Homework

Exercise 1.1 Load the image dog.jpg and compute the full SVD. Choose a rank $r < m$ and confirm that the matrix $\mathbf{U}^*\mathbf{U}$ is the $r \times r$ identity matrix. Now confirm that $\mathbf{U}\mathbf{U}^*$ is *not* the identity matrix. Compute the norm of the error between $\mathbf{U}\mathbf{U}^*$ and the $n \times n$ identity matrix as the rank r varies from 1 to n and plot the error.

Exercise 1.2 Load the image dog.jpg and compute the economy SVD. Compute the relative reconstruction error of the truncated SVD in the Frobenius norm as a function of the rank r. Square this error to compute the fraction of missing variance as a function of r. You may also decide to plot 1 minus the error or missing variance to visualize the amount of norm or variance captured at a given rank r. Plot these quantities along with the cumulative sum of singular values as a function of r. Find the rank r where the reconstruction captures 99% of the total variance. Compare this with the rank r where the reconstruction captures 99% in the Frobenius norm and with the rank r that captures 99% of the cumulative sum of singular values.

Exercise 1.3 Load the Yale B image database and compute the economy SVD using a standard `svd` command. Now compute the SVD with the method of snapshots. Compare the singular value spectra on a log plot. Compare the first 10 left singular vectors using each method (remember to reshape them into the shape of a face). Now compare a few singular vectors farther down the spectrum. Explain your findings.

Exercise 1.4 Generate a random 100×100 matrix, i.e., a matrix whose entries are sampled from a normal distribution. Compute the SVD of this matrix and plot the singular values. Repeat this 100 times and plot the distribution of singular values in a box-and-whisker plot. Plot the mean and median singular values as a function of r. Now repeat this for different matrix sizes (e.g., 50×50, 200×200, 500×500, 1000×1000, etc.).

Exercise 1.5 Compare the singular value distributions for a 1000×1000 uniformly distributed random matrix and a Gaussian random matrix of the same size. Adapt the Gavish–Donoho algorithm to filter uniform noise based on this singular value distribution. Add uniform noise to a data set (either an image or the test low-rank signal) and apply this thresholding algorithm to filter the noise. Vary the magnitude of the noise and compare the results. Is the filtering good or bad?

Exercise 1.6 This exercise will test the concept of condition number. We will test the accuracy of solving $\mathbf{A}\mathbf{x} = \mathbf{b}$ when noise is added to $\mathbf{b}$ for matrices $\mathbf{A}$ with different condition numbers.

(a) To build the two matrices, generate a random $\mathbf{U} \in \mathbb{R}^{100\times100}$ and $\mathbf{V} \in \mathbb{R}^{100\times100}$ and then create two $\mathbf{\Sigma}$ matrices: the first $\mathbf{\Sigma}$ will have singular values spaced logarithmically from 100 to 1, and the second $\mathbf{\Sigma}$ will have singular values spaced logarithmically from 100 to 10^{-6}. Use these matrices to create two $\mathbf{A}$ matrices, one with a condition number of 100 and the other with a condition number of 100 million. Now create a random $\mathbf{b}$ vector, solve for $\mathbf{x}$ using the two methods, and compare the results. Add a small $\boldsymbol{\epsilon}$ to $\mathbf{b}$, with norm 10^{-6} smaller than the norm of $\mathbf{b}$. Now solve for $\mathbf{x}$ using this new $\mathbf{b} + \boldsymbol{\epsilon}$ and compare the results.

(b) Now repeat the experiment above with many different noise vectors $\boldsymbol{\epsilon}$ and compute the distribution of the error; plot this error as a histogram and explain the shape.

(c) Repeat the above experiment comparing two $\mathbf{A}$ matrices with different singular value distributions: the first $\mathbf{\Sigma}$ will have values spaced linearly from 100 to 1 and the second $\mathbf{\Sigma}$ will have values spaced logarithmically from 100 to 1. Does anything change? Please explain why yes or no.

(d) Repeat the above experiment, but now with an $\mathbf{A}$ matrix that has size 100×10. Explain any changes.

Exercise 1.7 Load the data set for fluid flow past a cylinder (you can either download this from our book http://DMDbook.com or generate it using the IBPM code on GitHub). Each column is a flow field that has been reshaped into a vector.

(a) Compute the SVD of this data set and plot the singular value spectrum and the leading singular vectors. The $\mathbf{U}$ matrix contains eigenflow fields and the $\mathbf{\Sigma V}^*$ represents the amplitudes of these eigenflows as the flow evolves in time.

(b) Write a code to plot the reconstructed movie for various truncation values r. Compute the r value needed to capture 90%, 99%, and 99.9% of the flow energy based on the singular value spectrum (recall that energy is given by the Frobenius norm squared). Plot the movies for each of these truncation values and compare the fidelity. Also compute the squared Frobenius norm of the error between the true matrix $\mathbf{X}$ and the reconstructed matrix $\tilde{\mathbf{X}}$, where $\mathbf{X}$ is the flow field movie.

(c) Fix a value $r = 10$ and compute the truncated SVD. Each column $\mathbf{w}_k \in \mathbb{R}^{10}$ of the matrix $\mathbf{W} = \tilde{\mathbf{\Sigma}}\tilde{\mathbf{V}}^*$ represents the mixture of the first 10 eigenflows in the kth column of $\mathbf{X}$. Verify this by comparing the kth snapshot of $\mathbf{X}$ with $\tilde{\mathbf{U}}\mathbf{w}_k$.

(d) Now, build a linear regression model for how the amplitudes $\mathbf{w}_k$ evolve in time. This will be a dynamical system:

$$\mathbf{w}_{k+1} = \mathbf{A}\mathbf{w}_k.$$

Create a matrix $\mathbf{W}$ with the first 1 through $m-1$ columns of $\mathbf{\Sigma V}^*$ and another matrix $\mathbf{W}'$ with the 2 through m columns of $\mathbf{\Sigma V}^*$. We will now try to solve for a best-fit $\mathbf{A}$ matrix so that

$$\mathbf{W}' \approx \mathbf{A}\mathbf{W}.$$

Compute the SVD of $\mathbf{W}$ and use this to compute the pseudo-inverse of $\mathbf{W}$ to solve for $\mathbf{A}$. Compute the eigenvalues of $\mathbf{A}$ and plot them in the complex plane.

(e) Use this $\mathbf{A}$ matrix to advance the state $\mathbf{w}_k = \mathbf{A}^{k-1}\mathbf{w}_1$ starting from $\mathbf{w}_1$. Plot the reconstructed flow field using these predicted amplitude vectors and compare with the true values.

This exercise derived the dynamic mode decomposition from Section 7.2.