10707
Deep Learning: Spring 2020

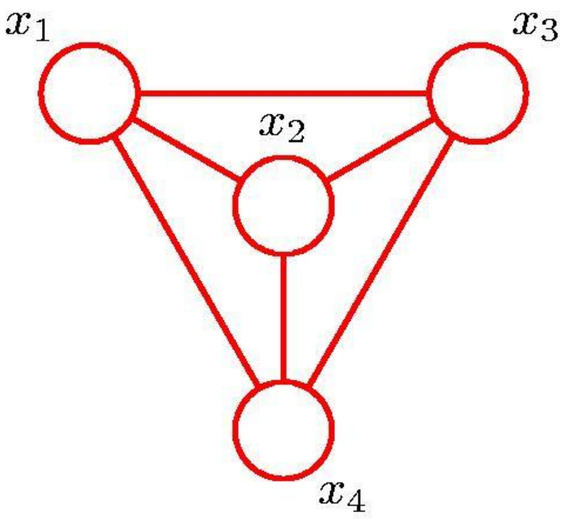
Andrej Risteski

Machine Learning Department

Lecture 10:
Common parametric distributions, Bayesian networks

Graphical Models

Recall: graph contains a set of nodes connected by edges.



In a probabilistic graphical model, each node represents a random variable, links represent "probabilistic dependencies" between random variables.

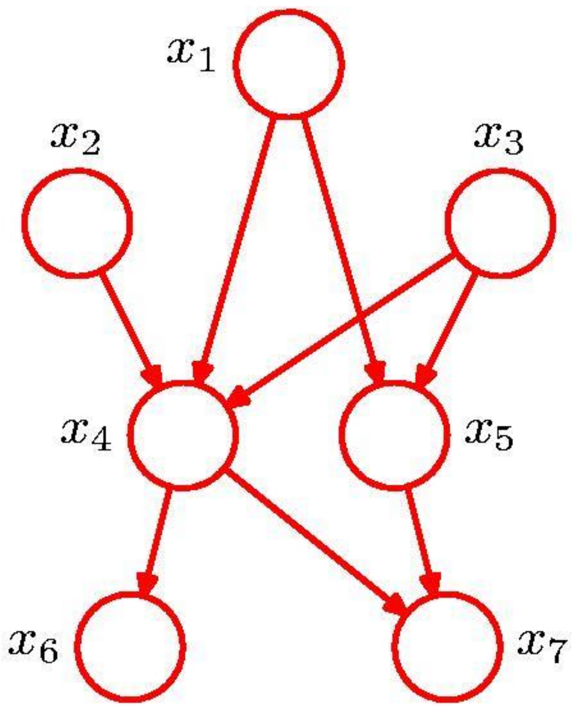
Graph specifies how joint distribution over all random variables decomposes into a **product** of factors, each factor depending on a subset of the variables.

Two types of graphical models:

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

Bayesian networks

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



$$p(\mathbf{x}) = \prod_{k=1}^{K} p(x_k|\mathrm{pa}_k)$$

where pa_k denotes a set of parents for the node x_k.

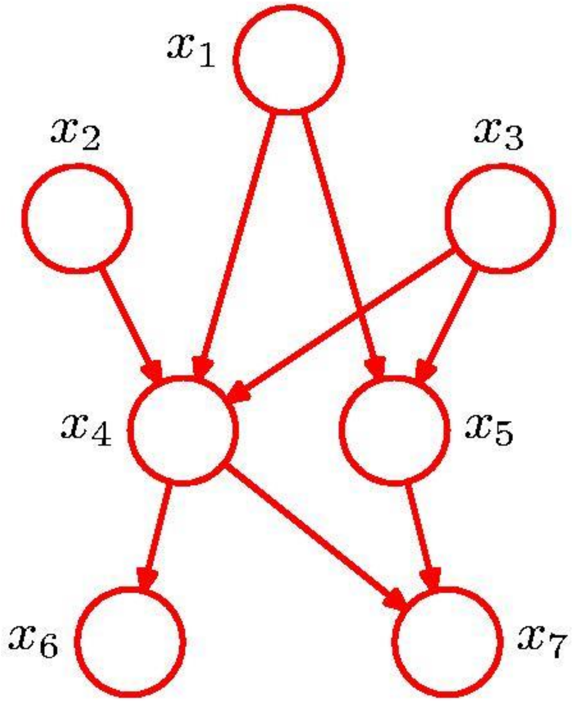
Each of the conditional distributions will typically have some parametric form. (e.g. product of Bernoullis in the noisy-OR case)

Important restriction: There must be **no directed cycles!** (i.e. graph is a DAG)

Crucial property: easy sampling

Consider a joint distribution over K random variables $p(x_1, x_2, \ldots, x_K)$ that factorizes as:

$$p(\mathbf{x}) = \prod_{k=1}^{K} p(x_k | \mathrm{pa}_k)$$



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.

$$\hat{x}_1 \sim p(x_1)$$
$$\hat{x}_2 \sim p(x_2)$$
$$\hat{x}_3 \sim p(x_3)$$
$$\hat{x}_4 \sim p(x_4 | \hat{x}_1, \hat{x}_2, \hat{x}_3)$$
$$\hat{x}_5 \sim p(x_5 | \hat{x}_1, \hat{x}_3)$$

The parent variables are set to their sampled values

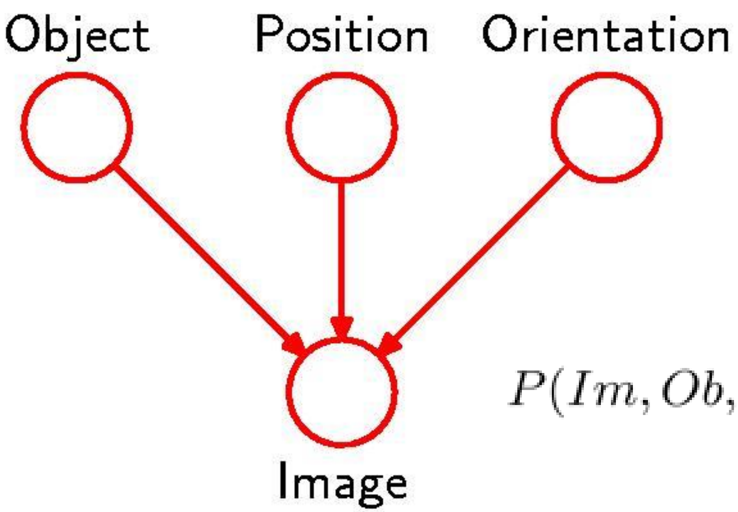
To obtain a sample from the marginal distribution, e.g. $p(x_2, x_5)$, sample from the full joint distribution, retain $\hat{x}_2, \hat{x}_5$, discard the remaining values.

Typical deep learning application

Higher-up nodes will typically represent **latent** (hidden) random variables.

The role of latent variables is to allow modeling a **complicated** distribution over observed variables **constructed** from **simpler** conditional distributions.

Latent-variable model of image



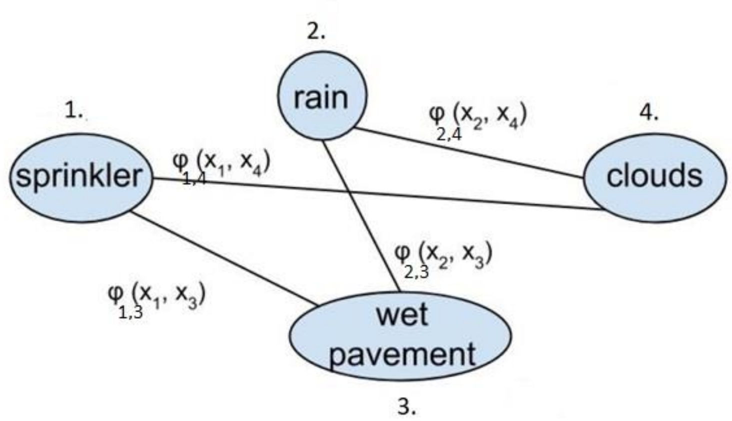
Object identity, position, and orientation have independent *prior probabilities*.

Image has probability distr that depends on object identity, position, and orientation (*conditional distribution/likelihood*).

$$P(Im, Ob, Po, Or) = \underbrace{P(Im|Ob, Po, Or)}_{\text{Likelihood}}\underbrace{P(Ob)P(Po)P(Or)}_{\text{Prior}}$$

Likelihood and prior are modeled by parametric distribution whose parameters are fitted throughout training.

Undirected graphical models or Markov Random Fields (MRFs)



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

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

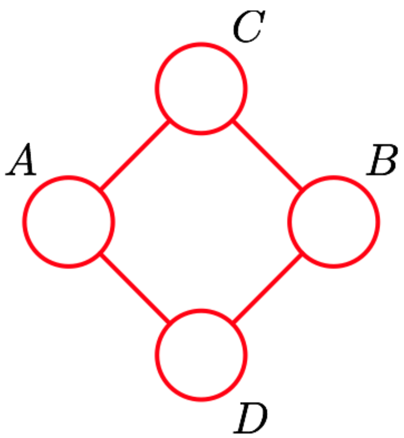
Typically the interactions are thought of as "*soft constraints*"

Unlike Bayesian networks, **sampling is hard**: *partition function*

$$Z = \sum_x \Pi_{(i,j)\in E(G)} \phi_{ij}(x_i, x_j)$$ is hard to calculate.

(In fact, there are simple cases where classical results in TCS show it is #P-hard to calculate.)

Undirected graphical models or Markov Random Fields (MRFs)



More generally, we'd like to be able to define distribution in terms of "local" interactions.

The correct way to formalize this is in terms of **maximal cliques** C (clique = fully connected subset of nodes).

$$p(x) \propto \Pi_C \phi_C(x_C)$$

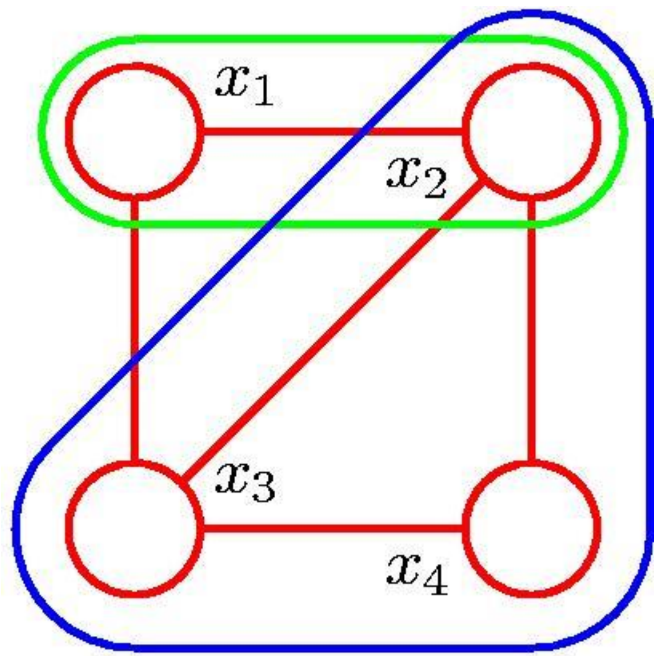
For example, the joint distribution above factorizes as:

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

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 an 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.

This graph has 5 cliques:

$$\{x_1, x_2\}, \{x_2, x_3\}, \{x_3, x_4\},$$
$$\{x_4, x_2\}, \{x_1, x_3\}.$$

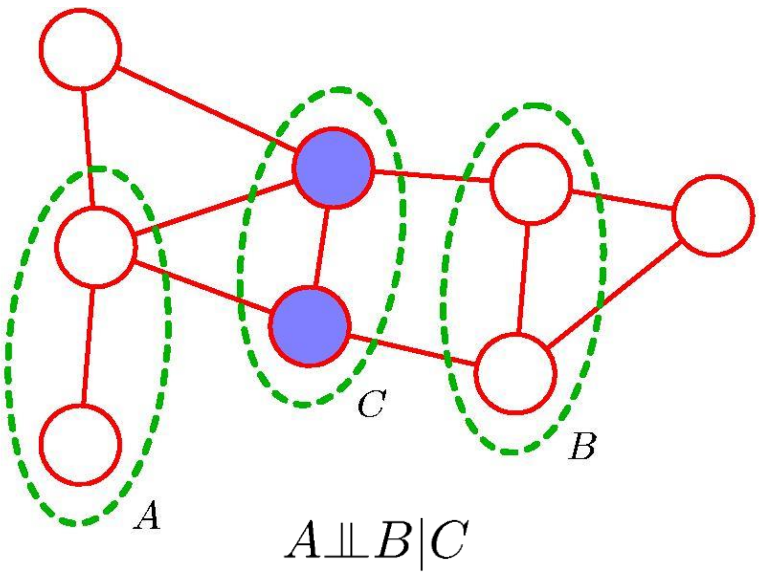
Two maximal cliques:

$$\{x_1, x_2, x_3\}, \{x_2, x_3, x_4\}.$$

Why this odd parametrization?

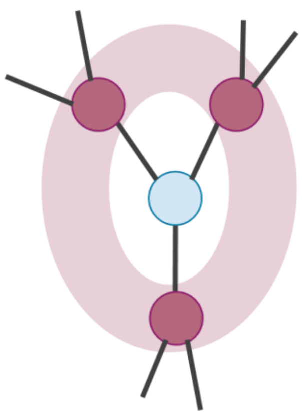
- Why Markov Random Fields
  - Conditional independence (Hammersley-Clifford Theorem)

Conditional Independence



Nodes in A, B are independent, given a set of nodes C separating A, B

$$p(x_A \mid x_C,\ x_B) = p(x_A|x_C)$$



Special case: node is independent of the rest of the graph, given values of the neighbors

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

Conditional Independence and Factorization

Surprisingly, converse holds too. 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.

Hammersley-Clifford theorem: these two sets of distributions are the same.

Why this odd parametrization?

- Why Markov Random Fields
  - Conditional independence (Hammersley-Clifford)
  - 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

$$p(x) \propto \exp\left(\sum_C w_c \phi_C(x_C)\right)$$

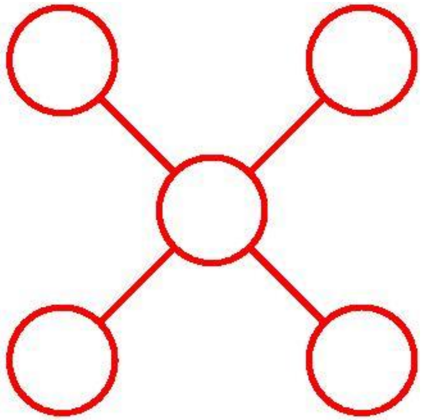
for some weights w_c depending on μ_C

Why this odd parametrization?

- Why Markov Random Fields
  - Conditional independence (Hammersley-Clifford)
  - Maximum entropy (Jaynes; sufficient statistics)
  - Encode “energy” as distribution

Energy interpretation

Rewriting the form of the distribution ever so slightly, we get:



$$p(\mathbf{x}) = \frac{1}{\mathcal{Z}} \prod_C \phi_C(x_C) = \frac{1}{\mathcal{Z}} \exp(-\sum_C E(x_c))$$

Thus, p is a distribution putting more mass on configurations that minimize a certain energy

(Like a *sampling version* of loss minimization)

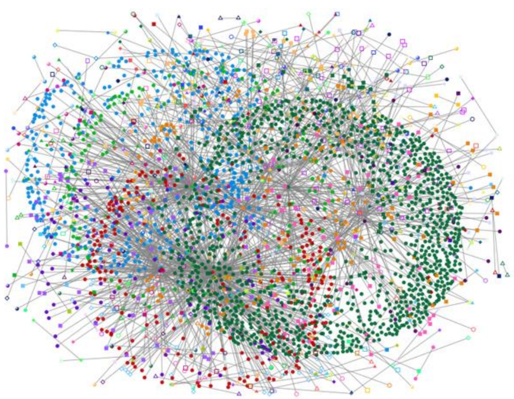
Configurations with high probabilities are those that find a *good balance* in satisfying the (possibly conflicting) influences of the clique potentials.

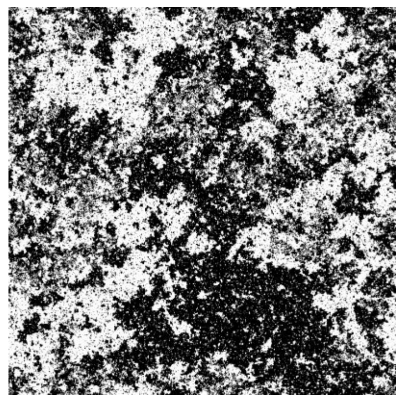
Some motivations for parametrization

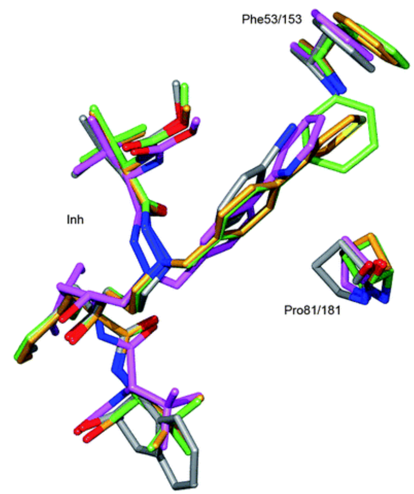
- Why Markov Random Fields
  - Conditional independence (Hammersley-Clifford)
  - Maximum entropy (Jaynes; sufficient statistics)
  - Encode "energy" as distribution
  - Interpretable, compact

Can be misleading!
There are simple cases where variables far away in the graph are more strongly correlated than neighbors!!

Some applications





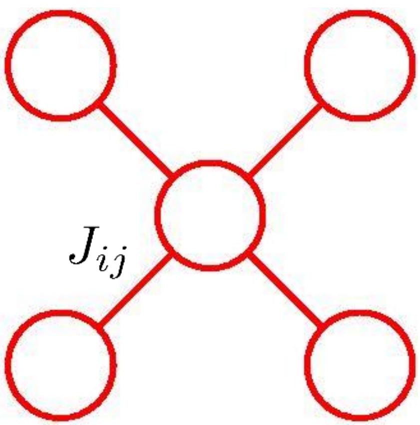


Simplest example: multivariate Gaussian

Recall the multivariate Gaussian

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

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



$$P(\mathbf{x}) = \frac{1}{\mathcal{Z}} \exp(-\frac{1}{2}\mathbf{x}^T J \mathbf{x} + \mathbf{g}^T \mathbf{x}),$$

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

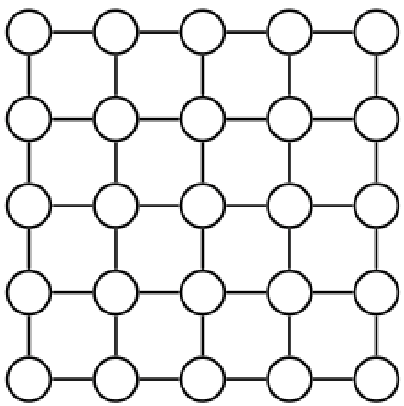
$$\mathbf{x}^T J \mathbf{x} = \sum_i J_{ii} x_i^2 + 2 \sum_{ij \in E} J_{ij} x_i x_j,$$

Thus, the interactions are given by the precision matrix Λ.

(Note: precision mx being sparse **does not** imply the covariance mx is sparse.)

Discrete MRFs

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:

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

The conditional distribution is given by logistic (only depends on nbrhood!):

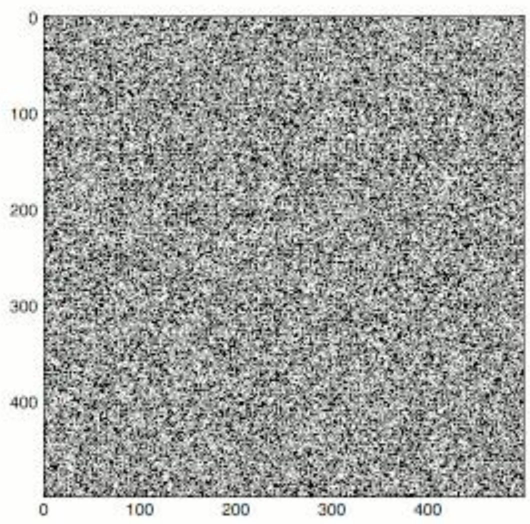
$$P_\theta(x_i = 1 | \mathbf{x}_{-i}) = \frac{1}{1 + \exp(-\theta_i - \sum_{ij \in E} x_j \theta_{ij})},$$

where x_{-i} denotes all nodes except for i.

If $\theta_{ij} \geq 0$: the nodes i,j, prefer to be the same. If $\theta_{ij} \leq 0$: they prefer to be different.

Example: Ferromagnetic Ising models

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

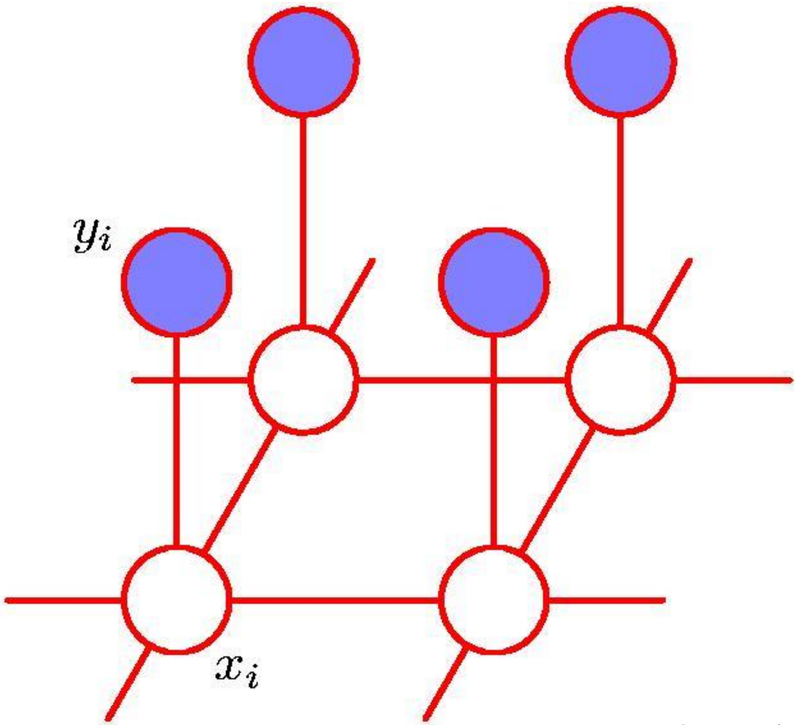


If $\theta_{ij} \geq 0$: the model is called ferromagnetic, and is used in physics to model the atomic structure (spins) of iron.

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.

Bias term

Neighboring pixels are likely to have the same sign

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

Noisy and clean pixels are likely to have the same sign

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

Modeling pros/cons of directed and undirected models

MRFs

- Hard to draw samples ✗

(In fact, #P-hard provably, even in Ising models)

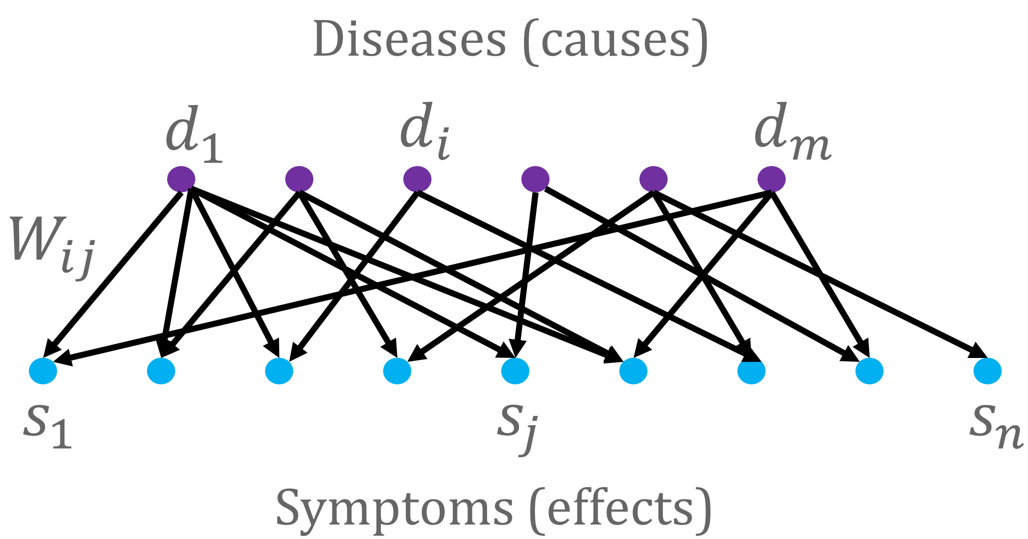
- Models independence structure directly ✓
- Captures soft constraints/energy

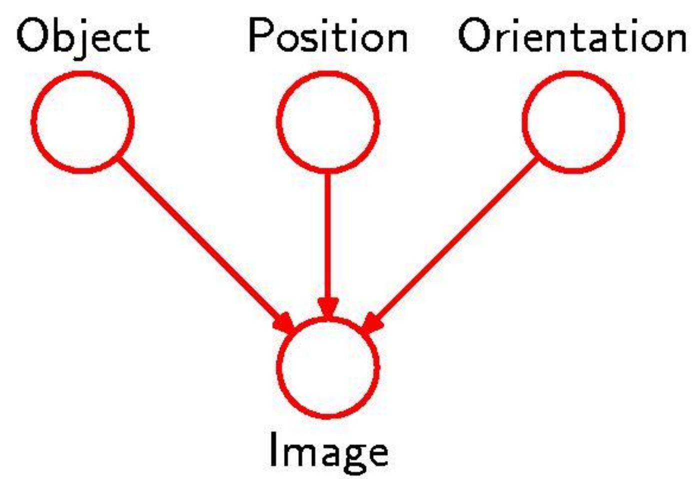
Bayesian networks

- Easy to draw samples ✓
- Models independence structure indirectly ✗
- Captures “causal structure”

Latent variable models

More often than not, we need to model part of the data that is **not observable**. We already saw examples of this:



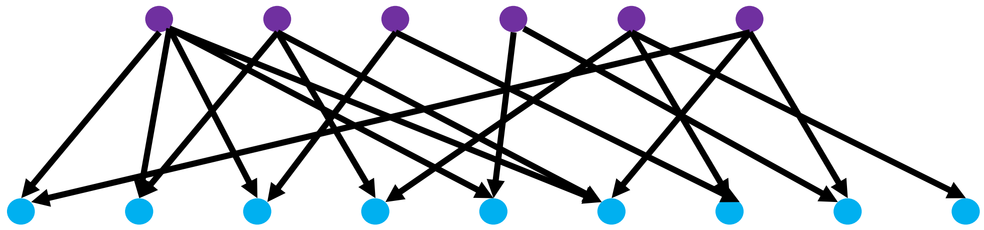


We proceed to see examples of this paradigm, in both the directed and undirected case (we will see many more!!)

Bayesian networks with latent variables

Simple, but powerful paradigm:
single-layer Bayesian networks, where top nodes are latent.

Latent variables Z



Observable variables X

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

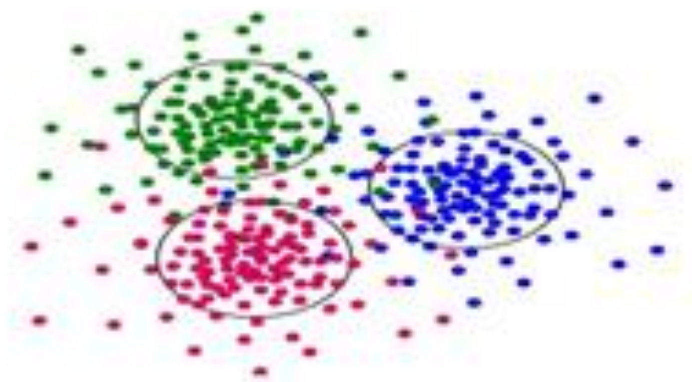
Example 1: Mixture distributions

Mixture models: observables = points; latent = clustering

To draw a sample (X,Z):

Sample Z from a categorial distr. on K components with parameters $\{\pi_i\}$

Sample X from the corresponding component in the mixture.



$$\forall k : \pi_k \geqslant 0 \qquad \sum_{k=1}^{K} \pi_k = 1$$

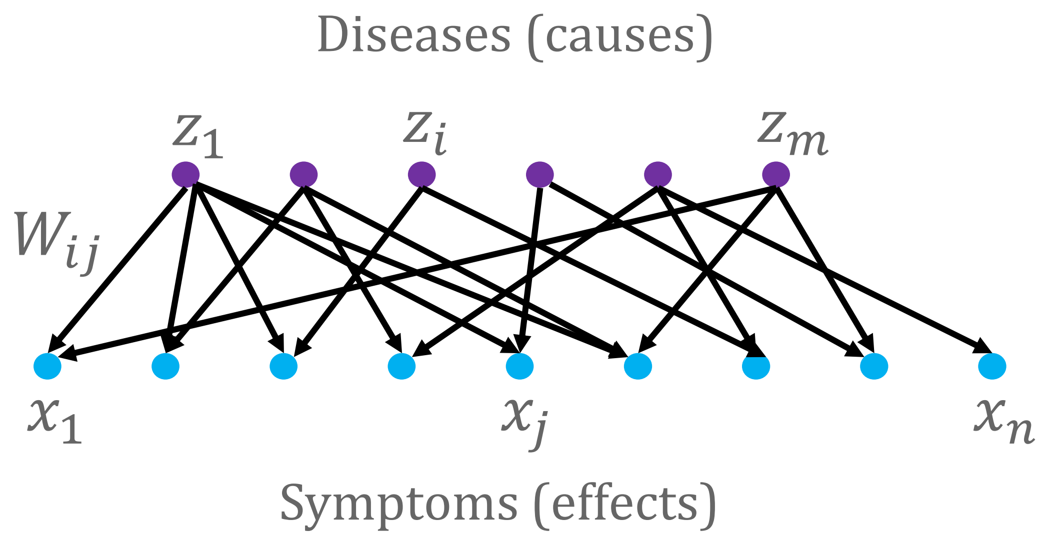
$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \underbrace{\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}_{\text{Component}}$$

Component

Mixing coefficient

Example 2: Noisy-OR networks

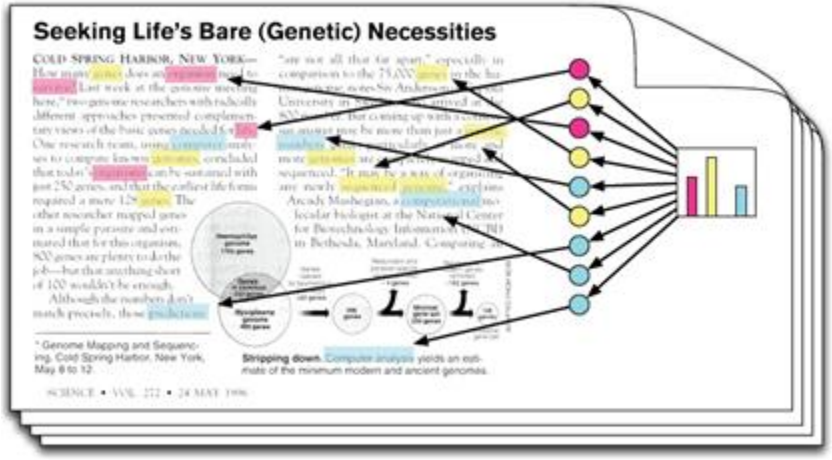
$x_i, z_j \in \{0,1\}$
$W_{ij} \geq 0$



- Sample each z_i is on **independently** with prob. ρ
- When z_i is on, it **activates** x_j with probability $1 - \exp(-W_{ij})$.
- x_j is **on** if one of z_i's **activates** x_j

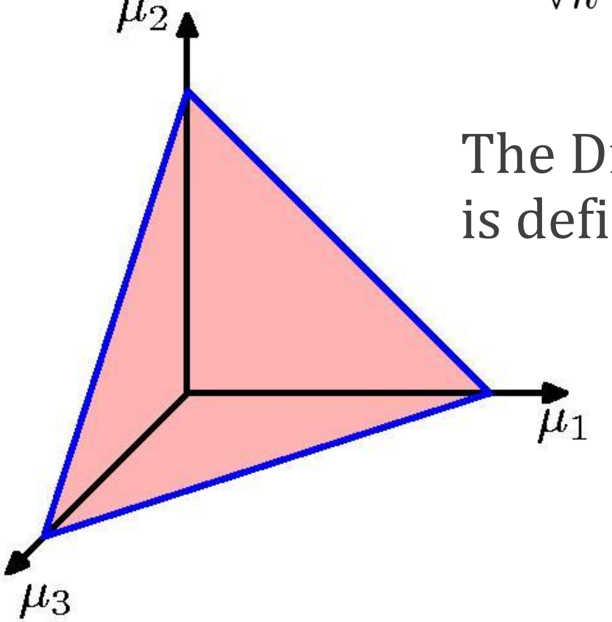
Example 3: Topic models (LDA)

Latent Dirichlet Allocation: famous model for modeling topic structure of documents of text.



Dirichlet Distribution

Consider a distribution over simplex, namely over points $\{\mu_i\}_{i=1}^K$



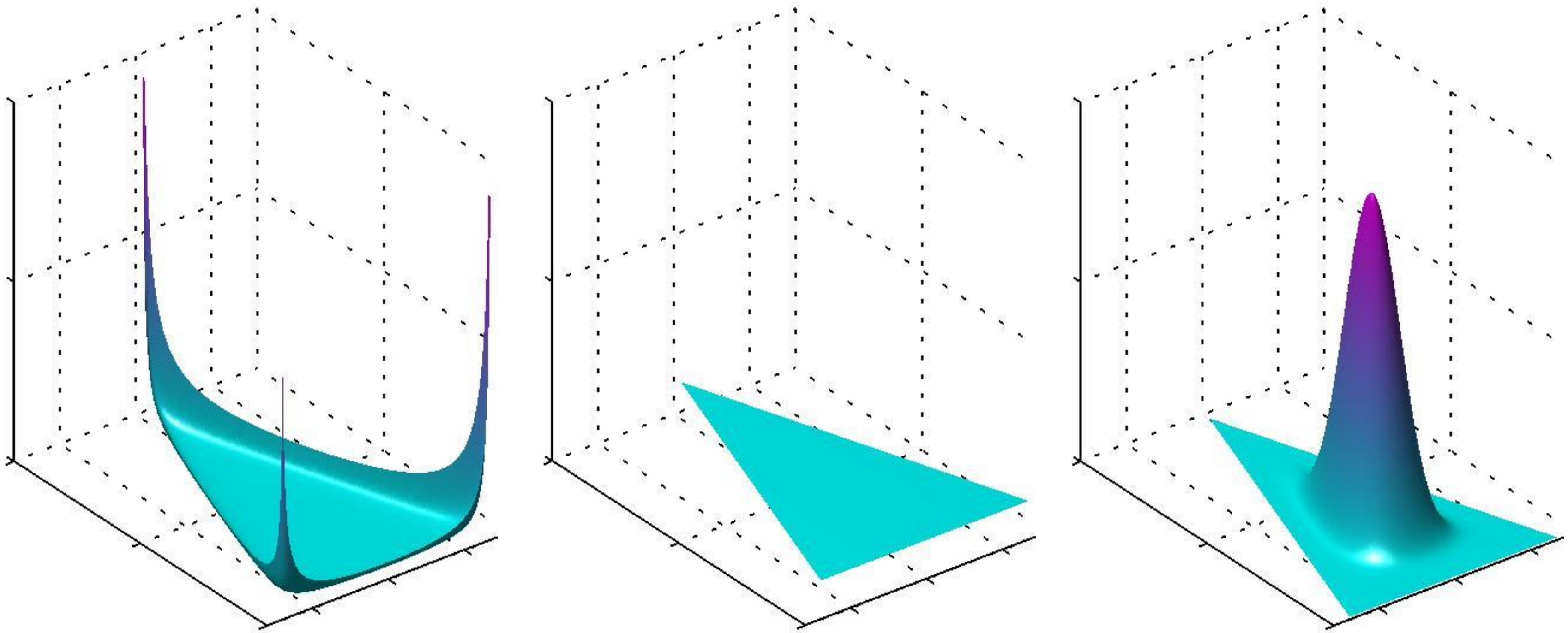
$$\forall k : \mu_k \geqslant 0 \quad \text{and} \quad \sum_{k=1}^{K} \mu_k = 1$$

The Dirichlet distribution (with params $\{\alpha_i \geq 0\}_{i=1}^K$) is defined as:

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

Dirichlet Distribution

Plots of the Dirichlet distribution over three variables.



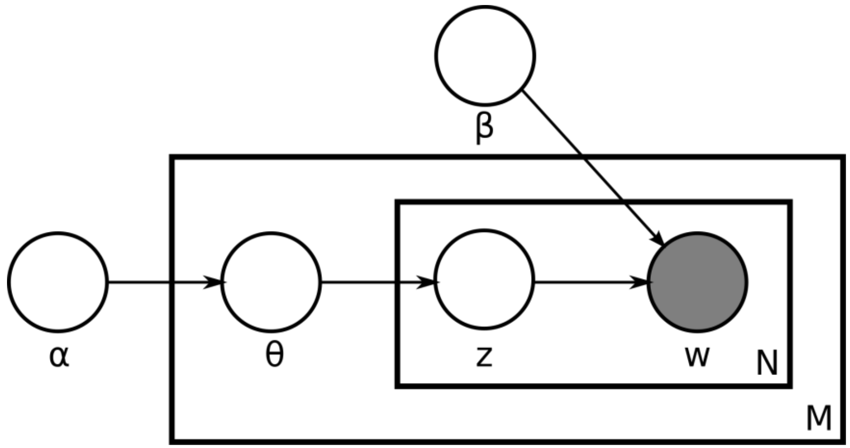
$\alpha_k = 10^{-1}$ $\alpha_k = 10^{0}$ $\alpha_k = 10^{1}$

Example 3: Topic models (LDA)

Defines a distribution over documents, involving K topics.

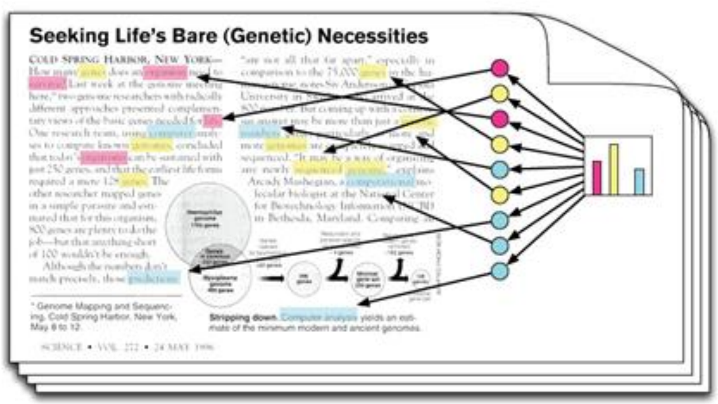
The parameters are: $\{\alpha_i\}_{i=1}^{K}$ (Dirichlet parameters) and matrix $\beta \in \mathbb{R}_+^{N \times K}$, where N is the size of the vocabulary.

The columns of β satisfy $\sum_{j=1}^{N} \beta_{ij} = 1$ (the proportion of words in a topic i)



To produce document:

- First, sample $\theta \sim \mathrm{Dir}(\cdot \mid \alpha)$: this will be the topic proportion vector for the document.
- Each word in the document is generated in order, independently.
- To generate word i:
  - Sample topic z_i with categorical distribution with parameters θ
  - Sample word with categorical distribution with parameters β_{z_i}



The main algorithmic difficulty

Recall, sampling from Bayesian networks is easy.

But: sampling from the posterior distribution P(Z|X) is **hard**:

$$P(\text{Diseases}, \text{Symptoms}) = P(\text{Diseases}) \;\; P(\text{Symptoms}|\text{Diseases})$$

Latent

Data

Simple, explicit

By Bayes rule, $P(\text{Diseases}|\text{Symptoms}) \propto P(\text{Diseases}, \text{Symptoms})$

Up to normalizing const, simple…

Complicated normalizing const:

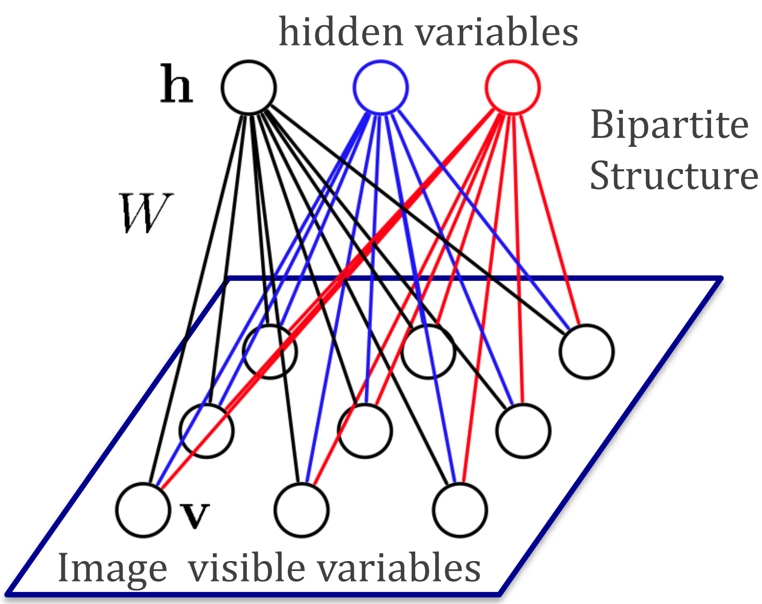
$$\sum_{\text{Diseases}} P(\text{Diseases}, \text{Symptoms})$$

Again, can be #P-hard to sample from!!

Restricted Boltzmann Machines

An **undirected** latent-variable model

We denote visible and hidden variables with vectors **v, h** respectively:



Visible variables $\mathbf{v} \in \{0,1\}^D$
are connected to hidden variables $\mathbf{h} \in \{0,1\}^F$
.

The energy of the joint configuration:

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

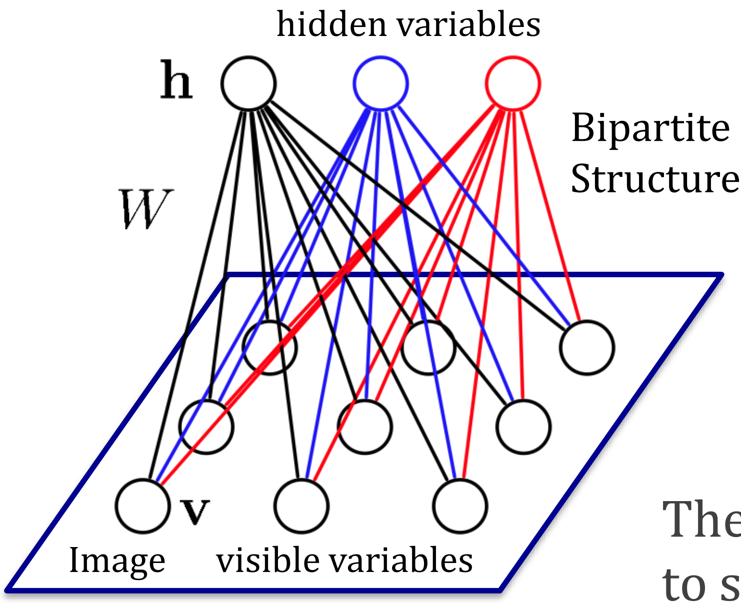
$\theta = \{W, a, b\}$ model parameters.

Probability of the joint configuration is given by the Boltzmann distribution:

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

$$\mathcal{Z}(\theta) = \sum_{\mathbf{h},\mathbf{v}} \exp\big(-E(\mathbf{v},\mathbf{h};\theta)\big)$$

Restricted Boltzmann Machines



Restricted: No interaction between hidden variables

The **posterior** over the hidden variables is easy to sample from! (Conditional independence!)

$$P(\mathbf{h}|\mathbf{v}) = \underbrace{\prod_j P(h_j|\mathbf{v})}_{\text{Factorizes}} \qquad P(h_j = 1|\mathbf{v}) = \frac{1}{1 + \exp(-\sum_i W_{ij} v_i - a_j)}$$

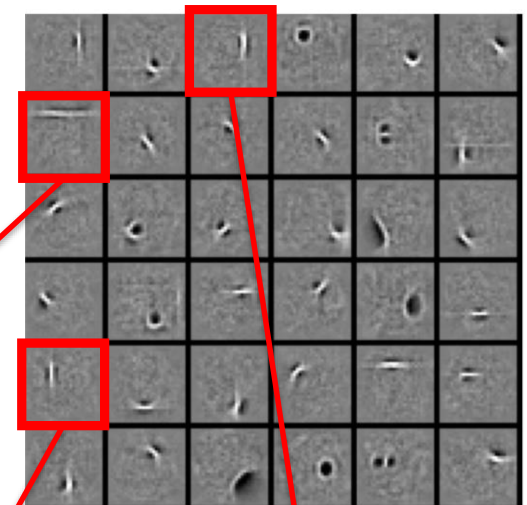
Similarly:

$$P(\mathbf{v}|\mathbf{h}) = \prod_i P(v_i|\mathbf{h}) \qquad P(v_i = 1|\mathbf{h}) = \frac{1}{1 + \exp(-\sum_j W_{ij} h_j - b_i)}$$

Restricted Boltzmann Machines

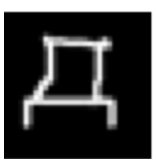
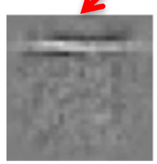
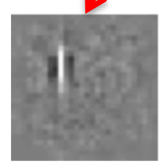
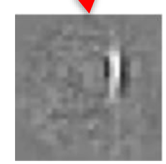
Observed Data
Subset of 25,000 characters

Learned W: “edges”
Subset of 1000 features



Most hidden variables are off

New Image: $p(h_7 = 1|v)$ $p(h_{29} = 1|v)$

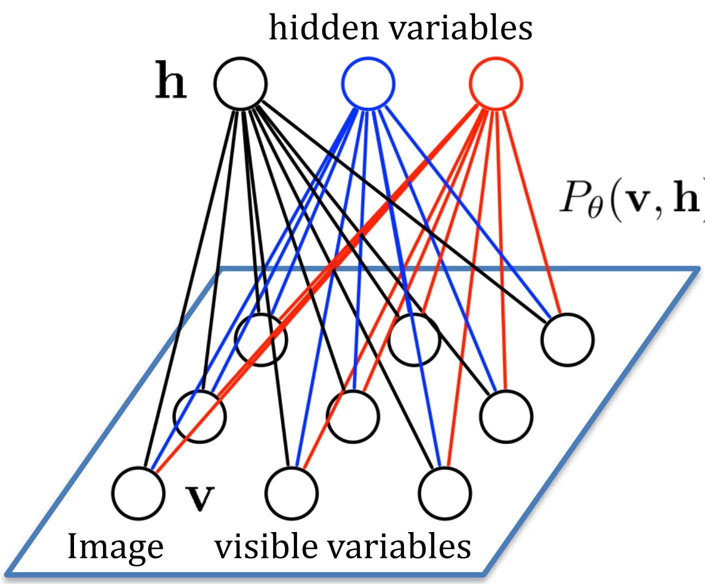
 = $\sigma\Big(0.99 \times$  $+\ 0.97 \times$  $+\ 0.82 \times$  $\cdots\Big)$

$\sigma(x) = \frac{1}{1+\exp(-x)}$

Logistic Function: Suitable for modeling binary images

Represent:  as $P(\mathbf{h}|\mathbf{v}) = [0,\ 0,\ 0.82,\ 0,\ 0,\ 0.99,\ 0,\ 0\ \ldots\ ]$

Gaussian Bernoulli RBMs



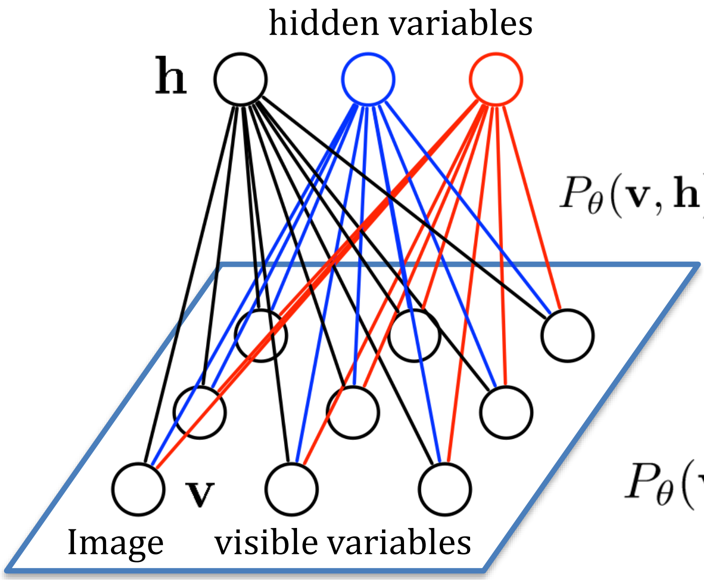
$$P_\theta(\mathbf{v},\mathbf{h}) = \frac{1}{\mathcal{Z}(\theta)} \exp\left(\sum_{i=1}^{D}\sum_{j=1}^{F} \overbrace{W_{ij} h_j \frac{v_i}{\sigma_i}}^{\textbf{Pair-wise}} + \sum_{i=1}^{D} \overbrace{\frac{(v_i - b_i)^2}{2\sigma_i^2}}^{\textbf{Unary}} + \sum_{j=1}^{F} \overbrace{a_j h_j}^{} \right)$$

$$\theta = \{W, a, b\}$$

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

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

Gaussian Bernoulli RBMs



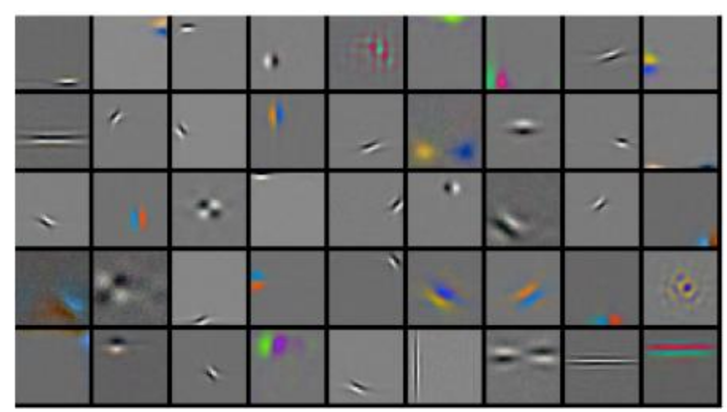
$$P_\theta(\mathbf{v},\mathbf{h}) = \frac{1}{\mathcal{Z}(\theta)} \exp\left(\sum_{i=1}^{D}\sum_{j=1}^{F} \overbrace{W_{ij} h_j \frac{v_i}{\sigma_i}}^{\text{Pair-wise}} + \sum_{i=1}^{D} \overbrace{\frac{(v_i - b_i)^2}{2\sigma_i^2}}^{\text{Unary}} + \sum_{j=1}^{F} \overbrace{a_j h_j}^{} \right)$$

$$\theta = \{W, a, b\}$$

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

4 million **unlabelled** images

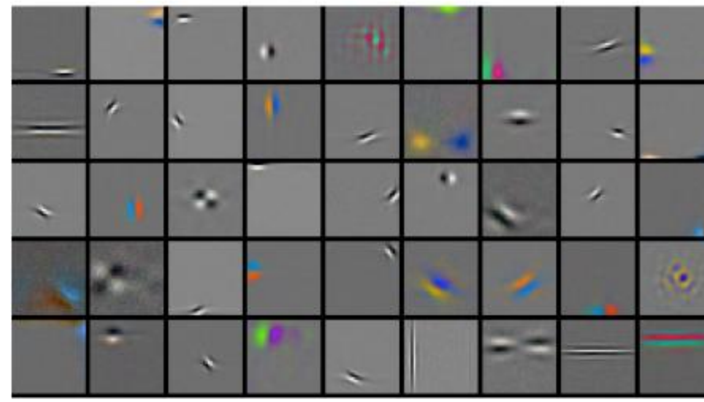
Learned features (out of 10,000)



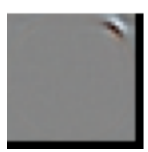
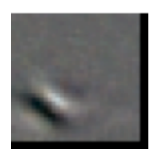
Gaussian Bernoulli RBMs

4 million **unlabelled** images

Learned features (out of 10,000)

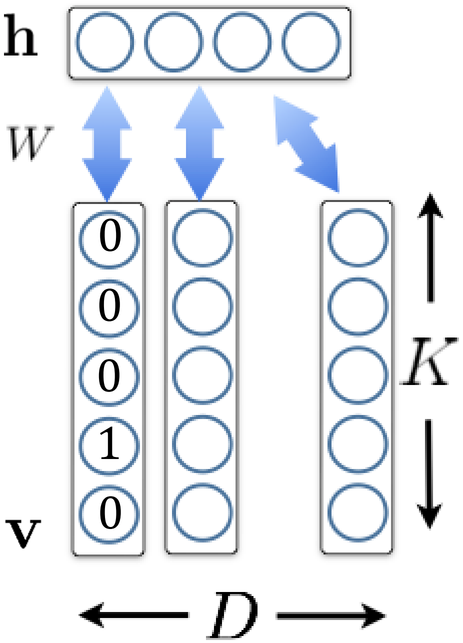


$p(h_7 = 1|v)$ $p(h_{29} = 1|v)$

 = 0.9 *  + 0.8 *  + 0.6 *  ...

New Image

RBMs for Word Counts



$$P_\theta(\mathbf{v},\mathbf{h}) = \frac{1}{\mathcal{Z}(\theta)} \exp\left(\overbrace{\sum_{i=1}^{D}\sum_{k=1}^{K}\sum_{j=1}^{F} W_{ij}^k v_i^k h_j}^{\textbf{Pair-wise}} + \overbrace{\sum_{i=1}^{D}\sum_{k=1}^{K} v_i^k b_i^k + \sum_{j=1}^{F} h_j a_j}^{\textbf{Unary}} \right)$$

$$\theta = \{W, a, b\}$$

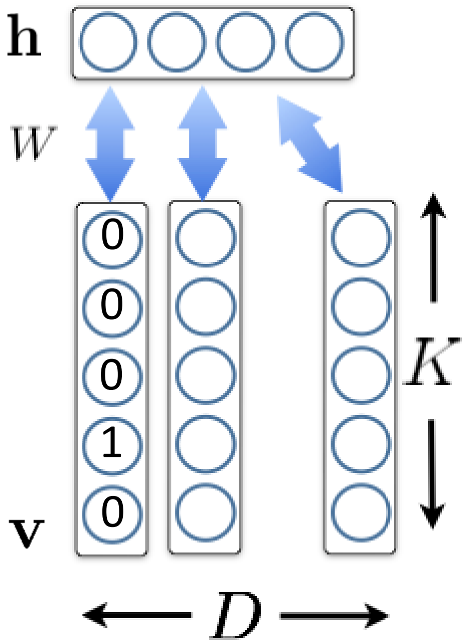
$$P_\theta(v_i^k = 1|\mathbf{h}) = \frac{\exp\left(b_i^k + \sum_{j=1}^F h_j W_{ij}^k\right)}{\sum_{q=1}^K \exp\left(b_i^q + \sum_{j=1}^F h_j W_{ij}^q\right)}$$

Replicated Softmax Model: *undirected* topic model:

- Stochastic 1-of-K visible variables.
- Stochastic binary hidden variables **h**
- Bipartite connections.

(Salakhutdinov & Hinton, NIPS 2010, Srivastava & Salakhutdinov, NIPS 2012)

RBMs for Word Counts



$$P_\theta(\mathbf{v}, \mathbf{h}) = \frac{1}{\mathcal{Z}(\theta)} \exp\left(\overbrace{\sum_{i=1}^{D} \sum_{k=1}^{K} \sum_{j=1}^{F} W_{ij}^k v_i^k h_j}^{\textbf{Pair-wise}} + \overbrace{\sum_{i=1}^{D} \sum_{k=1}^{K} v_i^k b_i^k}^{\textbf{Unary}} + \overbrace{\sum_{j=1}^{F} h_j a_j}^{} \right)$$

$$\theta = \{W, a, b\}$$

$$P_\theta(v_i^k = 1 | \mathbf{h}) = \frac{\exp\left(b_i^k + \sum_{j=1}^{F} h_j W_{ij}^k\right)}{\sum_{q=1}^{K} \exp\left(b_i^q + \sum_{j=1}^{F} h_j W_{ij}^q\right)}$$

Reuters dataset:
804,414 **unlabeled**
newswire stories
Bag-of-Words

Learned features: ``topics''

russian	clinton	computer	trade	stock
russia	house	system	country	wall
moscow	president	product	import	street
yeltsin	bill	software	world	point
soviet	congress	develop	economy	dow

RBMs for Word Counts

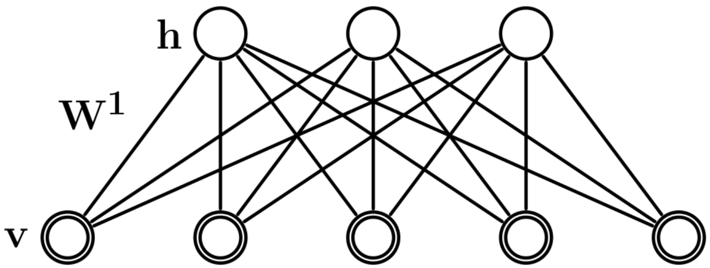
One-step reconstruction from the Replicated Softmax model.

Input	**Reconstruction**
chocolate, cake	cake, chocolate, sweets, dessert, cupcake, food, sugar, cream, birthday
nyc	nyc, newyork, brooklyn, queens, gothamist, manhattan, subway, streetart
dog	dog, puppy, perro, dogs, pet, filmshots, tongue, pets, nose, animal
flower, high, 花	flower, 花, high, japan, sakura, 日本, blossom, tokyo, lily, cherry
girl, rain, station, norway	norway, station, rain, girl, oslo, train, umbrella, wet, railway, weather
fun, life, children	children, fun, life, kids, child, playing, boys, kid, play, love
forest, blur	forest, blur, woods, motion, trees, movement, path, trail, green, focus
españa, agua, granada	españa, agua, spain, granada, water, andalucía, naturaleza, galicia, nieve

Collaborative Filtering

$$P_\theta(\mathbf{v},\mathbf{h}) = \frac{1}{\mathcal{Z}(\theta)} \exp\left(\sum_{ijk} W_{ij}^k v_i^k h_j + \sum_{ik} b_i^k v_i^k + \sum_j a_j h_j\right)$$

Binary hidden: user preferences



Multinomial visible: user ratings

Netflix dataset:
480,189 users
17,770 movies
Over 100 million ratings

Learned features: ``genre''

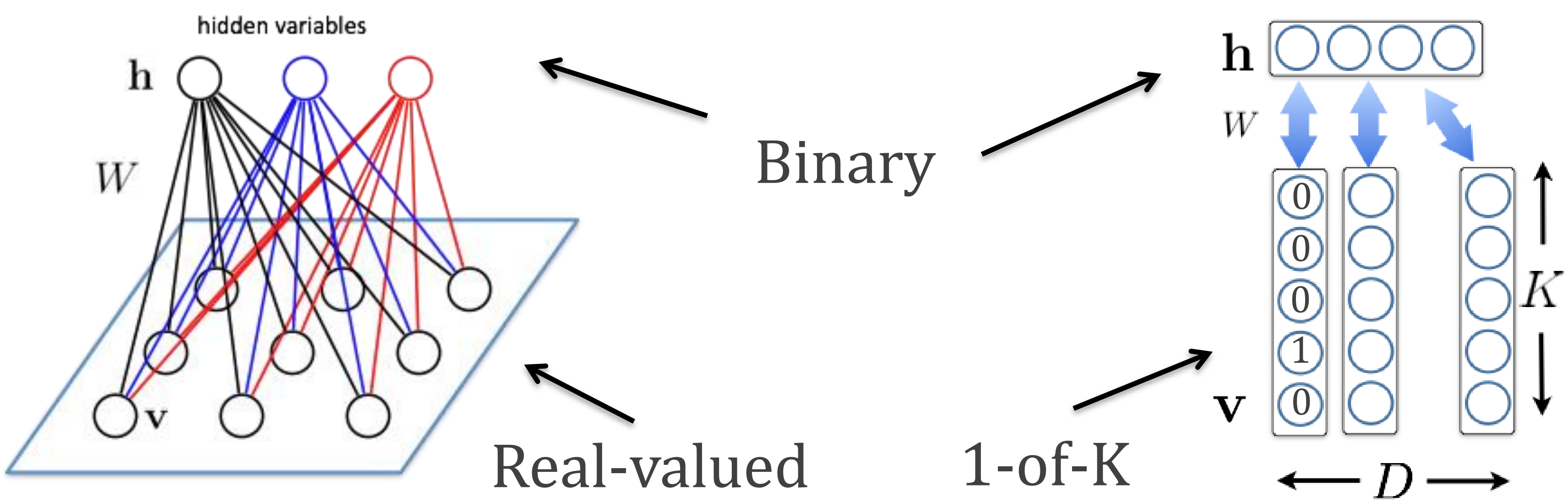
Fahrenheit 9/11	Independence Day
Bowling for Columbine	The Day After Tomorrow
The People vs. Larry Flynt	Con Air
Canadian Bacon	Men in Black II
La Dolce Vita	Men in Black
Friday the 13th	Scary Movie
The Texas Chainsaw Massacre	Naked Gun
Children of the Corn	Hot Shots!
Child's Play	American Pie
The Return of Michael Myers	Police Academy

State-of-the-art performance on the Netflix dataset.

(Salakhutdinov, Mnih, Hinton, ICML 2007)

Different Data Modalities

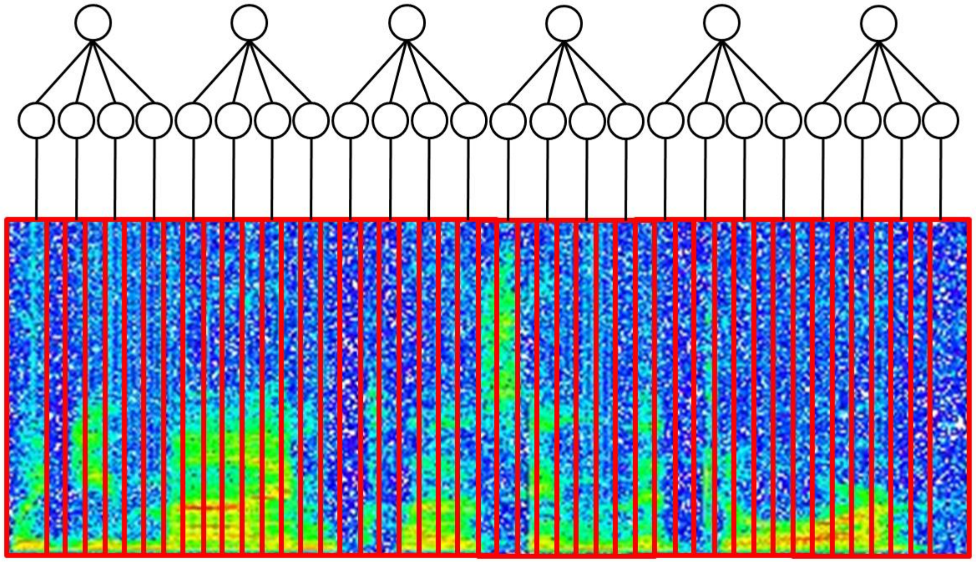
Binary/Gaussian/Softmax RBMs: All have binary hidden variables but use them to model different kinds of data.

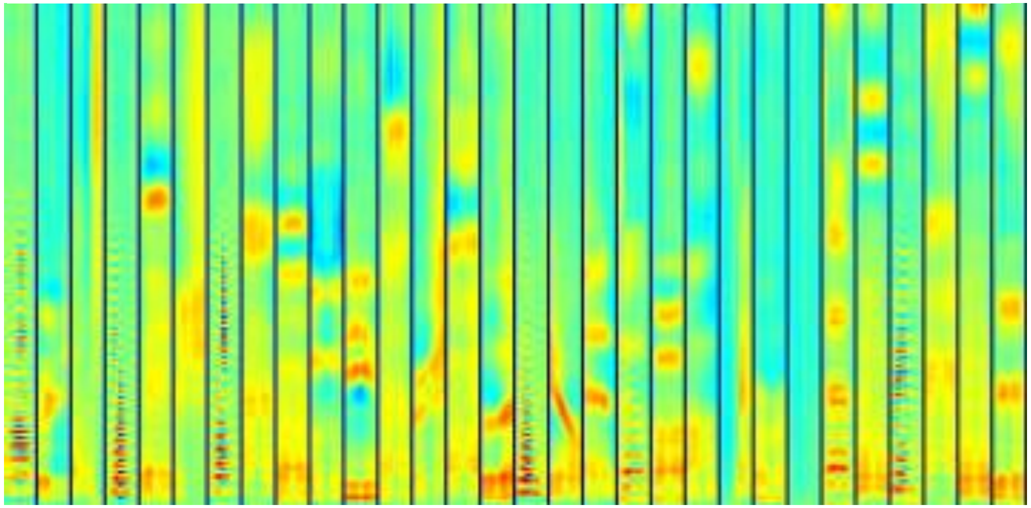


It is easy to infer the states of the hidden variables:

$$P_\theta(\mathbf{h}|\mathbf{v}) = \prod_{j=1}^{F} P_\theta(h_j|\mathbf{v}) = \prod_{j=1}^{F} \frac{1}{1 + \exp(-a_j - \sum_{i=1}^{D} W_{ij} v_i)}$$

Speech

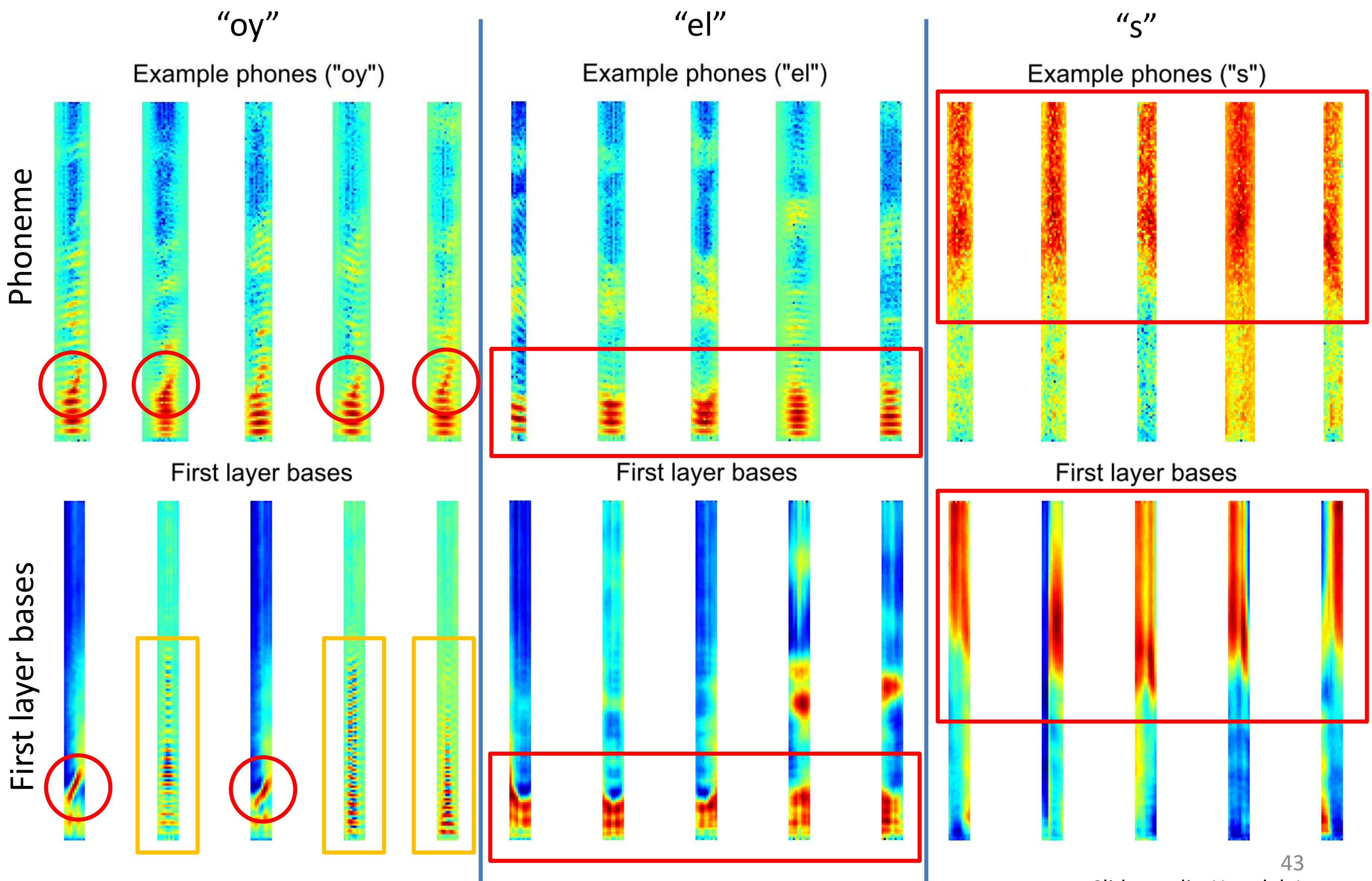




Learned first-layer bases

Lee et.al., NIPS 2009

Comparison of bases to phonemes



Slide credit: Honglak Lee

Algorithmic pros/cons of latent-variable models (so far)

RBM's

- Hard to draw samples ✗

(In fact, #P-hard provably, even in Ising models)

- Easy to sample posterior distribution over latents ✓

Directed models

- Easy to draw samples ✓

- Hard to sample posterior distribution over latents ✗

(In fact, #P-hard provably, even in Ising models)