This homework is due on Wednesday, April 1, 2020, at 11:59PM.

Self-grades are due on Monday, April 6, 2020, at 11:59PM.

1 SVD

Consider the matrix

$$A = \begin{bmatrix} -1 & 1 & 5 \\ 3 & 1 & -1 \\ 2 & -1 & 4 \end{bmatrix}.$$

a) **Calculate $A^T A$ and AA^T.**

b) **Use the procedure from class to find the SVD of A in the form $\sum_{i=1}^{r} \sigma_i \vec{u}_i \vec{v}_i^T$. You may use either $A^T A$ or AA^T, whichever appears advantageous.**

2 Rank 1 Decomposition

In this problem, we will decompose a few images into linear combinations of rank 1 matrices. Remember that outer product of two vectors $\vec{s}\vec{g}^T$ gives a rank 1 matrix. It has rank 1 because clearly, the column span is one-dimensional — multiples of $\vec{s}$ only — and the row span is also one dimensional — multiples of $\vec{g}^T$ only.

a) Consider a standard 8×8 chessboard shown in Figure 1. Assume that black colors represent -1 and that white colors represent 1.

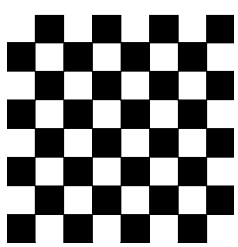

Figure 1: 8×8 chessboard.

Hence, that the chessboard is given by the following 8×8 matrix C_1:

$$C_1 = \begin{bmatrix} 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \end{bmatrix}$$

Express C_1 as a linear combination of outer products. *Hint: In order to determine how many rank 1 matrices you need to combine to represent the matrix, find the rank of the matrix you are trying to represent.*

b) For the same chessboard shown in Figure 1, now assume that black colors represent 0 and that white colors represent 1.

Hence, the chessboard is given by the following 8×8 matrix C_2:

$$C_2 = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Express C_2 as a linear combination of outer products.

c) Now consider the Swiss flag shown in Figure 2. Assume that red colors represent 0 and that white colors represent 1.

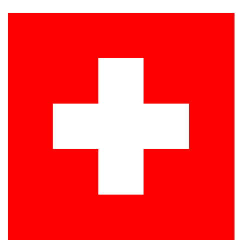

Figure 2: Swiss flag.

Assume that the Swiss flag is given by the following 5×5 matrix S:

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

Furthermore, we know that the Swiss flag can be viewed as a superposition of the following pairs of images:

Figure 3: Pairs of images - Option 1

Figure 4: Pairs of images - Option 2

Express the S in two different ways: i) as a linear combination of the outer products inspired by the Option 1 images and ii) as a linear combination of outer products inspired by the Option 2 images.

3 Open-Loop Control of SIXT33N

Last time, we learned that the ideal input PWM for running a motor at a target velocity v^* is:

$$u(t) = \frac{v^* + \beta}{\theta}$$

In this problem, we will extend our analysis from one motor to a two-motor car system and evaluate how well our open-loop control scheme does.

$$v_L(t) = d_L(t+1) - d_L(t) = \theta_L u_L(t) - \beta_L$$
$$v_R(t) = d_R(t+1) - d_R(t) = \theta_R u_R(t) - \beta_R$$

a) In reality, we need to "kickstart" electric motors with a pulse in order for them to work. That is, we can't go straight from 0 to our desired input signal for $u(t)$, since the motor needs to overcome its initial inertia in order to operate in accordance with our model.

Let us model the pulse as having a width (in timesteps) of t_p. In order to model this phenomenon, we can say that $u(t) = 255$ for $t \in [0, t_p - 1]$[1]. In addition, the car initially (at $t = 0$) hasn't moved, so we can also say $d_L(0) = d_R(0) = 0$.

Firstly, let us examine what happens to d_L and d_R at $t = t_p$, that is, after the kickstart pulse has passed. Find $d_L(t_p)$ and $d_R(t_p)$. (*Hint:* If it helps, try finding $d_L(1)$ and $d_R(1)$ first and then generalizing your result to the t_p case.)

Note: It is very important that you distinguish θ_L and θ_R, as the motors we have vary in their parameters, just as how real resistors vary from their ideal resistance.

b) Let us define $\delta(t) = d_L(t) - d_R(t)$ as the difference in positions between the two wheels. If both wheels of the car are going at the same velocity, then this difference δ should remain constant since no wheel will advance by more ticks than the other. As a result, this will be useful in our analysis and in designing our control schemes.

Find $\delta(t_p)$. For both an ideal car ($\theta_L = \theta_R$ and $\beta_L = \beta_R$) where both motors are perfectly ideal and a non-ideal car ($\theta_L \neq \theta_R$ and $\beta_L \neq \beta_R$), did the car turn compared to before the pulse?

Note: Since $d(0) = d_L(0) = d_R(0) = 0$, $\delta(0) = 0$.

c) Even if the car turns a little bit during the initial pulse (t_p will be very short in lab), we can apply a control scheme that makes the car go straight afterwards; that is, make $\delta(t \to \infty)$ converge to a constant value (as opposed to growing without bounds).

Let's try applying the open-loop control scheme to each of the motors independently, and see if our car still goes straight.

$$u_L(t) = \frac{v^* + \beta_L}{\theta_L}$$
$$u_R(t) = \frac{v^* + \beta_R}{\theta_R}$$

Let $\delta(t_p) = \delta_0$. Find $\delta(t)$ for $t \geq t_p$ in terms of δ_0. (*Hint:* As in part (a), if it helps you, try finding $\delta(t_p + 1)$, $\delta(t_p + 2)$, etc., and generalizing your result to the $\delta(t)$ case.)

Does $\delta(t \to \infty)$ deviate from δ_0? Why or why not?

[1] $x \in [a, b]$ means that x goes from a to b inclusive.

d) Unfortunately, in real life, it is hard to capture the precise parameters of the car motors like θ and β, and even if we did manage to capture them, they could vary as a function of temperature, time, wheel conditions, battery voltage, etc. In order to model this effect of **model mismatch**, we consider model mismatch terms (such as $\Delta\theta_L$), which reflects the discrepancy between the model parameters and actual parameters.

$$d_L(t+1) - d_L(t) = (\theta_L + \Delta\theta_L)u_L(t) - (\beta_L + \Delta\beta_L)$$
$$d_R(t+1) - d_R(t) = (\theta_R + \Delta\theta_R)u_R(t) - (\beta_R + \Delta\beta_R)$$

Let us try applying the open-loop control scheme again to this new system. Note that **no model mismatch terms appear below** – this is intentional since our control scheme is derived from the model parameters for θ and β, not from the actual $\theta + \Delta\theta$, etc.[2]

$$u_L(t) = \frac{v^* + \beta_L}{\theta_L}$$
$$u_R(t) = \frac{v^* + \beta_R}{\theta_R}$$

As before, let $\delta(t_p) = \delta_0$. Find $\delta(t)$ for $t \geq t_p$ in terms of δ_0.

Does $\delta(t \to \infty)$ change from δ_0? Why or why not, and how is it different from the previous case of no model mismatch?

You may have noticed that open-loop control is insufficient in light of non-idealities and mismatches. In the next problem, we will analyze a more powerful form of control (closed-loop control) which should be more robust against these kinds of problems.

4 Closed-Loop Control of SIXT33N

To make our control more robust, we introduce feedback, turning our open-loop controller into a closed-loop controller. In this problem, we derive the closed-loop control scheme you will use to make SIXT33N reliably drive straight.

We introduce $\delta[k] = d_L[k] - d_R[k]$ as the difference in positions between the two wheels. We will consider a proportional control scheme, which introduces a feedback term into our input equation in which we apply gains k_L and k_R to $\delta[k]$ to modify our input at each timestep in an effort to prevent $|\delta[k]|$ from

[2] Why not just do a better job of capturing the parameters, one may ask? Well, as noted above, the mismatch can vary as a function of an assortment of factors including temperature, time, wheel conditions, battery voltage, and it is not realistic to try to capture the parameters under every possible environment, so it is up to the control designer to ensure that the system can tolerate a reasonable amount of mismatch.

growing without bound. To do this, we will modify our inputs $u_L[k]$ and $u_R[k]$ to be:

$$u_L[k] = \frac{v^* + \beta_L}{\theta_L} - k_L \frac{\delta[k]}{\theta_L}$$
$$u_R[k] = \frac{v^* + \beta_R}{\theta_R} + k_R \frac{\delta[k]}{\theta_R}$$

Substituting into the open-loop equations

$$d_L[k+1] - d_L[k] = \theta_L u_L[k] - \beta_L \tag{1}$$
$$d_R[k+1] - d_R[k] = \theta_R u_R[k] - \beta_R$$

we obtain:

$$d_L[k+1] - d_L[k] = v^* - k_L \delta[k] \tag{2}$$
$$d_R[k+1] - d_R[k] = v^* + k_R \delta[k]$$

a) Let's look a bit more closely at picking k_L and k_R. First, we need to figure out what happens to $\delta[k]$ over time. Find $\delta[k+1]$ in terms of $\delta[k]$.

b) Given your work above, what is the eigenvalue of the system defined by $\delta[k]$? For discrete-time systems like our system, $\lambda \in (-1, 1)$ is considered stable. Are $\lambda \in [0, 1)$ and $\lambda \in (-1, 0]$ identical in function for our system? Which one is "better"? (*Hint:* Preventing oscillation is a desired benefit.)

   Based on your choice for the range of λ above, how should we set k_L and k_R in the end?

c) Let's re-introduce the model mismatch in order to model environmental discrepancies, disturbances, etc. How does closed-loop control fare under model mismatch? Find $\delta_{ss} = \delta[k \to \infty]$, assuming that $\delta[0] = \delta_0$. What is δ_{ss}? (To make this easier, you may leave your answer in terms of appropriately defined c and λ obtained from an equation in the form of $\delta[k+1] = \delta[k]\lambda + c$.)

   Check your work by verifying that you reproduce the equation in part (c) if all model mismatch terms are zero. Is it better than the open-loop model mismatch?

$$d_L[k+1] - d_L[k] = (\theta_L + \Delta\theta_L)u_L[k] - (\beta_L + \Delta\beta_L)$$
$$d_R[k+1] - d_R[k] = (\theta_R + \Delta\theta_R)u_R[k] - (\beta_R + \Delta\beta_R)$$

$$u_L[k] = \frac{v^* + \beta_L}{\theta_L} - k_L \frac{\delta[k]}{\theta_L}$$
$$u_R[k] = \frac{v^* + \beta_R}{\theta_R} + k_R \frac{\delta[k]}{\theta_R}$$

5 (OPTIONAL) Write Your Own Question And Provide a Thorough Solution.

Writing your own problems is a very effective way to really learn material. Having some practice at trying to create problems helps you study for exams much better than simply solving existing practice problems. This is because thinking about how to create an interesting problem forces you to really consolidate your understanding of the course material.

6 Homework Process and Study Group

Citing sources and collaborators are an important part of life, including being a student! We also want to understand what resources you find helpful and how much time homework is taking, so we can change things in the future if possible.

a) **What sources (if any) did you use as you worked through the homework?**

b) **If you worked with someone on this homework, who did you work with?** List names and student ID's. (In case of homework party, you can also just describe the group.)

c) **How did you work on this homework?** (For example, *I first worked by myself for 2 hours, but got stuck on problem 3, so I went to office hours. Then I went to homework party for a few hours, where I finished the homework.*)

d) **Do you have any feedback on this homework assignment?**

e) **Roughly how many total hours did you work on this homework?**