

Pattern Recognition –HW#2

About the Assignment

The main aim of the assignment is to make you familiar with a traditional classifier by using KNN. Contributions of this lab are;

- Learning the K-NN classification method.
- Understanding the idea behind the classification task.

Step1:

Download the Cifar-10 dataset python version by using the following commands

```
from keras.datasets import cifar10
(x_train, y_train), (x_test, y_test) = cifar10.load_data()
```

Step3:

Convert images to vector format by writing this snipped code. It means that an image with 32x32x3 channels converted as 1x3072 vector. There 50,000 train vectors and 10,000 test vectors.

```
x_train = x_train.reshape(-1, 3072)
x_test = x_test.reshape(-1, 3072)
```

Step4:

Write a function send the parameters of ***x_train, y_train, sample_test, k*** as input, then return the most similar class name for sample_test. In case of computing the similarity, you are expected to use the **cosine similarity distance**, which is explained in lecture notes by Teacher in class. You can use any code, but the following function can help you.

```
import math
def cosine_similarity(v1,v2):
    "compute cosine similarity of v1 to v2: (v1 dot v2)/(|v1|*|v2|)"
    sumxx, sumxy, sumyy = 0, 0, 0
    for i in range(len(v1)):
        x = v1[i]; y = v2[i]
        sumxx += x*x
        sumyy += y*y
        sumxy += x*y
    return sumxy/math.sqrt(sumxx*sumyy)

v1,v2 = [3, 45, 7, 2], [2, 54, 13, 15]
print(v1, v2, cosine_similarity(v1,v2))
```

sample_test refers to a test vector from x_test. You can set the sample_test like this way.

```
sample_test = x_test[1,:];
```

The test code look like this.

```
sample_test = x_test[0,:];  
k=5  
similar_class_name = knnClassifier(x_train, y_train, sample_test, k )  
print(similar_class_name)
```

You have to implement code on my own template, namely HW2.py

Submit the Assignment

Ex: No_Name_Surname_HW#.zip

Hint

No hint

Note in English:

You will graded over 5 points if you send your friend's code or any snipped code available on internet. You have to consider following 3 three rules in case of uploading the hw.

- 1) Send a screen shot that shows your code worked successfully without error.
- 2) Send a screen shot related to generated results in spyder
- 3) Only send **py extension file**, other file format will be graded over 5 points You have to use **spyder** when implementing any homework.

Screen shots must be related to spyder.

If one of the rule above is ignored, then your homework will be graded only over 50 points.

Take care yourself.

Note in Turkish:

Vereceğim ödevlerde internetteki **hazır kodu** benimle paylaşmanız veya kendi **arkadaşınızın kodunu** paylaşmanız. Direk bakılmadan **5 puan** olarak değerlendirilecektir.

Ödevleri yüklerken aşağıdaki 3 şeyi dikkate alarak yükleme yapınız.

- 1) Ödevin **hatasız** çalıştığına dair ekran çıktısı
- 2) Ödevin ürettiği **sonuçlara** ait ekran çıktısı
- 3) Ödevi yüklerken sadece **py uzantılı** dosyayı yükleyiniz. Bunun dışındaki formatta gönderenler sadece **5 puan** alır. Ödevleri **spyderda** yazmanız gerekmektedir.

Ekran çıktısı spydera ait olmalı.

Bu üç maddeden biri eksik olanların ödevleri 50 üzerinden değerlendirmeye alınacaktır