

---

# COMP 472: Artificial Intelligence

## Machine Learning *part 2*

### Naive Bayes Classification

### Application to Spam Filtering *video #3*

- Russell & Norvig: Sections 12.2 to 12.6

# Today

1. Introduction to ML
2. Naive Bayes Classification
  - a. Application to Spam Filtering
3. Decision Trees
4. ( Evaluation
5. Unsupervised Learning )
6. Neural Networks
  - a. Perceptrons
  - b. Multi Layered Neural Networks



# Recall

$$H_{NB} = \underset{H_i}{\operatorname{argmax}} \frac{P(H_i) \times P(E | H_i)}{P(E)} = \underset{H_i}{\operatorname{argmax}} P(H_i) \times P(E | H_i) = \underset{H_i}{\operatorname{argmax}} P(H_i) \times P(\langle a_1, a_2, a_3, \dots, a_n \rangle | H_i) = \underset{H_i}{\operatorname{argmax}} P(H_i) \times \prod_{j=1}^n P(a_j | H_i)$$

$$H_{NB} = \underset{H_i}{\operatorname{argmax}} P(H_i) \times \prod_{j=1}^n P(a_j | H_i)$$

// prior // conditionals

// attributes  
// features  
// piece of evidence

} hypothesis is  
} class  
} category  
} bucket -

# Application of Naive Bayes Classification: Spam Filtering

- Task: classify e-mails (documents) into a pre-defined class

- ex: spam / ham

- ex: sports, recreation, politics, war, economy,...

ex: product review  
positive    neutral    negative

## Given

- training set of documents already classified into the correct category



**SPAM**




**HAM**


# e-mail Representation

- each e-mail is represented by a vector of feature/value:


-  feature = actual words in the e-mail


-  value = number of times that word appears in the e-mail

 email #1 <airplane=0, banana=1, cat=5, duck=4, ..., zoo=0, class=SPAM>

 email #2 <airplane=2, banana=0, cat=0, duck=8, ..., zoo=3, class=SPAM>

...

 email <airplane=1, banana=1, cat=5, duck=8, ..., zoo=3, class=HAM>

 <airplane=1, banana=3, cat=5, duck=0, ..., zoo=6, class=HAM>

10000 words ... 100 000 words  $\rightarrow \mathcal{L}(x)$   
40 spam  
assume  
100 emails  
in  
training  
set

60 ham



Strictly speaking, what this is called a Multinomial Naïve Bayes classifier, because we use the frequency of words, as opposed to just using binary values for the presence/absence of words.

# Naïve Bayes Algorithm

// 1. training

for all classes  $c_i$  // ex. ham or spam  
for all words  $w_j$  in the vocabulary

$$\text{compute } P(w_j | c_i) = \frac{\text{count}(w_j, c_i)}{\sum_j \text{count}(w_j, c_i)}$$

// conditionals.

for all classes  $c_i$

$$\text{compute } P(c_i) = \frac{\text{count}(\text{documents in } c_i)}{\text{count}(\text{all documents})}$$

//  $P(Hi) = 0.6$   
 $P(\text{ham}) = 0.6$   $P(\text{spam}) = 0.4$

// 2. testing a new document  $D$

for all classes  $c_i$  // ex. ham or spam

$$\text{score}(c_i) = P(c_i)$$

for all words  $w_j$  in the  $D$

$$\text{score}(c_i) = \text{score}(c_i) \times P(w_j | c_i)$$

choose  $c^*$  = with the greatest  $\text{score}(c_i)$

	$w_1$	$w_2$	$w_3$	$w_4$	$w_5$	$w_6$
$c_1$ : SPAM	$p(w_1 c_1)$	$p(w_2 c_1)$	$p(w_3 c_1)$	$p(w_4 c_1)$	$p(w_5 c_1)$	$p(w_6 c_1)$
$c_2$ : HAM	$p(w_1 c_2)$	$p(w_2 c_2)$	$p(w_3 c_2)$	$p(w_4 c_2)$	$p(w_5 c_2)$	$p(w_6 c_2)$

# Example 1

vocabulary = { best, book, cheap, sale, trip, meds }

## Dataset

### c1: SPAM

doc1: "cheap meds for sale"

doc2: "click here for the best meds"

doc3: "book your trip"

$$p(\text{spam}) = 3/5$$

$$p(\text{best} | \text{spam}) = 1/7$$

$$p(\text{book} | \text{spam}) = 1/7$$



SPAM



HAM

### c2: HAM

doc4: "cheap book sale, not meds"

doc5: "here is the book for you"

$$p(\text{ham}) = 2/5$$

$$p(\text{best} | \text{ham}) = 0/5$$



## Question:

doc6: "the cheap book"

should it be classified as HAM or SPAM?

$$\epsilon = 5/5$$

# Example 1

a word in an email is

Assume

vocabulary = {best, book, cheap, sale, trip, meds}

If not in <sup>the</sup> vocabulary, ignore word

$|V| = 6$  words  
only  
not realistic.

1. Training:

- conditionals  $|V|$
- $P(\text{best}|\text{SPAM}) = 1/7$
  - $P(\text{book}|\text{SPAM}) = 1/7$
  - $P(\text{cheap}|\text{SPAM}) = 1/7$
  - $P(\text{sale}|\text{SPAM}) = 1/7$
  - $P(\text{trip}|\text{SPAM}) = 1/7$
  - $P(\text{meds}|\text{SPAM}) = 2/7$

$$P(\text{best}|\text{HAM}) = 0/5$$

$$P(\text{book}|\text{HAM}) = 2/5$$

$$P(\text{cheap}|\text{HAM}) = 1/5$$

$$P(\text{sale}|\text{HAM}) = 1/5$$

$$P(\text{trip}|\text{HAM}) = 0/5$$

$$P(\text{meds}|\text{HAM}) = 1/5$$

- priors
- $P(\text{SPAM}) = 3/5$

$$P(\text{HAM}) = 2/5$$

2. Testing: "the cheap book" <sup>trip</sup>

- argmax
- $\text{Score}(\text{HAM}) = P(\text{HAM}) \times P(\text{cheap}|\text{HAM}) \times P(\text{book}|\text{HAM}) \times 0$
  - $\text{Score}(\text{SPAM}) = P(\text{SPAM}) \times P(\text{cheap}|\text{SPAM}) \times P(\text{book}|\text{SPAM}) \times 1/2$



# Be Careful: Smooth Probabilities

- normally:  $P(w_i | c_j) = \frac{(\text{frequency of } w_i \text{ in } c_j)}{\text{total number of words in } c_j}$
- what if we have a  $P(w_i | c_j) = 0 \dots ?$ 
  - ex. the word "dumbo" never appeared in the class SPAM?
  - then  $P(\text{"dumbo"} | \text{SPAM}) = 0$
- so if a text contains the word "dumbo", the class SPAM is completely ruled out !
- to solve this: we assume that every word always appears at least once (or a smaller value)
  - ex: add-1 smoothing:

$$P(w_i | c_j) = \frac{(\text{frequency of } w_i \text{ in } c_j) + 1}{\text{total number of words in } c_j + \text{size of vocabulary}}$$

*original values* (circled around the numerator and denominator)

*for smoothing purposes* (circled around the +1 in the denominator)

# Smoothing - add-1 smoothing

- Assume:

- vocabulary  $V = \{\text{ball, heat, kitchen, referee, stove, the, ...}\}$
- $|V| = 100$

- Training set:

original data set

c1: COOKING	c2: SPORTS
doc <sub>1</sub> : ... stove... kitchen... the... heat	doc <sub>1</sub> : ... ball... heat...
doc <sub>2</sub> : ... kitchen... pasta... stove...	doc <sub>2</sub> : ... the... referee... player...
doc <sub>100000</sub> : ... stove... heat... ball...	doc <sub>75000</sub> : goal... injury ...

$|V|$

ball heat kitchen  
100 extra words

100 extra words

new smoothed values for the conditional probs will be based on  $\boxed{\phantom{00}} + \boxed{\phantom{00}}$

# Be Careful: Use Logs

- if we really do the product of probabilities...

- $\operatorname{argmax}_{c_j} P(c_j) \prod P(w_i | c_j)$
- we soon have numerical underflow...
- ex:  $0.01 \times 0.02 \times 0.05 \times \dots$

- so instead, we add the log of the probs

- $\operatorname{argmax}_{c_j} \log(P(c_j)) + \sum \log(P(w_i | c))$
- ex:  $\log(0.01) + \log(0.02) + \log(0.05) + \dots$

*- use the base (log) that you prefer*

*so we keep  
the same  
ranking*

*= -3 ✓  
-4*

# Example 2

## ■ Training set:

c1: COOKING	c2: SPORTS
doc <sub>1</sub> : ... stove... kitchen... the... heat doc <sub>2</sub> : ... kitchen... pasta... stove... ... doc <sub>100000</sub> : ... stove...heat... ball... <i>Σ words = 500,000</i>	doc <sub>1</sub> : ... ball... heat... doc <sub>2</sub> : ... the... referee... player... ... doc <sub>75000</sub> : goal... injury ... <i>Σ = 300,000 words</i>

## ■ Assume:

- vocabulary  $V = \{\text{ball, heat, kitchen, referee, stove, the, ...}\}$
- $|V| = 100$  *// not realistic*
- 500,000 words in Cooking
- 300,000 words in Sports
- 100,000 docs in Cooking
- 75,000 docs in Sports

# Example 2

## Training - Unsmoothed / Smoothed probs:

assume these numerators

$$\frac{10,000 + 1}{500,000 + 100}$$

conditionals  
| V |  
= 100 rows  
= 100 words

$$P(\text{ball}|\text{COOKING}) =$$

$$\frac{10,000}{500,000}$$

$$\frac{??}{??}$$

$$P(\text{heat}|\text{COOKING}) =$$

$$\frac{255}{500,000}$$

$$\frac{??}{??}$$

$$P(\text{kitchen}|\text{COOKING}) =$$

$$\frac{2,600}{500,000}$$

$$\frac{??}{??}$$

$$P(\text{referee}|\text{COOKING}) =$$

$$\frac{0}{500,000}$$

$$\frac{??}{??}$$

$$P(\text{stove}|\text{COOKING}) =$$

$$\frac{3,600}{500,000}$$

$$\frac{??}{??}$$

$$P(\text{the}|\text{COOKING}) =$$

$$\frac{400,000}{500,000}$$

$$\frac{??}{??}$$

...

$$P(\text{COOKING}) = \frac{10,000}{175,000}$$

$$P(\text{ball}|\text{SPORTS}) =$$

$$\frac{10,000}{300,000}$$

$$\frac{??}{??}$$

$$P(\text{heat}|\text{SPORTS}) =$$

$$\frac{1,8000}{300,000}$$

$$\frac{??}{??}$$

$$P(\text{kitchen}|\text{SPORTS}) =$$

$$\frac{0}{300,000}$$

$$\frac{??}{??}$$

$$P(\text{referee}|\text{SPORTS}) =$$

$$\frac{1,500}{300,000}$$

$$\frac{??}{??}$$

$$P(\text{stove}|\text{SPORTS}) =$$

$$\frac{4}{300,000}$$

$$\frac{??}{??}$$

$$P(\text{the}|\text{SPORTS}) =$$

$$\frac{19,000}{300,000}$$

$$\frac{??}{??}$$

$$P(\text{SPORTS}) = \frac{75,000}{175,000}$$

$$\frac{0 + 1}{500,000 + 100}$$

## Testing: "the referee hit the blue bird"

argmax {

$$\text{Score}(\text{COOKING}) = \log\left(\frac{100,000}{175,000}\right) + \log(P(\text{the}|\text{COOKING})) + \log(P(\text{referee}|\text{COOKING})) + \log(P(\text{hit}|\text{COOKING})) + \log(P(\text{the}|\text{COOKING}))$$

$$\text{Score}(\text{SPORTS}) = \log\left(\frac{75,000}{175,000}\right) + \log(P(\text{the}|\text{SPORTS})) + \log(P(\text{referee}|\text{SPORTS})) + \log(P(\text{hit}|\text{SPORTS})) + \log(P(\text{the}|\text{SPORTS}))$$

assume not in V

# Today

1. Introduction to ML ✓
2. Naïve Bayes Classification ✓
  - a. Application to Spam Filtering ✓ video #3
3. Decision Trees
4. ( Evaluation
5. Unsupervised Learning )
6. Neural Networks
  - a. Perceptrons
  - b. Multi Layered Neural Networks

---

# Up Next

1. Introduction to ML
2. Naive Bayes Classification
  - a. Application to Spam Filtering
3. **Decision Trees** *video #4*
4. ( Evaluation
5. Unsupervised Learning )
6. Neural Networks
  - a. Perceptrons
  - b. Multi Layered Neural Networks