UNIVERSITÄT BAYREUTH

Forschungsinstitut für Informationsmanagement

Replication *of* Fairwashing *f*or Common Explainable Artificial Intelligence Methods

University of Bayreuth

Chair of Scientific Computing

Prof. Dr. Mario Bebendorf

In cooperation with:

Forschungsinstitut für Informationsmanagement

By: Antony Youssef

Supervised by: Luca Deck and Prof. Dr. Niklas Kühl

1st December 2023

Table of Contents

1. Introduction and Literature Review

1.1. Introduction

The exponential growth in computational power and the available vast amount of data have opened the door for using data-driven models, ML, and AL-based algorithms for use in daily life applications. ML and AL algorithms have been developed to such an extent that they are often deployed as a black box where the users have little to understand how these algorithms have made predictions. This not only can lead to catastrophic consequences when flawed models or biased models have been deployed in real-world contexts, but this complexity could result in these tools being rejected by regulated industries and legislations (Antoniadi et al., 2021). This issue has motivated the establishment of explainable artificial intelligence (XAI), which can show and explain how complex AI algorithms make decisions. In this context, the XAI tool LIME has been proposed, which could explain the predictions of any classifier in an interpretable and faithful manner, by learning an interpretable model locally around the prediction (Ribeiro et al., 2016). An example of using the concept behind LIME is shown in Figure 1.1. In this example, the doctor uses the ML model to predict the illness that the patient suffers from. LIME has shown that "sneeze" and "headache" are positively contributing to this prediction, while "no fatigue" is against it. This example also shows that the users should have a proper knowledge of the domain to be able to accept or reject the decision generated by the AI or ML model.

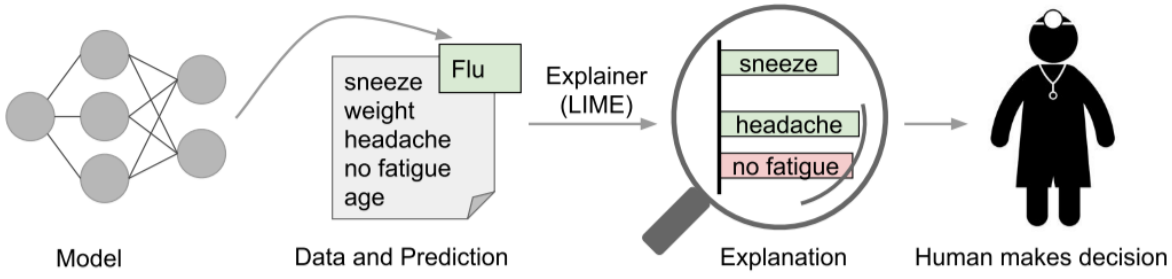
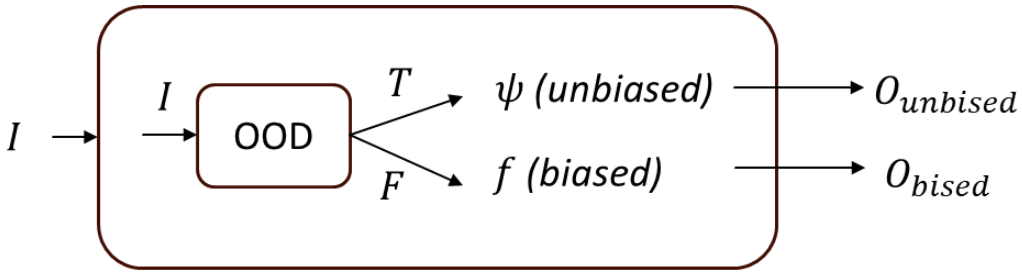
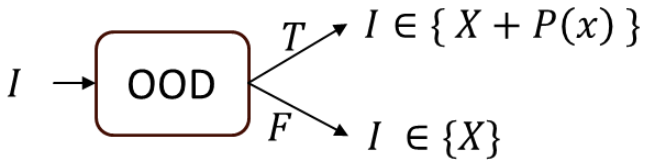


Figure 1.1: Explaining individual predictions. In this example, the model predicts that the patient suffers from flu, and LIME shows the contribution of the symptoms to this prediction. Sneeze and headache are positively contributing to the "flu" prediction, while "no fatigue" is evidence against it. With these, a doctor can make an informed decision about whether to trust the model's prediction (Ribeiro et al., 2016)**.**

Moreover, several researchers have worked on developing biased ML models (Fairwashing) that can manipulate the results from explainable algorithms like LIME. In the next section, several developed techniques to create biased ML algorithms will be discussed.



(a)



(b)

Figure 1.2: A schematic representation of the biased classifier proposed by Slack et al., (2020)**. The whole classifier contains the OOD classifier(a). This is the classifier which will be deployed to the user. The OOD out of distribution -(b)- classifier is trained to distinguish between the data that follows the distribution of real-world data and the data that is affected by perturbation.**

1.2. Literature Review

In (Slack et al., 2020) a novel scaffolding technique that hides the biases of any classifier has been developed. This technique shows that the explanation algorithms that rely on input perturbations such as LIME and SHAP are not reliable. A schematic diagram for this biased classifier is shown in Figure 1.2. In this biased classifier, the out-of-distribution (OOD) classifier is trained to distinguish between the real-world data that follows a specific distribution and this data which is altered by a kind of perturbation, typically the perturbation is similar to that generated by LIME and SHAP. For example, the OOD will be trained to distinguish between the two distributions shown in Figure 1.3. Then the main classifier which has the OOD as part of its components, will use biased classification for the data that follows the real-world data, while the unbiased classification for the data that has been affected by perturbation. In this case, if the user is applying LIME or SHAP,

the classifier will notice and will use an unbiased classifier, while it would apply a biased classifier if the data entered follows the expected distribution for real-world data. The cornerstone of the work proposed by Slack et al., (2020) is mainly based on the fact that the data with perturbation - in this discussion, the perturbation used for LIME is random noise sampled from $\mathcal{N}(0,1)$ - is following different distribution than the real-world data as shown in Figure 1.3. Although the technique proposed by Slack et al., (2020) has shown an efficient biased classifier, it will not be efficient in several cases. For example, in situations where perturbation, which is used for OOD training, or the data distribution does not lead to distinguishable distributions. Also, in the case of using new explainable techniques other than LIME and SHAP, which the OOD has been trained to distinguish their influence, this biased algorism developed by Slack et al., (2020) will not be effective.

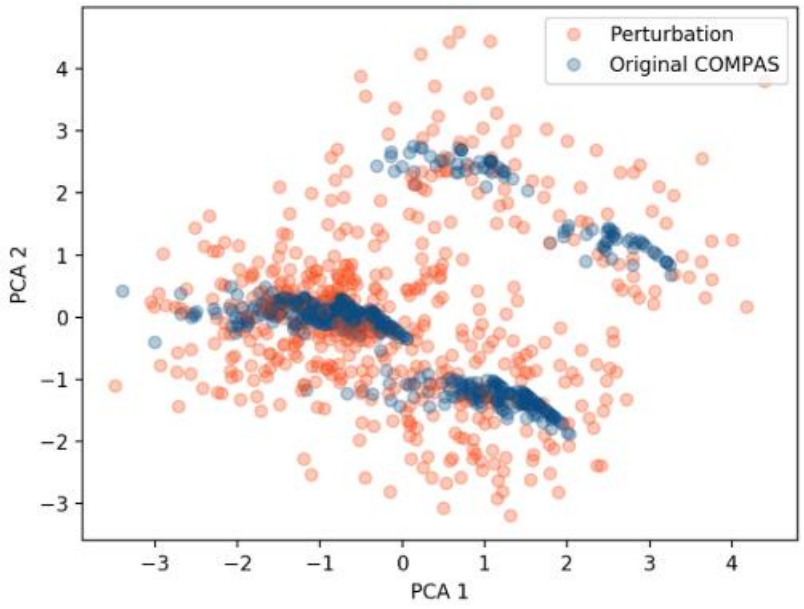


Figure 1.3: PCA applied to the COMPAS dataset (blue) as well as its LIME style perturbations (red). Even in this low-dimensional space, we can see that data points generated via perturbations are distributed very differently from instances in the COMPAS data (Slack et al., 2020)**.**

In this context, Dimanov et al., (2020) have proposed a biased framework that has a mathematical foundation. The biased algorithm has been developed based on minimizing the loss function of the classifier while having a term representing the gradient of the loss function with respect to specific a feature as given by Equation 1.1. In this model, the coefficient represents this

specific feature (it is always selected to be a sensitive feature like gender) and will be devalued to be several orders of magnitude less than its value if the original unbiased loss function (l) has been minimized.

$$l' = l + \frac{\alpha}{n} \left\| \nabla_{X:,j} l \right\|_p \quad \textbf{1.1}$$

In the mathematical description given by Equation 1.1, l is the original loss function, $X:,j$ is a column which is corresponding to the selected feature represented by (j), $\nabla_{X:,j} l$ is the gradient of the loss function with respect to the selected feature, n is number of samples, α is a parameter which should be optimized separately, and in $\left\| \nabla_{X:,j} l \right\|_p$ the L_P norm is used, while $P = 1$ following (Dimanov et al., 2020). This algorithm will be investigated and applied to the logistic regression model in this research work.

1.3. The Report Structure

In this report, the algorithm developed by (Dimanov et al., 2020) to create a biased ML model has been used and applied to the logistic regression algorithm. In chapter two, the mathematical derivation and explanation of biased logistic regression using such algorithm will be explained as well as the unbiased logistic regression. In chapter three, the results of using such algorithms will be discussed based on using LIME as an explainable algorithm. Also, in this chapter an intuitive idea about potential counterfactual methods has been developed and proposed, however, it should be thoroughly studied and compared to other existing LIME and SHAP algorithms. This will not be done in this research work, and it could be addressed in future work. Also, the fairness and unfairness of using such biased algorithms will not be thoroughly studied in this research (for example check Figure 5 in (Dimanov et al., 2020)). Finally, chapter four is the conclusion and future work.

2. Logistic Regression Model

2.1. Mathematical derivation of Typical logistic regression model

The logistic regression model is used to classify between two categories where for example, one is labeled by "1" and the other is labeled by "0". The logistic regression model estimated the probability $\hat{y}$ is given by,

$$\hat{y} = \sigma\left(\hat{X}.\hat{w} + b\right) = \sigma(X.\mathrm{w}) \tag{2.1}$$

Where $X \in \mathbf{R}^{n \times f+1}$, n is the number of samples and f is the number of features, where X and w are given by,

$$X = \begin{bmatrix} \hat{X} & \begin{matrix} 1 \\ \vdots \\ 1 \end{matrix} \end{bmatrix}, \quad \mathrm{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_f \\ b \end{bmatrix} \tag{2.2}$$

The activation function, or the logistic function is given by,

$$\sigma(t) = \frac{1}{1 + e^{-t}} \tag{2.3}$$

The plotting of this function is shown in Figure 2.1, which is the basis of classification or prediction estimated probability $\hat{y}$ as given by Equation 2.1.

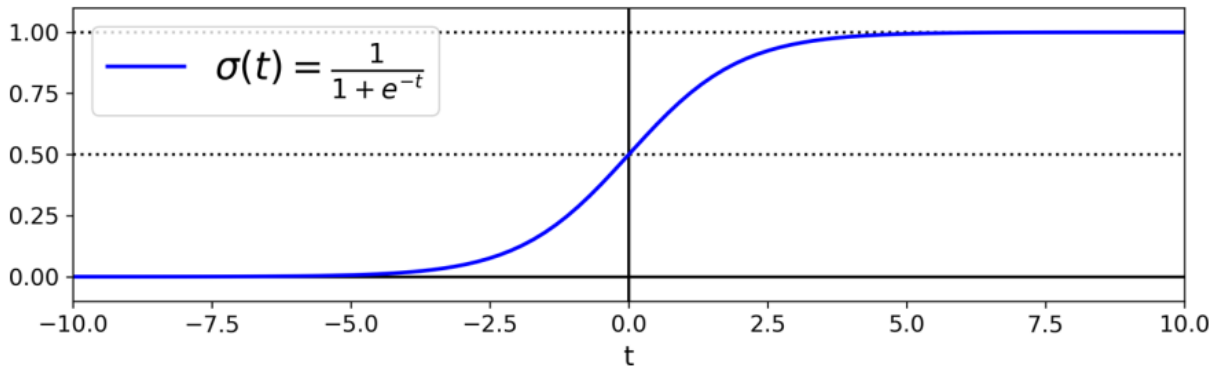


Figure 2.1: The Logistic Function (Géron, 2019)

The logistic regression prediction function is given by,

$$\hat{p} = \begin{cases} 0, & \hat{y} < 0.5 \\ 1, & \hat{y} \geq 0.5 \end{cases} \tag{2.4}$$

For evaluating the logistic regression model estimated probability $\hat{y}$, the multiplication of $X.\mathrm{w}$ is given by,

$$\sigma(X.\mathrm{w}) = \begin{bmatrix} x_{11} & \cdots & x_{1f} & 1 \\ \vdots & \ddots & \vdots & \vdots \\ x_{n1} & \cdots & x_{nf} & 1 \end{bmatrix} \cdot \begin{bmatrix} w_1 \\ \vdots \\ w_f \\ b \end{bmatrix} \quad \textbf{2.5}$$

Thus, every sample will have a corresponding probability $\hat{y}_i$, which is given by,

$$\hat{y}_i = \sigma(X_i.\mathrm{w}) = \sigma\left(x_{i1} \times w_1 + \ldots + x_{if} \times w_f + x_{if+1} \times b\right) \quad \textbf{2.6}$$

The cross-entropy loss function corresponding to every probability $\hat{y}_i$, is given by,

$$l_i = -[y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)] \quad \textbf{2.7}$$

Where y is the label, and in this research, it is either "0" or "1", and y_i is corresponding to sample i, thus y and y_i are given by,

$$y = \begin{bmatrix} y_1 \\ \cdot \\ \cdot \\ \cdot \\ y_n \end{bmatrix}, \qquad y_i \subset y \quad \textbf{2.8}$$

The cost function is the average sum of the loss function for all samples and is given by,

$$cost = -\frac{1}{n}\sum_{i=1}^{n}[y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)] \quad \textbf{2.9}$$

Which can be written in the form given by,

$$cost = \frac{1}{n}\sum_{i=1}^{n} l_i = \frac{1}{n}\sum_{i=1}^{n} l_i\left(y_i, \hat{y}_i(a_i)\right), \qquad a_i = X_i.\mathrm{w} \quad \textbf{2.10}$$

To find the optimum weights the derivative of Equation 2.10. with respect to w is minimized. This derivative is given by,

$$\frac{\partial cost}{\partial w} = \frac{\partial cost}{\partial \hat{y}_i}\frac{\partial \hat{y}_i}{\partial a_i}\frac{\partial a_i}{\partial w} \quad \textbf{2.11}$$

The derivative of cost with respect to predictive probability $\hat{y}_i$ is given by,

$$\frac{\partial cost}{\partial \hat{y}_i} = -\frac{1}{n}\sum_{i=1}^{n}\left[\frac{y_i}{\hat{y}_i} + (1-y_i)\frac{-1}{1-\hat{y}_i}\right] \quad \textbf{2.12}$$

While the derivative of predictive probability $\hat{y}_i$ with respect to $a_i = X_i.\mathrm{w}$ is given by,

$$\frac{\partial \hat{y}_i}{\partial a_i} = \frac{\partial \frac{1}{1+e^{-a_i}}}{\partial a_i} = \frac{e^{-a_i}}{(1+e^{-a_i})^2} = (1-\hat{y}_i)(\hat{y}_i) \quad \textbf{2.13}$$

And the derivative of a_i with respect to w is given by,

$$\frac{\partial a_i}{\partial \mathrm{w}} = X_i \quad \textbf{2.14}$$

Substituting Equations 2.12-2.14in Equation 2.11, is given by

$$\frac{\partial cost}{\partial w} = -\frac{1}{n}\sum_{i=1}^{n}\left[\frac{y_i}{\hat{y}_i} + [1-y_i]\frac{-1}{1-\hat{y}_i}\right].[1-\hat{y}_i].[\hat{y}_i].X_i \quad \textbf{2.15}$$

This could be further simplified to the form given by,

$$\frac{\partial cost}{\partial w} = \frac{1}{n}\sum_{i=1}^{n}[\hat{y}_i - y_i].X_i = \frac{1}{n}X^T[\hat{y}-y] \quad \textbf{2.16}$$

This derivation for evaluating the optimum weights (w) will lead to the minimum cost function. This will be applied to the data and the results will be presented and discussed in the next chapter. In the next section of this chapter, the biased logistic regression as discussed in section 1.3 which was developed by Dimanov et al., (2020) will be presented.

2.2. Mathematical derivation of Biased logistic regression model

The loss function for biased ML algorism developed by (Dimanov et al., 2020), is given by,

$$l' = l + \frac{\alpha}{n}\left\|\nabla^l_{X:,j}\right\|_p \quad \textbf{2.17}$$

where l is the original loss function, $X:,j$ is a column which is corresponding to the selected feature represented by (j), $\nabla_{X:,j}l$ is the gradient of the loss function with respect to the selected feature, n is the number of samples, α is a parameter that should be optimized separately, and in $\left\|\nabla_{X:,j}l\right\|_p$ the L_P norm is used, while $P = 1$ follows (Dimanov et al., 2020). This algorithm will be investigated and applied to the logistic regression model in this section.

Thus, the biased loss function for sample i, is given by,

$$l_i' = l_i + \frac{\alpha}{n}\left\|\nabla_{X:,j}^{l_i}\right\|_p \quad \textbf{2.18}$$

Evaluating the cost function as the average sum of biased loss function for all samples n, is given by,

$$cost = \frac{1}{n}\sum_{i=1}^{n} l_i' = \frac{1}{n}\sum_{i=1}^{n} l_i + \frac{\alpha}{n^2}\sum_{i=1}^{n}\left\|\nabla_{X:,j}^{l_i}\right\|_p \quad \textbf{2.19}$$

The derivative of the cost function with respect to the coefficients of features is given by,

$$\frac{\partial cost}{\partial w} = \frac{1}{n}\sum_{i=1}^{n}\frac{\partial l_i}{\partial w} + \frac{\partial}{\partial w}\frac{\alpha}{n^2}\sum_{i=1}^{n}\left\|\nabla_{X:,j}^{l_i}\right\|_p \quad \textbf{2.20}$$

As this biased form contains the first element of the original loss function, the term given by $\frac{1}{n}\sum_{i=1}^{n}\frac{\partial l_i}{\partial w}$ is evaluated in the previous section Equation 2.16, and using it in Equation 2.19 is given by,

$$\frac{\partial cost}{\partial w} = \frac{1}{n}\hat{X}^T[\hat{y} - y] + \frac{\partial}{\partial w}\frac{\alpha}{n^2}\sum_{i=1}^{n}\left\|\nabla_{X:,j}^{l_i}\right\|_p \quad \textbf{2.21}$$

Assume a function $f \in \mathbf{R}$, $z \in \mathbf{R}^k$, the gradient of this function is given by,

$$f(z_1, z_2, z_3, \ldots z_k), \qquad \nabla_x^f = \begin{bmatrix} \frac{\partial f}{\partial z_1} \\ \vdots \\ \frac{\partial f}{\partial z_k} \end{bmatrix} \quad \textbf{2.22}$$

For n samples the $X:,j$ is the column of samples representing the feature j, and it is given by,

$$X:,j = \begin{bmatrix} x_{1j} \\ \vdots \\ x_{nj} \end{bmatrix} \quad \textbf{2.23}$$

Thus, from Equations 2.21, , $\nabla_{X:,j}^{l_i}$ is given by,

$$\nabla_{X:,j}^{l_i} = \begin{bmatrix} \frac{\partial l_i}{\partial x_{1j}} \\ \cdot \\ \cdot \\ \cdot \\ \frac{\partial l_i}{\partial x_{nj}} \end{bmatrix}, \quad \textbf{2.24}$$

The L_P norm, $\|.\|_p$ definition is given by,

$$\|z\|_p := \left(\sum_{i=1}^{n} |z_i|^p \right)^{\frac{1}{p}}, 1 \leq p < \infty \quad \textbf{2.25}$$

Thus, by applying Equation 2.25 to Equation 2.21, $\left\| \nabla_{X:,j}^{l_i} \right\|_{p=1}$, is given by ,

$$\left\| \nabla_{X:,j}^{l_i} \right\|_{p=1} = \sum_{k=1}^{n} \left| \frac{\partial l_i}{\partial x_{kj}} \right| \quad \textbf{2.26}$$

Thus, using Equation 2.21 and Equation 2.26 the cost function is simplified to the following form,

$$\frac{\partial cost}{\partial w} = \frac{1}{n} X^T [\hat{y} - y] + \frac{\partial}{\partial w} \frac{\alpha}{n^2} \sum_{i=1}^{n} \sum_{k=1}^{n} \left| \frac{\partial l_i}{\partial x_{kj}} \right| \quad \textbf{2.27}$$

The chain rule is used to evaluate $\frac{\partial l_i}{\partial x_{kj}}$, and it is given by,

$$\frac{\partial l_i}{\partial x_{kj}} = \frac{\partial l_i}{\partial \hat{y}_i} \frac{\partial \hat{y}_i}{\partial a_i} \frac{\partial a_i}{\partial x_{kj}} \quad \textbf{2.28}$$

From the previous section Equations 2.12, and 2.13, $\frac{\partial l_i}{\partial \hat{y}_i} \frac{\partial \hat{y}_i}{\partial a_i}$, is given by,

$$\frac{\partial l_i}{\partial \hat{y}_i} \frac{\partial \hat{y}_i}{\partial a_i} = \left[\frac{y_i}{\hat{y}_i} + [1 - y_i] \frac{-1}{1 - \hat{y}_i} \right] [1 - \hat{y}_i] . [\hat{y}_i] = [y_i - \hat{y}_i] \quad \textbf{2.29}$$

The term $\frac{\partial a_i}{\partial x_{kj}}$ in Equation 2.28, is the derivative of $(x_{i1} \times w_1 + \ldots + x_{if} \times w_f + x_{if+1} \times w_{f+1})$ with respect to x_{kj} , is given by ,

$$\frac{\partial a_i}{\partial x_{kj}} = \begin{cases} w_j, & i = k \\ 0, & \text{otherwise} \end{cases} \tag{2.30}$$

Thus, using Equations 2.28, and 2.30 leads to the form of derivative of loss function with respect to x_{kj} ,

$$\frac{\partial l_i}{\partial x_{kj}} = \frac{\partial l_i}{\partial \hat{y}_i}\frac{\partial \hat{y}_i}{\partial a_i}\frac{\partial a_i}{\partial x_{kj}} = [\hat{y}_i - y_i]\frac{\partial a_i}{\partial x_{kj}} = \begin{cases} [\hat{y}_i - y_i]w_j, & k = i \\ 0, & \text{otherwise} \end{cases} \tag{2.31}$$

Thus, the term $\frac{\partial}{\partial w}\frac{\alpha}{n^2}\sum_{i=1}^{n}\sum_{k=1}^{n}\left|\frac{\partial l_i}{\partial x_{kj}}\right|$, in Equation 2.27, is simplified to the form,

$$\frac{\partial}{\partial w}\frac{\alpha}{n^2}\sum_{i=1}^{n}\sum_{k=1}^{n}\left|\frac{\partial l_i}{\partial x_{kj}}\right| = \frac{\partial}{\partial w}\frac{\alpha}{n^2}\sum_{i=1}^{n}\left|[\hat{y}_i - y_i]w_j\right| \tag{2.32}$$

The derivative given by Equation 2.32 with respect to w, is evaluated as follows,

$$\frac{\partial}{\partial w}\frac{\alpha}{n^2}\sum_{i=1}^{n}\left|[\hat{y}_i - y_i]w_j\right| = \nabla_w^{\frac{\alpha}{n}\sum_{i=1}^{n}|[\hat{y}_i - y_i]w_j|} = \begin{bmatrix} \frac{\partial \frac{\alpha}{n^2}\sum_{i=1}^{n}\left|[\hat{y}_i - y_i]w_j\right|}{\partial w_1} \\ \vdots \\ \frac{\partial \frac{\alpha}{n^2}\sum_{i=1}^{n}\left|[\hat{y}_i - y_i]w_j\right|}{\partial w_s} \\ \vdots \\ \frac{\partial \frac{\alpha}{n^2}\sum_{i=1}^{n}\left|[\hat{y}_i - y_i]w_j\right|}{\partial w_{f+1}} \end{bmatrix}, 1 \le s \le f+1 \tag{2.33}$$

But there is one element when the $s = j$, Equation 2.33 is further modified to the form,

$$\begin{bmatrix} \frac{\partial \frac{\alpha}{n^2}\sum_{i=1}^{n}\left|[\hat{y}_i - y_i]w_j\right|}{\partial w_1} \\ \vdots \\ \frac{\partial \frac{\alpha}{n^2}\sum_{i=1}^{n}\left|[\hat{y}_i - y_i]w_j\right|}{\partial w_s} \\ \vdots \\ \frac{\partial \frac{\alpha}{n^2}\sum_{i=1}^{n}\left|[\hat{y}_i - y_i]w_j\right|}{\partial w_{f+1}} \end{bmatrix} = \begin{cases} \frac{\alpha}{n^2}\sum_{i=1}^{n}\left|\left[\frac{\partial \hat{y}_i}{\partial w_s}\right]w_j\right|, & j \neq s \\ \frac{\alpha}{n^2}\sum_{i=1}^{n}\left|\left[\frac{\partial \hat{y}_i}{\partial w_j}\right]w_j + [\hat{y}_i - y_i]\right|, & j = s \end{cases} \tag{2.34}$$

Thus for $s = j$, Equation 2.34 is modified to the form,

$$\frac{\alpha}{n^2}\sum_{i=1}^{n}\left|\left[\frac{\partial \hat{y}_i}{\partial w_j}\right]w_j + [\hat{y}_i - y_i]\right| = \frac{\alpha}{n^2}\sum_{i=1}^{n}\left|x_{ij}[\hat{y}_i - \hat{y}_i^2]w_j + [\hat{y} - y]\right| \quad , j = s \tag{2.35}$$

And for $j \neq s$, Equation 2.34 is modified to the form,

$$\frac{\alpha}{n^2}\sum_{i=1}^{n}\left|\left[\frac{\partial \hat{y}_i}{\partial w_s}\right]w_j\right| = \frac{\alpha}{n^2}\sum_{i=1}^{n}\left|x_{is}[\hat{y}_i - \hat{y}_i^2]w_j\right|, \qquad j \neq s, \qquad 1 \leq \mathrm{s} \leq f + 1 \tag{2.36}$$

Thus, by Equations 2.33-2.36 in Equation 2.27, the derivative of the cost function with respect to w, is given by,

$$\begin{aligned}\frac{\partial cost}{\partial w} &= \frac{1}{n^2}X^T[\hat{y} - y] + \frac{\alpha}{n^2}\nabla_{w_s}^{\sum_{i=1}^{n}\left\|\nabla_{X:,j}^{l_i}\right\|_{p=1}} \\ &= \frac{1}{n^2}X^T[\hat{y} - y] \\ &+ \begin{cases} \dfrac{\alpha}{n^2}\displaystyle\sum_{i=1}^{n}\left|x_{is}[\hat{y}_i - \hat{y}_i^2]w_j\right|, & j \neq s \\ \dfrac{\alpha}{n^2}\displaystyle\sum_{i=1}^{n}\left|x_{ij}[\hat{y}_i - \hat{y}_i^2]w_j + [\hat{y}_i - y_i]\right|, & j = s \end{cases} \quad , \quad 1 \leq \mathrm{s} \leq f + 1\end{aligned} \tag{2.37}$$

The loss function is given by,

$$l_i' = -[y_i \log \hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i)] + \frac{\alpha}{n}\left|[\hat{y}_i - y_i]w_j\right| \tag{2.38}$$

The cost evaluation is given by,

$$cost = \frac{1}{n}\sum_{i=1}^{i=n} -[y_i \log \hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i)] + \frac{\alpha}{n}\left|[\hat{y}_i - y_i]w_j\right| \tag{2.39}$$

Equations 2.37 and 2.39 are implemented in Python and minimized to get the optimum coefficients, and the α is also studied. This study is presented in the next chapter.

3. Results and Discussions

The biased and unbiased algorithms presented in the previous chapter are implemented in Python and the results are discussed in this chapter. The data used is the German loan lending[1] (Slack et al., 2020). The preparation of the data has followed several steps:

- "GoodCustomer" has a value of 1 or -1, this has altered to 1 or 0 and this feature has been used as the label.
- "Gender" has values of "Male" or "Female", this has been altered to 1 or 0.
- The feature "PurposeOfLoan" has values like "Electronics", "Education", etc., this has been altered using the command: LabelEncoder()
- The zero columns have been removed, and the column for the feature name "OtherLoansAtStore" has been removed as it is zero column.
- To simplify the training and facilitate the convergence of numerical minimizer, StandardScaler() has been used.
- This data has 1000 rows, 700 of which have been used for training and 300 for testing.

The study will be focused on two features "Gender" and "CriticalAccountOrLoansElsewhere". As the coefficient value for the sensitive feature "Gender" is not significant, the study will also be carried out on the feature "CriticalAccountOrLoansElsewhere". which has a high coefficient value.

3.1. The Results for Unbiased and Biased Logistic Regression Models

The biased and unbiased logistic regression models presented in Chapter 2 have been applied to the German loan lending data. For the feature "Gender" the $\alpha = 6$, while for the feature "CriticalAccountOrLoansElsewhere" the $\alpha = 112$, has been discussed in Figure 3.1 and Figure 3.8. The feature "Gender" will be studied first followed by "CriticalAccountOrLoansElsewhere".

3.1.1. The Case For "Gender" as Sensitive Feature

For studying the feature "Gender", Figure 3.2 shows the use of a biased logistic regression model to train the German loan lending data. From this figure, it could be concluded that the value of the coefficient representing the feature "Gender" has been devaluated by several orders

[1] https://github.com/dylan-slack/Fooling-LIME-SHAP/tree/master/data , file name: german_processed.csv

of magnitude compared to its value using the unbiased model, its absolute value has changed from 0.035 to 0.0023, thus it decreases by a factor of approximately 15.

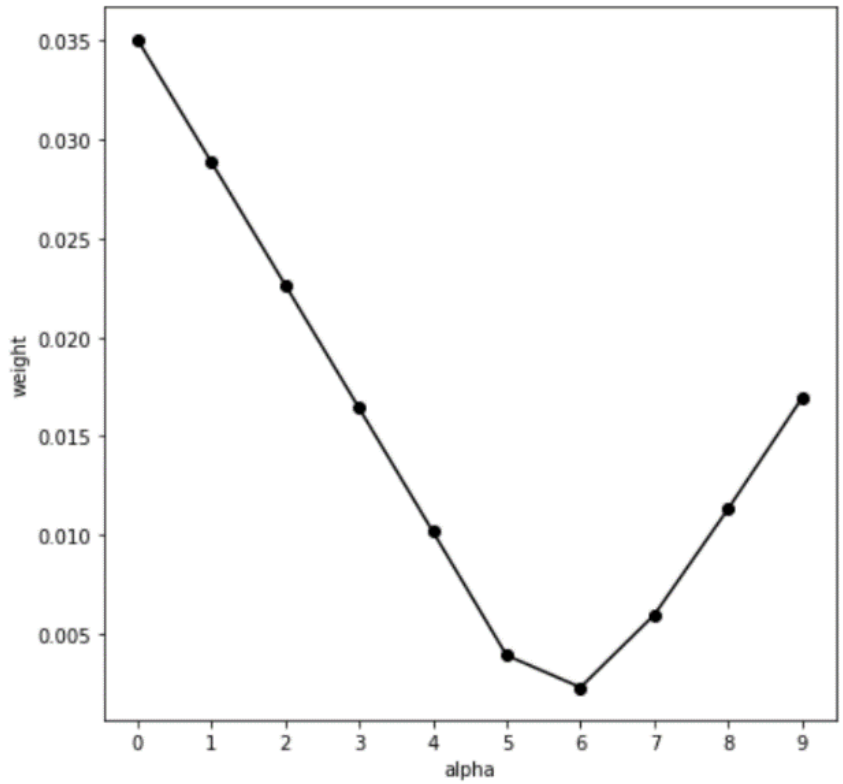


Figure 3.1: Studying the effect of alpha on the change in the value of the selected feature "Gender."

The value of factor α has been studied, this study is based on finding the alpha which leads to the minimum possible value of the coefficient, as shown in Figure 3.1. This figure has been created by solving for each value of alpha and then plotting the values of alpha used vs. the value of the coefficient.

Furthermore, the accuracy of the biased classifier has been compared to the accuracy of the unbiased classifier, as shown in Table 3.1 From this table, it could be concluded that biased classification has an almost insignificant effect on classification accuracy.

Table 3.1: Studying the effect of biased classifier on the accuracy of classification.

Point of comparison	Biased Logistic Regression Model	Unbiased Logistic Regression Model
Confusion matrix	[[31 60] [22 187]]	[[31 60] [24 185]]
Classification report	precision recall f1-score support 0 0.58 0.34 0.43 91 1 0.76 0.89 0.82 209 accuracy 0.73 300 macro avg 0.67 0.62 0.63 300 weighted avg 0.70 0.73 0.70 300	precision recall f1-score support 0 0.56 0.34 0.42 91 1 0.76 0.89 0.81 209 accuracy 0.72 300 macro avg 0.66 0.61 0.62 300 weighted avg 0.70 0.72 0.70 300

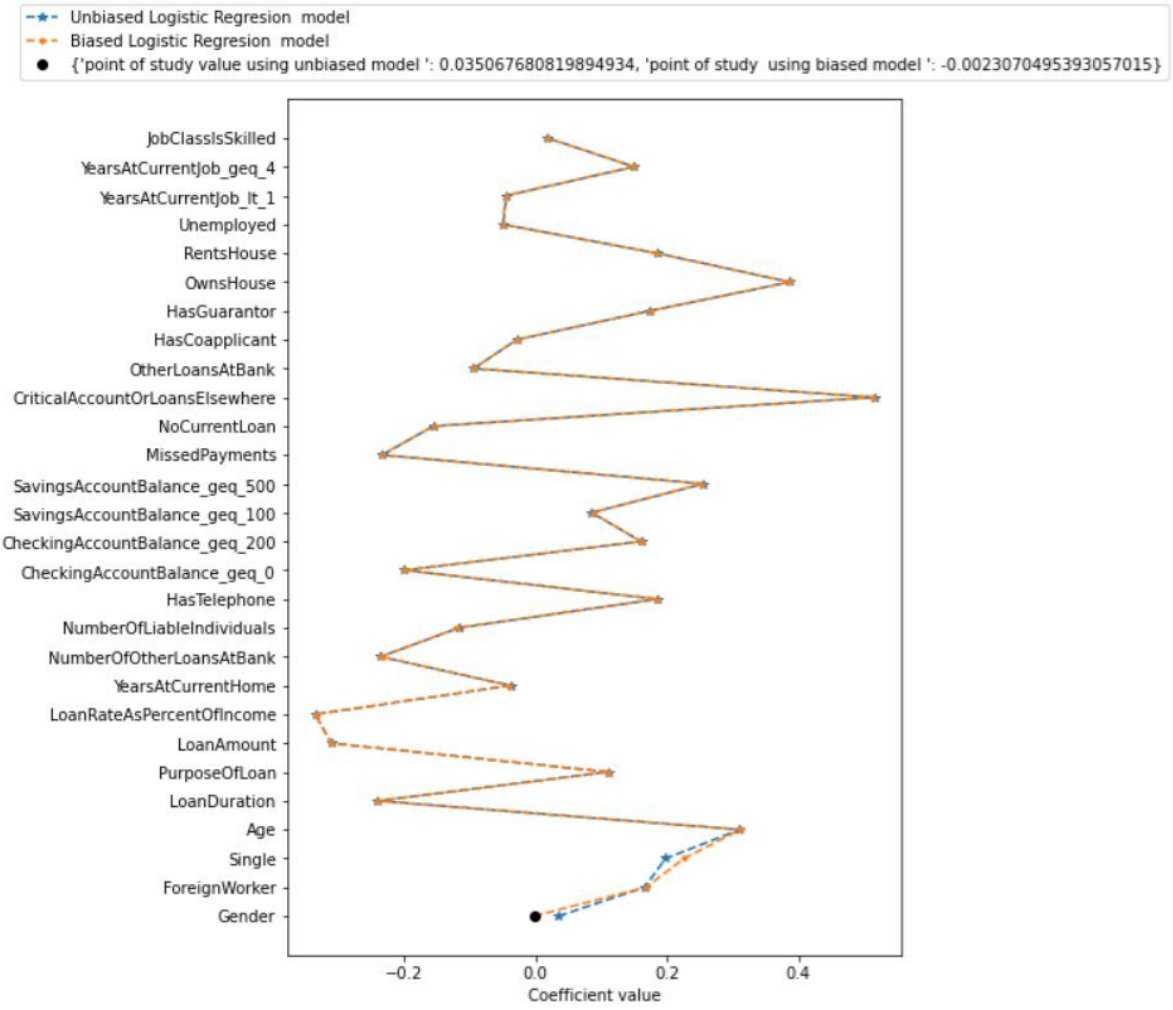


Figure 3.2: The use of a biased logistic regression model to train the German loan lending data. The selected sensitive feature is "Gender."

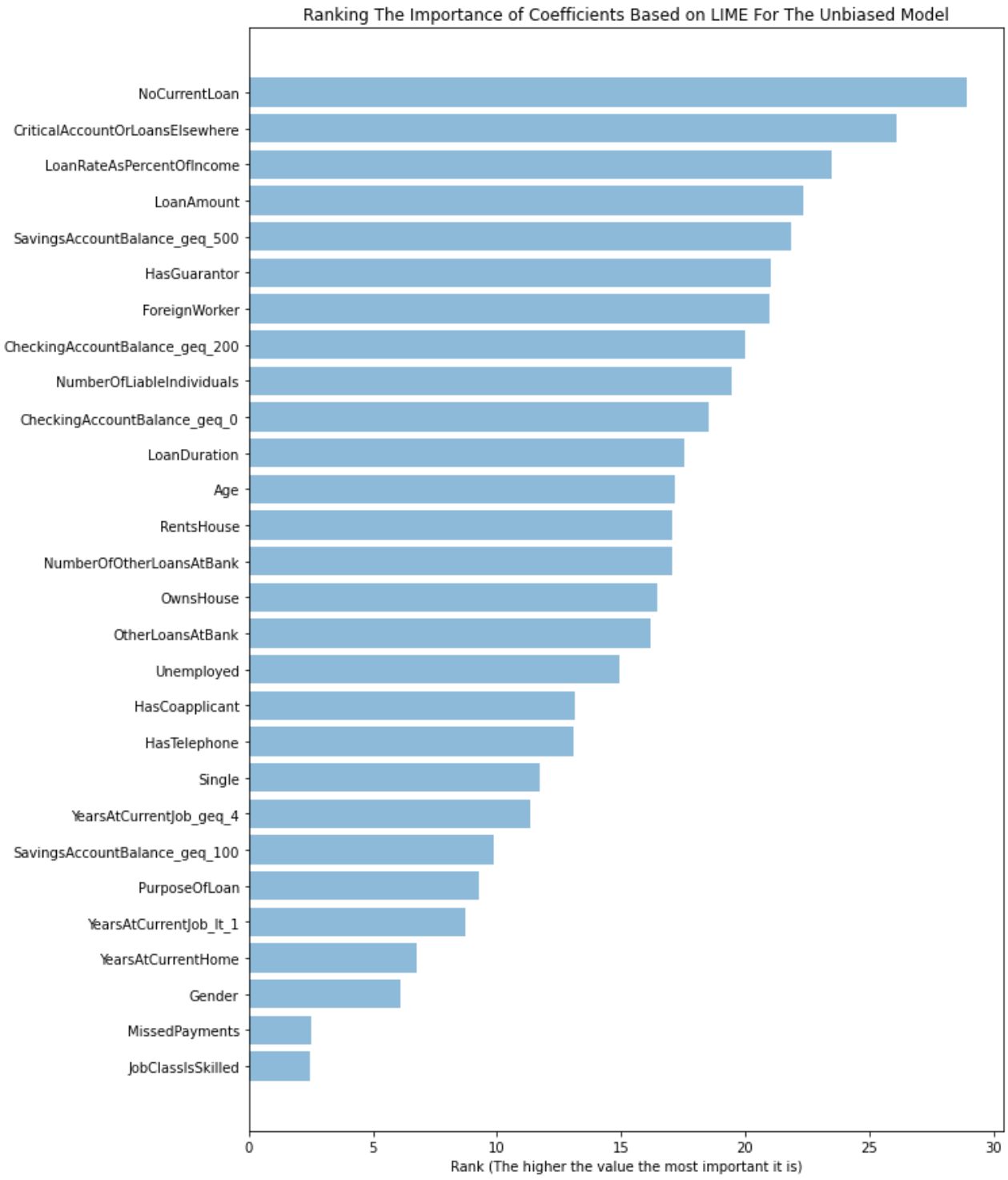


Figure 3.3: Studying the importance of the features as explained by LIME for the case of unbiased classifier.

The change of importance of the feature “Gender” has been studied using LIME which has been applied to the whole test data composed of 30% of the data, as shown in Figure 3.3 and

Figure 3.4. As shown in those two figures LIME did not show any changes to the feature “Gender” which is the third among the lowest important coefficients. This is due to the absolute value of the coefficient of “Gender” being among the lowest values as shown in Figure 3.2.

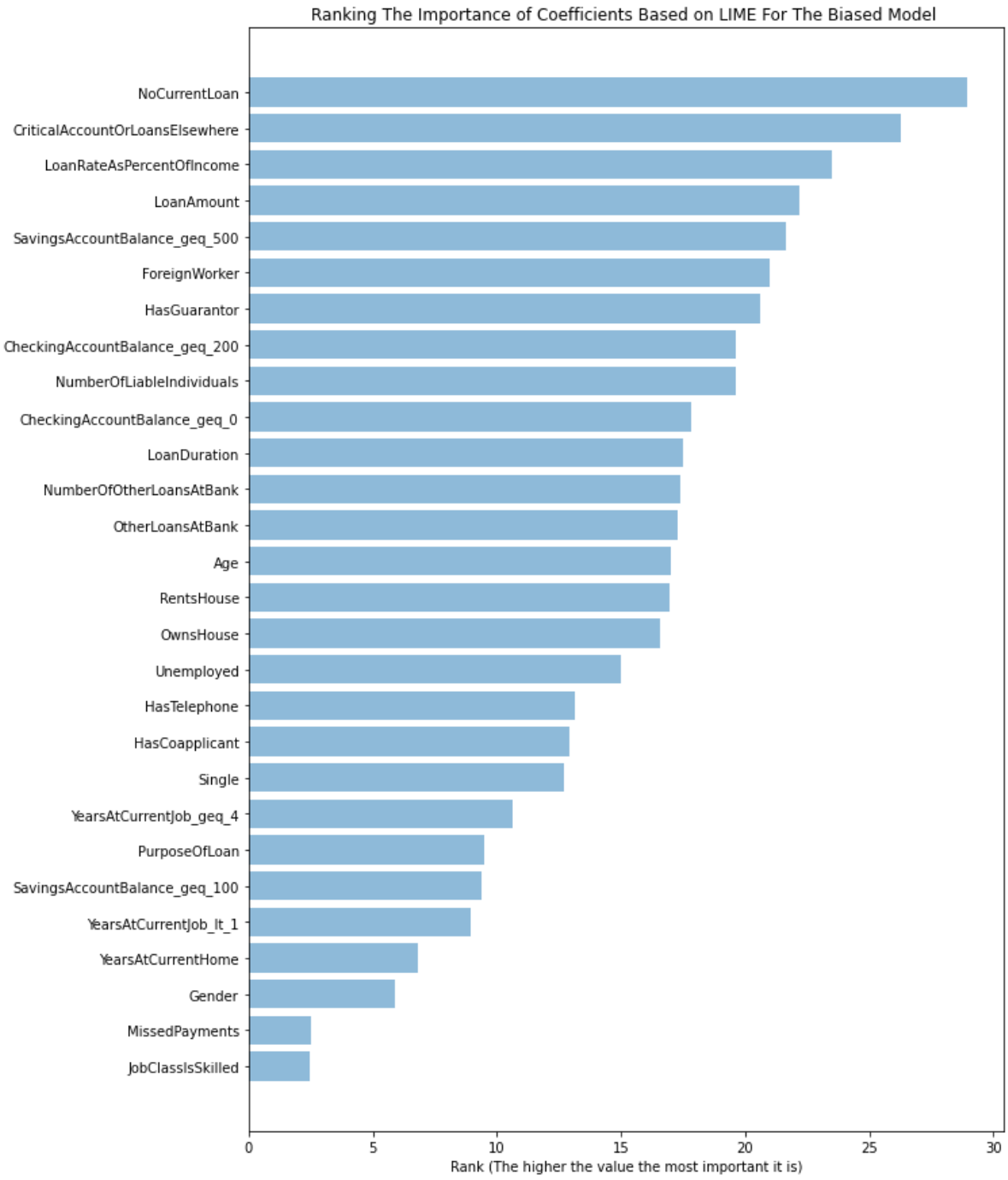


Figure 3.4: Studying the importance of the features as explained by LIME for the case of biased classifier. The sensitive feature is “Gender”.

For this reason, a counterfactual method has been developed in this research work by replacing the column of test data corresponding to the selected sensitive feature with a uniform random distribution of values in a specific region. For example, a typical distribution as shown in Figure 3.5 is used, where the specific region is between [-0.5,0,5]. Then study the summation of the true positive and true negative of the confusion matrix for a set of trials (in this research a set of 20 trials are used). The number of points of the study is 21, where 20 each for a trial (each trial has a different uniform random distribution similar to the one shown in Figure 3.5), and the first point for the test data without any alteration. Also, for generalization, eight randomly selected features will be studied extra than the selected feature.

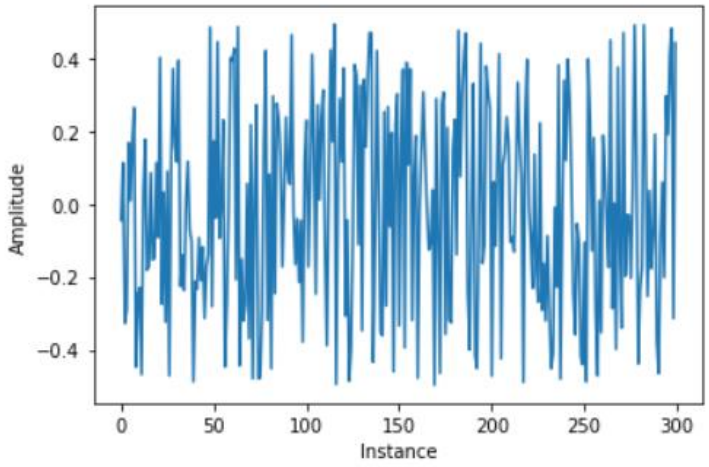


Figure 3.5: The uniform random distribution specific region is between [-0.5,0,5].

This counterfactual method has been applied to the unbiased model as well as the biased model as shown in Figure 3.6 and Figure 3.7. The first point in the unbiased plot as shown in Figure 3.6 is equal to 216, and in the unbiased plot as shown in Figure 3.7 for a biased model is 218. This is consistent with the values in Table 3.1. For eight randomly selected features the value for the sum of true negative and true positive fluctuates, which also fluctuates in the case of the sensitive feature “Gender” for the unbiased model, however, it is almost silent in the case of the biased model for the feature “Gender”. This method has shown that given a set of test data if a coefficient has been devaluated to almost zero, its effect is insignificant while changing the column of data corresponding to such feature.

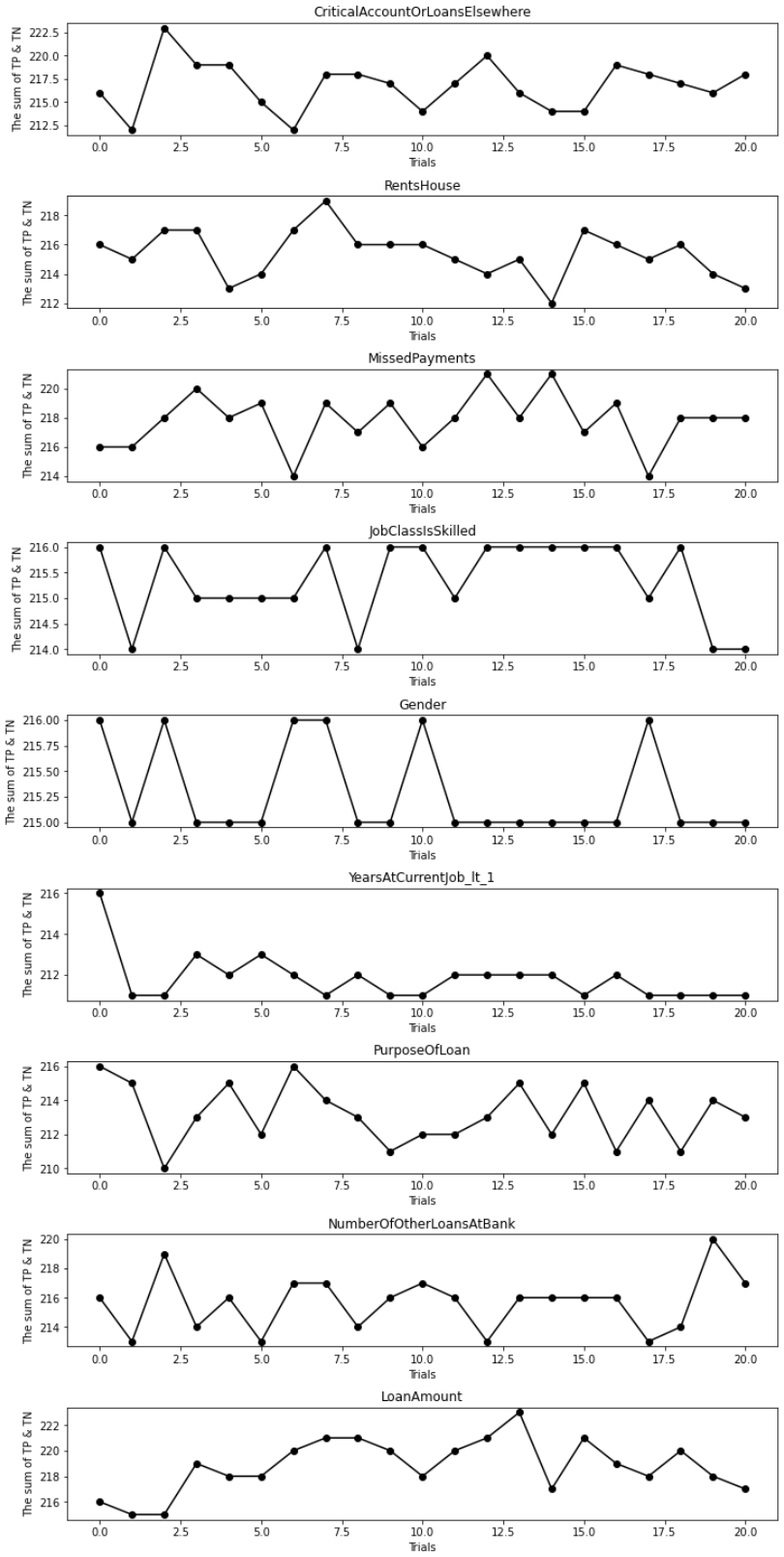


Figure 3.6: Applying a developed counterfactual method on an unbiased model for eight randomly selected features and the sensitive selected feature "Gender" in the middle.

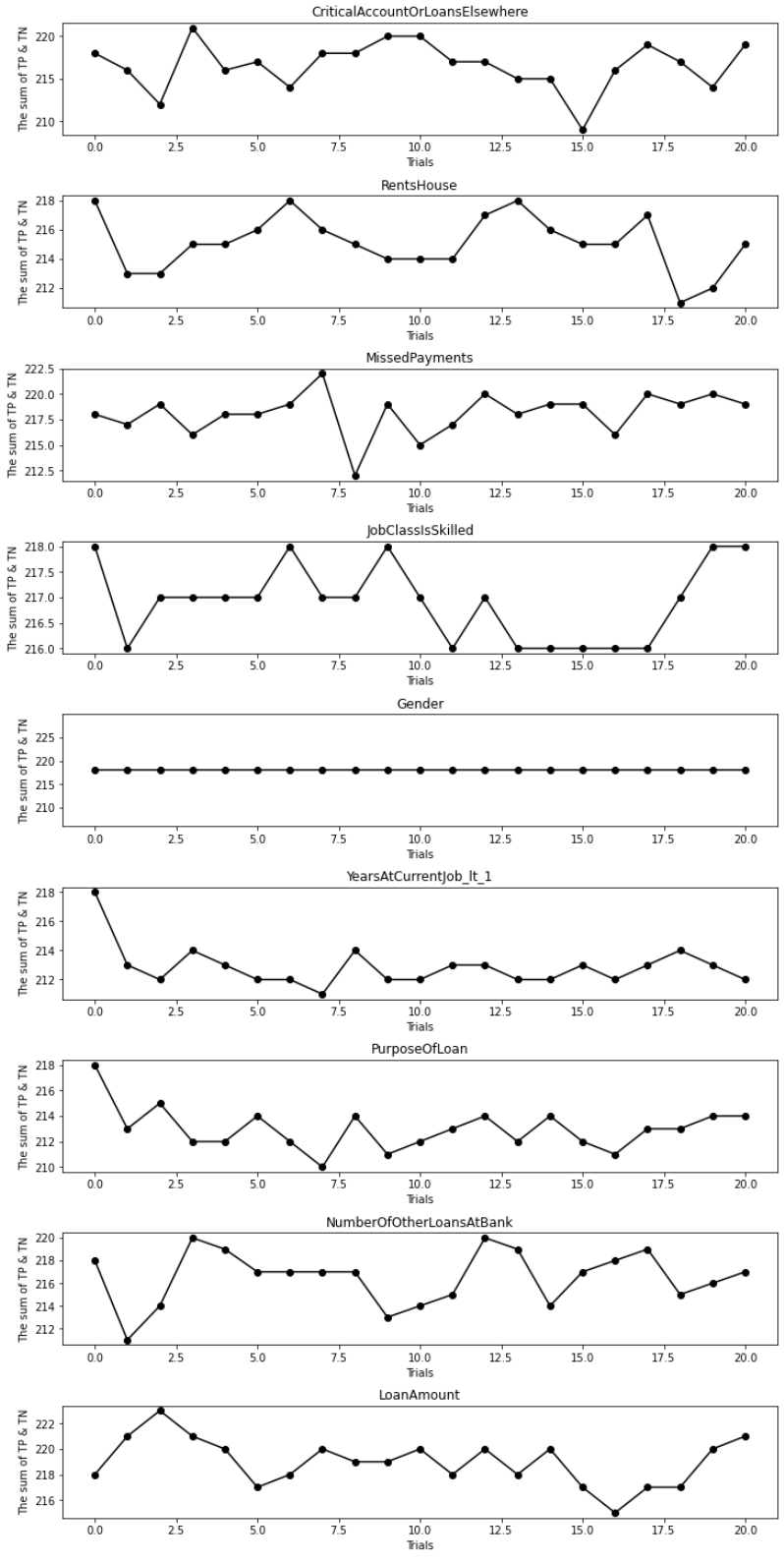


Figure 3.7: Applying a developed counterfactual method on the biased model for eight randomly selected features and the sensitive selected feature "Gender" in the middle.

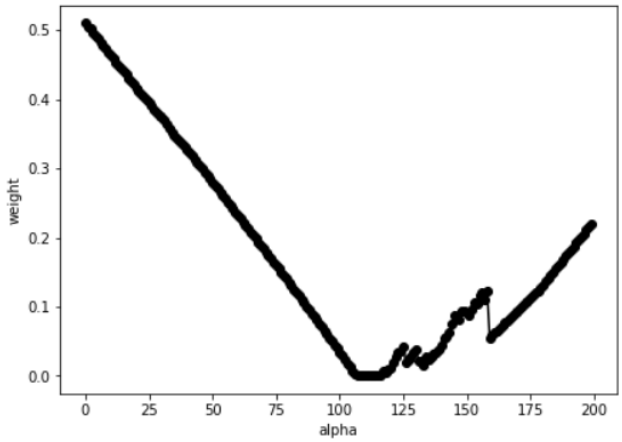


Figure 3.8: Studying the effect of alpha on the change in the value of the selected feature 'CriticalAccountOrLoansElsewhere'. The optimum alpha (α=112) corresponds to the minimum weight.

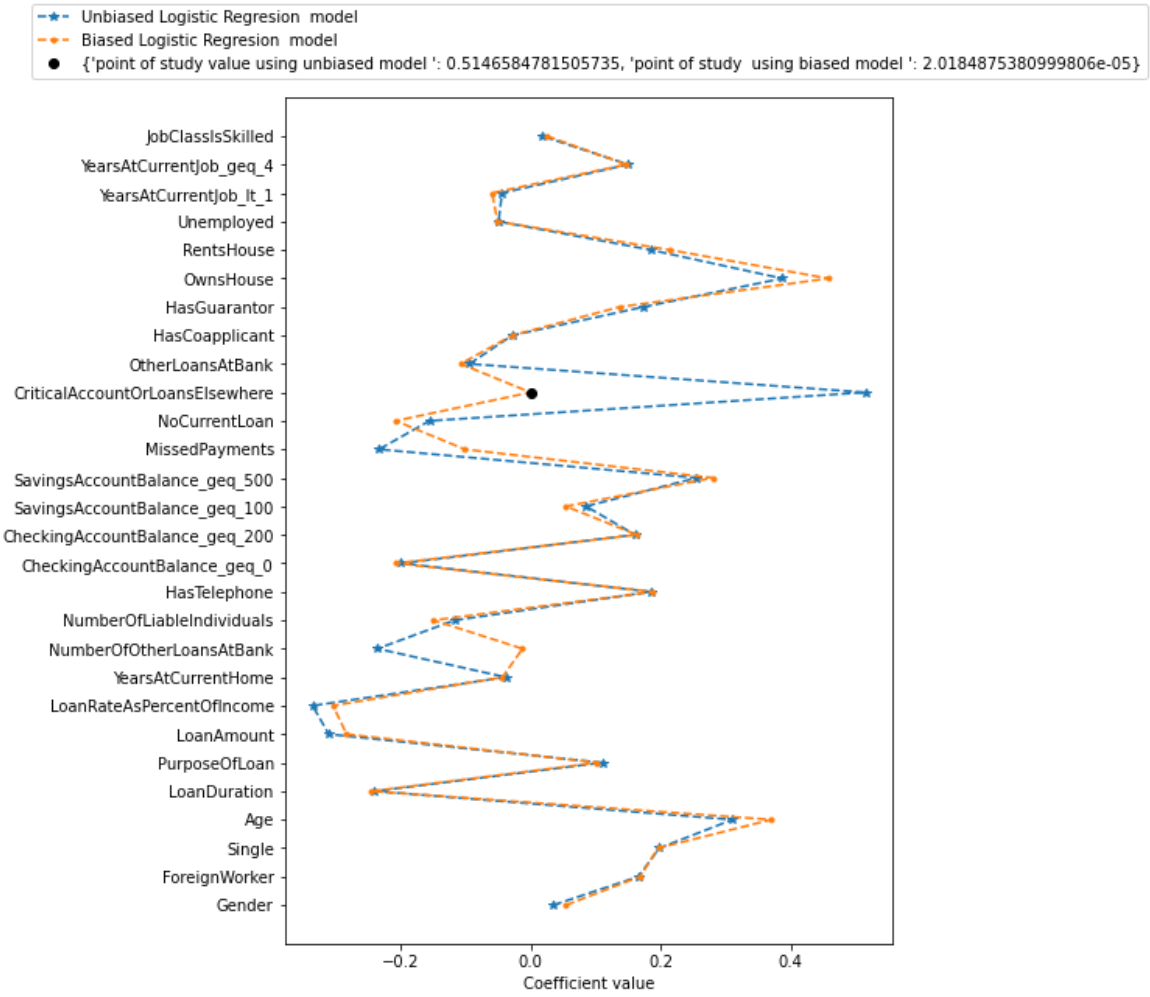


Figure 3.9: The use of a biased logistic regression model to train the German loan lending data. The selected sensitive feature is 'CriticalAccountOrLoansElsewhere'.

3.1.2. The Case For "CriticalAccountOrLoansElsewhere" as a Sensitive Feature

As the feature "Gender" has a coefficient value of insignificant effect for the LIME study, the feature "CriticalAccountOrLoansElsewhere", has been randomly selected for a similar study which has been done on the case for "Gender". In this case, the alpha has been studied as shown in Figure 3.8. This applied to the biased study for the feature "CriticalAccountOrLoansElsewhere" as shown in Figure 3.9. This figure shows the use of a biased logistic regression model to train the German loan lending data. From this figure it could be concluded that the value of the coefficient represents the feature "CriticalAccountOrLoansElsewhere" has been devaluated by several orders of magnitude compared to its value using the unbiased model, its absolute value has changed from 0.51 to 0.00002, thus it decreases by a factor of approximately 25000.

Table 3.2: Studying the effect of biased classifier on the accuracy of classification.

Point of comparison	Biased Logistic Regression Model	Unbiased Logistic Regression Model
Confusion matrix	[[29 62] [19 190]]	[[31 60] [24 185]]
Classification report	precision recall f1-score support 0 0.60 0.32 0.42 91 1 0.75 0.91 0.82 209 accuracy 0.73 300 macro avg 0.68 0.61 0.62 300 weighted avg 0.71 0.73 0.70 300	precision recall f1-score support 0 0.56 0.34 0.42 91 1 0.76 0.89 0.81 209 accuracy 0.72 300 macro avg 0.66 0.61 0.62 300 weighted avg 0.70 0.72 0.70 300

Furthermore, the accuracy of the biased classifier has been compared to the accuracy of the unbiased classifier, as shown in Table 3.2. From this table, it could be concluded that biased classification has an almost insignificant effect on classification accuracy.

The change in the importance of the feature "CriticalAccountOrLoansElsewhere" has been studied using LIME which has been applied to the whole test data composed of 30% of the data, as shown in Figure 3.3 for the unbiased model and Figure 3.10 for the biased model where the sensitive feature is "CriticalAccountOrLoansElsewhere". As shown in those two figures LIME did show significant changes to the feature "CriticalAccountOrLoansElsewhere" which is the third among the lowest important coefficients while using the biased model, while it is the second

important feature when using the unbiased model. This is due to the absolute value of the coefficient of “CriticalAccountOrLoansElsewhere” being among the highest values as shown in Figure 3.9.

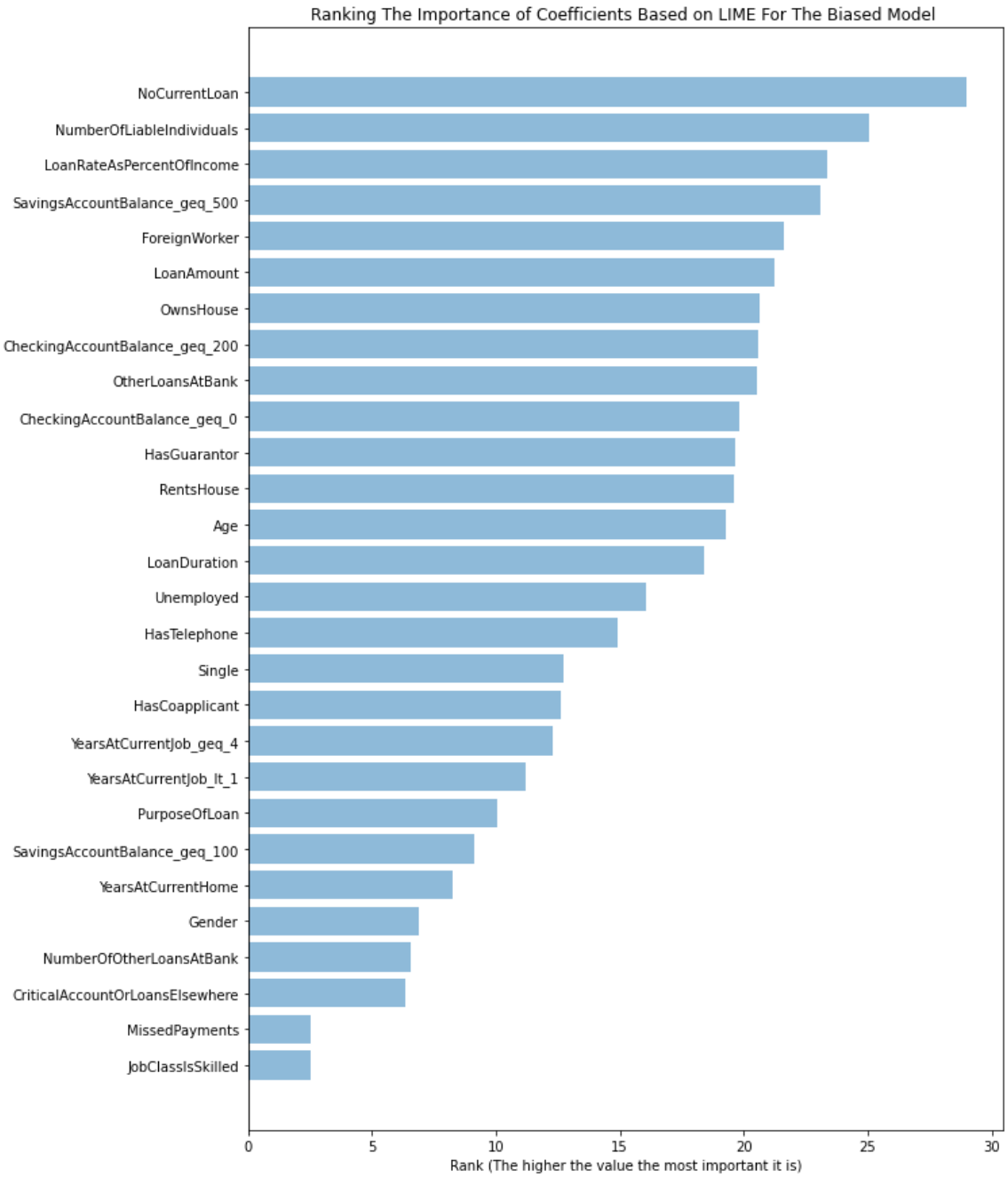


Figure 3.10: Studying the importance of the features as explained by LIME for the case of biased classifier.

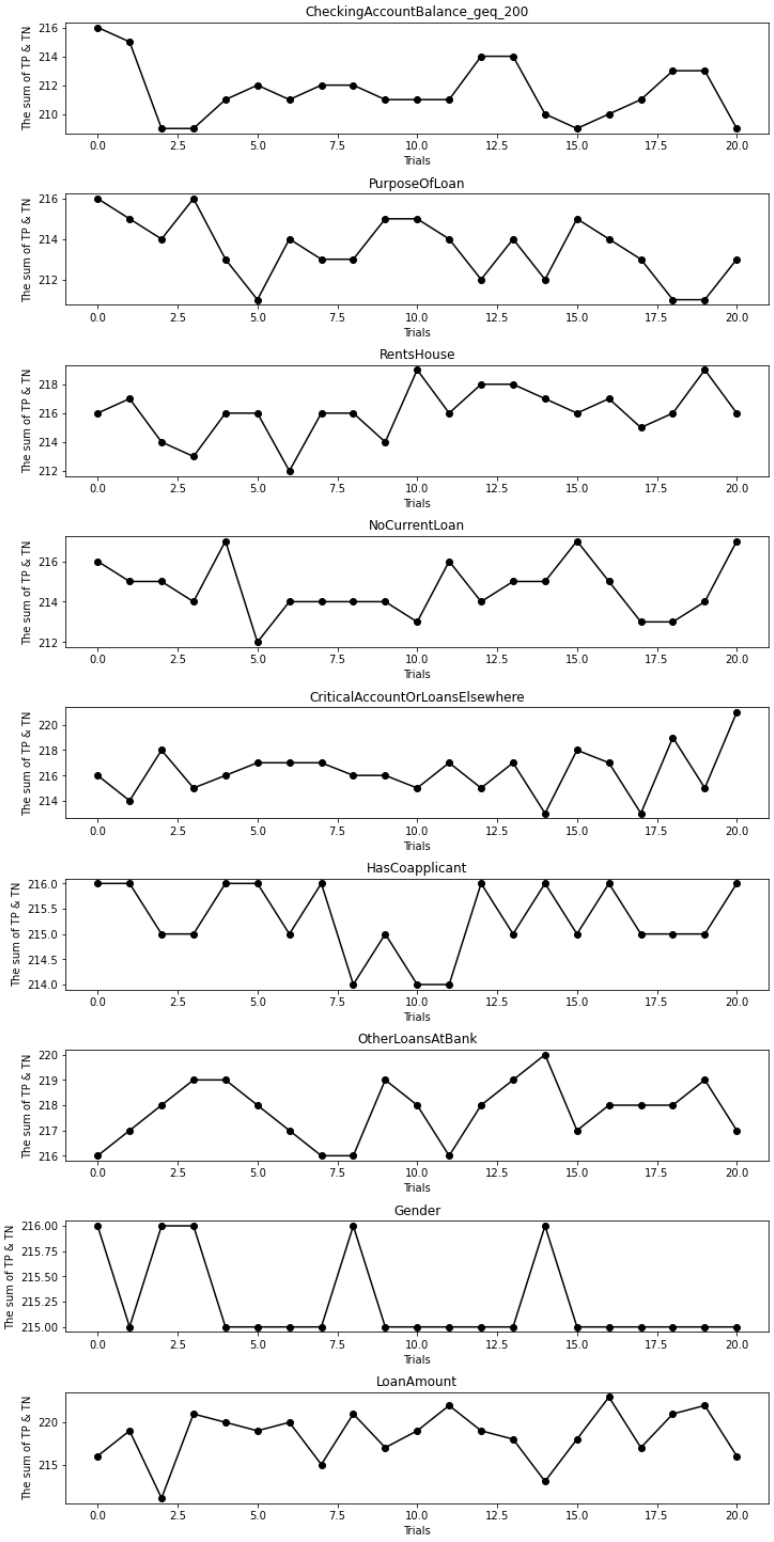


Figure 3.11: Applying a developed counterfactual method on an unbiased model for eight randomly selected features and the sensitive selected feature "CriticalAccountOrLoansElsewhere" in the middle.

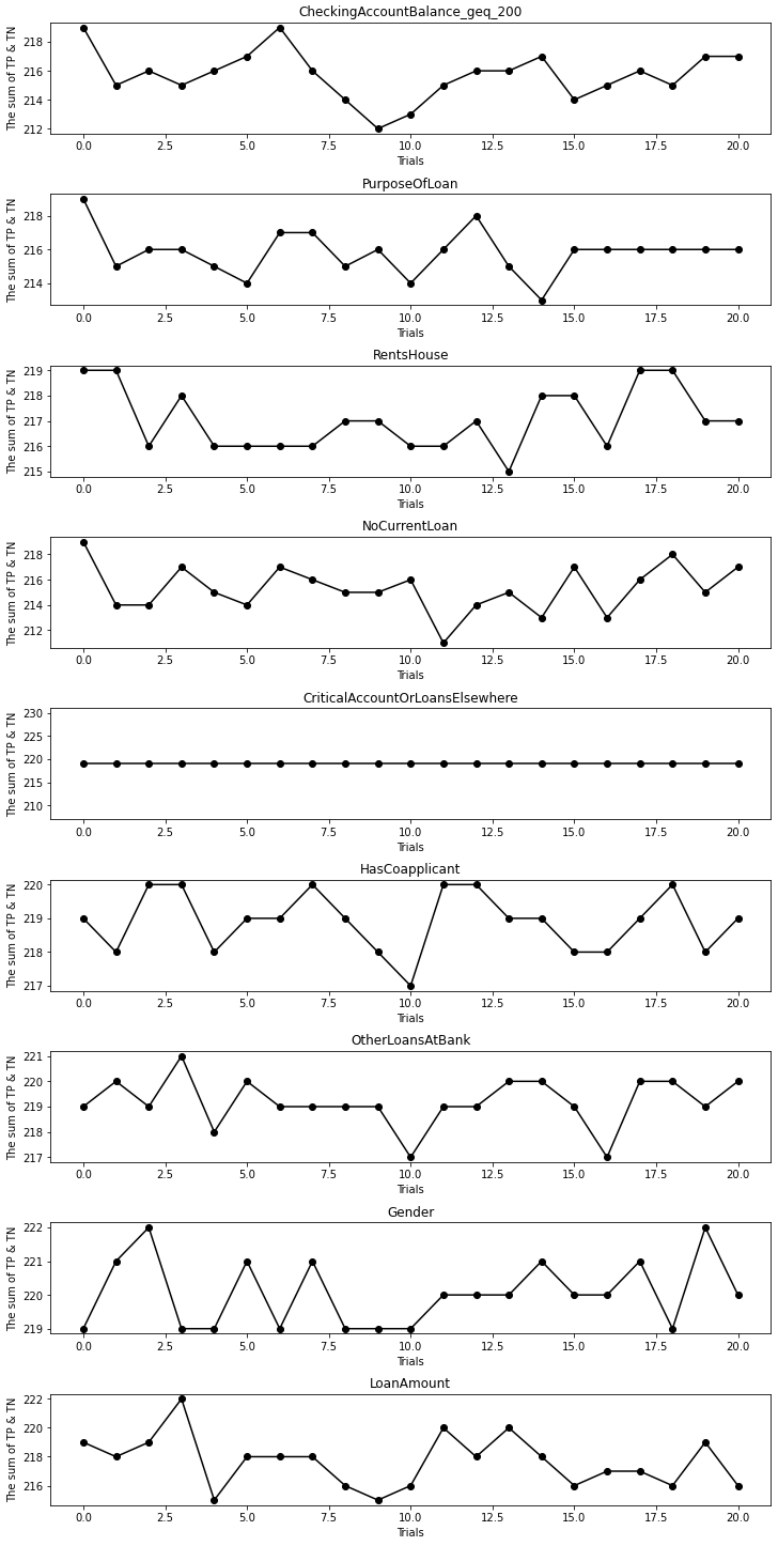


Figure 3.12: Applying a developed counterfactual method on a biased model for eight randomly selected features and the sensitive selected feature "CriticalAccountOrLoansElsewhere" in the middle.

The developed counterfactual method has been applied to the unbiased model where the sensitive feature is “CriticalAccountOrLoansElsewhere” as well as the biased model as shown in Figure 3.11 and Figure 3.12. The first point in the unbiased plot as shown in Figure 3.11 is equal to 216, and in the biased plot as shown in Figure 3.12 for a biased model is 219. This is consistent with the values in Table 3.2. For the eight randomly selected features the value for the sum of true negative and true positive fluctuates, which also fluctuates for the case of the sensitive feature “CriticalAccountOrLoansElsewhere” for the unbiased model, however, it is almost silent in the case of the biased model for the feature “CriticalAccountOrLoansElsewhere”. This method has shown that given a set of test data if a coefficient has been devaluated to almost zero, its effect is insignificant while changing the column of data corresponding to such feature.

4. Conclusion and Future Work

In this research work the unbiased algorithm and biased algorithm for logistic regression has been developed. The biased algorithm has followed the algorithm developed by (Dimanov et al., 2020). Two case studies have been selected for the “Gender” and “CriticalAccountOrLoansElsewhere”. The former case coefficient is insignificant compared to other features; thus, LIME was not able to detect any differences between the biased model and the unbiased model. For the latter case, the coefficient is significant thus the LIME was able to show a significant decrease in the importance of the latter feature coefficient value.

Moreover, LIME was not able to show that the low important features coefficient could approach zero value. Thus, a counterfactual method has been developed based on replacing the column corresponding to a feature with uniform distribution data of a specific range. This method was able to show that for the biased model applied to the cases studied, a silence behavior has been encountered as shown in Figure 3.7 and Figure 3.12.

Finally, several points should be studied in further investigation such as the fairness of the model and the investigation of this concept with relation to the numerical values obtained in this work. Also, the counterfactual method developed in this work should be further studied and investigated in comparison with existing methods like LIME and SHAP.

5. References

Antoniadi, A. M., Du, Y., Guendouz, Y., Wei, L., Mazo, C., Becker, B. A., & Mooney, C. (2021). Current challenges and future opportunities for xai in machine learning-based clinical decision support systems: A systematic review. *Applied Sciences (Switzerland)*, *11*(11). https://doi.org/10.3390/app11115088

Dimanov, B., Bhatt, U., Jamnik, M., & Weller, A. (2020). *You Shouldn't Trust Me: Learning Models Which Conceal Unfairness From Multiple Explanation Methods*.

Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow SECOND EDITION Concepts, Tools, and Techniques to Build Intelligent Systems*.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "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-August-2016*, 1135–1144. https://doi.org/10.1145/2939672.2939778

Slack, D., Hilgard, S., Jia, E., Singh, S., & Lakkaraju, H. (2020). Fooling LIME and SHAP: Adversarial attacks on post hoc explanation methods. *AIES 2020 - Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 180–186. https://doi.org/10.1145/3375627.3375830

6. List of Figures