Machine Learning 2 - Homework 4

Pascal M. Esser

April 30, 2018

Collaborators: Sindy Löwe, Andrew Sklyar

1

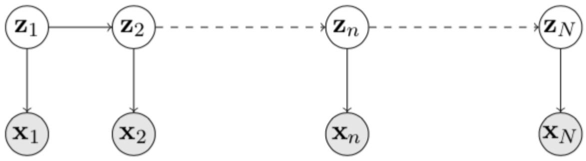

Figure 1

Given the Bayesian network in Figure 1, $\boldsymbol{X} = x_1, ..., x_N$ and $\boldsymbol{Z} = z_1, ..., z_N$:

1.1

Write down the factorized joint probability distribution $p(\boldsymbol{Z}, \boldsymbol{X})$.

$$p(\boldsymbol{Z}, \boldsymbol{X}) = p(z_1)p(x_1|z_1)\prod_{i=2}^{N} p(x_i|z_i)p(z_i|z_{i-1}) \tag{1}$$

1.2

Draw the the corresponding factor graph. See Figure 2

1.3

Write down the the joint probability distribution using the factors introduced in 2.

$$p(\boldsymbol{Z}, \boldsymbol{X}) = \alpha_1(z_1)\prod_{i=2}^{N} \alpha_i(z_i, z_{i-1})\prod_{j=1}^{N} \beta_i(x_i, z_i) \tag{2}$$

1.4

Given $\boldsymbol{X}$, we want to infer z_n such that

$$p(z_n|\boldsymbol{X}) = \frac{p(\boldsymbol{X}|z_n)p(z_n)}{p(\boldsymbol{X})} \tag{3}$$

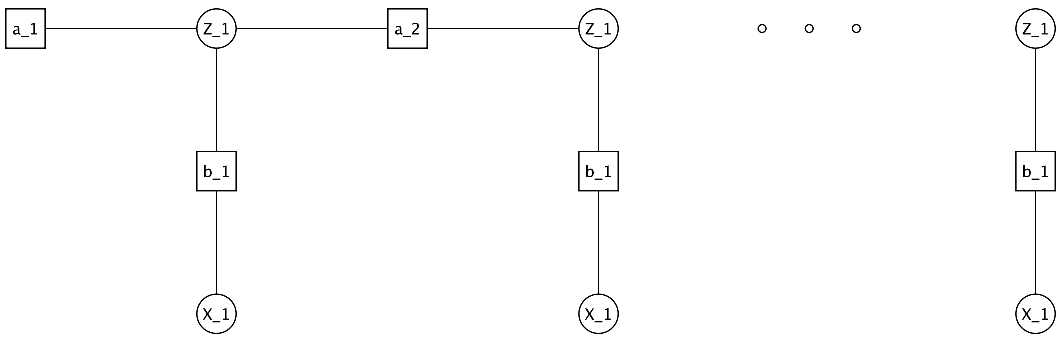

Figure 2: In the figure above we write $a\_i \equiv \alpha_i, b\_i \equiv \beta_i ...$ for all variables. Furthermore we define α and β to be:
$\alpha(z_n) = p(x_1, ..., x_N, z_n) = p(x_1, ..., x_n|z_n)p(z_n)$
$\beta(z_n) = p(x_{n+1}, ..., x_N|z_n)$

Using the conditional independencies of the graph in Figure 1, derive $\alpha(z_n)$ and $\beta(z_n)$ so that they are recursive definitions of themselves, i.e. $\alpha(z_n)$is calculated from $\alpha(z_{n-1})$ and $\beta(z_n)$ is calculated from $\beta(z_{n+1})$. Indicate where you use independencies inferred from the graphical model.

Deriving α and β:

$$
\begin{aligned}
p(X|z_n)p(z_n) =& p(x_1, ..., x_N|z_n)p(z_n) && (4)\\
=& p(x_1, ..., x_n|z_n)p(x_{n+1}, ..., x_N|z_n)p(z_n) && (5)\\
=& p(x_1, ..., x_N, z_n)p(x_{n+1}, ..., x_N|z_n) && (6)\\
=& \alpha(z_n)\beta(z_n) && (7)
\end{aligned}
$$

Rewrite α, where we use $z_{n-1} \perp\!\!\!\perp x_n|z_n$,$x_1, ..., x_{n-1} \perp\!\!\!\perp x_n|z_n$ and $x_{n-1} \perp\!\!\!\perp z_n|z_{n-1}$ from the graph.

as we write $\alpha(z_n)$ in therms of $\alpha(z_{n-1})$ the following holds for $n \geq 2$

$$
\begin{aligned}
\alpha(z_n) =& p(x_1, ..., x_N, z_n) && (8)\\
=& p(x_1, ..., x_n|z_n)p(z_n) && (9)\\
=& p(x_1, ..., x_{n-1}|z_n)p(x_n|z_n)p(z_n) && (10)\\
=& p(x_1, ..., x_{n-1}, z_n)p(x_n|z_n) && (11)\\
=& \sum_{z_{n-1}} p(x_1, ..., x_{n-1}, z_n|z_{n-1})p(z_{n-1})p(x_n|z_n) && (12)\\
=& \sum_{z_{n-1}} p(x_1, ..., x_{n-1}|z_{n-1})p(z_n|z_{n-1})p(z_{n-1})p(x_n|z_n) && (13)\\
=& \sum_{z_{n-1}} \alpha(z_{n-1})p(z_n|z_{n-1})p(x_n|z_n) && (14)\\
& && (15)
\end{aligned}
$$

Rewrite β where we use $x_N \perp\!\!\!\perp z_{N-2}|z_{N-1}$ and $x_N \perp\!\!\!\perp z_{N-1}|z_{N-1}$ from the graph.

as we write $\beta(z_n)$ in therms of $\beta(z_{n+1})$ the following holds for $n < N$

$$\beta(z_n) = p(x_{n+1}, ..., x_N | z_n) \tag{16}$$
$$= \frac{p(x_{n+1}, ..., x_N, z_n)}{p(z_n)} \tag{17}$$
$$= \sum_{z_{n+1}} \frac{p(x_{n+1}, ..., x_N, z_n | z_{n+1}) p(z_{n+1})}{p(z_n)} \tag{18}$$
$$= \sum_{z_{n+1}} \frac{p(z_n, x_{n+1} | z_{n+1}) p(x_{n+2}, ..., x_N | z_n + 1) p(z_{n+1})}{p(z_n)} \tag{19}$$
$$= \sum_{z_{n+1}} p(z_{n+1}, x_{n+1} | z_n) \beta(z_{n+1}) \tag{20}$$

2

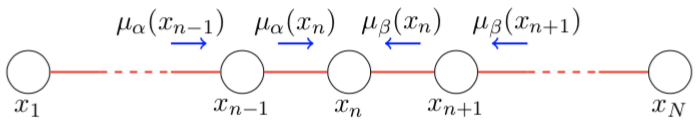

Figure 3

2.1

Apply the sum-product algorithm (as in Bishop's section 8.4.4) to the chain of nodes model in Figure 3 and show that the results of message passing algorithm (as in Bishop's section 8.4.1) are recovered as a special case, that is

$$p(x_n) = \frac{1}{Z} \mu_{x_i \to f_s(x_i)} \alpha(x_n) \mu_{x_i \to f_s(x_i)} \beta(x_n) \tag{21}$$
$$\mu_\alpha(x_n) = \sum_{x_n - 1} \psi_{n-1,n}(x_{n-1}, x_n) \mu_\alpha(x_{n-1}) \tag{22}$$
$$\mu_\beta(x_n) = \sum_{x_n + 1} \psi_{n+1,n}(x_{n-1}, x_n) \mu_\beta(x_{n+1}) \tag{23}$$

where $\psi_{i,i+1}(x_i, x_{i+1})$ is a potential function defined over clique $\{x_i, x_{i+1}\}$.

We can write Figure 3 as a factor graph as show in Figure 4.

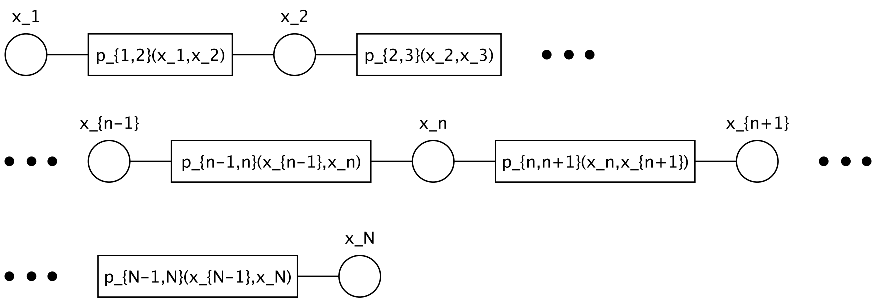

Figure 4: In the figure above we write $p\_i \equiv \psi_i, b\_i \equiv \beta_i ...$ for all variables.

Setting up the sum-product algorithm with root x_n and start propagation from x_1:

$$\mu_{x_1 \to \psi_{1,2}}(x_1) = 1 \tag{24}$$
$$\mu_{\psi_{1,2} \to x_2}(x_2) = \sum_{x_1} (x_1, x_2) \tag{25}$$
$$\mu_{x_2 \to \psi_{2,3}}(x_2) = \mu_{\psi_{1,2} \to x_2}(x_2) \tag{26}$$
$$\mu_{\psi_{2,3} \to x_3}(x_3) = \sum_{x_3} (x_2, x_3) \tag{27}$$
$$\dots \tag{28}$$
$$\mu_{x_{n-1} \to \psi_{n-1,n}}(x_{n-1}) = \mu_{\psi_{n-2,n-1} \to x_{n-1}}(x_{n-1}) \tag{29}$$
$$\mu_{\psi_{n-1,n} \to x_n}(x_n) = \sum_{x_{n-1}} \psi_{n-1,n}(x_{n-1}, x_n) \mu_{x_{n-1} \to \psi_{n-1,n}}(x_{n-1}) \tag{30}$$
$$\tag{31}$$

show that $\mu_\alpha(x_n) = \sum_{x_n - 1} \psi_{n-1,n}(x_{n-1}, x_n) \mu_\alpha(x_{n-1})$ holds:

$$\mu_\alpha(x_n) = \mu_{\psi_{n-1,n} \to x_n}(x_n) \tag{32}$$
$$= \sum_{x_{n-1}} \psi_{n-1,n}(x_{n-1}, x_n) \mu_{x_{n-1} \to \psi_{n-1,n}}(x_{n-1}) \tag{33}$$
$$= \sum_{x_{n-1}} \psi_{n-1,n}(x_{n-1}, x_n) \mu_{\psi_{n-2,n-1} \to x_{n-1}}(x_{n-1}) \tag{34}$$
$$= \sum_{x_n - 1} \psi_{n-1,n}(x_{n-1}, x_n) \mu_\alpha(x_{n-1}) \tag{35}$$

show that $\mu_\beta(x_n) = \sum_{x_n + 1} \psi_{n+1,n}(x_{n-1}, x_n) \mu_\beta(x_{n+1})$ holds:

$$\mu_\beta(x_n) = \mu_{\psi_{n+1,n} \to x_n}(x_n) \tag{36}$$
$$= \sum_{x_{n+1}} \psi_{n+1,n}(x_{n+1}, x_n) \mu_{x_{n+1} \to \psi_{n+1,n}}(x_{n+1}) \tag{37}$$
$$= \sum_{x_{n+1}} \psi_{n+1,n}(x_{n+1}, x_n) \mu_{\psi_{n+1,n+2} \to x_{n+1}}(x_{n+1}) \tag{38}$$
$$= \sum_{x_n + 1} \psi_{n+1,n}(x_{n+1}, x_n) \mu_\beta(x_{n+1}) \tag{39}$$

show that $p(x_n) = \frac{1}{Z} \mu_{x_i \to f_s(x_i)} \alpha(x_n) \mu_{x_i \to f_s(x_i)} \beta(x_n)$ holds

$$p(x_n) = \frac{1}{Z} \mu_{\psi_{n-1,n} \to x_n}(x_n) \mu_{\psi_{n+1,n} \to x_n}(x_n) \tag{40}$$
$$= \frac{1}{Z} \mu_\alpha(x_n) \mu_\beta(x_n) \tag{41}$$

2.2

Establish a relation of your results $\alpha(z_n)$ and $\beta(z_n)$ in 1.4 with the results of the sum-product algorithm $\mu_\alpha(x_n)$ and $\mu_\beta(x_n)$.

starting from the results obtained above we can write for $\alpha(z_n)$ and$\mu_\alpha(x_n)$:

$$\alpha(z_n) = \sum_{z_{n-1}} \alpha(z_{n-1})p(z_n|z_{n-1})p(x_n|z_n) \tag{42}$$
$$\mu_{\alpha_{n-1}\to z_n}(z_n) = \sum_{x_n-1} \alpha_{n-1}(z_{n-1}, z_n)\mu_\alpha(z_{n-1}) \tag{43}$$
$$\tag{44}$$

which gives us the following correspondences (expressed as $\rightarrow$)

$$\alpha(z_n) \rightarrow \mu_{\alpha_{n-1}\to z_n}(z_n) \tag{45}$$
$$\alpha(z_{n-1}) \rightarrow \mu_\alpha(z_{n-1}) \tag{46}$$
$$p(z_n|z_{n-1}) \rightarrow \alpha_{n-1}(z_{n-1}, z_n) \tag{47}$$

because the models differ we have $p(x_n|z_n)$ not in the model and therefore we don't have a corresponding mapping.

Similar we can write for $\beta(z_n)$ and $\mu_\beta(x_n)$:

$$\beta(z_n) = \sum_{z_{n+1}} \beta(z_{n+1})p(z_n|z_{n+1})p(x_n|z_n) \tag{48}$$
$$\mu_{\beta_{n+1}\to z_n}(z_n) = \sum_{x_n+1} \beta_{n+1}(z_{n+1}, z_n)\mu_\beta(z_{n+1}) \tag{49}$$
$$\tag{50}$$

which gives us the following correspondences (expressed as $\rightarrow$)

$$\beta(z_n) \rightarrow \mu_{\beta n+1\to z_n}(z_n) \tag{51}$$
$$\beta(z_{n+1}) \rightarrow \mu_\beta(z_{n+1})) \tag{52}$$
$$p(z_n|z_{n+1}) \rightarrow \beta_{n+1}(z_{n+1}, z_n) \tag{53}$$

again as we don't have additional terms for for x_n in the model of $\mu_{\beta_{n+1}\to z_n}(z_n)$, $p(x_n|z_n)$ falls out and has no corresponding term.

3

Consider the inference problem of evaluating $p(\boldsymbol{x}_n|\boldsymbol{x}_N)$ for the graph shown in Figure 3, for all nodes $n \in \{1, ..., N-1\}$. Show that the message passing algorithm can be used to solve this efficiently, and discuss which messages are modified and in what way.

We start from a setting where we treat $\boldsymbol{x}_N$ as an observed variable, which means that only $\psi(n_{N-1}, x_N)$ is dependent on $\boldsymbol{x}_N$. To introduce the desired dependency relationship on $\boldsymbol{x}_N$ we set: $p(\boldsymbol{x}_n|\boldsymbol{x}_N) = p(\boldsymbol{x}_n, \boldsymbol{x}_N)\mathbb{I}[\boldsymbol{x}_N = \xi]$ where $\mathbb{I}[\boldsymbol{x}_N = \xi]$ is an indicator function with $\mathbb{I}[\boldsymbol{x}_N = \xi] = \begin{cases} 1 & \textbf{iff } x_N = \xi \\ 0 & \textbf{else} \end{cases}$

This way $\boldsymbol{x}_N$ is treated as unobserved variable and the massage passing algorithm can be applied without further restrictions.

As the algorithm is applied iteratively, all messages going from the leaf node x_N towards the root node will change accordingly with the new introduced indicator function.Messages that are sent from the leaf node x_1 to the root node remain unchanged. In the backward pass we see the revers situation: from the root to x_1 the indicator function is taken included and from the root to x_N the indicator function is not included as the message originated from x_1 which does not include the indicator function.

4

Show that the marginal distribution for the variables $\boldsymbol{x}_s$ in a factor $f_s(\boldsymbol{x}_s)$ in a tree-structured factor graph, after running the sum-product message passing algorithm, can be written as

$$p(\boldsymbol{x}_s) = f_s(_s) \prod_{i \in ne(f_s)} \mu_{x_i \to f_s(x_i)} \tag{54}$$

where $ne(f_s)$ denotes the set of variable nodes that are neighbors of the factor node f_s

For this task we will use results and notations from Bishop [1] chapter 8.4. In the following we donate: $F_t(x_i, \boldsymbol{X}_t)$ to represents the product of all the factors in the group associated with factor f_s

$$\text{starting from the definition of the joint} \tag{55}$$

$$p(\boldsymbol{x}) = f_s(\boldsymbol{x}_s) \prod_{i \in ne(f_s)} \prod_{t \in ne(x_i) \backslash f_s} F_t(x_i, \boldsymbol{X}_t) \tag{56}$$

$$\text{use definiton of } p(\boldsymbol{x}_s) \text{which is given as follows:} \tag{57}$$

$$p(\boldsymbol{x}_s) = \sum_{\boldsymbol{x} \backslash x_s} p(\boldsymbol{x}) \tag{58}$$

$$\text{now put } p(\boldsymbol{x}) \text{ into the definition of } p(\boldsymbol{x}_s) \text{ which is given as follows:} \tag{59}$$

$$= \sum_{\boldsymbol{x} \backslash x_s} f_s(\boldsymbol{x}_s) \prod_{i \in ne(f_s)} \prod_{s \in ne(x_i) \backslash f_s} F_s(x_i, \boldsymbol{X}_s) \tag{60}$$

$$\text{push the sum in} \tag{61}$$

$$= f_s(\boldsymbol{x}_s) \prod_{i \in ne(f_s)} \prod_{s \in ne(x_i) \backslash f_s} \left[\sum_{x_s} F_s(x_i, \boldsymbol{X}_s) \right] \tag{62}$$

$$\text{use the definition of } \mu_{f_s \to x_i}(x_i) \text{ with } \mu_{f_s \to x_i}(x_i) \equiv \sum_{x_s} F_s(x_i, \boldsymbol{X}_s) \tag{63}$$

$$= f_s(\boldsymbol{x}_s) \prod_{i \in ne(f_s)} \prod_{s \in ne(x_i) \backslash f_s} \mu_{f_s \to x_i}(x_i) \tag{64}$$

$$= f_s(\boldsymbol{x}_s) \prod_{i \in ne(f_s)} \mu_{x_i \to f_s}(x_i) \tag{65}$$

References

[1] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics).* Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006. ISBN: 0387310738.