1. Introduction

Machine learning algorithms play a significant role in decision-making in today's world. The application of machine learning ranges from email spam filtering, product recommendations, speech recognition, and image recognition to credit scoring, facial recognition self-driving vehicles, and criminal justice. A common use of machine learning algorithms is to recognize objects and separate them into categories. This process is called classification. With the use of these pre-categorized training datasets, classification in machine learning systems employs a wide range of methods to classify future datasets into corresponding and relevant categories. However, some of these decision-making algorithms show bias towards certain people and can adversely affect the decision-making strategy. For example, in 2015, Amazon discovered that the algorithm they used to hire staff was skewed against women. The explanation behind this was that the algorithm was trained to favor men over women based on the number of resumes submitted over the previous 10 years and the majority of the candidates were men [1]. As learning models have become more advanced, concerns about fairness became more prominent. There are many techniques in machine learning to eliminate the bias in the model and generate a model which gives predictions that are fair and accurate. The most prevalent technique in fair machine learning is to integrate fairness as a constraint or penalization term in the prediction loss minimization, which eventually limits the information provided to decision-makers.

The main purpose of this project is to study the effect of regularization on the accuracy-fairness trade-off. This project will investigate the inherent bias of algorithms as well as evaluate fairness approaches devised to reduce prejudice, with an emphasis on accuracy deterioration, if any.

2. Datasets

For evaluating the effects of regularization on accuracy and fairness two datasets namely Bank Marketing and Adult datasets are used. Both the datasets are taken from UCI Machine Learning Repository[1]. The Bank Marketing dataset was drawn in June 2014. The information in the data relates to direct marketing activities run by a Portuguese bank. The dataset has 21 variables that are a mixture of categorical, ordinal, and numerical data types such as age, education, loan, housing, job, etc. There are a total of 41,1161 rows of data, and 10,614 with missing values, leaving 30,547 complete rows. There are two class values, where age > =25 and age <25, meaning it is a binary classification task. Each entry in the dataset represents a person and if the person has subscribed ('yes') or not ('no') to a bank term deposit.

The Adult dataset is drawn from the 1994 United States Census Bureau credited to Ronny Kohavi and Barry Becker. The dataset involves personal details such as education level to predict whether an individual will earn more or less than $50,000 per year [1]. The dataset provides 14 variables that are a mixture of categorical, ordinal, and numerical data types such as Age, Education, Age, Sex, Race, Occupation, etc., There are a total of 48,842 rows of data, and 3,620 with missing values, leaving 45,222 complete rows. There are two class values, where income > 50 K and income <=50 K[1], meaning it is a binary classification task.

3. Methodology and Background

3.1. Classification Model

There are various classification algorithms in machine learning like logistic regression, multilayer perceptron, support vector machines, etc. In this assignment, the Support Vector Machine algorithm is used for classification. Support Vector Machine algorithm finds the best margin that separates the classes, thus reducing the risk of error on the dataset [1]. The margin of SVM makes it more robust in getting closer to the target boundary. The risk of overfitting in SVM is less as compared to logistic regression [2] In multilayer perceptron, the dataset needs multiple hidden layers which control the complexity of the algorithm, whereas in SVM the complexity does not depend on the dimension of the dataset [4]. Therefore, the SVM algorithm is used in the assignment.

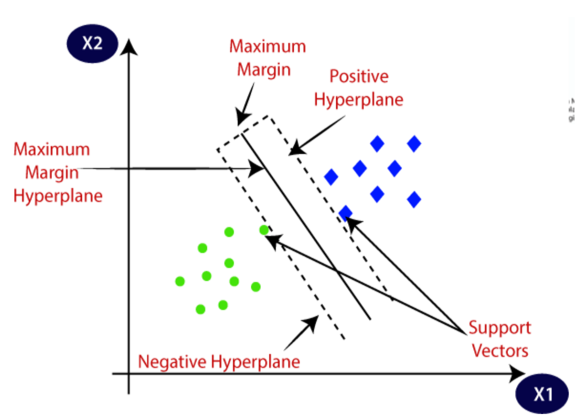

Figure 1: The SVM Model

The goal of the SVM method is to discover the best line or decision boundary for categorizing n-dimensional space into categories so that additional data points can be easily placed in the right category in the future. 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 optimal boundary is known as the SVM hyperplane. Support Vectors are the data points or vectors that are closest to the hyperplane and have a significant impact on the hyperplane's position. These vectors are called Support vectors because they support the hyperplane.

The characteristics and performance of a model depend on the way in which the model handles the empirical error. The loss function shows how far the estimated value is from its true value. In SVM the loss is calculated with the hinge loss function. The hinge loss is a form of a cost function that calculates the cost based on a margin or distance from the classification boundary. Even if additional observations are accurately classified, they may be penalized if the margin from the decision border is insufficient. The hinge loss is given by the formula, $L = \max(0, 1 - y_i(w^T x_i + b))$, [1] where 0 is for correct classification and 1 is for the wrong classification. In the assignment, a Standard SVM model with 5-fold cross-validation is used to determine the maximum accuracy and fairness of both datasets The scikit-learn library is used for the SVM model. Each section/fold of a given dataset is used as a testing set at some stage There are five sections to the data set. The first fold is used to test the model in the first iteration, while the others are used to train it. In the second iteration, the second fold is used as the testing set, while the remaining folds are used as the training set. This process is repeated until each of the five folds has been evaluated.

3.2 Fairness-Based Model

Unfairness can be either direct or indirect. Direct unfairness happens when a protected trait causes an adverse result directly, whereas indirect unfairness results from other factors that could be used to proxy the protected characteristic. There are two sources of unfairness in supervised machine learning [1] [3] To begin with, machine learning predictions are trained on data that may have inherent biases. As a result, by learning from biased or prejudiced targets, typical learning procedures' prediction outcomes are unlikely to be fair. Second, even if the targets are fair, the learning process may compromise fairness because the goal of machine learning is to create the most accurate predictions. The fact that models are essentially based on data means that they will generally reflect the biases found in the data, very often increasing them [1]. An important aspect influencing data bias is sample or representation bias: this occurs when a dataset is not representative of an entire co mmunity.

Even though biases are a problem, they could be mitigated or even eliminated. There are various ways to evaluate how fair a model is, but in this investigation, the "Equality of Opportunity Difference" is the central focus. This measure examines the ratio of groups that were positively categorized as protected versus unprotected. when everyone has the same opportunities. Fairness algorithms can aid in the reduction of bias during pre-processing, in-processing, or post-processing. There are many fairness algorithms that can be used for reducing bias like Adversarial debiasing, and Reweighing. In this investigation, Reweighing is utilized to mitigate the bias.

To reduce the bias from the model, the SVM standard model with hyperparameter, 5-fold cross-validation, and reweighing is utilized. The main goal of employing a fairness-based model is to improve model accuracy while reducing discrimination when it comes to sensitive or protected features. [3] Reweighing is a straightforward yet effective method of bias reduction. The protected attribute and the real label are examined by the algorithm. The chance of delivering a favorable label (y=1) is determined if the protected attribute and y are both independent. Following that, the algorithm splits the theoretical and empirical probabilities. These two vectors (protected variable and y) are used to build weight vectors for each observation in the data, which are then fed into the model. The model with the highest degree of fairness is picked. When compared to a regular SVM model, the model's accuracy may suffer, but boosting fairness improves the model's overall performance [3].

4. Investigation

The Support vector machine model is used with 5-fold cross-validation for all three tasks. For hyperparameter tuning with 5-fold cross-validation, a function is created instead of using GridSearchCV or RandomizedSearchCV as the results of fairness and bias from these libraries are not comprehensive enough. Both the datasets, i.e., Bank Marketing and Adult are normalized using scikit-learn's SandardScaler [9]and split into training and testing sets with a 70:30 ratio.

The regularization hyper-parameters, 'C' and 'gamma' were selected from a number of options using 5-fold cross-validation to verify the results. The values of hyperparameters were selected using np.linspace().The values of 'C' were given from a total of 20 equidistant values within the range C=0.01 to C=100 and values of 'gamma' were given from a total of 20 equidistant values within the range gamma=0.001 to gamma = 10. The hyperparameter ‘kernel’ was kept to ‘rbf’ as varying kernels did not give different results and ‘rbf’ is the most optimal. The optimum result for the most accurate model and most fair model was then evaluated for the entire test set of each dataset.

Three major tasks are performed on the SVM model to find out the model with optimal accuracy, fairness, and optimal model with best accuracy and fairness. Both the datasets Bank Marketing and Adult were used to analyze the results. The results of the standard machine learning model described in section 3.1 are compared with the fairness-based machine learning model described in and 3.2.

4.1. Task 1: SVM with 5-fold cross-validation

The task 1, a simple SVM classification model with 5-fold cross-validation is used while varying hyperparameters ‘C’ and ‘gamma’. The thorough check by providing a range of values for ‘C’ and ‘gamma’ increased computation time by a lot with no real improvement in results. Therefore, to take out the optimal values of hyperparameters for both the datasets first the hyperparameter ‘C’ was varied keeping ‘gamma’ constant’ and in the next run hyperparameter, ‘gamma’ was varied keeping ‘C’ constant. The results of the accuracy and fairness with respect to hyperparameters is shown in figures 2 ,3, 4, and 5.

A few selected values of C and gamma were provided to the SVM model, where C = 0.001, 0.01, 0.1, 1, 10 and 100 and gamma = 0.001, 0.01, 0.1, 1, and 10. Table 1 displays the results of the most accurate and fair models on training and testing sets for both datasets.

Dataset	Model	C	gamma	Train(A)	Train(F)	Test(A)	Test(F)
Bank	Most Accurate	1	0.01	89.89	0.6008	89.49	0.045
Bank	Most fair	10	0.01	89.85	0.5148	89.13	-0.009
Adult	Most Accurate	0.1	0.1	80.38	0.5634	80.38	-0.449
Adult	Most fair	1	0.001	78.69	0.4307	78.59	-0.22

Table 1: Task 1 models results.
A = Accuracy, F = Fairness

The bank Marketing dataset is already quite fair as compared to the Adult dataset. Although, when evaluating on the test set both the models for the bank dataset showed almost the same accuracy, interestingly the fairness was lower in the second (most Fair) model. For the Adult dataset, the test result shows some difference in fairness as compared to the cross-validation results. The fairer model of banks, as well as the Adult dataset, showed a better TRPD in comparison with training models. The plots of hyperparameter variations with respect to hyperparameters and dataset are available in Figures 2, 3 ,4 and 5.

4.2. Task 2: Fairness Model

As discussed in section 3.3, the reweighing algorithm was applied during the preprocessing stage to improve the fairness of the models. The original training and testing data for both the datasets were passed to the algorithm and the output of the algorithm with the weights generated in the process was then passed to the SVM classifier with 5-fold cross-validation. The results obtained for task 2 are shown in table 2 below

Dataset	Model	C	gamma	Train(A)	Train(F)	Test(A)	Test(F)
Bank	Most Accurate	1	0.01	89.89	0.5921	89.43	-0.209
Bank	Most fair	100	0.01	89.87	0.5125	89.06	-0.144
Adult	Most Accurate	0.1	1	79.06	0.4815	78.68	-0.188
Adult	Most fair	1	0.001	78.69	0.4187	78.59	-0.022

Table 2: Task 2 Fairness Algorithm Results
A=Accuracy, F=Fairness

In the bank dataset, after applying reweighing interestingly the accuracy did not change to a great extent. There is only a little drop in accuracy but the reweighing made both the models quite fair as compared to the models in Task 1. Whereas, the adult dataset showed a significant decrease in accuracy and an increase in fairness in both the models, improvements in fairness were observed in both the datasets and both the models (Most Accurate and Most Fair). Figures 2, 3 ,4 and 5. shows the variations of cross-validations.

4.3 Task 3: Model Selection Strategy

In this task, we have to create a strategy that will choose the best model that accounts for both accuracy as well as fairness. For the selection process, a scoring system is created. To come up with the most optimal hyperparameters, it is necessary to understand if the model is more skewed towards accuracy or fairness. To evaluate the hyperparameters, a scoring system is created. The results in after Task 1 and Task 2 were transformed into a table. The table includes the Hyperparameters, accuracy score and fairness scores. For Giving the scores to accuracy all the combinations of hyperparameters are taken into consideration with their accuracy scores. The model with highest accuracy is given a score of 10. The range between highest and lowest accuracy is split into 10 equal parts and the remaining models get the score between 1 and 10 depending on the accuracy. For the fairness metric, the model with the lowest TPR difference is given the score of 10. Again, the range between lowest and highest fairness metric is divided into 10 equal parts and the remaining model get a score between 1 to 10. After calculating all the scores, the scores of accuracies and fairness are simply added together to give a total score. The scoring system generated is then transformed into DataFrame for better visualizations. The DataFrames for both the datasets and both the models can be seen in Figures 6,7,8 and 9. The table below shows the results of applying scoring system to both datasets.

Dataset	Model	C	gamma	Train(A)	Train(F)	Test(A)	Test(F)
Bank	Most Accurate	10	0.01	89.85	0.515	89.13	-0.009
Bank	Most Fair	10	0.01	89.87	0.513	89.13	-0.009
Adult	Most Accurate	1	1	80.3	0.524	80.28	-0.495
Adult	Most Fair	0.1	0.01	78.79	0.43	78.66	-0.017

Table 3: Task 3 – Model Selection Strategy
A = Accuracy, F = Fairness

From the above we can see that the bank dataset shows exactly same accuracy and fairness for both the models. But interestingly when we compare the results of bank dataset models with Task 1 and Task 2 (Table 1 and table 2), we can see that the hyperparameters the model selection strategy selected are a combination of hyperparameters of Task 1 and Task 2 models. We can also see that the fairness looks quite balanced. For the Adult dataset shows similar results on accuracy but fairness metrics is better than task 2. This shows that by utilizing the combinations from Task 1 and Task 2, we can create a fairer model with good accuracy.

5. Evaluation

Comparison between task 1, task 2, and task 3 from the tables 1 and 2 in section 4, clearly shows that by using reweighing the fairness of the models increased to a great extent. Referring to tables 1, 2, and 3,in comparison of optimum values of accuracy and fairness, the accuracy remains unchanged in the bank dataset, but after applying reweighing algorithm there is a decrease of 0.43 % on the test accuracy. For the fairest models, the accuracy for the adults dataset also remains nearly the same, but there is a 26% increase in fairness on the test dataset.

The hyperparameter C adds a penalty for the misclassified data points. Increasing the hyperparameter C would means a higher penalty is imposed on the misclassified data points, which in turn increases the accuracy of the model. But higher values of C may lead to overfitting of data and thus we can see a decrease in accuracy [7]. The hyperparameter ‘gamma’ controls the influence of a single datapoint in the dataset. Low values of gamma means a high similarity and hence large groups of data points can be grouped together. As the value of gamma increases, the data points should be very similar to each other so that they can be considered in the same group. Large gamma values can also lead to overfitting of datapoints. This illustration of variation of hyperparameters can be seen in Figures 2,3,4,5. From the analysis carried in task 1, 2 and 3, it can be seen that the optimal hyperparameters for this dataset are 0.001<gamma<10 and 0.1< C <100. The evaluation on tasks 1,2 and 3 suggests that, there is a accuracy-fairness tradeoff if we want the model to be optimal and fair. For all the models, by trying to eliminate the bias, the accuracy suffered. Even though the decrease in accuracy in Bank dataset is not to a great extent, but the adult dataset shows a significant decrease in accuracy. The tradeoff also depends on how balanced the dataset. In this case, the Bank data set is more balanced as compared to adult dataset. And therefore, the trade-off between accuracy and fairness is visible.

6. Conclusion

In this study, we investigated the impact of hyperparameter tuning as well as fairness-based algorithm reweighing on the SVM model with 5-fold cross validation. Throughout the investigation it can be seen that higher values of C gives more accurate results but decreases the fairness. The lower value of gamma are useful in categorizing the datapoints into similar groups and as the gamma value increases, the model tends to get overfitted. Thus the optimal values for C and gamma for both the datasets are 0.01< C<100 and 0.001<gamma<10. The instigation’s main focus was to analyze the tradeoff between accuracy and fairness. It can be clearly seen from the Adult dataset that as the fairness increases, the accuracy decreases. In contrary to Adult dataset, the accuracy of Bank dataset was just decreased by few points. Reweighing increased the fairness in both the datasets to a good extent. From the above results it can be concluded that there is a tradeoff between accuracy-fairness in machine learning algorithms, but the tradeoff depends on the dataset and how balanced or imbalanced the dataset is. The more the data is imbalanced, the more is the trade-off. Eliminating bias from the models is very crucial task and we have various algorithms in machine learning to reduce it. The tradeoff most certainly depends on the dataset and hence unless we completely eliminate bias in society, the datasets will be imbalanced and hence the algorithms would not be able to completely eliminate the bias.

7. Future Work

In this work, the SVM model was used to study the classification o features. Although the model performed efficiently, there is still a need for improvement to get the optimal classification results. Future work on this model can include the study of feature selection using PCA, this will only select the features that are necessary for making predictions. Also comparing and combining SVM with other learning methods like deep learning and perceptron might increase the efficiency of the model.

1. References

[1]	T. Shin, "Towards Data Science," June 2020. [Online]. Available: https://towardsdatascience.com/real-life-examples-of-discriminating-artificial-intelligence-cae395a90070.
[2]	D. ,. G. C. Dua, "UCI} Machine Learning Repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml.
[3]	P. Bassey, "Medium," September 2019. [Online]. Available: https://medium.com/axum-labs/logistic-regression-vs-support-vector-machines-svm-c335610a3d16#:~:text=Difference%20between%20SVM%20and%20Logistic%20Regression&text=SVM%20works%20well%20with%20unstructured,is%20based%20on%20statistical%20approaches.
[4]	E. MartinezA, Sanchez and J. Velez, "Support vector machines versus multi-layer perceptrons for efficient off-line signature recognition," *ScieneceDirect*, vol. 19, September 2006.
[5]	F. M. N. S. K. L. A. G. Ninareh Mehrabi, "A Survey on Bias and Fairness in Machine Learning," *Arxiv*, 2019.
[6]	D. W. H. Y. P.-Y. C. S. L. K. R. V. Sanghamitra Dutta, "Is There a Trade-Off Between Fairness and Accuracy? A Perspective Using Mismatched Hypothesis Testing," *Arvix*, vol. v2, 2019.
[7]	V. Aliyev, "A definitive explanation to the Hinge Loss for Support Vector Machines.," November 2020. [Online]. Available: https://towardsdatascience.com/a-definitive-explanation-to-hinge-loss-for-support-vector-machines-ab6d8d3178f1.
[8]	M. K. Jake Lever and N. Altman, "Model selection and overfitting," *nature methods*, 2016.
[9]	G. V. ,. A. G. Fabian Pedregosa and B. T. Vincent Michel, "Scikit-learn: Machine Learning in Python".

Appendix

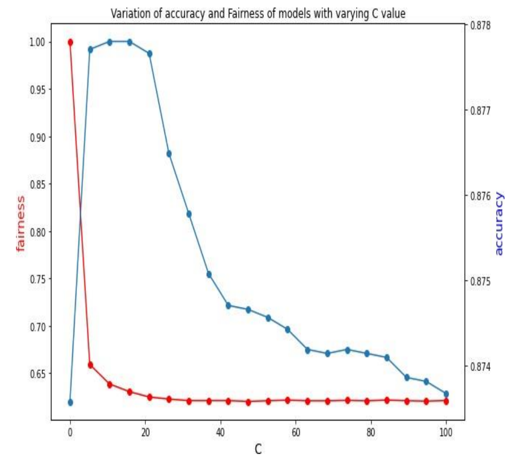

Figure 2: Task 1 Bank Accuracy Fairness with variations in C

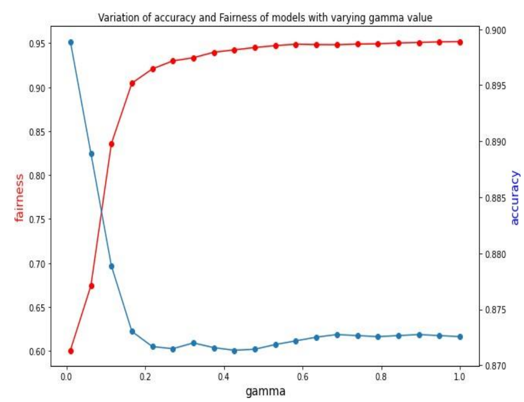

Figure 3: Task 1 Bank Accuracy Fairness with variations in gamma

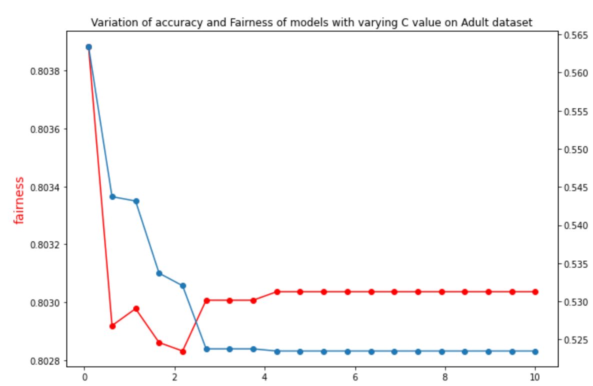

Figure 4: Task 1: Adult Accuracy Fairness with variations in C

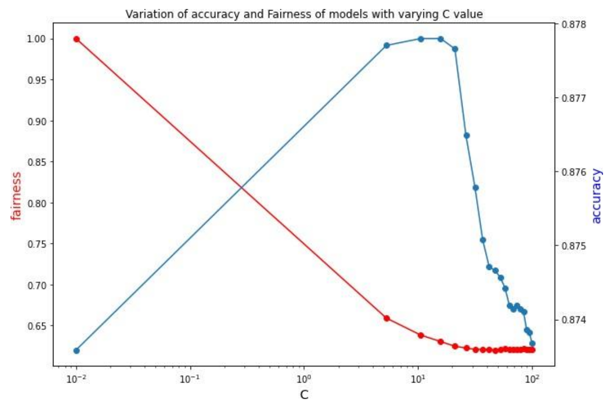

Figure 5: Task 1 Adult accuracy and fairness with variations in C

	Hyperparameters	Accuracy	Equality difference	Absolute difference	Accuracy score	Fairness score	Total_score
1	C=0.1_gamma=0.01	78.79	0.430	0.430	10.0	10.0	20.0
2	C=0.1_gamma=0.1	78.99	0.473	0.473	10.0	10.0	20.0
5	C=1_gamma=0.01	78.80	0.431	0.431	10.0	10.0	20.0
6	C=1_gamma=0.1	78.98	0.462	0.462	10.0	10.0	20.0
7	C=1_gamma=1	78.95	0.461	0.461	10.0	10.0	20.0
9	C=10_gamma=0.01	78.98	0.465	0.465	10.0	10.0	20.0
10	C=10_gamma=0.1	78.86	0.459	0.459	10.0	10.0	20.0
11	C=10_gamma=1	78.86	0.459	0.459	10.0	10.0	20.0
12	C=100_gamma=0.001	78.80	0.430	0.430	10.0	10.0	20.0
13	C=100_gamma=0.01	79.02	0.459	0.459	10.0	10.0	20.0
14	C=100_gamma=0.1	78.86	0.459	0.459	10.0	10.0	20.0
15	C=100_gamma=1	78.86	0.459	0.459	10.0	10.0	20.0
3	C=0.1_gamma=1	79.06	0.482	0.482	10.0	9.0	19.0
4	C=1_gamma=0.001	78.69	0.419	0.419	9.0	10.0	19.0
8	C=10_gamma=0.001	78.69	0.419	0.419	9.0	10.0	19.0
0	C=0.1_gamma=0.001	76.06	1.000	1.000	1.0	1.0	2.0

Figure 6: Task 3 Model selection Strategy for SVM with 5-folds CV for bank dataset

	Hyperparameters	Accuracy	Equality difference	Absolute difference	Accuracy score	Fairness score	Total_score
13	C=10_gamma=0.01	89.87	0.513	0.513	10.0	10.0	20.0
9	C=1_gamma=0.01	89.89	0.599	0.599	10.0	9.0	19.0
12	C=10_gamma=0.001	89.81	0.632	0.632	10.0	8.0	18.0
8	C=1_gamma=0.001	88.95	0.729	0.729	8.0	6.0	14.0
4	C=0.1_gamma=0.001	88.70	0.779	0.779	7.0	5.0	12.0
5	C=0.1_gamma=0.01	88.69	0.775	0.775	7.0	5.0	12.0
14	C=10_gamma=0.1	87.76	0.637	0.637	4.0	8.0	12.0
10	C=1_gamma=0.1	88.13	0.801	0.801	5.0	5.0	10.0
0	C=0.01_gamma=0.001	87.36	1.000	1.000	3.0	1.0	4.0
1	C=0.01_gamma=0.01	87.36	1.000	1.000	3.0	1.0	4.0
2	C=0.01_gamma=0.1	87.36	1.000	1.000	3.0	1.0	4.0
3	C=0.01_gamma=1	87.36	1.000	1.000	3.0	1.0	4.0
6	C=0.1_gamma=0.1	87.36	1.000	1.000	3.0	1.0	4.0
7	C=0.1_gamma=1	87.36	1.000	1.000	3.0	1.0	4.0
15	C=10_gamma=1	86.60	0.887	0.887	1.0	3.0	4.0
11	C=1_gamma=1	87.25	0.953	0.953	2.0	1.0	3.0

Figure 7 Task 3 Model selection Strategy for reweighted SVM with 5 folds for bank dataset

	Hyperparameters	Accuracy	Equality difference	Absolute difference	Accuracy score	Fairness score	Total_score
7	C=1_gamma=1	80.30	0.524	0.524	10.0	9.0	19.0
10	C=10_gamma=0.1	80.30	0.523	0.523	10.0	9.0	19.0
11	C=10_gamma=1	80.30	0.524	0.524	10.0	9.0	19.0
2	C=0.1_gamma=0.1	80.39	0.563	0.563	10.0	8.0	18.0
3	C=0.1_gamma=1	80.32	0.553	0.553	10.0	8.0	18.0
5	C=1_gamma=0.01	80.30	0.543	0.543	10.0	8.0	18.0
6	C=1_gamma=0.1	80.30	0.543	0.543	10.0	8.0	18.0
9	C=10_gamma=0.01	80.31	0.544	0.544	10.0	8.0	18.0
1	C=0.1_gamma=0.01	79.04	0.444	0.444	7.0	10.0	17.0
4	C=1_gamma=0.001	78.69	0.419	0.419	7.0	10.0	17.0
8	C=10_gamma=0.001	78.89	0.431	0.431	7.0	10.0	17.0
0	C=0.1_gamma=0.001	76.06	1.000	1.000	1.0	1.0	2.0

Figure 8: Task 3: Model Selection scores for SVM with 5-folds for Adult dataset

	Hyperparameters	Accuracy	Equality difference	Absolute difference	Accuracy score	Fairness score	Total_score
13	C=10_gamma=0.01	89.85	0.515	0.515	10.0	10.0	20.0
9	C=1_gamma=0.01	89.89	0.601	0.601	10.0	9.0	19.0
12	C=10_gamma=0.001	89.72	0.636	0.636	10.0	8.0	18.0
8	C=1_gamma=0.001	88.95	0.730	0.730	8.0	6.0	14.0
4	C=0.1_gamma=0.001	88.71	0.778	0.778	7.0	5.0	12.0
5	C=0.1_gamma=0.01	88.71	0.772	0.772	7.0	5.0	12.0
14	C=10_gamma=0.1	87.76	0.641	0.641	4.0	8.0	12.0
10	C=1_gamma=0.1	88.15	0.801	0.801	5.0	5.0	10.0
0	C=0.01_gamma=0.001	87.36	1.000	1.000	3.0	1.0	4.0
1	C=0.01_gamma=0.01	87.36	1.000	1.000	3.0	1.0	4.0
2	C=0.01_gamma=0.1	87.36	1.000	1.000	3.0	1.0	4.0
3	C=0.01_gamma=1	87.36	1.000	1.000	3.0	1.0	4.0
6	C=0.1_gamma=0.1	87.36	1.000	1.000	3.0	1.0	4.0
7	C=0.1_gamma=1	87.36	1.000	1.000	3.0	1.0	4.0
11	C=1_gamma=1	87.26	0.952	0.952	3.0	1.0	4.0
15	C=10_gamma=1	86.60	0.888	0.888	1.0	3.0	4.0

Figure 9: Task 3 Model selection scores for reweighted SVM model for Adult dataset