CSCI 1052 Problem Set 3

January 21, 2024

Submission Instructions

Please upload your solutions by **5pm Friday January 26, 2024.**

- You are encouraged to discuss ideas and work with your classmates. However, you **must acknowledge** your collaborators at the top of each solution on which you collaborated with others and you **must write** your solutions and code independently.
- Your solutions to theory questions must be typeset in LaTeX or markdown. I strongly recommend uploading the source LaTeX (found here) to Overleaf for editing.
- I recommend that you write your solutions to coding question in a Jupyter notebook using Google Colab.
- You should submit your solutions as a **single PDF** via the assignment on Gradescope. You can enroll in the class using the code GPXX7N.
- Once you uploaded your solution, **mark where you answered each part of each question**.

Problem 1: Distance Reconstruction

Suppose you are given all pairwise distances between a set of points $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$. You can assume that $d << n$. Let $\mathbf{D} \in \mathbb{R}^{n \times n}$ be the distance matrix with $\mathbf{D}_{i,j} = \|\mathbf{x}_i - \mathbf{x}_j\|_2^2$. You would like to recover the location of the original points, at least up to possible rotations and translations which do not change pairwise distances. Assume that $\sum_{i=1}^n \mathbf{x}_i = \mathbf{0}$.

We can learn the sum of norms $\sum_{i=1}^n \|\mathbf{x}_i\|_2^2$ from $\mathbf{D}$. In particular,

$$\sum_{i=1}^n \sum j = 1^n \mathbf{D}_{i,j} = \sum_i \sum_j \|\mathbf{x}_i\|_2^2 + \|\mathbf{x}_j\|_2^2 - 2\mathbf{x}_i^\top \mathbf{x}_j = \sum_i \left(\sum_j \|\mathbf{x}_i\|_2^2 + \|\mathbf{x}_j\|_2^- 2\mathbf{x}_i^\top \sum_j \mathbf{x}_j \right).$$

By our assumption that the points are centered around the origin i.e., $\sum_j \mathbf{x}_j = \mathbf{0}$, we can conclude that

$$\sum_i \sum_j \mathbf{D}_{i,j} = \sum_i \sum_j \|\mathbf{x}_i\|_2^2 + \|\mathbf{x}_j\|_2^2 = 2n \sum_i \|\mathbf{x}_i\|_2^2.$$

Part 1 (2 points)

Inspired by the above approach, describe an efficient algorithm for learning $\|\mathbf{x}_i\|_2^2$ for each i.

Next, describe an algorithm for recovering a set of points $\mathbf{x}_1, \ldots, \mathbf{x}_n$ which realize the distances in $\mathbf{D}$. **Hint:** This is where you will use the SVD! It might help to prove that $\mathbf{D}$ has rank $\leq d + 2$.

Part 2 (1 point)

Implement your algorithm and run it on the U.S. cities dataset provided in `UScities.txt`[1] or `UScities.csv`[2]. Note that the distances in the file are unsquared Euclidean distances, so you need to square them to obtain $\mathbf{D}$. Plot your estimated city locations on a 2D plot and label the cities to make it clear how the plot is oriented. Submit these images and your code with the problem set.

[1] `https://www.rtealwitter.com/rads2024/psets/UScities.txt`
[2] `https://www.rtealwitter.com/rads2024/psets/UScities.csv`