Introduction to Machine Learning for Social Scientists

Class 9: Regularization 1

Edgar Franco Vivanco

Stanford University
Department of Political Science

edgarf1@stanford.edu

Summer 2018

Logistics

Today:

- Instructions for Group project and Lottery
- Text Analysis
- Intro to regularization

Reading between the lines

American Political Science Review (2018) 112, 2, 358–375
doi:10.1017/S0003055417000570

Reading Between the Lines: Prediction of Political Violence Using Newspaper Text

HANNES MUELLER *Institut d'Analisi Economica*
CHRISTOPHER RAUH *University of Montreal*

This article provides a new methodology to predict armed conflict by using newspaper text. Through machine learning, vast quantities of newspaper text are reduced to interpretable topics. These topics are then used in panel regressions to predict the onset of conflict. We propose the use of the within-country variation of these topics to predict the timing of conflict. This allows us to avoid the tendency of predicting conflict only in countries where it occurred before. We show that the within-country variation of topics is a good predictor of conflict and becomes particularly useful when risk in previously peaceful countries arises. Two aspects seem to be responsible for these features. Topics provide depth because they consist of changing, long lists of terms that make them able to capture the changing context of conflict. At the same time, topics provide width because they are summaries of the full text, including stabilizing factors.

The conflict literature has made significant progress in understanding which countries are more at risk of suffering an armed conflict.[1] However, many factors that have been identified as leading to increased risk, like mountainous terrain or ethnic polarization, are time invariant or very slow moving, and therefore not useful in predicting the timing of conflict. Other factors, like GDP levels or political institutions, still vary more between countries than within countries over time. This means it is easier to predict whether a country is at risk *in general* rather than *when* a country is particularly at risk. Yet, understanding the timing of conflict is critical for policy.

An additional problem of forecasting the timing of armed conflict is that it is rare and at the same time relatively concentrated in some countries. This is problematic because it implies that the variation between countries can dominate the analysis unless the between- and within-country variations are separated explicitly. Empirical models that are overall quite accurate can therefore be of little use on the time dimension. We show, using a simple panel regression model, that many variables commonly used in the literature indeed face this problem. This means they predict conflict where it occurred before, and therefore fail to predict conflicts in previously peaceful countries.

As a solution to this problem, we propose data generated from news sources. To this end, we implement an automated method to quantify the content of news using the latent Dirichlet allocation (LDA) model (Blei, Ng, and Jordan 2003), which we apply to over 700,000 newspaper articles from English-speaking newspapers. There are two advantages that topics have over existing methods of analyzing text. First, topics provide depth because, by design, they put words into context. The context can be useful for forecasting. Second, topics provide width because they allow us to use the whole text, including stabilizing factors, when forecasting conflict. This means we can let the data speak without losing interpretability of the results.

At the prediction stage, we rely on a simple panel regression model, which uses all generated topics as explanatory variables. The result is a model able to forecast out of sample the onset of civil war, armed conflict, and even movements of refugees a year before they occur. It relies entirely on news text and can therefore provide forecasts without the need to extrapolate or wait for other data sources. Furthermore, the procedure can be implemented with only minimal personal judg-

Hannes Mueller is a tenured scientist at IAE (CSIC), Barcelona GSE Institut d'Analisi Economica, CSIC Campus UAB, 08193 Bellaterra, Spain (h.mueller.uni@gmail.com).

Christopher Rauh is an Assistant Professor at University of Montreal, Département de Sciences Économiques, Université de Montréal, C.P.6128 succ. Centre-Ville, Montréal H3C 3J7, Canada (christopher.raphael.rauh@umontreal.ca).

We thank Tim Besley, Melissa Dell, Vincenzo Galasso, Hector Galindo, Matt Gentzkow, Stephen Hansen, Ethan Kapstein, Daniel Ohayon, Akash Raja, Bernhard Reinsberg, Anand Shrivastava, Ron Smith, Jack Willis, Stephane Wolton, and the participants of the workshops and conferences ENCoRe Barcelona, Political Econ-

Text analysis

- A large amount of social interaction occurs in texts:
  - Congressional speeches, press releases, newsletters,
  - Facebook posts, tweets. emails, cell phone records.
  - Newspapers, magazines, news broadcasts,
  - Treaties, sermons, fatwas
- Pre 2000's social scientists avoided using texts/speech.
- Why?

Text analysis

- A large amount of social interaction occurs in texts:
  - Congressional speeches, press releases, newsletters,
  - Facebook posts, tweets. emails, cell phone records.
  - Newspapers, magazines, news broadcasts,
  - Treaties, sermons, fatwas
- Pre 2000's social scientists avoided using texts/speech.
- Why?
  - Hard to find
  - Time consuming
  - Not generalizable (each new data set ... new coding scheme)
  - Difficult to store/search
  - Idiosyncratic to coders/researchers
  - Statistical methods/algorithms, computationally intensive

Text analysis

Today:

- Massive increase in availability of unstructured text: In 2017, the number of emails sent and received per day total over 260 billion.
- Cheap storage: 1981: $ 500,000 per GB. 2017: $ 0.03 per GB
- Explosion in methods and programs to analyze texts
  - Generalizable
  - Systematic
  - Cheap

Preprocessing Texts

One (of many) recipe for preprocessing: retain useful information

Preprocessing Texts

One (of many) recipe for preprocessing: retain useful information

1) Remove capitalization, punctuation

Preprocessing Texts

One (of many) recipe for preprocessing: retain useful information

1) Remove capitalization, punctuation
2) Discard Word Order: (Bag of Words Assumption)

Preprocessing Texts

One (of many) recipe for preprocessing: retain useful information

1) Remove capitalization, punctuation
2) Discard Word Order: (Bag of Words Assumption)
3) Discard stop words

Preprocessing Texts

One (of many) recipe for preprocessing: retain useful information

1) Remove capitalization, punctuation
2) Discard Word Order: (Bag of Words Assumption)
3) Discard stop words
4) Combine similar terms: Stem, Lemmatize

Preprocessing Texts

One (of many) recipe for preprocessing: retain useful information

1) Remove capitalization, punctuation
2) Discard Word Order: (Bag of Words Assumption)
3) Discard stop words
4) Combine similar terms: Stem, Lemmatize
5) Discard less useful features $\rightsquigarrow$ depends on application

Preprocessing Texts

One (of many) recipe for preprocessing: retain useful information

1) Remove capitalization, punctuation
2) Discard Word Order: (Bag of Words Assumption)
3) Discard stop words
4) Combine similar terms: Stem, Lemmatize
5) Discard less useful features $\rightsquigarrow$ depends on application
6) Other reduction, weighting

Preprocessing Texts

One (of many) recipe for preprocessing: retain useful information

1) Remove capitalization, punctuation
2) Discard Word Order: (Bag of Words Assumption)
3) Discard stop words
4) Combine similar terms: Stem, Lemmatize
5) Discard less useful features $\rightsquigarrow$ depends on application
6) Other reduction, weighting
7) Output: Count vector, each element counts occurrence of terms

Preprocessing Texts

One (of many) recipe for preprocessing: retain useful information

1) Remove capitalization, punctuation
2) Discard Word Order: (Bag of Words Assumption)
3) Discard stop words
4) Combine similar terms: Stem, Lemmatize
5) Discard less useful features $\rightsquigarrow$ depends on application
6) Other reduction, weighting
7) Output: Count vector, each element counts occurrence of terms

1. Remove capitalization, punctuation, numbers

Assumption: capitalization, punctuation does not provide useful information.

1. Remove capitalization, punctuation, numbers

Assumption: capitalization, punctuation does not provide useful information.

```
Now we are engaged in a great civil war, testing
whether that nation, or any nation
```

1. Remove capitalization, punctuation, numbers

Assumption: capitalization, punctuation does not provide useful information.

```
Now we are engaged in a great civil war, testing
whether that nation, or any nation
```

```
now we are engaged in a great civil war testing
whether that nation or any nation
```

1. Remove capitalization, punctuation, numbers

Assumption: capitalization, punctuation does not provide useful information.

`Now we are engaged in a great civil war, testing whether that nation, or any nation`

`now we are engaged in a great civil war testing whether that nation or any nation`

Caution
`‘‘Turkey’’ = ‘‘turkey’’`

2. Discard Word Order (Bag of Words) $\rightsquigarrow$ Tokenize

Assumption: Word Order Doesn't Matter.

2. Discard Word Order (Bag of Words) $\rightsquigarrow$ Tokenize

Assumption: Word Order Doesn't Matter.

`now we are engaged in a great civil war testing whether that nation or any nation`

2. Discard Word Order (Bag of Words) $\rightsquigarrow$ Tokenize

Assumption: Word Order Doesn't Matter.

`now we are engaged in a great civil war testing whether that nation or any nation`

`[now, we, are, engaged, in, a, great, civil, war, testing, whether, that, nation, or, any, nation]`

2. Discard Word Order (Bag of Words) $\rightsquigarrow$ Tokenize

Assumption: Word Order Doesn't Matter.

`now we are engaged in a great civil war testing whether that nation or any nation`

`[now, we, are, engaged, in, a, great, civil, war, testing, whether, that, nation, or, any, nation]`

`[a, any, are, civil, engaged, great, in, nation, now, or, testing, that, war, we, whether]`

2. Discard Word Order (Bag of Words) $\rightsquigarrow$ Tokenize

Assumption: Word Order Doesn't Matter.

```
now we are engaged in a great civil war testing
whether that nation or any nation
```

```
[now, we, are, engaged, in, a, great, civil, war,
testing, whether, that, nation, or, any, nation]
```

```
[a, any, are, civil, engaged, great, in, nation, now,
or, testing, that, war, we, whether]
```

Tokenization

Tokenization

Unigrams `now we are engaged in a great civil war testing whether that nation or any nation`

Tokenization

Unigrams now we are engaged in a great civil war testing whether that nation or any nation

Bigrams [now we, we are, are engaged, engaged in, in a, a great, great civil, civil war, war testing, testing whether, whether that, that nation, nation or, or any, any nation]

Tokenization

Unigrams now we are engaged in a great civil war testing whether that nation or any nation

Bigrams [now we, we are, are engaged, engaged in, in a, a great, great civil, civil war, war testing, testing whether, whether that, that nation, nation or, or any, any nation]

Trigrams [now we are, we are engaged, are engaged in, engaged in a, in a great, a great civil, great civil war, civil war testing, war testing whether, testing whether that, whether that nation, that nation or, nation or any, or any nation]

Document Term Matrix:

	T1	T2	T3	T4	T5	T6	T7	T8
Doc1	2	0	4	3	0	1	0	2
Doc2	0	2	4	0	2	3	0	0
Doc3	4	0	1	3	0	1	0	1
Doc4	0	1	0	2	0	0	1	0
Doc5	0	0	2	0	0	4	0	0
Doc6	1	1	0	2	0	1	1	3
Doc7	2	1	3	4	0	2	0	2

The problem of high dimensionality :

- Most traditional statistics for regression and classification are intended for the *low dimension* setting ($n \geq p$).

The problem of high dimensionality :

- Most traditional statistics for regression and classification are intended for the *low dimension* setting ($n \geq p$).
- New technologies allow to collect an almost unlimited number of feature measurements.

The problem of high dimensionality :

- Most traditional statistics for regression and classification are intended for the *low dimension* setting ($n \geq p$).
- New technologies allow to collect an almost unlimited number of feature measurements.
- Text analysis is an example of $n \leq p$.

The problem of high dimensionality :

- Most traditional statistics for regression and classification are intended for the *low dimension* setting ($n \geq p$).
- New technologies allow to collect an almost unlimited number of feature measurements.
- Text analysis is an example of $n \leq p$.
- What goes wrong in high dimensions?

The problem of high dimensionality :

- Most traditional statistics for regression and classification are intended for the *low dimension* setting ($n \geq p$).
- New technologies allow to collect an almost unlimited number of feature measurements.
- Text analysis is an example of $n \leq p$.
- What goes wrong in high dimensions?
  - There is no longer a unique least square coefficient estimate: the variance is infinite.
  - We might be including irrelevant variables.
  - Risk of overfitting.

Three methods to deal with high dimensionality:

- **Subset selection:** Identify a subset of the p predictors that we believe to be related to the response.
- **Shrinkage or regularization:** Fitting a model with all the p predictors. However, the estimated coefficients are shrunken towards zero.
- **Dimension reduction:** Projecting the p predictors into a M-dimensional subspace, where M ¡p.

Analyzing News Stories

New York Times Annotated Corpus

Analyzing News Stories

New York Times Annotated Corpus
November 1-3, 2004 (Day Before, Of, And After General Election)

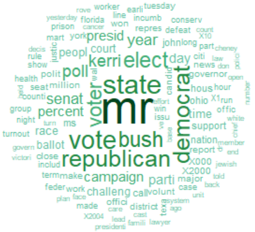
Analyzing News Stories

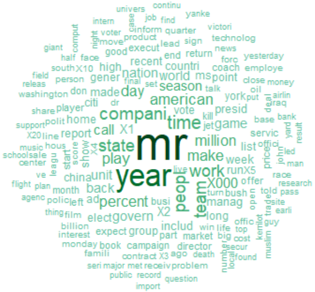
New York Times Annotated Corpus

November 1-3, 2004 (Day Before, Of, And After General Election)

We've preprocessed the data $\rightsquigarrow$ Create a Document-Term Matrix

Analyzing News Stories





R!

LASSO Regression

LASSO Regression

LASSO ("least absolute shrinkage and selection operator"): a regularization procedure that shrinks regression coefficients toward zero.

LASSO Regression

LASSO ("least absolute shrinkage and selection operator"): a regularization procedure that shrinks regression coefficients toward zero.

Document i, $(i = 1, \ldots, N)$

LASSO Regression

LASSO ("least absolute shrinkage and selection operator"): a regularization procedure that shrinks regression coefficients toward zero.

Document i, $(i = 1, \ldots, N)$
Count vector $\boldsymbol{x}_i$

LASSO Regression

LASSO ("least absolute shrinkage and selection operator"): a regularization procedure that shrinks regression coefficients toward zero.

Document i, $(i = 1, \ldots, N)$
Count vector $\boldsymbol{x}_i$
Labels Y_i

LASSO Regression

LASSO ("least absolute shrinkage and selection operator"): a regularization procedure that shrinks regression coefficients toward zero.

Document i, $(i = 1, \ldots, N)$
Count vector $\boldsymbol{x}_i$
Labels Y_i
Linear regression: Choose $\beta's$ to minimize sum of squared residuals

$$\boldsymbol{\beta}_{\text{OLS}} = \text{argmin}_{\boldsymbol{\beta}} \sum_{i=1}^{N} (Y_i - \boldsymbol{\beta} \cdot \boldsymbol{x}_i)^2$$

LASSO Regression

LASSO ("least absolute shrinkage and selection operator"): a regularization procedure that shrinks regression coefficients toward zero.

Document i, $(i = 1, \ldots, N)$
Count vector $\boldsymbol{x}_i$
Labels Y_i
LASSO Regression: Choose $\beta's$ to minimize sum of squared residuals and penalty on size of coefficients

LASSO Regression

LASSO ("least absolute shrinkage and selection operator"): a regularization procedure that shrinks regression coefficients toward zero.

Document i, $(i = 1, \ldots, N)$
Count vector $\boldsymbol{x}_i$
Labels Y_i
LASSO Regression: Choose $\beta's$ to minimize sum of squared residuals and penalty on size of coefficients

$$\boldsymbol{\beta}_{\text{LASSO}} = \text{argmin}_{\boldsymbol{\beta}} \sum_{i=1}^{N} (Y_i - \boldsymbol{\beta} \cdot \boldsymbol{x}_i)^2 + \lambda \underbrace{\sum_{p=1}^{P} |\beta_p|}_{\text{penalty}}$$

LASSO Regression

LASSO ("least absolute shrinkage and selection operator"): a regularization procedure that shrinks regression coefficients toward zero.

Document i, $(i = 1, \ldots, N)$
Count vector $\boldsymbol{x}_i$
Labels Y_i
LASSO Regression: Choose $\beta's$ to minimize sum of squared residuals and penalty on size of coefficients

$$\boldsymbol{\beta}_{\text{LASSO}} = \text{argmin}_{\boldsymbol{\beta}} \sum_{i=1}^{N} (Y_i - \boldsymbol{\beta} \cdot \boldsymbol{x}_i)^2 + \lambda \underbrace{\sum_{p=1}^{P} |\beta_p|}_{\text{penalty}}$$

What does λ do?

LASSO Penalty: Algebra

Why does LASSO shrink coefficients to 0?

LASSO Penalty: Algebra

Why does LASSO shrink coefficients to 0?

Contrast $\beta = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ and $\tilde{\beta} = (1, 0)$

LASSO Penalty: Algebra

Why does LASSO shrink coefficients to 0?

Contrast $\beta = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ and $\tilde{\beta} = (1, 0)$

Penalty for Coefficients:

LASSO Penalty: Algebra

Why does LASSO shrink coefficients to 0?

Contrast $\beta = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ and $\tilde{\beta} = (1, 0)$

Penalty for Coefficients:

$$\sum_{j=1}^{2} |\beta_j| = \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}} = \sqrt{2}$$

LASSO Penalty: Algebra

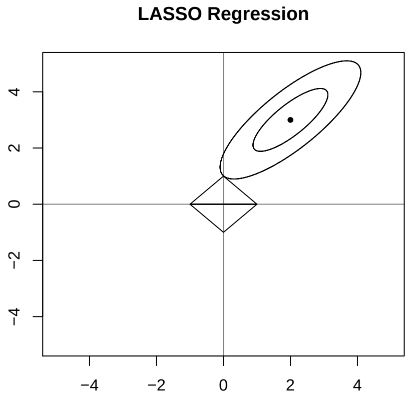
Why does LASSO shrink coefficients to 0?

Contrast $\beta = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ and $\tilde{\beta} = (1, 0)$

Penalty for Coefficients:

$$\begin{aligned} \sum_{j=1}^{2} |\beta_j| &= \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{2}} = \sqrt{2} \\ \sum_{j=1}^{2} |\tilde{\beta}_j| &= 1 + 0 = 1 \end{aligned}$$

LASSO Penalty: Geometry



R!

NEXT

- More on LASSO
- Homework 4 (*available on Wed 1st due on Wed 8th*)
- ggplot and data manipulation workshop (*Friday August 3rd*)