**Census Income Project.**

*“This is a tad bit lengthy article since I had to make sure I do not miss out on any of the important points to build a complete ML project but you have definitely made the right choice to invest your valuable time in something informative here.”*

Today I am going to write about a complete end-to-end project for Census income data which should serve as a guiding path for many Data Science aspirants. I agree that there are already many projects available into some deep web beneath multiple clicks and paths hidden like a core of an onion protected by multiple layers. But here I am just trying to do my bit in making things easier for a new comer to understand the basic architecture that’s required in the real world for creating a Data Science project.  
  
So, without any further ado please allow me to explain the agenda for this blog post. In this article, I have jotted down all the techniques in the form of sub-topics that I will be explaining one by one. And those pointers are as follows:

1. Problem Definition.
2. Data Analysis.
3. EDA and Pre-processing Data.
4. Data Pre-processing pipeline.
5. Building Machine Learning Models.
6. Concluding Remarks.
7. **Problem Definition**

This data was extracted from the 1994 Census bureau database by Ronny Kohavi and Barry Becker (Data Mining and Visualization, Silicon Graphics). To build a model that will predict if the income of any individual in the US is greater than or less than **$50K a year** based on the data available about that individual. You can find the dataset [here](https://raw.githubusercontent.com/dsrscientist/dataset1/master/census_income.csv).

This data set will help you understand how the income of a person varies depending on various factors such as the education background, occupation, marital status, age, number of working hours/weeks, etc.

Here’s a list of the independent or predictor variables used to predict whether an individual earns more than $50K or not:

* Age
* Work-class
* Final-weight
* Education
* Education-num (Number of years of education)
* Marital-status
* Occupation
* Relationship
* Race
* Sex
* Capital-gain
* Capital-loss
* Hours-per-week
* Native-country

The dependent variable is the “**Income**” that represents the level of income. This is a categorical variable and thus it can only take two values:

1. <=50k
2. >=50k

### Description of fnlwgt (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:

* A single cell estimates of the population 16+ for each state.
* Controls for Hispanic Origin by age and sex.
* Controls by Race, age and sex.

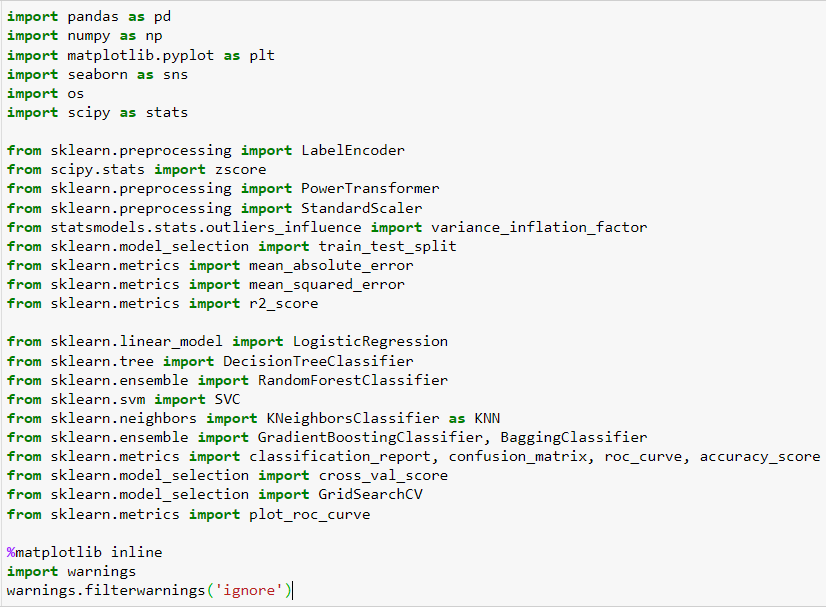
We use all three sets of controls in our weighting program and "rake" through them 6 times so that by the end we come back to all the controls we used. The term estimate refers to population totals derived from CPS by creating "weighted tallies" of any specified socio-economic characteristics of the 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.

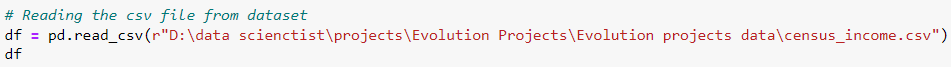
Now that we’ve defined our objective and collected the data, it is time to start with the analysis.

Detailed analysis and model building carried out with Python Jupiter notebook using some important libraries.

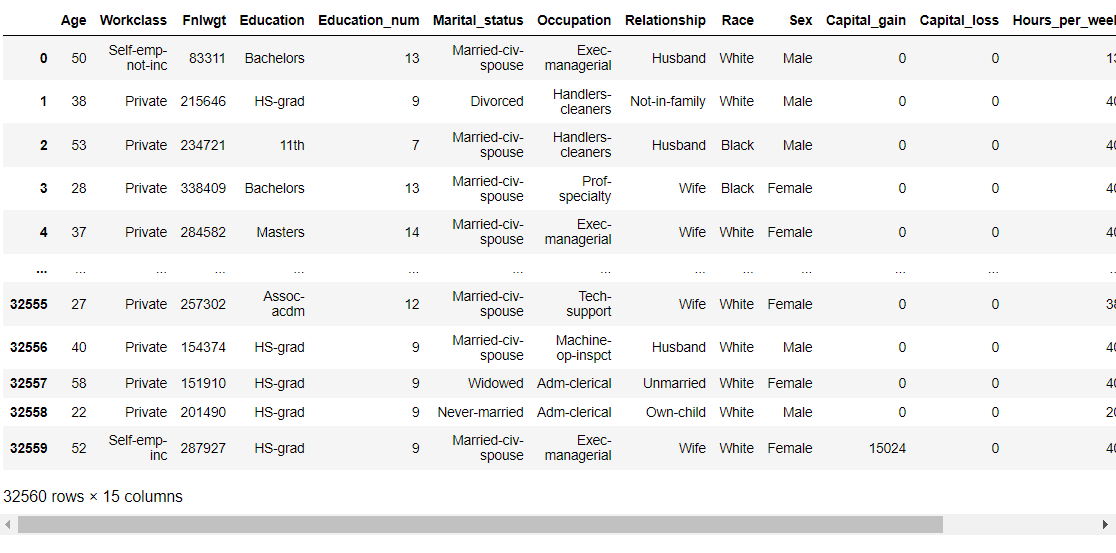
Step 1: Import the data.

First, we are going to import all the necessary dependencies here that will be used in our project and obtain the rest as and when required. Before we begin with any process, we need to get the dataset in our Jupyter Notebook that can be achieved by a single step.

**Importing Libraries:**

Next step is to import dataset

This gives us our entire dataset stored in the variable name “df” for our data frame.

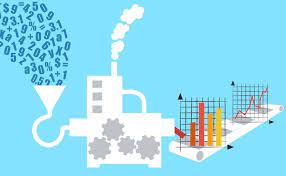
1. **Data Analysis:**

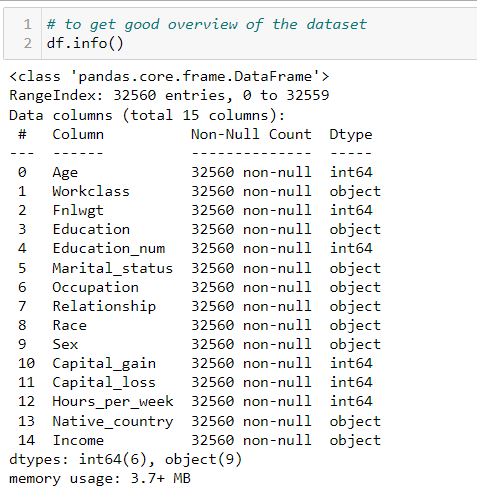
So here we can see, the dataset contains the details of Census bureau database (1994).

* This dataset is comprised of 32560 rows and 15 columns including one target variable i.e., 'Income'. where we need to predict whether a person makes over $50K a year or not.
* we can see dataset is having features with different data types.
* The target variable that is 'Income' has two classes of data, hence this is a classification data.

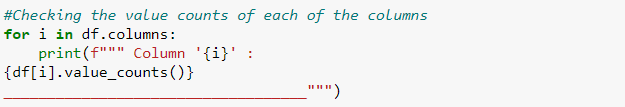
1. **EDA (Exploratory Data Analysis) & Pre-Processing Data:**

Exploratory data analysis (EDA) is used by data scientists to analyse and investigate data sets and summarize their main characteristics, often employing data visualization methods. It helps determine how best to manipulate data sources to get the answers you need, making it easier for data scientists to discover patterns, spot anomalies, test a hypothesis, or check assumptions.

****

****

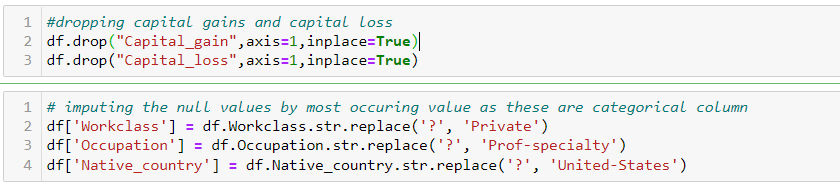
So here we can see the dataset contains different features with different data types.

* There are two types of data namely object type (9 columns) and int64 data type (6 columns).
* ****We will take care of the object datatype using encoding techniques later.

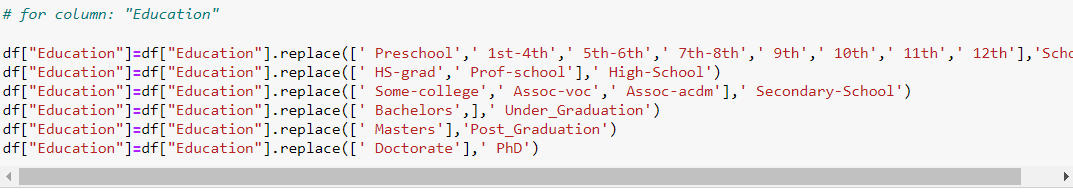
These three codes give us the missing values as well as unique values information.

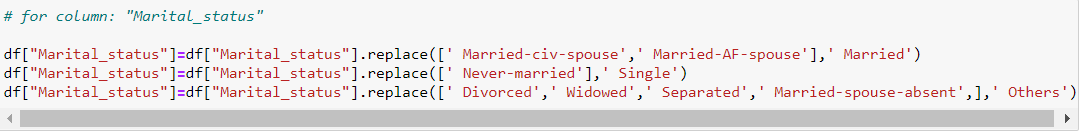
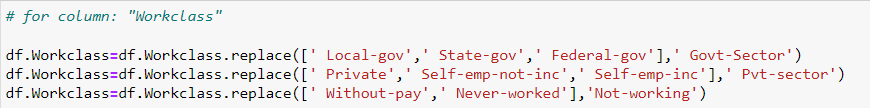
From this analysis we see the following points:

* The data set is not having any null values. But I came to know that some of the columns having entries marked as **“?”.** (Column: Workclass, Occupation and Native\_country)
* We have 29849, zero value in capital gain and 31041, zero values in capital loss, which is 90% of the data, so we can consider dropping them.
* we can see that there are multiple related classes in columns like "Workclass", "Education" and "Marital\_status". We will group them up and create new class for better analyzing.



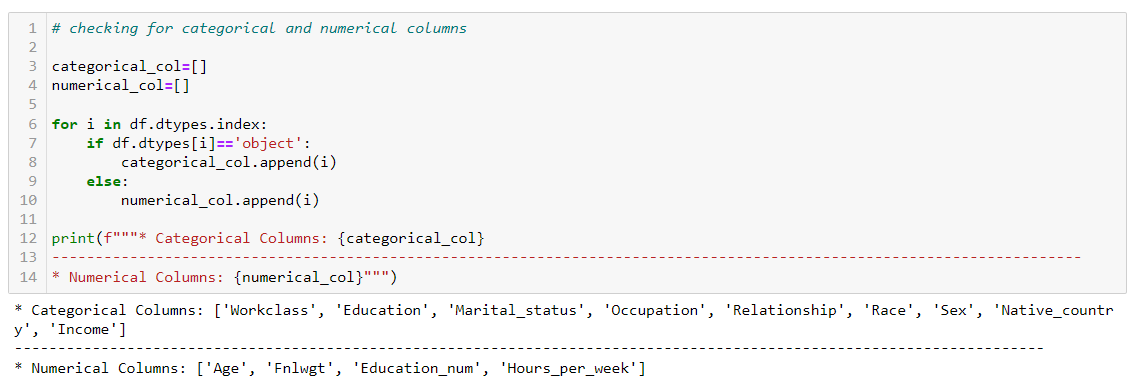
By these two codes I have deleted the column ‘Capital\_gain’ as well as ‘Capital\_loss’ and imputed the columns which were contain the “?” mark.

**Grouping related classes to one class and create new classes for easy analysis and visualization.**

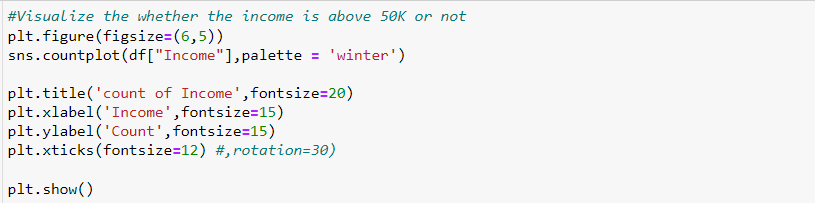
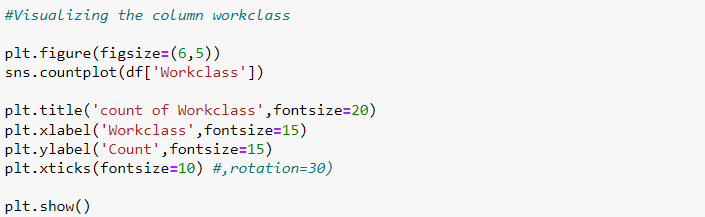
**** 

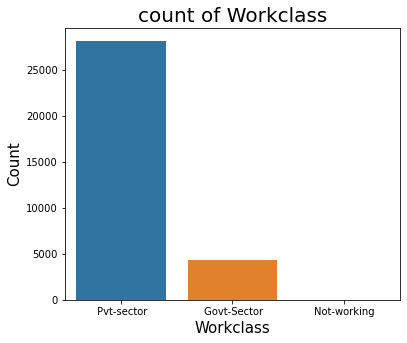
By this I have replaced multiple related-classes into a single new class for three columns (Workclass, Occupation and Native\_country).

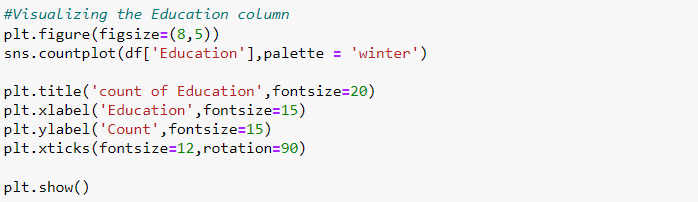
After this I have separate the numerical and categorical column for proper data visualization using below code.

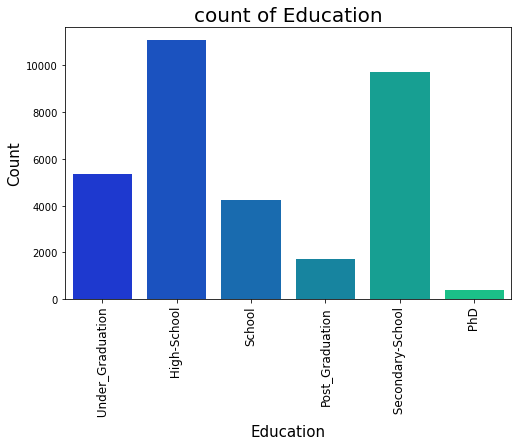


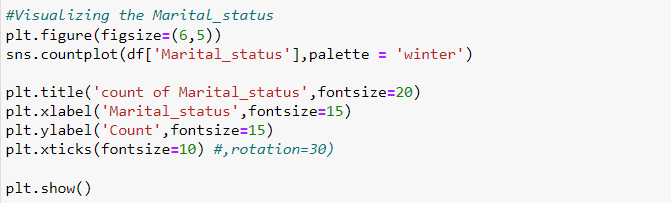
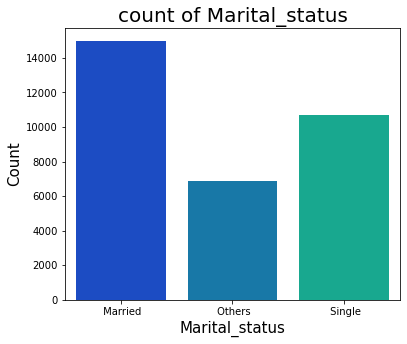
**Data visualization:**

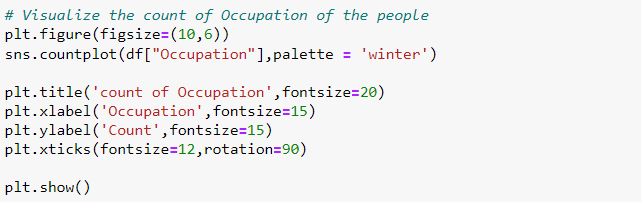
* **Univariate Analysis: For categorical columns**
* **Income**
* We see that there are 2 classes in target variable, and the dataset is imbalanced. So, We have to balance the dataset for modelling.
* **Workclass**

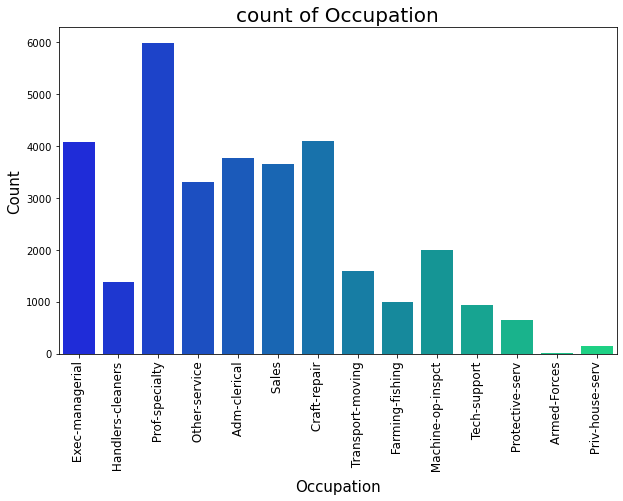
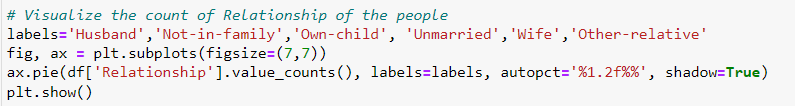
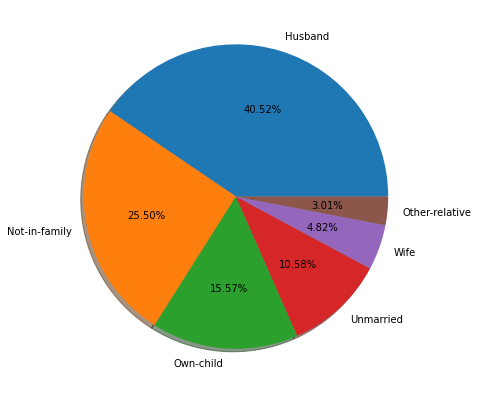
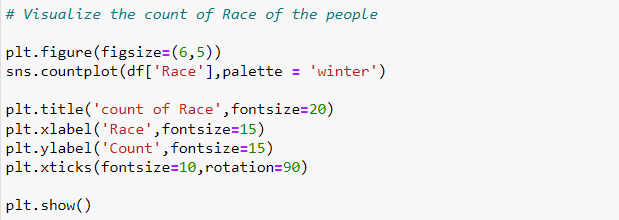


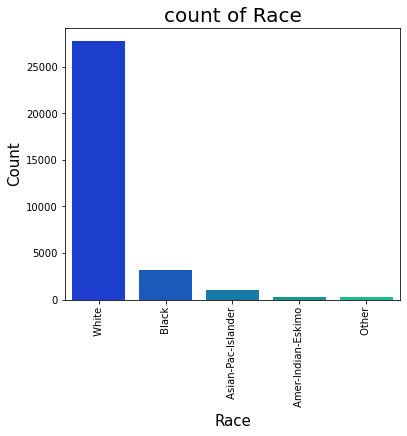
* Here we can see that, the population is max working in the private sector, followed by govt sector and none unemployed.
* **Education**



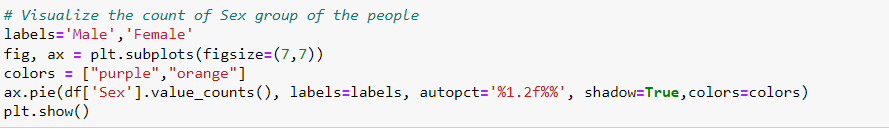
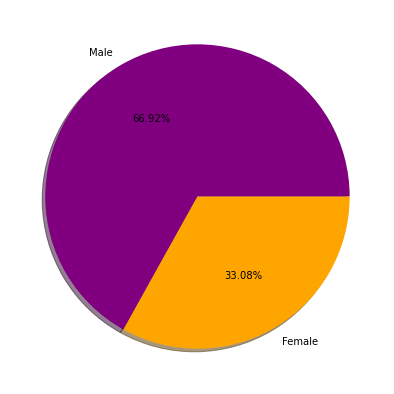
* Here we can see that the max population has the education level at high school, followed by secondary school.
* there are very few people who has done PhD and post\_graduation.
* **Marital\_status:**
* Here we can see that the population of married people are higher.
* **Occupation of the people:**



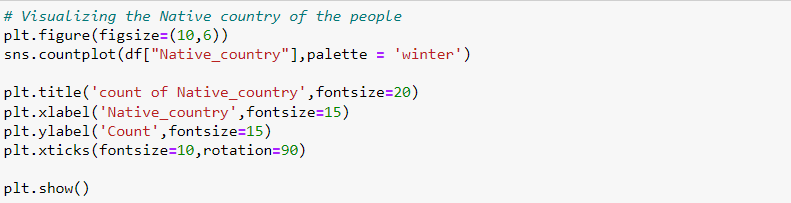
* The people who are in the position of Prof-speciality have highest count followed by craft-repair, Exec-managerial.
* And the people in the position Armed-Forces and priv-house-serv have very least counts.
* **Relationship of the people:**
* The count is high in the Husband category which has around 40% of count and other relative has very least count around 3%.
* **Race of people:**



* White family groups have high count which id more than 80% and Other race have least count.
* **Sex-Group of people:**

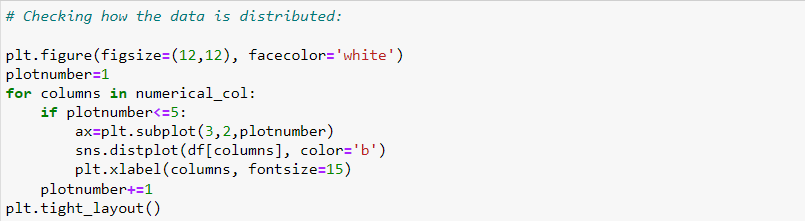


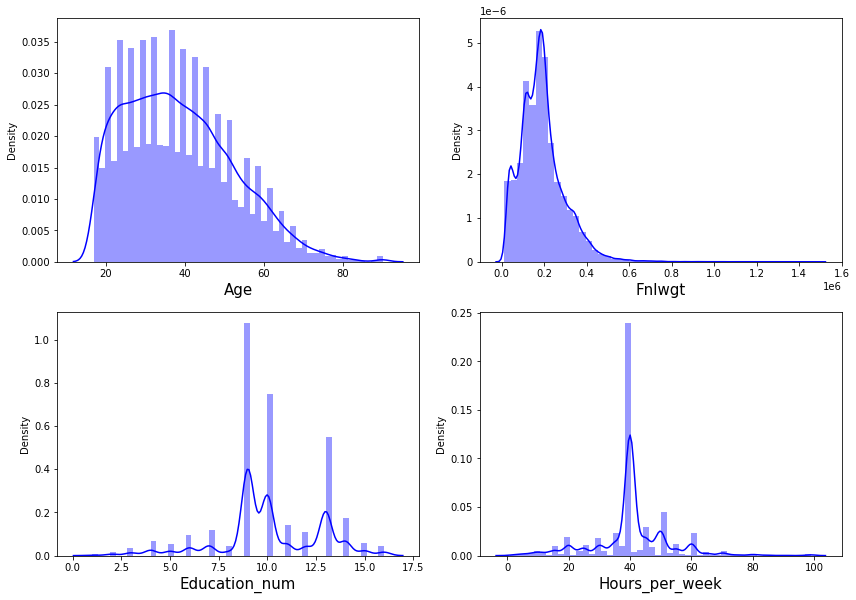
* The count of Male is high and has around 66% and only 33% of females are there.
* **Country of the people:**



* The United States country has highest count of around 29K and other countries have very less counts.
* **Univariate Analysis: For Numerical columns**

By this code we will visualize the distribution and skewness in the data of all the numerical columns.

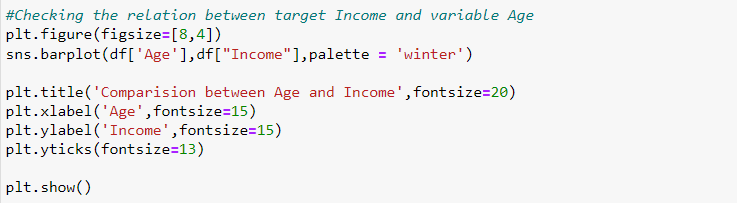




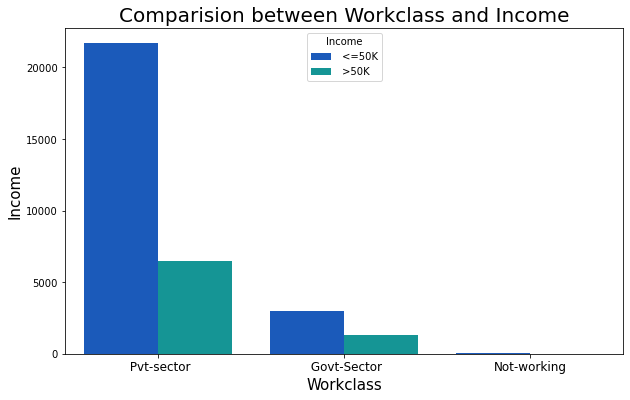
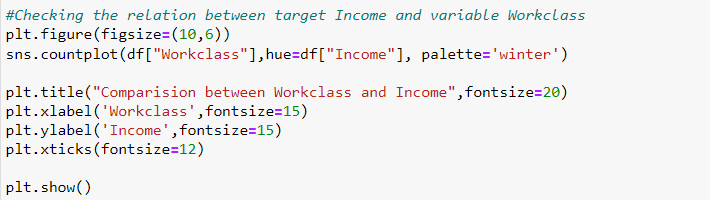
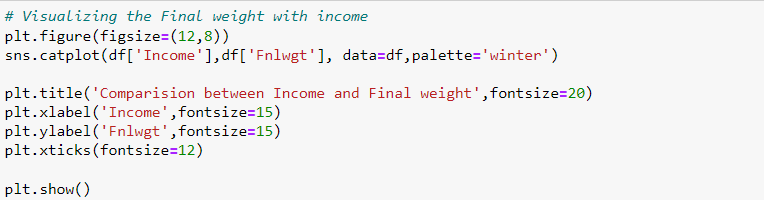
* We can clearly identify that the data in the dataset is not normal. We have 'Age' which is skewed right, Final weight is also skewed right, education is scattered and hours is also scattered with no skewness.

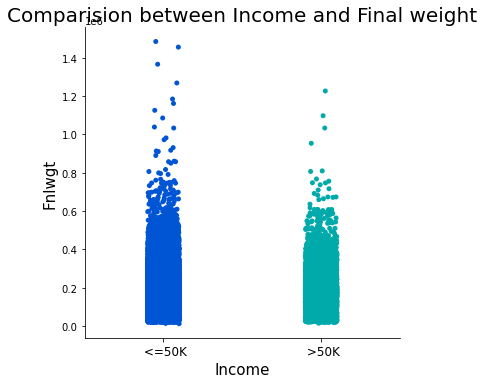
### **Bivariate Analysis (Checking relation between features and label):**

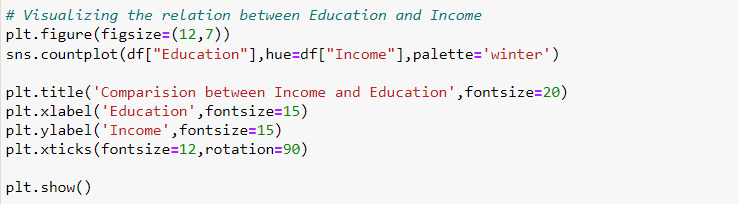
* **Relation between target Income and variable Age**

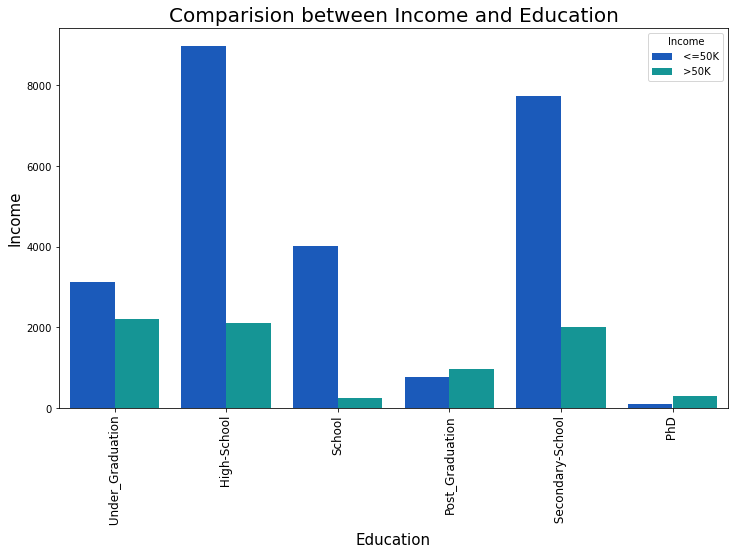


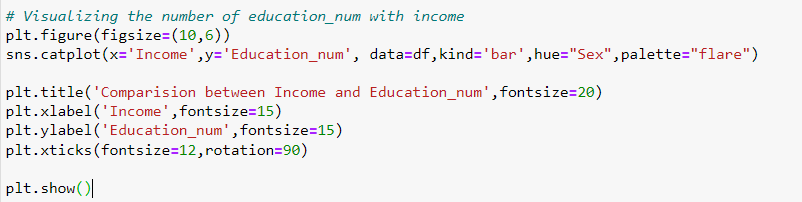
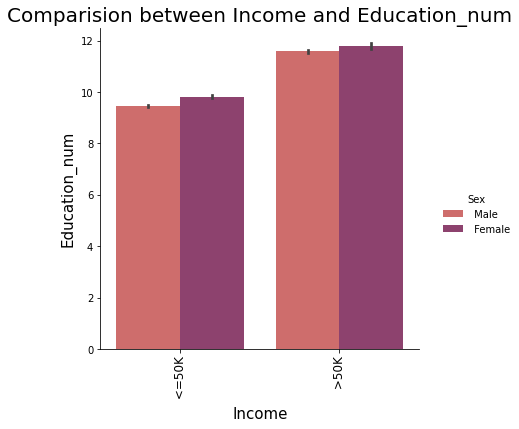
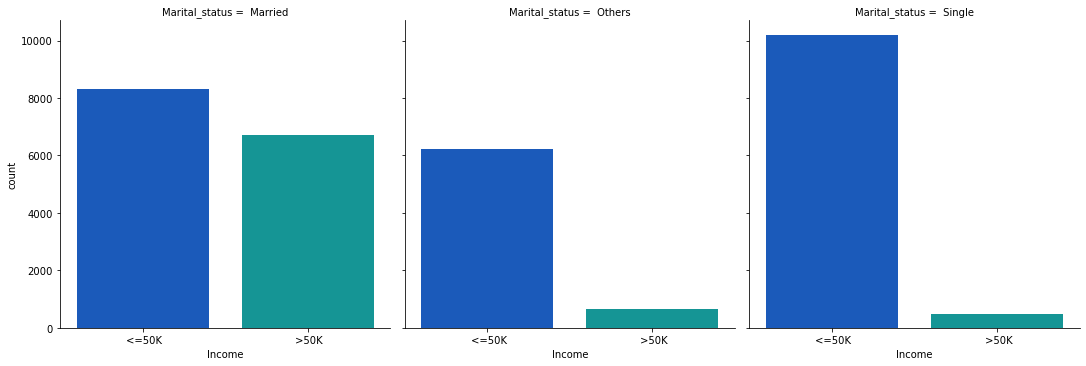
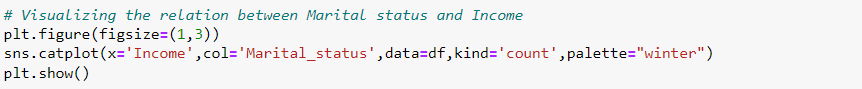


* The older age group has income more than 50K.
* **Relation between target Income and variable Workclass:**
* The people who are working in the private sectors have the income <=50K and the only few of the people in the same sector have income >50K. Also, the people who never worked they don't have the income.
* **Comparison between Income and Final weight:**



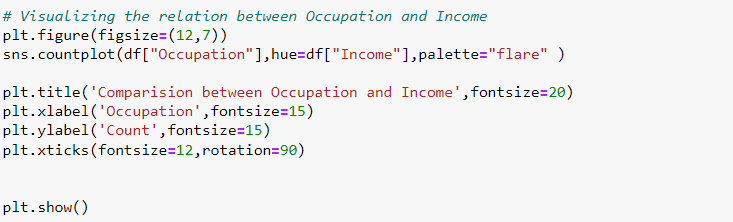
* There is no significant relation between final weight and income of the people.
* **Relation between Education and Income**

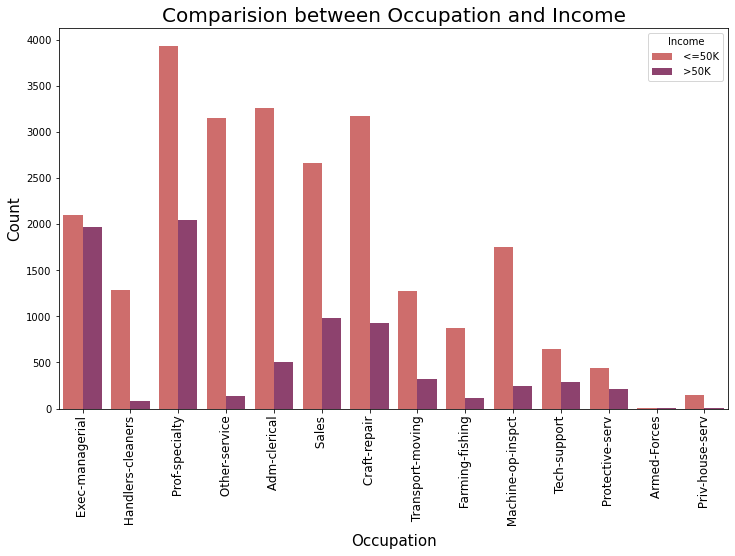


* The people who completed their high school have income <=50K followed by the people who done their Secondary School. Also, the people who done their Graduation they are earning more income that is >50K.
* **Visualizing the number of education\_num with income**
* The Income is more than 50K for the people having high education number. Here both genders have the income more than 50K.
* **Relation between Marital status and Income:**

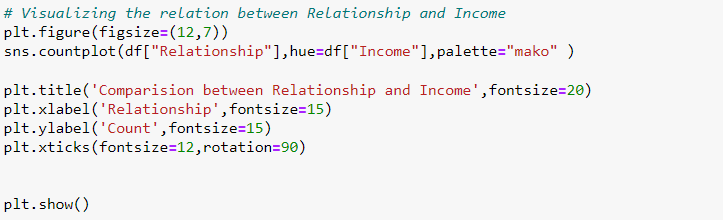
Here we can clearly see,

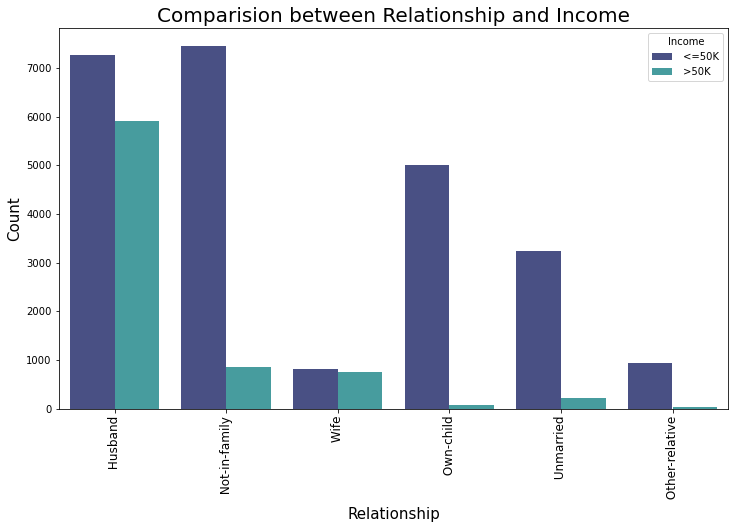
* Married population they have the income >50K compare to others.
* Most of Single population have the income <=50k.
* **Relation between Occupation and Income:**

****

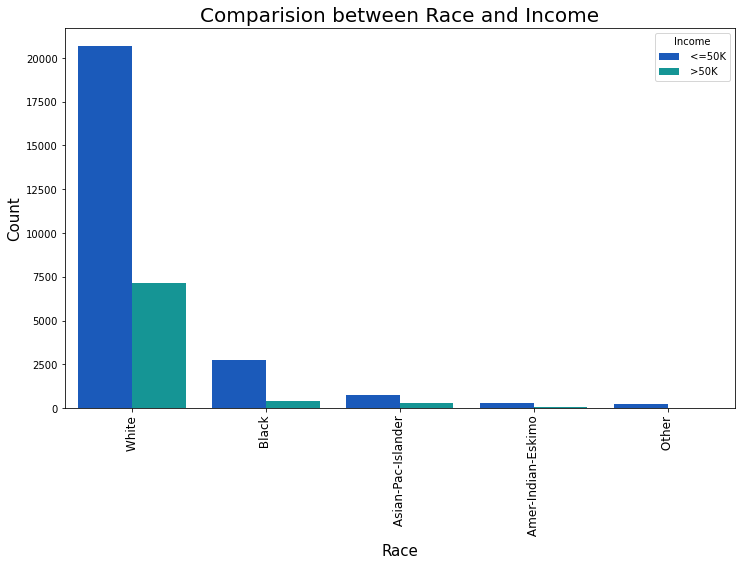
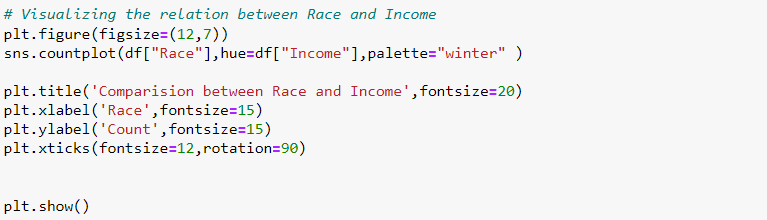


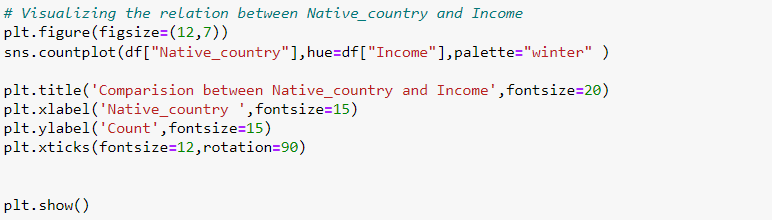
* Population belonging to Prof-speciality and Exce-managerial have the income more than 50K.
* **Relation between Relationship and Income:**

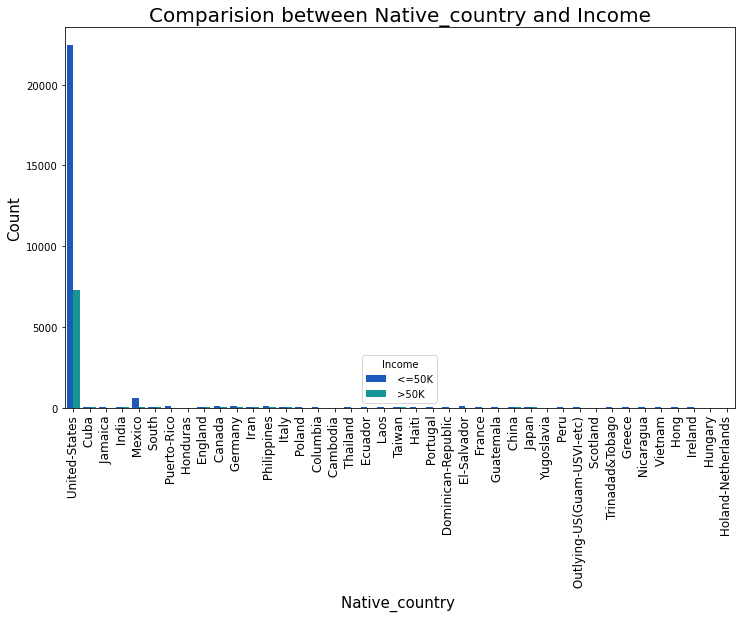




* Population in relationship as Husband has relatively higher income.
* Most of the population which is Not-in-family has income <= 50k.
* **Relation between Race and Income:**

****

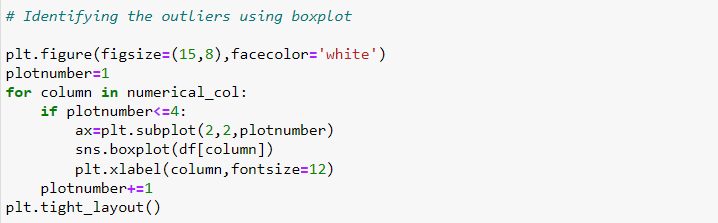
* The white family groups have high income >50K compare to other racial groups.
* **Relation between Native\_country and Income**

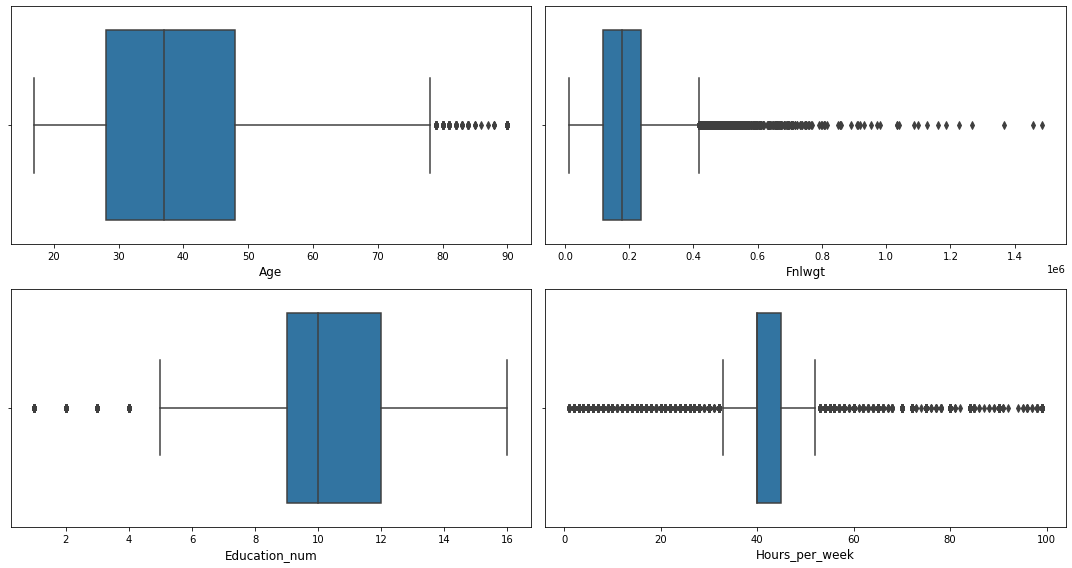


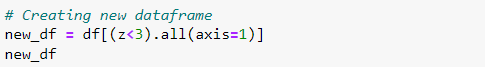
* Clearly United States earning more income compared to all the other countries.

1. **Data Pre-processing pipeline.**

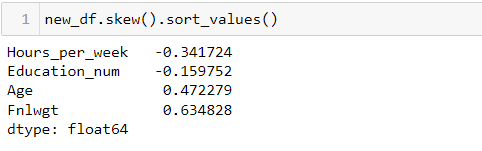
* **Checking for Outliers:**

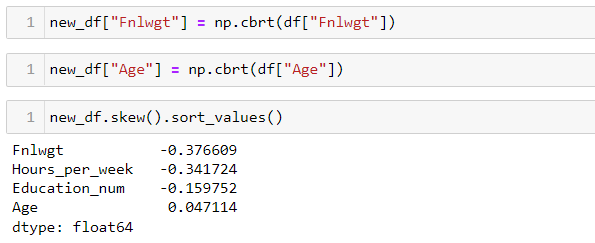
Below is the code for box plots for numerical columns



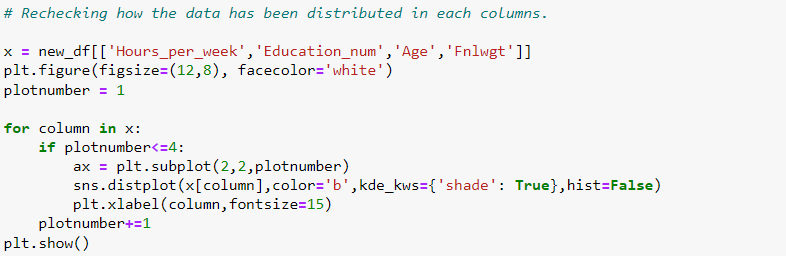
* With this visualization we can clearly see, the outliers present in all the columns we will remove it using Zscore method. Because if we use IQR method we will loose huge amount of data than Zscore.
* **Removing outliers (using Zscore)**
* By removing outliers using above codes, we lost 6% of data.

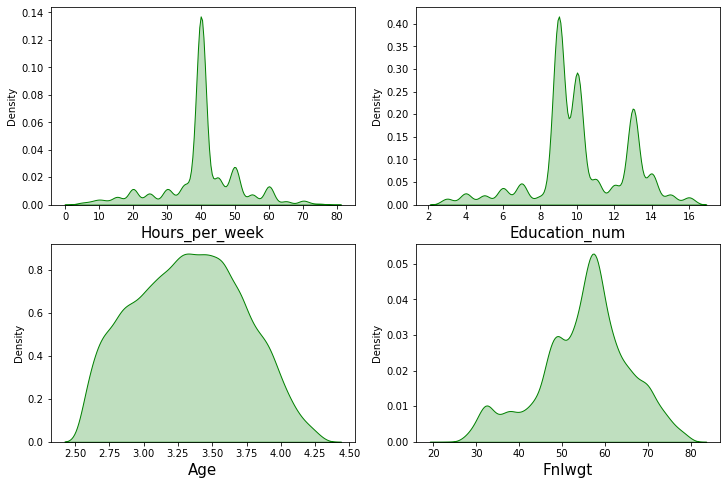
### **Checking the skewness of the dataset:**



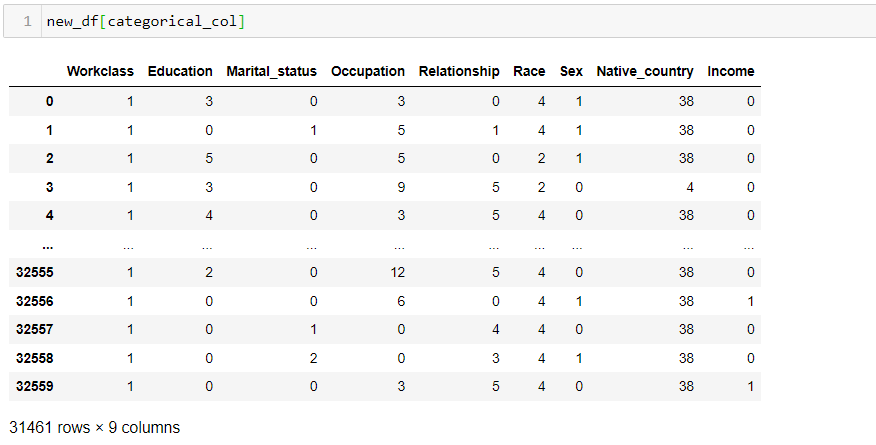
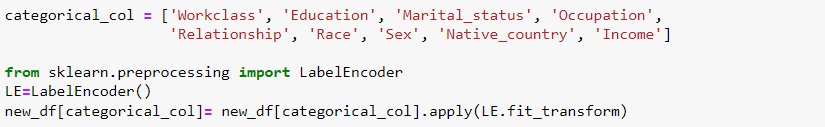
* As we have seen earlier that, data was skewed but now it’s clear. There is skewness present only in the ‘Final Weight’ column (more than +/-0.5 skewness).
* Also, we can see column ‘Age’ also has a considerable skewness.
* **Removing the skewness:**

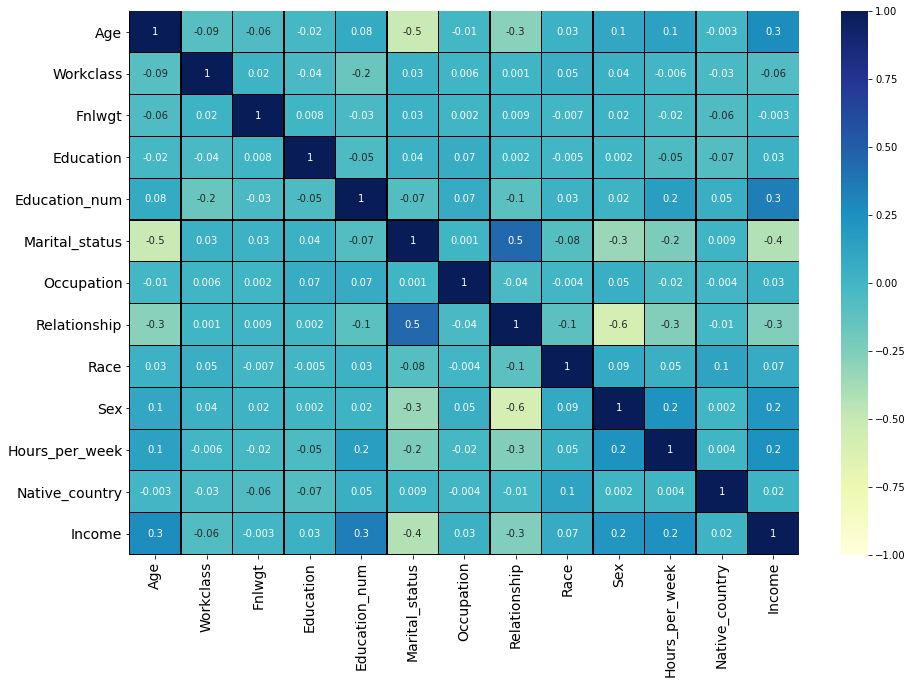
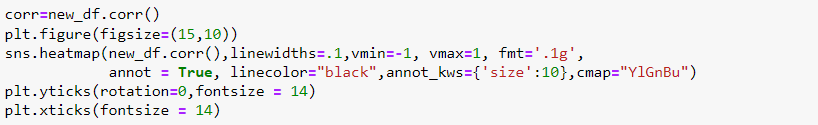
So, I have successfully removed the Skewness from the data. Let’s visualize the data distribution now with the code mentioned below.

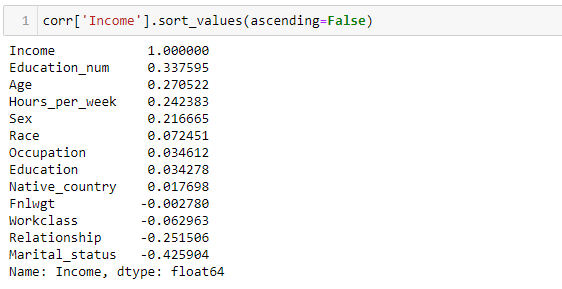




### **Encoding the categorical columns using Label Encoding:**

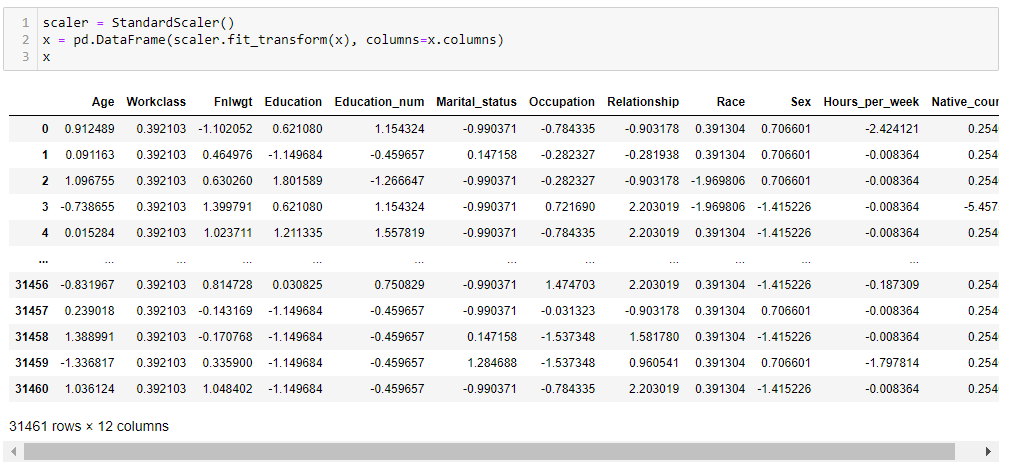
After removing outliers and skewness from the numerical data now we need to do encoding for our categorical features. Here I am using Label Encoder for this purpose.

* Here we can see Our categorical columns has been encoded and ready for modelling. Now we are going to check the Correlation using Heatmap.
* **Checking Correlation:**

****

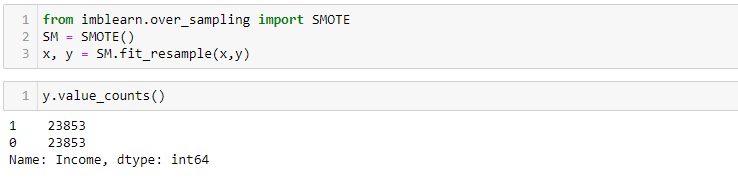
* It is visible from the heatmap and the table, that the relation of features vs target and features vs features. Also, the positive and negative relationship can be seen.
* Here Education\_num is most positive correlated while Marital\_status is most negative correlated with target variable “Income”.
* Column ‘Sex’ and ‘Relationship’ are highly correlated negatively (-60%) followed by column ‘Marital\_status’ and ‘Age’(-50%).

#### **Separating features and target & Feature Scaling.**

Now we will bring all the numerical features to common scale by applying Standard Scaler method. Standard Scaler will remove the mean and scales each feature to unit variance.

### **Oversampling (Class Imbalance):**

As we have seen earlier that the target variable is imbalanced.

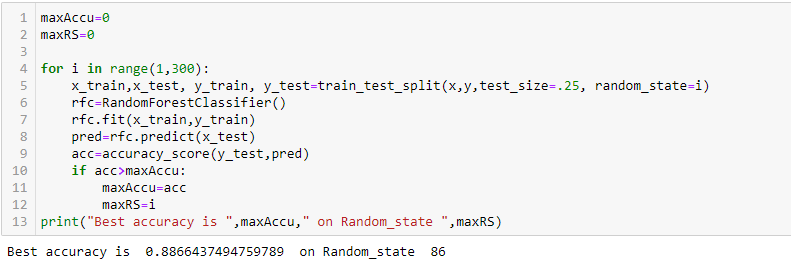
By seeing the count from our target variable, we can say there exist the problem of imbalance. To overcome this issue, I will go for oversampling the data using SMOTE. It will select random examples from minor class with replacement and will add them to training dataset.

* Great we have resolved the problem of imbalance. Now our Data is Balanced and ready for Modelling.

#### **Finding best random state**

Random state will **ensure that the splits that we generate will be reproducible.** The random state that we provide is used as a seed to the random number generator. This will ensure that the random numbers are generated in the same order.

I will select best random state that will give us maximum accuracy with Random Forest Classifier ML model. And with this random state we will split our data for every model.



* Great! We got an accuracy score of 88% at a random state of 86.

### **Creating Train\_Test\_Split:**



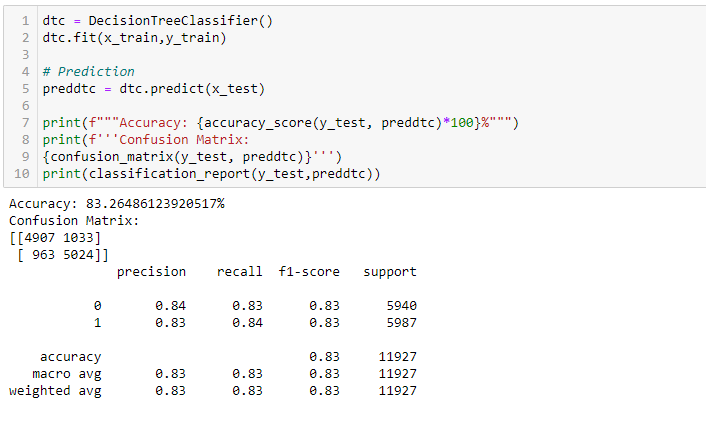
1. **Building Machine Learning Models:**

While building the machine learning model I am using different algorithms to train and test our data after that we will select a best suitable algorithm for final model.

**Evaluation:** I am using many evaluation metrics like cross-validation, confusion matrix , classification report and AUC and ROC curve for selecting best suitable algorithm.

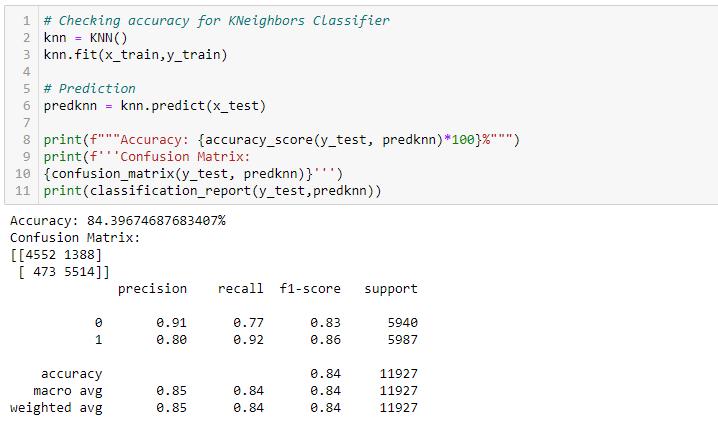
## **Logistic Regression:**

## **Decision Tree Classifier:**

****

## **Random Forest Classifier:**

## **KNNeighbors Classifier:**



## **Xtreme Gradient Boosting Classifier:**

# Checking the Cross Validation Score

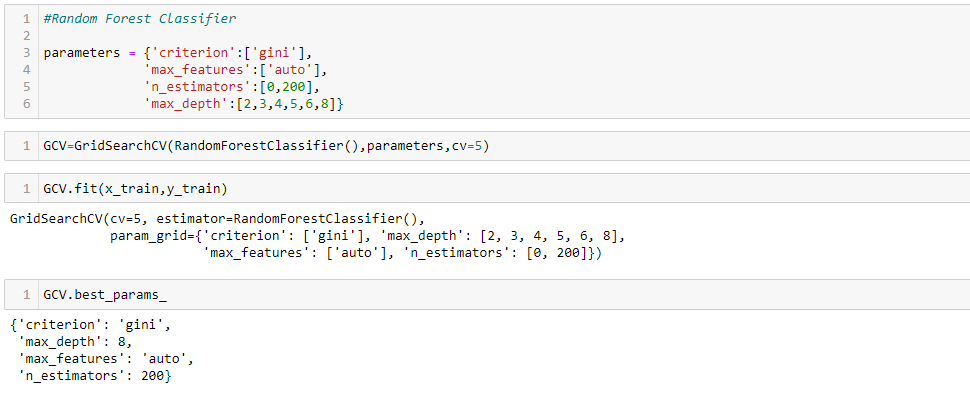


Above are the cross-validation score for all the models used:

* The difference between accuracy score and cross validation score of Gradient Boosting Classifier is very less compared to other models.
* So, **we can conclude that 'Random Forest Classifier' as our best fitting model.**

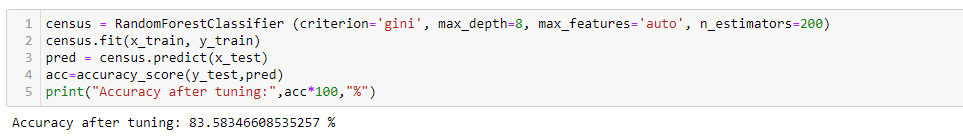
# Hyper Parameter Tuning

Below you can see the code of the hyperparameter tuning



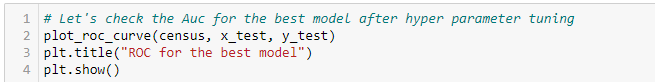
Great! we got best parameters for Random Forest Classifier: {'criterion': 'gini', 'max\_depth': 8, 'max\_features': 'auto', 'n\_estimators': 200}. So, lets check accuracy score with these parameters.

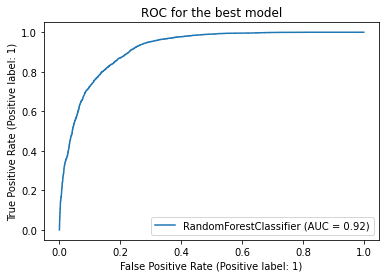
#### **Final Model**



* The accuracy of the best model after tuning 83.50 %.

## **Plotting ROC and Compare AUC for the best model:**





* We can see the above improved **AUC ROC curve** showing better performance for final model. (92%)

### **Conclusion:**

* In Data column Education\_num is most positive correlated while Marital\_status is most negative correlated with target variable “Income”.
* Column ‘Sex’ and ‘Relationship’ are most negatively correlated (-60%) followed by column ‘Marital\_status’ and ‘Age’(-50%).
* The final Accuracy for the prediction of 'Income' (whether the person has annual income <=50K or >50K) of Census bureau database (1994) using Classification algorithm is = 83%.

So, we just executed an entire Data Science Project from scratch.

**6.     Concluding Remarks**  
  
Kindly allow me to provide a quick recap on all the steps that we went through starting from understanding the Problem Definition then going through the Data Analysis and EDA processes. We went through the necessary Pre-processing Data steps before the final Building Machine Learning Models step came into picture.  
  
What I do is code my entire project on my own and then take a peek at the internet to look through other’s coding style for inspiration and understand if I can incorporate anything to improvise further on accuracy or beautify the visuals. However, I have seen many people doing the complete opposite whereupon they don’t practise or create their own unique coding style first and rather copy paste lines from the web and perform some sort of messy patch work and when asked to explain might not be capable of conveying functioning or usage of those code blocks.  
  
Before wrapping up my only advise to everyone is “No pain No gain” you will have to get your hands dirty with building your own code and trying out all the permutations and combinations. Create a self-made unique data story telling commandment list and follow it along with the standard project life cycle. Hope this at length article helps you in gaining the initial knowledge on building your first project from scratch.

**Disclaimer:** I am a newbie myself in this field with some accumulated knowledge over a year’s time studying Data Science and felt like since sharing is caring someone who’s stepping in now can benefit from my experience. I am also open to get some feedback from anyone that will help me in improving too! The content that I have written is solely my view of the project but it’s definitely inspired by others over the internet who have worked on similar projects before me.