10-707: Advanced Deep Learning **Spring 2020**

Lecture 10: February 19

Lecturer: Andrej Risteski *Scribes: Mary Bollinger, Shreyas Chaudhari*

Note: *LaTeX template courtesy of UC Berkeley EECS dept.*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications, if as reader, you find an issue you are encouraged to clarify it on Piazza. They may be distributed outside this class only with the permission of the Instructor.*

10.1 Review: Graphical Models

Recall from last lecture, a graph contains a set of nodes connected by edges. In a probabilistic graphical model, the nodes of the graph represent random variables, and the links represent the "probabilistic dependencies" between them.

There are two types of graphical models:

- Bayesian networks, also known as Directed Graphical Models. In these models, thee links have a particular directionality indicated by the arrows.
- Markov Random Fields, also known as Undirected Graphical Models. In these models the links do not carry arrows and have no directional significance.

10.1.1 Bayesian Networks

In a Bayesian Network, the joint distribution defined by the graph is given by the product of a conditional distribution for each node conditioned on its parents:

$$p(x) = \prod_{k=1}^{K} p(x_k|pa_k)$$

where pa_k denotes a set of parents for the node x_k
Each of the conditional distributions will typically have some parametric for (e.g. product of Bernoullis in the noisy-OR case)

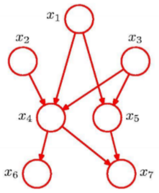


Figure 10.1: A Bayesian Network

Important restrictions: There must be no directed cycles! (i.e. in a Bayesian network the graph is a DAG)

Crucial Property: easy sampling
One of the nice properties of the Bayesian network is that it is easy to sample. consider a joint distribution over K random variables $p(x_1, x_2, ..., x_K)$ which factorizes as:

$$p(x) = \prod_{k=1} Kp(x_k|pa_k)$$

Suppose each of the conditional distributions are easy to sample from. How do we sample from the joint? Start at the top and sample in order, every time a sample is drawn, condition on the parents that were drew before. Example, consider Figure 10.1

$$\hat{x_1} \sim p(x_1)$$

$$\hat{x_2} \sim p(x_2)$$

$$\hat{x_3} \sim p(x_3)$$

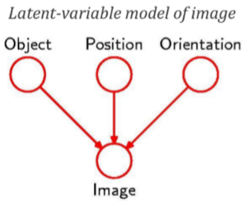
$$\hat{x_4} \sim p(x_4|\hat{x_1}, \hat{x_2}, \hat{x_3})$$

$$\hat{x_5} \sim p(x_5|\hat{x_1}, \hat{x_3})$$

Note: the parent variables are set to their sampled values. To obtain a sample from the marginal distribution, e.g. $p(x_2, x_5)$, sample form the full joint distribution, retain $\hat{x_2}, \hat{x_5}$, and discard the remaining values.

10.1.2 Typical deep learning application of Bayesian Networks

Sometimes it's convenient for higher-up nodes to represent latent (hidden) random variables. The role of latent variables is to allow modeling a complicated distribution over observed variables construction from simpler conditional distributions. One such example is in vision.



In this case, object identity, position, and orientation have independent prior probabilities. Image has a probablity distribution that depends on object identity, position and orientation (conditional distribution/likelihood)

$$p(Image, Object, Position, Orientation) = P(Image|Object, Position, Orientation)P(Object)P(Position)P(Orientation)$$

Where $P(Image|Object, Position, Orientation)$ is the likelihood, and $P(Object)P(Position)P(Orientation)$ is the prior.

The likelihood and prior are modeled by parametric distributions whose parameters are fitted throughout training.

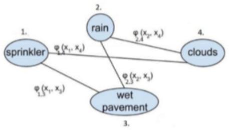

1.
sprinkler
2.
rain
4.
clouds
3.
wet pavement
$\varphi_{1,4}(x_1, x_4)$
$\varphi_{2,4}(x_2, x_4)$
$\varphi_{2,3}(x_2, x_3)$
$\varphi_{1,3}(x_1, x_3)$

Figure 10.2: A Markov Random Field

10.2 Undirected Graphical Models or Markov Random Fields (MRFs)

A pairwise undirected graphical model (MRF) expresses a distribution as a product of local potentials ϕ_{ij} (interactions), for example

$$p(x) = \frac{1}{Z} \prod_{i,j \in E(G)} \phi_{ij}(x_i, x_j)$$

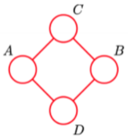
Where $Z =$ the partition function:

$$Z = \sum_{x} \prod_{i,j \in E(G)} \phi_{ij}(x_i, x_j)$$

The partition function is hard to calculate (In fact, there are simple cases where classical results in TCS show it is #P-hard to calculate), which makes sampling in MRFs hard, unlike Bayesian networks.
Typically the interactions are thought of as ”soft constraints” because they cause the probability distribution to like assignments so that the probability of these interactions is high.
More generally, we'd like to be able to define the distribution in terms of ”local” interactions. The correct way to formalize this is in terms of maximal cliques C

$$p(x) \propto \prod_{C} \phi_C(x_c)$$

For example, the joint distribution in the following figure factorizes as:



$$p(A, B, C, D) \propto \phi_{AC}(A, C)\phi_{BC}(B, C)\phi_{BD}(B, D)\phi_{AD}(A, D)$$

10.2.1 A brief review of cliques

The subsets that are used to define the potential functions are represented by maximal cliques in the undirected graph.

- Clique: a subset of nodes such that there exists and edge between all pairs of nodes in a subset.
- Maximal Clique: a clique such that it is not possible to include any other nodes in the set without it ceasing to be a clique.

The graph in Figure 10.3 has 5 cliques: $\{x_1, x_2\}, \{x_2, x_3\}, \{x_3, x_4\}, \{x_4, x_2\}, \{x_1, x_3\}$
And two maximal cliques: $\{x_1, x_2, x_3\}, \{x_2, x_3, x_4\}$

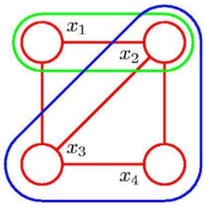
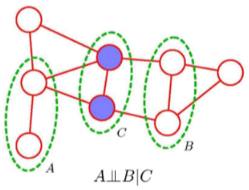


Figure 10.3: Looking at cliques

10.2.2 Why this odd parameterization?

Why Markov Random Fields?

- **Conditional independence (Hammersley-Clifford Theorem)**
  A property of Markov Random Fields is that if we have a distribution that factorizes over the cliques, nodes in A, B are independent, given a set of nodes C separating A, B

  $$p(x_A|x_C, x_B) = p(x_A|x_C) \; or \; p(x_A, x_B|x_c)p(x_A|x_C)p(x_b|x_c)$$

  A special case of this is that a node is independent of the rest of the graph, given values of the neighbors

  $$p(x_v|x_{N(v)}, x_{V/\{N(v),v\}}) = p(x_v|x_{N(v)})$$

  Surprisingly, the converse of the conditional Independence property holds. Consider the following sets of distributions:

  - The set of distributions consistent with the conditional independence relationships defined by the undirected graph.
  - The set of distributions consistent with the factorization defined by potential functions on maximal cliques of the graph

  The **Hammersley-Clifford theorem** states that these two sets of distributions are the same. This is the first reason we like MRFs and the reason why the clique factorization appears.

- **Maximum entropy (Jaynes; sufficient statistics) Jaynes principle**: The distribution p that maximizes the entropy H(p), subject to the constraints $\mathbb{E}_p[\phi_c(x_C)] = \mu_C$, for some pre-specified values of μ_C has the form

  $$p(x) \propto exp(\sum_C w_c \phi_c(x_c))$$

  for some weights w_c depending on μ_c
  This is nice because it makes the lease extra assumptions, subject to those constraints which is why this is the second reason we like MRFs.

- **Encode "energy" as a distribution**: We like MRFs because we can encode "energy" which can be viewed as a sampling equivalent of minimizing some loss. We can rewrite the for of the distribution ever so slightly to get
$$p(x) = \frac{1}{Z}\prod_C \phi_C(x_C) = \frac{1}{Z}exp(-\sum_C E(x_c))$$
Thus, p is a distribution putting more mass on configurations that minimize a certain energy. Configurations with high probabilities are those that find a good balance in satisfying the (possibly conflicting) influences of the clique potentials.

- The last reason we like Markov Random Fields is that they are interpretable and compact. Looking at a graph one can reason that variables related by an edge may be similar, while variables that are far away have little relation. However, this can be misleading. There are simple cases where variables far away in the graph are more strongly correlated than neighbors!

10.2.3 Some applications

There are many, many applications across many fields for Markov Random Fields. Some examples include image segmentation, separating foreground from background, and object detection. There are applications in biology such as finding the most likely folding of proteins. In physics, the Ising model is an example of an MRF. Community detection in networks is another application.

10.2.4 Simplest example: multivariate Gaussian

Recall the multivariate Gaussian:

$$\mathcal{N}(x|\mu,\Sigma) = \frac{1}{(2\pi)^{D/2}}\frac{1}{|\Sigma|^{1/2}}exp\{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\}$$

The term inside the exponential is quadratic: namely, we can write

$$P(x) = \frac{1}{Z}exp(-\frac{1}{2}x^TJx + g^Tx)$$

where $J = \Sigma^{-1}$, $\mu = J^{-1}g$

$$x^TJx = \sum_i J_{ii}x_i^2 + 2\sum_{ij\in E} J_{ij}x_ix_j$$

Thus, the interactions are given by the precision matrix Λ.
(Note: precision matrix being sparse does not imply the covariance matrix is sparse)

10.2.5 Discrete MRFs

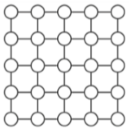


Figure 10.4: The Ising Model

MRFs with binary variables are sometimes called Ising models in statistical mechanics, and Boltzmann machines in machine learning literature.

Denoting the binary valued variable at node j by $x_j \in \{\pm 1\}$, the Ising model for the joint probabilities is given by:

$$P_\theta(x_i = 1 | x_{-i}) = \frac{1}{1 + exp(-\theta_i - \sum\limits_{ij \in E} x_j \theta_{ij})}$$

where x_i denotes all nodes except for i. If $\theta_{ij} \geq 0$: they prefer to be different.

10.2.6 Example: Ferromagnetic Ising Models

$$P_\theta(x) = \frac{1}{Z(\theta)} exp(\sum_{ij \in E} x_i x_j \theta_{ij} + \sum_{i \in V} x_i \theta_i)$$

If $\theta_{ij} \geq 0$: the model is called ferromagnetic, and is used in physics to model the atomic structure (spins) of iron.

10.2.7 Example: Image Denoising

Noise removal from a binary image: Let the observed noisy image be described by an array of binary pixel values: $y_j \in \{-1, +1\}$, i = 1,..,D. We take a noise-free image $x_j \in \{-1, +1\}$, randomly flip the sign of pixels with some small probability.

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

Where $h \sum\limits_i x_i$ is the bias term, $\beta \sum_{i,j} x_i x_j$ says that the neighboring pixels are likely to have the same sign, and $-\eta \sum\limits_i x_i y_i$ says that noisy and clean pixels are likely to have the same sign.

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

10.3 Modeling pros/cons of directed and undirected models

- MRFs
  - Con: hard to draw samples
  - Pro: Models independence structure directly
  - Captures soft constraints/energy
- Bayesian Networks
  - Pro: Easy to draw samples
  - Con: Models independence structure indirectly
  - Captures ”causal structure”

10.4 Latent Variable Models

Very often, the data at hand has some hidden (latent) variables and some observable variables. The relation between them can be modeled either by directed or undirected latent variables models, examples of both are covered in the sections that follow. A couple of broad use cases for these models are:

- Modeling part of the data that is non-observable (example: diseases when the symptoms are observable)
- Extracting features/representations for the data (example: position and orientation of the image for the observable raw pixels)

10.5 Directed Latent Variable Models

A single-layer Bayesian Network is a simple yet powerful paradigm, shown in Figure 10.5. It has one set of latent variables that have no parents, and the observables that depend on the latent variables. It follows all the properties of conditional independence that bayesian betworks do, and the joint parametric distribution is represented as:

$$p_\theta(X, Z) = p_\theta(Z)p_\theta(X|Z)$$

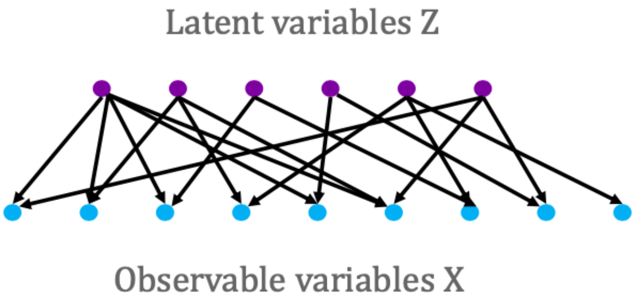


Figure 10.5: A single-layer Bayesian Network

10.5.1 Mixture Distributions

A mixture model used for clustering (Figure 10.6) is one of the examples a single layer BN with latent variables. The data points are the *observables*, and the clustering class/labels form the *latent* variables. Sampling a point (X, Z) in such models is a two-step process:

1. Sample Z from a categorical distribution on K components with parameters $\{\pi_i\}$
2. Sample X from the corresponding component in the mixture that was sampled above

The most common example of a mixture model is the Gaussian Mixture Model, which is specified by the equations below.

$$\forall k : \pi_k \geq 0 \qquad \textstyle\sum_{k=1}^K \pi_k = 1$$
$$p(\mathbf{x}) = \textstyle\sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

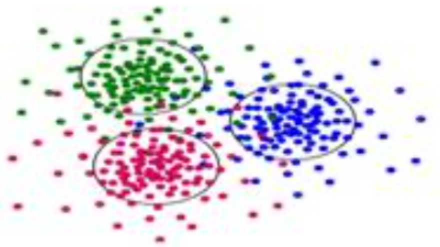


Figure 10.6: Mixture Model

10.5.2 Noisy-OR Networks

As the name suggestes, these models compute a ‘noisy-OR‘ of the latent variables (z). With a structure as shown in Figure 10.5, a noisy OR model has:

$$x_i, z_i \in \{0, 1\}$$
$$W_{ij} \geq 0$$

Since there are no connections between the latent variables, each z_i is on with probability ρ, independent of the other latent variables. Each active z_i activates x_j with probability $1 - \exp(-W_{ij})$. x_j can only be one if one of the z_i's activate it (the OR operation). It can be summarised as:

$$p(z_i = 1) = \rho, \ \ \forall i$$
$$p(x_j = 1 | z_i = 1) = 1 - \exp(-W_{ij}), \ \ \forall i, j$$

10.5.3 Topic Models (LDA)

Topic models are used to model the topical structure of documents. A particular instantiation of a topic model is the Latent Dirichlet Allocation model. The distribution that forms for basis of LDA is the Dirichlet distribution, which is discussed in the following section.

10.5.3.1 Dirichlet Distribution

The Dirichlet distribution is a over the probability simplex, i.e. over the points $\{\mu_i\}_{i=1}^K$ such that

$$\forall k : \mu_k \geq 0 \text{ and } \sum_{k=1}^{K} \mu_k = 1$$

The distribution is parameterized by $\{\alpha_i \geq 0\}_{i=1}^K$ as:

$$\text{Dir}(\mu|\alpha) \propto \prod_{k=1}^{K} \mu_k^{\alpha_k - 1}$$

The magnitude of α's determine the which portion of the probability simplex have a higher concentration.

With that background on the Dirichlet distribution, we can move on to analyse topic models.

Parameters:

- $\{\alpha_i\}_{i=1}^{K}$: Dirichlet parameters that determine a distribution over the K topics.
- $\beta \in R_{+}^{N \times K}$, where N is the size of the vocabulary. The columns of the matrix determine the distribution of words in a topc i, i.e., $\sum_{j=1}^{N} \beta_{ij} = 1$.

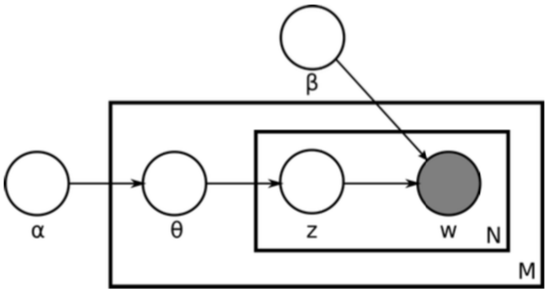


Figure 10.7: Bayesian Network modeling LDA

To produce a document, a sampling procedure similar to one described in Mixture Models is followed. Figure 10.7 shows the underlying graphical model.

1. Sample $\theta \sim \text{Dir}(.|\alpha)$: Topic proportion vector
2. Sample topic z_i from categorical distribution with parameters θ
3. Sample word with categorical distribution with parameters β_{z_i}

The main algorithmic difficulty arises in sampling from the **posterior** $P(Z|X)$. That is because computation of the normalising constant is complicated and thus the posterior can be #-P hard to sample from.

10.6 Undirected Latent Variable Model

The above mentioned difficulty in sampling from the posterior is tackled by some undirected latent variable models. However, that benefit comes at the cost of difficulty of sampling from the model. A few examples of such models are discussed in sections that follow.

10.6.1 Restricted Boltzmann Machines

The hidden and visible variables are binary: $\mathbf{v} \in \{0,1\}^D, \mathbf{h} \in \{0,1\}^F$. The bipartite structure as shown in Figure 10.8 facilitates easy sampling from the posterior for this model. The energy of the joint configuration

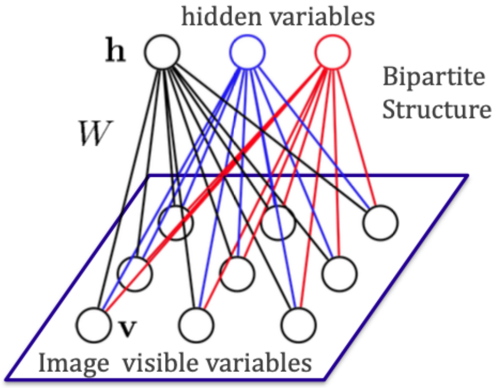


Figure 10.8: Restricted Boltzmann Machine

is given by (θ are model parameters):

$$E(\mathbf{v},\mathbf{h};\theta) = -\sum_{ij} W_{ij} v_i h_j - \sum_i b_i v_i - \sum_j a_j h_j$$

And the joint probability is given by the Boltzmann distribution:

$$P_\theta(\mathbf{v},\mathbf{h}) = \frac{1}{\mathcal{Z}(\theta)} \exp(-E(\mathbf{v},\mathbf{h};\theta)) = \frac{1}{\mathcal{Z}(\theta)} \prod_{ij} e^{W_{ij} v_i h_j} \prod_i e^{b_i v_i} \prod_j e^{a_j h_j}$$
$$\mathcal{Z}(\theta) = \sum_{\mathbf{h},\mathbf{v}} \exp(-E(\mathbf{v},\mathbf{h};\theta))$$

The term '*restricted*' comes from the structure in that there are not connections amongst hidden or visible variables. This makes sampling over the posterior easy.

$$P(\mathbf{h}|\mathbf{v}) = \prod_j \underbrace{P(h_j|\mathbf{v})}_{factorizes} \quad P(h_j = 1|\mathbf{v}) = \frac{1}{1 + \exp\left(-\sum_i W_{ij} v_i - a_j\right)}$$

And similarly, sampling visible variables given the hidden variables:

$$P(\mathbf{v}|\mathbf{h}) = \underbrace{\prod_i P(v_i|\mathbf{h})}_{factorizes} \quad P(v_i = 1|\mathbf{h}) = \frac{1}{1 + \exp\left(-\sum_j W_{ij} h_j - b_i\right)}$$

10.6.2 Gaussian Bernoulli RBMs

Removing the restriction on binary visible variables, we may sometimes want the observables to have a continuous distribution. The easiest way to model that would be using a Gaussian distribution. With a graphical model structure similar to Figure 10.8, the set of equations follow from the previous section.

$$P_\theta(\mathbf{v},\mathbf{h}) = \frac{1}{\mathcal{Z}(\theta)} \exp\left(\sum_{i=1}^{D}\sum_{j=1}^{F} W_{ij} h_j \frac{v_i}{\sigma_i} + \sum_{i=1}^{D} \frac{(v_i - b_i)^2}{2\sigma_i^2} + \sum_{j=1}^{F} a_j h_j\right)$$

By the conditional independence property of MRFs, the v_is are independent of each other conditioned on the hidden variables. Thus for each i, the distribution of v_i is:

$$P\left(v_i = x|\mathbf{h}\right) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{\left(x - b_i - \sum_j W_{ij}h_j\right)^2}{2\sigma_i^2}\right)$$

$$\implies P_\theta(\mathbf{v}|\mathbf{h}) = \prod_{i=1}^{D} P_\theta\left(v_i|\mathbf{h}\right) = \prod_{i=1}^{D} \mathcal{N}\left(b_i + \sum_{j=1}^{F} W_{ij}h_j, \sigma_i^2\right)$$

10.6.3 Applications

- **RBMs for Word Counts** [1], [2]
  It is an undirected analog of the topic models previously discussed. It has properties similar to the models discussed above (Figure 10.9):
  - 1-of-K visible variables
  - Binary hidden variables
  - Bipartite connections

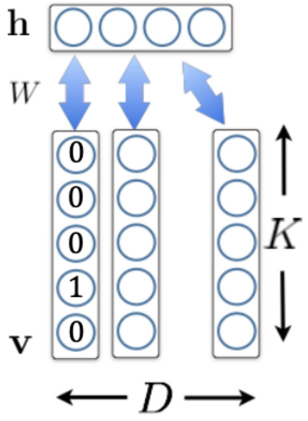


Figure 10.9: RBMs for word counts

  The probability of an observable conditioned on a hidden variable is defined by a softmax distribution, and this this model has gets its name - Replicated Softmax Model.

- **Collaborative Filtering**
  Another use case of RBMs is that they can be used to model tabular data, such as user's ratings of movies. [3] The modeling is similar to topic models with genres as the topics. These constitute the hidden variables, while the user ratings are the observables.

10.7 Conclusion

The algorithmic pros and cons of the latent variables models studied so far can be tabulated as follows:

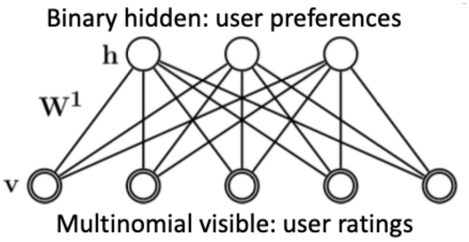

Binary hidden: user preferences
h
W¹
v
Multinomial visible: user ratings

Figure 10.10: RBMs for collaborative filtering

RBM's	Directed models
Hard to draw samples Easy to sample posterior distribution over latents	Easy to draw samples Hard to sample posterior distribution over latents

References

[1] R. Salakhutdinov and G. Hinton, "Replicated Softmax: an Undirected Topic Model," *Advances in Neural Information Processing Systems 25 (NIPS 2010)*

[2] N. Srivastava and R. Salakhutdinov, "Multimodal Learning with Deep Boltzmann Machines," *Advances in Neural Information Processing Systems 25 (NIPS 2012)*

[3] R. Salakhutdinov and A. Mnih and G. Hinton, "Restricted Boltzmann Machines for Collaborative Filtering," *Proceedings of the 24-th International Conference on Machine Learning*