Quantitative Research Methods IV - 17.806

Recitation, Week 10.

Topic: Text Analysis II.

Benjamín Muñoz

April 21, 2023

MIT

Table of contents

1/ PSet Review

PSet 3

- **Estimator:** statistic (function of the sample) used to infer some feature $\gamma(P)$ (estimand or parameter) of an unknown distribution/population P.
- **Sampling Distribution;** probability distribution of an estimator (RV, variation induced by sampling).
- **Finite Sample Properties:** How an estimator performs for a finite number of observations n (statistical properties of the estimator that are valid for any given sample size).
- **Bias:** the difference between this estimator's expected value and the true value of the parameter being estimated ($\text{Bias}(\hat{\theta}) = \mathbb{E}[\hat{\theta}] - \theta$).
- (Sampling) **Variance:** how far, on average, the collection of estimates are from the expected value of the estimates. ($MSE = \mathbb{E}[\text{Error}(\hat{\theta}, \theta)^2] = \mathbb{E}[(\hat{\theta} - \theta)^2] = \mathbb{E}[(\hat{\theta} - \mathbb{E}[\theta])^2] + (\mathbb{E}[\hat{\theta}] - \theta))^2 = \mathbb{V}(\hat{\theta}) + \text{Bias}(\hat{\theta})^2$).

PSet 3

- **Identification of the ATE:**
  1. **SUTVA:** No interference and Consistency $(Y(D_1, D_2, \ldots, D_N)_i = Y(D_1', D_2', \ldots, D_N')_i$ if $D_i = D_i')$.
  2. **Conditional Ignorability:** $Y_i^0, Y_i^1 \perp\!\!\!\perp D_i | X_i = x$, for any $x \in \mathcal{X}$.
  3. **Common Support:** $0 < P(D_i = 1 | X_i = x) < 1$ for any $x \in \mathcal{X}$.
  4. **Constant Treatment Effects** ($\tau_i = \bar{\tau} = Y_i^1 - Y_i^0$ for all i) and **Outcomes are Linear in X** ($Y_i(D) = \alpha + \tau D_i + \gamma^T X + \epsilon_i$).

- **Identification of the CATE:**
  1. Linear interactive effect that changes at a constant rate with the moderator.
  2. Proper functional form: controls have a linear and additive effect (No Omitted Variable Bias, including moderator-covariates interactions).

2/ Expectation-Maximization Alogrithm (E-M)

E-M Algorithm

- Iterative method to find the local maximum likelihood of parameters in statistical models.
- Latent Variables.
- Manifest Variables.

EM: overview

- An algorithm to conduct MLE when latent parameters are involved or some data is missing (recall Quant III!).
- Ideally we want to estimate parameters of interest by evaluating the log-Likelihood function (Score and Hessian).
- Sometimes it is impossible to evaluate it because of missing data or existence of latent variables(in some cases we intentionally include them to make the function tractable).

  $\rightsquigarrow$ introduce the complete log-Likelihood function.
- Since we never observe the complete log-Likehood function, we take the expectation of the latent variable given data and parameters .
- MLE we learned: known likelihood function, only update parameters.

  $\rightsquigarrow$ EM: iteratively update both likelihood function and parameters.

EM: overview

- Expectation of the complete log-Likelihood: $Q(\theta, \theta^{\text{old}})$.

$$
\begin{aligned}
Q(\theta, \theta^{\text{old}}) &= \mathbb{E}Z \mid X, \theta^{\text{old}}[\log p(X, Z \mid \theta) \mid X, \theta^{\text{old}}] \\
&= \int \log p(X, Z \mid \theta) \times f(Z \mid X, \theta^{\text{old}}) dZ \\
(&= \sum_{Z} \log p(X, Z \mid \theta) \times f(Z \mid X, \theta^{\text{old}}) \quad \text{for discrete } Z)
\end{aligned}
$$

- Note that the definition of expectation is $\mathbb{E}[x] = \int x f(x) dx$[1]
- Why $\mathbb{E}Z \mid X, \theta^{\text{old}}[p(X, Z \mid \theta) \mid X, \theta^{\text{old}}]$?

  $\rightsquigarrow$ Use existing information, X and θ^{old} to obtain Z (better than random guess).

[1]In the lecture slide, f is written as $p(Z \mid X, \theta^{\text{old}})$

E-step and M-step

- Expectation (E-step): evaluate the expectation of the complete log-Likelihood.
- "fill in" (or make a guess about) the missing data Z
- Once we get some information about Z, $f(Z \mid X, \theta^{\text{old}})$, we can evaluate Q.

  $\rightsquigarrow$ evaluating Q is equilavent to evaluating $f(Z \mid X, \theta^{\text{old}})$.

- Maximization (M-step): estimate the parameter of interest using the function obtained in E-step.
- Update θ by optimizing Q (MLE).

- E-step: update Q by **updating** Z **given** θ (and X).
- M-step: update θ given Z (and X).
- Repeat these two steps until estimated θ converges.

EM Derivation

- PDF Normal: $\mathcal{N}(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[\frac{-(x-\mu)^2}{2\sigma^2}\right]$
- Mixture: $p(x|\Theta) = \sum_k \alpha_k \mathcal{N}(x, \mu, \sigma) = \sum_k \alpha_k \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[\frac{-(x-\mu)^2}{2\sigma^2}\right]$
- Θ all parameters, $\sum_k \alpha_k = 1$
- Likelihood: $p(x|\Theta) = \prod p(x_i|\Theta) = \prod \left[\sum_k \alpha_k \mathcal{N}(x, \mu, \sigma)\right]$
- Log-Likelihood
  $l(x|\Theta) = \sum_i \ln\big(p(x_i|\mu_k, \sigma_k)\big) = \sum_i \ln\big(\sum_k \alpha_k \mathcal{N}(x, \mu, \sigma)\big)$
- New **latent** variable z, $z \in 1, 2, \ldots, K$.

EM Derivation

- Probability of an observation x_i belonging to cluster z: $p(z|x_i, \mu_k, \sigma_k)$
- $p(x_i|\Theta) = \sum_k p(x_i|z = k, \mu_k, \sigma_k)p(z = k)$
- α_k is the prior of $p(z = k)$ (Compare it with $p(x|\Theta) = \sum_k \alpha_k \mathcal{N}(x|\mu_k, \sigma_k)$)
- Conditional probability of x given z = k is $p(x_i|z = k, \mu_k, \sigma_k) = \mathcal{N}(x_i, \mu_k, \sigma_k)$
- Re-express the Log-Likelihood:

$$
\begin{aligned}
l(x|\Theta) &= \sum_i \ln\big[p(x_i, z|\mu_k, \sigma_k)\big] \\
&= \sum_i \ln\Big[\sum_k p(x_i|z = k, \mu_k, \sigma_k)p(z = k)\Big] \\
&= \sum_i \ln\Big[p(z = k|x_i, \mu_k, \sigma_k) \times \frac{p(x_i|z = k, \mu_k, \sigma_k)p(z = k)}{p(z = k|x, \mu_k, \sigma_k)}\Big]
\end{aligned}
$$

EM Derivation

- We use the **Jensen's Inequality**: $f(\mathbb{E}[x]) \geq \mathbb{E}[f(x)]$

$$l(x|\Theta) \geq \sum_i \sum_k p(z=k|x_i,\mu_k,\sigma_k) \ln \frac{p(x_i|z=k,\mu_k,\sigma_k)p(z=k)}{p(z=k|x_i,\mu_k,\sigma_k)}$$

- Now, we use the **Bayes' Theorem**: $f_{X|Y=y}(x) = \frac{f_{Y|X=x}(y)f_X(x)}{f_Y(y)}$

$$\begin{aligned} p(z=k|x_i,\mu_k,\sigma_k) &= \frac{p(x_i|z=k,\mu_k,\sigma_k)}{\sum_k p(x_i|z=k,\mu_k,\sigma_k)} \\ &= \frac{\alpha_k \mathcal{N}(x_i|\mu_k,\sigma_k)}{\sum_k \alpha_k \mathcal{N}(x_i|\mu_k,\sigma_k)} \\ &= \gamma_{i,k} \end{aligned}$$

EM Derivation

- Therefore, we rewrite the Log-Likelihood:

$$l(x|\Theta) = \sum_i \ln \sum_k \gamma_{i,k} \frac{\alpha_k \mathcal{N}(x_i|\mu_k, \sigma_k)}{\gamma_{i,k}}$$

- With the Jensen's Inequality:

$$\sum_i \ln \sum_k \gamma_{i,k} \frac{\alpha_k \mathcal{N}(x_i|\mu_k, \sigma_k)}{\gamma_{i,k}} \geq \sum_i \sum_k \gamma_{i,k} \ln \frac{\alpha_k \mathcal{N}(x_i|\mu_k, \sigma_k)}{\gamma_{i,k}}$$

EM Derivation

- This is the Lower Bound of the Log-Likelihood function, and we can use it as target (part of the Expectation Step).

$$
\begin{aligned}
Q(\Theta, \Theta^t) &= \sum_i \sum_k \gamma_{i,k}^t \ln\left[\frac{\alpha_k \mathcal{N}(x_i|\mu_k, \sigma_k)}{\gamma_{i,k}^t}\right] \\
&= \sum_i \sum_k \gamma_{i,k}^t \ln\left(\frac{\alpha_k}{\gamma_{i,k}^t \sqrt{2\pi\sigma_k^2}} exp\left[\frac{-(x_i - \mu_k)^2}{2\sigma_k^2}\right]\right) \\
&= \sum_i \sum_k \gamma_{i,k}^t \left[\ln \alpha_k - \ln \gamma_{i,k}^t - \ln \sqrt{2\pi\sigma_k^2} - \frac{(x_i - \mu_k)^2}{2\sigma_k^2}\right]
\end{aligned}
$$

EM for Univariate Gaussian Mixture Model

- The latent variable z is captured in the term $\gamma_{i,k}^{t}$. This term is the focus of the **Expectation Step**.

$$\gamma_{i,k}^{t} = \frac{\alpha_k \mathcal{N}(x_i|\mu_k, \sigma_k)}{\sum_k \alpha_k \mathcal{N}(x_i|\mu_k, \sigma_k)}$$

- The **Maximization Step** it is straightforward:

$$\Theta \equiv \arg\max_{\Theta} Q(\Theta, \Theta^t)$$

- Optimize α_k given the constraint $\sum_k \alpha_k = 1$:

$$\begin{aligned} \alpha_k^{t+1} &\equiv \arg\max_{\Theta} \sum_i \sum_k \gamma_{i,k}^{t} \ln \alpha_k \\ &= \frac{\sum_i \gamma_{i,k}^{t}}{N} \text{ Based on Lagrangian} \end{aligned}$$

EM for Univariate Gaussian Mixture Model

- Optimize μ_k:

$$\begin{aligned}\mu_k^{t+1} &\equiv \arg\max_{\mu_k} Q(\Theta, \Theta^t) \\ &= \frac{\sum_i \gamma_{i,k}^t x_i}{\sum_i \gamma_{i,k}^t} \text{ Based on Derivative}\end{aligned}$$

- Optimize σ_k:

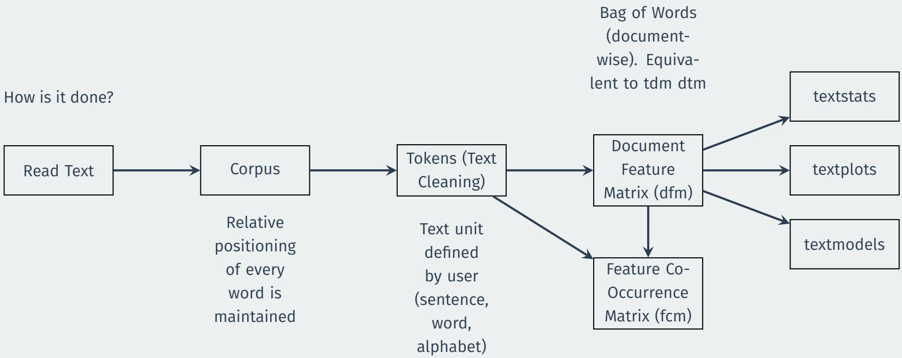
$$\begin{aligned}\sigma_k^{t+1} &\equiv \arg\max_{\sigma_k} Q(\Theta, \Theta^t) \\ (\sigma_k^2)^{t+1} &= \frac{\sum_i \gamma_{i,k}^t (x_i - \mu_k^{t+1})^2}{\sum_i \gamma_{i,k}^t} \text{ Based on Derivative}\end{aligned}$$

3/ Text Analysis: Applications

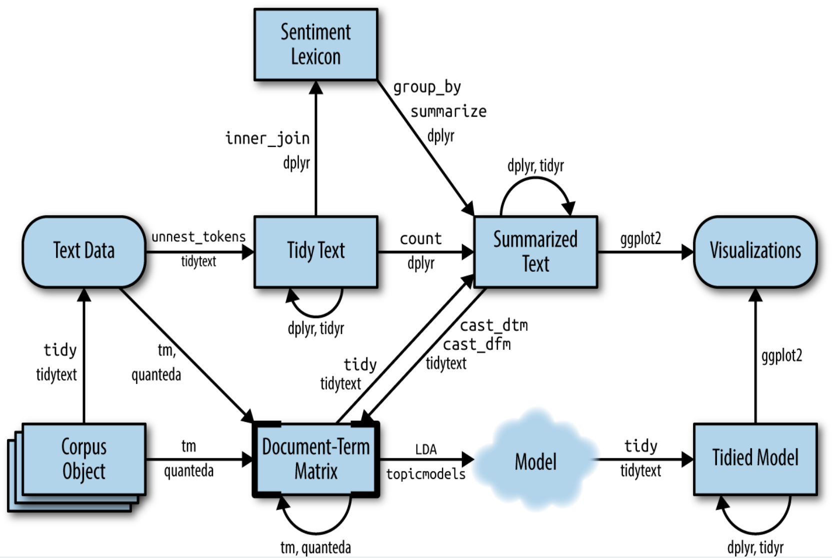
Concepts

- **Corpus:** "library" of **original** documents that have been converted to plain, UTF-8 encoded text, and stored along with meta-data at the corpus level and at the document-level (`docvars`: document-level metadata).
- **Tokens:** character vector (words or sentences) delimited by word boundaries.
- **Document-Feature Matrix:** tabulated structure with documents as rows and **features** (raw terms, stemmed terms, parts of speech of terms, etc,) as columns.
- Many alternative packages Comparison of Functions.

Quanteda FlowChart



General Text Analysis Workflow



Basic Steps: Load Data

- To import the data, you can use all kinds of functions available in packages like `tidyverse`, `data.table`, `sjlabelled`, etc.
- The `quanteda` package is complemented by the `readtext` package, which is dedicated to importing text strings from different formats.
- Next, you have to create the `corpus` (the key argument is `text_field`, especially if the original object is a `data.frame`).

R Code

```
### Load packages
library(quanteda)
library(readtext)

### Import data
raw_text <- readtext(file = "crime.csv")

### Create corpus
corpus_cr <- corpus(raw_text, text_field = "document")
summary(corpus_cr)

#    Text Types   Tokens  Sentences
#    doc1   407      646         10
#    doc2   581     1086         23
#    doc3   679     1330         41
```

Basic Steps: Pre-Processing

- The idea is to preserve the corpus intact. But there are several operations you can perform on these objects (`corpus_subset()`, `corpus_reshape` `corpus_sample()`).
- The quanteda package (or any text analysis package) has many functions for data pre-processing. However, it is important to be aware that there are many tools available in `stringr` (`str_to_lower()`, `str_flatten(` `str_squish`).
- The next step is to tokenize with the `tokens()` function. The function only introduces the pre-processing steps that you explicitly indicate.

R Code

```
### Tokenize
tokens_cr <- tokens(corpus_cr, remove_numbers = TRUE, remove_punct = TRUE, remove_symbols = TRUE,
                    remove_url  = TRUE)

tokens_esp <- tokens_wordstem(tokens_remove(x = tokens_cr, pattern = stopwords("esp")))
```

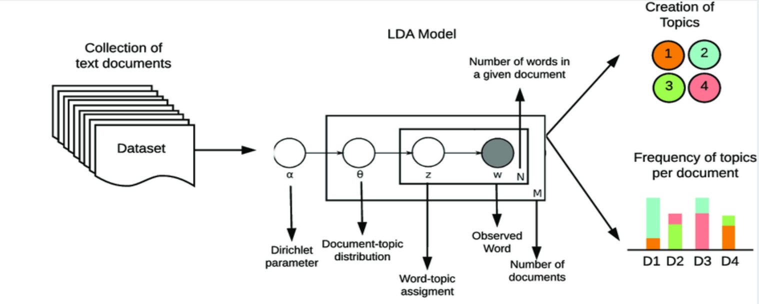
Data Modelling: DFM and LDA

- Tokenization is only an intermediate step. The ultimate goal is to create a document-feature matrix (or similar). This is the kind of goal you will use in most analyses (descriptive and inferential).
- In the Pset, you will use one of the most basic models: Latent Dirichlet Allocation (use the `seededlda` package).

R Code

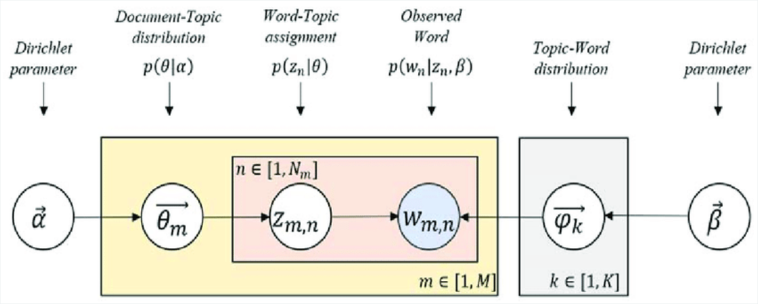
```
### Document-term matrix
dfm_cr <- dfm(tokens_esp)

### Load packages
library(seededlda)

### Estimate LDA
lda_exploratory <- textmodel_lda(dfm_cr, k = 25, maxiter = 5000)
```

LDA Workflow



	Word1	word2	word3	word4	.....
Topic1	0.01	0.23	0.19	0.03	
Topic2	0.21	0.07	0.48	0.02	
Topic3	0.53	0.01	0.17	0.04	

LDA Diagram



- Three-Level hierarchical model (Dirichlet: distribution over distributions).
- Documents are a mixture of topics.
- Topics are a mixture of words/terms/features.
- **Key Assumptions:** bag of words(ordering is unimportant), documents are exchangeable, topics are independent/uncorrelated.