Asmt 4: Clustering

Turn in through **Gradescope** by Wednesday, February 22 at 2:45pm, then come to class:
100 points

Overview

In this assignment, you will explore clustering: hierarchical and point-assignment. You will also experiment with high-dimensional data. **You must provide code only where we explicitly ask for it.**

You will use three data sets for this assignment: `C1`, `C2`, and `C3` available on Canvas. These data sets all have the following format. Each line is a data point. The lines have either 3 or 6 tab-separated items. The first one is an integer describing the index of the points. The next 2 (or 5 for `C3`) are the coordinates of the data point. `C1` and `C2` are in 2 dimensions, and `C3` is in 5 dimensions. `C1` should have n=21 points, `C2` should have n=1029 points, and `C3` should have n=1000 points. We will always measure distance with Euclidean distance.

Note: Homework assignments are intended to help you learn the course material, and successfully solve mid-term and final exams that will be done on paper in person.

As usual, it is recommended that you use LaTeX for this assignment. If you do not, you may lose points if your assignment is difficult to read or hard to follow. The latex source for this homework is available on Canvas. I recommend that you drop the two latex files (.tex and .sty) in a new Overleaf project, and compile/edit the .tex file there.

Submissions that are not uploaded on Gradescope will get 10% penalty.

1 Hierarchical Clustering (35 points)

There are many variants of hierarchical clustering; here we explore only 2.
`Single-Link`: measures the shortest link $d(S_1, S_2) = \min_{(s_1,s_2)\in S_1\times S_2} \|s_1 - s_2\|_2$.
`Complete-Link`: measures the longest link $d(S_1, S_2) = \max_{(s_1,s_2)\in S_1\times S_2} \|s_1 - s_2\|_2$.

A (30 points): Run the two hierarchical clustering variants on dataset `C1.txt` until there are $k = 3$ clusters, and report the results as sets. For example, you could report a table with columns for each cluster and two rows for two linkage criteria or a figure made by using scatter in `matplotlib`.

Answer:

B (5 points): Implement 1.A using `AgglomerativeClustering` in `scikit-learn`. Documentation is available here (click on the word "here"). Report your code.

Answer:

```python
from sklearn.cluster import AgglomerativeClustering

# your code goes here
```

C (1 *extra* point): You can show the output of `AgglomerativeClustering` as a dendrogram (a hierarchy of clusters) using `scipy` and `matplotlib` libraries. Report a dendrogram constructed with these libraries for the clusters obtained on dataset `C1.txt`.

Answer:

2 Assignment-Based Clustering (65 points)

Assignment-based clustering works by assigning every point $x \in X$ to the closest centeroid's cluster C. Let $\phi_C : X \to C$, $\phi_C(x) = \arg\min_{c \in C}(x, c)$, be this assignment map All points that map to the same cluster are in the same cluster.

Two good initialization methods for this type of clustering are the `Gonzalez` (Algorithm 8.2.1 in M4D book) and k-`means++` (Algorithm 8.3.2) algorithms.

A: (15 points) Run `Gonzalez` and `k-means++` on data set `C2.txt` for $k = 4$. To avoid too much variation in the results, choose c_1 as the point with index `1`.

(i) For `Gonzalez`, report the centroids and clusters (make a figure using scatter in `matplotlib`).

(ii) 4-center cost $\max_{x \in X}(x, \phi_C(x))$ and

(iii) 4-means cost $\sqrt{\frac{1}{|X|}\sum_{x \in X}((x, \phi_C(x)))^2}$
(Note this has been normalized so easy to compare to 4-center cost)

Answer (i):
Answer (ii):
Answer (iii):

B: (20 points) Since `k-means++` is a randomized algorithm, you will need to report the variation.

(i) Run it several trials (at least 20) and plot the *cumulative density function* of the 4-means cost.

(ii) Report what fraction of the time the subsets are the same as the result from `Gonzalez`.

Answer (i):
Answer (ii):

C: (30 points) Recall that Lloyd's algorithm for k-means clustering starts with a set of k centers C and runs as described in Algorithm 8.3.1 (in M4D).

(i) Run Lloyds Algorithm with C initially with points indexed `{1,2,3,4}`. Report the final clusters and the 4-means cost.

(ii) Run Lloyds Algorithm with C initially as the output of `Gonzalez` above. Report the final clusters and the 4-means cost.

(iii) Run Lloyds Algorithm with C initially as the output of each run of `k-means++` above. Plot a *cumulative density function* of the 4-means cost. Also report the fraction of the trials that the subsets are the same as the input (where the input is the result of `k-means++`).

Answer (i):
Answer (ii):
Answer (iii):

D: (1 *extra* point) Implement the first part of 2.C.(iii) using `KMeans` in `scikit-learn`. Documentation is available here (click on the word "here"). Report your code.

Answer:

```python
from sklearn.cluster import KMeans

# your code goes here
```

3 BONUS k-Median Clustering (5 points)

The k-median clustering problem on a data set P is to find a set of k-centers $C = \{c_1, c_2, \ldots, c_k\}$ to minimize $\texttt{Cost}_1(P, C) = \frac{1}{|P|} \sum_{p \in P}(p, \phi_C(p))$. We did not explicitly talk much about this formulation in class, but the techniques to solve it are typically all extensions of the approaches we did talk about. This problem will be more open-ended and will ask you to try various approaches to solve this problem. We will use data set `C3.txt`.

Find a set of 4 centers $C = \{c_1, c_2, c_3, c_4\}$ for the 4-medians problem on dataset `C3.txt`. Report the set of centers, as well as $\texttt{Cost}_1(P, C)$. The centroids should be in the write-up you turn in, but also in a file formatted the same way as the input so we can verify the cost you found. That is each line has 1 center with 6 tab-separated numbers. The first is the index (e.g., 1, 2, 3 or 4), and the next 5 are the 5-dimensional coordinates of that center. Upload this file to this folder (click on "this folder"). You must use your `gcloud.utah.edu` credentials to access this folder. Report your filename in the submission pdf.

Your score will be based on how small a $\texttt{Cost}_1(P, C)$ you can find. You can get 2 points for a reasonable solution. The smallest found score in the class will get all 5 points. Other scores will obtain points in between.

Very briefly describe how you found the centers.

Answer: