Income Prediction

Name:	**Parinita Nema**
Registration No./Roll No.:	2221001
Institute/University Name:	IISER Bhopal
Program/Stream:	DSE
Problem Release date:	January 22, 2023
Date of Submission:	April 16, 2023

1 Introduction

Income prediction is common in various fields such as finance, economics, and social sciences. Accurately predicting income can help individuals and organizations make informed decisions regarding financial planning, investment strategies, and resource allocation. In recent years, machine learning algorithms have been extensively used to predict income based on various factors such as age, education, occupation, etc.

The aim of this project is to increase awareness of the impact of income not only on individuals but also on society as a whole. By using an Income Prediction Model, we can gain a better understanding of consumer and market behavior. The project focuses on analyzing extracted data to determine how different factors influence an individual's income. This analysis will help to identify the role that various features play in predicting an individual's income. Specifically, the project aims to classify individuals earning fifty thousand dollars or less annually, based on factors such as age, occupation, and education. This is a binary classification problem, and we have utilized supervised models including KNN[1], Random Forest[2], Logistic Regression[3], Decision Tree[4], and SVM[5]. The models will be compared based on their F1 measure.

1.1 Dataset

The dataset consists of 43957 rows and 14 columns, with the target variable indicating whether an individual earns more or less than 50000 USD based on all 14 attributes. The dataset includes six numeric columns and eight categorical columns. However, there are null values in the "work class", "occupation", and "native-county" columns. The distribution of the target variable "Income" is unbalanced, with 76.07 % of the values indicating an income less than 50k and 23.93 % indicating an income more than 50k. Figure 1 displays the relationship between the attributes of the dataset and the dependent variable.

1.2 Data Preprocessing

Before fitting our training set to the machine learning models, there are three main data pre-processing steps that we need to carry out: **imputing missing values, one-hot encoding categorical attributes, and feature scaling**. To streamline these steps, we have created a data pre-processing pipeline. Since our missing values are in the categorical column, we first create a pipeline to impute the missing values and then one-hot encode the columns. For the imputer strategy, we use the **mode (most frequent) and replace the missing values with the previous values**. Since categorical values don't have a mean or median. We construct a column transformer that transforms the categorical column using the previously created pipeline and scales the numerical columns. Additionally, we have defined three functions for evaluation: confusion matrix, grid search for splitting and hyper

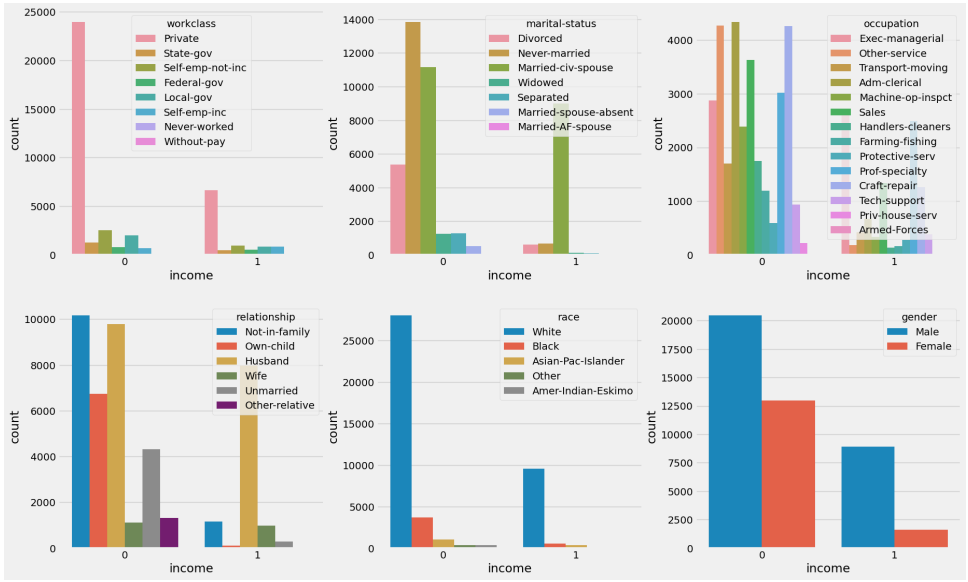

Figure 1: showing attributes of dataset varying with dependent variable

tuning, and prediction. The training set is passed through this pipeline, and we are ready to train our model.

2 Methods

For the binary classification task at hand, we have selected appropriate models for classification. These models include Logistic Regression, Decision Tree Classifier, Random Forest, KNN Classifier, Bernoulli Naive Bayes Classifier, and Support Vector Machines. Our goal is to improve the f1 score, and we have conducted experiments with these models in a particular order:

- The models were fitted on the train set to see how they performed.
- We have created the pipeline to see the best parameters.
- The model was fine-tuned again using the new pipeline and new parameters were chosen based on the best parameters gotten earlier. The evaluation metrics used are the accuracy score and confusion matrix.
- The evaluation metrics used are the accuracy F1 score and confusion matrix

github link

3 Evaluation Criteria

Simply relying on model accuracy may not be sufficient to evaluate a model's performance. Therefore, we have incorporated additional performance measures to evaluate the models.

Table 1: Confusion Matrix

Actual	Predicted	
	0	1
0	TP	FN
1	FP	TN

- Our dataset is not balanced therefore accuracy is not only the correct way to evaluate the model. So we have to follow the f1 score, precision, and recall to evaluate our models.

- True positive tells that model has correctly classified the test dataset that income less than or equal to 50k.
- True negative tells the model has correctly classified that its income is greater than 50k.
- Based on the confusion matrix built for the predictions on the training and test datasets. We have followed the f1 score, precision, and recall to evaluate our models.

4 Analysis of Results

1. Results when missing values are replaced with mode value.

Table 2: Model Evaluation with Hyper Tuning

Classification	**Precision**	**Recall**	**F-measure**	**Accuracy**
KNN	0.79	0.74	0.76	0.84
SVM	0.8	0.75	0.77	0.85
Random Forest	0.83	0.77	**0.79**	0.86
Decision Tree	0.83	0.74	0.77	0.86

Table 3: Model Evaluation without Hyper Tuning

Classification	**F1 score**	**Accuracy**
Logistic Regression	82.55	82.12
KNN	83.99	82.98
SVM	83.39	82.57
Decision Tree	92.02	91.59
Random Forest	**93.43**	93.13
BBernoulli Naive Bayes	78.91	77.59

2. Results when missing values are replaced with the previous value.

Table 4: Model Evaluation without Hyper Tuning

Classification	Accuracy	F1 Score
Logistic Regression	82.21	82.68
KNN Classifier	83.1	84.12
Support Vector Machine	82.67	83.49
Decision Tree Classifier	91.54	91.99
Random Forest Classifier	**92.86**	**93.18**

- The Dataset shows that most people work around 40 hours per week.
- The highest number of people in the training dataset is of age 38 approx.
- Self-employment peeps have a higher probability of getting a salary $> \$50k$.
- We can infer that doctorate and prof-school educated people have more probability of getting salary of $> \$50k$.
- The correlation between income and working hours per week of the person positive correlation i.e as working hours per week increases the probability of having salary $\$50k$ increases.

- In this dataset, the most number of people are young, white, male, high school graduates with 9 to 10 years of education and work 40 hours per week.

Result of different machine learning models: Results of different models and there hyper tuning are shown in Table 2, Table 3, and, Table 4:

- The best parameter of different models after hyper tuning is a follow for Random forest. The best parameter of the random forest model after hyper tuning is when the Best score: 'max_depth': None, 'max_features': auto, 'n estimators': 150, 'min_samples_leaf': 3, 'min_ samples_split: 2
- The best parameter of the KNN model after hyper tuning is when Best parameters: 'knn neighbors': 100, 'knn _ p':1'knn _ weights':'distance'
- The best parameter of SVM after hyper tuning is when 'gamma': = 'scale', Best parameters: 'gamma': 'scale', 'kernel': 'linear'
- The best parameter of the Decision Tree after hyper tuning is when 'criterion': = 'gini', Best parameters: 'criterion': 'gini', 'max_depth': '10', 'min_samples_leaf': '1', min_samples_split: '5'.

5 Discussions and Conclusion

A variety of experiments were conducted using three different machine learning algorithms on the Income dataset in order to predict income classes. When we replace the missing values with mode values, the results indicated that the Random Forest algorithm had the highest test accuracy at 93.43 % and an F1 score of 93.13. Similarly, when we replaced the missing value with the previous value we can see after comparing Table 3 and Table 4 that there are no measurable changes in accuracy and F1 Score. This research demonstrates the potential of machine learning to identify factors contributing to income inequality, and how this knowledge can inform efforts to address the issue.

Moving forward, researchers could consider expanding the dataset to include additional features and attributes, in order to evaluate their impact on the accuracy of the predictive model. Additionally, more advanced machine learning techniques could be explored and compared against the existing models utilized in this study. By continuing to improve our understanding of the factors influencing income inequality, we can take steps toward creating a more equitable society for all.

6 Contribution

Vishesh Kumar has done Univariate and Multivariate Analyses. He has helped me in data preprocessing and also He has evaluated different data models and helped in Hyperparameter tuning of the K-Nearest Neighbor and Decision Tree. Overall we together have finished this project and Report.

References

[1] Gongde Guo, Hui Wang, David Bell, Yaxin Bi, and Kieran Greer. Knn model-based approach in classification. In *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings*, pages 986–996. Springer, 2003.

[2] Gérard Biau and Erwan Scornet. A random forest guided tour. *Test*, 25:197–227, 2016.

[3] Michael P LaValley. Logistic regression. *Circulation*, 117(18):2395–2399, 2008.

[4] Sisay Menji Bekena. Using decision tree classifier to predict income levels. 2017.

[5] Alina Lazar. Income prediction via support vector machine. In *ICMLA*, pages 143–149, 2004.