TU München
Fakultät für Informatik
Dr. Rudolph Triebel
Matthias Vestner

Machine Learning for Robotics and Computer Vision
Winter term 2013

Solution Sheet 3
Topic: Graphical Models
November 29th

Exercise 1:

a)

$$p(A)p(C)p(B \mid A, C)p(D \mid C)p(E \mid B)$$

b) We have:

- $A \perp\!\!\!\perp C \mid \emptyset$,
- $D \perp\!\!\!\perp A, B \mid C$,
- $E \perp\!\!\!\perp A, C, D \mid B$.

c) Algorithm to check, whether X is d-seperated from Y by Z (X,Y,Z sets of nodes):

```
boolean is_dsep(X,Y,Z){
foreach x ∈ X, y ∈ Y
 foreach path p connecting x and y
  if (!is_blocked(p,Z)) return false;
 end;
end;
return true;
}

boolean is_blocked(p,Z){
foreach n ∈ p
 if (type(n) == hh)
  if (n ∉ Z ∧ m ∉ Z  ∀  n→ ... →m )
   return true; //case (b)
  end
 else //type(n) == ht or type(n) == tt
  if (n ∈ Z)
   return true; //case (a)
  end
 end
end
return false;
}
```

- B is d-separated from D by C: true (case (a)),
- A is d-separated from C by E: false (case (b) fails as $D \to E$),
- A is d-separated from C by D: true (case (b)),
- E is d-separated from D by B: true (case (a)),
- E is d-separated from D by A: false.

Exercise 2: Writing a graphical model

a) The observed variables are the map M and the goal G, and the size of the doors O_t.

b) The independence assumptions are

(i) $A_t \perp\!\!\!\perp O_t \mid S_t, M, G$
the action should be independent from the observation conditionally to the location, the map, and the goal. This is the whole point of having a state variable (here the location): abstracting the complexity of the observations into a represention where decision is easier.

(ii) $S_{t+1} \perp\!\!\!\perp O_t, G \mid A_t, S_t, M$
the next location should be independent from the observation and the goal conditionally to the present location, the map, and the action

(iii) $G \perp\!\!\!\perp M \mid \emptyset$
the goal is independent from the map

(iv) $O_t \perp\!\!\!\perp G \mid S_t$
the observation is independent from the goal, conditionally to the location

(v) $S_t \perp\!\!\!\perp M, G \mid \emptyset$
the location is independent from both the goal and the map

(vi) $O_t \perp\!\!\!\perp M \mid S_t$ (optional)
the observation should be independent on the map given the location

c) With the product rule, we have for example:

$$p(S_t, S_{t+1}, A_t, O_t, M, G) = p(G)p(M \mid G)p(S_t \mid M, G)p(O_t \mid S_t, M, G) \quad (1)$$
$$p(A_t \mid O_t, S_t, M, G)p(S_{t+1} \mid A_t, O_t, S_t, M, G) \quad (2)$$

Applying the independence assumptions in the corresponding factors, we deduce the following simplification for the joint probability distribution:

$$p(S_t, S_{t+1}, A_t, O_t, M, G) = p(G)p(M)p(S_t)p(O_t \mid S_t)p(A_t \mid S_t, M, G)p(S_{t+1} \mid A_t, S_t, M)$$

It turns out that the factorization (1) was a good choice as it can be simplified by applying (i)-(vi). Note however that starting from any other factorization one can obtain (1) by applying Bayes rule.

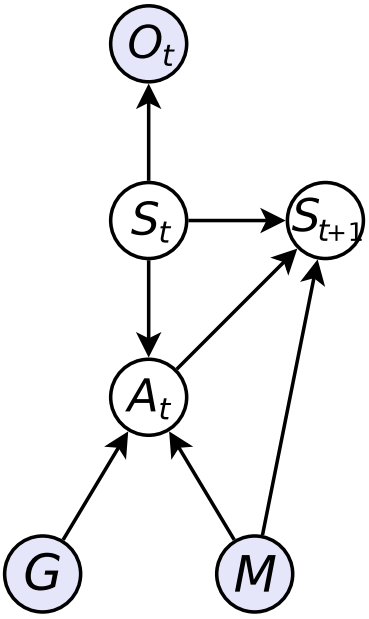

Figure 1: Model Proposal.

d) Figure 1 shows the graphical model corresponding to our joint distribution for which it is a perfect map. Aside from M and G (which are just contextual variable and do not change over time), this model is a Partially Observable Markov Decision Process (POMDP). It is a decision process as we want to issue actions A_t that have influence on the evolution of the state. It is Markovian as the state S_{t+1} does not depend on states earlier than S_t. And its state is partially observable as we don't observe the state variable S_t but only an observation that depends on it.

   For this particular problem, we could have reversed the arrow from S_t to O_t; that is, instead of assuming $S_t \perp\!\!\!\perp M, G \mid \emptyset$ and $O_t \perp\!\!\!\perp M, G \mid S_t$, we could have assumed $O_t \perp\!\!\!\perp M, G \mid$ and $S_t \perp\!\!\!\perp M, G \mid O_t$. However the form we've chosen is more common, as it's often easier to specify what should see a sensor in a particular situation (sensor model) than the inverse.

Exercise 3: Markov Chain

a)

$$p(A,B,C,D) = \frac{1}{Z}\psi_{A,B}(A,B)\psi_{B,C}(B,C)\psi_{C,D}(C,D)$$

b) All three potential functions are the same:

V_1 \ V_2	0	1
0	9	1
1	1	9

Notice that the values need not be normalized in any way.

c) μ_α and μ_β can be calculated recursively:

$$\mu_\alpha(x_n) = \sum_{x_{n-1}} \psi_{n-1,n}(x_{n-1}, x_n)\mu_\alpha(x_{n-1})$$
$$\mu_\beta(x_n) = \sum_{x_{n+1}} \psi_{n,n+1}(x_n, x_{n+1})\mu_\beta(x_{n+1})$$

With our potentials this yields:

- $\mu_\alpha(A) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ (initialization, optional),
- $\mu_\alpha(B) = \begin{pmatrix} \sum_A \mu_\alpha(A)\psi_{A,B}(A,0) \\ \sum_A \mu_\alpha(A)\psi_{A,B}(A,1) \end{pmatrix} = \begin{pmatrix} 1\times 9 + 1\times 1 \\ 1\times 1 + 1\times 9 \end{pmatrix} = \begin{pmatrix} 10 \\ 10 \end{pmatrix}$
- $\mu_\alpha(C) = \begin{pmatrix} \sum_B \mu_\alpha(B)\psi_{B,C}(B,0) \\ \sum_B \mu_\alpha(B)\psi_{B,C}(B,1) \end{pmatrix} = \begin{pmatrix} 10\times 9 + 10\times 1 \\ 10\times 1 + 10\times 9 \end{pmatrix} = \begin{pmatrix} 100 \\ 100 \end{pmatrix}$
- $\mu_\alpha(D) = \begin{pmatrix} \sum_C \mu_\alpha(C)\psi_{C,D}(C,0) \\ \sum_C \mu_\alpha(C)\psi_{C,D}(C,1) \end{pmatrix} = \begin{pmatrix} 100\times 9 + 100\times 1 \\ 100\times 1 + 100\times 9 \end{pmatrix} = \begin{pmatrix} 1000 \\ 1000 \end{pmatrix}$
- $\mu_\beta(D) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ (initialization, optional),
- $\mu_\beta(C) = \begin{pmatrix} \sum_D \mu_\beta(D)\psi_{C,D}(0,D) \\ \sum_D \mu_\beta(D)\psi_{C,D}(1,D) \end{pmatrix} = \begin{pmatrix} 1\times 9 + 1\times 1 \\ 1\times 1 + 1\times 9 \end{pmatrix} = \begin{pmatrix} 10 \\ 10 \end{pmatrix}$
- $\mu_\beta(B) = \begin{pmatrix} \sum_C \mu_\beta(C)\psi_{B,C}(0,C) \\ \sum_C \mu_\beta(C)\psi_{B,C}(1,C) \end{pmatrix} = \begin{pmatrix} 10\times 9 + 10\times 1 \\ 10\times 1 + 10\times 9 \end{pmatrix} = \begin{pmatrix} 100 \\ 100 \end{pmatrix}$
- $\mu_\beta(A) = \begin{pmatrix} \sum_B \mu_\beta(B)\psi_{A,B}(0,B) \\ \sum_B \mu_\beta(B)\psi_{A,B}(1,B) \end{pmatrix} = \begin{pmatrix} 100\times 9 + 100\times 1 \\ 100\times 1 + 100\times 9 \end{pmatrix} = \begin{pmatrix} 1000 \\ 1000 \end{pmatrix}$

Then we compute normalization factor Z at any point, for example B:

$$Z = \sum_B \mu_\alpha(B).\mu_\beta(B) = 2000$$

Finally we can compute the marginal distributions requested:

$$p(A) = \frac{1}{Z}.\mu_\alpha(A).\mu_\beta(A) = \frac{1}{2000}\begin{pmatrix} 1 \times 1000 \\ 1 \times 1000 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}$$

$$p(C) = \frac{1}{Z}.\mu_\alpha(C).\mu_\beta(C) = \frac{1}{2000}\begin{pmatrix} 100 \times 10 \\ 100 \times 10 \end{pmatrix} = \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}$$

The assumptions were only that neighboring nodes should be equal. The marginal on A and C both say that we have no idea on their value. That was to be expected.

d) We've learned that we could compute marginal distributions by decomposing the inference into messages to be passed between nodes.

How can we adapt this mecanism to observations?

If the chain contained only C and D, we would have:

$$p(C \mid [D = 1]) = \frac{1}{Z'}\begin{pmatrix} \psi_{C,D}(0,1) \\ \psi_{C,D}(1,1) \end{pmatrix}$$

This can be written in the same message passing form:

$$p(C \mid [D = 1]) = \frac{1}{Z'}\begin{pmatrix} 1 \\ 1 \end{pmatrix} . \begin{pmatrix} \sum_D \mu'_\beta(D)\psi_{C,D}(0,D) \\ \sum_D \mu'_\beta(D)\psi_{C,D}(1,D) \end{pmatrix}$$

with $\mu'_\beta(D) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$.

Actually, you just have to replace the $\mu_*(X)$ with a Dirac in order to account for an observation of the value of X (and recompute the normalization factor):

- $\mu_\alpha(A) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$,
- $\mu_\alpha(B) = \begin{pmatrix} \sum_A \mu_\alpha(A)\psi_{A,B}(A,0) \\ \sum_A \mu_\alpha(A)\psi_{A,B}(A,1) \end{pmatrix} = \begin{pmatrix} 1 \times 9 + 1 \times 1 \\ 1 \times 1 + 1 \times 9 \end{pmatrix} = \begin{pmatrix} 10 \\ 10 \end{pmatrix}$
- $\mu_\alpha(C) = \begin{pmatrix} \sum_B \mu_\alpha(B)\psi_{B,C}(B,0) \\ \sum_B \mu_\alpha(B)\psi_{B,C}(B,1) \end{pmatrix} = \begin{pmatrix} 10 \times 9 + 10 \times 1 \\ 10 \times 1 + 10 \times 9 \end{pmatrix} = \begin{pmatrix} 100 \\ 100 \end{pmatrix}$
- $\mu_\alpha(D) = \begin{pmatrix} \sum_C \mu_\alpha(C)\psi_{C,D}(C,0) \\ \sum_C \mu_\alpha(C)\psi_{C,D}(C,1) \end{pmatrix} = \begin{pmatrix} 100 \times 9 + 100 \times 1 \\ 100 \times 1 + 100 \times 9 \end{pmatrix} = \begin{pmatrix} 1000 \\ 1000 \end{pmatrix}$
- $\mu'_\beta(D) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ (observation),
- $\mu'_\beta(C) = \begin{pmatrix} \sum_D \mu'_\beta(D)\psi_{C,D}(0,D) \\ \sum_D \mu'_\beta(D)\psi_{C,D}(1,D) \end{pmatrix} = \begin{pmatrix} 0 \times 9 + 1 \times 1 \\ 1 \times 1 + 0 \times 9 \end{pmatrix} = \begin{pmatrix} 1 \\ 9 \end{pmatrix}$
- $\mu'_\beta(B) = \begin{pmatrix} \sum_C \mu'_\beta(C)\psi_{B,C}(0,C) \\ \sum_C \mu'_\beta(C)\psi_{B,C}(1,C) \end{pmatrix} = \begin{pmatrix} 1 \times 9 + 9 \times 1 \\ 1 \times 1 + 9 \times 9 \end{pmatrix} = \begin{pmatrix} 18 \\ 82 \end{pmatrix}$
- $\mu'_\beta(A) = \begin{pmatrix} \sum_B \mu'_\beta(B)\psi_{A,B}(0,B) \\ \sum_B \mu'_\beta(B)\psi_{A,B}(1,B) \end{pmatrix} = \begin{pmatrix} 18 \times 9 + 82 \times 1 \\ 18 \times 1 + 82 \times 9 \end{pmatrix} = \begin{pmatrix} 244 \\ 756 \end{pmatrix}$

As above, we can also compute $Z' = 1000$ and then:

$$p(A \mid [D=1]) = \frac{1}{Z'}.\mu_\alpha(A).\mu'_\beta(A) = \frac{1}{1000}\begin{pmatrix} 1 \times 244 \\ 1 \times 756 \end{pmatrix} = \begin{pmatrix} 0.244 \\ 0.756 \end{pmatrix}$$

$$p(C \mid [D=1]) = \frac{1}{Z'}.\mu_\alpha(C).\mu'_\beta(C) = \frac{1}{1000}\begin{pmatrix} 100 \times 1 \\ 100 \times 9 \end{pmatrix} = \begin{pmatrix} 0.1 \\ 0.9 \end{pmatrix}$$

Now, we know that node D equals 1 and we see it has become more probable for A and C to be equal to 1 (the more for C which is nearer D than A). At least the result makes sense.

e) With the same way, we can recompute μ'_α (which is symmetric to μ_β):

- $\mu'_\alpha(A) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$,
- $\mu'_\alpha(B) = \begin{pmatrix} \sum_A \mu'_\alpha(A)\psi_{A,B}(A,0) \\ \sum_A \mu'_\alpha(A)\psi_{A,B}(A,1) \end{pmatrix} = \begin{pmatrix} 1 \times 9 + 0 \times 1 \\ 1 \times 1 + 0 \times 9 \end{pmatrix} = \begin{pmatrix} 9 \\ 1 \end{pmatrix}$
- $\mu'_\alpha(C) = \begin{pmatrix} \sum_B \mu'_\alpha(B)\psi_{B,C}(B,0) \\ \sum_B \mu'_\alpha(B)\psi_{B,C}(B,1) \end{pmatrix} = \begin{pmatrix} 9 \times 9 + 1 \times 1 \\ 9 \times 1 + 1 \times 9 \end{pmatrix} = \begin{pmatrix} 82 \\ 18 \end{pmatrix}$
- $\mu'_\alpha(D) = \begin{pmatrix} \sum_C \mu'_\alpha(C)\psi_{C,D}(C,0) \\ \sum_C \mu'_\alpha(C)\psi_{C,D}(C,1) \end{pmatrix} = \begin{pmatrix} 82 \times 9 + 18 \times 1 \\ 82 \times 1 + 18 \times 9 \end{pmatrix} = \begin{pmatrix} 756 \\ 244 \end{pmatrix}$

Now $Z" = 244$ and:

$$p(C \mid [D=1]) = \frac{1}{Z"}.\mu'_\alpha(C).\mu'_\beta(C) = \frac{1}{244}\begin{pmatrix} 82 \times 1 \\ 18 \times 9 \end{pmatrix} \approx \begin{pmatrix} 0.336 \\ 0.664 \end{pmatrix}$$

It would be the reverse for B: $\begin{pmatrix} 0.664 \\ 0.336 \end{pmatrix}$. It is not exactly $\frac{2}{3}$. Actually, with a longer chain, both μ'_α and μ'_β would (exponentially) converge to uniforms as we consider node further from their origin. Therefore for a long chain, the probability will come from $\begin{pmatrix} 100\% \\ 0\% \end{pmatrix}$ to rest at the uniform $\begin{pmatrix} 50\% \\ 50\% \end{pmatrix}$ before setting to $\begin{pmatrix} 0\% \\ 100\% \end{pmatrix}$. In order to "straighten" the values, we could lower the probability of being different from neighboring nodes.

Note that the known nodes are at the boundary of our chain. If it was not the case, the d-separation property would have allowed us to split the chain in two independent subchains having both a copy of the observed variable as the new boundary.