**M12 Lab Naive Bayes Classifier**

Write a script that fits a Gaussian Naive Bayes classifier to the given dataset. The script should:

1. Read the dataset using proper function for reading a libsvm file.
2. Create a machine learning pipeline that includes a standard scalar and a [Gaussian Naive Bayes Classifier (Links to an external site.)](https://scikit-learn.org/stable/modules/naive_bayes.html).
3. Fit the pipeline to the data using k-fold cross validation with k=5.

from sklearn.datasets import load\_svmlight\_file

from sklearn.preprocessing import StandardScaler

from sklearn.tree import DecisionTreeClassifier

from sklearn.preprocessing import StandardScaler

from sklearn.pipeline import make\_pipeline

from sklearn.model\_selection import cross\_val\_score

from sklearn.naive\_bayes import GaussianNB

from sklearn.model\_selection import KFold

print("Loading Dataset...")

X,y = load\_svmlight\_file("a9a.txt")

print("Loaded!!")

print("Create a Nive Bayse Object ")

#clf = DecisionTreeClassifier(random\_state=0)

gnb = GaussianNB()

print("Create a Pipeline that contains two steps for SS -> gnb")

pipe = make\_pipeline(StandardScaler(with\_mean=0,with\_std=1), gnb)

pipe

print("Use the Cross Validation Score function & Print the average score")

cv = KFold(n\_splits=5, random\_state=1, shuffle=True)

print(cv)

scores = cross\_val\_score(pipe, X.todense(), y, cv=cv)

print(scores)

print("Training Complete!")

acc = scores.mean()

print("Cross Validation Mean Accuracy = %0.2f" % acc )

Graphical user interface, text, application

Description automatically generatedGraphical user interface, text, application, email

Description automatically generated

1. What is the accuracy returned by the cross validation function?

Obtained Cross Validation accuracy is 0.50.

1. Run your code, make sure it does not contain any errors. What is the obtained classification accuracy?

Code has no Errors.

Obtained Classification Accuracy is [0.48579764 0.51136364 0.51996314 0.49201474 0.48372236].

**Decision Tree**

1. Create another pipeline that includes a standard scalar and a Decision Tree Classifier.
2. Use a parameter grid that evaluates two impurity calculation metrics: entropy & gini, and a maximum tree depth of 10, 50 & 100.
3. Fit this pipeline to the data using k-fold cross validation with k=5.

print("Loading Dataset...")

X,y = load\_svmlight\_file("a9a.txt")

print("Loaded!!")

print("Create a standard Scaler Object: SS")

scaler = StandardScaler(with\_mean=0, with\_std=1)

print("Create a Decision Tree Object: DT")

clf = DecisionTreeClassifier(random\_state=0 )

# from sklearn.pipeline import Pipeline

print("Create a Pipeline that contains two steps for SS -> DT")

pipe =make\_pipeline(scaler, clf)

pipe

param\_grid ={'decisiontreeclassifier\_\_criterion':['entropy', 'gini'],'decisiontreeclassifier\_\_max\_depth' :[10,50,100]}

from sklearn.model\_selection import GridSearchCV

print("Use the Cross Validation Score function & Print the average score")

cv = KFold(n\_splits=5, random\_state=1, shuffle=True)

print(cv)

search = GridSearchCV(estimator=pipe, param\_grid=param\_grid, cv=cv)

search.fit(X,y)

mean = search.cv\_results\_["mean\_test\_score"]

print ("Mean Test Scores are ", mean )

print(search.best\_score\_)

Graphical user interface, text, application

Description automatically generatedGraphical user interface, text, application, email

Description automatically generated

1. Run your code, make sure it does not contain any errors. What is the obtained classification accuracy?

[0.83151623 0.79681209 0.79681209 0.83237602 0.79435517 0.79435517]

Best Score is 0.8323760200257206

Modify your script by adding a part that fits an SVM classifier to the data:

**SVM**

Modify your script by adding a part that fits an SVM classifier to the data:

1. Create another pipeline that includes a standard scalar and an SVM classifier.
2. Use a parameter grid that evaluates three kernels: linear, polynomial (with d values = 2 & 3), rbf (with gamma values = 0.001, 0.1, 2).
3. Fit this pipeline to the data using k-fold cross validation with k=5.

from sklearn.datasets import load\_svmlight\_file

from sklearn import svm

import pandas as pd

from sklearn.model\_selection import GridSearchCV

print("Loading Dataset...")

X,y = load\_svmlight\_file("a9a.txt")

# from sklearn.pipeline import Pipeline

print("Create a Pipeline that contains two steps for SS -> DT")

svc = svm.SVC()

pipe =make\_pipeline(scaler, svc)

pipe

print("Creating Parameter Grid...")

param\_grid = [

{ 'svc\_\_kernel': ['linear']},

{ 'svc\_\_gamma': [0.001, 0.1, 2], 'svc\_\_kernel': ['rbf']},

{ 'svc\_\_degree': [2,3], 'svc\_\_kernel': ['poly']},

]

from sklearn.model\_selection import GridSearchCV

print("Use the Cross Validation Score function & Print the average score")

cv = KFold(n\_splits=5, random\_state=1, shuffle=True)

print(cv)

search = GridSearchCV(estimator=pipe, param\_grid=param\_grid, cv=cv)

search.fit(X,y)

mean = search.cv\_results\_["mean\_test\_score"]

print ("Mean Test Scores are ", mean )

print(search.best\_score\_)

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

1. Run your code, make sure it does not contain any errors. What is the obtained classification accuracy?

[0.84773192, 0.84588921, 0.80716182, 0.77887639, 0.8462884, 0.84493715]

Best score: - 0.8477319222828206

which classifier gives the highest classification accuracy?

SVM gives the Higher Performance compared to Decision Tree and Naïve Bayes Classifiers.