Graphical Models

Vikas Kedia

August 18, 2005

Outline

- What are Graphical Models?
- Inferencing Algorithms
- Conditional Random Fields
- Application Example

Graphical Models

Introduction

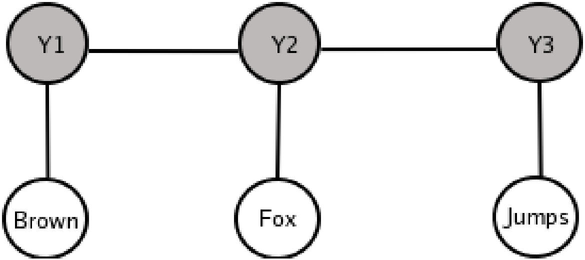
- Traditional Classification: each instance is labeled individually
- In many tasks this model is inadequate
  - POS tagging: tags of neighboring words important clues
  - Web Page Classification: classes of linked pages useful
- Collective Classification: classes/labels of all the instances inferred collectively
- Graphical Models a formalism for collective classification

Graphical Models

- Relations represented as a graph

  Vertices Labels/Observations eg: features of web page (observed), class of page (hidden)

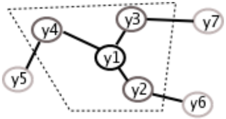
  Edges Dependencies eg: edge between linked web pages



Markov Property

- Probability distribution defined over values of all nodes in graph
- Local Markov Property : Given the values of its neighbours, value of the node is conditionally independent of values of other nodes

$$p(Y_v|Y_w, w \neq v) = p(Y_v|Y_w, w \sim v)$$

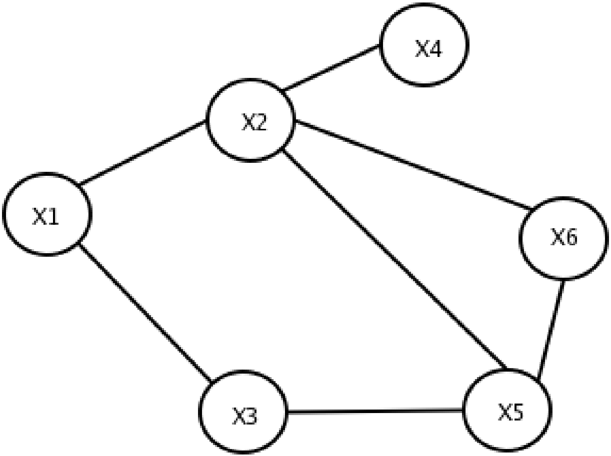


- Global Markov Property

$$p(Y_V) = \frac{\prod_C \phi_C(Y_{V_C})}{\sum_{Y_V} \prod_C \phi_C(Y_{V_C})}$$

ϕ_C Potential functions over labels of nodes in clique C

Example Graph



$$p(X_V) = \frac{\phi(x_1, x_2)\phi(x_1, x_3)\phi(x_2, x_4)\phi(x_3, x_5)\phi(x_2, x_5, x_6)}{Z}$$

Inferencing Algorithm

Two Inferencing Tasks

- Finding the most likely value of the variables

$$\max_{x_1,x_2,x_3,x_4,x_5,x_6} p(X_V)$$

- Finding the marginal probabilities

$$p(x_1) = \sum_{x_2,x_3,x_4,x_5,x_6} p(X_V)$$

Naive Approach

- Enumerate all possible combinations of values to all the variables
- Exponential number of possibilities, r^6 where r is the cardinality of each variable
- Clearly intractable for large graphs
- Insight: multiplication distributes over both max and sum operator

Example

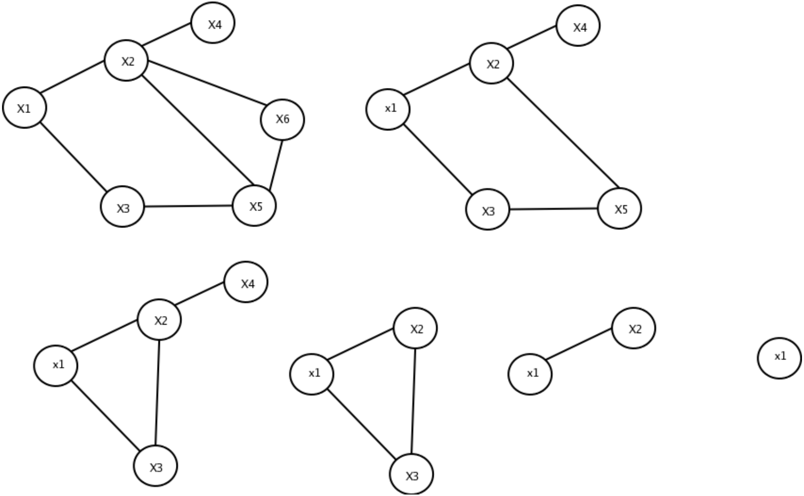
$$
\begin{aligned}
p(x_1) &= \frac{\sum_{x_2,x_3,x_4,x_5,x_6} \phi(x_1,x_2)\phi(x_1,x_3)\phi(x_2,x_4)\phi(x_3,x_5)\phi(x_2,x_5,x_6)}{Z} \\
&= \frac{\sum_{x_2} \phi(x_1,x_2) \sum_{x_3} \phi(x_1,x_3) \sum_{x_4} \phi(x_2,x_4) \sum_{x_5} \phi(x_3,x_5) \sum_{x_6} \phi(x_2,x_5,x_6)}{Z} \\
&= \frac{\sum_{x_2} \phi(x_1,x_2) \sum_{x_3} \phi(x_1,x_3) \sum_{x_4} \phi(x_2,x_4) \sum_{x_5} \phi(x_3,x_5) m_6(x_2,x_5)}{Z} \\
&= \frac{\sum_{x_2} \phi(x_1,x_2) \sum_{x_3} \phi(x_1,x_3) m_5(x_2,x_3) \sum_{x_4} \phi(x_2,x_4)}{Z} \\
&= \frac{\sum_{x_2} \phi(x_1,x_2) m_4(x_2) \sum_{x_3} \phi(x_1,x_3) m_5(x_2,x_3)}{Z} \\
&= \frac{\sum_{x_2} \phi(x_1,x_2) m_4(x_2) m_3(x_1,x_2)}{Z} \\
&= \frac{m_2(x_1)}{Z}
\end{aligned}
$$

Some Observations

- No more than 3 variables occur together in any summand
- Complexity is therefore r^3
- The order in which variables were chosen is elimination order
- Complexity would depend on the elimination order

Graph Theory Problem

- Variable that is summed over can be removed from the graph
- Intermediate function created is function of all the variables connected to the variable being summed over
- Therefore create a clique of all those variables
- Repeat till all the nodes are removed
- Largest clique created corresponds to the complexity


X4
X2
X1
X6
X3
X5
X4
X2
x1
X3
X5
X4
X2
x1
X3
X2
x1
X3
X2
x1
x1

Treewidth

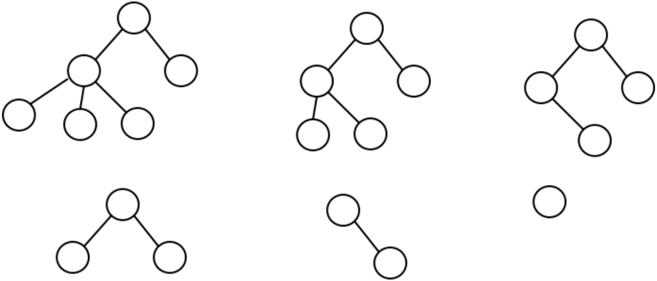
- Different elimination order give rise to different max clique size
- Treewidth is the minimum over all such max clique size
- To minimise complexity chose elimination order which gives rise to treewidth
- Unfortunately this problem is NP-Hard

Elimination Order in Specific Cases

- For specific types of graphs optimal elimination order is easy to see
- Example: for chains just keep on removing vertices from one end
- Gives rise to the Viterbi Algorithm
- Columns of the table correspond to the intermediate functions

Elimination Order for Trees

- Eliminate all children of a node before a node is eliminated



Conditional Random Fields

Limitation of HMM

- In Hidden Markov Models we assume that generation of a token depends only on the current state
- This restriction might be too limiting, we might want to include arbitrary features of data
- For example: we might want to look at some tokens on both sides of the current token
- Including such featuers in HMM increase the complexity of inferencing

Conditional Random Fields

- CRF introduced to overcome this limitation
- Nodes in the graph correspond only to labels
- Model globally conditioned on the observation data
- Potential functions therefore can be over entire data sequence

$$p(Y_V|X) = \frac{\prod_C \phi_C(Y_{V_C}, X)}{\sum_{Y_V} \prod_C \phi_C(Y_{V_C}, X)}$$

Linear CRF

- Potential functions are assumed to be exponential
- Parameters are tied across cliques of same type
- eg: For a chain CRF

$$p(y|x) = \frac{exp(\sum_{e \in E,k} \lambda_k f_k(y_e, x) + \sum_{v \in V,k} \mu_k g_k(y_v, x))}{Z(x)}$$

- λ_ks and μ_ks common for all edge and singleton cliques resp.
- Intuitively λ_ks are similar to state transition probabilities and μ_k to generation probabilities

Estimation and Inference

- λ_ks and μ_ks need to be learned from labelled training data
- Parameters are estimated using Maximum Likelihood hypothesis
- Log likelihood of data is maximised using numerical methods

$$L = \sum_j \left(\sum_{e \in E, k} \lambda_k f_k(y_e, x) + \sum_{v \in V, k} \mu_k g_k(y_v, x) - \log Z(x) \right)$$

- Gradient of log likelihood involves expected counts of feature values, calculated using dynamic programming

Graphical Models in Reconciliation

Reconciliation

- Reconciliation means finding duplicate records
- Traditionally based on syntactic similarity between pair of records
- Information flows from similarity of attributes to similarity of records
- But similarity between records also implies that the attributes are same
- eg similar citations means that the journal names in two also refer to same journal
- This bi-directional flow can be used for collective reconciliation

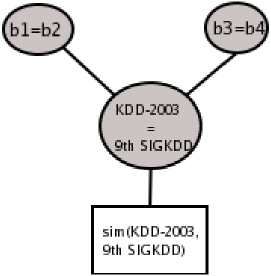
Example

Record	Title	Author	Venue
b1	Record Linkage using CRFs	Linda Stewart	KDD-2003
b2	Record Linkage using CRFs	Linda Stewart	9th SIGKDD
b3	Learning Boolean Formulas	Bill Johnson	KDD-2003
b4	Learning of Boolean Expressions	William Johnson	9th SIGKDD

Table: Duplicate Citations[3]

- b1=b2 means that KDD-2003 is same as 9th SIGKDD
- This will help in inferring similarity between b3 and b4

Collective Model



- Binary nodes for each pair of records
- Nodes for all possible pairs of values for each attribute called evidence nodes
- Value of evidence nodes is similarity measure and is observed
- Binary information nodes corresponding to each evidence node
- Information node represent whether the pair of attribute values are same

Cliques in the Model

- Singleton cliques for information and record nodes
- Edges connecting record nodes to the corresponding information nodes
- Edge connecting information nodes to the corresponding evidence nodes
- Inferencing done using graph partitioning

References I

M. I. Jordan.
Graphical models.
In *Statistical Science (Special Issue on Bayesian Statistics)*, pages 140–155, 2004.

John Lafferty, Andrew McCallum, and Fernando Pereira.
Conditional random fields: Probabilistic models for segmenting and labeling sequence data.
In *Proc. 18th International Conf. on Machine Learning*, pages 282–289. Morgan Kaufmann, San Francisco, CA, 2001.

Parag and P. Domingos.
Multi-relational record linkage.
In *Proceedings of the KDD-2004 Workshop on Multi-Relational Data Mining*, pages 31–48, 2004.