Machine Learning
Latent Variables

Rajesh Ranganath

So far in class

- Given an $\mathbf{x}$ and a $\mathbf{y}$
- Find relationship between $\mathbf{x}$ and $\mathbf{y}$

Linear models for all kinds of data

Gradients optimize

Regularization to control model complexity
especially when features > data

Underfitting vs Overfitting/Bias vs Variance

Neural networks and how convnets reduce variance

How to build random forests/why random forests don't overfit

Randomness prevents overfitting with more compute in random forests

Randomness makes finding feature combinations hard

Breakout Question

- *A model trained with finite data perfectly fits the training data, but does well on test. Is this possible?*

What happens if some of $x_{i,j}$ are missing?

How do we fill in x and predict y?

Regression is conditional

$$p(y \mid \mathbf{x})$$

Need a full model

$$p(y, \mathbf{x})$$

Can fill in missing x_j

$$p(\mathbf{x}_{\text{missing}} \mid \mathbf{x}_{\text{observed}}) = \frac{p(\mathbf{x}_{\text{missing}}, \mathbf{x}_{\text{observed}})}{p(\mathbf{x}_{\text{observed}})} = \frac{\int p(y, \mathbf{x}_{\text{missing}}, \mathbf{x}_{\text{observed}}) dy}{\int p(y, \mathbf{a}, \mathbf{x}_{\text{observed}}) dy d\mathbf{a}}$$

Regression is conditional

$$p(y \mid \mathbf{x})$$

Need a full model

$$p(y, \mathbf{x})$$

Can make predictions

$$p(y \mid \mathbf{x}) = \frac{p(y, \mathbf{x})}{p(\mathbf{x})} = \frac{p(y, \mathbf{x})}{\int p(y, \mathbf{x}) dy}$$

Is there any difference between $\mathbf{x}$ and y with

$$p(\mathbf{x}, y)$$

Is there any difference between $\mathbf{x}$ and y with

$$p(\mathbf{x}, y)$$

All are just variables

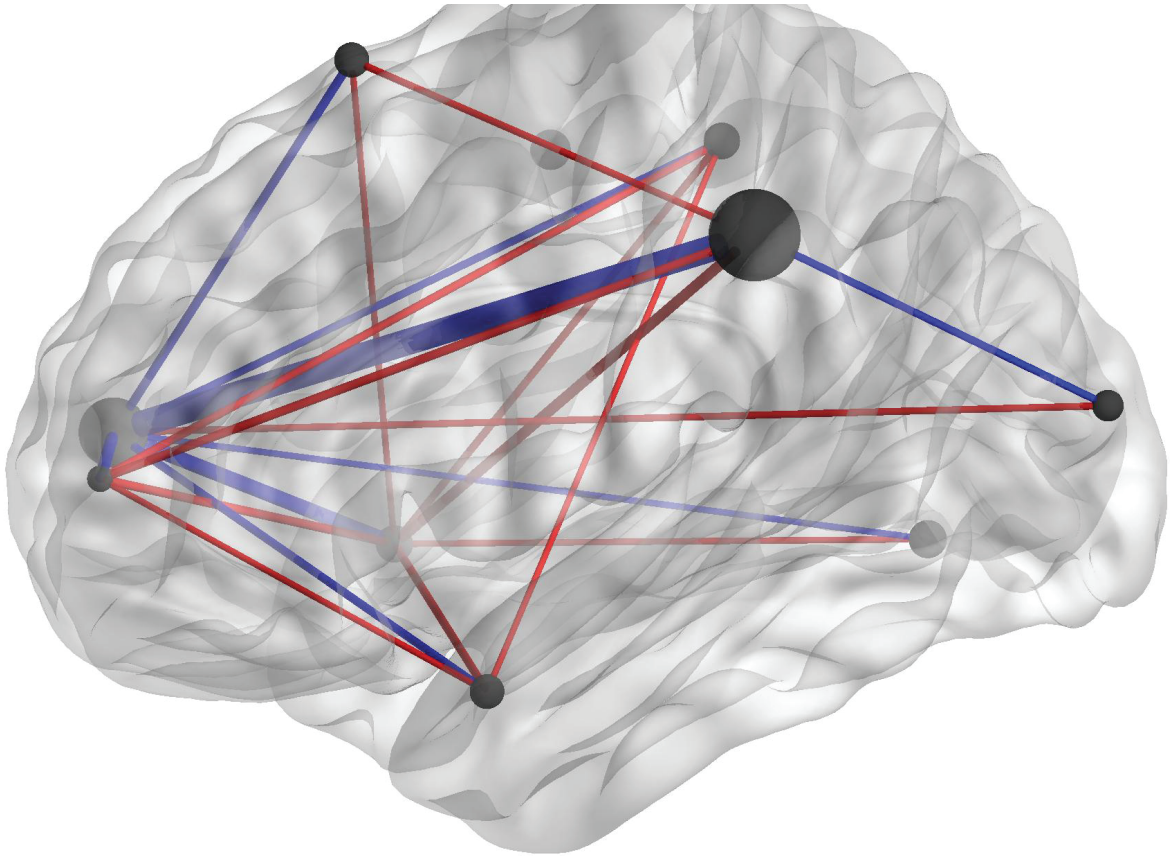
$$p(\mathbf{x}, y)$$

When predicting y, it's "missing"

Predicting with missing data is a more complex model

Are there other types of variables?

Neuroscience

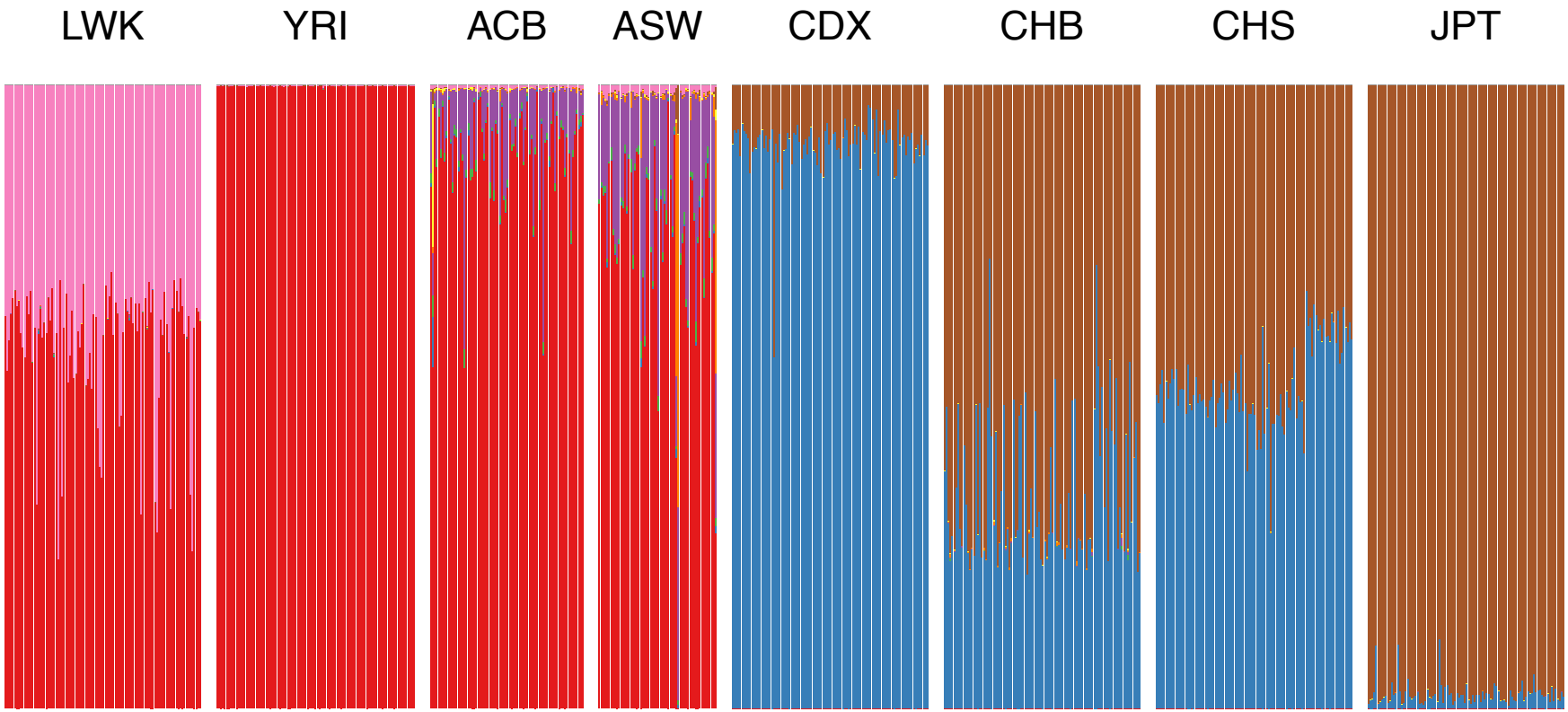


[Manning+ 2014]

Astrophysics

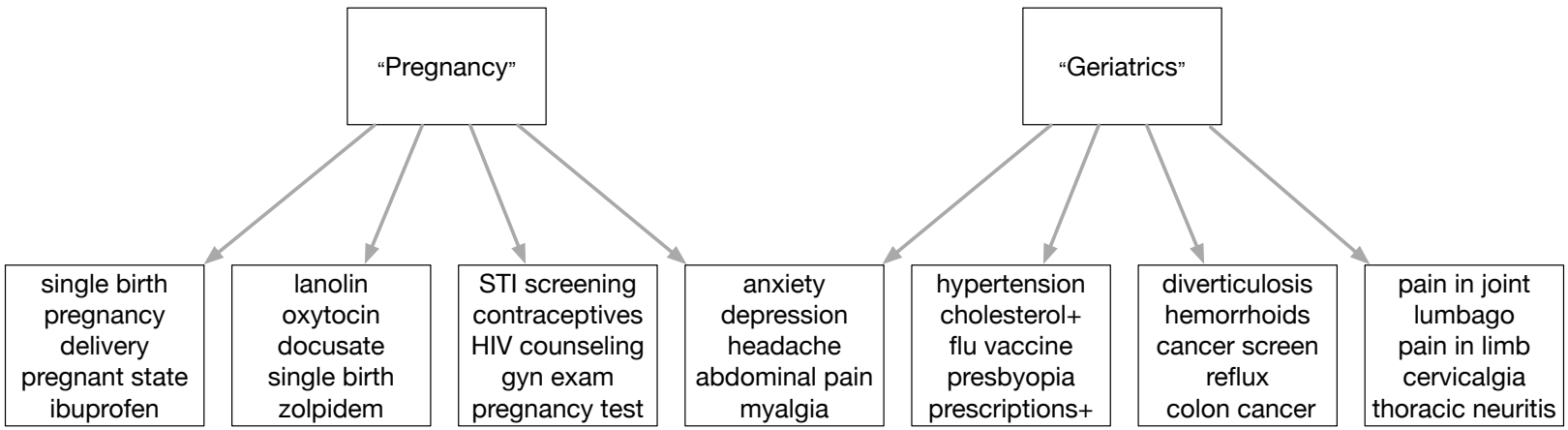
[Regier+ 2015]

Genetics



[Gopalan+ 2016]

Medicine


"Pregnancy"
"Geriatrics"
single birth
pregnancy
delivery
pregnant state
ibuprofen
lanolin
oxytocin
docusate
single birth
zolpidem
STI screening
contraceptives
HIV counseling
gyn exam
pregnancy test
anxiety
depression
headache
abdominal pain
myalgia
hypertension
cholesterol+
flu vaccine
presbyopia
prescriptions+
diverticulosis
hemorrhoids
cancer screen
reflux
colon cancer
pain in joint
lumbago
pain in limb
cervicalgia
thoracic neuritis

[R+]

None of these things are observed

- But they relate to the data generating process

- They are a summary of the data we hoped to exist

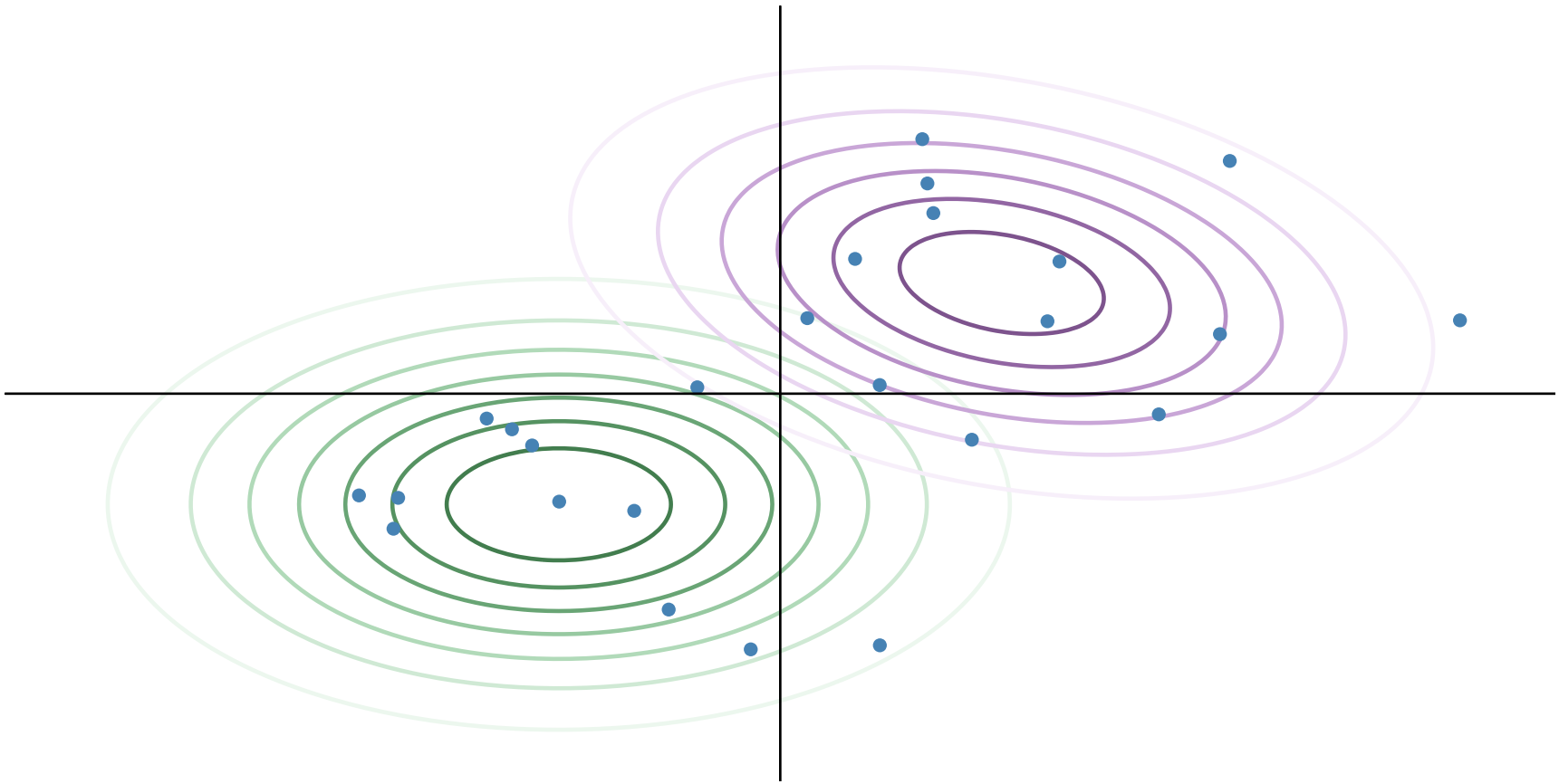
Formal definition

Probabilistic latent variable models:

- Data: $\mathbf{x}$
- Hidden Structure (latent variables): $\mathbf{z}$
- Model: $p(\mathbf{x}, \mathbf{z}) = p(\mathbf{z})p(\mathbf{x}|\mathbf{z})$ - prior $p(\mathbf{z})$ and likelihood $p(\mathbf{x}|\mathbf{z})$

An Example Latent Variable

Mixture model: z_n represents a group for each data point



A Mixture of Gaussians

Suppose data points $\mathbf{x}_1, ...\mathbf{x}_n$. How do we build a model?

A Mixture of Gaussians

Suppose data points $\mathbf{x}_1, \ldots \mathbf{x}_n$. How do we build a model?

What's the hidden structure to be found?
Assign class from 1…K for each data point

A Mixture of Gaussians

Suppose data points $\mathbf{x}_1, \ldots \mathbf{x}_n$. How do we build a model?

What's the prior for the hidden structure?

A Mixture of Gaussians

Suppose data points $\mathbf{x}_1, ...\mathbf{x}_n$. How do we build a model?

Need a prior for the class for each data points

$$p(z_i = k) = \frac{1}{K}$$

A Mixture of Gaussians

Suppose data points $\mathbf{x}_1, ...\mathbf{x}_n$. How do we build a model?

Need a way to define the each class 1...K

A Mixture of Gaussians

Suppose data points $\mathbf{x}_1, ...\mathbf{x}_n$. How do we build a model?

Define by how the data looks like from the class

$$\mathbf{x}_i \,|\, z_i = k \sim \text{Normal}(\mu_k, \Sigma_k)$$

Called the likelihood

A Mixture of Gaussians

Suppose data points $\mathbf{x}_1, ...\mathbf{x}_n$. How do we build a model?

Total likelihood for n datapoints

$$p(\mathbf{x}, \mathbf{z}) = \prod_{i=1}^{n} \underbrace{p(z_i)}_{prior} \underbrace{p(\mathbf{x}_i \mid z_i)}_{likelihood}$$

A Mixture of Gaussians

Suppose data points $\mathbf{x}_1, ...\mathbf{x}_n$. How do we build a model?

How to determine the structure (color) for a datapoint

$$p(z_i = k \,|\, \mathbf{x}_i) = \frac{p(z = k, \mathbf{x}_i)}{p(\mathbf{x}_i)}$$

Called the posterior distribution

A Mixture of Gaussians

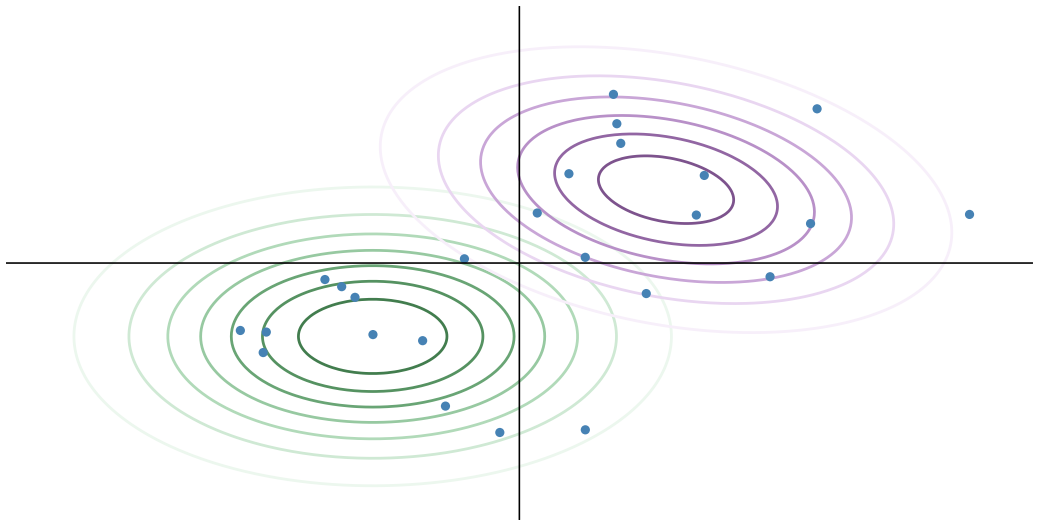
Suppose data points $\mathbf{x}_1, \ldots \mathbf{x}_n$. How do we build a model?

$$p(z_i = k \mid \mathbf{x}_i) = \frac{p(z = k, \mathbf{x}_i)}{p(\mathbf{x}_i)} = \frac{\frac{1}{K}\text{Normal}(\mathbf{x}_i; \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \frac{1}{K}\text{Normal}(\mathbf{x}_i; \mu_j, \Sigma_j)}$$
$$= \frac{\text{Normal}(\mathbf{x}_i; \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \text{Normal}(\mathbf{x}_i; \mu_j, \Sigma_j)}$$

Points are colored proportional to the density the Gaussian of a color assigns.

A Mixture of Gaussians

Suppose data points $\mathbf{x}_1, \ldots \mathbf{x}_n$. How do we build a model?



Training a Mixture of Gaussians

Parameters:

$$\mu_k, \Sigma_k, \quad k = 1...K$$

How do we train the models?

Training a Mixture of Gaussians

Parameters:

$$\mu_k, \Sigma_k, \quad k = 1...K$$

How do we train the models?

One way: Use the maximum likelihood recipe

Training a Mixture of Gaussians: Maximum Likelihood

Likelihood of data

$$\log p(\mathbf{x}_1, ..., \mathbf{x}_n; \mu_{1...K}, \Sigma_{1...K}) = \sum_{i=1}^{n} \log p(\mathbf{x}_i)$$

Introduce the latent variable

$$\sum_{i=1}^{n} \log p(\mathbf{x}_i) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} p(\mathbf{x}_i \mid \mathbf{z}_i = k; \mu_k, \Sigma_k) p(\mathbf{z}_i = k)$$

Train by traking gradients. Example for μ_k

$$\nabla_{\mu_k} \sum_{i=1}^{n} \log p(\mathbf{x}_i) = \sum_{i=1}^{n} \nabla_{\mu_k} \log \sum_{k=1}^{K} p(\mathbf{x}_i \mid \mathbf{z}_i = k; \mu_k, \Sigma_k) p(\mathbf{z}_i = k)$$

Training a Mixture of Gaussians: Maximum Likelihood

Likelihood of data

$$\log p(\mathbf{x}_1, ..., \mathbf{x}_n; \mu_{1...K}, \Sigma_{1...K}) = \sum_{i=1}^{n} \log p(\mathbf{x}_i)$$

Introduce the latent variable

$$\sum_{i=1}^{n} \log p(\mathbf{x}_i) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} p(\mathbf{x}_i \mid \mathbf{z}_i = k; \mu_k, \Sigma_k) p(\mathbf{z}_i = k)$$

Train by traking gradients. Example for μ_k

$$\nabla_{\mu_k} \sum_{i=1}^{n} \log p(\mathbf{x}_i) = \sum_{i=1}^{n} \nabla_{\mu_k} \log \sum_{k=1}^{K} p(\mathbf{x}_i \mid \mathbf{z}_i = k; \mu_k, \Sigma_k) p(\mathbf{z}_i = k)$$

Can the sum in the sum move outside the log?

What does maximum likelihood do?

- Take a model $p_{\boldsymbol{\theta}}$
- Take a data distribution F

$$\max_{\boldsymbol{\theta}} \mathbb{E}_F[\log p_{\boldsymbol{\theta}}]$$

Translate

$$\max_{\boldsymbol{\theta}} \mathbb{E}_F[\log p_{\boldsymbol{\theta}}] - E_F[\log F]$$

Transform

$$\mathbb{E}_F[\log p_{\boldsymbol{\theta}}] - E_F[\log F] = -\mathrm{KL}(F||p_{\boldsymbol{\theta}})$$

Maximum likelihood is the same as minimizing KL-divergence

Minimizing the KL divergence

Maximum likelihood is the same as minimizing the KL-divergence

$$\max_{\boldsymbol{\theta}} \mathbb{E}_p[\log p_{\boldsymbol{\theta}}]$$

KL-divergence properties

- $\mathrm{KL}(p||q) \geq 0$
- $\mathrm{KL}(p||q) = 0 \iff p = q$

Maximum likelihood

- Minimizes KL-divergence from the data generating distribution to the model
- This means maximum likelihood move the model closes to the data generating distribution

A more powerful mixture model

Use a “very-flexible” distribution

$$p_{\boldsymbol{\theta}}(\mathbf{x}|z=k)$$

Minimize KL-divergence

$$\mathrm{KL}(F||p_{\boldsymbol{\theta}}) = \mathrm{KL}\left(F||\frac{1}{K}\sum_{k=1}^{K} p_{\boldsymbol{\theta}}(\mathbf{x}|z=k)\right)$$

Solution?

A more powerful mixture model

Use a “very-flexible” distribution

$$p_{\boldsymbol{\theta}}(\mathbf{x}|z=k)$$

Minimize KL-divergence

$$\mathrm{KL}(F||p_{\boldsymbol{\theta}}) = \mathrm{KL}\left(F||\frac{1}{K}\sum_{k=1}^{K} p_{\boldsymbol{\theta}}(\mathbf{x}|z=k)\right)$$

Solution?

$$p_{\boldsymbol{\theta}}(\mathbf{x}|z=k) = F(\mathbf{x}) \implies \frac{1}{K}\sum_{k=1}^{K} p_{\boldsymbol{\theta}}(\mathbf{x}|z=k) = F(\mathbf{x})$$

$$\implies \mathrm{KL}(F||\frac{1}{K}\sum_{k=1}^{K} p_{\boldsymbol{\theta}}(\mathbf{x}|z=k)) = \mathrm{KL}(F||F) = 0$$

Optimal solution is to set each mixture component to the data generating distribution

A more powerful mixture model

Mixture model's goal was to figure out the color (hidden structure)

$$
\begin{aligned}
p(z_i = k \mid \mathbf{x}_i) &= \frac{p(z_i = k) p_{\boldsymbol{\theta}}(\mathbf{x}_i \mid z_i = k)}{\sum_{j=1}^{K} p(z_i = j) p_{\boldsymbol{\theta}}(\mathbf{x}_i \mid z_i = j)} \\
&= \frac{\frac{1}{K} F(\mathbf{x}_i)}{\sum_{j=1}^{K} \frac{1}{K} F(\mathbf{x}_i)} \\
&= \frac{\frac{1}{K} F(\mathbf{x}_i)}{F(\mathbf{x}_i)} \\
&= \frac{1}{K}
\end{aligned}
$$

All colors have same probability.

Was there a point to building mixture models?

Latent Variable Models

Imagine access to the data generating distribution

$$F(\mathbf{x})$$

Imagine a latent variable model

$$p(\mathbf{x}, \mathbf{z})$$

There are more variables in the model than the data. Lots of perfect fit solutions

$$p(\mathbf{x}, \mathbf{z}) = F(\mathbf{x})p(\mathbf{z}) \implies \mathrm{KL}\left(F || \int p(\mathbf{x}, \mathbf{z})dz\right) = 0$$

Latent variable models need assumptions

How do we place limitations?

Relationships between observables controlled by latent variables

$$p(\mathbf{x}[0], \mathbf{x}[1]) = \int p(\mathbf{x}[0] \mid z) p(\mathbf{x}[1] \mid z) p(z) dz$$

If the model learns to make $\mathbf{x}$ and $\mathbf{z}$ independent,

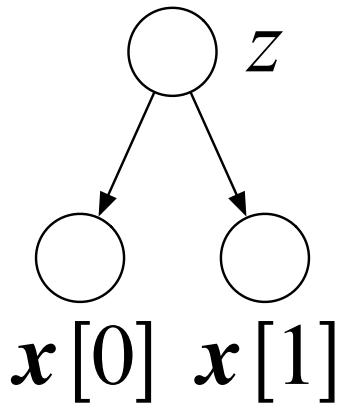
$$p(\mathbf{x}, \mathbf{z}) = p(\mathbf{x}) p(\mathbf{z})$$

The dimensions of $\mathbf{x}$ would also be independent

$$\begin{aligned} p(\mathbf{x}[0], \mathbf{x}[1]) &= \int p(\mathbf{x}[0] \mid z) p(\mathbf{x}[1] \mid z) p(z) dz \\ &= \int p(\mathbf{x}[0]) p(\mathbf{x}[1]) p(z) dz = p(\mathbf{x}[0]) p(\mathbf{x}[1]) \end{aligned}$$

This is testable in data

A Graphical View



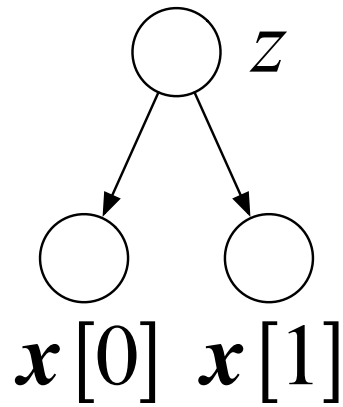
Graph describes factorization of the joint

$$p(z, \mathbf{x}[0], \mathbf{x}[1]) = p(z)p(\mathbf{x}[0]|z)p(\mathbf{x}[1]|z)$$

Graph defines a model family.

Can someone describe it in words?

A Graphical View



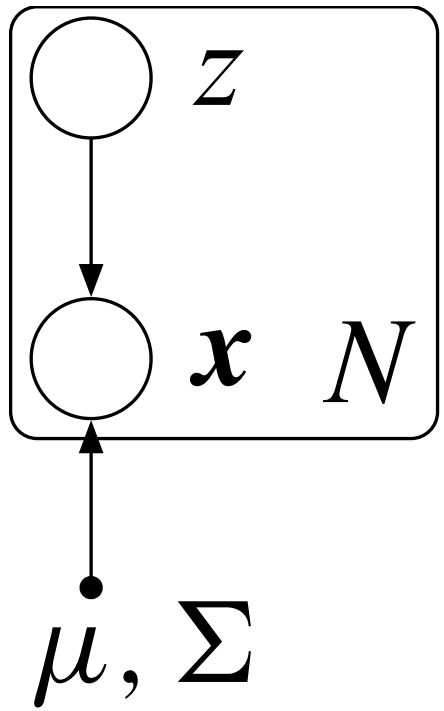
Graphs describe conditional independences

$$\mathbf{x}[0] \perp\!\!\!\perp \mathbf{x}[1] \,|\, z.$$

Proof:

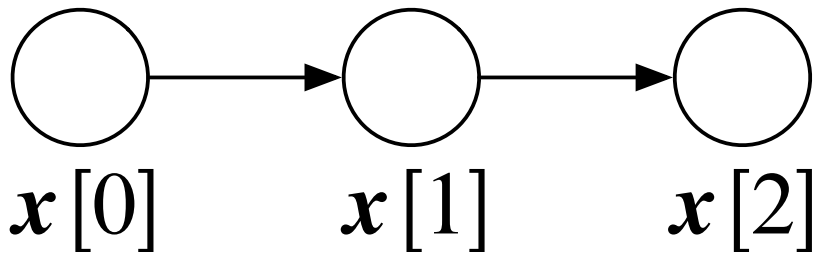
$$\begin{aligned} p(\mathbf{x}[0], \mathbf{x}[1] \,|\, z) &= \frac{p(\mathbf{x}[0], \mathbf{x}[1], z)}{p(z)} \\ &= \frac{p(z)p(\mathbf{x}[0] \,|\, z)p(\mathbf{x}[1] \,|\, z)}{p(z)} = p(\mathbf{x}[0] \,|\, z)p(\mathbf{x}[1] \,|\, z) \end{aligned}$$

A Graphical View



The plate denotes a collection of variables. What are the independence assumptions?

A Graphical View



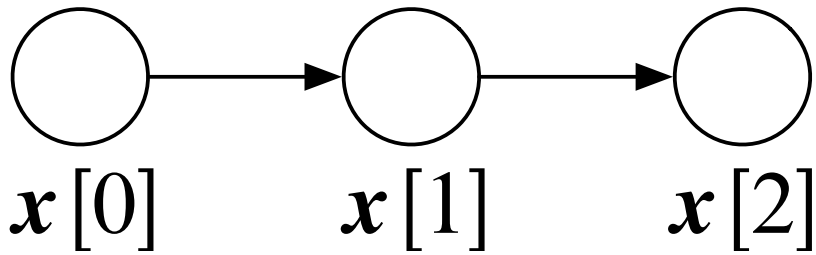
Joint distribution

$$p(\mathbf{x}[0], \mathbf{x}[1], \mathbf{x}[2]) = p(\mathbf{x}[0])p(\mathbf{x}[1]|\mathbf{x}[0])p(\mathbf{x}[2]|\mathbf{x}[1], \mathbf{x}[0])$$

Each variable sequentially depends on the next in model

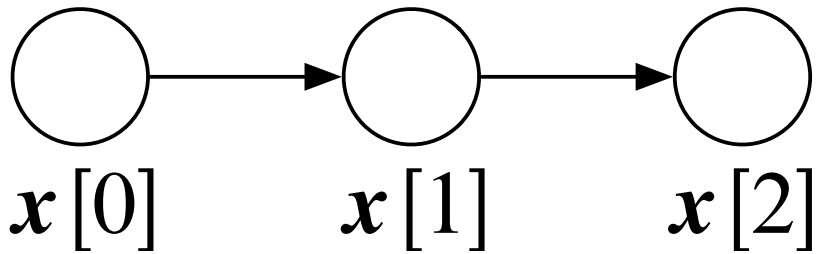
$$p(\mathbf{x}[0], \mathbf{x}[1], \mathbf{x}[2]) = p(\mathbf{x}[0])p(\mathbf{x}[1]|\mathbf{x}[0])p(\mathbf{x}[2]|\mathbf{x}[1])$$

A Graphical View



Is $\mathbf{x}[0]$ independent of $\mathbf{x}[2]$ given $\mathbf{x}[1]$?

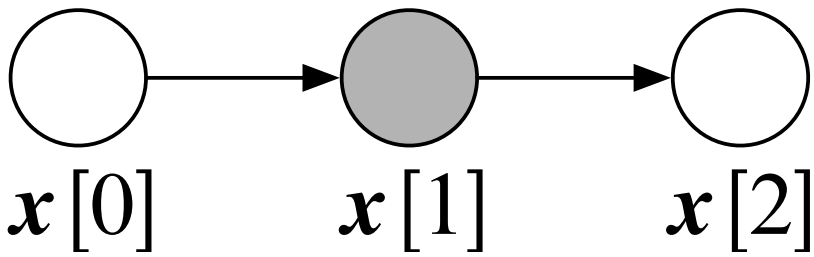
A Graphical View



Is $\mathbf{x}[0]$ independent of $\mathbf{x}[2]$ given $\mathbf{x}[1]$?

$$
\begin{aligned}
p(\mathbf{x}[0], \mathbf{x}[2] \mid \mathbf{x}[1]) &= \frac{p(\mathbf{x}[0], \mathbf{x}[2], \mathbf{x}[1])}{p(\mathbf{x}[1])} \\
&= \frac{p(\mathbf{x}[0])p(\mathbf{x}[1] \mid \mathbf{x}[0])p(\mathbf{x}[2] \mid \mathbf{x}[1])}{p(\mathbf{x}[1])} \\
&= \frac{p(\mathbf{x}[0], \mathbf{x}[1])p(\mathbf{x}[2] \mid \mathbf{x}[1])}{p(\mathbf{x}[1])} \\
&= \frac{p(\mathbf{x}[0] \mid \mathbf{x}[1])p(\mathbf{x}[1])p(\mathbf{x}[2] \mid \mathbf{x}[1])}{p(\mathbf{x}[1])} \\
&= p(\mathbf{x}[0] \mid \mathbf{x}[1])p(\mathbf{x}[2] \mid \mathbf{x}[1])
\end{aligned}
$$

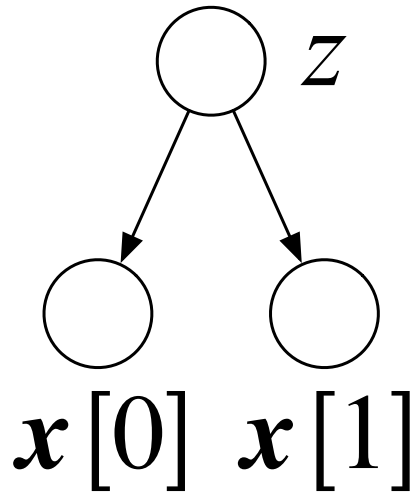
A Graphical View

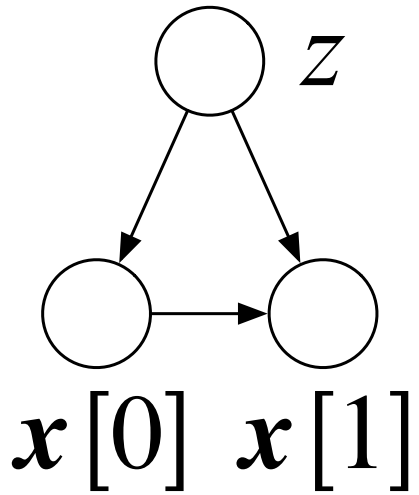


Path is blocked by shaded $\mathbf{x}[1]$.

Generic algorithm called Bayes ball.

A Graphical View





Graph describes factorization of the joint

$$p(z, \mathbf{x}[0], \mathbf{x}[1]) = p(z)p(\mathbf{x}[0] \mid z)p(\mathbf{x}[1] \mid z, \mathbf{x}[0])$$

No assumptions means latent structure is unneeded for data

Mixture of Gaussians

$$z_i \sim \text{Categorical}(1...K),$$
$$\mathbf{x}_i \,|\, z_i = k \sim \text{Normal}(\mu_k, \Sigma_k)$$

is okay, but mixture of arbitrary

$$z_i \sim \text{Categorical}(1...K),$$
$$\mathbf{x}_i \,|\, z_i = k \sim p_{\theta_k}(\mathbf{x}_i)$$

is not. Why?

How can we break this down?

- Is a mixture of Gaussian a Gaussian?

How can we break this down?

- Is a mixture of Gaussian a Gaussian?
- What is a mixture of arbitrary distributions?

How can we break this down?

- Is a mixture of Gaussian a Gaussian?
- What is a mixture of arbitrary distributions?
- Gaussian mixture is useful to capture “non-Gaussian” parts
- Arbitrary mixture is useful to capture “non-arbitrary” parts. What’s that?

How do we evaluate?

Same way as supervised learning. Compute

$$\log p(\mathbf{x})$$

on a held-out set of data. Why?

How do we evaluate?

Same way as supervised learning. Compute

$$\log p(\mathbf{x})$$

on a held-out set of data. Why?

$$\sum_{i=1}^{N} \log p(\mathbf{x}_i), \quad \mathbf{x}_i \sim p$$

Provides an estimate of negative KL divergence (up to a constant)

Breakout

What would be a good latent variable model where there's an underlying state that's responsible for dependence in a sequence of noisy measurements?

Formal definition

Probabilistic latent variable models:

- Data: $\mathbf{x}$
- Hidden Structure (latent variables): $\mathbf{z}$
- Model: $p(\mathbf{x}, \mathbf{z}) = p(\mathbf{z})p(\mathbf{x}|\mathbf{z})$ - prior $p(\mathbf{z})$ and likelihood $p(\mathbf{x}|\mathbf{z})$
- Posterior: $p(\mathbf{z}|\mathbf{x})$ - probability of the hidden structure in documents

Finding Topics in Documents

Data

- N number of documents
- V number of words
- W: $N \times V$ matrix of words

Hidden Structure

- A group of topics that describe the document
- Each document contains a distribution over topics

Breakout

What would be a latent variable model to uncover topics?

Data

- N number of documents
- V number of words
- W: A $N \times V$ matrix of words

How do we describe the topics?

Data

- N number of documents
- V number of words
- W: A $N \times V$ matrix of words

How do we describe the topics?

A distribution over the words called $\boldsymbol{\beta}_k$

- N number of documents
- V number of words
- W: A $N \times V$ matrix of words

How do we describe the documents composition over topics?

A distribution over the topics called $\boldsymbol{\theta}_i$

Have two distributions

- $\boldsymbol{\beta}_k$ distributions over words for each topic
- $\boldsymbol{\theta}_i$ distribution over topics for each documents

Priors?

Have two distributions

- $\boldsymbol{\beta}_k$ distributions over words for each topic
- $\boldsymbol{\theta}_i$ distribution over topics for each documents

Priors? Dirichlet distribution

Still need a likelihood for data

- N number of documents
- V number of words
- W: A $N \times V$ matrix of words

with hidden structure

- $\boldsymbol{\beta}_k$ distributions over words for each topic
- $\boldsymbol{\theta}_i$ distribution over topics for each documents

Assume all documents have same length?

For word m in document l

1. Draw word's topic from $z_{m,l} \sim \text{Categorical}(\boldsymbol{\theta}_i)$
2. Draw topic from for $w_{m,l} \sim \text{Categorical}(\boldsymbol{\beta}_{z_{m,l}})$

Topic Model

For each topic:

1. Draw distribution over words from Dirichlet(α)

For each document:

1. Draw distribution over topics from Dirichlet(κ)

For word m in document l:

1. Draw word's topic from $z_{m,l} \sim \text{Categorical}(\boldsymbol{\theta}_i)$
2. Draw word from topic: $w_{m,l} \sim \text{Categorical}(\boldsymbol{\beta}_{z_{m,l}})$

Topic Model

For each topic:

1. Draw distribution over words from Dirichlet(α)

For each document:

1. Draw distribution over topics from Dirichlet(κ)

For word m in document l:

1. Draw word's topic from $z_{m,l} \sim \text{Categorical}(\boldsymbol{\theta}_i)$
2. Draw word from topic: $w_{m,l} \sim \text{Categorical}(\boldsymbol{\beta}_{z_{m,l}})$

Compute posterior

$$p(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{z} \,|\, \mathbf{w}) = \frac{p(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{z}, \mathbf{w})}{p(\mathbf{w})}$$

The New York Times

music band songs rock album jazz pop song singer night	book life novel story books man stories love children family	art museum show exhibition artist artists paintings painting century works	game knicks nets points team season play games night coach	show film television movie series says life man character know
theater play production show stage street broadway director musical directed	clinton bush campaign gore political republican dole presidential senator house	stock market percent fund investors funds companies stocks investment trading	restaurant sauce menu food dishes street dining dinner chicken served	budget tax governor county mayor billion taxes plan legislature fiscal

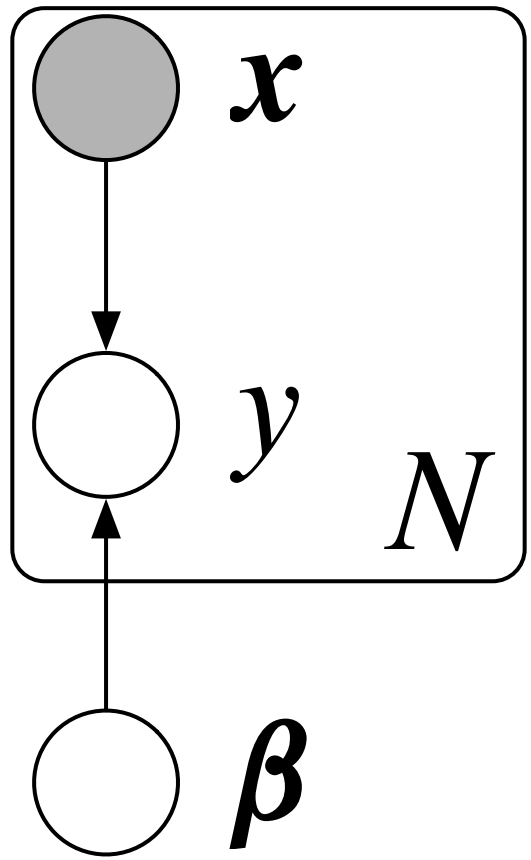
Nature

dna sequence gene sequences rna fragment cdna mrna genes fragments	channel channels receptor voltage currents membrane binding receptors neurons activation	visual stimulus subjects motion target stimuli trials response neurons spatial	ray emission pulsar radio radiation star sources stars neutron_star pulsars	glucose liver enzyme tissue phosphate rats fraction incorporation synthesis mgm
war social industrial policy economic planning men service management labour	stars star disk solar galaxy formation galaxies galactic massive objects	stars observatory the_sun star comet eclipse solar magnitude photographs planet	tube wire glass apparatus force heat instrument electric you iron	virus hiv infection disease infected aids vaccine viruses viral host

[Hoffman+ 2013]

What about parameters shared across data?

Things like regression coefficients are always missing:



Bayesian linear/logistic/generalized linear model.

Everything is inferred from the posterior

Computing the posterior. Is it easy?