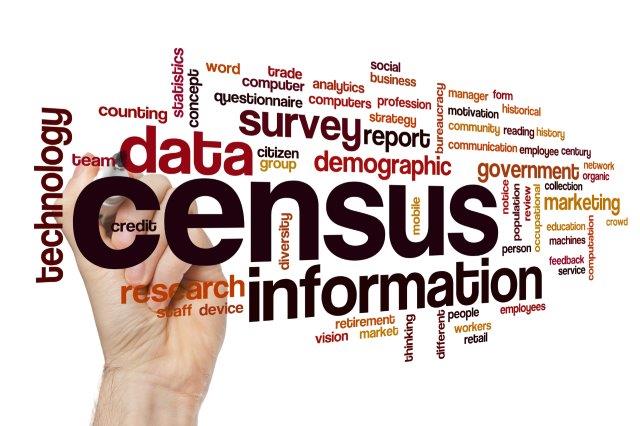
**Census Income Prediction with Machine Learning**

**ASHISH CHAND**

****

Machine learning is breaking grounds in numerous fields including Finance. What if we could use Machine Learning models to identify incomes of individuals? Here I have a dataset for this, called [Census Income Dataset](https://raw.githubusercontent.com/dsrscientist/dataset1/master/census_income.csv). I used the information in the dataset to predict if someone would earn an income greater than $50K/year.

**Census Money Income**

The Census Bureau collects income data on several major surveys, including the Annual Social and Economic Supplement (ASEC) of the Current Population Survey (CPS), the Survey of Income and Program Participation (SIPP), and the American Community Survey (ACS). The CPS is the source of the official national estimates of poverty and the most widely cited source of annual household income estimates for the United States.

**Description of Final weight**

The weights on the Current Population Survey (CPS) files are controlled to independent estimates of the civilian non-institutional population of the US. These are prepared monthly for us by Population Division here at the Census Bureau. We use 3 sets of controls. These are:

1. A single cell estimate of the population 16+ for each state.
2. Controls for Hispanic Origin by age and sex.
3. Controls by Race, age and sex.

We use all three sets of controls in our weighting program and "rake" through them 6 times so that by the end we come back to all the controls we used. The term estimate refers to population totals derived from CPS by creating "weighted tallies" of any specified socio-economic characteristics of the population. People with similar demographic characteristics should have similar weights. There is one important caveat to remember about this statement. That is that since the CPS sample is actually a collection of 51 state samples, each with its own probability of selection, the statement only applies within state.

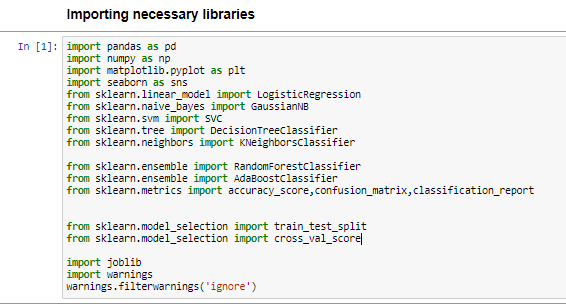
**Features/Columns**:

1. age - the age of an individual
2. workclass - a general term to represent the employment status of an individual
3. final weight - in other words, this is the number of people the census believes the entry represents
4. education - the highest level of education achieved by an individual
5. education\_num - the highest level of education achieved in numerical form.
6. marital\_status - marital status of an individual. Married civ spouse corresponds to a civilian spouse while Married AF spouse is a spouse in the Armed Forces
7. occupation - the general type of occupation of an individual
8. relationship - represents what this individual is relative to others
9. race - descriptions of an individual’s race
10. sex - the biological sex of the individual
11. capital\_gain - capital gains for an individual
12. capital\_loss - capital loss for an individual
13. hours\_per\_week - the hours an individual has reported to work per week continuous
14. income - whether or not an individual makes more than 50,000 dollars annually (the label)

**Problem description:**

**The prediction task is to determine whether a person makes over $50K a year.**

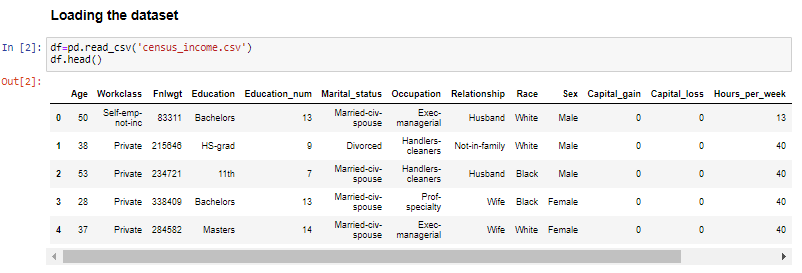
**Importing Library:**

****

I am importing the all library which I required for EDA, visualization, prediction and finding all matrices. The reason of doing this is that it become easier to use all the import statement at one go and we do not require to import the statement again at each point. We could find all the importing statement at one place without finding it on whole notebook and can update also.

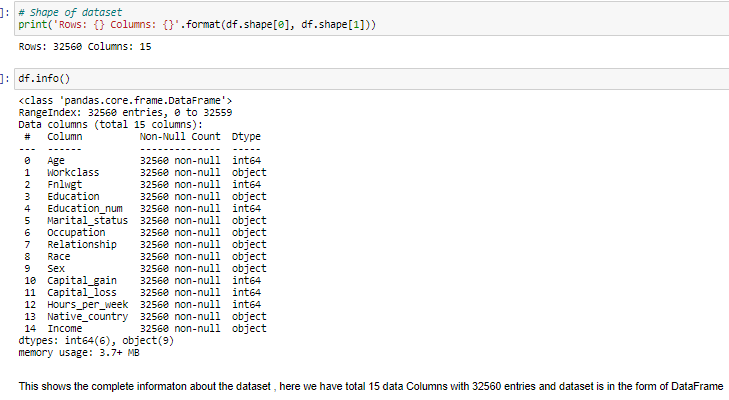
The main purpose of doing this importing is that we don’t need to import the library again and again. We can import the necessary libraries at the start and use them in the whole notebook.

**Load Data Set into variable:**

****

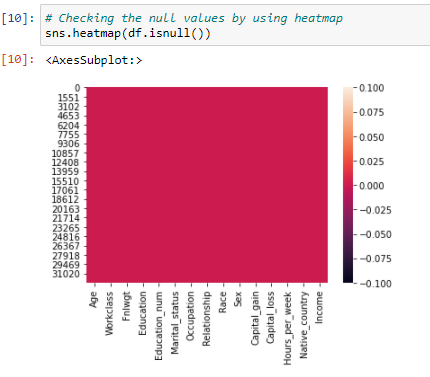
Here I am loading the data set into a variable i.e. “df” and processing the first 5 rows of the dataset. As in this data there are many object type data are present and some integer type data are present in the dataset.

**Exploratory Data Analysis:**

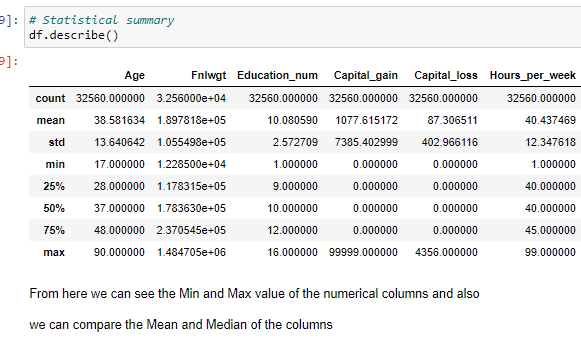
****

Here, I am checking the shape of the data set as there are 32560 records and 15 columns are present. Also, most of the columns are of same data type that is object and integer data type. Also the Income column is our Target variable or dependent variable while other columns are independent variables. Here I can see that the our dataframe is not null which I full confirm with help of heatmap by using the seaborn library .

**Checking the Null values:**



Above image shows the checking of null value with help of seaborn, it shows only color that means there was no null value present the dataset.



Above statistics data describe the column which are numerical and here we can see that

* By checking the difference between the 75% and max value there are outliers in some of the column, I will check it soon.
* And also the mean is greater than the median (50%) that means the data is skewed.
* Some columns statistics data are 0.

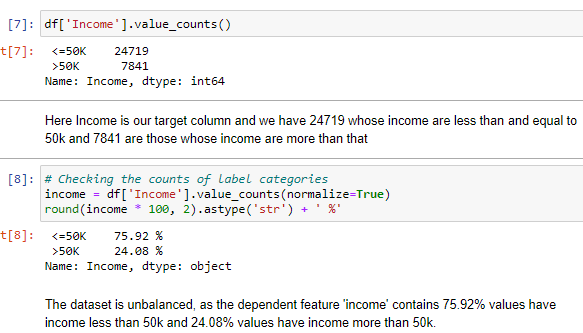
**Data visualization:**

In this portion we can plot different graph using different columns and try to visualize the data using matplotlib and seaborn library.

We use different graph include:

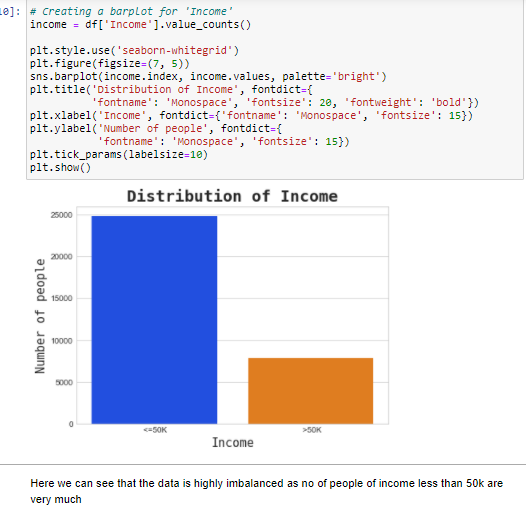
* Bar plot
* Count plot
* Line plot and distribution plot
* Histogram and Scatter plot
* Pie chart

**Distribution of Target variable:**

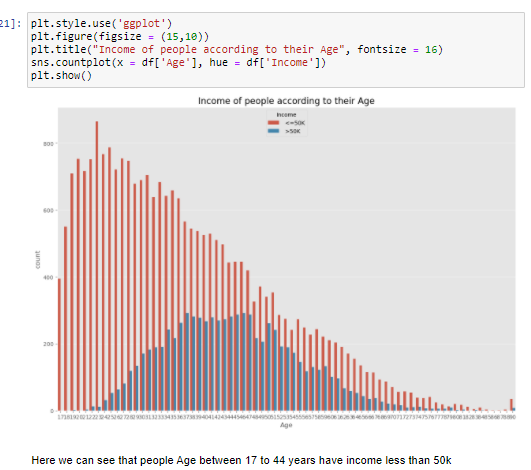


In above image I check our Target column where first count the values and found that the our target column have two type of values i.e. some people have income more than $50K and some have income less than and equal to $50K. And here I find that our data is imbalanced as 75.92 % people have income <= $50K and rest 24.08 % people have the income greater than $50K. This is the class imbalance problem here first I balance the dataset with sklearn library and also the criterion find the Auc Roc score and ROC curve of the model.

Checking the Distribution of Income :



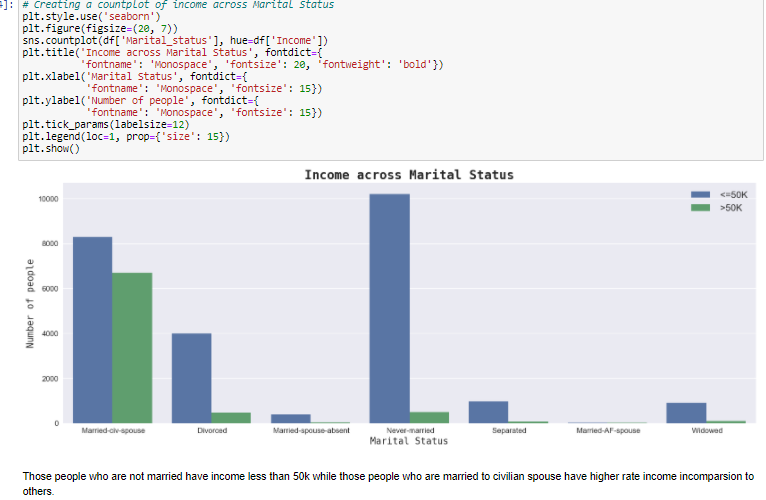
To check distribution of data more efficiently and in understandable form I plot the barplot of the Income column. And here it is very clear that it is class imbalanced problem. Here the people who have the income less and equal to the $ 50K per year are very high that means there are very middle class people are living in the United states where as above than $50K are very less in comparison to other half of United States people.





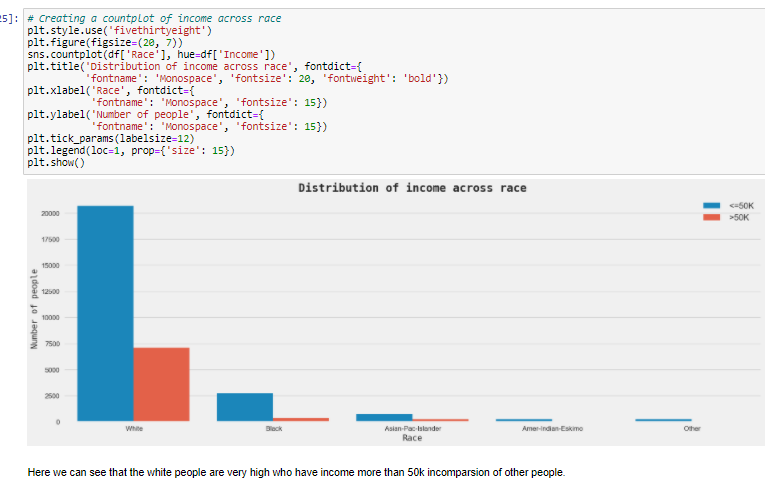
From above we came to know that:

* People with age group from 17 to 44 years have less income i.e most of the people of this age group have the income less than $50K and it is very high from 17 to 40 years.
* People with age after 30’s have started earning good income and that depends on the education and the experience which they had.
* In the next graph when we see the impact of education we came to that the High school grad people are the most who have less income, i.e. the more your education the more will be the income that’s why Bachelor degree people have high income.
* While one thing is to be noticed that people after high school grad. started earning and some college passed students too.
* Doctorate degree people have high income ratio.



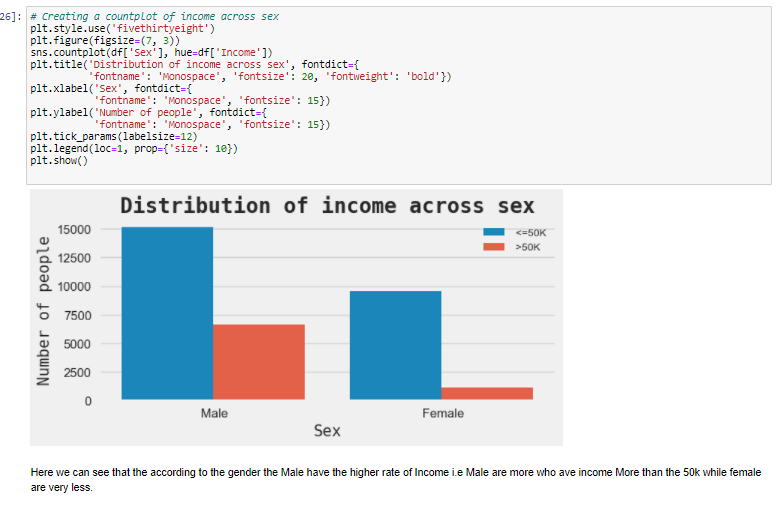
From above graph:

* It is clear that the ratio of people who civilian status married have high income
* While the most of the people who are single or never married have income less than to $50K.



From above graph we came to know that:

* In US most of the people are white and that’s why the ratio of income of people shows that people with income less than to $50k are very in comparison to the people who have more income. i.e out of 5 people 1 person have income greater than $50K
* Also there are very less black people and Asian people
* The ratio of income of black and Asian people are less than $50k.



From above countplot we can see the Distribution of income across sex/gender:

* The males who have high income greater than $50k are very while the females who have income greater than the $50K are very less.
* The ratio of male who have income less than $50K are in good numbers.
* The ratio of income of males are out of 10 people, 3 people have the income greater than the $50k while other have not.
* In female the ratio of high income is very less i.e most of the female have income less than $50K.

**Checking the correlation:**

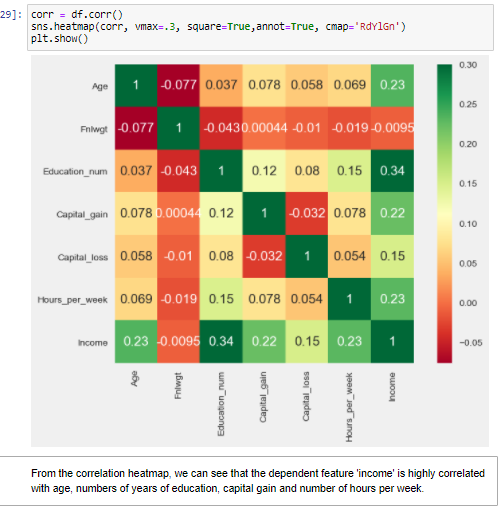
A **correlation matrix** is simply a table which displays the **correlation**. The measure is best used in variables that demonstrate a linear relationship between each other. Denoted by r, it takes values between -1 and +1.

Now I am finding the correlation value of each column, this value is categorized into mainly 2 parts that are:

- Positive correlated value

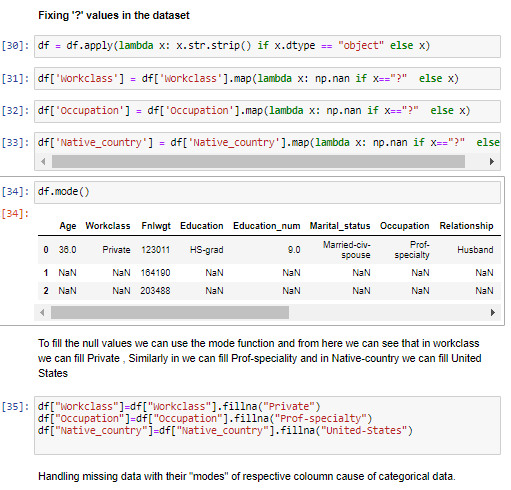
- Negative correlated value

The most the value is positive means that column is much co related and vice versa.

****

Here we can see that the dependent feature ’income’ is highly correlated with age and number of years of education.

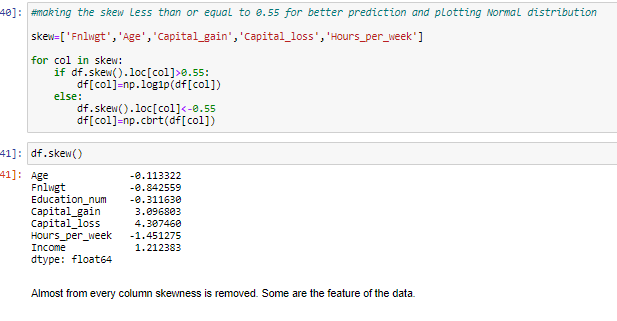
**Data Preprocessing :**



In above data set there are some missing values “?” are present. At first we change the missing values to NAN value with help of lambda function and mapping to the original dataset.

Generally the numerical column missing value are filled with mean and median of the particular column. While in categorical column we use the mode function.

Here our missing values are from the categorical columns and then I find the mode of that columns. And after that we fill the column “ Workclass” with “Private” and Occupation with “Prof-speciality” and Native Country with “United-States”.



In above image we are first calculating the skew value and some of the column skew value are far from zero so :

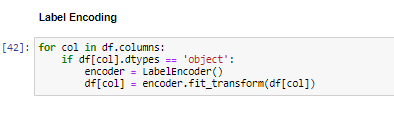
* The best skew value for normally distributes is very close to zero, so we are using “log1p” and cuberoot method to make the skew value near to zero.
* But in some cases by applying all skewness reduced method but still not skewness is reduced that means it is the nature of that data
* After performing “log1p” and cbrt method the columns have skewness value near to zero.

Note: Making the skewness value near to zero will help to get better score.

**Label Encoding:**

Sklearn provides a very efficient tool for **encoding** the levels of categorical features into numeric values. **Label Encoder encode labels** with a value between 0 and n\_classes-1 where n is the number of distinct **labels**. If a **label** repeats it assigns the same value to as assigned earlier.

Converting all categorical column into numeric value by using encoder.



In above image we select those columns whose datatype is object i.e categorical column. Here we use the for loop and perform the label encoding.

**Outliers:**

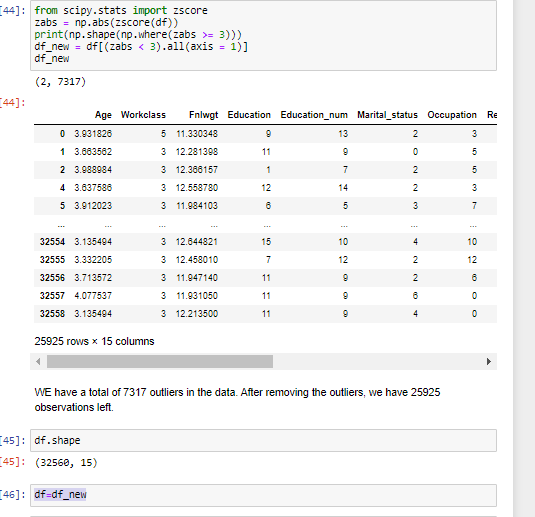
An **outlier** is a data point in a data set that is distant from all other observations. A data point that lies outside the overall distribution of the data set.

We check if any outliers are present in the continous attributes of the dataset. We check it both by visualisations and the zscore for the continous columns.



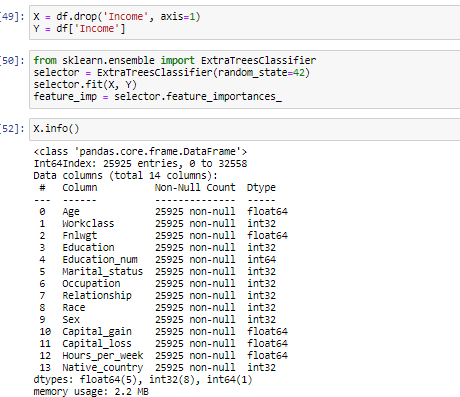
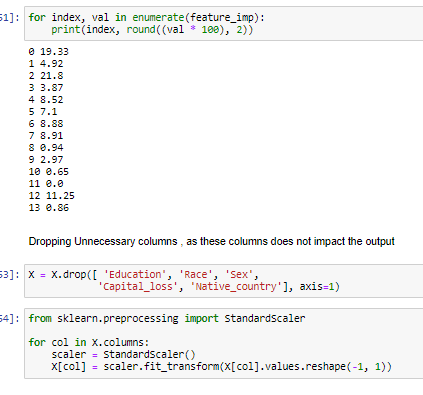
From above image we can clear see that there are number of black dots in most of the column which are referring to the outliers, so it means most of the data are outside the distribution. So now we detect the outliers now the second step is to remove the outliers, there are different way to remove the outliers that are find the IQR, zscore values. Here I use the z score method.

**Removing Outliers:**

****

So, I first find the zscore value and then I decide to make one threshold value as 3 which is standard of industry recommend value and then I remove all the outliers which zscore value is greater than 3. After, removing the outlier’s final there are 32560 and 15 column presents in the data set. After that we store the new variable value to the our already existed Variable i.e. “df”

**Feature Selection:**

** **

After preprocessing of the data , we done feature selection of the data. Here:

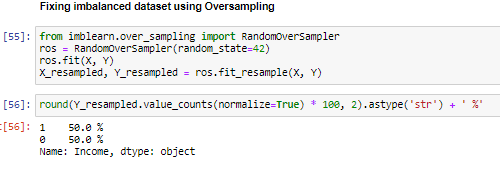
* We first the divide our data into two variable where in X only independent columns are present , no income is present
* In Y variable Only “Income” column is present
* After this we use the ExtraTreesClassifier method from the sklearn ensemble library where we train the variable X and Y , then select only those column of X variable which are important for prediction of Y
* Then we use a for loop where we check the values which according to their importance.
* After this we remove the unnecessary columns i.e remove those columns which have very less impact on the output.
* And at last we perform the Standard scaler method to balance the values of columns in a particular range.
* And then store the whole values and columns to our X variable.

**Imbalanced-learn (imported as imblearn ) :**

Undersampling is the way where we generate synthetic data so for the minority class to match the ratio with the majority class whereas in oversampling we reduce the majority class data points to match it to the minority class.

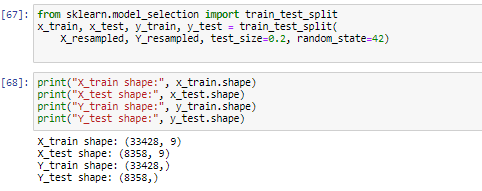
Here I use the Over sampling and then import the RandomOverSampler.

Random oversampling can be implemented using the **RandomOverSampler** class. This means that if the majority class had 1,000 examples and the minority class had 100, this strategy would oversampling the minority class so that it has 1,000 examples.



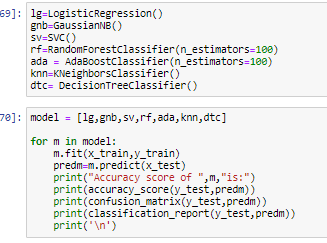
As we mention in the starting when we checking the distribution of our target variable that this is a class imbalance dataset. So to balance the imbalanced dataset and from the imblearn over sampling library we import RandomOversampling to balance the dataset in equal parts.

**Prediction with Income:**

****

In above image we use the scikit-learn lmodel selection library and from that I import the Train test split method and divided the balanced X and Y variable into the Test and training data.

After performing this method, we get X train and Y train variables and X test and Y test variables. This variables were used to perform to train the models.

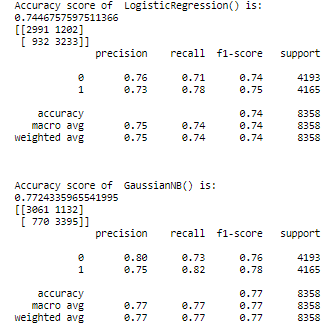
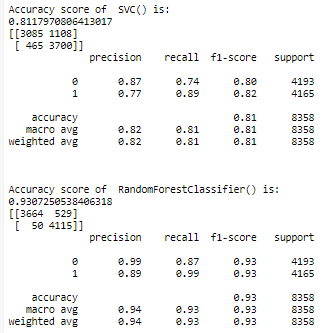


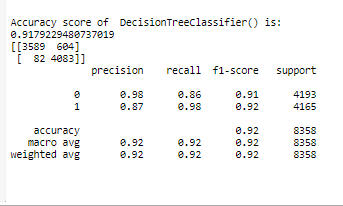
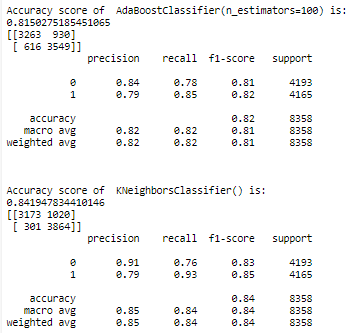
Here I store the different model into the variable and then use the for loop for training the data and perform the evaluation of matrices.

If we perform one by one for every model for training and testing the dataset than it takes very long time, that’s why we use loop here and make this a single function. The loop will run again and again for every model and train the every model and predict from the test data. And after that it perform matrices evaluation.

After performing this we get the Best model which have the highest accuracy score and that model will help for the further process of prediction.

**Let’s check the result:**



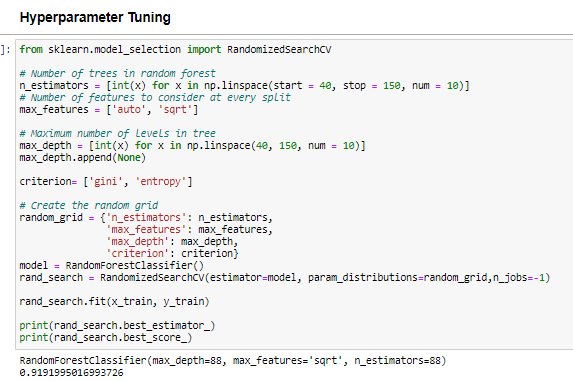
From above we get :

* the accuracy score of different models
* precision , recall value of every model
* The f1-score value of all models

**Hyperparameter Tuning :**

Hyperparameter tuning is choosing a set of optimal hyperparameters for a learning algorithm. A hyperparameter is a model argument whose value is set before the learning process begins. The key to machine learning algorithms is hyperparameter tuning.models can have many hyperparameters and finding the best combination of parameters can be treated as a search problem. Two best startegies for hyperparameter tuning are:

* GridSearchCV
* RandomizedSearchCV.



As above we have our best model and to make it more accurate and good for the final training we done the Hyperparameter tuning and here I used the method of RandomizedSearchCV and after this we find the best parameter which give more accurate prediction. After this I trained the model for the one final time with the help of hyper tuned parameters.



**In prediction:**

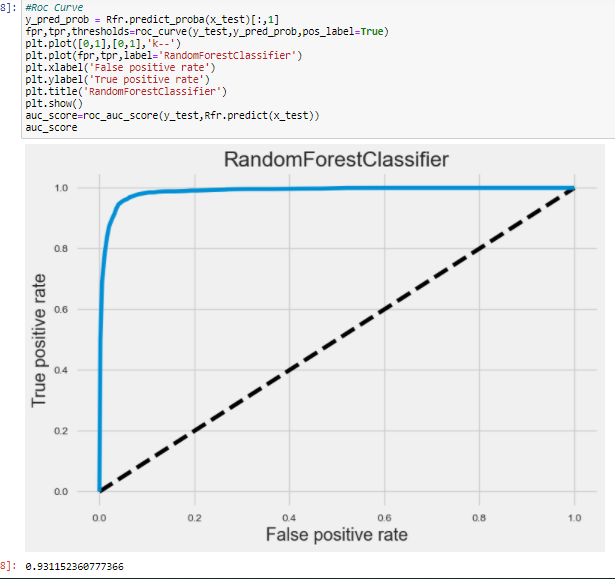
* I had done this prediction by taking Average price as an output variable which is continuity in nature so that why I’m using the regression technique
* Here the random state that is occurring is 42 which provide the best accuracy score for the model which is 93%.
* After using the RandomizedSeachCV, I can find the best parameters and then I used these parameters for that model.
* After using the best parameters I am able to find the best param and then find the best accuracy score that is 93.5% and the model is RandomForestClassifier.
* After the cross validation the score is 93.58 .
* Standard deviation after cross vcalidation is 0.
* There are following matrices which I find, and which are providing the best score.

**Roc Curve**:

A useful tool when predicting the probability of a binary outcome is the Receiver Operating Characteristic curve, or ROC curve.

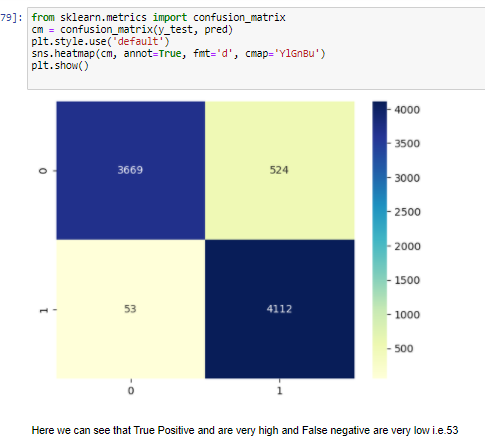
It is a plot of the false positive rate (x-axis) versus the true positive rate (y-axis) for a number of different candidate threshold values between 0.0 and 1.0. Put another way, it plots the false alarm rate versus the hit rate.

The true positive rate is calculated as the number of true positives divided by the sum of the number of true positives and the number of false negatives. It describes how good the model is at predicting the positive class when the actual outcome is positive.

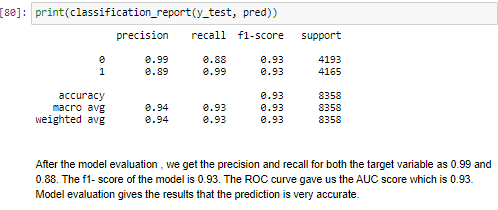


The Aucroc score of the RandomForestClassifier is 0.93.

Here we check the matrices of the selected model and check the observation:



Here true positive rate is high that means there are very less false outputs.



**Observation:**

* Taking income as y variable is predicting well.
* Also, I used the Label Encoder to make the categorical data into numeric data.
* The accuracy score of the selected model is 93%
* The precision and recall value of the model is 0.99 and 0.88 respectively
* The F1-score of the model is 0.93.
* There are no outliers in the data set after replacing it through mean value
* The 17 to 32 years age group have income less than to $50k while people after their 30’s earn more and their income increases i.e they have income more than $50k.
* People whose education is Bachelors have income more than the $50K while the high school grad people have income less than the $50K
* Males have the higher income than the females.
* The people who have Years of education 9 and 13 years have income very high.
* Married people have the income greater than the $50K, while most of the never married people have the income less than $50K
* I had done prediction using the Income and the prediction score is high.
* In this data set I used classification technique and also the Roc curve is used and after using roc the auc roc score we get is 93%, which is very good.