Text Classification – Naïve Bayes

June 17, 2016

Credits for slides: Allan, Arms, Manning, Lund, Noble, Page.

Why Text Classification?

- Users may have ongoing information needs
  - Might want to track developments in a particular topic such as “multicore computer chips”

- The classification of documents by topic capture the generality and scope of the problem space.

Classification Problems

- Email filtering: spam / non spam
- Email foldering / tagging: Work, Friends, Family, Hobby
- Research articles by topics: Machine Learning, Data Mining, Algorithms
- Sentiment Analysis: positive / negative
- Emotion Detection: anger, happiness, joy, sadness, etc.
- Tumor: malignant / benign
- Medical diagnosis: Not ill, Cold, Flu

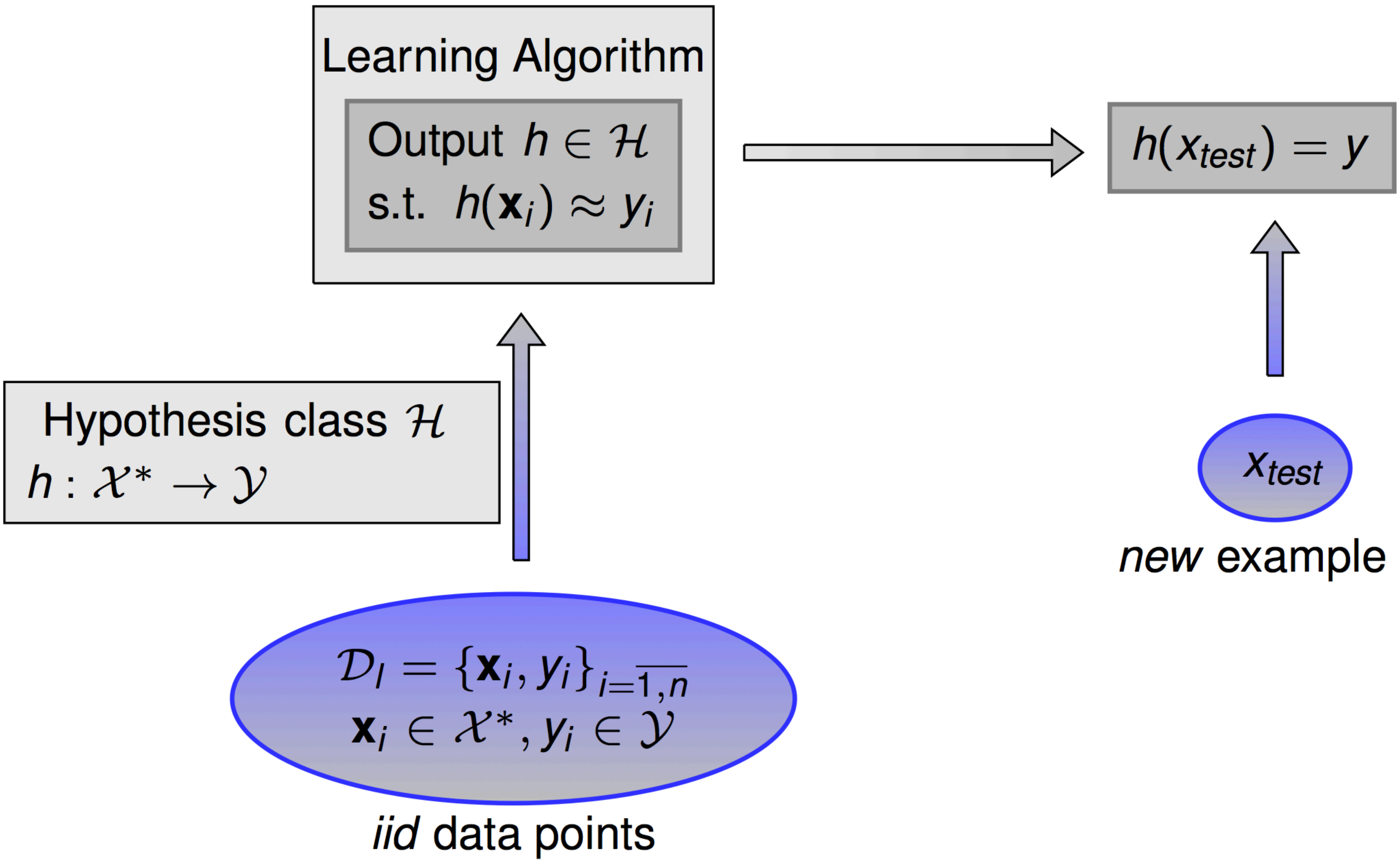
Data Representation

day	outlook	temperature	humidity	wind	play
1	sunny	hot	high	weak	no
2	sunny	hot	high	strong	no
3	overcast	hot	high	weak	yes
4	rain	mild	high	weak	yes
5	rain	cool	normal	weak	yes
6	rain	cool	normal	strong	no
7	overcast	cool	normal	strong	yes
8	sunny	mild	high	weak	no
9	sunny	cool	normal	weak	yes
10	rain	mild	normal	weak	yes
11	sunny	mild	normal	strong	yes
12	overcast	mild	high	strong	yes
13	overcast	hot	normal	weak	yes
14	rain	mild	high	strong	no

Data Representation

- N = number of training examples
- x's = “input” variable / features
- y's = “output” variable / “target” variable
- (x,y) – one training example
- $(x^{(i)},y^{(i)})$ – the i^{th} training example

Training and Classification



Summary of Basic Probability Formulas

- Product rule: probability of a conjunction of two events A and B

$$P(A \wedge B) = P(A \mid B)P(B) = P(B \mid A)P(A)$$

- Sum rule: probability of a disjunction of two events A and B

$$P(A \vee B) = P(A) + P(B) - P(A \wedge B)$$

- Bayes theorem: the posterior probability of A given B

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

- Theorem of total probability: if events $A_1, \ldots, A_n$ are mutually exclusive with $\sum_{i=1}^{n} P(Ai) = 1$

$$P(B) = \sum_{i=1}^{n} P(B \mid A_i)P(A_i)$$

Bayes Classifiers for Categorical Data

Task: Classify a new instance *x* based on a tuple of attribute values $x = \langle x_1, x_2, \ldots, x_n \rangle$ into one of the classes $c_j \in C$

$$c_{MAP} = \underset{c_j \in C}{\operatorname{argmax}} P(c_j \mid x_1, x_2, \ldots, x_n)$$

$$= \underset{c_j \in C}{\operatorname{argmax}} \frac{P(x_1, x_2, \ldots, x_n \mid c_j) P(c_j)}{P(x_1, x_2, \ldots, x_n)}$$

$$= \underset{c_j \in C}{\operatorname{argmax}} P(x_1, x_2, \ldots, x_n \mid c_j) P(c_j)$$

Example	Color	Shape	Class
1	red	circle	positive
2	red	circle	positive
3	red	square	negative
4	blue	circle	negative

← attributes

← values

Joint Distribution

- The joint probability distribution for a set of random variables, $X_1,\ldots,X_n$ gives the probability of every combination of values: $P(X_1,\ldots,X_n)$

positive

	circle	square
red	0.20	0.02
blue	0.02	0.01

negative

	circle	square
red	0.05	0.30
blue	0.20	0.20

- The probability of all possible conjunctions can be calculated by summing the appropriate subset of values from the joint distribution.

$$P(red \wedge circle) = ?$$

$$P(red) = ?$$

Joint Distribution

- The joint probability distribution for a set of random variables, $X_1,\ldots,X_n$ gives the probability of every combination of values: $P(X_1,\ldots,X_n)$

positive

	circle	square
red	0.20	0.02
blue	0.02	0.01

negative

	circle	square
red	0.05	0.30
blue	0.20	0.20

- The probability of all possible conjunctions can be calculated by summing the appropriate subset of values from the joint distribution.

$$P(red \wedge circle) = 0.20 + 0.05 = 0.25$$

$$P(red) = 0.20 + 0.02 + 0.05 + 0.3 = 0.57$$

Joint Distribution

- The joint probability distribution for a set of random variables, $X_1,\ldots,X_n$ gives the probability of every combination of values: $P(X_1,\ldots,X_n)$

positive

	circle	square
red	0.20	0.02
blue	0.02	0.01

negative

	circle	square
red	0.05	0.30
blue	0.20	0.20

- The probability of all possible conjunctions can be calculated by summing the appropriate subset of values from the joint distribution.

$$P(red \wedge circle) = 0.20 + 0.05 = 0.25$$

$$P(red) = 0.20 + 0.02 + 0.05 + 0.3 = 0.57$$

- Therefore, all conditional probabilities can also be calculated.

Joint Distribution

- The joint probability distribution for a set of random variables, $X_1,\ldots,X_n$ gives the probability of every combination of values: $P(X_1,\ldots,X_n)$

positive

	circle	square
red	0.20	0.02
blue	0.02	0.01

negative

	circle	square
red	0.05	0.30
blue	0.20	0.20

- The probability of all possible conjunctions can be calculated by summing the appropriate subset of values from the joint distribution.

$$P(red \wedge circle) = 0.20 + 0.05 = 0.25$$

$$P(red) = 0.20 + 0.02 + 0.05 + 0.3 = 0.57$$

- Therefore, all conditional probabilities can also be calculated.

$$P(positive \mid red \wedge circle) = ?$$

Joint Distribution

- The joint probability distribution for a set of random variables, $X_1,\ldots,X_n$ gives the probability of every combination of values: $P(X_1,\ldots,X_n)$

positive

	circle	square
red	0.20	0.02
blue	0.02	0.01

negative

	circle	square
red	0.05	0.30
blue	0.20	0.20

- The probability of all possible conjunctions can be calculated by summing the appropriate subset of values from the joint distribution.

$$P(red \wedge circle) = 0.20 + 0.05 = 0.25$$

$$P(red) = 0.20 + 0.02 + 0.05 + 0.3 = 0.57$$

- Therefore, all conditional probabilities can also be calculated.

$$P(positive \mid red \wedge circle) = \frac{P(positive \wedge red \wedge circle)}{P(red \wedge circle)} = \frac{0.20}{0.25} = 0.80$$

Bayes Classifiers

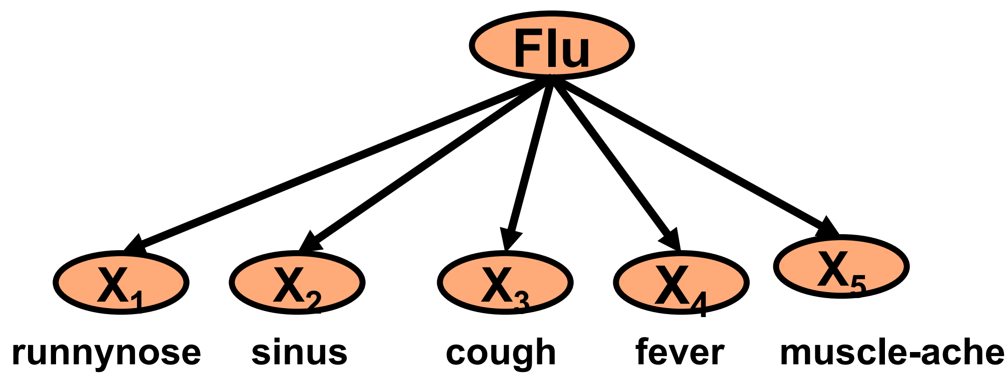
$$c_{MAP} = \underset{c_j \in C}{\operatorname{argmax}} P(x_1, x_2, \ldots, x_n \mid c_j) P(c_j)$$

Bayes Classifiers

$$c_{MAP} = \underset{c_j \in C}{\operatorname{argmax}} P(x_1, x_2, \ldots, x_n \mid c_j) P(c_j)$$

- $P(c_j)$
  - Can be estimated from the frequency of classes in the training examples.
- $P(x_1, x_2, \ldots, x_n | c_j)$
  - $O(|X|^n|C|)$ parameters
  - Could only be estimated if a very, very large number of training examples was available.
  - Need to make some sort of independence assumptions about the features to make learning tractable.

The Naïve Bayes Classifier

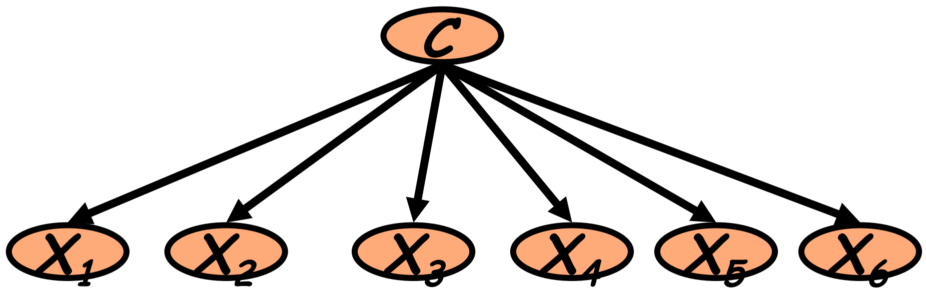


- **Conditional Independence Assumption:** attributes are independent of each other given the class:

$$P(X_1,\ldots,X_5 \mid C) = P(X_1 \mid C) \bullet P(X_2 \mid C) \bullet \cdots \bullet P(X_5 \mid C)$$

- Multi-valued variables: multivariate model
- Binary variables: multivariate Bernoulli model

Learning the Model

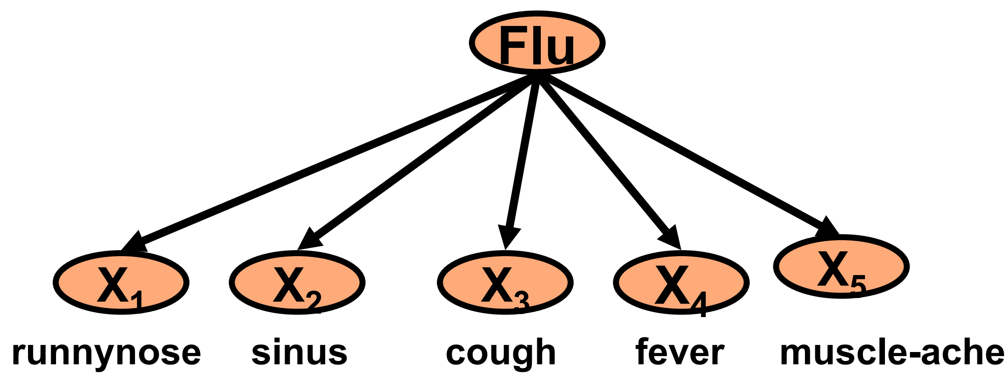


- First attempt: maximum likelihood estimates
  - simply use the frequencies in the data

$$\hat{P}(c_j) = \frac{N(C = c_j)}{N}$$

$$\hat{P}(x_i \mid c_j) = \frac{N(X_i = x_i, C = c_j)}{N(C = c_j)}$$

Problem with Max Likelihood



$$P(X_1,\ldots,X_5 \mid C) = P(X_1 \mid C) \bullet P(X_2 \mid C) \bullet \cdots \bullet P(X_5 \mid C)$$

- What if we have seen no training cases where patient had no flu and muscle aches?

$$\hat{P}(X_5 = t \mid C = nf) = \frac{N(X_5 = t, C = nf)}{N(C = nf)} = 0$$

- Zero probabilities cannot be conditioned away, no matter the other evidence!

$$\ell = \arg\max_c \hat{P}(c) \prod_i \hat{P}(x_i \mid c)$$

Smoothing to Improve Generalization on Test Data

$$\hat{P}(x_i \mid c_j) = \frac{N(X_i = x_i, C = c_j) + 1}{N(C = c_j) + k}$$

# of values of X_i

Underflow Prevention

- Multiplying lots of probabilities, which are between 0 and 1 by definition, can result in floating-point underflow.
- Since log(*xy*) = log(*x*) + log(*y*), it is better to perform all computations by summing logs of probabilities rather than multiplying probabilities.
- Class with highest final un-normalized log probability score is still the most probable.

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} \log P(c_j) + \sum_{i \in positions} \log P(x_i \mid c_j)$$

Probability Estimation Example

Ex	Size	Color	Shape	Class
1	small	red	circle	positive
2	large	red	circle	positive
3	small	red	triangle	negative
4	large	blue	circle	negative

Probability	positive	negative
P(*Y*)		
P(small | *Y*)		
P(medium | *Y*)		
P(large | *Y*)		
P(red | *Y*)		
P(blue | *Y*)		
P(green | *Y*)		
P(square | *Y*)		
P(triangle | *Y*)		
P(circle | *Y*)		

Probability Estimation Example

Ex	Size	Color	Shape	Class
1	small	red	circle	positive
2	large	red	circle	positive
3	small	red	triangle	negative
4	large	blue	circle	negative

Probability	positive	negative
P(*Y*)	0.5	0.5
P(small | *Y*)	0.5	0.5
P(medium | *Y*)	0.0	0.0
P(large | *Y*)	0.5	0.5
P(red | *Y*)	1.0	0.5
P(blue | *Y*)	0.0	0.5
P(green | *Y*)	0.0	0.0
P(square | *Y*)	0.0	0.0
P(triangle | *Y*)	0.0	0.5
P(circle | *Y*)	1.0	0.5

Naïve Bayes Example

Probability	positive	negative
P(*Y*)	0.5	0.5
P(small | *Y*)	0.4	0.4
P(medium | *Y*)	0.1	0.2
P(large | *Y*)	0.5	0.4
P(red | *Y*)	0.9	0.3
P(blue | *Y*)	0.05	0.3
P(green | *Y*)	0.05	0.4
P(square | *Y*)	0.05	0.4
P(triangle | *Y*)	0.05	0.3
P(circle | *Y*)	0.9	0.3

Test Instance:
<medium ,red, circle>

$$c_{MAP} = \arg\max_c \hat{P}(c) \prod_i \hat{P}(x_i \mid c)$$

Naïve Bayes Example

Probability	positive	negative
P(*Y*)	0.5	0.5
P(medium | *Y*)	0.1	0.2
P(red | *Y*)	0.9	0.3
P(circle | *Y*)	0.9	0.3

Test Instance:
<medium ,red, circle>

P(positive | *X*) =?

P(negative | *X*) =?

$$c_{MAP} = \arg\max_c \hat{P}(c) \prod_i \hat{P}(x_i \mid c)$$

Naïve Bayes Example

Probability	positive	negative
P(Y)	0.5	0.5
P(medium | Y)	0.1	0.2
P(red | Y)	0.9	0.3
P(circle | Y)	0.9	0.3

$$c_{MAP} = \operatorname{argmax}_c \hat{P}(c) \prod_i \hat{P}(x_i \mid c)$$

Test Instance:
<medium ,red, circle>

P(positive | X) = P(positive)*P(medium | positive)*P(red | positive)*P(circle | positive) / P(X)
0.5 * 0.1 * 0.9 * 0.9
= 0.0405 / P(X) = 0.0405 / 0.0495 = 0.8181

P(negative | X) = P(negative)*P(medium | negative)*P(red | negative)*P(circle | negative) / P(X)
0.5 * 0.2 * 0.3 * 0.3
= 0.009 / P(X) = 0.009 / 0.0495 = 0.1818

P(positive | X) + P(negative | X) = 0.0405 / P(X) + 0.009 / P(X) = 1

P(X) = (0.0405 + 0.009) = 0.0495

Naïve Bayes for Text Classification

Two models:

- Multivariate Bernoulli Model
- Multinomial Model

Model 1: Multivariate Bernoulli

- One feature X_w for each word in dictionary
- X_w = true (1) in document d if w appears in d
- Naive Bayes assumption:
  - Given the document's topic, appearance of one word in the document tells us nothing about chances that another word appears

- Parameter estimation

$$\hat{P}(X_w = 1 \mid c_j) = ?$$

Model 1: Multivariate Bernoulli

- One feature X_w for each word in dictionary
- X_w = true (1) in document *d* if *w* appears in *d*
- Naive Bayes assumption:
  - Given the document's topic, appearance of one word in the document tells us nothing about chances that another word appears

- Parameter estimation

$$\hat{P}(X_w = 1 \mid c_j) = \text{fraction of documents of topic } c_j \text{ in which word } w \text{ appears}$$

Multinomial Naïve Bayes

- Class conditional unigram language
  - Attributes are text positions, values are words.
  - One feature X_i for each word position in document
    - feature's values are all words in dictionary
  - Value of X_i is the word in position i
  - Naïve Bayes assumption:
    - Given the document's topic, word in one position in the document tells us nothing about words in other positions

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} P(c_j) \prod_i P(x_i \mid c_j)$$

$$= \underset{c_j \in C}{\operatorname{argmax}} P(c_j) P(x_1 = \text{"our"} \mid c_j) \cdots P(x_n = \text{"text"} \mid c_j)$$

  - Too many possibilities!

Multinomial Naive Bayes Classifiers

- Second assumption:
  - Classification is *independent* of the positions of the words (word appearance does not depend on position)

$$P(X_i = w \mid c) = P(X_j = w \mid c)$$

for all positions *i,j*, word *w*, and class *c*

- Use same parameters for each position
- Result is bag of words model (over tokens)

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} P(c_j) \prod_i P(w_i \mid c_j)$$

Multinomial Naïve Bayes for Text

- Modeled as generating a bag of words for a document in a given category by repeatedly sampling with replacement from a vocabulary $V = \{w_1, w_2, \ldots w_m\}$ based on the probabilities $P(w_j \mid c_i)$.

- Smooth probability estimates with Laplace *m*-estimates assuming a uniform distribution over all words ($p = 1/|V|$) and $m = |V|$

Naïve Bayes Classification

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} P(c_j) \prod_i P(x_i \mid c_j)$$

Parameter Estimation

- Multivariate Bernoulli model:

$$\hat{P}(X_w = 1 \mid c_j) = \text{fraction of documents of topic } c_j \text{ in which word } w \text{ appears}$$

- Multinomial model:

$$\hat{P}(X_i = w \mid c_j) = \text{fraction of times in which word } w \text{ appears across all documents of topic } c_j$$

  - Can create a mega-document for topic *j* by concatenating all documents in this topic
  - Use frequency of *w* in mega-document

Classification

- Multinomial vs Multivariate Bernoulli?
- Multinomial model is almost always more effective in text applications!

Naïve Bayes - Spam Assassin

- Naïve Bayes has found a home in spam filtering
  - Paul Graham's *A Plan for Spam*
    - A mutant with more mutant offspring...
  - Widely used in spam filters
    - Classic Naive Bayes superior when appropriately used
      - According to David D. Lewis
  - But also many other things: black hole lists, etc.

- Many email topic filters also use NB classifiers

Naive Bayes is Not So Naive

- Naïve Bayes: First and Second place in KDD-CUP 97 competition, among 16 (then) state of the art algorithms

  Goal: Financial services industry direct mail response prediction model: Predict if the recipient of mail will actually respond to the advertisement – 750,000 records.

- Robust to Irrelevant Features

  Irrelevant Features cancel each other without affecting results

- Very good in domains with many equally important features
- A good baseline for text classification!
- Very Fast: Learning with one pass of counting over the data; testing linear in the number of attributes, and document collection size
- Low Storage requirements