A causal perspective on model robustness: case studies in health and sensor data

Apinan Hasthanasombat

King's College

September 2022

Department of Computer Science and Technology

William Gates Building
15 JJ Thomson Avenue
Cambridge CB3 0FD
United Kingdom

This thesis is submitted for the degree of Doctor of Philosophy

Declaration

I Apinan Hasthanasombat, hereby declare that this report and the work described in it are my own work, unaided except as may be specified below, and that the report does not contain material that has already been used to any substantial extent for a comparable purpose.

This dissertation is copyright © Apinan Hasthanasombat
All trademarks used in this dissertation are hereby acknowledged.

A causal perspective on model robustness: case studies in health and sensor data

Abstract

Robustness of predictive deep models is a challenging problem with many implications. It is of particular importance when models are used in safety-critical applications, such as healthcare. However, there is yet to be agreement on a comprehensive definition on what it means for a model to be robust, and a theory on why these issues arise. Given the general nature of the problem, existing work related to robustness is spread across different areas of research. Existing research has considered a range of robustness aspects, for instance robustness to small input perturbations, which arise from the study of adversarial examples, but there is also robustness to different domains for the same task, and robustness issues which arise from object placement, transplanting, lighting, weather conditions, or object style, as some examples.

This thesis explores a formulation of robustness in terms of the assumed structural causal model (SCM) which generates the observed data. The SCM allows these different types of robustness issues to be viewed in a unifying way. Using this view, this work furthers the connection between prediction robustness and the assumed structural causal model by suggesting that optimising for prediction performance across a diverse set of distributions from the same SCM will move the model closer to the causal predictor of the target variable, providing a theoretical foundation to optimise purely for prediction in the setting where training and testing data are not independently and identically distributed.

Formulating robustness in this way suggests that large deep models should, in general, be more susceptible to robustness issues; while some of these issues have been observed in applications such as computer vision, it has been less discussed in others. We investigate the robustness of state-of-the-art deep (SotA) classifiers in human activity recognition using a new proposed benchmark informed by the causal formulation, and show that a simpler model is at least as robust as SotA deep models whilst being at least ten times faster to train. The causal view of robustness additionally hints at the idea that less data can be beneficial for robustness, contrary to popular belief that more data is always better. To test this idea, a data selection algorithm is proposed based on inverting the idea of a popular causal inference procedure for tabular data. The robustness of a model trained on the selected subset of data is evaluated through synthetic and semi-synthetic data experiments. Under certain conditions the data subset improves robustness and subsequently data efficiency.

Apinan Hasthanasombat

Acknowledgements

I owe thanks to many people. I'd like to thank Professor Cecilia Mascolo for the opportunity, her guidance, and her patience. I'd like to thank Dr Damon Wischik for introducing me to causality, for the countless hours he has spent discussing the finer details of the problems I grappled with during my studies, and for teaching me how to think - but all of the mistakes in this thesis are my own. I'd like to thank Lise for her help navigating the departmental processes, and being a friendly face throughout my time here.

I'm thankful for everyone I've met in college: Rob, Koen, Kerri, Allison, Emelyn, Agnes, Shaun, Nathaniel, Mathilda, Janeska, Sam, Tom, Georgia, James, Joe, and others who I may have missed, for expanding my horizons beyond Computer Science, and undoubtedly made me a less boring person.

I want to thank Angela and the catering team at King's who always made sure I am fed, especially on the tougher days.

I'd like to thank the people I've met in the department. The mobile systems group - Lorena, Andrea, Dimitris, Dionysis, Krittika, Xiao, Young, Tong, Abhirup, Hong, Andreas, Jing, Sally and others I may have missed - for sharing the journey. I'm greatful for all of the discussions with the causality reading group: Omar, Andrei, Jiaee, Ragul, and Andy. I'm also glad for the company of familiar faces from my MPhil days - Sian, Paul and Dan B.

Thank you Daniel, my good friend and flatmate during the last year, for putting up with my frequent disappearances and the drop in quality of flatmately duties.

Thank you Professor Robin Hirsch, and Mr Paul Press, who have taught me to think from first principles, and who greatly influenced my thinking.

I'm also very grateful of the Cambridge Trust and King's College, who have provided substantive financial support required to complete my studies, and the department of Computer Science and Professor Mascolo for additional financial support.

Finally I'd like to thank my family and Feli for putting up with me, in particular in the previous two years where I have been less present - mentally and physically - than I would have liked. I intend to make up for it.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Deep neural network (NN) models have made remarkable progress in a wide range of applications in the past decade, offering superior performance in a range of tasks; from code completion [1], protein structure prediction [2] to image generation [3]. However, these models are often treated as a black-box - we do not have much insight into what exactly they learn.

The black-box nature of these models mean that they often fail in unexpected ways. The most well-known case is perhaps that of adversarial examples in computer vision; an instance is shown in Figure 1.1. When a small amount of noise is added to an image of a panda, the model mistakenly classifies it as a gibbon, even though to a human the image remains identical. These adversarial examples have led to a field of research dedicated to training models which are robust to these kinds of small input perturbations [4].

However, there are other model failures observed in literature that do not come from small input perturbations. For instance, Figure 1.2 shows another vision model making incorrect predictions based on novel placements of objects

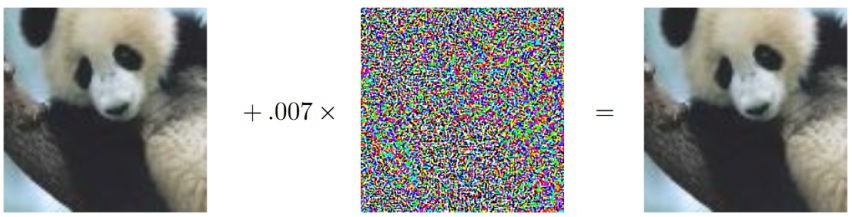


Figure 1.1: An adversarial example that makes an image model misclassify its input, figure from [5].

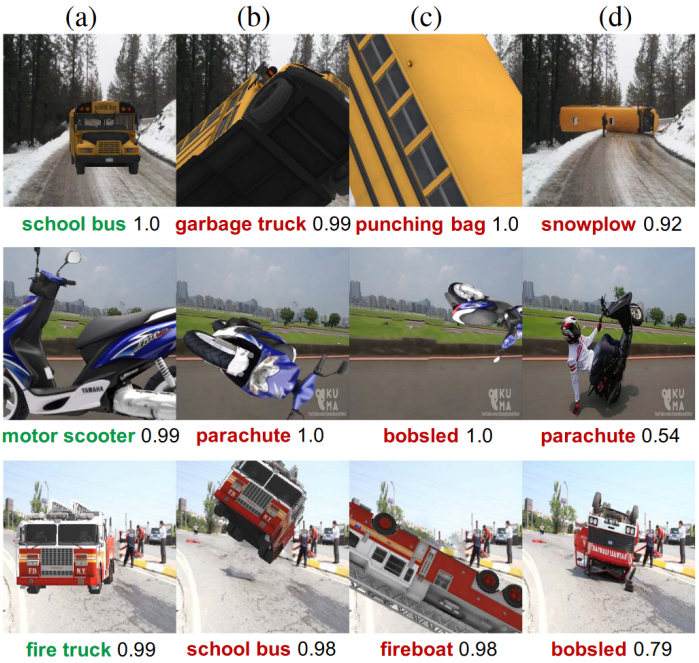


Figure 1.2: Computer vision models are fooled by novel placements of known objects, figure from [7].

it has been trained to recognise. Figure 1.3 shows three types of medical imaging models, two columns per type. The left column shows a negative sample and the right a positive sample corresponding to the condition each model is trained to detect. We can see that the model performs as expected in clean images (first row), whilst this performance degrades with various types of perturbations (second to fourth row) which are not limited to small input perturbations. Figure 1.4 shows different domains of an object class, corresponding to different styles which the object can take in an image (real image, sketch, painting etc.). A model which truly learns the concept of a class should be able to classify well across these different style domains; a task easily achieved by humans. However current models still struggle with this task [6].

Rather than training models to be specifically robust to any particular type of failure (style, object placement, small input perturbations and so on), we should aim to develop a general theory which connects these different failures and explain why they occur. Such a theory would lead to a more universal understanding of robustness that is not specific to any particular type of change to the input, which, in turn, could lead to methods to train models that are robust against a larger class of failures.

This more general theory of robustness is important to have as these black-

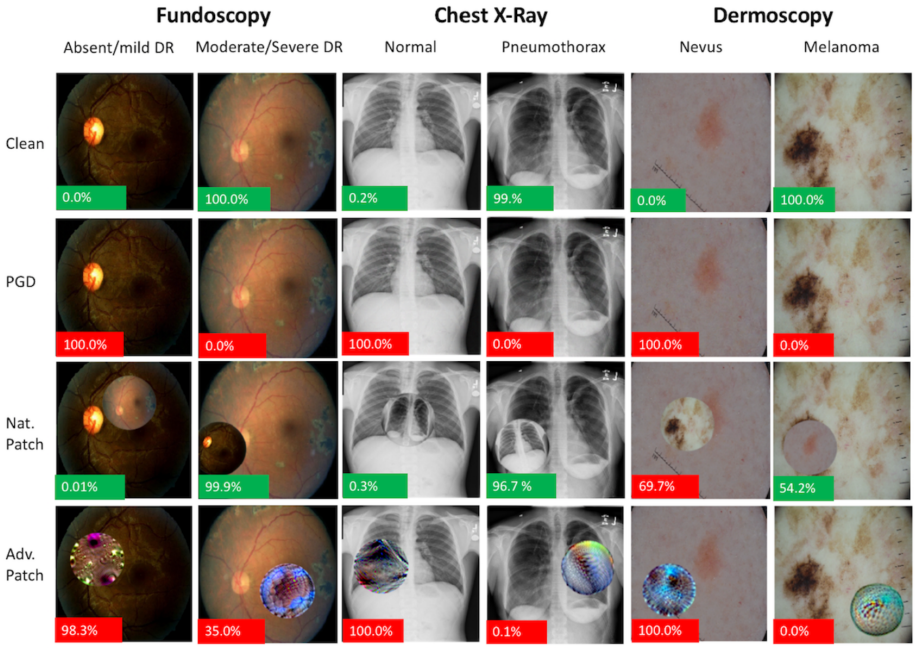


Figure 1.3: Predictions of three types of medical imaging models change according to different perturbations, figure from [8].

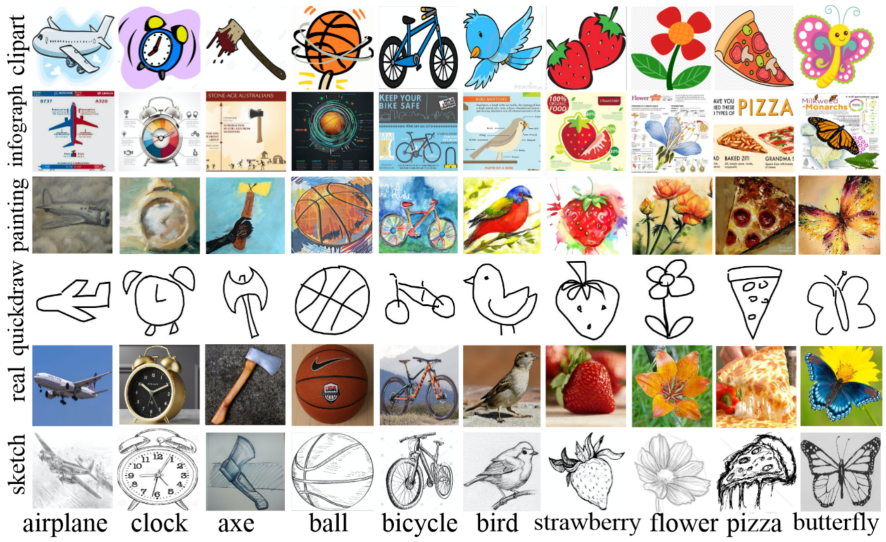


Figure 1.4: Different styles of various object classes, figure from [6].

box models are increasingly being used in safety-critical application areas, where it may encounter changes to the input unseen during training that can cause it to fail. 'Safety' in this context is interpreted broadly to encompass several aspects; self-driving cars and medical decision-making can be considered safety-critical as they pertain to physical safety, but so do models involved in informing loan applications, college admissions or policing. Poor predictions due to model failure (which is used either directly or in combination with other information to make decisions) could lead to unintended negative outcomes. It could compromise people's physical safety (self-driving cars), health (medical decision-making), access to economic opportunities or exacerbate inequalities (loan applications, college admissions, and policing) [9].

This is of particular importance in heath applications, which have recently seen a rapid increase in adoption of neural network models. From detecting pneumothorax and melanomas in the skin [8], Alzheimer's disease [10] to breast cancer screening [11]. It is of vital importance that these models are robust in the most general sense, but this is not yet the case. For example, a study of a deep learning system used to detect diabetic retinopathy deployed in Thailand found that it struggled with unideal images taken in real clinics [12]. A more recent and memorable example of attempts to use deep models in healthcare is during Covid-19, which resulted in models not fit for real deployments [13, 14, 15]. My own involvement in developing Covid-19 related models [16, 17] also suggest that robustness remains one of the biggest challenges to real deployment.

This thesis takes a step towards this general theory of robustness by adopting the view that data is generated by a structural causal model (SCM) and formulating robustness based on this foundation. This will be called the 'causal view' of robustness throughout the rest of this work. Based on this causal view this thesis explores the following main question: What implications does this causal formulation have on our understanding of model robustness? Specifically, can the ideas in the causality literature be used to develop methods that can help train more robustness models?

In the following chapters we will explore three main implications of the causal view of robustness. First, we make the case that exploiting data from different intervention distributions is the most practical approach to train robust models, based on the deficiencies in causal discovery and the strong requirement of knowing the causal graph in causal inference, which is impractical in most settings. However, there are currently no guarantees that using different intervention distributions will give a robust model, except work by Peters [18] which showed that the causal parent can be identified using appropriate hypothesis tests. However these tests are impractical. We instead derive an idea called empirical causal convergence, which suggests that optimising only for

prediction performance across different intervention distributions will move the model closer to the causal predictor. Secondly, the causal formulation suggests that robustness can be empirically evaluated by using different intervention distributions. As an example, we test this in the context of human activity recognition (HAR) which have been traditionally evaluated using a different setup. A starting benchmark based on different intervention distributions on a binary classification task is put together, whilst controlling for sensor location, sampling rate, and measurement units. The robustness of SotA deep models in the HAR literature is shown to be lower than previously thought, and unexpectedly similar to a much simpler model which uses an appropriate inductive bias that is much more efficient to train, suggests that deep end-to-end models aren't always a silver bullet in HAR. We also observed that training using multiple intervention distributions improve robustness across all considered models, consistent with what we would expect from the causal view. Finally, by using the causal view and inverting the idea of matching, a causal inference method, we investigate whether training with a selected subset of data from different intervention distributions can improve robustness in low-dimensional tabular data. Using synthetic and semi-synthetic data experiments there is empirical evidence to suggest that this is the case under certain conditions. This is contrary to the popular belief that more data is always beneficial, and questions our understanding of the problem of generalisation.

1.1 Research questions and contributions

Question 1. Can existing work in causality help train robust models?

The causality section in Chapter 2 summarises my interpretation of current limitations in causality methodology for use in training robust models. I argue that learning from multiple interventions is the best way forward, and this is also discussed in Chapter 3; I present the empirical causal convergence result which provides additional evidence that this approach is consistent with our goals of training a robust model.

Question 2. Do robustness issues also affect current HAR models, and does the understanding of robustness using causality help improve it?

I implement a way to measure robustness based on the causal interpretation and show that current models face robustness issues. I further show that making changes in accordance with the causal view (reducing hypothesis class, using an appropriate inductive bias, training with multiple domains) aids robustness.

Question 3. Can our interpretation of robustness using causality provide new methods to train robust models?

Based on the causal view I propose a way to select a subset of data for

training which can aid model robustness. This is interesting as it is generally thought that more data leads to better models. The result demonstrates that less data can be better for robustness under certain conditions, and is consistent with the causal interpretation. This raises questions for future research to better understand when being selective about training data could improve robustness.

1.2 Thesis outline

In Chapter 2, we cover some background on neural networks (NN), causality and robustness. Common NN architectures such as the feed-forward, convolutional and long short-term memory (LSTM) are introduced, along with the no free lunch theorem. Causality is introduced through the discussion of the randomised controlled trial (RCT), and the problem of confounding. This is followed by an outline on literature in causal inference and causal discovery. Other works related to robustness are reviewed, and their relation to the work in this thesis is discussed. Finally we touch on other emerging work which uses causality in machine learning.

In Chapter 3, we outline a theoretical perspective which lays the foundation to reason about model robustness and subsequently design algorithms which can improve it. Robustness is defined in terms of the underlying SCM, which begs the question of whether works from causality can be used to improve model robustness. We discuss that because of limitations in causal discovery, and that the causal graph is unknown in many practical settings, the most practical approach to finding causal variables is to exploit different data environments, first proposed by Peters [18]. However, the hypothesis tests adopted by Peters is impractical. We then cover the theory which motivates one of the main learning paradigms today - Empirical Risk Minimisation (ERM), and why this should lead to good performance on independently and identically distributed (iid) test datasets. However in real deployments, test data is not iid but can be thought of as having a common SCM. Combining these two separate works lead to the development of the empirical causal convergence idea. This suggests that optimising for prediction performance can still be a valid training paradigm even if test data is not iid. The formulation of robustness and ideas in this chapter is used to motivate work in subsequent chapters.

In Chapter 4, I take the causal view into a more practical setting by evaluating the robustness of Human Activity Recognition (HAR) models. Based on the perspective developed previously, a new benchmark is proposed to measure the robustness of HAR models. Two state-of-the-art deep models are then tested against this benchmark and are shown to significantly underperform compared to the traditional setup used for evaluation. A simpler model is proposed based

on an appropriate inductive bias, and is shown to perform at least as well as the SotA deep models (p-values 0.13 and $5.75\text{x}10^{-4}$) whilst being at least 10 times faster to train. It is further shown, in support of the empirical causal convergence idea, that training with multiple datasets under similar conditions improves the performance both in and out of distribution. This is not always the case, as seen by the negative transfer phenomenon reported in the transfer learning literature. Finally, it is shown that improvements to a specific target dataset can be achieved without complex transfer techniques, which raises questions about whether gains from transfer learning are data or method induced.

In Chapter 5, the causal view is used to develop a data selection algorithm which can train more robust models using less data under certain conditions. First, the idea of matching for causal effect estimation is introduced. Based on why matching works for causal estimation, a data selection heuristic is proposed such that the size of the hypothesis class considered by the learning procedure is reduced, whilst still containing the causal predictor. This should in theory improve model robustness using less data, under some conditions. Using synthetic data, it is shown that this reduced dataset does improve robustness as measured by the median accuracy over non-iid test sets. It is then further shown using semi-synthetic data, where real covariate values were used from the infant health development program (IHDP) dataset but where the outcomes were simulated, that this can also lead to more robust models. This is compared to training the model on the entire dataset using normal training (Empirical Risk Minimisation - ERM), training using invariant risk minimisation (IRM) and training using a randomly sampled subset of equal size.

Finally in Chapter 6, I reflect on the causal perspective of robustness explored in this thesis, its limitations, and its implications given the presented work. I discuss promising directions for future work and some interesting open questions.

1.3 Publications

I have been involved in the following works:

Related to this thesis

Less data can be better for domain generalization.
Apinan Hasthanasombat, Abhirup Ghosh, Cecilia Mascolo (Working paper)

Investigating domain-agnostic performance in activity recognition using accelerometer data.

Apinan Hasthanasombat, Abhirup Ghosh, Dimitris Spathis, Cecilia Mascolo (Ubicomp, workshop on human activity sensing corpus and its application (HASCA) 2022)

Other work

Understanding the effects of the neighbourhood built environment on public health with open data.
Apinan Hasthanasombat and Cecilia Mascolo. In Proceedings of The Web Conference 2019 (WWW2019).

Exploring longitudinal cough, breath, and voice data for COVID-19 disease progression prediction via sequential deep learning: model development and validation.
Ting Dang, Jing Han, Tong Xia, Dimitris Spathis, Erika Bondareva, Chloe Brown, Jagmohan Chauhan, Apinan Hasthanasombat, Andreas Grammenos, Andres Floto, Pietro Cicuta, Cecilia Mascolo. In Journal of Medical Internet Research (JMIR). 2022;24(6):e37004 DOI: 10.2196/37004

Sounds of COVID-19: exploring realistic performance of audio-based digital testing.
Jing Han, Tong Xia, Dimitris Spathis, Erika Bondareva, Chloe Brown, Jagmohan Chauhan, Ting Dang, Andreas Grammenos, Apinan Hasthanasombat, Andres Floto, Pietro Cicuta, Cecilia Mascolo. In Npj Digital Medicine. 1 (January 2022), 19. DOI:https://doi.org/10.1038/s41746-021-00553-x

COVID-19 sounds: a large-scale audio dataset for digital respiratory screening.
Tong Xia, Dimitris Spathis, Chloe Brown, Jagmohan Chauhan, Andreas Grammenos, Jing Han, Apinan Hasthanasombat, Erika Bondareva, Ting Dang, Andres Floto, Pietro Cicuta, Cecilia Mascolo. In Proceedings of 35th Conference on Neural Information Processing Systems (NeurIPS 2021) Track on Datasets and Benchmarks.

The INTERSPEECH 2021 computational paralinguistics challenge: COVID-19 cough, COVID-19 speech, escalation & primates.
Bjrn W. Schuller, Anton Batliner, Christian Bergler, Cecilia Mascolo, Jing Han, Iulia Lefter, Heysem Kaya, Shahin Amiriparian, Alice Baird, Lukas Stappen, Sandra Ottl, Maurice Gerczuk, Panagiotis Tzirakis, Chlo Brown, Jagmohan Chauhan, Andreas Grammenos, Apinan Hasthanasombat, Dimitris Spathis, Tong Xia, Pietro Cicuta, Leon J. M. Rothkrantz, Joeri Zwerts, Jelle Treep,

Casper Kaandorp. In Proceedings of INTERSPEECH 2021

Exploring automatic COVID-19 diagnosis via voice and symptoms from crowdsourced data.
Jing Han, Chloe Brown, Jagmohan Chauhan, Andreas Grammenos, Apinan Hasthanasombat, Dimitris Spathis, Tong Xia, Pietro Cicuta, Cecilia Mascolo. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP21)

Exploring automatic diagnosis of COVID-19 from crowdsourced respiratory sound data.
Chloe Brown, Jagmohan Chauhan, Andreas Grammenos, Jing Han, Apinan Hasthanasombat, Dimitris Spathis, Tong Xia, Pietro Cicuta, Cecilia Mascolo. In Proceedings of the ACM Conference on Knowledge Discovery and Data (KDD). Health Day: AI for COVID.

Chapter 2

Background

This chapter covers relevant ideas in neural networks, causality, and robustness which will be referred to throughout the thesis. The background on causality will mainly be referenced in Chapter 3, but randomised controlled trials (RCTs) and matching are also relevant to Chapter 5. The neural network background is applicable to Chapter 4 and 5, where experiments are conducted using the types of architectures described here. The background on robustness provides general context as to the empirical observations in the literature which motivates our causal formulation of robustness, and how other kinds of robustness studied in literature fits into this view.

Outline. The feed-forward, convolution, and long short-term memory neural network architectures are briefly described, followed by the back-propagation algorithm and no-free lunch theorem. There is a discussion of existing work in causality, starting from the randomised controlled trial (RCT), to traditional causal inference algorithms, which are divided into two categories - controlling confounders and special cases. This is followed by a discussion of the principles which underlie common causal discovery algorithms. Attention is then turned to robustness, first on the issue of its definition. Existing work on robustness to small input perturbations, distributional robustness, and several artifacts of non-robust models reported in the literature are covered, along with work at the intersection of robustness and causality.

2.1 Neural networks

Neural networks are high-dimensional parametric functions which is used to approximate some target function by training on data. Training is achieved via the backpropagation algorithm. It is sometimes referred to as a black-box model as it is often unclear how to interpret the learned weights, unlike a linear

regression with a small number of features. In this section three architectures found in this thesis, specifically in Chapter 4 and 5, is covered, along with how they are trained - the back-propagation algorithm. Additionally we briefly look at the no-free-lunch theorem, which introduces the idea that there is no single model that will perform well in all possible problems, this is referred to in Chapter 4.

2.1.1 Feed-forward

The feed-forward neural network, also known as the multilayer perceptron (MLP), is one of the simplest architectures. It is 'feed-forward' because input is passed through several iterations of transformations, each iteration often called a 'layer', and no information from later layers of the network are put back into the previous layers.

The general idea is that the network has an input and output layer, with one or more hidden layers. These layers in the simplest case can be a linear transformation. The key component is that after each hidden layer, there should be a non-linear transformation such as the rectified linear unit (ReLu), defined as $Relu(x) = max(0, x)$. This allows the whole network to be able to represent non-linear functions. An example is shown in Figure 2.1 where there are D-dimensional inputs, C-dimensional outputs and one hidden layer with H dimensions. n represents the index of an arbritary datapoint, v_{ij} denotes the weight which takes input dimension i to hidden dimension j, and w_{jk} denotes the weight which takes from hidden dimension j to output dimension k.

Let $\mathbf{w}$ denote the corresponding matrix of weights w_{jk} for all j, k and $\mathbf{v}_1\mathbf{v}_2, ...\mathbf{v}_H$ denote weights v_{ij} from input dimension i to all hidden dimensions j. Let $z(\mathbf{x})$ denote the result of the computation after the hidden layer, which is then fed through another linear transformation to obtain the output $\mathbf{y}$. The computation is then summarised as the following:

$$\mathbf{y} = \mathbf{w}^T z(\mathbf{x})$$
$$z(\mathbf{x}) = [Relu(\mathbf{v}_1^T\mathbf{x}), Relu(\mathbf{v}_2^T\mathbf{x}), ...Relu(\mathbf{v}_H^T\mathbf{x})]$$

It is often the case that in addition to multiplication by weights v_{ij} or w_{jk} there is a translation by adding a_{ij} or b_{jk} called a 'bias'. MLPs are practically seen as universal function approximators [20] given a non-linear activation function and a large number of hidden units. They are used in experiments in Chapter 4 and 5.

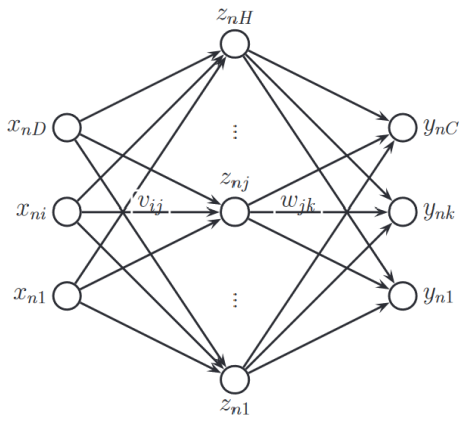


Figure 2.1: An illustration of a MLP with one hidden layer, diagram from [19].

2.1.2 Convolutions

The practical operation of a convolutional layer of a neural network is illustrated in Figure 2.2, from [21]. The kernel is moved along the 2D input from left to right and top to bottom, and the output is obtained from the dot product between the kernel and the corresponding section in the input. These convolution layers are used in two state-of-the-art (SotA) models for human activity recognition (HAR) which are tested in experiments in Chapter 4.

2.1.3 LSTM

The long short-term memory architecture (LSTM) [22] consists of h_t, the hidden state at time t, c_t the cell state at time t, and the input, forget, output and cell gates i_t, f_t, o_t, g_t at time t, respectively. Each variable is calculated as follows:

$$
\begin{aligned}
i_t &= \sigma\left(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}\right) \\
f_t &= \sigma\left(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}\right) \\
g_t &= \tanh\left(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}\right) \\
o_t &= \sigma\left(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}\right) \\
c_t &= f_t \odot c_{t-1} + i_t \odot g_t \\
h_t &= o_t \odot \tanh\left(c_t\right)
\end{aligned}
$$

where σ is the sigmoid function and $\odot$ the element-wise product. A diagram illustrating this computation is shown in Figure 2.3. Intuitively, the cell

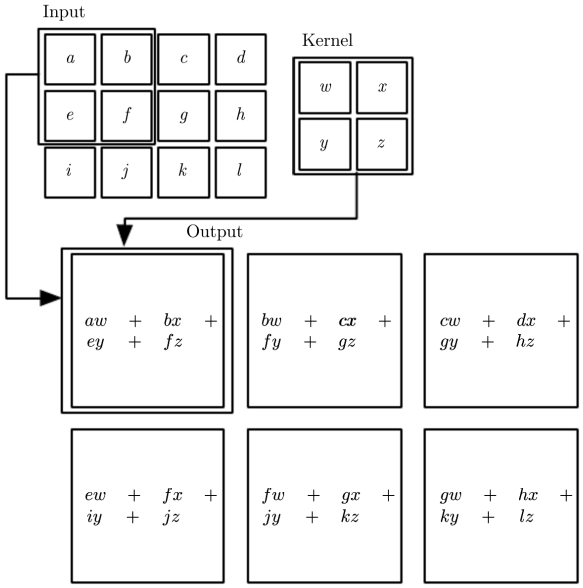


Figure 2.2: An illustration of the convolution operation, from Goodfellow et al [21].

state and hidden state keep information which can be used to process longer sequences. The forget gate allows the cell state to be reset so sequences which don't have natural breaks can be used. The input, output, and cell gates allows the cell state to be maintained over long periods of time without changing every timestep, which is helpful when processing longer sequences. The LSTM layer is used in the ConvLSTM model, a SotA model for HAR in experiments in Chapter 4.

2.1.4 Back-propagation

This section describes the principle behind back-propagation but without the implementation details. The output of a neural network is usually compared with the desired output on some metric which reflects how wrong the model's predictions are - the loss. The back-propagation algorithm calculates the changes in the parameters of a network required to reduce the loss of the output in a computationally efficient manner. It is based on the chain rule of calculus. Let $f : R \to R$ and $g : R \to R$. Let $z = f(y)$ and $y = g(x)$, the chain rule states:

$$\frac{dz}{dx} = \frac{dz}{dy}\frac{dy}{dx}$$

In its most general form, if $\mathbf{y}$ and $\mathbf{x}$ are tensors, the chain rule can be written as:

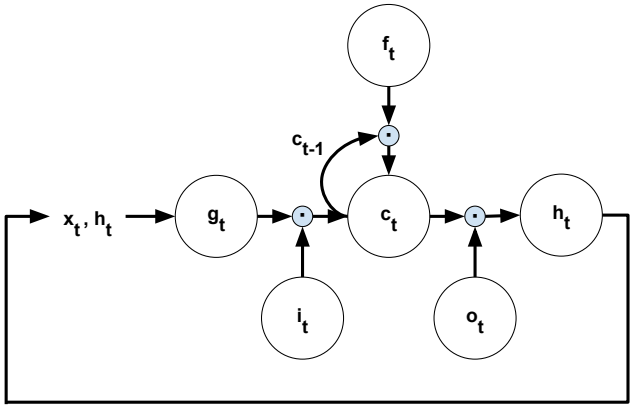


Figure 2.3: An illustration of the computation in a LSTM.

$$\nabla_{\mathbf{x}} z = \sum_j \frac{\partial z}{\partial \mathbf{y}_j} (\nabla_{\mathbf{x}} \mathbf{y}_j)$$

where j is an appropriate dimension tuple index into tensor $\mathbf{y}$ and $(\nabla_{\mathbf{x}} z)_i$ gives $\frac{\partial z}{\partial \mathbf{x}_i}$ with a tuple index i.

The general idea is as follows. The computation performed by a neural network can be represented by a computational graph. This graph consists of nodes which represent variables, and each variable is the result of an operation on one or more other variables (except the input nodes). If a variable y is obtained by performing an operation on variable x, then there is an edge from x to y in the graph. Having the computation represented this way means that the gradient of any particular node can be computed with respect to the nodes used as its input, and this can be applied iteratively from the output of the network all the way back to the input data. These gradients can then be used to adjust the weights of the network in the direction that reduces the loss.

2.1.5 No-free-lunch theorem

The no-free-lunch (NFL) theorem gives credence to the idea that there is no single model which can perform well on all tasks. It is referenced in the problem setup in Chapter 4. It is given here without proof. Let $\mathcal{X}$ be the feature space and $\mathcal{Y}$ the label space. The following discussion is restricted to the binary classification case i.e., $\mathcal{Y} = \{0, 1\}$. Denote by $S = \{(x, y)\}_{i=1}^m$ the training set,

which is sampled from $\mathcal{D}$. Denote by $h : \mathcal{X} \to \mathcal{Y}$ the model, also called the hypothesis, a function that maps from features to labels. h can be written as $A(S)$ to show that h was obtained by using learning algorithm A on training set S. Let $f : \mathcal{X} \to \mathcal{Y}$ be the 'true' labelling function i.e., $y_i = f(x_i)\ \forall i$. Define the 'true' loss of a model h, given data generating distribution $\mathcal{D}$ and labelling function f as

$$L_{\mathcal{D}}(h) = \mathbb{P}_{x,y\sim\mathcal{D}}[h(x) \neq f(x)]$$

i.e., the probability of choosing a random sample x from $\mathcal{D}$ such that the prediction is incorrect.

Theorem 1 (No Free Lunch) *Consider the binary classification setting described above. Let $m < |\frac{\mathcal{X}}{2}|$ be the training set size. For any learning algorithm A there exists a distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$ such that:*

1. *There exists a function $g : \mathcal{X} \to \mathcal{Y}$ such that $L_{\mathcal{D}}(g) = 0$*
2. *$L_{\mathcal{D}}(A(S)) \geq \frac{1}{8}$ with probability at least $\frac{1}{7}$ when choosing the training set S from $\mathcal{D}$.*

For every learner, there is a distribution $\mathcal{D}$ in which it performs poorly (statement 2), even though $\mathcal{D}$ can be learned successfully by another learner (statement 1).

2.2 Causality

This section introduces causality by discussing the two main problems tackled in the field - that of inferring causal effects and causal graphs. This is done first by examining why the randomised controlled trial allows the estimation of causal effects. This then motivates causal inference and causal discovery methods, some examples of which are discussed next. While these methods are not directly applied in the experiments in this thesis, the ideas described here are referenced throughout the thesis, in particular in Chapter 3.

2.2.1 Randomised controlled trials

Why do randomised controlled trials (RCT) work? The RCT underpins many of the important decisions we make as a society, particularly in medicine. How do we know that this particular experimental design achieves what we want? i.e., what problems does it solve, and how does it solve it? If a RCT can be performed for every treatment and outcome of interest, much of causality research would

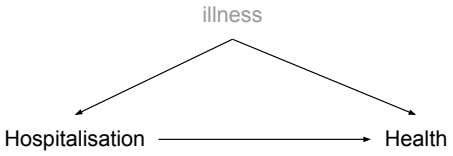


Figure 2.4: A graph depicting the scenario in our example of assessing the causal effect of hospitalisation on health. The illness variable is unobserved.

not exist. Since this is not the case, it serves as a good starting point into our discussion of causality. The following discussion is based on Angrist [23]. The one sentence summary of the answer is that RCTs solve the problem of (measured and unmeasured) confounding.

The problem of confounding is best discussed with an example. Lets say we wanted to assess whether going to a hospital improved peoples' health. Suppose we have data on several individuals in terms of whether they have been hospitalised in the last month, and the perception of their overall health on a point scale, through a survey.

The problem of confounding is formalised as follows. Let Y_i denote the observed health outcome of individual i. Let T_i denote whether individual i has been to the hospital in the past month. The observed value of Y_i is influenced by T_i as follows:

$$Y_i = \begin{cases} Y_{0i} & \text{if } T_i = 0 \\ Y_{1i} & \text{if } T_i = 1 \end{cases} \tag{2.1}$$

This setup introduces language to describe what could have happened for any particular individual i. Y_{0i} is the outcome of i if they did not go to hospital, regardless of whether they chose to go to hospital, and vice-versa with Y_{1i}. This is known as the potential outcomes framework [24].

What we are interested in is the expected difference between the potential outcomes i.e., $\mathbb{E}[Y_{1i} - Y_{0i}]$, also called the average treatment effect (ATE). But if we simply compared the average difference in health outcome grouped by hospitalisation, for each individual we only observe one of the potential outcomes conditioned on a particular treatment value T_i. This is not the same as the ATE.

$$E[Y_i|T_i = 1] - E[Y_i|T_i = 0] = E[Y_{1i}|T_i = 1] - E[Y_{0i}|T_i = 0]$$

Adding and subtracting a $\mathbb{E}[Y_{0i}|T_i = 1]$ term[1], the average difference in

[1]The substitution could have also been done using $E[Y_{1i}|T_i = 0]$ to get the average treat-

outcome by hospitalisation can be decomposed into two terms, the average treatment effect on the treated (ATT), and a bias.

$$E[Y_i|T_i=1] - E[Y_i|T_i=0] = \overbrace{E[Y_{1i}|T_i=1] - E[Y_{0i}|T_i=1]}^{ATT} + \underbrace{E[Y_{0i}|T_i=1] - E[Y_{0i}|T_i=0]}_{confounding\ bias} \tag{2.2}$$

The key property of RCTs that solve the problem of bias is the random assignment of treatment; this means the potential outcomes Y_{0i}, Y_{1i} are independent of T_i i.e.,

$$E[Y_{0i}|T_i=0] = E[Y_{0i}|T_i=1] \text{ and } E[Y_{1i}|T_i=0] = E[Y_{1i}|T_i=1]$$

If the above is true, now simply comparing the average health outcome grouped by hospitalisation gives the correct quantity of interest:

$$\begin{aligned} E[Y_i|T_i=1] - E[Y_i|T_i=0] &= E[Y_{1i}|T_i=1] - E[Y_{0i}|T_i=0] \\ &= E[Y_{1i}|T_i=1] - E[Y_{0i}|T_i=1] \\ &= E[Y_{1i} - Y_{0i}|T_i=1] \\ &= E[Y_{1i} - Y_{0i}] \end{aligned}$$

where the second and third equality uses the fact that potential outcomes are independent of T_i.

The reason why simply comparing the average difference in health grouped by hospitalisation doesn't give the correct quantity is because a hidden confounder, for instance illness, can determine both whether someone will be hospitalised and also their health. The (incomplete) causal graph for this scenario could for example resemble that of Figure 2.4, where illness (in grey) is unobserved. We say that the illness variable is a confounder. This affects the average difference in health grouped by hospitalisation through the bias term shown in 2.2. i.e., $\mathbb{E}[Y_{0i}|T_i=1] - \mathbb{E}[Y_{0i}|T_i=0]$ is the average difference in health if individual i wasn't hospitalised in the group that were hospitalised and the group that were not. Since people who are ill are more likely to be hospitalised this quantity will be non-zero and therefore affect our estimate.

ment effect on the untreated, combining these two will give the ATE.

2.2.2 Causal inference

What happens when a RCT cannot be performed? This is the main question addressed by causal inference methods. When we are interested in determining a causal effect from data that has not been collected from a RCT, it is often called an observational study. It is called so because the researchers have not *intervened* on the system, such as in a RCT. i.e., they are only observing the natural state of said system. In the previous example, it may not be ethical to randomly assign people to be hospitalised. So we only have observational data (people decide themselves to go to hospital), but may still be interested in determining whether hospitalisation improves health.

The field of causal inference is vast. It is used across much of the social sciences - economics, psychology, sociology, amongst others. I break down causal inference methods into two broad categories: one which control for confounders after-the-fact, and another called 'special cases', which allows a causal effect to be estimated without explicitly controlling for confounders.

Controlling confounders

In the previous section we discussed a example of measuring the causal effect of hospitalisation on people's health. Since the data does not come from a RCT, we cannot simply compare the healthiness of individuals who were hospitalised against those who were not to obtain the causal effect. This is because there is an unmeasured variable, illness, which caused both the health outcome of any individual and whether they are likely to be in hospital. In other words, people who are ill are more likely to go to hospital, and are less healthy, and therefore simply comparing the healthiness of the group that have been to hospital to those who have not will not reflect the causal effect of hospitalisation on health.

To obtain a valid causal effect, we need to control for (also called adjust for) the confounding variable(s), in this case illness, if the causal graph that generated the data is indeed given by Figure 2.4. Adjusting for a set of variables Z simply means that the effect of performing an intervention is calculated by comparing the outcome and treatment over the different values of $Z = z$, and combining them as a weighted average using the probability of each z, as given by the adjustment formula [25]:

$$p(Y = y \mid do(X = x)) = \sum_z p(Y = y \mid X = x, Z = z)p(Z = z)$$

Where the $do(X = x)$ notation denotes not conditional probability (which is what we observe in the dataset, and is used on the right-hand side), but rather the probability that Y takes a certain value if X were to be *set* at a particular

value x e.g through random assignment. The average causal effect is then given by:

$$\mathbb{E}[Y|do(X=1)] - \mathbb{E}[Y|do(X=0)]$$

in the case where the treatment X is binary, e.g., did go to hospital, or did not go to hospital.

The main question when controlling for confounders is which variables to include in Z. In our running example, it is intuitive that Z should include the illness variable, but in general, it is unclear. Given a causal graph, Z can be identified using a graphical criteria, e.g., the back-door criteria, the front-door criteria, or algebraically using the do-calculus [25]. The do-calculus is the most general way to find variables for adjustment, and will recover the same set of variables as the graphical techniques, and sometimes will find variables for adjustment which cannot be found using the graphical criteria. It is briefly outlined next.

Do-calculus. The following subsection builds up to the three rules of do-calculus at a high-level. In particular any proofs are omitted.

Compatibility. Given a graph $G = (V, E)$ and a distribution $P(V)$, if $P(V)$ factorises according to:

$$P(V) = \prod_{X \in V} P(X|PA_G(X))$$

where $PA_G(X)$ are the parents of X in graph G. Then $P(V)$ is said to be compatible with G.

d-separation. Given disjoint sets of vertices $X, Y, Z \subseteq V$, X is d-separated from Y given Z if every trial between any vertex in X and any vertex in Y is *blocked* in G. This is written as $(X \perp Y|Z)_G$.

A trial in G is blocked by a set Z if:

1. There is a chain $A \to B \to C$ and $B \in Z$
2. There is a fork $A \leftarrow B \to C$ and $B \in Z$
3. There is a collider$A \to B \leftarrow C$ and no descendent of B is in Z.

By using d-separation (a graphical condition) on a graph G, we can identify all the conditional independencies of a distribution that is compatible with G. d-separation is complete and sound. i.e., if $\neg(X \perp Y|Z)_G$ then there exists a distribution P compatible with G such that X is not conditionally independent to Y given Z. Additionally, if $(X \perp Y|Z)_G$ then X is conditionally independent of Y given Z in any distribution P compatible with G.

The three rules of do-calculus. Given a graph G, and disjoint sets of vertices X, Y and Z. Let $G_{\bar{X}}$ be G with all edges to X deleted. Let $G_{\underline{Z}}$ be G with all edges out of Z deleted. Let $(Y \perp Z|X, W)_G$ mean Y is d-separated from Z given X and W in graph G.

Rule 1. Insertion/deletion of observations

$$p(y|do(x), z, w) = p(y|do(x), w) \quad \text{if} \quad (Y \perp Z|X, W) \text{ in } G_{\bar{X}} \tag{2.3}$$

Rule 2. Action/observation interchange

$$p(y|do(x), do(z), w) = p(y|do(x), z, w) \quad \text{if} \quad (Y \perp Z|X, W) \text{ in } G_{\bar{X}\underline{Z}} \tag{2.4}$$

Rule 3. Insertion/deletion of actions

$$p(y|do(x), do(z), w) = p(y|do(x), w) \quad \text{if} \quad (Y \perp Z|X, W) \text{ in } G_{\overline{XZ(W)}} \tag{2.5}$$

where $Z(W) = Z \setminus An_{G_{\bar{X}}}(W)$, where $An_{G_{\bar{X}}}(W)$ are the ancestors of W in graph $G_{\bar{X}}$.

Matching. The key idea behind exact matching is that we want to find, for each treated unit, an 'identical twin' which did not receive treatment. If such a twin is available for all treated units, then the average treatment effect can be estimated by the average of the difference in the outcome for each pair of twins. This can be thought of as controlling for all variables, even though this may not be necessary, as discussed previously.

In most datasets it is infeasbile to find an exact twin for each treated unit. Instead, the closest unit which received a different treatment can be used - this is nearest neighbour matching. The distance used to measure closeness is usually the sum of the differences in each variable.

If the number of variables are high, nearest neighbour matching may still yield large distances between a pair of treated and untreated units, as the distance is calculated based on a large number of variables. This means the variables may remain imbalanced. To alleviate this, matching can instead be done on a single scalar quantity called the propensity score, which has been shown to balance variables in treated and untreated groups. This is called propensity score matching [26]. This method has been used in my work in estimating the effect of neighbourhood built environments on residents health [27]. Matching is covered in more detail in Chapter 5, where it is used to develop a data selection algorithm.

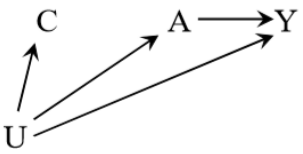


Figure 2.5: A causal graph where the difference-in-difference method may be used.

Special cases

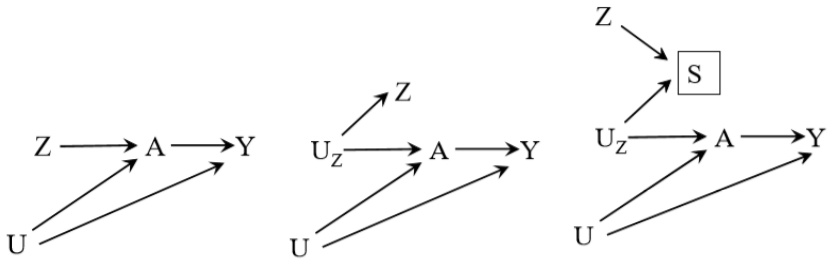
In the previous section, controlling confounders requires knowledge of the entire causal graph. In this section, I examine two special methods which do not require such knowledge. Instead, it is required that we know only certain properties of the causal graph. The first method is called difference-in-difference, and the second is instrumental variables. The following discussion is based off Hernan and Robins [28].

Difference-in-difference. Consider the causal graph given in Figure 2.5. This may represent, for instance, a scenario where we may want to estimate the causal effect of traffic policing A, on road fatalities Y, such as in DeAnglo & Hansen 2014 [29]. DeAnglo noticed that the state of Oregan failed to agree on a budget in 2003, resulting in mass layoffs in the police force, which serves as our intervention A. The outcome (road fatalities) after the layoffs is denoted Y. Y is further separated to Y_0, fatalities if layoffs did not happen, which is unobserved, and Y_1, fatalities if layoffs did happen, which is observed. The pre-intervention outcome (road fatalities under 'normal' conditions, before layoffs) is denoted C. There may be unmeasured confounding variables which also affect C, A and Y - denoted by U. U could be for instance, the number of people speeding.

We know that A has no effect on C, as A comes after C in time, by definition. However, $\mathbb{E}[C|A=1] - \mathbb{E}[C|A=0]$ is not 0 because of confounding by U (A and C have a common cause). This quantity actually measures the confounding effect of A on C by U, and if this confounding effect by U is the same as that of A on Y i.e.,

$$\mathbb{E}[Y_0|A=1] - \mathbb{E}[Y_0|A=0] = \mathbb{E}[C|A=1] - \mathbb{E}[C|A=0]$$

then we can find the ATT (average treatment effect on the treated):



(a) (b) (c)

Figure 2.6: Several causal graphs where Z is a valid instrument. U_z is unobserved. (a) is a causal instrument whilst (b) and (c) are proxy instruments.

$$\begin{aligned}\mathbb{E}[Y_1 - Y_0|A=1] &= (\mathbb{E}[Y_1|A=1] - \mathbb{E}[Y_0|A=0]) - (\mathbb{E}[C|A=1] - \mathbb{E}[C|A=0]) \\ &= \mathbb{E}[Y_1|A=1] - \mathbb{E}[Y_0|A=0]) - (\mathbb{E}[Y_0|A=1] - \mathbb{E}[Y_0|A=0]) \\ &= \mathbb{E}[Y_1|A=1] - \mathbb{E}[Y_0|A=1]\end{aligned}$$

which we can see why is called difference-in-difference. Intuitively, it means if the unmeasured confounding factors that effect road fatalities affect fatalities in the same way before and after an intervention (the budget incidence) then we can adjust for those by taking the difference in the difference after intervention and the difference before intervention. We do not need the knowledge of the full causal graph, in particular how the variables U affect each other. We only need to know that U causes C, A and Y and that A causes Y.

Instrumental variables. The second method is based on variables called the instrument. Consider the causal graphs in figure 2.6. The variable Z is called the instrument. A denotes the intervention, Y the outcome, and U unmeasured confounding variables.

For a variable to be an instrument, it has to satisfy four requirements, which can all be expressed in terms of graphical restrictions except the last one.

1. Z is not independent of A. Note that Z does not have to cause A, as we will see later. This is known as 'Relevance'.

2. Z does not cause Y in any way except through A. This is known as exclusion restriction.

3. Z does not share causes with Y. Also known as marginal exchangeability.

4. The effect of the intervention on the outcome is the same for all individuals. This is known as homogeneity.

The variable Z satisfies all graphical requirements in Figures 2.6a to 2.6c.

Instruments can be *causal* or a *proxy*. Causal instruments are the simplest ones and are variables that directly influence the treatment, such as in Figure 2.6a. Proxy instruments are those that have associations with the treatment but does not directly influence it, for instance when it shares a common cause with the treatment such as Figure 2.6b or when it influences a common cause of the treatment (S) that has been conditioned on, as in 2.6c.

As an example of an instrumental variable, let A be the amount of smoking by an individual, Y be the incidence of lung cancer, and Z be the tax on cigarettes. U are unmeasured variables which may affect both how much an individual smokes, and the risk of lung cancer. This may be for instance genetic factors or family history. The situation can be plausibly represented by the causal graph in Figure 2.6a and makes Z a valid instrument. In particular, tax on cigarettes do affect the amount of smoking engaged by individuals (relevance, condition 1), tax on cigarettes do not affect lung cancer in any other way except through its effect on the amount of smoking (exclusion restriction, condition 2), tax on cigarettes do not share any causes with Y (marginal exchangeability, condition 3), and it is plausible that the effect of smoking on the risk of lung cancer is the same for all individuals (homogeneity, condition 4).

If we assume the requirements are met, then the ATE in the binary treatment case, $\mathbb{E}[Y_1 - Y_0]$ is equal to:

$$\frac{\mathbb{E}[Y|Z=1] - \mathbb{E}[Y|Z=0]}{\mathbb{E}[A|Z=1] - \mathbb{E}[A|Z=0]} \tag{2.6}$$

or for continuous instruments

$$\frac{\text{Cov}(Y,Z)}{\text{Cov}(A,Z)} \tag{2.7}$$

where $Cov(X,Y) = \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])]$, for any random variables X and Y. Note that in the methods covered in this section, it is always the case that we need to know at least some structure of the causal graph. In many applications this is not the case - what can be done? In the next section, we cover another major subfield of causality - determining the graph when it is unknown. This is the realms of causal discovery.

2.2.3 Causal discovery

What can we do if we do not know the causal graph? In some cases, discovering the causal graph may be the ultimate goal - for instance in gene regulatory networks. In others, we may be interested in finding the graph as a first step to making causal inferences, for example using some of the methods previously

described.

In the subsequent discussion, we assume that the causal graphs considered are acyclic i.e., the graphs are directed acyclic graphs. We will also start by considering the case where there are no unmeasured variables. This is also known as the causal sufficiency assumption. This section will build up to a high-level description of the PC (stands for Peter Spirtes and Clark Glymour, the inventors) algorithm [30].

The key to finding graphical structure from data in the PC algorithm is through conditional independence tests. If there is a d-separation in the graph, then this places a conditional independence constraint on the distribution that is compatible with that graph. We can then work backwards: using conditional independencies in the data, we can narrow down the shape of the graph by inferring d-separation. However, to do this we need an additional assumption - the faithfulness assumption. Intuitively, this means that all conditional independence constraints follow from the graphical structure. To demonstrate faithfulness, it is helpful to consider when this is not the case. There are two ways faithfulness can be violated: 'balancing out' and from deterministic causal mechanisms. The following two examples demonstrate these violations.

Violation by determinism. Consider the causal graph in Figure 2.7. Let

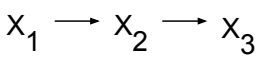


Figure 2.7: A causal graph which could violate the faithfulness assumption by determinism.

the mechanism for X_3 be some deterministic function of X_2 e.g., $X_3 := 2X_2$. We now have that X_1 is conditionally independent of X_2 given X_3 but X_1 is not d-separated from X_2 given X_3 i.e., $X_1 \perp X_2 | X_3$ but $X_1 \not\perp_d X_2 | X_3$.

Violation by balancing out. Consider the causal graph in Figure 2.8

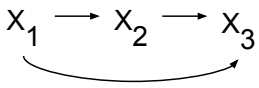


Figure 2.8: A causal graph which could violate the faithfulness assumption by balancing.

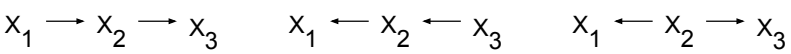
Let the mechanism be defined as follows:

$$
\begin{aligned}
X_1 &:= \epsilon_1 \\
X_2 &:= \alpha X_1 + \epsilon_2 \\
X_3 &:= \beta X_2 - \alpha\beta X_1 + \epsilon_3
\end{aligned}
$$

where $\epsilon_1, \epsilon_2, \epsilon_3 \sim N(0, 1)$. In this case, X_1 is independent of X_3 but they are not d-separated.

So far we have discussed acyclicity, faithfulness and causal sufficiency. However another question remains. If we were to test conditional independencies and work backwards, do the same set of conditional independencies always correspond to a unique causal graph?

This is unfortunately not the case, and two graphs of different structure which have the same set of conditional independencies are called **Markov equivalent**. For instance the three graphs in Figure 2.9 all imply $X_1 \perp X_3 | X_2$.



(a) (b) (c)

Figure 2.9: Markov equivalent graphs of three variables.

A group of Markov equivalent graphs can be represented by a single graph called a completed partial directed acyclic graph (CPDAG). A CPDAG has a directed edge if all graphs in the equivalence class have the same directed edge, otherwise the edge is undirected. As an example the CPDAG representing the three graphs in Figure 2.9 is shown below in Figure 2.10a. The other CPDAG representing the second and final Markov equivalent graphs in the case of three variables is show in Figure 2.10b.

(a) (b)

Figure 2.10: All CPDAGs for a graph with three variables.

The main ideas behind the PC algorithm is as follows. Start with the fully connected graph. Test marginal independencies to remove edges. Then test conditional independencies to remove edges. Then use colliders to orient the

edge, or the fact that we assume acyclic graphs. The remaining unoriented edges remain i.e., we can only discover the graph up to Markov equivalence. The number of conditional independence tests is exponential to the number of variables. This also assumes an appropriate test for conditional independence is available, which is a area of research in itself, and out of scope for our discussion.

The PC algorithm can be extended to work for cases with unmeasured confounding variables (removing the assumption of causal sufficiency), but the problem of Markov equivalent graphs remain. See for example the Fast Causal Inference (FCI) algorithm [31]. There are other types of methods, such as score-based ones, for instance Greedy equivalence search (GES). However they can only also identify graphs up to Markov equivalence i.e., they return CPDAGs.

The main takeaway point is that only graphs up to an equivalence class is able to be identified (with no additional parametric assumptions on the mechanisms), and that computation scales exponentially with the number of variables. As we will discuss in Chapter 3, this is unfortunate as knowing the causal graph can be helpful in training robust models.

2.3 Robustness

Robustness is an emerging field of research. To the best of my knowledge, there is no established consensus on the definition of 'robustness', or what it entails. Instead, some specific form of model failure is observed, and then a problem is formulated to fix said failure, which then leads to a specific definition. Some model failures do not yet have a theory to explain them.

In this section we will examine several model failures, and then two definitions of robustness - robustness to small input perturbations and distributional robustness. The former came from the study of adversarial examples - a particular form of model failure. Finally we turn to look at work related to robustness which use ideas from causality. In particular, the task of domain generalisation, and invariant risk minimisation.

2.3.1 Model failures and the many types of robustness

In the introduction, we have seen models misclassify images based on novel placements of known objects [7] and small perturbations to the input [5]. Model performance also degrades with variations in light [32], or weather [33]. We have also seen models misdiagnose diseases based on images modified with patches of data, both natural and adversarially constructed [8]. Models trained on data from one hospital performs poorly on data from a different hospital [34]. Models also fail under 'object-transplanting', where objects are placed post-hoc

within images, which then not only causes the model to fail when classifying the transplanted object, but existing objects previously correctly classified in the image become misclassified after transplant [35]. Two datasets, imagenet-O and imagenet-A, provides natural examples (meaning images not manipulated by the researcher, but rather are natural, real images) that reliably cause models to misclassify [36]. A new test set of CIFAR-10 images were constructed to test the ability of SotA models trained using the original CIFAR-10 dataset and found a significant drop in performance [37]. It is shown that the new test set has no significant change in distribution. The same is shown for the ImageNet dataset [38]. How can we understand these various different modes of model failure?

There is currently two distinct approaches to robustness which has an established field of research, and several other loosely related hypotheses. We will cover these in turn, starting from robustness to small perturbations.

Robustness against small input perturbations. This strand of robustness research came out of the observation that models fail when faced with adversarial examples. One of the first works which looked at adversarial examples (called evasion attacks in this particular work) was in security applications, e.g., evading spam or malware detectors [39]. The general idea is to minimise the classifiers discriminant function, or an estimate thereof ($\hat{g}(x)$), such that the model misclassifies the adversarial sample with high confidence; subject to some constraint on how far (for some distance function d) the adversarial sample ($x*$) is from a real sample (x_0) based on some maximum distance d_{max}.

$$\begin{aligned} \mathbf{x}^* = \arg\min_{\mathbf{x}} \; & \hat{g}(\mathbf{x}) \\ \text{s.t. } & d\left(\mathbf{x}, \mathbf{x}^0\right) \leq d_{\max} \end{aligned}$$

They demonstrate their approach by producing adversarial samples for MNIST [40] (a dataset of handwritten digits) classification, and for malware detection in PDF files.

The first work to construct adversarial examples for large image datasets (QuocNet, AlexNet) is by Szegedy et al. [41]. They pose the question of how we can reconcile the fact that the network seems to generalize well, whilst being susceptible to adversarial examples which are indistinguishable from real examples.

At a high level, the literature on adversarial robustness is based on solving the problem of finding parameters θ that minimises the empirical adversarial risk, defined as the worst case loss around some region around an input sample,

given by:

$$\min_{\theta \in \Theta} \left\{ \mathbb{E}_{(x,y)\sim D}[\max_{\delta \in \Delta(x)} \ell(h_\theta(x+\delta), y)] \right\} \tag{2.8}$$

where x, y are the input and label respectively, D is the distribution the dataset is drawn from, l some appropriate loss function, h_θ our predictive model with parameters θ, and $\Delta(x)$ is some perturbation region around an input x. According to Madry and Kolter, papers in adversarial robustness essentially propose different ways to solve either the inner maximization problem, or the outer minimization problem [4] in Equation 2.8.

Robustness to a class of distributions. Another strand of robustness research is to train a model which performs well on a class of several distributions - called distributionally robust optimisation. The class is often defined as some small region around the original test distribution, where this problem reduces to one similar to adversarial robustness. The formal problem of distributional robust optimisation is defined as minimising the worst case expected loss over a set of distributions $\mathcal{Q}$ [42].

$$\min_{\theta \in \Theta} \left\{ \sup_{Q \in \mathcal{Q}} \mathbb{E}_{(x,y)\sim Q}[\ell(h_\theta(x), y)] \right\} \tag{2.9}$$

Other works on robustness. There are other works which do not directly fall under the field of adversarial robustness, or distributional robustness. Some of these are reviewed next.

Hendrycks et al [43] studied seven robustness hypotheses, primarily on image data, and conclude that 1. there is no general method that consistently improves robustness, and 2. robustness is 'multivariate' i.e., not as simple as a single scalar quantity. Yin et al. [44] shows concrete evidence that models latch on to high-frequency and low-frequency correlations that are predictive of the label in image models. They also show that different types of training that tries to improve 'robustness' can improve certain kinds of robustness whilst being detrimental to other types of robustness (improve robustness to high/mid frequencies at the expense of low frequencies and vice-versa). Experiments in [45] show that in the case of CNN architectures on image data, models rely on 'surface statistical regularities' i.e., high-dimensional spurious correlations, and is the reason why despite performing well, they are susceptible to adversarial examples. Ilyas et al. [46] showed using CIFAR-10 and ImageNet data that adversarial examples are due to models using spurious features to classify the image.

These works seem to point to models using spurious correlations as features as the reason for its various failures. This seems to be a problem which can be tackled using ideas from causality, and indeed this is an emerging area of

research.

2.3.2 Causality and robustness

If spurious features are the reason for non-robustness then we would like our models to use causal features i.e., the features which cause the label to take any given value. The main challenge is how to learn these causal features; this is the problem the causal approach to robustness is trying to answer.

There are various methods which are inspired by ideas in causality, and there are two broad approaches. One approach is to learn disentangled representations, where in addition to the usual setup, a causal constraint is placed on the latent representation [47, 48]. These do not have robustness as an end goal, but learning a disentangled representation can be the first step to training a robust model. This is similar to first performing causal discovery, as we discussed in Section 2.2.3. However, the main problem of identifiability remains, and this is only explicitly considered in work by Khemakhem et al [49], which do not consider the problem of robust prediction. These methods allow causal discovery to be performed on complex high-dimensional data such as images, but face the same fundamental challenges as outlined in Section 2.2.3.

Another approach, which is the approach adopted in this thesis, is to leverage data from different environments, where the main ideas were proposed by Peters et al [18] in a statistical setting, and then made more widely applicable to neural network models by Arjovsky et al [50]. This main idea is discussed in more detail in Chapter 3, where we derive the empirical causal convergence result. There has been multiple derivative works which build on this main idea, such as anchor regression [51] and its extension [52], however their proposals are currently limited to linear settings. Other derivative work make different assumptions, such as assuming concept shift ($P(Y|X)$ changes) and that variations in training distributions are similar to variations at test time [53]. As evidence to the significance of the idea behind IRM, there are multiple works which try to better understand and refine it [54, 55, 56].

Instead of proposing any particular method for learning a robust model such as IRM or its variants, the work in Chapter 3 instead tries to establish a theoretical view of whether using only the information from different data environments can help obtain a robust predictive model, where robust is defined using a causal interpretation. The remaining chapters explore the consequences of this causal interpretation of robustness in the application of Human Activity Recognition (HAR), and then by exploring a method to achieve robustness which potentially uses less data.

In the remainder of this section, the optimisation performed by IRM is briefly

reviewed, as well as the intuition behind spurious features. We also talk about domain generalization, which aims to solve the same robustness problem as the one presented in this thesis, but without connection to causality.

Invariant Risk Minimisation (IRM). One of the first approaches to getting a neural network model to use causal (also called invariant) features is invariant risk minimisation [50]. The main idea behind IRM is that instead of merging all training data and discarding information related to which groups each datapoint belongs to, this information could help models learn invariant features. This idea is turned into an optimization goal by finding a data representation ϕ and classifier w that minimises invariant risk:

$$\begin{aligned} &\min_{\substack{\Phi:\mathcal{X}\to\mathcal{H} \\ w:\mathcal{H}\to\mathcal{Y}}} \sum_{e\in\mathcal{E}_{\text{tr}}} R^e(w\circ\Phi) \\ &\text{subject to} \quad w \in \arg\min_{\bar{w}:\mathcal{H}\to\mathcal{Y}} R^e(\bar{w}\circ\Phi), \text{ for all } e\in\mathcal{E}_{\text{tr}} \end{aligned}$$

This is reduced to a more practical version which doesn't require a bi-level optimization:

$$\min_{\Phi:\mathcal{X}\to\mathcal{Y}} \sum_{e\in\mathcal{E}_{\text{tr}}} R^e(\Phi) + \lambda \cdot \left\|\nabla_{w|w=1.0} R^e(w\cdot\Phi)\right\|^2$$

i.e., we are looking for a data representation ϕ such that the dummy classifier w minimises the invariant risk across each training environment, plus a gradient norm penalty, controlled by the hyperparameter λ.

Cows versus camels. The cows versus camels scenario is an example used to illustrate how models can learn spurious features to predict the label that can be easily understood by humans. It describes a setting where a classifier is trained to distinguish between cows and camels. The images of cows in the training set all have a green grassy background, and the images of camels were on a brown sandy background. The classifier, having successfully trained on the training set, would mistakenly classify cows and camels in any other background. The conclusion is that it had learnt to use the background, as opposed to the animal, to classify the label. In the original IRM paper this setting was tested by using MNIST digits with a background colour correlated to the label in training, and reversing the correlation in the testing set.

Domain generalization. In light of the problem with spurious correlations described, Gulrajani et al. [57] constructed a benchmark for a task called domain generalization. This task aims to capture the ability of a model to learn invariant features, also sometimes called generalising to new domains. The benchmark consists of multiple image datasets where each dataset may contain the same

object but in a different 'domain'. Whilst they offer no formal definition of what constitutes a 'domain', in practical settings it is relatively clear. A domain captures the same phenomenon of interest under a different environment. For instance, an image classifier should be able to classify an animal whether it is an image, painting, drawing or a sketch. This benchmark accommodates the task-centric view of machine-learning, and they provide a table summarising the differences between the domain generalisation task and other machine learning tasks, shown in Figure 2.11.

The task of domain generalization is mentioned for completeness, but methods which are used to tackle the task does not have to be connected to causality.

Setup	**Training inputs**	**Test inputs**
Generative learning	U^1	$\emptyset$
Unsupervised learning	U^1	U^1
Supervised learning	L^1	U^1
Semi-supervised learning	L^1, U^1	U^1
Multitask learning	$L^1, \ldots, L^{d_{\text{tr}}}$	$U^1, \ldots, U^{d_{\text{tr}}}$
Continual (or lifelong) learning	$L^1, \ldots, L^\infty$	$U^1, \ldots, U^\infty$
Domain adaptation	$L^1, \ldots, L^{d_{\text{tr}}}, U^{d_{\text{tr}}+1}$	$U^{d_{\text{tr}}+1}$
Transfer learning	$U^1, \ldots, U^{d_{\text{tr}}}, L^{d_{\text{tr}}+1}$	$U^{d_{\text{tr}}+1}$
Domain generalization	$L^1, \ldots, L^{d_{\text{tr}}}$	$U^{d_{\text{tr}}+1}$

Figure 2.11: The differences between domain generalization and other machine learning tasks. L^d and U^d denotes labelled and unlabelled data from domain d respectively.

In this section we have now seen various types of model failures, from small input perturbations, novel object placements, object transplanting, variations in light, weather and hospitals, amongst others. We see that there are two established fields of robustness research, robustness against small input perturbations, and distributional robustness, which tackle some but not all of the various model failures described. We also see various other theories of robustness proposed, many of which point to models using spurious features as the main culprit behind its failure. Is there a unifying way to view these robustness issues? As will be explained in further detail in the next chapter, causality can be used to unify these issues under one common formalisation.

If model failures arise from not using the causal features which determine the label, the question is then of course, how can the causal features be learned? We have seen that invariant risk minimisation (IRM) proposes a modified loss and training setup to encourage learning invariant features, and the task of domain generalisation. As will be discussed in Chapter 3, we will argue that there is no straight-forward method to learn causal features, especially in high-dimensional

data, but optimising only for prediction performance across many domains even in non iid settings is a valid approach towards a causal model.

Chapter 3

Causality and prediction robustness

We have seen various model failures and the types of robustness studied in literature, which can't yet be used to explain all of the observed failures. Many empirical observations seem to suggest that models using spurious features correlated with the label are the main culprit behind many observed failures. Can we better understand these failures, and hence robustness, by connecting robustness to ideas from causality? Additionally, given our discussion on causal inference and causal discovery methods in Chapter 2, can these methods be used to train more robust models? These are the motivating questions for the discussion in this chapter. The answer is not completely straightforward; current inference and discovery methods are not practical, but there is hope by using multiple data environments. The discussion in this chapter forms the 'causal view' of robustness that is referred to throughout this thesis.

Outline. The chapter proceeds as follows. We cover some related work and the motivation for connecting causality to robustness. We then introduce structural causal models (SCM), entailed distributions and interventions. We then discuss how robustness can be formulated in terms of SCMs, and how knowing the causal graph could help design more robust models. We discuss how existing causal discovery and causal inference methods come with challenges and is not very helpful in model robustness. We then cover Peters' [18] invariant causal prediction work, which first introduced the idea that the causal parents can be discovered using data from multiple interventions. We then cover Probably Approximately Correct (PAC) learning [58], which outlines why a model trained on a training set should also perform well on a separate test set, assuming both datasets are iid. Finally, the empirical causal convergence idea is presented,

which concludes the chapter.

3.1 Introduction

We have seen examples of several failures by deep neural networks in Chapter 2. This ranged from variations in object style, poses, lighting, weather to hospitals [34, 6, 7, 33, 32]. We have also seen various empirical observations regarding the robustness of said models. Some existing work points to models using features which are spuriously correlated with the label for prediction as a possible explanation for its non-robustness [44, 46, 45].

Intuitively speaking, using the cause of a target variable should avoid the issue of models using spuriously correlated features, and hence the issue with non-robustness. However, it is still unclear what the connection between the ideas in causality and that of robustness are. For instance, can the literature in causal inference and causal discovery, some of which are covered previously, be used to train more robust models? How can robustness be formulated using causality?

We have also additionally seen two fields of research which studies issues related to model robustness - that of adversarial and distributional robustness. If we were to define robustness using causality, how does this fit in with these fields of research? These questions motivate the work in this chapter.

We start by discussing how structural causal models (SCM), which is assumed to generate the observed data, is related to robustness. Whilst SCMs have been studied for some time, its connection to model robustness have not explicitly been discussed in literature. This is achieved by using the idea of intervention distributions, defined in section 3.3. This has the advantage of addressing some deficiencies in existing formulations of robustness. This is called the 'causal view' of robustness, which is referred to throughout this work. We then explore some general implications of this causal view.

In particular, this means that the relationship between different distributions of non-iid data of the same task is precisely defined. This provides a foundation to decide which variables should be used to construct a robust model - if we knew the causal graph. In practical settings, this is not the case. We then take a look back at our discussion of causal inference and discovery algorithms in Chapter 2 - can these be helpful? The short answer is no, if we don't assume any knowledge about the causal graph. We then argue that the most practical approach to find a robust model is to exploit data from different intervention distributions. This idea was first introduced by Peters [18], who showed that given data from different intervention distributions, the causal parents can be identified using appropriate hypothesis tests. However this is impractical with

a large number of variables due to the computational costs of these tests.

Instead, we propose the idea of empirical causal convergence (ECC); a model will become more robust if it achieves good prediction on a diverse range of intervention distributions of a common SCM. This idea has some interesting implications. Previously, the probably approximately correct (PAC) model [58] of learning provided justification to why optimising for prediction accuracy on the training set (minimising empirical risk) should give models that perform well on a test set it has never seen, given that training and testing data are iid. The proposed ECC idea provides a reason for optimising for prediction accuracy when training and test sets are not iid.

In summary, this chapter contributes the following:

1. a formulation of robustness in terms of the data-generating SCM
2. we investigate whether existing work in causality can be helpful in training robust models. It turns out they are not immediately applicable.
3. We argue that the most practical way to train a robust model, given current causal inference and causal discovery methods, is to use data from different environments. This was first proposed by Peters [18] but their method is not more widely applicable due to the hypothesis tests involved.
4. We propose the idea of empirical causal convergence, which connects the loss (risk) of a model to the causal mechanism in the SCM. This provides a reason to optimise for a model's loss in non-iid settings.

3.2 Related work

Consider equation 2.8 which states the general objective for adversarial robustness. This objective states that we are interested in optimising the worst case loss of a model around a region of an input sample x, repeated below for convenience.

$$\min_{\theta \in \Theta} \left\{ \mathbb{E}_{(x,y)\sim D}[\max_{\delta \in \Delta(x)} \ell(h_\theta(x+\delta), y)] \right\}$$

The formulation of the objective comes directly from considering adversarial examples, which are small perturbations to the input which causes the model to misclassify but does not change the input in a distinguishable way, such as that shown in Figure 1.1. An adversarially robust model can be thought of as one which can minimise this objective for some defined region Δx around the input. However, this formulation of robustness cannot capture other types of model failures. For instance those from novel placements of known objects,

or different styles of the same object, previously shown in Figure 1.3 and Figure 1.2 respectively. This stems from the difficulty of defining some region of perturbation Δx that would capture these different instances (placement, style).

Now lets consider equation 2.9, which states the objective for distributional robustness, also repeated below for convenience. This states that we are interested in optimising the worse case expected loss from a set of distributions $\mathcal{Q}$.

$$\min_{\theta \in \Theta} \left\{ \sup_{Q \in \mathcal{Q}} \mathbb{E}_{(x,y)\sim Q}[\ell(h_\theta(x), y)] \right\}$$

This formulation of robustness is more general than the adversarial formulation, and by defining the set $\mathcal{Q}$ appropriately, this objective can be used to train models robust to small input perturbations, similar to optimising for adversarial robustness above [59]. In theory this formulation of robustness should be able to capture the types of model failures discussed previously, provided we can define an appropriate set of distributions $\mathcal{Q}$. However it remains unclear how $\mathcal{Q}$ should be selected in general.

The empirical observations in literature regarding robustness seem to point to spurious features as the main culprit behind many model failures. The coloured MNIST experiment in Arjovsky et al. [50] showed that models will use the image background that is highly correlated with the label for prediction. Yin et al. and Jo et al. [44, 45] showed that image models use high-frequency and low-frequency features which correlate with the label for prediction. Ilyas et al. [46] showed on two popular image datasets that adversarial examples are caused by models using spurious features. It is then natural to question whether using causal features will mitigate these robustness issues. However the connection between causality and prediction robustness is unclear. As we will see in the coming discussion, the difficulty comes with identifying these causal features.

Domain generalisation [57] proposes an empirical vision benchmark motivated by models' inability to generalise out-of-distribution. While they refer to various model failures in their motivation, the connection to causality remains unclear. There is an additional difficulty of the precise definition of a 'domain'. As we will later see, formulating robustness with SCMs mean that this definition is well defined. IRM [50] states their motivation as learning stable or causal features, as opposed to spurious correlations, and proposes a loss function which promotes these features. However, the question of how this relates to SCMs, the main formalisation in causality, still remains. There is also the overarching question of how the literature in causal inference and causal discovery methods, seen previously, fits into robust prediction models. Are they of any use? If not, why not?

Given the empirical observations with regards to spurious features, the challenges that come with the adversarial and distributional robustness formulation, and the large existing literature on causal inference and discovery methods, it seems appropriate to question what, if any, is the connection between robustness and causality? The SCM construct makes this connection clear, and is the topic of the next section.

3.3 Structural causal models and robustness

This section introduces the structural causal model (SCM), which is used in causality to represent the mechanism which generated the observed data. It forms the basis of the causal view of robustness. This particular formalisation of causality is credited to Pearl [25]. It is not the only formalisation; there is also the potential outcomes framework [24], but it is not relevant to our discussion here.

3.3.1 Structural causal models

In modelling data, discussions are usually centred around a data distribution. Central to causality is the idea of a fixed data generating mechanism. To understand this, it is useful to look at the problems in areas where causal questions are often asked. These questions span a wide range of fields from economics, biology to psychology, and beyond. For instance, we might be interested in the causal effect of class size to final test scores, or we may be interested in the causal effect of smoking on incidence of lung cancer, or perhaps the causal effect of an advertising campaign to sales. In all of these instances, we assume there is some underlying way in which the cause (class size, smoking, advert) affects the outcome of interest (test score, risk of lung cancer, sales) which remains the same across the different units of the population (in all three examples, this is people) that we observe. As an example, the data generating mechanism for how smoking affects lung cancer may look something like Figure 3.1.

This idea of a data generating mechanism is formalised by the structural causal model (SCM)[1]. A SCM is a purely mathematical definition of how variables obtain their value. One can think of it as a generative model for the dataset.

Definition 1 *(Structural Causal Model). A SCM* $\mathbb{S} = (\mathbf{X}, \mathbf{f}, \mathbf{N})$ *is a three tuple consisting of a set of p variables* $\mathbf{X}$*, a set of p structural assignments* $\mathbf{f}$*, and a set of p exogenous noise variables* $\mathbf{N}$*. One of the variables is designated a*

[1] In some literature it may have a different name, e.g a causal Bayesian network, I will use SCM in this thesis.

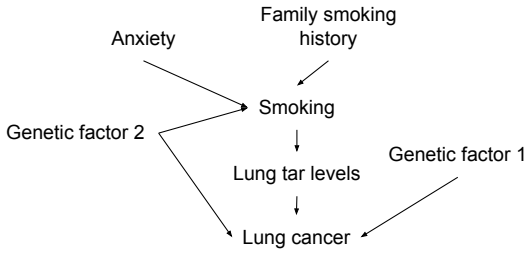


Figure 3.1: An illustration of what the data generating mechanism for how smoking affects lung cancer may look like.

'target' X_t. Each variable X_i has a value which is determined by its structural assignment function f_i, other variables, and its noise variable N_i:

$$X_i := f_i(PA(X_i), N_i) \quad i = 1, 2, ..., p \tag{3.1}$$

Where $PA(X_i)$ are the variables which influence the value of X_i, also called the parents of Xi. $N_1, N_2, ...N_p$ are jointly independent, i.e., The joint distribution of all noise variables $P_{\mathbf{N}}$ is a product distribution.

A SCM has a corresponding graph representation, by drawing a directed edge from variables which influence the value of X_i, to X_i. i.e., from $PA(X_i)$ to X_i. The SCMs considered in this thesis are assumed to correspond to directed acyclic graphs (DAGs) unless otherwise specified.

Example. A SCM with two variables, X_1 and X_t is defined as follows:

$$\begin{aligned} X_1 &:= Normal(400, 50) \\ X_t &:= 0.0005X_1 + Normal(15, 1) \end{aligned} \tag{3.2}$$

X_1 could be the altitude (m) and X_t the temperature (C). Here N_1 is Normal(400,50) and N_t is Normal(15,1). The corresponding graph is shown below in Figure 3.2.

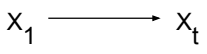


Figure 3.2: An example SCM

A SCM in its 'natural' state gives rise to a joint distribution over the variables considered.

Proposition 1 *(Observational distribution) A SCM $\mathbb{S}$ entails a distribution*

over the variables $\mathbf{X}$*, called the observational distribution.*

Since the SCM considered are acyclic, we can re-write each variable X_i as a function of its ancestral noise terms. For instance, We could generate a sample from the joint distribution of the two variable example above by first sampling X_1 from N_1 and then calculating X_t after sampling from N_t.

One of the main features of SCMs is that it allows interventions on the system to be represented.

Definition 2 *(Intervention). An intervention* I *on SCM* $\mathbb{S}$ *is a set of structural assignments* $\{\tilde{f}_j, \tilde{f}_{j+1}, \tilde{f}_{j+2}, ..., \tilde{f}_k\}$ *where* $\{j, j+1, j+2, ...k\} \subseteq \{1, 2, 3, ..., p\}$ *which replace the original structural assignments in* $\mathbb{S}$*:*

$$X_j := \tilde{f}_j(\widetilde{PA}(X_j), \tilde{N}_j))$$

replacing the original parents with $\widetilde{PA}(X_j)$ *and noise terms with* $\tilde{N}_j$*.*

This means we are able to change the way any variable X_j is determined. In the example SCM in Figure 3.2, temperature varies naturally as altitude is changed, for instance when moving up or down a mountain. An intervention on the temperature variable X_t may represent an event such as starting a fire, where the mechanism for determining temperature has changed.

Interventions can be broken down into different types, namely *do*-interventions[2] where the structural assignment is set to a particular fixed point, e.g., set X_t to 20. Or *soft* interventions, which encompasses more general changes to the mechanism. For instance where the intervened variable X_i may still be influenced by its parents, but in a different way, or only the noise distribution N_i has changed. For instance set $X_t := 0.0008X_1 + Normal(30, 1)$.

After an intervention, the SCM entails a different distribution - the intervention distribution.

Definition 3 *(Intervention Distribution). Consider a SCM* $\mathbb{S} = (\mathbf{X}, \mathbf{f}, \mathbf{N})$ *and its entailed distribution* $P_{\mathbf{X}}$*. If there has been an intervention* I *on* $\mathbb{S}$*, resulting in a SCM* $\mathbb{S}_I$ *and corresponding distribution* $P_{\mathbf{X}_I}$*.* $P_{\mathbf{X}_I}$ *is called an intervention distribution of* $\mathbb{S}$ *induced by intervention* I*.*

After an intervention I on an SCM S, a sample from the intervention distribution could be generated by following the same procedure described previously for the observational distribution.

The SCM, and interventions on SCMs, provide the language to be able to describe an underlying data generating mechanism, and changes to it. It has the

[2] also called atomic or surgical interventions

added benefit of being connected to literature on causal inference and discovery. Whether this literature is useful, and how robustness can be formulated in terms of SCMs is discussed next.

3.3.2 Finding a robust model

SCMs allow us to define the vague idea of a 'domain' of the same task. Some examples of domains we have discussed are style, object orientation, or hospitals. Each 'domain' of a task can be viewed as a particular intervention on a common SCM. Since an intervention on a SCM means it entails a different distribution, this is consistent with the fact that data from different domains have different distributions.

We can also now define what a robust model is. A robust model is one which performs well across all of the possible intervention distributions of a SCM representing the task. The question then becomes what type of interventions do we expect to see on the SCM, and this will differ according to the specific application. For example, consider the fictional SCM in Figure 3.3 relating the variables work hours, free time and time spent on exercise.

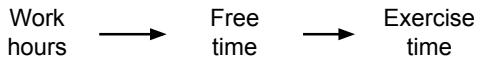


Figure 3.3: A fictional SCM on three variables

If the task was to predict the amount of free time a particular person has, an intuitive thing to do would be to use the causal variable, work hours. Specifically, the mechanism $f_{freetime}$ that is a function of work hours would make the ideal model. But if we do not expect any interventions on the exercise time variable, we should also use this downstream variable from the target for prediction. However, one can imagine different populations of people who have certain interventions on their exercise time. Consider a population of health-conscious people who may follow a strict exercise regime, or perhaps the population of professional athletes; the mechanism which determines their exercise time variable is different to a group of randomly selected people from a population. This means that a model which uses the exercise time variable in predicting free time is not robust to these different populations, as opposed to a model which only uses the variable work hours.

In general, the parent of a target variable is a robust predictor assuming that we will not see data that comes from an intervention on the target variable. If we had additional information about whether certain interventions are expected on each variable of the graph, we can select appropriate variables which should be included in a model. However, this assumes that we know the SCM. Of course, the issue is that in most practical scenarios, the SCM is not known. What can be done?

Existing work in the causality literature. Causal discovery is the research area concerned with discovering the graphical representation of a SCM using data from the entailed distribution. However, there are a few challenges, some of which have been covered in more detail in Chapter 2. 1. Many developed algorithms, such as PC and FCI [30] relies on conditional independence tests, which doesn't scale to high-dimensional data such as images or time-series. 2. Other causal discovery algorithms which do not rely on hypothesis tests such as LinGAMs or ANM's [60] rely on making assumptions about the causal mechanisms allowed in the SCM. 3. In many cases, the algorithms may not be able to identify a particular graph, but rather an equivalence class of graphs (CPDAGs and PAGs for the PC and FCI algorithms respectively). 4. Often we are not so interested in the whole graph, but only in the variables surrounding the target variable, in particular the parents.

On the other hand, causal inference methods can estimate the effect of intervening on a particular variable on the outcome, and this may be useful in eliminating non-causal variables, such as exercise time, for the example above. We could find the causal effect of all variables on the target and keep only ones with non-zero effect. However this does not give you the parents; it may give you an ancestor of a parent, or any variable on the causal path to the target. The main problem with most causal inference methods is that it requires knowing something about the causal graph, as discussed in Chapter 2.

One promising approach is an empirical way to learn the cause (parents) of a target variable, without the need to learn the entire graph. This was the main contribution of invariant causal prediction (ICP) [18], which does this by leveraging data from different environments - a requirement that is practically very applicable. However, ICP relies on hypothesis tests, and therefore suffers with computational cost when it comes to high-dimensional data.

Building on the idea behind ICP, it turns out that if a model is able to predict well under a large variety of environments, it is likely to be closer to the causal mechanism of the target variable. In other words, to find the cause of some target variable, optimising a model to perform well under diverse environments should suffice, replacing hypothesis tests with prediction loss. This suggests that purely optimising for prediction accuracy (under diverse test distributions,

i.e., non-iid settings) is well founded, similar to how minimising empirical risk is well founded in the case where training and test data are iid. This is the main result - Empirical Causal Convergence (ECC).

The idea behind ICP is covered in the next section, followed by the theory behind the current main learning paradigm, empirical risk minimization (ERM). Given these two ideas, we have sufficient context to introduce empirical causal convergence (ECC), which then concludes the chapter.

3.4 Finding causal variables through invariant prediction

The main idea behind ICP, by Peters et al. [18] is that the causal variables of some target can be identified by leveraging data from multiple environments i.e., instead of pooling data together, the data is kept separate. By not discarding this extra information, it could be used to narrow down the causal variables. We briefly discuss the main ideas behind their work; all of the following is credited to Peters et al. [18]. A more detailed summary is given in Appendix A.

ICP makes two assumptions - the invariance assumption, intuitively that there exists a model that can predict well under all of the data environments. This corresponds to assuming that there is a common SCM. Additionally, that each of the data environments come from an intervention on the SCM that is not on the target variable (Y).

Given these two assumptions, it shows that the causal variables can be identified using a generic three step algorithm. Given p predictor variables (X), for each subset S of the predictor variables: 1. perform a hypothesis test on S of whether it can be used to construct an invariant model for all data environments. 2. The plausible causal predictors are the intersection of subsets S such that the hypothesis is not rejected. 3. The causal coefficients are then given by the union of the coefficients of the subsets that are not rejected. The causal predictors and coefficients come with statistical guarantees given the hypothesis test used satisfies certain properties.

This is significant because it flips a well known relationship on its head. Whilst it is obvious that causal variables lead to robust predictors (as discussed in a preceding example), the inverse idea that robust predictors can lead to the causal variables have not been considered previously.

However, this all relies on statistical hypothesis tests, which does not scale to high-dimensional data, and are often impractical. Is there a way to directly relate prediction performance to causal variables? Before this question is discussed, we need to examine the theoretical reason for empirical risk minimisa-

tion. This is the topic of the next section.

3.5 A theoretical perspective on learning

Much of the progress in machine learning in the past decade is empirically driven. The availability of benchmark datasets such as MNIST [61] and ImageNet [62] are key to this progress - it is the way in which advances in the field are measured. This empiricism has led to many breakthroughs, from protein structure prediction [2], to generating realistic images [63].

As an empirically driven field, it is important to not lose sight of the less visible and less immediately applicable foundational work which underpins certain practices today. Not least because it gives these practices some form of justification, many researchers have declared their impatience for theoretical understanding (why does it matter if it works?); But because revisiting some of these foundational ideas may help us come up with new ways to think about the problem.

This section gives an overview of one of the main paradigms in ML today - Empirical Risk Minimization (ERM) [64] and its justification, PAC learning [58]. To many practitioners it is standard practice to pool available training data and split it into three partitions. Then, train a model on the first partition (training set), choose hyperparameters on the second (validation set), and evaluate the performance on the final partition (test set). Why is this the case?

We are interested in two aspects in particular. 1. why we can expect a model to perform well on the test set when it is only trained using the training set, and 2. the validity of using a validation set for model selection. To address these two questions, we look to some results from learning theory - the topic for this section.

As we will discuss in the coming sections, the answers to both these questions rests on the key idea that the training, validation and test sets are independently and identically distributed (iid) according to some distribution. However, what can we do when this is not the case? Thinking about causality, and using the formalisation that have been developed to study causality, may be helpful to answer these questions. But first, we cover some ideas from learning theory which led to ERM, the main learning paradigm today.

The remainder of this section covers existing results. PAC learning is credited to being first proposed by valiant [58], but many results have been developed since. The discussion here is based on Shalev-Shwartz and Ben-David [65].

A formal model for learning. A formal model of learning can be defined as follows. There is a training dataset S, sampled from an unknown distribution $\mathcal{D}$. Let the space of the samples be denoted by Z. For instance in prediction

problems, this is the product of the feature and label space $Z = \mathcal{X} \times \mathcal{Y}$. We are interested in finding a model, often called a *hypothesis*, h. In the prediction setting, the hypothesis takes as input the features and outputs a label $h : \mathcal{X} \to \mathcal{Y}$. The hypothesis is selected from a set of potential hypotheses $\mathcal{H}$, called the hypothesis class, by the learning algorithm using samples S. This is achieved by evaluating a loss function $l : \mathcal{H} \times Z \to \mathbb{R}_+$, which takes a hypothesis and a sample and returns a positive number. In the prediction case, the loss function may be for instance the mean squared error $l(h, (x, y)) := (h(x) - y)^2$.

The *true risk* of a hypothesis h is defined as the expected loss of h w.r.t distribution $\mathcal{D}$ over Z.

$$L_{\mathcal{D}}(h) := \mathop{\mathbb{E}}_{z \sim \mathcal{D}}[\ell(h, z)] \tag{3.3}$$

It is called a *true* risk because it is the risk if we have full knowledge of the distribution $\mathcal{D}$. In practice, this is not the case. Instead, we only know the *empirical risk*, which is the risk over the training dataset S of size m. Denote the emprical risk by $L_S(h)$.

$$L_S(h) := \frac{1}{m} \sum_{i=1}^{m} \ell\left(h, z_i\right) \tag{3.4}$$

Finding a model h that achieves low empirical risk $L_S(h)$ is called Empirical Risk Minimization (ERM).

3.5.1 Why minimize empirical risk?

We can now take a look at our first question, which can now be stated more precisely. When data is generated from some unknown distribution $\mathcal{D}$ how can we measure the true risk on $\mathcal{D}$ if we only have a finite sample from the distribution? In other words, how can we be sure that training on the finite sample will mean our model will perform well on all samples that can possibly be drawn from $\mathcal{D}$?

The short answer is yes, if the model is trained with enough samples. The following section outlines this argument. The argument first defines more precisely what it means to 'perform well' by defining the idea of probably approximately correct (PAC) learnability. It then defines a concept of a 'representative' training set S, which means that learning with S should give a risk that is 'close' (formalised as ϵ) to the true risk on the entire distribution. It is then shown that if a sample S is representative, then a hypothesis h_S chosen using ERM on S will have a risk close to the hypothesis h that will minimise the true risk in the class $\mathcal{H}$. We can then define a property of hypothesis classes $\mathcal{H}$ called *uniform convergence* which says that there exists a number of samples $m_{\mathcal{D}}^{UC}$

such that if more than $m_{\mathcal{D}}^{UC}$ samples are drawn from the distribution then there is a high probability (formalised by δ) that the sample is representative, which means that we would get a hypothesis with a risk close to the true risk by using ERM on S. It is then shown that all finite hypothesis classes have this property, and for our purposes the discussion ends here. There is theory that tries to extend this notion of 'learnability' to infinite hypothesis classes with addition constraints, but for our discussions this is less relevant. As our computers have finite precision, for practical purposes all the hypothesis classes considered are finite. The arguments that follow are given without proof.

Definition 4 *(PAC Learnability) A hypothesis class $\mathcal{H}$ is PAC learnable with respect to a set Z and a loss function $l : \mathcal{H} \times Z \to \mathbb{R}_+$ if there exist a function $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm with the following property: For every $(\epsilon, \delta) \in (0,1)$ and for every distribution $\mathcal{D}$ over Z, when running the learning algorithm on $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d samples generated by $\mathcal{D}$, the algorithm returns $h \in \mathcal{H}$ such that, with probability of at least $1 - \delta$*

$$L_{\mathcal{D}}(h) \leq \min_{h' \in \mathcal{H}} L_{\mathcal{D}}(h') + \epsilon$$

where $L_{\mathcal{D}}(h) = \mathbb{E}_{z \sim \mathcal{D}}[l(h, z)]$

This means that the hypothesis returned from a PAC learnable class based on $m_{\mathcal{H}}$ samples will have a true risk which is bounded from above by the true risk of the hypothesis in the class $\mathcal{H}$ that minimises the true risk, plus some small margin. In other words, the returned hypothesis will be 'close' to the optimal hypothesis in the class.

To establish that all finite hypotheses have this property using the ERM rule, we must first establish when learning using ERM will give a risk that is close to the true risk.

Definition 5 *(ϵ-representative sample) A training set S is ϵ-representative with respect to a domain Z, hypothesis class $\mathcal{H}$, loss function l, and distribution $\mathcal{D}$ if*

$$\forall h \in \mathcal{H}, |L_S(h) - L_{\mathcal{D}}(h)| < \epsilon$$

Lemma 1 *If a training set S is $\frac{\epsilon}{2}$-representative, then learning with ERM on hypothesis class $\mathcal{H}$ using S, i.e., any $h_S \in argmin_{h \in \mathcal{H}} L_S(h)$ satisfies*

$$L_{\mathcal{D}}(h_S) \leq \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + \epsilon$$

This implies that for ERM to be a PAC learner, we need to show that an $\epsilon-$representative sample is drawn with probability of at least $1 - \delta$. This is called the uniform convergence property.

Definition 6 *(Uniform Convergence) A hypothesis class $\mathcal{H}$ has the uniform convergence property with respect to domain Z, and loss function l if there exists a function $m_{\mathcal{H}}^{UC} : (0,1)^2 \to \mathbb{N}$ such that for every $\epsilon, \delta \in (0,1)$ and every distribution $\mathcal{D}$ over Z, if S is a sample of size $m > m_{\mathcal{H}}^{UC}(\epsilon, \delta)$ drawn i.i.d from $\mathcal{D}$, then with probability of at least $1-\delta$, S is $\epsilon-$representative.*

Lemma 2 *If a hypothesis class $\mathcal{H}$ has the uniform convergence property with function $m_{\mathcal{H}}^{UC}$ then it is PAC learnable with sample complexity $m_{\mathcal{H}}(\epsilon, \delta) \leq m_{\mathcal{H}}^{UC}(\epsilon 2, \delta)$ using ERM.*

What is left to show is that finite hypothesis classes have the uniform convergence property, which means that they are PAC learnable using ERM.

Theorem 2 *If $\mathcal{H}$ is a finite hypothesis class, let Z be a domain and $l : \mathcal{H} \times Z \to [0,1]$ be a loss function. Then $\mathcal{H}$ has the uniform convergence property with sample complexity*

$$m_{\mathcal{H}}^{UC}(\epsilon, \delta) \leq \left\lceil \frac{\log(2|\mathcal{H}|/\delta)}{2\epsilon^2} \right\rceil$$

Additionally, $\mathcal{H}$ is PAC learnable using ERM with sample complexity

$$m_{\mathcal{H}}(\epsilon, \delta) \leq m_{\mathcal{H}}^{UC}(\epsilon/2, \delta) \leq \left\lceil \frac{2\log(2|\mathcal{H}|/\delta)}{\epsilon^2} \right\rceil$$

We have defined what it means for a learning procedure to 'perform well' by a property called PAC learnability on hypothesis classes. It is then shown that the hypothesis obtained by learning using ERM will have an empirical risk that is close to the true risk of the optimal hypothesis in the class if the sample is representative. We then define a property called uniform convergence on hypothesis classes that means if over a certain number of samples are drawn, the sample is representative with high probability. It is then established that if a hypothesis class has the uniform convergence property, then it is PAC learnable using ERM. Finally, it is shown that finite hypothesis classes have the uniform convergence property, and therefore learnable using ERM.

This concludes our first question - why should we expect a model to perform well on test data from some underlying distribution when we only have access to finite samples from said distribution for training? If the data is i.i.d and the hypothesis class is finite, the argument is that minimising the empirical risk should be representative of minimising the true risk.

An additional validation set is often used to get a better estimate of the true risk of a hypothesis, and it can be used to select a particular hypothesis out of several candidate hypotheses, based on the empirical risk on the validation

set. This is less relevant to our discussion here, and the formal results are summarised in Appendix B.

We have now seen arguments why training using the ERM framework is sound, and should lead to models that perform well on the test set. This relies on the key idea that data is iid. What, if anything, can we say when the training and test data is not from the same distribution?

3.6 Empirical causal convergence

In the previous sections we've seen work by Peters [18] which showed that the causal variable can be identified with access to data from different environments under certain conditions using repeated hypothesis tests. This is computationally expensive and not practical, as the number of tests scales exponentially with the number of variables. We have also seen the reasoning behind ERM, namely that it is representative of minimising the true risk, given that the training and testing set are i.i.d.

At first it may seem that these two ideas are disparate, but they can both be used in combination with SCMs to connect prediction robustness across distributions with the causal mechanism. Specifically, if a model achieves low risk when predicting a target variable across many data environments, it will move closer to the causal mechanism of the target. This has interesting implications, including optimising empirical risk when data is not i.i.d, and finding the causal variables without using hypothesis tests.

From an assumption point of view, it has several advantages. 1. It relies on having access to data from different environments, a practical condition in many applications. 2. It does not require knowing where the intervention was performed, only that there was an intervention. 3. It makes no assumption about the graph structure except that it is acyclic.

Intuition. There are datasets $D_1, D_2, ..., D_n$ sampled from intervention distributions of a common SCM $\mathbb{S}$ where the target variable X_t has some mechanism f_{X_t}. Denote the model trained on these datasets as h. The main idea is that as the number of dataset increases, h will be forced to abandon spurious features which do not hold for some datasets, as this will result in a large empirical risk. In other words, f_{X_t} is the only low empirical risk hypothesis across all datasets.

This hinges on the fact that each dataset contains some intervention, so if h learns anything other than f_{X_t}, then some intervention can make h have high risk. Assuming these interventions have non-zero probability, with every additional dataset, the set of possible functions h which can predict X_t with low risk across all datasets will either stay the same (intervention already seen), or will decrease, eventually moving close to f_{X_t}.

In short, if h is not close to f_{X_t}, then there exists an intervention that entails a distribution D_I where h doesn't predict well on D_I. If h is f_{X_t} then such a D_I no longer exists. As the number of intervention distributions D_I increases h should become closer to f_{X_t}.

The following is an approximation result for when a model h has low risk across all possible interventions, so it can be stated concisely.

Assumption 1 *Datasets $D_1, D_2, ..., D_n$ share a common SCM S. i.e., datasets are drawn from an intervention distribution of S.*

Assumption 2 *Interventions are possible on all variables, except the target. For any input $\mathbf{x}$ and $\mathbf{x}'$ with distribution $\mathcal{D}$ there is an intervention which changes their probability arbitrarily such that $\mathcal{D}$ remains a distribution. Denote this class of distribution $\mathbf{E}$.*

Definition 7 *Recall that the 'true' risk of some hypothesis h with respect to a distribution $\mathcal{D}$ is defined as*

$$\mathcal{L}_{\mathcal{D}}(h) = \mathbb{E}_{x\sim\mathcal{D}}[l(h, f_{X_t}, \mathbf{x})]$$

for some loss function l. For instance l could be the mean-squared error.

Definition 8 *A hypothesis h is ϵ-invariant to datasets $D_1, D_2, ..., D_n$ if the true risks of h on all distributions the datasets are drawn from are smaller than ϵ i.e., $\forall\mathcal{D}\ \mathcal{L}_{\mathcal{D}}(h) \leq \epsilon$*

Theorem 3 *Let h be a hypothesis that is ϵ-invariant to all distributions $\mathcal{D}$ entailed by the class $\mathbf{E}$ of interventions defined in assumption 2 based on the mean-squared error loss. Then, the distance defined by the norm induced by the inner product between h and the causal mechanism of the target variable f_{X_t} decreases with ϵ:*

$$||h - f_{X_t}|| \leq \sqrt{\frac{\epsilon}{\kappa}}$$

where $\kappa := sup_{p(x)\in\mathbf{E}} \int_{-\infty}^{\infty} p(x)l(h, f_{X_t}, \mathbf{x})dx = p(c) \cdot \int_{-\infty}^{\infty} l(h, f_{X_t}, \mathbf{x})$ for some c.

Proof. Define the distance between h and f_{X_t} as the norm induced by the inner product:

$$||h - f_Y|| =< h - f_Y, h - f_Y >^{\frac{1}{2}} = \sqrt{\int_{-\infty}^{\infty} (h(x) - f_Y(x))^2 dx}$$

We know that h is ϵ-invariant to all distributions in $\mathbf{E}$, based on mean-squared error.

$$\forall D \in \mathbf{E}\ \mathbb{E}_{x\sim D}[(h(x) - f_Y(x))^2] \leq \epsilon$$

$$\therefore \forall p(x) \in \mathbf{E}\ \int_{-\infty}^{\infty} p(x)(h(x) - f_Y(x))^2 dx \leq \epsilon$$

The notation $\in \mathbf{E}$ is dropped in the remainder for readability. Using the mean value theorem, if f is continuous on [a,b] and g is non-negative and integrable on [a,b] then $\exists c \in [a, b]$ such that:

$$\int_a^b f(x)g(x)\ dx = f(c)\ \int_a^b g(x)\ dx$$

Let $f(x) = (h(x) - f_Y(x))^2$ and $g(x) = p(x)$:

$$\forall p(x)\ \exists c\ \int_{-\infty}^{\infty} p(x)(h(x) - f_Y(x))^2 dx = p(c) \cdot \int_{-\infty}^{\infty} (h(x) - f_Y(x))^2 dx \leq \epsilon \quad (3.5)$$

$$\therefore \forall p(x)\ \exists c \int_{-\infty}^{\infty} (h(x) - f_Y(x))^2 dx \leq \frac{\epsilon}{p(c)}$$

$$\therefore \forall p(x)\ \exists c\ ||h - f_Y|| \leq \sqrt{\frac{\epsilon}{p(c)}}$$

Setting κ equal to the maximum value $p(c) \in \mathbf{E}$ such that Eq 3.5 is true completes the proof.

Finite samples. Of course, so far the discussion has been using datasets interchangeably with distributions. Whereas in practice, we do not have access to $\mathcal{D}$, but rather samples S from $\mathcal{D}$. Since our definition of risk is the same as those used in the PAC model of learning, all of the guarantees also apply in our scenario, but an additional assumption is required:

Assumption 3 *Assume the datasets $D_1, D_2, ..., D_n$ are large enough to be ϵ-representative in the PAC learning sense. i.e., the empirical risks are close to the true risks of the optimal hypothesis in the class.*

3.7 Discussion

This chapter established the causal view of robustness. This has allowed us to precisely define what a 'domain' is by using SCMs. This has also made clear how different distributions may be related, and enabled the formulation of a robust model in terms of SCMs.

Using the causal view, we saw that existing literature in causality is not immediately helpful in training robust models. This is because we don't know

the causal graph in most practical scenarios. We argue that the most practical approach to learning a robust model is to exploit data from multiple intervention distributions, an idea first proposed by Peters [18]. Due to the limitations of the hypothesis tests used by Peters, we propose the idea of empirical causal convergence, which connects prediction loss to the causal mechanism.

The ECC idea has some interesting implications. 1. It provides a reason for only optimising for prediction performance even in non iid settings, similar to ERM when data is iid. 2. It provides a way to view model failures as artifacts of using non-causal features for prediction. 3. It provides a possible explanation of the unreasonable effectiveness of large models. The larger and more diverse the training set, the closer it is to the cause, and the more robust the prediction under interventions. 4. It is consistent with Popper's philosophy of science, where the cause is the hypothesis that robustly predicts all of the different observations seen so far. With each new additional data environment, the current model is the best guess of the cause. If a new data environment emerges which results in poor prediction, then the current best guess is incorrect and is updated to take into account the new data environment.

It is important to note that the ECC idea does not propose a practical way to find a robust model given several intervention distributions. It only says that minimising the loss across diverse interventions is a step in the right direction. i.e., loss across intervention distributions can be used as a model selection procedure. Additionally, the current formal proof only works in the case of MSE loss, although we suspect it should hold for more general losses, this remains to be solved.

Of course, the causal view is not the only view, and whether it is a good way to think about robustness remains to be seen. The SCMs considered here are acyclic, and there are real systems which we would like to study that cannot be represented by a DAG. Examining how these ideas translate into cyclic SCMs is a worthwhile future direction.

3.8 Conclusion

In the beginning of the chapter we looked at different model failures in literature, existing formulations of robustness, and various empirical observations that suggest connecting robustness to causality can be worthwhile. We presented a formulation of robustness using SCMs, and looked at whether existing literature in causality can be helpful in training robust models. We cover some existing work in detail, specifically ICP [18] and PAC learning [58], which led to the idea of empirical causal convergence. ECC has several interesting implications, the main one being that optimising for prediction loss across an increasing

number of intervention distributions will move the model towards robustness.

Ultimately, rather than being another pointless theoretical construct we hope that this perspective unites several ideas and inspires new ways to think about training robust (in the sense described in this chapter) models - some of which are explored in the remaining chapters.

In the next chapter, based on the causal perspective presented here, we propose a benchmark to measure the robustness of Human Activity Recognition (HAR) models using multiple datasets representing different intervention distributions. It is shown that two state of the art classification models face significant performance degradation which suggests that the models are using non-robust features. Then in Chapter 5, a data selection method is proposed to increase the robustness of the resulting trained model in the light of the discussion presented here.

Chapter 4

Evaluating robustness in human activity recognition models

In the previous chapter we have introduced the causal view of robustness - a robust model can be defined as one which performs well across different intervention distributions of a common SCM. We have also discussed that there is no straightforward way to learn causal features - causal discovery methods only produce graphs up to Markov equivalence. However, because of empirical causal convergence (ECC) we should move closer to a robust model by optimising purely for performance on different intervention distributions. On the flip side, this means that we can empirically measure a model's robustness by looking at the performance of said model under different intervention distributions. In this chapter this idea is explored for a particular application area; we investigate the robustness of state-of-the-art models developed for human activity recognition (HAR) using a binary classification task. The robustness of these models have not been as extensively studied as other applications such as vision.

To measure the robustness of HAR models, a different evaluation setup is needed from what has traditionally been used. Since 2004, HAR models have been evaluated mostly according to the ERM paradigm, with a single dataset using a train/validation/test split. More recently, transfer learning and domain adaptation techniques train on one dataset and tests on another dataset, but use a small subset from the test dataset for transfer/adaptation [66, 67, 68, 69, 70]. Here, to measure robustness we propose that performance is measured on multiple unseen datasets which can plausibly be different intervention distributions

of a common SCM.

We will see that state-of-the-art HAR models face severe performance degradation on this new benchmark. This suggests that these models are not robust. To further this point, it is shown that a much simpler model that uses features which are more likely to be robust performs equally well whilst being at least ten times faster to train.

Additionally, we observe, in accordance with ECC, that training on multiple domains improves model robustness (as seen in improved performance for both seen and unseen domains) - providing empirical support of ECC. We further observe that, when additional labelled training data is used from domain A, performance on the test split of domain A increases without any transfer or adaptation technique.

Outline. This chapter first introduces the problem from the perspective of the HAR literature, and covers related work in domain adaptation and transfer learning, in particular how the tasks are different to the proposed experiments for evaluation. We then connect the proposed experiments to the causal view, and more precisely define the problem and experimental setup. The results for two SotA deep models are presented. A simpler model is proposed and the results on the same evaluation experiment is shown. We then cover an additional experiment using multiple domain training, and a discussion on how this fits into the causal view of robustness, which concludes the chapter.

4.1 Introduction

Real-world deployment of HAR models face multiple challenges; a major one being test time data heterogeneity [71], which degrades performance significantly. These heterogeneities can arise, amongst many factors, from users, sensors or changes in the environment over time. Users may walk at different speeds, perform the same activities in slightly different ways; different sensors may be used, sensor data may be affected by the environment; the environment itself may affect how the user performs activities. Given that it is the unchanging activity that we are trying to detect, models trained to classify any particular activity should ideally perform well in all of these varying scenarios. We want a model that is robust to these changes in the sense discussed in Chapter 3.

Current research in HAR which tackles test time heterogeneity use transfer learning or domain adaptation techniques [67][66]. The key challenge with these approaches is that it requires some data from the target domain. This also means that the models are domain specific; for each new test domain, the models need to be retrained. For real deployments, this suggests having collected data for all possible domains the model may encounter, which can be impractical.

We need to develop models that are domain-agnostic; i.e., they perform well on unseen test domains of a previously seen activity. This corresponds to the notion of robustness we have discussed, each heterogeneity can be thought of as an intervention on the SCM. Humans already have this ability to recognise known activities in completely new scenarios. This requires stepping back and revisiting some fundamental questions about how we think about and develop HAR models. The ECC idea previously discussed suggests a way to measure model robustness. How do current models fair?

Current evaluation methodology in the HAR literature do not measure domain-agnostic performance. This work proposes a generalisation of the leave-one-subject-out regime to the dataset level - imaginatively called leave-datasets-out. An instance of this evaluation method for the task of HAR is given as a simple binary classification between two common activities across three openly available HAR datasets. Using this it is shown that two current state of the art deep models face significant performance degradation in unseen domains.

We show that a simple model using an appropriate inductive bias, which is informed by likely causal features, performs at least as well as two state of the art end-to-end deep learning models, whilst requiring significantly less resources to train. This unexpected result raises questions about using deep end-to-end models as a one-size-fits-all solution in applications with small labelled datasets such as HAR when unseen domain performance is key.

We make two additional observations when using datapoints from multiple domains for training. First, that this additional data diversity consistently increases the performance in the *unseen* domains across the three models considered in this work. And two, simply having access to a few datapoints from a target domain greatly increases the performance on said domain without any complex transfer or adaptation technique.

Contributions. In summary, the contributions of this work are the following:

1. The leave-datasets-out evaluation regime, and a corresponding benchmark which measures HAR models' performance on unseen domains using a binary classification. This serves as a better proxy to performance in real deployment.

2. A simple model using an appropriate inductive bias which performs at least as well as current deep models on the new benchmark that requires significantly less training resources.

3. Evidence to suggest that having a few datapoints in the target domain can improve performance in the corresponding domain without complex transfer and adaptation.

4. Additionally, datapoints from an additional domain improves unseen and seen domain performance, pointing to the importance of data diversity in training domain-agnostic models.

4.2 Related work

This section gives a brief overview of related work in HAR. Specifically, the general problem of data heterogeneity, domain adaptation and transfer learning. This work is placed in context to motivate domain-agnostic models. As will be shortly discussed, the domain adaptation and transfer task tackles a different problem to the one in this chapter, which mainly is interested in measuring domain-agnostic performance, i.e., robustness as defined in the causal view.

HAR using accelerometer sensors have been researched since at least 2004 [72]. The models have moved from hand-crafted features [73][74], to end-to-end deep learning models using CNN and LSTM architectures [75][70].

Chen et al [71] provides an overview of the field of activity recognition using sensors and outlines three categories of what they term 'heterogeneity', defined as when the training and testing data are not independent and identically distributed (i.i.d). According to this work, heterogeneity arise from differences in users, sensors, and overall environment. They mention 19 different works which use transfer learning to tackle the heterogeneity problem. It is noted that all these works require either labelled or unlabelled data from the target domain.

The two most common approaches that deal with heterogeneity in the HAR literature are *transfer learning* and *domain adaptation* methods. These two terms often refer to the same problem setting i.e., there is a source domain dataset $\mathcal{D}_s = \{\mathbf{x}_i, y_i\}_{i=1}^m$ and either a labelled or unlabelled target domain dataset $\mathcal{D}_t = \{\mathbf{x}_i\}_{j=1}^n$ where the goal is to learn a classifier using $\mathcal{D}_s$ that performs well on $\mathcal{D}_t$. Transfer learning is a general term which also includes trying to transfer knowledge learnt from other activities to aid in learning to classify a new activity, e.g., [76], or transfering knowledge (CNN filters) from training using one dataset to aid learning to classify in a new dataset [66]. [70] considers transfer (by reusing convolutional kernels) between sensor modalities, sensor location, users, and application domains. [67] considers the general problem of source domain selection (by minimising distribution distance), and of how to learn features for transfer. Zhao et al. [69] considers transfer between users, and sensor locations by proposing a 'local' transfer method based on pre-defined clusters of activities.

Chang et al. [68] focuses purely on the problem of adapting between sensor locations and evaluates three adaptation techniques. They assume access to unlabelled datapoints from the test domain by using unsupervised domain

adaptation, and primarily consider robustness in terms of the positioning of the sensors. They additionally observe that creating a model which is accurate across different sensor locations relies on the assumption that there exists a data representation that can be used to accurately classify activities that is fixed across each sensor position. It is unclear whether this is indeed true and is the motivation to fix sensor location in this work.

The key aspect of transfer and adaptation techniques covered so far is that it assumes access to data (either labelled or unlabelled) from the target domain during training. This poses a few challenges. 1. The model is adapted to specific target domains, which means retraining is required when it needs to work on a new target domain. 2. In real deployment, there are many potential target domains, and it is impractical to assume that we have data from all domains that might be encountered. 3. Even with data, retraining on each target domain can be resource-intensive, depending on the model, and this limits use on many resource constrained devices.

4.3 Measuring robustness in deep HAR models

In this section we briefly discuss how this fits into our work on model robustness. Then the problem setup is described more precisely, culminating in a definition of what 'domain-agnostic' means in the HAR context. The new HAR benchmark is motivated and introduced, and two existing state-of-the-art deep models are tested against this new benchmark.

4.3.1 The need for a robustness benchmark in HAR

HAR models have been evaluated under different setups, which means it is hard to compare models. Jordao et al. [77] standardised the evaluation of many HAR models and pointed out that the way that windows are usually generated from HAR datasets results in the same data potentially being in both the training and test set.

The leave-one-subject-out regime is now often used to mitigate the issue of having the same window in both training and test sets. However, evaluation is still performed using a single dataset.

According to our view of model robustness, this means that models can still latch on to specific spurious features inherent in the dataset that classifies the label well, as the dataset corresponds to only one possible distribution entailed by the SCM. If existing deep models have learned some useful non-spurious feature, then it should perform well on multiple datasets of the same task, corresponding to different distributions entailed by the SCM. This is the motivation behind

our proposal that models should be evaluated using a leave-datasets-out regime instead.

The particular benchmark used in this work is designed to be minimise any additional spurious factors. In particular, the location of the sensors across datasets are matched, the measurement units are normalised to match, and the sampling frequency are also matched. The task is also restricted to a binary classification as a starting point, whereas models in literature are used for multi-class classification. This should give an upper bound on the robustness of tested models.

4.3.2 Problem setup and domain-agnostic models

We are interested in the case where we have accelerometer data for a particular participant $\mathbf{x} \in \mathcal{X}$ and activity labels $y \in \mathcal{Y}$. We assume to have access to n datasets $\mathcal{D}_1, \mathcal{D}_2, ...\mathcal{D}_n$ each corresponding to the same activities but in a different domain (corresponding to a different distribution). Each dataset consists of m_k pairs $\mathcal{D}_k = \{(\mathbf{x}_i, y_i)_{i=1}^{m_k}\}$, where each pair corresponds to data from participant i, which in turn is assumed to be independently and identically distributed samples from the corresponding domain. The feature space $\mathcal{X}$ and label space $\mathcal{Y}$ are the same across all datasets. In this specific work, there are three datasets ($n = 3$) with 7, 10 and 9 participants respectively ($m_1 = 7$, $m_2 = 10$ and $m_3 = 9$).

In this work, we often refer to datasets used for training and testing in the following way. Let $\boldsymbol{\mathcal{D}_{tr}}$ denote the set of training dataset(s), and likewise $\boldsymbol{\mathcal{D}_{te}}$ for the testing dataset(s)[1]. For instance the n datasets can be partitioned into two groups, $\boldsymbol{\mathcal{D}_{tr}}$ and $\boldsymbol{\mathcal{D}_{te}}$. The goal is to only use $\boldsymbol{\mathcal{D}_{tr}}$ to train a model that will perform well on $\boldsymbol{\mathcal{D}_{te}}$. i.e., we want to minimise $\mathbb{E}_{\mathbf{x},y\sim\boldsymbol{\mathcal{D}_{te}}}\mathcal{L}(M, (\mathbf{x}, y))$ whilst only having access to $\boldsymbol{\mathcal{D}_{tr}}$, where $\mathcal{L}$ is some loss function, M is the trained model, and $\mathbf{x}, y$ is the data.

Difference to domain adaptation. We note the difference between the setup just described to the typical domain adaptation or transfer leaning setup where there is a designated source $\mathcal{D}_s$ and target $\mathcal{D}_t$ domain, usually corresponding to two different datasets. A model is trained on $\mathcal{D}_s$ and then adapted to work on the target using a subset of data from $\mathcal{D}_t$ [69, 70, 68, 67, 66] i.e., $\boldsymbol{\mathcal{D}}_{tr} = \{\mathcal{D}_s\}$ and $\boldsymbol{\mathcal{D}}_{te} = \{\mathcal{D}_t\}$. The main difference here being that we are *not* interested in the performance of any one particular domain $\mathcal{D}_t$, but rather the performance in domains where the model has not seen any data (i.e., not $\mathcal{D}_t$ or $\mathcal{D}_s$), in addition to the domains where it has already seen data.

Of course, it is impossible to only use $\boldsymbol{\mathcal{D}_{tr}}$ and perform well on arbritary

[1] A bold font is used to denote a set of datasets whereas a normal font is used when referring to single datasets.

$\mathcal{D}_{te}$, as a consequence of the no free lunch theorem. There must be some connection between them. Thinking in terms of the data generating mechanism can help clarify what the connections we are interested in are, and help guide our choices for model development. For instance, we can consider a simplified data generating mechanism for our HAR data, illustrated in figure 4.1.

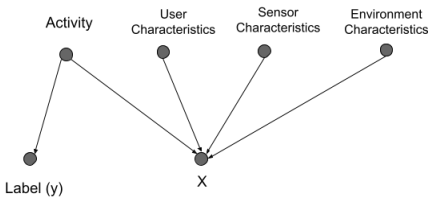


Figure 4.1: A possible data generating mechanism for HAR data. x is the observed data. A node represents a variable, or group of variables. An arrow from node A to node B signifies that A influences the value of B in the data.

Here, the user, sensor and environment characteristics are aggregate variables containing the individual factors which may influence the observed data, $\mathbf{x}$. For instance, sensor characteristics may include sensor location, rotation and sensor measuring units. Environment characteristics may include terrain, whether it is windy, or wet.

We can now give a more precise definition of the 'domain's we are interested in. A domain is a particular setting of the user, sensor and environment variables which give rise to a particular observed dataset of $(\mathbf{x}, y)$ pairs. These different datasets are connected because we assume that it comes from the same generating mechanism, but using different values for the variables that influence $\mathbf{x}$ and y. When we say we want to perform well on unseen domains (i.e., a domain-agnostic, or robust, model), it is not for *arbitrary* domains, but the collection of domains which arise from the different settings on the same mechanism. In other words, we are interested in recognising an activity regardless of changes in user, environment, and sensor characteristics[2]. In short, we are interested in training a model which performs well on different intervention distributions of the common SCM that generates the HAR data.

4.3.3 Measuring domain-agnostic performance

Now that we have better defined what 'domain-agnostic' means, how do we measure it? In this section we propose leave-datasets-out cross-validation, an extension of the usual leave-one-subject-out cross-validation commonly used in HAR. This follows from the idea discussed in Chapter 3; if a model becomes

[2]Assuming this is the entire causal graph

more robust after seeing all intervention distributions, then we can also use performance on the intervention distributions we have as a measure of robustness.

In traditional learning, k-fold cross validation (CV) is often used to estimate the true error of the model (defined as the loss over the unknown distribution the data was drawn from) by taking the average of the loss of each fold. Denote the partitions $1, 2, ..., k$ of a training dataset $\mathcal{D}$ as $\mathcal{D}^1, \mathcal{D}^2, ...\mathcal{D}^3$. The model in the ith fold is trained using all partitions except $\mathcal{D}^i$, i.e., $\boldsymbol{\mathcal{D}_{tr}} = \{\bigcup_{j \neq i} \mathcal{D}^j\}$, and tested on partition i, $\boldsymbol{\mathcal{D}_{te}} = \{\mathcal{D}^i\}$. Denote by $M(\boldsymbol{\mathcal{D}_{tr}})$ a model trained with datasets in $\boldsymbol{\mathcal{D}_{tr}}$. Let $\mathcal{L}(M(\boldsymbol{\mathcal{D}_{tr}}), \boldsymbol{\mathcal{D}_{te}}))$ denote the loss of a model trained on $\boldsymbol{\mathcal{D}_{tr}}$ and tested on $\boldsymbol{\mathcal{D}_{te}}$ for some loss function $\mathcal{L}$. The overall error in k-fold CV of a model M is then approximated by:

$$\text{Error}(M) = \frac{1}{k} \sum_{i \in 1,...,k} \mathcal{L}(M(\bigcup_{j \neq i} \mathcal{D}^j), \mathcal{D}^i)$$

using a single dataset $\mathcal{D}$. In the context of timeseries analysis, especially in HAR, a variant called leave-one-subject-out CV is often used. This is to avoid the same portion of data appearing in both the training and testing sets, due to the way the timeseries from each participant is split into samples using overlapping windows.

Leave-datasets-out (LDO) cross-validation. In this work, we use a natural extension of this idea to measure domain-agnostic performance, called leave-datasets-out CV. In the simplest setting, given n datasets $\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_n$, the domain-agnostic error is approximated by:

$$\text{Error}(M) = \frac{1}{n} \sum_{i \in 1,...,n} \sum_{j \in 1,...,n} \mathcal{L}(M(\mathcal{D}_i), \mathcal{D}_j) \tag{4.1}$$

where if $i = j$ then leave-one-subject-out CV is used, and when $i \neq j$ the model is trained on $\mathcal{D}_i$ and tested on the full dataset $\mathcal{D}_j$. In a later section of this work we will consider the case where we train on multiple datasets instead of a single D_i. This captures the idea that we are interested in the performance of the model on the collection of datasets which could have resulted from the same mechanism, such as that discussed previously.

A starter LDO benchmark for HAR. As there are many possible variations in the user, sensor and environment characteristics in the real-world, starting simple before moving on to more complex scenarios will help us understand model failures and hence how to improve them. In this work we set out to find three datasets which have as similar characteristics in the generating mechanism as possible, and two activities which are common across all datasets. This more restrictive test can indicate whether current models are able to deal

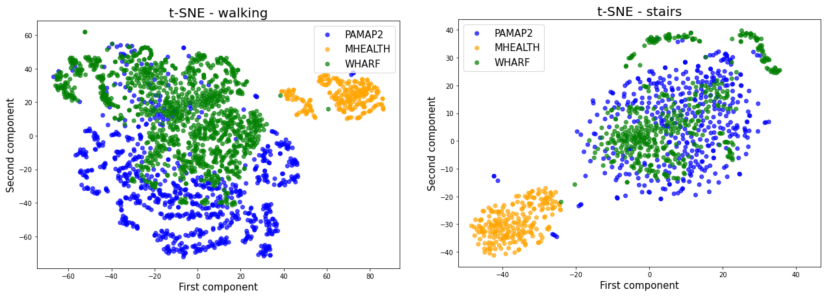


Figure 4.2: t-sne plots for walking and ascending stairs samples for PAMAP2, MHEALTH and WHARF datasets.

with a smaller subset of data heterogeneity.

In particular, we use three open HAR datasets which share the walking and stair climbing activities, and where the sensor are worn on the same body position. This leaves heterogeneity in the user, which is expected in real-world deployments, and any other heterogeneity in the sensors that are not related to its placement.

If we are unable to perform well with these more restrictive heterogeneity, then it is worthwhile to understand why before moving on to tackle more complex scenarios, such as location independent models [68] and scenarios with several activities. Additionally, in the HAR literature it is difficult to find several shared activities across many datasets.

Datasets. There are three datasets, MHEALTH [78], PAMAP2 [79], and WHARF [80], which contain data from sensors located on the right wrist of the participants. There are only two overlapping activities across all datasets, walking, and ascending stairs. To measure domain agnostic performance we will train a binary classifier between walking and ascending stairs using one dataset, and test on the other two. Some data characteristics are shown in Table 4.1. Additionally, t-sne plots for the three datasets are shown in Figure 4.2, for both the walking and climbing stairs activity. The three axes for each window sample were concatenated to one longer sample for each point in the plot. Perplexity and iteration parameters were ranged between (10,20,40) and (300, 900, 2700) respectively, which produced consistent results - there is some overlap between WHARF and PAMAP2 samples, with MHEALTH in its own cluster. Figure 4.2 was produced with perplexity of 40 and 2700 iterations.

Preprocessing. For each dataset, samples were filtered for the two common activities (walking, stairs), and only for readings captured from a sensor on the right-wrist of the participant. Invalid values and anomalies were removed. In particular, NaN sensor readings and portions of two subjects' recordings from the WHARF dataset which contained anomalous readings were removed. The

plots for the removed portions are shown in Figure 4.3. Participants 8 and 9 are excluded from the PAMAP2 dataset. Participant 8 had the sensor on the left wrist, and participant 9 has no data for the walking and stairs activities. Participants f6 and f2 are excluded from the WHARF dataset. Participant f6's data for stairs is too short to produce a single window sample, and f2 has anomalous walking data, shown in the bottom of Figure 4.3. The time-series for each participant was normalised to a common sampling rate of 50Hz, amplitude normalised, and values converted to a common unit (ms^2). Axes were aligned as much as possible. The subject-timeseries is then segmented into 5 second windows (250 samples at 50Hz) with an overlap of 2.5 seconds (125 samples at 50Hz). This resulted in 1207, 460 and 1589 samples from PAMAP2, MHEALTH and WHARF respectively; of which 412, 230 and 452 are stairs samples.

Training. Learning rates were determined for each model by picking from a range of values (0.01, 0.02, 0.5, 1 and 2) that lowered the validation loss the most. This was 1 for DeepConvLSTM and 0.01 for DeepConv. The models were trained for 10k iterations, where for each iteration a single window sample is processed. The number of iterations is largely arbitrary but was chosen to avoid overfitting on the training dataset. Training for longer in general reduced the performance on the validation set of other datasets. A learning rate scheduler was used to decrease the learning rate during training but this did not help with the validation loss for either model.

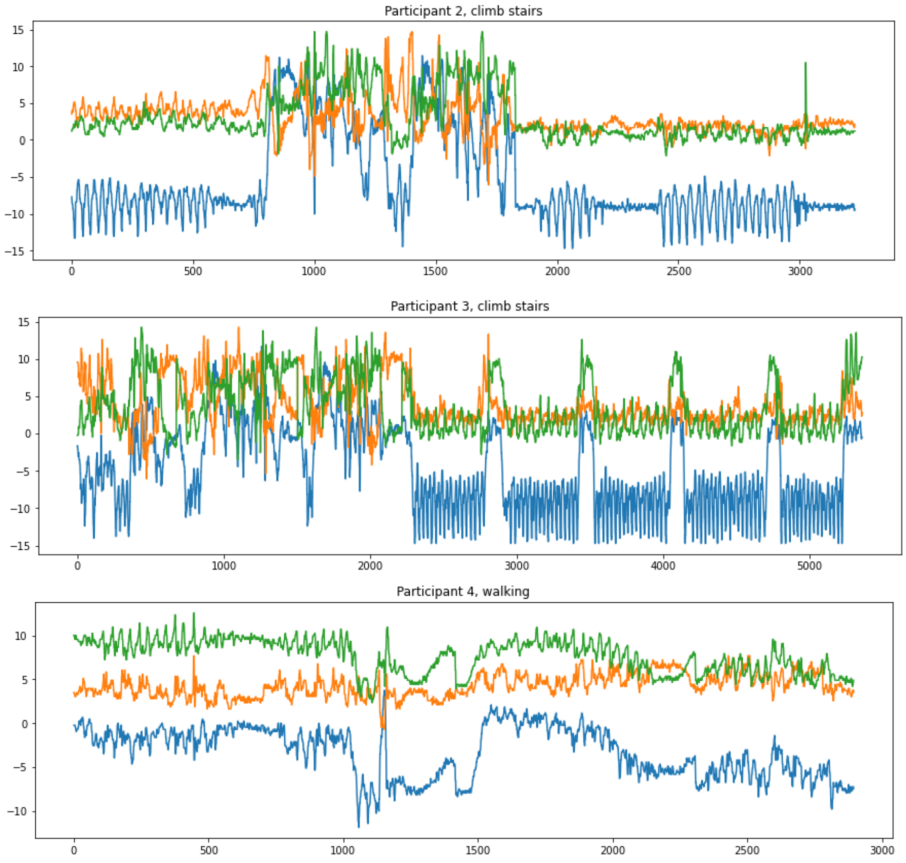


Figure 4.3: The portions of data from participant 2 and 3, for the ascending stairs activity, in the WHARF dataset that is anomalous and removed. Bottom: walking data from participant f2, which is anomalous.

Table 4.1: Dataset characteristics.

(a) Collected activities

Dataset	Activities	Collection Type
PAMAP2	lying, sitting, standing, walking, running, cycling, nordic walking, watching tv, computer work, car driving, ascending stairs, descending stairs, vacuum cleaning, ironing, folding laundry, house cleaning, playing soccer, rope jumping	Continuous
MHEALTH	standing, sitting, lying, walking, climbing stairs forward waist bends, front arm elevation, knees bending, cycling, jogging, running, jumping forwards and backwards	Continuous
WHARF	Brush teeth, Climb stairs, Comb hair, descend stairs, drink from glass, eat meat, eat soup, get up from bed, lie down in bed, pour water, sit on chair, stand from chair, use telephone, walk	Separate

(b) Sensor characteristics

Dataset	Sensor device	Sampling Rate	Range	Units	Sensor location
PAMAP2	Colibri wireless IMU	100 Hz	$\pm 20g$	ms^2	Right wrist
MHEALTH	Shimmer 2	50 Hz	$\pm 16g$	ms^2	Right wrist
WHARF	'Ad-hoc' accelerometer	32Hz	$\pm 6g$	Ad-hoc	Right wrist

(c) Participant characteristics

Dataset	Participants	Male/Female	Average Age	Average Weight	Average Height
PAMAP2	9	8/1	27.2	80.9 kg	179 cm
MHEALTH	10	Unknown	Unknown	Unknown	Unknown
WHARF	17	11/6	57.4	72.7 kg	Unknown

4.3.4 Current model performance

We test two state-of-the-art deep neural network models from the literature. One is attributed to [81], a convolutional model, which was chosen as it consistently outperforms in a standardised test [77]. Another is the DeepConvLSTM model which uses both convolutional and LSTM layers [75]. The particular implementation used here is credited to [82]. The results are shown in Table 4.2.

We can see that for the DeepConv model, training and testing on the same dataset had higher performance than when testing on a different dataset. Whilst for the DeepConvLSTM model, the model struggles to learn from MHEALTH, most likely due to the large number of parameters in the model compared to the small size of MHEALTH, as suggested by the small gradients propagating through the network. Additionally, varying learning rate, training duration and learning rate scheduling was unable to improve learning. As this model was suggested by a reviewer, this baseline is kept for completeness.

Since current models in the literature are multiclass classifiers, i.e., they are able to distinguish between many different activites, it is expected that they should perform well on a binary classification task.

(a) DeepConv model

Train/Test	PAMAP2	MHEALTH	WHARF
PAMAP2	73.7 (0.138)	57.7 (0.0321)	54.9 (0.0296)
MHEALTH	46.6 (0.0356)	84.3 (0.151)	53.3 (0.0335)
WHARF	60.4 (0.0453)	67.2 (0.0929)	68.7 (0.128)

(b) DeepConvLSTM model

Train/Test	PAMAP2	MHEALTH	WHARF
PAMAP2	75.2 (0.0819))	49.1 (0.0306)	52.6 (0.0221)
MHEALTH	53.1 (0.0162)	0.5 (0) [a]	49.9 (0.0077)
WHARF	50.0 (0.000655)	50.04 (0.00131)	69.0 (0.123)

[a]The accuracy is random likely due to the small size of the MHEALTH dataset relative to the large number of parameters in the DeepConvLSTM model.

Table 4.2: Results showing average accuracy of a model in percentages trained using the dataset in the left column, and tested on the dataset in the first row. If testing on the same dataset, the left out participant is used to test, otherwise the entire dataset is used for testing. The standard deviation is given in brackets.

4.4 Improving domain-agnostic performance

The previous section proposes a starting point to measure robustness of models in HAR. By considering the data generating mechanism, this section investigates two possible ways which, in theory, should help improve robustness, whilst also improving training efficiency.

4.4.1 Using an inductive bias

By considering the data generating mechanism (plausible SCM) shown in Figure 4.1 we can see that the observed data can be influenced by a number of different factors other than the activity performed by the user. This raises an important point: models can easily be fooled by spurious factors which may be predictive of the activity label. And the larger the models (in terms of parameters), the more likely that it is able to learn to use these spurious factors.

Lets take a concrete example. It may be the case that in one particular dataset, data collection for the walking activity was performed only on the elderly, whereas in more strenuous activities, such as running, data was collected on younger participants. This would suggest that a model would, in theory, be able learn to discriminate the walking activity by only using user characteristics that are present in elderly participants. When using this model on a different dataset where walking may also be performed by younger participants, the model would face performance degradation.

The two simplest ways to reduce the likelihood that a model is fooled by confounding factors is to reduce the size of the hypothesis class, and by incorporating the researcher's knowledge about the problem in the form of an inductive bias. In this particular case, based on our understanding of human activity, we know that motions associated with an activity is performed at a relatively low frequency i.e., at most a couple of times per second. We further know that we are not so interested in features that do not affect the general *shape* of the motion, such as amplitude of the time-series, as we know the general shape is what determines the activity rather than the range in which they are performed.

As the simplest implementation of this idea, the discrete log Fourier transform (DFT) power spectrum of low frequencies up to 3 Hz, (at a resolution of 0.25Hz, totalling 12 features) was used as features through a multi-layer perceptron (MLP) network. Albeit it's simplicity, it fulfils the two criteria: reducing the hypothesis class, and incorporating an inductive bias. A plot of the power spectrum for all window samples in the considered datasets is shown in Figure 4.4, supporting the idea that there is indeed discriminative power between the two activities using only the DFT power spectrum alone.

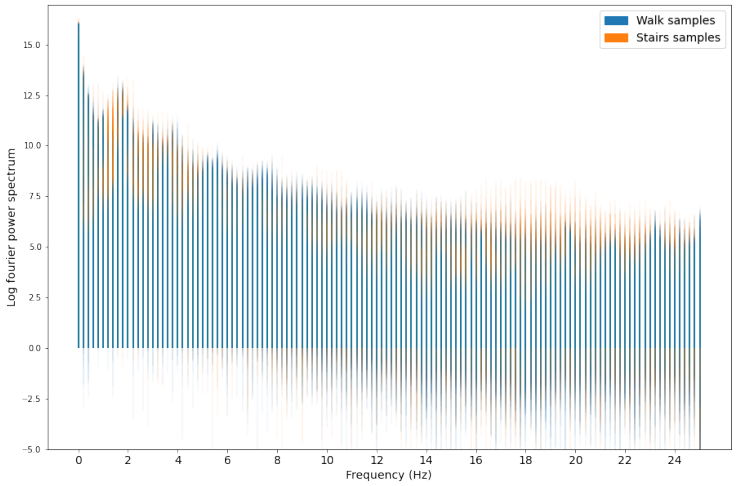
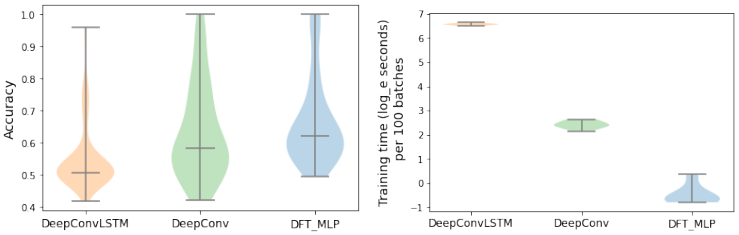


Figure 4.4: A (log) power spectrum plot of all window samples, red bars are stairs samples, and blue represents walking samples.

The results for different configurations of D_{tr} and D_{te}, similar to the previous test on two state-of-the-art deep models, is shown in Table 4.3. They were trained for 10k iterations and the selected learning rate, chosen in the same fashion to the deep models, was 1. A learning rate schedule was used which halved the learning rate every 2k iterations. The particular values were similarly selected using performance on the validation set. Direct comparisons with the deep models in terms of domain-agnostic performance is shown in Figure 4.5a, and a comparison of (log) training time is shown in Figure 4.5b, using commodity hardware on an Intel Core i7-8650. A two-sided statistical test was performed to detect whether the average performance of the DFT-MLP model is different to the DeepConv and DeepConvLSTM model yields a p-value of 0.131 and $5.75\text{x}10^{-4}$ respectively. The difference in training time across all models is significant (below 0.01).



(a) (b)

Figure 4.5: Left: Performance on the generalization benchmark of each model. Right: Training time for each model, note that due to the differences, the y-axis is on a log scale.

DFT MLP

Train/Test	PAMAP2	MHEALTH	WHARF
PAMAP2	74.61 (6.06)	57.23 (3.70)	63.63 (2.32)
MHEALTH	55.09 (2.59)	92.83 (12.5)	60.46 (1.88)
WHARF	65.24 (3.83)	56.70 (5.51)	70.77 (11.9)

Table 4.3: Results showing average accuracy in percentages of a model trained using the dataset in the left column, and tested on the dataset in the first row over 10 iterations. The standard deviation is shown in brackets.

The DFT-MLP model was additionally trained by varying the maximum frequency in the power spectrum that was used as features in the MLP to see if performance was sensitive to a change in this hyperparameter. The results are shown in Figure 4.6.

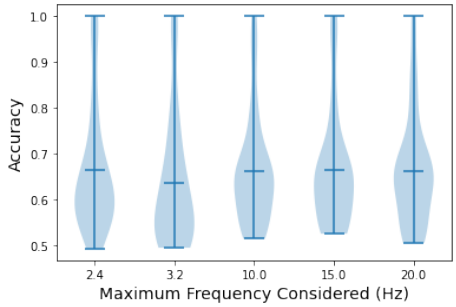


Figure 4.6: Domain-agnostic performance whilst varying the maximum frequency of the power spectrum used as features in the dft-mlp model. We see that the performance is not significantly sensitive to the maximum frequency considered.

4.4.2 Using more than one domain

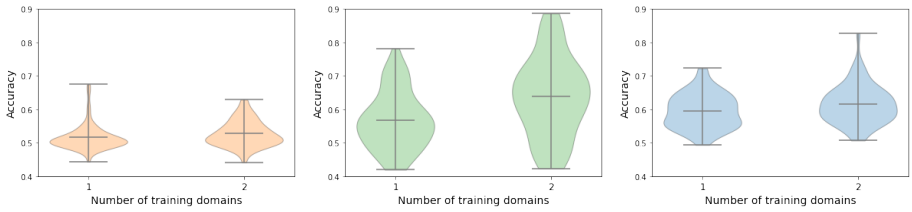
If our assumption that the datasets are connected by the same data generating mechanism is true, we should in theory improve domain-agnostic performance by training on more than one domain. In this section we investigate this idea.

Setup. The models were trained and evaluated according to Eq 4.1 as before. However, instead of using only one training dataset $\mathcal{D}_i$, two were used. i.e., $\boldsymbol{\mathcal{D}_{tr}} = \{\mathcal{D}_{tr,1}, \mathcal{D}_{tr,2}\}$. A small random sample of 128 windows is selected from $D_{tr,2}$ and included with the original training dataset $D_{tr,1}$ halfway through training. Performance is then measured on the remaining unseen dataset. The reason for such a small sample from the second training domain is to see the effect of performance based on data *diversity* rather than the effect of data quantity. The exact number of samples (128) is arbitrarily chosen as a relative

small number, compared to the size of adaptation sets in transfer learning which ranged up to thousands of samples [66]. The learning rate was 0.05, selected as before, with a rate scheduler which halved it every 2k iterations. The total number of iterations was 5k, as training for 10k showed signs of overfitting with increased training and validation loss.

The results comparing single domain to two domain training is shown in Figure 4.7. A similar statistical test is performed across all models to test whether the average performance using one or two domains are different with p-values of 0.139, $5.75\text{x}10^{-4}$ and $6.75\text{x}10^{-2}$ for Deepconvlstm, Deepconv and the DFT-MLP model respectively.

Additionally, it is interesting to note that when the small sample from the second domain is introduced we see a noticeable drop in validation loss in the original training domain, $\mathcal{D}_{tr,1}$. This is shown in the bottom portion of Figure 4.8. This suggests that the additional data diversity provided by the second domain also increases performance not only in unseen domains, but also in seen domains (excluding itself). We additionally observe that performance in its own domain also improve (as expected), but without using any transfer technique.

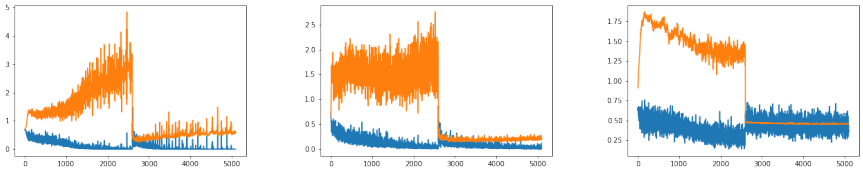


(a) deepconvlstm (b) deepconv (c) dft-mlp

Figure 4.7: Overall performance of the *unseen* dataset based on training with one or two domains across all considered models. p-values for the difference in means are 0.139, $5.75\text{x}10^{-4}$ and $6.75\text{x}10^{-2}$ respectively.

4.5 Discussion

We have made a case for a new way to evaluate HAR models that better align with performance under a variety of different real-world scenarios. We have done this by arguing that we should think of HAR datasets as being generated from a common underlying mechanism (SCM) which takes into account user, sensor and environment characteristics which may influence the observed data, as illustrated by Figure 4.1. It is important to note the very plausible limitation that the mechanism in Figure 4.1 is not able to represent all HAR datasets. There may be some other factors that are missing, or there may be influence between, or within, different groups of factors which may not be represented



(a) deepconvlstm (b) deepconv (c) dft-mlp

Figure 4.8: All models see a noticeable drop in validation loss (orange) on the original training domain $\mathcal{D}_{tr,1}$, when a small sample of data from an additional domain $\mathcal{D}_{tr,2}$ is introduced. Training loss is shown in blue. The training and validation loss is based only on data from $\mathcal{D}_{tr,1}$.

here. Regardless of the actual shape of the mechanism, this perspective allows us to define precisely what we mean by domain-agnostic performance, and this corresponds to robustness as discussed in the causal view.

From this foundation, we propose a benchmark to measure domain-agnostic performance. This is meant as the simplest starting point, since many of the confounding characteristics are as fixed as possible. Current state-of-the-art deep models in the literature were then evaluated.

The proposed benchmark is far from perfect. Ideally, the task should be multiclass classification over as many datasets as possible. Few publicly available HAR datasets have overlapping classes with the same sensor location. This is nonetheless a decent start, as models which perform well on multiclass should also do well in binary classification. In the future we hope that this benchmark can be extended when more data is available to the community.

The common data generating mechanism perspective also provides two immediate ways which should in theory improve domain-agnostic performance. The first is using an appropriate inductive bias, which in this case corresponds to picking features that closely relates to the causal variables of the label, using our knowledge of the problem. In this specific problem we know that the Fourier coefficients in the low frequencies likely contain information about the general shape of the activity, whilst discarding potential confounding information such as phase or amplitude. The second is increasing data diversity by training with more than one distribution, which as we've discussed in Chapter 3, the more intervention distributions the model performs well under, the closer is should be to a robust model.

On the first point, we implement a simple DFT-MLP model and have shown that the performance across all train-test combinations perform at least as well as the latest model in literature, whilst being 10-100 times faster in training on commodity hardware. We additionally provide evidence that this performance

is not sensitive to the maximum frequency considered, a hyperparameter. It is entirely possible that a better model can be designed given that more time is spent. However, the point of the work is to demonstrate that using an appropriate inductive bias by considering the data generating mechanism can easily help design better domain agnostic models, and at the same time increase efficiency. The point is to revisit the assumption that deep and larger models are always better in the context of HAR, especially when quality data is scarce, and efficient deployment is critical.

On the second point, we conducted an experiment that introduced a small random sample from an additional domain during training. The results suggests that this increased data diversity is beneficial to performance on unseen domains, as well as seen domains which aren't from the sampled domain. The question of whether increasing diversity always improves other domains performance remains an open question, in addition to how this relates to the negative transfer phenomenon seen in some transfer learning approaches. Characterising when additional domains are beneficial in terms of the SCM is an interesting area of future work.

4.6 Conclusion

The aim of the work in this chapter was to present the case for training models which are domain-agnostic, i.e., generalises to unseen domains of the same activity. This will bring us closer to robust real-world deployment of HAR models. To do so we have presented three main points.

First, we proposed using the leave-datasets-out cross-validation regime, which we argue is a better way to measure domain-agnostic performance than current evaluation methods, and better corresponds to real world performance. The LDO regime corresponds to measuring performance on multiple intervention distributions of an assumed common SCM. We present a starting point of this in the HAR context using a binary classification across three publicly open HAR datasets. We evaluate current state-of-the-art deep models for HAR, and find that they face significant performance degradation when tested against this new benchmark. We then show that by using a simple inductive bias from our knowledge of the problem, we can instead use a model that achieves similar, if not better performance than current deep models ($p=5.75\text{x}10^{-4}$ and 0.131) that is 10-100 times faster to train. The inductive bias corresponds to picking features which we believe are close to causal features of the label, and by using only specific features this reduces the size of the hypothesis class, decreasing the chance of the model learning spurious features. Finally, we show that training with even a small amount of data from an additional domain improves performance

on unseen, seen (excluding the same domain), and in the same domain without complex transfer or adaptation techniques, across all models and datasets considered. This aligns with our understanding from the causal view that using multiple interventions for training should be helpful for robustness.

These results suggest that end-to-end deep models may not always be the correct choice in HAR applications where large-scale training data is hard to come by, when deployment is likely to be on resource constrained devices, and where real deployed models face multiple sources of heterogeneity.

In the next chapter, we move away from evaluating robustness of deep models in a specific application. We explore a general way to improve robustness in low-dimensional regression settings by selecting subsets of data. Similar to the work in this chapter, the method of selection is informed by the causal view of robustness.

Chapter 5

Invariant exact matching - less data can be better for robustness

In the previous chapter, using the causal view we proposed an evaluation setup that could better reflect model robustness, and evaluated deep models in the HAR literature. The evaluation results suggest that large end-to-end deep models may not be a one-size-fits all solution to all HAR applications, as a simpler model is shown to be at least as robust as the SotA deep models. In this chapter, we explore an additional implication of the causal view - that training using less data can be beneficial to robustness under certain conditions. This is contrary to the popular belief that more data is always beneficial.

Specifically, in Chapter 3 we have seen how to define robustness with respect to intervention distributions of a common SCM. We have discussed that non-spurious variables under intervention (usually the parents) are ideal features for robust prediction. However, identifying them is challenging due to deficiencies in causal discovery - it is computationally inefficient in high-dimensions, and only recovers graphs up to Markov equivalence. To side-step causal discovery, we can instead exploit data heterogeneity - having access to different intervention distributions. In the previous chapter we have seen empirical evidence of non-robustness of real models in the application of HAR, by testing on different intervention distributions. In this chapter we investigate the idea that training on a selected subset of data from these different intervention distributions could lead to more robust models.

Outline. We introduce the problem of training a robust model in the context of the domain generalization task. We revisit the causal view of robustness and

explicitly state the assumptions, which informs the experimental setup in this chapter. We then cover the inspiration behind the proposed data selection procedure - matching for causal inference. We discuss the intuition behind invariant exact matching, in particular using the causal view of robustness. The experimental setup and procedure is described, both using synthetic and semi-synthetic data. The results are presented compared to normal ERM, IRM training and training with a random subset of equal size, showing that less data can train more robust models under certain conditions. Natural extensions to this idea are discussed to conclude the chapter.

5.1 Introduction

A fundamental assumption in supervised learning is that the training and test samples are drawn independently and identically from the same distribution. However, in practice, the distribution of the test data often differs from the training data [83]. This can be seen across a range of applications from computer vision [84], time series analysis [85], to medical applications [86]. Supervised models which use ERM consider fitting the training data as a proxy for the test distribution [64]. It is then no surprise that these models suffer significant performance degradation when the test distribution differs from the data seen at training. Collecting data for all possible test distributions (often called *domains*) is costly and often impossible. The task of *domain generalization* aims to train models that perform well in unseen test domains.

The main challenge for domain generalization is that the test domain is unknown at training time [57]. This makes it distinct from closely related problems such as domain adaptation [87] and transfer learning [88], which also study the problem of distribution shift from training to test time; the crucial difference being that they assume access to samples from the target domain, to some extent, during training. Adaptation and transfer methods target specific test domains, implying the need to retrain for each new domain; domain generalization aims to be domain-agnostic. Of course, it is impossible to generalize to arbitrary test domains, and methods must exploit some structure that is assumed to be present at test time.

There are currently two broad categories of strategies in domain generalization. One is to increase training set diversity by either collecting or synthesizing more data [44]. The other is to use models or training methods which promote certain inductive biases thought to be helpful for generalization [89].

Regardless of the approach taken, there is a general consensus that the more training data one possesses, the better the model generalizes. This work challenges this notion by demonstrating that using a selected subset of data can in

fact match, or even improve, generalization performance when compared with training on all available data. This has implications for our understanding of the domain generalization problem.

Contributions. In this chapter we study regression settings in domain generalization, where we have access to more than one training domain. We use Structural Causal Models (SCM) to formulate the problem, which allows us to state our assumptions explicitly in terms of graphical restrictions. By inverting the idea of a popular procedure for causal effect estimation, we propose a way to select subsets of data that, when used to train models, induce said model to use features that are likely to remain predictive of the target variable across domains, whilst making few assumptions about the causal structure.

We assess this method empirically using unseen test domains first on synthetic and then semi-synthetic data. We find the surprising result that when tested against unseen test distributions, filtered data up to 90% smaller than the original dataset can show either similar or better performance when compared to training on all available data.

5.2 Related work

There are many works in the domain adaptation and transfer learning literature which consider a similar problem of distribution shift from training to test time [88][87], some we have seen which are specific to the HAR application. As previously mentioned in Chapter 4, the problem setup in domain generalization (called domain-agnostic models in the context of HAR) is slightly different. The key is that there is no assumption of access to data from the target domain. We therefore restrict ourselves to covering related works in this area.

The problem of domain generalization has been studied from different perspectives; see [90] for a survey. Broadly there are two categories of approaches, *i*) Generating new or augmenting existing training samples to increase the diversity in the training data [84], or *ii*) using models and training methods which promote certain inductive biases. For instance, learning domain invariant representations [91] typically using a suitable regularizer, or adapting a learning strategy more amenable to generalization: learning an ensemble of models [92] or meta learning [93]. The work in this chapter looks at domain generalization by selecting certain subsets of data, informed by the causal view of robustness.

We have discussed the relationship between domain generalization to the causal view of robustness in Chapter 3. In short, 'domains' can be precisely defined as different intervention distributions of a SCM. A model which performs well on domain generalization is one which should do well across these different

distributions i.e., a robust model. Assuming interventions on all variables in the SCM are possible during testing except for on the target variable, this robust model should use the parents of the target variable. As we discussed in Chapter 3, since the causal graph is unknown in most scenarios, the most practical approach is to exploit data from different intervention distributions. In the previous chapter we used intervention distributions to evaluate robustness for existing models, and used our knowledge about the application domain (HAR) to select features which should be more robust. In this chapter we extend the idea of using different intervention distributions further and show that certain datapoints may be more useful than others in training robust models. This is achieved by inverting the idea of a well-known causal inference method - matching.

A different type of matching has been studied for domain generalization in computer vision. Mahajan [94] used a representation learning based procedure to improve performance on classification problems using a contrastive loss function, by matching objects of the same class across domains. Here, we do not work with high dimensional image data, and make different assumptions about the causal structure. However, extending the method to higher dimensional data is a worthwhile future direction.

5.3 Background

This section describes the problem setup, first using the language of distributions, and then using SCMs, introduced in Chapter 3. We will see that using SCMs instead of distributions allow us to talk about the link between training and unseen test domains more precisely, and makes our assumptions explicit. We then recap matching: a common procedure for causal effect estimation, first introduced in Chapter 2, before describing invariant exact matching.

5.3.1 Problem setup

The following formalizes the problem setup. First without using SCMs. The domains $\mathcal{D}_i \sim \mathbb{P}$ are sampled from a distribution $\mathbb{P}$. Each domain is again a distribution from which we sample a dataset $D_i \sim \mathcal{D}_i$. When $\mathbb{P}$ is fixed, that is the same domain gets sampled all the time, the setup produces classical *i.i.d.* datasets. A dataset i consists of tuples $(\mathbf{X}^{(i)}, Y^{(i)})$. The domains in $\mathbb{P}$ are assumed to be related because the data generating process for the target Y remains fixed across all domains:

$$Y^{(i)} = f(\phi(\mathbf{X}^{(i)})) + \varepsilon \tag{5.1}$$

Here f denotes the unknown data generating function, ε denotes the observation noise, $\phi(.)$ represents a subset operation on the features. That is, some features determine the value of Y. Equation 5.1 describes a primary assumption that the target variable is a function of only the causal subset of the features. The learning problem can be formalized as finding θ for some approximation of the data generating function f_θ, such that the following is minimised,

$$\mathbb{E}_{\mathbf{x}\sim\mathcal{D}_i}[l(f_\theta, f, \mathbf{x})] \tag{5.2}$$

Where $\mathcal{D}_i$ is some unseen domain, and l is some loss function, for example the mean squared error. The error on a domain can be approximated by the error on a dataset D_i sampled from the domain $\mathcal{D}_i$, so the error is $l(D_i, \theta) = \frac{1}{n_i}\sum_{j=1}^{n_i}(f_\theta(X_j^{(i)}) - Y_j^{(i)})^2$, where dataset D_i has size n_i. This is equivalent to the ERM setup described in Chapter 3, in the supervised learning context.

Without any assumption on $\mathbb{P}$ it is impossible to solve Equation 5.2 [95]. Here we assume that the distribution of $\mathbf{X}$ can change across the domains, but the parameters in Equation 5.1 are consistent across domains.

Using SCMs will allow us to talk about precisely how the different domains $\mathcal{D}_i \sim \mathbb{P}$ relate, and this is discussed next. For general context on SCMs, we defer to [60] and Chapter 3.

Assume there is a SCM $S = (\mathbf{X}, \mathbf{f}, \mathbf{N})$, where $\mathbf{X}$ is the set of all variables, $\mathbf{f}$ the set of structural assignments, and $\mathbf{N}$ the set of independent noise terms. For consistency, we denote by Y the single variable in $\mathbf{X}$ which we consider the target. S implies a distribution over $\mathbf{X}$, the natural distribution, which is a distribution in $\mathbb{P}$.

An intervention I is a modification of one or more of the structural assignments in $\mathbf{f}$, which results in S_I, a modified version of S, and therefore a different distribution $\mathcal{D}_i$ over $\mathbf{X}$. We call this an *intervention distribution* on S induced by I. The different interventions possible on the underlying SCM captures the different possible distributions $\mathcal{D}_i$ in $\mathbb{P}$. In this work, we assume that all interventions do not change the structural assignment of the target, Y.

The data generating process for the target Y in eq 5.1 is the mechanism f_Y in the SCM. The learning problem remains the same as in eq 5.2. Practically, there are m training datasets, $D_1^{tr}, D_2^{tr}, ...D_m^{tr}$ each of which are $i.i.d$ samples from a different intervention distribution on some common SCM S. We would

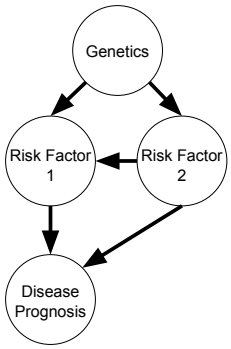


Figure 5.1: An example SCM illustrating how the assumptions map to a certain domain generalization task - predicting disease prognosis. The assumption that there is no intervention on Y reflects the fact that the way the risk factors determine prognosis in nature do not change, even when the way in which the risk factors themselves affect each other, or are distributed, may change in different populations.

like to learn a predictor $f_\theta(\mathbf{X} \setminus Y)$ such that the error on datasets sampled from *unseen* intervention distributions from S, $D_1^{test}, D_2^{test}, ..., D_k^{test}$ is minimised.

The assumption that there is a common SCM S, along with the restriction that the intervention cannot be on the structural assignment of Y captures the idea that there is some inherent structure to the unseen test domains which also exists in the training domains, even though they have different distributions. Consider predicting the outcome of some natural process, e.g., disease prognosis, we assume that the way in which the risk factors affect the prognosis remains constant, even when the distributions of the risk factors themselves may change in different populations. Figure 5.1 illustrates this example.

For simplicity, we also assume that all the parents of Y are observed; in the running example, that we observe both the risk factors that determine prognosis. If we only observed one, we are missing information to make correct predictions on the prognosis. Although this assumption can be relaxed in future work.

In summary, this work makes the following assumptions:

1. There is a common SCM S that is consistent across domains i.e., each data distribution is drawn from an intervention distribution implied by S_I, which is obtained by applying some intervention I on S.

2. S must be a DAG.

3. The parents of the target Y are observed in the covariates $\mathbf{X}$.

4. Each intervention I does not change the structural assignment of Y.

These assumptions are mild and not restrictive enough to be impractical.

5.3.2 Exact matching

Causal effects are usually studied in a randomized controlled trial (RCT). This is covered in detail in Chapter 2. In brief, in the simplest case this means units in the population are randomly assigned a 'treatment', and the outcome is compared between those who did and did not receive treatment. This yields a valid effect because treatment was randomised, and therefore any factors which could have also affected the outcome, called confounders, are balanced between the two groups.

When data does not come from a RCT, it is called an observational study. Matching is used to estimate the effect of some treatment on an outcome in observational studies, briefly introduced in Chapter 2. It is discussed in more detail here. The point of matching is to restore the balance in the confounders between both groups, and therefore make the outcomes for those that received treatment comparable against to those that did not when the treatment is not randomised.

In its simplest form, matching [28] is achieved as follows. 1. Given a SCM, we identify the confounders, $\mathbf{X}_{conf}$ which needs to be balanced ('requires adjustment'), using for example Pearl's backdoor criteria [25]. Then, for each unit, i in the population, we find another unit j with the exact same values for $\mathbf{X}_{conf}$, but which have received a different treatment. If an exact match cannot be found, one can instead find pairs such that dist(i, j) is minimised for some distance metric *dist*. This metric can for example be define as

$$dist(i,j) = \frac{||\mathbf{X}_{conf,i} - \mathbf{X}_{conf,j}||_1}{||z_i - z_j||_1}$$

$\mathbf{X}_{conf,i}$ are the covariates that require adjustment for unit i, and z_i is the treatment received by unit i.

Intuitively, for each unit, we are interested in finding an 'identical twin', or that closest to one, that has received a different treatment. The causal effect of the treatment on the outcome is then the difference in outcome between each unit in a matched pair, divided by the difference in the value of treatment of the units in said pair. Matching requires three assumptions, exchangeability, positivity, and consistency. For more details, please see Hernan et al. [28].

5.4 Invariant exact matching

Matching, described previously, aims to balance the set of confounders between two subgroups of data, and serves as an inspiration for the method proposed in this chapter. Here we describe the intuition behind invariant exact matching.

Intuition. Consider two datasets corresponding to two different interventional distributions of a common SCM S. Denote the datasets by A and B. Consider the SCM S_A which entails the distribution from which A is drawn. Without loss of generality, we can think of dataset B as being obtained by some intervention I on S_A to produce S_B with entailed distribution B. We can think of intervention I as a treatment on a control population A, which may affect both spurious (X_{conf}) and causal covariates (X_{cau}), and hence the outcome variable Y, to produce B.

Now consider matching datapoints which have large differences in Y, but the smallest possible difference in $\mathbf{X}$. i.e., define the distance between two data points as $d((X_i^{(A)}, Y_i^{(A)}), (X_j^{(B)}, Y_j^{(B)})) = \frac{||X_i^{(A)} - X_j^{(B)}||}{||Y_i^{(A)} - Y_j^{(B)}||}$, where $||.||$ denotes the L_1 norm, and where i and j are different datapoints from datasets A and B. If we match two datapoints in the same dataset, a large difference in Y would imply a large difference in X_{cau} while X_{conf} may also have large distance. As it is matched to values drawn from the same distribution. However if two datapoints which belong to different datasets A and B are matched, a large difference in Y would still imply a large difference in X_{cau}, but X_{conf} variables can be matched where they may be drawn from different distributions. This will be the case when the intervention I, which takes S_A to S_B acts on variables in X_{conf}. This means if we have access to many datasets, which correspond to many different interventions, for any particular datapoint, we can expect to find another datapoint from a different dataset where the value of X_{conf} is similar. However, since the mechanism of how X_{cau} determines Y doesn't change across datasets (assumption 1), and the matching forces Y to have a large difference, the X_{cau} variables in a matched pair will remain far apart. In short, if we force Y to be different and $\mathbf{X}$ to be similar, the subset of $\mathbf{X}$ which determines the value of Y (X_{cau}) will in general have a large difference in each matched pair. Using a subset of data where X_{cau} has large difference when Y has large difference and X_{conf} has small difference should encourage the model to use variables in X_{cau} to predict Y. No additional assumptions are made about the structure of the underlying SCM except those listed at the end of section 5.3.1.

If we have additional information on the shape of the causal graph, we could apply Pearl's backdoor criteria and only use those features which require adjustment. Since in most scenarios we do not know the underlying causal graph, we include all features.

We define the weight of a pair of datapoints as the inverse of the distance, $\frac{1}{d(.)}$. Further, a small number $\delta > 0$ is used to prevent division by zero. The weight of a pair of datapoints from two datasets D_A and D_B (with $A \neq B$) is therefore given by:

Algorithm 1 Data filtering using invariant matching

Input: $D_1^{train}, D_2^{train}, ...D_m^{train}$
Parameter: Threshold value $T \in \mathbb{R}$
Output: A subset of data $D_{IM} \subset \cup_{i=1}^{m} D_i^{train}$

1: Initialise a graph $G = (V, E)$
2: **for** pairs $(i, j) \in \cup_{k=1}^{m} D_k^{train}$ **do**
3: Calculate $w(i, j)$ as in Eq. 5.3
4: $V := V \cup (i, j)$
5: $E := E \cup (i, j)$ with $weight = w(i, j)$
6: **end for**
7: Find a maximal weighted matching M on G
8: **return** datapoint pairs in M with $w(i, j) \geq T$

$$w((X_i^{(A)}, Y_i^{(A)}), (X_j^{(B)}, Y_j^{(B)})) = \frac{||Y_i^{(A)} - Y_j^{(B)}||}{||X_i^{(A)} - X_j^{(B)}|| + \delta} \tag{5.3}$$

Once we have the weights of all possible sample pairs in all possible training dataset pairs ($(D_A^{train}, D_B^{train})$ for $A \neq B$ and $A, B \in [1, m]$, where m is the number of training datasets), we can construct a graph where each node represents a datapoint, and the edge weight between a pair contains $w(.)$. We then find the maximal weighted matching M on said graph. This is, in short, a list of datapoint pairs $M = \{(i_1, j_1), (i_2, j_2), ...(i_n, j_n)\}$ where each datapoint can appear only in one pair $i_1 \neq i_2 \neq ...i_n \neq j_1 \neq j_2... \neq j_n$ and the set of pairs, M, maximises $\sum_{i,j \in M} w(i, j)$. This is the bottleneck to the selection procedure as matching has a time complexity of $O(n^3)$. This can be solved using for example, Edmond's blossom algorithm [96].

Next we select all data point pairs $(i, j) \in M$ such that $w(i, j) \geq T$ for a given threshold T. This subset of data is then used to train a model. The procedure is summarised in Algorithm 1.

The value of T chosen will depend highly of the distribution of weights in the matched pair, and will be application specific. The experiments below choose the thresholds as percentiles of the weight distributions. The results here may improve with optimal choices of thresholds.

5.5 Experimental setup

We now describe the experiments on synthetic and semi-synthetic data. There are two main difficulties with using real data in experiments. First, we can never be certain whether the assumptions required for the method is satisfied

i.e., we do not know the causal graph, and therefore whether we have observed all relevant variables. This adds an extra dimension of complexity in evaluating the method.

Second, synthetic data allows control not only on the assumptions but additionally gives the ability to vary parameters to study the effects on the proposed method. This offers the advantage of testing on a wider range of settings. This is commonly used in literature [97][98], and is what is followed here. Whilst there are benchmarks for domain generalisation in image data [57], the proposed method currently works for tabular data, for which to the best of our knowledge, does not currently have a benchmark.

For the first experiment, synthetic features for each training and testing distribution, along with the outcomes, are generated. For the second experiment, real feature data is used, and only the outcome variable is generated. Additionally, since the real dataset only contains one domain, additional domains, both for training and testing, are generated by applying synthetic interventions to an assumed SCM.

5.5.1 Synthetic data

An SCM S with 5 normally distributed features and one target variable is used, shown in 5.4. There is only one causal variable, X_1, and 4 spurious variables. Spurious variables have the same distribution, and similar to the cause (X_1) and target (Y) such that it is not immediately obvious which variable is causal.

$$\begin{aligned} X_1 &:= Normal(10, 2) \\ X_2 &:= Normal(10, 5) \\ X_3 &:= Normal(10, 5) \\ X_4 &:= Normal(10, 5) \\ X_5 &:= Normal(10, 5) \\ Y &:= X_1 + Normal(0, 5) \end{aligned} \tag{5.4}$$

A list of 3 interventions on the noise are applied to create data from 3 different environments for training. The interventions are listed in 5.5, one per line. The corresponding variable in 5.4 is changed for each line in 5.5, which then becomes a new environment, and 200 samples are generated for each environment, totalling 600 samples for training. Using more interventions is computationally expensive to find an optimal matching, and environments are restricted to 3 for this reason. It is likely (as results in Table 5.4 suggest) that more training interventions improves robustness. The interventions are minimal

and not drastic, as this will easily give away the causal variable.

$$
\begin{aligned}
X_1 &:= Normal(10, 10) \\
X_2 &:= Normal(10, 10) \\
X_3 &:= Normal(10, 10)
\end{aligned} \tag{5.5}
$$

A separate, different set of 6 interventions are applied to create the 6 testing data distributions which modified both the mean and noise of a feature, listed one per line in 5.6, totalling 1200 samples. The interventions here are stronger compared to the training interventions, as we want to measure robustness in dissimilar conditions. The empty intervention means no variable in the original SCM was changed.

$$
\begin{aligned}
X_1 &:= Normal(5, 3) \\
X_2 &:= Normal(5, 3) \\
X_3 &:= Normal(5, 3) \\
X_4 &:= Normal(5, 3) \\
X_5 &:= Normal(5, 3) \\
&\text{Empty intervention}
\end{aligned} \tag{5.6}
$$

Invariant matching is performed at different threshold percentiles and the remaining subset of data is used to train a feed-forward network (MLP). There is no particular reason to choose one architecture over another, as the method concerns the data used for training. A simple architecture was used because they are quicker to train. The MLP architecture for synthetic data is:

$$
\begin{aligned}
&Linear(input = 5, 6) \\
&Linear(6, output = 1)
\end{aligned}
$$

The full dataset using both ERM and IRM, and a random selected subset of data with the same size as the matched subset is also used to train the same MLP architecture. All models are then tested on the generated unseen test distributions using median percentage error. For each threshold value, the experiment was repeated 10 times.

5.5.2 IHDP data

Instead of assuming features are distributed according to some standard distribution previously, features from a real dataset are used, which have a more natural distribution.

Outcomes were generated using these real features. This provides several advantages. This allows us to satisfy the assumption that the causal variable is included in $\mathbf{X}$ (Assumption 3), and that Y does not cause any variable in $\mathbf{X}$ (Assumption 2). We can then create different intervention distributions synthetically (Satisfying assumptions 1 and 4) for training and testing. This also provides the additional advantage of being able to use different parameters for generating the intervention distributions and the outcome, which means studying the method under a wider range of parameter values.

Dataset. The Infant Health Development Program (IHDP) data was first introduced by Hill et al. [99] to estimate average causal effects. The data came from a randomised experiment in 1985 which provided high-quality child care and home visits from a trained provider to the treatment group in order to assess its impact on cognitive test scores of treated children. The data consists of pre-treatment variables on the child, the mother, and behaviour during pregnancy, totalling 6 continuous and 19 binary covariates.

From the available features, four were randomly selected as $\mathbf{X}$ in the synthetic dataset. The outcome was generated in the same manner as the non-linear case in Hill et al. [99]:

$$Y = exp(\mathbf{X}\beta) + Normal(0, 1) \tag{5.7}$$

where coefficients β were randomly sampled from [0,0.1,0.2,0.3,0.4] with probabilities [0.6, 0.1, 0.1, 0.1, 0.1] respectively. Two distributions are then created for training by randomly picking a feature from the selected four, and multiplying by a random integer factor between -2 and 2. The same was performed again to create two different test environments.

For each run, a new set of features are sampled, a new set of parameters is sampled for generating the outcome, and new train and test distributions are generated.

Similar to the synthetic case, matching was performed on the two training environments, and pairs with weights under the threshold were discarded. The resulting subset was used to train a MLP. The architecture for IHDP data was:

Algorithm 2 Experiment procedure

1: **if** synthetic experiment **then**
2: Given an SCM S, generate $D_1^{tr}, D_2^{tr}, D_3^{tr}$ by applying three different interventions to S.
3: Generate $D_1^{test}, D_2^{test}, ..., D_6^{test}$ by applying six different interventions to S.
4: **else**
5: Randomly choose 4 covariates from IHDP data, and randomly sample parameters β to generate the outcome according to equation 5.7.
6: Randomly pick one covariate from the previously chosen 4, and multiply with a random integer between [-2, 2]. Repeat twice to create two training distributions D_1^{tr}, D_2^{tr}. Repeat twice again to create testing distributions D_1^{test}, D_2^{test}.
7: **end if**
8: Denote as D^{tr} the union of all training distributions $D_1^{tr} \cup D_2^{tr} \cup ...$ and similarly for D^{test}
9: Find matched pairs M using invariant matching on training data D^{tr}.
10: Using D^{tr}, filter for pairs with weight $>$ threshold T, denote filtered set M_T.
11: Randomly sample datapoints from D^{tr} with size equal to M_T without replacement, denote by D_{rand}^{tr}.
12: Train model f_{M_T} with subset M_T, f_{full} with full dataset D^{tr}, and f_{rand} with D_{rand}^{tr}
13: Test models f_{M_T}, f_{full} and f_{rand} on D^{test}

$$Linear(input = 4, 5)$$
$$Linear(5, output = 1)$$

The full dataset, and a randomly selected subset with the same size as the matched subset, was also used to train a MLP with the same architecture. The models were then tested on the two unseen test environments. For each threshold value, the experiments were repeated 10 times. The experimental setting is summarised in Algorithm 2.

5.6 Results

As testing using unseen test distributions incur very large outlier errors, all results are reported using the median percentage error over the unseen test distributions, averaged over 10 runs. The standard error (SE) is given in parentheses.

The results for the synthetic data experiment is shown in Table 5.1. It shows the median percentage error over the 6 unseen test distributions with varying

percentile thresholds T, averaged over 10 runs. The results are grouped by varying threshold values to unsure each method is compared using the same dataset; as for each threshold value, a new set of training/testing datasets are sampled from the corresponding distributions. We compare with training on the union of all training datasets (Normal ERM training), and a randomly sampled subset with the same size as the Invariant Matching (IM) subset (Random subset). We also compare with Invariant Risk Minimization (IRM) [50]. For the first set of experiments, which used 5 covariates, we find that models trained with the IM subset do better compared to other baselines across all threshold values, up to a 90% reduction in data size. However, when the number of covariates are changed between 3 to 6, with everything else kept constant, the results are not as clear cut, as shown in the top part of Table 5.2. Similarly, additional experiments were performed by using a polynomial (degree 2) and exponential generating function for y, and these results, which are also ambiguous, are shown in the top portion of Table 5.3.

Table 5.4, shows the same synthetic experiment but with changes in the number of training distributions, while the threshold was fixed at 60. Whilst we see that IM similarly outperforms, we note that with more distributions the median percentage error is lower across all methods, as expected, since there are more datapoints to choose from during matching.

Table 5.5 shows the results for the first set of IHDP experiments, with 4 covariates at varying thresholds. Since the IRM results are similarly poor as the synthetic experiments, they are omitted. The results with varying the number of covariates are shown in the bottom part of Table 5.2. Similar to the synthetic case, the results for changing the y generating function to a linear and polynomial (degree 2), are shown at the bottom of Table 5.3.

5.7 Discussion

Although we present some encouraging results on generalization and data efficiency, we note the following limitations. First, the matching procedure is computationally expensive, since we find the optimal match, which takes $O(n^3)$ time. Approximate matching methods can be explored for computational performance gains. Secondly, the behaviour of the matching procedure in high dimensions requires more exploration. We know for instance that the resulting match will be worse when we have lots of features, and this has been seen in the synthetic experiments with varying covariate numbers; one potential avenue is to match on propensity scores instead of on the features themselves. Crucially, we do not yet fully understand when this method yields significant improvements. While there are encouraging results for synthetic data, the results for

Method	Median error % (SE)	Training data size (%)	Threshold
Normal training	64.15 (4.582)	100%	N/A
Random subset	64.28 (4.317)	55%	N/A
IM subset	**57.10 (3.489)**	**55%**	40
IRM	91.84 (5.364)	100%	N/A
Normal Training	67.28 (2.262)	100%	N/A
Random subset	66.54 (2.592)	45.7%	N/A
IM subset	**59.67 (2.004)**	**45.7%**	50
IRM	99.74 (2.691)	100%	N/A
Normal Training	73.52 (4.042)	100%	N/A
Random subset	73.01 (4.352)	36%	N/A
IM subset	**63.58 (3.687)**	**36%**	60
IRM	96.50 (1.465)	100%	N/A
Normal Training	76.86 (5.246)	100%	N/A
Random subset	76.75 (5.098)	27.3%	N/A
IM subset	**65.64 (4.208)**	**27.3%**	70
IRM	92.43 (2.691)	100%	N/A
Normal Training	64.99 (4.738)	100%	N/A
Random subset	63.79 (5.196)	18%	N/A
IM subset	**55.81 (3.685)**	**18%**	80
IRM	92.44 (2.000)	100%	N/A
Normal Training	71.55 (4.200)	100%	N/A
Random subset	69.65 (4.349)	9.33%	N/A
IM subset	**61.96 (3.532)**	**9.33%**	90
IRM	98.40 (2.026)	100%	N/A

Table 5.1: Results from training with the matched (IM) subset, versus training with all data (Normal training), IRM training (IRM), and a randomly sampled subset of data with the same size as the matched subset (Random subset). Results for synthetic data. For each group of results where the threshold is varied, a new set of training/testing environments are sampled; methods should be compared within the same group as they use the same data.

IHDP are less so - why is this the case? Why is it that linear functions of y work best in the synthetic experiment and not in the IHDP experiment? Why is it that linear and polynomial functions of y do not see the same improvements as the exponential function of y? It may be that for each y generating function, an appropriate threshold value has to be picked, or it may be that appropriate hyperparameters need to be selected for each new scenario. In the additional experiments that varied covariates and y generating functions, the same hyperparameters were re-used, and the threshold value (60th percentile) was picked arbitrarily for simplicity.

In terms of the experiments, it is no surprise that the performance depends

Method	Median error % (SE)	# Covariates	Threshold
Normal training	79.47 (6.680)	3	N/A
Random subset	80.78 (6.620)	3	N/A
IM subset	**71.01 (5.661)**	**3**	60
IRM	99.27 (5.523)	3	N/A
Normal Training	61.07 (5.209)	4	N/A
Random subset	59.69 (4.807)	4	N/A
IM subset	**58.40 (4.720)**	**4**	60
IRM	94.45 (2.756)	4	N/A
Normal Training	**64.60(4.926)**	**6**	**N/A**
Random subset	65.19 (5.335)	6	N/A
IM subset	65.07 (5.567)	6	60
IRM	95.92 (3.570)	6	N/A
IHDP Experiment			
Normal Training	59.79 (5.890)	3	N/A
Random subset	**59.88 (5.555)**	**3**	N/A
IM subset	63.06 (6.162)	3	60
Normal Training	66.37 (6.207)	5	N/A
Random subset	**65.47 (6.598)**	**5**	N/A
IM subset	67.63 (6.097)	5	60
Normal Training	53.29 (2.647)	6	N/A
Random subset	**52.31 (2.383)**	**6**	N/A
IM subset	57.81 (3.164)	6	60

Table 5.2: Results for synthetic data whilst varying the number of covariates, threshold 60.

on the class of interventions chosen, the SCM, and the outcome generating function. Whilst we have attempted to minimise bias by randomly sampling aspects of the experiment, and varying different aspects, it is imperfect. It is infeasible to test all configurations, and there is no reason to choose any particular one; the method will have to be evaluated according to the nature of the intended application. A more theoretical analysis of the method has been attempted but remains an open question. There is the question of why IRM performs so poorly in these experiments, although the superiority of ERM to IRM have already been documented elsewhere [100] in addition to our experiments. Examining why this is the case is another interesting direction for future work.

The results here largely agree with the causal view of robustness, as well as the results from Chapter 4, that using multiple intervention distributions for training can improve robustness. The results also demonstrate empirically, in addition to the results in Chapter 4, that using multiple intervention distribu-

Method	Median error % (SE)	y function	Threshold
Normal training	97.47 (1.286)	deg.2 polynomial	N/A
Random subset	97.06 (1.334)	deg.2 polynomial	N/A
IM subset	**97.04 (1.334)**	**deg.2 polynomial**	60
IRM	100.02 (0.7463)	deg.2 polynomial	N/A
Normal training	99.99 (0.0048)	e^x	N/A
Random subset	99.99 (0.0049)	e^x	N/A
IM subset	99.99 (0.0049)	e^x	60
IRM	**99.99 (0.0047)**	e^x	**N/A**
IHDP Experiment			
Normal training	**101.52 (2.518)**	x	N/A
Random subset	101.64 (2.637)	x	N/A
IM subset	101.64 (2.192)	x	60
Normal training	**101.84 (1.776)**	**deg.2 polynomial**	N/A
Random subset	102.12 (1.827)	deg.2 polynomial	N/A
IM subset	102.80 (1.994)	deg.2 polynomial	60

Table 5.3: Results for synthetic data whilst changing the function that generates y, threshold 60.

tions is a practical way to improve robustness, as argued in Chapter 3. However, the experiments in this chapter also further suggest that certain datapoints may be more informative when the goal is to train a robust model. This opens up a ripe new area of further research. For instance, a measure of this 'information' is an interesting direction for future work. A similar idea that throwing away data improves worst-class error in classification has recently been reported [101], after the experiments conducted here were conceived and implemented. We suspect that a theoretical understanding of this method will be beneficial to understand the domain generalization problem as a whole.

5.8 Conclusion

We propose a simple data selection method which uses few assumptions about the causal structure. Experiments on synthetic and semi-synthetic data shows the counter-intuitive result of improved performance whilst simultaneously using less data on unseen intervention distributions, under some conditions. The exact circumstances of the data generating process, the observed dataset, and the resulting matching on improvements to robustness remains to be fully understood. We note the potential for this method to be extended to work with higher-dimensional and non-tabular data, and in helping us better understand the problem of domain generalization.

Method	Median error % (SE)	Training data size	Training distributions
Normal training	66.49 (4.763)	100%	2
Random subset	66.05 (4.747)	36%	2
IM subset	**60.58 (4.314)**	**36%**	2
Normal training	73.52 (4.042)	100%	3
Random subset	73.01 (4.352)	36%	3
IM subset	**63.58 (3.687)**	**36%**	3
Normal training	63.29 (4.888)	100%	4
Random subset	63.75 (5.194)	36%	4
IM subset	**56.21 (4.193)**	**36%**	4
Normal training	55.14 (3.502)	100%	5
Random subset	54.22 (3.354)	36%	5
IM subset	**46.98 (1.983)**	**36%**	5

Table 5.4: Performance as the number of training distribution increases, threshold 60, synthetic data.

Method	Median error % (SE)	Training data size	Threshold
Normal training	61.15 (2.054)	100%	N/A
Random subset	61.04 (2.102)	60.0%	N/A
IM subset	**58.49 (1.327)**	**60.0%**	40
Normal training	56.99 (3.575)	100%	N/A
Random subset	57.03 (3.582)	49.9%	N/A
IM subset	**54.87 (3.378)**	**49.9%**	50
Normal training	60.62 (1.522)	100%	N/A
Random subset	60.61 (1.663)	40.0%	N/A
IM subset	**57.71 (1.241)**	**40.0%**	60
Normal training	58.53 (2.508)	100%	N/A
Random subset	58.01 (2.254)	30.0%	N/A
IM subset	**55.93 (2.253)**	**30.0%**	70
Normal training	60.42 (2.317)	100%	N/A
Random subset	60.18 (2.630)	19.9%	N/A
IM subset	**57.58 (2.216)**	**19.9%**	80
Normal training	59.87 (3.417)	100%	N/A
Random subset	60.54 (3.452)	10.1%	N/A
IM subset	**56.63 (2.818)**	**10.1%**	90

Table 5.5: Results for matched (IM) subset versus training with all data (Normal training) and a randomly sampled subset of data (Random subset) for IHDP data.

We note that the preliminary findings in this chapter support certain aspects of the causal view of robustness, as well as provide additional empirical

evidence that exploiting multiple intervention distributions can be a practical way forward in many settings, as argued in Chapter 3. Additionally, by showing that some datapoints can be more informative under certain conditions, it opens up a new research agenda of understanding how to efficiently use data from different intervention distributions to train robust models.

Chapter 6

Conclusion

6.1 Summary

At the beginning of the thesis we looked at the different strands of research related to model robustness and asked how causality can help us better understand and train robust models. It was noted that causality could provide a unifying way to view the various effects of non-robust models, and is consistent with many other hypotheses proposed in literature. In particular, issues related to model misclassification in the presence of changes such as lighting, weather, object placement, adversarial perturbations, style, amongst others, can be viewed as models unable to predict on different intervention distributions of the same SCM, due to using non-causal features.

However, it is not immediate how works in the causal literature can be used to train more robust models. In particular, for low dimensional data inferring causal effects require knowledge of the causal graph at least partially, and inferring the causal graph is only feasible up to Markov equivalence. In high dimensional data these two tasks become even harder due to computation efficiency, but the question whether the causal graph formulation is even appropriate still remains.

In the first contribution, a theoretical view of the connection between causality and model robustness is developed, which in turn informed the direction of later work. As a starting point, the idea that even though data may not be from the same distribution, they can still be related by having the same underlying causal structure is introduced. The relationship between prediction robustness and causal parents, and in general how knowledge of the causal graph can aid robustness, is discussed. However, it is argued that applying causal discovery is imperfect, and estimating the causal effect of each variable requires some knowledge of the causal graph. Whilst we know that using the causal parents will

generate robust predictions, the reverse turns out to also be true. Peters [18] showed that finding robust predictors can allow us to recover the causal parents, using multiple statistical hypothesis tests. However these tests are impractical with a large number of variables. We then looked at Empirical Risk Minimization [64], the dominant learning paradigm today, and its theoretical justification in the form of the Probably Approximately Correct [58] model of learning. PAC learning shows that for finite hypothesis classes, there exists a number of samples such that the test performance is bounded given a low training loss, given that training and testing data is iid. It is natural to then ask about the case where train and test data is non-iid, a realistic scenario in real deployments. Combining the ideas behind invariant causal prediction (ICP) and ERM, we introduce empirical causal convergence, which connects prediction to causality; it suggests that optimising only for prediction performance in diverse intervention distributions will move a model closer to the causal mechanism.

After establishing this foundation, we turn our attention to a particular type of model - those used in human activity recognition. We look at the models being developed for HAR since 2003, and note that, due to the large number of model parameters, current HAR models are at risk from non-robustness. We test this by proposing a new benchmark to measure model robustness using the prediction performance of the same task over three datasets under similar conditions. After establishing that the models do indeed suffer from a lack of robustness, an illustrative fix is proposed by using a simpler model with an appropriate inductive bias. It is shown that on this benchmark, this simpler model performs at least as well as the two SotA models, whilst being at least 10-100 times faster to train. According to the causal view, it is expected that robustness should improve when multiple datasets from the same SCM are used in training, and this effect is also seen in the experiments.

Finally, we return to application-agnostic techniques but narrow the scope to regressions on tabular data. We first consider why randomised controlled trials allow us to estimate causal effects and how this is achieved using a common causal inference technique - matching. Matching allows us to estimate the causal effect by balancing the confounders in the treated and control group such that they are similar on average. We consider the inverse of this idea. Given several datasets, a subset of data is selected such that the confounding variables are as balanced as possible across different datasets, whilst maximising the distance in the outcome variable. We then outline an intuitive argument using the SCM formulation of robustness that given many datasets all spurious covariates should be well matched whilst causal covariates in matched pairs will have large distances. Using this subset of matched data for training should bias the model to use causal covariates to predict the label and improve robustness. This is

explored in randomised experiments on both synthetic and semi-synthetic data. As some assumptions are not empirically verifiable in real datasets, synthetic data was used to assess the methods effectiveness in the case where we know that the assumptions are true. Additionally, real covariate data is then used to simulate the outcome variable, and the method is assessed in comparison with normal ERM training, IRM training, and training on a random subset of equal size.

6.2 Limitations and directions for future work

There are several limitations to the work presented in this thesis. These can be largely categorised into those related to adopting the causal view, and those related to the experimental conditions adopted by empirical work based on said view; these are visited in turn.

The causal view is not the only possible formalisation, and may not be the correct one for every application. These are due to the assumptions adopted in this formalisation. We can never be certain that two distributions come from the same SCM. The restriction that causal graphs are acyclic is substantial, and there are many systems in which this is unlikely to be the case. Whilst the causal formalisation is promising in understanding robustness, it remains awkward for high-dimensional data such as images; what is the causal graph that generates the image? Finally, we will never know what the 'true' causal graph of some natural phenomenon is, only the best guess given all available data, but this is true for scientific theories in general.

Turning to the empirical work, the robustness benchmark used for HAR models can of course expand in the number of datasets, and the number of activities; this is a challenge given that high-quality HAR data is scarce, and those with overlapping activities captured under similar conditions even more so. A larger number of datasets will allow more conclusive evidence as to the increase in robustness gained from multiple dataset training, and the dominant role of data diversity to improved robustness in any specific test domain. Shallow and more traditional models in HAR were not tested, and according to the causal view these should be more robust than their deep counterparts. Of course, the causal view will be further substantiated if these hypotheses were also investigated in other applications areas, and there is some evidence in vision to suggest that these are largely consistent [102].

IEM is a first attempt at using the causal view to build a general purpose algorithm to improve robustness by being selective about the training data; The generality of the approach also means that currently it is restricted to tabular datasets. The improvements in high-dimensions, if any, has not been

tested, and according to the intuition behind the approach, is unlikely to be significant with the algorithm in its current form, further extensions have been discussed in Chapter 5. Whilst attempts have been made in this thesis to establish some theoretical guarantee as to the increase in robustness offered by IEM, this remains elusive. Of course, there can always be more datasets and baselines. An obvious remaining question is: can we move beyond synthetic and semi-synthetic data? With real data, it would be difficult to understand whether the assumptions of the method are satisfied, and hence whether it offers any improvement. Last but not least, there are unstudied connections to the field of variable selection, and active learning.

Notwithstanding the limitations noted above, it is hoped that the case for adopting the causal view in further developing both general purpose and application-specific robust models remain worthwhile.

Bibliography

[1] *Introducing GitHub Copilot: your AI pair programmer*. en-US. June 2021. URL: https://github.blog/2021-06-29-introducing-github-copilot-ai-pair-programmer/.

[2] John Jumper et al. "Highly accurate protein structure prediction with AlphaFold". en. In: *Nature* 596.7873 (Aug. 2021). Number: 7873 Publisher: Nature Publishing Group, pp. 583–589. ISSN: 1476-4687. DOI: 10.1038/s41586-021-03819-2. URL: https://www.nature.com/articles/s41586-021-03819-2.

[3] Aditya Ramesh et al. *Hierarchical Text-Conditional Image Generation with CLIP Latents*. en. arXiv:2204.06125 [cs]. Apr. 2022. URL: http://arxiv.org/abs/2204.06125.

[4] Zico Kolter and Aleksander Madry. *Adversarial Robustness - Theory and Practice - Neurips Tutorial*. en. Library Catalog: adversarial-ml-tutorial.org. 2018. URL: http://adversarial-ml-tutorial.org/.

[5] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. "Explaining and Harnessing Adversarial Examples". en. In: *arXiv:1412.6572 [cs, stat]* (Mar. 2015). arXiv: 1412.6572. URL: http://arxiv.org/abs/1412.6572.

[6] Xingchao Peng et al. *Moment Matching for Multi-Source Domain Adaptation*. en. arXiv:1812.01754 [cs]. Aug. 2019. URL: http://arxiv.org/abs/1812.01754.

[7] Michael A. Alcorn et al. "Strike (With) a Pose: Neural Networks Are Easily Fooled by Strange Poses of Familiar Objects". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. ISSN: 2575-7075. June 2019, pp. 4840–4849. DOI: 10.1109/CVPR.2019.00498.

[8] Samuel G. Finlayson et al. *Adversarial Attacks Against Medical Deep Learning Systems*. en. arXiv:1804.05296 [cs, stat]. Feb. 2019. URL: http://arxiv.org/abs/1804.05296.

[9] Cathy O'Neil. *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. en. Google-Books-ID: NgEwCwAAQBAJ. Crown, Sept. 2016. ISBN: 978-0-553-41882-8.

[10] Sheng Liu et al. "On the design of convolutional neural networks for automatic detection of Alzheimers disease". en. In: (2019), p. 17.

[11] Karoline Freeman et al. "Use of artificial intelligence for image analysis in breast cancer screening programmes: systematic review of test accuracy". en. In: *BMJ* 374 (Sept. 2021). Publisher: British Medical Journal Publishing Group Section: Research, n1872. ISSN: 1756-1833. DOI: 10.1136/bmj.n1872. URL: https://www.bmj.com/content/374/bmj.n1872.

[12] Emma Beede et al. "A Human-Centered Evaluation of a Deep Learning System Deployed in Clinics for the Detection of Diabetic Retinopathy". In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, Apr. 2020, pp. 1–12. ISBN: 978-1-4503-6708-0. URL: https://doi.org/10.1145/3313831.3376718.

[13] Will Heaven. *Hundreds of AI tools have been built to catch covid. None of them helped.* en. July 2021. URL: https://www.technologyreview.com/2021/07/30/1030329/machine-learning-ai-failed-covid-hospital-diagnosis-pandemic/.

[14] Michael Roberts et al. "Common pitfalls and recommendations for using machine learning to detect and prognosticate for COVID-19 using chest radiographs and CT scans". en. In: *Nature Machine Intelligence* 3.3 (Mar. 2021). Number: 3 Publisher: Nature Publishing Group, pp. 199–217. ISSN: 2522-5839. DOI: 10.1038/s42256-021-00307-0. URL: https://www.nature.com/articles/s42256-021-00307-0.

[15] Laure Wynants et al. "Prediction models for diagnosis and prognosis of covid-19: systematic review and critical appraisal". en. In: *BMJ* 369 (Apr. 2020). Publisher: British Medical Journal Publishing Group Section: Research, p. m1328. ISSN: 1756-1833. DOI: 10.1136/bmj.m1328. URL: https://www.bmj.com/content/369/bmj.m1328.

[16] Chloe Brown et al. "Exploring Automatic Diagnosis of COVID-19 from Crowdsourced Respiratory Sound Data". In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD '20. New York, NY, USA: Association for Computing Machinery, Aug. 2020, pp. 3474–3484. ISBN: 978-1-4503-7998-4. DOI: 10.1145/3394486.3412865. URL: https://doi.org/10.1145/3394486.3412865.

[17] Jing Han et al. “Sounds of COVID-19: exploring realistic performance of audio-based digital testing”. en. In: *npj Digital Medicine* 5.1 (Jan. 2022). Number: 1 Publisher: Nature Publishing Group, pp. 1–9. ISSN: 2398-6352. DOI: 10.1038/s41746-021-00553-x. URL: https://www.nature.com/articles/s41746-021-00553-x.

[18] Jonas Peters, Peter Buhlmann, and Nicolai Meinshausen. “Causal inference by using invariant prediction: identification and confidence intervals”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 78.5 (Nov. 2016), pp. 947–1012. ISSN: 1369-7412. DOI: 10.1111/rssb.12167. URL: https://rss.onlinelibrary.wiley.com/doi/full/10.1111/rssb.12167.

[19] Kevin P. Murphy. *Machine learning: a probabilistic perspective*. en. Adaptive computation and machine learning series. Cambridge, MA: MIT Press, 2012. ISBN: 978-0-262-01802-9.

[20] Kurt Hornik. “Approximation capabilities of multilayer feedforward networks”. en. In: *Neural Networks* 4.2 (Jan. 1991), pp. 251–257. ISSN: 0893-6080. DOI: 10.1016/0893-6080(91)90009-T. URL: https://www.sciencedirect.com/science/article/pii/089360809190009T.

[21] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[22] Hasim Sak, Andrew Senior, and Francoise Beaufays. *Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition*. arXiv:1402.1128 [cs, stat]. Feb. 2014. DOI: 10.48550/arXiv.1402.1128. URL: http://arxiv.org/abs/1402.1128.

[23] Joshua D. Angrist, Jorn-Steffen Pischke, and J Rn-Steffen Pischke. *Mostly Harmless Econometrics: An Empiricist's Companion*. New Jersey, UNITED STATES: Princeton University Press, 2008. ISBN: 978-1-4008-2982-8. URL: http://ebookcentral.proquest.com/lib/cam/detail.action?docID=475846.

[24] Donald B Rubin. “Causal Inference Using Potential Outcomes”. In: *Journal of the American Statistical Association* 100.469 (Mar. 2005), pp. 322–331. ISSN: 0162-1459. DOI: 10.1198/016214504000001880. URL: http://amstat.tandfonline.com/doi/abs/10.1198/016214504000001880.

[25] Judea Pearl. *Causality*. en. Google-Books-ID: f4nuexsNVZIC. Cambridge University Press, Sept. 2009. ISBN: 978-0-521-89560-6.

[26] Paul R. Rosenbaum and Donald B. Rubin. “The central role of the propensity score in observational studies for causal effects”. In: *Biometrika* 70.1 (Apr. 1983), pp. 41–55. ISSN: 0006-3444. DOI: 10.1093/biomet/70.1.41. URL: https://academic.oup.com/biomet/article/70/1/41/240879.

[27] Apinan Hasthanasombat and Cecilia Mascolo. “Understanding the Effects of the Neighbourhood Built Environment on Public Health with Open Data”. en. In: *The Web Conference* (2019), p. 11.

[28] Miguel A. Hernan and JM Robins. *Causal Inference*. 2019. URL: https://cdn1.sph.harvard.edu/wp-content/uploads/sites/1268/2019/02/hernanrobins_v3.21.6.pdf.

[29] Gregory DeAngelo and Benjamin Hansen. “Life and Death in the Fast Lane: Police Enforcement and Traffic Fatalities”. In: *American Economic Journal: Economic Policy* 6.2 (2014), pp. 231–257. ISSN: 1945-7731. URL: https://www.jstor.org/stable/43189384.

[30] Peter Spirtes, Clark N. Glymour, and Richard Scheines. *Causation, Prediction, and Search*. en. MIT Press, 2000. ISBN: 978-0-262-19440-2.

[31] Peter Spirtes, Christopher Meek, and Thomas Richardson. “Causal inference in the presence of latent variables and selection bias”. In: *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*. UAI’95. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., Aug. 1995, pp. 499–506. ISBN: 978-1-55860-385-1.

[32] Dengxin Dai and Luc Van Gool. *Dark Model Adaptation: Semantic Image Segmentation from Daytime to Nighttime*. Tech. rep. arXiv:1810.02575. arXiv:1810.02575 [cs] type: article. arXiv, Oct. 2018. DOI: 10.48550/arXiv.1810.02575. URL: http://arxiv.org/abs/1810.02575.

[33] Georg Volk et al. “Towards Robust CNN-based Object Detection through Augmentation with Synthetic Rain Variations”. In: *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. Oct. 2019, pp. 285–292. DOI: 10.1109/ITSC.2019.8917269.

[34] Ehab A. AlBadawy, Ashirbani Saha, and Maciej A. Mazurowski. “Deep learning for segmentation of brain tumors: Impact of cross-institutional training and testing”. en. In: *Medical Physics* 45.3 (2018), pp. 1150–1158. ISSN: 2473-4209. DOI: 10.1002/mp.12752.

[35] Amir Rosenfeld, Richard Zemel, and John K. Tsotsos. “The Elephant in the Room”. en. In: *arXiv:1808.03305 [cs]* (Aug. 2018). arXiv: 1808.03305. URL: http://arxiv.org/abs/1808.03305.

[36] Dan Hendrycks et al. "Natural Adversarial Examples". en. In: *arXiv:1907.07174 [cs, stat]* (Mar. 2021). arXiv: 1907.07174. URL: http://arxiv.org/abs/1907.07174.

[37] Benjamin Recht et al. "Do CIFAR-10 Classifiers Generalize to CIFAR-10?" In: *arXiv:1806.00451 [cs, stat]* (June 2018). arXiv: 1806.00451. URL: http://arxiv.org/abs/1806.00451.

[38] Benjamin Recht et al. "Do ImageNet Classifiers Generalize to ImageNet?" In: *arXiv:1902.10811 [cs, stat]* (June 2019). arXiv: 1902.10811. URL: http://arxiv.org/abs/1902.10811.

[39] Battista Biggio et al. "Evasion Attacks against Machine Learning at Test Time". In: vol. 7908. arXiv:1708.06131 [cs]. 2013, pp. 387–402. DOI: 10.1007/978-3-642-40994-3_25. URL: http://arxiv.org/abs/1708.06131.

[40] In: ().

[41] Christian Szegedy et al. *Intriguing properties of neural networks*. arXiv:1312.6199 [cs]. Feb. 2014. DOI: 10.48550/arXiv.1312.6199. URL: http://arxiv.org/abs/1312.6199.

[42] Shiori Sagawa et al. "Distributionally Robust Neural Networks for Group Shifts: On the Importance of Regularization for Worst-Case Generalization". In: *arXiv:1911.08731 [cs, stat]* (Apr. 2020). URL: http://arxiv.org/abs/1911.08731.

[43] Dan Hendrycks et al. "The Many Faces of Robustness: A Critical Analysis of Out-of-Distribution Generalization". In: *arXiv:2006.16241 [cs, stat]* (June 2020). arXiv: 2006.16241. URL: http://arxiv.org/abs/2006.16241.

[44] Dong Yin et al. "A Fourier Perspective on Model Robustness in Computer Vision". en. In: *arXiv:1906.08988 [cs, stat]* (Sept. 2020). arXiv: 1906.08988. URL: http://arxiv.org/abs/1906.08988.

[45] Jason Jo and Yoshua Bengio. "Measuring the tendency of CNNs to Learn Surface Statistical Regularities". en. In: *arXiv:1711.11561 [cs, stat]* (Nov. 2017). arXiv: 1711.11561. URL: http://arxiv.org/abs/1711.11561.

[46] Andrew Ilyas et al. "Adversarial Examples Are Not Bugs, They Are Features". In: *arXiv:1905.02175 [cs, stat]* (Aug. 2019). arXiv: 1905.02175. URL: http://arxiv.org/abs/1905.02175.

[47] Mengyue Yang et al. *CausalVAE: Structured Causal Disentanglement in Variational Autoencoder*. en. June 2022. URL: http://arxiv.org/abs/2004.08697 (visited on 12/16/2022).

[48] Xinwei Shen et al. *Weakly Supervised Disentangled Generative Causal Representation Learning*. Aug. 2022. URL: http://arxiv.org/abs/2010.02637.

[49] Ilyes Khemakhem et al. "Variational Autoencoders and Nonlinear ICA: A Unifying Framework". en. In: (Dec. 2020). arXiv:1907.04809 [cs, stat]. URL: http://arxiv.org/abs/1907.04809.

[50] Martin Arjovsky et al. "Invariant Risk Minimization". en. In: *arXiv:1907.02893 [cs, stat]* (Sept. 2019). arXiv: 1907.02893. URL: http://arxiv.org/abs/1907.02893.

[51] Dominik Rothenhausler et al. "Anchor regression: heterogeneous data meets causality". In: *arXiv:1801.06229 [stat]* (Jan. 2018). URL: http://arxiv.org/abs/1801.06229.

[52] Michael Oberst et al. *Regularizing towards Causal Invariance: Linear Models with Proxies*. arXiv:2103.02477 [cs, stat]. June 2021. URL: http://arxiv.org/abs/2103.02477.

[53] David Krueger et al. "Out-of-Distribution Generalization via Risk Extrapolation (REx)". In: *arXiv:2003.00688 [cs, stat]* (Feb. 2021). URL: http://arxiv.org/abs/2003.00688.

[54] Kartik Ahuja et al. *Empirical or Invariant Risk Minimization? A Sample Complexity Perspective*. en. Aug. 2022. URL: http://arxiv.org/abs/2010.16412.

[55] Pritish Kamath et al. *Does Invariant Risk Minimization Capture Invariance?* Feb. 2021. DOI: 10.48550/arXiv.2101.01134. URL: http://arxiv.org/abs/2101.01134.

[56] Kartik Ahuja et al. *Invariant Risk Minimization Games*. Mar. 2020. DOI: 10.48550/arXiv.2002.04692. URL: http://arxiv.org/abs/2002.04692.

[57] Ishaan Gulrajani and David Lopez-Paz. "In Search of Lost Domain Generalization". In: *arXiv:2007.01434 [cs, stat]* (July 2020). arXiv: 2007.01434. URL: http://arxiv.org/abs/2007.01434.

[58] L. G. Valiant. "A theory of the learnable". In: *Communications of the ACM* 27.11 (Nov. 1984), pp. 1134–1142. ISSN: 0001-0782. DOI: 10.1145/1968.1972. URL: https://doi.org/10.1145/1968.1972.

[59] Aman Sinha et al. *Certifying Some Distributional Robustness with Principled Adversarial Training*. en. arXiv:1710.10571 [cs, stat]. May 2020. URL: http://arxiv.org/abs/1710.10571.

[60] Jonas Peters, Dominik Janzing, and Bernhard Scholkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. en. MIT Press, 2017. ISBN: 978-0-262-03731-0.

[61] Yann LeCun, Corinna Cortes, and CJ Burges. "MNIST handwritten digit database". In: *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist* 2 (2010).

[62] Jia Deng et al. "ImageNet: A Large-Scale Hierarchical Image Database". en. In: (2009), p. 8.

[63] Tero Karras et al. *Analyzing and Improving the Image Quality of StyleGAN*. en. arXiv:1912.04958 [cs, eess, stat]. Mar. 2020. URL: http://arxiv.org/abs/1912.04958.

[64] Vladimir Vapnik. "Principles of Risk Minimization for Learning Theory". en. In: (1991), p. 8.

[65] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. en. Cambridge: Cambridge University Press, 2014. ISBN: 978-1-107-29801-9. DOI: 10.1017/CBO9781107298019. URL: http://ebooks.cambridge.org/ref/id/CBO9781107298019.

[66] Martin Gjoreski et al. "Cross-dataset deep transfer learning for activity recognition". en. In: *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*. London United Kingdom: ACM, Sept. 2019, pp. 714–718. ISBN: 978-1-4503-6869-8. DOI: 10.1145/3341162.3344865. URL: https://dl.acm.org/doi/10.1145/3341162.3344865.

[67] Xin Qin et al. "Cross-Dataset Activity Recognition via Adaptive Spatial-Temporal Transfer Learning". In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3.4 (Dec. 2019), 148:1–148:25. DOI: 10.1145/3369818. URL: https://doi.org/10.1145/3369818.

[68] Youngjae Chang et al. "A Systematic Study of Unsupervised Domain Adaptation for Robust Human-Activity Recognition". en. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4.1 (Mar. 2020), pp. 1–30. ISSN: 2474-9567, 2474-9567. DOI: 10.1145/3380985. URL: https://dl.acm.org/doi/10.1145/3380985.

[69] Jiachen Zhao et al. "Local Domain Adaptation for Cross-Domain Activity Recognition". In: *IEEE Transactions on Human-Machine Systems* 51.1 (Feb. 2021). Conference Name: IEEE Transactions on Human-

Machine Systems, pp. 12–21. ISSN: 2168-2305. DOI: 10.1109/THMS.2020.3039196.

[70] Francisco Javier Ordonez Morales and Daniel Roggen. "Deep convolutional feature transfer across mobile activity recognition domains, sensor modalities and locations". In: *Proceedings of the 2016 ACM International Symposium on Wearable Computers*. ISWC '16. New York, NY, USA: Association for Computing Machinery, Sept. 2016, pp. 92–99. ISBN: 978-1-4503-4460-9. DOI: 10.1145/2971763.2971764. URL: https://doi.org/10.1145/2971763.2971764.

[71] Kaixuan Chen et al. "Deep Learning for Sensor-based Human Activity Recognition: Overview, Challenges and Opportunities". en. In: *arXiv:2001.07416 [cs]* (Jan. 2020). arXiv: 2001.07416. URL: http://arxiv.org/abs/2001.07416.

[72] Ling Bao and Stephen S. Intille. "Activity Recognition from User-Annotated Acceleration Data". en. In: *Pervasive Computing*. Ed. by Alois Ferscha and Friedemann Mattern. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2004, pp. 1–17. ISBN: 978-3-540-24646-6. DOI: 10.1007/978-3-540-24646-6_1.

[73] Nishkam Ravi. "Activity Recognition from Accelerometer Data". en. In: (2005), p. 6.

[74] Jianqiang Shen. "Machine Learning for Activity Recognition". en. In: (2004), p. 15.

[75] Francisco Ordonez and Daniel Roggen. "Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition". en. In: *Sensors* 16.1 (Jan. 2016), p. 115. ISSN: 1424-8220. DOI: 10.3390/s16010115. URL: http://www.mdpi.com/1424-8220/16/1/115.

[76] Xin Du, Katayoun Farrahi, and Mahesan Niranjan. "Transfer learning across human activities using a cascade neural network architecture". en. In: *ISWC '19 Proceedings of the 23rd International Symposium on Wearable Computers*. ACM, Sept. 2019, pp. 35–44. DOI: 10.1145/3341163.3347730. URL: https://eprints.soton.ac.uk/434822/.

[77] Artur Jordao et al. "Human Activity Recognition Based on Wearable Sensor Data: A Standardization of the State-of-the-Art". en. In: *arXiv:1806.05226 [cs]* (Feb. 2019). arXiv: 1806.05226. URL: http://arxiv.org/abs/1806.05226.

[78] Oresti Banos et al. “mHealthDroid: A Novel Framework for Agile Development of Mobile Health Applications”. en. In: *Ambient Assisted Living and Daily Activities*. Ed. by Leandro Pecchia et al. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2014, pp. 91–98. ISBN: 978-3-319-13105-4. DOI: 10.1007/978-3-319-13105-4_14.

[79] Attila Reiss and Didier Stricker. “Introducing a New Benchmarked Dataset for Activity Monitoring”. In: *2012 16th International Symposium on Wearable Computers*. ISSN: 2376-8541. June 2012, pp. 108–109. DOI: 10.1109/ISWC.2012.13.

[80] Barbara Bruno et al. “Analysis of human behavior recognition algorithms based on acceleration data”. In: *2013 IEEE International Conference on Robotics and Automation*. ISSN: 1050-4729. May 2013, pp. 1602–1607. DOI: 10.1109/ICRA.2013.6630784.

[81] Yuqing Chen and Yang Xue. “A Deep Learning Approach to Human Activity Recognition Based on Single Accelerometer”. In: *2015 IEEE International Conference on Systems, Man, and Cybernetics*. Oct. 2015, pp. 1488–1492. DOI: 10.1109/SMC.2015.263.

[82] Lloyd Pellatt and Daniel Roggen. “CausalBatch: solving complexity/performance tradeoffs for deep convolutional and LSTM networks for wearable activity recognition”. en. In: *Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers*. Virtual Event Mexico: ACM, Sept. 2020, pp. 272–277. ISBN: 978-1-4503-8076-8. DOI: 10.1145/3410530.3414365. URL: https://dl.acm.org/doi/10.1145/3410530.3414365.

[83] Joaquin Quiñonero-Candela et al. *Dataset shift in machine learning*. Mit Press, 2008.

[84] Mohammad Mahfujur Rahman et al. “Multi-component image translation for deep domain generalization”. In: *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE. 2019, pp. 579–588.

[85] Sarah Erfani et al. “Robust domain generalisation by enforcing distribution invariance”. In: *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)*. AAAI Press. 2016, pp. 1455–1461.

[86] Daniel C. Castro, Ian Walker, and Ben Glocker. “Causality matters in medical imaging”. en. In: *Nature Communications* 11.1 (July 2020), p. 3673. ISSN: 2041-1723. DOI: 10.1038/s41467-020-17478-w. URL: https://www.nature.com/articles/s41467-020-17478-w.

[87] Mei Wang and Weihong Deng. "Deep visual domain adaptation: A survey". In: *Neurocomputing* 312 (2018), pp. 135–153.

[88] Chuanqi Tan et al. "A survey on deep transfer learning". In: *International conference on artificial neural networks*. Springer. 2018, pp. 270–279.

[89] Sarah M Erfani et al. "Robust Domain Generalisation by Enforcing Distribution Invariance". en. In: (), p. 7.

[90] Zheyan Shen et al. "Towards out-of-distribution generalization: A survey". In: *arXiv preprint arXiv:2108.13624* (2021).

[91] Shoubo Hu et al. "Domain generalization via multidomain discriminant analysis". In: *Uncertainty in Artificial Intelligence*. PMLR. 2020, pp. 292–302.

[92] Massimiliano Mancini et al. "Best sources forward: domain generalization through source-specific nets". In: *2018 25th IEEE international conference on image processing (ICIP)*. IEEE. 2018, pp. 1353–1357.

[93] Da Li et al. "Learning to generalize: Meta-learning for domain generalization". In: *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.

[94] Divyat Mahajan, Shruti Tople, and Amit Sharma. "Domain generalization using causal matching". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 7313–7324.

[95] Kartik Ahuja et al. "Invariance Principle Meets Information Bottleneck for Out-of-Distribution Generalization". In: *arXiv preprint arXiv:2106.06607* (2021).

[96] Jack Edmonds. "Paths, trees, and flowers". In: *Canadian Journal of mathematics* 17 (1965), pp. 449–467.

[97] Haoran Zhang et al. "An Empirical Framework for Domain Generalization in Clinical Settings". en. In: *arXiv:2103.11163 [cs]* (Apr. 2021). URL: http://arxiv.org/abs/2103.11163.

[98] Ruocheng Guo et al. "Out-of-distribution Prediction with Invariant Risk Minimization: The Limitation and An Effective Fix". en. In: (2021), p. 22.

[99] Jennifer L. Hill. "Bayesian Nonparametric Modeling for Causal Inference". In: *Journal of Computational and Graphical Statistics* 20.1 (Jan. 2011), pp. 217–240. ISSN: 1061-8600. DOI: 10.1198/jcgs.2010.08162. URL: https://doi.org/10.1198/jcgs.2010.08162.

[100] Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. "The Risks of Invariant Risk Minimization". en. In: *arXiv:2010.05761 [cs, stat]* (Mar. 2021). arXiv: 2010.05761. URL: http://arxiv.org/abs/2010.05761.

[101] Martin Arjovsky, Kamalika Chaudhuri, and David Lopez-Paz. *Throwing Away Data Improves Worst-Class Error in Imbalanced Classification*. en. arXiv:2205.11672 [cs, stat]. May 2022. URL: http://arxiv.org/abs/2205.11672.

[102] Dan Hendrycks and Thomas Dietterich. "Benchmarking Neural Network Robustness to Common Corruptions and Perturbations". en. In: *arXiv:1903.12261 [cs, stat]* (Mar. 2019). arXiv: 1903.12261. URL: http://arxiv.org/abs/1903.12261.

Appendix A

Summary of ICP

The summary is given without proof, please see [18] for details.

The Invariance Assumption

Let there be $|\mathcal{E}|$ different environments $e \in \mathcal{E}$. This represents the different intervention settings that we have observed. Let there be i.i.d data (X^e, Y^e), $X^e \in \mathbb{R}^p$, $Y^e \in \mathbb{R}$ where X are the predictors, and Y the target variable. Each dataset (X^e, Y^e) is drawn from an intervention distribution (as defined previously) i.e their joint distribution depends on the environment e. In the simplest case $|\mathcal{E}| = 2$, and we have, for instance, the observational distribution, and one interventional distribution from a (possibly unknown) intervention.

ICP rests on the following assumption being satisfied. For any set $S \subseteq \{1, ..., p\}$, let X_S denote the vector containing $X_k, k \in S$.

Assumption 1 (Invariance Assumption). There exists a subset of variables $S^* \subseteq \{1, ..., p\}$ such that

$$\forall e \in \mathcal{E},\ X^e \text{ is arbitrarily distributed} \tag{A.1}$$

and

$$Y^e = g(X^e_{S^*}, \epsilon^e) \qquad \epsilon^e \sim F_\epsilon, \epsilon^e \perp\!\!\!\perp X^e_{S^*} \tag{A.2}$$

for some function $g : \mathbb{R}^{|S^*|} \to \mathbb{R}$, and some distribution F_ϵ, which remains the same across all environments $e \in \mathcal{E}$.

Intuitively, this assumption states that we assume the existence of a model that predicts well (is invariant) across all environments (intervention distributions).

As an example, consider a model where the target Y is generated by a linear

function of X^e

$$Y^e := \mu + X^e\gamma^* + \epsilon^e$$

in this case, the subset S^* of predictors is given by the support of the coefficients γ^* i.e, $S^* = \{k : \gamma_k^* \neq 0\}$. In the linear model case, the invariance assumption states that there exists $\gamma^* = (\gamma_1^*, ..., \gamma_p^*)^T$ with support $S^* = \{k : \gamma_k^* \neq 0\} \subseteq \{1, ..., p\}$ such that $\forall e \in \mathcal{E}$, X^e is arbitrarily distributed and

$$Y^e = \mu + X^e\gamma^* + \epsilon^e, \qquad \epsilon^e \sim F_\epsilon, \epsilon^e \perp\!\!\!\perp X_{S^*}^e \tag{A.3}$$

where μ is an intercept term, ϵ^e has mean 0, finite variance and the same distribution F_ϵ across all environments $e \in \mathcal{E}$. Note that S^* does not necessarily have to be unique.

Note that the definition of Assumption 1 above can hold for any subset of variables, and is a general mathematical definition. The following proposition links this to SCMs, and in particular, show that the parents of a target variable satisfies Assumption 1 under an additional assumption.

Proposition 1. The parents of any target variable Y in a Linear SCM satisfies the invariance assumption (A.3) i.e,

$$S^* = PA(Y)$$

This assumes that data from each environment e arises from one or more interventions on $X_2, X_3, ...X_p$ but not on $Y := X_1$, and interventions are either do or soft interventions.

In the following two sections, relevant definitions are developed that leads up to a description of a general algorithm to identify causal predictors in linear SCMs.

Plausible and identifiable causal predictors

In the remaining discussion, the intercept μ is dropped for brevity. In general, (γ^*, S^*) are not the only pair that satisfies the invariance assumption in a linear SCM. Therefore, for any $\gamma \in \mathbb{R}^p$ and $S \subseteq \{1, ..., p\}$ define the null hypothesis $H_{0,\gamma,S}(\mathcal{E})$

$$H_{0,\gamma,S}(\mathcal{E}) : \gamma_k = 0 \text{ if } k \notin S \text{ and } \begin{cases} \exists F_\epsilon \text{ such that } \forall e \in \mathcal{E} \\ Y^e = X^e\gamma + \epsilon^e \qquad \text{where } \epsilon^e \perp\!\!\!\perp X_S^e,\ \epsilon^e \sim F_\epsilon \end{cases} \tag{A.4}$$

Definition. Variables $S \subseteq \{1, ..., p\}$ are called *plausible causal predictors*

under $\mathcal{E}$ if the following null hypothesis is true

$$H_{0,S}(\mathcal{E}) : \exists \gamma \in \mathbb{R}^p \text{ such that } H_{0,\gamma,S}(\mathcal{E}) \text{ is true.} \tag{A.5}$$

Definition. *Identifiable causal predictors* under $\mathcal{E}$ are the following subset of plausible causal predictors:

$$S(\mathcal{E}) := \bigcap_{S: H_{0,S}(\mathcal{E}) \text{ is true}} \tag{A.6}$$

In particular, under Assumption 1, $H_{0,\gamma^*,S^*}(\mathcal{E})$ is true, and therefore $H_{0,S^*}(\mathcal{E})$ is true, i.e, S^* is a plausible causal predictor. This means the identifiable causal predictors are a subset of the true causal predictors:

$$S(\mathcal{E}) \subseteq S^*$$

Additionally, the set of identifiable causal predictors under $\mathcal{E}$ is growing monotonically with larger $\mathcal{E}$,

$$S(\mathcal{E}_1) \subseteq S(\mathcal{E}_2) \quad \text{where } \mathcal{E}_1 \subseteq \mathcal{E}_2$$

In the next step, a similar concept is defined for coefficients γ.

Definition. Define $\Gamma_S(\mathcal{E})$, the set of *plausible causal coefficients* for the set $S \subseteq \{1, ..., p\}$ as

$$\Gamma_S(\mathcal{E}) := \{\gamma \in \mathbb{R}^p : H_{0,\gamma,S}(\mathcal{E}) \text{ is true}\} \tag{A.7}$$

Definition. Define the set $\Gamma(\mathcal{E})$ of global *plausible causal coefficients* under $\mathcal{E}$ as

$$\Gamma(\mathcal{E}) := \bigcup_{S \subseteq \{1,...,p\}} \Gamma_S(\mathcal{E}) \tag{A.8}$$

This means that the global plausible causal coefficients shrink with larger $\mathcal{E}$:

$$\Gamma(\mathcal{E}_1) \supseteq \Gamma(\mathcal{E}_2) \quad \text{if } \mathcal{E}_1 \subseteq \mathcal{E}_2$$

Re-writing $H_{0,S}$ allows us to see that the set of plausible causal coefficients for set S is either empty or contains only the population regression vector.

Let the least squares population regression coefficients for environment e be defined as

$$\beta^{pred,e}(S) := \min_{\beta \in \mathbb{R}^p : \beta_k = 0 \text{ if } k \notin S} \mathbb{E}[Y^e - X^e \beta]^2 \tag{A.9}$$

The null hypothesis for a set $S \subseteq \{1, ..., p\}$ can be re-written as

$$H_{0,S}(\mathcal{E}) : \begin{cases} \exists \beta \in \mathbb{R}^p \text{ and } \exists F_\epsilon \text{ such that } \forall e \in \mathcal{E} \\ Y^e = X^e \beta + \epsilon^e \quad \text{where } \beta^{pred,e}(S) \equiv \beta,\ \epsilon^e \perp\!\!\!\perp X_S^e,\ \epsilon^e \sim F_e \end{cases} \tag{A.10}$$

We can then conclude

$$\Gamma_s(\mathcal{E}) = \begin{cases} \emptyset & \text{if } H_{0,S}(\mathcal{E}) \text{ is false,} \\ \beta^{pred,e}(S) & \text{otherwise} \end{cases} \tag{A.11}$$

This fact is used next to construct a generic algorithm to compute estimates of identifiable causal predictors.

Estimating identifiable causal predictors

It is now possible to estimate the set $S(\mathcal{E})$ of identifiable causal predictors and confidence intervals for the linear causal coefficients when observing (X^e, Y^e) in different environments $e \in \mathcal{E}$.

1. For each $S \subseteq \{1, ..., p\}$ test whether $H_{0,S}(\mathcal{E})$ holds at level α
2. Let
$$\hat{S}(\mathcal{E}) := \bigcap_{S: H_{0,S}(\mathcal{E}) \text{ not rejected}} \tag{A.12}$$
3. Define
$$\hat{\Gamma}(\mathcal{E}) := \bigcup_{S \subseteq \{1,...,p\}} \hat{\Gamma}_S(\mathcal{E}) \tag{A.13}$$
where
$$\hat{\Gamma}_S(\mathcal{E}) := \begin{cases} \emptyset & H_{0,S}(\mathcal{E}) \text{ can be rejected at level } \alpha \\ \hat{C}(S) & \text{otherwise} \end{cases} \tag{A.14}$$
where $\hat{C}(S)$ is the $1 - \alpha$ confidence set of the regression vector $\beta^{pred}(S)$ obtained by pooling the data.

Coverage of the true causal predictors and coefficients can be guaranteed given that the hypothesis test and confidence interval has the claimed coverage probability given by the below theorem

Theorem 4 *If the estimator $\hat{S}(\mathcal{E})$ is constructed according to A.12 with a test for : $H_{0,S}(\mathcal{E})$ at level α. Consider γ^*, S^* such that Assumption 1 holds, then*

$$\mathbb{P}[\hat{S}(\mathcal{E}) \subseteq S^*] \geq 1 - \alpha$$

Additionally, for all (γ, S) *that satisfy Assumption 1 if* $\hat{C}(S)$ *in satisfies* $\mathbb{P}[\gamma \in \hat{C}(S)] \geq 1 - \alpha$ *then*

$$\mathbb{P}[\gamma^* \in \hat{\Gamma}(\mathcal{E})] \geq 1 - 2\alpha$$

Appendix B

Why use a validation set for model selection?

The following is also based on Shalev-schwartz, please see [65] for proofs.

Often, we'd like a better estimate of the true risk for a specific hypothesis, as the bounds in Section 3.5 are valid for all hypotheses in a class. This true risk can also guide our model selection process. Practically, this has been done using a validation set.

We can get a better estimate of the true risk by examining the empirical error of a validation set, a fresh sample from the distribution that is independent from the training set.

Theorem 5 *Given a hypothesis h and a loss function in [0,1]. For every $\delta \in (0,1)$ with probability $1-\delta$ over the choice of validation set V of size m_v:*

$$|L_V(h) - L_{\mathcal{D}}(h)| \leq \sqrt{\frac{\log(2/\delta)}{2m_v}}$$

Where L_V is the risk on the validation set. This is a tighter bound and is irrespective of the learning algorithm or the training sample S used originally to obtain h. This validation set can also be used to select a model from multiple candidates. For instance, we may have several hypotheses which use different hyperparameters and one can be chosen based on the empirical risk on the validation set.

Theorem 6 *Let $\mathcal{H} = \{h_1, h_2, ..., h_n\}$ be an arbritary set of hypotheses and that the loss function is in [0,1]. Given a validation set V of size m_v sampled independently of $\mathcal{H}$, with probability at least $1-\delta$ over the choice of V*

$$\forall h \in \mathcal{H}, \quad |L_{\mathcal{D}}(h) - L_V(h)| \leq \sqrt{\frac{\log(2|\mathcal{H}|/\delta)}{2m_v}}$$