

Markov Networks

November 12, 2009

CS 486/686

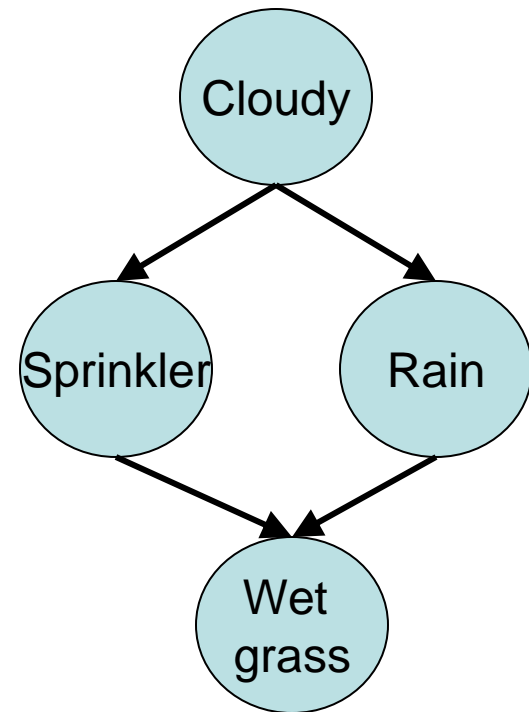
University of Waterloo

Outline

- Markov networks (a.k.a. Markov random fields)
- Reading: Michael Jordan, *Graphical Models*, *Statistical Science* (Special Issue on Bayesian Statistics), 19, 140-155, 2004.

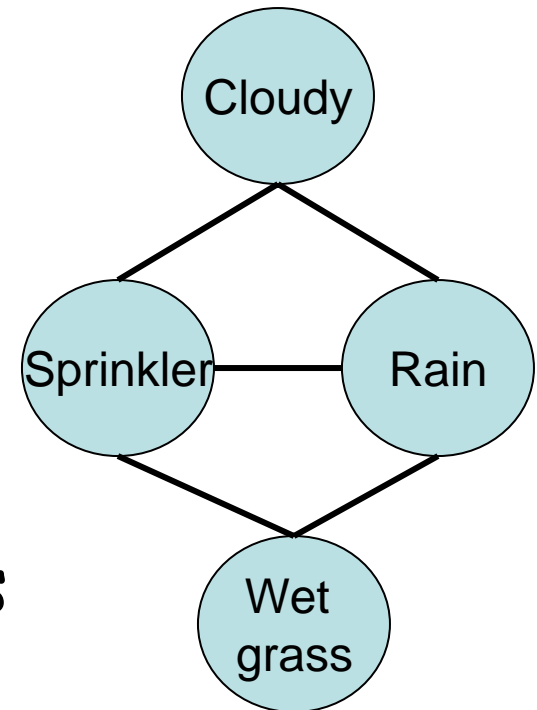
Recall Bayesian networks

- Directed acyclic graph
- Arcs often interpreted as causal relationships
- Joint distribution: product of conditional dist



Markov networks

- Undirected graph
- Arcs simply indicate direct correlations
- Joint distribution: normalized product of potentials
- Popular in computer vision and natural language processing



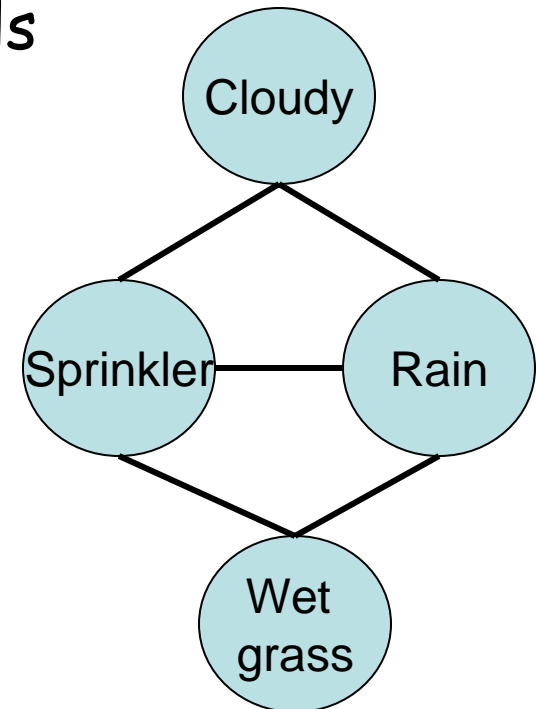
Parameterization

- Joint: normalized product of potentials
$$\Pr(\mathbf{X}) = \frac{1}{k} \prod_j f_j(\mathbf{CLIQUE}_j)$$
$$= \frac{1}{k} f_1(C, S, R) f_2(S, R, W)$$

where k is a normalization constant

$$k = \sum_{\mathbf{X}_i} \prod_j f_j(\mathbf{CLIQUE}_j)$$
$$= \sum_{C, S, R, W} f_1(C, S, R) f_2(S, R, W)$$

- Potential:
 - Non-negative factor
 - Potential for each maximal clique in the graph
 - Entries: "likelihood strength" of different configurations.



Potential Example

| $f_1(C,S,R)$ | |
|----------------------|-----|
| csr | 3 |
| $cs\sim r$ | 2.5 |
| $c\sim sr$ | 5 |
| $c\sim s\sim r$ | 5.5 |
| $\sim csr$ | 0 |
| $\sim cs\sim r$ | 2.5 |
| $\sim c\sim sr$ | 0 |
| $\sim c\sim s\sim r$ | 7 |

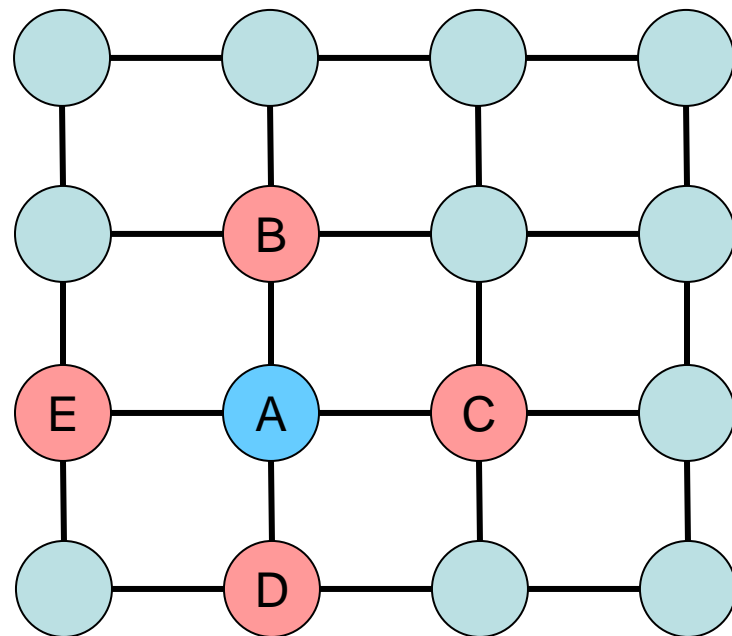
$c\sim sr$ is more likely than $cs\sim r$

impossible configuration

Markov property

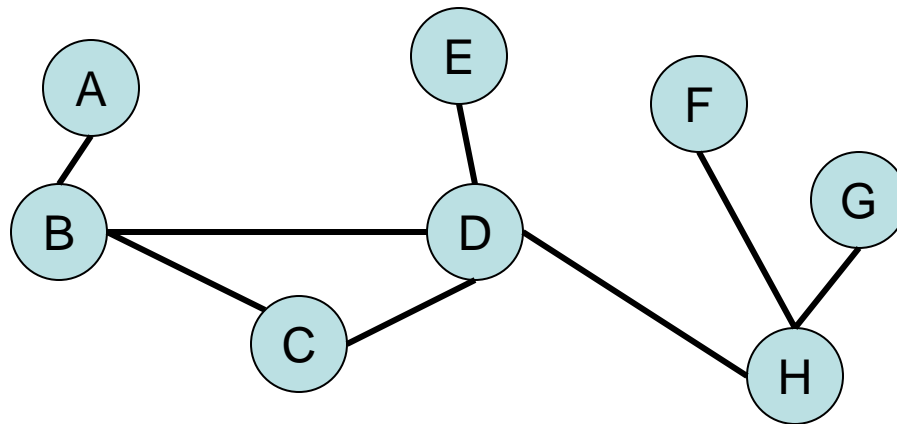
- **Markov property:** a variable is independent of all other variables given its immediate neighbours.
- **Markov blanket:** set of direct neighbours

$$MB(A) = \{B, C, D, E\}$$



Conditional Independence

- X and Y are independent given Z iff there doesn't exist any path between X and Y that doesn't contain any of the variables in Z
- Exercise:
 - A, E ?
 - $A, E | D$?
 - $A, E | C$?
 - $A, E | B, C$?



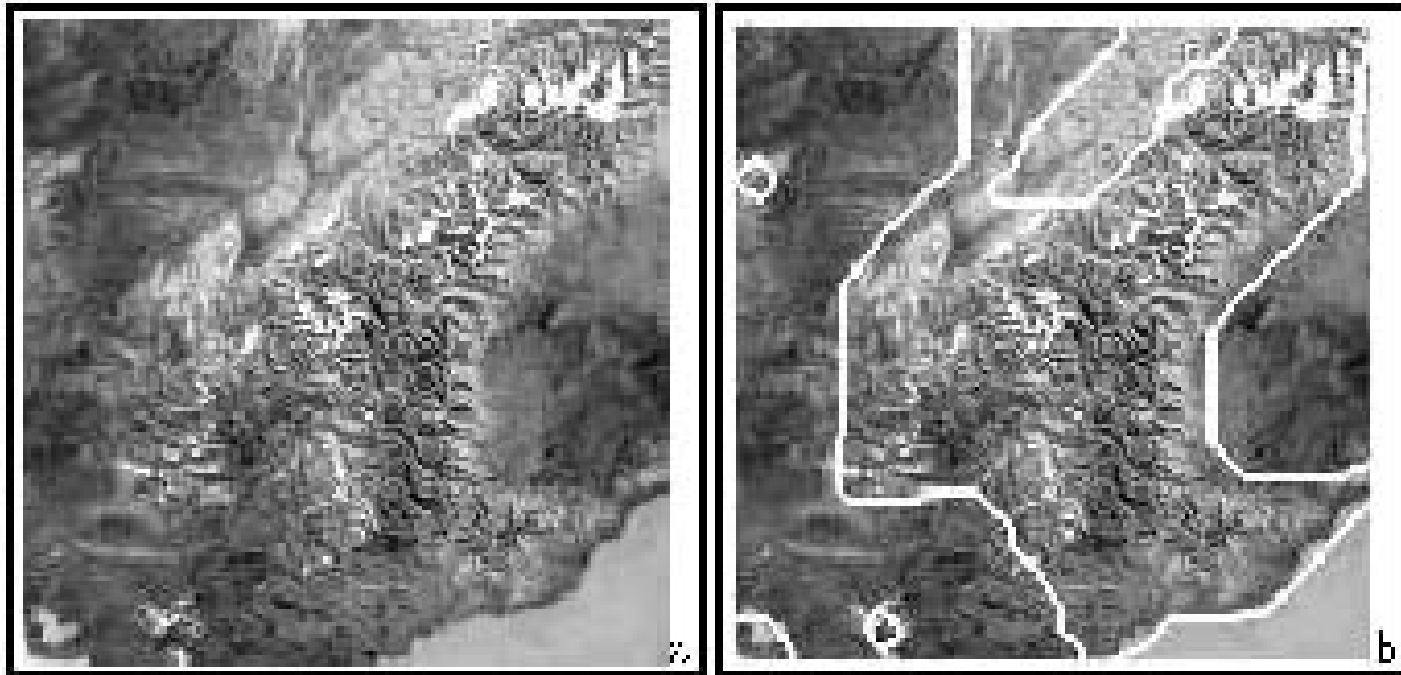
Interpretation

- Markov property has a price:
 - Numbers are not probabilities
- What are potentials?
 - They are indicative of local correlations
- What do the numbers mean?
 - They are indicative of the likelihood of each configuration
 - Numbers are usually learnt from data since it is hard to specify them by hand given their lack of a clear interpretation

Applications

- Natural language processing:
 - Part of speech tagging
- Computer vision
 - Image segmentation
- Any other application where there is no clear causal relationship

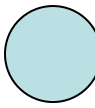

Image Segmentation

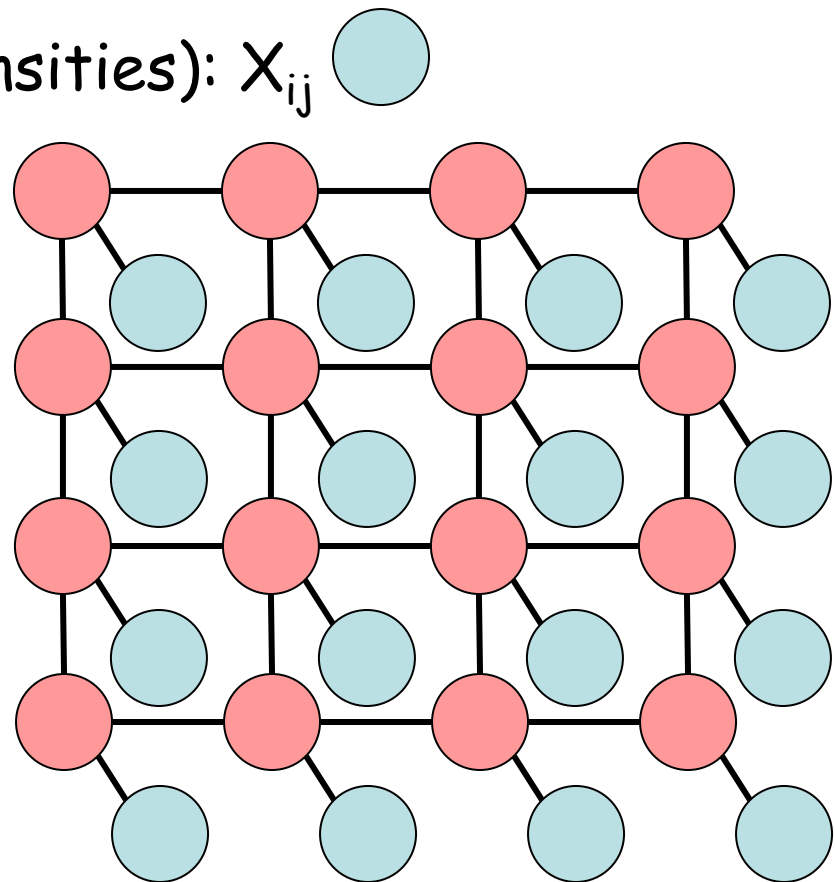


Segmentation of the Alps

Kervrann, Heitz (1995) A Markov Random Field model-based Approach to Unsupervised Texture Segmentation Using Local and Global Spatial Statistics, IEEE Transactions on Image Processing, vol 4, no 6, p 856-862

Image Segmentation

- Variables
 - Pixel features (e.g. intensities): X_{ij} 
 - Pixel labels: Y_{ij} 
- Correlations:
 - Neighbouring pixel labels are correlated
 - Label and features of a pixel are correlated
- Segmentation:
 - $\operatorname{argmax}_y \Pr(\mathbf{Y}|\mathbf{X})?$



Inference

- Markov nets: factored representation
 - Use variable elimination
- $P(X|E=e)$?
 - Restrict all factors that contain E to e
 - Sumout all variables that are not X or in E
 - Normalize the answer

Parameter Learning

- Maximum likelihood
 - $\theta^* = \operatorname{argmax}_{\theta} P(\text{data}|\theta)$
- Complete data
 - Convex optimization, but no closed form solution
 - Iterative techniques such as gradient descent
- Incomplete data
 - Non-convex optimization
 - EM algorithm

Maximum likelihood

- Let θ be the set of parameters and \mathbf{x}_i be the i^{th} instance in the dataset
- Optimization problem:
 - $\theta^* = \operatorname{argmax}_{\theta} P(\text{data}|\theta)$
 $= \operatorname{argmax}_{\theta} \prod_i \Pr(\mathbf{x}_i|\theta)$
 $= \operatorname{argmax}_{\theta} \prod_i \frac{\prod_j f(\mathbf{X}[j]=\mathbf{x}_i[j])}{\sum_{\mathbf{x}} \prod_j f(\mathbf{X}[j]=\mathbf{x}[j])}$

where $\mathbf{X}[j]$ is the clique of variables that potential j depends on and $\mathbf{x}[j]$ is a variable assignment for that clique

Maximum likelihood

- Let $\theta_x = f(\mathbf{X}=\mathbf{x})$
- Optimization continued:
 - $\theta^* = \operatorname{argmax}_{\theta} \Pi_i \frac{\Pi_j \theta_{\mathbf{x}_i[j]}}{\sum_{\mathbf{x}} \Pi_j \theta_{\mathbf{x}_i[j]}}$
 $= \operatorname{argmax}_{\theta} \log \Pi_i \frac{\Pi_j \theta_{\mathbf{x}_i[j]}}{\sum_{\mathbf{x}} \Pi_j \theta_{\mathbf{x}_i[j]}}$
 $= \operatorname{argmax}_{\theta} \sum_i \sum_j \log \theta_{\mathbf{x}_i[j]} - \log \sum_{\mathbf{x}} \Pi_j \theta_{\mathbf{x}_i[j]}$
- This is a non-concave optimization problem

Maximum likelihood

- Substitute $\lambda = \log \theta$ and the problem becomes **concave**:
 - $\lambda^* = \operatorname{argmax}_{\lambda} \sum_i \sum_j \lambda_{x_i[j]} - \log \sum_{\mathbf{x}} e^{\sum_j \lambda_{x_i[j]}}$
- Possible algorithms:
 - Gradient ascent
 - Conjugate gradient

Feature-based Markov Networks

- Generalization of Markov networks
 - May not have a corresponding graph
 - Use features and weights instead of potentials
 - Use exponential representation
- $\Pr(\mathbf{X}=\mathbf{x}) = 1/k e^{\sum_j \lambda_j \phi_j(\mathbf{x}[j])}$
where $\mathbf{x}[j]$ is a variable assignment for a subset of variables specific to ϕ_j
- Feature ϕ_j : Boolean function that maps partial variable assignments to 0 or 1
- Weight λ_j : real number

Feature-based Markov Networks

- Potential-based Markov networks can always be converted to feature-based Markov networks

$$\begin{aligned}\Pr(\mathbf{x}) &= 1/k \prod_j f_j(\mathbf{CLIQUE}_j = \mathbf{x}[j]) \\ &= 1/k e^{\sum_{j, \text{clique}_j} \lambda_{j, \text{clique}_j} \phi_{j, \text{clique}_j}(\mathbf{x}[j])}\end{aligned}$$

- $\lambda_{j, \text{clique}_j} = \log f_j(\mathbf{CLIQUE}_j = \mathbf{x}[j])$
- $\phi_{j, \text{clique}_j}(\mathbf{x}[j]) = 1$ if $\text{clique}_j = \mathbf{x}[j]$, 0 otherwise

Example

| $f_1(C,S,R)$ | |
|--------------|-----|
| csr | 3 |
| cs~r | 2.5 |
| c~sr | 5 |
| c~s~r | 5.5 |
| ~csr | 0 |
| ~cs~r | 2.5 |
| ~c~sr | 0 |
| ~c~s~r | 7 |

| weights | features | |
|--------------------------------|-------------------------|-------------------|
| $\lambda_{1,csr} = \log 3$ | $\phi_{1,csr}(CSR) =$ | 1 if CSR = csr |
| | | 0 otherwise |
| $\lambda_{1,*s~r} = \log 2.5$ | $\phi_{1,*s~r}(CSR) =$ | 1 if CSR = *s~r |
| | | 0 otherwise |
| $\lambda_{1,c~sr} = \log 5$ | $\phi_{c~sr}(CSR) =$ | 1 if CSR = c~sr |
| | | 0 otherwise |
| $\lambda_{1,c~s~r} = \log 5.5$ | $\phi_{1,c~s~r}(CSR) =$ | 1 if CSR = c~s~r |
| | | 0 otherwise |
| $\lambda_{1,~c^*r} = \log 0$ | $\phi_{1,~c^*r}(CSR) =$ | 1 if CSR = ~c^*r |
| | | 0 otherwise |
| $\lambda_{1,~c~s~r} = \log 7$ | $\phi_{~c~s~r}(CSR) =$ | 1 if CSR = ~c~s~r |
| | | 0 otherwise |

Features

- Features
 - Any Boolean function
 - Provide tremendous flexibility
- Example: text categorization
 - Simplest features: presence/absence of a word in a document
 - More complex features
 - Presence/absence of specific expressions
 - Presence/absence of two words within a certain window
 - Presence/absence of any combination of words
 - Presence/absence of a figure of style
 - Presence/absence of any linguistic feature

Next Class

- Conditional random fields