In [1]:
```
# lets start with importing neccesary libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import Ridge,Lasso,RidgeCV,LassoCV,ElasticNet,ElasticNetCV,LogisticRegression
from sklearn.model_selection import train_test_split
from statsmodels.stats.outliers_influence import variance_inflation_factor
from sklearn.metrics import accuracy_score,confusion_matrix,roc_curve,roc_auc_score
#import scikitplot as skl
sns.set()
```

In [2]:
```
data=pd.read_csv("G:\Downloads\Logistic-regression_final\Logistic-regression_final\diabetes.csv")
```

In [3]:
```
data.describe()
```

Out[3]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000
mean	3.845052	120.894531	69.105469	20.536458	79.799479	31.992578	0.471876	33.240885	0.348958
std	3.369578	31.972618	19.355807	15.952218	115.244002	7.884160	0.331329	11.760232	0.476951
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.078000	21.000000	0.000000
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.300000	0.243750	24.000000	0.000000
50%	3.000000	117.000000	72.000000	23.000000	30.500000	32.000000	0.372500	29.000000	0.000000
75%	6.000000	140.250000	80.000000	32.000000	127.250000	36.600000	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

In [4]:
```
data.columns
```

Out[4]:
```
Index(['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin',
       'BMI', 'DiabetesPedigreeFunction', 'Age', 'Outcome'],
      dtype='object')
```

In [5]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to 767
Data columns (total 9 columns):
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   Pregnancies               768 non-null    int64  
 1   Glucose                   768 non-null    int64  
 2   BloodPressure             768 non-null    int64  
 3   SkinThickness             768 non-null    int64  
 4   Insulin                   768 non-null    int64  
 5   BMI                       768 non-null    float64
 6   DiabetesPedigreeFunction  768 non-null    float64
 7   Age                       768 non-null    int64  
 8   Outcome                   768 non-null    int64  
dtypes: float64(2), int64(7)
memory usage: 54.1 KB
```

In [6]:
```
data.isnull()
```

Out[6]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...
763	False	False	False	False	False	False	False	False	False
764	False	False	False	False	False	False	False	False	False
765	False	False	False	False	False	False	False	False	False
766	False	False	False	False	False	False	False	False	False
767	False	False	False	False	False	False	False	False	False

768 rows × 9 columns

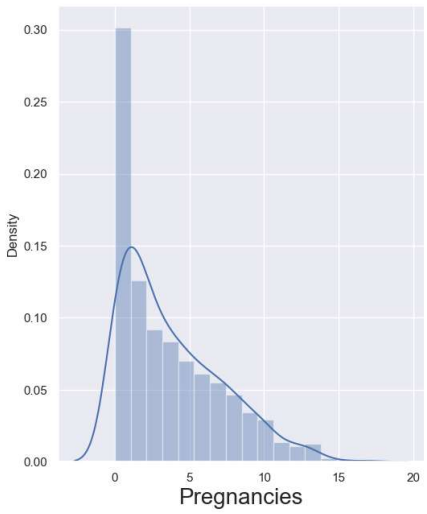
In [7]:
```
data.isnull().sum()
```

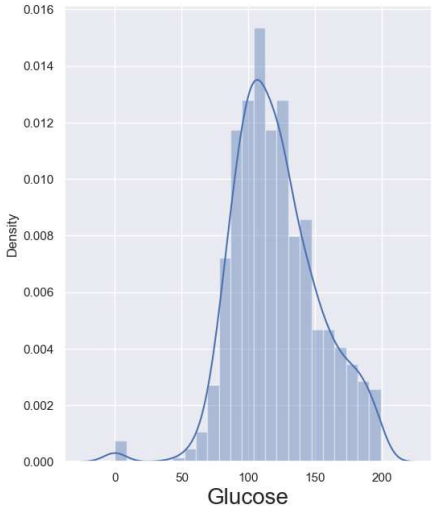
Out[7]:
```
Pregnancies                 0
Glucose                     0
BloodPressure               0
SkinThickness               0
Insulin                     0
BMI                         0
DiabetesPedigreeFunction    0
Age                         0
Outcome                     0
dtype: int64
```

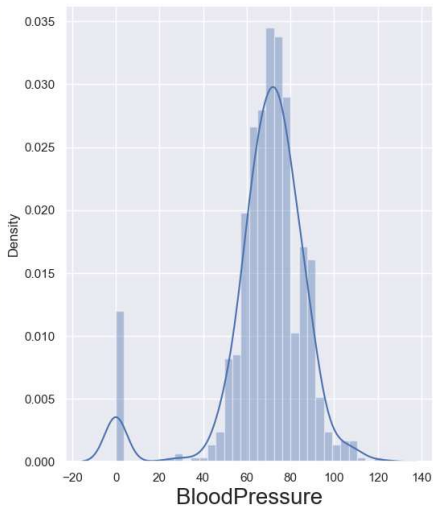
In [8]:

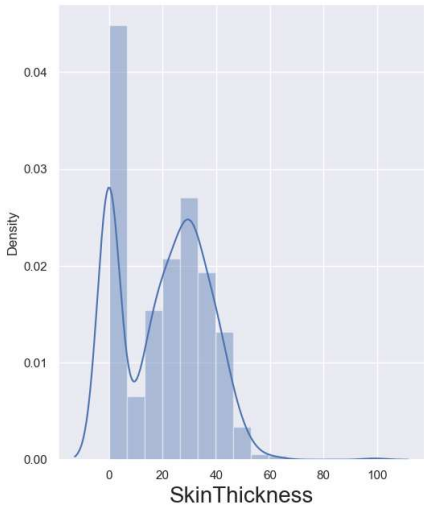
```python
# Lets see how data is distributed for every column
plt.figure(figsize=(20,25), facecolor='white')
plotnumber=1

for column in data:
    if plotnumber<=9:    #3 there are 9 columns in the data
        ax=plt.subplot(3,3,plotnumber)
        sns.distplot(data[column])
        plt.xlabel(column,fontsize=20)
        #plt.ylabel('salary',fontsize=20)
    plotnumber+=1
plt.show()
```

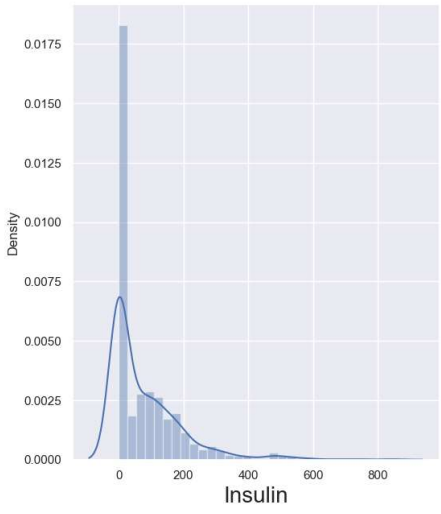
```
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

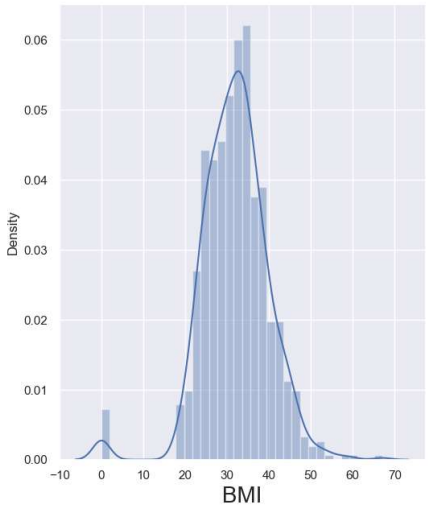

Density
Pregnancies

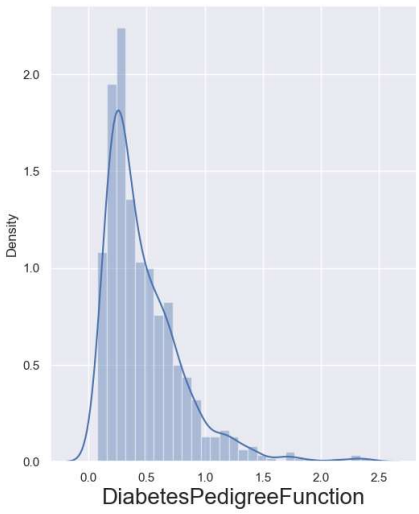

Density
Glucose

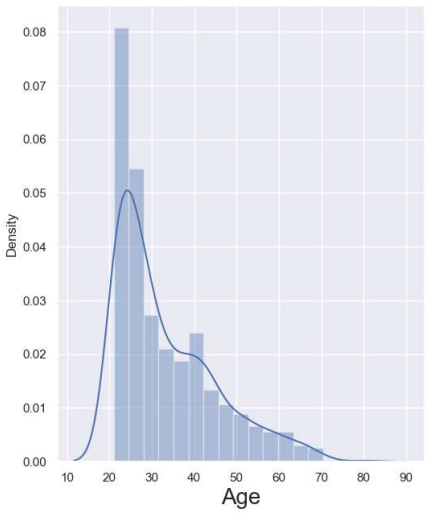

Density
BloodPressure

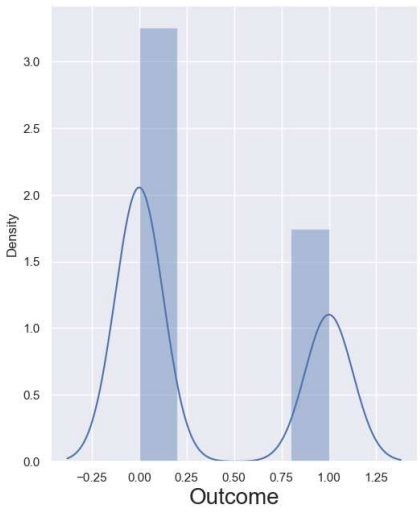

Density
SkinThickness


Density
Insulin


Density
BMI


Density
DiabetesPedigreeFunction

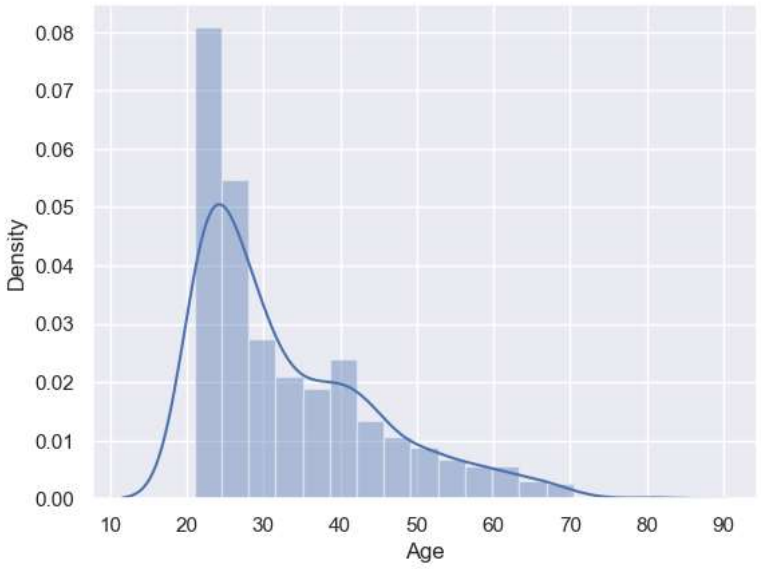

Density
Age


Density
Outcome

In [9]:
```
sns.distplot(data['Age'])
```

```
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[9]:
```
<AxesSubplot:xlabel='Age', ylabel='Density'>
```



we can see thete is some skewness in the data lets deal with the data

Also we can see there few data for columns Glucose.Insulin,skin thickness,BMI, and Blood Pressure which have value as 0.Thats not possible you can do a quick search to see that one cannot have 0 values for these.Lets deal with that.we can either remove such data or simply replace it with their respective mean value Lets do the latter

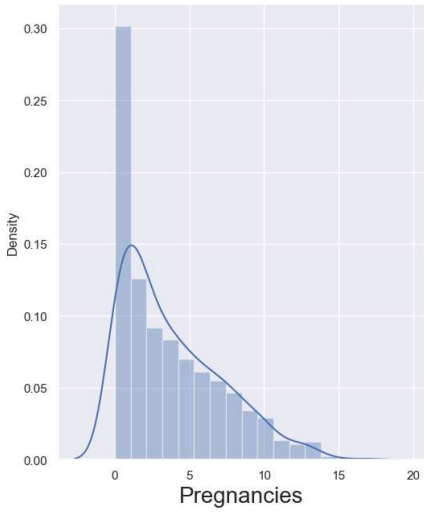
In [18]:
```
# replacing zero values with the mean of the column
data['BMI'] = data['BMI'].replace(0,data['BMI'].mean())
data['BloodPressure']=data['BloodPressure'].replace('BloodPressure').mean()
data['Glucose']=data['Glucose'].replace(0,data['Glucose'].mean())
data['Insulin']=data['Insulin'].replace(0,data['Insulin'].mean())
data['SkinThickness']=data['SkinThickness'].replace(0,data['SkinThickness'].mean())
```

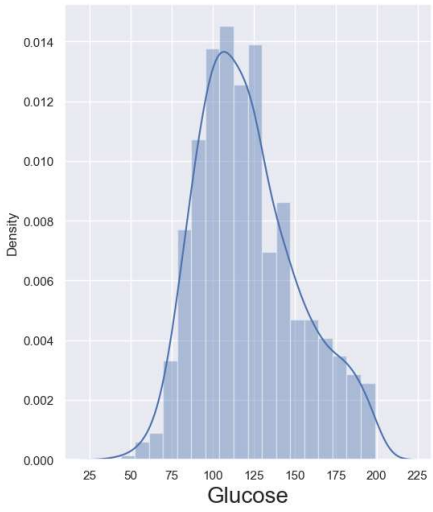
In [11]:
```
# calculate BMI of the presson
height/wieght

BMI=0
```

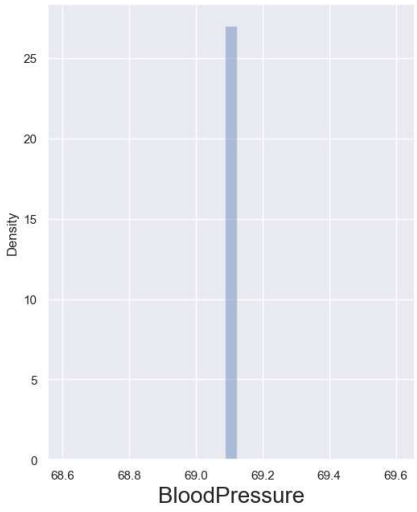
```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_11252\642500357.py in <module>
      1 # calculate BMI of the presson
----> 2 height/wieght
      3
      4 BMI=0

NameError: name 'height' is not defined
```

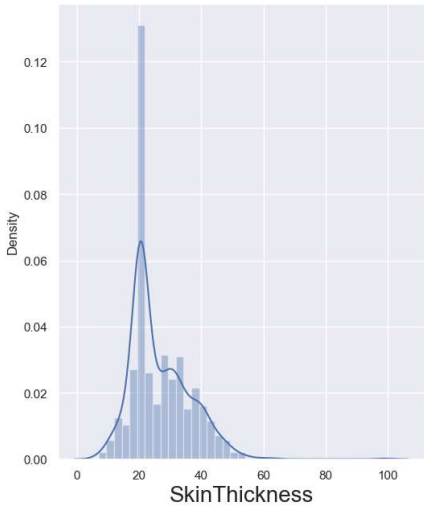
In [23]:
```python
# lets see how data is distributed for every column
plt.figure(figsize=(20,25), facecolor='white')
plotnumber=1

for column in data:
    if plotnumber<=9:
        ax = plt.subplot(3,3,plotnumber)
        sns.distplot(data[column])
        plt.xlabel(column,fontsize=20)
       #plt,ylabel('salary',fontsize=20)
    plotnumber += 1
plt.show()
```

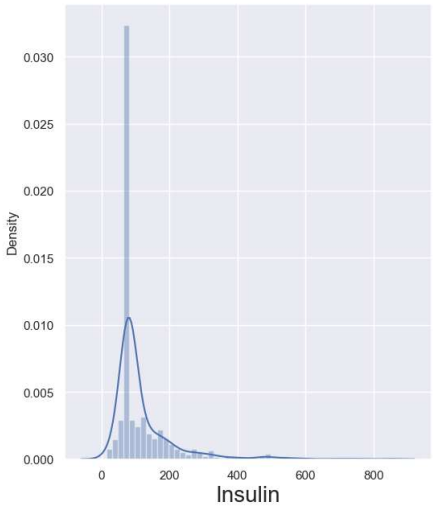
```
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:316: UserWarning: Dataset has 0 variance; skipping den
sity estimate. Pass `warn_singular=False` to disable this warning.
  warnings.warn(msg, UserWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

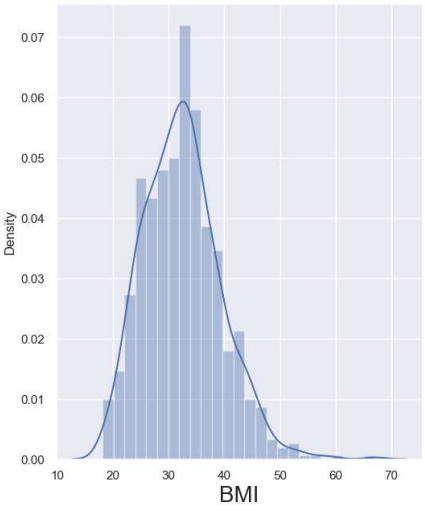

Density
Pregnancies

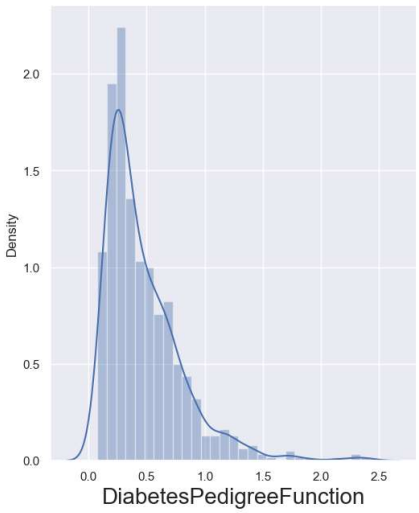

Density
Glucose

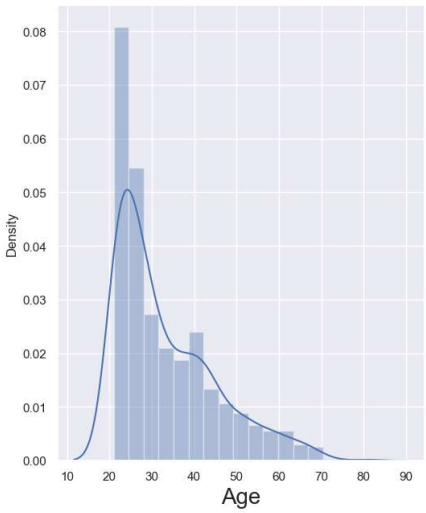

Density
BloodPressure

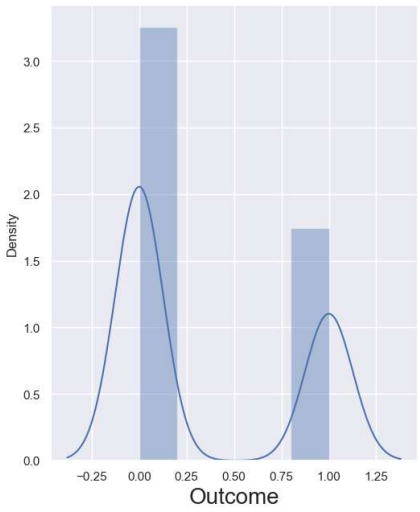

Density
SkinThickness


Density
Insulin


Density
BMI

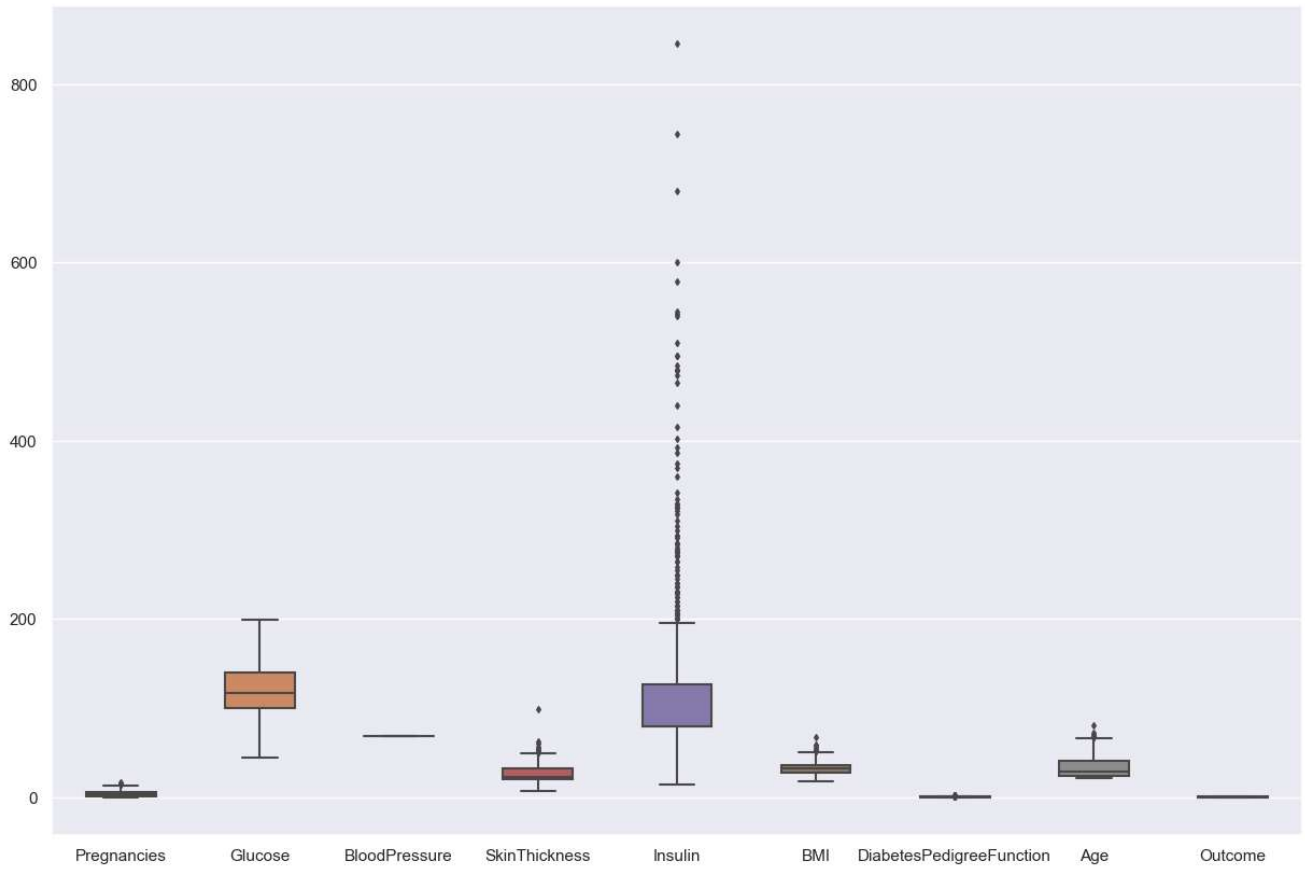

Density
DiabetesPedigreeFunction


Density
Age


Density
Outcome

In [20]:
```
fig, ax = plt.subplots(figsize=(15,10))
sns.boxplot(data=data,width=0.5, ax=ax, fliersize=3)
```

Out[20]: <AxesSubplot:>



In [24]:
```
data['Pregnancies'].shape
```

Out[24]: (768,)

In [25]:
```
data['Pregnancies'].quantile(0.98)
```

Out[25]: 12.0

In [27]:
```
q = data['Pregnancies'].quantile(0.98)
# we removing the top 2% data from the pregnancies column
data_cleaned = data[data['Pregnancies']<q]
```

In [28]:
```
q = data_cleaned = ['BMI'].quantile(0.99)
# we are removing the top 1% data from the BMI column
data_cleaned = data_cleaned[data_cleaned['BMI']<q]
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_11252\3478742158.py in <module>
----> 1 q = data_cleaned = ['BMI'].quantile(0.99)
      2 # we are removing the top 1% data from the BMI column
      3 data_cleaned = data_cleaned[data_cleaned['BMI']<q]

AttributeError: 'list' object has no attribute 'quantile'
```

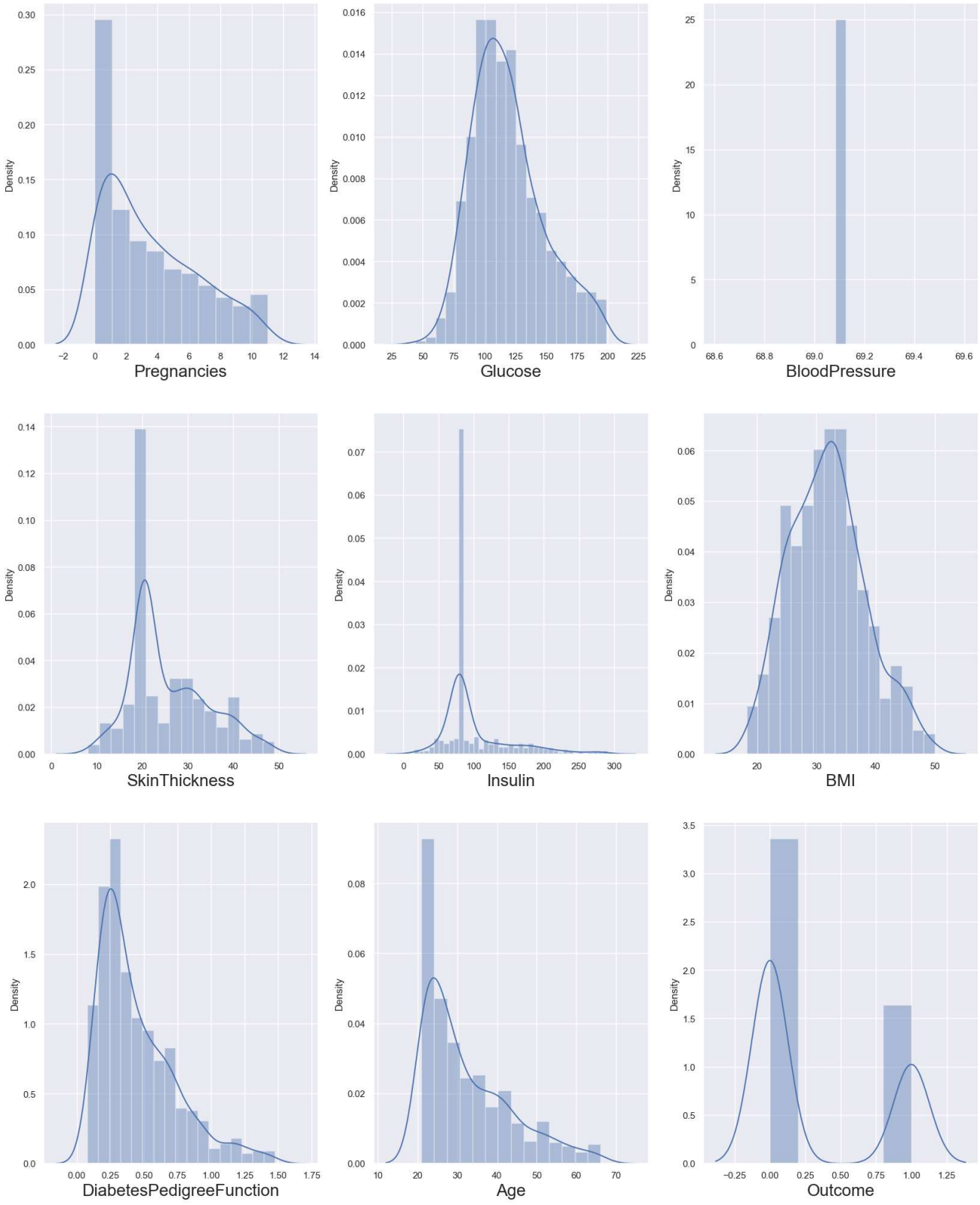
In [29]:
```
# we removing the top 2% data from the pregnancies column
data_cleaned = data[data['Pregnancies']<q]
q = data_cleaned = ['BMI'].quantile(0.99)
# we are removing the top 1% data from the BMI column
data_cleaned = data_cleaned[data_cleaned['BMI']<q]
q = data_cleaned['SkinThickness'].quantile(0.99)
# we are removing the top 1% data from the SkinThickness column
data_cleaned = data_cleaned[data_cleaned['SkinThickness']<q]
q = data_cleaned['Insulin'].quantile(0.95)
# we are removing the top 5% data from the Insulin column
data_cleaned = data_cleaned[data_cleaned['Insulin']<q]
q = data_cleaned['DiabetespedigreeFunction'].quantile(0.99)
# we are removing the top 1% data from the DiabetespedigreeFunction column column
data_cleaned = data_cleaned[data_cleaned['DiabetespedigreeFunction']<q]
q= data_cleaned['Age'].quantile(0.99)
# we are removing the top 1% data from the Age column
data_cleaned = data_cleaned[data_cleaned['Age']<q]
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_11252\4159511675.py in <module>
      1 # we removing the top 2% data from the pregnancies column
      2 data_cleaned = data[data['Pregnancies']<q]
----> 3 q = data_cleaned = ['BMI'].quantile(0.99)
      4 # we are removing the top 1% data from the BMI column
      5 data_cleaned = data_cleaned[data_cleaned['BMI']<q]

AttributeError: 'list' object has no attribute 'quantile'
```

In [30]:
```
q = data['Pregnancies'].quantile(0.98)
# we are removing the top 2% data from the Pregnancies column
data_cleaned = data[data['Pregnancies']<q]
q = data_cleaned['BMI'].quantile(0.99)
# we are removing the top 1% data from the BMI column
data_cleaned  = data_cleaned[data_cleaned['BMI']<q]
q = data_cleaned['SkinThickness'].quantile(0.99)
# we are removing the top 1% data from the SkinThickness column
data_cleaned  = data_cleaned[data_cleaned['SkinThickness']<q]
q = data_cleaned['Insulin'].quantile(0.95)
# we are removing the top 5% data from the Insulin column
data_cleaned  = data_cleaned[data_cleaned['Insulin']<q]
q = data_cleaned['DiabetesPedigreeFunction'].quantile(0.99)
# we are removing the top 1% data from the DiabetesPedigreeFunction column
data_cleaned  = data_cleaned[data_cleaned['DiabetesPedigreeFunction']<q]
q = data_cleaned['Age'].quantile(0.99)
# we are removing the top 1% data from the Age column
data_cleaned  = data_cleaned[data_cleaned['Age']<q]
```

In [31]:
```
# let's see how data is distributed for every column
plt.figure(figsize=(20,25), facecolor='white')
plotnumber = 1

for column in data_cleaned:
    if plotnumber<=9 :
        ax = plt.subplot(3,3,plotnumber)
        sns.distplot(data_cleaned[column])
        plt.xlabel(column,fontsize=20)
        #plt.ylabel('Salary',fontsize=20)
    plotnumber+=1
plt.show()
```

```
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:316: UserWarning: Dataset has 0 variance; skipping den
sity estimate. Pass `warn_singular=False` to disable this warning.
  warnings.warn(msg, UserWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



The data looks much better now than before we still start our analysis with this data now as we don't want to lose important information.if we dont want to lose important information.if our model doesnot work with accuracy we will come baack for more preprocessing

In [32]:
```
x = data.drop(columns = ['Outcome'])
y=data['Outcome']
```

Before we fit our data to a modelIt's visulaize the relatonship between our independent variables and the categories

In [33]:
```
data.shape
```

Out[33]:
```
(768, 9)
```

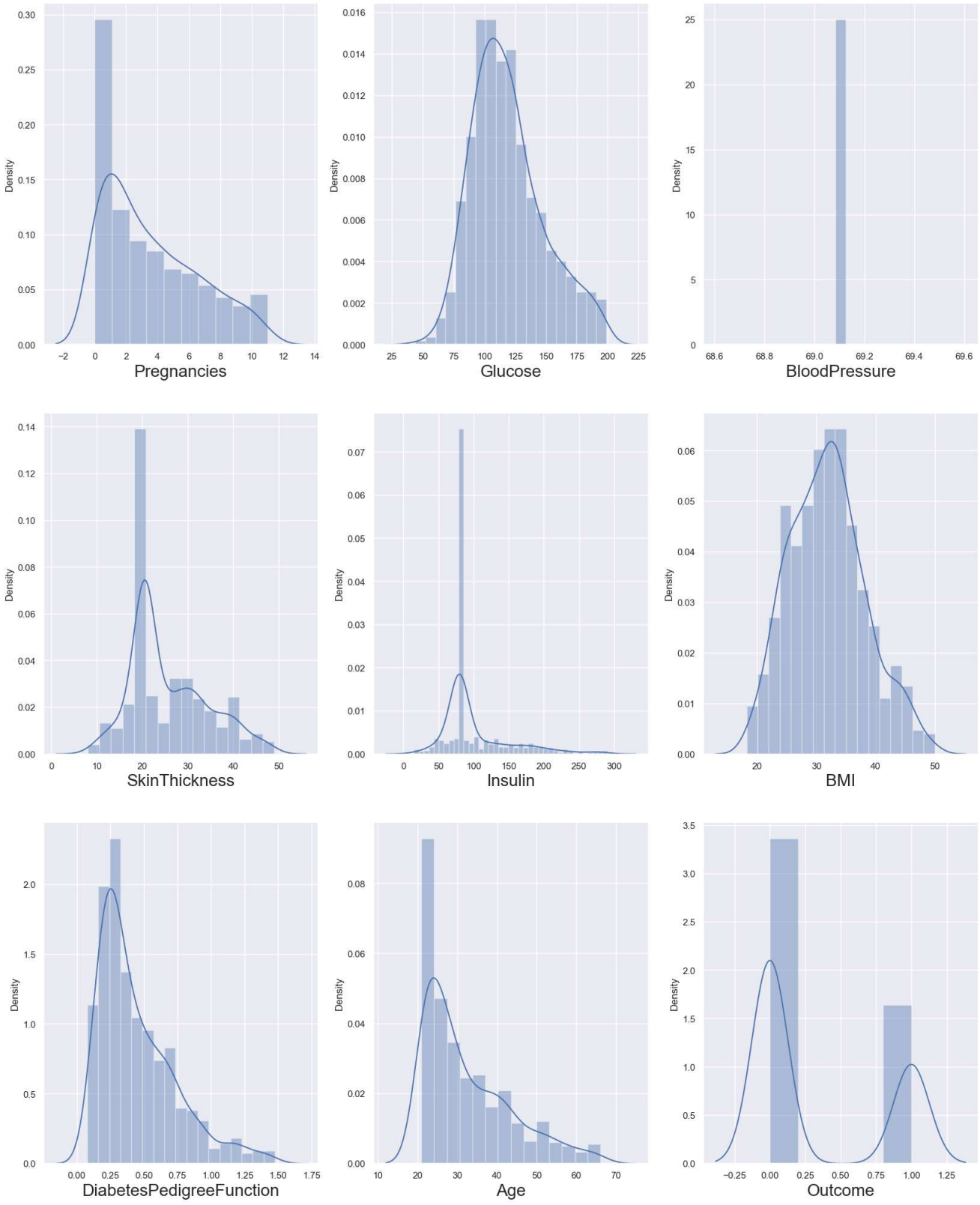
In [34]:
```
data_cleaned.shape
```

Out[34]:
```
(674, 9)
```

In [35]:

```
# Lets's see how data is distributed for every column
plt.figure(figsize=(20,25),facecolor='white')
plotnumber = 1

for column in data_cleaned:
    if plotnumber<=9 :
        ax =plt.subplot(3,3,plotnumber)
        sns.distplot(data_cleaned[column])
        plt.xlabel(column,fontsize=20)
        #plt.ylabel('salary',fontsize=20)
    plotnumber+=1
plt.show()
```

```
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:316: UserWarning: Dataset has 0 variance; skipping den
sity estimate. Pass `warn_singular=False` to disable this warning.
  warnings.warn(msg, UserWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated functi
on and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with simila
r flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



the data looks much better now than before.We will start our analysis with this data now as we dont want to lose important information.if our model doesn't work with accuracy we will come back for more preprocessing

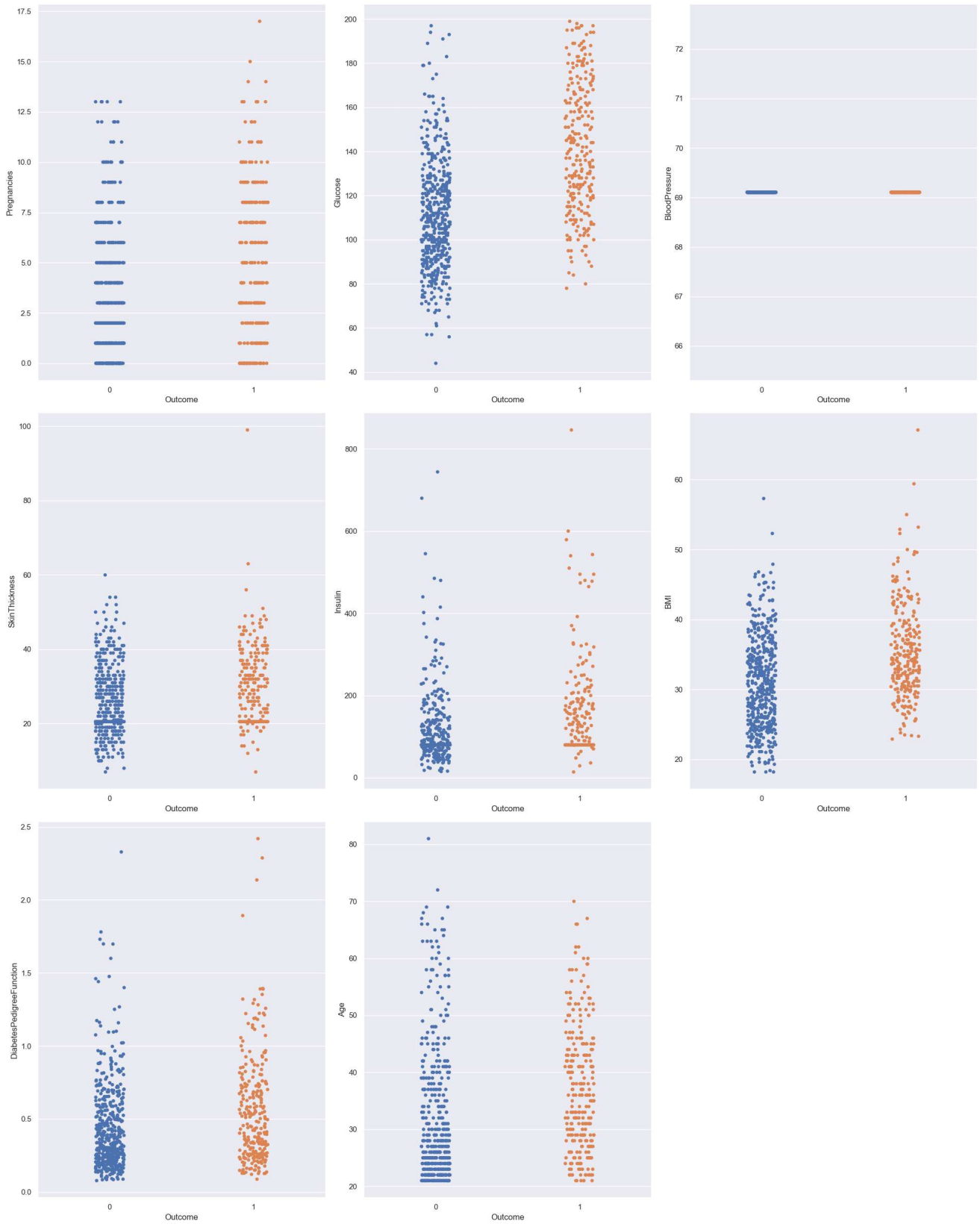
In [36]:
```
x = data.drop(columns= ['outcome'])
y = data['Outcome']
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_11252\1456187962.py in <module>
----> 1 x = data.drop(columns= ['outcome'])
      2 y = data['Outcome']

~\anaconda3\lib\site-packages\pandas\util\_decorators.py in wrapper(*args, **kwargs)
    309                     stacklevel=stacklevel,
    310                 )
--> 311             return func(*args, **kwargs)
    312
    313         return wrapper

~\anaconda3\lib\site-packages\pandas\core\frame.py in drop(self, labels, axis, index, columns, level, inplace, errors)
   4955                 weight  1.0     0.8
   4956         """
-> 4957         return super().drop(
   4958             labels=labels,
   4959             axis=axis,

~\anaconda3\lib\site-packages\pandas\core\generic.py in drop(self, labels, axis, index, columns, level, inplace, errors)
   4265         for axis, labels in axes.items():
   4266             if labels is not None:
-> 4267                 obj = obj._drop_axis(labels, axis, level=level, errors=errors)
   4268
   4269         if inplace:

~\anaconda3\lib\site-packages\pandas\core\generic.py in _drop_axis(self, labels, axis, level, errors, consolidate, only_slice)
   4309                 new_axis = axis.drop(labels, level=level, errors=errors)
   4310             else:
-> 4311                 new_axis = axis.drop(labels, errors=errors)
   4312             indexer = axis.get_indexer(new_axis)
   4313

~\anaconda3\lib\site-packages\pandas\core\indexes\base.py in drop(self, labels, errors)
   6659         if mask.any():
   6660             if errors != "ignore":
-> 6661                 raise KeyError(f"{list(labels[mask])} not found in axis")
   6662             indexer = indexer[~mask]
   6663         return self.delete(indexer)

KeyError: "['outcome'] not found in axis"
```

Before we fit our data to a model,let's visualize the relationship between our independent variables and the catagories.

In [37]:
```python
# let's see how data is distributed for every column
plt.figure(figsize=(20,25), facecolor='white')
plotnumber = 1

for column in x:
    if plotnumber<=9 :
        ax = plt.subplot(3,3,plotnumber)
        sns.stripplot(y,x[column])
    plotnumber+=1
plt.tight_layout()
```

```
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variables as keywo
rd args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an exp
licit keyword will result in an error or misinterpretation.
  warnings.warn(
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variables as keywo
rd args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an exp
licit keyword will result in an error or misinterpretation.
  warnings.warn(
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variables as keywo
rd args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an exp
licit keyword will result in an error or misinterpretation.
  warnings.warn(
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variables as keywo
rd args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an exp
licit keyword will result in an error or misinterpretation.
  warnings.warn(
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variables as keywo
rd args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an exp
licit keyword will result in an error or misinterpretation.
  warnings.warn(
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variables as keywo
rd args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an exp
licit keyword will result in an error or misinterpretation.
  warnings.warn(
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variables as keywo
rd args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an exp
licit keyword will result in an error or misinterpretation.
  warnings.warn(
C:\Users\Vishwanath\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variables as keywo
rd args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an exp
licit keyword will result in an error or misinterpretation.
  warnings.warn(
```



In [39]:
```
scaler = StandardScaler()
X_scaled = scaler.fit_transform(x)
```

This is how our data looks now after scaling .Great now we will check for multicollinerity using VIF(Varience inflation factor)

In [40]:
```
X_scaled
```

Out[40]:
```
array([[ 0.63994726,  0.86527574,  0.        , ...,  0.16725546,
         0.46849198,  1.4259954 ],
       [-0.84488505, -1.20598931,  0.        , ..., -0.85153454,
        -0.36506078, -0.19067191],
       [ 1.23388019,  2.01597855,  0.        , ..., -1.33182125,
         0.60439732, -0.10558415],
       ...,
       [ 0.3429808 , -0.02240928,  0.        , ..., -0.90975111,
        -0.68519336, -0.27575966],
       [-0.84488505,  0.14197684,  0.        , ..., -0.34213954,
        -0.37110101,  1.17073215],
       [-0.84488505, -0.94297153,  0.        , ..., -0.29847711,
        -0.47378505, -0.87137393]])
```

In [41]:
```
vif = pd.DataFrame()
vif['vif']=[variance_inflation_factor(X_scaled,i) for i in range(X_scaled.shape[1])]
vif['Features']= x.columns

# Let's check the values
vif
```

```
C:\Users\Vishwanath\anaconda3\lib\site-packages\statsmodels\regression\linear_model.py:1738: RuntimeWarning: invalid value enco
untered in double_scalars
  return 1 - self.ssr/self.uncentered_tss
```

Out[41]:

	vif	Features
0	1.429498	Pregnancies
1	1.328535	Glucose
2	NaN	BloodPressure
3	1.450335	SkinThickness
4	1.252188	Insulin
5	1.455069	BMI
6	1.054872	DiabetesPedigreeFunction
7	1.512401	Age

In [43]:
```
X_train,X_test,y_train,y_test = train_test_split(X_scaled,y, test_size= 0.25, random_state = 355)
```

In [44]:
```
log_reg = LogisticRegression()

log_reg.fit(X_train,y_train)
```

Out[44]:
```
LogisticRegression()
```

In [45]:
```
import pickle
# writing diffrent model files to file
with open ('modelForPrediction.sav', 'wb') as f :
    pickle.dump(log_reg,f)

with open('standardScalar.sav', 'wb') as f :
    pickle.dump(scalar,f)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_11252\1237901374.py in <module>
      5
      6 with open('standardScalar.sav', 'wb') as f :
----> 7     pickle.dump(scalar,f)

NameError: name 'scalar' is not defined
```

In [48]:
```
y_pred = log_reg.predict(X_test)
```

In [50]:
```
accuracy = accuracy_score(y_test,y_pred)*100
accuracy
```

Out[50]:
```
75.52083333333334
```

In [52]:
```
# confusion matrix
conf_mat = confusion_matrix(y_test, y_pred)
conf_mat
```

Out[52]:
```
array([[110,  15],
       [ 32,  35]], dtype=int64)
```

In [67]:
```
true_Positive = conf_mat[0][0]
false_Positive = conf_mat[0][1]
false_negative = conf_mat[1][0]
true_negative = conf_mat[1][1]
```

In [70]:
```
# Breaking down the formula for accuracy
Accuracy = (true_Positive + true_negative) / (true_Positive + false_Positive + false_negative + true_negative)
Accuracy
```

Out[70]:
```
0.7552083333333334
```

In [71]:
```
# precison
precision = true_Positve/(true_positive+false_positive)
precision
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_11252\909640034.py in <module>
      1 # precison
----> 2 precision = true_Positve/(true_positive+false_positive)
      3 precision

NameError: name 'true_Positve' is not defined
```

In [66]:
```
# Recall
Recall = true_positive/(true_positive+false_positive)
Recall
```

Out[66]:
```
0.88
```

In [64]:
```
# F1 Score
F1_Score = 2*(Recall * precision) / (Recall + precision)
F1_score
```

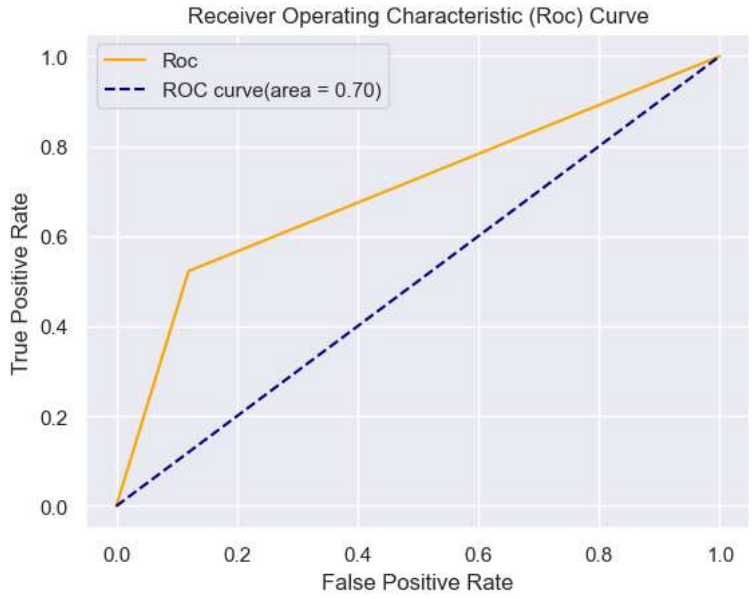
```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_11252\2833481420.py in <module>
      1 # F1 Score
----> 2 F1_Score = 2*(Recall * precision) / (Recall + precision)
      3 F1_score

NameError: name 'precision' is not defined
```

In [72]:
```
# Area Under Curve
auc = roc_auc_score(y_test,y_pred)
```

ROC

In [73]:
```
fpr, tpr, thresholds = roc_curve(y_test, y_pred)
```

In [74]:
```
plt.plot(fpr, tpr, color='orange', label='Roc')
plt.plot([0,1],[0,1],color='darkblue', linestyle='--',label='ROC curve(area = %0.2f)' % auc )
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic (Roc) Curve')
plt.legend()
plt.show()
```



what is the significance of ROc and AUC?

in real life, we create various models using diffrent algorithms that we can use for classification purpose.We use Auc to determine which model is the best one to use for a given dataset.Suppose we have created Logistic regression,SVM as well as a clustering model for classification ppurpose.we will calculate AUC for all the models seprately. The model with highest AUC value will be the best model to use

dvantages of Logisitic Regression It is very simple and easy to implement. The output is more informative than other classification algorithms It expresses the relationship between independent and dependent variables Very effective with linearly seperable data Disadvantages of Logisitic Regression Not effective with data which are not linearly seperable Not as powerful as other classification models Multiclass classifications are much easier to do with other algorithms than logisitic regression It can only predict categorical outcomes Cloud Deployment (Heroku) Once the training is completed, we need to expose the trained model as an API for the user to consume it. For prediction, the saved model is loaded first and then the predictions are made using it. If the web app works fine, the same app is deployed to the cloud platform. The application flow for cloud deployment looks like:

Pre-requisites for cloud deployment: Basic knowledge of flask framework. Any Python IDE installed(we are using PyCharm). A Heroku account. Basic understanding of HTML. Steps before cloud deployment: We need to change our code a bit so that it works unhindered on the cloud, as well.

Add a file called ‘gitignore’ inside the project folder. This folder contains the list of the files which we don’t want to include in the git repository. My gitignore file looks like:

.idea

As I am using PyCharm as an IDE, and it’s provided by the Intellij Idea community, it automatically adds the .idea folder containing some metadata. We need not include them in our cloud app.

Add a file called ‘Procfile’ inside the ‘reviewScrapper’ folder. This folder contains the command to run the flask application once deployed on the server:

web: gunicorn app:app

Here, the keyword ‘web’ specifies that the application is a web application. And the part ‘app:app’ instructs the program to look for a flask application called ‘app’ inside the ‘app.py’ file. Gunicorn is a Web Server Gateway Interface (WSGI) HTTP server for Python.

Open a command prompt window and navigate to your ‘reviewScrapper’ folder. Enter the command ‘pip freeze > requirements.txt’. This command generates the ‘requirements.txt’ file The requirements.txt helps the Heroku cloud app to install all the dependencies before starting the webserver.

After performing all the above steps the project structure will look like:

Deployment to Heroku: After installing the Heroku CLI, Open a command prompt window and navigate to your project folder. Type the command heroku login to login to your heroku account. After logging in to Heroku, enter the command heroku create to create a heroku app. It will give you the URL of your Heroku app after successful creation. Or alternatively, you can go to the heroku website and create an app directly. Before deploying the code to the Heroku cloud, we need to commit the changes to the git repository. Type the command git init to initialize a local git repository. Enter the command git status to see the uncommitted changes. Enter the command git add . to add the uncommitted changes to the local repository. Enter the command git commit -am "make it better" to commit the changes to the local repository. Enter the command git push heroku master to push the code to the heroku cloud. After deployment, heroku gives you the URL to hit the web API. Once your application is deployed successfully, enter the command heroku logs --tail to see the logs. All the code is available in iNeuron git repo.

In [ ]: