10-708: Probabilistic Graphical Models, Spring 2015

1 : Introduction to GM and Directed GMs: Bayesian Networks

Lecturer: Eric P. Xing *Scribes: Wenbo Liu, Venkata Krishna Pillutla*

1 Overview

This lecture introduces the notion of Probabilistic Graphical Models and a particular class of graphical models, called Bayesian Networks. Random variables shall be represented by uppercase letters, such as $X \sim \mathcal{N}(\mu, \sigma^2)$. A further assumption is that data are iid from some underlying distribution, unless stated otherwise.

2 Introduction

Probabilistic Graphical Models use a graph based representation to encode high-dimensional probability distributions succinctly. Graphical Models are used for reasoning under uncertainty. Example domains of application include Speech Recognition, Information Retrieval, Computer Vision and Robotic Control.

There are three aspects to a graphical model- the graph, a model of the data based on the graph and the data itself. Three fundamental issues arise: *representation, inference* and *learning*. The first is to find a representation that captures both the domain knowledge and uncertainties, and succinctly representing and quantifying these. The second is to infer probabilities based on the model and the data, for instance, $P(X_i|\mathcal{D})$. The third is to "learn" the parameters of the model, or the model itself or even the topology of the graph from the data.

3 Multivariate Distributions and Graphical Models

Consider a multivariate distribution on 8 binary-valued variables $X_1, ..., X_8$. Such a distribution contains 2^8 configurations and the specification of the joint distribution would require $2^8 - 1$ numbers. In practical applications, several of these configurations are unnecessary. In this form, inference is also expensive as it would, in general, require summing over an exponential number of configurations of unobserved variables. Learning is also problematic in this case because huge amounts of data are needed to accurately learn probabilities, especially of rare events. Even parametric forms would require a large number of parameters, for practical problems of modest size.

We can write:

$$P(X_1, ..., X_8) = P(X_1)P(X_2|X_1)P(X_3|X_2, X_1)...P(X_8|X_7, ..., X_1)$$

Even this representation requires an exponentially large conditional probability tables (CPTs). A solution to this problem is to use conditional independencies. Conditional independencies are statements of the form "X is independent of Y given Z" or $X \perp Y|Z$. If all the variables are independent, we can write:

$$P(X_1, ..., X_8) = P(X_1)P(X_2)P(X_3)...P(X_8)$$

Figure 1: Extending MLE to have a distribution on the parameter

This representation reduces parameter requirement from exponential to linear in the number of variables. But this representation is not rich enough to capture all possibilities. Probabilistic Graphical Models try to capture middle ground by using conditional independencies known from the domain. For instance, in the world of cellular signal transduction, molecules on the surface of the cell may not influence the behaviour of molecules inside the nucleus directly. Graph representation captures these conditional independencies compactly. Further, graphs provide a common language to domain experts and machine learning scientists.

Graphs also facilitate data integration. For instance, in the example of the gene network, different parts of the network may be investigated upon by different labs. Combing all this information amounts of putting different parts of the graph together.

Yet another benefit of using Graphical Models gives the unified view of Maximum Likelihood Estimation and Bayesian Learning. In Maximum Likelihood Estimation, the parameter is assumed to be fixed and has a direct causal link on the data. In Bayesian philosophy, the parameter is assumed to have an underlying distribution, parameterized by a "hyper-parameter". This extension can be viewed in the language of graphical models as adding another causal link from the hyper-parameter to the parameter. See figure 1.

To define formally, a probabilistic graphical model refers to a family of distributions on a set of random variables that are compatible with all the probabilistic independence propositions encoded by a graph that connects these variables. Two broad classes exist, those that use directed graphs and those that use undirected graphs- Bayesian Networks and Markov Random Fields respectively.

4 Bayesian Networks

4.1 Two types of GMs

There are two major classes of graphical models, one is based on directed graphs(Bayesian Network) and the other is based on undirected graphs(Markov Random Field).

A Bayesian Network (BN) is a probabilistic graphical model that represents a probability distribution through a directed acyclic graph (DAG) that encodes conditional dependency and independency relationships among variables in the model. In figure 2, the left figure is Bayesian Network and right figure is Markov Random Field.

The joint probability of Bayesian Network can be written as

$$P(X_1, ..., X_8) = P(X_1)P(X_2)P(X_3|X_1)P(X_4|X_2)P(X_5|X_2)P(X_6|X_3, X_4)P(X_7|X_6)P(X_8|X_5, X_6)$$

And the joint probability of Markov Random Field can be written as:

$$P(X_1, ..., X_8) = \frac{\mathbf{1}}{\mathbf{Z}} \exp(E(X1)+E(X2)+E(X3, X1)+E(X4, X2)+E(X5, X2)+E(X6, X3, X4)+E(X7, X6)+E(X8, X5, X6))$$

One can easily find the difference of in figure 3. The left figure is Bayesian Network. It gives causality relationships and facilitate a generative process. In this structure, a node is conditionally independent of

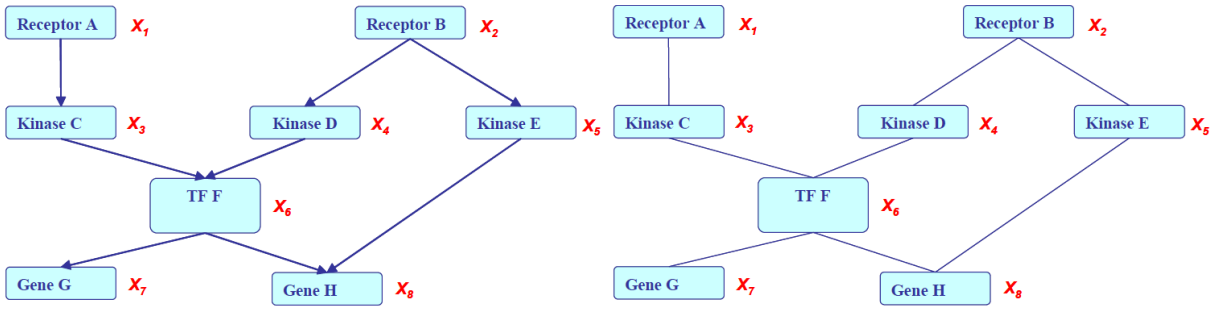

Figure 2: Examples of Directed GM and Undirected GM

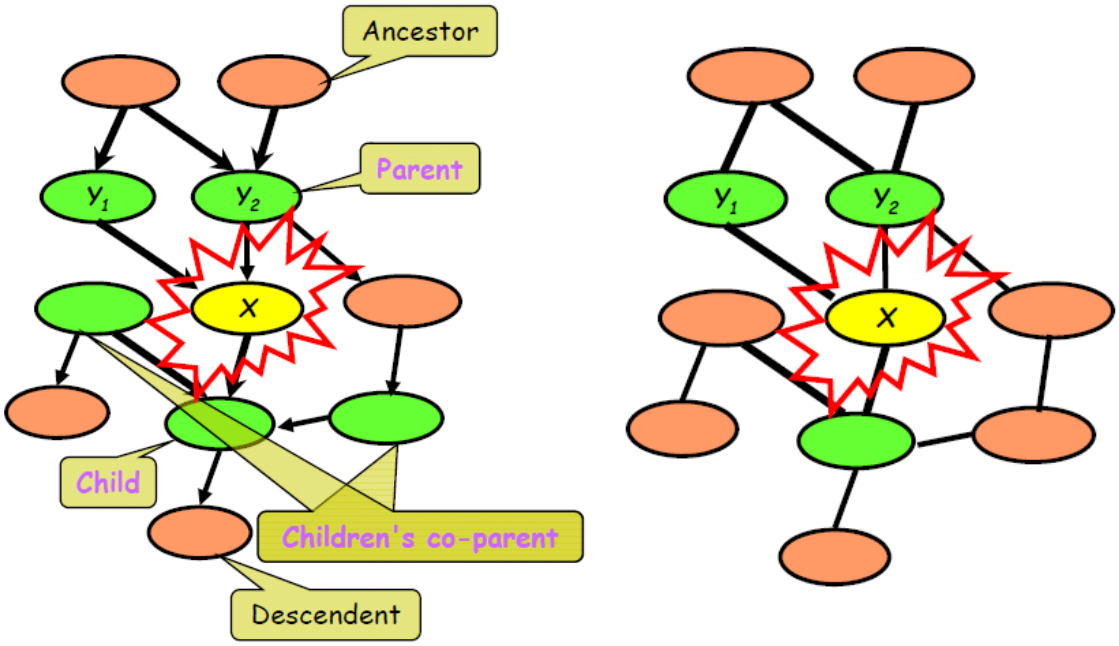

Figure 3: Structure of Directed GM and Undirected GM

every other node in the network outside its Markov blanket. It can be written as $P(X \mid \overline{X} = P(X \mid X_{MB}$. Here we say Markov blanket of node X means X's parents, its children, and its children's other parents.It is shown as green node on the figure. In inference algorithms, this method can be on and on called in iteratively algorithms.

The right figure is undirected graph.There is no concept of parents, children, and children's other parents concept in undirected graph. It simply gives correlations between variables. In this case, a node is conditionally independent of every other node in the network given its directed neighbors.

4.2 The Equivalence Theorem

One reason why we use graph to describe the conditional independency is we can use it to explain the decisions. For example we can say RV X is independent with Y given some cases, so we ignore X. A more important reason is that this is a way to establish the relationship between a graph and distribution. Given

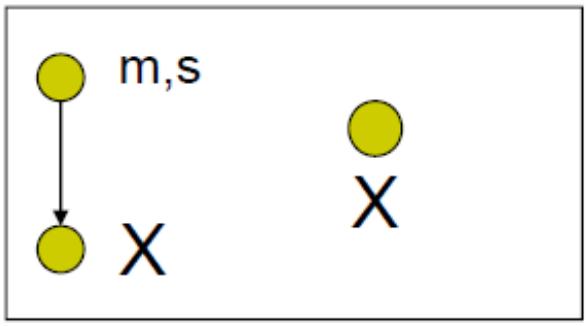

Figure 4: GM of Density Estimation

a graph, we can define a set of independencies. Given a set of distribution, we can do the same thing.

The Equivalence Theorem is describe as follows: For a graph G, Let $\mathcal{D}_1$ denote the family of all distributions that satisfy $I(\mathcal{G})$, Let $\mathcal{D}_2$ denote the family of all distributions that factor according to $\mathcal{G}$, Then $\mathcal{D}_1 \equiv \mathcal{D}_2$.

This is an important theorem to build the connection between graph and probabilistic model. Given a graph, one can first use the previous rules to cover all the conditional independencies encoded in this graph, and call this independency set $I(\mathcal{G})$. we can say there exist a set of distribution that satisfy $I(\mathcal{G})$.

And we have another family, which denotes the family of distributions factor according to $\mathcal{G}$, with the factor rules we used previously. It turns out that this two things are equivalent. Therefore we can now get the distribution by just using the factorizing rule, and get all conditional independencies in the graph.

4.3 GMs are your old friend

Graphical model is a familiar concept. We have used them implicitly earlier in certain contexts. For example, in density estimation, some observed data are given and we want to estimate density. This is basically a graphical model with one node. But if one wants to use a Bayesian framework, as mentioned earlier, one can elevate the parameter into a random variable, we have two node GM.

Similarly, Regression, classification and clustering can be describe as a two node GM. One node corresponds to data and the other corresponds to label. We can find that the difference between classification and clustering is the label. Classification has observed labels, whereas clustering has latent labels.

The point of using GM is to dealing with complete ones. Figure 4.3 is a graph of graphical models. Most of the models we have ever used are instances of graphical models. They have beautiful relationship cascading from simpler models to more complex models.

5 Fancier Graphical Models

GMs have a long history. In the past, physical scientists used graphical models before Computer Science even emerged as a field. For example, the famous Ising model and Potts model were originally proposed in physics. Physicists had already started applying these models to solid state physics and statistical mechanics. Later, computer scientists applied the concepts to a wide variety of machine learning and computer vision problems.

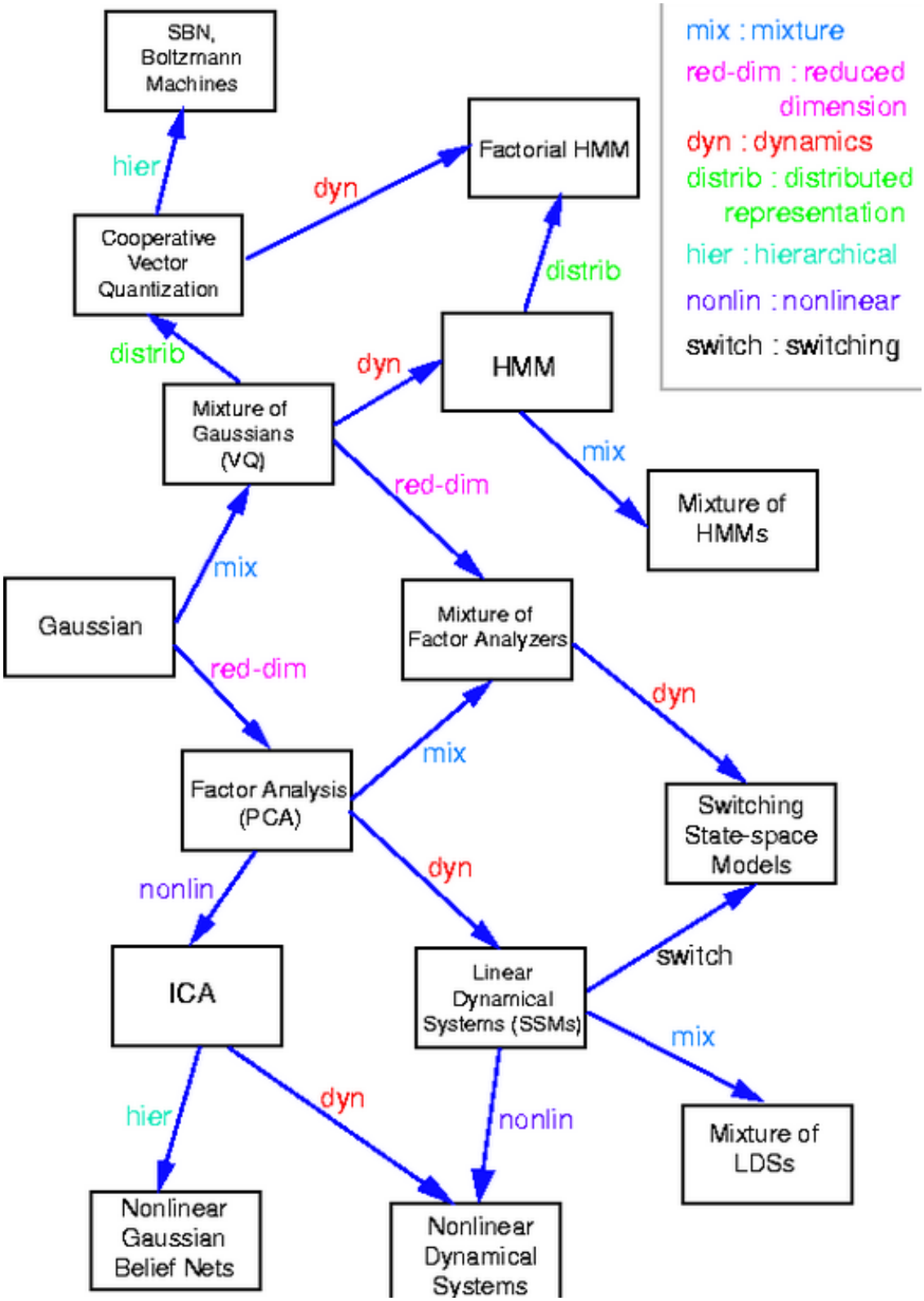

Figure 5: An (incomplete) genealogy of graphical models

6 Why graphical models?

- Communication: Domain expertise can be easily introduced into GMs. Thus, GM is a good communication tool between experts with different backgrounds. Imagine a machine learning people showing lots of tedious equations to a biologist. The biologist would probably not understand the model. However, a graphical model can easily convey the probabilistic dependencies of different variables, thus illustrating the model in a very intuitive way.
- Computation: Graphical model algorithms are often programmable and computable. Thus it is also a good tool for computation. a good tool for developing a wide variety of algorithms with very wide applications
- Development: GM is a good tool for developing a wide variety of algorithms with very wide applications.

7 Plan for the Class

Most of other graphical model courses only aim at introducing basic algorithms, which correspond to the first part in page 34 of the slides. However, the plan for this course is not only to introduce the basis, but also to show some more advanced and recent techniques and applications.These contents correspond to the second and third part of this page.

In general, we are going to move in a relatively fast pace, spending only a portion of the time on the basics and try to spend more time on additional knowledge.