14 Inference in complex models

What if graph is not a tree?

NP-hard even to approximate marginals and conditionals

General strategies

1. Exact methods — exponential time, but can still try to be smart
2. Approximation methods
3. Heuristic methods
4. Monte Carlo methods — estimate by random sampling

14.1 Exact methods

Elimination ordering

Try to find a good variable order that reduces work in summation

- push variable in
- eliminate variables by summing and pull result out

Variable clustering

Cluster variables to create a tree structured Bayesian network

- exponential in the size of the largest cluster

Cut sets

Choose a cut set of variables that turn factor graph into a tree

- sum over cut set configurations
- exponential in size of cut set

14.2 Approximation methods

"Variational approximation"

- Pick simple model structure (i.e. a tree)
- Set values in new CP tables so that new distribution approximates original distribution as closely as possible
- Perform efficient inference on simpler approximate distribution

A bit complicated to implement sometimes, but can be very effective

14.3 Heuristic methods

"Loopy probability propagation"

Ignore loops and use same message passing algorithm as for trees

- random initial messages
- keep passing messages around graph
- wait for product of incoming messages to converge
- if so, is the answer accurate?

This works way better than it should!

14.4 Monte Carlo methods

Use random sampling to *estimate* answers

14.4.1 Estimating marginals

To estimate $\mathrm{P}(X_i = x_i)$, draw joint configurations

$$\begin{matrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{t1} & x_{t2} & \dots & x_{tn} \end{matrix}$$

Use estimate: $\hat{\mathrm{P}}(X_i = x_i) = \dfrac{\#\ \text{matches}(X_i = x_i)}{t}$

Unbiased: $\mathrm{E}\hat{\mathrm{P}}(X_i = x_i) = \mathrm{P}(X_i = x_i)$

14.4.2 Estimating conditionals

Estimate $P(X_{k+1} = y_{k+1} | X_1 = x_1, \ldots, X_k = x_k)$

Draw joint configurations:

$$\begin{array}{cccc} x_{11} & x_{12} & \ldots & x_{1n} \\ x_{21} & x_{22} & \ldots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{t1} & x_{t2} & \ldots & x_{tn} \end{array}$$

Use estimate:

$$\begin{aligned} & \hat{P}(X_{k+1} = y_{k+1} | X_1 = x_1, \ldots, X_k = x_k) \\ = & \frac{\#\ \text{matches}(X_1 = x_1, \ldots, X_k = x_k, X_{k+1} = y_{k+1})}{\#\ \text{matches}(X_1 = x_1, \ldots, X_k = x_k)} \end{aligned}$$

This technique is called "logic sampling"
It is a bad estimator if $(X_1 = x_1, \ldots, X_k = x_k, X_{k+1} = y_{k+1})$ is unlikely:

- small effective sample size

14.4.3 Aside: General "importance sampling"

Consider estimating the expectated value of some function $f(x)$, where x is drawn randomly according to the distribution $P(x)$. That is, assume the expectation of $f(x)$ is defined

$$E_{P(x)}(f(x)) = \sum_x f(x)P(x)$$

Many problems (including estimating conditional probabilities) can be expressed as estimating the expected value of a function f.

The simplest way to estimate $E_{P(x)} f(x)$ is the Monte Carlo method

- Draw $x_1, x_2, \ldots, x_t$ from P
- Use estimate:

$$\hat{f} = \frac{1}{t} \sum_{i=1}^{t} f(x_i)$$

Problem: what if you cannot sample from P efficiently?

First assume that we can at least efficiently *evaluate* $P(x)$ at given points x.

Idea: Pick a proposed distribution Q that you *can* sample from

- Draw $x_1, x_2, \ldots, x_t$ from Q.
- Weight points by $w(x_i) = \frac{\mathrm{P}(x_i)}{\mathrm{Q}(x_i)}$
- Use estimate: $\hat{f} \;=\; \frac{1}{t}\sum_{i=1}^{t} f(x_i)w(x_i)$

This gives an unbiased estimate

$$\begin{aligned}
\frac{1}{t}\sum_{i=1}^{t} f(x_i)w(x_i) &\overset{t\to\infty}{\longrightarrow} \mathrm{E}_{\mathrm{Q}(x)} f(x)w(x) \\
&= \sum_x f(x)w(x)\mathrm{Q}(x) \\
&= \sum_x f(x)\frac{\mathrm{P}(x)}{\mathrm{Q}(x)}Q(x) \\
&= \sum_x f(x)\mathrm{P}(x) \\
&= \mathrm{E}_{\mathrm{P}(x)} f(x).
\end{aligned}$$

More realistically: You cannot even *evaluate* $\mathrm{P}(x)$ efficiently

However, in these cases, you often still have a function $\mathrm{R}(x) = \beta\ \mathrm{P}(x)$ that you can evaluate efficiently (up to some unknown value β). In which case you can use following *indirect* importance sampling procedure.

- Draw $x_1, x_2, \ldots, x_t$ from Q.
- Weight points by $u(x) = \frac{\mathrm{R}(x)}{\mathrm{Q}(x)}$
- Use the estimate
$$\hat{f} = \frac{\sum_{i=1}^{t} f(x_i)u(x_i)}{\sum_{i=1}^{t} u(x_i)}$$

This procedure is biased, but it is asymptotically unbiased:

$$\frac{1}{t}\sum_{i=1}^{t} f(x_i)u(x_i) \overset{t\to\infty}{\longrightarrow} \sum_x f(x)u(x)\mathrm{Q}(x) \;=\; \sum_x f(x)\mathrm{R}(x) \;=\; \beta\sum_x f(x)\mathrm{P}(x)$$

$$\frac{1}{t}\sum_{i=1}^{t} u(x_i) \;\xrightarrow{t\to\infty}\; \sum_x u(x)\mathrm{Q}(x) \;=\; \sum_x \mathrm{R}(x) \;=\; \beta\sum_x \mathrm{P}(x) \;=\; \beta$$

Therefore

$$\hat{f} \;\xrightarrow{t\to\infty}\; \frac{\beta\sum_x f(x)\mathrm{P}(x)}{\beta} \;=\; \mathrm{E}_{\mathrm{P}(x)} f(x).$$

14.4.4 Estimating conditionals using importance sampling

Want to estimate $\mathrm{P}(\mathbf{X}_\beta = \mathbf{y}_\beta | \mathbf{X}_\alpha = \mathbf{x}_\alpha)$ where α and β are sets of indices from $\{1,\ldots,n\}$ such that $\alpha \cap \beta = \emptyset$ and $\alpha \cup \beta = \{1,\ldots,n\}$. unfortunately it is both hard to sample from and evaluate $\mathrm{P}(\mathbf{X}_\beta = \mathbf{y}_\beta | \mathbf{X}_\alpha = \mathbf{x}_\alpha)$ directly. we proceed as follows

- clamp the variables $\mathbf{X}_\alpha = \mathbf{x}_\alpha$
- sample the remaining "free" variables in the usual way (keeping the clamped variables at their assigned values)
- repeat t times to create a sample of configurations $\mathbf{x}_1, ..., \mathbf{x}_t$
- Define the function

$$f(\mathbf{x}_\beta) \;=\; \begin{cases} 1 & \text{if } \mathbf{x}_\beta = \mathbf{y}_\beta \\ 0 & \text{otherwise} \end{cases}$$

- Calculate weights

$$u(\mathbf{x}_{\beta,i}) \;=\; \frac{\mathrm{R}(\mathbf{x}_{\beta,i})}{\mathrm{Q}(\mathbf{x}_{\beta,i})}$$

  where $\mathrm{R}(\mathbf{x}_{\beta,i}) = \mathrm{P}(\mathbf{X}_\alpha = \mathbf{x}_{\alpha,i}, \mathbf{X}_\beta = \mathbf{x}_{\beta,i})$

  and $\mathrm{Q}(\mathbf{x}_{\beta,i}) = \prod_{j\in\beta} \mathrm{P}(X_j = x_{j,i} | \mathbf{X}_{\pi(j)} = \mathbf{x}_{\pi(j),i})$

- Use the estimate

$$\hat{\mathrm{P}}(\mathbf{X}_\beta = \mathbf{y}_\beta | \mathbf{X}_\alpha = \mathbf{x}_\alpha) \;=\; \frac{\sum_{i=1}^{t} f(\mathbf{x}_{\beta,i}) u(\mathbf{x}_{\beta,i})}{\sum_{i=1}^{t} u(\mathbf{x}_{\beta,i})}$$

This method has larger effective sample size than logic sampling.

Works even if $P(\mathbf{X}_\alpha = \mathbf{x}_\alpha)$ is small.

Readings

Russell and Norvig 2nd Ed: Section 14.5
Dean, Allen, Aloimonos: Section 8.3