Today's Schedule

1. Text classification
2. Bag-of-words models
3. Multinomial document model
4. Bernoulli document model
5. Generative models
6. Zero Probability Problem

Identifying Spam

Spam?

I got your contact information from your country's information directory during my desperate search for someone who can assist me secretly and confidentially in relocating and managing some family fortunes.

How do we represent \( D \)?

- A sequence of words: \( D = (X_1, X_2, \ldots, X_n) \)
- Computational very expensive, difficult to train

Multinomial document model

- \( x = (x_1, \ldots, x_N) \quad x_i \in \mathbb{N} \)
- Ignore the position of the word
- Ignore the order of the word
- Consider the words in pre-defined vocabulary \( V \) (\( D = |V| \))

Bernoulli document model

- \( b = (b_1, \ldots, b_D) \quad b_i \in \{0, 1\} \)

BoW models: Bernoulli vs. Multinomial

Document \( D \): "Congratulations to you as we bring to your notice, the results of the First Category draws of THE HOLLAND CASINO LOTTO PROMO INT. We are happy to inform you that you have emerged a winner under the First Category, which is part of our promotional draws."

For the Multinomial model, the document representation is

\[
 x = (x_1, \ldots, x_N) \quad x_i \in \mathbb{N}
\]

For the Bernoulli model, the document representation is

\[
 b = (b_1, \ldots, b_D) \quad b_i \in \{0, 1\}
\]
Text Classification using Naive Bayes

**Classification with multinomial document model**

Assume a test document \( D \) is given as a sequence of words:
\[ (o_1, o_2, \ldots, o_n), \quad o_i \in V = \{ w_1, \ldots, w_D \} \]

Feature vectors: \( x = (x_1, \ldots, x_D) \) - word frequencies, \( \sum_{i=1}^{D} x_i = n \)

Document likelihood with multinomial distribution:
\[
P(x | C_k) = \frac{n!}{\prod_{i=1}^{D} x_i!} \prod_{i=1}^{D} P(w_i | C_k)^{x_i} \quad \text{NB: } P>1 \quad (P>0)
\]

For classification, we can omit irrelevant term, so that:
\[
P(x | C_k) \propto \prod_{i=1}^{D} P(w_i | C_k)^{x_i} = P(o_1 | C_k) P(o_2 | C_k) \cdots P(o_n | C_k)
\]

**Training of multinomial document model**

Features: \( x = (x_1, \ldots, x_D) \) - word frequencies in a doc.
Training data set:
\[
\begin{align*}
C_1 & : \begin{pmatrix} x_1^{(1)} \cdots x_D^{(1)} \end{pmatrix} = \begin{pmatrix} x_{11} \cdots x_{1D} \end{pmatrix} \\
& \cdots \\
C_N & : \begin{pmatrix} x_1^{(N)} \cdots x_D^{(N)} \end{pmatrix} = \begin{pmatrix} x_{11} \cdots x_{1D} \end{pmatrix}
\end{align*}
\]

**Multinomial doc. model – example**

See Note 7!

**Classification with Bernoulli document model**

A test document \( D \) with feature vector \( b = (b_1, \ldots, b_D) \)

Document likelihood with Bernoulli distribution:
\[
P(b | C_k) = \prod_{i=1}^{D} P(b_i | C_k) = \prod_{i=1}^{D} (b_i P(w_i | C_k) + (1-b_i)(1-P(w_i | C_k))]
\]

\[
P(w_i | C_k) = \frac{n_k(w_i)}{N_k} \quad \text{(fraction of class } k \text{ docs. with word } w_i)
\]

For classification, we can omit irrelevant term, so that:
\[
P(b | C_k) \propto \prod_{i=1}^{D} P(w_i | C_k)^{b_i} = P(b_1 | C_k) P(b_2 | C_k) \cdots P(b_D | C_k)
\]

**Training of Bernoulli document model**

Features: \( b = (b_1, \ldots, b_D) \) - \( D = |V| \), i.e. vocabulary binary vector of word occurrences in a document
Training data set:
\[
\begin{align*}
C_1 & : \begin{pmatrix} b_1^{(1)} \cdots b_D^{(1)} \end{pmatrix} = \begin{pmatrix} b_{11} \cdots b_{1D} \end{pmatrix} \\
& \cdots \\
C_N & : \begin{pmatrix} b_1^{(N)} \cdots b_D^{(N)} \end{pmatrix} = \begin{pmatrix} b_{11} \cdots b_{1D} \end{pmatrix}
\end{align*}
\]

**Bernoulli doc. model – example**

Classify documents as Sports (S) or Informatics (I)

**Vocabulary V:**
- \( w_0 = \text{goal} \)
- \( w_2 = \text{tutor} \)
- \( w_3 = \text{variance} \)
- \( w_4 = \text{speed} \)
- \( w_5 = \text{drink} \)
- \( w_6 = \text{defence} \)
- \( w_7 = \text{performance} \)
- \( w_8 = \text{field} \)

\( D = |V| = 8 \)

\[
\begin{align*}
\text{BSport} & : \begin{pmatrix} 0 \cdots 1 \cdots 0 \cdots 0 \cdots \end{pmatrix} \\
\text{Binf} & : \begin{pmatrix} 1 \cdots 0 \cdots 0 \cdots 0 \cdots \end{pmatrix}
\end{align*}
\]

**Bernoulli doc. model – example (cont.)**

**Summary of the document models**

<table>
<thead>
<tr>
<th>Class</th>
<th>Multinomial doc. model</th>
<th>Bernoulli doc. model</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P(w_i</td>
<td>C_k) )</td>
<td>( n_k(w_i)/N_k )</td>
</tr>
<tr>
<td>( P(w_i</td>
<td>C_k) )</td>
<td>( n_k(w_i)/N_k )</td>
</tr>
<tr>
<td>( \sum_{k=1}^{N} n_k(w_i) )</td>
<td>( N_k )</td>
<td>( N_k )</td>
</tr>
</tbody>
</table>

\[
\begin{align*}
P(b_1 | C_k) & \propto \prod_{i=1}^{D} P(b_i | C_k)^{x_i} = \prod_{i=1}^{D} P(o_i | C_k) \\
P(x | C_k) & \propto \prod_{i=1}^{D} P(w_i | C_k)^{x_i} = \prod_{i=1}^{D} P(o_i | C_k)
\end{align*}
\]

\( \Rightarrow \) Classify this document as \( S \).
Question

What’s the approximate value of:
\[ P(“the” | C) \]
(a) in the Bernoulli model
(b) in the multinomial model?

Common words, ‘stop words’, are often removed from feature vectors.

Generative models

- Models that generate observable data randomly based on a distribution
- Examples
  - Coin tossing models

\[ P(x) \overset{\sim}{=} x_1, x_2, x_3, \ldots \]

Coin

\[ H, T, T, H, H, H, H, \ldots \]

Unfair coin

\[ P(H) = 0.7, P(T) = 0.3 \]

\[ H, H, H, H, H, T, H, H, \ldots \]

Dice throwing models

\[ \text{Unbiased dice (} P(X) = 1/6, X \in \{1, \ldots, 6\} \text{)} \]

\[ 2, 4, 3, 5, 3, 6, 5, 4, 6, \ldots \]

\[ \text{Biased dice (} P(X) = (0.1, 0.1, 0.1, 0.1, 0.2, 0.4) \text{)} \]

\[ 6, 6, 5, 5, 6, 1, 2, 6, 6, 6, \ldots \]

Generative models (cont.)

- Spam mail generator

\[ \text{Congratulations to you as we bring to your notice, } \]
\[ \cdots \]

\[ \text{P(O|Spam)} \]

Generative model — Multinomial document model

\[ o_1, o_2, o_3, \ldots, o_L | c_k \]

Terms in Document \( D \) in \( C_k \)

\[ b_1 \]

\[ b_2 \]

……

\[ b_0 \]

\[ P(w_i | c_k) \]

\[ P(w_i | C_k) \]

\[ P(w_i | C_l) \]

\[ P(w_i | C_m) \]

Word relative-frequencies of spam emails

\[ \text{P(“the” | C)} \]

(a) in the Bernoulli model

(b) in the multinomial model?

Smoothing in multinomial document model

- Zero probability problem

\[ P(x | c_k) \overset{\sim}{=} \prod_{t=1}^{D} P(w_t | c_k) \overset{\sim}{=} 0 \text{ if } \exists j : P(w_j | c_k) = 0 \]

\[ P(w_t | C_k) = \frac{n_w(w_t)}{n_k(w_t)} \]

\[ P(w_t | C_k) = \frac{1 + \sum_{j=1}^{K} n_w(w_j) x_{jk}}{D + \sum_{j=1}^{K} n_k(w_j)} \]

known as Laplace’s rule of succession or add one smoothing.
Exercise 1
Use the Bernoulli model and the Naive Bayes assumption for the following.
Consider the vocabulary \( V = \{\text{apple, banana, computer}\} \). We have two classes of documents \( F \) (fruits) and \( E \) (electronics). There are four training documents in class \( F \); they are listed below in terms of the number of occurrences of each word from \( V \) in each document:

\[
\begin{align*}
\text{apple} & : 2; \\
\text{banana} & : 1; \\
\text{computer} & : 0.
\end{align*}
\]

There are also four training documents in class \( E \):

\[
\begin{align*}
\text{apple} & : 2; \\
\text{banana} & : 0; \\
\text{computer} & : 1; \\
\end{align*}
\]

Exercise 2 (cont.)
1. Write the training data as a matrix for each class, where each row corresponds to a training document.
2. Estimate the prior probabilities from the training data.
3. For each class \( F \) and \( E \) and for each word (apple, banana, and computer) estimate the likelihood of the word given the class.
4. Consider two test documents:
   - \( \text{apple}(1); \text{banana}(0); \text{computer}(0) \)
   - \( \text{apple}(2); \text{banana}(0); \text{computer}(1) \)

   For each test document, estimate the posterior probabilities of each class given the document, and hence classify the document.

Exercise 3
Consider two writers, Baker and Clark, who were twins, and who published four and six childrens books, respectively. The following table shows the frequencies of four words, \( \text{wizard, river, star, and warp} \), with respect to the first page of each book, and the information whether the book was a bestseller or not.

<table>
<thead>
<tr>
<th>Author</th>
<th>wizard</th>
<th>river</th>
<th>star</th>
<th>warp</th>
<th>Bestseller</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baker</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>No</td>
</tr>
<tr>
<td>Baker</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Yes</td>
</tr>
<tr>
<td>Baker</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>No</td>
</tr>
<tr>
<td>Clark</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>Yes</td>
</tr>
<tr>
<td>Clark</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>No</td>
</tr>
<tr>
<td>Clark</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>No</td>
</tr>
<tr>
<td>Clark</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Two unpublished book drafts, Doc 1 and Doc 2, were found after the death of the writers, but its not clear which of them wrote the documents.

Exercise 3 (cont.)
1. Without having any information about Doc 1 and Doc 2, decide the most probable author of each document in terms of minimum classification error, and justify your decision.
2. The same analysis of word frequencies was carried out for Doc 3 and Doc 4, whose result is shown below. Using the Naive Bayes classification with the multinomial document model without smoothing, find the author of each document.

<table>
<thead>
<tr>
<th>Author</th>
<th>wizard</th>
<th>river</th>
<th>star</th>
<th>warp</th>
<th>Bestseller</th>
</tr>
</thead>
<tbody>
<tr>
<td>Doc 3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Yes</td>
</tr>
<tr>
<td>Doc 4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>Yes</td>
</tr>
</tbody>
</table>

In addition to modifications to the vocabulary, discuss two possible methods for improving the classification performance.

1. Another document, Doc 5, was found later, and a publisher is considering its publication. Assuming the Naive Bayes classification with the multinomial document model with no smoothing, without identifying the author, predict whether Doc 5 is likely to be a bestseller or not based on the word frequency table for Doc 3 shown below.

<table>
<thead>
<tr>
<th>Author</th>
<th>wizard</th>
<th>river</th>
<th>star</th>
<th>warp</th>
<th>Bestseller</th>
</tr>
</thead>
<tbody>
<tr>
<td>Doc 5</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Using the same situations as in part (5) except that we now know the author of Doc 3 is Baker, predict whether Doc 5 is likely to be a bestseller or not.
Summary

- Our first ‘real’ application of Naive Bayes
- Two BoW models for documents: Multinomial and Bernoulli
- Generative models
- Smoothing (Add-one/Laplace smoothing)
  See Chapter 13 Text classification & Naive Bayes

- As always:
  be able to implement, describe, compare and contrast (see Lecture Note)