1. **Optimal operation of a two-state chemical reactor.** Consider a chemical reactor containing \( n \) compounds, labeled \( 1,\ldots,n \). Let \( x_i(t) \) be the amount of compound \( i \) in the reactor at time \( t \). The chemical reactor has two modes of operation, labeled 1 and 2. (For example, the first mode may be operating the reactor at a low temperature, and the second mode may be operating the reactor at a high temperature.) For simplicity we assume that the mode of operation can be changed instantaneously. When we operate the reactor in mode \( j \), the vector of compound amounts evolves according to the equation

\[
\dot{x}(t) = A_j x(t).
\]

We are given the vector \( x(0) \in \mathbb{R}^n \) of initial compound amounts, and the dynamics matrices \( A_1 \) and \( A_2 \). Our objective is to maximize the amount of compound \( k \) at time \( T \), where \( k \in \{1,\ldots,n\} \) and \( T > 0 \) are given.

a) Suppose the reactor operates in mode 1 for \( 0 \leq t \leq T_0 \), and mode 2 for \( T_0 < t \leq T \). Explain how to choose the time \( T_0 \) in order to maximize the amount of compound \( k \) at time \( T \). Your answer only needs to be accurate to two decimal digits.

b) Apply your method to the data given in `chemical_reactor_data.m`. Report the optimal value of \( T_0 \) and the corresponding amount of compound \( k \) at time \( T \); submit a plot showing all of the components of \( x(t) \) as functions of time on a single set of axes.

c) Suppose the reactor operates in mode 1 for \( 0 \leq t \leq T_1 \) and \( T_2 < t \leq T \), and mode 2 for \( T_1 < t \leq T_2 \). Explain how to choose the times \( T_1 \) and \( T_2 \) in order to maximize the amount of compound \( k \) at time \( T \). Your answers for \( T_1 \) and \( T_2 \) only need to be accurate to two decimal digits.

d) Apply your method to the data given in `chemical_reactor_data.m`. Report the optimal values of \( T_1 \) and \( T_2 \) and the corresponding amount of compound \( k \) at time \( T \); submit a plot showing all of the components of \( x(t) \) as functions of time on a single set of axes.

**Solution.**

a) Since the reactor operates in mode 1 for \( 0 \leq t \leq T_0 \), we have that

\[
x(T_0) = \exp(T_0 A_1) x(0).
\]

Similarly, because the reactor operates in mode 2 for \( T_0 < t \leq T \), we have that

\[
x(T) = \exp((T - T_0) A_2) x(T_0) = \exp((T - T_0) A_2) \exp(T_0 A_1) x(0).
\]
We want to maximize the amount of compound \( k \) at time \( T \):

\[
x_k(T) = e_k^T x(T) = e_k^T \exp((T - T_0)A_2) \exp(T_0A_1)x(0).
\]

Since our answer for \( T_0 \) only needs to be accurate to two decimal digits, we simply use the formula above to compute \( x_k(T) \) for all \( T_0 \in \{0.00, 0.01, 0.02, \ldots, T\} \), and then choose the value that maximizes \( x_k(T) \).

b) We find that the optimal time to change the mode of operation is \( T_0 = 0.61 \), and the corresponding amount of compound \( k \) at time \( T \) is 0.3655. A plot showing all of the components of \( x(t) \) as functions of time is given in ??.

c) Since the reactor operates in mode 1 for \( 0 \leq t \leq T_1 \), we have that

\[
x(T_1) = \exp(T_1A_1)x(0).
\]

Similarly, because the reactor operates in mode 2 for \( T_1 < t \leq T_2 \), we have that

\[
x(T_2) = \exp((T_2 - T_1)A_2)x(T_1) = \exp((T_2 - T_1)A_2) \exp(T_1A_1)x(0).
\]

Finally, since the reactor operates in mode 1 for \( T_2 < t \leq T \), we have that

\[
x(T) = \exp((T - T_2)A_1)x(T_2) = \exp((T - T_2)A_1) \exp((T_2 - T_1)A_2) \exp(T_1A_1)x(0).
\]

We want to maximize

\[
x_k(T) = e_k^T x(T) = e_k^T \exp((T - T_2)A_1) \exp((T_2 - T_1)A_2) \exp(T_1A_1)x(0).
\]

Since our answers for \( T_1 \) and \( T_2 \) only need to be accurate to two decimal digits, we simply use the formula above to compute \( x_k(T) \) for all \( T_1, T_2 \in \{0.00, 0.01, 0.02, \ldots, T\} \) such that \( T_1 < T_2 \), and then choose the values that maximize \( x_k(T) \).

d) We find that the optimal times to change the mode of operation are \( T_1 = 0.49 \) and \( T_2 = 0.87 \), and the corresponding amount of compound \( k \) at time \( T \) is 0.3763. A plot showing all of the components of \( x(t) \) as functions of time is given in ??.
xk(i) = sparse(1,k,1,1,n) ... 
    * expm((T - t(i)) * A2) ... 
    * expm(t(i) * A1) * x0;
end
[~, idx] = max(xk);
T0_opt = t(idx)
xk_opt = xk(idx)

% simulate the system using the optimal value of T0
tplot = 0.00:0.01:T;
dt = tplot(2) - tplot(1);
xplot = [x0 nan(n,length(tplot)-1)];
for i = 1:(length(tplot)-1)
    if tplot(i) <= T0_opt
        xplot(:,i+1) = expm(dt * A1) * xplot(:,i);
    else
        xplot(:,i+1) = expm(dt * A2) * xplot(:,i);
    end
end
figure();
plot(t, xplot);
xlabel('t');
ylabel('x(t)');
legend('x1(t)', 'x2(t)', 'x3(t)');
grid on;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% find the optimal times to change the mode of the system
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
xk = -Inf(length(t), length(t));
for i = 1:(length(t)-1)
    for j = (i+1):length(t)
        xk(i,j) = sparse(1,k,1,1,n) ... 
            * expm((T - t(j)) * A1) ... 
            * expm((t(j) - t(i)) * A2) ... 
            * expm(t(i) * A1) * x0;
    end
end
[~, idx] = max(xk(:));
[iopt, jopt] = ind2sub(size(xk), idx);
T1_opt = t(iopt)
T2_opt = t(jopt)
xk_opt = xk(idx)
Figure 1: the trajectory of $x(t)$ with a single optimal mode change

% simulate the system using the optimal values of T1 and T2
xplot = [x0 nan(n,length(tplot)-1)];
for i = 1:(length(tplot)-1)
    if tplot(i) <= T1_opt || tplot(i) > T2_opt
        xplot(:,i+1) = expm(dt * A1) * xplot(:,i);
    else
        xplot(:,i+1) = expm(dt * A2) * xplot(:,i);
    end
end

figure();
plot(t, xplot);
xlabel('t');
ylabel('x(t)');
legend('x1(t)', 'x2(t)', 'x3(t)');
grid on;
Figure 2: the trajectory of $x(t)$ with two optimal mode changes
2. **Harmonic oscillator.** The system \( \dot{x} = \begin{bmatrix} 0 & \omega \\ -\omega & 0 \end{bmatrix} x \) is called a harmonic oscillator.

a) Find the eigenvalues, resolvent, and state transition matrix for the harmonic oscillator. Express \( x(t) \) in terms of \( x(0) \).

b) Sketch the vector field of the harmonic oscillator.

c) The state trajectories describe circular orbits, i.e., \( \| x(t) \| \) is constant. Verify this fact using the solution from part (a).

d) You may remember that circular motion (in a plane) is characterized by the velocity vector being orthogonal to the position vector. Verify that this holds for any trajectory of the harmonic oscillator. Use only the differential equation; do not use the explicit solution you found in part (a).

**Solution.**

a) We have

\[
(sI - A)^{-1} = \frac{1}{s^2 + \omega^2} \begin{bmatrix} s & \omega \\ -\omega & s \end{bmatrix}.
\]

From this result it follows that the eigenvalues of \( A \) are given by \( \{ \pm j\omega \} \). The inverse Laplace transform gives

\[
\Phi(t) = \begin{bmatrix} \cos \omega t & \sin \omega t \\ -\sin \omega t & \cos \omega t \end{bmatrix}
\]

and we have \( x(t) = \Phi(t)x(0) \).

b) Here is the vector field:

![Figure 3: Vector field of harmonic oscillator](image-url)
c) First we note from basic trigonometric relations that $\Phi^T(t)\Phi(t) = I$. From this we conclude that $\Phi(t)$ is orthogonal. Now it follows that $x^T(t)x(t) = x^T(0)\Phi^T(t)\Phi(t)x(0) = x^T(0)x(0)$, i.e. $\|x(t)\| = \|x(0)\|$. 

d) Using previous relations we can write 

$$
\dot{x}^T x = x^T \begin{bmatrix} 0 & -\omega \\ \omega & 0 \end{bmatrix} x = \begin{bmatrix} -\omega x_2 \\ \omega x_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0
$$

This shows that the velocity vector is always orthogonal to the position vector, as claimed.

3. Interconnection of linear systems. Often a linear system is described in terms of a block diagram showing the interconnections between components or subsystems, which are themselves linear systems. In this problem you consider the specific interconnection shown below:

![Block Diagram](image)

Here, there are two subsystems $S$ and $T$. Subsystem $S$ is characterized by

$$
\dot{x} = Ax + B_1 u + B_2 w_1, \quad w_2 = Cx + D_1 u + D_2 w_1,
$$

and subsystem $T$ is characterized by

$$
\dot{z} = Fz + G_1 v + G_2 w_2, \quad w_1 = H_1 z, \quad y = H_2 z + J w_2.
$$

We don’t specify the dimensions of the signals (which can be vectors) or matrices here. You can assume all the matrices are the correct (i.e., compatible) dimensions. Note that the subscripts in the matrices above, as in $B_1$ and $B_2$, refer to different matrices. Now the problem. Express the overall system as a single linear dynamical system with input, state, and output given by

$$
\begin{bmatrix} u \\ v \end{bmatrix}, \quad \begin{bmatrix} x \\ z \end{bmatrix}, \quad y,
$$

respectively. Be sure to explicitly give the input, dynamics, output, and feedthrough matrices of the overall system. If you need to make any assumptions about the rank or invertibility of any matrix you encounter in your derivations, go ahead. But be sure to let us know what assumptions you are making.
Solution. This one is easier than it might appear. All we need to do is write down all the equations for this system, and massage them to be in the form of a linear dynamical system with the given input, state, and output. Substituting the expression for $w_1$ into the first set of equations gives

$$
\dot{x} = Ax + B_1u + B_2H_1z, \quad w_2 = Cx + D_1u + D_2H_1z.
$$

Similarly, substituting the expression for $w_2$ into the second set of equations yields

$$
\dot{z} = Fz + G_1v + G_2(Cx + D_1u + D_2H_1z), \quad y = H_2z + J(Cx + D_1u + D_2H_1z).
$$

Now we just put this together into the required form:

$$
\begin{bmatrix}
\dot{x} \\
\dot{z}
\end{bmatrix} =
\begin{bmatrix}
A & B_2H_1 \\
G_2C & F + G_2D_2H_1
\end{bmatrix}
\begin{bmatrix}
x \\
z
\end{bmatrix} +
\begin{bmatrix}
B_1 & 0 \\
G_2D_1 & G_1
\end{bmatrix}
\begin{bmatrix}
u \\
v
\end{bmatrix}
$$

and

$$
y = [JC \quad H_2 + JD_2H_1]
\begin{bmatrix}
x \\
z
\end{bmatrix} + [JD_1 \quad 0]
\begin{bmatrix}
u \\
v
\end{bmatrix}
$$

4. Analysis of investment allocation strategies. Each year or period (denoted $t = 0, 1, \ldots$) an investor buys certain amounts of one-, two-, and three-year certificates of deposit (CDs) with interest rates 5%, 6%, and 7%, respectively. (We ignore minimum purchase requirements, and assume they can be bought in any amount.)

- $B_1(t)$ denotes the amount of one-year CDs bought at period $t$.
- $B_2(t)$ denotes the amount of two-year CDs bought at period $t$.
- $B_3(t)$ denotes the amount of three-year CDs bought at period $t$.

We assume that $B_1(0) + B_2(0) + B_3(0) = 1$, i.e., a total of 1 is to be invested at $t = 0$. (You can take $B_j(t)$ to be zero for $t < 0$.) The total payout to the investor, $p(t)$, at period $t$ is a sum of six terms:

- $1.05B_1(t-1)$, i.e., principle plus 5% interest on the amount of one-year CDs bought one year ago.
- $1.06B_2(t-2)$, i.e., principle plus 6% interest on the amount of two-year CDs bought two years ago.
- $1.07B_3(t-3)$, i.e., principle plus 7% interest on the amount of three-year CDs bought three years ago.
- $0.06B_2(t-1)$, i.e., 6% interest on the amount of two-year CDs bought one year ago.
- $0.07B_3(t-1)$, i.e., 7% interest on the amount of three-year CDs bought one year ago.
- $0.07B_3(t-2)$, i.e., 7% interest on the amount of three-year CDs bought two years ago.
The total wealth held by the investor at period \( t \) is given by
\[
w(t) = B_1(t) + B_2(t) + B_2(t-1) + B_3(t) + B_3(t-1) + B_3(t-2).
\]
Two re-investment allocation strategies are suggested.

- **The 35-35-30 strategy.** The total payout is re-invested 35% in one-year CDs, 35% in two-year CDs, and 30% in three-year CDs. The initial investment allocation is the same: \( B_1(0) = 0.35 \), \( B_2(0) = 0.35 \), and \( B_3(0) = 0.30 \).

- **The 60-20-20 strategy.** The total payout is re-invested 60% in one-year CDs, 20% in two-year CDs, and 20% in three-year CDs. The initial investment allocation is \( B_1(0) = 0.60 \), \( B_2(0) = 0.20 \), and \( B_3(0) = 0.20 \).

a) Describe the investments over time as a linear dynamical system
\[
x(t+1) = Ax(t),
\]
\[
y(t) = Cx(t)
\]
with \( y(t) \) equal to the total wealth at time \( t \). Be very clear about what the state \( x(t) \) is, and what the matrices \( A \) and \( C \) are. You will have two such linear systems: one for the 35-35-30 strategy and one for the 60-20-20 strategy.

b) **Asymptotic wealth growth rate.** For each of the two strategies described above, determine the asymptotic growth rate, defined as \( \lim_{t \to \infty} w(t+1)/w(t) \). (If this limit doesn’t exist, say so.) Note: simple numerical simulation of the strategies (e.g., plotting \( w(t+1)/w(t) \) versus \( t \) to guess its limit) is not acceptable. (You can, of course, simulate the strategies to check your answer.)

c) **Asymptotic liquidity.** The total wealth at time \( t \) can be divided into three parts:

- \( B_1(t) + B_2(t-1) + B_3(t-2) \) is the amount that matures in one year (i.e., the amount of principle we could get back next year)
- \( B_2(t) + B_3(t-1) \) is the amount that matures in two years
- \( B_3(t) \) is the amount that matures in three years (i.e., is least liquid)

We define liquidity ratios as the ratio of these amounts to the total wealth:
\[
L_1(t) = (B_1(t) + B_2(t-1) + B_3(t-2))/w(t),
\]
\[
L_2(t) = (B_2(t) + B_3(t-1))/w(t),
\]
\[
L_3(t) = B_3(t)/w(t).
\]

For the two strategies above, do the liquidity ratios converge as \( t \to \infty \)? If so, to what values? Note: as above, simple numerical simulation alone is not acceptable.

d) Suppose you could change the initial investment allocation for the 35-35-30 strategy, i.e., choose some other nonnegative values for \( B_1(0), B_2(0), \) and \( B_3(0) \) that satisfy \( B_1(0) + B_2(0) + B_3(0) = 1 \). What allocation would you pick, and how would it be better than the (0.35, 0.35, 0.30) initial allocation? (For example, would the asymptotic growth rate be larger?) How much better is your choice of initial investment allocations? Hint for part d: think very carefully about this one. Hint for whole problem: watch out for nondiagonalizable, or nearly nondiagonalizable, matrices. Don’t just blindly type in matlab commands; check to make sure you’re computing what you think you’re computing.
Solution.

a) We take as state vector
\[
x(t) = \begin{bmatrix}
B_1(t) \\
B_2(t) \\
B_2(t-1) \\
B_3(t) \\
B_3(t-1) \\
B_3(t-2)
\end{bmatrix}.
\]

The components consist of the six critical quantities: the amount of one-, two-, and three-year CDs held of each possible maturity date (i.e., one, two, and three years). We can express the system as
\[
x(t+1) = Fx(t) + Gu(t)
\]
where \(u(t) \in \mathbb{R}^3\) gives the amount of one, two, and three year CDs purchased at \(t + 1\), and
\[
F = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0
\end{bmatrix}, \quad G = \begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 0 \\
0 & 0 & 1 \\
0 & 0 & 0 \\
0 & 0 & 0
\end{bmatrix}.
\]

The payout is given by \(p(t) = hx(t)\), where
\[
h = [1.05 \ 0.06 \ 1.06 \ 0.07 \ 0.07 \ 1.07].
\]

The allocation of the payout is given by \(u(t) = [0.35 \ 0.35 \ 0.30]^T p(t)\) for the 35-35-30 strategy and by \(u(t) = [0.60 \ 0.20 \ 0.20]^T p(t)\) for the 60-20-20 strategy. Finally, putting it all together, we end up with
\[
x(t+1) = Ax(t)
\]
for the 35-35-30 case, and similarly for the 60-20-20 case. To be fully explicit, we have
\[
A_{35} = F + G[0.35 \ 0.35 \ 0.30]^T h
\]
for the 35-35-30 case, and
\[
A_{60} = F + G[0.60 \ 0.20 \ 0.20]^T h
\]
for the 60-20-20 case. The following matlab code calculates the matrix \(A\) for each allocation strategy.

\[
AA = [0 \ 0 \ 0 \ 0 \ 0 \ 0];
\]
BB = [1 0 0; 0 1 0; 0 0 0; 0 0 1; 0 0 0; 0 0 0];
PP = [1.05 0.06 1.06 0.07 0.07 1.07]; % PP*x gives payout
alloc35 = [.35 .35 .3]'; % for 35-35-30 allocation
A35 = AA+BB*alloc35*PP;
alloc60 = [0.60 0.20 0.20]'; % for 60-20-20 allocation
A60 = AA+BB*alloc60*PP;
WW=ones(1,6); %WW*x gives total wealth

This yields:

A35 =
0.3675 0.0210 0.3710 0.0245 0.0245 0.3745
0.3675 0.0210 0.3710 0.0245 0.0245 0.3745
0 1.0000 0 0 0 0
0.3150 0.0180 0.3180 0.0210 0.0210 0.3210
0 0 0 1.0000 0 0
0 0 0 0 1.0000 0

A60 =
0.6300 0.0360 0.6360 0.0420 0.0420 0.6420
0.2100 0.0120 0.2120 0.0140 0.0140 0.2140
0 1.0000 0 0 0 0
0.2100 0.0120 0.2120 0.0140 0.0140 0.2140
0 0 0 1.0000 0 0
0 0 0 0 1.0000 0

There are several other correct answers. For example, some people decided to use a state vector that included the past three samples of each of the B_i's, i.e., a state of dimension 9. Provided no errors were made, this works fine. At the other end, some people found a state description that has a state dimension of 4.

b) Asymptotic wealth growth rate. This is going to depend on the eigenvalues of A, so first we check the eigenvalues, which turn out to be

1.0627, -0.3266 ± 0.4421i, 0, 0, 0

for A35, and

1.0598, -0.2019 ± 0.4015i, 0, 0, 0
for $A_{60}$. Each matrix has one real, positive eigenvalue that has larger magnitude than one, and than the other eigenvalues, which in fact all have magnitude less than one. Therefore, as $t \to \infty$,

$$A^t \to \lambda^t v w^T,$$

where $\lambda$ is the dominant eigenvalue, and $v$ and $w$ are the right and left eigenvectors associated with $\lambda$, normalized so that $w^T v = 1$. There is a subtlety in computing $v$ and $w$, since the matrices have nontrivial Jordan form. Forming a matrix of right eigenvectors and inverting won’t work because there is no set of independent eigenvectors! But finding the eigenvector of $A^T$ associated with the dominant eigenvector, and then normalizing properly, does work. It follows that as $t \to \infty$ we have

$$x(t) \to \lambda^t v w^T x(0).$$

A quick check shows that $w^T x(0) \neq 0$ (which means the initial condition does excite the dominant mode), so as $t \to \infty$, $x(t)$ grows exponentially with rate $\lambda$. The wealth satisfies

$$w(t) \to \lambda^t (1^T v)(w^T x(0))$$

as $t \to \infty$. Hence $w(t+1)/w(t)$ converges to $\lambda$. For the 35-35-30 scheme, the growth rate is 1.0627, i.e., 6.27% per year. For the 60-20-20 scheme, the growth rate is 1.0598, i.e., 5.98% per year. It makes sense that the first scheme has a higher growth rate since a higher fraction is invested in higher-yield CDs. A few people made numerical errors and ended up with growth rates of, say, 45%. How can investments in a portfolio of 5%, 6%, and 7% CDs end up yielding 45%? We can only assume this problem was solved very late at night . . .

\begin{center}
\begin{tabular}{c|c}
\hline
$t$ (year) & Growth rate \\
\hline
0 & 1.0597 \\
2 & 1.0600 \\
4 & 1.0605 \\
6 & 1.0610 \\
8 & 1.0615 \\
10 & 1.0620 \\
12 & 1.0625 \\
14 & 1.0630 \\
\hline
\end{tabular}
\end{center}

c) Asymptotic liquidity. The analysis above shows that as $t \to \infty$, the state vector grows exponentially with rate given by $\lambda$, and asymptotic distribution shape given by $v$, the right eigenvector associated with the dominant eigenvalue $\lambda$. Define

$$l_1 = [1 0 1 0 0 1]^T, \quad l_2 = [0 1 0 0 1 0]^T, \quad l_3 = [0 0 0 1 0 0]^T,$$
so that \( L_i(t) = l_i x(t) / \mathbf{1}^T x(t) \). Since \( x(t) \to \lambda^T w^T x(0) \) we have \( L_i(t) \to \frac{i^T x}{\mathbf{1}^T x} \). Therefore the asymptotic values of \( L_1, L_2, \) and \( L_3 \) for the 35–35–30 strategy are 0.5034, 0.3368, and 0.1598, respectively. For the 60–20–20 strategy, they are 0.6215, 0.2499, and 0.1286, respectively. It makes sense that the second scheme is asymptotically more liquid, since a higher fraction is invested in fast-maturing CDs.

\[
\begin{align*}
\text{Liquidity ratios for 35-35-30 strategy} \\
\text{Liquidity ratios for 60-20-20 strategy}
\end{align*}
\]

\( d) \) When we change \( x(0) \) we affect the term \( w^T x(0) \) in the asymptotic state. Hence by changing the initial condition we do not change the asymptotic growth rate. (Technically
there is one stupid case where we can change it: if we could arrange for \( w^T x(0) = 0 \), then asymptotic growth rate would be determined by the second largest eigenvalue. In fact for this problem you can prove that \( w^T x(0) = 0 \) is impossible since the vector \( w \) has all positive components.) The asymptotic total wealth, which is \( \lambda_1(w^T x(0))(1^T v) \), depends linearly on \( w^T x(0) \); evidently we want to choose \( x(0) \) to maximize \( w^T x(0) \). Thus our problem is to maximize

\[
w^T [B_1(0) B_2(0) 0 B_3(0) 0 0] = w_1 B_1(0) + w_2 B_2(0) + w_4 B_3(0),
\]

subject to

\[B_1(0) + B_2(0) + B_3(0) = 1, \quad B_1(0) \geq 0, \quad B_2(0) \geq 0, \quad B_3(0) \geq 0.\]

The solution is clear if you think about it: find the largest of \( w_1, w_2, \) and \( w_4 \) and allocate all the initial investment in the corresponding \( B_i \). We can obtain the dominant left eigenvector \( w \) in matlab, by finding the dominant right eigenvector of \( A^T \):

\[
\text{[W35,Lambda35]=eig(A35'); } \quad \text{to find the left eigenvectors}
\]
\[
w_1=W35(:,1) \quad \text{the dominant left eigenvector}
\]
\[
w_1=w_1/(w_1'*v1_35) \quad \text{normalize w1 so that } w_1'*v=1
\]

ans =
0.4045
0.4074
0.4084
0.4174
0.4149
0.4122

The optimal initial allocation will be \( B_1(0) = 0, B_2(0) = 0, \) and \( B_3(0) = 1 \). It’s just a bit better than the original scheme, since with original scheme \( w^T x(0) = 0.4094 \), whereas with our new optimal initial investment allocation we have \( w^T x(0) = 0.4174 \) (only 1.95% better). Plots below show the total wealth for the original allocation (solid line) and the optimal allocation (dashed line).
Several people guessed this solution, but didn’t justify it. Others attempted SVD methods, or attempted to set up the initial condition along \( v \), maybe by projecting \( v \) onto the first, second, and fourth components. None of these methods is correct.

5. Some basic properties of eigenvalues. Show the following:

   a) The eigenvalues of \( A \) and \( A^T \) are the same.

   b) \( A \) is invertible if and only if \( A \) does not have a zero eigenvalue.

   c) If the eigenvalues of \( A \) are \( \lambda_1, \ldots, \lambda_n \) and \( A \) is invertible, then the eigenvalues of \( A^{-1} \) are \( 1/\lambda_1, \ldots, 1/\lambda_n \).

   d) The eigenvalues of \( A \) and \( T^{-1}AT \) are the same.

  \( \text{Hint: you’ll need to use the facts that } \det A = \det(A^T), \det(AB) = \det A \det B, \text{ and, if } A \text{ is invertible, } \det(A^{-1}) = 1/\det A. \)

Solution.

   a) The eigenvalues of a matrix \( A \) are given by the roots of the polynomial \( \det(sI - A) \).

   From determinant properties we know that \( \det(sI - A) = \det(sI - A)^T = \det(sI - AT) \).

   We conclude that the eigenvalues of \( A \) and \( A^T \) are the same.

   b) First we recall that \( A \) is invertible if and only if \( \det(A) \neq 0 \). But \( \det(A) \neq 0 \iff \det(-A) \neq 0 \).

   i. If 0 is an eigenvalue of \( A \), then \( \det(sI - A) = 0 \) when \( s = 0 \). It follows that \( \det(-A) = 0 \) and thus \( \det(A) = 0 \), and \( A \) is not invertible. From this fact we conclude that if \( A \) is invertible, then 0 is not an eigenvalue of \( A \).

   ii. If \( A \) is not invertible, then \( \det(A) = \det(-A) = 0 \). This means that, for \( s = 0 \), \( \det(sI - A) = 0 \), and we conclude that in this case 0 must be an eigenvalue of \( A \). From this fact it follows that if 0 is not an eigenvalue of \( A \), then \( A \) is invertible.

   c) From the results of the last item we see that 0 is not an eigenvalue of \( A \). Now consider the eigenvalue/eigenvector pair \( \lambda_i, x_i \) of \( A \). This pair satisfies \( Ax_i = \lambda_i x_i \). Now, since \( A \) is invertible, \( \lambda_i \) is invertible. Multiplying both sides by \( A^{-1} \) and \( \lambda_i^{-1} \) we have \( \lambda_i^{-1} x_i = A^{-1} x_i \), and from this we conclude that the eigenvalues of the inverse are the inverse of the eigenvalues.

   d) First we note that \( \det(sI - A) = \det(I(sI - A)) = \det(T^{-1}T(sI - A)) \). Now, from determinant properties, we have \( \det(T^{-1}T(sI - A)) = \det(T^{-1}(sI - A)T) \). But this is also equal to \( \det(sI - T^{-1}AT) \), and the conclusion is that the eigenvalues of \( A \) and \( T^{-1}AT \) are the same.
6. **Optimal espresso cup pre-heating.** At time \( t = 0 \) boiling water, at 100°C, is poured into an espresso cup; after \( P \) seconds (the ‘pre-heating time’), the water is poured out, and espresso, with initial temperature 95°C, is poured in. (You can assume this operation occurs instantaneously.) The espresso is then consumed exactly 15 seconds later (yes, instantaneously). The problem is to choose the pre-heating time \( P \) so as to maximize the temperature of the espresso when it is consumed.

We now give the thermal model used. We take the temperature of the liquid in the cup (water or espresso) as one state; for the cup we use an \( n \)-state finite element model. The vector \( x(t) \in \mathbb{R}^{n+1} \) gives the temperature distribution at time \( t \): \( x_1(t) \) is the liquid (water or espresso) temperature at time \( t \), and \( x_2(t), \ldots, x_{n+1}(t) \) are the temperatures of the elements in the cup. All of these are in degrees C, with \( t \) in seconds. The dynamics are

\[
\frac{d}{dt}(x(t) - 20 \cdot 1) = A(x(t) - 20 \cdot 1),
\]

where \( A \in \mathbb{R}^{(n+1) \times (n+1)} \). (The vector \( 20 \cdot 1 \), with all components 20, represents the ambient temperature.) The initial temperature distribution is

\[
x(0) = \begin{bmatrix} 100 \\ 20 \\ \vdots \\ 20 \end{bmatrix}.
\]

At \( t = P \), the liquid temperature changes instantly from whatever value it has, to 95; the other states do not change. Note that the dynamics of the system are the same before and after pre-heating (because we assume that water and espresso behave in the same way, thermally speaking).

We have very generously derived the matrix \( A \) for you. You will find it in `espressodata.m`. In addition to \( A \), the file also defines \( n \), and, respectively, the ambient, espresso and preheat water temperatures \( T_a \) (which is 20), \( T_e \) (95), and \( T_l \) (100).

Explain your method, submit your code, and give final answers, which must include the optimal value of \( P \) and the resulting optimal espresso temperature when it is consumed. Give both to an accuracy of one decimal place, as in

\[ P = 23.5 \text{ s, which gives an espresso temperature at consumption of 62.3°C.} \]

(This is not the correct answer, of course.)

**Solution.** After \( P \) seconds of pre-heating, we will have

\[ x(P) - 20 \cdot 1 = e^{PA}(x(0) - 20 \cdot 1). \]

Define a new vector \( \tilde{x}(P) \) with \( \tilde{x}_i(P) = x_i(P) \) for \( i = 2, \ldots, n+1 \), and \( \tilde{x}_1(P) = 95 \). (Thus, \( \tilde{x}(P) \) is the state immediately after the water is replaced with espresso.) The temperature distribution at time \( P + 15 \) will be

\[ x(P + 15) - 20 \cdot 1 = e^{15A}(\tilde{x}(P) - 20 \cdot 1). \]
We now have a method for calculating the temperature of the espresso at the instant of consumption for a given $P$:

$$T(P) - 20 = e_1^T x(P + 15) = e_1^T e^{15A} (\tilde{x}(P) - 20 \cdot 1),$$

where $e_1$ is the first unit vector. Thus, we have

$$T(P) = e_1^T e^{15A} (\tilde{x}(P) - 20 \cdot 1) + 20.$$

To find the optimal value of $P$ we use a simple search method, by calculating $T(P)$ over a finely-sampled range of values of $P$, and selecting the maximum value.

The optimal preheating time for this example is 11.1 seconds. This will give an espresso temperature of 87.6°C.

Matlab code to calculate the answers appears below.

```matlab
% load data.
espressodata;

% Test a range of preheating times up to a minute.
Tphs = linspace(0, 60, 1000);
% Condition at instant when preheating liquid is added.
% Note change of coordinates by subtracting Ta (and elsewhere).
p0 = [Tl; Ta*ones(n,1)] - Ta;

y = zeros(size(Tphs));
for i = 1:length(Tphs)
    Tph = Tphs(i);
    % Find state after preheating by propagating forward.
    xph = expm(Tph*A)*p0;
    % Instantaneously add espresso, changing only the liquid portion of the
    % state.
    xph(1) = Te - Ta;
    % Record temperature at time 15.
    z = expm(15*A)*xph;
    y(i) = z(1);
end

[Tmax, i] = max(y+Ta);
```
The graph below shows how preheat time affects the drinking temperature.

The next graph shows the temperature of the espresso over a 5 minute period, with and without preheating.

7. **Real modal form.** Generate a matrix $A$ in $\mathbb{R}^{10 \times 10}$ using $A = \text{randn}(10)$. (The entries of $A$ will be drawn from a unit normal distribution.) Find the eigenvalues of $A$. If by chance they are all real, please generate a new instance of $A$. Find the real modal form of $A$, i.e., a matrix $S$ such that $S^{-1}AS$ has the real modal form given in lecture 11. Your solution should include a clear explanation of how you will find $S$, the source code that you use to find $S$, and some code that checks the results (i.e., computes $S^{-1}AS$ to verify it has the required form).
**Solution.** Assuming $A$ is diagonalizable, it can be written as $A = T\Lambda T^{-1}$, Here

$$
\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_r, \lambda_{r+1}, \ldots, \lambda_n),
$$

where $\lambda_1, \ldots, \lambda_r$ are the real eigenvalues of $A$ and $\lambda_{r+1}, \ldots, \lambda_n$ are the complex eigenvalues of $A$ and come in complex conjugate pairs. Let $t_i$ be the $i$th column of $T$. Take $S$ to be

$$
S = [t_1 \cdots t_r \ \Re(t_{r+1}) \ \Im(t_{r+1}) \ \cdots \ \Re(t_{n-1}) \ \Im(t_{n-1})].
$$

Let us now prove why constructing $S$ in this way will give us the desired result. Let $v = \Re(v) + i\Im(v)$ be a complex eigenvector of $A$ associated with the eigenvalue $\lambda = \sigma + i\omega$. We must have $Av = \lambda v$, i.e.,

$$
A(\Re(v) + i\Im(v)) = (\sigma + i\omega)(\Re(v) + i\Im(v)).
$$

This implies that

$$
A\Re(v) = \sigma \Re(v) - \omega \Im(v), \quad A\Im(v) = \omega \Re(v) + \sigma \Im(v),
$$

or equivalently

$$
A [\Re(v) \ \Im(v)] = [\Re(v) \ \Im(v)] \begin{bmatrix} \sigma & \omega \\ -\omega & \sigma \end{bmatrix}.
$$

If we do the same derivation for the other complex conjugate eigenvalue pairs, we get the construction of $S$ that was presented above.

Here is a short MATLAB script that checks that our solution is correct.

```matlab
randn('seed', 21395);
A = randn(10);
[V, D] = eig(A);
% Take eigenvectors of complex eigenvalues and arrange them in pairs.
S = zeros(10);
S(:,1) = V(:,5);
S(:,2) = V(:,6);
S(:,3) = V(:,7);
S(:,4) = V(:,10);
S(:,5) = real(V(:,1));
S(:,6) = imag(V(:,1));
S(:,7) = real(V(:,3));
S(:,8) = imag(V(:,3));
S(:,9) = real(V(:,8));
S(:,10) = imag(V(:,8));

% Inspect $S^{-1}AS$.
inv(S)*A*S
```
8. Jordan form of a block matrix. We consider the block $2 \times 2$ matrix

$$C = \begin{bmatrix} A & I \\ 0 & A \end{bmatrix}.$$  

Here $A \in \mathbb{R}^{n \times n}$, and is diagonalizable, with real, distinct eigenvalues $\lambda_1, \ldots, \lambda_n$. We’ll let $v_1, \ldots, v_n$ denote (independent) eigenvectors of $A$ associated with $\lambda_1, \ldots, \lambda_n$.

a) Find the Jordan form $J$ of $C$. Be sure to explicitly describe its block sizes.

b) Find a matrix $T$ such that $J = T^{-1}CT$.

Solution. The eigenvalues of $C$ are

$$\lambda_1, \lambda_1, \lambda_2, \lambda_2, \ldots, \lambda_n, \lambda_n,$$

i.e., each of the eigenvalues of $A$, given twice. Each $\lambda_i$, then, is either associated with two $1 \times 1$ Jordan blocks in $C$, or with one $2 \times 2$ Jordan block in $C$. It turns out that only the second possibility can occur. In other words, the Jordan form of $C$ is block diagonal, with $n$ $2 \times 2$ Jordan blocks, each associated with one $\lambda_i$:

$$J = \begin{bmatrix} \lambda_1 & 1 \\ & \lambda_1 \\ & & \lambda_2 & 1 \\ & & & \lambda_2 \\ & & & & \ddots \\ & & & & & \lambda_n & 1 \\ & & & & & & \lambda_n \end{bmatrix}.$$

This can be discovered several ways, but in any case below we’ll give a matrix $T$ such $T^{-1}AT$ has this form, which ends the discussion. This Jordan form isn’t unexpected. After all, $C$ looks like a sort of $2 \times 2$ matrix Jordan block. Let’s move on to part (b). We need to find a set of independent generalized eigenvectors for $C$, i.e., the columns of $T$, which we’ll call $t_1, \ldots, t_{2n}$. They satisfy the equations

$$Ct_{2i-1} = \lambda_it_{2i-1}, \quad i = 1, \ldots, n,$$

which states that the odd index $t_i$’s are true eigenvectors of $C$, and the generalized eigenvector equations,

$$Ct_{2i} = t_{2i-1} + \lambda_it_{2i}, \quad i = 1, \ldots, n.$$

Let’s first start with the odd indices, which correspond to true eigenvectors of $C$. We first note that $Av_i = \lambda_iv_i$, so we have

$$\begin{bmatrix} A & B \\ 0 & A \end{bmatrix} \begin{bmatrix} v_i \\ 0 \end{bmatrix} = \lambda_i \begin{bmatrix} v_i \\ 0 \end{bmatrix}.$$

Thus, we can take

$$t_{2i-1} = \begin{bmatrix} v_i \\ 0 \end{bmatrix}, \quad i = 1, \ldots, n.$$
Let’s find the even index generalized eigenvectors, \( t_2, t_4, \ldots, t_{2n} \). They satisfy the equations
\[
Ct_{2i} = t_{2i-1} + \lambda_i t_{2i}, \quad i = 1, \ldots, n.
\]
To find what \( t_{2i} \) must be, we express it as \( t_{2i} = [u^T \ w^T]^T \):
\[
Ct_{2i} = \begin{bmatrix} A & I \\ 0 & A \end{bmatrix} \begin{bmatrix} u \\ w \end{bmatrix} = \begin{bmatrix} Au + w \\ Aw \end{bmatrix} = t_{2i-1} + \lambda_i t_{2i} = \begin{bmatrix} v_i + \lambda_i u \\ \lambda_i w \end{bmatrix}.
\]
From the bottom entry, we see that \( Aw = \lambda_i w \), so must take \( w = v_i \) (or some multiple of \( v_i \)).
Plugging this in to the top entry yields \( Au + v_i = v_i + \lambda_i u \), from which we conclude \( u = v_i \) as well. Thus, we can take as even generalized eigenvectors
\[
t_{2i} = \begin{bmatrix} v_i \\ v_i \end{bmatrix}, \quad i = 1, \ldots, n.
\]
We’re done! By our construction we have \( T^{-1}CT = J \), where the columns of \( T \) are given above. In particular, we have
\[
T = \begin{bmatrix} v_1 & v_1 & v_2 & \cdots & v_n & v_n \\ 0 & v_1 & 0 & \cdots & 0 & v_n \end{bmatrix}.
\]

9. **Affine dynamical systems.** A function \( f : \mathbb{R}^n \to \mathbb{R}^m \) is called affine if it is a linear function plus a constant, i.e., of the form \( f(x) = Ax + b \). Affine functions are more general than linear functions, which result when \( b = 0 \). We can generalize linear dynamical systems to affine dynamical systems, which have the form
\[
\dot{x} = Ax + Bu + f, \quad y = Cx + Du + g.
\]
Fortunately we don’t need a whole new theory for (or course on) affine systems; a simple shift of coordinates converts it to a linear dynamical system. Assuming \( A \) is invertible, define \( \tilde{x} = x + A^{-1}f \) and \( \tilde{y} = y - g + CA^{-1}f \). Show that \( \tilde{x} \), \( u \), and \( \tilde{y} \) are the state, input, and output of a linear dynamical system.

**Solution.** All we have to do is to show that \( \tilde{x} \), \( u \) and \( \tilde{y} \) satisfy a linear dynamical system. First note that
\[
\frac{d\tilde{x}}{dt} = \frac{d}{dt}(x + A^{-1}f) = \frac{dx}{dt} \quad (A^{-1}f \in \mathbb{R}^n \text{ is constant})
\]
and therefore
\[
\dot{\tilde{x}} = Ax + Bu + f = A(x + A^{-1}f) + Bu = A\tilde{x} + Bu.
\]
Now, substituting in \( y = Cx + Du + g \) for \( y \) and \( x \) in terms of \( \tilde{y} \) and \( \tilde{x} \) gives
\[
\tilde{y} + g - CA^{-1}f = C(\tilde{x} - A^{-1}f) + Du + g = C\tilde{x} - CA^{-1}f + Du + g
\]
or
\[
\tilde{y} = C\tilde{x} + Du.
\]
Hence, \( \tilde{x} \), \( u \) and \( \tilde{y} \) satisfy the following LDS:
\[
\dot{\tilde{x}} = A\tilde{x} + Bu, \quad \tilde{y} = C\tilde{x} + Du.
\]