Day 7: Classification and Machine Learning

Kenneth Benoit

Essex Summer School 2013

July 30, 2013

Today's Road Map

Principles of “text as data” approaches

Introduction to the Naive Bayes Classifier

The k-Nearest Neighbour Classifier

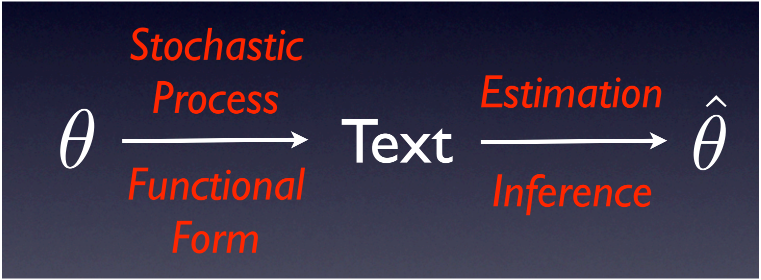
Lab session: Classifying Text Using Wordstat

“TEXT AS DATA”

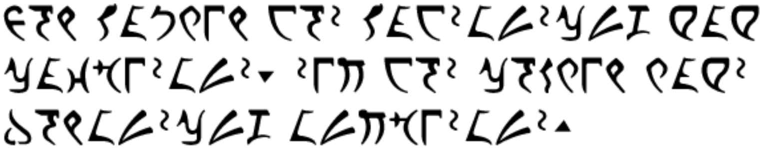
Text as Data: Basic Principles

- Data are observed characteristics of underlying tendencies to be estimated – and therefore not *intrinsically* interesting
- Analysis inherit properties of statistics:
  - Precise characterizations of uncertainty (efficiency of estimators)
  - Concerns with reliability (consistency of estimators)
  - Concerns with validity (unbiasedness of estimators)
- We must be concerned with the stochastic processes generating the data
- We must be concerned with functional relationships between characteristics of texts and authors and observed words

Text generation as a stochastic process



Scale this?



Pros and Cons of the “text as data” approach

- Fully automated technique with minimal human intervention or judgment calls – only with regard to reference text selection
- Language-blind
- (Pro) Inherits all the advantages of statistical data analysis
- (Con) very hard to understand the data-generating process

INTRODUCTION TO NAIVE BAYES

Prior probabilities and updating

A test is devised to automatically flag racist news stories.

- 1% of news stories in general have racist messages
- 80% of racist news stories will be flagged by the test
- 10% of non-racist stories will also be flagged

We run the test on a new news story, and it is *flagged as* racist.

Question: What is probability that the story is *actually* racist?

Any guesses?

Prior probabilities and updating

- What about without the test?
  - Imagine we run 1,000 news stories through the test
  - We expect that 10 will be racist
- With the test, we expect:
  - Of the 10 found to be racist, 8 should be flagged as racist
  - Of the 990 non-racist stories, 99 will be wrongly flagged as racist
  - That's a total of 107 stories flagged as racist
- So: the updated probability of a story being racist, conditional on being flagged as racist, is $\frac{8}{107} = 0.075$
- The *prior* probability of 0.01 is updated to only 0.075 by the positive test result

This is an example of Bayes' Rule:

$$P(R = 1 | T = 1) = \frac{P(T=1|R=1)P(R=1)}{P(T=1)}$$

Multinomial Bayes model of Class given a Word

Consider J word types distributed across I documents, each assigned one of K classes.

At the word level, Bayes Theorem tells us that:

$$P(c_k|w_j) = \frac{P(w_j|c_k)P(c_k)}{P(w_j)}$$

For two classes, this can be expressed as

$$= \frac{P(w_j|c_k)P(c_j)}{P(w_j|c_k)P(c_k) + P(w_j|c_{\neg k})P(c_{\neg k})} \tag{1}$$

Classification as a goal

- Machine learning focuses on identifying classes (classification), while social science is typically interested in locating things on latent traits (scaling)
- One of the simplest and most robust classification methods is the "Naive Bayes" (NB) classifier, built on a Bayesian probability model
- The class predictions for a collection of words from NB are great for classification, but useless for scaling
- But intermediate steps from NB turn out to be excellent for scaling purposes, and identical to Laver, Benoit and Garry's "Wordscores"
- Applying lessons from machine to learning to supervised scaling, we can
  - Apply classification methods to scaling
  - improve it using lessons from machine learning

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 exmplify constrasting 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 v. unsupervised methods: Examples

- General examples:
  - Supervised: Naive Bayes, k-Nearest Neighbor, Support Vector Machines (SVM)
  - Unsupervised: correspondence analysis, IRT models, factor analytic approaches
- Political science applications
  - Supervised: Wordscores (LBG 2003); SVMs (Yu, Kaufman and Diermeier 2008); Naive Bayes (Evans et al 2007)
  - Unsupervised "Wordfish" (Slapin and Proksch 2008); Correspondence analysis (Schonhardt-Bailey 2008); two-dimensional IRT (Monroe and Maeda 2004)

Focus today

- The focus today will be on Naive Bayes
- We will also cover the Laver, Benoit and Garry (2003) "Wordscores" scaling method

Multinomial Bayes model of Class given a Word

Consider J word types distributed across I documents, each assigned one of K classes.

At the word level, Bayes Theorem tells us that:

$$P(c_k|w_j) = \frac{P(w_j|c_k)P(c_k)}{P(w_j)}$$

For two classes, this can be expressed as

$$= \frac{P(w_j|c_k)P(c_j)}{P(w_j|c_k)P(c_k) + P(w_j|c_{\neg k})P(c_{\neg k})} \tag{2}$$

Moving to the document level

- The "Naive" Bayes model of a joint document-level class posterior assumes conditional independence, to multiply the word likelihoods from a "test" document, to produce:

$$P(c|d) = P(c) \, \frac{\prod_j P(w_j|c)}{P(w_j)}$$

- This is why we call it "naive": because it (wrongly) assumes:
  - *conditional independence* of word counts
  - *positional independence* of word counts

Multinomial Bayes model of Class given a Word

Class-conditional word likelihoods

$$P(c_k|w_j) = \frac{P(w_j|c_k)P(c_j)}{P(w_j|c_k)P(c_k) + P(w_j|c_{\neg k})P(c_{\neg k})}$$

- The word likelihood within class
- The maximum likelihood estimate is simply the proportion of times that word j occurs in class k, but it is more common to use Laplace smoothing by adding 1 to each oberved count within class

Multinomial Bayes model of Class given a Word

Word probabilities

$$P(c_k|w_j) = \frac{P(w_j|c_k)P(c_j)}{P(w_j)}$$

- This represents the word probability from the training corpus
- Usually uninteresting, since it is constant for the training data, but needed to compute posteriors on a probability scale

Multinomial Bayes model of Class given a Word

Class prior probabilities

$$P(c_k|w_j) = \frac{P(w_j|c_k)P(c_j)}{P(w_j|c_k)P(c_k) + P(w_j|c_{\neg k})P(c_{\neg k})}$$

- This represents the class prior probability
- Machine learning typically takes this as the document frequency in the training set
- This approach is flawed for scaling, however, since we are scaling the latent class-ness of an unknown document, not predicting class – uniform priors are more appropriate

Multinomial Bayes model of Class given a Word

Class posterior probabilities

$$P(c_k|w_j) = \frac{P(w_j|c_k)P(c_j)}{P(w_j|c_k)P(c_k) + P(w_j|c_{\neg k})P(c_{\neg k})}$$

- This represents the posterior probability of membership in class k for word j
- Under *certain conditions*, this is identical to what LBG (2003) called P_{wr}
- Under those conditions, the LBG "wordscore" is the linear difference between $P(c_k|w_j)$ and $P(c_{\neg k}|w_j)$

Naive Bayes Classification Example

(From Manning, Raghavan and Schütze, *Introduction to Information Retrieval*)

► **Table 13.1** Data for parameter estimation examples.

	docID	words in document	in $c = China$?
training set	1	Chinese Beijing Chinese	yes
	2	Chinese Chinese Shanghai	yes
	3	Chinese Macao	yes
	4	Tokyo Japan Chinese	no
test set	5	Chinese Chinese Chinese Tokyo Japan	?

Naive Bayes Classification Example

Example 13.1: For the example in Table 13.1, the multinomial parameters we need to classify the test document are the priors $\hat{P}(c) = 3/4$ and $\hat{P}(\overline{c}) = 1/4$ and the following conditional probabilities:

$$\begin{aligned}
\hat{P}(\mathsf{Chinese}|c) &= (5+1)/(8+6) = 6/14 = 3/7 \\
\hat{P}(\mathsf{Tokyo}|c) = \hat{P}(\mathsf{Japan}|c) &= (0+1)/(8+6) = 1/14 \\
\hat{P}(\mathsf{Chinese}|\overline{c}) &= (1+1)/(3+6) = 2/9 \\
\hat{P}(\mathsf{Tokyo}|\overline{c}) = \hat{P}(\mathsf{Japan}|\overline{c}) &= (1+1)/(3+6) = 2/9
\end{aligned}$$

The denominators are $(8+6)$ and $(3+6)$ because the lengths of $text_c$ and $text_{\overline{c}}$ are 8 and 3, respectively, and because the constant B in Equation (13.7) is 6 as the vocabulary consists of six terms.

We then get:

$$\begin{aligned}
\hat{P}(c|d_5) &\propto 3/4 \cdot (3/7)^3 \cdot 1/14 \cdot 1/14 \approx 0.0003. \\
\hat{P}(\overline{c}|d_5) &\propto 1/4 \cdot (2/9)^3 \cdot 2/9 \cdot 2/9 \approx 0.0001.
\end{aligned}$$

Thus, the classifier assigns the test document to c = *China*. The reason for this classification decision is that the three occurrences of the positive indicator Chinese in d_5 outweigh the occurrences of the two negative indicators Japan and Tokyo.

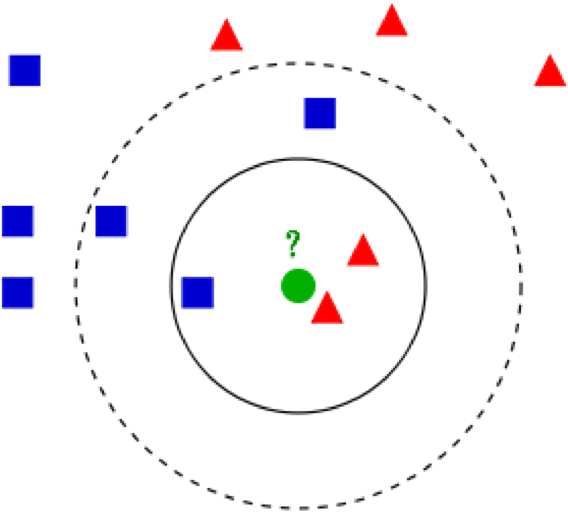
From Classification to Scaling

- The class predictions for a collection of words from NB can be adapted to scaling
- The intermediate steps from NB turn out to be excellent for scaling purposes, and identical to Laver, Benoit and Garry's "Wordscores"
- There are certain things from machine learning that ought to be adopted when classification methods are used for scaling
  - Feature selection
  - Stemming/pre-processing

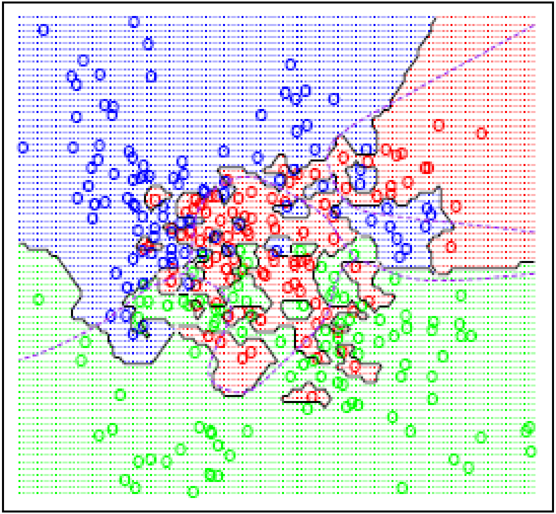
Other classification methods: *k*-nearest neighbour

- A non-parametric method for classifying objects based on the training examples taht are *closest* in the feature space
- A type of instance-based learning, or "lazy learning" where the function is only approximated locally and all computation is deferred until classification
- An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its k nearest neighbors (where k is a positive integer, usually small)
- Extremely *simple*: the only parameter that adjusts is k (number of neighbors to be used) - increasing k *smooths* the decision boundary

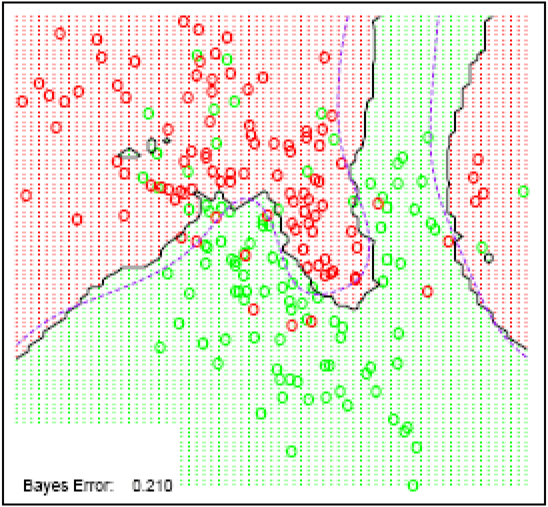
k-NN Example: Red or Blue?



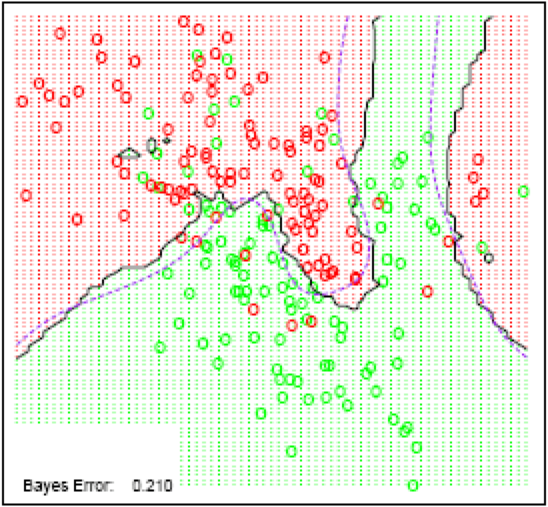
$k = 1$



$k = 7$



$k = 15$



k-nearest neighbour issues: Dimensionality

- Distance usually relates to all the attributes and assumes all of them have the same effects on distance
- Misclassification may results from attributes not confirming to this assumption (sometimes called the “curse of dimensionality”) – solution is to reduce the dimensions
- There are (many!) different *metrics* of distance