pip install matplotlib

pip install scikit-learn

**Program:**

import numpy as np

import matplotlib.pyplot as plt

from mpl\_toolkits.mplot3d import Axes3D

from sklearn import datasets

def distance(instance1, instance2):

""" Calculates the Euclidean distance between two instances"""

return np.linalg.norm(np.subtract(instance1, instance2))

def get\_neighbors(training\_set, labels, test\_instance, k, distance\_fn):

"""

get\_neighbors calculates a list of the k nearest neighbors of an instance 'test\_instance'.

The function returns a list of k 3-tuples. Each 3-tuple consists of (index, dist, label)

"""

distances = []

for index in range(len(training\_set)):

dist = distance\_fn(test\_instance, training\_set[index])

distances.append((index, dist, labels[index]))

distances.sort(key=lambda x: x[1])

neighbors = distances[:k]

return neighbors

# Load the Iris dataset

iris = datasets.load\_iris()

data = iris.data

labels = iris.target

np.random.seed(42)

indices = np.random.permutation(len(data))

n\_training\_samples = 12

learn\_data = data[indices[:-n\_training\_samples]]

learn\_labels = labels[indices[:-n\_training\_samples]]

test\_data = data[indices[-n\_training\_samples:]]

test\_labels = labels[indices[-n\_training\_samples:]]

for i in range(5):

neighbors = get\_neighbors(learn\_data, learn\_labels, test\_data[i], 3, distance\_fn=distance)

print("Index:", i, '\n',

"Testset Data:", test\_data[i], '\n',

"Testset Label:", test\_labels[i], '\n',

"Neighbors:", neighbors, '\n')

# Visualize the data of the learn set

colours = ("r", "g", "y")

X = []

for iclass in range(3):

X.append([[], [], []])

for i in range(len(learn\_data)):

if learn\_labels[i] == iclass:

X[iclass][0].append(learn\_data[i][0])

X[iclass][1].append(learn\_data[i][1])

X[iclass][2].append(sum(learn\_data[i][2:]))

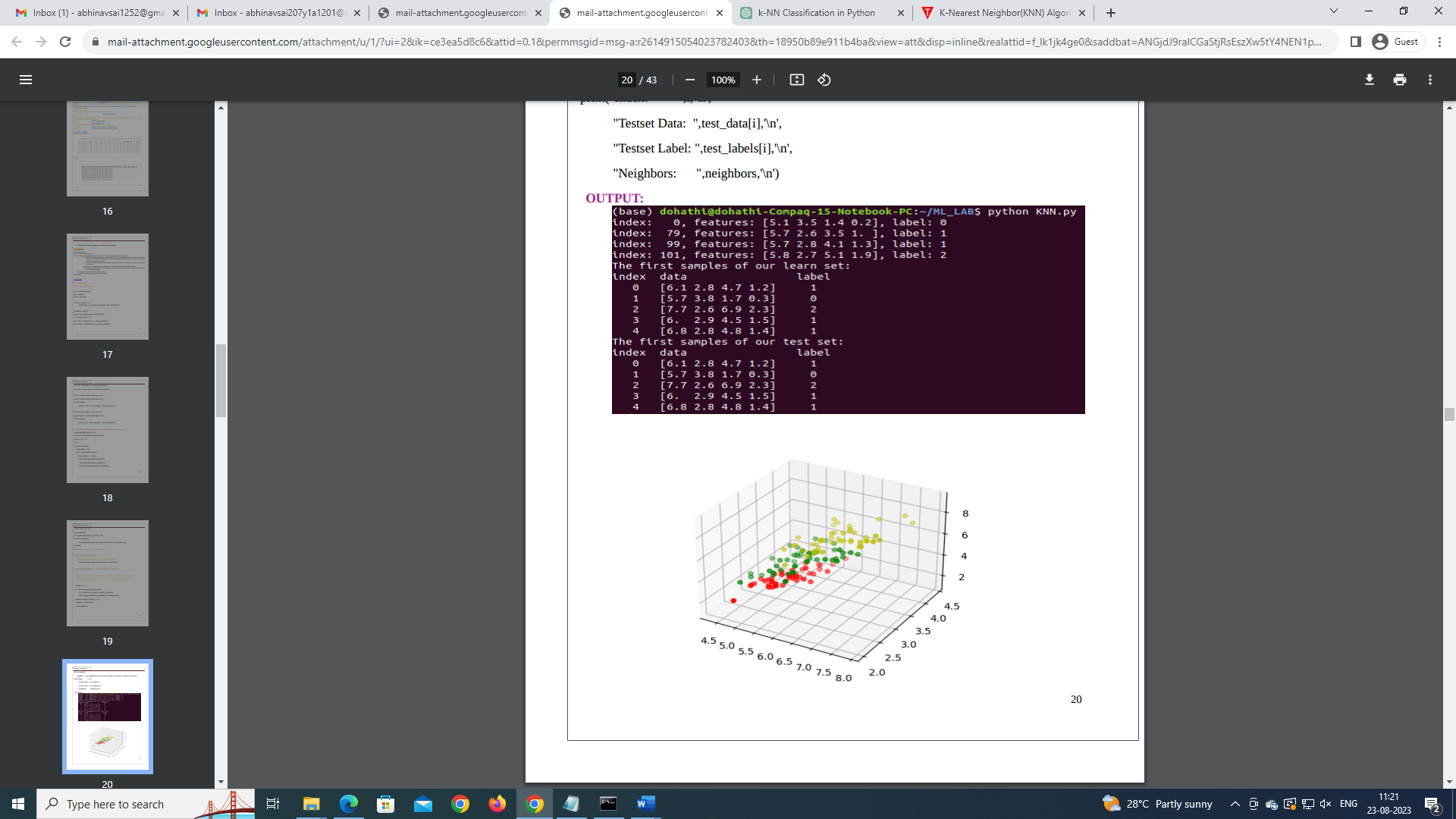
fig = plt.figure()

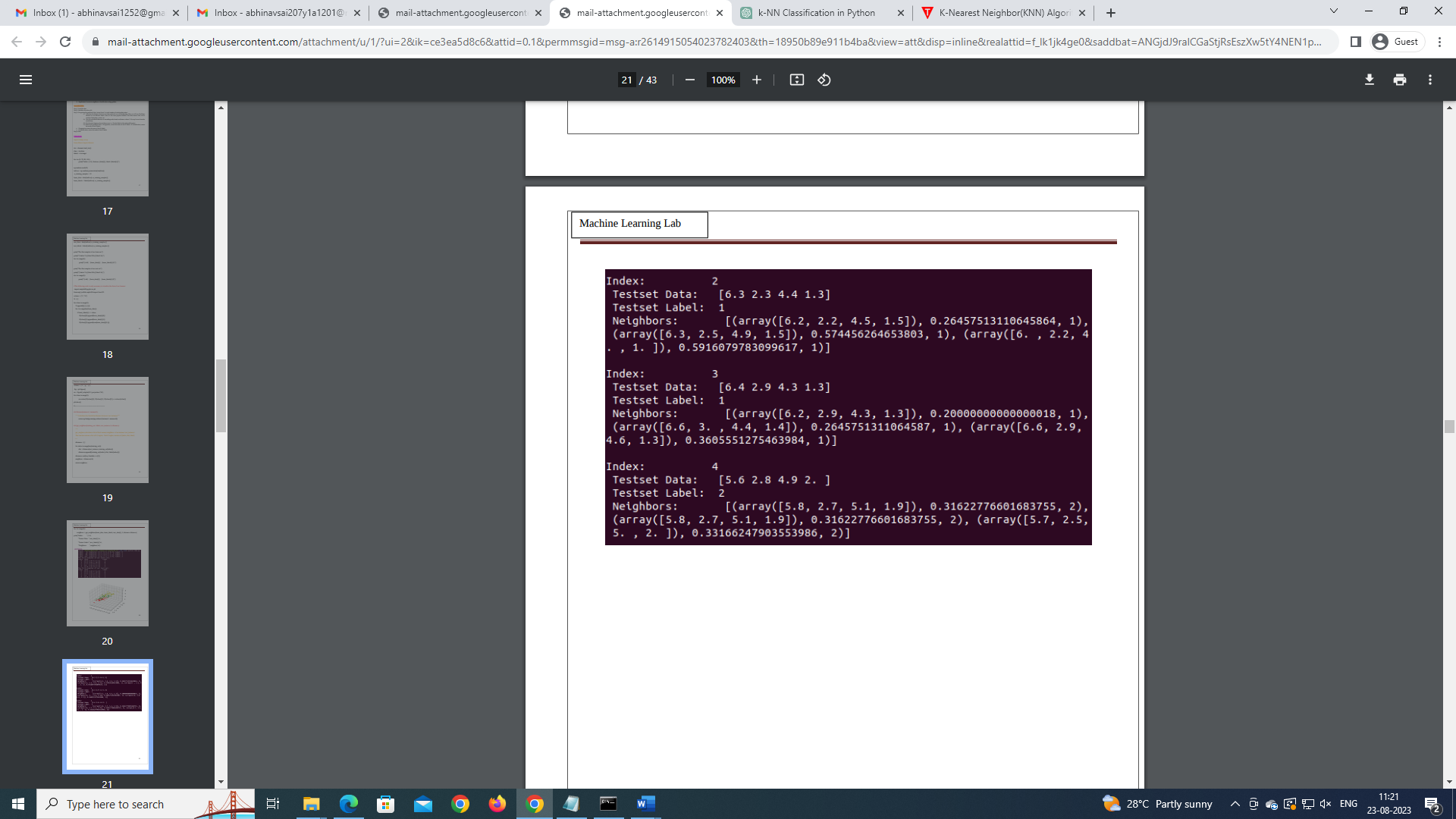
ax = fig.add\_subplot(111, projection='3d')

for iclass in range(3):

ax.scatter(X[iclass][0], X[iclass][1], X[iclass][2], c=colours[iclass])

plt.show()

****

****