



CS464

Chapter 4: Naïve Bayes

(slides based on the slides provided by Öznur Taştan and Mehmet Koyutürk)



Last Chapter: Density Estimation

- Maximum Likelihood Estimate MLE: choose θ that maximizes the probability of observed data \mathcal{D}

$$\hat{\theta}_{MLE} = \arg \max_{\theta} \mathbf{P}(\mathcal{D} | \theta)$$

- Maximum a Posteriori (MAP) estimate: choose θ that is most probable given the prior probability of θ and the data \mathcal{D}

$$\begin{aligned} \hat{\theta}_{MAP} &= \arg \max_{\theta} \mathbf{P}(\theta | \mathcal{D}) \\ &= \arg \max_{\theta} \frac{\mathbf{P}(\mathcal{D} | \theta) \mathbf{P}(\theta)}{\mathbf{P}(\mathcal{D})} \end{aligned}$$

Outline Today

- Naïve Bayes Classifier
 - Generalization of maximum a posteriori estimation
- Text Classification
 - Application of Naïve Bayes
 - Illustration of feature extraction/encoding and feature selection

A Bayesian Classifier

- Compute the conditional probability of each value of Y given the attributes
- Classify the example into the class that is most probable given the attributes

$$Y^* = \arg \max_{y_k} \mathbf{P} (Y = y_k \mid X)$$

Learning a Classifier By Learning $P(Y|X)$

Consider $Y = \text{Wealth}$, $X = \langle \text{Gender}, \text{HoursWorked} \rangle$

Joint probability table 

$P(G, W, H)$

W : Wealth

G : Gender

H : HoursWorked

gender	hours_worked	wealth	
Female	v0:<40.5-	poor	0.253122 
		rich	0.0245895 
	v1:>40.5+	poor	0.0421768 
		rich	0.0116293 
Male	v0:<40.5-	poor	0.331313 
		rich	0.0971295 
	v1:>40.5+	poor	0.134106 
		rich	0.105933 

Conditional probability table 
 $P(W|G, H)$

Gender	HrsWorked	$P(\text{rich} G, HW)$	$P(\text{poor} G, HW)$
F	<40.5	.09	.91
F	>40.5	.21	.79
M	<40.5	.23	.77
M	>40.5	.38	.62

A Bayesian Classifier

Predict the class label that is most probable given the attributes (values of features)

$$Y^* = \arg \max_{y_k} \mathbf{P} (Y = y_k \mid X)$$

Gender	HrsWorked	P(rich G,HW)	P(poor G,HW)
F	<40.5	.09	.91
F	>40.5	.21	.79
M	<40.5	.23	.77
M	>40.5	.38	.62

Building a Classifier By Learning $P(Y|X)$

- Two binary features, one class label

Consider $Y = \text{Wealth}$, $X = \langle \text{Gender}, \text{HoursWorked} \rangle$

gender	hours_worked	wealth		
Female	v0:40.5-	poor	0.253122	<div></div>
		rich	0.0245895	<div></div>
	v1:40.5+	poor	0.0421768	<div></div>
		rich	0.0116293	<div></div>
Male	v0:40.5-	poor	0.331313	<div></div>
		rich	0.0971295	<div></div>
	v1:40.5+	poor	0.134106	<div></div>
		rich	0.105933	<div></div>

Gender	HrsWorked	P(rich G,HW)	P(poor G,HW)
F	<40.5	.09	.91
F	>40.5	.21	.79
M	<40.5	.23	.77
M	>40.5	.38	.62

How many parameters do we need to estimate?

Gender	HrsWorked	P(rich G,HW)	P(poor G,HW)
F	<40.5	.09	.91
F	>40.5	.21	.79
M	<40.5	.23	.77
M	>40.5	.38	.62

- Suppose $X = [X_1, \dots, X_n]$ and all X_i s and Y are boolean variables.
- To estimate $P(Y|X_1, \dots, X_n)$ how many parameters do we need to estimate?

Can we reduce the number of parameters using Bayes' Rule?

$$\mathbf{P}(Y | X) = \frac{\mathbf{P}(X | Y)\mathbf{P}(Y)}{\mathbf{P}(X)}$$

- Suppose $X = [X_1, \dots, X_n]$ and all X_i s and Y are boolean variables.
- To estimate $\mathbf{P}(X_1, \dots, X_n | Y)$ how many parameters do we need to estimate?
- How many parameters are needed to define $\mathbf{P}(Y)$

Can we reduce the number of parameters using Bayes Rule?

$$\mathbf{P}(Y | X) = \frac{\mathbf{P}(X | Y)\mathbf{P}(Y)}{\mathbf{P}(X)}$$

- Suppose $X = [X_1, \dots, X_n]$ and all X_i s and Y are boolean variables.
 - To estimate $\mathbf{P}(X_1, \dots, X_n | Y)$ how many parameters do we need to estimate? $2(2^n - 1)$
 - How many parameters are needed to define $\mathbf{P}(Y)$?
- 30 features \rightarrow more than 30 billion parameters!

Naïve Bayes

- Naïve Bayes assumes

$$\mathbf{P} (X_1, \dots, X_n \mid Y) = \prod_i \mathbf{P} (X_i \mid Y)$$

- Random variables (features) X_i and X_j are conditionally independent of each other given the class label Y for all $i \neq j$

Conditional Independence

- X and Y are conditionally independent given Z iff the conditional probability of the joint variable can be written as product of conditional probabilities:

$$X \perp Y|Z \Leftrightarrow P(X, Y|Z) = P(X|Z) P(Y|Z)$$

Naïve Bayes in a Nutshell

- Train Naïve Bayes (examples)

for each* value y_k

estimate $\pi_k \equiv P(Y = y_k)$

for each* value x_{ij} of each attribute X_i

estimate $\theta_{ijk} \equiv P(X_i = x_{ij} | Y = y_k)$

How many parameters?

- Classify (X^{new})

- > $2n + 1$ if Y is binary

$$Y^{new} \leftarrow \arg \max_{y_k} P(Y = y_k) \prod_i P(X_i^{new} | Y = y_k)$$

$$Y^{new} \leftarrow \arg \max_{y_k} \pi_k \prod_i \theta_{ijk}$$

* probabilities must sum to 1, so need estimate only $n-1$ of these...

Example: Shall we play tennis?

Outlook	Temperature	Humidity	Windy	Play
sunny	hot	high	false	NO
sunny	hot	high	true	NO
overcast	hot	high	false	YES
rainy	mild	high	false	YES
rainy	cool	normal	false	YES
rainy	cool	normal	true	NO
overcast	cool	normal	true	YES
sunny	mild	high	false	NO
sunny	cool	normal	false	YES
rainy	mild	normal	false	YES
sunny	mild	normal	true	YES
overcast	mild	high	true	YES
overcast	hot	normal	false	YES
rainy	mild	high	true	NO

Applying Naïve Bayes Assumption

$$\mathbf{P}(Play | O, T, H, W) = \frac{\mathbf{P}(O, T, H, W | Play) \mathbf{P}(Play)}{\mathbf{P}(O, T, H, W)}$$

$$\propto \mathbf{P}(O, T, H, W | Play) \mathbf{P}(Play)$$

Applying the Naïve Bayes Assumption:

$$\mathbf{P}(O, T, H, W | Play) = \mathbf{P}(O | Play) \mathbf{P}(T | Play) \mathbf{P}(H | Play) \mathbf{P}(W | Play)$$

O: Outlook
T: Temperature
H: Humidity
W: Wind

Applying Naïve Bayes Assumption

$$\mathbf{P}(Play | O, T, H, W) = \frac{\mathbf{P}(O, T, H, W | Play) \mathbf{P}(Play)}{\mathbf{P}(O, T, H, W)}$$

$$\propto \mathbf{P}(O, T, H, W | Play) \mathbf{P}(Play)$$

Applying the Naïve Bayes Assumption:

$$\mathbf{P}(O, T, H, W | Play) = \mathbf{P}(O | Play) \mathbf{P}(T | Play) \mathbf{P}(H | Play) \mathbf{P}(W | Play)$$

Parameters to Estimate

Consider each feature independently and for each class label y_k and $x_{i,j}$ value of feature i estimate $\mathbf{P}(X_i = x_{i,j} \mid Y = y_k)$:

And estimate the class prior $\mathbf{P}(Y = y_k)$:

Parameters to Estimate

Consider each feature independently and for each class label y_k and $x_{i,j}$ value of feature i estimate $\mathbf{P}(X_i = x_{i,j} \mid Y = y_k)$:

$$\mathbf{P}(O = \text{sunny} \mid \text{Play} = \text{Yes})$$

$$\mathbf{P}(O = \text{sunny} \mid \text{Play} = \text{No})$$

$$\mathbf{P}(O = \text{overcast} \mid \text{Play} = \text{Yes})$$

$$\mathbf{P}(O = \text{overcast} \mid \text{Play} = \text{No})$$

...

$$\mathbf{P}(T = \text{hot} \mid \text{Play} = \text{Yes})$$

$$\mathbf{P}(T = \text{hot} \mid \text{Play} = \text{No})$$

...

$$\mathbf{P}(H = \text{high} \mid \text{Play} = \text{Yes})$$

$$\mathbf{P}(H = \text{high} \mid \text{Play} = \text{No})$$

...

$$\mathbf{P}(W = \text{true} \mid \text{Play} = \text{Yes})$$

$$\mathbf{P}(W = \text{true} \mid \text{Play} = \text{No}) \dots$$

And estimate the class prior $\mathbf{P}(Y = y_k)$:

Parameters to Estimate

Consider each feature independently and for each class label y_k and $x_{i,j}$ value of feature i estimate $\mathbf{P}(X_i = x_{i,j} \mid Y = y_k)$:

$$\mathbf{P}(O = \text{sunny} \mid \text{Play} = \text{Yes})$$

$$\mathbf{P}(O = \text{sunny} \mid \text{Play} = \text{No})$$

$$\mathbf{P}(O = \text{overcast} \mid \text{Play} = \text{Yes})$$

$$\mathbf{P}(O = \text{overcast} \mid \text{Play} = \text{No})$$

...

$$\mathbf{P}(T = \text{hot} \mid \text{Play} = \text{Yes})$$

$$\mathbf{P}(T = \text{hot} \mid \text{Play} = \text{No})$$

...

$$\mathbf{P}(H = \text{high} \mid \text{Play} = \text{Yes})$$

$$\mathbf{P}(H = \text{high} \mid \text{Play} = \text{No})$$

...

$$\mathbf{P}(W = \text{true} \mid \text{Play} = \text{Yes})$$

$$\mathbf{P}(W = \text{true} \mid \text{Play} = \text{No}) \dots$$

And estimate the class prior $\mathbf{P}(Y = y_k)$:

$$\mathbf{P}(\text{Play} = \text{Yes}) \quad (\text{Note that } \mathbf{P}(\text{Play} = \text{No}) = 1 - \mathbf{P}(\text{Play} = \text{Yes}))$$

Relative Frequencies

- Consider each feature independently and estimate:

$P(O | Play = Y)$, $P(T | Play = Y)$, $P(H | Play = Y)$, and $P(W | Play = Y)$
 $P(O | Play = N)$, $P(T | Play = N)$, $P(H | Play = N)$, and $P(W | Play = N)$

Outlook	Y	N
sunny	2/9	3/5
overcast	4/9	0/5
rainy	3/9	2/5

Temperature	Y	N
hot	2/9	2/5
mild	4/9	2/5
cool	3/9	1/5

Humidity	Y	N
high	3/9	4/5
normal	6/9	1/5

Windy	Y	N
false	6/9	2/5
true	3/9	3/5

$$P(Play = \text{Yes}) = \frac{9}{14} \text{ and } P(Play = \text{No}) = \frac{5}{14}$$

Applying Naïve Bayes

- Posterior probability for a new instance with the feature vector:
- $X_{\text{new}} = (\text{sunny}, \text{cool}, \text{high}, \text{true})$

Posterior

Likelihood

Prior

$$\mathbf{P} (Play | X) \propto \mathbf{P} (X | Play) \mathbf{P} (Play)$$

$$\mathbf{P} (Play = Y | X) \propto \mathbf{P} (X | Play = Y) \mathbf{P} (Play = Y)$$

$$\mathbf{P} (Play = N | X) \propto \mathbf{P} (X | Play = N) \mathbf{P} (Play = N)$$

Applying Naïve Bayes

X = (sunny, cool, humid, windy)

- Estimating the likelihood:

$$\begin{aligned}\mathbf{P}(X | \text{Play} = Y) &= \mathbf{P}(O = \text{sunny} | \text{Play} = Y) \mathbf{P}(T = \text{cool} | \text{Play} = Y) \mathbf{P}(H = \text{high} | \text{Play} = Y) \mathbf{P}(W = \text{true} | \text{Play} = Y) \\ &= \frac{2}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} \cdot \frac{3}{9} \approx 0.0082\end{aligned}$$

$$\begin{aligned}\mathbf{P}(X | \text{Play} = N) &= \mathbf{P}(O = \text{sunny} | \text{Play} = N) \mathbf{P}(T = \text{cool} | \text{Play} = N) \mathbf{P}(H = \text{high} | \text{Play} = N) \mathbf{P}(W = \text{true} | \text{Play} = N) \\ &= \frac{3}{5} \cdot \frac{1}{5} \cdot \frac{4}{5} \cdot \frac{3}{5} = 0.0576\end{aligned}$$

- Estimating the posterior:

$$\mathbf{P}(\text{Play} = Y | X) \propto \mathbf{P}(X | \text{Play} = Y) \mathbf{P}(\text{Play} = Y) = 0.0082 * \frac{9}{14} \approx 0.0052$$

$$\mathbf{P}(\text{Play} = N | X) \propto \mathbf{P}(X | \text{Play} = N) \mathbf{P}(\text{Play} = N) = 0.0576 * \frac{5}{14} \approx 0.0205$$

- Class label predicted for X is then Play = No

Numerical Issues

- Multiplying lots of probabilities, which are between 0 and 1 by definition, can result in floating--point underflow.
- Underflow occurs when you perform an operation that's smaller than the smallest magnitude non--zero number.
- Since $\log(xy) = \log(x) + \log(y)$, it is better to perform all computations by summing logs of probabilities rather than multiplying probabilities.
- Class with highest final un--normalized log probability score is still the most probable.

Underflow

- Therefore, instead of using this formulation:

$$Y^{new} \leftarrow \arg \max_{y_k} P(Y = y_k) \prod_i P(X_i^{new} | Y = y_k)$$

- Use the following equivalent rule:

$$Y^{new} \leftarrow \arg \max_{y_k} \left(\log P(Y = y_k) + \sum_i \log P(X_i^{new} | Y = y_k) \right)$$

$$Y^{new} \leftarrow \arg \max_{y_k} \left(\log \pi_k + \sum_i \log \theta_{ijk} \right)$$

- Avoiding underflow is an important implementation detail!

Text Classification Using Naïve Bayes

Spam?

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

Identifying Spam

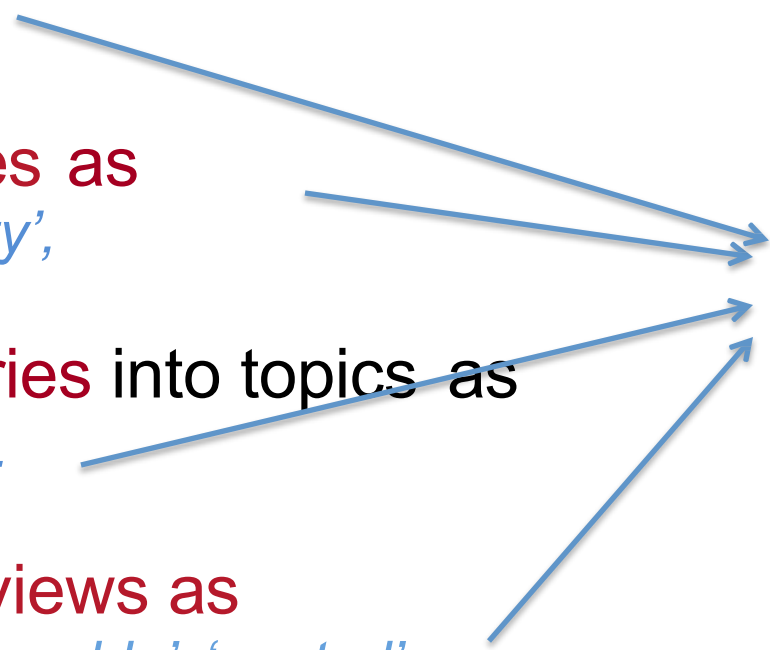
Spam?

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.

Other Text Classification Tasks

- Classify **email** as
'Spam', 'Ham'
- Classify **web pages** as
'Student', 'Faculty', 'Other'
- Classify **news stories** into topics as
'Sports', 'Politics'..
- Classify **movie reviews** as
'favorable', 'unfavorable', 'neutral'

Text Classification

- Classify **email** as
 - *'Spam', 'Ham'*
 - Classify **web pages** as
 - *'Student', 'Faculty', 'Other'*
 - Classify **news stories** into topics as
 - *'Sports', 'Politics'..*
 - Classify **movie reviews** as
 - *'favorable', 'unfavorable', 'neutral'*
 - What about the features X ?
 - How to represent the document?
- 
- Class labels, y

How do we represent a document?

- A sequence of words?
 - computationally very expensive, can be difficult to train
- A set of words (**Bag-of-Words**)
 - Ignore the position of the word in the document
 - Ignore the ordering of the words in the document
 - Consider the words in a predefined vocabulary



Image courtesy: Joseph Gonzalez

Document Models

- **Bernoulli document model:** a document is represented by a binary feature vector, whose elements indicate the absence or presence of corresponding word in the document
- **Multinomial document model:** a document is represented by an integer feature vector, whose elements indicate the frequency of corresponding word in the document

Bag-of-words document models

- **Document:**

Congratulations to you as we bring to your notice, the results of Category draws of THE CASINO LOTTO FROMO INT. We are Happy to inform you that you have emerged a winner under the First Category, which is part of our promotional draws.

Term	Bernoulli	Multinomial
A	1	1
AM	0	0
ARE	1	1
:	:	:
CAN	0	0
CASINO	1	1
CATEGORY	1	2
:	:	:
THE	1	4
TO	1	3
WINNER	1	1
YOU	1	3
YOUR	1	1

Example

- Classify documents as **Sports** and **Informatics**
- Assume the vocabulary contains 8 words
 - Good vocabularies usually do not include common words (a.k.a. stop words)

Vocabulary contains:

w_1	=	goal
w_2	=	tutor
w_3	=	variance
w_4	=	speed
w_5	=	drink
w_6	=	defence
w_7	=	performance
w_8	=	field

Training Data

- **Rows** are **documents**
 - 6 examples of sports documents
 - 5 examples of informatics documents
- **Columns** are **words** in the order of vocabulary

$$\mathbf{B}^{\text{Sport}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}$$
$$\mathbf{B}^{\text{Inf}} = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Vocabulary contains:

w_1 = goal
 w_2 = tutor
 w_3 = variance
 w_4 = speed
 w_5 = drink
 w_6 = defence
 w_7 = performance
 w_8 = field

Estimating Parameters

$$\mathbf{B}^{\text{Sport}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}$$
$$\mathbf{B}^{\text{Inf}} = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

$$\mathbf{P}(Y = \textit{Sport}) = 6/11, \quad \mathbf{P}(Y = \textit{Informatics}) = 5/11$$

$$(\mathbf{P}(w_t | Y = \textit{Sport})) = (3/6 \quad 1/6 \quad 2/6 \quad 3/6 \quad 3/6 \quad 4/6 \quad 4/6 \quad 4/6)$$

$$(\mathbf{P}(w_t | Y = \textit{Informatics})) = (1/5 \quad 3/5 \quad 3/5 \quad 1/5 \quad 1/5 \quad 1/5 \quad 3/5 \quad 1/5)$$

Bernoulli Document Model

Features: $X = (X_1, \dots, X_{|V|})$: length $|V|$ **binary vector** of word occurrences

Model's Generative Process:

for each word t in vocabulary:

Spin biased coin t

if: heads $x_t \leftarrow 1$ **else:** heads $x_t \leftarrow 0$

Classification with Bernoulli Model

Training data

Matrix X , document i feature vector: X_i presence of word t in the document i , X_{it}

Parameter Estimation(MLE)

$$\text{Priors: } \mathbf{P}(Y = y_k) \approx \frac{N_k}{N}$$

$$\text{Likelihoods: } \mathbf{P}(w_t | Y = y_k) \approx \frac{n_k(w_t)}{N_k} \quad (\text{Fraction of } k \text{ documents with word } w_t)$$

Classify new document D with feature vector X :

$$\mathbf{P}(Y = y_k | X) \propto \mathbf{P}(Y = y_k) \mathbf{P}(X | Y = y_k)$$

$$\mathbf{P}(X | Y = y_k) = \prod_{t=1}^{|V|} [X_t \mathbf{P}(w_t | Y = y_k) + (1 - X_t)(1 - \mathbf{P}(w_t | Y = y_k))]$$

Classifying a given sample

- A test document:

$$X^{new} = [1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1]$$

- Priors and likelihoods:

$$\mathbf{P}(Y = Sport) = 6/11, \quad \mathbf{P}(Y = Informatics) = 5/11$$

$$(\mathbf{P}(w_t | Y = Sport)) = (3/6 \quad 1/6 \quad 2/6 \quad 3/6 \quad 3/6 \quad 4/6 \quad 4/6 \quad 4/6)$$

$$(\mathbf{P}(w_t | Y = Informatics)) = (1/5 \quad 3/5 \quad 3/5 \quad 1/5 \quad 1/5 \quad 1/5 \quad 3/5 \quad 1/5)$$

Vocabulary contains:

w_1	=	goal
w_2	=	tutor
w_3	=	variance
w_4	=	speed
w_5	=	drink
w_6	=	defence
w_7	=	performance
w_8	=	field

- Posterior probabilities:

$$\mathbf{P}(Y = S) | X^{new} \propto \mathbf{P}(Y = S) \prod_{t=1}^8 X_{1,t} \mathbf{P}(w_t | S) + (1 - X_{1,t})(1 - \mathbf{P}(w_t | S)) = 5.6 \times 10^{-3}$$

$$\mathbf{P}(Y = I) | X^{new} \propto \mathbf{P}(Y = I) \prod_{t=1}^8 X_{1,t} \mathbf{P}(w_t | I) + (1 - X_{1,t})(1 - \mathbf{P}(w_t | I)) = 9.3 \times 10^{-6}$$

- Classify this document as **Sports**

Multinomial Document Model

Features: $X = (X_1, \dots, X_{|V|})$: length $|V|$ **integer vector** of word occurrences

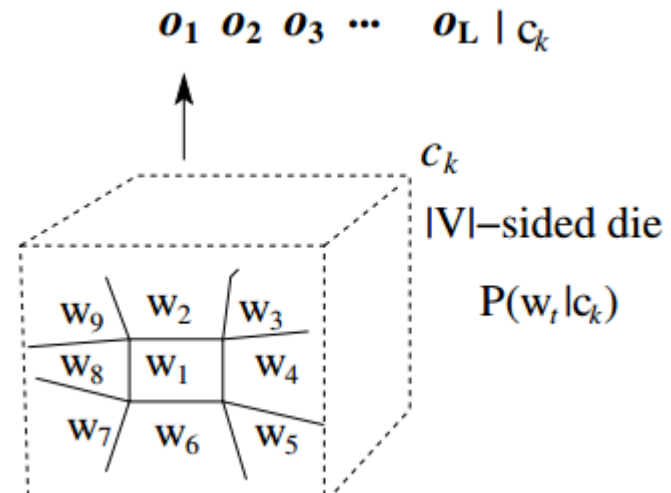
Model's Generative Process:

$x \leftarrow$ vector of zeros

for each word in the document:

Roll biased $|V|$ -sided die

$x_t \leftarrow x_t + 1$



Multinomial Document Model

Features: $X = (X_1, \dots, X_{|V|})$: length $|V|$ **integer vector** of word occurrences

Model's Generative Process:

$x \leftarrow$ vector of zeros

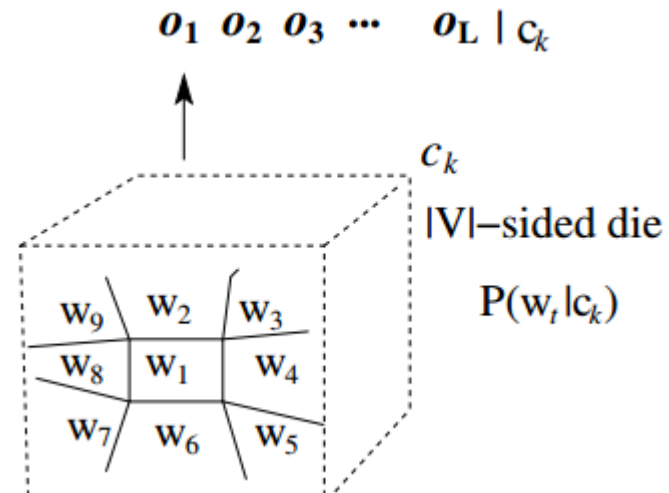
for each word in the document:

Roll biased $|V|$ -sided die

$x_t \leftarrow x_t + 1$



Words are i.i.d. samples from a multinomial distribution.



Classification with Multinomial Model

Training data:

Matrix X , document i feature vector: X_i

$X_{i,t}$: the count of number of times w_t occurs in document i

$z_{i,k}$: if document i is of class y_k , 0 otherwise

Parameter estimation (MLE)

$$\text{Priors: } \mathbf{P}(Y = y_k) \approx \frac{N_k}{N}$$

$$\text{Likelihoods: } \mathbf{P}(w_t | Y = y_k) \approx \frac{\sum_{i=1}^N X_{i,t} z_{i,k}}{\sum_{t'=1}^{|V|} \sum_{i=1}^N X_{i,t'} z_{i,k}}$$

The relative frequency of w_t in documents of class $Y = y_k$ with respect to the total number of words in documents of that class

Classify new document D with feature vector X :

$$\mathbf{P}(Y = y_k | X) \propto \mathbf{P}(Y = y_k) \mathbf{P}(X | Y = y_k)$$

$$\mathbf{P}(X | Y = y_k) = \prod_{t=1}^{|V|} \mathbf{P}(w_t | Y = y_k)^{X_t}$$

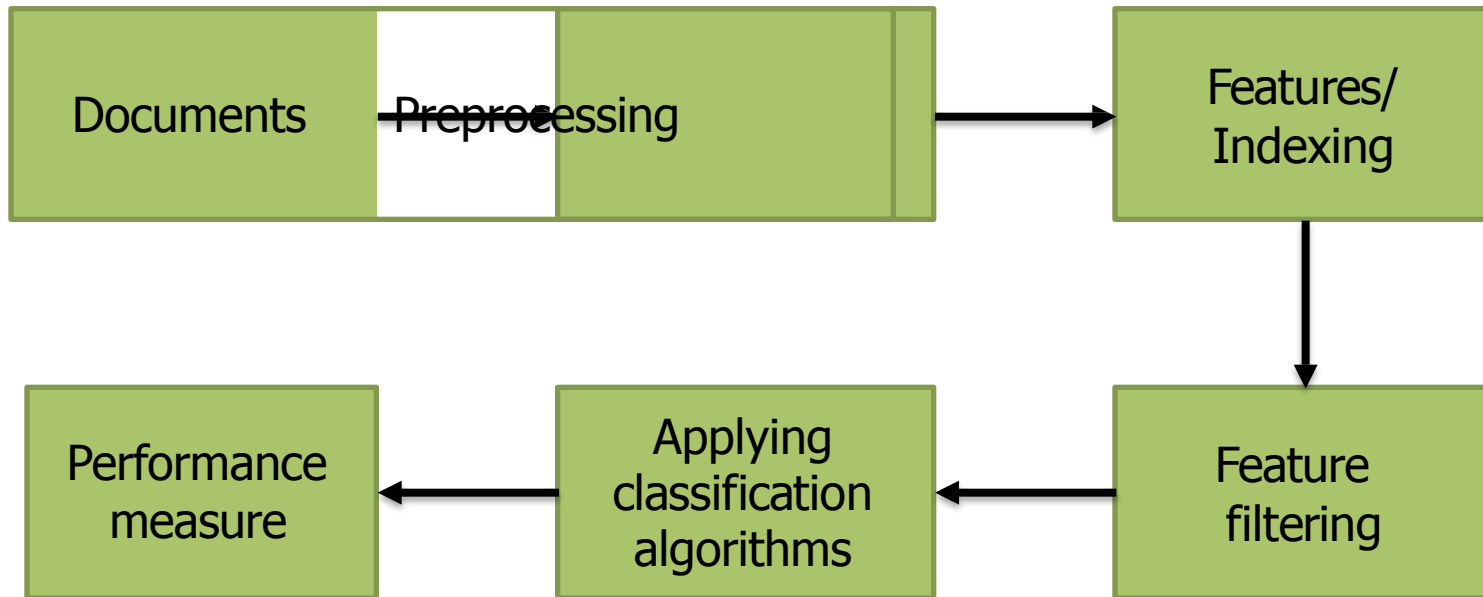
Add-one (Laplace) Smoothing

Parameter estimation (MAP, with $\alpha = 1$)

$$\text{Likelihoods: } \mathbf{P}(w_t | Y = y_k) \approx \frac{\alpha + \sum_{i=1}^N X_{i,t} z_{i,k}}{|V|\alpha + \sum_{t'=1}^{|V|} \sum_{i=1}^N X_{i,t'} z_{i,k}}$$

- Add one imaginary occurrence of every word to every document

Text Classification Framework



Preprocessing

- **Token normalization**

- Remove superficial character variances from words

normelization -> normalization

- **Stop--word removal**

- Remove predefined common words that are not specific or discriminatory to the different classes

is, a, the, you, as...

- **Stemming**

- Reduce different forms of the same word into a single word (base/root form)

swimming, swimmer, swims -> swim

- **Feature selection**

- Choose features that are more relevant and complementary

can be part of the design process, but in general it is done computationally by trying different combinations

Preprocessing

- Mr. O'Neill thinks that the boys' stories about Chile's capital aren't amusing.

How to handle special cases involving apostrophes, hyphens etc?

C++, C#, URLs, emails, phone numbers, dates
San Francisco, Los Angeles

Tokenization

- Divide the text into a sequence of words by combining, dividing words, handling special characters etc.
- Issues of tokenization are language specific
 - Requires the language to be known
German compound nouns
 - East Asian Languages (Chinese, Japanese, Korean, Thai)
 - Text is written without any spaces between words

Normalization

- Token normalization
 - Canonicalizing tokens so that matches occur despite superficial differences in the character sequences of the tokens
 - U.S.A vs USA
 - Anti--discriminatory vs antidiscriminatory
 - Car vs automobile?

Stop Words

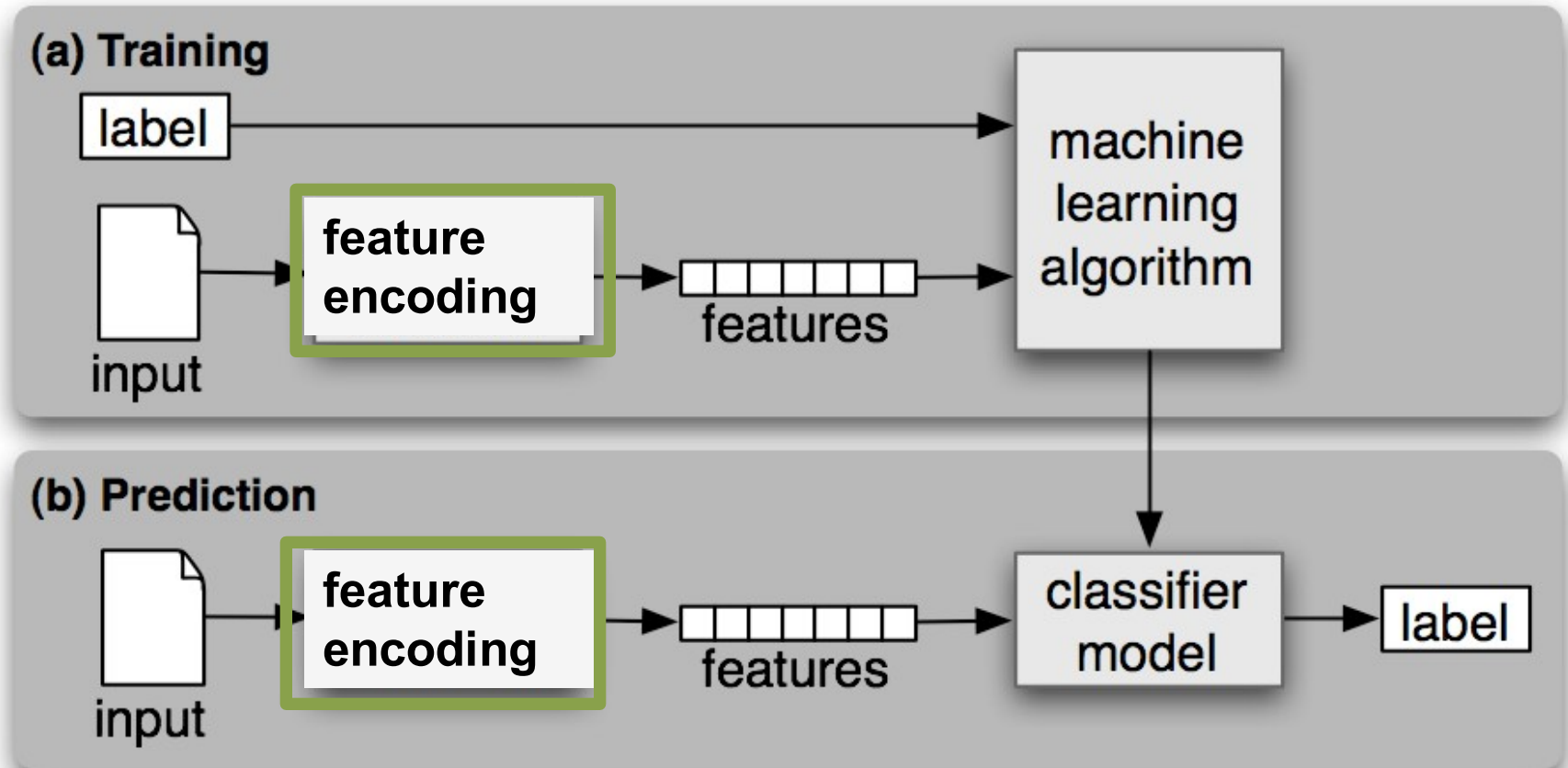
- Very common words that have no discriminatory power

a	an	and	are	as	at	be	by	for	from
has	he	in	is	it	its	of	on	that	the
to	was	were	will	with					

► **Figure 2.5** A stop list of 25 semantically non-selective words which are common in Reuters-RCV1.

- Sort terms by collection frequency and take the most frequent words
- For an application, an additional domain specific stopword list may be constructed
 - In a collection about insurance practices, “insurance” would be a stop word

Feature Encoding



Feature Encoding

- How to represent the features
- Feature encoding can have tremendous impact on the classifier

Feature Extraction vs Feature Selection

- **Feature extraction:**

- Transform data into a new feature space, usually by mapping existing features into a lower dimensional space (PCA, ICA, etc. **We will come back**)

- **Feature selection:**

- Select a subset of the existing features without a transformation

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \rightarrow \begin{bmatrix} x_{i_1} \\ x_{i_2} \\ \vdots \\ x_{i_M} \end{bmatrix}$$

Feature (Subset) Selection

- Necessary in a number of situations:
 - Features may be expensive to obtain
 - Evaluate a large number of features in the test bed and select a subset for the final implementation
- You want to extract meaningful rules from your classifier
- Fewer features means fewer model parameters
 - Improved generalization capabilities
 - Reduced complexity and run-time

Runtime of Naïve Bayes

- It is fast
- Computation of parameters can be done in $O(CD)$
 - C : number of classes
 - D : number of attributes/features

Incremental Updates

- If the model is going to be updated very often as new data come, you may implement it such that it allows easy incremental updates
- For example: Store raw counts instead of probabilities
 - New example of class k :
 - For each feature update the counts based on the example feature vector
 - Update the class counts, update the number of training data
 - When you need to classify, compute the probabilities

The Independence Assumption

$$P(X_1 \dots X_n | Y) \neq \prod_i P(X_i | Y)$$

- Usually features are not conditionally independent
 - That is why it is called naïve
- In practice it often works well
 - Naïve Bayes does not produce accurate probability estimates when its independence assumptions are violated, but it may still (and often) pick the correct maximum-probability class in many cases [Domingos&Pazzani, 1996].
- Typically handles noise well since it does not even focus on completely fitting the training data

What You Should Know

- Training and using classifiers based on Bayes rule
- Conditional independence
 - What it is
 - Why it is important
- Naïve Bayes
 - What it is
 - How to estimate the parameters
 - How to make predictions
- Mutual Information is a good measure for filtering features