CENG 484 - Data Mining
Assignment 2

In this assignment, you will perform the following **two main tasks**.

1. (48 p) Try to implement **image based search** without any text or tags. You will find related images from a given search space (images). A similarity measurement is required to find related images. The **cosine similarity** will be used to compare the searched image. The **three images** most similar to the searched image will be displayed. You must calculate similarity by **writing own function**, you can use mathematical computing libraries such as numpy. This program will take name of the image as input and will give the three most similar images as shown below.

   Input
(Image)
Search Similar Images
Output
(Related 3 Images)

   The **cosine similarity** will be calculated according to the formula below. You can review from **Chapter 2 Part-3 slides** and from the main textbook.

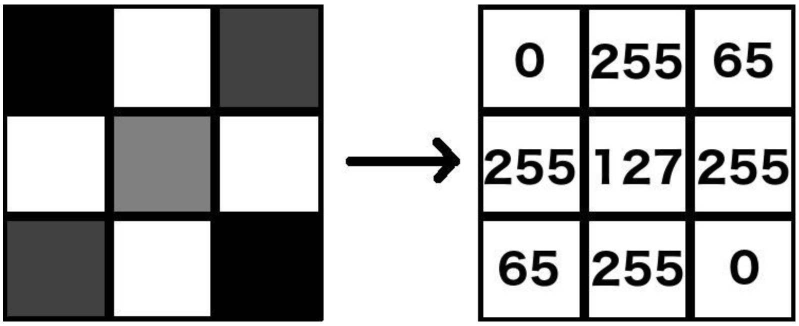
$$\cos\theta = \frac{\vec{a}\cdot\vec{b}}{\|\vec{a}\|\|\vec{b}\|}$$

   In this formula, a and b values are treated as **vector**. The top of the division is **dot product** and the bottom of the division is **Euclidean operation (L2 Norm)**. These operations can be done with the help of a calculation library like a numpy.

$$\|\vec{a}\| = \sqrt{a_1^2 + a_2^2 + a_3^2 + \cdots + a_n^2}$$

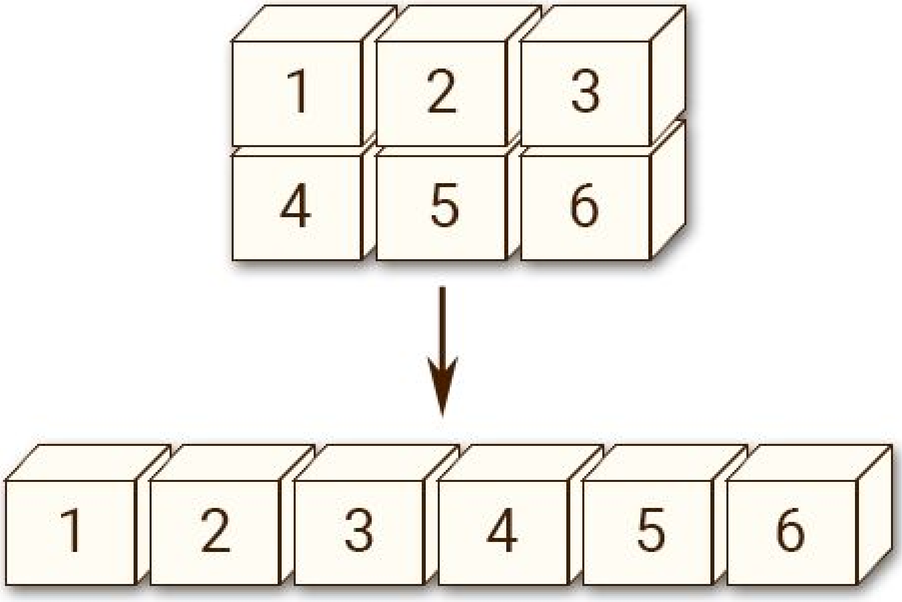
$$\|\vec{b}\| = \sqrt{b_1^2 + b_2^2 + b_3^2 + \cdots + b_n^2}$$

A digital image is presented in the computer by **pixels matrix**. Each pixel of an image is presented by one matrix element as a numeric value. The numeric values in pixel presentation are uniformly changed from 0 (black) to 255 (white). You have to use **32 bit float numbers** (In Python we can convert using astype('float32')) to represent pixels.



0	255	65
255	127	255
65	255	0

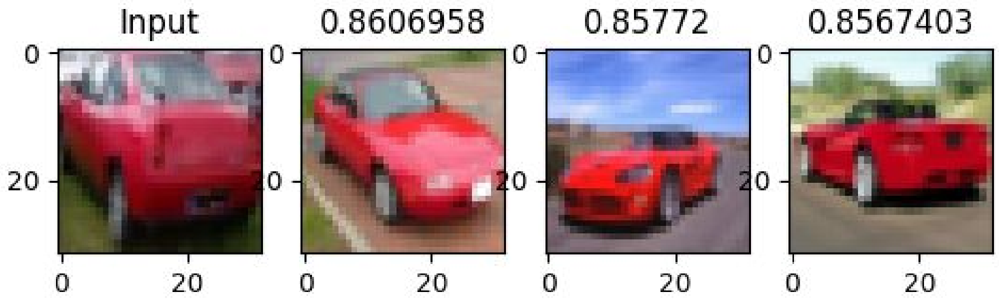
After obtained the image in the form of a matrix, you should convert 32 bit float and turn it into a **vector** for calculating similarity. Converting the image **matrix into a vector** is shown below.



After converting searched image into the vector, you will calculate **similarity** and found **related three image** according to similarity value. You can choose Python or R for coding.

Example program execution for **Input:** 3952.png.

Output: Most similar three images with similarity values: 4946.png, 4124.png and 3861.png as shown below.



Your program will be tested with the following steps, **add your own outputs** to the report for each input.

a) What will be the output for this input? **Input:** 4228.png

b) What will be the output for this input? **Input:** 3861.png

2. (52 p) While creating the decision tree, there are many measures that can be used to determine the best way to split the records. The measures developed for selecting the best split are often based on the **degree of impurity** of the child nodes, some of them are as follows. You can review from **Chapter 4 Part-2 slides**.

$$\begin{aligned}\text{Entropy}(t) &= -\sum_{i=0}^{c-1} p(i|t)\log_2 p(i|t), \\ \text{Gini}(t) &= 1-\sum_{i=0}^{c-1}[p(i|t)]^2,\end{aligned}$$

where c is the number of classes and $0\log_2 0 = 0$ in entropy calculations.

Try to implement **Entropy (E)** and **Gini Index (GI)** calculations by **writing own function** using Python or R. You will make these operations on heart disease data (heart_summary.csv).

a) Compute the **E and GI** for the overall collection of training examples. (8 p)

b) Compute the **E and GI** for the **age** attribute. (8 p)

c) Compute the **E and GI** for the **cp** attribute. (8 p)

d) Compute the **E and GI** for the **trestbps** attribute. (8 p)

e) Which attribute is better according to calculations? (10 p)

f) Which attribute can be chosen as the root ? Explain why. (10 p)

Note: Please upload **only** Python or R **codes** (task 1 and task 2) and your **report** (with answers to the questions) to CMS.

Please submit your solutions until **20 May 2020** 23:00. You should upload a zip file "Student_Number_Name.zip" as shown below.

```
Student_Number_Name.zip
              |
              |
              |_ _ _ _ _ _ _  Task_1.py or Task_1.R
              |
              |
              |_ _ _ _ _ _ _  Task_2.py or Task_2.R
              |
              |
              |_ _ _ _ _ _ _  Report.pdf
```