CPSC 440: Advanced Machine Learning

Topic Models

Mark Schmidt

University of British Columbia

Winter 2021

Last Time: Empirical Bayes and Hierarchical Bayes

- In Bayesian statistics we make decisions usingintegrals over parameters,

$$p(\text{something}) = \int_\theta (\text{something else, usually weighted by posterior}) d\theta$$

- We discussed empirical Bayes, where you optimize prior using marginal likelihood,

$$\underset{\alpha,\beta}{\text{argmax}}\, p(x \mid \alpha, \beta) = \underset{\alpha,\beta}{\text{argmax}} \int_\theta p(x \mid \theta) p(\theta \mid \alpha, \beta) d\theta.$$

  - Can be used to optimize λ_j, polynomial degree, RBF σ_i, polynomial vs. RBF, etc.
- We also considered hierarchical Bayes, where you put a prior on the prior,

$$p(\alpha, \beta \mid x, \gamma) = \frac{p(x \mid \alpha, \beta) p(\alpha, \beta \mid \gamma)}{p(x \mid \gamma)}.$$

  - Further protection against overfitting, and common model of non-IID data (today).

Hierarchical Bayes as a Graphical Model

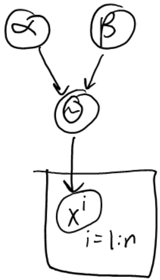
- Let x^i be a binary variable, representing if treatment works on patient i,

$$x^i \sim \mathsf{Ber}(\theta).$$

- As before, let's assume that θ comes from a beta distribution,

$$\theta \sim \mathcal{B}(\alpha, \beta).$$

- We can visualize this as a graphical model:

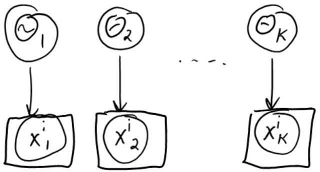


Hierarchical Bayes for Non-IID Data

- Now let x^i represent if treatment works on patient i in hospital j.
- Let's assume that treatment depends on hospital,

$$x^i_j \sim \text{Ber}(\theta_j).$$

- So the x^i_j are only IID given the hospital.



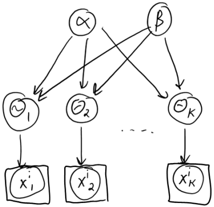
- Problem: we may not have a lot of data for each hospital.
  - Can we use data from one hospital to learn about others?
  - Can we say anything about a hospital with no data?

Hierarchical Bayes for Non-IID Data

- Common approach: assume the θ_j are drawn from common prior,

$$\theta_j \sim \mathcal{B}(\alpha, \beta).$$

- This introduces dependency between parameters at different hospitals:

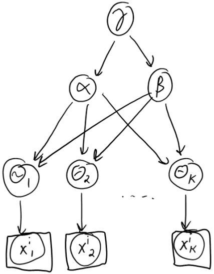


- But, if you fix α and β then you can't learn across hospitals:
  - The θ_j and d-separated given α and β.

- We actually want to learn about α and β.
  - For example, we could do Type II MLE and optimize α and β.

Hierarchical Bayes for Non-IID Data

- Or we could treat α and β as nuissance variables and use a hyperprior:



- Now there is a dependency between the different θ_j (for unknown α and β).

- Now you can combine the non-IID data across different hospitals.
  - Data-rich hospitals inform posterior for data-poor hospitals.
  - You even consider the posterior for new hospitals with no data.

Outline

Motivation for Topic Models

We want a model of the "factors" making up a set of documents.

- In this context, latent-factor models are called topic models.

Suppose you have the following set of sentences:

- I like to eat broccoli and bananas.
- I ate a banana and spinach smoothie for breakfast.
- Chinchillas and kittens are cute.
- My sister adopted a kitten yesterday.
- Look at this cute hamster munching on a piece of broccoli.

What is latent Dirichlet allocation? It's a way of automatically discovering **topics** that these sentences contain. For example, given these sentences and asked for 2 topics, LDA might produce something like

- **Sentences 1 and 2**: 100% Topic A
- **Sentences 3 and 4**: 100% Topic B
- **Sentence 5**: 60% Topic A, 40% Topic B
- **Topic A**: 30% broccoli, 15% bananas, 10% breakfast, 10% munching, … (at which point, you could interpret topic A to be about food)
- **Topic B**: 20% chinchillas, 20% kittens, 20% cute, 15% hamster, … (at which point, you could interpret topic B to be about cute animals)

http://blog.echen.me/2011/08/22/introduction-to-latent-dirichlet-allocation

- "Topics" could be useful for things like searching for relevant documents.

Classic Approach: Latent Semantic Indexing

- Classic methods are based on scores like TF-IDF:
  1. Term frequency: probability of a word occuring within a document.
     - E.g., 7% of words in document i are "the" and 2% of the words are "LeBron".
  2. Document frequency: probability of a word occuring across documents.
     - E.g., 100% of documents contain "the" and 0.01% have "LeBron".
  3. TF-IDF: measures like (term frequency)*log 1/(document frequency).
     - Seeing "LeBron" tells you a lot about document, seeing 'the" tells you nothing.

- Many many many variations exist.

- TF-IDF features are very redundant.
  - Consider TF-IDF of "LeBron", "Durant", and "Giannis".
  - High values of these typically just indicate topic of "basketball".
  - Basically a weighted bag of words.

- We want to find latent factors ("topics") like "basketball".

Modern Approach: Latent Dirichlet Allocation

- Latent semantic indexing (LSI) topic model:
  1. Summarize each document by its TF-IDF values.
  2. Run a latent-factor model like PCA or NMF on the matrix.
  3. Treat the latent factors as the "topics".

- LSI has largely been replace by latent Dirichlet allocation (LDA).
  - Hierarchical Bayesian model of all words in a document.
    - Still ignores word order.
    - Tries to explain all words in terms of topics.

- The most cited ML paper in the 00s?

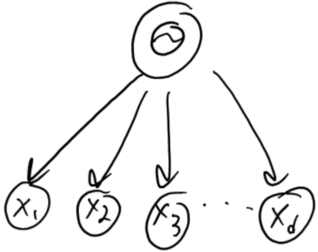
- LDA has several components, we'll build up to it by parts.
  - We'll assume all documents have d words and word order doesn't matter.

Model 1: Categorical Distribution of Words

- Base model: each word x_j comes from the same categorical distribution.

$$p(x_j = \text{“the”}) = \theta_{\text{“the”}} \quad \text{where} \quad \theta_{\text{word}} \geq 0 \quad \text{and} \quad \sum_{\text{word}} \theta_{\text{word}} = 1.$$

- So to generate a document with d words:
  - Sample d words from the categorical distribution.

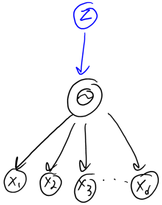


- Drawback: misses that documents are about different “topics”.
  - We want the word distribution to depend on the “topics”.

Model 2: Mixture of Categorical Distributions

- To represent "topics", we'll use a mixture model.
  - Each mixture has its own categorical distribution over words.
    - E.g., the "basketball" mixture will have higher probability of "LeBron".

- So to generate a document with d words:
  - Sample a topic z from a categorical distribution.
  - Sample d words from categorical distribution z.

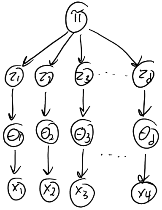


- Similar to a mixture of independent categorical distributions.
  - But we tie categorical distribution across the d variables, given cluster.
- Drawback: misses that documents may be about more than one topics.

Model 3: Multi-Topic Mixture of Categorical

- Our third model introduces a new vector of "topic proportions" π.
  - Gives percentage of each topic that makes up the document.
    - E.g., 80% basketball and 20% politics.
  - Called probabilistic latent semantic indexing (PLSI).

- So to generate a document with d words given topic proportions π:
  - Sample d topics z_j from categorical distribution π.
  - Sample a word for each z_j from corresponding categorical distribution.

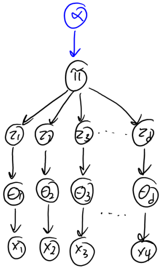


- Similar to HMM where each "time" has own cluster (but no Markov assumption).
- LDA can be viewed as a Bayesian version of this model (adds prior on π).

Model 4: Latent Dirichlet Allocation

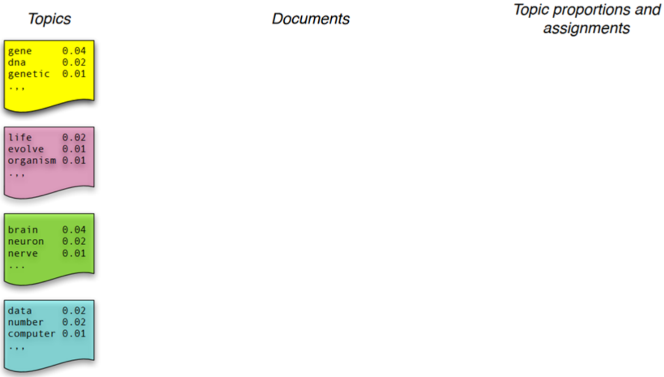
- Latent Dirichlet allocation (LDA) puts a prior on topic proportions.
  - Conjugate prior for categorical is Dirichlet distribution.

- So to generate a document with d words given Dirichlet prior:
  - Sample mixture proportions π from the Dirichlet prior.
  - Sample d topics z_j from categorical distribution π.
  - Sample a word for each z_j from corresponding categorical distribution.



- This is the generative model, typically fit with MCMC or variational methods.

Latent Dirichlet Allocation (LDA)



Each topic is like a "principal component" or "latent factor"

http://menome.com/wp/wp-content/uploads/2014/12/Blei2011.pdf

Latent Dirichlet Allocation (LDA)

1. Sample topic proportions θ from Dirichlet.

Each topic is like a "principal component" or "latent factor"

http://menome.com/wp/wp-content/uploads/2014/12/Blei2011.pdf

Latent Dirichlet Allocation (LDA)

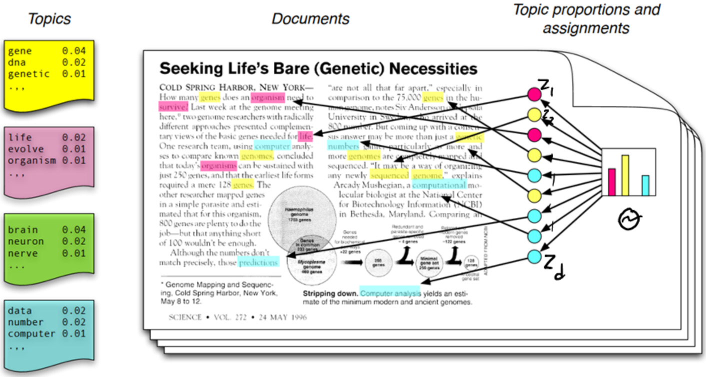
1. Sample topic proportions Θ from Dirichlet.
2. Sample 'd' topics z_j from Θ.

Each topic is like a "principal component" or "latent factor"

http://menome.com/wp/wp-content/uploads/2014/12/Blei2011.pdf

Latent Dirichlet Allocation (LDA)

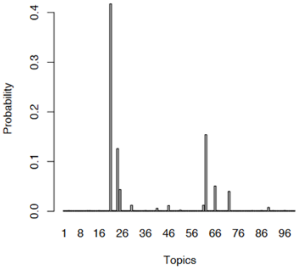
1. Sample topic proportions θ from Dirichlet.
2. Sample 'd' topics z_j from θ.
3. For each z_j sample a word based on frequencies for topic.



Each topic is like a "principal component" or "latent factor"

http://menome.com/wp/wp-content/uploads/2014/12/Blei2011.pdf

Latent Dirichlet Allocation Example



"Genetics"	"Evolution"	"Disease"	"Computers"
human	evolution	disease	computer
genome	evolutionary	host	models
dna	species	bacteria	information
genetic	organisms	diseases	data
genes	life	resistance	computers
sequence	origin	bacterial	system
gene	biology	new	network
molecular	groups	strains	systems
sequencing	phylogenetic	control	model
map	living	infectious	parallel
information	diversity	malaria	methods
genetics	group	parasite	networks
mapping	new	parasites	software
project	two	united	new
sequences	common	tuberculosis	simulations

Figure 2: **Real inference with LDA.** We fit a 100-topic LDA model to 17,000 articles from the journal *Science*. At left is the inferred topic proportions for the example article in Figure 1. At right are the top 15 most frequent words from the most frequent topics found in this article.

http://menome.com/wp/wp-content/uploads/2014/12/Blei2011.pdf

Latent Dirichlet Allocation Example

Figure 3: A topic model fit to the *Yale Law Journal*. Here there are twenty topics (the top eight are plotted). Each topic is illustrated with its top most frequent words. Each word's position along the x-axis denotes its specificity to the documents. For example "estate" in the first topic is more specific than "tax."

http://menome.com/wp/wp-content/uploads/2014/12/Blei2011.pdf

Latent Dirichlet Allocation Example

Health topics in social media:

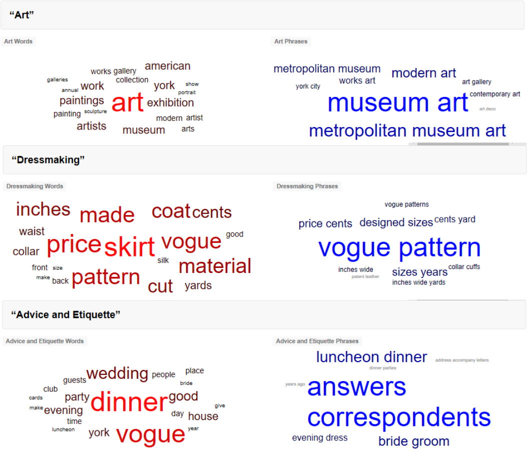
Non-Ailment Topics						
TV & Movies	**Games & Sports**	**School**	**Conversation**	**Family**	**Transportation**	**Music**
watch watching tv killing movie seen movies mr watched hi	killing play game playing win boys games fight lost team	ugh class school read test doing finish reading teacher write	ill ok haha ha fine yeah thanks hey thats xd	mom shes dad says hes sister tell mum brother thinks	home car drive walk bus driving trip ride leave house	voice hear feelin lil night bit music listening listen sound

Ailments						
	Influenza-like Illness	**Insomnia & Sleep Issues**	**Diet & Exercise**	**Cancer & Serious Illness**	**Injuries & Pain**	**Dental Health**
General Words	better hope ill soon feel feeling day flu thanks xx	night bed body ill tired work day hours asleep morning	body pounds gym weight lost workout lose days legs week	cancer help pray awareness diagnosed prayers died family friend shes	hurts knee ankle hurt neck ouch leg arm fell left	dentist appointment doctors tooth teeth appt wisdom eye going went
Symptoms	sick sore throat fever cough	sleep headache fall insomnia sleeping	sore throat pain aching stomach	cancer breast lung prostate sad	pain sore head foot feet	infection pain mouth ear sinus
Treatments	hospital surgery antibiotics fluids paracetamol	sleeping pills caffeine pill tylenol	exercise diet dieting exercises protein	surgery hospital treatment heart transplant	massage brace physical therapy crutches	surgery braces antibiotics eye hospital

http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0103408

Latent Dirichlet Allocation Example

Three topics in 100 years of “Vogue” fashion magazine:



http://dh.library.yale.edu/projects/vogue/topics/

Discussion of Topic Models

- There are *many* extensions of LDA:
  - We can put prior on the number of words (like Poisson).
  - Correlated and hierarchical topic models learn dependencies between topics.

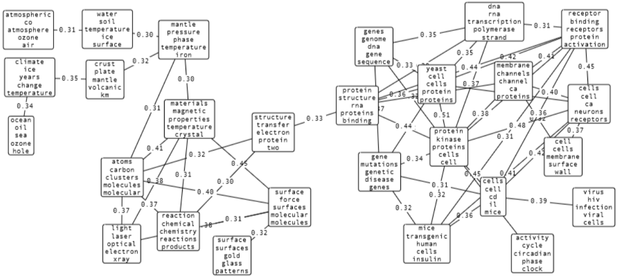
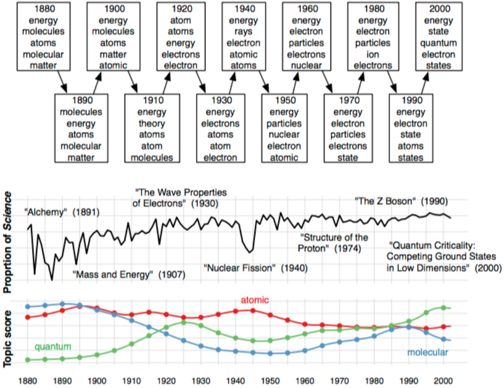


Figure 2: A portion of the topic graph learned from 15,744 OCR articles from *Science*. Each node represents a topic, and is labeled with the five most probable words from its distribution; edges are labeled with the correlation between topics.

http://people.ee.duke.edu/~lcarin/Blei2005CTM.pdf

Discussion of Topic Models

- There are *many* extensions of LDA:
  - We can put prior on the number of words (like Poisson).
  - Correlated and hierarchical topic models learn dependencies between topics.
  - Can be combined with Markov models to capture dependencies over time.



Discussion of Topic Models

- There are *many* extensions of LDA:
  - We can put prior on the number of words (like Poisson).
  - Correlated and hierarchical topic models learn dependencies between topics.
  - Can be combined with Markov models to capture dependencies over time.
  - Recent work on better word representations like "word2vec" (CPSC 340).
  - Now being applied beyond text, like "cancer mutation signatures":

http://journals.plos.org/plosgenetics/article?id=10.1371/journal.pgen.1005657

Discussion of Topic Models

- Topic models for analyzing musical keys:

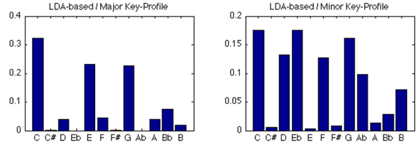


Figure 2: The C major and C minor key-profiles learned by our model, as encoded by the β matrix. Resulting key-profiles are obtained by transposition.

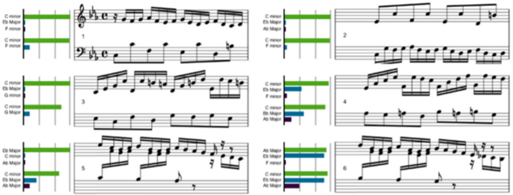


Figure 3: Key judgments for the first 6 measures of Bach's Prelude in C minor, WTC-II. Annotations for each measure show the top three keys (and relative strengths) chosen for each measure. The top set of three annotations are judgments from our LDA-based model; the bottom set of three are from human expert judgments [3].

Monte Carlo Methods for Topic Models

- Nasty integrals in topic models:

Inference [edit]

See also: Dirichlet-multinomial distribution

Learning the various distributions (the set of topics, their associated word probabilities, the topic of each word, and the particular topic mixture of each document) is a problem of Bayesian inference. The original paper used a variational Bayes approximation of the posterior distribution;[1] alternative inference techniques use Gibbs sampling[6] and expectation propagation.[7]

Following is the derivation of the equations for collapsed Gibbs sampling, which means φs and θs will be integrated out. For simplicity, in this derivation the documents are all assumed to have the same length N. The derivation is equally valid if the document lengths vary.

According to the model, the total probability of the model is:

$$P(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\theta}, \boldsymbol{\varphi}; \alpha, \beta) = \prod_{i=1}^{K} P(\varphi_i; \beta) \prod_{j=1}^{M} P(\theta_j; \alpha) \prod_{t=1}^{N} P(Z_{j,t} | \theta_j) P(W_{j,t} | \varphi_{Z_{j,t}}),$$

where the bold-font variables denote the vector version of the variables. First, $\boldsymbol{\varphi}$ and $\boldsymbol{\theta}$ need to be integrated out.

$$\begin{aligned} P(\boldsymbol{Z}, \boldsymbol{W}; \alpha, \beta) &= \int_{\boldsymbol{\theta}} \int_{\boldsymbol{\varphi}} P(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\theta}, \boldsymbol{\varphi}; \alpha, \beta) \, d\boldsymbol{\varphi} \, d\boldsymbol{\theta} \\ &= \int_{\boldsymbol{\varphi}} \prod_{i=1}^{K} P(\varphi_i; \beta) \prod_{j=1}^{M} \prod_{t=1}^{N} P(W_{j,t} \mid \varphi_{Z_{j,t}}) \, d\boldsymbol{\varphi} \int_{\boldsymbol{\theta}} \prod_{j=1}^{M} P(\theta_j; \alpha) \prod_{t=1}^{N} P(Z_{j,t} \mid \theta_j) \, d\boldsymbol{\theta}. \end{aligned}$$

https://en.wikipedia.org/wiki/Latent_Dirichlet_allocation

Monte Carlo Methods for Topic Models

- How do we actually *use* Monte Carlo for topic models?
- First we write out the posterior:

$$p(\underset{\text{topics}}{Z}, \underset{\text{word prob.}}{\Pi}, \overset{\text{topic prop.}}{\Theta} \mid \underset{\text{data (words)}}{X}, \underset{\text{prior on topic proportions}}{\alpha}, \underset{\text{prior on word probabilities}}{\beta}) = \left[\prod_{i=1}^{n} \underbrace{p(\theta^i \mid \alpha)}_{\substack{\text{topic proportion} \\ \text{probability} \\ \text{(document 'i')}}} \prod_{j=1}^{d} \underbrace{p(z_j^i \mid \theta^i)}_{\substack{\text{topic probability} \\ \text{(topic at position} \\ \text{'j' in document 'i')}}} \underbrace{p(x_j^i \mid z_j^i, \pi_j)}_{\substack{\text{word probability} \\ \text{(word at position} \\ \text{'j' in document 'i')}}}\right]\left[\prod_{c=1}^{k} \underbrace{p(\pi_c \mid \beta)}_{\substack{\text{word probability} \\ \text{parameters} \\ \text{(topic 'c')}}}\right]$$

Monte Carlo Methods for Topic Models

- How do we actually *use* Monte Carlo for topic models?
- Next we generate samples from the posterior:
  - With Gibbs sampling we alternate between:
    - Sampling topics given word probabilities and topic proportions.
    - Sampling topic proportions given topics and prior parameters α.
    - Sampling word probabilities given topics, words, and prior parameters β.
  - Have a burn-in period, use thinning, try to monitor convergence, etc.
- Finally, we use posterior samples to do inference:
  - Distribution of topic proportions for sample i is frequency in samples.
  - To see if words come from same topic, check frequency in samples.

Summary

- Relaxing IID assumption with hierarchical Bayes.
- Topic models: latent-factor model of discrete data text.
  - The latent "factors" are called "topics".
- Latent Dirichlet allocation: hierarchical Bayesian topic model.
  - Represent words in documents as coming from different topics.
  - Each document has its own proportion for each topic.
- Next time: we start talking about more-fancy sampling methods.