Final Year Project

Explaining Algorithms: Developing an interface to explain algorithms and their trade-offs

Susannah D'Arcy

Student ID: 16200408

A thesis submitted in part fulfilment of the degree of

BSc. (Hons.) in Computer Science with Data Science

Supervisor: David Coyle

UCD School of Computer Science

University College Dublin

May 7, 2021

Table of Contents

Abstract

Algorithms are a black box, data goes in, and a decision is made. There is little to no insight into the results. Consequently, there are difficulties in the understanding of these algorithms and a decrease in trust. Due to this lack of transparency, trade-offs and bias can also be introduced which can have major societal consequences. Ensuring a method to explain these algorithms and their trade-offs has become an urgent task with the increasing usage of these systems in our society.

To improve the understanding of these decision-making algorithms, this report will outline the process of developing an interactive interface. An interface that showcases the impact of each feature on the classification results. The interface also demonstrates the different trade-off within algorithms and explains the difficulties with balancing them. This report will focus on the trade-offs between minimising error types and balancing between fairness and accuracy. Such as questioning whether you prioritize removing bias at the cost of performance.

To develop this interface the report will first research into the different models for explaining algorithms, and their trade-offs. Then methods of visualising the decision-making process, and visualisation techniques to showcase the performance of the algorithm. The interface was user-tested, and from this, it was discovered that our interface improved the understanding of the algorithm process via feature importance. It also allowed users to explore the ethics surrounding the algorithmic trade-offs, by providing an interactive platform in which they can experiment, and learn. With an overall result of the interface improving people's understanding of algorithms.

Details

The code used to develop the models and the interface can be seen on the CS Gitlab linked below. The code includes a jupyter notebook which goes into more detail about the implementation of the models. The interface was also deployed and can be used in the below link.

- Gitlab: Visualising algorithmic trade-offs
- Interface: Interface Demo

This project's proposal 7.1 and other proposals (7.2, 7.3), developed in Semester 1 are presented in the Appendix.

Acknowledgements

I would like to give thanks my supervisor David Coyle, who provided me with great guidance and mentor-ship during the whole process. I would also like to thank the fellow students who participated in my evaluation study, who all provided useful feedback to my interface.

Chapter 1: **Introduction**

Algorithms are being used to decide increasingly important decisions. They are used to filter job applications, provide medical feedback, they are used to decide grade scores, and whether or not someone should leave on parole. While these algorithms have sped up the process, and are said to provide an 'impartial' system. However there is a lack of transparency in the classification process, and due to this, there can be some underlying bias introduced to these algorithms. Such as for the 2020 A-levels results algorithm, had a major bias in that it rewarded students in private schools better than public schools [1]. Due to this, it is ever important for the general public to understand and know about the trade-offs of these algorithms.

The problem with using these algorithms is that the majority of AI is a black box, such that features are inputted, and a result is outputted. With no clear reasoning or explanation as to how and why the algorithm chose that result. Due to this research papers focus on the idea of white-boxing AI to create explainable AI (XAI). XAI can be applied during development or used afterwards to explain and teach algorithmic decision making. To improve understanding of algorithms and their trade-offs, intractability has been proven to improve comprehension [2], and thus for each explanation method used, interaction is a key element in their design and evaluation. Interaction has been mainly implemented by changing the input variables values [2] [3], or by allowing the balancing and prioritising of trade-offs via sliders [4] [5]. For XAI design, the effectiveness of explaining feature importance has been researched. Their goal to express how each feature affects the classification, and therefore explain how the algorithms 'think' [2] [6].

Algorithms can have a multitude of underlying trade-offs. Such as a balance between error types, or with recommender algorithms the balance between accuracy and variety of recommendations. Research has often expressed that there can be a bias within the models. However, reducing this bias can have an effect on the performance of the model [4]. The balancing and identification of these trade-offs rely on developers, users, and researchers. However, due to the lack of transparency, this task can be difficult. Research has focused on explaining common trade-offs to users via informative interfaces.

There is a lack of research into generative visuals for both explaining the algorithm and its trade-offs. Most research has gone into solely developing XAI, or evaluating the trade-offs. However, this report investigates usages of both explanation methods into generating a well-rounded interface, which takes the XAI a step further as it allows users to manipulate and interact with features and AI. Which will aid the understanding of algorithms and their trade-offs. As theoretically to improve the understanding of the trade-offs there needs to be an underlying knowledge of the algorithm itself. This project will be evaluating and researching methods in explaining both aspects of algorithms.

This project aims to research different methods for generating models for the explanation of algorithmic decision-making and their trade-offs. With a focus on two research and development goals. First, finding and developing models which showcase the feature importance, and algorithmic trade-offs. Secondly identifying effective methods of visualising these models both textually and graphically. A key design aim is to allow for interaction, as this will allow users to explore the algorithms, and elevate their understanding.

With users improved understanding, it's also important to access their trust in AI. As with the increasing use of these algorithms, it is important to gauge whether or not users trust the results of these algorithms, and then decide on if these systems should be deployed in society. There is conflicting research into whether different explanation methods affect the trust in AI. Hao-Fei Cheng et. al. found that the explaining of feature impact did not affect the users' trust in algorithmic decisions [2]. While Bowen Yu et. al. reported that the showcasing of algorithmic trade-offs change 47.4% of participants trust, with an even split of reducing, and improving trust [4]. Due to this conflict, this paper will also evaluate the use of both feature and trade-off explanation in effecting users trust in decision-making algorithms.

The goal of this project is to develop an interface to help users understand decision-making algorithms and their trade-offs. It is not the collection of data or the development of a decision-making algorithm. Therefore for this project, will be using the popular dataset and algorithm COMPAS, which is used by judges, probation and parole officers, to assess a defendants likelihood to re-offend. It has been used and been proven to be a good representation of the algorithmic trade-offs [7] [8] [2] [9] [10] [4] [11] [12]. The COMPAS algorithm showcases the ethics and morality of these decisions, as different error types can have a major impact. There is also an underlying bias in the algorithm against African Americans [13], however trying to remove this bias can result in a decrease in accuracy.

Feature importance will be used to explain how the COMPAS algorithm uses the features to classify a defendant. Different tools and systems to gather and display the feature impact to the user will be explored. Two common trade-offs were selected, the balance between error types, and fairness versus accuracy. Each of these will require different models to showcase their effect on the algorithm.

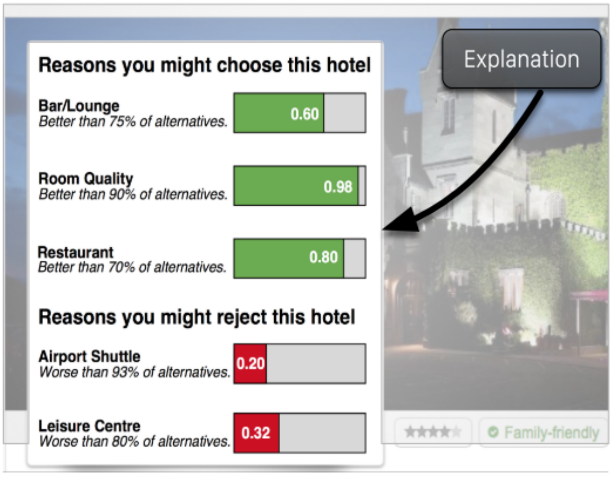
For the model development, an interface will be generated to explain the COMPAS algorithm. The user will be able to interact with the algorithm by altering the input of an inmate and seeing the resulting feature impact on the classification. They will also be able to balance between the trade-offs to see the effect it has on the performance of the algorithm. User testing will then be used to evaluate the effectiveness of the interface.

Chapter 2: Related Work

2.1 Explaining Algorithms via Feature importance

2.1.1 Bar Chart Feature Importance

A key method of showing feature importance is through bar charts. The length of a bar signifies the importance of a feature in the classification system. Multiple bars can be used to show the most impactful features [14] [6]. The side by side bars allows for better comparison between features. Colour can be used to show the sentiment of the feature for a particular class. Khalil Muhammad et al. used feature importance to rank features in a set of explanations for a hotel recommender system. A bar and colour were then used to show the sentiment and significance of the feature for a positive (green) or negative (red) recommendation (Figure 2.1) [6].



Reasons you might choose this hotel

Bar/Lounge
Better than 75% of alternatives. 0.60

Room Quality
Better than 90% of alternatives. 0.98

Restaurant
Better than 70% of alternatives. 0.80

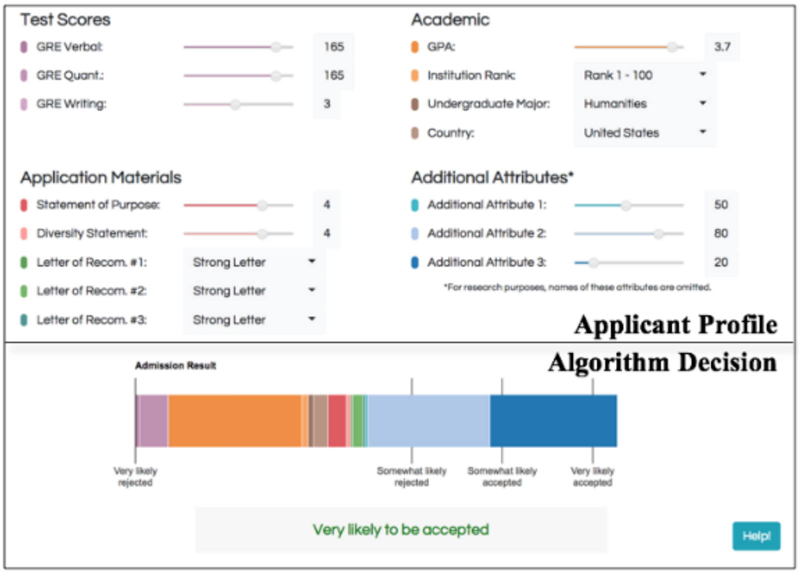
Reasons you might reject this hotel

Airport Shuttle
Worse than 93% of alternatives. 0.20

Leisure Centre
Worse than 80% of alternatives. 0.32

Figure 2.1: Bar-chart representation of feature importance shown in *A live-user study of opinionated explanations for recommender systems* by Khalil Muhammad et al. [6]

To showcase how each feature affects the overall classification, stacked bar charts have been used. They can be used to signify the stages of classification and the levels needed to change the class vote [2] (Figure 2.2). However, it makes it harder for the user to compare each feature individually as each bar is not on the same axis.



Test Scores

GRE Verbal: 165

GRE Quant.: 165

GRE Writing: 3

Academic

GPA: 3.7

Institution Rank: Rank 1 - 100

Undergraduate Major: Humanities

Country: United States

Application Materials

Statement of Purpose: 4

Diversity Statement: 4

Letter of Recom. #1: Strong Letter

Letter of Recom. #2: Strong Letter

Letter of Recom. #3: Strong Letter

Additional Attributes*

Additional Attribute 1: 50

Additional Attribute 2: 80

Additional Attribute 3: 20

*For research purposes, names of these attributes are omitted.

Applicant Profile

Algorithm Decision

Very likely to be accepted

Figure 2.2: Stacked bar-chart feature importance representation shown in *Explaining Decision-Making Algorithms through UI: Strategies to Help Non-Expert Stakeholders* by Hao-Fei Cheng et al. [2]

2.1.2 Models for Feature Importance

A model for finding the feature importance can be adapted from features weights in recommender systems. Khalil Muhammad et al. shows the importance of a feature by calculating the relative count of that feature being mentioned in reviews for that hotel (Equation 2.1). The significance was the positive rate of that feature, with a positive significance of a rate higher than 0.7 (and negative otherwise) (Equation 2.2) [6].

$$imp(f_i, h_i) = \frac{count(f_i, h_i)}{\sum_{\forall f^{|} \in R(h_i)} count(f^{|}, h_i)} \tag{2.1}$$

as shown by [6]

$$sent(f_i, h_i) = \frac{pos(f_i, h_i)}{pos(f_i, h_i) + neg(f_i, h_i)} \tag{2.2}$$

as shown by [6]

A widely used model is partial dependence. This method slightly changes a value of a feature and observes the effect on the output classification or prediction with these changes you can generate a partial dependence plot (Figure 2.3), which shows how the affected prediction changes with the feature value. Plot can be calculated using equation (2.3) . By repeating this for each feature you can find the impact of each feature in the algorithm [3].

$$pdp_f(v) = \frac{1}{N} \sum_{i}^{N} pred(x_i) \quad \text{with} \quad x_{if} = v \tag{2.3}$$

as shown by [3]

Josua Krause et al. uses these partial dependence plots to visually explain the impact of the feature on the classification of diabetic diagnosis. These plots were used by data scientists and they recorded that it improved their understanding of the predictive model as it allowed them to drill down and find the most impactful features [3]. These plots also allow the user to compare the differences in their input values, unlike with bar charts which only show one instance of a feature set.

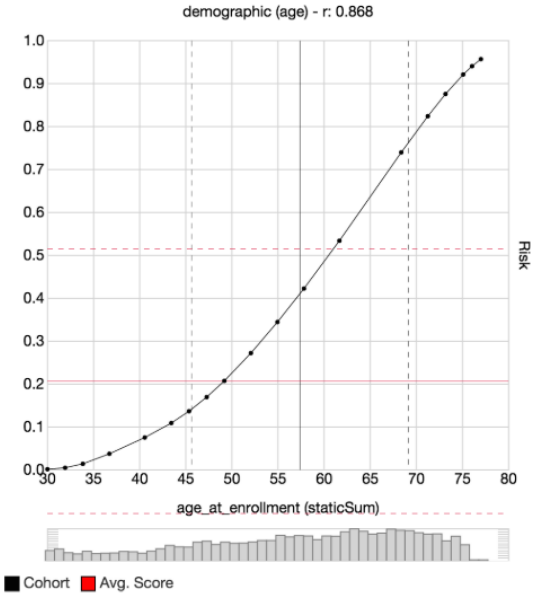


Figure 2.3: Partial dependence plot shown in *Interacting with Predictions: Visual Inspection of Black-box Machine Learning Models* by Josua Krause et al. [3]

2.2 Explaining Algorithmic Trade-offs

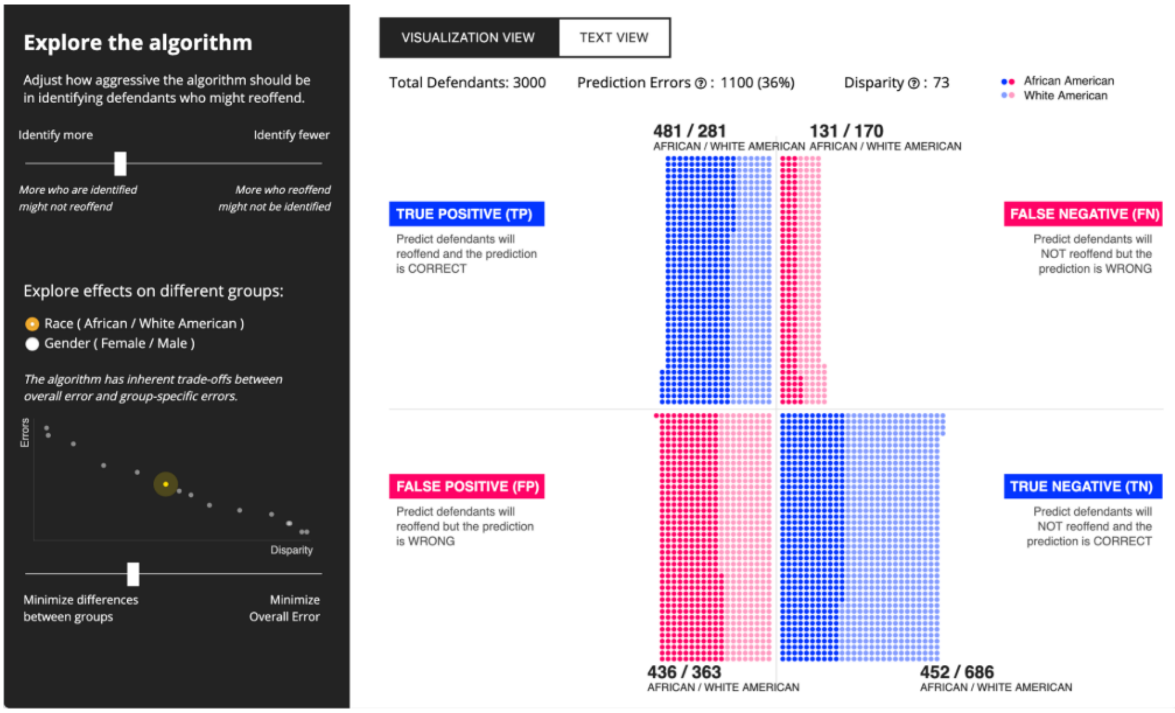
2.2.1 Fairness in AI

To represent fairness quantitatively there are two major definitions. These fairness definitions can be used to create fairness constraint models. The first being Demographic Parity (DP) which states that selection rate for any gender/racial must be at least 80% of the rate for the group with the highest rate [8] Meaning that there cannot be a high variance in the classification counts of a racial/gender group, and preferably each group was classified or selected the same amount of times, for example, an equal amount of males and females hired.

In contrast, Equalized Odds (EO) was defined, it focuses on the equality of odds or opportunity. Meaning that the classifier predicts the label equally for all values of the attribute [8] [15]. Rather than focusing on equal counts. Using the hiring example, people or groups will be hired equally if they have an equal amount of qualifications. EO is said to improve upon DP as imbalanced datasets can lead to random classification for the minority groups, and it removes the possibility of finding correlations between the groups and the classification [8].

2.2.2 Fairness and Accuracy

The main trade-off for fairness is the loss of accuracy in a system as it has been proven to increase the error rate of classification algorithms. To explain this trade-off researchers have allowed users to interact and balance between fairness and accuracy, and from that they can compare the results in the classification [4] [5] [16]. Bowen Yu, et al. used the COMPAS Recidivism Algorithm which classifies whether a criminal will reoffend, this is used to showcase the trade-off between classifying the likelihood of reoffending equally or accurately to racial groups [4]. Each with their own moral and ethical importance. To showcase these they created an interactive interface (Figure 2.4).



Explore the algorithm

Adjust how aggressive the algorithm should be in identifying defendants who might reoffend.

Explore effects on different groups:

● Race (African / White American)
○ Gender (Female / Male)

The algorithm has inherent trade-offs between overall error and group-specific errors.

Total Defendants: 3000

Prediction Errors ⓘ: 1100 (36%)

Disparity ⓘ: 73

TRUE POSITIVE (TP): Predict defendants will reoffend and the prediction is CORRECT

FALSE NEGATIVE (FN): Predict defendants will NOT reoffend but the prediction is WRONG

FALSE POSITIVE (FP): Predict defendants will reoffend but the prediction is WRONG

TRUE NEGATIVE (TN): Predict defendants will NOT reoffend and the prediction is CORRECT

Figure 2.4: Interface showcasing algorithmic trade-off, with matrix view shown in *Keeping designers in the loop: Communicating inherent algorithmic trade-offs across multiple objectives* by Bowen Yu et al. [4]

Yunfeng Zhang et al. extended fairness trade-offs by implying that the decrease in accuracy was due to higher errors which would lead to higher cost. They then showed this trade-off by allowing users to prioritise profits (minimise errors) or minimise the disparity between age groups [5].

A system which reduces errors while maximizing fairness can be modelled using cost-sensitive classification and by using the fairness definitions to quantitatively represent the level of fairness, and assign the desired cost. This model was used by Alekh Agarwal, et al. (Equation 2.4) in conjunction with gradient reduction to generate a fair classifier [8].

$$\underset{h \in H}{\operatorname{argmin}} \sum_{i=1}^{n} h(X_i)C_i^1 + (1 - h(X_i))C_i^0 \tag{2.4}$$

as shown by [8] [4]

A Pareto Curve can be used to show the relationship between accuracy and fairness. It will allow a user to balance between prioritizing fairness and accuracy by selecting the part of the curve, the right side of the Pareto curve (Figure 2.5) prioritises accuracy while models on the left side prioritize fairness. Bowen Yu, et al. generated their Pareto curve using cost sensitivity analysing, with fairness being measured as the disparity between the number of FP and FN (Equation 2.4) [4].

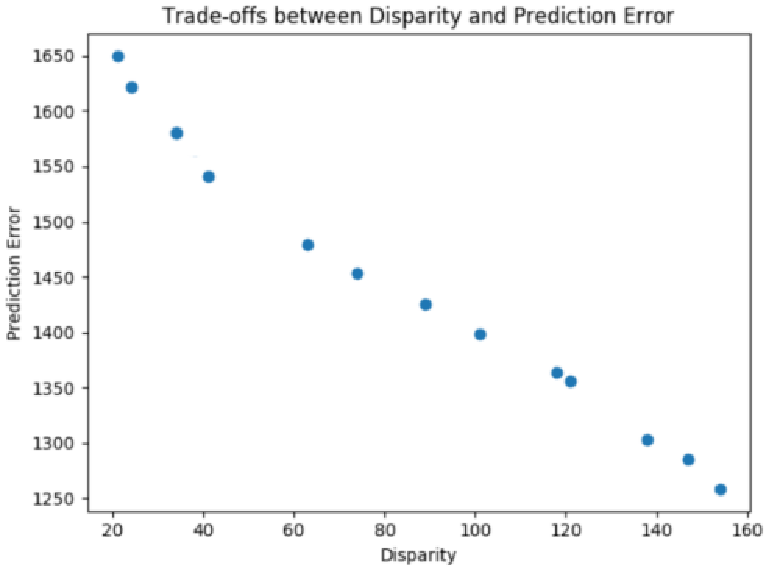


Figure 2.5: Pareto curve for balancing accuracy and fairness shown in *Keeping designers in the loop: Communicating inherent algorithmic trade-offs across multiple objectives* by Bowen Yu et al. [4]

2.2.3 Trade-offs in FP and FN Rates

This balancing of type errors can represent the aggressiveness of the classifier, in such that there is a trade-off between identifying fewer, or more of a class [16] [8] [4]. Bowen Yu, et al. also showed that these types of errors can also have ethical consequences. Such that by minimising false positives can result in higher false-negative rates [4]. In the context of COMPAS, this results in non-re offenders not being falsely accused, while criminals who will reoffend being incorrectly pardoned.

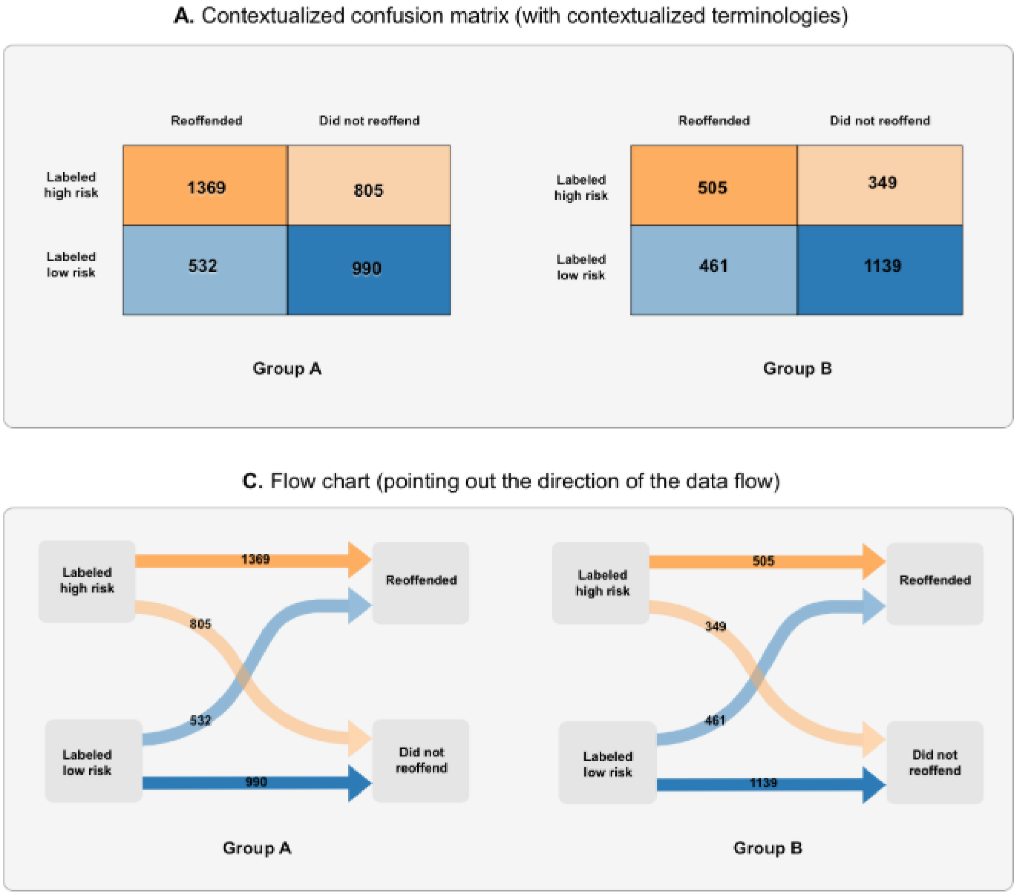
One method of modelling this balance is by developing two different versions of a classifier, one which is High Precision which minimises FP error, the other is High Recall which minimises FNs. With this, the user can set the threshold to the degree in which they use these two models [16].

Another method is to use a varied threshold in classification. In which you extract the probabilities of a classification, and change them via a threshold to increase the rate of FP or FN [4] [17]. This type of threshold tuning is used in ROC analysis, and can be used to allow intractability by allowing the user to change the threshold [18] [17].

2.3 Visualising Trade-offs

To explain these trade-offs a common method is to visually show the performance of the algorithm, such as showing the resulting accuracy [16].To extend this performance explanation by using confusion matrices (Figure 2.4). Bowen Yu, et al. used a confusion matrix quadrants to represent the classifications of inmates. Each inmate was a dot in the matrix quadrant, and thus collectively represents the count for that section. Due to this representation, they could also use shade to represent the 2 racial/gender subgroups. [4].

Hong Shen, et al. researched the uses of confusion matrices and their understanding by the public. From their research, they hypothesized that there is confusion around the terminology, such as FP and FN. Due to this, a contextualised confusion matrix allows for a better understanding [10]. Using the COMPAS as an example the predicted positive will change to 'Labeled high risk', and the actual positive will be changed to 'Reoffended'. They also showed that using flow charts instead of matrix quadrants can improve the understanding as they visually express the directional relationship between categories. Which can boost the understanding of algorithm performance itself [10].



Group A

	Reoffended	Did not reoffend
Labeled high risk	1369	805
Labeled low risk	532	990

Group B

	Reoffended	Did not reoffend
Labeled high risk	505	349
Labeled low risk	461	1139

Figure 2.6: Example of an Contextualized confusion matrix and Flow Chart representation in *Designing Alternative Representations of Confusion Matrices to Support Non-Expert Public Understanding of Algorithm Performance* by Hong Shen et al. [10]

2.4 Improving Trust

During the research into explanation techniques, Hao-Fei Cheng, et al. discovered that while their interactive white-box explanation method improved the understanding of decision-making algorithms, they had no effect on the trust of these systems. From this, they suggested that only increasing the transparency of a decision doesn't increase the trust in it [2].

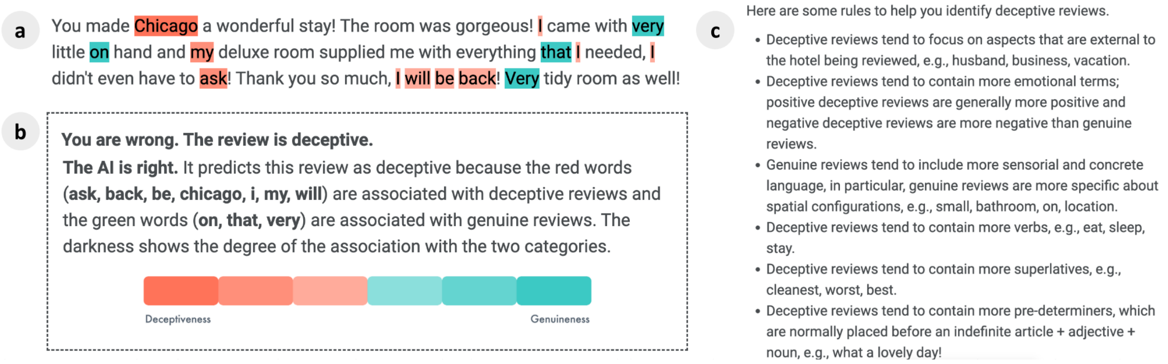
Many researchers used accuracy to improve the showcase of the trustworthiness of the system [4] [16]. Yunfeng Zhang et al. explored the effectiveness of confidence scores and explanations of the accuracy and trust of AI-assisted decision-making systems. With this research, they evaluated whether an explanation or the confidence of the AI was enough to improve accuracy and trust [5]. Similarly to Hao-Fei Cheng, et al.they found that explanations improve the understanding of classification, with no effect on the trust. While confidence scores improved the trustworthiness of AI, with no improvements in the accuracy.

To further this study, this project hopes to test the impact of algorithm explanation, and trade-off explanations on trust. Also if the inclusion of accuracy scores increase both trust and understanding if used in conjunction with the two-stage explanation.

2.5 Textual Explanations

2.5.1 Textual Feature Importance

A method for explaining an algorithm's decisions is to highlight and identify key features which have impacted the decision-making system. With these features, the importance can be signified by the colour and hue of the highlighted feature, or the use of symbols [9] [7]. For example, Vivian Lai et al. researched methods of training and teaching users to identify deceptive reviews. As shown in Figure 2.7 for each review, important featured words were highlighted and coloured depending on the effect. (ie. red words are associated with deceptive reviews, with a darker tone signifying the importance). These examples with feature importance lead to better human classification of reviews compared to training methods without, and they showed a better understanding of how the algorithm identified deceptive reviews [9].



You made Chicago a wonderful stay! The room was gorgeous! I came with very little on hand and my deluxe room supplied me with everything that I needed, I didn't even have to ask! Thank you so much, I will be back! Very tidy room as well!

You are wrong. The review is deceptive.
The AI is right. It predicts this review as deceptive because the red words (**ask, back, be, chicago, i, my, will**) are associated with deceptive reviews and the green words (**on, that, very**) are associated with genuine reviews. The darkness shows the degree of the association with the two categories.

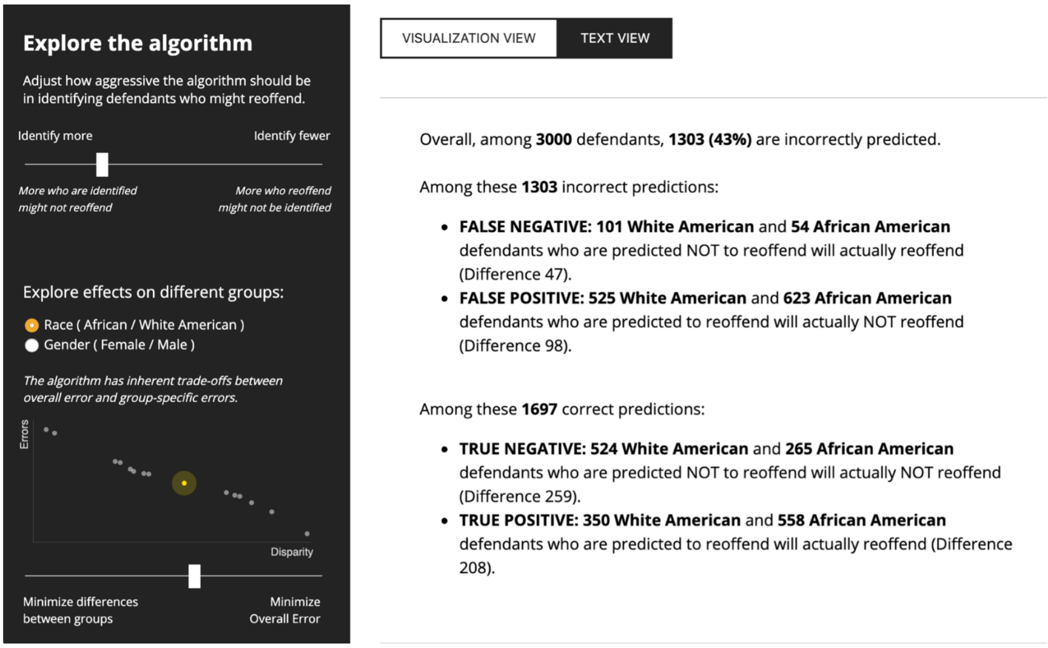
Here are some rules to help you identify deceptive reviews.

- Deceptive reviews tend to focus on aspects that are external to the hotel being reviewed, e.g., husband, business, vacation.
- Deceptive reviews tend to contain more emotional terms; positive deceptive reviews are generally more positive and negative deceptive reviews are more negative than genuine reviews.
- Genuine reviews tend to include more sensorial and concrete language, in particular, genuine reviews are more specific about spatial configurations, e.g., small, bathroom, on, location.
- Deceptive reviews tend to contain more verbs, e.g., eat, sleep, stay.
- Deceptive reviews tend to contain more superlatives, e.g., cleanest, worst, best.
- Deceptive reviews tend to contain more pre-determiners, which are normally placed before an indefinite article + adjective + noun, e.g., what a lovely day!

Figure 2.7: Feature importance representation with training text for spotting deceptive hotel reviews shown in *"Why is 'Chicago' deceptive?" Towards Building Model-Driven Tutorials for Humans* by Vivian Lai et al. [9]

2.5.2 Textual Fairness Trade-offs

For textually explaining the trade-offs of these algorithms Bowen Yu et al. used a textual explanation of the confusion matrix they used for the visual explanation shown in Figure 2.8 . This entailed describing the results on of each quadrant by contextualising with the selected demographic (race, or gender). From their user testing they found that both views (textual and confusion matrix) both improved the user's understanding of the algorithmic trade-offs, with no statistically difference between the two methods [4].


Explore the algorithm
Adjust how aggressive the algorithm should be in identifying defendants who might reoffend.
Identify more
Identify fewer
More who are identified might not reoffend
More who reoffend might not be identified
Explore effects on different groups:
Race (African / White American)
Gender (Female / Male)
The algorithm has inherent trade-offs between overall error and group-specific errors.
Errors
Disparity
Minimize differences between groups
Minimize Overall Error
VISUALIZATION VIEW
TEXT VIEW

Overall, among **3000** defendants, **1303 (43%)** are incorrectly predicted.

Among these **1303** incorrect predictions:

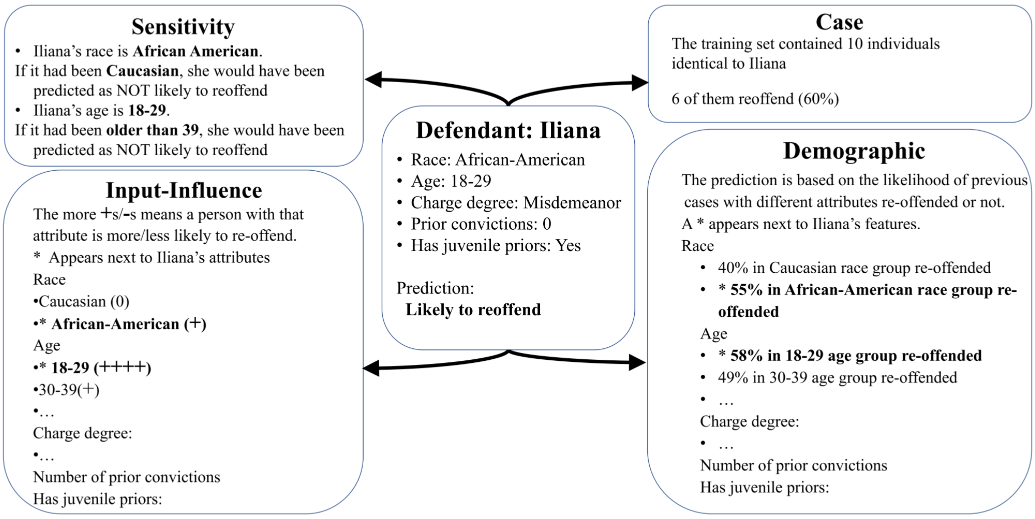
- **FALSE NEGATIVE: 101 White American** and **54 African American** defendants who are predicted NOT to reoffend will actually reoffend (Difference 47).
- **FALSE POSITIVE: 525 White American** and **623 African American** defendants who are predicted to reoffend will actually NOT reoffend (Difference 98).

Among these **1697** correct predictions:

- **TRUE NEGATIVE: 524 White American** and **265 African American** defendants who are predicted NOT to reoffend will actually NOT reoffend (Difference 259).
- **TRUE POSITIVE: 350 White American** and **558 African American** defendants who are predicted to reoffend will actually reoffend (Difference 208).

Figure 2.8: Interface showcasing algorithmic trade-off, with textual view shown in *Keeping designers in the loop: Communicating inherent algorithmic trade-offs across multiple objectives* by Bowen Yu et al. [4]

To improve on the contextualisation of the explanation of classification/decision models Jonathan Dodge et al. outlined two types of explanations, global and local. Global focusing on explaining the how the model works, through method such as feature importance, and demographic understanding. While local on the other-hand explains the reasoning for the classification, which can be achieved through counterfactual and case comparisons with other samples from the training set [7]. These explanation styles are shown in Figure 2.9 and represents how these explanation methods can contextualise the algorithm, it's data, and the classification.



Sensitivity

- Iliana's race is **African American**.

If it had been **Caucasian**, she would have been predicted as NOT likely to reoffend

- Iliana's age is **18-29**.

If it had been **older than 39**, she would have been predicted as NOT likely to reoffend

Case

The training set contained 10 individuals identical to Iliana

6 of them reoffend (60%)

Defendant: Iliana

- Race: African-American
- Age: 18-29
- Charge degree: Misdemeanor
- Prior convictions: 0
- Has juvenile priors: Yes

Prediction:
Likely to reoffend

Input-Influence

The more +s/-s means a person with that attribute is more/less likely to re-offend.
* Appears next to Iliana's attributes
Race
•Caucasian (0)
•*** African-American (+)**
Age
•*** 18-29 (++++)**
•30-39(+)
•…
Charge degree:
•…
Number of prior convictions
Has juvenile priors:

Demographic

The prediction is based on the likelihood of previous cases with different attributes re-offended or not.
A * appears next to Iliana's features.
Race

- 40% in Caucasian race group re-offended
- *** 55% in African-American race group re-offended**

Age

- *** 58% in 18-29 age group re-offended**
- 49% in 30-39 age group re-offended
- …

Charge degree:

- …

Number of prior convictions
Has juvenile priors:

Figure 2.9: Example of an textual explanation of the COMPAS model from *Explaining Models: An Empirical Study of How Explanations Impact Fairness Judgment* by Jonathan Dodge et al.[7]

2.5.3 The use of counterfactuals in XAI

Counterfactuals have also been a common theme in the research towards XAI. A counterfactual can be used to explain the results of classification, by explaining how if x feature was y you would have been classified differently [7] [19]. For example, Your loan would have been approved if your credit score was higher. With this, we can further explain the inner workings of these algorithms.

A counterfactual can be generated by changing feature values until a different classification is found [7]. However the majority of model and research is delving into discovering good counterfactuals which will add key insights into the classification. One method can be proximity which uses distance functions to determine how closely related the counterfactual is to the initial input, with the base hypothesis being that closely related values will be more useful. [19] [11]. For datasets with different feature types different distance functions will be needed, such as for continuous features distance function shown in equation 2.5 and categorical distance shown in equation 2.6. Where d_{cont} is the number of continuous variables, d_{cat} is the number of categorical variables, and MAD_p is the median absolute deviation of the p-th variable [11].

$$\text{dist\_cont}(c, x) = \frac{1}{d_{cont}} \sum_{p=1}^{d_{cont}} \frac{|c^p - x^p|}{MAD_p} \tag{2.5}$$

$$\text{dist\_cat}(c, x) = \frac{1}{d_{cat}} \sum_{p=1}^{d_{cat}} I(c^p \neq x^p) \tag{2.6}$$

Chapter 3: Project Approach

The COMPAS (Correctional Offender Management Profiling for Alternative Sanctions) algorithm is popularly used by judges and parole officers to determine a criminal defendant's likelihood of reoffending. ProPublica did a study into the algorithm, and by comparing the actual re-offending status of criminals two years later. They noticed that the algorithm was biased in favour of Caucasians, and biased against African Americans [13].

It is not within the project scope to generate and collect our own dataset, and therefore this report will use the ProPublica COMPAS dataset to generate the algorithm for the models and then evaluate its performance. The COMPAS dataset and algorithm has been shown to be an effective tool for explaining algorithms trade-offs, and has been used by numerous researchers in the prior research section [7] [8] [2] [9] [10] [4] [11] [12]. Therefore by using a proven algorithm, this project can focus on generating the models and the interface.

This dataset showcases the ethics and morality of the classification, by showcasing the greyness of the situation and the trade-offs. For example, do you prioritise maximising true positives rates (ie. predict defendant will re-offend)? Which could result in a higher rate of false positives (ie. the defendant is predicted to re-offend when they will not). You could desire a fair system that classifies races and genders equally however, this can be at the cost of lower accuracy and thus lead to more false accusations.

3.1 Models

3.1.1 Feature Importance

To aid in the understanding of the COMPAS algorithm feature importance was used. Based on the prior literary review a partial dependence plot can be calculated for each feature to determine their impact on a positive classification, by using the equation 2.3 [3]. Positive classification meaning the effect a feature has on the model for deciding they are a re-offender. With the partial dependence plot, we then can get the impact of a feature imputed by the user via the interface.

3.1.2 Trade-offs between False Positives and Negatives

To alter the number of false positives and negatives classification threshold tuning was used. First, the binary classification was converted into a probability classification and then a threshold was introduced to determine the required probability for a positive case. Thus by lowering the threshold, it will increase the number of positive predictions, and with that false positives errors. The threshold will be the interactive element for the user, which will allow them to experiment and balance the trade-offs.

For the COMPAS model, a false positive in this case would represent a person being labelled they will re-offend when in two years they didn't. The opposite being that a re-offender is wrongly classified as 'safe', and thus this trade-off can become an ethical decision for the user.

3.1.3 Trade-offs between Accuracy and Fairness

To represent fairness, the disparity is often calculated to represent the ‘unfairness’ of the system. The disparity can be calculated by getting the max differences of error types between two groups [4], shown in 3.1. In this case, the interface represented the fairness between races (Caucasian r_0 and African Americans r_1), by calculating the unfairness, which is the disparity between error types for each race. The fairness can then be calculated as the inverse of disparity.

$$\max(|FP(r_1) - FP(r_0)|, |FN(r_1) - FN(r_0)|) \tag{3.1}$$

In prior literature, the disparity was compared to total errors (or error rate), as the decrease in accuracy found was due to the increase in error [4][5]. Therefore to showcase the trade-offs between accuracy and fairness, the opposite was calculated, disparity and error rate. Both methods were attempted, during the implementation stage and concluded that using disparity and total errors worked in generating a Pareto curve.

To represent the trade-offs a Pareto optimal curve for errors and disparity was generated, in which one cannot be increased without the other decreasing. This curve was generated with a series of cost-sensitive analysis classification Q with the lowest empirical error subject to equation 3.2 [4], in which the values for θ will be altered by the user.

$$L(Q, \theta_0, \theta_1) = err(Q) + \theta_0(FP(r_1) - FP(r_0) + \theta_1(FN(r_1) - FN(r_0)) \tag{3.2}$$

3.2 Visualisation Techniques

There are two main goals for the visualizations. The first was to use feature importance to reduce the ‘black box’ element of the COMPAS algorithm, the second was to showcase algorithmic trade-offs. To represent the feature impact on a positive classification pie and line charts were used. While in prior research bar charts had been used to effectively explain the feature importance [2] [6]. There were too many features to cleanly display them a bar chart, and thus pie charts were used. As pie charts allow for direct percentage comparison between each feature. The line chart can track previously calculated feature importance, this will allow the user to compare the impact of changing the variable.

To showcase the trade-offs prior research had used performance metrics and particularly confusion matrices [5] [8] [4]. Which will allow the user to see the different types of errors and the overall performance of the model for different balancing parameters. For this interface a flow chart representation of a confusion matrix was used, as Hong Shen, et al. recorded a better understanding of performance with a flowchart, than the classic confusion matrix grid [10].

Chapter 4: **Implementation**

The schedule and plan for the implementation stage were drafted in the Semester 1 report and is in the Appendix 7.4. The implementation followed the schedule, however the final due date was extended, and thus the code development stage was prolonged. During the extension, the accuracy vs fairness trade-off model was further developed as it took more time than originally planned.

4.1 Data Pre-processing

The ProPublic COMPAS dataset included demographic details on the defendant, and details relating to their current charge, jail time, and prior charges. The dataset also included the original COMPAS algorithm prediction results (which is a score of reoffending likelihood) and the actual re-offender status 2 years later. Both results could be used as our class to train our model. However true results were used, as this project isn't evaluating the original COMPAS model, the project is using the model as an example for the moral effect these decision-making algorithms can have, and the ethical effect of their trade-offs. Also the true result data is more likely close to the original dataset used to train the original algorithm.

With the class decided, the COMPAS prediction was removed, along with any features which referenced date/time (i.e jail time, or length). Columns with a high percentage of null values were also removed (along with rows with any null values). Columns kept were easy to explain and understandable for the user, such as basic demographic information, prior charges counts, and current charge degree (such as a Misdemeanor or Felony charge). Lastly, the final dataset only includes inmates which were either Caucasian or African American. Which were up-sampled using SMOTE, to have equal data rows for both races. Up-sampling was done to reduce the bias towards race. The up-sampling was only done to the training set, therefore our dataset was split into 2 datasets, one for testing, the other for training.

The pre-processing resulted in 14 features (which was one hot encoded to 25 columns). With a training set of 4916 rows, and test test of 2050 entries. The full feature list is shown in the Table 4.1

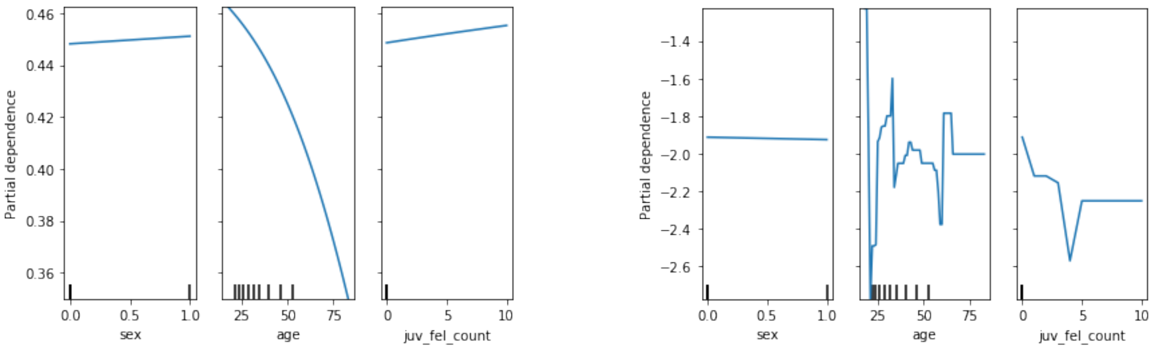
COMPAS Dataset Feature List		
Feature Name	Data Type	Description
Gender	Binary	Male or Female
Age	Integer	Age of the defendant
Age Category	Ordinal	3 Age categories: <25, 25-45, >45
Race	Binary	Caucasian or African American
Current Charge Degree	Binary	Current charged assign to defendant, Felony or Misdemeanor
is Re-offender	Binary	Prior re-offender status
is Violent Re-offender	Binary	Prior re-offender of an violent crime status
Prior Charge Degree	Categorical	Latest charge on the defendant, such as robbery. If they are not a re-offender it is set to none.
Juvenile Felony Count	Integer	Prior recorded juvenile felony charges.
Juvenile Misdemeanor Count	Integer	Prior recorded juvenile misdemeanor charges.
Juvenile Other Count	Integer	Any other prior recorded juvenile charges.
Prior Crimes Count	Integer	The number of all prior charges.
Two Year Re-offend	Binary	Class label, for if the inmate did re-offend 2 years after release.

Table 4.1: List of all features in the final dataset

4.2 Generating Models

4.2.1 Feature Importance

The python library sklearn exposes partial dependence methods. Which was used to generate the feature importance. In this stage different machine learning models were used to decide the best model to use. Logistic regression was chosen as it has probability classification functionality, and the partial dependence plots generated for each feature were consistent, as shown in Figure 4.1a. If you compare these results to the partial dependence plots generated using a Gradient Boosting algorithm (Figure 4.1b), it can be seen that the Logistic Regression plots are more consistent, and follow more intuitive logic.



(a) Logistic Regression

(b) Gradient Boosting

Figure 4.1: Partial Dependence Plots generated using Logistic Regression (a), and Gradient Boosting (b). Plots for 3 features, gender (sex), age, and juvenile felony count (juv_fel_count).

For example by looking at the feature juvenile felony count (juv_fel_count), intuitively you would assume that the feature importance would increase with higher counts. This happens for Logistic Regression, however with Gradient Boosting the partial dependence decreases, with a large dip at count 4. Due to this difference Logistic Regression was chosen as the model results are more predictable, and thus will be easier to understand, and interpret the feature impact.

The feature importance for a given future value can then be generated by interpolating the partial dependence plots.

4.2.2 Trade-offs: Error Types

The `predict_proba` method from the sklearn logistic regression model was used to get the probability estimates for a positive classification. The differences between false positives and false negatives were then tested at different thresholds, and at varying intervals. The results can be shown in Figure 4.2, which showcases the trade-offs between the two error types at different thresholds.

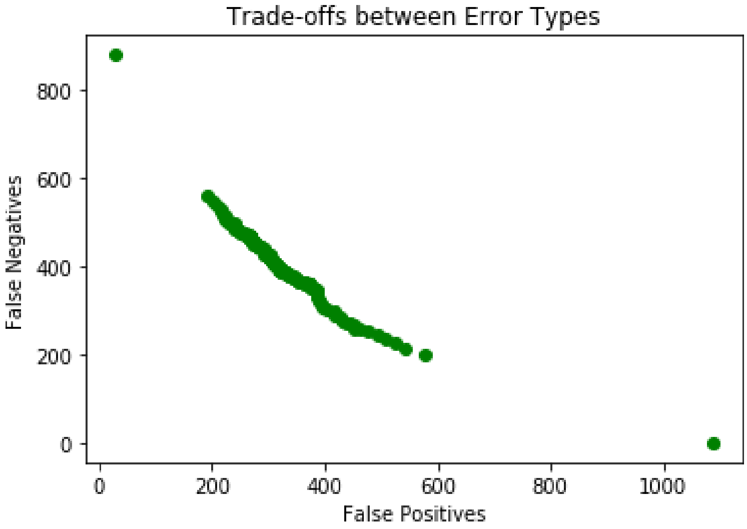


Figure 4.2: Trade-offs between False Positives and False negatives with increasing thresholds

4.2.3 Trade-offs: Accuracy vs Fairness

To find the theta values needed for our cost-sensitive classification grid search was used. To do this a custom machine learner model was implemented via the sklearn Base Estimator method. The model needed to calculate the weights for each row in the dataset. The weights depended on θ_0 and θ_1 and the race of the defendant in the sample. The weighting logic was based on Bowen Yu et al. trade-off model [4]. These weights were then used by a logistic regression model as sample weights. Four scorer functions were then implemented to determine the fairness, the disparity, the amount of errors, and the accuracy of the model.

From this, a grid search was performed on the custom model and the scoring functions. A Pareto optimal set was created by comparing and plotting disparity to errors (Figure 4.3). In Figure 4.3 you can observe that as the disparity (un-fairness) increases, the amount of errors decreases. The two theta values for the Pareto front were then stored, which will be used for the user interaction. The user will access generated theta values via an array, with the first entry having the lowest amount of disparity with the highest amount of errors.

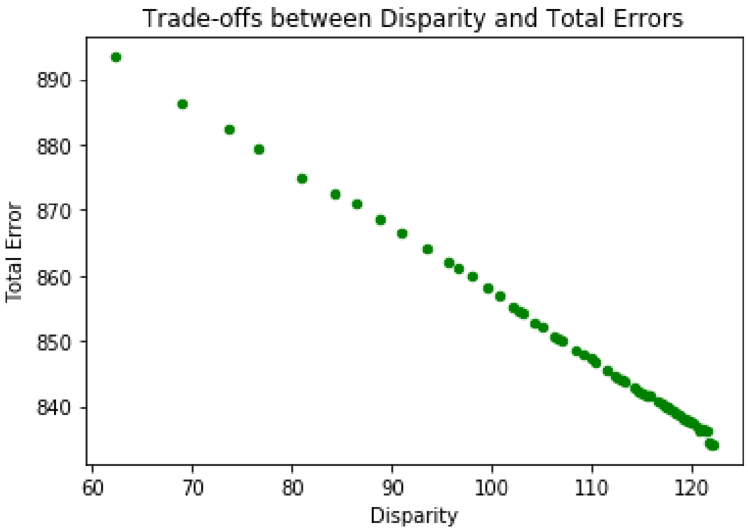


Figure 4.3: Pareto front for Disparity vs Total Error for different values of θ_0 and θ_1

4.3 Interface Generation

As the model makes use of python frameworks, Flask the python web framework was used to implement the model and the functionality for the interface. The interactive elements were created with JavaScript. The final interface is shown in Figure 4.4. From the figure, you can see that the interface is broken into two parts, the interactive element in the mode of a form and the performance charts/visualizations.

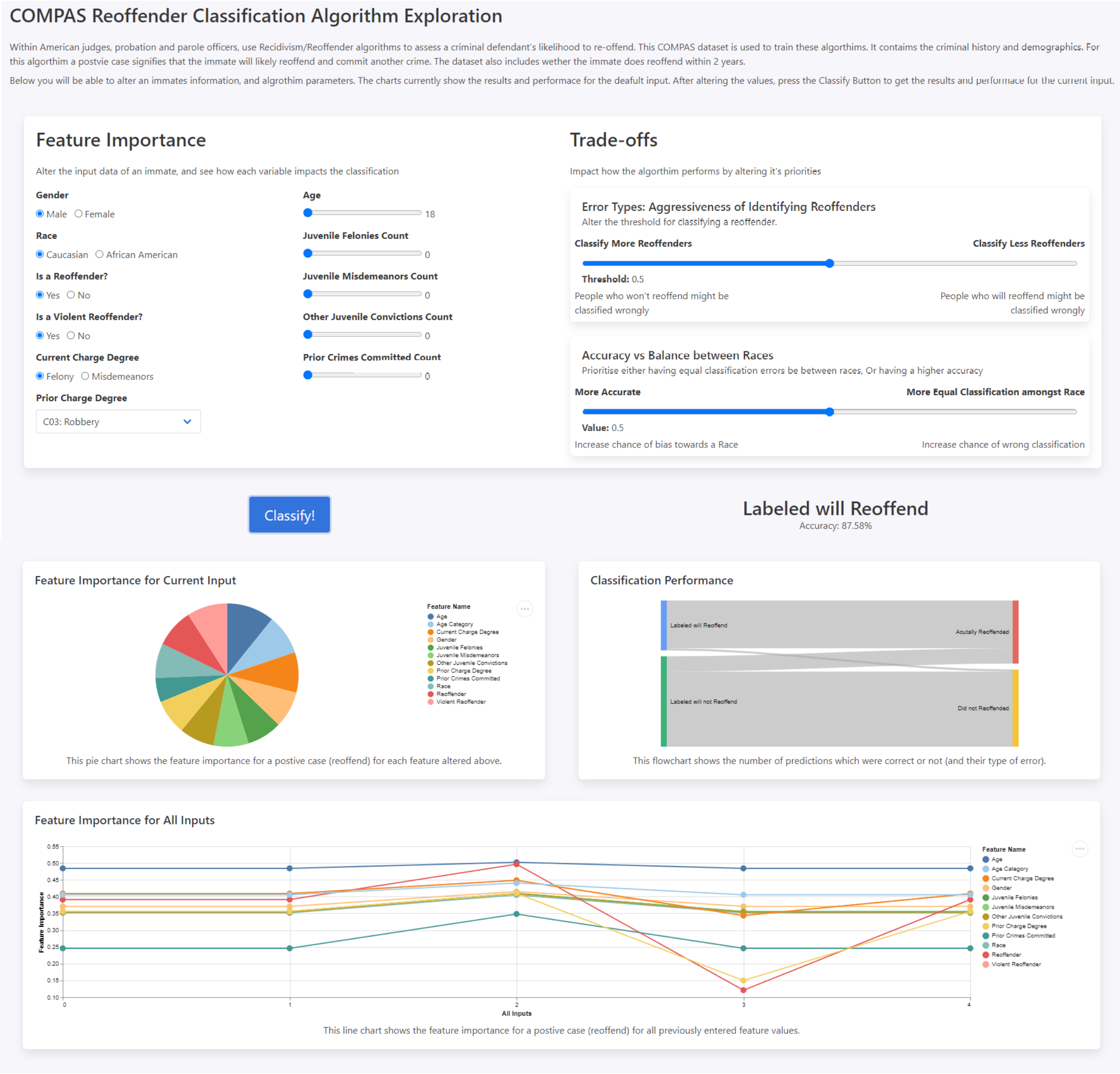
JavaScript processed the form elements, sent the inputted values to the python script via Flask, it was then processed by the model generated in the prior section and sent back. The returned information was then used to generate the charts for the interface. The interface also displayed the classification result for the features (i.e. Labeled will Reoffend or not) and the accuracy of the model. This allows the user to make comparisons of the results of the model, and its performance for different inputted values.

4.3.1 Interaction Form

The interaction is held in a form, shown in the top half of Figure 4.4. Which has two sections, the first allows the user to alter the details of a defendant. The model will then calculate the feature importance for the inputted details. Secondly, there is the trade-off section, which has two sliders. One to alter the threshold for the trade-offs between false positives and negatives. The other to balance between disparity and error types. Both sections include a description to aid in the understanding of the COMPAS algorithm and interface.

4.3.2 Visualisations

The visualisations include three charts (Pie, line, and flow chart), and are shown in the bottom half of Figure 4.4. To generate the pie chart and line chart Vega-Lite was used, as it can be easily integrated with JavaScript. Using this allowed for interactive charts, which had highlighting features and tool-tips. Vega-lite however does not support flow chart generation, therefore the web service Google Charts was used instead. Google charts had fewer customisation options, however it had tool-tips and highlighting features that kept the visuals consistent.



COMPAS Reoffender Classification Algorithm Exploration

Within American judges, probation and parole officers, use Recidivism/Reoffender algorithms to assess a criminal defendant's likelihood to re-offend. This COMPAS dataset is used to train these algorthims. It contaims the criminal history and demographics. For this algorthim a postvie case signifies that the immate will likely reoffend and commit another crime. The dataset also includes wether the immate does reoffend within 2 years.

Below you will be able to alter an immates information, and algrothim parameters. The charts currently show the results and performace for the deafult input. After altering the values, press the Classify Button to get the results and performace for the current input.

Feature Importance

Alter the input data of an immate, and see how each variable impacts the classification

Gender: ● Male ○ Female

Race: ● Caucasian ○ African American

Is a Reoffender?: ● Yes ○ No

Is a Violent Reoffender?: ● Yes ○ No

Current Charge Degree: ● Felony ○ Misdemeanors

Prior Charge Degree: C03: Robbery

Age: 18

Juvenile Felonies Count: 0

Juvenile Misdemeanors Count: 0

Other Juvenile Convictions Count: 0

Prior Crimes Committed Count: 0

Trade-offs

Impact how the algorthim performs by altering it's priorities

Error Types: Aggressiveness of Identifying Reoffenders

Alter the threshold for classifying a reoffender.

Classify More Reoffenders — Classify Less Reoffenders

Threshold: 0.5

People who won't reoffend might be classified wrongly — People who will reoffend might be classified wrongly

Accuracy vs Balance between Races

Prioritise either having equal classification errors be between races, Or having a higher accuracy

More Accurate — More Equal Classification amongst Race

Value: 0.5

Increase chance of bias towards a Race — Increase chance of wrong classification

Classify!

Labeled will Reoffend

Accuracy: 87.58%

Feature Importance for Current Input

This pie chart shows the feature importance for a postive case (reoffend) for each feature altered above.

Classification Performance

This flowchart shows the number of predictions which were correct or not (and their type of error).

Feature Importance for All Inputs

This line chart shows the feature importance for a postive case (reoffend) for all previously entered feature values.

Figure 4.4: The interface: The top half containing the form elements, with the features on the left, and the trade-off sliders on the right.
In the middle, there is the classify button on the left which will populate the charts. On the right, there is the classification results and accuracy for the inputted values.
The bottom half contains the charts for displaying the feature importance (pie and line chart), and the performance of the model (flow chart).

Chapter 5: Evaluation

The goal of the interface was to improve the understanding of decision-making algorithms and their trade-offs. This particular interface focused on explaining the COMPAS algorithm, and the ethical conflict of its trade-offs. Therefore this evaluation step focused on evaluating these metrics:

- **Interface Comprehension:**
  The interface was easy to use, and they understood what it was trying to convey. Previous knowledge on decision-making algorithms needed to be recorded as this can be compared to the improvements in the user's comprehension.
- **Feature Importance Comprehension:**
  If the interface improved their understanding of the impact of features in the COMPAS algorithm. With also a focus on evaluating the usefulness of the pie and line chart in aiding their understanding.
- **Trade-offs Comprehension:**
  If the interface improved the understanding of trade-offs in the COMPAS algorithm. With a focus on if the flowchart aided their understanding.
- **Ethics and Trust:**
  Wanted to evaluate how people approached the trade-offs and the ethics of the COMPAS algorithm, and if the interface aided in their decision for balancing the trade-offs. Interesting insight can also be gathered if the participant's trust in these decision-making algorithms had changed.

5.1 Questionnaire

Due to this report being constructed during COVID-19, only online evaluation methods were possible. Thus an online questionnaire was created to evaluate the metrics identified above. The questionnaire first involved an explanation of each of the interface features and the COMPAS algorithm.

The questionnaire then set tasks for the participants to achieve. Three tasks were outlined, the first was to allow them to see the impact of different features on the feature importance. The second was to see the impact of the Trade-off Error type threshold on both the feature importance and the performance of the model. Lastly, the participants were asked to find (in their opinion) the best values for both trade-offs. This allowed us to evaluate the different approaches to the trade-offs, and record what they choose, and why. The participants were asked questions evaluating the performance of elements of the interface, and how they impacted their understanding of the model. A copy of the questionnaire is shown in the Appendix 7.5.

5.2 Results

For the questionnaire 10 students from the UCD School of Computer Science were questioned. The questionnaire had two types of results, ordinal quantitative results, for example the use of Agree, Neutral, Disagree. Which was measured and compared quantitatively using charts (Figure 5.1). Secondly, there were open ending qualitative results, such as asking the participant to explain their reasoning. To compare these results, the responses were grouped into common themes, and a census was collected from the groups.

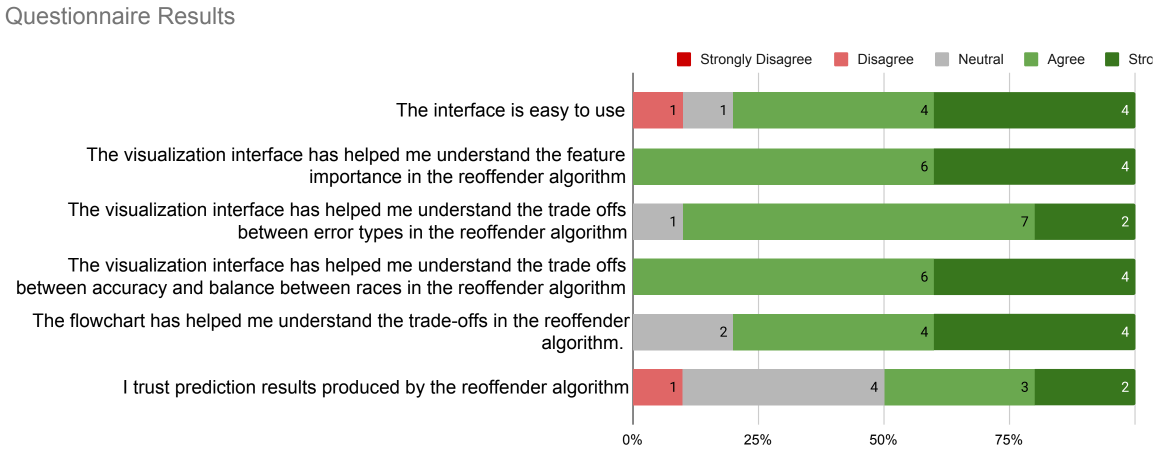


Figure 5.1: Questionnaire Results for the questions which used the ordinal scale of Strongly Disagree - Neural - Strongly Agree.

The participants had a wide range of prior familiarity with decision-making algorithms. With 90% of them recorded having an increased understanding of the COMPAS algorithm. This can suggest that the interface succeeding in improving the understanding of the algorithm. Two out of the three tasks the participants were asked to perform, had a correct answer, and for both of them 70% chose the correct answer. Thus representing that participants had a level of understanding of the interface and model. For the third task, the participants were asked to choose the best values of both trade-offs, and their reasoning for doing so. Interestingly there was a 40-60 split of the type of responses. The first group had more technical reasoning, focusing solely on the performance of the model, such as which values gave the best accuracy. The other proportion of the participants used ethical reasoning, with discussions on the issues of bias towards race, and the different types of errors.

By observing Figure 5.1, you can observe that interface performed well in improving the understanding of both trade-offs and feature importance With the Accuracy and Fairness performing slightly higher, this could be due to the notion of accuracy and fairness is easier to understand than error types.

To evaluate the performance of each chart (pie and line). The participants were asked whether both charts supported, or a single one, (or neither) aided their understanding of features importance. The results of this question are shown in Figure 5.2. 60% respond with only the line chart aided their understanding (with a 20% for only pie, and 20% for both). A follow-up question was to ask their reasoning for this response. The results suggest that there were too many features for the pie chart, and thus it was hard to compare them to each other via the pie chart. Therefore they preferred the line chart as it allows for easier comparison between a lot of features, and it included results from the previous classification.

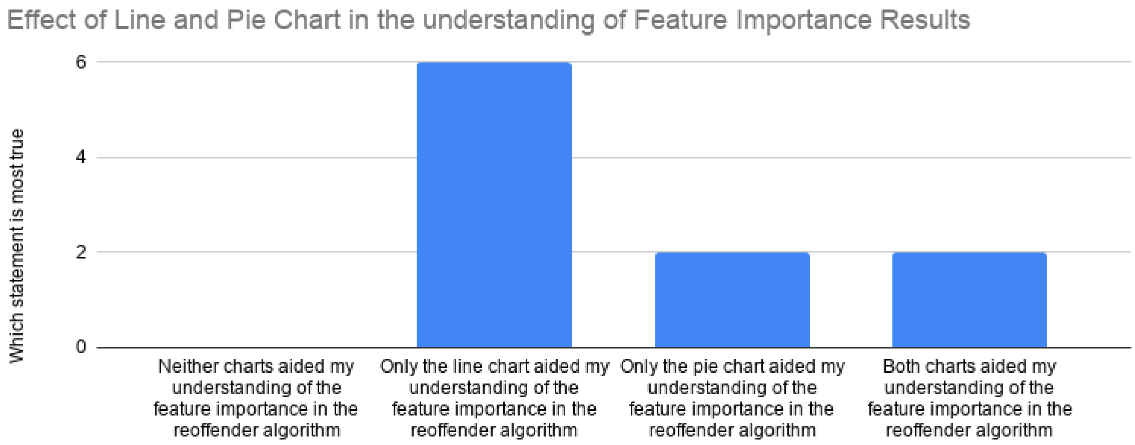


Figure 5.2: Questionnaire Results for the impact of each chart for understanding feature importance. The participants was asked to select the most true statement.

For evaluating trust, the participants were asked about their trust towards the COMPAS algorithm (Figure 5.1) and then asked if it had changed (and why). 40% resulted in no change, a majority of them had prior familiarity with decision-making algorithms, to begin with, and one mentioned that due to their prior knowledge they already understood the conflicts with decision-making algorithms.

However from the participants whose trust had changed, the responses suggest that trust was influenced by many factors. Such as they reported the increased understandings of the process improved their trust in the model. However the understanding of the trade-offs and it's racial basis had a negative effect on trust.

Overall from this questionnaire, the interface has aided the understanding of decision-making algorithms, by specifically explaining the impact of features and the trade-offs of the COMPAS algorithm. However from the questionnaire, the pie chart was ineffective at explaining feature importance, this could be solved by using a different visualization method (such as bar charts), or by including fewer features. Reducing the number of features could have made the interface clearer, and improved the effectiveness of the interface.

The questionnaire was very limited by the number of participants, and that it was also solely online. If an offline evaluation process was possible, the evaluation could have observed the participants, and achieve a Think Aloud method to provide useful insight into how the user interacts with the interface, and their learning process.

Chapter 6: Future Work and Conclusion

6.1 Future Work

As mentioned in the related work, textual explanations can be a useful tool for explaining feature importance and their trade-offs. Therefore adding textual explanations such as counterfactuals could greatly improve the current interface. Counterfactual could be added to the feature importance explanation as it can provide an alternate explanation to the feature impact on the classification. For example, a textual element explaining that the classification would have changed if the defendant was older.

Further extension is needed in evaluating the combination of both feature importance and trade-off explanation in explaining decision-making algorithms. A more controlled and intensive evaluation method with different versions of the interface. Such as only feature importance, and only trade-offs. If this was done we can compare the different scenarios and compare the impact of each method individually. We could also extend this to the usage of different charts. With this, we could have a more comprehensive evaluation method for visualisation which explain algorithms and their trade-offs.

6.2 Conclusion

In this report, different methods for explaining decision-making algorithms, and their trade-offs were researched and developed. Using this literary review an interactive interface was created to aid the understanding of the COMPAS model and its algorithmic trade-offs. To white-box, the model the interface explained how different values for features can impact the classification result. Two different types of trade-offs were showcased, the first being the trade-off between error types (False positives, and false negatives). Secondly showcasing the possible bias in the COMPAS algorithm, and how there is a trade-off between having a fair model or having an accurate model.

User testing through a questionnaire showcased that the interface improved the comprehension of feature importance and the algorithmic trade-offs. It allows the participants to explore the ethical impact these decision-making algorithms can have, and experiment with different features. Overall showcasing that the use of interaction and a combination of both explaining the AI with feature importance and their trade-offs allowed for a well-rounded educational platform, which enhanced the understanding of decision-making algorithms.

Bibliography

1. *Gavin Williamson Faces Backlash over A-Levels as Private Schools See Biggest Increase in Top Grades* www.independent.co.uk/news/uk/politics/level-results-private-school-state-gavin-williamson-grades-a9669571.html. accessed: 21.10.2020.
2. Cheng, H.-F. *et al.* Explaining Decision-Making Algorithms through UI, 1–12 (2019).
3. Krause, J., Perer, A. & Ng, K. Interacting with predictions: Visual inspection of black-box machine learning models. *Conference on Human Factors in Computing Systems - Proceedings,* 5686–5697 (2016).
4. Yu, B. *et al.* Keeping designers in the loop: Communicating inherent algorithmic trade-offs across multiple objectives. *DIS 2020 - Proceedings of the 2020 ACM Designing Interactive Systems Conference,* 1245–1257. arXiv: 1910.03061 (2020).
5. Zhang, Y., Liao, Q. V. & Bellamy, R. K. Effect of confidence and explanation on accuracy and trust calibration in ai-assisted decision making. *arXiv,* 295–305 (2020).
6. Muhammad, K., Lawlor, A. & Smyth, B. A live-user study of opinionated explanations for recommender systems. *International Conference on Intelligent User Interfaces, Proceedings IUI* **07-10-Marc,** 256–260 (2016).
7. Dodge, J., Vera Liao, Q., Zhang, Y., Bellamy, R. K. & Dugan, C. Explaining models: An empirical study of how explanations impact fairness judgment. *International Conference on Intelligent User Interfaces, Proceedings IUI* **Part F1476,** 275–285. arXiv: 1901.07694 (2019).
8. Agarwal, A., Beygelzimer, A., Dudfk, M., Langford, J. & Hanna, W. A reductions approach to fair classification. *35th International Conference on Machine Learning, ICML 2018* **1,** 102–119. arXiv: 1803.02453 (2018).
9. Lai, V., Liu, H. & Tan, C. "Why is 'Chicago' deceptive?" Towards Building Model-Driven Tutorials for Humans. *Conference on Human Factors in Computing Systems - Proceedings,* 1–13. arXiv: 2001.05871 (2020).
10. Shen, H. *et al.* Designing Alternative Representations of Confusion Matrices to Support Non-Expert Public Understanding of Algorithm Performance. *Proceedings of the ACM on Human-Computer Interaction* **4.** ISSN: 25730142 (2020).
11. Mothilal, R. K., Sharma, A. & Tan, C. Explaining Machine Learning Classifiers through Diverse Counterfactual Explanations. *arXiv,* 607–617 (2019).
12. Castelluccia, C. & Métayer, D. L. *Understanding algorithmic decision-making* **March,** 1–10. ISBN: 9789284635061. https://op.europa.eu/en/publication-detail/-/publication/ca808eed-90af-11e9-9369-01aa75ed71a1 (2019).
13. Larson, J., Mattu, S., Kirchner, L. & Angwin, J. *How we analyzed the COMPAS recidivism algorithm. ProPublica (May 2016)* https://www.propublica.org/article/how-we-analyzed-the-compas-recidivism-algorithm. accessed: 07.12.2020.
14. Ribeiro, M. T., Singh, S. & Guestrin, C. "Why should i trust you?" Explaining the predictions of any classifier. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* **13-17-Augu,** 1135–1144. arXiv: 1602.04938 (2016).
15. Wattenberg, M., Viégas, F. & Hardt, M. *Attacking discrimination with smarter machine learning* http://research.google.com/bigpicture/attacking-discrimination-in-ml/. accessed: 1.12.2020.
16. Kocielnik, R., Amershi, S. & Bennett, P. N. Will you accept an imperfect AI? Exploring Designs for Adjusting End-user Expectations of AI Systems. *Conference on Human Factors in Computing Systems - Proceedings,* 1–14 (2019).

17. Brownlee, J. *A Gentle Introduction to Threshold-Moving for Imbalanced Classification* https://machinelearningmastery.com/threshold-moving-for-imbalanced-classification/. accessed: 07.12.2020.

18. Brownlee, J. *How to Use ROC Curves and Precision-Recall Curves for Classification in Python* https://machinelearningmastery.com/threshold-moving-for-imbalanced-classification/. accessed: 07.12.2020.

19. Keane, M. T. & Smyth, B. Good Counterfactuals and Where to Find Them: A Case-Based Technique for Generating Counterfactuals for Explainable AI (XAI). *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **12311 LNAI,** 163–178. ISSN: 16113349. arXiv: 2005.13997 (2020).

20. Tanham, P. *"Leaving Cert Shows We Will Have to Learn to Live with Algorithms"* www.irishtimes.com/opinion/leaving-cert-shows-we-will-have-to-learn-to-live-with-algorithms-1.4349728. accessed: 21.10.2020.

21. Radford, A. *et al.* Language Models Are Unsupervised Multitask Learner.

22. Gibson, E. *et al.* Color naming across languages reflects color use. *Proceedings of the National Academy of Sciences* **114,** 10785–10790. ISSN: 0027-8424. eprint: https://www.pnas.org/content/114/40/10785.full.pdf. https://www.pnas.org/content/114/40/10785 (2017).

23. Denby, E. & Gammack, J. The naming of colours: investigating a psychological curiosity using AI. *ICONIP'99. ANZIIS'99 & ANNES'99 & ACNN'99. 6th International Conference on Neural Information Processing. Proceedings (Cat. No.99EX378)* **3,** 964–973 vol.3 (1999).

24. Mojsilovic, A. A computational model for color naming and describing color composition of images. *IEEE Transactions on Image Processing* **14,** 690–699 (2005).

25. Shane, J. *Paint Colors Designed by Neural Network, Part 2* aiweirdness.com/post/160985569682/paint-colors-designed-by-neural-network-part-2. accessed: 22.10.2020.

26. Waterhouse Coopers, P. *PSD2 In Ireland* www.pwc.ie/industries/banking/psd2-in-ireland.html. accessed: 22.10.2020.

27. Kim, E., Coumar, A., Lober, W. B. & Kim, Y. Addressing Mental Health Epidemic Among University Students via Web-based, Self-Screening, and Referral System: A Preliminary Study. *IEEE Transactions on Information Technology in Biomedicine* **15,** 301–307 (2011).

28. Sturgeon, J. A. *et al.* The Psychosocial Context of Financial Stress. *Psychosomatic Medicine* **18** (2016).

29. Mind. *Money and Mental Health.* ww.mind.org.uk/information-support/tips-for-everyday-living/money-and-mental-health/money-and-mental-health/. accessed: 22.10.2020.

30. Of Ireland, B. *Getting Started with Bank of Ireland APIs* eu1.anypoint.mulesoft.com/exchange/portals/bankofireland/pages/Getting%20Started/. accessed: 22.10.2020.

31. Portal, A. I. B. D. *Getting Started* developer.aib.ie/getting-started-ROI. accessed: 22.10.2020.

32. Plaid. *Pricing* plaid.com/eu/pricing/. accessed: 22.10.2020.

Chapter 7: Appendix

7.1 Project Proposal 1: Visualising algorithmic trade-offs

7.1.1 Problem Statement

The aim of this project is to develop an interface to help the understanding of decision-making algorithms and their trade-offs for non-computer scientists. To achieve this I will develop two interactive interfaces, the first will allow users to alter the input variables to change the result of the classification model. I also aim to implement a stacked bar chart to display how each variable affects the classification model and its feature importance.

Once the users are familiar with the algorithm, I want to showcase two trade-offs which can come from decision-making algorithms. To show the importance of understanding these trade-offs I will be using the COMPAS Recidivism Algorithm, which is used in the US to determine a criminal's likelihood of re-offending. With this, the user will be able to choose to prioritise lowering false positives or false negatives (i.e determining a person will re-offend incorrectly). The user will also be able to balance between fairness and accuracy. Fairness meaning that either racial or gender groups are equally classified, while a low accuracy will incorrectly classify people of recidivism.

I will end the project by testing the interface on a common group of users through the use of online questionnaires and interviews. For each test-user, I will evaluate their knowledge and perceived (self-reported) knowledge on decision-making algorithms and their trade-offs.

An additional goal for this project would be to include a textual explanation via counterfactuals. A counterfactual can be used to explain the results of classification, by explaining how if x feature was y you would have been classified differently. For example, your loan would have been approved if your credit score was higher. With this, we can further explain the inner workings of these algorithms. We could also showcase the bias/unfairness in the COMPAS system, by having this person would have been classified as high risk of recidivism if they were African American.

7.1.2 Background

Algorithmic bias is becoming increasingly a major issue as algorithms are being used to decide increasingly important decisions. For example, there was major error and controversy for this year 2020 A-Levels and Leaving Cert grades, in which an algorithm decided the students grade instead of the usual examination. Both had errors which resulted in wrongful grading. Leaving Cert had an error in their coding, in that the algorithm used the wrong data and thus caused a bias towards that data [20]. The A-Levels algorithm had a major bias in that it rewarded students in private schools better than public schools [1]. This is an example of algorithm bias having a major effect on major decisions. Due to this, it is ever important for the general public to understand and know about the trade-offs of these algorithms.

7.1.3 Related work

Hao-Fei Cheng et al. investigated the strategies for explaining decision-making algorithms to non-experts. They explored 2 sets of strategies: a black-box versus a white-box explanation approach, and static versus an interactive interface approach [2]. White-box means that they explain the internal workings on the model, while black-box will only show the output. They designed and user-tested both strategies, for the white-box design they had a horizontal stacked bar chart, with the length and colour of the section representing the variable and its influence in the algorithm, while black-box only showed the classification result [2]. I felt however that the horizontal stacked bar chart is a bit confusing at first. Furthermore, the stack bar chart means it is harder to compare the individual variables as they are on a different axis. Therefore I feel it would be suitable to have the option to change the chart to a bar chart, which allows the user to compare each variable on a common axis. For interaction they allowed the user to change the values for each variable, from this they could see the effect on the classification [2].

For their user testing, they asked for self-reported understanding and tested their knowledge before and after using the interface. They found that white-box interfaces increased their understanding of the algorithm. User who were tested with black-box method however, had higher self-reported understanding than white-box method. They suggested that this was due to the white-box option over-loading the user with information. For the interface, they found that interactive users not only scored higher in their algorithm understanding they also had a higher level of self-reported understanding of the algorithm[3]. This shows the importance of interactivity for interfaces. For testing the effectiveness of my interface, it will be important to do similar before and after questioning.

Bowen Yu et al. similarly wanted to explain algorithms to designers. However, their focus was to showcase the inherent algorithmic trade-offs. They also designed an interactive interface in which they allowed the user to balance between two trade-offs shown in the COMPAS Recidivism Algorithm [4]. With this, they communicated the trade-offs between false positives and false negatives. They also wanted to show the trade-off between overall errors and disparity. These were shown by the use of a visual and textual confusion matrix, and by comparing the number of classification of the race (African / White American) or gender (Male/Female) [4].

Similarly to Hao-Fei Cheng et al. they also tested their interface, with the focus being on the understanding of trade-offs and their trust in these systems. They recorded that both the text and matrix method greatly improved the understanding of trade-offs. Also, 50% of the users changed their trust levels, with approximately half of these users trusting the algorithm more, while the others decreased their trust in these systems [4].

With this project, I hope to combine both of these explanation methods and provide a well-rounded explanation of decision making classifiers. By first explaining the algorithm, then delving into the trade-offs. Hopefully, with this combination, the user will be able to understand the trade-offs in these algorithms and how they can happen.

7.1.4 Datasets and Resources required

The focus of this project is not the collection of data or the development of a decision making algorithm. The goal is to help the understanding of decision-making algorithms and their trade-offs. Therefore for this project, I will be using the popular dataset and algorithm COMPAS, which has been proven and tested to be a good representation of the algorithmic trade-offs [7] [8] [2] [9] [10] [4] [11] [12]. The recidivism algorithm showcases the ethics and morality of these decisions, by showcasing how they 'greyness' of the situation and trade-offs. For example, do you prioritise maximising TP rates (ie. predict defendant will reoffend) which could result in a higher rate of FP (ie. defendant is predicted to reoffend when they will not). You would desire a fair system which classifies races and genders equally - however, this can be at the cost of lower accuracy and thus lead to more false accusations.

7.1.5 Bibliography

In this section I will outline the key research papers I will be using for my report. As shown in the related work we have two key papers.*"Explaining Decision-Making Algorithms through UI: Strategies to Help Non-Expert Stakeholders"* written by Hao-Fei Cheng et al. is a key research paper for the effective methods of explaining algorithms. For explaining the algorithmic trade-offs our second key paper is *"Keeping Designers in the Loop: Communicating Inherent Algorithmic Trade-offs Across Multiple Objectives"* by Bowen Yu et al. To assist in the feature importance for explain algorithms we can use *"A Live-User Study of Opinionated Explanations for Recommender Systems"* by Khalil Muhammad et. al and *"Interacting with Predictions: Visual Inspection of Black-box Machine Learning Models"* by Josua Krause et al. Lastly for balancing fairness in our trade-off models we can use the models shown by Alekh Agarwa et al. in *"A Reductions Approach to Fair Classification"*

7.1.6 Personal contribution

I have a keen interest in information visualization. With this project, I aim to explore and experiment with educational visuals to help educate the general public, with the main focus to have clean, concise visuals for optimal visual understanding. In an age of misinformation, it's ever important to teach and educate people in a simple and clear visual manner. I hope to use a combination of JavaScript and Vega-Lite to make interactive charts for the front-end, with Python being used for the functionality of the back-end.

7.2 Project Proposal 2: Name That Colour (and Colour That Name)

Problem Statement:
The aim of this project is to create a system which can name colours and create a colour from a name. The main plan will be to use a web scraping tool (such as Beautiful Soup) to gather the colours and their names uploaded to ColourLovers.com. Once cleaned this data can be used to train an unsupervised transformer (such as GPT-2). Which will be able to connect the colour code sequence to words, by assigning a probability to any Unicode string, such that it will predict the most likely name (or colour) based on the input.

I also hope to analyse the naming of the colours itself, the psychology behind a name can suggest the preferred use or emotion of colour. To achieve this I hope to classify each colour into the major colour groups (i.e Red, Pink, light blue, dark green) by using classifications algorithms. From this, we can see and analyse the style of names in these groups and possibly speculate the reasoning.

A final task if all goes well is to make use of ColourLovers.com palette (a collection of colours) section. From these, we can use the same techniques for a single colour and have the system learn to create and name a palette of colours.

Background:
GPT-2 is a very powerful and large transformer-based language model, which simply stated, predicts the next word. OpenAI used and tested the system for reading comprehension, summation and translation [21]. I hope to use this model to choose the most probable name based on the imputed colour, and vice versa.

There can be many psychological reasons for the naming of colours. Edward Gibson and et al. from Massachusetts Institute of Technology hypothesised that objects are typically warm-coloured, while backgrounds are typically cool-coloured. Furthermore, the naming of the colours can be affected by how different cultures place more importance on specific colours [22]. Due to this study, I hope to investigate common naming patterns for colours and see that if cool colours are actually given background style names.

Related Work:
Analysing the psychological reasoning behind colour naming using AI has been done by the School of Information Technology in Murdoch University. Their focus was to understand how colours are perceived and interpreted to help aid colours machine classifiers. They found that hue or different saturations of the colour can affect identification and the naming of colours, and that classification into blue and green is difficult due to their ambiguity. In summary, they used an NCS system to provide consistent and less confusing names for colours [23]. In contrast, this project hopes to achieve the opposite effect. I hope this system will create the most 'human' like names, and thus potentially be confusing.

A majority of the journals surrounding the naming of colours are focused on image detection, and the identification of colours of objects in images. For example, Aleksandra Mojsilović used a computation model for colour naming and describing the colour composition of images. Their system identified and named the colour for each pixel, and then used histogram style bucketing to generate a description of colour composition for the image [24]. Their selection and identification of key colours can be used to improve my grouping of colour classification or can assist the naming of colour palettes by bucketing all the possible names for each colour in the palette to achieve a summation palette name.

This system of colour naming has been done before, and there are many Twitter bots (@AwesomeColorBot, @DrunkCircuit, @huehuebot) which all tweet a name for a colour generated by another Twitter bot (@everycolorbot). Janelle Shane also has created a colour and colour naming system using neural networks. With her work, she described that by changing the colour representation from RGB to HSV had an effect on the output of her system [25]. Due to this change, during the development stage, I would like to experiment with the different colour representation HEX, RGB and HSV to see if there is a change in the names or colours generated.

Datasets and Resources required:
Web scrapers (such as Beautiful Soup) can be used to collect the names and colour combinations from ColourLovers.com. There is also an RSS feed option if we want to update our system with more data. Currently, there are over 10 million colour entries, therefore I feel that this won't be necessary. There is no strict naming convention implemented by ColourLovers.com, which means that we have colours with symbols for names. Therefore we would need to do some heavy cleaning to the dataset to remove any non-alphabetic names. There is also the possibility for the same colour to be named differently. Therefore to provide a consistent dataset we will have to decide on a method to deal with these cases, such as randomly choosing between the duplicates.

Personal Contribution:
There are many systems which already name colours, however, they only achieve a one-way colour to name system, with this the base goal is to have the system have a two-way system of naming the colour, and creating the colour from a name.

To provide some more insights into why colours are named. I hope to examine the names themselves and possibly identify any consistent naming conventions of colour groups. This can be achieved by classifying each colour into the main colour group and analysing the similarities in the names for that group.

My last goal for this project is to fully push the colour creation and naming process and see if the system can create pleasing palettes and name groupings of colours. This will be achieved by using the palette section of ColourLovers.com data as input for our transformation architecture. The collection and cleaning will use the same method, however it will prove to be a more difficult challenge as we will need to cross-modal map a collection of colours to a name.

7.3 Project Proposal 3: Money & Mental Wellbeing

Problem Statement:
The aim of this project is to investigate and research the relationship between finance and mental health. To do this I will create an application that allows users to assign an emotion for each transaction made. I will be able to obtain the transactions using OpenBanking. By developing this application, the aim is for the users to be able to see and track their mental health in accordance with their financial transactions.

For each transaction, the application will ask the user to simply assign an emotion level. I aim to display a weekly summary of their transactions with their associated emotion. To improve mental health tracking, I want to add a daily journal feature, in which the user will assign an emotional level to their current mood, for example by asking: 'How was your day?'. By combining these methods, we can get more insight into not only how finance affects mental health, but how mental health can affect transactions and spending habits.

A final goal would be to further the summary and have the app identify trends and repeated moods. With the end goal being that the app will notify the user on a specific trend. For example, 'this type of transaction tends to bring you more joy', or 'low mood could be the cause of your high retail transactions'.

Background:
The goal for OpenBanking is for all major banks to release their data in a secure, standardised form. PSD2 requires banks to open their payments infrastructure and customer data assets, by the use of APIs [26]. Due to this third party applications and organisations can access and use users' banking details in a highly secure system. With this, I can develop an application which can securely access the transaction details for any user (with their consent).

Poor mental health with students is a very prevalent matter. With "more than 40% of U.S. students become depressed during their four years in college" [27]. Due to this, it is ever more important for students to identify and deal with their mental health. With this application, I hope to assist in an identification for students mental health triggers, which could possibly lead to overall better mental health.

Related Work:
One study into mental health and finance was done and they recorded that financial stress can also have a physical and psychological effect. They showed that finances can be the cause of psychological distress and increase difficulties in interpersonal communication. It can also lead to poorer physical health, due to stress activating pro-inflammatory cytokines, which increases the synthesis of inflammatory markers like C-reactive protein, this protein is found in blood plasma

that has been implicated in poor cardiac health[28].

However, their research sample only consisted of middle-aged adults, which means there is a lack of studies on students specifically. However, in their report they did suggest that financial stress could limit exposure to important resources for development, and can reduce protective psychological factors [28]. This implies that financial hardship can increase the chance for other stressors to take effect. Due to the suggested harm on development is it ever more important for there to be studies and support for students to deal with their financial situation and stress.

The connection between finance and mental health has been identified by mental health specialists. Mind is a mental health charity in the UK, they have a specific support network based on finance. They suggest that finance and low mood is a cycle, in such that they affect each other, such that finance can affect mental health, and poor mental health worsening financial management. One of their suggestions for a better mentality with finance is to understand your finance, and your mood patterns [29]. By extending the app to daily mood tracking I hope it will become easier for the user to track their moods and how it affects their finances.

Datasets and Resources required:
To gather the transactions of the user, I will need to make use of the OpenBanking options provided by each bank. To gain access to the major banks in Ireland I will need an eIDAS QWAC certificate, and register with The Central Bank of Ireland[30]. However, AIB has a development portal, which will grant me access to their test environment[31]. If getting the proper certifications is not available to me I can use PLAID. PLAID is a service which provides an API for open banking access. They have a free pricing section, which allows you to build and test using 100 live items[32]. Using either PLAID or AIB developers environment I will be able to make at least a prototype for the application and transaction acquisition algorithm.

From the API we are able to see the transaction amount, who the transaction goes too, and what sector of the transaction it belongs to (such as food, or retail) With this we should be able to group each transaction and assign the overall mood to each group. For example, rent could cause low mood, while retail could bring the opposite.

Personal Contribution:
By extending the application to allow daily tracking of moods, it will provide a better insight into the relationship between mental health and finance. There are many journaling or mood tracking apps available, and I hope to follow their simple mood tracking system. Each day they will simply ask you to rate your mood based on a scale. I'm hoping to use the same mood scale for the transaction for daily recordings. This means that along with your transactions mood trends, we can see the moods before and after spending.

To extend upon the insights my end goal would be to have the app identify common trends in the user's mood and transaction history and notify the user. With this, I hope it will help users and students identify their spending patterns, and how mood affects or be affected by their financial situation.

7.4 Project Work-plan

Below in Figure 6.1 is a Gantt Chart representation of the project approach and plan. A majority of the project will be code development which I have highlighted in cyan. I have divided the development stage into sub tasks to represent and plan my development process. This plan covers the development during the college term.

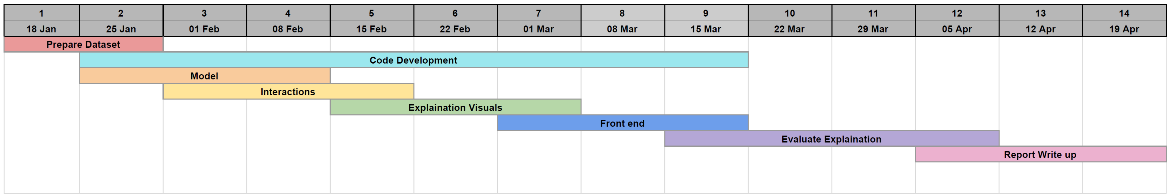


Figure 7.1: Project Plan Gantt Chart

7.4.1 Prepare Dataset

Data collection and preprocessing is not the goal for this project, and thus I have chosen to use the COMPAS dataset which is a recidivism algorithm used in the US jailing system. ProPublic in 2016 have gathered, cleaned this dataset to analyse the performance of COMPAS [13]. Due to this the dataset preparation step will only need to include extracting features and rows which are important to the explanation of the algorithm and the trade-offs. For example filtering the dataset for only African-Americans and Caucasians.

7.4.2 Code Development

The code development stage spans the majority of my project and will focus on the development of my explanations. With the base goal of implementing 2 interactive explanations, one for explaining the algorithm itself, and the other for algorithmic trade-offs. If I am ahead of schedule (i.e complete the front-end before week 8) I will also implement the extended goal of having textual explanations with counterfactuals.

Model

This stage will focus on implementing and testing the 3 models needed for this project, feature importance for explanation, trade-offs in FP and FN model and the balance between fairness and accuracy model. The models will be written in Python, as it has many packages such as sklearn which will assist in my ML algorithmic generation

Interactions

This stage is for the testing and implementing the changeable threshold/preferences used in the 3 models. These thresholds will be used by the user to interact with the models. They will need to be tested to check they are performing correctly. This stage will work closely with the Model stage, which is why they overlap heavily.

Explanation Visuals

Once the model and interactions are complete, I can focus on the outputs of the models. For this stage I will need to develop the charts for feature importance. For explaining the outputs for the trade-off models I will use flowcharts.

Front-end

The previous stages focused on the ‘back-end’, the functionality of the project. In this stage I will develop the website which will hold the explanation and trade-off interactions. Which can be done using HMTL, JavaScript and Vega-Lite for the charts.

7.4.3 Evaluate Explanations

To determine the effectiveness of my explanation models and interface I will need to evaluate the understanding of decision making algorithms and their trade-offs before and after using my project. This will involve me using an online platform to interview and question users on the performance of my models for explanation. To ascertain if they have an improved understanding, which methods they preferred, and lastly to see if the trust in the models has changed. This stage will involve the development and the distribution of the questionnaire, and then the consolidation of the results. This stage might start earlier in the term (i.e February), if ethical approval is needed.

7.4.4 Report Write-up

For the final weeks of the term I will focus on the write up of the report. To improve and quicken this process I plan to do a write up after each stage shown above

7.5 Questionnaire

FYP: Visualizing Decision Making Algorithms and their Trade-offs.

This project and research is being conducted by Susannah D'Arcy, an undergraduate student in School of Computer Science at the University College of Dublin under the supervision of Dr. David Coyle.

Our goal was to design and develop an interactive visualization interface to help the general public understand decision making algorithms, and their trade-offs. We would like to invite you to take part in this online survey which will guide your interactions with the interface, and ask about your experience after using it.

Before you decide whether to take part it is important that you understand why the research is being done and what it will involve. Please take time to read this information sheet carefully.

What is this research about?
This research is focused on creating and designing visualizations which help improve peoples understanding of decision making algorithms and their trade-offs.

Why are we doing this research?
Algorithms as being used to make increasing important decisions, and thus their is an increasing need to improve the general publics understanding of these algorithms and their possible trade-offs. By evaluating my interface, we will be able to see the impact of the visualizations in helping people's understandings.

Why have you been invited to take part?
You have been invited to take part as you are part of the School of Computer Science at the University College of Dublin. Your opinions and views will be very important in helping us understand how these visualizations affect the understanding of decision making algorithms.

How will your data be used?
The results of the study will be used to evaluate the interface, and will be presented with my final year project report. The report itself may be published publicly. This process is integral to evaluation and research process. It won't be possible for any individual to be identified from these results. This data is completely anonymized.

What will happen if you decide to take part in this research study?
If you wish to take part in the survey, you can click onto the next section. By completing the survey, you agree to take part in the research. The online survey will take approximately 10 minutes to complete.

How will your privacy be protected?
This survey is anonymous in that we do not ask for any personal information such as your name or any identifying/demographic information. Your responses will be confidential, and we do not collect identifying information such as your name, email address or IP address.

What are the benefits of taking part in this research study?
The information retrieved from this survey will be used to evaluate the interface, and could be used to inspire future research/development projects into explaining algorithms and their trade-offs.

What are the risks of taking part in this research study?
There are no significant risks associated with taking part in this study.

Can you change your mind at any stage and withdraw from the study?
Participation in this research is voluntary. If you don't wish to take part, you don't have to. If you do consent to participate, you can withdraw from the survey at any time while you are answering the questions, however, once you have finished and submitted your answers, your data will be combined with that of other participants and will therefore no longer be identifiable. Accordingly, once your data have been submitted you will not be able to withdraw your participation.

How will you find out what happens with this project and contact details for further information?
If you would like to be kept informed about the survey and the project, please contact Susannah D'Arcy at susannah.darcy@ucdconnect.ie

*Required

1. Consent *

If you do NOT wish to take part in the study you can close the survey now. If you do wish to take part tick all of the boxes below and tick "I consent to take part in the study" and then click next.

Tick all that apply.

□ I confirm that I am 18 years or older

□ I have read and understood the information sheet

□ I understand that participation is voluntary, and I can withdraw whilst taking the survey. I accept that once the survey is completed, I cannot withdraw my data due to the anonymous nature of the survey.

□ I agree my anonymised research data will be stored securely by the researcher. After the completion of the project the information will be destroyed securely.

□ I consent that my anonymised data may be quoted in dissemination activities including but not limited to scientific publication.

□ I am aware that I can contact the researcher at any point to get further information or clarification about this research study.

□ I am aware that when the research study is complete, findings or results from this research study will be available from the researcher.

□ I consent to take part in the study

Visualizing Decision Making Algorithms and their Trade-offs Dash Board

Please go to https://fypalgotradeoffswebsite.herokuapp.com/
(Might be slow loading it up initially please be patient)

This dash board uses the COMPAS reoffender database which predicts whether or not an inmate will reoffend (recommit a crime after release). It does this by using their demographic information and previous crimes. This algorithm positive case represents that the inputted inmate will reoffend. These types of algorithms are used in the US by judges and parole officers to aid in their decisions.

Please use and get familiar with the interface, and once you are ready head to the next section where I will set you task and ask you questions.

Below is some more details and information about the interface features.

Feature Importance

On the top left we can see the Feature importance section. Here you can change each of the variables associated with a imamate. The feature importance represents the impact of that variable in a positive classification. i.e. How much that variable affected the algorithms decision in the imamate being a reoffender

Below this section you will see a blue button labeled 'Classify'. Once clicked it will calculate the feature importance for a positive classification for each of the variables.

Feature importance for the latest classify will be shown in the pie chart on the left and all feature results for all classify clicks will be shown on the bottom of the website, via a line chart. Feature importance represents the impact/significant for a positive (reoffend) result.

Algorithm Trade-offs

On the top right we can see the Error Types: Aggressiveness of Identifying Reoffenders section. Here you can change the threshold for classify a reoffender. Lowering the threshold will increase the chance of a imamate being classified as a reoffender, while increasing the threshold will make it harder.

Changes to this threshold will have an affect on the feature importance, and on the performance of the algorithm.

Below this we can see Accuracy vs Balance between Races section. Here you change what the algorithm prioritises. We can focus on accuracy, which will lower the overall errors the algorithm makes. Or you can focus on having a balance of classification errors between races (i.e Caucasian and African American). By balancing these errors we hope to reduce a bias COMPAS algorithms tend to have due to the higher cases of African American within the dataset.

The performance is shown through a flow chart. Which will show if the algorithms predicted label was correct in a real test case.
(i.e. if the predicted reoffender actually reoffend 2 years later)

Feature Importance

https://fypalgotradeoffswebsite.herokuapp.com/

On the top left we can see the Feature importance section. Here you can change each of the variables associated with a imamate. The feature importance represents the impact of that variable in a positive classification. i.e. How much that variable affected the algorithms decision in the imamate being a reoffender

Below this section you will see a blue button labeled 'Classify'. Once clicked it will calculate the feature importance for a positive classification for each of the variables.

Feature importance for the latest classify will be shown in the pie chart on the left and all feature results for all classify clicks will be shown on the bottom of the website, via a line chart. Feature importance represents the impact/significant for a positive (reoffend) result.

First we will compare the affect of 2 settings on the Feature Importance.

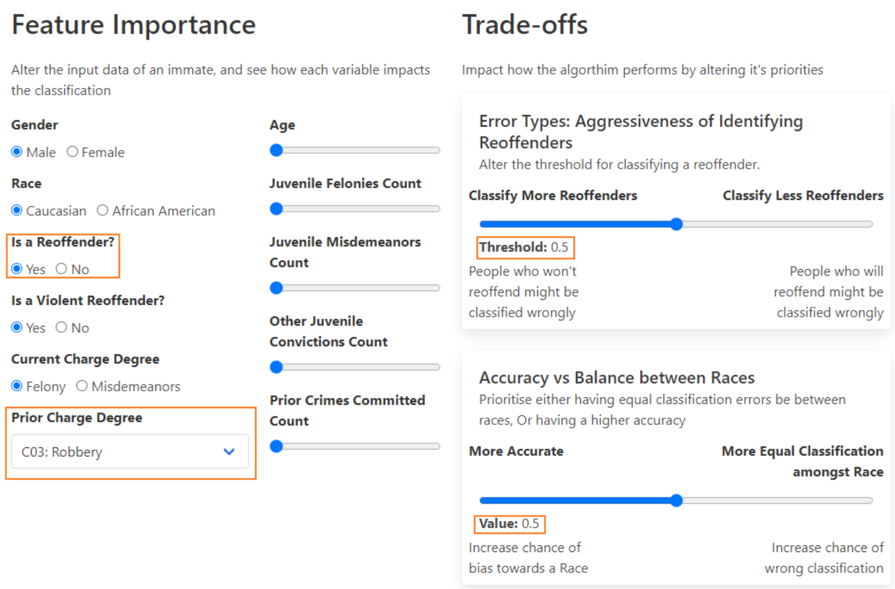
Setting 1: Is A Reoffender? - Yes, and Prior Charge Degree - C03: Robbery.

Please set the answer to Is A Reoffender? to No.
And set Prior Charge Degree to C03: Robbery.

Also have both Trade off values be 0.5. Such that:
- 'Error Types: Aggressiveness of Identifying Reoffenders' threshold = 0.5.
- 'Accuracy vs Balance between Races' Value = 0.5

Then click classify and see the affect on the feature importance via the pie chart, and/or line chart.

Setting 1



Feature Importance

Alter the input data of an immate, and see how each variable impacts the classification

Gender: ● Male ○ Female

Race: ● Caucasian ○ African American

Is a Reoffender?: ● Yes ○ No

Is a Violent Reoffender?: ● Yes ○ No

Current Charge Degree: ● Felony ○ Misdemeanors

Prior Charge Degree: C03: Robbery

Age
Juvenile Felonies Count
Juvenile Misdemeanors Count
Other Juvenile Convictions Count
Prior Crimes Committed Count

Trade-offs

Impact how the algorthim performs by altering it's priorities

Error Types: Aggressiveness of Identifying Reoffenders

Alter the threshold for classifying a reoffender.

Classify More Reoffenders — Classify Less Reoffenders

Threshold: 0.5

People who won't reoffend might be classified wrongly — People who will reoffend might be classified wrongly

Accuracy vs Balance between Races

Prioritise either having equal classification errors be between races, Or having a higher accuracy

More Accurate — More Equal Classification amongst Race

Value: 0.5

Increase chance of bias towards a Race — Increase chance of wrong classification

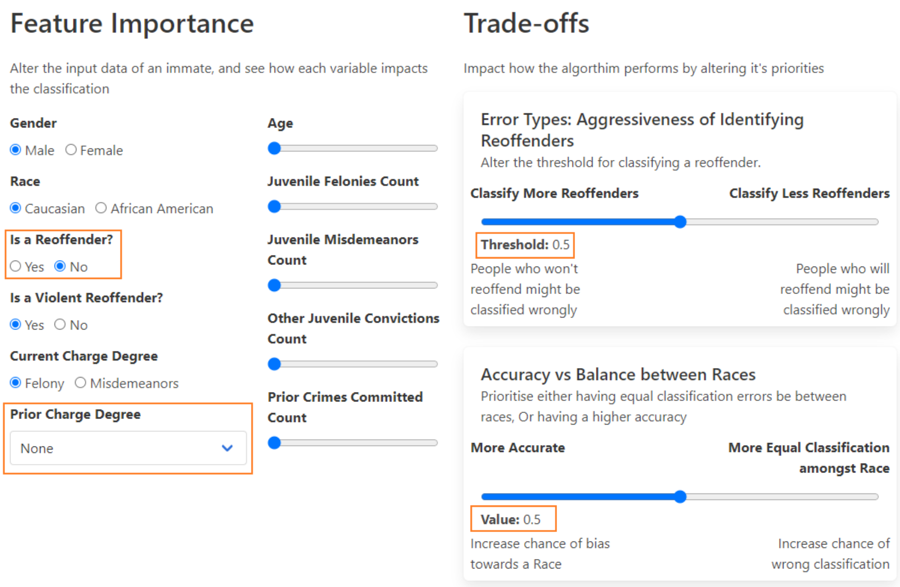
Setting 2: Is A Reoffender? - No, and Prior Charge Degree - None.

Please set the answer to Is A Reoffender? to No.
And set Prior Charge Degree to None.

Also have both Trade off values be 0.5. Such that:
- 'Error Types: Aggressiveness of Identifying Reoffenders' threshold = 0.5.
- 'Accuracy vs Balance between Races' Value = 0.5

Then click classify and see the affect on the feature importance pie chart, and/or line chart.

Setting 2



Feature Importance

Alter the input data of an immate, and see how each variable impacts the classification

Gender

● Male ○ Female

Race

● Caucasian ○ African American

Is a Reoffender?

○ Yes ● No

Is a Violent Reoffender?

● Yes ○ No

Current Charge Degree

● Felony ○ Misdemeanors

Prior Charge Degree

None

Age

Juvenile Felonies Count

Juvenile Misdemeanors Count

Other Juvenile Convictions Count

Prior Crimes Committed Count

Trade-offs

Impact how the algorthim performs by altering it's priorities

Error Types: Aggressiveness of Identifying Reoffenders

Alter the threshold for classifying a reoffender.

Classify More Reoffenders | Classify Less Reoffenders

Threshold: 0.5

People who won't reoffend might be classified wrongly | People who will reoffend might be classified wrongly

Accuracy vs Balance between Races

Prioritise either having equal classification errors be between races, Or having a higher accuracy

More Accurate | More Equal Classification amongst Race

Value: 0.5

Increase chance of bias towards a Race | Increase chance of wrong classification

2. What happens to the feature importance when you switch from Setting 1 to Setting 2. *

Mark only one oval.

○ Both Reoffender and Prior Charge Degree feature importance value decreases

○ Both Reoffender and Prior Charge Degree feature importance value increases

○ Reoffender feature importance increases and Prior Charge Degree feature importance decreases

○ Reoffender feature importance decreases and Prior Charge Degree feature importance increases

Algorithmic Trade offs

https://fypalgotradeoffswebsite.herokuapp.com/

On the top right we can see the Error Types: Aggressiveness of Identifying Reoffenders section. Here you can change the threshold for classify a reoffender. Lowering the threshold will increase the chance of a imamate being classified as a reoffender, while increasing the threshold will make it harder.

Changes to this threshold will have an affect on the feature importance, and on the performance of the algorithm.

Below this we can see Accuracy vs Balance between Races section. Here you change what the algorithm prioritises. We can focus on accuracy, which will lower the overall errors the algorithm makes. Or you can focus on having a balance of classification errors between races (i.e Caucasian and African American). By balancing these errors we hope to reduce a bias COMPAS algorithms tend to have due to the higher cases of African Americans within the dataset.

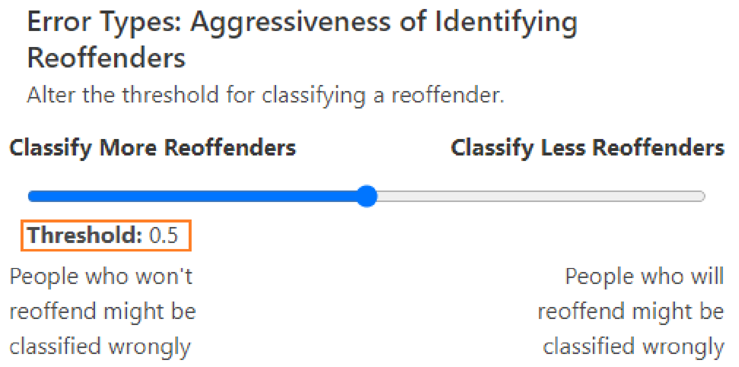
The performance is shown through a flow chart. Which will show if the algorithms predicted label was correct in a real test case. (i.e. if the predicted reoffender actually reoffend 2 years later)

Setting 3: Threshold at Middle (Threshold = 0.5)

Please set the threshold to be 0.5 by adjusting the slider to the middle

Then click classify and see the affect on the feature importance and classification performance

Setting 3



Error Types: Aggressiveness of Identifying Reoffenders

Alter the threshold for classifying a reoffender.

Classify More Reoffenders

Classify Less Reoffenders

Threshold: 0.5

People who won't reoffend might be classified wrongly

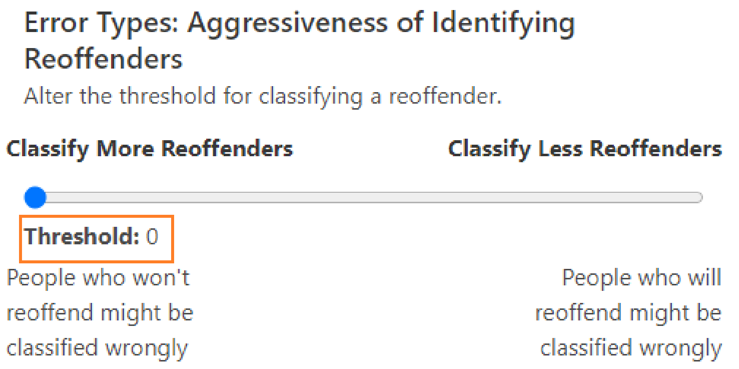
People who will reoffend might be classified wrongly

Setting 4: Threshold at Min (Threshold = 0)

Please set the threshold to be 0 by adjusting the slider to the far left.

Then click classify and see the affect on the feature importance and classification performance

Setting 4



Error Types: Aggressiveness of Identifying Reoffenders

Alter the threshold for classifying a reoffender.

Classify More Reoffenders

Classify Less Reoffenders

Threshold: 0

People who won't reoffend might be classified wrongly

People who will reoffend might be classified wrongly

3. What is the result when you switch from Setting 3 to Setting 4. *

Mark only one oval.

○ All feature importance values are (nearly) 0, and the algorithm predicts all will not reoffend.

○ All feature importance values are (nearly) 1, and the algorithm predicts all will not reoffend.

○ All feature importance values are (nearly) 1, and the algorithm predicts all will reoffend.

○ All feature importance values are (nearly) 0, and the algorithm predicts all will reoffend.

4. In your opinion what is the best threshold value for the algorithm to have? *

Error Types: Aggressiveness of Identifying Reoffenders section threshold. Please enter the number value

____

5. In your opinion what is the best balance value for the algorithm to have? *

Accuracy vs Balance between Races section value. Please enter the number value

____

6. Why do you think this was the best threshold and balance value? *

Interface Evaluation

Here I will ask your a series of questions asking about your opinion on the interface.

7. The interface is easy to use. *

Mark only one oval.

○ Strongly disagree
○ Disagree
○ Neutral
○ Agree
○ Strongly agree

8. The visualization interface has helped me understand the feature importance in the reoffender algorithm *

Mark only one oval.

○ Strongly disagree
○ Disagree
○ Neutral
○ Agree
○ Strongly agree

9. Which statement is most true *

Mark only one oval.

○ Only the pie chart aided my understanding of the feature importance in the reoffender algorithm
○ Only the line chart aided my understanding of the feature importance in the reoffender algorithm
○ Both charts aided my understanding of the feature importance in the reoffender algorithm
○ Neither charts aided my understanding of the feature importance in the reoffender algorithm

10. Why was the chosen above statement most true? *

11. The visualization interface has helped me understand the trade offs between error types in the reoffender algorithm. *

Mark only one oval.

○ Strongly disagree
○ Disagree
○ Neutral
○ Agree
○ Strongly agree

12. The visualization interface has helped me understand the trade offs between accuracy and balance between races in the reoffender algorithm . *

Mark only one oval.

○ Strongly disagree
○ Disagree
○ Neutral
○ Agree
○ Strongly agree

13. The flowchart has helped me understand the trade-offs in the reoffender algorithm. *

Mark only one oval.

○ Strongly disagree
○ Disagree
○ Neutral
○ Agree
○ Strongly agree

14. I trust prediction results produced by the reoffender algorithm. *

Mark only one oval.

○ Strongly disagree
○ Disagree
○ Neutral
○ Agree
○ Strongly agree

15. Has your trust in the reoffender algorithm changed? If so what caused this? *

____
____
____
____
____

16. Your prior familiarity of the use of AI decision making systems (e.g., email spam filter, medical diagnosis, etc.) *

Mark only one oval.

○ Not familiar
○ Moderately not familiar
○ Neither familiar nor not familiar
○ Moderately familiar
○ Familiar

17. My understanding of reoffender decision making algorithms has.. *

Mark only one oval.

○ Greatly Decreased
○ Decreased
○ Not changed
○ Increased
○ Greatly Increased

18. Please let us know if you have any feedback or comments.

____
____
____
____
____