Naïve Bayes

Naïve Bayes derivation

• Predict argmax, $P(y \mid x)$
  $= \arg \max_y P(x \mid y) P(y) / P(x)$
  $= \arg \max_y P(x \mid y) P(y)$

• Naïve independence assumption
  $P(x \mid y) = \prod_j P(x_j \mid y)$

• Predict the label $y$ maximizing
  $P(y) \prod_j P(x_j \mid y)$

• Uses generative model: pick $y$ then generate $x$ using $y$

Naïve Bayes example using max likelihood estimates (empirical counts)

• Data: (boolean)

<table>
<thead>
<tr>
<th>$x$</th>
<th>$y$</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>T,T</td>
<td>+1</td>
<td>4</td>
</tr>
<tr>
<td>T,F</td>
<td>+1</td>
<td>1</td>
</tr>
<tr>
<td>F,T</td>
<td>+1</td>
<td>4</td>
</tr>
<tr>
<td>F,F</td>
<td>+1</td>
<td>3</td>
</tr>
<tr>
<td>T,F</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>F,F</td>
<td>-1</td>
<td>2</td>
</tr>
</tbody>
</table>

• Predict on $(x_1, T, F)$ using max likelihood estimates from data

$P(y=+1) = 4/7$; $P(y=-1) = 3/7$
$P(x_1=\text{F} \mid y=+1) = 1/2$
$P(x_1=\text{T} \mid y=+1) = 2/3$
$P(x_1=\text{F} \mid y=-1) = 1/3$
$P(x_1=\text{T} \mid y=-1) = 2/3$

For "+1": $(4/7)(1/2)(2/3) = 1/14$
For "-1": $(3/7)(1/3)(2/3) = 2/21$
Predict "-1"

Naïve Bayes example using max likelihood estimates

• Data: (boolean)

<table>
<thead>
<tr>
<th>$x$</th>
<th>$y$</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>T,T</td>
<td>+1</td>
<td>4</td>
</tr>
<tr>
<td>T,F</td>
<td>+1</td>
<td>1</td>
</tr>
<tr>
<td>F,T</td>
<td>+1</td>
<td>4</td>
</tr>
<tr>
<td>F,F</td>
<td>+1</td>
<td>3</td>
</tr>
<tr>
<td>T,F</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>F,F</td>
<td>-1</td>
<td>2</td>
</tr>
</tbody>
</table>

• Predict on $(x_1, T, F)$ using max likelihood estimates from data

$P(y=+1) = 4/7$; $P(y=-1) = 3/7$
$P(x_1=\text{F} \mid y=+1) = 1/2$
$P(x_1=\text{T} \mid y=+1) = 1/4$
$P(x_1=\text{F} \mid y=-1) = 1/3$
$P(x_1=\text{T} \mid y=-1) = 2/3$

For "+1": $(4/7)(1/2)(1/4) = 1/14$
For "-1": $(3/7)(1/3)(2/3) = 2/21$
Predict "-1", even on +1 example!

Naïve Bayes discussion

• Straight from data, no searching
  – But need to estimate class conditional prob’s

• Successful applications:
  – Diagnosis,
  – Classifying text (Joachims, 1996) 89% accuracy for identifying source from 20 newsgroups (1000 documents each group, 2/3 train 1/3 test)
  – Newsweeder (Lang, 1995) interesting articles up from 16% to 59% after filtering

Newsgroup classification accuracy vs training size (Mitchell)
Naïve Bayes Issues
1. Conditional independence optimistic, but…
   Don’t have to get probabilities right, just the predictions.
2. What if an attributeValue-label pair not in training set?
   • Use Laplace estimation.
3. Numeric Features: use Gaussian or other density
   (Poisson, exponential)
4. Attributes for text classification?
   • Bag of words model

Naïve Bayes for Text
(see Mitchell’s book)
• Let V be the vocabulary (all words/
symbols in all training documents)
• For each class y, let Docs_y be the
   concatenation of all docs labeled y
• For each word w in V, let #w(Docs_y) be
   # of times w occurs in Docs_y
• Set P(w | y) to:
   (#w(docs_y) + 1) / (|V| + ∑_w #w(docs_y))

Naïve bayes for text (2)
• Predict on new document \( x \) with class \( y \)
  maximizing

\[
P(y) \prod_w P(w | y)
\]

Note: repeated words multiplied in
multiple times

Naïve Bayes for Text (2)
• For \( x \), the \( y \) maximizing: 
  \( P(y) \prod_w P(x_i | y) \)
  Also maximizes: \( \log(P(y)) + \sum_w \log(P(x_i | y)) \)
• Let \( a_i = \log(P(x_i = 1 | y)) \), \( b_i = \log(P(x_i = 0 | y)) \)
  \( \sum_w \log(P(x_i | y)) = \sum_w (a_i x_i + b_i (1-x_i)) \)
  \( = \sum_x x_i a_i + \sum b_i \)
  \( = w^\top \cdot x + c_y \)
• So predict with the class maximizing a set of linear
  functions - a LTU for two class with boolean features.

Exercise:
• Repeat slide 3 example
  using Laplacian
  probability estimates.
  Calculate the “vote” for
  each of the two classes
  for the new instance \( x = (T,F) \).
• Use Naïve Bayes in
  Weka for your iris2.arff
• Data: (boolean)
  T,T +1
  T,F +1
  F,T +1
  F,F +1
  T,F -1
  T,F -1
  F,T -1