Introduction to Machine Learning and Data Mining

Homework #1: Clustering methods. Part 1

Soft Deadline: 19.04.2020 23:59
Hard Deadline: 26.04.2020 23:59

Submission format

Submit your homework as a report as IPython notebook or PDF file:

- Describe each step of your work in detail;
- Provide necessary graphs and images;
- If you created a program, which is different from IPython notebook code, to do the work, put the code into your report, specify the programming language and add comments to key lines;
- Justify your decisions; Provide the reader with meaningful conclusions.

The report should be send as IPython notebook, PDF or DOC to <dmitrii.ignatov@gmail.com> with the email's topic [MLDM2020-HW1-Clust-P1]-<Name Family Name> with a CC to TA, Dmitry Egurnov <egurnovdima@gmail.com>.

Works submitted after the soft deadline will be penalised by one point, work submitted after the hard deadline will not be accepted without serious excuse.

Task 1 (10 points). Image clustering

You need to cluster one of the attached images (or choose your own) using the color description of each pixel. You can add pixel coordinates as features. The clustering procedure outputs:

1. Cluster centroids (as pixel colors);
2. The compressed image where pixel colors are replaced with corresponding cluster centroid's color;
3. Compare the amount of memory needed to store the original and compressed image.

Choose the number of clusters ad hoc between 4 and 16. Here is a small tutorial about obtaining color description of a pixel.

Listing 1: *Image processing tutorial*

```
# Import libraries
from imageio import imread
import matplotlib.pyplot as plt
%matplotlib inline
import numpy as np
```

```
# Read and show image
img1 = imread('data/1.jpg')
img1.shape # You can see, that images are stored as 3-dimensional arrays, where
    3rd dimension contains RGB description of a pixel.
fig = plt.figure()
plt.imshow(img1)

# Prepare data
ind = np.indices(img1.shape[:2]).transpose(1,2,0)
features = np.dstack((ind, img1)).reshape(-1, 5)
```

Task 2 (10 points). Determine the optimal number of clusters

Remember the K-Means method optimises the following criteria:

$$J(R) = \sum_{i=1}^{k} \sum_{j \in R'_i} d(c_i, y_j),$$

where R'_i is the i-th cluster, c_i is the centroid of the i-th cluster, y_j is a data-point in i-th cluster, and $d(\cdot, \cdot)$ is the distance between objects

The "Elbow method" is one of the approaches for making decisions on the number of clusters for the K-Means method. The idea is to calculate the target function $J(R)$ for every natural number k from a certain range. The optimal number of clusters is defined as the value of k, starting with which the value of $J(R)$ does not decrease so sharply. For example, the number of classes defined by figure 1 can be chosen equal to 4.

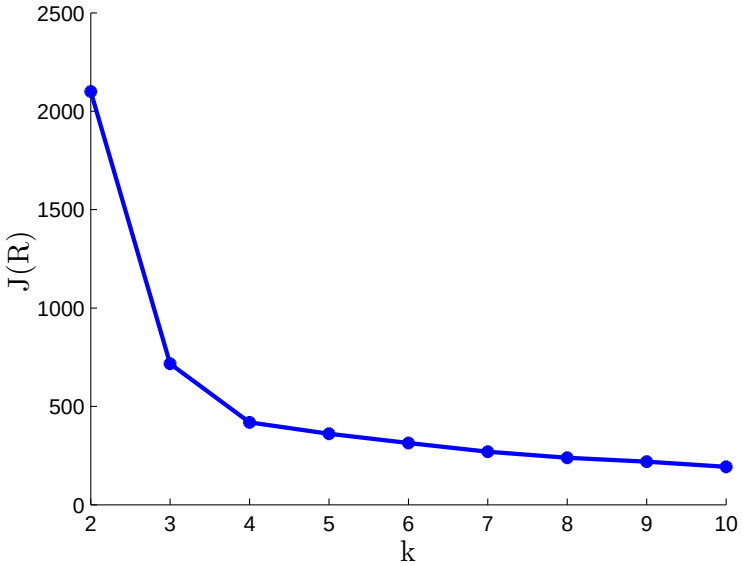

Figure 1: *Elbow method*

Using the data from `elbow.txt` and the elbow method, define the optimal number of clusters.