**Data Science Capstone – Healthcare**

**Report created by: Sravanthi Baman**

**DESCRIPTION**

NIDDK (National Institute of Diabetes and Digestive and Kidney Diseases) research creates knowledge about and treatments for the most chronic, costly, and consequential diseases.

* The dataset used in this project is originally from NIDDK. The objective is to predict whether or not a patient has diabetes, based on certain diagnostic measurements included in the dataset.
* Build a model to accurately predict whether the patients in the dataset have diabetes or not.

**Dataset Description**

The datasets consists of several medical predictor variables and one target variable (Outcome). A predictor variable includes the number of pregnancies the patient has had, their BMI, insulin level, age, and more.

| **Sr.no** | **VARIABLES** | **DESCRIPTION** |
| --- | --- | --- |
| 1. | PREGNANCIES | Number of times pregnant |
| 2. | GLUCOSE | Plasma glucose concentration in an oral glucose tolerance test |
| 3. | BLOODPRESSURE | Diastolic blood pressure (mm Hg) |
| 4. | SKINTHICKNESS | Triceps skinfold thickness (mm) |
| 5. | INSULIN | Two-hour serum insulin |
| 6. | BMI | Body Mass Index |
| 7. | DIABETESPEDIGREEFUNCTION | Diabetes pedigree function |
| 8. | AGE | Age in years |
| 9. | OUTCOME | Class variable (either 0 or 1). 268 of 768 values are 1, and the others |

**Project Task (Week 1)**

**Data Exploration:**

1. Perform descriptive analysis. Understand the variables and their corresponding values. On the columns below, a value of zero does not make sense and thus indicates missing value:

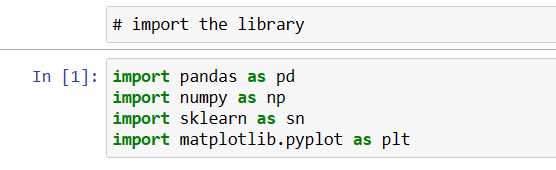
* Glucose
* Blood Pressure
* Skin Thickness
* Insulin
* BMI

1. Visually explore these variables using histograms. Treat the missing values accordingly.
2. There are integer and float data type variables in this dataset. Create a count (frequency) plot describing the data types and the count of variables.
3. Check the balance of the data by plotting the count of outcomes by their value. Describe your findings and plan future course of action.
4. Create scatter charts between the pair of variables to understand the relationships. Describe your findings.
5. Perform correlation analysis. Visually explore it using a heat map.

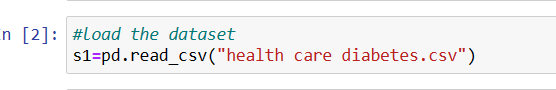
**Methodology for Task for Week -1**

**Note: All Python codes were written inside capstone. ipynb file. Attached this file set along with the report folder.**

**1. Import all necessary libraries for this project. The Screenshot are attached below.**

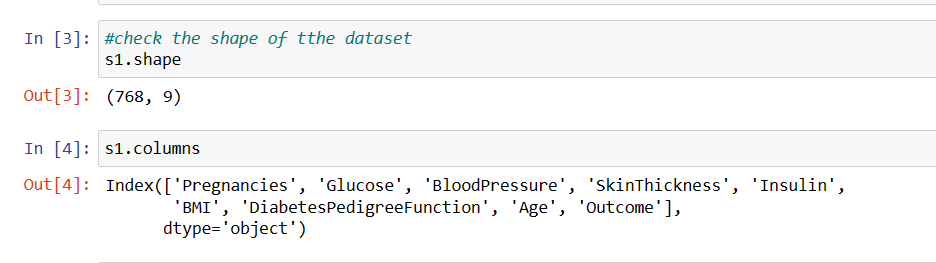


**2. Load the dataset and read the csv file with pandas**

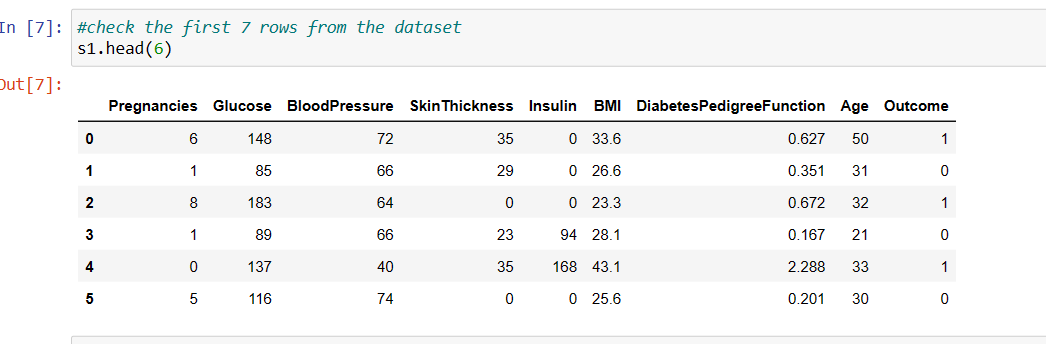
****

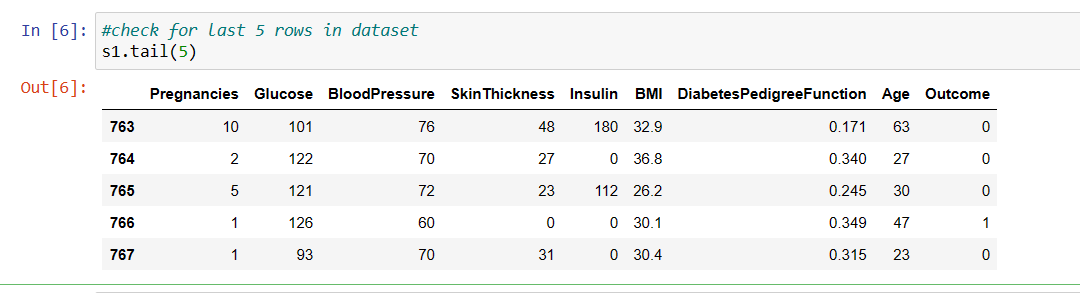
**3. After reading csv file, we have to perform some following steps on the data set.**

* **See the shape and columns in dataset**

****

* **Read first rows from the dataset.**

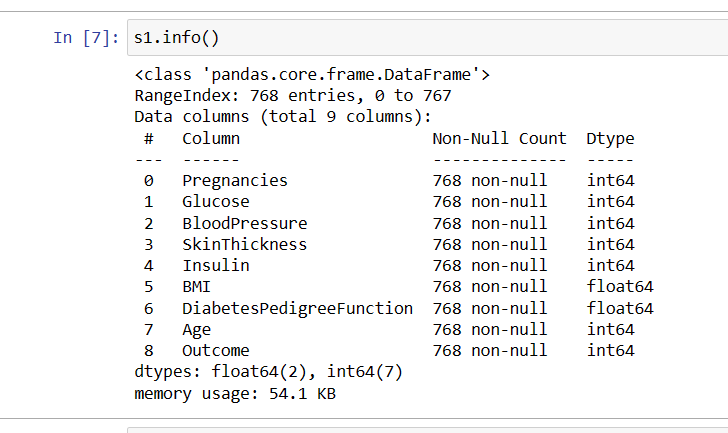
****

**2. Read the last five rows from the dataset**

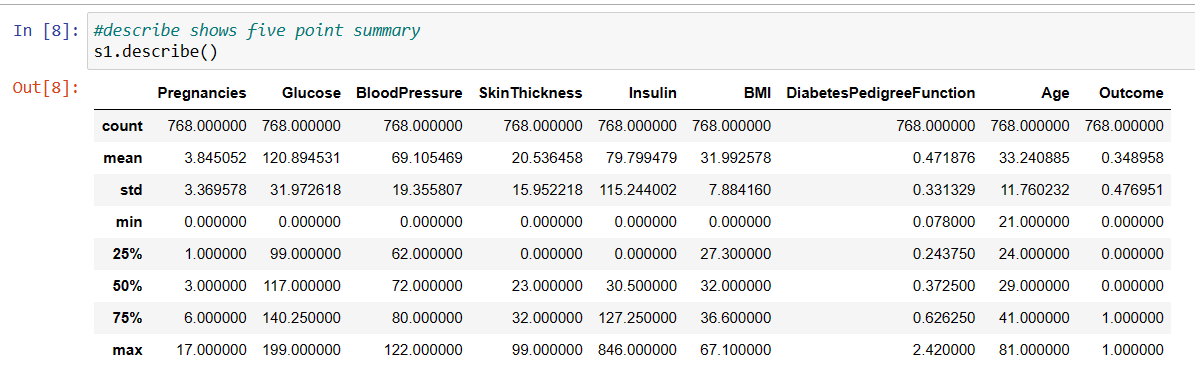
**3. Read the Info from Dataset**

**Info: it** prints information about the Data Frame, which contains the number of columns, column labels, column data types, memory usage, range index, and the number of cells in each column (non-null values).

**Note**: the info () method actually prints the info.

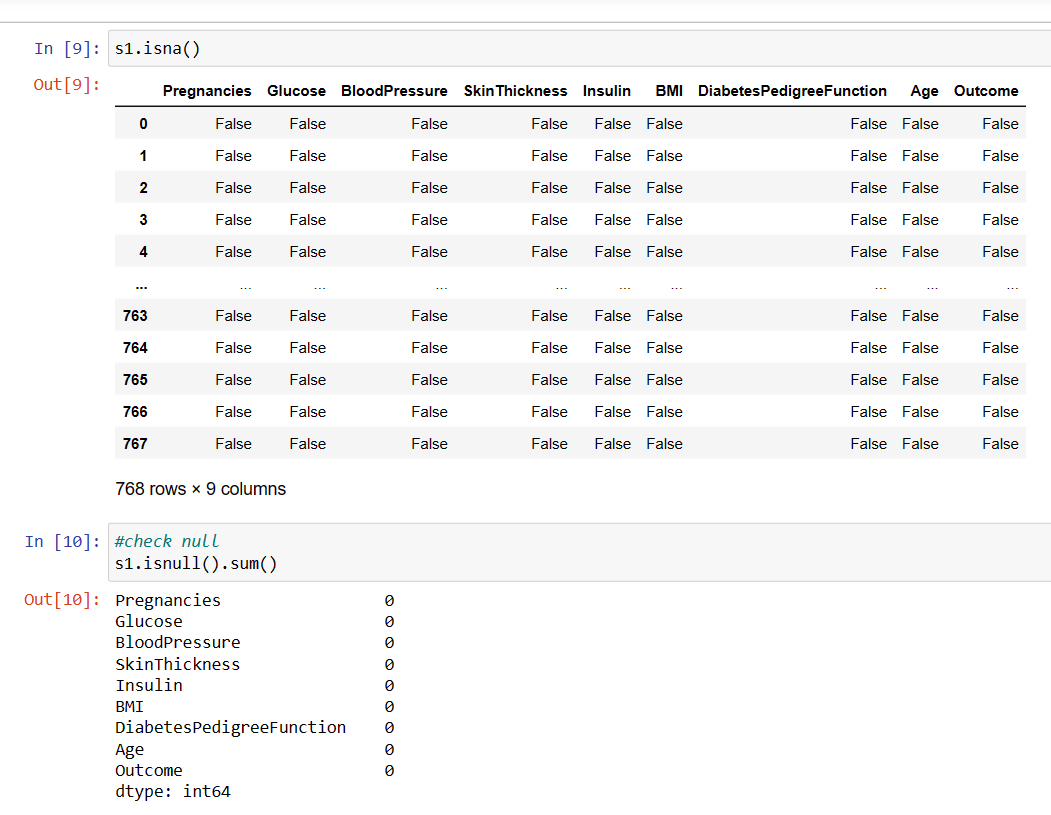


**Describe: It** returns a description of the data in the Data Frame. If the Data Frame contains numerical data, then the description shows the information for each column: count The number of not-empty values. mean: - The average (mean) value, std - The standard deviation.

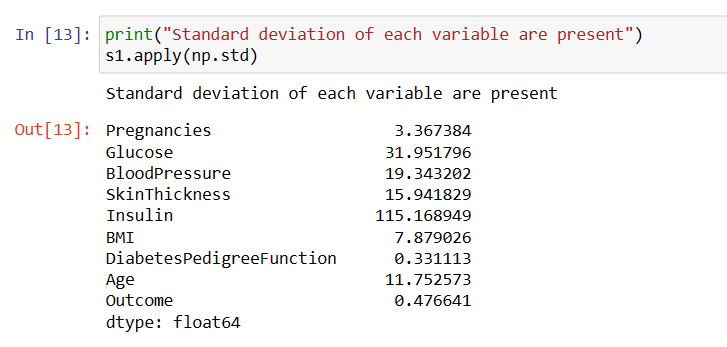


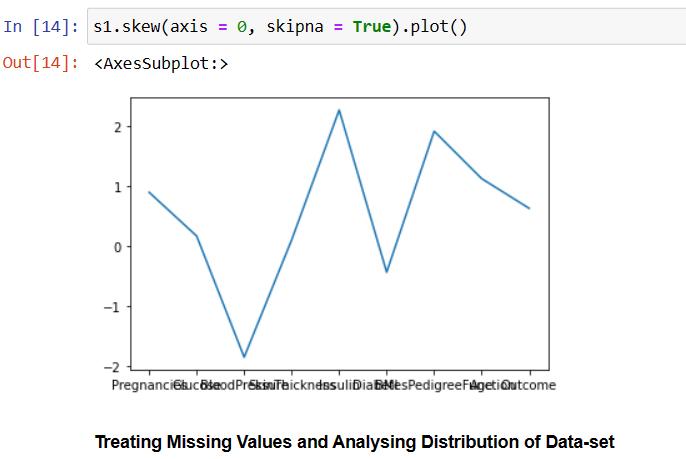
* **Check if any null value is present in dataset**

**Isnull & sum: It** returns how many null values are present inside this dataset & its total count.



* **Check Standard deviation and skew of each variable in dataset**

****

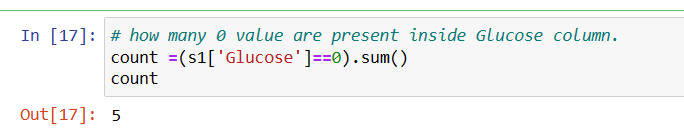
****

**Perform Descriptive Analysis**

**Visually explore these variables using histograms. Treat the missing values accordingly.**

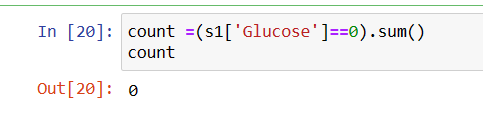
**Case for Variable Glucose**

**Before treating missing value.**

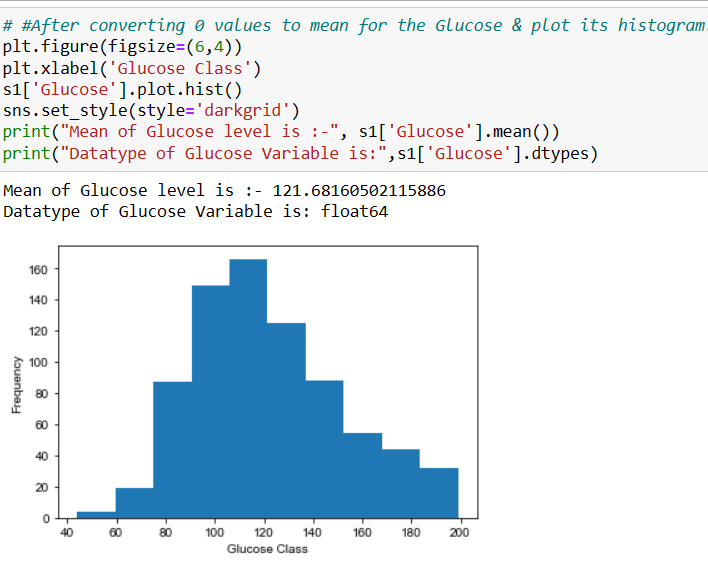
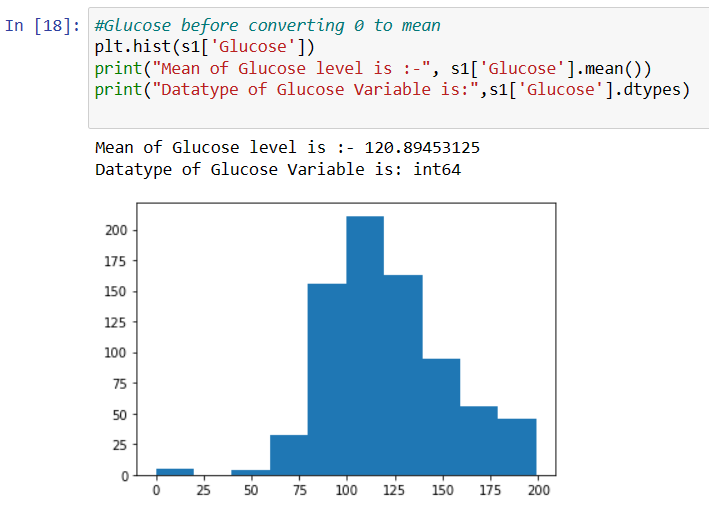


**Handling the missing value.**





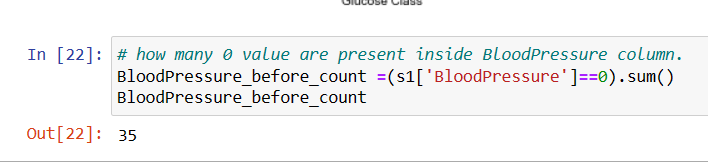
**Before**  **After**



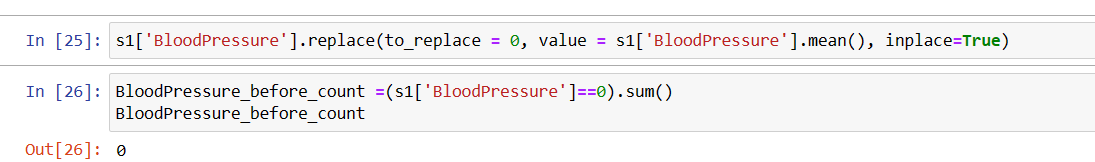
If, we compare the Histogram before & after the missing value treatment, it shows like this:

**Case for Variable Blood Pressure**

**Before treating missing value.**

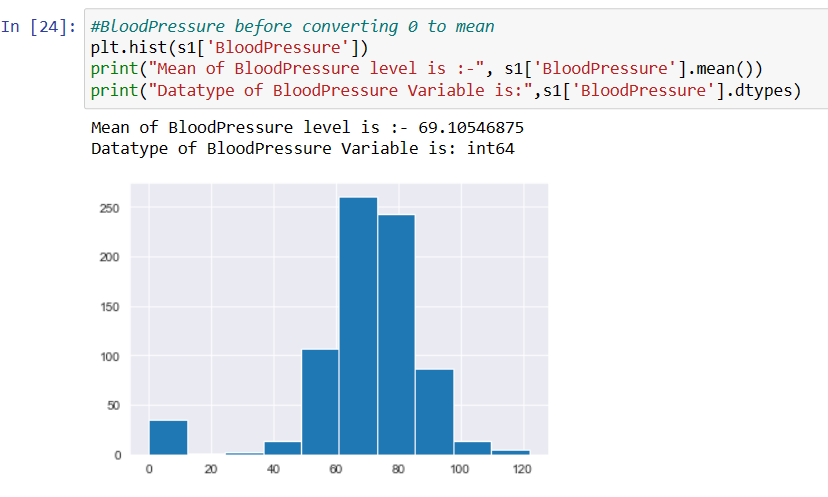
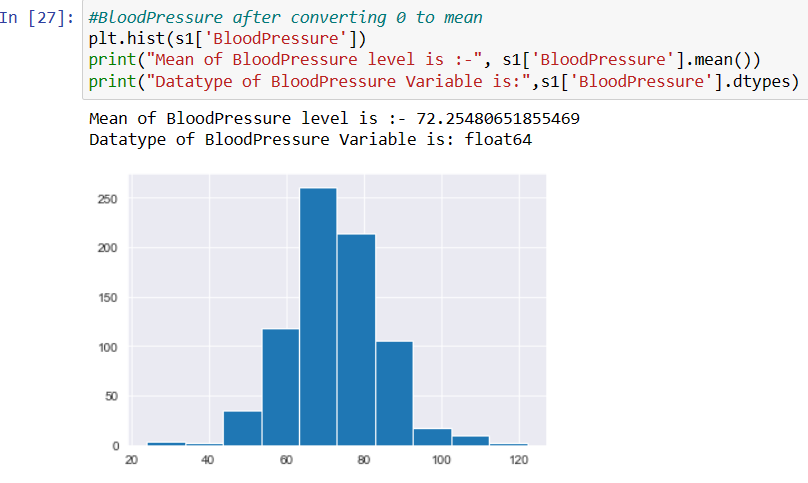


**Handling the missing value.**



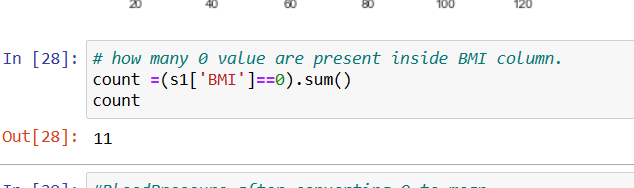
If, we compare the Histogram before & after the missing value treatment, it shows like this:

**Before**  **After**

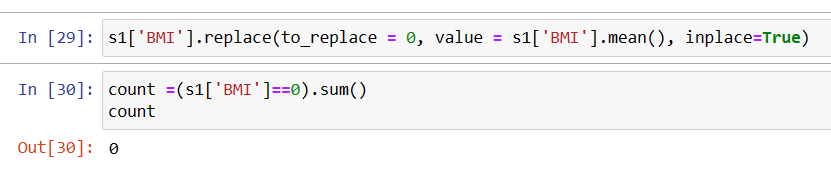


**Case for Variable BMI**

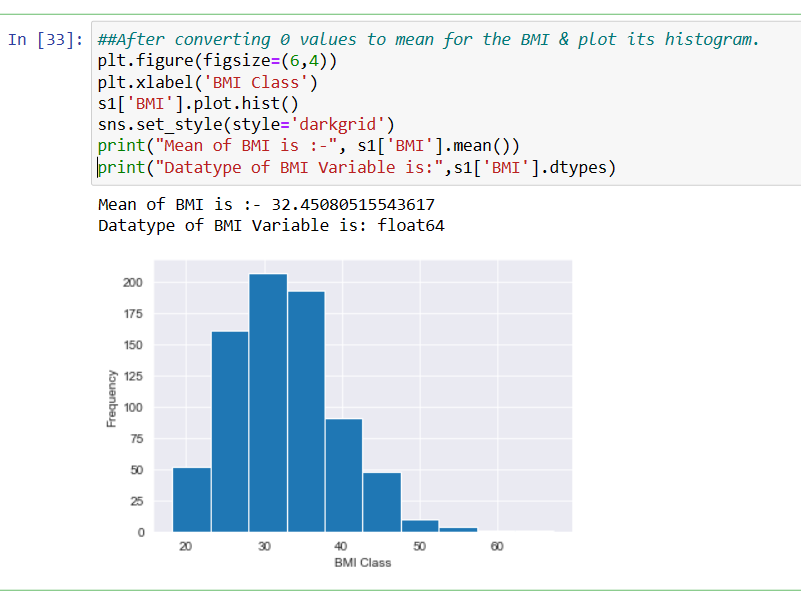
**Before treating missing value.**

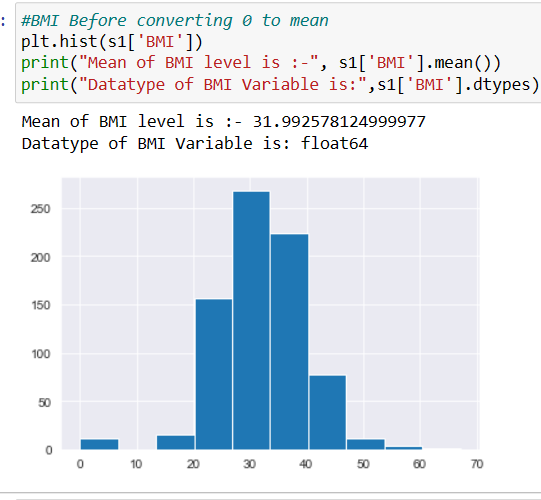


**Handling the missing value.**



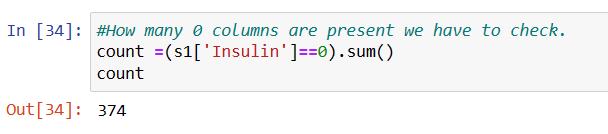
If, we compare the Histogram before & after the missing value treatment, it shows like this:

**Before After**

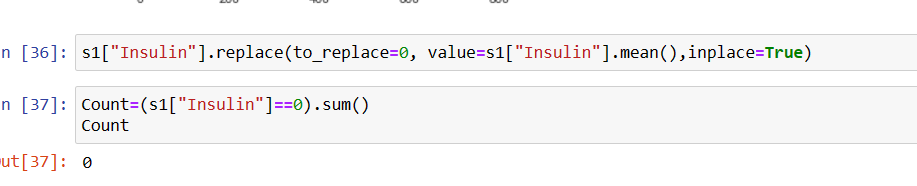
****

**Case for Variable Insulin**

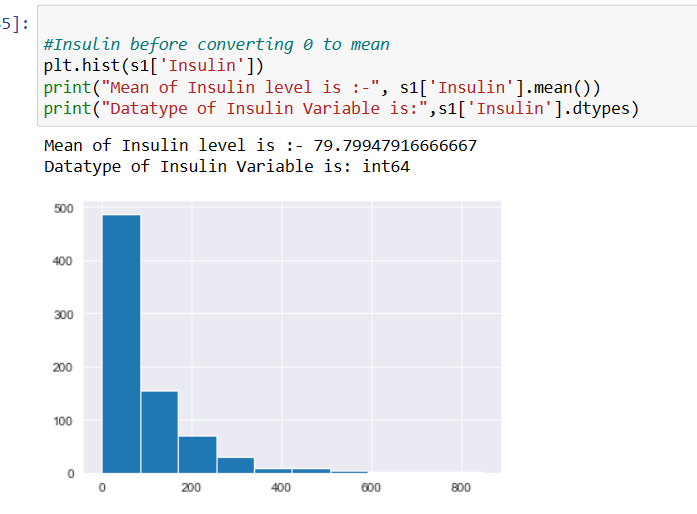
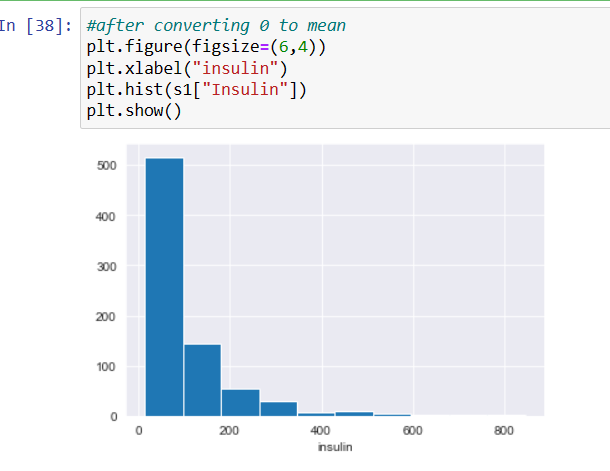
**Before treating missing value.**



**Handling the missing value.**

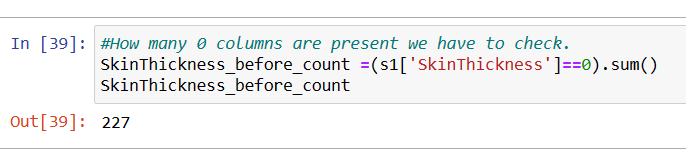


If, we compare the Histogram before & after the missing value treatment, it displays like this:

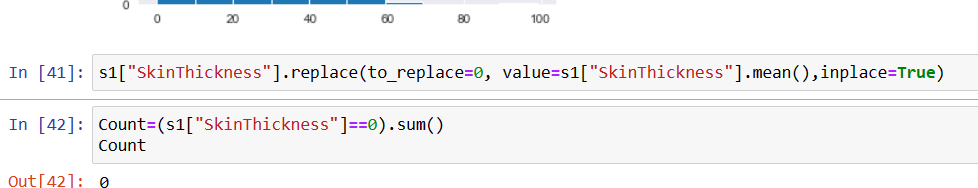
**Before**  **After**

**Case for Variable Skin thickness**

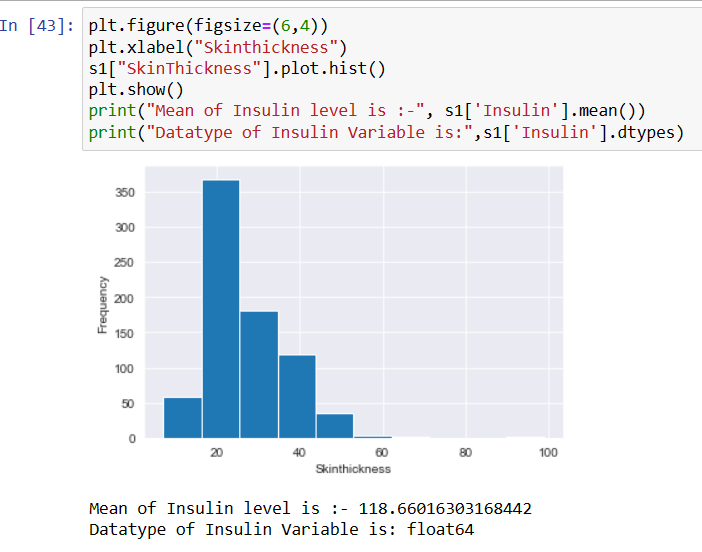
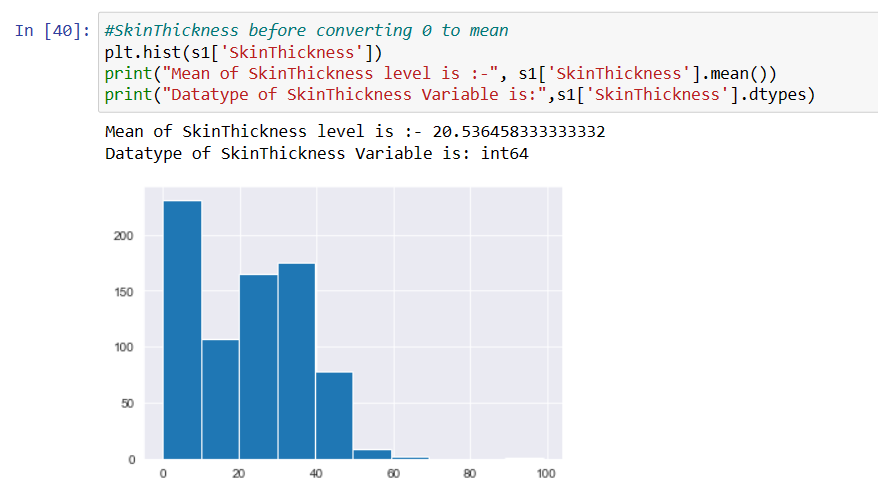
**Before treating missing value.**



**Handling the missing value.**

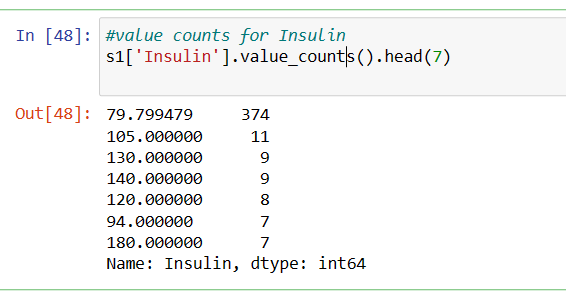


If, we compare the Histogram before & after the missing value treatment, it shows like this:

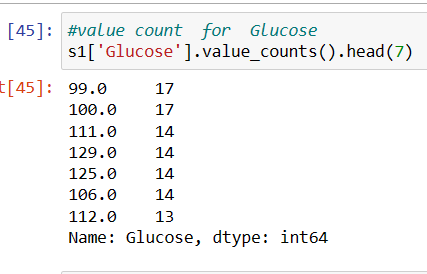
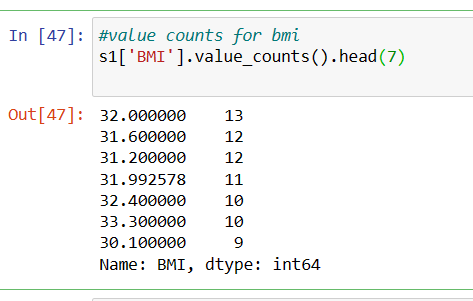
**Before**  **After**

**Create a count (frequency) plot describing the data types & the count of variables**

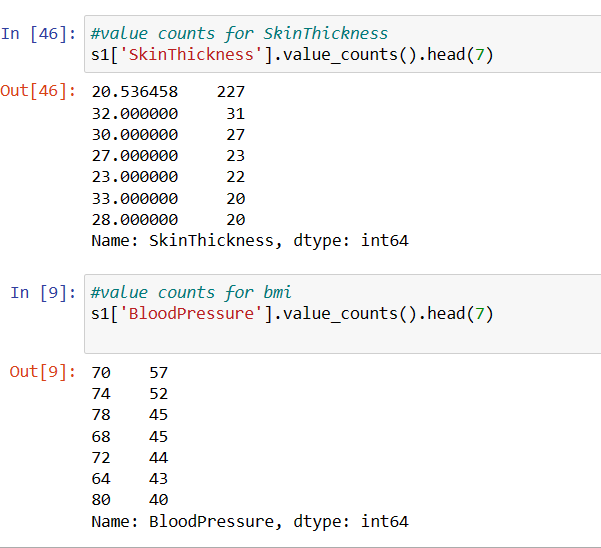
* **Value count for insulin**



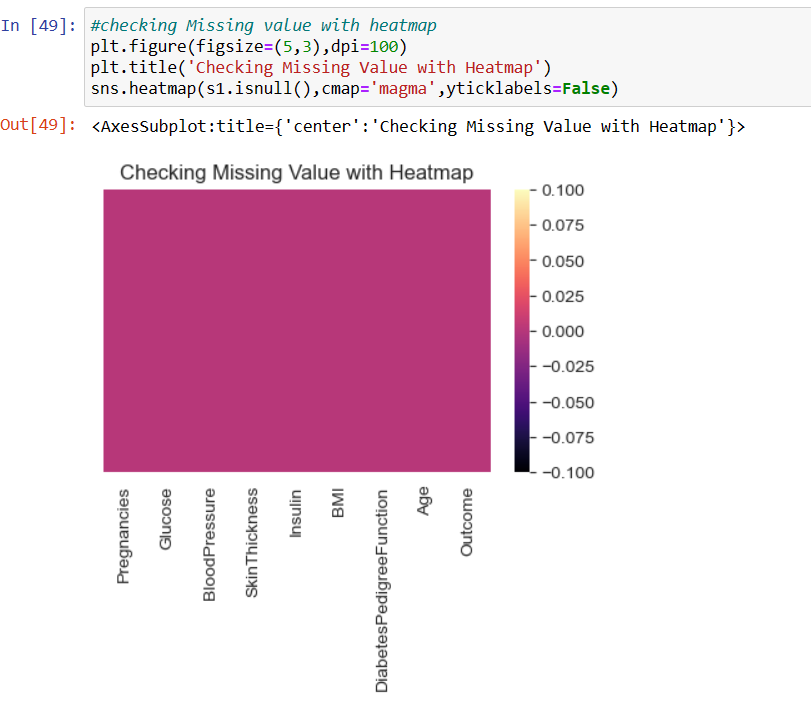
**Value count for glucose Value count for BMI**



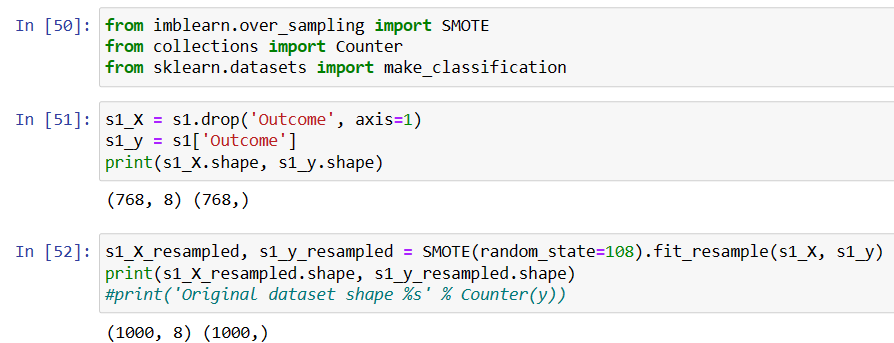
**Count table for value count of Skin thickness and Blood pressure**

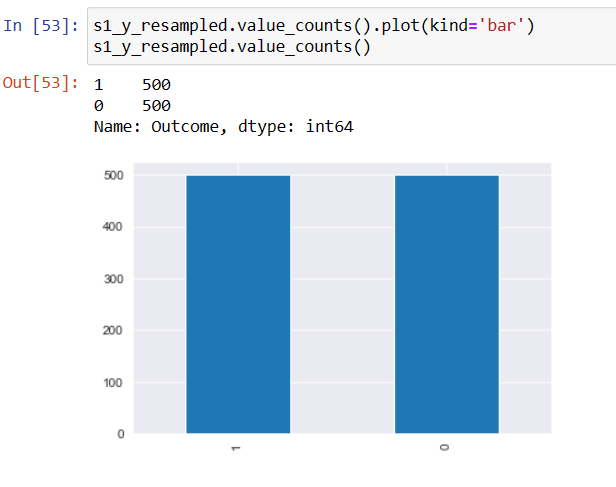


**Checking Missing Values with Heat-Map**

Plotting the count of outcomes by their values

**Describe the finding**

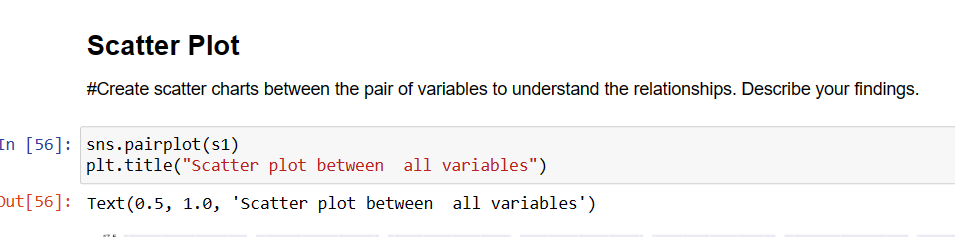
Since classes in Outcome are a little skewed, we will generate new samples using SMOTE (Synthetic Minority Oversampling Technique) for class '1', which is under represented in our data. We will use SMOTE out of many other techniques available because it generates new samples by interpolation and doesn't duplicate the data.****

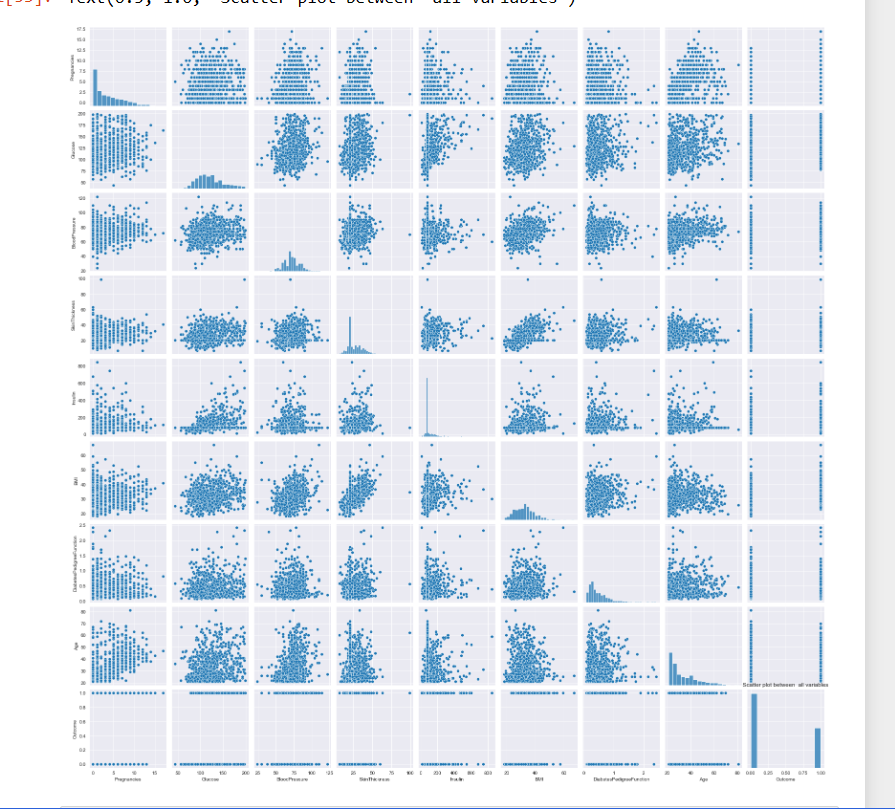
****

**Scatter plot**

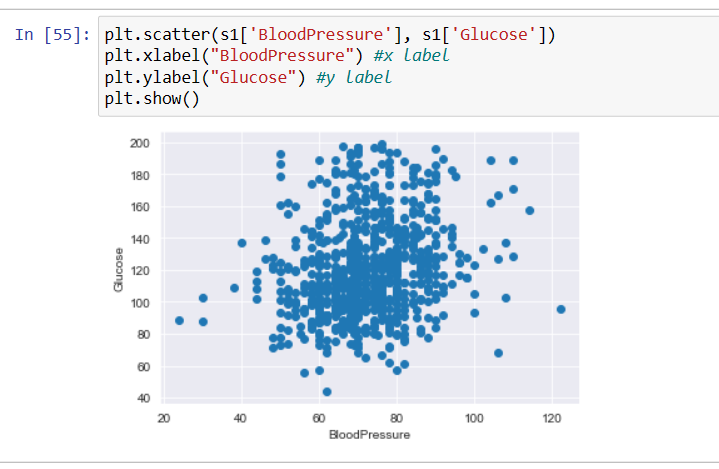
Create scatter charts between the pair of variables to understand the relationships. Describe

your findings.

****

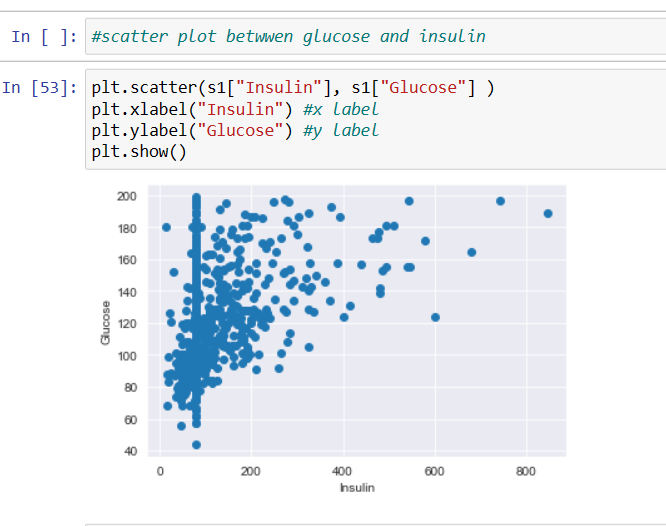
**Describe the observation**

We can see from scatter plot that there is no strong multicolinearity among features, but between skin thickness and BMI, Pregnancies and age it looks like there is small chance of positive correlation. Also create scatter plot to identify the co relation between variables observation.

**Scatter plot with Glucose and Blood-Pressure**

So, from this observation we can see that the positive (+) co-relation is present between Blood pressure & Glucose.

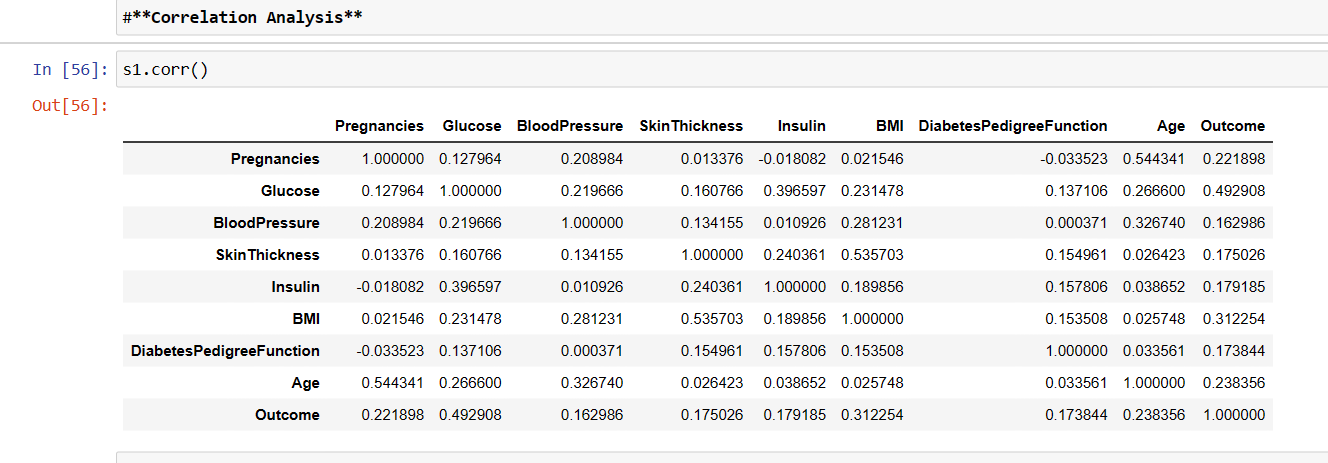
**Scatter plot with insulin and glucose**



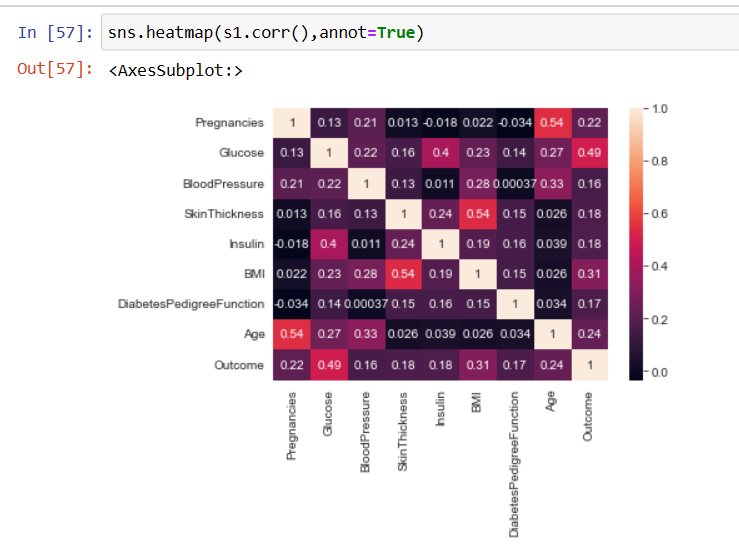
**Scatter plot with BMI and Insulin**

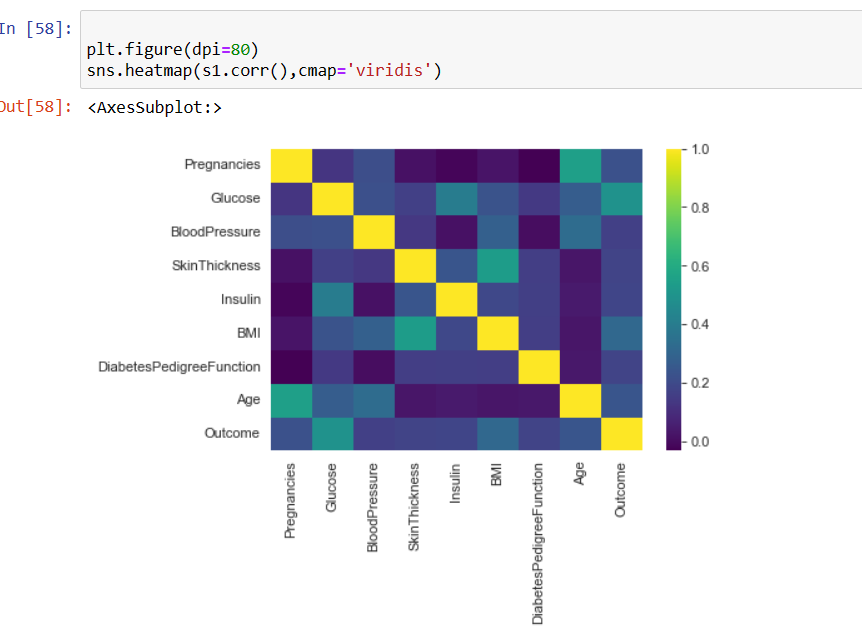
# 

So, from this observation we can see that the positive (+) correlation is present between BMI & Insulin.



**Perform Co-relation Analysis with Heat Map**



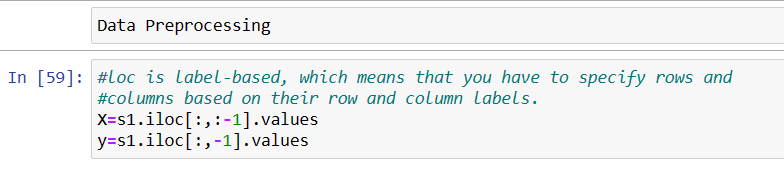


**Project Task: Week 2**

**Data Modelling**

1. Devise strategies for model building. It is important to decide the right validation framework. Express your thought process.
2. Apply an appropriate classification algorithm to build a model.
3. Compare various models with the results from KNN algorithm.
4. Create a classification report by analysing sensitivity, specificity, AUC (ROC curve), etc. Please be descriptive to explain what values of this parameter you have used.

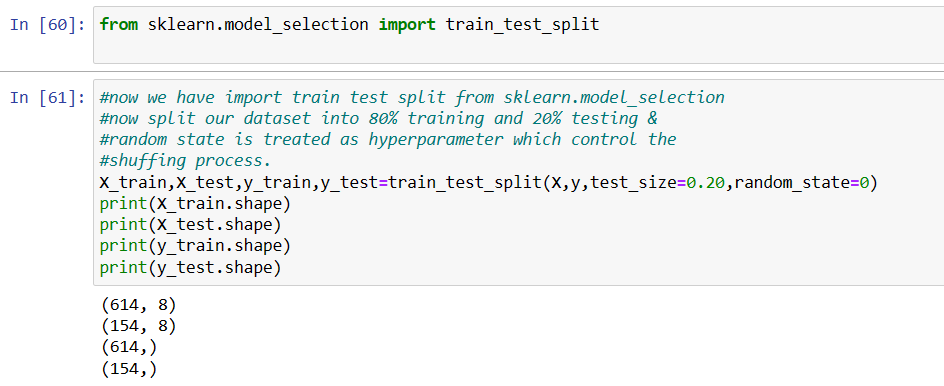
**Step -1:**

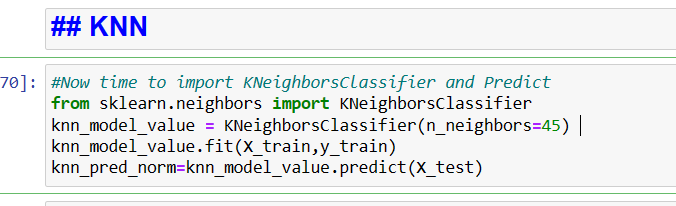


**Step-2**:

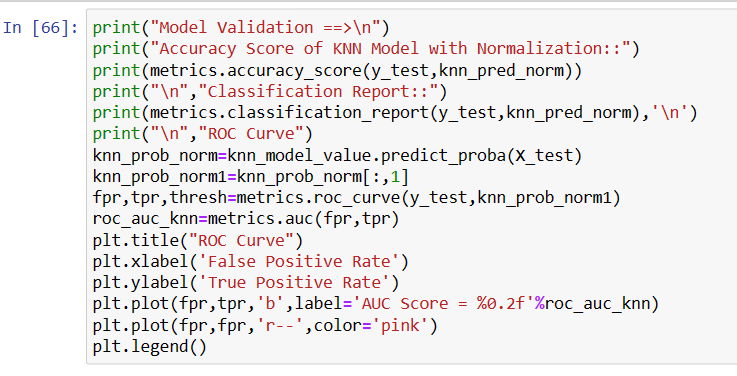
(Splitting the data-set into training and testing part ) and ( To identify how many rows and columns are present in X\_train, X\_test, y train and y\_test.

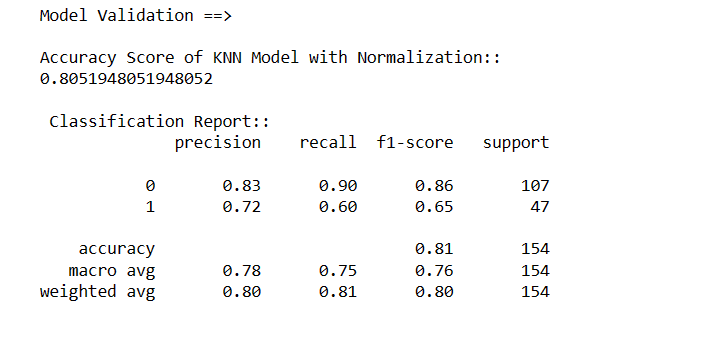
now we have to import train test split from sklearn. model\_ selection. now split our dataset into 80% training and 20% testing & random state is treated as hyper parameter which control the shuffling process. With random \_state=0, we get the same train and test sets across different executions.

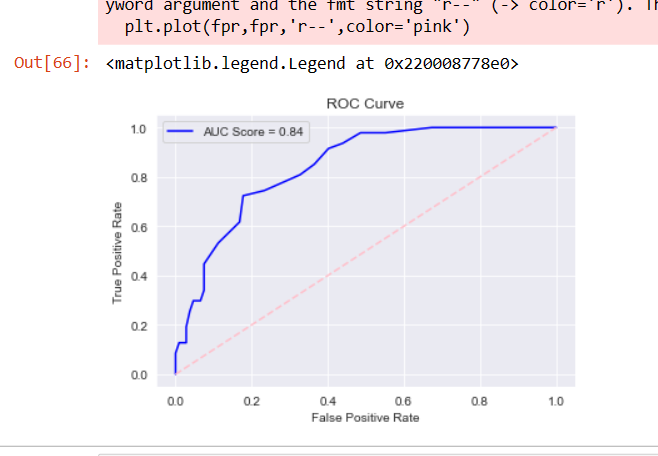
**Now Starting Different Model Building Process (KNN)**



K-NN algorithm stores all the available data and classifies a new data point based on the similarity.

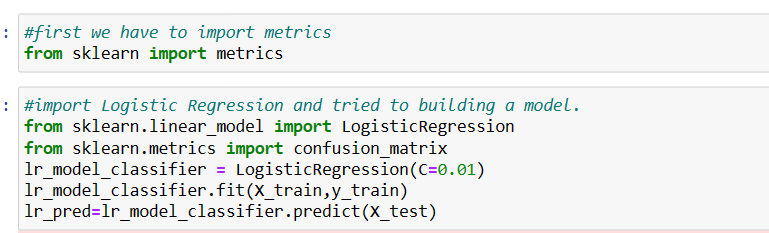


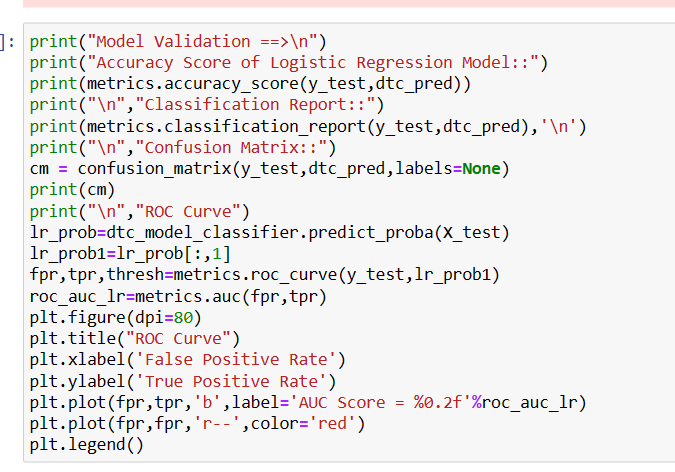


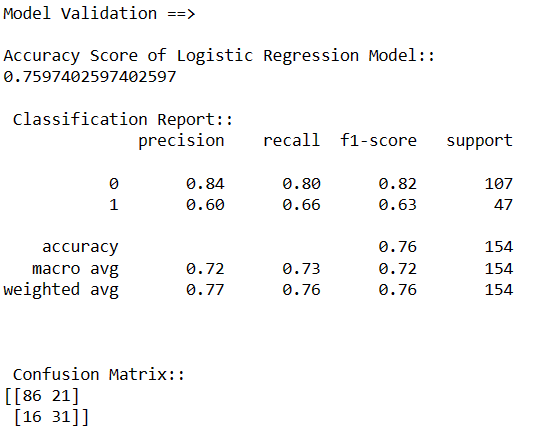


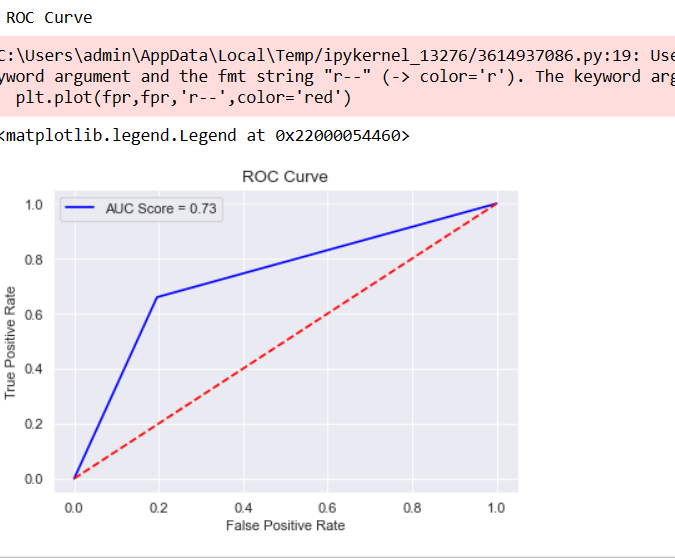
SO, from KNN standard scaling we got the Model Accuracy is 0.81 and from the classification report we got its precision, recall, f1-scoare and support value. With Addition its ROC Curve AUC score is 84%. That is good.

**Now Starting Different Model Building Process (LOGSTIC REGRESSION)**

 Logistic regression is an example of supervised learning. It is used to calculate or predict the probability of a binary (yes/no) event occurring.

**From the above code we got output for Model Accuracy, classification report, confusion Matrix & ROC curve**

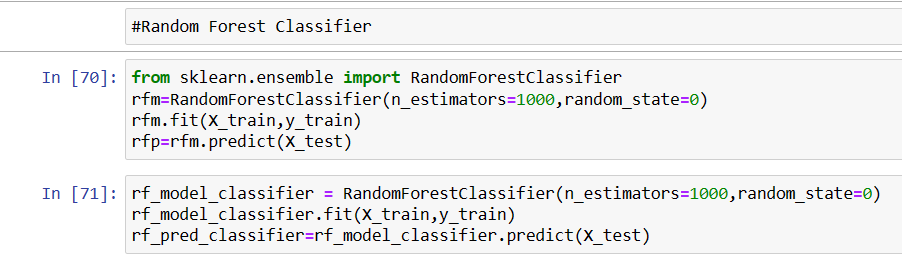
****

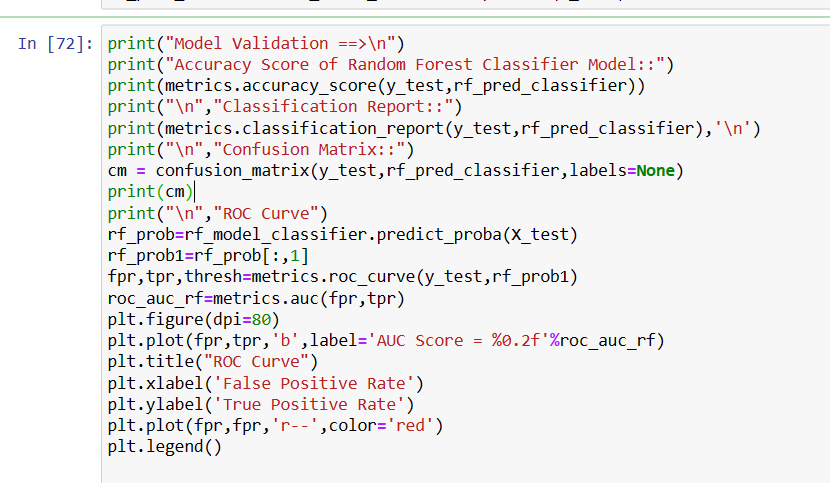


**Observation**: Accuracy, recall, f1-score of KNN is better than Logistic Regression.

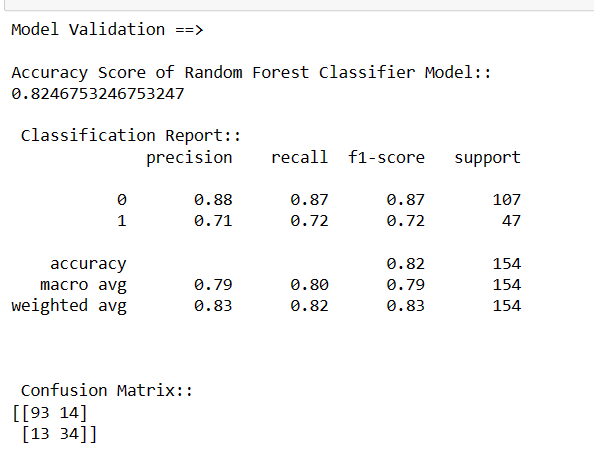
**Now Starting Different Model Building Process (Random Forest)**

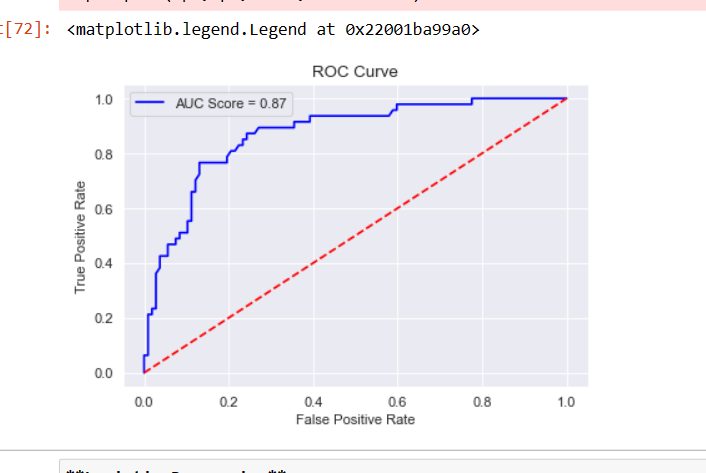
A Random forest is a meta-estimator that fits a number of decision tree classifiers on various subsamples of the dataset and uses averaging to improve predictive accuracy and control over fitting. Model Creation:





**From the above code we got output for Model Accuracy, classification report, confusion Matrix & ROC curve**

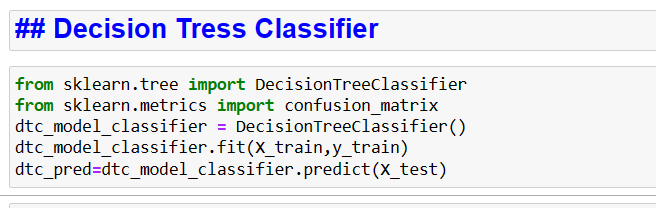


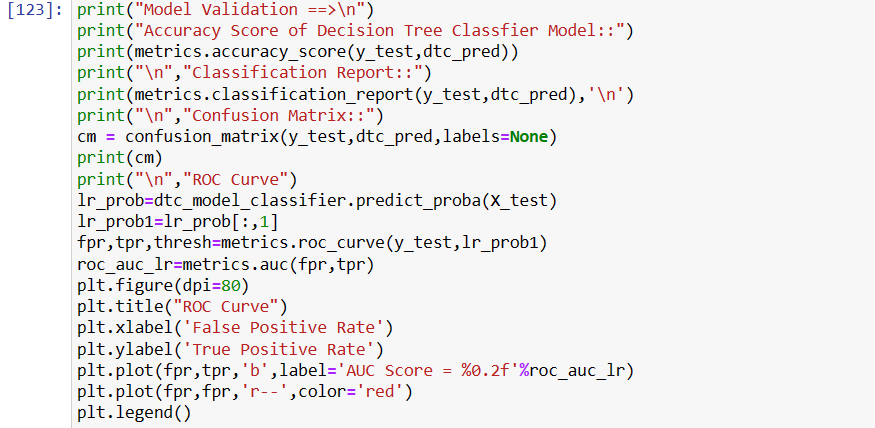


SO, from random Forest scaling we got the Model Accuracy is 0.82 and from the classification report we got its precision, recall, f1-scoare and support value. With Addition its ROC Curve AUC score is 87%. That is good.

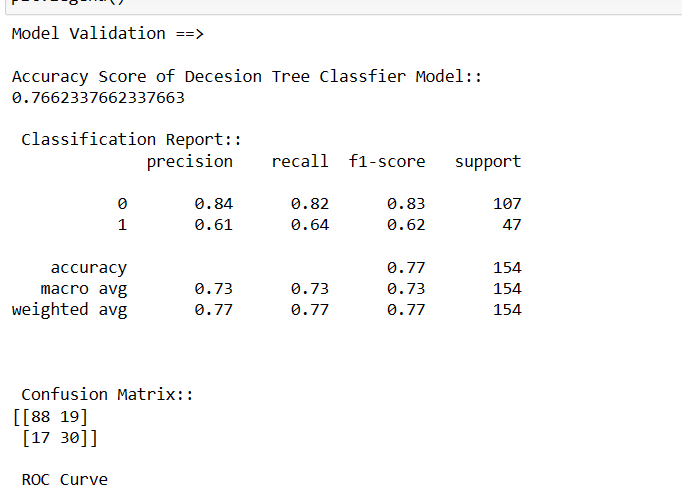
**Now Starting Different Model Building Process (Decision Tree Classifier)**

Decision Tree is a **Supervised learning technique**that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where**internal nodes represent the features of a dataset, branches represent the decision rules** and **each leaf node represents the outcome.**

****

****

**From the above code we got output for Model Accuracy, classification report, confusion Matrix & ROC curve**:

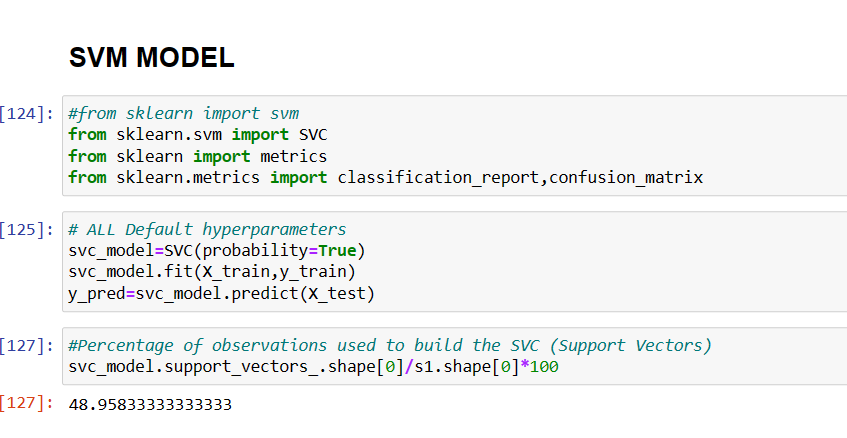




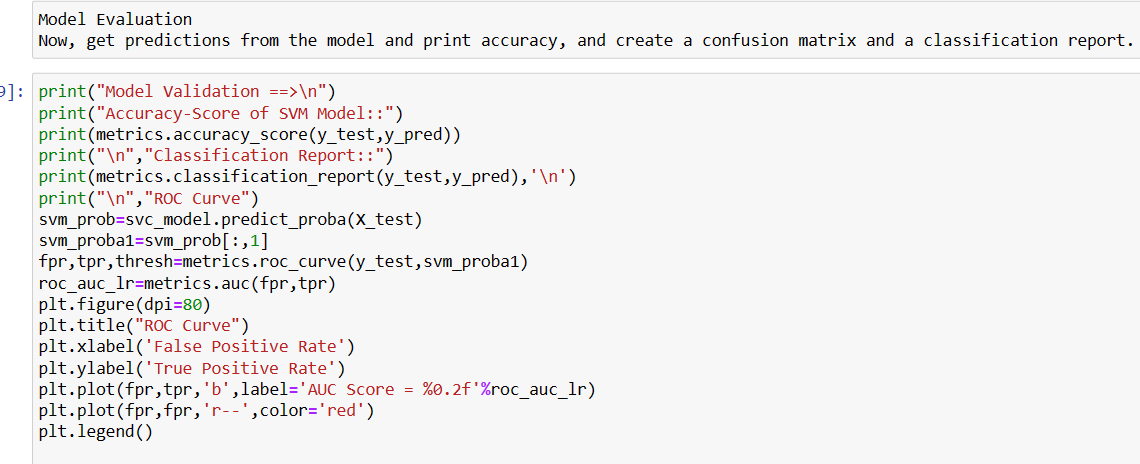
**Observation**: Accuracy, recall, f1-score of KNN is better than Decision Tree.

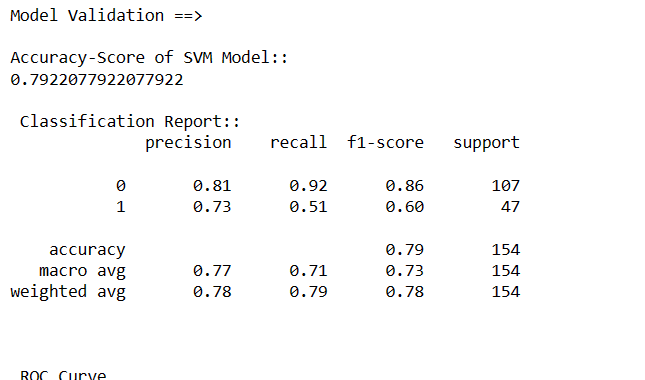
**Now Starting Different Model Building Process (SVM)**

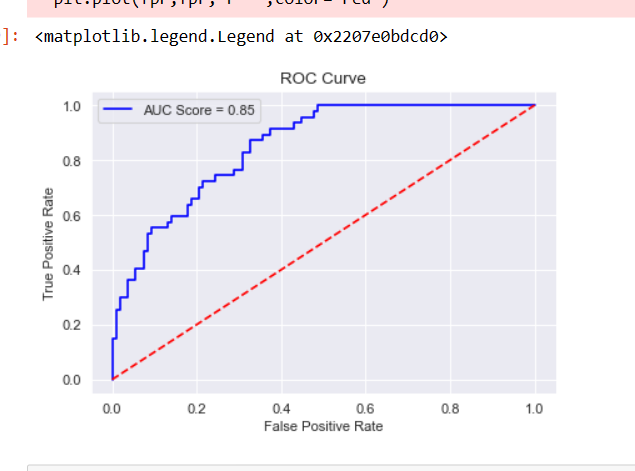
SVM is also a Supervised Machine Learning Algorithm that uses a set of rules to make decisions, similarly to how humans make decisions. One way to think of a Machine Learning classification algorithm is that it is built to make decisions.



**From the above code we got output for Model Accuracy, classification report, confusion Matrix & ROC curve**:



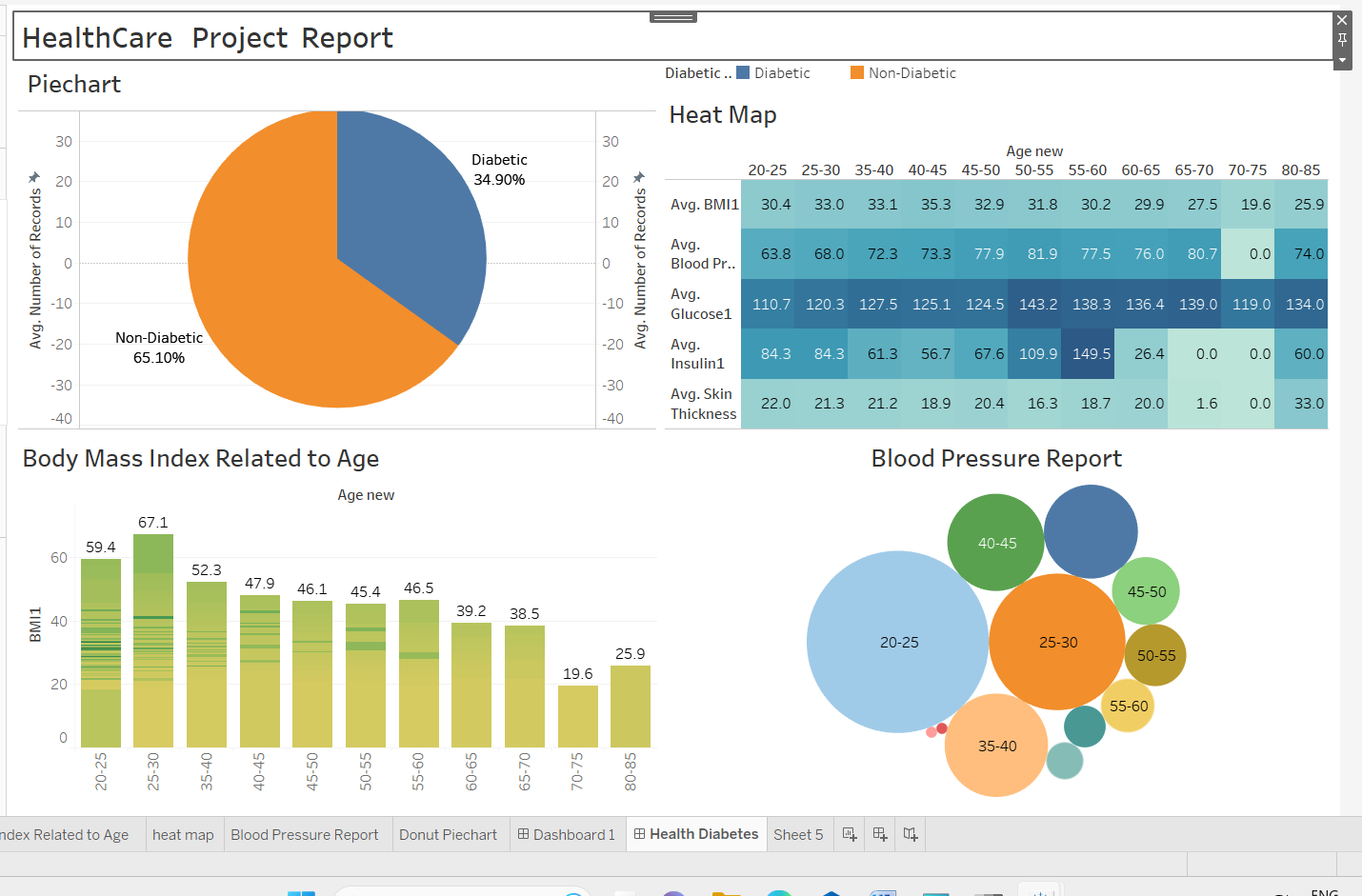


****

**Observation:**

We can see the SVM Classifier is the best among all. Accuracy of SVM is better than KNN. I also consider it to be the best. F1-score and recall (the fraction of relevant instances that were retrieved) is far better than other models. With Addition its ROC Curve AUC score is 85%. That is good.

**Methodology for Data Reporting**



WebLink:-

[health care diabetes | Tableau Public](https://public.tableau.com/app/profile/baman.sravanthi/viz/healthcarediabetes_16891399317590/HealthDiabetes?publish=yes)

https://public.tableau.com/app/profile/baman.sravanthi/viz/healthcarediabetes\_16891399317590/HealthDiabetes?publish=yes