AA 273 Midterm

Due no later than May 5 (Sunday), 2024 at 5 PM Pacific on Gradescope

The midterm must be completed within **a single 24 hour period**. The exam must be submitted no later than 5 PM (Pacific) on Sunday, May 5th through Gradescope.

Allowed resources

- Canvas (e.g., homework solutions, lecture notes/videos, starter code, past Ed posts)
- Textbooks
- Online documentation for any libraries that have been included in starter code (homework and exam). Examples include Numpy, Scipy, Matplotlib, Open3D, Pytorch, Pandas and standard Python libraries (math, csv).

Resources that are **NOT allowed**

- Material from previous iterations of the class (e.g., homework or exams)
- Discussion with classmates or anyone outside of the class
- The internet (except for documentation for the aforementioned libraries)
- You are not allowed to post questions relating to the midterm publicly on Ed between May 2 and May 5.

Additional Notes

- We prefer typeset solutions. However, if you choose to handwrite your solutions, the answers must be legible.
- If there are any questions regarding the exam, please direct them to the CAs through Ed as a private message or through email.

Q1 (20 points)

Probability Theory

Figure 1 below shows a probabilistic graphical model whose structure encodes conditional independence relationships between a set of random variables (in this case, X_1, X_2, Y_1, Y_2, and Y_3, all of which are binary). This network is fully defined by the distributions $P(X_1)$, $P(Y_1 \mid X_1)$, $P(X_2 \mid X_1)$, $P(Y_2 \mid X_2)$ and $P(Y_3 \mid X_2)$, which are given below as conditional probability tables (Table 1).

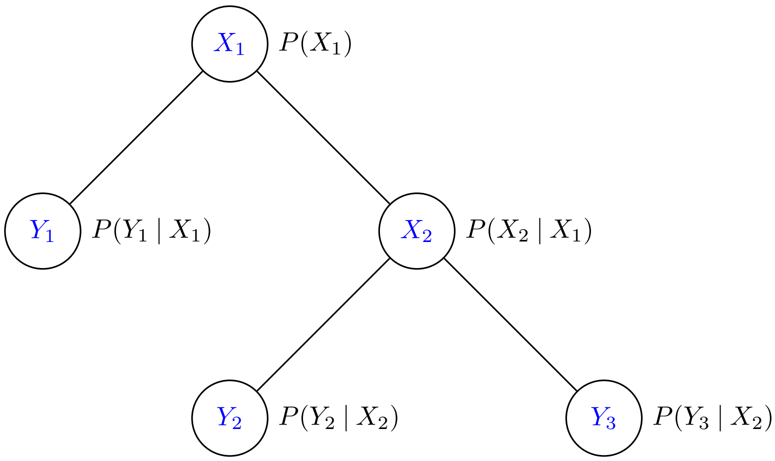

Figure 1: Probabilistic graphical model.

X_1	$P(X_1)$
0	0.5
1	0.5

X_1	Y_1	$P(Y_1 \mid X_1)$
0	0	0.8
0	1	0.2
1	0	0.3
1	1	0.7

X_1	X_2	$P(X_2 \mid X_1)$
0	0	0.6
0	1	0.4
1	0	0.1
1	1	0.9

X_2	Y_2	$P(Y_2 \mid X_2)$
0	0	0.6
0	1	0.4
1	0	0.2
1	1	0.8

X_2	Y_3	$P(Y_3 \mid X_2)$
0	0	0.9
0	1	0.1
1	0	0.1
1	1	0.9

Table 1: Conditional probability tables.

What is $P(X_2 = 0 \mid X_1 = 1,\ Y_2 = 1,\ Y_3 = 0)$? Show your work for full credit.

Q2 (20 points)

Bayes Filter
Let's use the discrete recursive Bayes filter to illustrate how a dynamics model and measurements work together over an extended state space (i.e., the state space is larger than the measurement dimension). A chores robot is required to complete 8 distinct tasks:

(i) vacuum the floor

(ii) clean the kitchen counters

(iii) wash the dishes

(iv) do the laundry

(v) take out the trash

(vi) make the beds

(vii) water the indoor plants

(viii) organize the living room

Whenever the robot finishes the list, it goes back to the start. Each hour, the robot continues its task with probability $p_s = 0.15$, goes onto the next task with probability $p_f = 0.8$, and goes back to the previous task with probability $p_b = 0.05$ (if at list item (i), it goes to item (viii) with probability p_b). This is a simplified example of a probabilistic, cyclic finite-state machine.

1. (4 points) Write the transition matrix for this system in terms of p_s, p_f, and p_b. Suppose we know the robot starts vacuuming with certainty. Propagate the system forward for 50 hours (i.e., 50 timesteps). Plot the distribution of the robot's tasks over time (i.e., a line plot where each robot task is a line). Please include a legend and axis labels.

   Hint: Make sure the distribution is a proper discrete distribution (i.e., entries between 0 and 1 with a sum of 1 at each time step without manually normalizing the result).

2. (4 points) Such a system always has an eigenvalue = 1 (among others), i.e., a discrete distribution π exists such as $\pi = T\pi$. Find the distribution π that satisfies $\pi = T\pi$ (i.e., the eigenvector associated with the eigenvalue of 1). How does this distribution connect to your answer to part 1?

   Hint: You can compute this π in code (e.g., with built-in functions like `np.linalg.eig`). Or you can do it by hand. Make sure your final x is a proper discrete distribution.

You will be gone for a few days, and ask your curious neighbor to keep an eye on your robot's tasks while you are away. Since your neighbor will not go inside your house, they can only observe your robot when it takes out the trash. Occasionally, your neighbor gets confused, thinking that the robot is taking out the trash when it actually is not, or likewise thinking the robot is not taking out the trash when it actually is. We summarize these probabilities in Table 2.

	Predicted Taking Trash Out	Predicted Not Taking Trash Out
Actually Taking Trash Out	$p_m = 0.95$	$1 - p_m = 0.05$
Not Actually Taking Trash Out	$1 - p_m = 0.05$	$p_m = 0.95$

Table 2: Curious neighbor confusion matrix

They have probability p_m of seeing the robot when it takes out the trash (i.e., true positive rate) and probability $1 - p_m$ of mistakingly seeing movement outside that is not actually a robot (i.e., false positive rate). Note that we are assuming that the true positive rate and true negative rate are symmetric. However, the neighbor is not always looking outside, so there will not be a measurement every hour.

3. (4 points) Write the measurement matrices for each prediction case. That is, show the matrices where (1) the neighbor believes they saw the robot take out the trash and (2) they believe they did not see the robot take out the trash.

4. (4 points) Implement a recursive Bayes filter to estimate the probability distribution over the robot's tasks. If the neighbor does not take a measurement, only propagate the system. Initialize the robot, starting with vacuuming, with certainty. Simulate the following scenario over 50 steps using the following measurements from your neighbor:

   - 10 steps of no measurement
   - 10 measurements of predicted taking out the trash
   - 20 measurements of predicted not taking out the trash
   - No further measurements

   Hint: Remember that you have to normalize the distribution after you include the measurement! (Check back at problem set 1)

5. (4 points) One general way to measure the unpredictability of a discrete random variable with probability distribution π is through computing it's "entropy," $H(\pi)$, defined as

$$H(\pi) = -\sum_{i=1}^{8} \pi_i \log_2(\pi_i)$$

   where the units are in "bits" since we use $\log_2$. Calculate the entropy of the distribution at each timestep, where π is the probability distribution over the robot's tasks and π_i is the probability of task i on the list. Importantly, if $\pi_i = 0$, we will have a singularity in $\log_2(\pi_i)$ with $\lim_{\pi_i \to 0^+} \pi_i \log_2(\pi_i) = 0$, by convention. Plot the entropy over time for the no-measurement case in part (1) and the with-measurement case in part (4).

   Hint: Like entropy in a thermodynamics context, we expect that entropy will increase over time if we only propagate the system. But what happens when we get a measurement?

   Hint: Don't forget the negative out front.

   Aside: This trend in entropy with propagation versus measurement is a more fundamental property of the Bayes Filter and general state estimation. This concept extends to continuous distributions and has analytical forms for distributions in the exponential family.

Q3 (20 points)

Graphical Models

For each graphical model below, reduce the conditional probability distribution to the fewest terms possible. **Be sure to explain your reasoning!** To gauge if your explanation is sufficiently thorough, your reasoning should be roughly 2-3 sentences, and the mathematical argument should use the graphical structure provided.

1. (2 points) Starting from $p(x_t \mid x_{0:t-1}, y_{1:t-1})$. Please refer to the graph in Figure 2.

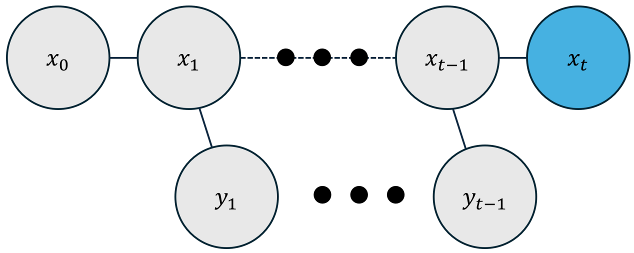

Figure 2: Graphical model for x_t as the last node

2. (3 points) Starting from $p(x_t \mid x_{0:t-1}, x_{t+1:t+2}, y_{1:t+2})$. Please refer to the graph in Figure 3.

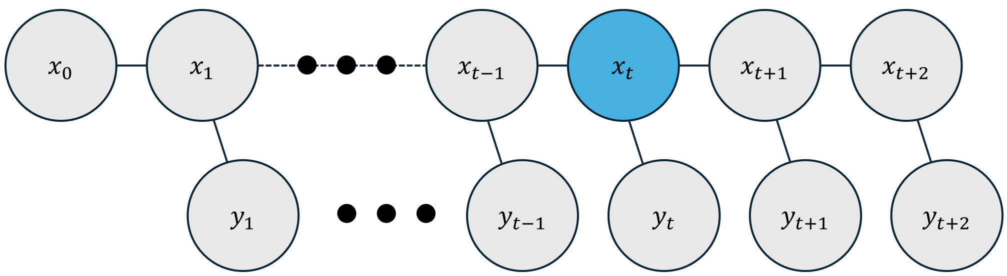

Figure 3: Graphical model for x_t with later states and measurements

3. (3 points) Starting from $p(y_t \mid x_{0:t+2}, y_{1:t-1}, y_{t+1:t+2})$. Please refer to the graph in Figure 4.

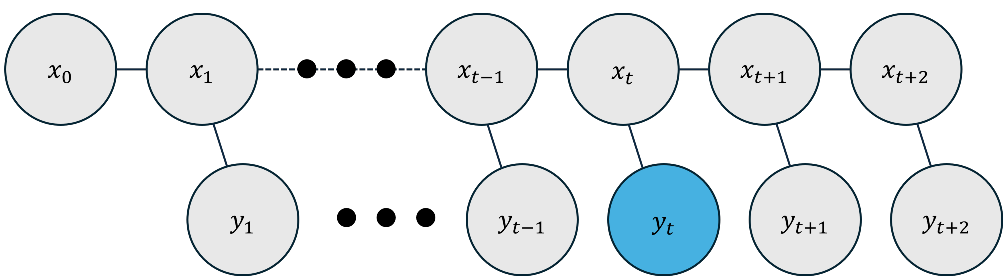

Figure 4: Graphical model for y_t with later states and measurements

4. (3 points) Starting from $p(x_t \mid x_{0:t-1}, x_{t+1:t+2}, y_{1:t+2}, y^{lc}_{(1,t)})$. This problem includes a loop closure constraint measurement $y^{lc}_{(1,t)}$. In robotics, this loop closure often comes through recognizing environmental landmarks, external information that authenticates relative states, or measurements with long latency. Please refer to the graph in Figure 5.

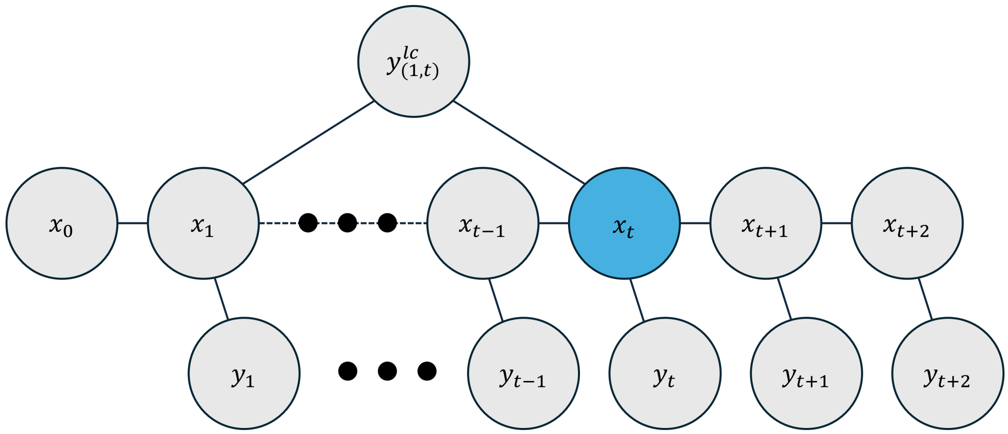

Figure 5: Graphical model for x_t with loop closure

5. (3 points) Starting from $p(y^{lc}_{(1,t)} \mid x_{0:t+2}, y_{1:t+2})$. Again, this problem includes a loop closure constraint measurement $y^{lc}_{(1,t)}$. Please refer to the graph in Figure 6.

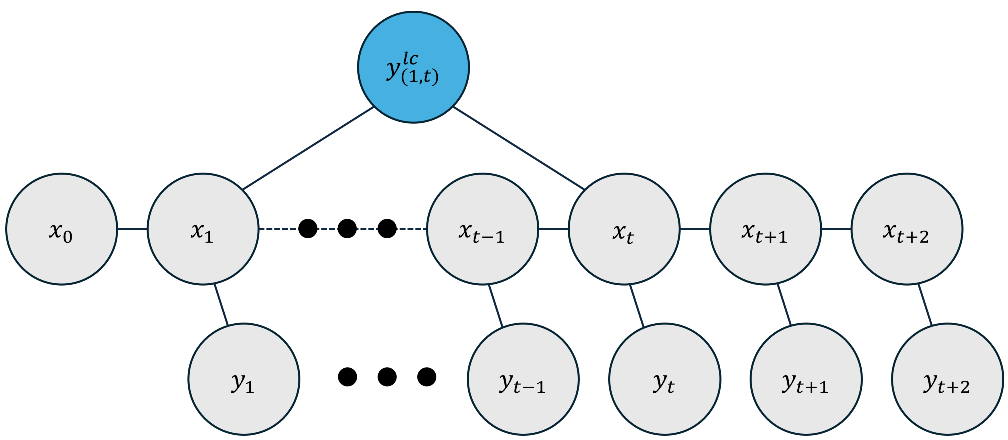

Figure 6: Graphical model for $y^{lc}_{(1,t)}$ with loop closure

6. (3 points) Starting from $p(x_t \mid x_{0:t-1}, x_{t+1}, y_{1:t+1}, y^{o}_{(0,1):(t,t+1)})$. This problem includes odometry constraint measurements $y^{o}_{(0,1):(t,t+1)}$. Specifically, we do not know the transition from x_i to x_{i+1}, but we can infer the transition from x_i to x_{i+1} through an odometry constrain measurement $y^{o}_{(i,i+1)}$. In robotics settings, odometry measurements come from cameras, lidars, wheel encoders, and more. Please refer to the graph in Figure 7.

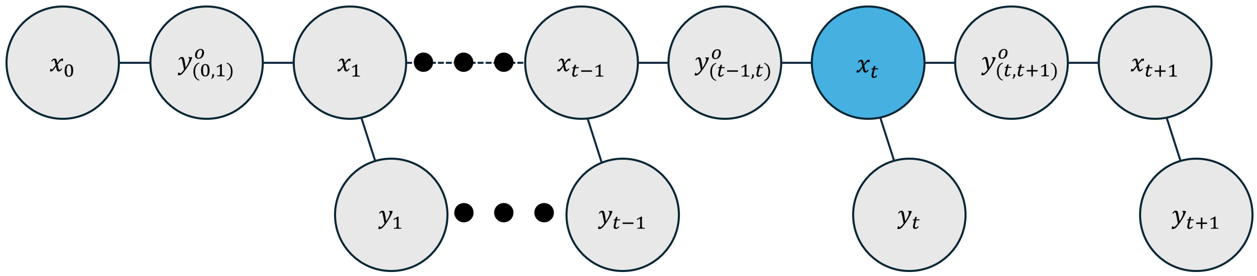

Figure 7: Graphical model for x_t with odometry measurements

7. (3 points) Starting from $p(y^o_{(t-1,t)} \mid x_{0:t+1}, y_{1:t+1}, y^o_{(0,1):(t-2,t-1)}, y^o_{(t,t+1)})$. Again, this problem includes odometry constraint measurements $y^o_{(0,1):(t,t+1)}$. Please refer to the graph in Figure 8.

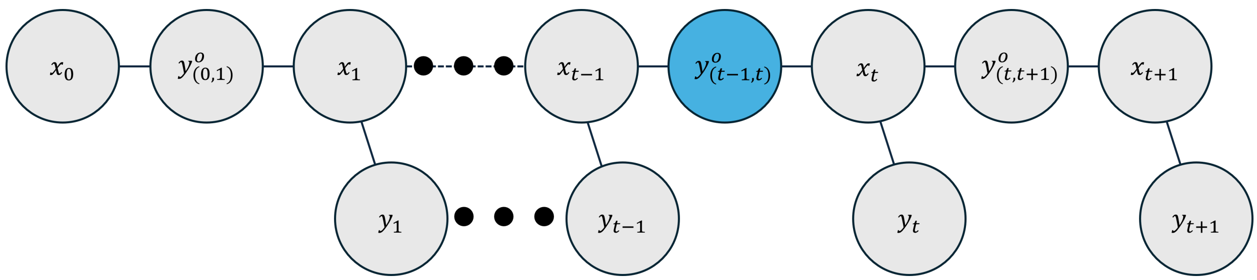

Figure 8: Graphical model for $y^o_{(t-1,t)}$ with odometry measurements

Q4 (20 points)

Gaussian Processes for Implicit Surfaces
Let's investigate using a Gaussian Process (GP) to represent the surface of an object. Consider an object $S \subset \mathbb{R}^3$ in 3D space, whose surface we write as $\partial S \subset \mathbb{R}^3$. ∂S is the continuous collection of points p in $\mathbb{R}^3$ that live on the object's surface. One common way to represent such a surface is as a zero level set of a function $f(p)$, so $\partial S = \{p \mid f(p) = 0\}$. For example, if $f(p)$ is a convex quadratic, its level sets are ellipsoids. We call such a representation an "implicit surface," since the points on the surface are given implicitly as those that solve an equation. This is in contrast to explicit representations, which give the points directly, e.g., as a collection of triangles.

Now consider a robot with a Lidar sensor collecting samples $P = (p_1, \ldots, p_N)$ of points on the surface of an object. How can we learn a function $\hat{f}(p)$ such that $\partial \hat{S} = \{p \mid \hat{f}(p) = 0\}$ is an approximation of the object's surface? Let's use a Gaussian Process to do this.

To make the problem more concrete, let's learn a function that approximates the signed distance function (SDF) of an object. A signed distance function is defined as

$$dist_{\mathcal{S}}(p) = \{+, -\} \min_{y \in \partial \mathcal{S}} ||p - y||_2,$$

so the function gives the distance from the point p to the closest point y that lies on the surface $\partial \mathcal{S}$. We use a '+' if p is outside the object S, and a '-' if inside the object, hence the term "signed" distance function. An example is given in Figure 9.

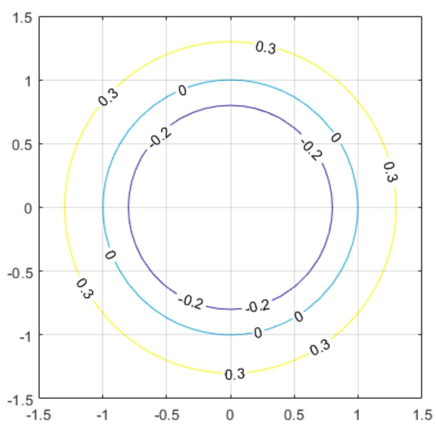

Figure 9: Example of the contours of a signed distance function for a circular object.

Let's assume we have a dataset of the form $\{(x, y) \in \mathbb{R}^2, dist_{\mathcal{S}}(x, y) \in \mathbb{R}_+\}_i^N$ where (x, y) are the 2D coordinate inputs and $dist_{\mathcal{S}}(x, y)$ the noisy distance outputs. Using this dataset, we will investigate the viability of two kernels: the linear kernel and the radial basis kernel.

1. (5 points) Let's assume we use the linear kernel function $K_{ij}(p_i, p_j) = \sigma^2 (p_i - c)^T (p_j - c)$ for parameters σ and c. First, show that this kernel is a positive semi-definite function.

   Hint: A positive semi-definite function satisfies the positive semi-definite inequality $v^T\ K(X, X)\ v \geq 0\ \forall\ X \in \mathbb{R}^{n \times k}$ *and* $v \in \mathbb{R}^n$.

2. (5 points) Using the given dataset `sdf.csv`, fit the Gaussian Process with a linear kernel to the data. You are free to pick parameters σ and c. Plot the posterior mean, as well as the ground-truth. Be sure to also report the parameters σ and c you used. Does this kernel fit the data well? If not, why do you think this is?

The csv is formatted such that the first column details what the rows represent. To parse the file into a dictionary, use the following code snippet.

```
with open('sdf.csv') as csv_file:
    reader = csv.reader(csv_file)
    data = dict(reader)
```

There are 3000 training points $(x, y, dist_{\mathcal{S}}(x, y))$ *and 3268* (43×76) *test points. In the plot, visualize as a 2D heatmap that is* $H = 43$ *pixels tall and* $W = 76$ *pixels wide. You can do this by reshaping the data into shape (H, W).*

3. (5 points) Now let's assume we have a radial basis kernel $K_{ij}(p_i, p_j) = \sigma^2 \exp\left[-\frac{1}{2\ell^2}||p_i - p_j||^2\right]$. Show that this is also a positive semi-definite function.

   Hint: Although the radial basis function is not a Gaussian PDF (p_i, p_j *are deterministic points), we can perform much of the same manipulation rules on the RBF as we did on Gaussians. We are free to define* $p_i = z + w$ *and* $p_j = z + q$ *for some dummy variable* z *and Gaussian random variables* w, q*. Assume* w, q *have the same diagonal covariances (i.e.* $\Sigma_W = \Sigma_Q = k^2 I$*). Consequently, your result from Homework 2 Problem 2 may come in handy.*

4. (5 points) Using the same dataset, fit a Gaussian Process with a radial basis kernel. Again, plot the posterior mean, as well as the ground-truth. Be sure to report the parameters you used. Does the radial basis function or the linear kernel perform better, and why is this the case?

Q5 (20 points)

Moment Generating Function for a Multivariate Gaussian Distribution
In class, we've shown that certain operations are closed under the Gaussian distribution family. For example, additions and subtractions of Gaussian random variables will remain Gaussian. We can show that this is the case more generally through the use of moment generating functions (MGFs), which uniquely define the distribution of any class of random vectors.

Let $\boldsymbol{X} = (X_1, X_2, \ldots, X_n)^T \in \mathbb{R}^n$ be a random vector and $\boldsymbol{t} = (t_1, t_2, \ldots, t_n)^T \in \mathbb{R}^n$. The moment generating function is defined as

$$M_{\boldsymbol{X}}(\boldsymbol{t}) = \mathbb{E}(e^{\boldsymbol{t}^T \boldsymbol{X}})$$

for all $\boldsymbol{t}$ for which the expectation exists.

1. (5 points) Assume $\boldsymbol{X}$ is an n-dimensional random vector. $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ and $\boldsymbol{b} \in \mathbb{R}^m$ are deterministic. Then show that the MGF of $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{b}$ is given at $\boldsymbol{t} \in \mathbb{R}^m$ by

$$M_{\boldsymbol{Y}}(\boldsymbol{t}) = e^{\boldsymbol{t}^T \boldsymbol{b}} M_{\boldsymbol{X}}(\boldsymbol{A}^T \boldsymbol{t}).$$

Hint: Expectation is a linear operator.

2. (5 points) Show that

$$M_{\boldsymbol{X}}(\boldsymbol{t}) = e^{\boldsymbol{t}^T \boldsymbol{\mu} + \frac{1}{2}\boldsymbol{t}^T \boldsymbol{\Sigma} \boldsymbol{t}}$$

is true if and only if $\boldsymbol{X}$ is an n-dimensional Gaussian random vector with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$.

Hint: You should only need to show one direction. The converse holds by equality. You may find the following substitutions helpful: $\Sigma^{-1/2}(X - \mu) = w$ *and* $u = w - \Sigma^{1/2} t$.

3. (5 points) Let $\boldsymbol{X} = (X_1, X_2, X_3)^T \in \mathbb{R}^3$ be a multivariate Gaussian random vector with mean

$$\boldsymbol{\mu} = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}^T$$

and covariance matrix

$$\boldsymbol{\Sigma} = \begin{bmatrix} 9 & 4 & 7 \\ 4 & 3 & 2 \\ 7 & 2 & 8 \end{bmatrix}.$$

Using parts (1) and (2), show that the random variable

$$Y = X_1 - X_2 + X_3$$

has a univariate Gaussian distribution.

4. (5 points) Find the mean and variance of Y using parts (1) and (2).