UNION
COLLEGE

CSC321 Data Mining & Machine Learning

Prof. Nick Webb
webbn@union.edu

Algorithms: The basic methods

- Statistical modeling
- Inferring rudimentary rules
- Constructing decision trees
- Constructing rules
- Association rule learning
- Linear models
- Instance-based learning
- Clustering

Simplicity first

- Simple algorithms often work very well!
- There are many kinds of simple structure, eg:
  - One attribute does all the work
  - All attributes contribute equally & independently
  - A weighted linear combination might do
  - Instance-based: use a few prototypes
  - Use simple logical rules
- Success of method depends on the domain

Statistical modeling

- Use all the attributes
- Two assumptions: Attributes are
  - *equally important*
  - *statistically independent* (given the class value)
    - I.e., knowing the value of one attribute says nothing about the value of another (if the class is known)
- Independence assumption is never correct!
- But … this scheme works well in practice

UNION COLLEGE

Outlook	Temp	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

UNION COLLEGE

Probabilities for weather data

Outlook			Temperature			Humidity			Windy			Play	
	Yes	*No*		*Yes*	*No*		*Yes*	*No*		*Yes*	*No*	*Yes*	*No*
Sunny	2	3	Hot	2	2	High	3	4	False	6	2	9	5
Overcast	4	0	Mild	4	2	Normal	6	1	True	3	3		
Rainy	3	2	Cool	3	1								
Sunny	2/9	3/5	Hot	2/9	2/5	High	3/9	4/5	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	Mild	4/9	2/5	Normal	6/9	1/5	True	3/9	3/5		
Rainy	3/9	2/5	Cool	3/9	1/5								

Outlook	Temp	Humidity	Windy	Play

UNION COLLEGE

Probabilities for weather data

Outlook			Temperature			Humidity			Windy			Play	
	Yes	*No*		*Yes*	*No*		*Yes*	*No*		*Yes*	*No*	*Yes*	*No*
Sunny	2	3	Hot	2	2	High	3	4	False	6	2	9	5
Overcast	4	0	Mild	4	2	Normal	6	1	True	3	3		
Rainy	3	2	Cool	3	1								
Sunny	2/9	3/5	Hot	2/9	2/5	High	3/9	4/5	False	6/9	2/5	9/14	5/14
Overcast	4/9	0/5	Mild	4/9	2/5	Normal	6/9	1/5	True	3/9	3/5		
Rainy	3/9	2/5	Cool	3/9	1/5								

Outlook	Temp.	Humidity	Windy	Play
Sunny	Cool	High	True	?

Likelihood of the two classes

For "yes" = 2/9 × 3/9 × 3/9 × 3/9 × 9/14 = 0.0053

For "no" = 3/5 × 1/5 × 4/5 × 3/5 × 5/14 = 0.0206

Conversion into a probability by normalization:

P("yes") = 0.0053 / (0.0053 + 0.0206) = 0.205

P("no") = 0.0206 / (0.0053 + 0.0206) = 0.795

UNION COLLEGE

Bayes's rule

- Probability of event *H* given evidence *E:*

$$Pr[H|E] = \frac{Pr[E|H]Pr[H]}{Pr[E]}$$

- *A priori* probability of *H* : $Pr[H]$
  - Probability of event *before* evidence is seen
- *A posteriori* probability of *H* : $Pr[H|E]$
  - Probability of event *after* evidence is seen

Thomas Bayes

Born: **1702 in London, England**
Died: **1761 in Tunbridge Wells, Kent, England**

Naïve Bayes for classification

- Classification learning: what's the probability of the class given an instance?
  - Evidence *E* = instance
  - Event *H* = class value for instance
- Naïve assumption: evidence splits into parts (i.e. attributes) that are *independent*

$$Pr[H|E] = \frac{Pr[E_1|H]Pr[E_2|H]\ldots Pr[E_n|H]Pr[H]}{Pr[E]}$$

Weather data example

Outlook	Temp.	Humidity	Windy	Play
Sunny	Cool	High	True	?

Evidence E

$$\Pr[yes|E] = \Pr[Outlook=Sunny|yes] \times \Pr[Temperature=Cool|yes] \times \Pr[Humidity=High|yes] \times \Pr[Windy=True|yes] \times \frac{\Pr[yes]}{\Pr[E]}$$

$$= \frac{\frac{2}{9}\times\frac{3}{9}\times\frac{3}{9}\times\frac{3}{9}\times\frac{9}{14}}{\Pr[E]}$$

Probability of class "yes"

The “zero-frequency problem”

- What if an attribute value doesn’t occur with every class value?
  (e.g. “Outlook = overcast” for class “no”)
  - Probability will be zero!
  - *A posteriori* probability will also be zero!
    (No matter how likely the other values are!)
- Remedy: add 1 to the count for every attribute value-class combination (*Laplace estimator)*
- Result: probabilities will never be zero!
  (also: stabilizes probability estimates)

Modified probability estimates

- In some cases adding a constant different from 1 might be more appropriate
- Example: attribute *outlook* for class *yes*

Sunny	*Overcast*	*Rainy*
$\frac{2+\mu/3}{9+\mu}$	$\frac{4+\mu/3}{9+\mu}$	$\frac{3+\mu/3}{9+\mu}$

- Weights don't need to be equal (but they must sum to 1)

$$\frac{2+\mu p_1}{9+\mu} \qquad \frac{4+\mu p_2}{9+\mu} \qquad \frac{3+\mu p_3}{9+\mu}$$

UNION COLLEGE

Missing values

- Training: instance is not included in frequency count for attribute value-class combination
- Classification: attribute will be omitted from calculation

Outlook	Temp.	Humidity	Windy	Play
?	Cool	High	True	?

Likelihood of "yes" = 3/9 × 3/9 × 3/9 × 9/14 = 0.0238

Likelihood of "no" = 1/5 × 4/5 × 3/5 × 5/14 = 0.0343

P("yes") = 0.0238 / (0.0238 + 0.0343) = 41%

P("no") = 0.0343 / (0.0238 + 0.0343) = 59%

Numeric attributes

Usual assumption: attributes have a *normal* or *Gaussian* probability distribution (given the class)

The *probability density function* for the normal distribution is defined by two parameters:

Sample mean μ

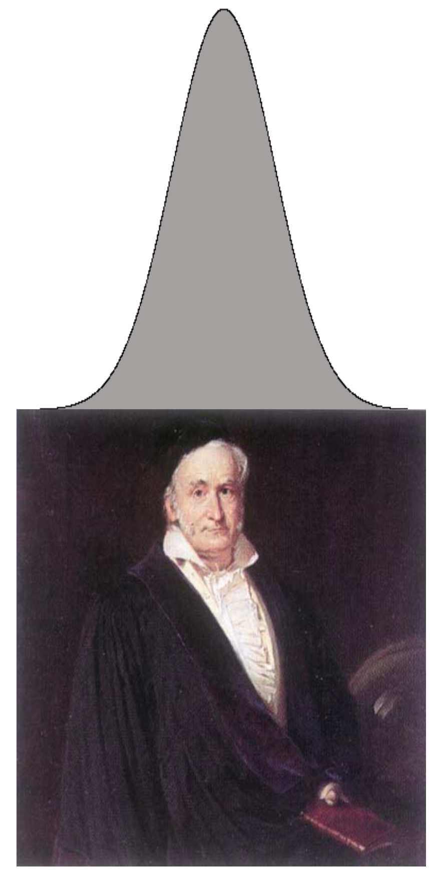
$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Standard deviation σ

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i-\mu)^2}$$

Then the density function *f(x) is*

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$



Statistics for weather data

Outlook			Temperature		Humidity		Windy			Play	
	Yes	*No*	*Yes*	*No*	*Yes*	*No*		*Yes*	*No*	*Yes*	*No*
Sunny	2	3	64, 68,	65,71,	65, 70,	70, 85,	False	6	2	9	5
Overcast	4	0	69, 70,	72,80,	70, 75,	90, 91,	True	3	3		
Rainy	3	2	72, …	85, …	80, …	95, …					
Sunny	2/9	3/5	μ =73	μ =75	μ =79	μ =86	False	6/9	2/5	9/ 14	5/ 14
Overcast	4/9	0/5	σ =6.2	σ =7.9	σ =10.2	σ =9.7	True	3/9	3/5		
Rainy	3/9	2/5									

- Example density value:

$$f(temperature=66|yes)=\frac{1}{\sqrt{2\pi}6.2}e^{-\frac{(66-73)^2}{2\cdot 6.2^2}}=0.0340$$

UNION COLLEGE

Classifying a new day

Outlook	Temp.	Humidity	Windy	Play
Sunny	66	90	true	?

Likelihood of "yes" = 2/9 × 0.0340 × 0.0221 × 3/9 × 9/14 = 0.000036

Likelihood of "no" = 3/5 × 0.0221 × 0.0381 × 3/5 × 5/14 = 0.000108

P("yes") = 0.000036 / (0.000036 + 0. 000108) = 25%

P("no") = 0.000108 / (0.000036 + 0. 000108) = 75%

Missing values during training are NOT included in calculation of mean and standard deviation

Multinomial naïve Bayes I

- Version of naïve Bayes used for document classification using *bag of words* model
- $n_1, n_2, \ldots, n_k$: number of times word *i* occurs in document
- $P_1, P_2, \ldots, P_k$: probability of obtaining word *i* when sampling from documents in class *H*
- Probability of observing document *E* given class *H* (based on *multinomial distribution*):

$$Pr[E|H] \approx N! \times \prod_{i=1}^{k} \frac{P_i^{n_i}}{n_i!}$$

- Ignores probability of generating a document of the right length (prob. assumed constant for each class)

Multinomial naïve Bayes II

- Suppose dictionary has two words, *yellow* and *blue*
- Suppose Pr[*yellow* | *H*] = 75% and Pr[*blue* | *H*] = 25%
- Suppose *E* is the document *"blue yellow blue"*
- Probability of observing document:

$$Pr[\{\text{blue yellow blue}\}|H] \approx 3! \times \frac{0.75^1}{1!} \times \frac{0.25^2}{2!} = \frac{9}{64} \approx 0.14$$

Suppose there is another class *H*' that has
Pr[*yellow* | *H'*] = 10% and Pr[*blue* | *H'*] = 90%:

$$Pr[\{\text{blue yellow blue}\}|H'] \approx 3! \times \frac{0.1^1}{1!} \times \frac{0.9^2}{2!} = 0.24$$

- Need to take prior probability of class into account to make final classification
- Factorials don't actually need to be computed
- Underflows can be prevented by using logarithms

Naïve Bayes: discussion

- Naïve Bayes works surprisingly well (even if independence assumption is clearly violated)
- Why? Because classification doesn't require accurate probability estimates *as long as maximum probability is assigned to correct class*
- However: adding too many redundant attributes will cause problems (e.g. identical attributes)
- Note also: many numeric attributes are not normally distributed (→ *kernel density estimators*)