Naïve Bayes

Naïve Bayes derivation

• Predict argmax, \( P(y \mid x) \)  
  \[ = \arg \max_y P(x \mid y) \frac{P(y)}{P(x)} \]
  
  \[ = \arg \max_y P(x \mid y) P(y) \]

• 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 Model: pick \( y \) then generate \( x \) using \( y \)

Naïve Bayes example using max likelihood estimates (empirical counts)

- Data: (boolean) 
  
  \[
  \begin{array}{c|c|c}
    x & y & \text{count} \\
    \hline
    T & T & +1 \\
    T & F & +1 \\
    F & T & +1 \\
    F & F & -1 \\
    \hline
  \end{array}
  \]

- Predict on \( x=(T,F) \) using max likelihood estimates from data 
  
  \[
  \begin{align*}
  P(y=+1) &= \frac{4}{7} \\
  P(y=-1) &= \frac{3}{7} \\
  P(x_1=T \mid y=+1) &= \frac{1}{2} \\
  P(x_2=F \mid y=+1) &= \frac{1}{4} \\
  P(x_1=T \mid y=-1) &= \frac{1}{3} \\
  P(x_2=F \mid y=-1) &= \frac{2}{3} \\
  \end{align*}
  \]

  For "+1": \( \frac{4}{7} \times \frac{1}{2} \times \frac{1}{4} = \frac{1}{14} \)

  For "-1": \( \frac{3}{7} \times \frac{1}{3} \times \frac{2}{3} = \frac{2}{21} \)

  Predict "-1".

Naïve Bayes example using max likelihood estimates

- Data: (boolean) 
  
  \[
  \begin{array}{c|c|c}
    x & y & \text{count} \\
    \hline
    T & T & +1 \\
    T & F & +1 \\
    F & T & +1 \\
    F & F & -1 \\
    \hline
  \end{array}
  \]

- Predict on \( x=(T,F) \) using max likelihood estimates from data 
  
  \[
  \begin{align*}
  P(y=+1) &= \frac{4}{7} \\
  P(y=-1) &= \frac{3}{7} \\
  P(x_1=T \mid y=+1) &= \frac{1}{2} \\
  P(x_2=F \mid y=+1) &= \frac{1}{4} \\
  P(x_1=T \mid y=-1) &= \frac{1}{3} \\
  P(x_2=F \mid y=-1) &= \frac{2}{3} \\
  \end{align*}
  \]

  For "+1": \( \frac{4}{7} \times \frac{1}{2} \times \frac{1}{4} = \frac{1}{14} \)

  For "-1": \( \frac{3}{7} \times \frac{1}{3} \times \frac{2}{3} = \frac{2}{21} \)

  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
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:
  \[ \frac{(\#w(Docs_y) + 1)}{|V| + \sum_w \#w(Docs_y))} \]

Naïve bayes for text (2)

• Predict on new document x with class y maximizing
  \[ P(y) \prod_w \text{P}(w|y) \]

Note: repeated words multiplied in multiple times

discriminant (boolean features)

• For x, the y maximizing:
  \[ P(y) \prod_x \text{P}(x|y) \]
  Also maximizes: \[ \log(P(y)) + \sum_x \log(P(x|y)) \]
  \[ = \sum_x \log(P(x|y)) = \sum_x (a_j x_j + b_j (1-x_j)) = \sum_x x_j a_j + \sum_x b_j = w \cdot x + c_y \]
  So predict with the class maximizing a set of linear functions - a LTU for two class.

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
  F,T -1