Student Number: 244853

1. Introduction

The history of machine learning methods used for classification dates back to 1967 when Frank Rosenblatt developed a perceptron model that could classify images based on their shapes such as circles, triangles, and so on. Many machine learning algorithms have been deployed in the industry for decision-making purposes over the years. [1] Machine learning is used in a variety of applications, including email spam filtering, product recommendations, speech recognition, candidate hiring as well as credit scoring, facial recognition, self-driving vehicles, and criminal justice. These algorithms are frequently used to recognize items and classify them. Classification in machine learning systems uses a variety of approaches to classify future information into appropriate and relevant categories using these pre-categorized training datasets. However, some of these decision-making algorithms exhibit classification bias, which might have a negative impact on the decision-making method. For example, researchers discovered in October 2019 that an algorithm used on over 200 million people in US hospitals to predict which patients will likely require additional medical care strongly favored white patients over black patients. [1] While the race was not a factor in this method, another factor that was substantially linked with race was healthcare expense history. The reasoning was that a person's healthcare needs are summarized by their cost. For various reasons, black patients with the same diseases had lower healthcare costs on average than white patients with the same conditions. Fortunately, researchers collaborated with Optum to cut the level of bias by 80%. [1] However, the bias would have continued to discriminate unfairly if they had not been interrogated in the first place. Concerns regarding fairness have grown in prominence as learning models have evolved. If a machine learning forecast handles people from different groups inequitably based on sensitive characteristics like gender, color, country, or handicap, it is called unfair. The most common approach in fair machine learning is to include fairness as a constraint or penalization term in the prediction loss minimization. In supervised machine learning, there are two sources of unfairness. Machine learning predictions, for starters, are taught on data that may contain biases. As a result, standard learning processes prediction outcomes are unlikely to be fair when learning from biased or prejudiced targets. Second, even if the aims are fair, the learning process may harm fairness because machine learning's goal is to make the most accurate forecasts possible. [2]The primary objective of this study is to examine the impact of various hyperparameters on machine learning models and to compare regular machine learning models to fairness-based algorithms.

2. Datasets

Two datasets are utilized to assess the effects of regularization on accuracy and fairness. Both datasets were downloaded from the UCI Machine Learning Repository. [4] Ronny Kohavi and Barry Becker credited the adult dataset to the United States Census Bureau in 1994. Personal information such as education level is used in the dataset to forecast whether an individual would earn more or less than $50,000 per year. The dataset contains 14 variables that are a combination of category, ordinal, and numerical data types, including Age, Education, Age, Sex, Race, Occupation, etc.

The German dataset was generated by Prof. Hofmann with 1000 items and 20 categorical features. This dataset represents each person who obtains credit from a bank and is classified as having excellent or bad credit based on a set of risk criteria. The dataset includes features such as age, gender, employment status, and bank account information [4]

.

Both datasets are preprocessed in sklearn with a minmax scaler. The minmax scaler scales and translates each feature independently so that it falls inside the training set's specified range, such as zero to one. [5]

2. Classification Model

Machine learning methods include logistic regression, multilayer perceptron, and support vector machines, among others. The Support Vector Machine algorithm is utilized to classify the data in this assignment. The Support Vector Machine algorithm determines the optimal margin between classes, lowering the probability of dataset inaccuracy. [6] SVM's margin makes it more robust in approaching the target boundary. In comparison to logistic regression, the likelihood of overfitting is lower in SVM. In a multilayer perceptron, the dataset requires numerous hidden layers that control the algorithm's complexity, but in an SVM, the difficulty is independent of the dataset's dimension. [7] As a result, the assignment employs the SVM method.

3.1. Support Vector Machine

One of the most often used Supervised Learning Algorithms for classification and regression issues is the Support Vector Machine. The goal of the SVM method is to discover the best line or decision boundary for categorizing n-dimensional space into categories in the future so that fresh data points may be easily placed in the relevant category. There may be numerous lines/decision boundaries to separate the classes in n-dimensional space, but we must choose the best decision boundary to help classify the data points. The SVM hyperplane is the most optimal boundary. The data points or vectors that are closest to the hyperplane and have a major impact on the hyperplane's position are called Support Vectors. [8]

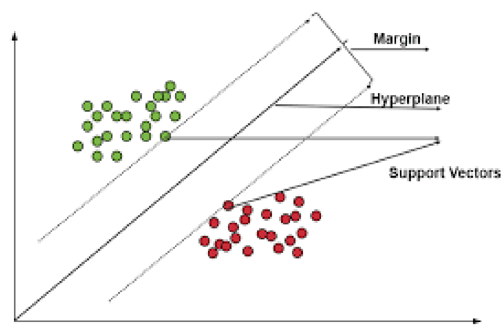

Figure 1: The SVM Model

The way in which a model handles empirical error determines its traits and performance. The loss function measures the distance between the estimated and true values. The hinge loss is a type of cost function that determines the cost depending on a margin or distance from the categorization border. Even if extra observations are correctly classified, if the margin from the decision boundary is insufficient, they may be punished. Hinge loss can be calculated with the formula - $L = max\left(0,1 - y_i\ (w^T\ x_i + b\ )\right)$

3.2. SVM Standard Model

To determine the maximum accuracy on both datasets, a Standard SVM model with k-fold cross-validation is employed in the assignment. A given data set is separated into K sections/folds, each of which is used as a testing set at some point. The model uses a 5-fold cross-validation (K=5) method. The data set is split into five sections. In the initial iteration, the first fold is used to test the model, while the others are used to train it. The second iteration employs the second fold as the testing set and the remaining folds as the training set. This procedure is continued until all five folds have been examined.

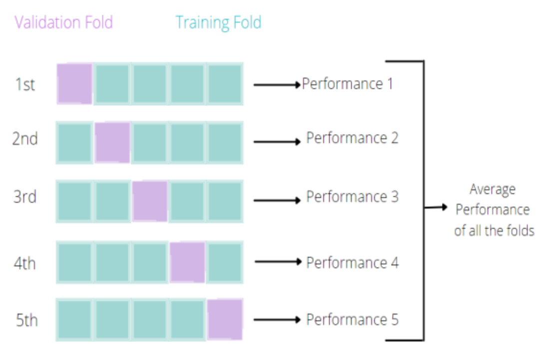

Figure 2: 5-fold cross-validation

3.3.Fairness Based Model

The SVM standard model with hyperparameter with 5-fold cross-validation and reweighing is used to remove the bias from the model. The major purpose of using a fairness-based model is to increase model accuracy while ensuring that models are less discriminatory when it comes to sensitive or protected traits. Reweighing is a simple but efficient method for reducing bias. [9]The algorithm looks at the protected attribute and the real label. If the protected attribute and y are both independent, the chance of giving a favorable label (y=1) is calculated. After that, the algorithm divides the theoretical probability by the empirical probability. These two vectors (protected variable and y) are used to construct weights vectors for each observation in the data, which are subsequently sent to the model. The model with maximum fairness is chosen. The accuracy of the model may decrease as compared to the standard SVM model, but increasing the fairness balances the performance of the model as a whole.

4. Experimental Analysis

Our task is to analyze two datasets and come up with the most accurate, most fair and most optimal models that can predict the results of that and similar other datasets. The two datasets that we will be using are ‘Adult’ and ‘German’. The algorithm we are using for our supervised learning model is the pre described SVM.

4.1. Analysis on Adult dataset

The Adult dataset has information pertaining to income of individuals. It has more than 40,000 entries and 18 features. The sensitive groups are ‘female’ and ‘male’ and are labelled 0 and 1 respectively in the initial steps of our analysis as part of preprocessing. Then we split the dataset into train and test sets in a 7:3 ratio and separate the features and labels for the next part of the analysis.

The most common way of performing hyperparameter tuning in python is by using GridSearchCV or RandomizedSearchCV functions that takes in the range of the parameters we want to vary and returns the accuracy score. Having the ability to perform k-fold cross validation is another utility of the function. But it does not give much information about fairness metrics. Therefore, we have used our own functions for cross validation and hyperparameter tuning. Throughout the task, we have studied the effect on accuracy and fairness by varying C and gamma parameters of the SVC function only. Kernel has been kept constant at ‘rbf’ throughout.

We first calculated accuracy and fairness metrics on the training set by i) Varying C and keeping gamma constant and, ii) Varying gamma keeping C constant and plotted them.

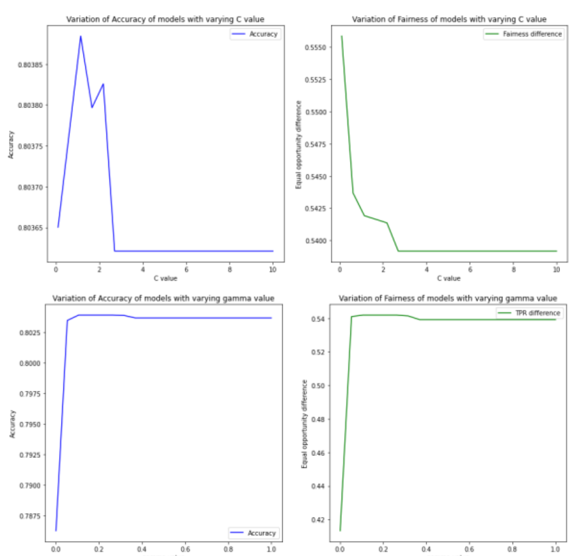

Figure 3: Accuracy and Fairness vs C value and gamma value

As we can see from the graphs, The accuracy value increases with an increasing C (for a while) and gamma. A very low value of C implies less penalties on misclassified datapoints that leads to underfitting and therefore low accuracy. Then as C increases, accuracy rises and then falls again as very high values of C implies overfitting which is also not preferrable for a good classification model. The gamma parameter is only useful for non linear classifications such as this one. It is a representation of the extent of similarity throughout the dataset. A low valued gamma denotes high similarity and therefore the decision boundary is very flexible. A moderately high value of gamma is required.

Given the discussion above, it can now be expected to see lower values of C and gamma for fairer models and comparatively higher values for accurate models, but the proper combination is derived by hyperparameter tuning.

Next, we trained our SVM model on the training set using 5-fold cross validation and this time varying both C and gamma. We treated the parameters that gave the highest accuracy value in that run as the most accurate model and the parameters that gave the lowest absolute difference as the fairest model. The “absolute value” column is a measure of the absolute difference in the true positive rate between the two sensitive classes. Values close to 0 are preferred as it denotes equal representation of both classes by the ML model.

On applying the most accurate model on the adult test set, we found a high value of equal opportunity difference or TPR difference. (Accuracy on test: 80.37%, Equal opportunity difference: **-0.437**). To minimize that, we used a fairness-based ML algorithm called ‘reweighing’ where we provide weights to each feature during the training phase. The weights are determined internally and depends upon the contribution of the said feature to the final result.

After reweighing the samples, we ran the previous task again of varying the C and gamma parameters to come up with the most accurate and fair models. Once done, we applied the most accurate model on the test set and now the equal opportunity difference was found to be much lower. Almost close to zero. Weighing the samples helped reduce the variance of the overall distribution. (Accuracy on test: 79.04%, Equal opportunity difference: **0.034**).

4.2. Analysis on German dataset

We performed the same tests as above on the German dataset as well with similar results. The German dataset is much smaller with around 1000 entries and 11 features. Computation time on this dataset was not as high as that of the adult dataset and therefore many more combinations (supplied as a range of values as compared to the discrete parameter values opted for in the adult dataset) of C – gamma pair could be tried on it.

Finally, for the models derived above, we have a rudimentary scoring system. To choose the best parameters for a model, we must first understand its requirements. If the balance is skewed toward accuracy or fairness. The combination with the lowest accuracy is given a score of 1 in the tables obtained after hyperparameter tuning and training the models, while the combination with the highest accuracy is given a score of 10. The accuracy range between the highest and lowest is divided into 10 equal sections, and the remaining models are given a score between 1 and 10 based on their accuracy. The fairness metric follows the same principle, with the lowest TPR difference receiving the highest score. At the end we just add the accuracy and fairness scores to have a rough idea about where each model stands.

	Hyperparameters	Accuracy	Equality difference	Absolute difference	Accuracy score	Fairness score	Total_score
2	C=0.1_gamma=0.1	8.037000e+11	0.556	0.556	10.0	8.0	18.0
3	C=0.1_gamma=1	8.038000e+11	0.549	0.549	10.0	8.0	18.0
5	C=1_gamma=0.01	8.027000e+11	0.538	0.538	10.0	8.0	18.0
6	C=1_gamma=0.1	8.038000e+11	0.543	0.543	10.0	8.0	18.0
7	C=1_gamma=1	8.037000e+11	0.539	0.539	10.0	8.0	18.0
9	C=10_gamma=0.01	8.030000e+11	0.531	0.531	10.0	8.0	18.0
10	C=10_gamma=0.1	8.036000e+11	0.539	0.539	10.0	8.0	18.0
11	C=10_gamma=1	8.037000e+11	0.539	0.539	10.0	8.0	18.0
1	C=0.1_gamma=0.01	7.884000e+11	0.429	0.429	7.0	10.0	17.0
4	C=1_gamma=0.001	7.863000e+11	0.413	0.413	7.0	10.0	17.0
8	C=10_gamma=0.001	7.872000e+11	0.417	0.417	7.0	10.0	17.0
0	C=0.1_gamma=0.001	7.594000e+11	1.000	1.000	1.0	1.0	2.0

Figure 4: Hyperparameter combinations sorted by descending order of scores.

Here are the summarized results of each model derived

	Adult		German	
	Not weighed	Weighed	Not weighed	Weighed
Most Accurate	Hyperparams: C=1_gamma=0.1 Accuracy: 80.37% Fairness: -0.437	Hyperparams: C=10_gamma=0.01 Accuracy: 79.04% Fairness: 0.034	Hyperparams: C=1_gamma=0.1 Accuracy: 71.67% Fairness: -0.067	Hyperparams: C=0.2894_gamma=0.5267 Accuracy: 69.67% Fairness: 0.02
Most Fair	Hyperparams: C=1_gamma=0.001 Accuracy: 78.74% Fairness: 0.033	Hyperparams: C=1_gamma=0.001 Accuracy: 78.74% Fairness: 0.033	Hyperparams: C=1_gamma=0.001 Accuracy: 70.33% Fairness: 0.0	Hyperparams: C=1_gamma=0.001 Accuracy: 70.33% Fairness: 0.0
Most Optimal	Hyperparams: C=0.1_gamma=0.1 Accuracy: 80.47% Fairness: -0.429	Hyperparams: C=0.1_gamma=0.01 Accuracy: 78.78% Fairness: 0.035	Hyperparams: C=C=0.6_gamma=0.889 Accuracy: 71.67% Fairness: -0.067	Hyperparams: C=0.2894_gamma=0.5793 Accuracy: 69.67% Fairness: 0.02

5. Conclusion

We studied the effects of hyperparameter tuning on the accuracy and fairness metrics of a support vector machine model to find out different combinations for different results and applied them to two datasets, namely, Adult and German. Then we weighed our training sample data to minimize variance and bias for fairer models. Finally, we suggested selection of the most optimal ML classification model parameters based on some scoring system. Ultimately, it is up to the requirements of the task and the sample datasets that determine the most optimal parameters for the best model.

1. Bibliography

[1] B. Marr, "A Short History of Machine Learning," forbes, 19 February 2016. [Online]. Available: https://www.forbes.com/sites/bernardmarr/2016/02/19/a-short-history-of-machine-learning-every-manager-should-read/?sh=d57861d15e78.

[2] T. Shin, "Real-life Examples of Discriminating Artificial Intelligence," June 2020. [Online]. Available: https://towardsdatascience.com/real-life-examples-of-discriminating-artificial-intelligence-cae395a90070.

[3] F. M. N. S. K. L. A. G. NINAREH MEHRABI, "A Survey on Bias and Fairness in Machine Learning," *arXiv,* 2022.

[4] D. ,. G. C. Dua, "UCI Machine Learning Repository," University of California, Irvine, School of Information and Computer Sciences, 2017. [Online]. Available: https://archive.ics.uci.edu/.

[5] F. Pedregosa, G. Varoquaux, A. Gramfort and V. Michel, "Scikit-learn: Machine Learning in {P}ython," *Journal of Machine Learning Research,* vol. 12, pp. 2825--2830, 2011.

[6] P. Bassey, "Logistic Regression Vs Support Vector Machines," September 2019. [Online]. Available: https://medium.com/axum-labs/logistic-regression-vs-support-vector-machines-svm-c335610a3d16#:.

[7] E. &. S. A. Frias-Martinez and J. Vélez, " Support vector machines versus multi-layer perceptrons for efficient off-line signature recognition," *ResearchGate,* 2006.

[8] W. S. Noble, "What is a support vector machine?," *Nature Biotechnology,* vol. 24, 2006.

[9] J. Wiśniewski, "fairmodels: let’s fight with biased Machine Learning models," Agust 2020. [Online]. Available: https://towardsdatascience.com/fairmodels-lets-fight-with-biased-machine-learning-models-f7d66a2287fc#:~:text=Reweighting%20is%20a%20simple%20but,attribute%20and%20y%20are%20independent..