Introduction to Text Mining with R

Supervised Classification

Simon Munzert | IPSDS

Supervised classification

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...

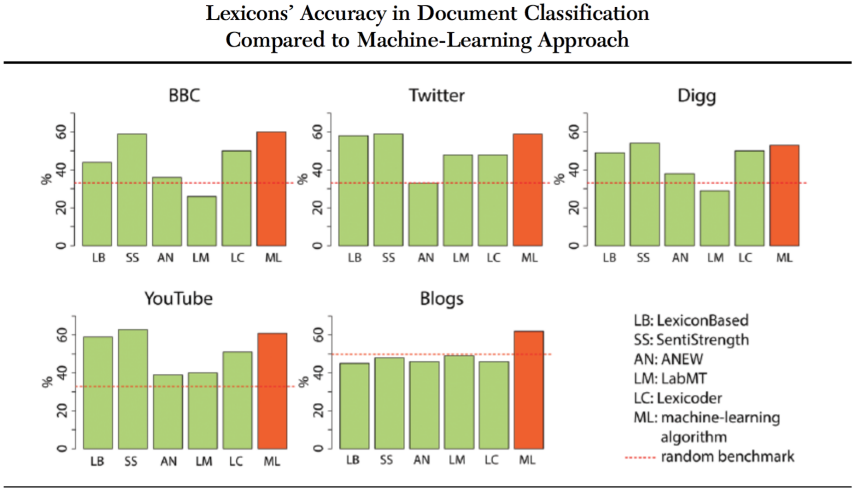
What we need:

- Hand-coded dataset (labeled), to be split into:
  - **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, ensemble methods...
- Approach to validate classifier: **cross-validation**
- **Performance metric** to choose best classifier and avoid overfitting: confusion matrix, accuracy, precision, recall...

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



Source: González-Bailón and Paltoglou (2015)

Supervised v. unsupervised methods

- 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

How to get started

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

Instructions

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:

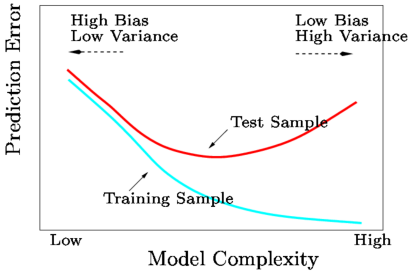
Tweet ID: **447**

Is this tweet related to the ongoing debate about law enforcement and race in the United States?

- ○ Yes
- ○ No
- ○ Don't Know

Measuring performance

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



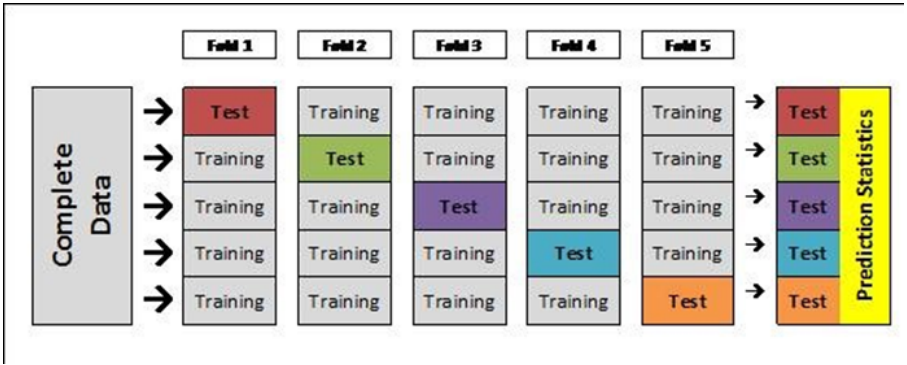
- Model is too complex, describes noise rather than signal (Bias-Variance trade-off)
- Focus on features that perform well in labeled data but may not generalize (e.g. unpopular hashtags)
- In-sample performance better than **out-of-sample performance**

- 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



Performance metrics

Confusion matrix:

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

Performance metrics

Confusion matrix:

	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:

	Actual label	
Classification (algorithm)	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

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

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

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

Types of classifiers

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
- Gradient boosting (XGBoost as popular and powerful variant)

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 β_0, β_1, ..., β_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:

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

or

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

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

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.

Example

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.

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")

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

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

Classifier performance (5-fold cross-validation)

		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