8CC00

The individual assignment of the clustering and classification case

Peter Hilbers

February 2021

In this assignment several Python programs have to be designed to cluster and classify data points.

The data that is to be used is the same as for your individual PCA assignment, so it consists of 145 data points, where each data point is a 244-dimensional vector of gene expression values. Each data point has as label the tumor type. Apart from using pandas to read the data and `random` for generating random numbers, no other methods from system modules, such as numpy, scipy, and sklearn, are allowed in your solutions. In this case the task is to program it yourself. As stated in the individual assignment of the PCA case your report should include documentation, including and with emphasis on specifications, the model, complexity analysis, and last but not least the interpretation of the results.

Several subtasks have to be performed:
Clustering:

1. Implement the k-means algorithm where k and the distance method should be parameters of your method.
2. Use as distance the squared Euclidean distance and run your code for different values of k and apply a silhouette analysis. Note that the results of the k-means algorithm strongly depend on the initial assignment, so it is advised to run the algorithm a several times.
3. Choose for k the value with the highest silhouette score. Discuss the k-means clusters that are generated and give possible interpretations.
4. In order to generate a graph in which edges connect nodes having a sufficiently high absolute value of their correlation coefficient, we need a threshold value. If we have n nodes, there are $n(n-1)/2$ pairs of nodes. For $0 \leq c \leq 1$ let $f(c)$ denote the fraction of those $n(n-1)/2$ pairs of nodes that have an absolute value of their correlation coefficient at least c. So if $c = 0$ all pairs satsify this criterion and hence $f(0) = 1$ while for $c = 1$ only the pairs of nodes with a perfect correlation would rest. Construct a plot with on the x-axis a value c ranging from 0 to 1, and $f(c)$ as y-value.
5. Implement the Highly Connected Subgraph(HCS) algorithm including the KargerCut method.
6. Choose a value c such that $f(c)$ is approximately 0.1, and construct the corresponding graph with approximately $0.1 * n(n-1)/2$ edges. Apply the HCS algorithm on this graph.
7. Discuss the differences of the results of the HCS algorithm with those of the k-means algorithm with for k the value with the highest silhouette score.

Classification:

Leave-one-out cross validation is the single method to be dealt with. In the method all but one of the data points are used as training set, while the algorithm has to predict the label of the point that is left out. Each of the 145 data points should serve once as the data point that is left out.

1. Implement the k-nearest neighbour algorithm where k and the distance method should be parameters of your method.
2. Use again the squared Euclidean as distance method. Determine for at least 5 different values of k what the error score of your k-nearest neighbour algorithm is, i.e., the number of times that the algorithm generates a different label than the real label when run for all 145 data points with the leave-one-out procedure.
3. Choose the same value as in the clustering subtask for c such that a graph exist with approximately $0.1 * n(n-1)/2$ edges. When the HCS algorithm is run then determine for each of the generated highly connected subgraphs its label, defined as the majority vote of the vertices of the subgraph. (When there is a draw multiple labels may be chosen.) Similar as in the k-nearest neighbour case determine the error score.
4. Discuss the differences between the results of both algorithms.

Success and enjoy!!!

Deadline for submission of your report: Monday April 5, 2021, 23.59h.