UNIVERSITAT POLITÈCNICA DE CATALUNYA

FACULTAT D'INFORMÀTICA DE BARCELONA

Understanding salary using Machine Learning

Machine Learning

Spring 2022

Authors:
Lorencio Abril, Jose Antonio, *email*: jose.antonio.lorencio@estudiantat.upc.edu
Mayorga Llano, Mariana, *email*: mariana.mayorga@estudiantat.upc.edu

Professor: **Coma Puig, Bernat**

Contents

List of Figures

1 Data Exploration and Preprocessing

The initial step in our exploratory data analysis was to 1) visualize the overall distribution of values per attribute by itself and against the class, 2) look for redundant attributes and 3) assess the registers with null variables.

1.1 Data distribution of variables

Figure 1 shows the class data distribution is about 25/75, indicating that the dataset is not entirely equal but can still be considered balanced. Furthermore, this specific distribution of the class can be a representative sample of the real population considering that statistically, the number of people with an income above $50,000 USD is a minority.

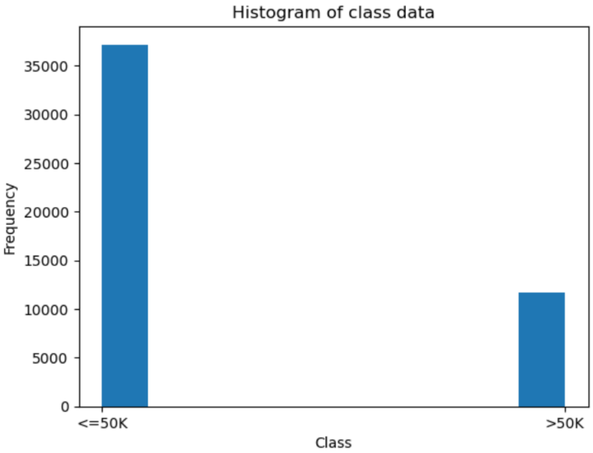


Figure 1: Class distribution

Furthermore, we explored the distribution of different variables. From the categorical attributes such as sex (Figure 6), age (Figure 4) and race (Figure 5), we found that the dataset is mainly composed of white males between 20 and 40 years old. This information is vital as it suggests that, even if the class distribution by itself can match with the real population overall, the dataset is not fairly representative of people of color, females, nor the eldest, which is a point to consider when training our models.

1.2 Data distribution of variables against income

To assess the dataset's bias towards different groups, we plotted the income distribution for these variables (Figure 10 for age, Figure 12 for education level, Figure 13 for marital status, Figure 15 for occupation, Figure 14 for gender and Figure 11 for race). The results showed that there are significant differences in the distribution of the class attribute in each group. As an example, Figure 14 illustrates that the percentage of men with salaries over 50K was about 30%, while it was only 10% for women. This case is particularly interesting since registers from women by themselves are a minority as shown in Figure 6, and considering that from that minority of registers, 90% of them belong to the same class of people with incomes under 50K, it reinforces that the dataset may be biased towards certain groups and the accuracy of predictions may differ from one demographic group to another, which reinforces the need to consider a measure of fairness in our model.

Regarding numerical variables, the distribution of the capital loss (Figure 16), capital gain (Figure 17) and working hours (Figure 18) were also analyzed. From this we discovered that capital-gain and capital-loss are highly skewed, therefore, we deepened into these two variables to understand if there was any correlation between each other or with other variables.

By analyzing the statistics of both variables presented in Table 1 and visually inspecting the scatter plot of the variables shown in Figure 7, we identified a correlation between the changes observed in the data. Specifically, we observed that whenever the capital-gain value is greater than 0, the capital-loss value is always 0, and vice versa. To simplify the analysis process and better understand the relationship between these two variables, we created a new variable called ”capital-diff” that represents the difference between capital-gain and capital-loss. This resulted in a single variable that represents the difference between the original 2 variables.

	Capital-Gain	**Capital-Loss**
count	30162.000000	30162.000000
mean	1092.007858	88.372489
std	7406.346497	404.298370
min	0.000000	0.000000
25%	0.000000	0.000000
50%	0.000000	0.000000
75%	0.000000	0.000000
max	99999.000000	4356.000000

Table 1: Statistical comparison between Capital Gain and Capital Loss

Although creating a new variable has potential risks, such as losing information about the individual variables’ magnitude and direction and the potential for skewed data if the original variables have different ranges or distributions, we believe that the benefits outweigh the risks. By reducing the number of variables, this approach simplifies the data analysis process, which can be especially helpful when working with large datasets or trying to avoid overfitting the model. Additionally, given the observed correlation between the two variables, we think that this approach is worth exploring further. The statistical information from this new variable can be appreciated in Table 2 and confirms the high correlation that was originally assumed and it can be appreciated how this variable is actually present in very few instances. Therefore, we categorized it into three clear segments: When the value is positive, when it is negative and when it is zero (Figure 8).

Statistic	Capital-Dif
Count	30162.000000
Mean	1003.635369
Standard Deviation	7430.372730
Minimum	-4356.000000
1%	-1980.000000
5%	0.000000
10%	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
90%	0.000000
95%	5013.000000
99%	15024.000000
Maximum	99999.000000

Table 2: Statistics for Capital-Dif

To investigate the potential correlations among the numerical variables, a comprehensive analysis was performed using a correlation matrix (Figure 9). The findings indicated a maximum correlation coefficient of 0.08, suggesting a weak correlation between the variables. Based on this observation, it was determined that each numerical variable contributes unique and independent information to the analysis. Therefore, all the numerical variables were retained for further examination and interpretation.

Furthermore, we discovered that the variable ”fnlwgt” does not exhibit any significant correlation with other variables. Further investigation revealed that ”fnlwgt” represents a weight assigned for sampling purposes and is unrelated to income. Consequently, we deemed it appropriate to exclude this column during the data cleaning process to ensure the accuracy and relevance of our analysis.

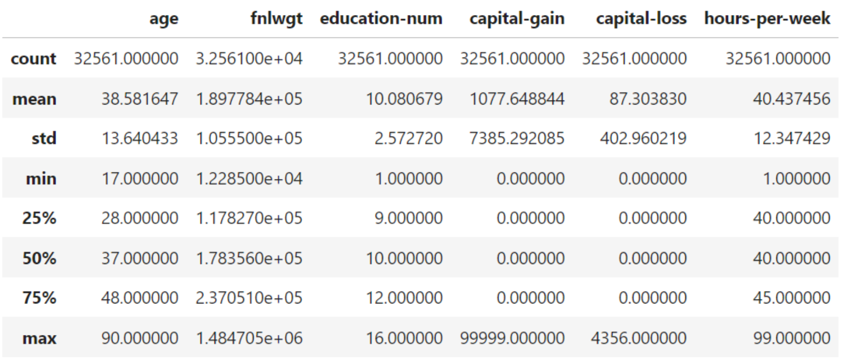
An analysis was conducted on the attributes ’education-num’ and ’education’ to explore any potential correlation (Table 3). The investigation revealed that both variables convey identical information. However, in order to maintain flexibility in modeling and analysis, as well as to preserve conceptual visualization and data validation, it was determined to retain both variables. This approach allows for versatile utilization of the variables in various analytical techniques and ensures comprehensive representation of the underlying information.

education	education-num	count
Preschool	1	45
1st-4th	2	151
5th-6th	3	288
7th-8th	4	557
9th	5	455
10th	6	820
11th	7	1048
12th	8	377
HS-grad	9	9840
Some-college	10	6678
Assoc-voc	11	1307
Assoc-acdm	12	1008
Bachelors	13	5044
Masters	14	1627
Prof-school	15	542
Doctorate	16	375

Table 3: Count of people by education level

Upon examining the comprehensive statistics of our variables (Figure 2), several unusual records became apparent, such as instances where individuals reported working 99 hours per week or having a capital gain of 99999. Furthermore, we identified missing values in three attributes. Consequently, was required an assessment regarding how relevant where these elements and how would we treat them to avoid noise and outliers that could affect our models learning.

A significant number of missing values were identified in the workclass (1836), occupation (1843), and native country (583) attributes. Notably, workclass and occupation exhibited a similar number of missing values. To further investigate, we examined the instances where both variables were absent, revealing that all cases lacking workclass also had missing occupation values, totaling 1836 instances. While considering the possibility of imputing these missing values, we



	age	fnlwgt	education-num	capital-gain	capital-loss	hours-per-week
count	32561.000000	3.256100e+04	32561.000000	32561.000000	32561.000000	32561.000000
mean	38.581647	1.897784e+05	10.080679	1077.648844	87.303830	40.437456
std	13.640433	1.055500e+05	2.572720	7385.292085	402.960219	12.347429
min	17.000000	1.228500e+04	1.000000	0.000000	0.000000	1.000000
25%	28.000000	1.178270e+05	9.000000	0.000000	0.000000	40.000000
50%	37.000000	1.783560e+05	10.000000	0.000000	0.000000	40.000000
75%	48.000000	2.370510e+05	12.000000	0.000000	0.000000	45.000000
max	90.000000	1.484705e+06	16.000000	99999.000000	4356.000000	99.000000

Figure 2: Attributes' Stadistics

recognized the absence of any other variables that could provide a reliable correlation. Consequently, imputing the missing data may introduce bias into the model. Given that these records account for only 5% of the dataset, we determined that removing them would be a more suitable course of action.

This discovery prompted us to explore whether the attributes of workclass and occupation provide redundant information. However, upon analyzing Table 5 and Table 4, we observed that the mapping between occupation and workclass is not unique, indicating that both attributes offer distinct and valuable information. As a result, we have decided to retain them as separate variables in our analysis.

Occupation	Workclass	Count
Adm-clerical	Federal-gov	317
Adm-clerical	Local-gov	283
Adm-clerical	Private	2833
Adm-clerical	Self-emp-inc	31
Adm-clerical	Self-emp-not-inc	50
...	...	...
Transport-moving	Private	1266
Transport-moving	Self-emp-inc	27
Transport-moving	Self-emp-not-inc	122
Transport-moving	State-gov	41
Transport-moving	Without-pay	1

Table 4: Count of people grouped by occupation and workclass

Regarding the outliers and potential noise, we focused on capital-diff and hours-per-week, since these are the only relevant numeric features we had at this point. For capital-diff, removing the maximum anomalous value of 99999 was enough to end up with a variable without obvious outliers. As for the working hours, we additionally set a thresh-hold of 70 hours per week and removed values that surpassed this limit, giving as a result the statistics shown in Figure 3. This step allowed us to mitigate extreme values and ensure the reliability of our analysis.

Additionally, by considering both entropy and mutual information (MI) as shown in Table 6, we gain insights into the data content and predictive strength of each feature. Features with low entropy and high MI (marital-status, relationship, education-num, capital-gain,capital-diff) are particularly valuable, as they exhibit a strong association with the target variable (high

workclass	occupation	count
Federal-gov	Adm-clerical	317
Federal-gov	Armed-Forces	9
Federal-gov	Craft-repair	64
Federal-gov	Exec-managerial	180
Federal-gov	Farming-fishing	8
...	...	...
Without-pay	Farming-fishing	6
Without-pay	Handlers-cleaners	1
Without-pay	Machine-op-inspct	1
Without-pay	Other-service	1
Without-pay	Transport-moving	1

Table 5: Count of people grouped by workclass and occupation

MI) and knowing their value gives a lot of information about the target variable (low entropy). Conversely, features with high entropy and low MI (education, capital-loss, race, workclass) may provide less predictive power, while features with high entropy and high MI (hours-per-week, occupation, age) are highly influential but not that informative by themselves. Lastly, features with high entropy and low MI (native-country) may have limited utility in our analysis. These hypotheses will be further discussed in comparison to the results of our learning models.

Feature	**Entropy**	**MI**
marital-status	3.972	0.113
relationship	3.167	0.069
education-num	8.850	0.059
capital-gain	1.736	0.058
age	48.898	0.056
capital-diff	6.930	0.052
capital-diff_categorical	2.684	0.039
hours-per-week	54.776	0.035
occupation	8.876	0.033
sex	1.398	0.030
education	8.850	0.017
capital-loss	6.007	0.013
race	3.161	0.012
workclass	5.273	0.011
native-country	26.487	0.000

Table 6: Entropy and Mutual Information

2 Models

After conducting preprocessing steps, we have prepared two final datasets for model training (DF1 and DF2). Both of them include the numerical variables (excluding fnlwgt, capital-gain, and capital-loss) and categorical variables (including capital-diff). These datasets do not contain the outliers nor missing values mentioned in the previous section.

The only difference between these datasets is the representation of the education variable. In

DF1, the numerical variable of education is used, while in DF2 is used the categorical version.

For evaluating the models, we split the dataset into training and testing sets with an 80/20 ratio. The training set will be used for training using cross-validation to identify the ideal hyperparameters for each of the models. As cross-validation metric, we decided to use the F1-Score, since it leverages the recall and precision metrics, which are important in our case, as the '<=50' class is dominant, especially over some subsets of the data, such as women.

The test set will not be used in the training of the models and it will be reserved for the last section where we will compare the performance of all models with the best hyperparameters, to be able to test the generalization capabilities of the models and conclude with the final chosen model.

2.1 Logistic regression

The datasets DF1 and DF2 were one-hot encoded for this implementation of logistic regression considering that the algorithm does not work well on categorical variables. We decided to use this model because it is good for binary classification tasks, making it suitable for our current objective of predicting whether an individual's income exceeds $50,000 per year.

Both datasets had very similar results. When the model predicts an individual's income to be below or equal to $50,000, it is correct approximately 88% of the time in DF1 and 87% of the time in DF2. Similarly, the recall values indicate that the model successfully captures a high proportion of individuals whose income falls below or equal to $50,000 in both DF1 and DF2.

However, these numbers decrease for the '>50K' class, were when the model predicts an individual's income to be above $50,000, it is accurate approximately 71% of the time in both datasets. The recall value for the >50K class shows a slight decline from 0.59 in DF1 to 0.59 in DF2. This suggests that the model captures a relatively smaller proportion of individuals with incomes above $50,000 in both datasets.

For the <=50K class, both DF1 and DF2 exhibit F1-scores of 0.90, indicating a balanced combination of precision and recall. In the case of the >50K class, the F1-scores range from 0.64 to 0.65 in both datasets, so the overall differences between the datasets are considered to be minimal.

Dataset	**Class**	**Precision**	**Recall**	**F1-Score**	**Support**	**Accuracy**
DF1	<=50K	0.88	0.92	0.90	17,709	0.84
	>50K	0.71	0.59	0.65	5,726	
DF2	<=50K	0.87	0.92	0.90	17,709	0.84
	>50K	0.71	0.59	0.64	5,726	

Table 7: Performance of the Logistic Regression Model

To improve the model, L2 regularization with different values of the hyperparameter C (inverse of the regularization strength) was applied using cross-validation. The mean cross-validation accuracy scores for each value of C are shown in Table 8, which indicates that based on the cross-validation results, the best value of C is 1.0 with the highest accuracy value of 0.838. This implies that the first model already incorporates the best hyperparameter setting and there is no need for additional training steps.

To assess the impact of preprocessing, the model was also trained on the raw dataset without

C	Weighted F1-Score
10.0	0.837
1.0	0.838
0.1	0.837
0.01	0.829
0.001	0.771

Table 8: Mean Cross-Validation Weighted F1-Scores for Different Values of C

any preprocessing. The model's performance is summarized in Table 9. This illustrates how the preprocessing step had a significant positive impact on the performance of the logistic regression model. The model trained on the preprocessed dataset achieved an accuracy of 0.84, surpassing the accuracy of 0.80 obtained from the model trained on the raw dataset. The precision and recall for the <=50K income category notably improved to 0.88 and 0.92, respectively, indicating better classification of instances within this category. Although the performance on the >50K class remained moderate, the results demonstrated the effectiveness of preprocessing in enhancing the model's predictive capabilities and accurate classification of individuals into income categories.

Class	Precision	Recall	F1-Score	Support	Accuracy
<=50K	0.81	0.97	0.88	4942	0.80
>50K	0.73	0.27	0.39	1571	

Table 9: Performance of Logistic Regression Model on Raw Dataset

By examining the coefficients of the logistic regression model we can identify which variables are considered more important by the model. The coefficient values reflect the extent to which each variable influences the prediction of ">50K" income. Comparing these results with the previous analysis of MI and entropy, we can confirm that some values related to the variables 'marital-status', 'relationship', 'education' are the most deterministic for this model as shown in Table 10. Conversely, variables with smaller or close to zero coefficients, like 'native-country' have minimal impact on predicting higher incomes, which also matches with the analysis of MI and entropy.

Variable	Coefficient
relationship_Wife	1.343038
education_Prof-school	1.289627
education_Doctorate	1.234793
education_Masters	1.054595
marital-status_Married-civ-spouse	0.903928
...	...
education_11th	-1.190532
education_7th-8th	-1.250870
relationship_Own-child	-1.409934
sex_Female	-1.412797
capital-diff_categorical_zero	-1.795055

Table 10: Most relevant Variable Coefficients

Overall, the logistic regression model demonstrates good performance in predicting income levels. Although the model achieves a relatively high accuracy of 84%, further analysis is required to improve its performance, especially in correctly identifying individuals with ">50K" incomes.

2.2 Decision tree

The decision tree model offers advantages over logistic regression as it can handle numeric encoding of education and is less prone to overfitting due to the reduced number of variables.

First, a comparison of cross-validated accuracy for different maximum depths of the decision tree models is provided for depths of 1, 2, 3, 4, 5, 10, 20, 50, 100, and 1000. Table 11 presents the cross-validated F1-score for different maximum depths of the decision tree models on both DF1 and DF2. As the depth increases, the accuracy initially improves, reaching its peak at depth 10 for both datasets with F1-scores of 0.8239 for DF1 and 0.8255 for DF2. Beyond depth 10, the accuracy starts decreasing, indicating a diminishing return on complexity.

Max. Depth	**CV F1-Score for DF1**	**CV F1-Score for DF2**
1	0.6505	0.6505
2	0.7988	0.7524
3	0.8221	0.7943
4	0.8237	0.8103
5	0.8239	0.8255
10	0.8247	0.8273
20	0.8049	0.8012
50	0.7910	0.7900
100	0.7912	0.7901
1000	0.7907	0.7895

Table 11: Comparison of Decision Tree Models

After defining 10 as the proper depth for the model, decision tree analysis was performed on both datasets (Table **??** and Table 12, respectively) in order to be evaluated in terms of performance. DF1 achieved an overall accuracy of 86%, with a precision of 0.88 and recall of 0.95 for the "<=50K" class. However, it showed lower performance for the ">50K" class, with precision, recall, and F1-score values of 0.78, 0.60, and 0.68, respectively. In contrast, DF2 achieved an accuracy of 86% with a precision of 0.90 and recall of 0.92 for the "<=50K" class, resulting in an F1-score of 0.91. However, the ">50K" class in DF2 exhibited lower precision, recall, and F1-score values of 0.74, 0.67, and 0.70, respectively. Although the differences are minimum, we decided to use the model trained with DF2 for future evaluations against the other models. After this assessment, we decided to utilize only the DF2 in the following evaluations, since at this point the observed differences were minor, and slightly better for DF2.

Dataset	**Class**	**Precision**	**Recall**	**F1-score**	**Support**	**Accuracy**
DF1	<=50K	0.88	0.95	0.91	17709	0.86
	>50K	0.78	0.60	0.68	5726	
DF2	<=50K	0.90	0.92	0.91	17709	0.86
	>50K	0.74	0.67	0.70	5726	

Table 12: Performance of Decision Tree

2.3 Naive Bayes

We applied the Naive Bayes algorithm to the dataset DF2 using the GaussianNB implementation from the scikit-learn library. Our first approach with this model exhibited imbalanced prediction behavior, favoring the ">50K" class. Despite achieving high precision for the "<=50K" class, the model struggles with recall for this class. The low overall accuracy suggests room for improvement, especially in handling the imbalanced nature of the dataset.

The results show that the model performs well in terms of precision for the "<=50K" class, achieving a high value of 0.96. However, the recall for this class is relatively low, indicating that the model identifies only 47% of the instances correctly. On the other hand, the model exhibits higher recall for the ">50K" class, suggesting its ability to correctly identify 93% of the instances belonging to this class. The F1-score for both classes was low, having values of 0.63 for "<=50K" class and 0.52 for the ">50K" class.

The overall accuracy of the model is 0.59, which is relatively low considering the imbalanced nature of the dataset. The low accuracy can be attributed to the model's tendency to predict the ">50K" class more frequently, as mentioned in the analysis. This behavior is counterintuitive, as the dataset is skewed towards the "<=50K" class, and we would expect the model to predict this class more often. Therefore we proceed to apply Laplace smoothing.

Dataset	Precision	Recall	F1-score	Support	Accuracy
DF2 (<=50K)	0.96	0.47	0.63	17709	0.59
DF2 (>50K)	0.36	0.93	0.52	5726	
Laplace (<=50K)	0.91	0.83	0.87	17709	0.81
Laplace (>50K)	0.58	0.75	0.66	5726	

Table 13: Naive Bayes Results

To improve the results, Laplace smoothing was applied to the Naive Bayes classifier. The hyperparameter alpha was tuned to determine the optimal level of smoothing. Several values of alpha were tested, and their corresponding F1-scores were evaluated. Based on these results shown in Table 14, the alpha value that maximizes the F1-score is 1.

Alpha	Test F1-score
0.001	0.814552719236507
0.01	0.8145970555596144
0.1	0.8146039669460009
1	0.814662271687012
10	0.8128405865467503
100	0.8140011875359647
1000	0.6504838416808123

Table 14: Cross-Validation F1-scores for different alpha values

The analysis reveals that the Multinomial Naive Bayes classifier with Laplace smoothing performs better than the initial model without smoothing (Table 13). The model shows improved balance in predicting both classes, with a higher emphasis on predicting the "<=50K" class, which is expected considering the dataset's class imbalance. However, it is worth noting that the model still tends to predict the ">50K" class more frequently compared to the logistic regression model.

2.4 Random forest

Random Forest was performed due to its ability to handle high-dimensional datasets, capture complex relationships between variables, and provide robust predictions. Its ensemble-based approach, combining multiple decision trees, allows for improved accuracy and reduced overfitting compared to individual decision trees. Additionally, Random Forest is known for its versatility in handling both categorical and numerical features, making it suitable for our dataset, which includes a mix of both types of variables.

We experimented with different values for the number of estimators (10, 50 and 100), the maximum depth of each tree (5, 10, 15, 20, 25 and 30) and the maximum number of features considered for splitting (0.1, 0.25, 0.5, 0.75 and 1.0) to identify the best combination of hyperparameters that yielded the highest accuracy. We trained multiple Random Forest models using different combinations of these values and evaluated their performance with cross-validation.

After analyzing the F1-scores obtained for each model, we identified the best-performing model. The model with 50 estimators, 0.75 features and a max depth of 10 achieved the highest F1-score of 0.837. Upon further examination of the results, we noticed a trend where increasing the number of estimators and the maximum depth generally improved the F1-score of the models. However, we also observed that there was a point of diminishing returns.

Class	**Precision**	**Recall**	**F1-Score**	**Support**	**Accuracy**
<=50K	0.89	0.94	0.92	17709	0.87
>50K	0.78	0.64	0.70	5726	

Table 15: Naive Bayes Performance

2.5 Gradient boosting

In this section, we applied the Gradient Boosting algorithm using the GradientBoostingClassifier from the sklearn library. We varied the values for the number of estimators (10, 50 and 100), the maximum depth of each tree (5, 10, 15, 20, 25 and 30) and the maximum number of features considered for splitting (0.1, 0.25, 0.5, 0.75 and 1.0). For each combination of hyperparameters, we trained the model and evaluated the performance. The model configuration with 100 estimators, a maximum depth of 5, and 0.25 features achieved the highest accuracy of 84.4% giving as a result the performance shown in Table 16.

	Precision	**Recall**	**F1-score**	**Support**	**Accuracy**
<=50K	0.89	0.93	0.91	17709	0.86
>50K	0.76	0.64	0.69	5726	

Table 16: Gradient Boosting Results

Overall, increasing the number of estimators and maximum depth tends to improve the cross-validated F1-scores. This suggests that a higher number of trees and deeper trees can capture more complex patterns in the data, leading to better performance. However, a tree too deep can risk overfitting. On the other hand, the effect of maximum features on the cross-validated F1-scores is not as straightforward. The cross-validated F1-scores tend to increase as maximum features increase from 0.1 to 0.5 and then decrease slightly as maximum features reach 1.0. This suggests that a moderate number of features (around 0.25-0.5) leads to better performance, while considering all features (1.0) might make our models too similar, leading to a smaller degree of independency, affecting the ensemble method negatively.

2.6 Neural network

This section presents the results of a neural network model using the MLPClassifier from the scikit-learn library. Neural networks are widely used for various classification tasks due to their ability to capture complex relationships in the data. However, their performance heavily depends on the choice of hyperparameters. For this reason, we employed a grid search approach to tune the regularization parameter alpha (0.01, 0.1 and 1.0), learning rate (0.001, 0.01 and 0.1), hidden layer sizes [(10,), (50,), (10, 10), (50, 50)], and activation function ('logistic' and 'relu'). To ensure that data's magnitudes does not affect the neural network training negatively, a standard scaling technique was applied using the StandardScaler.

The results indicate that different hyperparameter configurations lead to varying F1 scores. Among the tested hyperparameters, the best-performing combination was observed with an alpha value of 0.01, learning rate of 0.001, hidden layer sizes of (10, 10), and activation function 'relu', achieving an F1 score of 0.842381711. The performance of these model with the mentioned hyperparameters can be shown in Table 17.

	Precision	**Recall**	**F1-score**	**Support**	**Accuracy**
<=50K	0.89	0.93	0.91	17709	0.86
>50K	0.75	0.65	0.69	5726	

Table 17: Neural Networks Results

We could particularly notice that among the tested alpha values, 0.01 generally yielded the highest test f1 scores across different configurations, suggesting that a moderate amount of regularization, as represented by alpha = 0.01, is beneficial for achieving better generalization. Additionally, regarding the learning rate, we observed that a learning rate of 0.001 generally led to higher test f1 scores compared to a learning rate of 0.01. This suggests that a smaller learning rate allowed the model to converge more accurately and achieve better performance. This is expected, since we go slower in the direction of the gradient. But we need to be careful, because it could also happen that we don't reach the minimum if the steps are 'too short'.

3 Results comparison over Test Data

To compare the performance of the trained models and determine the best one, we conducted an evaluation using the test data. The accuracy and F1-score metrics were employed as the primary measures of overall model performance. Additionally, we examined precision and recall, to gain insights into the models' performance on individual classes.

Model	Accuracy	Precision	Recall	F1
Logistic Regression	0.843147	0.703316	0.581949	0.636902
Naive Bayes	0.805257	0.566304	0.752347	0.646202
Decision Tree	0.829322	0.645944	0.615162	0.630178
Random Forest	0.835125	0.679213	0.573285	0.621770
XGBoost	0.847244	0.706229	0.605776	0.652157
MLP	0.842806	0.691736	0.604332	0.645087

Table 18: Model Performance Metrics

Upon analyzing the performance metrics (Table 18), we observed that XGBoost achieved the highest accuracy of 0.847244, as well as the highest F1-score of 0.652157, among all models, along with the highest precision with 0.706229, demonstrating a relatively strong ability to correctly classify positive instances. In comparison, other models such as Logistic Regression, Decision Tree, and Random Forest showed lower precision values. This discrepancy indicates that these models were more prone to predicting instances as '<=50K', potentially resulting in a higher number of false positives.

While XGBoost exhibits lower recall (0.605776), which represents the proportion of correctly predicted positive instances out of the actual positive instances, the emphasis on precision is warranted due to the dataset's class imbalance. Prioritizing precision helps mitigate the issue of false positives, which is particularly important when making predictions related to the '>50K' class.

Considering the combination of high accuracy and strong precision, we select the XGBoost model with 100 estimators, a maximum depth of 5, and 0.25 features, as the best-performing model for this task. The XGBoost algorithm demonstrates favorable overall predictive capabilities, with an acceptable trade-off between accuracy and precision. Furthermore, XGBoost's complex ensemble approach allows it to capture non-linear relationships and interactions in the data effectively. By selecting XGBoost as the best model, we prioritize the need for accurate predictions while managing the risk of false positives.

4 Introducing Fairness

Within the context of our project, we have acknowledged the potential impact of demographic attributes and class imbalance on biased predictions. To address this concern, we decided to investigate the influence of the gender attribute on model performance.

Inspired by the research presented in a Google paper titled "Equality of Opportunity in Supervised Learning" [HPS16] and showcased in an online demo called "Attacking discrimination with smarter machine learning" [HWV], we sought to implement a threshold bias awareness/fairness approach specifically tailored for logistic regression.

The underlying concept revolves around the notion that a model can exhibit bias towards a particular class, which becomes problematic when the model is employed to make decisions affecting individuals. For instance, if a model is utilized to determine loan eligibility and exhibits bias towards a specific class, it results in discriminatory treatment against that particular class.

To mitigate this issue, a possible solution involves adopting different thresholds for each class, aiming to equalize a selected metric between the two classes. We followed the "Equal opportunity" approach outlined in the paper, which seeks to equalize the true positive rate between the classes. Our initial implementation involved a naive search for a pair of thresholds within a specified range that would bring the true positive rates of both classes closer together.

Firstly, we assessed whether the model exhibited differential behavior for men and women. Our observations indicated that the model displayed higher accuracy for women. However, this disparity could be attributed to the smaller percentage of women earning more than 50K and the model's general inclination to predict incomes as '<=50K'. As can be seen in the confusion matrices shown in Figure 19, the females higher accuracy is obtained due to the fact that most of them earn '<=50K', and very few are predicted to earn '>50K'.

Next, we examined the true positive rates for both genders. The sensitivities were found to be 0.61 for males and 0.40 for females, indicating room for improvement. Subsequently, we sought to enhance the model's performance by utilizing different thresholds for each class. This exploration was conducted using the training data, with the selected thresholds then applied to the test data to evaluate whether the model achieved improved balance.

Through iterative analysis, we identified the optimal thresholds that resulted in the best balance between the classes. The selected thresholds were found to be 0.53 for males and 0.37 for females, yielding a minute difference of 0.00007667. Upon assessing the true positive rates in the test data, we observed significant improvement, with values of 0.57 for males and 0.49 for females. The new resulting confusion matrices are shown in Figure 20, where we can see how now the model is able to identify much more of the women that earn more than '>50K', while the amount of false positive increase.

The model's accuracy was measured to be 0.80, while the F1-score reached 0.64. It is evident that the model achieved greater balance, with the true positive rate becoming more similar for both classes. However, it is worth noting that this adjustment resulted in a decreased true positive rate for the male class. Hence, the decision to implement such thresholds should be made considering the specific use case and the significance of the "Equal opportunity" metric in scenarios where biased decisions towards a particular class are deemed unacceptable.

For cases where bias mitigation is not the primary concern, and accuracy or the F1 score have been the primary focus, it is crucial to weigh the trade-off involved. In our analysis, we observed a slight decrease in both accuracy and F1 score. Consequently, striking the appropriate trade-off becomes imperative when considering the model's intended application.

By implementing the threshold bias awareness/fairness approach, we have taken steps to rectify the potential bias introduced by the logistic regression model. The adjusted thresholds have resulted in a more balanced representation of true positive rates across gender classes. This consideration enables decision-makers to align their priorities with the desired fairness objectives, thus ensuring equitable outcomes in their specific context.

5 Conclusion

This study provides valuable insights into the salaries dataset, offering significant findings and implications.

Firstly, we discovered several segments within the dataset, with certain groups being notably underrepresented. This observation emphasizes the importance of considering fairness, particularly when employing the model for decision-making processes that impact individuals. We demonstrated how the model can be tailored to enhance fairness by employing distinct thresholds for different classes, as exemplified in our approach to address the gender attribute.

Secondly, we identified a significant class imbalance within the dataset, as the majority of instances belonged to the '<=50K' class. This imbalance has a notable effect on the models, leading them to predict this class more frequently. Consequently, it is crucial to account for this class imbalance during model training. To evaluate model performance effectively, we adopted the F1-score, which places emphasis on the positive class, aligning with the specific requirements of our use case.

Thirdly, we observed that the models exhibited slightly improved performance when utilizing the DF2 dataset, where the education attribute was encoded as numerical values rather than one-hot encoding. This improvement can be attributed to the reduced number of variables, minimizing the risk of overfitting. Although the gain in performance was marginal, it remains a noteworthy consideration in model optimization.

Additionally, the applied preprocessing techniques proved to be valuable, enhancing the models' performance when operating on preprocessed data. This outcome emphasizes the significance of appropriate data preprocessing in achieving optimal model outcomes.

Furthermore, through feature selection and engineering, we gained an understanding of the varying relevance of different variables in salary prediction. For instance, we eliminated the 'fnlwgt' attribute, which exhibited negligible utility, and introduced the 'capital-diff' attribute, consolidating information from 'capital-gain' and 'capital-loss' attributes. These efforts contributed to refining the models' predictive capabilities.

Finally, our comprehensive evaluation determined that the XGBoost model demonstrated superior performance in terms of accuracy and F1-score. Moreover, we demonstrated the adaptability of logistic regression models, and any model capable of producing probabilities, in promoting fairness by utilizing distinct thresholds for different classes.

In conclusion, this study provides a deep understanding of the salaries dataset, revealing crucial insights and implications. By addressing fairness concerns, refining preprocessing techniques, and employing advanced modeling approaches, we have enhanced the models' accuracy and fairness. The results highlight the effectiveness of the XGBoost model and emphasize the trade-offs and considerations involved in utilizing models for decision-making processes that impact individuals.

6 Future extensions and known limitations

One possible area for improvement is in the realm of feature engineering. By exploring additional feature transformations, we may enhance the models' predictive power. For example, we could consider further subdividing the 'capital-diff' attribute into more granular categories or create new attributes based on the 'occupation' feature, which has shown significant relevance in salary predictions. Additionally, clustering techniques could be employed to group individuals and extract valuable information for constructing new attributes.

To optimize model performance, we can explore alternative hyperparameters or consider different machine learning algorithms. For instance, experimenting with different neural network architectures or exploring boosting algorithms like LightGBM could yield improved results. Evaluating the efficacy of these alternative models and hyperparameters is an essential step in refining the predictive capabilities of the system.

To augment the existing dataset, integrating external data sources might be beneficial. One promising approach is to cross-reference information from the census dataset to gain additional insights into individual characteristics. By incorporating more comprehensive information, particularly for underrepresented classes, the models may capture nuanced patterns and improve their overall performance.

Furthermore, while this project implemented a basic bias awareness/fairness mechanism, there is room for more sophisticated approaches. Exploring advanced techniques, such as the "Equalized Odds" approach, which aims to equalize the false positive rate between the two classes, could address potential biases more effectively. Implementing these advanced fairness strategies would provide a more robust and equitable decision-making framework when utilizing models that impact individuals' lives. However, for the scope of this project, the current naïve approach sufficiently illustrates the inherent trade-offs and considerations involved when employing models for decision-making in sensitive contexts.

Addressing these future extensions and known limitations will allow for the refinement and expansion of the current models, ultimately enhancing their performance and ensuring their fairness and effectiveness in real-world applications.

References

[HPS16] Moritz Hardt, Eric Price, and Nathan Srebro. *Equality of Opportunity in Supervised Learning*. 2016. arXiv: 1610.02413 [cs.LG].

[HWV] Moritz Hardt, Martin Wattenberg, and Fernanda Viégas. *Attacking discrimination with smarter machine learning*. URL: https://research.google.com/bigpicture/attacking-discrimination-in-ml/.

[KB] Ronny Kohavi and Barry Becker. *Adult Data Set*. url: https://archive.ics.uci.edu/ml/datasets/Adult.

Appendix A Appendix

A.1 ORIGINAL PROJECT PROPOSAL

A.1.1 Introduction

The original dataset used for this project is the Adult dataset [KB], which was extracted from the 1994 Census database by Ronny Kohavi and Barry Becker. It contains **14 features** describing demographic information about individuals, as well as a binary class based on income levels, using a threshold of $50,000 per year. The dataset includes **32561 instances**[1], each with 14 features, including information about age, workclass, education level, marital status, occupation, relationship, race, sex, native country, and more.

It is important to mention that the records of this set have already been filtered using the following conditions[2]: ((AAGE>16) and (AGI>100) and (AFNLWGT>1) and (HRSWK>0)). However, further pre-processing and cleaning will be done for this project.

A.1.2 Attributes

From the attributes included in this data-set, 6 of them are represented as numerical attributes and the remaining 8 fields belong to a categorical feature. The class is a binary property describing whether an individual's income level is above or below $50,000 per year. The attributes are shown in Table 19.

Attribute	**Type**	**Attribute**	**Type**
age	continuous	relationship	discrete (6 categories)
workclass	discrete (8 categories)	race	discrete (5 categories)
fnlwgt	continuous	sex	discrete (2 categories)
education	discrete (16 categories)	capital-gain	continuous
education-num	continuous	capital-loss	continuous
marital-status	discrete (7 categories)	hours-per-week	continuous
occupation	discrete (14 categories)	native-Country	discrete (41 categories)

Class	**Type**
Income	discrete (binary)

Table 19: Adult dataset attributes.

A.1.3 Project Objective

1. The first and most important objective of the project is to develop models that, given demographic characteristics, can predict if a subject earns more or less than $50000 per year.

2. Apart from this, given the nature of the dataset (e.g. there is unbalanced in the races representation or double as men as women), we plan to tackle classification fairness and bias, applying different methodologies that exist to deal with this.

   For this, we want to apply a similar approach to that in [HPS16], where the authors propose different methods to reduce the prediction biases of our models. This could be

[1] In the site it says it has more than 40000, but we have checked it and the correct value is 32561.

[2] Basically, they extracted relevant records from the census. For instance, people under 16 are not earning any salary.

useful, for example, if our model was part of a bigger system, in which it is important for us not to be biased against a particular group.

A.2 ADDITIONAL FIGURES

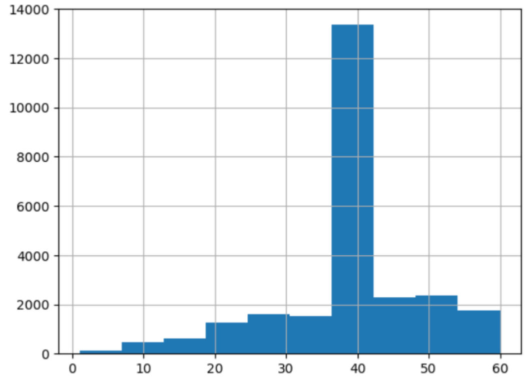


Figure 3: Working hours distribution after preprocessing

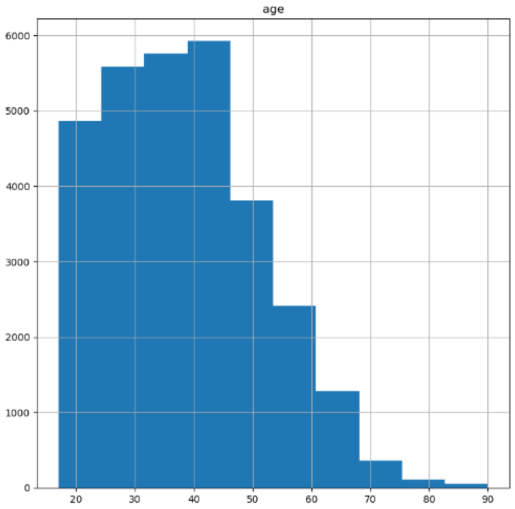


Figure 4: Age distribution

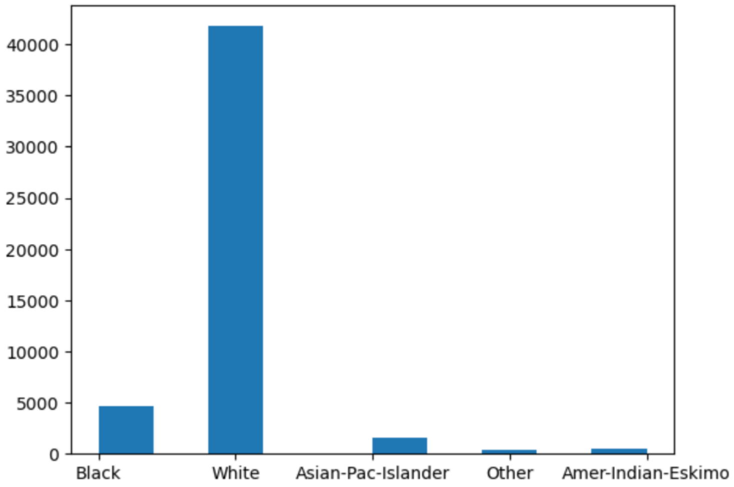


Figure 5: Race distribution

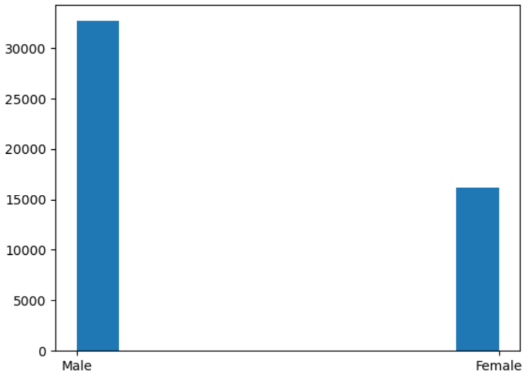


Figure 6: Sex distribution

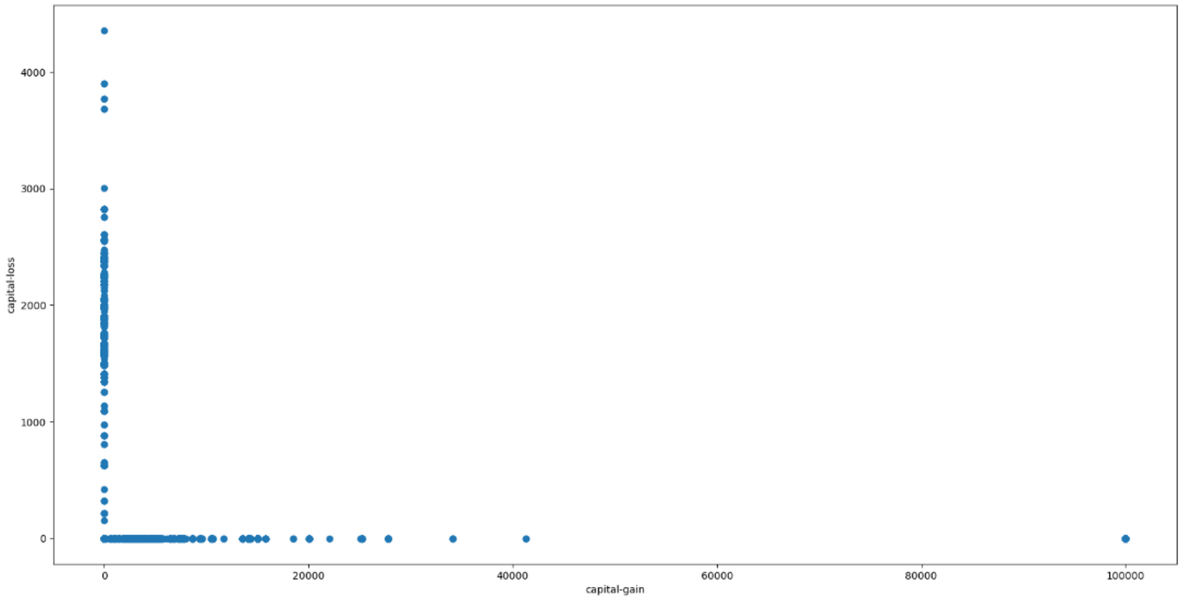


Figure 7: Scatter plot of Capital Gain vs Capital Loss

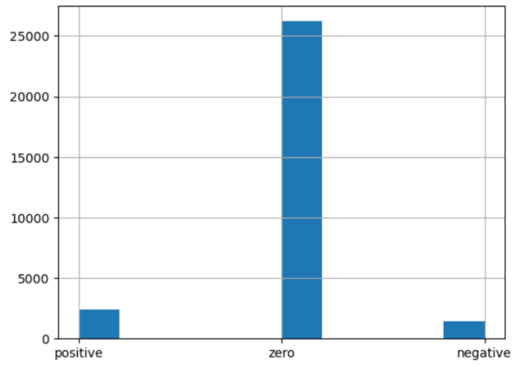


Figure 8: Segmentation of Capital Difference

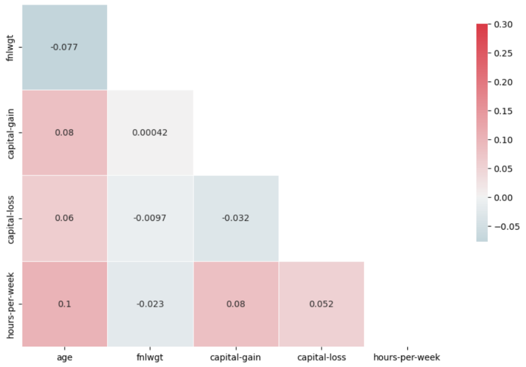


Figure 9: Correlation matrix among numerical variables

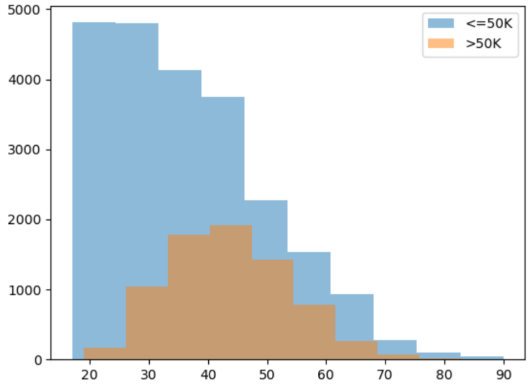


Figure 10: Distribution of salary by age

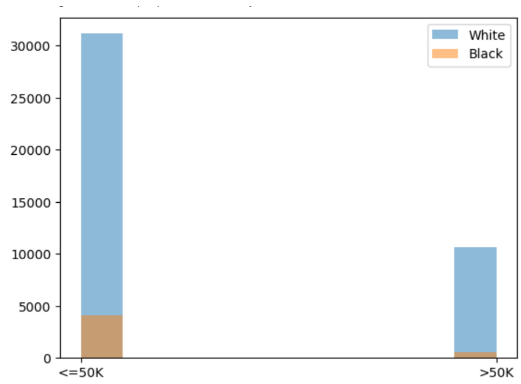


Figure 11: Salary distribution by race

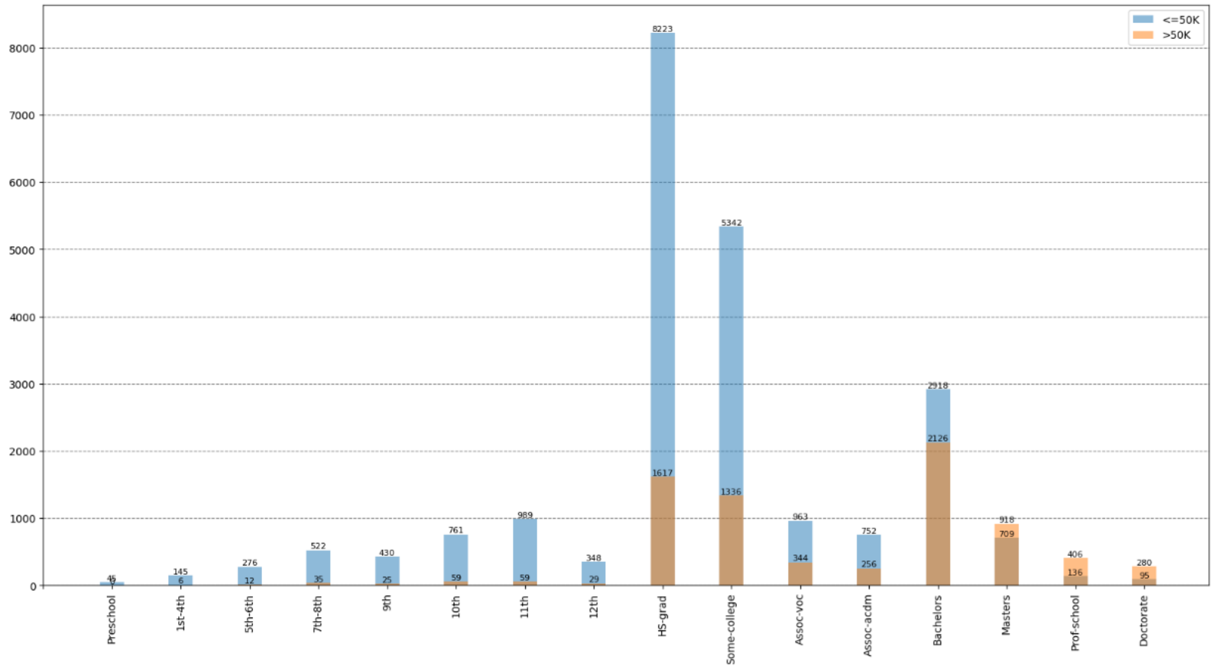


Figure 12: Distribution of salary by education

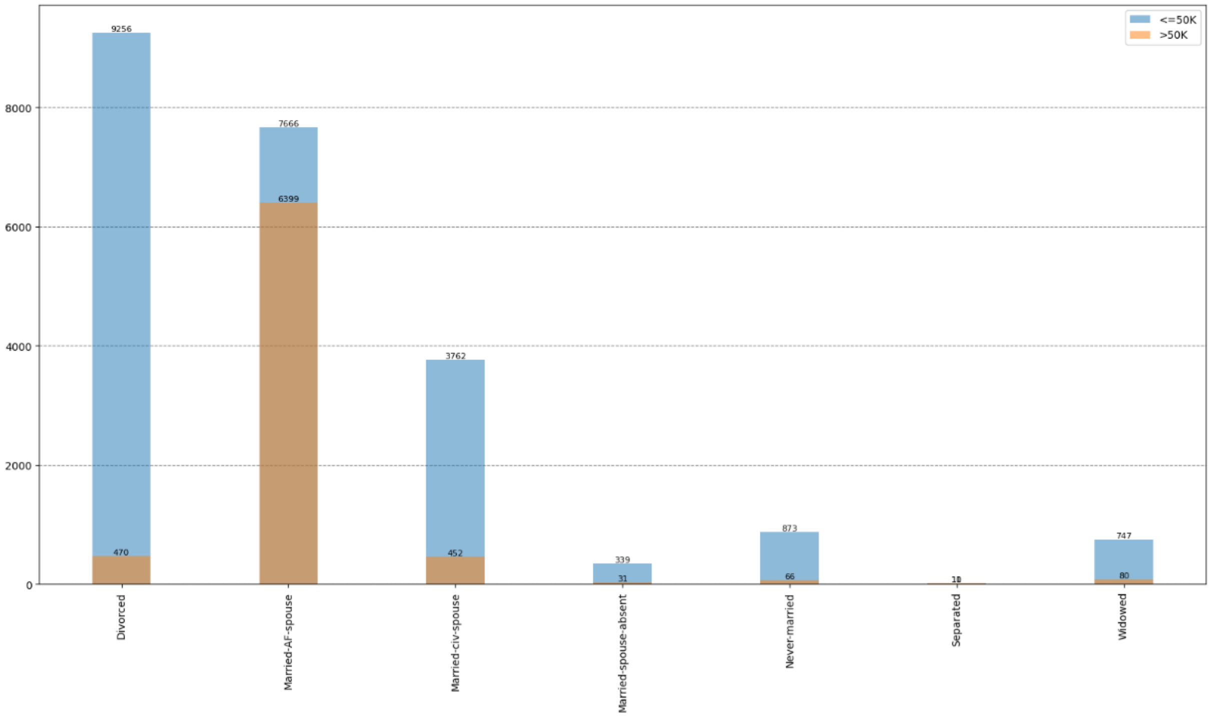


Figure 13: Distribution of salary by marital status

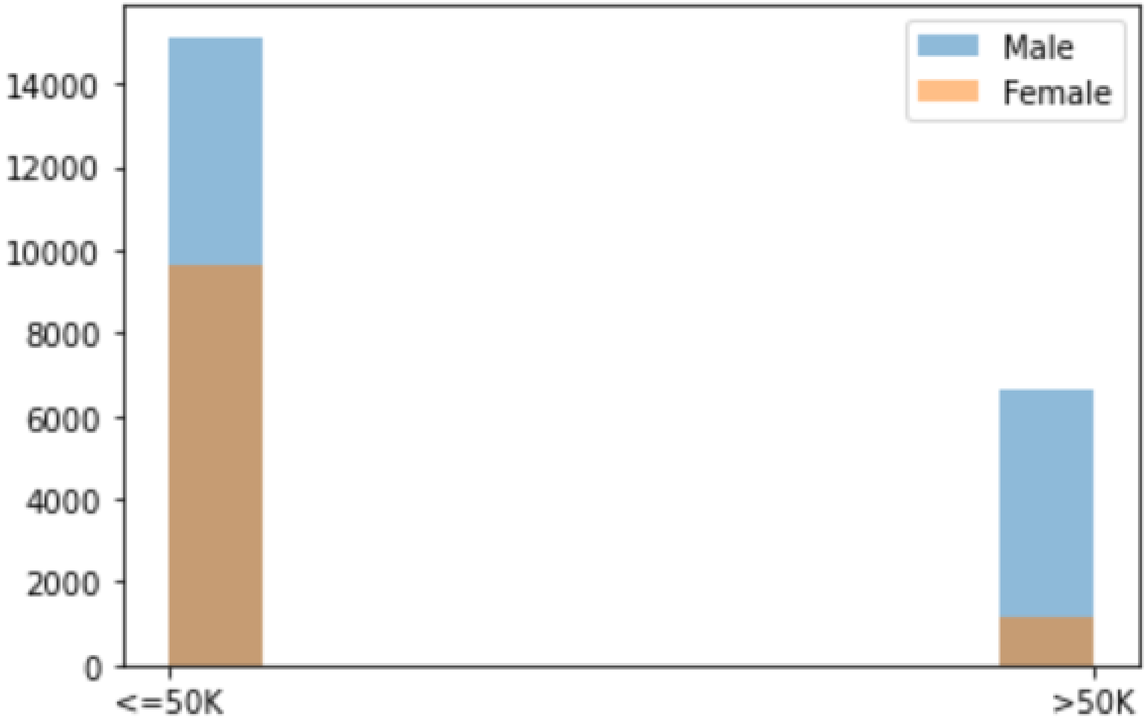


Figure 14: Distribution of salary by gender

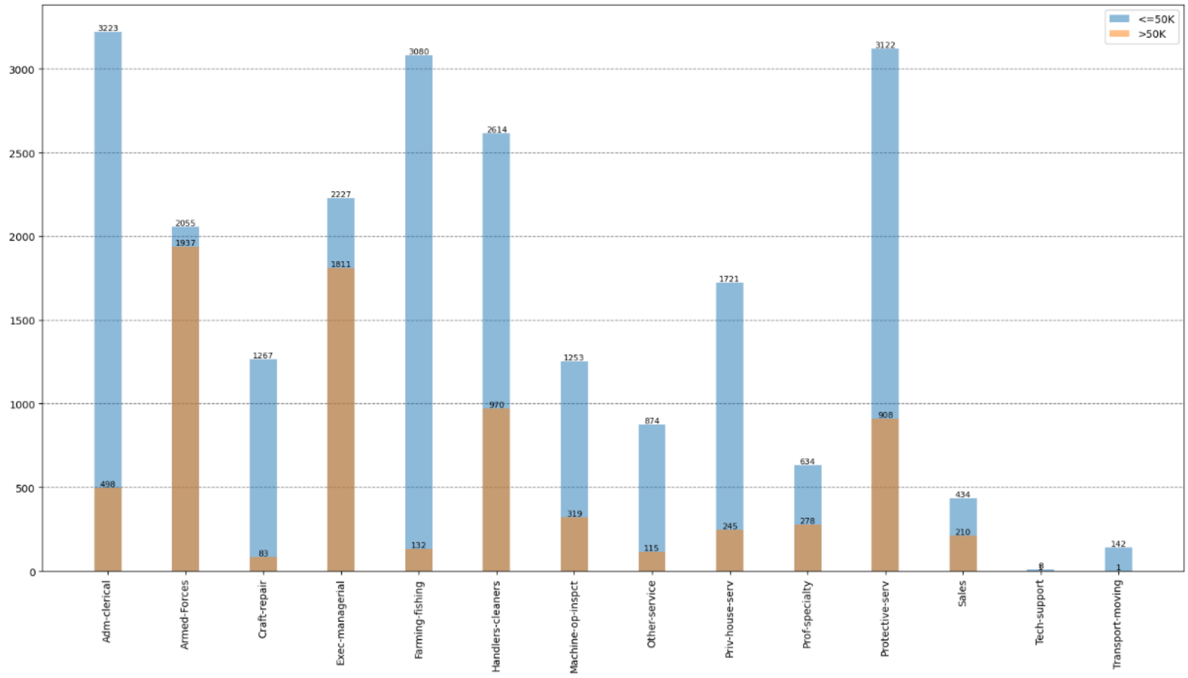


Figure 15: Salary distribution by occupation

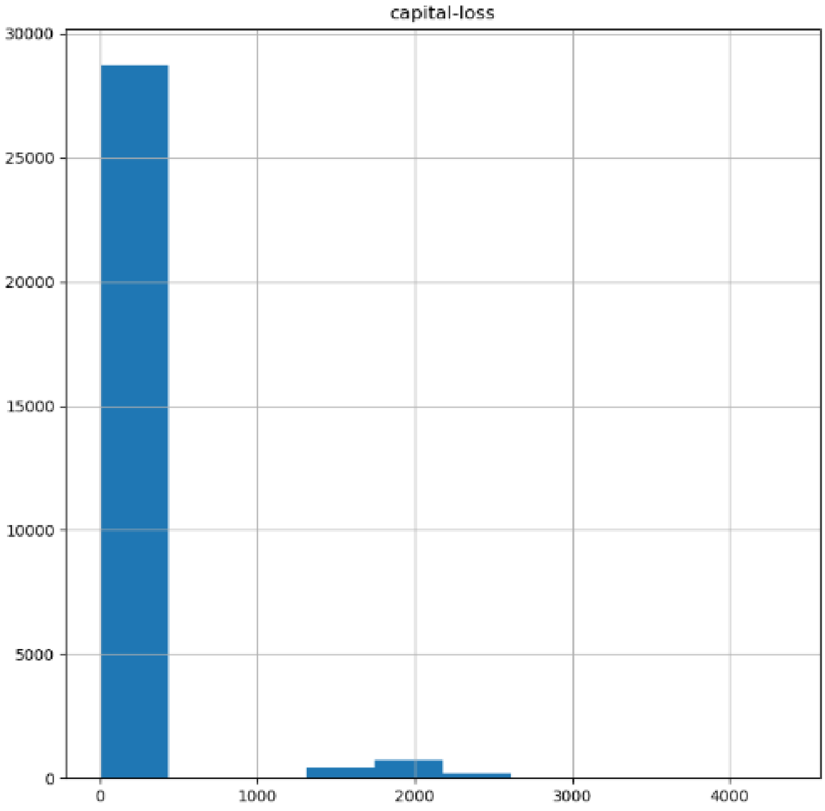


Figure 16: Capital loss distribution

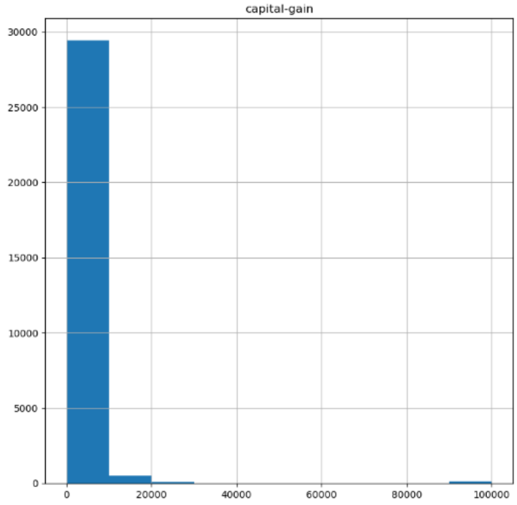


Figure 17: Capital gain distribution

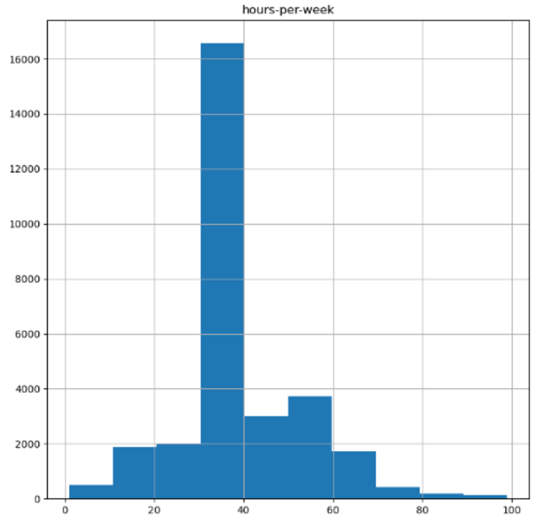


Figure 18: Working hours distribution

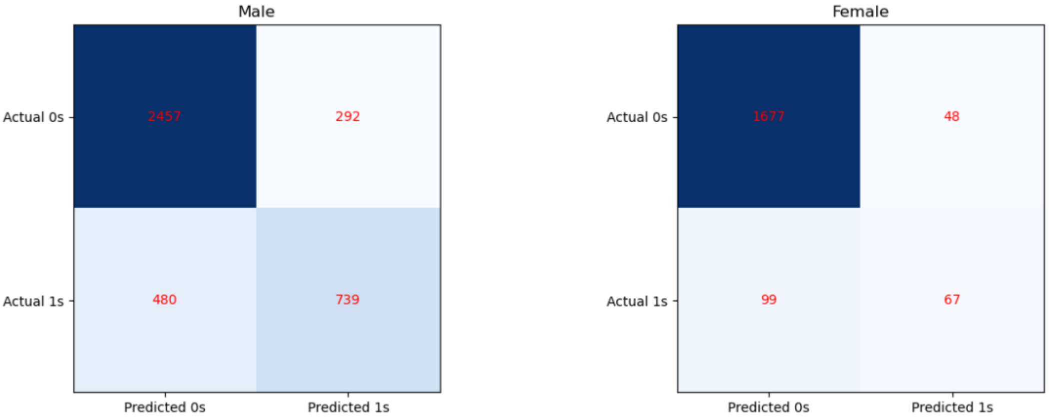


Figure 19: Performance by gender

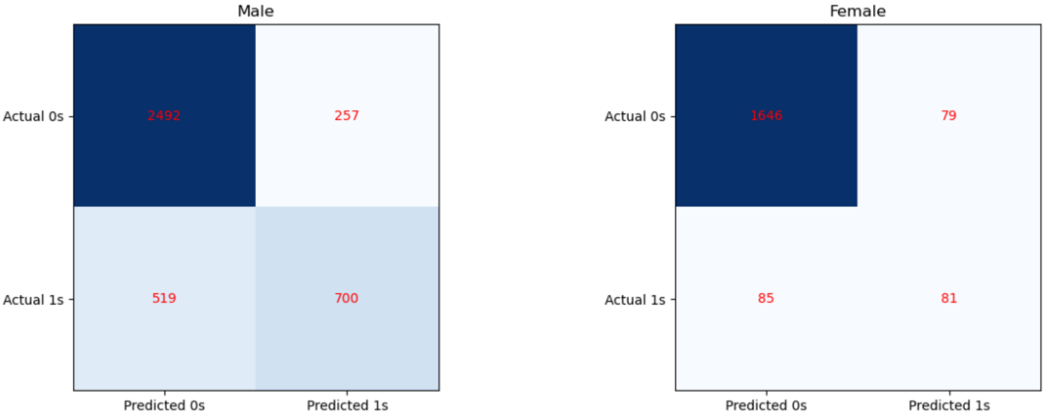


Figure 20: Unbiased performance by gender