Homework 3

AMATH 352, Fall 2022

Due on Oct 24, 2022 at midnight.

Directions, Reminders and Policies

Read these instructions carefully:

- **You are required to upload a PDF report to Canvas along with a zip of your code.**
- The report should be a maximum of 3 pages long with references included. Minimum font size 10pts and margins of at least 1inch on A4 or standard letter size paper.
- Do not include your code in the report. Simply create a zip file of your main scripts and functions, without figures or data sets included, and upload the zip file to Canvas.
- Your report should be formatted as follows:
  - Title/author: Title of report, your name and email address. This is not meant to be a separate title page.
  - Sec. 1. Introduction and overview of the problem.
  - Sec. 2. Theoretical background and description of algorithms.
  - Sec. 3. Computational Results
  - Sec. 4. Summary and Conclusions
  - References
- I suggest you use LaTeX(Overleaf is a great option) to prepare your reports. A template is provided on Canvas under the Syllabus tab. You are also welcome to use Microsoft Word or any other software that properly typesets mathematical equations.
- I encourage collaborations, however, everything that is handed in (both your report and your code) should be your work.
- Your homework will be graded based on how completely you solved it as well as neatness and little things like: did you label your graphs and include figure captions. **The homework is worth 10 points. 5 points will be given for the overall layout, correctness and neatness of the report, and 5 additional points will be for specific technical things and computational results that the TAs will look for in the report itself.**

Warm up

Look up the following functions and commands in Python:

- Generating random numbers: `numpy.random.randn`, `numpy.random.seed`
- Computing norms and inner products: `numpy.linalg.norm`, `numpy.dot`
- Arc cosine: `numpy.arccos`
- Histograms: `matplotlib.pyplot` and in particular the `hist` function, how to plot histograms, and how to label axes in a plot.
- How to implement for loops in Python.

Problem Description

Your goal in this HW is to investigate the linear dependence of certain collections of vectors and study the effectiveness of the Gram-Schmidt procedure.

Part I

1. Consider a collection of N random Gaussian vectors $\underline{a}_i \in \mathbb{R}^d$ for $i \in \{1, \ldots, N\}$. We write a_{ij} to denote the j entry of $\underline{a}_i$ for $j \in \{1, \ldots, d\}$ and the entries a_{ij} are independently drawn from the standard normal distribution $N(0, 1)$. You can use `numpy.random.randn` to generate such random vectors.
2. Use the `numpy.random.seed(0)` command in your code to set your random seed and ensure reporducibility of your results.
3. Write a computer program to compute the pairwise angles between the random vectors $\underline{a}_i, \underline{a}_k$ for $i \neq k$.
4. Fix $d = 2^4$ and $N = 2^7$ and plot the histogram of the pairwise angles between the $\underline{a}_i$'s. Provide the same plot for $N = 2^8$ and 2^9.
5. Discuss your observations from the previous step. What can you say about the linear independence of random Gaussian vectors?

Part II

1. Consider a collection of vectors $\underline{b}_i \in \mathbb{R}^N$ for $i = 1, \ldots, N$ where

$$b_{ij} = \frac{1}{i + j - 1}.$$

2. Provide a histogram of the pairwise angles between the $\underline{b}_i$'s for $N = 2^7, 2^8, 2^9$. Comment on the linear dependence of these vectors.
3. Implement the Gram-Schmidt algorithm to construct an orthonormal set of vectors $\{\underline{q}_i\}_{i=1}^N$ from the $\underline{b}_i$'s.
4. Provide a histogram of the pairwise angles between the $\underline{q}_i$'s for the different values of N.
5. Discuss your observations. Does Gram-Schmidt perform as expected?