

C S 519 Applied Machine Learning

Compare clustering methods

1. Objective

In this **individual** homework, you are required to understand and compare several clustering algorithms.

2. Requirements

2.1 Tasks

(1) [40 points] Write code to conduct clustering by

- (a) [15 points] using the K-means algorithm offered by the scikit-learn library,
- (b) [15 points] using a hierarchical approach offered by the SciPy library, and
- (c) [10 points] using a hierarchical approach offered by the scikit-learn library.

(2) [10 points] Use the elbow approach to decide a reasonable K for the K-means algorithm. Put the related figures and the analysis of deciding K in the report.

(3) [25 points] Each cluster algorithm needs to be tested and analyzed using two datasets:

- (a) The Iris dataset (iris.data) with description (iris.names.txt), which can be downloaded from the [iris data](#) folder in Canvas., and
- (b) The MNIST dataset, which can be loaded from sklearn.datasets using the function fetch_mldata (scikit-learn version before 0.19) or fetch_openml function (scikit-learn version from 0.20).

Note: The MNIST dataset has 70K instances. It might be too large for your computer (or our CS server) memory to do all the calculations. To work on your homework without having this memory problem, you can extract a subset of the dataset by using the train-test-split method with the proper stratify option to get an MNIST_subset. Then, you can use this MNIST_subset as the MNIST and later split MNIST_subset to X_train and X_test for the rest of your calculations. If the subset has 1000-2000 rows, the model should work without the memory issue.

You need to think how to utilize these datasets to conduct clustering because these datasets are generally used for classification.

(4) [20 points] Properly analyze the clustering algorithms' behavior by applying the knowledge that we discussed in class. Such analysis can be about the performance variation when changing the value of K (for K-means), or the linkage (for hierarchical method). Such analysis should include running time and Sum Squared Error (SSE). You can also use class labels as ground truth to examine the clustered results. Put your analysis of both datasets to a report file.

(5) [5 points] Write a readme file **readme.txt** with detailed instructions to run your program.

2.2 Other requirements

- Your Python code should be written for Python version 3.10 or higher.
- Please write proper comments in your code to help the instructor and teaching assistants to understand it.
- Please properly organize your Python code (e.g., create proper classes, modules). You can put your code to Jupyter Notebook or a .py file.

3. Submission instructions

Put all your files (Python code, readme file, report, datasets, etc.) to a zip file named **hw6_<YourName>.zip** and upload it to Canvas.

4. Grading criteria

- **ZERO point** will be given if your code does not work. Please do not submit code that you did not test and make sure it works.
- The score allocation has been put beside the questions.
- **FIVE** points will be deducted if files are not submitted in the required format.
- If the total points are more than 100. Your grades will be scaled to the range of [0,100].
- Please make sure that you test your code thoroughly by considering all possible test cases.