Bayesian Networks: Maximum Likelihood Estimation and Tree Structure Learning

Huizhen Yu

janey.yu@cs.helsinki.fi
Dept. Computer Science, Univ. of Helsinki

Probabilistic Models, Spring, 2010

Outline

Notices:

- I corrected a number of errors/typos in the slides of Lec. 11. This affected in particular slides 15, 16, 32, 34, 36. There may be other corrections after today's lecture. Please check the online version of the slides; I will put an update sign beside the link.
- Please do not hesitate to contact me if you have any questions before the exam.

Outline

Our Model and Data

Let $X = \{X_v, v \in V\}$ be a collection of discrete random variables.

- G: a DAG on V.
- Our model for X: the set of all distributions $P(X)$ that factorize recursively according to G.
- The true, unknown distribution of X: Q^*, not necessarily in our model.

Maximum likelihood (ML) estimation:

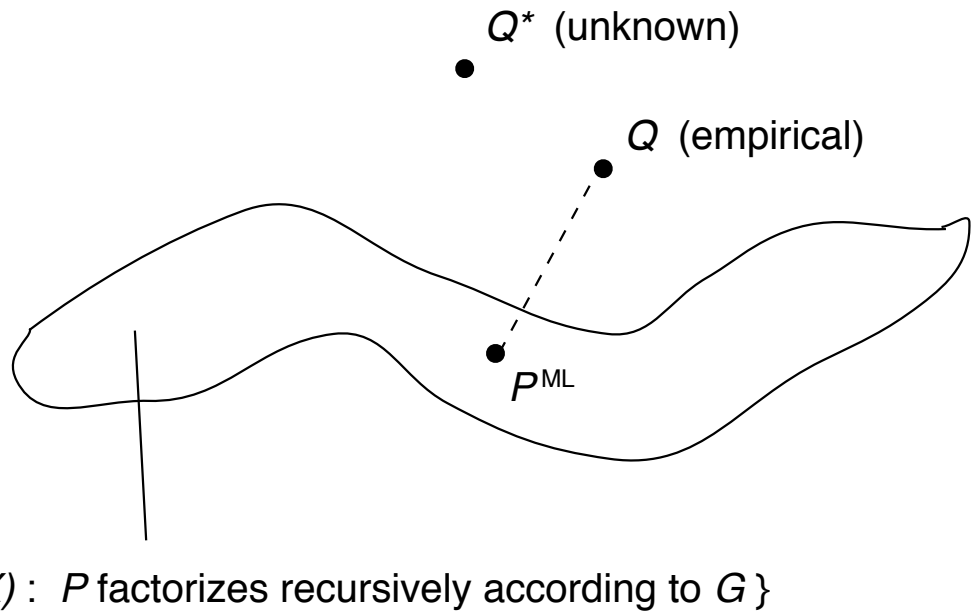
- Data: $\{x^1, x^2, \ldots, x^n\}$, n observations independently generated according to Q^*, (i.e., a random sample of size n).
- The empirical distribution $Q(X)$: $Q(X = \bar{x})$ is the observed frequency of the configuration $\bar{x}$ in the data.
- P^{ML}: the distribution in our model that maximizes the likelihood function based on the data,

$$L(P) = \prod_{i=1}^{n} P(X = x^i) = \prod_{i=1}^{n} \prod_{v \in V} p\left(x_v^i \,|\, x_{\mathrm{pa}(v)}^i\right).$$

(For simplicity, we do not use the θ notation for parameters here.)

Relation between the ML Estimate, the empirical and the true distributions

The relation between P^{ML}, Q and Q^*:



- Among all P in our model, P^{ML} is the closest distribution to Q in terms of the KL-divergence $\mathrm{KL}(q, p)$. (q is the PMF of Q.)

(See discussions in Lec. 3 and Problem 3 of Exercise 2.)

Expression of the ML Estimate

The ML estimate P^{ML} is the distribution given by

$$p^{\mathrm{ML}}(x) = \prod_{v \in V} p^{\mathrm{ML}}(x_v \,|\, x_{\mathrm{pa}(v)}),$$

where the component conditional distributions are defined by

$$p^{\mathrm{ML}}(x_v \,|\, x_{\mathrm{pa}(v)}) = Q(X_v = x_v \,|\, X_{\mathrm{pa}(v)} = x_{\mathrm{pa}(v)}) = \frac{n(x_v, x_{\mathrm{pa}(v)})}{n(x_{\mathrm{pa}(v)})}, \qquad (1)$$

and in the last expression,

- $n(x_{\mathrm{pa}(v)})$: the counts for the configuration $x_{\mathrm{pa}(v)}$ in the data;
- $n(x_v, x_{\mathrm{pa}(v)})$: the counts for the configuration $(x_v, x_{\mathrm{pa}(v)})$ in the data.

The maximized log likelihood can be expressed as

$$\ell(P^{\mathrm{ML}}) = n \,\mathrm{E}_Q\big[\ln p^{\mathrm{ML}}(X)\big] = n \,\mathrm{E}_Q\Big[\sum_{v \in V} \ln q(X_v \,|\, X_{\mathrm{pa}(v)})\Big], \qquad (2)$$

where E_Q denotes expectation with respect to the distribution Q.

(Eqs. (1)-(2) can be derived using the information inequality; see slides 18-19 for details.)

Outline

Learning a Rooted Tree

Problem:

- Given the data as described earlier, find a rooted tree G which maximizes the *profile log likelihood* $\ell_{\mathrm{p}}(G)$:

$$\ell_{\mathrm{p}}(G) \stackrel{def}{=} \ell(G, P_G^{\mathrm{ML}}) = \max_{P \in \mathcal{P}(G)} \ell(G, P).$$

  Here $\mathcal{P}(G)$ is the set of all distributions that factorize recursively according to G.

Such a tree is also called a *Chow-Liu tree*, and can be found by the Chow-Liu tree algorithm (Chow and Liu, 1968).

The algorithm can be generalized to solve similar types of problems (we will show one).

Recall Mutual Information and Conditional Mutual Information

Let X, Y, Z be discrete random variables with joint distribution P.

- The *mutual information* between X and Y is defined as

$$\mathcal{I}(X;Y) = \mathrm{E}\left[\ln\left(\frac{p(X,Y)}{p(X)p(Y)}\right)\right],$$

and equivalently,

$$\mathcal{I}(X;Y) = \sum_{x,y} p(x,y)\ln\left(\frac{p(x,y)}{p(x)p(y)}\right).$$

- The *conditional mutual information* between X and Y given Z is defined as

$$\mathcal{I}(X;Y\,|\,Z) = \mathrm{E}\left[\ln\left(\frac{p(X,Y\,|\,Z)}{p(X\,|\,Z)p(Y\,|\,Z)}\right)\right],$$

and equivalently,

$$\mathcal{I}(X;Y\,|\,Z) = \sum_{z} p(z)\sum_{x,y} p(x,y\,|\,z)\ln\left(\frac{p(x,y\,|\,z)}{p(x\,|\,z)p(y\,|\,z)}\right).$$

- By the information inequality,

$$\mathcal{I}(X;Y) \geq 0, \quad \text{and} \quad \mathcal{I}(X;Y) = 0 \ \text{ iff. } \ X \perp Y;$$

$$\mathcal{I}(X;Y\,|\,Z) \geq 0, \quad \text{and} \quad \mathcal{I}(X;Y\,|\,Z) = 0 \ \text{ iff. } \ X \perp Y \mid Z.$$

Deriving the Chow-Liu Tree Algorithm

We start with the profile log likelihood: by Eq. (2),

$$\ell_{\mathrm{p}}(G) = n\,\mathrm{E}_Q\Big[\sum_{v\in V} \ln q(X_v \,|\, X_{\mathrm{pa}_G(v)})\Big].$$

Here $\mathrm{pa}_G(v)$ is the parent of v in the rooted tree G.

Rewrite $\ell_{\mathrm{p}}(G)$ in terms of the mutual information $\mathcal{I}_Q(X_v; X_{\mathrm{pa}_G(v)})$, $v \in V$ (w.r.t. the distribution Q):

$$\begin{aligned}
\mathrm{E}_Q\Big[\ln q(X_v \,|\, X_{\mathrm{pa}_G(v)})\Big] &= \mathrm{E}_Q\left[\ln\left(\frac{q(X_v \,|\, X_{\mathrm{pa}_G(v)}) \cdot q(X_{\mathrm{pa}_G(v)}) \cdot q(X_v)}{q(X_v) \cdot q(X_{\mathrm{pa}_G(v)})}\right)\right] \\
&= \mathrm{E}_Q\left[\ln\left(\frac{q(X_v, X_{\mathrm{pa}_G(v)})}{q(X_v) \cdot q(X_{\mathrm{pa}_G(v)})}\right)\right] + \mathrm{E}_Q\big[\ln q(X_v)\big] \\
&= \mathcal{I}_Q(X_v; X_{\mathrm{pa}_G(v)}) + E_Q\big[\ln q(X_v)\big];
\end{aligned}$$

hence

$$\frac{1}{n}\ell_{\mathrm{p}}(G) = \sum_{v\in V} \mathcal{I}_Q(X_v; X_{\mathrm{pa}_G(v)}) + \sum_{v\in V} E_Q\big[\ln q(X_v)\big]. \tag{3}$$

Deriving the Chow-Liu Tree Algorithm

In the last equation,

$$\frac{1}{n}\ell_{\mathrm{p}}(G) = \sum_{v\in V} \mathcal{I}_Q(X_v; X_{\mathrm{pa}_G(v)}) + \sum_{v\in V} E_Q\big[\ln q(X_v)\big],$$

- the second term does not dependent on G and therefore can be left out when maximizing $\ell_{\mathrm{p}}(G)$ over G;
- the mutual information is symmetric: $\mathcal{I}_Q(X_v; X_{\mathrm{pa}_G(v)}) = \mathcal{I}_Q(X_{\mathrm{pa}_G(v)}; X_v)$.

Therefore,

$$\max_{G\in\{\text{rooted trees}\}} \ell_{\mathrm{p}}(G) \quad \Leftrightarrow \quad \max_{G^\sim\in\{\text{undirected trees}\}} \sum_{v \overset{G^\sim}{\sim} u} \mathcal{I}_Q(X_v; X_u), \qquad (4)$$

where the summation $\sum_{v \overset{G^\sim}{\sim} u}$ is over all edges of $G^\sim$.

Chow-Liu Tree Algorithm

(1) Compute all pairwise mutual information

$$\mathcal{I}_Q(X_v; X_u) = E_Q\left[\ln\left(\frac{q(X_v, X_u)}{q(X_v)q(X_u)}\right)\right], \quad v, u \in V.$$

(2) Find a *maximum spanning tree* of the undirected, fully connected graph on V with

- edge weight $\mathcal{I}_Q(X_v; X_u)$ between node v and u.

This can be done by Kruskal's algorithm:

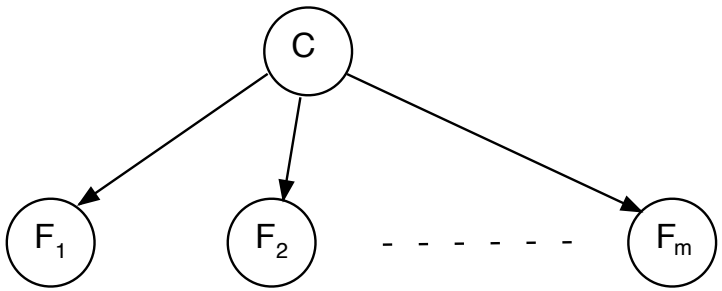
- repeatedly select an edge with maximum weight that does not create a cycle.

(3) Make any node of the spanning tree as the root and direct edges away from it.

The result is a rooted tree G that maximizes $\ell_{\mathrm{p}}(G)$.

Generalization to Learning Tree Augmented Naive Bayes

A naive Bayes classifier with class variable C and feature variables F_i:



Naive Bayes neglects the dependence between feature variables. This can be troublesome for rare classes that have characteristic combinations of features.

Tree augmented naive Bayes classifiers (TAN):

- Each feature variable has at most one other feature variable as its parent besides the class variable.
- In other words, the subgraph induced by the feature variables is a rooted tree or forest.

Consider the problem of learning a TAN G with maximum likelihood.

Learning TAN

Notation:

- $X_v, v \in V$: feature variables.
- $\widehat{G}$: the subgraph of G induced by the feature variables $X_v, v \in V$.
- $\mathrm{pa}_{\widehat{G}}(v)$: the parent of v in $\widehat{G}$, i.e., the parent of v in G besides C.

Note that a TAN G is uniquely determined by its associated $\widehat{G}$.

Apply the Chow-Liu tree algorithm to learning TAN:

- Replace all pairwise mutual information by the conditional mutual information between all pairs of feature variables given the class variable:

$$\mathcal{I}_Q(X_v; X_u \,|\, C) = E_Q\left[\ln\left(\frac{q(X_v, X_u \,|\, C)}{q(X_v \,|\, C) q(X_u \,|\, C)}\right)\right], \quad v, u \in V.$$

- The output of the algorithm is the subgraph $\widehat{G}$ whose associated TAN G maximizes the profile log likelihood among all TANs.

Deriving the Chow-Liu Tree Algorithm for TAN

Similarly to learning a rooted tree, we start with the profile log likelihood: by Eq. (2),

$$\ell_{\mathrm{p}}(\widehat{G}) = \ell_{\mathrm{p}}(G) = n\,\mathrm{E}_Q\Big[\sum_{v\in V} \ln q(X_v \,|\, X_{\mathrm{pa}_{\widehat{G}}(v)}, C)\Big] + n\,\mathrm{E}_Q\big[\ln q(C)\big]. \qquad (5)$$

We rewrite $\ell_{\mathrm{p}}(\widehat{G})$ in terms of the conditional mutual information $\mathcal{I}_Q\big(X_v; X_{\mathrm{pa}_{\widehat{G}}(v)} \,|\, C\big)$ between X_v and $X_{\mathrm{pa}_{\widehat{G}}(v)}$ given C for $v \in V$:

$$\begin{aligned}
\mathrm{E}_Q\big[\ln q(X_v \,|\, X_{\mathrm{pa}_{\widehat{G}}(v)}, C)\big] &= \mathrm{E}_Q\left[\ln\left(\frac{q\big(X_v \,|\, X_{\mathrm{pa}_{\widehat{G}}(v)}, C\big)\cdot q\big(X_{\mathrm{pa}_{\widehat{G}}(v)} \,|\, C\big)\cdot q\big(X_v \,|\, C\big)}{q\big(X_v \,|\, C\big)\cdot q\big(X_{\mathrm{pa}_{\widehat{G}}(v)} \,|\, C\big)}\right)\right] \\
&= \mathrm{E}_Q\left[\ln\left(\frac{q\big(X_v, X_{\mathrm{pa}_{\widehat{G}}(v)} \,|\, C\big)}{q\big(X_v \,|\, C\big)\cdot q\big(X_{\mathrm{pa}_{\widehat{G}}(v)} \,|\, C\big)}\right)\right] + \mathrm{E}_Q\big[\ln q(X_v \,|\, C)\big] \\
&= \mathcal{I}_Q\big(X_v; X_{\mathrm{pa}_{\widehat{G}}(v)} \,|\, C\big) + \mathrm{E}_Q\big[\ln q(X_v \,|\, C)\big];
\end{aligned}$$

hence

$$\frac{1}{n}\ell_{\mathrm{p}}(\widehat{G}) = \sum_{v\in V} \mathcal{I}_Q\big(X_v; X_{\mathrm{pa}_{\widehat{G}}(v)} \,|\, C\big) + \sum_{v\in V} \mathrm{E}_Q\big[\ln q(X_v \,|\, C)\big] + \mathrm{E}_Q\big[\ln q(C)\big]. \qquad (6)$$

Deriving the Chow-Liu Tree Algorithm for TAN

In the last equation,

$$\frac{1}{n}\ell_{\mathrm{p}}(\widehat{G}) = \sum_{v\in V}\mathcal{I}_Q\big(X_v; X_{\mathrm{pa}_{\widehat{G}}(v)} \,|\, C\big) + \sum_{v\in V}\mathrm{E}_Q\big[\ln q(X_v \,|\, C)\big] + \mathrm{E}_Q\big[\ln q(C)\big],$$

- the second and third terms do not dependent on $\widehat{G}$ and therefore can be left out when maximizing $\ell_{\mathrm{p}}(\widehat{G})$ over $\widehat{G}$;
- the conditional mutual information is symmetric:

$$\mathcal{I}_Q(X_v; X_{\mathrm{pa}_{\widehat{G}}(v)} \,|\, C) = \mathcal{I}_Q(X_{\mathrm{pa}_{\widehat{G}}(v)}; X_v \,|\, C);$$

- if $\widehat{G}$ is a forest, adding edges to make it a tree will not decrease $\ell_{\mathrm{p}}(\widehat{G})$.

Therefore,

$$\max_{\widehat{G}\in\{\text{rooted trees}\}} \ell_{\mathrm{p}}(\widehat{G}) \quad\Leftrightarrow\quad \max_{\widehat{G}^{\sim}\in\{\text{undirected trees}\}} \sum_{v \overset{\widehat{G}^{\sim}}{\sim} u} \mathcal{I}_Q(X_v; X_u \,|\, C), \qquad (7)$$

where the summation $\sum_{v \overset{\widehat{G}^{\sim}}{\sim} u}$ is over all edges of $\widehat{G}^{\sim}$.

This verifies the claim in slide 14, that we can apply the Chow-Liu tree algorithm with $\mathcal{I}_Q(X_v; X_u \,|\, C)$ replacing $\mathcal{I}_Q(X_v; X_u)$ for all $v, u \in V$, to obtain the desirable $\widehat{G}$.

Discussion

- Rooted trees and TANs are perfect DAGs: $G^m = G^\sim$.
- So the models are equivalent to those associated with the undirected graphs $G^\sim$, and it is not surprising that the structure learning algorithms we derived can disregard edge directions.

Further Readings

For TAN:

1. Finn V. Jensen and Thomas D. Nielsen. *Bayesian Networks and Decision Graphs*. Springer, 2007. Chap. 8.

For learning a singly connected network (under certain assumptions) with the Chow-Liu tree algorithm, see Pearl's 1988 book.

An old review article discussing the ideas and steps involved in developing a probabilistic expert system, using the example CHILD network:

2. David J. Spiegelhalter et al. Bayesian analysis in expert systems, *Statistical Science*, Vol. 8, No. 3, pp. 219-283, 1993.

(It includes Bayesian inference, which we did not talk about.) You may also find the related materials in the book by Cowell et al. 2007.

A recent book by Koller and Friedman, *Probabilistic Graphical Models*, 2009 has many materials on both approximate and exact inference algorithms.

Derivation of Eqs. (1)-(2)

The likelihood and log likelihood functions are

$$L(P) = \prod_{i=1}^{n} \prod_{v \in V} p\big(x_v^i \,|\, x_{\mathrm{pa}(v)}^i\big), \qquad \ell(P) = \sum_{i=1}^{n} \sum_{v \in V} \ln p\big(x_v^i \,|\, x_{\mathrm{pa}(v)}^i\big).$$

The variables in the maximization of $\ell(P)$ are the conditional distributions $p(x_v \,|\, x_{\mathrm{pa}(v)})$ of X_v for each configuration $x_{\mathrm{pa}(v)}$ of v's parents, for all $v \in V$. We next express $\ell(P)$ in terms of these variables (colored in blue below)

By exchanging the order of summations in the expression of $\ell(P)$,

$$\ell(P) = \sum_{v \in V} \sum_{i=1}^{n} \ln p\big(x_v^i \,|\, x_{\mathrm{pa}(v)}^i\big) = \sum_{v \in V} \sum_{x_{\mathrm{pa}(v)}} \sum_{x_v} n(x_v, x_{\mathrm{pa}(v)}) \ln p\big(x_v \,|\, x_{\mathrm{pa}(v)}\big).$$

where $n(x_v, x_{\mathrm{pa}(v)})$ is the counts for the configuration $(x_v, x_{\mathrm{pa}(v)})$ in the data.

Under our model, there are no constraints between the component conditional distributions we can choose. So the maximization problem $\max_P \ell(P)$ decomposes into separate maximization problems, one for each v and its parent configuration $x_{\mathrm{pa}(v)}$:

$$\max_{p(\cdot \,|\, x_{\mathrm{pa}(v)})} \sum_{x_v} n(x_v, x_{\mathrm{pa}(v)}) \cdot \ln p\big(x_v \,|\, x_{\mathrm{pa}(v)}\big). \tag{8}$$

($x_{\mathrm{pa}(v)}$ is fixed in the above subproblem.)

Derivation of Eqs. (1)-(2)

The subproblem (8) is equivalent to

$$\max_{p(\cdot\,|\,x_{\mathrm{pa}(v)})} \sum_{x_v} \frac{n(x_v, x_{\mathrm{pa}(v)})}{n(x_{\mathrm{pa}(v)})} \cdot \ln p(x_v \,|\, x_{\mathrm{pa}(v)}), \tag{9}$$

where $n(x_{\mathrm{pa}(v)}) = \sum_{x_v} n(x_v, x_{\mathrm{pa}(v)})$, and it is the counts of the parent configuration $x_{\mathrm{pa}(v)}$ in the data.

By the information inequality (see Lec. 3), the maximum of (9) is attained at

$$p(x_v \,|\, x_{\mathrm{pa}(v)}) = \frac{n(x_v, x_{\mathrm{pa}(v)})}{n(x_{\mathrm{pa}(v)})}, \quad \forall x_v,$$

which is the ML estimate $p^{\mathrm{ML}}(\cdot \mid x_{\mathrm{pa}(v)})$ given in Eq. (1).

The maximized log likelihood thus equals

$$\begin{aligned}
\ell(P^{\mathrm{ML}}) &= \sum_{v \in V} \sum_{x_{\mathrm{pa}(v)}} \sum_{x_v} n(x_v, x_{\mathrm{pa}(v)}) \cdot \ln \frac{n(x_v, x_{\mathrm{pa}(v)})}{n(x_{\mathrm{pa}(v)})} \\
&= n \cdot \sum_{v \in V} \sum_{x_{\mathrm{pa}(v)}} \sum_{x_v} \frac{n(x_v, x_{\mathrm{pa}(v)})}{n} \cdot \ln \frac{n(x_v, x_{\mathrm{pa}(v)})}{n(x_{\mathrm{pa}(v)})} \\
&= n \cdot \sum_{v \in V} \sum_{x_{\mathrm{pa}(v)}} \sum_{x_v} q(x_v, x_{\mathrm{pa}(v)}) \cdot \ln q(x_v \,|\, x_{\mathrm{pa}(v)}) = n \, \mathrm{E}_Q \Big[\sum_{v \in V} \ln q(X_v \,|\, X_{\mathrm{pa}(v)}) \Big].
\end{aligned}$$

(q is the PMF of Q.) This verifies Eq. (2).