RECSM Summer School:
Social Media and Big Data Research

Pablo Barberá

London School of Economics

`www.pablobarbera.com`

Course website:

pablobarbera.com/social-media-upf

Supervised Machine Learning Applied to Social Media Text

Supervised machine learning

Goal: classify documents into pre existing categories.

e.g. authors of documents, sentiment of tweets, ideological position of parties based on manifestos, tone of movie reviews...

Supervised machine learning

Goal: classify documents into pre existing categories.

e.g. authors of documents, sentiment of tweets, ideological position of parties based on manifestos, tone of movie reviews...

What we need:

- ▶ Hand-coded dataset (labeled), to be split into:

Supervised machine learning

Goal: classify documents into pre existing categories.

e.g. authors of documents, sentiment of tweets, ideological position of parties based on manifestos, tone of movie reviews...

What we need:

- ▶ Hand-coded dataset (labeled), to be split into:
    - ▶ Training set: used to train the classifier

Supervised machine learning

Goal: classify documents into pre existing categories.

e.g. authors of documents, sentiment of tweets, ideological position of parties based on manifestos, tone of movie reviews...

What we need:

- ▶ Hand-coded dataset (labeled), to be split into:
    - ▶ Training set: used to train the classifier
    - ▶ Validation/Test set: used to validate the classifier

Supervised machine learning

Goal: classify documents into pre existing categories.

e.g. authors of documents, sentiment of tweets, ideological position of parties based on manifestos, tone of movie reviews...

What we need:

- ▶ Hand-coded dataset (labeled), to be split into:
  - ▶ Training set: used to train the classifier
  - ▶ Validation/Test set: used to validate the classifier
- ▶ Method to extrapolate from hand coding to unlabeled documents (classifier):

Supervised machine learning

Goal: classify documents into pre existing categories.

e.g. authors of documents, sentiment of tweets, ideological position of parties based on manifestos, tone of movie reviews...

What we need:

- Hand-coded dataset (labeled), to be split into:
    - Training set: used to train the classifier
    - Validation/Test set: used to validate the classifier
- Method to extrapolate from hand coding to unlabeled documents (classifier):
    - Naive Bayes, regularized regression, SVM, K-nearest neighbors, BART, ensemble methods...

Supervised machine learning

Goal: classify documents into pre existing categories.

e.g. authors of documents, sentiment of tweets, ideological position of parties based on manifestos, tone of movie reviews...

What we need:

- ▶ Hand-coded dataset (labeled), to be split into:
    - ▶ Training set: used to train the classifier
    - ▶ Validation/Test set: used to validate the classifier
- ▶ Method to extrapolate from hand coding to unlabeled documents (classifier):
    - ▶ Naive Bayes, regularized regression, SVM, K-nearest neighbors, BART, ensemble methods...
- ▶ Approach to validate classifier: cross-validation

Supervised machine learning

Goal: classify documents into pre existing categories.

e.g. authors of documents, sentiment of tweets, ideological position of parties based on manifestos, tone of movie reviews...

What we need:

- ▶ Hand-coded dataset (labeled), to be split into:
    - ▶ Training set: used to train the classifier
    - ▶ Validation/Test set: used to validate the classifier
- ▶ Method to extrapolate from hand coding to unlabeled documents (classifier):
    - ▶ Naive Bayes, regularized regression, SVM, K-nearest neighbors, BART, ensemble methods...
- ▶ Approach to validate classifier: cross-validation
- ▶ Performance metric to choose best classifier and avoid overfitting: confusion matrix, accuracy, precision, recall...

Supervised v. unsupervised methods compared

- ▶ The goal (in text analysis) is to differentiate *documents* from one another, treating them as "bags of words"

Supervised v. unsupervised methods compared

- ▶ The goal (in text analysis) is to differentiate *documents* from one another, treating them as "bags of words"
- ▶ Different approaches:

Supervised v. unsupervised methods compared

- The goal (in text analysis) is to differentiate *documents* from one another, treating them as "bags of words"
- Different approaches:
    - *Supervised methods* require a training set that exemplify contrasting classes, identified by the researcher

Supervised v. unsupervised methods compared

- ▶ The goal (in text analysis) is to differentiate *documents* from one another, treating them as "bags of words"
- ▶ Different approaches:
    - ▶ *Supervised methods* require a training set that exemplify contrasting classes, identified by the researcher
    - ▶ *Unsupervised methods* scale documents based on patterns of similarity from the term-document matrix, without requiring a training step

Supervised v. unsupervised methods compared

- The goal (in text analysis) is to differentiate *documents* from one another, treating them as "bags of words"
- Different approaches:
  - *Supervised methods* require a training set that exemplify contrasting classes, identified by the researcher
  - *Unsupervised methods* scale documents based on patterns of similarity from the term-document matrix, without requiring a training step
- Relative advantage of supervised methods:
  You already know the dimension being scaled, because you set it in the training stage

Supervised v. unsupervised methods compared

- ▶ The goal (in text analysis) is to differentiate *documents* from one another, treating them as "bags of words"
- ▶ Different approaches:
  - ▶ *Supervised methods* require a training set that exemplify contrasting classes, identified by the researcher
  - ▶ *Unsupervised methods* scale documents based on patterns of similarity from the term-document matrix, without requiring a training step
- ▶ Relative advantage of supervised methods:
  You already know the dimension being scaled, because you set it in the training stage
- ▶ Relative disadvantage of supervised methods:
  You *must* already know the dimension being scaled, because you have to feed it good sample documents in the training stage

Supervised learning v. dictionary methods

- ▶ Dictionary methods:

Supervised learning v. dictionary methods

- Dictionary methods:
  - Advantage: not corpus-specific, cost to apply to a new corpus is trivial

Supervised learning v. dictionary methods

- Dictionary methods:
  - Advantage: not corpus-specific, cost to apply to a new corpus is trivial
  - Disadvantage: not corpus-specific, so performance on a new corpus is unknown (domain shift)

Supervised learning v. dictionary methods

- Dictionary methods:
    - Advantage: not corpus-specific, cost to apply to a new corpus is trivial
    - Disadvantage: not corpus-specific, so performance on a new corpus is unknown (domain shift)
- Supervised learning can be conceptualized as a generalization of dictionary methods, where features associated with each categories (and their relative weight) are learned from the data

Supervised learning v. dictionary methods

- Dictionary methods:
  - Advantage: not corpus-specific, cost to apply to a new corpus is trivial
  - Disadvantage: not corpus-specific, so performance on a new corpus is unknown (domain shift)
- Supervised learning can be conceptualized as a generalization of dictionary methods, where features associated with each categories (and their relative weight) are learned from the data
- By construction, they will outperform dictionary methods in classification tasks, as long as training sample is large enough

Dictionaries vs supervised learning

Lexicons' Accuracy in Document Classification
Compared to Machine-Learning Approach
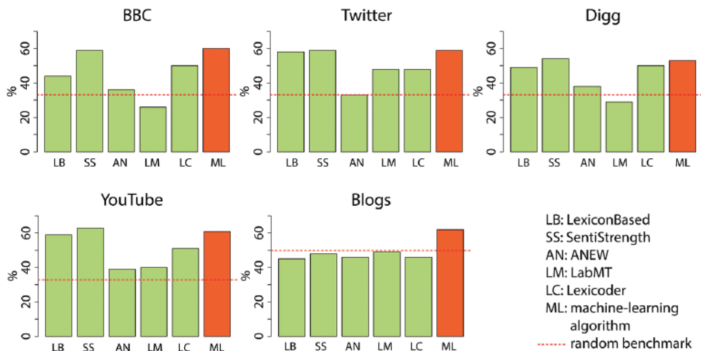
Source: González-Bailón and Paltoglou (2015)

Creating a labeled set

How do we obtain a **labeled set**?

- ► External sources of annotation

Creating a labeled set

How do we obtain a **labeled set**?

- ▶ External sources of annotation
    - ▶ Self-reported ideology in users' profiles

Creating a labeled set

How do we obtain a **labeled set**?

- ▶ External sources of annotation
    - ▶ Self-reported ideology in users' profiles
    - ▶ Gender in social security records

Creating a labeled set

How do we obtain a **labeled set**?

- ▶ External sources of annotation
    - ▶ Self-reported ideology in users' profiles
    - ▶ Gender in social security records
- ▶ Expert annotation

Creating a labeled set

How do we obtain a **labeled set**?

- ▶ External sources of annotation
    - ▶ Self-reported ideology in users' profiles
    - ▶ Gender in social security records

- ▶ Expert annotation
    - ▶ "Canonical" dataset: Comparative Manifesto Project

Creating a labeled set

How do we obtain a **labeled set**?

- ▶ External sources of annotation
    - ▶ Self-reported ideology in users' profiles
    - ▶ Gender in social security records

- ▶ Expert annotation
    - ▶ "Canonical" dataset: Comparative Manifesto Project
    - ▶ In most projects, undergraduate students (expertise comes from training)

Creating a labeled set

How do we obtain a **labeled set**?

- ► External sources of annotation
    - ► Self-reported ideology in users' profiles
    - ► Gender in social security records

- ► Expert annotation
    - ► "Canonical" dataset: Comparative Manifesto Project
    - ► In most projects, undergraduate students (expertise comes from training)

- ► Crowd-sourced coding

Creating a labeled set

How do we obtain a **labeled set**?

- ► External sources of annotation
    - ► Self-reported ideology in users' profiles
    - ► Gender in social security records

- ► Expert annotation
    - ► "Canonical" dataset: Comparative Manifesto Project
    - ► In most projects, undergraduate students (expertise comes from training)

- ► Crowd-sourced coding
    - ► **Wisdom of crowds**: aggregated judgments of non-experts converge to judgments of experts at much lower cost (Benoit et al, 2016)

Creating a labeled set

How do we obtain a **labeled set**?

- ► External sources of annotation
    - ► Self-reported ideology in users' profiles
    - ► Gender in social security records

- ► Expert annotation
    - ► "Canonical" dataset: Comparative Manifesto Project
    - ► In most projects, undergraduate students (expertise comes from training)

- ► Crowd-sourced coding
    - ► **Wisdom of crowds**: aggregated judgments of non-experts converge to judgments of experts at much lower cost (Benoit et al, 2016)
    - ► Easy to implement with CrowdFlower or MTurk

Code the Content of a Sample of Tweets

In this job, you will be presented with tweets about the recent protests related to race and law enforcement in the U.S.

You will have to read the tweet and answer a set of questions about its content.

Read the tweet below paying close attention to detail:

Tweet ID: **447**

El Cid
@JohnGalt2112

🐦 Follow

#BlackLivesMatter don't matter unless they are taken by a white cop.
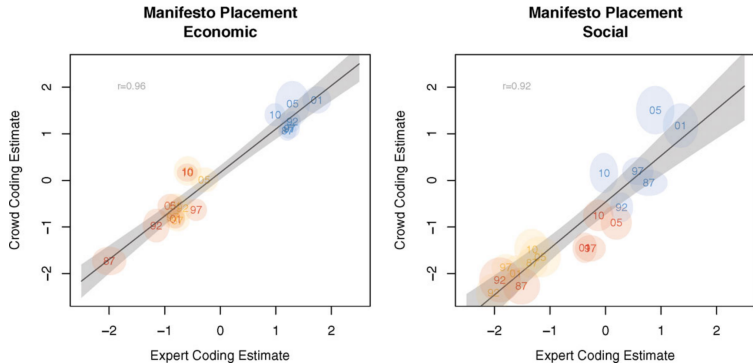
4:23 PM - 13 Dec 2014

↩ ⇄ ★

Is this tweet related to the ongoing debate about law enforcement and race in the United States?
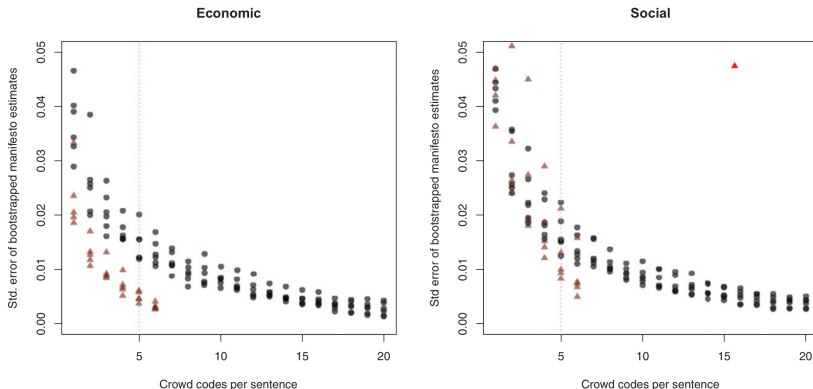
○ Yes
○ No
○ Don't Know

Crowd-sourced text analysis (Benoit et al, 2016 APSR)

FIGURE 3.   Expert and Crowd-sourced Estimates of Economic and Social Policy Positions

Crowd-sourced text analysis (Benoit et al, 2016 APSR)

FIGURE 5.   Standard Errors of Manifesto-level Policy Estimates as a Function of the Number of Workers, for the Oversampled 1987 and 1997 Manifestos

Note: Each point is the bootstrapped standard deviation of the mean of means aggregate manifesto scores, computed from sentence-level random n subsamples from the codes.

Performance metrics

Confusion matrix:

	Actual label	
Classification (algorithm)	Negative	Positive
Negative	True negative	False negative
Positive	False positive	True positive

Performance metrics

	Actual label	
Classification (algorithm)	Negative	Positive
Negative	True negative	False negative
Positive	False positive	True positive

$$\text{Accuracy} = \frac{\text{TrueNeg} + \text{TruePos}}{\text{TrueNeg} + \text{TruePos} + \text{FalseNeg} + \text{FalsePos}}$$

$$\text{Precision}_{\text{positive}} = \frac{\text{TruePos}}{\text{TruePos} + \text{FalsePos}}$$

$$\text{Recall}_{\text{positive}} = \frac{\text{TruePos}}{\text{TruePos} + \text{FalseNeg}}$$

Performance metrics: an example

Confusion matrix:

Classification (algorithm)	Actual label	
	Negative	Positive
Negative	800	100
Positive	50	50

Performance metrics: an example

Confusion matrix:

	Actual label	
Classification (algorithm)	Negative	Positive
Negative	800	100
Positive	50	50

$$\text{Accuracy} = \frac{800 + 50}{700 + 50 + 100 + 50} = 0.85$$

$$\text{Precision}_{\text{positive}} = \frac{50}{50 + 50} = 0.50$$

$$\text{Recall}_{\text{positive}} = \frac{50}{50 + 100} = 0.33$$

Measuring performance

- Classifier is trained to maximize in-sample performance

Measuring performance

- Classifier is trained to maximize in-sample performance
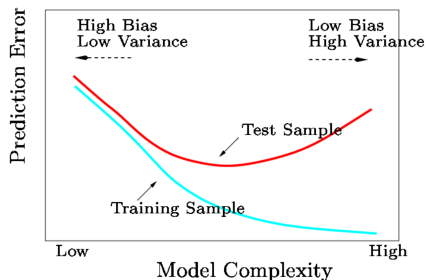- But generally we want to apply method to new data

Measuring performance

- Classifier is trained to maximize in-sample performance
- But generally we want to apply method to new data
- Danger: overfitting

Measuring performance

- Classifier is trained to maximize in-sample performance
- But generally we want to apply method to new data
- Danger: overfitting

Measuring performance

- Classifier is trained to maximize in-sample performance
- But generally we want to apply method to new data
- Danger: overfitting

Measuring performance

- Classifier is trained to maximize in-sample performance
- But generally we want to apply method to new data
- Danger: overfitting

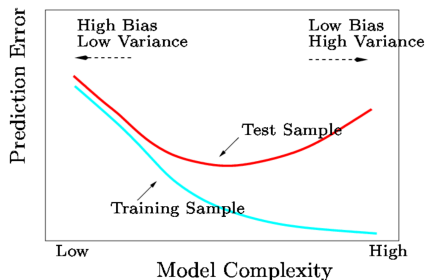# Measuring performance

- Classifier is trained to maximize in-sample performance
- But generally we want to apply method to new data
- Danger: overfitting

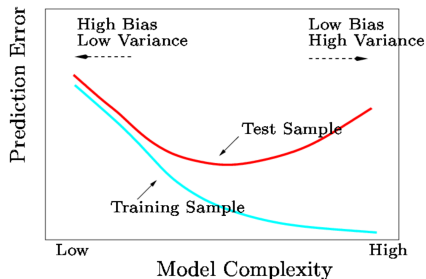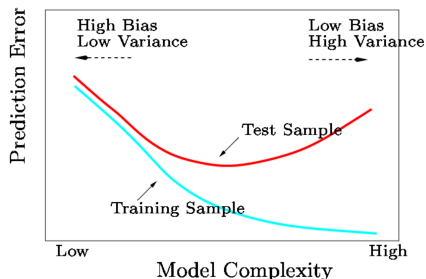- Model is too complex, describes noise rather than signal (Bias-Variance trade-off)

Measuring performance

- Classifier is trained to maximize in-sample performance
- But generally we want to apply method to new data
- Danger: overfitting

- Model is too complex, describes noise rather than signal (Bias-Variance trade-off)
- Focus on features that perform well in labeled data but may not generalize (e.g. unpopular hashtags)

Measuring performance

- Classifier is trained to maximize in-sample performance
- But generally we want to apply method to new data
- Danger: overfitting

- Model is too complex, describes noise rather than signal (Bias-Variance trade-off)

- Focus on features that perform well in labeled data but may not generalize (e.g. unpopular hashtags)

- In-sample performance better than out-of-sample performance

Measuring performance
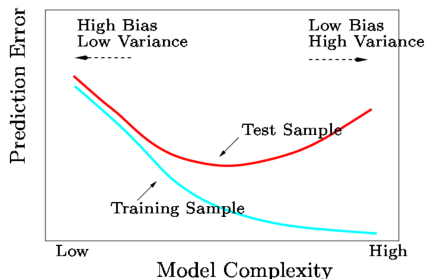
- Classifier is trained to maximize in-sample performance
- But generally we want to apply method to new data
- Danger: overfitting

- Model is too complex, describes noise rather than signal (Bias-Variance trade-off)

- Focus on features that perform well in labeled data but may not generalize (e.g. unpopular hashtags)

- In-sample performance better than out-of-sample performance

- Solutions?

Measuring performance

- Classifier is trained to maximize in-sample performance
- But generally we want to apply method to new data
- Danger: overfitting

- Model is too complex, describes noise rather than signal (Bias-Variance trade-off)
- Focus on features that perform well in labeled data but may not generalize (e.g. unpopular hashtags)
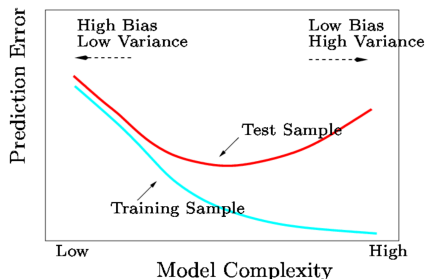- In-sample performance better than out-of-sample performance

- Solutions?
  - Randomly split dataset into training and test set

Measuring performance

- Classifier is trained to maximize in-sample performance
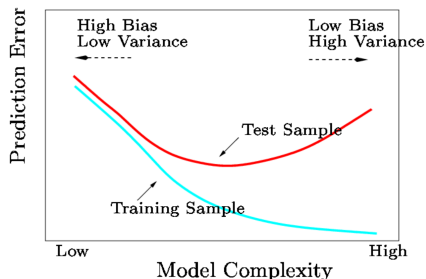- But generally we want to apply method to new data
- Danger: overfitting

- Model is too complex, describes noise rather than signal (Bias-Variance trade-off)
- Focus on features that perform well in labeled data but may not generalize (e.g. unpopular hashtags)
- In-sample performance better than out-of-sample performance

- Solutions?
  - Randomly split dataset into training and test set
  - Cross-validation

Cross-validation

Intuition:

► Create K training and test sets ("folds") within training set.

► For each k in K, run classifier and estimate performance in test set within fold.

► Choose best classifier based on cross-validated performance

Example: Theocharis et al (2016 JOC)

Why do politicians not take full advantage of interactive affordances of social media?

A politician's incentive structure

Democracy → Dialogue > Mobilisation > Marketing

Politician → Marketing > Mobilisation > Dialogue*

H1: Politicians make broadcasting rather than engaging use of Twitter

H2: Engaging style of tweeting is positively related to impolite or uncivil responses

Data collection and case selection

Data: European Election Study 2014, Social Media Study

- ▶ List of all candidates with Twitter accounts in 28 EU countries
  - ▶ 2,482 out of 15,527 identified MEP candidates (16%)
- ▶ Collaboration with TNS Opinion to collect all tweets by candidates *and* tweets mentioning candidates (tweets, retweets, @-replies), May 5th to June 1st 2014.

Data collection and case selection

Data: European Election Study 2014, Social Media Study

- List of all candidates with Twitter accounts in 28 EU countries
  - 2,482 out of 15,527 identified MEP candidates (16%)
- Collaboration with TNS Opinion to collect all tweets by candidates *and* tweets mentioning candidates (tweets, retweets, @-replies), May 5th to June 1st 2014.

Case selection: expected variation in politeness/civility

	Received bailout	Did not receive bailout
High support for EU	Spain (55.4%)	Germany (68.5%)
Low support for EU	Greece (43.8%)	UK (41.4%)

(% indicate proportion of country that considers the EU to be "a good thing")

Data collection and case selection

Data coverage by country

Country	Lists	Candidates	on Twitter	Tweets
Germany	9	501	123 (25%)	86,777
Greece	9	359	99 (28%)	18,709
Spain	11	648	221 (34%)	463,937
UK	28	733	304 (41%)	273,886

Coding tweets

Coded data: random sample of ~7,000 tweets from each country, labeled by undergraduate students:

1. **Politeness**

Coding tweets

Coded data: random sample of ∼7,000 tweets from each country, labeled by undergraduate students:

1. **Politeness**
   - ▶ Polite: tweet adheres to politeness standards.

Coding tweets

Coded data: random sample of ~7,000 tweets from each country, labeled by undergraduate students:

1. **Politeness**
   - Polite: tweet adheres to politeness standards.
   - Impolite: ill-mannered, disrespectful, offensive language...

Coding tweets

Coded data: random sample of ~7,000 tweets from each country, labeled by undergraduate students:

1. **Politeness**
   - Polite: tweet adheres to politeness standards.
   - Impolite: ill-mannered, disrespectful, offensive language...
2. **Communication style**

Coding tweets

Coded data: random sample of ∼7,000 tweets from each country, labeled by undergraduate students:

1. **Politeness**
   - Polite: tweet adheres to politeness standards.
   - Impolite: ill-mannered, disrespectful, offensive language...

2. **Communication style**
   - Broadcasting: statement, expression of opinion

Coding tweets

Coded data: random sample of ∼7,000 tweets from each country, labeled by undergraduate students:

1. **Politeness**
   - Polite: tweet adheres to politeness standards.
   - Impolite: ill-mannered, disrespectful, offensive language...

2. **Communication style**
   - Broadcasting: statement, expression of opinion
   - Engaging: directed to someone else/another user

Coding tweets

Coded data: random sample of $\sim$7,000 tweets from each country, labeled by undergraduate students:

1. **Politeness**
   - Polite: tweet adheres to politeness standards.
   - Impolite: ill-mannered, disrespectful, offensive language...

2. **Communication style**
   - Broadcasting: statement, expression of opinion
   - Engaging: directed to someone else/another user

3. **Political content: moral and democracy**

Coding tweets

Coded data: random sample of ~7,000 tweets from each country, labeled by undergraduate students:

1. **Politeness**
   - Polite: tweet adheres to politeness standards.
   - Impolite: ill-mannered, disrespectful, offensive language...

2. **Communication style**
   - Broadcasting: statement, expression of opinion
   - Engaging: directed to someone else/another user

3. **Political content: moral and democracy**
   - Tweets make reference to: freedom and human rights, traditional morality, law and order, social harmony, democracy...

Coding tweets

Coded data: random sample of ∼7,000 tweets from each country, labeled by undergraduate students:

1. **Politeness**
   - Polite: tweet adheres to politeness standards.
   - Impolite: ill-mannered, disrespectful, offensive language...

2. **Communication style**
   - Broadcasting: statement, expression of opinion
   - Engaging: directed to someone else/another user

3. **Political content: moral and democracy**
   - Tweets make reference to: freedom and human rights, traditional morality, law and order, social harmony, democracy...

Incivility = impoliteness + moral and democracy

Coding tweets

Coding process: summary statistics

	Germany	Greece	Spain	UK
Coded by 1/by 2	2947/2819	2787/2955	3490/1952	3189/3296
Total coded	5766	5742	5442	6485
Impolite	399	1050	121	328
Polite	5367	4692	5321	6157
% Agreement	92	80	93	95
Krippendorf/Maxwell	0.30/0.85	0.26/0.60	0.17/0.87	0.54/0.90
Broadcasting	2755	2883	1771	1557
Engaging	3011	2859	3671	4928
% Agreement	79	85	84	85
Krippendorf/Maxwell	0.58/0.59	0.70/0.70	0.66/0.69	0.62/0.70
Moral/Dem.	265	204	437	531
Other	5501	5538	5005	5954
% Agreement	95	97	96	90
Krippendorf/Maxwell	0.50/0.91	0.53/0.93	0.41/0.92	0.39/0.81

Machine learning classification of tweets

Coded tweets as training dataset for a machine learning classifier:

1. Text preprocessing: lowercase, remove stopwords and punctuation (except # and @), transliterating to ASCII, stem, tokenize into unigrams and bigrams. Keep tokens in 2+ tweets but <90%.

Machine learning classification of tweets

Coded tweets as training dataset for a machine learning classifier:

1. **Text preprocessing**: lowercase, remove stopwords and punctuation (except # and @), transliterating to ASCII, stem, tokenize into unigrams and bigrams. Keep tokens in 2+ tweets but <90%.

2. **Train classifier:** logistic regression with L2 regularization (ridge regression), one per language and variable

Machine learning classification of tweets

Coded tweets as training dataset for a machine learning classifier:

1. **Text preprocessing**: lowercase, remove stopwords and punctuation (except # and @), transliterating to ASCII, stem, tokenize into unigrams and bigrams. Keep tokens in 2+ tweets but <90%.

2. **Train classifier:** logistic regression with L2 regularization (ridge regression), one per language and variable

3. **Evaluate classifier:** compute accuracy using 5-fold crossvalidation

Machine learning classification of tweets

		UK	Spain	Greece	Germany
Communication Style	Accuracy	0.821	0.775	0.863	0.806
	Precision	0.837	0.795	0.838	0.818
	Recall	0.946	0.890	0.894	0.832
Polite vs. impolite	Accuracy	0.954	0.976	0.821	0.935
	Precision	0.955	0.977	0.849	0.938
	Recall	0.998	1.000	0.953	0.997
Morality and Democracy	Accuracy	0.895	0.913	0.957	0.922
	Precision	0.734	0.665	0.851	0.770
	Recall	0.206	0.166	0.080	0.061

Top predictive n-grams

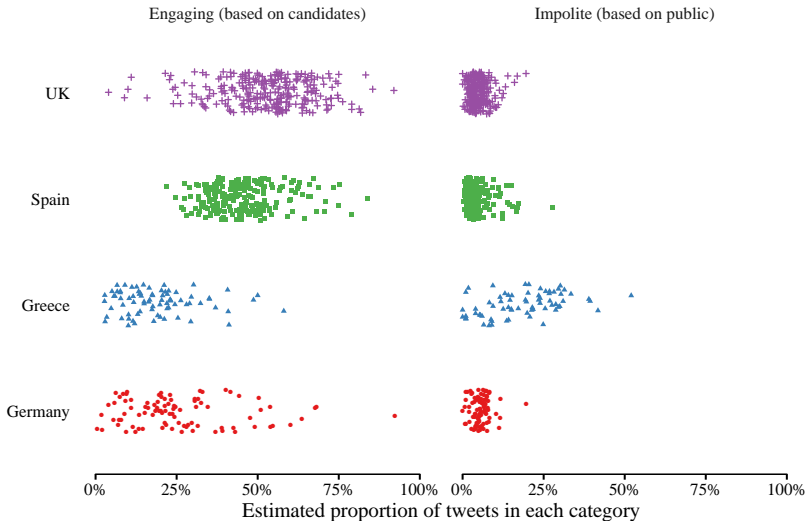|	
Broadcasting	just, hack, #votegreen2014, :, and, @ ', tonight, candid, up, tonbridg, vote @, im @, follow ukip, ukip @, #telleurop, angri, #ep2014, password, stori, #vote2014, team, #labourdoorstep, crimin, bbc news
Engaging	@ thank, @ ye, you'r, @ it', @ mani, @ pleas, u, @ hi, @ congratul, :), index, vote # skip, @ good, fear, cheer, haven't, lol, @ i'v, you'v, @ that', choice, @ wa, @ who, @ hope
Impolite	cunt, fuck, twat, stupid, shit, dick, tit, wanker, scumbag, moron, cock, foot, racist, fascist, sicken, fart, @ fuck, ars, suck, nigga, nigga ?, smug, idiot, @arsehol, arsehol
Polite	@ thank, eu, #ep2014, thank, know, candid, veri, politician, today, way, differ, europ, democraci, interview, time, tonight, @ think, news, european, sorri, congratul, good, :, democrat, seat
Moral/Dem.	democraci, polic, freedom, media, racist, gay, peac, fraud, discrimin, homosexu, muslim, equal, right, crime, law, violenc, constitut, faith, bbc, christian, marriag, god, cp, racism, sexist
Others	@ ha, 2, snp, nice, tell, eu, congratul, campaign, leav, alreadi, wonder, vote @, ;), hust, nh, brit, tori, deliv, bad, immigr, #ukip, live, count, got, roma

Predictive validity

Citizens are more likely to respond to candidates when they adopt an engaging style
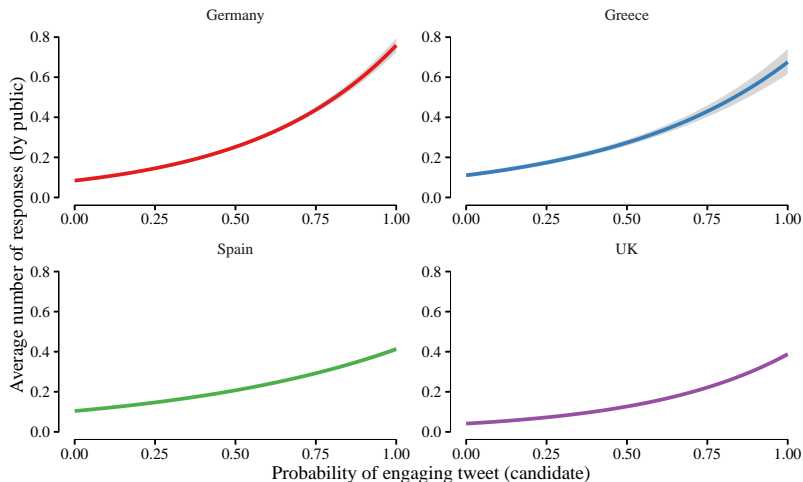
Results: H1

Proportion of engaging tweets sent and impolite tweets received, by candidate and country

Engaging (based on candidates)   Impolite (based on public)

Estimated proportion of tweets in each category

Is engaging style positively related to impolite responses?

Three levels of analysis:

Is engaging style positively related to impolite responses?

Three levels of analysis:

1. **Across candidates**: candidates who send more engaging tweets receive more impolite responses.

Is engaging style positively related to impolite responses?

Three levels of analysis:

1. **Across candidates**: candidates who send more engaging tweets receive more impolite responses.
2. **Within candidates, over time**: the number of impolite responses increases during the campaign for candidates who send more engaging tweets

Is engaging style positively related to impolite responses?

Three levels of analysis:

1. **Across candidates**: candidates who send more engaging tweets receive more impolite responses.

2. **Within candidates, over time**: the number of impolite responses increases during the campaign for candidates who send more engaging tweets

3. **Across tweets**: tweets that are classified as engaging tend to receive more impolite responses

Types of classifiers

General thoughts:
- Trade-off between accuracy and interpretability

Types of classifiers

General thoughts:

- ▶ Trade-off between accuracy and interpretability
- ▶ Parameters need to be cross-validated

Types of classifiers

General thoughts:

- ► Trade-off between accuracy and interpretability
- ► Parameters need to be cross-validated

Frequently used classifiers:

Types of classifiers

General thoughts:

- Trade-off between accuracy and interpretability
- Parameters need to be cross-validated

Frequently used classifiers:

- Naive Bayes

Types of classifiers

General thoughts:

- ▶ Trade-off between accuracy and interpretability
- ▶ Parameters need to be cross-validated

Frequently used classifiers:

- ▶ Naive Bayes
- ▶ Regularized regression

Types of classifiers

General thoughts:

- Trade-off between accuracy and interpretability
- Parameters need to be cross-validated

Frequently used classifiers:

- Naive Bayes
- Regularized regression
- SVM

Types of classifiers

General thoughts:

- Trade-off between accuracy and interpretability
- Parameters need to be cross-validated

Frequently used classifiers:

- Naive Bayes
- Regularized regression
- SVM
- Others: k-nearest neighbors, tree-based methods, etc.

Types of classifiers

General thoughts:

- ► Trade-off between accuracy and interpretability
- ► Parameters need to be cross-validated

Frequently used classifiers:

- ► Naive Bayes
- ► Regularized regression
- ► SVM
- ► Others: k-nearest neighbors, tree-based methods, etc.
- ► Ensemble methods

Regularized regression

Assume we have:

- $i = 1, 2, \ldots, N$ documents
- Each document i is in class $y_i = 0$ or $y_i = 1$
- $j = 1, 2, \ldots, J$ unique features
- And x_{ij} as the count of feature j in document i

We could build a linear regression model as a classifier, using the values of $\beta_0, \beta_1, \ldots, \beta_J$ that minimize:

$$RSS = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{J} \beta_j x_{ij} \right)^2$$

But can we?

- If $J > N$, OLS does not have a unique solution
- Even with $N > J$, OLS has low bias/high variance (overfitting)

Regularized regression

What can we do? Add a penalty for model complexity, such that we now minimize:

$$\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{J}\beta_j^2 \rightarrow \boxed{\text{ridge regression}}$$

or

$$\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{J}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{J}|\beta_j| \rightarrow \boxed{\text{lasso regression}}$$

where λ is the **penalty parameter** (to be estimated)

Regularized regression

Why the penalty (shrinkage)?

- ▶ Reduces the variance

Regularized regression

Why the penalty (shrinkage)?

- ▶ Reduces the variance
- ▶ Identifies the model if $J > N$

Regularized regression

Why the penalty (shrinkage)?

- Reduces the variance
- Identifies the model if $J > N$
- Some coefficients become zero (feature selection)

Regularized regression

Why the penalty (shrinkage)?

- Reduces the variance
- Identifies the model if $J > N$
- Some coefficients become zero (feature selection)

The penalty can take different forms:

Regularized regression

Why the penalty (shrinkage)?

- ▶ Reduces the variance
- ▶ Identifies the model if $J > N$
- ▶ Some coefficients become zero (feature selection)

The penalty can take different forms:

- ▶ Ridge regression: $\lambda \sum_{j=1}^{J} \beta_j^2$ with $\lambda > 0$; and when $\lambda = 0$ becomes OLS

Regularized regression

Why the penalty (shrinkage)?

- Reduces the variance
- Identifies the model if $J > N$
- Some coefficients become zero (feature selection)

The penalty can take different forms:

- Ridge regression: $\lambda \sum_{j=1}^{J} \beta_j^2$ with $\lambda > 0$; and when $\lambda = 0$ becomes OLS
- Lasso $\lambda \sum_{j=1}^{J} |\beta_j|$ where some coefficients become zero.

Regularized regression

Why the penalty (shrinkage)?

- ▶ Reduces the variance
- ▶ Identifies the model if $J > N$
- ▶ Some coefficients become zero (feature selection)

The penalty can take different forms:

- ▶ Ridge regression: $\lambda \sum_{j=1}^{J} \beta_j^2$ with $\lambda > 0$; and when $\lambda = 0$ becomes OLS
- ▶ Lasso $\lambda \sum_{j=1}^{J} |\beta_j|$ where some coefficients become zero.
- ▶ Elastic Net: $\lambda_1 \sum_{j=1}^{J} \beta_j^2 + \lambda_2 \sum_{j=1}^{J} |\beta_j|$ (best of both worlds?)

Regularized regression

Why the penalty (shrinkage)?

- Reduces the variance
- Identifies the model if $J > N$
- Some coefficients become zero (feature selection)

The penalty can take different forms:

- Ridge regression: $\lambda \sum_{j=1}^{J} \beta_j^2$ with $\lambda > 0$; and when $\lambda = 0$ becomes OLS
- Lasso $\lambda \sum_{j=1}^{J} |\beta_j|$ where some coefficients become zero.
- Elastic Net: $\lambda_1 \sum_{j=1}^{J} \beta_j^2 + \lambda_2 \sum_{j=1}^{J} |\beta_j|$ (best of both worlds?)

How to find best value of λ? Cross-validation.

Regularized regression

Why the penalty (shrinkage)?

- ▶ Reduces the variance
- ▶ Identifies the model if $J > N$
- ▶ Some coefficients become zero (feature selection)

The penalty can take different forms:

- ▶ Ridge regression: $\lambda \sum_{j=1}^{J} \beta_j^2$ with $\lambda > 0$; and when $\lambda = 0$ becomes OLS
- ▶ Lasso $\lambda \sum_{j=1}^{J} |\beta_j|$ where some coefficients become zero.
- ▶ Elastic Net: $\lambda_1 \sum_{j=1}^{J} \beta_j^2 + \lambda_2 \sum_{j=1}^{J} |\beta_j|$ (best of both worlds?)

How to find best value of λ? Cross-validation.
Evaluation: regularized regression is easy to interpret, but often outperformed by more complex methods.