11 - Text as Data

ml4econ, HUJI 2025

Itamar Caspi
June 15, 2025 (updated: 2025-06-15)

Packages

```
if (!require("pacman"))  install.packages("pacman")

pacman::p_load(
  quanteda,
  textdata,
  tidytext,
  tidyverse,
  knitr,
  xaringan,
  RefManageR
)
```

```
## package 'textdata' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##     C:\Users\internet\AppData\Local\Temp\RtmpwHla0e\downloaded_packages
```

The `quanteda` package is a Swiss army knife for handling text with R. More on that later.

Outline

- Representing Text as Data
- Text Regressions
- Dictionary-based Methods
- Topic Modeling

Representing Text as Data

Getting Started with Text Analysis

- For an insightful introduction to text analysis, featuring numerous real-world examples from the social sciences, consider reading "Text as Data" by Gentzkow, Kelly, and Taddy (JEL 2019).

- Comprehensive lecture notes provided by Maximilian Kasy from Harvard and Matt Taddy from Chicago and Amazon also serve as valuable resources.

Essential Terminology for Text Analysis

Let's define some basic terms:

- *Corpus*: This refers to a collection of D documents. These documents can be emails, tweets, speeches, articles, etc.
- *Vocabulary*: This is a comprehensive list of unique words appearing in the corpus.
- $\mathbf{X}$: This is a numerical array representation of text. Here, rows represent documents indexed as $i = 1, \ldots, D$ and columns correspond to words indexed as $j = 1, \ldots, N$.
- Y: This is a vector of predicted outcomes (e.g., spam/ham, trump/not trump, etc.), with one outcome allocated per document.
- F: This stands for a low-dimensional representation of $\mathbf{X}$.

Document Term Matrix (DTM)

In many applications, raw text transforms into a numerical array $\mathbf{X}$.

Here, the elements of the array, X_{ij}, represent counts of words or, more generally, *tokens*. We will discuss this in more detail later.

An Example: Distinguishing Spam from Ham

Consider the task of spam detection:

ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got amore wat...
ham	Ok lar... Joking wif u oni...
spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question(std txt rate)T&C's apply 08452810075over18's
ham	U dun say so early hor... U c already then say...
ham	Nah I don't think he goes to usf, he lives around here though
spam	FreeMsg Hey there darling it's been 3 week's now and no word back! I'd like some fun you up for it still? Tb ok! XxX std chgs to send, å£1.50 to rcv
ham	Even my brother is not like to speak with me. They treat me like aids patent.
ham	As per your request 'Melle Melle (Oru Minnaminunginte Nurungu Vettam)' has been set as your callertune for all Callers. Press *9 to copy your friends Callertune
spam	WINNER!! As a valued network customer you have been selected to receivea å£900 prize reward! To claim call 09061701461. Claim code KL341. Valid 12 hours only.
spam	Had your mobile 11 months or more? U R entitled to Update to the latest colour mobiles with camera for Free! Call The Mobile Update Co FREE on 08002986030

In this scenario:

- Documents refer to individual emails.
- Vocabulary comprises words that appear in *every single* email.

NOTE: Clearly, spam detection constitutes a supervised learning task where $Y_i = \{\text{spam}, \text{ham}\}$.

Converting a Corpus to a DTM

Let's take a look at a corpus with two documents ($D = 2$):

```
txt <- c(doc1 = "Shipment of gold damaged in a fire.",
         doc2 = "Delivery of silver, arrived in 2 silver trucks")

tokens_txt <- tokens(txt)  # tokenize the text

tokens_txt %>% quanteda::dfm() # transform tokes into a document term matrix
```

```
## Document-feature matrix of: 2 documents, 14 features (42.86% sparse) and 0 docvars.
##        features
## docs    shipment of gold damaged in a fire . delivery silver
##   doc1        1  1    1       1  1 1    1 1        0      0
##   doc2        0  1    0       0  1 0    0 0        1      2
## [ reached max_nfeat ... 4 more features ]
```

This example originates from the quanteda's getting started examples.

Do All Words Matter? ¯_(ツ)_/¯

We can considerably reduce the dimension of $\mathbf{X}$ by:

- Excluding highly common ("stop words") and rare words.
- Eliminating numbers and punctuation.
- Implementing stemming, i.e., replacing words with their roots (use *economi* instead of *economics*, *economists*, *economy*).
- Converting all text to lower case.

WARNING: Be judicious when using text preprocessing steps. These steps should be tailored to the specific application.

Demonstrating Common Preprocessing Steps

In the following example, we remove stop words, punctuation, numbers, and implement word stemming:

```
txt <- c(doc1 = "Shipment of gold damaged in a fire.",
         doc2 = "Delivery of silver, arrived in 2 silver trucks")

txt %>%
  tokens(remove_punct = TRUE, remove_numbers = TRUE) |>
  dfm() |>
  dfm_wordstem() |>
  dfm_remove(stopwords("english"))
```

```
## Document-feature matrix of: 2 documents, 8 features (50.00% sparse) and 0 docvars.
##        features
## docs    shipment gold damag fire deliveri silver arriv truck
##   doc1         1    1     1    1         0      0     0     0
##   doc2         0    0     0    0         1      2     1     1
```

Please note, the number of features has been reduced from 14 to 8.

Introduction to n-grams

- In certain scenarios, multiword expressions such as "not guilty" or "labor market" might be significant.
- We can define tokens (the fundamental units of text) as n-grams, which are sequences of n words from a given text sample.

NOTE: Using n-grams with n>2 typically becomes impractical as the column dimension of $\mathbf{X}$ grows exponentially with the order n.

DTM with Bigrams

In this example, our sample text includes just two "documents". Here, we define tokens as *bigrams* (sequences of two words):

```
txt %>%
  tokens(remove_punct = TRUE, remove_numbers = TRUE) %>%
  tokens_ngrams() %>%
  dfm()
```

```
## Document-feature matrix of: 2 documents, 12 features (50.00% sparse) and 0 docvars.
##       features
## docs   shipment_of of_gold gold_damaged damaged_in in_a a_fire delivery_of of_silver silver
##   doc1           1       1            1          1    1      1           0         0
##   doc2           0       0            0          0    0      0           1         1
##       features
## docs   arrived_in
##   doc1          0
##   doc2          1
## [ reached max_nfeat ... 2 more features ]
```

The Text Mining Playbook for Social Sciences

Follow these steps for effective text mining:

1. Collect text and create a corpus.

2. Represent the corpus as a DTM $\mathbf{X}$.

3. Next, choose one of the following steps:

    - Employ $\mathbf{X}$ to predict an outcome Y using high-dimensional methods (e.g., lasso, Ridge, etc.). In some scenarios, proceed with $\hat{Y}$ for subsequent analysis.
    - Use dimensionality reduction techniques (like dictionary, PCA, LDA, etc.) on $\mathbf{X}$ and proceed with the resulting $\mathbf{F}$ for further analysis.

> "*Text information is usually best as part of a larger system. Use text data to fill in the cracks around what you know. Don't ignore good variables with stronger signal than text!*" (Matt Taddy)

Text Regression

Familiar Territory: High Dimensionality Problem

Our aim is to predict a certain Y using $\mathbf{X}$. Evidently, dealing with text data introduces the high-dimensionality issue, where $\mathbf{X}$ has $M \times N$ elements.

Traditional methods like OLS fall short in this case $\Rightarrow$ thus the need for machine learning approaches.

Penalized linear/non-linear regression methods (like Lasso, Ridge, etc.) are typically suitable. Other methods such as random forest may also work.

EXAMPLE: Consider Lasso text regression `glmnet(Y, X)` where:

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^N}{\operatorname{argmin}} \sum_{i=1}^{N} (Y_i - X_i\beta)^2 + \lambda \|\boldsymbol{\beta}\|_1$$

This method easily extends to binary / categorical Y, e.g., `glmnet(X, Y, family = "binomial")`

Practical Guidelines for Using Penalized Text Regression

- DTM entries usually count the number of times word i appears in document d, which provides an "intuitive" interpretation for regression coefficients.
- Depending on the application, different transformations for $\mathbf{X}$ might be more suitable, such as:
  - Normalizing each row by document length.
  - Using a binary inclusion dummy instead of count.
- However, refrain from attributing a causal interpretation to the Lasso's coefficients (remember the irrepresentability condition).

Dictionary-based Methods

Dimensionality Reduction Using Dictionaries

- Dictionary-based methods offer a low-dimensional representation of high-dimensional text data.
- This is, by far, the most frequently employed method in social science literature that utilizes text (Gentzkow et al., forthcoming).
- Essentially, consider F as an unobserved characteristic of the text that we're trying to estimate. Dictionary-based methods provide a mapping from $\mathbf{X}$ onto a lower-dimensional F:

$$g : \mathbf{X} \rightarrow F$$

Example: Sentiment Analysis

- A common example of dictionary-based methods is sentiment analysis.
- The latent factor we aim to estimate is the writer's attitude towards the discussed topic.
- The prevalent approach relies on predefined dictionaries that classify words according to predetermined sentiment classes, such as "positive", "negative", and "neutral".
- The sentiment *score* of each document is typically a function of the relative frequencies of positive, negative, neutral, etc., words.

REMARK: Sentiment analysis can also be supervised. For instance, available labeled movie reviews (rated 1-5 stars) can be used to train a model, and its predictions can then be used to classify unlabeled reviews.

Example: Loughran and McDonald Financial Sentiment Dictionary

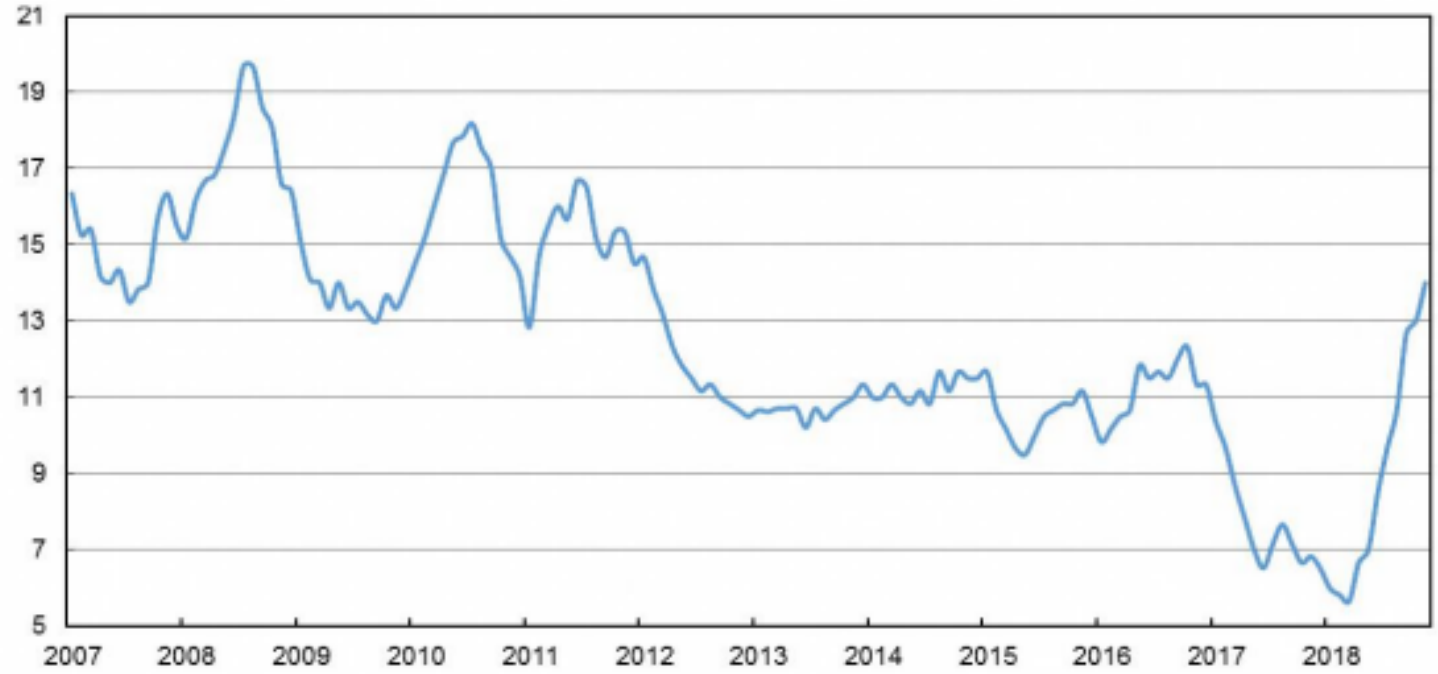
Below is a random list of words from the Loughran and McDonald (2011) financial sentiment dictionary, which includes positive, negative, litigious, uncertain, and constraining sentiments:

```
library(tidytext)
sample_n(get_sentiments("loughran"),8)
```

```
## Do you want to download:
##  Name: Loughran-McDonald Sentiment lexicon
##  URL: https://sraf.nd.edu/textual-analysis/resources/
##  License: License required for commercial use. Please contact tloughra@nd.edu.
##  Size: 6.7 MB (cleaned 142 KB)
##  Download mechanism: https
##
## 1: Yes
## 2: No
##
## Enter an item from the menu, or 0 to exit
```

Application: Bank of Israel Communication

Figure 4
Index of Uncertainty in the Interest Rate Announcements, 2007–18



Source: Benchimol and Caspi (2019)

Topic Modeling

Topic Models

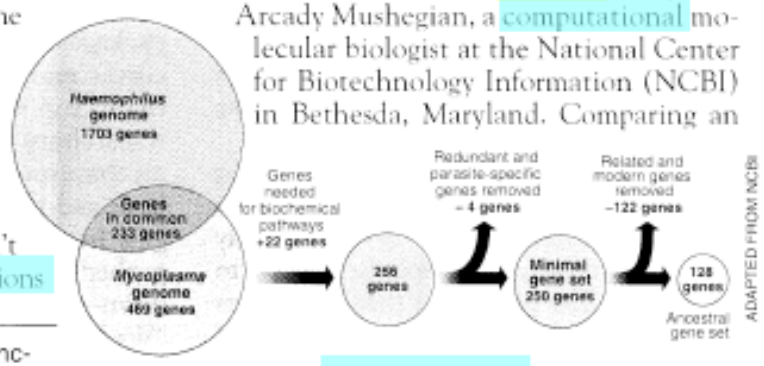
- Topic models enhance unsupervised learning methods for text data.
- They classify documents and words into latent topics, often serving as a precursor to more conventional empirical methods.
- The cornerstone of topic modeling is the Latent Dirichlet Allocation model (Blei, Ng, and Jordan, 2003), commonly referred to as LDA.

Intuition Behind LDA

Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

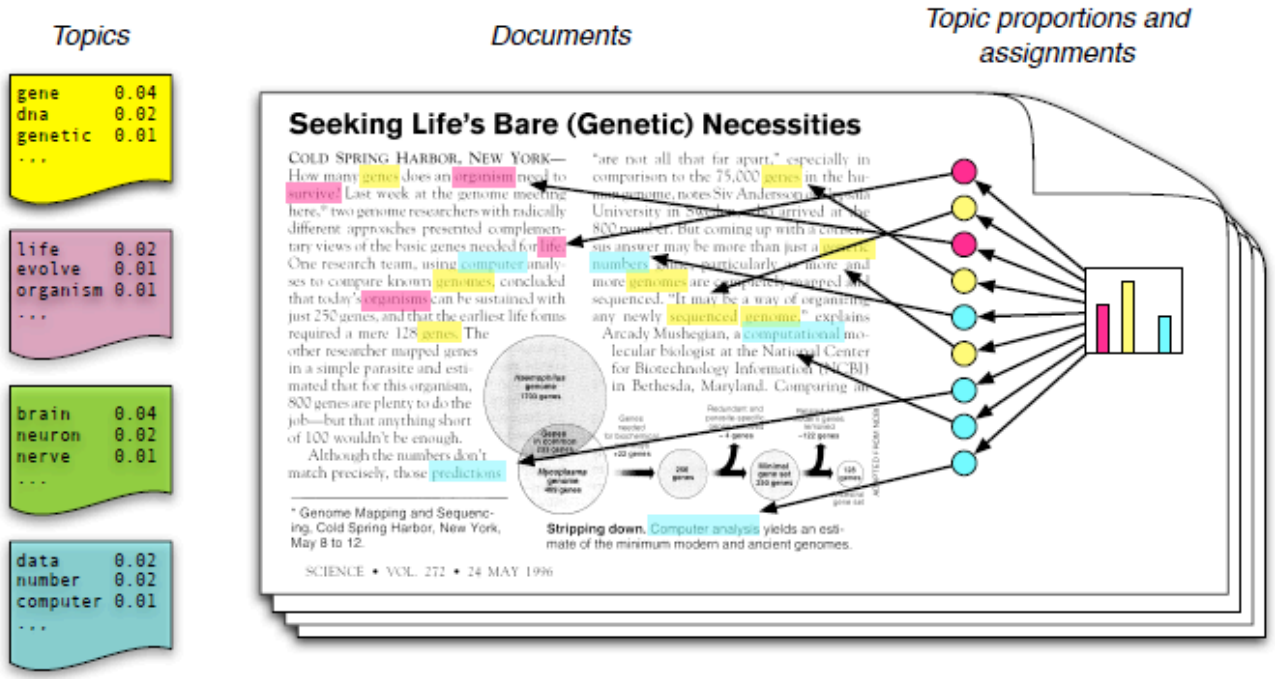
Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an



Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

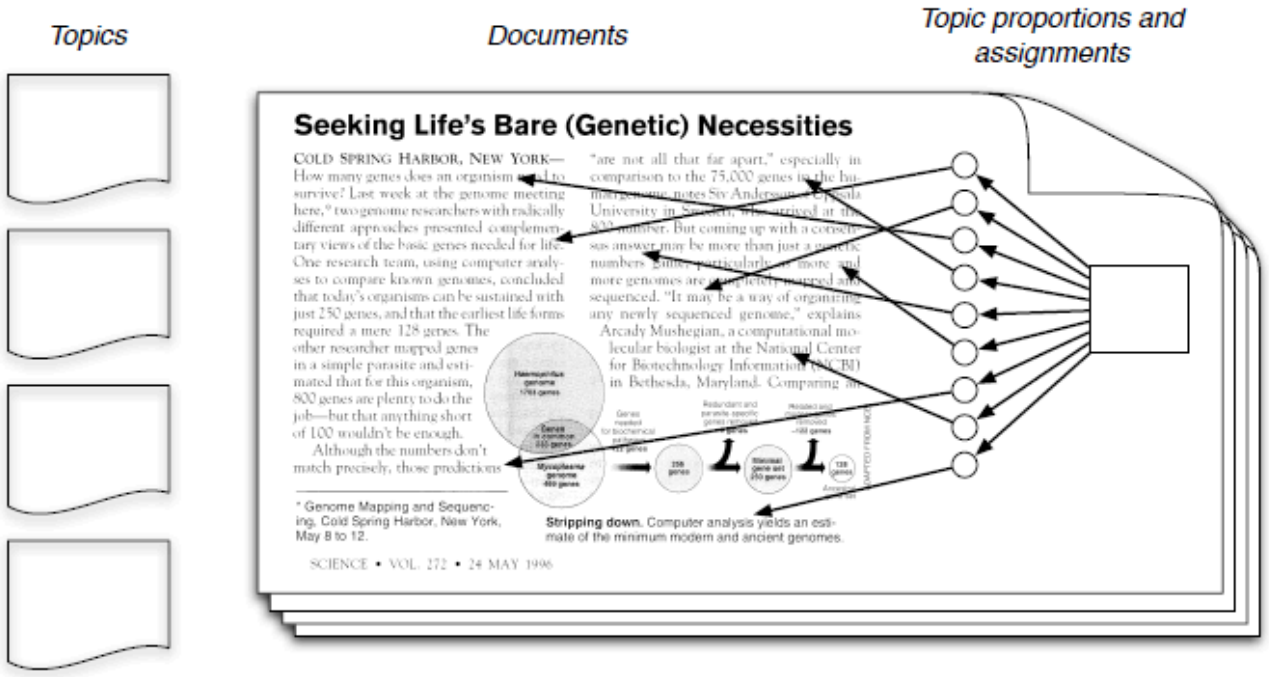
* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

The Intuition Behind LDA



- A topic is a distribution across *all* the words within a *fixed* vocabulary.
- A word can have non-zero probabilities in multiple topics (e.g., "bank").
- Each document is a mixture of different topics.
- Each word is selected from one of these topics.

Intuition Behind LDA



QUESTION: How realistic is the LDA setup? Does it matter? What's our goal here anyway?

Notation

- A *vocabulary* comprises of words represented by the vector $\{1, \ldots, V\}$.
- Each *word* is represented by a unit vector $\boldsymbol{\delta}_v = (0, \ldots, v, \ldots, 0)'$.
- A *document* is a sequence of N words denoted by $\mathbf{w} = (w_1, \ldots, w_N)$.
- A *corpus* is a collection of M documents denoted by $\mathcal{D} = (\mathbf{w}_1, \ldots, \mathbf{w}_M)$.

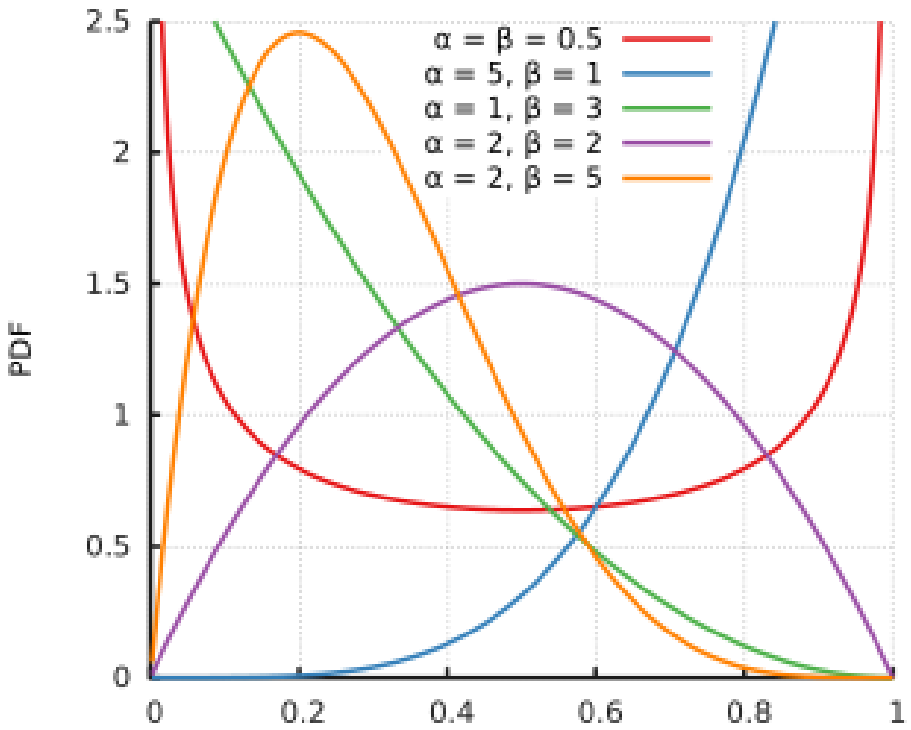
Prerequisite: The Beta Distribution

The probability density function (PDF) for the Beta distribution, denoted as $B(\alpha, \beta)$, is given by:

$$p(\theta|\alpha, \beta) \propto \theta^{\alpha-1}(1-\theta)^{\beta-1}$$

This function holds for $\theta \in [0, 1]$ and $\alpha, \beta > 0$.

Due to its properties, the Beta distribution is useful as a prior for probabilities.



The Dirichlet Distribution

The Dirichlet distribution, denoted as $\mathbf{Dir}(\boldsymbol{\alpha})$, is a multivariate generalization of the Beta distribution.

Let $\theta = (\theta_1, \theta_2, \ldots, \theta_K) \sim \mathbf{Dir}(\boldsymbol{\alpha})$.

The probability density function (PDF) for a K-dimensional Dirichlet distribution is

$$p(\boldsymbol{\theta}|\boldsymbol{\alpha}) \propto \prod_{i=1}^{K} \theta_i^{\alpha_i - 1}$$

Here, $K \geq 2$ is the number of categories, $\alpha_i > 0$ and $\theta_i \in (0, 1)$ for all i and $\sum_{i=1}^{K} \theta_i = 1$.

Remark: The parameter $\boldsymbol{\alpha}$ controls the sparsity of $\boldsymbol{\theta}$.

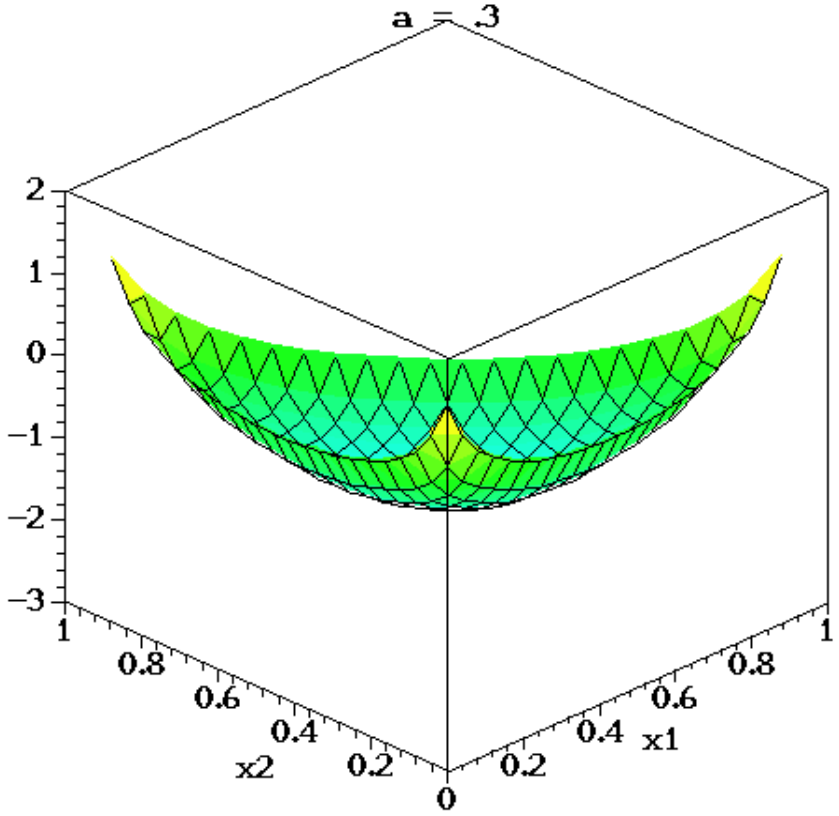
Bottom Line: Vectors drawn from a Dirichlet distribution represent probabilities.

Visualizing the Dirichlet Distribution

On the right:

The change in the density function ($K = 3$) as the vector $\boldsymbol{\alpha}$ changes from $\boldsymbol{\alpha} = (0.3, 0.3, 0.3)$ to $(2.0, 2.0, 2.0)$, while keeping $\alpha_1 = \alpha_2 = \alpha_3$.

Remark: Placing $\boldsymbol{\alpha} = (1, 1, 1)$ results in a uniform distribution over the simplex.



By Initial version by Panos Ipeirotis, later

The Data Generating Process Behind LDA

Assumption: The number of topics K and the size of the vocabulary V are fixed.

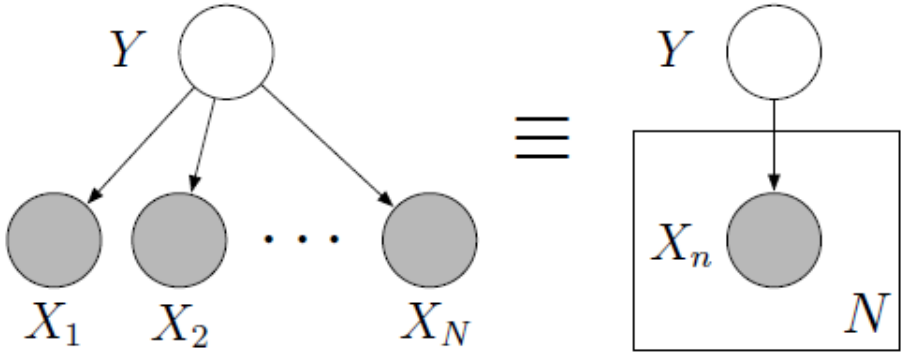
The Data Generating Process (DGP):

For each document $d = 1, \ldots, \mathcal{D}$:

1. Choose topic proportions $\theta_d \sim \mathbf{Dir}(\boldsymbol{\alpha})$.
2. For each word $n = 1, \ldots, N$:
    2.1. Choose a topic assignment $Z_{dn} \sim \mathbf{Mult}(\theta_d)$.
    2.2. Choose a word $W_{dn} \sim \mathbf{Mult}(\beta_{z_{dn}})$.

Remark: Note the "factor model" aspects of LDA, where topics act as factors and word probabilities act as loadings, both affecting the probability of selecting a word.

Aside: Plate Notation

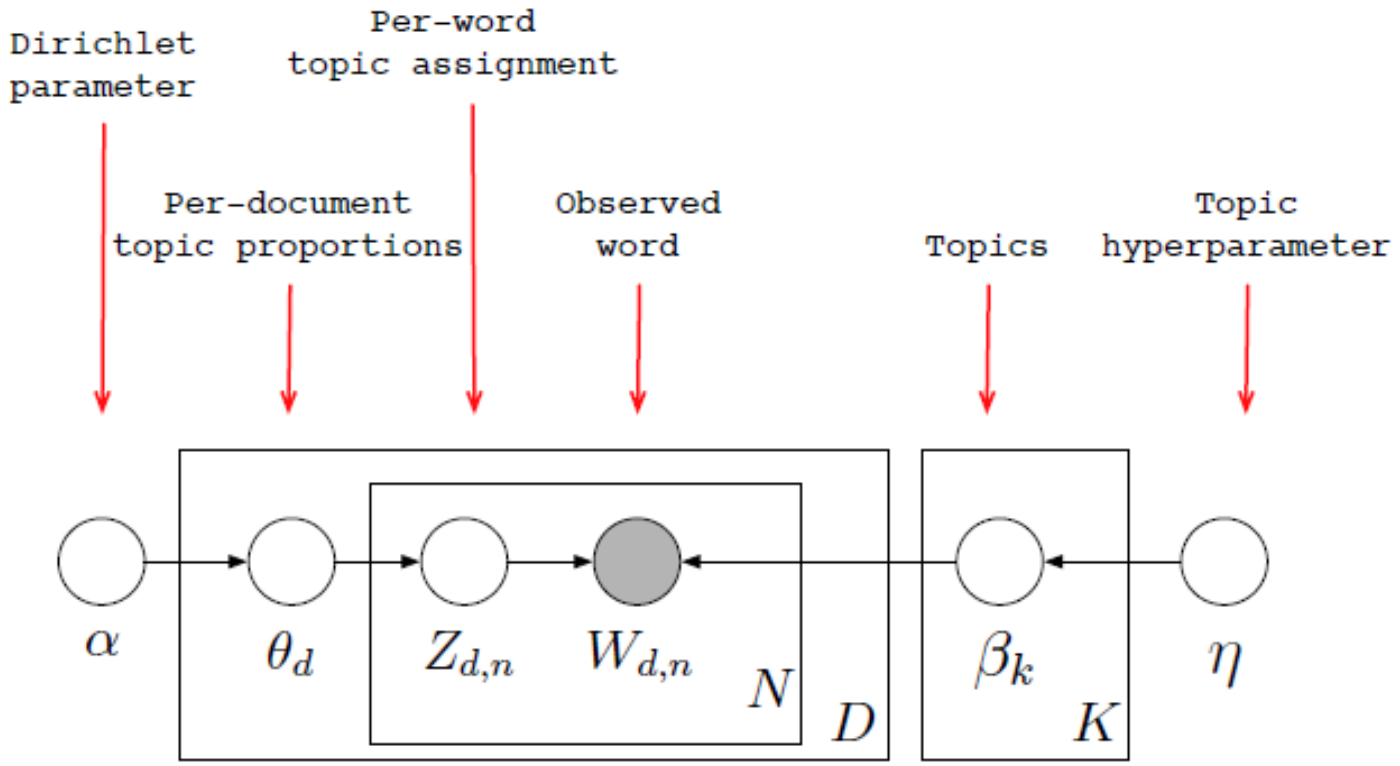


- Each *node* represents a random variable.
- *Shaded* nodes indicate observables.
- *Edges* represent dependencies.
- *Plates* indicate replicated structures.

The depicted graph corresponds to the following expression:

$$p\left(y, x_1, \ldots, x_N\right) = p(y) \prod_{n=1}^{N} p\left(x_n | y\right)$$

LDA in plate notation



Source: http://videolectures.net/mlss09uk_blei_tm/#.

Aside: Conjugate Priors

The Dirichlet distribution serves as a conjugate prior for the Multinomial distribution.

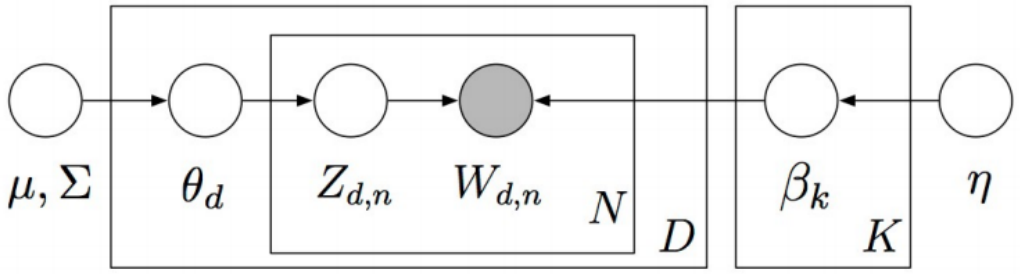
Let $n(Z_i)$ denote the count of topic i.

$$\boldsymbol{\theta}|Z_{1,\ldots,N} \sim \mathrm{Dir}(\boldsymbol{\alpha} + n(Z_{1,\ldots,N}))$$

In other words, as the number of times we observe topic i increases, our posterior distribution becomes more concentrated around the i^{th} component of $\boldsymbol{\theta}$.

Extension #1: Correlated Topic Models (Lafferty and Blei, 2005)

- LDA assumes that topics independently co-occur in documents.
- However, this assumption is clearly incorrect.
- For instance, a document about *economics* is more likely to also discuss *politics* than it is to talk about *cooking*.
- Lafferty and Blei address this issue by relaxing the independence assumption and drawing topic proportions from a logistic normal distribution, allowing for correlations between topic proportions:



Here, μ and Σ represent priors for the logistic normal distribution.

Extension #2: Dynamic LDA (Blei and Lafferty, 2006)

Dynamic LDA takes into account the ordering of documents and provides a more detailed posterior topical structure compared to traditional LDA.

In dynamic topic modeling, a topic is a *sequence* of distributions over words. Topics evolve systematically over time. Specifically, the parameter vector for topic k in period t evolves with Gaussian noise:

$$\beta_{t,k}|\beta_{t-1,k} \sim \mathcal{N}\left(\beta_{t-1,k}, \sigma^2 I\right).$$

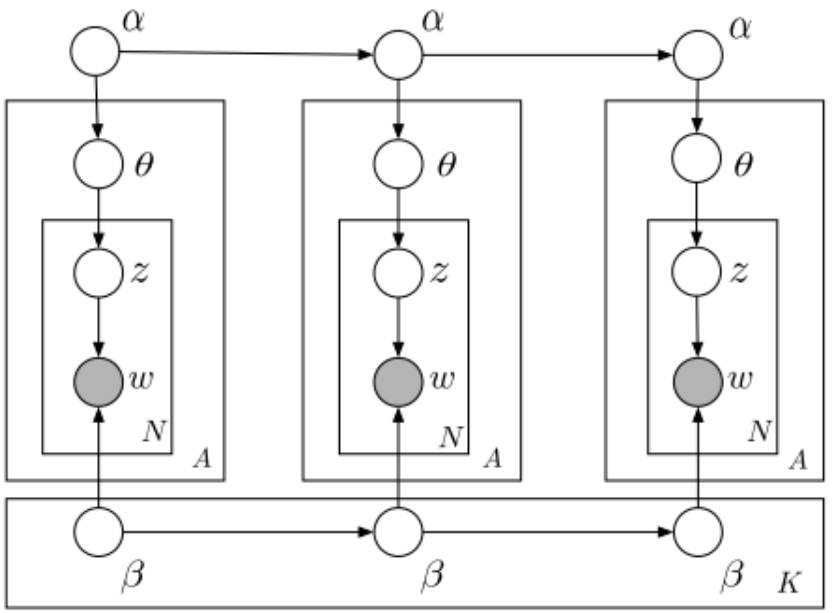
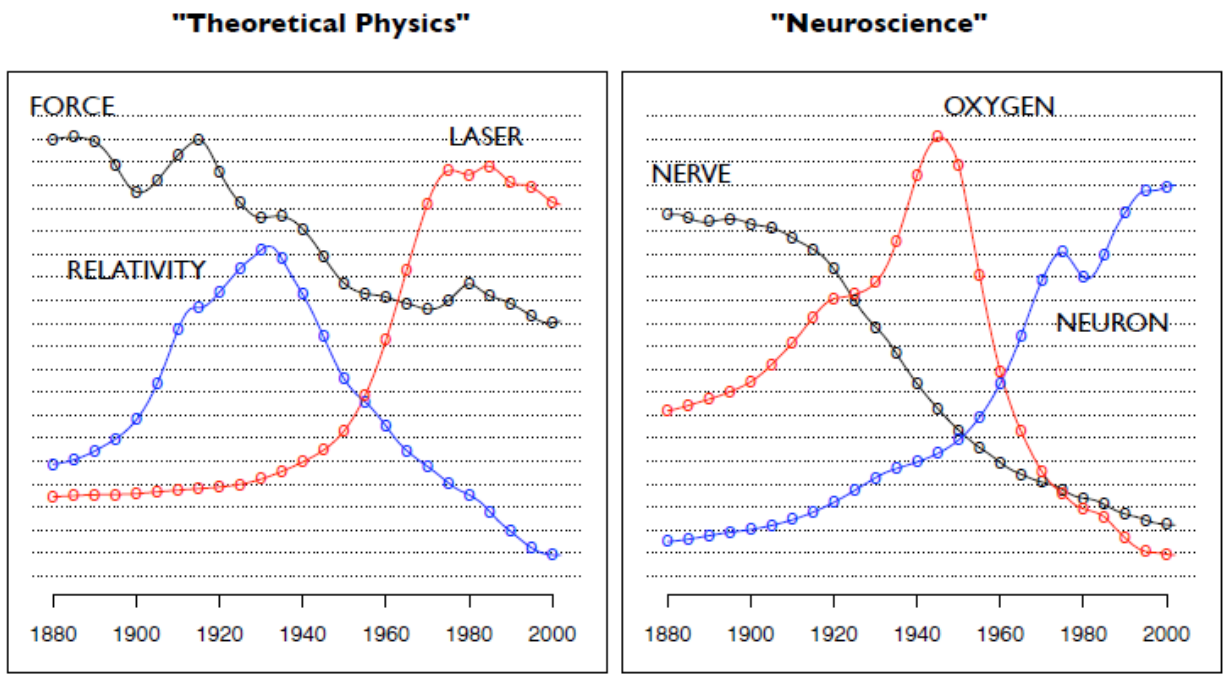


Figure 1. Graphical representation of a dynamic topic model (for three time slices). Each topic's natural parameters $\beta_{t,k}$ evolve over time, together with the mean parameters α_t of the logistic normal distribution for the topic proportions.

Dynamic LDA: *Science*, 1881-1999

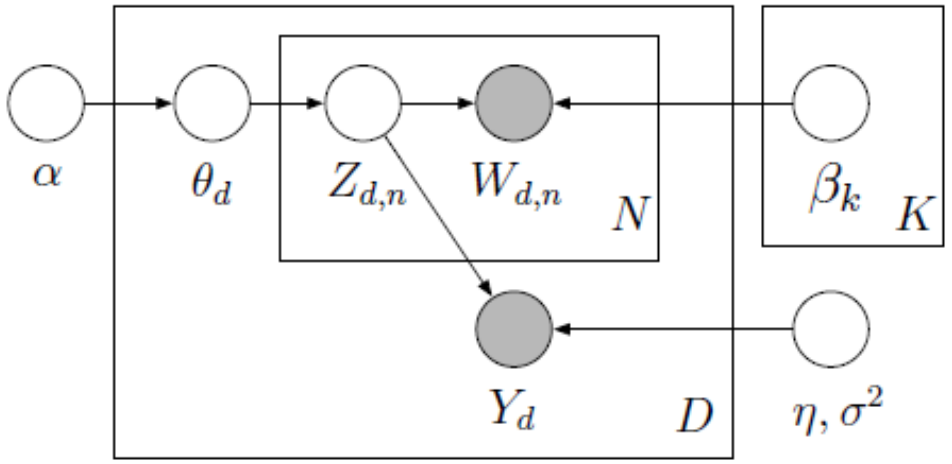
The posterior estimate of the frequency of several words as a function of year for two topics, "Theoretical Physics" and "Neuroscience":



Source: Blei and Lafferty (2006).

Extension #3: Supervised Topic Model (McAuliffe and Blei, 2008)

An additional connection is made between Z_{dn} and an observable attribute Y_d in the Supervised Topic Model:



Source: McAuliffe and Blei (2008).

Structural Topic Models (Roberts, Stewart, and Tingley)

About the Structural Topic Model (STM):

> *"The Structural Topic Model is a general framework for topic modeling with document-level covariate information. The covariates can improve inference and qualitative interpretability and are allowed to affect topical prevalence, topical content or both."*

In STM, topics are drawn from the following logistic normal distribution,

$$\boldsymbol{\theta}_d | X_d\gamma, \Sigma \sim \text{ LogisticNormal } (\mu = X_d\gamma, \Sigma)$$

where $\boldsymbol{X}_d$ is a vector of observed document covariates.

REMARK: In the case of no covariates, the STM reduces to a (fast) implementation of the Correlated Topic Model (Blei and Lafferty, 2007).

`stm`: R Package for Structural Topic Models

Roberts, Stewart, and Tingley (JSS, 2014)

About the `stm` R package:

> *"The software package implements the estimation algorithms for the model and also includes tools for every stage of a standard workflow from reading in and processing raw text through making publication quality figures."*

The package is available on CRAN and can be installed using:

```
install.packages("stm")
```

To get started, see the vignette which includes several example analyses.

Applying Topic Models to Measure the Effect of Transparency

Hansen, McMahon, and Prat (QJE 2017) examine the impact of increased transparency in the Federal Open Market Committee (FOMC) meetings on the level of debate.

Here are some key points:

- FOMC meetings have been recorded since the 1970s to create minutes.
- Committee members were under the impression that these tapes were erased afterward.
- In October 1993, Fed Chair Alan Greenspan discovered and revealed that the tapes had been transcribed and stored in archives all along before being erased.
- Following Greenspan's revelation, the Fed agreed to publish all past transcripts and extended this policy to include all future transcripts with a five-year time lag.
- This provides Hansen et al. with access to periods when policymakers believed their deliberations would and would not be made public.

Topic Modeling of FOMC Meeting Transcripts

Data:

- The dataset consists of 149 FOMC meeting transcripts during Alan Greenspan's tenure, spanning both pre-1993 and post-1993 periods.
- The unit of observation is a member-meeting.
- The outcomes of interest include:
  - The proportion of words devoted to K different topics.
  - The concentration of topic weights.
  - The frequency of data citation.

Estimation

To estimate the topics, LDA (Latent Dirichlet Allocation) is employed.

The LDA output is then utilized to construct the outcomes of interest.

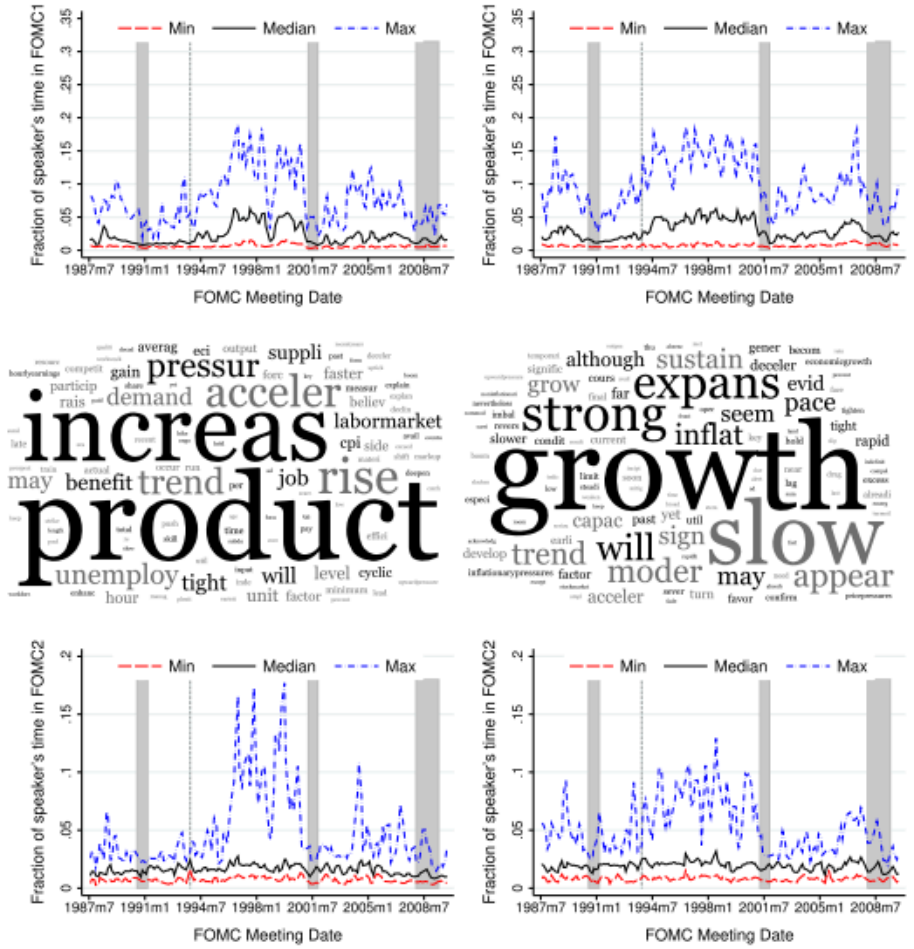
Difference-in-Differences regressions are applied to estimate the effects of the change in transparency on these outcomes. For instance, Hansen et al. estimate the following model:

$$y_{it} = \alpha_i + \gamma D(\mathbf{Trans})_t + \lambda X_t + \varepsilon_{it}$$

Here:

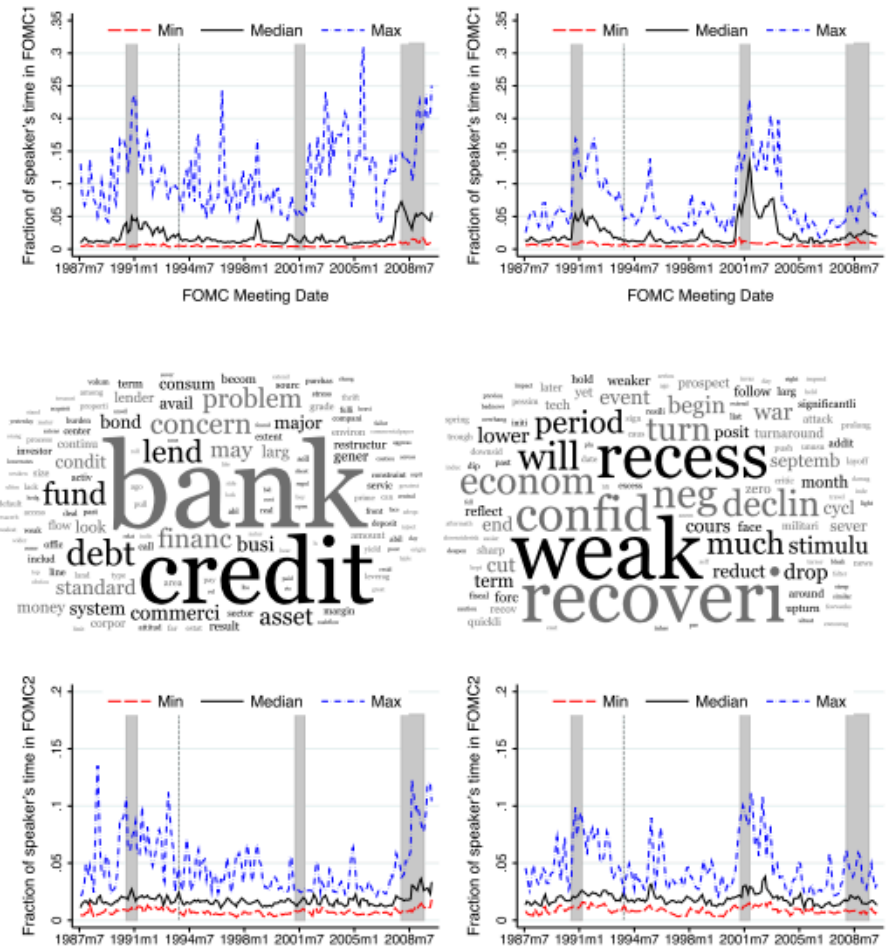
- y_{it} represents any of the communication measures for member i at time t.
- $D(\mathbf{Trans})$ is an indicator for being in the transparency regime (1 after November 1993, 0 before).
- X_t is a vector of macro controls for the meeting at time t.

Pro-cyclical Topics



Source: Hansen, McMahon, and Prat (QJE 2017).

Counter-cyclical Topics



Source: Hansen, McMahon, and Prat (QJE 2017).

Increased Accountability: More References to Data

TABLE V

DIFFERENCE RESULTS FOR ECONOMIC SITUATION DISCUSSION (FOMC1): COUNT MEASURES

Main regressors	Words (1)	Statements (2)	Questions (3)	Numbers (4)
D(Trans)	56.7*	−0.52	−0.039	3.71***
	[.076]	[.162]	[.659]	[.003]
D(Recession)	−1.95	−0.69	−0.19	−0.71
	[.952]	[.159]	[.314]	[.488]
EPU index	0.30	−0.00094	0.00088	0.0040
	[.186]	[.876]	[.586]	[.520]
D(2 day)	27.1	1.36*	0.56*	1.28
	[.256]	[.085]	[.051]	[.188]
# of PhDs	6.68	−0.45***	−0.11***	0.51
	[.561]	[.005]	[.009]	[.109]
Constant	528***	10.0***	2.44***	1.50
	[.002]	[.000]	[.000]	[.740]
Unique members	19	19	19	19
Observations	903	903	903	903
Member FE	Yes	Yes	Yes	Yes
Time FE	No	No	No	No
Meeting section	FOMC1	FOMC1	FOMC1	FOMC1
Transparency effect	9.5*	−10	−2.5	53.2***

Source: Hansen, McMahon, and Prat (QJE 2017).

Increased Conformity: Increased Document Similarity

TABLE VI

DIFFERENCE RESULTS FOR ECONOMIC SITUATION DISCUSSION (FOMC1): TOPIC MEASURES

Main regressors	Concentration (1)	Quant (2)	Avg Sim (B) (3)	Avg Sim (D) (4)	Avg Sim (KL) (5)
D(Trans)	0.0041	−0.00027	0.0082***	0.0012	0.032***
	[.205]	[.831]	[.001]	[.692]	[.000]
D(Recession)	0.0061**	−0.000056	0.0020	0.015***	−0.0017
	[.028]	[.968]	[.385]	[.000]	[.758]
EPU index	3.7e-06	−9.6e-06	0.000050*	0.000029	0.00015
	[.890]	[.541]	[.077]	[.300]	[.109]
D(2 day)	−0.0040*	0.0042**	0.00044	−0.0037***	0.00051
	[.093]	[.024]	[.802]	[.001]	[.914]
# of PhDs	0.0017	−0.00063	0.000097	0.00079	0.00018
	[.255]	[.292]	[.885]	[.671]	[.928]
# Stems	0.000075***	8.8e-06**	−3.5e-06	0.000030***	0.000049
	[.000]	[.049]	[.837]	[.001]	[.284]
Constant	0.13***	0.037***	0.89***	0.084***	0.62***
	[.000]	[.000]	[.000]	[.001]	[.000]
Unique members	19	19	19	19	19
Observation	903	903	903	903	903
Member FE	Yes	Yes	Yes	Yes	Yes
Time FE	No	No	No	No	No
Meeting section	FOMC1	FOMC1	FOMC1	FOMC1	FOMC1
Topics	P1	T4 & T23	P1	P1	P1
Similarity measure	—	—	Bhatta-charyya	Dot product	Kullback-Leibler
Transparency effect	2.5	−0.7	0.9***	1.1	4.9***

Source: Hansen, McMahon, and Prat (QJE 2017).

```
slides |> end()
```

Source code

Selected References

- Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan), 993-1022.
- Blei, D. M., & Lafferty, J. D. (2006, June). Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning* (pp. 113-120). ACM.
- Gentzkow, M., Kelly, B.T., & Taddy, M. (2019). Text as data. *Journal of Economic Literature* 57(3), 535-574.
- Hansen, S., McMahon, M., & Prat, A. (2017). Transparency and Deliberation Within the FOMC: A Computational Linguistics Approach. *The Quarterly Journal of Economics*, 133(2), 801–870.
- Lafferty, J. D., & Blei, D. M. (2006). Correlated topic models. In *Advances in neural information processing systems* (pp. 147-154).
- Loughran, T., & McDonald, B. (2011). When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks. *The Journal of Finance*, 66(1), 35-65.

Selected References

- Roberts, M.E., Stewart, B.M., Tingley, D., Lucas, C., Leder-Luis, J., Gadarian, S.K., Albertson, B., & Rand, D.G. (2014). Structural topic models for open-ended survey responses. *American Journal of Political Science*, 58(4), 1064-1082.

- Roberts, M.E., Stewart, B.M., & Tingley, D. (2014). stm: R package for structural topic models. *Journal of Statistical Software*, 10(2), 1-40.