CRF - session 2

Formal introduction

Amir Ghodrati

August 2013

Agenda

- Introduction
- Graphical Models
- Naïve-Bayes
- Logistic Regression
- Hidden Markov Models
- Conditional Random Fields

Real Introduction
(longest one ever
in the world)

Historical view

- Energy functions like what we have in CRFs go back at least as far as Horn & Schunk (1981)
- The Bayesian view was popularized by Geman and Geman (TPAMI 1984)
- Starting in the **late** 90's researchers re-discovered discrete optimization methods!
  - Graph cuts, belief prop, semi-definite programming, etc.

What we will explain

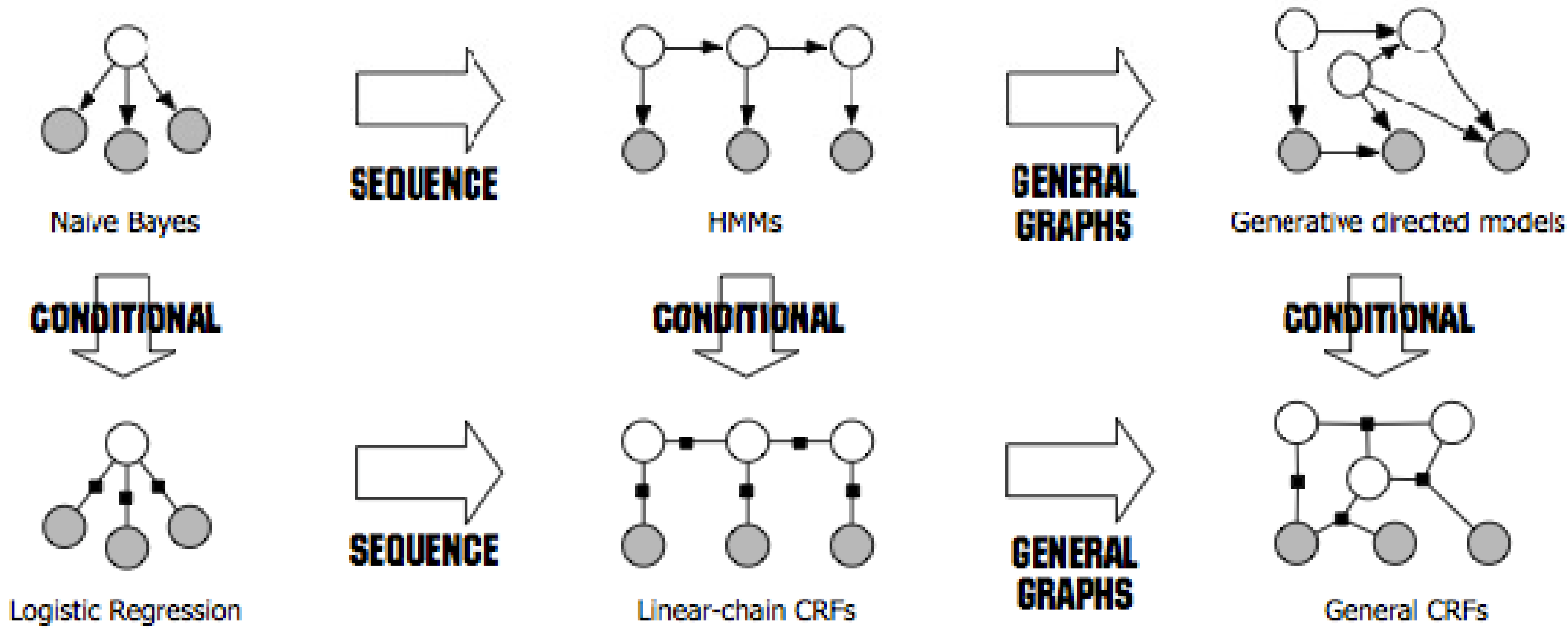


Figure 1.2 Diagram of the relationship between naive Bayes, logistic regression, HMMs, linear-chain CRFs, generative models, and general CRFs.

Introduction – toy example

- assume we have a sequence of snapshots from activities we are doing during one day. We want to label each image, x_i, with the activity it represents, y_i.

- simple approach: per-image classifier
  - Employ logistic regression as a discriminative log-linear model for classification
  - we lose a lot of information

- so what we can do? incorporate the labels of nearby images (we want sequential graphical model)
  - Employ CRF as a log-linear discriminative model for sequential labeling

A note on graphical models

- A graph which nodes are random variables
- We always have (chain rule)

$$p(x_1,\ldots,x_n \mid y) = p(x_n \mid x_{n-1},\ldots,x_1,y)\,p(x_{n-1} \mid x_{n-2},\ldots,x_1,y)\ldots p(x_1 \mid y)$$

$$p(x_1,x_2 \mid y) = p(x_2 \mid x_1,y)\,p(x_1 \mid y)$$

Conditional independency:

$$p(x_1,x_2 \mid y) = p(x_2 \mid y)\,p(x_1 \mid y)$$

A note on graphical models

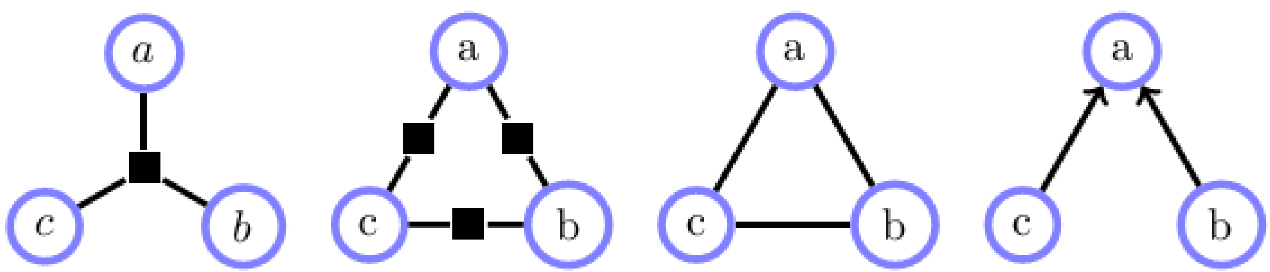
- **independency** as an important concept as it can be used to decompose complex probability distributions => makes complex computations more efficient

- GMs model independency between random variables (i.e. absence of edges is informative)

- => decompose complex probability distributions

A note on graphical models

- Belief networks -> directed graphs
- Markov networks -> undirected graphs
- Factor graphs connects factors and random variables. Each factor is a function(not necessarily a probability distribution) defined over the random variables it is connected to.
- Both directed/undirected graphs can be transformed to factor graphs



(a) (b) (c) (d)

$\phi(a,b,c)$ $\phi(a,b)\phi(b,c)\phi(c,a)$ $\phi(a,b,c)$ $p(a,b,c) = p(a \mid b,c)\,p(b)\,p(c)$

A note on graphical models

- **Factor graph** decompose the distributions into its factors.

$$p(\vec{v}) = \frac{1}{Z}\prod_{s}\Psi_s(v_s)$$

Ψ_s are so-called potentials. Should be positive

S is a subset of random variables. Usually maximal cliques (a set of nodes that make complete graph)

Naïve Bayes

- A generative approach model joint distribution

$$p(y, x) = p(y)\,p(x \mid y)$$

- Too complex to compute directly

$$x = [x_1, \ldots, x_n]$$

- Are all random variables x really dependent to each other?

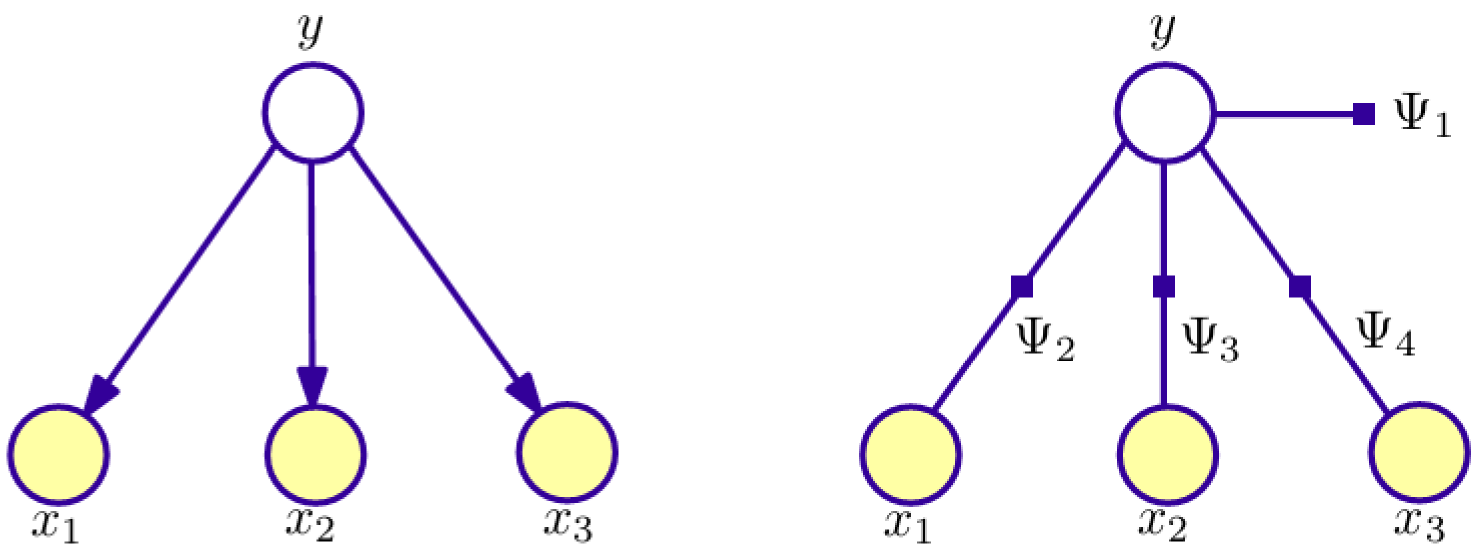
Naïve Bayes

- **Naive Bayes assumption**: all input variables x_i are conditionally independent of each other

$$p(y, x) = p(y) \prod_i p(x_i \mid y)$$

- (in)dependencies are not modeled.
- performs surprisingly well in many real world applications!

Naïve Bayes



(a) Independency graph (b) Factor graph

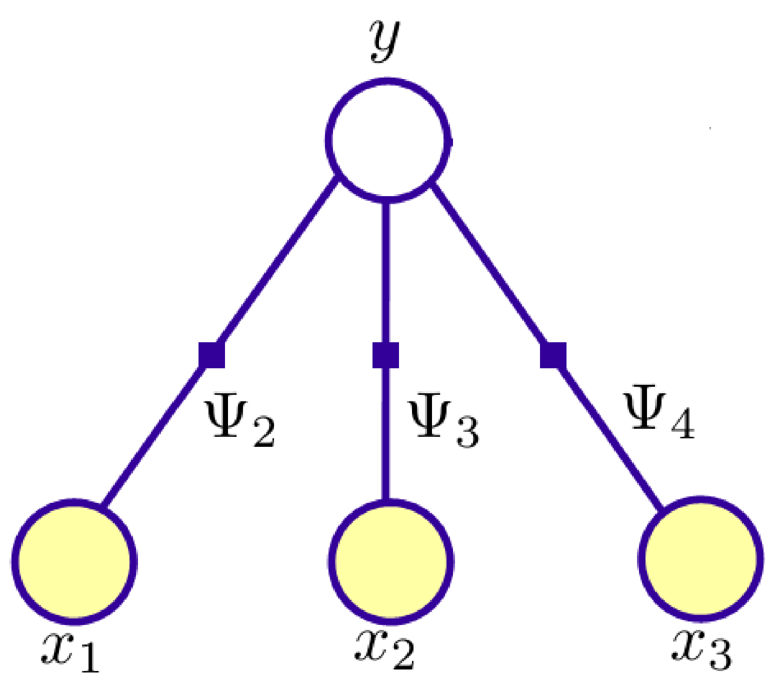
$$p(x_1, x_2, x_3, y) = p(x_1 \mid y)\, p(x_2 \mid y)\, p(x_3 \mid y)\, p(y)$$

$$p(x_1, x_2, x_3, y) = \Psi_1(x_1, y)\Psi_2(x_2, y)\Psi_3(x_3, y)\Psi_4(y)$$

Logistic regression

- Sometimes known as maximum entropy classifier in NLP community)
- A discriminative approach => model conditional probability $p(y|x)$

Logistic regression



$$p(y \mid \vec{x}) = \frac{1}{Z} \prod_{i=1}^{m} \exp(\lambda_i f_i(x, y))$$

Logistic regression

- Is not similar to factorization of distribution?
- potential functions = exponential function of weighted features

linear model **ax**+b

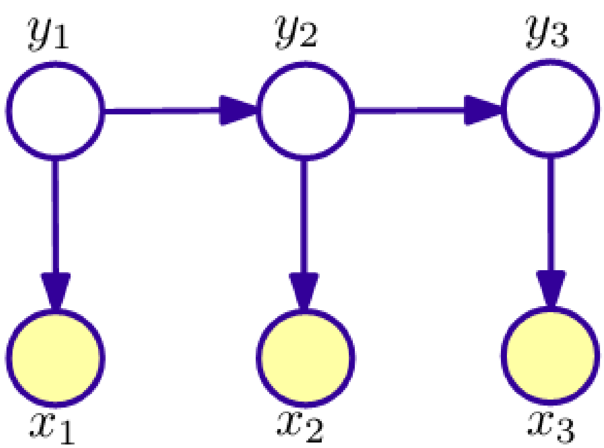
$$\Psi_i = \exp(\lambda_i f_i(x, y))$$

- fulfils the requirement of strict positivity of the potential functions

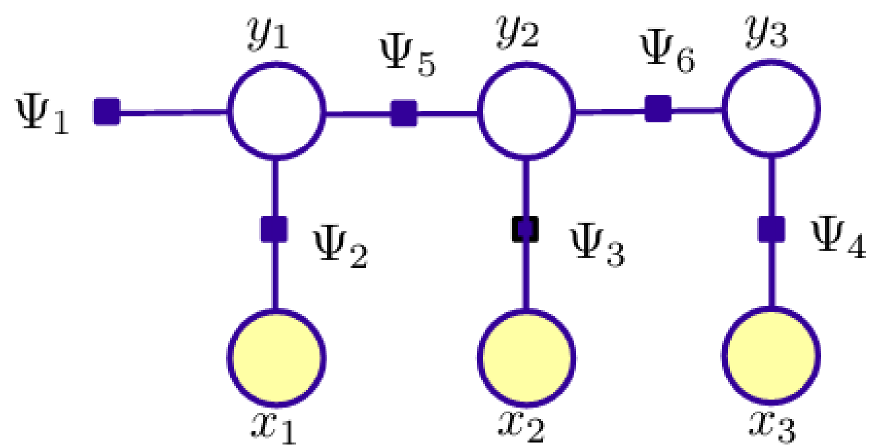
Hidden Markov Models (HMMs)

- Classifiers like Naïve-Bayes predict only a single class variable
- Suppose we want to do labeling in a sequences of images. It is reasonable to consider dependencies between the labels at consecutive sequence
  - sleep, sleep, travel, sleep, sleep
  - sleep, sleep, check mail, sleep, sleep
- A sequential version of Naïve-Bayes. (labels are not independent)

Hidden Markov Models (HMMs)



(a) Independency graph



(b) Factor graph

$$p(x_1, x_2, x_3, y_1, y_2, y_3) = p(y_1)\, p(x_1 \mid y_1)$$
$$p(y_2 \mid y_1)\, p(x_2 \mid y_2)\, p(y_3 \mid y_2)\, p(x_3 \mid y_3)$$

Hidden Markov Models (HMMs)

$$p(\vec{x}, \vec{y}) = \prod_{i=1}^{n} p(y_i \mid y_{i-1}) p(x_i \mid y_i)$$

- Again a generative model
- We will back to HMMs to have a comparison with CRFs

Finally CRF comes in

Conditional Random Fields

- A sequential version of logistic regression so it is a discriminative model as well.
- HMMs are tied to linear-sequence structure but CRFs can have arbitrary structures.
- We have a sequence of labels y (e.g. sleeping-drinking- sleeping again)

Conditional Random Fields

- Starting with

$$p(\vec{v}) = \frac{1}{Z}\prod_s \Psi_s(v_s)$$

$$p(\vec{y} \mid \vec{x}) = \frac{p(\vec{y},\vec{x})}{p(\vec{x})} = \frac{p(\vec{y},\vec{x})}{\sum_y p(\vec{y},\vec{x})}$$

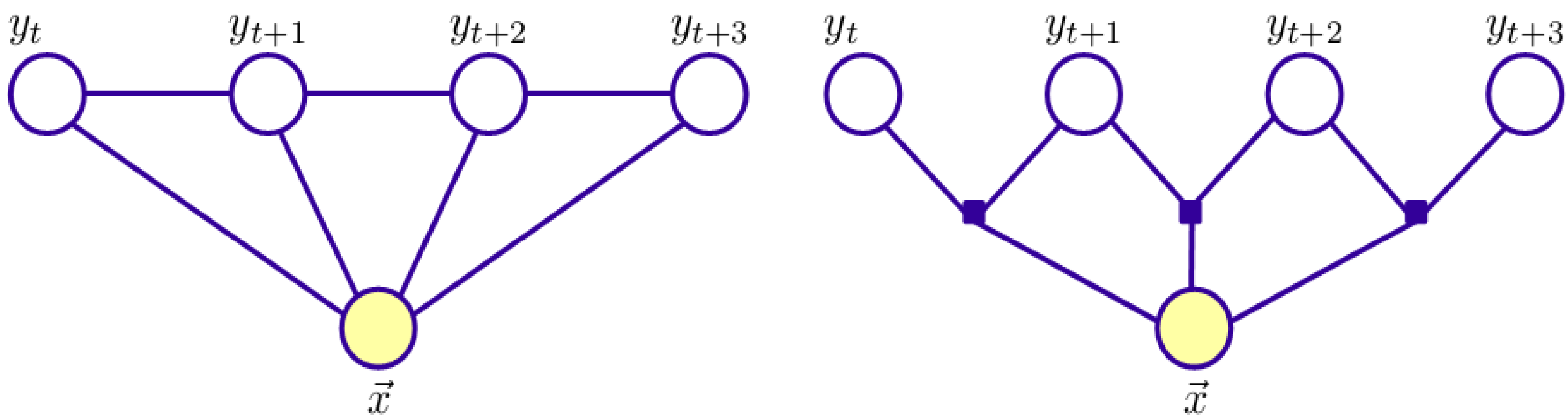
$$= \frac{\frac{1}{Z}\prod_s \Psi_s(\vec{x}_s, \vec{y}_s)}{\sum_y \frac{1}{Z}\prod_s \Psi_s(\vec{x}_s, \vec{y}_s)}$$

Conditional Random Fields

$$p(\vec{y} \mid \vec{x}) = \frac{1}{Z(\vec{x})} \prod_s \Psi_s(\vec{x}_s, \vec{y}_s)$$

Ψ_s is the factor corresponding to maximal clique s

Conditional Random Fields

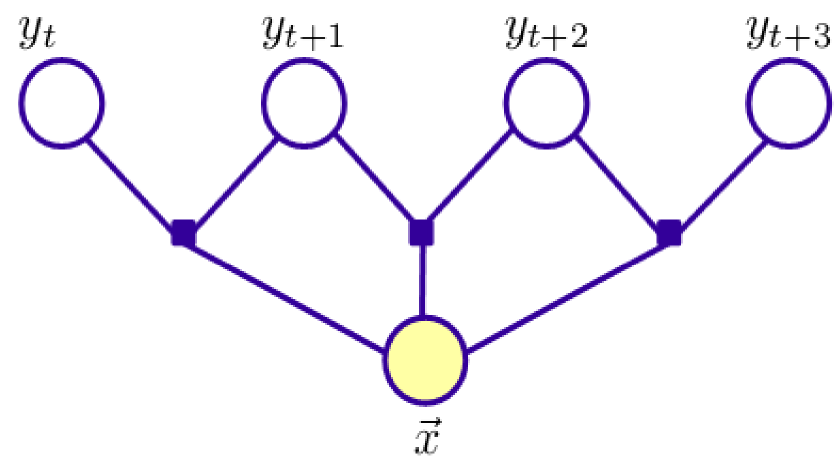


(a) Independency graph (b) Factor graph

$$p(\vec{y} \mid \vec{x}) = \Psi_1(y_t, y_{t+1}, \vec{x})\Psi_2(y_{t+1}, y_{t+2}, \vec{x})\Psi_3(y_{t+2}, y_{t+3}, \vec{x})$$

Conditional Random Fields

To define feature functions we can use observations from **any** time step, that is because we have written the observation vector x in one node. For e.g. it is possible to use the next image xt+1 to define a feature



(b) Factor graph

$$(y_{t+1}, y_{t+2}, \vec{x})\Psi_3(y_{t+2}, y_{t+3}, \vec{x})$$

Conditional Random Fields

- Now assume each potential function is a logistic function

$$\Psi_s(\vec{x}, y_s) = \exp(\sum_i \lambda_i f_i(\vec{x}, y_s))$$

- For example for a linear-chain CRFs

$$\Psi_s(\vec{x}, y_s) = \exp(\sum_i \lambda_i f_i(\vec{x}, y_j, y_{j-1}, j))$$

Conditional Random Fields

- So for a linear-chain CRF, the overall conditional probability is

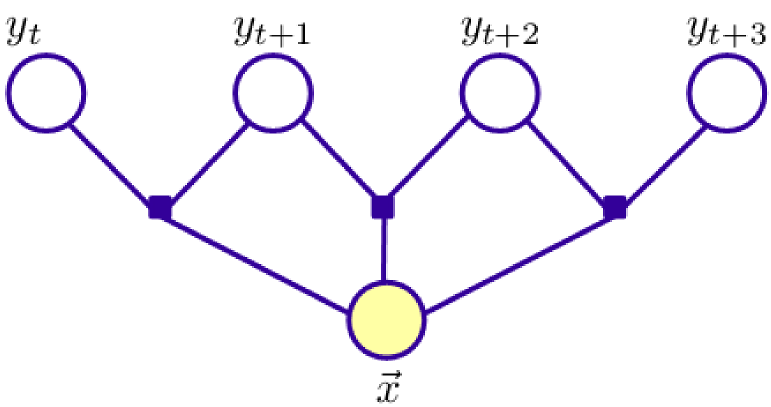
$$p(\vec{y} \mid \vec{x}) = \frac{1}{Z(\vec{x})} \exp(\sum_{j} \sum_{i} \lambda_i f_i(\vec{x}, y_j, y_{j-1}, j))$$

- The outer sum runs over each potential function j out of n frames of video.
- The inner sum runs over each feature i out of m features

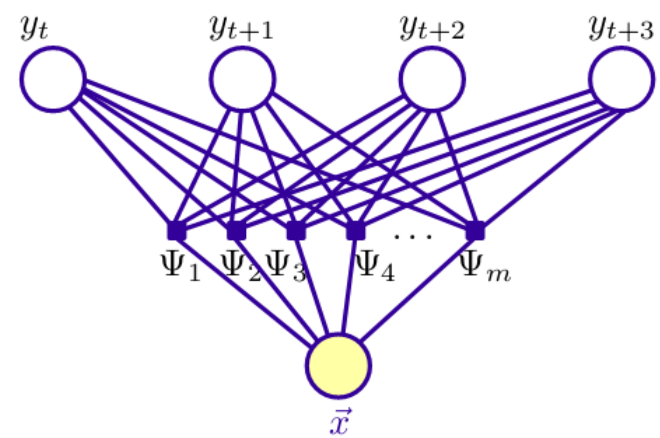
Conditional Random Fields

- Play with CRF equation result in different graphs

$$p(\vec{y} \mid \vec{x}) = \frac{1}{Z(\vec{x})} \prod_{j=1}^{n} \exp\left(\sum_{i} \lambda_i f_i(\vec{x}, y_j, y_{j-1}, j)\right)$$



$$p(\vec{y} \mid \vec{x}) = \frac{1}{Z(\vec{x})} \prod_{i=1}^{m} \exp\left(\sum_{j} \lambda_i f_i(\vec{x}, y_j, y_{j-1}, j)\right)$$



Notice to its similarity to logistic regression

CRFs

- Inference: Given observation x and a CRF λ: find the most probably fitting label sequence y
- Training: Given label sequences Y and observation sequences X: find parameters of a CRF, weights λ, to maximize $p(y|x; \lambda)$.

CRFs - Training

- MLE of model parameters λ
- regularization terms are often added to prevent over-fitting
- For linear-chain CRFs, (log-)likelihood function is concave (=> easy to maximize)

$$\boldsymbol{\lambda}^* = \arg\min_{\boldsymbol{\lambda}} L(\boldsymbol{\lambda}, D) + C\frac{1}{2}\|\boldsymbol{\lambda}\|^2$$

$$L(\boldsymbol{\lambda}, D) = -\log\left(\prod_{k=1}^{m} P(\mathbf{y}^k \mid \mathbf{x}^k, \boldsymbol{\lambda})\right)$$

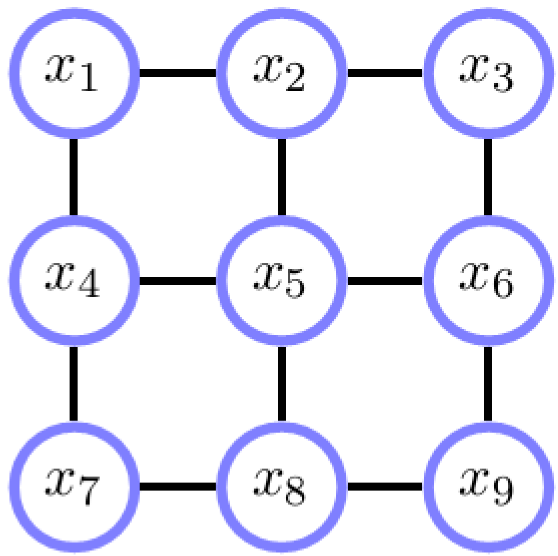
$$= -\sum_{k=1}^{m} \log\left[\frac{1}{Z(\mathbf{x}_m)} \exp \sum_{i=1}^{n} \sum_{j} \lambda_j f_j(y_{i-1}^k, y_i^k \ , \mathbf{x}^m, i)\right]$$

CRFs - Inference

- It is all about optimization.
- Belief propagation, Linear programming relaxations, Dual decomposition, Psedo-boolean optimization, . . .

- the well-known method, graph-cut, will be discussed next session

CRFs for images



- Consider image as a field of random variables
- **unary** potentials + **binary** potentials

CRFs for images

- Negative Log-likelihood of p(y|x) gives the so-called energy function

$$p(y \mid x) = \prod_{j=1}^{n} e^{-\Phi(y_p;x)} \prod_{p \propto q} e^{-\Psi(y_p, y_q; x)}$$

$$E(y_1, \ldots, y_n; x) = \sum_{p} \Phi(y_p; x) + \sum_{p \propto q} \Psi(y_p, y_q; x)$$

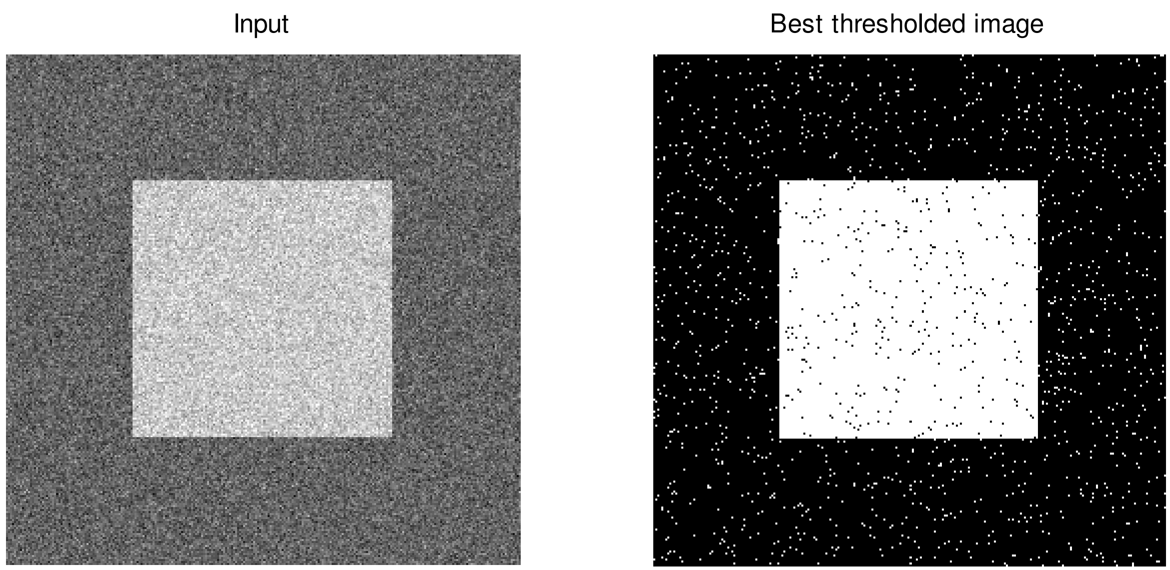
- Non-convex with thousands of dimension

- prior term
- unary term

- pair wise term
- smoothness term
- binary term

CRFs for images

- Segmentation as an intuitive problem



- If we only have unary term, the cheapest solution is the thresholded output
- The functionality of binary term is to keep the smoothness

connection to HMM

- long story short: CRFs are more powerful – they can model everything HMMs can and more:

$$p(\vec{x}, \vec{y}) = \prod_{i=1}^{n} p(y_i \mid y_{i-1})\, p(x_i \mid y_i)$$

$$\log(p(\vec{x}, \vec{y})) = \sum_i \log(p(y_i \mid y_{i-1})) + \sum_i \log(p(x_i \mid y_i))$$

connection to HMM

- For every state $p(y_i = A \mid y_{i-1} = B)$ define

$$f_{AB}(y_i, y_{i-1}, i, x) = [y_i = A, y_{i-1} = B]$$

$$\lambda_{AB} = \log(p(y_i = A \mid y_{i-1} = B))$$

- Do the same for $p(x_i = C \mid y_i = D)$
- [.] is indicator function
- =>Proportional to the score of CRFs $e^{\sum \lambda_{AB} f_{AB}}$

Is vision solved?
Can we all go home now?

- For many easy problems the technical problem of minimizing the energy is now effectively solved
  - Easy = sub-modular/regular, & first-order
  - Technical problem ≠ vision problem
  - "The energy"? Is the right one obvious??
- Still, this is vast progress in a relatively short period of time
  - These "easy" problems were impossible in '97!

What we explained

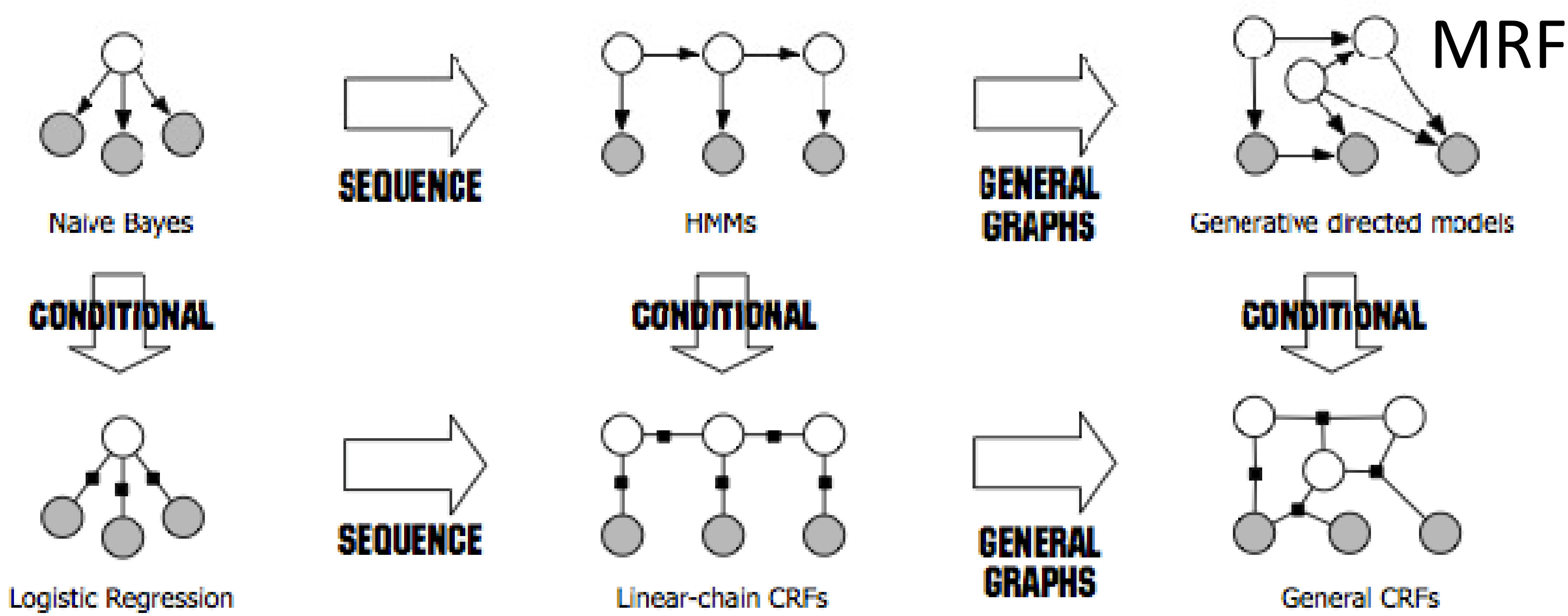


Figure 1.2 Diagram of the relationship between naive Bayes, logistic regression, HMMs, linear-chain CRFs, generative models, and general CRFs.

References

Edwin Chen's Blog

Blog | Archives

JAN 3RD, 2012

Introduction to Conditional Random Fields

technische universität dortmund

Classical Probabilistic Models and Conditional Random Fields

Roman Klinger
Katrin Tomanek

Algorithm Engineering Report
TR07-2-013
December 2007
ISSN 1864-4503

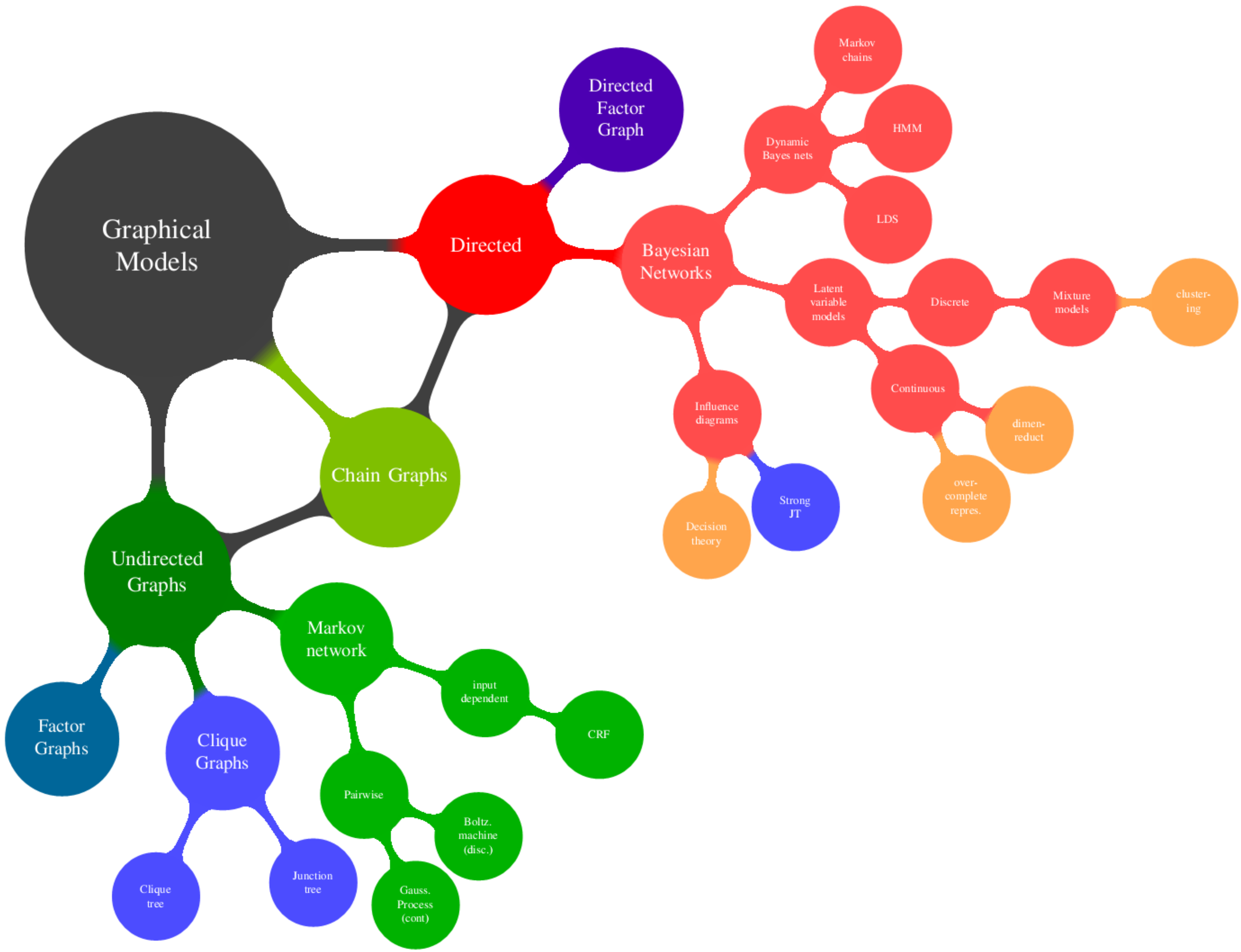
Bayesian Reasoning and Machine Learning

David Barber ©2007,2008,2009,2010,2011,2012

An Introduction to Conditional Random Fields

Charles Sutton
University of Edinburgh
csutton@inf.ed.ac.uk

Andrew McCallum
University of Massachusetts Amherst
mccallum@cs.umass.edu


Graphical Models
Directed
Directed Factor Graph
Bayesian Networks
Dynamic Bayes nets
Markov chains
HMM
LDS
Latent variable models
Discrete
Mixture models
cluster-ing
Continuous
dimen-reduct
over-complete repres.
Influence diagrams
Decision theory
Strong JT
Chain Graphs
Undirected Graphs
Markov network
input dependent
CRF
Pairwise
Boltz. machine (disc.)
Gauss. Process (cont)
Factor Graphs
Clique Graphs
Clique tree
Junction tree