

Part A - Choosing a Data Set The dataset I am choosing to work is PCOS dataset

PCOS = Polycystic Ovary Syndrome.

PCOS is one of the emerging diseases that is seen almost in every woman today due to our lifestyle, stress, anxiety, and malfunctioned food products. It is a complex condition where ovaries produce out of control androgens, i.e., male sex hormones which is present in women in usually small amount with no specific treatment or medication. When a woman is diagnosed with PCOS, it means that numerous small cysts have developed in the ovary hence disrupting normal ovary functions.

These days, women when diagnosed with PCOS, they do not share it with their family and it's a taboo disease that comes with a tag that a women cannot get pregnant when she is diagnosed with PCOS. PCOS is never discussed a lot and there is no proper medication to cure this condition. This condition results into high mood fluctuations, intense stress, weight gain, hair loss, acne, disrupted reproductive system, anxiety, skin patches, thyroid etc.

I am specifically choosing this dataset as this condition is more serious than it is described, and it cannot be diagnosed at early stage as of now. It becomes very important to diagnose PCOS at early stage to avoid the misconceptions about the conditions and for the proper treatment.

PCOS are of 4 types:

1) Insulin Resistant - High level on insulin drives up androgen levels and can develop Diabetes 2) Adrenal PCOS - Excess Adrenal hormones causing infertility, most difficult to be diagnosed 3) Inflammatory PCOS - Excess testosterone resulting in ovulation issues 4) After pill PCOS - This occurs in some people after they stop taking oral contraceptive pills

Feature selection becomes the most crucial part here as there are thousands of features available in the dataset and no one of them guarantees the successful diagnosis of PCOS. Here I will be using 4 different Machine learning algorithms:

1)Decision Tree 2)Logistic Regression 3)KNN 4)Random Forest

Hence my reason for choosing this dataset is to further extend my analysis by diagnosing the type of PCOS a patient is suffering from, hence avoiding future complications.

Currently, as per - the analysis is done only to detect PCOS.

The next target would be to determine the type of PCOS from its symptoms to help patient take precautions accordingly.

Part B - Obtain Data To proceed with our dataset, data needs to go through couple of steps here.

The data is downloaded from Kaggle, and the data is collected from 10 different hospitals in India.

The data downloaded is already in the CSV (Comma separated value format).

I am working on Anaconda Jupyter notebook. I downloaded Data from Kaggle which was already in .csv format and uploaded it to Jupyter notebook and assigned to 'data' variable to read CSV file in data frame.

I have already cleaned my data in excel and dropped columns such as - Sr no, Patient file, BMI, Hip, Waist, Hip:Waist Ratio. I cleaned up the extra space or any null values in excel itself.

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import time
from sklearn.metrics import confusion_matrix

#importing split training & Testing
from sklearn.model_selection import train_test_split
from sklearn import preprocessing

#importing classifiers
from sklearn.ensemble import RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.datasets import make_classification

#importing report function
from sklearn.metrics import precision_recall_fscore_support
from sklearn.metrics import classification_report
import sklearn.metrics as metrics

#reading of csv file to dataframe
df = pd.read_csv('/content/pcos.csv')
```

```
#Using head method to look for 1st few entries in the dataset and printing the data
df.head()
```



	Sl. No	Patient File No.	PCOS (Y/N)	Age (yrs)	Weight (Kg)	Height(Cm)	BMI	Blood Group	Pulse rate(bpm)	(breaths/min)	RR	...	Fast food (Y/N)	Reg.Exercise(Y/N)	BP _Systolic (mmHg)	_Diast (mmHg)
0	1	1	0	28	44.6	152.0	19.3	15	78	22	...	1.0	0	110		
1	2	2	0	36	65.0	161.5	24.9	15	74	20	...	0.0	0	120		
2	3	3	1	33	68.8	165.0	25.3	11	72	18	...	1.0	0	120		
3	4	4	0	37	65.0	148.0	29.7	13	72	20	...	0.0	0	120		
4	5	5	0	25	52.0	161.0	20.1	11	72	18	...	0.0	0	120		

5 rows × 17 columns

```
#Part C - Scrubbing and Formatting
```

```
#Data Analyzing becomes crucial step here to check for duplicates or missing values in the data. The data as of now contains 540 samples and
# Removing the Nan and infinite values:
#Displaying data with all features and the respective 5 rows
df.head().T
```



	0	1	2	3	4
SI. No	1	2	3	4	5
Patient File No.	1	2	3	4	5
PCOS (Y/N)	0	0	1	0	0
Age (yrs)	28	36	33	37	25
Weight (Kg)	44.6	65.0	68.8	65.0	52.0
Height(Cm)	152.0	161.5	165.0	148.0	161.0
BMI	19.3	24.9	25.3	29.7	20.1
Blood Group	15	15	11	13	11
Pulse rate(bpm)	78	74	72	72	72
RR (breaths/min)	22	20	18	20	18
Hb(g/dl)	10.48	11.7	11.8	12.0	10.0
Cycle(R/I)	2	2	2	2	2
Cycle length(days)	5	5	5	5	5
Marriage Status (Yrs)	7.0	11.0	10.0	4.0	1.0
Pregnant(Y/N)	0	1	1	0	1
No. of abortions	0	0	0	0	0
I beta-HCG(mIU/mL)	1.99	60.8	494.08	1.99	801.45
II beta-HCG(mIU/mL)	1.99	1.99	494.08	1.99	801.45
FSH(mIU/mL)	7.95	6.73	5.54	8.06	3.98
LH(mIU/mL)	3.68	1.09	0.88	2.36	0.9
FSH/LH	2.16	6.17	6.3	3.42	4.42
Hip(inch)	36	38	40	42	37
Waist(inch)	30	32	36	36	30
Waist:Hip Ratio	0.83	0.84	0.9	0.86	0.81
TSH (mIU/L)	0.68	3.16	2.54	16.41	3.57
AMH(ng/mL)	2.07	1.53	6.63	1.22	2.26
PRL(ng/mL)	45.16	20.09	10.52	36.9	30.09
Vit D3 (ng/mL)	17.1	61.3	49.7	33.4	43.8
PRG(ng/mL)	0.57	0.97	0.36	0.36	0.38
RBS(mg/dl)	92.0	92.0	84.0	76.0	84.0
Weight gain(Y/N)	0	0	0	0	0
hair growth(Y/N)	0	0	0	0	0
Skin darkening (Y/N)	0	0	0	0	0
Hair loss(Y/N)	0	0	1	0	1
Pimples(Y/N)	0	0	1	0	0
Fast food (Y/N)	1.0	0.0	1.0	0.0	0.0
Reg.Exercise(Y/N)	0	0	0	0	0
BP _Systolic (mmHg)	110	120	120	120	120
BP _Diastolic (mmHg)	80	70	80	70	80
Follicle No. (L)	3	3	13	2	3
Follicle No. (R)	3	5	15	2	4
Avg. F size (L) (mm)	18.0	15.0	18.0	15.0	16.0
Avg. F size (R) (mm)	18.0	14.0	20.0	14.0	14.0
Endometrium (mm)	8.5	3.7	10.0	7.5	7.0
Unnamed: 44	NaN	NaN	NaN	NaN	NaN

```
#Getting all the columns
```

```
col = df.columns
```

```
print(col)
```

```
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 abortions',
      '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')
```

```
print(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 abortions',
      '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')
```

```
# Print the actual column names to identify discrepancies
```

```
print(df.columns.tolist())
```

```
['Sl. No.', 'Patient File No.', 'PCOS (Y/N)', 'Age (yrs)', 'Weight (Kg)', 'Height(Cm)', 'BMI', 'Blood Group', 'Pulse rate(bpm)', 'RR (bre
```

```
drop_cols = ['PCOS (Y/N)', 'Marraige Status (Yrs)', 'Height(Cm)', 'Pulse rate(bpm)', 'Hip(inch)', 'Waist(inch)']
```

```
X = df.drop(drop_cols, axis=1)
```

```
Y = df["PCOS (Y/N)"]
```

```
print(X.columns)
```

```
Index(['Sl. No.', 'Patient File No.', 'Age (yrs)', 'Weight (Kg)', 'BMI',
      'Blood Group', 'RR (breaths/min)', 'Hb(g/dl)', 'Cycle(R/I)',
      'Cycle length(days)', 'Pregnant(Y/N)', 'No. of abortions',
      'I beta-HCG(mIU/mL)', 'II beta-HCG(mIU/mL)', 'FSH(mIU/mL)',
      'LH(mIU/mL)', 'FSH/LH', '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')
```

```
X
```



	Sl. No	Patient File No.	Age (yrs)	Weight (Kg)	BMI	Blood Group	RR (breaths/min)	Hb(g/dl)	Cycle(R/I)	Cycle length(days)	...	Fast food (Y/N)	Reg.Exercise(Y/N)	BF _Systolic (mmHg)
0	1	1	28	44.6	19.3	15	22	10.48	2	5	...	1.0	0	110
1	2	2	36	65.0	24.9	15	20	11.70	2	5	...	0.0	0	120
2	3	3	33	68.8	25.3	11	18	11.80	2	5	...	1.0	0	120
3	4	4	37	65.0	29.7	13	20	12.00	2	5	...	0.0	0	120
4	5	5	25	52.0	20.1	11	18	10.00	2	5	...	0.0	0	120
...
536	537	537	35	50.0	18.5	17	16	11.00	2	5	...	0.0	0	110
537	538	538	30	63.2	25.3	15	18	10.80	2	5	...	0.0	0	110
538	539	539	36	54.0	23.4	13	20	10.80	2	6	...	0.0	0	110
539	540	540	27	50.0	22.2	15	20	12.00	4	2	...	0.0	0	110
540	541	541	23	82.0	30.1	13	20	10.20	4	7	...	1.0	0	120

541 rows × 39 columns

Part D - Exploratory Data Analysis Using info () function, data type of the features is extracted. Here I have 541 instances and 33 features. Datatypes I have with my data is float64, int 64 and object.

X.info()



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541 entries, 0 to 540
Data columns (total 39 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Sl. No                                541 non-null    int64
1   Patient File No.                     541 non-null    int64
2   Age (yrs)                            541 non-null    int64
3   Weight (Kg)                          541 non-null    float64
4   BMI                                  541 non-null    float64
5   Blood Group                          541 non-null    int64
6   RR (breaths/min)                    541 non-null    int64
7   Hb(g/dl)                            541 non-null    float64
8   Cycle(R/I)                          541 non-null    int64
9   Cycle length(days)                  541 non-null    int64
10  Pregnant(Y/N)                       541 non-null    int64
11  No. of abortions                     541 non-null    int64
12  I   beta-HCG(mIU/mL)                 541 non-null    float64
13  II  beta-HCG(mIU/mL)                 541 non-null    object
14  FSH(mIU/mL)                         541 non-null    float64
15  LH(mIU/mL)                         541 non-null    float64
16  FSH/LH                             541 non-null    float64
17  Waist:Hip Ratio                     541 non-null    float64
18  TSH (mIU/L)                        541 non-null    float64
19  AMH(ng/mL)                         541 non-null    object
20  PRL(ng/mL)                         541 non-null    float64
21  Vit D3 (ng/mL)                     541 non-null    float64
22  PRG(ng/mL)                         541 non-null    float64
23  RBS(mg/dl)                         541 non-null    float64
24  Weight gain(Y/N)                   541 non-null    int64
25  hair growth(Y/N)                   541 non-null    int64
26  Skin darkening (Y/N)               541 non-null    int64
27  Hair loss(Y/N)                     541 non-null    int64
28  Pimples(Y/N)                       541 non-null    int64
29  Fast food (Y/N)                     540 non-null    float64
30  Reg.Exercise(Y/N)                   541 non-null    int64
31  BP _Systolic (mmHg)                 541 non-null    int64
32  BP _Diastolic (mmHg)                 541 non-null    int64
33  Follicle No. (L)                     541 non-null    int64
34  Follicle No. (R)                     541 non-null    int64
35  Avg. F size (L) (mm)                 541 non-null    float64
36  Avg. F size (R) (mm)                 541 non-null    float64
37  Endometrium (mm)                     541 non-null    float64
38  Unnamed: 44                          2 non-null     object
dtypes: float64(17), int64(19), object(3)
memory usage: 165.0+ KB
```

Performing 5 number data for numeric data using describe () function

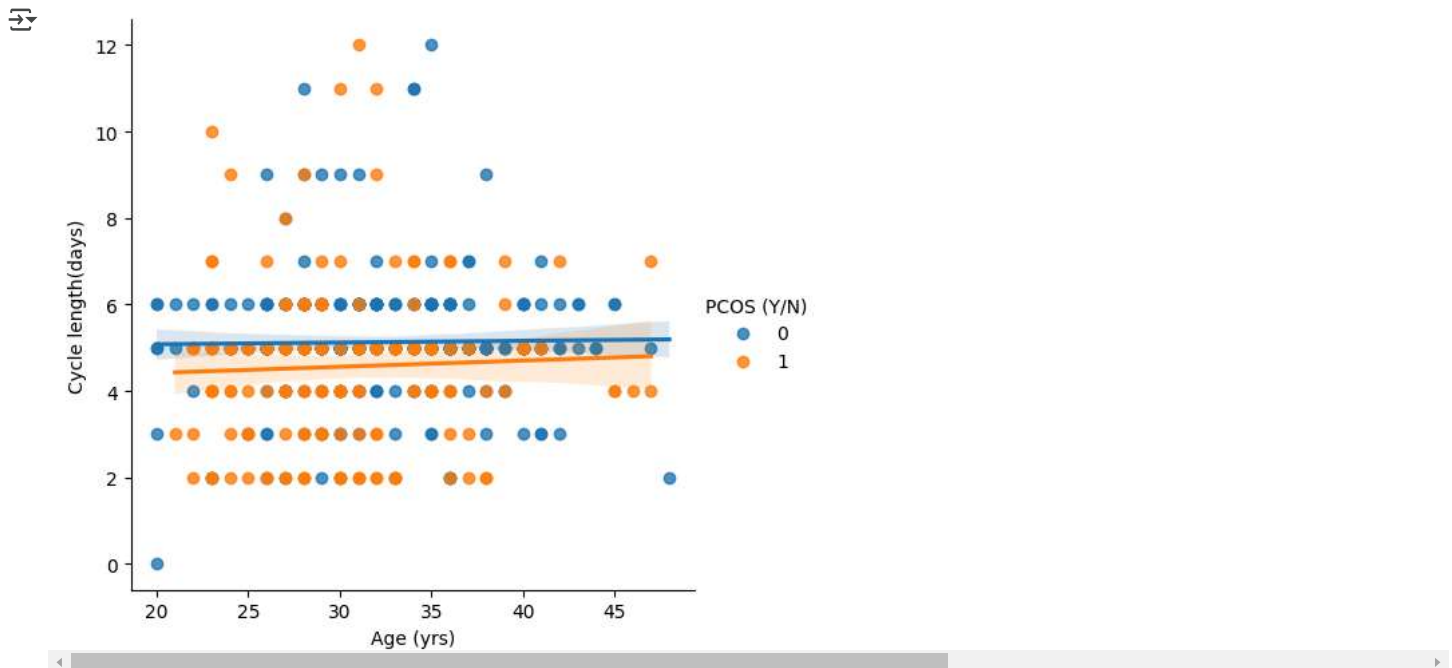
```
X.describe()
```

	Sl. No	Patient File No.	Age (yrs)	Weight (Kg)	BMI	Blood Group	RR (breaths/min)	Hb(g/dl)	Cycle(R/I)	Cycle length(days)	...	Pin
count	541.000000	541.000000	541.000000	541.000000	541.000000	541.000000	541.000000	541.000000	541.000000	541.000000	...	:
mean	271.000000	271.000000	31.430684	59.637153	24.307579	13.802218	19.243993	11.160037	2.560074	4.94085	...	
std	156.317519	156.317519	5.411006	11.028287	4.055129	1.840812	1.688629	0.866904	0.901950	1.49202	...	
min	1.000000	1.000000	20.000000	31.000000	12.400000	11.000000	16.000000	8.500000	2.000000	0.00000	...	
25%	136.000000	136.000000	28.000000	52.000000	21.600000	13.000000	18.000000	10.500000	2.000000	4.00000	...	
50%	271.000000	271.000000	31.000000	59.000000	24.200000	14.000000	18.000000	11.000000	2.000000	5.00000	...	
75%	406.000000	406.000000	35.000000	65.000000	26.600000	15.000000	20.000000	11.700000	4.000000	5.00000	...	
max	541.000000	541.000000	48.000000	108.000000	38.900000	18.000000	28.000000	14.800000	5.000000	12.00000	...	

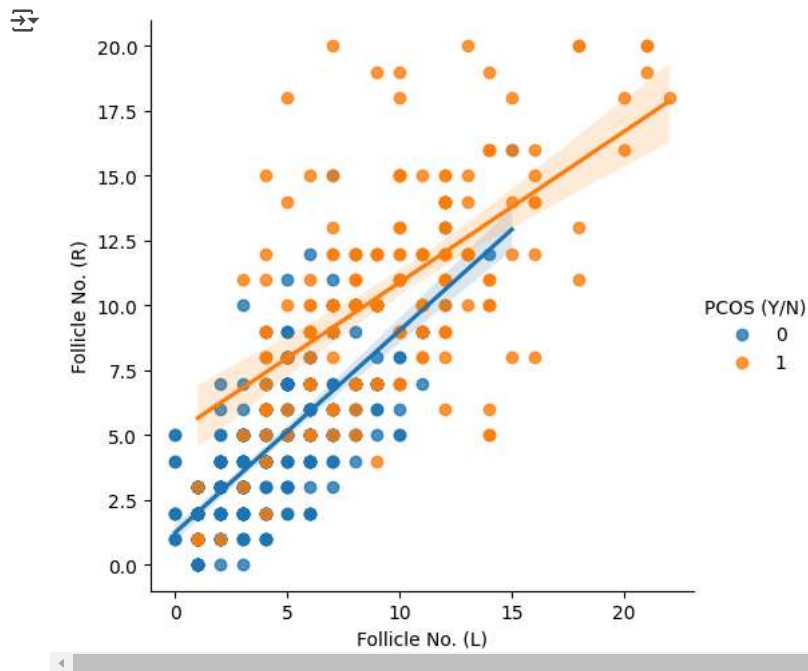
8 rows × 36 columns

Here are my findings when plotting features against each other (scatter). Here I am using seaborn library to plot features.

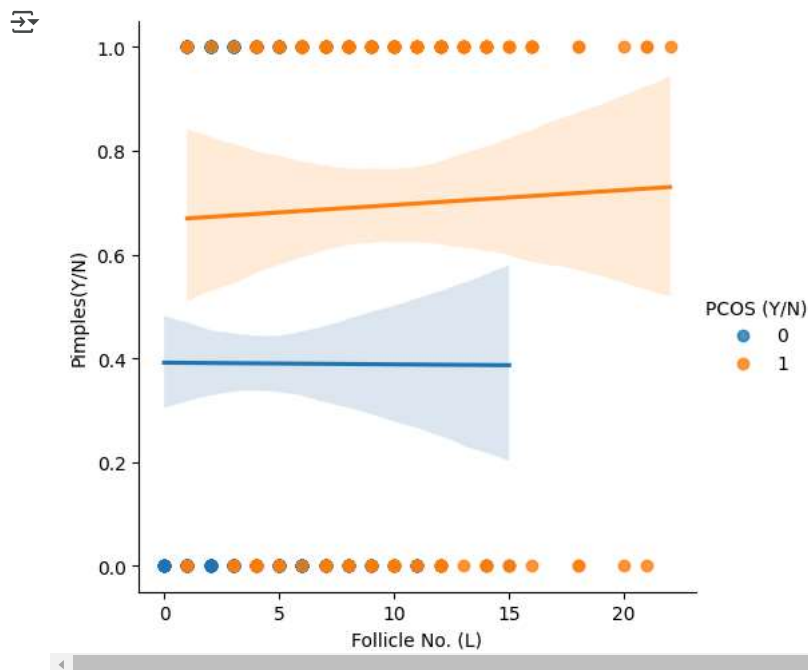
```
fig=sns.lmplot(data=df,x="Age (yrs)",y="Cycle length(days)", hue="PCOS (Y/N)")
plt.show(fig)
```



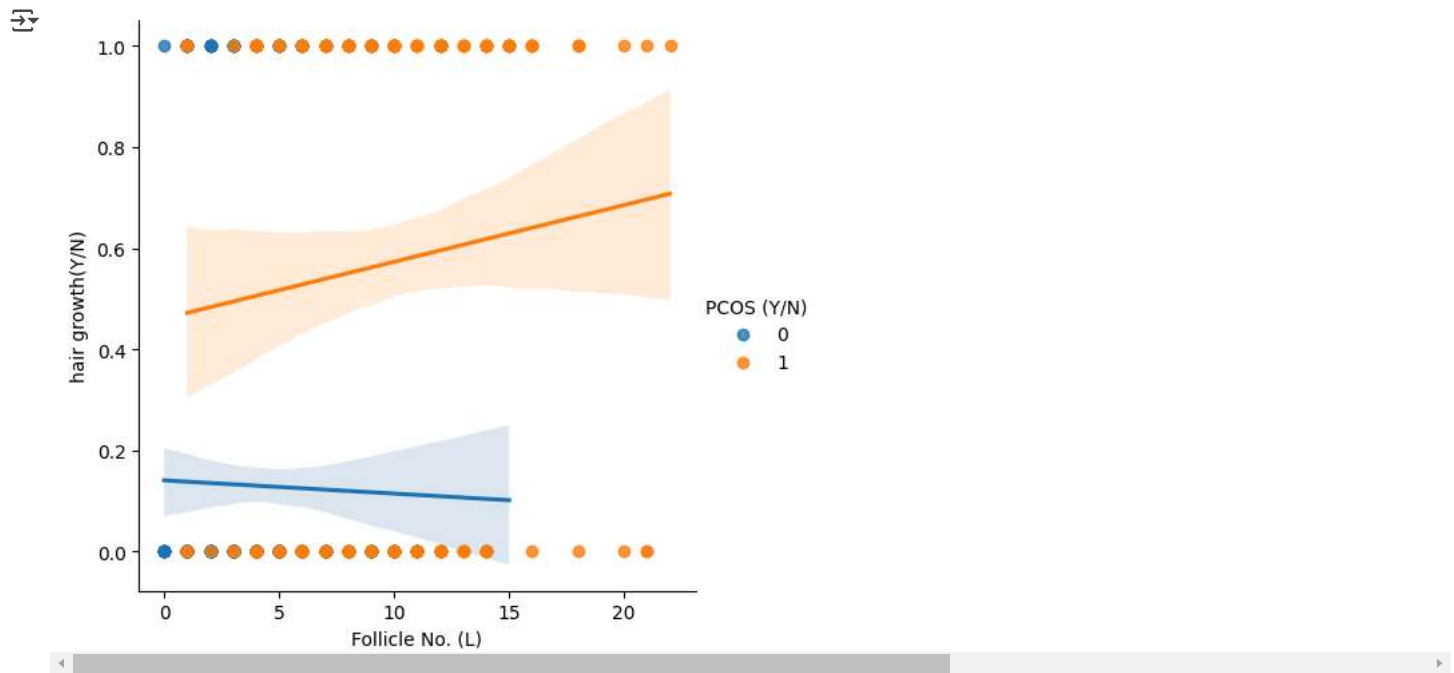
```
fig=sns.lmplot(data=df,x="Follicle No. (L)",y="Follicle No. (R)", hue="PCOS (Y/N)")
plt.show(fig)
```



```
fig=sns.lmplot(data=df,x="Follicle No. (L)",y="Pimples(Y/N)", hue="PCOS (Y/N)")
plt.show(fig)
```




```
fig=sns.lmplot(data=df,x="Follicle No. (L)",y="hair growth(Y/N)", hue="PCOS (Y/N)")
plt.show(fig)
```

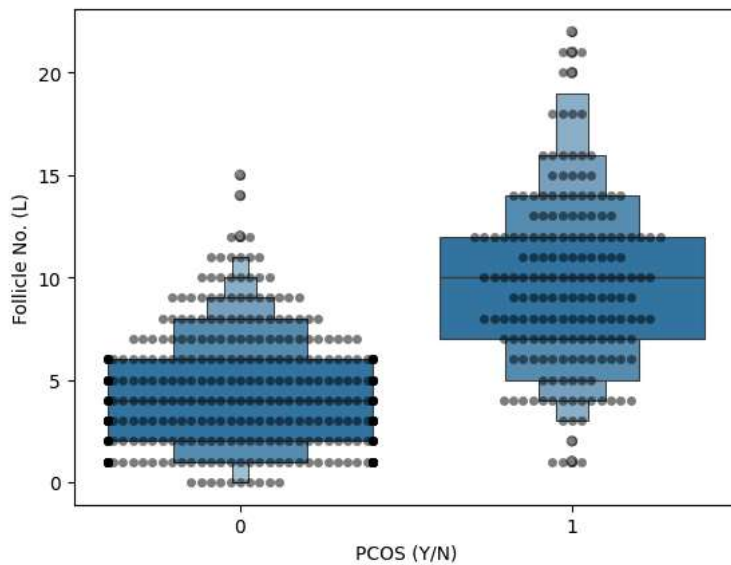


```

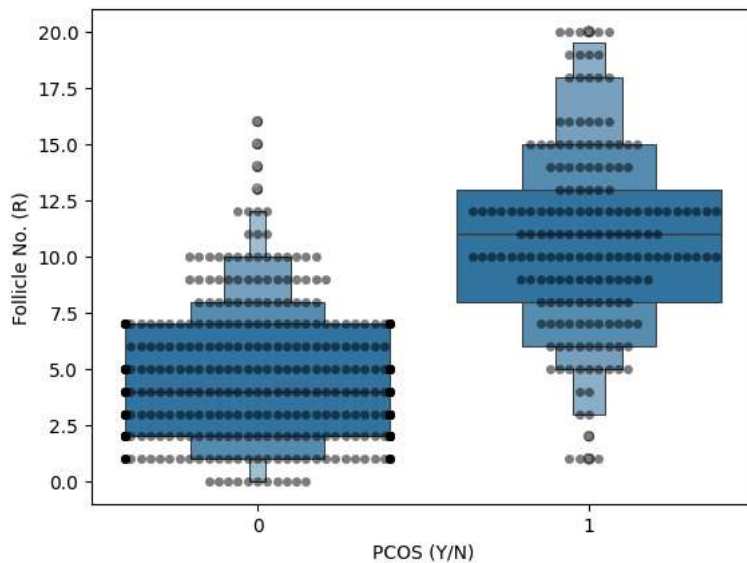
features = ["Follicle No. (L)","Follicle No. (R)"]
for i in features:
    sns.swarmplot(x=df["PCOS (Y/N)"], y=df[i], color="black", alpha=0.5 )
    sns.boxenplot(x=df["PCOS (Y/N)"], y=df[i])
plt.show()

```


 /usr/local/lib/python3.11/dist-packages/seaborn/categorical.py:3399: UserWarning: 32.4% of the points cannot be placed; you may want to warnings.warn(msg, UserWarning)



/usr/local/lib/python3.11/dist-packages/seaborn/categorical.py:3399: UserWarning: 30.5% of the points cannot be placed; you may want to warnings.warn(msg, UserWarning)

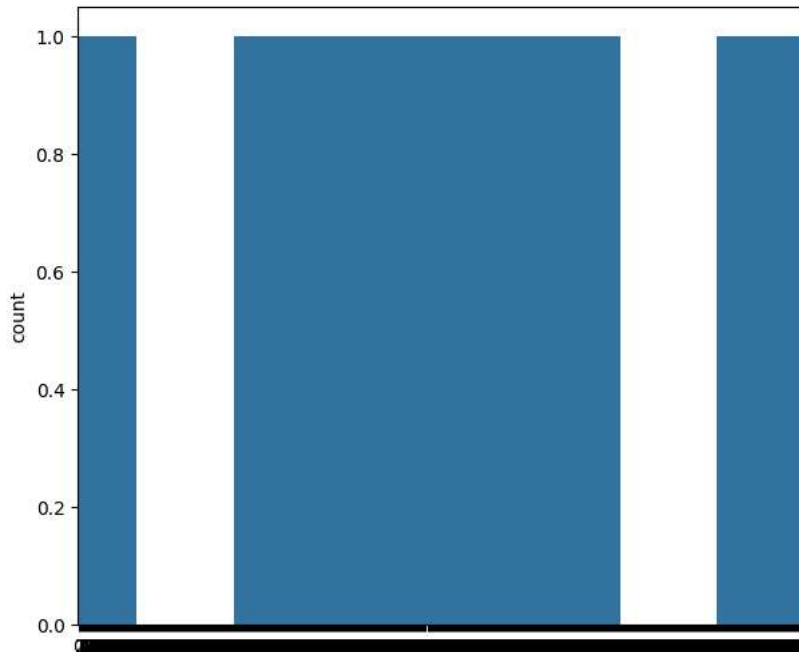


```
target = df['PCOS (Y/N)']
df.drop('PCOS (Y/N)',axis=1,inplace=True)
```

```
plt.figure(figsize=(7,6))
sns.countplot(target)
plt.title('Data imbalance')
plt.show()
```



Data imbalance



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

# Assuming df is your DataFrame

# Convert all columns to numeric, forcing errors to NaN
df = df.apply(pd.to_numeric, errors='coerce')

# Optionally, you can check for any remaining non-numeric values
print(df.isnull().sum()) # Check for NaN values

# Generate the correlation matrix
corrmat = df.corr()

# Set the size of the figure
plt.subplots(figsize=(18, 18))

# Create the heatmap
sns.heatmap(corrmat, cmap="coolwarm", square=True, annot=True, fmt=".2f", linewidths=.5)

# Show the plot
plt.title('Correlation Matrix Heatmap')
plt.show()
```

Sl. No	0
Patient File No.	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 abortions	0
I beta-HCG(mIU/mL)	0
II beta-HCG(mIU/mL)	1
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)	1
PRL(ng/mL)	0