

```
In [1]: import sklearn
print(sklearn.__version__)
```

1.3.2

```
In [2]: from sklearn.datasets import load_breast_cancer
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn import tree
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.naive_bayes import GaussianNB
from sklearn.ensemble import AdaBoostClassifier
from sklearn import svm
```

```
In [3]: # Loading the dataset
x, y = load_breast_cancer(return_X_y=True)    # return X is always capita
# Splitting the dataset into training and testing sets
x_train, x_test, y_train, y_test = train_test_split(x, y, random_state=0)
```

```
In [4]: x_train , x_test, y_train, y_test = train_test_split(x,y , random_state=0 ,train_size=0.8)
```

```
In [5]: clf_svm = svm.SVC()
dt = tree.DecisionTreeClassifier()
rf = RandomForestClassifier()
lr = LogisticRegression()
nb = GaussianNB()
adaboost = AdaBoostClassifier()
```

```
In [6]: clf_svm.fit(x_train, y_train)
dt.fit(x_train, y_train)
rf.fit(x_train, y_train)
lr.fit(x_train, y_train)
nb.fit(x_train, y_train)
adaboost.fit(x_train, y_train)
```

```
C:\Users\Arooj\AppData\Roaming\Python\Python311\site-packages\sklearn\linear_model\_logistic.py:460: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.
```

```
Increase the number of iterations (max_iter) or scale the data as shown in:
https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
https://scikit-learn.org/stable/modules/linear\_model.html#logistic-regression
n_iter_i = _check_optimize_result(
```

Out[6]: ▾ AdaBoostClassifier

AdaBoostClassifier()

```
In [7]: svm_acc_test = clf_svm.score(x_test, y_test)
svm_acc_train = clf_svm.score(x_train, y_train)

dt_acc = dt.score(x_test, y_test)
rf_acc = rf.score(x_test, y_test)
lr_acc = lr.score(x_test, y_test)
nb_acc = nb.score(x_test, y_test)

adaboost_acc_test = adaboost.score(x_test, y_test)
adaboost_acc_train = adaboost.score(x_train, y_train)
```

```
In [8]: print(svm_acc_test)
print(svm_acc_train)
```

```
0.9298245614035088
0.9098901098901099
```

```
In [9]: cv_svm = cross_val_score(clf_svm, x,y, cv=5)

cv_dt = cross_val_score(dt, x,y, cv=5)
cv_rf = cross_val_score(rf, x,y, cv=5)
cv_lr = cross_val_score(lr, x,y, cv=5)
cv_nb = cross_val_score(nb, x,y, cv=5)
cv_adaboost = cross_val_score(adaboost, x, y, cv=5)
print (cv_svm)

print (cv_dt)
print (cv_rf)
print (cv_lr)
print (cv_nb)

print (cv_adaboost)
```

```
C:\Users\Arooj\AppData\Roaming\Python\Python311\site-packages\sklearn\linear_model\_logistic.py:460: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```
n_iter_i = _check_optimize_result(
```

```
C:\Users\Arooj\AppData\Roaming\Python\Python311\site-packages\sklearn\linear_model\_logistic.py:460: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```
n_iter_i = _check_optimize_result(
```

```
C:\Users\Arooj\AppData\Roaming\Python\Python311\site-packages\sklearn\linear_model\_logistic.py:460: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```
n_iter_i = _check_optimize_result(
```

```
C:\Users\Arooj\AppData\Roaming\Python\Python311\site-packages\sklearn\linear_model\_logistic.py:460: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```
n_iter_i = _check_optimize_result(
```

```
C:\Users\Arooj\AppData\Roaming\Python\Python311\site-packages\sklearn\linear_model\_logistic.py:460: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.
```

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

```
n_iter_i = _check_optimize_result(  
[0.85087719 0.89473684 0.92982456 0.94736842 0.9380531 ]  
[0.9122807 0.92105263 0.9122807 0.92982456 0.91150442]  
[0.92982456 0.94736842 0.98245614 0.97368421 0.97345133]  
[0.92982456 0.93859649 0.95614035 0.9122807 0.95575221]  
[0.92105263 0.92105263 0.94736842 0.94736842 0.95575221]  
[0.93859649 0.96491228 0.99122807 0.99122807 0.96460177])
```

In [10]: print (cv_svm * 100)

```
print (cv_dt * 100)  
print (cv_rf * 100)  
print (cv_lr * 100)  
print (cv_nb * 100)  
print(cv_adaboost * 100)
```

```
[85.0877193 89.47368421 92.98245614 94.73684211 93.80530973]  
[91.22807018 92.10526316 91.22807018 92.98245614 91.15044248]  
[92.98245614 94.73684211 98.24561404 97.36842105 97.34513274]  
[92.98245614 93.85964912 95.61403509 91.22807018 95.57522124]  
[92.10526316 92.10526316 94.73684211 94.73684211 95.57522124]  
[93.85964912 96.49122807 99.12280702 99.12280702 96.46017699]
```

In [11]: print (cv_svm.mean())

```
print (cv_dt.mean())  
print (cv_rf.mean())  
print (cv_lr.mean())  
print (cv_nb.mean())  
print (cv_adaboost.mean())
```

```
0.9121720229777983  
0.9173886042539978  
0.9613569321533924  
0.9385188635305075  
0.9385188635305075  
0.9701133364384411
```

```
In [12]: print (cv_svm.std())  
  
         print (cv_dt.std())  
         print (cv_rf.std())  
         print (cv_lr.std())  
         print (cv_nb.std())  
         print(cv_adaboost.mean())
```

```
0.03544367440584207  
0.0071397903393633395  
0.019664087183305364  
0.016562265413438516  
0.014585994424363306  
0.9701133364384411
```

```
In [ ]:
```

```
In [ ]:
```