• This is a 24-hour take-home exam. Please turn it in on Gradescope. Be aware that you 
  must turn it in within 24 hours of downloading it. After that, Gradescope will not let 
  you turn it in and we cannot accept it.

• You may use any books, notes, or computer programs, including searching online. You 
  may not discuss the exam or course material with others, or work in a group.

• The exam should not be discussed at all until 12/10 after everyone has taken the exam.

• If you have a question, please submit a private question on Ed, or email the staff mailing 
  list. We have tried very hard to make the exam unambiguous and clear, so unless there 
  is a mistake on the exam we’re unlikely to say much.

• 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 during the exam, just in case we need to send out a clarification 
  or other announcement.

• Start each solution on a new page.

• When a problem involves some computation (say, using Julia), we do not want just the 
  final answers. We want a clear discussion and justification of exactly what you did as 
  well as the final numerical result.

• Because this is an exam, you must turn in your code. Include the code in your pdf 
  submission. We reserve the right to deduct points for missing code.

• In the portion of your solutions where you explain the mathematical approach, you 
  cannot refer to Julia operators, such as the backslash operator. (You can, of course, 
  refer to inverses of matrices, or any other standard mathematical constructs.)

• Some of the problems require you to download data or other files. These files can be 
  found at the URL

  http://ee263.stanford.edu/goodbye21.html

• Good luck!
1. **Chasing a sea monster.** A sea monster is loose in the Pacific Ocean! Your monster-chasing colleague has been measuring the sea monster’s movements and has predicted it will surface at \(m\) positions \(p_i \in \mathbb{R}^2\) at times \(s_i\). Here \(p_i\) is the \(i\)th column of the matrix \(P\) given by

\[
P = \begin{bmatrix}
1 & 1.75 & 2.4 & 2 & 0.5 & 0 \\
0.75 & 0.6 & 1.2 & 2.3 & 0.75 & 0
\end{bmatrix}
\]

and the times \(s = (2, 5, 8, 11, 17, 20)\). You plan to observe the monster with a drone. Unfortunately the sea monster ate the last two drones you sent and you are almost out of research funding so your drone’s sensors are not very good, and the drone must be exactly in the right position to observe the monster.

a) The dynamics of the drone are

\[
\ddot{q} = u
\]

where \(q \in \mathbb{R}^2\) is the position of the drone, and \(u \in \mathbb{R}^2\) is an input force. Write this as a linear dynamical system of the form

\[
\begin{align*}
\dot{x} &= Ax + Bu \\
y &= Cx
\end{align*}
\]

where \(y \in \mathbb{R}^2\) is the position of the drone.

b) We will use sample period \(h\). Assume that the force input is piecewise constant on sample intervals, and construct the exact discretization

\[
\begin{align*}
x_d(k + 1) &= A_d x_d(k) + B_d u_d(k) \\
y_d(k) &= C_d x_d(k)
\end{align*}
\]

where \(x_d(k) = x(kh)\), and similarly for \(y_d\) and \(u_d\).

c) The drone starts at the origin with zero velocity, and we would like to move the drone so that \(y(s_i) = p_i\) for \(i = 1, \ldots, m\). We will operate the drone on the time interval \([0, T]\) where \(T = s_m\). For convenience, let \(N = T/h\). Since drone batteries are limited, we would like to minimize

\[
J = \sum_{k=0}^{N-1} \|u_d(k)\|^2
\]

Explain in detail how you would solve this problem.

d) Use your method to compute the optimal input \(u\), and plot \(u\) versus time. Use \(h = 0.1\).

e) Report the optimal value of \(J\) that you obtained.

f) Plot the trajectory of the drone. Use axes \(q_1\) and \(q_2\), so that the plot shows the path followed by the drone. Mark on your plot the points \(p_i\) where the monster surfaces.

g) Draw a sea monster for 1 point of extra credit.
2. **System in a box.** You are given a mysterious box containing two unit masses connected via springs and dampers as follows.

![Diagram of system in a box]

The equations of motion for this system are

\[
\ddot{q}_1 = -k_1 q_1 + k_2 (q_2 - q_1) - b_1 \dot{q}_1 + b_2 (\dot{q}_2 - \dot{q}_1)
\]

\[
\ddot{q}_2 = -k_2 (q_2 - q_1) - b_2 (\dot{q}_2 - \dot{q}_1)
\]

where \(k_i > 0\) are spring constants, \(b_i > 0\) are damping constants, and \(q_i\) is the displacement of mass \(i\). Both masses are \(m_i = 1\).

a) This mysterious box can be modeled as a continuous-time linear dynamical system

\[
\dot{x} = Ax
\]

Find \(A\) in terms of \(k_1, k_2, b_1, b_2\). Use state \(x = (q_1, q_2, \dot{q}_1, \dot{q}_2)\).

b) Use the forward Euler discretization

\[
x(t + h) = (I + hA)x(t)
\]

to simulate this system, with parameters \(k_1 = 1, k_2 = 2, b_1 = 1, b_2 = 3\). Set the initial conditions so that \(q_1(0) = 1, q_2(0) = 2, \dot{q}_1(0) = \dot{q}_2(0) = 0\). Use time step \(h = 0.1\), and simulate on time interval \([0, T]\) with \(T = 15\). Plot \(q_1, q_2, \dot{q}_1, \dot{q}_2\) (on one plot) as functions of time.

c) Unfortunately your dog ate the documentation for this system in a box so you do not know the parameters \(k_1, k_2, b_1, b_2\). However, we have experimental data, consisting of measurements of \(x(t)\) with sample period \(h = 0.1\) on time interval \([0, T]\) where \(T = 15\). These may be found in the file `box.json`. We will use this data to estimate \(k_1, k_2, b_1, b_2\). To do this, we will find the matrix \(A\) that minimizes

\[
\sum_{k=0}^{N-1} \| (I + hA)x(kh) - x(kh + h) \|^2
\]

where \(N = T/h\), and \(x\) is the given data. Explain how you would do this.

d) Apply your method from part (c) to estimate \(A\), and report the estimate you find.

e) Using initial conditions of \(x(0)\) in the dataset, and again with \(h = 0.1\) and \(T = 15\), simulate the system using your estimate of \(A\) and plot \(q_1, q_2, \dot{q}_1, \dot{q}_2\) (on one plot) as functions of time.
3. Measuring network latency. We have a network of three computers. There are six unidirectional links connecting them, so any computer may send a message directly to any of the others. If computer 1 sends a message to computer 2, it takes $l_{21}$ seconds to travel across the link connecting them. The quantity $l_{21}$ is called the latency of the link from 1 to 2. For a real network the link sends data at close to the speed of light, and so if the machines are in the same datacenter then we would expect $l_{21}$ to be measured in nanoseconds.

We would like to determine the latencies of the six links $l_{12}, l_{21}, l_{13}, l_{31}, l_{23}, l_{32}$. At first glance the problem seems simple; simply subtract the time at which the message is sent from the time at which the message is received. The difficulty is that this requires the two machines to have synchronized clocks, because any offset in the clock will directly give an error in the latency.

We can also use the computers to forward messages, so that for example computer 1 can send a message to computer 2, who forwards it to computer 3. The total time taken for the communication is then $l_{21} + l_{32}$, and we assume throughout that forwarding adds no additional waiting time at the intermediate computer (which is called a relay.)

To address the synchronization problem, we avoid using two clocks by forwarding messages in round trips. For example, we send a message from 1 to 2 to 3 to 1. We write this as $1 \rightarrow 2 \rightarrow 3 \rightarrow 1$. Then computer 1 can measure the round-trip time using only its clock. We assume that clocks can measure the duration of time accurately, even if they are not synchronized with some absolute standard. For this round-trip, we therefore measure $l_{21} + l_{32} + l_{13}.$

a) We measure the five round trips

$$1 \rightarrow 2 \rightarrow 1 \quad 1 \rightarrow 3 \rightarrow 1 \quad 2 \rightarrow 3 \rightarrow 2 \quad 1 \rightarrow 2 \rightarrow 3 \rightarrow 1 \quad 3 \rightarrow 2 \rightarrow 1 \rightarrow 3$$

which give measurements $y_1, \ldots, y_5$. Let $x$ be the vector of latencies

$$x = (l_{12}, l_{21}, l_{13}, l_{31}, l_{23}, l_{32})$$

Find the matrix $A$ such that $y = Ax$.

b) What is the rank of $A$?

c) It turns out that one cannot produce any vector $y \in \mathbb{R}^5$. There is a constraint that any measurement $y$ must satisfy. This constraint can be expressed in the form $c^T y = 0$, for some unit vector $c \in \mathbb{R}^5$. Find $c$. Give an interpretation of this constraint.

d) In general, we cannot determine the latencies $x$ given the measurements $y$. It is proposed that, if we had exact knowledge of two of the latencies, we could determining the other four. Your lab team sits down to discuss this, and comes up with three suggestions. We say that the scheme works if it enables us to determine all of the unknown latencies.

i. Bob’s suggestion: knowledge of any two of the six latencies will work.

ii. Jeff’s suggestion: we need to choose the two latencies carefully. Some choices will work, and some will not.

iii. Ruskin’s suggestion: there is no choice of two latencies that will work.

Who is right? If you think Jeff is right, give an example of two latencies that work, and two that will not, and explain. If you think Bob or Ruskin is right, prove it.
4. Converging to the minimum-norm solution.

a) We have a discrete-time linear dynamical system

\[ x(t + 1) = Ax(t) + b \quad x(0) = x_0 \]

where \( A \in \mathbb{R}^{n \times n} \) and \( b \in \mathbb{R}^n \). Recall the spectral radius of a square matrix \( A \) is

\[ \rho(A) = \max_i |\lambda_i(A)| \]

Suppose \( \rho(A) < 1 \). Find an expression for \( \lim_{t \to \infty} x(t) \) in terms of \( A \), \( b \), and \( x_0 \).

b) There are some systems with \( \rho(A) = 1 \) for which \( \lim_{t \to \infty} x(t) \) converges. Suppose \( T \in \mathbb{R}^{n \times n} \) is invertible, and

\[ A = T \begin{bmatrix} \hat{A} & 0 \\ 0 & I \end{bmatrix} T^{-1} \]

where \( \hat{A} \in \mathbb{R}^{r \times r} \) and \( \rho(\hat{A}) < 1 \). Give conditions on \( T, \hat{A}, b, x_0 \) such that \( \lim_{t \to \infty} x(t) \) converges.

c) Under your conditions of the previous part, what is the limit \( \lim_{t \to \infty} x(t) \)

d) One system for which this condition is useful is as follows. We have a matrix \( B \in \mathbb{R}^{m \times n} \).

We emphasize here that \( B \) is not known to be skinny or full rank, so the usual formula for the least-squares solution \( (B^T B)^{-1} B^T y \) does not make sense. We would nonetheless like to solve the least-squares problem

\[ \min_{x \in \mathbb{R}^n} \|Bx - y\| \]

Suppose \( B \) has full singular value decomposition \( B = U \Sigma V^T \). What are all the optimal solutions for this problem, in terms of the SVD?

e) When \( B \) is very large, one commonly used method to solve this problem is the gradient method. Define the function

\[ f(x) = \frac{1}{2} \|Bx - y\|^2 \]

and consider the gradient update rule

\[ x(t + 1) = x(t) - h \nabla f(x(t)) \]

where \( h > 0 \) is a (small) step size. The update for the gradient method is a discrete-time linear dynamical system of the form

\[ x(t + 1) = Ax(t) + b \]

Find \( A \) and \( b \) in terms of \( h \), \( B \) and \( y \).
f) Show that, if $x(0) = 0$ and $h$ is sufficiently small, then
\[
\lim_{t \to \infty} x(t) = B^t y
\]

The importance of this result is that it shows that the gradient algorithm for this problem converges to the minimum-norm solution.

5. **Filling-in missing data.** In this problem we have a signal, $y_i \in \mathbb{R}$ for $i = 1, \ldots, n$, which we view as $y \in \mathbb{R}^n$. We will have $n = 100$. The signal $y$ comes from measurements of a physical system, and so $y_{i+1}$ is measured a short time interval after $y_i$. Unfortunately, during the data acquisition process some of the data was lost and so the signal we have has gaps in it. Specifically, we have a known set $K \subset \mathbb{Z}$ and we know $y_i$ only for values $i \in K$.

The data for this problem is in the file `missing.json`. The supplied vector `known` contains the list of known points $K$, and the vector `yknown` is the list of values of $y$ at the points in $K$. The length of `yknown` is therefore $|K|$.

a) For a signal $z \in \mathbb{R}^n$, we define the discrete derivative $z_{\text{der}} \in \mathbb{R}^{n-1}$ by
\[
z_{\text{der}}^i = z^i_{i+1} - z^i_i \quad \text{for} \quad i = 1, \ldots, n - 1
\]
Find the matrix $G$ such that $z_{\text{der}} = Gz$

b) Our first approach will be to find the signal $z$ which minimizes $\|z_{\text{der}}\|$ and satisfies
\[
z_i = y_i \quad \text{if} \quad i \in K
\]
Give a method finding the optimal $z$.

c) Find the optimal $z$ in the previous part and plot $z_i$ against $i$. Be sure to plot the points $(i, z_i)$, not just a line joining them.

d) One way to do a better job at filling in the missing data is to put additional criteria on our estimate. Here we will do this by additionally penalizing the second derivative of $z$. Define the discrete second derivative $z_{\text{hes}} \in \mathbb{R}^{n-2}$ by
\[
z_{\text{hes}}^i = z^i_{i+2} - 2z^i_{i+1} + z^i_i \quad \text{for} \quad i = 1, \ldots, n - 2
\]
Find the matrix $H$ such that $z_{\text{hes}} = Hz$

e) Define the two objective functions
\[
J_1 = \|Gz\|^2 \quad J_2 = \|Hz\|^2
\]
We would like to find the signal $z$ that minimizes
\[
J_1 + \mu J_2
\]
and satisfies
\[
z_i = y_i \quad \text{if} \quad i \in K
\]
Give a method for finding the optimal $z$.

f) Plot the trade-off curve of $J_2$ (on the vertical axis) versus $J_1$ (on the horizontal axis). Give the interpretation of the endpoints of this curve.

g) Find the optimal $z$ for the three different cases $\mu = 5, 20, 100$. 

6
6. Some true/false questions. For each of the following statements, if it is true give a proof, and if it is false give a counterexample.

a) Suppose \( A = \begin{bmatrix} B & C \\ 0 & D \end{bmatrix} \) where \( B, C, D \) are matrices of compatible dimension. Then \( s \) is a singular value of \( A \) if and only if it is a singular value of \( B \) or \( D \).

b) Suppose \( A, B \in \mathbb{R}^{n \times n} \). There exists \( u \in \mathbb{R}^n \) such that
\[
A_{ij} = B_{ij} + u_i - u_j \quad \text{for all } i, j
\]
if and only if \( A + A^T = B + B^T \) (that is, \( A \) and \( B \) have the same symmetric part.)

c) Suppose \( Q \) is nonzero and \( Q + Q^T = 0 \). Then \( \text{rank}(Q) \geq 2 \).

d) Suppose \( A \in \mathbb{R}^{n \times n} \). Then \( ||A|| < 1 \) if and only if \( |\lambda_i(A)| < 1 \) for all \( i \).

e) Suppose \( A \in \mathbb{R}^{n \times n} \) and \( A \neq 0 \). Then \( A \) has at least one nonzero eigenvalue.

f) Suppose \( ||x|| = 1 \) and \( A = I - xx^T \). Then all eigenvalues of \( A \) are 0 or 1.

g) Suppose \( A \in \mathbb{R}^{m \times n} \) and \( ||A|| < 1 \). Then \( ||AB|| < ||B|| \) for all matrices \( B \).

h) Suppose \( A \) is not full rank. Then \( A^\dagger \) is neither a left-inverse nor a right-inverse of \( A \).

i) Suppose \( A \) is a matrix, and \( ABA = A \). Then \( \text{range}(I - BA) = \text{null}(A) \).