1.Random Forest Classification

```
from sklearn.ensemble import RandomForestClassifier
```

```
(n_estimators = 100, criterion = 'gini',max_features = 'sqrt', random_state = 0)

classifier.fit(X_train, y_train)
```

```
              precision    recall  f1-score   support

       False       0.98      0.98      0.98        51
        True       0.99      0.99      0.99        82

    accuracy                           0.98       133
   macro avg       0.98      0.98      0.98       133
weighted avg       0.98      0.98      0.98       133


[[50  1]
 [ 1 81]]
```

```
(n_estimators = 100, criterion = 'entropy',max_features = 'log2', random_state = 0)

classifier.fit(X_train, y_train)
```

```
[[51  0]
 [ 1 81]]

              precision    recall  f1-score   support

       False       0.98      1.00      0.99        51
        True       1.00      0.99      0.99        82

    accuracy                           0.99       133
   macro avg       0.99      0.99      0.99       133
weighted avg       0.99      0.99      0.99       133
```

```
(n_estimators = 100, criterion = 'entropy',max_features = 'sqrt', random_state = 0)

classifier.fit(X_train, y_train)
```

```
[[50  1]
 [ 1 81]]

              precision    recall  f1-score   support

       False       0.98      0.98      0.98        51
        True       0.99      0.99      0.99        82

    accuracy                           0.98       133
   macro avg       0.98      0.98      0.98       133
weighted avg       0.98      0.98      0.98       133
```

```
(n_estimators = 100, criterion = 'log_loss',max_features = 'sqrt', random_state = 0)

classifier.fit(X_train, y_train)
```

```
[[50  1]
 [ 1 81]]

              precision    recall  f1-score   support

       False       0.98      0.98      0.98        51
        True       0.99      0.99      0.99        82

    accuracy                           0.98       133
   macro avg       0.98      0.98      0.98       133
weighted avg       0.98      0.98      0.98       133
```

```
(n_estimators = 100, criterion = 'log_loss',max_features = 'log2', random_state = 0)

classifier.fit(X_train, y_train)
```

```
[[51  0]
 [ 1 81]]

           precision    recall  f1-score   support

       False      0.98      1.00      0.99        51
        True      1.00      0.99      0.99        82

    accuracy                          0.99       133
   macro avg      0.99      0.99      0.99       133
weighted avg      0.99      0.99      0.99       133
```

2.LogisticRegression

(penalty = 'l1', solver = 'saga' , multi_class = 'multinomial' , random_state=0)

```
          precision    recall  f1-score   support

     False       0.00      0.00      0.00        51
      True       0.62      1.00      0.76        82

  accuracy                           0.62       133
 macro avg       0.31      0.50      0.38       133
weighted avg       0.38      0.62      0.47       133
```

LogisticRegression(penalty = 'l2', solver = 'lbfgs' , multi_class = 'auto' , random_state=0)

```
          precision    recall  f1-score   support

     False       0.87      0.92      0.90        51
      True       0.95      0.91      0.93        82

  accuracy                           0.92       133
 macro avg       0.91      0.92      0.91       133
weighted avg       0.92      0.92      0.92       133
```

LogisticRegression(penalty = 'l2', solver = 'liblinear' , multi_class = 'auto' , random_state=0)

```
          precision    recall  f1-score   support

     False       1.00      0.96      0.98        51
      True       0.98      1.00      0.99        82

  accuracy                           0.98       133
 macro avg       0.99      0.98      0.98       133
weighted avg       0.99      0.98      0.98       133
```

```
LogisticRegression( penalty = 'l2', solver = 'liblinear' , multi_class = 'ovr' , random_state=0)
```

```
classifier.fit(X_train, y_train)
```

```
              precision    recall  f1-score   support

       False       1.00      0.96      0.98        51
        True       0.98      1.00      0.99        82

    accuracy                           0.98       133
   macro avg       0.99      0.98      0.98       133
weighted avg       0.99      0.98      0.98       133
```

In [ ]:

```
LogisticRegression( penalty = 'l2', solver = 'newton-cg' , multi_class = 'multinomial' ,
random_state=0)
```

```
classifier.fit(X_train, y_train)
```

```
              precision    recall  f1-score   support

       False       1.00      1.00      1.00        51
        True       1.00      1.00      1.00        82

    accuracy                           1.00       133
   macro avg       1.00      1.00      1.00       133
weighted avg       1.00      1.00      1.00       133
```

```
LogisticRegression( penalty = 'l2', solver = 'newton-cg' , multi_class = 'auto' , random_state=0)
```

```
              precision    recall  f1-score   support

       False       1.00      1.00      1.00        51
        True       1.00      1.00      1.00        82
```

```
    accuracy                           1.00       133
   macro avg       1.00      1.00      1.00       133
weighted avg       1.00      1.00      1.00       133
```

LogisticRegression(penalty = 'l2', solver = 'newton-cg' , multi_class = 'ovr' , random_state=0)

```
              precision    recall  f1-score   support

       False       1.00      1.00      1.00        51
        True       1.00      1.00      1.00        82

    accuracy                           1.00       133
   macro avg       1.00      1.00      1.00       133
weighted avg       1.00      1.00      1.00       133
```

LogisticRegression(penalty = 'l2', solver = 'newton-cg' , multi_class = 'auto' , random_state=0)

```
              precision    recall  f1-score   support

       False       1.00      1.00      1.00        51
        True       1.00      1.00      1.00        82

    accuracy                           1.00       133
   macro avg       1.00      1.00      1.00       133
weighted avg       1.00      1.00      1.00       133
```

LogisticRegression(penalty = 'l2', solver = 'newton-cholesky' , multi_class = 'auto' , random_state=0)

```
              precision    recall  f1-score   support

       False       1.00      1.00      1.00        51
        True       1.00      1.00      1.00        82

    accuracy                           1.00       133
   macro avg       1.00      1.00      1.00       133
weighted avg       1.00      1.00      1.00       133
```

LogisticRegression(penalty = 'l2', solver = 'newton-cholesky' , multi_class = 'ovr' , random_state=0)

```
            precision    recall  f1-score   support
```

```
       False       1.00      1.00      1.00        51
        True       1.00      1.00      1.00        82

    accuracy                           1.00       133
   macro avg       1.00      1.00      1.00       133
weighted avg       1.00      1.00      1.00       133
```

LogisticRegression(penalty = 'l2', solver = 'sag' , multi_class = 'multinomial' , random_state=0)

```
              precision    recall  f1-score   support

       False       0.00      0.00      0.00        51
        True       0.62      1.00      0.76        82

    accuracy                           0.62       133
   macro avg       0.31      0.50      0.38       133
weighted avg       0.38      0.62      0.47       133
```

In [ ]:

LogisticRegression(penalty = 'l2', solver = 'saga' , multi_class = 'multinomial' , random_state=0)

```
              precision    recall  f1-score   support

       False       0.00      0.00      0.00        51
        True       0.62      1.00      0.76        82

    accuracy                           0.62       133
   macro avg       0.31      0.50      0.38       133
weighted avg       0.38      0.62      0.47       133
```

LogisticRegression(penalty = 'l2', solver = 'saga' , multi_class = 'ovr' , random_state=0)

```
              precision    recall  f1-score   support

       False       0.00      0.00      0.00        51
        True       0.62      1.00      0.76        82

    accuracy                           0.62       133
   macro avg       0.31      0.50      0.38       133
weighted avg       0.38      0.62      0.47       133
```

LogisticRegression(penalty = 'l2', solver = 'saga' , multi_class = 'auto' , random_state=0)

```
          precision    recall  f1-score   support

       False       0.00      0.00      0.00        51
        True       0.62      1.00      0.76        82

    accuracy                           0.62       133
   macro avg       0.31      0.50      0.38       133
weighted avg       0.38      0.62      0.47       133
```

3.Decision Tree classification

DecisionTreeClassifier(criterion = 'gini', splitter = 'random' , max_features = 'sqrt' , random_state = 0)

```
          precision    recall  f1-score   support

       False       0.96      1.00      0.98        51
        True       1.00      0.98      0.99        82

    accuracy                           0.98       133
   macro avg       0.98      0.99      0.98       133
weighted avg       0.99      0.98      0.99       133
```

DecisionTreeClassifier(criterion = 'gini', splitter = 'best' , max_features = 'sqrt' , random_state = 0)

```
          precision    recall  f1-score   support

       False       0.88      0.96      0.92        51
        True       0.97      0.91      0.94        82

    accuracy                           0.93       133
   macro avg       0.92      0.94      0.93       133
weighted avg       0.94      0.93      0.93       133
```

DecisionTreeClassifier(criterion = 'gini', splitter = 'best' , max_features = 'log2' , random_state = 0)

```
              precision    recall  f1-score   support

       False       0.93      1.00      0.96        51
        True       1.00      0.95      0.97        82

    accuracy                           0.97       133
   macro avg       0.96      0.98      0.97       133
weighted avg       0.97      0.97      0.97       133
```

DecisionTreeClassifier(criterion = 'gini', splitter = 'random' , max_features = 'log2' , random_state = 0)

```
              precision    recall  f1-score   support

       False       0.94      0.88      0.91        51
        True       0.93      0.96      0.95        82

    accuracy                           0.93       133
   macro avg       0.93      0.92      0.93       133
weighted avg       0.93      0.93      0.93       133
```

DecisionTreeClassifier(criterion = 'entropy', splitter = 'random' , max_features = 'log2' , random_state = 0)

```
              precision    recall  f1-score   support

       False       0.93      0.98      0.95        51
        True       0.99      0.95      0.97        82

    accuracy                           0.96       133
   macro avg       0.96      0.97      0.96       133
weighted avg       0.96      0.96      0.96       133
```

DecisionTreeClassifier(criterion = 'entropy', splitter = 'random' , max_features = 'sqrt' , random_state = 0)

```
              precision    recall  f1-score   support

       False       0.96      1.00      0.98        51
        True       1.00      0.98      0.99        82

    accuracy                           0.98       133
```

```
   macro avg       0.98      0.99      0.98       133
weighted avg       0.99      0.98      0.99       133
```

DecisionTreeClassifier(criterion = 'entropy', splitter = 'best' , max_features = 'sqrt' , random_state = 0)

```
              precision    recall  f1-score   support

       False       0.88      0.96      0.92        51
        True       0.97      0.91      0.94        82

    accuracy                           0.93       133
   macro avg       0.92      0.94      0.93       133
weighted avg       0.94      0.93      0.93       133
```

DecisionTreeClassifier(criterion = 'entropy', splitter = 'best' , max_features = 'log2' , random_state = 0)

```
              precision    recall  f1-score   support

       False       0.96      0.98      0.97        51
        True       0.99      0.98      0.98        82

    accuracy                           0.98       133
   macro avg       0.97      0.98      0.98       133
weighted avg       0.98      0.98      0.98       133
```

DecisionTreeClassifier(criterion = 'log_loss', splitter = 'best' , max_features = 'log2' , random_state = 0)

```
              precision    recall  f1-score   support

       False       0.96      0.98      0.97        51
        True       0.99      0.98      0.98        82

    accuracy                           0.98       133
   macro avg       0.97      0.98      0.98       133
weighted avg       0.98      0.98      0.98       133
```

DecisionTreeClassifier(criterion = 'log_loss', splitter = 'random' , max_features = 'log2' , random_state = 0)

```
              precision    recall  f1-score   support
```

```
       False       0.93      0.98      0.95        51
        True       0.99      0.95      0.97        82

    accuracy                           0.96       133
   macro avg       0.96      0.97      0.96       133
weighted avg       0.96      0.96      0.96       133
```

```
DecisionTreeClassifier(criterion = 'log_loss', splitter = 'random' , max_features = 'sqrt' ,
random_state = 0)
```

```
              precision    recall  f1-score   support

       False       0.96      1.00      0.98        51
        True       1.00      0.98      0.99        82

    accuracy                           0.98       133
   macro avg       0.98      0.99      0.98       133
weighted avg       0.99      0.98      0.99       133
```

```
DecisionTreeClassifier(criterion = 'log_loss', splitter = 'best' , max_features = 'sqrt' ,
random_state = 0)
```

```
              precision    recall  f1-score   support

       False       0.88      0.96      0.92        51
        True       0.97      0.91      0.94        82

    accuracy                           0.93       133
   macro avg       0.92      0.94      0.93       133
weighted avg       0.94      0.93      0.93       133
```

5. KNeighborsClassifier

KNeighborsClassifier(n_neighbors = 5, weights = 'uniform', algorithm = 'ball_tree' , p = 2)

```
              precision    recall  f1-score   support

       False       0.61      0.86      0.72        51
        True       0.89      0.66      0.76        82

    accuracy                           0.74       133
   macro avg       0.75      0.76      0.74       133
weighted avg       0.78      0.74      0.74       133
```

KNeighborsClassifier(n_neighbors = 5, weights = 'uniform', algorithm = 'kd_tree' , p = 2)

```
              precision    recall  f1-score   support

       False       0.61      0.86      0.72        51
        True       0.89      0.66      0.76        82

    accuracy                           0.74       133
   macro avg       0.75      0.76      0.74       133
weighted avg       0.78      0.74      0.74       133
```

KNeighborsClassifier(n_neighbors = 5, weights = 'distance', algorithm = 'brute' , p = 2)

```
              precision    recall  f1-score   support

       False       0.63      0.90      0.74        51
```

```
        True       0.92      0.67      0.77        82

    accuracy                           0.76       133
   macro avg       0.77      0.79      0.76       133
weighted avg       0.81      0.76      0.76       133
```

KNeighborsClassifier(n_neighbors = 5, weights = 'distance', algorithm = 'auto' , p = 2)

```
              precision    recall  f1-score   support

       False       0.63      0.90      0.74        51
        True       0.92      0.67      0.77        82

    accuracy                           0.76       133
   macro avg       0.77      0.79      0.76       133
weighted avg       0.81      0.76      0.76       133
```

KNeighborsClassifier(n_neighbors = 5, weights = 'distance', algorithm = 'ball_tree' , p = 2)

```
              precision    recall  f1-score   support

       False       0.63      0.90      0.74        51
        True       0.92      0.67      0.77        82

    accuracy                           0.76       133
   macro avg       0.77      0.79      0.76       133
weighted avg       0.81      0.76      0.76       133
```

KNeighborsClassifier(n_neighbors = 5, weights = 'distance', algorithm = 'kd_tree' , p = 2)

```
              precision    recall  f1-score   support

       False       0.63      0.90      0.74        51
        True       0.92      0.67      0.77        82

    accuracy                           0.76       133
   macro avg       0.77      0.79      0.76       133
```

```
weighted avg     0.81     0.76     0.76      133
```

KNeighborsClassifier(n_neighbors = 5, weights = 'distance', algorithm = 'brute' , p = 2)

```
              precision    recall  f1-score   support

       False       0.63      0.90      0.74        51
        True       0.92      0.67      0.77        82

    accuracy                           0.76       133
   macro avg       0.77      0.79      0.76       133
weighted avg       0.81      0.76      0.76       133
```

In [17]:

```
print(cm)
```

6.SVM-kernel Classification

SVC(kernel = 'linear', gamma = 'scale' , decision_function_shape = 'ovo' , random_state = 0)

```
              precision    recall  f1-score   support

       False       0.98      0.98      0.98        51
        True       0.99      0.99      0.99        82

    accuracy                           0.98       133
   macro avg       0.98      0.98      0.98       133
weighted avg       0.98      0.98      0.98       133
```

SVC(kernel = 'linear', gamma = 'auto' , decision_function_shape = 'ovo' , random_state = 0)

```
              precision    recall  f1-score   support

       False       0.98      0.98      0.98        51
        True       0.99      0.99      0.99        82
```

```
    accuracy                           0.98       133
   macro avg       0.98      0.98      0.98       133
weighted avg       0.98      0.98      0.98       133
```

```
SVC(kernel = 'linear', gamma = 'auto' , decision_function_shape = 'ovr' ,  random_state = 0)
```

```
              precision    recall  f1-score   support

       False       0.98      0.98      0.98        51
        True       0.99      0.99      0.99        82

    accuracy                           0.98       133
   macro avg       0.98      0.98      0.98       133
weighted avg       0.98      0.98      0.98       133
```

```
SVC(kernel = 'linear', gamma = 'scale' , decision_function_shape = 'ovr' ,  random_state = 0)
```

```
            precision    recall  f1-score   support

       False       0.98      0.98      0.98        51
        True       0.99      0.99      0.99        82

    accuracy                           0.98       133
   macro avg       0.98      0.98      0.98       133
weighted avg       0.98      0.98      0.98       133
```

```
SVC(kernel = 'poly', gamma = 'scale' , decision_function_shape = 'ovo' ,  random_state = 0)
```

```
              precision    recall  f1-score   support

       False       0.00      0.00      0.00        51
        True       0.62      1.00      0.76        82

    accuracy                           0.62       133
   macro avg       0.31      0.50      0.38       133
weighted avg       0.38      0.62      0.47       133
```

```
SVC(kernel = 'rbf', gamma = 'auto' , decision_function_shape = 'ovr' ,  random_state = 0)
```

```
           precision    recall  f1-score   support

       False      0.00      0.00      0.00        51
        True      0.62      1.00      0.76        82

    accuracy                          0.62       133
   macro avg      0.31      0.50      0.38       133
weighted avg      0.38      0.62      0.47       133
```

```
SVC(kernel = 'rbf', gamma = 'auto' , decision_function_shape = 'ovo' ,  random_state = 0)
```

```
           precision    recall  f1-score   support

       False      0.00      0.00      0.00        51
        True      0.62      1.00      0.76        82

    accuracy                          0.62       133
   macro avg      0.31      0.50      0.38       133
weighted avg      0.38      0.62      0.47       133
```

```
SVC(kernel = 'rbf', gamma = 'scale' , decision_function_shape = 'ovo' ,  random_state = 0)
```

```
           precision    recall  f1-score   support

       False      0.00      0.00      0.00        51
        True      0.62      1.00      0.76        82

    accuracy                          0.62       133
   macro avg      0.31      0.50      0.38       133
weighted avg      0.38      0.62      0.47       133
```

```
SVC(kernel = 'rbf', gamma = 'scale' , decision_function_shape = 'ovr' ,  random_state = 0)
```

```
           precision    recall  f1-score   support

     False       0.00      0.00      0.00        51
      True       0.62      1.00      0.76        82

  accuracy                           0.62       133
 macro avg       0.31      0.50      0.38       133
weighted avg       0.38      0.62      0.47       133
```

SVC(kernel = 'sigmoid', gamma = 'scale' , decision_function_shape = 'ovr' , random_state = 0)

```
          precision    recall  f1-score   support

     False       0.27      0.12      0.16        51
      True       0.59      0.80      0.68        82

  accuracy                           0.54       133
 macro avg       0.43      0.46      0.42       133
weighted avg       0.47      0.54      0.48       133
```

SVC(kernel = 'sigmoid', gamma = 'scale' , decision_function_shape = 'ovo' , random_state = 0)

```
          precision    recall  f1-score   support

     False       0.27      0.12      0.16        51
      True       0.59      0.80      0.68        82

  accuracy                           0.54       133
 macro avg       0.43      0.46      0.42       133
weighted avg       0.47      0.54      0.48       133
```

SVC(kernel = 'sigmoid', gamma = 'auto' , decision_function_shape = 'ovo' , random_state = 0)

```
          precision    recall  f1-score   support

     False       0.00      0.00      0.00        51
      True       0.62      1.00      0.76        82

  accuracy                           0.62       133
 macro avg       0.31      0.50      0.38       133
weighted avg       0.38      0.62      0.47       133
```

In [ ]:

```
SVC(kernel = 'sigmoid', gamma = 'auto' , decision_function_shape = 'ovr' ,  random_state = 0)
```

```
              precision    recall  f1-score   support

       False       0.00      0.00      0.00        51
        True       0.62      1.00      0.76        82

    accuracy                           0.62       133
   macro avg       0.31      0.50      0.38       133
weighted avg       0.38      0.62      0.47       133
```

7.DC-GridGrid

```
param_grid = {'criterion':['gini','entropy'],

          'max_features': ['auto','sqrt','log2'],

          'splitter':['best','random']}
```

The report:

```
              precision    recall  f1-score   support

       False       0.94      0.98      0.96        51
        True       0.99      0.96      0.98        82

    accuracy                           0.97       133
   macro avg       0.97      0.97      0.97       133
weighted avg       0.97      0.97      0.97       133
```

8.Grid RandomForestClassifier

```
param_grid = {'criterion':['gini','entropy'],

          'max_features': ['auto','sqrt','log2'],

          'n_estimators':[10,100]}
```

```
          precision    recall  f1-score   support

     False       0.98      0.98      0.98        51
      True       0.99      0.99      0.99        82

  accuracy                           0.98       133
 macro avg       0.98      0.98      0.98       133
weighted avg       0.98      0.98      0.98       133
```

9.Logistic-Grid-Classification

```
param_grid = {'solver':['newton-cg', 'lbfgs', 'liblinear', 'saga'],
          'penalty':['l2']}
```

The report:

```
          precision    recall  f1-score   support

     False       0.98      1.00      0.99        51
      True       1.00      0.99      0.99        82

  accuracy                           0.99       133
 macro avg       0.99      0.99      0.99       133
weighted avg       0.99      0.99      0.99       133
```

10.MultinomialNB

```
          precision    recall  f1-score   support
```

```
       False       0.68      0.98      0.81        51
        True       0.98      0.72      0.83        82

    accuracy                           0.82       133
   macro avg       0.83      0.85      0.82       133
weighted avg       0.87      0.82      0.82       133

[[50  1]
 [23 59]]
```

11.BernoulliNB

```
              precision    recall  f1-score   support

       False       0.86      1.00      0.93        51
        True       1.00      0.90      0.95        82

    accuracy                           0.94       133
   macro avg       0.93      0.95      0.94       133
weighted avg       0.95      0.94      0.94       133

[[51  0]
 [ 8 74]]
```

12.ComplementNB

```
              precision    recall  f1-score   support

       False       0.68      0.98      0.81        51
        True       0.98      0.72      0.83        82

    accuracy                           0.82       133
   macro avg       0.83      0.85      0.82       133
weighted avg       0.87      0.82      0.82       133

[[50  1]
 [23 59]]
```