Classification with Naïve Bayes

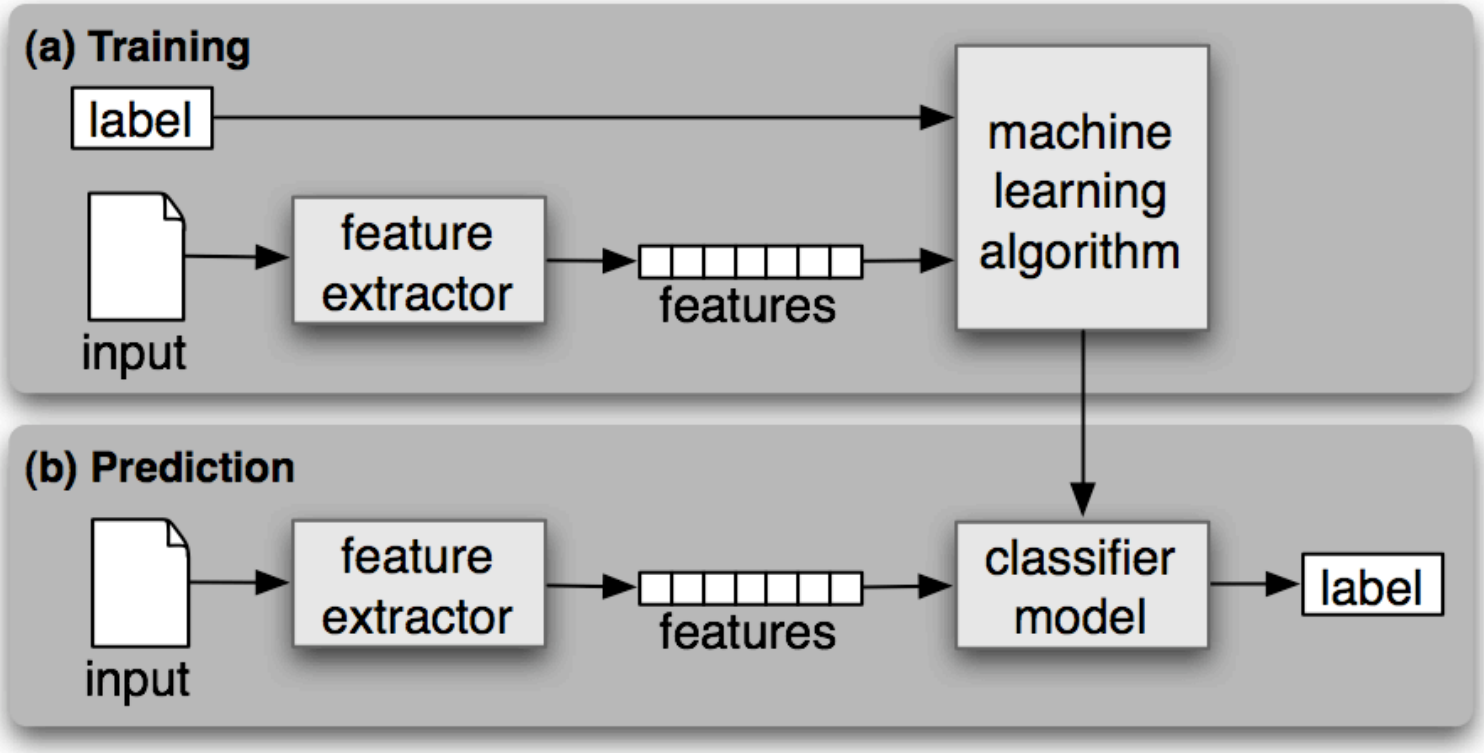
MAS.S60

Rob Speer

Catherine Havasi

Supervised classification

- Distinguish things from other things based on examples



Steps in supervised classification

- Collect a corpus
- Separate it into training and test data
- Create a **feature extractor**
- Train a classifier with those features
- Test it on *some of* your test data
- Repeat

Applications

- Spam filtering
  - "A Plan for Spam" (Paul Graham, 2002)
  - Later, detecting **important** e-mails
- Topic detection
- Language detection
- Affect (emotion) detection

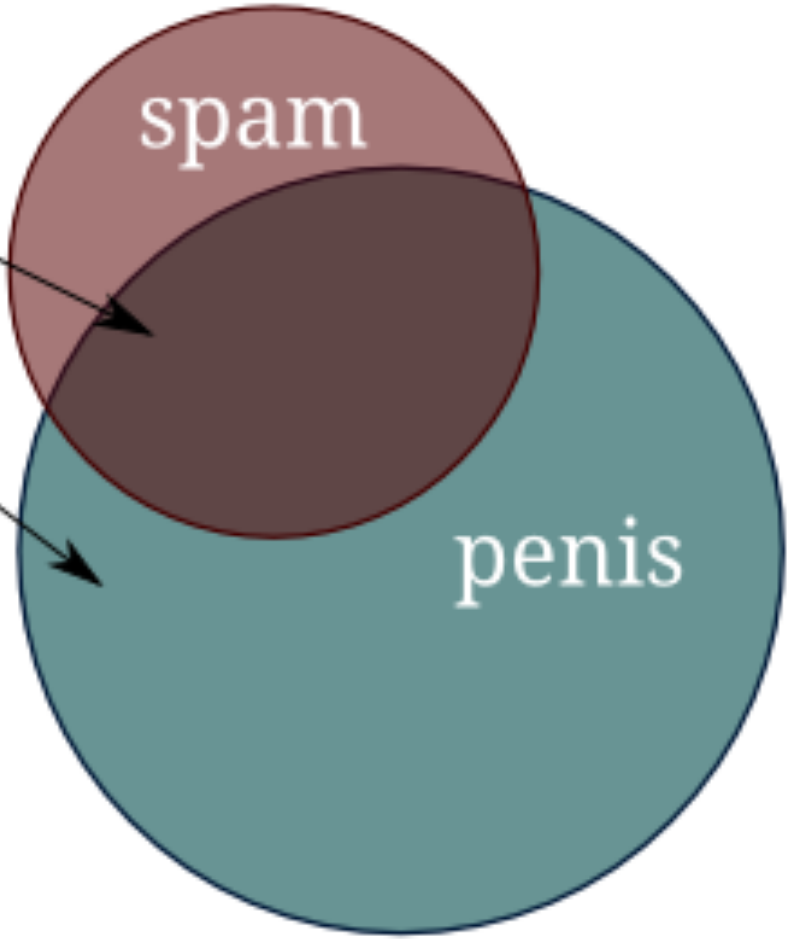
Applications

- Spam filtering
  - "A Plan for Spam" (Paul Graham, 2002)
  - Later, detecting **important** e-mails
- Topic detection
- Language detection
- Affect (emotion) detection
- Determining when to say "That's what she said" (Kiddon and Brun, 2011)

Bayesian probability

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

$$= \frac{P(B|A)P(A)}{P(B)}$$



Alexandru Nedelcu, “How to Build a Naive Bayes Classifier”

Combining probabilities

- Alexandru gets some interesting e-mail
- He has 74 e-mails in his training set
- He has marked 30 of them as spam
- 51 contain the word “penis”, 20 of those marked as spam
- 25 contain the word “viagra”, 24 of those marked as spam
- How do you compute P(spam | penis, viagra)?

Messy joint probabilities

$$\frac{P(penis|spam \cap viagra) * P(viagra|spam) * P(spam)}{P(penis|viagra) * P(viagra)}$$

Pretend all observations are independent!

$$P(spam|penis, viagra)$$

$$= \frac{P(penis|spam) * P(viagra|spam) * P(spam)}{P(penis) * P(viagra)}$$

$$= \frac{\frac{24}{30} * \frac{20}{30} * \frac{30}{74}}{\frac{25}{74} * \frac{51}{74}} = 0.928$$

Demo: classifying names

Lab: classifying movie reviews

- We're going to bring back the venerable bag of words
- Doesn't quite fit into NLTK's features
- We could write an adapter
- Or with a few more lines of code, we can just write naïve Bayes from scratch

Beware of overfitting

Example (Movie review categorization)

```
featuresets = [(document_features(d), c) for (d,c) in
    documents]
train_set, test_set = featuresets[100:], featuresets[:100]
classifier = nltk.NaiveBayesClassifier.train(train_set)
print nltk.classify.accuracy(classifier, test_set)
# 0.86
classifier.show_most_informative_features(5)
Most Informative Features
 contains(outstanding) = True  pos : neg = 11.1 : 1.0
      contains(seagal) = True  neg : pos =  7.7 : 1.0
 contains(wonderfully) = True  pos : neg =  6.8 : 1.0
       contains(damon) = True  pos : neg =  5.9 : 1.0
      contains(wasted) = True  neg : pos =  5.8 : 1.0
```

Beware of creepiness

Mail Online

Home | U.K. Home | News | Sport | U.S. Showbiz | Femail | Health | Science | Money | RightM

News Home | Arts | Headlines | Pictures | Most read | News Board

How Target knows when its shoppers are pregnant - and figured out a teen was before her father did

By NINA GOLGOWSKI

Last updated at 1:19 PM on 18th February 2012

Hearing that stores like Target monitor their customers' spending habits to determine their future ones may not be much of a shock to most.

What may be more of one is that Target's department for Guest Marketing Analytics is so in tune with their consumers based on their spending, they can predict major changes in their lives.

Revisiting the Twitter API

- Twitter's tracking API relies on spaces, so it does not understand East Asian languages
- Has anyone found a way to work around this?

Revisiting PMI

- Pointwise Mutual Information makes sense in theory, but has one really unsatisfying edge case
- Dunning, 1993: likelihood ratio or "G-test"
- "On Log-Likelihood-Ratios and the Significance of Rare Events" (Robert Moore, 2004)
  - "It is clear that if standard statistical tests are used naively, the results make no sense in these applications"

Dunning's G-test

- NLTK calls this the "likelihood ratio" test

MI(vice, president)	MI(~vice, president)
MI(vice, ~president)	MI(~vice, ~president)

- Find the MI of all the things you see and all the things you *don't* see
- Weight them by the number of observations

No assignment this week

- Take this opportunity to catch up if you've fallen behind