_i - \hat{\gamma}'_{VW} W_i, \quad \hat{\gamma}_{VW} \in \arg \min_{\gamma} \mathbb{E}_n (V_i - \gamma' W_i)^2.$$

From Theorem 1.1.1, we know that using sample linear regression for partialling-out will provide high-quality estimates of the residuals when  $p/n$  is small. When  $p/n$  is not small, using sample linear regression for partialling-out won't be such a good idea and an alternative is to use penalized regression or dimension reduction. We will cover this in Chapter 3, but we can definitely try it out in the empirical example that concludes this chapter before we even attempt to understand it.

Why not?

## Adaptive Inference

We next consider the large sample properties of the estimator  $\hat{\beta}_1$ .

**Theorem 1.2.2** (Adaptive Inference) *Under regularity conditions and if  $p/n \approx 0$ , the estimation error in  $\check{D}_i$  and  $\check{Y}_i$  has no first order effect on the stochastic behavior of  $\hat{\beta}_1$ , namely*

$$\sqrt{n}(\hat{\beta}_1 - \beta_1) \approx \sqrt{n} \mathbb{E}_n \tilde{D} \varepsilon / \mathbb{E}_n \tilde{D}^2 \quad (1.2.4)$$

and consequently,

$$\sqrt{n}(\hat{\beta}_1 - \beta_1) \stackrel{a}{\sim} N(0, V)$$

where

$$V = (\mathbb{E} \tilde{D}^2)^{-1} \mathbb{E}(\tilde{D}^2 \varepsilon^2) (\mathbb{E} \tilde{D}^2)^{-1}.$$

The notation  $A_n \stackrel{a}{\sim} N(0, V)$  reads as  $A_n$  is approximately distributed as  $N(0, V)$ , and formally means that  $\sup_{R \in \mathcal{R}} |\mathbb{P}(A_n \in R) - \mathbb{P}(N(0, V) \in R)| \approx 0$ , where  $\mathcal{R}$  is the collection of rectangular sets (intervals for the case of  $A_n$  being a scalar random variable).

We can equivalently write

$$\hat{\beta}_1 \stackrel{a}{\sim} N(\beta_1, V/n).$$

That is,  $\hat{\beta}_1$  is approximately normally distributed with mean  $\beta_1$  and variance  $V/n$ . Thus,  $\hat{\beta}_1$  concentrates in a  $\sqrt{V/n}$ -neighborhood of  $\beta_1$  with deviations controlled by the normal law.

The first result in Theorem 1.2.2, (1.2.4), states the estimator minus the estimand is an approximate centered average. The remaining properties stated in the theorem then follow from the central limit theorem.

The *adaptivity* refers to the fact that estimation of residuals  $\check{D}$  has a negligible impact on the large sample behavior of the OLS estimator – the approximate behavior is the same as if we had used true residuals  $\tilde{D}$  instead. This adaptivity property will be derived later as a consequence of a more general phenomenon which we shall call *Neyman orthogonality*.

The estimated standard error of  $\hat{\beta}_1$  is  $\sqrt{\hat{V}/n}$ , where  $\hat{V}$  is any estimator of  $V$  based on the plug-in principle, such that  $\hat{V} \approx V$ :<sup>5</sup>

$$\hat{V} = (\mathbb{E}_n \check{D}^2)^{-1} \mathbb{E}_n (\check{D}^2 \hat{\varepsilon}^2) (\mathbb{E}_n \check{D}^2)^{-1}.$$

Consider the set, called the *confidence interval*,

$$[l, u] := [\hat{\beta}_1 - z_{1-\alpha/2} \sqrt{\hat{V}/n}, \hat{\beta}_1 + z_{1-\alpha/2} \sqrt{\hat{V}/n}],$$

where  $z_{1-\alpha/2}$  denotes the  $(1 - \alpha/2)$ -quantile of the standard normal distribution. For example, the 95% confidence interval is given by

$$[\hat{\beta}_1 - 1.96 \sqrt{\hat{V}/n}, \hat{\beta}_1 + 1.96 \sqrt{\hat{V}/n}].$$

This set covers the population coefficient  $\beta_1$  for most realizations of the sample. More precisely, a  $(1 - \alpha)\%$  confidence interval contains  $\beta_1$  in approximately  $(1 - \alpha)\%$  of the realizations of the sample:

$$P(\beta_1 \in [l, u]) \approx 1 - \alpha.$$

In other words, if our sample is not atypical in the sense of occurring with probability smaller than  $\alpha$ , the confidence interval contains the population value of the best linear predictor coefficient  $\beta_1$ .

5: The standard estimator for independent data is called the Eicker-Huber-White robust variance estimator.

A common alternative is to use  $\tilde{V} = \mathbb{E}_n \varepsilon^2$  instead of  $\hat{V}$  and the  $(1 - \alpha/2)$ -quantile of Student's T-distribution with  $n - p$  degrees of freedom instead of  $z_{1-\alpha/2}$ . The reason is that if  $\varepsilon \perp\!\!\!\perp X$  and  $\varepsilon$  is normal then the below approximate equality to  $1 - \alpha$  becomes an exact equality. The *robust* variance estimator  $\hat{V}$  is preferred because it ensures the approximate equality to  $1 - \alpha$  regardless. Moreover, any difference between the quantiles of Student's T-distribution and the normal distribution dissipates as the number of data grows. Sometimes we still use quantiles of Student's T-distribution because they are only bigger and therefore give more conservative confidence intervals.

### 1.3 Application: Wage Equation

In labor economics an important question is what determines the wage of workers. Interest in this question goes back to the work of Jacob Mincer (see [3]). While determining the inputs that lead to a worker's wage is a causal question, we can begin to investigate it from a predictive perspective. We aim to answer two main questions:

- ▶ The Prediction Question: How can we use job-relevant characteristics, such as education and experience, to best predict wages?
- ▶ The Predictive Effect or Association Question: What is the difference in predicted wages between men and women with the same job-relevant characteristics?

We illustrate using data from the 2015 March Supplement of the U.S. Current Population Survey. As outcome,  $Y$ , we use the log hourly wage, and we let  $X$  denote various characteristics of workers. We focus on the (sub) sample of single (never married) workers, which is of size  $n \approx 5,000$  and has the following mean characteristics.

	Sample mean
Log Wage	2.97
Female	0.44
Some High School	0.02
High School Graduate	0.24
Some College	0.28
College Graduate	0.32
Advanced Degree	0.14
Experience	13.76

**Table 1.1:** Descriptive statistics for sample of never married workers.

We will estimate a linear predictive (regression) model for log hourly wage using job-relevant characteristics

$$Y = \beta'X + \varepsilon, \quad \varepsilon \perp X,$$

assess the quality of the empirical prediction rule  $\hat{\beta}'X$  using out-of-sample prediction performance, and analyze if there is a difference in pay for men and women (*gender wage gap*). The gender wage gap may partly reflect discrimination against women in the labor market (we will discuss this question in more detail in Chapter 6).

## Prediction of Wages

Our goal here is to predict (log) wages using various characteristics of workers, and assess the predictive performance of two linear models using adjusted MSE and  $R^2$ , and out-of-sample MSE and  $R^2$ .

We employ two different specifications for prediction:

- ▶ In the **Basic Model**  $X$  consists of a set of raw regressors (e.g. gender, experience, education indicators, occupation and industry indicators, and regional indicators), for a total of  $p = 51$  regressors. Our basic specification is known in labor economics as the Mincer equation, and is derived from a theoretical economic analysis; see, e.g., [3] for a review.
- ▶ In the **Flexible Model**,  $X$  consists of all raw regressors from the basic model as well as *technical regressors*, which are transformations of the raw regressors, namely, polynomials in experience  $\exp^2$  and  $\exp^3$  and additional two-way interactions of polynomials in experience with all other raw regressors. An example of a regressor created through a two-way interaction is *experience times the indicator of having a college degree*. In total we have  $p = 246$  technical regressors.

	$p$	$R^2_{sample}$	$MSE_{sample}$	$R^2_{adj}$	$MSE_{adj}$
basic reg	51	0.31	0.22	0.30	0.23
flexible reg	246	0.35	0.21	0.32	0.22

[Predicting Wages R Notebook](#) contains the predictive exercise for wages.

Table 1.2 shows measures of predictive performance. The flexible regression model performs slightly better than the basic model (higher  $R^2_{adj}$  and lower  $MSE_{adj}$ ). Note also that the discrepancy between the unadjusted and adjusted measures is not large – this is expected given that

$$p/n \text{ is small.}$$

We report the results for evaluation of prediction rules from the data splitting procedure in Table 1.3. We report results based on randomly selecting 20% of the data to serve as the testing sample and using the remaining 80% of the observations as the training sample.

**Table 1.2:** Assessment of predictive performance with sample  $R^2$  and  $MSE$ .

	$MSE_{test}$	$R^2_{test}$
basic regression	0.197	0.328
flexible regression	0.206	0.296
lasso regression	0.213	0.275

**Table 1.3:** Assessment of predictive performance on a 20% validation sample.

Based on this exercise, it now appears that the basic regression model works slightly better than the flexible regression at predicting log wages for new observations. That is, we see that the test (out-of-sample)  $MSE$  and  $R^2$  for the basic regression model are respectively slightly lower and higher than those of the flexible regression model, indicating slightly superior out-of-sample predictive performance. This behavior is different from that obtained when looking at the within sample fit statistics reported in Table 1.2.

Table 1.3 also provides the test  $MSE$  of the flexible model that has been estimated via lasso regression. Lasso (*least absolute shrinkage and selection operator*) is a penalized regression method that can be used to reduce the complexity of a regression model when the ratio  $p/n$  is not small. We introduce this method in Chapter 3, but this does not prevent us from trying it here even though it may appear as a black box at this point. The out-of-sample  $MSE$  can be computed for any other black-box prediction method as well.

## Gender Wage Gap

An important question is whether there is a difference in (predicted) wages between men and women with the same job-relevant characteristics. To answer this question, we estimate the log-linear regression model:

$$Y = \beta_1 D + \beta'_2 W + \varepsilon, \quad (1.3.1)$$

where  $Y$  is log-wage,  $D$  is the indicator of being female (1 if female and 0 otherwise) and the  $W$ 's are other determinants of wages:  $W$  includes education, polynomials in experience, region, and occupation and industry indicators plus all two-way interactions of polynomial in experience with region, occupation, and industry indicators.

As we have log-transformed wages, we are analyzing the relative difference in pay for men and women. Table 1.4 tabulates mean characteristics given gender. It shows that the difference in average log-wage between men and women is equal to

[Gender Pay Gap R Notebook](#) contains the analysis of this section.

	All	Men	Women
Log Wage	2.9708	2.9878	2.9495
Less than High School	0.0233	0.0318	0.0127
High School Graduate	0.2439	0.2943	0.1809
Some College	0.2781	0.2733	0.2840
College Graduate	0.3177	0.2940	0.3473
Advanced Degree	0.1371	0.1066	0.1752
Experience	13.7606	13.7840	13.7313

**Table 1.4:** Empirical means given gender for never-married workers.

0.038. Thus, the unconditional gender wage gap is about 3.8<sup>6</sup> for the group of never married workers (women get paid less on average in our sample). We also observe that never married working women are relatively more educated than never married working men.

Table 1.5 summarizes the regression results. Overall, we see that the unconditional wage gap of size 3.8% for women increases to about 7% after controlling for worker characteristics. This means we would predict a woman's wage to be about 7% less per hour on average than the wage of a man who had the same experience, education, geographical region and occupation.

The partialling-out approach provides a numerically identical estimate for the coefficient  $\beta_1$  ( $\beta_1 \approx 7\%$ ), numerically confirming the FWL theorem. Using lasso for partialling-out (*partial reg via lasso*) gives similar results to using OLS. This similarity is expected here, since

$$p/n \text{ is small,}$$

and the partialling out by least squares will work just about as well as partialling out by lasso.

6: This interpretation relies on the approximation  $\log(a) - \log(b) \approx (a - b)/b$ , which is accurate whenever  $(a - b)/b$  is small and  $b > 0$ .

	Estimate	Std. Error
reg without controls	-0.038	0.0159
reg with controls	-0.070	0.0150
partial out reg w/ controls	-0.070	0.0150
double lasso (p-out w/ lasso)	-0.072	0.0154

**Table 1.5:** Estimated gender wage gap for never married worker.

Finally, we note the similarity of the standard errors reported for the last three rows of the table which include controls. This is not a coincidence and is in line with the adaptivity property.

To sum up, our estimate of the conditional gender hourly wage gap for never-married workers using OLS is about  $-7\%$  and the 95% confidence interval is about  $[-10\%, -4\%]$ .

In order to wrap up and highlight some finer points about the impact of dimensionality  $p$  on inference, we try out an extra-flexible model by generating controls  $W$  as all of the possible two-way interactions of raw controls. This gives us about  $p \approx 1000$  technical controls, and  $p/n$  is no longer small, at about 20%.

	Estimate	Std. Error
full regression	-0.061	0.0169
double lasso (p-out w lasso)	-0.072	0.0153

**Table 1.6:** The estimated gender wage gap for never married workers with approximately 1000 controls generated as all possible two-way interactions of raw controls.

In this case, where  $p/n$  is no longer small, we start seeing the differences between unregularized partialling out (full regression) and regularized partialling out with lasso (aka double lasso). The results based on double lasso have rigorous guarantees in this non-small  $p/n$  regime under approximate sparsity conditions, which we will define formally later in the course.

OLS is no longer adaptive in the “ $p/n$  not small” regime, and we need to account for this in the reported standard errors ([4]). Under assumptions laid out in [4], the correct standard error scales like  $\sqrt{n/(n-p)}$  in the regime  $p/n < 1$ , showing exactly how the properties of the OLS coefficient estimate will deteriorate as  $p/n$  increases towards 1.

## Notebooks

- ▶ **Predicting Wages R Notebook** contains a simple predictive exercise for wages. We will return to this dataset and prediction problem repeatedly in future chapters, re-estimating it using a broad range of ML estimators and providing a means of comparing their performance. (URL: <https://www.kaggle.com/janniskueck/ols-and-lasso-for-wage-prediction>)
- ▶ **Gender Pay Gap R Notebook** contains a simple analysis of the gender pay gap. (URL: <https://www.kaggle.com/jan>

niskueck/ ols-and-lasso-for-gender-wage-gap-inference)

- The Linear Model Overfitting R Notebook contains a set of simple simulations that show how measures of fit perform in a high  $p/n$  setting. (URL: <https://www.kaggle.com/victorcherzhukov/r-notebook-linear-model-overfitting>)

## 1.4 Notes

Least squares were invented by Legendre and Gauss around 1800. Frisch, Waugh, and Lovell discovered the partialling out interpretation of the least squares coefficients in the 1930s. The asymptotic theory mentioned in the note is more recent and has been developed since early work of Huber in the 70s on  $m$ -estimators (that minimize the sum of losses objective functions) under moderately high dimensions.

For a good, concise treatment of classical least squares, see for example, Chapter 1 in Amemiya's classical graduate econometrics text [1]; Bruce Hansen's new text Econometrics [5], which is available online for free [here](#), is also a wonderful resource for this material and much beyond.

Regularity conditions under which Theorem 1.1.1 and Theorem 1.2.2 hold under  $p \rightarrow \infty$  and  $p/n \rightarrow 0$  asymptotics can be found in [6] and [4]. The results of the latter reference allow for  $p/n \rightarrow c < 1$ , which introduces an additional asymptotic variance term when  $c > 0$ ; the case with  $c = 0$  recovers Theorem 1.2.2. See also review [7] for some recent understanding of properties of least squares estimators.

## Study Questions

1. Write a notebook (R, Python, etc.) where you briefly explain the idea of sample splitting to evaluate the performance of prediction rules to a fellow student, and show how to use it on the wage data. The explanation should be clear and concise (one paragraph suffices) so that a fellow student can understand. You can take our notebooks as a starting point, but provide a bit more explanation and modify them by exploring different specifications of the models (or looking at an interesting subset of the data or even other data – for example, the data you use for your research or thesis work).
2. Write a notebook (R, Python etc), where you carry out a gender pay gap analysis, focusing on the subset of college-educated workers. The analysis should be analogous to what we've presented – explaining "partialing out", generating point estimates and standard errors – but don't hesitate to experiment and explain more. Exploring other data-sets or similar questions, e.g. race pay gap, is always welcome.
3. The half-serious link to Pythagoras was serious in its half. Consider sample linear regression with  $n = 2$  and just one regressor, so that  $Y_i = \hat{\beta}X_i + \hat{\varepsilon}_i$  for  $i = 1, 2$ , where  $\hat{\beta}$  is the ordinary least squares estimator, a scalar quantity in this case. Let  $Y = (Y_1, Y_2)'$ ,  $X = (X_1, X_2)'$ ,  $\hat{\varepsilon} = (\hat{\varepsilon}_1, \hat{\varepsilon}_2)'$ , and let  $\hat{Y} = \hat{\beta}X$ . Find the connection between ANOVA decomposition  $Y'Y/n = \hat{Y}'\hat{Y}/n + \hat{\varepsilon}'\hat{\varepsilon}/n$  and the Pythagorean theorem. Find the geometric interpretation for  $\hat{Y}$ , and write the explicit formula for  $\hat{\beta}$  in this case. If you get stuck, google the "geometric interpretation of least squares".

Modern notebooks, including Jupyter Books and R-studio, offer a simple way to integrate code cells and explanations (text and formulas) in a single notebook. This allows the user to execute code in discretized chunks for clarity and ease of debugging as well as to better provide commentary on what the code is doing. See the Notebooks section above for examples in Kaggle. In completing study questions, you are welcome to use R-studio and R markdown reports, if you are more used to this way of working.

# Bibliography

- [1] Takeshi Amemiya. *Advanced Econometrics*. Cambridge, MA: Harvard University Press, 1985 (cited on pages 11, 22).
- [2] *Data Splitting | R-bloggers*. <https://www.r-bloggers.com/2016/08/data-splitting/>, Accessed: 2022-25-02 (cited on page 12).
- [3] Thomas Lemieux. ‘The “Mincer equation” thirty years after schooling, experience, and earnings’. In: *Jacob Mincer a pioneer of modern labor economics*. Springer, 2006, pp. 127–145 (cited on pages 17, 18).
- [4] Matias D. Cattaneo, Michael Jansson, and Whitney K. Newey. ‘Inference in linear regression models with many covariates and heteroscedasticity’. In: *Journal of the American Statistical Association* 113.523 (2018), pp. 1350–1361 (cited on pages 21, 22).
- [5] Bruce E. Hansen. *Econometrics*. Princeton University Press, 2022 (cited on page 22).
- [6] Alexandre Belloni, Victor Chernozhukov, Denis Chetverikov, and Kengo Kato. ‘Some New Asymptotic Theory for Least Squares Series: Pointwise and Uniform Results’. In: *Journal of Econometrics* 186.2 (2015), pp. 345–366 (cited on page 22).
- [7] Arun K. Kuchibhotla, Lawrence D. Brown, and Andreas Buja. ‘Model-free study of ordinary least squares linear regression’. In: *arXiv preprint arXiv:1809.10538* (2018) (cited on page 22).

# Causal Inference via Randomized Experiments

2

In this block we begin discussion of causal inference by focusing on Randomized Control Trials (RCTs). In a randomized control trial, units are randomly divided into those that receive a treatment and those that receive no treatment. Under randomization and other assumptions, the difference in average outcomes between the treated and untreated groups is an average treatment (causal) effect (ATE). By considering pre-treatment covariates, we can improve the precision of the ATE estimate, explore heterogeneity across subgroups, or both. We describe methods for doing so and apply them to several RCTs. We introduce causal diagrams as a means of visualizing RCTs and their underlying causal assumptions. We conclude by outlining the limitations of RCTs.

2.1 Potential Outcomes Framework and Average Treatment Effects . . . . .	26
Random Assignment/ Randomized Control Trials . . . . .	29
Statistical Inference with two sample means . . . . .	30
Pfizer/BioNTech Covid Vaccine RCT . . . . .	31
2.2 Pre-treatment Covariates and Heterogeneity . . . . .	33
Regression and Statistical Inference for ATEs . . . . .	35
The Role of Covariates: Improve Precision of Estimating ATE . . . . .	36
The Role of Covariates: Discover Heterogeneity through CATE . . . . .	37
Reemployment Bonus RCT . . . . .	38
2.3 Drawing RCTs via Causal Diagrams . . . . .	39
2.4 The limitations of RCTs . . . . .	40
Externalities, Stability, and Equilibrium Effects . . . . .	41
Ethical, Practical and Generalizability Concerns . . . . .	41
2.A Approximate Distribution of the Two Sample Means . . . . .	44
2.B Approximate Distribution for Intercept and Slope Estimators . . . . .	45

## 2.1 Potential Outcomes Framework and Average Treatment Effects

In this section, we discuss the potential outcomes framework for analyzing causality and treatment effects. It offers a simple way to formalize causality as a mathematical concept.

We begin by introducing the two *latent* (unobserved) variables

$$Y(1) \text{ and } Y(0).$$

They represent the potential or counterfactual random outcomes for an observational unit when the unit is subject to treatment (treatment state  $d = 1$ ) or no treatment (control or untreated state  $d = 0$ ) in an idealized experiment [1]. In an economic context, the treatment might be a training program or a policy intervention. In what follows, it is also useful to introduce the potential response or structural function:

$$d \mapsto Y(d),$$

which maps the potential treatment state  $d \in \{0, 1\}$  to the random potential outcome  $Y(d)$ .

In this formulation we have dependence of the potential outcome  $Y(d)(\omega)$  on the underlying state of the world  $\omega$ . In our formalization,  $\omega$  will represent randomness across observational units and from any other sources.<sup>1</sup>

The quantities  $Y(1)$  and  $Y(0)$  are “counterfactual” because they can’t be simultaneously observed. That is, we generally do not have identical replicas of the observational units that are simultaneously subject to both treatment and control.

The individual treatment effect is

$$Y(1) - Y(0).$$

This effect will vary across individuals as well as with other sources of randomness encoded in  $\omega$ . It is generally unrealistic to uncover the individual treatment effect,<sup>2</sup> but we can hope to estimate averages and the distribution of  $Y(d)$  at the population level to compute quantities such as the average treatment effect (ATE):

$$\delta = E(Y(1) - Y(0)) = EY(1) - EY(0).$$

Let  $D$  denote the realized treatment indicator, a random variable, which takes a value of 1 if the observational unit participated in the treatment and 0 otherwise.

1: Recall that a random variable  $V$  is a mapping  $\omega \mapsto V(\omega)$  from the underlying state of the world  $\omega \in \Omega$  to a real line (or other metric space, more generally) such that we can assign a probability law to it.

2: Unless, say, we had identical twins that can be put in treatment and control groups, and the only difference in outcomes between them is induced by treatment, i.e.  $\omega$  only depends on genetic makeup

**Assumption 2.1.1** (Consistency). *We observe*

$$Y := Y(D).$$

The assumption says that the observed wage outcome is equal to  $Y(1)$  after completion of a training program for a given person if she has completed the program (has  $D = 1$ ), and is equal to  $Y(0)$  if this person has not completed the training program (has  $D = 0$ ).

The following analytical example may help gain better understanding of the potential outcomes framework.

**Example 2.1.1** [Analytical Example] Consider the following model

$$\begin{aligned} Y(1) &:= \theta_1 + \epsilon_1 \\ Y(0) &:= \theta_0 + \epsilon_0 \\ D &:= 1(U > 0), \\ Y &:= Y(D), \end{aligned}$$

where  $\theta_0$  and  $\theta_1$  are constants, and  $(\epsilon_0, \epsilon_1, U)$  are jointly normal random stochastic disturbances with mean 0 and covariance matrix  $\Sigma$ . In this example  $EY(1) = \theta_1$ ,  $EY(0) = \theta_0$ , and the ATE is  $\delta = \theta_1 - \theta_0$ .

Under Assumption 2.1.1 we can identify the following conditional averages:

$$E[Y | D = d] = E[Y(d) | D = d], \text{ for } d \in \{0, 1\},$$

directly from the population data. The difference of the two averages gives us the average predictive effect (APE) of treatment status on the outcome:

$$\pi = E[Y | D = 1] - E[Y | D = 0].$$

It measures the association of the treatment status with the outcome.

While APE is identified – meaning computable from the population data – it may seem surprising (or not at all) that APE in general does not agree with the ATE  $\delta$ :

$$\delta \neq \pi. \tag{2.1.1}$$

The difference between the APE and ATE is generally said to

be due to *selection bias*. The meaning of selection bias is clarified through the following example, and clarified theoretically below.

**Example 2.1.2** (Selection Bias in Observational Data) Suppose we want to study the impact of smoking marijuana on life longevity. Suppose that smoking marijuana has no causal effect on life longevity:

$$Y = Y(0) = Y(1),$$

so that

$$\delta = EY(1) - EY(0) = 0.$$

However, the observed smoking behavior,  $D$ , is not assigned in an experimental study. The behavior is merely observed, and can be associated with poor health choices such as drinking alcohol, which are known to cause shorter life expectancy. In this case

$$\pi = E[Y | D = 1] - E[Y | D = 0] < 0 = \delta.$$

To sum up, in the smoking example, the observed "treatment" variable is potentially negatively associated with the potential health outcome, inducing the inequality  $E[Y(d)|D = 1] < E[Y(d)]$ .

**Example 2.1.3** (Analytical Version of the Smoking Example) To capture dependence between  $Y(d)$  and  $U$  in the smoking context analytically, we can go back to the Example 2.1.1, and make variables  $\epsilon_d$  and  $U$  be negatively associated:

$$E\epsilon_d U < 0.$$

The negative association between the  $\epsilon_d$  and  $U$  then results in the observed smoking status,  $D$ , being negatively associated with the potential outcomes  $Y(d)$ . Specifically, we have

$$E[Y|D = 1] < E[Y|D = 0],$$

which can be verified through additional analytical calculations or via simulation experiments (a homework).

It is useful to emphasize the main reason for having selection bias (2.1.1) is that

$$E[Y(d)|D = 1] \neq E[Y(d)]$$

whenever  $D$  is not independent of  $Y(d)$ . If  $D$  and  $Y(d)$  were independent,

$$E[Y(d)|D = 1] = E[Y(d)]$$

would hold since in this case  $D$  is uninformative about the potential outcome and drops out from the conditional expectation.

The problem with observational studies like our contrived Example 2.1.2 is that the "treatment" variable  $D$  is determined by individual behaviors which may be linked to potential outcomes. This linkage generates selection bias - the disagreement between APE and ATE. There are many ways of addressing selection bias, one of which is through an experiment, where we randomly assign the treatment to the units.

## Random Assignment/ Randomized Control Trials

The cleanest way to remove selection bias is through random assignment of treatment.

**Assumption 2.1.2** (Random Assignment/Exogeneity). *Suppose that treatment status is randomly assigned, namely  $D$  is statistically independent of each potential outcomes  $Y(d)$  for  $d \in \{0, 1\}$ , which is denoted as*

$$D \perp\!\!\!\perp Y(d)$$

*and  $0 < P(D = 1) < 1$ .*

This assumption states that the treatment assignment mechanism is purely random, and ensures that there are units in treatment and in control.

**Example 2.1.4** (Analytical Example Continued) In the analytical Example 2.1.1, Assumption 2.1.2 is satisfied if the stochastic shock  $U$  determining  $D$  is independent of stochastic shocks  $\epsilon_0$  and  $\epsilon_1$  determining  $Y(1)$  and  $Y(0)$ , i.e.

$$U \perp\!\!\!\perp (\epsilon_0, \epsilon_1)$$

A Randomized Control Trial, abbreviated RCT, occurs whenever the treatment  $D$  is randomly assigned.

**Theorem 2.1.1** (Randomization Removes Selection Bias) *Under Assumption 2.1.2, the average outcome in treatment group  $d$  recovers the average potential outcome under the treatment status*

$d$ :

$$\mathbb{E}[Y | D = d] = \mathbb{E}[Y(d) | D = d] = \mathbb{E}[Y(d)],$$

for each  $d \in \{0, 1\}$ . Hence the average predictive effect and average treatment effects coincide:

$$\begin{aligned}\pi &:= \mathbb{E}[Y | D = 1] - \mathbb{E}[Y | D = 0] \\ &= \mathbb{E}Y(1) - \mathbb{E}Y(0) =: \delta.\end{aligned}$$

Because of this property, RCTs are considered golden standard in causal inference, and are routinely employed in a variety of important settings. Examples uses include evaluating the efficacy of medical treatment, vaccinations, training programs, and other kinds of interventions.

**Example 2.1.5** (No Selection Bias in Experimental Data) Suppose that in the smoking example (Example 2.1.2), we worked with data where smoking or non-smoking was generated by perfectly enforced random assignment. In this case, we would have agreement between average predictive and treatment effects:  $\pi = \delta$ . While it is difficult to imagine a long-run RCT where study participants could be forced to smoke or not smoke marijuana (we discuss such limitations in Section 4), RCTs are routinely employed in a variety of other important settings.

## Statistical Inference with two sample means

Inference is based on the independent sample  $(Y_i, D_i)_{i=1}^n$  obtained from the experiment, where index  $i$  denotes the observational unit. We assume that each  $(Y_i, D_i)$  has the same distribution as  $(Y, D)$ . Estimation of two means  $\theta_d = \mathbb{E}[Y | D = d]$  for  $d = 0$  and  $d = 1$ , can be done via linear regression or, equivalently, by considering two group means

$$\hat{\theta}_d = \frac{\mathbb{E}_n Y 1(D = d)}{\mathbb{E}_n 1(D = d)}.$$

While the two means example can be treated as a special case of linear regression,<sup>3</sup> it is instructive to work out the details directly for the two group means, which are given in the appendix.

Under mild regularity conditions, we have that

$$\sqrt{n}\{\hat{\theta}_d - \theta_d\}_{d \in \{0,1\}} \xrightarrow{\text{a}} N(0, V),$$

3: Indeed, we can regress  $Y$  on  $D$  and  $1-D$ , that is estimate the model  $Y = \theta_1 D + \theta_0(1 - D) + U$ . We can then apply the inferential machinery developed in the previous chapter.

where

$$V = \begin{pmatrix} \frac{\text{Var}(Y_1(D=0))}{P(D=0)} & 0 \\ 0 & \frac{\text{Var}(Y_1(D=1))}{P(D=1)} \end{pmatrix}$$

so that  $\hat{\delta} = \hat{\theta}_1 - \hat{\theta}_0$  obeys

$$\sqrt{n}(\hat{\delta} - \delta) \xrightarrow{a} N(0, V_{11} + V_{22}).$$

To use this result in practice, variance components are usually estimated using the plug-in principle, which amounts to using the sample analogues of the expressions above.

Sometimes we are interested in relative effectiveness of treatment effects (for example, vaccine efficiency):

$$f(\theta) = (\theta_1 - \theta_0)/\theta_0 = \delta/\theta_0.$$

Relative effectiveness can be estimated by  $\hat{\delta}/\hat{\theta}_0 = f(\hat{\theta})$ , where  $\hat{\theta} = \{\hat{\theta}_d\}_{d \in \{0,1\}}$  and  $\theta = \{\theta_d\}_{d \in \{0,1\}}$ , with approximate distribution obtained using the *delta method*:

$$\sqrt{n}(f(\hat{\theta}) - f(\theta)) \approx G\sqrt{n}(\hat{\theta} - \theta) \xrightarrow{a} N(0, G'VG),$$

where  $G = \nabla f(\theta)$ ,  $\hat{\theta} = (\hat{\theta}_0, \hat{\theta}_1)', \theta = (\theta_0, \theta_1)'$ .<sup>4</sup>

The group mean estimation problem can also be formulated as a linear regression problem. We turn to exploring this formulation in Section 2.

- 4: The intuition is that the approximation  $\approx$  follows from application of the first order Taylor expansion and continuity of the derivative  $\nabla f$  at  $\theta$ .

## Pfizer/BioNTech Covid Vaccine RCT

Pfizer/BNTX<sup>5</sup> was the first vaccine approved for emergency use in the EU and US to reduce the risk of Covid-19 disease. See the Food and Drug Administration (FDA) [briefing](#) for details about the RCT and the summary data. Volunteers were randomly assigned to receive either a treatment (2-dose vaccination) or a placebo, without knowing which they received, and the doctors making the diagnoses did not know whether a given volunteer received a vaccination or not. In other words, the trial was a double-blind randomized control trial. The results of the study are presented in the following table.



**Figure 2.1:** Tozinameran (Pfizer-BioNTech Covid-19 vaccine); Image Source: Wikipedia

- 5: [Vaccinations RCT R Notebook](#) contains the analysis of the Pfizer-BioNTech Covid-19 Vaccine and Polio Vaccine RCTs.

Efficacy Endpoint Subgroup	BNT162b2 N=19965 Cases n <sup>b</sup>	Surveillance Time <sup>c</sup> (n <sup>d</sup> )	Placebo N=20172 Cases n <sup>b</sup>	Surveillance Time <sup>c</sup> (n <sup>d</sup> )	Vaccine Efficacy % (95% CI) <sup>e</sup>
Overall	9 2.332 (18559)		169 2.345 (18708)		94.6 (89.6, 97.6)
Age group (years)					
16 to 17	0 0.003 (58)		1 0.003 (61)		100.0 (-3969.9, 100.0)
18 to 64	8 1.799 (14443)		149 1.811 (14566)		94.6 (89.1, 97.7)
65 to 74	1 0.424 (3239)		14 0.423 (3255)		92.9 (53.2, 99.8)
≥75	0 0.106 (805)		5 0.109 (812)		100.0 (-12.1, 100.0)

The trial had to be large, because the rate of Covid-19 infection was relatively low at the time. Specifically, the treatment group saw 9 Covid-19 cases per 19,965, while the control group saw 169 cases per 20,172.

The estimated average treatment effect is about

$$-792.7$$

cases per 100,000, and the 95% confidence band is<sup>6</sup>

$$[-922, -664].$$

Under Assumptions 2.1.2 and 2.2.1 the confidence band suggests that the Covid-19 vaccine caused the reduction in the risk of contracting Covid-19.

We also compute the Vaccine Efficacy metric, which according to the [CDC](#) refers to the following measure:

$$VE = \frac{\text{Risk for Unvaccinated} - \text{Risk for Vaccinated}}{\text{Risk for Unvaccinated}}.$$

It describes the relative reduction in risk caused by vaccination. Estimating the VE is simple as we can plug-in the estimated group means. We can compute standard errors using the delta method or by simulation. We obtain that the overall vaccine efficacy is 94.6%, replicating the results shown in Figure 2.2. Our 95% confidence interval for VE, based on the normal approximation, is

$$[90.9\%, 98.2\%],$$

which differs only slightly from the FDA briefing table.<sup>7</sup>

**Remark 2.1.1** We notice that the confidence intervals for the VE for the two age groups of seniors are very wide, so to increase precision we pool them together and calculate the effectiveness of the vaccine for the two groups that are 65 or older. The resulting VE estimate is 95% and the two-sided

**Figure 2.2:** The aggregate data from the Pfizer RCT; source: FDA [briefing](#).

6: In this example, we don't need the underlying individual data to evaluate the effectiveness of the vaccine. This is because the outcomes are Bernoulli random variables  $Y(d) \in \{0, 1\}$  with mean  $EY(d)$  and variance  $\text{Var}(Y(d)) = EY(d)(1 - EY(d))$ .

7: The analysis in the FDA table is based on the inversion of exact binomial tests, the Cornfield procedure.

confidence interval based on the normal approximation is:

$$[82\%, 100\%]$$

A more refined approach is possible, based on the inversion of exact binomial ratio Cornfield tests [2], which we report in the notebook. This approach yields a much wider confidence interval of

$$[67\%, 99\%]$$

The reason is that the accumulated counts of binomials are too few for the Gaussian approximations to provide a high-quality approximation, so the exact binomial ratio test inversion delivers a more accurate confidence interval.

## 2.2 Pre-treatment Covariates and Heterogeneity

Sometimes we also have additional **pre-treatment** or **pre-determined** covariates  $W$ . We might be interested in either using these covariates to estimate average effects more precisely or to describe heterogeneity of the treatment effects. For example, we might be interested in the impact of a treatment across age or income groups.

For this purpose, we consider conditional average treatment effects (CATE):

$$\delta(W) = E[Y(1) | W] - E[Y(0) | W],$$

which compare the average potential outcomes conditional on a set of covariates  $W$ .

We can use RCTs to identify conditional average treatment effects by estimating the conditional predictive effects (CPE):

$$\pi(W) = E[Y | D = 1, W] - E[Y | D = 0, W].$$

**Assumption 2.2.1** (Random Assignment Independent of Covariates). *Suppose that treatment status is randomly assigned, namely  $D$  is statistically independent of both the potential outcomes and a set of pre-determined covariates:*

$$D \perp\!\!\!\perp (Y(d), W),$$

*and  $0 < P(D = 1) < 1$ .*

This assumption spells out that, if we plan to use covariates in the analysis, randomization has to be made with respect to these covariates as well. In practice, it is often tempting to use post-treatment covariates, but the use of such variables runs the danger of violating Assumption 2.2.1.

A common scenario where this may occur is when researchers encounter missing data from imperfect data collection in following-up with control and treated units to collect demographic information. When we drop observations with missing data, we implicitly condition on a post-treatment variable (missingness) which can cause violations of this assumption.

The desire to assess randomization with respect to covariates motivates the following diagnostic procedure.

**Testing Covariance Balance.** The random assignment assumption induces covariate balance. Namely, the distribution of covariates should be the same under both treatment and control:

$$W|D = 1 \sim W|D = 0,$$

and, equivalently,

$$D|W \sim D.$$

A useful implication is that  $E[W|D = 1] = E[W|D = 0]$ . That is,  $D$  does not predict any covariate feature; and moreover,  $D$  is not predictable by  $W$ :

$$E(D | W) = ED.$$

These latter conditions are easily testable using regression tools.

Theorem 2.1.1 continues to hold, but we now have a stronger result.

**Theorem 2.2.1** (Randomization with Covariates) *Under Assumption 2.2.1, the expected value of  $Y$  conditional on treatment status  $D = d$  and covariates  $W$  coincides with the expected value of potential outcome  $Y(d)$  conditional on covariates  $W$ :*

$$E[Y | D = d, W] = E[Y(d) | D = d, W] = E[Y(d)|W],$$

for each  $d$ . Hence the conditional predictive and average treatment effects agree:

$$\pi(W) = \delta(W).$$

## Regression and Statistical Inference for ATEs

We can base our inference on the ATE on linear regression.

Letting  $X = (1, W)$  be an intercept and the pre-treatment covariates  $W$ , let us write the BLP of each of  $Y(0)$  and  $Y(1)$  using  $X$  as

$$Y(d) = X'\beta_d + \varepsilon_d, \quad \varepsilon_d \perp X, \quad d = 0, 1. \quad (2.2.1)$$

Under Assumption 2.2.1, this coincides with the BLP of  $Y$  using  $X$  in the  $D = d$  population: letting  $\varepsilon = D\varepsilon_1 + (1 - D)\varepsilon_0$ , we have

$$Y = X'\beta_d + \varepsilon, \quad E(\varepsilon | D = d) = 0, \quad d = 0, 1. \quad (2.2.2)$$

The BLPs in each of the two populations,  $D = 0$  and  $D = 1$ , can be combined across the populations to state the BLP of  $Y$  using  $(X, DX)$  marginally:

$$Y = X'\beta_0 + DX'\beta_\delta + \varepsilon, \quad \varepsilon \perp (X, DX), \quad (2.2.3)$$

where  $\beta_\delta = \beta_1 - \beta_0$ . Such a linear rule is called *interactive* because it includes the interaction (meaning, product) of  $D$  and  $W$  as a regressor, in addition to  $D$  and  $W$ .

We assume that covariates are centered:<sup>8</sup>

$$EW = 0.$$

Since  $X$  contains an intercept,  $\varepsilon_d \perp X$  implies  $E\varepsilon_d = 0$ . Together with centered covariates, we find that

$$EY(d) = E(X'\beta_d + \varepsilon_d) = \beta_{d,1}.$$

This means that the ATE coincides with the coefficient on  $D$  in the BLP of  $Y$  using  $(X, DX)$ . That is,  $\beta_{\delta,1} = \delta$ .

We are often interested in the ATE and Relative ATE<sup>9</sup>

$$\delta \quad \text{and} \quad \delta/EY(0).$$

If we use OLS to estimate the BLP of  $Y$  using  $(X, DX)$ , then an application of the OLS theory in the previous chapter gives us that, under regularity conditions:

$$\begin{pmatrix} \sqrt{n}(\hat{\beta}_{\delta,1} - \delta) \\ \sqrt{n}(\hat{\beta}_{0,1} - EY(0)) \end{pmatrix} \xrightarrow{a} N(0, V),$$

<sup>8</sup>: Theoretically, this is implemented by re-defining  $W := W - EW$ . In estimation, this is implemented by re-defining  $W_i := W_i - \mathbb{E}_n W_i$ . This is equivalent to partialling out 1 from  $W$ .

<sup>9</sup>: Relative ATE is often called *lift* in many business applications.

where covariance matrix  $V$  has components:

$$(need to derive for the interactive model)$$

where  $\tilde{D} = D - ED$  is the residual after partialling out linearly  $X$  from  $D$ , where  $\tilde{1} := (1 - D)$  is the residual after partialling out  $D$  and  $X$  from 1; see Appendix for details.

We can then obtain the approximate normality for the Relative ATE using the delta method:

$$\sqrt{n}(\hat{\beta}_{\delta,1}/\hat{\beta}_{0,1} - \delta/EY(0)) \xrightarrow{d} N(0, G'VG),$$

where

$$G = [1/EY(0), -\delta/(EY(0))^2]'$$

## The Role of Covariates: Improve Precision of Estimating ATE

One role of covariates in randomized experiments is “denoising” or improving the precision of estimation of quantities like the ATE. We can rewrite (2.2.3) as

$$Y = \beta_{0,1} + D\beta_{\delta,1} + U, \quad U = W'\beta_0 + DW'\beta_\delta + \varepsilon.$$

From  $\varepsilon \perp (X, D, DX)$ ,  $EW = 0$ , and Assumption 2.2.1, we obtain that  $U \perp (1, D)$ , meaning that  $\beta_{0,1} + D\beta_{\delta,1}$  is the BLP of  $Y$  using  $(1, D)$ . We can therefore estimate the ATE as the coefficient on  $D$  either in the OLS of  $Y$  on  $(1, D)$  or in the OLS of  $Y$  on  $(X, DX)$ . The former exactly coincides with the unadjusted estimator  $\hat{\delta}$  from Section 2.1. The latter, nonetheless, has asymptotic variance that is always no larger than that of the former, and is therefore preferable. Both variances can be derived from Theorem 1.2.2, but in the former we partial out 1 from  $D$  and in the latter we partial out  $(1, W, DW)$  from  $D$ .

Estimating the ATE using OLS on  $(X, DX)$  is termed *post-stratification*. In particular, if  $W \in \{0, 1\}^p$  satisfies  $\sum_{j=1}^p W_j = 1$  so that  $W_j = 1$  is an indicator for a unit belonging to the  $j$ -th of  $p$  exhaustive and disjoint strata of the population, then this OLS procedure is equivalent to computing the sample ATE within each stratum as a simple difference of sample means, as we had done at first without any covariates but restricted to the stratum, and then averaging these over the strata using weights equal to the proportion of each stratum in the sample.

**Remark 2.2.1** (The Fragility of OLS with a Non-Interactive Model) Under Assumption 2.2.1 and  $EW = 0$ , the ATE also coincides with the coefficient on  $D$  in the BLP of  $Y$  using  $(D, X)$ :

$$Y = D\delta + X'\beta + V, \quad V \perp (D, X).$$

This suggests another estimator: OLS on  $(D, X)$ . While common and consistent, this estimator is ill-advised because it is **not** guaranteed to improve on the unadjusted estimator, and its asymptotic variance is never smaller than that of the post-stratification estimator using the interactive model. One special case is when  $V \perp\!\!\!\perp (D, X)$ , meaning that  $E(Y | D, X) = D\delta + X'\beta$  is linear in  $(D, X)$ ; then the variances coincide and therefore no degradation relative to the unadjusted estimator is guaranteed. This, however, may be too strong an assumption. The R notebook cited below constructs an example where controlling for predetermined covariates linearly lowers the precision (increases robust standard errors).

We can always **verify** whether there is an improvement by simply checking if the robust (Eicker-Huber-White) standard errors are smaller than the robust standard errors for the estimator without covariates. We present such a verification as part of the analysis of the Reemployment bonus RCT. Post-stratification with an interactive model does not suffer from this: it offers no degradation (and often stark improvement) regardless of linearity. Lin [3] provides an extensive comparison and is credited with popularizing the use of post-stratification with an interactive model.

## The Role of Covariates: Discover Heterogeneity through CATE

We may also want to explore heterogeneity in the treatment effects via CATE. In subsequent chapters we will discuss estimation of CATE using flexible models. For now, we focus on the BLP of CATE using  $X$ , which coincides with the BLP of  $Y(1) - Y(0)$  and hence with the difference of the BLPs of  $Y(1)$  and  $Y(0)$ : letting  $\varepsilon_\delta = \varepsilon_1 - \varepsilon_0$ ,

$$Y(1) - Y(0) = X'\beta_\delta + \varepsilon_\delta, \quad \varepsilon_\delta \perp X.$$

Recall that, as before,  $X = (1, W)$  and  $EW = 0$ , hence  $\beta_{\delta,1} = \delta$ . Note that while the BLPs of CATE and of  $Y(1) - Y(0)$  coincide, the regression errors do not.

[Covariates in RCT R Notebook](#) explores the use of covariates to both improve precision and learn about heterogeneity via a simulation experiment.

In particular, in the special case that  $\varepsilon_\delta \perp\!\!\!\perp X$ , we have that

$$\delta(X) = X'\beta_\delta,$$

so that CATE is *equal* to its BLP with *zero* error. However, even if this is not the case,  $X'\beta_\delta$  still represents a best linear projection of a nonlinear CATE. In either case, inference on  $\beta_\delta$  can provide valuable insight into the drivers of treatment effect heterogeneity. Since  $\beta_\delta$  is the coefficient on  $DX$  in the BLP of  $Y$  on  $(X, DX)$ , we can treat

$$\bar{D} := DX$$

as a vector of technical treatments<sup>10</sup> and invoke the “partialling out” approach for inference on components of  $\beta_\delta$ .

**Remark 2.2.2** (A Pseudo-Outcome Approach) Under Assumption 2.2.1, another way to express the BLP of CATE (equivalently, of  $Y(1) - Y(0)$ ) using  $X$  is as equal to the BLP of  $\frac{D - P(D=1)}{P(D=1)(1-P(D=1))} Y$  using  $X$ . Unlike CATE and  $Y(1) - Y(0)$ , this latter variable is directly observed in the data. Therefore, we can compute and make inferences on  $\beta_\delta$  by applying OLS to this so-called “pseudo-outcome.” In chapter ?? we will extend this pseudo-outcome approach to more general settings.

10: A technical treatment refers to any variable obtained as a transformation of the original treatment variable.

## Reemployment Bonus RCT

Here we re-analyze the Pennsylvania re-employment bonus experiment [4], which was conducted in the 1980s by the U.S. Department of Labor to test the incentive effects of alternative compensation schemes for unemployment insurance (UI). In these experiments, UI claimants were randomly assigned either to a control group or one of five treatment groups. We focus our discussion on treatment group 4. In the control group the current rules of the UI applied. Individuals in the treatment groups were offered a cash bonus if they found a job within some pre-specified period of time (qualification period), provided that the job was retained for a specified duration; see the Penn Data Codebook for further details on the data.

We consider

- ▶ classical 2-sample approach, no adjustment (CL)
- ▶ classical linear regression adjustment (CRA)
- ▶ interactive regression adjustment (IRA)
- ▶ interactive regression adjustment with double lasso (partialling out by lasso) (IRA-DL)

Reemployment Bonus RCT R Notebook explores the use of covariates to improve precision and learn about heterogeneity in a Reemployment Bonus RCT. The regression adjustment also corrects for imbalances in this RCT.

We use the last approach in the spirit of exploration and experimentation. We describe the last approach and establish its validity in Chapter 5.

Estimates of the ATE (est) on (log) unemployment duration and corresponding estimated standard errors (se) are given in the following table.

	CL	CRA	IRA	IRA-L
est	-0.08546	-0.07968	-0.07550	-0.07889
se	0.03586	0.03559	0.03560	0.03555

We see that treatment group 4 experiences an average decrease of about 7.8% in the duration of unemployment spells. The three regression estimators deliver estimates that are slightly more precise (have lower standard errors) than the simple difference in two means estimator, but essentially all methods have very similar standard errors.

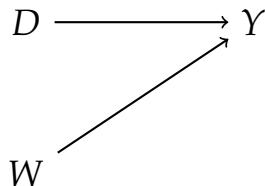
We also see that the regression estimators offer slightly lower estimates of the ATE than the difference in means estimator. These differences likely occur due to **imbalances** in the treatment allocation: people younger than 35 tended to receive the treatment more than other groups of qualified UI claimants. Loosely speaking, the regression estimators try to correct for this **imbalance** by "partialling out" the effect of this oversampling (see the notebook for the results from the balance check) and averaging over differences net of these "imbalancing" effects. We will explain how regression adjustment corrects for imbalances in the Chapter 4.

Finally, though not reported here, we see that there is not any statistically detectable heterogeneity in effects from looking at the IRA results given in the R Notebook for this example.

## 2.3 Drawing RCTs via Causal Diagrams

RCTs can be visualized using causal diagrams. These enable us to simply and clearly show the causal assumptions that underpin our model for retrieving treatment effects. The causal diagrams were introduced as early as 1920s by Sewall and Philip Wright ([5],[6]) and emerged as a fully formal tool due to the work of Judea Pearl and James H. Robins ([7], [8]).

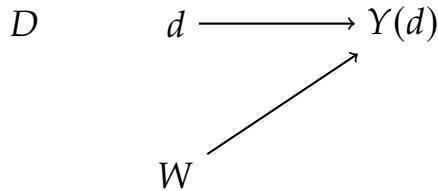
In causal diagrams, random variables are denoted by nodes and arrows between nodes represent causal effects. In our pure RCT set-up, we have that the assigned treatment variable causes outcome variable  $Y$ , and the pre-treatment variables  $W$  also cause the outcome variable  $Y$ , but they don't cause the treatment assignment  $D$ . This causal diagram for the pure RCT case is illustrated in Figure 2.3 below.<sup>11</sup>



<sup>11</sup>: In later chapters, we also consider generalized (stratified) RCTs, where pre-treatment variables also cause  $D$ .

**Figure 2.3:** A Causal Diagram for RCT

Figure 2.4 depicts a version of the diagram that also includes potential outcomes as nodes.



**Figure 2.4:** A Causal Diagram for the RCT Research Design

In Figure 2.4, we show the potential outcomes  $Y(d)$  as a single node. The pre-treatment covariates affect this node, which is represented by the arrow from  $W$  to the  $Y(d)$  node. The assigned treatment variable  $D$  is independent of the node  $Y(d)$ , which is shown by the absence of an arrow connecting the two nodes. The arrow from  $d$  to  $Y(d)$  shows the causal dependency of  $Y(d)$  on the deterministic node  $d$ . The assigned treatment  $D$  is also shown to be independent of the node  $W$ . The potential outcome process  $d \mapsto Y(d)$  and treatment assignment jointly determine the realized outcome variable  $Y$  via the assignment  $Y := Y(D)$ .

We further develop the use of these concepts and the use of causal diagrams as a formal tool in subsequent chapters.

## 2.4 The limitations of RCTs

Here, we briefly outline some of the primary limitations of RCTs as an identification tool. We first consider threats to identification, outlining settings in which the stable unit treatment value assumption (SUTVA), an important assumption that underpins

causal inference in an RCT setting, is unlikely to hold, and the implications for inference. We then address ethical and practical concerns in RCT implementation and generalizability.

## Externalities, Stability, and Equilibrium Effects

The traditional formulation of Rubin's causal model relies on SUTVA. This assumption requires that the potential outcome observation of one unit should be unaffected by the assignment of treatments to other units [9].

We consider some cases where this assumption might not hold:

In a vaccine example, this assumption holds if treatment and control populations are "small" (infinitesimal) subpopulations of the entire general population. Our methods measure the average vaccine effects in these settings. However, if we vaccinate a sufficiently large percentage of people, reaching herd immunity, the outcomes for the control group would be essentially the same as outcomes for the treated. SUTVA therefore would not hold.<sup>12</sup>

In economics we refer to such spillover effects as externalities, and in some contexts, we may call them general equilibrium effects. For example, there is a positive externality created by people who take the vaccine (and people that don't take vaccine "free ride", once the vaccination level is high enough).

Consider another example. We might want to study the earning effect of getting a college degree versus not having a college degree. If treatment will target a relatively small subpopulation of people, there likely won't be any large general equilibrium wage effects. On the other hand, if the treatment will target a large subpopulation, the equilibrium wage will likely adjust (the college wage premium might decrease, for example). In another example, the outcomes for one individual in large-scale training programs may be affected by the number of people trained to perform the same job.

12: Because SUTVA does not hold in the vaccination context, it is customary to use relative measures of impact like "vaccine efficiency" because they may be a somewhat more stable measure when generalizing from "small" treated subpopulations to a "large" treated population.

## Ethical, Practical and Generalizability Concerns

Many RCTs are infeasible because implementing them would be unethical. The general ethical principles and guidelines for research involving human subjects are set out in the 1978 Belmont report ([10]). The key ethical principles are: "Respect for persons", "Beneficence" and "Justice". Human subject trials

are subject to regulation by an institutional review board, which determines whether the trial is ethical with reference to these guiding principles, or whether it should be prevented from registering.

For example, we previously considered a hypothetical RCT where individuals are assigned to a smoking treatment group. The trial would violate the principle of "beneficence" as the researcher might be causing physical harm to study participants by assigning them to smoking. Thus, RCTs are rarely a feasible means of retrieving the causal effects of harmful interventions as they tend to be unethical.

RCTs may also face practical issues. They can be prohibitively expensive where the treatment is costly, data collection costs are high or the sample size required for adequate power is high. These issues make it difficult to implement long-term RCTs and find evidence on the long-term effects of interventions, particularly because they are more likely to suffer from attrition. It may also be politically infeasible for policymakers to enforce randomization of receipt of a desirable treatment.

Even in the best case, where an RCT is successfully implemented and we are confident in our retrieved average treatment effect, it may be difficult to generalize (or extrapolate) the result of an RCT in a specific context to a general finding. This difficulty might be because local conditions or implementation capacity materially differ between where interventions are staged or because the scale of the intervention is important.

## Notebooks

- ▶ [Vaccination RCT R Notebook](#) and [Vaccination RCT Python Notebook](#) contain the analysis of vaccination examples.
- ▶ [Covariates in RCT R Notebook](#) explores the use of covariates to improve precision and learn about heterogeneity via a simulation experiment.
- ▶ [Reemployment Bonus RCT R Notebook](#) explores the use of covariates to improve precision and learn about heterogeneity in a Reemployment Bonus RCT. The regression adjustment also corrects for imbalances in this RCT.

## Notes

RCTs have had a profound influence on business, economics and science more generally. For example, RCTs are routinely used to study the efficacy of drugs and efficacy of various programs in labor and development economics, among other subfields of economics. The FDA moved to RCTs as the gold standard of proving that treatments work in 1970s-80s. In the tech industry and marketing, RCTs are also called "A/B Testing" and are now widely used. Many major tech companies have their own experimental platforms to carry out thousands of experiments.

The expansion of the use of experimentation in economics is associated with the work of Abhijit Banerjee, Esther Duflo, and Michael Kremer, the recipients of Alfred Nobel Memorial Prize in Economics.<sup>13</sup>

We touched upon very basic ideas here. The basic random design is just one of many possible randomized designs that allow us to uncover causal effects. For an in-depth analysis of design of experiments, please see lecture notes by Art Owen ([11]). For standard RCTs and causal analysis more generally, see the book by Imbens and Rubin [12]. Duflo et al. [13] is another good investment. For how RCTs are done and designed in practice, see the FDA registry of RCTs; and see the American Economic Association for a registry of RCTs in economics. At MIT, [The Poverty Action Lab](#) runs large scale experiments in economics.

See, for example, [ExP platform at Microsoft](#), and the [WebLab platform](#) at Amazon.

13: "for their experimental approach to alleviating global poverty." Source: [NobelPrize.org](#)

## Study Questions

1. Set-up a simulation experiment that illustrates the contrived smoking example, following the analytical example we've presented in the text. Illustrate the difference between RCT (smoking generated independently of potential outcomes) and observational study (smoking choice is correlated with potential outcomes).
2. Sketch out the proof of the large sample properties of the two means estimator.
3. Study the notebook on vaccinations RCTs. Try to replicate the results in the FDA briefing table for each age 18-64

(exact replication is not required). Explain your calculations.

4. Study the notebook on the Reemployment example; experiment with putting even more flexible controls (e.g. use extra interactions of some controls); report your findings.
5. Work and experiment with the Covariates in RCT notebook. Explain the main points being made.
6. Skim over the information on the Pfizer RCT design [briefing](#). Write down one paragraph summarizing the study design.
7. Skim over one of the RCTs registered with [AEA RCT Registry](#). Write down one paragraph summarizing the study design.
8. Think of some RCTs where stability (SUTVA) is likely to hold and some RCTs where it likely does not.
9. Why can't we learn individual treatment effects by first putting a unit in treatment, and then in control second? Or the other way around? A hint is to think of all sources of randomness represented by  $\omega$ . Would the situation be different if you had a time machine?

## 2.A Approximate Distribution of the Two Sample Means

To demonstrate the result in the text we note that

$$\{\hat{\theta}_d - \theta_d\}_{d \in \{0,1\}} = \frac{\mathbb{E}_n\{Y(d) - \mathbb{E}Y(d)\}1(D = d)}{\mathbb{E}_n1(D = d)},$$

because we can re-write the population group average is

$$\theta_d = \mathbb{E}Y(d) = (\mathbb{E}Y(d))\frac{\mathbb{E}_n1(D = d)}{\mathbb{E}_n1(D = d)}.$$

Hence

$$\sqrt{n}\{\hat{\theta}_d - \theta_d\}_{d \in \{0,1\}} = \sqrt{n}\frac{\mathbb{E}_n\{Y(d) - \mathbb{E}Y(d)\}1(D = d)}{\mathbb{E}_n1(D = d)}.$$

By the law of large numbers,  $\mathbb{E}_n 1(D = d) \approx P(D = d)$ ; so we have the approximation

$$\sqrt{n} \{\hat{\theta}_d - \theta_d\}_{d \in \{0,1\}} \approx \sqrt{n} \frac{\mathbb{E}_n \{Y(d) - EY(d)\} 1(D = d)}{P(D = d)}.$$

Note also that the terms being averaged are

$$\frac{\{Y_i(d) - EY_i(d)\} 1(D_i = d)}{P(D_i = d)}.$$

These terms have zero mean (why?) and variance

$$\frac{[E\{Y(d) - EY(d)\}^2 1(D = d)^2]}{P(D = d)^2} = \frac{\text{Var}(Y 1(D = 0))}{P(D = d)}.$$

Also note the zero covariance:

$$E \left[ \frac{\{Y_i(1) - EY_i(1)\} 1(D_i = 1)}{P(D_i = 1)} \frac{\{Y_i(0) - EY_i(0)\} 1(D_i = 0)}{P(D_i = 0)} \right] = 0.$$

The application of the central limit theorem then yields the claimed result.

## 2.B Approximate Distribution for Intercept and Slope Estimators

Here we explain the details of the approximate normality for the estimators of  $\hat{\beta}_1$  and  $\hat{\alpha}$  in Section 2.2. The previous theory we've developed implies that the OLS estimator  $\hat{\alpha}$  obeys

$$\sqrt{n}(\hat{\alpha} - \alpha) \approx \sqrt{n} \frac{\mathbb{E}_n \epsilon \tilde{D}}{\mathbb{E}_n \tilde{D}^2} \stackrel{a}{\sim} N(0, V_{11}),$$

where  $\tilde{D} = D - ED$  is the residual after partialling out linearly  $X$  from  $D$  (why?), and where

$$V_{11} = \frac{E\epsilon^2 D^2}{(E\tilde{D}^2)^2}.$$

Applying the same theory for  $\beta_1$  (the intercept coefficient), yields:

$$\sqrt{n}(\hat{\beta}_1 - \beta_1) \approx \sqrt{n} \frac{\mathbb{E}_n \epsilon \tilde{1}}{\mathbb{E}_n \tilde{1}^2} \stackrel{a}{\sim} N(0, V_{22}),$$

where  $\tilde{1} := (1 - D)$  is the residual after partialling out  $D$  and  $X$  from 1 and

$$V_{22} = \frac{\text{E}\epsilon^2\tilde{1}^2}{(\text{E}\tilde{1}^2)^2}.$$

From the approximations above we can also establish that the estimators are jointly approximately normal with covariance:

$$V_{12} = \frac{\text{E}\epsilon^2\tilde{D}\tilde{1}}{\text{E}\tilde{1}^2\text{E}\tilde{D}^2}.$$

To explain the derivation, note that by partialling out  $D$  and  $W$  in  $X = (1, W)'$  from 1 and  $Y$ , we obtain

$$\tilde{Y} = \beta_1\tilde{1} + \epsilon; \quad \tilde{1} := (1 - D).$$

The projection of 1 on  $D$  and  $W$  is given by  $D$  since  $D$  is binary and we've assumed  $\text{EW} = 0$ . Hence we apply our previous OLS theory to obtain the claim.

# Bibliography

- [1] Donald B. Rubin. 'Estimating causal effects of treatments in randomized and nonrandomized studies.' In: *Journal of educational Psychology* 66.5 (1974), pp. 688–701 (cited on page 26).
- [2] Jerome Cornfield. 'A statistical problem arising from retrospective studies'. In: *Proceedings of the third Berkeley symposium on mathematical statistics and probability*. Vol. 4. University of California Press Berkeley, CA. 1956, pp. 135–148 (cited on page 33).
- [3] Winston Lin et al. 'Agnostic notes on regression adjustments to experimental data: Reexamining Freedman's critique'. In: *Annals of Applied Statistics* 7.1 (2013), pp. 295–318 (cited on page 37).
- [4] Yannis Bilias. 'Sequential testing of duration data: the case of the Pennsylvania 'reemployment bonus' experiment'. In: *Journal of Applied Econometrics* 15.6 (2000), pp. 575–594 (cited on page 38).
- [5] Philip G. Wright. *The tariff on animal and vegetable oils*. New York: The Macmillan company, 1928 (cited on page 39).
- [6] Sewall Wright. 'Correlation and Causation'. In: *Journal of Agricultural Research* 20.7 (1921), pp. 557–585 (cited on page 39).
- [7] Judea Pearl. 'Causal diagrams for empirical research'. In: *Biometrika* 82.4 (1995), pp. 669–688 (cited on page 39).
- [8] Sander Greenland, Judea Pearl, and James M. Robins. 'Causal diagrams for epidemiologic research'. In: *Epidemiology* 10.1 (1999), pp. 37–48 (cited on page 39).
- [9] David R. Cox. *Planning of experiments*. Wiley, 1958 (cited on page 41).
- [10] *The Belmont report: Ethical principles and guidelines for the protection of human subjects of research*. Tech. rep. National Commission for the Protection of Human Subjects of Biomedical and Behavioral Research, 1978 (cited on page 41).
- [11] Art Owen. '[A First Course in Experimental Design: Notes from Stat 263/363](#)'. Lecture notes. 2020 (cited on page 43).

- [12] Guido W. Imbens and Donald B. Rubin. *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press, 2015 (cited on page 43).
- [13] Esther Duflo, Rachel Glennerster, and Michael Kremer. ‘Using randomization in development economics research: A toolkit’. In: *Handbook of development economics* 4 (2007), pp. 3895–3962 (cited on page 43).

# Predictive Inference via Modern High Dimensional Linear Regression

## 3

Here we discuss the use of penalized regressions for constructing predictions in high-dimensional settings, particularly when  $p > n$ . We first motivate the high-dimensional setting as arising both from having a high-dimensional regressor set and from constructing technical regressors from raw regressors. We then discuss Lasso, which penalizes the size of the model by the sum of the absolute value of its coefficients. We then provide an overview of other penalized regression methods.

3.1 Linear Regression with High-Dimensional Covariates . . . . .	50
The Framework . . . . .	50
Lasso . . . . .	54
3.2 Predictive Performance of Lasso and Post-Lasso . . . . .	59
3.3 A Helicopter Tour of Other Penalized Regression Methods for Prediction . . . . .	61
3.4 Choice of Regression Methods in Practice . . . . .	66
3.A Additional Discussion and Results . . . . .	68
Iterative Estimation of $\sigma$ . . . . .	68
Some Lasso Heuristics via Convex Geometry* . . . . .	68
Other Variations on Lasso70 Cross-Validation . . . . .	71

## 3.1 Linear Regression with High-Dimensional Covariates

### The Framework

We consider a regression model

$$Y = \beta'X + \epsilon, \quad \epsilon \perp X,$$

where  $\beta'X$  is the population best linear predictor of  $Y$  using  $X$ , or simply the population linear regression function. The vector  $X = (X_j)_{j=1}^p$  is  $p$ -dimensional. That is, there are  $p$  regressors, and

$p$  is large, possibly much larger than  $n$ .

This case where  $p$  is very large is what we call a "high-dimensional" setting. High-dimensional settings arise when

- ▶ data have large dimensional features (i.e. many covariates are available for use as regressors),
- ▶ we construct many technical regressors<sup>1</sup> from raw regressors, or
- ▶ both.

1: A *technical regressor* is any variable obtained as a transformation of a basic regressor.

Examples of datasets where many covariates are available and potential corresponding exemplary applications include

- ▶ country characteristics in cross-country wealth analysis,
- ▶ housing characteristics in house pricing/appraisal analysis,
- ▶ individual health information in electronic health records and claims data, and
- ▶ product characteristics at the point of purchase in demand analysis.

Another source of high-dimensionality is the use of constructed features or regressors. If  $W$  are "raw" regressors/features, **technical (constructed) regressors** are of the form

$$X = P(W) = (P_1(W), \dots, P_p(W))',$$

where the set of transformations  $P(W)$  is sometimes called the "dictionary" of transformations. Example transformations include polynomials, interactions between variables, and applying functions such as the logarithm or exponential. In the wage analysis in Chapter 1, for example, we used quadratic and cubic transformations of experience, as well as interactions

(products) of these regressors with education and geographic indicators.

The main motivation for the use of constructed regressors is to build **more flexible and potentially better** prediction rules. The potential for improved prediction arises because we are using prediction rules  $\beta'X = \beta'P(W)$  that are **nonlinear** in the original raw regressors  $W$  and may thus capture more complex patterns that exist in the data. Conveniently, the prediction rule  $\beta'X$  is still linear with respect to the parameters,  $\beta$ , and with respect to the constructed regressors  $X = P(W)$ , so inherits much from the previous discussion of linear regression provided in Chapter 1.

In the population, the **best predictor** of  $Y$  given  $W$  is

$$g(W) = E(Y | W),$$

the **conditional expectation** of  $Y$  given  $W$ . The function  $g(W)$  is called the **regression function** of  $Y$  on  $W$ . Specifically, the conditional expectation function  $g(W)$  solves the best prediction problem<sup>2</sup>

$$\min_{m(W)} E(Y - m(W))^2$$

Here we minimize the mean squared prediction error (MSE) among all prediction rules  $m(W)$  (linear or nonlinear in  $W$ ).

As the conditional expectation solves the same problem as the best linear prediction rule among a larger class of candidate rules, the conditional expectation generally provides better predictions than the best linear prediction rule (unless the conditional expectation function turns out to be linear, in which case the conditional expectation and best linear prediction rule coincide).

By using  $\beta'P(W)$  we are implicitly approximating the best predictor  $E(Y|W)$ . Indeed, it can be shown that for any parameter  $b$

$$E(Y - b'P(W))^2 = E(g(W) - b'P(W))^2 + E(Y - g(W))^2,$$

That is, the mean squared prediction error is equal to the mean squared approximation error of  $b'P(W)$  to  $g(W)$  plus a constant that does not depend on  $b$ . Therefore, minimizing the mean squared prediction error is the same as minimizing the mean squared approximation error. Thus, the **BLP**  $\beta'P(W)$  is the **Best Linear Approximation (BLA)** to the best predictor – the regression function  $g(W)$ .

2: This follows by rewriting the objective function as

$$\min_{m(W)} E[E[(Y - m(W))^2 | W]],$$

and noting that it is bounded below by

$$E[\min_{\mu \in \mathbb{R}} E[(Y - \mu)^2 | W]]$$

$$= E[E(Y - g(W))^2].$$

By using a richer and richer dictionary  $P(W)$  of transformations, the BLA  $\beta'P(W)$  approximates  $g(W)$  better and better.

**Example 3.1.1** (Approximating A Smooth Function with a Polynomial Dictionary) Suppose  $W \sim U(0, 1)$ , and

$$g(W) = \exp(4 \cdot W).$$

We use

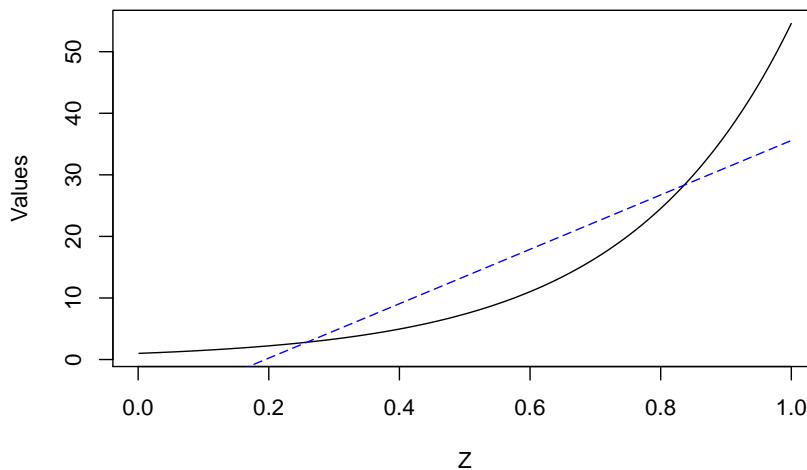
$$P(W) = \underbrace{(1, W, W^2, \dots, W^{p-1})'}_{p \text{ terms}}$$

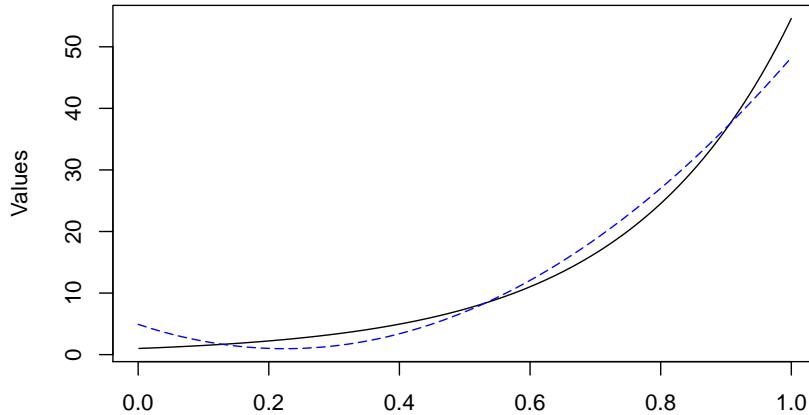
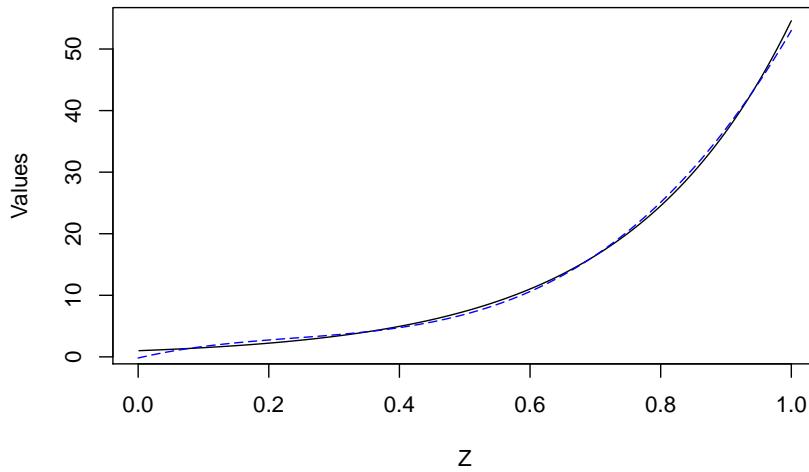
to form the BLA/BLP,  $\beta'P(W)$ . Figure ?? provides a sequence of panels that illustrate the approximation properties of the BLA/BLP corresponding to  $p = 2, 3$ , and  $4$ :

- ▶ With  $p = 2$  we get a linear-in- $W$  approximation to  $g(W)$ . As the figure shows, the quality of this approximation is poor.
- ▶ With  $p = 3$  we get a quadratic-in- $W$  approximation to  $g(W)$ . Here, the approximation quality is markedly improved relative to  $p = 2$  though approximation errors are still clearly visible.
- ▶ With  $p = 4$  we get a cubic-in- $W$  approximation to  $g(W)$ , and the quality of approximation appears to be excellent.

This simple example highlights the motivation for using non-linear transformations of raw regressors in linear regression analysis.

**Best Linear Approximation of  $g(Z)$  with  $p=2$**



**Best Linear Approximation of  $g(Z)$  with  $p=3$** 

**Best Linear Approximation of  $g(Z)$  with  $p=4$** 


There are many ways of generating flexible approximations, which are studied by approximation theory and nonparametric statistical learning theory. See, e.g., Tsybakov [1]. We will also consider nonlinear approximations using trees and neural networks in Chapter 7.

When we have multiple variables, we may generate transformations of each of the variables and employ interactions – products involving these terms. We already applied this approach in the wage analysis example in Chapter 1. As another simple concrete example, consider a case with two raw regressors,  $W_1$  and  $W_2$ . We could build polynomials of second order in each of the raw regressors –  $(1, W_1, W_1^2)$ ,  $(1, W_2, W_2^2)$ . We may then collect these variables along with the interaction in the raw regressors,  $W_1 W_2$  in a vector

$$(1, W_1, W_2, W_1^2, W_2^2, W_1 W_2)$$

for use in a regression model. There are, of course, many other possibilities such as considering higher order polynomial terms, e.g.  $W_1^3$ ; higher order interactions, e.g.  $W_1^2 W_2$ ; other nonlinear

transformations, e.g.  $\log(W_1)$ ; and many others.

In summary, we have provided two motivations for using high-dimensional regressors in prediction:

- ▶ The first motivation is that modern datasets have high-dimensional features that can be used as regressors.
- ▶ The second motivation is that we can use non-linear transformations of features or raw regressors and their interactions to form constructed regressors. Using transformations allows us to better approximate the ultimate and best prediction rule – the conditional expectation of the outcome given raw regressors.

## Lasso

Recall that we are considering a regression model

$$Y = \beta'X + \epsilon = \sum_{j=1}^p \beta_j X_j + \epsilon, \quad \epsilon \perp X$$

where  $p$  is possibly much larger than  $n$ .

We further assume that regressors are normalized,

$$EX_j^2 = 1,$$

to discuss theoretical properties. However, the estimation algorithms provided are stated without assuming this normalization.

Classical linear regression or least squares fails in these high-dimensional settings because it overfits the data. This is especially apparent when  $p \geq n$ . We therefore make some assumptions and modify the regression method in order to deal with cases where  $p$  is large.

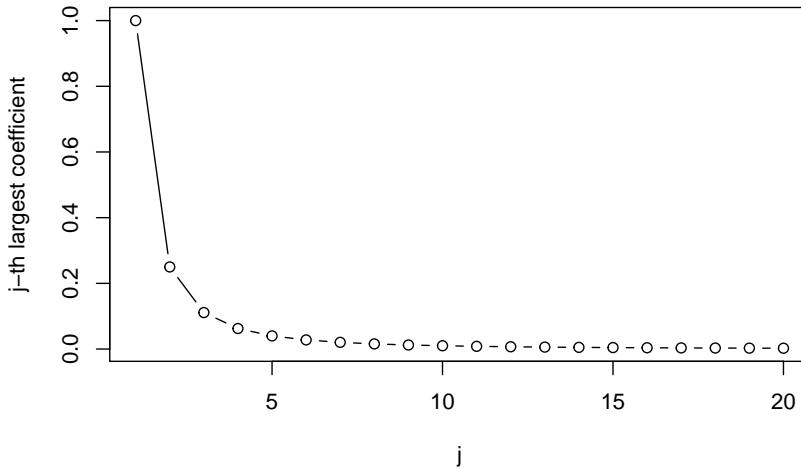
An intuitive starting point is the assumption of **approximate sparsity**. Under approximate sparsity, there is a small group of regressors with relatively large coefficients whose use alone suffices to approximate the BLP  $\beta'X$  well. The rest of the regressors are assumed to have relatively small coefficients and contribute little to the approximation of the BLP.

An example of approximate sparsity is captured by regression coefficients of the form<sup>3</sup>

$$\beta_j \propto 1/j^2, \quad j = 1, \dots, p.$$

3: The notation  $\propto$  reads as “proportional to”.

Here, the first few coefficients capture almost all of the explanatory power of the full vector of coefficients as shown in Figure 3.1.



**Figure 3.1:** Example of regression coefficients,  $\beta_j = 1/j^2$  that satisfy approximate sparsity.

Next, we define approximate sparsity formally.

**Definition 3.1.1 Approximate sparsity:** The sorted absolute values of the coefficients decay fast enough. Specifically, the  $j^{th}$  largest coefficient (in absolute value) denoted by  $|\beta|_{(j)}$  obeys

$$|\beta|_{(j)} \leq Aj^{-a}, \quad a > 1/2, \quad (3.1.1)$$

for each  $j$ , where the constants  $a$  and  $A$  do not depend on the sample size  $n$ .

For estimation purposes we have a random sample  $(Y_i, X_i)_{i=1}^n$ . We seek to construct a good linear predictor  $\hat{\beta}'X$ , which works well when  $p/n$  is not small.

**Lasso** constructs  $\hat{\beta}$  as the solution of the following penalized least squares problem:

$$\min_{b \in \mathbb{R}^p} \sum_i (Y_i - b'X_i)^2 + \lambda \cdot \sum_{j=1}^p |b_j| \hat{\psi}_j, \quad (3.1.2)$$

which is called the Lasso regression problem. The first term is  $n$  times the sample mean squared error, and the second term is called a *penalty term*. The penalty term introduces a cost to the size of the prospective model where size is captured by the sum of the products of the absolute values of the coefficients  $b_j$  with the *penalty loadings*  $\hat{\psi}_j$  all multiplied by the *penalty level*  $\lambda$ . The penalty loadings are typically set

as

$$\hat{\psi}_j = \sqrt{\mathbb{E}_n X_{ij}^2}.$$

The use of this penalty ensures invariance of Lasso predictions to rescaling  $X'_j$ . It is also desirable to demean  $X'_j$ 's other than the intercept, as this will ensure invariance of predictions to both location and scale transformations of  $X'_j$ 's.

As long as  $\lambda > 0$ , the introduction of the penalty term in (3.1.2) leads to a prediction rule which is less complex than the rule that would be obtained via solving the unpenalized least squares problem. This preference for less complex models then helps guard against overfitting which can intuitively be understood as adding complexity to a prediction rule to capture some small variation in the sample that does not generalize out of sample.

A crucial point is thus the choice of the penalization parameter  $\lambda$ . A theoretically valid choice is <sup>4</sup>

$$\lambda = 2 \cdot c \hat{\sigma} \sqrt{n} \Phi^{-1}(1 - \alpha/2p) \quad (3.1.3)$$

with  $\hat{\sigma} \approx \sigma = \sqrt{\mathbb{E}\epsilon^2}$  obtained via an iteration method defined in Appendix A, where  $c > 1$  and  $1 - \alpha$  is a confidence level.<sup>5</sup> We can further simplify the choice using Feller's tail inequality:

$$\Phi^{-1}(1 - \alpha/2p) \leq \sqrt{2 \log(2p/\alpha)},$$

where the inequality becomes sharp as  $p \rightarrow \infty$ .

This penalty level ensures that the Lasso predictor  $\hat{\beta}'X$  does not overfit the data and delivers good predictive performance under approximate sparsity ([2, 3]). Another good way to pick the penalty level is by cross-validation ([4]).<sup>6</sup>

Lasso imposes the approximate sparsity condition on the coefficients  $b = \hat{\beta}$ . Approximate sparsity is produced because the penalty function has a kink at zero, so the marginal cost of including regressor  $X_j$  is always positive ( $\lambda \hat{\psi}_j > 0$ ). Therefore, Lasso includes a regressor  $X_j$  only if its marginal predictive ability is higher than this threshold. We explain this point and how this feature of Lasso means that Lasso does variable selection in more detail below.

4:  $\Phi^{-1}$  denotes the quantile function (inverse) of the distribution function the standard normal variable  $N(0, 1)$ ,  $\Phi(z) = P((N(0, 1) \leq z)$

5: Practical recommendations, based on theory and working well in practice, include the choices of  $c = 1.1$  and  $\alpha = .05$ .

6: Cross-validation is a form of a repeated data-splitting method to choose penalty parameters for Lasso and to choose among predictive models more generally. We outline the basic idea of cross-validation in Section 7.

**Example 3.1.2** (Simulation Example) Consider

$$Y = \beta'X + \epsilon, \quad X \sim N(0, I_p), \quad \epsilon \sim N(0, 1),$$

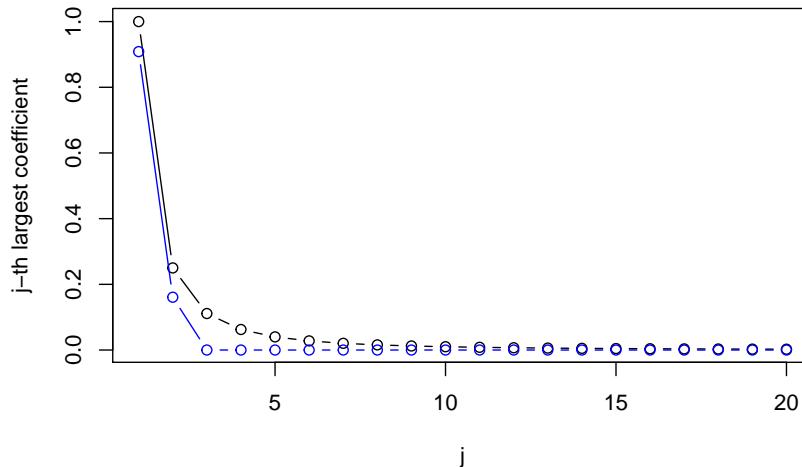
with approximately sparse regression coefficients:

$$\beta_j = 1/j^2, \quad j = 1, \dots, p$$

and

$$n = 300, \quad p = 1000.$$

Figure 3.2 shows that  $\hat{\beta}$  is sparse and is close to  $\beta$ . We see that Lasso sets most of regression coefficients to zero. It figures out approximately the right set of regressors, including only those with the two largest coefficients.



**Figure 3.2:** The true coefficients (black) vs. coefficients estimated by Lasso (blue) in Example 3.1.2.

**Remark 3.1.1** (Quick Intuitions and Heuristics\*) Assume  $\hat{\psi}_j = 1$  for simplicity. The  $j$ -th component  $\hat{\beta}_j$  of the Lasso estimator  $\hat{\beta}$  is set to zero if the marginal predictive benefit of changing  $\hat{\beta}_j$  away from zero is smaller than the marginal increase in penalty:

$$\hat{\beta}_j = 0 \text{ if } \left| \partial_{\beta_j} \sum_i (Y_i - \hat{\beta}' X_i)^2 \right| < \lambda.$$

That is,

$$\hat{\beta}_j = 0 \text{ if } |\hat{S}_j| < \lambda, \quad \hat{S}_j = 2 \sum_i (Y_i - \hat{\beta}' X_i) X_{ij}.$$

We discuss more detailed heuristics for penalty level selection in the appendix, but the rough idea is that the penalty should dominate the noise  $S_j = 2 \sum_i (Y_i - \beta' X_i) X_{ij}$  in the measurement of the marginal predictive ability. By the high-

dimensional central limit theorem ([5]), we have that

$$(S_j)_{j=1}^p \stackrel{d}{\sim} 2\sqrt{n}\sigma(\mathcal{N}_j)_{j=1}^p, \quad \mathcal{N}_j \sim N(0, 1).$$

Therefore, we need to dominate

$$2\sigma\sqrt{n} \max_{j=1,\dots,p} |\mathcal{N}_j|$$

with probability at least  $1 - \alpha$ . Then by the union bound and symmetry of centered normal variables:

$$\begin{aligned} & P\left(\max_{j=1,\dots,p} |\mathcal{N}_j| > \Phi^{-1}(1 - \alpha/2p)\right) \\ & \leq 2 \sum_{j=1}^p P(\mathcal{N}_j > \Phi^{-1}(1 - \alpha/2p)) \\ & = 2p\left(1 - \Phi(\Phi^{-1}(1 - \alpha/2p))\right) = 1 - \alpha. \end{aligned}$$

The union bound here is crude but the bound is not very loose; in particular when  $\mathcal{N}_j$ 's are independent, the bound is becoming sharp as  $p \rightarrow \infty$ . Finally setting

$$\lambda = 2\sigma\sqrt{n}\Phi^{-1}(1 - \alpha/2p)$$

we conclude that

$$P\left(\max_j |S_j| \leq \lambda\right) \geq 1 - \alpha,$$

up to a vanishing error.

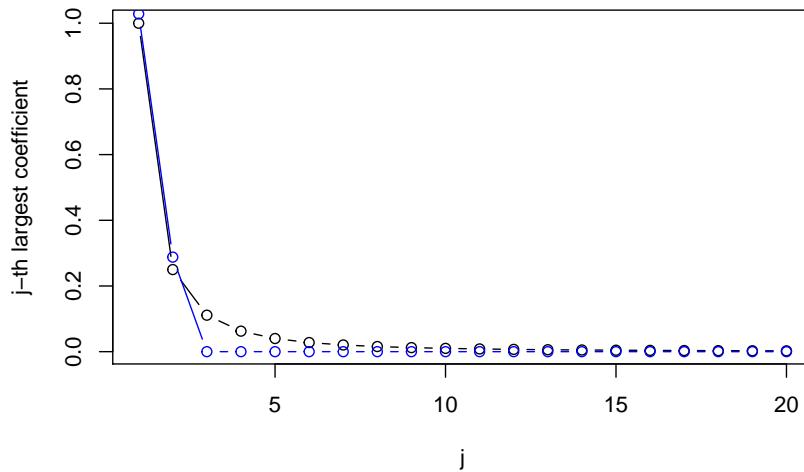
## OLS Post-Lasso

We can use the Lasso-selected set of regressors, those regressors whose Lasso coefficient estimates are non-zero, to refit the model by least squares. This method is called “least squares post Lasso” or simply **Post-Lasso** ([3]).

**Post-Lasso.** We define the Post-Lasso

$$\tilde{\beta} \in \arg \min_{\beta \in \mathbb{R}^p} \sum_i (Y_i - X'_i \beta)^2 : \beta_j = 0 \text{ if } \hat{\beta}_j = 0, \text{ for each } j, \quad (3.1.4)$$

where  $\hat{\beta}$  is the Lasso coefficient estimator. The formal properties of the Post-Lasso estimator  $\tilde{\beta}$  are similar to those of Lasso  $\hat{\beta}$ , as recorded in the next section.



**Figure 3.3:** The true coefficients (black) vs. coefficients estimated by Post-Lasso (blue) in the Example 3.1.2.

## 3.2 Predictive Performance of Lasso and Post-Lasso

The best linear prediction rule (out-of-sample) is  $\beta'X$ .

- Does  $\hat{\beta}'X$  provide a good approximation to  $\beta'X$ ?

We are trying to estimate  $p$  parameters  $\beta_1, \dots, \beta_p$ , imposing approximate sparsity via penalization. Under approximate sparsity, only a few, say  $s$ , parameters will be "important". We can call  $s$  the **effective dimension**. Lasso approximately figures out which parameters are important to keep. Further, intuitively, to estimate each of the "important"  $s$  parameters well, we need many observations for each such parameter. This means that  $n/s$  must be large, or, equivalently  $s/n$  must be small. Using previous reasoning from least squares theory with  $s < n$  regressors, we might also conjecture that the key determinant of the rate at which Lasso approximates the best linear predictor is  $\sqrt{s/n}$ . This conjecture is almost correct.

**Theorem 3.2.1** Under the approximate sparsity assumption 3.1.1, restricted isometry conditions stated below, and other regularity conditions stated e.g. in [3, 6], with probability approaching  $1 - \alpha$  as  $n \rightarrow \infty$ , the following bound holds:

$$\sqrt{\text{E}_X(\beta'X - \hat{\beta}'X)^2} \leq \text{const} \cdot \sqrt{\text{E}\epsilon^2} \sqrt{\frac{s \log(p \vee n)}{n}},$$

where  $\text{E}_X$  denotes expectation with respect to  $X$ , and the effective dimension is

$$s = \text{const} \cdot A^{1/a} \cdot n^{\frac{1}{2a}},$$

where constant  $a$  is the speed of decay of sorted coefficient values

in the approximate sparsity assumption. Moreover, the number of regressors selected by Lasso is bounded by

$$\text{const} \cdot s$$

with probability approaching  $1 - \alpha$  as  $n \rightarrow \infty$ . The constants const are different in different places and may depend on the law  $F$  of  $(Y, X)$  and  $\alpha$ .

Therefore, if  $s \log(p \vee n)/n$  is small, Lasso and Post-Lasso regression come close to the population regression function/best linear predictor. Relative to our conjectured rate  $\sqrt{s/n}$ , there is an additional factor  $\sqrt{\log(p \vee n)}$  in the bound. This factor captures the price of not knowing *a priori* which of the  $p$  regressors are the  $s$  important ones. Lasso approximately finds these important predictors, but correspondingly suffers a loss relative to a predictor estimated with knowledge of the best  $s$ -dimensional model. A theoretical guarantee similar to Theorem 3.2.1 has been established for cross-validated Lasso [4].

Under approximate sparsity and with appropriate choice of penalty parameters, Lasso and Post-Lasso will approximate the best linear predictor well. Theoretically, they will not overfit the data, and we can thus use the sample and adjusted  $R^2$  and  $MSE$  to assess out-of-sample predictive performance. Of course, it is always a good idea to verify the out-of-sample predictive performance by using sample splitting.

**On regularity conditions\***. A key regularity condition under which Theorem 3.2.1 can be established is the restricted isometry condition: Uniformly in all subvectors  $Z$  of  $X$  of dimension  $L = s \log(n \vee p)$ , the empirical Gram matrices  $\mathbb{E}_n ZZ'$  are close to their population analogues  $EZZ'$  in the operator norm, and the  $EZZ'$  have eigenvalues bounded away from zero and from above. A formal definition is as follows:

**Definition 3.2.1** (Restricted Isometry) *The following two conditions hold:*

*uniformly in  $Z \subset X : \dim(Z) \leq L$ ,*

$$\sup_{\|a\|=1} |a'(\mathbb{E}_n ZZ' - EZZ')a| \approx 0,$$

$$0 < C_1 \leq \inf_{\|a\|=1} a'EZZ'a \leq \sup_{\|a\|=1} a'EZZ'a \leq C_2 < \infty,$$

*where  $C_1$  and  $C_2$  are constants.*

This condition says that "small groups" of regressors are not

collinear and are well-behaved. This condition is simple and intuitive but is stronger than necessary. Results similar to Theorem 3.2.1 have been shown to hold under considerably weaker conditions. The condition  $\sup_{\|a\|=1} |a'(\mathbb{E}_n ZZ' - \mathbb{E} ZZ')a| \approx 0$  has been demonstrated to be valid under various more primitive conditions.

### 3.3 A Helicopter Tour of Other Penalized Regression Methods for Prediction

Instead of the Lasso penalty term, other penalty schemes can be used, leading to different regression estimators with different properties. These estimators are motivated by different structures for the coefficients on the set of regressors in a high-dimensional model. We consider three important settings: sparse, dense, and sparse+dense. Figure 3.4 illustrates each setting.

Throughout this section, we assume that regressors have been normalized to have second empirical moment equal to 1. We thus ignore coefficient specific penalty parameters like the  $\hat{\psi}_j$  in the Lasso problem (3.1.2).

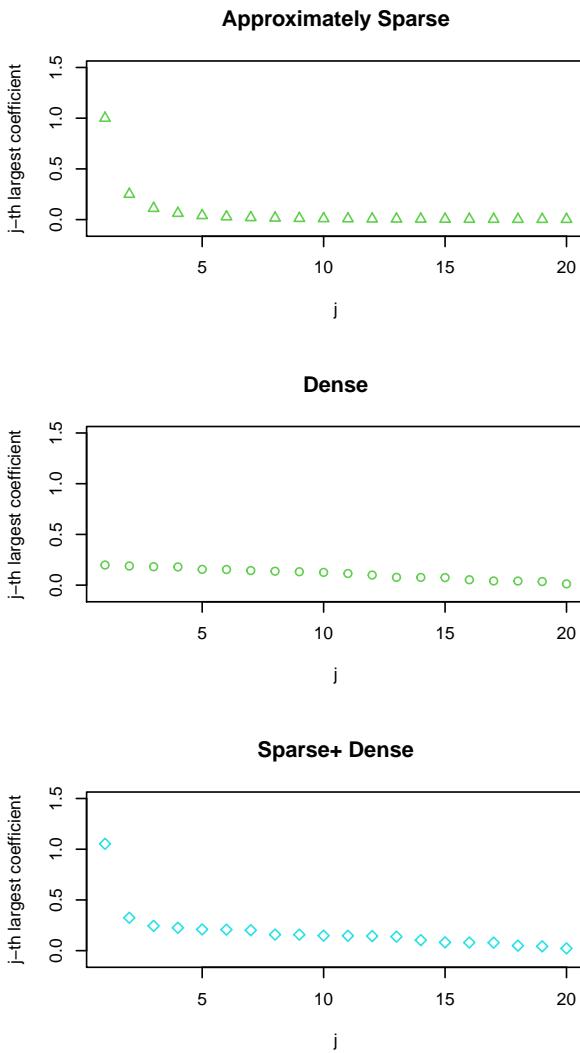
We have already outlined Lasso regression, which performs best in an approximately sparse setting.

We next consider the Ridge method, which performs best in the dense setting.

**Ridge.** The Ridge method estimates coefficients by penalized least squares, where we minimize the sum of squared prediction error plus the penalty term given by the sum of the squared values of the coefficients times a penalty level  $\lambda$ :

$$\hat{\beta}(\lambda) = \arg \min_{b \in \mathbb{R}^p} \sum_{i=1}^n (Y_i - b' X_i)^2 + \lambda \sum_j b_j^2.$$

Ridge balances the complexity of the model measured by the sum of squared coefficients with the goodness of in-sample fit. In contrast to Lasso, Ridge penalizes the large values of coefficients much more aggressively and small values much less aggressively – indeed, squaring big values makes them



**Figure 3.4:** The Lasso penalty is best suited for approximately sparse models, and the Ridge penalty for models with small dense coefficients. The Elastic Net can be tuned to perform well with either sparse or dense coefficients. The Lava penalty is best suited for models with coefficients generated as the sum of approximately sparse coefficients and small dense coefficients.

even bigger and squaring small numbers makes them even smaller.

Because of the latter property,

- ▶ Ridge does not set estimated coefficients to zero and so it does not do variable selection.
- ▶ The Ridge predictor  $\hat{\beta}'X$  is especially well suited for prediction in “dense” models, where the  $\beta_j$ ’s are all small without necessarily being approximately sparse.
- ▶ Ridge regression is also well suited when the matrix  $EX_iX_i'$  is poorly behaved, as measured by the decay of its eigenvalues to zero.

In the dense case, the Ridge predictor can easily outperform the Lasso predictor.

Like Ridge, the Lasso predictor can also deal with ill-behaved designs, although we don't understand its theoretical properties well in this case (in terms of conventional properties of matrices).

**Remark 3.3.1** (Theoretical Properties of the Ridge Procedure\*)

For excellent analysis of Ridge properties, see [7], who present the following bound for the fixed (conditional on)  $X_1, \dots, X_n$  case holding with high probability:

$$\mathbb{E}_n(\hat{\beta}'X - \beta'X)^2 \lesssim \sum_{j=1}^p \frac{\lambda^2 \lambda_j^2 \beta_j^2}{(\lambda_j^2 + \lambda)^2} + \frac{\mathbb{E}\epsilon^2}{n} \sum_{j=1}^p \left( \frac{\lambda_j^2}{(\lambda_j + \lambda)^2} \right),$$

where  $(\lambda_j)_{j=1}^p$  are eigenvalues of  $\mathbb{E}_n X_i X_i'$ . The theoretically optimal penalty level can be chosen to minimize the right hand side; in practice penalty is chosen by cross-validation. An analogous result is derived for random  $X_1, \dots, X_n$ ; see [7] for discussion.

The first component on the right hand side can be thought of as squared bias, and the second component is mean squared estimation error. Observe that when  $\lambda_j = 1$  and  $\lambda$  is bounded, the second term is of order  $p/n$ , which translates to the rate of  $\sqrt{p/n}$  after taking the square root. Having the second term go to 0 thus requires  $\sqrt{p/n} \rightarrow 0$ . In contrast,  $p$  can be larger than  $n$  and the second term can still vanish when eigenvalues decay to zero. In this case, the effective dimension for a given  $\lambda$  is

$$d(\lambda) = \sum_{j=1}^p \frac{\lambda_j^2}{(\lambda_j + \lambda)^2},$$

and the second term is of order  $d(\lambda)/n$ . The ratio  $d(\lambda)/n$  then determines the rate at which the Ridge predictor approximates the optimal predictor if the square bias term is of smaller order.

**Remark 3.3.2** (Connection to Principal Components\*) Consider the maximal set of mutually orthogonal rotations of the original  $X_i$ 's:

$$P_{ik} = c_k' X_i : \quad \mathbb{E}_n P_{ik} P_{il} = 0, \quad \mathbb{E}_n P_{ik} P_{ik} = 1; k = 1, \dots, n.$$

These rotations are called principal components of  $X_i$ . The

We will consider Principal Component Analysis in detail when we outline feature engineering in a later chapter.

Ridge prediction is given by

$$X_i' \hat{\beta} = \sum_{k=1}^n P_{ik} \frac{\lambda_j}{\lambda_j + \lambda} \mathbb{E}_n P_{ik} Y_i.$$

In principle, we can use principal components as features in all penalized methods; this is not a particular advantage of Ridge. For connection to "principal" components, see [Ridge vs PCA](#).

Finally, we note that, in practice, we can choose the penalty level  $\lambda$  in Ridge by cross-validation (or sample splitting).

Ridge and Lasso have other useful modifications or hybrids that can perform well in the sparse, dense or sparse + dense settings. One popular modification is the Elastic Net [8] that can perform well in either the sparse or the dense scenario with appropriate tuning (though not in the sparse+dense case).

**Elastic Net.** The Elastic Net method estimates coefficients by penalized least squares with the penalty given by a linear combination of the Lasso and Ridge penalties:

$$\hat{\beta}(\lambda_1, \lambda_2) = \arg \min_{b \in \mathbb{R}^p} \sum_i (Y_i - b' X_i)^2 + \lambda_1 \sum_j b_j^2 + \lambda_2 \sum_j |b_j|.$$

We see that the penalty function has two penalty levels  $\lambda_1$  and  $\lambda_2$ , which could be chosen by cross-validation in practice.

- ▶ By selecting different values of penalty levels  $\lambda_1$  and  $\lambda_2$ , we have more flexibility with Elastic Net for building a good prediction rule than with just Ridge or Lasso.
- ▶ The Elastic Net performs variable selection unless we completely shut down the Lasso penalty by setting  $\lambda_2 = 0$ .
- ▶ With proper tuning, Elastic Net works well in regression models where regression coefficients are either approximately sparse or dense.

We don't yet have good theoretical guarantees on predictive performance for the Elastic Net method.

Another way to combine the Lasso and ridge penalties is the Lava method, which is intended to work well in sparse+dense settings.

**Lava.** The Lava method ([9], [10]) estimates coefficients by solving the penalized least squares problem:

$$\hat{\beta}(\lambda_1, \lambda_2) = \arg \min_{b: b = \gamma + \delta \in \mathbb{R}^p} \sum_i (Y_i - b' X_i)^2 \\ + \lambda_1 \sum_j \gamma_j^2 + \lambda_2 \sum_j |\delta_j|.$$

Here components of the parameter vector are split into a "dense part"  $\gamma_j$  and "sparse part"  $\delta_j$ , where the  $\gamma_j$ 's are penalized like in Ridge, and the  $\delta_j$ 's are penalized like in Lasso. The minimization program automatically determines the best split into the dense and sparse parts. There are two corresponding penalty levels  $\lambda_1$  and  $\lambda_2$ , which can be chosen by cross-validation in practice.

- ▶ Compared to the Elastic Net, the Lava method penalizes large and small coefficients much less aggressively – large coefficients are penalized like Lasso and small coefficients like Ridge. Like Ridge, Lava does not do variable selection.
- ▶ Lava is designed to work well in

"sparse + dense"

regression models where there are several large coefficients and many small coefficients but which are not necessarily approximately sparse.

- ▶ With proper tuning that allows to set large values of either  $\lambda_1$  or  $\lambda_2$ , Lava can also work in either "sparse" or "dense" models.

Theoretical guarantees for this methods are given in [9] and [10]. Theoretically and practically, Lava can significantly outperform Lasso, Ridge and Elastic Net in "sparse+dense" models, and, with appropriate tuning, has comparable performance to Lasso in "sparse" models and to Ridge in "dense" models.

## 3.4 Choice of Regression Methods in Practice

How should we select the appropriate penalized regression method? The answer is simple if we are interested in building the best prediction. We can use data splitting into training and testing sets, and simply choose the method that performs the best on the test set. Rigorous theoretical guarantees for this approach have been provided by [11].

We show an example of this approach in the R notebook on predicting wages in CPS 2015 data below. We can also use ensemble methods to aggregate prediction methods to get boosts in predictive performance – we describe these aggregation methods in the the chapter on nonlinear regression methods.

## Notebooks

- ▶ [R Notebook on Penalized Regressions](#) provides details of implementation of different penalized regression methods and examines their performance for approximating regression functions in a simulation experiment. The simulation experiment includes one case with approximate sparsity and another case with both approximately sparse and dense components.
- ▶ [R Notebook on ML for Prediction of Wages](#) provides details of implementation of different penalized regression methods and examines their performance for predicting log-wages using CPS 2015 data. It also considers nonlinear models, which we will address in detail in Chapter 7.

## Notes

Lasso was introduced by Frank and Friedman [12], and its geometric and computational properties were elaborated on by Tibshirani [13], who also gave it its name. The first general theoretical analysis of Lasso was done by Bickel, Ritov, and Tsybakov [2]. There are many variations on the basic Lasso theme, only some of which we mentioned in this chapter. The properties of the post Lasso estimator in approximately sparse models (without assuming that Lasso perfectly selects the "right model") were first established in [3]. The properties of Lasso

and post Lasso don't hinge on the assumption of Gaussian or sub-Gaussian errors, though such assumptions are often imposed, as proven in [6]. Fundamentally, the properties of these procedures rely on a high-dimensional central limit theorem ([5]) that allows Gaussian approximations to key average-like quantities. While cross-validation has been frequently used to select the penalty level, validity of this approach for Lasso was only proven recently [4]. The lasso has been extended to time series and many time series by [14], with the corresponding package available at this [Link](#). The approach takes into account the temporal dependencies in the data when fitting lasso.

There is a large literature on Ridge estimation, with the reference [7] providing what seems to be the state of the art. The Lava approach is very recent and has been proposed and analyzed in [9] and [10]; the latter reference also discusses applications to problems with latent confounding (we discuss dealing with latent confounding in the chapter on instrumental variable methods), and for this reason refer to Lava as the spectral deconfounder.

## Study Problems

1. Solve the Lasso optimization problem analytically with only one regressor and interpret the solution.
2. Experiment with the R Notebook on Penalized Regressions, trying out modifications of the Monte-Carlo experiments. As examples, you might change parameters that govern the speed of decay of coefficients to zero, change the error distribution, or alter the structure of dependence among the design variables. Try to explain the results to a fellow student, linking explanations to the theoretical properties of these methods.
3. Experiment with the R Notebook on ML Prediction of Wages. Try to explain the results to a fellow student, linking explanations to the theoretical properties of these methods.

### 3.A Additional Discussion and Results

#### Iterative Estimation of $\sigma$

We can estimate  $\sigma$  using the following iterative method. Let  $X^0$  be a small set of regressors (a trivial choice is just the intercept, but we may include, for example, the five regressors that are most strongly correlated with  $Y_i$ 's). Let  $\hat{\beta}_0$  be the least squares estimator of the coefficients on the covariates associated with  $X^0$ , and define

$$\hat{\sigma}_0 := \sqrt{\mathbb{E}_n[(Y_i - X_i^0 \hat{\beta}_0)^2]}.$$

Set  $k = 0$ , and specify a small constant  $\nu \geq 0$  as a tolerance level and a constant  $K > 1$  as an upper bound on the number of iterations:

1. Compute the Lasso estimator  $\hat{\beta}$  based on the penalty level  $\lambda$  using  $\hat{\sigma}_k$ .
2. Set  $\hat{\sigma}_{k+1} = \sqrt{\mathbb{E}_n[(Y_i - X_i' \hat{\beta})^2]}$ .
3. If  $|\hat{\sigma}_{k+1} - \hat{\sigma}_k| \leq \nu$  or  $k > K$ , report  $\hat{\sigma} = \hat{\sigma}_{k+1}$ ; otherwise set  $k \leftarrow k + 1$  and go to (1).

#### Some Lasso Heuristics via Convex Geometry\*

Assume  $\hat{\psi}_j = 1$  for each  $j$  for simplicity, which amounts to normalizing regressors to have the second empirical moment equal to 1. Consider

$$\hat{\beta} \in \arg \min_{\beta \in \mathbb{R}^p} \hat{Q}(\beta) + \frac{\lambda}{n} \|\beta\|_1, \quad (3.A.1)$$

where

$$\hat{Q}(\beta) = \mathbb{E}_n[(Y_i - X_i' \beta)^2].$$

The key quantity in the analysis of (3.A.1) is the score – the gradient of  $\hat{Q}$  at the true value:<sup>7</sup>

$$S = \nabla \hat{Q}(\beta_0) = 2\mathbb{E}_n[X_i \epsilon_i].$$

The score  $S$  is the effective “noise” in the problem that should be dominated by the regularization. However, we would like to make the regularization bias as small as possible. This reasoning suggests choosing the smallest penalty level  $\lambda$  that is just large enough to dominate the noise with high probability, say  $1 - \gamma$ ,

7: In the case of a nonparametric model, the score is similar to the gradient of  $\hat{Q}$  at  $\beta_0$  but ignores the approximation errors  $r_i$ 's.

which yields

$$\lambda > c\Lambda, \text{ for } \Lambda := n\|S\|_\infty. \quad (3.A.2)$$

Here,  $\Lambda$  is the maximal score scaled by  $n$ , and  $c > 1$  is a theoretical constant that guarantees that the score is dominated.

It is useful to mention some simple heuristics for the principle (3.A.2) which arise from considering the simplest case where all of the regressors are irrelevant so that  $\beta_0 = 0$ . We want our estimator to perform at a near-oracle level in all cases, including this case, but here the oracle estimator  $\beta^*$  sets  $\beta^* = \beta_0 = 0$ . We also want  $\hat{\beta} = \beta_0 = 0$  in this case, at least with a high probability, say  $1 - \gamma$ . From the subgradient optimality conditions for (3.A.1), we must have

$$-S_j + \lambda/n > 0 \text{ and } S_j + \lambda/n > 0 \text{ for all } 1 \leq j \leq p \quad (3.A.3)$$

for the Lasso estimator for each coefficient to be exactly 0. We can guarantee (3.A.3) holds by setting the penalty level  $\lambda/n$  such that  $\lambda > n \max_{1 \leq j \leq p} |S_j| = n\|S\|_\infty$  with probability at least  $1 - \gamma$ , which is precisely what the rule (3.A.2) does.

Gaussian approximations to this score motivate the X-dependent penalty implementation in the package hdm.

**Remark 3.A.1** (Refining Penalty Levels) The X-dependent penalty level is specified as follows:

$$\lambda = c \cdot 2\hat{\sigma}\Lambda(1 - \gamma|X_1^n), \quad (3.A.4)$$

where

$$\Lambda(1 - \gamma|X_1^n) = (1 - \gamma) - \text{quantile of } n\|\mathbb{E}_n[X_i g_i / \Psi]\|_\infty \mid X_1^n,$$

$g_i$  are i.i.d.  $N(0, 1)$ , and  $\Psi = \text{diag}(\hat{\psi}_j)_{j=1}^p$ .  $\Lambda(1 - \gamma|X_1^n)$  can be thus be easily approximated by simulation. The use of normal errors  $g_i$  could be motivated by assuming the Gaussian errors  $\epsilon_i$  in the model or by appealing to a high-dimensional central limit theorem. We note that by the union bound and properties of the normal quantile function

$$\Lambda(1 - \gamma|X) \leq \sqrt{n}\Phi^{-1}(1 - \gamma/2p) \leq \sqrt{2n \log(2p/\gamma)}, \quad (3.A.5)$$

where  $\Phi^{-1}(a)$  denotes the  $a$ -th quantile of standard normal distribution. Thus,  $\sqrt{2n \log(2p/\gamma)}$  provides a simple upper bound on the penalty level.

Refined penalty levels are important when components of  $X_i$  are highly correlated, in which case the  $X$ -dependent penalty will be much lower than the bounds given in 3.A.5. Using the lower penalty level can offer both practical and theoretical boosts in performance in such cases.

## Other Variations on Lasso

Here and below we assume to simplify notation that

$$\hat{\psi}_j = 1, \quad j = 1, \dots, p.$$

A variant, called the *Square-root Lasso* estimator ([15],[16]), is defined as a solution to the following program:

$$\min_{\beta \in \mathbb{R}^p} \sqrt{\mathbb{E}_n[(Y_i - X'_i \beta)^2]} + \frac{\lambda}{n} \|\beta\|_1, \quad (3.A.6)$$

with the penalty level

$$\lambda = c \cdot \tilde{\Lambda}(1 - \alpha | X), \quad (3.A.7)$$

where  $c > 1$  and

$$\tilde{\Lambda}(1 - \alpha | X) = (1 - \alpha) - \text{quantile of } n \|\mathbb{E}_n[X_i g_i]\|_\infty / \sqrt{\mathbb{E}_n[g_i^2] | X_1^n},$$

with  $g_i \sim N(0, 1)$  independent for  $i = 1, \dots, n$ . As with Lasso, there is also a simple asymptotic option for setting the penalty level:

$$\lambda = c \cdot 2\sqrt{n} \Phi^{-1}(1 - \alpha/2p). \quad (3.A.8)$$

The main attractive feature of (3.A.6) is that the penalty level  $\lambda$  is independent of the value  $\sigma$ . This estimator has statistical performance that is as good as the iterative or cross-validated Lasso. Moreover, the estimator is a solution to a highly tractable conic programming problem:

$$\min_{t \geq 0, \beta \in \mathbb{R}^p} t + \frac{\lambda}{n} \|\beta\|_1 : \sqrt{\mathbb{E}_n[(Y_i - X'_i \beta)^2]} \leq t, \quad (3.A.9)$$

where the criterion function is linear in parameters  $t$  and positive and negative components of  $\beta$ , while the constraint can be formulated with a second-order cone, informally known also as the "ice-cream cone".

There are several other estimators that make use of penalization by the  $\ell_1$ -norm. An important case includes the Dantzig selector estimator [17]. It also relies on  $\ell_1$ -regularization but exploits the

notion that the residuals should be nearly uncorrelated with the covariates. The estimator is defined as a solution to:

$$\min_{\beta \in \mathbb{R}^p} \|\beta\|_1 : \|\mathbb{E}_n[X_i(Y_i - X'_i\beta)]\|_\infty \leq \lambda/n \quad (3.A.10)$$

where  $\lambda = \sigma\Lambda(1 - \alpha|X)$ . Here we focused our discussion on Lasso but virtually all theoretical results carry over to other  $\ell_1$ -regularized estimators including (3.A.6) and (3.A.10). We also refer to [18] for a feasible Dantzig estimator that combines the square-root Lasso method (3.A.9) with the Dantzig method.

## Cross-Validation

Cross-validation is a common practical tool that provides a way to choose tuning parameters such as the penalty level. The idea of cross-validation is to rely on repeated splitting of the training data to estimate the potential out-of-sample predictive performance.

### Definition 3.A.1 (Cross-Validation in Words)

- ▶ We partition the data into  $K$  blocks called "folds", for example, with  $K = 5$ , we split the data into 5 non-overlapping blocks.
- ▶ Leave one block out. Fit a prediction rule on all the other blocks. Predict the outcome observations in the left out block, and record the empirical Mean Squared Prediction Error. Repeat this for each block.
- ▶ Average the empirical Mean Squared Prediction Errors over blocks.
- ▶ We do these steps for several or many values of the tuning parameters and choose the value of the tuning parameter that minimizes the Averaged Mean Squared Prediction Error.

We can also consider many different methods for constructing prediction rules as well. For example, we could try Lasso with many different values of the penalty parameter and Ridge with many different values of the penalty parameter and choose the tuning parameter and method (Lasso or Ridge) that minimizes Mean Squared Prediction Error.

### Definition 3.A.2 (Cross-Validation: Formal Description)

- ▶ Randomly select a partition of observation indices  $1, \dots, n$  in  $K$  random folds  $B_1, \dots, B_K$ .

- For each  $k = 1, \dots, K$ , fit a prediction rule denoted by  $\hat{f}^{[k]}(\cdot; \theta)$ , where  $\theta$  denotes the tuning parameters such as penalty levels and  $\hat{f}^{[k]}$  depends only on observations with indices not in the fold  $B_k$ .
- For each  $k = 1, \dots, K$ , the empirical out-of-sample MSE for the block  $B_k$  is

$$MSE_k(\theta) = \frac{1}{m_k} \sum_{i \in B_k} (Y_i - \hat{f}^{[k]}(X_i; \theta))^2,$$

where  $m_k$  is the size of the block  $B_k$ .

- Compute the cross-validated MSE as

$$CV\text{-}MSE(\theta) = \frac{1}{K} \sum_{k=1}^K MSE_k(\theta).$$

- Choose the tuning parameter  $\hat{\theta}$  as a minimizer of  $CV\text{-}MSE(\theta)$ .

**Remark 3.A.2** (On Guarantees of Cross-Validated Predictors)

A common step people do in practice is to retrain the predictor  $\hat{f}(X)$  on the entire data with the best tuning parameter  $\hat{\theta}$  found by cross-validation. Theoretical properties of the resulting cross-validated predictor  $\hat{f}(X)$  are only well understood for some high dimensional problems (Lasso, see [4]).

**Remark 3.A.3** (Guarantees for Pooled Cross-Validated Estimator) On the other hand, there are rigorous theoretical guarantees for the pooled cross-validated predictor:

$$\hat{f}(X) = \frac{1}{K} \sum_{k=1}^K \hat{f}^{[k]}(X; \hat{\theta}),$$

which are provided by [19] and [11] who establish that the resulting prediction rule has optimal or near-optimal rates for approximating the best predictor in a given class.

Note that the pooled procedure is different from the default CV procedure implemented in many software packages and used in many applications.

# Bibliography

- [1] Alexandre B. Tsybakov. *Introduction to nonparametric estimation*. Springer, 2009 (cited on page 53).
- [2] Peter J. Bickel, Ya'acov Ritov, and Alexandre B. Tsybakov. 'Simultaneous analysis of Lasso and Dantzig selector'. In: *Annals of Statistics* 37.4 (2009), pp. 1705–1732 (cited on pages 56, 66).
- [3] Alexandre Belloni and Victor Chernozhukov. 'Least Squares After Model Selection in High-dimensional Sparse Models'. In: *Bernoulli* 19.2 (2013). ArXiv, 2009, pp. 521–547 (cited on pages 56, 58, 59, 66).
- [4] Denis Chetverikov, Zhipeng Liao, and Victor Chernozhukov. 'On cross-validated lasso in high dimensions'. In: *Annals of Statistics* 49.3 (2021), pp. 1300–1317 (cited on pages 56, 60, 67, 72).
- [5] Victor Chernozhukov, Denis Chetverikov, and Kengo Kato. 'Central Limit Theorems and Bootstrap in High Dimensions'. In: *Annals of Probability* 45.4 (2017), pp. 2309–2352 (cited on pages 58, 67).
- [6] Alexandre Belloni, Daniel L. Chen, Victor Chernozhukov, and Christian B. Hansen. 'Sparse Models and Methods for Optimal Instruments with an Application to Eminent Domain'. In: *Econometrica* 80.6 (2012). Arxiv, 2010, pp. 2369–2429 (cited on pages 59, 67).
- [7] Daniel Hsu, Sham M. Kakade, and Tong Zhang. 'Random design analysis of ridge regression'. In: *Conference on learning theory*. Vol. 23. JMLR Workshop and Conference Proceedings. 2012, pp. 9.1–9.24 (cited on pages 63, 67).
- [8] Hui Zou and Trevor Hastie. 'Regularization and variable selection via the elastic net'. In: *Journal of the royal statistical society: series B (statistical methodology)* 67.2 (2005), pp. 301–320 (cited on page 64).
- [9] Victor Chernozhukov, Christian Hansen, and Yuan Liao. 'A lava attack on the recovery of sums of dense and sparse signals'. In: *Annals of Statistics* 45.1 (2017), pp. 39–76 (cited on pages 65, 67).
- [10] Domagoj Ćevid, Peter Bühlmann, and Nicolai Meinshausen. 'Spectral deconfounding via perturbed sparse linear models'. In: *Journal of Machine Learning Research* 21 (2020), pp. 1–41 (cited on pages 65, 67).

- [11] Marten Wegkamp. 'Model selection in nonparametric regression'. In: *The Annals of Statistics* 31.1 (2003), pp. 252–273 (cited on pages 66, 72).
- [12] Ildiko E. Frank and Jerome H. Friedman. 'A statistical view of some chemometrics regression tools'. In: *Technometrics* 35.2 (1993), pp. 109–135 (cited on page 66).
- [13] Robert Tibshirani. 'Regression shrinkage and selection via the Lasso'. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1 (1996), pp. 267–288 (cited on page 66).
- [14] Victor Chernozhukov, Wolfgang Karl Härdle, Chen Huang, and Weining Wang. 'Lasso-driven inference in time and space'. In: *The Annals of Statistics* 49.3 (2021), pp. 1702–1735 (cited on page 67).
- [15] Alexandre Belloni, Victor Chernozhukov, and Lie Wang. 'Square-root lasso: pivotal recovery of sparse signals via conic programming'. In: *Biometrika* 98.4 (2011). Arxiv, 2010, pp. 791–806 (cited on page 70).
- [16] Alexandre Belloni, Victor Chernozhukov, and Lie Wang. 'Pivotal estimation via square-root lasso in nonparametric regression'. In: *Annals of Statistics (arXiv 2011)* 42.2 (2014), pp. 757–788 (cited on page 70).
- [17] Emmanuel Candès and Terence Tao. 'The Dantzig selector: statistical estimation when  $p$  is much larger than  $n$ '. In: *The Annals of Statistics* 35.6 (2007), pp. 2313–2351 (cited on page 70).
- [18] Eric Gautier and Alexander B. Tsybakov. 'High-Dimensional Instrumental Variables Regression and Confidence Sets'. In: *ArXiv working report* (2011) (cited on page 71).
- [19] Guillaume Lecué and Charles Mitchell. 'Oracle inequalities for cross-validation type procedures'. In: *Electronic Journal of Statistics* 6 (2012), pp. 1803–1837 (cited on page 72).

# Statistical Inference on Predictive and Causal Effects in High Dimensional Linear Regression Models

# 4

Here we discuss inference on predictive effects using double lasso methods, where we use lasso (at least) twice to residualize outcomes and a target covariate of interest, whose predictive effect we'd like to infer. The predictive effects coincide with causal structural effects under random assignment of treatment conditional on controls. The approach relies on approximate sparsity of the best linear predictors for the outcome and for the target covariate. The resulting estimator concentrates in a  $1/\sqrt{n}$  neighborhood of the true value and is approximately Gaussian, enabling construction of confidence bands. We explain the low bias property of the double lasso method using Neyman orthogonality, and isolate the latter as a critical property for further generalizations.

4.1 Introduction . . . . .	76
4.2 Inference with Double Lasso . . . . .	76
Inference on One Coefficient . . . . .	76
Application to Testing the Convergence Hypothesis .	79
Inference on Many Coefficients . . . . .	80
Discovering Heterogeneity in the Wage Pay Gap Analysis . . . . .	82
4.3 Why Partialling-out works: Neyman Orthogonality . .	84
Neyman Orthogonality . . . . .	84
What happens if we don't have Neyman Orthogonality? . . . . .	86
4.4 Other Approaches that have the Neyman Orthogonality Property . . . . .	89
Double Selection . . . . .	89
Desparsified Lasso . . . . .	89

## 4.1 Introduction

We recall the predictive effect question:

- ▶ How does the predicted value of  $Y$  change if a regressor  $D$  increases by a unit, while other regressors  $W$  remain unchanged?

As before, we denote the set of regressors as  $X = (D, W)$ . In Chapter 1, we discussed how we could use the population regression coefficient corresponding to the variable  $D$ , denoted  $\alpha$ , to answer this question. We also discussed how to estimate this effect and construct confidence intervals with regression. Now we turn to estimation and construction of confidence intervals for  $\alpha$  in the high-dimensional setting, using the tools we developed in Chapter 3.

Here we focus on using Lasso methods. We can use other penalized methods with the caveat that theoretical guarantees are not available unless we perform additional data-splitting. We will discuss the use of data-splitting and more general machine learning methods in detail when we introduce "Double Machine Learning" in Chapter 9.

**Remark 4.1.1** (Causal interpretation of predictive effects) The predictive effect question may have a causal interpretation within any model where conditioning on  $W$  is sufficient for identification of the structural/causal effect of  $D$  on  $Y$ . If conditioning on  $W$  is sufficient for identification of the causal effect of  $D$  on  $Y$ , the predictive effect question becomes a causal effect question:

- ▶ How does the predicted value of  $Y$  change if we **intervene** and increase  $D$  by a unit, holding  $W$  fixed?

## 4.2 Inference with Double Lasso

### Inference on One Coefficient

The key to inference will be the application of Frisch-Waugh-Lovell partialling-out. Consider the simple predictive model:

$$Y = \alpha D + \beta' W + \epsilon, \quad (4.2.1)$$

where  $D$  is the target regressor and  $W$  consists of  $p$  controls. After partialling-out  $W$ ,

$$\tilde{Y} = \alpha \tilde{D} + \epsilon, \quad E\epsilon \tilde{D} = 0, \quad (4.2.2)$$

where the variables with tildes are residuals retrieved from taking out the linear effect of  $W$  (practically, via linear regression)

$$\tilde{Y} = Y - \gamma'_{YW} W, \quad \gamma_{YW} = \arg \min_{\gamma \in \mathbb{R}^p} E(Y - \gamma' W)^2,$$

$$\tilde{D} = D - \gamma'_{DW} W, \quad \gamma_{DW} = \arg \min_{\gamma \in \mathbb{R}^p} E(D - \gamma' W)^2.$$

$\alpha$  can then be recovered from population linear regression of  $\tilde{Y}$  on  $\tilde{D}$ :

$$\alpha = \arg \min_{a \in \mathbb{R}} E(\tilde{Y} - a \tilde{D})^2 = (E \tilde{D}^2)^{-1} E \tilde{D} \tilde{Y}.$$

Note also that  $a = \alpha$  solves the moment equation:

$$E(\tilde{Y} - a \tilde{D}) \tilde{D} = 0.$$

We now consider estimation of  $\alpha$  in a high-dimensional setting. For estimation purposes, we maintain that we have a random sample  $(Y_i, X_i)_{i=1}^n$ .

To estimate  $\alpha$ , we will mimic the partialling-out procedure in the population in the sample. In Chapter 1, where  $p/n$  was small, we employed ordinary least squares as the prediction method in the partialling-out steps. We are now considering cases where  $p/n$  is not small, and we instead employ Lasso-based methods in the partialling-out steps.

The estimation procedure for a target parameter  $\alpha$  in a high-dimensional linear model setting can be summarized as follows:

**The Double Lasso procedure:**

1. We run Lasso regressions of  $Y_i$  on  $W_i$  and  $D_i$  on  $W_i$ :

$$\hat{\gamma}_{YW} = \arg \min_{\gamma \in \mathbb{R}^p} \sum_i (Y_i - \gamma' W_i)^2 + \lambda_1 \sum_j \hat{\psi}_j |\gamma_j|,$$

$$\hat{\gamma}_{DW} = \arg \min_{\gamma \in \mathbb{R}^p} \sum_i (D_i - \gamma' W_i)^2 + \lambda_2 \sum_j \hat{\psi}_j |\gamma_j|,$$

and obtain the resulting residuals:

$$\check{Y}_i = Y_i - \hat{\gamma}'_{YW} W_i,$$

$$\check{D}_i = D_i - \hat{\gamma}'_{DW} W_i.$$

In place of Lasso, we can use Post-Lasso or other Lasso relatives (the Dantzig selector, square-root Lasso, and others).

2. We run the least squares regression of  $\check{Y}_i$  on  $\check{D}_i$  to obtain the estimator  $\check{\alpha}$ :

$$\begin{aligned}\check{\alpha} &= \arg \min_{a \in \mathbb{R}} \mathbb{E}_n (\check{Y}_i - a\check{D}_i)^2 \\ &= (\mathbb{E}_n \check{D}_i^2)^{-1} \mathbb{E}_n \check{D}_i \check{Y}_i.\end{aligned}\quad (4.2.3)$$

We can use standard results from this regression, ignoring that the input variables were previously estimated, to perform inference about the predictive effect,  $\alpha$ .

Good performance of the Double Lasso procedure relies on approximate sparsity of the population regression coefficients  $\gamma_{YW}$  and  $\gamma_{DW}$ , with a sufficiently high speed of decrease in the sorted coefficients, namely,

$$|\gamma_{YW}|_{(j)} \leq Aj^{-a}, \quad |\gamma_{DW}|_{(j)} \leq Aj^{-a} \quad a > 1, \quad j = 1, \dots, p.$$

The following theorem can be shown for the Double Lasso procedure:

**Theorem 4.2.1** (Adaptive Inference with Double Lasso in High-Dimensional Regression) *Under the stated approximate sparsity and additional regularity conditions, the estimation error in  $\check{D}_i$  and  $\check{Y}_i$  has no first order effect on  $\check{\alpha}$ , and*

$$\sqrt{n}(\check{\alpha} - \alpha) \approx \sqrt{n} \mathbb{E}_n \tilde{D} \epsilon / \mathbb{E}_n \tilde{D}^2 \stackrel{d}{\sim} N(0, V),$$

where

$$V = (\mathbb{E} \tilde{D}^2)^{-1} \mathbb{E}(\tilde{D}^2 \epsilon^2) (\mathbb{E} \tilde{D}^2)^{-1}.$$

The above statement means that  $\check{\alpha}$  concentrates in a  $\sqrt{V/n}$ -neighborhood of  $\alpha$ , with deviations controlled by the normal law. Observe that the approximate behavior of the double lasso estimator is the same as the approximate behavior of the least

squares estimator in low-dimensional models; see Theorem 1.2.2 in Chapter 1.

Just like in the low-dimensional case, we can use these results to construct a confidence interval for  $\alpha$ . The standard error of  $\check{\alpha}$  is

$$\sqrt{\hat{V}/n},$$

where  $\hat{V}$  is a plug-in estimator of  $V$ . The result implies, for example, that the interval

$$[\check{\alpha} \pm 2\sqrt{\hat{V}/n}]$$

covers  $\alpha$  about 95% of the time.

## Application to Testing the Convergence Hypothesis

We provide an empirical example of partialling-out with Lasso to estimate the regression coefficient  $\alpha$  in the high-dimensional linear regression model:

$$Y = \alpha D + \beta' W + \epsilon.$$

In this example, we are interested in how economic growth rates ( $Y$ ) are related to the initial wealth levels in each country ( $D$ ) controlling for a country's institutional, educational, and other similar characteristics ( $W$ ).

The relationship is captured by  $\alpha$ , the "speed of convergence/-divergence", which predicts the speed at which poor countries catch up ( $\alpha < 0$ ) or fall behind ( $\alpha > 0$ ) rich countries, after controlling for  $W$ . Our inference question here is do poor countries grow faster than rich countries, controlling for educational and other characteristics? In other words, is the speed of convergence negative:  $\alpha < 0$ ? This is the Convergence Hypothesis predicted by the Solow growth model. Under some strong assumptions that we won't state here, the predictive exercise we are doing can be given causal interpretation.

In our data, the outcome ( $Y$ ) is the realized annual growth rate of a country's wealth (Gross Domestic Product per capita). The target regressor ( $D$ ) is the initial level of the country's wealth. The controls ( $W$ ) include measures of education levels, quality of institutions, trade openness, and political stability in the country. The sample contains 90 countries and about 60 controls. Thus  $p \approx 60$ ,  $n = 90$  and  $p/n$  is not small. We expect

Robert M. Solow is a world-renowned MIT economist who won the Nobel Prize in Economics in 1987.

the least squares method to provide a poor/ noisy estimate of  $\alpha$ . We expect the method based on partialling-out with Lasso to provide a high-quality estimate of  $\alpha$ .

	Estimate	Std. Error	95% CI
OLS	-0.009	0.030	[-0.071, 0.052]
Double Lasso	-0.050	0.014	[-0.078, -0.022]

**Table 4.1:** Estimates for the convergence coefficient

Least squares provides a rather noisy estimate of the speed of convergence which does not allow drawing strong conclusions about the convergence hypothesis. For example, the 95% confidence interval is very wide and includes both positive and negative values. In sharp contrast, Double Lasso provides a precise estimate. The lasso-based point estimate is  $-5\%$  and the 95% confidence interval for the (annual) rate of convergence is  $-7.8\%$  to  $-2.2\%$ . This empirical evidence is consistent with the conditional convergence hypothesis.

## Inference on Many Coefficients

If we are interested in more than one coefficient, we can repeat the one-by-one Double Lasso procedure for each of the coefficients of interest and obtain valid estimation and inference on each component under regularity conditions.

Here we consider the model

$$\begin{array}{lcl} \text{Outcome} & = & \sum_{\ell=1}^{p_1} \underbrace{\alpha_\ell D_\ell}_{\text{Target Predictors}} + \sum_{j=1}^{p_2} \underbrace{\beta_j \bar{W}_j}_{\text{Controls}} + \epsilon, \end{array}$$

where the number of interesting predictors  $p_1$  could be very large and the number of controls  $p_2$  could also be very large.

There are at least three motivations for considering many coefficients of interest:

- ▶ there can be multiple policies whose predictive effect we would like to infer;
- ▶ we can be interested in heterogeneous predictive effects across groups;
- ▶ we can be interested in nonlinear effects of policies.

The methodology can be used to study **heterogeneous effects**, where  $D'_\ell$ 's are generated as

$$D_\ell = D_0 \bar{X}_\ell, \quad \ell = 1, \dots, p_1,$$

where  $D_0$  is the base variable of interest (a treatment indicator, price, group indicator), and  $(\bar{X}_l)_{l=1}^{p_1}$  are known transformations of controls  $\bar{W}$ , for example various subgroup indicators.

The methodology can also be used to study **nonlinear effects**. For example, we could consider  $D_\ell$ 's generated as polynomial transformations of the base treatment variable:

$$D_\ell = D_0^\ell, \quad \ell = 1, \dots, p_1.$$

We could also further interact these transformations with other variables to study nonlinear heterogeneous effects.

**One by One Double Lasso for Many Target Parameters.**

For each  $\ell = 1, \dots, p_1$ , we apply the one-by-one Double Lasso procedure for estimation and inference on the coefficient  $\alpha_\ell$  in the model

$$Y = \alpha_\ell D_\ell + \gamma'_\ell W_\ell + \epsilon, \quad W_\ell = ((D_k)'_{k \neq \ell}, \bar{W}')'.$$

Under approximate sparsity conditions, the Double Lasso method provides a high-quality estimate  $\hat{\alpha} = (\hat{\alpha}_\ell)_{\ell=1}^{p_1}$  of  $\alpha = (\alpha_\ell)_{\ell=1}^{p_1}$  and we can construct individual confidence intervals, or even joint confidence bands. Under regularity conditions, this allows for simultaneous inference on  $p_1 > n$  coefficients.

**Theorem 4.2.2** (Double Lasso for Many Coefficients) *Under regularity conditions that impose approximate sparsity in all partialling out steps we have that uniformly in  $\ell = 1, \dots, p_1$ :*

$$\sqrt{n}(\hat{\alpha}_\ell - \alpha_\ell) \approx (\mathbb{E}_n \tilde{D}_\ell^2)^{-1} \sqrt{n} \mathbb{E}_n \tilde{D}_\ell \epsilon,$$

where the approximation holds in the supremum distance. Further, if  $(\log p_1)^5/n$  is small, we have that

$$\sqrt{n}(\hat{\alpha} - \alpha) \stackrel{\text{a}}{\sim} N(0, V),$$

where

$$V_{\ell k} = (\mathbb{E} \tilde{D}_\ell^2)^{-1} \mathbb{E} \tilde{D}_\ell \tilde{D}_k \epsilon^2 (\mathbb{E} \tilde{D}_k^2)^{-1}.$$

Recall that the above distributional approximation formally means that

$$\sup_{A \in \mathcal{A}} \left| \mathbb{P} \left( \sqrt{n}(\hat{\alpha} - \alpha) \in A \right) - \mathbb{P} (N(0, V) \in A) \right| \rightarrow 0,$$

where  $\mathcal{A}$  is a collection of all (hyper) rectangles. The latter result allows the construction of simultaneous confidence bands on all target parameters  $\alpha_\ell$ 's of the form:

$$CR = \times_{\ell=1}^{p_1} \left[ \hat{\alpha}_\ell \pm c \sqrt{\hat{V}_{\ell\ell}/n} \right],$$

with critical value  $c$  chosen so that

$$\begin{aligned} P(\alpha \in CR) &= P\left(\sqrt{n}(\alpha - \hat{\alpha}) \in \sqrt{n}(CR - \hat{\alpha})\right) \\ &= P\left(\sqrt{n}(\alpha - \hat{\alpha}) \in \times_{\ell=1}^{p_1} \left[ \pm c \sqrt{\hat{V}_{\ell\ell}} \right]\right) \approx 1 - a \end{aligned}$$

where  $1 - a$  denotes the confidence level.

**Remark 4.2.1** (Details on critical values\*) It can be shown that an "ideal" choice of  $c$  is

$$c = (1 - a) - \text{quantile of } \|N(0, D^{-1/2} V D^{-1/2})\|_\infty,$$

where  $D = \text{diag}(V)$ .  $c$  can therefore be approximated by simulation, having plugged in  $V = \hat{V}$ . Please see references for more details. Note that  $c$  is generally no smaller than the  $(1 - a/2)$ -quantile of a  $N(0, 1)$ , so the simultaneous confidence bands are always no smaller than the component-wise confidence bands.

## Discovering Heterogeneity in the Wage Pay Gap Analysis

We apply the method of the preceding section to analyze heterogeneity of the wage pay gap using CPS 2012 data. To explore heterogeneity, we interact the female indicator with group indicators capturing marital status, education groups, geographical regions, and a third degree polynomial in experience. Table 4.2 provides estimated coefficients, standard errors, and pointwise p-values. Table 4.3 provides the simultaneous 90% confidence band for the heterogeneous effects.

Among other things, we see that being married and not having a high-school diploma predict the largest drops in wages for women, controlling for all other characteristics. Having a high potential experience also predicts larger drops in wages for women. A more complete analysis and interpretation is left as a homework.

	Estimate.	Std. Error	p-value
female	-0.15	0.05	0.00
female:widowed	0.14	0.09	0.13
female:divorced	0.14	0.02	0.00
female:separated	0.02	0.05	0.66
female:nevermarried	0.19	0.02	0.00
female:hsd08	0.03	0.12	0.82
female:hsd911	-0.12	0.05	0.02
female:hsg	-0.01	0.02	0.50
female:cg	0.01	0.02	0.58
female:ad	-0.03	0.02	0.16
female:mw	-0.00	0.02	0.96
female:so	-0.01	0.02	0.67
female:we	-0.00	0.02	0.84
female:exp1	0.00	0.01	0.53
female:exp2	-0.16	0.05	0.00
female:exp3	0.04	0.01	0.00

**Table 4.2:** Estimates of Heterogeneous Predictive Effects in the CPS 2012 data

	5 %	95 %
female	-0.29	-0.02
female:widowed	-0.12	0.39
female:divorced	0.08	0.20
female:separated	-0.11	0.15
female:nevermarried	0.13	0.24
female:hsd08	-0.35	0.41
female:hsd911	-0.26	0.02
female:hsg	-0.06	0.04
female:cg	-0.04	0.06
female:ad	-0.09	0.03
female:mw	-0.05	0.05
female:so	-0.06	0.04
female:we	-0.06	0.05
female:exp1	-0.02	0.03
female:exp2	-0.28	-0.04
female:exp3	0.02	0.06

**Table 4.3:** Simultaneous 90% Confidence Intervals for the Estimates of Heterogeneous Predictive Effects in the CPS 2012 data.

## 4.3 Why Partialling-out works: Neyman Orthogonality

### Neyman Orthogonality

In the double lasso approach,  $\alpha$  is the target parameter and  $\eta$  are **nuisance** projection **parameters** with true value

$$\eta^0 = (\gamma'_{DW}, \gamma'_{YW})'.$$

As the learned value of  $\alpha$  depends on the values of the nuisance parameters, it is useful to explicitly consider the dependence of  $\alpha$  on the nuisance parameters:

$$\alpha(\eta).$$

The main idea of the double lasso approach is that it provides a procedure for learning the target parameter  $\alpha$  that is first-order insensitive to local perturbations of the nuisance parameters around their true values,  $\eta^0$ :

$$D = \partial_\eta \alpha(\eta^0) = 0. \quad (4.3.1)$$

We will call local insensitivity of target parameters to nuisance parameters as in (4.3.1) Neyman orthogonality.

Neyman orthogonality is important for providing high-quality estimation and inference, especially in high-dimensional settings. In high-dimensional settings, we use regularization procedures to estimate the nuisance parameters as solutions to suitable prediction problems. The use of regularization generally results in bias, and we may heuristically view using regularized estimates of nuisance parameters as plugging in estimates of these parameters that are close to, but not exactly equal to, the true values of the nuisance parameters  $\eta^0$ . Neyman orthogonality, which guarantees that the target parameter is locally insensitive to perturbations of the nuisance parameters around their true values, then ensures that this bias does not transmit to the estimation of the target parameter, at least to the first order.

To explain the claim  $D = 0$ , note that the double lasso exploits the empirical analogue of the following moment condition for estimating  $\alpha$ :<sup>1</sup>

$$M(a, \eta) = E[(\tilde{Y}(\eta_1) - a\tilde{D}(\eta_2))\tilde{D}(\eta_2)] = 0. \quad (4.3.2)$$

1: Indeed, our double Lasso estimator solves the empirical analogue of equation (4.3.2):

$$\hat{M}(a, \hat{\eta}) = \mathbb{E}_n[(\check{Y} - a\check{D})\check{D}] = 0,$$

where  $\check{Y} = \tilde{Y}(\hat{\eta}_1)$ ,  $\check{D} = \tilde{D}(\hat{\eta}_2)$ .

where the notation

$$\tilde{Y}(\eta_1) = Y - \eta'_1 W, \quad \tilde{D}(\eta_2) = D - \eta'_2 W$$

emphasizes dependence on the nuisance parameters.

Here the true parameter value  $\alpha = \alpha$  solves this equation when

$$\eta := (\eta'_1, \eta'_2)' = \eta^o := (\gamma'_{DW}, \gamma'_{YW})'.$$

Here the true residuals correspond to  $\eta := \eta^o$

$$\tilde{Y} = Y - \gamma'_{YW} W, \quad \tilde{D} = D - \gamma'_{DW} W.$$

By the **implicit function theorem**:

$$D = -\partial_\alpha M(\alpha, \eta^o)^{-1} \partial_\eta M(\alpha, \eta^o),$$

and

$$\partial_\eta M(\alpha, \eta^o)$$

consists of two components

$$\partial_{\eta_1} M(\alpha, \eta^o) = E[W \tilde{D}] = 0$$

and

$$\partial_{\eta_2} M(\alpha, \eta^o) = -E[W \tilde{Y}] + 2E[\alpha W \tilde{D}] = 0.$$

We summarize the discussion as follows:

**Neyman Orthogonality.** The parameter of interest  $\alpha$  that depends on nuisance parameters  $\eta$  with true value  $\eta^o$  is Neyman Orthogonal with respect to these parameters if

$$D = \partial_\eta \alpha(\eta^o) = 0.$$

If the parameter  $\alpha$  is defined as a root in  $\alpha$  of the equation  $M(\alpha, \eta) = 0$ , which depends on the nuisance parameters  $\eta$  with true value  $\eta^o$ , then the equation is Neyman orthogonal if

$$\partial_\eta M(\alpha, \eta^o) = 0.$$

The principle is applicable to problems outside the high-dimensional linear model problem considered in this chapter.

## What happens if we don't have Neyman Orthogonality?

If we don't have Neyman orthogonality, we should not expect to get high-quality estimates of the target parameters. For example, a seemingly sensible approach that has been considered for statistical inference in the high-dimensional linear model context is as follows:

**(Invalid) Single Selection/Naive Method.**

In this invalid method, one applies Lasso regression of  $Y$  on  $D$  and  $W$  to select relevant covariates  $W_Y$ , in addition to the covariate of interest, then refits the model by least squares of  $Y$  on  $D$  and  $W_Y$ . Inference for the target parameter is then carried out using conventional inference based on the latter regression.

Despite its simplicity and seeming intuitive appeal, the approach outlined above is not a valid approach if the goal is to perform inference on  $\alpha$ . It is a fine approach if the goal is solely prediction of the outcome, but it can result in very misleading conclusions about the parameter of interest  $\alpha$ , as we demonstrate in Example 4.3.1 below.

The naive approach relies on the moment condition

$$M(a, b) = E[(Y - aD - b'X)D] = 0.$$

This moment condition is satisfied by the true value,  $a = \alpha$  when  $b = \beta$ , where it coincides with the classical moment condition for  $\alpha$  underlying low-dimensional ordinary least squares which sets prediction errors to be orthogonal to each predictor variable.

However, this moment condition does not exhibit Neyman Orthogonality since

$$\partial_b M(\alpha, \beta) = E[DX] \neq 0$$

unless  $D$  is orthogonal to  $X$ .<sup>2</sup> Because this moment is not Neyman orthogonal, we would expect that the bias and the relatively slow rate

$$\sqrt{s \log(p \vee n)/n}$$

of convergence of our estimate of  $\beta'X$  would transmit to bias and slow convergence in estimates of  $\alpha$  provided by solving

2: In pure RCTs,  $D$ 's are orthogonal to  $X$ , after de-meaning  $D$ , so Neyman Orthogonality would hold in this setting.

the empirical analog of the naive moment condition. The naive procedure outlined above exactly provides the solution to this moment condition. Consequently, while this naive procedure provides an estimator of  $\alpha$  that will approach the true value in large samples (at a slower than  $\sqrt{n}$ -rate), the bias of the estimator converges too slowly for standard inference methods to provide reliable inference.

We can set up a simulation experiment to verify that this naive approach provides low quality estimates for  $\alpha$ .

**Example 4.3.1** We compare the performance of the naive and orthogonal methods in a computational experiment where

$$p = n = 100,$$

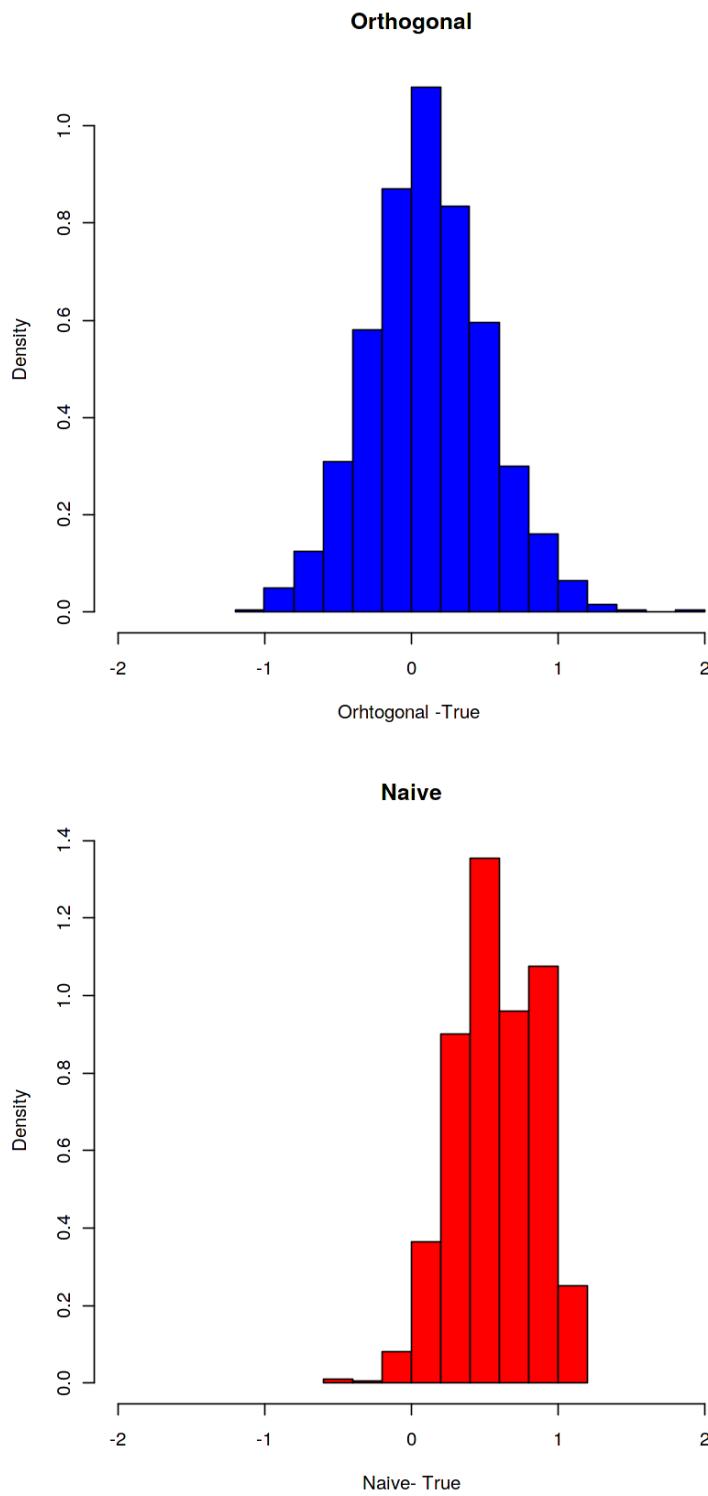
$$\beta_j = 1/j^2, (\gamma_{DW})_j = 1/j^2, \text{ and}$$

$$Y = 1 \cdot D + \beta' W + \varepsilon_Y, \quad W \sim N(0, I), \quad \varepsilon_Y \sim N(0, 1)$$

$$D = \gamma'_{DW} W + \tilde{D}, \quad \tilde{D} \sim N(0, 1)/4.$$

From the histograms shown in Figure 4.1, we see that the naive estimator is heavily biased, as expected from the lack of Neyman orthogonality in its estimation strategy, while the orthogonal estimator, based on partialling out, is approximately unbiased and Gaussian.

The reason that the naive estimator does not perform well is that it only selects controls that are strong predictors of the outcome, thereby omitting weak predictors of the outcome. However, weak predictors of the outcome could still be strong predictors of  $D$ , in which case dropping these controls results in a strong omitted variable bias. In contrast, the orthogonal approach solves two prediction problems – one to predict  $Y$  and another to predict  $D$  – and finds controls that are relevant for either. The resulting residuals are therefore approximately “de-confounded”.



**Figure 4.1:** **Top Panel:** Simulated distribution of the orthogonal estimator centered around the true value. **Bottom Panel:** Simulated distribution of the naive (single-selection) non-orthogonal estimator centered around the true value.

## 4.4 Other Approaches that have the Neyman Orthogonality Property

### Double Selection

One way to fix "single selection" would be to have "double selection":

#### Double Selection

- ▶ find controls  $W_Y$  that predict  $Y$  as judged by lasso;
- ▶ find controls  $W_D$  that predict  $D$  as judged by lasso;
- ▶ regress  $Y$  on  $D$  and the union of controls  $W_Y \cup W_D$ ; proceed with standard inference.

This procedure is approximately equivalent to the partialling out approach, and therefore inherits the orthogonality property. This approach is more conservative compared to single selection, as it makes sure that we have not omitted controls that are strong confounders for  $D$ . It therefore guards against large omitted variable biases.

### Desparsified Lasso

Yet another procedure that has the orthogonality property and is approximately equivalent to the partialling out approach under suitable conditions is desparsified lasso.

This approach uses the fact that  $\alpha = \alpha$  solves the equation:

$$M(a, \eta) = E[(Y - aD - b'W)\tilde{D}(\gamma)] = 0,$$

when  $\eta = (b', \gamma')' = \eta^o := (\beta', \gamma'_{DX})'$ , where

$$\tilde{D}(\gamma) = D - \gamma'W.$$

One can verify that

$$\alpha(\eta) = (ED\tilde{D}(\gamma))^{-1}E((Y - b'W)\tilde{D}(\gamma)),$$

and that

$$\alpha = \alpha(\eta^o).$$

Further, the moment condition has the orthogonality property, verification of which is left to the reader, which implies that

$$\partial_\eta \alpha(\eta^o) = 0,$$

similarly to the previous argument.

### Desparsified Lasso

- ▶ Run a lasso estimator of  $Y$  on  $D$  and  $W$ , and save the coefficient estimate  $\hat{\beta}$ .
- ▶ Run a lasso estimator of  $D$  on  $W$  and save the coefficient estimate  $\hat{\gamma}$ .
- ▶ The estimator  $\hat{\alpha}$  is then the solution of the empirical analog of the moment condition above:

$$\mathbb{E}_n[(Y - \hat{\alpha}D - \hat{\beta}'W)\tilde{D}(\hat{\gamma})] = 0,$$

which has the explicit form:

$$\hat{\alpha} = (\mathbb{E}_n D\tilde{D}(\hat{\gamma}))^{-1} \mathbb{E}((Y - \hat{\beta}'W)\tilde{D}(\hat{\gamma})),$$

where  $\hat{\beta}$  and  $\hat{\gamma}$  are lasso estimators.

Estimators of this form are referred to in econometrics as "instrumental variable estimators". In purely technical terms we are using residualized  $\tilde{D}$  to "instrument" for  $D$ .

## Notebooks

- ▶ [R Notebook with Experiment on Orthogonal vs Non-Orthogonal Learning](#) presents the simulation experiment comparing orthogonal (partialling-out) with non-orthogonal learning (naive method).
- ▶ [R Notebook on double lasso for Growth Convergence](#) presents a double lasso analysis of the conditional convergence hypothesis in growth economics.
- ▶ [R Notebook on double lasso for the Heterogeneous Gender Pay Gap](#) presents a double lasso analysis of the heterogenous gender pay gap.

## Notes

We mainly follow the double lasso approach developed in [1] and [2], because it is nicely connected to partialling out and will later generalize seamlessly to double machine learning [3]. Desparsified lasso was developed by [4] and [5]; a closely

related approach is debiased lasso proposed by [6]. The double selection method was developed by [7] and [8]. Inference on many coefficients using double lasso was first developed by [9] and [10]. The double lasso and desparsified lasso approaches have been extended to time series and many time series by [11], with the corresponding package available at this [Link](#). The approach takes into account the temporal dependencies in the data when fitting lasso and performing inference on the coefficients of interest.

Failure of single selection even when  $p$  is small is discussed in simple terms in [8], but the problem was first systematically examined by [12]. The package `hdm` [13] in R implements the main methods discussed here. Another R package called `hid` [14] implements the approach by [4] and [5]. A recent paper [15] develops debiasing methods for shape constrained high-dimensional linear regression models.

## Study Problems

1. Experiment with the first notebook, [R Notebook with Experiment on Orthogonal vs Non-Orthogonal Learning](#). Try different models. For example, try different coefficient structures for  $\beta$  and  $\gamma_{DW}$  and/or different covariance structures for  $W$ . Provide an explanation to a friend for what each step in the Double Lasso procedure is doing.
2. Use the second notebook, [R Notebook on double lasso for Growth Convergence](#). Provide an explanation to a friend for what each step in the Double Lasso procedure is doing. Explain the empirical results to a friend. Experiment with making the set of controls more flexible and higher-dimensional by adding nonlinear and/or interaction terms that seem potentially interesting. Comment on how the results differ from the baseline results.
3. Experiment with the third notebook, [R Notebook on double lasso for the Heterogeneous Gender Pay Gap](#). Provide an explanation to a friend for what each step in the inference procedure is doing. Explain the empirical results to a friend.
4. Verify that Neyman orthogonality holds for the "desparsified" lasso strategy.

# Bibliography

- [1] Victor Chernozhukov, Christian Hansen, and Martin Spindler. 'Valid post-selection and post-regularization inference: An elementary, general approach'. In: *Annu. Rev. Econ.* 7.1 (2015), pp. 649–688 (cited on page 90).
- [2] Alexandre Belloni, Victor Chernozhukov, and Lie Wang. 'Pivotal estimation via square-root lasso in nonparametric regression'. In: *Annals of Statistics (arXiv 2011)* 42.2 (2014), pp. 757–788 (cited on page 90).
- [3] Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. 'Double/debiased machine learning for treatment and structural parameters'. In: *The Econometrics Journal* 21.1 (2018), pp. C1–C68 (cited on page 90).
- [4] Cun-Hui Zhang and Stephanie S. Zhang. 'Confidence intervals for low dimensional parameters in high dimensional linear models'. In: *Journal of the Royal Statistical Society: Series B: Statistical Methodology* 76.1 (2014), pp. 217–242 (cited on pages 90, 91).
- [5] Sara Van de Geer, Peter Bühlmann, Ya'acov Ritov, and Ruben Dezeure. 'On asymptotically optimal confidence regions and tests for high-dimensional models'. In: *Annals of Statistics* 42.3 (2014), pp. 1166–1202 (cited on pages 90, 91).
- [6] Adel Javanmard and Andrea Montanari. 'Confidence intervals and hypothesis testing for high-dimensional regression'. In: *The Journal of Machine Learning Research* 15.1 (2014), pp. 2869–2909 (cited on page 91).
- [7] Alexandre Belloni, Victor Chernozhukov, and Christian B. Hansen. 'Inference for High-Dimensional Sparse Econometric Models'. In: *Advances in Economics and Econometrics: Tenth World Congress*. Ed. by Daron Acemoglu, Manuel Arellano, and Eddie Editors Dekel. Vol. 3. Econometric Society Monographs. Cambridge University Press, 2013, pp. 245–295. doi: [10.1017/CBO9781139060035.008](https://doi.org/10.1017/CBO9781139060035.008) (cited on page 91).
- [8] Alexandre Belloni, Victor Chernozhukov, and Christian Hansen. 'Inference on Treatment Effects After Selection Amongst High-Dimensional Controls'. In: *Review of Economic Studies* 81.2 (2014), pp. 608–650 (cited on page 91).

- [9] Alexandre Belloni, Victor Chernozhukov, and Kengo Kato. ‘Uniform post-selection inference for least absolute deviation regression and other Z-estimation problems’. In: *Biometrika* 102.1 (2015), pp. 77–94 (cited on page 91).
- [10] Alexandre Belloni, Victor Chernozhukov, Denis Chetverikov, and Ying Wei. ‘Uniformly valid post-regularization confidence regions for many functional parameters in z-estimation framework’. In: *Annals of statistics* 46.6B (2018), p. 3643 (cited on page 91).
- [11] Victor Chernozhukov, Wolfgang Karl Härdle, Chen Huang, and Weining Wang. ‘Lasso-driven inference in time and space’. In: *The Annals of Statistics* 49.3 (2021), pp. 1702–1735 (cited on page 91).
- [12] Hannes Leeb and Benedikt M. Pötscher. ‘Model selection and inference: Facts and fiction’. In: *Econometric Theory* 21.1 (2005), pp. 21–59 (cited on page 91).
- [13] Victor Chernozhukov, Chris Hansen, and Martin Spindler. ‘hdm: High-dimensional metrics’. In: *arXiv preprint arXiv:1608.00354* (2016) (cited on page 91).
- [14] Ruben Dezeure, Peter Bühlmann, Lukas Meier, and Nicolai Meinshausen. ‘High-dimensional inference: confidence intervals, p-values and R-software hdi’. In: *Statistical science* 30.4 (2015), pp. 533–558 (cited on page 91).
- [15] Yufei Yi and Matey Neykov. ‘A New Perspective on Debiasing Linear Regressions’. In: *arXiv preprint arXiv:2104.03464* (2021) (cited on page 91).

# Causal Inference via Conditional Exogeneity

5

Here we discuss how average causal effects may be identified using regression when treatment is not randomly assigned but instead depends on observed covariates. We discuss the regression method, where we compute the average difference between expected outcomes for treated and untreated units that are comparable (formally, identical) in terms of their characteristics  $X$ . If treatment is as good as randomly assigned conditional on  $X$ , then this approach recovers average causal or treatment effects. This key condition is commonly referred to as conditional ignorability or conditional exogeneity.

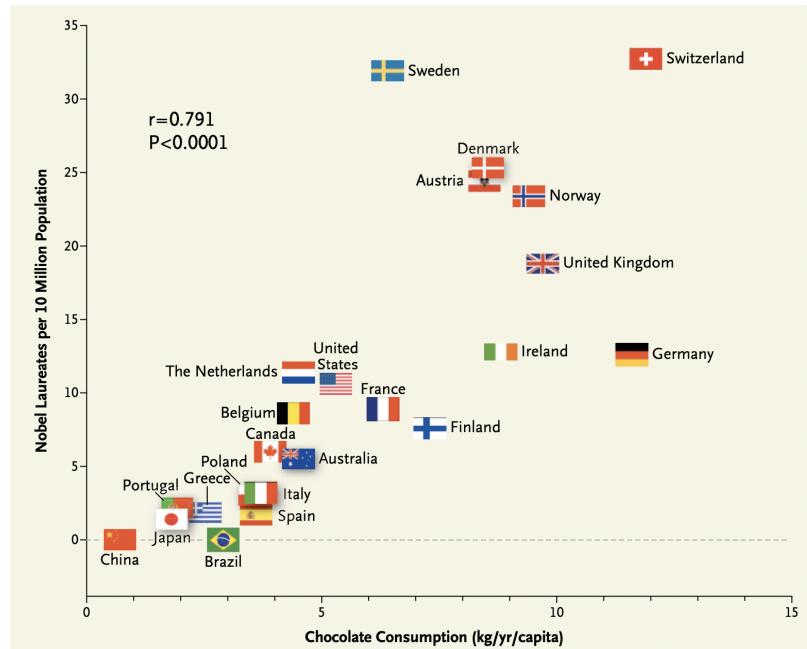
"compare apples and/to/with apples: to compare things that are very similar". Merriam Webster.

"magic: the use of means (such as charms or spells) believed to have supernatural power over natural forces". Merriam Webster.

5.1 Introduction . . . . .	95
5.2 Potential Outcomes Framework and Ignorability . . . . .	96
Identification by Conditioning . . . . .	96
Conditional Ignorability via Causal Diagrams . . . . .	99
5.3 Identification Using Propensity Scores . . . . .	100
Conditioning On Propensity Scores . . . . .	100
Identification by Propensity Score Reweighting . . . . .	100
Stratified RCTs . . . . .	101
Covariate Balance Checks . . . . .	102
5.4 Average Treatment Effect for Groups and on the Treated* . . . . .	102
5.5 Connection to Linear Regression . . . . .	103
What if the propensity score is known? . . . . .	105
5.A Rosenbaum-Rubin's result . . . . .	106
5.B Details of ATET . . . . .	106

## 5.1 Introduction

In a cross-country analysis, higher chocolate consumption predicts a higher number of Nobel laureates per capita.

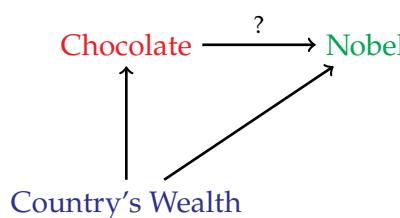


**Figure 5.1:** Source: Franz H. Messerli, "Chocolate Consumption, Cognitive Function, and Nobel Laureates", New England Journal of Medicine. 2012

Is this a reflection of a true causal effect and therefore an actionable insight? If it were, countries could generate more Nobel laureates per capita by making chocolate abundant to everyone. (This wouldn't be a bad thing.) Is this perhaps what Switzerland did? Switzerland has the highest number of Nobel laureates per capita.

Or is there a common cause<sup>1</sup> that creates non-causal association – perhaps wealthy countries invest more in science and higher wealth causes people to consume luxury goods like chocolate? Comparative analysis, where we compare nations with identical or similar wealth, would probably reveal that the correlation is not causal.<sup>2</sup> Probably we should be comparing Switzerland to similar countries in terms of wealth – the “apples-to-apples” comparison, so to speak. This type of analysis is very common in

- 1: We often refer to these common causes as “omitted variables” that give rise to “omitted variable bias”.
- 2: It remains a fundamental empirical problem to confirm this conjecture or disprove this conjecture. The causal channel through which chocolate (and other flavonoids) may affect Nobel production is by documented improvement in the cognitive function.



**Figure 5.2:** A Contrived Causal Path Diagram for the Effect of Country's Wealth on Chocolate Consumption and Nobel Prize Production per capita.

causal inference and is implemented via a set of tools introduced in this chapter.

In what follows, we work within Rubin's [1] potential outcomes framework, as introduced in Chapter 2. The idea is that if we can think of observed treatment  $D$  as generated randomly – independently of potential outcomes – conditional on some pre-treatment variables  $X$ , then we can learn the average causal (treatment) effects by regression

of  $Y$  on  $D$  and  $X$ ,

or, as is often said, by "adjusting" or "controlling" for  $X$ .

## Notation

Recall that we denote the independence of two random variables (these can include random vectors)  $U$  and  $V$  as<sup>3</sup>

$$U \perp\!\!\!\perp V.$$

Independence, conditional on a third variable  $X$ , is denoted by

$$U \perp\!\!\!\perp V | X.$$

<sup>3</sup>: Note that this notation  $\perp\!\!\!\perp$  is different from the notation  $\perp$  that is used to denote orthogonality (uncorrelatedness of a centered random variable with another). If  $U$  is centered, then  $U \perp\!\!\!\perp V$  implies  $U \perp V$ , but the reverse implication is not true in general.

## 5.2 Potential Outcomes Framework and Ignorability

### Identification by Conditioning

Recall that we use  $Y(d)$  to denote potential outcome in the treatment state  $d$ . We also recall our example of smoking from Chapter 2. Suppose we want to study the impact of smoking marijuana on life longevity. Suppose that smoking marijuana has no causal/treatment effect on life longevity:

$$Y = Y(0) = Y(1), \text{ so that } \delta = EY(1) - EY(0) = 0.$$

However, the observed smoking behavior,  $D$ , results not from an experimental study, but from observational data in which an individual's smoking behavior is associated with poor health choices  $X$  (drinking alcohol for example) which cause shorter life longevity. In this case, the predictive effect recovered by

regression without adjusting for  $X$  does not match the average causal effect

$$E[Y | D = 1] - E[Y | D = 0] < 0 = \delta,$$

because higher  $D$  predicts higher  $X$ , which predicts lower  $Y$ . This difference between the predictive effect and average causal effect is the result of confounding or **selection bias**.

In this example, conditioning on  $X$  can remove the selection bias

$$E[E[Y | D = 1, X] - E[Y | D = 0, X]] = \delta.$$

provided that conditional on  $X$  variation in  $D$  is independent of the potential health outcomes.

The following provides a formal assumption under which we can eliminate the confounding bias by controlling for  $X$ .<sup>4</sup>

**Assumption 5.2.1** (Conditional Ignorability and Consistency). *Ignorability: Suppose that treatment status  $D$  is independent of potential outcomes  $Y(d)$  conditional on a set of covariates  $X$ ; that is, for each  $d$ :*

$$D \perp\!\!\!\perp Y(d) | X.$$

*Consistency: Suppose that  $Y$  is generated as  $Y := Y(D)$ .*

The ignorability assumption<sup>5</sup> says that variation in treatment assignment  $D$  is as good as random conditional on  $X$ .<sup>6</sup> This assumption means that if we look at units with the same value of the covariates, e.g. units with  $X = x$ , then treatment variation among these observationally identical units,  $D | X = x$ , is indeed produced as if by a formal randomized control trial.

Therefore, we can learn about the causal effect of  $D$  by comparing outcomes across treated and control units who have identical characteristics  $X = x$  under the conditional ignorability assumption. The idea of comparing observations who have identical characteristics is the essence of the so-called **conditioning or adjustment** strategy to learning causal effects. As conditioning approaches produce a different contrast for every potential value of  $X$ , we may also wish to average the contrasts at different values of  $X$  over the distribution of characteristics to produce a summary measure of the causal effects.

The conditional probability of receiving treatment, *the propensity score*, plays an important role in this approach.

4: The assumption is untestable, but delivers powerful results. Cf. definition of magic. Given scientific contexts encoded in causal DAGs, we study a systematic way of finding  $X$  in subsequent chapters.

5: You may wonder why the term "ignorability" is used. The distribution of  $Y(d)$  depends only on  $X$  and not on  $D$ , so the latter is "ignorable".

6: Note that the conventional name used in econometrics for the ignorability assumption is the *conditional exogeneity* or *conditional independence* assumption. Since we emphasize potential outcomes as a framework to think of causality here, we use the naming conventions of the potential outcomes literature.

**Assumption 5.2.2** (Overlap/Full Support). *The probability of receiving treatment given  $X$ , the propensity score*

$$p(X) := P(D = 1|X),$$

*is non-degenerate:*

$$P(0 < p(X) < 1) = 1.$$

The overlap assumption requires that there is proper randomization or variation in  $D$  at each value  $x$  in the support of  $X$ . Without this condition, there are values  $x$  in the support of  $X$  where we cannot construct a contrast between treatment and control units. We cannot learn the conditional average treatment effect at these values of  $X$  and thus are also unable to learn the unconditional average effect of the treatment.

**Remark 5.2.1** Assumption 5.2.2 is also often called the **full support** condition because it requires

$$\text{support}(D, X) = \{0, 1\} \times \text{support}(X).$$

The following is the most important theoretical result that states that we can recover expectations of potential outcomes from regressions.

**Theorem 5.2.1** (Conditioning on  $X$  Removes Selection Bias)

*Under Conditional Ignorability and Overlap, the conditional expectation function of observed outcome  $Y$  given  $D = d$  and  $X$  recovers the conditional expectation of the potential outcome  $Y(d)$  given  $X$ :*

$$E[Y | D = d, X] = E[Y(d) | D = d, X] = E[Y(d) | X].$$

To prove Theorem 5.2.1, note that the overlap assumption makes it possible to condition on the events  $\{D = 0, X\}$  and  $\{D = 1, X\}$  at any value in the support of  $X$  and that the second equality holds by ignorability.

Hence, the Conditional Average Predictive Effect (CAPE),

$$\pi(X) = E[Y | D = 1, X] - E[Y | D = 0, X],$$

is equal to the Conditional Average Treatment Effect (CATE),

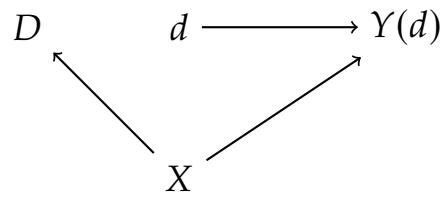
$$\delta(X) = E[Y(1) | X] - E[Y(0) | X].$$

Thus, the APE and ATE also agree:

$$\delta = E\delta(X) = E\pi(X) = \pi.$$

## Conditional Ignorability via Causal Diagrams

It is possible to illustrate the previous set-up and assumptions graphically as follows:<sup>7</sup>

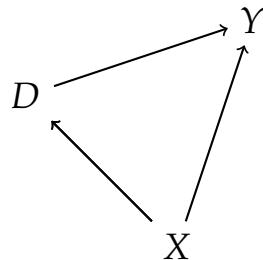


7: Note that what we present is just one of many causal diagrams that are compatible with the conditional ignorability condition. There are others, as will become apparent in subsequent chapters.

**Figure 5.3:** A Causal Diagram for the Conditional Ignorability Research Design

In this graph, we show the potential outcome  $Y(d)$  as a node and the potential treatment status  $d$  as another node. The latter node is deterministic. There is an arrow from  $d$  to  $Y(d)$  indicating the dependency. The pre-treatment covariates  $X$  affect both the realized treatment variable  $D$  and the potential outcomes  $Y(d)$ , as shown by the arrow from  $X$  to  $D$  and from  $X$  to  $Y(d)$ . The assigned treatment variable  $D$  is independent of the node  $d \mapsto Y(d)$ , conditional on  $X$ , which is shown by the absence of the arrow between these two nodes.

The potential outcome process  $d \mapsto Y(d)$  and treatment assignment jointly determine the realized outcome variable  $Y$  via the assignment  $Y := Y(D)$ . This generates the following causal diagram. This graph says that  $X$  is generated first.  $D$  is then



**Figure 5.4:** A Causal Diagram with Conditional Ignorability

generated, with the distribution of  $D$  depending on  $X$ . Finally,  $Y$  is generated, with its distribution depending on both  $D$  and  $X$ . Here, after conditioning on  $X$ , the statistical dependence

(association) between  $D$  and  $Y$  only reflects the causal channel,  $D \rightarrow Y$  allowing us to uncover the ATE, for example.

## 5.3 Identification Using Propensity Scores

### Conditioning On Propensity Scores

The fact that conditioning on the right set of controls removes selection bias has long been recognized by researchers employing regression methods. Rosenbaum and Rubin [2] made the much more subtle point that conditioning on only the propensity score

$$p(X) = P[D = 1 | X]$$

also suffices to remove the selection bias.

**Theorem 5.3.1** (Rosenbaum and Rubin: Conditioning on the Propensity Score Removes Selection Bias) *Under Ignorability and Overlap,  $D$  is generated independently of  $Y(d)$  for each  $d$ , conditional on the propensity score  $p(X)$ : For each  $d$ ,*

$$D \perp\!\!\!\perp Y(d) | p(X).$$

In other words, conditional on  $p(X) = p$ , variation in  $D$  is as good as randomly assigned. Hence, whenever it suffices to use  $X$  for identification by conditioning, it also suffices to use  $p(X)$ . This fact makes  $p(X)$  a “minimal sufficient” statistic, conditioning on which removes selection bias under ignorability.

Conditioning on the propensity score rather than  $X$  itself is a useful empirical strategy when  $X$  is high dimensional and  $p(X)$  is available (in stratified RCTs) or can be approximated accurately.<sup>8</sup> In such scenarios, we can simply use  $p(X)$  as a control in place of the high dimensional set of characteristics,  $X$ , and thus bypass a potentially complicated high dimensional estimation problem. After controlling for  $p(X)$ , we can also consider the use of high dimensional methods for further using  $X$  in order to improve precision.

### Identification by Propensity Score Reweighting

An alternative approach, known as the Horvitz-Thompson method [4], uses propensity score reweighting to recover averages of potential outcomes.

8: An interesting example where the propensity score is not known but can be well-approximated is the examination in [3] of the causal effect of attendance at a particular school or group of schools relative to one or more alternative schools (e.g., “elite” vs. “non-elite” schools) in settings where matching algorithms are used to assign students to schools. In this example, we can think of these student assignment mechanisms as  $p(X)$ .

**Theorem 5.3.2** (Horvitz-Thompson: Propensity Score Reweighting Removes Bias) *Under Conditional Ignorability and Overlap, the conditional expectation of an appropriately reweighed observed outcome  $Y$ , given  $X$ , identifies the conditional average of potential outcome  $Y(d)$  given  $X$ :*

$$E[Y_1(D = d)/P(D = d|X) | X] = E[Y(d) | X]$$

*Then, averaging over  $X$  identifies the average potential outcome:*

$$E[Y_1(D = d)/P(D = d|X)] = E[Y(d)]$$

To prove this result, note

$$\begin{aligned} & E[Y_1(D = d) | X]/P(D = d|X) \\ &= E[Y(d) | X]E[1(D = d) | X]/P(D = d|X) \\ &= E[Y(d) | X], \end{aligned}$$

where we used the conditional ignorability in the second equality.

As a consequence, we can identify average treatment effects by simple averaging of transformed outcomes:

$$\delta = E[YH], \quad H = \frac{1(D = 1)}{P(D = 1|X)} - \frac{1(D = 0)}{P(D = 0|X)},$$

where  $H$  is called the Horvitz-Thompson transform.

Note that this reduces to the difference of means in the control and treatment groups when the propensity score is constant.

## Stratified RCTs

In the case where the propensity score  $p(X)$  is known, we are essentially back to a classical RCT.

**Definition 5.3.1** (Generalized/Stratified RCT) *If under Assumption 5.2.1, the propensity score  $p(X)$  is known, the setting is called a generalized or stratified RCT.*

**Remark 5.3.1** Propensity score reweighting is generally not the most efficient approach to estimating treatment effects from statistical point of view because it ignores any depen-

dence between the outcomes and controls,  $X$ , that is not captured by the propensity score. By exploiting dependence between the outcomes and  $X$  not captured by the propensity score, more efficient estimation of treatment can occur as using this dependence "de-noises" the outcome. Moreover, estimation based on only propensity score reweighting fails under imbalances that might arise due to imperfect data collection. Later, we will use **both** regression and reweighting as part of "double machine learning" to operationalize efficient statistical inference on treatment effects in fully nonlinear (nonparametric) models.

## Covariate Balance Checks

Given a propensity score  $p(X)$ , which is available in the stratified RCT settings outlined above, we can check if the RCT is valid (randomization is successful) by performing a **covariate balance check**:

$$E[H | X] = 0.$$

In a linear framework, this check can be done by running a regression of  $H$  on  $W$ , a dictionary of transformations of  $X$ , and testing if  $W$  predicts  $H$ .  $W$  predicting  $H$  suggests that the RCTs randomization protocol did not go as planned.

One approach to deal with the failure of the robustness check is to control for  $X$  by including  $W$  in addition to using  $p(X)$ . Including  $W$  can reduce the selection bias (and, hopefully, set it equal to zero). In the reemployment experiment, for example, we observed that the balance did not seem satisfied across age groups; specifically, we found more younger workers in the control group. Hence, further controlling for age makes sense and results in modest changes to estimates of the treatment effect. Of course, there is no guarantee that controlling for observed covariates can overcome selection bias in compromised RCTs in general because unobserved covariates may be driving the bias.

## 5.4 Average Treatment Effect for Groups and on the Treated\*

In addition to unconditional average treatment effects (ATE) or average treatment effects at specific values of the covariates

$X = x$ , we may be interested in average effects within specific subpopulations.

A leading example of an interesting subpopulation treatment effect is a group ATE (GATE):

$$\bar{\delta} = E[Y(1) - Y(0)|G = 1]$$

where  $G$  is a group indicator defined in terms of  $X$ 's. For example, we might be interested in the effects of a training program among younger people, say between 18 and 30 years old ( $G = 1(18 \leq age \leq 30)$ ); among people older than 30 years old (so  $G = 1(30 < age)$ ); and differences between these two groups.

We can immediately obtain the GATE using the identification results above and law of iterated expectations:

$$\begin{aligned} E[Y(1) - Y(0)|G = 1] &= E[E[Y|D = 1, X] - E[Y|D = 0, X]|G = 1] \\ &= E[HY|G = 1]. \end{aligned}$$

That is, we can identify GATEs either by taking the difference in regression functions or applying propensity score reweighting of outcomes and then averaging over group  $G$ .

We next consider treatment effects for the subpopulation of treated units, the *average treatment effect on the treated* (ATET):<sup>9</sup>

$$\delta_1 = E(Y(1) - Y(0) | D = 1).$$

9: Rather than ATET, some use the abbreviation AToT or ATT.

For example, consider training completion as a treatment,  $D$ , and  $X$  a vector of pre-treatment variables such that unconfoundedness holds. Consider the question:

- ▶ How much more do trainees earn on average after going through the training program?

The ATET,  $\delta_1$ , is the parameter that answers such questions about counterfactuals. The ATET is identified by

$$E[E[Y|D = 1, X] - E[Y|D = 0, X] | D = 1]$$

similarly to what we had above, but it is possible to bypass the use of  $E[Y|D = 1, X]$ ; see the Appendix for more details.

## 5.5 Connection to Linear Regression

The tools from Chapter 1 and Chapter 4 can be used to perform statistical inference on ATEs. We briefly discuss how (high

dimensional) regression can be used to retrieve causal estimates when conditional ignorability holds in this section.

The simplest instance of the problem is when the conditional expectation function of  $Y$  given  $D$  and  $X$  is linear,

$$E[Y | D, X] = \alpha D + \beta' W,$$

which gives a model

$$Y = \alpha D + \beta' W + \epsilon, \quad E[\epsilon | D, X] = 0.$$

Here it is understood that  $W$  may include  $X$  as well as pre-specified nonlinear transformations of  $X$ .

In this model,  $\alpha$  identifies  $\delta$

$$\delta = \alpha$$

under the linearity assumption and ignorability, and our inference tools for  $\alpha$  automatically carry over to  $\delta$ .

Of course, the assumption of linearity is restrictive. A simple way to relax this is to consider interactions. One version of this approach takes all interactions and assumes

$$E[Y | D, X] = \alpha_1 D + \alpha_2 D W + \beta_1 + \beta' W,$$

where we also maintain that we are working with centered covariates:  $EW = 0$ .<sup>10</sup>

We then recover the ATE as

$$\delta = \alpha_1$$

and CATE as

$$\delta(X) = \alpha_1 + \alpha'_2 W.$$

<sup>10</sup>: This model is still linear and results for linear models carry over to this case as well.

We can use partialling out methods, such as OLS in low-dimensional case and Double Lasso (and variants) in high-dimensional case, to perform inference on  $\alpha_1$  and components of  $\alpha_2$ , or even  $\beta_1$ .

Note, this is the approach we illustrated in the heterogeneous pay gap example in Chapter 1. The discussion of whether the pay gap analysis has a causal interpretation is given in the next causal inference chapter, Chapter 6.

What about fully nonlinear strategies? We will explore them in Chapter 9.

## What if the propensity score is known?

When the propensity score  $p(X)$  is known, we can include it as part of  $W$  in the formulations from the previous section. We can also incorporate polynomials or other transformations of  $p(X)$  to make things more flexible. Finally, we can also employ nonlinear machine learning methods discussed in the sequel, to overcome the limitations of the linear models.

Another useful strategy is to consider regression models where  $HY$  is the dependent variable:<sup>11</sup>

11: See, e.g., [5].

$$E[HY | H, X] = \alpha_1 + \alpha'_2 W + \beta_1 H + \beta'_2 HW.$$

In this regression model, we recover the ATE as

$$\delta = \alpha_1$$

and CATE as

$$\delta(X) = \alpha_1 + \alpha'_2 W.$$

We can use partialling out methods, such as Double Lasso, to perform inference on  $\alpha_1$  and components of  $\alpha_2$ . We also discuss inference on CATE using more general machine learning methods in Chapter 13.

## Study Problems

1. Use one or two paragraphs to explain conditioning and its use in learning treatment effects/causal effects in observational data and randomized trials where treatment probability depends on pre-treatment variables. This discussion should be non-technical as if you were writing an explanation for a smart friend with relatively little exposure to causal modeling.
2. Use one or two paragraphs to explain the propensity score reweighting approach for identification of average treatment effects. This discussion should be non-technical as if you were writing an explanation for a smart friend with relatively little exposure to causal modeling.

3. Use one or two paragraphs to explain why group ATE and the ATE on the treated may be of interest in empirical work. This discussion should be non-technical as if you were writing an explanation for a smart friend with relatively little exposure to causal modeling.

## 5.A Rosenbaum-Rubin's result

We defined above the propensity score as

$$p(X) := P(D = 1|X),$$

which is the probability of receiving treatment given  $X$ . Under conditional ignorability, we can represent the treatment selection rule statistically as

$$D = 1\{U \leq p(X)\}, \quad U \perp\!\!\!\perp X, \quad U \sim U(0, 1),$$

and

$$U \perp\!\!\!\perp Y(d) | X.$$

Using this independence property, we can claim that<sup>12</sup>

$$\begin{aligned} E[Y | D = 1, p(X)] &= E[Y(1) | U \leq p(X), p(X)] \\ &= E[Y(1) | p(X)]. \end{aligned}$$

We similarly conclude that

$$\begin{aligned} E[Y | D = 0, p(X)] &= E[Y(0) | U > p(X), p(X)] \\ &= E[Y(0) | p(X)]. \end{aligned}$$

Hence, we obtain the theorem of Rosenbaum and Rubin.

12: Indeed, using the independence property and the law of iterated expectations,

$$\begin{aligned} &E[Y(1) | U \leq p, p(X) = p] \\ &= E[E[Y(1) | U \leq p, p(X) = p, X] | U \leq p, p(X) = p] \\ &= E[E[Y(1) | p(X) = p, X] | U \leq p, p(X) = p] \\ &= E[E[Y(1) | p(X) = p, X] | p(X) = p] \\ &= E[Y(1) | p(X) = p]. \end{aligned}$$

## 5.B Details of ATET

In observational studies, the ATET is identified under weaker conditions than the ATE because

$$E(Y(1) | D = 1) = E(Y | D = 1),$$

so we only need to identify  $E(Y(0) | D = 1)$ . We can state the weaker version of the ignorability and overlap conditions as follows:

**Assumption 5.B.1** (Ignorability and Overlap for Treated). (a) *Ignorability. Suppose that the treatment status  $D$  is independent of  $Y(0)$  conditional on a set of covariates  $X$ , that is*

$$D \perp\!\!\!\perp Y(0) \mid X.$$

(b) *Weak Overlap. Suppose that the propensity score satisfies:*

$$P(p(X) < 1) = 1.$$

**Theorem 5.B.1** (Identification of ATET) *Under Assumption 5.B.1,*

$$\delta_1 = E(Y \mid D = 1) - E[E(Y \mid X, D = 0) \mid D = 1].$$

Theorem 5.B.1 follows because, by iterated expectations and ignorability,

$$\begin{aligned} E(Y(0) \mid D = 1) &= E[E(Y(0) \mid D = 0, X) \mid D = 1] \\ &= E[E(Y \mid D = 0, X) \mid D = 1], \end{aligned}$$

where the outer expectation is well-defined because the support of  $X$  conditional of  $D = 1$  is a subset of the support of  $X$  conditional on  $D = 0$  by the overlap condition.

The Horvitz-Thompson method can be also used to recover averages of potential outcomes for the treated. Indeed,

$$E(DY)/ED = E(DY(1))/ED = E(Y(1) \mid D = 1)$$

and

$$\begin{aligned} E[(1 - D)p(X)Y/(1 - p(X))] / ED &= E[(1 - p(X))p(X)E(Y(0) \mid X)/(1 - p(X))] / ED \\ &= E[p(X)E[Y(0) \mid X]] / ED \\ &= E(DY(0)) / ED = E[Y(0) \mid D = 1]. \end{aligned}$$

Hence we obtain the following result.

**Theorem 5.B.2** (Propensity Score Reweighting for the Treated) *Under Assumption 5.B.1,*

$$E[Y\bar{H}] = \delta_1, \quad \bar{H} = Hp(X)/ED.$$

# Bibliography

- [1] Donald B. Rubin. 'Estimating causal effects of treatments in randomized and nonrandomized studies.' In: *Journal of educational Psychology* 66.5 (1974), pp. 688–701 (cited on page 96).
- [2] Paul R. Rosenbaum and Donald B. Rubin. 'The Central Role of the Propensity Score in Observational Studies for Causal Effects'. In: *Biometrika* 70.1 (1983), pp. 41–55 (cited on page 100).
- [3] Atila Abdulkadiroğlu, Joshua D. Angrist, Yusuke Narita, and Parag A. Pathak. 'Research Design Meets Market Design: Using Centralized Assignment for Impact Evaluation'. In: *Econometrica* 85.5 (2017), pp. 1373–1432 (cited on page 100).
- [4] Daniel G. Horvitz and Donovan J. Thompson. 'A generalization of sampling without replacement from a finite universe'. In: *Journal of the American statistical Association* 47.260 (1952), pp. 663–685 (cited on page 100).
- [5] Victor Chernozhukov, Mert Demirer, Esther Duflo, and Iván Fernández-Val. *Generic machine learning inference on heterogeneous treatment effects in randomized experiments, with an application to immunization in India*. Tech. rep. National Bureau of Economic Research, 2018 (cited on page 105).

# 6

## Causal Inference via Structural Equations and Conditional Exogeneity

Here we present the linear structural equation model framework and causal diagrams. The advantage of these models is they can be expressly linked to underlying structural models in economics and others fields, and they allow for transparent derivation of the conditional ignorability/conditional exogeneity assumption from the structure of the model. While linearity is imposed in this chapter, it will be dispensed with in later chapters.

6.1 Structural Equation Modelling and Conditional Exogeneity . . . . .	110
A Simple Triangular Structural Equation Model (SEM) . . . . .	110
6.2 Drawing the Model: Causal Diagrams, aka DAGs . . . . .	113
6.3 When Conditioning Can Go Wrong: Collider Bias aka Heckman Selection Bias . . . . .	115
6.4 Wage Gap Analysis and Discrimination . . . . .	117
6.A Details of the Wage Discrimination Analysis . . . . .	123

## 6.1 Structural Equation Modelling and Conditional Exogeneity

Basic ideas that appeared in econometrics between the 20s and 40s (P. Wright [1], S. Wright [2], J. Tinbergen [3], T. Haavelmo [4]) provide another take on and language for causality that is closely related to the potential outcomes framework.

### A Simple Triangular Structural Equation Model (SEM)

We shall illustrate the basic ideas using a simple model of a household's (say weekly) demand for gasoline, motivated by Hausman and Newey [5].

We start with a log-linear (Cobb-Douglas [6]) model for log-demand  $y$  given the log-price  $p$

$$y(p) := \delta p,$$

where  $\delta$  is the elasticity of demand. Demand is random across households, and we may model this randomness as

$$Y(p) := \delta p + U, \quad EU = 0, \tag{6.1.1}$$

where  $U$  is a stochastic shock that describes variation of demand across households (or across time, but assume that we are just looking at a particular time point). We immediately recognize that  $Y(p)$  plays the same role as a potential outcome in Rubin's potential outcome model.<sup>1</sup>

The stochastic function

$$p \mapsto Y(p)$$

describes a household's log-demand at a given log-price  $p$ . The expected log-demand at log-price  $p$  is given by  $EY(p) = \delta p$ . The function encodes various structural causal effects: If we change  $p$  from  $p_0$  to  $p_1$ , the expected demand change would be

$$EY(p_1) - EY(p_0) = \delta(p_1 - p_0).$$

The model (6.1.1) is very simple, and we may want to introduce covariates to capture other observable factors that may be associated with demand. That is, we may think there are observable parts of the stochastic shock, characterized by  $X$ , which help us predict household demand. Leading examples

<sup>1</sup>: The subtle difference here is that  $U$  does not depend on the index  $p$ , though we could make  $U$  be indexed by  $p$  at the cost of more complicated exposition. The distinction drawn is not superficial. Later on, when we discuss models with instruments, the dependence of  $U$  on  $p$  can create non-trivial problems which are not present in this section.

are household characteristics; e.g. we may think demand is associated with features such as family size, income, number of cars, or geographical location. We can incorporate these features by modelling  $U = X'\beta + \epsilon_Y$ , where  $\epsilon_Y$  is independent of  $X$  and has mean zero. Employing this model structure, we can write our augmented model as

$$Y(p) := \delta p + X'\beta + \epsilon_Y, \quad \epsilon_Y \perp\!\!\!\perp X. \quad (6.1.2)$$

Equation (6.1.2) is a structural stochastic model of economic outcomes. This model has nothing to do with regression or a statistical predictive model. Rather, it is a model that provides counterfactual predictions: If log-price is set to  $p$ , then a household with characteristics  $X$  can be predicted to purchase

$$\delta p + X'\beta$$

log-units. Here  $p$  is not a random variable – it is an index describing potential values of the price.

Then we ask the question:

- What data  $(Y, P, X)$  on quantities, prices, and characteristics should we collect to allow us to estimate the structural parameter  $\delta$ ?

**Assumption 6.1.1** (Conditional Exogeneity). (i) (Consistency) Suppose the observed variables  $(Y, P, X)$  are such that

$$Y = Y(P)$$

i.e. the outcome is generated from the structural model, (ii) (Conditional Exogeneity) The observed  $P$  is determined outside of the model, independently of  $\epsilon_Y$  conditional on  $X$ :

$$P \perp\!\!\!\perp \epsilon_Y | X \implies P \perp\!\!\!\perp \{Y(p)\}_{p \in \mathbb{R}} | X$$

Assumption 6.1.1 is the econometric analog of ignorability.<sup>2</sup> In the context of household demand, this condition requires that  $P$  is determined independently of a household's demand shock  $\epsilon_Y$ , conditional on characteristics  $X$ . This assumption seems plausible for household level decisions, especially if we include geography in the set of covariates  $X$ .

2: At a general level, gasoline prices are determined by aggregate supply and demand conditions, with small local geographic adjustments (e.g., gasoline prices in areas with higher prices of land may be higher than in other areas to reflect the higher land costs for gasoline stations). Conditional on being in a given small geographic region, we may think of price fluctuations as independent of household-specific demand shocks.

If the conditional exogeneity condition holds, then

$$Y = Y(P) = \delta P + X'\beta + \epsilon_Y, \quad \epsilon_Y \perp (P, X).$$

This means that the projection parameters of  $Y$  on  $P$  and  $X$  coincide with the structural parameters  $\delta$  and  $\beta$ .

We stress that our parameters  $\delta$  and  $\beta$  are not defined by regression; they are defined by the model. Under the conditional exogeneity condition, these parameters coincide with the projection parameters.<sup>3</sup>

We might further think that household characteristics are generally determined well before gasoline prices faced by individual households in any specific time period are set. Thus, we can postulate a structural equation for log-prices:

$$P(x) := x'\nu + \epsilon_P,$$

where  $P(x) = P(x_1)$  is the stochastic price process indexed by a subvector  $x_1$  of  $x$  (e.g., geographical characteristic  $x_1$ ), and  $\epsilon_P$  describes the centered stochastic price shock. We assume that observed  $X$  is independent of price shock  $\epsilon_P$ ,

$$X \perp\!\!\!\perp \epsilon_P.$$

This independence implies that  $\nu$  coincides with the projection coefficient of  $P$  on  $X$ .

Putting the equations together, we have a triangular structural equation model (TSEM):

$$\begin{aligned} Y &:= \delta P + X'\beta + \epsilon_Y, \\ P &:= X'\nu + \epsilon_P, \\ X, \end{aligned} \tag{6.1.3}$$

where  $\epsilon_Y$ ,  $\epsilon_P$ , and  $X$  are mutually independent (or at least uncorrelated) and determined outside of the model. They are called exogenous variables.  $Y$  and  $P$  determined within the model and called the endogenous variables. The structural parameter  $\delta$  can be identified by linear regression provided  $\text{Var}(\epsilon_P) > 0$ , and the structural parameter  $\nu$  can be identified by linear regression provided  $\text{Var}(X) > 0$ .

3: A weaker starting condition than the conditional exogeneity condition for the above result is simply

$$(P, X) \perp \epsilon_Y,$$

that is, the observed  $P$  and  $X$  are orthogonal to the structural error  $\epsilon_Y$ .

Under the conditions stated above the parameters of these structural equations coincide with the projection parameters.

**What do we mean by the model being structural?** The term structural means that each of the equations is **assumed** to provide comparative statics and answers counterfactual questions. Setting the right-hand-side variables to their potential values, we have

$$Y(p, x) := \delta p + x'\beta + \epsilon_Y,$$

$$P(x) := x'\nu + \epsilon_P.$$

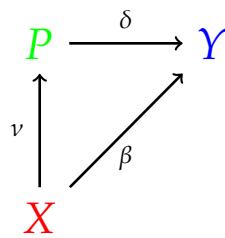
The conceptual operation of "setting" or "fixing" the variables is supposed to leave the structure invariant. More generally, the structural parameters are supposed to be invariant to changes in the distribution of exogenous variables –  $X, \epsilon_Y, \epsilon_P$  – that have been generated outside of the model. Therefore, we can use these structural parameters to generate counterfactual predictions.

The jargon *comparative statics* refers to the determination of how endogenous variables change in response to changes in exogenous variables. Similarly, *counterfactual questions* coincide with asking how outcomes or endogenous variables change when variables are set to new values with other features of the model remaining fixed; e.g. asking how demand changes when price is set to some new value by a firm with household characteristics, price shocks, and demand shocks unaffected.

## 6.2 Drawing the Model: Causal Diagrams, aka DAGs

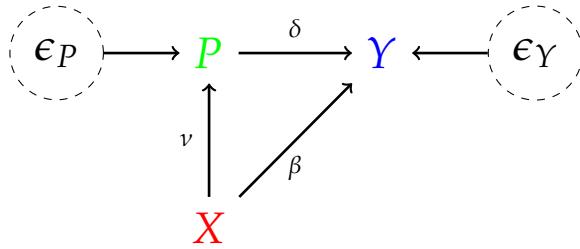
Sewall and Philip Wright [1], [2] would have depicted system of equations (6.1.3) graphically as a causal (path) diagram as in Figure 6.1. Observed variables are shown as nodes, causal paths are shown by directed arrows, and the structural (causal) parameters are given by the symbols placed next to the arrows.

The graph represents a structural economic model that can answer causal (comparative statics) questions. For example, the elasticity parameter  $\delta$  tells us how household demand will respond to a firm *setting* a new price. Note that a firm setting a new price will not alter household characteristics or the other exogenous features of the model, and thus only the parameter  $\delta$  is relevant for answering this question within the model.



**Figure 6.1:** A simple causal diagram representation of the TSEM for the household gasoline demand example.

We could have expanded the previous graph to include unobserved shocks  $\epsilon_P$  and  $\epsilon_Y$  as follows:



**Figure 6.2:** An expanded causal diagram representation of the TSEM that shows the unobserved shocks  $\epsilon_P$  and  $\epsilon_Y$  as root nodes.

The graph initiates with the *root nodes*  $\epsilon_P$ ,  $X$  and  $\epsilon_Y$ . The absence of links between the root nodes signifies the orthogonality between the nodes: namely, the absence of correlation. This structure is important because it allows identification of various structural parameters via projection as noted above. The nodes  $X$  and  $\epsilon_P$  are *parents* of  $P$ ; the nodes  $P$ ,  $X$ , and  $\epsilon_Y$  are *parents* of  $Y$ . The node  $Y$  is a *collider* on all paths, because it contains only incoming arrows.

The main effect of interest is  $\delta$ , which we call the structural causal effect of  $P$  on  $Y$ . This effect is identified after adjusting for  $X$ . In terms of the graph above, there are two paths connecting  $P$  and  $Y$ :

$$P \rightarrow Y \text{ and } P \leftarrow X \rightarrow Y.$$

The second path is called a "back-door path" because there is an arrow pointing back to  $P$  from  $X$ . This connection indicates that there is a common cause for  $P$  and  $Y$ . Figuratively speaking, controlling or adjusting for  $X$  is said to be like "closing the back-door" path, shutting down the non-causal sources of statistical dependence between  $Y$  and  $P$ .

This visual characterization of the adjustment for  $X$  is due to J. Pearl [7] and generalizes to much more complicated graphs. We revisit these ideas throughout subsequent chapters.

How do household characteristics impact our model?  $X$  affects  $Y$  through two paths:

- ▶ the direct effect  $\beta$  via  $X \rightarrow Y$ ,
- ▶ and the indirect effect  $v\delta$  via  $X \rightarrow P \rightarrow Y$ .

The indirect effect is said to be "mediated" by  $P$ . We saw in Section 6.1 that we can identify  $\delta$  and  $\beta$  from projection of  $Y$  on  $P$  and  $X$ , and we can identify  $\nu$  by projection of  $P$  on  $X$ . Therefore both the direct and indirect effects are identified.

The total effect of  $X$  on  $Y$  is

$$\nu\delta + \beta,$$

which can be identified in this case by projection of  $Y$  on  $X$ . To verify this, we plug the first equation from the TSEM in (6.1.3) into the second equation producing

$$Y = (\nu\delta + \beta)'X + V; \quad V = \epsilon_Y + \delta\epsilon_P.$$

We see that the composite disturbance  $V$  is orthogonal to  $X$ ,

$$V \perp X,$$

and, therefore,  $(\nu\delta + \beta)$  coincides with the projection coefficient in the projection of  $Y$  on  $X$ . The latter point can be seen graphically: There are no "back-door" paths from  $X$  to  $Y$ , so it is not necessary to adjust or control for anything to identify the total effect of  $X$  on  $Y$ .

In fact, while conditioning on  $P$  would allow us to identify the direct effect of  $X$ ,  $\beta$ , it would prevent us from retrieving the total effect  $\nu\delta + \beta$ . In empirical practice, we may think of conditioning on  $P$  as "conditioning on the outcome," as  $P$  is determined by its parents, including  $X$ , so may be thought of as an outcome relative to  $X$ .

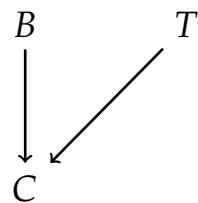
To retrieve a causal parameter of interest, then, we must first define the causal parameter of interest and then carefully consider the choice of what to condition on to learn this effect. These choices are particularly important given the existence of *collider bias*.

Mediation structures appeared right at the outset in the Wrights' work [1], [2].

### 6.3 When Conditioning Can Go Wrong: Collider Bias aka Heckman Selection Bias

Consider the following SEM:

$$\begin{aligned} T &:= \epsilon_T \\ B &:= \epsilon_B \\ C &:= T + B + \epsilon_C \end{aligned} \tag{6.3.1}$$



**Figure 6.3:** DAG with a collider representing SEM (6.3.1).

where  $\epsilon_T$ ,  $\epsilon_B$ , and  $\epsilon_C$  are independent  $N(0, 1)$  shocks.<sup>4</sup> Here the average structural function for  $T$ , which does not depend on what values  $B$  might take, is zero,

$$\mathbb{E}[T] = 0.$$

Regression without conditioning on  $C$  correctly identifies that  $T$  is not causally impacted by  $B$ :

$$\mathbb{E}[T \mid B = b] = 0.$$

However, further conditioning on  $C$  removes the causal interpretation of the projection coefficient:<sup>5</sup>

$$\mathbb{E}[T \mid B, C] = (C - B)/2; \implies \mathbb{E}[\mathbb{E}[T \mid B = b, C]] = -b/2 < 0.$$

This regression suggests that, controlling for  $C$ , the predictive effect of  $B$  on  $T$  is  $-1/2$ . This predictive effect is not a causal effect.

Collider bias illustrates that conditioning on outcomes may produce the wrong conclusions about causality, so conditioning on outcomes should be always approached with care. In econometrics, collider bias is known as a form of sample selection bias<sup>6</sup> ("conditioning on endogenous variables" or Heckman selection bias [8]).

**A Fun Digression on Colliders\***. Within our SEM framework, regression on a collider is clearly a wrong thing to do for identifying the causal effect of  $B$  on  $T$ . However, it is nonetheless *very useful* for other predictive tasks.

The following example draws on the discussion given in the "Book of Why" [9] to illustrate the collider bias.

**Example 6.3.1** (Structural Model of Hollywood) Suppose that the preceding SEM describes the people of Hollywood. In Hollywood, talent,  $T$ , coolness,  $B$  and congeniality,  $C$  are generated such that  $C$  is bigger than 0. As we showed, the causal effect of  $B$  on  $T$  in this SEM is 0. However, the best linear predictor of  $T$  given  $B$  conditional on  $C > 0$  is

$$\approx .6 - B/4.$$

That is, coolness and talent are negatively correlated in Hollywood. This correlation is useful for making predictions. For example, given the picture of the actor presented in the

4: [Link to An R Notebook](#) simulating SEM (6.3.1).

5: Dividing by 2 may seem counterintuitive, but it is correct. See the [R Notebook](#) for detail.

6: J. Heckman was awarded the Nobel Memorial prize "for his development of theory and methods for analyzing selective samples." Source: [Nobelprize.org](#)



**Figure 6.4:** Our SEM predicts that this actor, A. Terminator, is (essentially) the most talented actor in Hollywood.

margin, where  $B$  is clearly -20 standard deviations in the overall population, we predict the expected value of his talent to be  $t \in [+5.6 \pm 2]$ , which is at least 3.6 standard deviations above the average of zero in the overall population. From that we must infer that he is (practically) the most talented man or actor, for that matter.

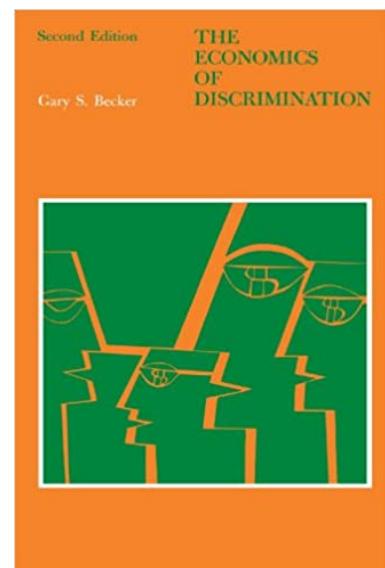
The example illustrates how simple theoretical models are often used in economics. Causal reasoning is made within a simple model, such as the SEM (6.3.1). This reasoning then leads to some testable restrictions, such as negative correlation between  $T$  and  $B$  conditional on  $C > 0$ . Even though we may not believe that the stylized model provides a complete model of reality, the implications of the simple model provide some insight into how observed phenomena, such as a negative correlation between  $T$  and  $B$  conditional on  $C > 0$ , may arise.

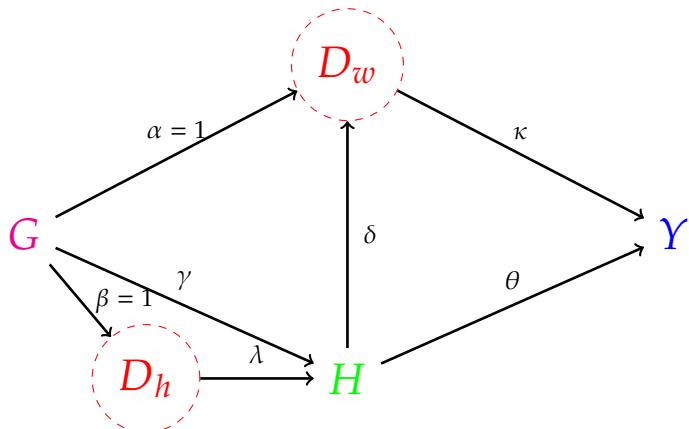
## 6.4 Wage Gap Analysis and Discrimination

"The central question in any employment-discrimination case is whether the employer would have taken the same action had the employee been of a different race (age, sex, religion, national origin etc.) and everything else had remained the same." (In Carson versus Bethlehem Steel Corp., 70 FEP Cases 921, 7th Cir. (1996) [10]).

Wage regressions are widely used [11][12] by labor economists to characterize the pay gap between men and women and to link the pay gap to discrimination. Several economists have asserted that it is wrong to study discrimination by doing wage gap regressions [13] and that we should instead look at the unconditional difference in outcomes across groups. Their reasoning is based on the argument that key job characteristics – e.g., education and occupation – are determined in response to both a group identity and discrimination and are therefore (intermediate) outcomes. Controlling for these characteristics may then introduce a form of selection bias. Which of the two sets of economists is right?

In what follows, we present a simple SEM in (6.4.1), which postulates that different groups receive equal wages if there are no conditional productivity differences between the groups. We will see that, in this SEM, wage gap regressions do uncover well-defined discrimination effects that occur in wage-setting





**Figure 6.5:** A Simple Model of Dis-crimination

mechanisms. In contrast, the unconditional average pay gap uncovers a more complicated causal object, which absorbs discrimination in wage setting, discrimination in human capital and occupational acquisitions, as well as group specific preferences for occupations.

Here we begin with the linear SEM and the equivalent DAG shown in Figure 6.5:

$$\begin{aligned}
 Y &:= \kappa D_w + \theta H + \epsilon_Y, \\
 D_w &:= \alpha G + \delta H + \epsilon_{Dw}, \\
 H &:= \gamma G + \lambda D_h + \epsilon_H, \\
 D_h &:= \beta G + \epsilon_{Dh}, \\
 G,
 \end{aligned} \tag{6.4.1}$$

where the shocks  $\epsilon_Y, \epsilon_{Dw}, \epsilon_H, \epsilon_{Dh}$ , and  $G$  are all mean zero and uncorrelated.

The outcome  $Y$  is wage,  $G$  is group (e.g., sex),  $H$  is human capital (a scalar index that includes labor-relevant characteristics such as education, occupation, etc.),<sup>7</sup>  $D_w$  is latent wage discrimination arising in the work-place, and  $D_h$  is latent discrimination arising in acquisition of human capital. There could be other observed confounders that we don't show for the sake of simplicity.

The discrimination variables  $D_w$  and  $D_h$  are latent variables that are important for our model but cannot be directly observed. We maintain throughout that these variables are non-degenerate and related to group identity  $G$ . Under these assumptions, the scale of these latent variables is non-zero but arbitrary, so we normalize the effect  $G \rightarrow D_w$  to unity,  $\alpha = 1$ , and the effect  $G \rightarrow D_h$  to unity as well,  $\beta = 1$ . There is no edge from

7:  $H$  can be easily made a vector with a slightly more complicated notation.

$G$  to  $Y$ , reflecting our assumption that there is no systematic group difference in productivity conditional on  $H$  and  $D_w$ . In the absence of productivity differences between workers, economic reasoning suggests that they would be assigned the same wage in a discrimination-free economy [14]. Thus, we would expect  $\kappa = 0$  in a discrimination-free economy in the case that  $H$  captures all sources of productivity differences between workers.

Within this model, the parameter of interest is then the causal or structural effect of discrimination on wages given by

$$\kappa.$$

If  $\kappa \neq 0$ , we can conclude that wages are assigned unfairly within the framework of this SEM.

If we observed  $D_w$  directly, we could learn the effect of discrimination on wages,  $\kappa$ , by regression of  $Y$  on  $D_w$  and  $H$ . Identification of  $\kappa$  from this regression follows from the backdoor criterion discussed in Section 6.2. We don't observe  $D_w$  directly, but we postulate that this variable is determined only by  $G$ ,  $H$ , and a stochastic shock. Dependence on  $H$  captures the idea that discrimination may be larger or smaller depending on education level, profession, etc. We return to using this additional structure to learn about  $\kappa$  below.

Discrimination may operate through channels other than simple wage differences. For example, in the 1960s, there were relatively few women or African American lawyers, a highly paid occupation. Discrimination that operates through occupational choice or human capital formation is captured by latent variable  $D_h$ . In our model,  $H$ , which captures productivity differences between individuals, can be determined as a result of both discrimination and group preferences.<sup>8</sup> The parameter  $\gamma$  then captures the effect of group preferences on the formation of  $H$ , while the effect of discrimination on  $H$  is captured by  $\lambda$ . Since  $D_h$  is not observed, there is no way to separately identify these two effects.

It is easy to show, within the model, that the population linear regression of  $Y$  on  $G$  and  $H$  recovers the wage discrimination effect,

$$\kappa,$$

8: For example, 90% of firefighters in the US are men, which may reflect a genuine preference for this occupation among men. At the same time, even preference for occupation may be a result of cultural institutions that could themselves be interpreted as discriminatory in broader, cross-cultural contexts.

and that the linear regression of  $H$  on  $G$  recovers

$$\gamma + \lambda,$$

the sum of the group preference effect for occupation and the occupation discrimination effect; see Appendix C for details. If a further strong assumption is made that there is no group preference effect,  $\gamma = 0$ , the linear regression of  $Y$  on  $G$  recovers the total discrimination effect:

$$\kappa + \lambda(\kappa\delta + \theta).$$

**Endogenous Sample Selection.** There is an important issue with our empirical example. We are only able to look at earnings of people who are employed. Thus, we are conditioning on

$$Y > R,$$

where  $R$  is the reservation wage. In other words, we are conditioning on the outcome which may cause major selectivity issues: People get employed, and end up in our data, only if the offered wage is higher than some reservation wage. This sample selection on the basis of the outcome can cause major biases in the analysis. The potential for large biases was recognized by James J. Heckman [8] in the 70s and led to the development of the celebrated Heckman selection correction and related methods.

An alternate approach to applying a selection correction in our example is to select a subset  $S$  of people who are employed with probability one (or very close to one). For example, one could look at highly educated, unmarried people. Within this subset, we would then have

$$P(Y > R|S) \approx 1.$$

That is, the value of the wage offer,  $Y$ , is approximately unrelated to whether we observe individual wages for this subset of people. This type of strategy has been employed by Casey Mulligan [15]. Mulligan continues to find evidence in favor of the existence of wage gaps in his analysis of a subsample where selection effects are small. This finding then suggests that the broad conclusion of the existence of wage gaps is not driven entirely by sample selection issues.

In summary, we have the following observations:

- ▶ In general, wage gap regressions just estimate predictive effects or associations.
- ▶ When we assume an SEM like the one above holds and there are no endogenous sample selection effects, wage gap regressions estimate wage discrimination effects.
- ▶ Unconditional wage gaps generally reflect a combination of different types of discrimination and group preferences and thus do not isolate solely the effects of discrimination.

## Notebooks

- ▶ [Collider Bias R Notebook](#) provides a simple simulated example of collider bias, informing our discussion of conditioning on Congeniality in our Structural Model of Hollywood.

## Notes

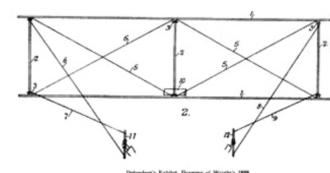
This chapter presented an approach to causal inference that goes back to the works of Sewall and Philip Wright [1], [2], Tinbergen [3], Haavelmo [4], and others. This tradition lives in modern structural causal models used in econometrics (especially, industrial organization) and in the artificial intelligence community. The latter community, inspired by the foundational work of J. Pearl [7], strongly adopted the use of causal diagrams, known as directed acyclical graphs (DAGs). We continue exploring this approach throughout the remainder of our treatment on causal inference.

## Study Problems

1. Explain collider bias to a friend in simple terms. Use no more than two paragraphs. Illustrate your explanation using a simulation experiment.
2. Empirical: Revisit the group wage gap analysis from Chapter 4, focusing on college-educated workers. Is there a structural/causal interpretation for the estimated wage gap? Is there a group gap in education achievement? Does

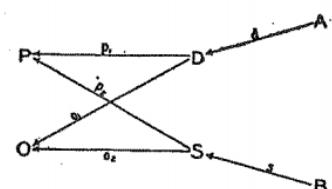


**Figure 6.6:** Early 20th century: The work of Sewall and Philip Wright made it possible for humans to begin to "fly" in the space of causal models. Another family of Wrights made it possible for humans to begin to fly in the air.



**Figure 6.7:** An early drawing for an airplane appears very much like an early drawing of a DAG.

**FIGURE 10.**



**Figure 6.8:** DAG for Supply-Demand Systems in P. Wright's work in 1928 [1].

this group gap in education have a structural/causal interpretation? Some of these questions are open ended and have no simple answers, but it is useful to think about them. (If you have other data sets that might illuminate discrimination in other settings, please use them in place of the wage data set).

3. Free-style exercise: The model for wage discrimination presented in our notes is very stylized and subject to multiple criticisms. For example, it does not deal with promotion and hiring decisions. There are several interesting models of discrimination in hiring, college admissions, and pay. For example, see "The Book of Why"[\[9\]](#) and [the Bickel et al. 1975 paper](#) [\[16\]](#) for an analysis of Berkeley undergraduate admissions decisions. [Nina Roussile's](#) (2020) [\[17\]](#) job market paper isolates the ask gap as the central mechanism for the subsequent pay gap. Referring to one such analysis, draw or write down a linear structural causal models that captures the structural idea of the analysis and discuss identification in the model.

## 6.A Details of the Wage Discrimination Analysis

We write out some of the structural equations corresponding to the DAG:

$$Y := \kappa D_w + \theta H + \epsilon_Y, \quad \epsilon_Y \perp D_w, H, G$$

$$D_w := G + \delta H + \epsilon_{Dw}, \quad \epsilon_{Dw} \perp G, H$$

where the orthogonality relations are implied by the model.

Linear regression analysis would use observable variables only, so we substitute the model for the unobserved  $D_w$  in terms of  $G$  and  $H$  into the equation for  $Y$  to obtain

$$Y = \kappa G + (\kappa\delta + \theta)H + U, \quad U := \kappa\epsilon_{Dw} + \epsilon_Y \perp (G, H).$$

The composite error term  $U$  is orthogonal to  $G$  and  $H$ . Therefore, regression of  $Y$  on  $G$  and  $H$  learns  $\kappa$  and  $(\kappa\delta + \theta)$ , with our main target being  $\kappa$ . We can also see that by partialling out  $H$ ,

$$\tilde{Y} = \kappa \tilde{G} + U, \quad U \perp \tilde{G}.$$

"This is elementary, my dear Watson," said Sherlock Holmes after seeing this.

Thus,  $\kappa$  is retrievable only if there is non-zero variation in  $\tilde{G}$  after taking out the linear effect of  $H$ .

Now suppose we want to study discrimination effects in occupational choices, captured by  $H$  in our model. We write out the relevant structural equations:

$$H := \gamma G + \lambda D_h + \epsilon_H, \quad \epsilon_H \perp (G, D_h),$$

$$D_h := G + \epsilon_{Dh}, \quad \epsilon_{Dh} \perp G.$$

Recall that  $\gamma$  is the group preference effect and  $\lambda$  is the discrimination effect. Since  $D_h$  is not directly observed, we substitute it out to arrive at

$$H = (\gamma + \lambda)G + V; \quad V := \gamma\epsilon_{Dh} + \epsilon_H \perp G.$$

Therefore,  $\gamma + \lambda$  is the projection coefficient in the projection of  $H$  on  $G$ . Hence, we can identify  $\gamma + \lambda$ , but we can't identify  $\gamma$  and  $\lambda$  separately.

Going further, suppose that the group preference effect is zero, so  $\gamma = 0$ . Then, the previous argument would identify  $\lambda$  and we could identify the total discrimination effect arising from two different channels:

$$\kappa + \lambda(\kappa\delta + \theta).$$

from the regression of  $Y$  on  $G$ .

We can assert that the unconditional difference in wages measures discrimination only if the group preference effect in determining  $H$  is zero ( $\gamma = 0$ ). Of course, most economists would probably not agree with the assumption that  $\gamma = 0$ . Empirically, there are large differences in group composition among different professions. These differences likely reflect both discrimination and genuine preferences.

# Bibliography

- [1] Philip G. Wright. *The tariff on animal and vegetable oils*. New York: The Macmillan company, 1928 (cited on pages 110, 113, 115, 121).
- [2] Sewall Wright. 'Correlation and Causation'. In: *Journal of Agricultural Research* 20.7 (1921), pp. 557–585 (cited on pages 110, 113, 115, 121).
- [3] Jan Tinbergen. 'Bestimmung und Deutung von Angebotskurven Ein Beispiel'. In: *Zeitschrift für Nationalökonomie* 1.5 (1930), pp. 669–679 (cited on pages 110, 121).
- [4] Trygve Haavelmo. 'The probability approach in econometrics'. In: *Econometrica: Journal of the Econometric Society* 12 (1944), pp. iii–vi+1–115 (cited on pages 110, 121).
- [5] Jerry A. Hausman and Whitney K. Newey. 'Nonparametric estimation of exact consumers surplus and dead-weight loss'. In: *Econometrica: Journal of the Econometric Society* 63.6 (1995), pp. 1445–1476 (cited on page 110).
- [6] Charles W. Cobb and Paul H. Douglas. 'A Theory of Production'. In: *The American Economic Review* 18.1 (1928), pp. 139–165 (cited on page 110).
- [7] Judea Pearl. *Causality*. Cambridge university press, 2009 (cited on pages 114, 121).
- [8] James J. Heckman. 'Sample selection bias as a specification error'. In: *Econometrica: Journal of the econometric society* 47.1 (1979), pp. 153–161 (cited on pages 116, 120).
- [9] Judea Pearl and Dana Mackenzie. *The Book of Why* (cited on pages 116, 122).
- [10] 'Carson v. Bethlehem Steel Corp.' In: 82 F.3d 157, 158, 7th Cir. (1996) (cited on page 117).
- [11] Francine D. Blau and Lawrence M. Kahn. 'The gender wage gap: Extent, trends, and explanations'. In: *Journal of economic literature* 55.3 (2017), pp. 789–865 (cited on page 117).
- [12] Sonja C. Kassenboehmer and Mathias G. Sinning. 'Distributional changes in the gender wage gap'. In: *ILR Review* 67.2 (2014), pp. 335–361 (cited on page 117).
- [13] Elise Gould, Jessica Schieder, and Kathleen Geier. 'What is the gender pay gap and is it real'. In: *Economic Policy Institute* (2016) (cited on page 117).

- [14] Gary S. Becker. *The economics of discrimination*. University of Chicago press, 2010 (cited on page 119).
- [15] Casey B. Mulligan and Yona Rubinstein. ‘Selection, Investment, and Women’s Relative Wages Over Time’. In: *The Quarterly Journal of Economics* 123.3 (2008), pp. 1061–1110. doi: [10.1162/qjec.2008.123.3.1061](https://doi.org/10.1162/qjec.2008.123.3.1061) (cited on page 120).
- [16] Peter J. Bickel, Eugene A. Hammel, and J. William O’Connell. ‘Sex Bias in Graduate Admissions: Data from Berkeley: Measuring bias is harder than is usually assumed, and the evidence is sometimes contrary to expectation.’ In: *Science* 187.4175 (1975), pp. 398–404 (cited on page 122).
- [17] Nina Roussille. ‘The central role of the ask gap in gender pay inequality’. In: URL: [https://ninaroussille.github.io/files/Roussille\\_askgap.pdf](https://ninaroussille.github.io/files/Roussille_askgap.pdf) 34 (2020), p. 35 (cited on page 122).

# Predictive Inference via Modern High Dimensional Nonlinear Regression

# 7

Here we discuss nonlinear regression methods based on tree models and (deep) neural network models. Tree-based methods include regression trees, random forests, and boosted trees. Regression trees are great for exploration and explainable analytics, while random forests and boosted trees are great predictive tools for structured data and data sets of intermediate size (say, up to several million observations). Neural networks are extremely flexible nonlinear regression methods and are particularly successful for data sets of larger size.

7.1 Introduction . . . . .	128
7.2 Regression Trees and Random Forests . . . . .	128
Introduction to Regression Trees . . . . .	128
Random Forests . . . . .	131
Boosted Trees . . . . .	133
7.3 Neural Nets / Deep Learning . . . . .	134
Basic Ideas . . . . .	134
Deep Neural Networks	139
7.4 Prediction Quality of Modern Nonlinear Regression Methods . . . . .	140
Learning Guarantees of DNNs . . . . .	141
Learning Guarantees of Trees and Forests . . . . .	143
Trust but Verify . . . . .	146
A Simple Case Study using Wage Data . . . . .	147
7.5 Combining Predictions - Aggregation - Ensemble Learning . . . . .	148
Auto ML Frameworks	149
7.6 When do Neural Networks win? . . . . .	150
7.A Variable Importance via Permutations . . . . .	153

## 7.1 Introduction

We are interested in predicting an outcome  $Y$  using raw regressors  $Z$ , which are  $k$ -dimensional. The best prediction rule  $g(Z)$  under square loss is the conditional expectation (CE) of  $Y$  given  $Z$ :

$$g(Z) = E(Y|Z).$$

In previous chapters, we used best linear prediction rules to approximate  $g(Z)$  and linear regression or Lasso regression for estimation. Now we consider nonlinear prediction rules to approximate  $g(Z)$ , focusing on tree-based methods and neural networks.

The use of Best Prediction rules (CEs) is not just important for generating good predictions, but is crucial for causal inference. Identification of causal parameters such as ATE via conditioning strategies requires us to work CEs rather than with best linear prediction rules. Previously we tried to make best linear prediction rules flexible to try to approximate best prediction rules. Here we explore fully nonlinear strategies.

## 7.2 Regression Trees and Random Forests

### Introduction to Regression Trees

Regression Trees are based on partitioning the regressor space (the space where  $Z$  takes on values) into a set of rectangles. A simple model is then fit within each rectangle.

The most common approach fits a simple constant model within each rectangle, which corresponds to approximating the unknown function by a “step function.” Given a partition into  $M$  regions, denoted  $R_1, \dots, R_M$  the approximating function when a constant is fit within each rectangle is given by

$$f(z) = \sum_{m=1}^M \beta_m 1(z \in R_m),$$

where  $\beta_m, m = 1, \dots, M$  denotes a constant for each region and  $1(\cdot)$  denotes the indicator function.

Suppose we have  $n$  observations  $(Z_i, Y_i)$  for  $i = 1, \dots, n$ . The estimated coefficients for a given partition are obtained by

minimizing the in-sample MSE:

$$\hat{\beta} = \arg \min_{b_1, \dots, b_M} \mathbb{E}_n \left( Y_i - \sum_{m=1}^M b_m 1(Z_i \in R_m) \right)^2,$$

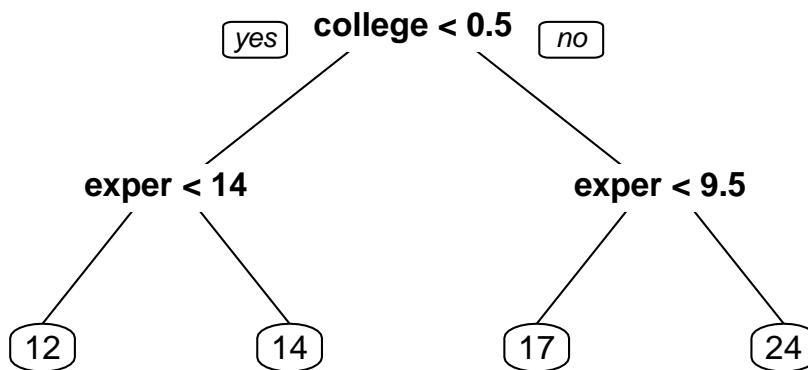
so that

$$\hat{\beta}_m = \text{average of } Y_i \text{ where } Z_i \in R_m.$$

The regions  $R_1, \dots, R_M$  are called nodes, and each node  $R_m$  has a predicted value  $\hat{\beta}_m$  associated with it.

A nice feature of regression trees is that you get to draw cool pictures, so let's explore their usage graphically in the context of our wage example. In this example, the outcome variable  $Y$  is hourly wage; and  $Z$  includes experience, geographic, and educational characteristics.

Figure 7.1 illustrates a simple regression tree for the wage data. This tree has a depth of two, meaning that predictions are produced as a sequence of two binary decisions (or partitions of the data). Starting at the top of the tree and working down provides a simple prediction rule for any observation. For example, the predicted wage for a worker without a college degree ( $\text{college} = 0$ ) and with less than 14 years of experience ( $\text{exper} < 14$ ) is 12 dollars an hour. We obtain this prediction by starting at the top of the tree and taking the left branch because  $\text{college} = 0 < .5$ . We then go left again at the second step because  $\text{exper} < 14$  and arrive at the predicted value of 12.



**Figure 7.1:** Regression tree based on wage data. The bottom nodes on the tree provide prediction rules for different subsets of observations. For example, the predicted hourly wage for a college educated worker with 9.5 or more years of experience (a worker with  $\text{college} = 1$  and  $\text{exper} \geq 9.5$ ) is 24 dollars.

The key feature of trees is that the cut points for the partitions are adaptively chosen based on the data. That is, the splits are not pre-specified but are purely data dependent. So, how did we use the data to grow the tree in Figure 7.1?

- **Growing the Tree: Level 1.** To make computation tractable, we use recursive binary partitioning or splitting of the regressor space. First, we cut the regressor space into two regions by choosing the regressor and splitting point

such that using the prediction rule fit within each region produces the best improvement in the in-sample MSE.<sup>1</sup>

Applying this procedure in the wage data gives us the depth 1 tree shown Figure 7.2. In this case, the best regressor to split on is the indicator of college degree, that takes values 0 or 1. Here splitting at any point between 0 and 1 provides the same rule, and an often used convention for binary variables is to use the “natural” split point of .5. Applying this split point yields the initial prediction rule: an hourly wage of \$20 for college graduates and \$13 for others.

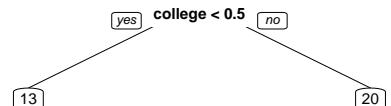
- ▶ **Growing the Tree: Level 2.** To grow the tree to depth 2, we then repeat the procedure for choosing the first partition rule within the two regions resulting from the first step. This step will result in a partition of the covariate space into four new regions. It is important to note that the two splits produced at this point may use different variables/splitting points than before. This feature means that the tree algorithm can create “interactions” and “nonlinearities” without requiring input from the user.

In our example, the regions resulting from applying the first splitting rule correspond to college graduates and non-college graduates). For college graduates, the partitioning rule that minimizes in-sample MSE is to split this group into those with less than 9.5 years of experience and those with 9.5 years or more of experience. We have thus refined the prediction rule for graduates to be \$24 an hour if experience is greater than or equal to 9.5 years, and \$17 an hour otherwise. For non-graduates the procedure works similarly, though here the in-sample MSE minimizing split is produced by dividing non-graduates into those with less than 14 years of experience and those with 14 years of experience or more.

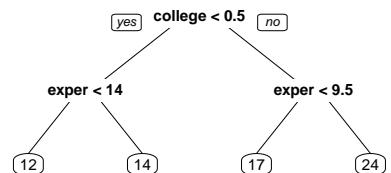
- ▶ **Growing the Tree: Higher Levels and Stopping Rule.** To grow deeper trees corresponding to more complex prediction rules, we simply keep repeating. We stop when the desired depth of the tree is reached,<sup>2</sup> or when a prespecified minimal number of observations per region, called minimal node size, is reached.

In the wage example, we can grow a depth 3 tree by repeating the basic procedure within each of the four nodes of the depth 2 tree. The resulting tree is illustrated in Figure 7.4. Here, we see that the gender indicator (female), high-school graduate indicator (hsg), and “Southern re-

1: To be clear, note that, in principle, finding this split point requires trying the partition produced by splitting the data along every possible value of every observed variable. That is, we are neither pre-specifying which variables nor which split points are important in providing a good prediction rule.



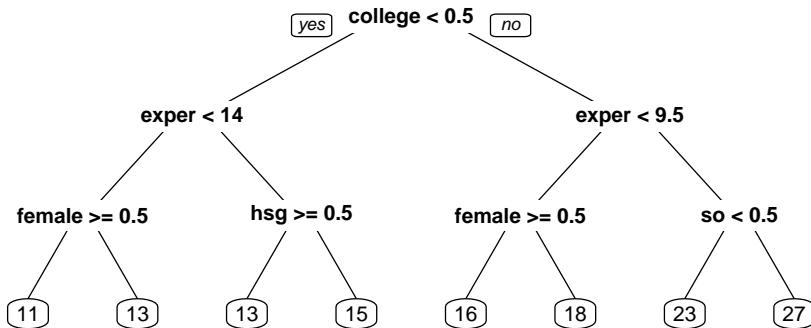
**Figure 7.2:** Depth 1 tree in the wage example



**Figure 7.3:** Depth 2 tree in the wage example

2: One practical choice of the depth of a tree is to stop just before we get a headache from looking at a complicated tree. This rule is indeed useful if we want to present the tree as a communication device.

gion” indicator ( $so$ ) are the splitting variables chosen in the third level.



**Figure 7.4:** Depth 3 tree in the wage example. The depth of three was chosen to avoid getting headaches from looking at a more complicated tree.

**Pruning Regression Trees.** We now make several observations.

First, the deeper we grow the tree, the better is our approximation to the regression function  $g(Z)$ . However, the deeper the tree, the noisier our estimate  $\hat{g}(Z)$  becomes, since there are fewer observations per terminal node to estimate the predicted value for this node. From a prediction point of view, we can try to find the right depth or the structure of the tree by sample-splitting (cross-validation). For example, in the wage example, the tree of depth 2 performs better in terms of cross-validated MSE than the tree of depth 3 or 1. The process of cutting down the branches of the tree to improve predictive performance is called “Pruning the Tree”.

Often for business analytics and explainability, simple trees like the ones shown are used. If we only care about building good prediction rules, we may build complicated trees and apply pruning to improve predictive performance. A simple penalty for the complexity of the tree is the number of leaves (terminal nodes) times a penalty level, where the penalty level is chosen heuristically; see, e.g, [1]. For example, we can always try sample splitting or cross-validation to settle on a penalty level. There is not a rigorously justified plug-in penalty level for trees like there is for lasso. Figuring out such a plug-in rule is actually a good research problem.



**Figure 7.5:** “To prune a tree”.  
Source: Wikipedia

## Random Forests

In practice, regression trees often do not provide the best predictive performance, because a single regression tree provides a relatively crude approximation to a smooth regression function  $g(Z)$ . We illustrate the potential poor approximation of regression trees in Figures 7.6 and 7.7. These figures simply

illustrate that step functions, which are the outputs of typical regression tree implementations, struggle in approximating smooth functions.

A powerful and widely used approach that aims to improve upon simple regression trees is to build a Random Forest, as proposed by Leo Breiman [2]. The idea of a Random Forest is to grow many different deep trees that have low approximation error and then average the prediction rules across trees.

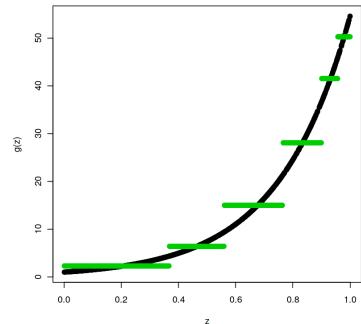
To produce different trees using only the observed data, the trees going into a random forest are grown from artificial data generated by sampling randomly with replacement from the original data; that is, each tree in a random forest is fit to a bootstrap sample. Within the bootstrap samples, trees are grown deep to keep approximation error low. Averaging across the trees produced in the bootstrap samples is then meant to reduce the noisiness of the individual trees. The procedure of averaging noisy prediction rules over bootstrap samples is called Bootstrap Aggregation or Bagging. When the data set is large, we can also rely on fitting trees within subsamples instead of using the bootstrap. Using subsamples offers some computational advantages and also simplifies theoretical analysis.

The idea seems very unusual, so let us explain again.

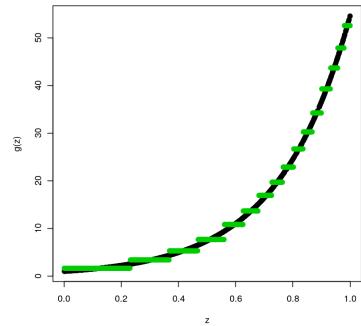
Each bootstrap sample is created by sampling from our data on pairs  $(Y_i, Z_i)$  randomly, with replacement. Hence, some observations are drawn multiple times and some aren't redrawn at all. Given a bootstrap sample, indexed by  $b$ , we build a tree-based prediction rule  $\hat{g}_b(Z)$ . We repeat the procedure  $B$  times in total, and then average the prediction rules that result from each of the bootstrap samples:

$$\hat{g}_{\text{random forest}}(Z) = \frac{1}{B} \sum_{b=1}^B \hat{g}_b(Z).$$

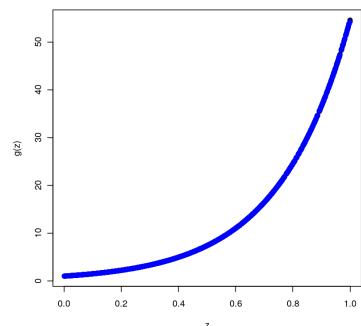
The use of the bootstrap here is an unusual, yet corresponds to an intuitive idea: If we could have many independent copies of the data, we could obtain low-bias but potentially very noisy prediction rules in each copy of the data and then average the prediction rules obtained over these copies to reduce the noise. Since we don't have many copies in reality, we rely on the bootstrap to create many quasi-copies of the data. Another feature of this idea is that the cut-points defining partitions for the tree obtained within each bootstrap sample will be different,



**Figure 7.6:** Approximation of  $g(Z) = \exp(4Z)$  by a shallow Regression Tree in the noiseless case.



**Figure 7.7:** Approximation of  $g(Z) = \exp(4Z)$  by a deep Regression Tree in the noiseless case.



**Figure 7.8:** Approximation of  $g(Z) = \exp(4Z)$  by a Random Forest in the noiseless case.

producing a different step function approximation. Averaging over many step functions with steps at different locations will potentially produce a much smoother approximation to the underlying function. The improved approximation relative to simple trees is illustrated in Figure 7.8.

There are many different modifications of the simple version of bootstrap aggregation that we have discussed. The most important modification is the use of additional randomization to "decorrelate" the trees: When we build trees over different bootstrap samples, we also randomize over the variables that trees are allowed to use in forming partitions. This additional layer of randomization results in trees having different structure at the deepest levels in different bootstrap samples.

In a summary, a Random Forest is an average of tree based prediction rules (a forest) produced from bootstrap or subsample data (generated randomly).

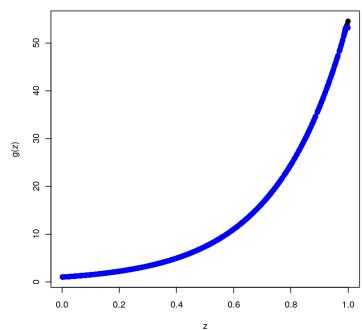
## Boosted Trees

The idea of boosting is that of a recursive fitting: We estimate a simple prediction rule, then take the residuals and estimate another simple prediction rule for these residuals, and so on. A sum of the prediction rules for the residuals then gives us the prediction rule for the outcome.

A common use of boosting is with regression trees. Here we use shallow trees as the simple prediction rule. Shallow trees produce low noise prediction rules, but also tend to have high approximation error. However, each step where a model is fit to the residuals from the previous step reduces the approximation error. In order to avoid overfitting, we can stop the procedure once there is no marginal improvement to the cross-validated MSE. The improved approximation of boosted trees relative to simple trees is illustrated in Figure 7.9.

### The boosting algorithm

1. Initialize the residuals:  $R_i := Y_i, i = 1, \dots, n$ .
2. For  $j = 1, \dots, J$ 
  - a) fit a tree-based prediction rule  $\hat{g}_j(Z)$  to the data  $(Z_i, R_i)_{i=1}^n$ ;
  - b) update the residuals  $R_i := R_i - \lambda \hat{g}_j(Z_i)$ , where  $\lambda$  is called the learning rate.



**Figure 7.9:** Approximation of  $g(z) = \exp(4z)$  by Boosted Trees in the noiseless case with a sufficient number of steps  $J$ .

3. Output the boosted prediction rule:

$$\hat{g}(Z) := \sum_{j=1}^J \lambda \hat{g}_j(Z).$$

In practice, using boosted trees requires making several choices. One needs to define the tree-based prediction rule used at each step and also choose the number of learning steps,  $J$ , and the learning rate,  $\lambda$ . These tuning parameters can be chosen by cross-validation.<sup>3</sup> Note that the boosting algorithm is quite general and can be applied to non-tree uses. A very popular implementation widely used in industry is `xgboost`, which has the capability to impose qualitative shape constraints like monotonicity in one or several variables.

<sup>3</sup>: A default value for  $\lambda$  is 0.1,  $0 < \lambda < 1$ . The idea is to fit simple prediction rules, so one will typically specify the prediction rule by setting the depth of the trees to a small number. For example, at each step, the prediction rule may be a regression tree of depth two.

## 7.3 Neural Nets / Deep Learning

Neural networks are a very powerful tool for modelling non-linear relationships. They rely on many constructed regressors to approximate  $g(Z)$ , the conditional expectation given the regressors. The method and the name "neural networks" were loosely inspired by the mode of operation of the human brain, and developed by scientists working in Artificial Intelligence. They can be represented by cool graphs and diagrams.

### Basic Ideas

First, we focus on a single layer neural network to introduce the more formal definition of neural nets. The estimated prediction rule will take the form:

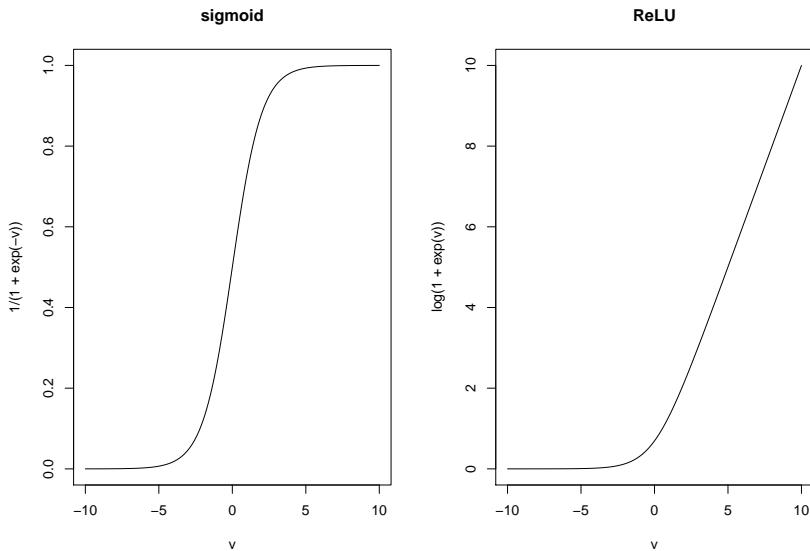
$$\hat{g}(Z) := \hat{\beta}' X(\hat{\alpha}) := \sum_{m=1}^M \hat{\beta}_m X_m(\hat{\alpha}_m),$$

where the  $X_m(\hat{\alpha}_m)$ 's are constructed regressors called *neurons*,

$$\alpha = (\alpha_m)_{m=1}^M, \quad \beta = (\beta_m)_{m=1}^M, \quad X(\alpha) = (X_m(\alpha_m))_{m=1}^M.$$

We always take  $Z$  to include a constant of 1 as a component and set  $X_1(\alpha) = 1$ . The remaining neurons are generated as

$$X_m(\alpha_m) = \sigma(\alpha'_m Z), \quad m = 2, \dots, M,$$



**Figure 7.10:** The sigmoid (logit) and smoothed ReLU activation functions

where  $\alpha_m$ 's are neuron-specific vectors of parameters called weights, and  $\sigma$  is an activation function chosen by the practitioner. Example activation functions are

- the sigmoid function,

$$\sigma(v) = \frac{1}{1 + e^{-v}},$$

- the rectified linear unit function (ReLU),

$$\sigma(v) = \max(0, v),$$

- the smoothed rectified linear unit function (SReLU),

$$\sigma(v) = \log(1 + \exp(v)),$$

- or the linear function,

$$\sigma(v) = v.$$

The use of nonlinear activation functions is critical for generating high-quality approximations.

The estimators  $\{\hat{\alpha}_m\}$  and  $\{\hat{\beta}_m\}$ , for  $m = 1, \dots, M$ , are obtained as the solution to a penalized nonlinear least squares problem. For example, we could obtain parameter estimates by solving

$$\min_{\{\alpha_m\}, \{\beta_m\}} \sum_i \left( Y_i - \sum_{m=1}^M \beta'_m X_{im}(\alpha_m) \right)^2 + \text{pen}(\alpha, \beta; \lambda), \quad (7.3.1)$$

where  $\text{pen}(\alpha, \beta; \lambda)$  is a penalty function with penalty parameter

λ. Common penalty functions are lasso-type  $\ell_1$  penalties,

$$\lambda \left( \sum_m \sum_j |\alpha_{mj}| + \sum_m |\beta_m| \right),$$

and Ridge-type  $\ell_2$  penalties,

$$\lambda \left( \sum_m \sum_j (\alpha_{mj})^2 + \sum_m (\beta_m)^2 \right).$$

Neural network estimates are typically computed using stochastic gradient descent (SGD) algorithms. In SGD, gradients are computed on subsamples of data (often consisting on a single observation) called batches, and a single cycle through all subsamples is termed an “epoch.” By only making use of batches of observations, SGD algorithms are able to scale to massive data sets. Using subsamples of data introduces “stochasticity” relative to using the “full” gradient computed on the entire data. This noise in the computation of gradients also seems to have advantages in helping SGD algorithms avoid local saddle points. There are many fine practical details in terms of efficient computation of gradients for deep neural nets, how updating is done in SGD algorithms in general, and in the application of SGD to learning parameters of deep neural nets.<sup>4</sup>

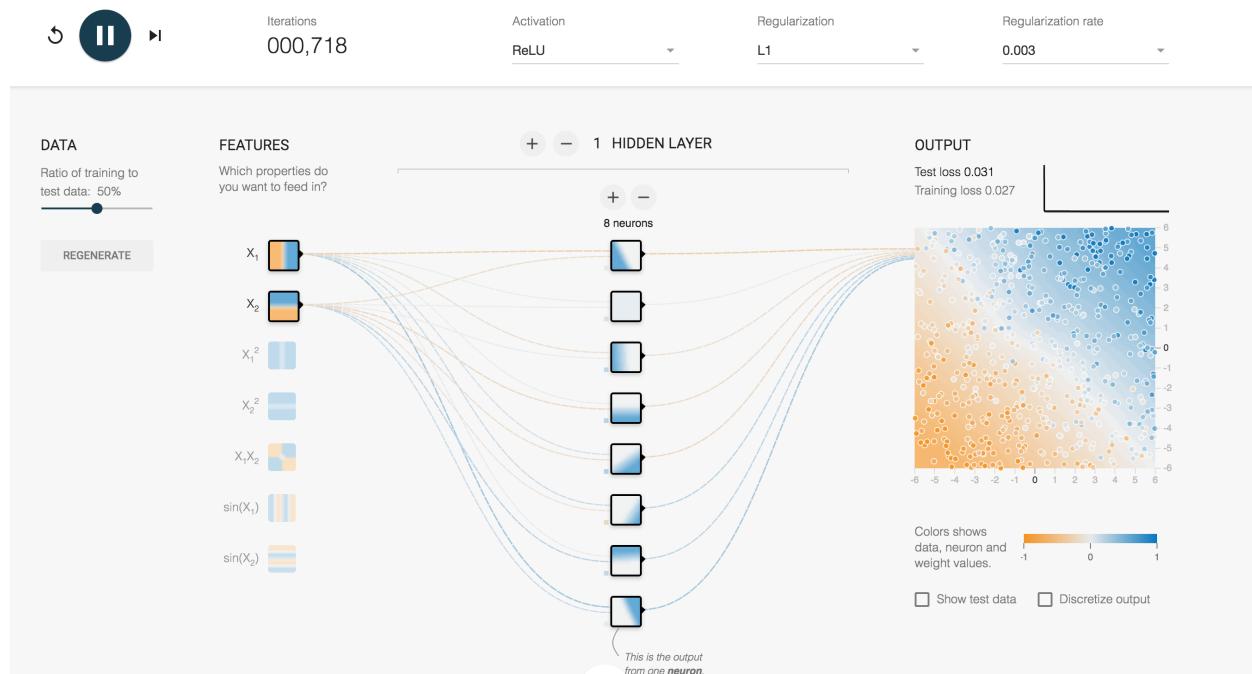
The optimization methods employed for learning neural network parameters provide avenues for regularization beyond simply penalizing the size of the coefficients. A popular regularization method is *dropout* regularization where each neuron in a given layer can be set to zero with a given probability – for example, .1 – during parameter update steps. Dropout encourages more robust networks: If a particular neuron is important, the dropout regularization encourages creation of very similar neurons that can replicate the properties of the given neuron. Therefore, dropout regularization can be viewed as a penalty that forces similar weights for groups of neurons.

Another commonly used regularization device used with neural networks is *early stopping*. With early stopping, a measure of out-of-sample prediction accuracy is monitored along with the value of the in-sample objective function (7.3.1). Rather than optimizing until the in-sample objective function is minimized, optimization proceeds until out-of-sample performance appears to start to degrade. By updating parameters based on in-sample fit but stopping based on out-of-sample performance, early stopping helps guard against overfitting.

4: These details are outside of the scope of this monograph. Interested readers might refer to *Deep Learning* by Goodfellow, Bengio, and Courville [3] for a textbook treatment of these issue. A popular method for training neural networks is called Adam; see [this Towards Data Science blog](#) for a detailed explanation [4].

As can be seen from the preceding paragraphs, using neural networks in practice relies on the choice of many tuning parameters. As there is relatively little theoretical guidance on these choices, tuning parameters are typically chosen using data splitting. An important choice that clearly relates to model flexibility is the number of neurons and neuron layers when considering the deeper networks discussed below. Having more neurons or layers gives us additional flexibility, just like having more constructed regressors provides more flexibility in high-dimensional linear models. Other choices about regularization then interact with the choice of how many neurons and layers to use in preventing overfitting.

To visualize the working of a neural network, we rely on a resource called [playground.tensorflow.org](http://playground.tensorflow.org) [5], with which we produced a prediction regression model using a simple single layer neural network model based on two input variables. A screenshot taken after training the model is shown below.



The network depicts the process of taking raw regressors and transforming them into predicted values. In the second column (labeled “FEATURES”), we see the inputs – our two raw regressors. The third column depicts a “hidden layer” made up of eight neurons.<sup>5</sup> Each neuron is constructed as a (weighted) linear combination of the raw regressors transformed by an activation function. Here we use the ReLU activation function. The neurons are connected to the inputs and the connections represent the  $\hat{\alpha}_m$  coefficients. The coloring represents the sign of the coefficients (orange is negative and blue positive) and the

5: “Hidden” refers to the fact that these layers are typically not reported. However, these layers can be extracted and used as technical regressors for other tasks. We discuss using hidden layers as features in Chapter 10 which deals with feature engineering.

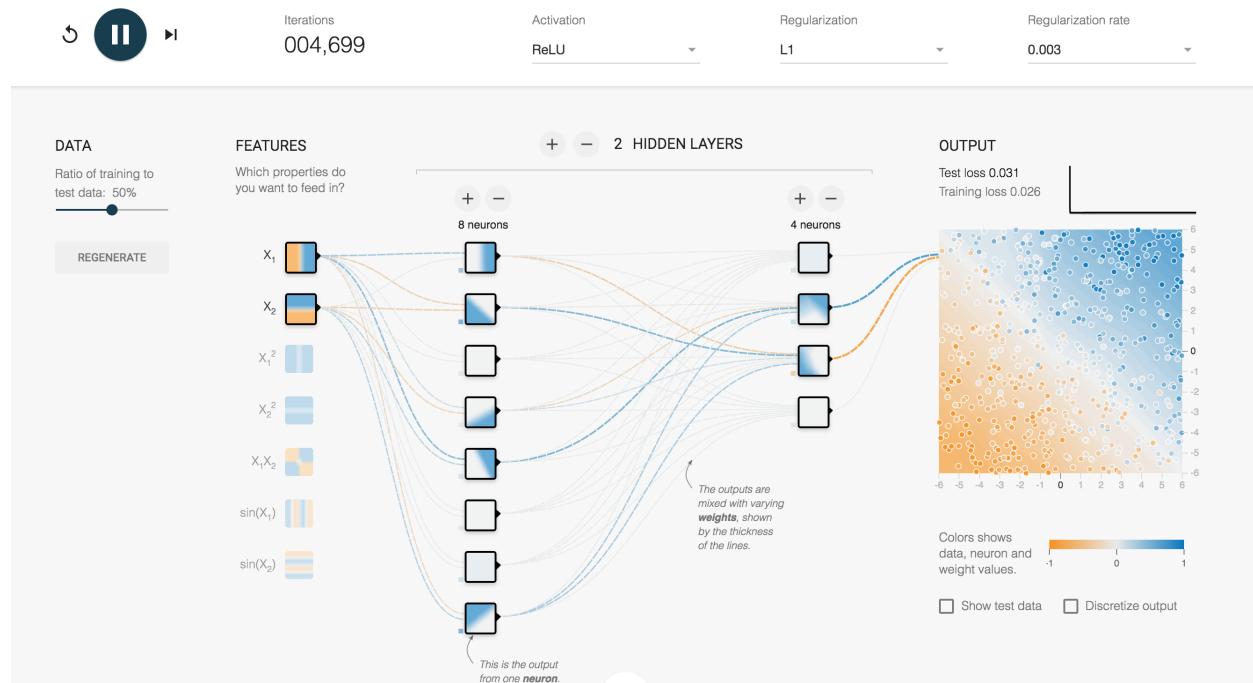
width of the connections represents the size of the coefficients.

Finally, the neurons are combined linearly to produce the output – the prediction rule. The connections going outwards from the neurons to the output represent the coefficients  $\hat{\beta}_m$  of the linear combination of the neurons that produce the final output. The coloring and the width again represent the sign and the size of these coefficients.

The output (prediction) is shown here by the “heat” map in the box on the right. On the horizontal and vertical axes we see the values of the two inputs. The color and its intensity in the “heat” map represent the predicted value.

At the top of the screenshot, we also see that we used “L1” for the type of regularization, which corresponds to using the Lasso type penalty. Here, the penalty level is called the regularization rate and is provided as the last entry in the top line of the screenshot.

In this example, we used a single layer neural network. If we add one or two additional layers of neurons constructed from the previous layer of neurons we get a “deep” network. We illustrate a two-layer network in the following figure.

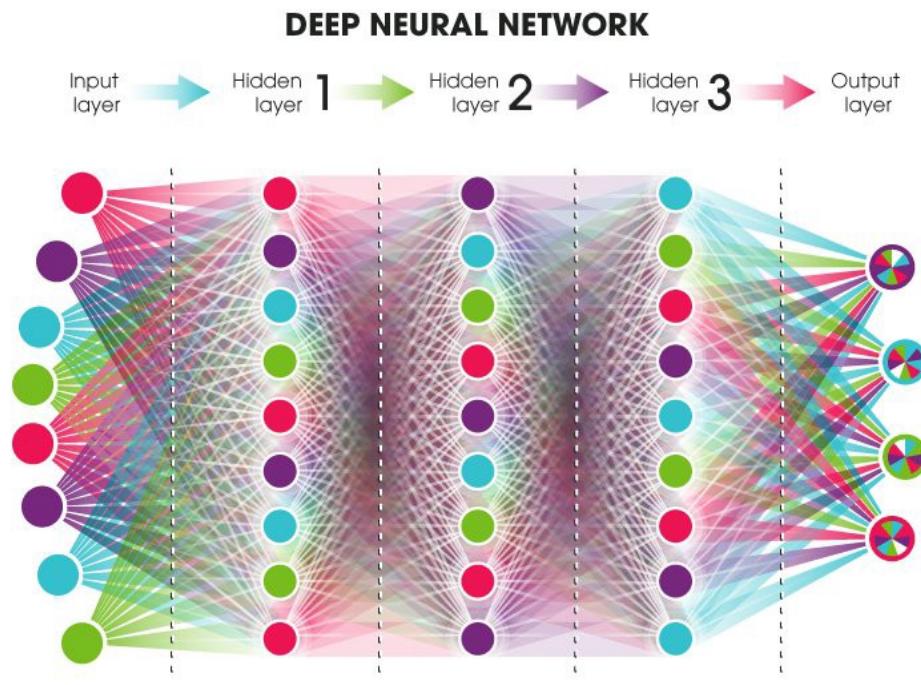


Prediction methods based on neural networks with several layers of neurons are called “deep learning” methods.

## Deep Neural Networks

Here, we present the structure of a neural network with general depth. Networks with depth greater than one are called deep neural networks (DNN).

In addition, for the sake of generality, we consider networks of the multitask form, where we try to predict multiple outputs  $Y^t$ ,  $t = 1, \dots, T$ , where  $t$  stands for the "task".\* A typical scenario is to just have one task,  $T = 1$ , as in our preceding discussion, but there are many cases where we can use a single DNN to solve multiple tasks.



neuralnetworksanddeeplearning.com - Michael Nielsen, Yoshua Bengio, Ian Goodfellow, and Aaron Courville, 2016.

**Figure 7.11:** Standard Architecture of a Deep Neural Network as depicted in Nielsen [6]. The input is mapped nonlinearly into the first hidden layer of the neurons. The output of this first mapping is then mapped nonlinearly into the second layer. This process is then repeated  $m$  times. The output of the penultimate layer is finally mapped (linearly or nonlinearly) into the output layer, which can have multiple outputs corresponding to different tasks.

The general nonlinear regression model we work with takes the form

$$Z \xrightarrow{f_1} H^{(1)} \xrightarrow{f_2} \dots \xrightarrow{f_m} H^{(m)} \xrightarrow{f_{m+1}} \{X^t\}_{t=1}^T, \quad (7.3.2)$$

\* For example, we might be interested in predicting the price of a product using product characteristics across multiple markets or time periods,  $t$ . In treatment effect analysis, we may build a single neural network to predict both the outcome,  $Y$ , and the treatment,  $D$ , using other covariates. We could view this as a multitask learning problem where we are interested in two outputs,  $Y^1 = Y$  and  $Y^2 = D$ .

where

$$H^{(\ell)} = \{H_k^{(\ell)}\}_{k=1}^{K_\ell}$$

are called neurons,  $Z$  is the original input, and the map  $f_\ell$  maps one layer of neurons to the next:

$$f_\ell : v \mapsto \{H_k^{(\ell)}(v)\}_{k=1}^{K_\ell} := (1, \{\sigma_{k,\ell}(v' \alpha_{k,\ell})\}_{k=2}^{K_\ell}), \quad (7.3.3)$$

where  $\sigma_{k,\ell}$  is the activation function that can vary with the layer  $\ell$  and across neurons  $k$  in a given layer. We always include a constant of 1 as a component of  $Z$ , and we always designate one of the neurons in each layer up to  $m$  to be 1. The final layer,  $f_{m+1}$ , does not output the constant of 1 as a component.<sup>6</sup>

$$f_{m+1} : v \mapsto \{X^t(v)\}_{t=1}^T := (\{\sigma_{t,m+1}(v' \alpha_{t,\ell})\}_{t=1}^T). \quad (7.3.4)$$

The network mapping (7.3.2) consists of repeated composition of nonlinear mappings. This structure has been shown to be an extremely powerful tool for generating flexible functional forms which yields successful approximations in a wide range of empirical problems and is backed by approximation theory. Good approximations can be achieved by both considering sufficiently many neurons and sufficiently many layers (Yarotsky, 2017 [7]; Schmidt-Hieber, 2020 [8]; Farrell et. al, 2021 [9]; Kidger and Lyons, 2020 [10]). In empirical economic examples, it is common to just use a few hidden layers, while much deeper networks are typically used in image processing and text applications.

Similarly to single layer neural networks, the DNN model can be trained by minimizing the loss function

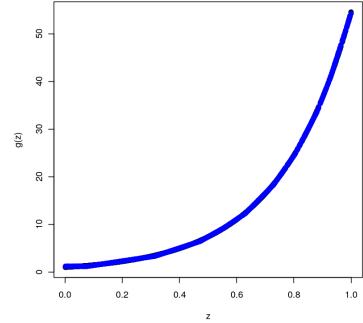
$$\min_{\eta \in \mathcal{N}} \sum_t \sum_i (Y_i^t - X_i^t(\eta))^2 w_t + \text{pen}(\eta; \lambda), \quad (7.3.5)$$

where  $\eta$  denotes all of the parameters of the mapping

$$Z_i \mapsto X_i^t(\eta),$$

$w_t$  denotes the weight given to a task  $t$ , and  $\text{pen}(\eta; \lambda)$  is a penalty function with  $\lambda$  denoting the penalty level.

6: Common architectures employ activation functions that do not vary with  $k$ . However, custom architectures, such as ResNet50 discussed in Figure 7.13, can be viewed as having an activation function that depends on  $k$ , with some neurons linearly activated and some non-linearly.



**Figure 7.12:** Approximation of  $g(Z) = \exp(4Z)$  by a Neural Network

## 7.4 Prediction Quality of Modern Nonlinear Regression Methods

The best prediction rule for an outcome  $Y$  using features/regressors  $Z$  is the function  $g(Z)$ , equal to the conditional expectation

of  $Y$  using  $Z$ :

$$g(Z) = E(Y | Z).$$

Modern Nonlinear Regression Methods, when appropriately tuned and under some regularity conditions, provide estimated prediction rules  $\hat{g}(Z)$  that approximate the best prediction rule  $g(Z)$  well.

Theoretical work demonstrates that under appropriate regularity conditions and with appropriate choices of tuning parameters, the mean squared approximation error of prediction rules produced by modern nonlinear regression methods is small once the sample size  $n$  is sufficiently large, namely,

$$\|\hat{g} - g\|_{L^2(Z)} = \sqrt{E_Z(\hat{g}(Z) - g(Z))^2} \rightarrow 0, \quad \text{as } n \rightarrow \infty,$$

where  $E_Z$  denotes the expectation taken over  $Z$ , holding everything else fixed. To deliver these guarantees in high-dimensional settings where the number of features is large, we rely on structured assumptions, such as sparsity in the case of Lasso. Under these conditions we expect that the in-sample MSE and  $R^2$  would agree with the out-of-sample MSE and  $R^2$ .

## Learning Guarantees of DNNs

We say that a function  $g : \mathbb{R}^d \rightarrow \mathbb{R}$  is  $\beta$ -smooth if it has  $\beta \geq 1$  continuous and uniformly bounded derivatives.<sup>7</sup> If the regression function  $g$  is only known to be  $\beta$ -smooth, then the best estimator of this function has estimation error, in the worst case, that converges at the rate

$$n^{-\beta/(2\beta+d)},$$

as shown by Charles Stone [11]. When  $d$  is not small, this rate of convergence is extremely slow, suggesting that learning a function in  $d$  variables is difficult if the dimension  $d$  is moderate and the target function is only known to be  $\beta$ -smooth.

We can achieve better rates of convergence if there are some kind of structured sparsity or parsimony assumptions as we saw in the rates for high-dimensional linear models in Chapter 3. DNNs are able to take advantage of a nonlinear form of these assumptions that we formulate below following Schmidt-Hieber [8].

<sup>7</sup>: A more general definition allows  $\beta$  to be non-integer, but we focus on integer  $\beta$  for simplicity.

**Assumption 7.4.1** (Structured Sparsity of Regression Function). We assume that  $g$  is generated as a composition of  $q + 1$  vector-valued functions:

$$g = f_q \circ \dots \circ f_0$$

where the  $i$ -th function  $f_i$

$$f_i : \mathbb{R}^{d_i} \rightarrow \mathbb{R}^{d_{i+1}},$$

has each of its  $d_{i+1}$  components  $\beta_i$ -smooth and depends only on  $t_i$  variables, where  $t_i$  can be much smaller than  $d_i$ .

The rate guarantee will depend on the parsimony/smoothness pairs:

$$(t_i, \beta_i), \quad i = 0, \dots, q.$$

For example, consider  $g : \mathbb{R}^{100} \mapsto \mathbb{R}$ ,

$$g(x_1, x_2, x_3, x_4, \dots, x_{100}) = f_1(f_{01}(x_3), f_{02}(x_2)).$$

Then

$$g_0 = f_1 \circ f_0; \quad d_0 = 100, d_1 = 2; \quad t_0 = 1, t_1 = 2.$$

**Theorem 7.4.1** (Learning Guarantee for DNNs under Approximate Sparsity) Suppose that (a) the regression function  $g$  obeys the structured sparsity assumption (Assumption 7.4.1); (b) the depth of the DNN model is proportional to  $\log n$ , (c) the width of the DNN model is no less than

$$s \cdot \log n$$

where  $s$  is the effective dimension of the regression function  $g$ ,

$$s := \max_{i=0, \dots, q} n^{\frac{t_i}{2\beta_i+t_i}};$$

and (d) other regularity conditions hold as specified in [8]. Then, there exists a sparse DNN estimator  $\hat{g}$  with order  $s \log n$  non-zero parameters such that, with probability approaching 1,

$$\|\hat{g} - g\|_{L^2(Z)} \leq \text{const}_F \sigma \sqrt{\frac{s}{n}} \text{polylog}(n),$$

where  $\text{polylog}(n)$  is a polynomial in  $\log(n)$ ,  $\sigma^2 = E(Y - g(Z))^2$ ,

and  $\text{const}_F$  is a constant that depends on the distribution of the data  $F$ .

This fundamental result is due to Schmidt-Hieber [8], where the reader may find the complete statement of regularity conditions and further technical details of the result.

In the example above, despite the high-dimensional setting,  $d = 100$ , if  $f_{01}, f_{02}, f_{11}$  are  $\beta$ -smooth with  $\beta \geq 2$ , a sparse DNN is able to achieve the rate (ignoring logs):

$$\sqrt{\frac{s}{n}} = n^{-\beta/(2\beta+2)} \leq n^{-1/3}$$

where the effective dimension is

$$s = n^{\frac{2}{2\beta+2}}.$$

## Learning Guarantees of Trees and Forests

One important property of adaptively built trees is that they are able to identify the relevant dimensions along which the regression function varies. To isolate this type of behavior of trees and forests, we consider a setting where all the regressors are binary, i.e.  $Z \in \{0, 1\}^d$ . This is without loss of generality for categorical (discrete-valued) regressors, since each level of the regressor can be coded as a binary indicator.<sup>8</sup>

Without further assumptions on the regression function  $g : \{0, 1\}^d \rightarrow \mathbb{R}$ , the best convergence rates that one could hope for scale at least at a  $\sqrt{2^d/n}$  rate. Even for a moderate number of variables  $d$ , this rate of convergence can be prohibitively slow.

Adaptively built trees are particularly successful when there is only a small subset  $S$ , of size  $|S| = r$ , among the  $d$  variables that is relevant. Using this principle, we can formulate a non-parametric analogue of the sparsity assumption that we analyzed in the case of high-dimensional linear regression with lasso that allows us to improve on the convergence rate obtained without restrictions.

<sup>8</sup>: Continuous regressors can also be discretized. However, discretization entails some loss of generality, and approximation properties following discretization have not been formally investigated.

**Assumption 7.4.2** (Non-Parametric Sparsity of a Regression Function with Binary Regressors). *We assume that there exists a subset  $S$  of size  $|S| = r$ , such that the function  $g$  can be written as a function of only the variables in  $S$ ; i.e. we can write*

$$g(Z) = f(Z_S)$$

where  $Z_S$  is the subvector of  $Z$  containing only the coordinates in  $S$ .

The assumption can probably be relaxed to "approximate" sparsity.<sup>9</sup>

Observe that, unlike the sparsity assumption we made in the case of high-dimensional penalized linear regression, Assumption 7.4.2 imposes no restrictions on the form of the function  $f$  that takes as input the relevant variables. Here, under the non-parametric sparsity assumption together with several other regularity conditions, we can prove that the mean squared approximation error of shallow regression trees or "honest" and arbitrarily deep regression forests<sup>10</sup> scales at a

$$\sqrt{2^r \log(d) \log(n)/n}$$

rate. Thus, the convergence rate only depends on the sparsity level  $r$ , and not on the overall number of regressors  $d$ . Moreover, even if we knew the relevant variables  $S$ , we could not hope for a rate faster than  $\sqrt{2^r/n}$  since we make no further assumptions on the function  $f$ . Thus not knowing the relevant set of regressors  $S$  adds an extra multiplicative cost on the achievable rate that only grows logarithmically with the number of regressors and the sample size. See [12] for results of similar flavor for variants of regression trees in settings beyond the binary regressor case.

9: This relaxation has not been formally investigated.

10: An "honest" training approach makes use of subsampling. See Theorem 7.4.3 and the discussion immediately preceding its statement.

**Theorem 7.4.2** (Learning Guarantee for Shallow Regression Trees) Suppose that (a) the regressors are binary and the outcome variable is bounded; (b) the regression function  $g$  obeys Assumption 7.4.2; (c) regularity conditions hold that lower bound the density of the support of the distribution of covariates and upper bound the degree of variance reduction in MSE that can be achieved by features not in  $S$  [13]. Then a regression tree estimator  $\hat{g}$ , where the regression tree is greedily grown based on the MSE criterion up to a depth that is at least  $r$  and at most some constant multiple of  $r$ , satisfies, for  $n \geq \text{const}_F 2^r \log(d/\delta)$ , with probability  $1 - \delta$ ,

$$\|\hat{g} - g\|_{L^2(Z)} \leq \text{const}_F \sigma \sqrt{\frac{2^r \log(d/\delta) \log(n)}{n}},$$

where  $\sigma^2 = E(Y - g(Z))^2$  and  $\text{const}_F$  is a constant that depends on the distribution of the data  $F$ .

A greedy algorithm is any algorithm that follows the problem-solving heuristic of making the locally optimal choice at each stage. In our case, a greedily grown tree optimizes over the name of regressor and splitting point that achieve the best one-step improvement in the in-sample MSE at each node.

Capping the depth of the regression tree as in Theorem 7.4.2 helps avoid overfitting, since otherwise we could potentially construct binary trees that achieve zero error on the training data and have large error out-of-sample.

An alternative to avoiding overfitting is to use an ensemble approach based on sub-sampled data. To implement an ensemble approach, we train multiple regression trees, each on a random sub-sample (without replacement) of the original data-set of size  $s < n$  and average the predictions of each of these trees. Moreover, to formally argue about the approximation error of such sub-sampled forests, we will require the forests to be trained in an “honest” manner.

In our setting, an honest training approach is as follows: When we train a tree on a sub-sample, we randomly partition the data in half and we use half of the data to find the best splits in a greedy manner, and the other half of the data to construct the estimates at each node of the tree. Such sub-sampled honest forests have been recently introduced and popularized by the work of [14]. Subsequent work of [13] showed that honest forests provably adapt to non-parametric sparsity of the regression function.

**Theorem 7.4.3** (Learning Guarantee for Sub-Sampled Honest Forests) *Suppose that (a) the regressors are binary and outcome variable is bounded; (b) the regression function  $g$  obeys Assumption 7.4.2; (c) regularity conditions hold that lower bound the density of the support of the distribution of covariates and upper bound the degree of variance reduction in MSE that can be achieved by features not in  $S$  [13]. Then a regression forest estimator  $\hat{g}$ , where each regression tree is built in an honest manner and on a random sub-sample (without replacement) of size  $s = \text{const}_F 2^r \log(d/\delta)$  of the original data, satisfies, for  $n \geq \text{const}_F 2^r \log(d/\delta)$  with probability  $1 - \delta$ ,*

$$\|\hat{g} - g\|_{L^2(Z)} \leq \text{const}_F \sigma \sqrt{\frac{2^r \log(d/\delta) \text{polylog}(n)}{n}}$$

where  $\sigma^2 = E(Y - g(Z))^2$  and  $\text{const}_F$  is a constant that depends on the distribution of the data  $F$  and  $\text{polylog}(n)$  is a polynomial factor of  $\log(n)$ .

The rate guarantee for Honest Forests in Theorem 7.4.3 is the same as the rate for shallow trees in Theorem 7.4.2. This theory thus does not shed light on why Random Forests seem to achieve superior predictive performance over simple trees in many applications. Moreover, practical Random Forest algorithms tend to work well with default tuning choices, whereas the theory requires a careful alignment of the tuning parameters to get good rate guarantees. The regularity conditions also require the explanatory power of the subset of the covariates that are

relevant,  $S$ , to dominate the explanatory power of the irrelevant covariates.<sup>11</sup> This condition on signal strength is a sufficient condition, but it may not be necessary for good performance. That is, there seem to remain substantial gaps in our theoretical understanding of the performance of tree-based algorithms. Further exploring these properties may be an interesting area for further study.

<sup>11</sup>: Irrelevance here only means that, given the set  $S$  of relevant covariates, the other variables do not contribute to the best prediction rule. It does not mean that the irrelevant covariates have no predictive power on their own.

## Trust but Verify

Both tree-based methods and neural networks provide powerful, flexible models that can deliver high-quality approximations of regression functions. However, the high degree of flexibility can lead to overfitting. Therefore, it is always important to verify the performance on test data to make sure that the predictive model being used is actually a good one.

A simple verification procedure is data splitting, which can be performed in the following way:

1. We use a random subset of data for estimating/training the prediction rule.
2. We use the other part of the data to evaluate the quality of the prediction rule, recording out-of-sample mean squared error,  $R^2$ , or some other desired measure of prediction quality.

Recall that the part of the data used for estimation is called the training sample. The part of the data used for evaluation is called the testing or validation sample. We have a data sample containing observations on outcomes  $Y_i$  and features  $Z_i$ . Suppose we use  $n$  observations for training and  $m$  for testing/-validation. We use the training sample to compute prediction rule  $\hat{g}(Z)$ . Let  $V$  denote the indices of the observations in the test sample. Then the out-of-sample/test mean squared error is

$$\text{MSE}_{test} = \frac{1}{m} \sum_{k \in V} (Y_k - \hat{g}(Z_k))^2.$$

The out-of-sample/test  $R^2$  is

$$R^2_{test} = 1 - \frac{\text{MSE}_{test}}{\frac{1}{m} \sum_{k \in V} Y_k^2}.$$

## A Simple Case Study using Wage Data

We illustrate ideas using a data set of 5150 observations from the March Current Population Survey Supplement 2015.  $Y_i$ 's are log wages of never-married workers living in the U.S.  $Z_i$ 's include experience, education, 23 industry and 22 occupation indicators, and some other characteristics. We consider a variety of linear and nonlinear rules for predicting  $Y$  with  $Z$ .

For the linear models, we estimate prediction rules of the form  $\hat{g}(Z) = \hat{\beta}'X$  using  $X$  generated in two ways:

- ▶ (basic model)  $X$  consists of the 51 raw regressors in  $Z$ .
- ▶ (flexible model)  $X$  consists of 246 variables composed of the 51 raw regressor in  $Z$ , a fourth order polynomial in experience, and two-way interactions between the polynomial terms in experience and the non-experience variables in  $Z$ .

We estimate  $\hat{\beta}$  by linear regression/least squares and by the following penalized regression methods: Lasso and Post-Lasso with plug-in choice of  $\lambda$ , cross-validated Lasso, Ridge, and Elastic Net.

For the nonlinear models, we estimate prediction rules of the form  $\hat{g}(Z)$  without imposing that  $\hat{g}(Z) = \hat{\beta}'X$ . That is, we do not assume prediction rules to be linear. We estimate the prediction models by Random Forests, Regression Trees, Boosted Trees, and Neural Networks. We use an implementation of the Random Forest where, at the step of growing a regression tree, we choose the best variable to split upon among  $\sqrt{p} \ll p$  randomly selected variables.

Table 7.1 displays results based upon a single split of data into training and testing sets. It shows the test MSE in column 1, the standard error in column 2 and the test  $R^2$  in column 3. We see that the best performing prediction rules are provided by OLS using the raw 51 regressors and Lasso using the basic 51 predictors with penalty parameter selected by cross-validation. The performance of both Elastic Net with the basic set of regressors and boosted trees are also nearly identical to those of the two best methods. Looking at standard errors, we see that the vast majority of methods have test MSE's that are within one standard error of the best test MSE, suggesting relatively little difference in performance across methods.

The outliers, in terms of performing relatively poorly, are OLS using the flexible set of covariates as well as the regression tree and the neural net. OLS with the flexible set of predictors uses

	MSE	S.E.	$R^2$
Least Squares (basic)	0.229	0.016	0.282
Least Squares (flexible)	0.243	0.016	0.238
Lasso	0.234	0.015	0.267
Post-Lasso	0.233	0.015	0.271
Lasso (flexible)	0.235	0.015	0.265
Post-Lasso (flexible)	0.236	0.016	0.261
Cross-Validated Lasso	0.229	0.015	0.282
Cross-Validated Ridge	0.234	0.015	0.267
Cross-Validated Elastic Net	0.230	0.015	0.280
Cross-Validated Lasso (flexible)	0.232	0.015	0.275
Cross-Validated Ridge (flexible)	0.233	0.015	0.271
Cross-Validated Elastic Net (flexible)	0.231	0.015	0.276
Random Forest	0.233	0.015	0.270
Boosted Trees	0.230	0.015	0.279
Pruned Tree	0.248	0.016	0.224
Neural Net	0.276	0.012	0.148

**Table 7.1:** Prediction Performance for the Test/Validation Sample.

a relatively large number of variables relative to the sample size and seems likely to be overfit. On the other hand, neither the regression tree nor the neural net is fully tuned. Thus, there may be room to improve the performance of these methods.

## 7.5 Combining Predictions - Aggregation - Ensemble Learning

Given different prediction rules, we can choose either a single method or an aggregation of several methods as our prediction approach. An aggregated prediction is a linear combination of the basic predictors.

Specifically, we consider an aggregated prediction rule of the form:

$$\tilde{g}(Z) = \sum_{k=1}^K \tilde{\alpha}_k \hat{g}_k(Z),$$

where  $\hat{g}_k$ 's denote basic predictors, potentially including a constant. The basic predictors are computed on the training data.

If the number of prediction rules,  $K$ , is small, we can figure out the coefficients of the optimal linear combination of the rules,  $\tilde{\alpha}_k$ , using test data  $V$  by simply running least squares of the outcomes in the test data on their associated predicted

In econometrics and statistics, the procedures for combining several methods are called "model averaging" and "aggregation". In machine learning, these terms are relabeled as "ensembles" and "stacking".

values:

$$\min_{(\alpha_k)_{k=1}^K} \sum_{i \in V} (Y_i - \sum_{k=1}^K \alpha_k \hat{g}_k(Z_i))^2.$$

We wish to emphasize that here we are minimizing the sum of squared prediction errors in the test sample using the prediction rules from the training sample as the regressors. If  $K$  is large, we can instead use Lasso for aggregation:

$$\min_{(\alpha_k)_{k=1}^K} \sum_{i \in V} (Y_i - \sum_{k=1}^K \alpha_k \hat{g}_k(Z_i))^2 + \lambda \sum_{k=1}^K |\alpha_k|.$$

### Aggregation Results for the Case Study

We consider the prediction rules based on OLS, Post-Lasso, Elastic Net, Pruned Tree, Random Forest and Boosted Trees to build an ensemble method.

	Weight OLS	Weight Lasso
Constant	-0.162	-0.147
Least Squares (basic)	0.281	0.293
Post-Lasso (flexible)	0.237	0.223
CV elnet (flexible)	-0.068	-0.056
Pruned Tree	-0.140	0.000
Random Forest	0.377	0.344
Boosted Trees	0.367	0.245

**Table 7.2:** Weights of the ensemble method.

The estimated weights are shown in Table 7.2. The adjusted  $R^2$  for the test sample gets improved by about 1%.

### Auto ML Frameworks

There are a variety of new frameworks emerging that do automated search and aggregation of different prediction methods. These automatic aggregation procedures use approaches like the one we outlined above or other heuristics. Examples of automatic aggregation methods include H20, AutoML [15], and Auto Gluon [16] (which relies on Neural Nets).

We've tried H20 on the wage data. It produced a model that beats OLS with the basic predictor set, which gave a test MSE of 0.229, by producing a test MSE of

0.21.

(The difference is not statistically significant.) H20 is similar to the ensemble method that we constructed above. The performance was very impressive because we gave H20 a time budget of just 100 seconds!

## 7.6 When do Neural Networks win?

The wage example may give a pessimistic impression on the power of deep learning (and machine learning more generally). A more optimistic impression emerges from examining performance of deep learning in data-rich settings, where large samples and rich features are available.

A recent example comes from Bajari et al. (2021) [17]. Here we are interested in predicting prices of products given their characteristics, which include both text and images. The resulting predictions are called hedonic prices. In this example, neural networks (specifically BERT and ResNet50) are first used to convert the text and image data into several thousand-dimensional numerical features  $X$  (called embeddings). These features extracted from the text and image data are then used as input variables in a deep neural network for predicting product prices. The deep neural network used in the example consists of 3 hidden layers, with the penultimate layer consisting of about 400 neurons.

The data set used in this example is larger than 10 million observations. The accuracy of prediction for the deep neural network described above, as measured by the  $R^2$  on the test sample, is about

90%.

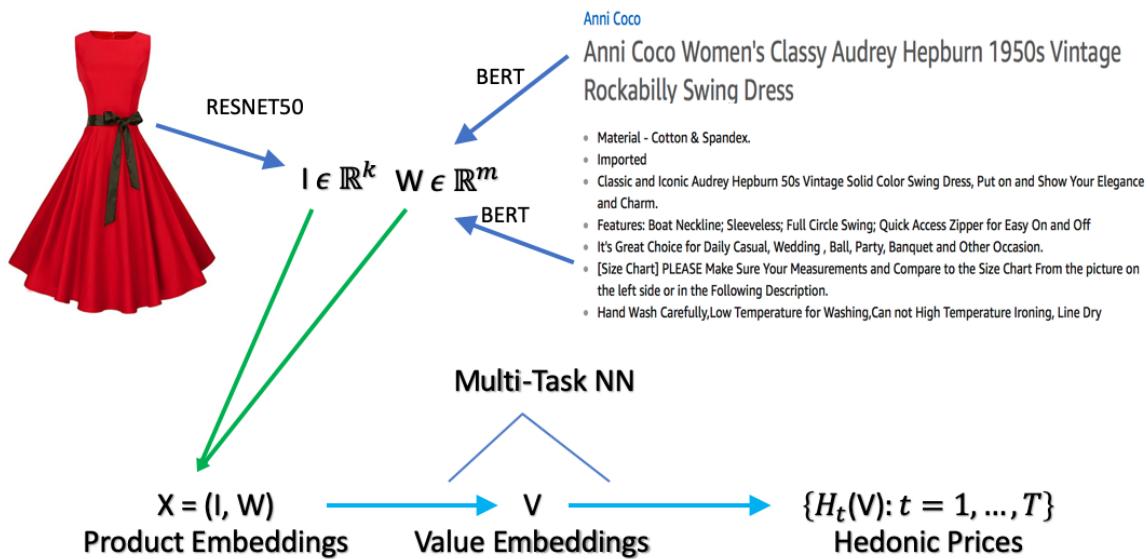
In contrast, random forests applied to predict prices using the text and image embeddings as inputs deliver an  $R^2$  in the test sample that is in the ballpark of 80%, and a linear model estimated via least squares that uses the text and image embeddings as predictor variables delivers an  $R^2$  in the test sample of only around 70%. Ignoring the neural network embeddings of the text and image data and using only simple catalog features, the  $R^2$  is lower than

40%.

We will discuss details of generating embeddings in Chapter 10.

To sum up, we have discussed assessment of predictive performance of modern linear and non-linear regression methods

The features produced in the penultimate layer in a deep neural network are often referred to as embeddings as they encode or “embed” the information from the previous layers that is directly used in producing the final predictions. In the case of hedonic pricing, we may refer to these features as “value embeddings” as the final target is price or value of the product.



**Figure 7.13:** The structure of the predictive model in Bajari et al. (2021) [17]. The input consists of images and unstructured text data. The first step of the process creates the moderately high-dimensional numerical embeddings  $I$  and  $W$  for images and text data via state-of-the-art deep learning methods, such as ResNet50 and BERT. The second step of the process takes as input  $X = (I, W)$  and creates predictions for hedonic prices  $H_t(X)$  using deep learning methods with a multi-task structure. The models of the first step are trained on tasks unrelated to predicting prices (e.g., image classification or word prediction), where embeddings are extracted as hidden layers of the neural networks. The models of the second step are trained by price prediction tasks. The multitask price prediction network creates an intermediate lower dimensional embedding  $V = V(X)$ , called value embedding and then predicts the final prices in all time periods  $\{H_t(V), t = 1, \dots, T\}$ . Some variations of the method include fine-tuning the embeddings produced by the first step to perform well for price prediction tasks (i.e. optimizing the embedding parameters so as to minimize price prediction loss).

using splitting of data into training and testing samples. The results could be used to pick the best prediction rule generated by the classical or modern regression methods or to aggregate prediction rules into an ensemble rule, which can result in some improvements. We illustrated these ideas using the wage data from the 2015 Current Population Survey. We finally introduced Auto ML frameworks and commented that Neural Networks perform best in very data-rich settings.

## Notebooks

- ▶ [R Notebook on ML-based Prediction of Wages](#) provides details of implementation of penalized regression, regression trees, random forest and boosted tree methods, a comparison of various methods and a way to choose the best method or create an ensemble of methods.
- ▶ [R Notebook on Deep Neural Network Prediction of Wages](#) provides details for implementing a simple deep neural

network in the wage prediction problem.

- ▶ [R Notebook on AutoML Prediction of Wages](#) provides an application of the H2O AutoML framework to the wage prediction problem. With a time budget of only 100 seconds, H2O found the model that worked best for predicting wages.
- ▶ [R Notebook on Approximation of a Function by Random Forest and Neural Network](#) illustrates the flexibility of these methods in approximating the function  $\exp(4x)$ .

## Additional resources

- ▶ Andrej Karpathy [18] 's [Recipe for Training Neural Networks](#) provides a good workflow and practical tips for training good neural network models.
- ▶ For practical details of tree-based methods, please see Hastie et al. [19] 's book "[Introduction to Statistical Learning](#)".
- ▶ For an in-depth treatment of deep learning, see Zhang's et al. [20] 's book "[Dive Into Deep Learning](#)", Goodfellow et al. [3] "[Deep Learning](#)", and Nielsen [6] "[Neural Networks and Deep Learning](#)".

## Notes

Many of the formative developments in modern nonlinear regression were led by the statistics and artificial intelligence communities. The methods were rebranded as machine learning in the 90s, and learning with neural networks was rebranded as deep learning when it was realized that deep network architectures produced phenomenal results in image classification (and later in natural language processing tasks). The success of deep neural networks was a breakthrough associated with advances in both computing power and the ability to collect very large data sets. See the textbooks mentioned above for in-depth treatments of deep learning.

In Chapter 9, we will study the use of the machine learning and deep learning for statistical inference on causal and predictive effects in high-dimensional nonlinear regression settings; and in Chapter 10, we'll be using deep learning for engineering

features from text and data (e.g. using images and product descriptions as "regressors").

## Study Problems

1. Use two paragraphs to explain to a friend how one of the tree-based strategies works.
2. Use two paragraphs to explain to a friend how a basic neural network works.
3. Experiment with one of the empirical notebooks provided and summarize your findings. For example, try to see if you can build a better performing neural network in the wage example. One possibility is to use [custom models in Keras](#), where we can construct a partially linear model that borrows the strength of the basic linear model and corrects it slightly with a nonlinear deviation function.
4. Experiment with the last (non-empirical) notebook. See, for example, if you can find a (much) simpler neural network that provides the same quality of fit as the current example in the notebook.

## 7.A Variable Importance via Permutations

There are many ways of assessing variable importance in non-linear models. A very simple one is the following permutation method.

The importance of variable  $j$  in any machine learning algorithm (linear or nonlinear) can be defined by computing the loss in predictive performance that results from replacing the observations of the  $j$ -th feature  $(Z_{ij})_{i=1}^n$  with their random permutation

$$(Z_{\pi(i)j})_{i=1}^n,$$

where  $\pi : \{1, \dots, n\} \rightarrow \{1, \dots, n\}$  is a permutation map, generated at random. The loss is averaged over many random permutations, to obtain an average loss measure  $L_j$ . Then the variables are ranked in terms of  $L_j$ , from largest to smallest. The top-ranked variables are the most important ones. This idea, that appeared in the original paper by L. Breiman [2], mimics the situation where the permuted regressor is an irrelevant

predictor having the same marginal distribution as the observed regressor.

# Bibliography

- [1] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*. Vol. 1. Springer series in statistics New York, 2001 (cited on page 131).
- [2] Leo Breiman. 'Random forests'. In: *Machine learning* 45.1 (2001), pp. 5–32 (cited on pages 132, 153).
- [3] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. <http://www.deeplearningbook.org>. MIT Press, 2016 (cited on pages 136, 152).
- [4] Lili Jiang. *A Visual Explanation of Gradient Descent Methods (Momentum, AdaGrad, RMSProp, Adam)*. 2020. URL: <https://towardsdatascience.com/a-visual-explanation-of-gradient-descent-methods-momentum-adagrad-rmsprop-adam-f898b102325c> (visited on 04/03/2022) (cited on page 136).
- [5] *playground.tensorflow.org*. <https://playground.tensorflow.org/>. Accessed: 2022-04-03 (cited on page 137).
- [6] Michael A. Nielsen. *Neural networks and deep learning*. Determination Press, 2015 (cited on pages 139, 152).
- [7] Dmitry Yarotsky. 'Error bounds for approximations with deep ReLU networks'. In: *Neural Networks* 94 (2017), pp. 103–114 (cited on page 140).
- [8] Johannes Schmidt-Hieber. 'Nonparametric regression using deep neural networks with ReLU activation function'. In: *Annals of Statistics* 48.4 (2020), pp. 1875–1897 (cited on pages 140–143).
- [9] Max H. Farrell, Tengyuan Liang, and Sanjog Misra. 'Deep Neural Networks for Estimation and Inference'. In: *Econometrica* 89.1 (2021), pp. 181–213 (cited on page 140).
- [10] Patrick Kidger and Terry Lyons. 'Universal Approximation with Deep Narrow Networks'. In: *Proceedings of Thirty Third Conference on Learning Theory*. Ed. by Jacob Abernethy and Shivani Agarwal. Vol. 125. Proceedings of Machine Learning Research. PMLR, 2020, pp. 2306–2327 (cited on page 140).
- [11] Charles J. Stone. 'Optimal global rates of convergence for nonparametric regression'. In: *The annals of statistics* 10.4 (1982), pp. 1040–1053 (cited on page 141).

- [12] Stefan Wager and Guenther Walther. ‘Adaptive concentration of regression trees, with application to random forests’. In: *arXiv preprint arXiv:1503.06388* (2015) (cited on page 144).
- [13] Vasilis Syrgkanis and Manolis Zampetakis. ‘Estimation and Inference with Trees and Forests in High Dimensions’. In: *Proceedings of Thirty Third Conference on Learning Theory*. Ed. by Jacob Abernethy and Shivani Agarwal. Vol. 125. Proceedings of Machine Learning Research. PMLR, 2020, pp. 3453–3454 (cited on pages 144, 145).
- [14] Stefan Wager and Susan Athey. ‘Estimation and Inference of Heterogeneous Treatment Effects using Random Forests’. In: *Journal of the American Statistical Association* 113.523 (2018), pp. 1228–1242 (cited on page 145).
- [15] Erin LeDell and Sebastien Poirier. ‘H2o automl: Scalable automatic machine learning’. In: *Proceedings of the AutoML Workshop at ICML*. Vol. 2020. 2020 (cited on page 149).
- [16] Nick Erickson, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li, and Alexander Smola. ‘Autogluon-tabular: Robust and accurate automl for structured data’. In: *arXiv preprint arXiv:2003.06505* (2020) (cited on page 149).
- [17] Patrick L. Bajari, Zhihao Cen, Victor Chernozhukov, Manoj Manukonda, Jin Wang, Ramon Huerta, Junbo Li, Ling Leng, George Monokroussos, Suhas Vijaykumar, et al. *Hedonic prices and quality adjusted price indices powered by AI*. Tech. rep. cemmap working paper, 2021 (cited on pages 150, 151).
- [18] Andrej Karpathy. *A Recipe for Training Neural Networks*. 2019. URL: <http://karpathy.github.io/2019/04/25/recipe/> (visited on 04/06/2022) (cited on page 152).
- [19] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*. Springer, 2013 (cited on page 152).
- [20] Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola. ‘Dive into deep learning’. In: URL: <https://d2l.ai> (2020) (cited on page 152).

# Causal Inference via Directed Acyclical Graphs and Nonlinear Structural Equation Models

# 8

"caused; causing: to compel by command, authority, or force". Merriam-Webster Dictionary

Here we explore a fully nonlinear, nonparametric formulation of causal diagrams and associated structural equation models. These provide a general tool for understanding causal identification. We discuss graphical criteria of identification of average causal effects via regression adjustment.

8.1 Introduction . . . . .	158
8.2 From Causal Diagrams to Causal DAGs . . . . .	159
Identification by Regression . . . . .	161
Interventions . . . . .	162
8.3 General Acyclic SEMs, Causal DAGs and Counterfactuals . . . . .	164
DAGs and Acyclic SEMs (ASEMs) . . . . .	164
General definitions . .	165
Testable Restrictions by D-Separation . . . . .	168
8.4 Counterfactuals Induced by Interventions . . . . .	170
8.5 Identification by Conditioning . . . . .	172
Main Idea . . . . .	172
Useful Adjustment Strategies as Corollaries . . . .	175
8.6 Falsifiability and Causal Discovery* . . . . .	179
8.A Counterfactual Distributions via Markov Networks	184
8.B Causal Discovery Algorithms . . . . .	185
PC Algorithm . . . . .	185
FCI Algorithm . . . . .	187

## 8.1 Introduction

The purpose of this module is to provide a more formal and general treatment of acyclic nonlinear (and nonparametric) structural equation models (SEMs) and corresponding causal directed acyclic graphs (CDAGs). We discuss the concepts and identification results provided by Judea Pearl and his collaborators and by James H. Robins and his collaborators.

These models and concepts allow us to rigorously define structural causal effects in fully nonlinear models and obtain their nonparametric identification from the structure of the CDAGs alone. Structural causal effects are defined as hypothetical effects of interventions in systems of equations. We discuss identification of effects of "do-interventions" introduced by Pearl [1] and "fix-interventions" introduced by Heckman and Pinto [3] and Robins and Richardson [4]. They also had appeared as part of do-calculus in Pearl [1]. Fix-interventions induce counterfactual DAGs called SWIGs (Single World Intervention Graphs) and can recover the causal graphs we've seen in previous chapters.

Whether causal effects derived from SEMs approximate policy or treatment effects in the real world depends to a large extent on the degree to which the posited SEM approximates real phenomena. In thinking about the approximation quality of a model, it is important to keep in mind that we will never be able to establish that a model is fully correct using statistical criteria. However, we may be able to reject a given model using formal falsifiability criteria, though not all models are statistically falsifiable, or contextual knowledge. Further, some causal effects can be verified by an explicit randomized controlled trial, though the use of experiments is not an option in many cases. Ultimately, contextual knowledge is often crucial for making the case that a given structural model represents real phenomena sufficiently well to produce credible estimates of causal effects when using observational data.

In 2011, J. Pearl was awarded the A.M. Turing award, the highest award in the field of Computer Science and Artificial Intelligence: "For fundamental contributions to artificial intelligence through the development of a calculus for probabilistic and causal reasoning." In the Biometrika 1995 article [1], J. Pearl presents his work as a generalization of the SEMs put forward by T. Haavelmo [2] in 1944 and others.

### Notation

Consider a pair of random variables (or equivalently, random vectors)  $U$  and  $V$  with joint distribution probability (mass) function  $p_{UV}(u, v)$  at generic evaluation points  $(u, v)$ . We will simply denote  $p_{UV}(u, v)$  by  $p(u, v)$  whenever there is no ambiguity. We will denote the marginal probability (mass) functions by  $p_U(u)$  and  $p_V(v)$ , or simply by  $p(u)$  and  $p(v)$ . The random

variables  $U$  and  $V$  are independent, which we denote as

$$U \perp\!\!\!\perp V,$$

if and only if the joint probability density (or mass) function  $p(u, v)$  can be factorized as

$$p(u, v) = p(u)p(v)$$

or equivalently if and only if

$$\text{E}g(U)\ell(V) = \text{E}g(U)\text{E}\ell(V)$$

for any bounded functions  $g$  and  $\ell$ . This definition of independence implies the ignorability or exclusion results,

$$p(u | v) = p(u), \quad p(v | u) = p(v),$$

which follow from Bayes' law:

$$p(u | v) = \frac{p(u)p(v)}{p(v)}.$$

Conditional independence is defined similarly by replacing distributions and expectations with their conditional analogues.

## 8.2 From Causal Diagrams to Causal DAGs

Formal causal nonlinear DAGs generalize linear parametric models to general nonparametric forms. Recall our previous discussion of a model for a household's log-demand for gasoline ( $Y$ ), which is a function of log-price ( $P$ ) and household characteristics ( $X$ ). We can generalize the simple TSEM to a nonlinear DAG as follows.

**Example 8.2.1** (TSEM) We have a system of triangular structural equations:

$$\begin{aligned} Y &:= f_Y(P, X, \epsilon_Y), \\ P &:= f_P(X, \epsilon_P), \\ X &:= \epsilon_X. \end{aligned} \tag{8.2.1}$$

where  $f$ 's are said to be deterministic structural functions and  $\epsilon_Y, \epsilon_P, \epsilon_X$  are structural shocks that are independent of each other. The dimension of structural shocks is not restricted.

Also note that

$$\epsilon_Y \perp\!\!\!\perp P, X, \quad \epsilon_P \perp\!\!\!\perp X.$$

A causal diagram depicting the algebraic relationship defining the TSEM in Example 8.2.1 is shown in Figure 8.1. The absence of edges between nodes encodes the model's independence restrictions. Thus, as before, we can see that we can view graphs as representations of statistical models. The graph visually depicts independence restrictions and the propagation of information or structural shocks from root nodes to their children, grandchildren, and so forth.

It is also common to draw graphs based on only observed variables. We can erase the latent root nodes from Figure 8.1 to produce the equivalent diagram illustrated in Figure 8.2.

The TSEM is purely a statistical model. We can view this model as structural under invariance restriction, following Haavelmo [2].

**Definition 8.2.1** (Structural Form) *When we say that the TSEM is structural, we mean that it is defined by a structure made up of a set of stochastic processes:*

$$\begin{aligned} Y(p, x) &:= f_Y(p, x, \epsilon_Y), \\ P(x) &:= f_P(x, \epsilon_P), \\ X &:= \epsilon_X, \end{aligned}$$

indexed by  $(p, x) \in \mathcal{P} \times \mathcal{X}$ , called structural functions or structural potential outcome processes. Moreover,

- ▶ (Exogeneity) Stochastic shocks  $\epsilon_P, \epsilon_X$ , and  $\epsilon_Y$  are generated as independent variables outside of the model;
- ▶ (Consistency) The endogenous variables are generated by recursive substitutions:

$$Y := Y(P, X), \quad P := P(X), \quad X := \epsilon_X;$$

- ▶ (Invariance) The structure remains invariant to changes of the distribution of stochastic shocks  $\epsilon$ .

Moreover, the structure will be assumed to be preserved under various interventions as defined below.

While SEMs are statistical models, assumptions akin to those in Definition 8.2.1 endow them with a structural meaning. For

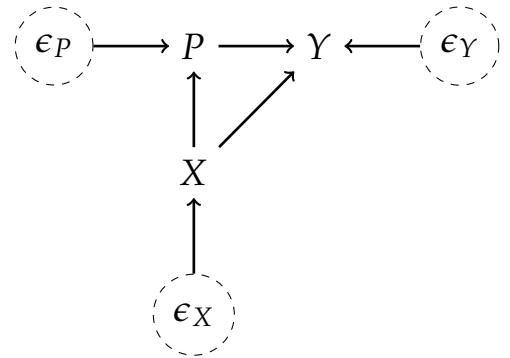


Figure 8.1: The causal DAG equivalent to the TSEM in Example 8.2.1.

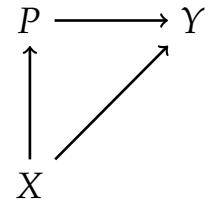


Figure 8.2: The causal DAG corresponding to the TSEM in Example 8.2.1 with latent root nodes erased.

example, structural meaning may be generated by economic reasoning. For example, structural functions may correspond to demand functions, supply functions, and expenditure functions, with these notions going back at least to Marshall [5] in the 19<sup>th</sup> century.

**Remark 8.2.1** (Link to Potential Outcomes) Consider binary  $p \in \{0, 1\}$  for simplicity. Consider potential outcomes, given by the structure:

$$Y(p) := g(p, X, \epsilon_Y(p)).$$

We can view POs through a SEM framework as follows. Let  $\epsilon_Y := \{\epsilon_Y(p) : p \in \{0, 1\}\}$ , then we have that

$$Y(p) = g(p, X, \epsilon_Y(p)) = f_Y(p, X, \epsilon_Y),$$

for

$$f_Y(p, x, e) := 1(p=0)g(p, x, e(0)) + 1(p=1)g(p, x, e(1))$$

for the argument  $e = \{e(p) : p \in \{0, 1\}\}$ . This example emphasizes that the dimensionality of  $\epsilon$ 's is not restricted in the general framework.

## Identification by Regression

By conditioning on  $X = x$  in the graph in Figure 8.1, we obtain the graph shown in Figure 8.3. We can equivalently express the relationship shown in Figure 8.3 in terms of equations as

$$Y(x) = f_Y(P(x), x, \epsilon_Y), \quad \epsilon_Y \perp\!\!\!\perp P(x).$$

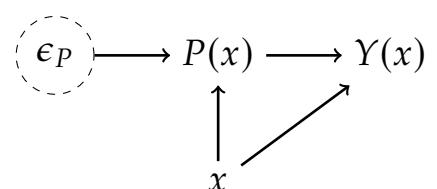
If  $P(x)$  is non-degenerate, we can further condition on  $P(x) = p$  to learn the average structural function

$$\text{Ef}_Y(p, x, \epsilon_Y)$$

via regressions. We formally record this result as follows.

- In the TSEM, the conditional average structural function

$$\text{Ef}_Y(p, x, \epsilon_Y)$$



**Figure 8.3:** The graph produced from Figure 8.1 by conditioning on  $X = x$ . Here  $X$  is a parent to both  $P$  and  $Y$ . After conditioning, the remaining source of variation in  $P(x)$  is  $\epsilon_P$ .  $\epsilon_P$  is determined exogenously – as if by an experiment – which allows measurement of the causal effect  $P(x) \rightarrow Y$ .

can be identified by conditioning on  $P$  and  $X$ :

$$\begin{aligned} \mathbb{E}[Y|P = p, X = x] &= \mathbb{E}[f_Y(P, X, \epsilon_Y)|P = p, X = x] \\ &= \mathbb{E}[f_Y(p, x, \epsilon_Y)|P = p, X = x] \\ &= \mathbb{E}[f_Y(p, x, \epsilon_Y)] \end{aligned}$$

provided the event  $\{P = p, X = x\}$  is assigned positive density.

- ▶ This average structural function has the interpretation as the expected outcome when  $P$  and  $X$  are exogenously set (set outside of the model as if by a policy maker or experiment) to  $P = p$  and  $X = x$ .
- ▶ Hence, we can use the average structural function to provide counterfactual predictions – predictions for the outcome under exogenous assignment of the policy variable  $P$  at fixed values for  $X$ . Within the TSEM, these counterfactual predictions align with the usual prediction rule  $\mathbb{E}[Y|P = p, X = x]$ .
- ▶ If the confounder  $X$  is not observed, the causal relationship  $P(x) \rightarrow Y$  is not identified.

If we can identify the conditional average structural function, we can also identify the conditional average structural causal effect:

$$\begin{aligned} Ef_Y(p_1, x, \epsilon_Y) - Ef_Y(p_0, x, \epsilon_Y) &= \mathbb{E}[Y|P = p_1, X = x] - \mathbb{E}[Y|P = p_0, X = x]. \end{aligned} \quad (8.2.2)$$

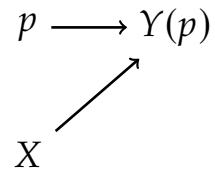
The right hand side of (8.2.2) is a statistical quantity that can clearly be learned from data on  $Y$ ,  $P$ , and  $X$  under reasonable assumptions. The left hand side of (8.2.2) defines a structural quantity of interest: the average effect of exogenously changing  $P$  from  $p_0$  to  $p_1$  at  $X = x$ .

## Interventions

**Do Interventions.** The do-operation  $do(P = p)$  or do-intervention corresponds to creating the counterfactual

graph shown in Figure 8.4. On the graph, we remove  $P$  and replace it with a deterministic node  $p$  instead. In terms of equations (8.2.1) defining the TSEM, we replace the equation for  $P$  with  $p$  and then set  $P$  equal to  $p$  in the first equation. The corresponding counterfactual SEM is

$$\begin{bmatrix} Y(p) \\ p \\ X \end{bmatrix} := \left( \begin{bmatrix} Y \\ P \\ X \end{bmatrix} : \text{do}(P = p) \right) := \begin{bmatrix} f_Y(p, X, \epsilon_Y) \\ p \\ X \end{bmatrix}.$$



**Figure 8.4:** Causal DAG describing the counterfactual SEM induced by doing  $P = p$ .

The variables  $Y(p)$  and  $X$  are the counterfactuals generated by the intervention  $\text{do}(P = p)$ . Note that the intervention keeps  $X$  and stochastic shocks  $\epsilon_Y$  invariant.

The do-operation is important, but it is just one of many ways to generate counterfactuals.<sup>1</sup>

As an additional general example, we now consider *fix interventions* that induce single-world intervention graphs (SWIGs).<sup>2</sup>

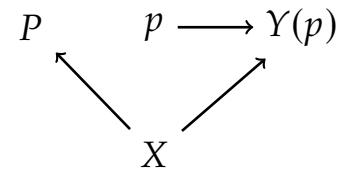
**Fix Interventions and SWIGs.** Instead of removing  $P$  from the graph in Figure 8.2, we can split it into two nodes –  $P$  and a deterministic node  $p$  – where all the outgoing arrows from  $P$  are removed. The fixed node  $p$  then inherits the outgoing arrows from the original  $P$ .

The corresponding counterfactual SEM is

$$\begin{bmatrix} Y(p) \\ P \\ X \end{bmatrix} := \left( \begin{bmatrix} Y \\ P \\ X \end{bmatrix} : \text{fix}_Y(P = p) \right) := \begin{bmatrix} f_Y(p, X, \epsilon_Y) \\ P \\ X \end{bmatrix}.$$

1: The ideas of constructing counterfactuals go back at least since P. Wright's work in 1928 [6] which involved replacing one structural equation by a different equation to define a counterfactual SEM. Specifically, Wright replaced the demand equation by another one reflecting a tax hike.

2: The Fix intervention was introduced in and Heckman and Pinto [3], as a variant of do-operation, and SWIGs were developed by Richardson and Robins [4]



**Figure 8.5:** Causal DAG describing the counterfactual SEM induced by setting  $P = p$  in the  $Y$  equation in (8.2.1) (formally a SWIG).

The fix intervention merely says that we are setting  $P = p$  in the  $Y$  equation. Figuratively speaking, it is a "localized do" operation. The variables  $Y(p)$ ,  $P$ , and  $X$  are the counterfactuals generated by this intervention. The intervention does not affect the  $P$  and  $X$  equations, nor does it affect  $\epsilon_Y$  in the  $Y$  equation.

The SWIG allows us to immediately see that conditional exogeneity (ignorability) holds:

$$Y(p) \perp\!\!\!\perp P \mid X,$$

Therefore we can identify the counterfactual regression  $E[Y(p) | X]$  by the "factual" regression  $E[Y | P = p, X]$ ,

$$E[Y(p) | X] = E[Y(p) | P = p, X] = E[Y | P = p, X],$$

invoking conditional independence and consistency arguments.

The do and fix interventions generate the same counterfactual distribution for  $(Y(p), X)$ , so the average causal effects of simple interventions coincide in the two approaches. However, the fix intervention creates a triple  $(Y(p), X, P)$ , which is useful for answering more complicated counterfactual questions.

For example, the counterfactual prediction  $E[Y(0) | P = 1]$  tells us what trainees ( $P = 1$ ) would have earned on average, had they not gone through the training program ( $p = 0$ ). In treatment effect analysis, this quantity is crucial for defining the average treatment effects for the treated:

$$E[Y(1) | P = 1] - E[Y(0) | P = 1].$$

Thus, the fix intervention allows us to seamlessly talk about conditional on  $P$  counterfactuals:<sup>3</sup>

$$E\left(Y(p) | P = \bar{p}\right) := E\left((Y | P = \bar{p}) : \text{fix}_Y(P = p)\right).$$

### 8.3 General Acyclic SEMs, Causal DAGs and Counterfactuals

#### DAGs and Acyclic SEMs (ASEMs)

We will give a sequence of formal definitions, but we first begin with examples which introduce these definitions informally.

The definitions can be understood by looking at just a single example.

**Example 8.3.1** (Less Simple DAG (LS-DAG)) A directed acyclic graph (DAG) is a collection of nodes and directed edges with no cycles.

Consider the DAG in Figure 8.6: Here we can say that

- $X$  is a parent of its children  $D$  and  $Y$ ;
- $D$  and  $Y$  are descendants of  $Z$ ;
- There is a directed path from  $Z$  to  $Y$ ;

3: The same statement is not true with the do operation in place of the fix operation. Of course, one can also define these conditional counterfactuals by reverting to potential outcomes notation within causal DAGs without using the do notation; see [7].

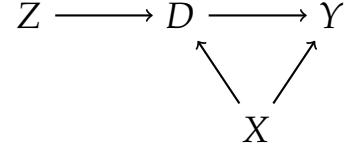


Figure 8.6: LS-DAG Example

- There are two paths from  $Z$  to  $X$ , but no directed path;
- $D$  is a collider of the path  $Z \rightarrow D \leftarrow X$ ;
- $D$  is a noncollider of the path  $Z \rightarrow D \rightarrow Y$ ;
- $Y \leftarrow X \rightarrow D$  is the backdoor path from  $Y$  to  $D$ .
- There are no cycles (there is no directed path that returns to the same node).

**Example 8.3.2** (ASEM Corresponding to the LS-DAG) A system of triangular structural equations corresponding to Example 8.3.1 is

$$\begin{aligned} Y &:= f_Y(D, X, \epsilon_Y), \\ D &:= f_D(Z, X, \epsilon_D), \\ X &:= \epsilon_X, \\ Z &:= \epsilon_Z, \end{aligned}$$

where  $\epsilon_Y, \epsilon_X, \epsilon_D$ , and  $\epsilon_Z$  are mutually independent.

Factual distributions in DAG models have a beautiful Markov factorization structure, which allows for a simple representation of the joint distribution of all variables.

**Example 8.3.3** (Factual Law in LS-DAG) Noting the dependencies of each variable in the LS-DAG, we can write the joint distribution (density)  $p$  of  $Y, D, X, Z$  as

$$p(y, d, x, z) = p(y|d, x) p(d|x, z) p(x) p(z).$$

Indeed,

$$p(y, d, x, z) = p(y|d, x, z) p(d, x, z),$$

by Bayes' law. Then  $p(y|d, x, z) = p(y|d, x)$  as the distribution of  $Y$  is independent of  $Z$ , given its parents  $D$  and  $X$ . Further,  $p(d, x, z) = p(d|x, z) p(x, z)$ , by Bayes' law, and  $p(x, z) = p(z) p(x)$  by independence.

## General definitions

The purpose of the rest of this section is to give concise general definitions.

A graph  $G$  is an ordered pair  $(V, E)$ , where  $V = \{1, \dots, J\}$  is a collection of vertices/nodes and  $E$  is a matrix of edges  $e_{ij} \in \{0, 1\}$  – that is,  $E = \{e_{ij} : (i, j) \in V^2\}$ .

Given a collection of random variables  $X = (X_j)_{j \in V}$ , we identify each  $j$  with the name " $X_j$ " whenever convenient. If the edge  $(i, j)$  is present, namely  $e_{ij} = 1$ , we read it as

" $X_i \rightarrow X_j$ " or " $X_i$  is an immediate cause of  $X_j$ ".

Consider a strict partial order  $<$  on  $V$  induced by  $E$ , where  $X_j < X_k$  (we read this as " $X_j$  is determined before  $X_k$ ") means that either  $X_j \rightarrow X_k$  or  $X_j \rightarrow X_{v_1} \rightarrow \dots \rightarrow X_{v_m} \rightarrow X_k$  is true for some  $v_\ell$ 's in  $V$ . A partial ordering of  $V$  exists if for each  $j$  the statement  $X_j < X_j$  is not true.<sup>4</sup> By the identification above, we can replace names  $X_\ell$ 's with their indices  $\ell$ 's.

4: The latter statement means that there are no cycles.

**Definition 8.3.1 (DAG)** *The graph  $G = (V, E)$  is a DAG if the graph has no cycles, that is, if  $V$  is partially ordered by the edge structure  $E$ .*

**Example 8.3.4 (LS-DAG continued)** In our example (Example 8.3.1), we had vertices  $V = \{1, 2, 3, 4\}$  identified with  $Y, D, X, Z$ , and the edge set

$$E = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

The partial ordering is  $X < D, X < Y, Z < D, D < Y$ .

**Definition 8.3.2 (Parents, Ancestors, Descendants on a DAG)** *The parents of  $X_j$  are the set  $Pa_j := \{X_k : X_k \rightarrow X_j\}$ . The children of  $X_j$  are the set  $Ch_j := \{X_k : X_j \rightarrow X_k\}$ . The ancestors of  $X_j$  are the set  $An_j := \{X_k : X_k < X_j\}$ . The descendants of  $X_j$  are the set  $Ds_j := \{X_k : X_k > X_j\}$ .*

**Definition 8.3.3 (Paths and Backdoor Paths on DAGs)** *A directed path is a sequence  $X_{v_1} \rightarrow X_{v_2} \rightarrow \dots \rightarrow X_{v_m}$ ; a non-directed path is a path, where some arrows (but not all) are replaced by  $\leftarrow$ . A collider node is a node  $X_j$  such that  $\rightarrow X_j \leftarrow$ . A backdoor path from  $X_l$  to  $X_k$  is an undirected path that starts at  $X_l$  and ends with an incoming arrow  $\rightarrow X_k$ .*

**Definition 8.3.4 (ASEM)** *The ASEM corresponding to the DAG*

$\mathbf{G} = (V, E)$  is the collection of random variables  $\{X_j\}_{j \in V}$  such that

$$X_j = f_j(Pa_j, \epsilon_j), \quad j \in V,$$

where the disturbances  $(\epsilon_j)_{j \in V}$  are jointly independent.

**Definition 8.3.5** (Linear ASEM) *The linear ASEM is an ASEM where the equations are linear:*

$$f_j(Pa_j, \epsilon_j) = f'_j Pa_j + \epsilon_j;$$

here we identify functions  $\{f_j\}$  with coefficient vectors  $\{f'_j\}$ .

In linear ASEMs we may replace the requirement of independent errors by the weaker requirement of uncorrelated errors.

**Definition 8.3.6** (Structural/Potential Response Processes)

*The structural potential response processes for the ASEM corresponding to the DAG  $\mathbf{G} = (V, E)$  are given by the structure:*

$$X_j(pa_j) = f_j(pa_j, \epsilon_j), \quad j \in V,$$

*viewed as stochastic processes indexed by the potential parental values  $pa_j$ .*

**Definition 8.3.7** (Consistency) *The observable variables are generated by drawing  $\{\epsilon_j\}_{j \in V}$  and then solving the system of equations for  $\{X_j\}_{j \in V}$ .*

The stochastic shocks  $\{\epsilon_j\}_{j \in V}$  are called exogenous variables, and the variables  $\{X_j\}_{j \in V}$  are called endogenous variables. Endogenous variables are determined by the model equations, while exogenous variables are not.

The joint distribution of variables in ASEMs is generally characterized as follows.

**Theorem 8.3.1** (Factual Law via Markovian Factorization) *The general ASEM model, given by  $(X_j)_{j \in V}$  with an associated DAG  $\mathbf{G}(V, E)$ , obeys the following equivalent properties:*

- **Factorization:** The law admits factorization:

$$\mathsf{p}(\{x_\ell\}_{\ell \in V}) = \prod_{\ell \in V} \mathsf{p}(x_\ell | pa_\ell).$$

- **Local Markov Property:** All variables are independent of their non-descendants given their parents.

## Testable Restrictions by D-Separation

The DAG structure implies another key property: a global Markov condition. In order to define this property, we need a few more definitions.

**Definition 8.3.8** (Blocked or D-Separated Path) A path  $\pi$  is said to be blocked by a subset of nodes  $S$  if and only if

- $\pi$  contains a chain  $i \rightarrow m \rightarrow j$  or a fork  $i \leftarrow m \rightarrow j$  such that  $m$  is in  $S$ ;
- Or,  $\pi$  contains a collider  $i \rightarrow m \leftarrow j$ , where neither  $m$  nor any descendant of  $m$  is in  $S$ .

A path that is not blocked is called open.

The definition allows empty sets as conditioning sets.

In Figure 8.7 the (backdoor) path  $Y \leftarrow X \rightarrow D$  is blocked by  $S = X$ .

**Definition 8.3.9** (Opening a Path by Conditioning) A path containing a collider is opened by conditioning on it or its descendant.

In Figure 8.8 the path  $Y \rightarrow C \leftarrow D$  is blocked, but becomes open by conditioning on the collider  $S = C$ .

The following is a deep result concerning the conditional independence relations encoded in the graphs.

**Definition 8.3.10** (D-Separation) Given a DAG  $G$ , a set of nodes  $S$  d-separates nodes  $X$  and  $Y$  if nodes in  $S$  block all paths between  $X$  and  $Y$ . D-separation is denoted as

$$(Y \perp\!\!\!\perp_d X | S)_G.$$

**Theorem 8.3.2** (Verma and Pearl [8]; Conditional Independence from D-Separation) d-Separation implies conditional independence:

- **Global Markov:**  $(Y \perp\!\!\!\perp_d X | S)_G \implies Y \perp\!\!\!\perp X | S$ .

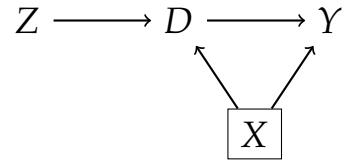


Figure 8.7: The path  $Y \leftarrow X \rightarrow D$  is blocked by conditioning on  $X$ .

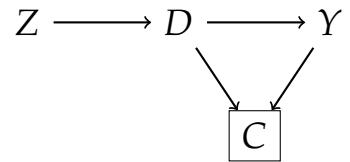


Figure 8.8: The path  $Y \rightarrow C \leftarrow D$  is blocked, but becomes open by conditioning on  $C$ .

Figuratively speaking, conditioning on  $S$  breaks the information flow between  $Y$  to  $X$ , meaning that  $Y$  can't be predicted by  $X$ , conditional on  $S$ , and vice versa.

This fundamental result is very intuitive and can be verified directly in simple examples. However, the formal proof is hard (trying to find a simple proof is a good exercise). The reverse implication is not true in general, but is argued to hold "generically" as we discuss in Section 8.6.

**Example 8.3.5** We show a couple of examples illustrating that  $d$ -separation implies conditional independence:

1. In Figure 8.9, the variables  $X$  and  $Y$  are  $d$ -separated by  $S = (Z, U)$ , because  $S$  blocks all paths between  $X$  and  $Y$ . We also have  $Y$  is independent of  $X$  conditional on  $S$ : By the Markov factorization property,  $p(y, x | u, z) = p(y | x, z, u) p(x | z, u) = p(y | u, z) p(x | z, u)$ . This equality provides a testable restriction.
2. In Figure 8.10, the variables  $X$  and  $Y$  are  $d$ -separated by  $S = Z$ , because  $S$  blocks all paths between  $X$  and  $Y$ . We also have  $Y$  is independent of  $X$  conditional on  $S$ : By the Markov factorization property,  $p(y, x | z) = p(y | z) p(x | z)$ . This equality provides a testable restriction.

These testable restrictions are called exclusion restrictions in econometrics because

$$Y \perp\!\!\!\perp X | Z \text{ is equivalent to } p(y | x, z) = p(y | z), \quad (8.3.1)$$

where the equivalence follows from Bayes' law. In particular,

$$E[g(Y) | X, Z] = E[g(Y) | Z], \quad (8.3.2)$$

for any bounded function  $g$  of  $Y$ . (8.3.2) means that  $X$  is excluded from the best predictor of  $g(Y)$  using  $X$  and  $Z$ . There are many tests of such restrictions available in the literature.<sup>5</sup>

One of the notebooks below provides an example of using Verma-Pearl tests [8] for linear ASEMs.

**Implementation of Tests in Linear ASEMs.** Consider the hypothesis that  $Y$  is independent of  $X$ , given  $Z$ . In linear ASEMs, we can test this hypothesis by testing whether the

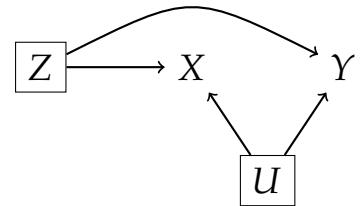


Figure 8.9: Example of  $d$ -separation.

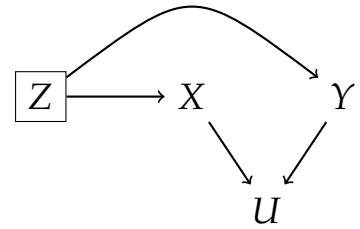


Figure 8.10: Example of  $d$ -separation.

5: E.g., the reader can search Google Scholar for conditional independence tests, exclusion restrictions tests, or conditional moment tests.

coefficient  $\alpha = 0$  in the projection equation

$$Y = \alpha' X + \beta' Z + \epsilon, \epsilon \perp Z.$$

We can perform this test easily with the tools we've developed so far; see the Dagitty Notebook for an example.

Such tests are available much more generally than in the linear model.

**Remark 8.3.1** (Equivalence of Local and Global Markov Properties) The local Markov property, the Markov factorization, and the global Markov property are equivalent (Pearl [7]). Therefore one can use any of these properties to set up tests of the validity of the Markov structure.

## 8.4 Counterfactuals Induced by Interventions

We next discuss counterfactuals generated by interventions. We first consider counterfactuals in the Less Simple DAG example (Example 8.3.1).

**Example 8.4.1** (CF-ASEM Induced by Do for LS-DAG Example) Consider the ASEM from Example 8.3.1. A counterfactual system induced by  $do(D = d)$  is

$$\begin{aligned} Y(d) &:= f_Y(X, d, \epsilon_Y), \\ d, \\ Z &= \epsilon_Z, \\ X &= \epsilon_X, \end{aligned}$$

where  $\epsilon_X, \epsilon_D, \epsilon_Y$  are mutually independent. The corresponding graph, provided in Figure 8.11, is denoted by  $G(d)$ .

**Example 8.4.2** (CF-ASEM Induced by Fix for LSDAG Example) Consider the ASEM from Example 8.3.1. A counterfactual SEM induced by  $fix(D = d)$  is assumed to take the following

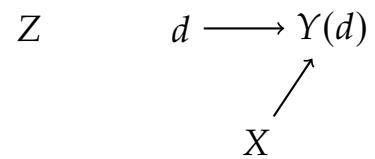
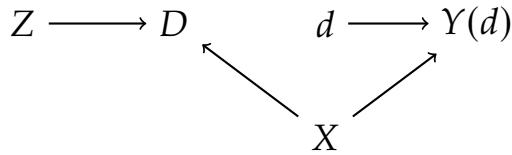


Figure 8.11: CF LS-DAG induced by  $do(D = d)$  intervention.



**Figure 8.12:** CF LS-DAG (SWIG) induced by the  $\text{fix}_Y(D = d)$  intervention.

form:

$$\begin{aligned} Y(d) &:= f_Y(X, d, \epsilon_Y), \\ d, \\ D &:= f_D(X, Z, \epsilon_D), \\ Z &:= \epsilon_Z, \\ X &:= \epsilon_X, \end{aligned}$$

where  $\epsilon_X, \epsilon_Z, \epsilon_D, \epsilon_Y$  are mutually independent. The corresponding graph, provided in Figure 8.12, is denoted by  $\tilde{G}(d)$ .

We now give a more general definition.

**Definition 8.4.1** (Counterfactual ASEM induced by Do Intervention) *The intervention  $\text{do}(X_j = x_j)$  on an ASEM is said to create the CF-ASEM defined by the modified graph*

$$G(x_j) = (V, E^*)$$

and collection of counterfactual variables

$$(X_k^*)_{k \in V}$$

where

- the edges incoming to the node  $j$  are set to zero, namely  $e_{ij}^* = 0$  for all  $i$ ,
- the remaining edges are preserved, namely  $e_{ik}^* = e_{ik}$ , for all  $i$  and  $k \neq j$ , and
- the counterfactual random variables are defined as

$$\begin{aligned} X_k^* &:= f_k(Pa_k^*, \epsilon_k), \text{ for } k \neq j, \\ X_j^* &:= x_j \end{aligned}$$

where  $Pa_k^*$  are parents of  $X_k^*$  ( $k \neq j$ ) under  $E^*$ .

The do intervention modifies the graph  $G$  to  $G(x_j)$  by removing edges. Pearl [7] has described this process as "surgery".<sup>6</sup> We next define the *do* notation to mean

$$\left( (X_\ell)_{\ell \in V} : \text{do}(x_j) \right) := (X_\ell^*)_{\ell \in V}.$$

6: This sounds a bit painful.

**Definition 8.4.2** (Counterfactual ASEM induced by Fix Intervention) *The intervention  $\text{fix}(X_j = x_j)$  on an ASEM is said to create the CF-ASEM defined by the modified SWIG*

$$\tilde{\mathbf{G}}(x_j) := (\tilde{V}, \tilde{E}),$$

and collection of counterfactual variables

$$(X_k^*)_{k \in V} \cup (X_a^*)$$

where we split the node  $X_j$  into  $X_j^* := X_j$  and the new deterministic node  $a$

$$X_a^* := x_j,$$

where

- ▶ the node  $X_a$  inherits only outgoing edges from  $X_j$  and no incoming edges; namely  $\tilde{e}_{ai} = e_{ji}$  for all  $i$  and  $\tilde{e}_{ia} = 0$  for all  $i$ ;
- ▶ the node  $X_j^*$  inherits only incoming edges from  $X_j$  and no outgoing edges, namely  $\tilde{e}_{ij} = e_{ij}$  for all  $i$  and  $\tilde{e}_{ji} = 0$  for all  $i$ ;
- ▶ all the remaining edges are preserved, namely  $\tilde{e}_{ik} = e_{ik}$ , for all  $i$  and  $k \neq j$  and  $k \neq a$ ; and
- ▶ the counterfactual random variables are assigned according to

$$X_k^* := f_k(Pa_k^*, \epsilon_k), \text{ for } k \neq a,$$

where  $Pa_k^*$  are parents of  $X_k^*$  ( $k \neq j$ ) under  $\tilde{E}$ .

Intervention induces new counterfactual distributions for the endogenous variables; see the Appendix for details.

## 8.5 Identification by Conditioning

### Main Idea

An adjustment set  $S$  is said to be valid for identification of the causal effect of  $D$  on  $Y$  if the conditional exogeneity/ignorability condition holds

$$Y(d) \perp\!\!\!\perp D \mid S.$$

In what follows, we present an exhaustive (complete) approach for finding valid adjustment sets by using SWIGs.

We write down the counterfactual SWIG induced by the

$$\text{fix}(D = d)$$

intervention, which operates on all structural equations defining the descendants of  $D$ , by setting  $D = d$  in these equations.

Then, if we have that the potential outcome  $Y(d)$  is  $d$ -separated from the (policy) variable  $D$  by a set of variables  $S$ , conditional exogeneity/ignorability holds:

$$Y(d) \perp\!\!\!\perp D \mid S.$$

As before, given that conditional exogeneity/ignorability holds, we can identify counterfactual predictions

$$E[Y|S = s : \text{do}(d)] := E[Y(d)|S = s]$$

from actual predictions:

$$E[Y|S = s, D = d],$$

provided that the positivity condition  $p(s, d) > 0$  holds. The agreement between counterfactual and conditional predictions follows because

$$E[Y(d)|S = s] = E[Y(d)|D = d, S = s]$$

by exogeneity and

$$E[Y(d)|D = d, S = s] = E[Y|D = d, S = s]$$

by consistency.

We can recover unconditional counterfactual means by integration:

$$E[Y : \text{do}(d)] := E[Y(d)] = E[E[Y|S, D = d]],$$

provided that the positivity condition  $p(s, d) > 0$  for each  $s$  in the support of  $S \mid D = d$  holds.

**Example 8.5.1** (Identification in LS-DAG.) In the SWIG graph in Figure 8.13, we see that either  $S = X$  or  $S = (X, Z)$  d-

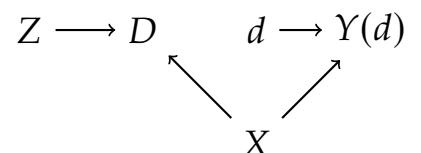


Figure 8.13: CF LS-DAG induced by  $\text{fix}(D = d)$  intervention.

separates  $Y(d)$  from  $D$ . Therefore either choice of  $S$  provides a valid adjustment set for identifying counterfactual predictions. Here conditioning on  $Z$  is not necessary, though we maintain robustness with respect to the presence of a directed edge from  $Z$  to  $Y$  in including  $Z$  in the conditioning set.

We can identify the entire conditional distribution

$$P(Y(d) \leq t \mid S = s)$$

from the conditional distribution

$$P(Y \leq t \mid D = d, S = s).$$

We achieve identification of the distribution by replacing  $Y$  with  $1(Y < t)$  in all previous statements and applying the same arguments for each  $t \in \mathbb{R}$ . The unconditional distribution of potential outcomes is retrieved by integrating out  $S$ :

$$P(Y(d) \leq t) := E[P(Y(d) \leq t \mid S)].$$

The following theorem, essentially due to [4], records the discussion formally.

**Theorem 8.5.1** (A Complete Criterion for Identification by Conditioning) Consider any ASEM with DAG  $\mathbf{G}$ . Let us re-label a policy node  $X_j$  as  $D$ , and let  $Y$ , an outcome of interest, be any other descendant of  $D$ .

Consider a Swig DAG  $\tilde{\mathbf{G}}(d)$  which is induced by the  $\text{fix}(D = d)$  intervention. Consider any other subset of nodes  $S$  that appears in both  $\mathbf{G}$  and  $\tilde{\mathbf{G}}(d)$ , such that

$Y(d)$  is  $d$ -separated from  $D$  by  $S$  in  $\tilde{\mathbf{G}}(d)$ .

► Then the following conditional exogeneity/ignorability holds:

$$Y(d) \perp\!\!\!\perp D \mid S.$$

► Then

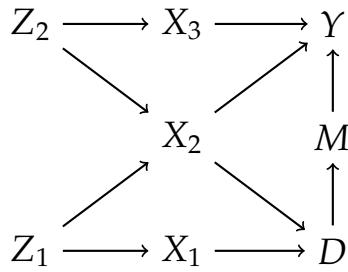
$$E[Y(d)|S = s] = E[Y \mid D = d, S = s]$$

holds for all  $s$  such that  $p(d, s) > 0$ .

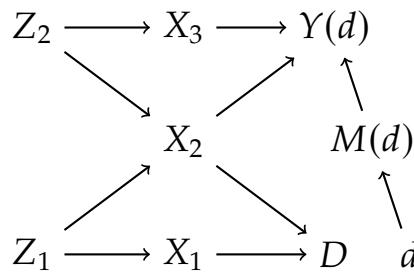
**Example 8.5.2** (Pearl's Example) Consider the DAG in Figure 8.14 and the corresponding ASEM, which we don't write out. Here we are interested in the causal effect  $D \rightarrow Y$ , that is, the effect  $d \mapsto Y(d)$ . The corresponding SWIG-intervention DAG is shown in Figure 8.15. In this DAG, valid adjustment sets  $S$  include

$$\{X_1, X_2\}, \{X_2, X_3\}, \{X_2, Z_2\}, \{X_2, Z_1\},$$

because each d-separates  $Y(d)$  and  $D$  by blocking all open paths. Conditioning on just  $X_2$  won't work, because it blocks the inner backdoor paths from  $Y(d)$  to  $D$ , but opens the outer path on which  $X_2$  is a collider. To close this opened path it suffices to also condition on  $X_1$  or  $X_3$  or  $Z_1$  or  $Z_2$ .



**Figure 8.14:** A DAG in Pearl's Example



**Figure 8.15:** The DAG induced by the Fix/Swig intervention  $\text{fix}(D = d)$  in Pearl's Example.

## Useful Adjustment Strategies as Corollaries

The strategy given above provides a complete (exhaustive) criterion for finding valid adjustment sets. We now discuss other frequently used strategies for obtaining valid adjustment sets which are strictly less general. Some of these strategies are quite helpful because they are either very simple to apply or can also be used under partial knowledge of the DAG.<sup>7</sup>

We consider three approaches that allow us to identify the causal effect of  $D$  on  $Y$ :

7: See [9] for a more detailed discussion of identification by conditioning under limited knowledge of DAGs.

- ▶ **Conditioning on all parents** of  $Y$  (that are not descendants of  $D$ ) or of  $D$ , or both, is sufficient. This approach is valid, irrespective of the remaining structure of the problem.
- ▶ Conditioning using the **backdoor criterion** enables us to find all minimal adjustment sets.
- ▶ **Conditioning on all common causes** of  $D$  and  $Y$  is also sufficient.

### Conditioning on Parents

A very simple strategy is conditioning on the parents of  $D$  or  $Y$  or both, which we have already seen in the introduction.

**Example 8.5.3** (Pearl's Example Continued) One simple principle is that conditioning on parents of  $D$ , namely  $X_1$  and  $X_2$ , is sufficient. Alternatively, conditioning on all parents of  $Y$  that are non-descendants of  $D$ , namely  $X_2$  and  $X_3$ , is also sufficient. Here we should not condition on  $M$ , because it is a descendant of  $D$ .

**Corollary 8.5.2** (Adjustment for Parents) Consider any ASEM. Re-label a policy node  $X_j$  as  $D$ , and let  $Y$ , an outcome of interest, be any other descendant of  $D$ .

- ▶ Let  $Z$  be all parents of  $D$ , and let  $A$  be any other set of nodes that are not descendants of  $D$ . Then  $S = (A, Z)$  is a valid adjustment set.
- ▶ Let  $Z$  be the set of all parents of  $Y$  that are non-descendants of  $D$  and let  $A$  be any other set that are not descendants of  $D$ . Then  $S = (A, Z)$  is a valid adjustment set.

Note that  $A$  is allowed to be an empty set. Also note that, in the second case, the additional adjustment set  $A$  is redundant, since  $p(y | a, z, d) = p(y | z, d)$  in this case.

Adjusting for parents is a very useful strategy, because it only requires knowledge of parents in a DAG without precise knowledge of the remaining graph structure. Conditioning on parents is also behind the propensity score strategies used in many experimental or quasi-experimental empirical analyses. If the propensity score is known, it can be used as a parent of  $D$  itself. Finally, conditioning on parents of  $Y$  is most useful for attaining maximal statistical efficiency, but may be less robust than conditioning on *both* sets of parents under unforeseen

deviations from the given graph structure. See [9] for further detailed discussion of robustness of adjusting for both sets of parents.

### Conditioning by Backdoor Blocking

Pearl [7] developed the following powerful criterion.

**Corollary 8.5.3** (Backdoor Criterion) Consider any ASEM. Relabel a policy node  $X_j$  as  $D$ , and let  $Y$ , an outcome of interest, be any other descendant of  $D$ . The adjustment set  $S$  is valid if the backdoor criterion is satisfied: No element of  $S$  is a descendant of  $D$ , and all backdoor paths from  $Y$  to  $D$  are blocked by  $S$ .

In other words, if a collection of random variables  $S$  satisfies the backdoor criterion with respect to  $(D, Y)$ , then conditioning on  $S$  identifies the causal effect of  $D$  on  $Y$ . The basic idea is that if we block the backdoor path, we remove all channels of non-causal association between  $D$  and  $Y$ .

**Example 8.5.4** (Pearl's Example Again, using Backdoor) The unintervened graph in Figure 8.14 has two backdoor paths from  $D$  to  $Y$ : the inner path  $D \leftarrow X_2 \rightarrow Y$  and the outer path  $D \leftarrow X_1 \leftarrow Z_1 \rightarrow X_2 \leftarrow Z_2 \rightarrow X_3 \rightarrow Y$ . Conditioning on just  $X_2$  won't work, because it blocks the inner backdoor paths from  $Y$  to  $D$ , but opens the outer path on which  $X_2$  is a collider. To close this opened path it suffices to condition on  $X_1, X_3, Z_1$ , or  $Z_2$ . For example, conditioning sets  $S_1 = \{X_1, X_2\}$  or  $S_2 = \{X_2, X_3\}$  are valid. Figuring out other valid conditioning sets is left as an exercise (one of the notebooks provides the answers). Conditioning on  $M$  is obviously not valid – it is a descendant of  $D$ , an intermediate outcome.

Application of the backdoor criterion can produce all minimal adjustment sets. Relative to the complete strategy formalized in Theorem 8.5.1, we exclude the descendants of  $D$  from valid adjustment sets when we focus on backdoor paths. A simple example of a graph where the backdoor criterion does not find all valid adjustment sets is

$$Z \leftarrow D \rightarrow Y.$$

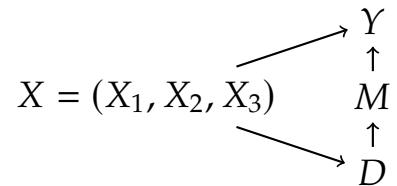
Here conditioning on  $Z$  is valid but unnecessary. Conditioning on  $Z$  may thus decrease statistical efficiency.<sup>8</sup>

<sup>8</sup>: We may think that conditioning on  $Z$  here could be useful to uncover heterogeneity, but  $Y(d)$  does not depend on  $Z$ , so here conditioning on  $Z$  is not useful (for describing heterogeneity) and can decrease the efficiency of the estimator.

### Conditioning on All Common Causes of $D$ and $Y$

Another simple and widely used adjustment strategy is conditioning on all common causes of the outcome variable of interest and the treatment variable.

**Example 8.5.5** (Pearl's Example Again, using the All Common Causes Criterion) The set of common causes of  $D$  and  $Y$  is  $\{Z_1, Z_2, X_2\}$ . This set is a valid adjustment set that differs from the sets ones found using the parental strategy. We can push the All Common Causes criterion further. For example, we can omit  $Z_1$  and  $Z_2$  from the DAG, and we can create a new node  $X = (X_1, X_2, X_3)$  producing the DAG shown in Figure 8.16. This DAG corresponds to a valid ASEM model where  $X$  now represents all common causes of  $D$  and  $Y$ , making it a sufficient adjustment set. This set is bigger than some of the sets found by the previous criteria. It is also tempting to see if the "root common" causes  $Z_1$  and  $Z_2$  in the original DAG, Figure 8.14, form a valid adjustment set – and they actually do not (why?).



**Figure 8.16:** Reduced DAG for Pearl's Example

**Corollary 8.5.4** (Adjustment for All Common Causes) Consider any ASEM. Re-label a policy node  $X_j$  as  $D$ , and let  $Y$ , an outcome of interest, be any other descendant of  $D$ . Let  $S$  be the intersection of the ancestors of  $D$  and  $Y$ , called the common causes:

$$S = An_D \cap An_Y.$$

Then  $S$  is a valid adjustment set. Furthermore, the set of variables  $S'$  that completely mediates the effects of  $S$  on  $Y$  and  $D$  also constitutes a valid adjustment set.

The strategy above is commonly used in empirical work. However, [9] recommend adjusting for the union  $S$  of causes of  $Y$  or  $D$  (excluding descendants of  $D$ ) in practice, formally quantifying this strategy as the maximally robust strategy under perturbations of a specified DAG structure that preserves  $S$ . This strategy is useful when we don't know the parents of  $Y$  or  $D$ , but only know that  $S$  are their ancestors.

**Corollary 8.5.5** (Adjustment for the Union of Causes) Consider any ASEM. Re-label a policy node  $X_j$  as  $D$ , and let  $Y$ , an outcome of interest, be any other descendant of  $D$ . Let  $S$  be the union of the

ancestors of  $D$  and  $Y$  that excludes descendants of  $D$ :

$$S = An_D \cup An_Y \setminus Ds_D.$$

Then  $S$  is a valid adjustment set.

**Example 8.5.6** (Pearl's Example Continued) Application of the Union of Causes criterion gives  $\{Z_1, Z_2, X_1, X_2, X_3\}$  as a valid adjustment set. If  $Z_1$  and  $Z_2$  are not observed, then adjusting for  $\{X_1, X_2, X_3\}$  suffices.

## 8.6 Falsifiability and Causal Discovery\*

### Equivalence Classes and Falsifiability

**Definition 8.6.1** (Equivalence Classes) *The class of DAGs that induce the same joint distribution of variables is called an equivalence class, and members of an equivalence class may be described as Markov equivalent.*

Pearl [7] shows that the equivalence class of a DAG is given by reversing any edges such that any such reversal does not destroy existing or create new  $v$ -structures: converging arrows whose tails are not connected by an edge.

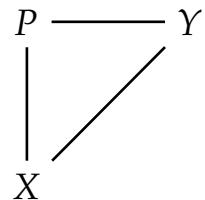
The equivalence classes of a DAG are called PDAGs (partially directed acyclic graphs). We plot them by erasing arrowheads that can be oriented in the opposite direction. See, for example, Figures 8.17 and 8.18.

The edge matrix  $E$  is *triangular* if rows of  $E$  can be rearranged to have 1's below the diagonal, like in the TSEM example, Example 8.2.1.

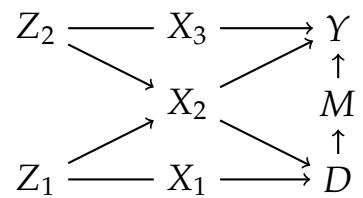
**Remark 8.6.1** (Falsifiability) In the absence of any further restrictions, an ASEM with graph  $G = (V, E)$  has testable implications if  $E$  is not triangular. If  $E$  is triangular, then any law  $p$  of any arbitrary collection of random variables  $(X_j)_{j \in V}$  indexed by  $V$  can be factorized as

$$p(\{x\}_{j \in V}) = \prod_{j \in V} p(x_j | pa_j).$$

With population data we have  $p$  and can check if it factorizes according to  $V$ . If matrix  $E$  is triangular,  $p$  always obeys the



**Figure 8.17:** The equivalence class for DAGs in the TSEM (Example 8.2.1). The undirected edges mean that they can be directed in any direction as long as this does not create a cycle. In empirical analysis directionality must therefore be deduced and assumed from the context.



**Figure 8.18:** The Equivalence Class for the DAG in Pearl's Example (Example 8.5.2). Only two edges can be reoriented here.

factorization property. This is to say that there are no exclusion restrictions in the model.

**Example 8.6.1** (TSEM continued) In the TSEM example (Example 8.2.1, we have vertices  $V = \{1, 2, 3\}$  identified with  $Y, P, X$  and the "triangular" edge set

$$E = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}.$$

In the absence of other assumptions, the corresponding TSEM implies no falsifiable restrictions. The equivalence class of the DAG model for this case is generated by rearranging the rows of  $E$  in  $3!$  ways, which is equivalent to rearranging the names  $(Y, P, X)$  for the nodes.

## Faithfulness and Causal Discovery

d-Separation implies conditional independence, but the reverse implication

$$Y \perp\!\!\!\perp X|S \implies (Y \perp\!\!\!\perp_d X|S)_G \quad (8.6.1)$$

is not true in general. If we restrict attention to the set of distributions  $p$  of random variables associated with graph  $G$  such that implication (8.6.1) holds, we are said to impose the *faithfulness* assumption on  $p$ .

**Example 8.6.2** (Unfaithfulness) A trivial example is the DAG

$$X \rightarrow Y$$

where

$$Y := \alpha X + \epsilon_Y; \quad X := \epsilon_X;$$

with  $\epsilon_X$  and  $\epsilon_Y$  independent standard normal variables. Consider  $S$  to be the empty set. In this model we have that  $Y \perp\!\!\!\perp X$  when  $\alpha = 0$ , but  $Y$  and  $X$  are not d-separated in the DAG  $X \rightarrow Y$ . The distribution  $p$  of  $(Y, X)$  corresponding to  $\alpha = 0$  is said to be unfaithful. However, the exceptional point  $\alpha = 0$  has a measure 0 on the real line, so this exception is said to be non-generic.

The observation about the simple example above generalizes: If probabilities  $p$  themselves are viewed as generated by Nature as a draw from a continuum  $P$ , where each  $p \in P$  factorizes according to  $G$ , then the set of models where this reverse

implication does not hold has measure zero. This observation motivates the faithfulness assumption as a weak requirement; that is, a given  $p$  is "very unlikely" to be unfaithful.

**Remark 8.6.2** (Causal Discovery) The use of the faithfulness assumption should allow us to discover the equivalence class of the true DAG from the population distribution  $p$ : We can compute all valid conditional independence relations and then discover the equivalence class of DAGs. See, for example, the PC algorithm [10] for an explicit causal discovery algorithm. We can then apply contextual knowledge to further orient the edges of the graph.

Even though the set of unfaithful distributions has measure zero, the neighborhood of this set may not be small in high-dimensional graphs, which creates difficulty in inferring the DAG structure from an estimated version  $\hat{p}$ .

See Uhler et al's [11] figure.

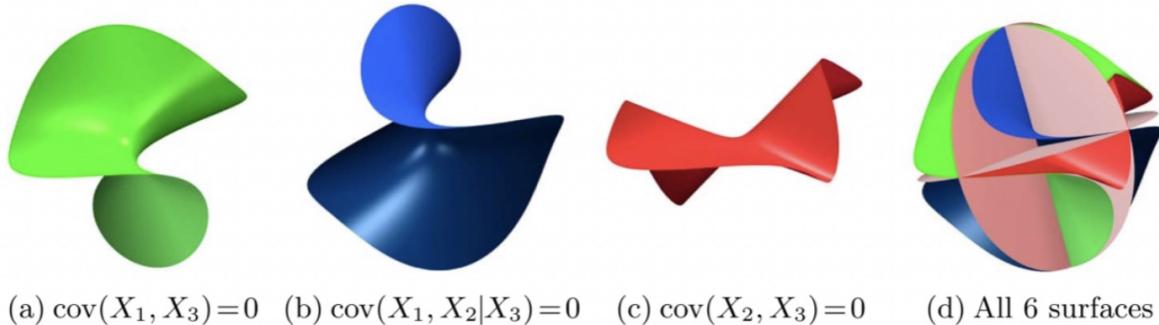
**Example 8.6.3** (Unfaithfulness Continued) In the trivial example above, suppose that we have that  $\hat{\alpha} = .1$  and  $\hat{\alpha} \sim N(\alpha, \sigma^2)$  where  $\sigma = .1$ . Then we can't be sure whether  $\alpha = 0$ ,  $\alpha = .1$ , or  $\alpha$  equals any other number, though say a 95% confidence interval would have  $\alpha$  between  $-.1$  and  $.3$ . Therefore, we can't be sure whether the true model is

$$X \rightarrow Y \text{ or } X \leftarrow Y.$$

Informally speaking, it is impossible to discover the true graph structure in this example when  $\alpha \approx 0$ . In econometrics jargon, this statement amounts to saying that we can't distinguish exact exclusion restrictions from "approximate" exclusion restrictions.

Thus, it is hard to distinguish exact independence from approximate independence with finite data. In high-dimensional graphs, the possibility that  $\hat{p}$  lands in the "near-unfaithful" regions can be substantial, as Uhler et. al.[11]'s analysis shows.

The observations above motivate a form of sensitivity analysis – e.g., Conley et al [12] – where one replaces exact exclusion restrictions by approximate exclusion restrictions that can't be distinguished from exact exclusion restrictions and examines the sensitivity of the causal effect estimates.



**Figure 8.19:** Uhler et. al [11]: A set of "unfaithful" distributions  $p$  in the simple triangular Gaussian SEM/DAG:  $X_1 \rightarrow X_2, (X_1, X_2) \rightarrow X_3$ . The set is parameterized in terms of the covariance of  $(X_1, X_2, X_3)$ . The right panel shows the set, and the three panels show 3 of 6 components of the set. Each of the cases corresponds to the non-generic case which would make faithfulness fail, leading to discovery of the wrong DAG structure. In finite samples, we are not able to distinguish models that are close to the set of unfaithful distributions from unfaithful distributions and may also discover the wrong DAG structure.

## Notes

Any econometric study that relies on conditioning to learn the causal effect must have a thought process that justifies this approach. The DAG/ASEM framework is a rigorous form of this process, which enables explicit incorporation of domain knowledge, automatic checking of identifiability, and automatic deduction of testable restrictions. Graphs also offer an effective way of communicating models.

## Notebooks

- ▶ **R: Dagitty Notebook** employs the R package "dagitty" to analyze Pearl's example as well as simpler ones. This package automatically finds adjustment sets and also lists testable restrictions in a DAG. We then go ahead and test those restrictions assuming a linear ASEM structure.
- ▶ **R: Dosearch Notebook** employs the R-package "dosearch" to analyze Pearl's example. This package automatically finds identification answers to causal queries, allowing us to also answer these types of queries under different data sources, sample selection, and other deviations from the standard framework.

## Additional resources

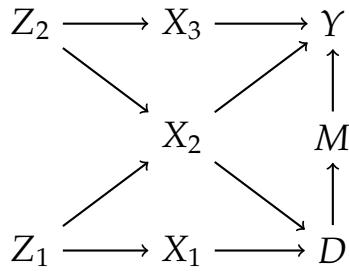
- ▶ **Dagitty.Net** is an excellent online resource where you can

plot and analyze causal DAG models online. It contains many interesting examples of DAGs used in empirical analysis in various fields.

- [Causalfusion.Net](#) is another excellent online resource where you can plot and analyze causal DAG models. This resource covers many different deviations from the standard framework.

## Study Problems

The study problems ask learners to re-analyze Pearl's example DAG. The provided notebooks are a useful starting point. Recall that Pearl's example is structured as follows:



**Figure 8.20:** Pearl's Example

1. For Pearl's example, write out the parents, non-parents, descendants, and non-descendants of nodes  $X_2$  and  $M$ . List all the backdoor paths between  $Y$  and  $X_2$ . Can you identify the effect of  $X_2$  on  $Y$  by conditioning?
2. Consider Pearl's example and answer the following questions. The best way to answer this question is to use computational packages (but please explain the principles the package is using).
  - a) What are the testable implications of the assumptions embedded in the model? Hint: The testable implications are derived from the d-separation criterion.
  - b) Assume that only variables  $D$ ,  $Y$ ,  $X_2$  and  $X_3$  are measured, are there any testable implications?
  - c) Now assume only  $D$ ,  $Y$ , and  $X_2$  are measured. Are there any testable implications?
  - d) Now assume that all of the variables but  $X_3$  (8 in total) are measured. Are there any testable restrictions?

- e) Assume that an alternative model, competing with Model 1, has the same structure, but with the  $X_2 \rightarrow D$  arrow reversed. What statistical test would distinguish between the two models?
- 3. (Front-Door-Criterion) For Pearl's example, show that we can identify the effects  $D \rightarrow M$  by conditioning on an empty set and  $M \rightarrow Y$  by conditioning on  $D$ . Combining the two results, we can identify the total effect of  $D$  on  $Y$  (this is known as the Front-Door criterion). An interesting open problem is to figure out credible economic applications of this approach. This is a nice exercise to solve analytically (compare your results against causal identification packages).
- 4. Add an arrow  $Z_2 \rightarrow Z_1$  in Pearl's example and figure out how to identify the effect of  $D \rightarrow Y$  by conditioning, of  $D \rightarrow M$  by conditioning, and of  $M \rightarrow Y$  by conditioning. (Note that valid conditioning sets may be empty.) Can you identify the effect of  $X_2 \rightarrow Y$ ? If so, how? You may solve this analytically or using any of the causal identification packages.
- 5. Add an arrow  $X_1 \rightarrow M$  in Pearl's example and figure out how to identify the effect of  $D \rightarrow Y$  by conditioning, of  $D \rightarrow M$  by conditioning, and of  $M \rightarrow Y$  by conditioning. Can you identify the effect of  $X_2 \rightarrow Y$ ? If so, how? You may solve this analytically or using any of the causal identification packages.
- 6. (Advanced). Study the Verma and Pearl paper. Work through the definitions and statement of main results (Theorem 2), and write down a brief discussion. If you can figure out a much shorter proof of their main result, it could be a publishable paper.

## 8.A Counterfactual Distributions via Markov Networks

Interventions induce new counterfactual distributions for endogenous variables. We can readily compute these distributions from the definitions of interventions, as illustrated in the following for the do-intervention.

**Example 8.A.1** (Counterfactual Law for Do Intervention in LS-DAG) We can write the counterfactual distribution of  $Y(d), Z, X$  in terms of the factual distribution as

$$p(y, z, x : \text{do}(d)) = p(y|d, x) p(z) p(x).$$

Indeed,

$$p(y, z, x : \text{do}(d)) = p(y|z, x : \text{do}(d)) p(z, x : \text{do}(d)),$$

by definition and Bayes' law. We also have  $p(y|z, x : \text{do}(d)) = p(y|d, x)$  and  $p(z, x : \text{do}(d)) = p(z, x)$  by the definition of the counterfactual ASEM, and  $p(z, x) = p(z)p(x)$  by independence of  $Z$  and  $X$ .

**Theorem 8.A.1** (Counterfactual Law Induced by the Do Intervention) *The induced law  $p_{X^*}$  of the counterfactual variables  $X^* = (X_\ell^*)_{\ell \in V \setminus j}$  induced by  $\text{do}(X_j = x_j)$  can be stated in terms of the factual law as follows:*

$$p(\{x_\ell\}_{\ell \in V \setminus j} : \text{do}(x_j)) := p_{X^*}(\{x\}_{\ell \in V \setminus j}) = \prod_{\ell \in V \setminus j} p(x_\ell | pa_j),$$

where  $\{x\}_{\ell \in V \setminus j}$  denotes the point where the density function is evaluated,  $pa_j$  denotes the parental values under the new edge structure, and  $p$  denotes the factual law.

The result follows immediately from the Markov factorization property and the definition of counterfactuals under the do intervention. This characterization is interesting in its own right, because it can be used for identification and inference on the counterfactual laws directly, provided that we are willing to model the distribution of the variables. The use of Bayesian methods can be fruitful for this purpose.

These type of formulas are often called "g-formulas" and first appeared in the work [13] of James Robins in 1986 (using another "tree-based" form of causal graphs).

## 8.B Causal Discovery Algorithms

### PC Algorithm

Let the true structure be as in Figure 8.21-A . By  $d$ -separation, this structure implies  $X \perp\!\!\!\perp Y$ , and that  $X$  and  $Y$  are each

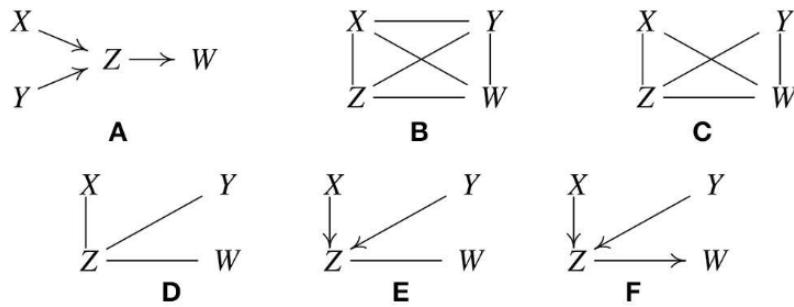


Figure 8.21: Illustration of PC Algorithm due to Glymour et al

independent of  $W$  conditional on  $Z$ , written  $\{X, Y\} \perp\!\!\!\perp W | Z$ . Suppose we have a statistical testing procedure that is able to determine these relations. The PC algorithm relies on the idea that under the faithfulness assumption and no latent nodes, two variables are directly causally related if and only if there does not exist any subset of the remaining variables conditioning on which they are independent ([10]). The PC algorithm works as follows, which we reproduce verbatim following [14].

1. Form a complete undirected graph (Figure 8.21-B).
2. Eliminate edges between variables that are unconditionally independent; in this case that is the  $X - Y$  edge (Figure 8.21-C)
3. For each pair of variables  $(A, B)$  having an edge between them, and for each variable  $C$  with an edge connected to either of them, eliminate the edge between  $A$  and  $B$  if  $A \perp\!\!\!\perp B | C$  (Figure 8.21-D)
4. For each pair of variables  $A, B$  having an edge between them, and for each pair of variables  $\{C, D\}$  with edges both connected to  $A$  or both connected to  $B$ , eliminate the edge between  $A$  and  $B$  if  $A \perp\!\!\!\perp B | \{C, D\}$ .

We then continue checking independencies conditional on subsets of variables of increasing size  $n$  until there are no more adjacent pairs  $(A, B)$ , such that there is a subset of variables of size  $n$  such that all of the variables in the subset are adjacent to  $A$  or all adjacent to  $B$ . In the considered example,  $Z$  and  $W$  are not independent conditional on  $X$  or on  $Y$  or on both  $X$  and  $Y$ , so there are no further statistical decisions to make. Similarly for  $X$  and  $Z$ , and for  $Y$  and  $Z$ .

5. For each triple of variables  $(A, B, C)$  such that  $A$  and  $B$  are adjacent,  $B$  and  $C$  are adjacent, and  $A$  and  $C$  are not adjacent, orient the edges  $A - B - C$  as  $A \rightarrow B \leftarrow C$ , if  $B$  was not in the set conditioning on which  $A$  and  $C$  became independent and the edge between them was accordingly eliminated. Such triple of variables are called  $v$ -structures.

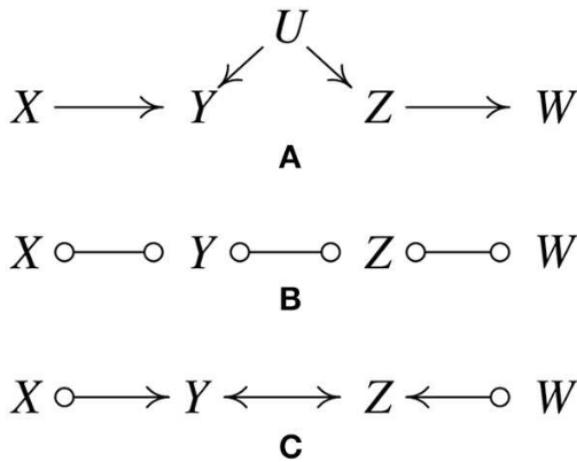


Figure 8.22: Illustration of FCI Algorithm due to [14].

In the example,  $Z$  was not conditioned on in eliminating the  $X - Y$  edge, so orient  $X - Z - Y$  as  $X \rightarrow Z \leftarrow Y$  (Figure 8.21-E).

6. For each triple of variables such that  $A \rightarrow B - C$ , and  $A$  and  $C$  are not adjacent, orient the edge  $B - C$  as  $B \rightarrow C$ . This is called orientation propagation.

In Figure 8.21-F,  $Y \rightarrow Z - W$  is oriented as  $Y \rightarrow Z \rightarrow W$ . In this example, the true structure is recovered uniquely.

## FCI Algorithm

An important extension of the PC algorithm is the Fast Causal Inference (FCI) Algorithm ([10]), which allows for unmeasured confounding variables. In what follows we reproduce, nearly verbatim, an example due to [14] to illustrate the workings of this algorithm.

Figure 8.22-A presents the true structure, where  $U$  is an unmeasured confounder. A in the PC procedure, FCI uses tests of statistical independence to eliminate edges, yielding 8.22-B. The "o" mark means it can be an arrow head or an arrow tail. FCI orients edges by a procedure similar to PC, but without assuming that every edge is directed one way or the other. The  $X \circ - Z$  edge is eliminated, because  $X$  and  $Z$  are unconditionally independent; the  $X \circ - o Y \circ - o Z$  triple is therefore oriented as a collider,  $X \circ \rightarrow Y \leftarrow o Z$ . In the same way,  $Y \circ - o Z \circ - o W$  is found to be a collider,  $Y \circ \rightarrow Z \leftarrow o W$ , yielding Figure 2C.

The bidirected edge between  $Y$  and  $Z$  indicates that there is at least one unmeasured confounder of  $Y$  and  $Z$ . The remaining "o" symbols at  $X$  and  $W$  indicate that the algorithm cannot tell whether the  $X$  to  $Y$  connection is a directed edge from  $X$  to  $Y$ ,

or an unmeasured confounder, or both; the same for the  $W$  to  $Z$  connection.

In contrast to this example, in which one can determine that there is at least one unmeasured confounder of  $Y$  and  $Z$ , there are other situations in which one can exclude the possibility of having confounders. For instance, consider the causal graph in Figure 8.21-A. Then in the output of FCI, we know that there cannot be any confounder of  $Z$  and  $W$ , because otherwise  $X$  and  $W$  cannot be independent conditioning on  $Z$  ( $X$  and  $W$  are not  $d$ -separated by  $Z$  if  $Z$  and  $W$  have a confounder).

# Bibliography

- [1] Judea Pearl. ‘Causal diagrams for empirical research’. In: *Biometrika* 82.4 (1995), pp. 669–688 (cited on page 158).
- [2] Trygve Haavelmo. ‘The probability approach in econometrics’. In: *Econometrica: Journal of the Econometric Society* 12 (1944), pp. iii–vi+1–115 (cited on pages 158, 160).
- [3] James Heckman and Rodrigo Pinto. ‘Causal analysis after Haavelmo’. In: *Econometric Theory* 31.1 (2015 (NBER 2013)), pp. 115–151 (cited on pages 158, 163).
- [4] Thomas S. Richardson and James M. Robins. *Single world intervention graphs (SWIGs): A unification of the counterfactual and graphical approaches to causality*. Working Paper No. 128, Center for the Statistics and the Social Sciences, University of Washington. 2013. URL: <https://csss.uw.edu/files/working-papers/2013/wp128.pdf> (cited on pages 158, 163, 174).
- [5] Alfred Marshall. *Principles of economics: unabridged eighth edition*. Cosimo, Inc., 2009 (cited on page 161).
- [6] Philip G. Wright. *The tariff on animal and vegetable oils*. New York: The Macmillan company, 1928 (cited on page 163).
- [7] Judea Pearl. *Causality*. Cambridge university press, 2009 (cited on pages 164, 170, 171, 177, 179).
- [8] Thomas Verma and Judea Pearl. *Influence diagrams and d-separation*. Tech. rep. Cognitive Systems Laboratory, Computer Science Department, UCLA, 1988 (cited on pages 168, 169).
- [9] Tyler J. VanderWeele and Ilya Shpitser. ‘A new criterion for confounder selection’. In: *Biometrics* 67.4 (2011), pp. 1406–1413 (cited on pages 175, 177, 178).
- [10] Peter Spirtes, Clark N. Glymour, Richard Scheines, and David Heckerman. *Causation, prediction, and search*. MIT press, 2000 (cited on pages 181, 186, 187).
- [11] Caroline Uhler, Garvesh Raskutti, Peter Bühlmann, and Bin Yu. ‘Geometry of the faithfulness assumption in causal inference’. In: *The Annals of Statistics* 41.2 (2013), pp. 436–463 (cited on pages 181, 182).
- [12] Timothy G. Conley, Christian B. Hansen, and Peter E. Rossi. ‘Plausibly exogenous’. In: *Review of Economics and Statistics* 94.1 (2012), pp. 260–272 (cited on page 181).

- [13] James Robins. '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* 7.9-12 (1986), pp. 1393–1512 (cited on page 185).
- [14] Clark Glymour, Kun Zhang, and Peter Spirtes. 'Review of causal discovery methods based on graphical models'. In: *Frontiers in genetics* 10 (2019), p. 524 (cited on pages 186, 187).

# Statistical Inference on Predictive and Causal Effects in Modern Nonlinear Regression Models

# 9

Here we discuss debiased machine learning (DML) methods for performing inference on average predictive or causal effects in two important classes of models: partially linear regression models and interactive regression models. We also present a general DML method for performing inference on a low-dimensional target parameter in the presence of high dimensional nuisance parameters that are learned using ML methods. Two case studies illustrate the approach.

9.1 Introduction . . . . .	192
9.2 DML Inference in the Partially Linear Regression Model (PLM) . . . . .	193
Discussion of DML Construction . . . . .	197
The Effect of Gun Ownership on Gun-Homicide Rates . . . . .	200
9.3 DML Inference in the Interactive Regression Model (IRM) . . . . .	202
DML Inference on APEs and ATEs . . . . .	202
DML Inference for GATEs and ATET . . . . .	205
The effect of 401(k) Eligibility on Net Financial Assets	206
9.4 Generic Debiased (or Double) Machine Learning .	210
Key Ingredients . . . . .	210
Neyman Orthogonal Scores for Regression Problems	213
The DML Inference Method . . . . .	214
Properties of the general DML estimator . . . . .	216
9.5 Bias Bounds with Proxy Treatments . . . . .	221
9.6 Illustrative Neyman Orthogonality Calculations . . . . .	222

## 9.1 Introduction

We recall the predictive effect question:

- ▶ How does the predicted value of outcome,

$$\mathbb{E}[Y | D, X],$$

change if a regressor value  $D$  increases by a unit, while regressor values  $X$  remain unchanged?

This question may have a causal interpretation within any SEM, where conditioning on  $X$  is sufficient for identification of the causal effect of  $D$  on  $Y$ . When this condition holds, the question becomes the causal effect question:

- ▶ How does the predicted value of potential outcome,

$$\mathbb{E}[Y(d) | X],$$

change if we intervene and change the treatment value  $d$  by a unit, conditional on the observed  $X$ ?

Both questions are interesting and useful to ask, depending on the application. In what follows, we set up debiased machine learning (DML) methods for answering these questions with data. These statistical inference methods *do not* distinguish between the two types of questions, so the methods are equally applicable to answering both types.

Here we discuss DML methods for performing inference on average predictive or causal effects in two important classes of nonlinear regression models. After presenting these two special cases, we also present a general DML method for performing inference on a low-dimensional target parameter in the presence of high-dimensional nuisance parameters that are learned using ML methods.

The DML method requires a Neyman-orthogonal representation of the target parameters to reduce the spillover of regularization biases inherent in ML methods onto the estimation of the target parameter. The method also makes use of cross-fitting: an efficient form of sample splitting that eliminates biases that may arise from overfitting.

To illustrate the general principles, we provide two case studies. In the first, we perform inference on the effect of gun ownership on homicide rates. In the second, we perform inference on the effect of 401(k) eligibility on financial assets.

## 9.2 DML Inference in the Partially Linear Regression Model (PLM)

We first answer the questions posed above within the context of the partially linear regression model:

$$Y = \beta D + g(X) + \epsilon, \quad E[\epsilon | D, X] = 0, \quad (9.2.1)$$

where  $Y$  is the outcome variable,  $D$  is the regressor of interest, and  $X$  is a high-dimensional vector of other regressors or features, called "controls." The coefficient  $\beta$  answers the predictive effect question. In this segment we discuss estimation and confidence intervals for  $\beta$ . We also provide a case study, in which we examine the effect of gun ownership on homicide rates.

The model allows a part of the regression function,  $g(X)$ , to be fully nonlinear, which generalizes the approach from Chapter 4. However, the model is still not fully general, because it imposes additivity in  $g(X)$  and  $D$ . We shall consider the fully unrestricted model in Section 9.3, where we analyze the fully interactive regression model in the context of a binary treatment  $D$ . It is worth pointing out though that the partially linear model is not as restrictive as it appears at a first sight since we can consider explicit interactions within the partially linear framework.

**Remark 9.2.1** (Interactions within PLM) Given a raw treatment and a set of controls,  $\bar{D}$  and  $Z$ , we can create the technical treatment  $D := \bar{D}P(Z)$ , where  $P(Z)$  is an  $L$ -dimensional dictionary of transformations of  $Z$ . For example,  $P(Z)$  could be indicators of various subgroups. Then we can consider the model

$$Y = \sum_{l=1}^L \beta_l D_l + g(Z) + \epsilon,$$

where  $E[\epsilon | Z, D] = 0$ . We can re-write this as

$$Y = \beta_l D_l + g_l(X_l) + \epsilon, \quad E[\epsilon | D_l, X_l] = 0,$$

where  $g_l(X_l) := \sum_{k \neq l} \beta_k D_k + g(Z)$  and  $X_l := ((D_k)_{k \neq l}, Z)$ . We therefore obtain exactly a model of the partially linear form (9.2.1). We can then apply DML methods to learn and perform inference on each element of  $(\beta_l)_{l=1}^L$  or even carry out joint inference (similarly to what we have done in Chapter 4).

In what follows, we will employ the partialling out  $X$  operation

of the form that inputs a random variable  $V$  and outputs the residualized form:

$$\tilde{V} := V - E[Y | X].$$

Applying this operation to (9.2.1) we obtain:

$$\tilde{Y} = \beta \tilde{D} + \epsilon, \quad E(\epsilon \tilde{D}) = 0, \quad (9.2.2)$$

where  $\tilde{Y}$  and  $\tilde{D}$  are the residuals left after predicting  $Y$  and  $D$  using  $X$ . Specifically, we have that

$$\tilde{Y} := Y - \ell(X), \quad \tilde{D} := D - m(X),$$

where  $\ell(X)$  and  $m(X)$  are defined as conditional expectations of  $Y$  and  $D$  given  $X$ :

$$\ell(X) := E[Y | X], \quad m(X) := E[D | X].$$

Here we recall that the conditional expectations of  $Y$  and  $D$  given  $X$  are the best predictors of  $Y$  and  $D$  using  $X$ .

The equation  $E\epsilon \tilde{D} = 0$  above is the Normal Equation for the population regression of  $\tilde{Y}$  on  $\tilde{D}$ . This equation implies the following result:

**Theorem 9.2.1** (FWL Partialling-Out for Partially Linear Model) Suppose that  $Y$ ,  $X$  and  $D$  have bounded second moments. Then the population regression coefficient  $\beta$  can be recovered from the population linear regression of  $\tilde{Y}$  on  $\tilde{D}$ :

$$\beta := \{b : E(\tilde{Y} - b \tilde{D}) \tilde{D} = 0\} := (E \tilde{D}^2)^{-1} E \tilde{D} \tilde{Y},$$

where  $\beta$  is uniquely defined if  $D$  cannot be perfectly predicted by  $X$ , i.e. if  $E \tilde{D}^2 > 0$ .

Thus,  $\beta$  can be interpreted as a regression coefficient of *residualized*  $Y$  on *residualized*  $D$ , where the residuals are defined by respectively subtracting the conditional expectation of  $Y$  given  $X$  and  $D$  given  $X$  from  $Y$  and  $D$ . This result generalizes the FWL from linear models to partially linear models.

Our estimation procedure for  $\beta$  in the sample will mimic the partialling out procedure in the population. We also rely on cross-fitting (outlined below) to make sure our estimated residualized quantities are not overfit.

### Double/Orthogonal ML for the Partially Linear Model

1. Partition data indices into random folds of approximately equal size:  $\{1, \dots, n\} = \bigcup_{k=1}^K I_k$ . For each fold  $k = 1, \dots, K$ , compute ML estimators  $\hat{\ell}_{[k]}$  and  $\hat{m}_{[k]}$  of the conditional expectation functions  $\ell$  and  $m$ , leaving out the  $k$ -th block of data. Obtain the cross-fitted residuals for each  $i \in I_k$ :

$$\check{Y}_i = Y_i - \hat{\ell}_{[k]}(X_i), \quad \check{D}_i = D_i - \hat{m}_{[k]}(X_i).$$

2. Apply ordinary least squares of  $\check{Y}_i$  on  $\check{D}_i$ , that is, obtain the  $\hat{\beta}$  as the root in  $b$  of the normal equations:

$$\mathbb{E}_n(\check{Y}_i - b\check{D}_i)\check{D}_i = 0.$$

3. Construct standard errors and confidence intervals as in standard least squares theory.

In what follows it will be convenient to use the notation

$$\|h\|_{L^2} := \sqrt{\mathbb{E}_X h^2(X)},$$

where, as before,  $\mathbb{E}_X$  computes the expectation over values of  $X$ .

**Theorem 9.2.2** (Adaptive Inference on a Target Parameter in PLM [1]) *Consider the PLM model. Suppose that estimators  $\hat{\ell}_{[k]}(X)$  and  $\hat{m}_{[k]}(X)$  provide approximations to the best predictors  $\ell(X)$  and  $m(X)$  that are of sufficiently high quality:*

$$n^{1/4}(\|\hat{\ell}_{[k]} - \ell\|_{L^2} + \|\hat{m}_{[k]} - m\|_{L^2}) \approx 0.$$

*Suppose that  $\mathbb{E}\tilde{D}^2$  is bounded away from zero; that is, suppose  $\tilde{D}$  has non-trivial variation left after partialling out. Suppose other regularity conditions listed in [1] hold.*

*Then the estimation error in  $\check{D}_i$  and  $\check{Y}_i$  has no first order effect on  $\hat{\beta}$ :*

$$\sqrt{n}(\hat{\beta} - \beta) \approx (\mathbb{E}_n \tilde{D}^2)^{-1} \sqrt{n} \mathbb{E}_n \tilde{D} \epsilon.$$

*Consequently,  $\hat{\beta}$  concentrates in a  $1/\sqrt{n}$  neighborhood of  $\beta$  with deviations approximated by the Gaussian law:*

$$\sqrt{n}(\hat{\beta} - \beta) \xrightarrow{a} N(0, V),$$

where

$$V = (\mathbb{E}\tilde{D}^2)^{-1} \mathbb{E}(\tilde{D}^2\epsilon^2)(\mathbb{E}\tilde{D}^2)^{-1}.$$

**Confidence Interval** The standard error of  $\hat{\beta}$  is  $\sqrt{\hat{V}/n}$ , where  $\hat{V}$  is an estimator of  $V$ . The result implies that the confidence interval

$$\left[ \hat{\beta} - 2\sqrt{\hat{V}/n}, \hat{\beta} + 2\sqrt{\hat{V}/n} \right]$$

covers  $\beta$  in approximately 95% of possible realizations of the sample. In other words, if our sample is not atypical, the interval covers the truth.

**Selecting the Best ML Learners of  $\ell$  and  $m$ .** There may be several methods that satisfy the quality requirements of Theorem 9.2.2, and we may therefore ask what ML methods we should use in practice. Consider a collection of ML methods indexed by  $j \in \{1, \dots, J\}$ . Our goal would be to select the methods that minimize an upper bound on the bias of the DML estimator.

The bias of the DML estimator is controlled by the mean square approximation errors (MSAE):

$$\frac{1}{K} \sum_{k=1}^K \|\hat{\ell}_{[k]} - \ell\|_{L^2}^2 \text{ and } \frac{1}{K} \sum_{k=1}^K \|\hat{m}_{[k]} - m\|_{L^2}^2. \quad (9.2.3)$$

Therefore, we can select the best ML method for estimating  $m$  and the best method for estimating  $\ell$  to minimize the upper bound on the bias. We will be using mean square prediction errors as proxies for MSAEs.

#### Selection of the Best ML Methods for DML to Minimize Bias.

Consider a set of ML methods enumerated by  $j \in \{1, \dots, J\}$ .

- For each method  $j$ , compute the cross-fitted MSPEs

$$\mathbb{E}_n \check{Y}_{i,j}^2 \text{ and } \mathbb{E}_n \check{D}_{i,j}^2,$$

where the index  $j$  reflects the dependency of residuals on the method.

- Select the ML methods  $j \in \{1, \dots, J\}$  that give the smallest MSPEs:

$$\hat{j}_\ell = \arg \min_j \mathbb{E}_n \check{Y}_{i,j}^2 \text{ and } \hat{j}_m = \arg \min_j \mathbb{E}_n \check{D}_{i,j}^2.$$

- Use the method  $\hat{j}_\ell$  as a learner of  $\ell$ , and  $\hat{j}_m$  as a learner of  $m$  in the DML algorithm above.

Two different ML methods may be the best for predicting  $Y$  and predicting  $D$ . By doing MSPE minimization we in fact minimize MSAEs, since MSPEs approximate MSAEs plus terms that do not depend on  $j$ .

Rather than selecting the single best predictors of  $Y$  and  $D$ , we can also use residuals to form linear ensembles of ML methods that minimize MSPEs.

**Corollary 9.2.3** *The previous inferential result continues to hold if the best or aggregated prediction rules are used as estimators  $\hat{m}$  and  $\hat{\ell}$  of  $m$  and  $\ell$  in the DML algorithm. A simple sufficient condition is that the number of ML prediction rules  $J$  over which we aggregate or choose from is fixed (meaning small in practice).*

In practical terms, the result of Corollary 9.2.3 means that we should only choose among or aggregate over relatively few ML methods. Otherwise, we may end up overfitting (since we are “cheating” here by using validation data to form the aggregator).

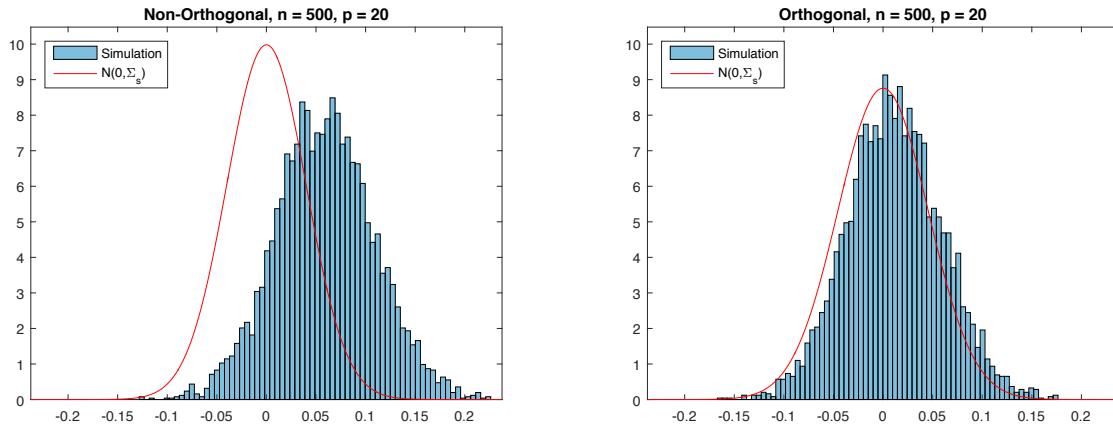
**Remark 9.2.2** (More Technical Condition) A sufficient condition for data dependent selection of which predictor to use when forming residuals to perform well in theory often boils down to requiring  $\sqrt{\log J}n^{-1/4} \approx 0$  for choosing the single best method and  $\sqrt{J}n^{-1/4} \approx 0$  when using the linear aggregation of methods. However, much work in this area is yet to be formally developed.

## Discussion of DML Construction

The partialling out operation causes the moment equations defining  $\beta$  to be Neyman-orthogonal. That is, the moment conditions are insensitive to perturbations of the nuisance parameters  $\ell$  and  $m$ .<sup>1</sup> We discussed Neyman-orthogonality in the context of high-dimensional linear regression models in Chapter 4. We return to and generalize this discussion formally in Section 9.4. This property allows us to get rid of the bias in estimation of  $m$  and  $\ell$  that arises when ML estimators are applied in high-dimensional settings.

Naive application of machine learning methods directly to outcome equations may lead to highly biased estimators, because

1: Generally we use the term nuisance parameters to name parameters that are not the target parameters. Here the target parameter is  $\beta$  and  $\ell$  and  $m$  are nuisance parameters.



**Figure 9.1:** Left: Behavior of a conventional (non-orthogonal) ML estimator. Right: Behavior of the orthogonal, DML estimator.

the resulting strategy is not Neyman-orthogonal. The biases in estimation of  $g$ , which are unavoidable in high-dimensional estimation, create a non-trivial bias in the estimate of the main effect. This bias is large enough to cause failure of conventional inference.

The left panel of Figure 9.1 illustrates the bias arising due to the use of a non-orthogonal, naive approach for learning  $\beta$ . Specifically, the figure shows the behavior of a conventional (non-orthogonal) ML estimator,  $\tilde{\beta}$ , in the partially linear model in a simple simulation experiment where we learn  $g$  using a random forest. The  $g$  in this experiment is a very smooth function of a small number of variables, so the experiment is seemingly favorable to the use of random forests a priori. The histogram shows the simulated distribution of the centered estimator,  $\tilde{\beta} - \beta$ . The estimator is badly biased, shifted much to the right relative to the true value  $\beta$ . Furthermore, the distribution of the estimator (approximated by the blue histogram) is substantively different from a normal approximation (shown by the red curve) derived under the assumption that the bias is negligible.\*

\* This biased performance of the naive estimator can also be explained analytically. The naive strategy relies on the moment equation:

$$E[(Y - \beta D - g(X))D] = 0$$

to identify  $\beta$  and uses a biased estimate of  $g$  in place of  $g$ . This moment strategy is sensitive to deviations away from the true value. Indeed, let us compute the directional derivative in the direction  $\Delta$  away from the true value:

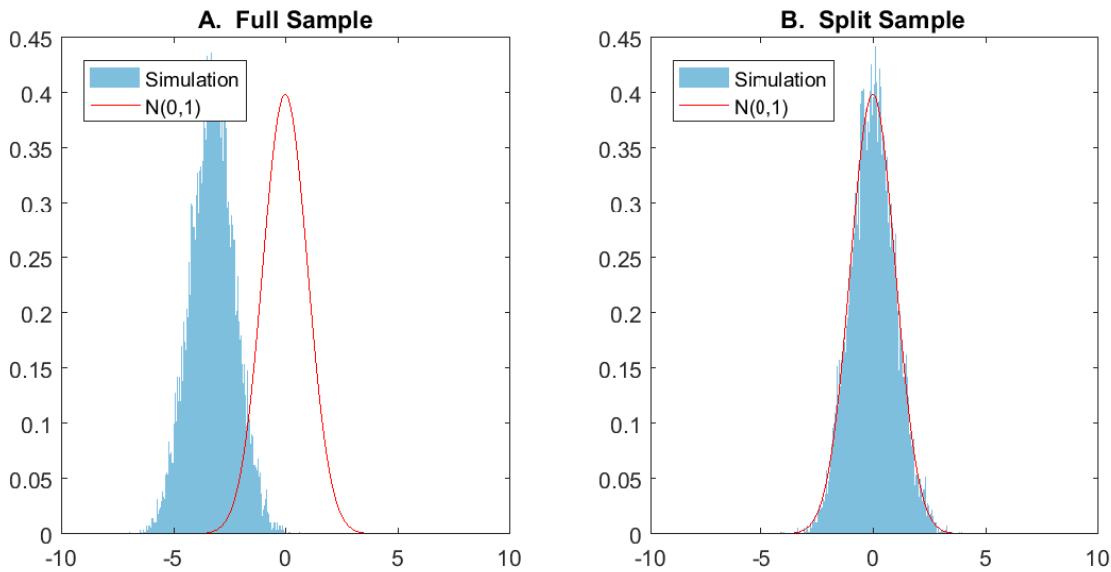
$$\partial_t E[(Y - \beta D - g(X) + t\Delta(X))D] \Big|_{t=0} = E\Delta(X)D \neq 0.$$

The derivative generally does not vanish, and the biases in estimation of  $g$  will transmit to the estimation of  $\beta$ .

The right panel of Figure 9.1 illustrates the behavior of the (Neyman) orthogonal DML estimator,  $\hat{\beta}$ , in the partially linear model in a simple experiment where we learn nuisance functions  $m$  and  $\ell$  using random forests. Note that the simulated data are exactly the same as those underlying the left panel. The simulated distribution of the centered estimator,  $\hat{\beta} - \beta$ , (given by the blue histogram) illustrates that the estimator is approximately unbiased, concentrates around  $\beta$ , and is approximately normally distributed. The low bias arises because DML uses the Neyman-orthogonal moment equations.

The DML algorithm uses a form of sample splitting, called cross-fitting, to make sure our estimated residualized quantities are not overfit. Biases arising from overfitting could result from using highly complex fitting methods such as boosting, deep neural networks and random forests. If we don't do sample splitting and the ML estimates overfit, we may end up with very large biases.

Figure 9.2 illustrates how the bias resulting from overfitting in the estimation of nuisance functions can cause the DML (without sample splitting) to be biased and how sample splitting eliminates this problem. In the left panel the histogram shows the finite-sample distribution of the DML estimator in the partially linear model where nuisance parameters are estimated with overfitting using the full sample, i.e. without sample splitting. The finite-sample distribution is clearly shifted to the left of the true parameter value, demonstrating the substantial bias. In the right panel, the histogram shows the finite-sample distribution of the DML estimator in the partially linear model where nuisance parameters are estimated with sample-splitting using the cross-fitting estimator. Here, we see that the use of sample-splitting has completely eliminated the bias induced by overfitting.



**Figure 9.2:** Left: DML distribution without sample-splitting. Right: DML distribution with cross-fitting.

**Remark 9.2.3 (On overfitting)** Note that previously in the context of high-dimensional approximately sparse linear models we were using lasso (either with the plug-in or cross-validated penalty levels) that ensure that overfitting is sufficiently well-controlled that we didn't have to use sample splitting. Such refined, theoretically rigorous choices of tuning parameters are not yet available for other machine learning methods. In practice, experienced researchers and machine learning engineers often use intuition, heuristics, and other empirical tools (six packs or witchcraft tables, for example) to set the tuning parameters. While the resulting methods can perform well for prediction purposes, even modest overfitting can result in large biases in DML, as we illustrate in the simulation experiment. Therefore, it is simply safer to rely on sample-splitting in real settings with complicated learners to make sure our estimated residualized quantities are not overfit.



**Figure 9.3:** Witchcraft tables used by some ML hackers to tune parameters. There are no known theoretical guarantees attached to this tuning method.

## The Effect of Gun Ownership on Gun-Homicide Rates

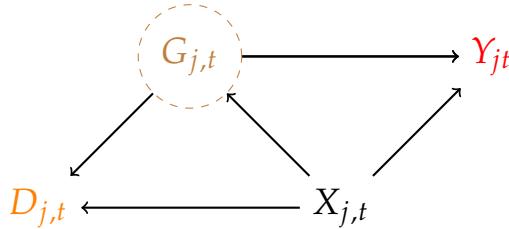
We consider the problem of estimating the effect of gun ownership on the homicide rate. For this purpose, we estimate the partially linear model:

$$Y_{j,t} = \beta D_{j,(t-1)} + g(X_{j,t}) + \epsilon_{j,t}.$$

R Notebook on DML for Impact of Gun Ownership on Homicide Rates using DNNs

$Y_{j,t}$  is the log homicide rate in county  $j$  at time  $t$ .  $D_{j,t-1}$  is the log fraction of suicides committed with a firearm in county  $j$  at time  $t - 1$ , which we use as a proxy for gun ownership  $G_{j,t}$ , which is not observed.  $X_{j,t}$  is a set of demographic and economic characteristics of county  $j$  at time  $t$ .

The intent here is that parameter  $\beta$  is an approximation of the causal effect of gun ownership  $G_{j,t}$  on homicide rates  $Y_{j,t}$ , controlling for county-level demographic and economic characteristics. We provide further detail about the use of proxy treatments in Section 9.5. To account for fixed heterogeneity across counties and time trends in all variables, we have removed county-specific and time-specific effects from all variables prior to estimation. The sample covers 195 large United States counties between the years 1980 through 1999, giving us 3900 observations.



**Figure 9.4:** A Possible DAG Structure for the Gun Ownership Example. Here we approximate the average causal effect  $G_{j,t} \rightarrow Y_{j,t}$  only if  $G_{j,t} \approx D_{j,t}$ . Under additive error of  $D_{j,t}$ , the target parameter  $\beta$  will be attenuated relative to the true causal effect; see Section 9.5.

Control variables  $X_{j,t}$  are from the U.S. Census Bureau and contain demographic and economic characteristics of the counties such as the age distribution, the income distribution, crime rates, federal spending, home ownership rates, house prices, educational attainment, voting patterns, employment statistics, and migration rates.

As a summary statistic we first look at a simple regression of  $Y_{j,t}$  on  $D_{j,t-1}$  without controls. The point estimate is 0.282 with the confidence interval ranging from 0.17 to 0.39. These results suggest that increases in gun ownership rates are associated with (predict) gun homicide rates – if gun ownership increases by 1% relative to the estimated trend then the predicted gun homicide rate goes up by 0.28%, without controlling for time-varying county characteristics. Since our goal is to estimate the effect of gun ownership after controlling for a rich set of county characteristics, we next include the controls and estimate the model by an array of the modern regression methods that we've learned.

The table shows the estimated effects of the lagged gun ownership rate on the gun homicide rate as well as the standard error. We first focus on the Lasso method: The estimated effect is

	Estimate	Standard Error
Baseline OLS	0.282	0.065
Least Squares with controls	0.191	0.052
Lasso	0.223	0.057
Post-Lasso	0.227	0.056
CV Lasso	0.200	0.058
CV Elnet	0.206	0.057
CV Ridge	0.201	0.058
Random Forest	0.192	0.058
DNN	0.176	0.116
Best	0.219	0.057

about .22. This means that a 1% increase in gun ownership rate (as measured by the proxy) leads to a predicted near quarter percent increase in gun homicide rates. The 95% confidence interval for the effect ranges from 0.12 to 0.32. These estimates are slightly higher than the ones obtained by the Least Squares Method. Random Forest also gives similar estimates to OLS, though with somewhat wider confidence bands.

The last row of the table provides the “best” estimates. To obtain “best” estimates we evaluate the performance of predictors  $\hat{l}(X)$  and  $\hat{m}(X)$  estimated by different methods on auxiliary samples using the main sample. Then we pick the methods giving the lowest MSE. In our case ridge regression and lasso give the best performances in predicting  $Y_{j,t}$  and  $D_{j,t-1}$ , respectively. We then use the best methods as predictors in the estimation procedure described above. The resulting estimate of the gun ownership effect and standard error are similar to that of Lasso.

### 9.3 DML Inference in the Interactive Regression Model (IRM)

#### DML Inference on APEs and ATEs

We consider estimation of average treatment effects when treatment effects are fully heterogeneous and the treatment variable is binary. We consider vectors  $W = (Y, D, X)$  and the pair of regression equations:

$$Y = g_0(D, X) + \epsilon, \quad E[\epsilon | X, D] = 0, \quad (9.3.1)$$

$$D = m_0(X) + \tilde{D}, \quad E[\tilde{D} | X] = 0, \quad (9.3.2)$$

where the second regression equation is presented for convenience. Here  $Y$  is an outcome of interest,  $D \in \{0, 1\}$  is a binary policy or treatment variable, and  $X$  are controls/confounding factors. Since  $D$  is not additively separable in the first equation, this model is more general than the partially linear model for the case of binary  $D$ .

A common target parameter of interest in this model is the average predictive effect (APE),

$$\theta_0 = E[g_0(1, X) - g_0(0, X)].$$

This quantity is the average predictive effect of switching  $D = 0$  to  $D = 1$ . Under conditional exogeneity discussed in Chapter 5 and Chapter 6, the APE coincides with the average treatment effect (ATE) of the intervention that moves  $D = 0$  to  $D = 1$ .

The confounding factors  $X$  affect the policy variable via the propensity score  $m_0(X)$  and the outcome variable via the function  $g_0(D, X)$ . Both of these functions are unknown (except for the case of RCTs, where  $m_0(X)$  is known) and potentially complicated, and we can employ ML methods to learn them.

Our construction of the efficient estimator for ATE will be based upon the relation<sup>2</sup>

$$\theta_0 = E\varphi_0(W), \quad (9.3.3)$$

where

$$\varphi_0(W) = g_0(1, X) - g_0(0, X) + (Y - g_0(D, X))H_0$$

and

$$H_0 = \frac{1(D = 1)}{m_0(X)} - \frac{1(D = 0)}{1 - m_0(X)}$$

is the Horvitz-Thompson transformation.

2: This representation is known as "doubly robust" parameterization, which refers to the fact that  $\theta_0$  is recovered whenever the  $g$  or  $H$  is specified correctly. We don't dwell on this property here – for us, only the Neyman orthogonality property is important.

**Remark 9.3.1** (Regression Adjustment or Propensity Score Reweighting? Use both) We realize that this representation encompasses two equally valid representations of the target parameter: the regression adjusted representation,

$$\theta_0 = E[g_0(1, X) - g_0(0, X)],$$

and the propensity score reweighting representation,

$$\theta_0 = E[YH_0].$$

Unfortunately *neither* of these representations is Neyman orthogonal, making them unsuitable for plugging-in machine learning estimators. In sharp contrast, the representation (9.3.3) is Neyman orthogonal, which implies that we can readily deploy ML methods for estimation using the empirical analog of this expression coupled with cross-fitting.

The construction provided in (9.3.1) is equally applicable in cases where the propensity score  $P(D = 1|X)$  is known, as in stratified randomized experiments, and in cases where the propensity score is unknown. When the propensity score is known, the role of regression adjustment in (9.3.1) is to reduce estimation noise.

We will employ the Neyman orthogonal parameterization and cross-fitting to construct a high-quality estimator and perform statistical inference on the target parameter.

Recall we introduced Neyman orthogonality in Chapter 4. We continue this discussion formally in Section 9.4.

### DML for APEs/ATEs in IRM

1. Partition sample indices into random folds of approximately equal size:  $\{1, \dots, n\} = \cup_{k=1}^K I_k$ . For each  $k = 1, \dots, K$ , compute estimators  $\hat{g}_{[k]}$  and  $\hat{m}_{[k]}$  of the conditional expectation functions  $g_0$  and  $m_0$ , leaving out the  $k$ -th block of data, such that  $\epsilon \leq \hat{m}_{[k]} \leq 1 - \epsilon$ , and for each  $i \in I_k$  compute

$$\hat{\phi}(W_i) = \hat{g}_{[k]}(1, X_i) - \hat{g}_{[k]}(0, X_i) + (Y_i - \hat{g}_{[k]}(D_i, X_i))\hat{H}_i$$

with

$$\hat{H}_i = \frac{1(D_i = 1)}{\hat{m}_{[k]}(X_i)} - \frac{1(D_i = 0)}{1 - \hat{m}_{[k]}(X_i)}.$$

2. Compute the estimator

$$\hat{\theta} = \mathbb{E}_n \hat{\phi}(W_i)$$

3. Construct standard errors via

$$\sqrt{\hat{V}/n}, \quad \hat{V} = \mathbb{E}_n (\hat{\phi}(W_i) - \hat{\theta})^2$$

and use standard normal critical values for inference.

**Remark 9.3.2** (Trimming) An important practical issue is trimming  $|\hat{H}_i|$  from taking explosively large values. Large values can occur when estimated propensity scores are near 0 or 1, which may indicate failure of the overlap condition –

Assumption 5.2.2 in Chapter 5 and restated in Theorem 9.3.1 below. In the algorithm above,  $\hat{H}_i$  can take on the largest absolute value of  $\bar{H} = 1/\epsilon$ . Therefore, setting  $\epsilon = .01$  corresponds to  $\bar{H} = 100$ . There does not seem to be a good theoretical or practical resolution on how to do trimming.

**Theorem 9.3.1** (Adaptive Inference on ATE with DML) *Suppose conditions specified in [1] hold. In particular, suppose that the overlap condition holds, namely for some  $\epsilon > 0$  with probability 1*

$$\epsilon < m_0(X) < 1 - \epsilon.$$

If estimators  $\hat{g}_{[k]}(D, X)$  and  $\hat{m}_{[k]}(X)$  are such that  $\epsilon \leq \hat{m}_{[k]}(X) \leq 1 - \epsilon$  and provide sufficiently high quality approximations to the best predictors  $g_0(D, X)$  and  $m_0(X)$  such that

$$\|\hat{g}_{[k]} - g_0\|_{L^2} + \|\hat{m}_{[k]} - m_0\|_{L^2} + \sqrt{n}\|\hat{g}_{[k]} - g_0\|_{L^2}\|\hat{m}_{[k]} - m_0\|_{L^2} \approx 0,$$

then the estimation error in these nuisance parameter has no first order effect on  $\hat{\theta}$ :

$$\sqrt{n}(\hat{\theta} - \theta_0) \approx \sqrt{n}\mathbb{E}_n(\varphi_0(W) - \theta_0).$$

Consequently, the estimator concentrates in  $1/\sqrt{n}$  neighborhood of  $\theta_0$ , with deviations controlled by the Gaussian law:

$$\sqrt{n}(\hat{\theta} - \theta_0) \xrightarrow{\text{a}} N(0, V)$$

where

$$V = E(\varphi_0(W) - \theta_0)^2.$$

The condition on the quality of estimators of  $\ell_0$  and  $m_0$  provides a possibility of "trading off" the quality of each estimator while retaining the adaptive inference property. The better we estimate the propensity score  $m_0$ , the worse our estimate of the regression function  $g_0$  can be; and vice versa.

## DML Inference for GATEs and ATET

We can also be interested in group ATEs (GATEs):

$$\theta_0 = E[g_0(1, X) - g_0(0, X)|G = 1],$$

where  $G$  is a group indicator defined in terms of  $X$ 's. For example, we might be interested in the impact of a vaccine on teenagers, in which case we could set  $G = 1(13 \leq \text{Age} \leq 19)$ , or on older individuals, in which case we might set  $G =$

$1(65 \leq Age)$ . DML estimation and inference for GATEs can be carried out similarly to estimation and inference for the ATE by exploiting the relation

$$\theta_0 = E[\varphi_0(X)|G = 1] = E[\varphi_0(X)G]/P(G = 1).$$

GATEs are of interest for describing heterogeneity of the average treatment effects across groups. This parameter also has a predictive interpretation in a non-causal sense: It measures the average change in prediction as  $D$  switches from 0 to 1, averaging over characteristics of the group  $G = 1$ .

Another common target parameter is the average treatment effect on the treated (ATET):

$$\theta_0 = E[g_0(1, X) - g_0(0, X)|D = 1].$$

In business applications, the ATET is often of the interest for attribution calculations. For example, if the treatment of interest is having experience with a new product, the ATET captures the effect of the new product on those that actually received it.

The construction of DML estimators for GATEs and ATETs is given in Section 9.4.

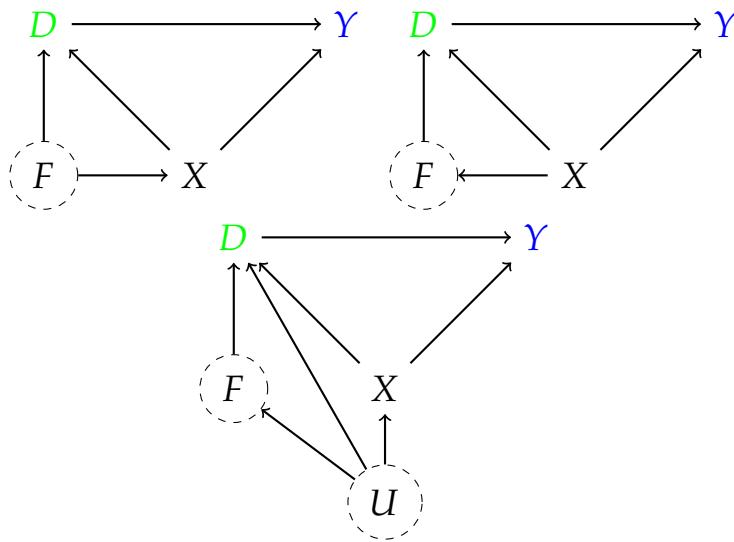
## The effect of 401(k) Eligibility on Net Financial Assets

Here we re-analyze the impact of 401(k) eligibility on financial assets (Poterba et al., [2] and [3]). The data covers a short period a few years after the introduction of 401(k)'s when they were rapidly increasing in popularity.

R Notebook on DML for Impact of 401(K) Eligibility on Financial Wealth

The key problem in determining the effect of 401(k) eligibility is that working for a firm that offers access to a 401(k) plan is not randomly assigned. To overcome the lack of random assignment, we follow the strategy developed in [2] and [3]. In these papers, the authors use data from the 1991 Survey of Income and Program Participation and argue that eligibility for enrolling in a 401(k) plan in this data can be taken as exogenous after conditioning on a few observables of which the most important for their argument is income.

The basic idea of their argument is that, at least around the time 401(k)'s initially became available, people were unlikely to be basing their employment decisions on whether an employer offered a 401(k) but would instead focus on income and other



**Figure 9.5:** Three Causal DAGs for analysis of the 401(K) example in Which adjusting for  $X$  is a valid identification strategy. The bottom figure encompasses the other two as special cases.

aspects of the job. Following this argument, whether one is eligible for a 401(k) may then be taken as exogenous after appropriately conditioning on income and other control variables related to job choice.

A key component of the argument underlying the exogeneity of 401(k) eligibility is that eligibility may only be taken as exogenous after conditioning on income and other variables related to job choice that may correlate with whether a firm offers a 401(k). [2] and [3] and many subsequent papers adopt this argument but control for parsimonious, pre-specified functions of what they deem to be relevant characteristics. One might wonder whether such specifications are able to adequately control for income and other related confounders. At the same time, the power to learn about treatment effects decreases as one allows more flexible models. The principled use of flexible ML tools offers one resolution to this tension.

In what follows, we use net financial assets<sup>3</sup> as the outcome variable,  $Y$ , in the analysis. The treatment variable,  $D$ , is an indicator for being eligible to enroll in a 401(k) plan. The vector of raw covariates,  $X$ , consists of age, income, family size, years of education, a married indicator, a two-earner status indicator, a defined benefit pension status indicator, an IRA participation indicator, and a home ownership indicator.

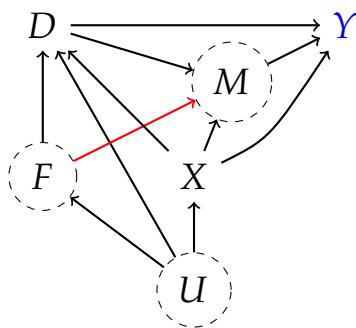
It is useful to think about a causal diagram that represents our thinking about identification in this example. In Figure 9.5, we provide three example DAGs for  $Y$ , the outcome;  $D$ , the 401(K) eligibility offer which depends on firm characteristics,  $F$ , which are not observed; and  $X$ , the worker characteristics. In one structure,  $F$  determines the workers characteristics (via

Compare this argument to the one given below using DAGs.

3: Defined as the sum of IRA balances, 401(k) balances, checking accounts, U.S. saving bonds, other interest-earning accounts in banks and other financial institutions, other interest-earning assets (such as bonds held personally), stocks, and mutual funds less non-mortgage debt.

the hiring decision), so we have  $F \rightarrow X$ . In another structure, workers determine the characteristics of the company they choose to work at,  $X \rightarrow F$ . Finally, in the last structure  $F$ ,  $X$ , and  $D$  are jointly determined by a set of latent factors  $U$ . In any of these cases,  $X$  a valid adjustment set because it is the only parent of  $Y$  (other than  $D$ ).

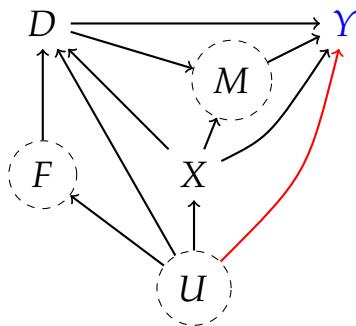
It is also useful to consider structures that would break down the identification strategy. We illustrate two such structures in Figures 9.6 and 9.7. In these figures, we introduce a node for the employer match amount,  $M$ ,<sup>4</sup> which could mediate the effect of 401(k) eligibility and have an important effect on financial wealth.



R Notebook on Dagitty-Based Identification in 401(K) Example

4: Employers often offer a benefit where they will match a proportion of an employee's contribution to their 401k, up to a limit. The limit is referred to as the employer match amount, and averages between 4 and 5% of employee's salaries.

**Figure 9.6:** A DAG Structure where adjusting for  $X$  is not sufficient. If there is no arrow from  $F$  to  $M$ , adjusting for  $X$  is sufficient.



**Figure 9.7:** Another DAG Structure where adjusting for  $X$  is not sufficient. Here the latent confounder  $U$  affects all variables, so even in the absence of an arrow connecting  $F$  to  $M$ , causal effects cannot be determined after adjusting for  $X$ . The presence of such latent confounders is always a threat to causal interpretability of any observational study.

In Figure 9.6, we suppose that  $M$  is determined by unobserved firm characteristics,  $F$ , and worker characteristics,  $X$ . In this case, adjustment for  $X$  is not sufficient as there is a path from latent firm characteristics, which are related to the treatment, to the outcome that is not closed by  $X$ . However, if  $M$  is determined solely by  $D$  and  $X$  so the red arrow is erased, adjustment for  $X$  is sufficient. Therefore, interpreting our the target parameter of our estimation strategy as a causal effect is only valid if the match amount is independent of  $F$  given  $D$  and  $X$ , that is, if there is no arrow from  $F$  to  $M$  in the graph. Otherwise, the default interpretation is that we are estimating predictive effects of 401(k) eligibility.

In the second example, Figure 9.7, we maintain the assumption that  $M$  is independent of  $F$  given  $D$  and  $X$  by eliminating the

	Lasso	Forest	Boost	NNet	Ens	Best
<i>A. Interactive Regression Model</i>						
ATE	7993 [1201]	8105 [1242]	7713 [1155]	7788 [1238]	7839 [1134]	7753 [1237]
<i>B. Partially Linear Regression Model</i>						
ATE	8871 [1298]	9247 [1295]	9110 [1314]	9038 [1322]	9166 [1299]	9215 [1294]

**Note:** Estimated ATE and standard errors from a partially linear model (Panel B) and heterogeneous effect model (Panel A) based on orthogonal estimating equations. Column labels denote the method used to estimate nuisance functions.

**Table 9.1:** Estimated Effect of 401(k) Eligibility on Net Financial Assets

arrow between nodes  $F$  and  $M$ . However, we now allow for the possibility that latent variables  $U$  have a direct effect on  $Y$ ; that is, we have an unobserved confounder or omitted variable. In this example, such a confounder may be unobserved risk preferences that relate to an individual's preference over jobs, an individual's characteristics, but also have direct effects on savings decisions not channeled purely through observed individual or job characteristics. In general, the possibility of latent confounders always poses a challenge to obtaining estimates of causal effects in non-experimental data. The presence or absence of latent confounders cannot be determined solely from the data in general, and thus their presence must be argued against based on scientific and institutional knowledge in different contexts. See, e.g., discussion in the original papers, [2] and [3], underlying this example. As in the previous example, we must interpret our estimates as predictive effects of 401(k) eligibility if we believe the connection from  $U$  to  $Y$  exists.

In Table 9.1, we report DML estimates of ATE of 401(k) eligibility on net financial assets both in the partially linear model and the interactive regression model allowing for heterogeneous treatment effects. To reduce the disproportionate impact of extreme propensity score weights in the interactive model, we trim the propensity scores at 0.01 and 0.99.

Turning to the results, it is first worth noting that when no controls are used, the estimated ATE of 401(k) eligibility on net financial assets is \$19,559 with an estimated standard error of 1413. Of course, this number is not a valid estimate of the causal effect of 401(k) eligibility on financial assets if there are neglected confounding variables as suggested by [2] and [3]. When we turn

to the estimates that flexibly account for confounding reported in Table 9.1, we see that they are substantially attenuated relative to this baseline that does not account for confounding, suggesting much smaller causal effects of 401(k) eligibility on financial asset holdings.

It is interesting and reassuring that the results obtained from the different flexible methods are broadly consistent with each other. This similarity is consistent with the theory that suggests that results obtained through the use of orthogonal estimating equations and any method that provides sufficiently high-quality estimates of the necessary nuisance functions should be similar. Finally, it is interesting that these results are also broadly consistent with those reported in the original work of [2] and [3] which used a simple, intuitively-motivated functional form, suggesting that this intuitive choice was sufficiently flexible to capture much of the confounding variation in this example.

Finally, we can conclude the discussion with a more sobering note that there are credible deviations in the graph structure (e.g. unobserved firm characteristics may affect the match amount) that challenges causal interpretation of the estimates. One approach to dealing with such deviations would be to conduct thorough sensitivity analysis.<sup>†</sup>

## 9.4 Generic Debiased (or Double) Machine Learning

### Key Ingredients

A general construction upon which DML estimation and inference can be built relies on a method-of-moments estimator for some low-dimensional target parameter  $\theta_0$  based upon the empirical analog of the moment condition

$$\mathbb{E}\psi(W; \theta_0, \eta_0) = 0, \quad (9.4.1)$$

where we call  $\psi$  the score function,  $W$  denotes a data vector,  $\theta_0$  denotes the true value of a low-dimensional parameter of interest, and  $\eta$  denotes nuisance parameters with true value  $\eta_0$ .

---

<sup>†</sup>We have done some informal simulations to assess the impact of this threat (using the observation that firms match up to 5% of income), and we estimated the size of the bias to be in the ball park of 10%. Given this, we believe the results reported here are reasonable approximations to the causal effects.

The first key input of the generic DML procedure is using a score function  $\psi(W; \theta, \eta)$  such that

$$M(\theta, \eta) = E\psi(W; \theta, \eta)$$

identifies  $\theta_0$  when  $\eta = \eta_0$  – that is,

$$M(\theta, \eta_0) = 0 \text{ if and only if } \theta = \theta_0 -$$

and the Neyman orthogonality condition is satisfied:

$$\partial_\eta M(\theta_0, \eta) \Big|_{\eta=\eta_0} = 0. \quad (9.4.2)$$

Here, (9.4.2) ensures that the moment condition (9.4.1) used to identify and estimate  $\theta_0$  is insensitive to small perturbations of the nuisance function  $\eta$  around  $\eta_0$ .

**Remark 9.4.1** The orthogonality condition is named after Neyman [4], because he was the first to propose it in the context of parametric models with nuisance parameters that are estimated at slower than  $1/\sqrt{n}$  rates.

Using a Neyman-orthogonal score eliminates the first order biases arising from the replacement of  $\eta_0$  with a ML estimator  $\hat{\eta}_0$ . Eliminating this bias is important because estimators  $\hat{\eta}_0$  must be heavily regularized in high dimensional settings, so these estimators will be biased in general. The Neyman orthogonality property is responsible for the adaptivity of these estimators – namely, their approximate distribution will not depend on the fact that the estimate  $\hat{\eta}_0$  contains error as long as the error is sufficiently mild.

**Remark 9.4.2** (Definition of the Derivative) The derivative  $\partial_\eta$  denotes the pathwise (Gateaux) derivative operator. Formally it is defined via usual derivatives taken in various directions: Given any "admissible" direction  $\Delta = \eta - \eta_0$  and scalar deviation amount  $t$ , we have that

$$\partial_\eta M(\theta, \eta)[\Delta] := \partial_t M(\theta, \eta + t\Delta) \Big|_{t=0}.$$

The statement

$$\partial_\eta M(\theta_0, \eta_0) = 0$$

means that  $\partial_\eta M(\theta_0, \eta_0)[\Delta] = 0$  for any admissible direction  $\Delta$ . The direction  $\Delta$  is admissible if  $\eta_0 + t\Delta$  is in the parameter space for  $\eta$  for all small values of  $t$ .

The second key input is the use of high-quality machine learning estimators of the nuisance parameters. A sufficient condition in the examples given includes the requirement

$$n^{1/4} \|\hat{\eta} - \eta_0\|_{L^2} \approx 0.$$

Different structured assumptions on  $\eta_0$  allow us to use different machine-learning tools for estimating  $\eta_0$ . For instance,

- 1) approximate sparsity for  $\eta_0$  with respect to some dictionary calls for the use of lasso, post-lasso, or other sparsity-based techniques;
- 2) well-approximability of  $\eta_0$  by trees calls for the use of regression trees and random forests;
- 3) well-approximability of  $\eta_0$  by sparse deep neural nets calls for the use of  $\ell_1$ -penalized deep neural networks;
- 4) well-approximability of  $\eta_0$  by at least one model mentioned in 1)-3) above calls for the use of an ensemble/best choice method over the estimation methods mentioned in 1)-3).

There are performance guarantees for most of these ML methods that make it possible to satisfy the conditions stated above. Ensemble and best choice methods ensure that the performance guarantee is no worse than the performance of the best method.

The third key input is to use a form of sample splitting at the stage of producing the estimator of the main parameter  $\theta_0$ , which allows us to avoid *biases* arising from overfitting.

Overfitting can easily occur when using highly complex fitting methods such as boosting, random forests, deep nets, ensembles, and other hybrid machine learning methods. We may heuristically think of overfitting as capturing noise that is particular to the observations used to fit a model in addition to signal. Using overfit estimates of nuisance parameters obtained using the same data as used to estimate the target parameter then heuristically leads to estimation error in these parameters being correlated to outcomes which introduces a type of bias. This bias can be very large, as illustrated in Figure 9.2, if the

ML estimates overfit. We specifically use cross-fitted forms, i.e. sample splitting, of the empirical moments, as detailed below, in estimation of  $\theta_0$  to avoid this problem.

## Neyman Orthogonal Scores for Regression Problems

**Scores for Partially Linear Regression Model.** In the PLM, we employ the score function

$$\begin{aligned}\psi(W; \theta, \eta) := & \\ & \{Y - \ell(X) - \theta(D - m(X))\}(D - m(X)),\end{aligned}\tag{9.4.3}$$

where  $W = (Y, D, X)$  is a data vector, and  $\eta$  is the nuisance parameter  $\eta = (\ell, m)$  with true value  $\eta_0 = (\ell_0, m_0)$ . Here,  $\ell$  and  $m$  are square-integrable functions mapping the support of  $X$  to  $\mathbb{R}$  whose true values are given by

$$\ell_0(X) = E[Y | X], \quad m_0(X) = E[D | X].$$

The score above is Neyman orthogonal by elementary calculations delegated to Section 9.6. We also see the connections to the residualized system of equations.

**Scores for Interactive Regression Model.** For estimation of the ATE parameter in the IRM model, we employ the score

$$\begin{aligned}\psi_1(W; \theta, \eta) := & (g(1, X) - g(0, X)) \\ & + H(D, X)(Y - g(D, X)) - \theta,\end{aligned}\tag{9.4.4}$$

where

$$H(D, X) := \frac{D}{m(X)} - \frac{(1 - D)}{1 - m(X)},\tag{9.4.5}$$

$W = (Y, D, X)$  is a data vector, and  $\eta := (g, m)$  is the nuisance parameter with true value  $\eta_0 = (g_0, m_0)$ . Here,  $g$  is a square-integrable function mapping the support of  $(D, X)$  to  $\mathbb{R}$ , and  $m$  is a function mapping the support of  $X$  to  $(\varepsilon, 1 - \varepsilon)$  for some  $\varepsilon \in (0, 1/2)$ . The true values of  $g$  and  $m$  are given by

$$g_0(D, X) = E[Y | D, X], \quad m_0(X) = P[D = 1 | X].\tag{9.4.6}$$

The score above is Neyman orthogonal by elementary calculations delegated to Section 9.6.

For estimation of GATEs we use the score

$$\psi(W; \theta, \eta) := \frac{G}{p} \psi_1(W; \theta, \eta); \quad (9.4.7)$$

where  $G$  denotes the group membership indicator, the nuisance parameter  $\eta$  is  $(g, m, p)$  with true value  $\eta_0 = (g_0, m_0, p_0)$  for  $g_0$  and  $m_0$  defined in (9.4.6) and  $p_0 = P(G = 1)$ , and  $\psi_1$  is the score for the ATE parameter defined in (9.4.4).

For estimation of the ATET parameter, we use the score

$$\psi(W; \theta, \eta) := H(D, X) \frac{m(X)}{p} (Y - g(0, X)) - \frac{D\theta}{p}, \quad (9.4.8)$$

where  $H(D, X)$  is given in (9.4.5), and  $\eta = (g, m, p)$  is the nuisance parameter with the true value  $\eta_0 = (g_0, m_0, p_0)$  for  $g_0$  and  $m_0$  defined in (9.4.6) and  $p_0 = P[D = 1]$ . Note that this score does not require estimating  $g_0(1, X)$ .

The scores for GATEs and ATET can be shown to be Neyman orthogonal by calculations similar to those in Section 9.6.

## The DML Inference Method

We assume that we have a sample  $(W_i)_{i=1}^n$ , modeled as i.i.d. copies of data vector  $W$ , whose law is determined by the probability measure  $P$ . Recall that  $\mathbb{E}_n$  denotes the empirical expectation:

$$\mathbb{E}_n[g(W_i)] := \frac{1}{n} \sum_{i=1}^n g(W_i).$$

Let  $\mathbb{V}_n$  denote the empirical variance:

$$\mathbb{V}_n[g(W_i)] := \mathbb{E}_n g(W_i) g(W_i)' - \mathbb{E}_n[g(W_i)] \mathbb{E}_n[g(W_i)]'.$$

### Generic DML

1. **Inputs:** Provide the data frame  $(W_i)_{i=1}^n$ , the Neyman-orthogonal score/moment function  $\psi(W, \theta, \eta)$  that identifies the statistical parameter of interest, and the name and model for ML estimation method(s) for  $\eta$ .
2. **Train ML Predictors on Folds:** Take a K-fold random partition  $(I_k)_{k=1}^K$  of observation indices  $\{1, \dots, n\}$  such that the size of each fold is about the same. For each  $k \in \{1, \dots, K\}$ , construct a high-quality machine learning estimator  $\hat{\eta}_{[k]}$  that depends only on a subset

of data  $(X_i)_{i \notin I_k}$  that excludes the  $k$ -th fold.

3. **Estimate Moments:** Letting  $k(i) = \{k : i \in I_k\}$ , construct the moment equation estimate

$$\hat{M}(\theta, \hat{\eta}) = \mathbb{E}_n[\psi(W_i; \theta, \hat{\eta}_{[k(i)]})]$$

4. **Compute the Estimator:** Set the estimator  $\hat{\theta}$  as the solution to the equation.

$$\hat{M}(\hat{\theta}, \hat{\eta}) = 0. \quad (9.4.9)$$

5. **Estimate Its Variance:** Estimate the asymptotic variance of  $\hat{\theta}$  by

$$\hat{V} = \mathbb{V}_n \hat{\phi}(W_i),$$

where

$$\hat{\phi}(W_i) = -\hat{J}_0^{-1} \psi(W_i; \hat{\theta}, \hat{\eta}_{[k(i)]})$$

and

$$\hat{J}_0 := \partial_\theta \mathbb{E}_n \psi(W_i; \hat{\theta}, \hat{\eta}_{[k(i)]}).$$

6. **Confidence Intervals:** Form an approximate  $(1 - \alpha)\%$  confidence interval for any functional  $\ell' \theta_0$ , where  $\ell$  is a vector of constants, as

$$[\ell' \hat{\theta} \pm c \sqrt{\ell' \hat{V} \ell / n}],$$

where  $c$  is the  $(1 - \alpha/2)$  quantile of  $N(0, 1)$ .

7. **Outputs:** Output the results of all steps.

**Remark 9.4.3** (The Case of Linear Scores) The score for most of our examples is linear in  $\theta$ ; that is, the score can be written as

$$\psi(W; \theta, \eta) = \psi^b(W; \eta) - \psi^a(W; \eta)\theta.$$

In such cases the estimator takes the form

$$\hat{\theta} = \hat{J}_0^{-1} \mathbb{E}_n [\psi^b(W_i; \hat{\eta}_{[k(i)]})]. \quad (9.4.10)$$

where  $\hat{J}_0 = \mathbb{E}_n \psi^a(W_i; \hat{\eta}_{[k(i)]})$ .

**Remark 9.4.4** (Sample Splitting) In step 2), the estimator  $\hat{\eta}_{[k]}$  can be an ensemble or aggregation of several estimators as long as we only use the data  $(X_i)_{i \notin I_k}$  outside the  $k$ -th fold to

construct the estimators.

**Remark 9.4.5** (Choosing the number of folds) The choice  $K \geq 4-5$  works well based on a variety of empirical examples and in simulations for medium-sized data sets. The choice  $K \geq 10$  works well for small data sets.

## Properties of the general DML estimator

We turn now to the properties of the estimator under the assumption of strong identification.

**Definition 9.4.1** (Strong Identification) We have that  $M(\theta, \eta_0) = 0$  if and only if  $\theta = \theta_0$ , and that

$$J_0 := \partial_\theta E\psi(W; \theta_0, \eta_0)$$

has singular values that is bounded away from zero.

In the context of the PLM, the latter condition is satisfied if  $E\tilde{D}^2$  is bounded away from 0, that is, if  $\tilde{D}$  has non-trivial variation left after partialing-out controls. In the context of IRM, the latter condition is satisfied if the overlap condition holds.

**Theorem 9.4.1** (Generic Adaptive Inference with DML) Assume that estimates of nuisance parameters are of sufficiently high quality, as specified in [1]. Assume strong identification holds.

Then, estimation of nuisance parameter does not affect the behavior of the estimator to the first order; namely,

$$\sqrt{n}(\hat{\theta} - \theta_0) \approx \sqrt{n}E_n\varphi_0(W),$$

where

$$\varphi_0(W) = -J_0^{-1}\psi(W; \theta_0, \eta_0), \quad J_0 := \partial_\theta E\psi(W; \theta_0, \eta_0),$$

and  $J_0 = E\psi^a(W; \eta_0)$  for linear scores.

Consequently,  $\hat{\theta}$  concentrates in a  $1/\sqrt{n}$ -neighborhood of  $\theta_0$  and the sampling error  $\sqrt{n}(\hat{\theta} - \theta_0)$  is approximately normal:

$$\sqrt{n}(\hat{\theta} - \theta_0) \stackrel{d}{\sim} N(0, V), \quad V := E\varphi_0(W)\varphi_0(W)'.$$

**Theorem 9.4.2** Under the same regularity conditions, the interval

$[\ell' \hat{\theta} \pm c \sqrt{\ell' \hat{V} \ell / n}]$  where  $c$  is the  $(1 - \alpha/2)$  quantile of a  $N(0, 1)$  contains  $\ell' \theta_0$  for approximately  $(1 - \alpha) \times 100$  percent of data realizations:

$$P\left(\ell' \theta_0 \in [\ell' \hat{\theta} \pm c \sqrt{\ell' \hat{V} \ell / n}]\right) \approx (1 - \alpha).$$

**Selection of the Best ML Methods for DML to Minimize Upper Bounds on Bias.** In many problems the nuisance parameters are regression functions

$$\eta_m = E[V_m | X_m], \quad m \in \{1, \dots, M\},$$

where  $V_m$  are some response variables and  $X_m$  are covariate vectors. Consider a set of ML methods enumerated by  $j \in \{1, \dots, J\}$  that produce estimates  $\hat{\eta}_{mj[k]}$  when applied to data excluding the  $k$ -th fold. We have that

$$\check{V}_{i,mj} = V_i - \hat{\eta}_{mj[k]}(X_i), \quad i \in I_k.$$

**Selection of the Best ML Methods for DML to Minimize Bias.**

- For each method  $j$ , compute the cross-fitted MSPEs

$$\mathbb{E}_n \check{V}_{i,mj}^2.$$

- Select the best ML method for predicting  $V_m$  via

$$\hat{j}_m = \arg \min_j \mathbb{E}_n \check{V}_{i,mj}^2.$$

- Use the method  $\hat{j}_m$  as a learner of  $\eta_m$  in the Generic DML Algorithm.

**Corollary 9.4.3** *The results of Theorems 9.4.1 and 9.4.2 continue to hold if  $J$  is small.*

The precise conditions may depend on the problem at hand. See the Remark 9.2.2 for discussion in the context of the partially linear model.

## Notebooks

- R Notebook on DML for Impact of Gun Ownership on

[Homicide Rates](#) provides application of DML inference to learn predictive/causal effects of gun ownership on homicide rates across U.S. counties.

- ▶ [R Notebook on DML for Impact of Gun Ownership on Homicide Rates using DNNs](#) provides application of DML inference-based DNNs to learn predictive/causal effects of gun ownership on homicide rates across U.S. counties.
- ▶ [R Notebook on Dagitty-Based Identification in 401\(K\) Example](#) analyses graph structures that enable identification of the causal effect of 401(K) eligibility on net financial wealth.
- ▶ [R Notebook on DML for Impact of 401\(K\) Eligibility on Financial Wealth](#) provides application of DML inference to learn predictive/causal effects of 401(K) eligibility on net financial wealth. (Note: The results produced in this notebook and provided in the text are slightly different than those in the original paper [1]. The replication files for [1] are given at the following [Github repository](#). The difference is due to our use of a single split of the sample in producing the results for this text while the results in [1] are based on a method that aggregates results across multiple data splits.)
- ▶ [R Notebook on DML for Growth Regression Analysis](#) provides application of DML inference based on ML on predictive/causal effects of countries' initial wealth on the rate of economic growth.

## Notes

For a detailed literature review and technical regularity conditions needed for each of theorems, see [1], which also gives an overview of various analytical methods for generating Neyman-orthogonal scores in a wide variety of problems.

The paper [5] goes further and describes methods for generating higher-order orthogonal scores:

$$\partial_\eta \partial_\eta E\psi(\theta_0, \eta_0) = 0.$$

The use of higher-order orthogonal scores allows even weaker requirements for the quality of machine learning estimators of

the form,

$$n^{1/6} \|\hat{\eta} - \eta_0\|_{L^2} \approx 0,$$

with the caveat that such higher-order orthogonal scores may not always exist for certain subsets of distributions.

The DML method, developed in Chernozhukov, Chetverikov, Demirer, Duflo, Hansen, Newey, and Robins [1], is simply a practical meta-recipe that explicitly incorporates many classical ideas from the parametric and semi-parametric econometrics and statistics literature; see, e.g., Neyman [4]; Bickel, Klassen, Ritov, Wellner [6]; Newey [7]; Robinson [8]; and Robins and Rotnizky [9]. The intent was to combine ideas from the classical semi-parametric learning literature and prediction methods from the modern machine learning literature to provide immediately practical methods that are ready for rigorous statistical inference on predictive and causal effects. In essence, the approach can be viewed as a modernized version of the "one"-step debiasing correction proposed by Neyman; see, e.g. [10] for a review.

The partialling-out approach has long been employed in classical econometrics. Robinson [8] was the first to employ it in the context of kernel regressions. [1] extended this approach to more modern settings where ML estimators are used for partialling out, with cross-fitting enabling the extension.

For ATE, GATEs and ATET parameters, DML (or "doubly robust" ML) reduces to the use of machine learned "doubly robust scores" with cross-fitting. The idea of using doubly robust scores (also called augmented inverse propensity score weighted scores) is due to Robins and Rotnizky [9], but also arises as a special case of Newey's [7] fundamental analysis.

Targeted maximum likelihood estimation (TMLE) is another general approach for building orthogonal estimators [11]. This approach relies on doing maximum likelihood estimation for a target parameter, using a least favorable parametric submodel for the parameter of interest as the likelihood function. As with DML, TMLE needs to be combined with cross-fitting in order to deal with general ML estimators to avoid overfitting. The DML and cross-fitted TMLE should generally produce first order equivalent answers under correct specification. However, using TMLE can refine the finite-sample properties.

In the context of ATE, TMLE can be seen as applying a calibrated correction to a nonlinear regression function. We regress  $\check{Y}_i = Y_i - \hat{g}(D_i, X_i)$  on  $\hat{H}_i$ , obtaining

$$\hat{b} = \mathbb{E}_n \check{Y}_i \hat{H}_i / \mathbb{E}_n \hat{H}_i^2.$$

Then we correct the regression function estimate by  $\bar{g}(D_i, X_i) = \hat{g}(D_i, X_i) + \hat{b}\hat{H}_i$ . This correction was first proposed by Sharfstein and Robins. The basic idea is that we know that  $Y_i - g(D_i, X_i)$  should be orthogonal to  $H_i$ . Thus, if our estimate of the regression function does not have this property, we can recalibrate the regression function so the property holds.

## Study Problems

1. Experiment with one of the notebooks for the partially linear models (Guns example, Guns with DNNs, or Growth example). For example,
  - (a) Apply the methods to a different empirical example (e.g., Penn reemployment experiment from CI-1),
  - (b) or, using the same empirical example, try to use the H2O Auto ML framework as the machine learning tool to estimate  $m$  and  $\ell$  functions. (See Chapter 7 H2O Auto ML to get started).

Explain what you are doing to a fellow student.

2. Study the 401(K) identification notebook that uses Dagitty. Extend it to another empirical example of your choice. Explain the principles you are using to a fellow student.
3. Study the 401(K) empirical analysis notebook (the part that does not deal with instrumental variables and LATE). Extend it to another empirical example of your choice (Penn reemployment experiment from Chapter 1, for example) or estimate ATE for 401(K) eligibility for a subset of low income (or high-income) workers (Group ATEs).
4. (Theoretical). Explain to a friend the concept of Neyman orthogonality, illustrating it with one of the examples in Appendix B. Extend the calculations in Appendix B to verify Neyman orthogonality for the ATET score specified in (9.4.8).
5. (Theoretical). Explain to a friend the concept of Neyman orthogonality, and explain why the formulations given in Remark 9.3.1 are not Neyman orthogonal.

## 9.5 Bias Bounds with Proxy Treatments

Here we explain the measurement error bias in the partially linear structural equation model where treatment is measured with error:

$$\begin{aligned} Y &:= \alpha G + g_Y(X) + \epsilon_Y; \\ D &:= G + g_D(X) + \epsilon_D; \\ G &:= g_G(X) + \epsilon_G; \\ X &:= \epsilon_X; \end{aligned}$$

where  $\epsilon$ 's are independent and centered. The second equation states that  $D$  is generated as a proxy for the actual treatment  $G$  using a partially linear structure. In partialled-out form

$$\begin{aligned} \tilde{Y} &:= \alpha \epsilon_G + \epsilon_Y; \\ \tilde{D} &:= \epsilon_G + \epsilon_D; \\ \tilde{G} &:= \epsilon_G. \end{aligned}$$

The projection of  $\tilde{Y}$  on  $\tilde{D}$  recovers the projection coefficient:

$$\beta = E\tilde{Y}\tilde{D}/E\tilde{D}^2 = \alpha E\epsilon_G^2/(E\epsilon_G^2 + E\epsilon_D^2).$$

It follows that there is attenuation bias in the estimable quantity  $\beta$  relative to the target parameter  $\alpha$ :

$$|\beta| < |\alpha|.$$

As the proxy error  $E\epsilon_D^2$  becomes small, the difference between  $\beta$  and  $\alpha$  becomes small. Specifically, if  $E\epsilon_D^2 \rightarrow 0$ , then  $\beta \rightarrow \alpha$ .

If we somehow knew that

$$R_{D \sim G}^2 := E\epsilon_G^2/(E\epsilon_G^2 + E\epsilon_D^2) \geq 2/3$$

that is, the true treatment  $G$  explains at least two thirds of variance of the proxy treatment  $D$  – then we could construct the upper and lower bound on  $\alpha$  from  $\beta$ . E.g. when  $\beta > 0$ , we would have

$$\beta \leq \alpha \leq \beta/R_{D \sim G}^2 = (3/2)\beta.$$

## 9.6 Illustrative Neyman Orthogonality Calculations

**The Score in the Partially Linear Model.** Consider the score for the PLM given in (9.4.3). We have that

$$\mathbb{E}\psi(W; \beta_0, \eta_0) = 0$$

by definition of  $\beta_0$  of  $\eta_0$ ; recall the 0 indices denote true values. Let  $U = (Y - \ell_0(X)) - (D - m_0(X))\beta_0$ . Then, for any  $\eta = (m, \ell)$  that are square integrable, the Gateaux derivative in the direction

$$\Delta = \eta - \eta_0 = (m - m_0, \ell - \ell_0)$$

is given by

$$\begin{aligned} \partial_\eta \mathbb{E}\psi(W; \theta_0, \eta_0)[\Delta] &= -\mathbb{E}[U(m(X) - m_0(X))] \\ &\quad - \mathbb{E}\left[\left((m(X) - m_0(X))\beta_0 + (\ell(X) - \ell_0(X))\right)(D - m_0(X))\right] \\ &= 0, \end{aligned}$$

by the law of iterated expectations since  $\mathbb{E}[D - m_0(X)|X] = 0$  and  $\mathbb{E}[U|D, X] = 0$ .

**The Score for IRM.** Consider the score for the ATE in the IRM given in (9.4.4). We have that

$$\mathbb{E}\psi(W; \theta_0, \eta_0) = 0$$

by definition of  $\theta_0$  and  $\eta_0$ . Also, for any  $\eta = (g, m)$  that are square integrable with  $1/m + 1/(1-m)$  uniformly bounded, the Gateaux derivative in the direction

$$\Delta = \eta - \eta_0 = (g - g_0, m - m_0)$$

is given by

$$\begin{aligned}
 & \partial_\eta E\psi(W; \theta_0, \eta_0)[\Delta] \\
 &= E\left[g(1, X) - g_0(1, X)\right] \\
 &\quad - E\left[g(0, X) - g_0(0, X)\right] \\
 &\quad - E\left[\frac{D(g(1, X) - g_0(1, X))}{m_0(X)}\right] \\
 &\quad + E\left[\frac{(1 - D)(g(0, X) - g_0(0, X))}{1 - m_0(X)}\right] \\
 &\quad - E\left[\frac{D(Y - g_0(1, X))(m(X) - m_0(X))}{m_0^2(X)}\right] \\
 &\quad - E\left[\frac{(1 - D)(Y - g_0(0, X))(m(X) - m_0(X))}{(1 - m_0(X))^2}\right],
 \end{aligned}$$

which is 0 by the law of iterated expectations since  $E[D | X] = m_0(X)$ ,  $E[1 - D | X] = 1 - m_0(X)$ ,  $E[D(Y - g_0(1, X)) | X] = 0$ , and  $E[(1 - D)(Y - g_0(0, X)) | X] = 0$ .

# Bibliography

- [1] Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. ‘Double/debiased machine learning for treatment and structural parameters’. In: *The Econometrics Journal* 21.1 (2018), pp. C1–C68 (cited on pages 195, 205, 216, 218, 219).
- [2] James M. Poterba, Steven F. Venti, and David A. Wise. ‘401(k) Plans and Tax-Deferred savings’. In: *Studies in the Economics of Aging*. Ed. by D. A. Wise. Chicago, IL: University of Chicago Press, 1994, pp. 105–142 (cited on pages 206, 207, 209, 210).
- [3] James M. Poterba, Steven F. Venti, and David A. Wise. ‘Do 401(k) Contributions Crowd Out Other Personal Saving?’ In: *Journal of Public Economics* 58.1 (1995), pp. 1–32 (cited on pages 206, 207, 209, 210).
- [4] Jerzy Neyman. ‘Optimal asymptotic tests of composite hypotheses’. In: *Probability and statsitics* (1959), pp. 213–234 (cited on pages 211, 219).
- [5] Lester Mackey, Vasilis Syrgkanis, and Ilias Zadik. ‘Orthogonal machine learning: Power and limitations’. In: *International Conference on Machine Learning*. PMLR. 2018, pp. 3375–3383 (cited on page 218).
- [6] Peter J. Bickel, Chris A.J. Klaassen, Ya'acov Ritov, and Jon A. Wellner. *Efficient and adaptive estimation for semiparametric models*. Johns Hopkins University Press Baltimore, 1993 (cited on page 219).
- [7] Whitney K. Newey. ‘The asymptotic variance of semi-parametric estimators’. In: *Econometrica: Journal of the Econometric Society* 62.6 (1994), pp. 1349–1382 (cited on page 219).
- [8] Peter M. Robinson. ‘Root- $N$ -consistent semiparametric regression’. In: *Econometrica* 56.4 (1988), pp. 931–954. doi: [10.2307/1912705](https://doi.org/10.2307/1912705) (cited on page 219).
- [9] James M. Robins and Andrea Rotnitzky. ‘Semiparametric efficiency in multivariate regression models with missing data’. In: *J. Amer. Statist. Assoc.* 90.429 (1995), pp. 122–129 (cited on page 219).

- [10] Victor Chernozhukov, Christian Hansen, and Martin Spindler. 'Valid post-selection and post-regularization inference: An elementary, general approach'. In: *Annu. Rev. Econ.* 7.1 (2015), pp. 649–688 (cited on page 219).
- [11] Mark J. van der Laan and Sherri Rose. *Targeted learning: causal inference for observational and experimental data*. Springer Science & Business Media, 2011 (cited on page 219).

Here we discuss feature engineering as an approach to transform complex objects such as text and images into a collection of relatively low-dimensional numerical features (embeddings) that can be used for standard predictive or causal applications, for example as regressors in a prediction problem. We consider principle components, variational autoencoders and neural networks as general approaches to generate embeddings. We then consider text embeddings in detail, introducing two popular neural network-based Natural Language Processing (NLP) algorithms: ELMO and BERT. We finally consider image embeddings, applying a hedonic price model to apparel data using a neural network algorithm (ResNet50) to generate embeddings.

10.1 Introduction . . . . .	227
10.2 From Principal Components to Variational Autoencoders . . . . .	228
10.3 From Auto-Encoders to General Embeddings . . . . .	231
10.4 Text Embeddings . . . . .	232
10.5 Image Embeddings . . . . .	240
10.6 Constructing Hedonic Prices Using Apparel Data . . . . .	242

## 10.1 Introduction

Thus far, we have imposed a significant restriction on the kinds of data on which we can perform inference. While empiricists often consider simple datasets that include variables that have a numeric representation (binary, factor and continuous variables), researchers are increasingly presented with complex forms of data, such as images and text, that encode a vast amount of information. In this section, we generalize our approach to allow for consideration of these types of data.

As a motivating example, we consider the problem of predicting prices of products using the types of characteristics that one might find on a webpage, namely the text in the product description and the product's image. The resulting predicted prices are called hedonic prices, and predictive modeling of this form is motivated by the hedonic price models of economics.

In order to predict prices, we have to convert text and images into relatively low-dimensional numerical features, called "embeddings". The minimal requirement on embeddings is that similar products should have similar embeddings. This requirement guarantees that price predictions for similar products are also similar. The maximal requirement on embeddings is that they should parsimoniously approximate maximal information from text and images that is relevant for price predictions.

The main methods for generating successful embeddings include the following, in order of increasing generality:

- ▶ classical principal component analysis,
- ▶ variational auto-encoders, and
- ▶ neural networks solving auxiliary prediction tasks.

The auxiliary tasks in the final method may include solving image processing problems, such as object classification and image compression, or natural language processing problems, such as summarization and machine translation.

These auxiliary tasks are not the same as the "main" task. In our price prediction example, the main task is predicting product prices. Before turning to the primary price prediction task, we consider ResNet-50, which is a Residual Network of depth 50, which is designed to perform well on various tasks of object type classification. Consequently, application of ResNet-50 produces embeddings that are useful inputs for solving this auxiliary object classification task. However, because product type is an important determinant of price, the embeddings produced by

ResNet-50 that help classify products can also serve as useful inputs to the main task – price prediction.

Analogously, a neural network such as BERT is trained on auxiliary tasks aimed to make it learn word similarity and contextual meaning of words. Consequently, BERT can produce embeddings that provide a useful numerical summary of a product's text description. Because the product description is an important determinant of the price, these embeddings can also serve as useful inputs to the price prediction task.

Embeddings are useful in a variety of predictive and causal inference problems. For example, we can imagine using

- ▶ embeddings of product images and descriptions for modeling variety and demand for products;
- ▶ embeddings of text resumes for studying the wage offer structure;
- ▶ embeddings of countries' characteristics for studying the effect of institutions;
- ▶ and please list many of your own here (homework).

There is an emerging literature on the use of embeddings for causal inference; see this [Causal Text Repository](#).

## 10.2 From Principal Components to Variational Autoencoders

Principal components are probably the earliest classical example of embeddings.

Let  $(W_1, \dots, W_n)$  be a sample of  $n$  observations on high-dimensional centered random vector  $W_i$  in  $\mathbb{R}^d$ , and let  $\Sigma_n = \mathbb{E}_n[WW'] \in \mathbb{R}^{d \times d}$  denote the empirical covariance matrix. In order to reduce the dimension of  $W_i$ , we consider  $K \ll d$  mutually orthogonal rotations

$$X_{ik} := c'_k W_i, \quad k = 1, \dots, K,$$

of the original  $W_i$ 's where

$$c'_\ell c_k = 0 \text{ for } \ell \neq k \text{ and } c'_k c_k = 1 \text{ for each } k.$$

The condition on the coefficients ensures that for all  $\ell \neq k$ :

$$\mathbb{E}_n X_{ik} X_{i\ell} = 0.$$

These rotations are called principal components of  $W_i$ . In applications,  $W_i$  represent high-dimensional raw features (images,

for example), and

$$X_i^K = (X_{i1}, \dots X_{iK})'$$

represent a lower-dimensional encoding or embedding of  $W_i$ .

The principal components can be seen as the solution of the following least squares problem

$$\min_i \sum_j (W_{ij} - \hat{W}_{ij})^2$$

subject to

$$\hat{W}_{ij} := a'_j X_i^K, \quad X_{ik} = c'_k W_i, \quad k = 1, \dots, K.$$

Here we are trying to reconstruct the original data points  $W_{ij}$  using a linear combination of the principal components.

**Remark 10.2.1** The analytical solution to the principal components problem is as follows: The optimal  $C_K = [c_1, \dots, c_K]$  are the eigenvectors of  $\Sigma_n$  corresponding to the  $K$  largest eigenvalues  $\lambda_1, \dots, \lambda_K$  of  $\Sigma_n$ . That is,  $\Sigma_n c_k = \lambda_k c_k$  for each  $k$ . Furthermore, the optimal  $a_j$  is the  $j$ -th column of  $C'_K$ .

Once we produce the encodings/embeddings, we can look at how similar the raw inputs  $W_k$  and  $W_l$  are via the cosine similarity of the embeddings:

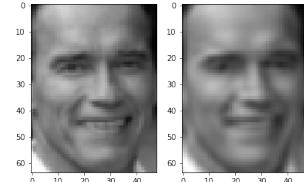
$$\text{sim}(W_k, W_l) = X'_k X_l / (\|X_k\| \|X_l\|).$$

In the context of product embeddings, this approach can be used, for example, to find products that are similar to a given product.

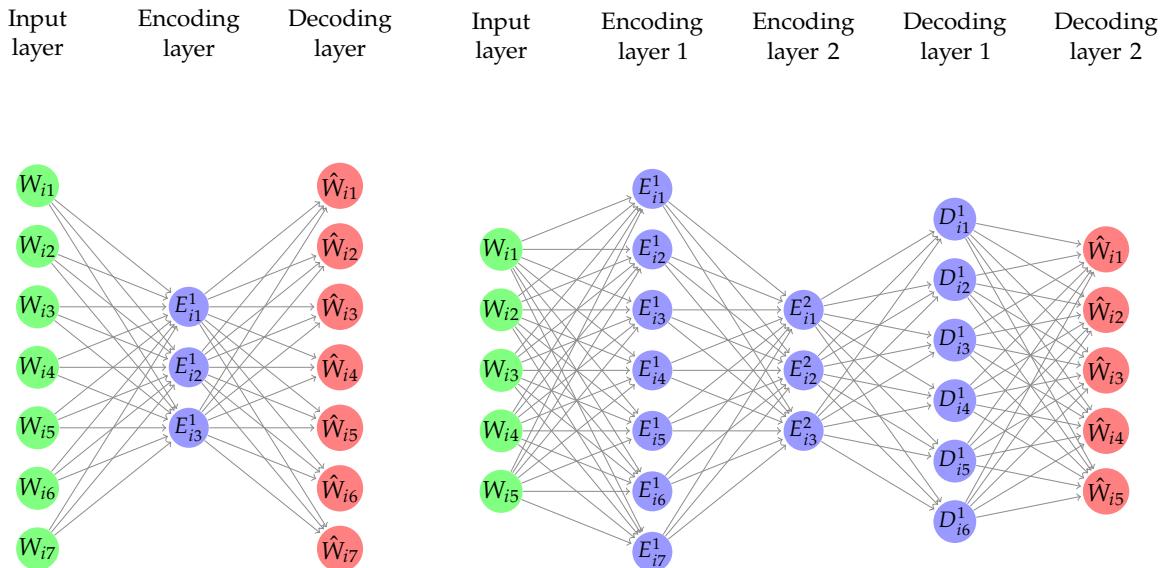
This predictive exercise underlying principal components can be seen as a linear neural network:

$$W_i \xrightarrow[d \times 1]{} C'_K W_i =: E \xrightarrow[k \times 1]{} A'E =: \hat{W}_i,$$

for  $A = [a_1, \dots, a_d]$ . The first step is said to be "encoding" the information in the input, and the second step is said to be "decoding" in the sense of returning the encoded information to the original space. Therefore, principal components are embeddings generated by a linear "encoder-decoder" network (an *autoencoder*, for short).



**Figure 10.1:** Featurizing the most talented man: The original 3072-dimensional image  $W$  and image  $\hat{W}$  produced from 256-dimensional embedding. (As a by-product, we've just made an important causal discovery that, surprisingly, doing embedding causes one to be younger).



**Figure 10.2:** The left panel shows a linear single layer autoencoder, such as linear principal components. The right panel shows a three layer nonlinear autoencoder; the middle layers can be used as embeddings.

This framing suggests that we can immediately generalize this approach to nonlinearly generated encoders and decoders that have multiple layers:

$$W_i \xrightarrow{g_1} E_i^1 \dots \xrightarrow{g_k} E_i^k \xrightarrow{g_{k+1}} D_i^{k+1} \dots \xrightarrow{g_m} D_i^m =: \hat{W}_i,$$

where maps  $g_\ell$ 's are neuron-generating maps. The middle layer or layers of low dimension, represented by the  $E_i^k$ , are taken to be encoders. The layers of neurons are mnemonically labelled as either "E" or "D", depending on whether they are doing "encoding" or "decoding," though note that there is no strict formal distinction between these types of layers.

Variational autoencoders are a way of discovering latent, low-dimensional structures in a dataset. In particular, a random data vector  $W \in \mathbb{R}^d$  can be said to have low-dimensional structure if we can find some "well-behaved" functions  $e : \mathbb{R}^d \rightarrow \mathbb{R}^k$  and  $d : \mathbb{R}^k \rightarrow \mathbb{R}^d$ , with  $k \ll d$ , such that

$$(d(e(W))) \approx W.$$

In other words,  $X = e(W)$  is a parsimonious,  $k$ -dimensional representation of  $W$  that contains all of the information necessary to approximately reconstruct the full vector  $W$ . Traditionally, the map  $e(\cdot)$  is called an encoder, and the map  $d(\cdot)$  is called a decoder function. Given this, a general formulation of autoencoders is to minimize the average

reconstruction loss,

$$\mathbb{E}_n[\text{loss}(W, \mathbf{d}(\mathbf{e}(W)))],$$

over "well-behaved" functions  $\mathbf{d} \in \mathcal{D}$  and  $\mathbf{e} \in \mathcal{E}$ . These classes are often linear, as in principal components, or generated via neural networks.

The qualification of "well-behaved" is important since it is always possible to write down some (completely wild) one-to-one function  $\mathbf{e} : \mathbb{R}^d \rightarrow \mathbb{R}^1$  such that  $\mathbf{e}^{-1}\mathbf{e}(W) = W$ .<sup>1</sup>

1: Google "Borel Isomorphism."

### 10.3 From Auto-Encoders to General Embeddings

The notion of loss can also be generalized to other loss functions, where the target outcome  $A$  may be not  $W$ , but something else. We can search for embeddings that minimize average prediction loss,

$$\mathbb{E}_n[\text{loss}(A, \mathbf{f}(\mathbf{e}(W)))],$$

where the role of  $\mathbf{f}(\cdot)$  is no longer to just to decode but to predict  $A$  (rather than  $W$ ).

For example, in feature engineering from images,  $A$  could be a product type or subtype, and  $W$  could be the image. In feature engineering from text,  $A$  could be a masked word in a sentence and  $W$  the sentence containing this word. These alternative approaches could be more useful in relation to the final learning task. For example, to build good hedonic price models, we may be more interested in image or text embeddings that best help to accurately describe the type or subtype of a product (rather than reconstruct the image or text itself).

This approach is generally implemented via neural networks as follows

$$W_i \xrightarrow{g_1} E_i^1 \dots \xrightarrow{g_k} E_i^k \xrightarrow{g_{k+1}} F_i^{k+1} \dots \xrightarrow{g_m} F_i^m =: \hat{A}_i,$$

where  $g_\ell$ 's are neuron-generating maps. The middle layers  $E_i^k$  are taken to be embedding layers, and  $F_i^k$ 's are predictive layers, aimed to create good predictions of auxiliary targets.

## 10.4 Text Embeddings

### First generation: Word2Vec Embeddings

We first review some basic ideas underlying the Word2Vec algorithm (Mikolov et al., 2013). The  $j$ -th word in a product description can be represented by a binary encoding

$$e_j = (0, \dots, 0, 1, 0, \dots, 0)',$$

with 1 in the  $j$ -th position. This encoding has a very high dimension  $d$ , corresponding to all the potential words in the corpus of documents under consideration, limiting its usefulness. Furthermore, this representation does not capture word similarity – e.g., cosine similarity between two different words  $j$  and  $k$  is always zero since  $e_j' e_k = 0$ .

Instead we aim to represent words by vectors of much lower dimension,  $r$ , that are able to capture word similarity. We denote the representation of the  $j$ -th word by  $u_j$ , so the dictionary is an  $r \times d$  matrix

$$\omega = \{u_1, \dots, u_d\},$$

where  $r$  is the reduced dimensionality of the dictionary. This dictionary is a linear rotation of the original dictionary  $E = \{e_1, \dots, e_d\}$ , where

$$\omega = \omega E.$$

Therefore, the problem of finding the rotation  $\omega$  is analogous to the problem of finding principal components, except that our goal is now to find representations  $\omega$  that are able to capture word similarity. Once we are done, each word  $t_j$  in a human-readable dictionary can be represented by a new “word”  $u_j$ . The goal of Word2Vec is to find an effective representation with the dimension  $r$  of the embedding being much smaller than  $d$ . We achieve this goal by treating  $\omega$  as parameters and estimating them so that the model performs well in some basic natural language processing tasks. These tasks are typically not related to downstream tasks, such as predicting hedonic prices or performing causal inference using text as control features, but are related to language prediction tasks.

Figure 10.3 shows components of dense embedding of several words produced by a trained Word2Vec map. The numbers presented in the table are not particularly interpretable in isolation. Each column represents a “trait” and the cell entry represents the loading of the word in the row in that trait. The numbers are more useful in comparison with each other across

### Example of Wor2Vec features

womens	0.387542	0.03051	-0.19703	0.179724	-0.222901	-0.606905	0.306091	-0.597467
mens	0.758868	0.372418	0.370116	0.706623	-0.124954	0.5088	0.106177	0.208935
clothing	0.149283	0.5161	-0.027684	0.218484	-0.851416	-0.409885	0.386088	0.170605
shoes	1.323812	-0.358704	-0.007683	-0.552144	0.011261	0.365239	0.228273	-0.565655
women	0.601477	-0.045845	-0.099481	0.010576	-0.096852	-0.605281	0.25606	-0.550759
girls	0.417473	-0.005265	-0.40939	-0.531189	-1.31938	-0.034746	-0.940507	-0.361215
men	0.778298	0.406613	0.426292	0.534272	-0.056103	0.51756	0.107846	0.245275
boys	0.896637	-0.016821	-0.001602	-0.181901	-1.313441	0.449006	-0.828408	0.52121
accessories	0.8625	-0.378385	-1.247708	1.541265	0.323952	0.282909	-0.491176	0.081314
socks	0.27636	0.354296	0.185734	0.301311	-0.643142	-0.021945	0.320751	0.240676
luggage	0.796763	1.749548	-2.30671	-0.559585	0.03054	0.921458	0.417333	0.313436
dress	0.282053	0.233192	0.043318	0.174759	-0.50114	-0.381047	0.297995	-0.026033
baby	0.346065	-0.550016	-1.136202	-0.043899	-0.004979	0.689747	-1.091575	0.009901
jewelry	-0.315784	0.347808	-0.308736	0.878713	-0.766016	1.124318	-0.079883	-2.039485
black	0.427496	0.030204	-0.019082	0.224096	-0.162242	-0.325359	0.170407	-0.172714
boots	1.009074	-0.30359	0.03197	-0.334004	-0.095679	0.111328	0.11769	-0.51878
shirts	0.444152	0.452918	0.393656	0.517929	-0.531462	0.099621	0.146202	0.204338
shirt	0.328998	0.421561	0.226565	0.455649	-0.700352	0.067224	0.106364	0.233862
underwear	0.230821	0.490978	0.226338	0.202376	-0.774363	0.004693	0.228712	0.310215

**Figure 10.3:** Examples of words converted to numerical features via Word2Vec. Compare embeddings for words "shirt" and "shirts" and for "luggage" and "dress".

different rows which allows us to understand word similarity. For example, we can see that the very similar words "shirt" and "shirts" have very similar embeddings while the embeddings for the seemingly relatively different words "luggage" and "dress" are quite dissimilar.

In our context, we can think of each word appearing in a datum (e.g. a product description) as a random variable  $T$  and denote its corresponding embedding representation by  $U$ .

One of the ways to train the word embeddings is to predict the middle word from the words that surround it in word sentences.

Given a subsentence  $s$  of  $K + 1$  words, we have a central word  $T_{c,s}$  whose identity we would like to predict. As predictors, we have the context words  $\{T_{o,s}\}$  that surround central word  $T_{c,s}$ . One approach for forming the prediction starts by collapsing the embeddings for context words by a sum,<sup>2</sup>

$$\bar{U}_o = \frac{1}{K} \sum_o U_{o,s},$$

where  $U_{o,s}$  is the element of  $\omega$  corresponding to the word  $T_{o,s}$ . This step imposes drastically simplifying assumption that the context words are exchangeable – i.e. the position of each word is not important.

The probability of the middle word  $T_{c,s}$  being equal to  $t$  is modeled via the multinomial logit function:

$$p_s(t; \pi, \omega) := P(T_{c,s} = t \mid \{T_{o,s}\}; \omega) = \frac{\exp(\pi'_t \bar{U}_s(\omega))}{\sum_{\bar{t}} \exp(\pi'_{\bar{t}} \bar{U}_s(\omega))},$$

2: Why not? We can try it and see if it works.

where  $\pi = (\pi_1, \dots, \pi_d)$  is an  $m \times d$  matrix of parameter vectors defining the choice probabilities. The model constrains the choice probabilities  $\pi$  to be  $\omega$ , and estimates  $\omega$  using the maximum quasi-likelihood method:

$$\max_{\omega=\pi} \sum_{s \in \mathcal{S}} \log p_s(T_{i,s}; \pi, \omega),$$

where we sum the log-probabilities over many examples  $\mathcal{S}$  of subsentences  $s$ . Once we are done training, we can generate the embedding for the title or description of product  $i$ , containing the embedded words  $\{U_{j,i}\}_{j=1}^J$  by simply averaging them:

$$W_i = \frac{1}{J} \sum_{j=1}^J U_{j,i}. \quad (10.4.1)$$

**Remark 10.4.1** In summary, the Word2Vec algorithm transforms text into a vector of numbers that can be used to compactly represent words. The algorithm trains a neural network in a supervised manner such that contextual information is used to predict another part of the text.

For example, let's say that the title description of the item is: "Hiigoo Fashion Women's Multi-pocket Cotton Canvas Handbags Shoulder Bags Totes Purses". The model will be trained using many  $n$ -word subsentence examples, such that the center word is predicted from the rest. If we just use  $n = 3$  subsentence examples, then we train the model using the following examples: (Hiigoo,Women's) → Fashion, (Fashion,Multi-pocket) → Women's, (Women's,Cotton) → Multi-pocket, and so on.

How do we judge whether the text embedding is successful or not? In the hedonic price context, we can check whether Word2Vec features improve the quality of prediction of the price by the hedonic model. We can also check if similar words  $T_k$  and  $T_l$  have similar embeddings. We can measure the similarity through cosine similarity:

$$\text{sim}(T_k, T_l) = U'_k U_l / (\|U_k\| \|U_l\|) \in [-1, 1].$$

The more similar the words are, according to our human notion of similarity, the higher the value our formal measure of similarity should take. For example, the following are the two words that are most similar to "tie" under the similarity measure: "necktie" and "bowtie". The dense embedding also induces an

interesting vector space on the set of words, which seems to encode analogues well. For example, the word "briefcase" is very cosine-similar to the artificial latent word

$$\text{Word2Vec}(\text{men's})$$

$$+ \text{Word2Vec}(\text{handbag}) - \text{Word2Vec}(\text{women's}).$$

This similarity between a real word and our constructed latent word gives some justification for the "averaging" of embeddings to summarize whole sentences or descriptions.

Word2vec embeddings were among the first generation of early successful embedding algorithms. These algorithms have been improved by the next generation of NLP algorithms, such as ELMO and BERT, which are discussed next.

## Second Generation: ELMO

The Embeddings from Language Models (ELMO) algorithm [1] uses the idea of the Shannon game where we aim to guess a word in a sentence,  $m$ , consisting of  $n$  total words. Specifically, we consider the problem of predicting word  $k + 1$  using the preceding  $k$  words via

$$p_{k,m}^f(t) = P[T_{k+1,m} = t | T_{1,m}, \dots, T_{k,m}; \theta]$$

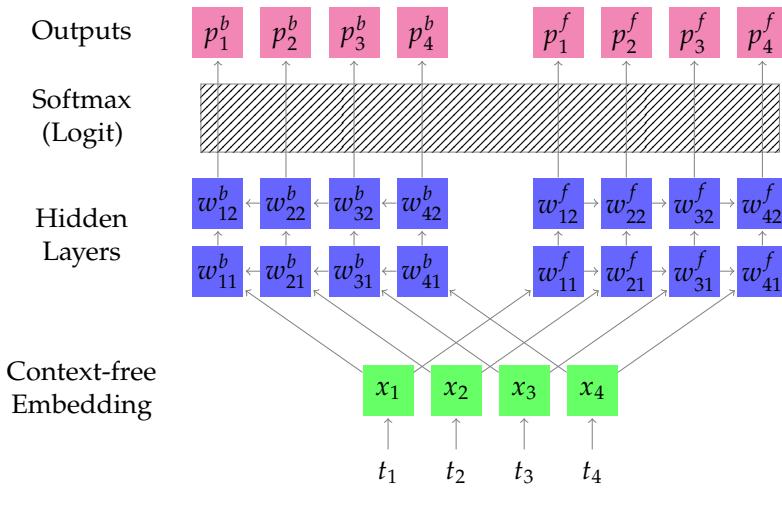
and similarly consider the reverse prediction via

$$p_{k,m}^b(t) = P[T_{k-1,m} = t | T_{k,m}, \dots, T_{n,m}; \theta],$$

where  $\theta$  is a parameter vector. Recursive neural networks with a single or multiple hidden layers are used to model these probabilities, where here recursive simply means that we use neurons from the previous prediction to make the current prediction. Parameters are estimated using quasi-maximum log-likelihood methods, where the forward and backward log quasi-likelihoods are added together.

To give a simple example, suppose we wanted to grasp the positional context better in the previous example. Rather than start by collapsing the embeddings for context words surrounding a target central word via a sum, we could instead keep track of word order and assign individual parameters to each context. For example, we could model the forward predicted probability

An early example of a low-dimensional RNN is the GARCH model of Bollerslev, which is used to model volatility of financial assets recursively.



**Figure 10.4:** ELMO Architecture. This is the ELMO network for a string of 4 words, with  $L = 2$  hidden layers. Here, the softmax layer (multinomial logit) is a single function mapping each input in  $\mathbb{R}^d$  to a probability distribution over the dictionary  $\Sigma$ .

of word  $k$  in sentence  $m$  as

$$P(T_{k,m} = t \mid \{T_{j,m}\}_{j=1}^{k-1}) = \frac{e^{\sum_{j=1}^{k-1} \pi'_{t,k} U_{k,m}(\omega)}}{\sum_{\tilde{t}} e^{\sum_{j=1}^{k-1} \pi'_{\tilde{t},k} U_{k,m}(\omega)}},$$

and similarly model the reverse prediction problem. ELMO uses a more sophisticated (and more parsimonious) nonlinear recursive nonlinear regression (specifically a recurrent neural network) model to build these probabilities. We illustrate a simple ELMO structure in Figure 10.4.

### The basic structure of ELMO.

Given a sentence  $m$  of  $n$  words,

1. Words are mapped to context-free embeddings in  $\mathbb{R}^d$
2. A network is trained to predict each word  $T_{k,m}$  of a string given (a) words  $(T_{1,m}, \dots, T_{k-1,m})$  or (b) words  $(T_{k+1,m}, \dots, T_{n,m})$ . The objective is to maximize the average over the sum of the log-likelihoods of the  $2n - 2$  words being predicted, where the average is taken over all sentences.
3. The embedding of word  $T_{k,m}$  is given by a weighted average of outputs of certain hidden neurons corresponding to this word's entire context. Importantly, the same final multinomial logistic ("softmax") layer is used for prediction objectives (2a) and (2b). Thus the inputs to this layer, which represent the forward and backward context, are constrained to lie in "the same space".

A softmax layer assigns probabilities to each class in a multi-class problem. It is a multi-class generalization of logistic regression that assumes mutually exclusive classes.

## Training

In Figure 10.4, the output probability distribution  $p_k^f$  is taken as a prediction of  $T_{k+1,m}$  using words  $(T_{1,m}, \dots, T_{k,m})$ . Similarly,  $p_k^b$  is taken as a prediction of  $T_{k-1,m}$  using words  $(T_{k,m}, \dots, T_{n,m})$ . The parameters of the network,  $\theta$ , are obtained by maximizing the quasi-log-likelihood:

$$\max_{\theta} \sum_{m \in \mathcal{M}} \left( \sum_{k=1}^{n-1} \log p_{k,m}^f(T_{k+1,m}; \theta) + \sum_{k=2}^n \log p_{k,m}^b(T_{k-1,m}; \theta) \right),$$

where  $\mathcal{M}$  is a collection of sentences. In our example,  $\mathcal{M}$  is the collection of titles and product descriptions taken from product web pages.

## Producing embeddings

To produce embeddings from the trained network, each word  $t_k$  in a sentence  $m = (t_1, \dots, t_n)$  is mapped to a weighted average of the outputs of the hidden neurons indexed by  $k$ :

$$t_k \mapsto w_k := \sum_{i=1}^L (\gamma_i w_{ki}^f + \bar{\gamma}_i w_{ki}^b).$$

The embedding for the sentence (or an entire product description in our example) is produced by summing the embeddings for each individual word. The weights  $\gamma$  and  $\bar{\gamma}$  can be tuned by the neural network performing the final task. In principle, however, the whole network could be plugged in to the network performing the final task and allowed to update.

## Second generation: BERT

Bidirectional Encoder Representations from Transformers (BERT) is another contextualized word embedding learned from deep language model [2]. It is a successor of ELMO and achieved state of the art results on multiple NLP tasks. Instead of using a Recurrent Neural Network as in ELMO, BERT uses a Transformer structure with an attention mechanism [3] that considers the whole sentence or context.

Unlike the language model in ELMO which predicts the next word from previous words, the BERT model is trained on two self-supervised tasks simultaneously:

- ▶ Mask Language Model: Randomly mask a certain percentage of the words in a sentence and predict the masked words.
- ▶ Next Sentence Prediction: Given a pair of sentences, predict whether one sentence precedes another.

### The basic structure of Bert.

1. Each word in the input sentence is broken into subwords and tokenized using a context-free embedding called WordPiece. A special token [cls] is added to the beginning of the sequence.  $x\%$  of the tokens representing individual words are replaced by [mask].
2. For each token, its input representation consists of i) its token embedding from (1), ii) its position embedding indicating the position of the token in the sentence, and iii) its segment embedding indicating whether it belongs to sentence A or B.
3. The input representation of tokens in the sequence is fed into the main model architecture: L layers of Transformer-Encoder blocks. Each block consists of a multi-head attention layer, followed by a feed forward layer.
4. The output representation of the mask token [mask] is used to predict the masked word via a softmax layer, and the output representation of the special [cls] token is used for Next Sentence Prediction. The loss function is a combination of the two losses.

We next focus in detail on the main structure step in (3), especially the “multi-head attention” layer.

### Computing the Attention

We begin with  $n$  word context-free embeddings  $(x_1, x_2, \dots, x_n)$ , with each  $x_k \in \mathbb{R}^d$ . Let  $X$  denote the matrix whose  $k$ th row is  $x_k$ . The Multi-Head Attention mapping is applied on  $X$  directly:

$$X \longmapsto \text{MultiHead}(X, X, X),$$

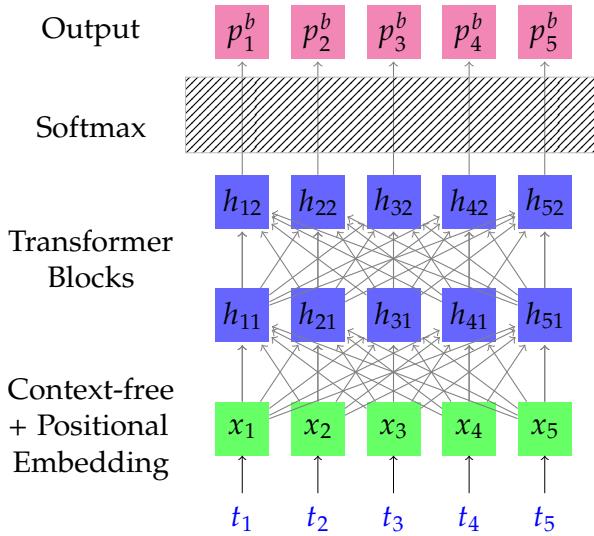


Figure 10.5: BERT Architecture

where

$$\text{MultiHead}(Q, K, V) = \text{Concatenate}(\text{Head}_1, \dots, \text{Head}_h)\omega^O,$$

$$\text{Head}_i = \text{Attention}(Q\omega_i^Q, K\omega_i^K, V\omega_i^V),$$

$$\text{Attention}(\tilde{Q}, \tilde{K}, \tilde{V}) = \text{softmax}\left(\tilde{Q}\tilde{K}^T/\sqrt{d_k}\right)\tilde{V},$$

with  $\omega^O$  and  $(\omega_i^Q, \omega_i^K, \omega_i^V)$  matrix parameters that are trained to maximize the model performance. In other words, each word embedding is replaced by the weighted average of all other words, and the weights are learned from the scaled dot-product of different projections of the word embeddings themselves.

### Generating product embeddings

Depending on specific tasks and resources, Devlin et al. [2] suggested to construct BERT embeddings in various ways:

- ▶ Use the last layer, second-to-last layer, or concatenate the last 4 layers of the encoder outputs from the pre-trained BERT model.
- ▶ Fine tune the whole BERT model using the downstream task.
- ▶ Train the BERT language model from scratch on new data.

In the hedonic price example below, the feature-based approach was chosen, where the second-to-last layer from a pre-trained BERT model was extracted as embeddings. Each product's text embedding is the average of the embeddings of each word/token from the input text field.

## Comparing ELMO and BERT

ELMO and BERT are both recent breakthroughs in NLP. The former marked the first contextual word embedding trained from a deep language model, and the latter was the first contextual word embedding using Transformer architecture. Note that since the BERT paper was published second and could respond directly to the ELMO paper (but not vice-versa), the comparisons are likely to be biased towards the latter.

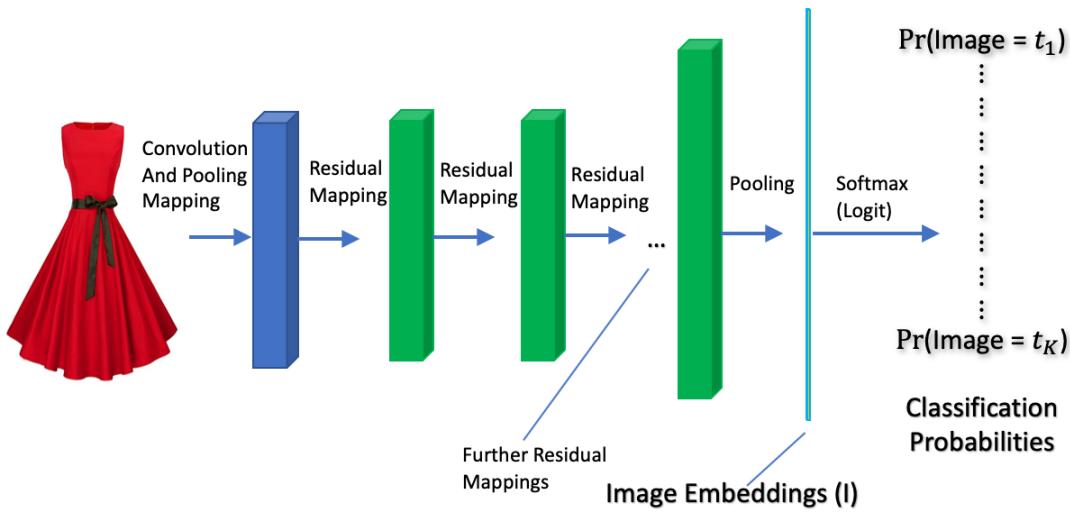
There are several key differences between the two approaches:

- ▶ The biggest difference lies in the choice of fundamental architectures: ELMO is based on a Recurrent Neural Network (RNN), while BERT is based on the transformer architecture. RNNs are known for not being able to capture long-term dependencies, whereas the transformer architecture is more efficient at capturing long-range dependencies in the text. Furthermore, ELMO creates context by using the left-to-right and right-to-left language model representations, while BERT models the entire context simultaneously.
- ▶ They use different initial context-free embeddings.
- ▶ ELMO applies an initial convolutional layer to a *character* embedding, while BERT augments the WordPiece embedding at the *sub-word* level with positional data.
- ▶ The ELMO implementation only allows the averaging weights to be fine-tuned, whereas BERT proposes fine-tuning the whole network.

## 10.5 Image Embeddings

One of the most successful deep learning models for image classification was the ResNet50 model developed by He et al. [4]. At the time of the release, the paper achieved the best results in image classification, in particular for the ImageNet and COCO datasets.

The central idea of the paper is to exploit "partial linearity": traditional nonlinearly-generated neurons are combined (or added together) with the previous layer of neurons. More specifically, ResNet50 takes a standard feed-forward convolutional neural network and adds skip connections that bypass two (or one or several) convolutional layers at a time. Each skipping step generates a residual block in which the convolution layers predict a residual.



**Figure 10.6:** The ResNet50 operates on numerical 3-dimensional arrays representing images. It first does some pre-processing by applying convolutional and pooling filters, then it applies many L-residual block mappings, producing the arrays shown in green. The penultimate layer produces a high-dimensional vector  $I$ , the image embedding, which is then used to predict the image type.

Formally, each  $k$ -th residual block is a neural network mapping

$$\begin{aligned} v &\mapsto (v, \sigma_k^0(\omega_k^0 v)) \mapsto (v, \sigma_k^1 \circ \omega_k^1 \sigma_k^0(\omega_k v)) \\ &\mapsto v + \sigma_k^1 \circ \omega_k^1 \sigma_k^0(\omega_k^0 v), \end{aligned}$$

where  $\omega$ 's are matrix-valued parameters or "weights". This structure can be seen as a special case of general neural network architecture, designed so that it is easy to learn the identity sub-maps (entering the composition of the entire network). Putting together many blocks like these sequentially results in the overall architecture depicted in Figure 10.6.

The deep feed-forward convolutional networks developed in prior work suffered from major optimization problems – once the depth was sufficiently high, additional layers often resulted in much higher validation and training error. It was argued that this phenomenon was a result of "vanishing gradients," where in a network of  $n$  layers, computation by backpropagation using the chain rule involves multiplying  $n$  small numbers (if using traditional activation functions, recent popular activation functions such as RELU do not induce such a small derivative), causing the gradient to "vanish" for early layers and posing a computational challenge. The residual network architecture addresses this by using the residual block architecture: including the residual directly via skip connections reduces the minimizing impact of the activation function. The creation of this

architecture has allowed for high quality training even for very deep networks.

Just like with text embeddings, we are not interested in the final predictions of these networks but rather in the last hidden layer, which is taken to be the image embedding. In the example in Section 10.6, we rely on a publicly trained ResNet50 model to generate the image embeddings.<sup>3</sup>

## 10.6 Constructing Hedonic Prices Using Apparel Data

Here we apply our new knowledge of embeddings to a prediction problem: the hedonic price model application of Bajari et al. [5]. An empirical hedonic model is a predictive model for price given a traded object's characteristics. Here, we predict the price of apparel bought and sold on Amazon.com using the product's image and description:

$$P_{it} = H_{it} + \epsilon_{it} = h_t(X_{it}) + \epsilon_{it}, \quad \mathbb{E}[\epsilon_{it} | X_{it}] = 0, \quad (10.6.1)$$

where  $P_{it}$  is the price of product  $i$  at time  $t$  (in months),  $X_{it}$  are the product features, and the price function  $x \mapsto h_t(x)$  can change from period to period, reflecting the fact that product attributes/features may be valued differently in different periods.

One of the main uses of hedonic prices is construction of cost of living indices. The use of hedonic prices allows us to "price" the product attributes as well as entire "baskets of attributes" that consumers buy. Then, given a reference "basket of attributes," we can look at the hedonic cost of a basket today compared to its cost in an earlier reference period to determine whether the cost increased or decreased. These types of calculations underlie the construction of the commonly used consumer price indices (measuring inflation rates), at least for categories such as apparel products.

In Bajari et al. [5], most of the product attributes  $X_{it}$  remain time-invariant, but they are allowed to change over time. The data from time period  $t$  is used to estimate the function  $h_t$  using modern nonlinear regression methods, such as deep neural network methods. The results are contrasted with classical linear regression methods as well as other modern regression methods, such as the random forest.

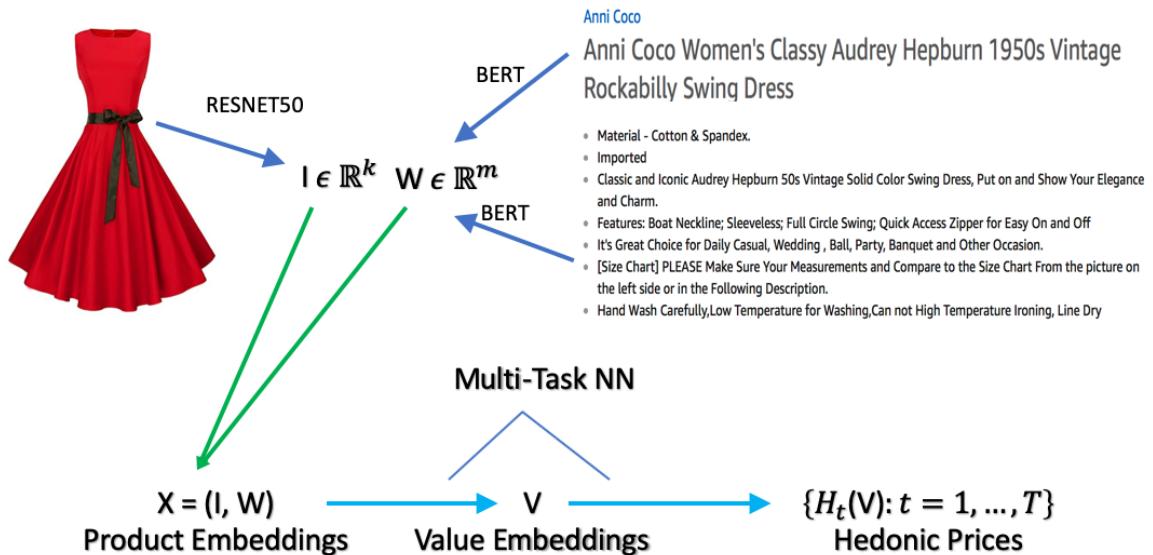
<sup>3</sup>: We also tested an image model fine-tuned on the Amazon.com Catalog with a similar structure to ResNet50 and observed some small performance improvement.

A key component of the approach taken in Bajari et al. [5] is the use of product features  $X_{it}$  generated as neural network embeddings of text and image information about the product. Specifically,  $X_{it}$  consists of text embedding features  $W_{it}$ , constructed by converting the title and product description available on a product's web page into numeric vectors, and image embedding features  $I_{it}$  constructed by converting the product image into numeric vectors:

$$X_{it} = (W'_{it}, I'_{it})'. \quad (10.6.2)$$

These embedding features are generated respectively by applying the BERT and ResNet50 mappings.

The model takes high-dimensional text and image features as inputs, converts them into a lower dimensional vector of value embeddings using deep learning methods, and outputs simultaneous predictions of price in all time periods.



**Figure 10.7:** The structure of the predictive model in Bajari et al. [5]. The input consists of images and unstructured text data. The first step of the process creates numerical embeddings  $I$  and  $W$  for images and text data via state of the art deep learning methods, such as ResNet50 and BERT. The second step of the process takes as its input  $X = (I, W)$  and creates predictions for hedonic prices  $H_t(X)$  using deep learning methods with a multi-task structure. The models of the first step are trained on tasks unrelated to predicting prices (e.g., image classification or word prediction), where embeddings are extracted as hidden layers of the neural networks. The models of the second step are trained by price prediction tasks. The multitask price prediction network creates an intermediate lower dimensional embedding  $V = V(X)$ , called a value embedding, and then predicts the final prices in all time periods  $\{H_t(V), t = 1, \dots, T\}$ . Some variations of the method include fine-tuning the embeddings produced by the first step to perform well for price prediction tasks (i.e. optimizing the embedding parameters so as to minimize price prediction loss).

The general structure of the model takes the form

$$\begin{aligned} Z_i &= \begin{bmatrix} \text{Text}_i \\ \text{Image}_i \end{bmatrix} \xrightarrow{e} X_i \\ &\xrightarrow{g^1} E_i^{(1)} \dots \xrightarrow{g^m} E_i^{(m)} =: V_i \xrightarrow{\theta'} \{H_{it}\}_{t=1}^T := \{\beta'_t V_i\}_{t=1}^T. \end{aligned}$$

Here  $Z_i$ , the original input which lies in a very high-dimensional space, is nonlinearly mapped into an embedding vector  $X_i$  which is of moderately high dimension (up to 5120 dimensions in this example).  $X_i$  is then further nonlinearly mapped into a lower dimension vector  $E_i^{(1)}$ . This process is repeated to produce the final hidden layer,  $V_i = E_i^{(m)}$ , which is then linearly mapped to the final output that consists of hedonic price  $H_{it}$  for product  $i$  in all time periods  $t = 1, \dots, T$ .

The last hidden layer  $V = E^{(m)}$  is called the *value embedding* in this context – the value embedding represents latent attributes to which dollar values are attached. The embeddings produced in this example are moderately high-dimensional (up to 512 dimensions) summaries of the product, derived from the most common attributes that directly determine the price of the predicted hedonic price of the product. Note that the embeddings  $V$  in this example do not depend on time and so may be thought of as representing intrinsic, potentially valuable attributes of the product. However, the predicted price does depend on time  $t$  via the coefficient  $\beta_t$ , reflecting the fact that the different intrinsic attributes are valued differently across time.

The network mapping above comprises a deep neural network with neurons  $E_{k,\ell}$  of the form

$$g_\ell : v \mapsto \{E_{k,\ell}(v)\}_{k=1}^{K_\ell} := \{\sigma_{k,\ell}(v' \alpha_{k,\ell})\}_{k=1}^{K_\ell}. \quad (10.6.3)$$

Here  $\sigma_{k,\ell}$  is the activation function that can vary with the layer  $\ell$  and can vary with  $k$ , from one neuron to another.

The model is trained by minimizing the loss function

$$\min_{\eta \in \mathcal{N}, \{\beta_t\}_{t=1}^T} \sum_t \sum_i (P_{it}^c - \beta'_t V_i(\eta))^2 Q_{it}, \quad (10.6.4)$$

where  $\eta$  denotes all of the parameters of the mapping

$$X_i \mapsto V_i(\eta)$$

and  $\mathcal{N}$  represents the parameter space. Here, we are using a weighted loss where we weight by the quantity of product  $i$  sold at time  $t$ ,  $Q_{it}$ .

Next we review how the initial embedding is generated. A multilingual BERT model is used to convert text information and the ResNet50 model is used to convert images into a subvector of  $E_i^{(1)}$ . These models are trained on auxiliary prediction tasks with auxiliary outputs  $A_{T_i}$  for text and  $A_{I_i}$  for images. Introducing these auxiliary tasks can be illustrated diagrammatically as

$$X_i = \begin{bmatrix} \text{Text}_i \\ \text{Image}_i \end{bmatrix} \xrightarrow{e} W_i = E_i^{(1)} \dots \mapsto E_i^{(m)} := V_i \mapsto \{\hat{P}_{it}^*\}_{t=1}^T,$$

(10.6.5)

The embeddings  $W_i$  and  $X_i$  forming  $E_i^{(1)}$  are obtained by mapping them into auxiliary outputs  $A_{T_j}$  and  $A_{I_j}$  that are scored on natural language processing tasks and image classification tasks respectively. This step uses data that are not related to prices, as described in detail in the previous sections. The parameters of the mapping generating  $E_i^{(1)}$  are considered as fixed in our analysis.

The price prediction network we employ in this example contains three hidden layers, with the last hidden layer containing 400 neurons. The network is trained on a large data set with more than 10 million observations. A large enough data set is crucial for training successful neural networks.

The accuracy of prediction as measured by the  $R^2$  on the test sample is about

90%.

In contrast,

- ▶ random forests using embeddings deliver  $R^2$  in the ballpark of 80%;
- ▶ the linear model using least squares applied to embeddings delivers  $R^2$  in the ballpark of 70%;
- ▶ the linear model, using simple catalogue features (without embeddings), delivers an  $R^2$  lower than 40%.

Thus, embeddings offer a means of making use of complex data for predictions and, at least for large data sets, neural nets can offer predictive improvements relative to competing machine learning approaches.

## Notebooks

- ▶ [Python Auto-Encoders Notebook](#) provides an introduction to variational auto-encoders, starting from classical principal components.
- ▶ [Python Toys and Prices Notebook](#) provides an introduction to text embeddings via BERT and provides an application to predicting demand for toys.

## Study Problems

1. Work through the Auto-Encoders notebook. Try to improve the performance of the neural auto-encoders. Report your findings (even if you don't manage to improve them! :-)).
2. Work through the BERT notebook. Try to experiment with the structure of the neural nets and demand estimation procedure. Report your findings.

# Bibliography

- [1] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. ‘Deep contextualized word representations’. In: *CoRR* abs/1802.05365 (2018) (cited on page 235).
- [2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. ‘BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding’. In: *CoRR* abs/1810.04805 (2018) (cited on pages 237, 239).
- [3] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. ‘Attention is All you Need’. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc., 2017 (cited on page 237).
- [4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. ‘Deep residual learning for image recognition’. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778 (cited on page 240).
- [5] Patrick L. Bajari, Zhihao Cen, Victor Chernozhukov, Manoj Manukonda, Jin Wang, Ramon Huerta, Junbo Li, Ling Leng, George Monokroussos, Suhas Vijaykumar, et al. *Hedonic prices and quality adjusted price indices powered by AI*. Tech. rep. cemmap working paper, 2021 (cited on pages 242, 243).

# **ADVANCED CORE MATERIAL**

# Advanced Core 1: Unobserved Confounders, Instrumental Variables, and Proxy Controls

# 11

"Without Philip Wright  
would there have been causal DAGs?  
Who can really say?"  
Kei Hirano.\*

In this module we discuss various models with unobserved confounders, where the adjustment strategies we have discussed no longer work. We start with sensitivity analysis of causal inference to the presence of unobserved confounders. Then we discuss identification of causal effects when instrumental variables or proxy controls are available.

11.1 The Impossibility of Causal Inference with an Unobserved Confounder . . . . .	250
11.2 Impact of Confounders on Causal Effect Identification and Sensitivity Analysis . . . . .	251
11.3 Partially Linear IV Models . . . . .	254
A Wage Equation with Unobserved Ability . . . . .	254
Aggregate Market Demand . . . . .	256
SEMs with Griliches-Chamberlain Proxy Controls . . . . .	257
11.4 Nonlinear IV Models . . . . .	259
The LATE Model . . . . .	259
The IV Quantile Model* . . . . .	261
11.5 Nonlinear Models with Proxy Controls* . . . . .	262
11.6 Study Problems . . . . .	264
11.7 Proofs . . . . .	266
Latent Confounder Bias Result: Theorem 11.2.1 . . . . .	266
Linear Proxy Model: Theorem 11.3.2 . . . . .	267

---

\* Sewall Wright, son, and Philip Wright, father, were responsible for some of the greatest ideas in causal inference. Sewall Wright invented causal path diagrams (linear DAGs), and Philip Wright wrote down DAGs for supply-demand equations, proposed IV methods for their identification, and even proposed weather conditions as instruments. Just one of these contributions would probably be enough to get a QJE publication in 1970s and later, but it was not good enough in 1926 or so. Philip Wright is a (causal) parent of Sewall Wright, so he is one of the causes of DAGs (hence the haiku).

## 11.1 The Impossibility of Causal Inference with an Unobserved Confounder

"All happy statisticians are happy in their own way; but all the unhappy ones are all alike — they all do causal inference with observational data". L. Tolstoy in Anna Karenina (Source: [Twitter](#))

Here we consider models with an unobserved confounding variable. The key result is that in the presence of unobserved confounding, causal effects are not identified. For example, consider the following two basic models shown in the margin figure, where we can think of  $Y$  as wages,  $D$  as education, and  $A$  as latent ability.

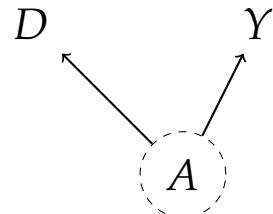
If  $A$  is not observed, the two models in Figures 11.1 and 11.2 are statistically indistinguishable from each other. In the first model  $D$  has a causal effect on  $Y$ , and in the second it does not. Even with strong restrictions, as in Gaussian linear SEMs, the observed correlation between  $D$  and  $Y$  can always be rationalized either as a causal effect of  $D$  on  $Y$  or the result of a common cause  $A$  (homework). This observation applies more generally. While we cannot precisely pin down causal effects in these cases, we can still learn about causal effects by performing sensitivity analysis if we are willing to assume a bound on the strength of unobserved confounders. We discuss a practical and intuitive approach to sensitivity analysis in Section 11.2.

We may also make progress in learning causal effects in the presence of unobserved confounders by considering the use of instrumental variables (IVs) – additional random vectors  $Z$  that create exogenous variation in  $D$ . This approach was introduced by Philip Wright in 1928 [1]. The use of instruments renders many linear ASEM identifiable, allowing us to perform inference on structural effects  $D \rightarrow Y$ . Some nonlinear ASEM also become identifiable, though identification still fails for completely unrestricted nonlinear models. We discuss the use of instruments in Sections 11.3-11.4.

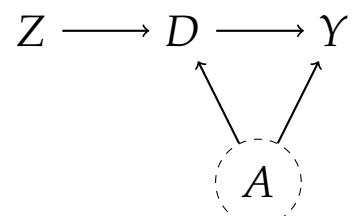
A related set of problems is when we observe multiple proxy measurements of the latent confounder  $A$ . For example, we may observe  $S$ , the SAT score, and  $Q$ , the ACT score, which may both be proxies for latent confounder,  $A$ , ability. Note that conditioning on  $Q$  and  $S$  does not block the backdoor path  $Y \leftarrow A \rightarrow D$ . Hence we cannot use the regression adjustment method for identification of  $D \rightarrow Y$ . However, this problem is

$$D \longrightarrow Y$$

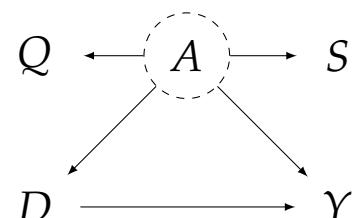
**Figure 11.1:**  $D$  causes  $Y$



**Figure 11.2:**  $D$  and  $Y$  are caused by a latent factor  $A$



**Figure 11.3:** A DAG with Latent Confounder  $A$  and Instrument  $Z$ .



**Figure 11.4:** A DAG with two proxies for latent confounders.

related to IVs, because we can effectively use one measurement in place of  $A$  and instrument it with another measurement to deal with the measurement error. This process can provide identification of the main effect  $D \rightarrow Y$ . In other words, we can use instrumental variable regression of  $Y$  on  $D$  and  $S$ , using  $D$  and  $Q$  as technical instrumental variables. This approach was introduced by Zvi Griliches in 1977 [2]. This model has also been extensively studied for nonlinear models as well, e.g., Miao et al. [3] and Deaner [4], especially in the recent literature. We discuss proxy approaches in Section 11.5.

## 11.2 Impact of Confounders on Causal Effect Identification and Sensitivity Analysis

**Example 11.2.1** (Partially Linear SEM) Consider the SEM

$$\begin{aligned} Y &:= \alpha D + \delta A + f_Y(X) + \epsilon_Y, \\ D &:= \gamma A + f_D(X) + \epsilon_D, \\ A &:= f_A(X) + \epsilon_A, \\ X &:= \epsilon_X, \end{aligned}$$

where, conditional on  $X$ ,  $\epsilon_Y, \epsilon_D, \epsilon_A$  are both centered and mutually uncorrelated. We further normalize

$$\text{E}\epsilon_A^2 = 1.$$

The key structural parameter is  $\alpha$ :

$$\alpha = \partial_d Y(d)$$

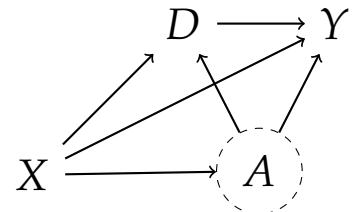
where

$$Y(d) := (Y : do(D = d)).$$

To give context to our example, we can interpret  $Y$  as earnings,  $D$  as education,  $A$  as ability, and  $X$  as a set of observed background variables. In this example, we can interpret  $\alpha$  as the returns to schooling.

We start by applying the partialling out operator to get rid of the  $X$ 's in all of the equations. Define the partialling out operation of any random vector  $V$  with respect to another random vector  $X$  as the residual that is left after subtracting the best predictor of  $V$  given  $X$ :

$$\tilde{V} = V - \text{E}[V | X].$$



**Figure 11.5:**  $X$  are observed confounders, and  $A$  are unobserved confounders.

If  $f$ 's are linear, we can replace  $E[V | X]$  by linear projection. After partialling out, we have a simplified system:

$$\begin{aligned}\tilde{Y} &:= \alpha\tilde{D} + \delta\tilde{A} + \epsilon_Y, \\ \tilde{D} &:= \gamma\tilde{A} + \epsilon_D, \\ \tilde{A} &:= \epsilon_A,\end{aligned}$$

where  $\epsilon_Y$ ,  $\epsilon_D$ , and  $\epsilon_A$  are uncorrelated.

Then the projection of  $\tilde{Y}$  on  $\tilde{D}$  recovers

$$\beta = E\tilde{Y}\tilde{D}/E\tilde{D}^2 = \alpha + \phi,$$

where

$$\phi = \delta\gamma/E(\gamma^2 + \epsilon_D^2),$$

is the omitted confounder bias.

Omitted confounder bias is also often referred to as omitted variables bias.

The formula follows from inserting expression for  $\tilde{D}$  and then simplifying the resulting expression using the assumptions on the  $\epsilon$ 's.

We can use this formula to bound  $\phi$  directly by making assumptions on the size of  $\delta$  and  $\gamma$ . An alternative approach can be based on the following characterization, based on partial  $R^2$ 's. This characterization essentially follows from Cinelli and Hazlett [5], with the slight difference that we have adapted the result to the partially linear model.<sup>1</sup>

**Theorem 11.2.1** (Omitted Confounder Bias in Terms of Partial  $R^2$ 's) *In the setting given in Example 11.2.1,*

$$\phi^2 = \frac{R_{\tilde{Y} \sim \tilde{A} | \tilde{D}}^2 R_{\tilde{D} \sim \tilde{A}}^2}{(1 - R_{\tilde{D} \sim \tilde{A}}^2)} \frac{E(\tilde{Y} - \beta\tilde{D})^2}{E(\tilde{D})^2},$$

where  $R_{V \sim W | X}^2$  denotes the population  $R^2$  in the linear regression of  $V$  on  $W$ , after partialling out linearly  $X$  from  $V$  and  $W$ .

1: [6] recently obtained a similar result for fully nonlinear models.

Therefore, if we place bounds on how much of variation in  $\tilde{Y}$  and in  $\tilde{D}$  the unobserved confounder  $\tilde{A}$  is able to explain, we can bound the omitted confounder bias by

$$\sqrt{\phi^2}.$$

**Example 11.2.2** We consider an empirical example based on data surrounding the Darfur war. Specifically, we are interested in the effect of having experienced direct war violence on attitudes towards peace. The observed controls explain 12-15% of the variance of  $Y$ , beyond what's explained by the "treatment" variable, and 1% of the variance of treatment  $D$ . Therefore, suppose we are willing to accept that

$$R^2_{\tilde{Y} \sim \tilde{A} | \tilde{D}} \leq .15, \quad R^2_{\tilde{D} \sim \tilde{A}} \leq .01;$$

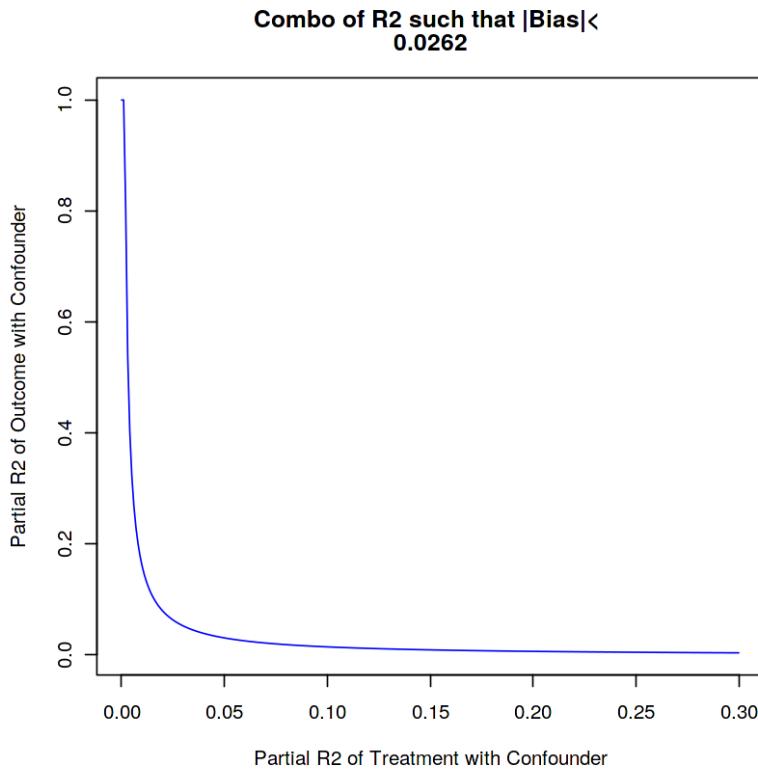
that is, we have a latent confounder that is no stronger than the observed controls for predicting  $Y$  and for predicting  $D$ .

Then, the upper/lower bound on  $\alpha$  is given by

$$\beta \pm \phi, \quad \phi^2 = \frac{.0015}{.99} \frac{E(\tilde{Y} - \beta \tilde{D})^2}{E(\tilde{D})^2}.$$

The estimated  $\beta$  is about .1. Plugging in estimates of  $E(\tilde{Y} - \beta \tilde{D})^2$  and  $E(\tilde{D})^2$  yields an estimated lower bound on  $\alpha$  of around .074. In Figure 11.6, we show the combination of all partial  $R^2$  such that the bias is less than .026. It shows that our conclusions about causal effects are not very sensitive to the presence of unknown confounders whose power is limited by the stated assumptions.

DML Sensitivity R Notebook carries out sensitivity analysis based on DML and the R package Sensemakr for the analysis of the Darfur wars data.



**Figure 11.6:** Sensitivity contour plots: The graph shows values of  $R_{\tilde{Y} \sim \tilde{D} | \tilde{A}}^2$  and  $R_{\tilde{D} \sim \tilde{A}}^2$  that give a given value of the bias  $|\hat{\phi}| = .026$ .

### 11.3 Partially Linear IV Models

When instrumental variables are available, it becomes possible to point identify causal effects in partially linear models and certain types of causal effects in nonlinear models. Here we begin with partially linear models.

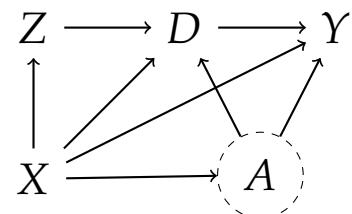
#### A Wage Equation with Unobserved Ability

**Example 11.3.1** (Returns to Education with Omitted Ability; Generalization of Griliches, 1977 [2]) Consider the ASEM

$$\begin{aligned} Y &:= \alpha D + \delta A + f_Y(X) + \epsilon_Y, \\ D &:= \beta Z + \gamma A + f_D(X) + \epsilon_D, \\ Z &:= f_Z(X) + \epsilon_Z, \\ A &:= f_A(X) + \epsilon_A, \\ X &:= \epsilon_X, \end{aligned}$$

where, conditional on  $X, \epsilon_Y, \epsilon_D, \epsilon_Z, \epsilon_A$  have mean zero and are mutually uncorrelated.

We can interpret  $Y$  as earnings,  $D$  as education,  $A$  as ability,  $Z$



**Figure 11.7:** An IV model with observed and unobserved confounders.

as an observed shifter of education, and  $X$  as a set of observed background variables. The key structural parameter is  $\alpha$ , the returns to schooling, i.e.

$$\alpha = \partial_d Y(d),$$

where

$$Y(d) = Y : do(D = d).$$

Examples of instruments for schooling,  $Z$ , that have appeared in the literature include

- ▶ distance to college (Card [7]),
- ▶ compulsory schooling laws (Angrist [8]),
- ▶ offer to participate/offer to treat in a training program (many studies), and
- ▶ subsidies to finance education (Griliches, Heckman).

We apply the partialling-out operator to get rid of the  $X$ 's in all of the equations. As before, we define the partialling out operation of any random vector  $V$  with respect to another random vector  $X$  as the residual that is left after subtracting the best predictor of  $V$  given  $X$ :

$$\tilde{V} = V - E[V | X].$$

If  $f$ 's are linear, we replace  $E[V | X]$  with linear projection. After partialling-out, we have a simplified system.

$$\begin{aligned}\tilde{Y} &:= \alpha\tilde{D} + \delta\tilde{A} + \epsilon_Y, \\ \tilde{D} &:= \beta\tilde{Z} + \gamma\tilde{A} + \epsilon_D, \\ \tilde{Z} &:= \epsilon_Z, \\ \tilde{A} &:= \epsilon_A,\end{aligned}$$

where  $\epsilon_Y, \epsilon_D, \epsilon_Z$ , and  $\epsilon_A$  are uncorrelated.

We immediately obtain the following result:

**Theorem 11.3.1** In Example 11.3.1, we can rewrite an econometric measurement model for identification of  $\alpha$ :

$$\tilde{Y} := \alpha\tilde{D} + U, \quad U \perp \tilde{Z},$$

where  $U = \delta\tilde{A} + \epsilon_Y$ . Alternatively, we can equivalently identify  $\alpha$  using the moment restriction

$$E(\tilde{Y} - \alpha\tilde{D})\tilde{Z} = 0.$$

The identification of  $\alpha$  follows from solving this equation,

$$\alpha = E\tilde{Y}\tilde{Z}/E\tilde{D}\tilde{Z},$$

provided the instruments are relevant:  $E\tilde{D}\tilde{Z} \neq 0$  or  $\beta \neq 0$ .

**Remark 11.3.1** (Neyman Orthogonality and DML) The target parameter  $\alpha$  is Neyman orthogonal with respect to nuisance parameters – the regression functions  $E[Y | X]$ ,  $E[D | X]$ , and  $E[Z | X]$ . Therefore we can use Debiased ML for learning and performing statistical inference on the parameter  $\alpha$ .

## Wright's Causal Path Derivation

Starting from the DAG given in Figure 11.7, we obtain Figure 11.8 after partialling.

Philip Wright (1928) [1] observed that the structural parameter  $\beta\alpha$ , the effect  $\tilde{Z} \rightarrow \tilde{Y}$ , is identified from the projection of  $\tilde{Y} \sim \tilde{Z}$ :

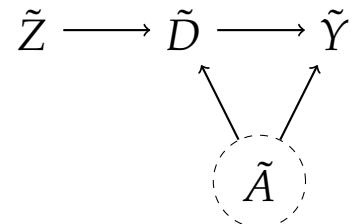
$$\beta\alpha = E\tilde{Y}\tilde{Z}/E\tilde{Z}^2.$$

The structural parameter  $\beta$ , the effect of  $Z \rightarrow D$ , is identified from the projection of  $\tilde{D} \sim \tilde{Z}$ :

$$\beta = E\tilde{D}\tilde{Z}/E\tilde{Z}^2.$$

$\alpha$ , the effect of  $D \rightarrow Y$ , is then identified by the ratio of the two provided  $\beta \neq 0$ :

$$\alpha = \frac{\beta\alpha}{\beta} = E\tilde{Y}\tilde{Z}/E\tilde{D}\tilde{Z}.$$

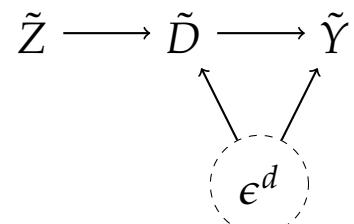


**Figure 11.8:** DAG corresponding to Figure ?? after partialling out observed confounder  $X$ .

## Aggregate Market Demand

Let's apply our approach to a canonical example in economics: the identification of the price elasticity of demand using a supply shifter as an instrument.

**Example 11.3.2** (Market Demand; Generalization of P. Wright,



**Figure 11.9:** A DAG for aggregate demand, with the latent node  $\epsilon^d$  representing the demand shock

1928 [1]) Consider the ASEM

$$\begin{aligned} Y &:= \alpha D + f_Y(X) + \epsilon^d, \\ D &:= \beta Z + f_D(X) + \rho \epsilon^d + \gamma \epsilon^s, \\ Z &:= f_Z(X) + \epsilon_Z \end{aligned}$$

where  $\epsilon^d$ ,  $\epsilon^s$  and  $\epsilon_Z$  are mean zero and uncorrelated conditional on  $X$ . In this example,  $Y$  is (log) demand,  $D$  is (log) price,  $Z$  is an observed supply shifter,  $X$  is a vector of observed demand shifters,  $\epsilon^d$  is a demand shock, and  $\epsilon^s$  is a supply shock. The key parameter is  $\alpha$ , the price elasticity of demand:

$$\alpha = \partial_d Y(d),$$

where  $Y(d) := (Y : do(D = d))$ . Here we focus on only the demand side of the market and do not attempt to explicitly model the supply side.

Example 11.3.2 is equivalent to the previous Example 11.3.1 – set  $A = \epsilon^d$ ,  $\epsilon_Y = 0$ ,  $\epsilon^s = \epsilon_D$ , and so on. Hence, the identification method is the same as before.

In econometrics, the set-up here is sometimes referred to as a *limited information* model or formulation because we are focusing on identifying only a single equation in a more complicated underlying system.

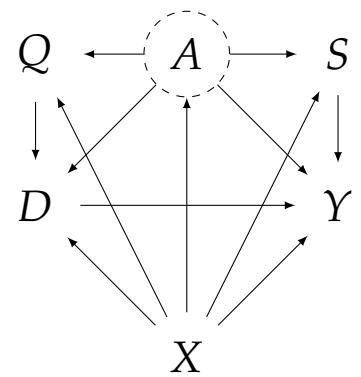
## SEMs with Griliches-Chamberlain Proxy Controls

Suppose we are interested in the causal effect of college education on earnings in the presence of an unobserved confounder – individual ability. Here we show that we can recover the effect of college education on earnings in the presence of latent ability using proxies for ability, but not the effect of ability itself.

**Example 11.3.3** (Earnings with Omitted Ability; Griliches, 1977 [2]; Griliches and Chamberlain, 1977 [9]) Consider the ASEM

$$\begin{aligned} Y &:= \alpha D + \delta A + \iota S + f_Y(X) + \epsilon_Y, \\ D &:= \gamma A + \beta Q + f_D(X) + \epsilon_D, \\ Q &:= \eta A + f_Q(X) + \epsilon_Q, \\ S &:= \phi A + f_S(X) + \epsilon_S, \\ A &:= f_A(X) + \epsilon_A, \\ X &:= \epsilon_X, \end{aligned}$$

where  $\epsilon_Y, \epsilon_D, \epsilon_Q, \epsilon_S, \epsilon_A, \epsilon_X$  have mean zero and uncorrelated conditional on  $X$ . Interpret  $Y$  as earnings,  $D$  as college



**Figure 11.10:** A DAG with Controls and Proxy Controls

degree,  $A$  as ability,  $Q$  and  $S$  as proxies of ability, and  $X$  as a set of observed background variables. Example proxies  $Q$  and  $S$  are

- $Q$  is test scores or grades in some period  $t_0$  and  $S$  is test scores or grades at a later period  $t_1$ .

The key structural parameter is  $\alpha$ , the returns to schooling; i.e.

$$\alpha = \partial_d Y(d),$$

where  $Y(d) = Y : do(D = d)$ .

After partialling out we are left with the DAG in Figure 11.11:

$$\begin{aligned}\tilde{Y} &:= \alpha\tilde{D} + \delta\tilde{A} + \iota\tilde{S} + \epsilon_Y, \\ \tilde{D} &:= \gamma\tilde{A} + \beta\tilde{Q} + \epsilon_D, \\ \tilde{Q} &:= \eta\tilde{A} + \epsilon_Q, \\ \tilde{S} &:= \phi\tilde{A} + \epsilon_S, \\ \tilde{A} &:= \epsilon_A,\end{aligned}$$

where  $\epsilon_Y, \epsilon_D, \epsilon_Q, \epsilon_S, \epsilon_A$  are uncorrelated. The idea now is to replace  $\tilde{A}$  in the equation for  $\tilde{Y}$  with  $\tilde{S}$ . Note that because  $S$  enters the  $Y$  equation directly, we cannot consider using  $\tilde{Q}$  to proxy for  $\tilde{A}$ . We still cannot learn  $\alpha$  from the regression of  $\tilde{Y}$  on  $\tilde{D}$  and  $\tilde{S}$  though as  $S$  is an imperfect proxy for  $A$ . The following result, which provides an IV approach to identify  $\alpha$ , is immediate via substitution.<sup>2</sup>

**Theorem 11.3.2** Assume that all variables in Example 11.3.3 are square-integrable. Then we have the following measurement equation:

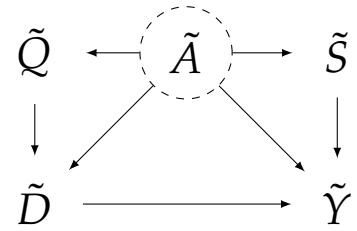
$$\tilde{Y} = \alpha\tilde{D} + \bar{\delta}\tilde{S} + U, \quad EU(\tilde{D}, \tilde{Q}) = 0,$$

$$U = -\delta\epsilon_S/\phi + \epsilon_Y; \quad \bar{\delta} = \iota + \delta/\phi.$$

Here  $\alpha$  is identified provided that  $\tilde{D}$  and the best linear predictor of  $\tilde{S}$  using  $\tilde{Q}$  and  $\tilde{D}$  have non-degenerate covariance matrix.

Note that  $\tilde{Q}$  here plays the role of a *technical instrument* for  $\tilde{S}$ . This approach recovers  $\alpha$ , but not  $\delta$ . For inference, we can employ the DML method for IV models; see also Chapter 13.

**Remark 11.3.2** (Neyman Orthogonality and DML) The formulation of the target parameter given above is Neyman-orthogonal, and high-quality estimation and statistical in-



**Figure 11.11:** A DAG with Proxy Controls After Partialling Out

2: Prove the result as a reading exercise. Substitute  $\tilde{A} = (\tilde{S} - \epsilon_S)/\phi$  in the first equation and use the assumptions on the disturbances.

ference can be carried out using DML. In essence, we just residualize the system, using cross-fitted residuals, and then apply standard instrumental variable methods from econometrics to perform inference on the structural parameter of interest.

## 11.4 Nonlinear IV Models

Once we consider nonlinear models, identification becomes a much more delicate matter. We first consider the local average treatment effect (LATE) model, and then we turn to quantile models.

### The LATE Model

An important nonlinear IV model is the local average treatment effect model (LATE), proposed by Imbens and Angrist [10].

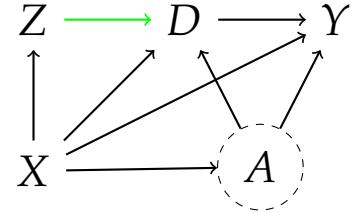
**Example 11.4.1 (LATE)** Consider the SEM, where

$$\begin{aligned} Y &:= f_Y(D, X, A, \epsilon_Y) \\ D &:= f_D(Z, X, A, \epsilon_D) \in \{0, 1\}, \\ Z &:= f_Z(X, \epsilon_Z) \in \{0, 1\}, \\ X &:= \epsilon_X, \quad A = \epsilon_A, \end{aligned}$$

where  $\epsilon$ 's are all independent, and

$z \mapsto f_D(z, A, X, \epsilon_D)$  is weakly increasing (weakly monotone).

Suppose the instrument  $Z$  is an offer to participate in a training program and that  $D$  is the actual endogenous participation in the training program. Participation in the program may depend on unobservables  $A$ , such as ability or perseverance, that also affect the eventual outcome  $Y$ . We can also have background exogenous covariates  $X$  in the model.



**Figure 11.12:** LATE models. Green arrow denotes a monotone functional relation.

Define

$$Y(d) := f_Y(d, X, A, \epsilon_Y) \text{ and } D(z) := f_D(z, X, A, \epsilon_D)$$

as the potential outcomes that result from applying fix-interventions in the corresponding equations from Example 11.4.1.

The model allows us to identify the local average treatment effect (LATE), defined as

$$\theta = E[Y(1) - Y(0) | D(1) > D(0)],$$

where  $\{D(1) > D(0)\}$  is the compliance event, where switching instrument value from  $Z = 0$  to  $Z = 1$  induces participation. Therefore LATE measures the average treatment effect conditional on compliance.

**Theorem 11.4.1** *In the LATE model, we have that  $\theta$  is identified by the ratio of two statistical parameters,*

$$\theta = \theta_1 / \theta_2,$$

where

$$\theta_1 := E(E[Y | X, Z = 1] - E[Y | X, Z = 0]),$$

and

$$\theta_2 := E(E[D | X, Z = 1] - E[D | X, Z = 0]),$$

provided that the instrument  $Z$  is relevant,  $\theta_2 > 0$ , and  $Z$  has full conditional support – namely  $0 < P(Z = 1 | X) < 1$ . Moreover,  $\theta_2$  identifies the probability of compliance:

$$\theta_2 = P[D(1) > D(0)]$$

The result has an intuitive interpretation.<sup>3</sup> In the event of compliance, the instrument moves the treatment as if experimentally, which induces quasi-experimental variation in the outcome. We measure the probability of compliance with  $\theta_2$  and the average induced changes in outcome by  $\theta_1$ . Taking the ratio is then like conditioning on the compliance event. See the proof in Section 11.7 for details.

The ratio can be recognized as the ratio of average treatment effects of  $Z$  on  $Y$  and  $D$ ,

$$\theta_1 = ATE(Z \rightarrow Y),$$

$$\theta_2 = ATE(Z \rightarrow D).$$

3: In the model with no  $X$  the ratio  $\theta_1 / \theta_2$  is equivalent to Wright [1]'s IV estimand.

This assertion follows from the application of the backdoor criterion. Therefore in order to perform inference on LATE, we can simply re-use the tools for performing inference on two ATEs.

**Remark 11.4.1** (DML for  $\theta_1/\theta_2$ ) We can apply DML to obtain  $\hat{\theta}_1$  and  $\hat{\theta}_2$  and then construct the estimator  $\hat{\theta} = \hat{\theta}_1/\hat{\theta}_2$  via the plug-in principle. This approach automatically has the Neyman orthogonality property.

## The IV Quantile Model\*

Another nonlinear IV model is the following model that exploits monotonicity in the unobservable shock in the outcome equation to obtain identification.

**Example 11.4.2** (IV Quantile Model) Consider the SEM

$$\begin{aligned} Y &= f_Y(D, X, \epsilon_Y), \\ D &= f_D(Z, X, \epsilon_Y, \epsilon_D), \\ Z &= f_Z(X, \epsilon_Z), \\ X &= \epsilon_X, \end{aligned}$$

where  $\epsilon$ 's are all independent,

$$f_Y(D, X, \cdot) : [0, 1] \mapsto \mathbb{R} \text{ is strictly increasing,}$$

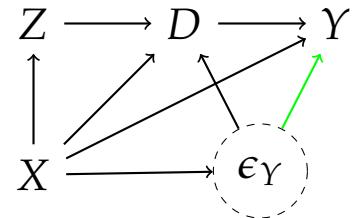
and  $\epsilon_Y$  is normalized to have uniform distribution on  $(0, 1)$ . The context could be given from the demand example, where  $Y$  is demand,  $D$  price,  $\epsilon_Y$  a demand shock,  $\epsilon_D$  a supply shock;  $X$  the set of background variables, and  $Z$  a set of instrumental variables. The function  $f_Y(d, x, u)$  is the  $u$ -th quantile of the structural function of  $f_Y(d, x, \epsilon_Y)$ , which is the demand function in this context. For example,  $f_Y(d, x, 1/2)$  is the median structural function.

The testable implication of the IV Quantile Model is the following.

**Theorem 11.4.2** In the IV Quantile Model, the testable moment restriction is

$$P[Y \leq f_Y(D, X, u) | Z, X] = u,$$

for each  $u \in (0, 1)$ . There exist regularity conditions, analogous to instrument relevance, under which the structural function  $f_Y$  is identified from this restriction.



**Figure 11.13:** IV Quantile Model. The green arrow represents a strictly monotonic effect.

In practice, linear forms  $f_Y(D, X, u) = \alpha(u)'D + \beta(u)'X$  are often used. Adopting a linear functional form leads to method of moments approaches such as the IV quantile regression for performing inference on structural quantile functions.

Code for IV Quantile Models can be found [here](#).

**Remark 11.4.2** (DML for IVQR Models) The problem of constructing DML for IVQR problems is considered open. Neyman-orthogonal approaches for the partially linear IVQR models are sketched out in the review [11] and may be a good place to start.

## 11.5 Nonlinear Models with Proxy Controls\*

An important recent development is "proximal causal inference", which generalizes early work by Griliches and Chamberlain [9].<sup>†</sup>

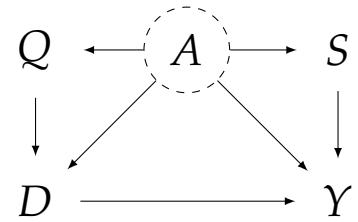
**Example 11.5.1** (Miao, Geng, and Tchetgen Tchetgen [3]) We consider the following model encoded in the DAG in Figure 11.14:

$$\begin{aligned} Y &:= f_Y(D, S, A, \epsilon_Y), \\ D &:= f_D(A, Q, \epsilon_D), \\ Q &:= f_Q(A, \epsilon_Q), \\ S &:= f_S(A, \epsilon_S), \\ A &:= \epsilon_A, \end{aligned}$$

where  $\epsilon$ 's are mutually independent. We can endow the same context to this model as in Example 11.3.3.

Here we can introduce background exogenous controls  $X$  in each of the equations, but we don't do so to save notation. Notice that the model in Example 11.5.1 generalizes the Example 11.3.3 to the nonparametric case.

**Assumption 11.5.1.** In Example 11.5.1, assume



**Figure 11.14:** A SEM with Proxy Controls  $Q$  and  $S$ . Note that conditioning on  $Q$  and  $S$  does not block the backdoor path  $Y \leftarrow A \rightarrow D$ , hence we cannot use the regression adjustment method for identification of  $D \rightarrow Y$ .

<sup>†</sup> The most relevant papers include, amongst others, the stream of work by Tchetgen Tchetgen and collaborators, as well as the dissertation work of Deaner. Here we describe some results of the first group specialized to the discrete case.

- (a) Variables  $Q$ ,  $S$  and  $A$  are finitely discrete and take on the same number of values.
- (b) The matrix  $\Pi(S \mid Q, d)$ , whose  $s^{\text{th}}$  row and  $q^{\text{th}}$  column is  $p(s \mid q, d)$ , is invertible for each value  $d$ .

Condition (b) is analogous to the usual relevance condition in IV and basically says that the two proxies  $S$  and  $Q$  have sufficient joint variation at any value of  $d$  to allow  $Q$  to serve as an “instrument” for  $S$ . The discreteness assumption can be generalized to a more general completeness condition; see Miao et al.[3] and Deaner [4]. As with the usual IV relevance condition, Condition (b) is testable from the data. In contrast, the DAG itself and the other conditions involve an unobserved variable  $A$  and are therefore generally untestable. The validity of these untestable conditions must be assessed using contextual knowledge about the empirical problem.

**Theorem 11.5.1** Under Assumption 11.5.1,  $p(y : do(d))$  is identifiable by the proximal formula:

$$p(y : do(d)) = \Pi(y \mid d, Q) \Pi(S \mid Q, d)^{-1} \Pi(S), \quad (11.5.1)$$

where  $\Pi(y \mid d, Q)$  and  $\Pi(S)$  are row and column vectors whose entries are of the form  $p(y \mid d, q)$  and  $p(s)$ .

The mnemonic way to think about the formula above is that we are doing a kind of instrumental regression of  $Y$  on  $S$ , while instrumenting  $S$  with  $Q$ , which is exactly how we dealt with the linear version of this problem in Section 11.5.

**Remark 11.5.1** [12] and [13] provide moment functions defined in terms of efficient influence functions, which possess the Neyman orthogonality property, for estimating of the average treatment effect within this proxy control setting in the presence of a high-dimensional set of control variables. These moment functions can thus serve as the foundation for the use of DML inference methods for the average treatment effect in such settings.

## Notebooks

- [DML Sensitivity R Notebook](#) analyses the sensitivity of the DML estimate in the Darfur wars example to unobserved confounders using the Sensemakr package in R.

- ▶ [DML for Partially Linear IV R Notebook](#) carries out DML IV analysis of the Acemoglu-Johnson-Robinson example, which considers the impact of the quality of institutions on economic growth, instrumenting quality of institutions with settler mortality. The notebook explores the partially linear IV model and tests for the presence of weak instruments. See Chapter 13 for further discussion of this example as well as discussion of weak identification/instruments.
- ▶ [DML for LATE Models R Notebook](#) estimates the Local Average Treatment Effects of 401(K) participation on net financial wealth. (See the second part of the notebook; the first deals with estimation of ATE of 401(k) eligibility on the financial wealth).

## 11.6 Study Problems

1. Explain omitted confounder bias to a fellow student (one paragraph). Explore using sensitivity analysis to aid in understanding robustness of economic conclusions to the presence of unobserved confounders in an empirical example of your choice. The [DML Sensitivity R Notebook](#) can be a helpful starting point but apply the ideas to a different empirical example. (You could use any of the previous examples we have analyzed).
2. Write a brief explanation of the idea of the instrumental variables regression model that would be appropriate for educating a fellow student. Discuss the idea of identifying the causal effect in this setting via path analysis in the spirit of what Philip Wright did. Illustrate your discussion with an empirical example. For example, revisit the analysis in [DML for Partially Linear IV R Notebook](#).
3. (Simulation.) Create a notebook to simulate one of the linear IV or proxy controls models that we've described. Assume there are no X's for simplicity. Demonstrate numerically why using least squares may not be appropriate due to unobserved confounding. Demonstrate numerically how using instrumental variable regression overcomes the issue.
4. (LATE etc.) Explain to a fellow student in writing one of the nonlinear models (e.g. LATE, IV quantile model,

or the nonlinear model with proxy controls) and how causal parameters in these models are identified. [DML for LATE Models R Notebook](#) could be a starting point for explaining LATE and illustrating your explanation with empirical results. (If you have a good empirical example for proxy controls, please let me know. )

## 11.7 Proofs

### Latent Confounder Bias Result: Theorem 11.2.1

The proof heavily relies on the Frisch-Waugh-Lovell partialling out theorem (FWL) and the normalization on the variance of the latent confounder:

$$E\tilde{A}^2 = 1. \quad (11.7.1)$$

The proof also relies on the properties of  $R_{U \sim V}^2$  which measures the proportion of variance of centered random variable  $U$  that is linearly explained by another centered random variable  $V$ :

$$R_{U \sim V}^2 = \frac{E\beta^2 V^2}{EU^2} = 1 - \frac{E\epsilon^2}{EU^2} = \frac{(EUV)^2}{EU^2 EV^2} = \text{Cor}^2(U, V),$$

where  $\beta = EVU/EV^2$  is the coefficient of the best linear projection of  $U$  onto  $V$ ,  $\epsilon = U - \beta V$  is the projection residual, and  $\text{Cor}(U, V)$  denotes the correlation between  $U$  and  $V$ . Note that  $R^2$  is symmetric in  $U$  and  $V$ :  $R_{U \sim V}^2 = R_{V \sim U}^2$ .

By FWL and the normalization (11.7.1), we have

$$\gamma = E\tilde{A}\tilde{D}, \quad \delta = E\bar{A}\bar{Y}/E\bar{A}^2,$$

where

$$\begin{aligned} \bar{Y} &= \tilde{Y} - \beta\tilde{D}; & \beta &= E\tilde{Y}\tilde{D}/E\tilde{D}^2; \\ \bar{A} &= \tilde{A} - \tilde{\beta}\tilde{D}; & \tilde{\beta} &= E\tilde{A}\tilde{D}/E\tilde{D}^2. \end{aligned}$$

It follows that

$$\phi^2 = \frac{\gamma^2 \delta^2}{(E\tilde{D}^2)^2} = \frac{(E\tilde{A}\tilde{D})^2}{(E\tilde{D}^2)^2} \frac{(E\bar{Y}\bar{A})^2}{(E\bar{A}^2)^2}.$$

Then the result follows from the normalization (11.7.1) and the following relations:

$$(E\tilde{D}\tilde{A})^2 = \text{Cor}^2(\tilde{D}, \tilde{A})E\tilde{D}^2 = R_{\tilde{D} \sim \tilde{A}}^2 E\tilde{D}^2,$$

$$(E\bar{Y}\bar{A})^2 = \text{Cor}^2(\bar{Y}, \bar{A})E\bar{Y}^2 E\bar{A}^2 = R_{\bar{Y} \sim \bar{A}}^2 E\bar{Y}^2 E\bar{A}^2,$$

$$E\bar{A}^2 = 1 - R_{\bar{A} \sim \tilde{D}}^2 = 1 - R_{\tilde{D} \sim \tilde{A}}^2.$$

and noting that by definition  $R_{\bar{Y} \sim \bar{A}}^2 = R_{\bar{Y} \sim \bar{A} | \tilde{D}}^2$ .

□

### Linear Proxy Model: Theorem 11.3.2.

We substitute  $\tilde{A} = (\tilde{S} - \epsilon_S)/\phi$  in the equation  $\tilde{Y} := \alpha\tilde{D} + \delta\tilde{A} + \iota\tilde{S} + \epsilon_Y$ , to obtain:

$$\tilde{Y} = \alpha\tilde{D} + \bar{\delta}\tilde{S} + U,$$

$$U = -\delta\epsilon_S/\phi + \epsilon_Y; \quad \bar{\delta} = \iota + \delta/\phi.$$

To verify

$$EU(\tilde{D}, \tilde{Q}) = 0$$

we observe using repeated substitutions that:

- $\tilde{D}$  is a linear combination of  $(\epsilon_A, \epsilon_Q, \epsilon_D)$ ,
- $\tilde{Q}$  is a linear combination of  $\epsilon_A$  and  $\epsilon_Q$ .
- $U$  is a linear combination of  $(\epsilon_S, \epsilon_Y)$ .

The conclusion follows from the assumption that

$$(\epsilon_A, \epsilon_Q, \epsilon_D, \epsilon_S, \epsilon_Y)$$

are all uncorrelated. The conclusion that  $\alpha$  is identified provided that  $\tilde{D}$  and the best linear predictor of  $\tilde{S}$  using  $\tilde{Q}$  and  $\tilde{D}$  have non-degenerate covariance matrices is left as an exercise.

□

### The LATE Result: Theorem 11.4.1

We can use, for example, the backdoor criterion to conclude that

$$\text{EE}[D \mid Z = z, X] = \text{EE}[D(z) \mid X] = ED(z).$$

Similarly,

$$\text{EE}[Y \mid Z = z, X] = \text{EE}[Y(D(z)) \mid X] = EY(D(z)).$$

Furthermore, by monotonicity, we have both

$$\theta_2 = E[D(1) - D(0)] = P(D(1) > D(0))$$

and

$$\begin{aligned} \theta_1 &= EY(D(1)) - Y(D(0)) \\ &= E(\{Y(1) - Y(0)\}1\{D(1) > D(0)\}). \end{aligned}$$

Therefore

$$\theta_1/\theta_2 = E(Y(1) - Y(0) \mid D(1) > D(0)).$$

□

## Testable Restriction for the IV Quantile Model: Theorem 11.4.2

The result is immediate from (i) the equivalence of the event  $Y \leq f_Y(D, X, u)$  and the event  $\epsilon_Y \leq u$ , which holds under the strict monotonicity assumption, and (ii) the independence of  $\epsilon_Y$  from  $Z$  and  $X$ , which follows from the stated independence conditions. Using (i) and (ii), we have

$$\begin{aligned} P[Y \leq f_Y(D, X, u) | Z, X] &= P[\epsilon_Y \leq u | Z, X] \\ &= P[\epsilon_Y \leq u] = P[U(0, 1) \leq u] = u. \end{aligned}$$

□

## Identification in the Nonlinear Proxy Variables Model: Theorem 11.5.1

To sketch a proof, the DAG implies that the observed variables  $D, Y, Q, S$  and the unobserved variable  $A$  obey the two conditional independence relations:

$$(i) \quad S \perp\!\!\!\perp (Q, D) | A \quad (ii) \quad Q \perp\!\!\!\perp Y | (A, D). \quad (11.7.2)$$

These in turn imply

$$\begin{aligned} \Pi(S | Q, d) &= \Pi(S | Q, A, d)\Pi(A | Q, d) \\ &= \Pi(S | A)\Pi(A | Q, d) \end{aligned}$$

and

$$\begin{aligned} \Pi(y | Q, d) &= \Pi(y | Q, A, d)\Pi(A | Q, d) \\ &= \Pi(y | A, d)\Pi(A | Q, d). \end{aligned}$$

We now want to solve these equations for

$$\Pi(y | A, d)$$

in terms of quantities that could be learned in the data.

We will need invertibility of  $\Pi(S | Q, d)$  which requires invertibility of both  $\Pi(S | A)$  and  $\Pi(A | Q, d)$ . Under these invertibility conditions, we have

$$\Pi(A | Q, d) = \Pi(S | A)^{-1}\Pi(S | Q, d)$$

and

$$\Pi(y | Q, d) = \Pi(y | A, d)\Pi(S | A)^{-1}\Pi(S | Q, d),$$

which yield

$$\Pi(y | A, d) = \Pi(y | Q, d)\Pi(S | Q, d)^{-1}\Pi(S | A).$$

Next, because  $A$  blocks backdoor paths between  $D$  and  $Y$ , we have that

$$p(y | a : do(d)) = p(y | a, d) \quad (11.7.3)$$

or, after integrating out  $a$ ,

$$p(y : do(d)) = \Pi(y | A, d)\Pi(A),$$

which can be further expressed as

$$\Pi(y | d, Q) \Pi(S | Q, d)^{-1} \Pi(S), \quad (11.7.4)$$

using the derivations above. □

# Bibliography

- [1] Philip G. Wright. *The tariff on animal and vegetable oils*. New York: The Macmillan company, 1928 (cited on pages 250, 256, 257, 260).
- [2] Zvi Griliches. 'Estimating the returns to schooling: Some econometric problems'. In: *Econometrica: Journal of the Econometric Society* 45.1 (1977), pp. 1–22 (cited on pages 251, 254, 257).
- [3] Wang Miao, Zhi Geng, and Eric J. Tchetgen Tchetgen. 'Identifying causal effects with proxy variables of an unmeasured confounder'. In: *Biometrika* 105.4 (2018), pp. 987–993 (cited on pages 251, 262, 263).
- [4] Ben Deaner. 'Proxy controls and panel data'. In: *arXiv preprint arXiv:1810.00283* (2018) (cited on pages 251, 263).
- [5] Carlos Cinelli and Chad Hazlett. 'Making sense of sensitivity: Extending omitted variable bias'. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 82.1 (2020), pp. 39–67 (cited on page 252).
- [6] Victor Chernozhukov, Carlos Cinelli, Whitney Newey, Amit Sharma, and Vasilis Syrgkanis. 'Omitted Variable Bias in Machine Learned Causal Models'. In: *arXiv preprint arXiv:2112.13398* (2021) (cited on page 252).
- [7] David Card. 'Using Geographic Variation in College Proximity to Estimate the Return to Schooling'. In: *Aspects of Labor Market Behavior: Essays in Honor of John Vanderkamp*. Ed. by L. N. Christofides and R. Swidinsky. Toronto: University of Toronto Press, 1995, pp. 201–222 (cited on page 255).
- [8] Joshua D. Angrist and Alan B. Krueger. 'Does Compulsory School Attendance Affect Schooling and Earnings?' In: *The Quarterly Journal of Economics* 106.4 (1991), pp. 979–1014 (cited on page 255).
- [9] Gary Chamberlain and Zvi Griliches. *More on brothers. In "Kinometrics: Determinants of Socioeconomic Success Within and Between Families "(P. Taubman, Ed.)* 1977 (cited on pages 257, 262).
- [10] Guido W. Imbens and Joshua D. Angrist. 'Identification and Estimation of Local Average Treatment Effects'. In: *Econometrica* 62.2 (1994), pp. 467–475 (cited on page 259).

- [11] Victor Chernozhukov, Christian Hansen, and Kaspar Wuthrich. ‘Instrumental variable quantile regression’. In: *arXiv preprint arXiv:2009.00436* (2020) (cited on page 262).
- [12] Yifan Cui, Hongming Pu, Xu Shi, Wang Miao, and Eric Tchetgen Tchetgen. ‘Semiparametric proximal causal inference’. In: *arXiv preprint arXiv:2011.08411* (2020) (cited on page 263).
- [13] Nathan Kallus, Xiaojie Mao, and Masatoshi Uehara. ‘Causal inference under unmeasured confounding with negative controls: A minimax learning approach’. In: *arXiv preprint arXiv:2103.14029* (2021) (cited on page 263).

# **Advanced Core 2: Debiased ML for IV and Proxy Controls Models and Robust DML Inference under Weak Identification**

# **12**

Here we specialize DML methods to partially linear models with instruments, arising either through endogeneity of the policy variable or through the use of proxy controls that we outlined in the previous chapter. We also present DML methods for LATE parameters in the fully nonlinear model with a binary endogenous treatment and binary instrument. We further examine how DML inference method can be modified to cope with weak instruments and weak identification in generic moment problems through the use of Neyman-orthogonal scores and Neyman's  $C(\alpha)$  statistic.

<b>12.1 DML Inference in Partially Linear IV Models . . . . .</b>	273
The Effect of Institutions on Economic Growth . . . . .	275
<b>12.2 DML Inference in the Interactive IV Regression Model (IRM) . . . . .</b>	278
DML Inference on LATE	278
The effect of 401(k) Participation on Net Financial Assets	279
<b>12.3 DML Inference with Weak Instruments . . . . .</b>	281
Motivation . . . . .	281
DML Inference Robust to Weak-IV in PLMs . . . . .	283
The Effect of Institutions on Economic Growth Revisited . . . . .	284
<b>12.4 Generic DML Inference under Weak Identification .</b>	286

## 12.1 DML Inference in Partially Linear IV Models

Here we consider estimation of parameters that obey the following instrumental exclusion restriction:

$$E\epsilon \tilde{Z} = 0,$$

where

$$\epsilon := \tilde{Y} - \theta_0' \tilde{D},$$

and

$$\tilde{Y} = Y - \ell_0(X), \quad \ell_0(X) = E[Y | X],$$

$$\tilde{D} = D - r_0(X), \quad r_0(X) = E[D | X],$$

$$\tilde{Z} = Z - m_0(X), \quad m_0(X) = E[Z | X].$$

Here we take the dimension of  $\tilde{Z}$  to be the same as that of  $\tilde{D}$  for simplicity.

Two key examples leading to this statistical structure are

- ▶ the partially linear instrumental variable model, and
- ▶ the partially linear model with proxy controls.

We discussed these examples and showed they fit into this structure in Chapter 11.

To estimate  $\theta_0$  and to perform inference on it we can apply the general DML algorithm with the score

$$\psi(W; \theta, \eta) := (Y - \ell(X) - \theta'(D - r(X)))(Z - m(X)), \quad (12.1.1)$$

where  $W = (Y, D, X, Z)$  and  $\eta = (\ell, m, r)$  with  $\ell, m$ , and  $r$  being  $P$ -square-integrable functions mapping the support of  $X$  to  $\mathbb{R}$ . By definition, we have that

$$E\psi(W; \theta_0, \eta_0) = 0;$$

and it is not difficult to check (via homework) that the Neyman orthogonality condition,

$$\partial_\eta E\psi(W; \theta_0, \eta_0) = 0,$$

holds at the true value  $\eta_0 = (\ell_0, m_0, r_0)$  of the nuisance parameters.

**DML for Partially Linear IV and Proxy Models**

1. Partition data indices into  $k$  folds of approximately equal size:  $\{1, \dots, n\} = \bigcup_{k=1}^K I_k$ . For each fold  $k = 1, \dots, K$ , compute ML estimators  $\hat{\ell}_{[k]}(X)$ ,  $\hat{m}_{[k]}(X)$ ,  $\hat{r}_{[k]}(X)$  of the best predictors  $\ell_0(X)$ ,  $m_0(X)$ ,  $r_0(X)$ , leaving out the  $k$ -th block of data, and obtain the cross-fitted residuals for each  $i \in I_k$ :

$$\begin{aligned}\check{Y}_i &= Y_i - \hat{\ell}_{[k]}(X_i), \\ \check{D}_i &= D_i - \hat{r}_{[k]}(X_i), \\ \check{Z}_i &= Z_i - \hat{m}_{[k]}(X_i).\end{aligned}$$

2. Compute the standard IV regression of  $\check{Y}_i$  on  $\check{D}_i$  using  $\check{Z}_i$  as the instrument; that is, obtain  $\hat{\theta}$  as the root in  $\theta$  of the following equation:

$$\mathbb{E}_n(\check{Y}_i - \theta' \check{D}_i) \check{Z}_i = 0.$$

3. Construct standard errors and confidence intervals as in the standard linear instrumental variables regression theory.

In what follows it will be convenient to use the following notation

$$\|h\|_{L^2} := \sqrt{\mathbb{E}_X h^2(X)},$$

where, as before,  $\mathbb{E}_X$  computes the expectation over values of  $X$ .

**Theorem 12.1.1** (Adaptive Inference in the Partially Linear IV Model) *Impose technical regularity conditions as in [1] which include the following key conditions: (1) the instruments are strong – namely, the singular values of  $E\tilde{D}\tilde{Z}$  are well-separated from zero – and (2) the estimators  $\hat{\ell}_{[k]}(X)$ ,  $\hat{m}_{[k]}(X)$ ,  $\hat{r}_{[k]}(X)$  provide high-quality approximations to the best predictors  $\ell_0(X)$  and  $m_0(X)$  and  $r_0(X)$  – namely,*

$$n^{1/4} \|\hat{\ell}_{[k]} - \ell_0\|_{L^2} \approx 0, \quad n^{1/4} \|\hat{m}_{[k]} - m_0\|_{L^2} \approx 0,$$

and

$$n^{1/4} \|\hat{r}_{[k]} - r_0\|_{L^2} \approx 0.$$

*Then the estimation error in  $\check{D}_i$  and  $\check{Y}_i$  has no first order effect on the behavior of  $\hat{\theta}$ :*

$$\sqrt{n}(\hat{\theta} - \theta_0) \approx (\mathbb{E}_n \tilde{D} \tilde{Z})^{-1} \sqrt{n} \mathbb{E}_n \tilde{Z} \epsilon,$$

and, as a result,  $\hat{\theta}$  concentrates in a  $1/\sqrt{n}$  neighborhood of  $\theta$  with deviations approximated by the Gaussian law:

$$\sqrt{n}(\hat{\theta} - \theta_0) \xrightarrow{d} N(0, V),$$

where

$$V = (E\tilde{D}\tilde{Z}')^{-1}E(\tilde{Z}\tilde{Z}'\epsilon^2)(E\tilde{Z}\tilde{D})^{-1}.$$

The standard error of  $\hat{\theta}$  is estimated as  $\sqrt{\hat{V}/n}$ , where  $\hat{V}$  is an estimator of  $V$  based on the plug-in principle. The result implies that the confidence interval

$$[\hat{\theta} - 2\sqrt{\hat{V}/n}, \hat{\theta} + 2\sqrt{\hat{V}/n}]$$

covers  $\theta$  for approximately 95% of the realizations of the sample. In other words, if our sample is not atypical, the interval covers the truth.

## The Effect of Institutions on Economic Growth

To demonstrate DML estimation of partially linear structural equation models with instrumental variables, we consider estimating the effect of institutions on aggregate output following the work of [2] (AJR).

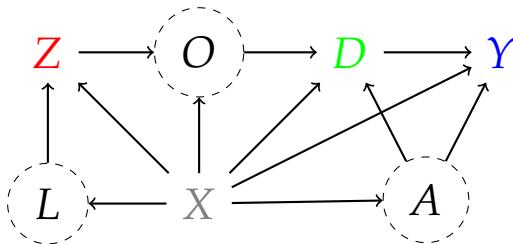
We use the same set of 64 country-level observations as AJR. The data set contains measurements of GDP, settler mortality, an index which measures protection against expropriation risk and geographic information. The outcome variable,  $Y$ , is the logarithm of GDP per capita and the endogenous explanatory variable,  $D$ , is a measure of the strength of individual property rights that is used as a proxy for the strength of institutions. To deal with endogeneity, we use an instrumental variable  $Z$ , which is mortality rates for early European settlers. Our raw set of control variables,  $X$ , include distance from the equator and dummy variables for Africa, Asia, North America, and South America.

Estimating the effect of institutions on output is complicated by the clear potential for simultaneity between institutions and output: Better institutions may generate higher incomes, but higher incomes may also lead to the development of better institutions. To help overcome this simultaneity, AJR use mortality rates for early European settlers as an instrument for institution quality. The validity of this instrument hinges on the argument that settlers set up better institutions in places where they were more likely to establish long-term settlements,

that where they were likely to settle for the long term is related to settler mortality at the time of initial colonization, and that institutions are highly persistent. The exclusion restriction for the instrumental variable is then motivated by the argument that GDP, while persistent, is unlikely to be strongly influenced by mortality in the previous century, or earlier, except through institutions.

In their paper, AJR note that their instrumental variable strategy will be invalidated if other factors are also highly persistent and related to the development of institutions within a country and to the country's GDP. A leading candidate for such a factor, as they discuss, is geography. AJR address this by assuming that the confounding effect of geography is adequately captured by a linear term in distance from the equator and a set of continent dummy variables. Using DML allows us to relax this assumption and replace it by a weaker assumption that geography can be sufficiently controlled by an unknown function of distance from the equator and continent dummies which can be learned by ML methods.

We present the verbal identification argument above in the form of a DAG in Figure 12.1. In the DAG,  $Y$  is wealth,  $O$  the quality of early institutions,  $D$  the quality of modern institutions,  $X$  observed measures of geography,  $Z$  early settler mortality,  $A$  the present day latent factors jointly determining modern institutions and wealth, and  $L$  early latent factors affecting early settler mortality. Applying the IV method here requires the identification of the causal effect of  $Z \rightarrow D$  and  $Z \rightarrow Y$ . From the DAG, we see that  $X$  blocks the backdoor paths from  $Y$  to  $Z$  and from  $D \rightarrow Z$ . This means that the instrument satisfies the required exogeneity condition conditional on  $X$ .



**Figure 12.1:** DAG for the Effect of Quality of Institutions on Wealth.

We think the story sounds plausible, but it is always important to consider threats to identification. The direct threat to identification would be if  $L$  directly affected  $Z$  and either  $O$ ,  $D$ , or  $Y$ , or, in words, if early latent factors directly affected early settler mortality and either present-day quality of institutions or present day wealth. In such cases we would need to include  $L$  as additional controls.  $L$  could represent many different latent

Lasso	Forest	Best
0.77	0.88	0.88
(0.17)	(0.32)	(0.32)

**Note:** Estimated coefficient from DML based estimation of a linear instrumental variables model based on orthogonal estimating equations in the AJR example. Column labels denote the method used to estimate nuisance functions. We used  $K = 20$  folds for cross-fitting.

**Table 12.1:** DML Estimates of the Effect of Institutions on Output

factors. For example, one might conjecture that the religion of early European settlers (e.g., Catholic vs Protestant) is related to the type of institutions they would establish and to their mortality rates upon colonization. In their original study, AJR did examine this threat by checking robustness of their result with respect to the inclusion of religion variables. They also examined the use of other additional controls to assess robustness to other potential sources of confounding.<sup>1</sup>

We report results from applying DML following the procedure outlined in Section 9.4 in Table 12.1. For cross-fitting, we use 20 folds given how small the data set is. Here we just tried two successful methods, Lasso and Random Forests, for learning the nuisance functions  $\eta$ . As predictors in the Lasso estimates, we used a dictionary formed by taking latitude and longitude<sup>2</sup> interacted with continent dummies as technical controls. For the Random Forest estimates, we simply included taking latitude and continent dummies as raw controls. The Random forest predicts outcomes  $Y$ ,  $D$ , and  $Z$  better than Lasso. The resulting best DML estimate is therefore based on DML with Random forest used in all ML steps.

In this example, we see uniformly large and positive point estimates across all procedures considered, and estimated effects are statistically significant at the 5% level in all cases. We note the estimates are somewhat smaller than the baseline estimates reported in AJR – an estimated coefficient of 1.10 with estimated standard error of 0.46 ([2], Table 4, Panel A, column 7) – but are qualitatively similar, indicating a strong and positive effect of institutions on output.

1: It is good to revisit their analysis using ML tools. See their [Data archive](#) to get started.

## 12.2 DML Inference in the Interactive IV Regression Model (IRM)

### DML Inference on LATE

In this section, we consider estimation of local average treatment effects (LATE) with a binary treatment variable,  $D \in \{0, 1\}$ , and a binary instrument,  $Z \in \{0, 1\}$ . As before,  $Y$  denotes the outcome variable, and  $X$  is the vector of covariates. Consider the following statistical parameter:

$$\theta_0 = \frac{\mathbb{E}[\mathbb{E}(Y | Z = 1, X) - \mathbb{E}(Y | Z = 0, X)]}{\mathbb{E}[\mathbb{E}(D | Z = 1, X) - \mathbb{E}(D | Z = 0, X)]}.$$

This parameter is the ratio of the average predictive effects of  $Z$  on  $Y$  and of  $D$  on  $Y$ . Under the assumptions laid out in Chapter 11, this statistical parameter is a causal parameter – the average treatment effect for compliers (LATE).

To set up estimation, define the regression functions:

$$\begin{aligned}\mu_0(Z, X) &= \mathbb{E}[Y | Z, X] \\ m_0(Z, X) &= \mathbb{E}[D | Z, X] \\ p_0(X) &= \mathbb{E}[Z | X],\end{aligned}$$

Define the nuisance parameter  $\eta = (\mu, m, p)$  to denote square-integrable functions  $\mu$ ,  $m$ , and  $p$ , with  $\mu$  mapping the support of  $(Z, X)$  to  $\mathbb{R}$  and  $m$  and  $p$  respectively mapping the support of  $(Z, X)$  and  $X$  to  $(\varepsilon, 1 - \varepsilon)$  for some  $\varepsilon \in (0, 1/2)$ . The true value of the nuisance parameter is  $\eta_0 = (\mu_0, m_0, p_0)$ , the regression functions defined above.

The DML estimator of  $\theta_0$  employs the orthogonal score

$$\begin{aligned}\psi(W; \theta, \eta) &:= \mu(1, X) - \mu(0, X) + H(p)(Y - \mu(Z, X)) \\ &\quad - \left( m(1, X) - m(0, X) + H(p)(D - m(Z, X)) \right) \theta,\end{aligned}$$

for  $W = (Y, D, X, Z)$  and

$$H(p) := \frac{Z}{p(X)} - \frac{(1 - Z)}{1 - p(X)}.$$

It is easy to verify (for homework) that this score satisfies the moment condition

$$\mathbb{E}\psi(W; \theta_0, \eta_0) = 0$$

and also the Neyman orthogonality condition

$$\partial_\eta \mathbb{E}\psi(W; \theta_0, \eta_0) = 0$$

at the true value  $\eta_0 = (\mu_0, m_0, p_0)$  of the nuisance parameter.

Therefore we can apply the generic ML algorithm to this problem, including the selection of the best ML methods to estimate the nuisance parameters.

**Theorem 12.2.1** (DML for LATE) *Suppose conditions specified in [1] hold. In particular, suppose that the overlap condition holds; namely, for some  $\epsilon > 0$  with probability 1,*

$$\epsilon < p_0(X) < 1 - \epsilon.$$

*Further, suppose  $\epsilon < \hat{p}_{[k]}(X) < 1 - \epsilon$  and that estimators  $\hat{p}_{[k]}$ ,  $\hat{m}_{[k]}$ ,  $\hat{\mu}_{[k]}$  provide high quality approximation to  $p_0$ ,  $m_0$ , and  $\mu_0$  in the sense that*

$$n^{1/2} \|\hat{p}_0 - p_0\|_{L^2} \times \left( \|\hat{\mu}_0 - \mu_0\|_{L^2} + \|\hat{m}_0 - m_0\|_{L^2} \right) \approx 0.$$

*Then estimation of the nuisance parameters does not affect the behavior of the estimator to the first order; namely,*

$$\sqrt{n}(\hat{\theta} - \theta_0) \approx \sqrt{n} \mathbb{E}_n \varphi_0(W),$$

*where*

$$\varphi_0(W) = -J_0^{-1} \psi(W; \theta_0, \eta_0), \quad J_0 := \mathbb{E}(m_0(1, X) - m_0(0, X)).$$

*Consequently,  $\hat{\theta}$  concentrates in a  $1/\sqrt{n}$ -neighborhood of  $\theta_0$  and the sampling error  $\sqrt{n}(\hat{\theta} - \theta_0)$  is approximately normal*

$$\sqrt{n}(\hat{\theta} - \theta_0) \stackrel{\text{a}}{\sim} N(0, V), \quad V := \mathbb{E}\varphi_0(W)\varphi_0(W)'.$$

Variance estimation and confidence intervals are constructed as in the generic DML algorithm.

## The effect of 401(k) Participation on Net Financial Assets

Here we continue to re-analyze the effects of 401(k)'s on household financial assets, picking up from Section 9.3. In this section, we report the LATE in this example where we take the endogenous treatment variable to be *participating* in a 401(k) plan

using 401(k) *eligibility* as instrument. Even after controlling for features related to job choice, it seems likely that the actual choice of whether to participate in an offered plan would be endogenous. Of course, we can use eligibility for a 401(k) plan as an instrument for participation in a 401(k) plan under the conditions that were used to justify the exogeneity of eligibility for a 401(k) plan outlined in Section 9.3.

We report DML results of estimating the LATE of 401(k) participation using 401(k) eligibility as an instrument in Table 12.2. We employ the procedure outlined in Section 12.2 using the same ML estimators to estimate the quantities used to form the orthogonal estimating equation as we employed to estimate the ATE of 401(k) eligibility in Section 9.3, so we omit the details for brevity. Looking at the results, we see that the estimated causal effect of 401(k) participation on net financial assets is uniformly positive and statistically significant across all of the considered methods. As when looking at the ATE of 401(k) eligibility, it is reassuring that the results obtained from the different flexible methods are broadly consistent with each other. It is also interesting that the results based on flexible ML methods are broadly consistent with, though somewhat attenuated relative to, those obtained by applying the same specification for controls as used in [3] and [4] and using a linear IV model which returns an estimated effect of participation of \$13,102 with estimated standard error of (1922). The attenuation may suggest that the simple intuitive control specification used in the original baseline specification is not sufficiently flexible.

R Notebook on DML for Impact of 401(K) Participation on Financial Wealth

Lasso	Forest	Boosting	Neural Net.	Ensemble	Best
8944 (2259)	11764 (1788)	11133 (1661)	11186 (1795)	11173 (1641)	11113 (1645)

**Note:** Estimated ATE and standard errors from a linear model (Panel B) and heterogeneous effect model (Panel A) based on orthogonal estimating equations. Column labels denote the method used to estimate nuisance functions.

**Table 12.2:** DML Estimates of LATE on 401(k) Participation on Net Financial Assets

## 12.3 DML Inference with Weak Instruments

### Motivation

As a simple motivating example, consider a statistical model with instrumental moment conditions and one-dimensional endogenous variable  $D$  when there are either no controls or we are able to partial them out perfectly. In this case, the IV estimator takes the form

$$\hat{\theta} = \mathbb{E}_n \tilde{Z} \tilde{Y} / \mathbb{E}_n \tilde{Z} \tilde{D},$$

and we have that

$$\sqrt{n}(\hat{\theta} - \theta) = \sqrt{n} \mathbb{E}_n \tilde{Z} \epsilon / \mathbb{E}_n \tilde{Z} \tilde{D}.$$

When  $\mathbb{E}_n \tilde{Z} \tilde{D}$  is well-separated away from zero, we invoke the approximation

$$\sqrt{n} \mathbb{E}_n \tilde{Z} \epsilon / \mathbb{E}_n \tilde{Z} \tilde{D} \stackrel{a}{\sim} N(0, \text{E}\tilde{Z}^2 \epsilon^2) / \text{E}\tilde{Z} \tilde{D}. \quad (12.3.1)$$

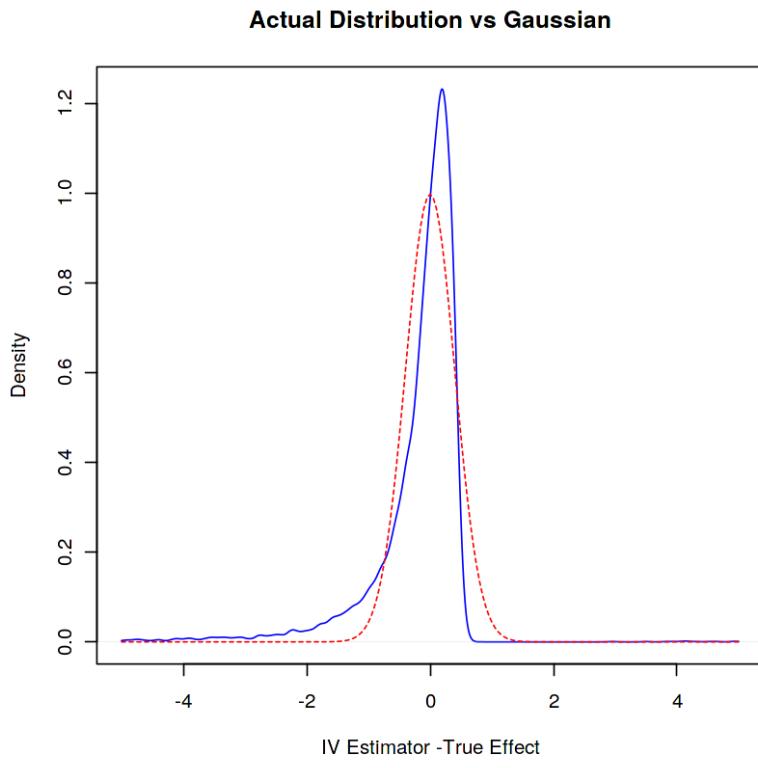
However, this approximation is not reliable when instruments are “weak” – when  $\mathbb{E}_n \tilde{Z} \tilde{D}$  appears close to zero. Intuitively, we may worry that small changes in a sample that result in relatively small changes in  $\mathbb{E}_n \tilde{Z} \tilde{D}$  may still have large impacts on the estimator  $\hat{\theta}$  when  $\mathbb{E}_n \tilde{Z} \tilde{D}$  is near zero because  $\mathbb{E}_n \tilde{Z} \tilde{D}$  shows up in the denominator. That is, (12.3.1), which essentially ignores sampling variation in  $\mathbb{E}_n \tilde{Z} \tilde{D}$ , may provide a very poor approximation to the actual finite sample sampling behavior of the IV estimator.

We illustrate the potential poor performance of the usual asymptotic approximation (12.3.1) in Figure 12.2 which reports results from a simulation experiment in which  $\text{E}[\tilde{Z} \tilde{D}]$  is close to zero. Here we see the sampling distribution (given by the blue curve) of the IV estimator deviates strongly from the normal approximation (given by the red curve). Note that by varying how close  $\text{E}[\tilde{Z} \tilde{D}]$  is to zero, one can make the differences more or less pronounced.

In principle, we can detect the weak instrument problem by testing whether  $\beta = 0$  in the projection equation

$$\tilde{D} = \beta \tilde{Z} + U, \quad \text{E}\tilde{Z} \tilde{D}.$$

“Weak identification” (or “weak instruments” in IV models) refers to settings in which we cannot confidently conclude a testable identifying assumption holds in our data. In our simple IV model, the parameter  $\theta$  is not identified when  $\text{E}[\tilde{Z} \tilde{D}] = 0$  as solving the population moment condition requires solving  $\text{E}[\tilde{Z} \tilde{D}] \theta = \text{E}[\tilde{Z} \tilde{Y}]$ .



**Figure 12.2:** Actual sampling distribution of the IV estimator in a simulation experiment vs the normal approximation of the IV Estimator using weak instrument.

Econometricians have found that the normal approximation above is adequate for inferential properties if the t-statistics for the null  $\beta = 0$  is bigger than 4:<sup>2</sup>

$$|\hat{\beta} - \beta| / \text{se}(\hat{\beta}) > 4.$$

If this happens, then we are said to have a "strong" instrument. If this test for the strong instrument is passed, then it is relatively safe to use the normal approximation for inference with the IV estimator. If not, using the usual asymptotic approximation is not safe, but is there anything else that we can do?

Of course there is. There are a variety of alternative inferential procedures whose behavior does not hinge on the well-separation of  $\mathbb{E}_n[\tilde{Z}\tilde{D}]$  from zero. Here, we consider one specific approach based upon the statistic

$$C(\theta) = \frac{|\mathbb{E}_n[(\tilde{Y}_i - \theta\tilde{D}_i)\tilde{Z}_i]|^2}{\mathbb{V}_n[(\tilde{Y}_i - \theta\tilde{D}_i)\tilde{Z}_i]/n}.$$

If  $\theta_0 = \theta$ , then  $C(\theta) \stackrel{a}{\sim} N(0, 1)^2 = \chi^2(1)$ . Therefore, we can reject the hypothesis  $\theta_0 = \theta$  at level  $a$  (for example  $a = .05$  for a 5% level test) if  $C(\theta) > c(1 - a)$  where  $c(1 - a)$  is the  $(1 - a)$ -quantile of a  $\chi^2(1)$  variable. The probability of falsely rejecting the true hypothesis is approximately  $a \times 100\%$ . To construct

2: These are called "rules of thumb" and are based on simulation experiments.

a  $(1 - \alpha) \times 100\%$  confidence region for  $\theta$ , we can then simply invert the test by collecting all parameter values that are not rejected at the  $\alpha$  level:

$$CR(\theta) = \{\theta \in \Theta : C(\theta) \leq c(1 - \alpha)\}.$$

In more complex settings with many controls or controls that enter with unknown functional form, we can simply replace the residuals  $\tilde{Y}$ ,  $\tilde{D}$ , and  $\tilde{Z}$  by machine learned cross-fitted residuals  $\check{Y}$ ,  $\check{D}$ , and  $\check{Z}$ . Thanks to the orthogonality of the IV moment condition underlying the formulation outlined above, we can formally assert that the properties of  $C(\theta)$  and the subsequent testing procedure and confidence region for  $\theta$  continue to hold when using cross-fitted residuals. We will further be able to apply the general procedure to cases where  $D$  is a vector, with a suitable adjustment of the statistic  $C(\theta)$ .

## DML Inference Robust to Weak-IV in PLMs

Here, we present a more general version of weak identification robust inference, including implementation and theoretical details, in settings where we want to use machine learning to aid in controlling for confounding variables  $X$ .

### DML Weak-IV-Robust Inference for PLIV Model

1. **Initialize:** Let  $\Theta$  be a known parameter space that contains the true value  $\theta_0$ . Using the DML-PLIV algorithm, produce the cross-fitted residuals:  $\check{Y}_i$ ,  $\check{D}_i$ , and  $\check{Z}_i$ . Using the cross-fitted residuals and for  $\theta \in \Theta$ , compute the moment function

$$\check{M}(\theta) := \mathbb{E}_n[(\check{Y}_i - \theta' \check{D}_i) \check{Z}_i],$$

the empirical covariance function

$$\check{\Omega}(\theta) := \mathbb{V}_n[(\check{Y}_i - \theta' \check{D}_i) \check{Z}_i],$$

and the score statistic

$$C(\theta) := n \check{M}(\theta)' \check{\Omega}^{-1}(\theta) \check{M}(\theta).$$

2. **Robust Confidence Region:** Construct the approximate  $(1 - \alpha) \times 100\%$  confidence region as

$$CR(\theta_0) = \{\theta \in \Theta : C(\theta) \leq c(1 - \alpha)\},$$

where  $c(1 - a) := (1 - a)$ -quantile of a  $\chi^2(m)$  variable,  
 where  $m = \dim(Z_i)$ .

In order to state the next result, define the oracle version of the moment and covariance functions given in Step 1 of the DML Weak-IV-Robust Inference algorithm,

$$\hat{M}(\theta) = \mathbb{E}_n[(\tilde{Y}_i - \theta' \tilde{D}_i) \tilde{Z}_i]$$

and

$$\hat{\Omega}(\theta) = \mathbb{V}_n[(\tilde{Y}_i - \theta' \tilde{D}_i) \tilde{Z}_i],$$

which are defined in terms of the true residuals  $\tilde{Y}_i$ ,  $\tilde{D}_i$ , and  $\tilde{Z}_i$ .

**Theorem 12.3.1** *Under regularity conditions, estimation of the nuisance parameters does not affect the behavior of the C statistic in the sense that*

$$C(\theta_0) \approx n \hat{M}(\theta_0)' \hat{\Omega}^{-1}(\theta_0) \hat{M}(\theta_0) \stackrel{a}{\sim} \chi^2(m).$$

Consequently, the test rejects the true value with approximate probability  $a$ ,

$$P(C(\theta) \geq c(1 - a)) \approx a,$$

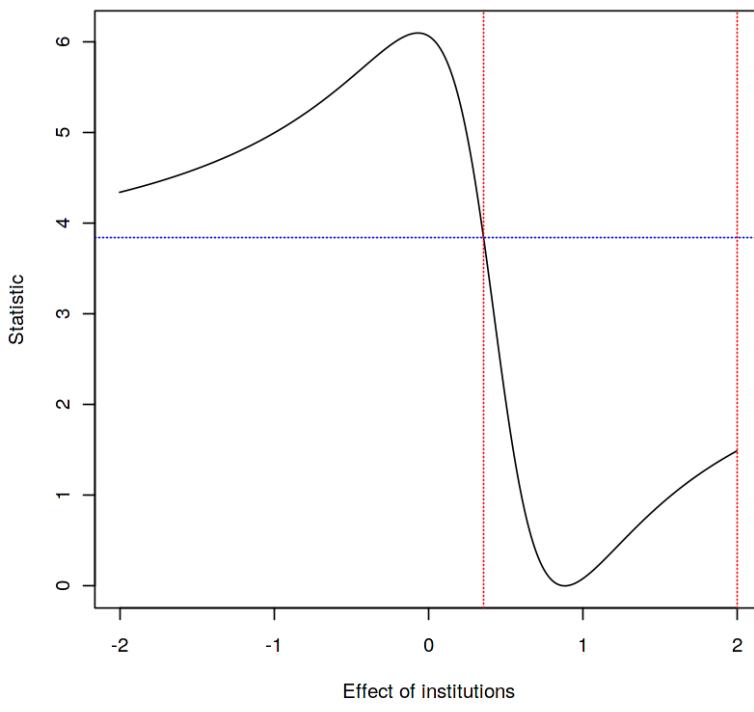
and the confidence region  $CR(\theta_0)$  contains  $\theta_0$  with approximate probability  $(1 - a)$ ,

$$P(\theta_0 \in CR(\theta_0)) \approx (1 - a).$$

## The Effect of Institutions on Economic Growth Revisited

We illustrate the use of DML weak identification robust inference by revisiting the AJR example from Section 12.1. Recall that Random Forests performed best in all auxiliary predictive steps in our original exercise in this example, so we only consider the use of Random Forests to form residuals in this section.

After partialling out controls using Random Forests, we run the regression of  $\tilde{D}$  on  $\tilde{Z}$  to assess the strength of the instruments. The resulting t-statistic is approximately 2, much lower than the "safety" threshold of 4. As such, we conclude that we have a weak instrument and proceed with weak identification robust inference.



**Figure 12.3:** Construction of weak IV robust confidence regions for the effect of institutions on output using DML. Values of the  $C(\theta)$  statistic are shown on the vertical axis; values of  $\theta$  tested on the horizontal axis. The 90% confidence region is given by the red vertical bars.

We implement the robust inferential approach from the previous subsection considering  $\Theta = [-2, 2]$  as our parameter space for the causal effect of institutions on wealth. We note that, because the outcome we consider is the logarithm of GDP per capita, the range  $[-2, 2]$  includes extremely (likely implausibly) large negative and positive effects, so restricting attention to this range *a priori* seems reasonable. We illustrate the procedure in Figure 12.3 which plots the value of the test statistic  $C(\theta)$  for  $\theta \in [-2, 2]$ .

The resulting 95% confidence region is

$$[.35, 2].$$

We can compare this region to the confidence region produced by the usual Gaussian asymptotic approximation which is not robust to weak identification:

$$[.88 \pm 2 \cdot 0.32] = [.24, 1.52].$$

Both the usual and robust confidence regions are consistent with relatively large positive effects of institutions on wealth. However, it is interesting that the lower end of the robust confidence region is larger than the lower end of the usual region and that this difference is economically meaningful. That

is, we could not rule out that a one unit increase in quality of institutions causes an approximately a 27% increase in GDP per capita looking at the usual interval, while we could rule out all effect sizes smaller than 42% with the robust interval. The difference between a 27% and 42% increase in GDP per capita is certainly economically relevant. Given that the instruments are weak, we should, of course, rely on the robust confidence interval.

## 12.4 Generic DML Inference under Weak Identification

We now present a generally applicable formulation of weak identification robust inference. This formulation covers the problem of weak instruments in the context of LATE estimation as well as other problems where Neyman-orthogonal scores are available.

The initialization and first two steps to our approach to weak identification robust inference are the same as in the Generic DML Algorithm: We then use these estimates of the nuisance parameters in conjunction with the score function at a fixed value of  $\theta$  to construct a score test statistic analogous to  $C(\theta)$  from the previous section which can be used to test the hypothesis that  $\theta_0 = \theta$  and to form confidence regions. We collect this procedure in the following algorithm:

### Generic DML Robust to Weak Identification

1. **Initialize:** Provide the data frame  $(W_i)_{i=1}^n$ , the Neyman-orthogonal score/moment function  $\psi(W, \theta, \eta)$  and the name and model for ML estimation method(s) for learning nuisance parameters  $\eta$ . Specify  $\Theta$  to be a known parameter space that contains the true value  $\theta_0$ . We then take a K-fold random partition  $(I_k)_{k=1}^K$  of observation indices  $\{1, \dots, n\}$  such that the size of each fold is about the same, and for each  $k \in \{1, \dots, K\}$ , we construct a machine learning estimator  $\hat{\eta}_{[k]}$  using data  $(X_i)_{i \notin I_k}$ , that is, all the data *except* the data from the  $k^{\text{th}}$  fold.
2. **Estimate Moments and Their Variance:** Letting  $k(i) = \{k : i \in I_k\}$ , construct the moment function

$$\check{M}(\theta) = \mathbb{E}_n[\psi(W_i; \theta, \hat{\eta}_{[k(i)]})],$$

covariance function,

$$\check{\Omega}(\theta) = \mathbb{V}_n[\psi(W_i; \theta, \hat{\eta}_{[k(i)]})],$$

and score statistic

$$C(\theta) = n\check{M}(\theta)' \check{\Omega}^{-1}(\theta) \check{M}(\theta).$$

**3. Confidence Region:** Construct the approximate  $(1 - a) \times 100\%$  confidence region as

$$CR(\theta_0) = \{\theta \in \Theta : C(\theta) \leq c(1 - a)\}$$

where  $c(1 - a)$  is the  $(1 - a)$ -quantile of a  $\chi^2(m)$  variable where  $m = \dim(\check{M}(\theta))$ .

Note that this confidence region simply collects all values  $\theta \in \Theta$  that are not rejected by testing  $\theta_0 = \theta$  using test statistic  $C(\theta)$  at the  $a$ -level.

As in the previous section, we define oracle versions of the moment and covariance functions from the preceding algorithm for use in stating formal results:

$$\hat{M}(\theta) = \mathbb{E}_n[\psi(W_i; \theta, \eta_0)],$$

$$\hat{\Omega}(\theta) = \mathbb{V}_n[\psi(W_i; \theta, \eta_0)].$$

**Theorem 12.4.1** Under regularity conditions, estimation of nuisance parameters does not affect the behavior of the  $C(\theta)$  statistic in the sense that

$$C(\theta_0) \approx n\hat{M}(\theta_0)\hat{\Omega}^{-1}(\theta_0)\hat{M}(\theta_0) \stackrel{a}{\sim} \chi^2(m).$$

Consequently, a test that rejects when  $C(\theta) \geq c(1 - a)$ , for  $c(1 - a)$  the  $(1 - a)$ -quantile of a  $\chi^2(m)$  variable, rejects the true value with approximate probability  $a$ :

$$P(C(\theta_0) \geq c(1 - a)) \approx a.$$

Similarly, the confidence region corresponding to this test,  $CR(\theta_0)$ , contains  $\theta_0$  with approximate probability  $(1 - a)$ :

$$P(\theta_0 \in CR(\theta_0)) \approx (1 - a).$$

## Notebooks

- ▶ [The R Notebook on Weak IV](#) provides a simulation experiment illustrating the weak instrument problem with IV estimators.
- ▶ [R Notebook on DML for Impact of Institutions on Output](#) provides DML analysis of the impact of institutions on a country's wealth following AJR. The notebook first proceeds with the analysis assuming strong identification. It then notes the weak instrument problem and performs DML analysis that is robust to weak identification.
- ▶ [R Notebook on DML for Impact of 401\(K\) Eligibility on Financial Wealth](#) provides application of DML inference to learn predictive/causal effects of 401(K) eligibility on net financial wealth. (Note: The results produced in this notebook and provided in the text are slightly different than those in the original paper [1]. The replication files for [1] are given at the following [Github repository](#). The difference is due to our use of a single split of the sample in producing the results for this text while the results in [1] are based on a method that aggregates results across multiple data splits.)

## Notes

The statistic  $C(\theta)$  is Neyman's  $C(\alpha)$  statistic.

## Study Problems

1. Experiment with [The R Notebook on Weak IV](#), varying the strength of the instrument. How strong should the instrument be in order for the conventional normal approximation based on strong identification to provide accurate inference? Based on your experiments, provide a brief explanation of the weak IV problem to a friend.
2. Experiment with [R Notebook on DML for Impact of 401\(K\) Eligibility on Financial Wealth](#). Apply the analysis to another data-set, for example JTPA data from our colleague [Joshua Angrist's data archive](#). Don't forget to draw your

DAGs!

3. Experiment with [R Notebook on DML for Impact of Institutions on Output](#). Try to extend the analysis by including religion as a control variable or consider another empirical application to another IV example. (See some potential applications at the [the Angrist data archive](#)). In the case of a new application, don't forget to draw your DAGs!
4. (Theoretical). Verify that the scores for the partially linear IV methods are Neyman orthogonal.

# Bibliography

- [1] Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. ‘Double/debiased machine learning for treatment and structural parameters’. In: *The Econometrics Journal* 21.1 (2018), pp. C1–C68 (cited on pages 274, 279, 288).
- [2] Daron Acemoglu, Simon Johnson, and James A. Robinson. ‘The colonial origins of comparative development: An empirical investigation’. In: *American economic review* 91.5 (2001), pp. 1369–1401 (cited on pages 275, 277).
- [3] James M. Poterba, Steven F. Venti, and David A. Wise. ‘401(k) Plans and Tax-Deferred savings’. In: *Studies in the Economics of Aging*. Ed. by D. A. Wise. Chicago, IL: University of Chicago Press, 1994, pp. 105–142 (cited on page 280).
- [4] James M. Poterba, Steven F. Venti, and David A. Wise. ‘Do 401(k) Contributions Crowd Out Other Personal Saving?’ In: *Journal of Public Economics* 58.1 (1995), pp. 1–32 (cited on page 280).

# **TOPICS**

# Inference on Heterogeneous Treatment Effects

# 13

Here, we introduce DML for inference on heterogenous treatment effects. We first review conditional and group average treatment effects as methods to analyse differences in the impact of treatment arising from the value of covariates. We show how these effects can be estimated using OLS and then extend the approach to high-dimensional settings using ML techniques. We illustrate the approach using the 401(k) example. This chapter provides a sketch of some ideas that have emerged in this literature, mostly following Semenova and Chernozhukov (2021) [1].

13.1 Inference on CATEs and Best Linear Predictors of CATEs . . . . .	293
Using Least Squares Methods for Learning CATEs . . . . .	295
Using ML Methods for Learning CATEs . . . . .	297
Application to 401(k) Example . . . . .	298

### 13.1 Inference on CATEs and Best Linear Predictors of CATEs

We consider the standard setup for analyzing the effect of a binary treatment in the presence of a high dimensional set of controls  $Z$ . Specifically, we have potential outcomes  $Y(0)$  and  $Y(1)$  and assigned treatment  $D$  that obey the usual conditional exogeneity condition:

$$D \perp Y(d) \mid Z.$$

We observe the outcome  $Y := Y(D)$ , the treatment assignment  $D$ , and the high-dimensional set of controls  $Z$ .

Our main interest in this section is the Conditional Average Treatment Effect (CATE) defined as

$$t(X) = E[Y(1) - Y(0)|X],$$

where  $X$  is (typically) a low-dimensional subset of covariates  $Z$ . Our main goal is summarizing the potentially complex and high-dimensional treatment effect function, which may depend on the entire vector  $Z$ , in terms of a lower-dimensional object  $X$ . We may be interested in such summaries for aiding interpretation or for policy reasons where we are interested in effects among particular recipients defined by observable characteristics.

For example, in the context of the 401(K) analysis from previous chapters, we have that  $Y$  is a household's total net financial assets,  $D$  is 401(k) eligibility status, and  $Z$  is the entire set of household characteristics. We might then take  $X$  to be income in which case the CATE  $t(X)$  shows the expected effect of 401(k) eligibility on total financial assets for a subject whose income level is  $X$ .

The key to adaptively estimating and potentially performing inference for the CATE is expressing it as a conditional expectation of an unbiased signal:

$$t(X) = E[Y(\eta_0) \mid X],$$

where the signal takes the form

$$Y(\eta) = H(\mu)(Y - g(D, Z)) + g(1, Z) - g(0, Z),$$

We focus on the binary treatment case, but note that the approach readily extends to more general settings.

with nuisance parameters  $\eta := (\mu, g)$  and

$$H(\mu) := \frac{D}{\mu(Z)} - \frac{1-D}{1-\mu(Z)}.$$

Here,  $g(D, Z)$  and  $\mu(Z)$  are square integrable functions with  $\mu(Z)$  taking on values in  $[\epsilon, 1-\epsilon]$  for some  $\epsilon > 0$ . The true values of these nuisance parameters are  $\eta_0 := (\mu_0, g_0)$  defined as

$$\mu_0(Z) := P(D = 1 | Z), \quad g_0(D, Z) := E[Y | Z, D].$$

Importantly, the signal has the Neyman orthogonality property:

$$\partial_\eta E[Y(\eta_0) | X] = 0.$$

Making use of the representation of the CATE as the conditional expectation of  $Y(\eta_0)$ , we then estimate the CATE using the following steps:

### Generic DML for CATE

1. Partition data indices into  $k$  folds of approximately equal size:  $\{1, \dots, n\} = \bigcup_{k=1}^K I_k$ . For each fold  $k = 1, \dots, K$ , compute ML estimators  $\hat{g}_{[k]}(D, Z)$  and  $\hat{\mu}_{[k]}(Z)$  of the best predictors  $g_0(D, Z)$  and  $\mu_0(Z)$  leaving out the  $k$ -th block of data. For any observation  $i \in I_k$ , define

$$\begin{aligned} Y_i(\widehat{\eta}) &= Y_i(\widehat{\eta}_k) \\ &= H_i(Y_i - \hat{g}_{[k]}(D_i, Z_i)) + \hat{g}_{[k]}(1, Z_i) - \hat{g}_{[k]}(0, Z_i) \end{aligned}$$

$$\text{where } H_i = \frac{D_i}{\hat{\mu}_{[k]}(Z_i)} - \frac{1-D_i}{1-\hat{\mu}_{[k]}(Z_i)}.$$

2. Use low-dimensional or high-dimensional regression methods to regress  $Y_i(\widehat{\eta})$  on covariate features  $X_i$ . If low-dimensional methods are used, inference on CATE can proceed using standard results for low-dimensional methods.

Under regularity conditions, the second step is adaptive, meaning all the learning guarantees and confidence intervals are approximately the same as if we knew the nuisance parameters  $\eta_0$ . This adaptation holds true because of the conditional Neyman orthogonality of  $Y(\eta)$ . We note that this adaptivity *does not imply* that inferential objects, e.g. confidence intervals,

can readily be obtained if high-dimensional methods are used in Step 2. We discuss implementation and inferential issues in more detail in the following sections.

## Using Least Squares Methods for Learning CATEs

Here we focus on using least squares in the second step of the general approach given above.

Consider approximating or summarizing the function  $t(x)$  by a linear combination of basis functions:

$$p(x)'\beta_0,$$

where  $p(x)$  is K-dimensional dictionary with

$$K \ll n.$$

For example,  $p(x)$  could be a vector of group indicators or a vector of orthogonal polynomials or splines.

The parameter  $\beta_0$  is chosen to minimize approximation error to the CATE:

$$\min_{\beta} E(t(X) - p(X)'\beta)^2.$$

$p(x)'\beta_0$  is thus the best linear predictor for the CATE; that is,

$$\beta_0 = (E p(X)p(X))^{-1} E p(X)Y(\eta_0).$$

An important, easily interpretable special case is when we choose to use group indicators in forming the basis functions  $p(x)$ . Specifically, we define group indicators as

$$G_k(X) = 1(X \in R_k),$$

where  $R'_k$ s are mutually exclusive regions in the covariate space. For example, in the 401(k) example, we may be interested in average treatment effects for observations with household income less than \$10,000, observations with income between \$10,000 and \$20,000, etc. which we could capture by defining  $G_1(X) = 1(\text{Income} < \$10,000)$ ,  $G_2(X) = 1(\$10,000 \leq \text{Income} < \$20,000)$ , etc. With the group indicators defined, we then set

$$p(X) = (G_1(X), \dots, G_K(X))'.$$

In this case, the Best Linear Predictor  $\beta_0$  recovers the GATEs (group average treatment effects).

More generally,  $p(x) \in \mathbb{R}^d$  represents a  $d$ -dimensional dictionary of series/sieve basis functions – e.g., polynomials or splines – and  $p(x)' \beta_0$  corresponds to the best linear approximation to the target function  $t(x)$  in the given dictionary. Under some smoothness conditions,  $\pi(x) = p(x)' \beta_0$  will approximate  $t(x)$  as the dimension of the dictionary becomes large, and our inference will target this function.

Taking the approach motivated above to a sample of data, we have that the natural estimator of the best linear predictor of the CATE is

$$p(x)' \hat{\beta},$$

where  $\hat{\beta}$  is the ordinary least squares estimate of  $\beta_0$  defined on the random sample  $(X_i, D_i, Y_i)_{i=1}^N$ :

$$\hat{\beta} := \left( \frac{1}{N} \sum_{i=1}^N p(X_i)p(X_i)' \right)^{-1} \frac{1}{N} \sum_{i=1}^N p(X_i)Y_i(\hat{\eta}).$$

Semenova et al. [1] derive a complete set of results for the properties of  $p(x)' \hat{\beta}$  as an estimator of the best linear predictor curve  $x \mapsto p(x)' \beta_0$ . Importantly, these results establish an asymptotic approximation that allows simultaneous inference on all parameters of the best linear predictor curve. The key result verifies that the large sample properties of  $\hat{\beta}$  are the same as those of

$$\bar{\beta} := \left( \frac{1}{N} \sum_{i=1}^N p(X_i)p(X_i)' \right)^{-1} \frac{1}{N} \sum_{i=1}^N p(X_i)Y_i(\eta_0),$$

when ML tools are used to estimate the nuisance parameter  $\eta_0$  so long as the ML tools perform sufficiently well. Thus, we can employ standard methods for inference about  $\beta_0$  and the best linear predictor curve functional  $x \mapsto p(x)' \beta_0$ .

Specifically, leveraging that  $\hat{\beta}$  and  $\bar{\beta}$  have the same large sample properties, we have

$$\hat{\beta} - \beta_0 \sim_a N(0, \hat{\Omega}/N),$$

where

$$\hat{\Omega} := \hat{Q}^{-1} \left[ \mathbb{E}_n p(X_i)p(X_i)' (Y_i(\hat{\eta}) - p(X_i)' \hat{\beta})^2 \right] \hat{Q}^{-1} \quad (13.1.1)$$

for  $\hat{Q} = \mathbb{E}_n p(X_i)p(X_i)'$ .

This result can be used to construct uniform confidence bands

for

$$x \mapsto p(x)'\beta_0,$$

which can be interpreted as confidence intervals for CATE  $x \mapsto t(x)$  if the approximation error is small.

## Using ML Methods for Learning CATEs

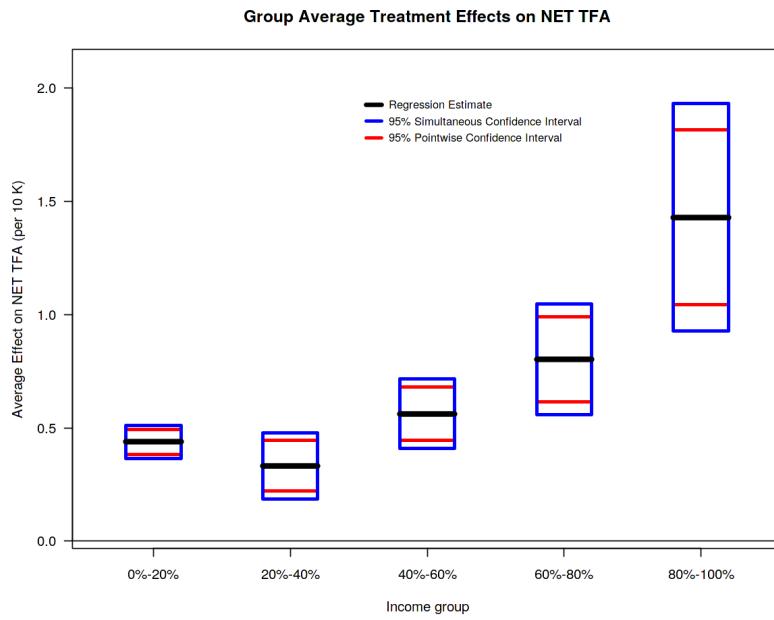
In principle, any high-quality predictive method, including all of those we have discussed in this text, may be used in Step 2 of the Generic DML for CATE algorithm:

- ▶ Athey and Wager [2] consider the use of Random Forests in Step 2, terming their procedure “Generalized Random Forests.”
- ▶ A related idea, termed “Orthogonal Forests” was developed by Syrgkanis et al. [3] as part of the Econ ML toolkit.

Application of these methods is conceptually straightforward: we simply use the estimated observation specific score,  $Y(\hat{\eta})$ , in place of the raw outcome  $Y$  when applying the learning method. The resulting estimates target the CATE specifically and will achieve the same performance guarantees as would be obtained if  $\eta_0$  were known as long as  $\hat{\eta}$  provides a sufficiently high-quality estimate of  $\eta_0$ .

While adaptive estimation of the CATE can be obtained fairly generally, it is important to note that  $X$  should be low dimensional if we want to obtain confidence intervals or perform hypothesis tests. Genovese and Wasserman (Annals of Stats, 2008) [4] show that there do not exist adaptive confidence bands for estimation of the curve  $t(X)$  except under very restrictive assumptions more generally. They suggest instead to construct adaptive bands that cover a surrogate function  $\pi$  which is close to, but simpler than,  $t$ .

In our construction above where we discuss the use of OLS with low-dimensional  $X$ , the surrogate  $\pi$  represents either GATEs or the best linear approximation of the CATE. Inferential guarantees are also available for the case where  $X$  is low-dimensional and Random Forests are used. Inferential results for low-dimensional surrogates  $\pi$  based on other methods should also be possible, though we note that GATEs and best linear predictors more generally are readily interpretable and will likely be useful in many settings.



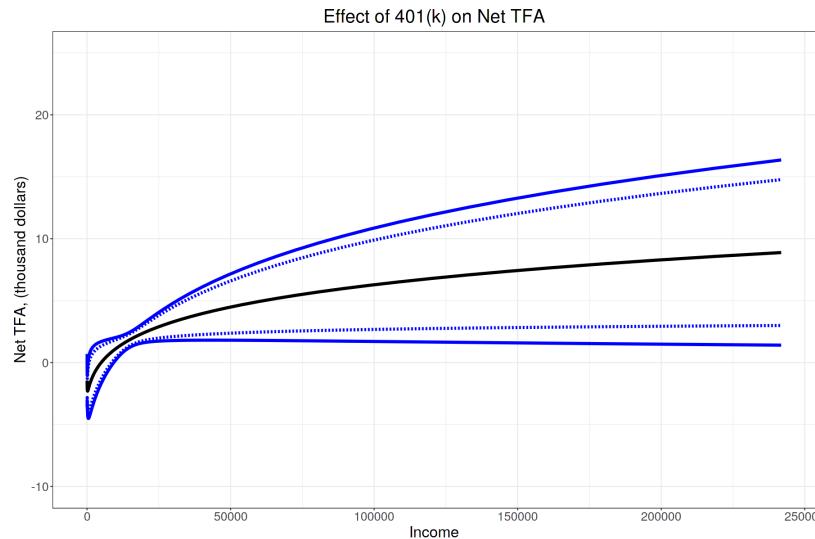
**Figure 13.1:** Inference on ATE of 401(k) Eligibility by Income Group

## Application to 401(k) Example

We illustrate estimation of CATEs and GATEs by revisiting the 401(k) example. Here, we consider the effect of 401(k) eligibility on net total financial assets controlling for household characteristics. We consider heterogeneity of this effect as a function of income. We consider two different ways to summarize these heterogeneous effects: GATEs based on coarse income categories and a summary of the CATE given income based on a collection of polynomial terms in  $\log(\text{Income})$ .

We show estimates and confidence bands on GATEs by income groups in Figure 13.1. Here, groups correspond to income quintiles; e.g. the first group has households with income smaller than the 20<sup>th</sup> percentile, second group has households with income between the 20<sup>th</sup> and 40<sup>th</sup> percentile, and so on. Point estimates are provided by the central solid black bands. We represent pointwise confidence bands with the red lines in the interior of the box for each GATE. These bands would be appropriate for inference if one were interested *ex ante* in a single, pre-specified GATE. For example, one might be specifically interested in the eligibility effect among low income individuals and thus focus on the pointwise intervals over the first GATE. Finally, uniform confidence bands are given by the upper and lower bounds of the box for each GATE. These uniform bands provide a coverage guarantee for all five reported GATEs and would be appropriate for inference in settings where one was interested in all five effects and did not *ex ante* have a single specific GATE of interest.

R Notebook for DML on CATE analyzes the ATE of 401(K) conditional on income.



**Figure 13.2:** Inference on CATE of 401(k) Eligibility Conditional on Log-Income

We illustrate using a polynomial in log income to approximate the CATE in Figure 13.2. Point estimates are given by the central black line while the blue lines provide confidence bands. The narrower – dashed – confidence bands are pointwise and would be appropriate for a scenario in which one had a single, pre-specified value of income of interest. The wider confidence bands are uniform, providing a coverage guarantee for the *entire* best linear predictor curve  $x \mapsto p(x)\beta_0$ . That is, any path for the entire curve that would not be rejected will lie entirely within the uniform confidence band. Finally, note that coverage guarantee extends to the true CATE function  $x \mapsto t(x)$  if the approximation error of the polynomial to the true CATE is small.

## Notebooks

- [R Notebook for DML on CATE](#) analyzes ATE of 401(K) conditional on income.

# Bibliography

- [1] Vira Semenova and Victor Chernozhukov. 'Debiased machine learning of conditional average treatment effects and other causal functions'. In: *The Econometrics Journal* 24.2 (2021), pp. 264–289 (cited on pages 292, 296).
- [2] Stefan Wager and Susan Athey. 'Estimation and Inference of Heterogeneous Treatment Effects using Random Forests'. In: *Journal of the American Statistical Association* 113.523 (2018), pp. 1228–1242 (cited on page 297).
- [3] Miruna Oprescu, Vasilis Syrgkanis, and Zhiwei Steven Wu. 'Orthogonal random forest for causal inference'. In: *International Conference on Machine Learning*. PMLR. 2019, pp. 4932–4941 (cited on page 297).
- [4] Christopher Genovese and Larry Wasserman. 'Adaptive confidence bands'. In: *The Annals of Statistics* 36.2 (2008), pp. 875–905 (cited on page 297).

# **APPENDIX**

# Index

- ANOVA, 9
- Best Linear Prediction, 6
- Best Linear Predictor, 6
  - collider bias, 115
  - confidence band, 75
  - cross-validation, 71
  - linear regression, 6
  - multitask, 139
- Neyman orthogonality, 75
- overfitting, 11
- parents, 114
- partialling-out, 76
- predictive effects, 13
- Random Forest, 132
- Sample splitting, 12