

# Assignment 2

## Statistical Methods in Applied Computer Science

### DD2447

Jim Holmström 890503-7571  
[jimho@kth.se](mailto:jimho@kth.se)

November 25, 2012

**Exercise A1** Compute  $p(D|T \in \text{Polytree})$  with Bernoulli CPD's  
Show how to compute  $p(D|T)$  where  $T$  is a DGM which is a polytree and  $D = \{x_1, \dots, x_N\}$  (and  $x_i$  is an assignment of values to all variables of  $T$ ). Assume that all variables are binary and all CPD's Bernoulli.

*Solution.*

*Theorem 0.1.* Chain rule

$$p(\bigcap_k x_k) = \prod_k p(x_k | \bigcap_{j:j < k} x_j) \quad (1)$$

*Proof.* We can extend the rule

$$p(x_i, x_j) = p(x_i | x_j) p(x_j) \quad (2)$$

since  $x_i, x_j$  is just events we might as well have  $x_j = \bigcap_{j'} x_{j'}$  which makes the expression look like this

$$p(x_i \cap (\bigcap_{j'} x_{j'})) = p(x_i | \bigcap_{j'} x_{j'}) p(\bigcap_{j'} x_{j'}) \quad (3)$$

by tail-recursion on  $p(\bigcap_{j'} x_{j'}) = p(x_a \cap (\bigcap_{j' \neq a} x_{j'}))$  we will exhaust all  $j'$  which leaves us with the base-case  $p(x_a)$ . Writing out the entire trace of the recursion will give us the wanted expression for  $p(\bigcap_k x_k)$ .  $\square$

The first goal is to break down  $p(D|T)$  into the CPD's for each node

$$p(D|T) = \phi(\{p(x_t|\cdot)\}_t) \quad (4)$$

this gives an explicit expression for the  $p(D|T)$ .

Since  $T$  is a polytree  $\subset$  DAG it is always possible to sort the data topologically according to  $T$ .<sup>1</sup> We arrange the indices of the data this way so that it will be topologically sorted according to  $T$  and even if it's bad practice in math we replace the old indices with the new sorted ones. This results in variables always having the property  $i < j \Rightarrow i$  higher up or equally high as  $j$ .<sup>2</sup>

Now apply the chain rule

$$p(D|T) = p(\{x_t\}_k|T) = \prod_k p(x_k | \{x_j\}_{j:j < k}, T) \quad (5)$$

Next we note that since we have that the data is topologically sorted we know that

$$(D \setminus \{x_j\}_{j:j < k}) \cap pa(x_k) = \emptyset \quad (6)$$

in other words the parent cannot be at the same or lower topological level. This results in

$$\{x_j\}_{j:j < k} \supset pa(x_k) \quad (7)$$

which basically states that the parents, denoted  $pa(\cdot)$ , always is at a higher level.

The next step is to use the conditional independence information from  $T$ , which is that a r.v. is only dependent on the parents r.v.'s<sup>3</sup>, together with the fact (??) in (??) we get

$$p(D|T) = \prod_k p(x_k | \{x_j\}_{j:j < k}, T) = \prod_k p(x_k | pa(x_k)) \quad (8)$$

Finally since we only have cpt's<sup>4</sup>  $f_t$ <sup>5</sup>, we have that

$$x_t \sim Ber(f_t(pa(x_t))) \quad (9)$$

note that if  $pa(x_t) = \emptyset$  then  $f_t$  will simply define with one row the probability for  $x_t$  directly. Now as explicitly written out as possible

$$p(D|T) = \prod_k p(x_k | pa(x_k)) = \prod_k f_k(pa(x_k))^{x_k} (1 - f_k(pa(x_k)))^{1-x_k} \quad (10)$$

□

---

<sup>1</sup>How we sort the data within a topological level doesn't matter.

<sup>2</sup>"higher" in the context of topological sorting means that it is higher up in the sorted DAG that grows downwards.

<sup>3</sup>At least for a completely visible graph.

<sup>4</sup>According to mail correspondence with the teacher.

<sup>5</sup>Since each distribution has only two states and needs to be normalized to 1, this is the same thing as setting  $p \in (0, 1)$  in a bernoulli

**Exercise A2** Marginalize over non-observed variables

Assume instead that each  $x_i$  is an assignment to a subset of the variables say  $O$ . Show how to marginalize over  $V \setminus O$  (i.e., the non-observed variables).

*Solution.* We denote the observed variables with  $x_v$  and the hidden with  $x_h$ , both are generally vectors. By general marginalization <sup>6</sup> we have the probability of the observed data

$$p(x_v|T) = \sum_{h'} p(x_v, x_{h'}|T) \quad (11)$$

**Exercise 11.3** EM for the mixtures of Bernoullis

- Show that the M step for ML estimation of a mixture of Bernoullis is given by

$$\mu_{kj} = \frac{\sum_i r_{ik} x_{ij}}{\sum_i r_{ik}} \quad (12)$$

- Show that the M step for MAP estimation of a mixture of Bernoullis with a  $\beta(\alpha, \beta)$  prior is given by

$$\mu_{kj} = \frac{(\sum_i r_{ik} x_{ij}) + \alpha - 1}{(\sum_i r_{ik}) + \alpha + \beta - 2} \quad (13)$$

*Solution.* The M step is to optimize the auxiliary function  $Q$  with respect to  $\pi, \theta'$ .  $Q$  is the expected posterior log-likelihood<sup>7</sup> with respect to the last parameter  $\theta$  and the observed data  $D$ .<sup>8</sup> The expression for this is

$$Q(\theta', \theta) = E [\log \mathcal{L}_{MAP}(\theta') | D, \theta] = E [\log (\mathcal{L}(\theta') p(\theta')) | D, \theta] = \quad (14)$$

$$= E [\ell(\theta') + \log p(\theta') | D, \theta] = E [\ell(\theta') | D, \theta] + \log p(\theta') \quad (15)$$

and to derive this expression we introducing the latent variable  $z_i$  which corresponds to the hidden or missing variables which basically is the *r.v.* for how  $x_i$  belongs to the class  $k$ .<sup>9</sup> We start with MAP and then derive ML by setting a uniform priori.

$$Q(\theta', \theta) = E \left[ \sum_i \log p(x_i, z_i | \theta') \right] + \log p(\theta') = \quad (16)$$

since  $E$  is linear and we can factor on the different classes and the factors become given with the parameters  $(\pi_k, \theta'_k)$  of class  $k$ :

$$= \sum_i \left[ E \log \left( \prod_k (\pi_k p(x_i | \theta'_k))^{\mathbb{I}(z_i=k)} \right) \right] + \log \prod_k p(\theta'_k) = \quad (17)$$

<sup>6</sup> $p(A) = \sum_{b \in \Omega_B} p(A, B = b)$

<sup>7</sup>In the book it's actually just log-likelihood, but this will work in the same way instead of using  $Q(\theta', \theta) + \log p(\theta')$  without derivation, we will use posterior-Q as  $Q$  instead.

<sup>8</sup>It can be shown that  $Q$  is so that the new parameters is always better or as good as the last one, but exclude the proof for this since it's not needed by the exercise.

<sup>9</sup>Responsibility  $r_{ik} \triangleq p(z_i = k | x_i, \theta)$

then log the inner parts and let E operate on the expression

$$= \sum_i \sum_k \left[ \mathbb{E} [\mathbb{I}(z_i = k)] \log [p(x_i | \theta'_k)] \right] + \sum_k \log p(\theta'_k) = \quad (18)$$

then we have that the expected  $\mathbb{E} [\mathbb{I}(z_i = k)]$  will be  $p(z_i = k | x_i, \theta) = r_{ik}$  which is the expected class-belonging of  $x_i$  given the previous parameters  $\theta$ . The log product rule gives us

$$= \sum_i \sum_k r_{ik} \log \pi_k + \sum_i \sum_k \{r_{ik} \log p(x_i | \theta'_k)\} + \sum_k \log p(\theta'_k) \quad (19)$$

Now since we have that  $\sum_{i,k} r_{ik} \log \pi_k \perp \sum_{i,k} \{r_{ik} \log p(x_i | \theta'_k)\} + \sum_k \log p(\theta'_k)$  we can optimize the parameters  $\pi_k$  and  $\theta'_k$  separately and only the  $\theta'_k$ -term is asked for in the exercise which we denote  $\ell(\theta'_k)$ .

The model parameters in this case is denoted by  $\mu_k$  which is a vector and in index-notation  $\mu_{kj}$ .<sup>10</sup>

We start with the MAP

$$\hat{\mu}'_k = \underset{\mu'_k}{\operatorname{argmax}} \left\{ \sum_i \{r_{ik} \log p(x_i | \mu'_k)\} + \log p(\mu'_k) \right\} \quad (20)$$

which we find by  $\frac{\partial \ell(\mu'_k)}{\partial \mu'_{kj}} = 0$ , where  $p(x_i | \mu'_k) = \prod_j p(x_{ij} | \mu'_{kj})$  is the multivariate Bernoulli<sup>11</sup> distribution for class  $k$ . The priori in the same way is  $p(\mu'_k) = \prod_j p(\mu'_{kj})$ . In our case for MAP this is  $\beta(\alpha, \beta) = \frac{\mu'_{kj}{}^{\alpha-1} (1-\mu'_{kj})^{\beta-1}}{B(\alpha, \beta)}$ .

---

<sup>10</sup>Not using Einstein notation for tensor product.

<sup>11</sup>Often called multinoulli.

$$\frac{\partial(\ell(\mu'_k))}{\partial\mu'_{kj}} = \frac{\partial\sum_i \left\{ r_{ik} \log \prod_{j'} p(x_{ij'} | \mu'_{kj'}) \right\} + \log \prod_{j'} p(\mu'_{kj'})}{\partial\mu'_{kj}} = (21)$$

$$= \frac{\partial\sum_{i,j'} \left\{ r_{ik} \log p(x_{ij'} | \mu'_{kj'}) \right\} + \sum_{j'} \log p(\mu'_{kj'})}{\partial\mu'_{kj}} = (22)$$

$$= \left\{ \frac{\partial \sum_{j \neq j'} (\cdot)}{\partial\mu'_{kj}} = 0 \right\} = \frac{\partial\sum_i \left\{ r_{ik} \log \left( \mu'_{kj} x_{ij} (1 - \mu'_{kj})^{(1-x_{ij})} \right) \right\} + \log \frac{\mu'_{kj}{}^{\alpha-1} (1-\mu'_{kj})^{\beta-1}}{B(\alpha, \beta)}}{\partial\mu'_{kj}} = (23)$$

$$= \frac{\partial\sum_i r_{ik} \left( x_{ij} \log \mu'_{kj} + (1 - x_{ij}) \log(1 - \mu'_{kj}) \right)}{\partial\mu'_{kj}} + (24)$$

$$+ \frac{\partial(\alpha - 1) \log \mu'_{kj} + (\beta - 1) \log(1 - \mu'_{kj}) - \log B}{\partial\mu'_{kj}} = (25)$$

$$= \frac{(\sum_i r_{ik} x_{ij}) + \alpha - 1}{\mu'_{kj}} - \frac{(\sum_i r_{ik} (1 - x_{ij})) + \beta - 1}{1 - \mu'_{kj}} = (26)$$

$$= \frac{(1 - \mu'_{kj}) (\sum_i r_{ik} x_{ij}) - \mu'_{kj} (\sum_i r_{ik} (1 - x_{ij}))}{\mu'_{kj} (1 - \mu'_{kj})} + (27)$$

$$+ \frac{(1 - \mu'_{kj}) (\alpha - 1) - \mu'_{kj} (\beta - 1)}{\mu'_{kj} (1 - \mu'_{kj})} = (28)$$

$$= \frac{(\sum_i r_{ik} x_{ij}) - \mu'_{kj} (\sum_i r_{ik}) + (\alpha - 1) - \mu'_{kj} (\alpha + \beta - 2)}{\mu'_{kj} (1 - \mu'_{kj})} (29)$$

and now find the zero of this expression

$$\frac{(\sum_i r_{ik} x_{ij}) - \mu'_{kj} (\sum_i r_{ik}) + (\alpha - 1) - \mu'_{kj} (\alpha + \beta - 2)}{\mu'_{kj} (1 - \mu'_{kj})} = 0 \quad (30)$$

$$(\sum_i r_{ik} x_{ij}) - \mu'_{kj} (\sum_i r_{ik}) + (\alpha - 1) - \mu'_{kj} (\alpha + \beta - 2) = 0 \quad (31)$$

$$\mu'_{kj} [(\sum_i r_{ik}) + \alpha + \beta - 2] = (\sum_i r_{ik} x_{ij}) + \alpha - 1 \quad (32)$$

$$\mu'_{kj} = \frac{(\sum_i r_{ik} x_{ij}) + \alpha - 1}{(\sum_i r_{ik}) + \alpha + \beta - 2} \quad (33)$$

which is the same as the equation (??).  $\square$

By just looking at the expression for  $\beta(\cdot)$  we see that

$$\beta(1,1) = U(0,1) \quad (34)$$

and knowing that ML is a MAP with a uniform priori we can just set  $\alpha, \beta = 1$  in the derived expression to get the ML.

$$\mu'_{kj} = \frac{(\sum_i r_{ik} x_{ij}) + 1 - 1}{(\sum_i r_{ik}) + 1 + 1 - 2} = \frac{\sum_i r_{ik} x_{ij}}{\sum_i r_{ik}} \quad (35)$$

which is the same as the equation (??). □