5. Supervised Techniques II

DS-GA 1015, Text as Data
Arthur Spirling

March 9, 2021

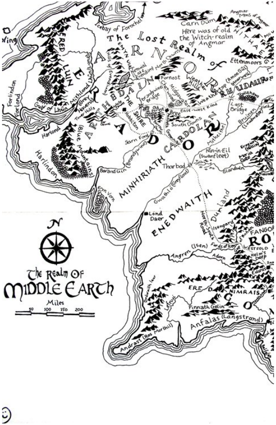
Where Are We?

Where Are We?

The Lost Realm of
ARNOR
Carn Dûm
Here was of old the Witch-realm of Angmar
Ettenmoors
Weather Hills
Fornost
North Downs
Bree
East-West Road
South Downs
Last Bridge
THE SHIRE
Sarn Ford
CARDOLAN
Mitheithel
Tharbad
Nîn-in-Eilph (Swanfleet)
Glanduin
MINHIRIATH
Eryn Vorn
Lond Daer
ENEDWAITH
Dunland
Isengard
Angren (Isen)
Adorn
Helm's Deep
Forlindon
Forlond
Gulf of Lhûn
Harlond
Harlindon
ERED NIMRAIS
Pinnath Gelin
Anfalas (Langstrand)
Andrast (Ras Morthil)
The Realm Of Middle Earth
Miles

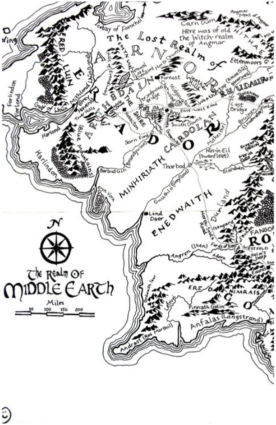
☺

Where Are We?



Covered dictionary and related approaches to document classifications

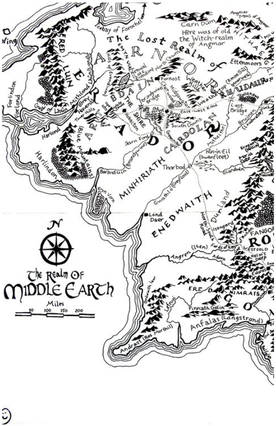
Where Are We?



Covered dictionary and related approaches to document classifications

Continue this idea,

Where Are We?



Covered dictionary and related approaches to document classifications

Continue this idea, but in a more formal modeling way: Naive Bayes

Where Are We?

Covered dictionary and related approaches to document classifications

Continue this idea, but in a more formal modeling way: Naive Bayes

and look at ways to classify/scale specifically political texts.

Where Are We?

Covered dictionary and related approaches to document classifications

Continue this idea, but in a more formal modeling way: Naive Bayes

and look at ways to classify/scale specifically political texts.

also consider ways to estimate proportions of documents in different categories.

Where Are We?

Covered dictionary and related approaches to document classifications

Continue this idea, but in a more formal modeling way: Naive Bayes

and look at ways to classify/scale specifically political texts.

also consider ways to estimate proportions of documents in different categories.

plus opportunities for fast, reliable coding of training set.

Remember. . .

Remember. . .

Unsupervised techniques:

Remember...

Unsupervised techniques: learning (hidden or latent) structure in unlabeled data.

e.g. PCA of legislators's votes:

Remember...

Unsupervised techniques: learning (hidden or latent) structure in unlabeled data.

e.g. PCA of legislators's votes: want to see how they are organized—

Remember...

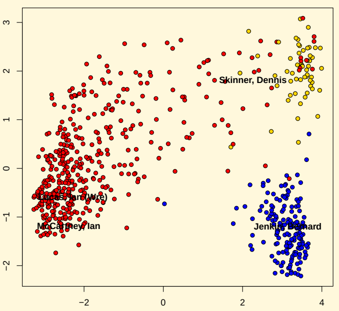
Unsupervised techniques: learning (hidden or latent) structure in unlabeled data.

e.g. PCA of legislators's votes: want to see how they are organized—by party? by ideology? by race?

Remember...

Unsupervised techniques: learning (hidden or latent) structure in unlabeled data.

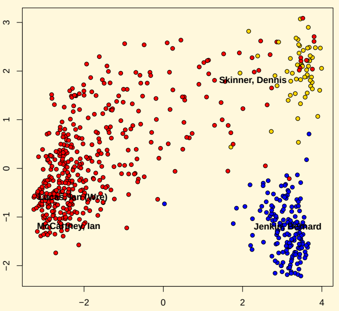
e.g. PCA of legislators's votes: want to see how they are organized—by party? by ideology? by race?



Remember. . .

Unsupervised techniques: learning (hidden or latent) structure in unlabeled data.

e.g. PCA of legislators's votes: want to see how they are organized—by party? by ideology? by race?

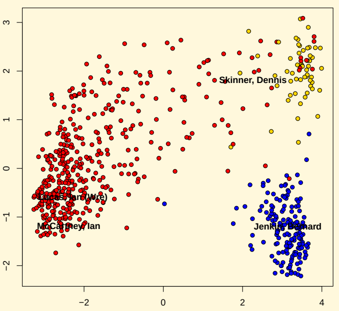


Supervised techniques:

Remember...

Unsupervised techniques: learning (hidden or latent) structure in unlabeled data.

e.g. PCA of legislators's votes: want to see how they are organized—by party? by ideology? by race?

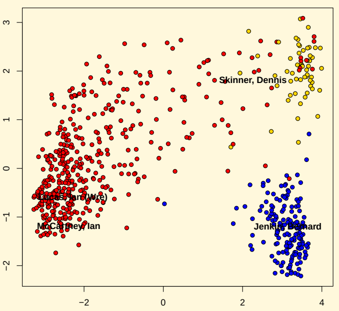


Supervised techniques: learning relationship between inputs and a labeled set of outputs.

Remember. . .

Unsupervised techniques: learning (hidden or latent) structure in unlabeled data.

e.g. PCA of legislators's votes: want to see how they are organized—by party? by ideology? by race?



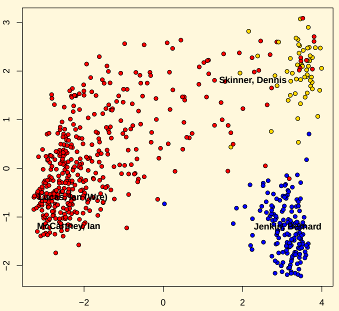
Supervised techniques: learning relationship between inputs and a labeled set of outputs.

e.g. opinion mining:

Remember. . .

Unsupervised techniques: learning (hidden or latent) structure in unlabeled data.

e.g. PCA of legislators's votes: want to see how they are organized—by party? by ideology? by race?



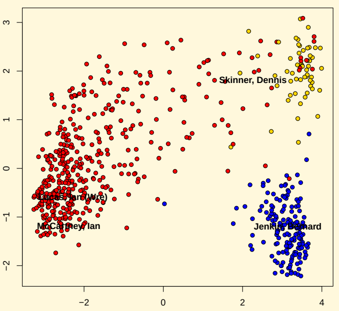
Supervised techniques: learning relationship between inputs and a labeled set of outputs.

e.g. opinion mining: what makes a critic like or dislike a movie ($y \in \{0, 1\}$)?

Remember. . .

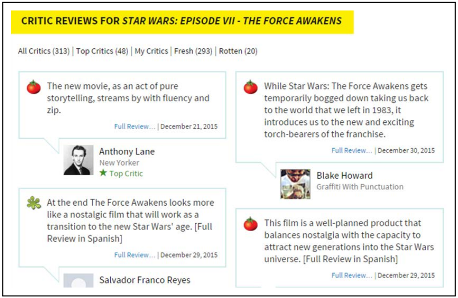
Unsupervised techniques: learning (hidden or latent) structure in unlabeled data.

e.g. PCA of legislators's votes: want to see how they are organized—by party? by ideology? by race?



Supervised techniques: learning relationship between inputs and a labeled set of outputs.

e.g. opinion mining: what makes a critic like or dislike a movie ($y \in \{0, 1\}$)?



CRITIC REVIEWS FOR *STAR WARS: EPISODE VII - THE FORCE AWAKENS*

All Critics (313) | Top Critics (48) | My Critics | Fresh (293) | Rotten (20)

The new movie, as an act of pure storytelling, streams by with fluency and zip.

Full Review... | December 21, 2015

Anthony Lane
New Yorker
★ Top Critic

While Star Wars: The Force Awakens gets temporarily bogged down taking us back to the world that we left in 1983, it introduces us to the new and exciting torch-bearers of the franchise.

Full Review... | December 30, 2015

Blake Howard
Graffiti With Punctuation

At the end The Force Awakens looks more like a nostalgic film that will work as a transition to the new Star Wars' age. [Full Review in Spanish]

Full Review... | December 29, 2015

Salvador Franco Reyes

This film is a well-planned product that balances nostalgia with the capacity to attract new generations into the Star Wars universe. [Full Review in Spanish]

Full Review... | December 29, 2015

Naive Bayes

Naive Bayes Classification

Naive Bayes Classification

Motivation: emails d arrive and must classified as belonging to one of two classes $c \in \{\text{spam}, \text{ham}\}$.

Naive Bayes Classification

Motivation: emails d arrive and must classified as belonging to one of two classes $c \in \{\text{spam}, \text{ham}\}$.

by using the words/features frequencies the emails contain.

Naive Bayes Classification

Motivation: emails d arrive and must classified as belonging to one of two classes $c \in \{\text{spam}, \text{ham}\}$.

by using the words/features frequencies the emails contain.

use Naive Bayes,

Naive Bayes Classification

Motivation: emails d arrive and must classified as belonging to one of two classes $c \in \{\text{spam},\text{ham}\}$.

by using the words/features frequencies the emails contain.

use Naive Bayes, also simple Bayes, or independence Bayes,

Naive Bayes Classification

Motivation: emails d arrive and must classified as belonging to one of two classes $c \in \{\text{spam}, \text{ham}\}$.

by using the words/features frequencies the emails contain.

use Naive Bayes, also simple Bayes, or independence Bayes,

is a family of classifiers which apply Bayes's theorem and make 'naive' assumptions about independence between the features of a document.

Naive Bayes Classification

Motivation: emails d arrive and must classified as belonging to one of two classes $c \in \{$spam,ham$\}$.

by using the words/features frequencies the emails contain.

use Naive Bayes, also simple Bayes, or independence Bayes,

is a family of classifiers which apply Bayes's theorem and make 'naive' assumptions about independence between the features of a document.

$\rightarrow$ fast, simple, accurate, efficient and therefore popular.

Set up

Set up

We're interested in the probability that an email is in a given category,

Set up

We're interested in the probability that an email is in a given category, given its features—i.e. frequency of terms.

Set up

We're interested in the probability that an email is in a given category, given its features—i.e. frequency of terms.

The conditional probability of a term t_k occurring in a document, given that document is of class c, is

Set up

We're interested in the probability that an email is in a given category, given its features—i.e. frequency of terms.

The conditional probability of a term t_k occurring in a document, given that document is of class c, is $= \Pr(t_k|c)$

Set up

We're interested in the probability that an email is in a given category, given its features—i.e. frequency of terms.

The conditional probability of a term t_k occurring in a document, given that document is of class c, is $= \Pr(t_k|c)$

e.g. probability of seeing '`beneficiary`' in a spam email might be 0.9,

Set up

We're interested in the probability that an email is in a given category, given its features—i.e. frequency of terms.

The conditional probability of a term t_k occurring in a document, given that document is of class c, is $= \Pr(t_k|c)$

e.g. probability of seeing '`beneficiary`' in a spam email might be 0.9, because a lot of spam emails use that term.

Set up

We're interested in the probability that an email is in a given category, given its features—i.e. frequency of terms.

The conditional probability of a term t_k occurring in a document, given that document is of class c, is $= \Pr(t_k|c)$

e.g. probability of seeing '`beneficiary`' in a spam email might be 0.9, because a lot of spam emails use that term.

NB we are assuming terms basically occur randomly throughout the document/no position effects

Set up

We're interested in the probability that an email is in a given category, given its features—i.e. frequency of terms.

The conditional probability of a term t_k occurring in a document, given that document is of class c, is $= \Pr(t_k|c)$

e.g. probability of seeing '`beneficiary`' in a spam email might be 0.9, because a lot of spam emails use that term.

NB we are assuming terms basically occur randomly throughout the document/no position effects

We can write the probability that a given email d contains all the terms,

Set up

We're interested in the probability that an email is in a given category, given its features—i.e. frequency of terms.

The conditional probability of a term t_k occurring in a document, given that document is of class c, is $= \Pr(t_k|c)$

e.g. probability of seeing '`beneficiary`' in a spam email might be 0.9, because a lot of spam emails use that term.

NB we are assuming terms basically occur randomly throughout the document/no position effects

We can write the probability that a given email d contains all the terms, if it's from a class c, as

Set up

We're interested in the probability that an email is in a given category, given its features—i.e. frequency of terms.

The conditional probability of a term t_k occurring in a document, given that document is of class c, is $= \Pr(t_k|c)$

e.g. probability of seeing '`beneficiary`' in a spam email might be 0.9, because a lot of spam emails use that term.

NB we are assuming terms basically occur randomly throughout the document/no position effects

We can write the probability that a given email d contains all the terms, if it's from a class c, as

$$\Pr(d|c) = \prod_{k=1}^{K} \Pr(t_k|c)$$

Set up

We're interested in the probability that an email is in a given category, given its features—i.e. frequency of terms.

The conditional probability of a term t_k occurring in a document, given that document is of class c, is $= \Pr(t_k|c)$

e.g. probability of seeing '`beneficiary`' in a spam email might be 0.9, because a lot of spam emails use that term.

NB we are assuming terms basically occur randomly throughout the document/no position effects

We can write the probability that a given email d contains all the terms, if it's from a class c, as

$$\Pr(d|c) = \prod_{k=1}^{K} \Pr(t_k|c)$$

but this is not what we want:

Set up

We're interested in the probability that an email is in a given category, given its features—i.e. frequency of terms.

The conditional probability of a term t_k occurring in a document, given that document is of class c, is $= \Pr(t_k|c)$

e.g. probability of seeing '`beneficiary`' in a spam email might be 0.9, because a lot of spam emails use that term.

NB we are assuming terms basically occur randomly throughout the document/no position effects

We can write the probability that a given email d contains all the terms, if it's from a class c, as

$$\Pr(d|c) = \prod_{k=1}^{K} \Pr(t_k|c)$$

but this is not what we want: we want $\Pr(c|d)$.

Reminder: Bayes' Theorem

Reminder: Bayes' Theorem

Reminder: Bayes' Theorem

Recall that:

Reminder: Bayes' Theorem

Recall that:

$$\Pr(A|B) = \frac{\Pr(A, B)}{\Pr(B)}$$

Reminder: Bayes' Theorem

Recall that:

$$\Pr(A|B) = \frac{\Pr(A, B)}{\Pr(B)}$$

- the probability that A occurs given that B occurred $=$ the probability of both A and B occurring, divided by the probability that B occurs.

Reminder: Bayes' Theorem

Recall that:

$$\Pr(A|B) = \frac{\Pr(A, B)}{\Pr(B)}$$

- the probability that A occurs given that B occurred $=$ the probability of both A and B occurring, divided by the probability that B occurs.

e.g. you know a die shows an odd number, what is the probability that this odd number is 3?

Reminder: Bayes' Theorem

Recall that:

$$\Pr(A|B) = \frac{\Pr(A,B)}{\Pr(B)}$$

- the probability that A occurs given that B occurred $=$ the probability of both A and B occurring, divided by the probability that B occurs.

e.g. you know a die shows an odd number, what is the probability that this odd number is 3? $\Pr(3|\text{odd}) = \frac{\frac{1}{6}}{\frac{1}{2}}$

Reminder: Bayes' Theorem

Recall that:

$$\Pr(A|B) = \frac{\Pr(A,B)}{\Pr(B)}$$

- the probability that A occurs given that B occurred $=$ the probability of both A and B occurring, divided by the probability that B occurs.

e.g. you know a die shows an odd number, what is the probability that this odd number is 3? $\Pr(3|\text{odd}) = \frac{\frac{1}{6}}{\frac{1}{2}} = \frac{1}{3}$.

Reminder: Bayes' Theorem

Recall that:

$$\Pr(A|B) = \frac{\Pr(A, B)}{\Pr(B)}$$

- the probability that A occurs given that B occurred = the probability of both A and B occurring, divided by the probability that B occurs.

e.g. you know a die shows an odd number, what is the probability that this odd number is 3? $\Pr(3|\text{odd}) = \frac{\frac{1}{6}}{\frac{1}{2}} = \frac{1}{3}$.

- of course, it is also true that $\Pr(B|A) = \frac{\Pr(B,A)}{\Pr(A)}$.

Reminder: Bayes' Theorem

Recall that:

$$\Pr(A|B) = \frac{\Pr(A, B)}{\Pr(B)}$$

- the probability that A occurs given that B occurred = the probability of both A and B occurring, divided by the probability that B occurs.

e.g. you know a die shows an odd number, what is the probability that this odd number is 3? $\Pr(3|\text{odd}) = \frac{\frac{1}{6}}{\frac{1}{2}} = \frac{1}{3}$.

- of course, it is also true that $\Pr(B|A) = \frac{\Pr(B,A)}{\Pr(A)}$.
- but then, since $\Pr(A, B) = \Pr(B, A)$, we must have $\Pr(A|B)\Pr(B) = \Pr(B|A)\Pr(A)$, and thus…

Reminder: Bayes' Theorem

Recall that:

$$\Pr(A|B) = \frac{\Pr(A, B)}{\Pr(B)}$$

- the probability that A occurs given that B occurred = the probability of both A and B occurring, divided by the probability that B occurs.

e.g. you know a die shows an odd number, what is the probability that this odd number is 3? $\Pr(3|\text{odd}) = \frac{\frac{1}{6}}{\frac{1}{2}} = \frac{1}{3}$.

- of course, it is also true that $\Pr(B|A) = \frac{\Pr(B,A)}{\Pr(A)}$.
- but then, since $\Pr(A, B) = \Pr(B, A)$, we must have $\Pr(A|B)\Pr(B) = \Pr(B|A)\Pr(A)$, and thus... Bayes' law

Reminder: Bayes' Theorem

Recall that:

$$\Pr(A|B) = \frac{\Pr(A, B)}{\Pr(B)}$$

- the probability that A occurs given that B occurred $=$ the probability of both A and B occurring, divided by the probability that B occurs.

e.g. you know a die shows an odd number, what is the probability that this odd number is 3? $\Pr(3|\text{odd}) = \frac{\frac{1}{6}}{\frac{1}{2}} = \frac{1}{3}$.

- of course, it is also true that $\Pr(B|A) = \frac{\Pr(B,A)}{\Pr(A)}$.
- but then, since $\Pr(A, B) = \Pr(B, A)$, we must have $\Pr(A|B)\Pr(B) = \Pr(B|A)\Pr(A)$, and thus... Bayes' law

$$\boxed{\Pr(A|B) = \frac{\Pr(A)\Pr(B|A)}{\Pr(B)}.}$$

And. . .

And...

- interest is in $\Pr(A|B) = \frac{\Pr(A)\Pr(B|A)}{\Pr(B)}$.

And...

- interest is in $\Pr(A|B) = \frac{\Pr(A)\Pr(B|A)}{\Pr(B)}$.

- Notice that $\Pr(B)$ itself does not tell us whether a particular value of A is more or less likely to be observed,

And...

- interest is in $\Pr(A|B) = \frac{\Pr(A)\Pr(B|A)}{\Pr(B)}$.
- Notice that $\Pr(B)$ itself does not tell us whether a particular value of A is more or less likely to be observed, so drop it and rewrite:

And...

- interest is in $\Pr(A|B) = \frac{\Pr(A)\Pr(B|A)}{\Pr(B)}$.

- Notice that $\Pr(B)$ itself does not tell us whether a particular value of A is more or less likely to be observed, so drop it and rewrite:

$$\Pr(A|B) \propto \Pr(A)\Pr(B|A)$$

Here, $\Pr(A)$ is our prior for A, while $\Pr(B|A)$ will be the likelihood for the data we saw.

So. . .

So. . .

We can express our quantity of interest as:

So. . .

We can express our quantity of interest as:

$$\Pr(c|d) = \frac{\Pr(c)\Pr(d|c)}{\Pr(d)}$$

So. . .

We can express our quantity of interest as:

$$\Pr(c|d) = \frac{\Pr(c)\Pr(d|c)}{\Pr(d)}$$

and

$$\Pr(c|d) \propto \underbrace{\Pr(c)}_{\text{prior}}$$

So...

We can express our quantity of interest as:

$$\Pr(c|d) = \frac{\Pr(c)\Pr(d|c)}{\Pr(d)}$$

and

$$\Pr(c|d) \propto \underbrace{\Pr(c)}_{\text{prior}} \underbrace{\prod_{k=1}^{K} \Pr(t_k|c)}_{\text{likelihood}}$$

So...

We can express our quantity of interest as:

$$\Pr(c|d) = \frac{\Pr(c)\Pr(d|c)}{\Pr(d)}$$

and

$$\Pr(c|d) \propto \underbrace{\Pr(c)}_{\text{prior}} \underbrace{\prod_{k=1}^{K} \Pr(t_k|c)}_{\text{likelihood}}$$

where $\Pr(c)$ is the prior probability of a document occurring in class c;

So...

We can express our quantity of interest as:

$$\Pr(c|d) = \frac{\Pr(c)\Pr(d|c)}{\Pr(d)}$$

and

$$\Pr(c|d) \propto \underbrace{\Pr(c)}_{\text{prior}} \underbrace{\prod_{k=1}^{K} \Pr(t_k|c)}_{\text{likelihood}}$$

where $\Pr(c)$ is the prior probability of a document occurring in class c; and $\Pr(t_k|c)$ is interpreted as "measure of the how much evidence t_k contributes that c is the correct class"

Goal

Goal

We want to classify new data,

Goal

We want to classify new data, based on patterns we observe in our training set (which we will classify by hand).

Goal

We want to classify new data, based on patterns we observe in our training set (which we will classify by hand).

e.g. We look at 10,000 emails to this point in time, and classify them as $c \in \{\text{spam}, \text{ham}\}$.

Goal

We want to classify new data, based on patterns we observe in our training set (which we will classify by hand).

e.g. We look at 10,000 emails to this point in time, and classify them as $c \in \{\text{spam}, \text{ham}\}$. We use that information, and the terms associated with the two classes,

Goal

We want to classify new data, based on patterns we observe in our training set (which we will classify by hand).

e.g. We look at 10,000 emails to this point in time, and classify them as $c \in \{\text{spam}, \text{ham}\}$. We use that information, and the terms associated with the two classes, to categorize tomorrow's email.

Goal

We want to classify new data, based on patterns we observe in our training set (which we will classify by hand).

e.g. We look at 10,000 emails to this point in time, and classify them as $c \in \{\text{spam}, \text{ham}\}$. We use that information, and the terms associated with the two classes, to categorize tomorrow's email.

In particular, we typically want to assign the document to a single best class.

Goal

We want to classify new data, based on patterns we observe in our training set (which we will classify by hand).

e.g. We look at 10,000 emails to this point in time, and classify them as $c \in \{\text{spam}, \text{ham}\}$. We use that information, and the terms associated with the two classes, to categorize tomorrow's email.

In particular, we typically want to assign the document to a single best class.

→ The 'best' class is the maximum a posteriori class, c_{map}:

Goal

We want to classify new data, based on patterns we observe in our training set (which we will classify by hand).

e.g. We look at 10,000 emails to this point in time, and classify them as $c \in \{\text{spam}, \text{ham}\}$. We use that information, and the terms associated with the two classes, to categorize tomorrow's email.

In particular, we typically want to assign the document to a single best class.

→ The 'best' class is the maximum a posteriori class, c_{map}:

$$c_{map} = \arg\max_c \widehat{\Pr(c|d)}$$

Goal

We want to classify new data, based on patterns we observe in our training set (which we will classify by hand).

e.g. We look at 10,000 emails to this point in time, and classify them as $c \in \{\text{spam}, \text{ham}\}$. We use that information, and the terms associated with the two classes, to categorize tomorrow's email.

In particular, we typically want to assign the document to a single best class.

→ The 'best' class is the maximum a posteriori class, c_{map}:

$$c_{map} = \arg\max_c \widehat{\Pr(c|d)} = \arg\max_c \widehat{\Pr(c)} \prod_{k=1}^{K} \widehat{\Pr(t_k|c)}$$

Estimation Notes I

Estimation Notes I

The 'hats' appear because neither $\widehat{\Pr(c)}$ nor $\widehat{\Pr(t_k|c)}$ are known.

Estimation Notes I

The 'hats' appear because neither $\widehat{\Pr(c)}$ nor $\widehat{\Pr(t_k|c)}$ are known.

$\rightarrow$ they are (can be) estimated from the training set.

Estimation Notes I

The 'hats' appear because neither $\widehat{\Pr(c)}$ nor $\widehat{\Pr(t_k|c)}$ are known.

$\rightarrow$ they are (can be) estimated from the training set.

We can use $\frac{N_c}{N}$ for $\widehat{\Pr(c)}$,

Estimation Notes I

The 'hats' appear because neither $\widehat{\Pr(c)}$ nor $\widehat{\Pr(t_k|c)}$ are known.

$\rightarrow$ they are (can be) estimated from the training set.

We can use $\frac{N_c}{N}$ for $\widehat{\Pr(c)}$, where N_c is the number of documents in class c in our training set (MLE).

Estimation Notes I

The 'hats' appear because neither $\widehat{\Pr(c)}$ nor $\widehat{\Pr(t_k|c)}$ are known.

$\rightarrow$ they are (can be) estimated from the training set.

We can use $\frac{N_c}{N}$ for $\widehat{\Pr(c)}$, where N_c is the number of documents in class c in our training set (MLE).

We can use $\frac{T_{ct}}{\sum_{t' \in V} T_{ct'}}$ for $\widehat{\Pr(t_k|c)}$ (MLE).

Estimation Notes I

The 'hats' appear because neither $\widehat{\Pr(c)}$ nor $\widehat{\Pr(t_k|c)}$ are known.

$\rightarrow$ they are (can be) estimated from the training set.

We can use $\frac{N_c}{N}$ for $\widehat{\Pr(c)}$, where N_c is the number of documents in class c in our training set (MLE).

We can use $\frac{T_{ct}}{\sum_{t' \in V} T_{ct'}}$ for $\widehat{\Pr(t_k|c)}$ (MLE).

here T_{ct} is the number of occurrences of t in training documents that come from class c, including multiple occurrences.

Estimation Notes I

The 'hats' appear because neither $\widehat{\Pr(c)}$ nor $\widehat{\Pr(t_k|c)}$ are known.

$\rightarrow$ they are (can be) estimated from the training set.

We can use $\frac{N_c}{N}$ for $\widehat{\Pr(c)}$, where N_c is the number of documents in class c in our training set (MLE).

We can use $\frac{T_{ct}}{\sum_{t' \in V} T_{ct'}}$ for $\widehat{\Pr(t_k|c)}$ (MLE).

here T_{ct} is the number of occurrences of t in training documents that come from class c, including multiple occurrences.

and denominator is the total number of all terms in the training documents in c.

Example

Example

	email	words	classification
training	1	money inherit prince	spam
	2	prince inherit amount	spam

Example

	email	words	classification
training	1	money inherit prince	spam
	2	prince inherit amount	spam
	3	inherit plan money	ham
	4	cost amount amazon	ham
	5	prince william news	ham

Example

	email	words	classification
training	1	money inherit prince	spam
	2	prince inherit amount	spam
	3	inherit plan money	ham
	4	cost amount amazon	ham
	5	prince william news	ham
test	6	prince prince money	?

Example

	email	words	classification
training	1	money inherit prince	spam
	2	prince inherit amount	spam
	3	inherit plan money	ham
	4	cost amount amazon	ham
	5	prince william news	ham
test	6	prince prince money	?

$\Pr(\text{prince}|\text{ham}) = \frac{1}{9}$
$\Pr(\text{prince}|\text{ham}) = \frac{1}{9}$
$\Pr(\text{money}|\text{ham}) = \frac{1}{9}$

Example

	email	words	classification
training	1	money inherit prince	spam
	2	prince inherit amount	spam
	3	inherit plan money	ham
	4	cost amount amazon	ham
	5	prince william news	ham
test	6	prince prince money	?

$$\Pr(\text{prince}|\text{ham}) = \frac{1}{9}$$
$$\Pr(\text{prince}|\text{ham}) = \frac{1}{9}$$
$$\Pr(\text{money}|\text{ham}) = \frac{1}{9}$$
$$\Pr(\text{ham}|\text{d}) \propto \frac{3}{5}\frac{1}{9}\frac{1}{9}\frac{1}{9} = 0.00082$$

Example

	email	words	classification
training	1	money inherit prince	spam
	2	prince inherit amount	spam
	3	inherit plan money	ham
	4	cost amount amazon	ham
	5	prince william news	ham
test	6	prince prince money	?

$$\Pr(\text{prince}|\text{ham}) = \frac{1}{9}$$
$$\Pr(\text{prince}|\text{ham}) = \frac{1}{9}$$
$$\Pr(\text{money}|\text{ham}) = \frac{1}{9}$$
$$\Pr(\text{ham}|\text{d}) \propto \frac{3}{5}\frac{1}{9}\frac{1}{9}\frac{1}{9} = 0.00082$$

$$\Pr(\text{prince}|\text{spam}) = \frac{2}{6}$$
$$\Pr(\text{prince}|\text{spam}) = \frac{2}{6}$$
$$\Pr(\text{money}|\text{spam}) = \frac{1}{6}$$

Example

	email	words	classification
training	1	money inherit prince	spam
	2	prince inherit amount	spam
	3	inherit plan money	ham
	4	cost amount amazon	ham
	5	prince william news	ham
test	6	prince prince money	?

$$\Pr(\text{prince}|\text{ham}) = \frac{1}{9}$$
$$\Pr(\text{prince}|\text{ham}) = \frac{1}{9}$$
$$\Pr(\text{money}|\text{ham}) = \frac{1}{9}$$
$$\Pr(\text{ham}|\text{d}) \propto \frac{3}{5}\frac{1}{9}\frac{1}{9}\frac{1}{9} = 0.00082$$

$$\Pr(\text{prince}|\text{spam}) = \frac{2}{6}$$
$$\Pr(\text{prince}|\text{spam}) = \frac{2}{6}$$
$$\Pr(\text{money}|\text{spam}) = \frac{1}{6}$$
$$\Pr(\text{spam}|\text{d}) \propto \frac{2}{5}\frac{2}{6}\frac{2}{6}\frac{1}{6} = 0.0074$$

Example

	email	words	classification
training	1	money inherit prince	spam
	2	prince inherit amount	spam
	3	inherit plan money	ham
	4	cost amount amazon	ham
	5	prince william news	ham
test	6	prince prince money	?

$$\Pr(\text{prince}|\text{ham}) = \frac{1}{9}$$
$$\Pr(\text{prince}|\text{ham}) = \frac{1}{9}$$
$$\Pr(\text{money}|\text{ham}) = \frac{1}{9}$$
$$\Pr(\text{ham}|d) \propto \frac{3}{5}\frac{1}{9}\frac{1}{9}\frac{1}{9} = 0.00082$$

$$\Pr(\text{prince}|\text{spam}) = \frac{2}{6}$$
$$\Pr(\text{prince}|\text{spam}) = \frac{2}{6}$$
$$\Pr(\text{money}|\text{spam}) = \frac{1}{6}$$
$$\Pr(\text{spam}|d) \propto \frac{2}{5}\frac{2}{6}\frac{2}{6}\frac{1}{6} = 0.0074$$

$$\rightarrow \boxed{c_{map} = \text{spam}}$$

Estimation Notes II

$\widehat{\Pr(t_k|c)}$ is the fraction of tokens in documents from class c that are t.

Estimation Notes II

$\widehat{\Pr(t_k|c)}$ is the fraction of tokens in documents from class c that are t. Can also see it as fraction of positions in documents from class c that contain term t.

Estimation Notes II

$\widehat{\Pr(t_k|c)}$ is the fraction of tokens in documents from class c that are t. Can also see it as fraction of positions in documents from class c that contain term t. This is a multinomial NB model.

Estimation Notes II

$\widehat{\Pr(t_k|c)}$ is the fraction of tokens in documents from class c that are t. Can also see it as fraction of positions in documents from class c that contain term t. This is a multinomial NB model.

aside Could have $\widehat{\Pr(t_k|c)}$ as fraction of documents containing t,

Estimation Notes II

$\widehat{\Pr(t_k|c)}$ is the fraction of tokens in documents from class c that are t. Can also see it as fraction of positions in documents from class c that contain term t. This is a multinomial NB model.

aside Could have $\widehat{\Pr(t_k|c)}$ as fraction of documents containing t, which implies Bernoulli model (ignores number of occurrences).

Estimation Notes II

$\widehat{\Pr(t_k|c)}$ is the fraction of tokens in documents from class c that are t. Can also see it as fraction of positions in documents from class c that contain term t. This is a multinomial NB model.

aside Could have $\widehat{\Pr(t_k|c)}$ as fraction of documents containing t, which implies Bernoulli model (ignores number of occurrences).

As usual,

Estimation Notes II

$\widehat{\Pr(t_k|c)}$ is the fraction of tokens in documents from class c that are t. Can also see it as fraction of positions in documents from class c that contain term t. This is a multinomial NB model.

aside Could have $\widehat{\Pr(t_k|c)}$ as fraction of documents containing t, which implies Bernoulli model (ignores number of occurrences).

As usual, working with products of probabilities is difficult computationally

Estimation Notes II

$\widehat{\Pr(t_k|c)}$ is the fraction of tokens in documents from class c that are t. Can also see it as fraction of positions in documents from class c that contain term t. This is a multinomial NB model.

aside Could have $\widehat{\Pr(t_k|c)}$ as fraction of documents containing t, which implies Bernoulli model (ignores number of occurrences).

As usual, working with products of probabilities is difficult computationally

$\rightarrow$ take logs:

Estimation Notes II

$\widehat{\Pr(t_k|c)}$ is the fraction of tokens in documents from class c that are t. Can also see it as fraction of positions in documents from class c that contain term t. This is a multinomial NB model.

aside Could have $\widehat{\Pr(t_k|c)}$ as fraction of documents containing t, which implies Bernoulli model (ignores number of occurrences).

As usual, working with products of probabilities is difficult computationally

$\rightarrow$ take logs:

$$c_{map} = \arg\max_c \left[\log \widehat{\Pr(c)} + \sum \log \widehat{\Pr(t_k|c)}\right]$$

Estimation Notes III

Estimation Notes III

Sparsity can be a problem in the training set.

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails,

Estimation Notes III

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails, we never see the word ‘`cost`’ (but it does occur in the ham set),

Estimation Notes III

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails, we never see the word '`cost`' (but it does occur in the ham set), and it shows up in our actual email tomorrow.

Estimation Notes III

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails, we never see the word '`cost`' (but it does occur in the ham set), and it shows up in our actual email tomorrow.

Q What's the probability that email is spam?

Estimation Notes III

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails, we never see the word '`cost`' (but it does occur in the ham set), and it shows up in our actual email tomorrow.

Q What's the probability that email is spam?

$\rightarrow$ well, $\widehat{\Pr(t_k|c)} = \Pr(\widehat{\text{'cost'}|}\text{spam}) = 0$.

Estimation Notes III

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails, we never see the word '`cost`' (but it does occur in the ham set), and it shows up in our actual email tomorrow.

Q What's the probability that email is spam?

→ well, $\widehat{\Pr(t_k|c)} = \widehat{\Pr(\text{'cost'}|}\text{spam}) = 0$. And that will be multiplied into the product.

Estimation Notes III

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails, we never see the word '`cost`' (but it does occur in the ham set), and it shows up in our actual email tomorrow.

Q What's the probability that email is spam?

$\rightarrow$ well, $\widehat{\Pr(t_k|c)} = \Pr(\text{`}\widehat{\text{cost'}|}\text{spam}) = 0$. And that will be multiplied into the product. So, $\Pr(\text{spam}|d) = 0$.

Estimation Notes III

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails, we never see the word 'cost' (but it does occur in the ham set), and it shows up in our actual email tomorrow.

Q What's the probability that email is spam?

$\rightarrow$ well, $\widehat{\Pr(t_k|c)} = \widehat{\Pr(\text{'cost'}|\text{spam})} = 0$. And that will be multiplied into the product. So, $\Pr(\text{spam}|d) = 0$.

So may want to add one to each count:

Estimation Notes III

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails, we never see the word '`cost`' (but it does occur in the ham set), and it shows up in our actual email tomorrow.

Q What's the probability that email is spam?

→ well, $\widehat{\Pr(t_k|c)} = \Pr(\widehat{\text{'cost'}|}\text{spam}) = 0$. And that will be multiplied into the product. So, $\Pr(\text{spam}|d) = 0$.

So may want to add one to each count: $\frac{T_{ct}+1}{\sum_{t' \in V}(T_{ct'}+1)}$

Estimation Notes III

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails, we never see the word ‘`cost`’ (but it does occur in the ham set), and it shows up in our actual email tomorrow.

Q What's the probability that email is spam?

→ well, $\widehat{\Pr(t_k|c)} = \Pr(\widehat{\text{‘cost’}|}\text{spam}) = 0$. And that will be multiplied into the product. So, $\Pr(\text{spam}|d) = 0$.

So may want to add one to each count: $\frac{T_{ct}+1}{\sum_{t' \in V}(T_{ct'}+1)}$ to avoid wiping out the products (or causing problems for taking logs).

Estimation Notes III

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails, we never see the word ‘`cost`’ (but it does occur in the ham set), and it shows up in our actual email tomorrow.

Q What’s the probability that email is spam?

→ well, $\widehat{\Pr(t_k|c)} = \Pr(\widehat{\text{‘cost’}|}\text{spam}) = 0$. And that will be multiplied into the product. So, $\Pr(\text{spam}|d) = 0$.

So may want to add one to each count: $\frac{T_{ct}+1}{\sum_{t' \in V}(T_{ct'}+1)}$ to avoid wiping out the products (or causing problems for taking logs). Equivalent to adding size of the vocabulary to the counts within the class.

Estimation Notes III

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails, we never see the word '`cost`' (but it does occur in the ham set), and it shows up in our actual email tomorrow.

Q What's the probability that email is spam?

→ well, $\widehat{\Pr(t_k|c)} = \Pr(\text{`}\widehat{\text{cost'}|}\text{spam}) = 0$. And that will be multiplied into the product. So, $\Pr(\text{spam}|d) = 0$.

So may want to add one to each count: $\frac{T_{ct}+1}{\sum_{t' \in V}(T_{ct'}+1)}$ to avoid wiping out the products (or causing problems for taking logs). Equivalent to adding size of the vocabulary to the counts within the class.

→ Laplace smoothing,

Estimation Notes III

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails, we never see the word ‘`cost`’ (but it does occur in the ham set), and it shows up in our actual email tomorrow.

Q What’s the probability that email is spam?

$\rightarrow$ well, $\widehat{\Pr(t_k|c)} = \widehat{\Pr(\text{`cost'}|\text{spam})} = 0$. And that will be multiplied into the product. So, $\Pr(\text{spam}|d) = 0$.

So may want to add one to each count: $\frac{T_{ct}+1}{\sum_{t' \in V}(T_{ct'}+1)}$ to avoid wiping out the products (or causing problems for taking logs). Equivalent to adding size of the vocabulary to the counts within the class.

$\rightarrow$ Laplace smoothing, equivalent to a uniform prior on term (each term occurs once for each class).

Estimation Notes III

Sparsity can be a problem in the training set. Suppose, in our training set of spam emails, we never see the word '`cost`' (but it does occur in the ham set), and it shows up in our actual email tomorrow.

Q What's the probability that email is spam?

→ well, $\widehat{\Pr(t_k|c)} = \widehat{\Pr(\text{'cost'}|\text{spam})} = 0$. And that will be multiplied into the product. So, $\Pr(\text{spam}|d) = 0$.

So may want to add one to each count: $\frac{T_{ct}+1}{\sum_{t' \in V}(T_{ct'}+1)}$ to avoid wiping out the products (or causing problems for taking logs). Equivalent to adding size of the vocabulary to the counts within the class.

→ Laplace smoothing, equivalent to a uniform prior on term (each term occurs once for each class). Use slightly different smoother for Bernoulli case.

Classifier is 'Naive'. . .

Classifier is 'Naive'...

1 we assume conditional independence:

Classifier is 'Naive'...

1. we assume conditional independence: probability that a particular feature occurs is independent of any other feature occurring, once we condition on a given category.

Classifier is 'Naive'...

1 we assume conditional independence: probability that a particular feature occurs is independent of any other feature occurring, once we condition on a given category.

e.g. probability of observing '`money`' is independent of probability of observing '`dollars`' given the emails are spam.

Classifier is 'Naive'...

1 we assume conditional independence: probability that a particular feature occurs is independent of any other feature occurring, once we condition on a given category.

e.g. probability of observing '`money`' is independent of probability of observing '`dollars`' given the emails are spam. This implies $\Pr(\text{money}|c) = \Pr(\text{money}|c, \text{dollars})$,

Classifier is 'Naive'...

1. we assume conditional independence: probability that a particular feature occurs is independent of any other feature occurring, once we condition on a given category.

e.g. probability of observing '`money`' is independent of probability of observing '`dollars`' given the emails are spam. This implies $\Pr(\text{money}|c) = \Pr(\text{money}|c, \text{dollars})$, enables as to write everything as a simple product,

Classifier is 'Naive'...

1 we assume conditional independence: probability that a particular feature occurs is independent of any other feature occurring, once we condition on a given category.

e.g. probability of observing '`money`' is independent of probability of observing '`dollars`' given the emails are spam. This implies $\Pr(\text{money}|c) = \Pr(\text{money}|c, \text{dollars})$, enables as to write everything as a simple product, $\prod_{k=1}^{K} \widehat{\Pr(t_k|c)}$.

Classifier is ‘Naive’...

1 we assume conditional independence: probability that a particular feature occurs is independent of any other feature occurring, once we condition on a given category.

e.g. probability of observing ‘`money`’ is independent of probability of observing ‘`dollars`’ given the emails are spam. This implies $\Pr(\text{money}|c) = \Pr(\text{money}|c, \text{dollars})$, enables as to write everything as a simple product, $\prod_{k=1}^{K} \widehat{\Pr(t_k|c)}$.

2 we assume positional independence:

Classifier is 'Naive'...

1 we assume conditional independence: probability that a particular feature occurs is independent of any other feature occurring, once we condition on a given category.

e.g. probability of observing '`money`' is independent of probability of observing '`dollars`' given the emails are spam. This implies $\Pr(\text{money}|c) = \Pr(\text{money}|c, \text{dollars})$, enables as to write everything as a simple product, $\prod_{k=1}^{K} \widehat{\Pr(t_k|c)}$.

2 we assume positional independence: probability that a term occurs in a particular place is constant for the entire document.

Classifier is 'Naive'...

1. we assume conditional independence: probability that a particular feature occurs is independent of any other feature occurring, once we condition on a given category.

e.g. probability of observing '`money`' is independent of probability of observing '`dollars`' given the emails are spam. This implies $\Pr(\text{money}|c) = \Pr(\text{money}|c, \text{dollars})$, enables as to write everything as a simple product, $\prod_{k=1}^{K} \widehat{\Pr(t_k|c)}$.

2. we assume positional independence: probability that a term occurs in a particular place is constant for the entire document. This implies we only need one probability distribution of terms, and that it's valid for every position.

Classifier is 'Naive'...

1. we assume conditional independence: probability that a particular feature occurs is independent of any other feature occurring, once we condition on a given category.

e.g. probability of observing '`money`' is independent of probability of observing '`dollars`' given the emails are spam. This implies $\Pr(\text{money}|c) = \Pr(\text{money}|c, \text{dollars})$, enables as to write everything as a simple product, $\prod_{k=1}^{K} \widehat{\Pr(t_k|c)}$.

2. we assume positional independence: probability that a term occurs in a particular place is constant for the entire document. This implies we only need one probability distribution of terms, and that it's valid for every position.

e.g. probability of observing '`dear`' is the same regardless of which word in the document we are considering (1st, 2nd, 3rd etc). Equivalent to bag of words. (not an issue for Bernoulli)

Classifier is 'Naive'...

1. we assume conditional independence: probability that a particular feature occurs is independent of any other feature occurring, once we condition on a given category.

e.g. probability of observing '`money`' is independent of probability of observing '`dollars`' given the emails are spam. This implies $\Pr(\text{money}|c) = \Pr(\text{money}|c, \text{dollars})$, enables as to write everything as a simple product, $\prod_{k=1}^{K} \widehat{\Pr(t_k|c)}$.

2. we assume positional independence: probability that a term occurs in a particular place is constant for the entire document. This implies we only need one probability distribution of terms, and that it's valid for every position.

e.g. probability of observing '`dear`' is the same regardless of which word in the document we are considering (1st, 2nd, 3rd etc). Equivalent to bag of words. (not an issue for Bernoulli)

Example: Jihadi Clerics

Example: Jihadi Clerics

Indonesian cleric's support for ISIS increases the security threat

July 20, 2014 10.14pm EDT

Noor Huda Ismail
PhD Candidate in Politics and International Relations , Monash University

Example: Jihadi Clerics

Indonesian cleric's support for ISIS increases the security threat

July 20, 2014 10.14pm EDT

Noor Huda Ismail
PhD Candidate in Politics and International Relations , Monash University

Nielsen (2012) investigates why certain scholars of Islam become Jihadi:

Example: Jihadi Clerics

Indonesian cleric's support for ISIS increases the security threat

July 20, 2014 10.14pm EDT

Noor Huda Ismail
PhD Candidate in Politics and International Relations , Monash University

Nielsen (2012) investigates why certain scholars of Islam become Jihadi: i.e. why they encourage armed struggle (especially against the west)

Example: Jihadi Clerics

Indonesian cleric's support for ISIS increases the security threat

July 20, 2014 10.14pm EDT

Noor Huda Ismail
PhD Candidate in Politics and International Relations , Monash University

Nielsen (2012) investigates why certain scholars of Islam become Jihadi: i.e. why they encourage armed struggle (especially against the west)

Requires that he first classifies scholars as Jihadi and $\neg$ Jihadi:

Example: Jihadi Clerics

Indonesian cleric's support for ISIS increases the security threat

July 20, 2014 10.14pm EDT

Noor Huda Ismail
PhD Candidate in Politics and International Relations , Monash University

Nielsen (2012) investigates why certain scholars of Islam become Jihadi: i.e. why they encourage armed struggle (especially against the west)

Requires that he first classifies scholars as Jihadi and $\neg$ Jihadi: has 27,142 texts from 101 clerics,

Example: Jihadi Clerics

Indonesian cleric's support for ISIS increases the security threat

July 20, 2014 10.14pm EDT

Noor Huda Ismail

PhD Candidate in Politics and International Relations , Monash University

Nielsen (2012) investigates why certain scholars of Islam become Jihadi: i.e. why they encourage armed struggle (especially against the west)

Requires that he first classifies scholars as Jihadi and $\neg$ Jihadi: has 27,142 texts from 101 clerics, and difficult to do by hand.

Example: Jihadi Clerics


Indonesian cleric's support for ISIS increases the security threat

July 20, 2014 10.14pm EDT

Noor Huda Ismail
PhD Candidate in Politics and International Relations , Monash University

Nielsen (2012) investigates why certain scholars of Islam become Jihadi: i.e. why they encourage armed struggle (especially against the west)

Requires that he first classifies scholars as Jihadi and $\neg$ Jihadi: has 27,142 texts from 101 clerics, and difficult to do by hand.

Jihadi Clerics

Jihadi Clerics

Training set:

Jihadi Clerics

Training set: self-identified Jihadi texts (765),

Jihadi Clerics

Training set: self-identified Jihadi texts (765), and sample from Islamic website as $\neg$ Jihadi (1951)

Jihadi Clerics

Training set: self-identified Jihadi texts (765), and sample from Islamic website as $\neg$ Jihadi (1951)

Preprocess: drops terms occurring in less than 10%, or more than 40% of documents,

Jihadi Clerics

Training set: self-identified Jihadi texts (765), and sample from Islamic website as $\neg$ Jihadi (1951)

Preprocess: drops terms occurring in less than 10%, or more than 40% of documents, and uses 'light' stemmer for Arabic

Jihadi Clerics

Training set: self-identified Jihadi texts (765), and sample from Islamic website as $\neg$ Jihadi (1951)

Preprocess: drops terms occurring in less than 10%, or more than 40% of documents, and uses 'light' stemmer for Arabic

Can assign a *Jihad Score* to each document: basically the logged likelihood ratio, $\sum_i \log \frac{\Pr(t_k|\text{Jihad})}{\Pr(t_k|\neg\ \text{Jihad})}$ (note: doesn't know what 'real world' priors are, so drops them here)

Jihadi Clerics

Training set: self-identified Jihadi texts (765), and sample from Islamic website as $\neg$ Jihadi (1951)

Preprocess: drops terms occurring in less than 10%, or more than 40% of documents, and uses 'light' stemmer for Arabic

Can assign a *Jihad Score* to each document: basically the logged likelihood ratio, $\sum_i \log \frac{\Pr(t_k|\text{Jihad})}{\Pr(t_k|\neg \text{ Jihad})}$ (note: doesn't know what 'real world' priors are, so drops them here)

Then for each cleric,

Jihadi Clerics

Training set: self-identified Jihadi texts (765), and sample from Islamic website as $\neg$ Jihadi (1951)

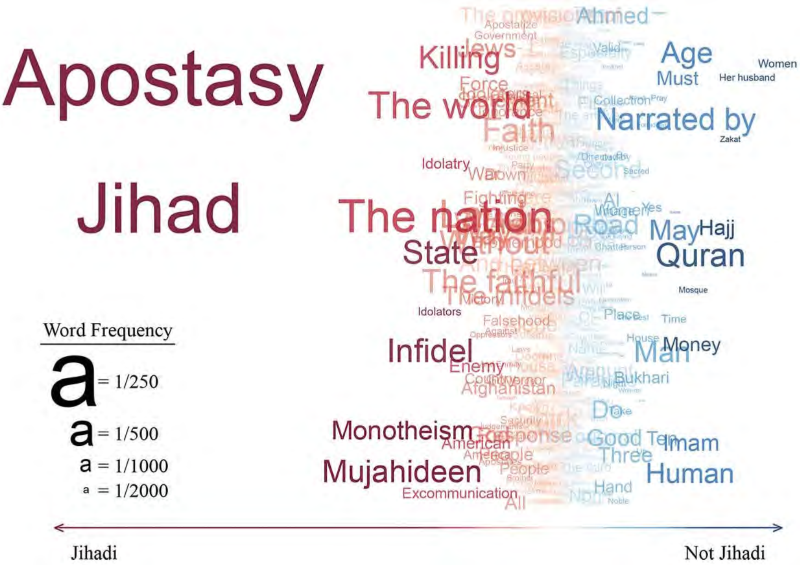
Preprocess: drops terms occurring in less than 10%, or more than 40% of documents, and uses 'light' stemmer for Arabic

Can assign a *Jihad Score* to each document: basically the logged likelihood ratio, $\sum_i \log \frac{\Pr(t_k|\text{Jihad})}{\Pr(t_k|\neg \text{ Jihad})}$ (note: doesn't know what 'real world' priors are, so drops them here)

Then for each cleric, concatenate all works into one and give this 'document'/cleric a score.

Discriminating Words

Discriminating Words



Validation: *Exoneration*

Validation: *Exoneration*

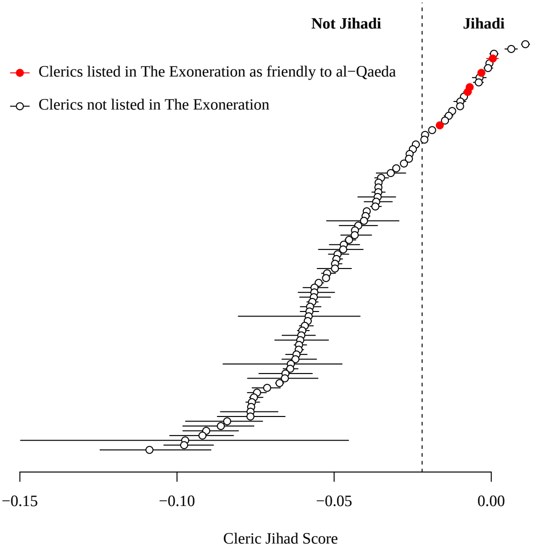


Figure 4.9: *Jihad Scores Predict Inclusion in The Exoneration*

Scoring and Scaling Political Texts

Wordscores (Laver, Benoit & Garry, 2003)

Wordscores (Laver, Benoit & Garry, 2003)

Wordscores (Laver, Benoit & Garry, 2003)

Wordscores (Laver, Benoit & Garry, 2003)

Long standing interest in scaling political texts relative to one another:

Wordscores (Laver, Benoit & Garry, 2003)

Long standing interest in scaling political texts relative to one another:

e.g. are parties moving together over time, such that manifestos are converging?

Wordscores (Laver, Benoit & Garry, 2003)

Long standing interest in scaling political texts relative to one another:

- e.g. are parties moving together over time, such that manifestos are converging?
- e.g. do members of parliament speak in line with their constituency's ideology (roll calls typically uninformative)?

Wordscores (Laver, Benoit & Garry, 2003)

Long standing interest in scaling political texts relative to one another:

- e.g. are parties moving together over time, such that manifestos are converging?
- e.g. do members of parliament speak in line with their constituency's ideology (roll calls typically uninformative)?

→ LBG suggest a way of scoring documents in a NB style, so that we can answer such questions.

Basics

1. Begin with a reference set (training set) of texts that have known positions.

Basics

1. Begin with a reference set (training set) of texts that have known positions.

e.g. we find a 'left' document and give it score -1; and a 'right' document and give it score 1

Basics

1. Begin with a reference set (training set) of texts that have known positions.

e.g. we find a ‘left’ document and give it score -1; and a ‘right’ document and give it score 1

2. Generate word scores from these reference texts

Basics

1 Begin with a reference set (training set) of texts that have known positions.

e.g. we find a 'left' document and give it score -1; and a 'right' document and give it score 1

2 Generate word scores from these reference texts

3 Score the virgin texts (test set) of texts using those word scores, possibly transform virgin scores to original metric.

Scoring the words

Scoring the words

Suppose we have a given reference document R,

Scoring the words

Suppose we have a given reference document R, which is scored as $A_R = 1$. E.g. Neo-Nazi manifesto.

Scoring the words

Suppose we have a given reference document R, which is scored as $A_R = 1$. E.g. Neo-Nazi manifesto.

In document R, count the number of times word i occurs, denote as f_{iR}.

Scoring the words

Suppose we have a given reference document R, which is scored as $A_R = 1$. E.g. Neo-Nazi manifesto.

In document R, count the number of times word i occurs, denote as f_{iR}. Also record the total number of words in document R, and denote as W_R .

Scoring the words

Suppose we have a given reference document R, which is scored as $A_R = 1$. E.g. Neo-Nazi manifesto.

In document R, count the number of times word i occurs, denote as f_{iR}. Also record the total number of words in document R, and denote as W_R .

Do the same for Communist party manifesto L, which we score as $A_L = -1$.

Scoring the words

Suppose we have a given reference document R, which is scored as $A_R = 1$. E.g. Neo-Nazi manifesto.

In document R, count the number of times word i occurs, denote as f_{iR}. Also record the total number of words in document R, and denote as W_R .

Do the same for Communist party manifesto L, which we score as $A_L = -1$. Then calculate f_{iL} and W_L.

Scoring the words

Suppose we have a given reference document R, which is scored as $A_R = 1$. E.g. Neo-Nazi manifesto.

In document R, count the number of times word i occurs, denote as f_{iR}. Also record the total number of words in document R, and denote as W_R .

Do the same for Communist party manifesto L, which we score as $A_L = -1$. Then calculate f_{iL} and W_L.

Define P_{iR} as (approximately the probability of word i given we are in document R),

Scoring the words

Suppose we have a given reference document R, which is scored as $A_R = 1$. E.g. Neo-Nazi manifesto.

In document R, count the number of times word i occurs, denote as f_{iR}. Also record the total number of words in document R, and denote as W_R .

Do the same for Communist party manifesto L, which we score as $A_L = -1$. Then calculate f_{iL} and W_L.

Define P_{iR} as (approximately the probability of word i given we are in document R),

$$P_{iR} = \frac{\frac{f_{iR}}{W_R}}{\frac{f_{iR}}{W_R} + \frac{f_{iL}}{W_L}}$$

Scoring the words

Suppose we have a given reference document R, which is scored as $A_R = 1$. E.g. Neo-Nazi manifesto.

In document R, count the number of times word i occurs, denote as f_{iR}. Also record the total number of words in document R, and denote as W_R .

Do the same for Communist party manifesto L, which we score as $A_L = -1$. Then calculate f_{iL} and W_L.

Define P_{iR} as (approximately the probability of word i given we are in document R),

$$P_{iR} = \frac{\frac{f_{iR}}{W_R}}{\frac{f_{iR}}{W_R} + \frac{f_{iL}}{W_L}}$$

and define P_{iL} in similar way.

Score of a given word i

Score of a given word i

is then

Score of a given word i

is then

$$S_i = A_L P_{iL} + A_R P_{iR},$$

Score of a given word i

is then

$$S_i = A_L P_{iL} + A_R P_{iR},$$

which in our simple case is $S_i = P_{iR} - P_{iL}$.

Score of a given word i

is then

$$S_i = A_L P_{iL} + A_R P_{iR},$$

which in our simple case is $S_i = P_{iR} - P_{iL}$.

and the score of a virgin document is then

$$S_V = \sum_i \frac{f_{iV}}{W_V} \cdot S_i$$

Score of a given word i

is then

$$S_i = A_L P_{iL} + A_R P_{iR},$$

which in our simple case is $S_i = P_{iR} - P_{iL}$.

and the score of a virgin document is then

$$S_V = \sum_i \frac{f_{iV}}{W_V} \cdot S_i$$

NB S_V is the mean of the scores of the words in V weighted by their term frequency.

Score of a given word i

is then

$$S_i = A_L P_{iL} + A_R P_{iR},$$

which in our simple case is $S_i = P_{iR} - P_{iL}$.

and the score of a virgin document is then

$$S_V = \sum_i \frac{f_{iV}}{W_V} \cdot S_i$$

NB S_V is the mean of the scores of the words in V weighted by their term frequency.

NB any new words in the virgin document that were *not* in the reference texts are ignored: the sum is only over the words we've seen in the reference texts.

Example

Example

Neo-Nazi manifesto uses ‘`immigrant`’ 25 times in 1000 words, while Communists use it only 5 times.

Example

Neo-Nazi manifesto uses '`immigrant`' 25 times in 1000 words, while Communists use it only 5 times.

then $P_{iR} = \frac{0.025}{0.025+0.005}$

Example

Neo-Nazi manifesto uses '`immigrant`' 25 times in 1000 words, while Communists use it only 5 times.

then $P_{iR} = \frac{0.025}{0.025+0.005} = 0.83.$

Example

Neo-Nazi manifesto uses ‘`immigrant`’ 25 times in 1000 words, while Communists use it only 5 times.

then $P_{iR} = \frac{0.025}{0.025+0.005} = 0.83.$

and $P_{iL} = \frac{0.005}{0.025+0.005}$

Example

Neo-Nazi manifesto uses '`immigrant`' 25 times in 1000 words, while Communists use it only 5 times.

then $P_{iR} = \frac{0.025}{0.025+0.005} = 0.83.$

and $P_{iL} = \frac{0.005}{0.025+0.005} = 0.16.$

Example

Neo-Nazi manifesto uses '`immigrant`' 25 times in 1000 words, while Communists use it only 5 times.

then $P_{iR} = \frac{0.025}{0.025+0.005} = 0.83.$

and $P_{iL} = \frac{0.005}{0.025+0.005} = 0.16.$

so $S_i = 0.83 - 0.16 = 0.66$

Example

Neo-Nazi manifesto uses '`immigrant`' 25 times in 1000 words, while Communists use it only 5 times.

then $P_{iR} = \frac{0.025}{0.025+0.005} = 0.83.$

and $P_{iL} = \frac{0.005}{0.025+0.005} = 0.16.$

so $S_i = 0.83 - 0.16 = 0.66$

we see a virgin manifesto, from the Conservative party,

Example

Neo-Nazi manifesto uses '`immigrant`' 25 times in 1000 words, while Communists use it only 5 times.

then $P_{iR} = \frac{0.025}{0.025+0.005} = 0.83.$

and $P_{iL} = \frac{0.005}{0.025+0.005} = 0.16.$

so $S_i = 0.83 - 0.16 = 0.66$

we see a virgin manifesto, from the Conservative party, and it mentions `immigrant` 20 times in a thousand words.

Example

Neo-Nazi manifesto uses '`immigrant`' 25 times in 1000 words, while Communists use it only 5 times.

then $P_{iR} = \frac{0.025}{0.025+0.005} = 0.83.$

and $P_{iL} = \frac{0.005}{0.025+0.005} = 0.16.$

so $S_i = 0.83 - 0.16 = 0.66$

we see a virgin manifesto, from the Conservative party, and it mentions `immigrant` 20 times in a thousand words.

well the relevant calculation for that word is $0.02 \times 0.66 = 0.0132$.

Example

Neo-Nazi manifesto uses '`immigrant`' 25 times in 1000 words, while Communists use it only 5 times.

then $P_{iR} = \frac{0.025}{0.025+0.005} = 0.83.$

and $P_{iL} = \frac{0.005}{0.025+0.005} = 0.16.$

so $S_i = 0.83 - 0.16 = 0.66$

we see a virgin manifesto, from the Conservative party, and it mentions `immigrant` 20 times in a thousand words.

well the relevant calculation for that word is $0.02 \times 0.66 = 0.0132$.

but virgin manifesto, from Labour party,

Example

Neo-Nazi manifesto uses ‘`immigrant`’ 25 times in 1000 words, while Communists use it only 5 times.

then $P_{iR} = \frac{0.025}{0.025+0.005} = 0.83.$

and $P_{iL} = \frac{0.005}{0.025+0.005} = 0.16.$

so $S_i = 0.83 - 0.16 = 0.66$

we see a virgin manifesto, from the Conservative party, and it mentions `immigrant` 20 times in a thousand words.

well the relevant calculation for that word is $0.02 \times 0.66 = 0.0132.$

but virgin manifesto, from Labour party, mentions it 10 times in a thousand words: $0.01 \times 0.66 = 0.006$

Example

Neo-Nazi manifesto uses '`immigrant`' 25 times in 1000 words, while Communists use it only 5 times.

then $P_{iR} = \frac{0.025}{0.025+0.005} = 0.83.$

and $P_{iL} = \frac{0.005}{0.025+0.005} = 0.16.$

so $S_i = 0.83 - 0.16 = 0.66$

we see a virgin manifesto, from the Conservative party, and it mentions `immigrant` 20 times in a thousand words.

well the relevant calculation for that word is $0.02 \times 0.66 = 0.0132$.

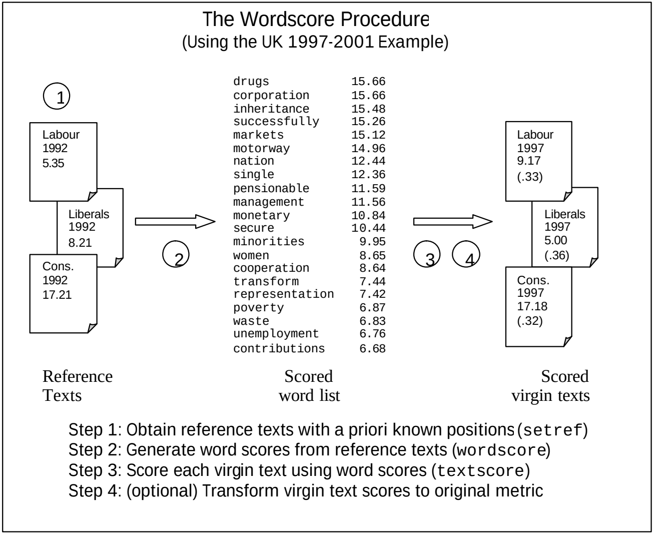
but virgin manifesto, from Labour party, mentions it 10 times in a thousand words: $0.01 \times 0.66 = 0.006$

$\rightarrow$ can rescale these back to original $(-1, 1)$ dimension.

New Labour Moderates its Economic Policy

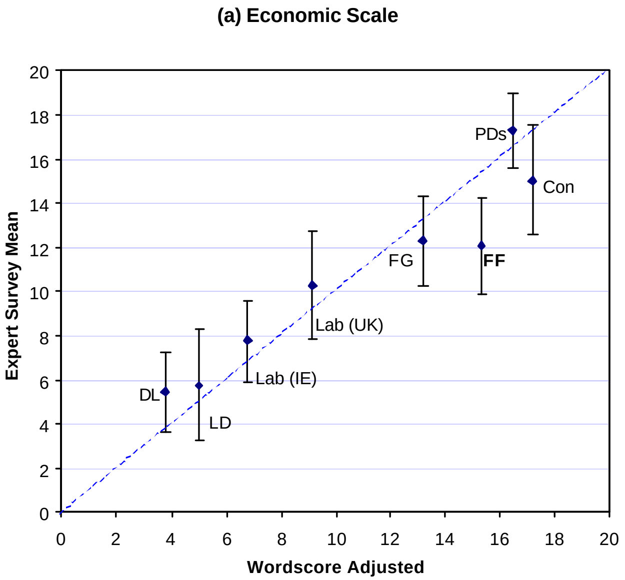
New Labour Moderates its Economic Policy

New Labour Moderates its Economic Policy



Compared to Expert Surveys

Compared to Expert Surveys



Comments

Comments

Extremely influential approach:

Comments

Extremely influential approach: avoids having to pick features of interest

Comments

Extremely influential approach: avoids having to pick features of interest (features that don't distinguish between reference texts have $S_i = 0$)

Comments

Extremely influential approach: avoids having to pick features of interest (features that don't distinguish between reference texts have $S_i = 0$)

and helpful/valid in practice,

Comments

Extremely influential approach: avoids having to pick features of interest (features that don't distinguish between reference texts have $S_i = 0$)

and helpful/valid in practice, and can have uncertainty estimates to boot.

Comments

Extremely influential approach: avoids having to pick features of interest (features that don't distinguish between reference texts have $S_i = 0$)

and helpful/valid in practice, and can have uncertainty estimates to boot.

very important to obtain extreme and appropriate reference,

Comments

Extremely influential approach: avoids having to pick features of interest (features that don't distinguish between reference texts have $S_i = 0$)

and helpful/valid in practice, and can have uncertainty estimates to boot.

very important to obtain extreme and appropriate reference, and score them appropriately.

Comments

Extremely influential approach: avoids having to pick features of interest (features that don't distinguish between reference texts have $S_i = 0$)

and helpful/valid in practice, and can have uncertainty estimates to boot.

very important to obtain extreme and appropriate reference, and score them appropriately. Need to be from domain of virgin texts,

Comments

Extremely influential approach: avoids having to pick features of interest (features that don't distinguish between reference texts have $S_i = 0$)

and helpful/valid in practice, and can have uncertainty estimates to boot.

very important to obtain extreme and appropriate reference, and score them appropriately. Need to be from domain of virgin texts, and have lots of words.

Comments

Extremely influential approach: avoids having to pick features of interest (features that don't distinguish between reference texts have $S_i = 0$)

and helpful/valid in practice, and can have uncertainty estimates to boot.

very important to obtain extreme and appropriate reference, and score them appropriately. Need to be from domain of virgin texts, and have lots of words.

but Lowe unhappy (2008):

Comments

Extremely influential approach: avoids having to pick features of interest (features that don't distinguish between reference texts have $S_i = 0$)

and helpful/valid in practice, and can have uncertainty estimates to boot.

very important to obtain extreme and appropriate reference, and score them appropriately. Need to be from domain of virgin texts, and have lots of words.

but Lowe unhappy (2008): no statistical model,

Comments

Extremely influential approach: avoids having to pick features of interest (features that don't distinguish between reference texts have $S_i = 0$)

and helpful/valid in practice, and can have uncertainty estimates to boot.

very important to obtain extreme and appropriate reference, and score them appropriately. Need to be from domain of virgin texts, and have lots of words.

but Lowe unhappy (2008): no statistical model, inconsistent scoring assumptions,

Comments

Extremely influential approach: avoids having to pick features of interest (features that don't distinguish between reference texts have $S_i = 0$)

and helpful/valid in practice, and can have uncertainty estimates to boot.

very important to obtain extreme and appropriate reference, and score them appropriately. Need to be from domain of virgin texts, and have lots of words.

but Lowe unhappy (2008): no statistical model, inconsistent scoring assumptions, and difficult to pick up 'centrist language' (is equivalent to any language used commonly by all parties for linguistic reasons).

Comments

Extremely influential approach: avoids having to pick features of interest (features that don't distinguish between reference texts have $S_i = 0$)

and helpful/valid in practice, and can have uncertainty estimates to boot.

very important to obtain extreme and appropriate reference, and score them appropriately. Need to be from domain of virgin texts, and have lots of words.

but Lowe unhappy (2008): no statistical model, inconsistent scoring assumptions, and difficult to pick up 'centrist language' (is equivalent to any language used commonly by all parties for linguistic reasons).

while Beauchamp (2011) provides comparison and extension to more purely Bayesian approach.

Classifier Performance and Comparison

Performance of Classifiers

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem.

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem. Suppose we have particular interest in identifying all the Jihadist documents.

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem. Suppose we have particular interest in identifying all the Jihadist documents.

TP the document should be placed in c,

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem. Suppose we have particular interest in identifying all the Jihadist documents.

TP the document should be placed in c, and method placed it in c,

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem. Suppose we have particular interest in identifying all the Jihadist documents.

TP the document should be placed in c, and method placed it in c, we have a true positive.

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem. Suppose we have particular interest in identifying all the Jihadist documents.

TP the document should be placed in c, and method placed it in c, we have a true positive.

FP the document should be placed in $\neg c$,

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem. Suppose we have particular interest in identifying all the Jihadist documents.

TP the document should be placed in c, and method placed it in c, we have a true positive.

FP the document should be placed in $\neg c$, and method placed it in c,

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem. Suppose we have particular interest in identifying all the Jihadist documents.

- TP the document should be placed in c, and method placed it in c, we have a true positive.
- FP the document should be placed in $\neg c$, and method placed it in c, we have a false positive (type I error).

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem. Suppose we have particular interest in identifying all the Jihadist documents.

- TP the document should be placed in c, and method placed it in c, we have a true positive.
- FP the document should be placed in $\neg c$, and method placed it in c, we have a false positive (type I error).
- FN the document should be placed in c,

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem. Suppose we have particular interest in identifying all the Jihadist documents.

- **TP** the document should be placed in c, and method placed it in c, we have a true positive.
- **FP** the document should be placed in $\neg c$, and method placed it in c, we have a false positive (type I error).
- **FN** the document should be placed in c, and method placed it in $\neg c$,

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem. Suppose we have particular interest in identifying all the Jihadist documents.

TP the document should be placed in c, and method placed it in c, we have a true positive.

FP the document should be placed in $\neg c$, and method placed it in c, we have a false positive (type I error).

FN the document should be placed in c, and method placed it in $\neg c$, we have a false negative (type II error).

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem. Suppose we have particular interest in identifying all the Jihadist documents.

- **TP** the document should be placed in c, and method placed it in c, we have a true positive.
- **FP** the document should be placed in $\neg c$, and method placed it in c, we have a false positive (type I error).
- **FN** the document should be placed in c, and method placed it in $\neg c$, we have a false negative (type II error).
- **TN** the document should be placed in $\neg c$,

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem. Suppose we have particular interest in identifying all the Jihadist documents.

- TP the document should be placed in c, and method placed it in c, we have a true positive.
- FP the document should be placed in $\neg c$, and method placed it in c, we have a false positive (type I error).
- FN the document should be placed in c, and method placed it in $\neg c$, we have a false negative (type II error).
- TN the document should be placed in $\neg c$, and method placed it in $\neg c$,

Performance of Classifiers

How do we evaluate whether our classifier (for documents) is any good?

Think about a two class problem. Suppose we have particular interest in identifying all the Jihadist documents.

TP the document should be placed in c, and method placed it in c, we have a true positive.

FP the document should be placed in $\neg c$, and method placed it in c, we have a false positive (type I error).

FN the document should be placed in c, and method placed it in $\neg c$, we have a false negative (type II error).

TN the document should be placed in $\neg c$, and method placed it in $\neg c$, we have a true negative.

Confusion Matrix

Confusion Matrix

		Predicted		
		J	$\neg J$	Total
Actual	J	a TP	b FN	$a+b$
	$\neg J$	c FP	d TN	$c+d$
	Total	$a+c$	$b+d$	N

Confusion Matrix

		Predicted		
		J	$\neg J$	Total
Actual	J	a TP	b FN	$a+b$
	$\neg J$	c FP	d TN	$c+d$
	Total	$a+c$	$b+d$	N

Accuracy : $\frac{\text{number correctly classified}}{\text{total number of cases}} = \frac{a+d}{a+b+c+d}$

Confusion Matrix

		Predicted		
		J	$\neg J$	Total
Actual	J	a TP	b FN	$a+b$
	$\neg J$	c FP	d TN	$c+d$
	Total	$a+c$	$b+d$	N

Accuracy : $\frac{\text{number correctly classified}}{\text{total number of cases}} = \frac{a+d}{a+b+c+d}$

Precision : $\frac{\text{number of TP}}{\text{number of TP}+\text{number of FP}} = \frac{a}{a+c}$.

Confusion Matrix

		Predicted		
		J	$\neg J$	Total
Actual	J	a TP	b FN	$a+b$
	$\neg J$	c FP	d TN	$c+d$
	Total	$a+c$	$b+d$	N

Accuracy : $\frac{\text{number correctly classified}}{\text{total number of cases}} = \frac{a+d}{a+b+c+d}$

Precision : $\frac{\text{number of TP}}{\text{number of TP}+\text{number of FP}} = \frac{a}{a+c}$.
Fraction of the documents predicted to be J, that were in fact J.

Confusion Matrix

		Predicted		
		J	$\neg J$	Total
Actual	J	a TP	b FN	$a+b$
	$\neg J$	c FP	d TN	$c+d$
	Total	$a+c$	$b+d$	N

Accuracy : $\frac{\text{number correctly classified}}{\text{total number of cases}} = \frac{a+d}{a+b+c+d}$

Precision : $\frac{\text{number of TP}}{\text{number of TP}+\text{number of FP}} = \frac{a}{a+c}$.
Fraction of the documents predicted to be J, that were in fact J.

Recall : $\frac{\text{number of TP}}{\text{number of TP} + \text{number of FN}} = \frac{a}{a+b}$.

Confusion Matrix

		Predicted		
		J	$\neg J$	Total
Actual	J	a TP	b FN	$a+b$
	$\neg J$	c FP	d TN	$c+d$
	Total	$a+c$	$b+d$	N

Accuracy : $\frac{\text{number correctly classified}}{\text{total number of cases}} = \frac{a+d}{a+b+c+d}$

Precision : $\frac{\text{number of TP}}{\text{number of TP}+\text{number of FP}} = \frac{a}{a+c}$.
Fraction of the documents predicted to be J, that were in fact J.

Recall : $\frac{\text{number of TP}}{\text{number of TP} + \text{number of FN}} = \frac{a}{a+b}$.
Fraction of the documents that were in fact J, that method predicted were J.

Confusion Matrix

		Predicted		
		J	$\neg J$	Total
Actual	J	a TP	b FN	$a+b$
	$\neg J$	c FP	d TN	$c+d$
	Total	$a+c$	$b+d$	N

Accuracy : $\frac{\text{number correctly classified}}{\text{total number of cases}} = \frac{a+d}{a+b+c+d}$

Precision : $\frac{\text{number of TP}}{\text{number of TP}+\text{number of FP}} = \frac{a}{a+c}$.
Fraction of the documents predicted to be J, that were in fact J.

Recall : $\frac{\text{number of TP}}{\text{number of TP} + \text{number of FN}} = \frac{a}{a+b}$.
Fraction of the documents that were in fact J, that method predicted were J.

F : $2\frac{\text{precision}\cdot\text{recall}}{\text{precision}+\text{recall}}$. Harmonic mean of precision and recall.

Confusion Matrix

		Predicted		
		J	$\neg J$	Total
Actual	J	a TP	b FN	$a+b$
	$\neg J$	c FP	d TN	$c+d$
	Total	$a+c$	$b+d$	N

Accuracy : $\frac{\text{number correctly classified}}{\text{total number of cases}} = \frac{a+d}{a+b+c+d}$

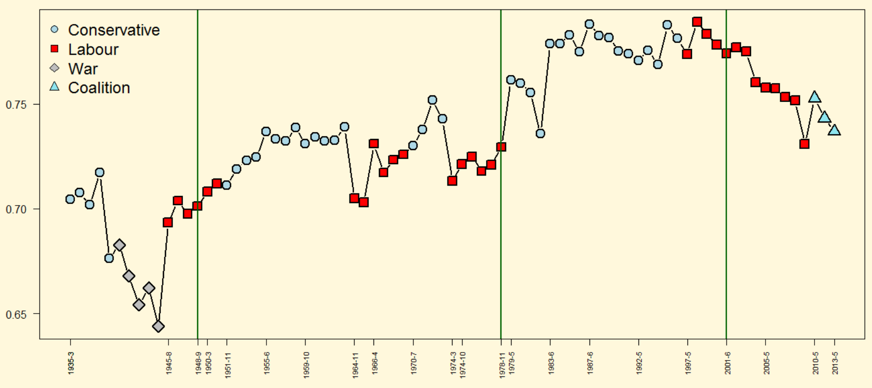
Precision : $\frac{\text{number of TP}}{\text{number of TP}+\text{number of FP}} = \frac{a}{a+c}$.
Fraction of the documents predicted to be J, that were in fact J.

Recall : $\frac{\text{number of TP}}{\text{number of TP} + \text{number of FN}} = \frac{a}{a+b}$.
Fraction of the documents that were in fact J, that method predicted were J.

F : $2\frac{\text{precision}\cdot\text{recall}}{\text{precision}+\text{recall}}$. Harmonic mean of precision and recall.

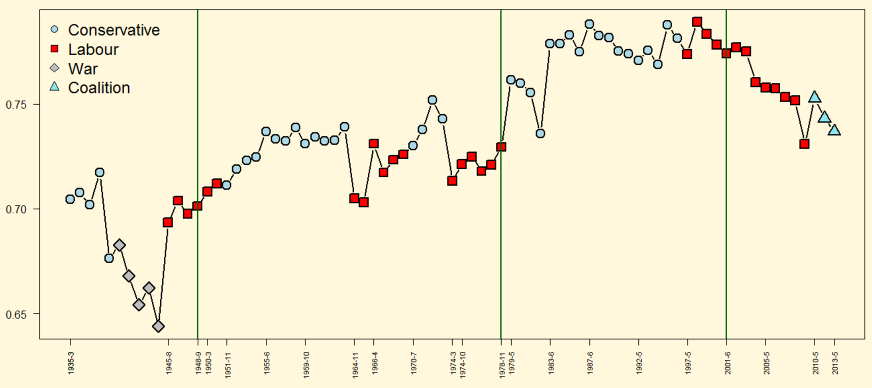
Aside: Sometimes Classifier Performance is Substantively Meaningful

Aside: Sometimes Classifier Performance is Substantively Meaningful



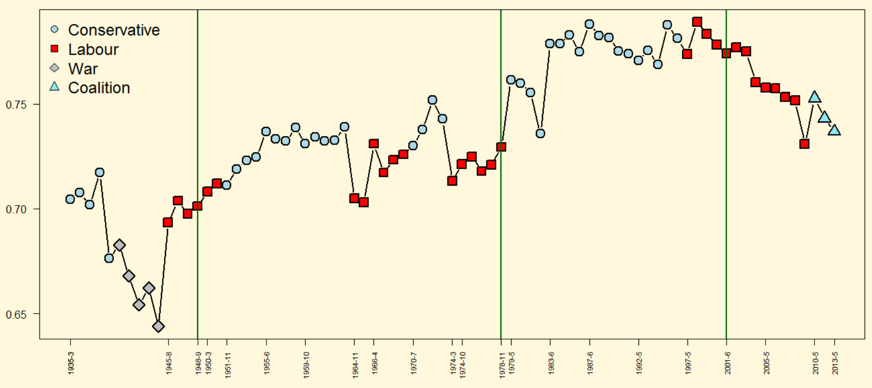
Use machine to classify left (-1) vs right ($+1$) MPs in UK and record classification accuracy.

Aside: Sometimes Classifier Performance is Substantively Meaningful



Use machine to classify left (-1) vs right ($+1$) MPs in UK and record classification accuracy. When high, parties are more polarized.

Aside: Sometimes Classifier Performance is Substantively Meaningful



Use machine to classify left (-1) vs right ($+1$) MPs in UK and record classification accuracy. When high, parties are more polarized.

Makes sense in terms of historical record!

Proportions

Estimating Proportions, Hopkins & King (2010)

Estimating Proportions, Hopkins & King (2010)

Suppose you want to know the proportion of e.g. blog posts or Facebook updates that are sympathetic to Trump.

Estimating Proportions, Hopkins & King (2010)

Suppose you want to know the proportion of e.g. blog posts or Facebook updates that are sympathetic to Trump.

$\rightarrow$ could train a (Naive Bayes) classifier on documents,

Estimating Proportions, Hopkins & King (2010)

Suppose you want to know the proportion of e.g. blog posts or Facebook updates that are sympathetic to Trump.

$\rightarrow$ could train a (Naive Bayes) classifier on documents, and then calculate the proportion of the test set that fits into the class of interest.

Estimating Proportions, Hopkins & King (2010)

Suppose you want to know the proportion of e.g. blog posts or Facebook updates that are sympathetic to Trump.

→ could train a (Naive Bayes) classifier on documents, and then calculate the proportion of the test set that fits into the class of interest.

But problematic when labeled set is not random sample of population

Estimating Proportions, Hopkins & King (2010)

Suppose you want to know the proportion of e.g. blog posts or Facebook updates that are sympathetic to Trump.

→ could train a (Naive Bayes) classifier on documents, and then calculate the proportion of the test set that fits into the class of interest.

But problematic when labeled set is not random sample of population (perhaps due to 'drift'—sample collected once, and population moves on)

Estimating Proportions, Hopkins & King (2010)

Suppose you want to know the proportion of e.g. blog posts or Facebook updates that are sympathetic to Trump.

→ could train a (Naive Bayes) classifier on documents, and then calculate the proportion of the test set that fits into the class of interest.

But problematic when labeled set is not random sample of population (perhaps due to 'drift'—sample collected once, and population moves on)

and DGP is typically $\Pr(t_k|c)$ not $\Pr(c|t_k)$, which is what aggregating would imply (causes some problems for inference, though H&K are v vague here)

Estimating Proportions, Hopkins & King (2010)

Suppose you want to know the proportion of e.g. blog posts or Facebook updates that are sympathetic to Trump.

$\rightarrow$ could train a (Naive Bayes) classifier on documents, and then calculate the proportion of the test set that fits into the class of interest.

But problematic when labeled set is not random sample of population (perhaps due to 'drift'—sample collected once, and population moves on)

and DGP is typically $\Pr(t_k|c)$ not $\Pr(c|t_k)$, which is what aggregating would imply (causes some problems for inference, though H&K are v vague here)

$\rightarrow$ would like unbiased approach (and be nice if non-parametric), that avoids the intermediate step of document classification.

What to do

What to do

Convert all features to K stems S, and count binary instances only

What to do

Convert all features to K stems S, and count binary instances only (stem occurs, or not, in document).

What to do

Convert all features to K stems S, and count binary instances only (stem occurs, or not, in document).Hand code say, 500, texts into the J classes.

What to do

Convert all features to K stems S, and count binary instances only (stem occurs, or not, in document). Hand code say, 500, texts into the J classes.

Notice that this has linear regression form:

$$\overbrace{\underbrace{\Pr(\mathbf{S})}_{2^K \times 1}}^{y} = \overbrace{\underbrace{\Pr(\mathbf{S}|c)}_{2^K \times J}}^{X} \overbrace{\underbrace{\Pr(c)}_{J \times 1}}^{\beta}$$

What to do

Convert all features to K stems S, and count binary instances only (stem occurs, or not, in document).Hand code say, 500, texts into the J classes.

Notice that this has linear regression form:

$$\underbrace{\overbrace{\Pr(\mathbf{S})}^{y}}_{2^K \times 1} = \underbrace{\overbrace{\Pr(\mathbf{S}|c)}^{X}}_{2^K \times J} \underbrace{\overbrace{\Pr(c)}^{\beta}}_{J \times 1}$$

where $\Pr(\mathbf{S})$ is the probability of each of the word stem (binary) combinations occurring, which we can tabulate from the target texts (test set).

What to do

Convert all features to K stems S, and count binary instances only (stem occurs, or not, in document).Hand code say, 500, texts into the J classes.

Notice that this has linear regression form:

$$\underbrace{\overbrace{\Pr(\mathbf{S})}^{y}}_{2^K \times 1} = \underbrace{\overbrace{\Pr(\mathbf{S}|c)}^{X}}_{2^K \times J} \underbrace{\overbrace{\Pr(c)}^{\beta}}_{J \times 1}$$

where $\Pr(\mathbf{S})$ is the probability of each of the word stem (binary) combinations occurring, which we can tabulate from the target texts (test set).

and $\Pr(\mathbf{S}|c)$ is the probability of each of the word stem combinations occurring within class c: which we assume is identical to the same quantity in the labeled set (and then tabulate).

What to do

Convert all features to K stems S, and count binary instances only (stem occurs, or not, in document).Hand code say, 500, texts into the J classes.

Notice that this has linear regression form:

$$\underbrace{\overbrace{\Pr(\mathbf{S})}^{y}}_{2^K \times 1} = \underbrace{\overbrace{\Pr(\mathbf{S}|c)}^{X}}_{2^K \times J} \underbrace{\overbrace{\Pr(c)}^{\beta}}_{J \times 1}$$

where $\Pr(\mathbf{S})$ is the probability of each of the word stem (binary) combinations occurring, which we can tabulate from the target texts (test set).

and $\Pr(\mathbf{S}|c)$ is the probability of each of the word stem combinations occurring within class c: which we assume is identical to the same quantity in the labeled set (and then tabulate).

while $\Pr(c)$ is the proportion of documents in class c,

What to do

Convert all features to K stems S, and count binary instances only (stem occurs, or not, in document).Hand code say, 500, texts into the J classes.

Notice that this has linear regression form:

$$\underbrace{\overbrace{\Pr(\mathbf{S})}^{y}}_{2^K \times 1} = \underbrace{\overbrace{\Pr(\mathbf{S}|c)}^{X}}_{2^K \times J} \underbrace{\overbrace{\Pr(c)}^{\beta}}_{J \times 1}$$

where $\Pr(\mathbf{S})$ is the probability of each of the word stem (binary) combinations occurring, which we can tabulate from the target texts (test set).

and $\Pr(\mathbf{S}|c)$ is the probability of each of the word stem combinations occurring within class c: which we assume is identical to the same quantity in the labeled set (and then tabulate).

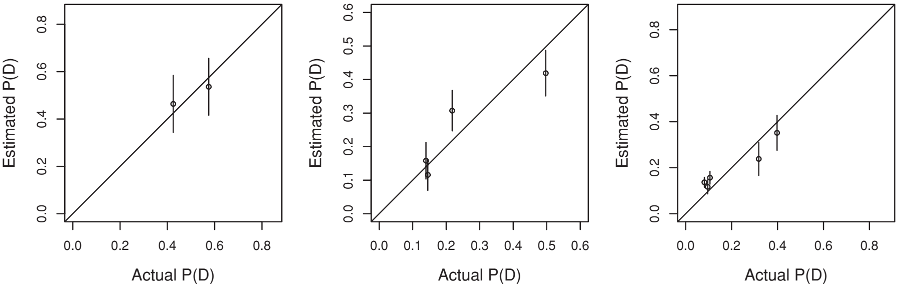
while $\Pr(c)$ is the proportion of documents in class c, which is what we want to know.

Performance: Congress, Editorials, Enron

Performance: Congress, Editorials, Enron

▸ Estimation details

FIGURE 4 Additional Out-of-Sample Validation



Crowdsourcing

Crowdsourcing

Crowdsourcing

So far, the methods have assumed that we already have a training set,

Crowdsourcing

So far, the methods have assumed that we already have a training set, which will typically have been coded by experts.

Crowdsourcing

So far, the methods have assumed that we already have a training set, which will typically have been coded by experts.

but that can be very expensive,

Crowdsourcing

So far, the methods have assumed that we already have a training set, which will typically have been coded by experts.

but that can be very expensive, and it would be good to make it easier to replicate.

Crowdsourcing

So far, the methods have assumed that we already have a training set, which will typically have been coded by experts.

but that can be very expensive, and it would be good to make it easier to replicate.

if we had a large number of 'experts',

Crowdsourcing

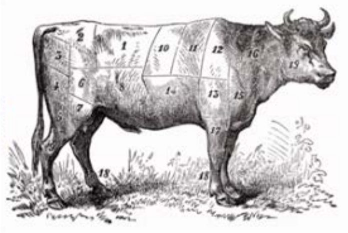
So far, the methods have assumed that we already have a training set, which will typically have been coded by experts.

but that can be very expensive, and it would be good to make it easier to replicate.

if we had a large number of 'experts', we could (depending on the size of the problem) have everything as a 'training' set and avoid modeling at all.

Galton and the Wisdom of Crowds

Galton and the Wisdom of Crowds



average of 800 guesses = 1,197

actual weight of the ox = 1,198

Crowdsourcing as Concept

Crowdsourcing as Concept

Benoit, Conway, Lauderdale, Laver and Mikhaylov (2016)

Crowdsourcing as Concept

Benoit, Conway, Lauderdale, Laver and Mikhaylov (2016) note classification jobs could be given to a large number of relatively cheap online workers.

Crowdsourcing as Concept

Benoit, Conway, Lauderdale, Laver and Mikhaylov (2016) note classification jobs could be given to a large number of relatively cheap online workers.

If those workers make the same judgements ('this document is left wing, this document is right wing') when faced with the same stimuli (on average),

Crowdsourcing as Concept

Benoit, Conway, Lauderdale, Laver and Mikhaylov (2016) note classification jobs could be given to a large number of relatively cheap online workers.

If those workers make the same judgements ('this document is left wing, this document is right wing') when faced with the same stimuli (on average), then the set of them together should obtain the truth (on average) (to the extent that is well-defined!)

Crowdsourcing as Concept

Benoit, Conway, Lauderdale, Laver and Mikhaylov (2016) note classification jobs could be given to a large number of relatively cheap online workers.

If those workers make the same judgements ('this document is left wing, this document is right wing') when faced with the same stimuli (on average), then the set of them together should obtain the truth (on average) (to the extent that is well-defined!)

NB Don't care whether they are 'representative' of some broader population or not:

Crowdsourcing as Concept

Benoit, Conway, Lauderdale, Laver and Mikhaylov (2016) note classification jobs could be given to a large number of relatively cheap online workers.

If those workers make the same judgements ('this document is left wing, this document is right wing') when faced with the same stimuli (on average), then the set of them together should obtain the truth (on average) (to the extent that is well-defined!)

NB Don't care whether they are 'representative' of some broader population or not: this is not a survey to estimate their opinions of the labels—

Crowdsourcing as Concept

Benoit, Conway, Lauderdale, Laver and Mikhaylov (2016) note classification jobs could be given to a large number of relatively cheap online workers.

If those workers make the same judgements ('this document is left wing, this document is right wing') when faced with the same stimuli (on average), then the set of them together should obtain the truth (on average) (to the extent that is well-defined!)

NB Don't care whether they are 'representative' of some broader population or not: this is not a survey to estimate their opinions of the labels—we care about the labels themselves.

Crowdsourcing as Concept

Benoit, Conway, Lauderdale, Laver and Mikhaylov (2016) note classification jobs could be given to a large number of relatively cheap online workers.

If those workers make the same judgements ('this document is left wing, this document is right wing') when faced with the same stimuli (on average), then the set of them together should obtain the truth (on average) (to the extent that is well-defined!)

NB Don't care whether they are 'representative' of some broader population or not: this is not a survey to estimate their opinions of the labels—we care about the labels themselves.

BTW crowdsourcing can certainly be used for such 'survey' tasks—

Crowdsourcing as Concept

Benoit, Conway, Lauderdale, Laver and Mikhaylov (2016) note classification jobs could be given to a large number of relatively cheap online workers.

If those workers make the same judgements ('this document is left wing, this document is right wing') when faced with the same stimuli (on average), then the set of them together should obtain the truth (on average) (to the extent that is well-defined!)

NB Don't care whether they are 'representative' of some broader population or not: this is not a survey to estimate their opinions of the labels—we care about the labels themselves.

BTW crowdsourcing can certainly be used for such 'survey' tasks—see Berinsky et al (2012) for a review of Mechanical Turk for political science use.

Crowdsourcing in practice

Crowdsourcing in practice

BCLLM study data from the Manifesto Project:

Crowdsourcing in practice

BCLLM study data from the Manifesto Project: sentence labels by experts,

Crowdsourcing in practice

BCLLM study data from the Manifesto Project: sentence labels by experts, from over 4000 manifestos.

Crowdsourcing in practice

BCLLM study data from the Manifesto Project: sentence labels by experts, from over 4000 manifestos.

Break up manifestos into random sentences (in context),

Crowdsourcing in practice

BCLLM study data from the Manifesto Project: sentence labels by experts, from over 4000 manifestos.

Break up manifestos into random sentences (in context), and ask CrowdFlower workers to classify into economic or social policy,

Crowdsourcing in practice

BCLLM study data from the Manifesto Project: sentence labels by experts, from over 4000 manifestos.

Break up manifestos into random sentences (in context), and ask CrowdFlower workers to classify into economic or social policy, and then into one of 5 categories (‘very left/liberal’–‘very right/conservative’).

Crowdsourcing in practice

BCLLM study data from the Manifesto Project: sentence labels by experts, from over 4000 manifestos.

Break up manifestos into random sentences (in context), and ask CrowdFlower workers to classify into economic or social policy, and then into one of 5 categories ('very left/liberal'–'very right/conservative').

Model allows for correcting for reader and text fixed effects,

Crowdsourcing in practice

BCLLM study data from the Manifesto Project: sentence labels by experts, from over 4000 manifestos.

Break up manifestos into random sentences (in context), and ask CrowdFlower workers to classify into economic or social policy, and then into one of 5 categories (‘very left/liberal’–‘very right/conservative’).

Model allows for correcting for reader and text fixed effects, though simply taking means works well.

Crowdsourcing in practice

BCLLM study data from the Manifesto Project: sentence labels by experts, from over 4000 manifestos.

Break up manifestos into random sentences (in context), and ask CrowdFlower workers to classify into economic or social policy, and then into one of 5 categories ('very left/liberal'–'very right/conservative').

Model allows for correcting for reader and text fixed effects, though simply taking means works well.

NB can reduce uncertainty around crowd estimates by increasing number of workers for that sentence.

Crowdsourcing in practice

BCLLM study data from the Manifesto Project: sentence labels by experts, from over 4000 manifestos.

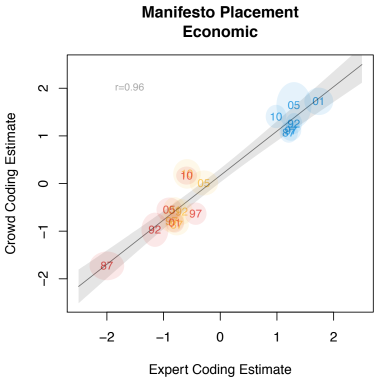
Break up manifestos into random sentences (in context), and ask CrowdFlower workers to classify into economic or social policy, and then into one of 5 categories ('very left/liberal'–'very right/conservative').

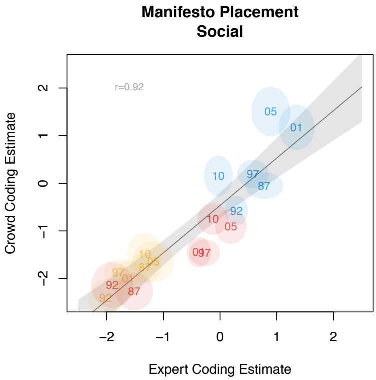
Model allows for correcting for reader and text fixed effects, though simply taking means works well.

NB can reduce uncertainty around crowd estimates by increasing number of workers for that sentence.

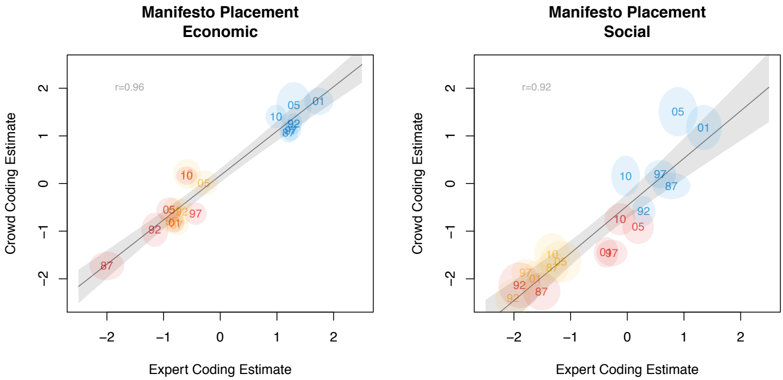
Comparing Experts and CF workers

Comparing Experts and CF workers



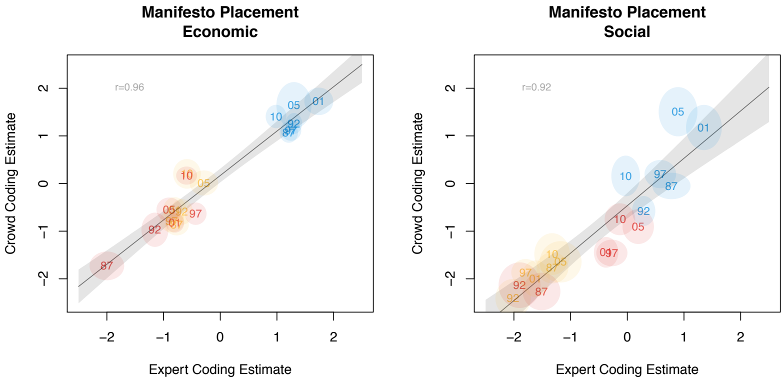


Comparing Experts and CF workers



Note that this method allows replication of the data used in an analysis,

Comparing Experts and CF workers



Note that this method allows replication of the data used in an analysis, not just the analysis itself!