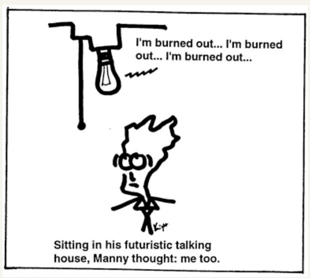


http://www.isi.edu/natural-language/people/knight3.html

Text Classification

David Kauchak
cs458
Fall 2012

adapted from:
http://www.stanford.edu/class/cs276/handouts/lecture10-textcat-naivebayes.ppt
http://www.stanford.edu/class/cs276/handouts/lecture11-vector-classify.ppt
http://www.stanford.edu/class/cs276/handouts/lecture12-SVMs.ppt

Administrative

- Lunch talk today
- CS lunch tomorrow, Ross – LaForce 121
- Finalized proposals
- Start working on the project now!

Git repository

https://github.com/dkauchak/cs458-f12.git

Getting your own copy:
- sign up for a github account
- https://help.github.com/articles/fork-a-repo

Other Tutorials:
- http://schacon.github.com/git/gittutorial.html
- http://www.cs.middlebury.edu/~dkauchak/classes/s12/cs312/lectures/lecture4-git.pdf

Git

Each project will “fork” their own GitHub project

Your team can interact with this project as much as you want without affecting the general project

When you want to merge with the main code base:

- git pull upstream master
  (make sure you have the latest changes)
- git status
  (Make sure all the files you’re using are in the git repository)
- Make sure your code compiles!
- Make sure your code runs (run your tests)
- git push origin master
- Issue a pull request on github

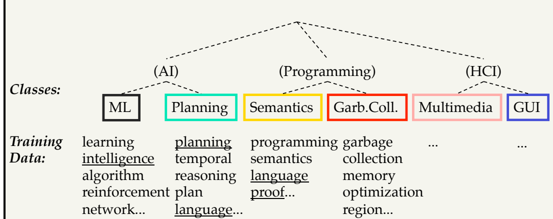
Git

Don’t wait too long to merge with the main project

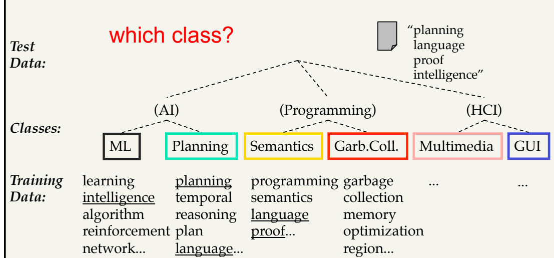
But… don’t bug me too often with pull requests

I’ll manage the project repository for now… I won’t be very happy if you issue pull requests that break the main code base ☹

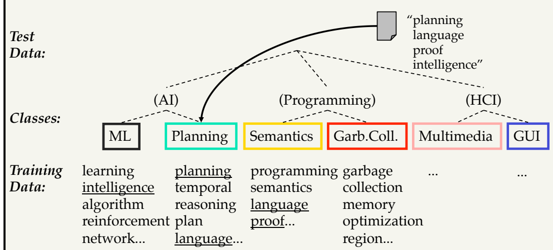
Document Classification: training



Document Classification: testing

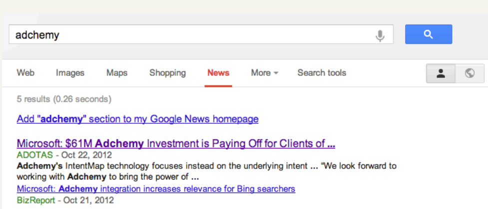


Document Classification: testing



How might this be useful for IR?

Standing queries



Standing queries

You have an information need, say:

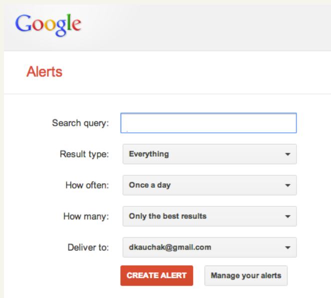
- Unrest in the Niger delta region
- Adchemy, Inc
- …

You want to rerun an appropriate query periodically to find new news items on this topic

You will be sent new documents that are *found*

- it's classification not ranking

Standing queries



Spam filtering

From: "" <takworlld@hotmail.com>
Subject: real estate is the only way... gem oalvgkay

Anyone can buy real estate with no money down

Stop paying rent TODAY !

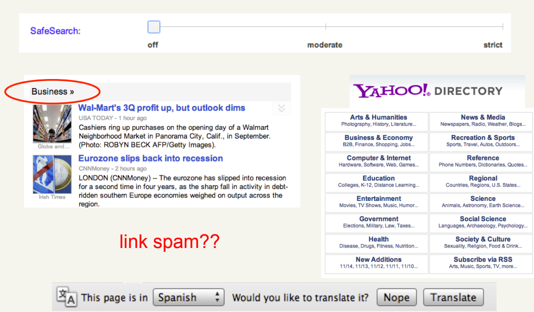
There is no need to spend hundreds or even thousands for similar courses

I am 22 years old and I have already purchased 6 properties using the methods outlined in this truly INCREDIBLE ebook.

Change your life NOW !

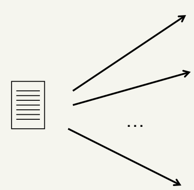
===
Click Below to order:
http://www.wholesaledaily.com/sales/nmd.htm
===

Many other applications…



link spam??

How would you do it?



Pros and cons of different approaches?

Manual approach

- Used by Yahoo! (originally; now present but downplayed), Looksmart, about.com, ODP, PubMed
- Very accurate when job is done by experts
- Consistent when the problem size and team is small
- Difficult and expensive to scale
  - Means we need automatic classification methods for big problems

A slightly better manual approach

- Hand-coded, rule-based systems
- A few companies provide an "IDE" for writing such rules
- Accuracy is often very high if a rule has been carefully refined over time by a subject expert
- Building and maintaining these rules is expensive

A complex classification rule

```
comment line                  # Beginning of art topic definition
top-level topic               art ACCRUE
                                  /author = "fsmith"
topic definition modifiers        /date   = "30-Dec-01"
                                  /annotation = "Topic created
                                                 by fsmith"
subtopictopic                 * 0.70 performing-arts ACCRUE
  evidencetopic               ** 0.50 WORD
  topic definition modifier       /wordtext = ballet
  evidencetopic               ** 0.50 STEM
  topic definition modifier       /wordtext = dance
  evidencetopic               ** 0.50 WORD
  topic definition modifier       /wordtext = opera
  evidencetopic               ** 0.30 WORD
  topic definition modifier       /wordtext = symphony
subtopic                      * 0.70 visual-arts ACCRUE
                              ** 0.50 WORD
                                  /wordtext = painting
                              ** 0.50 WORD
                                  /wordtext = sculpture
subtopic                      * 0.70 film ACCRUE
                              ** 0.50 STEM
                                  /wordtext = film
subtopic                      ** 0.50 motion-picture PHRASE
                              *** 1.00 WORD
                                  /wordtext = motion
                              *** 1.00 WORD
                                  /wordtext = picture
                              ** 0.50 STEM
                                  /wordtext = movie
subtopic                      * 0.50 video ACCRUE
                              ** 0.50 STEM
                                  /wordtext = video
                              ** 0.50 STEM
                                  /wordtext = vcr
                              # End of art topic
```

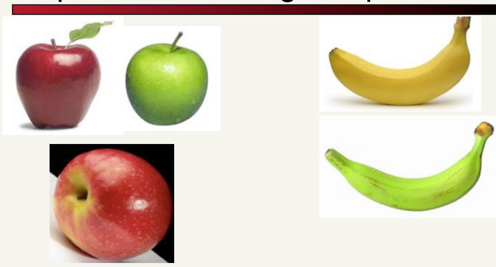
maintenance issues!

Hand-weighting of terms

Automated approaches

- Supervised learning of a document-label assignment function
  - Many systems partly rely on machine learning (Autonomy, MSN, Verity, Enkata, Yahoo!, …)
    - k-Nearest Neighbors (simple, powerful)
    - Naive Bayes (simple, common method)
    - Support-vector machines (new, more powerful)
    - … plus many other methods
- Many commercial systems use a mixture of methods
- Pros/Cons?
- Results can be very good!
- **No free lunch**: requires hand-classified training data

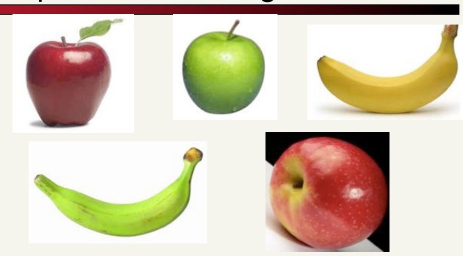
Supervised learning setup



APPLES **BANANAS**

Given labeled data…

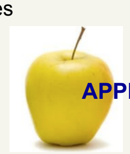
Unsupervised learning



Unupervised learning: given data, but no labels

Supervised learning

Given labeled examples, learn to label unlabeled examples

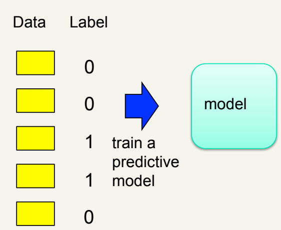


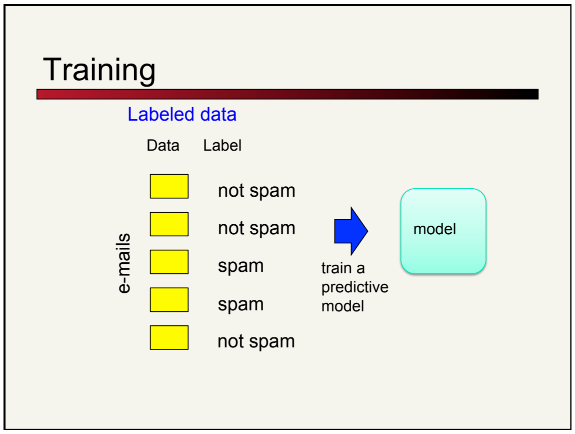
APPLE or BANANA?

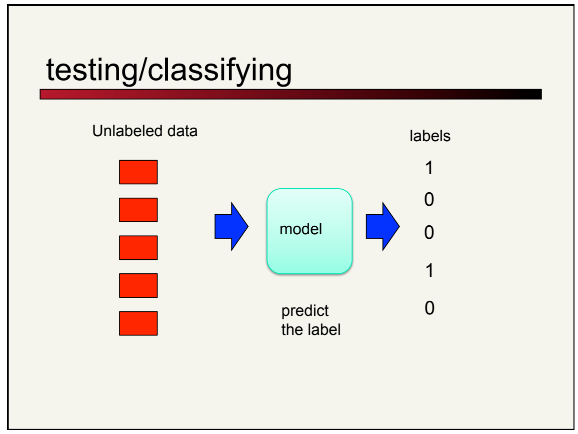
learn to classify unlabeled

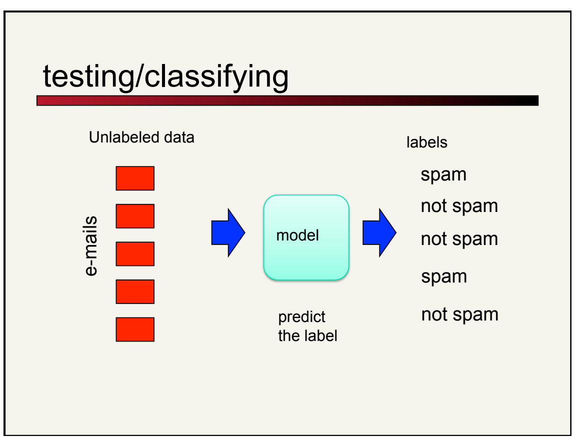
Training

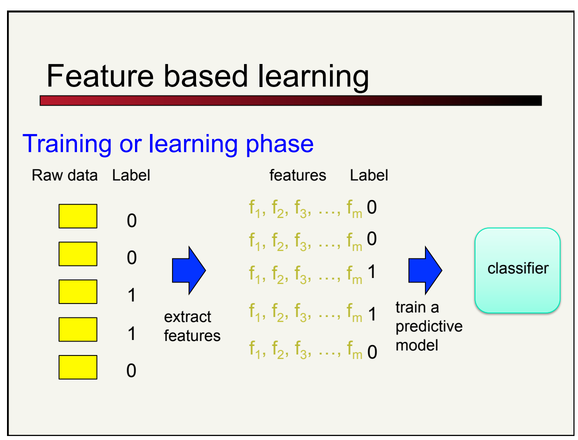
Labeled data

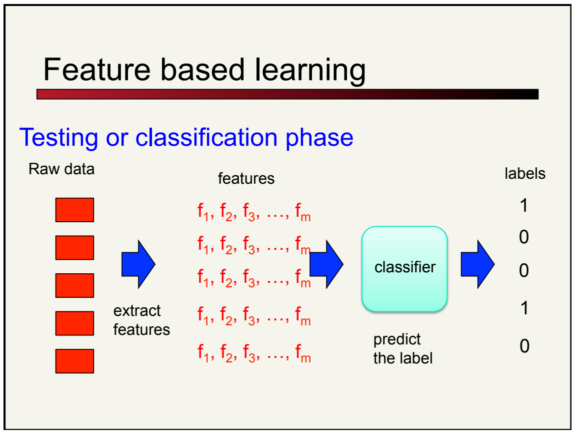


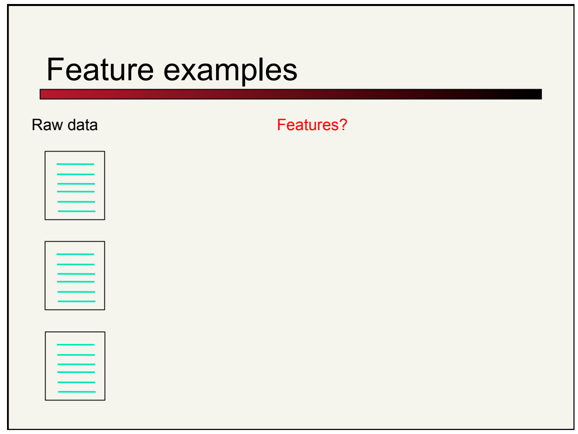

Training
Labeled data
Data
Label
e-mails
not spam
not spam
spam
spam
not spam
train a
predictive
model
model

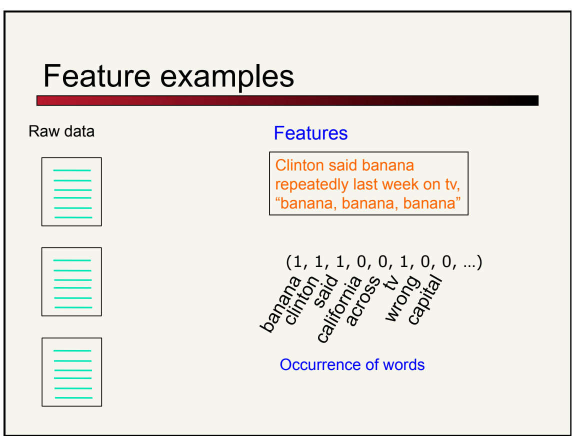

testing/classifying
Unlabeled data
model
predict
the label
labels
1
0
0
1
0

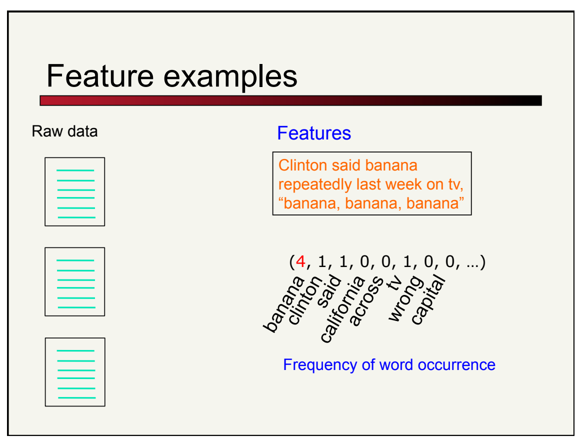

testing/classifying
Unlabeled data
e-mails
model
predict
the label
labels
spam
not spam
not spam
spam
not spam


Feature based learning
Training or learning phase
Raw data
Label
0
0
1
1
0
extract
features
features
Label
$f_1, f_2, f_3, \ldots, f_m$ 0
$f_1, f_2, f_3, \ldots, f_m$ 0
$f_1, f_2, f_3, \ldots, f_m$ 1
$f_1, f_2, f_3, \ldots, f_m$ 1
$f_1, f_2, f_3, \ldots, f_m$ 0
train a
predictive
model
classifier

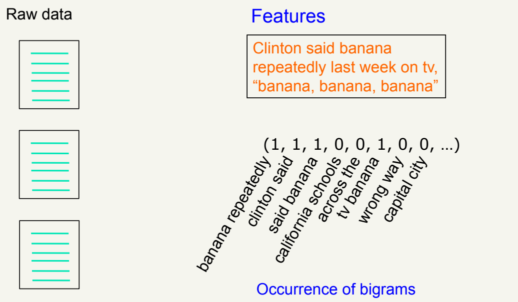

Feature based learning
Testing or classification phase
Raw data
features
labels
f1, f2, f3, …, fm
f1, f2, f3, …, fm
f1, f2, f3, …, fm
f1, f2, f3, …, fm
f1, f2, f3, …, fm
extract features
classifier
predict the label
1
0
0
1
0


Feature examples
Raw data
Features?


Feature examples
Raw data
Features
Clinton said banana repeatedly last week on tv, "banana, banana, banana"
(1, 1, 1, 0, 0, 1, 0, 0, …)
banana
clinton
said
california
across
tv
wrong
capital
Occurrence of words


Feature examples
Raw data
Features
Clinton said banana repeatedly last week on tv, "banana, banana, banana"
(4, 1, 1, 0, 0, 1, 0, 0, …)
banana
clinton
said
california
across
tv
wrong
capital
Frequency of word occurrence

Feature examples



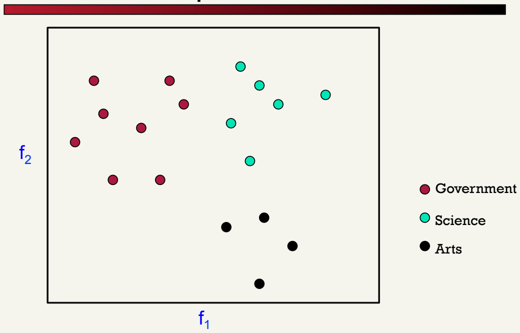
Lots of other features

- POS: occurrence, counts, sequence
- Constituents
- Whether 'V1agra' occurred 15 times
- Whether 'banana' occurred more times than 'apple'
- If the document has a number in it
- …
- Features are very important, but we're going to focus on the models today

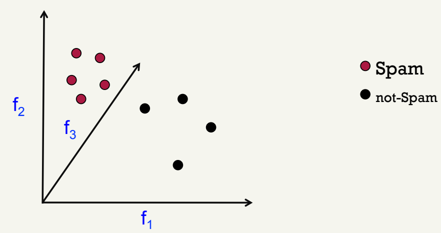
Power of feature-base methods

General purpose: any domain where we can represent a data point as a set of features, we can use the method

The feature space



The feature space



Feature space

$f_1, f_2, f_3, \ldots, f_m$ m-dimensional space

How big will m be for us?

Bayesian Classification

We represent a data item based on the features:

$$D = \langle f_1, f_2, \ldots, f_n \rangle$$

Training

a: $p(a \mid D) = p(a \mid f_1, f_2, \ldots, f_n)$

b: $p(b \mid D) = p(b \mid f_1, f_2, \ldots, f_n)$

$$\to P(Label \mid f_1, f_2, \ldots, f_n)$$

For each label/class, **learn** a probability distribution based on the features

Bayesian Classification

We represent a data item based on the features:

$$D = \langle f_1, f_2, \ldots, f_n \rangle$$

Classifying

$$label = \operatorname*{argmax}_{l \in Labels} P(l \mid f_1, f_2, \ldots, f_n)$$

Given an *new* example, classify it as the label with the largest conditional probability

Bayes' Rule

$$P(C,D) = P(C \mid D)P(D) = P(D \mid C)P(C)$$

$$\boxed{P(C \mid D) = \frac{P(D \mid C)P(C)}{P(D)}}$$

How can we use this?

Bayes rule for classification

conditional (posterior) probability

prior probability

$$P(Label \mid Data) = \frac{P(D \mid L)P(L)}{P(D)}$$

Why not model P(Label|Data) directly?

Bayesian classifiers

$$label = \underset{l \in Labels}{\operatorname{argmax}} P(l \mid f_1, f_2, \ldots, f_n)$$

different distributions for different labels

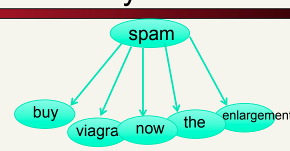
$$= \underset{l \in Labels}{\operatorname{argmax}} \frac{P(f_1, f_2, \ldots, f_n \mid l)P(l)}{P(f_1, f_2, \ldots, f_n)}$$

Bayes rule

$$= \underset{l \in Labels}{\operatorname{argmax}} P(f_1, f_2, \ldots, f_n \mid l)P(l)$$

two models to learn *for each label/class*

The Naive Bayes Classifier



assume binary features for now

Conditional Independence Assumption: features are independent of each other given the class:

$$P(x_1, \ldots, x_n) = P(f_1, f_2, \ldots, f_n \mid l)P(l)$$

$$P(x_1, \ldots, x_n \mid l) = P(x_1 \mid l)P(x_2 \mid l) \cdots P(x_n \mid l)$$

Estimating parameters

p('v1agra' | spam)

p('the' | spam)

p('enlargement' | not-spam)

…

For us:

$$label = \underset{l \in Labels}{\operatorname{argmax}} P(f_1 \mid l) P(f_2 \mid l) \ldots p(f_n \mid l) P(l)$$

How do we estimate these probabilities?

Maximum likelihood estimates

$$\hat{P}(l) = \frac{N(l)}{N} \qquad \frac{\text{number of items with label}}{\text{total number of items}}$$

$$\hat{P}(f_i \mid l) = \frac{N(f_i, l)}{N(l)} \qquad \frac{\text{number of items with the label with feature}}{\text{number of items with label}}$$

Naïve Bayes Text Classification

Features: word occurring in a document (though others could be used…)

$$label = \underset{l \in Labels}{\operatorname{argmax}} P(word_1 \mid l) P(word_2 \mid l) \ldots p(word_n \mid l) P(l)$$

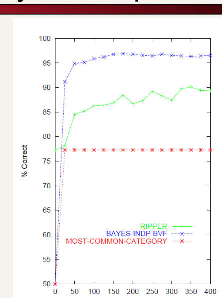
Does the Naïve Bayes assumption hold?

- Are word occurrences independent given the label?

Lot's of text classification problems

- sentiment analysis: positive vs. negative reviews
- category classification
- spam

Naive Bayes on spam email



http://www.cnbc.cmu.edu/~jp/research/email.paper.pdf

SpamAssassin

Naive Bayes has found a home in spam filtering
- Paul Graham's *A Plan for Spam*
  - A mutant with more mutant offspring...
- Naive Bayes-like classifier with weird parameter estimation
- Widely used in spam filters
- But also many other things: black hole lists, etc.

Many email topic filters also use NB classifiers

NB: The good and the bad

Good
- Easy to understand
- Fast to train
- Reasonable performance

Bad
- We can do better
- Independence assumptions are rarely true
- Smoothing is challenging
- Feature selection is usually required

Recall: Vector Space Representation

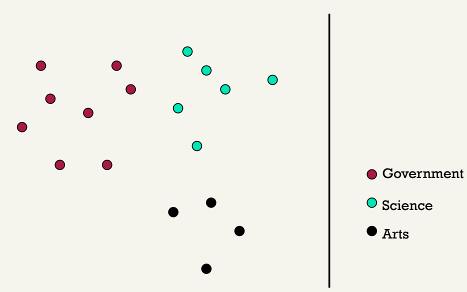
Each document is a vector, one component for each term/word

Normally normalize vectors to unit length

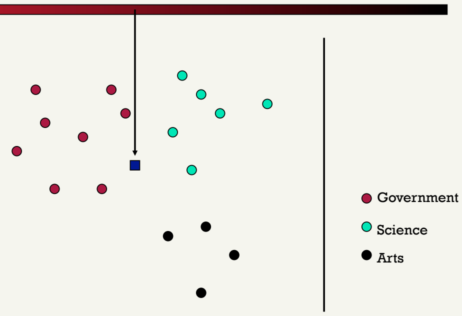
High-dimensional vector space:
- Terms are axes
- 10,000+ dimensions, or even 100,000+
- Docs are vectors in this space

How can we do classification in this space?

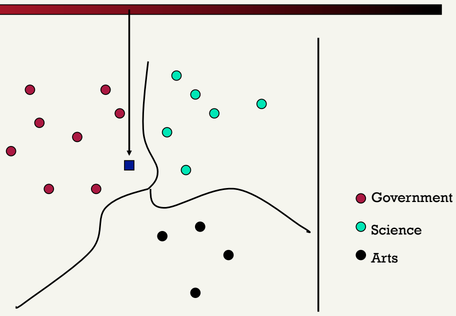
Documents in a Vector Space



Test Document of what class?



Test Document = Government

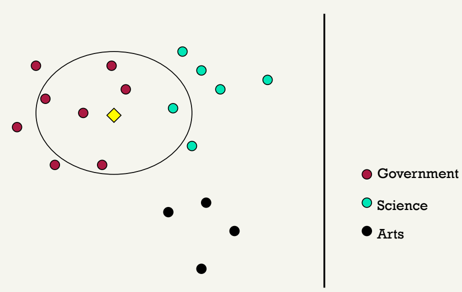


k-Nearest Neighbor (k-NN)

To classify document ***d***:

- Find ***k*** nearest neighbors of ***d***
- Choose as the class the majority class within the ***k*** nearest neighbors

Example: k=6 (6-NN)

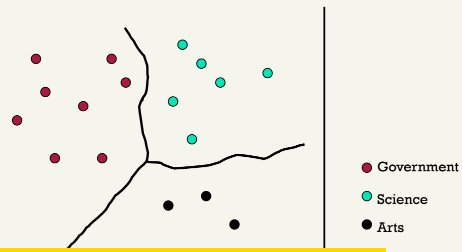


k Nearest Neighbor

What value of k should we use?

- Using only the closest example (1NN) to determine the class is subject to errors due to:
  - A single atypical example
  - Noise
- Pick k too large and you end up with looking at neighbors that are not that close
- Value of *k* is typically odd to avoid ties; 3 and 5 are most common.

k-NN decision boundaries



k-NN gives locally defined decision boundaries between classes – far away points do not influence each classification decision (unlike in Naïve Bayes, etc.)

Similarity Metrics

Nearest neighbor methods depends on a similarity (or distance) metric

Ideas?

Euclidean distance.

Binary instance space is *Hamming distance* (number of feature values that differ)

For text, cosine similarity of tf.idf weighted vectors is typically most effective

k-NN: The good and the bad

- Good
  - No training is necessary
  - No feature selection necessary
  - Scales well with large number of classes
    - Don't need to train *n* classifiers for *n* classes
- Bad
  - Classes can influence each other
    - Small changes to one class can have ripple effect
  - Scores can be hard to convert to probabilities
  - Can be more expensive at test time
  - "Model" is all of your training examples which can be large

Bias/variance trade-off

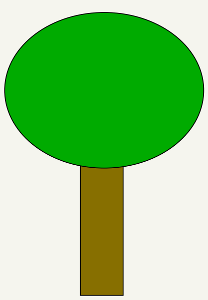
Is this a tree?

Bias/variance trade-off

Is this a tree?

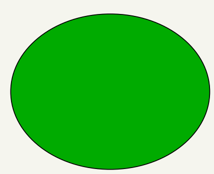
Bias/variance trade-off

Is this a tree?



Bias/variance trade-off

Is this a tree?



Bias/Variance

Bias: How well does the model predict the training data?

- high bias – the model doesn't do a good job of predicting the training data (high training set error)
- The model predictions are *biased by the model*

Variance: How sensitive to the training data is the learned model?

- high variance – changing the training data can drastically change the learned model

Bias/Variance

Another way to think about it is model complexity

Simple models

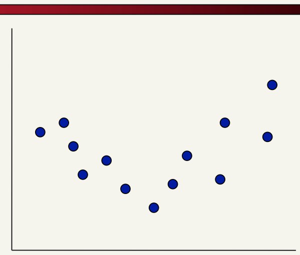
- may not model data well
- high bias

Complicated models

- may overfit to the training data
- high variance

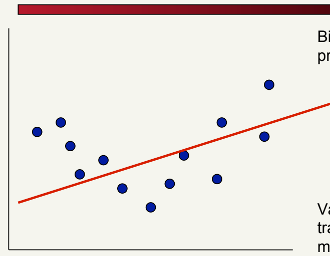
Why do we care about bias/variance?

Bias/variance trade-off



We want to fit a polynomial to this, which one should we use?

Bias/variance trade-off



Bias: How well does the model predict the training data?

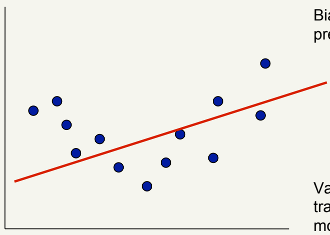
- high bias – the model doesn't do a good job of predicting the training data (high training set error)
- The model predictions are biased by the model

Variance: How sensitive to the training data is the learned model?

- high variance – changing the training data can drastically change the learned model

High variance OR high bias?

Bias/variance trade-off



High bias

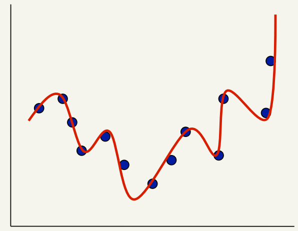
Bias: How well does the model predict the training data?

- high bias – the model doesn't do a good job of predicting the training data (high training set error)
- The model predictions are biased by the model

Variance: How sensitive to the training data is the learned model?

- high variance – changing the training data can drastically change the learned model

Bias/variance trade-off



High variance OR high bias?

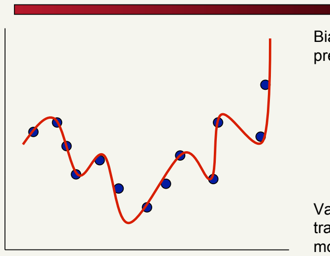
Bias: How well does the model predict the training data?

- high bias – the model doesn't do a good job of predicting the training data (high training set error)
- The model predictions are biased by the model

Variance: How sensitive to the training data is the learned model?

- high variance – changing the training data can drastically change the learned model

Bias/variance trade-off



High variance

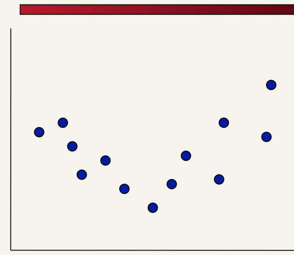
Bias: How well does the model predict the training data?

- high bias – the model doesn't do a good job of predicting the training data (high training set error)
- The model predictions are biased by the model

Variance: How sensitive to the training data is the learned model?

- high variance – changing the training data can drastically change the learned model

Bias/variance trade-off



What do we want?

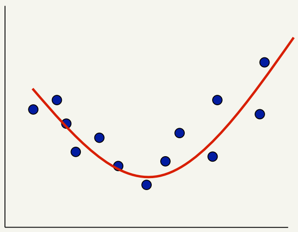
Bias: How well does the model predict the training data?

- high bias – the model doesn't do a good job of predicting the training data (high training set error)
- The model predictions are biased by the model

Variance: How sensitive to the training data is the learned model?

- high variance – changing the training data can drastically change the learned model

Bias/variance trade-off



Compromise between bias and variance

Bias: How well does the model predict the training data?

- high bias – the model doesn't do a good job of predicting the training data (high training set error)
- The model predictions are biased by the model

Variance: How sensitive to the training data is the learned model?

- high variance – changing the training data can drastically change the learned model

k-NN vs. Naive Bayes

How do k-NN and NB sit on the variance/ bias spectrum?

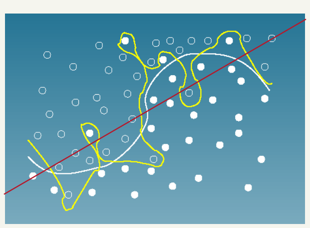
k-NN has high variance and low bias.

- more complicated model
- can model any boundary
- but very dependent on the training data

NB has low variance and high bias.

- Decision surface has to be linear
- Cannot model all data
- but, less variation based on the training data

Bias vs. variance: Choosing the correct model capacity



Which separating line should we use?