Nicholas Carolan
email@princeton.edu
COS302-F23

Assignment #4

Due: 6:00pm Wednesday 4 October 2023

Discussants: N/A

Upload at: https://www.gradescope.com/courses/606160/assignments/3377028

Problem 1 (20pts) (A) Compute an orthonormal basis of the kernel of

$$\boldsymbol{A} = \begin{bmatrix} 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & 1 \end{bmatrix}$$

(B) Write down an orthonormal basis for the image of $\boldsymbol{A}$.

(A) We first find the kernel (null space) of A by finding the non-trivial solutions to $Ax = 0$.

$$\begin{bmatrix} 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 & -1 & 1 \\ 0 & 2 & -2 & 2 & 0 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 & -1 & 1 \\ 0 & 1 & -1 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & -1 & 1 & 0 \end{bmatrix}$$

The general solution is $x_1 + x_5 = 0$, $x_2 - x_3 + x_4 = 0$. Rearranged, $x_1 = -x_5$ and $x_2 = x_3 - x_4$.

The kernel space is $x = \begin{bmatrix} -x_5 \\ x_3 - x_4 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = x_3 \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} + x_4 \begin{bmatrix} 0 \\ -1 \\ 0 \\ 1 \\ 0 \end{bmatrix} + x_5 \begin{bmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$.

We can perform Gram-Schmidt orthogonalization on the kernel vectors to find an orthonormal basis:

$$u_1 = \frac{v_1}{||v_1||} = \frac{\begin{bmatrix} 0 & 1 & 1 & 0 & 0 \end{bmatrix}^T}{\sqrt{1^2 + 1^2}} = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \end{bmatrix}^T$$

$$u_2 = \frac{v_2 - (v_2^T u_1)u_1}{||v_2 - (v_2^T u_1)u_1||} = \frac{1}{||v_2 - (v_2^T u_1)u_1||}\left(\begin{bmatrix} 0 \\ -1 \\ 0 \\ 1 \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ -1 \\ 0 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \\ 0 \end{bmatrix}\right) = \frac{1}{||v_2 - (v_2^T u_1)u_1||}\left(\begin{bmatrix} 0 \\ -1 \\ 0 \\ 1 \\ 0 \end{bmatrix} - \frac{-1}{\sqrt{2}} \begin{bmatrix} 0 \\ \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ 0 \\ 0 \end{bmatrix}\right) = \frac{1}{\sqrt{1.5}} \begin{bmatrix} 0 \\ \frac{-1}{2} \\ \frac{1}{2} \\ 1 \\ 0 \end{bmatrix}$$

$$u_3 = \frac{v_3 - (v_3^T u_1)u_1 - (v_3^T u_2)u_2}{||v_3 - (v_3^T u_1)u_1 - (v_3^T u_2)u_2||} = \frac{\begin{bmatrix} -1 & 0 & 0 & 0 & 1 \end{bmatrix}^T - 0 - 0}{\sqrt{(-1)^2 + 1^2}} = \frac{1}{\sqrt{2}} \begin{bmatrix} -1 & 0 & 0 & 0 & 1 \end{bmatrix}^T$$

We can confirm the basis u_1, u_2, u_3 is orthonormal: $u_1 \cdot u_2 = u_1 \cdot u_3 = u_2 \cdot u_3 = 0$ and $||u_1|| = ||u_2|| = ||u_3|| = 1$.

(B) The image of A is the span of its linearly independent columns. From the reduced row-echelon form in (A) above, we see that x_1 and x_2 are the pivot columns. We can construct $\text{IM}(A)$ from those columns: $\text{IM}(A) = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$

$$u_1 = \frac{v_1}{||v_1||} = \frac{\begin{bmatrix} 1 & 1 \end{bmatrix}^T}{\sqrt{1^2 + 1^2}} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$u_2 = \frac{v_2 - (v_2^T u_1)u_1}{||v_2 - (v_2^T u_1)u_1||} = \frac{\begin{bmatrix} -1 \\ 1 \end{bmatrix} - \begin{bmatrix} -1 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}}{||v_2 - (v_2^T u_1)u_1||} = \frac{\begin{bmatrix} -1 \\ 1 \end{bmatrix}}{\sqrt{(-1)^2 + 1^2}} = \frac{1}{\sqrt{2}} \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

Problem 2 (23pts)

You've encountered power series before in other classes, but one thing you may not've realized is that you can construct *matrix functions* from *matrix power series*. That is, if you have a function $f(\cdot)$ that has a convergent power series representation:

$$f(x) = \sum_{i=0}^{\infty} a_i x^i$$

then you can generally write a similar matrix version for square symmetric matrices $\boldsymbol{X}$ using the same a_i:

$$F(\boldsymbol{X}) = \sum_{i=0}^{\infty} a_i \boldsymbol{X}^i$$

(A) The matrix version F turns out to just apply the scalar f to each eigenvalue independently. Explain why. (Hint: Write down the diagonalization of $\boldsymbol{X}$ and then multiply it by itself. What happens? How would a diagonalized version of $\boldsymbol{X}$ interact with the power series?)

(B) In power series there is a notion of radius of convergence. How would you expect this concept to generalize to square symmetric matrices?

(C) One important example is where the function $f(x)$ is the exponential function. I can take any square symmetric matrix and if I compute its matrix exponential, I get a positive definite matrix. Explain why.

(A) We can diagonalize X as $X = Q\Lambda Q^T$. We know that the powers of this expression only change the inner term. For example: $X^2 = (Q\Lambda Q^\top)(Q\Lambda Q^\top) = Q\Lambda^2 Q^\top$. In general, $X^k = Q\Lambda^k Q^\top$. We can redefine our function F to take advantage of this:

$$F(X) = \sum_{i=0}^{\infty} a_i X^i = \sum_{i=0}^{\infty} a_i \, Q\Lambda^i Q^\top = Q\Big(\sum_{i=0}^{\infty} a_i \Lambda^i\Big)Q^\top.$$

Because the eigenvalue matrix Λ is diagonal, the result is simply applying $f(x)$ to each eigenvalue λ in Λ.

$$f(\lambda_1), \ldots, f(\lambda_n) = \sum_{i=0}^{\infty} a_i \lambda_1^i, \ldots, \sum_{i=0}^{\infty} a_i \lambda_n^i$$

(B) For scalars, the power series for f converges when $|x|$ is less than the radius of convergence R.

In square symmetric matrices, the matrix power series will diverge if applying f to any eigenvalue diverges, because diagonalization means that that the power applies only to the eigenvalues and the constant a. This applies a limit across all eigenvalues:

$$F(X) = Q\Big(\sum_{i=0}^{\infty} a_i \Lambda^i\Big)Q^\top \quad \text{converges if } \max_k |\lambda_k| < 1 \text{ for all } \lambda_k \text{ in } \Lambda.$$

(C) We can express the matrix exponential as a power series, and diagonalize similarly to **(A)**:

$$e^X = \sum_{k=0}^{\infty} \frac{X^k}{k!} = \sum_{k=0}^{\infty} \frac{(Q\Lambda Q^\top)^k}{k!} = Q\left(\sum_{k=0}^{\infty} \frac{\Lambda^k}{k!}\right)Q^\top = Q\, e^\Lambda \, Q^\top$$

Note that $e^\Lambda = \operatorname{diag}(e^{\lambda_1}, \ldots, e^{\lambda_n})$.

Because e^λ is always positive, every eigenvalue of e^X is positive. A symmetric matrix is positive definite iff all its eigenvalues are positive.

Problem 3 (25pts)
Consider the following set of vectors in $\mathbb{R}^3$:

$$v_1 = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \qquad v_2 = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \qquad v_3 = \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix}.$$

(A) Perform Gram-Schmidt orthogonalization on this set of vectors to obtain an orthonormal set $\{u_1, u_2, u_3\}$. Show all your intermediate steps and calculations.

(B) Once you have the vectors, verify that they are orthogonal by calculating the dot products between all possible pairs of vectors in the set.

(A)

$$u_1 = \frac{v_1}{||v_1||} = \frac{\begin{bmatrix} 1 & 2 & 1 \end{bmatrix}^T}{\sqrt{1^2+2^2+1^2}} = \frac{1}{\sqrt{6}} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}.$$

$$u_2 = \frac{v_2 - (v_2^T u_1)u_1}{||v_2 - (v_2^T u_1)u_1||} = \frac{\begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} - \begin{bmatrix} 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{6}} \\ \frac{2}{\sqrt{6}} \\ \frac{1}{\sqrt{6}} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{6}} \\ \frac{2}{\sqrt{6}} \\ \frac{1}{\sqrt{6}} \end{bmatrix}}{||v_2 - (v_2^T u_1)u_1||} = \frac{\begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} - 0}{||1^2 + (-1)^2||} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$$

$$u_3 = \frac{v_3 - (v_3^T u_1)u_1 - (v_3^T u_2)u_2}{||v_3 - (v_3^T u_1)u_1 - (v_3^T u_2)u_2||} = \frac{\begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix} - \left(\frac{7}{\sqrt{6}}\right) \frac{1}{\sqrt{6}} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} - \left(-\frac{1}{\sqrt{2}}\right) \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}}{\left\| \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix} - \frac{7}{6} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \right\|} = \frac{\begin{bmatrix} \frac{4}{3} \\ -\frac{4}{3} \\ \frac{4}{3} \end{bmatrix}}{\frac{4}{3}\sqrt{3}} = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}$$

(B)

$$u_1 \cdot u_2 = \frac{1}{\sqrt{6}\sqrt{2}} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} = \frac{1}{\sqrt{12}} (1 \cdot 1 + 2 \cdot 0 + 1 \cdot (-1)) = 0$$

$$u_1 \cdot u_3 = \frac{1}{\sqrt{6}\sqrt{3}} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{18}} (1 \cdot 1 + 2 \cdot (-1) + 1 \cdot 1) = 0$$

$$u_2 \cdot u_3 = \frac{1}{\sqrt{2}\sqrt{3}} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{6}} (1 \cdot 1 + 0 \cdot (-1) + (-1) \cdot 1) = 0$$

Problem 4 (30pts)

One of the most important algorithms in data analysis is principal component analysis or PCA. PCA tries to find a way to represent high-dimensional data in a low-dimensional way so that human brains can reason about it. It tries to identify the "important" directions in a data set and represent the data just in that basis. PCA does this by computing the empirical covariance matrix of the data (we'll learn more about that in a couple of weeks), and then looking at the eigenvectors of it that correspond to the largest eigenvalues.

(A) Load `mnist2000.pkl` into a Colab notebook. Take the $2000 \times 28 \times 28$ tensor of training data and reshape it so that it is a 2000×784 matrix, where the rows are "unrolled" image vectors. Typically in PCA, one first centers the data. Center the data by subtracting off the mean image; you did a very similar procedure in HW2.

(B) Now compute the "scatter matrix" which is the 784×784 matrix you get from multiplying data matrix by its transpose, making sure that you get it so the data dimension is the one being summed over.

(C) This scatter matrix is square and symmetric, so use the `eigh` function in the `numpy.linalg` package to compute the eigenvalues and eigenvectors. Plot the eigenvalues in decreasing order.

(D) Read the documentation for `eigh` and figure out how to get the "big" eigenvectors. For each of the top five eigenvectors, reshape them into 28×28 images and use `imshow` to render them.

(E) Now, create a low-dimensional representation of the data. Take the 2000×784 matrix and multiply it by each of the top two eigenvectors. This takes all 2000 data, each of which are 784-dimensional, and gives them two-dimensional coordinates. Make a scatter plot of these two-dimensional coordinates.

(F) That scatter plot doesn't really give you much of a visualization. Here's some starter code to build a more interesting figure. It takes the two-dimensional projection and builds a "scatter plot" where the images themselves are rendered instead of dots. Here I have the projections in a 2000×2 matrix called `proj`, which I modify so that all the values are in $[0, 1]$.

```
# Make the projections into [0,1]
proj = proj - np.min(proj, axis=0)
proj = proj / np.max(proj, axis=0)

# Create a 12" x 12" figure.
viz_fig = pl.figure(figsize=(12.,12.))

# Get the figure width and height in pixels.
width, height = viz_fig.get_size_inches()*viz_fig.dpi

pl.plot() # Colab seems to require this to render.

# Loop over images.  Could do all 2000 but it's crowded.
for ii in range(400):
  # Render each image in a location on the figure.
  pl.figimage(train_images[ii,:,:],
              xo=proj[ii,1]*width,
              yo=(proj[ii,0]*height-150), # hack to make visible
              origin='upper')
```

Modify this code to work with your projections and make a visualization of the MNIST digits. Do you see any interesting structure?

Problem 5 (2pts)
Approximately how many hours did this assignment take you to complete?

My notebook URL: `https://colab.research.google.com/drive/1edbRi4f_yBe9lKe7bH2uTd-yvu3CghBA?usp=sharing`

Changelog

- 20 Sept 2023 – Initial F23 version