EE263 final exam, December 2022

- 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. 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 12/16 after everyone has taken it.
- 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 question on a new page. Correctly assign pages to problems in gradescope. We may take off points if a submission does not do so.
- Please show the work you do, as it especially helps us give partial credit.
- 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/ultimate22.html`
- Good luck!

1. **Foot geometry reconstruction using pressure measurements.** In order to measure the shape of a person's foot, a technique called *plantar pressure cartography* can be used. The idea is that the person stands on a soft foam pad, under which is an array of pressure sensors. The pressure measured at a sensor depends on the compression of the foam directly above the sensor, which is determined by the shape of the foot. We will look at a simplified version of this problem in one dimension.

The base of the foot is shown as the red line in the figure below. We would like to make multiple measurements of the pressure, in order to determine the profile of the foot, that is h as a function of x. Here the blue line is a reference, since the foot may be held at an angle θ to the ground by the ankle joint.

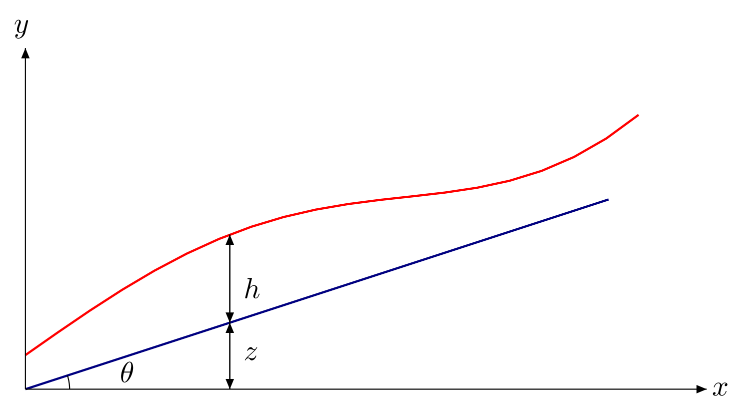
In order to reduce errors due to noise in the measurements, we ask the subject to stand on the foam pad, and make multiple measurements of the pressure. Since the pressure sensors are independent, these translate into multiple measurements of $y = h + z$. However typically during this process the subject continually adjusts their position, so the repeated measurements correspond to different values for θ, which we cannot measure.

The figure shows a large value of θ for clarity, but in practice θ is very small, and so we can use a small angle approximation $z \approx \theta x$. We will take measurements at positions $x_1, \ldots, x_q$, and take p sets of such measurements. The ith set of measurements is measured at angle θ_i. We use the notation Y_{ij} to denote measurement j at position i, and so we have

$$Y_{ij} = x_i \theta_j + h_i + w_{ij}$$

where w_{ij} is sensor noise. In practice, each measurement is a function of Y_{ij}, given by the pressure, but for simplicity here we assume that the sensor returns a position measurement.

We order the sensor locations so that the sequence $x_1, \ldots, x_q$ is increasing, and so that $x_1 > 0$.



a) The $q \times p$ matrix Y is given by its columns

$$Y = \begin{bmatrix} c_1 & c_2 & \ldots & c_p \end{bmatrix}$$

Define the pq length vector

$$y = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_p \end{bmatrix}$$

If we ignore the noise w, then there exist matrices B and C such that

$$y = C\theta + Bh$$

Give expressions for B and C. You may either give expressions for their entries, or use matrix notation.

b) We are given x, and measure Y. We would like to use least squares squares to find an estimate of h. If h^{est} is an estimate of the foot profile, then if we knew θ, then the error of this estimate is

$$E(h, \theta) = \sum_{i=1}^{q} \sum_{j=1}^{p} (Y_{ij} - x_i\theta_j - h_i)^2$$

We will estimate *both* h and θ together, by finding h^{est} and θ^{est} which mimimize $E(h, \theta)$. Show that the solution to this problem is not unique.

c) Because this is a least squares problem, we can give a parameterization of all solutions. Specifically, there is a matrix P and vector r such that every optimal h^{est} and θ^{est} is given by

$$\begin{bmatrix} \theta^{\text{est}} \\ h^{\text{est}} \end{bmatrix} = Pv + r$$

for some vector v. Explain how to find P and r given Y and x. Your method must give a matrix P of full column rank.

d) Give a brief interpretation of the nonuniqueness, in terms of the experimental setup.

e) To remove the nonuniqueness, we choose a *representative* solution. Let $\mathcal{H}$ be the set of all optimal solutions

$$\mathcal{H} = \{Pv + r \mid v \in \mathbb{R}^k\}$$

then the representative h^{rep} is the h-component (last q elements) of a vector $\begin{bmatrix} \theta \\ h \end{bmatrix} \in \mathcal{H}$ such that $h_i \geq 0$ for all i and $\sum_i h_i$ is minimized. The P you found above has a particular form which makes the representative easy to find. Give an algorithm to find it, and a brief interpretation.

f) The file `foot.json` contains Y and x. Find the h-component of an optimal least-squares solution h^{est}. Plot h_i^{est} versus i.

g) For the representative solution h^{rep}. Plot h_i^{rep} versus i.

Solution. Here is the solution.

a) Since w_{ij} is noise, we do not factor it into our expression for y. Let C_k be the $q \times p$ matrix such that $(C_k)_{ij} = x_i$ if $j = k$ and 0 otherwise. We let

$$C = \begin{bmatrix} x & & & \\ & x & & \\ & & \ddots & \\ & & & x \end{bmatrix} \qquad B = \begin{bmatrix} I \\ \vdots \\ I \end{bmatrix}$$

where C is a block diagonal matrix containing p copies of x, and B is a matrix containing p copies of the $q \times q$ identity matrix. Then the equation

$$y = C\theta + Bh$$

is equivalent to

$$Y_{ij} = x_i \theta_j + h_i$$

Alternatively, we can write

$$C = I_p \otimes x \qquad B = \mathbf{1}_p \otimes I_q$$

where I_p is the $p \times p$ identity matrix, and $\mathbf{1}_p$ is the p length vector with all entries equal to one.

b) The solution is not unique because the matrix

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

has a nonzero null-space. Specifically, the vector $\begin{bmatrix} \mathbf{1} \\ -x \end{bmatrix}$ is an element of the null space.

In more detail, notice that $E(h, \theta)$ is equivalent to $\|y - C\theta - Bh\|_2^2$. Suppose that h', θ' are vectors such that

$$\|y - C\theta' - Bh'\|_2^2 = \min_{h,\theta} \|y - C\theta - Bh\|_2^2.$$

It follows that h', θ' are solutions for minimizing $E(h, \theta)$. Suppose g, ϕ are vectors such that

$$\|y - C\theta' - Bh'\|_2^2 = \|y - C(\theta' + \phi) - B(h' + g)\|_2^2.$$

Notice that

$$\|y - C(\theta' + \phi) - B(h' + g)\|_2^2 = \|y - C\theta' - Bh' - (C\phi + Bg)\|_2^2.$$

Since $C\theta' - Bh'$ is the unique projection of y into the linear subspace generated by the union of the column space of B and C, $C\phi + Bg$ must equal 0. It follows that

$$x_i \phi_j + g_i = 0$$

for $1 \leq i \leq q$ and $1 \leq j \leq p$. We are given that $x_1 = 0$ and x is increasing, so $x_i > 0$ for $i > 1$. Thus, in order for $x_2\phi_j = -g_2$ for all j, we need that all the elements of ϕ are the same. Additionally, $-g_i$ must always equal cx_i for some scalar c for all i. Thus, we have that $\phi = c\mathbf{1}$ and $g = -cx$ if h', θ' and $h' + g, \theta' + \phi$ are solutions to our minimization problem. This solution set has dimension 1, as all of the solutions are along a line parameterized by c.

c) We let

$$P = \begin{bmatrix} \mathbf{1} \\ -x \end{bmatrix}$$

where $\mathbf{1}$ is the p-length vector of ones. This is a nonzero vector, so clearly it's a full column rank matrix. It follows that v is a scalar replacing c from the previous part. The product Pv gives us the uncertainty $\begin{bmatrix} \phi \\ g \end{bmatrix}$ in our solution. Thus, $r = \begin{bmatrix} \theta' \\ h' \end{bmatrix}$ must be a particular solution to minimizing $E(\theta, h)$. We let $A = \begin{bmatrix} C & B \end{bmatrix}$. We need that r minimizes $\|y - Ax\|_2$. We can find q by taking the product $A^\dagger y$ where $A^\dagger$ is the pseudo-inverse of A.

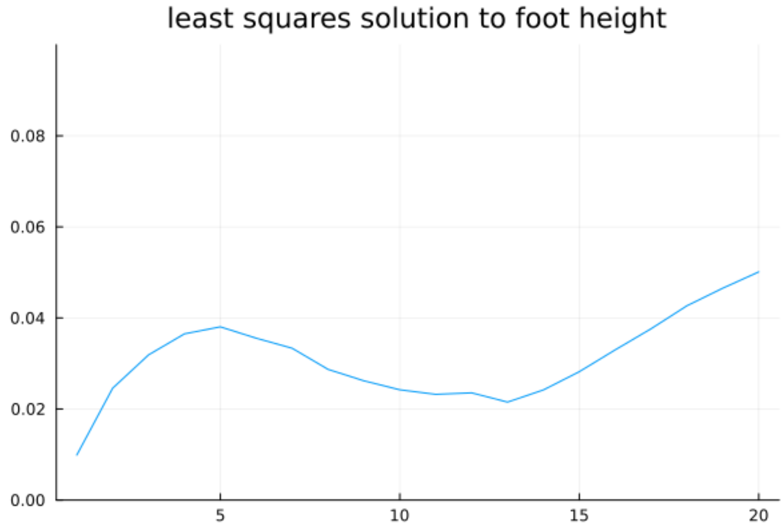
d) The reason we have this uncertainty is as follows. If someone raises their foot (increases θ) slightly by the same amount in every measurement, we see that all of the final measurements are higher. We can't determine from these measurements unambiguously that the person raised their foot slightly in every measurement, as their foot could be inclined/slanted proportionally to the sensor positions x. Thus, if there's a slight offset to ever measurement y that is proportional to x, we can't determine if this is due to the angle of the foot with respect to the ankle θ or if it's due to the slanting/shape of the foot.

e) To find the representative solution, we start by finding the least squares solution r to the problem $\|y - Ax\|_2$. Clearly, the least squares solution is in the solution set. We let h denote the last q elements of r. We divide h by the elements of x element-wise. We let c be the minimum element of the resulting vector of quotients. We then let

$$r^{\text{rep}} = r - Pc.$$

Since x is the last q elements of P, subtraction by Pc will ensure that the h-component of our representative vector will have non-negative entries and the sum of the entries will be minimized. We then let h^{rep} be the last q elements of r^{rep}. What our algorithm does is first find a consistent mathematical solution to minimize E. Next, we use the assumption that the foot is flat on the measurement pad, so all the heights should be positive, and at some point, the height is roughly zero (where the pressure is greatest). Based on this assumption, we can "tilt" the foot contour we get from the least squares solution into one we believe better represents the foot in actuality.

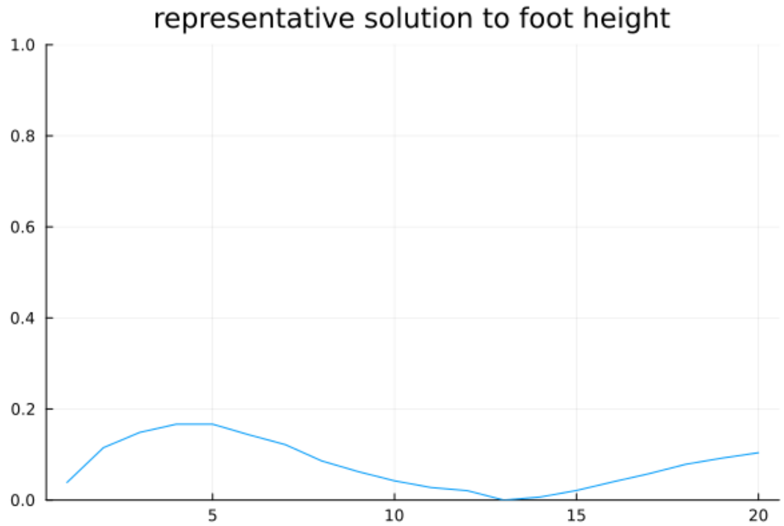
f) We generate the following plot for the least squares solution to the height of the foot:



We use the following Julia code:

```
using LinearAlgebra
using Plots
include("readclassjson.jl");

data = readclassjson("../data/foot.json")
x = data["x"]
Y = data["Y"]
q, p = size(Y)
y = vec(Y);

C = kron(I(p),x)
B = reduce(vcat,[I(q) for i in 1:p])
A = [C B];

hest = (A \ y)[p+1:end]
plot(1:q,hest,legend=false,title="least squares solution to foot height",ylims=(0,1))
```

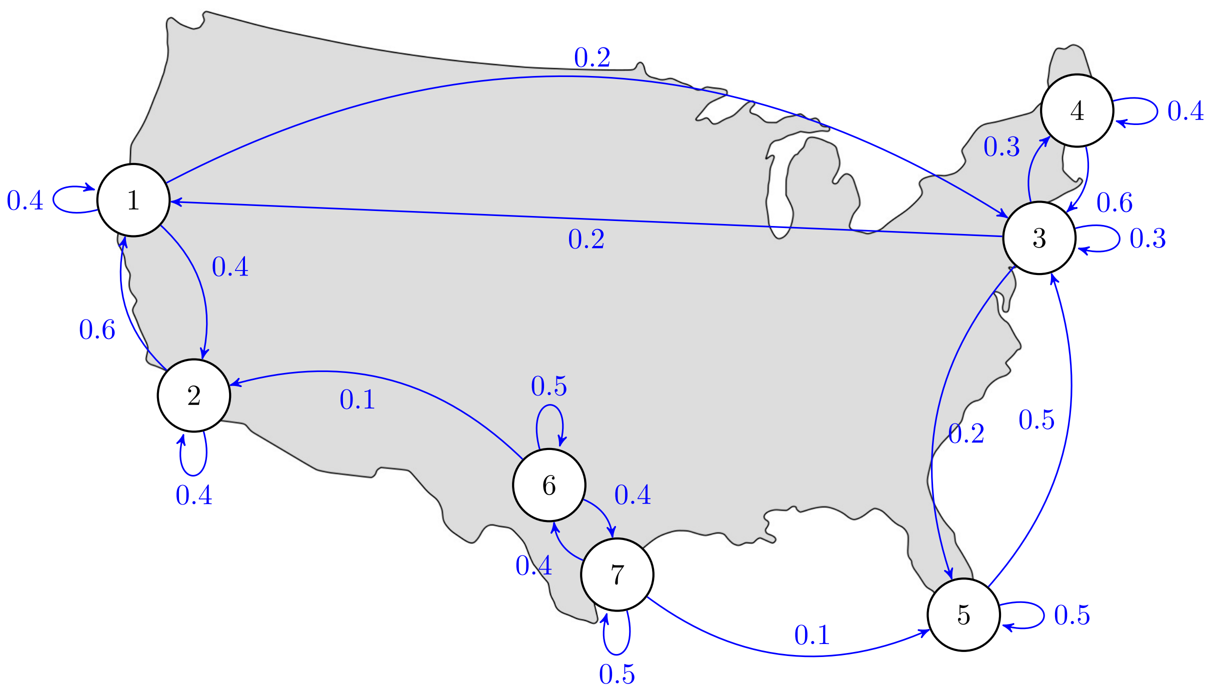
g) We generate the following plot for the representative solution to the height of the foot:



We use the following Julia code:

```
c = minimum(hest./x)
hrep = hest -c*x
plot(1:q,hrep,legend=false,title="representative solution to foot height",ylims=(0,1))
```

2. The great master of linear algebra. A great master of linear algebra has recently decided to visit the US for the first time! He will be staying in 7 major cities, and since he easily gets bored of staying in one place, after one week of staying in a city, he will either choose his next destination or decide to stay in the same place for another week. The seven cities are San Francisco, Los Angeles, New York, Boston, Miami, Austin, and Houston, which we will enumerate from 1 to 7. Since the master really loves linear dynamical systems, he chooses one of the most famous models for planning his trip, the Markov chain (with which we are familiar from the lecture slides). This means that if he is in the ith city in week t, then he will choose his destination for week $t+1$ based on the *transition probabilities*. These probabilities are shown in the Figure below. If there is no edge from city i to j, it means that the probability of visiting the jth city directly after city i is zero.



Let $z(t) \in \{1,2,3,4,5,6,7\}$ be the location of the master at week t, and define $x(t) \in \mathbb{R}^7$ as the vector of the probability distribution of $z(t)$, *i.e.*

$$x(t) = \begin{bmatrix} \text{Prob}(z(t)=1) \\ \vdots \\ \text{Prob}(z(t)=7) \end{bmatrix},$$

where t is any non-negative integer.

a) We know from the lecture slides that $x(t)$ can be expressed as an autonomous linear dynamical system, *i.e.*, there exists $A \in \mathbb{R}^{7\times 7}$ such that

$$x(t+1) = Ax(t).$$

Using the figure, find the matrix A.

b) Next, we would like to study how the probability of finding the master in different cities changes with time. We know that, for some reason, he has chosen Houston as his port of entry to the US, which means that $x(0) = e_7 = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}^\mathsf{T}$. Plot $x_i(t)$ for $i = 1, \cdots, 7$ and $t = 0, 1, \cdots, T$, where $T = 50$. You should generate one figure with all 7 lines on one plot, where the ith plot shows $x_i(t)$ versus t. We call this the *graph of probabilities*, and we will be plotting it in the next sections as well.

How do the probabilities behave for large t? Which cities have the property that for large enough t, we know for sure that the master will never be there? Use the probabilities in the figure to briefly explain why this happens.

c) Show that $w_1^\mathsf{T} = \mathbf{1}^T = \begin{bmatrix}1 & 1 & \cdots & 1\end{bmatrix}$ is a left eigenvector of A. What is the corresponding eigenvalue, namely λ_1? Now, let v_1 be the right eigenvector corresponding to λ_1. Explain how v_1 is related to the values observed on the plot in part (**??**).

d) We have been informed that in 50 weeks from now, there will be an international online linear algebra contest, and since we have taken EE263, we are really hopeful that we can win the contest. If at the time of the competition, i.e., at $t = T = 50$ the great master happens to be in San Francisco, we can ask him to be our team leader, and that will guarantee our win! However, from parts (b) and (c), we know that there is no way to make sure he will be in the Bay Area at that time, and there is a non-zero probability that he will not be here. He starts in Houston. Fortunately, we have been able to access the server where he runs his Markov chain (while we don't really know why he runs it on a server since it can be done on any personal computer), and at time t, we can directly modify his location probabilities as we wish. In other words, we get to choose a vector $u(t) \in \mathbb{R}^7$ and perturb the master's Markov chain as

$$x(t+1) = Ax(t) + u(t).$$

However, the challenge is that it is super-expensive to perturb the Markov chain, and at each time step t, it costs $\|u(t)\|^2 \times \$100K$ to apply the input $u(t)$. So, we would like to make sure that he will visit San Francisco at $t = T$, while we incur the minimum cost, *i.e.*, minimize $\sum_{t=0}^{T} \|u(t)\|^2$. Note that making sure he will be here at $t = T$ means that the probability vector at $t = T$ is given by $x(T) = e_1 = \begin{bmatrix}1 & 0 & \cdots & 0\end{bmatrix}^\mathsf{T}$. Formulate the problem of minimizing the cost subject to the constraint $x(T) = e_1$ in the form of

$$\begin{array}{ll} \text{minimize} & \|A_1 u - b_1\|^2 \\ \text{subject to} & C_1 u = d_1, \end{array}$$

where $u = \begin{bmatrix} u(0) \\ \vdots \\ u(T) \end{bmatrix}$. Express A_1, b_1, C_1, and d_1 explicitly. Then solve this problem for the given A with $T = 50$ and plot the graph of probabilities and report the cost. You must observe that the resulting vectors $x(t)$ are not actual probability vectors, *i.e.*, they may have negative values and may also sum to a value other than 1. Confirm that they may have negative values by referring to your graph of probabilities. Also, confirm that they may sum to a value other than one by plotting $s(t) = \sum_{i=1}^{7} x_i(t)$ versus t.

e) In part (**??**), we observed that our method may result in values of $x(t)$ that are not actual probability vectors. If the master notices this, he will get very angry and probably leave the US for good! So, we need to solve this issue. First, we concentrate on the constraint that for each t, we must have $s(t) = \sum_{i=1}^{7} x_i(t) = 1$. Formulate the problem of minimizing the cost subject to the constraint $x(T) = e_1$ and $s(t) = 1$ for all t in the form of

$$\begin{array}{ll} \text{minimize} & \|A_2 u - b_2\|^2 \\ \text{subject to} & C_2 u = d_2. \end{array}$$

Express A_2, b_2, C_2, and d_2 explicitly. Then solve this problem for the given A with $T = 50$, plot the graph of probabilities, and report the cost. Confirm that the resulting vectors $x(t)$ may have negative values by referring to your graph of probabilities. Also, confirm that the issue of their summation has been resolved by plotting $s(t)$ versus t. In addition, plot $u_1(t), \cdots, u_7(t)$ versus t on the same figure. This will be a figure containing 7 plots, each corresponding to one input. We will call this the *graph of inputs*. You must observe that $u_i(t)$ is zero for small t. Intuitively explain why this happens. A brief qualitative description will suffice.

f) Finally, we need to address the last issue, which is to enforce the non-negativity constraint for the probability values, *i.e.*, $x_i(t) \geq 0$. If we knew how to deal with inequality constraints in optimization problems, we could easily solve this issue. However, since that is not covered in this course, we will take an alternative approach. Let J_1 be the objective that we minimized in part (**??**). We now define

$$J_2 = \sum_{t=0}^{T} \sum_{i=1}^{7} (x_i(t) - p)^2 ,$$

where $p = \frac{1}{7}$, and focus on a multi-objective minimization problem by minimizing $J_1 + \mu J_2$. Explain why adding the term μJ_2 to our objective from part (**??**) can solve the issue of having negative $x_i(t)$'s, and why the choice of $p = \frac{1}{7}$ might be a reasonable choice. Formulate the problem of minimizing $J_1 + \mu J_2$ with the same constraints as the last part in the form of

$$\begin{array}{ll} \text{minimize} & \|A_3 u - b_3\|^2 \\ \text{subject to} & C_3 u = d_3. \end{array}$$

Express A_3, b_3, C_3, and d_3 explicitly. Then solve this problem for the given A with $T = 50$ for a range of values of μ. Plot the trade-off curve of J_1 and J_2. Plot the graph of probabilities for $\mu = 100$ and justify what you observe. Next, by inspection, find the smallest value of μ for which the constraint $x_i(t) \geq 0$ holds for all i and t. Report this value, namely μ^*, along with the corresponding value of J_1. Plot the corresponding graph of probabilities and the graph of inputs. Compare the latter with your results from part (**??**) and justify what you observe.

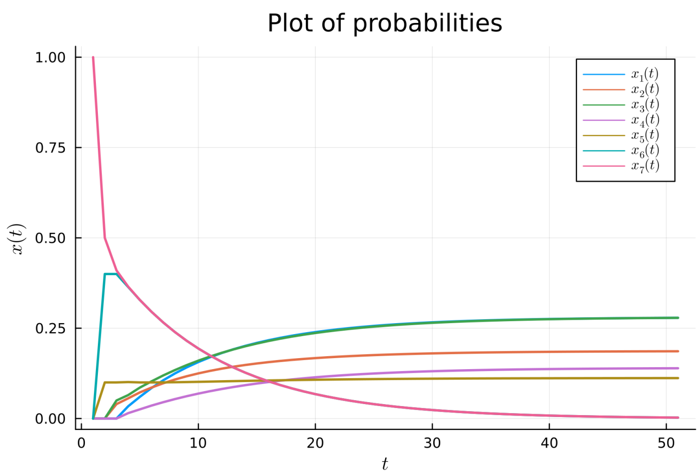
g) Based on your graph of inputs from the last part, you may notice that we seem to be incurring some unnecessary costs for perturbing the Markov chain for small values of t. It seems reasonable that we shouldn't have to perturb the Markov chain at early stages if we only need to modify the location probabilities for some distant $t = T$. Use your intuition to modify the way we defined J_2 to address this issue and achieve a lower cost compared to part (**??**) while we maintain the condition $x_i(t) \geq 0$. You don't need to implement your new method, but briefly justify why it will work. Note that the answer to this part is not unique.

Solution.

a)

$$A = \begin{bmatrix} 0.4 & 0.6 & 0.2 & 0 & 0 & 0 & 0 \\ 0.4 & 0.4 & 0 & 0 & 0 & 0.1 & 0 \\ 0.2 & 0 & 0.3 & 0.6 & 0.5 & 0 & 0 \\ 0 & 0 & 0.3 & 0.4 & 0 & 0 & 0 \\ 0 & 0 & 0.2 & 0 & 0.5 & 0 & 0.1 \\ 0 & 0 & 0 & 0 & 0 & 0.5 & 0.4 \\ 0 & 0 & 0 & 0 & 0 & 0.4 & 0.5 \end{bmatrix}$$

b) For large t, all the probabilities converge to a limit. For cities 6 and 7 this limit is zero, and the reason is that once we get out of them, there is no probability of getting back to them. In other words, no edge from the other five nodes enters these two, so in the long run, with probability one, the master will not be there.



c) We have that $w_1^T A = w_1$ since the sum of each columns of A is one. So, the corresponding eigenvalue is $\lambda_1 = 1$.

By definition, we have that $Av_1 = v_1$. So whenever the probability vector is v_1, it will always remain v_1, and we observe that v_1 is the stationary distribution, *i.e.*

$$\lim_{t \to \infty} x(t) = v_1(t).$$

This means that the entries of v_1 are the limits that we observed in the figure above.

d) Let $B = I$, $C = I$, and $D = 0$. Then, we can model the problem as the LDS

$$x(t+1) = Ax(t) + Bu(t), \quad y(t) = Cx(t) + Du(t),$$

where $y(t) = x(t)$.

We know that

$$y = \begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ \vdots \\ y(T) \end{bmatrix} = \begin{bmatrix} D & & & \\ CB & D & & \\ CAB & CB & D & \\ \vdots & & \ddots & \\ CA^{T-1}B & CA^{T-2}B & \cdots CB & D \end{bmatrix} \begin{bmatrix} u(0) \\ u(1) \\ u(2) \\ \vdots \\ u(T) \end{bmatrix} + \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^T \end{bmatrix} x(0),$$

which can be represented as

$$y = \mathcal{C}_T u + d_T.$$

So, letting

$$C_1 = \begin{bmatrix} CA^{T-1}B & CA^{T-2}B & \cdots CB & D \end{bmatrix},$$

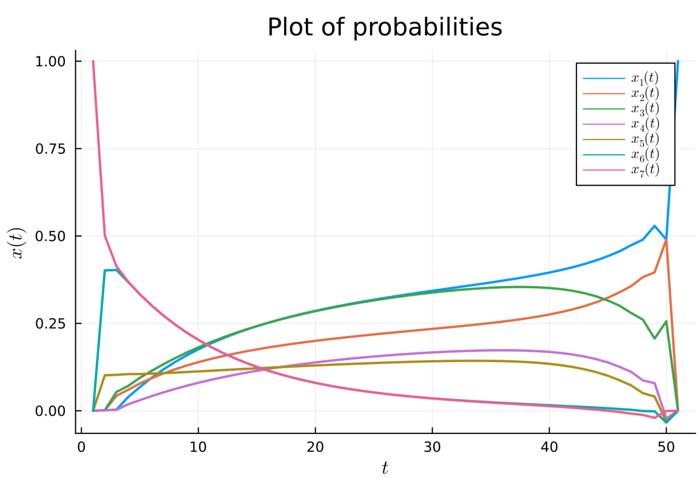
we have that

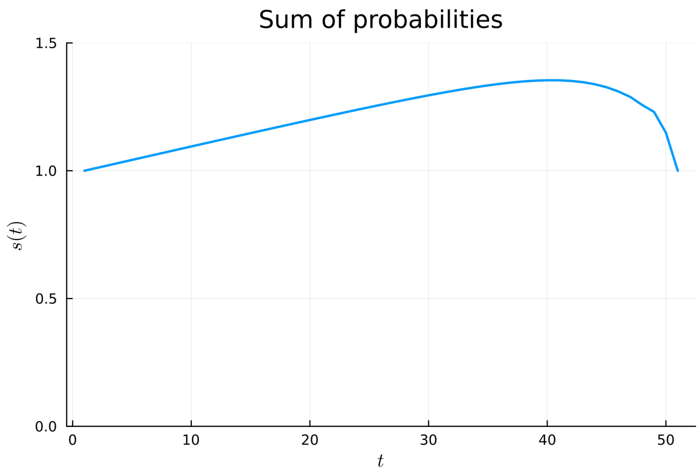
$$x(T) = C_1 u + CA^T x(0) = e_1 \Rightarrow C_1 u = e_1 - CA^T x(0) = d_1.$$

Thus,

$$\begin{aligned} A_1 &= I \\ b_1 &= 0 \\ C_1 &= \begin{bmatrix} CA^{T-1}B & CA^{T-2}B & \cdots CB & D \end{bmatrix} \\ d_1 &= e_1 - CA^T x(0). \end{aligned}$$

The plots are given in the figures below, and the cost is $J_1^{(1)} = 0.457$.

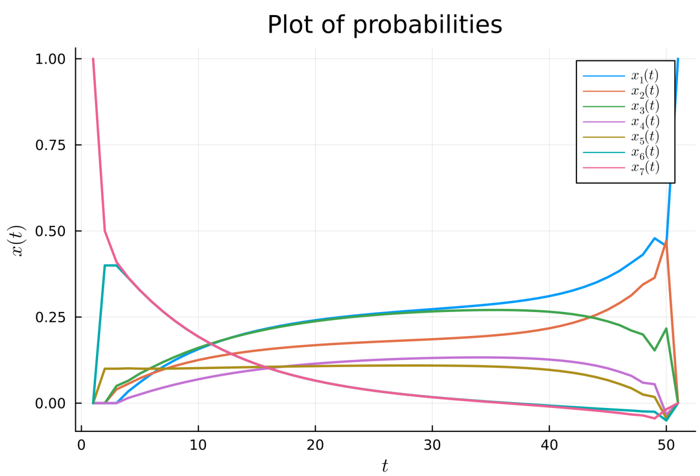


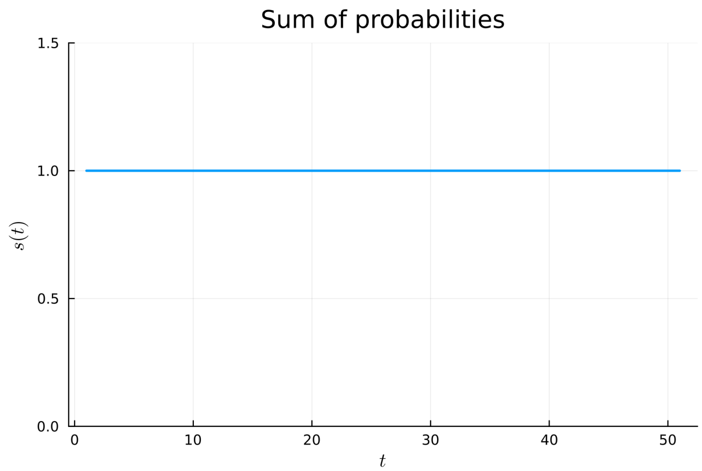


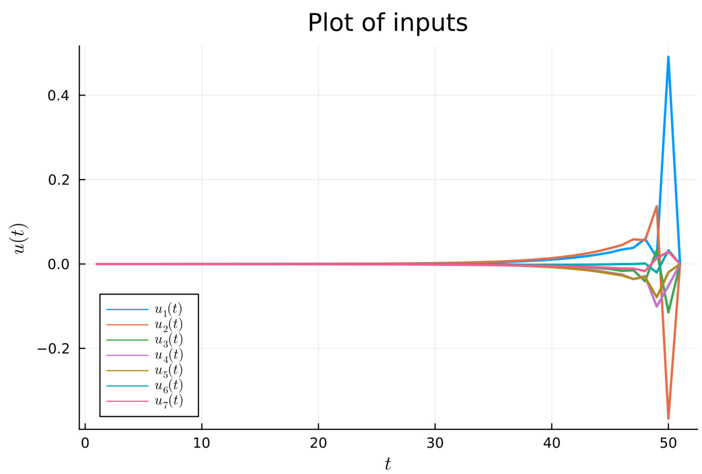
We observe that $x_6(t)$ and $x_7(t)$ are negative for some large values of t. We also observe that $s(t) \neq 1$ in general, which shows that the vectors do not sum to 1.

e) The new constraints are given by $\mathbf{1}^T x(t) = 1$ for $t = 0, 1, \cdots, T$. We also know that $x = y = \mathcal{C}_T u + d_T$, where $\mathcal{C}_T$ and d_T were introduced in the previous part. So, for each $t = 1, \cdots, T-1$ (note that we do not need to consider the constraint for $t = 0$ and $t = T$ since it is already satisfied), we have $\mathbf{1}^T x(t) = \mathbf{1}^T \hat{C}_t u + \mathbf{1}^T \hat{d}_t$, where $\hat{C}_t$ and $\hat{d}_t$ are the corresponding submatrices of $\mathcal{C}_T$ and d_T. Concatenating these values of $\hat{C}_t$ and $\hat{d}_t$ to C_1 and d_1, we get C_2 and d_2. Also, since the objective has not changed, $A_2 = A_1$ and $b_2 = b_1$.

The plots are given below, and the cost is $J_1^{(2)} = 0.464$.







We observe that $x_6(t)$ and $x_7(t)$ are still negative for some large values of t but $s(t)$ is identical to one, and the summation issue has been resolved.

We also observe that the inputs are zero for small values of t because the only constraint that we need to enforce is at $t = T$. So, for small t, no perturbation is needed. It suffices to start injecting the input when we get close enough to $t = T$.

f) The proposed method pushes the probability distributions towards being uniform, and $\frac{1}{7}$ is the probability mass of the uniform distribution. If we push the probabilities towards being uniform, the negative values of the probability vectors tend to get zero and then positive, so for large enough μ, we will definitely solve the issue.

We know that $x = y = \mathcal{C}_T u + d_T$, and $J_2 = \left\|x - \frac{1}{7}\mathbf{1}\right\|^2$. So, we have that

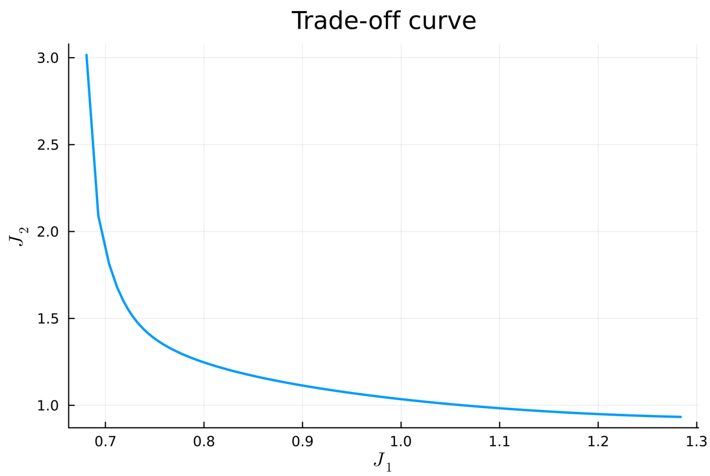
$$J_2 = \left\|x - \frac{1}{7}\mathbf{1}\right\|^2 = \left\|\mathcal{C}_T u - \left(\frac{1}{7}\mathbf{1}^2 - d_T\right)\right\|^2.$$

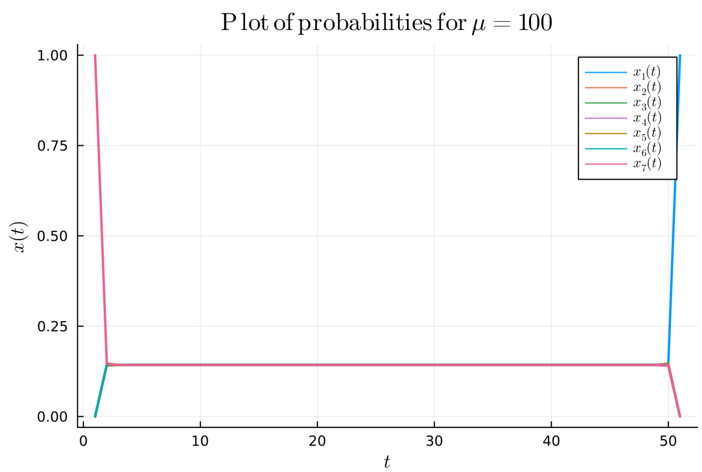
Thus,

$$A_3 = \begin{bmatrix} A_2 \\ \sqrt{\mu}\mathcal{C}_T \end{bmatrix}, \quad b_3 = \begin{bmatrix} b_2 \\ \sqrt{\mu}\left(\frac{1}{7}\mathbf{1}^2 - d_T\right) \end{bmatrix}.$$

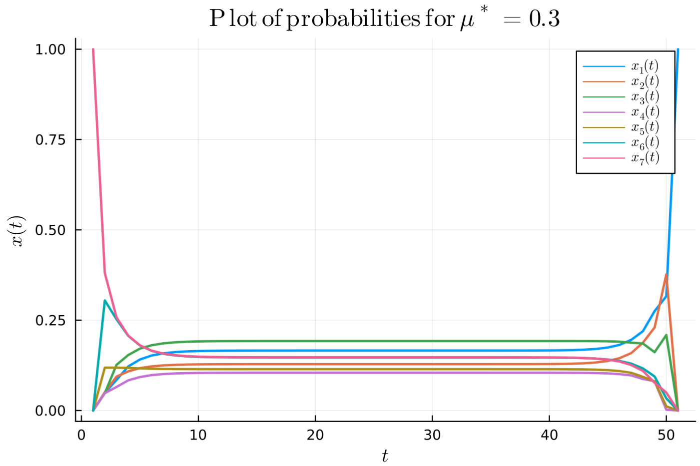
Note that C_3 and d_3 are the same as C_2 and d_2.

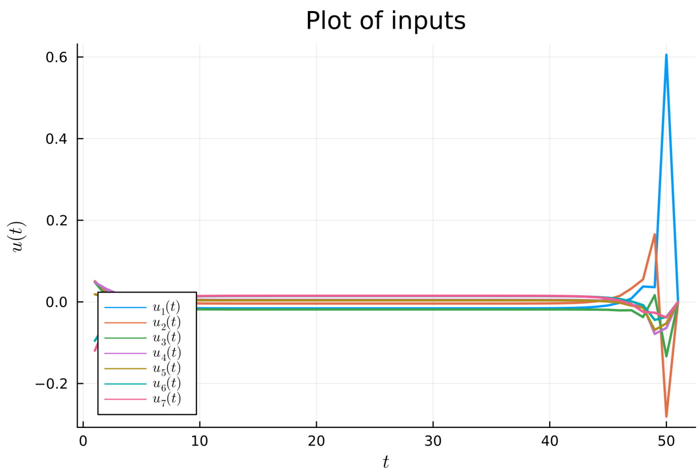
The trade-off curve is given in the figure below. The graph of probabilities for $\mu = 100$ is also below. We observe that except for $t = 0, T$ where the distribution is fixed, for all other values of t the distribution is uniform, which is exactly what we expect if we give too much weight to J_2.





By inspection, μ^* is approximately equal to 0.3, and $J_1^* = 0.609$. The corresponding plots of probabilities and inputs are given below.





We observe that $u_i(t)$ is not identical to zero for small t as opposed to the result of the last part. The reason is that we need to push the probabilities in each time step towards being uniform, and this always needs an input.

g) As we observed, we always have to apply an input which seems to be a source of unnecessary cost because we only had the issue of negative probability values for large t, so we can define J_2 only for larger values of t and let the system work as the previous part for small t. The new definition can look like

$$J_2 = \sum_{t=T_1}^{T} \sum_{i=1}^{7} (x_i(t) - p)^2 ,$$

where we choose T_1 by inspection. The larger we can make T_1, the lower the final cost will get.

3. **Pricing interchangeable goods.** Bytes cafe is introducing two new seasonal coffee beverages, the peppermint cappuccino and Christmas tree mocha. The final prices of these beverages have not yet been determined. Initial polling reveals the two beverages are partially interchangeable. If the price of a peppermint cappuccino is low enough, customers who prefer the

Christmas tree mocha will buy a peppermint cappuccino and vice versa. Let p_1 be the price of a peppermint cappuccino and p_2 be the price of a Christmas tree mocha. The prices of the two beverages will change according to demand, but the prices of the two beverages have a certain degree of "stickiness", which reduces the changes in price due to demand.

As such, we get the following dynamical system for the prices where $k_1, k_2 \geq 0$ are real scalars representing the relative price elasticities of the two goods and $c_1, c_2 \geq 0$ are real scalars representing the stickiness of the prices:

$$\ddot{p}_1 = -c_1\dot{p}_1 - k_1 p_1 + k_2 p_2 \tag{1}$$

$$\ddot{p}_2 = -c_2\dot{p}_2 + k_1 p_1 - k_2 p_2. \tag{2}$$

a) Choose an appropriate state vector x for the system described in equations (**??**) and (**??**). Construct a matrix A such that the linear dynamical system $\dot{x} = Ax$ describes all of the information in equations (**??**) and (**??**).

b) We will now use the forward Euler discretization $x(t+h) = (I + hA)x(t)$ of this system. Let $h = 0.1$, $k_1 = 5$, $k_2 = 4$, $c_1 = 1.5$, and $c_2 = 3$. Bytes cafe initially prices the beverages such that $p_1(0) = 3$ and $p_2(0) = 5.5$. Additionally, $\dot{p}_1(0) = 0 = \dot{p}_2(0)$. Plot the components of x from $t = 0$ to $T = 10$.

c) We determine that equilibrium is reached at time t if $\|x(t) - x(t-h)\| \leq 0.01$. Is equilibrium reached in this system, and why? Please formally justify your answer based on some property of A. If so, what is the time t at which it is reached? What are the prices of the beverages at equilibrium?

d) We continue to consider the forward Euler discretization from part (**??**), but we no longer know the values of k_1, k_2, c_1, c_2. The market researchers at Bytes cafe have determined that $c_1 = 1$ and $c_2 = 3$. We have taken some noisy measurements (noise has mean zero) of $p_1, p_2, \dot{p}_1, \dot{p}_2$ from $t = 0$ to $t = 10$ with $h = 0.1$ in `prices.json`. Based on these measurements and the given values of c_1, c_2, describe a method to determine the values of k_1 and k_2 that minimize the expression

$$\sum_{t=0}^{10/h-1} \|(I + hA)x(ht) - x(ht + h)\|_2^2.$$

Hint: Try to express the given expression in terms of the vector $[k_1, k_2]$.

e) Implement the method you described in part (**??**) to determine the the values of k_1 and k_2, and construct the matrix A.

f) Plot $p_1, p_2, \dot{p}_1, \dot{p}_2$ from $t = 0$ to $t = 10$ according to the forward Euler discretization $x(t+h) = (I + hA)x(t)$ with $h = 0.1$ and the matrix A determined in part (**??**). The initial value of p_1 is 4, the initial value of p_2 is 6, and both $\dot{p}_1$ and $\dot{p}_2$ start as 0. Compare this to the plot of the measurements in `prices.json`.

Solution.

a) The appropriate choice of x for this problem is the vector

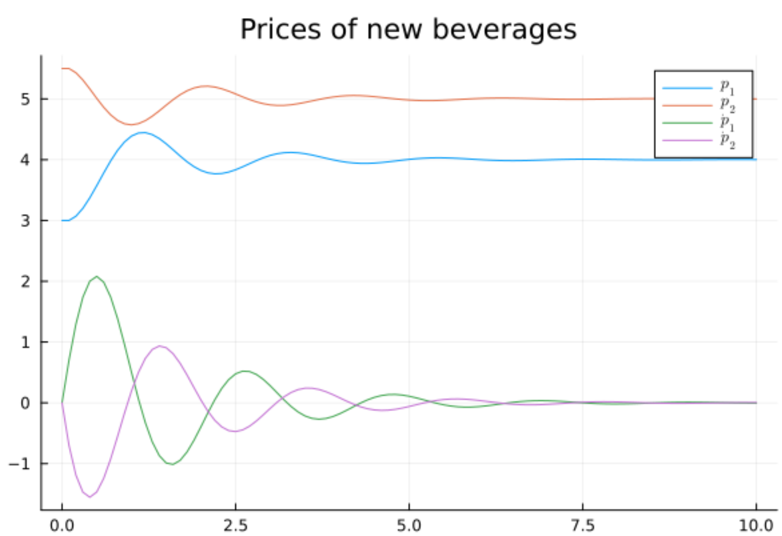
$$x = \begin{bmatrix} p_1 \\ p_2 \\ \dot{p}_1 \\ \dot{p}_2 \end{bmatrix}.$$

Note that any permutation of this described x vector is equally permissible. The correct A matrix for this given x vector is

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -k_1 & k_2 & -c_1 & 0 \\ k_1 & -k_2 & 0 & -c_2 \end{bmatrix}.$$

Note that if a permutation of the chosen x is used, the permutation must be reflected in A. With these choices, we get that $\dot{x} = Ax$.

b) We let $A_e = (I + hA)$. Our forward Euler discretization becomes $x(t+h) = A_e x(t)$. We generate the following plot if we follow the given directions:



In Julia this can be expressed in the following code:

```
using LinearAlgebra
using Plots

k1 = 5
k2 = 4
c1 = 1.5
```

```
c2 = 3
A = [0 0 1 0; 0 0 0 1; -k1 k2 -c1 0; k1 -k2 0 -c2];

h = 0.1
Ae = (I+h*A)
x0 = [3, 5.5, 0, 0]
xs = [x0]

for i=h:h:10
push!(xs,Ae*xs[end])
end

p1 = [x[1] for x in xs]
p2 = [x[2] for x in xs]
dp1 = [x[3] for x in xs]
dp2 = [x[4] for x in xs]
plot(0:h:10, [p1, p2, dp1, dp2],label=[L"p_1" L"p_2" L"\dot{p}_1" L"\dot{p}_2"])
plot!(title="Prices of new beverages")
```

c) In Julia, we can check the eigenvalues of A with the command `eigvals(A)`. Since all of the eigenvalues have negative real part or are identically zero (see the answer to HW problem 12.1930), we deduce that the system will converge to an equilibrium state where $\dot{x}(t) \approx 0$. With the given rule for determining when the system reaches an equilibrium state, we find that the system reaches an equilibrium state at time $t = 6.7$. The equilibrium value of p_1 is 3.99, and the equilibrium value of p_2 is 5.01. The following Julia code can be used to determine this:

```
using LinearAlgebra
using Plots

k1 = 5
k2 = 4
c1 = 1.5
c2 = 3
A = [0 0 1 0; 0 0 0 1; -k1 k2 -c1 0; k1 -k2 0 -c2];

h = 0.1
Ae = (I+h*A)
x0 = [3, 5.5, 0, 0]
xs = [x0]
t = 0
p1e = 0
p2e = 0

for i=h:h:10
push!(xs,Ae*xs[end])
```

```
if norm(xs[end]-xs[end-1]) <= 0.01
t = i
p1e = (xs[end])[1]
p2e = (xs[end])[2]
break
end

end

t, p1e, p2e
```

d) The matrix A is of the form

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -k_1 & k_2 & -c_1 & 0 \\ k_1 & -k_2 & 0 & -c_2 \end{bmatrix}.$$

Based on the given information, the first two rows and first two columns of A are known. Thus, the only ambiguity in A is in the bottom left 2×2 block of A. We split A into four 2×2 block matrices as follows

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}.$$

We additionally split x into two 2-vectors as follows

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}.$$

We get that

$$\begin{aligned} \sum_{t=0}^{10/h-1} \|(I - hA)x(ht) - x(ht+h)\|_2 &= \sum_{t=0}^{10/h-1} \|hAx(ht) - (x(ht+h) - x(ht))\|_2 \\ &= \sum_{t=0}^{10/h-1} \|h \begin{bmatrix} A_{11} & A_{12} \end{bmatrix} x(ht) - (x_1(ht+h) - x_1(ht))\|_2 \\ &+ \sum_{t=0}^{10/h-1} \|h \begin{bmatrix} A_{21} & A_{22} \end{bmatrix} x(ht) - (x_2(ht+h) - x_2(ht))\|_2. \end{aligned}$$

Notice that the first sum $\sum_{t=0}^{10/h-1} \|h \begin{bmatrix} A_{11} & A_{12} \end{bmatrix} x(ht) - (x_1(ht+h) - x_1(ht))\|_2$ is entirely determined, as $\begin{bmatrix} A_{11} & A_{12} \end{bmatrix}$ is precisely known. Thus, we minimize the original expression, when we minimize the term $\sum_{t=0}^{10/h-1} \|h \begin{bmatrix} A_{21} & A_{22} \end{bmatrix} x(ht) - (x_2(ht+h) - x_2(ht))\|_2$. The block A_{22} is also known, so

$$\begin{aligned}
\sum_{t=0}^{10/h-1} \|h \begin{bmatrix} A_{21} & A_{22} \end{bmatrix} x(ht) - (x_2(ht+h) - x_2(ht))\|_2 \\
&= \sum_{t=0}^{10/h-1} \|hA_{21}x_1(ht) + hA_{22}x_2(ht) - (x_2(ht+h) - x_2(ht))\|_2 \\
&= \sum_{t=0}^{10/h-1} \|hA_{21}x_1(ht) - (x_2(ht+h) - x_2(ht) - hA_{22}x_2(ht))\|_2.
\end{aligned}$$

We now let $a = \operatorname{vec} A_{21}$. We define the matrix W_{ht} such that

$$W_{ht} := \begin{bmatrix} -p_1(ht) & p_2(ht) \\ p_1(ht) & -p_2(ht) \end{bmatrix}$$

where I is the 2×2 identity matrix. It follows that $W_{ht}a = A_{21}x_1(ht)$. Thus,

$$\sum_{t=0}^{10/h-1} \|h \begin{bmatrix} A_{21} & A_{22} \end{bmatrix} x(ht) - (x_2(ht+h) - x_2(ht))\|_2 = \sum_{t=0}^{10/h-1} \|hW_{ht}a - (x_2(ht+h) - x_2(ht) - hA_{22}x_2(ht))\|_2.$$

Thus, a is the vector that minimizes the norm $\|(hW)a - b\|_2$ where

$$W := \begin{bmatrix} W_0 \\ W_h \\ \vdots \\ W_{10-h} \end{bmatrix}, \quad b := \begin{bmatrix} x_2(h) - x_2(0) - hA_{22}x_2(0) \\ x_2(2h) - x_2(h) - hA_{22}x_2(h) \\ \vdots \\ x_2(10) - x_2(h10 - h) - hA_{22}x_2(10 - h) \end{bmatrix}.$$

We can use least squares to solve for a, and then we generate A from a and the known blocks of A.

e) We get that $k_1 = 2.486$ and 2.783. The resulting matrix A should be

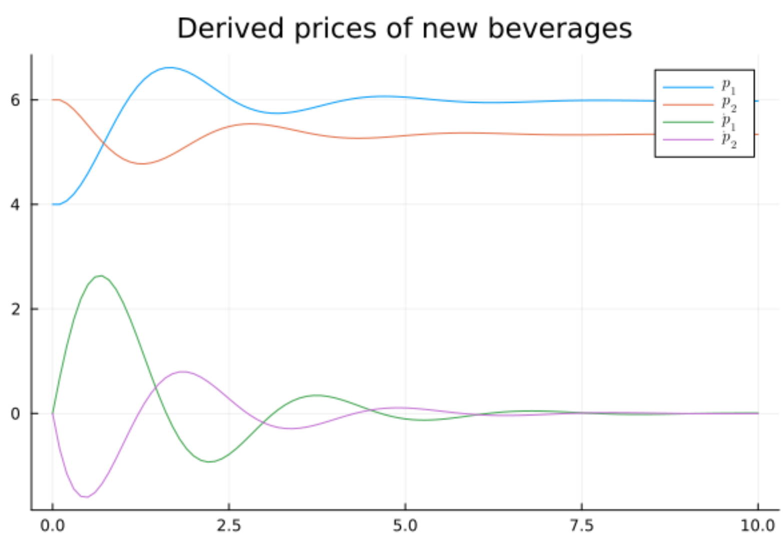
$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -2.486 & 2.783 & -1 & 0 \\ 2.486 & -2.783 & 0 & -3 \end{bmatrix}.$$

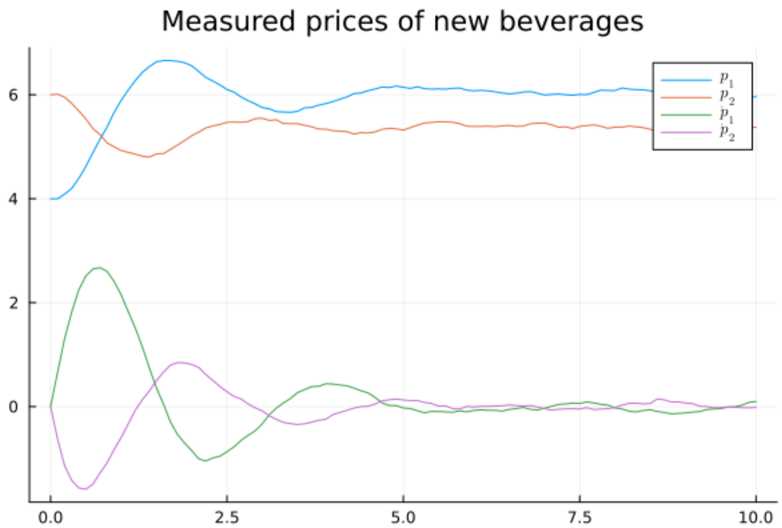
This part can be done with the following Julia code:

```
using LinearAlgebra
using LaTeXStrings
using Plots
include("readclassjson.jl");

x = readclassjson("../data/prices.json")["x"]
h = 0.1
A22 = [-1 0; 0 -3];
```

```
W = zeros(0,2)
b = zeros(0)

for i=1:Int64(10/h)
xt = x[i,:]
xt1 = x[i+1,:]
row = h*[-xt[1] xt[2]; xt[1] -xt[2]]
vec = xt1[3:4]-xt[3:4]-h*A22*xt[3:4]
W = vcat(W, row)
b = vcat(b, vec)
end

a = W\b
A21 = [-a[1] a[2]; a[1] -a[2]]
A = [0 0 1.0 0; 0 0 0 1.0];
A = vcat(A,hcat(A21,A22))
```

f) We generate the following two plots:





These plots can be generated with the following Julia code:

```
h = 0.1
Ae = (I+h*A)
x0 = [4.0, 6.0, 0, 0]
xs = [x0]

for i=h:h:10
push!(xs,Ae*xs[end])
end

p1 = [x[1] for x in xs]
p2 = [x[2] for x in xs]
dp1 = [x[3] for x in xs]
dp2 = [x[4] for x in xs]
plot(0:h:10, [p1, p2, dp1, dp2],label=[L"p_1" L"p_2" L"\dot{p}_1" L"\dot{p}_2"])
plot!(title="Derived prices of new beverages")

p1 = x[:,1]
p2 = x[:,2]
dp1 = x[:,3]
dp2 = x[:,4]
plot(0:h:10, [p1, p2, dp1, dp2],label=[L"p_1" L"p_2" L"\dot{p}_1" L"\dot{p}_2"])
plot!(title="Measured prices of new beverages")
```

4. Some true or false questions. For each of the statements below, state whether it is true or false. If true, give a brief one-sentence explanation why. If false, give a counterexample.

a) If $A \in \mathbb{R}^{n\times n}$, $A^2 = 0$, and λ is an eigenvalue of A, then $\lambda = 0$

b) If $A \in \mathbb{R}^{n\times n}$ and $A^2 = 0$, then rank of A is at most 2.

c) If $A \in \mathbb{R}^{n\times n}$ and $A^2 = A$, then $A + I$ is invertible.

d) Suppose $n > 2$. If $A \in \mathbb{R}^{n\times n}$ and $A^2 = A$, then A can have n distinct eigenvalues.

e) For a square matrix A, $\operatorname{rank}(A^2) = \operatorname{rank}(A)$ if and only if range $A \cap$ null $A = \{0\}$.

f) If $A, B, P \in \mathbb{R}^{n\times n}$ and $B = P^{-1}AP$, then A and B have the same eigenvalues.

g) If $A, B, P \in \mathbb{R}^{n\times n}$ and $B = P^{-1}AP$, then A and B have the same singular values.

h) If $A \in \mathbb{R}^{n\times n}$ has a repeated eigenvalue, then A is not diagonalizable.

i) If $A \in \mathbb{R}^{n\times n}$ is symmetric and $A^k = 0$ for some $k \in \mathbb{N}$, then $A = 0$.

j) If $A \in \mathbb{R}^{n\times n}$ is symmetric and $A^k = I$ for some $k \in \mathbb{N}$, then $A = I$.

k) If A, B are real symmetric matrices and $A \geq B$, then $\lambda_{\max}(A) \geq \lambda_{\max}(B)$.

l) If A, B are real symmetric matrices and $A, B \geq 0$, then for any $x \in \mathbb{R}^n$, $x^T ABx \geq 0$.

m) If $A \in \mathbb{R}^{n\times n}$, symmetric, and $A > 0$, then $A + A^{-1} \geq 2I$.

n) If A is a real symmetric positive definite matrix, then $\lambda_i(A) = \sigma_i(A)$.

o) If $A \in \mathbb{R}^{m\times n}, B \in \mathbb{R}^{n\times p}$, then $\sigma_{\max}(AB) \leq \max\{\sigma_{\max}(A), \sigma_{\max}(B)\}$.

p) If $A = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R_1 \\ 0 \end{bmatrix}$ is the QR decomposition of A, then A and R_1 have the same singular values.

Solution.

a) If $A \in \mathbb{R}^{n\times n}$, $A^2 = 0$, and λ is an eigenvalue of A, then $\lambda = 0$

True. This is true, since if λ is an eigenvalue of A then λ^2 is an eigenvalue of A^2, and therefore $\lambda^2 = 0$, hence $\lambda = 0$.

b) If $A \in \mathbb{R}^{n\times n}$ and $A^2 = 0$, then rank of A is at most 2.

False. A counterexample is

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

c) If $A \in \mathbb{R}^{n\times n}$ and $A^2 = A$, then $A + I$ is invertible.

True. Because $(I - A/2)(I + A) = I$.

d) Suppose $n > 2$. If $A \in \mathbb{R}^{n \times n}$ and $A^2 = A$, then A can have n distinct eigenvalues.

False. Because if λ is an eigenvalue of A we must have $\lambda^2 = \lambda$ and hence λ is 0 or 1.

e) For a square matrix A, $\text{rank}(A^2) = \text{rank}(A)$ if and only if range $A \cap$ null $A = \{0\}$.

True. Suppose $\text{range}(A) \cap \text{null}(A) \neq \{0\}$. Then there exists nonzero x, z such that $x = Az$ and $Ax = 0$. So $A^2 z = 0$, and hence $\text{null}(A^2)$ is larger than $\text{null}(A)$ and so $\text{rank}(A^2) < \text{rank}(A)$. Conversely, suppose $\text{rank}(A^2) < \text{rank}(A)$. Then there exists $z \in \text{null}(A^2)$ such that $z \notin \text{null}(A)$. Let $y = Az$, then $y \in \text{range}(A) \cap \text{null}(A)$.

f) If $A, B, P \in \mathbb{R}^{n \times n}$ and $B = P^{-1}AP$, then A and B have the same eigenvalues.

True. Because $\det(sI - A) = \det(sI - B)$.

g) If $A, B, P \in \mathbb{R}^{n \times n}$ and $B = P^{-1}AP$, then A and B have the same singular values.

False. For example

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$$

h) If $A \in \mathbb{R}^{n \times n}$ has a repeated eigenvalue, then A is not diagonalizable.

False. For example $A = I$ is diagonal.

i) If $A \in \mathbb{R}^{n \times n}$ is symmetric and $A^k = 0$ for some $k \in \mathbb{N}$, then $A = 0$.

True. Diagonalization shows that all eigenvalues must be zero, in which case A must be zero.

j) If $A \in \mathbb{R}^{n \times n}$ is symmetric and $A^k = I$ for some $k \in \mathbb{N}$, then $A = I$.

False. For example, pick $A = -I$ and $k = 2$.

k) If A, B are real symmetric matrices and $A \geq B$, then $\lambda_{\max}(A) \geq \lambda_{\max}(B)$.

True. Because $A \geq B$ implies $x^\mathsf{T} A x \geq x^\mathsf{T} B x$ for all x, hence

$$\max_{x \mid \|x\| \leq 1} x^\mathsf{T} A x \geq \max_{x \mid \|x\| \leq 1} x^\mathsf{T} B x$$

l) If A, B are real symmetric matrices and $A, B \geq 0$, then for any $x \in \mathbb{R}^n$, $x^T ABx \geq 0$.

False. For example

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 200 \end{bmatrix} \quad B = \begin{bmatrix} 5 & -2 \\ -2 & 5 \end{bmatrix} \quad x = \begin{bmatrix} -5 \\ -1 \end{bmatrix}$$

m) If $A \in \mathbb{R}^{n \times n}$, symmetric, and $A > 0$, then $A + A^{-1} \geq 2I$.

True. Diagonalize, and use the fact that $x + 1/x \geq 2$ for all real x.

n) If A is a real symmetric positive definite matrix, then $\lambda_i(A) = \sigma_i(A)$.

True. Since the eigenvalue decomposition is also the singular value decomposition.

o) If $A \in \mathbb{R}^{m\times n}, B \in \mathbb{R}^{n\times p}$, then $\sigma_{\max}(AB) \leq \max\{\sigma_{\max}(A), \sigma_{\max}(B)\}$.

False. Pick $A = 2$ and $B = 2$.

p) If $A = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R_1 \\ 0 \end{bmatrix}$ is the QR decomposition of A, then A and R_1 have the same singular values.

True. If $A = U\Sigma V^\mathsf{T}$ then $R = Q^\mathsf{T} U\Sigma V^\mathsf{T}$, which is a singular value decomposition of R

5. A layout problem. In this question, there are m sources with locations $s_j \in \mathbb{R}^2$ for $j = 1, \dots, m$ and n destinations with locations $d_i \in \mathbb{R}^2$ for $i = 1, \dots, n$. We have K links, each of which connects a single source to a single destination. The k'th link connects source $s_{\text{src}(k)}$ to destination $d_{\text{dst}(k)}$. The source locations are fixed, but we would like to decide the positions of the destinations.

You are given an incidence matrix $B \in \mathbb{R}^{K\times m}$ which specifies which source is connected to each link. Specifically

$$B_{kj} = \begin{cases} 1 & \text{if } j = \text{src}(k) \\ 0 & \text{otherwise} \end{cases}$$

Similarly the matrix $C \in \mathbb{R}^{K\times n}$ specifies which destination is connected to each link, via

$$C_{ki} = \begin{cases} 1 & \text{if } i = \text{dst}(k) \\ 0 & \text{otherwise} \end{cases}$$

We also define the matrices $S \in \mathbb{R}^{2\times m}$ and $D \in \mathbb{R}^{2\times n}$ by

$$S = \begin{bmatrix} s_1 & s_2 & \dots & s_m \end{bmatrix} \qquad D = \begin{bmatrix} d_1 & d_2 & \dots & d_n \end{bmatrix}$$

a) Let $A = C^\mathsf{T} B$. Give an interpretation for A_{ij}.

b) Each link k has an operational cost, given by

$$c_k = w_k \|s_{\text{src}(k)} - d_{\text{dst}(k)}\|^2$$

Here w_k is a positive constant associated with link k. The total cost is

$$J_1 = \sum_{k=1}^{K} c_k$$

Express J_1 in terms of S, D, C, B and w.

c) As above, with a source and port connected via link k, the cost is the weight w_k multiplied by the corresponding distance squared. To reduce the cost of links, we would like to minimize the total weighted distance between all the sources and ports. The file `layout.json` contains B, C, w, and S. Find the set of destination locations D that minimizes J. Make a plot showing the sources and the optimal destinations. (Use one color for sources and another for destinations.)

d) In this problem, we are also going to connect each destination i to its neighbors. For convenience, define the wrapping function

$$\text{wrap}(i) = \begin{cases} i+n & \text{if i} < 1 \\ i-n & \text{if i} > \text{n} \\ i & \text{otherwise} \end{cases}$$

for $-1 \leq i \leq n+1$. We define the smoothness of the set of destinations by $q_1, q_2, \ldots, q_n \in \mathbb{R}^2$ given by

$$q_i = d_{\text{wrap}(i-1)} - 2d_i + d_{\text{wrap}(i+1)}$$

Let the matrix Q be

$$Q = \begin{bmatrix} q_1 & q_2 & \ldots & q_n \end{bmatrix}$$

Find the matrix H such that $Q = DH$.

e) We prefer not to have sharp corners on the path through the destinations. So we have a secondary objective

$$J_2 = \sum_{i=1}^{n} \|q_i\|^2$$

We would like to find the destination positions D that minimize

$$J_1 + \mu J_2$$

Explain how you would do this.

f) Find the optimal destination positions when $\mu = 1$. Plot the sources and destinations.

g) Find the optimal destination positions when $\mu = 10$. Plot the sources and destinations.

Solution.

a) We have $A_{ij} = 1$ if there is a link between destination i and source j, otherwise $A_{ij} = 0$.

b) We have

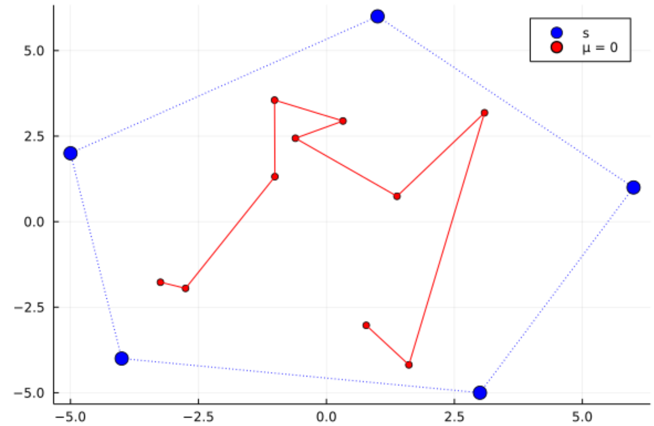
$$\begin{aligned} J_1 &= \sum_{k=1}^{K} c_K \\ &= \sum_k w_k \|s^{\text{src}(k)} - d^{\text{dst}(k)}\|^2 \\ &= \|W^{\frac{1}{2}}(BS^\mathsf{T} - CD^\mathsf{T})\|_F^2 \end{aligned}$$

where $W = \text{diag}(w)$.

c) The optimal D is given by

$$D^\mathsf{T} = (W^{\frac{1}{2}}C)^\dagger W^{\frac{1}{2}} BS^\mathsf{T}$$

The optimal destination locations are



d) The matrix H is given by

$$H = \begin{bmatrix} -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 \end{bmatrix}$$

e) We can write the objective J_2 as

$$J_2 = \|DH\|_F^2$$

hence

$$\begin{aligned} J_1 + \mu J_2 &= \|W^{\frac{1}{2}}(BS^\mathsf{T} - CD^\mathsf{T})\|_F^2 + \mu\|HD^\mathsf{T}\|_F^2 \\ &= \|\tilde{A}D^\mathsf{T} - \tilde{B}\|_F^2 \end{aligned}$$

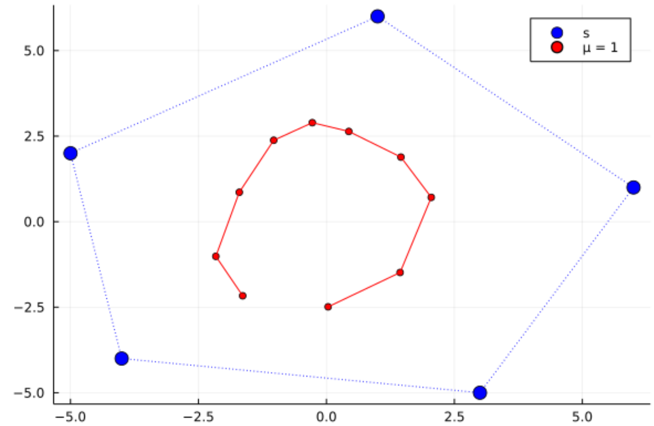
where

$$\tilde{A} = \begin{bmatrix} W^{\frac{1}{2}}C \\ \sqrt{\mu}H \end{bmatrix} \quad \tilde{B} = \begin{bmatrix} W^{\frac{1}{2}}BS^\mathsf{T} \\ 0 \end{bmatrix}$$

Then the optimal D is given by

$$D^\mathsf{T} = \tilde{A}^\dagger \tilde{B}$$

f) With $\mu = 1$ we have the plot below.



g) With $\mu = 10$ we have the plot below.