from lib import \*

#from implementation import \*

from collections import defaultdict

chartoascii={0:48,

1 :49,

2 :50,

3 :51,

4 :52,

5 :53,

6 :54,

7 :55,

8 :56,

9 :57,

10 :65,

11 :66,

12 :67,

13 :68,

14 :69,

15 :70,

16 :71,

17 :72,

18 :73,

19 :74,

20 :75,

21 :76,

22 :77,

23 :78,

24 :79,

25 :80,

26 :81,

27 :82,

28 :83,

29 :84,

30 :85,

31 :86,

32 :87,

33 :88,

34 :89,

35 :90,

36 :97,

37 :98,

38 :100,

39 :101,

40 :102,

41 :103,

42 :104,

43 :110,

44 :113,

45 :114,

46 :116}

def data\_distribution():

plt.style.use("fivethirtyeight")

digits\_counter=Counter()

digits\_counter.update(label)

d=[]

for x in digits\_counter.items():

d.append(list(x))

digits\_counter=d

digits\_counter.sort(key=lambda x:x[0])

digits=[]

counts=[]

for i,j in digits\_counter:

digits.append(i)

counts.append(j)

plt.bar(digits,counts)

plt.title("Distribution of Digits")

plt.ylabel("total counts")

plt.tight\_layout()

plt.show()

#function create\_img is use to sample image in our training data

def create\_img():

d=defaultdict()

l=data.iloc[:,1:]

op=data.iloc[:,0]

print(op)

for i in range(2000):

if op[i] not in d:

d[op[i]]=1

gray\_scale\_array=list(l.iloc[i])

mode='L'

size=28,28

image\_out = PIL.Image.new(mode,size)

image\_out.putdata(gray\_scale\_array)

image\_out.save('sample/test\_out{}.png'.format(chr(chartoascii[op[i]])))

#create\_img()

#img\_show function display any image passed as array

def img\_show(a):

mode='L'

size=28,28

i=PIL.Image.new(mode,size)

i.putdata(a)

i.show()

i.save('demo\_digit/pca\_{}.png'.format("imp\_image"))

#variance

def variance\_pca\_plot():

\_sum=0

var=list(variance)

for i in range(len(var)):

\_sum+=var[i]

var[i]=\_sum

x\_label=[ i for i in range(1,len(var)+1)]

y\_label=var

print(len(x\_label))

print(len(y\_label))

plt.bar(x\_label,y\_label)

plt.title("The cumulative variance of Principal components")

plt.ylabel("Percentage variance")

plt.show()

#variance\_pca\_plot()

def img\_to\_array(path):

im=PIL.Image.open(path)

print(im.size)

im.thumbnail((28,28))

print(im.size)

x=np.array(im)

ans=[]

for i in range(len(x)):

for j in range(len(x[0])):

ans.append((0.3 \*x[i][j][0] ) + (0.59 \* x[i][j][1]) + (0.11 \* x[i][j][2]))

#print(ans)

#print(len(ans))

#img\_show(ans)

return np.array([np.array(ans)])

#x=img\_to\_array("C:/Users/ashish agarwal/Desktop/pavan imp videos/MPR Project/drawn.png")

#print(x)