Assignment - Kmeans Clustering

Software Project (0368-2161) Due: 18.07.2024

1 Introduction

The K-means algorithm is a popular clustering method for finding a partition of N unlabeled observations into K distinct clusters, where K is a parameter of the method. In this assignment, you will implement this algorithm in both Python and C.

1.1 K-means

Given a set of N datapoints $x_1, x_2, \ldots, x_N \in R^d$, the goal is to group the data into $K \in \mathbb{N}$ clusters, each datapoint is assigned to exactly one cluster and the number of clusters K is such that $1 < K < N$. Each cluster k is represented by its centroid, which is the mean $\mu_k \in R^d$ of the cluster's members.

Algorithm 1 k-means clustering algorithm

1: Initialize centroids as first k datapoints: $\mu_k = x_k,\ \forall k \in K$
2: **repeat**
3: Assign every x_i to the closest cluster k: $\underset{k}{\operatorname{argmin}}\ d(x_i, \mu_k),\ \forall k\ 1 \leq k \leq K$
4: Update the centroids: $\mu_k = \frac{1}{|k|}\sum_{x_i \in k} x_i$
5: **until** convergence: $(\Delta\mu_k < \epsilon)\ OR\ (iteration\_number = iter)$

Where:

- $d(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 \cdots (p_d - q_d)^2}$: More Info: Euclidean Distance.
- $\Delta\mu_k$: Eclidean Distance, between the updated centroid to the previous one. (this should be checked for every centroid).

2 Assignment Description

Implement the k-means algorithm as detailed in Algorithm 1 both in C and Python.

2.1 Requirements

1.

Input Variable	Valid Values	Error Message
K	$1 < K < N,\ K \in \mathbb{N}$	"Invalid number of clusters!"
iter	$1 < iter < 1000,\ iter \in \mathbb{N}$	"Invalid maximum iteration!"
input_data	.txt file [1]	NA

[1] Only applied to the Python program. You can assume that it's valid, and is provided.

2. Apply K-means on the input, and print the final centroids.
3. Use $\epsilon = 0.001$
4. If `iter` is not provided, default value is 200.

2.2 Compile and Running

2.2.1 C

1. The program must compile cleanly on the Nova server (no errors, no warnings) when running the following command:

```
$gcc -ansi -Wall -Wextra -Werror -pedantic-errors kmeans.c -o kmeans
```

2. After successfully running the above command, an executable file called `kmeans` will be created. Now you can run your program by executing the following line on Nova:

```
# K = 3, iter = 100
$./kmeans 3 100 < input_data.txt
-5.6720,7.7200,-9.1478,-9.1870,2.1394
-5.3645,8.0872,5.4206,9.6482,-6.5119
-6.8116,-2.9751,3.7578,-0.8476,-10.5475
```

3. Reading the input from `STDIN`, you can use any of: `getline(), getchar(), scanf()`.
4. Output the calculated centroids to the screen `STDOUT`, you can use any of: `putchar(), printf()`.
5. Please note, if you use the ***math.h*** library, you have to add the *-lm* flag to the compilation command.
6. You can use: *stdlib.h, stdio.h, math.h*

2.2.2 Python

1. Your program must be executed by (no errors, no warnings) the following line on Nova:

```
# K = 3, iter = 100
$python3 kmeans.py 3 100 input_data.txt
```

```
-5.6720,7.7200,-9.1478,-9.1870,2.1394
-5.3645,8.0872,5.4206,9.6482,-6.5119
-6.8116,-2.9751,3.7578,-0.8476,-10.5475
```

2. Please note, in Python, read the data **directly** from the file.
3. Consider using the following: `split()` in your implementation.
4. You can use any builtin package that comes with the interpreter. (ones that doesn't need installation).
5. Print the output to the screen. (same as in C)

2.3 Assumptions

Note that the following list applies to both programs in this assignment:

1. Validate that the command line arguments meet the requirements see Table. 1.
2. Outputs must be formatted to 4 decimal places (use: '%.4f') in both languages, for example:
   - $8.88885 \Rightarrow 8.8888$
   - $5.92237098749999997906 \Rightarrow 5.9224$
   - $2.231 \Rightarrow 2.2310$
3. 3 input files and their corresponding output files examples are provided within the assignment in Moodle. (**YES**, the input files have an extra empty row and this is the expected behaviour)
4. Handle errors as following:
   (a) In case of invalid input, print the relevant message as detailed in Table. 1 and terminate the program.
   (b) Else, print "An Error Has Occurred" and terminate.
5. Do not forget to free any memory you allocated.
6. For successful running, the C program must return 0 otherwise 1.
7. You can assume that all given data points are different.
8. You may not import external includes (in C) or modules (in Python) that are not mentioned in this document.
9. Use `double` in C and `float` in Python for all vector's elements.

3 Submission

1. Please submit a file named `id1_id2_assignment1.zip` via Moodle, where id1 and id2 are the ids of the partners (replace id1, id2 with your ids).

   (a) In case of individual submission, id2 must be 111111111

   (b) Put the following files **ONLY** in a folder called `id1_id2_assignment1`:

      i. `kmeans.c`

      ii. `kmeans.py`

   (c) Zip the folder using the following Linux cmd:

```
$zip -r id1_id2_assignment1.zip id1_id2_assignment1
```

   Do not use other ways to create the zip!