# 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. Naïve 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

$$H_{NB} = \underset{H_i}{\operatorname{argmax}} \frac{P(H_i) \times P(E \mid H_i)}{P(E)} = \underset{H_i}{\operatorname{argmax}}\ P(H_i) \times P(E \mid H_i) = \underset{H_i}{\operatorname{argmax}}\ P(H_i) \times P(< a_1, a_2, a_3, \ldots, a_n > \mid H_i) = \underset{H_i}{\operatorname{argmax}}\ P(H_i) \times \prod_{j=1}^{n} P(a_j \mid H_i)$$

//prior //conditionals

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

// attributes
// features
// piece of evidence

hypothesis
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

email #2

email #1

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

10000 words ... 100 000 words → f(x)

X

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>

40 spam

60 ham

assume 100 emails in training set

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
        compute
```

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

// conditionals

```
for all classes c_i
    compute
```

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

// P(Hi)

P(ham) = 0.6 P(spam) = 0.4

// 2. testing a new document D

```
for all classes c_i  // ex. ham or spam
    score(c_i) = P(c_i)
    for all words w_j in the D
        score(c_i) = score(c_i) x P(w_j | c_i)

choose c* = with the greatest score(c_i)
```

$h_{NB}$

| | $w_1$ | $w_2$ | $w_3$ | $w_4$ | $w_5$ | $w_6$ |
|---|---|---|---|---|---|---|
| c1 : SPAM | $p(w_1 \mid c_1)$ | $p(w_2 \mid c_1)$ | $p(w_3 \mid c_1)$ | $p(w_4 \mid c_1)$ | $p(w_5 \mid c_1)$ | $p(w_6 \mid c_1)$ |
| c2 : HAM | $p(w_1 \mid c_2)$ | $p(w_2 \mid c_2)$ | $p(w_3 \mid c_2)$ | $p(w_4 \mid c_2)$ | $p(w_5 \mid c_2)$ | $p(w_6 \mid c_2)$ |

# Example 1

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

- Dataset

  Training set

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

  - c1: SPAM
    - doc1: "cheap meds for sale"
    - doc2: "click here for the best meds"
    - doc3: "book your trip"

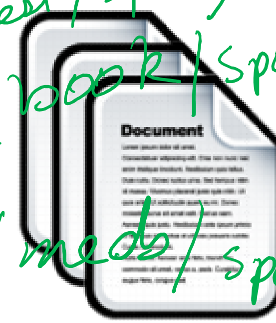
  $/7$

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

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

  $p(\text{meds} \mid \text{spam}) = 2/7$

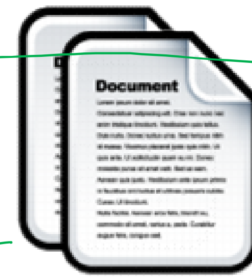
  $\Sigma\ 7/7$



SPAM

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

  - c2: HAM
    - doc4: "cheap book sale, not meds"
    - doc5: "here is the book for you"



HAM

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

  $\Sigma = 5/5$

- Question:
  - doc6: “the cheap book”
  - should it be classified as HAM or SPAM?

?

# Example 1

a word in an email is

Assume

vocabulary = {best, book, cheap, sale, trip, meds} |V| = 6 words only

If not in the vocabulary, ignore word

not realistic.

1. Training:

conditionals |V|

| | |
|---|---|
| P(best\|SPAM) = 1/7 | P(best\|HAM) = 0/5 |
| P(book\|SPAM) = 1/7 | P(book\|HAM) = 2/5 |
| P(cheap\|SPAM) = 1/7 | P(cheap\|HAM) = 1/5 |
| P(sale\|SPAM) = 1/7 | P(sale\|HAM) = 1/5 |
| P(trip\|SPAM) = 1/7 | P(trip\|HAM) = 0/5 |
| P(meds\|SPAM) = 2/7 | P(meds\|HAM) = 1/5 |

priors

| | |
|---|---|
| P(SPAM) = 3/5 | P(HAM) = 2/5 |

2. Testing: "~~the~~ cheap book" trip

argmax

- Score(HAM)= P(HAM) x P(cheap|HAM) x P(book|HAM) × 0
- Score(SPAM)= P(SPAM) x P(cheap|SPAM) x P(book|SPAM) × 1/7

# Be Careful: Smooth Probabilities

- normally: $P(w_i \mid c_j) = \dfrac{(\text{frequency of } w_i \text{ in } c_j)}{\text{total number of words in } c_j}$
- what if we have a $P(w_i|c_j) = 0$...?
  - ex. the word "dumbo" never appeared in the class SPAM?
  - then P("dumbo"| 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:

original values

for smoothing purposes

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

# Smoothing – add-1 smoothing

- Assume:
  - vocabulary V = {ball, heat, kitchen, referee, stove, the, ... }
  - |V| = 100
- Training set:

| c1: COOKING | c2: SPORTS |
|---|---|
| $doc_1$: ... stove... kitchen... the... heat<br>$doc_2$: ... kitchen... pasta... stove...<br>...<br>$doc_{100000}$: ... stove...heat... ball... | $doc_1$: ... ball... heat...<br>$doc_2$: ... the... referee... player...<br>...<br>$doc_{75000}$: goal... injury ... |
| ball heat kitchen<br>+1 +1<br>100 extra words | 100 extra words |

original data set

|V|

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

# Be Careful: Use Logs

- if we really do the product of probabilities…
  - $\text{argmax}_{cj}\ P(c_j) \prod P(w_i|c_j)$
  - we soon have numerical underflow…
  - ex: 0.01 x 0.02 x 0.05 x …
- so instead, we add the log of the probs
  - $\text{argmax}_{cj}\ \log(P(c_j)) + \sum \log(P(w_i|c))$
  - ex: log(0.01) + log(0.02) + log(0.05) + …

so we keep the same ranking

= −3 ✓
= −4

– use the base (log) that you prefer

# Example 2

- Training set:

| c1: COOKING | c2: SPORTS |
|---|---|
| $doc_1$: … stove… kitchen… the… heat<br>$doc_2$: … kitchen… pasta… stove…<br>…<br>$doc_{100000}$: … stove…heat… ball…<br>$\Sigma$ words = 500 000 | $doc_1$: … ball… heat…<br>$doc_2$: … the… referee… player…<br>…<br>$doc_{75000}$: goal… injury …<br>$\Sigma$ = 300 000 words |

- Assume:
  - vocabulary V = {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

assume these numerators

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

- Training – Unsmoothed / Smoothed probs:

conditionals

$|V| = 100$ rows $= 100$ words

- P(ball|COOKING) = $\frac{10{,}000}{500{,}000}$ $\frac{??}{??}$
- P(heat|COOKING) = $\frac{255}{500{,}000}$ $\frac{??}{??}$
- P(kitchen|COOKING) = $\frac{2{,}600}{500{,}000}$ $\frac{??}{??}$
- P(referee|COOKING) = $\frac{0}{500{,}000}$ $\frac{??}{??}$
- P(stove|COOKING) = $\frac{3{,}600}{500{,}000}$ $\frac{??}{??}$
- P(the|COOKING) = $\frac{400{,}000}{500{,}000}$ $\frac{??}{??}$
- …

$\boxed{100000^*}$

- P(ball|SPORTS) = $\frac{10{,}000}{300{,}000}$ $\frac{??}{??}$
- P(heat|SPORTS) = $\frac{1{,}8000}{300{,}000}$ $\frac{??}{??}$
- P(kitchen|SPORTS) = $\frac{0}{300{,}000}$ $\frac{??}{??}$ $\frac{1}{300\,100}$
- P(referee|SPORTS) = $\frac{1{,}500}{300{,}000}$ $\frac{??}{??}$
- P(stove|SPORTS) = $\frac{4}{300{,}000}$ $\frac{??}{??}$ $\frac{5}{300\,100}$
- P(the|SPORTS) = $\frac{19{,}000}{300{,}000}$ $\frac{??}{??}$

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

priors

- P(COOKING) = $\frac{10{,}000}{175{,}000}$
- P(SPORTS) = $\frac{75{,}000}{175{,}000}$

- Testing: "the referee hit the ~~blue bird~~"

argmax

- Score(COOKING)= $\log(\frac{100{,}000}{175{,}000})$ + log(P(the|COOKING)) + log(P(referee|COOKING)) + log(P(hit|COOKING)) + log(P(the|COOKING))
- Score(SPORTS)= $\log(\frac{75{,}000}{175{,}000})$ + log(P(the|SPORTS)) + log(P(referee|SPORTS)) + log(P(hit|SPORTS)) + log(P(the| 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. Naïve 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