Machine Learning

Lecture 9 – Graphical models and message passing

UPPSALA UNIVERSITET

Thomas Schön
Division of Systems and Control
Department of Information Technology
Uppsala University.

Email: thomas.schon@it.uu.se,
www: `user.it.uu.se/~thosc112`

Contents – lecture 9

1. Summary of lecture 8
2. Undirected graphs (Markov random fields)
   - General properties
   - Conditional independence
   - Relation with directed graphs
3. Factor graphs
4. Inference in graphical models
5. Belief propagation (sum-product algorithm)

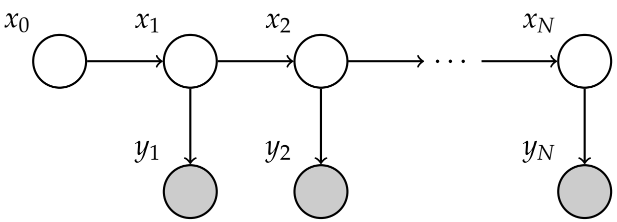
(Chapter 8.3-8.4)

Summary of lecture 8 (I/II)

A **graphical model** is a probabilistic model where a graph is used to represent the CI structure between random variables.

We introduced basic concepts for **graphical models** $\mathcal{G} = (\mathcal{V}, \mathcal{E})$,

1. a set of **vertices** $\mathcal{V}$ (a.k.a. nodes) representing the random variables and
2. a set of **edges** $\mathcal{E}$ (a.k.a. links or arcs) containing elements $(i, j) \in \mathcal{E}$ connecting a pair of nodes $(i, j) \in \mathcal{V}$ and thereby encoding the probabilistic relations between nodes.



Summary of lecture 8 (II/II)

The set of parents to node j (pa_j) is defined as

$$\text{pa}_j \triangleq \{i \in \mathcal{V} \mid (i, j) \in \mathcal{E}\}.$$

The directed graph describes how the joint distribution $p(x)$ **factors** into a product of factors $p(x_i \mid x_{\text{pa}_i})$ only depending on a subset of the variables,

$$p(x_{\mathcal{V}}) = \prod_{i \in \mathcal{V}} p(x_i \mid x_{\text{pa}_i}).$$

Hence, for the state space model on the previous slide, we have

$$p(X, Y) = p(x_0) \prod_{t=1}^{N} p(x_t \mid x_{t-1}) \prod_{t=1}^{N} p(y_t \mid x_t).$$

D-separation was used as a means to check conditional independence among random variables.

2014 Gibbs lecture

"The visible world is awash with ambiguity, and probability, the calculus of uncertainty, is an important element of the computer systems that resolve that ambiguity."

Title: **Machines that see, powered by probability**

Speaker: Andrew Blake (laboratory director of Microsoft Research Cambridge)

The talk is available here: `research.microsoft.com/en-us/about/andrew-blake-gibbs-lecture-2014.pdf`

Conclusion:

- Vision must address ambiguity and noise.
- Seeing machines need probabilistic elements
    - Variational methods are not enough
- Generative models alone are insufficient

Example – Gaussian mixture (I/II)

Suppose we have $x_{1:N}$ i.i.d. and distributed as

$$x_i \sim p(x_i | \pi_{1:K}, \mu_{1:K}, \Lambda_{1:K}) = \sum_{k=1}^{K} \pi_k \mathcal{N}\left(x_i; \mu_k, \Lambda_k^{-1}\right)$$

for $i = 1, \dots, N$.

In a Bayesian model, all the unknowns $\{\pi_{1:K}, \mu_{1:K}, \Lambda_{1:K}\}$ are modelled as random variables.

$$\pi_{1:K} \sim \text{Dir}(\pi_{1:K} | \alpha_0) \overset{\triangle}{\propto} \prod_{k=1}^{K} \pi_k^{\alpha_0 - 1}$$

$$\mu_{1:K}, \Lambda_{1:K} \sim p(\mu_{1:K}, \Lambda_{1:K}) \triangleq \prod_{k=1}^{K} \mathcal{N}(\mu_k; m_0, (\beta_0 \Lambda_k)^{-1}) \mathcal{W}(\Lambda_k | W_0, \nu_0)$$

Example – Gaussian mixture (II/II)

Define the latent variables $z_n \triangleq [z_{n1}, \cdots, z_{nK}]^{\mathsf{T}}$ for $n = 1, \dots, N$ as we did in the construction used for EM and VB.

Then the joint density can be written as

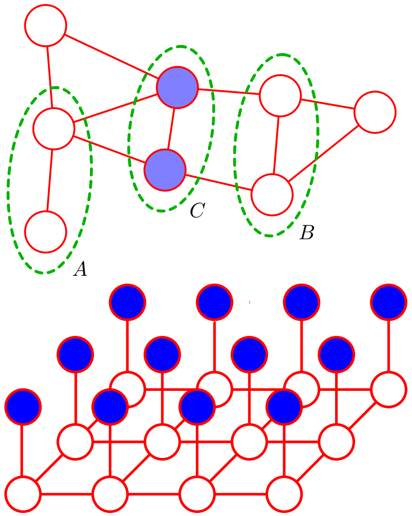
$$p(x_{1:N}, z_{1:N}) = \prod_{n=1}^{N} \prod_{k=1}^{K} \pi_k^{z_{nk}} \mathcal{N}\left(x_n; \mu_k, \Lambda_k^{-1}\right)^{z_{nk}}$$

Undirected graphical model (Markov random fields)

- Nodes and edges carry similar meanings.
- Conditional independence is determined by graphical separation.

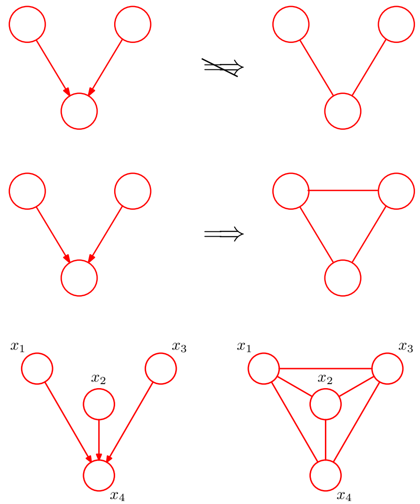
$$A \perp B | C$$

- A more natural representation for some models, e.g., images.
- One must take special care while converting directed graphs to undirected ones.



Conversion from directed to undirected

- When conversion is done directly some correlations that would be present in the original model can be lost.
- One must “marry” the parents to get those correlations back, this is called **moralization**.
- Moralization has to be performed for all the pairs of parents.



Close connection to statistical physics

- The Hammersley-Clifford theorem has a physics interpretation when the functions $\psi_C(x_C)$ are non-zero everywhere.
- In this case, we can write

$$\psi_C(x_C) = \exp(-E(x_C))$$

  where $E(\cdot)$ is called an **energy function**.
- The overall graph can then be considered as a lattice with a potential energy function described by $E(x_C)$.
- Finding the maximum of the density can then be considered as finding the point where the total potential energy is minimized.

$$p(x_{1:N}) = \frac{1}{Z}\prod_C \exp(-E(x_C)) = \frac{1}{Z}\exp\left(-\sum_C E(x_C)\right)$$

- A local maximum then corresponds to an equilibrium.

Application – image de-noising (I/II)

Suppose we have a noisy image and want to remove the noise.

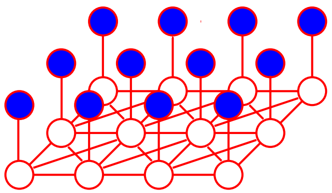
- Model the true pixel values as $x_{i,j}$.
- Model the measured image pixel values as

$$y_{i,j} = x_{i,j} + v_{i,j}, \quad v_{i,j} \sim \mathcal{N}(0, \beta^2).$$

- Choose the energy functions as

$$E_y(x_{i,j}, y_{i,j}) = \frac{1}{\beta^2}(y_{i,j} - x_{i,j})^2$$

$$E_x(x_{i_1,j_1}, x_{i_2,j_2}) = \min\left(\frac{1}{\alpha^2}(x_{i_1,j_1} - x_{i_2,j_2})^2, \gamma\right)$$

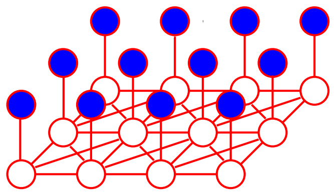


Application – image de-noising (II/II)

- The density is then

$$-\log p(x_{1:N_x,1:N_y}, y_{1:N_x,1:N_y}) = \sum_{i,j} E_y(x_{i,j}, y_{i,j}) + E_x(x_{i,j}, x_{i+1,j+1})$$
$$+ E_x(x_{i,j}, x_{i-1,j-1}) + E_x(x_{i,j}, x_{i-1,j+1}) + E_x(x_{i,j}, x_{i+1,j-1}) + C.$$

- If the image is 8 bit grayscale, maximization in general requires the calculation of $256^{(N_x \times N_y)}$ different combinations.
- We instead maximize w.r.t. only one pixel keeping the others fixed at their last values.
- This is called **iterative conditional modes (ICM)**.



Run example!

Conditional random field

A **conditional random field (CRF)** is a particular MRF where all the clique potentials are conditioned on input features:

$$p(x \mid y) = \frac{1}{Z(y)} \prod_{c \in C} \psi_c(x_c \mid y).$$

This opens up for the possibility of making the potentials (factors) data dependent.

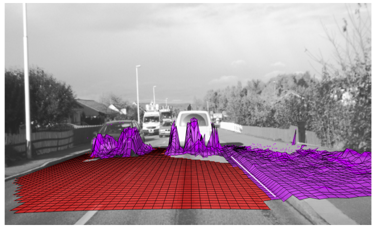
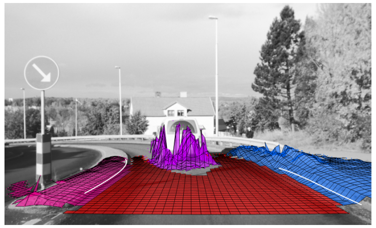
CRFs do not model things that we observe, means that we are “saving resources”.

Sutton, C. and McCallum, A. **An introduction to conditional random fields**. *Foundations and Trends in Machine Learning*, 4(4): 267–373, 2011.

Road surface estimation

Aim: Estimate the road surface using images from a stereo camera.

Solved using a CRF model and message passing.

Show movie

Lorentzon, M. and Andersson, T. **Road surface modeling using stereo vision**, Master’s thesis, LiTH-ISY-EX–12/4582–SE, Linköping university, Sweden, 2012.
http://liu.diva-portal.org/smash/record.jsf?searchId=2&pid=diva2:532767

Inference in graphical models

Inference in graphical models amounts to computing the posterior distribution of one or more of the nodes that are not observed.

The **structure** in the graphical model is exploited in finding inference algorithms.

Most inference algorithms can be expressed in terms of **message passing** algorithms, where local messages are propagated around the graph.

Inference on a chain

Hence, inference on a graph consisting of a chain of nodes can be performed efficiently at a computational cost that is linear in the number of nodes.

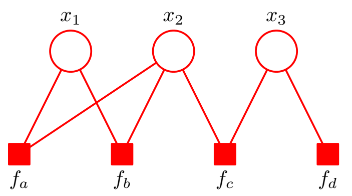
The algorithm can be interpreted as passing messages around in the graph.

The generalization of this message passing idea to trees is referred to as the **sum-product algorithm**.

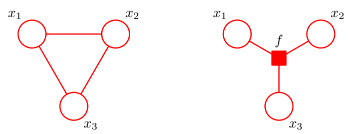
Definition (**Tree**): in an undirected graph a tree is defined as a graph where there is one, and only one, path between any pair of nodes.

Factor graphs

- Both directed and undirected graphs give a factorial representation for the joint density.
- Factor graphs make this factorization more explicit by adding nodes for each factor.
- Both directed and undirected graphs can be converted into factor graphs.



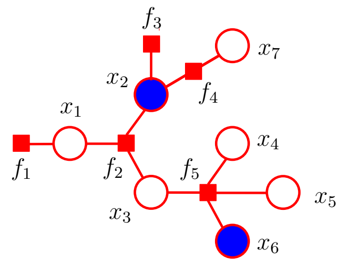
$$p(x_{1:3}) = f_a(x_1, x_2) f_b(x_1, x_2) f_c(x_2, x_3) f_d(x_3)$$



Inference in factor graphs (I/V)

- We have the joint density for the graph on the right given as

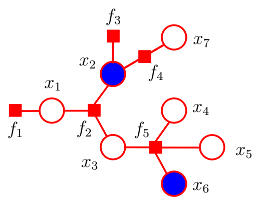
$$p(x_{1:7}) \propto f_1(x_1) f_2(x_{1:3}) f_3(x_2) f_4(x_2, x_7) f_5(x_{3:6})$$



- When we have measurements of some variables, we might need the posteriors of some or all unobserved variables.

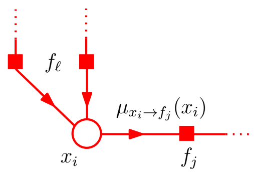
$$p(x_1, x_3, x_4, x_5, x_7 \mid x_2, x_6) = \frac{p(x_{1:7})}{p(x_2, x_6)} = \frac{p(x_{1:7})}{\sum_{x_1, x_3, x_4, x_5, x_7} p(x_{1:7})}$$

Inference in factor graphs (II/V)



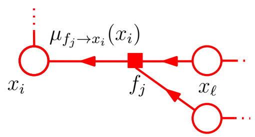
- Performing inference requires marginals.
- It is possible to calculate the marginals on a graph efficiently by passing local messages along the graph.
- Two interconnected types of messages are considered
  - Messages from variable nodes to factor nodes

$$\mu_{x_i \to f_j}(x_i) = \prod_{f_\ell \in \mathrm{ne}(x_i) \setminus f_j} \mu_{f_\ell \to x_i}(x_i)$$



  - Messages from factor nodes to variable nodes

$$\mu_{f_j \to x_i}(x_i) = \sum_{\bar{x}} f_j(x_i, \bar{x}) \prod_{x_\ell \in \mathrm{ne}(f_j) \setminus x_i} \mu_{x_\ell \to f_j}(x_\ell)$$



Inference in factor graphs (III/V)

Sum-product algorithm

- Calculate messages from variable nodes to factor nodes

$$\mu_{x_i \to f_j}(x_i) = \prod_{f_\ell \in \mathrm{ne}(x_i) \setminus f_j} \mu_{f_\ell \to x_i}(x_i)$$

- Calculate messages from factor nodes to variable nodes

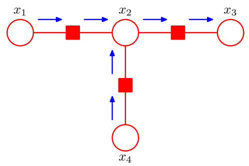
$$\mu_{f_j \to x_i}(x_i) = \sum_{\bar{x}} f_j(x_i, \bar{x}) \prod_{x_\ell \in \mathrm{ne}(f_j) \setminus x_i} \mu_{x_\ell \to f_j}(x_\ell)$$

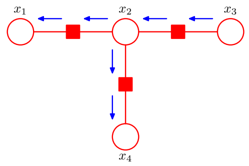
- Iterate messages until convergence. (Different iteration schemes can be designed.)
- After convergence, the marginals are calculated as

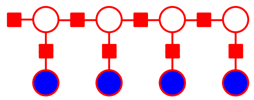
$$p(x_i) \propto \prod_{f_\ell \in \mathrm{ne}(x_i)} \mu_{f_\ell \to x_i}(x_i)$$

Inference in factor graphs (IV/V)

- The values in the observed nodes are just substituted into the factors and not integrated out.
- If the graph is a tree, the algorithm can calculate all the marginals by making
  - a forward pass from the root to the leaves
  - a backward pass from the leaves to the root.
- The sum-product algorithm gives the exact results in a tree structured graph.
- The sum-product algorithm is **equivalent to a Kalman smoother** for linear Gaussian dynamical systems.
- (Chapter 13.3)

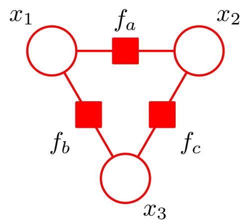






Inference in factor graphs (V/V)

- When the sum-product algorithm is applied to directed graphs without loops the resulting algorithm is sometimes referred to as **belief propagation**.
- In a graph with loops, the sum-product algorithm is not exact and actually might not converge.
- Despite this, it is applied to graphs with loops, which is called **loopy belief propagation**.



Even in this form, it has important applications in communications (decoding of error correcting codes).

Kschischang, F. R., Frey, B. J. and Loeliger, H-A. **Factor graphs and the sum-product algorithm**. *IEEE Transactions on information theory*, 47(2):498–519, 2001.

Current research – SMC for general GMs (I/IV)

Inference in GMs does typically **not** allow for analytical solutions, confining us to various approximative methods (recall the conclusion of Andrew Blake's Gibbs lecture).

Derived a new **sequential Monte Carlo (SMC) algorithm** for inference in general GMs.

Delivers an unbiased estimate of the partition function (normalization constant), can be used within an MCMC sampler for learning.

SMC methods (e.g. particle filters and particle smoothers) can be used to approximate a sequence of probability distributions on a sequence of probability spaces of increasing dimension.

PhD course available on SMC methods

`http://user.it.uu.se/~thosc112/CIDS.html`

Current research – SMC for general GMs (II/IV)

Constructing an artificial sequence of intermediate (auxiliary) target distributions in order to be able to employ an SMC sampler is a powerful (and **quite possibly underutilized**) idea.

Key idea: Perform and make use of a sequential decomposition of the graphical model.

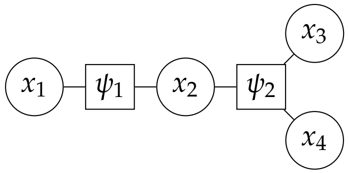
Defines a sequence of intermediate (auxiliary) target distributions defined on an increasing sequence of probability spaces.

Target this sequence using SMC.

Current research – SMC for general GMs (III/IV)

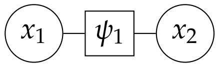
The joint PDF of the set of random variables indexed by $\mathcal{V}$, $X_{\mathcal{V}} \triangleq \{x_1, \dots, x_{|\mathcal{V}|}\}$

$$p(X_{\mathcal{V}}) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(X_C).$$

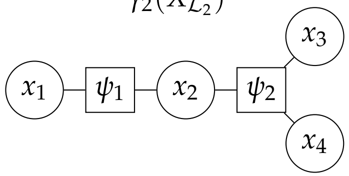


Sequential decomposition of the above factor graph (the target distributions are built up by adding factors at each iteration),

$\gamma_1(X_{\mathcal{L}_1})$



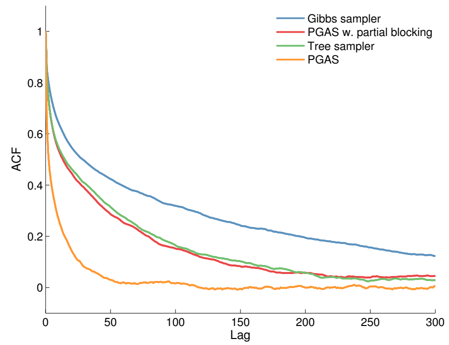
$\gamma_2(X_{\mathcal{L}_2})$



Current research – SMC for general GMs (IV/IV)

Consider a standard squared lattice Gaussian MRF of size 10×10,

$$p(X_{\mathcal{V}}, Y_{\mathcal{V}}) \propto \prod_{i \in \mathcal{V}} e^{\frac{1}{2\sigma_i^2}(x_i - y_i)^2} \prod_{(i,j) \in \mathcal{E}} e^{\frac{1}{2\sigma_{ij}^2}(x_i - x_j)^2}$$



Full details and a loopy, non-Gaussian and non-discrete PGM example,

Christian A. Naesseth, Fredrik Lindsten and Thomas B. Schön, **Sequential Monte Carlo methods for graphical models**. *Preprint at arXiv:1402:0330*, February, 2014.

A few concepts to summarize lecture 9

Markov random fields: (Undirected graphs, no directed arrows) A graphical representation where conditional independence is given by graph separation.

conditional random field (CRF): A CRF is a particular MRF where all the clique potentials are conditioned on input features.

Tree: In an undirected graph a tree is defined as a graph where there is one, and only one, path between any pair of nodes.

Factor graphs: An extension of directed and undirected graphs which makes the probabilistic factors explicit.

Belief propagation: A message passing algorithm for performing inference on graphical models, where local messages are propagated among the graph nodes.