Alin Tomescu, http://people.csail.mit.edu/~alinush

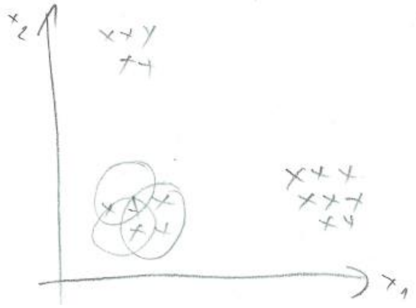
6.867 Machine learning | Prof. Tommi Jaakkola | Week 10, Tuesday, November 5th, 2013| Lecture 17

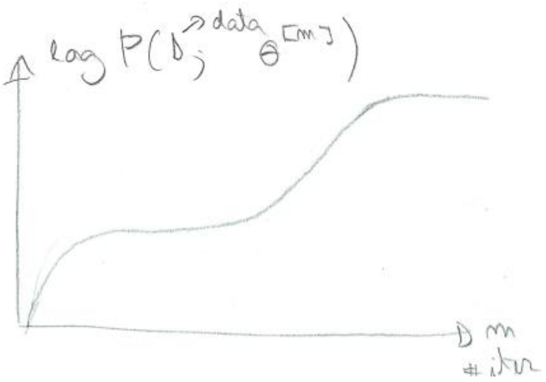
Lecture 17: More Gaussian Mixture Models

Gaussian mixture models

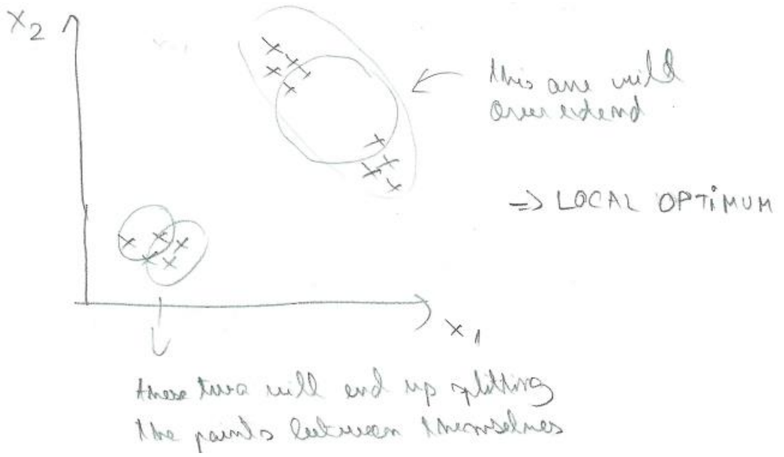
$$P(x;\theta) = \sum_{y=1}^{k} P_y N(x; \mu_y, \Sigma_y)$$

P_y are called the **mixing proportions.**





Understand **importance of initialization** and how it could lead to a local optimum:


x_2
x_1
this one will over extend
=> LOCAL OPTIMUM
these two will end up splitting the points between themselves

How can we **avoid getting a local optimum**?

- Regularize the covariance matrix and don't let the Gaussian get too wide
  - We want all covariance matrices to be the same, so we can regularize by simply enforcing all of the matrices to be $\sigma^2 I$.
- Run the EM algorithm multiple times

Bayesian information criterion (BIC)

How can we figure out **how many clusters** we have in the data set? How do we determine k?

Every time you add a component you are increasing the complexity of your model, so you will fit better and get a higher likelihood.

You have to **balance the increase in the likelihood** you would expect to get by adding a new cluster/Gaussian **with a penalty that penalizes the complexity of the model.**

BIC assigns a score for a model given the data:

$$l\left(D;\hat{\theta}_{ML}\right) = \log \prod_{i-1}^{n} P\left(x^{(i)},\hat{\theta}_{ML}\right)$$

Assuming you found the best solution for that number of clusters. But as you increase k, $l\left(D;\hat{\theta}_{ML}\right)$ will go up.

$$BIC_{\text{score}} = l\left(D;\hat{\theta}_{ML}\right) - \frac{\#\text{ parameters}}{2}\log n$$

$$\text{where } n = \#\text{ of data points}$$

If I add one component (probability distribution, Gaussian, etc.) to the mixture, how much higher than the BIC does $l\left(D;\hat{\theta}_{ML}\right)$ have to become for me to select this new component?

The gap that I need to achieve in terms of likelihood is the number of additional parameters that I'm adding with that $k+1$ component.

Each component has parmeters μ_y $(d \text{ dimensions}), \Sigma_y \left(\frac{d(d+1)}{2}\ dimensions\right), P_y$ $(k \text{ such probabilities, but once we}$ have $k-1$ of them we can compute the last one).

$$\#\text{ param in mixture} = \text{k}\left(d + \frac{d(d+1)}{2} + 1\right) - 1$$

E-step: $q^{[m]}(y|i) = P\left(y|x^{(i)};\theta^{[m]}\right)$

Hard-margin EM

In many cases we can't even enumerate the values of y (might be combinatorially high), so we have a few alternatives for the **E-step**:

This means that we cannot compute $P\left(y|x^{(i)};\theta^{[m]}\right)$ and as a result we cannot compute all the $q^{[m]}(y|i)$.

Give example where you cannot enumerate the values of y:

Example: We want to predict the body state of a person in an image as either "standing" or "sitting." Build a classifier for such a dataset of images.

$$\log P(\text{body state } p \mid \text{image}) = \sum_{\text{index } i \text{ of body part}} \left(\sum_{\text{location } l_i \text{ of body part } i} \log P(l_i \mid \text{image}) \right) \log P(p \mid l_1, \dots, l_n)$$

But the i^{th} body part along with its location and the $(i+1)^{th}$ body part along with its location are not independent of each other, since once you picked an l_i for $i = \text{head}$ the position of the waist, for instance, will be constrained (i.e., it has to probably be below the head, not to the right of it)

In the independent case:

$$P(l_1, l_2, \dots, l_k \mid image) = \prod_i P(l_i \mid \text{image})$$

$$\log P(\text{body state } p \mid \text{image}) = \sum_{\text{index } i \text{ of body part}} \left(\sum_{\text{location } l_i \text{ of body part } i} \log P(l_i \mid \text{image}) \right) \log P(p \mid l_1, \dots, l_n)$$

In the (dependent) tree case:

$$P(l_1, l_2, \dots, l_k \mid image) = \prod_i P\big(l_i \mid l_{\pi(i)}, \text{image}\big), \pi(i) = \text{parent of body part } i$$

$$P(\text{body state } p \mid image) = \sum_{\vec{l} \in \{l_1, \dots, l_k\}} P\big(\vec{l} \mid \text{image}\big) P\big(p \mid \vec{l}, \text{image}\big)$$

And as a result you now have an exponential number of subsets to consider: $\vec{l} \in \{l_1, \dots, l_k\}$

Hard-margin EM

For every data point we do this:

$$\hat{y}^{(i)} = \operatorname*{argmax}_{\text{y}} P\big(y | x^{(i)}; \theta^{[m]}\big)$$

This is easier because it's a maximization problem, not a counting problem like $P\big(y|x^{(i)}; \theta^{[m]}\big)$.

Sample from the posterior

$$\hat{y}^{(i)} \sim P\big(y|x^{(i)}; \theta^{[m]}\big)$$

If you consider the body part example, sampling locations for body parts is much easier than computing the probabilities of all the possible permutations of body parts locations. This is because once you sample a position for the head, the other things fall into place easier, since they are restricted in where they can be.

M-step becomes ML estimation using completed data $\big(x^{(i)}, y^{(i)}\big), i = 1, \dots, n$

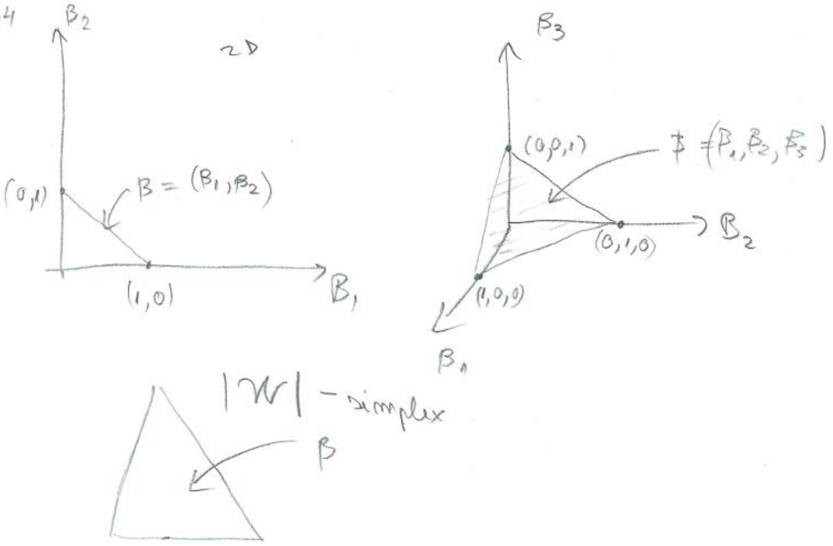
Topic models: models over documents

Modeling **documents** that are just sequences of words $d = \{w_1, \dots, w_n\}$.

The probability of a word would be:

$$P(w; \beta) = \beta_w, \beta_w \geq 0, \sum_{w \in \mathcal{W}} \beta_w = 1$$

FIG4



Probability simplex: triangle in 3D, line in 2D

The probability of a document $d = \{w_1, \dots, w_n\}$ becomes:

$$P(d; \beta) = \prod_{w \in d} P(w) = \prod_{i=1}^{n} \beta_{w_i} = \prod_{w \in \mathcal{W}} \beta_w^{n(w)},$$

where $n(w)$ is the number of times the word w appears in the document d

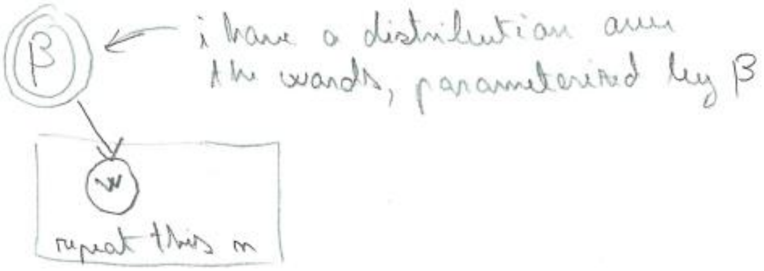
This assumes words are independent of each other, which they are not (for instance "are" is rarely or never followed by another "are").

Document generation model

For $i = 1, \dots, n$ do:

$$w_i \sim Multinomial(\beta_1, \dots, \beta_{|\mathcal{W}|})$$

Graphically this type of model would look as follows:


β
i have a distribution over the words, parameterized by β
w
repeat this n

Let's try to decompose these and introduce topics.

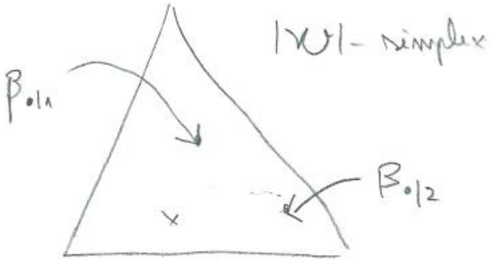
Document topics

Idea: A news article and it can have a certain topic, so there should be a **probability distribution over words for that topic**. We want to define $P(w \mid \text{topic})$.

The topic can be $z = 1, \dots, k$

$$P(w|z;\beta) = \beta_{w|z}, \beta_{w|z} \geq 0,$$

$$\text{for a fixed } z, \sum_{w \in \mathcal{W}} \beta_{w|z} = 1$$



each topic will be a point on the simplex since they are just prob. distr.

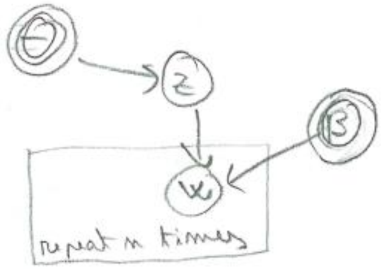
Mixture 1

These are mixing proportions. θ_z is the mixing proportion. θ_z is the distribution over topics in the document d.

$$\theta_z \geq 0, \sum_{z=1}^{k} \theta_z = 1$$

$$P(d;\theta,\beta) = \sum_{z \in \text{topics}} P(z) \prod_{w \in d \text{ with topic z}} P(w|z) = \sum_{z=1}^{k} \theta_z \left(\prod_{i=1}^{n} B_{w_i|z} \right)$$

Again, we are assuming the words are independent and the document has a single topic.



In order to generate the document, I sample z from a multinomial:

$$z \sim Multinomial(\theta_1, \dots, \theta_k)$$

$$\text{For } i = 1, \dots, n \text{ do:}$$

$$w_i \sim Multinomial\left(\beta_{1|z}, \dots, \beta_{|\mathcal{W}|\,|\,z}\right)$$

The assumption here is that **each document has exactly one topic.**

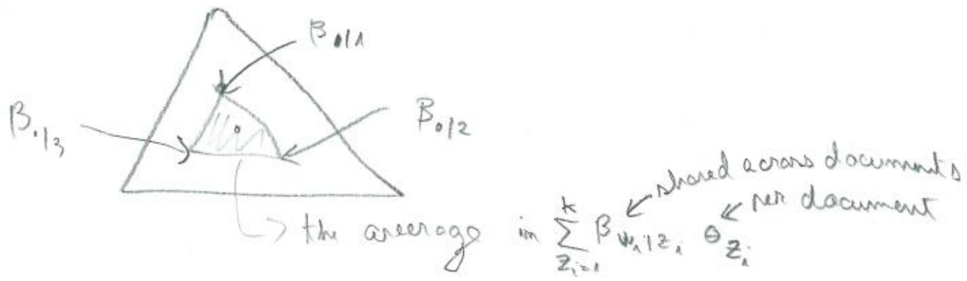
Mixture 2

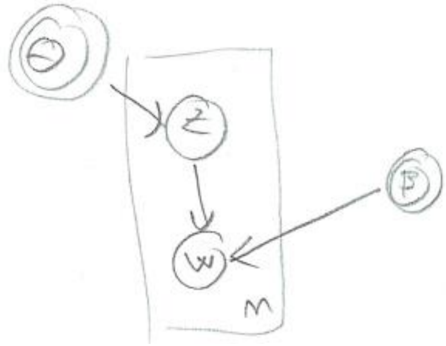
We put a probability distribution on every topic. Now we consider the case where words in the document can have multiple documents.

$$P(d;\theta,\beta) = \prod_{w\in d}\ \sum_{z\in\text{topics (for word }w)} P(w|z)P(z) = \prod_{i=1}^{n}\left(\sum_{z=1}^{k}\beta_{w_i\,|\,z_i}\theta_z\right)$$

$$\theta_z = P(\text{topic } z \text{ in document})$$

$$z_i = \text{word } i \text{ has topic } z$$


β·|1
β·|3
β·|2
the average in $\sum_{z_i=1}^{k} \beta_{w_i|z_i}\ \theta_{z_i}$
shared across documents
per document



For $i = 1, \ldots, n$ do:

$$z_i \sim Multinomial(\theta_1, \ldots, \theta_k)$$

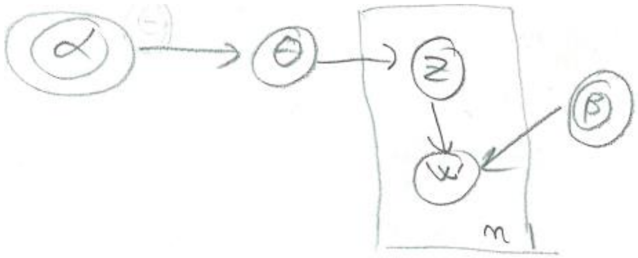
$$w_i = Multinomial\left(\beta_{1|z_i}, \ldots, \beta_{|W|\,|\,z_i}\right)$$

This mixture entertains one θ. Thee next one will entertain all θ's in the simplex.

Mixture 3 (Latent Dirichlet Allocation)

Now we entertain all possible distributions on topics (out of a family of distributions I think).

$$P(d;\alpha,\beta) = \int_{K-simplex} P(\theta;\alpha)\prod_{i=1}^{n}\sum_{z_i=1}^{k}\beta_{w_i\,|\,z_i}\theta_{z_i}\,d\theta$$



How we sample:

$$\theta \sim P(\theta;\alpha)$$

For $i = 1, \ldots, n$ do:

$$z_i \sim Multinomial(\theta_1, \ldots, \theta_k)$$

$$w_i = Multinomial\left(\beta_{1|z_i}, \dots, \beta_{|W|\,|\,z_i}\right)$$

ML estimation:

Given $d^1, \dots, d^T$, maximize:

$$\sum_{t=1}^{T} \log P(d^t; \alpha, \beta)$$

We will see what $P(\theta; \alpha)$ looks like. It will actually be a Dirichlet distribution.