import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns  
from sklearn.datasets import make\_classification  
from sklearn.linear\_model import LogisticRegression  
from sklearn.model\_selection import train\_test\_split  
from sklearn.metrics import confusion\_matrix  
import pandas as pd  
df=pd.read\_csv("ASD.csv")

df.head()

id A1\_Score A2\_Score A3\_Score A4\_Score A5\_Score A6\_Score A7\_Score \  
0 1 1 1 1 1 0 0 1   
1 2 1 1 0 1 0 0 0   
2 3 1 1 0 1 1 0 1   
3 4 1 1 0 1 0 0 1   
4 5 1 0 0 0 0 0 0   
  
 A8\_Score A9\_Score ... gender ethnicity jundice austim \  
0 1 0 ... f White-European no no   
1 1 0 ... m Latino no yes   
2 1 1 ... m Latino yes yes   
3 1 0 ... f White-European no yes   
4 1 0 ... f ? no no   
  
 contry\_of\_res used\_app\_before result age\_desc relation Class/ASD   
0 United States no 6 18 and more Self NO   
1 Brazil no 5 18 and more Self NO   
2 Spain no 8 18 and more Parent YES   
3 United States no 6 18 and more Self NO   
4 Egypt no 2 18 and more ? NO   
  
[5 rows x 22 columns]

df.isnull().sum()

id 0  
A1\_Score 0  
A2\_Score 0  
A3\_Score 0  
A4\_Score 0  
A5\_Score 0  
A6\_Score 0  
A7\_Score 0  
A8\_Score 0  
A9\_Score 0  
A10\_Score 0  
age 0  
gender 0  
ethnicity 0  
jundice 0  
austim 0  
contry\_of\_res 0  
used\_app\_before 0  
result 0  
age\_desc 0  
relation 0  
Class/ASD 0  
dtype: int64

df.info()

<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 704 entries, 0 to 703  
Data columns (total 22 columns):  
 # Column Non-Null Count Dtype   
--- ------ -------------- -----   
 0 id 704 non-null int64   
 1 A1\_Score 704 non-null int64   
 2 A2\_Score 704 non-null int64   
 3 A3\_Score 704 non-null int64   
 4 A4\_Score 704 non-null int64   
 5 A5\_Score 704 non-null int64   
 6 A6\_Score 704 non-null int64   
 7 A7\_Score 704 non-null int64   
 8 A8\_Score 704 non-null int64   
 9 A9\_Score 704 non-null int64   
 10 A10\_Score 704 non-null int64   
 11 age 704 non-null object  
 12 gender 704 non-null object  
 13 ethnicity 704 non-null object  
 14 jundice 704 non-null object  
 15 austim 704 non-null object  
 16 contry\_of\_res 704 non-null object  
 17 used\_app\_before 704 non-null object  
 18 result 704 non-null int64   
 19 age\_desc 704 non-null object  
 20 relation 704 non-null object  
 21 Class/ASD 704 non-null object  
dtypes: int64(12), object(10)  
memory usage: 121.1+ KB

df.isnull().sum()

id 0  
A1\_Score 0  
A2\_Score 0  
A3\_Score 0  
A4\_Score 0  
A5\_Score 0  
A6\_Score 0  
A7\_Score 0  
A8\_Score 0  
A9\_Score 0  
A10\_Score 0  
age 0  
gender 0  
ethnicity 0  
jundice 0  
austim 0  
contry\_of\_res 0  
used\_app\_before 0  
result 0  
age\_desc 0  
relation 0  
Class/ASD 0  
dtype: int64

df["ethnicity"] = df["ethnicity"].replace("?", "No\_Ethinicy")

df["ethnicity"].value\_counts()

White-European 233  
Asian 123  
No\_Ethinicy 95  
Middle Eastern 92  
Black 43  
South Asian 36  
Others 30  
Latino 20  
Hispanic 13  
Pasifika 12  
Turkish 6  
others 1  
Name: ethnicity, dtype: int64

df["age"] = df["age"].replace("?", "55")

df["A1\_Score"].value\_counts()

1 508  
0 196  
Name: A1\_Score, dtype: int64

df["A2\_Score"].value\_counts()

0 385  
1 319  
Name: A2\_Score, dtype: int64

df["A3\_Score"].value\_counts()

0 382  
1 322  
Name: A3\_Score, dtype: int64

df["A4\_Score"].value\_counts()

0 355  
1 349  
Name: A4\_Score, dtype: int64

df["A5\_Score"].value\_counts()

0 353  
1 351  
Name: A5\_Score, dtype: int64

df["A6\_Score"].value\_counts()

0 504  
1 200  
Name: A6\_Score, dtype: int64

df["A7\_Score"].value\_counts()

0 410  
1 294  
Name: A7\_Score, dtype: int64

df["A8  
   
 \_Score"].value\_counts()

File "C:\Users\siddh\AppData\Local\Temp\ipykernel\_18180\3141580971.py", line 1  
 df["A8  
 ^  
SyntaxError: EOL while scanning string literal

df["A9\_Score"].value\_counts()

0 476  
1 228  
Name: A9\_Score, dtype: int64

df["A10\_Score"].value\_counts()

1 404  
0 300  
Name: A10\_Score, dtype: int64

df["gender"].value\_counts()

m 367  
f 337  
Name: gender, dtype: int64

df["jundice"].value\_counts()

no 635  
yes 69  
Name: jundice, dtype: int64

df["austim"].value\_counts()

no 613  
yes 91  
Name: austim, dtype: int64

df["used\_app\_before"].value\_counts()

no 692  
yes 12  
Name: used\_app\_before, dtype: int64

df["result"].value\_counts()

4 131  
3 110  
5 83  
2 74  
6 70  
7 57  
8 55  
9 47  
1 33  
10 30  
0 14  
Name: result, dtype: int64

df["Class/ASD"].value\_counts()

NO 515  
YES 189  
Name: Class/ASD, dtype: int64

df=df.drop(['id'], axis=1)  
df=df.drop(['used\_app\_before'], axis=1)  
df=df.drop(['result'], axis=1)  
df=df.drop(['age\_desc'], axis=1)  
df=df.drop(['relation'], axis=1)

df.head()

A1\_Score A2\_Score A3\_Score A4\_Score A5\_Score A6\_Score A7\_Score \  
0 1 1 1 1 0 0 1   
1 1 1 0 1 0 0 0   
2 1 1 0 1 1 0 1   
3 1 1 0 1 0 0 1   
4 1 0 0 0 0 0 0   
  
 A8\_Score A9\_Score A10\_Score age gender ethnicity jundice austim \  
0 1 0 0 26 f White-European no no   
1 1 0 1 24 m Latino no yes   
2 1 1 1 27 m Latino yes yes   
3 1 0 1 35 f White-European no yes   
4 1 0 0 40 f No\_Ethinicy no no   
  
 contry\_of\_res Class/ASD   
0 United States NO   
1 Brazil NO   
2 Spain YES   
3 United States NO   
4 Egypt NO

from sklearn.preprocessing import LabelEncoder  
le = LabelEncoder()  
df["gender"] = le.fit\_transform(df["gender"])  
df["gender"].value\_counts()

1 367  
0 337  
Name: gender, dtype: int64

print(le.inverse\_transform([0]))  
print(le.inverse\_transform([1]))

['f']  
['m']

df["ethnicity"] = le.fit\_transform(df["ethnicity"])  
df["ethnicity"].value\_counts()

10 233  
0 123  
5 95  
4 92  
1 43  
8 36  
6 30  
3 20  
2 13  
7 12  
9 6  
11 1  
Name: ethnicity, dtype: int64

print(le.inverse\_transform([0]))  
print(le.inverse\_transform([1]))  
print(le.inverse\_transform([2]))  
print(le.inverse\_transform([3]))  
print(le.inverse\_transform([4]))  
print(le.inverse\_transform([5]))  
print(le.inverse\_transform([6]))  
print(le.inverse\_transform([7]))  
print(le.inverse\_transform([8]))  
print(le.inverse\_transform([9]))  
print(le.inverse\_transform([10]))  
print(le.inverse\_transform([11]))

['Asian']  
['Black']  
['Hispanic']  
['Latino']  
['Middle Eastern ']  
['No\_Ethinicy']  
['Others']  
['Pasifika']  
['South Asian']  
['Turkish']  
['White-European']  
['others']

df["jundice"] = le.fit\_transform(df["jundice"])  
df["jundice"].value\_counts()

0 635  
1 69  
Name: jundice, dtype: int64

print(le.inverse\_transform([0]))  
print(le.inverse\_transform([1]))

['no']  
['yes']

df["austim"] = le.fit\_transform(df["austim"])  
df["austim"].value\_counts()

0 613  
1 91  
Name: austim, dtype: int64

print(le.inverse\_transform([0]))  
print(le.inverse\_transform([1]))

['no']  
['yes']

df["contry\_of\_res"] = le.fit\_transform(df["contry\_of\_res"])  
df["contry\_of\_res"].value\_counts()

64 113  
62 82  
43 81  
29 81  
63 77  
 ...   
17 1  
16 1  
38 1  
14 1  
19 1  
Name: contry\_of\_res, Length: 67, dtype: int64

for i in range(64):  
 print(le.inverse\_transform([i]))

['Afghanistan']  
['AmericanSamoa']  
['Angola']  
['Argentina']  
['Armenia']  
['Aruba']  
['Australia']  
['Austria']  
['Azerbaijan']  
['Bahamas']  
['Bangladesh']  
['Belgium']  
['Bolivia']  
['Brazil']  
['Burundi']  
['Canada']  
['Chile']  
['China']  
['Costa Rica']  
['Cyprus']  
['Czech Republic']  
['Ecuador']  
['Egypt']  
['Ethiopia']  
['Finland']  
['France']  
['Germany']  
['Hong Kong']  
['Iceland']  
['India']  
['Indonesia']  
['Iran']  
['Iraq']  
['Ireland']  
['Italy']  
['Japan']  
['Jordan']  
['Kazakhstan']  
['Lebanon']  
['Malaysia']  
['Mexico']  
['Nepal']  
['Netherlands']  
['New Zealand']  
['Nicaragua']  
['Niger']  
['Oman']  
['Pakistan']  
['Philippines']  
['Portugal']  
['Romania']  
['Russia']  
['Saudi Arabia']  
['Serbia']  
['Sierra Leone']  
['South Africa']  
['Spain']  
['Sri Lanka']  
['Sweden']  
['Tonga']  
['Turkey']  
['Ukraine']  
['United Arab Emirates']  
['United Kingdom']

df["Class/ASD"] = le.fit\_transform(df["Class/ASD"])  
df["Class/ASD"].value\_counts()

0 515  
1 189  
Name: Class/ASD, dtype: int64

print(le.inverse\_transform([0]))  
print(le.inverse\_transform([1]))

['NO']  
['YES']

df.head()

A1\_Score A2\_Score A3\_Score A4\_Score A5\_Score A6\_Score A7\_Score \  
0 1 1 1 1 0 0 1   
1 1 1 0 1 0 0 0   
2 1 1 0 1 1 0 1   
3 1 1 0 1 0 0 1   
4 1 0 0 0 0 0 0   
  
 A8\_Score A9\_Score A10\_Score age gender ethnicity jundice austim \  
0 1 0 0 26 0 10 0 0   
1 1 0 1 24 1 3 0 1   
2 1 1 1 27 1 3 1 1   
3 1 0 1 35 0 10 0 1   
4 1 0 0 40 0 5 0 0   
  
 contry\_of\_res Class/ASD   
0 64 0   
1 13 0   
2 56 1   
3 64 0   
4 22 0

from sklearn.datasets import make\_classification  
from sklearn.linear\_model import LogisticRegression  
from sklearn.model\_selection import train\_test\_split  
from sklearn.metrics import confusion\_matrix  
from sklearn.metrics import accuracy\_score

df["age"] = df["age"].replace("?", "55")

x=df.drop(["Class/ASD"],axis=1)  
y=df["Class/ASD"]  
print(x)

A1\_Score A2\_Score A3\_Score A4\_Score A5\_Score A6\_Score A7\_Score \  
0 1 1 1 1 0 0 1   
1 1 1 0 1 0 0 0   
2 1 1 0 1 1 0 1   
3 1 1 0 1 0 0 1   
4 1 0 0 0 0 0 0   
.. ... ... ... ... ... ... ...   
699 0 1 0 1 1 0 1   
700 1 0 0 0 0 0 0   
701 1 0 1 1 1 0 1   
702 1 0 0 1 1 0 1   
703 1 0 1 1 1 0 1   
  
 A8\_Score A9\_Score A10\_Score age gender ethnicity jundice austim \  
0 1 0 0 26 0 10 0 0   
1 1 0 1 24 1 3 0 1   
2 1 1 1 27 1 3 1 1   
3 1 0 1 35 0 10 0 1   
4 1 0 0 40 0 5 0 0   
.. ... ... ... .. ... ... ... ...   
699 1 1 1 25 0 10 0 0   
700 1 0 1 34 1 2 0 0   
701 1 0 1 24 0 5 0 0   
702 0 1 1 35 1 8 0 0   
703 1 1 1 26 0 10 0 0   
  
 contry\_of\_res   
0 64   
1 13   
2 56   
3 64   
4 22   
.. ...   
699 51   
700 40   
701 51   
702 47   
703 19   
  
[704 rows x 16 columns]

print(y)

0 0  
1 0  
2 1  
3 0  
4 0  
 ..  
699 1  
700 0  
701 1  
702 0  
703 1  
Name: Class/ASD, Length: 704, dtype: int32

x\_train,x\_test,y\_train,y\_test=train\_test\_split(x,y,test\_size=0.2,random\_state=42)

modelLR=LogisticRegression()  
modelLR.fit(x\_train,y\_train)

C:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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(

LogisticRegression()

y\_predLR = model.predict(x\_test)  
accuracy = accuracy\_score(y\_test, y\_predLR)  
print("Accuracy:", accuracy)

Accuracy: 0.7872340425531915

C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)

confusion\_matrix(y\_test,y\_predLR)

array([[103, 2],  
 [ 28, 8]], dtype=int64)

from sklearn.svm import SVC  
modelSVC=SVC()  
modelSVC.fit(x\_train,y\_train)  
y\_pred\_SVM=classifier.predict(x\_test)

accuracy = accuracy\_score(y\_test, y\_pred\_SVM)  
print("Accuracy:", accuracy)

Accuracy: 0.7446808510638298

confusion\_matrix(y\_test,y\_pred\_SVM)

array([[105, 0],  
 [ 36, 0]], dtype=int64)

from sklearn.naive\_bayes import GaussianNB  
modelGAUS = GaussianNB()  
modelGAUS.fit(x\_train, y\_train)  
y\_predGAUS = model.predict(x\_test)  
accuracy = accuracy\_score(y\_test, y\_predGAUS)  
print("Accuracy:", accuracy)

Accuracy: 0.9858156028368794

confusion\_matrix(y\_test,y\_predGAUS)

array([[104, 1],  
 [ 1, 35]], dtype=int64)

from sklearn.neighbors import KNeighborsClassifier  
modelKNN = KNeighborsClassifier(n\_neighbors=10)  
modelKNN.fit(x\_train, y\_train)  
y\_predKNN = model.predict(x\_test)  
accuracy = accuracy\_score(y\_test, y\_pred)  
print("Accuracy:", accuracy)

Accuracy: 0.8581560283687943

confusion\_matrix(y\_test,y\_predKNN)

array([[103, 2],  
 [ 28, 8]], dtype=int64)

from sklearn.tree import DecisionTreeClassifier  
clf = DecisionTreeClassifier()  
clf.fit(x\_train, y\_train)

DecisionTreeClassifier()

y\_predDTC = clf.predict(x\_test)  
from sklearn.metrics import accuracy\_score  
print("Accuracy:", accuracy\_score(y\_test, y\_pred))

Accuracy: 0.8581560283687943

confusion\_matrix(y\_test,y\_predDTC)

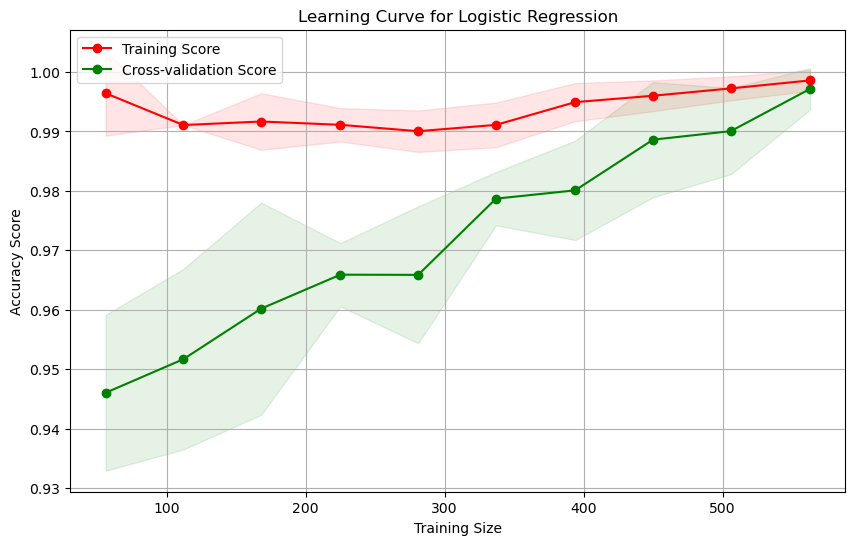
array([[89, 16],  
 [ 4, 32]], dtype=int64)

from sklearn.model\_selection import learning\_curve  
train\_sizes = np.linspace(0.1, 1.0, 10)  
train\_sizes, train\_scores, validation\_scores = learning\_curve(  
 modelLR, x, y, train\_sizes=train\_sizes, cv=5, scoring='accuracy')

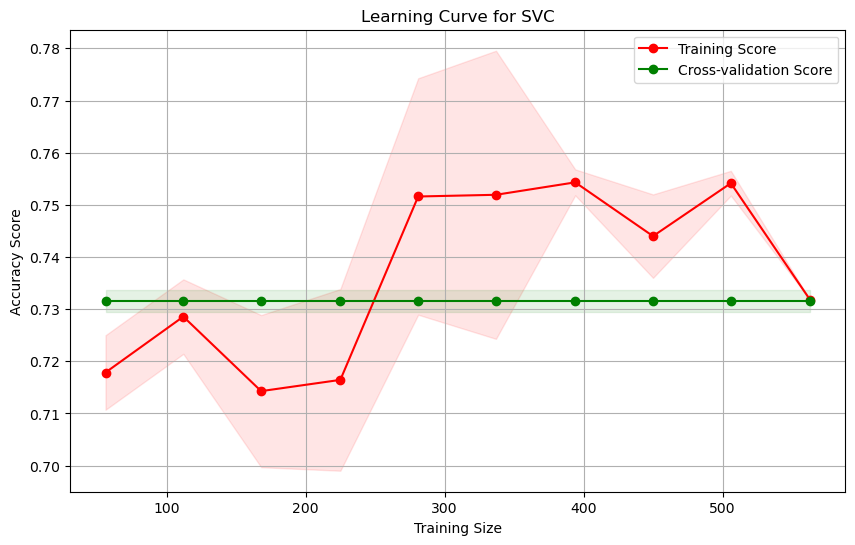
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: ConvergenceWarning: lbfgs failed to converge (status=1):  
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.  
  
Increase the number of iterations (max\_iter) or scale the data as shown in:  
 https://scikit-learn.org/stable/modules/preprocessing.html  
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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear\_model\\_logistic.py:814: 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(

train\_scores\_mean = np.mean(train\_scores, axis=1)  
train\_scores\_std = np.std(train\_scores, axis=1)  
validation\_scores\_mean = np.mean(validation\_scores, axis=1)  
validation\_scores\_std = np.std(validation\_scores, axis=1)

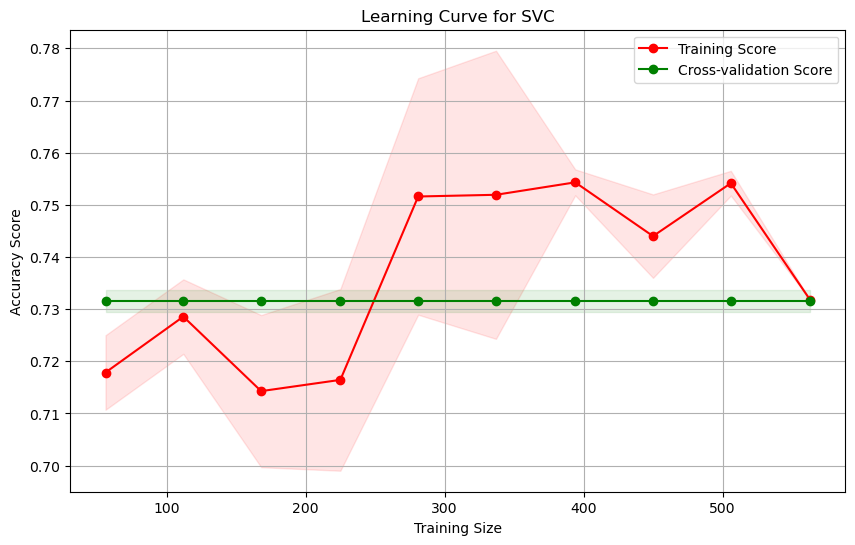
plt.figure(figsize=(10, 6))  
plt.title("Learning Curve for Logistic Regression")  
plt.xlabel("Training Size")  
plt.ylabel("Accuracy Score")  
plt.grid()  
plt.fill\_between(train\_sizes, train\_scores\_mean - train\_scores\_std,  
 train\_scores\_mean + train\_scores\_std, alpha=0.1, color="r")  
plt.fill\_between(train\_sizes, validation\_scores\_mean - validation\_scores\_std,  
 validation\_scores\_mean + validation\_scores\_std, alpha=0.1, color="g")  
plt.plot(train\_sizes, train\_scores\_mean, 'o-', color="r",  
 label="Training Score")  
plt.plot(train\_sizes, validation\_scores\_mean, 'o-', color="g",  
 label="Cross-validation Score")  
plt.legend(loc="best")  
plt.show()



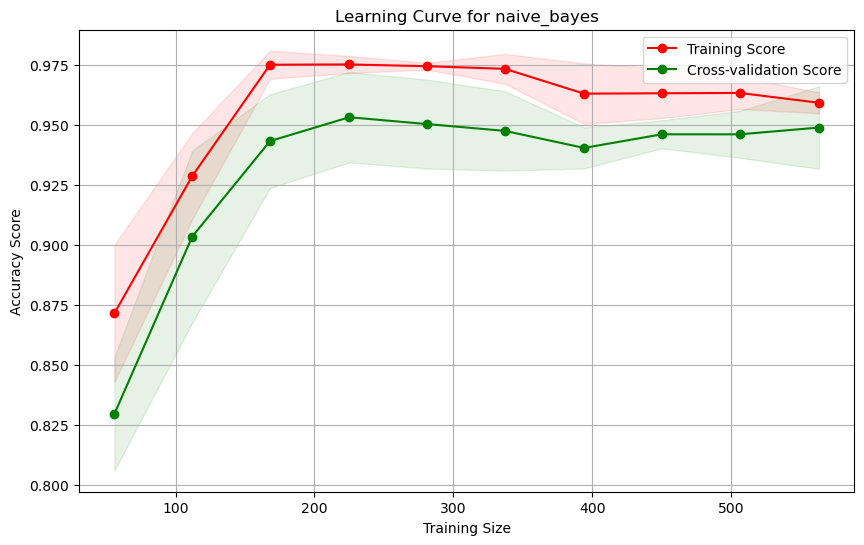
from sklearn.model\_selection import learning\_curve  
train\_sizes = np.linspace(0.1, 1.0, 10)  
train\_sizes, train\_scores, validation\_scores = learning\_curve(  
 modelSVC, x, y, train\_sizes=train\_sizes, cv=5, scoring='accuracy')  
train\_scores\_mean = np.mean(train\_scores, axis=1)  
train\_scores\_std = np.std(train\_scores, axis=1)  
validation\_scores\_mean = np.mean(validation\_scores, axis=1)  
validation\_scores\_std = np.std(validation\_scores, axis=1)  
plt.figure(figsize=(10, 6))  
plt.title("Learning Curve for SVC")  
plt.xlabel("Training Size")  
plt.ylabel("Accuracy Score")  
plt.grid()  
plt.fill\_between(train\_sizes, train\_scores\_mean - train\_scores\_std,  
 train\_scores\_mean + train\_scores\_std, alpha=0.1, color="r")  
plt.fill\_between(train\_sizes, validation\_scores\_mean - validation\_scores\_std,  
 validation\_scores\_mean + validation\_scores\_std, alpha=0.1, color="g")  
plt.plot(train\_sizes, train\_scores\_mean, 'o-', color="r",  
 label="Training Score")  
plt.plot(train\_sizes, validation\_scores\_mean, 'o-', color="g",  
 label="Cross-validation Score")  
plt.legend(loc="best")  
plt.show()



from sklearn.model\_selection import learning\_curve  
train\_sizes = np.linspace(0.1, 1.0, 10)  
train\_sizes, train\_scores, validation\_scores = learning\_curve(  
 modelSVC, x, y, train\_sizes=train\_sizes, cv=5, scoring='accuracy')  
train\_scores\_mean = np.mean(train\_scores, axis=1)  
train\_scores\_std = np.std(train\_scores, axis=1)  
validation\_scores\_mean = np.mean(validation\_scores, axis=1)  
validation\_scores\_std = np.std(validation\_scores, axis=1)  
plt.figure(figsize=(10, 6))  
plt.title("Learning Curve for SVC")  
plt.xlabel("Training Size")  
plt.ylabel("Accuracy Score")  
plt.grid()  
plt.fill\_between(train\_sizes, train\_scores\_mean - train\_scores\_std,  
 train\_scores\_mean + train\_scores\_std, alpha=0.1, color="r")  
plt.fill\_between(train\_sizes, validation\_scores\_mean - validation\_scores\_std,  
 validation\_scores\_mean + validation\_scores\_std, alpha=0.1, color="g")  
plt.plot(train\_sizes, train\_scores\_mean, 'o-', color="r",  
 label="Training Score")  
plt.plot(train\_sizes, validation\_scores\_mean, 'o-', color="g",  
 label="Cross-validation Score")  
plt.legend(loc="best")  
plt.show()

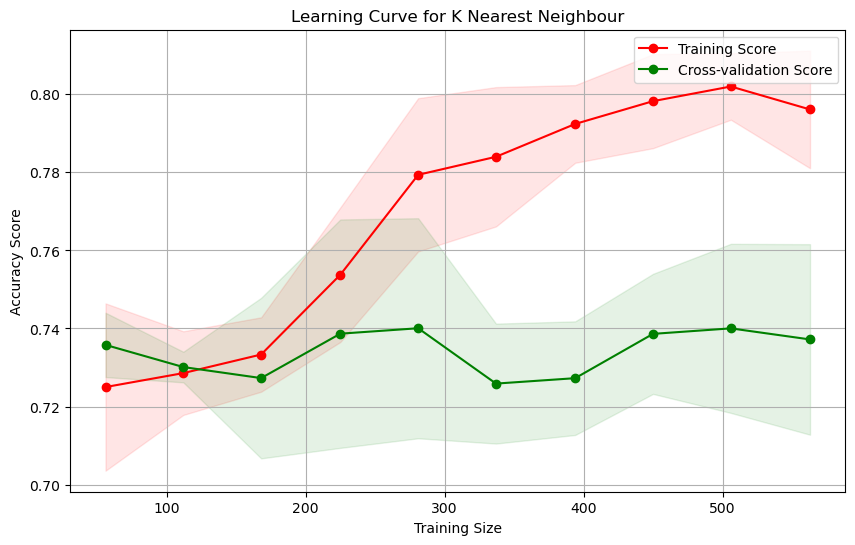


from sklearn.model\_selection import learning\_curve  
train\_sizes = np.linspace(0.1, 1.0, 10)  
train\_sizes, train\_scores, validation\_scores = learning\_curve(  
 modelGAUS, x, y, train\_sizes=train\_sizes, cv=5, scoring='accuracy')  
train\_scores\_mean = np.mean(train\_scores, axis=1)  
train\_scores\_std = np.std(train\_scores, axis=1)  
validation\_scores\_mean = np.mean(validation\_scores, axis=1)  
validation\_scores\_std = np.std(validation\_scores, axis=1)  
plt.figure(figsize=(10, 6))  
plt.title("Learning Curve for naive\_bayes")  
plt.xlabel("Training Size")  
plt.ylabel("Accuracy Score")  
plt.grid()  
plt.fill\_between(train\_sizes, train\_scores\_mean - train\_scores\_std,  
 train\_scores\_mean + train\_scores\_std, alpha=0.1, color="r")  
plt.fill\_between(train\_sizes, validation\_scores\_mean - validation\_scores\_std,  
 validation\_scores\_mean + validation\_scores\_std, alpha=0.1, color="g")  
plt.plot(train\_sizes, train\_scores\_mean, 'o-', color="r",  
 label="Training Score")  
plt.plot(train\_sizes, validation\_scores\_mean, 'o-', color="g",  
 label="Cross-validation Score")  
plt.legend(loc="best")  
plt.show()



from sklearn.model\_selection import learning\_curve  
train\_sizes = np.linspace(0.1, 1.0, 10)  
train\_sizes, train\_scores, validation\_scores = learning\_curve(  
 modelKNN, x, y, train\_sizes=train\_sizes, cv=5, scoring='accuracy')  
train\_scores\_mean = np.mean(train\_scores, axis=1)  
train\_scores\_std = np.std(train\_scores, axis=1)  
validation\_scores\_mean = np.mean(validation\_scores, axis=1)  
validation\_scores\_std = np.std(validation\_scores, axis=1)  
plt.figure(figsize=(10, 6))  
plt.title("Learning Curve for K Nearest Neighbour")  
plt.xlabel("Training Size")  
plt.ylabel("Accuracy Score")  
plt.grid()  
plt.fill\_between(train\_sizes, train\_scores\_mean - train\_scores\_std,  
 train\_scores\_mean + train\_scores\_std, alpha=0.1, color="r")  
plt.fill\_between(train\_sizes, validation\_scores\_mean - validation\_scores\_std,  
 validation\_scores\_mean + validation\_scores\_std, alpha=0.1, color="g")  
plt.plot(train\_sizes, train\_scores\_mean, 'o-', color="r",  
 label="Training Score")  
plt.plot(train\_sizes, validation\_scores\_mean, 'o-', color="g",  
 label="Cross-validation Score")  
plt.legend(loc="best")  
plt.show()

C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)  
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\neighbors\\_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.  
 mode, \_ = stats.mode(\_y[neigh\_ind, k], axis=1)



from sklearn.model\_selection import learning\_curve  
train\_sizes = np.linspace(0.1, 1.0, 10)  
train\_sizes, train\_scores, validation\_scores = learning\_curve(  
 clf, x, y, train\_sizes=train\_sizes, cv=5, scoring='accuracy')  
train\_scores\_mean = np.mean(train\_scores, axis=1)  
train\_scores\_std = np.std(train\_scores, axis=1)  
validation\_scores\_mean = np.mean(validation\_scores, axis=1)  
validation\_scores\_std = np.std(validation\_scores, axis=1)  
plt.figure(figsize=(10, 6))  
plt.title("Learning Curve for Decesion Tree Classifier")  
plt.xlabel("Training Size")  
plt.ylabel("Accuracy Score")  
plt.grid()  
plt.fill\_between(train\_sizes, train\_scores\_mean - train\_scores\_std,  
 train\_scores\_mean + train\_scores\_std, alpha=0.1, color="r")  
plt.fill\_between(train\_sizes, validation\_scores\_mean - validation\_scores\_std,  
 validation\_scores\_mean + validation\_scores\_std, alpha=0.1, color="g")  
plt.plot(train\_sizes, train\_scores\_mean, 'o-', color="r",  
 label="Training Score")  
plt.plot(train\_sizes, validation\_scores\_mean, 'o-', color="g",  
 label="Cross-validation Score")  
plt.legend(loc="best")  
plt.show()

