The original code implemented these steps:

1. read the libsvm data file
2. define a SVM-Classify object
3. call the cross validation function to fit the SVM classifier using some k cross validation counts.
4. print the mean & standard deviation of the obtained classification accuracies from the multiple fits.

The code becomes:

from sklearn.datasets import load\_svmlight\_file  
from sklearn import svm  
from sklearn.model\_selection import cross\_val\_score   
  
print("Loading Dataset...")  
X,y = load\_svmlight\_file("a9a")  
print("Creating classifier object...")  
clf = svm.SVC(kernel='linear', C=1, random\_state= 42)   
print("Training classifier with cross validation, k=5")  
scores = cross\_val\_score(clf, X, y, cv=5)  
print("Training Complete!")  
acc = scores.mean()   
stdiv = scores.std()  
print("Cross Validation Mean Accuracy = %0.2f" % acc )  
print("Standard Deviation of the Mean Accuracy across all runs = %0.2f" % stdiv)

Graphical user interface, text, application

Description automatically generated

Run this code. What is the obtained result?

The Obtained Result is 0.83

Modify the code so that it prints the confusion matrix instead of the mean accuracy.

* Instead of using the ***cross\_val\_score()*** function, it uses the ***cross\_val\_predict()*** function which returns the predicted class labels, call them ***y\_pred***.  Read more about this function here: [https://scikit-learn.org/stable/modules/generated/sklearn.model\_selection.cross\_val\_predict.html (Links to an external site.)](https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.cross_val_predict.html)
* Notice that you will need to remove the lines that compute the mean & standard deviation over the accuracy.
* Use the ***confusion\_matrix()*** function to print out the confusion matrix of your classifier. This function should take the original class labels ***y*** and the predicted class labels ***y\_pred*** as input. Print the confusion matrix. Read more about the***confusion\_matrix()*** function here: [https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion\_matrix.html (Links to an external site.)](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html)

from sklearn.datasets import load\_svmlight\_file

from sklearn import svm

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

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

from sklearn.metrics import confusion\_matrix

print("Loading Dataset...")

X,y = load\_svmlight\_file("a1a.txt")

print("Creating classifier object...")

print("Creating classifier object...")

clf = svm.SVC(kernel='linear' ,C=1, random\_state= 42)

print("Training classifier with cross validation, k=5")

y\_pred = cross\_val\_predict(clf, X, y, cv=5)

print("Training Complete!")

confusion\_matrix(y, y\_pred)

Graphical user interface, text, application

Description automatically generated

* Modify the code to use an rbf kernel with gamma = 0.1. Print the obtained confusion matrix.

from sklearn.datasets import load\_svmlight\_file

from sklearn import svm

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

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

from sklearn.metrics import confusion\_matrix

print("Loading Dataset...")

X,y = load\_svmlight\_file("a1a.txt")

print("Creating classifier object...")

print("Creating classifier object...")

clf = svm.SVC(kernel='rbf', gamma=0.1 ,C=1, random\_state= 42)

print("Training classifier with cross validation, k=5")

y\_pred = cross\_val\_predict(clf, X, y, cv=5)

print("Training Complete!")

confusion\_matrix(y, y\_pred)

Graphical user interface, text, application

Description automatically generated

* Modify the code to use an rbf kernel with gamma = 0.01. Print the obtained confusion matrix.

from sklearn.datasets import load\_svmlight\_file

from sklearn import svm

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

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

from sklearn.metrics import confusion\_matrix

print("Loading Dataset...")

X,y = load\_svmlight\_file("a1a.txt")

print("Creating classifier object...")

print("Creating classifier object...")

clf = svm.SVC(kernel='rbf', gamma=0.01 ,C=1, random\_state= 42)

print("Training classifier with cross validation, k=5")

y\_pred = cross\_val\_predict(clf, X, y, cv=5)

print("Training Complete!")

confusion\_matrix(y, y\_pred)

Graphical user interface, text, application

Description automatically generated

from sklearn.datasets import load\_svmlight\_file

from sklearn import svm

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

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

from sklearn.metrics import confusion\_matrix

print("Loading Dataset...")

X,y = load\_svmlight\_file("a1a.txt")

print("Creating classifier object...")

print("Creating classifier object...")

clf = svm.SVC(kernel='poly', degree=2 ,C=1, random\_state= 42)

print("Training classifier with cross validation, k=5")

y\_pred = cross\_val\_predict(clf, X, y, cv=5)

print("Training Complete!")

confusion\_matrix(y, y\_pred)

Graphical user interface, text, application

Description automatically generated