Project: Census Income Prediction

Kesav Adithya Venkidusamy

Census Income Prediction

Introduction

Income prediction is important for a variety of areas in the private and nonprofit sectors. One critical area this affects is marketing, where income segmentation of the population is an extremely important tool. Businesses may make different variations of their items designated for certain subgroups of the population, and these subgroups often include the income of individuals. Income prediction also helps to identify those individuals who are of a lower income that may need the most assistance, who some nonprofits strive to identify and assist. The ability to predict the income of individuals from this information has far-reaching impacts for every industry.

The data for our project was extracted from the 1994 Census bureau database by Ronny Kohavi and Barry Becker. The problem statement of this project is to identify the dataset feature or features which are mostly related to or affecting the income of a household. With the dataset having 48,842 records, we will be able to predict an answer to our problem statement. The dataset consists of 15 features of which 6 are numerical and rest are categorical with “income” being the target. The target variable income contains 2 values. It will indicate if the individual's income is less than 50 thousand dollars per year or greater than 50 thousand dollars income per year.

A set of clean records was extracted using the following conditions: ((AAGE>16) && (AGI>100) && (AFNLWGT>1) && (HRSWK>0))

Abstract: Predict whether income exceeds $50K/year based on census data

Data Set Characteristics	Multivariate
Attribute Characteristics	Categorical, Integer
Associated Tasks	Classification
Number of Instances	48842
Number of Attributes	14
Missing Values	Yes
Area	Social
Date Donated	5/1/1996

Attribute Information:

Feature Name	**Feature Description**	**Feature Type**
age	Age of the person	Continuous
workclass	Work class of the person	Discrete
fnlwgt	Final Weight	Continuous
education	Education of the person	Discrete
education-num	Number of years the person	Continuous
marital-status	Marital status of the person	Discrete
occupation	Occupation of the person	Discrete
relationship	Relationship of the person to the family	Discrete
race	Race of the person	Discrete
sex	Sex of the person	Discrete
capital-gain	Capital Gain	Continuous
capital-loss	Capital Loss	Continuous
hours-per-week	Hour the person worked for a week	Continuous
native-country	Native country	Discrete
Income	Income of the person	Target

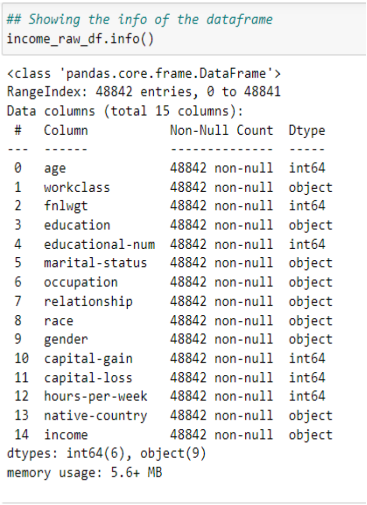
Exploratory Data Analysis

The problem statement of this project is to identify the dataset feature(s) which are mostly related to or affecting the income of household. With dataset having total number of records as 48842, we would be able to predict or answer our problem statement. The dataset consists of 15 features of which 6 are numerical and rest all are categorical with “income” being the target. The target variable income contains 2 values <=50K and >50 which would be subsequently converted to 0 and 1 respectively. The details are shown in figure 1.

Among 14 features, we see missing/null values present only for below features. I have given the percentage of missing values for each of the feature in table 1. We noticed that the values where ‘workclass’ is missing, also has ‘occupation’ missing. While trying to identify the extra rows where ‘occupation’ is missing, we observed the workclass is ‘Never-Worked’. Since the percentage of null values present in these features is low, the rows will be removed from the dataset.

Feature Name	# Of missing Values	Percentage
workclass	2799	5.7%
occupation	2809	5.8%
native-country	857	1.8%

Table 1: Features with null values and percentage



```
## Showing the info of the dataframe
income_raw_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48842 entries, 0 to 48841
Data columns (total 15 columns):
 #   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
dtypes: int64(6), object(9)
memory usage: 5.6+ MB
```

Figure 1: Features and dtypes

The income column is our target variable with 2 values - ‘<=50K’ and ‘>50K’. The count of these values is 37155 and 11687 respectively, suggesting that people with income higher than 50K are significantly less, and our data set is kind of imbalanced considering the target variable. However, we will evaluate the outcome and apply filter to the dataset, if required.

Visualizations

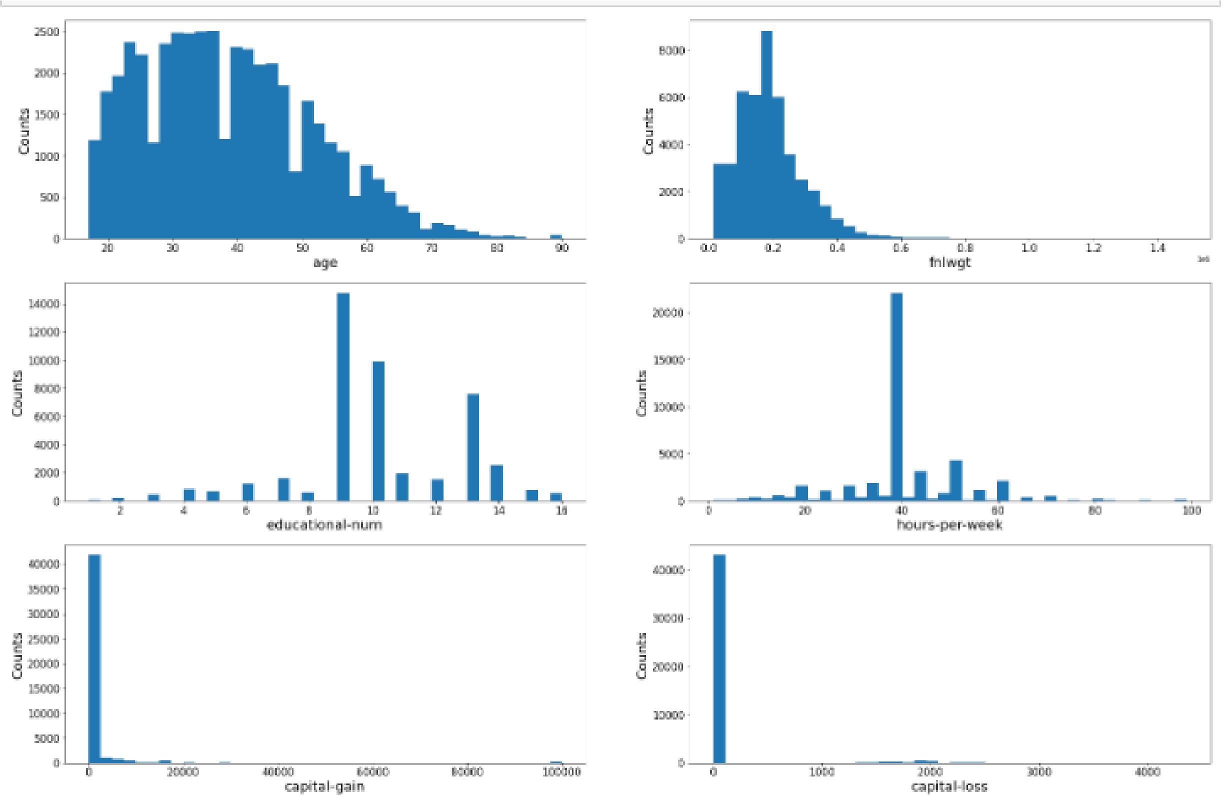
As mentioned before, the dataset contains 6 numerical features and 8 categorical features as follows, and ‘income’ feature being the target variable.

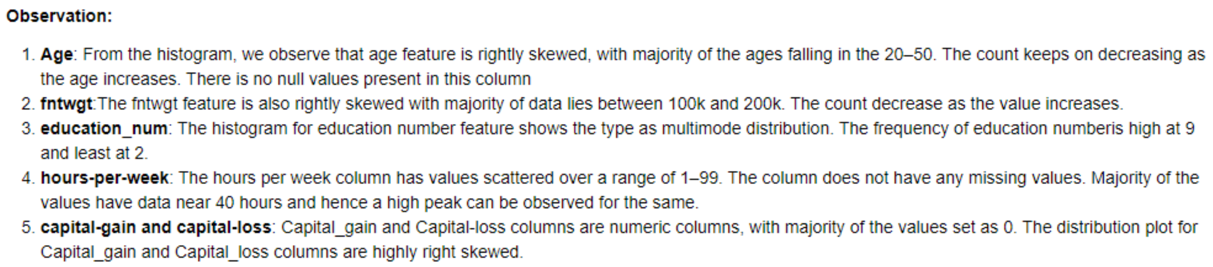
Numerical: age, fnlwgt, educational-num, hours-per-week, capital-gain, capital-loss

Categorical: workclass, education, marital-status, occupation, native-county, relationship, race, gender

The following are the visualizations used based on nature of the feature.

Histogram*:* Histogram is used to identify the distribution of numerical features present in the dataset.

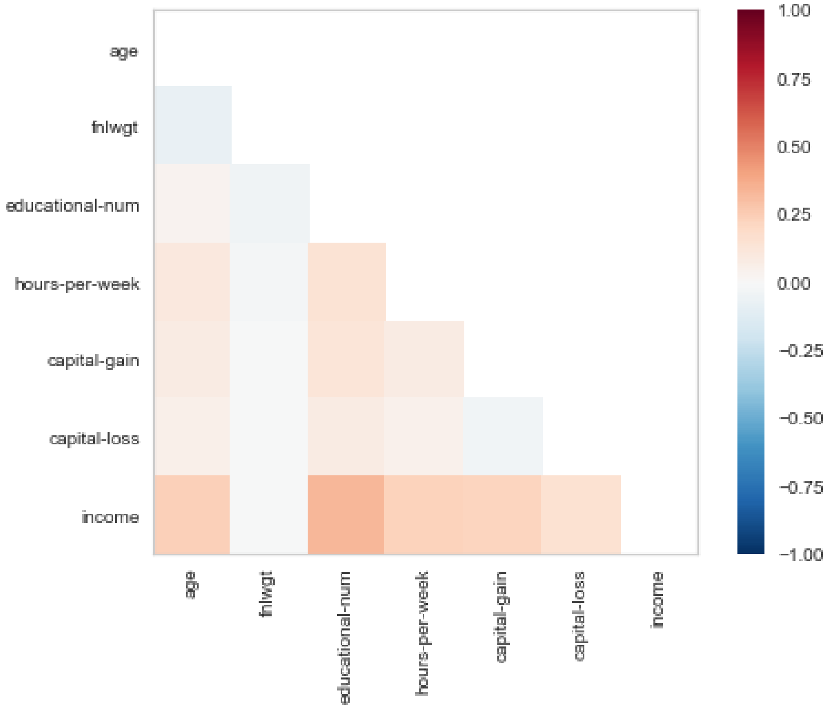




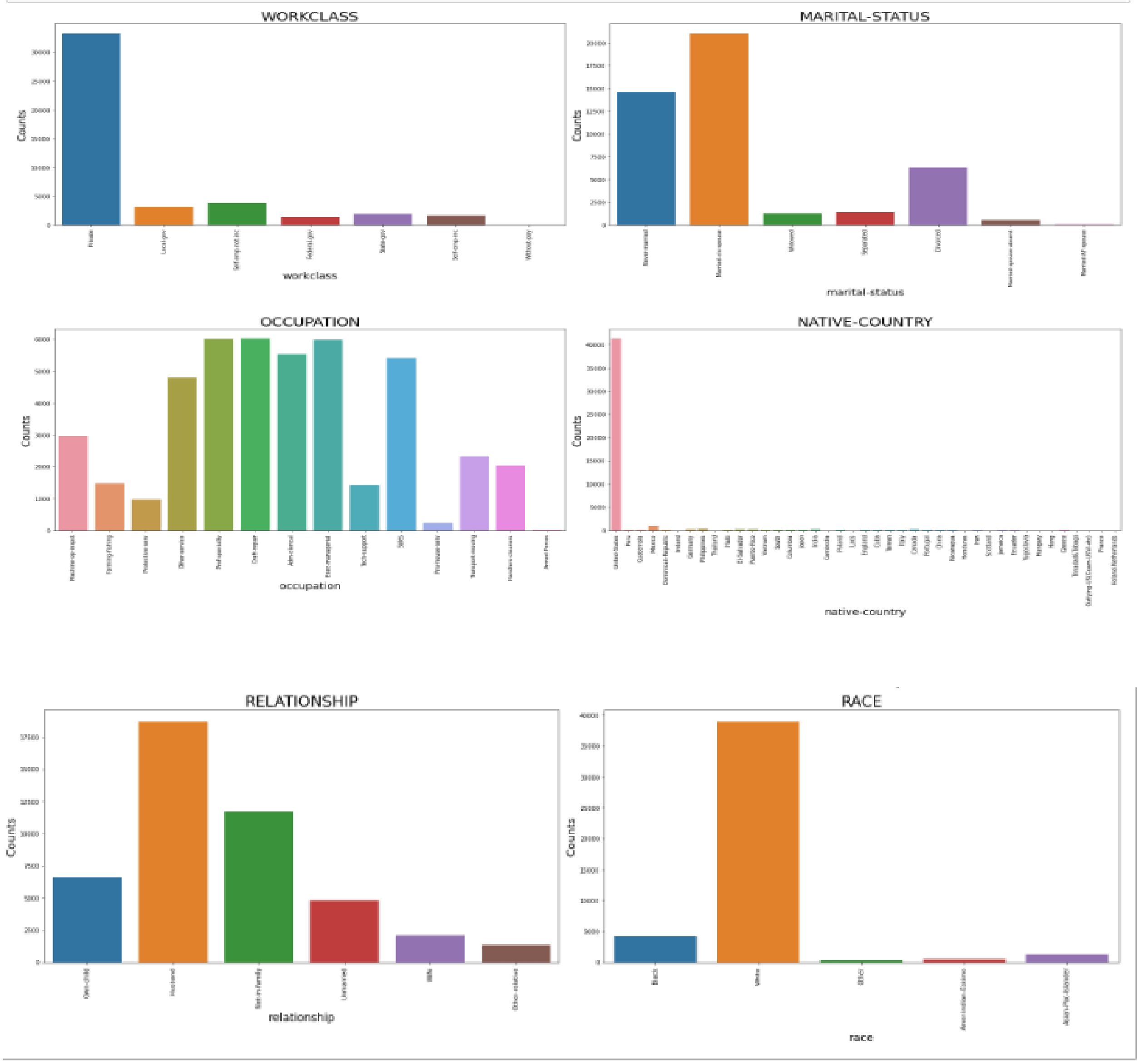
Observation:

1. **Age**: From the histogram, we observe that age feature is rightly skewed, with majority of the ages falling in the 20–50. The count keeps on decreasing as the age increases. There is no null values present in this column
2. **fntwgt**:The fntwgt feature is also rightly skewed with majority of data lies between 100k and 200k. The count decrease as the value increases.
3. **education_num**: The histogram for education number feature shows the type as multimode distribution. The frequency of education numberis high at 9 and least at 2.
4. **hours-per-week**: The hours per week column has values scattered over a range of 1–99. The column does not have any missing values. Majority of the values have data near 40 hours and hence a high peak can be observed for the same.
5. **capital-gain and capital-loss**: Capital_gain and Capital-loss columns are numeric columns, with majority of the values set as 0. The distribution plot for Capital_gain and Capital_loss columns are highly right skewed.

Heat Map*:* Heat map has been created to understand Pearson's correlation between the target variable 'income' and other numerical variables. We observed that all numerical features have positive correlation with the target except fnlwgt feature. Among the features having positive correlation, age and education-num features are having high value. The details are shown in the below heat map chart.



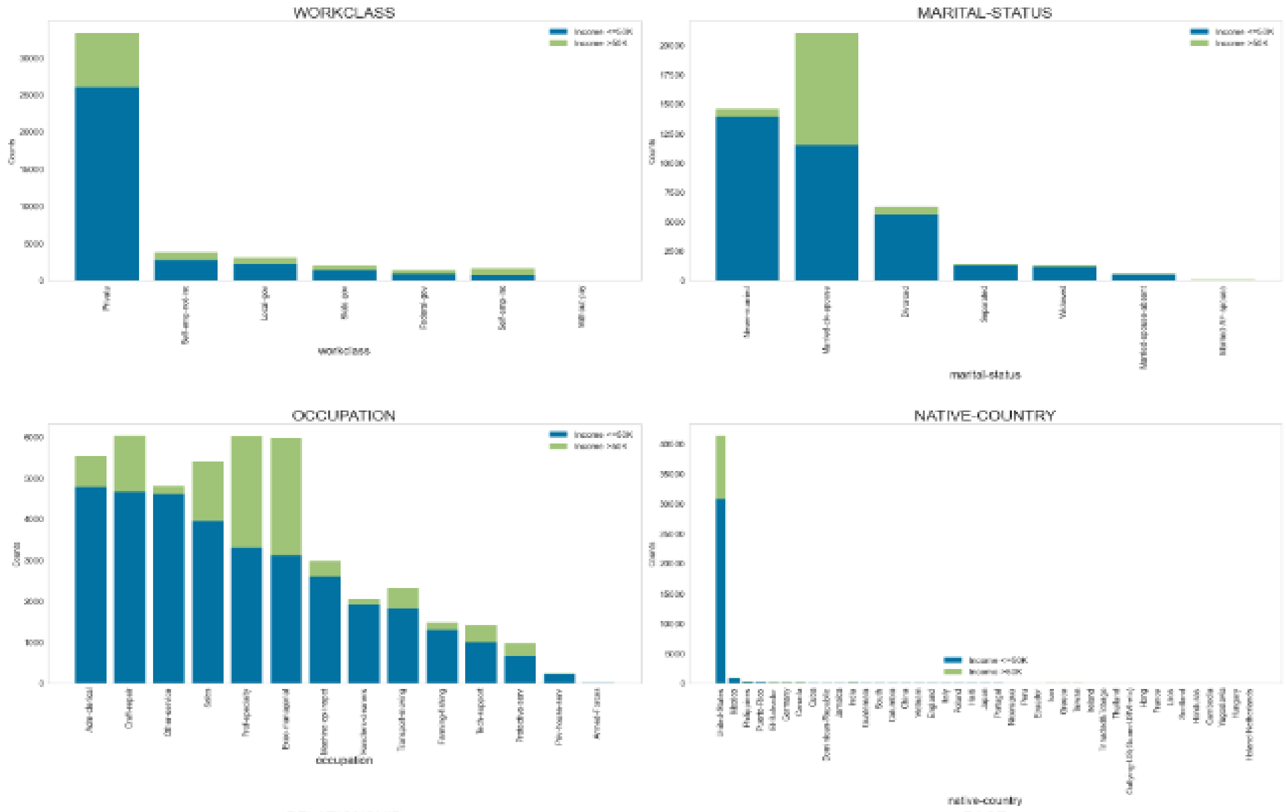
Bar and Stacked Bar Graph: Bar graph has been plotted for all categorical features to understand the distribution of data among unique values. Stacked bar chart has been plotted to compared those earning less than or equal to 50K (represented as 0) and greater than 50K (represented as 1) for all the categorical features.

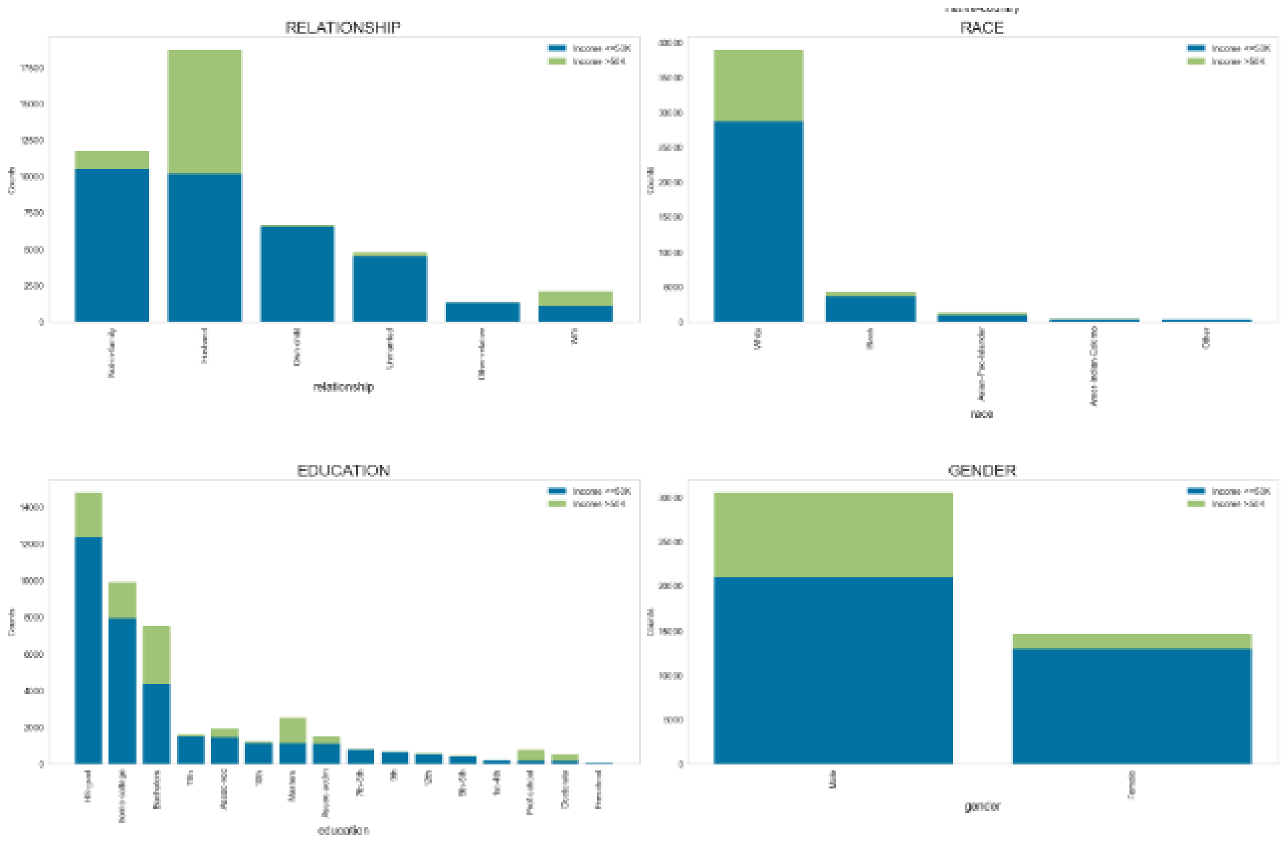


Observation: ¶

1. **workclass**: There are 7 unique values present for this feature. We had around 2799 null values present in this column which contributes only 5% of total records. So, all these null values are removed from the dataset. Based on the result, these null values might be replaced with mode. Majority of the people belong to 'Private' sector workclass. The values where 'Workclass' is missing, also has 'Occupation' missing!
2. **marital-status**: The 'Marital_Status' column has 7 different categories available, and has no missing values. Majority of the people have 'Marital_Status' as 'Married-civ-spouse', and least have 'Married-AF-spouse'. Count of 'Never-married' is also quite high
3. **occupation**: The occupation column contains 14 different categories, and have missing values (which we have already observed, and combined with 'Workclass' column).
4. **native-country**: The Native_country column contains the highest count set to 'United-States', and rest of the rows contain quite few numbers. We also have 857 missing values in this column, which are removed from the datasets.
5. **relationship**: The relationship column contains 6 different types of values, with highest number set for 'Husband' and lowest for 'Other-relative'. The column does not have any missing value.
6. **race**: The Race column has 5 different categories, and no missing data. Highest number of people have race as 'White' (significantly high numbers).

```
<Figure size 2160x2160 with 0 Axes>
```



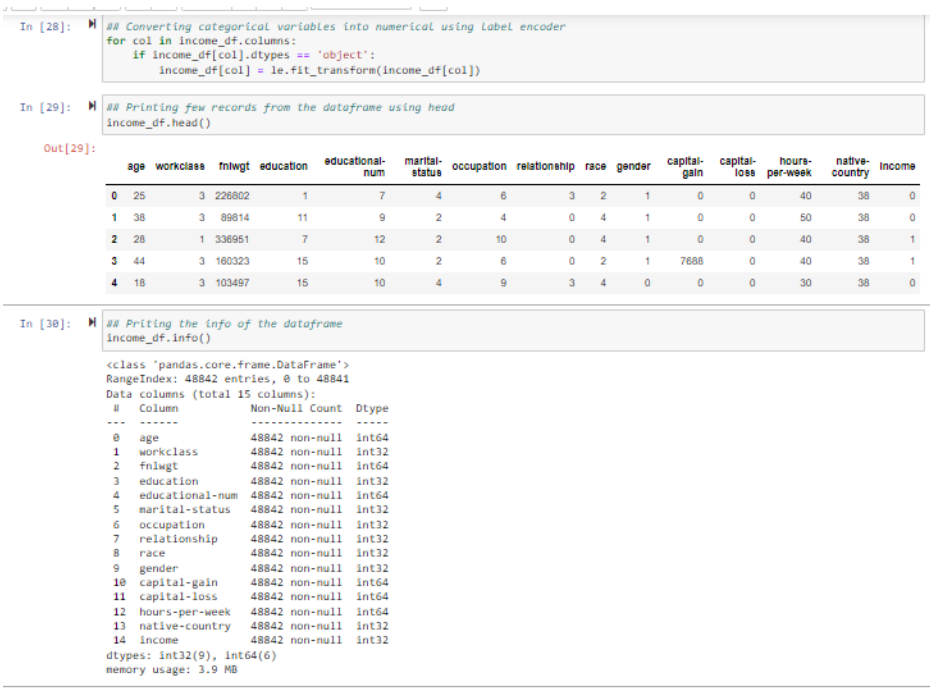


Observation:

1. **workclass**: Majority of the people belong to 'Private' sector workclass. Among Privte class most of the people belong to <=50K incme group. Most of the people belong to <=50K compared to >50K for all other income group as well.
2. **marital-status**: The 'Marital_Status' column has 7 different categories available, and has no missing values. Majority of the people have 'Marital_Status' as 'Married-civ-spouse', and least have 'Married-AF-spouse'. Count of 'Never-married' is also quite high. The data is almost equally split between the income group for "Married-civ-spouse'
3. **occupation**: The occupation column contains 14 different categories, and have missing values (which we have already observed, and combined with 'Workclass' column). Except few values where the data is equally split between the income class, all other group have higher count for <=50K income.
4. **native-country**: The Native_country column contains the highest count set to 'United-States', and rest of the rows contain quite few numbers. We also have 857 missing values in this column, which are removed from the datasets. The count is higher for <=50K income group compared to other for US.
5. **relationship**: The relationship column contains 6 different types of values, with highest number set for 'Husband' and lowest for 'Other-relative'. The column does not have any missing value. If the relationship in family is either 'Husband/Wife', the chances of earning more than 50K is high.
6. **race**: The Race column has 5 different categories, and no missing data. Highest number of people have race as 'White' (significantly high numbers). A person has high chance of earning >50K in case his/her race is 'White'/'Asian-pac-islander'.
7. **education**: People with education level as 'Masters/Doctorate/Prof-school' have higher ratios of >50K earning, than <=50K. Bachelors degree also has around 10:7 ratio of <=50K : >50K.
8. **gender**: Males have a higher chance of earning more than 50K, than females.

While evaluating the data for null values, the workclass column has 2799 null values, occupation column has 2809 null values, and native-country column has 857 null values. The mode was calculated for each column and was used to replace the null values respectfully. A comparison was done to find duplicates within the data. A duplicate was decided using all columns for comparison. This evaluation determined 52 duplicates which were removed from the base dataset accordingly. After much discussion, we decided to use all the columns presented in the original data load.

We started by using a label encoder method for the categorical variables, see figure 4.1. After the data was split into train and test, this method was then applied to our model choices and the accuracy measurements were evaluated. We then applied a standard scaler to our train and test data and applied the same model choices. Those results helped us decide the best encoding method.



```
In [28]:  ## Converting categorical variables into numerical using label encoder
          for col in income_df.columns:
              if income_df[col].dtypes == 'object':
                  income_df[col] = le.fit_transform(income_df[col])
```

```
In [29]:  ## Printing few records from the dataframe using head
          income_df.head()
```

Out[29]:

	age	workclass	fnlwgt	education	educational-num	marital-status	occupation	relationship	race	gender	capital-gain	capital-loss	hours-per-week	native-country	income
0	25	3	226802	1	7	4	6	3	2	1	0	0	40	38	0
1	38	3	89814	11	9	2	4	0	4	1	0	0	50	38	0
2	28	1	336951	7	12	2	10	0	4	1	0	0	40	38	1
3	44	3	160323	15	10	2	6	0	2	1	7688	0	40	38	1
4	18	3	103497	15	10	4	9	3	4	0	0	0	30	38	0

```
In [30]:  ## Priting the info of the dataframe
          income_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48842 entries, 0 to 48841
Data columns (total 15 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   age              48842 non-null  int64
 1   workclass        48842 non-null  int32
 2   fnlwgt           48842 non-null  int64
 3   education        48842 non-null  int32
 4   educational-num  48842 non-null  int64
 5   marital-status   48842 non-null  int32
 6   occupation       48842 non-null  int32
 7   relationship     48842 non-null  int32
 8   race             48842 non-null  int32
 9   gender           48842 non-null  int32
 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  int32
 14  income           48842 non-null  int32
dtypes: int32(9), int64(6)
memory usage: 3.9 MB
```

Figure: 4.1: Dataset after applying label encoder

Model Building

Logistic Regression

For this project, the first model we chose to build is Logistic Regression with all the features. The accuracy for this model was 83%. To further evaluate the accuracy, a confusion matrix was created. See figure 4.2. Although there was a high level of predicting incomes below 50K, the prediction on incomes above 50K was not enough as the false negatives seem to be higher. See figure 4.2.

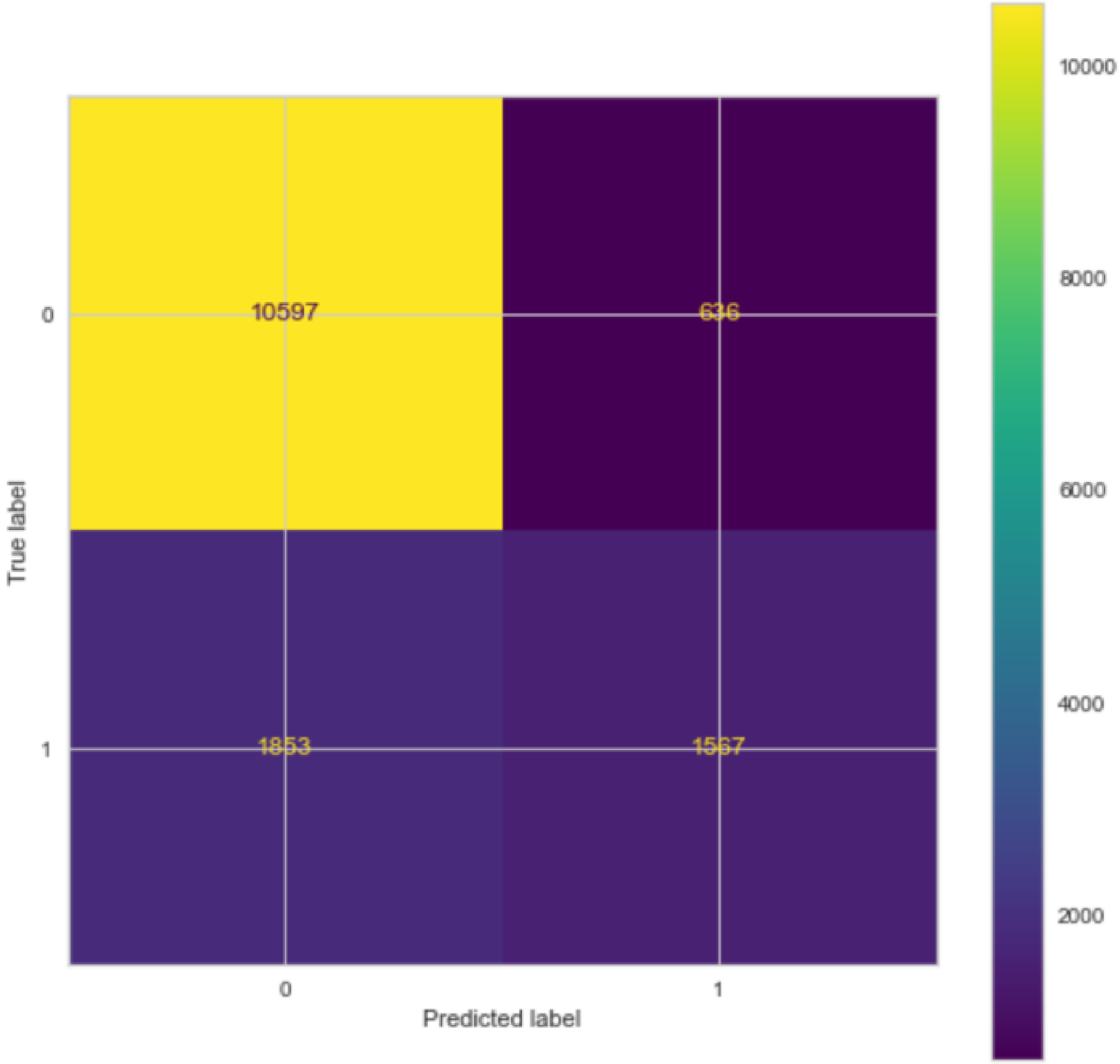


Figure: 4.2 Confusion matrix for Logistic Regression

Initially, logistic regression has been applied of the dataset with all 14 features and got AUC score is 0.74. Although the AUC score is in acceptable range of 0.7 to 0.8, we tried to apply StandardScalar on the dataset to resize the distribution of variables so that mean of the observed value is 0 and the standard deviation is 1. We received slight improvement in the score to 0.83 (83%). Then, we tried to remove irrelevant features like capital-gain, capital-loss and fnlwgt from the dataset and ran Logistic regression with and without StandardScalar. We received the score as .8074 (80.74%) and .8108 (81.08%) respectively.

We also noticed the F1 score for the minority class (earning income > 50K) gradually increases when we ran logistic regression after applying StandardScalar and removing irrelevant features from the dataset.

Decision tree

Next, we moved to the Decision tree model to see if this model improved the accuracy score of prediction. All the features were evaluated. The accuracy score for this model was came to 82% which is a slight reduction that the Logistic Regression model. From the confusion matrix for this model, see figure 4.3, we observe that the model did seem to improve the predictions for income >50K but degraded in predicting incomes < 50K.

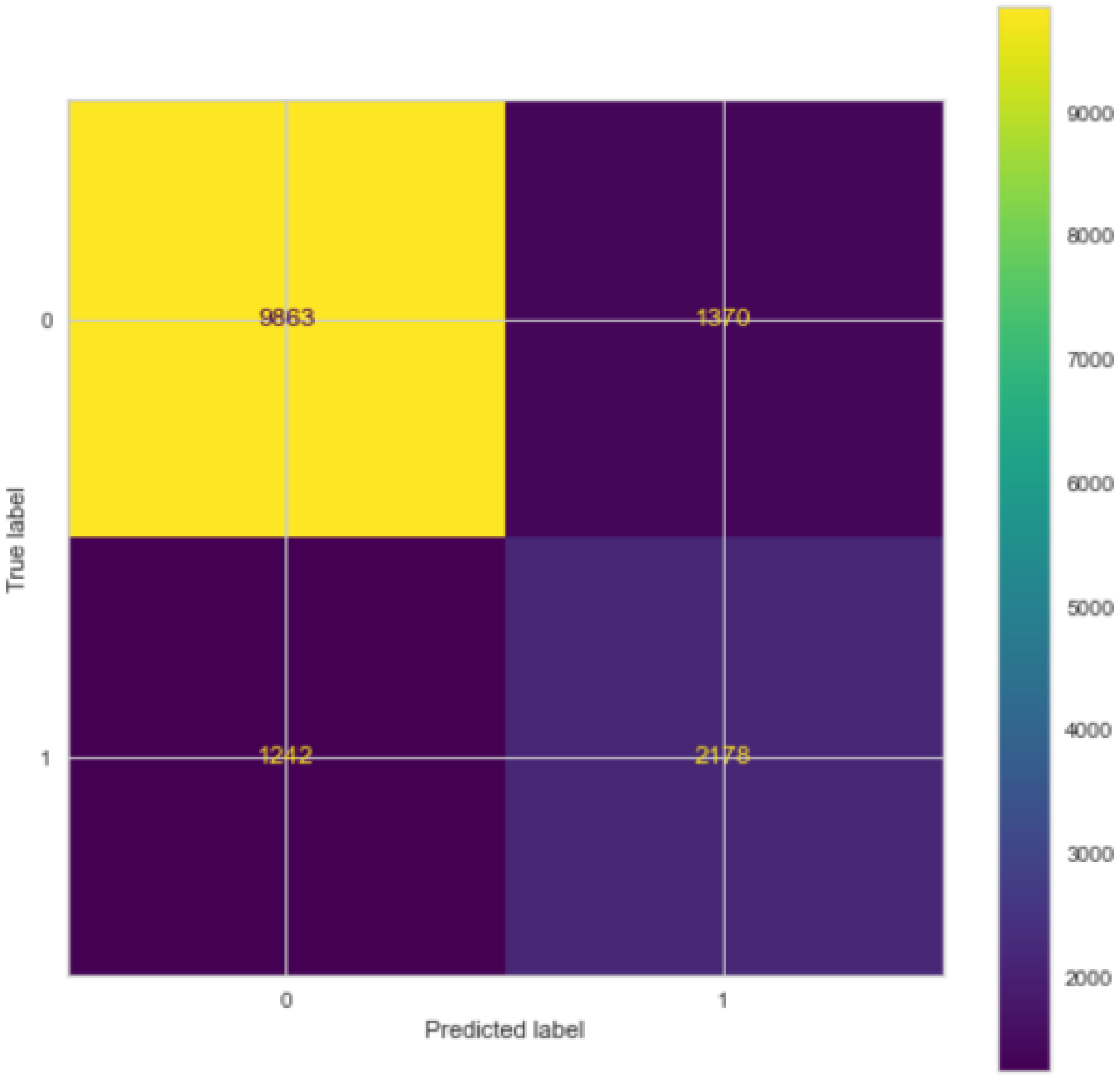


Figure: 4.3 Confusion matrix for Decision Tree

We followed similar approach for decision tree classifier as we did for logistic regression and received the accuracy as .8214 (82.14%) when we consider all the features from the dataset. This is slightly high compared to what we received for logistic regression. However, when we tried to run decision tree classifier on the dataset standardized using StandardScalar, the score didn't improve much. We also noticed that score got decreased when we apply decision tree classifier algorithm on the dataset after removing irrelevant features.

We noticed the F1 score for the minority class (earning income > 50K) gradually decreases when we ran decision tree classifier model after applying StandardScalar and removing irrelevant features from the dataset. Initially, the F1 score for minority class is 0.62 and decreased to 0.53 during the subsequent modeling.

Random Forest classifier

The third model is Random Forest classifier with all features. The accuracy score for this model is observed to be at about 87% which is the highest so far. The confusion matrix for the random forest classifier is represented in figure 4.4.

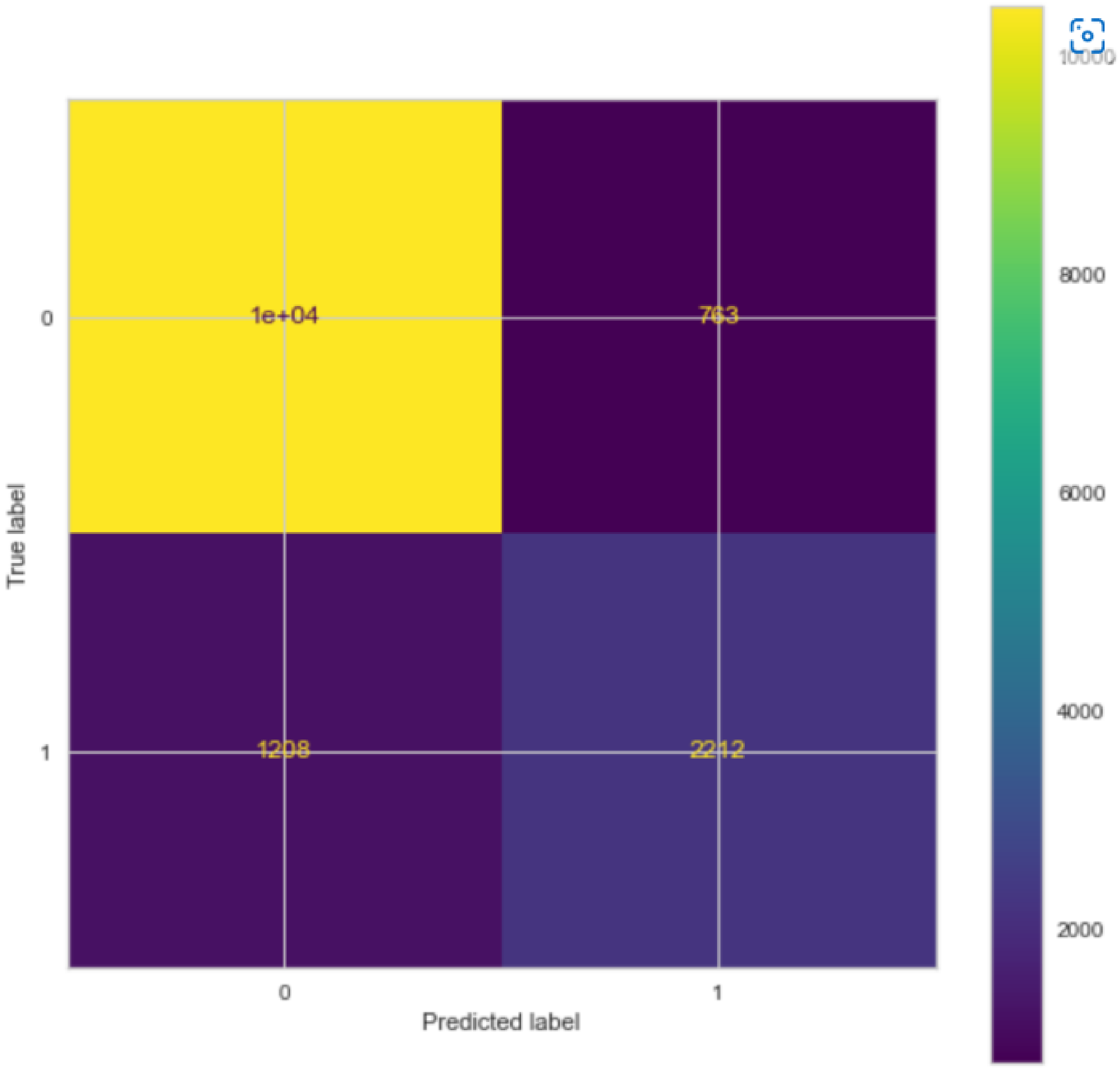


Figure: 4.4 Confusion matrix for Random Forest Classifier

We can observe from the confusion matrix that the predicted values for the target feature have greatly improved with 10470 predictions for income under 50K and 2212 correct predictions for income over 50K.

The same accuracy has been obtained after applying StandardScalar. Upon applying modeling on the dataset after removing irrelevant features, the score for random forest model has been reduced to 82.32%. The F1 score for the minority class (earning income > 50K) gradually decreases when we ran decision tree classifier model after applying StandardScalar and removing irrelevant features from the dataset. Initially, the F1 score for minority class is 0.69 and decreased to 0.53 during the subsequent modeling.

Features removed from the dataset during 2nd iteration:

- Capital-gain - Most of the rows are having value as 0
- Capital-loss - Most of the rows are having value as 0
- Fnlwgt - Not giving any meaningful information

Model Evaluation

The scores that we received for each model are represented in the table below.

Model	# Of Features	Accuracy	F1 Score (Income <=50K)	F1 Score (Income >50K)	AUC Score
Logistic Regression	All (14)	79.80%	0.88	0.40	0.74
Decision Tree	All (14)	82.14%	0.88	0.62	0.76
Random Forest	All (14)	86.55%	0.91	0.69	0.91
Logistic Regression with StandardScalar	All (14)	83.01%	0.89	0.56	
Decision Tree with StandardScalar	All (14)	82.17%	0.88	0.63	
Random Forest with StandardScalar	All (14)	86.54%	0.91	0.69	
Logistic Regression	After removing unwanted fields (11)	80.74%	0.88	0.49	0.81

Decision Tree	After removing unwanted fields (11)	78.44%	0.86	0.53	0.72
Random Forest	After removing unwanted fields (11)	82.32%	0.89	0.60	0.87
Logistic Regression with StandardScalar	After removing unwanted fields (11)	81.07%	0.88	0.50	
Decision Tree with StandardScalar	After removing unwanted fields (11)	78.44%	0.86	0.53	
Random Forest with StandardScalar	After removing unwanted fields (11)	82.18%	0.89	0.60	
Logistic Regression – Target variable with 5 best features using X2	With Top 5 Feature	79.99%	0.88	0.48	0.81
Decision Tree – Target variable with 5 best features using X2	With Top 5 Feature	79.56%	0.87	0.54	0.77
Random Forest – Target variable with 5 best features using X2	With Top 5 Feature	80.86%	0.88	0.57	0.85

Accuracy: Accuracy represents the number of correctly classified data instance over the total number of data instances.

F1 Score: F1-Score is a metric which considers both precision and recall.

- **Precision:** Positive predictive value
- **Recall:** true positive rate

AUC Score: What area under the ROC curve describes good discrimination? We will use the following rule of thumb

- 0.5: This suggests no discrimination, so we might as well flip coin
- 0.5-0.7: We consider this as poor discrimination, not much better than a coin toss
- 0.7-0.8: Acceptable discrimination
- 0.8-0.9: Excellent discrimination
- >0.9: Outstanding discrimination

Among all 3 models and multiple iterations, we noticed the AUC score is high for the Random Forest which is 0.91 when we run on the dataset with all 14 features without Standard Scalar. However, the score got reduced to 0.87 when we run on the dataset after removing unwanted features.

Feature Selection

Using SelectKBest, we tried to find the 5 best features after removing unwanted features from the dataset and following are the best features in the dataset which shows higher impact to the target variable "income" compared to other features present in the dataset.

- Age
- Education-num
- Marital-Status
- Relationship
- Hours-per-week

Upon running logistic regression, decision tree classifier and random forest classifier model on the dataset with top 5 best features, the score turned out as 79.99%, 79.5% and 80.86% respectively. The corresponding AUC scores are 0.81, 0.77 and 0.85 respectively.

Figure 4.5 represents a "Scores Plot", that shows the accuracy results of the different scenarios compiled in our analysis.

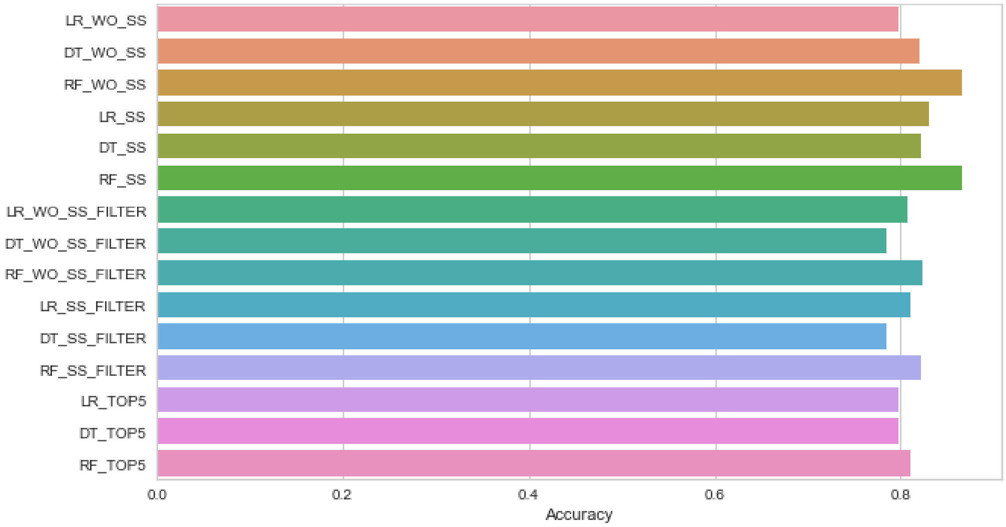


Figure: 4.5 Accuracy for different models applied on income dataset

Test	Features
Pearson's correlation matrix (figure 4.6) - Feature correlation to target variable "income"	**Numerical Variables:** Age Education-num Hours-per-week **Categorical Variables:** Gender Education
Chi-Squared (X2) Test - 5 Best features correlated to "Income"	Relationship education-num age marital-status hours-per-week
Using Feature Importance of Random Forest Classifier (figure 4.7)	Relationship education-num age marital-status hours-per-week

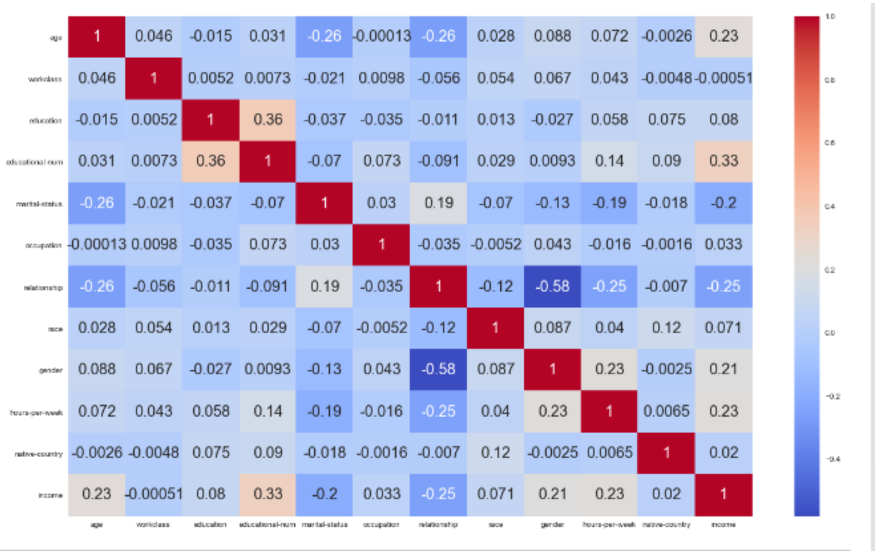


Figure 4.6: Pearson’s correlation matrix

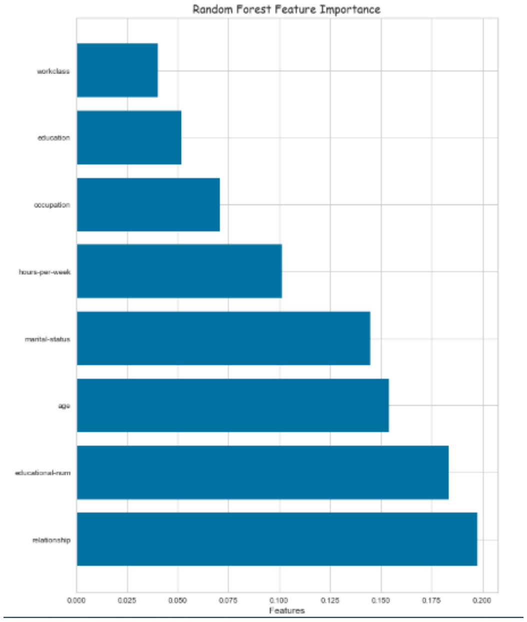


Figure 4.7: Feature importance method of Random Forest

Conclusion:

Out of three model, Random Forest Classifier is the best model to predict the income of the household as the score is higher compared to Logistic Regression and Decision Tree Classifier when we try to run the model for different scenarios.

Among various methods used to find 5 best features in the dataset, all the methods (Pearson's correlation, chi-squared and feature importance of RF) provided age, education-num and hours-per-week as best features which have high impact on the target variable "income" compared to other features in the dataset. Pearson's correlation gave Gender and Education as next best features, while chi-squared and feature importance of RF gave relationship and marital-status as best features having high impact.

From figure 4.7, the following features having high impact of the target variable "income"

1. Relationship: Relationship of the person to the family
2. Education-num: Number of years the person had education
3. Age: Age of the person
4. Marital-status: Marital status of the person
5. Hours-per-week: Hour the person worked for a week

The following are some of the recommendations to earn a higher income >50K.

1. You should work more hours per week to earn more income (Example: one who works 80 hours earn more compared to the one who works only 40 hours)
2. You should study for a greater number of years (number of education years) to earn high income.
3. The model also predicted that married persons earn more compared to unmarried persons. This may be due to the fact that a person with more experience (in term of years) is earning more compared to freshers. Mostly, the persons with more experience are married.
4. Model predicts that older people earn more compared to younger people. This implies that older people would have more experience who in turn earn more. So, you should get old to earn more income.

Reference:

Abbott, D. (2014). *Applied Predictive Analytics: Principles and Techniques for the Professional Data Analyst*. John Wiley & Sons, Inc.

https://www.census.gov/en.html

https://www.kaggle.com/datasets/uciml/adult-census-income

https://www.techtarget.com/searchbusinessanalytics/definition/logistic-regression

https://support.minitab.com/en-us/minitab-express/1/help-and-how-to/modeling-statistics/regression/how-to/binary-logistic-regression/before-you-start/data-considerations/