1 Coding (Favorite Monte Carlo algorithm).

Since this class is wrapping up, what is your favorite Monte Carlo algorithm and why? Provide a numerical illustration.

2 Coding (Randomized subspace iteration).

Apply the randomized SVD algorithm (randomized subspace iteration with $q = 1$) to approximate the diagonal matrix

$$\boldsymbol{A} = \operatorname{diag}(e^{-.1}, e^{-.2}, \ldots, e^{-999.9}).$$

The approximation will be quite accurate if you use a block size $k \geq 50$. Next, try to approximate a matrix $\boldsymbol{B}$ which is a small entrywise perturbation of $\boldsymbol{A}$:

$$b_{ij} = a_{ij} + Z_{ij}, \qquad Z_{ij} \overset{\text{iid}}{\sim} \mathcal{N}(0, 0.002^2).$$

Approximating $\boldsymbol{B}$ will be hard, necessitating a large block size k and/or number of iterations q. Why?

3 Coding (Randomly pivoted Cholesky)

Randomly pivoted Cholesky selects "diverse" columns to approximate a matrix $\boldsymbol{A}$. To see this, generate random data points $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}$ from a mixture of Gaussians with centers at $(+2, +2)$, $(+2, -2)$, $(-2, +2)$, and $(-2, -2)$ and variance $1/4$. Define the kernel matrix $\boldsymbol{A} \in \mathbb{R}^{N \times N}$ with entries $a_{ij} = \exp(-\|\boldsymbol{x}^{(i)} - \boldsymbol{x}^{(j)}\|^2/2)$, which quantifies the similarity between data points (close to one means similar, close to zero means dissimilar). Apply randomly pivoted Cholesky to approximate the matrix $\boldsymbol{A}$. Which data points correspond to the first 4 selected rows? How similar is the approximate matrix $\hat{\boldsymbol{A}}$ to the target matrix $\boldsymbol{A}$?

4 Coding (k-means++).

If you have ever used k-means to cluster the rows of a data matrix $\boldsymbol{A} \in \mathbb{R}^{L \times N}$, you have used randomized numerical linear algebra. The standard initialization for k-means, called k-means++ builds up a randomized rank-k approximation $\hat{\boldsymbol{A}} \in \mathbb{R}^{L \times N}$. We initialize the approximation by setting $\hat{\boldsymbol{A}} = \boldsymbol{0}$. Then, we perform k updates by randomly selecting a row index $s \in \{1, \ldots, L\}$ with

$$\mathbb{P}\{s = i\} = \sum\nolimits_{j=1}^{N} |a_{ij} - \hat{a}_{ij}|^2.$$

and update each row of $\hat{A}$ as $\hat{a}_{ij} = a_{sj}$ for $1 \leq j \leq N$ if

$$\sum\nolimits_{j=1}^{N} |a_{ij} - a_{sj}|^2 \leq \sum\nolimits_{j=1}^{N} |a_{ij} - \hat{a}_{ij}|^2.$$

Apply k-means++ to the kernel matrix $\boldsymbol{A} \in \mathbb{R}^{N \times N}$ constructed in problem #3. Which data points correspond to the first 4 selected rows? How similar is the approximate matrix $\hat{\boldsymbol{A}}$ to the target matrix $\boldsymbol{A}$? Compare and contrast with randomly pivoted Cholesky.

5 Choose-your-own-adventure (Monte Carlo connection).

Can we view randomized subspace iteration and randomly pivoted Cholesky as Monte Carlo algorithms? If so, how?

6 Math (Nyström approximation for psd matrices).

To approximate a general, rectangular matrix $\boldsymbol{A} \in \mathbb{R}^{L \times N}$, we can form a low-rank approximation

$$\hat{\boldsymbol{A}} = \boldsymbol{\Pi}_{\boldsymbol{X}} \boldsymbol{A} = \boldsymbol{X}\boldsymbol{Y}^*, \qquad \boldsymbol{Y} = \boldsymbol{A}^* \boldsymbol{X}, \tag{1}$$

where $\boldsymbol{X} \in \mathbb{R}^{L \times k}$ is a rank-k orthogonal matrix. If $\boldsymbol{A} \in \mathbb{R}^{N \times N}$ is positive semidefinite (psd), we can alternatively form the Nyström approximation

$$\hat{\boldsymbol{A}} = \boldsymbol{A}^{1/2} \boldsymbol{\Pi}_{\boldsymbol{A}^{1/2}\boldsymbol{X}} \boldsymbol{A}^{1/2} = \boldsymbol{Y}(\boldsymbol{X}^*\boldsymbol{Y})^{\dagger}\boldsymbol{Y}^*. \tag{2}$$

Which approximation leads to a smaller Frobenius norm error, (1) or (2)? Provide a proof.

7 Math (Largest eigenvalue).

Let's prove some bounds for randomized subspace iteration.

(a) If we use randomized subspace iteration to approximate the largest singular value $\sigma_1(\boldsymbol{A})$ of a matrix $\boldsymbol{A} \in \mathbb{R}^{L \times N}$, prove that

$$\hat{\sigma}_1(\boldsymbol{A})^2 = \max_{\boldsymbol{\omega} \in \operatorname{range}(\boldsymbol{\Omega})} \frac{\boldsymbol{\omega}^*(\boldsymbol{A}^*\boldsymbol{A})^{2q}\boldsymbol{\omega}}{\boldsymbol{\omega}^*(\boldsymbol{A}^*\boldsymbol{A})^{2q-1}\boldsymbol{\omega}}, \tag{3}$$

where $\boldsymbol{\Omega} \in \mathbb{R}^{N \times k}$ is the Gaussian initialization matrix.

(b) Show that the distribution of $\hat{\sigma}_1(\boldsymbol{A})^2$ only depends on the singular values of $\boldsymbol{A}$, regardless of the singular vectors.

(c) Now assume without loss of generality that $\boldsymbol{A}$ is diagonal and psd, with non-increasing diagonal entries and rewrite (3) in a simpler form.

(d) Partition $\boldsymbol{\Omega} = \left[\begin{smallmatrix} \boldsymbol{\Omega}_1 \\ \boldsymbol{\Omega}_2 \end{smallmatrix}\right]$ and set $\boldsymbol{\omega}$ to be the first column of $\boldsymbol{\Omega}\boldsymbol{\Omega}_1^{\dagger}$. Use (3) to obtain write down relatively simple, explicit lower and upper bounds on $\hat{\sigma}_1(\boldsymbol{A})^2$.

(e) What random matrix theory do we need to bound the expected value of $\hat{\sigma}_1(\boldsymbol{A})^2$?