In [ ]:

```
## Navie Bayes
### check the  patient  have diabetic or not
```

In [122…

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

In [29]:

```
### import data
data = pd.read_csv(r"C:\Users\shubham lokare\Downloads\archive (4)\Naive-Bayes-Classification-Data.csv")
```

In [30]:

```
data
```

Out[30]:

	glucose	bloodpressure	diabetes
0	40	85	0
1	40	92	0
2	45	63	1
3	45	80	0
4	40	73	1
...	...	...	...
990	45	87	0
991	40	83	0
992	40	83	0
993	40	60	1
994	45	82	0

995 rows × 3 columns

In [31]:

```
data.head(10)
```

Out[31]:

	glucose	bloodpressure	diabetes
0	40	85	0
1	40	92	0
2	45	63	1
3	45	80	0
4	40	73	1
5	45	82	0
6	40	85	0
7	30	63	1
8	65	65	1
9	45	82	0

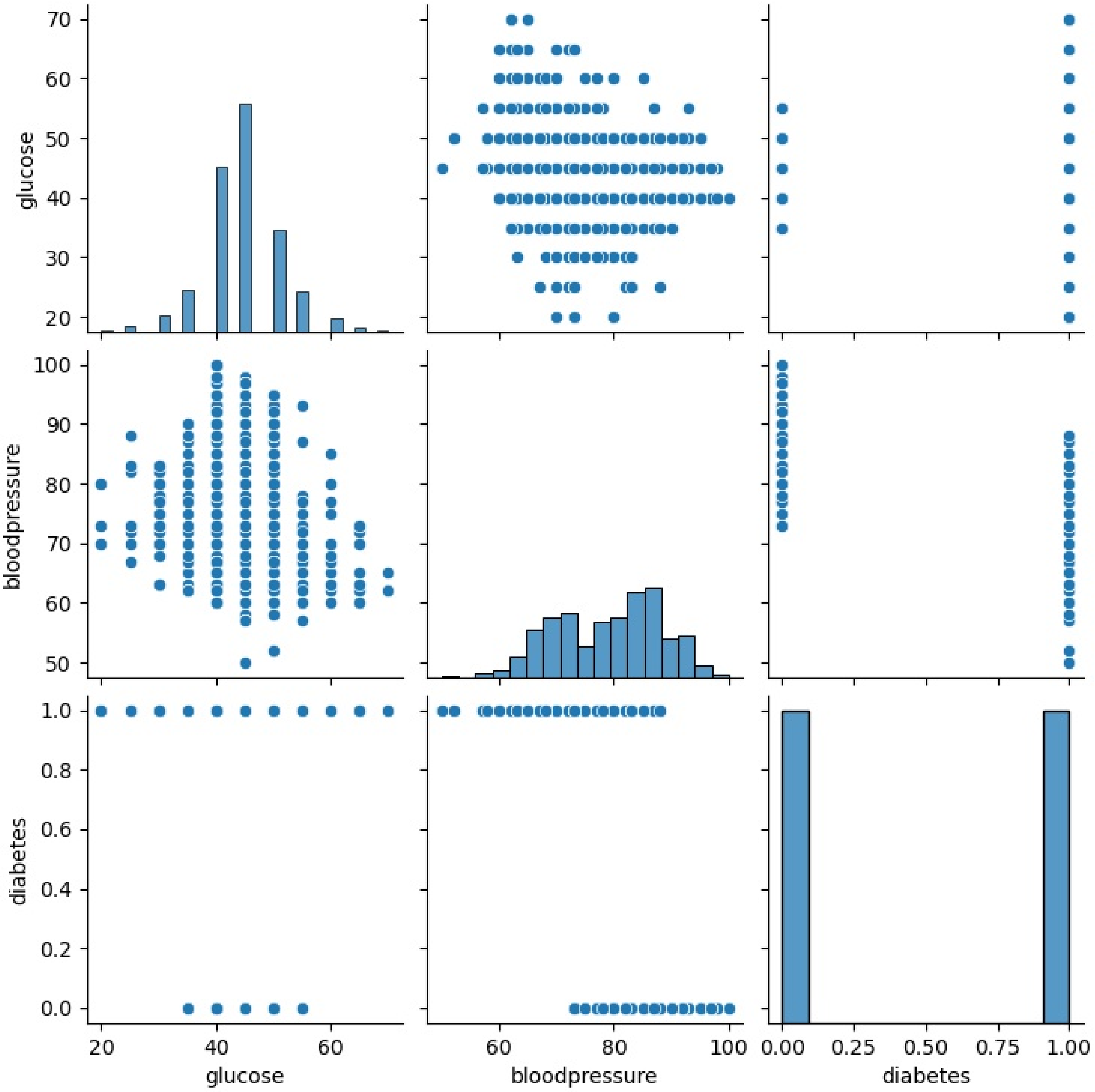
In [123…

```
#### plot pairplot
sns.pairplot(data)
```

```
C:\Users\shubham lokare\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has
changed to tight
  self._figure.tight_layout(*args, **kwargs)
```

Out[123]:

```
<seaborn.axisgrid.PairGrid at 0x229019aef10>
```



In [54]:

```
#### split data into input and output variable

X = pd.DataFrame(data.iloc[: ,:2])
Y = pd.DataFrame(data.iloc[: ,2])
```

In [55]:

```
### input variable
X
```

Out[55]:

	glucose	bloodpressure
0	40	85
1	40	92
2	45	63
3	45	80
4	40	73
...	...	...
990	45	87
991	40	83
992	40	83
993	40	60
994	45	82

995 rows × 2 columns

In [56]:

```
### target
Y
```

Out[56]:

	diabetes
0	0
1	0
2	1
3	0
4	1
...	...
990	0
991	0
992	0
993	1
994	0

995 rows × 1 columns

In [57]:

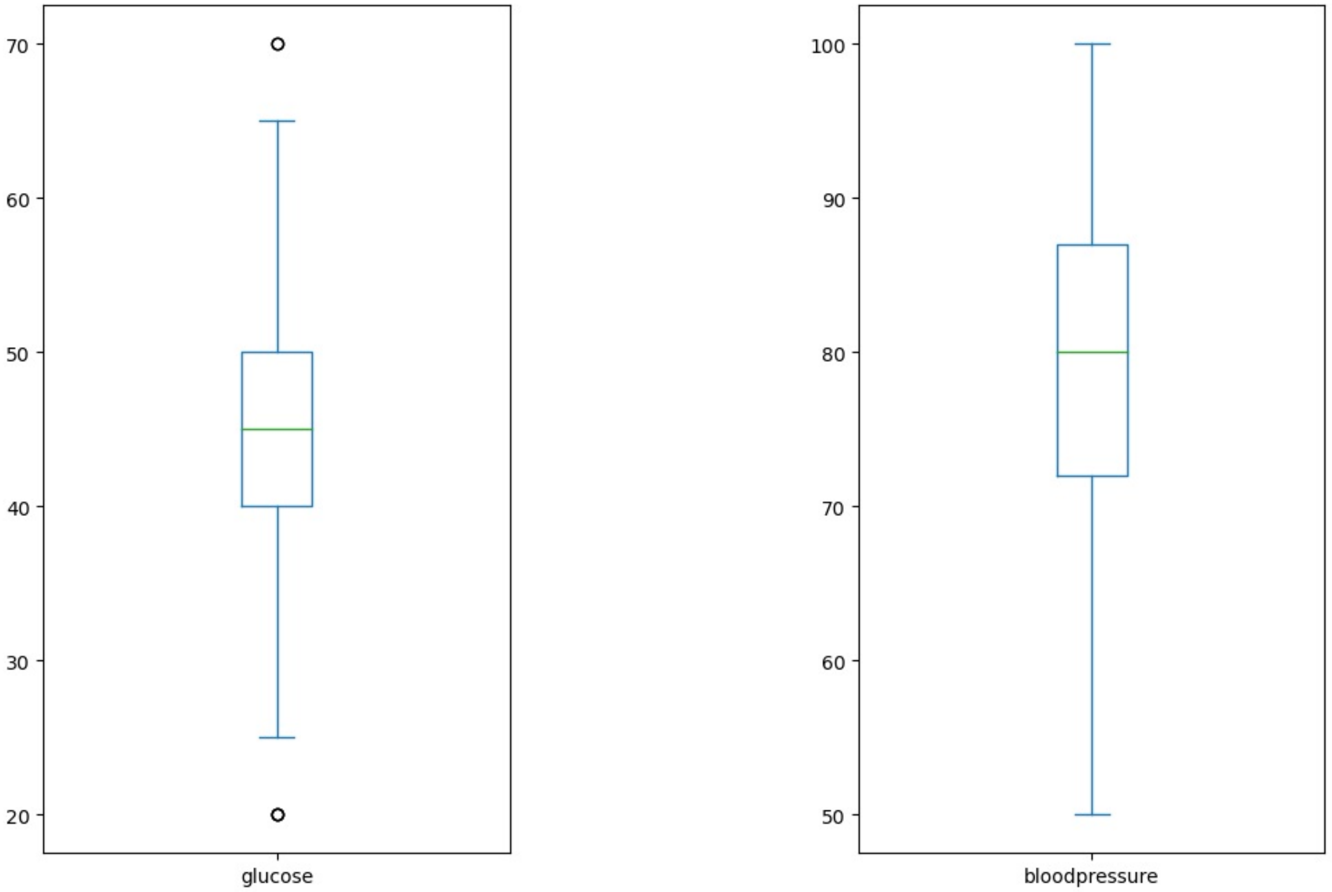
```
# chech missing values
data.isna().sum()
```

Out[57]:

```
glucose          0
bloodpressure    0
diabetes         0
dtype: int64
```

In [58]:

```
### check outliers
X.plot(kind = 'box' , subplots = True , figsize = (12,8))
plt.subplots_adjust(wspace = 0.75)
plt.show()
```



In [62]:

```
### use winsorzer method to remove the outliers
from feature_engine.outliers import Winsorizer

winsor = Winsorizer(capping_method= 'iqr' ,
                    tail= 'both' ,
                    fold = 1.5 ,
                    variables=list(X.columns))
```

In [63]:

```
new_data = winsor.fit_transform(X)
```

In [64]:

```
new_data
```

Out[64]:

	glucose	bloodpressure
0	40	85
1	40	92
2	45	63
3	45	80
4	40	73
...	...	...
990	45	87
991	40	83
992	40	83
993	40	60
994	45	82

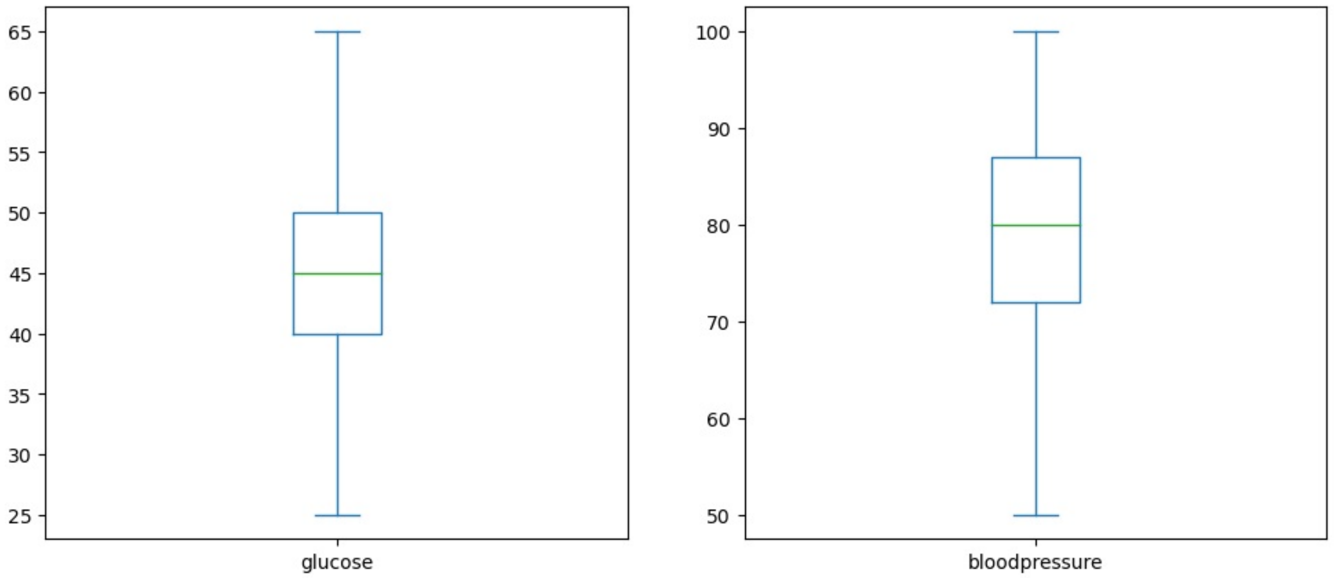
995 rows × 2 columns

In [65]:

```
### check the outliers are remove or not
new_data.plot(kind = 'box' , subplots = True , figsize = (12,5))
```

Out[65]:

```
glucose               Axes(0.125,0.11;0.352273x0.77)
bloodpressure      Axes(0.547727,0.11;0.352273x0.77)
dtype: object
```



In [66]:
```
### make the data into  scale
from sklearn.preprocessing import StandardScaler

scale = StandardScaler()

scale_data = scale.fit_transform(new_data)
```

In [67]:
```
clean_data = pd.DataFrame(scale_data)
```

In [68]:
```
clean_data
```

Out[68]:

	0	1
0	-0.651307	0.622899
1	-0.651307	1.372724
2	0.103997	-1.733695
3	0.103997	0.087309
4	-0.651307	-0.662516
...	...	...
990	0.103997	0.837134
991	-0.651307	0.408663
992	-0.651307	0.408663
993	-0.651307	-2.055049
994	0.103997	0.301545

995 rows × 2 columns

In [69]:
```
clean_data.describe()
```

Out[69]:

	0	1
count	9.950000e+02	9.950000e+02
mean	-2.303015e-16	3.034982e-17
std	1.000503e+00	1.000503e+00
min	-2.917216e+00	-3.126227e+00
25%	-6.513066e-01	-7.696339e-01
50%	1.039965e-01	8.730915e-02
75%	8.592996e-01	8.371343e-01
max	3.125209e+00	2.229667e+00

In [70]:
```
#### apply model navies bayes
```

```
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score ,classification_report
```

In [71]:

```
### train and test data

X_train ,X_test , Y_train ,Y_test =train_test_split(clean_data , Y , test_size = 0.2 , random_state =0)
```

In [72]:

```
## train data
X_train
```

Out[72]:

	0	1
864	0.103997	-0.662516
874	0.103997	-2.055049
878	-0.651307	0.837134
113	1.614603	-2.055049
608	0.103997	-0.662516
...	...	...
835	-0.651307	-1.519459
192	0.103997	0.622899
629	-0.651307	0.408663
559	0.859300	1.158488
684	0.859300	-0.448280

796 rows × 2 columns

In [73]:

```
#### test data
Y_test
```

Out[73]:

	diabetes
420	1
985	1
31	1
692	1
553	0
...	...
769	1
958	1
382	1
271	0
643	1

199 rows × 1 columns

In [75]:

```
### apply gaussianNB model

model = GaussianNB()
model.fit(X_train ,Y_train)
```

```
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
```

Out[75]:

```
▾ GaussianNB
GaussianNB()
```

In [76]:

```
predict = model.predict(X_test)
```

In [77]:

```
print(predict)
```

```
[1 1 1 0 0 0 1 1 0 1 0 0 1 1 1 1 1 0 1 1 0 0 0 0 1 1 1 0 0 0 1 1 0 0 0 1 0
 1 1 1 1 1 0 0 0 1 0 1 0 0 0 0 1 0 0 0 0 1 1 0 1 0 1 0 1 0 0 1 1 1 0 1 0 1
 0 1 1 1 0 1 0 1 1 1 0 1 0 0 1 0 1 0 0 0 1 0 1 0 0 1 0 0 1 0 0 0 0 1 1 0 0
 1 1 0 1 1 0 0 0 1 1 0 1 1 1 0 0 1 1 0 1 1 1 1 0 0 1 1 0 0 1 1 1 1 0 1 0 0
 0 1 1 0 1 1 1 0 1 1 1 1 0 1 0 0 1 0 1 1 1 0 0 0 1 0 0 0 0 1 1 1 1 0 1 1 0
 0 1 1 1 1 0 1 0 1 1 0 1 0 1]
```

In [79]:

```
## check accuracy
```

```
print('test accuracy :' , accuracy_score(Y_test , predict))
```

```
test accuracy : 0.9095477386934674
```

In [80]:

```
### classification_reoprt
score =classification_report(Y_test , predict)
```

In [81]:

```
print(score)
```

```
              precision    recall  f1-score   support

           0       0.87      0.93      0.90        88
           1       0.94      0.89      0.92       111

    accuracy                           0.91       199
   macro avg       0.91      0.91      0.91       199
weighted avg       0.91      0.91      0.91       199
```

In [82]:

```
### cross check the model

pd.crosstab(Y_test.diabetes, predict)
```

Out[82]:

col_0	0	1
diabetes		
0	82	6
1	12	99

In [83]:

```
### also check confusion metrics
from sklearn.metrics import confusion_matrix
```

In [84]:

```
conf = confusion_matrix(Y_test , predict)
```

In [85]:

```
conf
```

Out[85]:

```
array([[82,  6],
       [12, 99]], dtype=int64)
```

In [87]:

```
print("True Positive :" , conf[1,1])
```

```
True Positive : 99
```

In [89]:

```
print("Flase Positive:" ,conf[1,0])
```

```
Flase Positive: 12
```

In [90]:

```
print("False negitive :" , conf[0,1])
```

```
False negitive : 6
```

In [91]:

```
print("True negitive :" , conf[0 ,0])
```

```
True negitive : 82
```

In [115…

```
### hyperparameter tuning

from sklearn.model_selection import GridSearchCV

gnb = GaussianNB()
```

In [116…

```
param_grid = {
    'priors': [None],  # Example values for priors
    'var_smoothing': [1e-9, 1e-8, 1e-7]  # Example values for var_smoothing
}
```

In [117…

```
grid = GridSearchCV(gnb , param_grid , scoring='accuracy' , cv =8 ,verbose=1)

grid.fit(X_train , Y_train)
```

```
Fitting 8 folds for each of 3 candidates, totalling 24 fits
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
```

```
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
C:\Users\shubham lokare\anaconda3\Lib\site-packages\sklearn\utils\validation.py:1184: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for exa
mple using ravel().
  y = column_or_1d(y, warn=True)
```

Out[117]:

```
▸ GridSearchCV
▸ estimator: GaussianNB
  ▸ GaussianNB
```

In [118…

```
grid.best_params_
```

Out[118]:

```
{'priors': None, 'var_smoothing': 1e-09}
```

In [119…

```
predit = grid.predict(X_test)
```

In [120…

```
predit
```

Out[120]:

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

In [121…

```
print("test Accureacy :" , accuracy_score(Y_test , predit))
```

```
test Accureacy : 0.9095477386934674
```

In [ ]:

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js