#!/usr/bin/env python

# coding: utf-8

# In[1]:

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

# In[2]:

dfx=pd.read\_csv('xdata.csv')

dfy=pd.read\_csv('ydata.csv')

# In[3]:

X=dfx.values

Y=dfy.values

X=X[:,1:]

Y=Y[:,1:].reshape((-1,))

print(X)

print(X.shape)

print(Y.shape)

# In[4]:

plt.scatter(X[:,0],X[:,1],c=Y)

plt.show()

# In[5]:

query\_x=np.array([2,3])

plt.scatter(X[:,0],X[:,1],c=Y)

plt.scatter(query\_x[0],query\_x[1],color='red')

plt.show()

# In[6]:

def dist(x1,x2):

return np.sqrt(sum((x1-x2)\*\*2))

def knn(X,Y,queryPoint,k=5):

vals=[]

m=X.shape[0]

for i in range(m):

d=dist(queryPoint,X[i])

vals.append((d,Y[i]))

vals=sorted(vals)

vals=vals[:k]

vals=np.array(vals)

# print(vals)

new\_vals=np.unique(vals[:,1],return\_counts=True)

print(new\_vals)

index=new\_vals[1].argmax()

pred=new\_vals[0][index]

return pred

# In[7]:

knn(X,Y,query\_x)

# #MNIST DataSets

# In[8]:

df=pd.read\_csv('train.csv')

print(df.shape)

# In[9]:

print(df.columns)

# In[10]:

df.head()

# In[11]:

#Create Numpy Array

data=df.values

print(data.shape)

print(type(data))

# In[12]:

X=data[:,1:]

Y=data[:,0]

print(X.shape,Y.shape)

# In[13]:

split=int(0.8\*X.shape[0])

print(split)

# In[14]:

X\_train=X[:split,:]

Y\_train=Y[:split]

X\_test=X[split:,:]

Y\_test=Y[split:]

print(X\_train.shape,Y\_train.shape)

print(X\_test.shape,Y\_test.shape)

# In[15]:

def drawImg(sample):

img=sample.reshape((28,28))

plt.imshow(img,cmap='gray')

plt.show()

# #Hand Written Digit Prediction System# In[21]:

predict=knn(X\_train,Y\_train,X\_test[70])

print("The predicted digit is=",predict)

# In[22]:

drawImg(X\_test[70])

print("The digit is",Y\_test[70])

Output:

