Lecture 12
Graphical Models

Dr Christopher Brunner
AMME4710 – Computer Vision and Image Processing
Semester 2, 2013

By the end of this lecture you will be able to:

• Appreciate the importance of graphical models in computer vision
• Understand the different types of graphical models and their applications
• Describe the main approaches for learning and inference in graphical models
Resources

• Readings
  – Szeliski, Computer Vision, Appdx. B and Chp. 3.7
    http://szeliski.org/Book/

• Additional material (optional)

• Slides credits
  – C. Bishop, G. Hinton, L. Getoor, M. Jordan, A. Simma, S. Rosenberg, K. Murphy, A. Smola
Target Tracking

Radar-based tracking of multiple targets

Visual tracking of articulated objects

(L. Sigal et. al., 2006)

• Estimate motion of targets in 3D world from indirect, potentially noisy measurements
Robot Navigation: SLAM
Simultaneous Localization and Mapping

- As robot moves, estimate its pose & world geometry

Landmark SLAM
(E. Nebot, Victoria Park)

CAD Map
(S. Thrun, San Jose Tech Museum)

Estimated Map
Speech Recognition

• Given an audio waveform, would like to robustly extract & recognize any spoken words

• Statistical models can be used to
  – Provide greater robustness to noise
  – Adapt to accent of different speakers
  – Learn from training

S. Roweis, 2004
Financial Forecasting

- Predict future market behavior from historical data, news reports, expert opinions, …
Temporal models can be adapted to exploit more general forms of *sequential* structure, like those arising in DNA sequences.
Graphical Models

• Examples:
  – Bayesian networks
  – Bayes nets
  – Belief nets
  – Markov networks
  – Hidden Markov models
  – Conditional random fields
  – Dynamic Bayes nets
  – Neural nets
  – Deep belief nets
  – Kalman filter
  – Particle filter, etc

• Applications:
  – Robotics
  – Computer vision
  – Image processing
  – Speech processing
  – Natural language processing
  – Document processing
  – Pattern recognition
  – Bioinformatics
  – Economics
  – Physics
  – Social sciences, etc
Analysis of Sequential/Structured Data

• Structured data arises in a huge range of applications
  – Repeated measurements of a temporal process
  – Online decision making & control
  – Text, biological sequences, etc

• Standard machine learning methods are often difficult to directly apply
  – Do not exploit temporal/spatial correlations
  – Computation & storage requirements typically scale poorly to realistic applications
Computer Vision Applications

• Segmentation
• Stereo
• Motion estimation
• Labeling shading and reflectance
• Many others…
Outline

• Basics of graphical models
  – Directed, undirected, and factor graphs
  – Examples

• Inference
  – Message passing

• Learning
  – Parameters
BASICS OF GRAPHICAL MODELS
Key aspects

• Separation between knowledge and reasoning
  – Formulation of the model (graph structure)
  – Inference algorithms

• Graphical models provide a rich framework for describing large-scale multivariate statistical models
  – Set of (conditional) independencies are encoded
Some broad questions

• **Representation:**
  – What phenomena can be captured by different classes of graphical models?
  – Link between graph structure and representational power?

• **Statistical issues:**
  – How to perform inference (data → hidden phenomenon)?
  – How to fit parameters and choose between competing models?

• **Computation:**
  – How to structure computation so as to maximize efficiency?
  – Links between computational complexity and graph structure?
Models
- Mixtures Clusters
- Matrix Factorization
- Factor Models
- Chains HMM
- MRF CRF
- directed undirected

Statistics
- I1, I2 Priors
- Conjugate Prior
- Exponential Families
- I1, I2 Priors

Inference Methods
- Exact
- k-means (direct)
- Gibbs Sampling
- variational EM
- EM

Efficient Computation
- Dynamic Programming
- Message Passing
- Convex Optimization
Probability Theory

Apples and Oranges

Fruit is orange, what is probability that box was blue?
Probability Theory

Marginal Probability

\[ p(X = x_i) = \frac{c_i}{N}. \]

Joint Probability

\[ p(X = x_i, Y = y_j) = \frac{n_{ij}}{N} \]

Conditional Probability

\[ p(Y = y_j | X = x_i) = \frac{n_{ij}}{c_i} \]
Probability Theory

Sum Rule

\[ p(X = x_i) = \frac{c_i}{N} = \frac{1}{N} \sum_{j=1}^{L} n_{ij} \]

\[ = \sum_{j=1}^{L} p(X = x_i, Y = y_j) \]

Product Rule

\[ p(X = x_i, Y = y_j) = \frac{n_{ij}}{N} = \frac{n_{ij}}{c_i} \cdot \frac{c_i}{N} \]

\[ = p(Y = y_j | X = x_i) p(X = x_i) \]
Probability Theory

- **Sum rule**
  \[ p(x) = \sum_y p(x, y) \]

- **Product rule**
  \[ p(x, y) = p(x|y)p(y) \]

- From these we have Bayes’ theorem
  \[ p(y|x) = \frac{p(x|y)p(y)}{p(x)} \]
  - with normalization
  \[ p(x) = \sum_y p(x|y)p(y) \]
A graphical representation of a set of conditional probabilities

- Each node represents a random variable
- Each directed edge represents an explicit dependency on a “parent”
- For general distributions, the graph is fully connected

\[ p(a, b, c) = p(c \mid a, b)p(b \mid a)p(a) \]
Directed Acyclic Graphs

- The structure of a less general distribution can be represented by the missing edges.
- If the directed graph is acyclic and the distribution of each node conditional on its parents is normalized, the whole distribution will be consistent.

\[
p(x) = \prod_{k} p(x_k | p a_k) = p(x_1)p(x_2)p(x_3)p(x_4 | x_1, x_2, x_3) \ldots
\]

parents

\[
p(x_5 | x_1, x_3)p(x_6 | x_4)p(x_7 | x_4, x_5)
\]
Undirected Graphs

- Provided $p(x) > 0$ then joint distribution is product of non-negative functions over the *cliques* of the graph

\[
p(x) = \frac{1}{Z} \prod_C \psi_C(x_C)
\]

where $\psi_C(x_C)$ are the *clique potentials*, and $Z$ is a normalization constant

\[
p(w, x, y, z) = \frac{1}{Z} \psi_A(w, x, y) \psi_B(x, y, z)
\]
Conditioning on Evidence

- Variables may be hidden (latent) or visible (observed)

- Latent variables may have a specific interpretation, or may be introduced to permit a richer class of distribution
Conditional Independences

• $x$ independent of $y$ given $z$ if, for all values of $z$,

$$p(x, y|z) = p(x|z)p(y|z)$$

• For undirected graphs this is given by graph separation!
“Explaining Away”

- C.I. for directed graphs is similar, but with one subtlety
- Illustration: pixel colour in an image

\[
p(I, L, S) = p(I|L, S)p(L)p(S)
\]

\[
p(L, S) = p(L)p(S)
\]

\[
p(L, S|I) \neq p(L|I)p(S|I)
\]
Bayesian polynomial regression

\[ p(t \mid x, x_t) = \int p(t \mid x, w) p(w \mid x, t) dw \]

- The modeled random variables are \( t \) and \( w \)
- The inputs, \( x \), are given. They are not random variables in the model.
- The “plate” notation is used for multiple variables with the same dependencies.

\[ p(t, w) = p(w) \prod_{n=1}^{N} p(t_n \mid w) \]
Showing dependencies on deterministic parameters

- We can use a small solid circle for a parameter such as:
  - Output noise variance
  - Input vector for a case
  - Parameter determining the prior distribution of the weights.

\[
p(t, w \mid x, \alpha, \sigma^2) = p(w \mid \alpha) \prod_{n=1}^{N} p(t_n \mid w, x_n, \sigma^2)
\]
The distributions for which directed and undirected graphs can give perfect maps

- A graph is a perfect map of a distribution if its conditional independencies are exactly the same as those in the distribution.
## Markov Nets vs. Bayes Nets

<table>
<thead>
<tr>
<th>Property</th>
<th>Markov Nets</th>
<th>Bayes Nets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Form</td>
<td>Prod. potentials</td>
<td>Prod. potentials</td>
</tr>
<tr>
<td>Potentials</td>
<td>Arbitrary</td>
<td>Cond. probabilities</td>
</tr>
<tr>
<td>Cycles</td>
<td>Allowed</td>
<td>Forbidden</td>
</tr>
<tr>
<td>Partition func.</td>
<td>$Z = ?$</td>
<td>$Z = 1$</td>
</tr>
<tr>
<td>Indep. check</td>
<td>Graph separation</td>
<td>D-separation</td>
</tr>
<tr>
<td>Indep. props.</td>
<td>Some</td>
<td>Some</td>
</tr>
<tr>
<td>Inference</td>
<td>MCMC, BP, etc.</td>
<td>Convert to Markov</td>
</tr>
</tbody>
</table>
Factor graphs: A better graphical representation for undirected models with higher-order factors

- Each potential has its own factor node that is connected to all the terms in the potential.
- Factor graphs are always bipartite.

\[ p(\mathbf{x}) = \prod_s f_s(\mathbf{x}_s) \]

If the potentials are not normalized we need an extra factor of 1/Z.
‘Supervised’ Models

Classification
Regression

spam filtering
tiering
crawling
categorization
bid estimation
tagging
‘Unsupervised’ Models

Density Estimation

Clustering

Novelty Detection
forecasting
intrusion detection

webpages
news
users
ads
queries
images
‘Unsupervised’ Models

Density Estimation

Factor Analysis
Chains

Markov Chain

Hidden Markov Model
Kalman Filter
Given $n$ observations $x_1, x_2, \ldots, x_n$, we model each $x_i$ with a latent factor $\phi_i$:

$$x_i | \phi_i \sim F(\phi_i)$$

$$\phi_i | G \sim G$$

We put a Dirichlet process prior on $G$

$$G \sim DP(\alpha_0, G_0)$$

The marginal distribution on the theta’s obtained by marginalizing out the Dirichlet process is the *Chinese restaurant process*
Deep Networks

- Assume a layered graph structure.
- How many layers? How wide should each layer be?
- Graph structure can be learned by an *Indian Buffet Process*
- The IBP defines a distribution on sparse binary matrices with a countably infinite number of columns.

\[
p(x) = \prod_{i=1}^{K} p(x_i|x_{\pi_i})
\]

\[
Z^{(m)} \sim IBP(\alpha, \beta) \quad \text{for} \ m = 0, 1, 2, \ldots
\]
Image denoising with an MRF

• The true value of a pixel is \( x \) and the measured noisy value is \( y \)

• We can define an energy function on pairs of nodes

\[
E(x, y) = h \sum_i x_i - \beta \sum_{i<j} x_i x_j + \eta \sum_i (x_i - y_i)^2
\]

\[
(x_i - y_i)^2 = x_i^2 - 2x_i y_i + y_i^2
\]

so we could use \( -x_i y_i \)
A simple, greedy MAP inference procedure

- **Iterated conditional modes:** Visit the unobserved nodes sequentially and set each $x$ to whichever of its two values has the lowest energy
  - This only requires us to look at the Markov blanket, i.e. the connected nodes
- It would be better to flip in order of confidence
MRF nodes as patches

\[ \Phi(x_i, y_i) \]

\[ \Psi(x_i, x_j) \]
Conditional Random Field

Lafferty, McCallum and Pereira 2001

\[
p(h | I, \theta) = \frac{1}{Z(I, \theta)} \left[ \prod_i \phi_i(h_i, I | \theta_i) \prod_{ij} \psi_{ij}(h_i, h_j, I | \theta_{ij}) \right]
\]

- Dependency on \( I \) allows introduction of pairwise terms that make use of image.

- For example, neighboring labels should be similar only if pixel colors are similar \( \rightarrow \) Contrast term

\[ \text{e.g. Kumar and Hebert 2003} \]

I (pixels)
Probabilistic Latent Semantic Analysis (pLSA)

Sivic et al. ICCV 2005
Latent Dirichlet Allocation (LDA)

Fei-Fei et al. ICCV 2005

“beach”