**‘Census Income Dataset’ model creation to check the income of person if it is >50K or <=50K**

By *Chandralekha Raut*

With more emphasis on knowledge based industry, the payment forecasting is becoming a key strategic area for industries to ensure continuous growth and success. One of the problems which industries face till today is retaining high performing employees and also hires talented people from other industries. In both the cases, salary is a key significant aspect of tempting current as well as future employees. Hence a better salary offer is extremely important for retaining or attracting employees to any industry.

The prominent inequality of wealth and income is a huge concern especially in the United States. The likelihood of diminishing poverty is one valid reason to reduce the world’s surging level of economic inequality. The principle of universal moral equality ensures sustainable development and also improves the economic stability of a nation. Governments in different countries have been trying their best to address this problem and also provide an optimal solution.

Extraction was done by Barry Becker from the 1994 Census database. A set of reasonably clean records was extracted using the following conditions: ((AAGE>16) && (AGI>100) && (AFNLWGT>1)&& (HRSWK>0))

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

**We will cover below points in the blog**

*1. Problem Definition*

*2. Data Analysis*

*3. EDA Concluding Remarks*

*4. Pre-processing Pipeline*

*5. Building Machine Learning Models*

*6. Concluding Remarks*

**Source of Dataset**: The dataset for this project is received from below link

[*http://archive.ics.uci.edu/ml/datasets/Census+Income*](http://archive.ics.uci.edu/ml/datasets/Census+Income)

**Problem Definition:**

Income is money (or some equivalent value) that an individual or business receives, usually in exchange for providing a good or service or through investing capital. Income is used to fund day-to-day expenditures.

Income directly affects the standard of living. It affects many things as we can say it will be social life too. Many times people differentiate on the basis of economic condition. Sometimes, we can see the situation like if people earn more then only community people mingle and makes good relationships; otherwise they just ignore the person who earns less.

Many classification tasks do not have equal of examples from each class, data might be skewed. The data set contains the features about the people who earn more than $50K or less than $50K a year. The prediction task is to determine whether a person makes over $50K a year. Exploratory data analysis is done on the data set to achieve insights and to get the important features then the preprocessing is done to get the data ready for the training.

The dataset provided from adult census income dataset to predict the income.

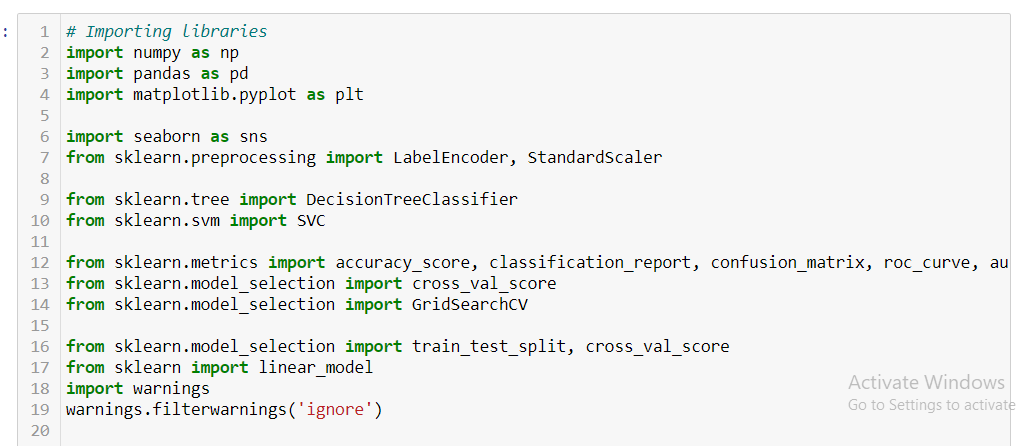
**Data Analysis:**

The income of a person depends on various factors. Data may be number or categorical. The provided dataset has a number of features. Census Data to come up with crucial and exciting attributes of the data. The dataset comes with various information such as 'Age', 'workclass', 'fnlwgt', 'education', 'education-num', 'marital-status', 'occupation', 'relationship', 'race', 'sex', 'capital-gain', 'capital-loss', 'hours-per-week', 'native-country', 'Goal' etc.

For data analysis, we need to get the data and observe it properly.

**Importing Libraries**

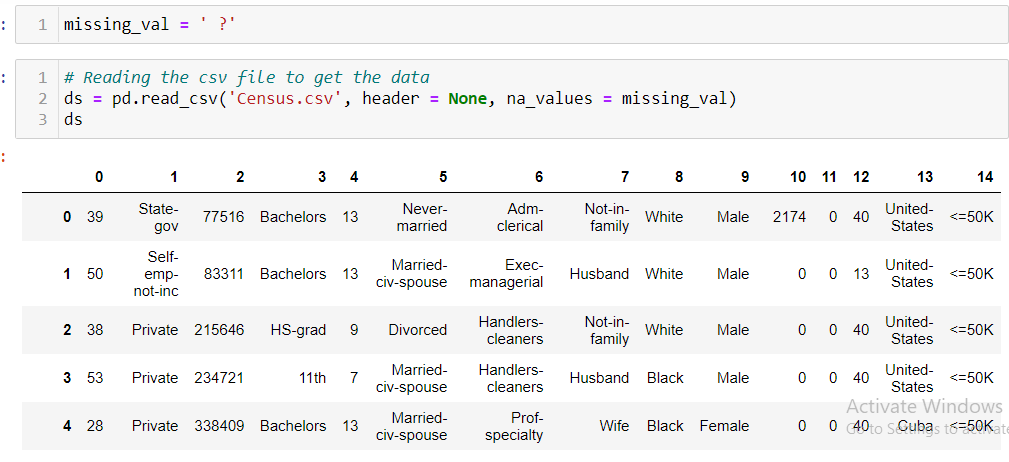
First, we need to import the libraries that will be used in creating a model.



**Reading the csv file**

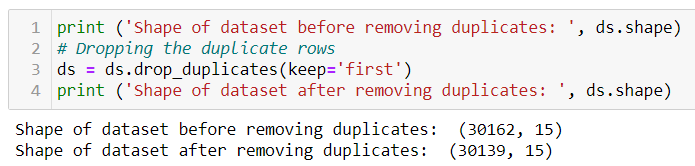
Reading the csv file which is the main source of our model creation. This is the dataset which contains all the information about a person’s income, on the basis of which we will be predicting whether a person is earning more than 50K or not.

The provided dataset contains the ‘ ?‘ instead of value, we are considering it as NaN values while reading the file, so that later we can handle it properly.



**EDA/Exploratory Data Analysis**

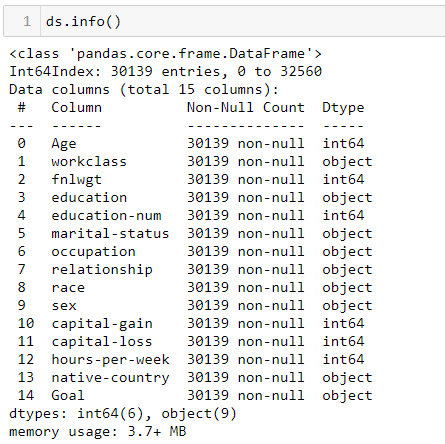
In the provided dataset we have some unwanted value i.e. ’ ?’, we might have NaN values, we need to either remove it or replace it with some other value as it will affect our target value. Sometimes we have duplicates also in dataset. As we all know, duplicates are nothing but the unnecessary stuff, so we should remove it always.



We can see from the above code snippet that duplicates are not available in our dataset. Initial size of the dataset is *32561 rows × 15 columns.*

It has various data types’ columns which includes *‘category’, ‘int64’ and ‘float64’.* Our dataset has only integer and categorical data.

It has 6 integer columns and 9 categorical data.



**Feature Description:**

Now let’s see the small description about the features.

*Age* : Age of a person

*workclass* : Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.

*fnlwgt*  : continuous

*education* : Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.

*education-num* : continues

*marital-status* : Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse.

*occupation* : Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces.

*relationship*  : Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.

*race*  : White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.

*sex* : Female, Male.

*capital-gain* : continuous.

*capital-loss* : continuous

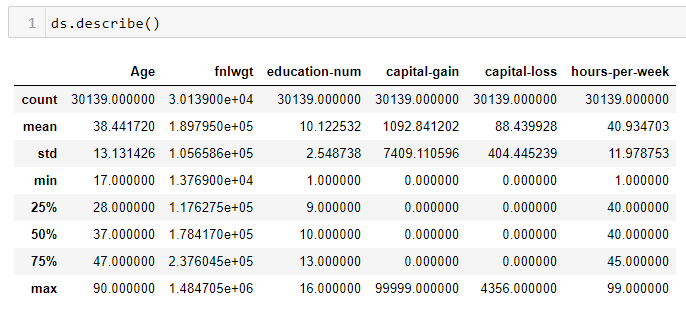
*hours-per-week* : continuous

*native-country* : United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinadad&Tobago, Peru, Hong, Holand-Netherlands.

*Goal*  : There are two class values ‘*>50K*‘ and ‘*<=50K*‘,

Meaning it is a **binary classification task**. The classes are imbalanced.

**Statistical Summary**

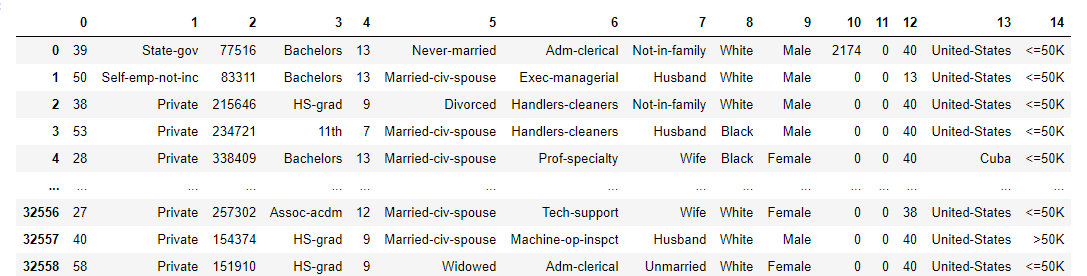


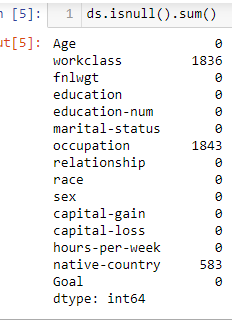
We can observe that starting age for earning the income is 17 and a person can earn till the age of 90. Most of the persons are the age of 38. Average working hours for a week is 40 hrs.

**Viewing Data**

We can see the data by using head command; it will show the first 5 rows of data. We can see any number of rows by writing the number in the bracket. There are different types of data present in the dataset. We will convert the non-number data into numbers by using the encoding methods. For NaN values, we will either replace it or will remove it. The data in the dataset in widely distribute, need to use scaling techniques to convert into same scale. Target value depends on the all preprocessing which we will do on the data.

We can observe that some of the values are unexpected. As the data is too large, so it’s not visible.





*There are NaN/missing values in the dataset in 3 columns which are ‘workclass’, ‘occupation’* and *‘native-country’*.

workclass has 1836, occupation has 1843 and native-country has 583 missing values. We will handle these values significantly.

**Columns of Dataset**

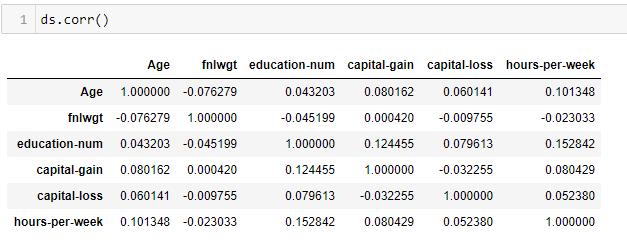
Below are the columns of dataset which plays a vital role in predicting the target.

*'Age', 'workclass', 'fnlwgt', 'education', 'education-num', 'marital-status', 'occupation', 'relationship', 'race',*

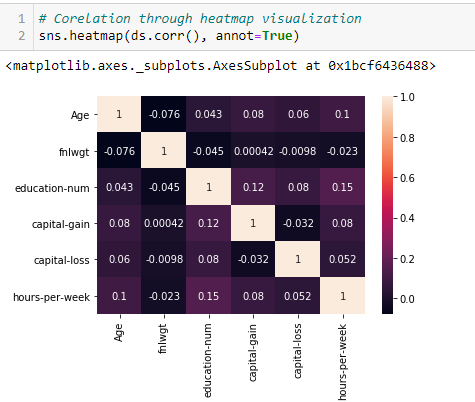
*'sex', 'capital-gain', 'capital-loss', 'hours-per-week', 'native-country', 'Goal'*

**Correlation of features**

Correlation between the features as shown below.



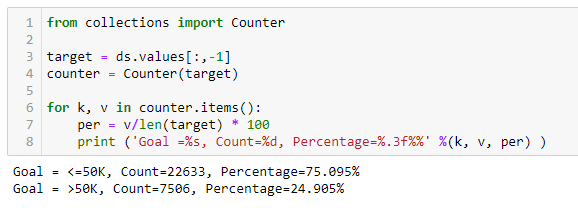
Below is correlation through heat map visualization.



All the above features play an important role in model creation.

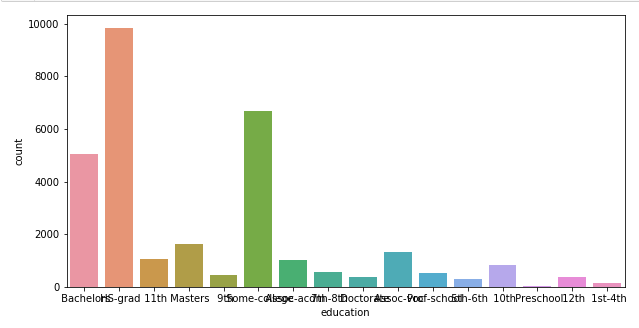
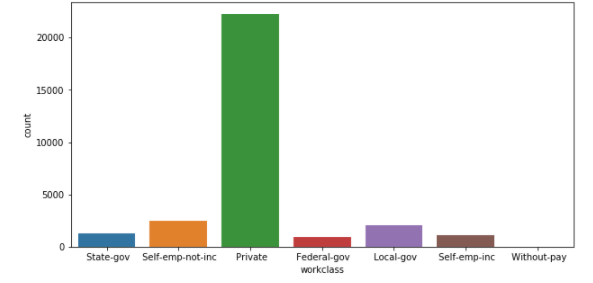
**Summarize the target for classification**

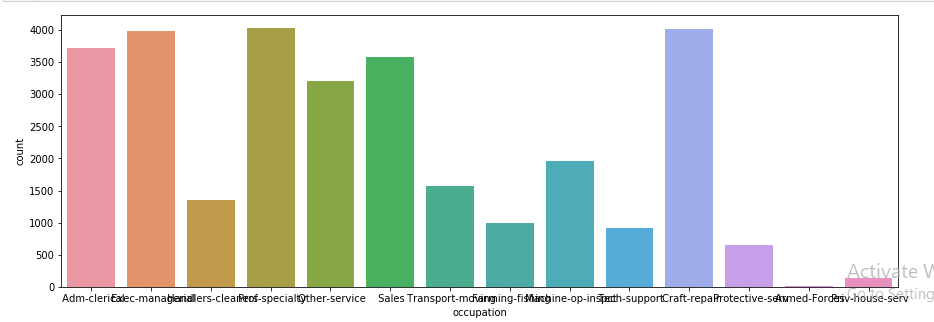
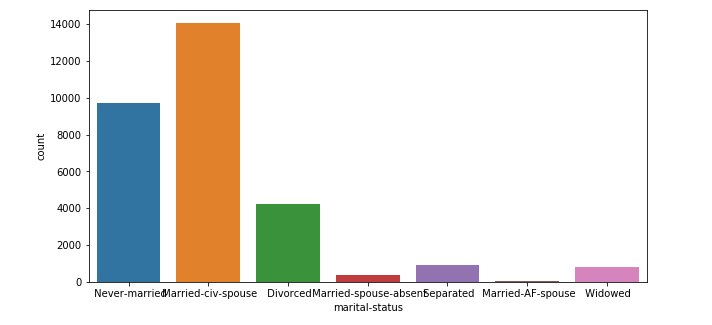
Below is the summary of the target. We can do this for classification as it has multiple classes and the values are not continues.

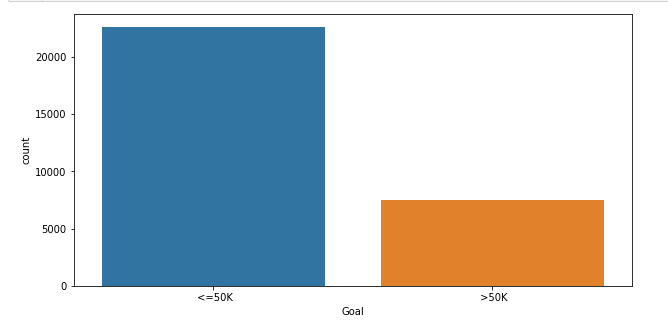


We can see that, 22633 people have income <50K while 7506 people have income >50K. Percentage calculation is available for the same.

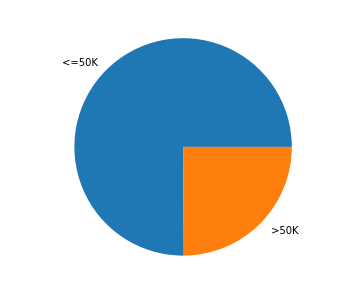
**Graphical representation of the features**



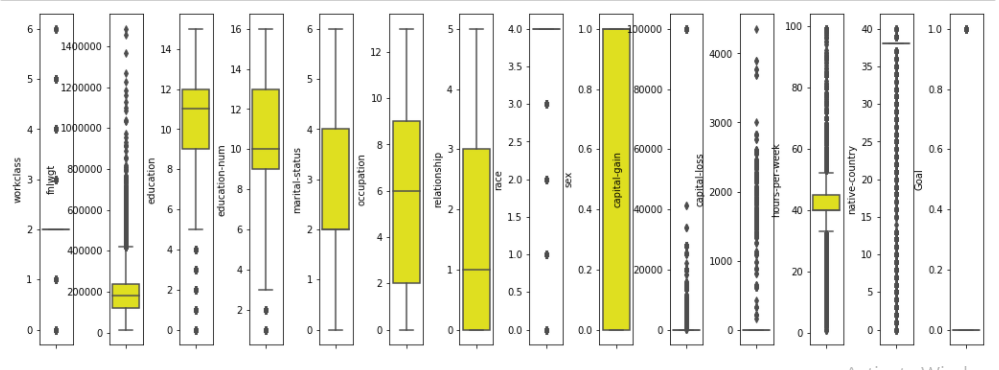




Visualization of income in pieplot

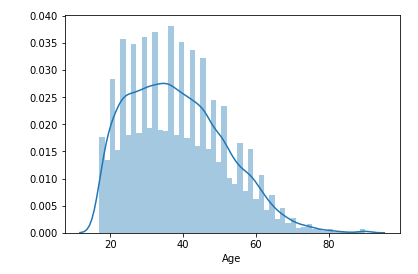
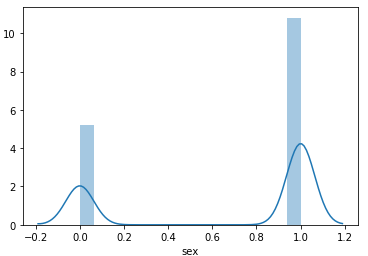


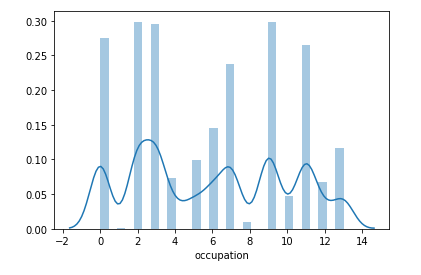
Boxplot is one of the ways to plot outliers.



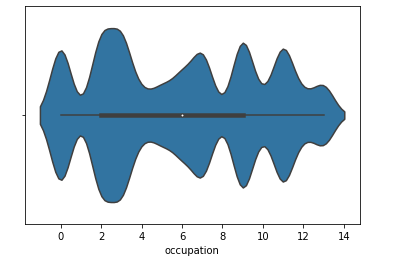
Skew ness of the data can be checked by using the distribution plots. The skew ness is data out of scale.

We will do it for some of the columns.

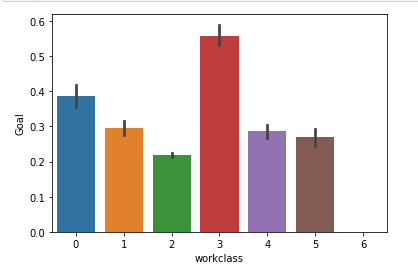
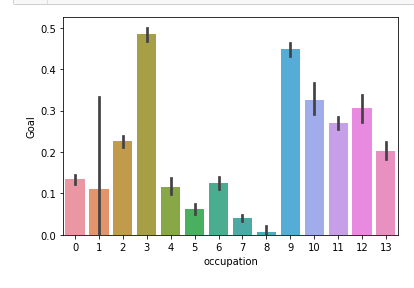


Below is the violin plot which shows which value is occurred more in occupation in the dataset.

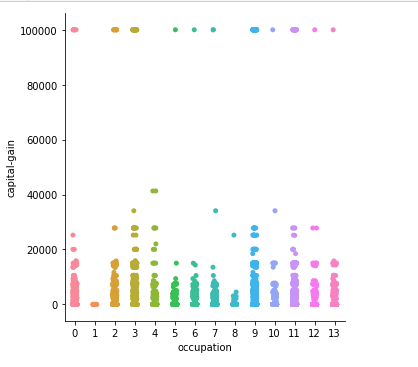
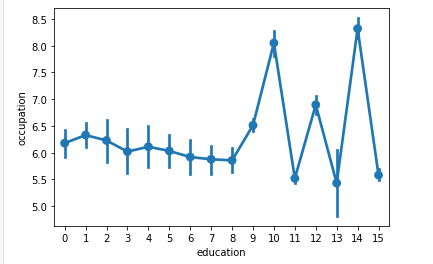


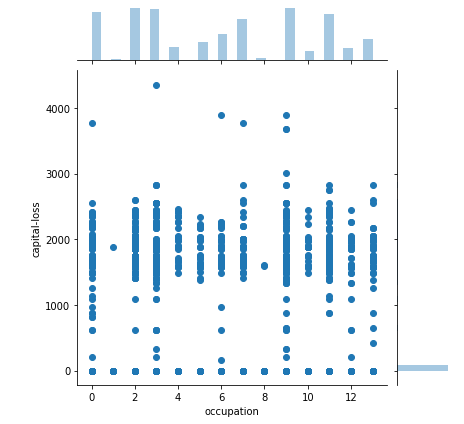
Most of the person’s occupation is 2.

Now, we will compute the target with other features, its bivariate analysis.

Most of the people work in workclass 3 with occupation is also 3.



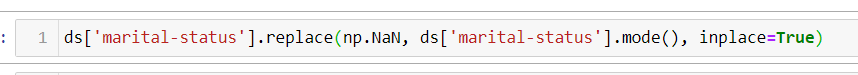
Bivariate analysis is used to check how one feature varies with other. Bivariate analysis is one of the simplest forms of quantitative (statistical) analysis. It involves the analysis of two variables (often denoted as X, Y), for the purpose of determining the empirical relationship between them. Bivariate analysis can be helpful in testing simple hypotheses of association.

* We can compute many things like which country has more number of people earning more than 50K
* Which occupation provides more income to people
* Which education is best for higher earning
* We can know what will be best occupation so than a person can earn more than 50K
* We can also compute how working hours in a week vary and how it affects the salary.
* We can plot multiple graphs which will show us how the increase or decrease in one parameter affect the other

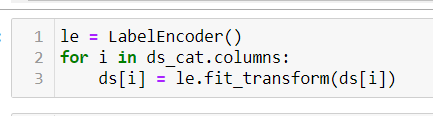
**Data Preprocessing**

Before we move ahead for model creation, we should clean the data and make it proper so that our target value will be accurate. For that we need to perform some preprocessing on the data. We should handle missing values, NaN values and unexpected values accordingly. We will remove the skew ness of data by using log transform or any other available methods. We will then remove the outliers by using zscores and then will scale the data using StandardScalar. We will use LabelEncoder to convert categorical data to number data.

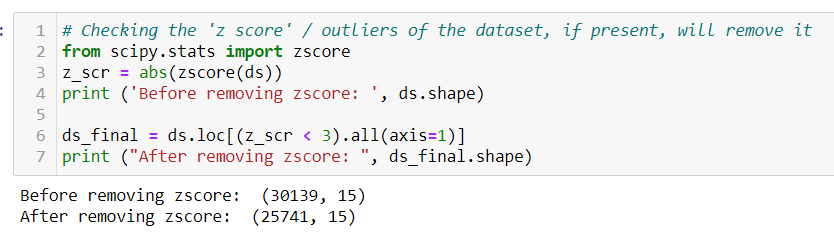
**Missing values substitution**



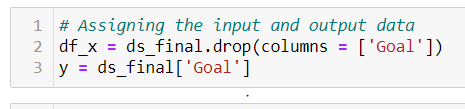
**Label Encoder to convert column data type to number**



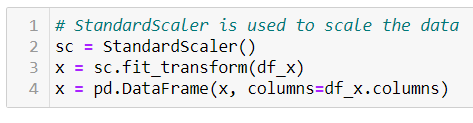
**Outliers removal using zscore**



**Setting input and output variable**



**Using StandardScalar to scale the data**



**Building Machine Learning Model**

Now we will train several Machine Learning models and compare their results. Later on, we will use cross validation.

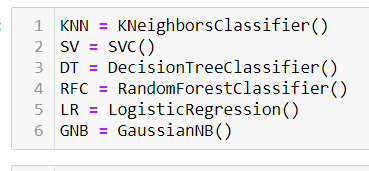
We will *check the maximum accuracy score based on the best random state.*

Once the data is in usable shape and you know the problem you're trying to solve, it's finally time to move to the step you long to do: Train the model to learn from the [good quality data](https://searchdatamanagement.techtarget.com/feature/Proactive-practices-for-data-quality-improvement) you've prepared by applying a range of techniques and algorithms.

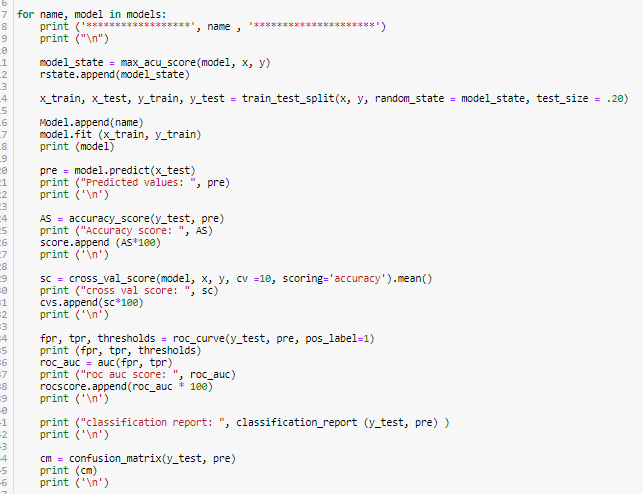
This phase requires model technique selection and application, model training, model hyperparameter setting and adjustment, model validation, ensemble model development and testing, algorithm selection, and model optimization. To accomplish all that, the following actions are required:

* Select the right algorithm based on the learning objective and data requirements.
* Configure and tune hyperparameters for optimal performance and determine a method of iteration to attain the best hyperparameters.
* Identify the features that provide the best results.
* Determine whether [model explainability](https://searchenterpriseai.techtarget.com/feature/How-to-achieve-explainability-in-AI-models) or interpretability is required.
* Develop ensemble models for improved performance.
* Test different model versions for performance.
* Identify requirements for the model's operation and deployment.
* The resulting model can then be evaluated to determine whether it meets the business and operational requirements.

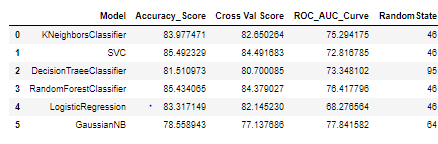
We are going to use the below algorithms for model creation.



**Model Processing**



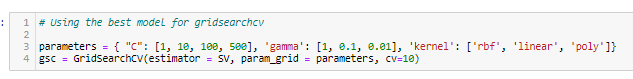
After performing the above process, we get the results of the various algorithms as mentioned below.

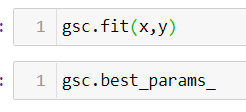


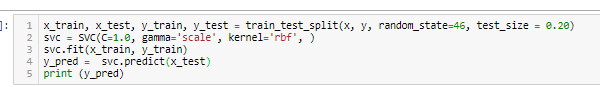
Now, we are going to select the best model from the above which give high accuracy score along with high cross validation score. SVC is the one.

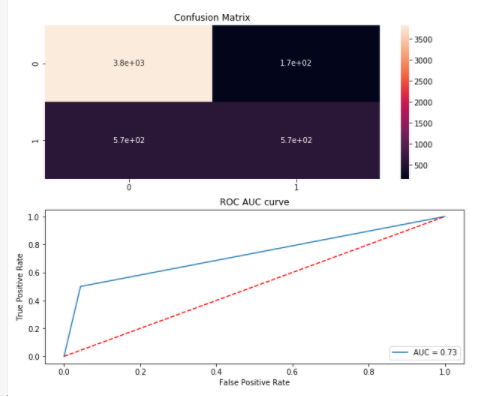
**Hyperparamter Tuning**

In [machine learning](https://en.wikipedia.org/wiki/Machine_learning), hyperparameter optimization or tuning is the problem of choosing a set of optimal [hyperparameters](https://en.wikipedia.org/wiki/Hyperparameter_(machine_learning)" \o "Hyperparameter (machine learning)) for a learning algorithm. A hyperparameter is a [parameter](https://en.wikipedia.org/wiki/Parameter) whose value is used to control the learning process. By contrast, the values of other parameters (typically node weights) are learned.









* Evaluate the models using a validation approach and a validation data set.
* Determine confusion matrix values for classification problems.
* Identify methods for k-fold cross-validation if that approach is used.
* Further tune hyperparameters for optimal performance.
* Compare the machine learning model to the baseline model or heuristic.

**Conclusion**

Every time the model is created with the best suited algorithm. The best algorithms will always give a best performance with more accuracy. Always need to achieve the more accuracy while creating the model.

The dataset which we receive for creating the model is not always clean and perfect; we should make it perfect for the better model. We need to apply some techniques to fit it in some scale. We need to use LabelEncoder to convert categorical data to numbers.

At a conceptual level, we’re building a machine that given a certain set of inputs will produce a certain desired output by finding patterns in data and learning from it. A very common case is for a machine to start by looking at a given set of inputs and a set of outputs that correspond to those inputs. It identifies patterns between them and creates a set of complex rules that it can then apply to new inputs it hasn’t seen before and produce the desired output.