Graduate Artificial Intelligence 15-780

Homework 1: *Probabilistic Inference*

Out on January 30
Due on February 13

Problem 1: Independence relations in probabilistic graphical models [Felipe, 40pts]

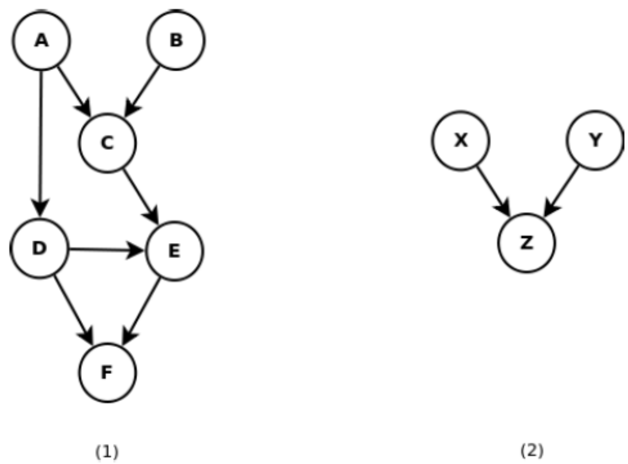


Figure 1: Directed models for Problem 1

a) For the independence relations below, say if they hold or not in Figure 1.1 and explain your answer. [8pts]

(a) $A \perp F | D, E$
Answer: Yes because all the possible trails are blocked:

- $A \rightarrow D \rightarrow F$ is blocked by D.
- $A \rightarrow D \rightarrow E \rightarrow F$ is blocked by D and also by E.
- $A \rightarrow C \rightarrow E \rightarrow F$ is blocked by E.
- $A \rightarrow C \rightarrow E \leftarrow D \rightarrow F$ is blocked by D.

(b) $B \perp D$
Answer: Yes because all the possible trails are blocked:

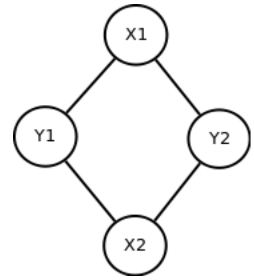
- $B \rightarrow C \leftarrow A \rightarrow D$ is blocked by C.
- $B \rightarrow C \rightarrow E \leftarrow D$ is blocked by E.
- $B \rightarrow C \rightarrow E \rightarrow F \leftarrow D$ is blocked by F.

(c) $B \perp E | C$
Answer: No because the trail $B \rightarrow C \leftarrow A \rightarrow D \rightarrow E$ is active (i.e. not block).

(d) $B \perp F | E$
Answer: No because the trail $B \rightarrow C \rightarrow E \leftarrow D \rightarrow F$ is active (i.e. not block).

b) Construct the Undirected Model structure that captures all the following independence relations: [8 pts]

- $X_1 \perp X_2 | Y_1, Y_2$
- $Y_1 \perp Y_2 | X_1, X_2$
- $X_1 \not\perp X_2 | Y_1$
- $X_1 \not\perp X_2 | Y_2$
- $Y_1 \not\perp Y_2 | X_1$
- $Y_1 \not\perp Y_2 | X_2$



Answer: The undirected model in the right is satisfies all the required independence relations. This because there are only 2 paths from X_1 to X_2, namely X_1—Y_1—X_2 and X_1—Y_2—X_2, therefore X_1 is independent of X_2 if both paths are blocked, that is, if both Y_1 and Y_2 are observed. Using a similar argument, we have that in the given undirected mode, Y_1 and Y_2 are independent only when both X_1 and X_2 are observed.

c) Is it possible to build a Directed Model such that its structure captures the same independence relations as in item b? Show the structure or proof that it is not possible. [16 pts]
Answer: No, it's not possible. Let's proof this by contradiction. Assume that such directed model $\mathscr{B}$ exists. If there is an arc between X_1 and X_2 (no matter the direction) we have $X_1 \not\perp X_2|Y_1,Y_2$. We can conclude the same for Y_1 and Y_2. Also, if there is a node Z that is not connected to the remaining ones, then Z is independent of all other nodes in $\mathscr{B}$, what violates at least one of the last four constraints. Now, let's look at the number of arcs in $\mathscr{B}$:

- *3 arcs*: If there is no v-structure in $\mathscr{B}$, then wlog $\mathscr{B}$ is $X_1 \to Y_1 \to X_2 \to Y_2$ and $X_1 \perp X_2|Y_1$, a contradiction. If there is a v-structure in $\mathscr{B}$, then wlog $\mathscr{B}$ is $X_1 \to Y_1 \leftarrow X_2 \to Y_2$ and $X_1 \perp X_2|Y_2$, another contradiction.
- *4 arcs*: If there is a v-structure in $\mathscr{B}$, suppose wlog that this v-structure is $X_1 \to Y_1 \leftarrow X_2$, then the trail $X_1 \to Y_1 \leftarrow X_2$ is active (i.e. not block) given Y_1, therefore $X_1 \not\perp X_2|Y_1,Y_2$, a contradiction. If there is no v-structure in $\mathscr{B}$, then there is a cycle of at least 3 nodes, a contradiction since directed models are acyclic by definition. □

d) The Directed Model structure in Figure 1.2, known as v-structure, is such that $X \perp Y$ and $X \not\perp Y|Z$. Considering X, Y and Z as binary random variables, $P(X=1)=0.7$ and $P(Y=1)=0.1$, show a conditional probability table for $P(Z|X,Y)$ such that $X \perp Y|Z$ holds for this Directed Model. [8 pts]
Answer: In the given Directed Model, if $X \perp Y|Z$, then we have that:

$$\begin{aligned} P(Z|X,Y) &= \frac{P(X,Y|Z)P(Z)}{P(X,Y)} \\ &= \frac{P(X|Z)P(Y|Z)P(Z)}{P(X)P(Y)} \end{aligned} \tag{1}$$

Notice that in equation (1) we only know $P(X)$ and $P(Y)$. Therefore, we can give arbitrary values to $P(X|Z)$, $P(Y|Z)$ and $P(Z)$ such that they are valid probability distributions and compute the corresponding value of $P(Z|X,Y)$. Also, in order to avoid problems when conditioning on Z, we need that $P(Z=z)>0$ for all $z \in \{0,1\}$. One possible solution is to set all the three probability distribution to uniform, i.e. $P(X=x|Z=z) = P(Y=y|Z=z) = P(Z=z) = 0.5$ resulting in $P(Z=z|X=x,Y=y)=0.5$ for all $x,y,z \in \{0,1\}^3$

Problem 2: Broken Sparrow [John, 60pts][1]

The Story So Far...

Class, the unthinkable has happen.

An unmanned aerial vehicle (UAV) loaded with literally tens of hundreds of dollars worth of atmospheric sensors has gone rogue. Well, technically it malfunctioned. A standard flight started to go wrong at time-step $t=0$. This was the was the last time-step that we received telemetry directly from the UAV.

[1] Modified from a problem suggested by Erik Zawadzki.

Since then we have been tracking it from two far-off radar towers. At $t = 0$ the UAV reported a system failure that not only disrupted its communication systems, but also knocked-out its guidance system.

You, and a special case of the Hidden Markov Model (HMM), are our only hope to find where the UAV is now.

Technical Briefing: Linear Dynamic Systems

A *Linear Dynamic System* is a special case of the HMM where the latent (unobserved or hidden) state at $t+1$ and the observation at t are both linear functions of the latent state at t with additive Gaussian noise. This special case is extremely attractive mathematically because updating the model in the face of evidence can be written as a series of linear algebra operations.

Formally the system is defined by the following equations:

$$z_{t+1} = A \cdot z_t + w_t \tag{2}$$

$$x_t = C \cdot z_t + v_t. \tag{3}$$

Here $z_t = \left(d_t^x, d_t^y, v_t^x, v_t^y, a_t^y, a_t^x\right)$ is the latent state and $x_t = \left(a_t^x, a_t^y, b_t^x, b_t^y\right)$ is observed state (the observations of (x, y) from the two towers). A and C are matrices, called the *transition* and *observation* matrices. The noise terms, w_t and v_t are distributed by zero-mean multivariate Gaussians:

$$w_t \sim \mathcal{N}(w_t \mid 0, \Gamma) \tag{4}$$

$$v_t \sim \mathcal{N}(v_t \mid 0, \Sigma) \tag{5}$$

The initial state is also a multivariate Gaussian:

$$z_0 \sim \mathcal{N}(z_0 \mid \mu_0, V_0). \tag{6}$$

Filling in the Numbers

- **Dynamics**:

  We can model the transition of the UAV with the following simple 2D dynamics:

$$a_{t+1}^x = a_t^x \tag{7}$$

$$v_{t+1}^x = v_t^x + a_t^x \cdot \Delta t \tag{8}$$

$$d_{t+1}^x = d_t^x + v_t^x \cdot \Delta t + \frac{1}{2} a_t^x \cdot (\Delta t)^2 \tag{9}$$

  A similar set of equations could be written for the y-dimension. These equations define your transition matrix. We receive observations from the radar towers every second, so $\Delta t = 1$.

  The UAV is reasonably aerodynamically stable, but because of its malfunctioning guidance system is is believed to be accelerating randomly. In particular:

$$\Gamma = \begin{pmatrix} 0.0001 & & & & & \\ & 0.0001 & & & & \\ & & 0.0001 & & & \\ & & & 0.0001 & & \\ & & & & 0.1 & \\ & & & & & 0.1 \end{pmatrix} \tag{10}$$

- **Observation**

  The two towers both provide unbiased but imprecise estimates of the UAV's $\left(d_i^x, d_i^y\right)$ position every time-step. Luckily, their noise is completely independent of each other. Therefore, the system's observation noise has the following measurement-noise covariance matrix:

$$\Sigma_1 = \Sigma_2 = \begin{pmatrix} 100 & 10 \\ 10 & 100 \end{pmatrix} \tag{11}$$

$$\Sigma = \begin{pmatrix} \Sigma_1 & \\ & \Sigma_2 \end{pmatrix} \tag{12}$$

- **Initial Position**

  The UAV was initially at $(0,0)m$ and was traveling at $(6,6)m/s$. It was not accelerating at that time. However the initial telemetry data is also subject to noise; $V_{-1} = \mathbb{I}_6$ (the 6×6 identity matrix).

Tasks

Your mission, should you choose to accept it:

a) Draw the Bayes net for time steps $t-1$, t, and $t+1$ for arbitrary time-step t. *I.e.* draw all the variables for these three time steps, and all the edges that connect them.[10 pts]

   Answer: Figure 2 shows a Bayes net representing the causal relationships between the different state variables in our problem. We see that both observations o_t at time t are controlled only by the latent location variable d_t (as well as Gaussian noise). Similarly, the velocity v_t at time t is controlled not only by v_{t-1}, but also any acceleration a_{t-1}. This velocity affects both the next time step's velocity v_{t+1}, as well as the location of the UAV d_{t+1}. Note that we leave out explicit state variables for both noise factors. If we wanted to include these explicitly, we could view the Bayes net as deterministic given two unobserved noise states; however, viewing the BN as probabilistic with implicit Gaussian noise seems more elegant.

b) Understand how to infer the posterior distribution of latent state at t from the observed data at steps $0, \ldots, t$. This inference in an LDS is performed by a *Kalman Filter*. More information can be found on Wikipedia's excellent *Kalman Filter* page, in §15.4 of Russell and Norvig, and in §13.3 of *Pattern Recognition and Machine Learning* by Bishop. [15 pts]

   For the following two problems, I based derivations off of those found in Russell & Norvig [1], as well as `http://www.cs.cmu.edu/~jch1/research/gaussmarkov/gaussmarkov.pdf`.

   (a) Derive the joint distribution for two successive states and one observation: $P(x_t, z_t, z_{t+1})$. Assume that the previous state $z_t \sim \mathcal{N}(z_t|\mu_t, V_t)$.

   Answer: By the power rule, we see:

$$P(x_t, z_t, z_{t+1}) = P(z_{t+1}|x_t, z_t) \cdot P(z_t, x_t) = P(z_{t+1}|x_t, z_t) \cdot P(x_t|z_t) \cdot P(z_t)$$

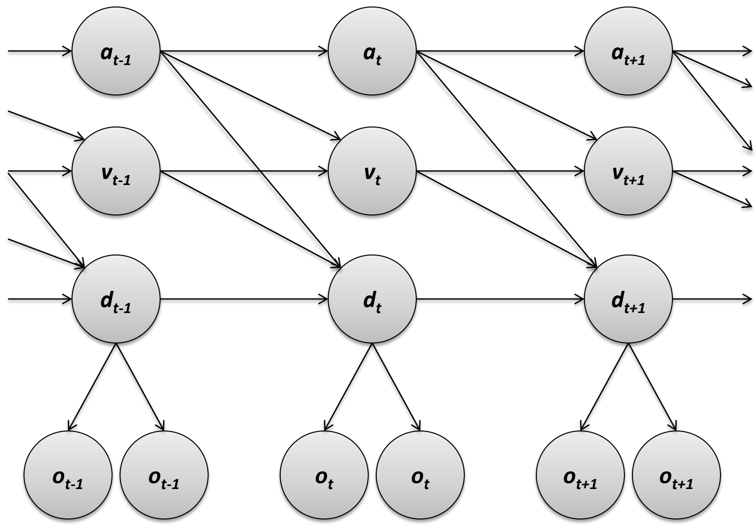


Figure 2: Bayes network for arbitrary time steps $t-1$, t, and $t+1$.

Apply the Markov property (see our Bayes Net above, or just imagine a standard HMM) to get:

$$= P(z_{t+1}|z_t) \cdot P(x_t|z_t) \cdot P(z_t)$$

We have all of these values already. First, we are given $P(z_t) \sim \mathcal{N}(\mu_t, V_t)$. The other two probabilities are just the two equations of the Kalman Filter:

i. $P(z_{t+1}|z_t) \sim \mathcal{N}(Az_t, \Gamma)$

ii. $P(x_t|z_t) \sim \mathcal{N}(Cz_t, \Sigma)$

This gives us:

$$P(z_{t+1}|z_t) \cdot P(x_t|z_t) \cdot P(z_t) \sim \mathcal{N}(Az_t, \Gamma)\mathcal{N}(Cz_t, \Sigma)\mathcal{N}(\mu_t, V_t)$$

(b) Suppose $z_t \sim \mathcal{N}(z_t|\mu_t, V_t)$ and the distribution $P(z_{t+1}|x_{t+1})$ is normal. Provide expressions for the mean and covariance.

Answer: This answer is taken pretty much verbatim from Wikipedia (or from any of the myriad Kalman Filter pages online). We are interested in the *a posteriori* probability distribution, with the mean and covariance corresponding to where we think the UAV is at the end of the Update stage. From Wikipedia:

PREDICT

$$\begin{aligned} z'_t &= A \cdot \mu_t \\ \mathscr{C}' &= A\mathscr{C}A^T + V_t \end{aligned}$$

Note that $\mathscr{C}$ is the old covariance—basically, the covariance matrix given one time step ago.

UPDATE

$$\begin{aligned} y'_{t+1} &= x_{t+1} - Cz'_t \\ S_{t+1} &= C\mathscr{C}'C^T + \Sigma \\ K_{t+1} &= \mathscr{C}'C^T S_{t+1}^{-1} \end{aligned}$$

Here, S_{t+1} is the residual covariance at the next timestep, and K_{t+1} is the Kalman gain at the next timestep. This lets us calculate both the mean position and covariance matrix of our probability distribution $P(z_{t+1}|x_{t+1})$ of the UAV's location:

MEAN: $z_{t+1} = z'_t + K_{t+1}y'_{t+1}$

COVARIANCE: $\mathscr{C} = (I - K_{t+1}C)\mathscr{C}'$

c) Plot the sensor data and sequence of posterior distribution means for the (x, y)-positions of the UAV. [25 pts]

Answer: Figure 3 shows the predicted position of the UAV across 100 time points, given our model above and data from two observation towers.

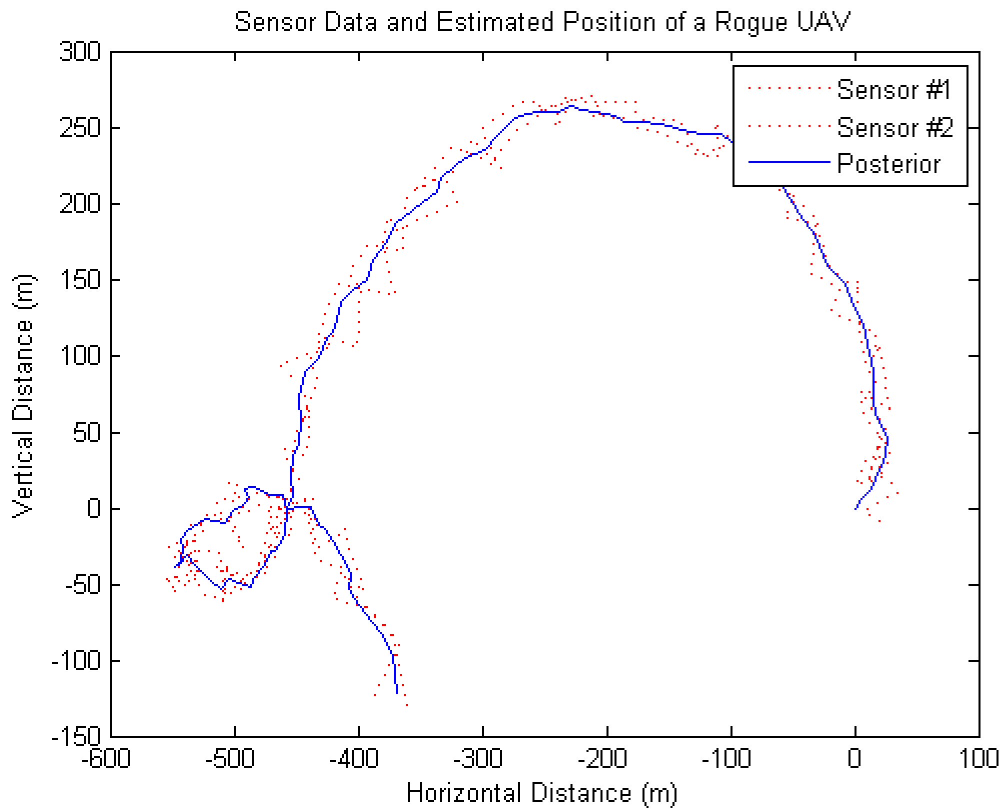


Figure 3: The UAV's predicted path (blue) from $t = 0$ to $t = 100$. Observations from both towers are shown in red.

d) Report UAV's posterior position distribution at $t = 100$. This is a multivariate Gaussian, and can be reported as its mean and 6×6 covariance matrix. [10 pts]

Answer: The posterior position distribution for the UAV at time $t = 100$ is a multivariate Gaussian with parameters

$$\left(\begin{bmatrix} -368.4868 \\ -121.5435 \\ 7.9340 \\ -20.3596 \\ 0.1156 \\ -1.8174 \end{bmatrix}, \begin{bmatrix} 25.4015 & 2.2512 & 8.9183 & 0.6453 & 1.5659 & 0.0878 \\ 2.2512 & 25.4015 & 0.6453 & 8.9183 & 0.0878 & 1.5659 \\ 8.9183 & 0.6453 & 5.3655 & 0.2875 & 1.3347 & 0.0483 \\ 0.6453 & 8.9183 & 0.2875 & 5.3655 & 0.0483 & 1.3347 \\ 1.5659 & 0.0878 & 1.3347 & 0.0483 & 0.5690 & 0.0093 \\ 0.0878 & 1.5659 & 0.0483 & 1.3347 & 0.0093 & 0.5690 \end{bmatrix} \right)$$

Appendix: Kalman Filter Matlab Code

```
%
% Load the T={0,..,100} data from each of the two observation towers
%
x = load('obs.dat')';

%
% Invariant definitions for the problem
%
% Time step is 1 second
delta_t = 1;

% Random acceleration, noise for w_t's Gaussian
Gamma = zeros(6,6);
Gamma(1,1) = 0.0001;
Gamma(2,2) = 0.0001;
Gamma(3,3) = 0.0001;
Gamma(4,4) = 0.0001;
Gamma(5,5) = 0.1;
Gamma(6,6) = 0.1;

% Position noise in observation data for v_t's Gaussian
Sigma = zeros(4,4);
Sigma(1,1) = 100;
Sigma(1,2) = 10;
Sigma(2,1) = 10;
Sigma(2,2) = 100;
Sigma(3,3) = 100;
Sigma(3,4) = 10;
Sigma(4,3) = 10;
Sigma(4,4) = 100;

% A is the transition matrix
A = zeros(6,6);
% Update distance based on d_t + v_t*t + 1/2*a_t*t^2
A(1,:) = [1 0 delta_t 0 (0.5 * (delta_t)^2) 0];
A(2,:) = [0 1 0 delta_t 0 (0.5 * (delta_t)^2)];
% Update velocity based on v_t + a_t*t
A(3,:) = [0 0 1 0 delta_t 0];
A(4,:) = [0 0 0 1 0 delta_t];
% Update acceleration based on a_t
A(5,:) = [0 0 0 0 1 0];
A(6,:) = [0 0 0 0 0 1];

% C is the observation matrix
C = zeros(4,6);
% Each observation is derived solely from the position of the UAV at a
% certain time t, plus some Gaussian noise that we'll add in later
```

```
C(1,:) = [1 0 0 0 0 0];
C(2,:) = [0 1 0 0 0 0];
C(3,:) = [1 0 0 0 0 0];
C(4,:) = [0 1 0 0 0 0];


%
% Kalman filter
% Start with an initial state (subject to some noise), then work our way
% through the next t_max timesteps using Kalman equations
%
% t_max is total number of time steps
t_max = 100;
z = zeros(6,t_max);

% Initial state: (0,0)m at velocity (6,6)m/s and acceleration (0,0)m/s^2
% subject to noise V_0 = 6x6 identity matrix
z(:,1) = [0; 0; 6; 6; 0; 0];
% Add on the z_0 = N(z_0 | u_0, V_0) Gaussian noise
z(:,1) = z(:,1) + mvnpdf(z(:,1), 0, eye(6));
% Error covariance matrix of our system
cov_V = eye(6);


% z_{t+1} = A z_t + w_t
% x_t = C z_t + v_t
for i=2:(t_max)

    % Kalman Filter equations from:
    % http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf
    % Predict Step
    temp_z = A * z(:,i-1);
    temp_cov_V = A * cov_V * A' + Gamma;

    % Correct Step
    gain = temp_cov_V * C' / (C * temp_cov_V * C' + Sigma);
    z(:,i) = temp_z + gain*(x(:,i) - C * temp_z);
    cov_V = (eye(6) - gain*C) * temp_cov_V;
end


%
% Plot the sensor data and sequence of posterior distribution means for the
% (x, y)-positions of the UAV.
%
plot(x(1,:),x(2,:),'r:',x(3,:),x(4,:),'r:',z(1,:),z(2,:),'b-');
xlabel('Horizontal Distance (m)');
ylabel('Vertical Distance (m)');
title('Sensor Data and Estimated Position of a Rogue UAV');
```

```
legend('Sensor #1', 'Sensor #2', 'Posterior');


%
% Report UAV's posterior position distribution at t = 100. This is a
% multivariate Gaussian, and can be reported as its mean and 6x6 covariance
% matrix.
%
z(:,t_max)
cov_V
```

Bibliography

[1] S.J. Russell, P. Norvig, J.F. Canny, J.M. Malik, and D.D. Edwards, *Artificial intelligence: a modern approach*, vol. 74, Prentice hall Englewood Cliffs, NJ, 1995.