University for the Creative Arts

BERLIN SCHOOL OF BUSINESS & INNOVATION

Assignment Title: PCOS prediction using Machine learning models

Program title: Fundamentals of Data Analytics_ MSc. Data analytics

Name: Surya Nagesh Babu

Student ID: Q1050807

Year: 2024

TABLE OF CONTENTS

Statement of compliance with academic ethics and the avoidance of plagiarism

I honestly declare that this dissertation is entirely my own work and none of its part has been copied from printed or electronic sources, translated from foreign sources, and reproduced from essays of other researchers or students. Wherever I have been based on ideas or other people texts I clearly declare it through the good use of references following academic ethics.

(In the case that is proved that part of the essay does not constitute an original work, but a copy of an already published essay or from another source, the student will be expelled permanently from the postgraduate program).

Name and Surname (Capital letters):

SURYA NAGESH BABU

Date: **30/01/2024**

1. INTRODUCTION

1.1. Predictive analytics & Machine learning for decision-makers

With the evolution that the process of decision-making has undergone, businesses today can exploit even the rarest data points which eventually accelerates the decision-making process. Machine learning and Artificial Intelligence have revolutionized the way people consume data and individuals, and a study by Alagar (2023) says that decision-makers can now leverage the potential of these technologies to scale their business into a more resilient and successful future like never before.

1.2. Business problem and scope of study

Medicine and Science have not been an exception when it comes to fostering these new technologies. In our problem statement of PCOS detection, we will address one such use case where machine learning plays a quintessential role in predicting the presence of this condition in female individuals.

2. PROBLEM FORMULATION AND DATA COLLECTION

2.1. Problem Statement

Polycystic ovary syndrome (PCOS) as stated by (Polycystic Ovary Syndrome (PCOS), 2022) is a disorder occurring in women who are most commonly in their childbearing age in which ovaries produce abnormal levels of male sex hormones namely androgens which otherwise are produced in very small amounts in women. The term polycystic syndrome is termed after its effects on the ovaries, women with PCOS develop small fluid-filled cysts in the ovaries. Women with PCOS are prone to other serious health conditions like insulin resistance eventually resulting in weight gain and obesity etc. Therefore, prior detection of the symptoms or detecting the potential for an individual to develop this condition could result in requisite steps toward prevention.

In this report, we leverage the potential of predictive analytics and machine learning algorithms to analyze and explore the relationship between the different biometric variables, and we will train the machine learning model to learn the hidden patterns and further use them for predicting the presence of PCOS in individuals.

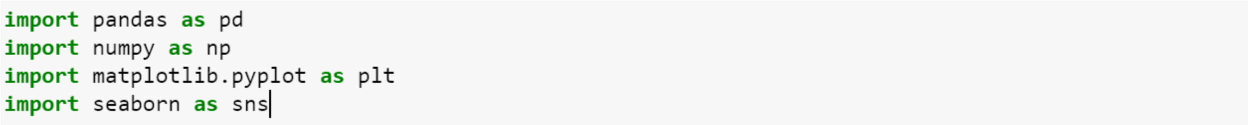
2.2. Data Collection

The dataset that is used as a part of this project comprises data collected from 10 different hospitals across India. The dataset contains information on all the physical and clinical parameters used to detect the presence of Polycystic ovary syndrome in women. This dataset was downloaded directly from the data repository of an online platform called Kaggle.

3. DATA PREPROCESSING

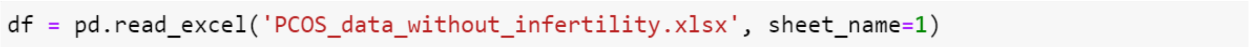
3.1. Pre-requisites

Importing all the necessary libraries



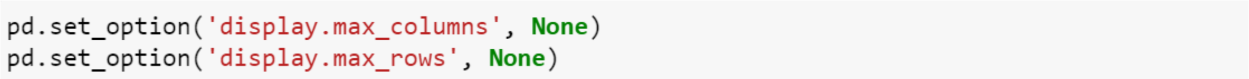
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Importing the dataset



```
df = pd.read_excel('PCOS_data_without_infertility.xlsx', sheet_name=1)
```

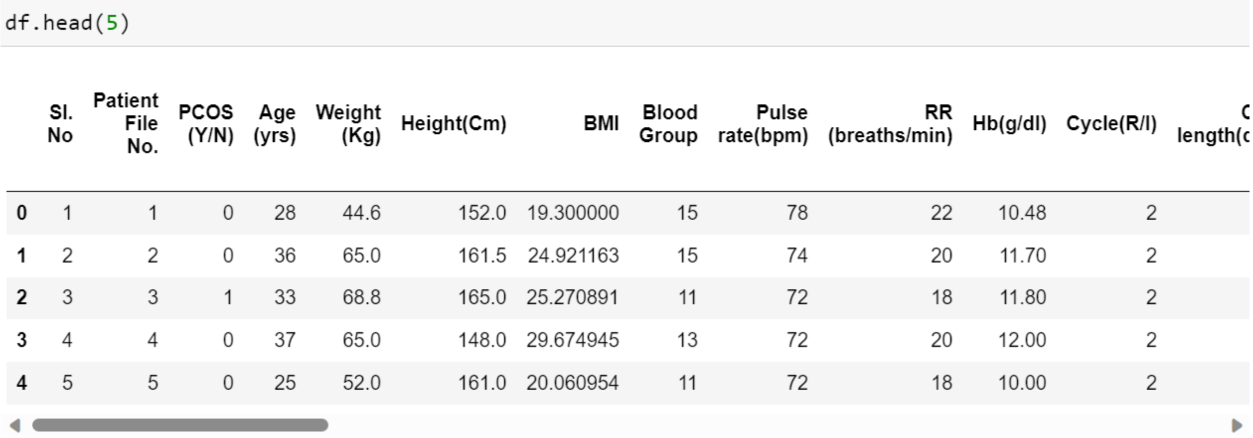
To display all the rows and columns of the dataset



```
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
```

3.2. Preliminary analysis

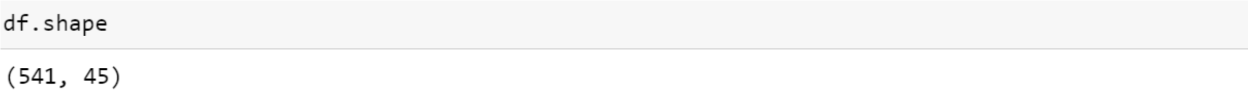
Getting an overview of first 'n' records



```
df.head(5)
```

	Sl. No	Patient File No.	PCOS (Y/N)	Age (yrs)	Weight (Kg)	Height(Cm)	BMI	Blood Group	Pulse rate(bpm)	RR (breaths/min)	Hb(g/dl)	Cycle(R/I)	C length(c
0	1	1	0	28	44.6	152.0	19.300000	15	78	22	10.48	2	
1	2	2	0	36	65.0	161.5	24.921163	15	74	20	11.70	2	
2	3	3	1	33	68.8	165.0	25.270891	11	72	18	11.80	2	
3	4	4	0	37	65.0	148.0	29.674945	13	72	20	12.00	2	
4	5	5	0	25	52.0	161.0	20.060954	11	72	18	10.00	2	

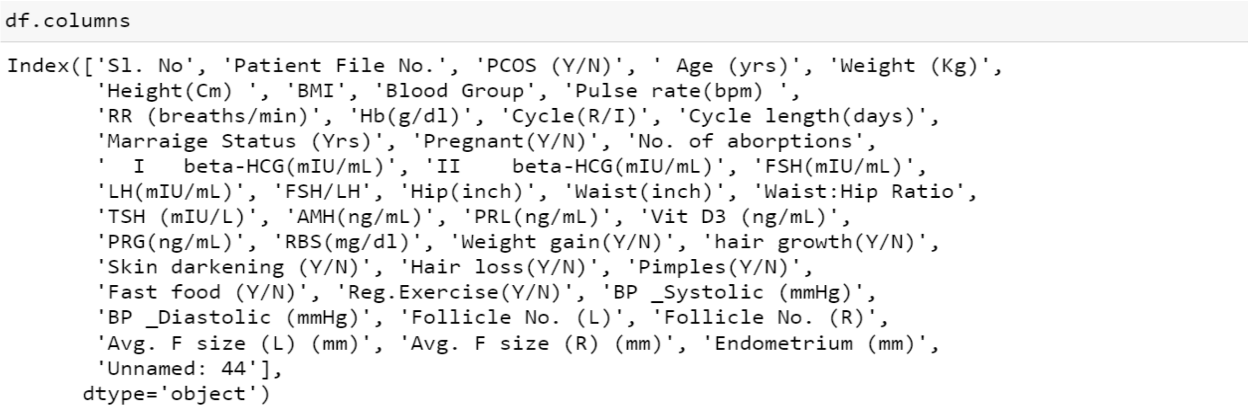
Dimensions check



```
df.shape
```

```
(541, 45)
```

Getting an overview of the columns

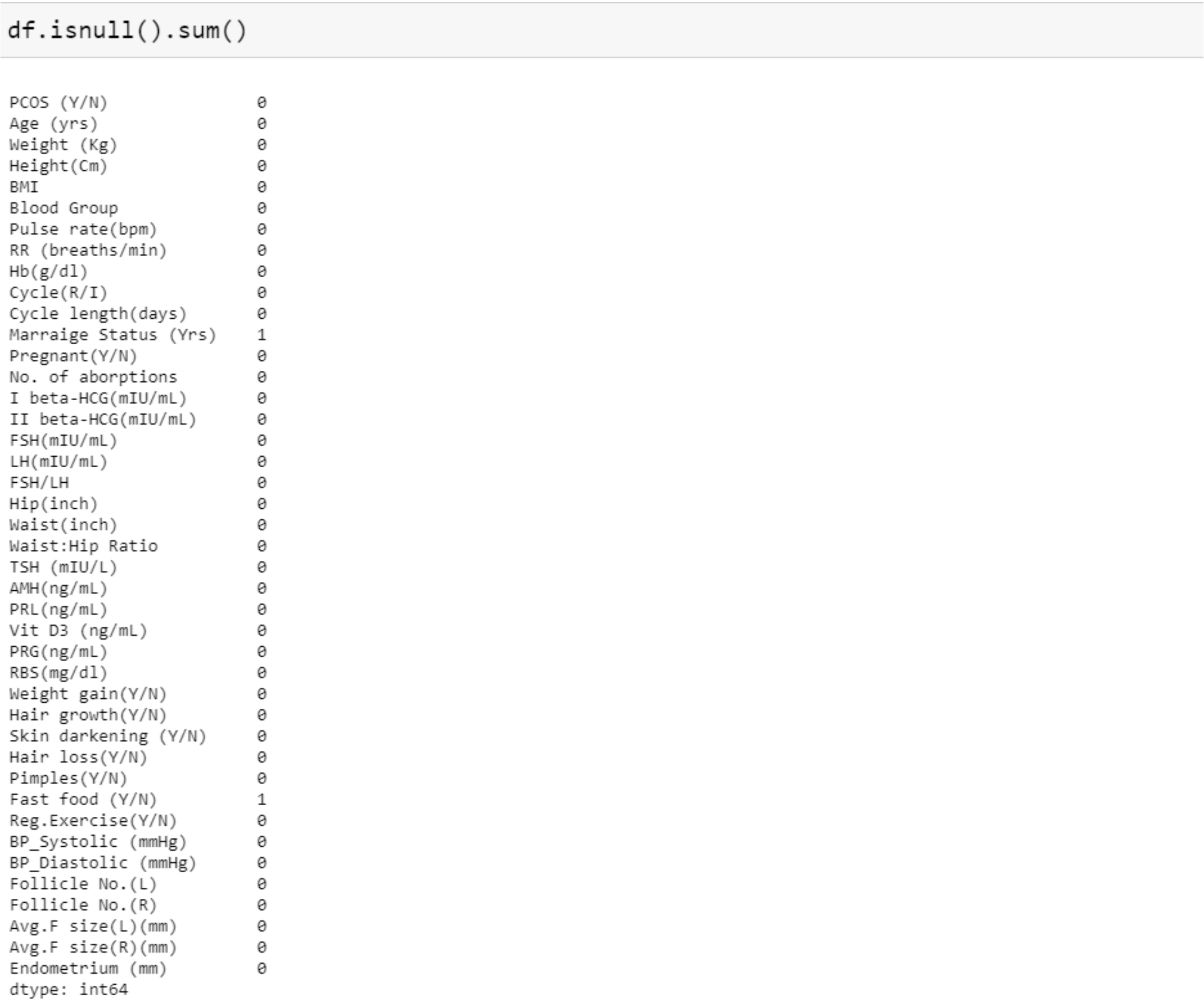


```
df.columns
```

```
Index(['Sl. No', 'Patient File No.', 'PCOS (Y/N)', ' Age (yrs)', 'Weight (Kg)',
       'Height(Cm) ', 'BMI', 'Blood Group', 'Pulse rate(bpm) ',
       'RR (breaths/min)', 'Hb(g/dl)', 'Cycle(R/I)', 'Cycle length(days)',
       'Marraige Status (Yrs)', 'Pregnant(Y/N)', 'No. of aborptions',
       '  I   beta-HCG(mIU/mL)', 'II    beta-HCG(mIU/mL)', 'FSH(mIU/mL)',
       'LH(mIU/mL)', 'FSH/LH', 'Hip(inch)', 'Waist(inch)', 'Waist:Hip Ratio',
       'TSH (mIU/L)', 'AMH(ng/mL)', 'PRL(ng/mL)', 'Vit D3 (ng/mL)',
       'PRG(ng/mL)', 'RBS(mg/dl)', 'Weight gain(Y/N)', 'hair growth(Y/N)',
       'Skin darkening (Y/N)', 'Hair loss(Y/N)', 'Pimples(Y/N)',
       'Fast food (Y/N)', 'Reg.Exercise(Y/N)', 'BP _Systolic (mmHg)',
       'BP _Diastolic (mmHg)', 'Follicle No. (L)', 'Follicle No. (R)',
       'Avg. F size (L) (mm)', 'Avg. F size (R) (mm)', 'Endometrium (mm)',
       'Unnamed: 44'],
      dtype='object')
```

3.3. Data cleaning and transformation

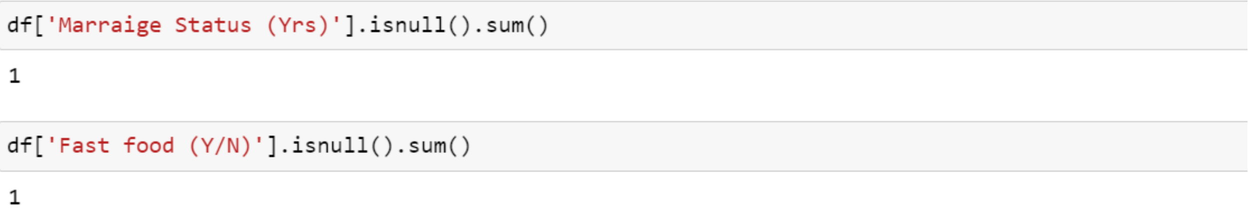
Check for the sum of missing values in the columns



```
df.isnull().sum()
```

```
PCOS (Y/N)               0
Age (yrs)                0
Weight (Kg)              0
Height(Cm)               0
BMI                      0
Blood Group              0
Pulse rate(bpm)          0
RR (breaths/min)         0
Hb(g/dl)                 0
Cycle(R/I)               0
Cycle length(days)       0
Marraige Status (Yrs)    1
Pregnant(Y/N)            0
No. of aborptions        0
I beta-HCG(mIU/mL)       0
II beta-HCG(mIU/mL)      0
FSH(mIU/mL)              0
LH(mIU/mL)               0
FSH/LH                   0
Hip(inch)                0
Waist(inch)              0
Waist:Hip Ratio          0
TSH (mIU/L)              0
AMH(ng/mL)               0
PRL(ng/mL)               0
Vit D3 (ng/mL)           0
PRG(ng/mL)               0
RBS(mg/dl)               0
Weight gain(Y/N)         0
Hair growth(Y/N)         0
Skin darkening (Y/N)     0
Hair loss(Y/N)           0
Pimples(Y/N)             0
Fast food (Y/N)          1
Reg.Exercise(Y/N)        0
BP_Systolic (mmHg)       0
BP_Diastolic (mmHg)      0
Follicle No.(L)          0
Follicle No.(R)          0
Avg.F size(L)(mm)        0
Avg.F size(R)(mm)        0
Endometrium (mm)         0
dtype: int64
```

Fwe see that the column ‘Fast Food (Y/N) and column ‘Marriage Status (Yrs) consist of missing/null values



```
df['Marraige Status (Yrs)'].isnull().sum()
```

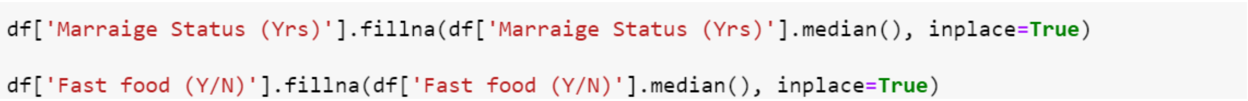
```
1
```

```
df['Fast food (Y/N)'].isnull().sum()
```

```
1
```

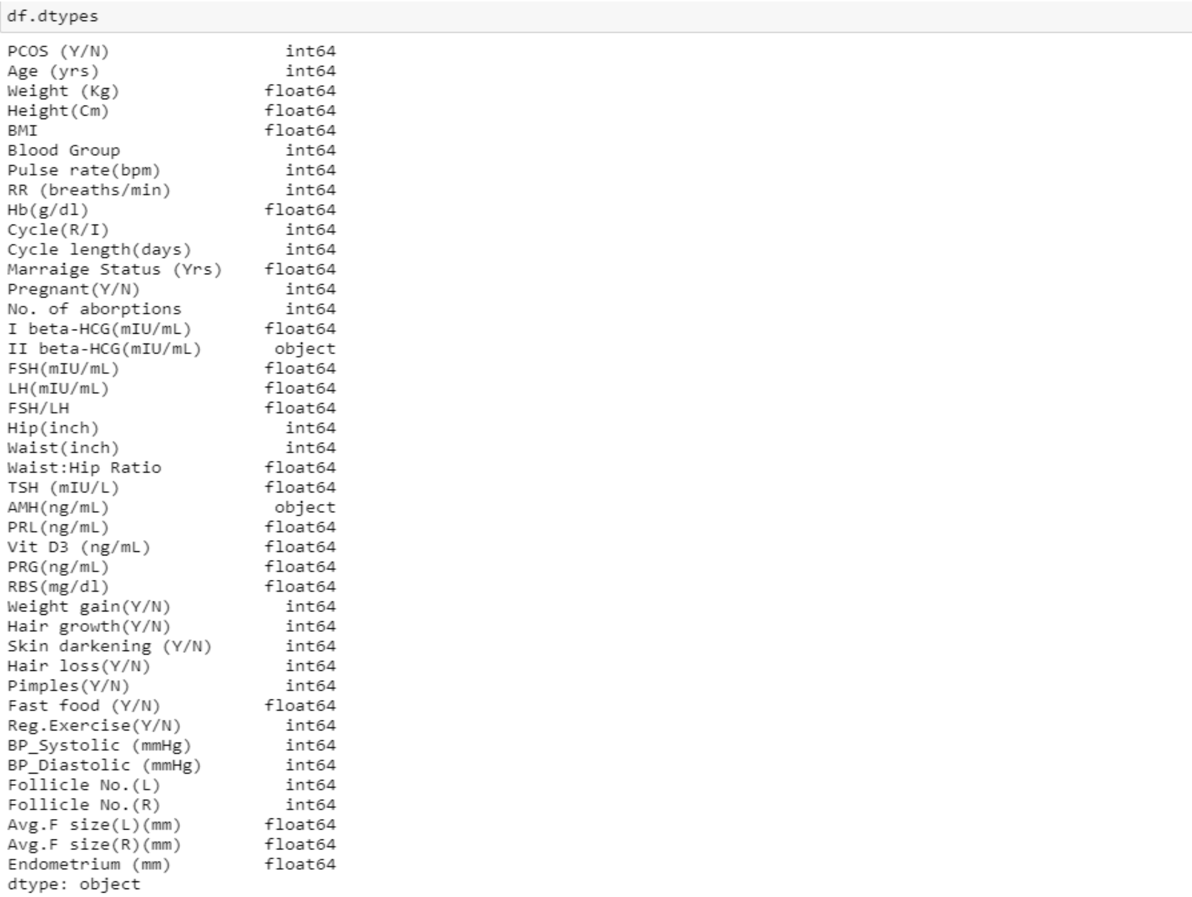
Fill the null values in columns

Since the median is less affected by any potential outliers in the data, we will impute the missing values with their median.



```
df['Marraige Status (Yrs)'].fillna(df['Marraige Status (Yrs)'].median(), inplace=True)

df['Fast food (Y/N)'].fillna(df['Fast food (Y/N)'].median(), inplace=True)
```

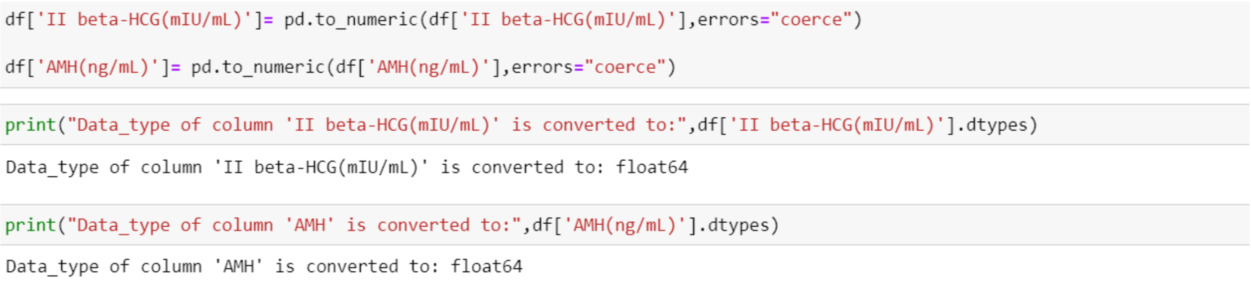
Checking for the data types



```
df.dtypes
```

```
PCOS (Y/N)                 int64
Age (yrs)                  int64
Weight (Kg)              float64
Height(Cm)               float64
BMI                      float64
Blood Group                int64
Pulse rate(bpm)            int64
RR (breaths/min)           int64
Hb(g/dl)                 float64
Cycle(R/I)                 int64
Cycle length(days)         int64
Marraige Status (Yrs)    float64
Pregnant(Y/N)              int64
No. of aborptions          int64
I beta-HCG(mIU/mL)       float64
II beta-HCG(mIU/mL)       object
FSH(mIU/mL)              float64
LH(mIU/mL)               float64
FSH/LH                   float64
Hip(inch)                  int64
Waist(inch)                int64
Waist:Hip Ratio          float64
TSH (mIU/L)              float64
AMH(ng/mL)                object
PRL(ng/mL)               float64
Vit D3 (ng/mL)           float64
PRG(ng/mL)               float64
RBS(mg/dl)               float64
Weight gain(Y/N)           int64
Hair growth(Y/N)           int64
Skin darkening (Y/N)       int64
Hair loss(Y/N)             int64
Pimples(Y/N)               int64
Fast food (Y/N)          float64
Reg.Exercise(Y/N)          int64
BP _Systolic (mmHg)        int64
BP _Diastolic (mmHg)       int64
Follicle No.(L)            int64
Follicle No.(R)            int64
Avg.F size(L)(mm)        float64
Avg.F size(R)(mm)        float64
Endometrium (mm)         float64
dtype: object
```

We see data types of a few columns(II beta-HCG(mIU/mL),AMH(ng/mL)) need to be changed, both the columns contain continuous values and need to be converted to type: 'numeric'



```
df['II beta-HCG(mIU/mL)']= pd.to_numeric(df['II beta-HCG(mIU/mL)'],errors="coerce")

df['AMH(ng/mL)']= pd.to_numeric(df['AMH(ng/mL)'],errors="coerce")
```

```
print("Data_type of column 'II beta-HCG(mIU/mL)' is converted to:",df['II beta-HCG(mIU/mL)'].dtypes)
```

```
Data_type of column 'II beta-HCG(mIU/mL)' is converted to: float64
```

```
print("Data_type of column 'AMH' is converted to:",df['AMH(ng/mL)'].dtypes)
```

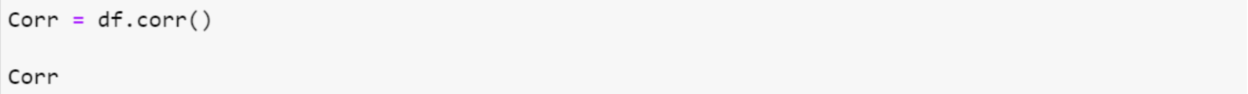
```
Data_type of column 'AMH' is converted to: float64
```

4. EXPLORATORY DATA ANALYSIS

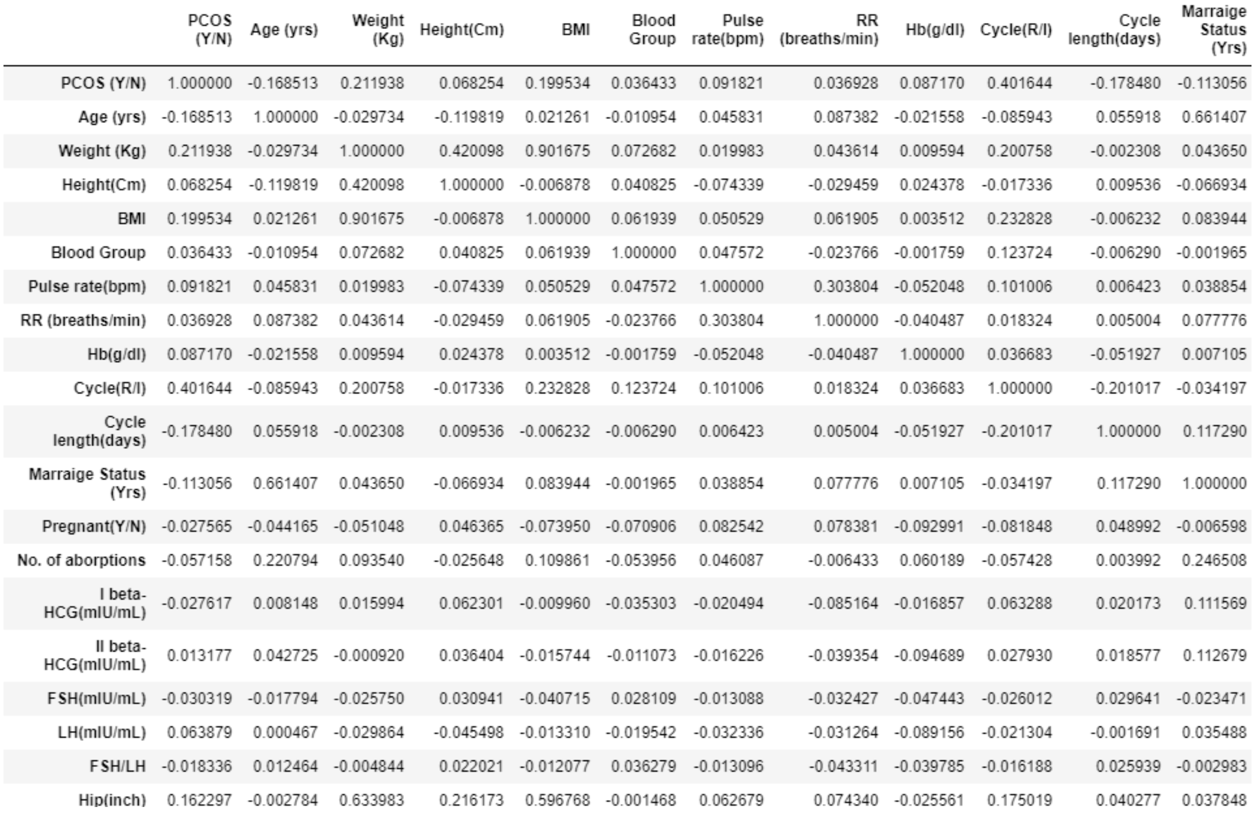
4.1. Filtering features using Pearson correlation.

Before we start exploring all the different variables, it is important to acknowledge that including too many features as a part of our training could negatively impact the accuracy of prediction of our model.

We use the Pearson Correlation to identity and filter out the features which are highly correlated between each other but at the same time make sure we retain the features which are high correlated with the target variable.



```
Corr = df.corr()

Corr
```



	PCOS (Y/N)	Age (yrs)	Weight (Kg)	Height(Cm)	BMI	Blood Group	Pulse rate(bpm)	RR (breaths/min)	Hb(g/dl)	Cycle(R/I)	Cycle length(days)	Marraige Status (Yrs)
PCOS (Y/N)	1.000000	-0.168513	0.211938	0.068254	0.199534	0.036433	0.091821	0.036928	0.087170	0.401644	-0.178480	-0.113056
Age (yrs)	-0.168513	1.000000	-0.029734	-0.119819	0.021261	-0.010954	0.045831	0.087382	-0.021558	-0.085943	0.055918	0.661407
Weight (Kg)	0.211938	-0.029734	1.000000	0.420098	0.901675	0.072682	0.019983	0.043614	0.009594	0.200758	-0.002308	0.043650
Height(Cm)	0.068254	-0.119819	0.420098	1.000000	-0.006878	0.040825	-0.074339	-0.029459	0.024378	-0.017336	0.009536	-0.066934
BMI	0.199534	0.021261	0.901675	-0.006878	1.000000	0.061939	0.050529	0.061905	0.003512	0.232828	-0.006232	0.083944
Blood Group	0.036433	-0.010954	0.072682	0.040825	0.061939	1.000000	0.047572	-0.023766	-0.001759	0.123724	-0.006290	-0.001965
Pulse rate(bpm)	0.091821	0.045831	0.019983	-0.074339	0.050529	0.047572	1.000000	0.303804	-0.052048	0.101006	0.006423	0.038854
RR (breaths/min)	0.036928	0.087382	0.043614	-0.029459	0.061905	-0.023766	0.303804	1.000000	-0.040487	0.018324	0.005004	0.077776
Hb(g/dl)	0.087170	-0.021558	0.009594	0.024378	0.003512	-0.001759	-0.052048	-0.040487	1.000000	0.036683	-0.051927	0.007105
Cycle(R/I)	0.401644	-0.085943	0.200758	-0.017336	0.232828	0.123724	0.101006	0.018324	0.036683	1.000000	-0.201017	-0.034197
Cycle length(days)	-0.178480	0.055918	-0.002308	0.009536	-0.006232	-0.006290	0.006423	0.005004	-0.051927	-0.201017	1.000000	0.117290
Marraige Status (Yrs)	-0.113056	0.661407	0.043650	-0.066934	0.083944	-0.001965	0.038854	0.077776	0.007105	-0.034197	0.117290	1.000000
Pregnant(Y/N)	-0.027565	-0.044165	-0.051048	0.046365	-0.073950	-0.070906	0.082542	0.078381	-0.092991	-0.081848	0.048992	-0.006598
No. of aborptions	-0.057158	0.220794	0.093540	-0.025648	0.109861	-0.053956	0.046087	-0.006433	0.060189	-0.057428	0.003992	0.246508
I beta-HCG(mIU/mL)	-0.027617	0.008148	0.015994	0.062301	-0.009960	-0.035303	-0.020494	-0.085164	-0.016857	0.063288	0.020173	0.111569
II beta-HCG(mIU/mL)	0.013177	0.042725	-0.000920	0.036404	-0.015744	-0.011073	-0.016226	-0.039354	-0.094689	0.027930	0.018577	0.112679
FSH(mIU/mL)	-0.030319	-0.017794	-0.025750	0.030941	-0.040715	0.028109	-0.013088	-0.032427	-0.047443	-0.026012	0.029641	-0.023471
LH(mIU/mL)	0.063879	0.000467	-0.029864	-0.045498	-0.013310	-0.019542	-0.032336	-0.031264	-0.089156	-0.021304	-0.001691	0.035488
FSH/LH	-0.018336	0.012464	-0.004844	0.022021	-0.012077	0.036279	-0.013096	-0.043311	-0.039785	-0.016188	0.025939	-0.002983
Hip(inch)	0.162297	-0.002784	0.633983	0.216173	0.596768	-0.001468	0.062679	0.074340	-0.025561	0.175019	0.040277	0.037848

Since we have a lot of variables, we will summarize the results using a heatmap & further sort the independent variables with relevance to their correlation with the target variable.

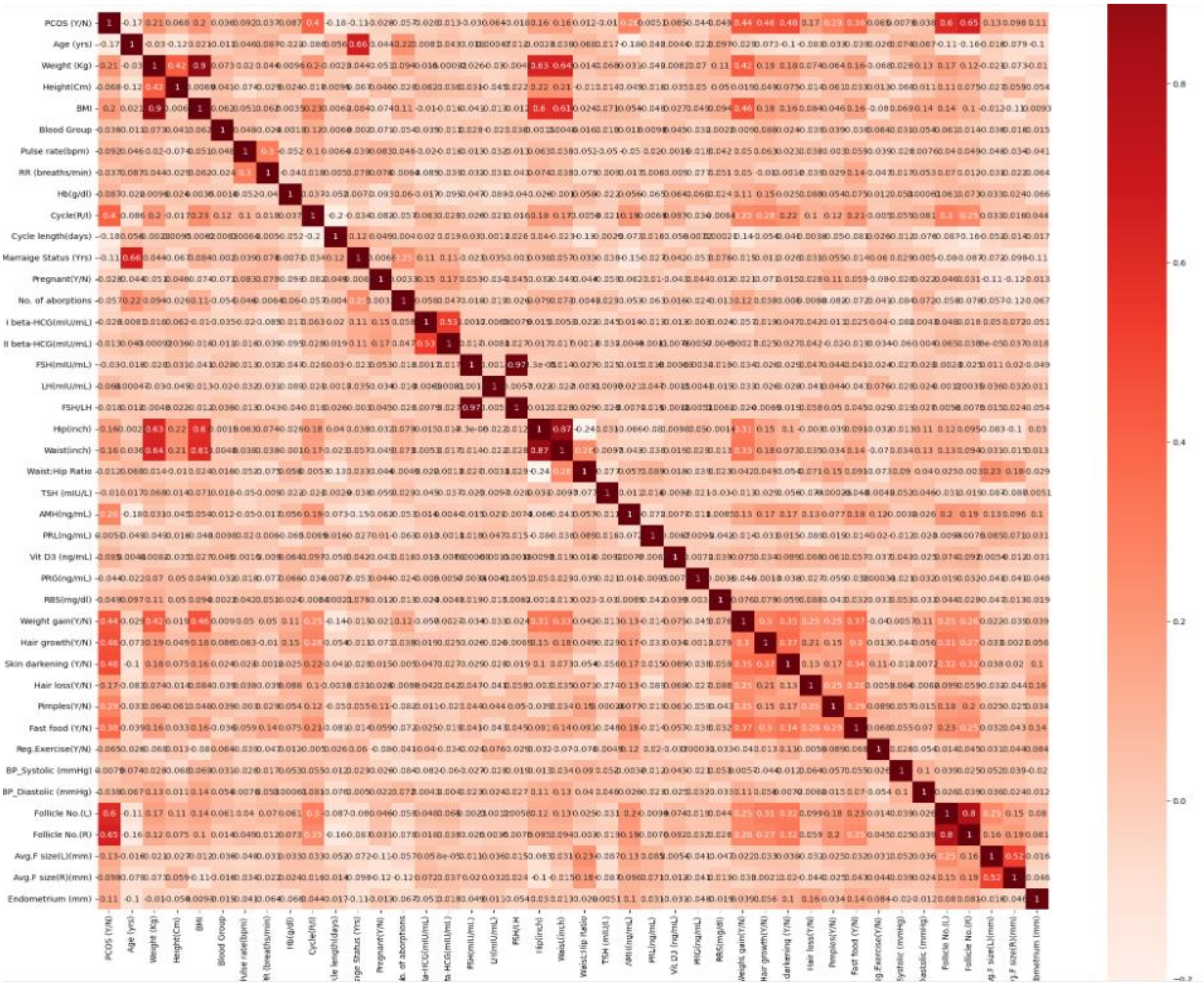


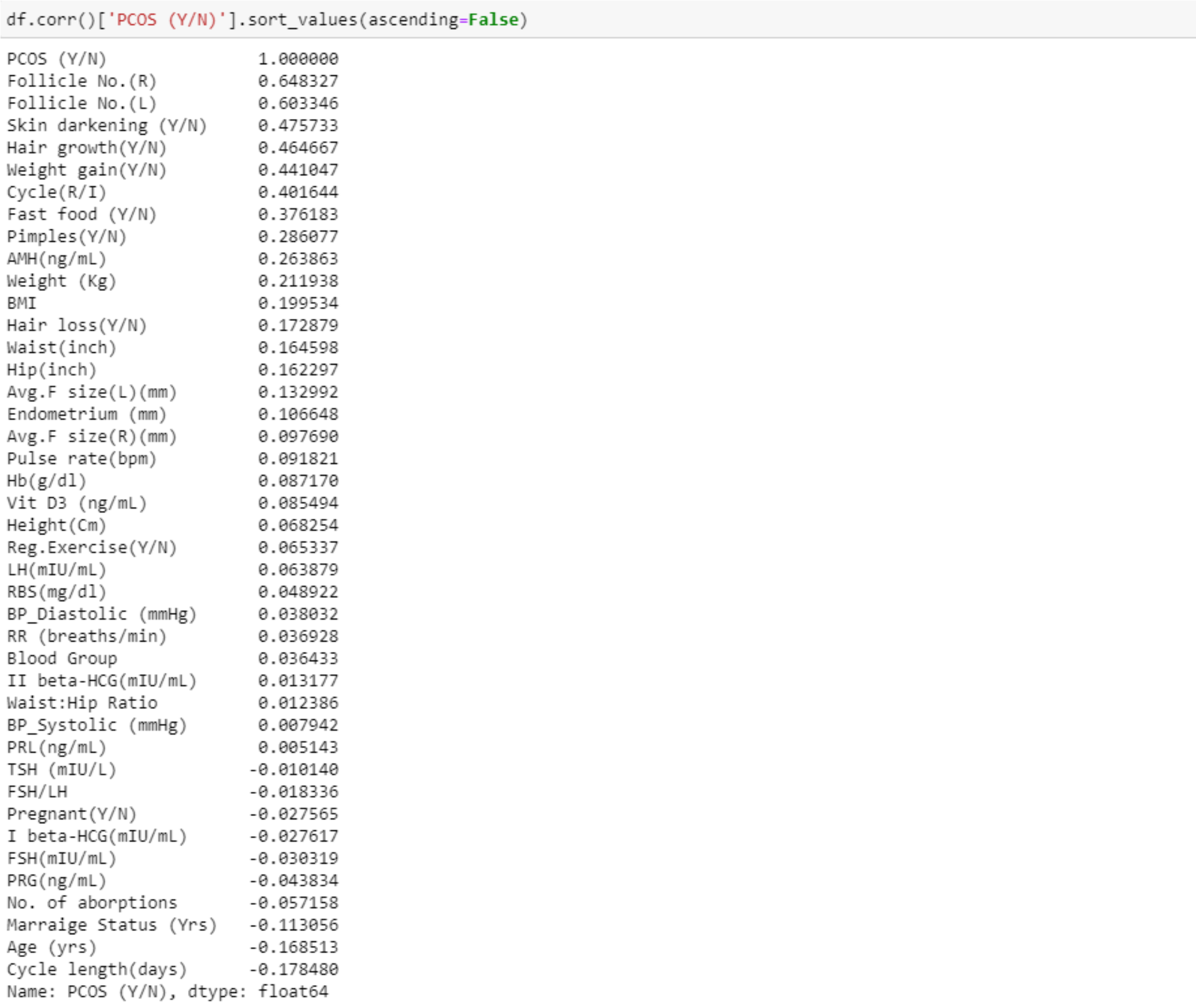
Fig.1. Heatmap of the Correlation matrix

Since our dataset comprises of more than 40 features, we will use the Pearson correlation matrix to select only features that will have a greater impact on the performance of the model. Assuming a threshold of 0.8 for correlation between the features, all the independent features which are correlated by 0.8 or more with each other will be examined and one of those correlated independent features will be excluded from our training dataset.

Based on the results from heatmap of the correlation matrix fig.1 and the correlation of the independent features with the target variable (below)

- We see that the independent features Weight and BMI have a correlation of 0.9, therefore we exclude the independent feature BMI from our training data.

- Similarly features FSH(mIU/mL) and FSH/LH are highly correlated between each other and have a correlation of 0.97, therefore considering their relevance with the target variable, we exclude the variable FSH/LH from our training dataset.
- Features Waist(inch) and Hip(inch) share a correlation of 0.87, therefore we exclude Hip(inch) from our final training dataset.
- Features Follicle No.(R) and Follicle No.(L) have a correlation of 0.8, but since they are both highly correlated with the target variable , we decide not to exclude either of these.

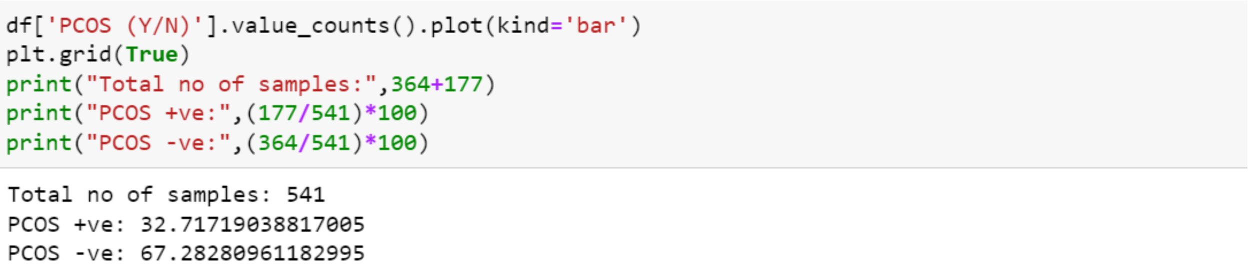


```
df.corr()['PCOS (Y/N)'].sort_values(ascending=False)
```

```
PCOS (Y/N)                1.000000
Follicle No.(R)           0.648327
Follicle No.(L)           0.603346
Skin darkening (Y/N)      0.475733
Hair growth(Y/N)          0.464667
Weight gain(Y/N)          0.441047
Cycle(R/I)                0.401644
Fast food (Y/N)           0.376183
Pimples(Y/N)              0.286077
AMH(ng/mL)                0.263863
Weight (Kg)               0.211938
BMI                       0.199534
Hair loss(Y/N)            0.172879
Waist(inch)               0.164598
Hip(inch)                 0.162297
Avg.F size(L)(mm)         0.132992
Endometrium (mm)          0.106648
Avg.F size(R)(mm)         0.097690
Pulse rate(bpm)           0.091821
Hb(g/dl)                  0.087170
Vit D3 (ng/mL)            0.085494
Height(Cm)                0.068254
Reg.Exercise(Y/N)         0.065337
LH(mIU/mL)                0.063879
RBS(mg/dl)                0.048922
BP_Diastolic (mmHg)       0.038032
RR (breaths/min)          0.036928
Blood Group               0.036433
II beta-HCG(mIU/mL)       0.013177
Waist:Hip Ratio           0.012386
BP_Systolic (mmHg)        0.007942
PRL(ng/mL)                0.005143
TSH (mIU/L)              -0.010140
FSH/LH                   -0.018336
Pregnant(Y/N)            -0.027565
I beta-HCG(mIU/mL)       -0.027617
FSH(mIU/mL)              -0.030319
PRG(ng/mL)               -0.043834
No. of aborptions        -0.057158
Marraige Status (Yrs)    -0.113056
Age (yrs)                -0.168513
Cycle length(days)       -0.178480
Name: PCOS (Y/N), dtype: float64
```

Going further we will explore & analyze only the features which have either high positive or high negative correlations with the target variable (also excluding those which were filtered using the Pearson correlation technique)

4.2. Checking the balance of classes in the data set



```
df['PCOS (Y/N)'].value_counts().plot(kind='bar')
plt.grid(True)
print("Total no of samples:",364+177)
print("PCOS +ve:",(177/541)*100)
print("PCOS -ve:",(364/541)*100)
```

```
Total no of samples: 541
PCOS +ve: 32.71719038817005
PCOS -ve: 67.28280961182995
```

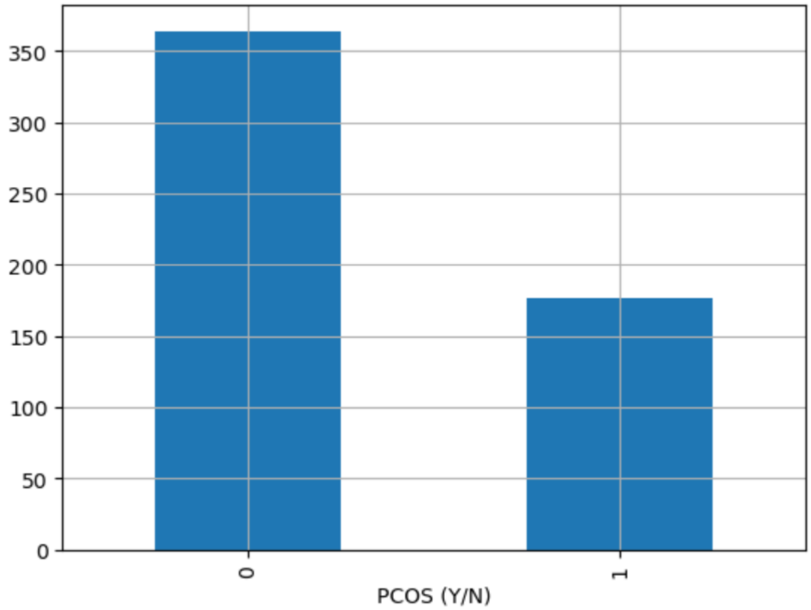
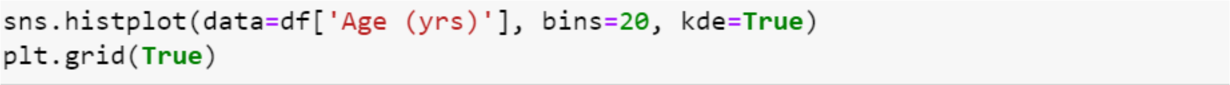


Fig.2. Balance of target classes

Here we see that 67.28% samples of the dataset belong to the class '0' i.e. Negative PCOS, and 32.71% of the samples belong to the class '1' i.e. Positive PCOS.

From the above viz, we infer that there is a mild imbalance in the distribution of the classes. (*Imbalanced Data*, n.d.) highlights that this would not have any significant impact on performance of the model, so we proceed without any further treatment.

4.3. Distribution of the feature ‘Age’



```
sns.histplot(data=df['Age (yrs)'], bins=20, kde=True)
plt.grid(True)
```

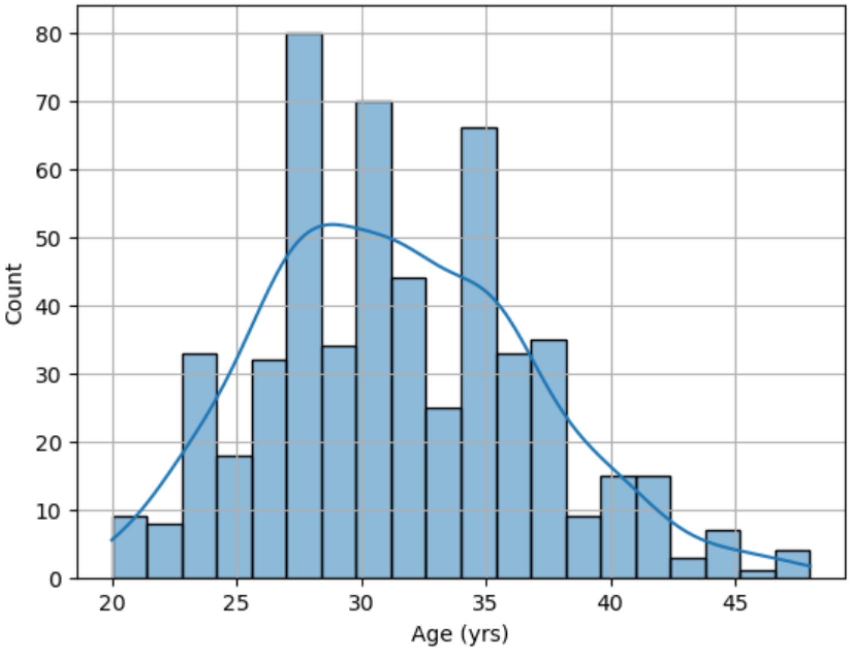
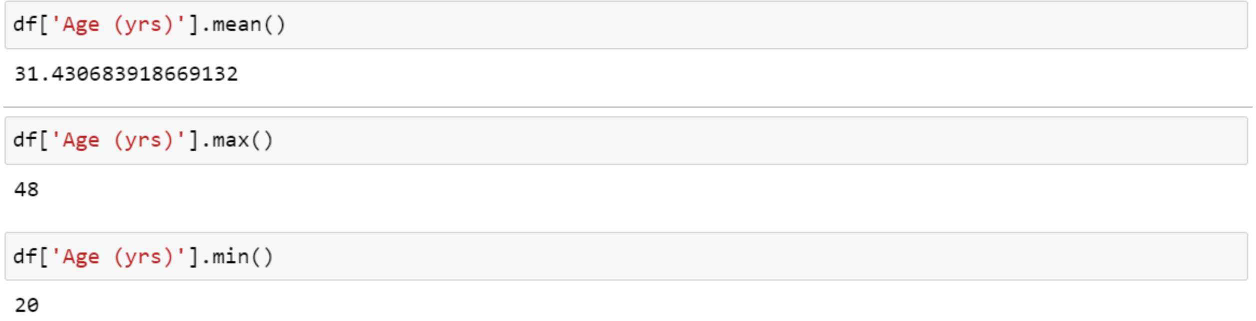


Fig.3. Distribution of Age groups



```
df['Age (yrs)'].mean()
```

```
31.430683918669132
```

```
df['Age (yrs)'].max()
```

```
48
```

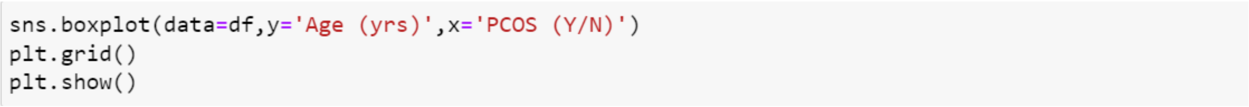
```
df['Age (yrs)'].min()
```

```
20
```

Age of the samples is normally distributed with peaks between 28years- 35years, and the mean age of the samples in the dataset is around 31 years. The age groups of the samples range between 20 to 48 years old.

4.4. Relationship between Age and PCOS

We will use a box plot to visualize this relationship



```
sns.boxplot(data=df,y='Age (yrs)',x='PCOS (Y/N)')
plt.grid()
plt.show()
```

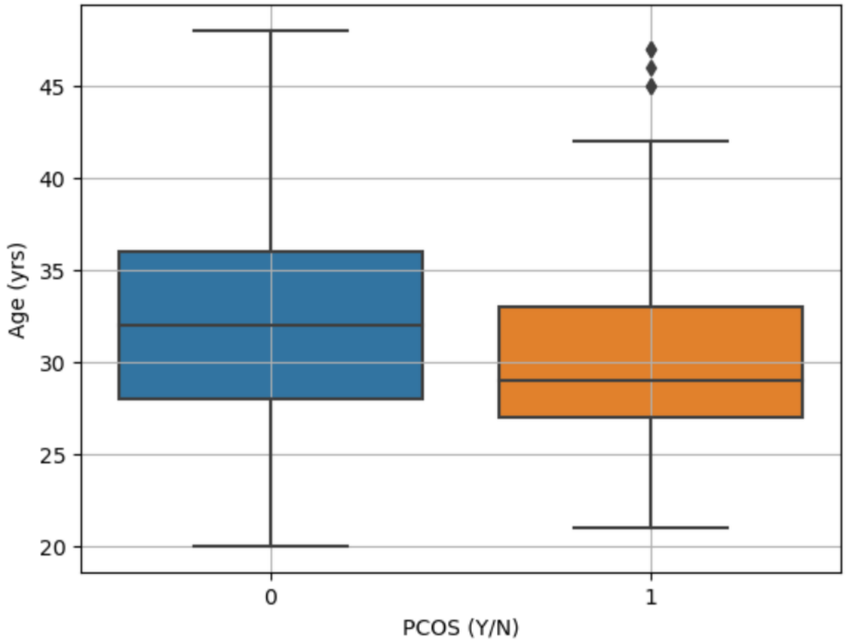
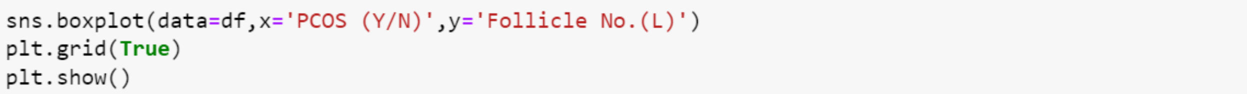


Fig.3. Age vs PCOS

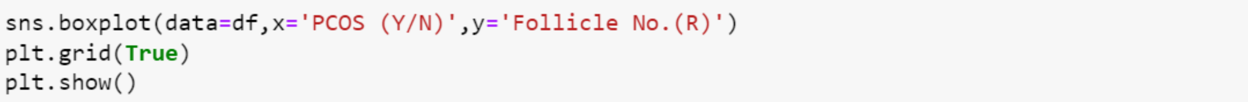
From the above boxplot, we can infer that average age of the women with PCOS is approximately around 28 years whereas the average age of women without PCOS is approximately 32 years.

We see that 75% of the women with PCOS are in their 20s and 30s, this could also mean that PCOS is more common in women in their reproductive age. Reproductive age in women as referred by (*Having a Baby After Age 35: How Aging Affects Fertility and Pregnancy*, n.d.),is usually between the late teens and late 20s.

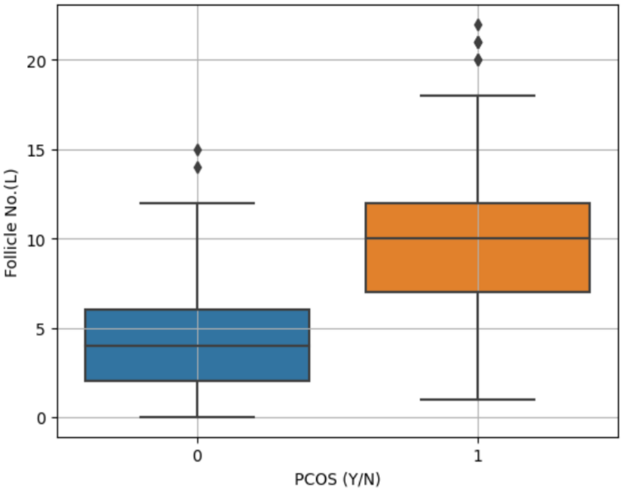
4.5. Relationship between follicle numbers in the left/right ovaries with PCOS

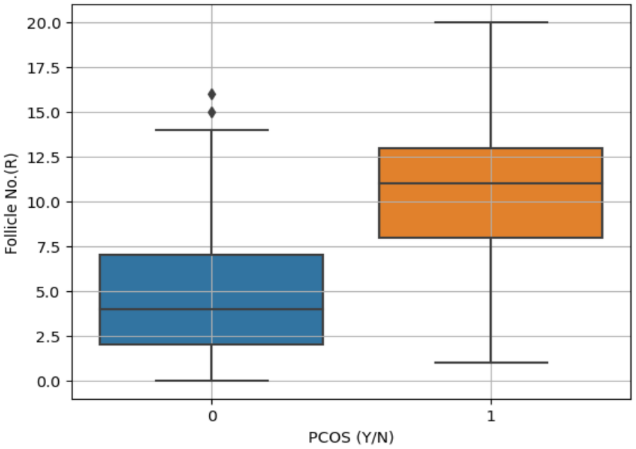


```
sns.boxplot(data=df,x='PCOS (Y/N)',y='Follicle No.(L)')
plt.grid(True)
plt.show()
```



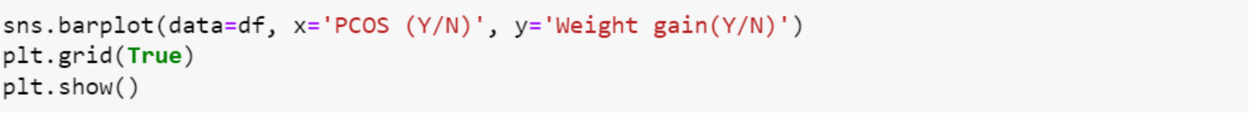
```
sns.boxplot(data=df,x='PCOS (Y/N)',y='Follicle No.(R)')
plt.grid(True)
plt.show()
```





Here the class 1 represents the individuals with PCOS and 0 represents the individuals without PCOS. The average number of follicles in women affected with PCOS is very high in comparison to those women without PCOS. It can also be seen that more than 95% of the women who are not affected with PCOS have follicles less 12. Therefore, we infer that the women with PCOS tend to develop more follicles.

4.6. Relationship between Weight gain and PCOS



```
sns.barplot(data=df, x='PCOS (Y/N)', y='Weight gain(Y/N)')
plt.grid(True)
plt.show()
```

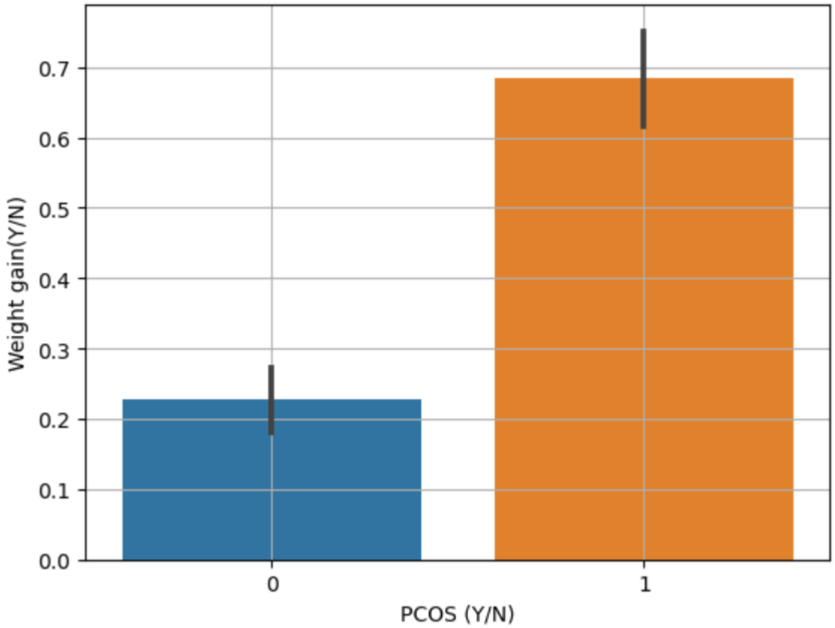
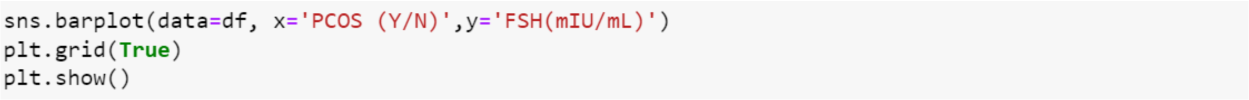


Fig.6. Weight gain vs PCOS

With this plot it is evident that women with PCOS are more prone to weight gain and could be obese. Women with PCOS have an increase in Insulin resistance this could also lead to weight gain.

4.7. Relationship between FSH levels and PCOS

As per the research by Johansson and Stener-Victorin (2013) women with PCOS tend to exhibit lower levels of FSH. After having visualized the samples in fig.7 it now evident that the Levels of follicle stimulating hormone (FSH) in women with PCOS is low or is within the lower follicular range.



```
sns.barplot(data=df, x='PCOS (Y/N)',y='FSH(mIU/mL)')
plt.grid(True)
plt.show()
```

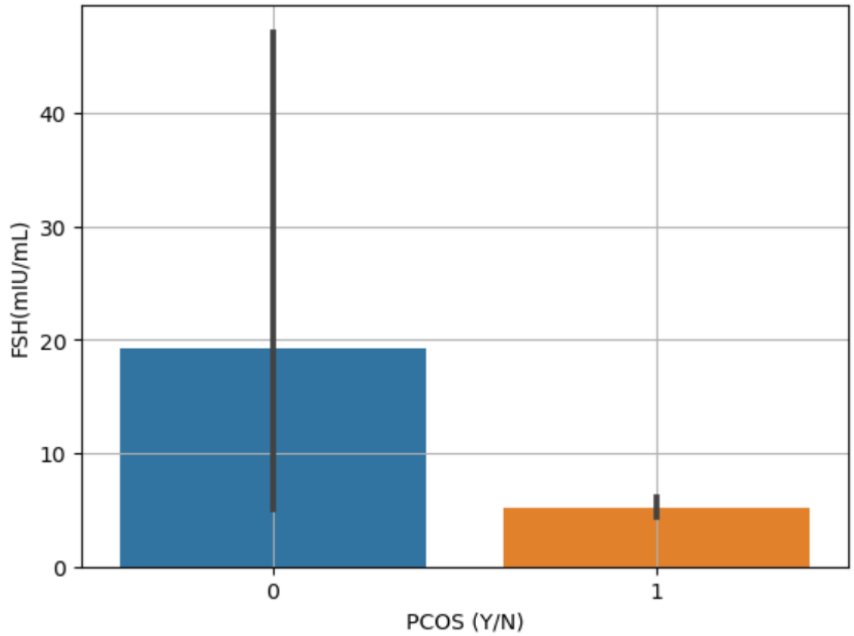
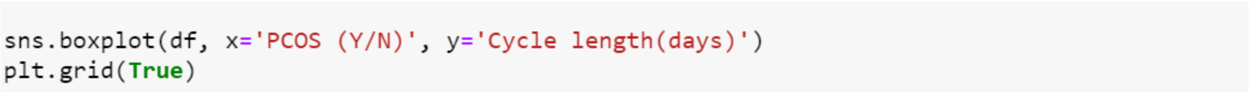


Fig.7. FSH levels vs PCOS

4.8. Understanding the relationship between Cycle length (days) and PCOS



```
sns.boxplot(df, x='PCOS (Y/N)', y='Cycle length(days)')
plt.grid(True)
```

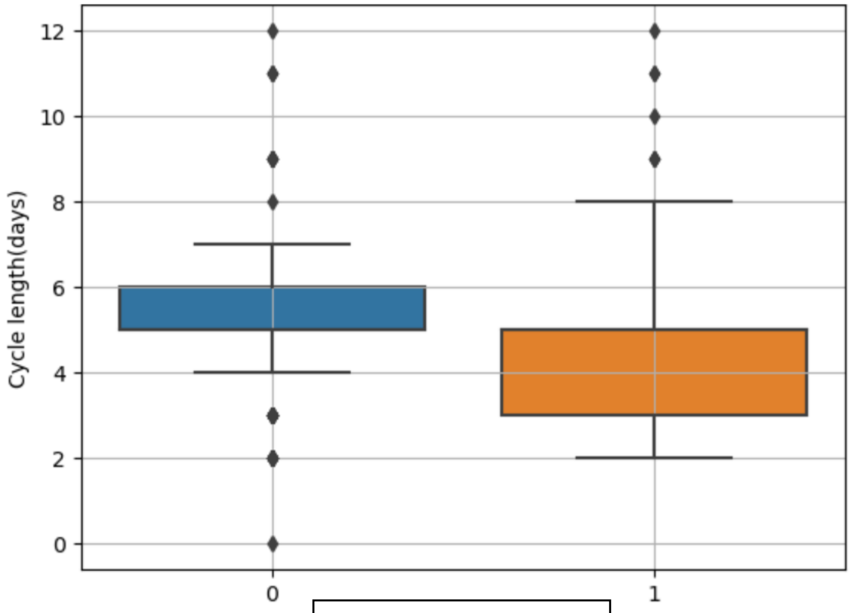


Fig.8. FSH levels vs PCOS

Here we have generated a Boxplot to visualize and understand how cycle lengths(days) vary in the women with PCOS in comparison to those without PCOS.

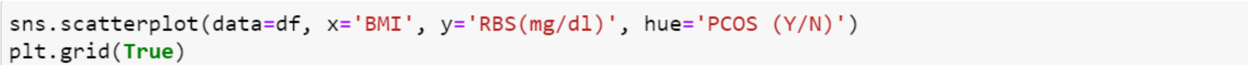
Women without PCOS

And from the above visualization we can infer that cycle lengths of 50% of women ranges between 5-6 days and for more than 90 percent of the women between 4-7 days.

Women with PCOS

In women with PCOS the average cycle length ranges around 4 days, 75% of women with PCOS have cycle lengths ranging between 2-5 days but the range of cycle lengths of all the samples with PCOS is much wider, which could mean that those with PCOS have irregular cycle lengths.

4.9. Relationship between Blood sugar levels, BMI & PCOS



```
sns.scatterplot(data=df, x='BMI', y='RBS(mg/dl)', hue='PCOS (Y/N)')
plt.grid(True)
```

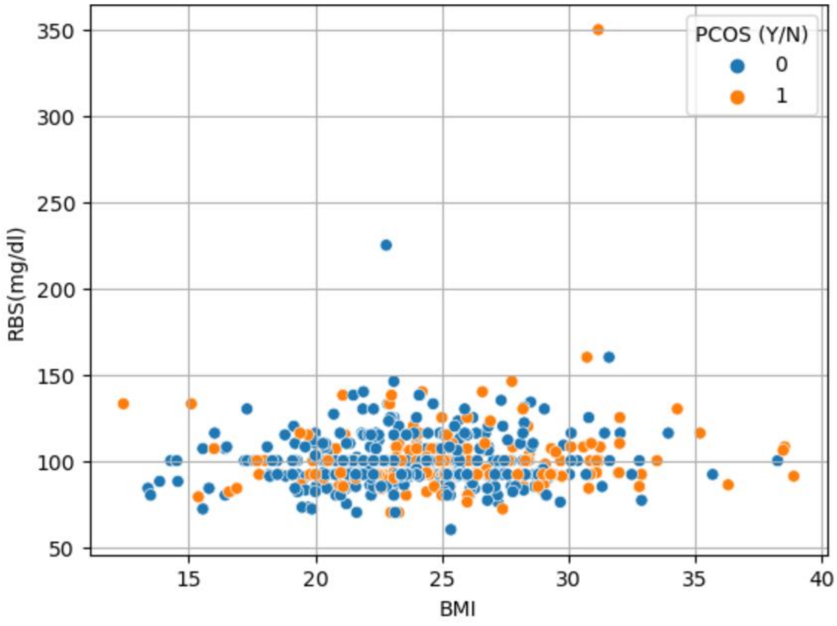


Fig.9. RBS vs BMI , PCOS

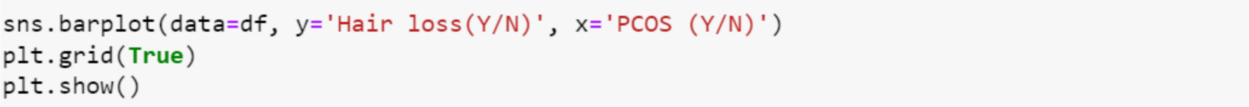
Based on a general assumption that the samples in our dataset were collected from a fasting blood sugar test:

Study Giant and Giant (2022) shows that blood tests performed after fasting, it is considered that a reading of 126mg/dL indicates a possibility of diabetes whereas 100mg/dL is normal.

Hence, we can infer that most of the samples which have blood sugar levels above 100mg/dL are prone to diabetes and have PCOS.

By plotting BMI alongside the Blood sugar levels we get a bigger picture of how different variables like the BMI of an individual, Blood sugar levels and PCOS are correlated.

4.9.1. Relationship between Hair loss and PCOS



```
sns.barplot(data=df, y='Hair loss(Y/N)', x='PCOS (Y/N)')
plt.grid(True)
plt.show()
```

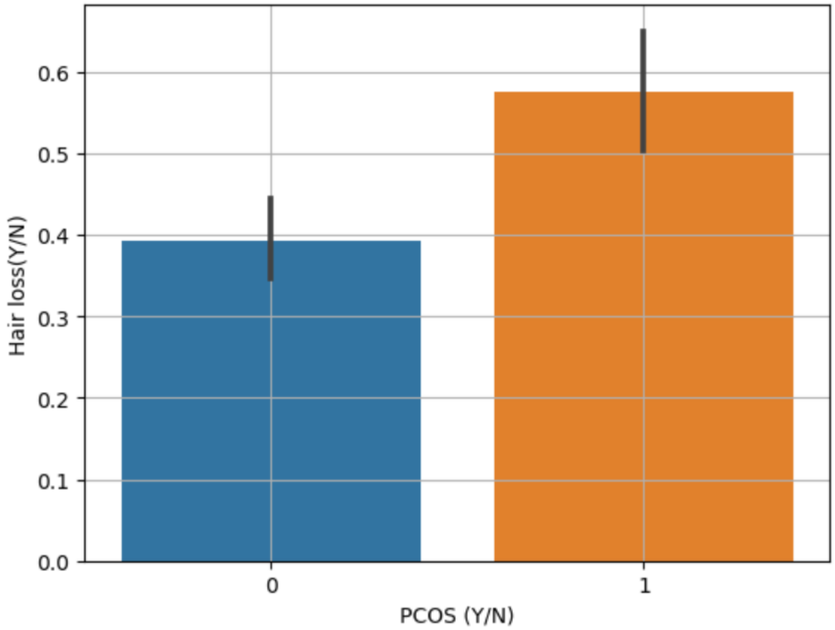


Fig.10. Hair loss vs PCOS

Hair loss seems to be more prevalent in women with PCOS than in women without PCOS.

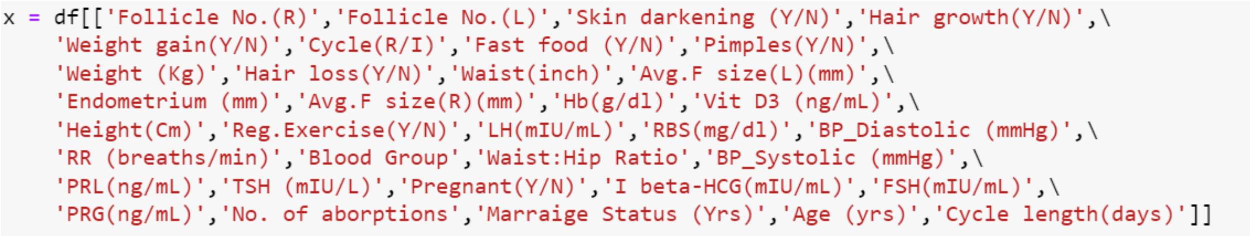
5. MODEL SELECTION AND IMPLEMENTATION

5.1. Building the model

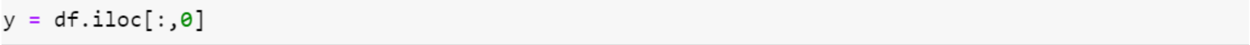
To better present the process , we have split the process into multiple granular steps namely:

- Arranging the data
- Splitting the datasets
- Training the model
- Testing the model
- Evaluating the model

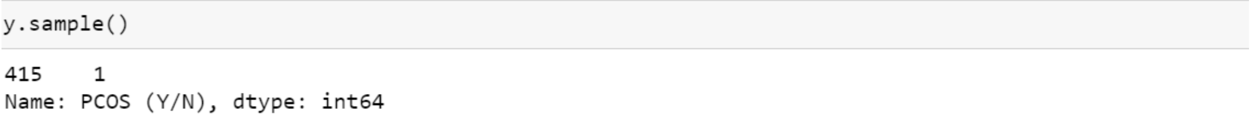
5.1.1. Arranging the data



```
x = df[['Follicle No.(R)','Follicle No.(L)','Skin darkening (Y/N)','Hair growth(Y/N)',\
    'Weight gain(Y/N)','Cycle(R/I)','Fast food (Y/N)','Pimples(Y/N)',\
    'Weight (Kg)','Hair loss(Y/N)','Waist(inch)','Avg.F size(L)(mm)',\
    'Endometrium (mm)','Avg.F size(R)(mm)','Hb(g/dl)','Vit D3 (ng/mL)',\
    'Height(Cm)','Reg.Exercise(Y/N)','LH(mIU/mL)','RBS(mg/dl)','BP_Diastolic (mmHg)',\
    'RR (breaths/min)','Blood Group','Waist:Hip Ratio','BP_Systolic (mmHg)',\
    'PRL(ng/mL)','TSH (mIU/L)','Pregnant(Y/N)','I beta-HCG(mIU/mL)','FSH(mIU/mL)',\
    'PRG(ng/mL)','No. of aborptions','Marraige Status (Yrs)','Age (yrs)','Cycle length(days)']]
```



```
y = df.iloc[:,0]
```



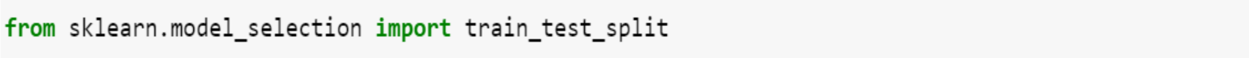
```
y.sample()
```

```
415    1
Name: PCOS (Y/N), dtype: int64
```

We have now assigned our target variable (PCOS (Y/N)) to object x and we have assigned the top independent features to object y.

5.1.2. Splitting the datasets

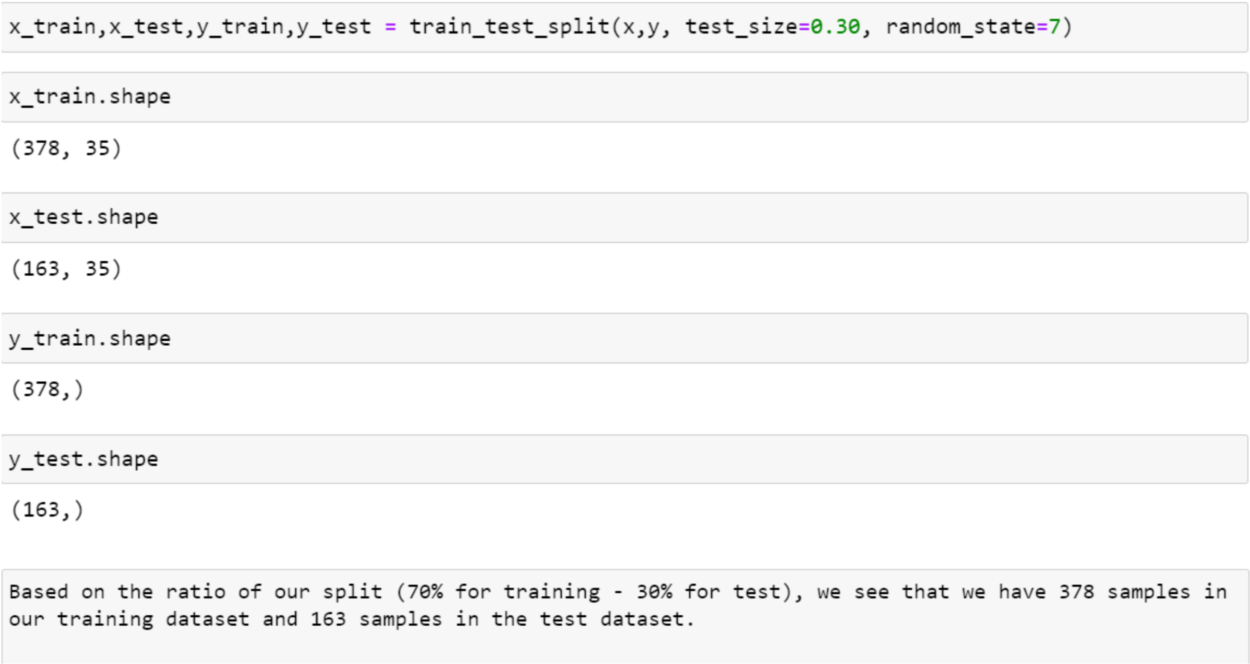
In this phase we split the dataset into two, Training, and the Testing datasets. The model sees and learns the underlying patterns in the data from the training dataset, and further uses the test dataset to validate its predictions.



```
from sklearn.model_selection import train_test_split
```

Before the dataset is split into Training and Test datasets, it is important to decide on the size of the splits. Though the size of the Training dataset and the Test dataset that is to be split depends on the objective we are trying to achieve.

However, in our case we use the general rule of thumb ratio to split the datasets. We will split 70% of the total samples in our dataset as our training dataset and the rest 30% of the samples as our test dataset.



```
x_train,x_test,y_train,y_test = train_test_split(x,y, test_size=0.30, random_state=7)
```

```
x_train.shape
```

```
(378, 35)
```

```
x_test.shape
```

```
(163, 35)
```

```
y_train.shape
```

```
(378,)
```

```
y_test.shape
```

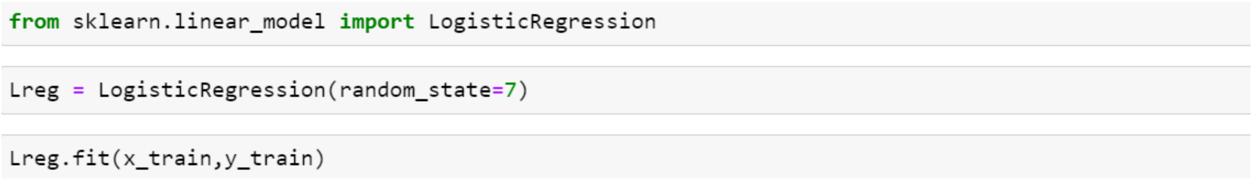
```
(163,)
```

```
Based on the ratio of our split (70% for training - 30% for test), we see that we have 378 samples in
our training dataset and 163 samples in the test dataset.
```

5.2. Training, Testing and Evaluation - Logistic Regression

5.2.1. Training the model

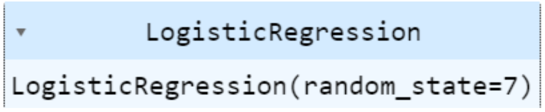
Since the objective of this project is to develop a model that will predict a binary result addressing the presence of PCOS based on a set of Individual features, we will focus primarily on working with the classification models like Logistic Regression and further compare the performance of a model with other different classification models on our dataset.



```
from sklearn.linear_model import LogisticRegression
```

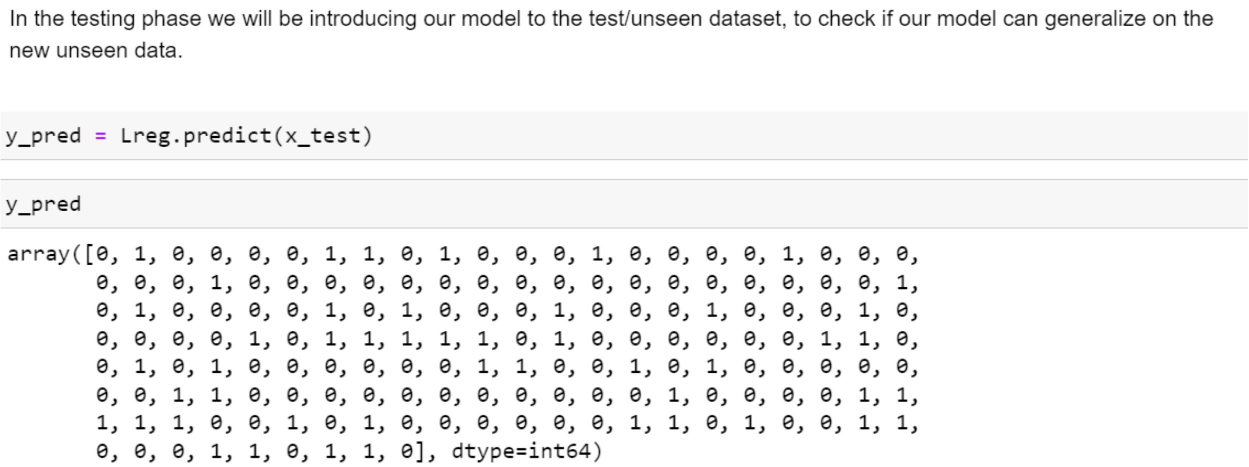
```
Lreg = LogisticRegression(random_state=7)
```

```
Lreg.fit(x_train,y_train)
```



```
▾ LogisticRegression
LogisticRegression(random_state=7)
```

5.2.2. Testing the model



In the testing phase we will be introducing our model to the test/unseen dataset, to check if our model can generalize on the new unseen data.

```
y_pred = Lreg.predict(x_test)
```

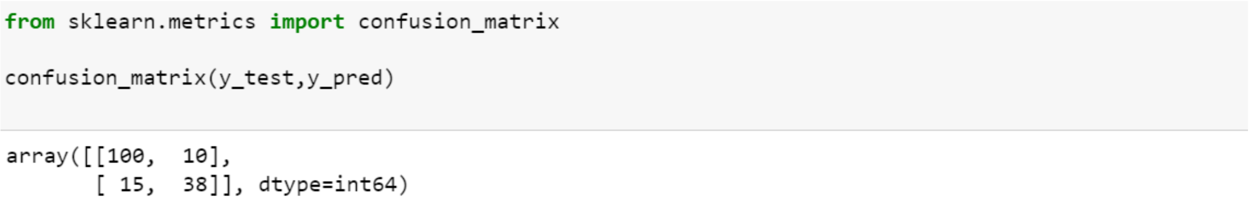
```
y_pred
```

```
array([0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0,
       0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1,
       0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0,
       0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0,
       0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0,
       0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1,
       1, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1,
       0, 0, 0, 1, 1, 0, 1, 1, 0], dtype=int64)
```

5.2.3. Evaluating the model

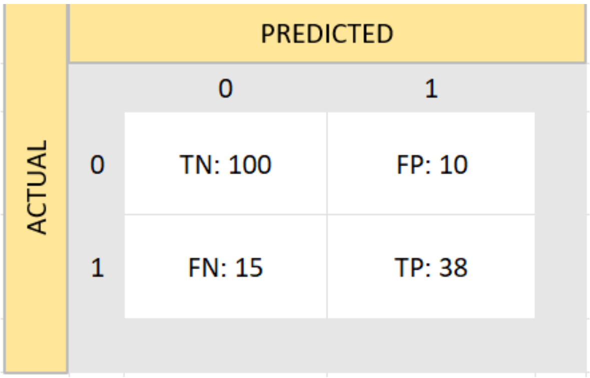
In the Evaluation phase we will check the performance of the test data against the predicted data, to check if our model can generalize on test/ unseen data.

Confusion matrix and its metrics



```
from sklearn.metrics import confusion_matrix

confusion_matrix(y_test,y_pred)
```

```
array([[100,  10],
       [ 15,  38]], dtype=int64)
```



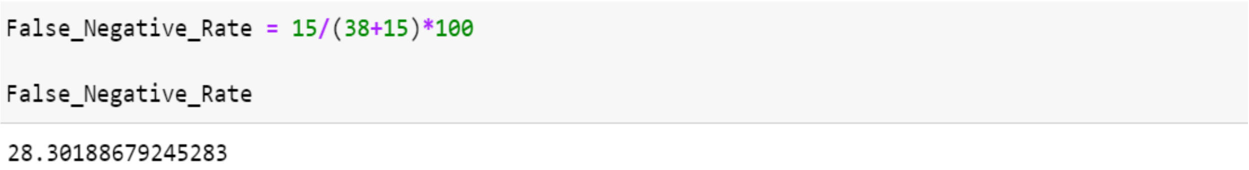
		PREDICTED	
		0	1
ACTUAL	0	TN: 100	FP: 10
	1	FN: 15	TP: 38

Fig.11. Confusion matrix - 1

In the fig.11 of confusion matrix, the value corresponding to our problem statement it means that (out of 163 samples, i.e. y_test)

- TN: True negative values, our model has correctly predicted 100 women as PCOS Negative.
- TP: True positive - the model has correctly predicted 38 women as PCOS positive.
- FN: False negative - the model has predicted 15 women who are PCOS Positive as PCOS Negative.
- FP: False positive - the model has predicted 10 women who are PCOS Negative as PCOS Positive.

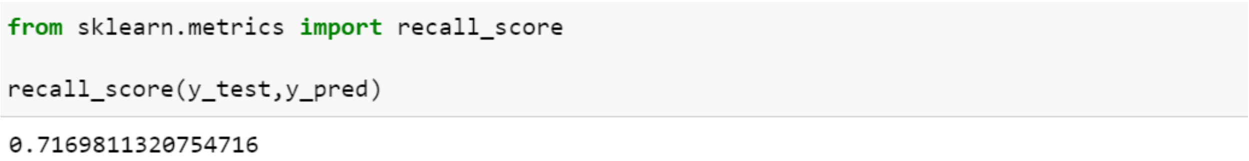
False negative rate



```
False_Negative_Rate = 15/(38+15)*100

False_Negative_Rate
```

```
28.30188679245283
```

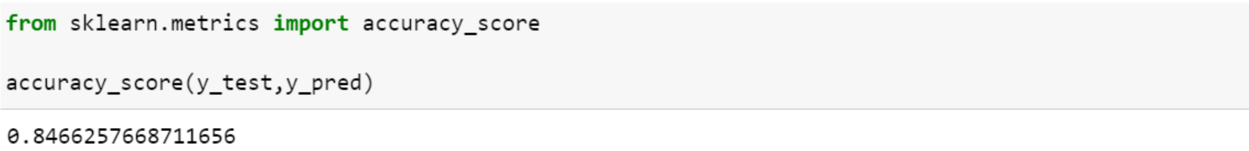
Recall score



```
from sklearn.metrics import recall_score

recall_score(y_test,y_pred)
```

```
0.7169811320754716
```

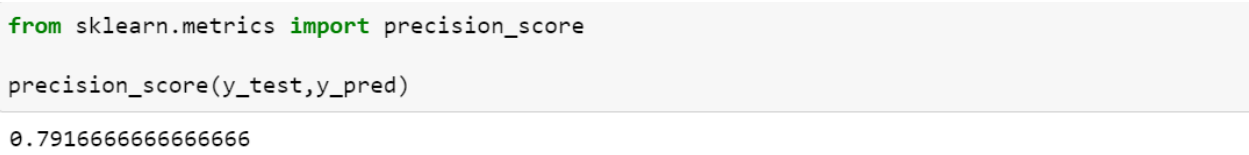
Accuracy score



```
from sklearn.metrics import accuracy_score

accuracy_score(y_test,y_pred)
```

```
0.8466257668711656
```

Precision score

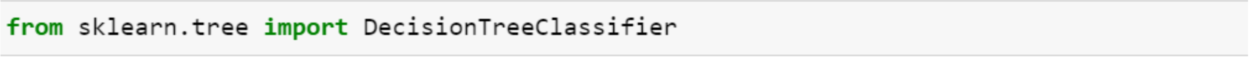


```
from sklearn.metrics import precision_score

precision_score(y_test,y_pred)
```

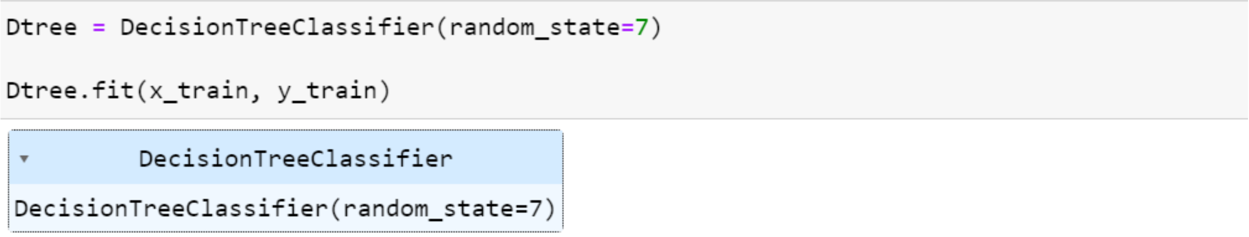
```
0.7916666666666666
```

5.3. Training, Testing and Evaluation – Decision Tree Classifier

5.3.1. Training the model



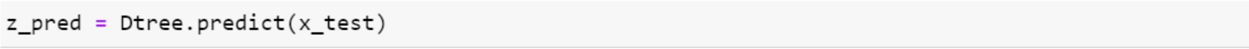
```
from sklearn.tree import DecisionTreeClassifier
```



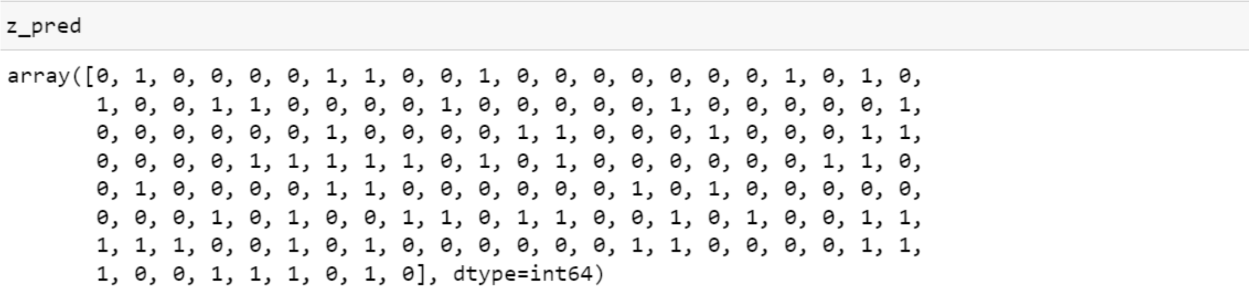
```
Dtree = DecisionTreeClassifier(random_state=7)

Dtree.fit(x_train, y_train)
```

```
▾        DecisionTreeClassifier
DecisionTreeClassifier(random_state=7)
```

5.3.2. Testing the model



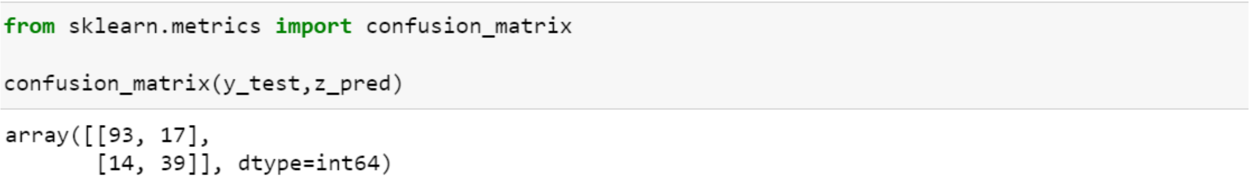
```
z_pred = Dtree.predict(x_test)
```



```
z_pred
```

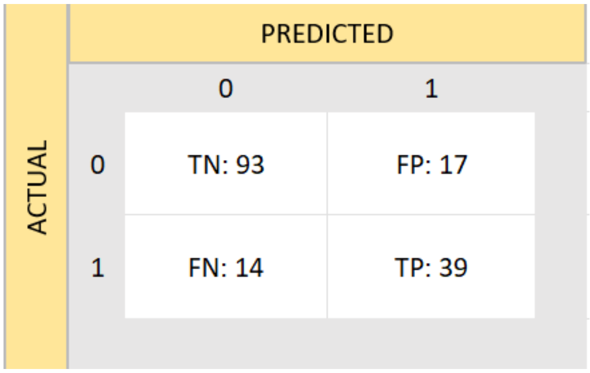
```
array([0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0,
       1, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1,
       0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1,
       0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0,
       0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0,
       0, 0, 0, 1, 0, 1, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1,
       1, 1, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1,
       1, 0, 0, 1, 1, 1, 0, 1, 0], dtype=int64)
```

5.3.3. Evaluating the model



```
from sklearn.metrics import confusion_matrix

confusion_matrix(y_test,z_pred)
```

```
array([[93, 17],
       [14, 39]], dtype=int64)
```

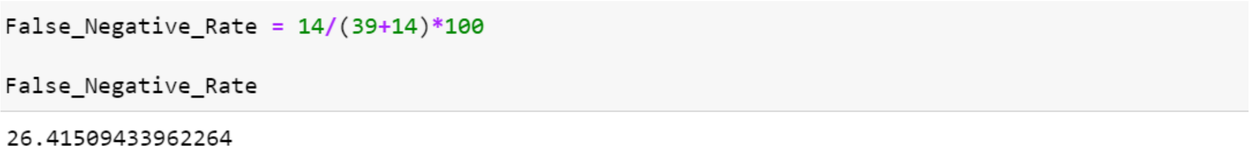


		PREDICTED	
		0	1
ACTUAL	0	TN: 93	FP: 17
	1	FN: 14	TP: 39

Fig.12. Confusion matrix - 2

- TN: the model has correctly predicted 93 women as PCOS Negative.
- TP: the model has correctly predicted 39 women as PCOS positive.
- FN: the model has predicted 14 women who are PCOS Positive as PCOS Negative.
- FP: the model has predicted 17 women who are PCOS Negative as PCOS Positive.

False negative rate

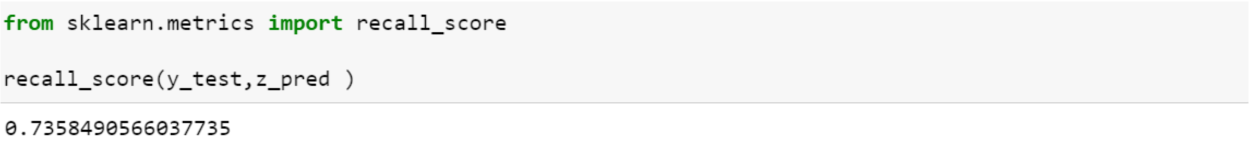


```
False_Negative_Rate = 14/(39+14)*100

False_Negative_Rate
```

```
26.41509433962264
```

The False negative rate in the model using Decision Tree Classifer is comparitvely smaller.

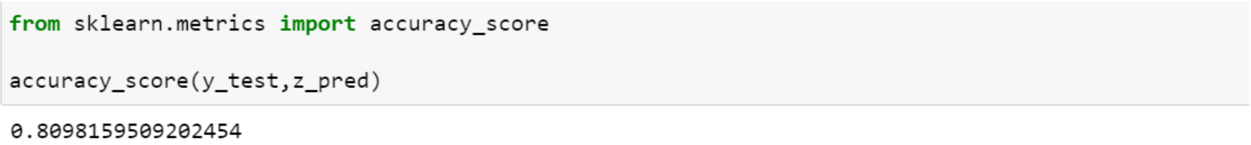
Recall score



```
from sklearn.metrics import recall_score

recall_score(y_test,z_pred )
```

```
0.7358490566037735
```

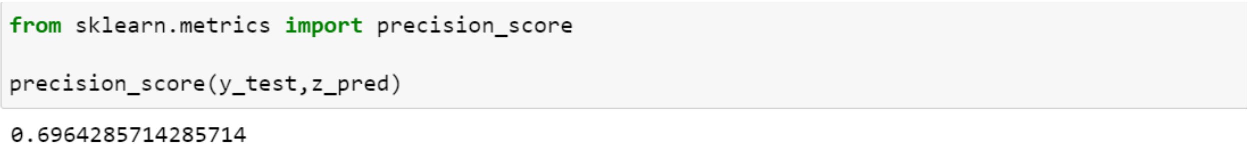
Accuracy score



```
from sklearn.metrics import accuracy_score

accuracy_score(y_test,z_pred)
```

```
0.8098159509202454
```

Precision score



```
from sklearn.metrics import precision_score

precision_score(y_test,z_pred)
```

```
0.6964285714285714
```

6. CONCLUSION AND RECOMMENDATIONS

6.1. Conclusion on evaluation metrics

Now that we have successfully developed a model with different machine learning algorithms, we can now compare these models based on their performance metrics.

False negative rate: metric plays a key role in model evaluation when it comes to medical datasets, False negative rate emphasizes on the percentage of samples which are falsely predicted as Negative.

Here we can see that both our models Linear regression and Decision Tree Classifier have a False negative rate of 28.30% and 26.41% , research Ferrer-Urbina et al. (2023) shows that the accepted threshold for false negatives is 25% and hence this could prove to be risky especially when working with the medical data, these women who have been falsely detected to not have PCOS are at risk and hence might miss out on necessary treatments for PCOS.

Accuracy, Recall and Precision evaluation metrics: The decision tree classifier model performed better on both the other metrics except on the precision metric.

6.2. Recommendation

Acknowledging the fact that every model should address the need of the business or the problem statement, in our case it would be ideal to further tune the deploy the model which was based on the Decision tree classifier because:

The decision tree classifier model performed well on the metrics that have a higher relevance to our problem statement of disease prediction (i.e. predicting PCOS in women).

BIBLIOGRAPHY

Alagar. (2023, November 10). The role of predictive analytics in decision making. *IABAC®*. https://iabac.org/blog/the-role-of-predictive-analytics-in-decision-making

Polycystic ovary Syndrome (PCOS). (2022, February 28). Johns Hopkins Medicine. https://www.hopkinsmedicine.org/health/conditions-and-diseases/polycystic-ovary-syndrome-pcos

Imbalanced data. (n.d.). Google for Developers. https://developers.google.com/machine-learning/data-prep/construct/sampling-splitting/imbalanced-data

Having a baby after age 35: How aging affects fertility and pregnancy. (n.d.). ACOG. https://www.acog.org/womens-health/faqs/having-a-baby-after-age-35-how-aging-affects-fertility-and-pregnancy#:~:text=A%20woman's%20peak%20reproductive%20years,getting%20pregnant%20naturally%20is%20unlikely.

Johansson, J., & Stener-Victorin, E. (2013). Polycystic ovary Syndrome: Effect and mechanisms of acupuncture for ovulation induction. *Evidence-based Complementary and Alternative Medicine*, *2013*, 1–16. https://doi.org/10.1155/2013/762615

Giant, C., & Giant, C. (2022, June 22). Fasting vs Non-Fasting: Glucose Test Results - Speedy Sticks | Mobile Medical Services. *Speedy Sticks | Mobile Medical Services - Mobile Phlebotomy, On site Healthcare, Home Lab*. https://www.speedysticks.com/blog/fasting-versus-nonfasting-glucose/

Ferrer-Urbina, R., Pardo, A., Arrindell, W. A., & Puddu-Gallardo, G. (2023). Comparison of false positive and false negative rates of two indices of individual reliable change: Jacobson-Truax and Hageman-Arrindell methods. *Frontiers in Psychology, 14*. https://doi.org/10.3389/fpsyg.2023.1132128