SVM classifier model:

1) Test size=.20, Parameter(default)

```
In [14]: runcell(0, '/Users/myyntiimac/untitled1.py')
[[55  3]
 [ 1 21]]
Out[14]: 0.95

In [15]: runcell(0, '/Users/myyntiimac/untitled1.py')
[[55  3]
 [ 1 21]]
Out[15]: 0.95

In [16]: runcell(0, '/Users/myyntiimac/untitled1.py')
[[55  3]
 [ 1 21]]
Out[16]: '              precision    recall  f1-score   support\n\n           0       0.98      0.95      0.96        58\n           1
0.88      0.95      0.91        22\n\n    accuracy                           0.95        80\n   macro avg       0.93      0.95      0.94
80\nweighted avg       0.95      0.95      0.95        80\n'

In [17]: runcell(0, '/Users/myyntiimac/untitled1.py')
[[55  3]
 [ 1 21]]
Out[17]: 0.903125

In [18]: runcell(0, '/Users/myyntiimac/untitled1.py')
[[55  3]
 [ 1 21]]
Out[18]: 0.95
```

2)

Test size=.25,parameter(C=2,kernel="rbf",degree=2,gamma="auto")

```
In [7]: runcell(0, '/Users/myyntiimac/Desktop/Support vector classification  default  and future prediction.py')
[[64  4]
 [ 3 29]]

In [8]: from sklearn.metrics import accuracy_score
   ...: ac = accuracy_score(y_test, y_pred)
   ...: ac
Out[8]: 0.93

In [9]: from sklearn.metrics import classification_report
   ...: cr = classification_report(y_test, y_pred)
   ...: cr
Out[9]: '              precision    recall  f1-score   support\n\n           0       0.96      0.94      0.95        68\n           1
0.88      0.91      0.89        32\n\n    accuracy                           0.93       100\n   macro avg       0.92      0.92      0.92
100\nweighted avg       0.93      0.93      0.93       100\n'

In [10]: bias = classifier.score(X_train, y_train)
    ...: bias
Out[10]: 0.91

In [11]: variance = classifier.score(X_test, y_test)
    ...: variance
Out[11]: 0.93

In [12]:
```

3)

Test size=.30,parameter(C=1,kernel="sigmoid",degree=3,gamma=auto)

```
    ...: ac
Out[17]: 0.7416666666666667

In [18]: from sklearn.metrics import classification_report
    ...: cr = classification_report(y_test, y_pred)
    ...: cr
Out[18]: '              precision    recall  f1-score   support\n\n           0       0.79      0.82      0.81        79\n           1
0.63      0.59      0.61        41\n\n    accuracy                           0.74       120\n   macro avg       0.71      0.70      0.71
120\nweighted avg       0.74      0.74      0.74       120\n'

In [19]: bias = classifier.score(X_train, y_train)
    ...: bias
Out[19]: 0.6535714285714286

In [20]: #check the variance
    ...: variance = classifier.score(X_test, y_test)
    ...: varianc
Traceback (most recent call last):

  Cell In[20], line 3
    varianc

NameError: name 'varianc' is not defined


In [21]: #check the variance
    ...: variance = classifier.score(X_test, y_test)
    ...: variance
Out[21]: 0.7416666666666667
```

4)

Test size=.15,parameter= C=2,kernel="poly",degree=4,gamma="scale"

```
 [ 9  6]]

In [26]: from sklearn.metrics import confusion_matrix
    ...: cm = confusion_matrix(y_test, y_pred)
    ...: print(cm)
[[42  3]
 [ 9  6]]

In [27]: from sklearn.metrics import accuracy_score
    ...: ac = accuracy_score(y_test, y_pred)
    ...: ac
Out[27]: 0.8

In [28]: from sklearn.metrics import classification_report
    ...: cr = classification_report(y_test, y_pred)
    ...: cr
Out[28]: '              precision    recall  f1-score   support\n\n           0       0.82      0.93      0.87        45\n           1
0.67      0.40      0.50        15\n\n    accuracy                           0.80        60\n   macro avg       0.75      0.67      0.69
60\nweighted avg       0.78      0.80      0.78        60\n'

In [29]: bias = classifier.score(X_train, y_train)
    ...: bias
Out[29]: 0.8235294117647058

In [30]: variance = classifier.score(X_test, y_test)
    ...: variance
Out[30]: 0.8

In [31]:
```

5)

Test size=.35,parameter(C=1,kernel="rbf",degree=5,gamma="scale")

```
In [32]: runcell(0, '/Users/myyntiimac/Desktop/Support vector classification  default  and future prediction.py')
[[80  9]
 [ 4 47]]

In [33]: from sklearn.metrics import accuracy_score
    ...: ac = accuracy_score(y_test, y_pred)
    ...: ac
Out[33]: 0.9071428571428571

In [34]: from sklearn.metrics import classification_report
    ...: cr = classification_report(y_test, y_pred)
    ...: cr
Out[34]: '              precision    recall  f1-score   support\n\n           0       0.95      0.90      0.92        89\n           1
0.84      0.92      0.88        51\n\n    accuracy                           0.91       140\n   macro avg       0.90      0.91      0.90
140\nweighted avg       0.91      0.91      0.91       140\n'

In [35]: bias = classifier.score(X_train, y_train)
    ...: bias
Out[35]: 0.9192307692307692

In [36]: variance = classifier.score(X_test, y_test)
    ...: variance
Out[36]: 0.9071428571428571

In [37]:
```

KNN classifier

1) T est size=.20.Parameter(default)

```
Out[17]: '              precision    recall  f1-score   support\n\n           0       0.98      0.95      0.96        58\n           1
0.88      0.95      0.91        22\n\n    accuracy                           0.95        80\n   macro avg       0.93      0.95      0.94
80\nweighted avg       0.95      0.95      0.95        80\n'

In [18]: runcell(0, '/Users/myyntiimac/Desktop/untitled2.py')
[[55  3]
 [ 1 21]]
Traceback (most recent call last):

  File ~/anaconda3/lib/python3.10/site-packages/spyder_kernels/py3compat.py:356 in compat_exec
    exec(code, globals, locals)

  File ~/Desktop/untitled2.py:56
    bias = classifier.score(X_train, y_train)

NameError: name 'classifier' is not defined


In [19]: runcell(0, '/Users/myyntiimac/Desktop/untitled2.py')
[[55  3]
 [ 1 21]]
Out[19]: 0.91875

In [20]: runcell(0, '/Users/myyntiimac/Desktop/untitled2.py')
[[55  3]
 [ 1 21]]
Out[20]: 0.95
```

2) Test size= .30 parameter=,(n=3,algo=kd_tree, weight=uniform , pint=2)

```
In [19]: from sklearn.metrics import confusion_matrix
    ...: cm = confusion_matrix(y_test, y_pred)
    ...: print(cm)
[[73  6]
 [ 5 36]]

In [20]: from sklearn.metrics import accuracy_score
    ...: ac = accuracy_score(y_test, y_pred)
    ...: ac
Out[20]: 0.9083333333333333

In [21]: #check the vclassification report
    ...: from sklearn.metrics import classification_report
    ...: cr = classification_report(y_test, y_pred)
    ...: cr
Out[21]: '              precision    recall  f1-score   support\n\n           0       0.94      0.92      0.93        79\n           1
0.86      0.88      0.87        41\n\n    accuracy                           0.91       120\n   macro avg       0.90      0.90      0.90
120\nweighted avg       0.91      0.91      0.91       120\n'

In [22]: bias = knn.score(X_train, y_train)
    ...: bias
Out[22]: 0.925

In [23]: variance = knn.score(X_test, y_test)
    ...: variance
Out[23]: 0.9083333333333333
```

3) Test size=.25,parameter(n=4,alg=ball_tree, p=2,distance,p1)

```
 [ 3 29]]

In [28]: runcell(0, '/Users/myyntiimac/Desktop/KNN.py')
[[64  4]
 [ 3 29]]

In [29]: from sklearn.metrics import accuracy_score
    ...: ac = accuracy_score(y_test, y_pred)
    ...: ac
Out[29]: 0.93

In [30]: from sklearn.metrics import classification_report
    ...: cr = classification_report(y_test, y_pred)
    ...: cr
Out[30]: '              precision    recall  f1-score   support\n\n           0       0.96      0.94      0.95        68\n           1
0.88      0.91      0.89        32\n\n    accuracy                           0.93       100\n   macro avg       0.92      0.92      0.92
100\nweighted avg       0.93      0.93      0.93       100\n'

In [31]: bias = knn.score(X_train, y_train)
    ...: bias
Out[31]: 1.0

In [32]: #check the variance
    ...: variance = knn.score(X_test, y_test)
    ...: variance
Out[32]: 0.93

In [33]:
```

4)Test size=.15, parameter (n=6,algo=ball_ttree,distance,p1)

```
 [ 1 14]]

In [37]: runcell(0, '/Users/myyntiimac/Desktop/KNN.py')
[[42  3]
 [ 1 14]]

In [38]: from sklearn.metrics import accuracy_score
    ...: ac = accuracy_score(y_test, y_pred)
    ...: ac
Out[38]: 0.9333333333333333

In [39]: from sklearn.metrics import classification_report
    ...: cr = classification_report(y_test, y_pred)
    ...: cr
Out[39]: '              precision    recall  f1-score   support\n\n           0       0.98      0.93      0.95        45\n           1
0.82      0.93      0.87        15\n\n    accuracy                           0.93        60\n   macro avg       0.90      0.93      0.91
60\nweighted avg       0.94      0.93      0.93        60\n'

In [40]: bias = knn.score(X_train, y_train)
    ...: bias
Out[40]: 0.9970588235294118

In [41]: variance = knn.score(X_test, y_test)
    ...: variance
Out[41]: 0.9333333333333333

In [42]:
```

4) Test size=.15,Parameter(n=7,algo=brute, weight =uni,p=2)

```
    ...: print(cm)
[[42  3]
 [ 1 14]]

In [46]: from sklearn.metrics import accuracy_score
    ...: ac = accuracy_score(y_test, y_pred)
    ...: ac
Out[46]: 0.9333333333333333

In [47]: from sklearn.metrics import classification_report
    ...: cr = classification_report(y_test, y_pred)
    ...: cr
Out[47]: '              precision    recall  f1-score   support\n\n           0       0.98      0.93      0.95        45\n           1
0.82      0.93      0.87        15\n\n    accuracy                           0.93        60\n   macro avg       0.90      0.93      0.91
60\nweighted avg       0.94      0.93      0.93        60\n'

In [48]: bias = knn.score(X_train, y_train)
    ...: bias
Out[48]: 0.9176470588235294

In [49]: variance = knn.score(X_test, y_test)
    ...: variance
Out[49]: 0.9333333333333333

In [50]:
```