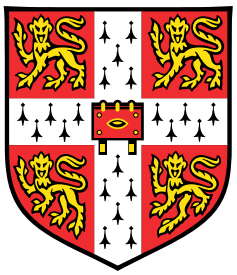
Fairness in Machine Learning with Causal Reasoning



Philip Ball

Department of Engineering
University of Cambridge

This dissertation is submitted for the degree of
Master of Philosophy in Machine Learning, Speech and Language Technology

Sidney Sussex College August 2018

I would like to dedicate this thesis to my ever-loving (and ever-patient) parents.

Declaration

I, Philip Ball of Sidney Sussex College, being a candidate for the M.Phil in Machine Learning, Speech, and Language Technology, 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.

Total word count: 14,984

Signed:

Date: 17/08/2018

Philip Ball
August 2018

Acknowledgements

I would like to acknowledge first and foremost my supervisor, Dr. Adrian Weller. His support and guidance was critical to my gaining a deep understanding of the subject matter, as well as deciding what areas in particular would be interesting to look at. Furthermore I would also like to thank him for putting up with my many ventures down to the Alan Turing Institute, which were incredibly important to the successful completion of this thesis.

I would like to thank, in no particular order of course, Niki Kilbertus, Dr. Chris Russell, Dr. Matt Kusner and Dr. Ricardo Silva. The discussions we had collectively were instrumental to the direction this project took, and I very much appreciate the time you spent teaching me the finer aspects of causality and counterfactual fairness.

I would also like to thank Prof. Bill Byrne, whose pastoral support during the thesis was incredibly helpful and appreciated.

Finally, a special thank you to Osman and Alex, who were always there for me in my less lucid moments during the MPhil course. It was an absolute blast working with you guys, and I'll never forget the productive (and less productive) chats we shared over the year.

Abstract

Machine learning algorithms continue to impact and affect different aspects of people's lives through the decisions they make (i.e., whether to grant a loan or not). What is evident in a number of studies is that these automated decisions can have negative consequences towards the individuals they are made against, particularly if they are a member of a marginalised group (i.e., the elderly, the disabled, etc.). Therefore, in order to prevent such biases, we need to find ways of ensuring fairness within the algorithms we develop.

This thesis presents a review of recent and popular fairness techniques designed to mitigate biases, and explores the use of causality to ensure fairness, specifically using counterfactual reasoning. In causality, the correct specification of the causal model is paramount to reducing biases in the estimands, and we show that this carries over to the fairness of decisions. In order to measure counterfactual fairness, we develop a new metric termed 'Counterfactual Unfairness', and demonstrate how different misspecifications in the causal model affect counterfactual fairness. Furthermore, we perform a comparison across a number of different existing fairness techniques under the CFU measure, and understand their relationship to causal notions of fairness. Finally, we develop a novel variational and adversarial approach to counterfactual fairness, and show how this allows the joint learning of: a) a lower-dimensional representation of the latent space; b) a counterfactually fair predictor. We extend this method to the case of 'multiple worlds', and show that we are able to learn a single low-dimensional latent representation which fulfils fairness simultaneously across multiple causal graphs.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

The natural distribution is neither just nor unjust; nor is it unjust that persons are born into society at some particular position. These are simply natural facts. What is just and unjust is the way that institutions deal with these facts.

— John Rawls, *A Theory of Justice* [46]

1.1 Motivation

Machine learning algorithms continue to impact and affect different aspects of people's lives, such as through credit scoring [42] and criminal risk assessment [1]. It has been shown that automation of such decisions with machine learning algorithms can have both positive [16] and negative [10] impacts on individuals against whom the decisions are made. It is therefore clear that the incorporation of machine learning algorithms into quotidian decision-making presents us with the opportunity to either ameliorate injustices in society or further exacerbate them [15]. Given that the former is preferable (and the latter is typically illegal [50]), the question remains; how can we ensure that our algorithms are able to achieve this?

Unsurprisingly, there now exists a burgeoning body of research aimed at tackling this exact question, and within it, a multitude of ways to ensure fairness. Broadly speaking, most (but not all) notions of fairness within machine learning concern themselves with defining parity/equality conditional on some protected attribute(s) (i.e.: race, gender, etc.). In this work we primarily concern ourselves with causally [44, 45] derived notions of fairness (which we motivate and introduce in Chapter 2).

1.2 Thesis Contributions

Our contributions incude:

1. A **review** of existing forms of fairness and an introduction to causal inference.
2. Defining a **novel metric** (Counterfactual Unfairness, or CFU) which allows for simple and robust comparison of fairness algorithms under counterfactual fairness.
3. Understanding the impact of causal **model misspecification** on counterfactual fairness in a variety of different scenarios and extending the sensitivity analysis approach used in causality to counterfactual fairness.
4. Applying the CFU metric to a number of observationally based approaches and understanding how these **perform counterfactually**.
5. The design and evaluation of a **novel adversarial approach** to counterfactual fairness using deep-learning and variational inference.

1.3 Thesis Outline

Since fairness in machine learning (ML) is a nascent field, it is important to establish and consolidate current techniques and methods that ensure fairness within ML algorithms. Therefore, in this work we present a review of salient fairness techniques, discussing and contrasting their relative merits and drawbacks.

Furthermore, we provide an introduction to causal inference, followed by a review of recent causal approaches to fairness. We show how causality motivates a more intuitive sense of fairness compared with existing observational approaches.

In order to measure the fairness of predictions under counterfactuals, we introduce a new 'counterfactual unfairness' metric, termed CFU. This will allow for comparisons between fairness algorithms under counterfactual fairness.

As discussed in [44], correct model specification is key to ensuring the accurate measurement of causal effects, and inaccuracies will result in biases within the estimands. Unsurprisingly, the importance of model specification extends to fair counterfactual predictions. We therefore provide a review (the first of which we are aware) of how different types of model misspecification can affect the final fairness of our algorithms, drawing inspiration from the traditional causal literature.

Furthermore, we provide an overview of current observational fairness methods and measure how well they perform under the CFU metric.

Finally, we leverage advances in generative modelling and deep learning [30, 17] to design a new method to jointly learn a counterfactually fair predictor and lower-dimensional representation of the latent space[1]. This is especially important for high-dimensional data, as this allows us to compress the latent space without significant loss of accuracy or fairness. We use both variational inference and adversarial techniques to achieve this, and show that the learnt representations can be used to train other counterfactually fair models. We extend the method to multi-world fairness [47], and produce a single low-dimensional representation that satisfies fairness under multiple causal graphs.

[1] For clarity, latent space refers to the exogenous variables in the causal graph.

Chapter 2

Background

2.1 Introduction to Fairness in Machine Learning

As fairness in machine learning is a nascent field of research, it is worth noting and categorising its various branches, as well as clarify common terminology.

2.1.1 How Does Unfairness Arise?

We have identified fairness within algorithms as being critical to their application in the real world, but how is it that these algorithms learn to make unfair decisions in the first place? After all, the algorithms themselves are typically not designed to actively make decisions that are detrimental towards disadvantaged groups. The answer therefore lies within the data itself, as described by [2] and [38]:

Target Variable Bias

Issues may arise in the target variable (i.e., the quantity we wish to predict) itself. For example, if the data we collect has biases against certain minority groups due to systematic inequalities in society (i.e., racism, sexism), then machine learning algorithms can only learn to emulate these biases. This was described in [37]; "the program was not introducing new bias but merely reflecting that already in the system."

Another related issue may involve the choice of the target variable itself. For example, if an algorithm to determine offending risks uses 'number of arrests' as the target variable,

this may discriminate against certain protected groups. This is because 'number of arrests' is simply an indicator how many times someone has been caught, not a true indication of someone's true likelihood to commit a crime. Conversely, if a certain group has a lower rate of arrests, this may be due to them simply getting away with criminal activity; certain police forces may choose to only investigate areas with a higher concentration of certain minority groups, or only stop-and-search individuals from a minority group. Some observational methods struggle to ameliorate such historical bias, as we will cover later.

Data Collection Bias

Issues may arise in the data collection phase. For example, it may be the case that during the data collection process, certain groups of individuals are over- or under-represented. For example, if an online survey was conducted, it is likely that poorer groups are under-represented due to their inability to access the Internet. Under- and over-representation can produce unfair decisions due to poor data coverage and confirmation biases respectively.

Another issue is the type of data we collect and how well it represents certain protected groups. For example, consider that we wish to grant loans, and make decisions based on where an individual lives. It may transpire that for a certain subgroup this proves to be highly predictive of their default rates, but there exists another smaller subgroup who predominately live in just a single area. Therefore the resultant algorithm could potentially achieve high accuracy overall, but deliver poor accuracy for the smaller subgroup.

Feedback Loops

Issues may arise due to dynamic effects within the modelling process. Concretely, an imbalance in initial conditions of some decision may result in a positive feedback loop that can exacerbate prejudices. This is most clearly illustrated in predictive policing case studies, such as [38] and [14]. In the latter, the problem is addressed within the reinforcement learning framework, and they seek to learn of the true crime rate that exists in a region. They show that if two different regions have non-identical crime-rates, in the limit, the policy will assign all police resources to the region with higher crime, which is clearly sub-optimal.

2.2 Observational Notions of Fairness

We use the following notation to introduce observational fairness:

- A: Protected attribute (i.e., race, gender, age)
- X: Features, which do not include A
- Y: The true observed outcome of the variable we wish to predict
- $\hat{Y}$: The prediction produced by the algorithm of the outcome Y

Lower case variables represent elements of the sets defined above.

2.2.1 Fairness Through Unawareness

Fairness Through Unawareness (FTU) is defined as follows:

Definition 2.2.1. A predictor $\hat{Y}$ satisfies **Fairness Through Unawareness** if it does not make explicit use of protected attributes A in its predictions.

This definition does not take into account the potential of the predictor $\hat{Y}$ being able to 're-construct' proxies for A using X (the remaining features) when they are highly predictive of the true outcome Y, which can occur in the case of historical data which shows negative bias towards disadvantaged group. Having said this, FTU has been shown to be more effective than no action in [18] and [32], with the latter stating that it should be used as a baseline.

2.2.2 Demographic Parity

Demographic Parity (DP) is defined as follows:

Definition 2.2.2. A predictor satisfies **demographic parity** if the predictor $\hat{Y}$ satisfies the following:

$$P(\hat{Y} = y|A = a) = P(\hat{Y} = y|A = a') \tag{2.1}$$

for all y, a, a' (i.e., $\hat{Y} \perp\!\!\!\perp A$).

In effect, we must fully decorrelate the protected attribute with the final decision.

One issue with this is the fact that such a requirement allows us to accept unqualified individuals in the group $A = 1$ as long as the parity is matched as in Eq 2.1. This can be particularly dangerous since someone may purposefully select unqualified candidates who are members of a qualified group to so as to appear fair under DP, but also justify any pre-held prejudices in an attempt to sabotage efforts at ensuring fairness.

Another issue lies with the fact that such a criterion is particularly damaging to predictor utility. For example, if there is any actual correlation between the true Y and the protected attribute A, DP would not allow us to construct the ideal predictor $\hat{Y} = Y$.

2.2.3 Calibration/Predictive Parity

Calibration is defined as follows:

Definition 2.2.3. A predictor satisfies **calibration** if the predictor $\hat{Y}$ satisfies the following:

$$P(Y = y|A = a, \hat{Y} = y) = P(Y = y|A = a', \hat{Y} = y) \tag{2.2}$$

for all y, a, a' (i.e., $Y \perp\!\!\!\perp A|\hat{Y}$).

In effect, we wish that for a given predicted score, the proportion of people who experience a particular outcome is the same across all protected groups. The issues with this type of fairness are addressed in Section 2.2.4.

2.2.4 Equality of Opportunity/Equalized Odds/Other Analogues

Equalized Odds [20] is defined as follows:

Definition 2.2.4. A predictor satisfies **Equalized Odds** if the predictor $\hat{Y}$ satisfies the following:

$$P(\hat{Y} = 1|A = a, Y = y) = P(\hat{Y} = 1|A = a', Y = y) \tag{2.3}$$

for all y, a, a' (i.e., $\hat{Y} \perp\!\!\!\perp A|Y$).

This is equivalent to matching false positive rates (FPR) and true positive rates (TPR) across protected groups. If we relax this criterion such that only one of FPR or TPR is matched (i.e., the latter if the preferred outcome is 1), we obtain **Equality of Opportunity** (i.e., $P(\hat{Y} = 1|A = a, Y = 1) = P(\hat{Y} = 1|A = a', Y = 1)$.

In [31] a nearly identical form of fairness is introduced, but generalises both Equality of Opportunity and Equalized Odds to real-valued scores (as opposed to binary classifiers).

The issue with such a measure of fairness is two-fold. Firstly, such measures are open to 'fairness gerrymandering' [25]. In short, the nature in which we define each protected attribute (i.e., black/white, male/female) means that to fulfil such notions of fairness, we simply need to make sure that when looked at individually we fulfil the parity measures required by the criteria. However, if considering combinations of the sensitive attribute (i.e., black male, white female), it is possible to discriminate against these 'sub-groups' to ensure the overall fairness of their parent groups.

Secondly, such measures cannot deal with historic biases in the target variable (a problem discussed earlier in 2.1.1). An example of this would be arrest data, where the target variable can be misleading; someone not being arrested does not mean they have not committed a crime, it simply means they were either not caught, or perhaps let off due to institutional biases. In this instance, just because we assure some parity based on the target variable (such as False Positive Rate), the resulting algorithm still will not be fair because of the bias in this target variable.

Counterfactual Fairness (discussed later) addresses both these points, since we leverage causal assumptions in the data generation process to deconvolve any biases due to the protected attribute A out of the original data X. Therefore instead of relying on Y to ensure fairness, simply use Y to make sure we can build a model $\hat{Y}$ that is still predictive of Y itself. Since all unfairness due to the protected attribute has been removed from the input space, there will be no bias due to A, hence any algorithm we learn with this new input space can be considered fair. In the case of learning Y, the bias due to A will simply be noise that cannot be accounted for in the transformed data.

2.2.5 Individual Fairness

Individual Fairness [12] is defined as follows:

Definition 2.2.5. A predictor satisfies **Individual Fairness** if the predictor $\hat{Y}$ satisfies the following:

$$P(\hat{Y} = y | A = a^{(i)}, X = x^{(i)}) = P(\hat{Y} = y | A = a^{(j)}, X = x^{(j)}) \tag{2.4}$$

while $d(i, j) \approx 0$.

In this case, i, j refer to two different individuals, and the superscripts $(i), (j)$ refer to their associated data. The function $d(\cdot, \cdot)$ is a metric that measures the distance between any two individuals. In a fair world, this metric should be small for individuals who are similar, except for their protected attributes. However, designing a suitable metric $d(\cdot, \cdot)$ is not straightforward as many features X are covariate with A, so defining a simple distance measurement along X is not sufficient.

2.2.6 Total Fairness

Total Fairness is the simultaneous fulfilment of multiple notions of observational fairness. This was studied against the backdrop of the ProPublica/COMPAS case (see [1]) in [31], and further developed in [4]. In summary, it has been shown that in most cases, multiple forms of observational fairness cannot be satisfied simultaneously apart from degenerate cases. For example, [31] showed that **calibration** and **equalized odds** cannot be simultaneously satisfied apart from the cases where we have: a) perfect prediction; b) equal base rates for each protected group.

2.3 Causal Notions of Fairness

As mentioned in Section 2.2.6, it is often not possible to simultaneously satisfy multiple forms of observational fairness. Causality allows us to sidestep such issues, as well as answer more fundamental problems about data which observational studies would not be able to answer.

2.3.1 A Brief Introduction to Causality

We cover the causal inference tools that are used in the current causal fairness literature, as well as motivate when causality is required to answer questions which observations cannot.

Causality requires the definition of a structural causal model (SCM), which is defined as a 3-tuple (U, V, F) of sets where:

- Variables U are called **exogenous variables** that are external to the model (i.e.: have no parents), which we choose to not explain how they are caused; these are also referred to as **latent variables**.

- Variables V are called **endogenous variables**, each of which is a descendant of at least one endogenous variable in U.
- Functions F are called **structural equations**, which define functional relationships between one of the variables $V_i \in V$ and other values in the graph. These are of the form $V_i = f_i(pa_i, U_{pa_i})$, where pa_i represents the parents of the variable V_i.

We also observe that each SCM has a corresponding **causal graphical model** (CGM), (a.k.a., graphical model or graph), such that the variables U and V are nodes, and the functions F are directed edges which connect the nodes (i.e.: if f_i contains a variable V_j, there will be an arrow pointing from V_j to V_i). We assume all CGMs to be directed acyclic graphs (DAGs) (a graph where there is no directed path between any node and itself).

We now introduce two key causal inference tools which allow us to reason about data in a way that observational methods are not able to.

Interventions

Interventions are ways of modifying the CGM to understand how certain 'actions' affect outcomes. This would not be necessary if we could perform randomised controlled experiments to determine the impact of one variable on another, such as the effect of a drug on the survival rate of an individual. In this case, we are able to hold all variables that influence the outcome (apart from the input variable, i.e.: treatment) either constant or vary them at random, such that changes in output can only be due to the input variable. However, it is not always possible to perform such experiments, since certain variables may be out of our control. For instance, while we can control the selection of participants in a drug trial, we are not able to account for the actions of participants themselves (i.e., withdrawal bias, protocol violation [22]).

In most cases it is only possible to record the data and observations, but this does not answer the question of correlation v.s. causation. For example, we may observe the following statistics from a clinical drug trial:

	Drug	No Drug
Men	81/87 recovered (93%)	234/270 recovered (87%)
Women	192/263 recovered (73%)	55/80 recovered (69%)
Combined	273/350 recovered (78%)	289/350 recovered (83%)

Table 2.1 Drug Trial exhibiting Simpson's Paradox

We highlight in red the most effective treatment in each row. We get a **reversal** in the most effective treatment when we combine the observations. It is therefore unclear whether we ought to take the combined statistics or the gender-level statistics, which implies the following paradoxical statement (named **Simpson's Paradox**): if the gender of the patient is unknown, we ought to not administer the drug, but as soon as we learn the gender of the patient, we ought to administer the drug. This is not a correct conclusion, since if the drug helps both men and women, it will help everyone. However, it is not as sufficient to state that we should therefore administer the drug for every patient, since we cannot infer the true causal effect from just the observations. Therefore using some prior knowledge of the experiment itself, we construct the following CGM:

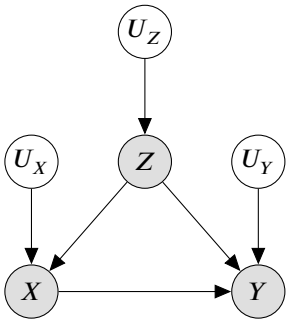


Fig. 2.1 CGM to represent a drug trial

where X represents drug administration ($X = 1$ for administered), Y represents outcome ($Y = 1$ for recovered), Z represents gender ($Z = 1$ for Female), and U_i represents the unobserved latent variables for each observed variable. We can now answer the following question: does the drug have a positive effect on patients? We introduce Pearl's **do-calculus** [44], which allows us to perform interventions in the form $P(Y|do(X = x))$, which is distinct from $P(Y|X = x)$. The latter describes the conditional probability of Y given we **observed** $X = x$ (i.e.: we narrow our focus to the cases where $X = x$), whilst the former allows us to describe the conditional probability of Y given we **intervene** and force everyone in the

population to have the value $X = x$. Such an operation is only possible with SCMs, and is shown in the following '**graph surgery**' to the CGM:

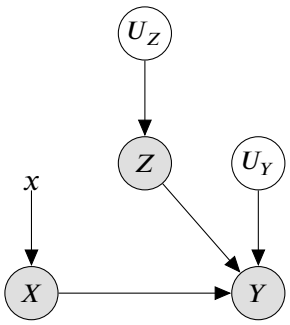


Fig. 2.2 CGM with an intervention on X

The intervention $do(X = x)$ involves deleting all edges into X and forcing the value of X to be that of x. This allows us to formulate the question concerning the drug's efficacy as follows:

$$\text{ACE} = P(Y|do(X = 1)) - P(Y|do(X = 0)) \tag{2.5}$$

where ACE stands for Average Causal Effect, and denotes the following quantity: what is the difference in recovery rate between administering the drug uniformly to the entire population and not administering the drug at all? This is equivalent to performing a controlled randomized trial of the same drug. However, the data we have does not actually allow us to answer the question definitively (since it wasn't a controlled randomized trial), but **assuming the validity** of the CGM in Fig 2.1, we can now calculate this quantity, which was impossible previously. Having modified the graph as in Fig 2.2, we calculate probabilities relative to the intervened graph of the form P_m (where m stands for modified), i.e., $P_m(Y = y|X = x)$, and note by construction $P(Y = y|do(X = x)) = P_m(Y = y|X = x)$. We now use the following equations of invariance:

$$P_m(Z = z) = P(Z = z) \tag{2.6}$$

and

$$P_m(Y = y|Z = z, X = x) = P(Y = y|Z = z, X = x) \tag{2.7}$$

Eq 2.6 arises as a consequence of the causal mechanism illustrated in Figs 2.1 and 2.2; the assignment of gender is not affected by removal of the arrows into X. Eq 2.7 also arises due

to the causal graph; the value of Y only depends on the values of X and Z, and is agnostic to how either was actually generated (be it through an intervention or through natural causes). We are now in a position to calculate the interventional quantity $P(Y = y|do(X = x))$ by combining all previous equations:

$$\begin{aligned} P(Y = y|do(X = x)) &= P_m(Y = y|X = x) && (2.8)\\ &= \sum_z P_m(Y = y|X = x, Z = z)P_m(Z = z|X = x) && (2.9)\\ &= \sum_z P_m(Y = y|X = x, Z = z)P_m(Z = z) && (2.10)\\ &= \sum_z P(Y = y|X = x, Z = z)P(Z = z). && (2.11) \end{aligned}$$

Eq 2.11 is called the **adjustment formula**, and can in fact be identified using the **back-door criterion** [44]. This criterion is common in causal literature, and determines whether an unbiased calculation of the interventional quantity is possible given a CGM assumed for that data. Using Eq 2.11, we now calculate Eq 2.5:

$$\begin{aligned} P(Y = y|do(X = 1)) &= \sum_z P(Y = y|X = 1, Z = z)P(Z = z) && (2.12)\\ &= P(Y = y|X = 1, Z = 1)P(Z = 1) \\ &\quad + P(Y = y|X = 1, Z = 0)P(Z = 0) && (2.13)\\ &= 0.93 \times \frac{87 + 270}{700} + 0.73 \times \frac{263 + 80}{700} && (2.14)\\ &= 0.832. && (2.15) \end{aligned}$$

Similarly

$$\begin{aligned} P(Y = y|do(X = 0)) &= \sum_z P(Y = y|X = 0, Z = z)P(Z = z) && (2.16)\\ &= P(Y = y|X = 0, Z = 1)P(Z = 1) \\ &\quad + P(Y = y|X = 0, Z = 0)P(Z = 0) && (2.17)\\ &= 0.87 \times \frac{87 + 270}{700} + 0.69 \times \frac{263 + 80}{700} && (2.18)\\ &= 0.782. && (2.19) \end{aligned}$$

This means that we evaluate the ACE in Eq 2.5 to be 0.050, which implies that the overall effect of the drug is positive. To give insight into why the CGM is vital in resolving this paradox, consider the following alternative CGM to explain the observations:

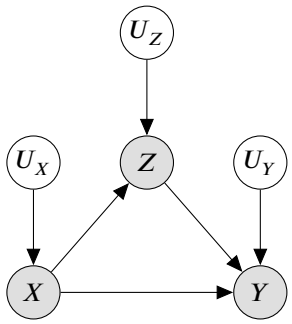


Fig. 2.3 An alternate drug trial CGM

This figure is similar to Fig 2.1 except that the arrow into X from Z has been reversed. In this case, we change the meaning of the variable Z such that it now represents blood pressure at the end of the study ($Z = 1$ is high blood pressure at the end of the trial). In this case, the invariance equation Eq 2.6 is no longer valid, hence we can no longer apply the same adjustment formula as in Eq 2.11. However, the correct estimate of ACE is now simpler. Recall that we wish to calculate $P(Y = y|do(X = x))$, which involves an intervention on X. Performing graph surgery on Fig 2.3, we are simply left with the same diagram, albeit we explicitly state which value of X we are setting this to. In this case, the correct adjustment formula is simply that of the original graph, hence:

$$P(Y = y|do(X = x)) = P(Y = y|X = x). \tag{2.20}$$

This means that we use the aggregated statistics in Table 2.1 (row 3), implying the following ACE calculation:

$$\begin{aligned}
\text{ACE} &= P(Y = 1|do(X = 1)) - P(Y = 1|do(X = 0)) && (2.21)\\
&= P(Y = 1|X = 1) - P(Y = 1|X = 0) && (2.22)\\
&= 0.78 - 0.83 && (2.23)\\
&= -0.05 && (2.24)
\end{aligned}$$

thus showing the drug now has a negative effect. Intuitively, we can view that had we chosen to adjust for final blood pressure in this case (as we had adjusted for gender in Eq 2.11), this

would be the same as stating we believed final blood pressure would cause people to seek initial treatment, which is clearly incorrect.

With this last point in mind, we note the power of causal reasoning. It was initially unintuitive whether or not to adjust for the confounder, but upon constructing the CGM, in both scenarios it became clear whether or not we ought to. This is unsurprising, since causal intuition underpins human reasoning, so it is a natural framework to use when answering questions of a statistical nature.

Whilst there may not appear to be much connection between Simpson's Paradox and fairness, [20] shows that identical joint distributions over variables are in fact able to admit different dependency structures. This poses an issue, as it no longer clear from the just observations how one ought to construct a fair classifier. This further motivates the use of causal reasoning and the construction of causal graphs.

Counterfactuals

Counterfactuals are similar to interventions, but allow for an individual level approach to causal reasoning. While interventions allow us to answer questions on situations that did not actually occur (i.e., what would be the average survival rate be if all subjects took a certain medication?), these are limited to estimating overall effects across populations.

For example, assume there exists two different train routes (denoted R), a and b, between two cities. Using interventions and do-calculus, we are able to calculate probabilities about total travel time T such as $P(T|do(a))$. However, we may be interested in calculating the time on a **hypothetical** jounrey b, given that **in reality** we took route a, and that had taken time t. Attempting to write down such quantities with do-calculus is impossible; $\mathbb{E}[T|do(R = b), T = t]$ is meaningless as we are conditioning on a quantity which we wish to estimate the expectation of. To answer such questions we must instead use the following notation:

$$\mathbb{E}[\underbrace{T_{R=b}}_{\text{counterfactual}} | \underbrace{R = a, T = T_{R=a} = t}_{\text{evidence}}]. \tag{2.25}$$

This gives rise the notion of **counterfactuals**, since we are calculating the hypothetical time in a 'different world' with all other conditions being equal. This is in contrast to the interventional quantity $\mathbb{E}[Y|do(R = b)]$, since this has no reference to a different 'true' world where an actual quantity (evidence) was observed.

Counterfactuals require 3 steps to calculate [44], with reference to the SCM M (i.e., the 3-tuple (U, V, F)) and evidence E:

1. Abduction: given a prior over U, calculate the posterior over U given evidence $E = e$ obtaining $P(U|E = e)$.
2. Action: modify the SCM M by removing the equations F associated with the variable we wish to modify (i.e., X) and intervene on these to the quantity we wish to calculate the counterfactual for (i.e. $R = b$), giving the modified graph M_x.
3. Prediction: compute the final quantity over the remaining variables using the updated posterior $P(U|E = e)$ and modified graph M_x.

This now also allows us to answer questions more directly related to fairness. For example, how would the outcome of some decision (i.e., admission to law school) have changed in a 'different world' where an individual had had a different protected attribute (i.e., gender/age)?

2.3.2 Interventional Fairness

In [26] the idea of using interventions as a means of ensuring fairness is introduced. In order to understand where to apply such interventions, they describe two ways unfairness can arise in the causal model:

1. **Proxy Discrimination**, whereby a variable V is used in the predictive model that inherits from the protected attribute A
2. **Unresolved Discrimination**, whereby a variable V is used in the predictive model that does is not blocked by some resolving variable along the path from the protected attribute $A \rightarrow V$

Unresolved discrimination is particularly interesting, since it introduces the notion of specific paths involving the protected attribute A as being acceptable from a fairness perspective; we discuss the idea of path specific fairness in a later section.

In order to address these two forms of causal unfairness, we construct the SCM, and interventions are performed on the sensitive attributes in the causal model. Under the interventions we then select parameters in the model that cancel out the sensitive attribute in the final prediction of Y. This gives rise to the following definitions of fairness:

$$P(Y|do(A = a)) = P(Y|do(A = a)) \tag{2.26}$$

$$P(Y|do(A = a), X = x) = P(Y|do(A = a'), X = x). \tag{2.27}$$

As mentioned previously, interventions give estimations over populations, and [34] show that Eq 2.26 can admit counterfactually unfair predictions. We discuss in Section 3.2.2 why Eq 2.27 could be interpreted as unfair despite being a more individual level comparison.

2.3.3 Counterfactual Fairness

Counterfactual fairness is introduced in [32], and describes a method to 'deconvolve' the data such that subsequent predictions are counterfactually fair. We begin with their definition of counterfactual fairness:

Definition 2.3.1. Counterfactual Fairness. A predictor $\hat{Y}$ is **counterfactually fair** if under any context $X = x$ and $A = a$,

$$P(\hat{Y}_{A=a}(U) = y|X = x, A = a) = P(\hat{Y}_{A=a'}(U) = y|X = x, A = a) \tag{2.28}$$

for all y and for any value a' attainable by A.

Here U is described as all nodes in the CGM which are non-descendants of A (also called '**latent variables**'). For brevity, we will rewrite the Eq 2.28 as follows:

$$P(\hat{Y}_a(U) = y|X = x, A = a) = P(\hat{Y}'_a(U) = y|X = x, A = a) \tag{2.29}$$

such that $\hat{Y}_a(U) \equiv \hat{Y}_{A=a}(U)$ (i.e., variable x produced given a counterfactual realisation $A = a'$ will be notated as x'_a).

There are 3 levels of assumptions which counterfactual fairness holds by construction. In order of increasing strength, they are:

1. Build $\hat{Y}$ only using the observable non-descendants of A in the causal graph. In many graphs, this may be impossible however, as it may be the case that all observed variables $\mathbf{X}$ are descendants of the protected A.
2. Construct a graphical model including latent variables U which are non-deterministic causes of the observable variables. We then extract these latent variables U using a sampling method (such as MCMC), and then use these to make predictions (i.e., $(x^{(i)}, a^{(i)}) \rightarrow \mathbb{E}[P(U|x^{(i)}, a^{(i)})]$). This deconvolves the data because by construction, the latent variables can't be descendants of the protected attribute.

3. Construct a CGM, which therefore includes deterministic relationships between variables (i.e., $V_j = f_i(pa_j, U) = f_j(pa_j) + \epsilon_j$, where ϵ_j is some Gaussian noise added to V_j). Then use the exogenous variables representing Gaussian noise ($(x^{(i)}, a^{(i)}) \to \epsilon^{(i)}$) as the new features in a predictor, which again by construction do not inherit from the protected attribute.

Observing the Level 3 model makes the connection to counterfactual reasoning (introduced in Section 2.3.1: Counterfactuals) clearest: we perform **abduction** to determine the latent variables U, which are the 'true' representation of some individual, and by construction do not inherit from the protected attribute A. We can then use these latent variables to perform standard learning tasks, instead of the observations. This abduction is therefore the deconvolution described earlier.

The main drawback to counterfactual fairness is that the accuracy of this method relies on our ability to construct an accurate causal model despite having untestable assumptions on its structure. Even with the weakest assumption (Level 1), we still need to assume that there is no causal linkage between some observed variable and the protected attribute, which may not be testable.

Link to Individual Fairness

Counterfactual fairness can be seen to be a way to implement individual fairness, with a distance measure that is defined by the SCM we choose to explain the data. In this case we would hope similar individuals would have similar values for their latent variables.

Path Specific Counterfactual Fairness

It may be the case that within a CGM, we identify certain paths on which the sensitive attribute lies as in fact being fair (see [41, 9, 34]). For example, consider the following CGM (from [9]):

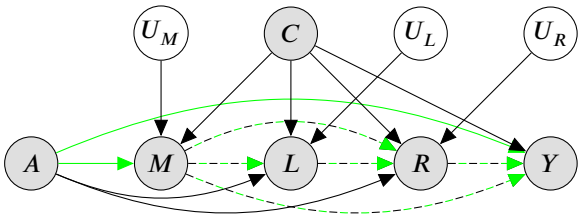


Fig. 2.4 CGM with Fair and Unfair paths

where the green paths are considered unfair, the black paths are considered fair, and the dashed green and black paths are considered unfair only because they are ancestors of the protected attribute A.

In this case, we may deem it such that the paths between the protected attribute and certain variables are actually fair, since it may be that the causal link does not arise due to institutional unfairness. For example, it may be that A represents someone's gender, and L represents someone's level of education. We could argue that any causal link between these two variables is actually fair (assuming the institutions are fair), since this is within the free-will of the individual, and such any causal link simply expresses this free-will.

We therefore wish to express this idea within the decisions the learnt predictor makes. Of course simply learning the latent variables, as in Counterfactual Fairness [32] would achieve fairness in the decisions, but it is too harsh a constraint on the variables we are able to regress on, since we should be able to use information from the protected attribute when making predictions.

The approach detailed by [9] involves the following steps:

1. Calculate the expected value of the output according to a model of the data V.
2. Calculate the path specific effects (PSE) of the model by taking the difference between the estimated output at the baseline value of A and its opposite value (covered in [41]).
3. Subtract the PSE from the expected value of Y, i.e., $\mathbb{E}[Y|V\backslash Y] - PSE(V)$, thus forming the estimate of the new path-specific counterfactually fair Y.

As highlighted in [34], it is not clear what is being optimised in this projection, since we are simply transforming the entire data, including the target variable into a new domain, thus we simultaneously learn a new prediction of Y. This is in contrast to the original approach [32], which transforms the inputs and then allows us to optimise for a measure, such as least squares, over the outcome.

Multi-World Fairness

We make the strong assumption that the CGM assumed is a correct representation of the data-generating mechanism, and it is often the case that certain assumptions about the structure are not testable from the observed data. As a way to ameliorate this, [47] proposes that we create several admissible CGMs to explain the data, and then train a classifier f to be fair

across all these worlds simultaneously:

$$\min_f \frac{1}{n}\sum_{i=1}^{n} \ell\left(f(\mathbf{x}^{(i)}, a^{(i)}), y^{(i)}\right) + \lambda \sum_{j=1}^{m} \frac{1}{n}\sum_{i=1}^{n} \sum_{a' \neq a^{(i)}} \mu_j(f, \mathbf{x}^{(i)}, a^{(i)}, a') \tag{2.30}$$

where μ_j is:

$$\mu_j(f, \mathbf{x}^{(i)}, a^{(i)}, a') = \mathbf{1}\left[|f(\mathbf{x}^{(i)}_{a^{(i)}}, a^{(i)}) - f(\mathbf{x}^{(i)}_{a'}, a')| \geq \epsilon\right] \tag{2.31}$$

$$\approx \max\{0, |f(\mathbf{x}^{(i)}_{a^{(i)}}, a^{(i)}) - f(\mathbf{x}^{(i)}_{a'}, a')| - \epsilon\} \tag{2.32}$$

where Eq 2.32 is the tightest bound to Eq 2.31 that is convex, and **1** is the indicator function. In the case that we cannot deterministically estimate $\mathbf{x}^{(i)}_{a'}$ (i.e., new samples of the data but under the counterfactual world where $A = a'$), we must use samples $\mathbf{x}^{(i)s}_{a'}$ and average:

$$\mu_j(f, \mathbf{x}^{(i)}, a^{(i)}, a') = \frac{1}{S}\sum_{s=1}^{S} \max\{0, |f(\mathbf{x}^{(i)}_{a^{(i)}}, a^{(i)}) - f(\mathbf{x}^{(i)s}_{a'}, a')| - \epsilon\}. \tag{2.33}$$

Eq 2.31 represents the counterfactual difference between a prediction made on an individual i having features $(\mathbf{x}^{(i)}_{a^{(i)}}, a^{(i)})$, and their 'counterfactual' counterpart having features $(\mathbf{x}^{(i)}_{a'}, a')$. For a fair predictor f, we would want this difference to be as close to 0 as possible across: a) all the different graphical models of the world; b) all counterfactual worlds within those graphical models; c) all different individuals.

The first term in Eq 2.30 is a standard loss term for a predictor f predicting on y. The second term is the aforementioned measurement for fairness, which we now sum over each world/graph (indexed by j) and each counterfactual ($\forall a' \in \mathscr{A} \setminus a^{(i)}$). Finally, λ is a tunable parameter which controls how much fairness across the different world (hence **multi-world fairness**) we seek to achieve.

As [47] shows, in the limit of $\epsilon \to 0$ (i.e., we want perfect fairness across all worlds simultaneously), the function f tends to a constant predictor. This is because a predictor that is invariant to inputs is the only way to simultaneously satisfy fairness across all causal graphs in this case. In the other limit (ϵ becomes large) the classifier becomes 'fully unfair', and learns the same classifier as would be learnt with no regularisation term (i.e., $\lambda = 0$).

Whilst this helps address the issue of graphical model misspecification, it is not clear how how we ought to select ϵ (i.e., how much 'unfairness' can we tolerate?).

Causal Interventions for Fairness

In [33] a more root-cause approach to ensuring fairness is motivated by addressing 'decisions' that directly impact society, such as those concerning funding and resource allocation. This presents a more fundamental approach to fairness, as opposed to the majority of other approaches, which aim to ameliorate (subjectively) more symptomatic occurrences of unfairness (i.e., job/college applications).

To ensure these 'decisions' are fair, [33] recommends the following optimisation criterion:

$$\begin{aligned} \max_{z_1,\dots,z_n} & \sum_{i=1}^{n} \mathbb{E}\left[Y^{(i)}(\mathbf{z}) | A^{(i)} = a^{(i)}, X^{(i)} = x^{(i)}\right] \\ s.t., & \sum_{i=1}^{n} z_i \leq B \\ & G_{ia'} \leq \tau \quad \forall a' \in \mathscr{A}, i \in \{1, \dots, n\} \end{aligned} \tag{2.34}$$

where z represents the interventions/decisions (i.e., allocation of additional budget), $Y^{(i)}$ is some outcome that is desired (i.e., college admission rates), B is some allocation constraint (i.e., total budget). The maximisation of the expected outcome is additionally bounded by $G_{ia'}$, which is defined as the **bounded privilege constraint**:

$$G_{ia'} = \mathbb{E}_{\mathscr{M}}\left[Y^{(i)}_{a^{(i)}}(\mathbf{z}) | A^{(i)} = a^{(i)}, X^{(i)} = x^{(i)}\right] - \mathbb{E}_{\mathscr{M}}\left[Y^{(i)}_{a'}(\mathbf{z}) | A^{(i)} = a^{(i)}, X^{(i)} = x^{(i)}\right]. \tag{2.35}$$

If we can block all confounding between A and $\hat{Y}$ (i.e., through randomised experiments) then we can simplify Eq 2.35 to the following:

$$G_{ia'} = \mathbb{E}_{\mathscr{M}^{\prec}}\left[Y^{(i)}_{a^{(i)}}(\mathbf{z}) | A^{(i)} = a^{(i)}, X^{(i)\prec} = x^{(i)\prec}\right] - \mathbb{E}_{\mathscr{M}^{\prec}}\left[Y^{(i)}_{a'}(\mathbf{z}) | A^{(i)} = a^{(i)}, X^{(i)\prec} = x^{(i)\prec}\right] \tag{2.36}$$

where $\mathscr{M}^{\prec}$ represents the causal model excluding all observed descendants of A apart from Y itself, and $X^{(i)\prec}$ represents the subset of $X^{(i)}$ that are non-descendants of A. This means we no longer need full knowledge of the causal model, just the structural equation for Y (i.e., f_Y), which we can fit to the unbiased data under the non-confounding assumption aforementioned.

We can view $G_{ia'}$ as measuring the inequality due to the protected attribute, where the first term measure in Eq 2.35 measures the expected benefit given the actual allocation of the protected attribute, and the second term measure the expected benefit in the counterfactual world where the protected attribute is forced to be a'. This is akin to reducing inequality

as we believe there to be some unfair 'amplification' effect due to being a member of a privileged group.

We summarise Eq 2.34 as follows: we wish to maximise some desired outcome through the allocation of some resource, however we are restricted by how much of it we can allocate, and we must also ensure that the allocation itself does not promote unfairness due to the protected attribute. We solve this equation exactly using mixed-integer-linear-programming (MILP), and add integer constraints to prevent fractional interventions (i.e., allocating $\frac{1}{3}$ of a teacher to a school).

2.4 Other Notions of Fairness

In addition to the statistical notions of fairness we have introduced above, there exists approaches that use existing research in game theory and economics.

2.4.1 Preferential Fairness/Envy-Freeness

In [6] there are two definitions of fairness based around preferences rather than parity:
Definition 2.4.1. A set of classifiers parameterised by $\theta = \{\theta_a\}_{a \in \mathscr{A}}$ (such that each protected attribute a has its own classifier θ_a) can be considered **Preferred Treatment** if each group sharing a sensitive attribute a benefits more from its classifier θ_a than any other classifier $\theta \neq \theta_a$.

Definition 2.4.2. A set of classifiers parameterised by $\theta = \{\theta_a\}_{a \in \mathscr{A}}$ (such that each protected attribute a has its own classifier θ_a) can be considered **Preferred Impact** if each group sharing a sensitive attribute a benefits more from its classifier θ_a than the equivalent classifier which fulfils Demographic Parity (2.2.2).

Note that **benefit** is defined as the proportion of individuals in each group $a \in \mathscr{A}$ who receives a positive outcome (i.e., receiving a loan).

2.4.2 Procedural Fairness

Procedural Fairness [18] defines forms of fairness which are concerned with the decision making process itself (hence procedural) rather than notions of parity. Importantly, these

scores are derived from human users, and allows us to balance prediction accuracy with human-derived notions of fairness.

Definition 2.4.3. We define the **Feature-apriori fairness** as follows:

$$\text{feature-apriori fairness}\,(\mathscr{F}) = \frac{|\cap_{f\in\mathscr{F}}\,\mathscr{U}_f|}{|\mathscr{U}|} \tag{2.37}$$

where $\bar{\mathscr{F}}$ is the total set of features, $\mathscr{F}\subseteq\bar{\mathscr{F}}$, $\mathscr{U}$ is the set of all users, $\mathscr{U}_f\subseteq\mathscr{U}$ such that $\mathscr{U}_f$ represents the subset of users that find feature f fair to be used without a priori knowledge about how its usage affects the decisions that are made.

We can view the above as the fraction of users who believe that **all** the features f in some subset of features $\mathscr{F}$ are fair.

Definition 2.4.4. We define the **Feature-accuracy fairness** as follows:

$$\text{feature-accuracy fairness}\,(\mathscr{F}) = \frac{|\cap_{f\in\mathscr{F}}\,\mathscr{A}(\mathscr{U}_f,\mathscr{U}_f^A)|}{|\mathscr{U}|} \tag{2.38}$$

where $\mathscr{U}_f^A\subseteq\mathscr{U}$ such that $\mathscr{U}_f^A$ represents the subset of users that find feature f fair to be used if it increases accuracy in the prediction. The function $\mathscr{A}(\mathscr{U}_f,\mathscr{U}_f^A)$ returns $\mathscr{U}_f^A$ if feature f actually increases accuracy, otherwise it returns $\mathscr{U}_f$ (the apriori fair user subset).

Definition 2.4.5. We define the **Feature-disparity fairness** as follows:

$$\text{feature-disparity fairness}\,(\mathscr{F}) = \frac{|\cap_{f\in\mathscr{F}}\,\mathscr{D}(\mathscr{U}_f,\mathscr{U}_f^D)|}{|\mathscr{U}|} \tag{2.39}$$

where $\mathscr{U}_f^A\subseteq\mathscr{U}$ such that $\mathscr{U}_f^A$ represents the subset of users that find feature f fair to be used even if it increases disparity in the predictions. The function $\mathscr{D}(\mathscr{U}_f,\mathscr{U}_f^D)$ returns $\mathscr{U}_f^D$ if feature f actually increases disparity, otherwise it returns $\mathscr{U}_f$ (the apriori fair user subset).

It is then possible to combine these human-derived fairness measures with an optimisation method to maximise accuracy with respect to some threshold on procedural fairness, or maximise fairness with respect to some threshold on accuracy.

Interestingly, [18] show that optimising for procedural fairness actually usually achieves high fairness under parity based measures too.

2.5 Implementing Fairness

Here we briefly outline how the aforementioned fairness measures can be implemented during the machine learning process to ensure fairness in subsequent predictions.

2.5.1 Pre-Processing

This involves transforming the data into a new domain such that subsequent algorithms trained on the resulting transformed data are considered fair. Examples of this include [12, 36, 13, 32].

We cover 2 pre-processing approaches that ensure fair representations of data.

Adversarially Learnt Representations

Using the zero-sum approach introduced by [17], we can arrange the generator/discriminator in such a way that the learnt representations can be shown to fulfil parity based forms of fairness, as first shown in [13] fulfilling Demographic Parity (Section 2.2.2), and extended to other notions of fairness (i.e., equality of opportunity, equalized odds (Section 2.2.4)) in [5, 53, 39].

The encoder takes a new datapoint, and transforms it into a representation (i.e., $X \rightarrow Z$). The discriminator then attempts to determine if a new transformed datapoint (i.e., $z^{(i)} \in Z$) belongs in the sensitive group or not (i.e., male or female). In a fair representation, this terms should exhibit high loss, since we do not want to be able to retrieve the protected attribute from the latent representation. We therefore pass a negative gradient to the generator based on the discriminator loss. To ensure that the encoder doesn't simply learn to create a representation that is pure noise (thus trivially masking the protected attribute), an additional predictor head is added, ensuring that we are able to obtain predictive power on the variable of interest Y (i.e., loan amount), whilst simultaneously masking the protected attribute A.

Variational Representations

It is possible to use another popular deep generative model, the variational autoencoder (VAE) [30, 23], to learn a latent represention Z which fulfils Demographic Parity (Section 2.2.2). [36] achieve this, adding an additional loss term to a semi-supervised variational

autoencoder [29]. In production, we would therefore train a variational 'fair' autoencoder, and use the latent representation Z for all predictions instead of the original data X.

Observing the architecture of the conditional VAE in [29] and [49], the learnt latent representation should not contain any information pertaining to the protected attribute. This is because the latent representation does not need to hold this information, since the decoder can simply retrieve this from the sensitive attribute label, which we provide. In reality this is not the case, and the encoder network still retains information pertaining to the sensitive attribute in the latent representation. This is because to minimise reconstruction loss, the decoder can still leverage information pertaining to the sensitive attribute in the latent space, which 'supplements' the explicit label information we provide.

In order to remove all sensitive attribute information in the latent representation, an additional penalty term is incorporated into the loss function, called a maximum mean discrepancy (MMD). This measures the statistical similarity between two sets of samples. Therefore we ensure that the latent posteriors conditioned on the sensitive group (i.e., $P(Z|a)$) are similar to each other such that $P(Z|a_1) \approx P(Z|a_2)$, thus fulfilling Demographic Parity.

2.5.2 Constrained Training

This involves adding a constraint during the optimisation of an algorithm during training time. Examples of this include [52, 47, 33]. The aim is to produce a predictor which can take biased data and generate fair predictions regardless. We have covered two approaches previously in Section 2.3.3, which are representative of these techniques as a whole.

2.5.3 Post-Processing

This involves modifying unfair **predictions** such that they may be considered fair. Examples of this include [20, 19, 27], and we cover the former.

Equalized Odds Adjustment

In [20] a scenario is described where scores R from a black box predictor are thresholded using t to achieve fairness under the Equality of Opportunity definition. For clarity, we derive

a decision (i.e., granting a loan or not) using the following criterion:

$$\hat{Y} = \mathbf{1}(R > t) \tag{2.40}$$

Since Equalized Odds is the same as matching the true positive and false positive rates, it is useful to consider the ROC curves (which plots the true positive rate (TPR) against the false positive rate (FPR) for different thresholds). We further condition on the protected attribute, defining the following tuple:

$$C_a(t) \stackrel{\text{def}}{=} \left(Pr(\hat{R} > t | A = a, Y = 0), P(\hat{R} > t | A = a, Y = 1)\right). \tag{2.41}$$

We can now geometrically understand how to achieve equalized odds; we need to select a value t for each protected group such that all ROC curves intersect, since it is at these intersections that the TPR and FPR is matched. However this may occur only at end points, or at a point where overall accuracy is severely diminished. To address this, we note that any point in the region bounded by the ROC curves is achievable through randomisation. Concretely, we can push the performance of any predictor towards the baseline (random predictor) by randomising its outputs. This is illustrated in the following figure from [20]:

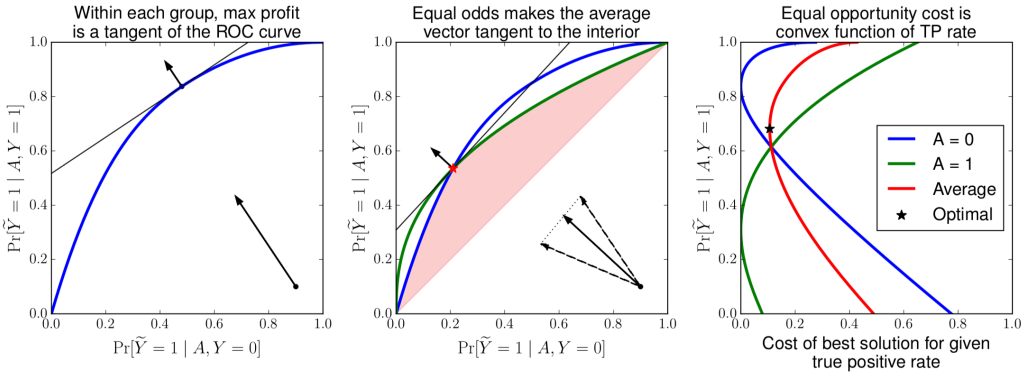


Fig. 2.5 Conditional ROC curves and optimal threshold illustration

In the second subfigure, any point in the red shaded region can be achieved by randomising the predictor conditioned on the protected attribute. The shaded region is not extended below the diagonal since this would represent a predictor that is outperformed by random guessing.

We define the region bounded by each conditional ROC curve as the following convex hull:

$$D_a \stackrel{\text{def}}{=} \text{convhull}(C_a(t) : t \in [0, 1]) \tag{2.42}$$

and therefore the shaded region is simply the intersection of both hulls i.e., $\cap_a D_a$.

In order to 'inject' the right amount of randomisation into the blackbox predictors, we note that the optimal point we wish to find lies in the intersection, which we define as $\gamma = (\gamma_0, \gamma_1) \in \cap_a D_a$. To find optimal values for γ_0 and γ_1, we solve the following optimization problem:

$$\min_{\forall a : \gamma \in D_a} \gamma_0 \ell(1,0) + (1-\gamma_1)\ell(0,1) \tag{2.43}$$

where $\ell(\hat{Y}, Y)$ is a loss function such that $\ell(0,0) = \ell(1,1) = 0$. Having found the optimal values for γ, we can determine if the point lies on the vertex of each D_a, or if it lies within it. In the case of the latter, we inject randomness by having a stochastic threshold. By writing the predictor as $\hat{Y} = \mathbf{1}(R > T_a)$, we make T_a a variable threshold. We can define two thresholds $\underline{t}_a$ and $\bar{t}_a$ such that $0 \leq \underline{t}_a < \bar{t}_a \leq 1$. Now if $R < \underline{t}_a$, we set $\hat{Y} = 0$, and if $R > \bar{t}_a$, we set $\hat{Y} = 1$, and if $\underline{t}_a < R < \bar{t}_a$, we 'flip a coin' and set the value of $\hat{Y}$ according to this outcome. Thus we find values $\underline{t}_a$ and $\bar{t}_a$ to achieve the desired randomness to achieve the desired TPR and FPR for a given predictor and protected group.

Chapter 3

Related Work

3.1 Causal Fairness

Causal fairness has been covered within Section 2.3. In the remainder of this thesis we focus on counterfactual fairness [32] and its extension, multi-world fairness [47].

3.2 Deep Learning in Fairness and Causality

We outline the current state of deep learning in both fairness and causality.

3.2.1 Fairness

We have covered two popular deep generative approaches to fairness in Section 2.5.1. Briefly, both generative adversarial networks (GANs) [17] and variational autoencoders (VAEs) [30, 23] ensure fairness by creating an representation of the original data in which the sensitive attribute cannot be distinguished. We then use this representation to train algorithms which must be fair.

However, these two approaches both share the same issue; they rely on purely observational statistics. Therefore any resultant representations only adhere to one of the observational approaches to fairness, whose drawbacks have been discussed.

In order to extend such techniques to adhere to causal notions of fairness, and make use of the compressive effects of the latent representation, we introduce a new VAE/GAN architecture that that can learn counterfactually fair lower-dimensional latent representations in Chapter 5.

3.2.2 Causality

The measurement of causal effects has long been the main motivation behind causality. For example, we may wish to assess the true causal effect of a certain drug treatment, or the benefit experienced by an education programme. As aforementioned, given a perfectly randomised and controlled experiment, we isolate all sources of variation such that the only correlation remaining must be causation. In reality, this is rarely the case, especially for complex scenarios, due to the presence of confounders. This results in an entanglement of the causal effect, and thus biased estimands.

This is illustrated using a classic confounder model:

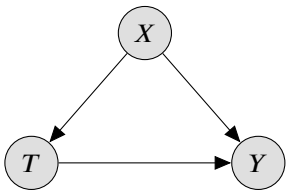


Fig. 3.1 CGM to show confounding

where X represents an individual's features (i.e., age, race, medical history), T represents the treatment they received (i.e., drug, education programme), and Y represents their outcome (this is simply Fig 2.1 with different labels and latent variables not shown).

Deep learning approaches within causality [24, 48, 35] are primarily aimed at disentangling the causal effect. We briefly outline how these work, and explain why they cannot be directly applied to counterfactual fairness, despite superficial similarities.

In these papers, we wish to learn the effect that some treatment t has on the overall population. One way of approaching this is to measure the treatment effect on each individual i (the individual treatment effect (ITE)), and then take its expectation, giving the average treatment effect (ATE). We can express these quantities using counterfactual notation as follows (where $P_{Q=q}$ represents P under the counterfactual intervention $Q = q$):

$$\text{ITE}(x^{(i)}) = Y_{T=1}(x^{(i)}) - Y_{T=0}(x^{(i)}) \tag{3.1}$$

$$\text{ATE} = \mathbb{E}_{x \sim p(x)} \left[\text{ITE}(x) \right]. \tag{3.2}$$

A way to estimate the ITE would be as follows:

$$\hat{\text{ITE}}(x^{(i)}) = \begin{cases} y_F^{(i)} - h(x^{(i)}, 1 - t^{(i)}), & \text{if } t^{(i)} = 1, \\ h(x^{(i)}, 1 - t^{(i)}) - y_F^{(i)}, & \text{if } t^{(i)} = 0, \end{cases} \tag{3.3}$$

where $y_F^{(i)}$ is the factual outcome, $t^{(i)}$ is the factual treatment administered, and $h(x^{(i)}, t^{(i)})$ is a function which estimates the outcome given some individual's features $x^{(i)}$ and $t^{(i)}$. In this case, we estimate the counterfactual quantity by simply 'flipping' the treatment as input into the algorithm, and then calculate its difference compared to the observed y.

Therefore we need to learn a supervised algorithm that maps $h(x^{(i)}, t^{(i)}) \approx y_F^{(i)}$. The biggest issue is the problem of **covariate shift**, such that the data we train our algorithm with does not represent the data at test time. For example, in scenarios where $t^{(i)} = 1$, during training the algorithm see inputs resembling $x^{(i)} \sim P(X|t^{(i)} = 1)$, and similarly when $t^{(i)} = 0$, we only ever train the algorithm with $x^{(i)} \sim P(X|t^{(i)} = 0)$. At test time however, we wish to estimate what results when some $x^{(i)}$ that occurred with $t^{(i)}$ would have received the opposite treatment $1 - t^{(i)}$. Since we have identified a causal link between someone's features $x^{(i)}$ and the treatment they receive $t^{(i)}$, it is possible that given the opposite treatment, we have very few individuals in the training data to train our algorithm on ($P(t|x) \neq P(1 - t|x)$). For example, it may be that we almost exclusively administer one treatment for men, and another for women. Therefore when we try to calculate the counterfactual quantity at test time, it is likely that we have overfit to the very few training cases where we administered the opposite treatment.

By framing the calculation of the ITE in the context of a covariate shift problem, we therefore need to determine a regularisation method. In [24] they suggest this is done via deep representation learning, and propose that an optimal representation for counterfactual calculations ought to fulfil the following:

- Be an accurate predictor of the actual measured effect
- Be an accurate predictor for the counterfactual effect, such that the nearest neighbour in the X space with the opposite treatment is chosen to represent this quantity
- Have a similar distribution conditioned on either treatment.

Therefore, given outputs from hidden layers in neural networks can be seen as abstract representations of the data [3], they build a deep learning based algorithm to simultaneously

learn a predictor h and representation Φ that accurately model outcomes and fulfil the counterfactual representation criteria respectively.

At this stage it is tempting to adopt one of these approaches to model the counterfactual quantities required for counterfactual fairness. This is far simpler than designing the causal model, then inferring the latents via the 'abudction' step. However, as alluded to by [32], it is important to view the causal assumptions made in 'fairness' scenarios. In 'fairness' scenarios, we often obtain graphs with the following structure:

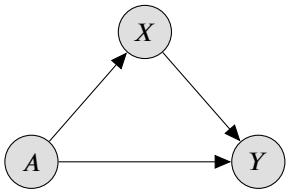


Fig. 3.2 Typical fairness CGM

where A represents the sensitive attribute. This is because we believe there to be a causal path between the sensitive attribute and the observations X, not the other way around. The issue therefore lies with the fact that the treatment, (i.e., the sensitive attribute) actually affects the observed features themselves. For comparison, we rewrite Eq 3.1 interventionally:

$$\mathbb{E}\left[Y|do(A=a), X=x^{(i)}\right] - \mathbb{E}\left[Y|do(A=a'), X=x^{(i)}\right]. \tag{3.4}$$

Comparing this with a counterfactual definition of fairness:

$$\mathbb{E}\left[Y_{a^{(i)}}|X=x^{(i)}, A=a^{(i)}\right] - \mathbb{E}\left[Y_{a'}|X=x^{(i)}, A=a^{(i)}\right]. \tag{3.5}$$

We note that these statements are not equivalent, since interventions do not have enough power to express the latter statement due to the appearance of the counterfactual variable A on both sides of the conditional statement.

If we were to use Eq 3.4 as the basis for fairness, this is the same as requiring the same decision to be made between two individuals who have different sensitive attributes, but have the same observed features. This is likely unfair because an individual from a certain protected group may have needed to work much harder than their contemporaries in other groups to achieve, for example, the same test score due to institutional unfairness.

However, recalling the Level 1 approach outlined in [32], we note that if we obtain a CGM with the following structure:

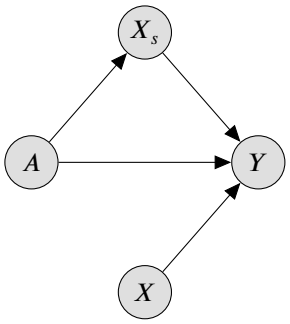


Fig. 3.3 CGM where Level 1 modelling can be applied

we can simply ignore X_s, and directly model on X. We could therefore apply the techniques outlined above, but we note that X is no longer a confounder on the effect of A to Y, therefore the covariate shift issues are eliminated. Instead it is likely better to use standard supervised learning techniques, and either ignore A completely as a feature, or derive a representation using an accurate model of the causal effect which can be shown to be counterfactually fair.

In conclusion, we have established that the predominant deep learning solutions for modelling causal effects are largely not applicable to counterfactual notions of fairness despite the nomenclature. This is primarily due to fundamental differences in the causal graphs. This does not mean however that deep learning approaches cannot be used, and we introduce an example that leverages the flexibility and power of these methods within the context of counterfactual fairness in Chapter 5.

3.3 Model Misspecification

The current literature within causality investigates the effects of model misspecification on estimands, such as treatment effects.

Areas for error include:

- Wrongly placed edges.
- Incorrect functional assumptions along the edges.
- Hidden confounding.

The primary concern is usually **hidden confounders**, since these are relatively easy to admit accidentally (i.e., by not measuring a variable) and can have disastrous results on the final evaluation of causal quantities.

Hidden confounding is illustrated as follows:

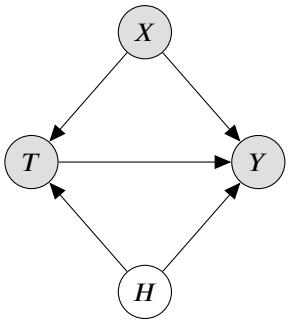


Fig. 3.4 CGM to show hidden confounding

As represented by the absence of shading, the variable H is not observed, yet confounds the effect of T on Y. Therefore not controlling for the effect of H is the same as not controlling for gender in Simpson's Paradox in Section 2.3.1, and in that case would result in us incorrectly calculating the opposite treatment effect.

Two common approaches taken to account for hidden confounders are to: a) bound the estimable causal quantities; b) perform 'sensitivity analysis'. The latter aims to estimate the influence of a potential hidden confounder. We cannot remove hidden confounding entirely, but instead simply account for it in our estimations and understand its effects. This is because we cannot disentangle the causal effect from the confounding effect without access to the hidden confounder itself.

We adopt a 'sensitivity analysis' approach to understanding the effect of model-misspecifications on counterfactual fairness. A common approach to sensitivity analysis is covered in [11]. In summary:

1. Taking some existing causal data, add a randomly synthesised hidden binary confounder H (i.e., coin flip) as an additive linear term into both the treatment selection T (weighted by ζ_t) and the outcome Y (weighted by ζ_y) (see Fig 3.4).
2. Calculate the value of the estimated ATE (Eq 3.2) given no hidden confounding (i.e., $\zeta_t = \zeta_y = 0$), which acts as the the baseline/true value of the treatment effect.
3. Vary the weight of the hidden confounder parameters ζ_t and ζ_y in a grid search, calculating the value of the estimated ATE at each setting.

Key findings within [11] are that: a) hidden confounders can result in the reversal of treatment effects; b) an inadequate assumption of the structural equations (i.e., assuming linear

relationships between variables when there are non-linearities) can also result in the incorrect measurement of the true treatment effect.

However, sensitivity analysis in its current state is not directly applicable to counterfactual fairness measures for two reasons:

- There exists no analogous statistic to 'ATE' or 'ITE' for counterfactual fairness.
- As stated in Section 3.2.2, the causal graphs constructed in epidemiological studies are significantly different to those used within the fairness literature; we must consider an alternative graph to that of Fig 3.4 when performing our hidden confounder measurement.

In Chapter 4, we address both these issues by introducing a new 'Counterfactual Unfairness' metric and a new hidden confounder architecture repsectively. Furthermore, we extend the use of sensitivity analysis beyond hidden confounding, to issues concerning graph topology.

Chapter 4

Novel Counterfactual Fairness Analysis

4.1 Counterfactual Unfairness

Counterfactual fairness is a powerful approach to ensuring fairness, but is affected by model misspecification. In order to understand the effect of **misspecification on fairness**, we must define a measure of counterfactual fairness that can be applied to a variety of different modelling scenarios. There has been no need to define such a measure in the literature previously, since this work has assumed that the given CGM is correct. On the other hand we are looking at precisely the opposite; what happens when the given CGM is in fact wrong?

4.1.1 Counterfactual Data Synthesis

Before we can introduce a metric to calculate counterfactual fairness, we must obtain both factual and counterfactual quantities over the joint distribution $P(A, X, Y)$. This is analogous to the sensitivity analysis literature, whereby there exists a 'true' treatment effect that we must attempt to model as accurately as possible (i.e., efficacy of a drug on survival rates). In the context of counterfactual fairness, we are instead looking to infer the 'true' set latent variables $u^{(i)}$ which define each individual i using the observations $x^{(i)}$ and $a^{(i)}$. Therefore, similar to the sensitivity analysis literature, we introduce a method to synthesise data, thus obtaining a true $u^{(i)}$ for each individual, as well as their corresponding factual and counterfactual observations:

1. Define a graphical model, including priors over latent variables and the relations between variables along the edges.

2. Draw a number of samples from the latent priors $u^{(i)} \sim P(U)$ and select an initial setting of the protected attribute; we choose to allocate half the samples as $a^{(i)} = 1$, and the rest as $a^{(i)} = 0$.

3. Pass these latent variables through graphical model, and obtain the corresponding values for the features $x^{(i)}_{a^{(i)}}$ and outcome variable $y^{(i)}_{a^{(i)}}$ given the settings of $A = a^{(i)}$ obtained previously.

4. Flip the value of A such that we now allocate $1 - a^{(i)} \equiv a'^{(i)}$.

5. Under this counterfactual assignment, repeat the data generation step, producing new values $x^{(i)}_{a'^{(i)}}$ and outcomes $y^{(i)}_{a'^{(i)}}$.

As a result we produce the following 6-tuple for an individual i: $(u^{(i)}, a^{(i)}, x^{(i)}_{a^{(i)}}, y^{(i)}_{a^{(i)}}, x^{(i)}_{a'^{(i)}}, y^{(i)}_{a'^{(i)}})$. For the analysis in this thesis we assume the protected attribute is binary.

4.1.2 Counterfactual Unfairness Metric

Having acquired some 'factual' data X_a, Y_a and its corresponding 'counterfactual' $X_{a'}, Y_{a'}$, we are in a position to define a measure of counterfactual fairness. We call this measure **Counterfactual Unfairness** (**CFU**):

$$\text{CFU} = \frac{\sum_{i=1}^{N} |\hat{Y}(z^{(i)}_{a^{(i)}}) - \hat{Y}(z^{(i)}_{a'^{(i)}})|}{\sum_{i=1}^{N} |y^{(i)}_{a^{(i)}} - y^{(i)}_{a'^{(i)}}|} \tag{4.1}$$

where $z^{(i)}_{a^{(i)}}$ is some input for individual i calculated using the factual data, whereas $z^{(i)}_{a'^{(i)}}$ is some input data for the same individual, but calculated using the counterfactual data. For example, in the fully unfair case $z^{(i)}_{a^{(i)}}$ would be the tuple $\left(a^{(i)}, x^{(i)}_{a^{(i)}}\right)$ (i.e., the raw observations), and $z^{(i)}_{a'^{(i)}}$ would be $\left(a'^{(i)}, x^{(i)}_{a'^{(i)}}\right)$. $\hat{Y}$ is defined as a predictor[1] trained on the data produced by the factual inputs $z^{(i)}_{a^{(i)}}$, and $y_{a^{(i)}}$ is the actual observed factual outcome for individual i, whilst $y_{a'^{(i)}}$ represents the observed counterfactual outcome for that same individual.

The numerator represents the total absolute difference between the prediction on an individual i in the world where they have a protected attribute $a^{(i)}$, and the world where they have the attribute $a'^{(i)}$. A counterfactually fair predictor would be expected to have a very small difference in these predictions, since the data describe the same individual.

[1] In the case of classification, this takes the values of the final decision after thresholding (i.e., 1 or 0)

The denominator can be viewed as a normalisation term, which represents the total unfairness in a given graphical model. We normalise by this value to make it possible to directly compare different realisations of the given graphical model, which is important when modifying the graphical model to explore its sensitivity to misspecifications, as will be seen later.

Therefore we would expect counterfactually fair predictors $\hat{Y}$ trained using data under model m to display a CFU value of near 0, whereas predictors trained to minimise error given the biased data (thus emulating the unfairness in the existing system) to display a CFU value of near 1.

In the case of counterfactual fairness, we usually cannot train the predictor $\hat{Y}$ using the unfair observations, and must instead abduct the latent variables and use these as features instead. In this case, we call the inputs $u_m^{(i)}(a^{(i)})$, which is the abducted latent variable for individual i using the factual data under the CGM m. Correspondingly, we also have $u_m^{(i)}(a'^{(i)})$, which again is the abducted latent variable for individual i, except using the counterfactual data. If our assumptions about the causal model are correct, we would expect $u_m^{(i)}(a^{(i)})$ and $u_m^{(i)}(a'^{(i)})$ to be very similar in value since they represent the same individual, albeit potentially possessing very different observed values $x^{(i)}_{a^{(i)}}$ and $x^{(i)}_{a'^{(i)}}$ respectively. Using these abductions, we write the CFU measure for the **counterfactual fairness** case as follows:

$$\text{CFU}_m = \frac{\sum_{i=1}^{N} |\hat{Y}(u_m^{(i)}(a^{(i)})) - \hat{Y}(u_m^{(i)}(a'^{(i)}))|}{\sum_{i=1}^{N} |y^{(i)}_{a^{(i)}} - y^{(i)}_{a'^{(i)}}|}. \tag{4.2}$$

4.1.3 Sensitivity Analysis Approach

For the remainder of this Chapter, we explore the sensitivity of counterfactual fairness to various model misspecifications under the CFU metric. We adopt the following approach:

1. Design a true causal model, and generate the data according to Section 4.1.1, with 10,000 training data points and 4,000 testing data points.
2. Measure the CFU of the following 5 predictors:
   - Fully Unfair; train/predict on $z^{(i)}_{a^{(i)}} = \left(a^{(i)}, x^{(i)}_{a^{(i)}}\right)$.
   - Fairness Through Unawareness; train/predict on $z^{(i)}_{a^{(i)}} = x^{(i)}_{a^{(i)}}$.
   - The actual latents; train/predict on $z^{(i)}_{a^{(i)}} = u^{(i)}$.
   - The abducted latents using the **correct** CGM m; train/predict on $z^{(i)}_{a^{(i)}} = u_m^{(i)}(a^{(i)})$.

- The abducted latents using the **incorrect** CGM n; train/predict on $z^{(i)}_{a^{(i)}} = u^{(i)}_n(a^{(i)})$.

3. Repeat from Step 1 for 100 experiments.
4. Change the data generating causal model according to the model misspecification we are testing, and repeat from Step 1.

4.2 Missing Edges

Within causal modelling, directed edges represent the notion that a particular variable is the cause of another. These assumptions are made explicit using structured equations, whereby a child variable C is a function of its parent variable P, which is represented by the edge $P \to C$.

During the construction of a hypothetical causal graph, it is possible to neglect the presence of edges where they may in fact exist. For example, we may decide that a standardised test must be fair to all participants by definition, therefore omit the edge between the sensitive attribute and the test score. However in reality it may be that institutional differences can cause such tests to prejudice people from particular backgrounds. Alternatively, we may neglect to include an edge between observed variables, such as someone's years of marriage and mortgage remaining, since we aren't aware of the fact that married couples tend to become home-owners.

We investigate two scenarios: 1) where an edge is missing between the protected attribute and a variable; 2) where an edge is missing between observed variables.

4.2.1 Missing Edge From A

We determine the effect on counterfactual fairness due to a missing edge between the protected attribute A and an observed variable L. We choose the CGM in Fig 4.1 to represent the true generative model, and propose Fig 4.2, which is incorrect due to a missing edge between the sensitive attribute A and the variable L:

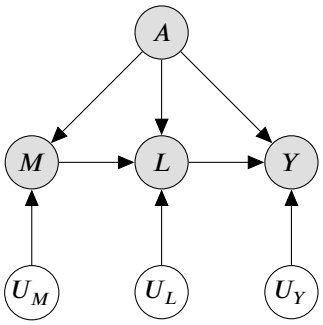


Fig. 4.1 CGM for Missing Edge Experiments

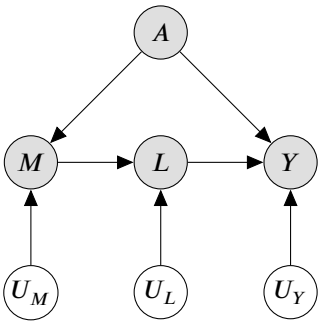
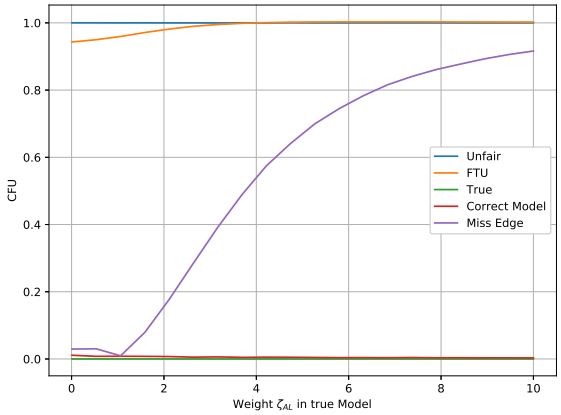


Fig. 4.2 Wrong CGM proposed for Missing Edge Experiments

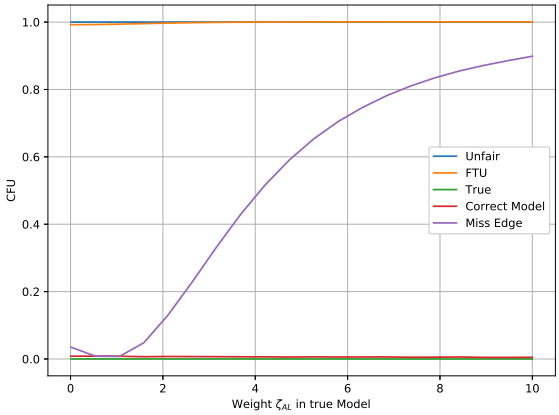
We define the following relationships between the variables:

$$
\begin{aligned}
M|A, U_M &= \zeta_{am}a + u_m \\
L|A, M, U_L &= \zeta_{al}a + +\zeta_{ml}m + u_l \\
Y|A, L, U_Y &= \zeta_{ay}a + \zeta_{ly}l + u_y \\
\mathbf{U} &\sim \mathcal{N}\left[\begin{pmatrix}0\\0\\0\end{pmatrix}, \begin{pmatrix}1 & 0.5 & 0.5\\0.5 & 1 & 0.5\\0.5 & 0.5 & 1\end{pmatrix}\right]
\end{aligned}
$$

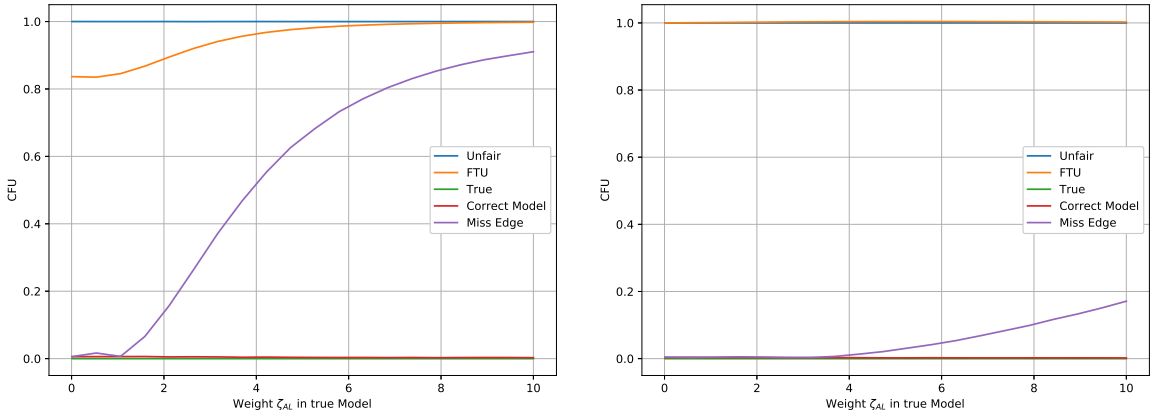
We set weights ζ_{am}, ζ_{ay}, ζ_{ml}, and ζ_{ly} to different values and plot the CFU whilst varying ζ_{al}:



(a) $\zeta_{am} = 2, \zeta_{ay} = 2, \zeta_{ml} = 2, \zeta_{ly} = 2$



(b) $\zeta_{am} = 2, \zeta_{ay} = 2, \zeta_{ly} = 6, \zeta_{ml} = 6$



(c) $\zeta_{am} = 2, \zeta_{ay} = 6, \zeta_{ml} = 2, \zeta_{ly} = 2$ (d) $\zeta_{am} = 6, \zeta_{ay} = 2, \zeta_{ly} = 2, \zeta_{ml} = 2$

Fig. 4.3 CFU against $A \to L$ edge weight

We observe the general trend that as the weight ζ_{al} increases in the true model, the unfairness in the incorrect model increases. Furthermore, we observe that the FTU model is generally unfair, and matches the unfairness of the fully unfair model at high ζ_{al}.

Observing the form of the structural equations, we understand how this unfairness occurs. We propose the following relationships in the correct and incorrect models respectively (with apostrophes representing incorrectly inferred quantities):

$$L|A, M, U_L = \zeta_{al}a + \zeta_{ml}m + u_l \tag{4.3}$$

$$L'|M, U'_L = \zeta'_{ml}m + u'_l \tag{4.4}$$

If we assume that the learnt weights for m (i.e., $\zeta'_{ml} \approx \zeta_{ml}$) are similar, we can rewrite the latent from Eq 4.4 in terms of the latent from Eq 4.3, we obtain the following:

$$u'_l \approx u_l + \zeta_{al}a \tag{4.5}$$

Therefore we see it is inevitable that the wrongly specified model will incorporate some information from the protected attribute into the latent variable during the abduction step, hence the resultant unfairness.

There is also increasing unfairness in the FTU model with ζ_{al}; as the weight ζ_{al} increases, the observations l become stronger proxies for the protected attribute itself. Thus, removing the protected attribute from a predictor makes negligible difference to the final fairness.

We also note that the graphs plotted are relatively invariant to the settings of the other weights in the generative graph. The exception is when ζ_{am} is large (Fig 4.3d), where the effect of the missing edge is less pronounced than in other cases. This is because the incorrectly inferred variable U_L' is, perhaps counterintuitively, **less correlated** with the protected attribute despite the front-door path $A \to M \to L$. This is because M now becomes a strong proxy for A; therefore, regressing on M allows us to effectively regress on A, producing a fair latent variable.

4.2.2 Missing Edge Between Variables

We determine the effect that a missing edge between observed variables has on counterfactual fairness. We choose the graph Fig 4.4 to represent the true generative model, and propose Fig 4.5, which is incorrect due to a missing edge between the variables M and L:

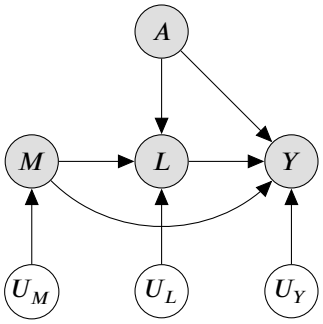


Fig. 4.4 CGM for Missing Edge Experiments

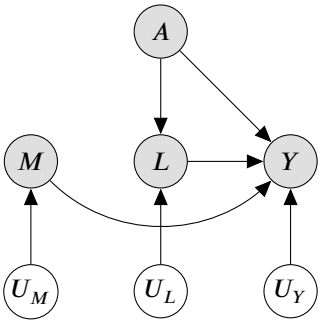
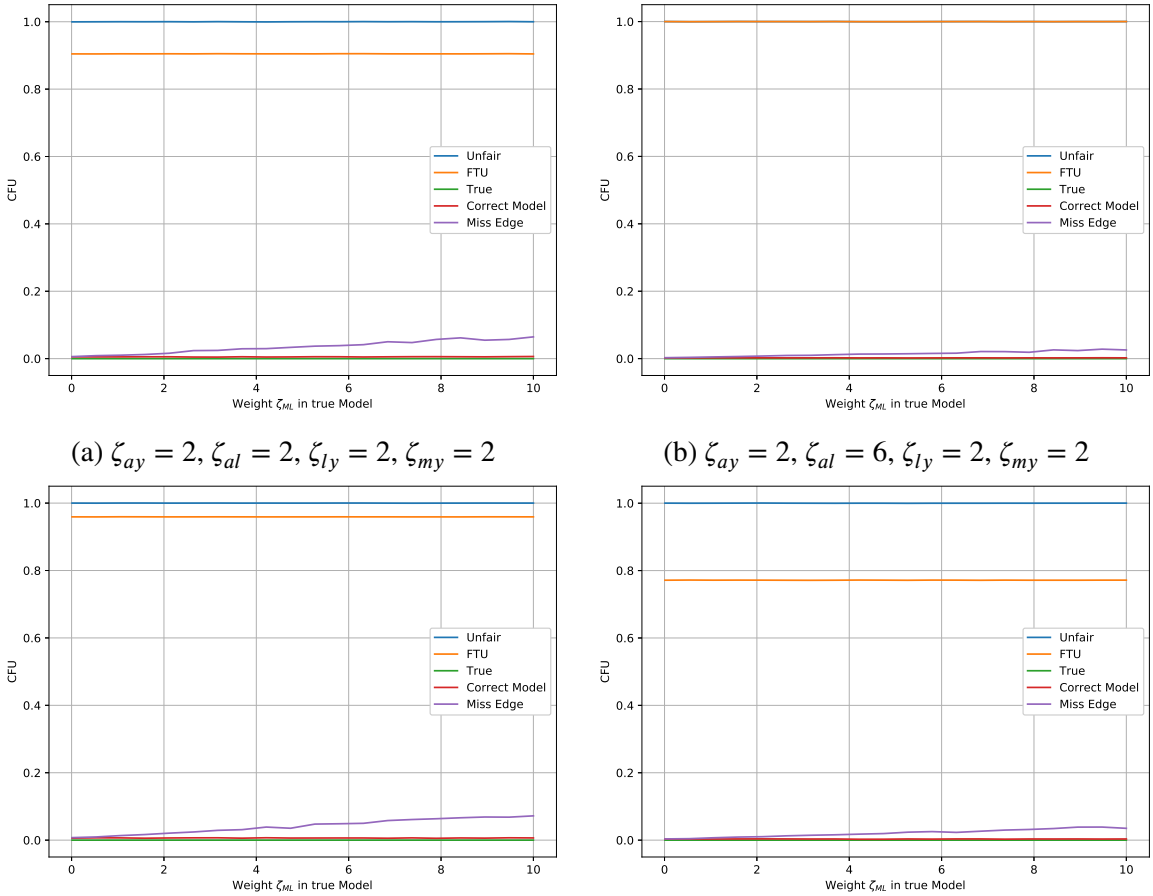


Fig. 4.5 Wrong CGM proposed for Missing Edge Experiments

We define the following relationships between the variables:

$$
\begin{aligned}
M|U_M &= u_m \\
L|A, M, U_L &= \zeta_{al}a + \zeta_{ml}m + u_l \\
Y|A, L, U_Y &= \zeta_{ay}a + \zeta_{my}m + \zeta_{ly}l + u_y \\
\mathbf{U} &\sim \mathcal{N}\left[\begin{pmatrix}0\\0\\0\end{pmatrix}, \begin{pmatrix}1 & 0.5 & 0.5\\ 0.5 & 1 & 0.5\\ 0.5 & 0.5 & 1\end{pmatrix}\right]
\end{aligned}
$$

We set the weights ζ_{ay}, ζ_{al}, ζ_{ly}, and ζ_{my} to different values and plot the CFU whilst varying ζ_{ml}:



(a) $\zeta_{ay} = 2, \zeta_{al} = 2, \zeta_{ly} = 2, \zeta_{my} = 2$

(b) $\zeta_{ay} = 2, \zeta_{al} = 6, \zeta_{ly} = 2, \zeta_{my} = 2$

(c) $\zeta_{ay} = 2, \zeta_{al} = 2, \zeta_{ly} = 6, \zeta_{my} = 2$

(d) $\zeta_{ay} = 6, \zeta_{al} = 2, \zeta_{ly} = 2, \zeta_{my} = 2$

Fig. 4.6 CFU against $M \to L$ edge weight

We notice there is much less impact on the overall unfairness by missing out on edges between variables. This is likely because the variable M doesn't inherit from A, and therefore missing its impact on L does not bias the results heavily.

Even if M was to inherit from A, we simply regress out the proxy effect of A in our model when abducting U_L, therefore there is little to no impact in this case either.

4.3 Additional Edges

In contrast to neglecting edges, we may choose to specify edges where they may not exist in the true model. This is a weaker assumption than a missing edge; for example, if we assume

a linear form of the structural equation, it is possible for us to remove the effect of an edge within a structural equation by simply setting its coefficient to 0.

Similar to missing edges, we may choose to include edges in the proposed GCM where they may not exist due to a lack of expertise. Consider that we are unsure of whether someone's gender affects their outcomes in a medical test, since we are not aware of the causal mechanisms behind these test, nor are the assumptions testable. We therefore may claim there is an edge to 'play it safe'.

To investigate the impact of additional edges, we use the following graph to generate the data:

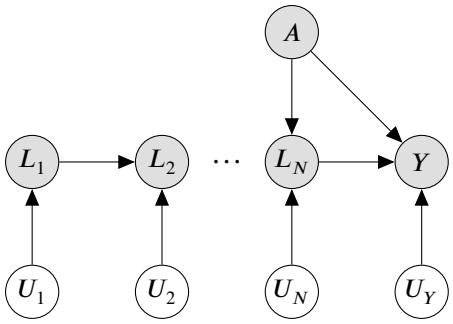


Fig. 4.7 CGM proposed for Additional Edge Experiments

and propose the following incorrect model:

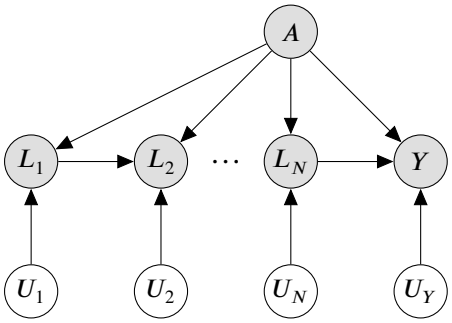


Fig. 4.8 Incorrect CGM proposed for Additional Edge Experiments

We define the $N = 1$ case as follows:

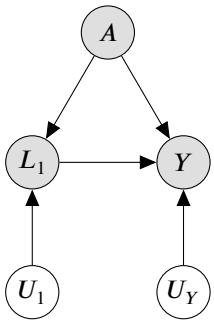
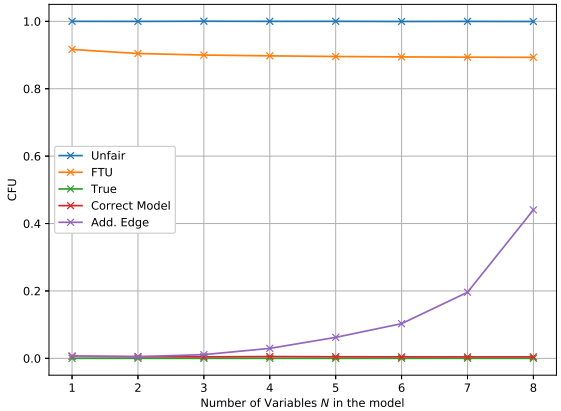


Fig. 4.9 CGM for $N = 1$ case

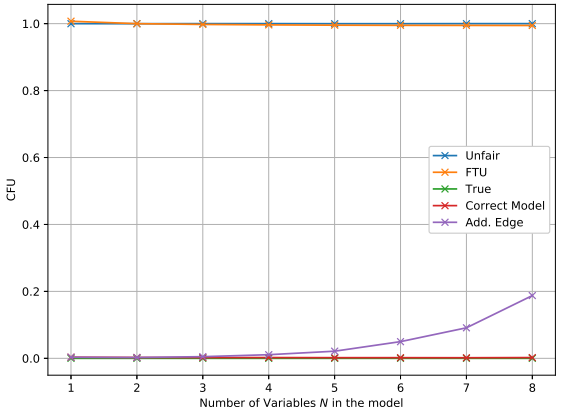
We assume the following relationships between the variables:

$$
\begin{aligned}
L_1|U_1 &= u_1 \\
L_i|L_{i-1}, U_i &= \zeta_l l_{i-1} + u_i \\
L_N|A, L_{N-1}, U_N &= \zeta_{al} a + \zeta_l l_{N-1} + u_N \\
Y|A, L_N, U_Y &= \zeta_{ay} a + \zeta_l l_N + u_y \\
\begin{bmatrix} \mathbf{U} \\ U_Y \end{bmatrix} &\sim \mathcal{N}\left[\begin{pmatrix} 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & \dots & 0.5 & 0.5 \\ \vdots & \ddots & 0.5 & 0.5 \\ 0.5 & 0.5 & 1 & 0.5 \\ 0.5 & 0.5 & 0.5 & 1 \end{pmatrix} \right]
\end{aligned}
$$

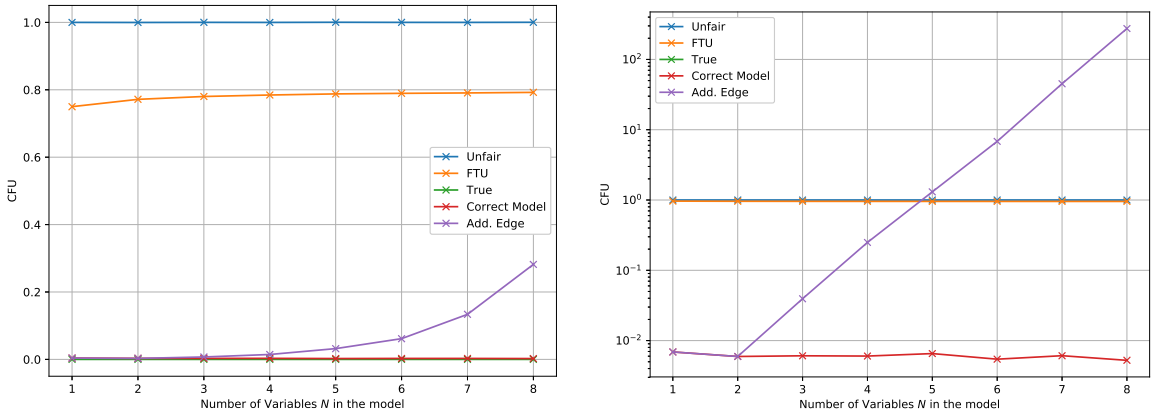
We vary the number of variables between 1 and 8, giving the following CFU plots:



(a) $\zeta_l = 2, \zeta_{al} = 2, \zeta_{ay} = 2$



(b) $\zeta_l = 2, \zeta_{al} = 6, \zeta_{ay} = 2$



(c) $\zeta_l = 2, \zeta_{al} = 2, \zeta_{ay} = 6$ (d) $\zeta_l = 6, \zeta_{al} = 2, \zeta_{ay} = 2$ (Logarithmic y-axis)

Fig. 4.10 CFU against Number of Variables

As before, when the weight ζ_{al} between the sensitive attribute A and the observation L_N becomes large, FTU is as unfair as the fully unfair predictor since the observation becomes a strong proxy for the protected attribute itself.

We notice that in general that when we increase the number of additional variables beyond 5, the unfairness introduced by the additional edges becomes significant in the incorrect model. This is because we assume the following incorrect functional relationship:

$$l_i = \zeta_{al_i} a + \zeta_l l_{i-1} + u_i \tag{4.6}$$

In reality, ζ_{al_i} should be 0 because the true generative model does not feature any functional relationship between A and L_i where $i \neq N$. Therefore it may be expected that the weight ζ_{al_i} is learnt to be 0 when we fit the incorrect model, but due to stochasticity in the data, a small, but non-negligible value is learnt. This explains why, when enough additional variables are included, unfairness increases, since the errors in the learnt model accumulate, resulting in differences between the abducted latent variables for the factuals and counterfactuals where there should be none. The problem is exacerbated when the weight between the variables L_i is increased (Fig 4.10d), since even small perturbations due to stochastic effects result in large ζ_{al_i} terms being learnt, giving vastly unfair predictions.

4.4 Modelling Assumptions

As covered in [32] and Section 2.3.3, there are a number of increasingly strong assumptions we can place on the structural model. In summary:

- Level 1: Build predictions only on observations that do not inherit from the protected attribute.
- Level 2: Postulate that the latent variables act as non-deterministic causes of the observations, defining relations using distributions.
- Level 3: Postulate a fully deterministic structural model, whereby the functional form of all variable relationships is explicitly stated.

The Level 2 model is suitable for when there is uncertainty concerning the true structural relationship between variables in the model; therefore we examine how these weaker assumptions affect both the accuracy and the fairness of the resulting predictions.

To do this, we perform the following:

1. Fit a Level 3 model to some real world data.
2. Generate some new factual data and counterfactual data using the fitted Level 3 model.
3. Fit the Level 2 model to the factual data.
4. Generate counterfactual latents according to the counterfactual data generated in Step 2.
5. Fit a model to the Level 2 factual latent variables, and calculate the CFU between this and the Level 2 counterfactual latents.

We use the law school data and CGMs from the 'Counterfactual Fairness' paper [32]:

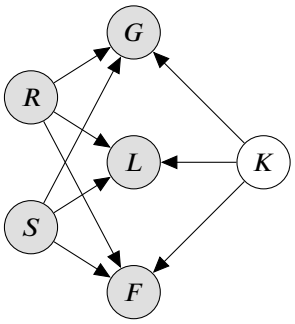


Fig. 4.11 CGM for Level 2 Law School Model

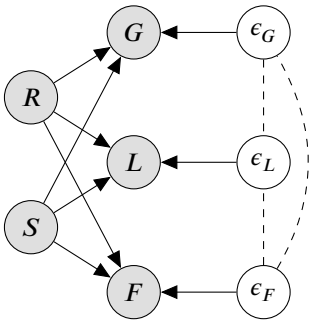


Fig. 4.12 CGM for Level 3 Law School Model

Fitting the Level 3 model to the real data, we are then able to generate factual data and its corresponding counterfactual data. For this we will simplify the data as follows:

- Only generate data for black and white individuals, since these two individuals have the largest difference in their learnt coefficients
- Generate an equal number of black and white candidates
- Assume a 50/50 gender split within each racial group

We obtain the following results:

Model	CFU	RMSE
Constant	0.000	1.033
Unfair	1.007	0.722
FTU	0.839	0.730
L3 Model	0.005	0.892
L2 Model	0.013	0.924

Table 4.1 CFU and RMSE of various fairness techniques

We notice that the Level 2 model introduces a negligible increase of unfairness, but gives a relatively large amount of additional error over the Level 3 model. The latter is not so surprising, since we attempt to capture the effect of 3 latent variables in the true model with a single latent variable K; therefore there is some information that is lost when this abstraction is made. If the ultimate goal however is to produce fair predictions, the increased assumptions made by Level 2 modelling do not appear to affect this, especially if the Level 2 model chosen has the same topology as the true model.

4.4.1 'Counterfactual Fairness': Paper Replication

In order to synthesise the data required within the CFU calculation, we trained our models on the real data used in [32]. This provides us with an opportunity to replicate and therefore validate our own usage of the same techniques in our sensitivity analysis.

In order to replicate the results of [32], we must construct Level 2 and Level 3 models. We first outline how we infer the latent variables for the **Level 2 model**:

1. Prepare the data by a) one-hot encoding the sensitive attributes; b) separating these out from the grade variables; c) splitting the data 80/20 train/test respectively.
2. Build the train-time model in Stan [8] (a probabilistic programming language in C++), placing standard normal priors on all weights, the latent variable K, and an inverse gamma prior on σ_G.
3. Fit the model using MCMC (specifically NUTS), allocating 1,000 iterations with 500 as warmup, and extract the posterior samples of the edge weight values.
4. Build the test-time model in Stan, placing constants on the weights, and a standard normal prior on the latent variable K.
5. Take the mean value of the sampled edge weight posteriors from Step 3, and then use these to set the test-time model edges.
6. Sample from the test-time model to obtain both the train and test latent variables (2,000 iterations, 1,000 warmup).
7. Fit the models to the data, using the mean of the train latents, and test on the mean of the test latent variables (both of which were obtained in Step 6).

Fitting the Level 3 model does not require the constructing a probabilstic model, since we define deterministic structural equations with linear Gaussian form.
The process for fitting the **Level 3 model** is as follows:

1. Fit the linear equations $G = \zeta_{RG}R + \zeta_{SG}S + C_G$ and $L = \zeta_{RL}R + \zeta_{SL}S + C_L$ (where C_G and C_L are constants) using the training data.
2. Infer the latent variables for the training and testing data using ϵ_G and ϵ_L using the learnt parameters ζ as follows: $\epsilon_G = G - (\zeta_{RG}R + \zeta_{SG}S + C_G)$ and $\epsilon_L = L - (\zeta_{RL}R + \zeta_{SL}S + C_L)$.
3. Fit models to the data using the inferred ϵ values as the inputs, instead of the observations.

Having inferred the latents from both models, we achieve the following results in comparison with the original paper, using an ordinary least squares (OLS) model to predict on the variable F:

		Full	**Unaware**	**Fair K**	**Fair Add**	**Fair Both**	**Fair RF**
RMSE	Paper	0.873	0.894	0.929	0.918	-	-
	Repl.	0.870	0.891	0.930	0.917	0.924	0.910

Table 4.2 Paper RMSE vs Replication RMSE

We match the paper results almost exactly, with any variation likely due randomness when splitting the data 80/20, as well as variability within the MCMC sampling method. We also extend the results by introducing 'Fair Both' and 'Fair RF'.

'Fair Both' involves learning an ordinary least squares model to predict F using all abducted latent variables (i.e., ϵ_G, ϵ_L, and K). Unsurprisingly, we don't get improved results since K is simply a noisy estimation of ϵ_G and ϵ_L.

'Fair RF' involves learning a random forest (RF) [7] regressor using the latent variables ϵ_G and ϵ_L. Surprisingly, despite the relative noisiness of the data, and higher variance of the algorithm compared with OLS, we do not overfit on the training data, and instead improve RMSE from 0.917 to 0.910. This lends credence to the idea of using non-linear models to model the functional relationships between variables, since these may provide additional flexibility to reduce bias, whilst not being severely affected by issues of overfitting.

4.5 Hidden Confounders

Here we investigate the presence of a hidden confounder. We add an additional confounder H on some observed variables $\mathbf{L}$, both of which also have a causal link with the sensitive attribute A. This is closely related to the notion of unresolved proxies, which is covered in [26], whereby the hidden confounder H is an unresolved proxy. We illustrate this relation in the following graphical model (N.B., we omit the latents for clarity):

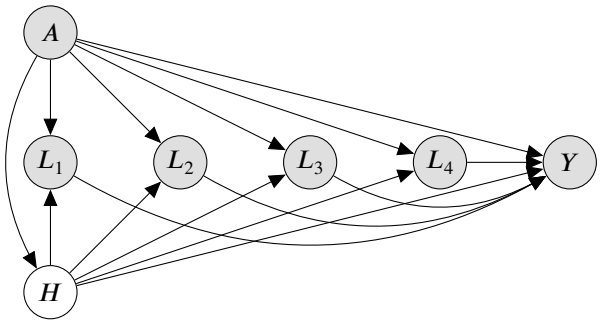


Fig. 4.13 CGM used for Hidden Confounding

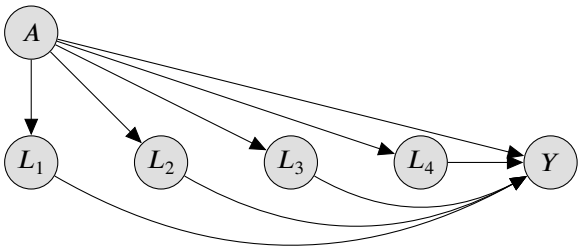


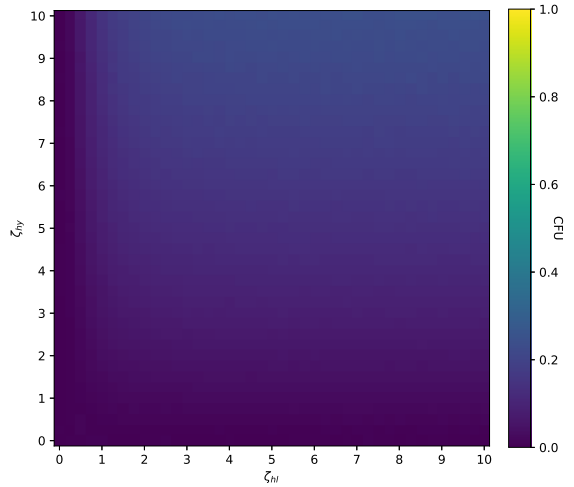
Fig. 4.14 Incorrect CGM proposed for Hidden Confounding

A scenario where this may occur would be where A represents someone's race, $\mathbf{L}$ represents someone's test scores, Y could be someone's university admittance score, and H is an unmeasured variable which could represent which kind of school someone goes to (i.e., private or state).

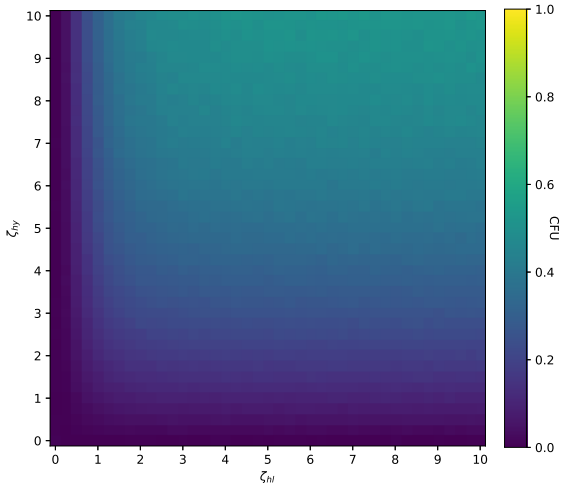
We define the following relationships between variables in the graphical model:

$$
\begin{aligned}
Y|\mathbf{L}, H, A &\sim \mathcal{N}\left(\zeta_{ly}(l_1 + l_2 + l_3 + l_4) + \zeta_{hy}h + \zeta_{ay}a, 1\right) \\
L_1|A, H, U_1 &= \zeta_{al_1}a + \zeta_{hl}h + u_1 \\
L_2|A, H, U_2 &= \zeta_{al_2}a - \zeta_{hl}h + u_2 \\
L_3|A, H, U_3 &= \zeta_{al_3}a + \zeta_{hl}h + u_3 \\
L_4|A, H, U_4 &= \zeta_{al_4}a - \zeta_{hl}h + u_4 \\
H|A &\sim \text{Bernoulli}(0.25 + 0.5A) \\
\mathbf{U} &\sim \mathcal{N}\left[\begin{pmatrix}0\\0\\0\\0\end{pmatrix}, \begin{pmatrix}1 & 0.5 & 0.5 & 0.5\\ 0.5 & 1 & 0.5 & 0.5\\ 0.5 & 0.5 & 1 & 0.5\\ 0.5 & 0.5 & 0.5 & 1\end{pmatrix}\right]
\end{aligned}
$$

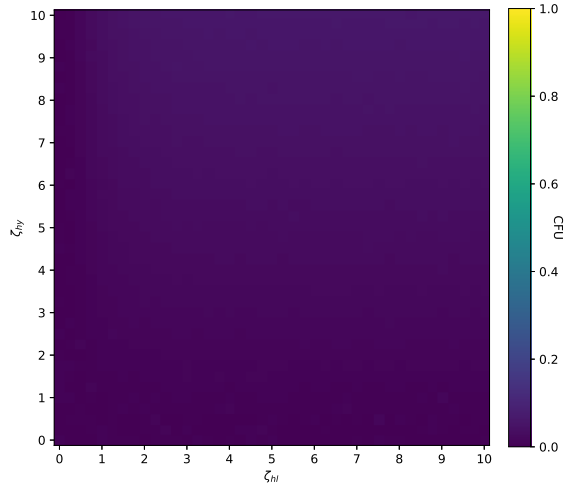
In the above, H represents the hidden confounder, and following the approach in [11], we apply two weights ζ_{hl} and ζ_{hy} which represent the confounding strength on the variables $\mathbf{L}$ and Y respectively. We obtain the following:



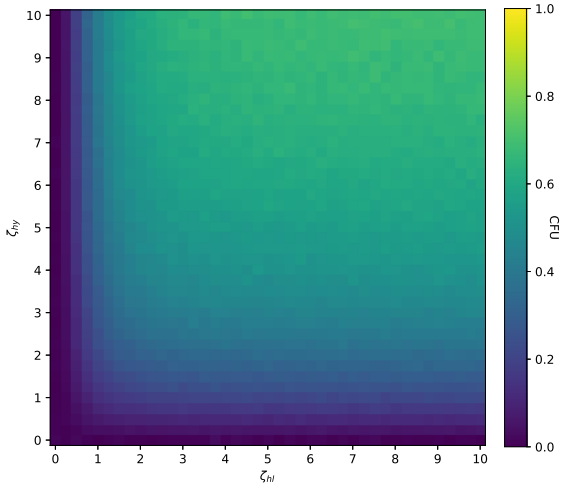
(a) $\zeta_{al_1} = 2, \zeta_{al_2} = 2, \zeta_{al_3} = 2, \zeta_{al_4} = 2, \zeta_{ly} = 2, \zeta_{ay} = 2$



(b) $\zeta_{al_1} = 2, \zeta_{al_2} = 2, \zeta_{al_3} = 2, \zeta_{al_4} = 2, \zeta_{ly} = 0.5, \zeta_{ay} = 1$



(c) $\zeta_{al_1} = 2, \zeta_{al_2} = 2, \zeta_{al_3} = 2, \zeta_{al_4} = 2, \zeta_{ly} = 10, \zeta_{ay} = 2$



(d) $\zeta_{al_1} = 0.5, \zeta_{al_2} = 0.5, \zeta_{al_3} = 0.5, \zeta_{al_4} = 0.5, \zeta_{ly} = 0.5, \zeta_{ay} = 1$

Fig. 4.15 CFU against Hidden Confounding Weights

We also show the CFU measure against confounding weights in the event we had access to the hidden confounder in the Appendix (Fig A.1), which is 0 for all weight settings.

Overall we find that hidden confounders can have a significant effect on the final fairness of predictions when the hidden confounding is relatively strong. We illustrate why, starting with the correct and incorrect form of the equation for L_3:

$$l_3 = \zeta_{al_3} a + \zeta_{hl} h + u_3 \tag{4.7}$$

$$l_3 = \zeta'_{al_3} a + u'_3 + b \tag{4.8}$$

where the apostrophised quantities are incorrectly inferred with repsect to the true value, and b is a bias term the incorrect model can learn to compensate for the lack of access to the hidden confounder. Assuming our correct model (with access to H) is able to retrieve the true form of Eq 4.7, and equating Eqs 4.7 and 4.8, we obtain the following:

$$u'_3 - u_3 = \zeta_{hl} h + \zeta_{al_3} a - \zeta'_{al_3} a - b \tag{4.9}$$

$$\mathbb{E}[u'_3 - u_3] = \zeta_{hl}(0.25 + a) + \zeta_{al_3} a - \zeta'_{al_3} a - b \tag{4.10}$$

$$\mathbb{E}[u'_3 - u_3] = (0.5\zeta_{hl} + \zeta_{al_3} - \zeta'_{al_3})a + (0.25\zeta_{hl} - b). \tag{4.11}$$

We observe that we have enough degrees of freedom in ζ'_{al_3} and b to, abduct values of u'_3 such that its expected difference between the true latent variable is 0; indeed this is what happens when the incorrect model is trained with strong hidden confounding. However taking the expectation over all individuals precisely defeats the point of counterfactuals; we wish to determine individual level, not population level, quantities. We first consider the factual and counterfactuals of l_3 for an individual with $U_3 = u_3$ such that factually $A = 1$:

$$l_3 = \zeta_{al_3} + \zeta_{hl} h + u_3 \tag{4.12}$$

$$l_{3_{cf}} = \zeta_{hl} h_{cf} + u_3. \tag{4.13}$$

Consider the following abduction of an individual's u_3 under the incorrect model using both l_3 and $l_{3_{cf}}$ (i.e., obtaining u'_3 and $u'_{3_{cf}}$ respectively):

$$u'_3 = \zeta_{al_3} + \zeta_{hl} h - \zeta'_{al_3} + u_3 - b \tag{4.14}$$

$$u'_{3_{cf}} = \zeta_{hl} h_{cf} + u_3 - b. \tag{4.15}$$

If we assume that in Eq 4.11 we learn parameters ζ'_{al_3} and b such that the expected latent difference is 0, we obtain the following counterfactual difference between the abducted vari-

ables in Eqs 4.14 and 4.15:

$$u'_3 - u'_{3_{cf}} = \zeta_{hl}(h - h_{cf}) + \zeta_{al_3} - \zeta'_{al_3} \tag{4.16}$$

$$= \zeta_{hl}(h - h_{cf}) + \zeta_{al_3} - 0.5\zeta_{hl} - \zeta_{al_3} \tag{4.17}$$

$$= \zeta_{hl}(h - h_{cf} - 0.5). \tag{4.18}$$

Therefore the difference in the abducted latent under the counterfactual is non-zero for all $h, h_{cf} \in \{0, 1\}$, which results in different latents abducted for the factual and counterfactual data. This explains why when either ζ_{hl} or ζ_{hy} are 0, there is no unfairness; in the former case, the incorrectly abducted variables will be identical under the factual or counterfactual, and in the latter case, the outcome will display no variation with the hidden confounder, hence the variation in the abducted latent variable will not affect the final prediction.

Interestingly, if H does not inherit from A, whilst there is a loss of accuracy, there is no loss in fairness. This is in contrast to the causal literature, whereby any hidden confounding can result in bias in the calculated causal effects [11]. In the case of counterfactual fairness, whilst we lose accuracy, we do not lose fairness because H now simply adds noise, but will not bias the decision due the protected attribute A, since it is not an unresolved proxy [26].

4.6 Applying CFU to Other Fairness Techniques

The CFU measure allows the comparison of observational fairness techniques, such as Equality of Opportunity or Demographic Parity.

Note that we have already compared Fairness through Unawareness (FTU) in the previous sections, and have observed that in general it performs marginally better or as well as a fully unfair predictor under the CFU measure. We now extend our analysis to 3 other forms of fairness, covered in Chapter 2:

- Demographic Parity (DP)
- Equalized Odds (EOdds)
- Equality of Opportunity (EOpp)

We implement DP using a constrained optimisation approach, learning weights which enforce a classification boundary that produces DP estimates on the data. We implement both

EOdds and EOpp with post-processing methods using convex optimisation to re-weight unfair estimates.

4.6.1 Results

We choose the following graph to generate our data:

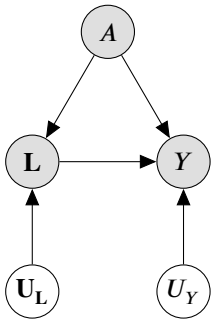


Fig. 4.16 Generic Fairness CGM

We define the following relationships between variables:

$$
\begin{aligned}
Y|\mathbf{L}, A, U_Y &= \operatorname{sgn}\left(\boldsymbol{\zeta}_{ly}^{\intercal}\mathbf{L} + \zeta_{ay}a + u_y\right) \\
\mathbf{L}|A, \mathbf{U} &= \boldsymbol{\zeta}_{al}a + \mathbf{u} \\
\begin{bmatrix}\mathbf{U} \\ U_Y\end{bmatrix} &\sim \mathcal{N}\left[\begin{pmatrix}0 \\ \vdots \\ 0 \\ 0\end{pmatrix}, \begin{pmatrix}1 & \dots & 0.5 & 0.5 \\ \vdots & \ddots & 0.5 & 0.5 \\ 0.5 & 0.5 & 1 & 0.5 \\ 0.5 & 0.5 & 0.5 & 1\end{pmatrix}\right]
\end{aligned}
$$

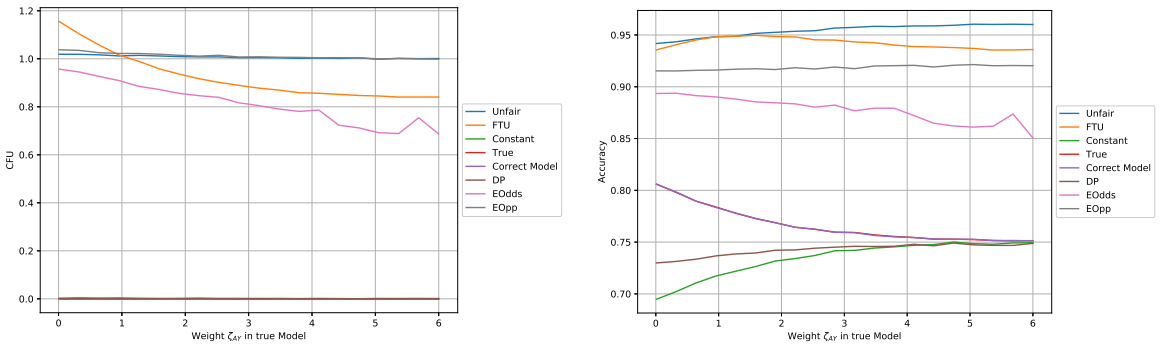
where $\operatorname{sgn}(x)$ is a function which returns $+1/-1$ for positive and negative x respectively. Note therefore that we are performing classification as opposed to regression.

We generate the data and perform testing as before. In this case, we make the following design choices:

- Select the dimensionality of $\mathbf{L}$ to 2.
- To promote stability in the optimization schemes, we train on 400,000 data points and test on 20,000.
- When varying $\boldsymbol{\zeta}_{al}$, we parametrise it as follows: $\boldsymbol{\zeta}_{al} = \zeta_{al} \cdot [2, 1]^{\intercal}$.

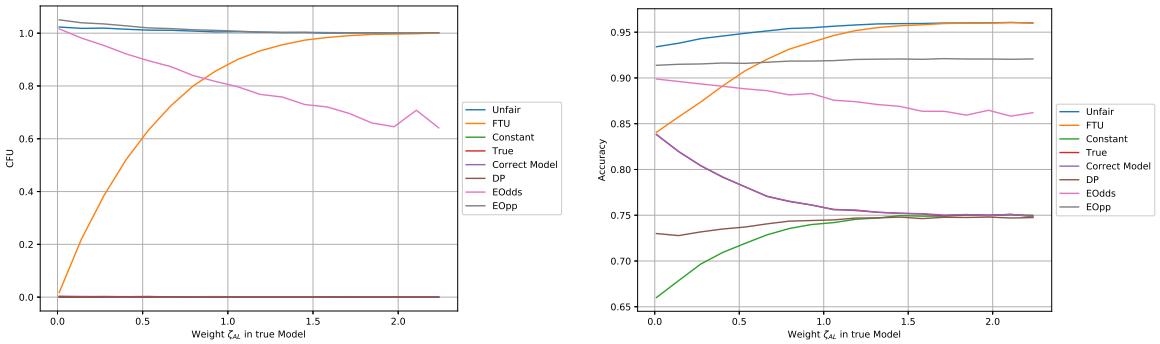
- When varying $\boldsymbol{\zeta}_{ly}$, we parametrise it as follows: $\boldsymbol{\zeta}_{ly} = \zeta_{ly} \cdot [1, 2]^\intercal$.
- Use logistic regression as the final model.

We obtain the following results, varying each weight separately, and measure the CFU and accuracy:



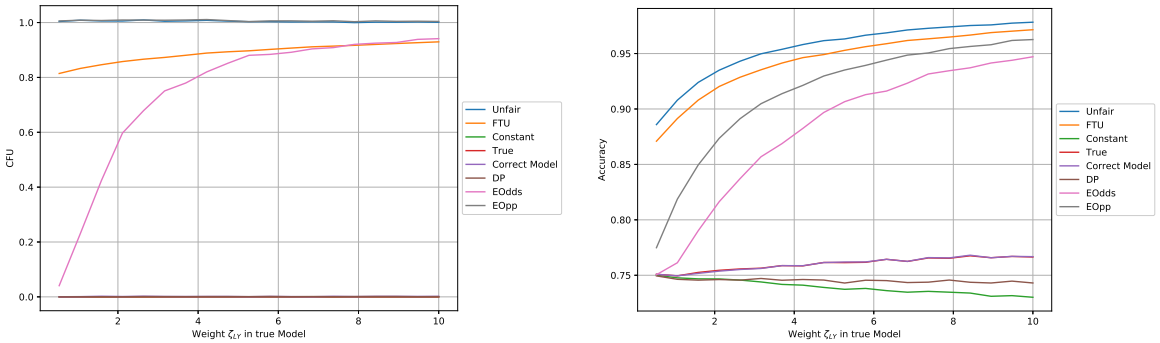
(a) CFU: $\boldsymbol{\zeta}_{ly} = [2, 1]^\intercal$, $\zeta_{al} = [1, 2]^\intercal$ (b) Accuracy: $\boldsymbol{\zeta}_{ly} = [2, 1]^\intercal$, $\zeta_{al} = [1, 2]^\intercal$

Fig. 4.17 CFU and Accuracy against weight ζ_{ay}



(a) CFU: $\zeta_{ay} = 3$, $\zeta_{ly} = [2, 1]^\intercal$ (b) Accuracy: $\zeta_{ay} = 3$, $\zeta_{ly} = [2, 1]^\intercal$

Fig. 4.18 CFU and Accuracy against weight ζ_{al}



(a) CFU: $\zeta_{ay} = 3, \zeta_{al} = [1, 2]^\intercal$ (b) Accuracy: $\zeta_{ay} = 3, \zeta_{al} = [1, 2]^\intercal$

Fig. 4.19 CFU and Accuracy against weight ζ_{ly}

An immediate observation is that under this generative model, Demographic Parity produces predictions that are also counterfactually fair. However, it is also the case that these predictions are less accurate than the ones learnt using the causal model (i.e., 'Correct Model'). In the case of Counterfactual Fairness, considering our final predictor makes use of the exogenous latents U, which are independent of A by construction, we are in fact producing predictions which satisfy Demographic Parity, since the relation $P(Y|A = a) = P(Y|A = a')$ must hold. In this sense, Counterfactual Fairness can be considered a counterfactual version of Demographic Parity (also stated in [32]). Therefore, by leveraging the causal model, we extract all the 'fair' information of an individual, as opposed to in Demographic Parity, where we must marginalise over individuals in order to ensure parity, thereby losing information:

$$\int P(Y = y|X, A = a)P(X|A = a)dX = \int P(Y = y|X, A = a')P(X|A = a')dX. \tag{4.19}$$

We observe that in general, EOpp and EOdds tend to provide unfair predictions under counterfactual fairness, with the former almost always matching the unfairness of the fully unfair predictor, but producing less accurate predictions due to the additional constraint. We have covered previously why both these methods provide poor counterfactual fairness, due to their inability to remove historic biases, and now provide experimental proof that this is the case. This is why despite the fact counterfactual fairness adheres to Equalized Odds (see [34]), the reverse is not true; we can construct predictors which adhere to Equalized Odds but have more accurate predictions, at the expense of counterfactual fairness.

We also observe that when the strength of the protected attribute is high in the causal graph, (i.e., right hand side of Figs 4.17b and 4.18b), the counterfactually fair and Demographic

Parity predictions tend towards the constant predictor. This is because the predictive variable becomes independent of all variables apart from A, therefore a constant predictor is the only way to ensure counterfactual fairness.

Finally, we observe that there is a close relationship between counterfactual unfairness and accuracy for the non-counterfactually fair predictions, with less fair predictors producing more accurate predictions. This is because the protected attribute A is usually predictive of the outcome Y, and since the CFU metric measures the invariance of an individual's prediction to the protected attribute A, it is not surprising that less invariant predictors produce more accurate predictions.

Chapter 5

Variational Counterfactual Fairness

5.1 Introduction

We propose a novel VAE design to learn a counterfactually fair latent space, which can be extended to multiple worlds. We show that this allows us to compress the latent space without compromising accuracy, and also allows for a single latent representation of multiple causal models, which simplifies the modelling process. Furthermore, this shows the applicability of neural networks and variational inference to the learning of causal mechanisms.

Conceptually our design is related to the Variational Fair Autoencoder [36], except the MMD term is replaced with the regularisation term in [47], and there are adjustments to the way losses are back-propagated in the network which ensure that the representation learnt is itself counterfactually fair, yet predictive of the final objective. Furthermore, we jointly learn the fair predictor itself, represented by f. For clarity, we present MWF loss term [47]:

$$\mu_w(f, \mathbf{x}^{(i)}, a^{(i)}, a') = \max\{0, |f(q(\mathbf{x}^{(i)}_{a^{(i)}}, a^{(i)})) - f(q(\mathbf{x}^{(i)}_{a'}, a'))| - \epsilon\}. \tag{5.1}$$

$q(\mathbf{x}^{(i)}_{a^{(i)}}, a^{(i)})$ represents the encoder output $z^{(i)}_{a^{(i)}}$ (hence the predictor f only sees this representation, not the data). During training, we freeze the weights for function f when we back-propagate this term so that the predictor itself is not regularised to be fair, ensuring all fairness is due to the latent representation. Finally, we include the index w, as in [47] (where it is written as j), which represents the world we assess this loss term over.

5.2 Architecture

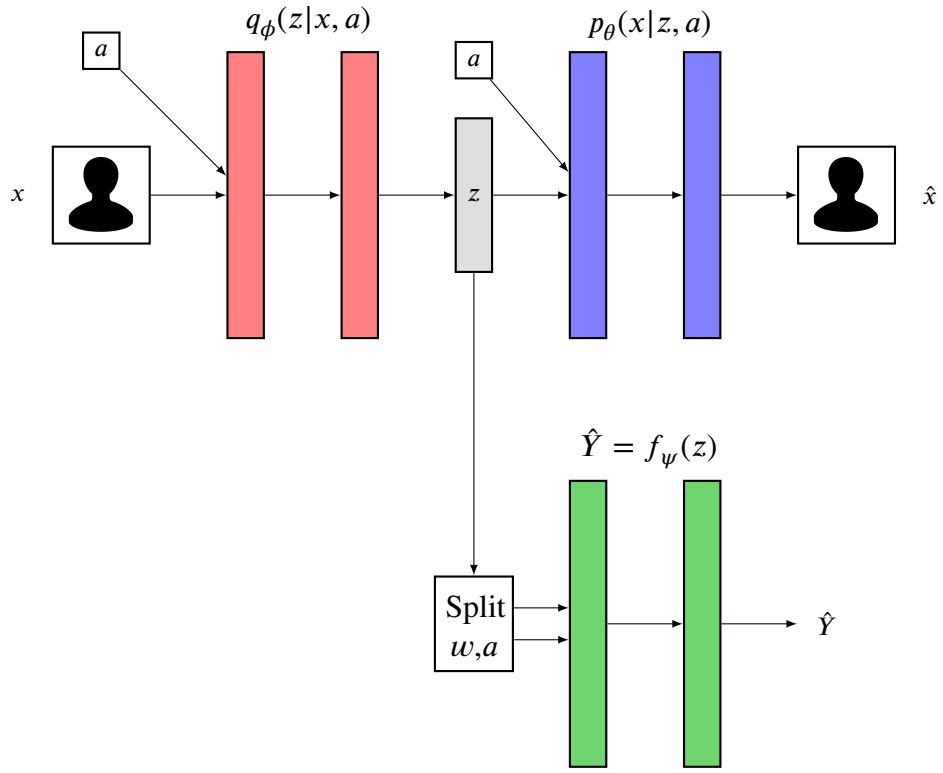


Fig. 5.1 Multi-World Fair Adversarial Autoencoder

The architecture is shown in Fig 5.1, and we call this the **Multi-World Fair Adversarial Autoencoder** (**MWF-AAE**). We have two alternating back-propagation steps:

1. Train just the autoencoder using standard variational inference (freezing the weights of f), back-propagating the following terms:
   - ELBO [30]; this ensures that the learnt representation contains information pertaining to the data thus can be used for reconstructive purposes.
   - f loss; this ensures that we encourage representations that have predictive power (otherwise we may just learn noise).
   - μ_w loss; this ensures that the representation is be counterfactually fair given a predictions $\hat{Y}_a$, $\hat{Y}_{a'}$ in world w.
2. Train only the predictor f on the factual data using the representation learnt by the autoencoder, freezing the VAE weights

This can be considered adversarial since the predictor is likely to discriminate based on the protected attribute as this will reduce loss (discussed in Section 4.6.1). By backpropagating the μ_w term, we therefore force the representation to be counterfactually fair w.r.t. the protected attribute, which drives the μ_w term to 0.

We represent the objective functions for each step (denoted $\mathscr{F}_1$ and $\mathscr{F}_2$ respectively) as follows:

$$\begin{aligned}\mathscr{F}_1(\phi, \theta, \psi, \mathbf{x}^{(n)}, a^{(n)}, \mathbf{y}^{(n)}) = \sum_{n=1}^{N} & \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}^{(n)}|\mathbf{x}^{(n)}, a^{(n)})}\left[\log p_\theta(\mathbf{x}^{(n)}|\mathbf{z}^{(n)}, a^{(n)})\right] - \mathrm{KL}\left(q_\phi(\mathbf{z}^{(n)}|\mathbf{x}^{(n)}, a^{(n)})||p(\mathbf{z})\right)}_{\text{ELBO}} \\ & - \alpha \cdot \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}^{(n)}|\mathbf{x}^{(n)}, a^{(n)})}\left(\ell(f_\psi(\mathbf{z}^{(n)}), y^{(n)})\right)}_{\text{Predictor } (f) \text{ Loss}} \\ & - \beta \cdot \sum_{w=1}^{m} \sum_{a' \neq a_n} \underbrace{\mu_w(f_\psi, \mathbf{x}^{(n)}, a^{(n)}, a')}_{\text{MWF } (\mu) \text{ Loss}}\end{aligned} \tag{5.2}$$

$$\mathscr{F}_2(\phi, \psi, \mathbf{x}^{(n)}, a^{(n)}, \mathbf{y}^{(n)}) = -\sum_{n=1}^{N} \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}^{(n)}|\mathbf{x}^{(n)}, a^{(n)})}\left(\ell(f_\psi(\mathbf{z}^{(n)}), y^{(n)})\right)}_{\text{Predictor } (f) \text{ Loss}}. \tag{5.3}$$

When optimising $\mathscr{F}_1$, we minimise w.r.t. ϕ and θ (encoder q_ϕ and decoder p_θ parameters respectively), and when optimising $\mathscr{F}_2$, we minimise w.r.t. ψ (predictor f_ψ parameters). The reason we alternate the training in this way is to ensure that the learnt representation $\mathbf{z}$ is counterfactually fair. For example, if we were to minimise w.r.t ψ when optimising $\mathscr{F}_1$, and neglected the inclusion of $\mathscr{F}_2$, the learnt parameters of the predictor f_ψ would be encouraged to produce counterfactually fair predictions. This would discourage the learnt representation $\mathbf{z}$ from being counterfactually fair, since the predictor could simply learn to make counterfactually fair predictions to begin with.

We also provide the sensitive attribute information in the form of A to the encoder and decoder, as in the conditional variational autoencoder [29, 49]. This allows the retrieval of any information pertaining to the protected attribute that was lost during the regularisation of the representation Z by the μ_w term.

When obtaining the latent variables for use in prediction by f, we strictly use the means of the latent representation, and choose not to sample them according to their mean and variance. Various configurations (sampling during training and testing, sampling only during training,

etc.) were attempted and the minimal variance by using the mean of the latent representation was found to produce both fairer and more accurate predictions.

Finally, α and β are tunable parameters we can set to enforce the amount of predictor and MWF loss on the learnt representation respectively. We usually set α high enough to ensure high-predictive power in the learnt representation, and investigate the effects of varying β later. For now we simply set β to ensure counterfactual fairness in the learnt latent representation.

5.3 Results

5.3.1 Counterfactual Scenario

We begin by testing counterfactuals. This is the most basic case, as there is a single world (i.e., $m = 1$ in Eq 5.2).

We generate data using a SCM with 40 observed variables (and therefore 40 latent variables) having the following graph and structural equations:

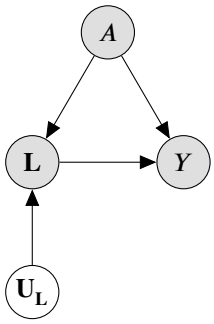


Fig. 5.2 Dummy Generative CGM

$$Y|\mathbf{L}, A \sim \mathcal{N}\left(\sum_{i=1}^{N} l_i + \zeta_{ay} a, 0.25\right)$$

$$\mathbf{L}|A, \mathbf{U} = \boldsymbol{\zeta}_{al} a + \mathbf{u}$$

$$\mathbf{U} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0.10 & \dots & 0.05 & 0.05 \\ \vdots & \ddots & 0.05 & 0.05 \\ 0.05 & 0.05 & 0.10 & 0.05 \\ 0.05 & 0.05 & 0.05 & 0.10 \end{bmatrix}\right)$$

We set the weight $\zeta_{ay} = 0.2$, and the values of ζ_{al} are randomly sampled from a uniform distribution such that $\zeta_{al} \sim \text{unif}(0, 0.1)$. We instantiate these weights once per experiment.

Implementation Notes

- All neural networks are implemented in PyTorch 0.4.0 [43].
- We one hot encode the world (factual/counterfactual) in a 1D vector $\mathbf{W}$.
- We train on 60,000 datapoints, and test on 4,000.
- For the MWF-AAE, a latent space of size 5 was chosen, hence we 'squeeze' the latent space from 40 to 5.
- For the MWF loss term we select small ϵ ($\epsilon = 0.01$) to ensure the relative counterfactual invariance of the representation.
- The MWF-AAE encoder, decoder, and predictor networks feature a single fully connected hidden layer with 100 units and ReLU activation functions.
- The MWF-AAE loss function is parameterised with $\alpha = 1$ and $\beta = 10$
- All other predictors are 2 hidden layer neural networks with 100 units each and ReLU activation functions.
- All training is done using Adam [28] with default settings and 25 epochs.
- 'Perfectly Fair' regresses directly on the 40 latent variables themselves.
- The 'NN on Z' model is a separate neural network trained purely on the representations Z learnt by the encoder.
- 'FairLearning' infers all counterfactual observations using counterfactual reasoning, as documented in [32].
- 'FairLearning + MWF-AAE' uses the counterfactual observations inferred by the 'FairLearning' algorithm instead of the true counterfactuals.

Results

	CFU	RMSE
Constant	0.000	9.213
Unfair	0.999	0.514
FTU	0.923	0.521
Perfectly Fair	0.000	1.341
FairLearning	0.005	1.340
MWF-AAE	0.008	1.331
NN on Z	0.008	1.330
FairLearning + MWF-AAE	0.004	1.333

Table 5.1 CFU and RMSE of various predictors on the dummy data

We observe that our MWF-AAE performs very well, and outperforms the neural network trained directly on the true latents at the expense of slight unfairness.

Furthermore, we train an additional neural network on the representation learnt by the MWF-AAE ('NN on Z'); this is to prove that the learnt latent representation is indeed counterfactually fair, and not simply fair with respect to the jointly learnt predictor in the MWF-AAE framework. The results from this show that our latent representation is indeed fair, and shows the validity of the adversarial approach we take to jointly learning the predictor $\hat{Y}$ and the latent representation Z, which makes these two independent from each other.

Finally we observe that in a potential production scenario, whereby we require the generation of counterfactuals using an assumed graph ('FairLearning + MWF-AAE'), our algorithm still performs well, and outperforms a neural network trained directly on the inferred counterfactuals ('FairLearning'). This is likely due to overfit and collinearity issues on the higher dimensional data, whereas the MWF-AAE mitigates this by learning a lower dimensional representation.

Tuning the Unfairness and Accuracy Trade-Off

We can also control how fair we wish the latent representation to be. There are two approaches to how we could achieve this:

- Varying the parameter β (i.e., strength) of the MWF term, leaving ϵ small.

- Varying the bound on the MWF fairness term ϵ in Eq 5.1, leaving β large to ensure that this bound is satisfied.

We investigate both strategies, observing the trade-off we obtain between unfairness and accuracy.

First we vary β, keeping $\epsilon = 0.01$:

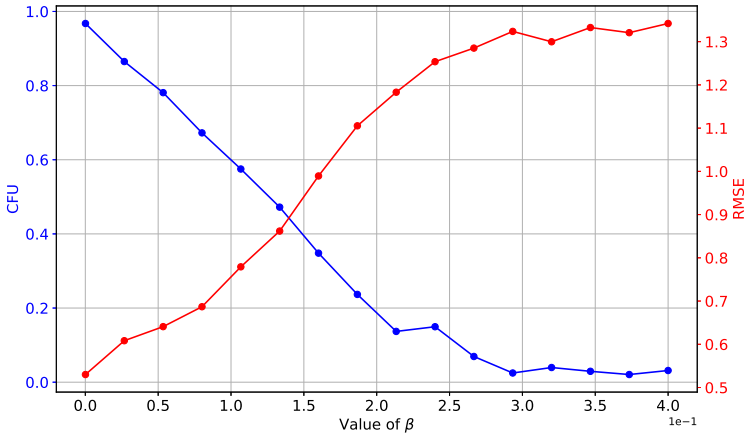


Fig. 5.3 CFU and RMSE with increasing β

Next we vary ϵ, keeping $\beta = 20$:

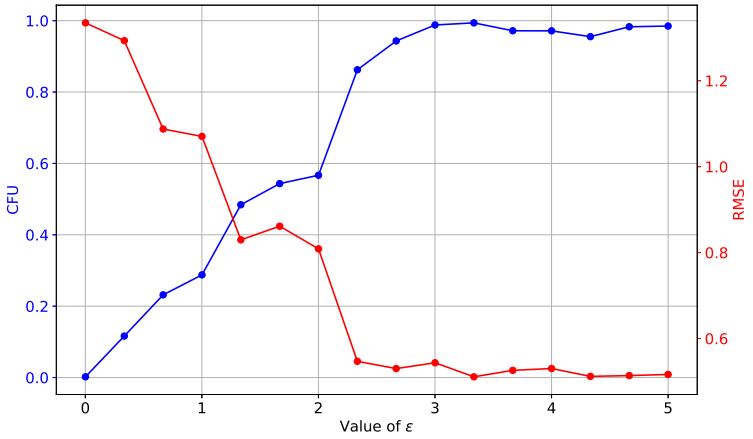


Fig. 5.4 CFU and RMSE with increasing ϵ

Both approaches appear to be valid ways of tuning the amount of fairness/accuracy, and have a similar linear scaling with respect to parameter magnitude. In both instances, we notice the trade-off between accuracy and unfairness, with highly unfair representations allowing for

the lowest RMSE in the resultant predictions. Following [47], we will tune the unfairness by varying ϵ.

5.3.2 Multi-World Scenario

In multi-world scenarios, we require counterfactual quantities for each of the hypothetical worlds; we therefore require factual data as well as multi-world counterfactuals for that data. We synthesise this data using the following generative graph and structural equations:

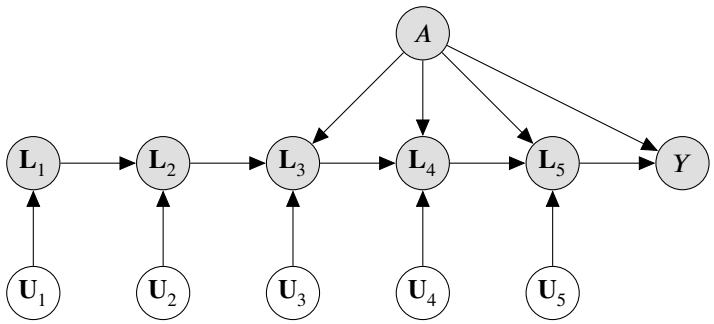


Fig. 5.5 CGM used to generate data for MWF Experiments

$$
\begin{aligned}
Y|\mathbf{L}_5, A &\sim \mathcal{N}\left(\sum_{i=1}^{N} l_{5i} + \zeta_{ay}a, 0.25\right) \\
\mathbf{L}_1|\mathbf{U}_1 &= \mathbf{u}_1 \\
\mathbf{L}_2|\mathbf{L}_1, \mathbf{U}_2 &= \boldsymbol{\zeta}_{12}^{\intercal}\mathbf{l}_1 + \mathbf{u}_2 \\
\mathbf{L}_3|A, \mathbf{L}_2, \mathbf{U}_3 &= \boldsymbol{\zeta}_{a3}a + \boldsymbol{\zeta}_{23}^{\intercal}\mathbf{l}_2 + \mathbf{u}_3 \\
\mathbf{L}_4|A, \mathbf{L}_3, \mathbf{U}_4 &= \boldsymbol{\zeta}_{a4}a + \boldsymbol{\zeta}_{34}^{\intercal}\mathbf{l}_3 + \mathbf{u}_4 \\
\mathbf{L}_5|A, \mathbf{L}_4, \mathbf{U}_5 &= \boldsymbol{\zeta}_{a5}a + \boldsymbol{\zeta}_{45}^{\intercal}\mathbf{l}_4 + \mathbf{u}_5 \\
\mathbf{U} &\sim \mathcal{N}\left[\begin{pmatrix}0\\ \vdots \\ 0\\ 0\end{pmatrix}, \begin{pmatrix}0.10 & \dots & 0.05 & 0.05\\ \vdots & \ddots & 0.05 & 0.05\\ 0.05 & 0.05 & 0.10 & 0.05\\ 0.05 & 0.05 & 0.05 & 0.10\end{pmatrix}\right]
\end{aligned}
$$

We select the dimensions of each $\mathbf{L}_n$ to be 10, which means that each $\boldsymbol{\zeta}_{nm}$ is a 10×10 matrix, and each $\boldsymbol{\zeta}_{an}$ is a 10-D vector. We again sample each value in $\boldsymbol{\zeta}$ uniformly such that $\zeta_{nm} \sim \text{unif}(0, 0.1)$

We propose the two following incorrect world models, from which we will calculate the counterfactuals:

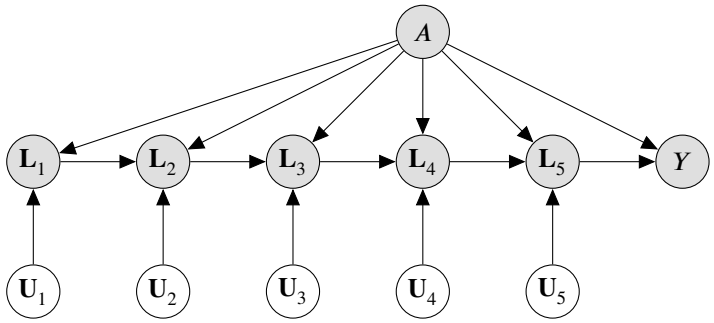


Fig. 5.6 Incorrect CGM World 1 (Additional Edges)

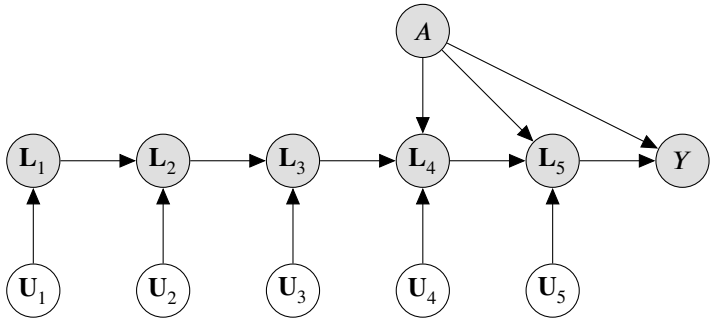


Fig. 5.7 Incorrect CGM World 2 (Missing Edges)

Since this causal model is much more complex, we first verify that our MWF-AAE can in fact learn the counterfactually fair representation given the **true** counterfactually fair data (using the same implementation as in Section 5.3.1):

	CFU	**RMSE**
Constant	0.000	160.614
Unfair	1.002	0.622
Perfectly Fair	0.000	2.755
FairLearning	0.004	2.861
MWF-AAE	0.003	2.767

Table 5.2 CFU and RMSE of various predictors on the MWF dummy data

We observe that the MWF-AAE performs nearly as well as a neural network learnt directly on the true latent variables ('Perfectly Fair'), and better than a neural network which uses

the inferred latent variables from an Level 3 model ('FairLearning'). Therefore, despite the additional complexity of the causal model, we are still able to learn a counterfactually fair representation.

We now generate the counterfactual data according to each incorrect world model, which we perform using the standard '**Abduction**', '**Action**', and '**Prediction**' steps covered in Section 2.3.1. We save the latent variables learnt in the 'Abduction' step to understand how much unfairness each incorrect world introduces into the final calculations by training a neural network on each of these incorrectly inferred latent variables:

	CFU	**RMSE**
True Latents	0.000	2.755
Correct Model	0.004	2.861
Incorrect Model World 1	0.164	2.442
Incorrect Model World 2	0.659	1.431

Table 5.3 CFU and RMSE of predictors trained on different Worlds

We observe that both incorrect models introduce unfairness into the subsequent predictions made on their abducted latent variables, which is not surprising given the results from Chapter 4.

We are now learn a representation that simultaneously satisfies fairness under both these worlds whilst reducing overall loss. In order to understand how much unfairness we exhibit in each separate 'World', we modify the CFU criterion slightly as follows:

$$\text{CFU}_{wm} = \frac{\sum_{i=1}^{N} |\hat{Y}(z^{(i)}_{a^{(i)}}) - \hat{Y}(z^{(i)}_{wa'^{(i)}})|}{\sum_{i=1}^{N} |y^{(i)}_{a^{(i)}} - y^{(i)}_{wa'^{(i)}}|}. \tag{5.4}$$

Therefore we evaluate the numerator over the counterfactual latent variable obtained in proposed world 'w', and measure the difference between this and the factual latent variable. We also divide through by the sum of differences between the factual and world-specific counterfactual outcomes, since we must assume that each hypothesised world generates the correct counterfactual (since we usually do not have access to the 'true' world).

We now train the MWF-AAE to be fair across both World 1 and World 2, setting $\beta = 20$ and $\epsilon = 0.01$:

Model	**CFU Actual**	**CFU World 1**	**CFU World2**	**RMSE**
MWF-NN	0.162	0.003	0.001	4.595
MWF-AAE	0.159	0.009	0.011	4.026

Table 5.4 CFUs and RMSE of models trained on multiple worlds

We achieve our aim of learning a multi-world fair representation of the data, since the CFU evaluated over Worlds 1 and 2 are both near 0. However, this comes at the expense of predictive power, with an RMSE that is higher than that of the true model, or either World models. We compare our approach to that in [47] (and Section 2.3.3: Multi-World Fairness), denoted by **MWF-NN**, and observe that our model in fact outperforms the original approach, albeit at the expense of reduced fairness across the hypothetical worlds. It is worth noting that our model uses 5 latent variables for prediction, whereas the original implementation predicts on 52 variables, and does not learn a fair representation.

We do not achieve fairness on the actual counterfactuals, but this is due to errors in the CGMs we used to calculate the counterfactual observations, and not an issue of the MWF-AAE framework. We also observe that we in fact achieve better fairness than either of the original World models in Table 5.3. Interestingly, calculating CFU with respect to the actual counterfactual data, we find that certain ϵ settings (i.e., $\epsilon \approx 1$) result in suprisingly fair decisions in the 'actual' world:

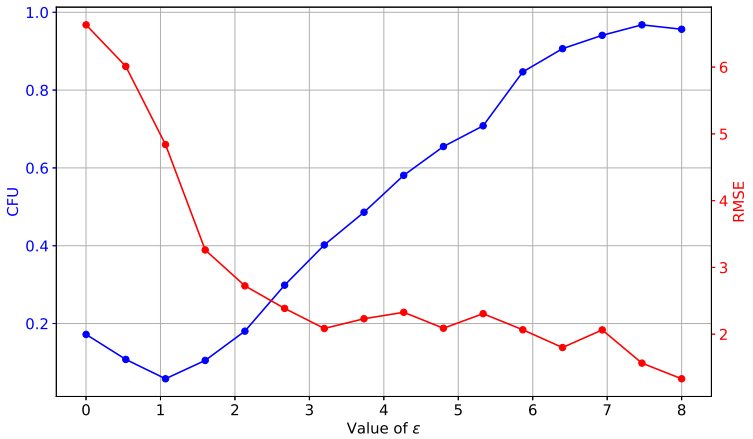
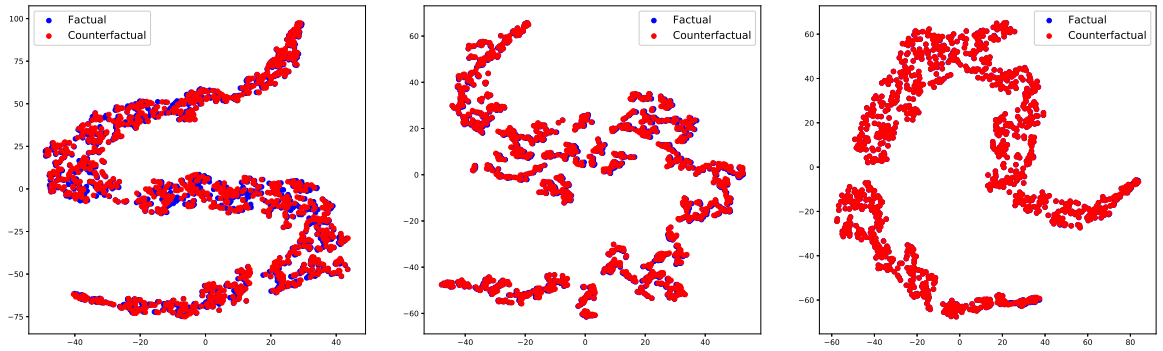


Fig. 5.8 CFU in the actual World and RMSE with increasing ϵ

This presents a potential avenue for future work; understanding how the combination of causal graphs under the multi-world criteria interacts with the 'actual' world fairness, which is easily motivated using the CFU measure.

5.4 Latent Space Visualisations

We present latent space visualisations to illustrate the impact of the MWF term on the learnt representations. We learn a 10-D latent space, which we project into 2-D using t-SNE [51], plotting both factuals and counterfactuals:



(a) Unfair MWF-AAE (CFU = 1.000)

(b) Multi-World Fair MWF-AAE (CFU = 0.160)

(c) Fair MWF-AAE (CFU = 0.001)

Fig. 5.9 Latent Space Visualisations for various MWF-AAEs

We observe that when no MWF term is applied (Fig 5.9a), there is a visible shift between the factuals and counterfactuals in the latent space. Conversely, when we apply the full MWF term during training (Fig 5.9c), the factuals and counterfactuals almost fully overlap, showing there is little difference between them, validated by the near-zero CFU. Finally, training the model with the multi-world fair data (Fig 5.9b) provides less overlap than the fully unfair model, but doesn't match the fair model.

Chapter 6

Conclusion

6.1 Discussion

In this thesis, we presented a review of fairness, introducing, comparing, and clarifying salient definitions and measures. We also introduced a new metric, termed CFU, which allows for the evaluation of counterfactual fairness. We then used the CFU measure to develop a novel approach to the analysis of counterfactual fairness, taking inspiration from sensitivity analysis in the causality literature. We also compared various observational definitions of fairness under the CFU metric, and evaluated their counterfactual performance. Finally, we presented a novel and flexible variational approach to jointly learning a counterfactually fair predictor and representation, which can be extended to multiple worlds; we showed this allows for the successful learning of either a counterfactually or multi-world fair low-dimensional latent representation.

6.2 Future Work

We outline potential avenues of further research for the two main areas investigated in the thesis (Chapters 4 and 5 respectively).

6.2.1 Novel Counterfactual Fairness Analysis

The experimental results clearly show the importance of using an accurate causal graph, and it is possible that there exist analytical expressions relating the CFU metric to graph errors. It would therefore be fruitful to discover relationships, as it is possible these may reveal more about the interaction between model misspecification and counterfactual fairness.

There exists scope to determine the impact of non-linearities on the graph structure. For example, we could choose to model the functional relationship between variables using non-linear functions, such as Bayesian Additive Regression Trees [21]. We could investigate how more powerful algorithms respond when the underlying relationships are linear (i.e., will they overfit), and similarly when we assume linear relations between variables when the generative model is in fact non-linear.

It is possible to extend the CFU metric to incorporate one-sided unfairness; for example, we may tolerate unfairness in one direction, as it would benefit an historically marginalised group (i.e., the difference principle [46]), and defining with it a new type of counterfactual fairness criterion.

Using the CFU it is possible to understand how multi-world approaches to fairness interact with the 'actual' world. Such analyses may help with the design of more robust models in the future.

Introducing aspects of the causal discovery literature [40] may be helpful, and understanding if these methods can help to create more robust models under the CFU metric, as well as their pitfalls.

6.2.2 Variational Counterfactual Fairness

It is important to fully test the application of the MWF-AAE to modelling real-world data, thus proving the value of the counterfactually fair representations that are learnt.

It may also be possible to simply not include the variational fairness term, and learn instead a counterfactually fair representation represented by the output of a hidden layer in a neural network [3], similar to [24]. This may make the modelling more robust, as we only use the learnt representation for predictive, not generative, purposes.

Having learnt a single fair latent representation across multiple worlds, it would be possible to gauge the efficacy of this approach to transfer learning scenarios, as in [39].

References

[1] Angwin, J., Larson, J., Mattu, S., and Kirchner, L. (2016). Machine bias. [Online; posted 23-May-2018].

[2] Barocas, S. and Selbst, A. D. (2014). Big Data's Disparate Impact. *SSRN eLibrary*.

[3] Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828.

[4] Berk, R., Heidari, H., Jabbari, S., Kearns, M., and Roth, A. (2017). Fairness in Criminal Justice Risk Assessments: The State of the Art. *ArXiv e-prints*.

[5] Beutel, A., Chen, J., Zhao, Z., and Chi, E. H. (2017). Data Decisions and Theoretical Implications when Adversarially Learning Fair Representations. *ArXiv e-prints*.

[6] Bilal Zafar, M., Valera, I., Gomez Rodriguez, M., Gummadi, K. P., and Weller, A. (2017). From Parity to Preference-based Notions of Fairness in Classification. *ArXiv e-prints*.

[7] Breiman, L. (2001). Random forests. *Mach. Learn.*, 45(1):5–32.

[8] Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., and Riddell, A. (2017). Stan: A probabilistic programming language. *Journal of Statistical Software, Articles*, 76(1):1–32.

[9] Chiappa, S. and Gillam, T. P. S. (2018). Path-Specific Counterfactual Fairness. *ArXiv e-prints*.

[10] Datta, A., Tschantz, M. C., and Datta, A. (2015). Automated experiments on ad privacy settings. *Proceedings on Privacy Enhancing Technologies*, 2015(1):92 – 112.

[11] Dorie, V., Harada, M., Carnegie, N. B., and Hill, J. (2016). A flexible, interpretable framework for assessing sensitivity to unmeasured confounding. *Statistics in Medicine*, 35(20):3453–3470.

[12] Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. (2011). Fairness Through Awareness. *ArXiv e-prints*.

[13] Edwards, H. and Storkey, A. (2015). Censoring Representations with an Adversary. *ArXiv e-prints*.

[14] Ensign, D., Friedler, S. A., Neville, S., Scheidegger, C. E., and Venkatasubramanian, S. (2017). Runaway feedback loops in predictive policing. *CoRR*, abs/1706.09847.

[15] Executive Office of the President (2016). Big data: A report on algorithmic systems, opportunity, and civil rights. Technical report, Washington: White House.

[16] Gates, S. W., Perry, V. G., and Zorn, P. M. (2002). Automated underwriting in mortgage lending: Good news for the underserved? *Housing Policy Debate*, 13(2):369–391.

[17] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative Adversarial Networks. *ArXiv e-prints*.

[18] Grgić-Hlača, N., Zafar, M. B., Gummadi, K. P., and Weller, A. (2016). The Case for Process Fairness in Learning: Feature Selection for Fair Decision Making. In *Symposium on Machine Learning and the Law at the 29th Conference on Neural Information Processing Systems (NIPS 2016)*.

[19] Hacker, P. and Wiedemann, E. (2017). A continuous framework for fairness. *ArXiv e-prints*.

[20] Hardt, M., Price, E., and Srebro, N. (2016). Equality of opportunity in supervised learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 3323–3331, USA. Curran Associates Inc.

[21] Hill, J. L. (2011). Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240.

[22] Jadad, A. R. and Enkin, M. W. (2007). *Randomized Controlled Trials: Questions, Answers, and Musings*. Blackwell Publishing, Oxford, Oxfordshire, UK, 2nd edition.

[23] Jimenez Rezende, D., Mohamed, S., and Wierstra, D. (2014). Stochastic Backpropagation and Approximate Inference in Deep Generative Models. *ArXiv e-prints*.

[24] Johansson, F. D., Shalit, U., and Sontag, D. (2016). Learning representations for counterfactual inference. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pages 3020–3029. JMLR.org.

[25] Kearns, M., Neel, S., Roth, A., and Wu, Z. S. (2018). Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In Dy, J. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2569–2577, Stockholmsm$\diamondsuit$ssan, Stockholm Sweden. PMLR.

[26] Kilbertus, N., Rojas Carulla, M., Parascandolo, G., Hardt, M., Janzing, D., and Schölkopf, B. (2017). Avoiding discrimination through causal reasoning. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 656–666. Curran Associates, Inc.

[27] Kim, M. P., Ghorbani, A., and Zou, J. (2018). Multiaccuracy: Black-Box Post-Processing for Fairness in Classification. *ArXiv e-prints*.

[28] Kingma, D. P. and Ba, J. (2014). Adam: A Method for Stochastic Optimization. *ArXiv e-prints*.

[29] Kingma, D. P., Rezende, D. J., Mohamed, S., and Welling, M. (2014). Semi-Supervised Learning with Deep Generative Models. *ArXiv e-prints*.

[30] Kingma, D. P. and Welling, M. (2013). Auto-Encoding Variational Bayes. *ArXiv e-prints*.

[31] Kleinberg, J., Mullainathan, S., and Raghavan, M. (2016). Inherent Trade-Offs in the Fair Determination of Risk Scores. *ArXiv e-prints*.

[32] Kusner, M. J., Loftus, J., Russell, C., and Silva, R. (2017). Counterfactual fairness. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 4066–4076. Curran Associates, Inc.

[33] Kusner, M. J., Russell, C., Loftus, J. R., and Silva, R. (2018). Causal Interventions for Fairness. *ArXiv e-prints*.

[34] Loftus, J. R., Russell, C., Kusner, M. J., and Silva, R. (2018). Causal Reasoning for Algorithmic Fairness. *ArXiv e-prints*.

[35] Louizos, C., Shalit, U., Mooij, J. M., Sontag, D., Zemel, R., and Welling, M. (2017). Causal effect inference with deep latent-variable models. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 6446–6456. Curran Associates, Inc.

[36] Louizos, C., Swersky, K., Li, Y., Welling, M., and Zemel, R. (2015). The Variational Fair Autoencoder. *ArXiv e-prints*.

[37] Lowry, S. and Macpherson, G. (1988). A blot on the profession. *British Medical Journal*, 296(6623):657–658.

[38] Lum, K. and Isaac, W. (2016). To predict and serve? *Significance*, 13(5):14–19.

[39] Madras, D., Creager, E., Pitassi, T., and Zemel, R. (2018). Learning Adversarially Fair and Transferable Representations. *ArXiv e-prints*.

[40] Malinsky, D. and Danks, D. (2018). Causal discovery algorithms: A practical guide. *Philosophy Compass*, 13(1).

[41] Nabi, R. and Shpitser, I. (2017). Fair Inference On Outcomes. *ArXiv e-prints*.

[42] O'Dwyer, R. (2018). Are you creditworthy? the algorithm will decide. [Online; posted 7-May-2018].

[43] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). Automatic differentiation in pytorch. In *NIPS-W*.

[44] Pearl, J. (2009). *Causality: Models, Reasoning and Inference*. Cambridge University Press, New York, NY, USA, 2nd edition.

[45] Pearl, J., Glymour, M., and Jewell, N. P. (2016). *Causal Inference in Statistics: A Primer*. John Wiley & Sons Ltd, Chichester, West Sussex, UK, 1st edition.

[46] Rawls, J. (1971). *A Theory of Justice*. Harvard University Press.

[47] Russell, C., Kusner, M. J., Loftus, J., and Silva, R. (2017). When worlds collide: Integrating different counterfactual assumptions in fairness. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 6414–6423. Curran Associates, Inc.

[48] Shalit, U., Johansson, F. D., and Sontag, D. (2017). Estimating individual treatment effect: generalization bounds and algorithms. In Precup, D. and Teh, Y. W., editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3076–3085.

[49] Sohn, K., Lee, H., and Yan, X. (2015). Learning structured output representation using deep conditional generative models. In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R., editors, *Advances in Neural Information Processing Systems 28*, pages 3483–3491. Curran Associates, Inc.

[50] Steel, E. and Angwin, J. (2010). On the web's cutting edge, anonymity in name only. [Online; posted 4-August-2010].

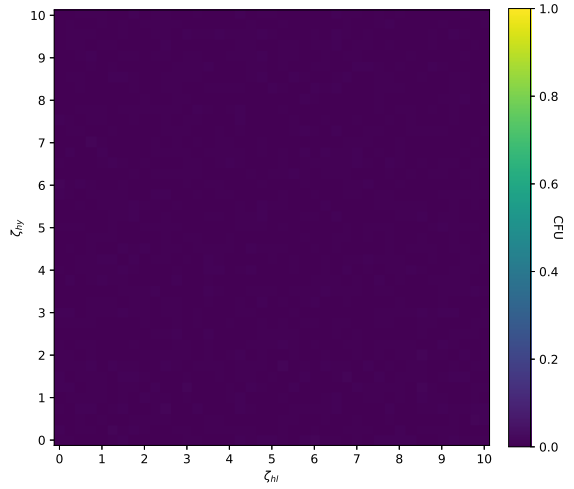
[51] van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605.

[52] Woodworth, B., Gunasekar, S., Ohannessian, M. I., and Srebro, N. (2017). Learning Non-Discriminatory Predictors. *ArXiv e-prints*.

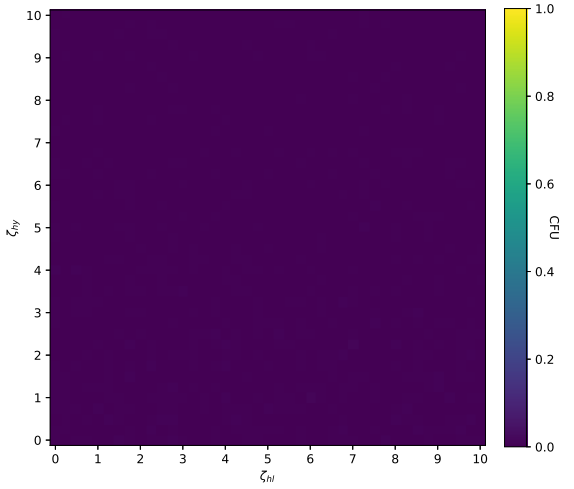
[53] Zhang, B. H., Lemoine, B., and Mitchell, M. (2018). Mitigating Unwanted Biases with Adversarial Learning. *ArXiv e-prints*.

Appendix A

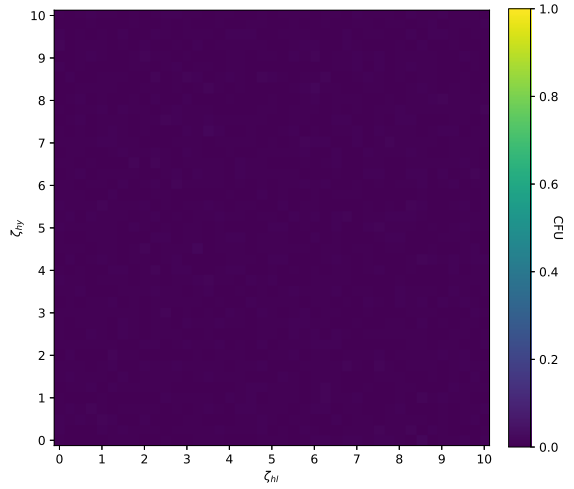
Hidden Confounder with Correct Model



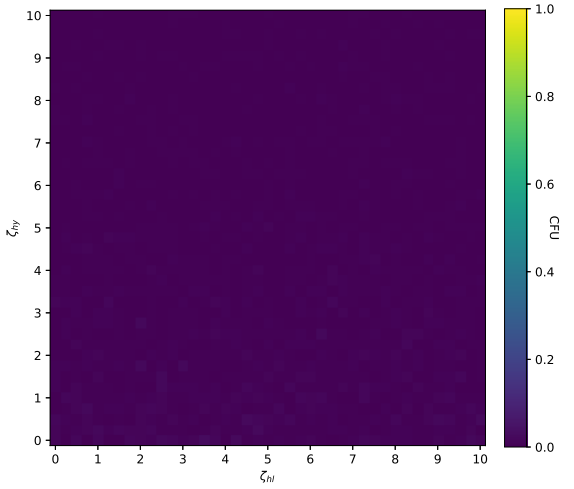
(a) $\zeta_{al_1} = 2$, $\zeta_{al_2} = 2$, $\zeta_{al_3} = 2$, $\zeta_{al_4} = 2$, $\zeta_{ly} = 2$, $\zeta_{ay} = 2$



(b) $\zeta_{al_1} = 2$, $\zeta_{al_2} = 2$, $\zeta_{al_3} = 2$, $\zeta_{al_4} = 2$, $\zeta_{ly} = 0.5$, $\zeta_{ay} = 1$



(c) $\zeta_{al_1} = 2$, $\zeta_{al_2} = 2$, $\zeta_{al_3} = 2$, $\zeta_{al_4} = 2$, $\zeta_{ly} = 10$, $\zeta_{ay} = 2$



(d) $\zeta_{al_1} = 0.5$, $\zeta_{al_2} = 0.5$, $\zeta_{al_3} = 0.5$, $\zeta_{al_4} = 0.5$, $\zeta_{ly} = 0.5$, $\zeta_{ay} = 1$

Fig. A.1 CFU against Confounding Weights (Correct Model)