

Homework Assignment #2

Due date: 9/20/18, before class. Please L^AT_EX or handwrite your homework solution and submit an electronic version.

Exercise 1 (PCA and low-rank compression) We are given a $n \times m$ matrix $X = [x_1, \dots, x_m]$, with $x_i \in \mathbb{R}^n$, $i = 1, \dots, m$ the data points. We assume that the data matrix is centered, in the sense that $x_1 + \dots + x_m = 0$. In lecture, it was asserted that there is equivalence between three problems:

- (P_1) Finding a line going through the origin that maximizes the variance of the points projected on the line.
- (P_2) Finding a line going through the origin that minimizes the sum of squares of the distances from the points to their projections;
- (P_3) Finding a rank-one approximation to the data matrix.

In this exercise, you are asked to show the equivalence between these three problems.

1. Consider the problem of projecting a point x on a line $\mathcal{L} = \{x_0 + vu : v \in \mathbb{R}\}$, with $x_0 \in \mathbb{R}^n$, $u^T u = 1$, given. Show that the projected point z is given by $z = x_0 + v^* u$, with $v^* = (x - x_0)^T u$, and that the minimal squared distance $\|z - x\|_2^2$ is equal to $\|x - x_0\|_2^2 - ((x - x_0)^T u)^2$.
2. Show that problems P_1, P_2 are equivalent.
3. Show that P_3 is equivalent to P_1 . *Hint:* show that the data matrix is rank-one if and only if the points are all on a line that goes through the origin.

Exercise 2 (Generalized Eigenvalues and Image Segmentation)

In this exercise, we will implement an algorithm for image segmentation from a graph-theoretic approach. As a simplified problem, we will focus on separating the foreground of an image from its background. We provide you with a skeleton code in the archive `image_segmentation.zip` that imports an image and guide you through the implementation.

An image is described as a matrix M of shape (N, M) whose values represent the gray scale color normalized between zero and one. Even though we restrict now to gray images, please note that the same algorithm could be easily extended to colored images.

An undirected graph is a triplet (V, E, W) where V is the set of nodes, E the set of edges, and W the set of weights for each edge. As a first step, we would like to construct a graph that represents the image. For that purpose, we choose V to be the set of pixels. We say that there is an edge (i, j) between pixel i and pixel j , $1 \leq i, j \leq NM$, if they are neighbors in the image (adjacent in the matrix M). For each edge (i, j) , we will use as weight $w_{ij} = 1 - \text{abs}(M_{x_i, y_i} - M_{x_j, y_j})$, where (x_i, y_i) represents the location of pixel i and (x_j, y_j) represents the location of pixel j . This is a measure of pixel similarity. Finally, we are interested in the whole affinity matrix W of shape (NM, NM) , where each entry has the value zero if there is no edge between the pixels and the value w_{ij} if there is an edge.

1. Complete the code in the notebook `image-seg.ipynb` that transforms the image `img-cup-small.png` into the affinity matrix W .

Let D be the diagonal matrix of shape (NM, NM) , where the i^{th} entry equals the sum of weights associated with edges incident on pixel i . That is, $D(i) = \sum_j W(i, j)$. We wish to solve the generalized eigenvalue problem $(D - W)y = \lambda Dy$ for all eigenvectors and all eigenvalues.

This eigenvalue problem is a known approximation to finding a partition of a graph that minimize the normalized cut in graph theory. Also, if the graph is connected, then the largest eigenvalue of the adjacency matrix as well as the smallest eigenvalue of the Laplacian have multiplicity 1. We can expect that the gap between this and the nearest eigenvalue is related to some kind of connectivity measure of the graph (for more information on the general topic of graph eigenvalues, one can refer to <http://web.cs.elte.hu/~lovasz/eigenvals-x.pdf>). The partition that approximate the solution to the normalized cut problem is derived from the generalized eigenvector with the second smallest eigenvalue and can be straightforwardly used in image segmentation.

2. Explain how to relate this problem to the symmetric eigenvalue problem we saw in class. Furthermore, explain how you would solve this problem provided routine code to compute a SVD.

3. Find the generalized eigenvector y for the eigenvector with the second smallest eigenvalue in the case of your given image. You will rely on linear algebra routine libraries like `numpy.linalg`.

The i -th entry of y can be viewed as a soft indicator of the component membership of the i -th pixel (foreground or background).

4. Use this indication to transform your image into a binary image (the so-called result of image segmentation). Try separating pixels corresponding to positive / negative values and above the median / below the median values of the entries of y to split the image into two components. Plot the obtained binary image and comment on your results.

Exercise 3 (PCA and Senate Voting Data) We return to the Senate voting data examined in HW1, with X the $m \times n$ data matrix, where each row corresponds to a Senator, and each column to a bill.

1. Find a n -vector a and scalar b such that the variance of the corresponding score function $f(x) = a^T x + b$ is maximized.
2. How does the variance obtained previously compare to the one obtained with a set to the center of the data points, and b set so that the average score is zero? Comment on the phrase “senators vote according to the party average”.
3. What is the total variance explained by the first two principal components? Plot the data projected on the corresponding plane.
4. Based on the first principal component, which bill(s) would you say have been the most important? Which Senators are the most “extreme”?

Exercise 4 (Diet Planning)

We consider a set of n basic foods (such as rice, beans, apples) and a set of m nutrients or components (such as protein, fat, sugar, vitamin C). Food j has a cost given by c_j (say, in dollars per gram), and contains an amount N_{ij} of nutrient i (per gram). (The nutrients are given in some appropriate units, which can depend on the particular nutrient.) A daily diet is represented by an n -vector d , with d_i the daily intake (in grams) of food i .

1. Express the condition that a diet d contains exactly the total nutrient amounts given by the m -vector n_{des} , and has a total cost B (the budget) as a set of linear equations in the variables d_1, \dots, d_n . (The entries of d must be nonnegative, but we ignore this issue here.)
2. Now suppose we are given $n + 1$ new constraints that requires how different the daily diet d is from some predetermined set diet $s_i \in \mathbb{R}^n$, for $i = 1, \dots, n + 1$, using the 2-norm as a measure of difference. That is, we have

$$\|d - s_i\|_2 = p_i$$

Express these constraints as a set of n constraints **linear** in d .

Exercise 5 (A result related to Gaussian distributions)

Let $\Sigma \in S_{++}^n$ be a symmetric, positive definite matrix. Show that

$$\int_{\mathbb{R}^n} e^{-\frac{1}{2}x^T \Sigma^{-1} x} dx = (2\pi)^{\frac{n}{2}} \sqrt{\det \Sigma}.$$

You may assume known that the result holds true when $n = 1$. The above shows that the function $p = \mathbb{R}^n \rightarrow \mathbb{R}$ with (non-negative) values

$$p(x) = \frac{1}{(2\pi)^{n/2} \cdot \sqrt{\det \Sigma}} e^{-\frac{1}{2}x^T \Sigma^{-1} x}$$

integrates to one over the whole space. In fact, it is the density function of a probability distribution called the multivariate Gaussian (or normal) distribution, with zero mean and covariance matrix Σ .

$$\int_{x \in \mathbb{R}^n} f(x) = |\det P| \cdot \int_{z \in \mathbb{R}^n} f(Pz) dz.$$

Exercise 6 (Analysis of calendar days) Consider a matrix made up of numerical representations of a range \mathcal{T} of calendar days. For example, the matrix below represents the range \mathcal{T} starting on November 1, 2013 and ending on November 3, 2013:

$$X = \begin{pmatrix} 2013 & 11 & 1 \\ 2013 & 11 & 2 \\ 2013 & 11 & 3 \end{pmatrix}$$

1. Form a matrix X for the date range \mathcal{T} starting on November 1, 2003 and ending on October 31, 2013.
2. Find an SVD of the matrix, and print out the right singular vectors and corresponding singular values.
3. What is the exact rank of X ? Numerically, would you say that the rank of the matrix is “low”? Comment.
4. Plot the left singular vectors. Do these vectors exhibit a pattern? How would you interpret that pattern?
5. Explain the significance of the right singular vector v corresponding to the smallest singular value. How would you approximate such a vector by hand?
6. How would you expect the condition number (ratio from largest to smallest singular value) to behave as the number of calendar days in \mathcal{T} increases?