

EE263 Final Exam, December 2019

This is a 24 hour take-home exam. Please turn it in using Gradescope 24 hours after you receive it.

- Please turn in your code along with your solutions.
- You may use any books, notes, or computer programs (*e.g.*, Julia, Python or Matlab). You can Google anything. But **you may not discuss the exam with anyone** until Dec. 11, after everyone has taken the exam. The only exception is that you can ask the TAs or Professor Lall for clarification, by emailing to the staff email address `ee263-aut1920-staff@lists.stanford.edu`. We've tried pretty hard to make the exam unambiguous and clear, so we're unlikely to say much.
- Since you have 24 hours, we expect your solutions to be legible, neat, and clear. Do not hand in your rough notes, and please try to simplify your solutions as much as you can. We will deduct points from solutions that are technically correct, but much more complicated than they need to be.
- When a problem involves some computation, we do not want just the final answers. We want a clear discussion and justification of exactly what you did. Be sure to show us your verification that your computed solution satisfies whatever properties it is supposed to, at least up to numerical precision. For example, if you compute a vector x that is supposed to satisfy $Ax = b$ (say), show us the code that checks this, and the result. (This might be done by the code `norm(A*x-b)`; be sure to show us the result, which should be very small.) *We will not check your numerical solutions for you, in cases where there is more than one solution.*
- In the portion of your solutions where you explain the mathematical approach, you *cannot* refer to programming language operators, such as the backslash operator. (You can, of course, refer to inverses of matrices, or any other standard mathematical construct.)
- Some of the problems may be described in a practical setting, (imaging systems, energy consumption, population dynamics, or wireless communications). Others may discuss mathematical areas (Markov chains, differential equations). *You do not need to understand anything about these practical settings or mathematical areas to solve these problems.* We've taken special care to make sure all the information and math needed to solve the problem is given in the problem description.
- Some of the problems require you to download data files. These files can be found at the URL

<http://ee263.stanford.edu/finalquestions.html>

There are no links on the course web page pointing there, so you'll have to type in the whole URL yourself. The data files are in `json` format. In Julia, use `readclassjson` to load the files. In Matlab, use `jason_helper.m` to load the files. In Python, use `jason_helper.py` to load the files. These are available at the above web page.

- At the same web page as the data files we will post notification of any exam typos or problems. We recommend you check this site occasionally during the exam.
- For problems involving coding, please include your codes as part of your pdf submission and correctly assign them to the corresponding problems on gradescope. Make sure to double check that the pages have been correctly assigned for each problem, and don't wait until the last minute to submit.
- Good luck!

1. Periodic drone motion. You are operating a drone with the goal of continuously observing a particular set of locations. For example, you may be monitoring bird nesting sites to keep track of populations and behavior. You know that the birds tend to feed at particular locations at particular times of day, so you want to have the drone be at those locations at the right times.

We'll model the drone as obeying Newton's laws, which is a reasonable model at this scale. At time t , the drone will be at position $y(t) \in \mathbb{R}^2$ and acted on by a force $u(t) \in \mathbb{R}^2$. We'll discretize, assuming piecewise constant forces, with a sampling period of 1. This gives the equations of motion as a discrete-time linear dynamical system as follows.

$$x(t+1) = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} \frac{1}{2} & 0 \\ 1 & 0 \\ 0 & \frac{1}{2} \\ 0 & 1 \end{bmatrix} u(t)$$

The state is $x = (y_1, \dot{y}_1, y_2, \dot{y}_2)$. We have m nesting sites, with the i th nesting site given by the i th column of P , where

$$P = \begin{bmatrix} 0.8 & 1.2 & -0.9 & -1.0 & 0.2 \\ -1.2 & 1.0 & 1.2 & -1.0 & -1.1 \end{bmatrix}$$

We would like to visit these sites at times s_i , where

$$s = [10.0 \quad 15.0 \quad 40.0 \quad 50.0 \quad 70.0]$$

- a) Our first objective is to have the drone visit the sites. Design a trajectory for which $y(s_i) = p_i$ for all $i = 1, \dots, m$. We will have the drone start stationary at the origin. We would of course like to minimize battery usage, and so we decide to minimize

$$J_{\text{effort}} = \sum_{t=0}^{T-1} \|u(t)\|^2$$

Here $T = 90$ is the flight time. Find the optimal trajectory of the drone. What is the minimum value of J_{effort} ?

Plot u as a function of time. Also make a plot which shows the route the drone takes, and includes the waypoints. This should have axes y_1 and y_2 .

- b) Actually we want the drone to continually monitor these sites, and so we want the drone to go back to visiting p_1 again after finishing visiting p_m . So we would like to design a periodic trajectory y which satisfies

$$y(t+T) = y(t) \text{ for all } t \geq 0$$

This of course means that the drone will visit the nesting sites periodically, so we will have

$$y(s_i + kT) = p_i \text{ for all } k \geq 0$$

We also do *not* need the drone to start from stationary at the origin. The position and velocity of the drone at time $t = 0$ are also to be chosen. Find the periodic trajectory that minimizes J_{effort} . What is the minimum value of J_{effort} ? Make the same two plots as in the previous part.

- c) The drone is actually controlled remotely from a base station at the origin, and so we need the drone to stay within range $r = 2$. To encourage this behavior, we decide to add an additional objective, of minimizing

$$J_{\text{range}} = \sum_{t=0}^{T-1} \|y(t)\|^2$$

Of course, we still want the drone to visit the sites at the right times. For positive γ , find the trajectory that minimizes

$$J_{\text{range}} + \gamma J_{\text{effort}}$$

Plot the trade-off curve of J_{range} versus J_{effort} . For each value of γ in

$$\gamma = 1, 10, 100, 10^3, 10^4, 10^5, 10^6$$

plot the route the drone takes, and the waypoints.

- d) Find (by searching) the largest γ such that the drone stays within range? What is the corresponding J_{range} and J_{effort} ? For this γ , plot the route the drone takes, and the waypoints.

2. Affine sets.

- a) An affine set $\mathcal{A} \subset \mathbb{R}^m$ is a translated subspace. More precisely it is a set of the form

$$\mathcal{A} = \{x \mid Ax = a\}$$

where $A \in \mathbb{R}^{m \times n}$ and $a \in \mathbb{R}^m$. This way of writing an affine set is called the *implicit form*. An affine set can always be written in another form, called the *explicit form*, as follows. There is always an integer k , a matrix $C \in \mathbb{R}^{n \times k}$ and a vector $c \in \mathbb{R}^n$ such that

$$\mathcal{A} = \{Cz + c \mid z \in \mathbb{R}^k\}$$

Suppose you are given A and a . Give a method for finding k , C and c . What determines the smallest k which can be used?

- b) We can also convert from explicit to implicit form. Give an algorithm for finding A, a given C, c .

c) Apply your method to

$$C = \begin{bmatrix} 37 & -100 & -15 \\ -16 & 230 & 20 \\ -66 & 100 & -10 \\ 96 & -34 & 52 \\ 11 & 24 & -20 \end{bmatrix} \quad c = \begin{bmatrix} 2 \\ 15 \\ 2 \\ -13 \\ 2 \end{bmatrix}$$

d) We now turn to computing the distance between two affine sets. For two sets $\mathcal{A}, \mathcal{B} \subset \mathbb{R}^m$, define the distance between them to be

$$\text{dist}(\mathcal{A}, \mathcal{B}) = \min_{x \in \mathcal{A}, y \in \mathcal{B}} \|x - y\|$$

We have two affine sets in implicit form, which are given by

$$\mathcal{A} = \{x \mid Ax = a\} \quad \mathcal{B} = \{y \mid By = b\}$$

We are given the matrices $A, B \in \mathbb{R}^{m \times n}$ and the vectors $a, b \in \mathbb{R}^m$, and would like to compute the distance $\text{dist}(\mathcal{A}, \mathcal{B})$. We would like to convert this problem to a least-squares problem. Explain how to find a matrix F and a vector g such that

$$\text{dist}(\mathcal{A}, \mathcal{B}) = \min_z \|Fz + g\|$$

3. Constraints. For the following problems, if relevant, let $A \in \mathbb{R}^{n \times n}$. For each part, you must justify your answer for credit.

a) Let $f(t) = t^{10} + 7t^9 - 6t^8 + \pi t^7 - \pi t^6 + 5t^3 + 2t - 7$ and $\mathcal{C} = \text{span}(\{1, t^2, t^4, t^6, t^8, t^{10}\})$. What is the optimal $g(t)$ to

$$\min_{g(t) \in \mathcal{C}} \int_{-1}^1 |f(t) - g(t)|^2 dt?$$

b) Find the solution (*i.e.*, the optimal X) to

$$\min_{X \in \mathbb{R}^{n \times n}} \|X - A\|_F^2.$$

c) Find the solution (*i.e.*, the optimal X) to

$$\min_{X \in \mathcal{A}} \|X - A\|_F^2,$$

where \mathcal{A} is the set of all $n \times n$ matrices that are a scalar multiple of the identity matrix.

d) Find the solution (*i.e.*, the optimal X) to

$$\min_{X \in \mathcal{B}} \|X - A\|_F^2,$$

where \mathcal{B} is the set of all $n \times n$ symmetric matrices.

e) Find the solution (*i.e.*, the optimal X) to

$$\min_{X \in \mathcal{D}} \|X - H\|_F^2,$$

where $\mathcal{D} = \{Z \mid Z \succeq \text{diag}(d)\}$, $d \in \mathbb{R}^n$ is a known vector, and $H \in \mathbb{R}^{n \times n}$ is a symmetric matrix. Recall $\text{diag} : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times n}$ constructs a diagonal matrix with diagonal equal to its argument. For example, $\text{diag}(\mathbf{1}) = I$. Also recall that a matrix $W \succeq 0$ if and only if W is positive semidefinite.

(Fun fact / very small hint: \mathcal{D} is a generalization of the set of positive semidefinite matrices.)

4. Solving risk-constrained portfolio optimization in closed form. We consider the problem of selecting a portfolio composed of n assets. Here $x_i \in \mathbb{R}$ denotes the investment, measured in dollars, in asset i . If $x_i < 0$ this means that we hold a short position in this asset. The vector $x \in \mathbb{R}^n$ is called the *holding* vector. By holding such a portfolio, we have a mean return, given by $\mu^\top x$. Here $\mu \in \mathbb{R}^n$ is called the *return* vector.

There is a type of risk exposure in holding such a portfolio due to the fluctuations in the returns of the assets. This fluctuation is characterized by the matrix $\Sigma \in \mathbb{R}^{n \times n}$, which is symmetric and positive definite. (One can make a statistical justification for using the return covariance matrix, but we don't need that in this question.) The amount of risk associated with a particular holding is then $x^\top \Sigma x$.

The goal is to maximize the total return, subject to the constraint that $x^\top \Sigma x \leq \rho$, where $\rho > 0$ is a number which expresses our tolerance for overall risk. That is, we would like to solve

$$\begin{aligned} & \text{maximize} && \mu^\top x \\ & \text{subject to} && x^\top \Sigma x \leq \rho \end{aligned}$$

- a) It turns out that this problem can be solved analytically; there is an explicit expression for the optimal x in terms of μ , ρ and Σ . Find such an expression. You may use common techniques from linear algebra (SVD, eigenvalues, norms, etc.)
- b) Is the solution unique? Please verify your claim.
- c) Show that the optimal holding vector x satisfies $\|x\| \geq \sqrt{\rho / \|\Sigma\|}$.
- d) For $\Sigma = I$, provide a geometric interpretation of the solution(s).

5. Embedding data based on pairwise distances. In this problem, you will design an algorithm to visualize relationships between potentially high dimensional or non-numeric data in lower dimensions. Consider the setting where m objects have been analyzed, and for each object $i = 1, \dots, m$ we have a *feature vector* $x_i \in \mathbb{R}^n$. Unfortunately n is large, and so we would like to replace these high-dimensional data points with lower dimensional data points $y_1, \dots, y_m \in \mathbb{R}^d$. Often we want $d = 2$ or $d = 3$ so we can visualize the data. We would like the new data points y_i to roughly preserve the arrangement of the high-dimensional data x_i . Specifically, we want

$$\|y_i - y_j\| \approx \|x_i - x_j\| \quad \text{for all } i, j$$

- a) We construct a matrix of the data $X = \begin{bmatrix} x_1 & x_2 & \dots & x_m \end{bmatrix}$. Let D^X be the corresponding matrix of squared pairwise distances squared

$$D_{ij}^X = \|x_i - x_j\|^2$$

and let G_{ij}^X be the matrix of pairwise inner products

$$G_{ij}^X = x_i^\top x_j$$

For a square matrix $G \in \mathbb{R}^{m \times m}$, use the notation $g = \text{diag } G$ to mean $g \in \mathbb{R}^m$ and $g_i = G_{ii}$; that is g is the vector of elements on the diagonal of G . Show that

$$D^X = (\text{diag } G^X) \mathbf{1}^\top + \mathbf{1}(\text{diag } G^X)^\top - 2G^X$$

- b) Define the square matrix

$$H = I - \frac{\mathbf{1}\mathbf{1}^\top}{m}$$

Suppose $XH = X$. What does this mean geometrically?

- c) Suppose $XH = X$ (you may assume this for the rest of the problem). Show that

$$HD^X H = -2G^X$$

Notice this means that we have a one-to-one map between G^X and D^X .

- d) Given any symmetric matrix $D \in \mathbb{R}^{m \times m}$, we can ask the question whether the entries of the matrix D could possibly be the pairwise distances between a hypothetical set of data; that is, does there exist n and $X \in \mathbb{R}^{n \times m}$ such that $D = D^X$. Show that this holds if and only if

$$HDH \preceq 0.$$

Also recall that a matrix $W \succeq 0$ if and only if W is positive semidefinite.

- e) Now let's address our original problem, of finding points y_1, \dots, y_m whose pairwise distance are close to those of x_1, \dots, x_m . That is, we are looking to make an embedding of the m data points in d dimensions. Ideally we would like $D^Y \approx D^X$. We know G^X and D^X are related as above, and so

$$G^X - G^Y = -\frac{1}{2}H(D^X - D^Y)H$$

So if $D^X \approx D^Y$ then we must have $G^X \approx G^Y$. We therefore decide to choose $Y \in \mathbb{R}^{d \times m}$ to minimize

$$\|G^X - G^Y\|_F$$

Explain how to do this.

- f) And now let's do it! We have provided 3 data sets. From the file `USmap.json`, `Dmap` is the pairwise spherical distances between US cities. Use the technique you developed above to find an embedding in 2 dimensions (this corresponds to flattening a globe onto a map). The list `labels` contains the names of the cities. Please plot your embedding with the cities labeled (what you see should look familiar).

The files `digits2and6.json` and `digits8and9.json` contain `D2and6` and `D8and9` corresponding to the pairwise distances squared of 8x8 images of handwritten digits (the former from a subset of 2's and 6's and the latter from 8's and 9's), as well as the identities of each image (which digit it was) stored in `labels`. Use the same technique to embed the originally 64 dimensional dataset in 2 dimensions, and color the points according to their label (one color for each of the two digits). In one or two sentences comment on what you notice between the two datasets's embeddings.

6. Worst-case confidence ellipsoid. Consider the scenario where we observe

$$y = Ax + e,$$

Here $y \in \mathbb{R}^m$ is measured, $x \in \mathbb{R}^n$ is unknown, and $A \in \mathbb{R}^{m \times n}$ is a matrix that represents the effects of our sensors. You may assume A is skinny and full rank. In a sensing problem, we often measure y and try to determine x . This is made more difficult by e , which represents noise or measurement uncertainty. We do not know the vector e , but we do know that $\|e\| \leq 1$.

- a) Since we do not know e , we cannot typically determine x with certainty. Instead, if we know A and y , there is a set of x consistent with the model above. This is called the *worst-case confidence ellipsoid* associated with the measurement. Let S be this set. That is

$$S = \{x \in \mathbb{R}^n \mid y = Ax + e \text{ for some } e \text{ with } \|e\| \leq 1\}$$

For example, let

$$A = \begin{bmatrix} 3 \\ 1 \end{bmatrix} \quad y = \begin{bmatrix} 4 \\ 2 \end{bmatrix}$$

What is \mathcal{S} ?

Sketch this situation. Clearly label y , the range of A , and the set \mathcal{S} .

b) In general there exists $c \in \mathbb{R}^n$ and $Q \in \mathbb{R}^{n \times n}$ such that

$$\mathcal{S} = \{x \mid (x - c)^T Q^{-1} (x - c) \leq 1\}$$

Find c and Q .

c) Given

$$A = \begin{bmatrix} -2.3 & -1.1 \\ -1.3 & -1.5 \\ 0.5 & 0.2 \\ 0.2 & 0.4 \\ -0.3 & 0.5 \end{bmatrix}, \quad y = (0, 1, 0, 0, 1),$$

and $\|e\|_2 \leq 1$, find the set of all possible x consistent with this information.