Lecture 2: Undirected Models

Probabilistic Graphical Models, Koller and Friedman:

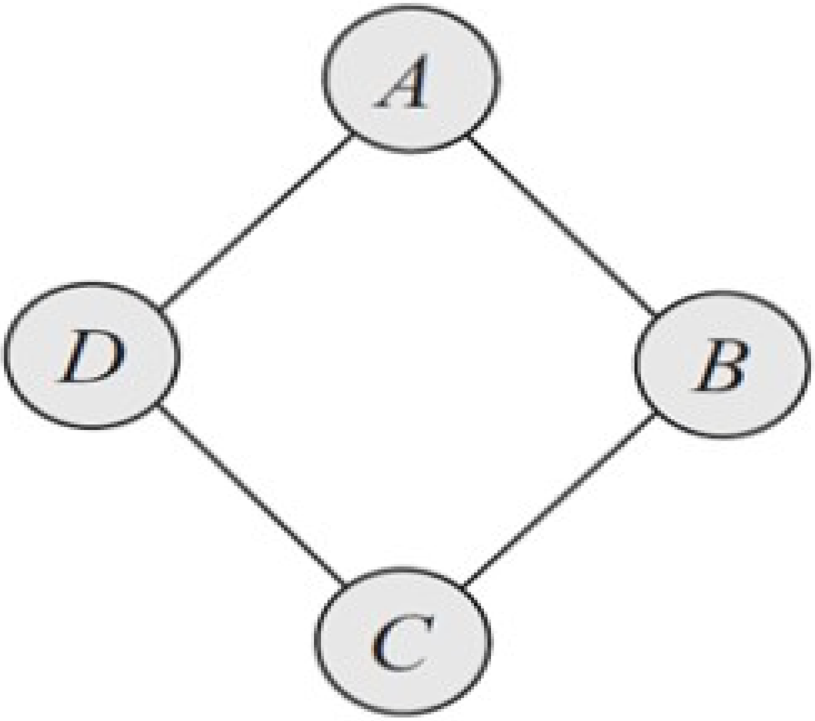
- Chap 4, 7, and 8
- Markov Nets, Max Cliques, Factors, Hammersley-Clifford, Log-Linear Models, Exponential Family, Sufficient Statistics, Entropy, K-L Divergence, I & M Projections.

Factorization

$P(a,b,c,d) \propto \phi1(a,b)\ \phi2(b,c)\ \phi3(c,d)\ \phi4(d,a)$

$\propto [\phi1(a,b)\ \phi2(b,c)\ ]\ [\ \phi3(c,d)\ \phi4(d,a)\ ]$

$\propto F(b, a,c)\ G(d, a,c)$



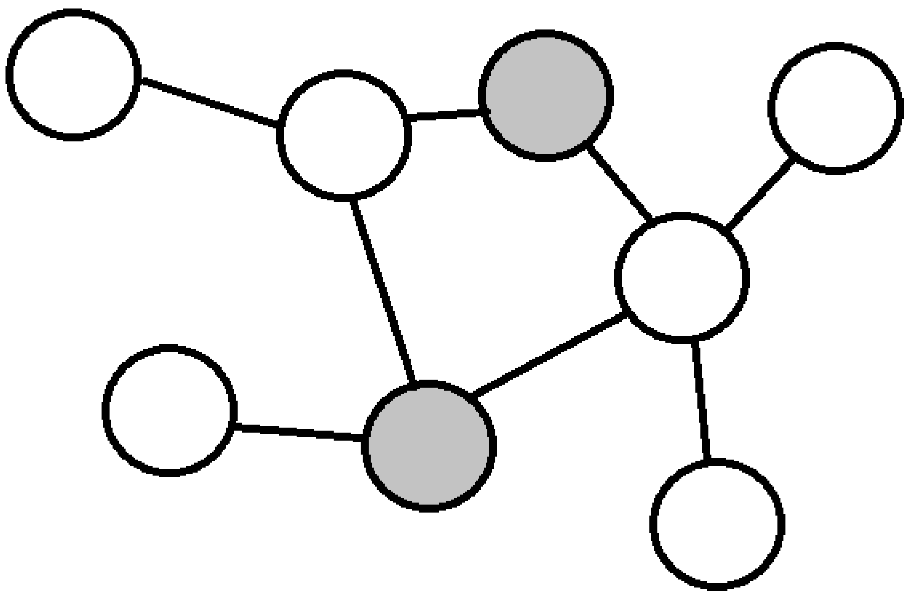
- which implies $D \perp B \mid A,C$

- The normalization constant is the 'partition function' Z.

- $Z=\Sigma_{abcd} \phi1(a,b)\ \phi2(b,c)\ \phi3(c,d)\ \phi4(d,a)$

Blocking is much simpler to see

- If we observed the shaded the two separated unshaded subgraphs are conditionally independent. (**Markov blanket= neighbors**)



Cliques and **Maximal Cliques**

- Blue is a maximal clique.
- Edges describe interaction
- Markov nets correspond to a distribution that can be factored by including a 'factor' for all (maximal) cliques.

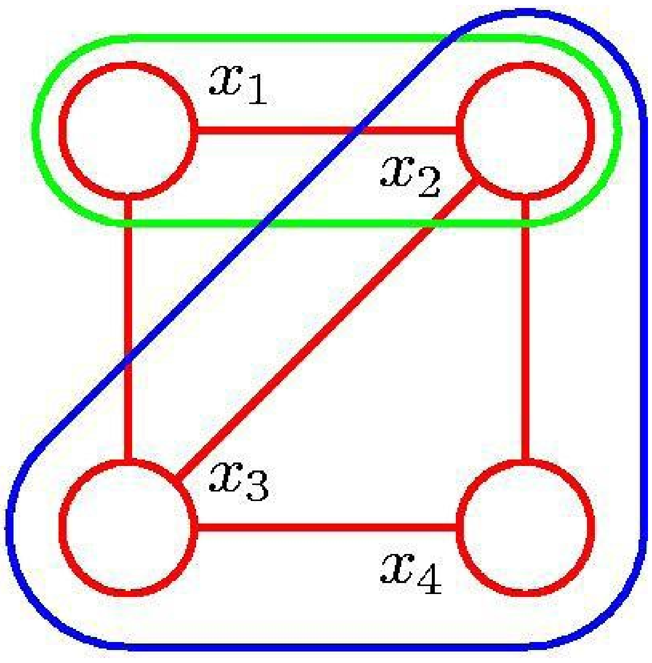


Figure from Bishop- Pattern Recognition and ML

Factorization of Markov Nets

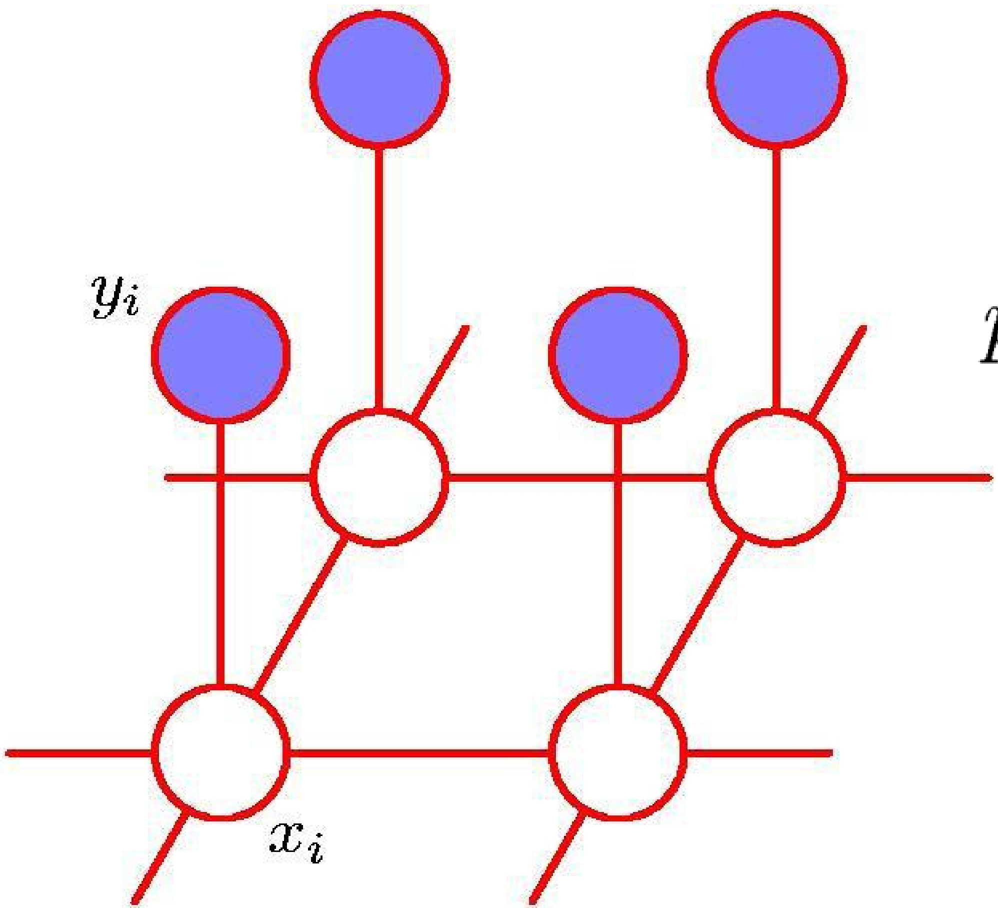
- One factor per maximal clique.
- Factors must be non-negative
- The joint:

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{C} \psi_C(\mathbf{x}_C)$$

- Example Boltzman distribution with Energy terms

$$\psi_C(\mathbf{x}_C) = \exp\left\{-E(\mathbf{x}_C)\right\}$$

Image De-noising



$$E(\mathbf{x}, \mathbf{y}) = h \sum_i x_i - \beta \sum_{\{i,j\}} x_i x_j - \eta \sum_i x_i y_i$$

$$p(\mathbf{x}, \mathbf{y}) = \frac{1}{Z} \exp\{-E(\mathbf{x}, \mathbf{y})\}$$

Figure from Bishop- Pattern Recognition and ML

De-noised on right

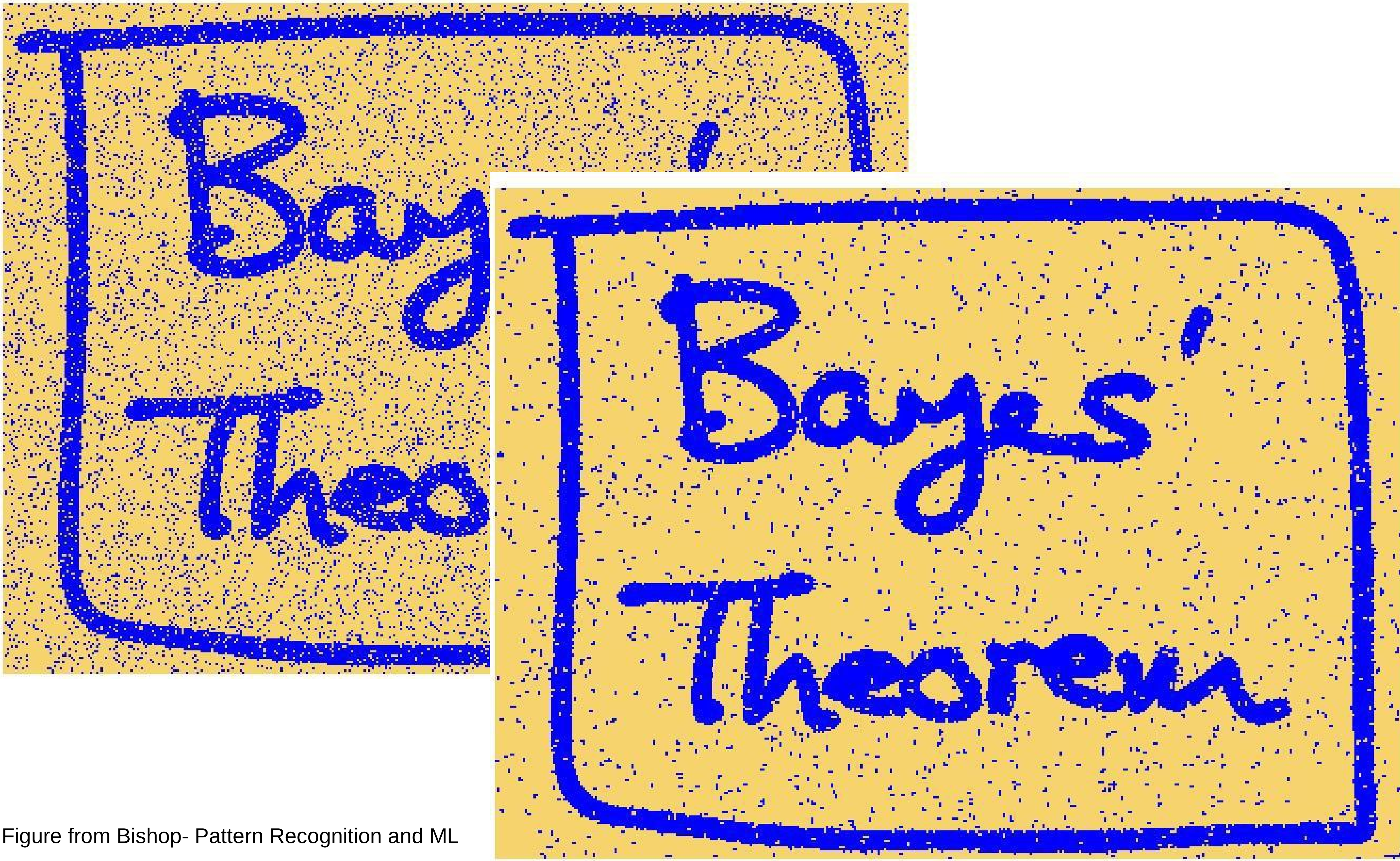
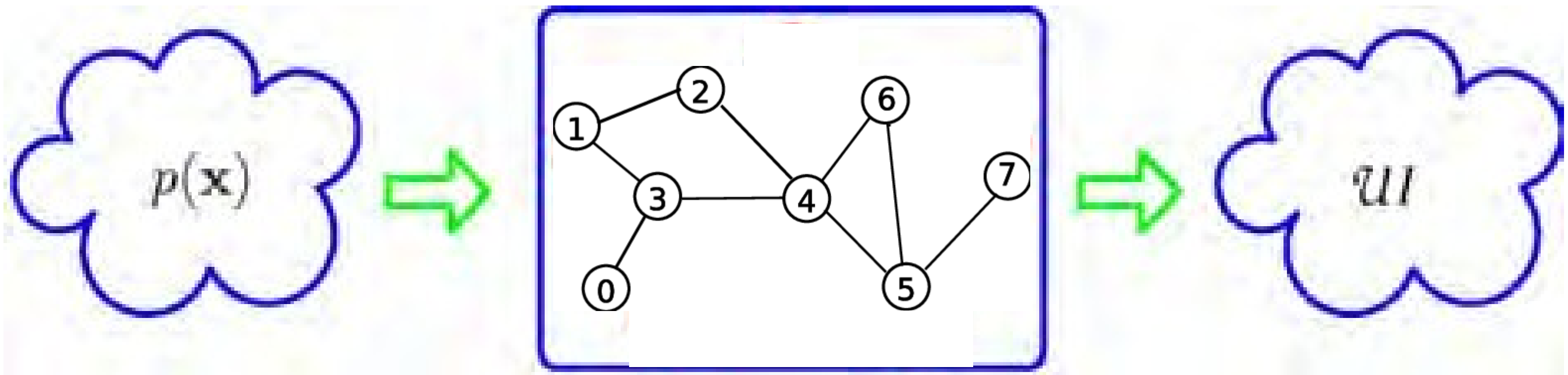


Figure from Bishop- Pattern Recognition and ML

Filter View of a PGM



- Let $\mathcal{UI}$ denote the distributions that can pass
  ie. those that satisfy all conditional independence statements, $\mathcal{I}$(G)
- Let $\mathcal{UF}$ denote the distributions with factorization over cliques
- Hammersley-Clifford says for MRF: $\mathcal{UI} = \mathcal{UF}$ (except if some P=0)
- Similar result for DAG, for example Theorem 3.1:
  which says Graph $\rightarrow$ Factorization for BN

Figure from the lectures of: Bjoern Andres and Bernt Schiele

Directed vs. Undirected Graphs

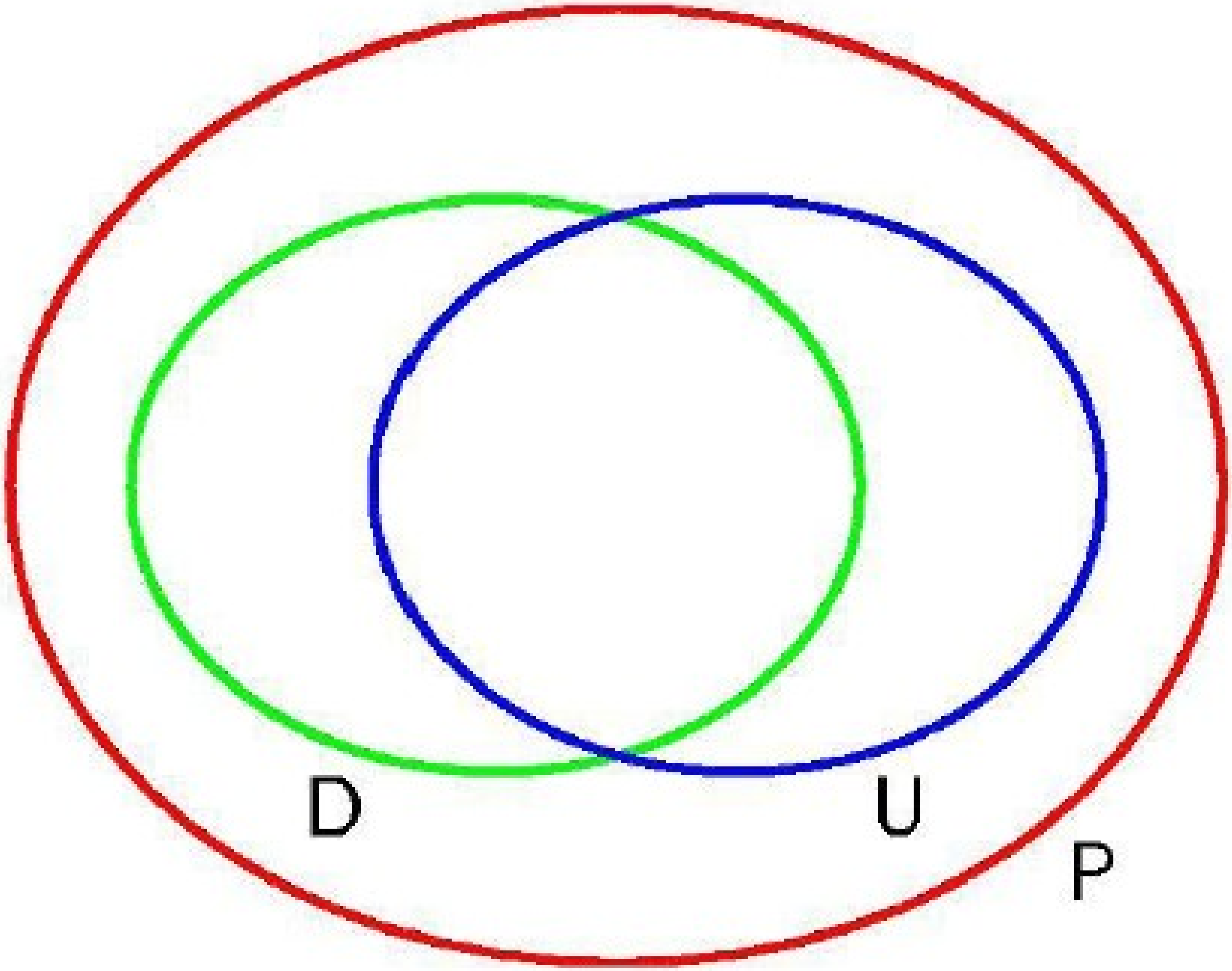
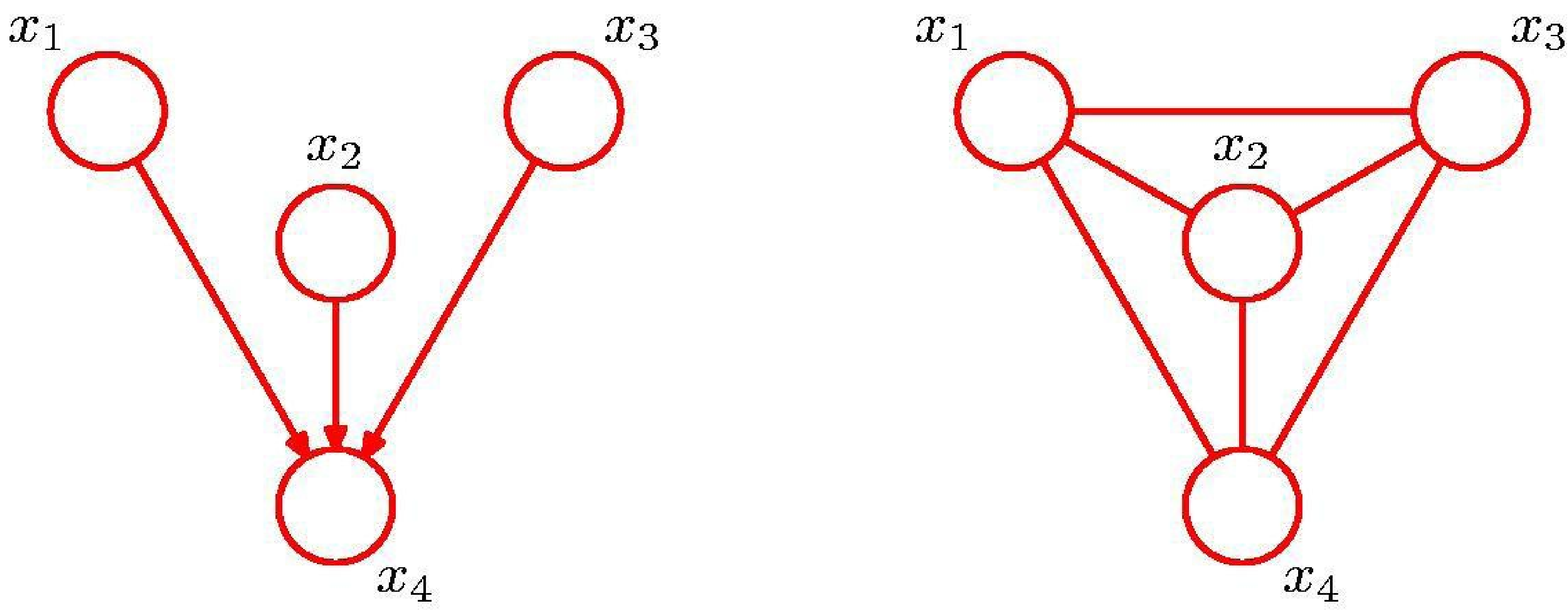


Figure from Bishop- Pattern Recognition and ML

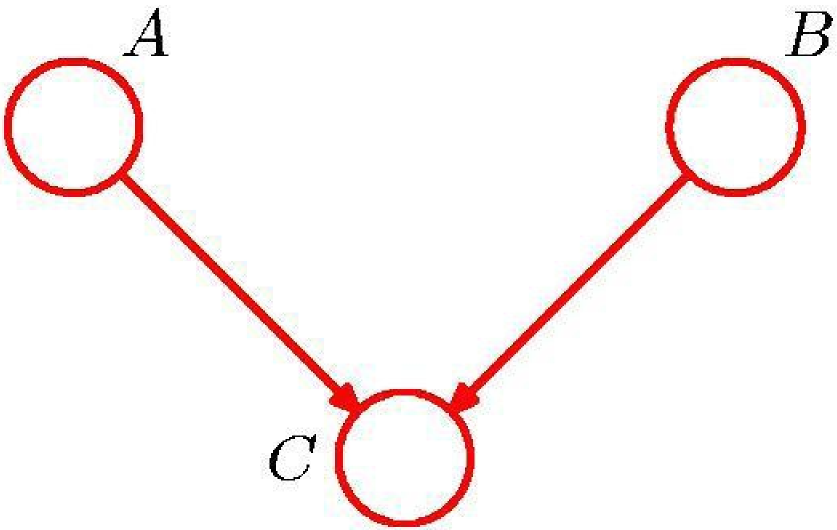
Moralizing (child → married)



$$p(x_1, x_2, x_3, x_4) = p(x_1)p(x_2)p(x_3)p(x_4 \mid x_1, x_2, x_3)$$

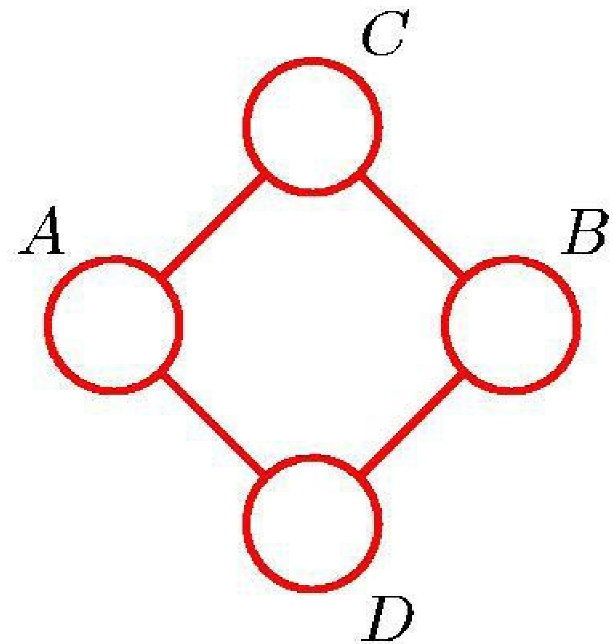
Figure from Bishop- Pattern Recognition and ML

Directed vs. Undirected Graphs



$$A \perp\!\!\!\perp B \mid \emptyset$$

$$A \not\perp\!\!\!\perp B \mid C$$



$$A \not\perp\!\!\!\perp B \mid \emptyset$$

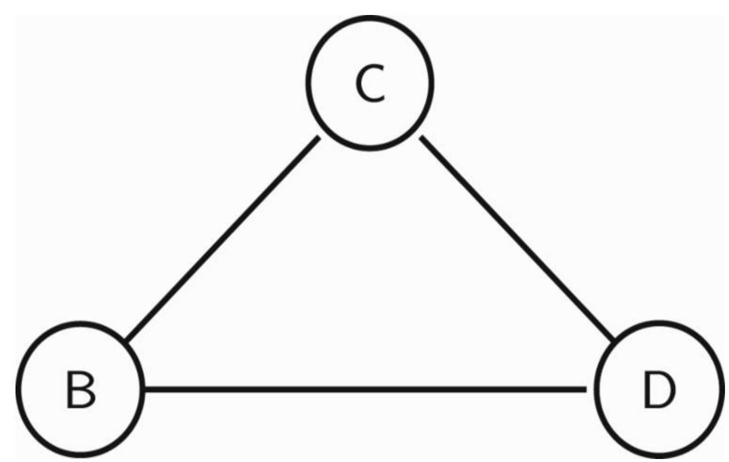
$$A \perp\!\!\!\perp B \mid C \cup D$$

$$C \perp\!\!\!\perp D \mid A \cup B$$

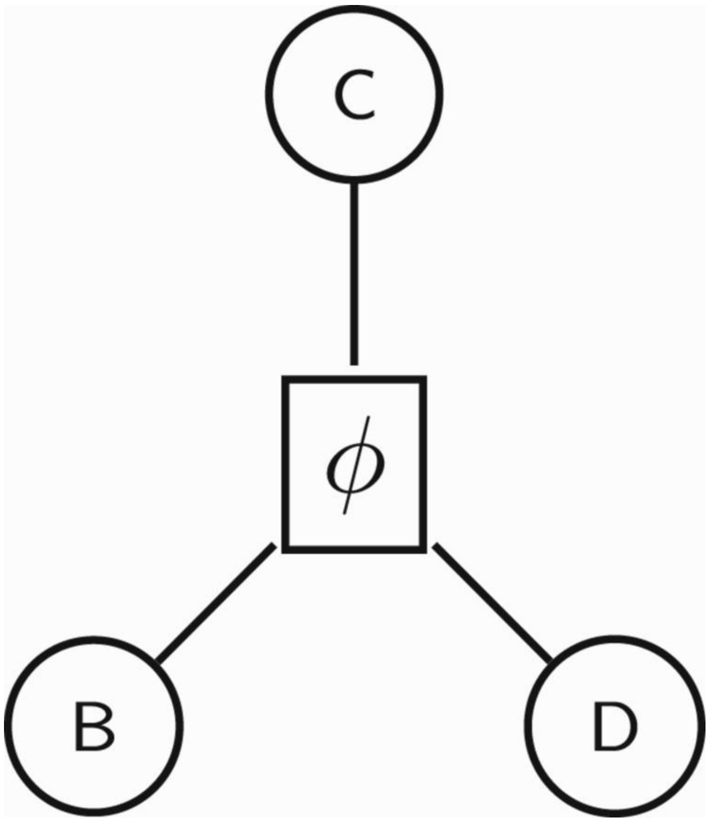
Figure from Bishop- Pattern Recognition and ML

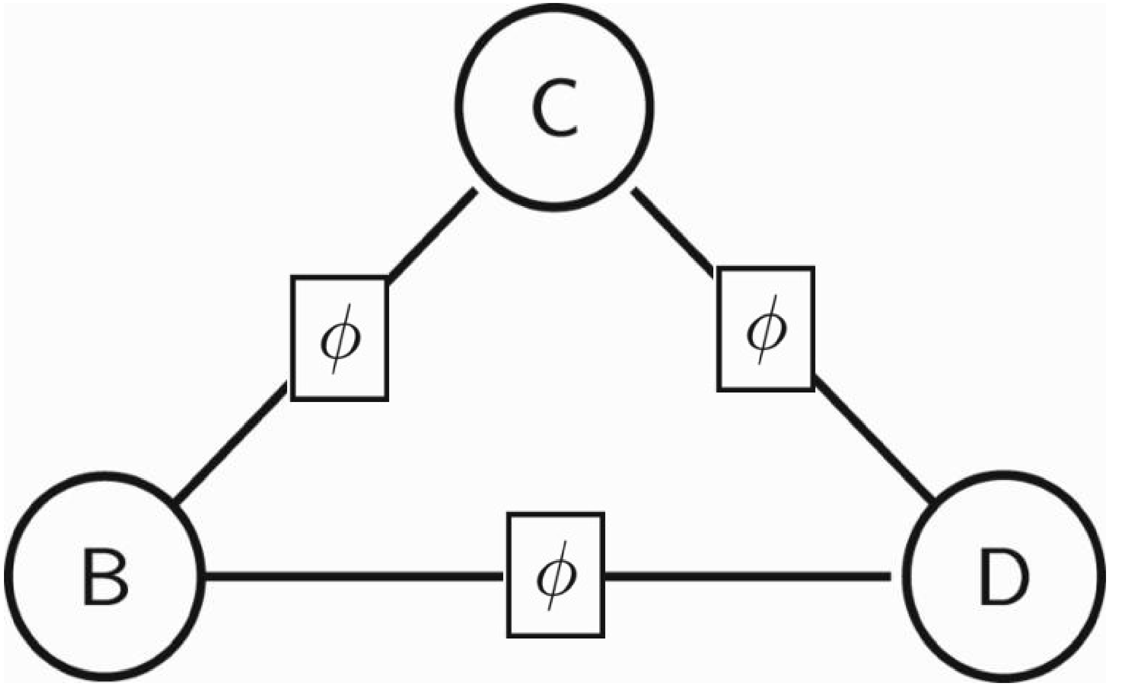
Why Factor Graphs

- Consider $p(\text{d, b, c}) = 1/Z\ \varphi(\text{d, b})\varphi(\text{b, c})\varphi(\text{c, d})$
- What is the corresponding Markov network (graphical representation)?
- A MRF with three nodes pairwise connected?
- But that also represents
- $p(\text{d, b, c}) = 1/Z\ \varphi(\text{d, b, c})$



Factor Graph can be specific





Factor Graphs

- Given a function

$$f(x_1, \ldots, x_n) = \prod_i \psi_i(\chi_i);$$

- the factor graph has a factor node for each factor, $\psi_i(\chi_i)$.
- and a variable node for each variable, x_j.
- When used to represent a distribution

$$p(x_1, \ldots, x_n) = (1/Z) \prod_i \psi_i(\chi_i),$$

- a normalization constant, Z, is assumed.

Bi-partite Graph

- A bi-partite graph has every edge connecting nodes from each of two disjoint sets.
- A factor graph is a bi-partite graph where the sets of nodes are variable nodes and factor nodes.

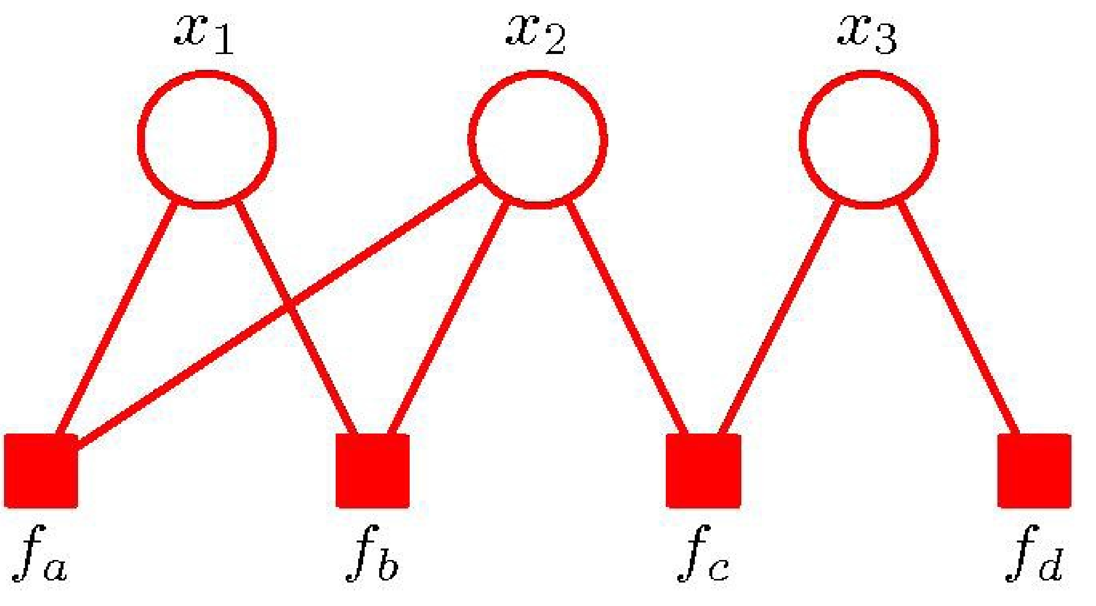
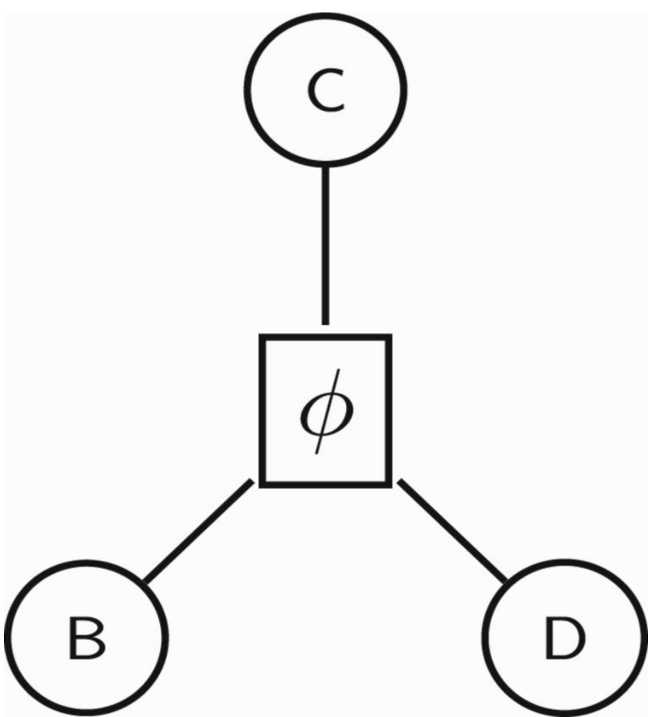
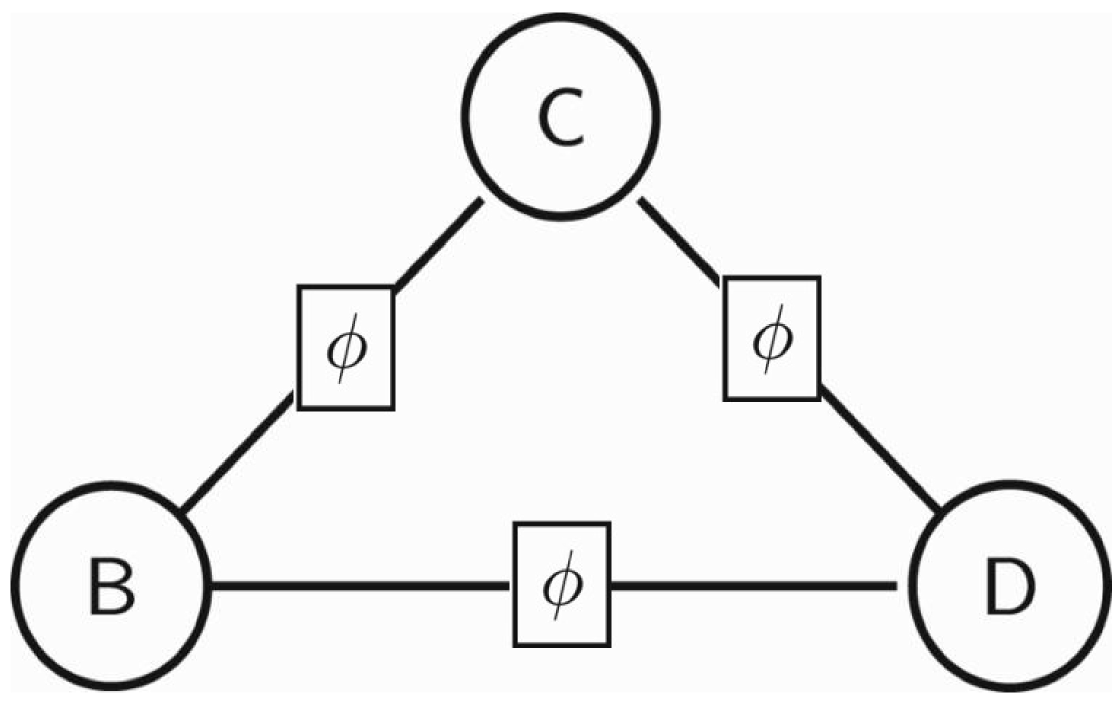


Figure from the lectures of: Bjoern Andres and Bernt Schiele

Factor Graph vs MRF

- The set of independences ($\mathcal{I}$ maps) that can be represented by both types is the same.
- Factor graphs are able to represent additional factorization beyond the I-map, ie factorizations are not generating more independences.





Log-Linear Models

- Factor graphs are more explicit, but still require bulky tables for all the factor values
- Can use features to capture patterns that we'd like reflected in clique potentials:
- $P(X_1,\ldots,X_N) = \phi_1(D_1)\ \phi_2(D_2)\ \ldots\ \phi_K(D_K)$
- Define $\phi_i(D_i) = \exp(-w_i\, f_i(D_i))$

  $f_i(D_i)$ tell us something indicative about some random variables, (yes $w_i\, f_i$ is basically the 'energy')
- So $P(X_1,\ldots,X_N) = \exp(-w_1\, f_1(D_1))\ldots \exp(-w_k\, f_k(D_k))$
- $-\log(P(X_1,\ldots,X_N)) = =\Sigma w_i\, f_i(D_i)$

Gaussian Network Models

$P_{\Phi}(X_1,\ldots,X_N)$ = exp(-$w_1 f_1(D_1)$)… exp(-$w_k f_k(D_k)$)

- For Gaussians these would have all features as quadratic in the 'X_i'. These might be 'sensor measurements'.
- The multivariant Gaussian distribution:

$$p(\mathbf{x}) = [(2\pi)^n |\Sigma|]^{-1/2} \exp[(-1/2)(\mathbf{x}-\boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x}-\boldsymbol{\mu})]$$

- 'Information Matrix', $\Omega = \Sigma^{-1}$.
- The Covariance matrix, Σ, must be non-negative .

Gaussian Network Models

$$G(\mathbf{x},\mu,\Sigma)=[(2\pi)^n|\Sigma|]^{-1/2}\exp[(-1/2)(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)]$$

$\mathbf{x}$ and $\mathbf{y}$ are Gaussian vector variables and A, B and C are matricies.

$$p(\mathbf{x},\mathbf{y}) = G\left(\begin{pmatrix}\mathbf{x}\\ \mathbf{y}\end{pmatrix},\begin{pmatrix}\mu_x\\ \mu_y\end{pmatrix},\begin{pmatrix}A & C\\ C^T & B\end{pmatrix}\right)$$

$$p(\mathbf{y}) = \int_{-\infty}^{\infty} p(\mathbf{x},\mathbf{y})d\mathbf{x} = G(\mathbf{y},\mu_y,B)$$

$$p(\mathbf{x}|\mathbf{y}) = p(\mathbf{x},\mathbf{y})/p(\mathbf{y}) = G(\mathbf{x},\mu_x + CB^{-1}(\mathbf{y}-\mu_y), A - CB^{-1}C^T)$$

Notice that this general Gaussian as a graph is fully connected so the graph is not very helpful. But sometimes the Gaussian model can have a simple graph which can be helpful.

Gaussian Network Models

$N(\boldsymbol{\mu}, \Sigma)=[(2\pi)^n|\Sigma|]^{-1/2} \exp[(-1/2)(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})]$

- So called 'linear Gaussian' (BN) models lead to a multivariant Gaussian (MRF) model.
- Linear Gaussian BN has each node y with parents **x** with conditional pdf's as Gaussians:
  - $P(y\,|\mathbf{x}) \sim N(\beta_0 + \boldsymbol{\beta}^T\mathbf{x}, \sigma^2)$
- Product of two Gaussians is a Gaussian.
  - Its own 'Conjugate prior' (a later lecture)

Exponential Family

$$A(\chi) \exp \{ < t(\theta), \tau(\chi) > \} / Z(\theta)$$

- Features are replaced by 'sufficient statistic' functions, $\tau(\chi)$, from the random variable space to a 'feature space'
- The w_i generalize to 'natural parameter' functions, $t(\theta)$, from a parameter space to the feature space
- Some sort of inner product between them.
- Add an 'axillary measure', $A(\chi)$, that multiplies each exponential term.

Projections - Entropy

- Entropy:

$$H_P(\chi) = - E_P[\ \ln P(\chi)\ ]$$

- Relative Entropy:

$$D(P \parallel Q) = E_P[\ \ln (P(\chi) / Q(\chi))$$

$$= -H_p(\chi) - E_P[\ \ln Q(\chi)\ ] >= 0$$

- Called the Kullback-Leibler divergence (distance) but not symmetric.

Projections - I and M

- I -Projection of P to Q

  $Q^I = \arg\min_Q D(Q \parallel P)$

  - Focus more on peaks in P.

- M -Projection of P to Q

  $Q^M = \arg\min_Q D(P \parallel Q)$

  - Focus more on spread of P.

  Notice Q is also being restricted to a given family

Projections – Q is Exponential

- $Q_\theta\,(\chi) = A(\chi)\,\exp\{ < t(\theta), \tau(\chi) > \} / Z(\theta)$
- If we find parameters, θ, such that:

  $E_{Q\theta}[\,\tau(\chi)\,] = E_P[\,\tau(\chi)\,]$

  then Q_θ is the M – Projection of P.

- This is what leads to the moment matching method and the name M projection.

SLAM Factor Graph

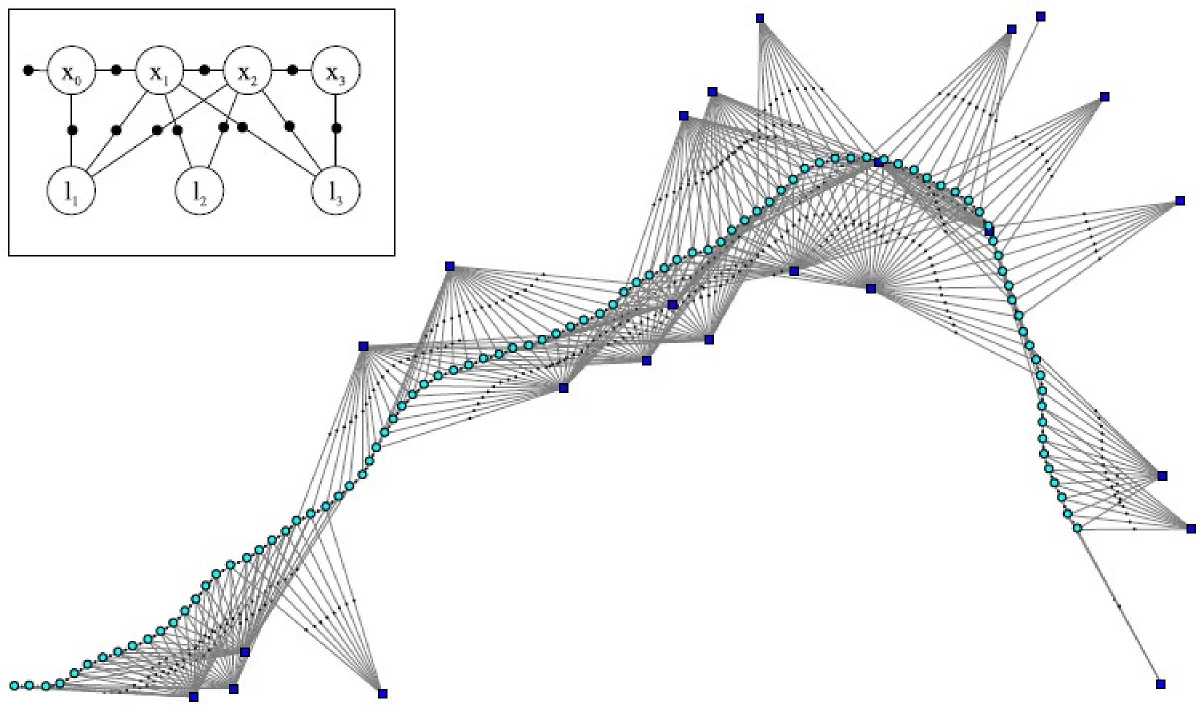


Figure from Dellart and Kaess ‘Square Root SAM: Simultaneous localization and mapping via square root information smoothing’.

Conditional Random Field CRF

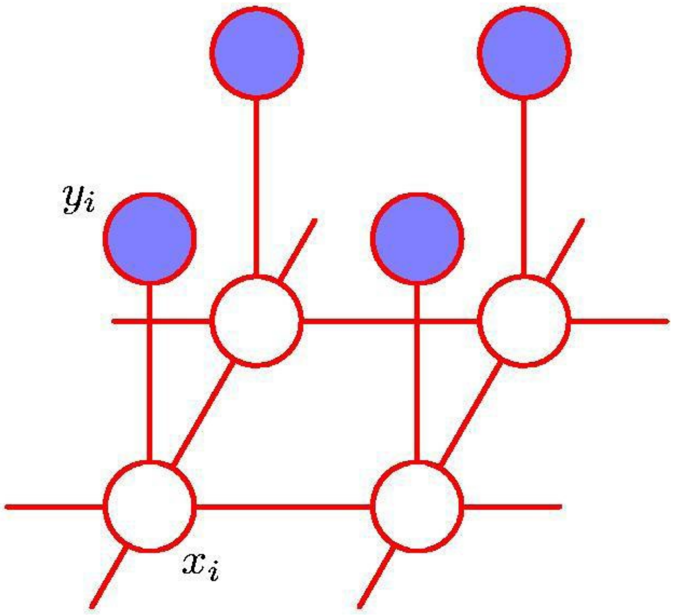
- A CRF works like a MRF only we interpret the factorization as being a conditional probability instead of a joint
- P(**Y** | **X**) instead of P(**Y**,**X**)
- Remember P(**Y** | **X**) = P(**Y**,**X**) / P(**X**)
- And if **X** is observed we sort of get the same answer in many cases however we interpret the factorization.

Conditional Random Field CRF

- $P(\mathbf{Y} \mid \mathbf{X})$ instead of $P(\mathbf{X,Y})$
- **Y** is called the target variables
- **X** is called the observed variables
- $P(\mathbf{Y} \mid \mathbf{X}) = (1/Z(\mathbf{X}))\, \tilde{P}(\mathbf{X,Y})$
- $\tilde{P}(\mathbf{X,Y}) = \Pi_{i=1..m}\, \phi_i(D_i)$
- $Z(\mathbf{X}) = \Sigma_Y\, \tilde{P}(\mathbf{X,Y})$

- So our Denoising example using a MRF can be done by interpreting it as a CRF.

Tutorial 3: MRF-Graph Cuts

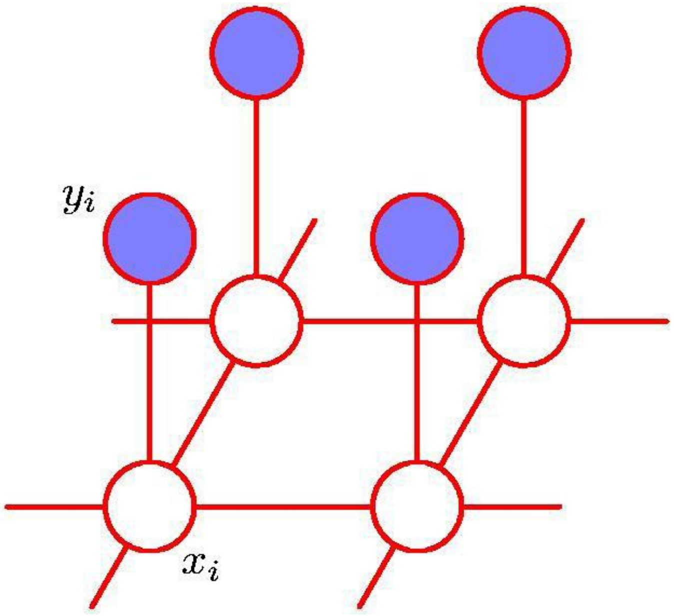


- Here the x_i are a hidden segment label.
- So foreground vs background.
- y_i are the observed image pixel value.
- So we want a to find the MAP,

  maximum a posteori, estimate **x** given **y.**
- Uses an exponential model for $\phi(x , y) \propto \exp(-E(\mathbf{x},\mathbf{y}))$
- The 'Gibbs Energy', E, Has terms for each type of edge above.
- The 'prior' is $U(x_i, y_i)$ and is given by (log of) a histogram over a user

  provided background/foreground regions.
- The smoothness term $V(x_i, x_j)$ is a constant for neighboring pixels with different labels.
- Or in a more refined model it is given a dependence on the y values (so add which edges to the graph above?)

Tutorial 3: MRF-Graph Cuts



- Uses an exponential model for $\phi(x\ ,\ y) \propto \exp(-E(\mathbf{x},\mathbf{y}))$
- MAP is same as minimize E with respect to labels x
- Cleverly this can be transformed to a Graph Cut or Max Flow problem that is easy.
- It is also solved using loopy message passing.