IMT2017028 - U Khalid
IMT2017504 - Anurag P
IMT2017524 - K Sailesh

Machine Learning Assignment 1

PROBLEM STATEMENT

Build a model to accurately classify/ predict if a person has diabetes or not.

OVERVIEW

- **Technique used to build the model** : Logistic regression
- **Data set used to build the model** : PIMA_Indian_Diabetes
- **Parameters/ Features present in the data set** :
    - Pregnancies : Number of times the person is pregnant
    - Glucose : Plasma glucose concentration
    - Blood Pressure : Diastolic Blood Pressure
    - Skin Thickness : Triceps skinfold thickness
    - Insulin : 2hr serum insulin
    - BMI : Body Mass Index (Weight / Height^2)
    - Age : Age of the person
    - Diabetes Pedigree Function : A function that gives an idea of hereditary risk one might have diabetes.

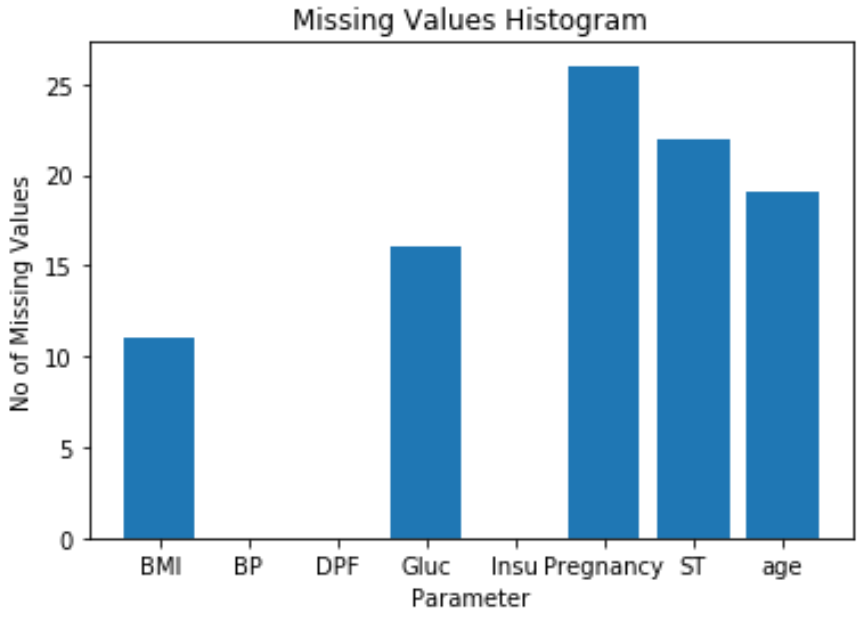
EXPLORATORY DATA ANALYSIS

Exploratory Data Analysis is an approach to analyze the data and summarize the characteristics of the given data set.

This is the description of the initial data we have.

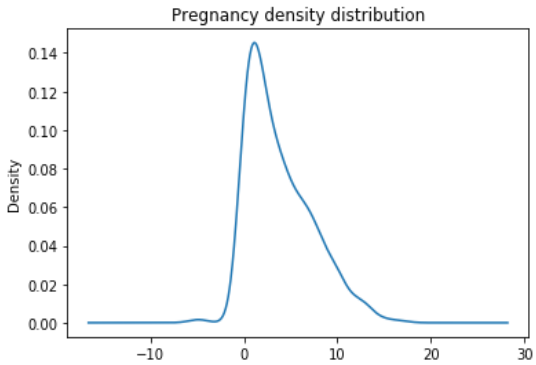
	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
count	742.000000	752.000000	768.000000	746.000000	768.000000	757.000000	768.000000	749.000000	768.000000
mean	3.866601	119.966097	68.886078	20.309879	79.799479	31.711151	0.471876	33.761336	0.348958
std	3.479971	32.367659	19.427448	15.974523	115.244002	8.544789	0.331329	12.297409	0.476951
min	-5.412815	0.000000	-3.496455	-11.945520	0.000000	-16.288921	0.078000	21.000000	0.000000
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.100000	0.243750	24.000000	0.000000
50%	3.000000	116.000000	72.000000	23.000000	30.500000	32.000000	0.372500	29.000000	0.000000
75%	6.000000	140.000000	80.000000	32.000000	127.250000	36.500000	0.626250	41.000000	1.000000
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	2.420000	81.000000	1.000000

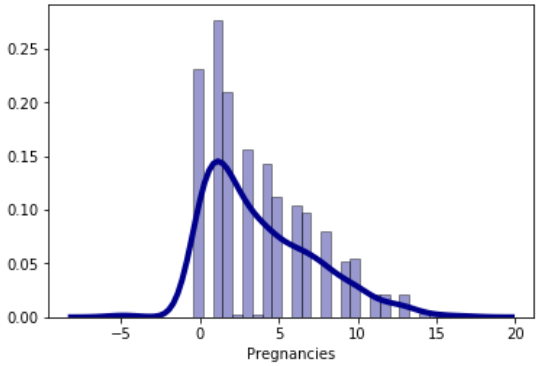
This histogram here represents the no. of missing data points in each column of the data set.



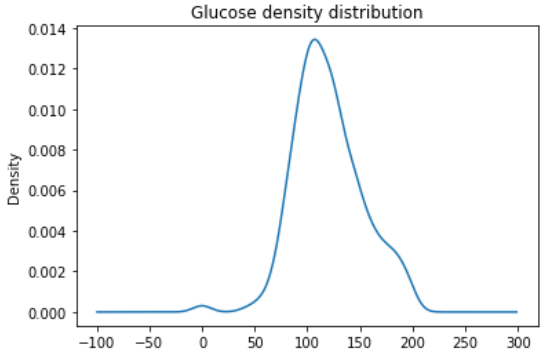
Distribution of each input column/ parameter in the data set:

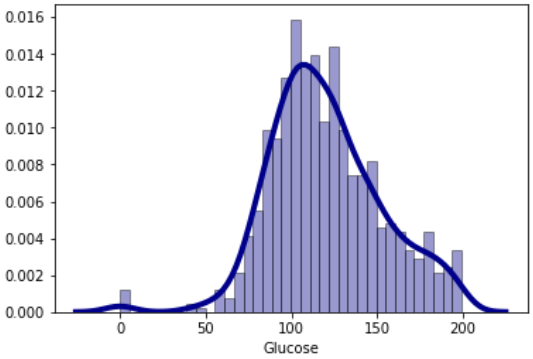
A. Pregnancies density distribution and density histogram



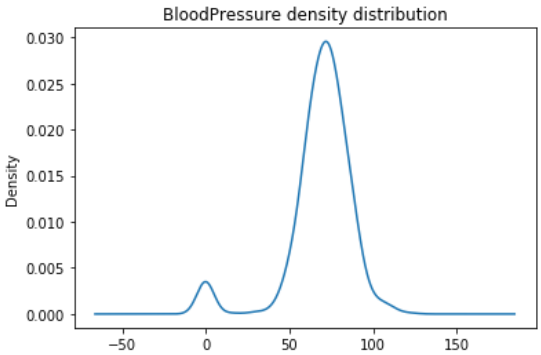


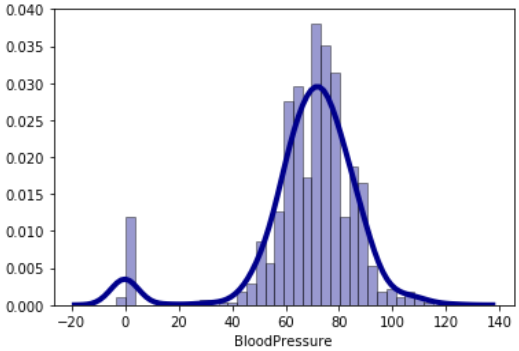
B. Glucose density distribution and density histogram



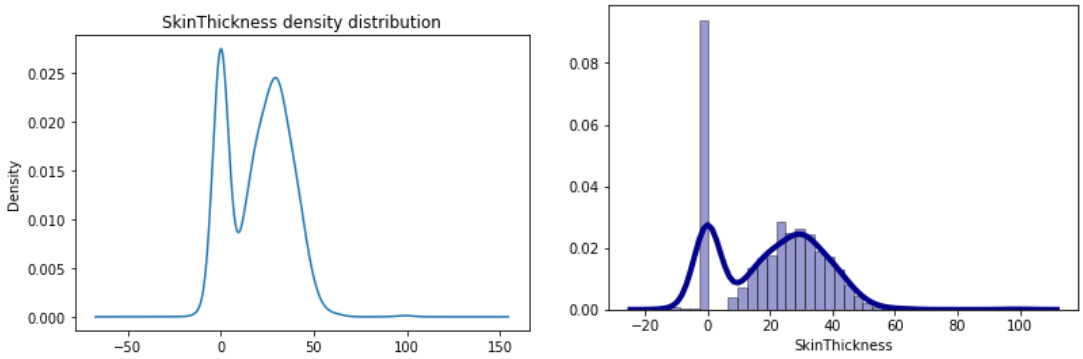


C. BloodPressure

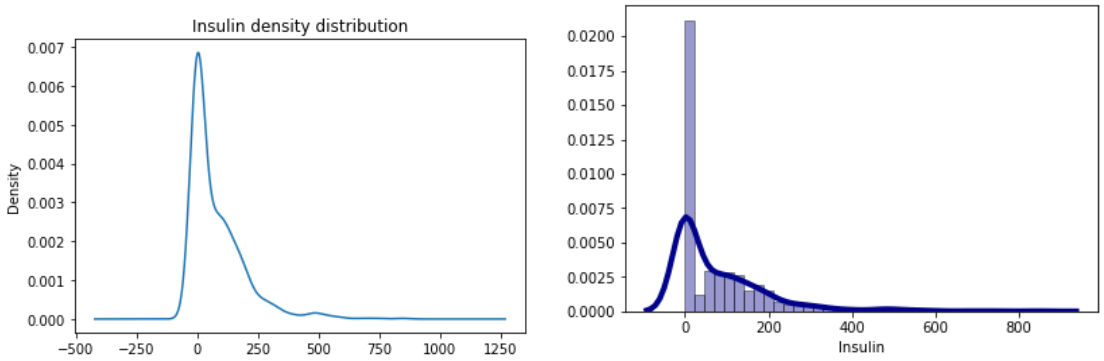




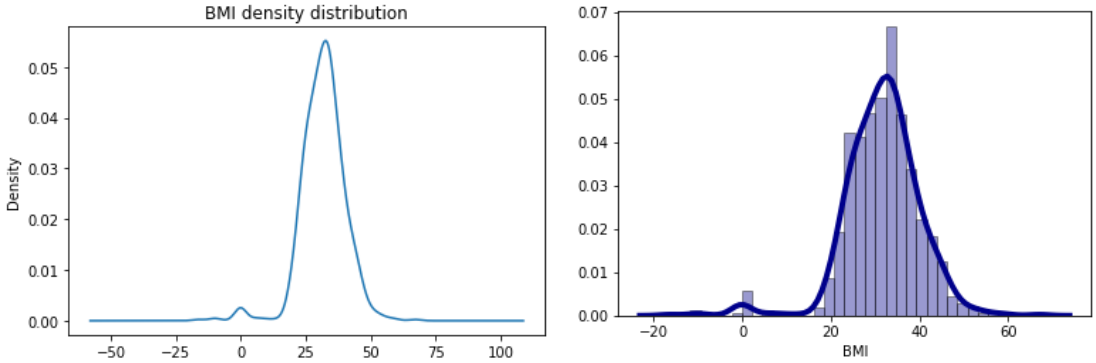
D. SkinThickness



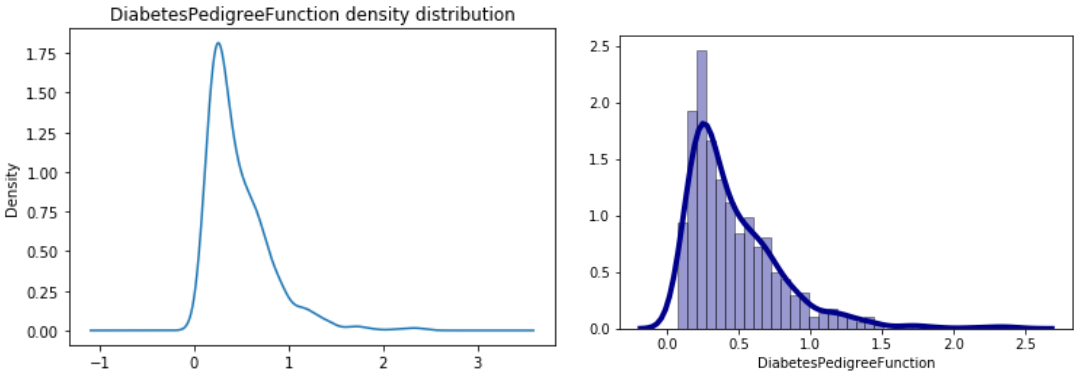
E. Insulin



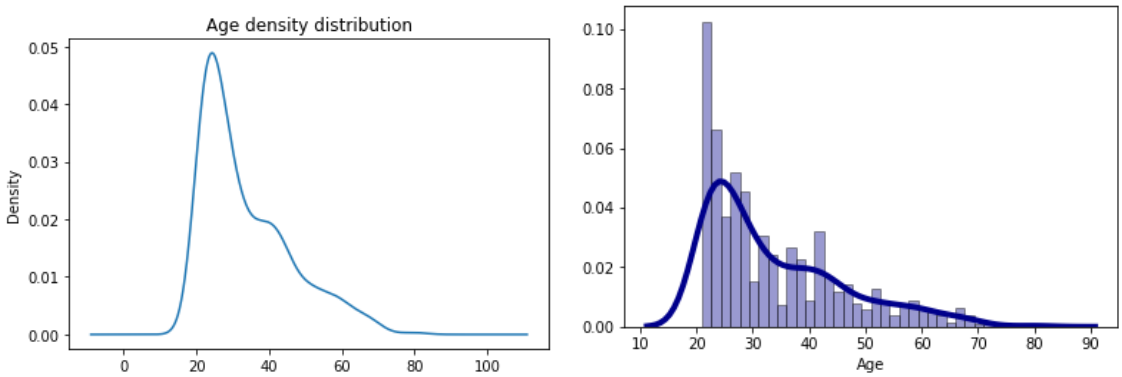
F. BMI



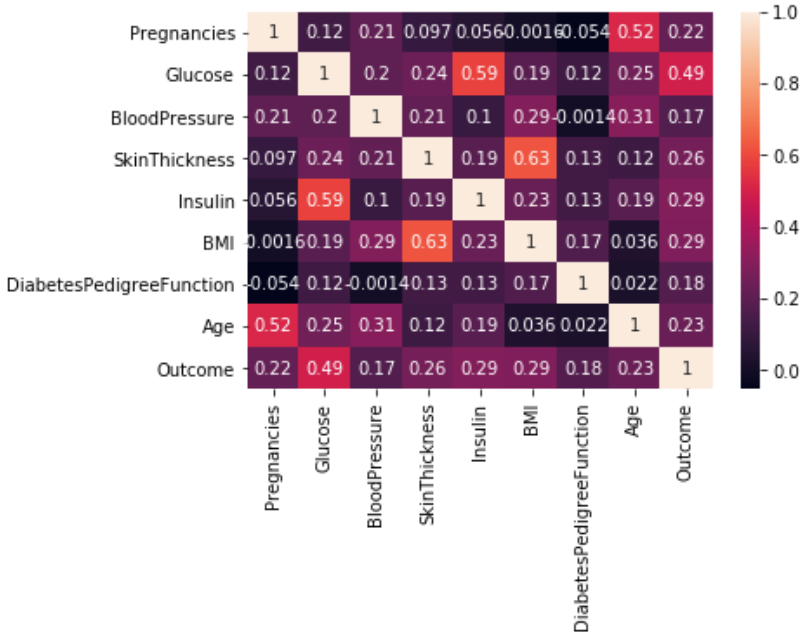
G. DiabetesPedigreeFunction



H. Age



The correlation matrix for the data in the data set is :



(A correlation matrix is a table showing correlation coefficients between variables here input parameters.)

DATA PREPROCESSING

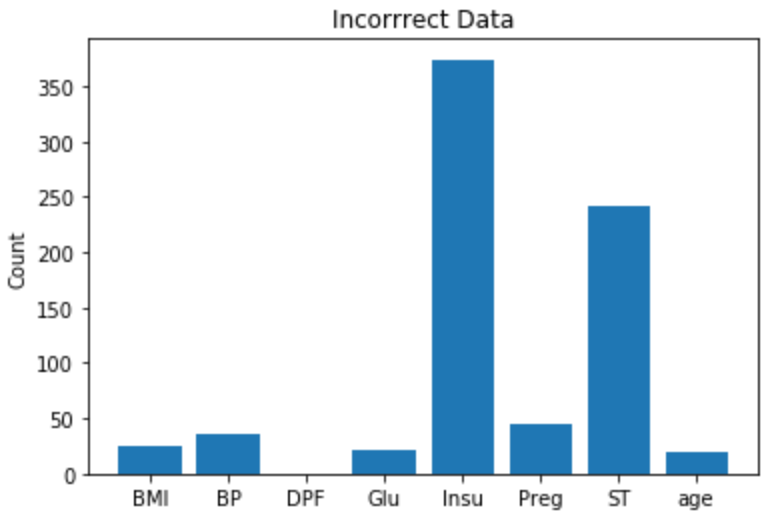
Data Preprocessing is the process in which false/ noisy data is detected and removed to prevent training the model with wrong data. This step is to be essentially performed on any data set before using to get good results.

In this data set there are negative and non integer values for Pregnancies which is not possible. These are made null

Similarly there are negative values for SkinThickness,Pregnancies,Glucose,BloodPressure,BMI and Insulin which are also made null. The incorrect data is first removed(Becomes missing data which is handled in the next step).

Insulin and SkinThickness contain a lot of zeros. This data is faulty as the general ranges for these values suggest they would take different values. These values are also made null

This graph shows the amount of incorrect + missing data in each of the parameter.



Data description after removing the noisy/missing data.

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
count	668.000000	712.000000	712.000000	506.000000	381.000000	695.000000	712.000000	694.000000	712.000000
mean	3.836826	120.963081	72.159634	28.899778	155.241470	32.434736	0.477538	33.724529	0.345506
std	3.361573	31.040408	12.573856	10.559644	119.326264	7.098789	0.336771	12.059196	0.475867
min	0.000000	42.974768	15.372031	7.000000	14.000000	6.699051	0.078000	21.000000	0.000000
25%	1.000000	99.000000	64.000000	21.000000	76.000000	27.500000	0.245000	24.000000	0.000000
50%	3.000000	116.500000	72.000000	29.000000	125.000000	32.400000	0.380000	29.000000	0.000000
75%	6.000000	140.118555	80.000000	36.000000	190.000000	36.600000	0.629500	41.000000	1.000000
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	2.420000	70.000000	1.000000

MISSING DATA HANDLING

Missing data is a problem which has to be tackled carefully. There are different methods to handle missing data like:

- Removing the data points with missing data if we dont get data starved .
- Imputing the missing data of a parameter with mean/ mode of the distribution of the same parameter if it is not correlated with other parameters.
- If there is a significant amount of correlation between different parameters the missing data in a parameter can be predicted using the other parameters.

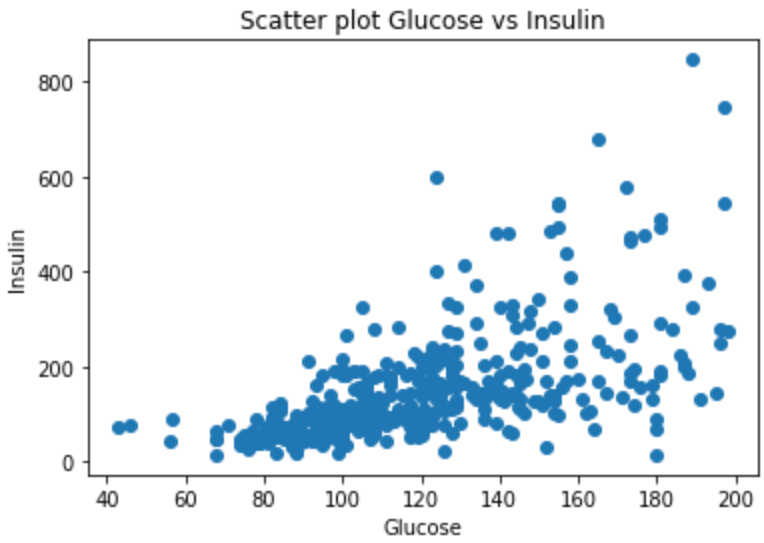
The missing data points in Glucose and BloodPressure are dropped. This can be done as the medical data/parameters can take any value (Medical Problems arise for this same reason) and predicting them or filling with mean/mode could tamper the correctness of the data.

Parameters like Age, BMI and Pregnancies have very few missing data points. These can be imputed by replacing the missing points in Age, BMI and Pregnancies are replaced by the mode. The can be done as the BMI, Pregnancies are, in general, almost the same for everyone and also filling a few missing values with mode does not change the distribution of data (People with lower height would tend to have lower weight and people who are taller would be heavier. BMI is compensated in this way).

There are a lot of points missing in Insulin and SkinThickness. To impute the missing data linear regression is used. The columns of Insulin and SkinThickness are taken as target columns. The correlation table tells us about the similarity between two parameters. The parameters which have maximum correlation are taken as input parameters to predict the missing values. Filling with mean/mode is not preferable as we cannot find the distribution of the data correctly with large missing data rows. Also we cannot just drop them as we would become data starved.

Predictions are made in the following ways:

- Insulin has 0.59 correlation with Glucose thus it is used as input feature of linear regression to predict the missing Insulin values. Also the values are nearly linear.



- SkinThickness has 0.63 correlation with BMI thus it is used as input feature of linear regression to predict the missing SkinThickness values.

The data description after imputing the missing values is as follows:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
count	712.000000	712.000000	712.000000	712.000000	712.000000	712.000000	712.000000	712.000000	712.000000
mean	3.836826	120.963081	72.159634	28.501143	153.842420	32.414805	0.477538	33.428122	0.345506
std	3.255897	31.040408	12.573856	9.573556	99.072382	7.014569	0.336771	12.047187	0.475867
min	0.000000	42.974768	15.372031	4.453716	-18.121266	6.699051	0.078000	21.000000	0.000000
25%	1.000000	99.000000	64.000000	22.000000	89.128299	27.600000	0.245000	24.000000	0.000000
50%	3.000000	116.500000	72.000000	28.000000	131.581251	32.050000	0.380000	29.000000	0.000000
75%	6.000000	140.118555	80.000000	34.444238	190.250000	36.500000	0.629500	41.000000	1.000000
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	2.420000	70.000000	1.000000

On comparing with the data before imputing, Not much difference is seen in the distribution which is good.

FEATURE EXTRACTION

PCA (Principal Component Analysis) has been used to extract features here..

In the data there are 8 input parameter columns where as the data points/ training example are around 720. Having many features with lesser data may result in overfitting. Overfitting is the production of an analysis that corresponds too closely or exactly to a particular set of data, and may therefore fail to fit additional data or predict future observations reliably

To avoid that, the number of features can be reduced or more data should be collected. Second option is not always feasible (In this case a data set is already given). So going forward with option one is always a good idea.

MODEL BUILDING

Logistic Regression has been used to perform the data prediction (In this case we are doing binary classification either as 0, 1)

In order to train the model better the variations in data have to be covered properly in the training set. This can be done by making use of the function datasplit which divides the data into training and test sets to cover enough data points of both classes (0 and 1) in both the sets. 75% of the data has been used to train the model. The remaining 25% has been used to test the model.

First the model is trained and a hypothesis is created using training data. To check if the model built is good enough it is validated with test set.

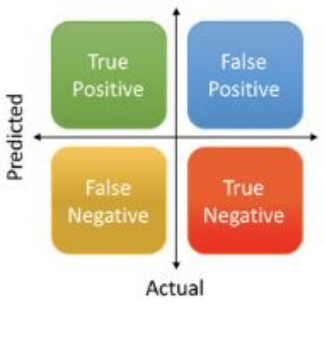
The metrics being used to evaluate the model built are precision, recall and accuracy.

Precision and Recall

$$\text{Precision} = \frac{\text{True Positive}}{\text{Actual Results}} \quad \text{or} \quad \frac{\text{True Positive}}{\text{True Positive + False Positive}}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{Predicted Results}} \quad \text{or} \quad \frac{\text{True Positive}}{\text{True Positive + False Negative}}$$

$$\text{Accuracy} = \frac{\text{True Positive + True Negative}}{\text{Total}}$$



Where

- True positive is number of examples belonging to positive class which have been predicted correctly
- True negative is number of examples belonging to negative class which have been predicted correctly
- False positive is the number of examples belonging to negative class which have been predicted as belonging to the positive class
- False negative is the number of examples belonging to positive class which have been predicted as belonging to negative class

Accuracy of the model when PCA dimensionality is varied:

PCA Dimensionality	Accuracy (%)	Precision 0 (%)	Precision 1 (%)	Recall 0 (%)	Recall 1 (%)
1	76	78	70	87	55
2	76	78	70	87	55
3	75	78	68	87	54
4	76	79	70	88	55
5	76	79	69	87	56
6	76	79	70	87	56
7	78	80	71	87	59
8	75	78	67	86	55

Thus choosing PCA Dimensions to be 2 or 4 would help build a good model to predict whether a person has diabetes or not.

Without using dataSplit function accuracy was close to 80% was obtained but using dataSplit helps to handle any dataset which are far more diverse than the given one.