Lecture 11: Topic Models

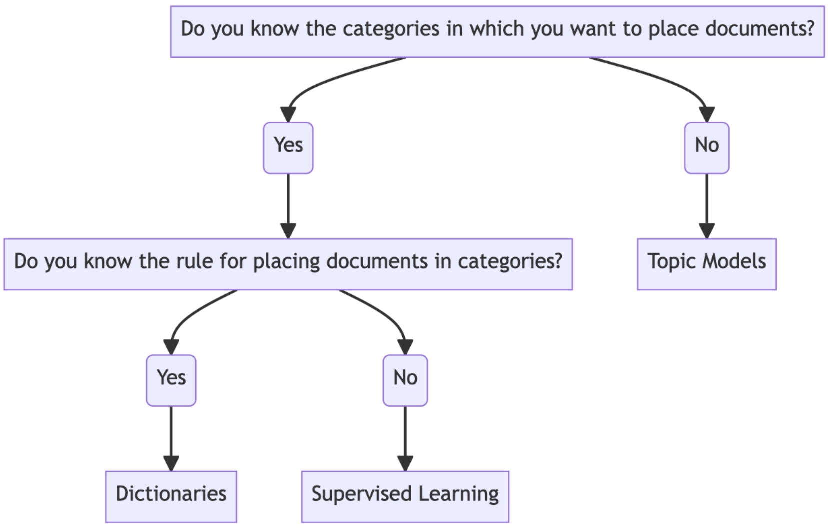
Jack Blumenau

Today's lecture

Topic Models

Topic Models

- Topic models allow us to cluster similar documents in a corpus together.
- Wait. Don’t we already have tools for that?
- Yes! Dictionaries and supervised learning.
- So what do topic models add?




```
# ```{mermaid}
# %%| fig-width: 10
# %%| fig-height: 5
#
# flowchart TD
#   A[Do you know the categories in which you want to place documents?] --> B(Yes)
#   A[Do you know the categories in which you want to place documents?] --> G(No)
#   B --> C[Do you know the rule for placing documents in categories?]
#   C --> D(Yes)
#   C --> E(No)
#   D --> Fa[Dictionaries]
#   E --> Fb[Supervised Learning]
#   G --> H[Topic Models]
```

Topic Models

Pause for motivating material!

Topic Models

- Topic models offer an automated procedure for discovering the main “themes” in an unstructured corpus
- They require no prior information, training set, or labelling of texts before estimation
- They allow us to automatically organise, understand, and summarise large archives of text data.
- Latent Dirichlet Allocation (LDA) is the most common approach (Blei et al., 2003), and one that underpins more complex models
- Topic models are an example of *mixture* models:
  - Documents can contain multiple topics
  - Words can belong to multiple topics

Topic Models as Language Models

- In the last lecture, we introduced the idea of a *probabilistic language model*
  - These models describe a story about how documents are generated using probability
- A language model is represented by a probability distribution over words in a vocabulary
- The Naive Bayes text classification model is *one* example of a generative language model where
  - We estimate separate probability distributions for each category of interest
  - Each document is assigned to a single category
- Topic models are also language models
  - We estimate separate probability distributions for each topic
  - Each document is described as belonging to *multiple* topics

What is a “topic”?

A “topic” is a probability distribution over a fixed word vocabulary.

- Consider a vocabulary: gene, dna, genetic, data, number, computer
- When speaking about **genetics**, you will:
  - frequently use the words “gene”, “dna” & “genetic”
  - infrequently use the words “data”, “number” & “computer”
- When speaking about **computation**, you will:
  - frequently use the words “data”, “number” & “computation”
  - infrequently use the words “gene”, “dna” & “genetic”

Topic	gene	dna	genetic	data	number	computer
Genetics	0.4	0.25	0.3	0.02	0.02	0.01
Computation	0.02	0.01	0.02	0.3	0.4	0.25

Note that no word has probability of exactly 0 under either topic.

What is a “document”?

- In a topic model, each document is described as being composed of a **mixture** of corpus-wide topics
- For each document, we find the topic proportions that maximize the probability that we would observe the words in that particular document

Imagine we have two documents with the following word counts

Document word counts

Doc	gene	dna	genetic	data	number	computer
A	2	3	1	3	2	1
B	2	4	2	1	2	1

Topic probability distribu

Topic	gene	dna	genetic	data
Genetics	0.4	0.25	0.3	0.02
Computation	0.02	0.01	0.02	0.3

Topic Models

A topic model simultaneously estimates two sets of probabilities

1. The probability of observing each word for each topic
2. The probability of observing each topic in each document

These quantities can then be used to organise documents by topic, assess how topics vary across documents, etc.

Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation (LDA)

[PDF] **Latent dirichlet allocation**

DM Blei, AY Ng, MI Jordan - Journal of machine Learning research, 2003 - jmlr.org

We describe **latent Dirichlet allocation** (LDA), a generative probabilistic model for collections of discrete data such as text corpora. LDA is a three-level hierarchical Bayesian model, in …

Latent Dirichlet Allocation (LDA)

LDA is a probabilistic language model.

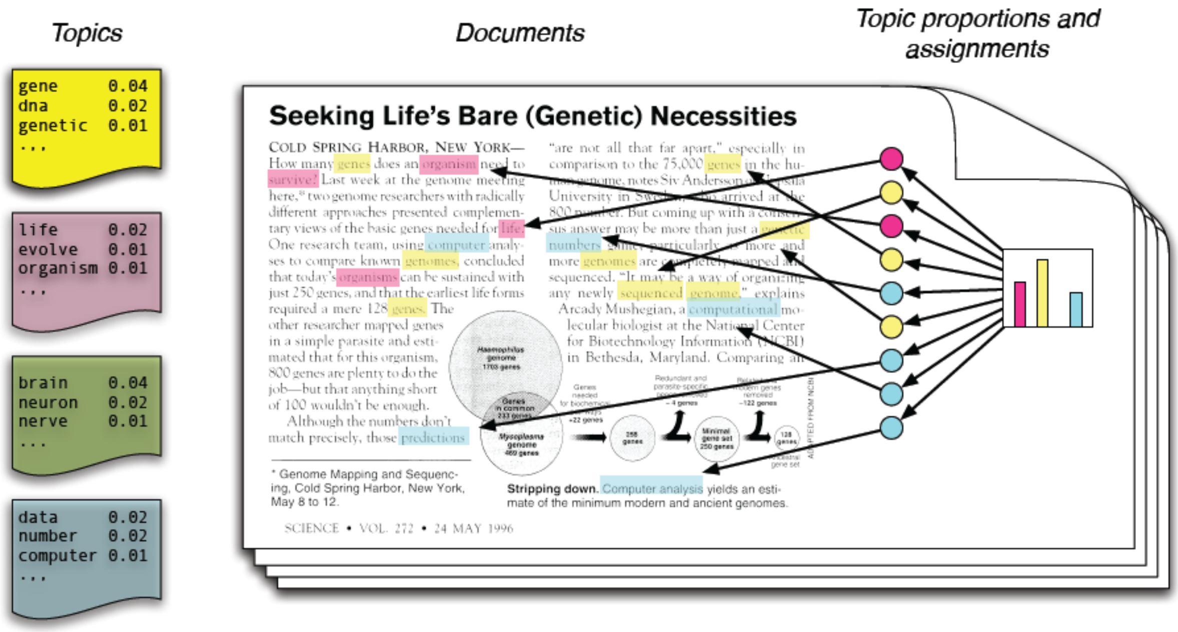
Each document d in the corpus is generated as follows:

1. A set of K topics exists before the data
   - Each topic k is a probability distribution over words (β)
2. A specific mix of those topics is randomly extracted to generate a document
   - More precisely, this mix is a specific probability distribution over topics (θ)
3. Each word in a document is generating by:
   - First, choosing a topic k at random from the probability distribution over topics θ
   - Then, choosing a word w at random from the topic-specific probability distribition over documents (β_k)

However, we only observe documents!

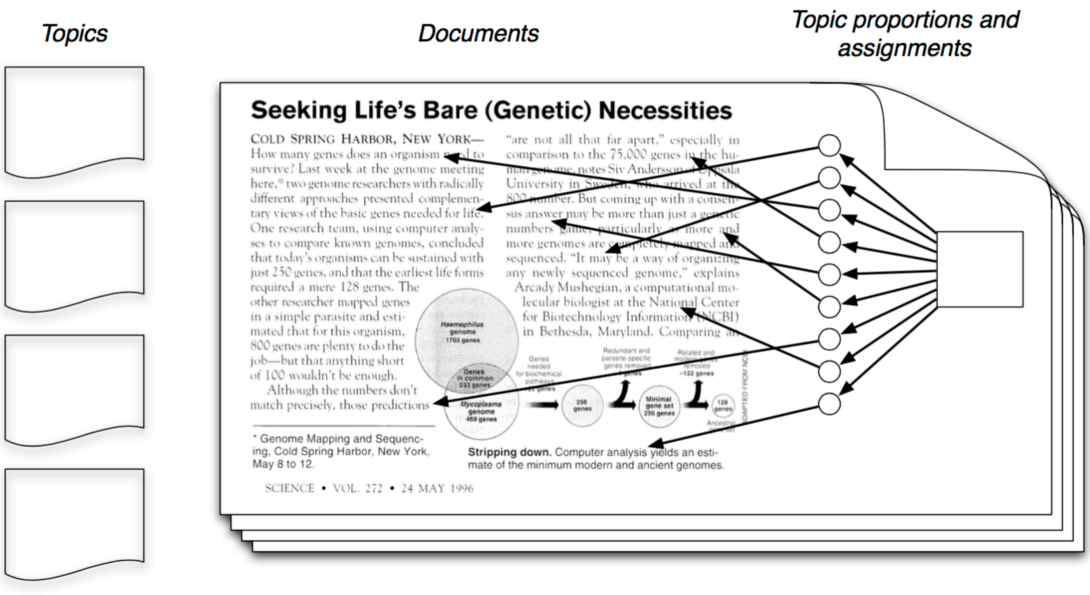
The goal of LDA is to estimate hidden parameters (β and θ) starting from w.

Latent Dirichlet Allocation (LDA)



- The researcher picks a number of topics, K .
- Each *topic* (k) is a distribution over words
- Each *document* (d) is a mixture of corpus-wide topics

Latent Dirichlet Allocation (LDA)



Latent Dirichlet Allocation (LDA)

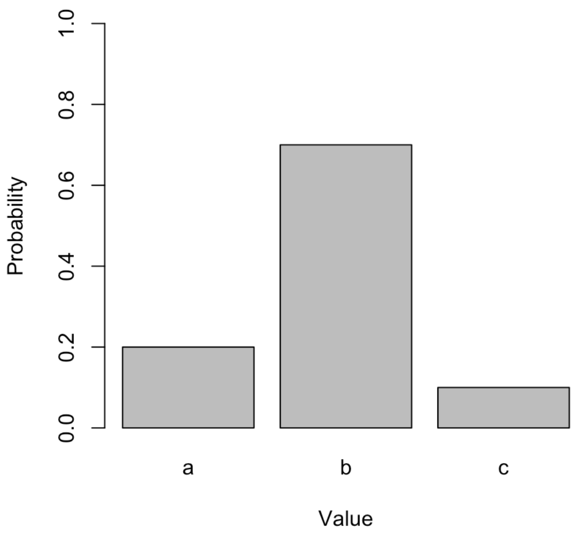
- The LDA model is a Bayesian mixture model for discrete data which describes how the documents in a dataset were created
- The number of topics, K , is selected by the researcher
- Each of the K *topics* is a probability distribution over a fixed vocabulary of N words
  - Modeled as a Dirichlet distribution
- Each of the D *documents* is a probability distribution over the K topics
  - Modeled as a Dirichlet distribution
- Each word in each document is drawn from the topic-specific probability distribution over words
  - Modeled as a multinomial distribution

Probability Distributions Review

- A probability distribution is a function that gives the probabilities of the occurrence of different possible outcomes for a random variable
- Probability distributions are defined by their parameters
  - E.g. In a normal distribution, μ describes the mean and σ^2 describes the variance
- Different parameter values change the distribution's shape and describe the probabilities of the different events
  - E.g. If $\sigma_1^2 > \sigma_2^2$, then $N(\mu, \sigma_1^2)$ has higer variance, fatter tails, describing a higher probability of extreme values
- The notation "$\sim$" means to "draw" from the distribution
  - E.g. $x \sim N(0, 1)$ means to draw one value from a standard normal, which might result in $X = 1.123$
- There are two key distributions that we need to know about to understand topic models: the Multinomial and the Dirichlet distributions

Multinomial Distribution

- The multinomial distribution is a probability d
describing the results of a random variable th
on one of K possible categories
- The multinomial distribution depicted has pr
$[0.2, 0.7, 0.1]$
- A draw (of size one) from a multinomial distril
returns one of the categories of the distributic
    - E.g.
    $c \sim \text{Multinomial}(1, [0.2, 0.7, 0.1])$
    might return $c = a$
- A draw of a larger size from a multinomial dist
returns several categories of the distribution i
to their probabilities
    - E.g.
    $C \sim \text{Multinomial}(10, [0.2, 0.7, 0.1])$
    might return $c_1 = a$, $c_2 = b$, $c_3 = b$ etc.



Dirichlet Distribution

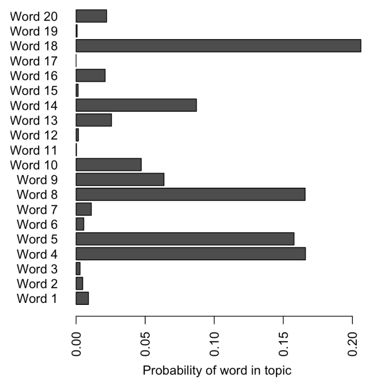
- The Dirichlet distribution is a distribution over the simplex, i.e., positive vectors that sum to one
- A draw from a dirichlet distribution returns a vector of positive numbers that sum to one
  - E.g. $b \sim \text{Dirichlet}(\alpha)$ might return $b = [0.2, 0.7, 0.1]$
- In other words, we can think of draws from a Dirichlet distribution being themselves multinomial distributions

- The parameter α controls the sparsity of the draws from the Dirichlet distribution.
  - When α is larger, the probabilities will be more evenly spread across categories

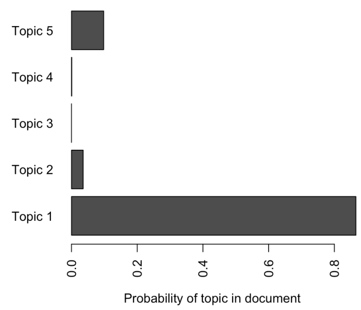
LDA Generative Process

LDA assumes a generative process for documents:

1. Each *topic* is a probability distribution over words
    - $\beta_k \sim \text{Dirichlet}(\eta)$, whith $\beta_k \in (0, 1)$ and $\sum_{j=1}^{J} \beta_{j,k} = 1$
    - $\rightarrow$ probability that each word (w) occurs in a given topic (k)

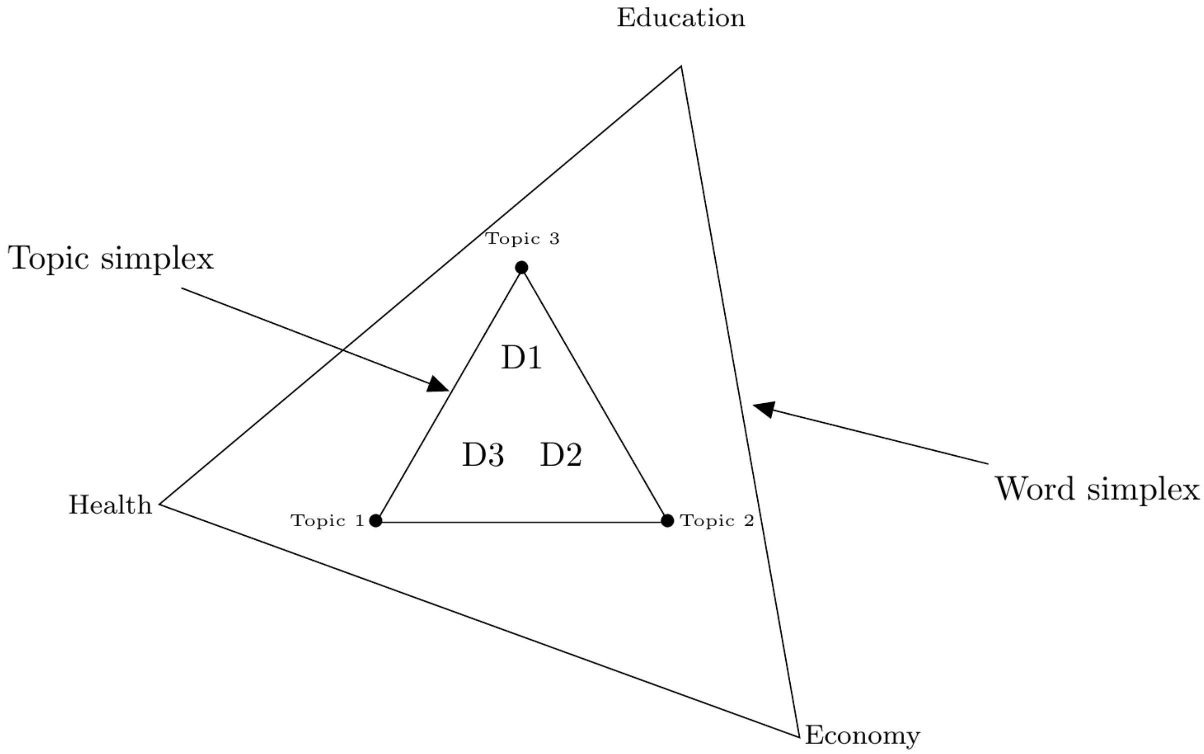


2. For each *document*, draw a probability distribution over topics
    - $\theta_d \sim \text{Dirichlet}(\alpha)$, with $\theta_{d,k} \in [0, 1]$ and $\sum_{k=1}^{K} \theta_{d,k} = 1$
    - $\rightarrow$ probability that each topic (k) occurs in a given document (d)

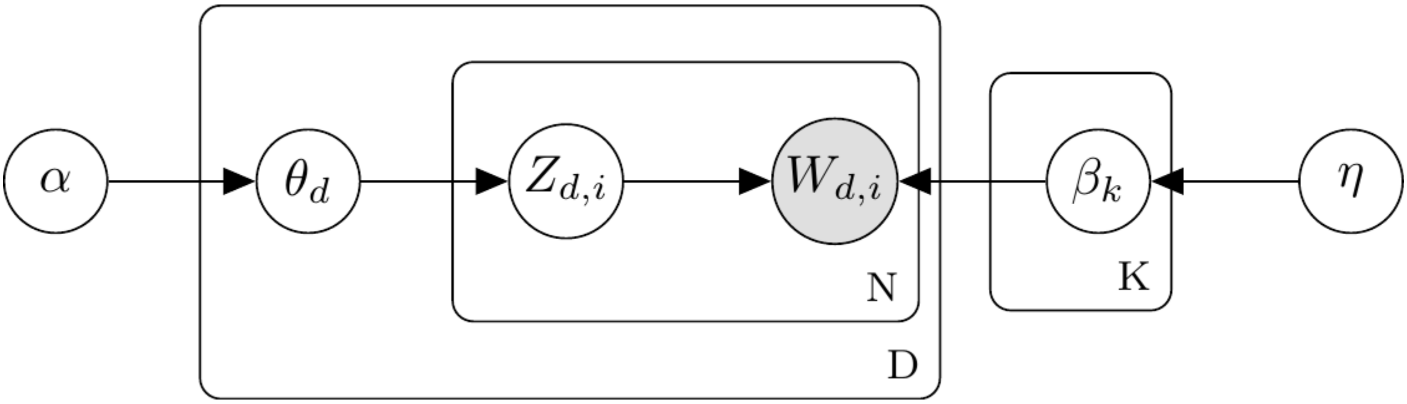


3. For each *word* in each document

Latent Dirichlet allocation (LDA)



LDA as a graphical model



LDA Estimation

- Assuming the documents have been generated in such a way, in return makes it possible to back out the shares of topics within documents and the share of words within topics
- Estimation of the LDA model is done in a Bayesian framework
- Our $Dir(\alpha)$ and $Dir(\eta)$ are the prior distributions of the θ_d and β_k
- We use Bayes' rule to update these prior distributions to obtain a posterior distribution for each θ_d and β_k
- The means of these posterior distributions are the outputs of statistical packages and which we use to investigate the θ_d and β_k
- Estimation is performed using either collapsed Gibbs sampling or variational methods
  - See Blei, 2012 for more details
- Fortunately, for us these are easily implemented in R

Why does LDA “work”?

- LDA trades off two goals.
  1. For each document, allocate its words to as few topics as possible (α)
  2. For each topic, assign high probability to as few terms as possible (η)
- These goals are at odds.
  1. Putting a document in a single topic makes (2) hard: All of its words must have probability under that topic.
  2. Putting very few words in each topic makes (1) hard: To cover a document’s words, it must assign many topics to it.
- Trading off these goals finds groups of **tightly co-occurring words**

LDA output

Imagine we have $D = 1000$ documents, $J = 10,000$ words, and $K = 3$ topics.

The key outputs of the topic model are the β and θ matrices:

$$\theta = \underbrace{\begin{pmatrix} \theta_{1,1} & \theta_{1,2} & \theta_{1,3} \\ \theta_{2,1} & \theta_{2,2} & \theta_{2,3} \\ \dots & \dots & \dots \\ \theta_{D,1} & \theta_{D,2} & \theta_{D,3} \end{pmatrix}}_{D\times K} = \underbrace{\begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.1 & 0.8 & 0.1 \\ \dots & \dots & \dots \\ 0.3 & 0.3 & 0.4 \end{pmatrix}}_{1000\times 3}$$

$$\beta = \underbrace{\begin{pmatrix} \beta_{1,1} & \beta_{1,2} & \dots & \beta_{1,J} \\ \beta_{2,1} & \beta_{2,2} & \dots & \beta_{2,J} \\ \beta_{3,1} & \beta_{3,2} & \dots & \beta_{3,J} \end{pmatrix}}_{K\times J} = \underbrace{\begin{pmatrix} 0.04 & 0.0001 & \dots & 0.003 \\ 0.0004 & 0.001 & \dots & 0.00005 \\ 0.002 & 0.0003 & \dots & 0.0008 \end{pmatrix}}_{3\times 10,000}$$

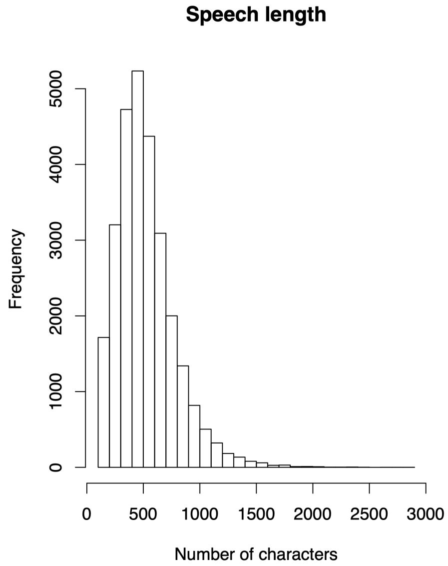
LDA example

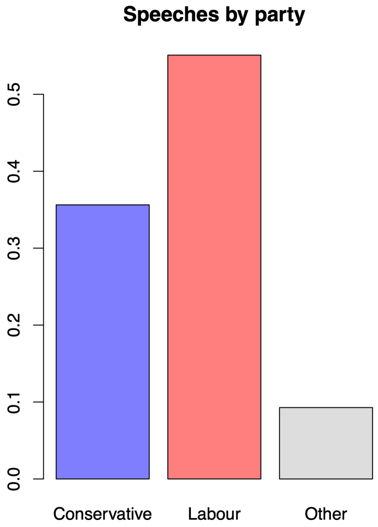
- Data: UK House of Commons' debates (PMQs)
  - ≈ 30000 parliamentary speeches from 1997 to 2015
  - ≈ 3000 unique words
  - $\approx 2m$ total words

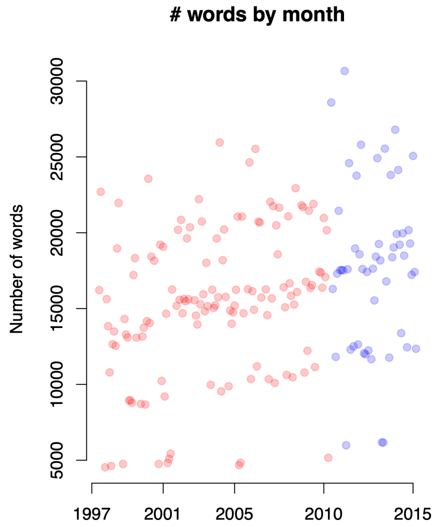
```
Rows: 27,885
Columns: 4
$ name         <chr> "Ian Bruce", "Tony Blair", "Denis MacShane", "Tony Blair"…
$ party        <chr> "Conservative", "Labour", "Labour", "Labour", "Liberal De…
$ constituency <chr> "South Dorset", "Sedgefield", "Rotherham", "Sedgefield", …
$ body         <chr> "In a written answer, the Treasury has just it made clear…
```

- Estimate a range of topic models ($K \in \{20, 30, \ldots, 100\}$) using the `topicmodels` package

LDA example


Speech length
Frequency
0
1000
2000
3000
4000
5000
0
500
1000
1500
2000
2500
3000
Number of characters


Speeches by party
0.0
0.1
0.2
0.3
0.4
0.5
Conservative
Labour
Other


words by month
Number of words
5000
10000
15000
20000
25000
30000
1997
2001
2005
2010
2015

Implementation in R

```r
library(quanteda)
library(topicmodels)

## Create corpus
pmq_corpus <- pmq %>%
  corpus(text_field = "body")

pmq_dfm <- pmq_corpus %>%
  tokens(remove_punct = TRUE) %>%
  dfm() %>%
  dfm_remove(stopwords("en")) %>%
  dfm_wordstem() %>%
  dfm_trim(min_termfreq = 5)

## Convert for usage in 'topicmodels' package
pmq_tm_dfm <- pmq_dfm %>%
  convert(to = 'topicmodels')
```

```r
## Estimate LDA
ldaOut <- LDA(pmq_tm_dfm, k = 40, method = "Gibbs")

save(ldaOut, file = "../data/scripts/ldaOut_40.Rdata")
```

LDA example

We will make use of the following score to visualise the posterior topics:

$$\text{term-score}_{k,v} = \hat{\beta}_{k,v} \log\left(\frac{\hat{\beta}_{k,v}}{\left(\prod_{j=1}^{K} \hat{\beta}_{j,v}\right)^{\frac{1}{K}}} \right)$$

- The first term, $\hat{\beta}_{k,v}$, is the probability of term v in topic k and is akin to the term frequency
- The second term down-weights terms that have high probability under all topics

This formulation is akin to the TFIDF term score

Implementation in R

```
# Extract estimated betas
topics <- tidy(ldaOut, matrix = "beta")

# Calculate the term scores

top_terms <- topics %>%
  group_by(term) %>%
  mutate(beta_k = prod(beta)^(1/20)) %>%
  ungroup() %>%
  mutate(term_score = beta*log(beta/(beta_k))) %>%
  group_by(topic) %>%
  slice_max(term_score, n = 10)

# Extract the terms with the largest scores per topic

top_terms$term[top_terms$topic==3]
```

```
[1] "economi"   "econom"    "interest" "plan"     "rate"     "countri"
[7] "deficit"   "s"         "growth"   "debt"
```

```
top_terms$term[top_terms$topic==19]
```

```
[1] "forc"        "iraq"        "defenc"      "british"     "afghanistan"
[6] "troop"       "secur"       "arm"         "war"         "weapon"
```

LDA example

LDA example

Topic 1	Topic 2	Topic 3	Topic 4
bank	terror	european	school
financi	terrorist	europ	educ
regul	secur	britain	children
england	attack	union	teacher
crisi	protect	british	pupil
fiscal	agre	referendum	class
market	act	constitut	parent

Topic 5	Topic 6	Topic 7	Topic 8
prison	nhs	plan	iraq
justic	wait	economi	weapon
crimin	hospit	econom	war
crime	cancer	growth	un
releas	patient	grow	resolut
court	list	longterm	iraqi
sentenc	health	deliv	saddam

Top Document by Topic

Advantages and Disadvantages of LDA

Advantages

- Automatically finds substantively interesting collections of words
- Automatically labels documents in “meaningful” ways
- Easily scaled to large corpora (millions of documents)
- Requires very little prior work (no manual labelling of texts/dictionary construction etc)

Disadvantages

- Generated topics may not reflect substantive interest of researcher
- Many estimated topics may be redundant for research question
- Requires extensive post-hoc interpretation of topics
- Sensitivity to number of topics selected (what is the best choice for K ?)

LDA Example (Alvero et al, 2021)

LDA Example (Alvero et al, 2021)

- **Research question:** Is the content of written essays less correlated with income than SATs?
- **Research Design**:
  - Topic model ($k = 70$) applied to 60k student admission essays.
  - Calculate correlation between a) topics and SAT scores, b) topics and student family income.
  - Additional analysis of essay "style" (using the LIWC dictionary)

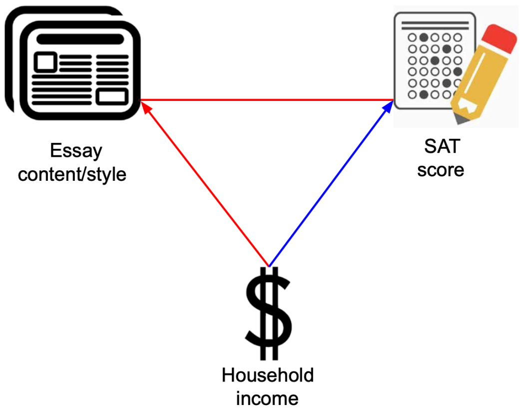


Fig. 1. Conceptual model. Visualization of previous work, represented by a blue line, and our study, represented by red lines, on the relationship between application materials and household income.

LDA Example (Alvero et al, 2021)

LDA Example (Alvero et al, 2021)

Conclusions

1. Topical content strongly predicts household income
2. Topical content strongly predicts SAT scores
3. Even conditional on income, topics predict SAT scores

“Our results strongly suggest that the imprint of social class will be found in even the fuzziest of application materials.”

Break

Extensions

Extending LDA

- LDA can be **embedded in more complicated models**, embodying further intuitions about the structure of the texts.
  - E.g., it can be used in models that account for syntax, authorship, word sense, dynamics, correlation, hierarchies, and other structure.
- The **data generating distribution** can be changed. We can apply mixed-membership assumptions to many kinds of data.
  - E.g., we can build models of images, social networks, music, purchase histories, computer code, genetic data, and other types.
- The **posterior** can be used in creative ways.
  - E.g., we can use inferences in information retrieval, recommendation, similarity, visualization, summarization, and other applications.

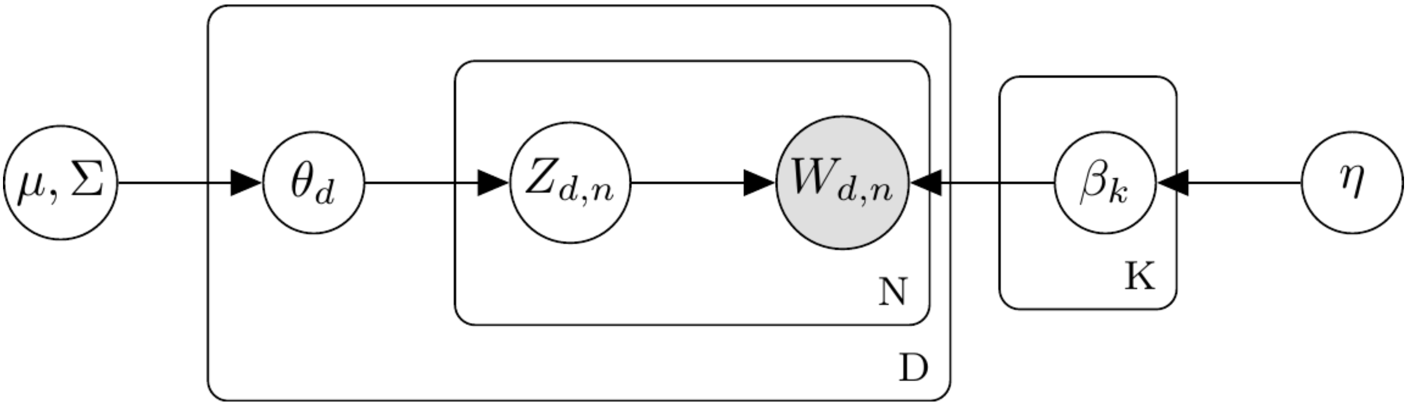
LDA Extensions

1. **Correlated Topic Model (CTM)**
    - LDA assumes that topics are uncorrelated across the corpus
    - The correlated topic model allows topics to be correlated
    - Closer approximation to true document structure, but estimation is slower
2. **Dynamic Topic Model (DTM)**
    - LDA assumes that topics are fixed across documents
    - In some settings, we have documents from many different time periods
    - The assumption that topics are fixed may not be sensible
    - The dynamic topic model allows topical content to vary smoothly over time
3. **Structural Topic Model (STM)**
    - Social scientists are typically interested in how topics vary with covariates
    - The structural topic model incorporates covariates into the LDA model
    - When estimated without covariates, the STM is the same as the CTM

Correlated Topic Model

- The Dirichlet is a distribution on the simplex (positive vectors that sum to 1).
- It assumes that components are nearly independent.
- In real data, an article about fossil fuels is more likely to also be about geology than about genetics.
- The logistic normal is a distribution on the simplex that can model dependence between components.
- Amend the model so that the logit transformation of the topic-proportion parameters are drawn from a multivariate normal distribution

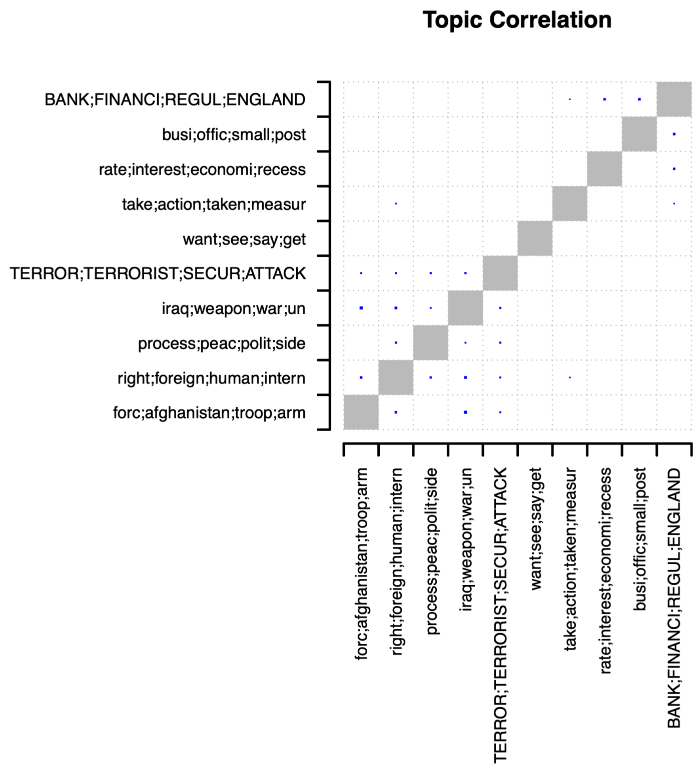
Correlated Topic Model



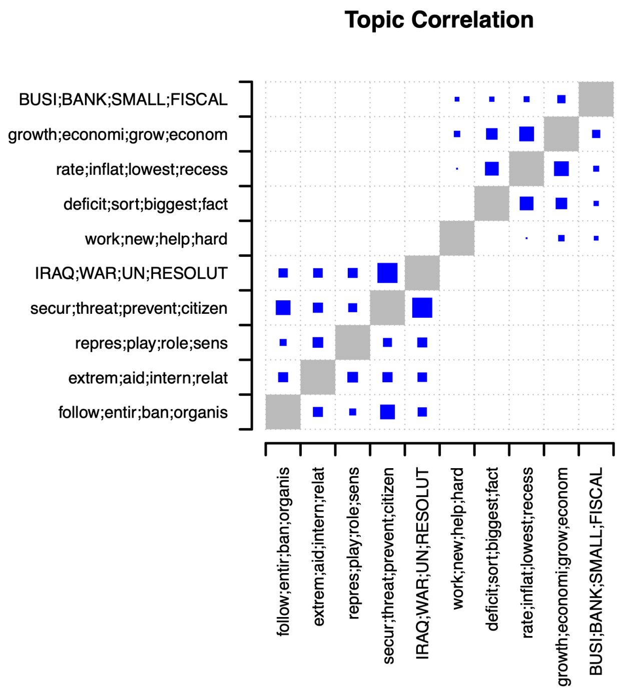
where the first node is logistic normal prior.

- Draw topic proportions from a logistic normal.
- This allows topic occurrences to exhibit correlation.
- Provides a “map” of topics and how they are related
- Provides a better fit to text data, but computation is more complex

LDA topic correlation



CTM topic correlation



CTM pros and cons

Advantages:

1. More reasonable approximation of the “true” data generating process of documents
2. Possible that correlations between topics might be a quantity of interest
3. CTM tends to have better statistical fit to data than LDA

Disadvantages:

1. CTM is somewhat more computationally demanding than LDA
2. CTM tends to have lower topic interpretability than LDA

Dynamic Topic Model

- LDA assumes that the order of documents does not matter.
- Not appropriate for sequential corpora (e.g., that span hundreds of years)
- We may want to track how language changes over time.
  - How has the language used to describe neuroscience developed from “The Brain of Professor Laborde” (1903) to “Reshaping the Cortical Motor Map by Unmasking Latent Intracortical Connections” (1991)
  - How has the language used to describe love developed from “Pride and Prejudice” (1813) to “Eat, Pray, Love” (2006)
- Dynamic topic models let the topics drift in a sequence.

Dynamic Topic Model

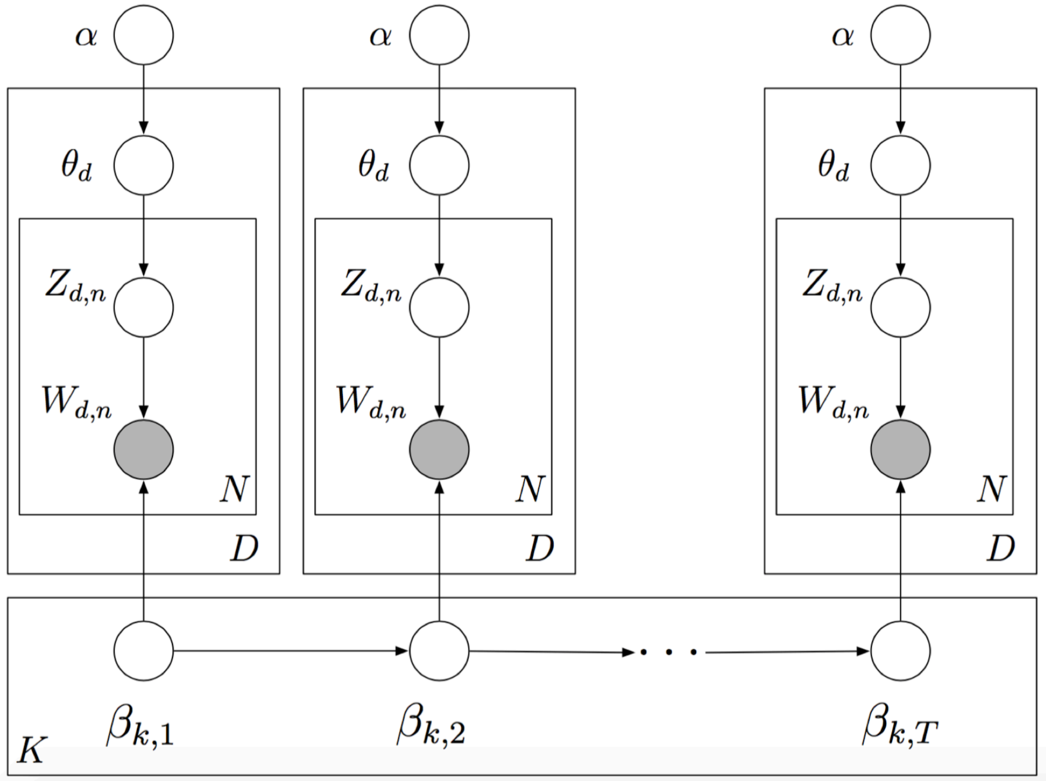
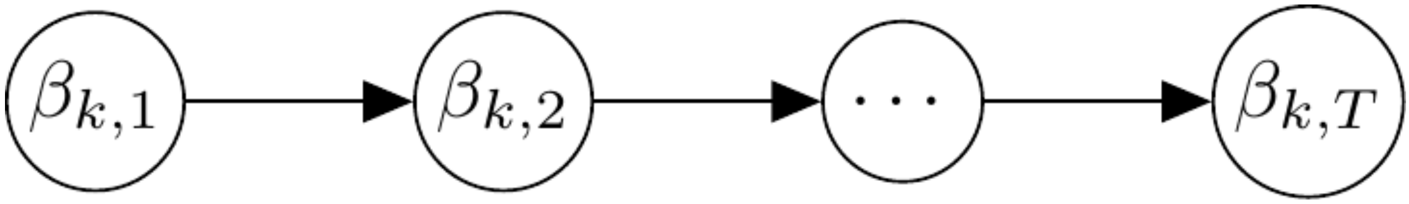


Plate (K) allows topics to “drift” through time.

Dynamic Topic Models



- Use a logistic normal distribution to model topics evolving over time.
    - The k th topic at time 2 has evolved smoothly from the k th topic at time 1
- As for CTMs, this makes computation more complex. But it lets us make inferences about sequences of documents.

Dynamic Topic Model Example (Mimno and Lafferty, 2006)

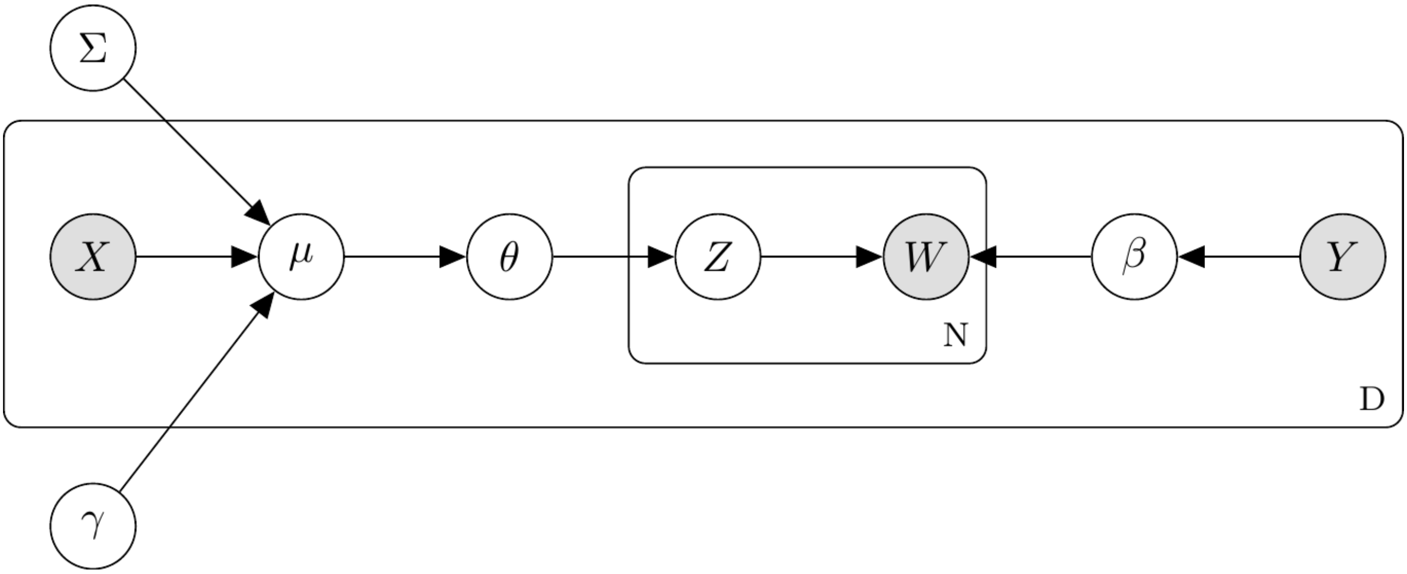
"Neuroscience" topic based on DTM of 30,000 articles from *Science*

Structural Topic Model (STM)

Structural Topic Model

- Typically, when estimating topic models we are interested in how some covariate is associated with the prevalence of topic usage (Gender, date, political party, etc)
- The Structural Topic Model (STM) allows for the inclusion of arbitrary covariates of interest into the generative model
- **Topic prevalence** is allowed to vary according to the covariates X
  - Each document has its own prior distribution over topics, which is defined by its covariates, rather than sharing a global mean
- **Topical content** can also vary according to the covariates Y
  - Word use *within* a topic can differ for different groups of speakers/writers

Structural topic model



Structural Topic Model Application

- In the legislative domain, we might be interested in the degree to which MPs from different parties represent distinct interests in their parliamentary questions

- We can use the STM to analyse how topic prevalence varies by party

- Specify a linear model with:
  - the topic proportions of speech d, by legislator i as the outcome
  - the party of legislator i as the predictor

$$\theta_{dk} = \alpha + \gamma_{1k} * \text{labour}_{d(i)}$$

- The γ_k coefficients give the estimated difference in topic proportions for Labour and Conservative legislators for each topic

Structural Topic Model Application

```r
library(stm)

## Estimate STM
stmOut <- stm(
            documents = pmq_dfm,
            prevalence = ~party.reduced,
            K = 30,
            seed = 123
            )

save(stmOut, file = "stmOut.Rdata")
```

Structural Topic Model Application

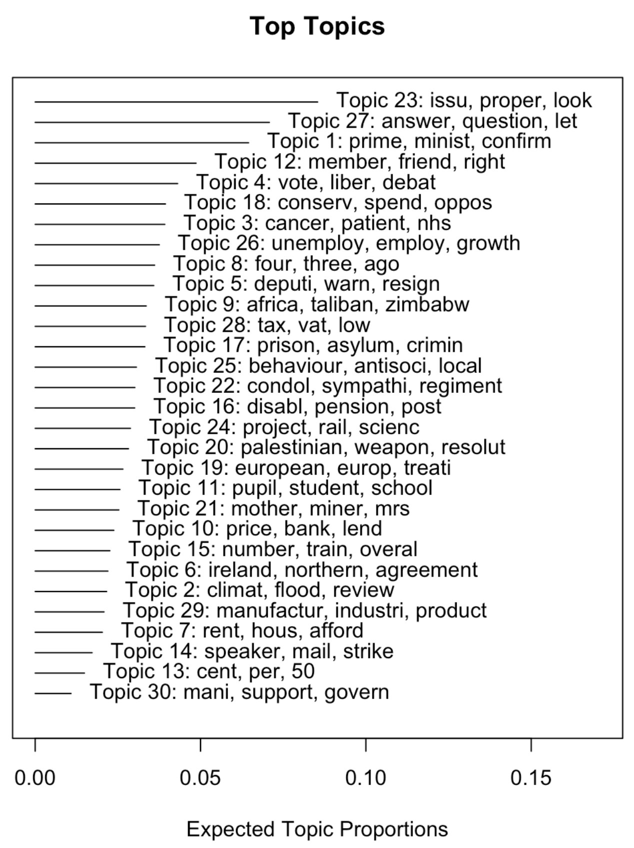
```
labelTopics(stmOut)
```

```
Topic 1 Top Words:
     Highest Prob: minist, prime, govern, s, tell, confirm, ask
     FREX: prime, minist, confirm, failur, paymast, lack, embarrass
     Lift: protectionist, roadshow, harrison, booki, arrog, googl, pembrokeshir
     Score: prime, minist, s, confirm, protectionist, govern, tell
Topic 2 Top Words:
     Highest Prob: chang, review, made, target, fund, meet, depart
     FREX: climat, flood, review, chang, environ, emiss, carbon
     Lift: 2050, consequenti, parrett, dredg, climat, greenhous, barnett
     Score: chang, flood, climat, review, target, environ, emiss
Topic 3 Top Words:
     Highest Prob: servic, health, nhs, care, hospit, nation, wait
     FREX: cancer, patient, nhs, health, hospit, gp, doctor
     Lift: horton, scotsman, wellb, clinician, herceptin, polyclin, healthcar
     Score: health, nhs, servic, hospit, cancer, patient, nurs
Topic 4 Top Words:
     Highest Prob: decis, vote, made, parti, elect, propos, debat
     FREX: vote, liber, debat, scottish, decis, recommend, scotland
     Lift: calman, gould, imc, wakeham, in-built, ipsa, jenkin
     Score: vote, democrat, decis, parti, debat, liber, elect
Topic 5 Top Words:
     Highest Prob: secretari, said, state, last, week, inquiri, report
     FREX: deputi, warn, resign, inquiri, alleg, statement, servant
```

- `Highest Prob` is the raw β coefficients
- `Score` is the term-score measure we defined above
- `FREX` is a measure which combines word-topic frequency with word-topic exclusivity
- `Lift` is a normalised version of the word-probabilities

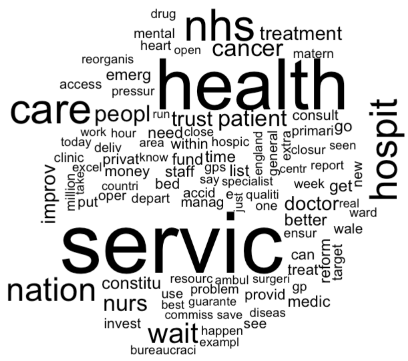
Structural Topic Model Application

```
plot(stmOut, labeltype = "frex")
```



Structural Topic Model Application

```r
cloud(stmOut, topic = 3)
```



Structural Topic Model Application

```r
findThoughts(model = stmOut,
             texts = texts(pmq_corpus),
             topic = 3)
```

```
 Topic 3:
     I suspect that many Members from all parties in this House will agree that mental health services have for too long been treated as a poor cousin a Cinderella service in the NHS and have been systematically underfunded for a long time. That is why I am delighted to say that the coalition Government have announced that we will be introducing new access and waiting time standards for mental health conditions such as have been in existence for physical health conditions for a long time. Over time, as reflected in the new NHS mandate, we must ensure that mental health is treated with equality of resources and esteem compared with any other part of the NHS.
    I am sure that the Prime Minister will join me in congratulating Cheltenham and Tewkesbury primary care trust on never having had a financial deficit and on living within its means. Can he therefore explain to the professionals, patients and people of Cheltenham why we are being rewarded with the closure of our 10-year-old purpose-built maternity ward, the closure of our rehabilitation hospital, cuts in health promotion, cuts in community nursing, cuts in health visiting, cuts in access to acute care and the non-implementation of new NICE-prescribed drugs such as Herceptin?
    I am sure that the Prime Minister will join me in congratulating Cheltenham and Tewkesbury primary care trust on never having had a financial deficit and on living within its means. Can he therefore explain to the professionals, patients and people of Cheltenham why we are being rewarded with the closure of our 10-year-old purpose-built maternity ward, the closure of our rehabilitation hospital, cuts in health promotion, cuts in community nursing, cuts in health visiting, cuts in access to acute care and the non-implementation of new NICE-prescribed drugs such as Herceptin?
```

Structural Topic Model Application

```r
dim(stmOut$theta)
```

```
[1] 27885    30
```

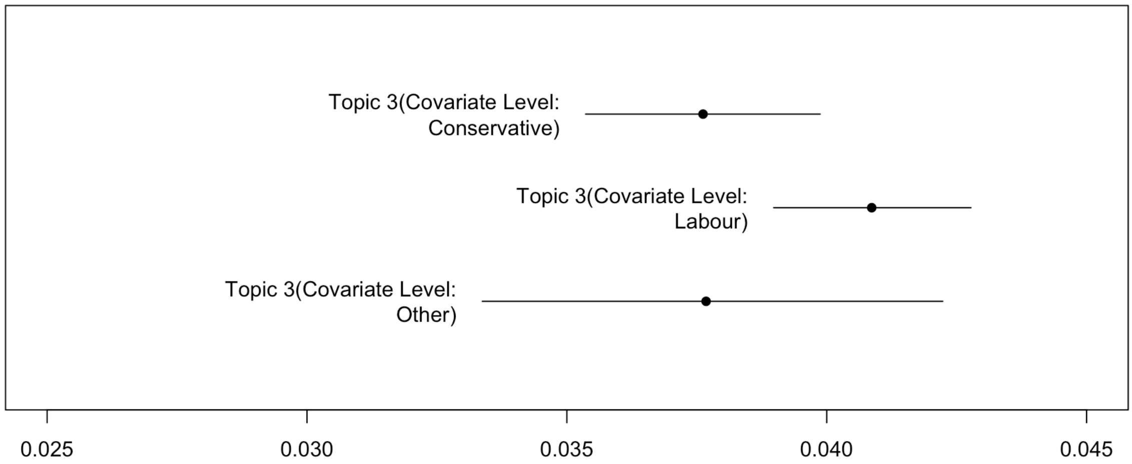
Structural Topic Model Application

Do MPs from different parties speak about healthcare at different rates?

```r
stm_effects <- estimateEffect(formula = c(3) ~ party.reduced,
                              stmobj = stmOut,
                              metadata = docvars(pmq_dfm))

plot.estimateEffect(stm_effects,
                    covariate = "party.reduced",
                    method = "pointestimate",
                    xlim = c(0.025, 0.045))
```

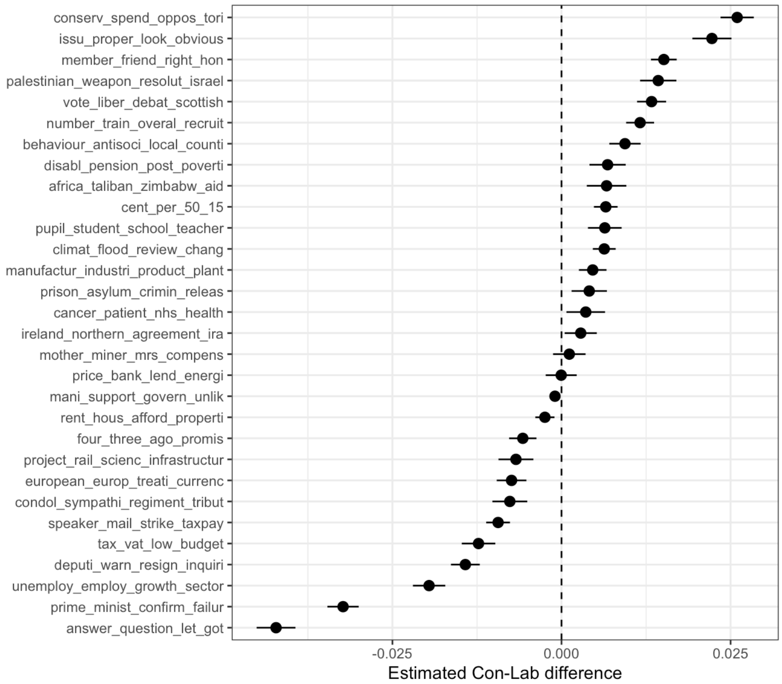
Structural Topic Model Application



Structural Topic Model Application

On which topics do Conservative and Labour MPs differ the most?

```r
stm_effects <- estimateEffect(formula = c(1:30) ~ party.reduced,
                              stmobj = stmOut,
                              metadata = docvars(pmq_dfm))
```



Structural Topic Model Application – Content

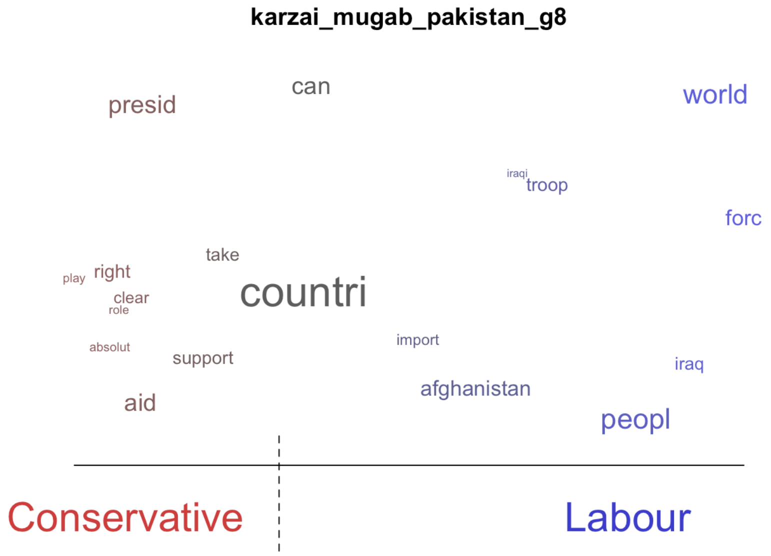
```r
library(stm)

## Estimate STM
stmOut2 <- stm(
            documents = pmq_dfm,
            content = ~party.reduced,
            K = 30,
            seed = 123
            )

save(stmOut2, file = "../data/scripts/stmOut2.Rdata")
```

Structural Topic Model Application – Content

```
plot(stmOut2,
     topics = c(3),
     type = "perspectives",
```



STM Application

> **Do liberal and conservative newspapers report on the economy in different ways?**
>
> Lucy Barnes and Tim Hicks (UCL) study the determinants of voters' attitudes toward government deficits. They argue that individual attitudes are largely a function of media framing. They examine whether and how the Guardian (a left-leaning) and the Telegraph (a right-leaning) report on the economy.

Data and approach:

- $\approx 10,000$ newspaper articles
  - All articles using the word "deficit" from 2010-2015
- STM model
- $K = 6$
  - "We experimented with topic counts up to 20. Six was the value at which the

Validating Topic Models

Validating Topic Models

- LDA, and topic models more generally, require the researcher to make several implementation decisions
- In particular, we must select a value for K , the number of topics
- How can we select between different values of K? How can we tell how well a given topic model is performing?

Validating Topic Models – Quantitative Metrics

- **Held-out likelihood**
  - Ask which words the model believes will be in a given document and comparing this to the document's actual word composition (i.e. calculate the held-out likelihood)
  - E.g. Splitting texts in half, train a topic model on one half, calculate the held-out likelihood for the other half
- **Semantic coherence**
  - Do the most common words from a topic also co-occur together frequently in the same documents?
- **Exclusivity**
  - Do words with high probability in one topic have low probabilities in others?

Problems:

- Prediction is not always important in exploratory or descriptive tasks. We may want

Quantitative Evaluation of STM

We can apply many of these metrics across a range of topic models using the `searchK` function in the `stm` package.

```
search_stm_out <- searchK(documents = pmq_dfm,
                          K = c(5,10,15,20,25,30,35,40),
                          N = 2000)
```

Semantic validity (Chang et al. 2009)

Word intrusion: Test if topics have semantic coherence by asking humans identify a spurious word inserted into a topic.

Topic	w_1	w_2	w_3	w_4	w_5	w_6
1	bank	financ	terror	england	fiscal	market
2	europe	union	eu	referendum	vote	school
3	act	deliv	nhs	prison	mr	right

Assumption: When humans find it easy to locate the "intruding" *word*, the topics are more coherent.

Semantic validity (Chang et al. 2009)

Topic intrusion: Test if the association between topics and documents makes sense by asking humans to identify a topic that was not associated with a document.

> Reforms to the banking system are an essential part of dealing with the crisis, and delivering lasting and sustainable growth to the economy. Without these changes, we will be weaker, we will be less well respected abroad, and we will be poorer.

Topic	w_1	w_2	w_3	w_4	w_5	w_6
1	bank	financ	regul	england	fiscal	market
2	plan	econom	growth	longterm	deliv	sector
3	school	educ	children	teacher	pupil	class

Assumption: When humans find it easy to locate the "intruding" *topic*, the mappings are more sensible.

Semantic validity (Chang et al. 2009)

Conclusion:

> "Topic models which perform better on held-out likelihood may infer less semantically meaningful topics." (Chang et al. 2009.)

Validating Topic Models – Substantive approaches

- *Semantic validity*
  - Does a topic contain coherent groups of words?
  - Does a topic identify a coherent groups of texts that are internally homogenous but distinctive from other topics?
- *Predictive validity*
  - How well does variation in topic usage correspond to known events?
- *Construct validity*
  - How well does our measure correlate with other measures?

Implication: All these approaches require careful human reading of texts and topics, and comparison with sensible metadata.

Conclusion

Summing Up

- Topic models offer an approach to automatically inferring the substantive themes that exist in a corpus of texts
- A topic is described as a probability distribution over words in the vocabulary
- Documents are described as a mixture of corpus wide topics
- Topic models require very little up-front effort, but require extensive interpretation and validation