CS 181 Spring 2021 Section 9
Topic Models, Factor Analysis, and PCA

1 Topic Models

Topic modeling is a form of mixture modeling commonly used in analyzing text corpora. As this is a form of unsupervised learning, the goal is to extract a set of parameters from our data that we can use to make inferences about it. In particular, in topic modeling, we are interested in learning the topics that exist in a corpus - we will define this concept more formally.

Like mixture models that we study in this course, we will describe topic modeling as a generative model: one in which we assume our corpus is generated by some process. Then, we will seek to use an estimation technique, like EM, to determine the parameters of this model, and we can then interpret these parameters to understand the corpus.

We begin by describing the generative process by which a document i is generated. For topic modeling, similar to K-means, we have to begin by picking the number of topics, K, to look for. We define a topic $\boldsymbol{\phi}_k$ to be a distribution over the words, so $\boldsymbol{\phi}_k \in [0,1]^{|\mathcal{W}|}$, where $\mathcal{W}$ is the set of words. For each document, we have a document-topic distribution $\boldsymbol{\theta}_m \in [0,1]^K$. This is one of the parameters we will find by estimation.

We now describe the data generation process:

1. Let $\alpha \in \mathbb{R}_+^K$ and $\beta \in \mathbb{R}_+^{|\mathcal{W}|}$.

2. For each document $m = 1, \ldots, M$, sample a mixture over topics: $\boldsymbol{\theta}_n \sim Dir(\alpha)$.

3. For each topic $k = 1, \ldots, K$, sample a mixture over words in that topic: $\boldsymbol{\phi}_k \sim Dir(\beta)$.

4. For each word $w_{m,n}$ (for $m = 1, \ldots, M$ and $n = 1, \ldots, N$, the length of the document), first sample the topic $z_{m,n} \sim Cat(\boldsymbol{\theta}_m)$, then sample the word $w_{m,n} \sim Cat(\boldsymbol{\phi}_{z_{m,n}})$.

At a high level, a topic model is a mixture over mixtures: within a single document, D_m, $\boldsymbol{\theta}_m$ specifies a distribution over topics in that document, and for each topic, k, in that document, $\boldsymbol{\phi}_k$ specifies a distribution over words.

This process is summarized in the following plate diagram:
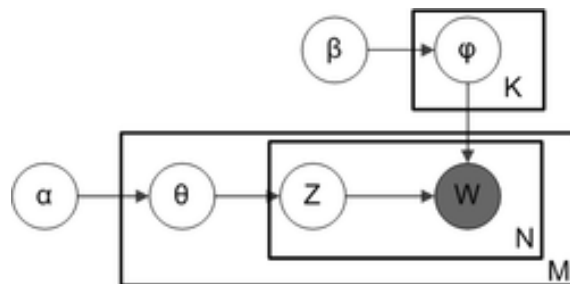
Exercise 1. *In class, we discussed the EM algorithm applied to topic models. Describe, at a high level, how the EM algorithm for topic models can be viewed as alternating between two optimizations. What are these two optimizations?*

2   Factor Analysis

In factor analysis, we have the following generative process (where $D > K$ below):

1. $\mathbf{A}_{d,k} \overset{i.i.d}{\sim} \mathcal{N}(0,1)$ for $d = 1, \ldots, D, k = 1, \ldots, K$

2. For each n:

   - $\mathbf{z_n} \sim \mathcal{N}(0, \mathbf{I}_K)$
   - $\mathbf{x_n} \sim \mathcal{N}(\mathbf{A}z_n, \sigma^2 \mathbf{I}_D)$ for some variance $\sigma^2 > 0$

In particular, the matrix $\mathbf{A}$ and the variables $\mathbf{z_n}$ are *not* observable.

Estimating $\mathbf{A}$ (without access to the $\mathbf{z}$'s) and/or the distribution of the $\mathbf{z}$'s (without access to $\mathbf{A}$) seems like a very difficult problem! However, suppose that we could see the latent variables $\mathbf{z_n}$. Then finding $\mathbf{A}$ reduces to a linear regression problem per coordinate of $\mathbf{x_n}$ (which we know how to solve). Alternatively, noting that $p(\mathbf{z_n}|\mathbf{x_n}, \mathbf{A}) \propto p(\mathbf{x_n}|\mathbf{z_n}, \mathbf{A}) \cdot p(\mathbf{z_n})$, if we could see the vector $\mathbf{A}$, then we can write down the PDFs $p(\mathbf{x_n}|\mathbf{z_n}, \mathbf{A})$ and $p(\mathbf{z_n})$ and thus can find $p(\mathbf{z_n}|\mathbf{x_n}, \mathbf{A})$ (i.e., since we have its unnormalized PDF and everything is Gaussian). We can then solve for $\mathbf{z_n}$ by returning the mode of this PDF.

> **Exercise 2.** *In the real-world setting where we don't have access to the latent variables or $\mathbf{A}$, describe how we can use the above ideas to alternate between optimizing $\mathbf{A}$ give the $\mathbf{z}$'s and optimizing the $\mathbf{z}$'s given $\mathbf{A}$. In particular, propose a Max-Max algorithm to do so.*

3 Principal Component Analysis

3.1 Motivation

In many supervised learning problems, we try to find rich features that increase the expressivity of our model. In practice, this often involves using basis functions to transform model input into a higher dimensional space (eg. given data x, using x and x^2 as features, or using features learned by a neural network).

However, sometimes we want to reduce the dimensionality of our data.

Exercise 3. *Why would we want to reduce the dimensionality of our data? Can you think of example cases?*

One method for dimensionality reduction through **linear projections** of the original data is PCA. When reducing the dimensionality of our data from m to d where $d < m$, PCA can be interpreted as minimizing the reconstruction loss of projecting data onto d basis vectors, or as maximizing the variance in data that can be explained by d basis vectors.

3.2 Finding the lower dimensional representation

To perform PCA, or project each data point $\mathbf{x}$: $(m \times 1)$ to $\mathbf{z}$: $(d \times 1)$,

1. Center the data by subtracting the mean of each feature from each data point. Steps 2 - 5 will be performed on the centered data $\mathbf{X}$: $(n \times m)$.

2. Calculate the **empirical covariance** matrix:

$$\mathbf{S} = \frac{1}{n}(\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^\top) = \frac{1}{n}\mathbf{X}^\top \mathbf{X}$$

3. Decide how many dimensions d out of the original m that we want to keep in the final representation. For visualizations, often this will be $d = 2$ or $d = 3$.

4. Find the d largest eigenvalues of $\mathbf{S}$. The $m \times 1$ eigenvectors $(\mathbf{u}_1, \ldots, \mathbf{u}_d)$ corresponding to these eigenvalues will be our lower-dimensional basis.

5. Thus, we reduce the dimensionality of a data point $\mathbf{x}$ by projecting it onto this basis - we combine the eigenvectors into the $m \times d$ matrix $\mathbf{U}$, and compute

$$\langle \mathbf{u}_1^\top \mathbf{x}, \mathbf{u}_2^\top \mathbf{x}, \ldots, \mathbf{u}_d^\top \mathbf{x}, 0 \rangle = \mathbf{U}^\top \mathbf{x} = \mathbf{z}$$

$\mathbf{z}$ is called the reconstruction coefficients where $\mathbf{U}\mathbf{z}$ is the reconstruction of $\mathbf{x}$.

Exercise 4. *You are given the following data set:*

$$\mathbf{x}_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \quad \mathbf{x}_2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad \mathbf{x}_3 = \begin{bmatrix} -2 \\ -1 \end{bmatrix}$$

You would like to use PCA to find a 1-dimensional representation of the data.

1. *Plot the data set.*

2. *Compute the empirical covariance matrix $\mathbf{S}$.*

3. *You find that $\mathbf{S}$ has eigenvector $\begin{bmatrix} -1 & 1 \end{bmatrix}^{\top}$ with eigenvalue 1 and eigenvector $\begin{bmatrix} 1 & 1 \end{bmatrix}^{\top}$ with eigenvalue 3. What is the (normalized) basis vector $\mathbf{u}_1$ of your 1-dimensional representation? Add the basis vector $\mathbf{u}_1$ to your plot.*

4. *Compute the coefficients z_1, z_2, z_3. Add the lower-dimensional representations $z_1\mathbf{u}_1, z_2\mathbf{u}_1, z_3\mathbf{u}_1$ to your plot. Based on your plot, what is the relationship between $z_i\mathbf{u}_1$ and $\mathbf{x}_i$ with respect to the new basis?*

5. *Based on your plot, what would happen if you chose the unused eigenvector to be your basis vector?*

Exercise 5. *Suppose that our data are centered (i.e., have sample mean 0). Recall that in lecture, we showed that, when optimizing over (semi)-orthogonal matrices $\mathbf{U} \in \mathbb{R}^{m \times d}$ (i.e., where $\mathbf{U}^T\mathbf{U} = I$) to minimize the reconstruction loss,*

$$\mathcal{L}(\mathbf{U}) = \frac{1}{n} \sum_{i=1}^{n} ||\mathbf{x_i} - \mathbf{U}\mathbf{U}^T\mathbf{x_i}||_2^2,$$

we found that $\mathbf{U}_d$, the matrix whose first d columns are (in order) the top d eigenvectors of the empirical covariance matrix $\mathbf{\Sigma} = \frac{1}{n}\mathbf{X}^T\mathbf{X}$, will achieve the minimum (i.e., $\mathbf{z} = \mathbf{U}^T\mathbf{x}$ is the projection of $\mathbf{x}$ into d dimensions, and $\mathbf{U}\mathbf{z}$ is its reconstruction in $\mathbb{R}^m$). In class, we showed this for case when $d = m - 1$ by using Lagrange multipliers. Show this, in general, for d.

Hint: you may use the following theorem:

Theorem 1 (Courant-Fischer)**.** *Let $\mathbf{A}$ be a symmetric $n \times n$ matrix with eigenvalues $\lambda_1 \leq \cdots \leq \lambda_n$ and corresponding eigenvectors $\mathbf{v_1}, \ldots, \mathbf{v_n}$. Then*

$$\lambda_1 = \min_{||x||=1} \mathbf{x}^T\mathbf{A}\mathbf{x}$$

$$\lambda_2 = \min_{||\mathbf{x}||=1, \mathbf{x} \perp \mathbf{v_1}} \mathbf{x}^T\mathbf{A}\mathbf{x}$$

$$\vdots$$

$$\lambda_i = \min_{||\mathbf{x}||=1, \mathbf{x} \perp \mathbf{v_1}, \mathbf{x} \perp \mathbf{v_2}, \ldots, \mathbf{x} \perp \mathbf{v_{i-1}}} \mathbf{x}^T\mathbf{A}\mathbf{x}$$

$$\vdots$$

$$\lambda_n = \min_{||\mathbf{x}||=1, \mathbf{x} \perp \mathbf{v_1}, \mathbf{x} \perp \mathbf{v_2}, \ldots, \mathbf{x} \perp \mathbf{v_{n-1}}} \mathbf{x}^T\mathbf{A}\mathbf{x}.$$