9. Text as Data

Irene Iodice

Bielefeld University

Over time it has became easier to store vast quantities of digital text

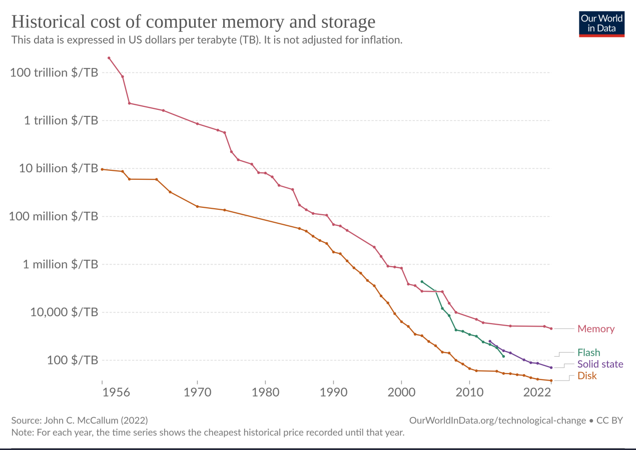


Fig: Historical cost of Computer Memory and Storage

Over time it has became easier to store vast quantities of digital text , explosion of empirical economics research using text as data

1. Finance

   predict asset price movements from news (Frank (2004) and Tetlock (2007))

Over time it has became easier to store vast quantities of digital text , explosion of empirical economics research using text as data

1. Finance

   predict asset price movements from news (Frank (2004) and Tetlock (2007))

2. Macroeconomics

   forecast variation in inflation and unemployment from google searches

Over time it has became easier to store vast quantities of digital text , explosion of empirical economics research using text as data

1. Finance
   predict asset price movements from news (Frank (2004) and Tetlock (2007))
2. Macroeconomics
   forecast variation in inflation and unemployment from google searches
3. Industrial Organization
   product reviews is used to study the drivers of consumer decision making

Text as Data

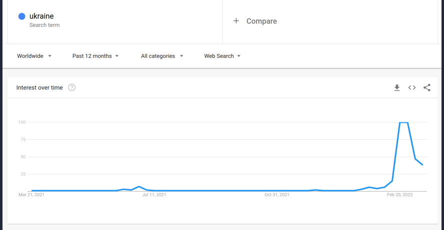
Strengths

Weakness

Text as Data

Strengths

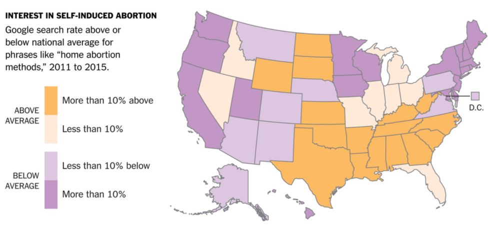
- "Always on"



Text as Data

Strengths

- ”Always on”
- ”Non-Reactive”



Text as Data

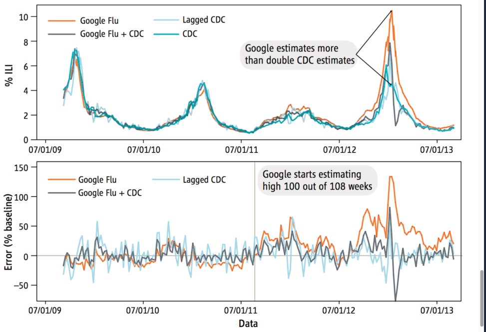
Weakness

- Incomplete
- Inaccessible or Sensitive
- Non-Representative

Text as Data

Weakness

- Confounding. Read about this here



GFT overestimation. GFT overestimated the prevalence of flu in the 2012–2013 season and overshot the actual level in 2011–2012 by more than 50%. From 21 August 2011 to 1 September 2013, GFT reported overly high flu prevalence 100 out of 108 weeks. **(Top)** Estimates of doctor visits for ILI. “Lagged CDC” incorporates 52-week seasonality variables with lagged CDC data. “Google Flu + CDC” combines GFT, lagged CDC estimates, lagged error of GFT estimates, and 52-week seasonality variables. **(Bottom)** Error [as a percentage {[Non-CDC estmate)−(CDC estimate)]/(CDC) estimate)}. Both alternative models have much less error than GFT alone. Mean absolute error (MAE) during the out-of-sample period is 0.486 for GFT, 0.311 for lagged CDC, and 0.232 for combined GFT and CDC. All of these differences are statistically significant at $P < 0.05$. See SM.

Which type of data?

Which type of data?

Imagine a document of w words where each word is drawn independently from a vocabulary of p possible words.

Which is the dimension of the unique representation?

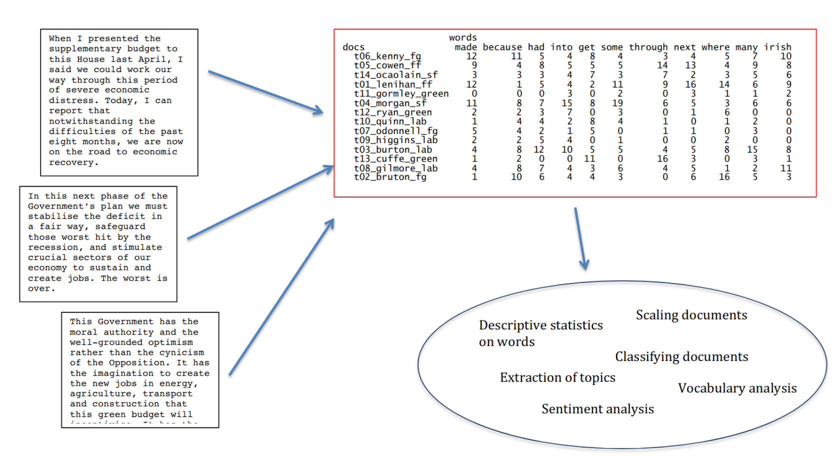
Which type of data?

Imagine a document of w words where each word is drawn independently from a vocabulary of p possible words.

Which is the dimension of the unique representation? p^w

High-dimensional data

Texts → Feature matrix → Analysis



docs \ words	made	because	had	into	get	some	through	next	where	many	irish
t06_kenny_fg	12	11	5	4	8	4	3	4	5	7	10
t05_cowen_ff	9	4	8	5	5	5	14	13	4	9	8
t14_ocaolain_sf	3	3	3	4	7	3	7	2	3	5	6
t01_lenihan_ff	12	1	5	4	2	11	9	16	14	6	9
t11_gormley_green	0	0	0	3	0	2	0	3	1	1	2
t04_morgan_sf	11	8	7	15	8	19	6	5	3	6	6
t12_ryan_green	2	2	3	7	0	3	0	1	6	0	0
t10_quinn_lab	1	4	4	2	8	4	1	0	1	2	0
t07_odonnell_fg	5	4	2	1	5	0	1	1	0	3	0
t09_higgins_lab	2	2	5	4	0	1	0	0	2	0	0
t03_burton_lab	4	8	12	10	5	5	4	5	8	15	8
t13_cuffe_green	1	2	0	0	11	0	16	3	0	3	1
t08_gilmore_lab	4	8	7	4	3	6	4	5	1	2	11
t02_bruton_fg	1	10	6	4	4	3	0	6	16	5	3

Source: Kenneth Benoit in his Course on Quantitative Text Analysis (TCD 2016)

Roadmap

1. How to represent Text as Data
   - Bag of words representation
   - Text Pre-processing
2. Statistical Methods to analyze data
   - Dictionary Based Methods
   - Generative Models
   - Text Regression Methods
   - Scaling
3. Applications

R packages to handle text data

Operation	R packages
Data preparation	
importing text	readtext jsonlite, XML, antiword, readxl, pdftools
string operations	stringi stringr
preprocessing	quanteda stringi, tokenizers, snowballC, tm, etc.
document-term matrix (DTM)	quanteda tm, tidytext, Matrix
filtering and weighting	quanteda tm, tidytext, Matrix
Analysis	
dictionary	quanteda tm, tidytext, koRpus, corpustools
supervised machine learning	quanteda RTextTools, kerasR, austin
unsupervised machine learning	topicmodels quanteda, stm, austin, text2vec
text statistics	quanteda koRpus, corpustools, textreuse
Advanced topics	
advanced NLP	spacyr coreNLP, cleanNLP, koRpus
word positions	and syntax corpustools quanteda, tidytext, koRpu

Section 1

Representing Text as Data

How to represent news?

FINANCIAL TIMES

NEWS PROVIDER OF THE YEAR

Venezuela dilemma
How Maduro has strengthened his grip on power

Deep impact
Business world sizes up key figures, risks and trends for 2021

Reformation period
Britain needs to set bold long-term goals

High flyer
Bitcoin value tops $30,000

$34,822

$7,258

Bitcoin price

Briefing

Johnson tells England to prepare for tighter Covid-19 restrictions

Hospitalisations up 33% • Disarray over schools • Labour seeks nationwide lockdown

Datawatch

Push by Trump's Republican rebels to overturn US vote faces fierce backlash

World Markets

A Nikkei Company

THE ULTIMATE GUIDE TO 2021

FTWeekend

FTWEEKEND
DIGITAL FESTIVAL
SPRING EDITION
18-20 March 2021

How to represent news?

```r
library(wordcloud)
```

Source: Financial Time Blog on March 24th 2020

How to represent news?

```r
library(wordcloud)
```

Source: Financial Time Blog on March 24th 2020

What would you have done differently?

Processing Text Data

1. Divide text into documents
2. Split documents into features
3. Reduce the number of language elements

Processing Text Data

1. Divide text into documents

   e.g. newspaper by day, topics

   level of aggregation not always obvious

2. Split documents into features
3. Reduce the number of language elements

Processing Text Data

1. Divide text into documents
2. Split documents into features
   ”tokenize” documents limiting dependencies
3. Reduce the number of language elements

Processing Text Data

1. Divide text into documents
2. Split documents into features
3. Reduce the number of language elements
    3.1 Remove Stop words
    3.2 Stemming and lemmatization

Tokenization

It involves breaking down text into smaller units or tokens (words, characters, n-grams)

Tokenization

It involves breaking down text into smaller units or tokens (words, characters, n-grams)

```
> gutenbergr::gutenberg_download(1184)[1,]
# A tibble: 1 × 2
  gutenberg_id text
         <int> <chr>
1         1184 THE COUNT OF MONTE CRISTO
> library(tokenizers)
> gutenberg_download(1184)[1,]  %>%
+   unnest_tokens(input=text, output=word, token = "words")
# A tibble: 5 × 2
  gutenberg_id word
         <int> <chr>
1         1184 the
2         1184 count
3         1184 of
4         1184 monte
5         1184 cristo
```

❦ ”capital gains tax” is a trigram, to detect diagram/triagram use collocation methods which involves statistical tests of independence

Bag of words representation

When text (sentence or a document) is represented as the bag (multiset) of its words

- disregard grammar and word order
- keep multiplicity (multiset)

Bag of words representation

Example of 2 movie reviews

1. "This movie is spooky and is original"
   - BoW_{R1} = {"This":1,"movie":1,"is":2,"spooky":1,"and":1,"original":1}
2. "This movie is original but long"
   - BoW_{R2} = {"This":1,"movie":1,"is":1,"original":1,"but":1,"long":1}

	This	movie	is	spooky	and	original	but	long
BoW_{R1}	1	1	2	1	1	1	0	0
BoW_{R2}	1	1	1	0	0	1	1	1

Bag of words representation

Example of 2 movie reviews

1. "This movie is spooky and is original"
   - BoW_{R1} = {"This":1,"movie":1,"is":2,"spooky":1,"and":1,"original":1}
2. "This movie is original but long"
   - BoW_{R2} = {"This":1,"movie":1,"is":1,"original":1,"but":1,"long":1}

	This	movie	is	spooky	and	original	but	long
BoW_{R1}	1	1	2	1	1	1	0	0
BoW_{R2}	1	1	1	0	0	1	1	1

new words $\implies \uparrow$ vocabulary size $\implies \uparrow$ dimension of the problem: pre-processing

Feature Selection

It involves stripping out elements that are not signal

1. apply lowercase, remove punctuation and ”stop words” using pre-build dictionaries

```
library(hcandersenr)
library(tidytext)

tidy_fir_tree <- hca_fairytales() %>%
  filter(book == "The fir tree") %>%
  unnest_tokens(word, text) %>%
  filter(!(word %in% stopwords(source = "snowball")))

setdiff(stopwords(source = "snowball"),
        stopwords(source = "smart"))
>  [1] "she's"   "he'd"    "she'd"   "he'll"   "she'll"   "shan't"   "mustn't"
>  [8] "when's"  "why's"   "how's"
```

Feature Selection

It involves stripping out elements that are not signal

1. apply lowercase, remove punctuation and ”stop words” using pre-build dictionaries
2. build your own dictionaries, e.g. via “term frequency–inverse document frequency” (tf-idf)
   - word j in document i has $tf_{ij} \times idf_j$
   - tf_{ij} is the count of occurrences of a word/feature j in document i
   - idf_j is the log of one over the share of docs containing j, i.e. $log\left(\frac{1}{s_j}\right)$ with $s_j = \frac{\sum_i^n 1[tf_{ij} > 0]}{n}$

Example of tf-idf

We have 100 political party manifestos, each with 1000 words. The first document contains 16 times the word "inequality"; 40 of the manifestos contain the word "inequality"

- tf_{ij} = 16/1000 = 0.016
- idf_j = 100/40 = 2.5, and ln(2.5) = 0.916
- tf-idf = 0.016 $\times$ 0.916 = 0.0147

$\uparrow tf_{ij} \times idf_j$ is reached by a high term frequency (in the given document) and a low document frequency of the term in the whole collection of documents $\rightarrow$ filter out common terms

Stemming and Lemmatization

They refer to the process of reducing words to their base or root form

- am, are, $\Rightarrow$ be
- car, cars, car's, cars' $\Rightarrow$ car

Stemming and Lemmatization

Stemming usually refers to a crude heuristic process that chops off the ends of words

```
library(textstem)
x <- c('doggies', ',', 'they', "aren\'t", 'Joyfully', 'running', '.')
stem_words(x)
[1] "the" "doggi" "," "well" "thei" "aren't" "Joyfulli" "run"  "."
```

Most famous algorithm is by Porter in 1980

Stemming and Lemmatization

Lemmatization is more structured, uses vocabulary and morphological analysis of words

```
library(textstem)
x <- c("the", 'doggies', 'well', "aren\'t", 'Joyfully', 'running', '.')
stem_words(x)
[1] "the"      "doggi"    "well"     "aren't"   "Joyfulli" "run"      "."
lemmatize_words(x)
[1] "the"      "doggy"    "good"     "aren't"   "Joyfully" "run"      "."
```

Similarity across texts

	This	movie	is	spooky	and	original	but	long
BoW_{R1}	1	1	2	1	1	1	0	0
BoW_{R2}	1	1	1	0	0	1	1	1

Define $a = |BoW_{R1} \cap BoW_{R2}|$, b =$|BoW_{R1}| - a$ and c =$|BoW_{R2}| - a$

1. Cosine Similarity:

$$s_{cosine} = \frac{a}{\sqrt{(a+b)(b+c)}} \tag{1}$$

2. Jaccard Similarity

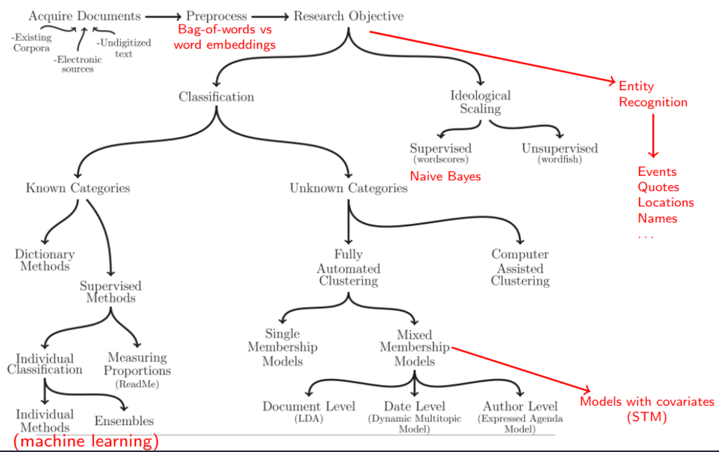
$$s_{jacc} = \frac{a}{\sqrt{(a+b+c)}} \tag{2}$$

How much in our example? What after stemming ?

Section 2

Statistical Methods

Overview of Methods



Grimmer and STewart (2013), expanded by Kennet Benoit

Classifying and Scaling Documents

Two types of measurement schemes:

1. Classification of documents: involves categorical (often binary) measures
2. Scaling of documents: involves continuous measure

Common goal: Assign a text to a particular category, or a particular position on a scale

From text tokens to attributes

Let $\mathbf{C}$ be the document-token matrix and $\mathbf{V}$ the matrix of attributes

- $\mathbf{C}^{\text{train}}$ include those for which we have obs. $\mathbf{V}^{\text{train}}$ of $\mathbf{V}$
- $\mathbf{C}^{\text{test}}$ those for which $\mathbf{V}$ is unobserved
- $\mathbf{C}^{\text{train}}$ is $n^{train} \times p$
- $\mathbf{V}^{\text{train}}$ is $n^{train} \times k$

From text tokens to attributes

Let $\mathbf{C}$ be the document-token matrix and $\mathbf{V}$ the matrix of attributes

- $\mathbf{C}^{\text{train}}$ include those for which we have obs. $\mathbf{V}^{\text{train}}$ of $\mathbf{V}$
- $\mathbf{C}^{\text{test}}$ those for which $\mathbf{V}$ is unobserved
- $\mathbf{C}^{\text{train}}$ is $n^{train} \times p$
- $\mathbf{V}^{\text{train}}$ is $n^{train} \times k$

How to map $\mathbf{C}$ to predictions $\widehat{\mathbf{V}}$?

From text tokens to attributes

Let $\mathbf{C}$ be the document-token matrix and $\mathbf{V}$ the matrix of attributes

- $\mathbf{C}^{\text{train}}$ include those for which we have obs. $\mathbf{V}^{\text{train}}$ of $\mathbf{V}$
- $\mathbf{C}^{\text{test}}$ those for which $\mathbf{V}$ is unobserved
- $\mathbf{C}^{\text{train}}$ is $n^{train} \times p$
- $\mathbf{V}^{\text{train}}$ is $n^{train} \times k$

How to map $\mathbf{C}$ to predictions $\widehat{\mathbf{V}}$?

Main methods in the eco literature

From ”Text as Data” by Gentzkow et al. here:

1. Dictionary-Based Methods

   a prespecified dictionary characterizes $f(.)$, s.t. $\hat{v}_i = f(c_i)$

2. Text Regression methods

   model $p(v_i|c_i)$, use $\mathbf{C}^{\text{train}}$ $\mathbf{V}^{\text{train}}$ to estimate $E(v_i|c_i)$

3. Generative model

   model $p(c_i|v_i)$, e.g. fit $f_\theta(c_i, v_i)$ and then invert to predict v_i

4. Word Embeddings

   representation of words in vector space , e.g. Word2Vec

Subsection 1

Dictionary Based Methods

Dictionary Based Methods

It consists in classifying documents when categories are known

Dictionary Based Methods

It consists in classifying documents when categories are known

1. identify a set of words that correspond to each category
    - thesaurus: vote = $\{$poll, suffrage, franchis*, ballot*, $vot\}$
    - sentiment: positive or negative
    - emotions: sad, happy, angry, anxious
    - topics: economics, culture, etc.

Dictionary Based Methods

It consists in classifying documents when categories are known

1. identify a set of words that correspond to each category
2. count number of times these words appear in each document
3. Normalize by document length
4. Validate

Dictionary Based Methods

It consists in classifying documents when categories are known

1. identify a set of words that correspond to each category
2. count number of times these words appear in each document
3. Normalize by document length
4. Validate
    - Code a few documents manually and see if dictionary prediction aligns

Dictionary Based Methods

It consists in classifying documents when categories are known

1. identify a set of words that correspond to each category
   A few? Decide a sample size based on the the power of your test

Existing Dictionaries

Existing lists of words associated with sentiment, emotions, topics ...

1. General Inquirer (Stone et al 1966): propietary :-(but.. a sample accessible via:

```
library("qdapDictionaries")
data(power.words)
force(power.words)[1:8]
[1] "abolish"          "accomplish"          "accomplishment"    "accord"
[8] "achievement"     "adjudication"    "administer"      "administration"
```

Open source alternatives:

- Valence Aware Dictionary and sEntiment Reasoner on github here
- LexiCoder (media text), SentiStrength (social media text)

Existing Dictionaries

❦ Highly specific to context

Ex: Loughran and McDonald (2010): use Harvard-IV-4 TagNeg (H4N) to classify sentiment for firms 10-K filings: 3/4 of the "negative" words of H4N were typically not negative in a financial context e.g. cancer, or tax, cost, capital, board, liability and foreign

=> polysemes – words that have multiple meanings

=> lacks of negative financial words, e.g. felony, litigation, restated, misstatement, and unanticipated

Build your own dictionary

1. Identify “extreme texts” with “known” positions
2. Search for deferentially occurring words using word frequencies
3. Use these words (or their lemmas) for categories

Build your own dictionary

Contingency tables on the use of the keywords in Parliament Meetings

	Government	Opponents
labor flexibility	100	20
environment	115	25

Expected frequency if keywords are independent of the group

	Government	Opponents
labor flexibility	$(120 \times 215)/260$	$(120 \times 45)/260$
environment	$(140 \times 215)/260$	$(140 \times 45)/260$

Test independence, $\chi^2 = \sum \frac{(O_{ij}-E_{ij})^2}{O_{ij}}$, How much?

Practical Corner

Regular Expressions: algebraic notations for characterizing a set of strings, useful to search patterns of text

```
# select strings that contain any digit
> grep("[0-9]", "Chapter 2", value=TRUE)
[1] "Chapter 2"
# select strings that starts with either l or L + "ov"
> grep("^[lL]ov", c("love", "Lovely", "very lovely"), value = TRUE)
[1] "love"   "Lovely"
# select strings that starts with beg and ends with n
> grep("beg.n", c("begun", "beg3n", "begin"), value = TRUE)
[1] "begun" "beg3n" "begin"
```

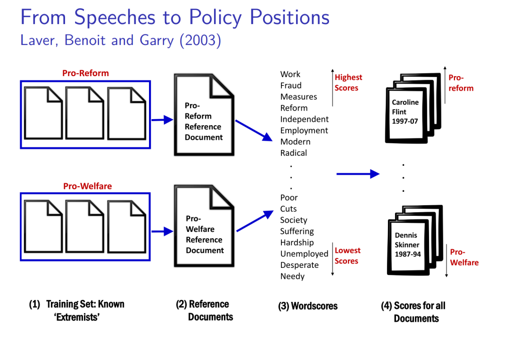
Useful cheatsheet can be found here

Subsection 2

Generative Models

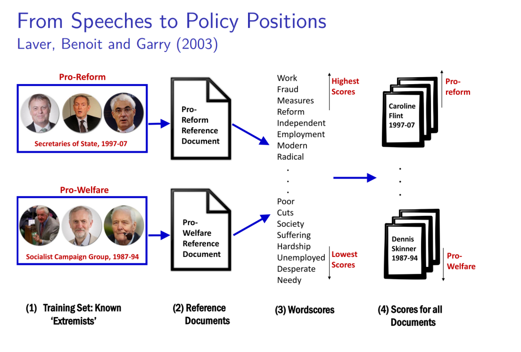
Scaling Using Wordscores

Wordscores is a type of supervised scaling, meaning that we have some documents for which we already know the outcome variables which we then use to build our model



Scaling Using Wordscores

Wordscores is a type of supervised scaling, meaning that we have some documents for which we already know the outcome variables which we then use to build our model



Scaling Using Wordscores

1. Pre-assign a score, Ar to each reference document
2. Calculate relative frequency of every word w in each reference document F_{wr}
3. Calculate probability that we are reading r , given that we are seeing

$$P_{wr} = \frac{F_{wr}}{\sum_r F_{wr}} \tag{3}$$

4. Produce a score for each word

$$S_w = \sum_w P_{wr} \times A_r \tag{4}$$

5. Use the wordscores to score each unlabelled document v

Subsection 3

Text Regression Methods

Which type of data? High-dimensional data, i.e. $p > n$

Which type of data? High-dimensional data, i.e. $p > n$

High-dimensional regression methods

1. Subset selection Identifying a relevant subset of the p < n predictors, and fitting an OLS model on the reduced set of variables
2. Shrinkage Fitting a model involving all predictors, but penalizing (regularizing) the coefficients in such as way that they are shrunken towards zero relative to the least squares estimates
3. Dimension Reduction Replacing the p predictors with projections (linear combinations) of the predictors onto M-dimensional subspace, where M < p, and then fitting an OLS model on the reduced set of (combination) variables

Penalized linear models

$$\min_{\beta \in \mathbb{R}^p} \left\{ \underbrace{l(\alpha, \beta)}_{\text{loss function}} + n\lambda \sum_{j=1}^{p} \underbrace{k_j(|\beta_j|)}_{\text{penality shrinkage}} \right\}$$

where:

- $l(\alpha, \beta) = \frac{1}{n} \sum_{i=1}^{n} \left(v_i - (\alpha + \mathbf{x}_i' \beta) \right)^2$ in Gaussian linear reg. (RSS)
- $k_j(.)$ increasing cost function that penalizes dev of β_j from zero
- $\lambda \geq 0$ adjusts the margin (or 'complexity') of the solution (typically chosen using a held-out sample or K-fold Cross Validation)
- The sample size n term scales down the penalty term to compensate for the increased amount of information present in larger dataset.

Common functions for $k_j(.)$

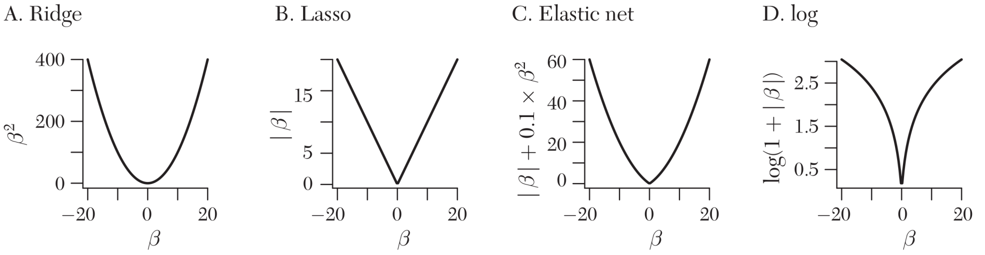


Figure 1

Note: From left to right, L_2 costs (ridge, Hoerl and Kennard 1970), L_1 (lasso, Tibshirani 1996), the "elastic net" mixture of L_1 and L_2 (Zou and Hastie 2005), and the log penalty (Candès, Wakin, and Boyd 2008).

L_1 Regularization

$$\min_{\beta \in \mathbb{R}^p} \left\{ l(\alpha, \beta) + n\lambda \sum_{j=1}^{p} \omega_j |\beta| \right\}$$

- ω_j is usually the covariates scaled by the SD
- in text analysis ω_j are usually weights of text tokens such as ”rare feature up-weighting” - similar to tf-idf!

Classifications problems

SVM classifier

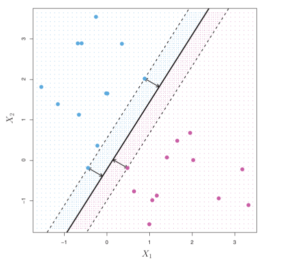
$$\min_{\mathbf{w}\in\mathbb{R}^p} \left\{ \frac{1}{n}\sum_{i=1}^{n} \underbrace{\max\{0, 1 - v_i \mathbf{w}^\top \mathbf{x}_i\}}_{\text{'Hinge' loss function on } (\mathbf{x}_i, v_i)} + \underbrace{\lambda\|\mathbf{w}\|_2^2}_{\text{k() is usually L2}} \right\}$$

- v_i represents the true label of the example, which can take the values of either -1 or +1 for binary classification.
- The hinge loss is zero when the predicted score multiplied by the true label y is greater than or equal to 1, indicating that the example is correctly classified
- f(x) by y is less than 1 is a case of misclassification or insufficient margin, the hinge loss becomes positive and increases linearly with the magnitude of the margin

Non-linear text regression

SVM classifier

$$\min_{\mathbf{w}\in\mathbb{R}^p} \left\{ \frac{1}{n}\sum_{i=1}^{n} \underbrace{\max\{0, 1 - v_i \mathbf{w}^\top \mathbf{x}_i\}}_{\text{'Hinge' loss function on } (\mathbf{x}_i, v_i)} + \lambda \|\mathbf{w}\|_2^2 \right\}$$



Other methods: Regression Trees, Deep Learning

Section 3

Applications

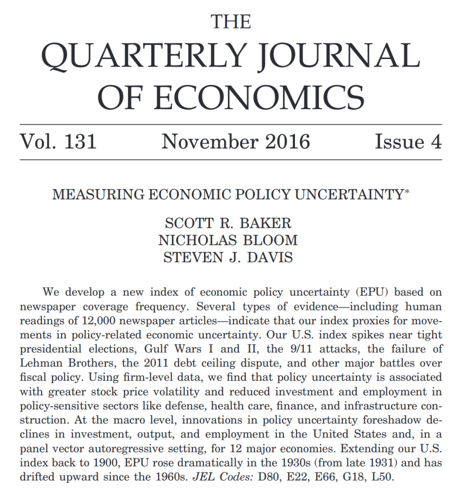
Frame Title

2 examples of classification:

1. Dictionary based methods
2. Clustering by similarity of text

Application 1

Can we measure policy uncertainty in the US, how does this look like and does it matter?



THE QUARTERLY JOURNAL OF ECONOMICS

Vol. 131 November 2016 Issue 4

MEASURING ECONOMIC POLICY UNCERTAINTY*

SCOTT R. BAKER
NICHOLAS BLOOM
STEVEN J. DAVIS

We develop a new index of economic policy uncertainty (EPU) based on newspaper coverage frequency. Several types of evidence—including human readings of 12,000 newspaper articles—indicate that our index proxies for movements in policy-related economic uncertainty. Our U.S. index spikes near tight presidential elections, Gulf Wars I and II, the 9/11 attacks, the failure of Lehman Brothers, the 2011 debt ceiling dispute, and other major battles over fiscal policy. Using firm-level data, we find that policy uncertainty is associated with greater stock price volatility and reduced investment and employment in policy-sensitive sectors like defense, health care, finance, and infrastructure construction. At the macro level, innovations in policy uncertainty foreshadow declines in investment, output, and employment in the United States and, in a panel vector autoregressive setting, for 12 major economies. Extending our U.S. index back to 1900, EPU rose dramatically in the 1930s (from late 1931) and has drifted upward since the 1960s. *JEL Codes:* D80, E22, E66, G18, L50.

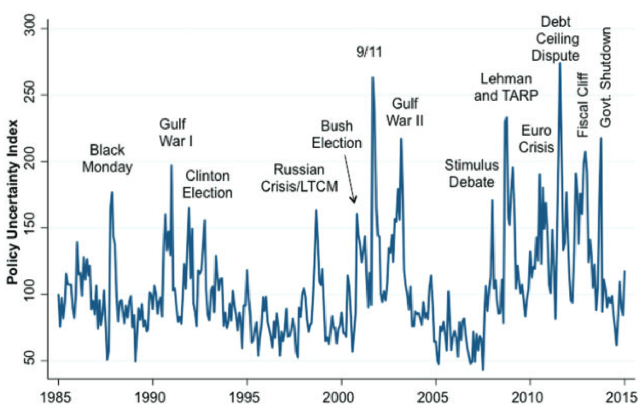
Methodology

Let i be a country-month pair, j be a newspaper and a_j an article, with $j = 1, .., n$ and $a = 1, .., m_j$

- $c_{ij} = \frac{1}{m_j} \sum_a 1 \left[\sum_{t=\{E,P,U\}} 1 \left[BoW_{ijat} \cap K_t \neq \emptyset \right] = 3 \right]$ is the share of articles that contain at lease one keyword in each of the following sets:
  - $K_E = \{$"economy", "economics"$\}$
  - $K_U = \{$"uncertain", "uncertain"$\}$
  - $K_P = \{$"regulation", "deficit", "federal reserve", "legislation", "white house"$\}$

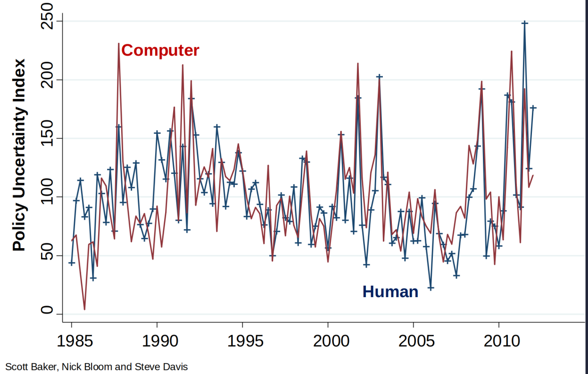
- $c_i = \frac{1}{n} \sum_j c_{ij}$ is the avg. across newspapers
- $\hat{v}_i = c_i$, where $\hat{v}_i$ called Economic Policy Uncertainty (EPU) index

Economic Policy Uncertainty Index



Index reflects scaled monthly counts of articles containing 'uncertain' or 'uncertainty', 'economic' or 'economy', and one or more policy relevant terms: 'regulation', 'federal reserve', 'deficit', 'congress', 'legislation', or 'white house'. The series is normalized to mean 100 from 1985-2009 and based on queries run on 2 February, 2015 for the USA Today, Miami Herald, Chicago Tribune, Washington Post, LA Times, Boston Globe, SF Chronicle, Dallas Morning News, NY Times, and the Wall Street Journal.

Validation of the index



Testing economic hypothesis

Negative Uncertainty Effects

- Utility functions (risk-aversion, e.g. Tobin (1958))
- Adjustment costs (real options Bernanke (1983), Dixit Pindyck(1994))
- Financial frictions (e.g. Gilchrist et al. (2010))
- Ambiguity (robustness, e.g. Hansen Sargent, Ilut Schneider)

Testing economic hypothesis

Negative Uncertainty Effects

- Utility functions (risk-aversion, e.g. Tobin (1958))
- Adjustment costs (real options Bernanke (1983), Dixit Pindyck(1994))
- Financial frictions (e.g. Gilchrist et al. (2010))
- Ambiguity (robustness, e.g. Hansen Sargent, Ilut Schneider)

$\uparrow$ Uncertainty $\rightarrow$ $\uparrow$ real option to wait $\rightarrow$ $\downarrow$ investment

Regression analysis

Microdata: Firm-level estimates exploit differences in industry exposure to government

Match Federal Registry of Contracts data to Compustat (via D&B)

Table 4: Highest Contract Intensities by SIC Code

SIC Code	SIC Description	Total Contracts (B$)	Total Revenue (B$)	Contract Intensity
3760	Guided Missiles And Space Vehicles And Parts	392.63	511.65	0.767
3790	Miscellaneous Transportation Equipment	184.72	388.13	0.476
3812	Search, Detection, Navigation, Guidance, Aeronautical, and Nautical Systems	315.28	694.00	0.454
3480	Ordnance And Accessories, Except Vehicles And Guided Missiles	22.15	54.64	0.405
2780	Blankbooks, Looseleaf Binders, And Bookbinding	18.19	46.91	0.388
8711	Engineering Services	86.76	369.00	0.235
1623	Water, Sewer, Pipeline, and Communications and Power Line Construction	26.64	135.44	0.197
1600	Heavy Construction Other Than Building Construction Contractors	87.71	543.66	0.161
3720	Aircraft And Parts	83.49	584.59	0.143
8050	Nursing And Personal Care Facilities	1.44	15.32	0.094
7373	Computer Integrated Systems Design	162.05	1819.42	0.089
3714	Motor Vehicle Parts and Accessories	161.90	2134.25	0.076
3844	X-Ray Apparatus and Tubes and Related Irradiation Apparatus	1.77	24.22	0.073

Generate average industry contracts/revenue (1999 to 2012)

Regression analysis

Microdata: Firm-level estimates exploit differences in industry exposure to government

$$Y_{it} = FE_i + FE_t + \beta \underbrace{INT_i \times \hat{v}_{it}}_{\text{Firm gov. exposure} \times \text{EPU}} + \alpha \underbrace{INT_i \times GS_t}_{\ldots \times \text{gov. expenditure}} + \epsilon_{it} \tag{5}$$

where

- i=firm, j=industry, t=quarter
- $INT_i = \sum_j w_{ij} INT_j$ where w_{ij} is the relvance of business in j for firm i
- Y_{it} represents investment or hiring
- Estimated firm by quarter 1996-2012, standard-errors clustered by j

Results

TABLE IV

POLICY UNCERTAINTY AND FIRM-LEVEL INVESTMENT, EMPLOYMENT, AND SALES

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
Dependent variable	I/K	I/K	I/K	I/K	ΔEmp	ΔEmp	ΔEmp	ΔEmp	ΔRev
Δ Log(EPU) $\times$ intensity	−0.032***	−0.032***	−0.024**	−0.029***	−0.213**	−0.227**	−0.220**	−0.220**	−0.128
	(0.010)	(0.010)	(0.011)	(0.010)	(0.084)	(0.089)	(0.118)	(0.094)	(0.096)
$\Delta\frac{\text{Federal purchases}}{\text{GDP}} \times$ intensity	8.20***	8.04***	12.12***	8.85***	10.79	15.60***	3.19	10.99	20.39**
	(2.86)	(2.86)	(3.18)	(2.87)	(7.41)	(8.04)	(12.56)	(7.88)	(9.43)
$\Delta\frac{\text{Forecasted Federal purchases}}{\text{GDP}} \times$ intensity		1.01				−4.65***			
		(0.828)				(2.89)			
Δ Log(defense EPU) $\times$ defense firm				0.002				0.018	
				(0.004)				(0.017)	
Δ Log(health care EPU) $\times$ health firm				−0.012***				−0.005	
				(0.002)				(0.025)	
Δ Log(fin. reg. EPU) $\times$ finance firm				−0.002***				0.003	
				(0.001)				(0.005)	
Periodicity	Quarterly	Quarterly	Quarterly	Quarterly	Yearly	Yearly	Yearly	Yearly	Yearly
3 yrs Fed purchase leads	No	No	Yes	No	No	No	Yes	No	No
Observations	708,398	708,398	411,205	708,398	162,006	162,006	107,205	162,006	151,473
Number of firms	21,636	21,636	13,563	21,636	17,151	17,151	11,505	17,151	15,749

Notes. The sample period runs from 1985 to 2012. All columns include a full set of firm and time effects. I/K is the investment rate defined as $\frac{\text{CapEx}_t}{(\text{NetPlant, Property and Equipment})_{t-1}}$. ΔEmp is the employment growth rate measured as $\frac{\text{emp}_t - \text{emp}_{t-1}}{0.5\times\text{emp}_t + 0.5\times\text{emp}_{t-1}}$, and ΔRev is the corresponding revenue growth rate. $\Delta\frac{\text{Federal purchases}}{\text{GDP}} \times$ intensity is the change in $\frac{\text{federal purchases}}{\text{GDP}}$ from NIPA tables in the next quarter in quarterly specifications and in the next year in annual specifications, multiplied by the firm-level policy exposure intensity variable. $\Delta\frac{\text{Forecasted federal purchases}}{\text{GDP}} \times$ intensity instead uses the mean forecasted change in $(\frac{\text{federal purchases}}{\text{GDP}})$ from the Federal Reserve Bank of Philadelphia's Survey of Professional Forecasters, drawing on NIPA data for the current values and forecast data for the future values. See the notes to Table II for additional variable definitions. Standard errors based on clustering at the firm level. $***p < 0.01$, $**p < 0.05$, $*p < 0.1$

Application 2

How to define a product market which is endogeneous to firms' choices?

Text-Based Network Industries and Endogenous Product Differentiation

Gerard Hoberg

University of Southern California

Gordon Phillips

Dartmouth College and National Bureau of Economic Research

We study how firms differ from their competitors using new time-varying measures of product similarity based on text-based analysis of firm 10-K product descriptions. This year-by-year set of product similarity measures allows us to generate a new set of industries in which firms can have their own distinct set of competitors. Our new sets of competitors explain specific discussion of high competition, rivals identified by managers as peer firms, and changes to industry competitors following exogenous industry shocks. We also find evidence that firm R&D and advertising are associated with subsequent differentiation from competitors, consistent with theories of endogenous product differentiation.

Methodology

- web scrawl 50,673 firm annual 10-Ks filed
- use the product description
- text pre-processing steps
  - only focus on nouns as defined by Webster.com
  - $\frac{1}{idf} < 25\%$
  - tokenize text and generate BoW

Methodology

Let $p^i \in \{0,1\}^K$ be a vector representation of product description for firm i, where K is $K = |BoW_1 \cup ... \cup BoW_{50,673}|$ (full dictionary dimension)

Pair-wise cosine similarity between firm P^i and P^j

$$S_C(P^i, P^j) := \cos(\theta) = \frac{\mathbf{P^i} \cdot \mathbf{P^j}}{\|\mathbf{P^i}\| \|\mathbf{P^j}\|} = \underbrace{\frac{\sum_{k=1}^{K} P_k^i P_k^j}{\sqrt{\sum_{k=1}^{K} (P_k^i)^2} \sqrt{\sum_{k=1}^{K} (P_k^j)^2}}}_{\text{\# of words in common normalised by length}}$$

Alternative, define $p^i \in \mathbb{R}^k$, using TF-IDF

Validity of the new industry classification

TABLE 3
FIRM CHARACTERISTICS AND INDUSTRY CLASSIFICATIONS

Industry Controls	OI/Sales	OI/ Assets	Sales Growth	Market Beta	Asset Beta
	A. Across-Industry Standard Deviations: Firm-Weighted Results; All Industry Classifications				
1. SIC-3 fixed effects	.204	.111	.126	.283	.271
2. NAICS-4 fixed effects	.205	.112	.136	.289	.276
3. 10-K-based 300 fixed effects	.231	.128	.157	.298	.285
4. TNIC equal-weighted average	.248	.142	.163	.332	.324
5. TNIC similarity-weighted average (excluding the focal firm)	.267	.153	.199	.384	.369
	B. Across-Industry Standard Deviations: Industry-Weighted Results; Transitive Industry Classifications Only				
1. SIC-3 fixed effects	.156	.111	.179	.347	.308
2. NAICS-4 fixed effects	.169	.126	.210	.414	.362
3. 10-K-based 300 fixed effects	.202	.139	.224	.469	.432

NOTE.—For a given variable indicated in the left-hand column, across-industry standard deviations are computed as the standard deviation of the industry average of the given variable across all firms in our sample (panel A) and across all industries (panel B). TNIC refers to text-based network industries.

Results

TABLE 6
EX ANTE ADVERTISING AND R&D VERSUS FUTURE SIMILARITY

Dependent Variable	Positive Advertising Dummy	Positive R&D Dummy	Log Industry Adver./Sales	Log Industry R&D/Sales	Industry Past Stock Return	Log Assets	Industry Log B/M Ratio	Adjusted R^2
	A. Text-Based Network Industry Regressions							
1. Δ total similarity	−.414	−.152	−.034	−.005	.055	.018	−.004	.127
	(−13.72)	(−5.80)	(−8.55)	(−1.37)	(1.63)	(2.71)	(−.25)	
2. Δ number of rivals	−12.301	−1.997	−1.195	.156	2.184	1.636	−1.616	.102
	(−6.94)	(−1.70)	(−4.83)	(.87)	(1.41)	(4.38)	(−1.23)	
3. Δ profitability	.038	.039	.004	.005	−.022	−.010	.014	.078
	(4.61)	(6.61)	(5.15)	(7.38)	(−4.92)	(−8.43)	(3.74)	
	B. Industry-Adjusted Firm-Level Regressions							
4. Δ total similarity	−.037	−.116	−.005	−.020	.059	.016	.020	.121
	(−1.33)	(−5.15)	(−.83)	(−4.25)	(1.76)	(2.40)	(1.29)	
5. Δ number of rivals	−.775	−3.768	−.103	−.738	2.208	1.464	−1.291	.100
	(−.62)	(−4.03)	(−.37)	(−3.71)	(1.42)	(3.99)	(−.96)	
6. Δ profitability	.031	.053	.007	.013	.009	.015	.001	.015
	(4.83)	(9.22)	(5.21)	(10.87)	(2.97)	(11.15)	(.27)	

NOTE.—Ordinary least squares regressions with ex post product changes in total similarity and the number of rivals and profitability as the dependent variables. Panel A is based on advertising and R&D computed at the text-based network industry level. Panel B is based on firm-level network-industry-adjusted advertising and R&D. All specifications include 10-K-based fixed industry classification and yearly fixed effects; *t*-statistics in parentheses are based on standard errors adjusted for clustering by year and industry. The sample has 49,246 observations.

Frame Title

https://web.stanford.edu/ jurafsky/slp3/