PROBLEM SET 3

SUBMIT BY 17 NOVEMBER 2023

Instructions

Please submit on Canvas as a Jupyter Notebook file (an `ipynb` file) by 23:59 on **17.11.2023**.

Three problem sets count for 40% of the module assessment, and the exam counts for the other 60%. The lowest of the three sets will be ignored.

Develop your own Python code rather than simply using existing Python modules for computing the relevant mathematics we discuss in class (e.g. `statsmodels`). If in doubt, send me an email. The Jupyter notebook should contain your responses and/or proofs as markdown cells.

The homework will be graded according to a scheme in which *content* (i.e. correctness of your answers, choice of methods, python code) is weighted at 80% and *presentation* (i.e. manner in which you present your answers, methods, and code) is weighted at 20%.

Problems

Problem 1. Write your own Python function, called `BasicKMeans`, to compute a k-means clustering. The input and output of `BasicKMeans` should satisfy the following:

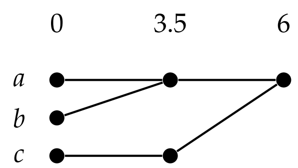
Input: a positive integer `k`, and a list (or other list-like iterable) of `n` data points in $\mathbb{R}^m$.
Output: a list `L` of length `n` such that `L[i] = j` implies the ith point is contained in the jth cluster.

Problem 2. The following table gives distances between vertices of a graph.

	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
v_1	0	9	6	1	6	7	8	2
v_2	9	0	3	8	8	4	6	10
v_3	6	3	0	6	6	2	4	8
v_4	1	8	6	0	5	7	7	2
v_5	6	8	6	5	0	7	3	6
v_6	7	4	2	7	7	0	5	9
v_7	8	6	4	7	3	5	0	8
v_8	2	10	8	2	6	9	8	0

Suppose $t \geqslant 0$, and let G_t be the graph with vertices $\{v_1, \ldots, v_8\}$ and edges $\{\{v_i, v_j\} \mid \mathrm{d}(v_i, v_j) \leqslant t\}$.

(1) Give the adjacency matrices for the three graphs: G_2, G_5, and G_8.
(2) Give two dendrograms that encodes the inclusion between the connected components of the graphs $G_0, G_1, \ldots, G_{10}$. One dendrogram should use single-linkage and the other dendrogram should use complete-linkage.
Note: Instead of drawing a dendrogram, we will represent the same information in Python as a list `D` of pairs `(t, C)`, where `t` is a real number and `C` is a list of sets, each set in `C` corresponds to a cluster. For example the dendrogram



can be represented as

```
D = [
  (0, [{‘a’}, {‘b’}, {‘c’}]),
  (3.5, [{‘a’, ‘b’}, {‘c’}]),
  (6, [{‘a’, ‘b’, ‘c’}])
]
```

Date: October 31, 2023.