Cse23254 lab3

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

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

from sklearn.neighbors import KNeighborsClassifier

from sklearn.metrics import confusion\_matrix, classification\_report

data = pd.read\_csv("features\_raw (1).csv")

X = data.iloc[:, :-1].values

y = data.iloc[:, -1].values

# A1

classes = np.unique(y)

X\_class1 = X[y == classes[0]]

X\_class2 = X[y == classes[1]]

centroid1 = X\_class1.mean(axis=0)

centroid2 = X\_class2.mean(axis=0)

spread1 = X\_class1.std(axis=0)

spread2 = X\_class2.std(axis=0)

dist = np.linalg.norm(centroid1 - centroid2)

# A2

feature = X[:, 0]

plt.hist(feature, bins=20)

plt.show()

mean\_f = np.mean(feature)

var\_f = np.var(feature)

# A3

vec1, vec2 = X[0], X[1]

distances = []

for r in range(1, 11):

d = np.linalg.norm(vec1 - vec2, ord=r)

distances.append(d)

plt.plot(range(1, 11), distances)

plt.show()

# A4

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3)

# A5

knn3 = KNeighborsClassifier(n\_neighbors=3)

knn3.fit(X\_train, y\_train)

# A6

acc3 = knn3.score(X\_test, y\_test)

# A7

predictions = knn3.predict(X\_test)

# A8

accs = []

for k in range(1, 12):

knn = KNeighborsClassifier(n\_neighbors=k)

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

accs.append(knn.score(X\_test, y\_test))

plt.plot(range(1, 12), accs)

plt.show()

# A9

y\_pred = knn3.predict(X\_test)

cm = confusion\_matrix(y\_test, y\_pred)

report = classification\_report(y\_test, y\_pred, output\_dict=True)