Adult Income Prediction in US Based on Census Data

Shivam Pal
Dept. of Data Science and Analytics
Central University of Rajasthan
Ajmer, India
email: shivampal7722@gmail.com

Kishan Pahadiya
Dept. of Data Science and Analytics
Central University of Rajasthan
Ajmer, Rajasthan
email: kishanvir4321@gmail.com

Raushan Kumar
Dept. of Data Science and Analytics
Central University of Rajasthan
Ajmer, Rajasthan
email: idealraushan@gmail.com

***Abstract*—The prominent inequality of wealth and income is a huge concern in the United States. The principle of universal moral equality ensures sustainable development and improve the economic stability of a nation.**
In this paper, our work aim is to analyse and predict whether a person's income of U.S is above 50K dollars per year or below above 50K dollars per year based on several attributes from the census data. The Adult Census Income data was extracted from 1994 US Census database by Ronny Kohavi and Barry Becker. The dataset contains features about the people and the labels which we have to predict and the labels are discrete and binary, So the problem is a Supervised Classification type.
This study shows the usage of machine learning and data mining technique that provide a solution to the income equality problem. The adult census dataset has been used for this purpose.

Keywords - Data Analysis, Classification, Machine Learning, LGBM Classifier, Data mining.

I. Introduction

Since the start of the twenty-first century, the volume of data produced annually has been rapidly rising. This vast amount of information can be used by the fields of data mining and machine learning to simplify daily life. Additionally, ML can make use of this data to look into specific obscure models and ideas that led to the estimation of future events. The census data, which is gathered by the census bureau every ten years, is one of the largest structured data sets. We can utilize this information to anticipate an individual's income in the future.

The dataset contains 48842 records with various attributes. Exploratory data analysis will be doe between dependent and independent variables and various insights are concluded from EDA.

The main purpose of the census dataset is to classify the income of peoples are 50K or less than 50K based on certain features like – education, occupation, native-country, age and gender etc. This project is based on supervised machine learning algorithm.

With the help of this model we can find out whether a person is getting the correct salary or not according to their current occupation and education qualification.

From this algorithm it can be known that to which class that person belongs like lower, middle and upper class and also the government can classify which families should get the benefit of welfare schemes.

II. Literature Review

Bhavin Patel et. al. [1] Analysed census dataset and used several algorithms like SVM, K-Means, Logistic Regression, Decision Forest, Boosted Decision Tree Regression, Bayesian Linear Regression, Neural Network with the accuracy of 90%.

Navoneel Chakrabarty et. al. [2] Applied the application of Ensemble Learning Algorithm like Gradient Boosting Classifier with extensive Hyper-Parameter Tuning, XGBoost and also used PCA. Finally, the Validation Accuracy, so obtained, 88.16%.

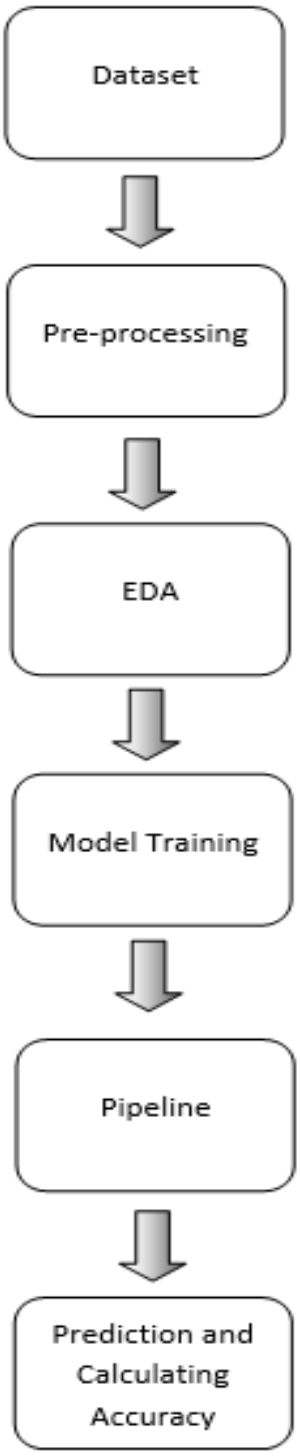
Sumit Mishra et. al. [3] Implemented the logistics regression, Random forest classifier, Gradient boosting, BernoulliNB Classifier, Support Vector Classifeir with final accuracy of 86.99%.

Ahmed Ali et. al. [4] Choosing the Decision Tree Classifier whose max-leaf-nodes set to 16 branches with the accuracy of 84.69%.

Jinglin Wang et. al. [5] Explored the five supervised machine learning algorithms which are Random forest classifier, K Nearest Neighbour (KNN), Logistic Regression, Support Vector Machine (SVM) and Naive Bayes algorithm in which Random Forest model is achieved 97.98%.

III. Methodology

The dataset is first collected from the Kaggle platform. First thoroughly analysed the data to understand it properly. The dataset is first pre-processed. For handling the missing values analyzed the entire dataset, then it was found that the dataset contains two different forms of symbolic null values (i.e. '?' and '0'). Then replaced the symbolic null values with NaN and after that fill all null values with the help of SimpleImputer.



Procedure fore implementing methodology

A. *Data-set*

The dataset contains 48842 records with various attributes.
RangeIndex: 48842 entries, 0 to 48841
Data columns (total 15 columns)
dtypes: int64(6), object(9)
memory usage: 5.6+ MB

S.No	Column	Non-Null Count	Dtype
0	age	48842 non-null	int64
1	workclass	48842 non-null	object
2	fnlwgt	48842 non-null	int64
3	education	48842 non-null	object
4	educational-num	48842 non-null	int64
5	marital-status	48842 non-null	object
6	occupation	48842 non-null	object
7	relationship	48842 non-null	object
8	race	48842 non-null	object
9	gender	48842 non-null	object
10	capital-gain	48842 non-null	int64
11	capital-loss	48842 non-null	int64
12	hours-per-week	48842 non-null	int64
13	native-country	48842 non-null	object
14	income	48842 non-null	object

B. *Pre-Processing*

*1) **Handling the Null values**:* Columns which have **"?"** or missing values are :
* workclass (categorical)
* occupation (categorical)
* native-country (categorical)

Columns which have **"0"** or missing values are :
* capital-gain
* capital-loss

- **replacing '?' with NaN.**

```
df.replace('?',np.nan,inplace=True)
df.isnull().sum()
```

```
age                0
workclass          2795
fnlwgt             0
educational-num    0
marital-status     0
occupation         2805
relationship       0
race               0
gender             0
capital-gain       0
capital-loss       0
hours-per-week     0
native-country     856
income             0
dtype: int64
```

- **Performing SimpleImputer.**

```
imputer = SimpleImputer(strategy='most_frequent')
imputer
```

```
SimpleImputer
SimpleImputer(strategy='most_frequent')
```

*2) **Handling duplicate values**:* The dataset had 52 number of duplicated rows that have been removed.

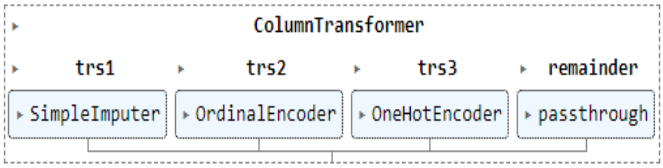
*3) **Encoding of features**:* By using SimpleImputer, OneHotEncoder and OrdinalEncoder categorical features are converted into numerical features with the help of column transformation.

• OneHotEncoder : It can be used to transform one or more categorical features into numerical features (dummy features). OneHotEncoder is a class of sklearn library.
Here, 'marital-status', ' relationship', 'race', 'gender', 'native-country' features have not any ordered relations. So, we used the OneHotEncoder for encoding in numerical features.

• OrdinalEncoder : It is similar to Label Encoding where a list of categories (features) converted into integers.
Here, 'workclass' 'occupation' features are ordered features, because of that used the OrdinalEncoder to covert into numerical form.

• SimpleImputer : In SimpleImputer missing values can be imputed with a provided constant value (i.e mean, median or most-frequent) of each column in which the missing values are located.
Here, 'age', 'fnlwgt' and 'hours-per-week' features are converted in numerical features through the SimpleImputer.

```
transformer1 = ColumnTransformer(transformers=[
    ('trs1',SimpleImputer(),[0,2,11]),
    ('trs2',OrdinalEncoder(),[1,5]),
    ('trs3',OneHotEncoder(),[4,6,7,8,12])],
    remainder='passthrough')
transformer1
```



C. *EDA*

Done the exploratory data analysis is to learn about the feature dependencies and gain insights into the data.

a) UNIVARIATE-Analysis : The simplest type of data analysis is called a uni-variate analysis. Uni stands for one, hence there is just one variable in the data. Each variable must be analysed separately for uni-variate data.

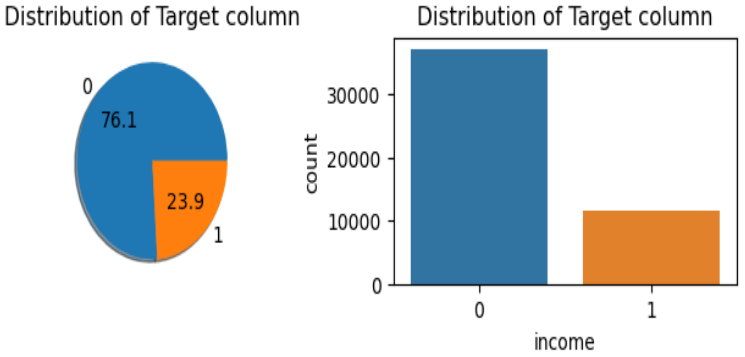


Fig. - 1

According to the Fig. - 1 shows the people who earn less or more than 50K dollar grouped according to number of peoples.

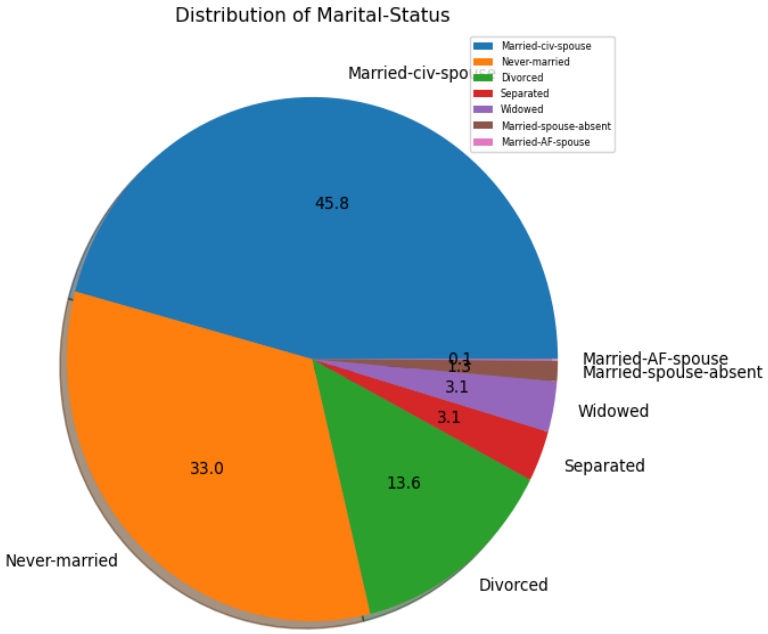


Fig. - 2

According to the Fig. - 2 the pie chart shows the distribution of marital-status column values in the form of percentage. feature values are :

S.No	values
1	Married-civ-spouse
2	Never-married
3	Divorced
4	Separated
5	Widowed
6	Married-spouse-absent
7	Married-AF-spouse

b) BIVARIATE-Analysis : One type of statistical analysis where two variables are observed is known as bi-variate analysis. Here we have a dependent variable and an independent variable. Therefore, in this section, we examine how much the two variables changed from one another.

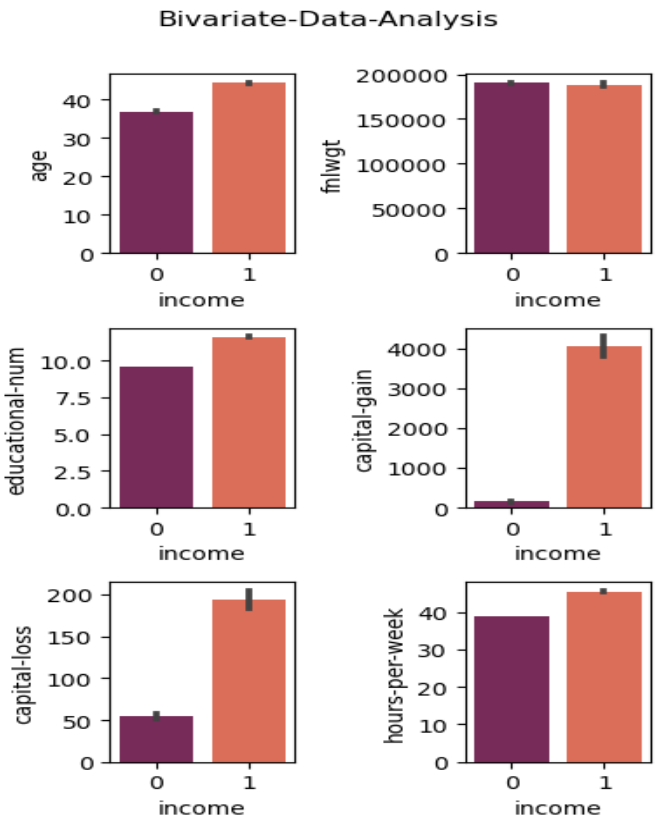


Fig. - 3

Observations :

- The bar plot above are bi-variate plots.
- In terms of age, older the person, more is the probability of income getting higher.
- capital-gain and capital-loss are more experienced by people having higher income.
- People having higher income are working more hours-per-week compare to people with lower income.

D. *Feature Engineering and Selection*

In the data, two columns named 'education' and 'educational-num' were showing the same type of details. Out of which the 'educational-num' column is showing data in numerical format while the other 'education' column is showing data in string format. In Machine Learning algorithm data always will be given in numerical format, so the 'education' column which is in string form was dropped.

E. *Algorithms to be used*

We will split the dataset into training and testing data before we implement any algorithms so that we can train the model using the training data. The results of the testing data can be used to measure the accuracy of different algorithms.

We will try applying different algorithms like logistic regression, decision trees, Random Forest and LGBMClassifier etc. The description of these algorithms are given below.

*1) **Logistic Regression**:* Based on a given dataset of independent variables, logistic regression calculates the likelihood that an event will occur, such as voting or not voting. Given that the result is a probability, the dependent variable's range is 0 to 1. In logistic regression, the odds—that is, the probability of success divided by the probability of failure—are transformed using the logit formula.

LogisticRegression(solver='liblinear', penalty='l1')

Logistic Regression formula are given in below :

$$P = \frac{e^{a+bX}}{1+e^{a+bX}} \quad \text{or} \quad P = \frac{1}{1+e^{-(a+bX)}}$$

*2) **Decision Tree**:* A decision tree is a supervised machine learning tool that may be used to classify or forecast data based on how queries from the past have been answered. The model is supervised learning in nature, which means that it is trained and tested using data sets that contain the required categorisation.

DecisionTreeClassifier(max_depth=5)

Decision tree formula are given in below :

$$Information\,Gain = 1 - Entropy$$

or

$$Entropy = -\sum_{i=1}^{n} p_i \log_2 p_i$$

*3) **Random Forest Classifier**:* Regression or classification issues can be resolved with the random forest classifier. Each decision tree in the ensemble of the random forest method is made up of a data sample taken from a training set with replacement, known as the bootstrap sample.

RandomForestClassifier(n_estimators=50, random_state=2)

Random forest classifier formula are given in below :

$$Gini = 1 - \sum_{i=1}^{C} (p_i)^2$$

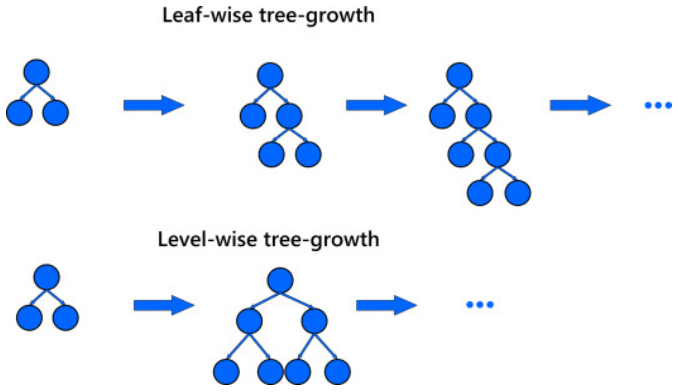
or

$$Entropy = \sum_{i=1}^{C} -p_i * \log_2(p_i)$$

*4) **LGBM Classifier**:* LightGBM, also known as the light gradient-boosting machine is based on decision tree algorithms and used for ranking, classification and other machine learning tasks.
While other algorithms grow trees horizontally, Light GBM grows trees vertically, which converts to Light GBM growing trees leaf-wise while other algorithms grow levels-wise. The leaf with the greatest delta loss will be chosen to grow. Leaf-wise method can reduce loss more than a level-wise strategy when expanding the same leaf.

LGBM classifier formula are given in below :

$$\tilde{V}_j(d) = \frac{1}{n}\left(\frac{\left(\sum_{x_i \in A_l} g_i + \frac{1-a}{b}\sum_{x_i \in B_l} g_i\right)^2}{n_l^j(d)} + \frac{\left(\sum_{x_i \in A_r} g_i + \frac{1-a}{b}\sum_{x_i \in B_r} g_i\right)^2}{n_r^j(d)}\right)$$



IV. Results and Analysis

Out of 48842 records 39032 records have been utilized for training model while the left 9758 records have been splitted for testing the models. After complete execution of models following measurements are as follows:
Two metrics, precision and recall are used to evaluate the performance of classification model.

Recall : the capacity of a model to locate all relevant instances in a data-set. Recall is calculated mathematically as the number of true positives divided by the sum of the true positives and false negatives.

$$Recall = \frac{TP}{TP + FN} \quad (1)$$

Precision : A classification model's capacity to identify only the relevant data points. Precision is calculated mathematically as the number of true positives divided by the sum of the true positives and false positives.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

F1-Score: F1-Score is a harmonic mean of Precision and Recall.

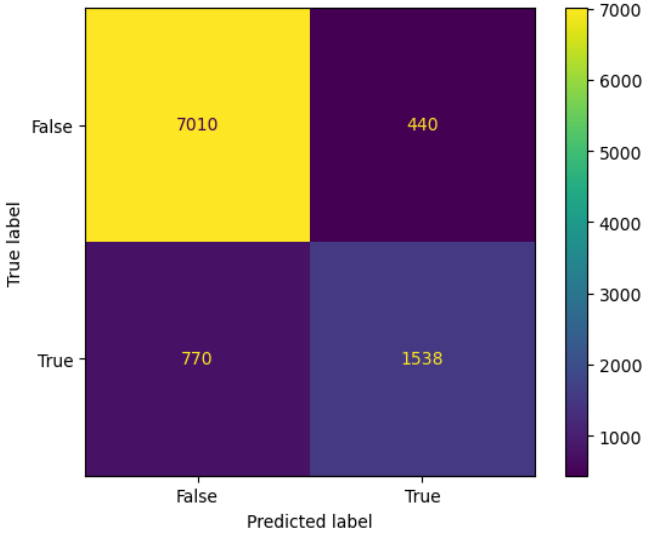
$$F1 - Score = 2 \times \frac{(Precison \times Recall)}{(Precison + Recall)} \quad (3)$$

Confusion matrix : A confusion matrix is a complete summary of prediction results on a classification problems.
Confusion matrices have two types of errors: Type-I and Type-II

Type-I error is known as "false positive".
Type-I error is known as "false-negative".

		Actual Class	
		Positive (**P**)	Negative (**N**)
Predicted Class	Positive (**P**)	True Positive (**TP**)	False Positive (**FP**)
	Negative (**N**)	False Negative (**FN**)	True Negative (**TN**)

The final confusion matrix generated by our model is given as :



Accuracy of models are given as :

	Algorithm	Accuracy	f1_score
3	LGBM Classifier	88.0%	72.0%
2	Random Forest Classifier	86.0%	67.0%
0	Logistic Regression	85.0%	64.0%
1	Decision Tree	86.0%	64.0%

Logistic Regression, which has an accuracy of 85.0% with f1-score 64.0%.

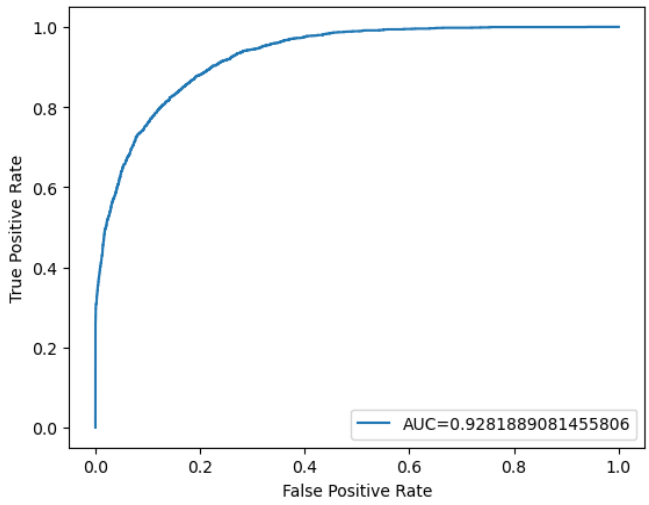
Decision Tree, which has an accuracy of 86.0% with f1-score 64.0%.

Random Forest Classifier, which has an accuracy of 86.0% with f1-score 67.0%.

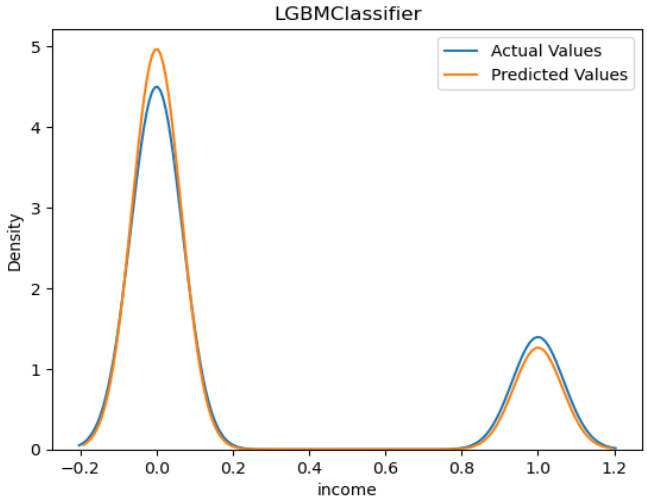
Here, our last model trained was the LGBM Classifier, which has an accuracy of 88.0% with f1-score 72.0%.

ROC Curve : The performance of two or more diagnostic tests is compared using the ROC curve, which is used to evaluate a test's overall diagnostic performance. It is also used to choose the best cut-off value for assessing whether a disease is present or not. Although clinicians without a background in statistics do not necessarily need to comprehend both the intricate mathematical equation and the ROC curve analysis procedure, doing so is necessary for the ROC curve to be used and interpreted correctly. The area under the ROC curve (AUC), the partial AUC are described in this paper.

AUC stands for the level or measurement of separability, and ROC is a probability curve. It reveals how well the model can differentiate across classes.



The plot of actual values versus predicted values is as follows:



V. Conclusion

This paper proposed the application of Ensemble Learning Algorithm like Random Forest Classifier and Light GBM Classifier. The main objective was to compare the performance of several classification algorithms in order to predict the person's income from the data-set. From the observed data, we found that the performance of the LGBM Classifier is the best with the accuracy of 88.0% and f1-score 72.0%.
The future scope of this work includes adopting hybrid models that combine Machine Learning and Deep Learning, or by using numerous more advanced pre-processing approaches without further reducing the accuracy, to achieve an overall superior set of findings.

References

References

[1] Patel, Bhavin Kakulapati, Vijayalakshmi Balaram, V V S S S. (2017). International Journal on Recent and Innovation Trends in Computing and Communication Comparative Analysis of Classification Models on Income Prediction.
[2] Chakrabarty, N., Biswas, S. (2018, October). A statistical approach to adult census income level prediction. In 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN) (pp. 207-212). IEEE.
[3] shorturl.at/lopQ6
[4] Ali, Ahmed. (2020). Machine-learning analysis for adult census income dataset.
[5] Jinglin Wang Year: 2022 Research on Income Forecasting based on Machine Learning Methods and the Importance of Features ICIDC EAI DOI: 10.4108/eai.17-6-2022.2322745