

# CS145: INTRODUCTION TO DATA MINING

## 11: Text Data: Naïve Bayes

---

**Instructor: Yizhou Sun**

[yzsun@cs.ucla.edu](mailto:yzsun@cs.ucla.edu)


October 20, 2021

# Methods to be Learnt

	Vector Data	Set Data	Sequence Data/Time Series	Text Data	Graph Data
Classification	Logistic Regression; Decision Tree; NN			Naïve Bayes for Text	Label Propagation
Clustering	K-means; Mixture Models			PLSA	Spectral Clustering
Prediction	Linear Regression; Regression Tree; NN GLM*		AR Model		
Frequent Pattern Mining		Apriori; FP growth	GSP; PrefixSpan		
Similarity Search			DTW		P-PageRank
Ranking					PageRank

# Naïve Bayes for Text

---

- Text Data 
- Revisit of Multinomial Distribution
- Multinomial Naïve Bayes
- Summary

# Text Data

- Word/term
- Document
  - A sequence of words
- Corpus
  - A collection of documents



# Text Classification Applications

- Spam detection

From: [airak@medicana.com.tr](mailto:airak@medicana.com.tr)

Subject: Loan Offer

Do you need a personal or business loan urgent that can be process within 2 to 3 working days? Have you been frustrated so many times by your banks and other loan firm and you don't know what to do? Here comes the Good news Deutsche Bank Financial Business and Home Loan is here to offer you any kind of loan you need at an affordable interest rate of 3% If you are interested let us know.

- Sentiment analysis



The Lion King, complete with jaunty songs by Elton John and Tim Rice, is undeniably and fully worthy of its glorious Disney heritage. It is a gorgeous triumph -- one lion in which the studio can take justified pride.



Between traumas, the movie serves up soothingly banal musical numbers (composed by Elton John and Tim Rice) and silly, rambunctious comedy.

July 31, 2013 | [Full Review...](#)

# Represent A Document


---

Ex: “Do you need a personal or business loan urgent?”

- A document  $d$  is represented by a sequence of words selected from a **vocabulary**
  - $\mathbf{w}_d = (w_{d1}, w_{d2}, \dots, w_{dN_d})$ , where  $w_{di}$  is the id of  $i$ -th word in document  $d$  and  $N_d$  is the length of document  $d$
- A bag-of-words representation
  - $\mathbf{x}_d = (x_{d1}, x_{d2}, \dots, x_{dN})$ , where  $x_{dn}$  is the number of words for  $n$ th word in the vocabulary
  - $x_{dn} = \sum_i 1(w_{di} == n)$

# Example

- c1: *Human machine interface* for Lab ABC computer applications  
 c2: A survey of user opinion of computer system response time  
 c3: The *EPS user interface* management system  
 c4: *System* and *human system* engineering testing of *EPS*  
 c5: Relation of *user-perceived response time* to error measurement
- m1: The generation of random, binary, unordered *trees*  
 m2: The intersection *graph* of paths in *trees*  
 m3: *Graph minors IV*: Widths of *trees* and well-quasi-ordering  
 m4: *Graph minors*: A survey




	c1	c2	c3	c4	c5	m1	m2	m3	m4
<i>human</i>	1	0	0	1	0	0	0	0	0
<i>interface</i>	1	0	1	0	0	0	0	0	0
<i>computer</i>	1	1	0	0	0	0	0	0	0
<i>user</i>	0	1	1	0	1	0	0	0	0
<i>system</i>	0	1	1	2	0	0	0	0	0
<i>response</i>	0	1	0	0	1	0	0	0	0
<i>time</i>	0	1	0	0	1	0	0	0	0
<i>EPS</i>	0	0	1	1	0	0	0	0	0
<i>survey</i>	0	1	0	0	0	0	0	0	1
<i>trees</i>	0	0	0	0	0	1	1	1	0
<i>graph</i>	0	0	0	0	0	0	1	1	1
<i>minors</i>	0	0	0	0	0	0	0	1	1

$x_d$

# Naïve Bayes for Text

---

- Text Data
- Revisit of Multinomial Distribution 
- Multinomial Naïve Bayes
- Summary

# Bernoulli and Categorical Distribution

---

- Bernoulli distribution
  - Discrete distribution that takes two values  $\{0,1\}$ 
    - $P(X = 1) = p$  and  $P(X = 0) = 1 - p$
    - E.g., toss a coin with head and tail
- Categorical distribution
  - Discrete distribution that takes more than two values, i.e.,  $x \in \{1, \dots, K\}$ 
    - Also called generalized Bernoulli distribution, multinoulli distribution
    - $P(X = k) = p_k$  and  $\sum_k p_k = 1$
    - E.g., get 1-6 from a dice with  $1/6$



# Binomial and Multinomial Distribution

---

- Binomial distribution

- Number of successes (i.e., total number of 1's) by repeating  $n$  trials of independent Bernoulli distribution with probability  $p$

- $x$ : number of successes

- $P(X = x) = \binom{n}{x} p^x (1 - p)^{n-x}$

- Multinomial distribution (multivariate random variable)


- Repeat  $n$  trials of independent categorical distribution

- Let  $x_k$  be the number of times value  $k$  has been observed, note  $\sum_k x_k = n$

- $P(X_1 = x_1, X_2 = x_2, \dots, X_K = x_K) = \frac{n!}{x_1! x_2! \dots x_K!} \prod_k p_k^{x_k}$

# Naïve Bayes for Text

---

- Text Data
- Revisit of Multinomial Distribution
- Multinomial Naïve Bayes 
- Summary

# Bayes' Theorem: Basics

---

- Bayes' Theorem: 
$$P(h|\mathbf{X}) = \frac{P(\mathbf{X}|h)P(h)}{P(\mathbf{X})}$$
- Let  $\mathbf{X}$  be a data sample (“*evidence*”)
- Let  $h$  be a *hypothesis* that  $\mathbf{X}$  belongs to class  $C$
- $P(h)$  (*prior probability*): the probability of hypothesis  $h$ 
  - E.g., the probability of “spam” class
- $P(\mathbf{X}|h)$  (*likelihood*): the probability of observing the sample  $\mathbf{X}$ , given that the hypothesis holds
  - E.g., the probability of an email given it’s a spam
- $P(\mathbf{X})$ : marginal probability that sample data is observed
  - $P(\mathbf{X}) = \sum_h P(\mathbf{X}|h) P(h)$
- $P(h|\mathbf{X})$ , (i.e., *posterior probability*): the probability that the hypothesis holds given the observed data sample  $\mathbf{X}$

# Classification: Choosing Hypotheses

---

- *Maximum a posteriori* (maximize the posterior):
  - Useful observation: it does not depend on the denominator  $P(X)$

$$h_{MAP} = \arg \max_{h \in H} P(h | X) = \arg \max_{h \in H} P(X | h)P(h)$$

# Classification by Maximum A Posteriori

---

- Let  $D$  be a training set of tuples and their associated class labels, and each tuple is represented by an  $p$ -D attribute vector  $\mathbf{x} = (x_1, x_2, \dots, x_p)$
- Suppose there are  $m$  classes  $y \in \{1, 2, \dots, m\}$
- Classification is to derive the maximum posteriori, i.e., the maximal  $P(y=j | \mathbf{x})$

- This can be derived from Bayes' theorem

$$p(y = j | \mathbf{x}) = \frac{p(\mathbf{x} | y = j)p(y = j)}{p(\mathbf{x})}$$

- Since  $p(\mathbf{x})$  is constant for all classes, only  $p(\mathbf{x} | y)p(y)$  needs to be maximized

# Now Come to Text Setting: Example

- Data:

	Doc	Words	Class
Training	1	Chinese Beijing Chinese	c
	2	Chinese Chinese Shanghai	c
	3	Chinese Macao	c
	4	Tokyo Japan Chinese	j
Test	5	Chinese Chinese Chinese Tokyo Japan	?

## Vocabulary:

Index	1	2	3	4	5	6
Word	Chinese	Beijing	Shanghai	Macao	Tokyo	Japan

# Modeling

---

- A document is represented as
  - $\mathbf{w}_d = (w_{d1}, w_{d2}, \dots, w_{dN_d})$
  - $w_{di}$  is the  $i$ -th word of  $d$  and  $N_d$  is the length of document  $d$
- Model  $p(\mathbf{w}_d | y = j)$  for class  $j$ 
  - Each word in the sequence  $w_{di}$  is sampled from categorical distribution with parameter vector  $\boldsymbol{\beta}_j = (\beta_{j1}, \beta_{j2}, \dots, \beta_{jN})$  independently
    - $p(w_{di} | y = j) = \beta_{jw_{di}}$  and  $p(\mathbf{w}_d | y = j) = \prod_i \beta_{jw_{di}} = \prod_n \beta_{jn}^{x_{dn}}$
    - Where  $x_{dn}$  is the number of words for  $n$ th word in the vocabulary
- Model  $p(y = j)$ 
  - Follow categorical distribution with parameter vector  $\boldsymbol{\pi} = (\pi_1, \pi_2, \dots, \pi_m)$ , i.e.,
    - $p(y = j) = \pi_j$

# Classification Process Assuming Parameters are Given: Inference

- Find  $y$  that maximizes  $p(y|\mathbf{x}_d)$ , which is equivalently to maximize

$$\begin{aligned}y^* &= \operatorname{argmax}_y p(\mathbf{x}_d, y) \\ &= \operatorname{argmax}_y p(\mathbf{x}_d|y)p(y) \\ &= \operatorname{argmax}_y \prod_n \beta_{yn}^{x_{dn}} \times \pi_y \\ &= \operatorname{argmax}_y \sum_n x_{dn} \log \beta_{yn} + \log \pi_y\end{aligned}$$

# Parameter Estimation via MLE

- Given a corpus and labels for each document

- $D = \{(\mathbf{x}_d, y_d)\}$

- Find the MLE estimators for  $\Theta = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \dots, \boldsymbol{\beta}_m, \boldsymbol{\pi})$

- The log likelihood function for the training dataset

$$\begin{aligned} \log L &= \log \prod_d p(\mathbf{x}_d, y_d | \Theta) = \sum_d \log p(\mathbf{x}_d, y_d | \Theta) \\ &= \sum_d \log p(\mathbf{x}_d | y_d) p(y_d) = \sum_d (x_{dn} \log \beta_{y_d n} + \log \pi_{y_d}) \end{aligned}$$

- The optimization problem

$$\max_{\Theta} \log L$$

s. t.

$$\pi_j \geq 0 \text{ and } \sum_j \pi_j = 1$$

$$\beta_{jn} \geq 0 \text{ and } \sum_n \beta_{jn} = 1 \text{ for all } j$$

# Solve the Optimization Problem

---

- Use the Lagrange multiplier method
- Solution

- $$\hat{\beta}_{jn} = \frac{\sum_{d:y_d=j} x_{dn}}{\sum_{d:y_d=j} \sum_{n'} x_{dn'}}$$

- $\sum_{d:y_d=j} x_{dn}$ : total count of word  $n$  in class  $j$

- $\sum_{d:y_d=j} \sum_{n'} x_{dn'}$ : total count of words in class  $j$

- $$\hat{\pi}_j = \frac{\sum_d 1(y_d=j)}{|D|}$$

- $1(y_d = j)$  is the indicator function, which equals to 1 if  $y_d = j$  holds

- $|D|$ : total number of documents

# Smoothing

---

- What if some word  $n$  does not appear in some class  $j$  in training dataset?
  - $\hat{\beta}_{jn} = \frac{\sum_{d:y_d=j} x_{dn}}{\sum_{d:y_d=j} \sum_{n'} x_{dn'}} = 0$
  - $\Rightarrow p(\mathbf{x}_d | y = j) \propto \prod_n \beta_{yn}^{x_{dn}} = 0$
  - But other words may have a strong indication the document belongs to class  $j$
- Solution: add-1 smoothing or Laplacian smoothing
  - $\hat{\beta}_{jn} = \frac{\sum_{d:y_d=j} x_{dn} + 1}{\sum_{d:y_d=j} \sum_{n'} x_{dn'} + N}$
  - $N$ : total number of words in the vocabulary
  - Check:  $\sum_n \hat{\beta}_{jn} = 1$ ?

# Example

- Data:

	Doc	Words	Class
Training	1	Chinese Beijing Chinese	c
	2	Chinese Chinese Shanghai	c
	3	Chinese Macao	c
	4	Tokyo Japan Chinese	j
Test	5	Chinese Chinese Chinese Tokyo Japan	?

- Vocabulary:

Index	1	2	3	4	5	6
Word	Chinese	Beijing	Shanghai	Macao	Tokyo	Japan

- Learned parameters (with smoothing):

$$\hat{\beta}_{c1} = \frac{5+1}{8+6} = \frac{3}{7}$$

$$\hat{\beta}_{c2} = \frac{1+1}{8+6} = \frac{1}{7}$$

$$\hat{\beta}_{c3} = \frac{1+1}{8+6} = \frac{1}{7}$$

$$\hat{\beta}_{c4} = \frac{1+1}{8+6} = \frac{1}{7}$$

$$\hat{\beta}_{c5} = \frac{0+1}{8+6} = \frac{1}{14}$$

$$\hat{\beta}_{c6} = \frac{0+1}{8+6} = \frac{1}{14}$$

$$\hat{\beta}_{j1} = \frac{1+1}{3+6} = \frac{2}{9}$$

$$\hat{\beta}_{j2} = \frac{0+1}{3+6} = \frac{1}{9}$$

$$\hat{\beta}_{j3} = \frac{0+1}{3+6} = \frac{1}{9}$$

$$\hat{\beta}_{j4} = \frac{0+1}{3+6} = \frac{1}{9}$$

$$\hat{\beta}_{j5} = \frac{1+1}{3+6} = \frac{2}{9}$$

$$\hat{\beta}_{j6} = \frac{1+1}{3+6} = \frac{2}{9}$$

$$\hat{\pi}_c = \frac{3}{4}$$

$$\hat{\pi}_j = \frac{1}{4}$$

# Example (Continued)

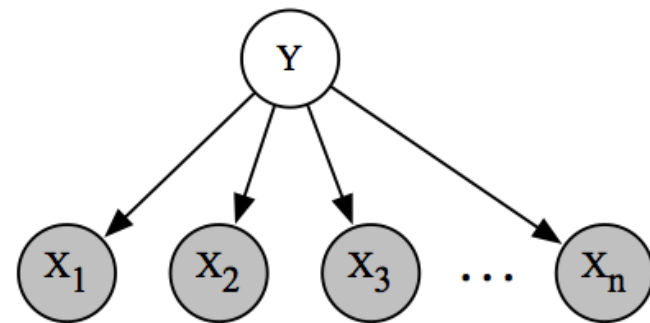
---

- Classification stage
  - For the test document  $d=5$ , compute
  - $p(y = c | \mathbf{x}_5) \propto p(y = c) \times \prod_n \beta_{cn}^{x_{5n}} = \frac{3}{4} \times \left(\frac{3}{7}\right)^3 \times \left(\frac{1}{14}\right) \times \left(\frac{1}{14}\right) \approx 0.0003$
  - $p(y = j | \mathbf{x}_5) \propto p(y = j) \times \prod_n \beta_{jn}^{x_{5n}} = \frac{1}{4} \times \left(\frac{2}{9}\right)^3 \times \left(\frac{2}{9}\right) \times \left(\frac{2}{9}\right) \approx 0.0001$
  - Conclusion:  $\mathbf{x}_5$  should be classified into c class

# A More General Naïve Bayes Framework

- Let  $D$  be a training set of tuples and their class labels, and each tuple is represented by an  $p$ -D attribute vector  $\mathbf{x} = (x_1, x_2, \dots, x_p)$
- Suppose there are  $m$  classes  $y \in \{1, 2, \dots, m\}$
- Goal: Find  $y = \arg \max_y p(y|\mathbf{x}) = p(y, \mathbf{x})/p(\mathbf{x}) \propto p(\mathbf{x}|y)p(y)$
- A simplified assumption: attributes are **conditionally independent given the class** (class conditional independency):

- $p(\mathbf{x}|y) = \prod_k p(x_k|y)$
- $p(x_k|y)$  can follow any distribution,
  - e.g., Gaussian, Bernoulli, categorical, ...




# Generative Model vs. Discriminative Model

---

- Generative model
  - *model joint probability  $p(\mathbf{x}, y)$*
  - E.g., naïve Bayes
- Discriminative model
  - *model conditional probability  $p(y|\mathbf{x})$*
  - E.g., logistic regression

# Naïve Bayes for Text

---

- Text Data
- Revisit of Multinomial Distribution
- Multinomial Naïve Bayes
- Summary 

# Summary

---

- Text data
  - Bag of words representation
- Naïve Bayes for Text
  - Multinomial naïve Bayes