7) Program to apply Logistic Regression on the iris dataset

from sklearn import datasets, decomposition

import pandas as pd

from sklearn.linear\_model import LogisticRegression

from sklearn.datasets import load\_digits

from sklearn.model\_selection import train\_test\_split

# Import the dataset

iris = datasets.load\_iris()

data = pd.DataFrame(data=iris.data, columns=iris.feature\_names)

X = data

y = iris.target

# Split the data into training and test sets

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

# Create an instance of LogisticRegression

# with the 'lbfgs' solver and L2 penalty

'''

Warning

The choice of the algorithm depends on the penalty chosen. Supported penalties by solver:

‘lbfgs’ - [‘l2’, None]

‘liblinear’ - [‘l1’, ‘l2’]

‘newton-cg’ - [‘l2’, None]

‘newton-cholesky’ - [‘l2’, None]

‘sag’ - [‘l2’, None]

‘saga’ - [‘elasticnet’, ‘l1’, ‘l2’, None]

'''

clf = LogisticRegression(solver='lbfgs', penalty='l2', max\_iter=10000)

# Fit the model to the training data

clf.fit(X\_train, y\_train)

# Evaluate the model on the test data

accuracy = clf.score(X\_test, y\_test)

print("Logistic regression Accuracy:", accuracy)