Naive Bayes
Examples

The Naïve Bayes Algorithm:

Likelihood

Class Prior Probability

$$P(c \mid x) = \frac{P(x \mid c)\,P(c)}{P(x)}$$

Posterior Probability

Evidence
(Normalization)

$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

Bayes Theorem provides a way of calculating the poster probability: P(c|x).
Naïve Bayes classifier assumes the effect of predictor x is **independent** of the values of other predictors. This assumption is called conditional independence.

Naïve Bayes:
Example : SPAM vs. HAM

D1	send us your password	SPAM
D2	send us your review	HAM
D3	review your password	HAM
D4	review us	SPAM
D5	send your password	SPAM
D6	send us your account	SPAM

New Document.. Spam or Ham?

D7	review us	?

Naïve Bayes:
Example : SPAM vs. HAM

SPAM	HAM	
2/4	1/2	password
1/4	2/2	review
3/4	1/2	send
3/4	1/2	us
3/4	1/2	your
1/4	0/2	account

P(review us | spam) = P(0,1,0,1,0,0 | spam) =
(1-2/4)(1/4)(1-3/4)(3/4)(1-3/4)(1-1/4) = 0.00439

P(review us | ham) = P(0,1,0,1,0,0 | ham) =
(1-1/2)*1*(1-1/2)*(1/2)*(1-1/2)*(1-0) = 0.0625

P(ham| review us) = ((0.625)* 1/3) / (0.0625* 1/3 + 0.0044* 2/3) =
0.87

Naïve Bayes: How do we handle words with zero probabilities?

- Method 1: Additive smoothing
  - Add a constant δ to the counts of each word

Counts of w in d

$$p(w \mid d) = \frac{c(w,d)+1}{|d|+|V|}$$

"Add one", Laplace smoothing

Vocabulary size

Length of d (total counts)

For the previous example : account

Previous: P (account| ham) = 0 /2

Now : P(account | ham) = 0 + 1 /(2 + 6)

Naïve Bayes: The Multinomial Approach

	Document	Class
1	Hamburger NYC Hamburger	A
2	Hamburger Hamburger Texas	A
3	Hamburger Cheeseburger	A
4	Montreal Toronto Hamburger	C
5	**Hamburger Hamburger Hamburger Toronto Montreal**	**??**

1) Develop our likelihoods for 'seeing' each word in given the class:

P(Hamburger| A) = (6 + 1) / (8 + 6) =

smoothing

times 'hamburger' appeared in 'A' Docs

of unique words in all docs

# of words in 'A' docs

Naïve Bayes: The Multinomial Approach

	Document	Class
1	Hamburger NYC Hamburger	A
2	Hamburger Hamburger Texas	A
3	Hamburger Cheeseburger	A
4	Montreal Toronto Hamburger	C
5	**Hamburger Hamburger Hamburger Toronto Montreal**	**??**

1) Develop our likelihoods for ‘seeing’ each word in given the class:

P (Hamburger | A) = (5 + 1) / (8 + 6) = 3/7
P(Toronto | A) = (0 + 1) / (8 + 6) = 1/14
P(Montreal | A) = (0 + 1) / (8 + 6) = 1/14

P(Hamburger | C) = (1+ 1) / (3+6) = 2 /9
P (Toronto | C) = (1 + 1) /(3+6) = 2/ 9
P(Montreal | C) = (1 + 1) / (3+6) = 2/9

P (A | d5) ~ (3/4)*((1/14)^2)(3/7)^3 = 0.0003
P (C | d5) ~ (1/4)*(2/9)^3*(2/9)(2/9)=0.0001

Naïve Bayes: The Bernouli Approach

	Document	Class
1	Hamburger NYC Hamburger	A
2	Hamburger Hamburger Texas	A
3	Hamburger Cheeseburger	A
4	Montreal Toronto Hamburger	C
5	**Hamburger Hamburger Hamburger Toronto Montreal**	**??**

1) Develop our likelihoods for ‘seeing’ each word in given the class:

P(Hamburger| A) = (3+1) / (3+ 2)   = 4/5

Constant = 2

of ‘A’ docs

smoothing

of ‘A’ docs where ‘Hamburger’ appears

Naïve Bayes: The Bernouli Approach

	Document	Class
1	Hamburger NYC Hamburger	A
2	Hamburger Hamburger Texas	A
3	Hamburger Cheeseburger	A
4	Montreal Toronto Hamburger	C
5	**Hamburger Hamburger Hamburger Toronto Montreal**	**??**

1) Develop our likelihoods for 'seeing' each word in given the class:

P(Hamburger | A) = (3+1)/(3+2) = 4/5

P (NYC | A) =P(Texas| A) = P(Cheeseburger| A) = (1+1)/5 = 2/5

P(Montreal| A) = P(Toronto| A) = (0+1)/5

P(Hamburger | C) =P(Toronto| C) =P(Montreal | C) (1+1)/(1+2) = 2/3

P(NYC | C) = P(Texas | C) =P(Cheeseburger | C) = 1/3

P (A | d5) ~(3/4)*(4/5) (1/5)^2 (1-2/5)^3 = .005

P (C | d5) ~ (1/4)*(2/3)^3(1-1/3)^3 = .022

Naïve Bayes

Bernoulli : models the fraction of documents of class C that contain the word 'w' (ignores number of occurrences)

Vs.

Multinomial: models the fraction of *positions* in documents of class C that contain the word 'w' (keeps track of number of occurrences)

But why does Naïve Bayes work so well-
(Considering that it is Naïve) ?

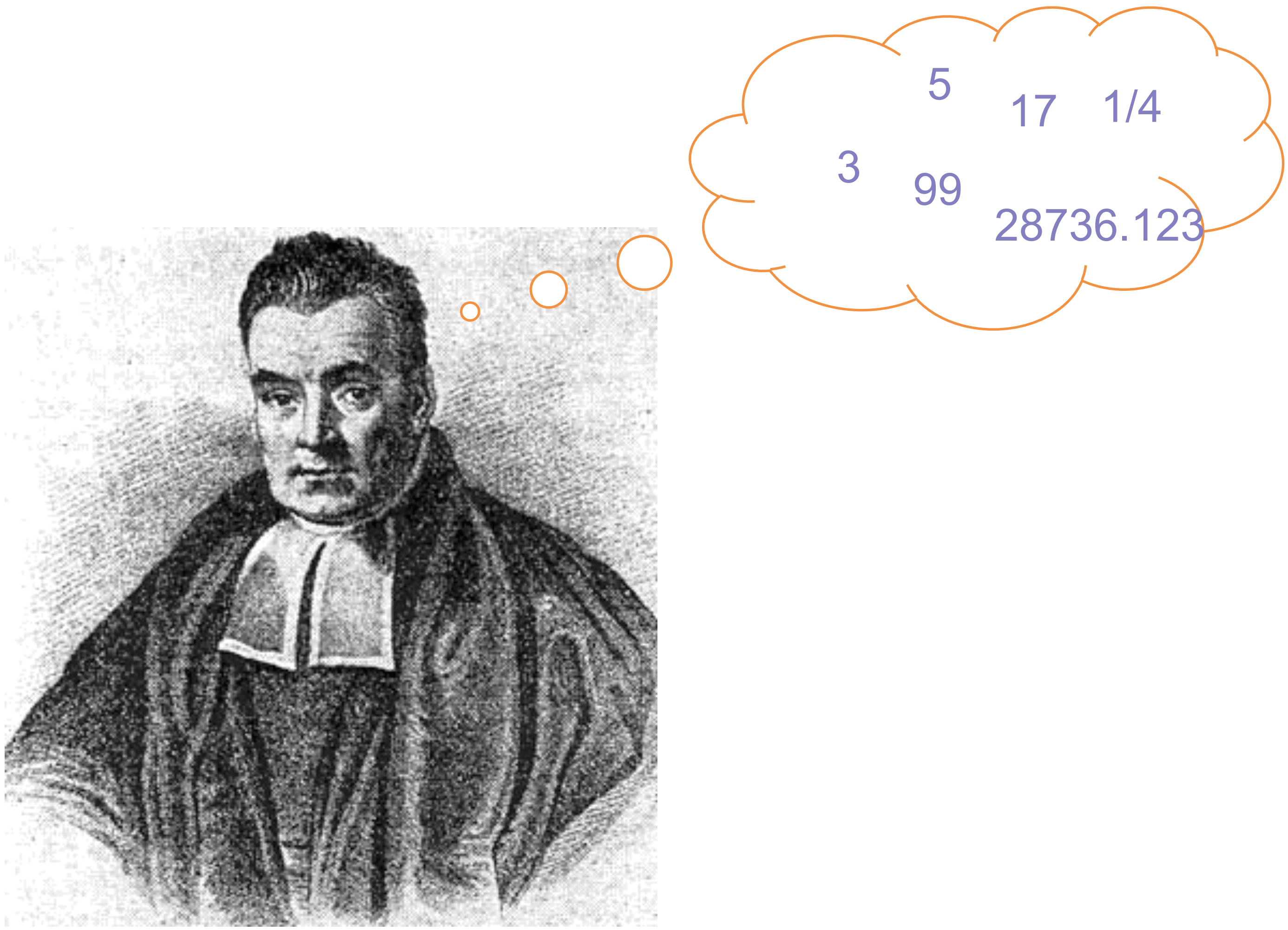
NB chooses among possible classes to find the class with the highest associated probability.

Naiveté doesn’t hurt, because correctness is based on classification, not prediction

Advantages of Naïve Bayes:

- Simple & Fast. Just doing a bunch of counts!
- Will converge quickly. Requires less training data
- Can handle sparse matrices
- Can handle multiple classes well

How about numeric features?



Naïve Bayes: The Gaussian Approach

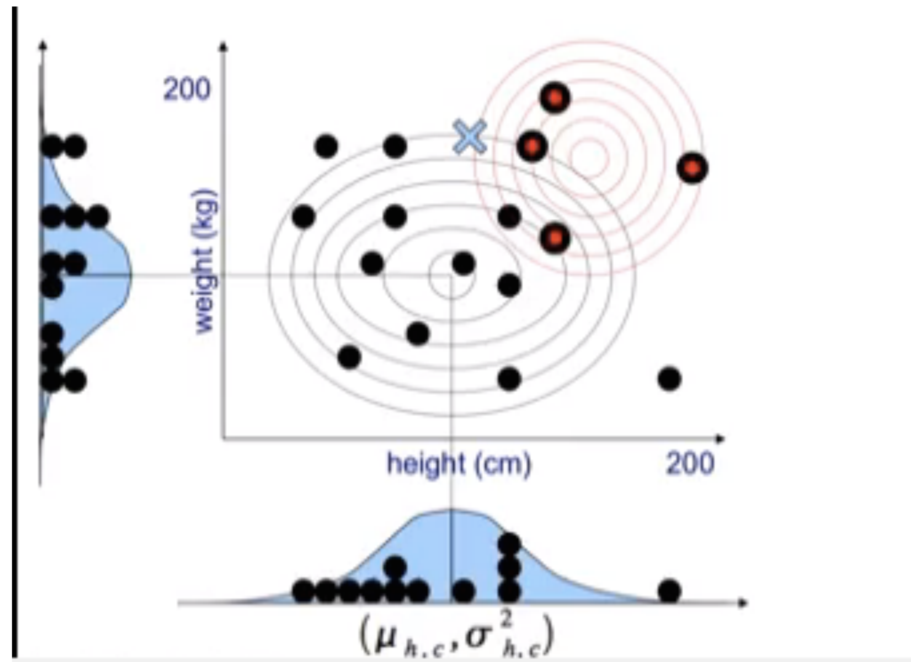
$$p(h_x|c)=\frac{1}{\sqrt{2\pi\sigma^2_{h,c}}}\exp-\frac{1}{2}\left(\frac{(h_x-\mu_{h,c})^2}{\sigma^2_{h,c}}\right)$$

$$p(w_x|c)=\frac{1}{\sqrt{2\pi\sigma^2_{w,c}}}\exp-\frac{1}{2}\left(\frac{(w_x-\mu_{w,c})^2}{\sigma^2_{w,c}}\right)$$

$$p(h_x|a)=\frac{1}{\sqrt{2\pi\sigma^2_{h,a}}}\exp-\frac{1}{2}\left(\frac{(h_x-\mu_{h,a})^2}{\sigma^2_{h,a}}\right)$$

$$p(w_x|a)=\frac{1}{\sqrt{2\pi\sigma^2_{w,a}}}\exp-\frac{1}{2}\left(\frac{(w_x-\mu_{w,a})^2}{\sigma^2_{w,a}}\right)$$

$$P(x|a)=p(h_x|a)\,p(w_x|a)$$



P(a|x) ~ P(x|a)*P(a)