EE263 Prof. S. Lall

Dec. 6–7, Dec. 7–8, Dec. 8–9, Dec. 9–10 or Dec. 10–11, 2013.

Final exam

This is a 24 hour take-home final exam. Please turn it in at Bytes Cafe in the Packard building, 24 hours after you pick it up.

Please read the following instructions carefully.

- You may use any books, notes, or computer programs (*e.g.*, Matlab), but you may not discuss the exam with anyone until December 11, after everyone has taken the exam. The only exception is that you can ask the TAs or Professor Lall for clarification, by sending an email to the staff email address. 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.

- Please check your email a few times during the exam, just in case we need to send out a clarification or other announcement. It's unlikely we'll need to do this, but you never know.

- Attach the official exam cover page (available when you pick up or drop off the exam) to your exam, and assemble your solutions to the problems in order, *i.e.*, problem 1, problem 2, . . . , problem 6. Start each solution on a new page. Do not collect all plots or code (for example) at the end of the exam; plots for problem 3 (say) should be with your solution to problem 3. Failure to follow these instructions will result in a penalty.

- Please make a copy of your exam before handing it in. We have never lost one, but it might occur.

- When a problem involves some computation (say, using Matlab), we do not want just the final answers. We want a clear discussion and justification of exactly what you did, the Matlab source code that produces the result, and the final numerical result. 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 Matlab code that checks this, and the result. (This might be done with the Matlab 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 Matlab 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 are described in (what appears to be) a practical setting. *You do not need to understand anything about the application area 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.

- Please be careful to avoid committing crimes against matrices, as outlined in the notes on the course web page. Special penalties will apply for any of these crimes committed during the final exam.

- Some of the problems require you to download and run a Matlab file to generate the data needed. These files can be found at the URL

  `http://www.stanford.edu/class/ee263/aut1314_final/FILENAME`

  where you should substitute the particular filename (given in the problem) for `FILENAME`. *There are no links on the course web page pointing to these files, so you'll have to type in the whole URL yourself.*

- Please respect the honor code. Although we encourage you to work on homework assignments in small groups, *you cannot discuss the final with anyone*, with the exception of Professor Lall and the TAs, until everyone has taken it.

1. *Planning a maximal excursion and return.* Consider a vehicle that moves in the plane with dynamics

$$x(t+1) = Ax(t) + Bu(t), \qquad p(t) = Cx(t), \qquad t = 0, 1, 2, \ldots,$$

where $p(t) \in \mathbf{R}^2$ is the position of the vehicle, $x(t) = (p(t), \dot{p}(t)) \in \mathbf{R}^4$ is the state, and $u(t) \in \mathbf{R}$ is the input. The initial state is $x(0) = 0$. Given times T_1 and T_2 with $0 < T_1 < T_2$, you want to move the vehicle as far from the origin as possible at time T_1: that is, you want to maximize

$$\|p(T_1)\|,$$

while returning the vehicle to rest at the origin at time T_2: that is,

$$x(T_2) = (p(T_2), \dot{p}(T_2)) = 0.$$

In addition, your input sequence must satisfy a fuel-consumption constraint:

$$\sum_{t=0}^{T_2-1} u(t)^2 \leq F^{\max}.$$

(a) Given A, B, C, T_1, T_2, and $F^{\max}$, explain how to find an input sequence that maximizes $\|p(T_1)\|$ subject to $x(T_2) = 0$, and the fuel-consumption constraint.

(b) Apply your method to the data given in `max_excursion_data.m`. Plot the trajectory of the vehicle in the plane (clearly indicate the initial and final locations), plot the optimal input sequence as a function of time, and report the optimal value of $\|p(T_1)\|$.

(c) For the data in `max_excursion_data.m`, is the optimal input sequence unique? If not, how many distinct optimal input sequences are there?

2. *Intersections of subspaces.* Suppose $A \in \mathbf{R}^{m \times n}$ and $B \in \mathbf{R}^{m \times p}$. Recall that if S is a subset of $\mathbf{R}^n$, then we define

$$AS = \{y \in \mathbf{R}^m \mid y = Ax \text{ for some } x \in S\}.$$

(a) Prove that $\mathcal{R}(A) \cap \mathcal{R}(B) = \begin{bmatrix} A & 0 \end{bmatrix} \mathcal{N}(\begin{bmatrix} A & B \end{bmatrix})$.

(b) Prove that $\mathcal{R}(A) \cap \mathcal{N}(B^T) = A\mathcal{N}(B^T A)$.

(c) Prove that $\mathcal{N}(A^T) \cap \mathcal{N}(B^T) = \mathcal{N}(\begin{bmatrix} A & B \end{bmatrix}^T)$.

(d) Prove that if A and B are skinny and full rank and $p+n > m$, then $\mathcal{R}(A) \cap \mathcal{R}(B)$ contains a nonzero vector.

3. *Quality control with destructive measurements.* You are in charge of quality control for a widget factory. In order to determine if a widget is satisfactory, you use m quality metrics, labeled $1, \dots, m$. A widget is deemed satisfactory if quality metric i is greater than or equal to a given threshold t_i, for all $i = 1, \dots, m$. You want to evaluate a batch of n widgets, labeled $1, \dots, n$. Define the matrix $Q \in \mathbf{R}^{m \times n}$ such that

$$q_{ij} = \text{ the value of quality metric } i \text{ for widget } j.$$

It is easy to measure quality metrics $1, \dots, m_1$, where $m_1 < m$, but measuring quality metrics $m_1 + 1, \dots, m$ destroys the widget. For example, one such destructive quality metric may be the lifetime of the widget – the only way to determine this quantity is to operate the widget until it fails. You propose the following system for quality control.

- Choose some $n_1 < n$, and evaluate quality metrics $1, \dots, m$ for widgets $1, \dots, n_1$. These widgets are completely characterized, but destroyed in the process.
- Evaluate quality metrics $1, \dots, m_1$ for widgets $n_1 + 1, \dots, n$, and then estimate quality metrics $m_1 + 1, \dots, m$ for these widgets based on your measurements. Reject widget j if the measured quality metrics $1, \dots, m_1$ or the estimated quality metric $m_1 + 1, \dots, m$ do not meet the given thresholds.

Let $m_2 = m - m_1$ and $n_2 = n - n_1$. Partition the matrix Q into blocks:

$$Q = \begin{bmatrix} A & B \\ C & X \end{bmatrix},$$

where $A \in \mathbf{R}^{m_1 \times n_1}$, $B \in \mathbf{R}^{m_1 \times n_2}$, $C \in \mathbf{R}^{m_2 \times n_1}$ and $X \in \mathbf{R}^{m_2 \times n_2}$. In terms of this partition, your scheme is to measure A, B and C, and then use these measurements to estimate X. If the ensemble of quality metrics for a widget depends mostly on a few crucial steps in the production process. Then, we expect Q to be low rank, and it makes sense to choose X in order to minimize $\mathbf{Rank}(Q)$.

(a) Show that for any choice of X, we have that

$$\mathcal{R}\left(\begin{bmatrix} A \\ C \end{bmatrix}\right) \subseteq \mathcal{R}\left(\begin{bmatrix} A & B \\ C & X \end{bmatrix}\right).$$

(b) Now suppose that A is fat and full rank. Explain how to find an X such that

$$\mathcal{R}\left(\begin{bmatrix} A \\ C \end{bmatrix}\right) = \mathcal{R}\left(\begin{bmatrix} A & B \\ C & X \end{bmatrix}\right).$$

Conclude that such an X is a solution of

$$\min_X \mathbf{Rank}(Q) = \min_X \mathbf{Rank}\left(\begin{bmatrix} A & B \\ C & X \end{bmatrix}\right).$$

(c) Apply your method to the specific instance of this problem defined in the file `quality_control_data.m`. Report the minimum rank of Q, and which widgets are deemed unsatisfactory.

Special note. You must solve this problem exactly. You will not receive any credit for using the alternating-projections heuristic.

4. *Magnitude and phase plots.* A polar representation of a vector $x \in \mathbf{R}^2$ is a pair $(\|x\|, \angle(x))$ such that

$$x = \begin{bmatrix} \|x\| \cos(\angle(x)) \\ \|x\| \sin(\angle(x)) \end{bmatrix}.$$

In particular, any unit vector $z \in \mathbf{R}^2$ has the form

$$z(\theta) = \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix}.$$

Consider a matrix $A \in \mathbf{R}^{2\times 2}$. A *magnitude plot* for A is a plot of $\|Az(\theta)\|$ as a function of θ, and a *phase plot* for A is a plot of $\angle(Az(\theta))$ as a function of θ. We can learn a lot about a matrix by inspecting its magnitude and phase plots.

- Figures 1-4 contain magnitude and phase plots for two matrices, A_1 and A_2. Use these plots to complete the tables below.
- No justification is needed, and we will not grade any answers that are not in the form of such tables.
- Each of your answers must be in terms of labeled quantities on the magnitude and phase plots (for example, you could write something like $90° + r_3 \log(\theta_2)$, although this particular answer is nonsense), except for the answers in the last table, which must be either "Yes" or "No".
- There may be multiple correct answers for some entries of the table, but please only give one.

(a) Let a_1 and a_2 be the columns of A. Complete the following table.

	$\|a_1\|$	$\angle(a_1)$	$\|a_2\|$	$\angle(a_2)$
A_1				
A_2				

(b) Let $A = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T$ be a singular-value decomposition of A, where σ_i is the ith singular value of A, and u_i and v_i are corresponding left and right singular vectors, respectively. Complete the following table.

	σ_1	u_1	v_1	σ_2	u_2	v_2
A_1						
A_2						

(c) Let λ_1 and λ_2 be the eigenvalues of A, where $|\lambda_1| \geq |\lambda_2|$, and let q_1 and q_2 be corresponding unit eigenvectors. Complete the following table.

	λ_1	q_1	λ_2	q_2
A_1				
A_2				

(d) We say that a matrix A is positive semidefinite, and write $A \geq 0$, if $x^T A x \geq 0$ for all $x \in \mathbf{R}^2$; in particular, we do *not* require that A be symmetric. Complete the following table.

	$A \geq 0$
A_1	
A_2	

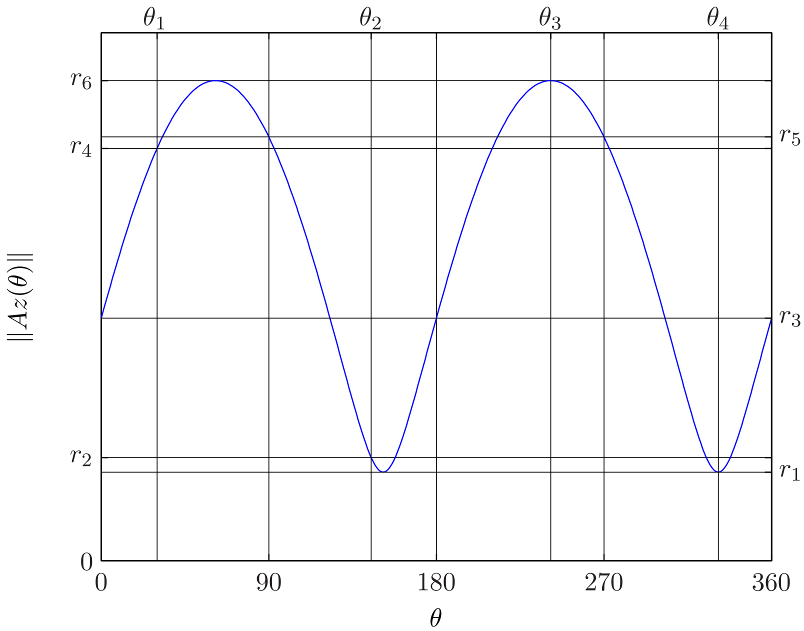

Figure 1: magnitude plot for the matrix A_1

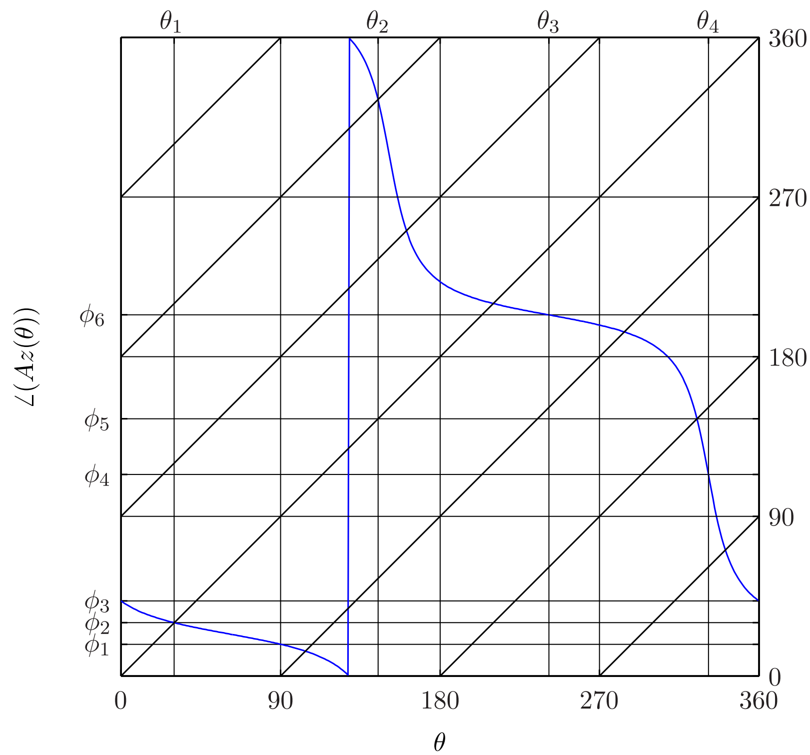

Figure 2: phase plot for the matrix A_1

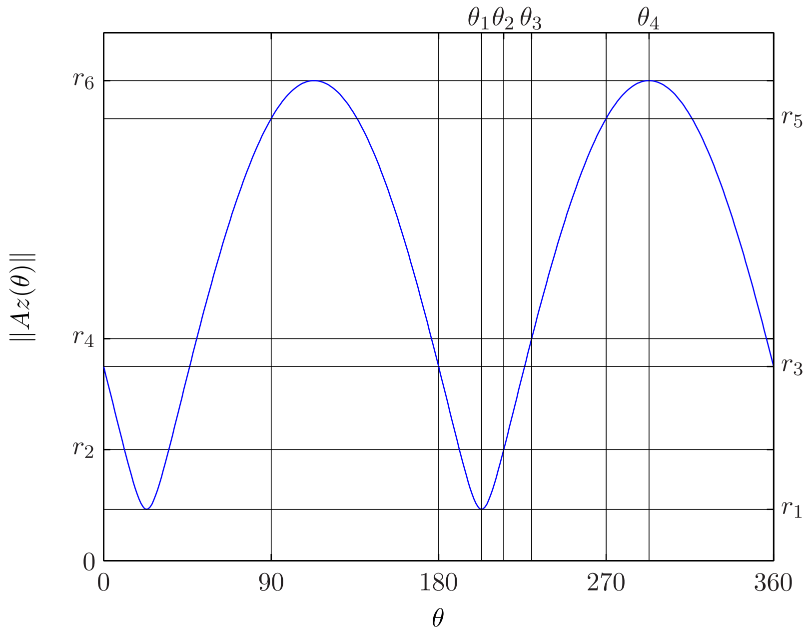

Figure 3: magnitude plot for the matrix A_2

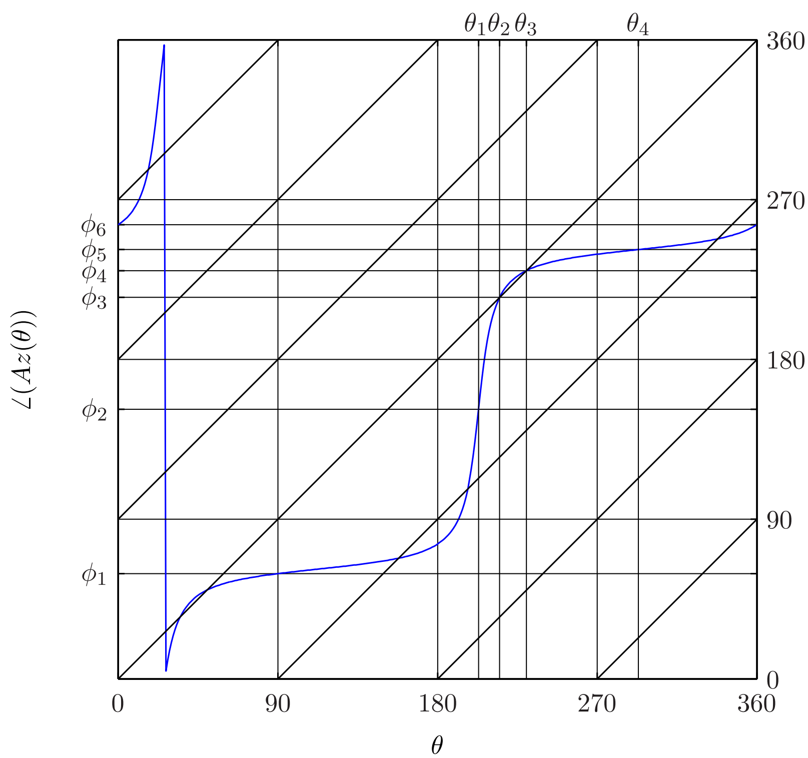

Figure 4: phase plot for the matrix A_2

5. *A market with two firms.* In this problem we consider a market with 2 firms, labeled 1 and 2, and n goods, labeled $1, \ldots, n$. Let $q_i^{(j)} \in \mathbf{R}$ denote the quantity of good i produced by firm j. (In a more realistic model, we might insist that $q_i^{(j)} \geq 0$, but you may ignore this complication throughout the problem.) Additionally, let c_i denote the cost of producing one unit of good i, and let p_i denote the market price of one unit of good i. We assume that $c_1, \ldots c_n$ are given, and that the vector $p = (p_1, \ldots, p_n) \in \mathbf{R}^n$ of market prices is a function of $q^{(1)}$ and $q^{(2)}$:

$$p = a - B(q^{(1)} + q^{(2)}),$$

where the vector $a \in \mathbf{R}^n$, and the symmetric, positive-definite matrix $B \in \mathbf{R}^{n \times n}$ are given. The profit of firm j is

$$\pi_j = (p - c)^T q^{(j)}.$$

We assume that $q^{(1)}$ and $q^{(2)}$ are chosen sequentially: first, firm 1 chooses $q^{(1)}$; then, firm 2 chooses $q^{(2)}$. Additionally, we assume that firm 2 knows $q^{(1)}$ when it chooses $q^{(2)}$.

(a) Having observed firm 1's choice of $q^{(1)}$, how should firm 2 choose $q^{(2)}$ in order to maximize π_2? Show that there is a unique optimal choice of $q^{(2)}$ corresponding to each value of $q^{(1)}$.

(b) Assuming firm 2 will observe $q^{(1)}$ and then choose $q^{(2)}$ in order to maximize π_2, how should firm 1 choose $q^{(1)}$ in order to maximize π_1?

(c) Assuming firm 1 chooses $q^{(1)}$ as in (b), and firm 2 choooses $q^{(2)}$ as in (a), what are π_1 and π_2? Is it possible for π_1 or π_2 to be negative? Which is greater: π_1 or π_2?

(d) The file `two_firms_data.m` defines values of a, B and c. Assuming firm 1 chooses $q^{(1)}$ as in (b), and firm 2 chooses $q^{(2)}$ as in (a), compute $q^{(1)}$, $q^{(2)}$, p, π_1 and π_2. Report π_1 and π_2.

6. *Fingerprint identification.* You are given a fingerprint found at a crime scene, represented by the locations $x^{(1)}, \dots, x^{(N)} \in \mathbf{R}^2$ of a set of markers. You also have such representations $y^{(1)}, \dots, y^{(N)} \in \mathbf{R}^2$ and $z^{(1)}, \dots, z^{(N)} \in \mathbf{R}^2$ of the fingerprints of two suspects. In this problem, you will use forensic linear algebra to determine which suspect's fingerprint most closely matches the one found at the crime scene. The file `fingerprint_data.m` defines the following variables.

   - `N`, the number of markers
   - `X`, a $2 \times N$ matrix whose kth column is $x^{(k)}$
   - `Y`, a $2 \times N$ matrix whose kth column is $y^{(k)}$
   - `Z`, a $2 \times N$ matrix whose kth column is $z^{(k)}$

   Running the data file also generates a scatterplot of the fingerprint data. The black `x`s represent the fingerprint found at the crime scene, the red `o`s represent the fingerprint of suspect Y, and the green □s represent the fingerprint of suspect Z.

   (a) Given a matrix $A \in \mathbf{R}^{n \times n}$, explain how to find an orthogonal matrix $Q \in \mathbf{R}^{n \times n}$ that maximizes $\mathbf{Tr}(AQ)$.

   (b) Suppose you are given two data sets: $x^{(1)}, \dots, x^{(N)} \in \mathbf{R}^n$ and $y^{(1)}, \dots, y^{(N)} \in \mathbf{R}^n$. (Although $n = 2$ for the fingerprint data, you should analyze the problem for general n.) In order to compare these data sets, you want to find an isometry T that minimizes the alignment error:

   $$E = \sum_{k=1}^{N} \|T(x^{(k)}) - y^{(k)}\|^2.$$

   Since T is an isometry, there exist an orthogonal matrix $Q \in \mathbf{R}^{n \times n}$ and a vector $t \in \mathbf{R}^n$ such that $T(x) = Qx + t$. For a fixed value of Q, explain how to choose t in order to minimize the alignment error.

   (c) Explain how to choose an orthogonal matrix Q and a vector t in order to minimize the alignment error.

   (d) Apply your method to the data in `fingerprint_data.m`. Report the optimal alignment errors between the crime-scene fingerprint, and the fingerprint of each suspect. For each suspect, make a scatterplot with the fingerprint from the crime scene and the aligned fingerprint from the suspect (you can look in the data file to see how to make such a plot). Which suspect's fingerprint most closely matches the one found at the crime scene?