Bayesian Learning Method

Nguyen Van Vinh
UET-Hanoi VNU

Content

- Bayesian Learning (NB) method
- Examples for NB
- Application (Text Classification, Spam Mail)

Introduction

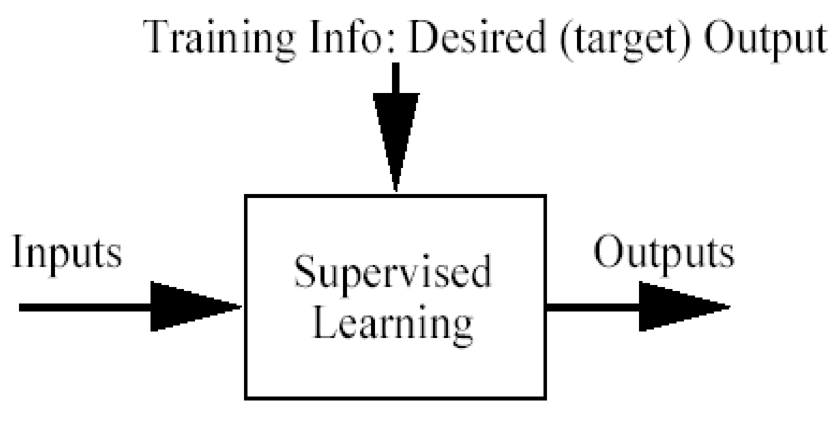
Thomas Bayes (c. 1702 – 17 April 1761) was a British mathematician and Presbyterian minister. (wikipedia)

Today we learn:

- Bayesian <u>classification</u>
  - E.g. How to decide if a patient is ill or healthy, based on
    - A probabilistic model of the observed data
    - Prior knowledge

Classification problem

- Training data: examples of the form (d,h(d))
  - where d are the data objects to classify (inputs)
  - and h(d) are the correct class info for d, $h(d) \in \{1, \ldots K\}$
- Goal: given d_{new}, provide $h(d_{new})$



Error = (target output - actual output)

Why Bayesian?

- Provides practical learning algorithms
  - E.g. Naïve Bayes
- Prior knowledge and observed data can be combined
- It is a generative (model based) approach, which offers a useful conceptual framework
  - E.g. sequences could also be classified, based on a probabilistic model specification
  - Any kind of objects can be classified, based on a probabilistic model specification

Bayes' Rule

$$p(h \mid d) = \frac{P(d \mid h)P(h)}{P(d)}$$

Understanding Bayes' rule

d = data

h = hypothesis (model)

- rearranging

$$p(h \mid d)P(d) = P(d \mid h)P(h)$$

$$P(d,h) = P(d,h)$$

the same joint probability

on both sides

Who is who in Bayes' rule

$P(h)$: prior belief (probability of hypothesis h before seeing any data)

$P(d \mid h)$: likelihood (probability of the data if the hypothesis h is true)

$P(d) = \sum_h P(d \mid h)P(h)$: data evidence (marginal probability of the data)

$P(h \mid d)$: posterior (probability of hypothesis h after having seen the data d)

Does patient have cancer or not?

- A patient takes a lab test and the result comes back positive. It is known that the test returns a correct positive result in only 98% of the cases and a correct negative result in only 97% of the cases. Furthermore, only 0.008 of the entire population has this disease.

  1. What is the probability that this patient has cancer?
  2. What is the probability that he does not have cancer?
  3. What is the diagnosis?

$$\left.\begin{array}{l} hypothesis1:'cancer' \\ hypothesis2:'\neg cancer' \end{array}\right\} \quad hypothesis\ space\ H$$

$$-\,data:'+'$$

$$1.P(cancer\,|\,+) = \frac{P(+\,|\,cancer)P(cancer)}{P(+)} = \frac{\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots}{\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots} = \ldots\ldots\ldots$$

$$P(+\,|\,cancer) = 0.98$$

$$P(cancer) = 0.008$$

$$P(+) = P(+\,|\,cancer)P(cancer) + P(+\,|\,\neg cancer)P(\neg cancer)$$

$$= \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$P(+\,|\,\neg cancer) = 0.03$$

$$P(\neg cancer) = \ldots\ldots\ldots$$

$$2.P(\neg cancer\,|\,+) = \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$3.Diagnosis??$$

Choosing Hypotheses

- *Maximum Likelihood* hypothesis:

$$h_{ML} = \arg\max_{h \in H} P(d \mid h)$$

- Generally we want the most probable hypothesis given training data.This is the *maximum a posteriori* hypothesis:
  - Useful observation: it does not depend on the denominator P(d)

$$h_{MAP} = \arg\max_{h \in H} P(h \mid d)$$

Now we compute the diagnosis

- To find the Maximum Likelihood hypothesis, we evaluate P(d|h) for the data d, which is the positive lab test and chose the hypothesis (diagnosis) that maximises it:

$$P(+ \mid cancer) = \ldots\ldots\ldots\ldots$$

$$P(+ \mid \neg cancer) = \ldots\ldots\ldots\ldots.$$

$$\Rightarrow Diagnosis : h_{ML} = \ldots\ldots\ldots\ldots\ldots\ldots$$

- To find the Maximum A Posteriori (MAP) hypothesis, we evaluate P(d|h)P(h) for the data d, which is the positive lab test and chose the hypothesis (diagnosis) that maximises it. This is the same as choosing the hypotheses gives the higher posterior probability.

$$P(+ \mid cancer)P(cancer) = \ldots\ldots\ldots\ldots\ldots\ldots$$

$$P(+ \mid \neg cancer)P(\neg cancer) = \ldots\ldots\ldots\ldots.$$

$$\Rightarrow Diagnosis : h_{MAP} = \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

Bayesian decision theory

- Let x be the value predicted by the agent and x* be the true value of X.
- The agent has a **loss function**, which is 0 if x = x* and 1 otherwise
- Expected loss for predicting x:

$$\sum_{x^*} L(x, x^*) P(x^* \mid e)$$

- What is the estimate of X that minimizes the expected loss?
  - The one that has the greatest posterior probability P(x|e)
  - This is called the Maximum a Posteriori (MAP) decision

MAP decision

- Value x of X that has the highest posterior probability given the evidence E = e:

$$x^* = \arg\max_x P(X = x \mid E = e) = \frac{P(E = e \mid X = x) P(X = x)}{P(E = e)}$$

$$\propto \arg\max_x P(E = e \mid X = x) P(X = x)$$

$$\underbrace{P(x \mid e)}_{\text{posterior}} \propto \underbrace{P(e \mid x)}_{\text{likelihood}} \underbrace{P(x)}_{\text{prior}}$$

- Maximum likelihood (ML) decision:

$$x^* = \arg\max_x P(e \mid x)$$

Naïve Bayes Classifier

- What can we do if our data *d* has several attributes?
- Naïve Bayes assumption: Attributes that describe data instances are conditionally independent given the classification hypothesis

$$P(\mathbf{d} \mid h) = P(a_1, \ldots, a_T \mid h) = \prod_t P(a_t \mid h)$$

  - it is a simplifying assumption, obviously it may be violated in reality
  - in spite of that, it works well in practice
- The Bayesian classifier that uses the Naïve Bayes assumption and computes the MAP hypothesis is called Naïve Bayes classifier
- One of the most practical learning methods
- Successful applications:
  - Medical Diagnosis
  - Text classification

Example. 'Play Tennis' data

Day	Outlook	Temperature	Humidity	Wind	Play Tennis
Day1	Sunny	Hot	High	Weak	*No*
Day2	Sunny	Hot	High	Strong	*No*
Day3	Overcast	Hot	High	Weak	Yes
Day4	Rain	Mild	High	Weak	Yes
Day5	Rain	Cool	Normal	Weak	Yes
Day6	Rain	Cool	Normal	Strong	*No*
Day7	Overcast	Cool	Normal	Strong	Yes
Day8	Sunny	Mild	High	Weak	*No*
Day9	Sunny	Cool	Normal	Weak	Yes
Day10	Rain	Mild	Normal	Weak	Yes
Day11	Sunny	Mild	Normal	Strong	Yes
Day12	Overcast	Mild	High	Strong	Yes
Day13	Overcast	Hot	Normal	Weak	Yes
Day14	Rain	Mild	High	Strong	*No*

Naïve Bayes solution

Classify any new datum instance $\mathbf{x}=(a_1,\ldots a_T)$ *as:*

$$h_{Naive\,Bayes} = \arg\max_h P(h)P(\mathbf{x}\mid h) = \arg\max_h P(h)\prod_t P(a_t \mid h)$$

- To do this based on training examples, we need to estimate the parameters from the training examples:
  - For each target value (hypothesis) *h*

$$\hat{P}(h) := \text{estimate } P(h)$$

  - For each attribute value a_t of each datum instance

$$\hat{P}(a_t \mid h) := \text{estimate } P(a_t \mid h)$$

Based on the examples in the table, classify the following datum ***x***:

x=(Outl=Sunny, Temp=Cool, Hum=High, Wind=strong)

- That means: Play tennis or not?

$$h_{NB} = \arg\max_{h\in[yes,no]} P(h)P(\mathbf{x} \mid h) = \arg\max_{h\in[yes,no]} P(h)\prod_t P(a_t \mid h)$$

$$= \arg\max_{h\in[yes,no]} P(h)P(Outlook = sunny \mid h)P(Temp = cool \mid h)P(Humidity = high \mid h)P(Wind = strong \mid h)$$

- Working:

$$P(PlayTennis = yes) = 9/14 = 0.64$$
$$P(PlayTennis = no) = 5/14 = 0.36$$
$$P(Wind = strong \mid PlayTennis = yes) = 3/9 = 0.33$$
$$P(Wind = strong \mid PlayTennis = no) = 3/5 = 0.60$$
$$etc.$$
$$P(yes)P(sunny \mid yes)P(cool \mid yes)P(high \mid yes)P(strong \mid yes) = 0.0053$$
$$P(no)P(sunny \mid no)P(cool \mid no)P(high \mid no)P(strong \mid no) = \mathbf{0.0206}$$
$$\Rightarrow answer : PlayTennis(x) = no$$

Example: Training Dataset

Class:
C1:buys_computer = ‘yes’
C2:buys_computer = ‘no’

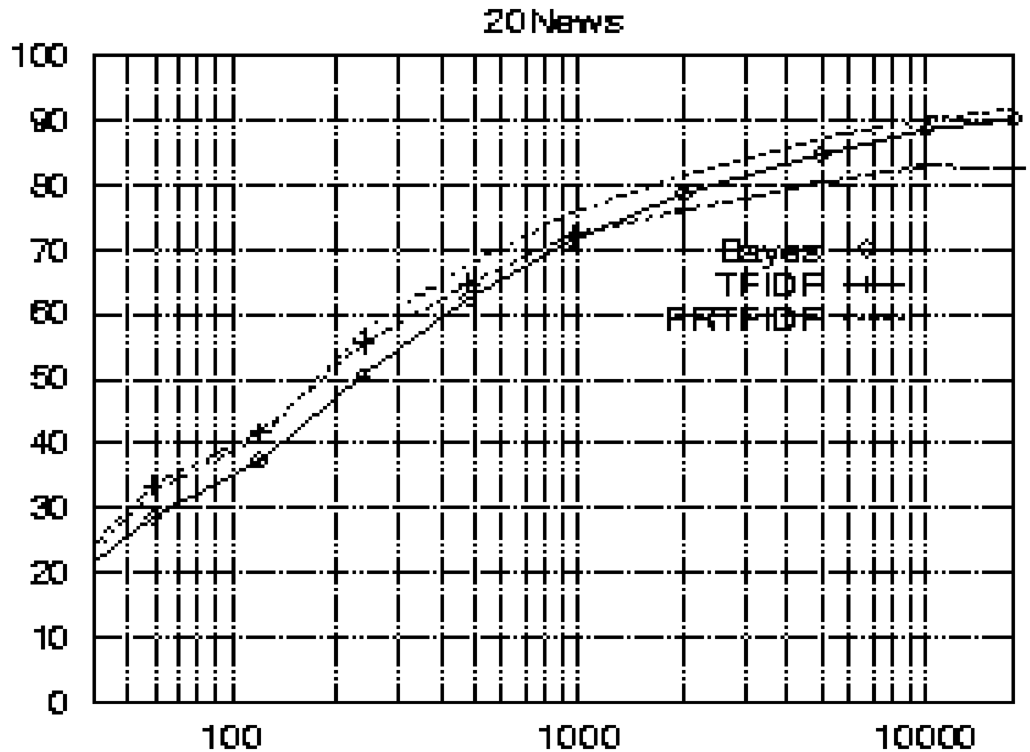
Data sample
X = (**age <=30,**
Income = medium,
Student = yes
Credit_rating = Fair)

age	income	student	credit_rating	_comp
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Learning to classify text

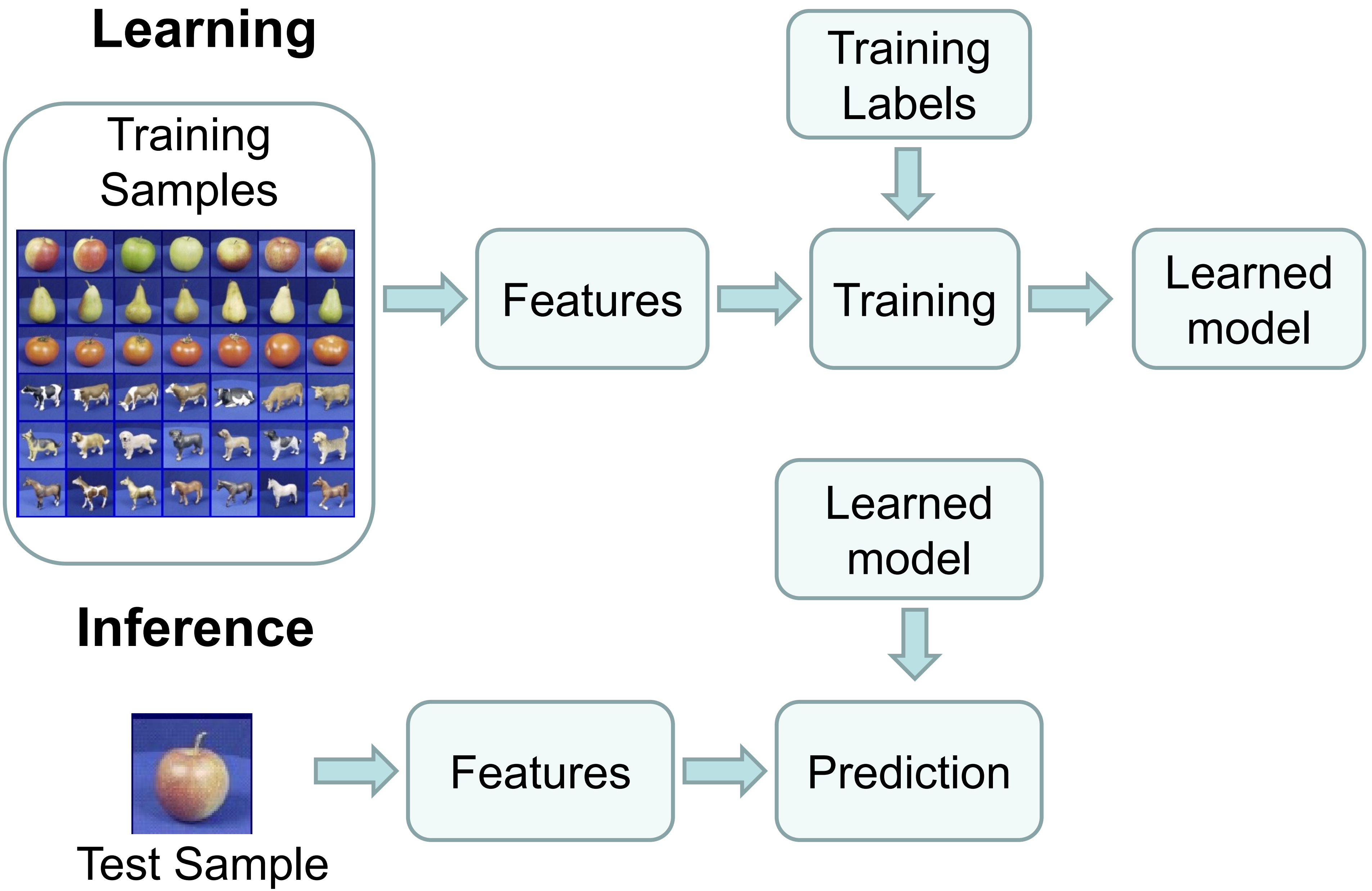
- Learn from examples which articles are of interest
- The attributes are the words
- Observe the Naïve Bayes assumption just means that we have a random sequence model within each class!
- NB classifiers are one of the most effective for this task
- Resources for those interested:
  - Tom Mitchell: Machine Learning (book) Chapter 6.

Results on a benchmark text corpus



Accuracy vs. Training set size (1/3 withheld for test)

Learning and inference pipeline



Features

- A measurable variable that is (rather, should be) distinctive of something we want to model.
- We usually choose features that are useful to **identify** something, i.e., to do **classification**
  - **Ex:** Cô gái đó **rất đẹp** trong bữa tiệc hôm đó.
- We often need **several** features to adequately model something – *but not too many*!

Feature vectors

- **Values** for **several** features of an **observation** can be put into a single **vector**

# proper nouns	# 1st person pronouns	# commas
2	0	0
5	0	0
0	1	1

Feature vectors

- Features should be useful in **discriminating** between categories.

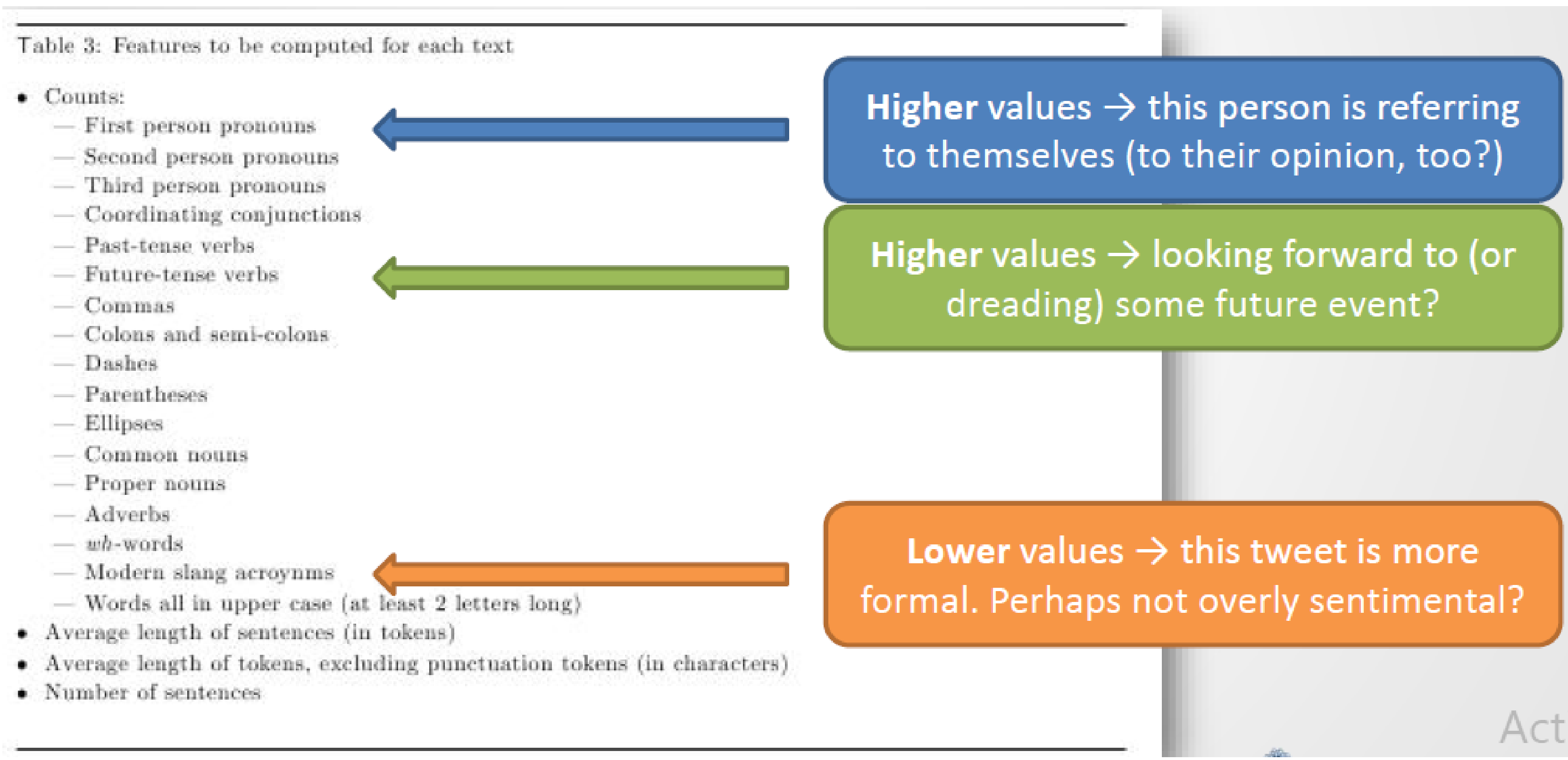


Table 3: Features to be computed for each text

- Counts:
  - First person pronouns
  - Second person pronouns
  - Third person pronouns
  - Coordinating conjunctions
  - Past-tense verbs
  - Future-tense verbs
  - Commas
  - Colons and semi-colons
  - Dashes
  - Parentheses
  - Ellipses
  - Common nouns
  - Proper nouns
  - Adverbs
  - *wh*-words
  - Modern slang acroynms
  - Words all in upper case (at least 2 letters long)
- Average length of sentences (in tokens)
- Average length of tokens, excluding punctuation tokens (in characters)
- Number of sentences

Feature Representation

this movie was great! would watch again Positive

- Convert this example to a vector using *bag-of-words features*

[contains *the*]	[contains *a*]	[contains *was*]	[contains *movie*]	[contains *film*] ...	
position 0	position 1	position 2	position 3	position 4	
$f(x)$ = [0	0	1	1	0	...

- Very large vector space (size of vocabulary), sparse features
- Requires *indexing* the features (mapping them to axes)
- More sophisticated feature mappings possible (m-idf), as well as lots of other features: character n-grams, parts of speech, …

Case study:
Text document classification

- **MAP decision:** assign a document to the class with the highest posterior P(class | document)

- Example: spam classification
  – Classify a message as spam if P(spam | message) > P(¬spam | message)

Dear Sir.

First, I must solicit your confidence in this transaction, this is by virture of its nature as being utterly confidencial and top secret. …

TO BE REMOVED FROM FUTURE MAILINGS, SIMPLY REPLY TO THIS MESSAGE AND PUT "REMOVE" IN THE SUBJECT.

99 MILLION EMAIL ADDRESSES
FOR ONLY $99

Ok, Iknow this is blatantly OT but I'm beginning to go insane. Had an old Dell Dimension XPS sitting in the corner and decided to put it to use, I know it was working pre being stuck in the corner, but when I plugged it in, hit the power nothing happened.

Case study: Text document classification

- **MAP decision:** assign a document to the class with the highest posterior P(class | document)
- We have P(class | document) $\propto$ P(document | class)P(class)
- To enable classification, we need to be able to estimate the likelihoods P(document | class) for all classes and priors P(class)

Naïve Bayes Representation

- Goal: estimate likelihoods P(document | class) and priors P(class)
- Likelihood: ***bag of words*** representation
  - The document is a sequence of words $(w_1, \ldots, w_n)$
  - The order of the words in the document is not important
  - Each word is conditionally independent of the others given document class

Dear Sir.

First, I must solicit your confidence in this transaction, this is by virture of its nature as being utterly confidencial and top secret. …

TO BE REMOVED FROM FUTURE MAILINGS, SIMPLY REPLY TO THIS MESSAGE AND PUT "REMOVE" IN THE SUBJECT.

99 MILLION EMAIL ADDRESSES FOR ONLY $99

Ok, Iknow this is blatantly OT but I'm beginning to go insane. Had an old Dell Dimension XPS sitting in the corner and decided to put it to use, I know it was working pre being stuck in the corner, but when I plugged it in, hit the power nothing happened.

Naïve Bayes Representation

- Goal: estimate likelihoods P(document | class) and priors P(class)
- Likelihood: ***bag of words*** representation
  - The document is a sequence of words $(w_1, \ldots, w_n)$
  - The order of the words in the document is not important
  - Each word is conditionally independent of the others given document class

$$P(document \mid class) = P(w_1, \ldots, w_n \mid class) = \prod_{i=1}^{n} P(w_i \mid class)$$

Bag of words illustration

US Presidential Speeches Tag Cloud
http://chir.ag/projects/preztags/

Bag of words illustration

US Presidential Speeches Tag Cloud
http://chir.ag/projects/preztags/

Bag of words illustration

US Presidential Speeches Tag Cloud
http://chir.ag/projects/preztags/

Naïve Bayes Representation

- Goal: estimate likelihoods P(document | class) and P(class)
- Likelihood: ***bag of words*** representation
  - The document is a sequence of words $(w_1, \ldots, w_n)$
  - The order of the words in the document is not important
  - Each word is conditionally independent of the others given document class

$$P(document \mid class) = P(w_1, \ldots, w_n \mid class) = \prod_{i=1}^{n} P(w_i \mid class)$$

  - Thus, the problem is reduced to estimating marginal likelihoods of individual words $P(w_i \mid class)$

Parameter estimation

- Model parameters: feature likelihoods P(word | class) and priors P(class)
  - How do we obtain the values of these parameters?

prior

spam:	0.33
¬spam:	0.67

P(word | spam)

the :	0.0156
to :	0.0153
and :	0.0115
of :	0.0095
you :	0.0093
a :	0.0086
with:	0.0080
from:	0.0075
...	

P(word | ¬spam)

the :	0.0210
to :	0.0133
of :	0.0119
2002:	0.0110
with:	0.0108
from:	0.0107
and :	0.0105
a :	0.0100
...	

Parameter estimation

- Model parameters: feature likelihoods P(word | class) and priors P(class)
  - How do we obtain the values of these parameters?
  - Need *training set* of labeled samples from both classes

$$P(\text{word} \mid \text{class}) = \frac{\text{\# of occurrences of this word in docs from this class}}{\text{total \# of words in docs from this class}}$$

  - This is the *maximum likelihood* (ML) estimate, or estimate that maximizes the likelihood of the training data:

$$\prod_{d=1}^{D} \prod_{i=1}^{n_d} P(w_{d,i} \mid class_{d,i})$$

d: index of training document, *i*: index of a word

Parameter estimation

- Parameter estimate:

$$P(\text{word} \mid \text{class}) = \frac{\text{\# of occurrences of this word in docs from this class}}{\text{total \# of words in docs from this class}}$$

- Parameter smoothing: dealing with words that were never seen or seen too few times
  - **Laplacian smoothing:** pretend you have seen every vocabulary word one more time than you actually did

$$P(\text{word} \mid \text{class}) = \frac{\text{\# of occurrences of this word in docs from this class} + 1}{\text{total \# of words in docs from this class} + V}$$

(V: total number of unique words)

Summary: Naïve Bayes for Document Classification

- Naïve Bayes model: assign the document to the class with the highest posterior

$$P(class \mid document) \propto P(class) \prod_{i=1}^{n} P(w_i \mid class)$$

- Model parameters:

prior

$$P(\text{class}_1)$$
$$\ldots$$
$$P(\text{class}_K)$$

Likelihood of class 1

$$P(w_1 \mid \text{class}_1)$$
$$P(w_2 \mid \text{class}_1)$$
$$\ldots$$
$$P(w_n \mid \text{class}_1)$$

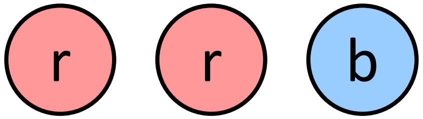
$\ldots$

Likelihood of class K

$$P(w_1 \mid \text{class}_K)$$
$$P(w_2 \mid \text{class}_K)$$
$$\ldots$$
$$P(w_n \mid \text{class}_K)$$

Laplace Smoothing

- Laplace's estimate:
  - Pretend you saw every outcome once more than you actually did



$$P_{ML}(X) =$$

$$P_{LAP}(x) = \frac{c(x)+1}{\sum_x [c(x)+1]}$$

$$= \frac{c(x)+1}{N+|X|}$$

$$P_{LAP}(X) =$$

Laplace Smoothing

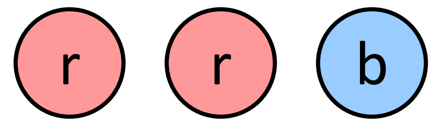
- Laplace's estimate (extended):
  - Pretend you saw every outcome k extra times

$$P_{LAP,k}(x) = \frac{c(x) + k}{N + k|X|}$$

  - What's Laplace with k = 0?
  - k is the strength of the prior

- Laplace for conditionals:
  - Smooth each condition independently:

$$P_{LAP,k}(x|y) = \frac{c(x,y) + k}{c(y) + k|X|}$$



$$P_{LAP,0}(X) =$$

$$P_{LAP,1}(X) =$$

$$P_{LAP,100}(X) =$$

Summarization

- Bayes' rule can be turned into a classifier
- Maximum A Posteriori (MAP) hypothesis estimation incorporates prior knowledge; Max Likelihood doesn't
- Naive Bayes Classifier is a simple but effective Bayesian classifier for vector data (i.e. data with several attributes) that assumes that attributes are independent given the class.
- Bayesian classification is a generative approach to classification

Reference

- Slides of ML Course, University of Birmingham
- Textbook reading (contains details about using Naïve Bayes for text classification):

  Tom Mitchell, Machine Learning (book), Chapter 6.
- Software: NB for classifying text:

  http://www-2.cs.cmu.edu/afs/cs/project/theo-11/www/naive-bayes.html

  https://julianstier.com/posts/2021/01/text-classification-with-naive-bayes-in-numpy/
- Useful reading for those interested to learn more about NB classification, beyond the scope of this module:

  http://www-2.cs.cmu.edu/~tom/NewChapters.html