Suicide Under Investigation:

Causal Inference, Deconfounding, Effects

Seamus Conlon

MSc Responsible Artificial Intelligence

Supervisor: Dr. Alexandros Koliousis
Northeastern University London
2023

Contents

1 Abstract

This project uses a dataset compiled of suicide statistics by country from 1985-2015 along with corresponding sociological data from the World Bank that could be predictive of suicide rates, given various theories of the causes of suicide. To try and improve upon the results that conventional regressors have in predicting suicide rates, this project builds a regressor from the causal inference package DoWhy, inspired by Judea Pearl's do-calculus. The function of the DoWhy package is to estimate the causal effect of a variable in a dataset by controlling for both observed and simulated unobserved confounders. This dissertation built two regressor models from DoWhy: one that assumes causal independence between the variables in the dataset but controls for simulated unobserved confounders; and one that controls for causal relationships between observed variables and uses a genetic algorithm to learn the most accurate graph representation of confounding relationships between observed variables. Results show that model controlling for unobserved confounders improved upon the performance of standard regressors from the Scikit-learn library. However, the model controlling for observed confounders was slightly inferior to standard regressors, whilst also being computationally costly.

2 Motivation

Identifying predictors of suicide risk has been a vexed issue in the fields of psychology and sociology. Whilst an individual's history (or lack thereof) of suicide attempts has been shown to robustly predict suicide risk, and cross-culturally men to tend to be at higher risk of suicide than women, psychologists, sociologists and mental health professionals have nevertheless found it difficult to identify reliable predictors and causes of suicide.

In addition, the dependence of much contemporary machine learning upon learning statistical associations means that it can be prone to recognising spurious correlations rather than causal relationships. Consequently, much machine learning, spanning both deep learning as well as non-connectionist data science techniques, is prone to predicting on the basis of statistical associations in the training data. This problem is inevitably connected to the issue of generalisability. A model that can determine which patterns and associations in its training data are causally connected and which are merely correlated will be less likely to predict on the basis of spurious correlation that may not exist in future unseen data.

3 Literature Review

3.1 Causal Inference

For those unfamiliar with causal inference, some it may be helpful to define some terminology that will be frequently used going forward. A **confounder** is a common cause of x and y that makes it difficult to measure how much causal effect (if any) x has on y. An independent variable x whose causal impact on y is being measured is referred to to as a **treatment**, whereas the dependent variable y is referred to as the **outcome**. The **Average Treatment Effect (ATE)** refers to the estimated causal effect that a treatment has upon an outcome.

Pearl's work on causal inference has been essential not just to the contemporary study of causality but also to the role of causal inference in machine learning, especially given Pearl's history with artificial intelligence and Bayesian networks. In both *Causality* (2000) and *The Book of Why* (2015), Pearl critiques contemporary machine learning for being dependent on unreconstructed, causality-free statistics. This undermines its ability to determine which statistical associations are due to confounding and which are causal. Pearl innovated and formalised the do-calculus for causal inference. The do-calculus is a calculus for how to calculate the causal effect of a given variable by controlling for possible confounding variables. Crucial to Pearl's work on causality is his invention of the 'do'-operator. The conventional notation for conditional probability $P(y|x)$ is agnostic as to causal relations. Pearl's use of the 'do' operator enables conditional probabilities to model controlled interventions. $P(y|\text{do}(x))$ denotes the probability of y given an intervention upon x, and intervention meaning that all potentially confounding variables have been controlled for. As such, $P(y|\text{do}(x))$ better models the probability of y given x when confounders are controlled for. It consequently comes closer to measuring causal relationships than the conventional symbols for conditional probability do.

The do-calculus contains three rules for how to arrive at causal estimates based on observed data - rather than by performing interventions and counterfactuals, which data science alone cannot do. The calculus uses a directed-acyclic-graph (DAG) to map causal relationships between variables, since the notation of the rules of the do-calculus specifies which causal pathways in a directed graph have to be blocked in order for variables to be discounted or for observations to be treated as equivalent to interventions.

The first rule specifies the conditions under which an observation can be ignored:

$$i)\ P(y|z, \text{do}(x), w) = P(y|\text{do}(x), w) \text{ if } (Y \perp Z|W, X)_{G_{\overline{X}}}$$

meaning that z can be ignored if y and z are conditionally independent, given w and x, once all causal pathways leading into x have been blocked.

The second rule specifies the conditions under which an intervention can be regarded as equivalent to an observation:

$$ii)\ P(y|\mathrm{do}(z),\mathrm{do}(x),w) = P(y|z,\mathrm{do}(x),w) \text{ if } (Y \perp Z|W,X)_{G_{\overline{X},\underline{Z}}}$$

meaning that z can be treated as equivalent to do(z) if y and z are conditionally independent, given w and x, once all causal pathways leading into x and all causal pathways leading out of z have been blocked.

The third rule specifies the conditions under which an intervention can be ignored:

$$iii)\ P(y|\mathrm{do}(z),\mathrm{do}(x),w) = P(y|\mathrm{do}(x),w) \text{ if } (Y \perp Z|W,X)_{G_{\overline{X},\overline{Z(W)}}}$$

meaning that do(z) can be ignored if y and z are conditionally independent, given w and x, once all causal pathways leading into x and any causal pathways leading out of z that are not ancestors of w have been blocked.

Using the aforementioned do-calculus rules for when observations can be treated as equivalent to interventions and when observations and interventions can be ignored, the below formula can be derived as the formula for how to block backdoor causal pathways and thereby control for confounders:

$$P(y|\mathrm{do}(x)) = \sum_z P(y|(x),z)\ \times\ P(z)$$

where z denotes other causes of y.

Seeking to apply Pearl's work on causal inference to data science and machine learning, Sharma and Kiciman (2020) created the DoWhy python library for causal inference. DoWhy requires both a dataset as well as DAG modelling causal pathways between features in the dataset. Accurate assumptions about the causal relationships (or lack thereof) between features in the dataset are expected from the user on the basis of their domain-specific knowledge, and cannot be learnt and provided by DoWhy. The package uses the terminology of causal inference and so refers to dependent variables as outcomes and independent variables as treatments. DoWhy has three stages of analysis: i) the identification of an estimand, meaning which effects (e.g. backdoor effects, frontdoor effects, instrumental variables etc.) can be controlled for on the basis of the DAG the model has been provided with; ii) an estimate of the causal effect of the treatment on the outcome, given the estimand; and iii) a refutation, whereby a sensitivity analysis is performed on the causal estimates by using simulated data and the causal estimates are refined in light of their robustness when simulated data is introduced.

For estimating the ATE of a treatment variable, DoWhy has a range of methods, including propensity scoring for discrete or binary treatment variables and modified linear regression (capable of deconfounding) for continuous treatment variables. The latter will be used in this project, which deals with largely continuous data. DoWhy has multiple methods for using simulated data for sensitivity analyses to test the robustness of causal estimates, including placebo treatments and dummy outcomes that should lead the causal estimate to collapse to zero. Of these methods, the one that is of greatest interest to this project is that which uses simulated data to add unobserved confounders. This is because the sociological predictors of suicide that will be used as features for this dissertation's dataset will hardly constitute an exhaustive list of the causes of suicide. Additionally, the vexed issue of unobserved confounders is one that plagues controlled trials and that randomised controlled trials (RCTs) seek to solve. Simulating unobserved confounders means avoiding the illusion of having controlled for all confounding variables and will come closer to approximating the rigor of an RCT. These refutation methods require the user to insert domain-specific knowledge that can help determine the nature of the simulated data.

There are other python packages for causal inference such as CausalML and EconML. However, Chen et al. (2020) make clear that CausalML should be used on data from RCTs and that CausalML should be used as a supplement since the library in itself cannot control for unobserved confounders. DoWhy has an advantage in this respect due to its ability to simulate unobserved confounders in its sensitivity analyses. Meanwhile, EconML (Battocchi et al., 2019) serves to measure heterogenous treatment effects and is therefore better suited for estimating a conditional average treatment effect (CATE) rather than estimating an ATE, which is the purpose of both RCTs and DoWhy.

3.2 The Use of Machine Learning to Predict Suicide

To date, most uses of machine learning to predict suicide risk have been to predict the risk of individuals. Since this involves personal information about the individuals whose risk is being predicted, these applications have typically been in either a professional mental health context or at social media companies (since these contexts enable the collection of private and sensitive data). When used by social media companies, these models typically use neural networks for sentiment analysis of social media posts. Roy et al. (2020) classified individuals according to their suicide risk and took into account what an individual's predicted suicide risk would have been prior to their explicit expressions of suicidal ideation, with AUC accuracy rates being 80% for individuals prior to the explicit expression of suicidal ideation and 88% for those who have explicitly expressed suicidal ideation.

Not all uses of machine learning to predict suicide by social media activity have used neural networks. A survey review by Castillo-Sanchez et al. (2020) of the literature on using machine learning to identify suicide risk from social media posts found that support vector machines were used in a majority of cases and that random forests were used in a substantial minority. It should be noted that causal inference techniques were not among those listed by the survey as methods being used for identifying suicide risk.

Goddard, Xiang and Bryan (2022) provided a rare attempt to use causal inference in the use of machine learning to predict suicide rates. The paper sought to predict suicide risk on an individual basis, collecting the participants in their study from volunteers willing to complete surveys in the waiting rooms of US military prime care clinics. The model sought to use invariance-based causal prediction (ICP) as a technique for determining which predictors were likely to prove not to be spurious correlations and then used logistic regression to estimate their predictive power. The purpose of the study was not to make testable predictions, but to generate a theory of which predictors are causal. However, Goddard, Xiang and Bryan's paper acknowledges that it does not approximate the randomness necessary to control for unobserved confounders.

Not all applications of ML to predict suicide risk have used intimate psychological and personal data to predict on an individual-by-individual basis. Kumar et al. (2022) used data science techniques to predict suicide rates by county across the United States, using data from over the course of the 2010s. The study used an extreme gradient boosted (XGBoost) decision tree regressor. The dataset consisted largely of demographic and sociological variables. An examination of the variables used as significant predictors demonstrates that the study would likely be prone to misgeneralisation and spurious correlation if extrapolated to international and cross-cultural data. The model found the proportion of a county's population that is white to be a positive predictor of suicide rate and the proportion of a county's population that is black to be a negative predictor of suicide rate. Needless to say, these variables will have weak predictive power in many non-American societies due to i) societies having populations which are largely neither black nor white; and ii) the statistical associations found between race and suicide risk may be spurious correlations that may evaporate when confounders are removed.

3.3 Theories of Suicide

There is a long history of theorising the causes of suicide, with the first prominent and canonical example being Durkheim's (1897) explanation of why Protestants were more likely to commit suicide than Catholics in late-nineteenth century Europe. Durkheim posited that it was 'anomie', a social atomisation resulting from subcultures less bound by family ties, that led to higher suicide rates. As the study of suicide has progressed, the fact that suicidal ideation does not reliably predict suicidal action has led to investigations into what predicts suicidal action in particular. One such theory is the interpersonal theory of suicide (IPTS), first outlined by Joiner (2005) and developed and expanded by Orden at al. (2010). The purpose of the IPTS is to identify the mechanism by which factors that predict suicide interact with each other to result in suicidal action (since the factors that result in suicidal ideation alone are not sufficient cause for suicidal action). The latter define the fundamental three necessary causes of suicide risk to be: i) perceived burdensomeness; ii) thwarted belongingness; and iii) suicide capability.

'Perceived burdensomeness' refers to an individual's perception that their existence is a material and financial burden for their family members. 'Thwarted belongingness' refers to the frustration of an individual's need for social integration and community. Thwarted belongingness differs from simpler definition of social isolation in that it defines loneliness as an unsatisfied need for or expectation of social interactions. The unavailability and/or unwillingness of people to spend time with an individual in person are cited as factors that could thwart the individual's belongingness. 'Suicide capability' denotes an individual's readiness to subject themselves to the bodily pain and harm that come with most methods of suicide. The IPTS assumes that we are evolutionarily wired to avoid bodily harm and that therefore environmental factors must lead to an individual becoming more willingness to expose themselves to bodily harm and pain. In their paper outlining the IPTS, Order et al. speculate that this could explain a causal link between a history of work in the military or police and suicide, since those careers could desensitise individuals to bodily harm.

Chu et al. (2017) claim that whilst there is consensus about how thwarted belongingness and perceived burdensomeness interact to causally increase an individual's risk of suicidal ideation, it is ambiguous as to whether or not suicide capability is environmentally acquired.

Joiner et al. (2017) developed the eusocial theory of suicide (ETS), a variation upon the IPTS. This theory assumes that i) human beings can have innate predispositions towards self-sacrifice; and that ii) this instinct, when combined with perceived burdensomeness on one's family members, can lead to suicide. According to this theory, the well-observed link between a history of military service and suicide is not directly causal, but is due to the confounding variable of a predisposition towards self-sacrificial behaviour, which can both make an individual inclined towards a career in the police or the military as well as incline an individual towards suicide due to perceived burdensomeness on their family members.

4 Methodology

4.1 Compiling The Dataset

This project's investigation into the predictors of suicide concern sociological predictors rather than individually-grained personal and psychological predictors. As has been noted in the literature review, there have been many applications of machine learning to identifying suicide risk in a psychiatric context and on an individual-by-individual basis. But these often took place in a clinical context where mental health professionals had access to private and confidential data regarding individuals' risk. Needless to say, this data is not publicly available. Not only that, but sociological data that covers a wide, even international scope, would more successfully test the generalisability of machine learning models. The predictive power that certain variables have to predict suicide may well be culturally contingent and may not replicate across different cultures and societies. For this reason, international and sociological data was chosen for this project's dataset.

The primary source for this project's dataset is a dataset from kaggle.com that is itself an compilation of multiple datasets. The Kaggle dataset, titled 'Suicide Rates Overview 1985-2016', breaks down suicide rates by sex and age group in every year from 1985 to 2016 for 101 different countries. Reliable data on suicide rates cannot be found for all countries, so the dataset does not constitute an exhaustive overview of suicide rates by country. The dataset also contains information on the corresponding GDP-per-capita for every country in the dataset for every relevant year. It is from this dataset that the crucial dependent variable for this project is sourced - the suicide rates per one hundred thousand people for given demographics. Suicide rates for any given year and country are aggregated by sex and age group.

The dataset was compiled and edited with the Python pandas library. For each country for any given year in the dataset, there are 12 rows/datapoints, one for each combination of sex and age groups (for which the options are 0-15 years, 15-25 years, 25-35 years, 35-55 years, 55-75 years, and 75+). LabelBinarizer() from SciKit Learn was used to transform the sex column from 'male'/'female' into binary digits. Using one-hot encoding to create a column (and therefore consequently a feature) for each age group was an option. However, since this project is building a regressor, it would be potentially unhelpful to have a dataset in which a substantial minority of the features are discrete variables. Additionally, creating 6 different features for each age group would potentially overstate the significance of age. For this reason, the age group column was turned into a continuous variable in which the numerical value for any given age-group would be the lower bound of that age-group (for instance, 75.0 for '75+ years').

To expand the number of predictors, datasets from data.worldbank.org regarding the following indicators were selected: unemployment rate and poverty rate (both of which may intuitively be expected to predict suicide for the same reasons that GDP-per-capita may); population density and urbanisation (since theories of suicide from Durkheim to the

Interpersonal Theory of Suicide have considered social atomisation and alienation to be significant causes of suicide); the percentage of the population over 65 and the fertility rate (since both the Eusocial Theory of Suicide and the Interpersonal Theory of Suicide both consider perceived-burdensomeness upon younger family members to be a relevant cause of the elevated rates of elderly suicide in some societies); the percentage of the population with access to the internet (as the Interpersonal Theory of Suicide suggests that a sense of thwarted belongingness can be caused by an inability to socialise offline); and the percentage of the labour force in the military (since military history is widely acknowledged to be a predictor of suicide risk, and has been theorised about by Joiner). Once all datasets were compiled together, the pandas dropna() method was used to eliminate from the dataframe any rows in which information was not contained for all variables. The result is the decrease from an initial dataset of approximately twenty thousand rows to one of just under ten thousand rows. However, this still provided a dataset of 9636 rows x 11 columns - more than large enough to be provide substantial training and test sets.

The first 8000 rows were used for the training set and the remaining 1636 for the test set (roughly proportional to a 80%-20%) split. Splitting the dataset chronologically (i.e. the training data being from 1985-2010 and the test data being from 2010-2016) was one method of splitting the data. However, splitting the dataset by country, so that no countries in the training set feature in the test set, is a better way to test the generalisability of the model. Predictors that may strongly predict suicide rates in one cultural context may have weak predictive power in other cultural contexts, due to confounding variables that may vary between countries. For this reason, splitting the dataset along geographical lines will better test how well the model can control for confounders (whether those confounders be unobserved or in the dataset) and generalise.

Whilst country and year are dropped from the dataset, these datapoints are independently preserved in a separate dataset so that they can ultimately be concatenated with the model's corresponding predictions. This will enable an analysis of how the model's predictive accuracy varies according to time, country, age group and sex.

4.2 Model Design

Two different regressors will be built from the DoWhy causal inference package. One assumes causal independence between all the variables in the compiled dataset and uses gradient descent to learn the most likely strength of unobserved confounders in the dataset. The other uses a genetic algorithm to learn the graph representation of causal relationships between treatment variables that has the most predictive accuracy when the DoWhy uses it to estimate ATEs. The first model will henceforth be referred to as the unobserved-confounders-learner whilst the latter will be referred to as the graph-learner. There is no reason in principle why both algorithms could not be combined, but due to the computationally expensive nature of genetic algorithms for combinatorial optimisation, this was not attempted in this project.

Both models use Mean Squared Error (MSE) as their loss function. The causal inference stage of the model (see Algorithm 1) learns its estimated ATEs from the first 80% of the training dataset. However, it is the performance of the model in predicting the entirety of the training dataset that is used to calculate the MSE. The reason for this is that the purpose of the sensitivity analysis stage of the model (in which unobserved confounders are simulated to test the robustness of the ATEs) is to avoid overfitting and maximise the generalisability of the model. The robustness of the model can be better measured by its performance on unseen data than on its accuracy on the data that it learned its ATEs from. Consequently, if the accuracy loss of the model was measured on the same data that it learns its ATEs, then this would undermine the purpose of simulating unobserved confounders to test the robustness of the ATEs. This is particularly important for the unobserved-confounders-learner, since the model's generalisability (or lack thereof) will determine how it needs to update its estimate of what the likely causal strength of unobserved confounders will be.

Additionally, there is a hyper-parameter for k-folding the training dataset to produce multiple different models and then select the model that proves to have the greatest ability to generalise. This differs from k-folds cross validation in two ways: i) the different models are trained on $\frac{k-1}{k}$ of the training data but are tested not on the unseen $\frac{1}{k}$ of the training data, but on the entirety of the training data; and ii) there is no averaging or integration of the models, but simply a winner-takes-all selection of the model which performs best on the entirety of the training data. This selects a model that performs well on unseen data but also prevents the risk of selecting a model whose estimates are overfitted to the unseen data.

4.2.1 Causal Estimator

Both the graph-learner and unobserved-confounders-learner use causal inference package to estimate the causal effects of the variables in an algorithm that is illustrated in pseudocode (see Algorithm 1). This algorithm iterates through each variable in the dataset (variables being referred to as treatments in the terminology of causal inference) and i) estimates the causal effect of the variable by controlling for the variables that have been specified as common causes of both the outcome variable and the treatment variable in question; and ii) tests and updates the estimated ATE by controlling for simulated confounders (of a specified strength) of both variable and target. For the later step, due to the element of randomness and inconsistency involved in the simulation of unobserved confounders, this step is repeated five times so that the mean updated ATE can then be used as the coefficient for the variable in question. The intercepts produced by the causal inference for each variable are also stored and then averaged to produce a single intercept.

Since it is in some sense a modified version of multivariate linear regression, the unobserved-confounders-learner will give the exact same intercept for every predictor that it estimates. However, in the version which uses a genetic algorithm to find an optimal DAG, the inter-

Algorithm 1 Causal Estimator

```
Take $E$ as estimate of unobserved confounders
for $treatment$ in $x$ do
    if DAG received from user then Instruct causal inference calculator to control for confounders of $treatment$ in the DAG
    else Instruct ausal inference calculator to treat all other treatments as confounders
    end if
    Instruct ausal inference calculator to estimate an ATE and intercept for $treatment$
    for $iteration = 1, 2, \dots 5$ do
        Instruct ausal inference calculator to simulate unobserved confounders that have a causal impact of 0.01 on $treatment$ and $E$ on $outcome$
        Update estimated ATE in light of its robustness to simulated confounders
    end for
    Store updated ATE and intercept for $treatment$
end for
Create an intercept from the mean of the stored intercepts
Treat each of the updated ATEs as a coefficient for its corresponding $treatment$
```

cept given by the causal estimator may vary. For this reason, the mean intercept is used as the single intercept used to make predictions.

4.2.2 Unobserved Confounders Learner

Algorithm 2 Unobserved Confounders Learner

```
Take $E$ estimate of unobserved confounders
for $iteration = 1, 2, \dots N$ do
    Use Causal Estimator to learn coefficients from first four fifths of training data
    Use $E$ to test robustness of coefficients
    Test updated coefficients on hole of training data
    Use derivative from MSE loss function to update $E$
end for
```

In this version of the model, during any given iteration of Causal Estimator, it will treat all other variables apart from the currently measured variable as confounders. Figure 1 illustrates how the estimator's model of confounding relationships would alter with each iteration. The global graph that results from combining all of these graphs together is a not a DAG, since for any two variables a and b, the causal inference package will have conditioned upon a when estimating the ATE of b and will have conditioned upon b when estimating the ATE of a. By contrast, a model that assumes a to be a cause of b would need to merely condition upon a when estimating the ATE of b. As such, the resulting global graph for the unobserved-confounders-learner assumes causal independence between variables. In this respect, it is no different from multivariate linear regression. The strength of the unobserved-confounders-learner is that it learns the appropriate strength of the unobserved confounders.

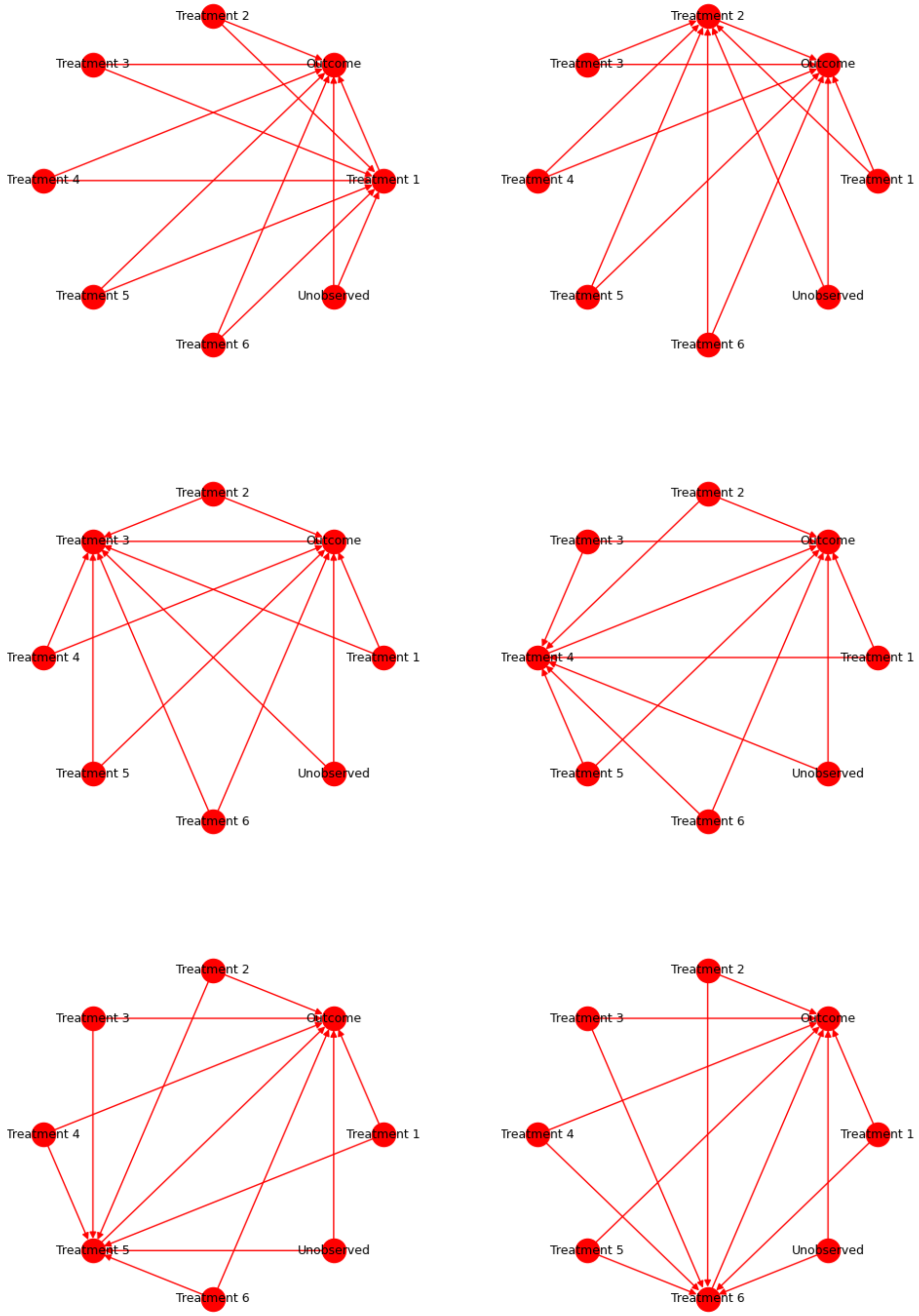


Figure 1: The changing confounding relationships durinig iteratioins of the OCL

The value strength of the simulated unobserved confounders upon the treatment variables is fixed at 0.01 for all epochs. But the model uses gradient descent to learn the optimal estimate for strength of unobserved confounders on the outcome variable. For each epoch, once the coefficients and intercept generated by Causal Estimator have been tested on the training data, a derivative from the resulting MSE is then used (along with a user-user-specified learning rate) to determine what the strength of simulated confounders on the outcome variable will be for the following epoch. Since the role played by unobserved confounders is similar to that played by an intercept in linear regression (i.e. it accounts for causes of y that cannot be reduced to x), the model uses a derivative to learn the optimal value for the strength of unobserved confounders on the outcome variable in the same way that linear regression learns an intercept:

$$\theta_0 := \theta_0 - lr\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})$$

Two decisions have been made here: i) the decision to learn the optimal value for only one of the hyper parameters concerning the strength of unobserved confounders; and ii) the decision to learn the optimal value for effect strength on outcome, as opposed to effect strength on treatment.

Decision i) was made because if both hyper-parameter values are learnt and updated by gradient descent, then the proportional relationship between the two values will be identical, regardless of what the initial values prior to learning are. Leaving one value fixed whilst learning the other also allows for the other value to move in either a positive or negative direction. If both hyper-parameter values increase or decrease in parallel with each other, then this precludes the possibility of the mode learning that unobserved confounders could have a positive causal impact on the treatment and negative causal impact on outcome (or vice versa).

Decision ii) was motivated partly by the dataset in question. There will no doubt be causes of suicide that are far more immediate and proximate than those used as predictors for this project's dataset. Not only that, but it is likely that there are more immediate causes of suicide which have little or negligible causal impact on the predictors featured in this project's dataset. For this reason, it is wiser to fix the estimated effect of unobserved confounders on treatments at a low number (0.01) and to allow the estimated effect of unobserved confounders on the outcome to become as large a (positive or negative) number as the gradient descent learning process leads it to become.

4.2.3 Graph Learner

The graph learner searches for the graph representation of confounding relationships that, when given to Causal Estimator to learn calculate the ATE for each variable, has the greatest accuracy. Since there are a discrete number of possible graph representations featuring

Algorithm 3 Genetic Algorithm For Graph Learner

```
Take $E$ estimate of unobserved confounders
for $iteration = 1$ do
    Generate random DAG of causal relationships between variables
    Use Causal Estimator to learn coefficients in light of DAG
    Use $E$ to test robustness of coefficients
    Test updated coefficients on hole of training data and store DAG along with its resulting coefficients and MSE in $History$
end for
for $iteration = 2, 3, \ldots N$ do
    Select the two DAGs in $History$ corresponding to the two lowest MSEs as $mutation_1$ and $mutation_2$
    for $iteration = 1, 2, \ldots N$ do
        if $iteration = even$ then $ancestor = mutation_1$
        else $ancestor = mutation_2$
        end if
        $mutation_i = ancestor$ with one edge added or removed
        Check $mutation_i$ is DAG and not already stored in $History$
        Use Causal Estimator to learn coefficients in light of DAG
        Use $E$ to test robustness of coefficients
        Test updated coefficients on hole of training data and store $mutation_i$ in $History$ along with its corresponding coefficients and MSE
    end for
end for
Select the mutation in $History$ with the lowest MSE as the optimal candidate
```

n number of variables, finding the optimal graph representation will be a combinatorial optimisation task. For this reason, a genetic algorithm is used to learn the optimal graph representation of causal relationships between observed variables.

For each epoch, a range of randomly mutated DAGs modelling possible causal relationships between variables in the dataset are created. Each mutation is given to the causal estimator algorithm and then stored in the model's mutation history along with the resulting coefficients, intercept and the MSE that they scored on the training data. At the end of each epoch, the two mutations in the mutation history that have the lowest MSEs are selected as the ancestors from which the mutations of the following epoch will be generated.

During the first epoch of the algorithm, each candidate is a randomly generated DAG, without precursor. This means that there is unlikely to be much resemblance between the candidate graph representations of the first epoch. For all subsequent epochs, all candidates will be mutations of one of either of the two past mutations selected to serve as ancestors. For each mutation, one edge will be added or removed from its ancestor. The number of mutations per epoch will be twice the number of variables in the dataset.

The number of possible DAGs with n nodes is a superexponential of n:

$$a_n = \sum_{k=1}^{n} (-1)^{k+1} \binom{n}{k} 2^{k(n-k)} a_{n-k}$$

The sample space of possible DAGs with 11 nodes is 31,603,459,396,418,917,607,425. Whereas the number of mutations that will be tested by the graph learner that trains for 10 epochs on a dataset with 11 treatment variables (as is the case with this project's dataset) is 220, meaning that the percentage of the sample space that will have been explored by the genetic algorithm will be 6.9612632e-19.

It is certainly suboptimal for the accuracy of the the graph-learner on datasets with many variables that the number of tested mutations grows linearly with the number of variables whilst the number of possible mutations grows superexponentially with the number of number variables. However, the computational cost of the graph learner will grow exponentially with the number of variables even if the number of mutations per epoch only grows linearly. This is because the larger the size of the DAG that Causal Estimator learns from, the more confounding relationships that the Causal Estimator will have to control for. For this reason, the number of mutations per epoch will only grow linearly with the number of variables in the dataset.

4.3 Implementation

A class titled DoWhyRegressor was created which contains within it both the graph-learner and the unobserved-confounders-learner. The fit() method of the class has multiple hyperparameters that can be altered by the user. These consist of: a Boolean hyperparameter to specify whether the model will learn a graph representation or the strength of unobserved confounders; the number of epochs; the learning rate (for the unobserved-confounders-learner's gradient descent process); the strength of unobserved confounders (which is initialised at 0.01 for the unobserved-confounders-learner but can be fixed at any value for the graph-learner); and the sparsity/density of the initial randomly generated DAGs that the graph-learner's genetic algorithm selects from. The DoWhyRegressor has methods for predicting a given dataset, for scoring its accuracy on a given dataset and also for returning what the model has determined to be the most accurate graph representation of confounding relationships between variables (in the case of the graph-learner model). Additionally, it has attributes for its history of training errors, its estimate of the strength of unobserved confounders, and the ATEs that it uses as coefficients for variables. For this purposes of this dissertation, a special attribute was also created to keep track of what the model's predictive accuracy on the test data would be for any given stage during its training.

NumPy was used for mathematical operations in both models, whilst for the graph-learner the Networkx and the Random libraries were used for graph representations and for the random mutation of graphs respectively. DoWhy's CausalModel class has the option for the user to specify the confounders of the treatment and the variable either with a graph representation or with a list of the confounders. The graph-learner uses the former option by converting the networkx graph representation of a given graph mutation into a gml file. The unobserved-confounders-learner uses the latter option by listing as confounders all variables apart from the variable currently being measured for its ATE.

5 Results

5.1 Baseline Models

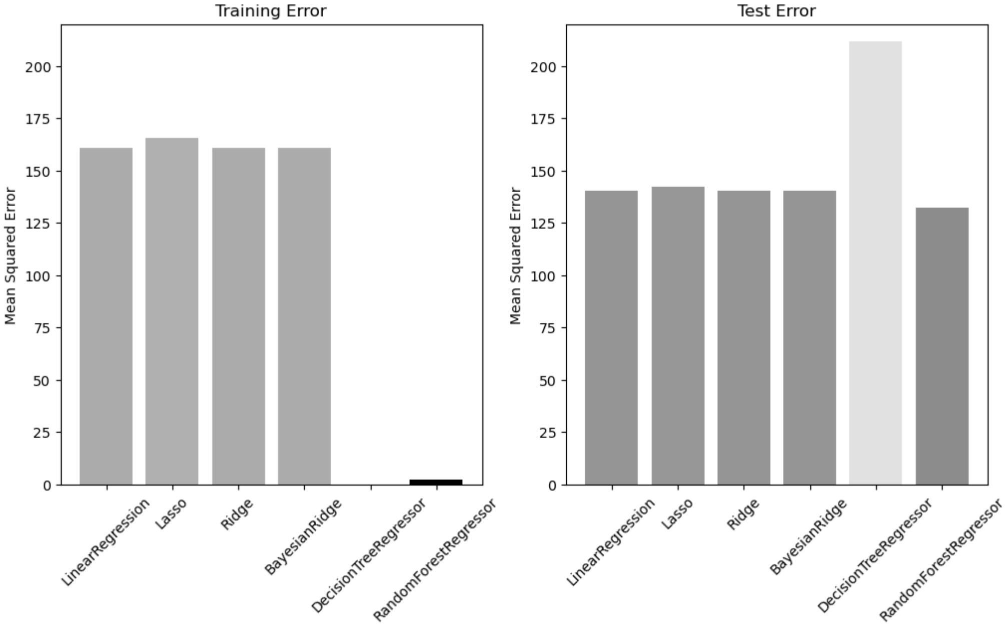
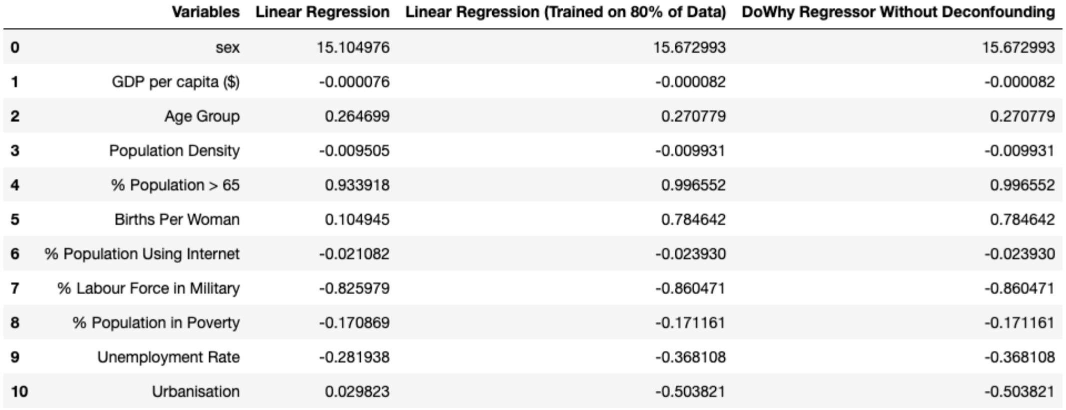


Figure 2: The performance of regressors from the Scikit-learn library on this project's dataset

Figure 2 visualises the performances of some standard regressors from the scikit-learn library on this project's dataset. The variants upon linear regression converge upon a mean squared error ≈ 160 on the training data and ≈ 140 on the test dataset. That these simple data science models perform better on the test dataset than the training dataset shows that being low variance and high bias could advantage a model on this dataset. By contrast, the DecisionTreeRegressor performs worst of all due to how it overfits the training data (on which it has zero error). The stochastic nature of the model means that its mean squared error on the test data can be anywhere between 170 and 230. However, the RandomForestRegressor does outperform the linear regression models on the test data, whilst still performing better on the training dataset than the test dataset. On the training dataset its mean squared error is close to zero and on the test dataset its mean squared error is 134 (but can range between 131 and 138, due to the stochastic nature of the regressor).

Figures 3 and 4 show that when the DoWhy regressor assumes causal independence between variables in the dataset but does not control for simulated unobserved confounders, its ATEs are identical to the coefficients learnt by a linear regression (when the linear regression learns its coefficients from the first 80% of the training data, as the DoWhy



	Variables	Linear Regression	Linear Regression (Trained on 80% of Data)	DoWhy Regressor Without Deconfounding
0	sex	15.104976	15.672993	15.672993
1	GDP per capita ($)	-0.000076	-0.000082	-0.000082
2	Age Group	0.264699	0.270779	0.270779
3	Population Density	-0.009505	-0.009931	-0.009931
4	% Population > 65	0.933918	0.996552	0.996552
5	Births Per Woman	0.104945	0.784642	0.784642
6	% Population Using Internet	-0.021082	-0.023930	-0.023930
7	% Labour Force in Military	-0.825979	-0.860471	-0.860471
8	% Population in Poverty	-0.170869	-0.171161	-0.171161
9	Unemployment Rate	-0.281938	-0.368108	-0.368108
10	Urbanisation	0.029823	-0.503821	-0.503821

Figure 3: Comparing linear regression's coefficients to the ATEs of an unobserved-confounders-learner that doesn't simulate unobserved confounders

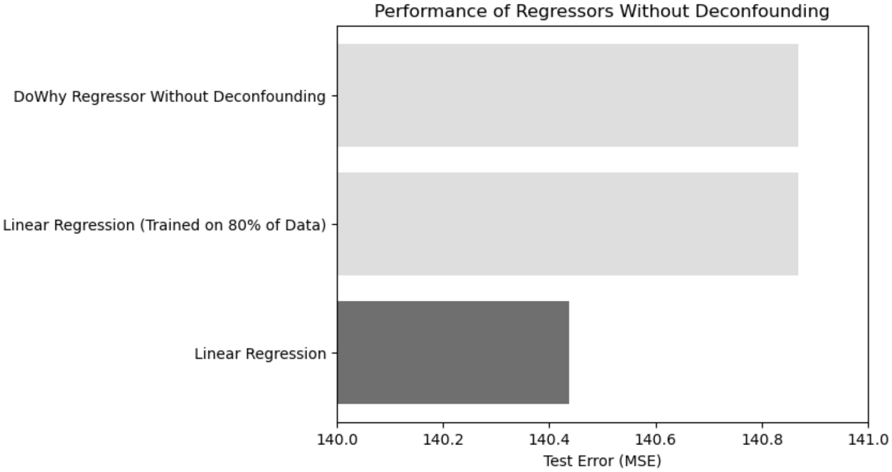


Figure 4: Comparing linear regression's performance to the performance of an unobserved-confounders-learner that doesn't simulate unobserved confounders

regressor does). The disastrous performances of polynomial regressions documented Figure

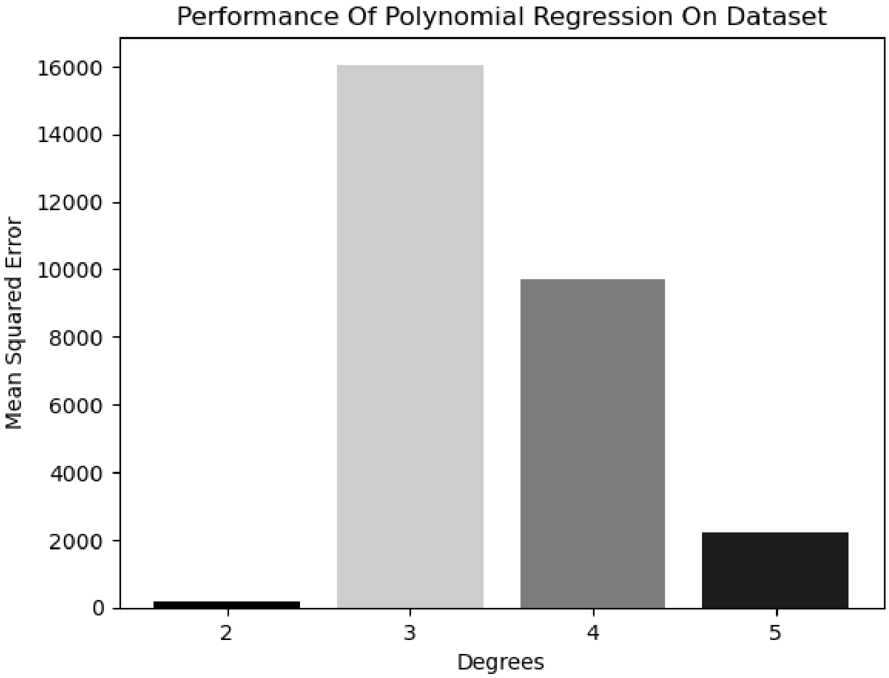


Figure 5: The performance of different polynomial regressions on this project's dataset

5 show that regressors that have high variance and low bias are unlikely to perform well on this dataset. Whereas the linear regressors and decision tree regressors generally have a mean squared error between 100 and 200, the mean squared errors for differently parameterized versions of polynomial regression are well into the thousands. This demonstrates that with this project's dataset, the risks of overfitting are high.

5.2 DoWhy Regressor

5.2.1 Learning Unobserved Confounders

The DoWhy regressor that learns the strength of unobserved confounders responds to changes in only two hyperparameters: epochs and learning rate. So that the trajectories of the regressor's training and test errors over time can be measured, the epochs hyperparameter is set to 20 for all fine-tuning experiments with this version of the regressor. The data visualisations of Figure 6 demonstrate how convergence is affected by different learning rates. Perhaps counter-intuitively, it is the model with the smallest learning rate, 0.01, that is most erratic in its convergence. However, as to be expected, Figure 7 demonstrates that the model with a learning rate of 0.01 is least sensitive to error when it comes to its estimate of the strength of unobserved confounders. The data visualisation in Figure 6 showing the trajectories of models' test errors shows that with a higher learning rate of 0.2, a model will almost immediately begin converging upon a test error superior to that of most of the linear regression variants in the scikit-learn library. With a learning rate of 0.1, convergence begins around 4 epochs and with a learning rate of 0.05 convergence begins around 10 epochs. Of the tested learning rates, only with 0.01 did the model have difficulty converging.

Whether it is advantageous to have more than 10 epochs is unclear, since the models tend not to improve once they converge around training and test errors of 165 and 137 respectively. Fortunately, there appears to be no risk of overfitting with extended training. The models' test errors are not convex functions of the time spent training and a decrease in training error quite reliably corresponds to a decrease in test error.

With the value for the causal impact of unobserved confounders on treatments set to 0.01, DoWhyRegressor models with differing learning rates appear to converge upon an estimate between 1.9 and 2.0 for the causal impact of unobserved confounders on the treatment. For this reason, the unobserved confounders hyperparameter will be set to 1.9 for experiments fine-tuning the hyperparameters of the version of DoWhyRegressor that uses a genetic algorithm to learn causal graphs.

5.2.2 Learning A Graph Representation Of Observed Confounders

Because of the considerable computational expense of the graph learner model, it would be unfeasible to spend 10 to 20 epochs learning the optimal strength of unobserved confounders for each mutation. Nevertheless, it cannot be assumed that the estimated strength of unobserved confounders (1.9 to 2.0) converged upon by the other model will also be optimal for the graph learner model. For this reason, three different values will be tested as the fixed estimate of unobserved confounders for three models, with other hyperparameters being controlled for. The results in Figure 8 show that of the three values tested, it is once again 0.2 that led to the greatest accuracy - an MSE of 144, which is a performance roughly equivalent to that of the Lasso regressor on this dataset. Curiously, model accuracy did not increase linearly with an increase in the estimated strength of unobserved confounders. One potential explanation is that 0.01 and 0.2 represent local minima for loss. Another is that this is simply a consequence of the stochastic nature of the genetic algorithm - especially the on built for this project, which only explores only a diminutive share of the sample space of candidate graphs. This makes convergence upon a global minimum less likely and increases the centrality of randomness to the model.

Additionally, a visual comparison between the trial errors and test errors of the three trialled models shows that how a model performs on training dataset does not reliably predict how it performs on the test dataset. Whilst the model with a value of 0.1 for the strength of unobserved confounders performed substantially less well than the other two, the models with unobserved confounder strengths of 0.01 and 0.2 converged upon very similar training errors. However, this did not translate into a convergence upon similar test errors. That two mutations can have similar performance on the training data but diverging performance on

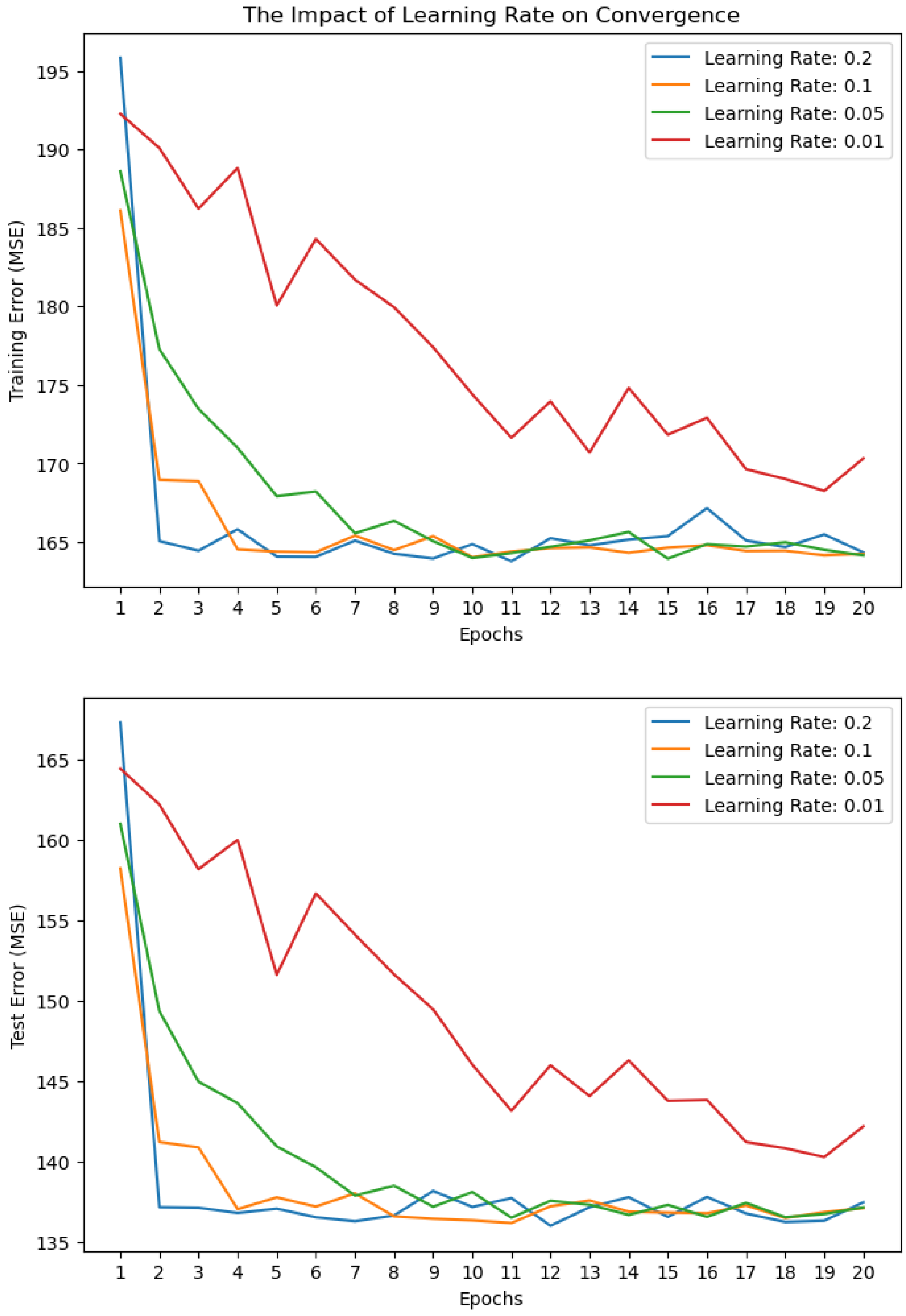


Figure 6: Impact of learning rate on training and test errors

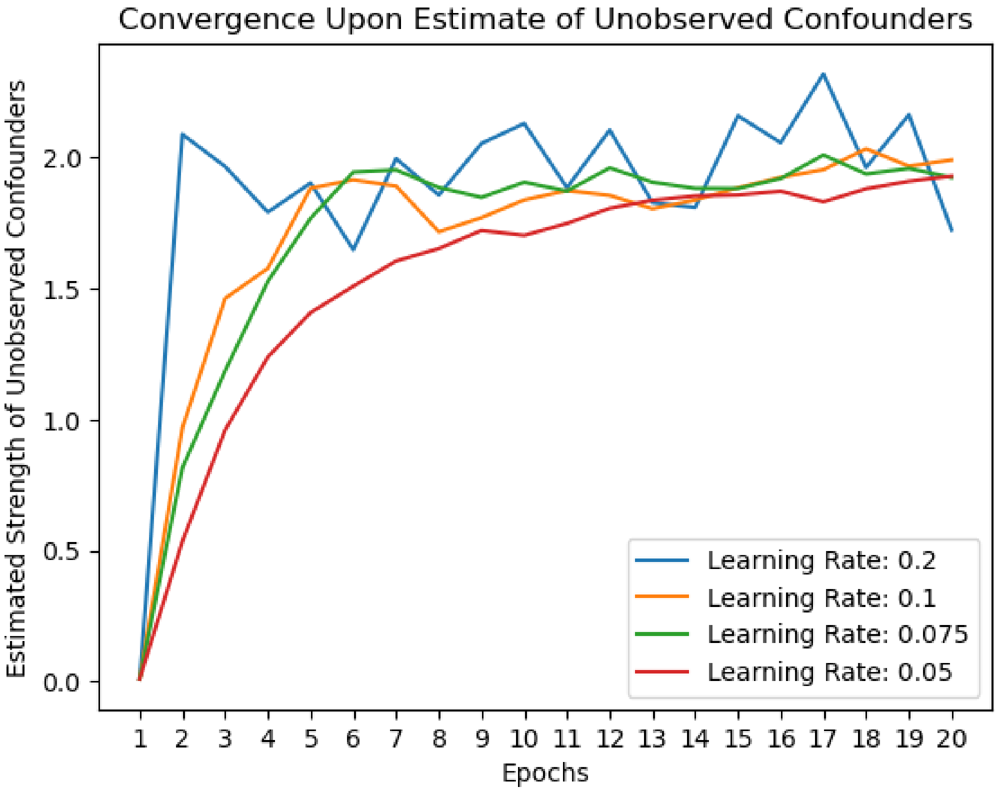


Figure 7: How UC learners with different learning rates converge upon similar estimates of the strength of UCs

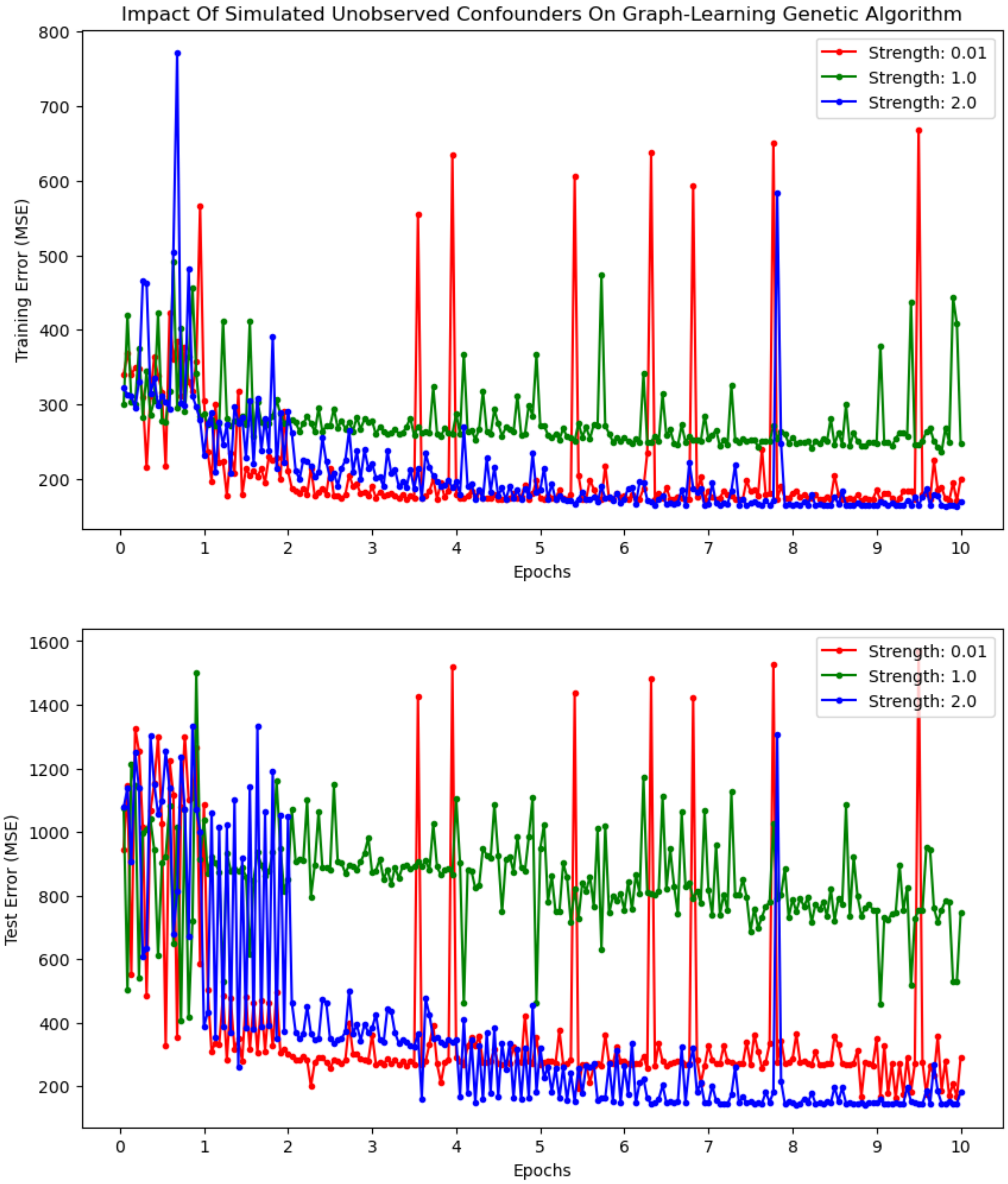


Figure 8: Impact of the strength of simulated nobserved onfounders on graph learner performance

the test data suggests that the graph learner model may be liable to converging upon local minima that are suboptimal when predicting the dataset. It also suggests that different causal graph representations can have equivalent predictive power on the dataset (but not necessarily on the test dataset). This casts doubt on the graph learner's ability to converge upon a global minimum.

With 2.0 selected as the fixed value for the strength of simulated unobserved confounders in future graph learn models, it was time to experiment with the other hyperparameter of interest. The level of confounding hyperparameter specifies how dense or sparse the initial candidate DAGs in the first epoch will be and takes arguments between 0 and 1. In the previously mentioned experiments, 0.15 was fixed as the value for this hyperparameter. Now, three other candidates have been tested - 0.1, 0.3 and 0.5. Figure 9 shows all three models performing worse than the previously tested model that had a value of 0.15 for the hyperparameter in question. The first candidate performed catastrophically, whilst the latter two achieved test errors of 185 and 228. This are test errors roughly comparable to those achieved by the decision tree regressor. Previous remarks about differing local minima that have diverging accuracies on the test data also prove relevant here. The two less inaccurate models that achieved test errors of 185 and 228 settled upon training errors of 167 and 168 respectively. Once the final graph learner experiment is added to the four graph learners that achieved an MSE between 160 and 170 on the training dataset, it can be seen in Figure 10 that for graph learners scoring in this region in the training dataset that their training error will have almost no ability to predict their test error.

In light of the results from experimenting with the density of initial DAG candidates and the strength of unobserved confounders, 0.15 and 0.2 will be selected as the optimal values for those respective hyperparameters. With these same hyperparameters once again used, the model achieved a mean squared error of 142.6 on the test dataset. This is slightly more accurate than the previous version model with the same hyperparameters scored (due to the stochastic nature of the model), but also slightly less accurate than linear regression. Figure 11 shows the graph representation that the algorithm found to have the most predictive accuracy when given to the DoWhy causal estimator. Some of the causal relationships implied by the graph are obviously nonsensical (for instance, the percentage of a population over 65 cannot *cause* sex and sex cannot *cause* urbanisation). However, the directed edges in the graph can be read as indications of what other variables to condition for when estimating a given variable's causal effect (i.e. the percentage of the population over 65 is a variable that should be conditioned upon when estimating the causal effect of sex on suicide). Age group has no edges directed towards it whilst sex and the percentage of population over 65 both have only one incoming edge but 5 and 6 outgoing edges respectively. That these variables are the ones that are the most causally effective and least causally effected, according to the graph, fits with established literature on the robustness of these demographic predictors of suicide. However, given the limited ability of the graph learner model to generalise, as well as the fact that this graph had slightly less predictive power than linear regression

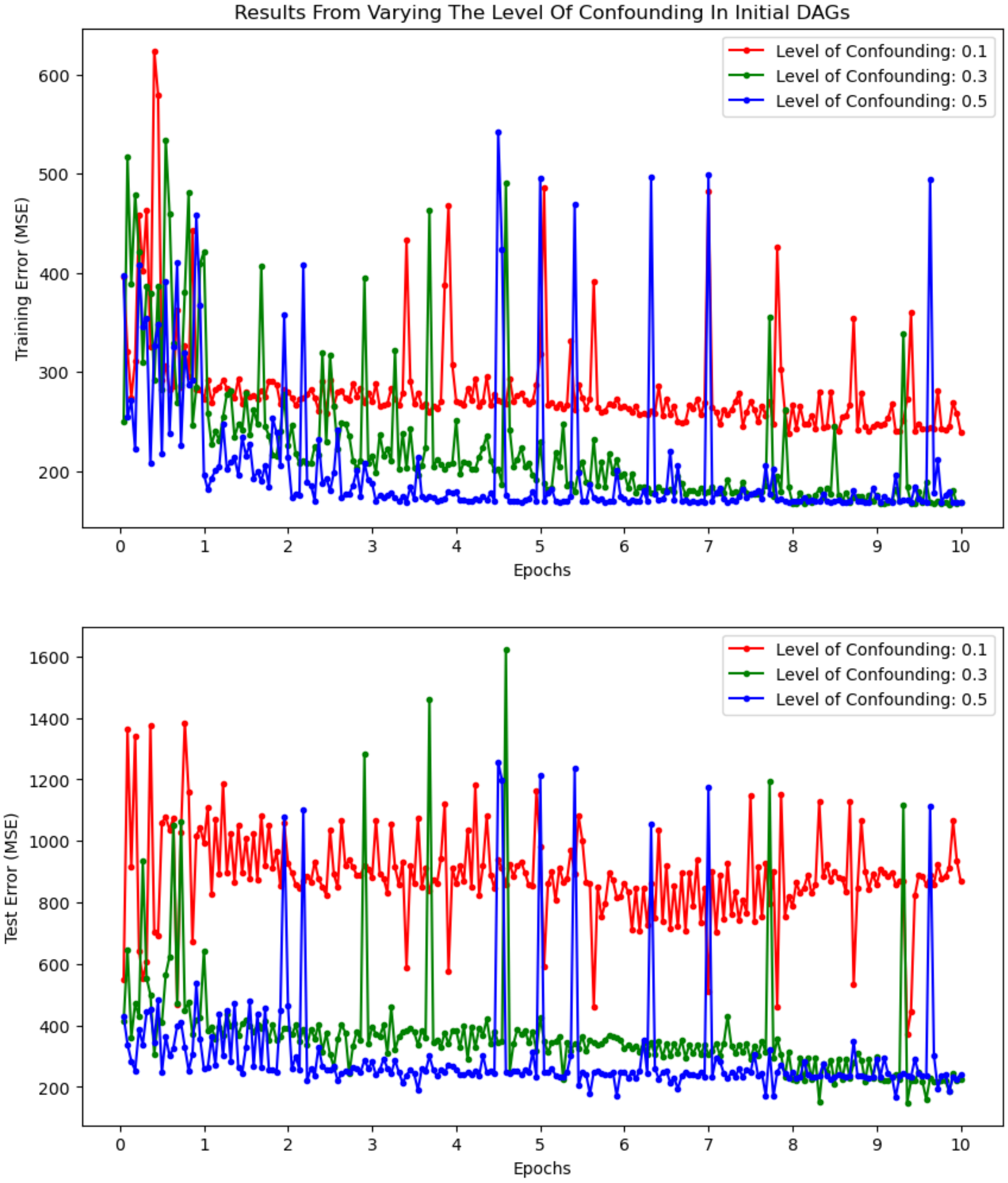


Figure 9: Impact Of Density Of Initial DAGs On Graph Learner Performance

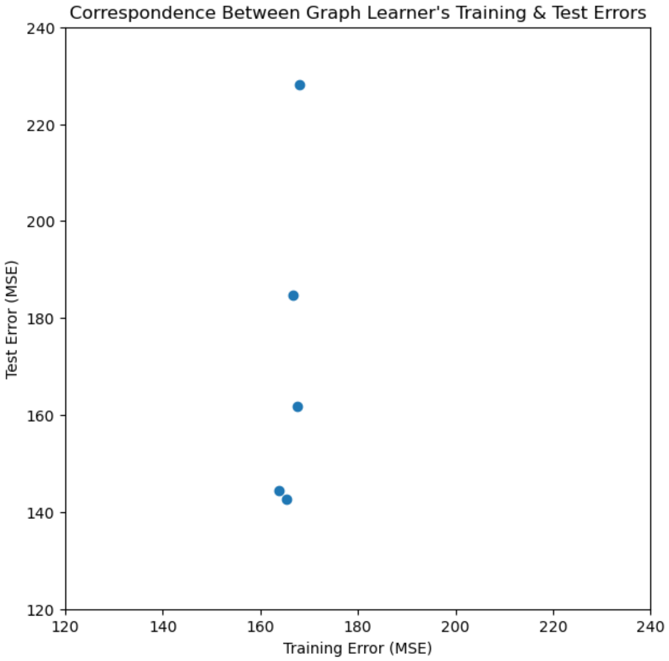


Figure 10: The correlation (or lack thereof) between training error and test error

does, it would be inappropriate to treat this graph as an authoritative description of causal relationships between the variables in this dataset.

The unobserved confounders learner is clearly superior to the graph learner at generalising. Although both the graph learner and the unobserved confounders learner tend to converge upon similar MSEs on the training dataset ≈ 165, the unobserved confounders learner reliably outperforms linear regression on the test dataset and often achieves a mean squared error only slightly higher than that of the random forest regressor. But the graph learner performs more similarly to the decision tree regressor, achieving a mean squared error from anywhere between 140 and 230 on the test dataset. It is also comes at vastly higher computational expense than the other models (Figure 12). For these reasons, the unobserved confounders learner will be the model used for subsequent experimentation and analysis.

5.2.3 Results With Optimised DoWhy Regressor

Using k-folding to select a model that has the greatest ability to generalise, it can be seen in Figure 13 that 3 is the optimal value for k for selecting the most robust version of the model. To accurately compare the performance of the model to that of linear and random forest regressors, all three regressors will be trained both on the dataset in the standard fashion as well as using the same k-folding technique for selecting a robust model. This is

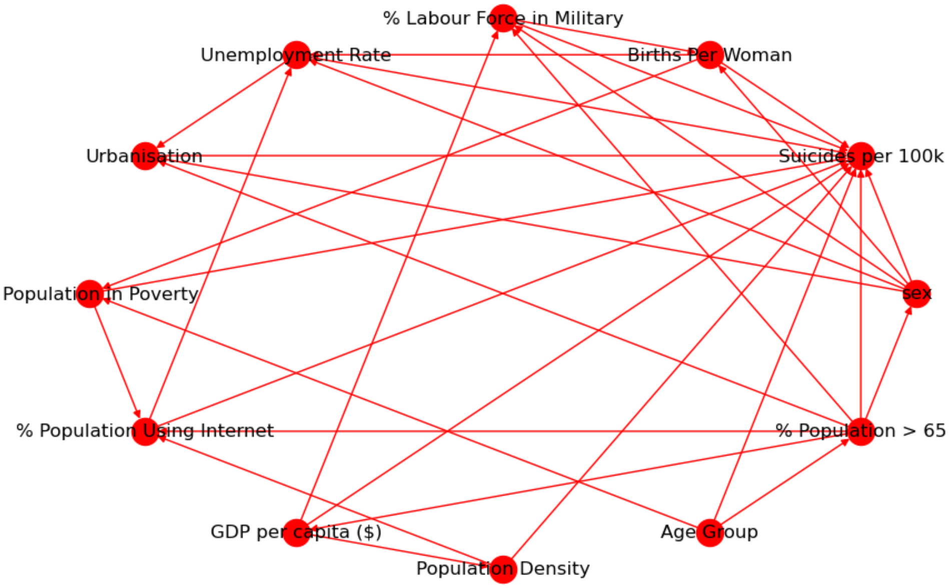


Figure 11: DAG that the graph learner judged to have the most predictive power, resulting in 142.6 MSE on the test data

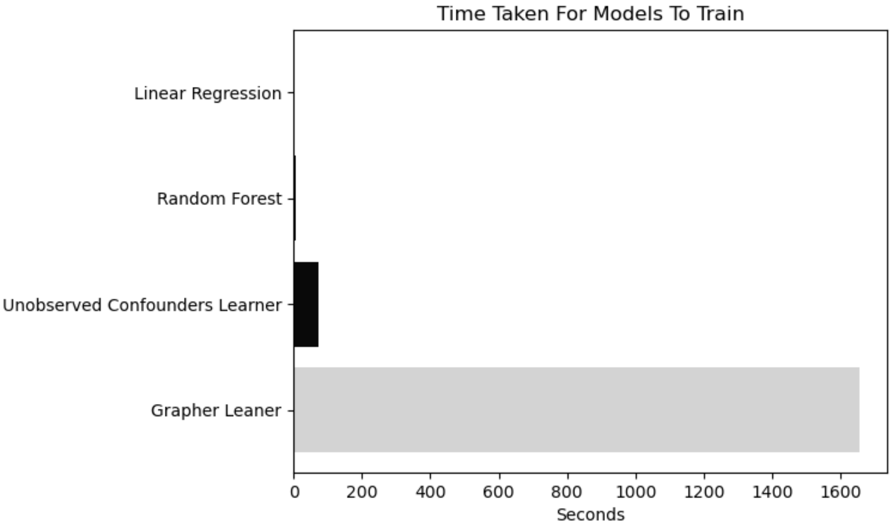


Figure 12: Running times for different regressors

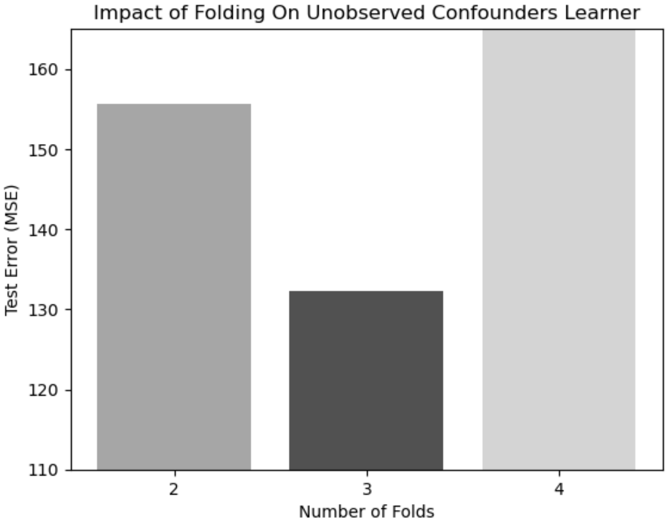


Figure 13: Impact Of K-Folding The Training Data On Performance

because the k-folding method may simply advantage a model by training it on a subset of the dataset (i.e. the first and last thirds of the dataset) that is unusually predictive of unseen data. Figure 14 shows the comparative performances of all aforementioned regressors. All three regressors see their performances improve thanks to the k-folding technique. However, the unobserved confounders learner improve more thanks to the k-folding method than its primary rival, the random forest regressor. When the models are trained in the standard fashion, the unobserved confounders learner, random forest and linear regressors achieve mean squared errors of 137.4, 136.7 and 140.4 respectively. With the threefolding technique to find the most robust version, the regressors then score 132.1, 133.1 and 135.4. This gives the k-folded unobserved confounders learner the best performance of the test dataset.

With the optimised version of the unobserved confounders learner now selected, it is possible to investigate how its performance varies across the test dataset, by linking each of the model's predictions to its corresponding year, age group, sex and country in the original dataset. From aggregating the data by sex and age group (Figure 15), it is clear that the model is considerably less accurate at predicting suicide in males than in females. An obvious explanation for this is that sex is a binary variable, meaning that the model's predictions for males will automatically always outsize their female counterparts by the magnitude of the model's coefficient for sex (this is an inherent flaw in using regression for a binary variable). Consequently, the model is likely to be systematically overestimating the suicide rates of males. With the exception of age group 5-14, the accuracy of the model's predictions for males does decrease as the age group gets older (the reason for age group 5-14 not fitting this trend is likely that suicide numbers for this age group are so small that predictive accuracy is difficult). With females, the model has pretty robust predictive

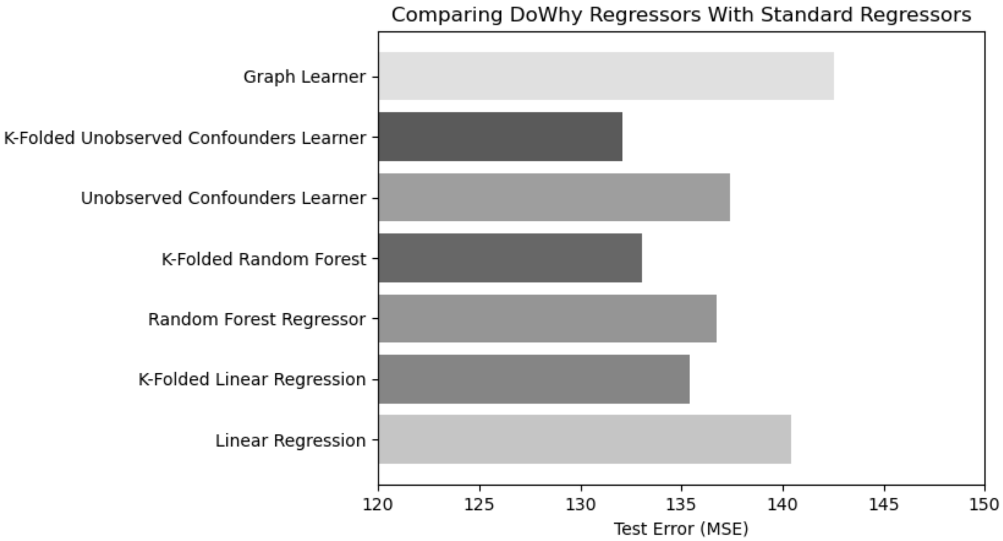


Figure 14: Comparison of DoWhy regressor with standard regressors

power across age groups, but does have a mean squared error of nearly 100 when predicting women over the age of 75 (the age group with which the model has by far the least predictive accuracy).

Aggregating performance data by country demonstrates that the accuracy of the model varies significantly between countries (Figure 16). Whilst being most accurate at predicting suicide rates in Sweden, the United States and Spain, the model performs very poorly on Ukraine and Suriname and catastrophically on Sri Lanka. Figure 17 visualises the correspondence (or lack thereof) between the model's predictions for the test dataset and the actual data. In Sri Lanka, Suriname and Ukraine, the model tended to underestimate. One reason for this is that the regressor lacks the knowledge that suicide rates cannot be below 0, and consequently sometimes predicts negative values for suicide rates (as can be seen in the bottom left corners of the scatter graphs). Across all countries, the model shows itself to be bad at predicting particularly high suicide rates, although a significant number of datapoints were also overestimated.

The performance of the model by year raises the question of how much model performance is dependent on the accuracy of the data. Figure 18 shows that with data from the 1990s up until the mid-2000s, the accuracy of the model is wildly erratic and inconsistent, but the model is relatively consistent in its accuracy for predicting data from the 2010s. It is hard not to consider that this is a reflection of data collection on suicide statistics becoming much more reliable over time - in which case, it is not unreasonable to consider if the wild variation in the model's accuracy by country is due in part to some countries collecting data more reliably than others.

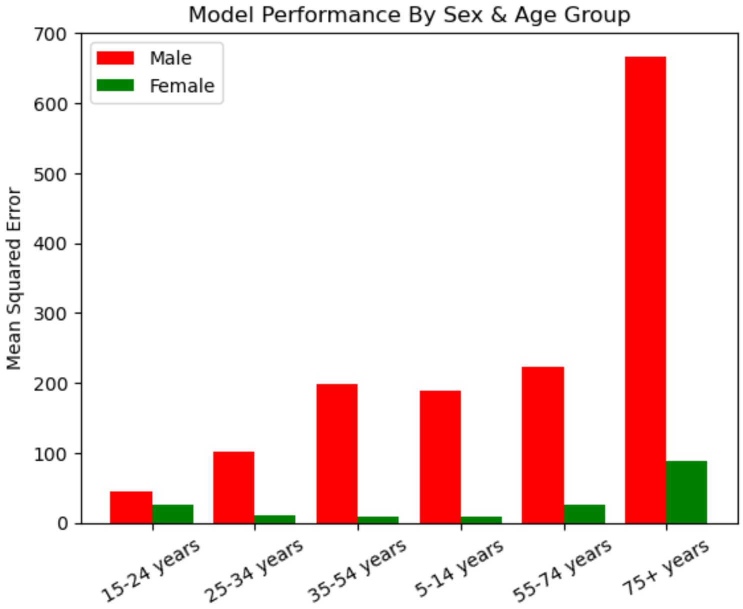


Figure 15: Age and Sex

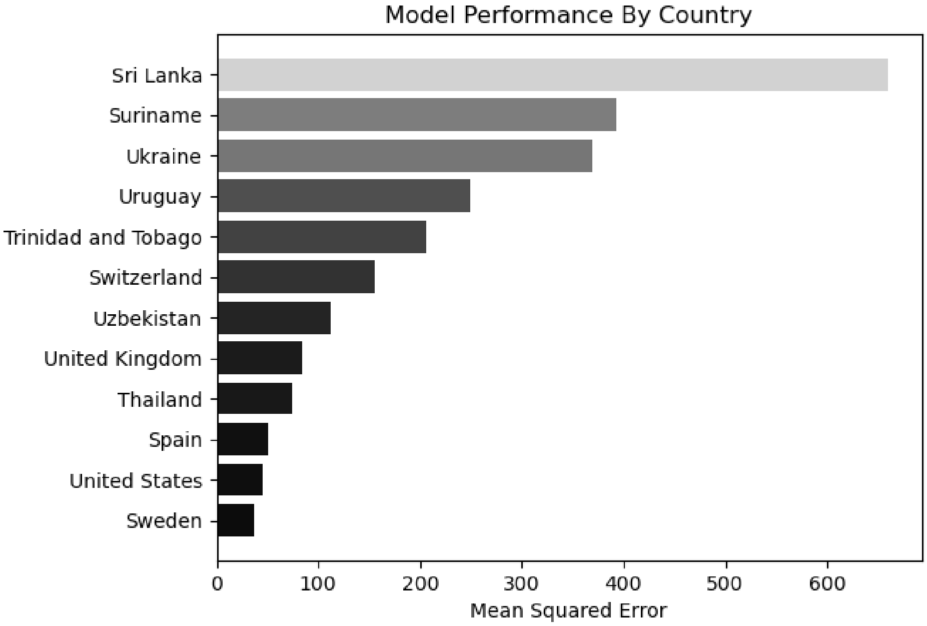


Figure 16: Accuracy By Country

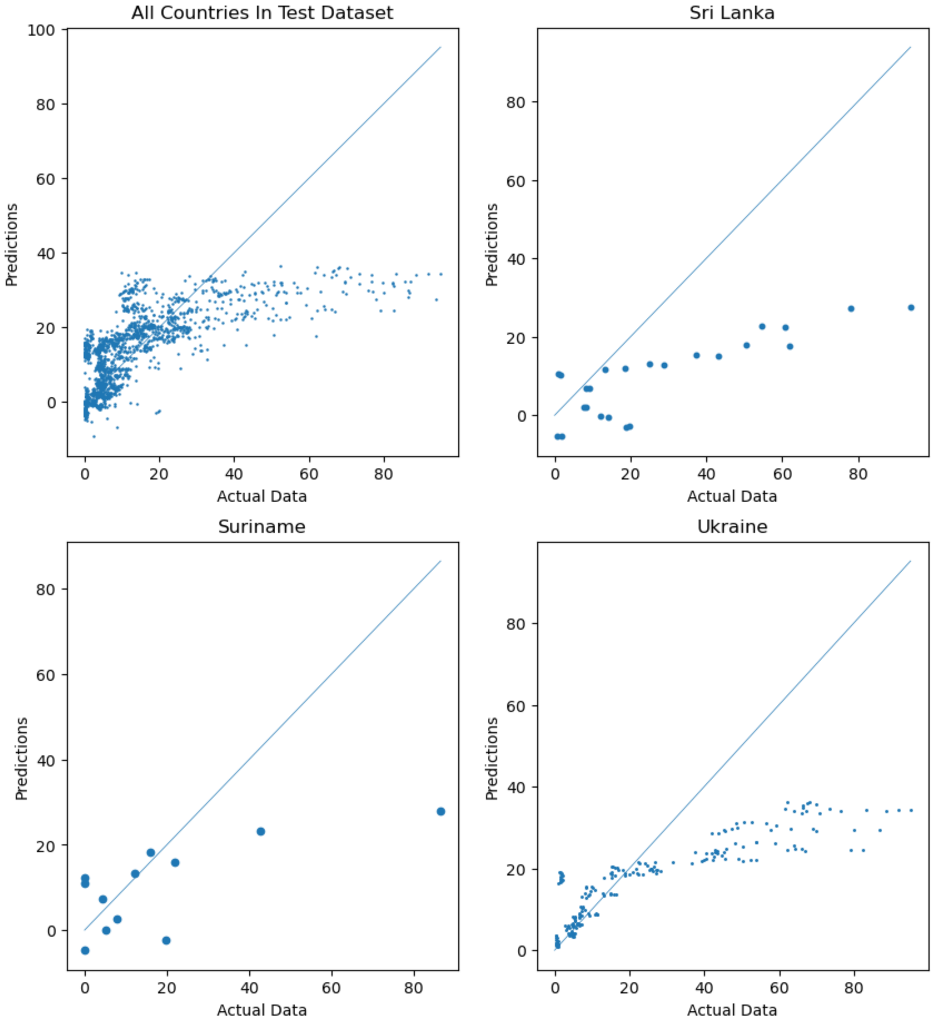


Figure 17: Correspondence Between Predictions And Actual Data

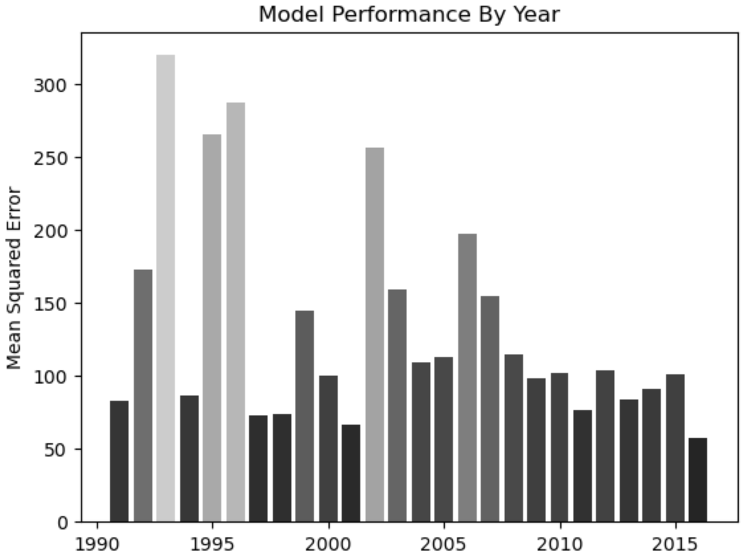
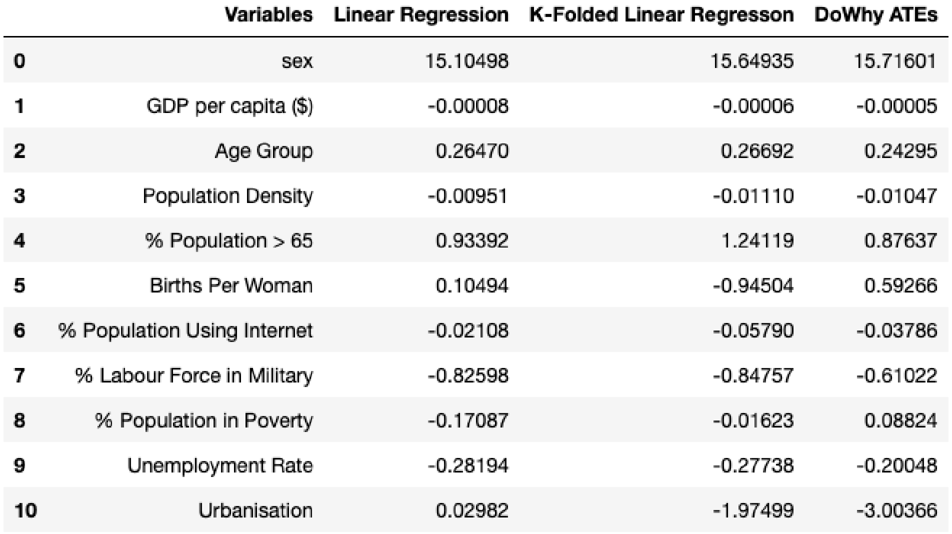


Figure 18: Accuracy aggregated by year

A comparison of the coefficients learnt by linear regression and the ATEs learnt by the unobserved-confounders-learner shows how robust estimates for the variables in the dataset were (Figure 19). Since the model selected from k-folding is trained on a subset of the dataset, it is also illuminating to compare its ATEs to the coefficients learnt by a linear regressor model that was also selected from k-folding. The robustness of sex as a predictor of suicide is demonstrated by the fact that DoWhy's estimated ATE for sex is marginally greater than the conventional regressor's estimate of it (although one consequence of this is the overestimation of suicide risk for males, as evidenced in the greater accuracy in predicting suicide rates for women). With the exception of sex, urbanisation and percentage of the population in poverty, all variables saw their strength diminish when controlling for unobserved confounders. Poverty rate is remarkable for being, along with fertility, a variable that changed from being a negative predictor to a positive predictor.

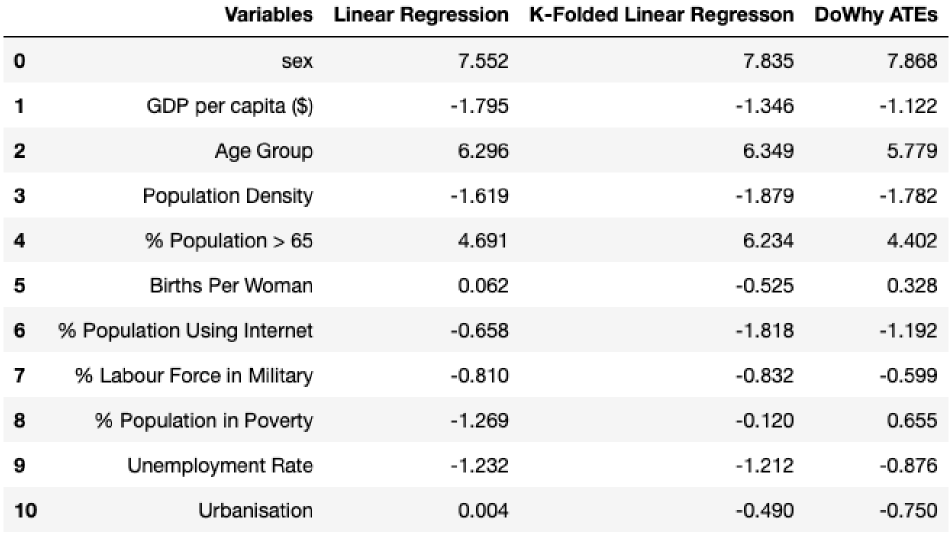
Figure 20 shows the coefficients scaled to standardised data so that the effect strengths of different variables can be more accurately compared. Sex, age and the percentage of the population over 65 prove to have the strongest predictors, although the latter two variables did see their strength diminished by sensitivity analyses. After these, population density, GDP per capita and the percentage of the population accessing the internet proved to be the strongest predictors, with internet access being a marginally stronger (negative) predictor of suicide rates than GDP per capita. In addition to the fact that causal inference techniques diminished the estimated predictive power of all economic variables apart from poverty rate, GDP per capita was the only economic variable to not be weak.

One caveat that has to be made about interpreting these ATEs as causal effects is that



	Variables	Linear Regression	K-Folded Linear Regresson	DoWhy ATEs
0	sex	15.10498	15.64935	15.71601
1	GDP per capita ($)	-0.00008	-0.00006	-0.00005
2	Age Group	0.26470	0.26692	0.24295
3	Population Density	-0.00951	-0.01110	-0.01047
4	% Population > 65	0.93392	1.24119	0.87637
5	Births Per Woman	0.10494	-0.94504	0.59266
6	% Population Using Internet	-0.02108	-0.05790	-0.03786
7	% Labour Force in Military	-0.82598	-0.84757	-0.61022
8	% Population in Poverty	-0.17087	-0.01623	0.08824
9	Unemployment Rate	-0.28194	-0.27738	-0.20048
10	Urbanisation	0.02982	-1.97499	-3.00366

Figure 19: Comparing coefficients and ATEs (unscaled)



	Variables	Linear Regression	K-Folded Linear Regresson	DoWhy ATEs
0	sex	7.552	7.835	7.868
1	GDP per capita ($)	-1.795	-1.346	-1.122
2	Age Group	6.296	6.349	5.779
3	Population Density	-1.619	-1.879	-1.782
4	% Population > 65	4.691	6.234	4.402
5	Births Per Woman	0.062	-0.525	0.328
6	% Population Using Internet	-0.658	-1.818	-1.192
7	% Labour Force in Military	-0.810	-0.832	-0.599
8	% Population in Poverty	-1.269	-0.120	0.655
9	Unemployment Rate	-1.232	-1.212	-0.876
10	Urbanisation	0.004	-0.490	-0.750

Figure 20: Comparing coefficients and ATEs (scaled)

the strength of the simulated unobserved confounders was the same for all variables. In reality, this is unlikely to be the case - especially in the case of variables that negatively predict suicide.

6 Conclusions And Future Directions

6.1 Conclusions

6.1.1 Comparing DoWhy Regressor Models

It is clear from the documented results that the DoWhy regressor that learns unobserved confounders is superior in both predictive accuracy and computational thrift to its alternative. With the graph learner, predictive accuracy on the training data did not reliably correspond to predictive accuracy on the test data.

An obvious explanation for their diverging performances is that the genetic algorithm only sampled a tiny share of the mutations from the sample space of possible mutations. Consequently, the algorithm's search could become trapped in a regions of the sample space in which local minima are suboptimal. This was clearly the case with instances of the graph learner that settled on a training error over 200. However, the graph learner mostly tended to arrive at training errors ≈ 165 - similar to that of the unobserved-confounders-learner. It was also the case that different graph learners with roughly the same training errors would have wildly diverging test errors.

This is relevant to the fact that the genetic algorithm for the graph learner is a combinatorial optimiser searching through a sample space of discrete candidates. Because the candidate mutations will differ from each other by kind rather than degree, this means the possibility of two very different and distinct graph representations that have roughly equivalent predictive power (on the training dataset, at least). Yet because these two different candidates are discrete alternatives, rather than gradations along a continuum, this means that when it comes to predicting unseen data, the success of one candidate model will not necessarily correspond to the success of the other candidate model (as already seen in Figures 8 and 9). By contrast, the unobserved-confounders-learner reliably converged upon an estimate for the strength of unobserved confounders that would in turn reliably predict the model's performance on both training and test datasets.

Beyond the aforementioned differences between genetic algorithms and gradient descent, there is another explanation for the discrepancy between the different versions of the DoWhy regressor. It is possible that there are simply not many confounding relationships between the variables featured in this project's dataset and as a result it was ill-advised to pursue a model of confounding relationships between variables as a way of arriving at accurate ATEs for the predictors in the dataset.

6.1.2 Causal Inference And Suicide

The primary result of this dissertation showed that incorporating causal inference techniques does improve the ability of regression to predict suicide rates across countries. This was essentially due to the ability of DoWhy to simulate and control for unobserved confounders. Despite its enormous computational expense, the graph learner proved not to be an improvement upon standard linear regression models. It is very plausible that this indicates that assuming causal independence between the sociological variables featured in the dataset will do little or nothing to impair our understanding of the causes of suicide.

If the ATEs estimated by DoWhy are assumed to be superior to linear regression's coefficients as estimates of the variables' causal effect on suicide, then the ATEs have interesting implications for theories of the causes of suicide. As already noted, causal inference techniques diminished the strength of most economic predictors relative to the strength linear regression gave those predictors. This does not necessarily conflict with any of the theories of suicide mentioned in the literature review, as those theories only address economic stress as a predictor of suicide when it impacts family relationships.

Durkheim's theory of differing suicide rates between Catholics and Protestants was that the communal nature of Catholic religious practice and the individualistic nature of Protestantism made Protestants more socially atomised relative to Catholics, and thus more likely to commit suicide. However, variables in this dissertation's dataset that could be said to be proxies for the strength of family ties and traditions do not necessarily support this theory. The k-folded unobserved-confounders-learner measured fertility rates to be a subtle positive predictor of suicide rates. Additionally, the unobserved-confounders-learner also magnified the negative predictive power of urbanisation relative to its linear regression counterparts. A full discussion of whether or not urbanisation is a proxy for social atomisation and the weakening of family ties is a sociological beyond the scope and domain of this dissertation. However, it is not unreasonable to view this project's estimation of urbanisation as a nontrivial negative predictor of suicide as a finding in conflict with Durkheim's theories.

Both linear regression and the causal inference methods identified the percentage of a country's population in the military as a negative predictor of suicide rate. The estimate was deflated by causal inference relative to the linear regression estimate, but remained negative nonetheless. As mentioned in the literature review, Joiner outlined two theories as to why a history in the military may be predictors of suicide risk: i) that a history of exposure to violence or bodily harm makes an individual less weak of suicide; and ii) that a willingness to enlist in the military is caused by a genetic predisposition towards self-sacrificial behaviour (which, according to Joiner, could also predispose someone to suicide). It would be inherently problematic to claim that these findings disprove Joiner's theory since military service does not entail experience of war, and in societies with conscription, membership of the military does not imply a predisposition towards self-sacrifice.

6.2 Future Directions

6.2.1 Improving The Graph Learner Model

The fact that the graph learner's performance on the training dataset did not reliably predict its performance on the test dataset has already been noted. However, amendments could be made to improve its ability to converge not just upon an acceptable training error, but also upon the same graph representation. This could be achieved not just be increasing the proportion of the sample space that the model explores, but also ensuring that mutations are of varying sizes. This would make the model less likely to get trapped in a particular region of the sample space and more likely to to explore a range of mutations that is more representative of the entire sample space. Increasing the number of mutations tested would necessarily come at a higher computational cost, which may require more advanced GPUs. Additionally, rather than a genetic algorithm, graph neural networks could be used as the means by which to learn the optimal graph representation confounding relationships between observed variables.

6.2.2 Learning Different Unobserved Confounders For Different Variables

Although it was the most important factor in the DoWhy regressor's ability to generalise is its simulation of unobserved confounders, this stage could certainly be improved upon. As has been remarked upon in the results section, it is certainly suboptimal and inaccurate that the strength of simulated unobserved confounders was uniform for all variables. Allowing the model to learn different effect strengths for the unobserved confounders of different variables would ensure that all variables have had their robustness equally tested. It is also particularly necessary for a set of variables where some negatively and some positively predict the target value.

7 References

Battocchi et al. (2019). *EconML: A Python Package for ML-Based Heterogeneous Treatment Effects Estimation.* [online] Available at: https://github.com/py-why/EconML.

Castillo-Sánchez, G., Marques, G., Dorronzoro, E., Rivera-Romero, O., Franco-Martín, M. and De la Torre-Díez, I., 2020. Suicide risk assessment using machine learning and social networks: a scoping review. *Journal of Medical Systems*, 44(12), p.205.

Chen, H., Harinen, T., Lee, J.Y., Yung, M. and Zhao, Z., 2020. Causalml: Python package for causal machine learning. *arXiv preprint arXiv:2002.11631.*

Chu, C., Buchman-Schmitt, J.M., Stanley, I.H., Hom, M.A., Tucker, R.P., Hagan, C.R., Rogers, M.L., Podlogar, M.C., Chiurliza, B., Ringer, F.B. and Michaels, M.S., 2017. The interpersonal theory of suicide: A systematic review and meta-analysis of a decade of cross-national research. *Psychological Bulletin*, 143(12), p.1313.

Durkheim, É., 1897. *Le Suicide: Étude De Sociologie.* Alcan, Paris.

Goddard, A.V., Xiang, Y. and Bryan, C.J., 2022. Invariance-based causal prediction to identify the direct causes of suicidal behavior. *Frontiers in Psychiatry*, p.2598.

Joiner, T.E., Hom, M.A., Hagan, C.R. and Silva, C., 2016. Suicide as a derangement of the self-sacrificial aspect of eusociality. *Psychological Review, 123*(3), p.235.

Kaddour, J., Lynch, A., Liu, Q., Kusner, M.J. and Silva, R., 2022. Causal machine learning: A survey and open problems. *arXiv preprint arXiv:2206.15475.*

Kaggle (2018). *Suicide Rates Overview 1985 to 2016.* [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/russellyates88/suicide-rates-overview-1985-to-2016.

Kumar, V., Sznajder, K.K. and Kumara, S., 2022. Machine learning based suicide prediction and development of suicide vulnerability index for US counties. *NPJ Mental Health Research, 1*(1), p.3.

Pearl, J., 2009. *Causality.* Cambridge University Press.

Pearl, J., Mackenzie, D., 2018. *The Book of Why: The New Science of Cause and Effect.* Basic Books.

Roy, A., Nikolitch, K., McGinn, R., Jinah, S., Klement, W. and Kaminsky, Z.A., 2020. A machine learning approach predicts future risk to suicidal ideation from social media data. *NPJ Digital Medicine, 3*(1), p.78.

Sharma, A. and Kiciman, E., 2020. DoWhy: An end-to-end library for causal inference. *arXiv preprint arXiv:2011.04216.*

Van Orden, K.A., Witte, T.K., Cukrowicz, K.C., Braithwaite, S.R., Selby, E.A. and Joiner Jr, T.E., 2010. The interpersonal theory of suicide. *Psychological Review, 117*(2), p.575.

World Bank (2022). *Armed forces personnel (% of total labor force) — Data.* [online] Worldbank.org. Available at: https://data.worldbank.org/indicator/MS.MIL.TOTL.TF.ZS.

World Bank (2021). *Urban population (% of total) — Data.* [online] Worldbank.org. Available at: https://data.worldbank.org/indicator/SP.URB.TOTL.IN.ZS.

World Bank (2020). *Population density (people per sq. km of land area) — Data.* [online] Worldbank.org. Available at: https://data.worldbank.org/indicator/EN.POP.DNST.

World Bank (2020). *Fertility rate, total (births per woman) — Data.* [online] Worldbank.org. Available at: https://data.worldbank.org/indicator/SP.DYN.TFRT.IN.

World Bank (2021). *Individuals Using the Internet (% of population) — Data.* [online] Worldbank.org. Available at: https://data.worldbank.org/indicator/IT.NET.USER.ZS.

World Bank (2022). *Population ages 65 and above (% of total) — data.* [online] Worldbank.org. Available at: https://data.worldbank.org/indicator/SP.POP.65UP.TO.ZS.

World Bank (2021). *Unemployment, total (% of total labor force) (modeled ILO estimate) — Data.* [online] Worldbank.org. Available at: https://data.worldbank.org/indicator/sl.uem.totl.zs.

World Bank (2022). *Poverty headcount ratio at national poverty lines (% of population) — Data.* [online] Worldbank.org. Available at: https://data.worldbank.org/indicator/SI.POV.NAHC.