Week 7: Text Analytics

Slava Mikhaylov

POLS3003 Data Science and Big Data Analytics

Week 7 Outline

Text as data

- Vector Space Model

Text Classification Models

- Naive Bayes Text Classifier
- Application 1
- Correspondence Analysis for Text Classification
- Application 2

Text Clustering

- Application 3

Text as data

- Text is ubiquitous in every aspect of our lives.
- From business, to government, to every day communication: Emails, Twitter, Facebook, news media, government reports, speeches, shareholder reports, product reviews, etc.
- Our goal is to generate insights and intelligence from these raw, unstructured (or semi-structured) data. In other words, we're in for knowledge discovery.
- In order to achieve that we need to make documents computable.

University College London

From Wikipedia, the free encyclopedia

Coordinates: 51°31′29.24″N 00°08′00.88″W

University College London (**UCL**) is a public research university in London, England and a constituent college of the federal University of London. Established in 1826 as **London University** by founders inspired by the radical ideas of Jeremy Bentham, UCL was the first university institution established in London and the earliest in England to be entirely secular, to admit students regardless of their religion and to admit women on equal terms with men.[5] UCL became one of the two founding colleges of the University of London in 1836 and has grown through mergers, including with the Institute of Neurology (in 1997), the Royal Free Hospital Medical School (in 1998), the Eastman Dental Institute (in 1999), the School of Slavonic and East European Studies (in 1999), the School of Pharmacy (in 2012) and the Institute of Education (in 2014). UCL is the largest higher education institution in London and the largest postgraduate institution in the UK by enrollment[6] and is regarded as one of the leading multidisciplinary research universities in the world.[7][8][9][10][11]

UCL's main campus is located in the Bloomsbury area of central London, with a number of institutes and teaching hospitals elsewhere in central London and satellite campuses in Adelaide, Australia and Doha, Qatar. UCL is organised into 11 constituent faculties, within which there are over 100 departments, institutes and research centres. UCL is responsible for several museums and collections in a wide range of fields, including the Petrie Museum of Egyptian Archaeology and the Grant Museum of Zoology and Comparative Anatomy. As of 2014, UCL has around 28,000 students and 11,000 staff (including around 6,000 academic staff and 980 professors) and had a total income of £1.18 billion in 2014/15, of which £427.5 million was from research grants and contracts.[1] UCL is a member of numerous academic organisations and is part of UCL Partners, the world's largest academic health science centre,[12] and the 'golden triangle' of elite English universities.[13]

UCL is one of the most selective British universities and ranks highly in national and international league tables.[11][14][15][16][17] UCL's graduates are ranked among the most employable by international employers[18][19] and its alumni include the "Father of the Nation" of each of India, Kenya and Mauritius, founders of Ghana,[20] modern Japan[21] and Nigeria,[22] the inventor of the telephone, and one of the co-discoverers of the structure of DNA. UCL academics have contributed to major advances in several disciplines; all five of the naturally-occurring noble gases were discovered at UCL by William Ramsay,[23] the vacuum tube was invented by UCL graduate John Ambrose Fleming while a faculty of UCL[24] and several foundational advances in modern statistics were made at UCL's statistical science department founded by Karl Pearson.[25] There are 33 Nobel Prize winners and three Fields Medalists amongst UCL's alumni and current and former

University College London

UCL logo since 2005

Motto	*Cuncti adsint meritaeque expectent praemia palmae* (Latin)
Motto in English	"Let all come who by merit deserve the most reward"
Established	1826
Type	Public research university
Endowment	£103.6 million (at 31 July 2015)[1]
Chancellor	The Princess Royal (as Chancellor of the University of London)
Provost	Michael Arthur
Chairman of the Council	Dame DeAnne Julius[2]
Academic staff	6,000
Administrative staff	11,000
Students	28,430 (2013/14)[3]
Undergraduates	15,415 (2013/14)[3]
Postgraduates	13,015 (2013/14)[3]
Location	London, United Kingdom

How to represent a document

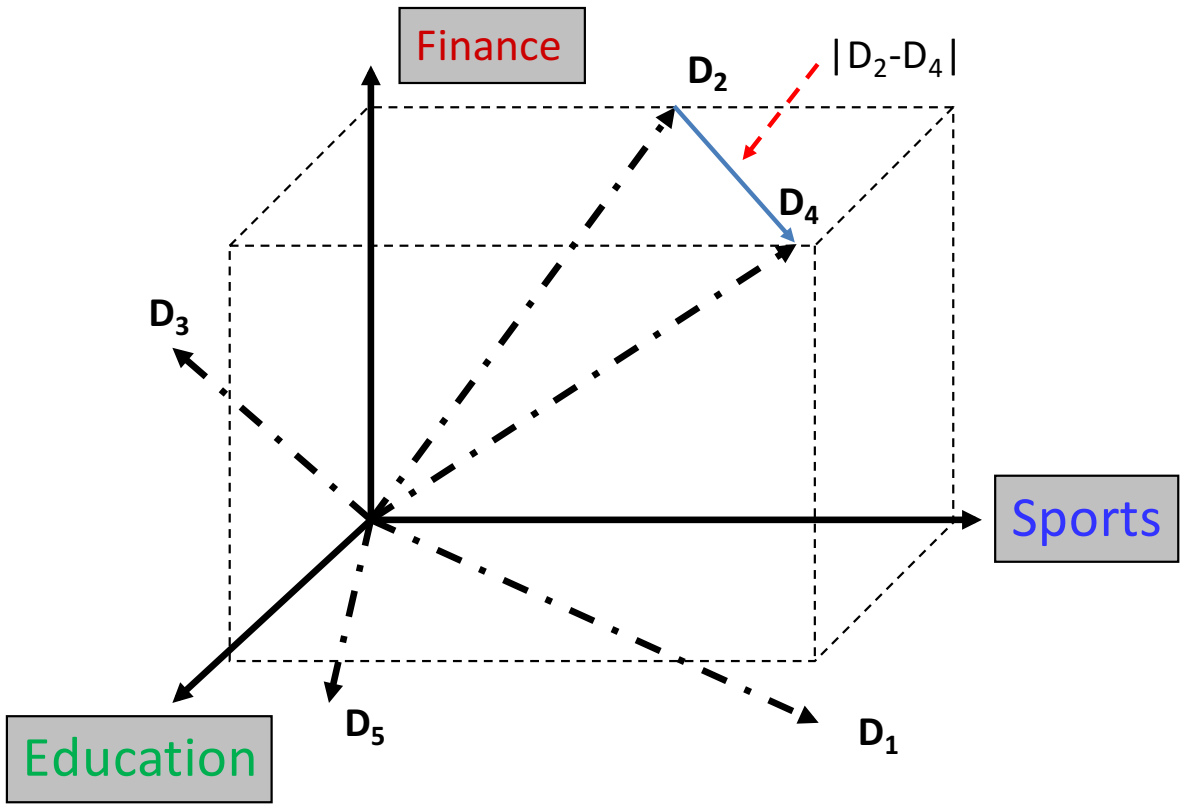
- We cannot represent by a string as it has no semantic meaning.
- We cannot represent by a list of sentences since a sentence is just like a short document (recursive definition).

Vector Space Model

- Represent documents by concept vectors
  - Each concept defines one dimension
  - k concepts define a high-dimensional space
  - Element of vector corresponds to concept weight: e.g. $d = (x_1, \ldots, x_k)$, x_i is "importance" of concept i in d.
- Distance between the vectors in this concept space – relationship among documents.

An illustration of VS model

All documents are projected into this concept space



What the VS model doesn't say

- How to define/select the "basic concept"
  - Concepts are assumed to be orthogonal
- How to assign weights
  - Weights indicate how well the concept characterizes the document
- How to define the distance metric

What is a good "Basic Concept"?

- Orthogonal
  - Linearly independent basis vectors: "Non-overlapping" in meaning
  - No ambiguity
- Weights can be assigned automatically and accurately
- Existing solutions
  - Terms or N-grams, a.k.a., Bag-of-Words
  - Topics

Bag-of-Words representation

Term as the basis for vector space

- Doc1: Text mining is to identify useful information.
- Doc2: Useful information is mined from text.
- Doc3: Apple is delicious.

	text	information	identify	mining	mined	is	useful	to	from	apple	delicious
Doc1	1	1	1	1	0	1	1	1	0	0	0
Doc2	1	1	0	0	1	1	1	0	1	0	0
Doc3	0	0	0	0	0	1	0	0	0	1	1

Tokenization

Break a stream of text into meaningful units

- Tokens: words, phrases, symbols

  Input: It's not straight-forward to perform so-called "tokenization."

  Output(1): 'It's', 'not', 'straight-forward', 'to', 'perform', 'so-called', '"tokenization".'

  Output(2): 'It', '"', 's', 'not', 'straight', '-', 'forward', 'to', 'perform', 'so', '-', 'called', '"', 'tokenization', '.', '"'
- Definition depends on language, corpus, or even context.
- We usually approach tokenization through regular expressions or use of statistical methods (Apache OpenNLP, Stanford NLP Parser).

Bag-of-Words representation

	text	information	identify	mining	mined	is	useful	to	from	apple	delicious
Doc1	1	1	1	1	0	1	1	1	0	0	0
Doc2	1	1	0	0	1	1	1	0	1	0	0
Doc3	0	0	0	0	0	1	0	0	0	1	1

- Assumption: Words are independent from each other
- Pros: Simple
- Cons: Basis vectors are clearly not linearly independent! Grammar and order are missing.
- The most frequently used document representation (also used in image, speech, gene sequence).

Bag-of-Words with N-grams

- N-grams: a contiguous sequence of n tokens from a given piece of text
  - E.g., 'Text mining is to identify useful information.'
  - Bigrams: 'text_mining', 'mining_is', 'is_to', 'to_identify', 'identify_useful', 'useful_information', 'information_.'
- Pros: capture local dependency and order
- Cons: makes computation more complicated due to term sparseness.

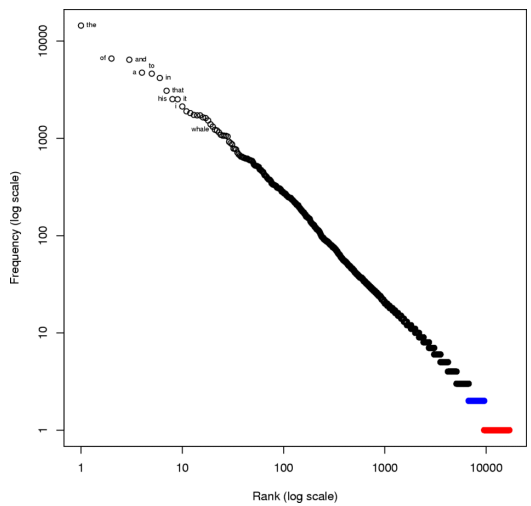
Automatic document representation

Represent a document with all the occurring words

- Pros: Preserve all information in the text (hopefully). Also, it's fully automatic.
- Cons: vocabulary gap – cars vs car, talk vs talking. Leads to large storage demands on computing.
- Solution: Construct controlled vocabulary.

Statistical properties of language

- Rank-frequency plot for words in the novel Moby-Dick.
- About 44% of the distinct set of words in this novel, such as "matrimonial", occur only once, so called hapax legomena (red).
- About 17%, such as "dexterity", appear twice (so-called dis legomena, in blue).
- Zipf's law predicts that the words in this plot should approximately fit a straight line – frequency of any word is inversely proportional to its rank in the frequency table.



https://en.wikipedia.org/wiki/Hapax_legomenon

Controlled vocabulary

- Head words take large portion of occurrences, but they are semantically meaningless. E.g., the, a, an, we, do, to.
- Tail words take major portion of vocabulary, but they rarely occur in documents. E.g., matrimonial in Moby-Dick.
- The rest is most representative, and should be included in the controlled vocabulary.

Normalization

- Convert different forms of a word to a normalized form in the vocabulary. For example, U.S.A. to USA, St. Louis to Saint Louis.
- Solution
  - Rule-based
    - Delete periods and hyphens,
    - All in lower cases.
  - Dictionary-based
    - Construct equivalent class: Car to "automobile, vehicle"; Mobile phone to "cellphone".

Stemming

Lemmatization refers to the algorithmic process of converting words to their lemma forms.

stemming the process for reducing inflected (or sometimes derived) words to their stem, base or root form. Different from *lemmatization* in that stemmers operate on single words without knowledge of the context.

both convert the morphological variants into stem or root terms

example: `produc` from
`production`, `producer`, `produce`, `produces`, `produced`

Stopwords

Useless words for document analysis

- Not all words are informative.
- Remove such words to reduce vocabulary size.
- No universal definition.
- We risk breakingthe original meaning and structure of text.

Common English stop words

```
a, able, about, across, after, all, almost, also, am, among,
an, and, any, are, as, at, be, because, been, but, by, can,
cannot, could, dear, did, do, does, either, else, ever,
every, for, from, get, got, had, has, have, he, her, hers,
him, his, how, however, I, if, in, into, is, it, its, just,
least, let, like, likely, may, me, might, most, must, my,
neither, no, nor, not, of, off, often, on, only, or, other,
our, own, rather, said, say, says, she, should, since, so,
some, than, that, the, their, them, then, there, these,
they, this, tis, to, too, twas, us, wants, was, we, were,
what, when, where, which, while, who, whom, why, will, with,
would, yet, you, your
```

- But no list should be considered universal

A more comprehensive list of stop words

a's, able, about, above, according, accordingly, across, actually, after, afterwards, again, against, ain't, all, allow, allows, almost, alone, along, already, also, although, always, am, among, amongst, an, and, another, any, anybody, anyhow, anyone, anything, anyway, anyways, anywhere, apart, appear, appreciate, appropriate, are, aren't, around, as, aside, ask, asking, associated, at, available, away, awfully, be, became, because, become, becomes, becoming, been, before, beforehand, behind, being, believe, below, beside, besides, best, better, between, beyond, both, brief, but, by, c'mon, c's, came, can, can't, cannot, cant, cause, causes, certain, certainly, changes, clearly, co, com, come, comes, concerning, consequently, consider, considering, contain, containing, contains, corresponding, could, couldn't, course, currently, definitely, described, despite, did, didn't, different, do, does, doesn't, doing, don't, done, down, downwards, during, each, edu, eg, eight, either, else, elsewhere, enough, entirely, especially, et, etc, even, ever, every, everybody, everyone, everything, everywhere, ex, exactly, example, except, far, few, fifth, first, five, followed, following, follows, for, former, formerly, forth, four, from, further, furthermore, get, gets, getting, given, gives, go, goes, going, gone, got, gotten, greetings, had, hadn't, happens, hardly, has, hasn't, have, haven't, having, he, he's, hello, help, hence, her, here, here's, hereafter, hereby, herein, hereupon, hers, herself, hi, him, himself, his, hither, hopefully, how, howbeit, however, i'd, i'll, i'm, i've, ie, if, ignored, immediate, in, inasmuch, inc, indeed, indicate, indicated, indicates, inner, insofar, instead, into, inward, is, isn't, it, it'd, it'll, it's, its, itself, just, keep, keeps, kept, know, knows, known, last, lately, later, latter, latterly, least, less, lest, let, let's, like, liked, likely, little, look, looking, looks, ltd, mainly, many, may, maybe, me, mean, meanwhile, merely, might, more, moreover, most, mostly, much, must, my, myself, name, namely, nd, near, nearly, necessary, need, needs, neither, never, nevertheless, new, next, nine, no, nobody, non, none, noone, nor, normally, not, nothing, novel, now, nowhere, obviously, of, off, often, oh, ok, okay, old, on, once, one, ones, only, onto, or, other, others, otherwise, ought, our, ours, ourselves, out, outside, over, overall, own, particular, particularly, per, perhaps, placed, please, plus, possible, presumably, probably, provides, que, quite, qv, rather, rd, re, really, reasonably, regarding, regardless, regards, relatively, respectively, right, said, same, saw, say, saying, says, second, secondly, see, seeing, seem, seemed, seeming, seems, seen, self, selves, sensible, sent, serious, seriously, seven, several, shall, she, should, shouldn't, since, six, so, some, somebody, somehow, someone, something, sometime, sometimes, somewhat, somewhere, soon, sorry, specified, specify, specifying, still, sub, such, sup, sure, t's, take, taken, tell, tends, th, than, thank, thanks, thanx, that, that's, thats, the, their, theirs, them, themselves, then, thence, there, there's, thereafter, thereby, therefore, therein, theres, thereupon, these, they, they'd, they'll, they're, they've, think, third, this, thorough, thoroughly, those, though, three, through, throughout, thru, thus, to, together, too, took, toward, towards, tried, tries, truly, try, trying, twice, two, un, under, unfortunately, unless, unlikely, until, unto, up, upon, us, use, used, useful, uses, using, usually, value, various, very, via, viz, vs, want, wants, was, wasn't, way, we, we'd, we'll, we're, we've, welcome, well, went, were, weren't, what, what's, whatever, when, whence, whenever, where, where's, whereafter, whereas, whereby, wherein, whereupon, wherever, whether, which, while, whither, who, who's, whoever, whole, whom, whose, why, will, willing, wish, with, within, without, won't, wonder, would, would, wouldn't, yes, yet, you, you'd, you'll, you're, you've, your, yours, yourself, yourselves, zero

Constructing a VSM representation

1. Tokenization;
2. Stemming/normalization;
3. N-gram construction;
4. Stopword/controlled vocabulary filtering.

Assigning weights

- Corpus-wise: some terms carry more information about the document content.
- Document-wise: not all terms are equally important.
- Two basic heuristics: TF (Term Frequency) is the Within-doc-frequency; IDF (Inverse Document Frequency).

Term Frequency normalization

- Idea: a term is more important if it occurs more frequently in a document.
- Two views of document length:
  - A doc is long because it is verbose.
  - A doc is long because it has more content.
- Raw TF is inaccurate
  - Document length variation.
  - "Repeated occurrences" are less informative than the "first occurrence."
  - Semantic information does not increase proportionally with number of term occurrences.
- Generally penalize long document, but avoid over-penalizing – pivoted length normalization

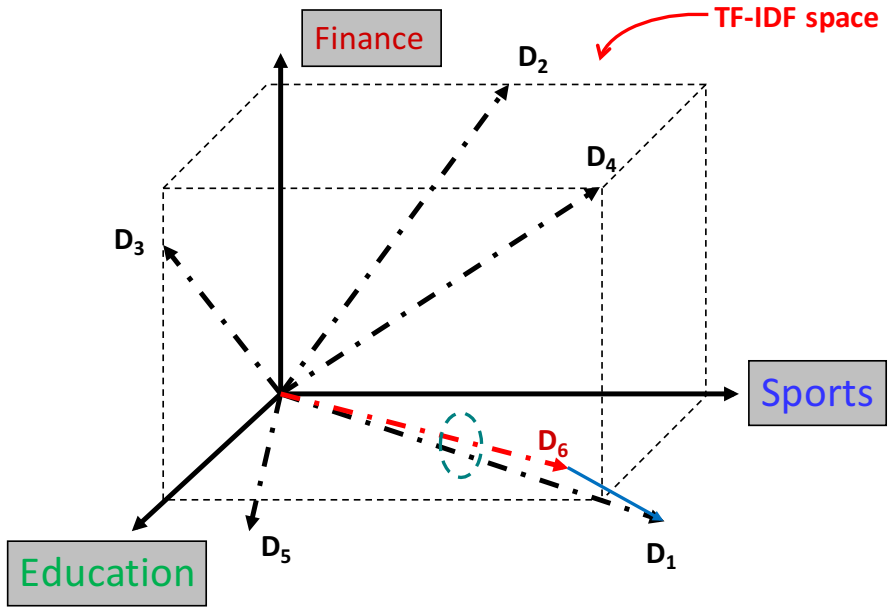
Document Frequency normalization

- Idea: a term is more discriminative if it occurs only in fewer documents.
- Inverse document frequency: Assign higher weights to the rare terms.
- This is a corpus-specific property. Independent of a single document.

TF-IDF weighting

- Combining TF and IDF
  - Common in document $\rightarrow$ high tf $\rightarrow$ high weight;
  - Rare in collection $\rightarrow$ high idf $\rightarrow$ high weight;
  - $w(t, d) = TF(t, d) \times IDF(t)$
- Most well-known document representation schema in information retrieval.

Defining a similarity metric



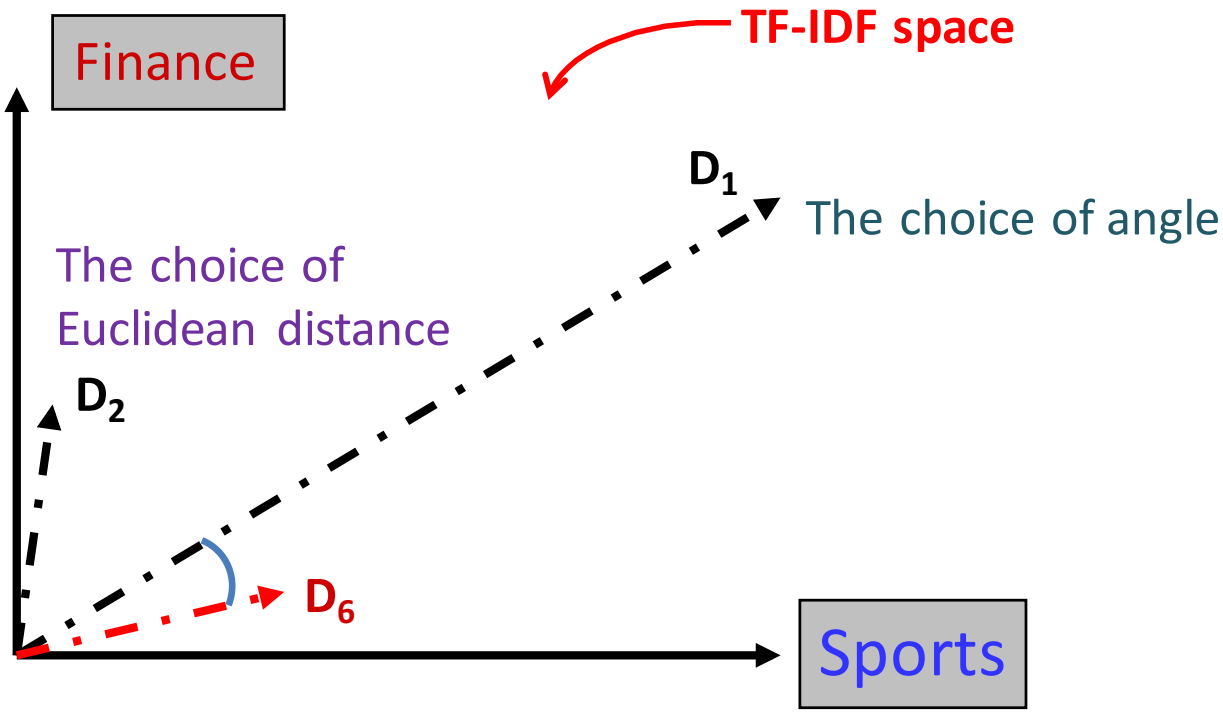
We can use Euclidean distance as a similarity metric:

$$dist(d_i, d_j) = \sqrt{\sum_{t \in V} [tf(t, d_i)idf(t) - tf(t, d_j)idf(t)]^2}$$

- Longer documents will be penalized by the extra words;
- We care more about how these two vectors overlap.

From distance to angle

- Focusing on the angle tells us how vectors overlap.
- Cosine similarity – projection of one vector onto another.



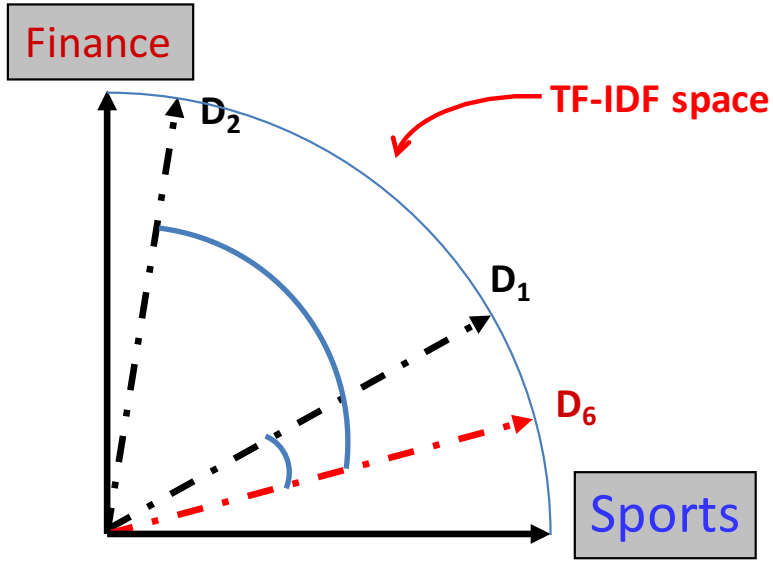
Cosine similarity

- Angle between two vectors

$$cosine(d_i, d_j) = \frac{v_{d_i} \times v_{d_j}}{|v_{d_i}|_2 \times |v_{d_j}|_2}$$

where v_{d_i} is the TF-IDF vector and $\frac{v_{d_j}}{|v_{d_i}|_2}$ is unit vector.

- Documents are normalized by length.



Advantages of VS model

- Empirically effective.
- Intuitive.
- Easy to implement.
- Well-studied/mostly evaluated.
- The Smart system developed at Cornell (1960-1999) but still widely used today.
- Warning: many variants of TF-IDF.

Disadvantages of VS model

- We assume term independence.
- Lack of "predictive adequacy" due to arbitrary term weighting and arbitrary similarity measure.
- Lots of parameter tuning!

What you should take out

- Basic ideas of vector space model.
- Procedures of constructing VS representation for a document.
- Two important heuristics in bag-of-words representation: TF and IDF.
- Similarity metric for VS model.

Text Classification Models

Text Classification

Text classifaction (TC) is applicable to a wide range of problems:

- Detecting document encoding;
- Spam filtering;
- Sentiment detection;
- Document sorting in smart email folders.

Text Classification

- Text has traditionally been classified manually (e.g. a library catalogue).
- However, this is not scalable.
- In machine learning, the decision criterion of the text classifier is learned automatically from training data.
- In such statistical text classification we require a number of good example documents (training documents) for each class.
- Training documents are manually labeled (annotated with class label).

The text classification problem

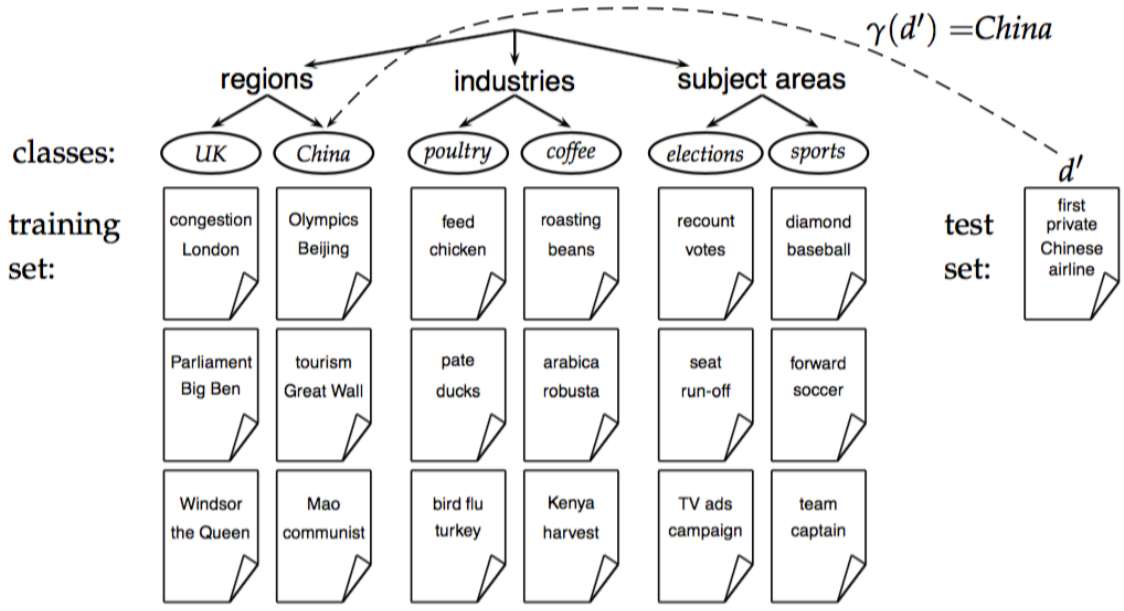
- In TC we are given a description $d \in X$ of a document, where X is the document space; and a fixed set of classes $C = \{c_1, c_2, \ldots, c_J\}$.
- Classes are also called categories or labels.
- The document space X is a high-dimensional space, and the classes are human defined for the need of an application.
- We are also given a training set D of labeled documents $\langle d, c \rangle \in X \times C$.
- For example:

$$\langle d, c \rangle = \langle \textit{Beijing joins the World Trade Organization}, \textit{China} \rangle$$

for the one-sentence document *Beijing joins the World Trade Organization* and the class (or label) *China*.

The text classification problem

- Function that maps documents to classes is a classifier $\gamma : X \rightarrow C$.
- We are using supervised learning here since a human needs to define the classes and label training documents.
- We apply the classifer to the test set, whose class is unknown. Our goal is to achieve high accuracy on the test data.
- In order to achieve that we have to make the assumption that training data and test data are similar or from the same distribution.



► **Figure 13.1** Classes, training set, and test set in text classification .

Naive Bayes text classifier

- Multinomial Naive Bayes (multinomial NB) is a probabilistic learning method, an extention of what we covered when we discussed classification models.
- The probability of a document d being in class c is computed as

$$P(c|d) \propto P(c) \prod_{1 \leq k \leq n_d} P(t_k|c)$$

  where $P(t_k|c)$ is the conditional probability of term t_k occurring in a document of class c.
- $P(t_k|c)$ can be interpreted as a measure ofhow much evidence t_k contributes that c is the correct class.
- $P(c)$ is the prior probability of a document occurring in class c.
- $\langle t_1, t_2, \ldots, t_{n_d} \rangle$ are the tokens in d that are part of the vocabulary we use for classification and n_d is the number of such tokens in d.

Maximum a posteriori class

- Our goal is to find the best class for the document.
- The best class in NB classification is the most likely or maximum a posteriori (MAP) class c_{map}:

$$c_{map} = \underset{c \in C}{argmax} \hat{P}(c|d) = \underset{c \in C}{argmax} \hat{P}(c) \prod_{1 \leq k \leq n_d} \hat{P}(t_k|c).$$

Underflow

- In the function above, many conditional probabilities are multiplied, one for each position $1 \leq k \leq n_d$.
- This can result in a floating point underlow (the result of a calculation is a smaller number than the computer can actually store in memory).
- Thus it's better to add logarithms of probabilities instead of multiplying probabilities.
- The class with the highest log probability score is still the most probable

$$log(xy) = log(x) + log(y)$$

  and the logarithm function is monotonic.
- NB maximization is then

$$c_{map} = \underset{c \in C}{argmax}[log\,\hat{P}(c) + \sum_{1 \leq k \leq n_d} log\,\hat{P}(t_k|c)].$$

Interpreting NB

- Each conditional parameter $log\,\hat{P}(t_k|c)$ is a weight that indicates how good an indicator t_k is for c.
- The prior $log\,\hat{P}(c)$ is a weight that indicates the relative frequency of c.
- More frequent classes are more likely to be the correct class than infrequent classes.
- The sum of log prior and term weights is then a measure of how much evidence there is for the document being in the class.
- c_{map} equation selects the class for which we have the most evidence.

Estimating NB model

- We can use MLE to estimate the parameters in the model. Here it's the relative frequency and corresponds to the most likely value of each parameter given the training data.
- For the priors this is:

$$\hat{P}(c) = \frac{N_c}{N},$$

where N_c is the number of documents in class c and N is the total number of documents.

- Conditional probability

$$\hat{P}(t|c) = \frac{T_{ct}}{\sum_{t' \in V} T_{ct'}},$$

where T_{ct} is the number of occurances of t in training documents from class c, including multiple occurances of a term in a document.

Positional independence assumption

- Here we made the positional independence assumption: T_{ct} is a count of occurances in all positions k in the documents in the training set.
- We do not compute different estimates for different positions and, for example, if a word occurs twice in a document, in positions k_1 and k_2, then $\hat{P}(t_{k_1}|c) = \hat{P}(t_{k_2}|c)$.

Zero probabilities in MLE

- The problem with the MLE estimate is that it is zero for a termclass combination that did not occur in the training data.
- If the term `WTO` in the training data only occurred in `China` documents, then the MLE estimates for the other classes, for example `UK`, will be zero:

$$\hat{P}(WTO|UK) = 0.$$

- Now, the one-sentence document *Britain is a member of the WTO* will get a conditional probability of zero for `UK` because we are multiplying the conditional probabilities for all terms in the NB equation.
- We should have a high probability for the `UK` class because the term `Britain` occurs.
- Zero probability for `WTO` remains regardless of the weight of evidence for the `UK` from other features.

Sparseness

- The estimate is 0 because of sparseness.
- The training data is not large enough to capture the frequency of rare events adequately.
- To eliminate zeros we use add-one or Laplace smoothing:

$$\hat{P}(t|c) = \frac{T_{ct} + 1}{\sum_{t' \in V}(T_{ct'+1})} = \frac{T_{ct} + 1}{(\sum_{t' \in V} T_{ct'1}) + B'},$$

where $B = |V|$ is the number of terms in the vocabulary.

Properties of NB

- Class membership of a document is decided based on the class with the maximum a posteriori probability computed as:

$$c_{map} = \underset{c \in C}{argmax} P(c|d) = \underset{c \in C}{argmax} \frac{P(d|c)P(c)}{P(d)} = \underset{c \in C}{argmax} P(d|c)P(c),$$

  where to get the last equality we apply Bayes' rule and drop the denominator because $P(d)$ is the same for all classes and does not affect the argmax.
- The last equality can be interpreted as a description of the generative process we assume in Bayesian text classification.

Text generative process

1. Choose class c with probability $P(c)$, where C is a random variable.
2. Generate the document given the class, corresponding to the conditional distribution $P(d|c)$:

$$P(d|c) = P(\langle t_1, \ldots, t_k, \ldots, t_{n_d} \rangle | c)$$

where $\langle t_1, \ldots, t_{n_d} \rangle$ is the sequence of terms as it occurs in d (minus terms that were excluded from the vocabulary).

NB CIA

- The number of parameters in such a model would be huge and we won't be able to perform text classification directly.
- Therefore, we introduce the Naive Bayes conditional independence assumption – attribute values are independent of each other given the class:

$$P(d|c) = P(\langle t_1, \ldots, t_k, \ldots, t_{n_d} \rangle | c) = \prod_{1 \leq k \leq n_d} P(X_k = t_k | c)$$

- X_k is the random variable for position k in the document.
- X_k takes as values terms from the vocabulary.
- $P(X_k = t|c)$ is the probability that in a document of class c the term t will occur in position k.

Positional independence

- Even with the NB CIA assumption there are still too many parameters to estimate.
- The position of a term in a document by itself does not carry information about the class.
- There is a difference between China sues France and France sues China. But the occurrence of China in position 1 versus position 3 of the document is not useful in NB classification because we look at each term separately.
- Also, estimating different set of parameters for each k could lead to problems due to data sparseness.

Positional independence assumption

- Hence we make the Positional Independence Assumption (PIA) – conditional probabilities for a term are the same independent of position in the document:

$$P(X_{k_1} = t|c) = P(X_{k_2} = t|c)$$

  for all positions k_1, k_2, terms t and classes c.
- This leads to a single distribution of terms valid for all positions k_i.
- PIA is equivalent to adopting the bag of words model.

NB as a good classifier

- Naive Bayes gets its name from CIA and PIA being very naive models of natural language.
- Due to such poor models of natural language NB probability estimates are of low quality.
- However, NB classification decisions are surprisingly good.
- NB classification decision is based on which class gets the highest score. It does not matter how accurate the estimates are.
- Correct estimation implies accurate prediction, but accurate prediction does not imply correct estimation.
- NB classifiers estimate badly, but often classify well.

Benefits of NB

- NB is robust to noise features and concept drift.
- Other classifiers (e.g. kNN) may be more accurate but also more tuned to idiosyncratic properties of a particular time period.
- They will suffer if documents in the following time period have slightly different properties.

Benefits of NB

- NB is extremely efficient with training and classification done in one pass over the data.
- NB through its combination of efficiency and good accuracy is often used as a baseline in text classification.
- Where CIA and PIA hold it can be shown that NB is an optimal classifier (minimal test error rate) for text.

Using NB

We often prefer NB when:

- marginal increase in accuracy is not that important for the text classification application at hand,
- a very large amount of training data is available and there is more to be gained from training on a lot of data than using a better classifier on a smaller training set, or
- its robustness to concept drift can be exploited.

$tf_{t_i,d}$ formulation of NB

- A useful alternative formalization of NB treats each document d as an M-dimensional vector of counts $\langle tf_{t_1,d}, \ldots, tf_{t_M,d} \rangle$ where $tf_{t_i,d}$ is the term frequency of t_i in d.
- Then $P(d|c)$ is computed as follows:

$$P(d|c) = P(\langle tf_{t_1,d}, \ldots, tf_{t_M,d} \rangle | c) \propto \prod_{1 \leq i \leq M} P(X = t_i|c)^{tf_{t_i,d}}$$

- This is equivalent to the sequence model in our first formulation of multinomial Naive Bayes earlier as $P(X = t_i|c)^{tf_{t_i,d}} = 1$ for terms that do not occur in d ($tf_{t_i,d} = 0$) and a term that occurs $tf_{t_i,d} \geq 1$ times will contribute $tf_{t_i,d}$ factors in both formulations of multinomial NB.

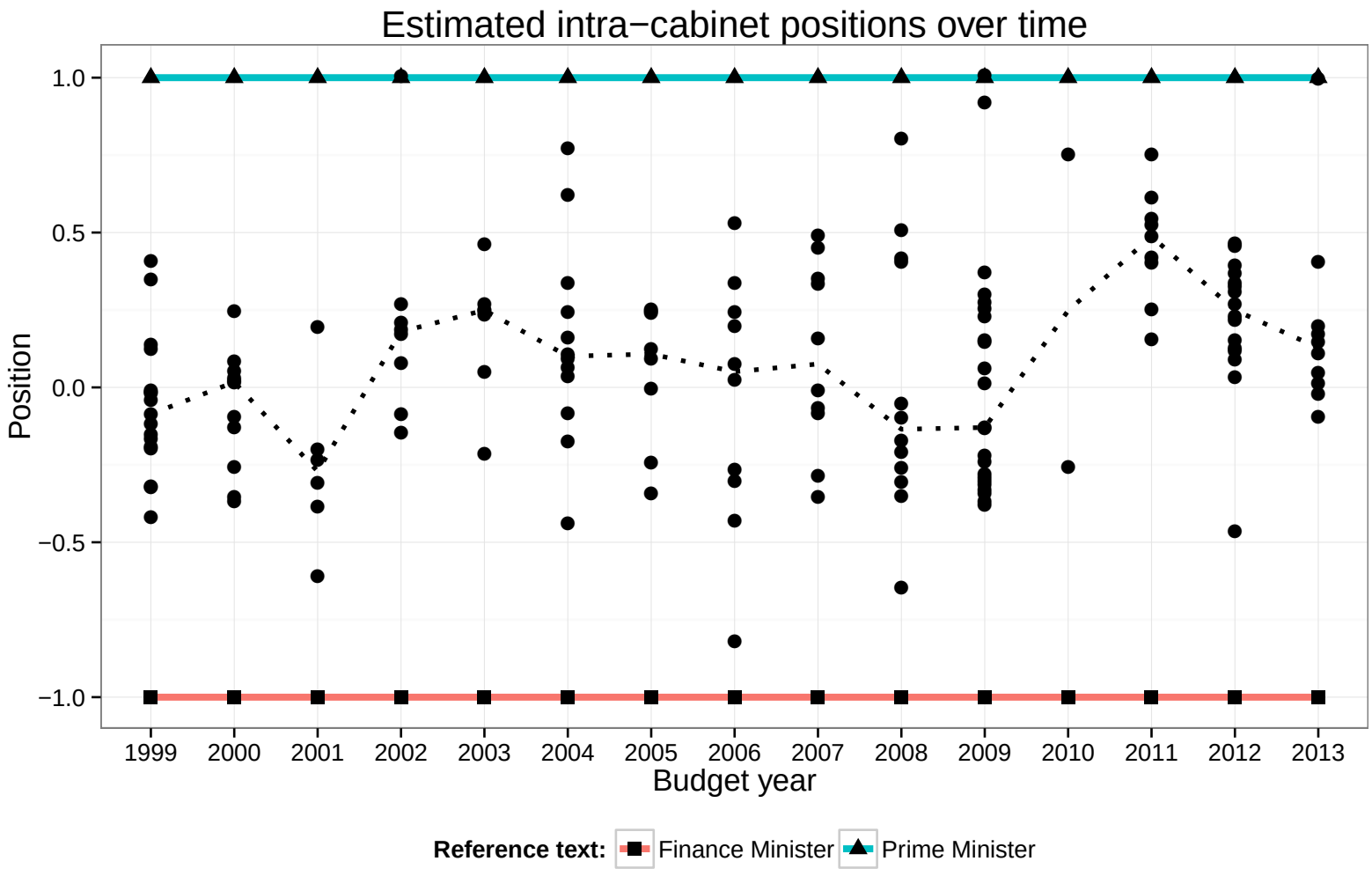
Analyzing intra-cabinet politics with NB

- Difficult to assess intra-cabinet politics. But it's important for many political science models to have estimates of ministers' positions on various policies.
- In Ireland most ministers speak during the budget debates. We can use these speeches to categorize ministers as belonging to one of the two classes (e.g. faction of PM vs faction of FM).
- We can use probabilities of belonging to each class (scores) as measures of positions on a policy dimension defined by the training documents (PM speeches and FM speeches).
- In political science jargon this is often called ideological scaling or position scaling.

NB with political texts

- One "native" implementation of the $tf_{t_i,d}$ formulation of NB is called Wordscore presented in Laver, Michael, Kenneth Benoit and John Garry. 2003. "Extracting Policy Positions from Political Texts Using Words as Data." *American Political Science Review* 97: 311-331.
- Computation implemented in R `quanteda` package: `https://github.com/kbenoit/quanteda`.

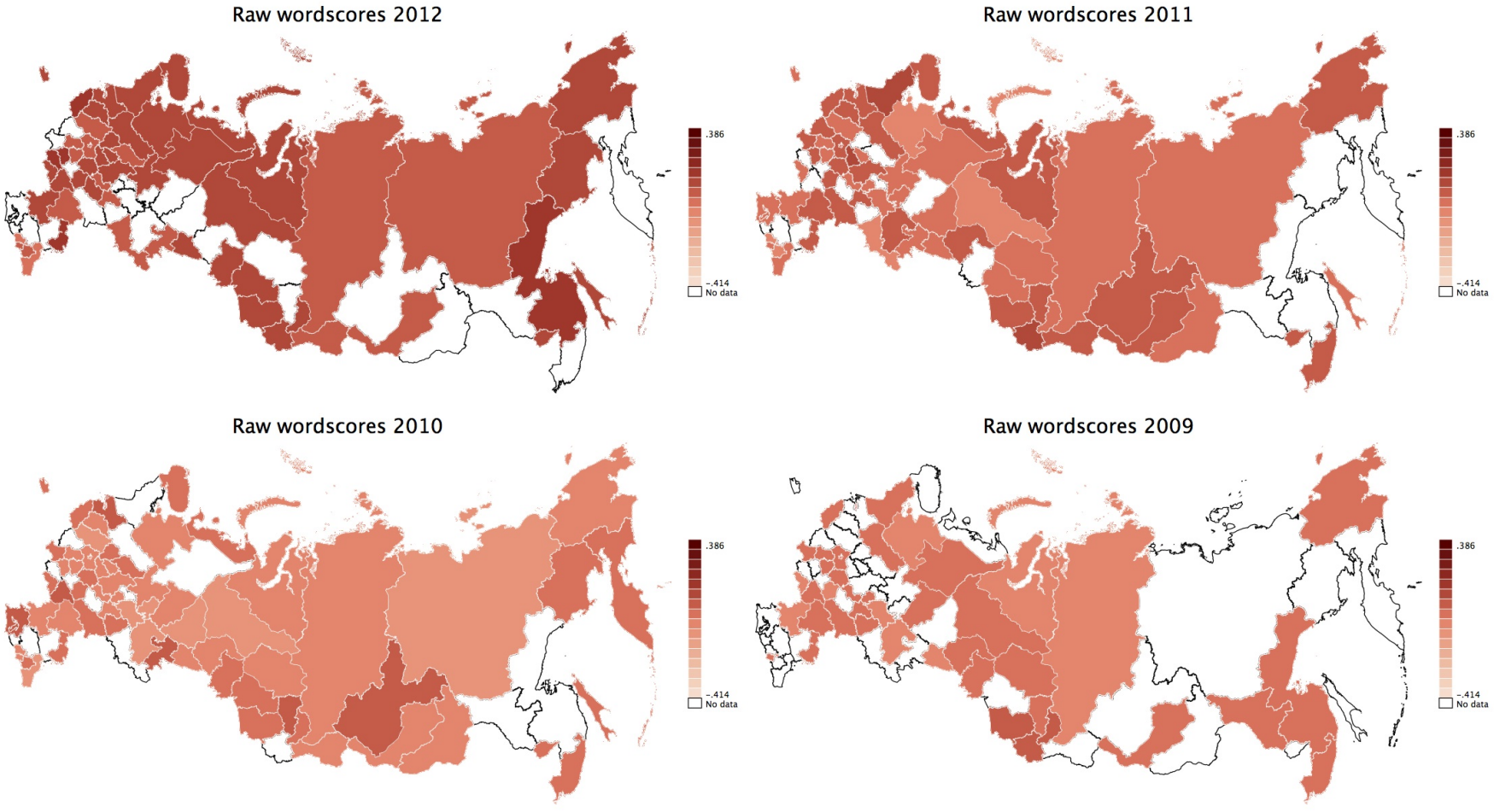
Applying NB to budget debates in Ireland



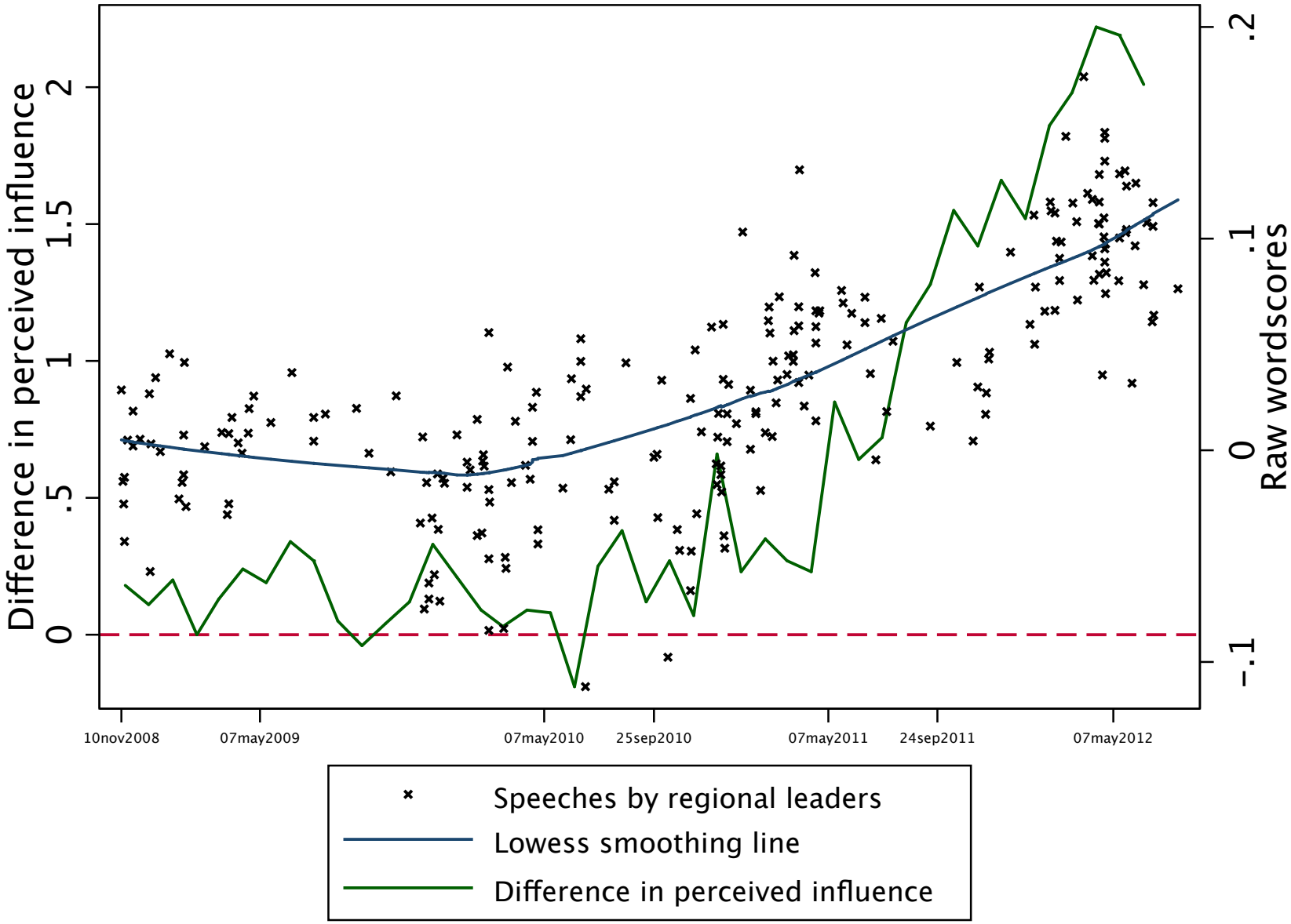
Using NB to analyze Russian Game of Thrones

Baturo, Alexander and Slava Mikhaylov. 2013. "Life of Brian Revisited: Assessing Informational and Non-Informational Leadership Tools." *Political Science Research and Methods* 1(01):139-157.

Positions of Russian governors 2009-2011



Validation


Difference in perceived influence
0
.5
1
1.5
2
Raw wordscores
−.1
0
.1
.2
10nov2008
07may2009
07may2010
25sep2010
07may2011
24sep2011
07may2012
Speeches by regional leaders
Lowess smoothing line
Difference in perceived influence

Correspondence Analysis for text classification

- Will Lowe (2008) showed that main text scaling models popular in political science (including Wordscore) can be formalized as special cases of correspondence analysis.
- Lowe, William (2008). "Understanding Wordscores." *Political Analysis* 16(4): 356-371.
- The assumptions of correspondence analysis fit nicely with natural language properties and model outputs interesting for political science.

Correspondence Analysis

- CA is an SVD technique, just like PCA.
- It replaces the Euclidean metric with Chi-square metric, and weights the sites and species by their totals.
- The differences may sound small and technical, but in practice, the difference is huge.

Correspondence Analysis

Invented independently numerous times:

1. Correspondence Analysis: Weighted Principal Components with Chi-squared metric.
2. Optimal Scaling: Find site and species scores so that
   - all species occurring in one site are as similar as possible, but
   - species of different sites are as different as possible, and
   - sites are dispersed as widely as possible relative to species scores.
3. Reciprocal Averaging: Species scores are weighted averages of site scores, and simultaneously, site scores are weighted averages of species scores.

Chi-squared metric

- Metric is a 'yardstick' to measure dissimilarities among points.
- PCA uses constant ('Euclidean') metric, but CA uses expected abundances as a metric.
- Expected abundances from marginal totals: Exactly like in χ^2 analysis of contingency tables.
- *Species profile* is the average proportion of columns in the data, and *site profile* is the avarage proportion of rows.
- All sites should have all species in the same proportions.
- Chi-squared distance is the difference between expected profile and real abundance distributions – both for the species and the sites.

Necessary vocabulary from ecological analysis

- Species response curve – a graphical portrayal of the abundance of a species as a function of an environmental gradient.
- Abundance – a measure of the amount of a species in a sample (e.g. density, frequency, territorial area).
- Environmental gradient – a spatially varying aspect of the environment which is expected to be related to species composition.

CA vs PCA

- Practical application of CA is often driven by an identified problem of PCA – "horseshoe effect" that makes it unsuitable for use in many fields (e.g. ecology, text mining).
- The problem is caused by the fact that species often have unimodal species response curves along environmental gradients.
- In the case of species response curves, a unimodal distribution means the species has one optimal environmental condition.
- If any aspect of the environment is greater or lesser than this optimum, the species will perform more poorly (i.e. it will have a lesser abundance).
- Some techniques (such as CA) perform best when species have unimodal distributions, others (such as PCA) perform better when species have monotonic distributions along gradients (i.e. the species either increase or decrease, but not both, as a function of environmental factors).

CA vs PCA

- PCA assumes that species are linearly (or at least monotonically) related to each other and to gradients.
- PCA fails because it represents site occurances in species space (i.e. emphasis is on uncovering trends in species abundance).
- CA represents species and sites as occuring in a postulated enviroment space (often called ordination space).
- CA assumes that species have unimodal species reponse curves. A species is located in that location of space where it is most abundant.

Application 2: Correspondence Analysis

Measuring state preferences from speeches in the UN General Debates.

UN General Debates corpus

- We collected all speeches from 1970 (Session 25) to 2014 (Session 69).
- Documents available from the United Nations Dag Hammarskjold Library in six official languages (Arabic, Chinese, **English**, French, Russian, and Spanish).
- 7,310 statements.
- Varying number of countries participating in GD (70 in 1970 and 193 in 2014).
- Each statement on average contains 123 sentences and 945 *unique forms* and 3,248 *individual* words.

The meeting was suspended at 11.25 a.m. and resumed at 11.50 a.m.

ADDRESS BY MR. GEORGE BUSH, PRESIDENT OF THE UNITED STATES OF AMERICA

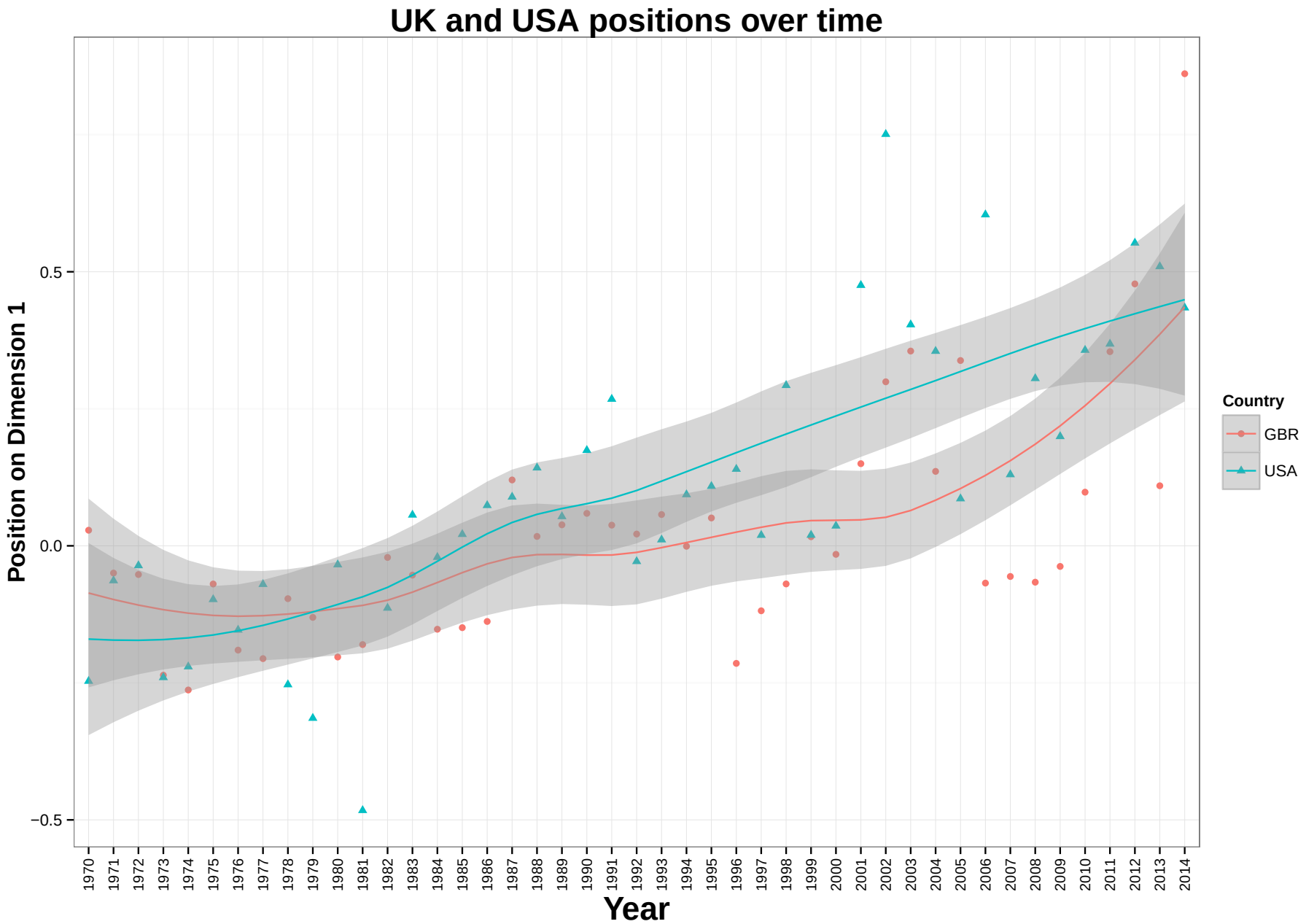
The PRESIDENT: The Assembly will now hear an address by the President of the United States of America.

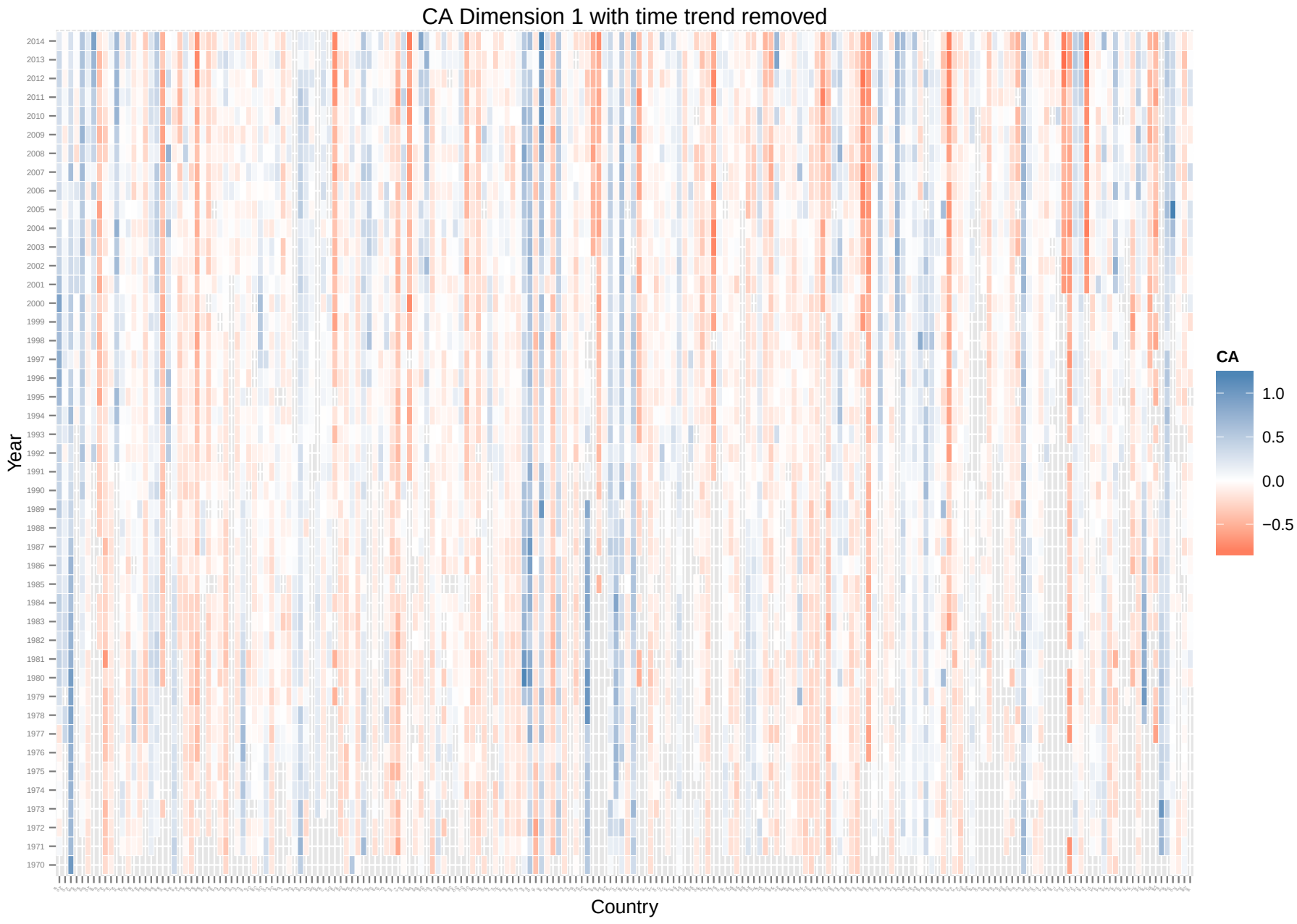
Mr. George Bush, President of the United States of America, was escorted into the General Assembly Hall.

The PRESIDENT: On behalf of the General Assembly I have the honour to welcome to the United Nations the President of the United States of America, His Excellency Mr. George Bush, and to invite him to address the General Assembly.

President BUSH: I am honoured to address the General Assembly today at the beginning of its forty-fourth session. I should like to congratulate Joseph Garba of Nigeria, a distinguished diplomat, on his election as President of this session of the General Assembly, and I wish him success in his presidency.

I feel a great personal pleasure on this occasion, for this is a homecoming


UK and USA positions over time
Position on Dimension 1
0.5
0.0
−0.5
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
Year
Country
GBR
USA


CA Dimension 1 with time trend removed
Year
2014
2013
2012
2011
2010
2009
2008
2007
2006
2005
2004
2003
2002
2001
2000
1999
1998
1997
1996
1995
1994
1993
1992
1991
1990
1989
1988
1987
1986
1985
1984
1983
1982
1981
1980
1979
1978
1977
1976
1975
1974
1973
1972
1971
1970
Country
CA
1.0
0.5
0.0
−0.5

Document-Term Matrix

Within the VSM approach computation in R (and most other software packages) requires creation of document-term matrix (DTM) also known as document-feature matrix:

dtm.2002

Filter

	far	stood	two	tower	symbol	freedom	prosper	progress	halfway	around	globe	magnific	buddha	repres	cultur	toler	nation	rich	histori
AFG_57_2002	1	2	3	1	2	2	5	1	1	1	1	1	1	5	3	4	17	2	3
AGO_57_2002	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	10	0	0
ALB_57_2002	0	0	1	0	0	1	0	0	0	0	0	0	0	1	2	1	19	0	0
AND_57_2002	0	0	2	6	1	0	0	0	0	0	0	0	0	0	2	6	9	0	5
ARE_57_2002	0	0	3	0	0	0	0	0	0	0	0	0	0	1	2	0	16	0	0
ARG_57_2002	0	0	0	0	0	0	1	1	0	0	0	0	0	2	0	0	13	0	0
ARM_57_2002	0	0	3	0	0	0	1	3	0	1	0	0	0	1	0	0	8	0	2
ATG_57_2002	0	0	0	0	0	0	0	2	0	0	2	0	0	2	0	0	11	0	0
AUS_57_2002	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	29	0	0
AUT_57_2002	0	0	3	0	0	2	0	1	0	0	0	0	0	0	3	1	15	0	0
AZE_57_2002	3	0	0	0	0	0	0	0	0	1	0	0	0	3	2	1	13	0	1
BDI_57_2002	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	7	1	0
BEL_57_2002	0	1	1	0	0	2	2	3	0	0	0	0	0	0	0	2	7	1	1
BEN_57_2002	0	0	0	0	0	0	0	2	0	0	0	0	0	1	0	0	19	0	0
BFA_57_2002	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
BGD_57_2002	0	0	3	0	0	1	0	1	0	0	0	0	0	1	2	1	5	1	1
BGR_57_2002	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	9	0	0
BHR_57_2002	0	1	0	0	0	0	3	1	0	0	0	0	0	0	4	1	20	0	0
BHS_57_2002	0	0	0	0	0	1	1	2	0	1	0	0	0	2	1	0	16	0	0
BIH_57_2002	1	0	1	0	0	0	1	3	0	0	0	0	0	4	1	3	7	0	0
BLR_57_2002	0	0	6	0	0	0	0	3	0	0	0	0	0	0	0	0	18	0	0
BLZ_57_2002	0	0	4	0	0	0	0	0	0	0	0	0	0	1	0	0	19	0	1
BOL_57_2002	0	0	1	0	0	0	1	1	0	0	0	0	0	1	0	1	7	2	2

Showing 1 to 23 of 188 entries

Other text classification models

Once you have a DTM you can apply any standard classification model. For example,

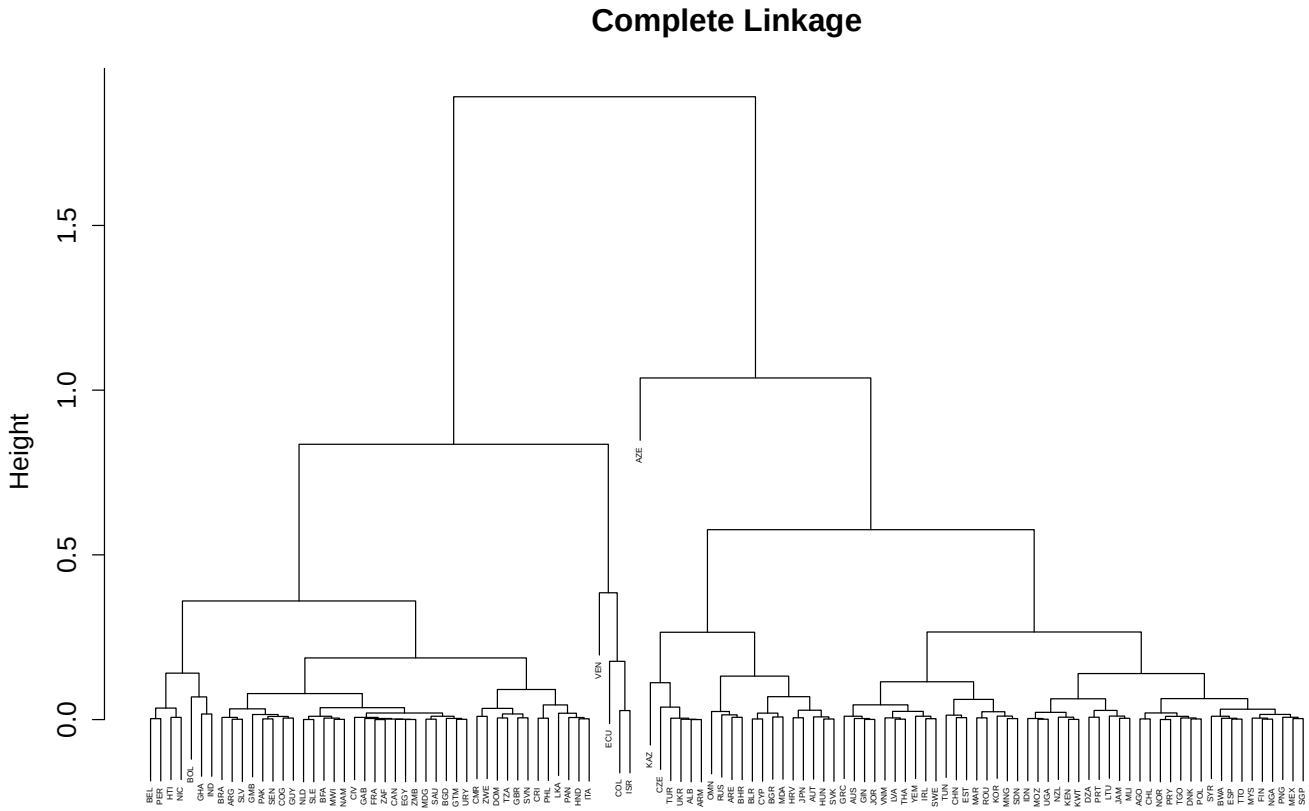
- kNN;
- Logistic regression;
- Support Vector Machines (SVMs);
- Decision trees.

More in *An Introduction to Information Retrieval*, Chapters 14-15.

Text clustering

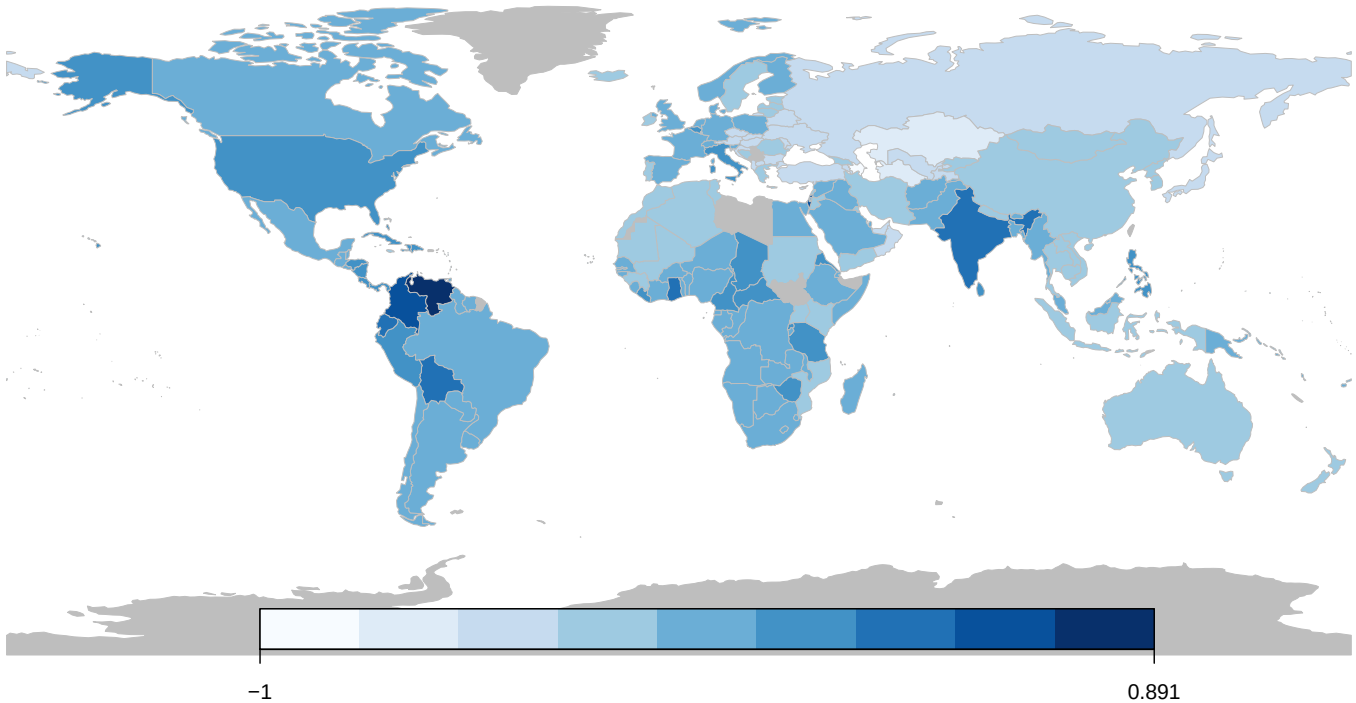
- In text clustering we are interested in identifying the clustering structure of a corpus of text documents and assigning documents to the identified cluster(s).
- We can apply to DTM standard clustering algorithms like k-means clustering (example of centroid-based clustering) and hierarchical clustering (also known as connectivity-based clustering).

Application 3: Hierarchical clustering



- Complete linkage hierarchical clustering.
- Scores on the second dimension of correspondence analysis of UN corpus for 2002.

Correspondence analysis Dimension 2: 2002



What you need to take out

- Vector space model allows us to treat text as data.
- With VSM we can apply any of the standard statistical tools we've learnt so far to discover knowledge and generate insights from text.