COMPUTER SCIENCE
UNIVERSITY OF MARYLAND

Classification, Linear Models, Naïve Bayes

CMSC 470

Marine Carpuat

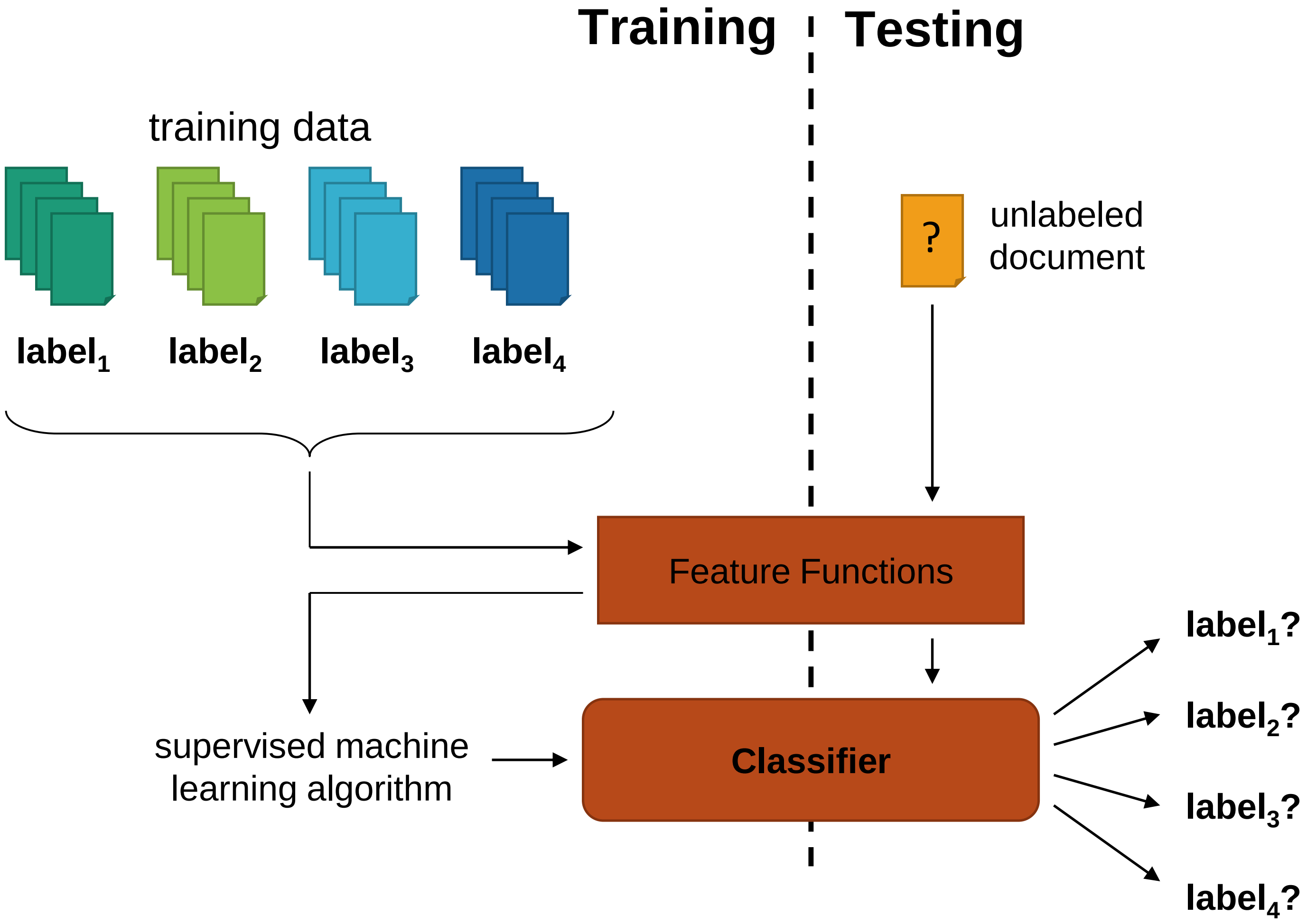
Slides credit: Dan Jurafsky & James Martin, Jacob Eisenstein

Today

- Text classification problems
    - and their evaluation
- Linear classifiers
    - Features & Weights
    - Bag of words
    - Naïve Bayes

Classification problems

Multiclass Classification



Is this spam?

From: "Fabian Starr“
<Patrick_Freeman@pamietaniepeerelu.pl>

Subject: Hey! Sofware for the funny prices!

Get the great discounts on popular software today
for PC and Macintosh
http://iiled.org/Cj4Lmx

70-90% Discounts from retail price!!!
All sofware is instantly available to download - No
Need Wait!

What is the subject of this article?

MEDLINE Article

Available online at www.sciencedirect.com

Brain and Cognition

Syntactic frame and verb bias in aphasia: Plausibility judgments of undergoer-subject sentences

Susanne Gahl,[a] Lise Menn,[b] Gail Ramsberger,[b] Daniel S. Jurafsky,[b] Elizabeth Elder,[b] Molly Rewega,[b] and L. Holland Audrey[c]

?

MeSH Subject Category Hierarchy

- Antogonists and Inhibitors
- Blood Supply
- Chemistry
- Drug Therapy
- Embryology
- Epidemiology
- …

Text Classification

- Assigning subject categories, topics, or genres
- Spam detection
- Authorship identification
- Age/gender identification
- Language Identification
- Sentiment analysis
- …

Text Classification: definition

- *Input*:
  - a document d
  - a fixed set of classes $Y = \{y_1, y_2, \ldots, y_J\}$

- *Output*: a predicted class $y \in Y$

Classification Methods: Supervised Machine Learning

- *Input*
  - a document d
  - a fixed set of classes $Y = \{y_1, y_2, \ldots, y_J\}$
  - a training set of m hand-labeled documents $(d_1, y_1), \ldots., (d_m, y_m)$
- *Output*
  - a learned classifier $d \rightarrow y$

Aside: getting examples for supervised learning

- Human annotation
  - By experts or non-experts (crowdsourcing)
  - Found data

- How do we know how good a classifier is?
  - Compare classifier predictions with human annotation
  - On held out test examples
  - Evaluation metrics: accuracy, precision, recall

The 2-by-2 contingency table

	correct	not correct
selected	tp	fp
not selected	fn	tn

Precision and recall

- **Precision**: % of selected items that are correct
  Recall: % of correct items that are selected

	correct	not correct
selected	tp	fp
not selected	fn	tn

A combined measure: F

- A combined measure that assesses the P/R tradeoff is F measure (weighted harmonic mean):

$$F = \frac{1}{a\frac{1}{P} + (1 - a)\frac{1}{R}} = \frac{(b^2 + 1)PR}{b^2 P + R}$$

- People usually use balanced F1 measure
  - i.e., with β = 1 (that is, α = ½):

    $F = 2PR/(P+R)$

Linear Models
for Multiclass Classification

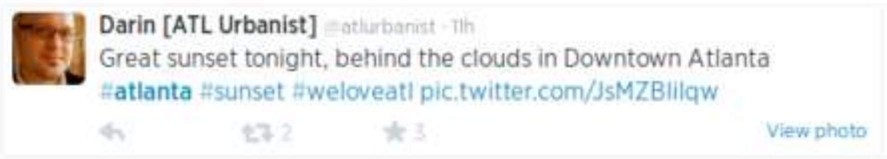
Linear Models for Classification

Feature function representation

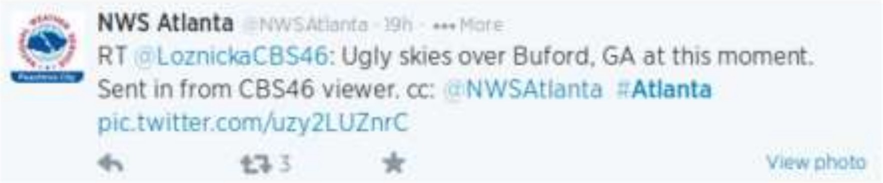
$$\hat{y} = \arg\max_{y} \boldsymbol{\theta}^{\top} \mathbf{f}(\mathbf{x}, y)$$

Weights

Defining features: Bag of words



Darin [ATL Urbanist] @atlurbanist · 11h
Great sunset tonight, behind the clouds in Downtown Atlanta #atlanta #sunset #weloveatl pic.twitter.com/JsMZBIilqw



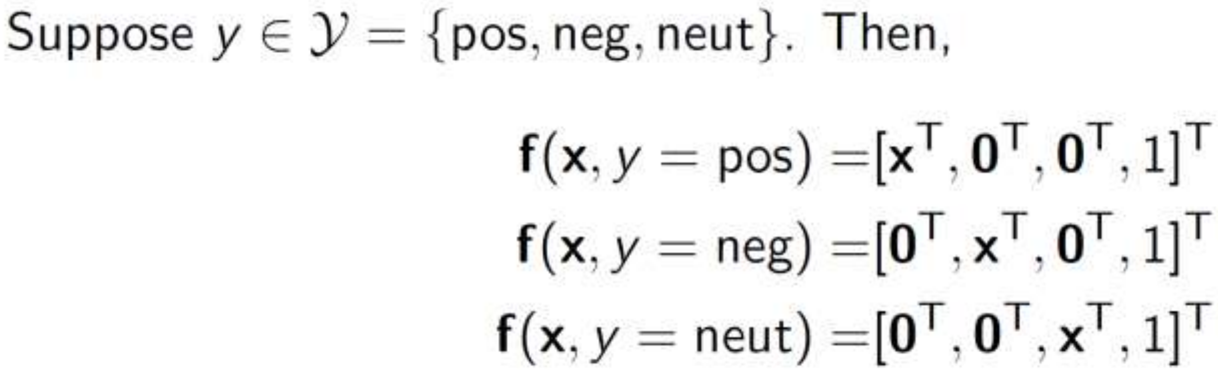
NWS Atlanta @NWSAtlanta · 19h · ••• More
RT @LoznickaCBS46: Ugly skies over Buford, GA at this moment. Sent in from CBS46 viewer. cc: @NWSAtlanta #Atlanta
pic.twitter.com/uzy2LUZnrC

$$\mathbf{w}_1 = \{\text{great}, \text{sunset}, \text{tonight}, \ldots\} \qquad \mathbf{w}_2 = \{\text{ugly}, \text{skies}, \text{buford}, \ldots\}$$

$$\begin{array}{l|cccccccccccc}
 & \text{aardvark} & \text{abacus} & \ldots & \text{behind} & \ldots & \text{buford} & \ldots & \text{clouds} & \ldots & \text{great} & \ldots & \text{ugly} & . \\
\hline
\mathbf{x}_1^{\mathsf{T}} = & 0 & 0 & 0\ldots0 & 1 & 0\ldots0 & 0 & 0\ldots0 & 0 & 0\ldots0 & 1 & 0\ldots0 & 0 & 0. \\
\mathbf{x}_2^{\mathsf{T}} = & 0 & 0 & 0\ldots0 & 0 & 0\ldots0 & 1 & 0\ldots0 & 0 & 0\ldots0 & 0 & 0\ldots0 & 1 & 0.
\end{array}$$

$$\mathbf{x}_1 = \{\text{great}:1, \text{sunset}:1, \text{tonight}:1, \ldots\}$$
$$\mathbf{x}_2 = \{\text{ugly}:1, \text{skies}:1, \text{buford}:1, \ldots\}$$

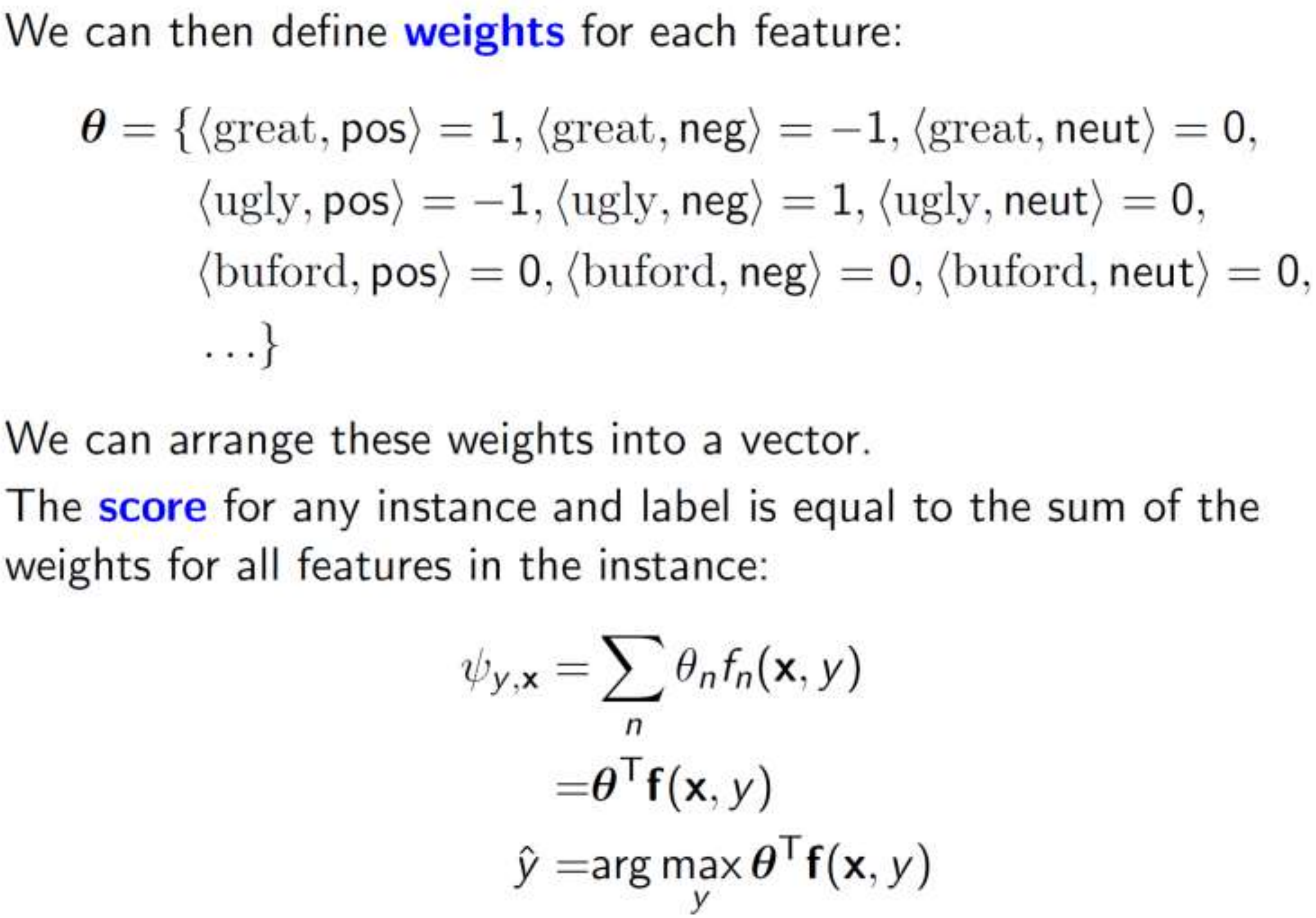
Defining features



Suppose $y \in \mathcal{Y} = \{\text{pos}, \text{neg}, \text{neut}\}$. Then,

$$\begin{aligned}
\mathbf{f}(\mathbf{x}, y = \text{pos}) =& [\mathbf{x}^\top, \mathbf{0}^\top, \mathbf{0}^\top, 1]^\top \\
\mathbf{f}(\mathbf{x}, y = \text{neg}) =& [\mathbf{0}^\top, \mathbf{x}^\top, \mathbf{0}^\top, 1]^\top \\
\mathbf{f}(\mathbf{x}, y = \text{neut}) =& [\mathbf{0}^\top, \mathbf{0}^\top, \mathbf{x}^\top, 1]^\top
\end{aligned}$$

Linear Classification



We can then define **weights** for each feature:

$$\begin{aligned}\boldsymbol{\theta} = \{&\langle \text{great}, \mathsf{pos}\rangle = 1, \langle \text{great}, \mathsf{neg}\rangle = -1, \langle \text{great}, \mathsf{neut}\rangle = 0,\\ &\langle \text{ugly}, \mathsf{pos}\rangle = -1, \langle \text{ugly}, \mathsf{neg}\rangle = 1, \langle \text{ugly}, \mathsf{neut}\rangle = 0,\\ &\langle \text{buford}, \mathsf{pos}\rangle = 0, \langle \text{buford}, \mathsf{neg}\rangle = 0, \langle \text{buford}, \mathsf{neut}\rangle = 0,\\ &\ldots\}\end{aligned}$$

We can arrange these weights into a vector.

The **score** for any instance and label is equal to the sum of the weights for all features in the instance:

$$\begin{aligned}\psi_{y,\mathbf{x}} =& \sum_n \theta_n f_n(\mathbf{x}, y)\\ =& \boldsymbol{\theta}^\mathsf{T}\mathbf{f}(\mathbf{x}, y)\\ \hat{y} =& \arg\max_y \boldsymbol{\theta}^\mathsf{T}\mathbf{f}(\mathbf{x}, y)\end{aligned}$$

Linear Models for Classification

Feature function representation

$$\hat{y} = \arg\max_{y} \boldsymbol{\theta}^{\top} \mathbf{f}(\mathbf{x}, y)$$

Weights

How can we learn weights?

- By hand

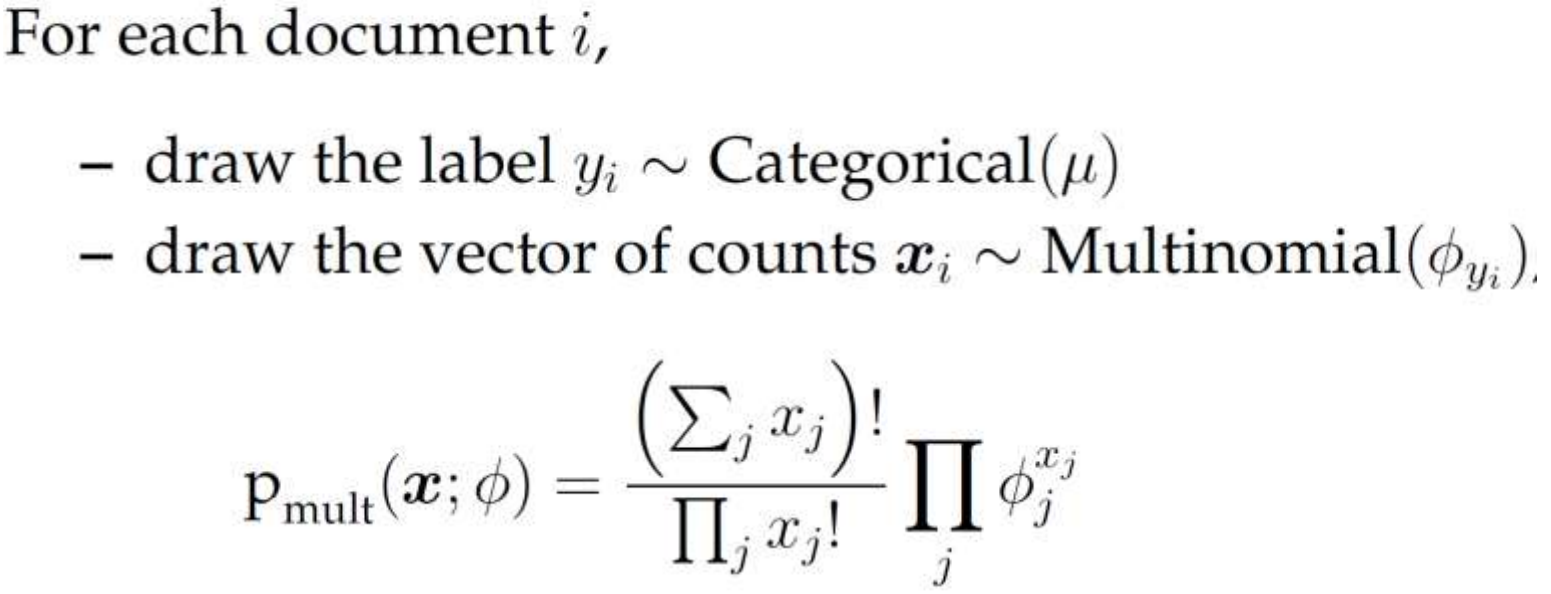
- Probability
  - e.g.,Naïve Bayes

- Discriminative training
  - e.g., perceptron, support vector machines

Naïve Bayes Models
for Text Classification

Generative Story for Multinomial Naïve Bayes

- A hypothetical stochastic process describing how training examples are generated



For each document i,

- draw the label $y_i \sim \text{Categorical}(\mu)$
- draw the vector of counts $\boldsymbol{x}_i \sim \text{Multinomial}(\phi_{y_i})$.

$$\mathrm{p}_{\text{mult}}(\boldsymbol{x};\phi) = \frac{\left(\sum_j x_j\right)!}{\prod_j x_j!} \prod_j \phi_j^{x_j}$$

Prediction with Naïve Bayes

Score(x,y) $:= \log P(\mathbf{x}, y; \phi, \mu)$

$$= \log P(\mathbf{x}|y; \phi) P(y; \mu)$$

$$= \log P(\mathbf{x}|y; \phi) + \log P(y; \mu)$$

Definition of conditional probability

Generative story assumptions

This is a linear model!

Prediction with Naïve Bayes

$$\text{Score}(x,y) := \log P(\mathbf{x}, y; \phi, \mu)$$

$$= \log P(\mathbf{x} | y; \phi) P(y; \mu)$$

Definition of conditional probability

$$= \log P(\mathbf{x} | y; \phi) + \log P(y; \mu)$$

$$= \log \text{Multinomial}(\mathbf{x}; \phi_y) + \log \text{Cat}(y; \mu)$$

Generative story assumptions

$$= \log \frac{(\sum_n x_n)!}{\prod_n x_n!} + \log \prod_n \phi_{y,n}^{x_n} + \log \mu_y$$

This is a linear model!

Prediction with Naïve Bayes

$$\text{Score(x,y)} := \log P(\mathbf{x}, y; \phi, \mu)$$

$$= \log P(\mathbf{x} | y; \phi) P(y; \mu)$$

Definition of conditional probability

$$= \log P(\mathbf{x} | y; \phi) + \log P(y; \mu)$$

$$= \log \mathrm{Multinomial}(\mathbf{x}; \phi_y) + \log \mathrm{Cat}(y; \mu)$$

Generative story assumptions

$$= \log \frac{(\sum_n x_n)!}{\prod_n x_n!} + \log \prod_n \phi_{y,n}^{x_n} + \log \mu_y$$

$$\propto \sum_n x_n \log \phi_{y,n} + \log \mu_y$$

This is a linear model!

$$= \boldsymbol{\theta}^{\top} \mathbf{f}(\mathbf{x}, y)$$

where

$$\boldsymbol{\theta} = [\log \boldsymbol{\phi}_1^{\top}, \log \mu_1, \log \boldsymbol{\phi}_2^{\top}, \log \mu_2, \ldots]^{\top}$$

$$\mathbf{f}(\mathbf{x}, y) = [\mathbf{0}, \ldots, \mathbf{0}, \mathbf{x}^{\top}, 1, \mathbf{0}, \ldots, \mathbf{0}]^{\top}$$

Parameter Estimation

- “count and normalize”
- Parameters of a multinomial distribution

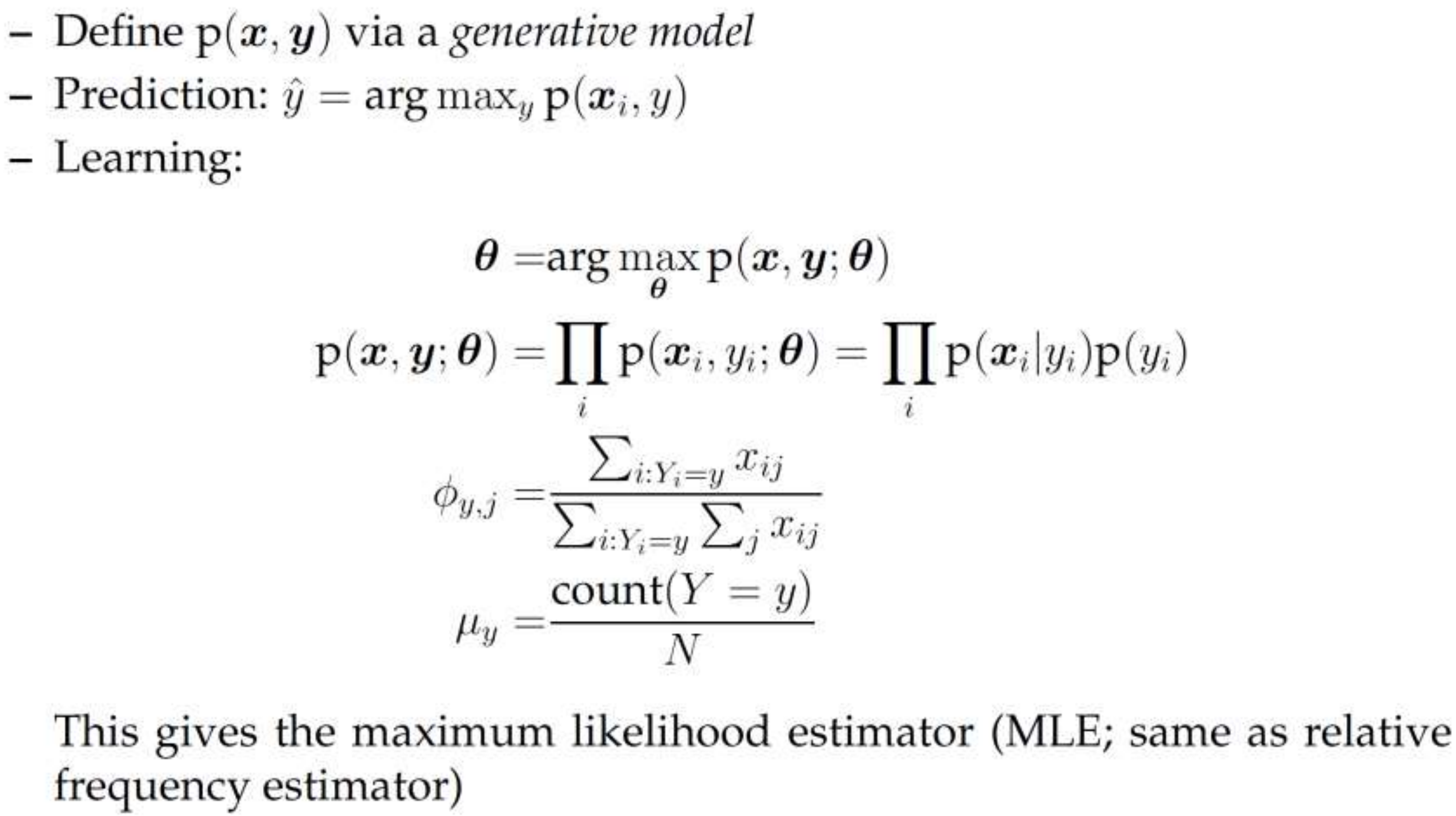
$$\phi_{y,j} = \frac{\sum_{i:Y_i=y} x_{i,j}}{\sum_{j'} \sum_{i:Y_i=y} x_{i,j'}} = \frac{\text{count}(y,j)}{\sum_{j'} \text{count}(y,j')}$$

  - Relative frequency estimator
  - Formally: this is the maximum likelihood estimate
    - See CIML for derivation

Smoothing (add alpha)

$$\phi_{y,j} = \frac{\alpha + \sum_{i:Y_i=y} x_{i,j}}{\sum_{j'=1}^{V} \left(\alpha + \sum_{i:Y_i=y} x_{i,j'}\right)} = \frac{\alpha + \textbf{count}(y,j)}{V\alpha + \sum_{j'=1}^{V} \textbf{count}(y,j')}$$

Naïve Bayes recap



- Define $\mathrm{p}(\boldsymbol{x}, \boldsymbol{y})$ via a *generative model*
- Prediction: $\hat{y} = \arg\max_y \mathrm{p}(\boldsymbol{x}_i, y)$
- Learning:

$$\boldsymbol{\theta} = \arg\max_{\boldsymbol{\theta}} \mathrm{p}(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{\theta})$$

$$\mathrm{p}(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{\theta}) = \prod_i \mathrm{p}(\boldsymbol{x}_i, y_i; \boldsymbol{\theta}) = \prod_i \mathrm{p}(\boldsymbol{x}_i | y_i)\mathrm{p}(y_i)$$

$$\phi_{y,j} = \frac{\sum_{i:Y_i=y} x_{ij}}{\sum_{i:Y_i=y} \sum_j x_{ij}}$$

$$\mu_y = \frac{\text{count}(Y = y)}{N}$$

This gives the maximum likelihood estimator (MLE; same as relative frequency estimator)

Why is this model called "Naïve Bayes"? Another view of the same model

$$
\begin{aligned}
\hat{y} &= argmax_y\, P(Y = y\, |X = x) \\
&= argmax_y P(Y = y)P(X = x\, |Y = y) \\
&= argmax_y P(Y = y) \prod_{i=1}^{d} P(X_i = x_i\, |Y = y)
\end{aligned}
$$

Bayes rule

+ Conditional independence assumption

Today

- Text classification problems
  - and their evaluation
- Linear classifiers
  - Features & Weights
  - Bag of words
  - Naïve Bayes