Naive Bayes

Evaluation: Precision, Recall, F-measure

The 2-by-2 contingency table

<table>
<thead>
<tr>
<th></th>
<th>correct</th>
<th>not correct</th>
</tr>
</thead>
<tbody>
<tr>
<td>selected</td>
<td>tp</td>
<td>fp</td>
</tr>
<tr>
<td>not selected</td>
<td>fn</td>
<td>tn</td>
</tr>
</tbody>
</table>
Precision and recall

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

<table>
<thead>
<tr>
<th></th>
<th>correct</th>
<th>not correct</th>
</tr>
</thead>
<tbody>
<tr>
<td>selected</td>
<td>tp</td>
<td>fp</td>
</tr>
<tr>
<td>not selected</td>
<td>fn</td>
<td>tn</td>
</tr>
</tbody>
</table>

A combined measure: F

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

\[ F = \frac{1}{\alpha \frac{1}{P} + (1 - \alpha) \frac{1}{R}} = \frac{(\beta^2 + 1)PR}{\beta^2 P + R} \]

- The harmonic mean is a conservative average
- People usually use balanced F1 measure
  - i.e., with \( \beta = 1 \) (that is, \( \alpha = \frac{1}{2} \)): \( F = \frac{2PR}{P+R} \)
Classification Methods: Review

- **Input:**
  - a document \( d \)
  - a fixed set of classes \( C = \{c_1, c_2, \ldots, c_j\} \)
  - A training set of \( m \) hand-labeled documents \((d_1, c_1), \ldots, (d_m, c_m)\)

- **Output:**
  - a (learned) classifier \( y:d \rightarrow c \)

Naïve Bayes: Review

- What type of classifier?
- Two simplifying assumptions (one specific to text classification)
- Two types of probabilities

\[
C_{NB} = \arg \max_{c \in C} P(C_j \prod_{x \in X} P(x | C))
\]

- Learning
More Than Two Classes: Sets of binary classifiers

• Dealing with any-of or multivalue classification
  • A document can belong to 0, 1, or >1 classes.

• For each class \( c \in C \)
  • Build a classifier \( y_c \) to distinguish \( c \) from all other classes \( c' \in C \)

• Given test doc \( d \),
  • Evaluate it for membership in each class using each \( y_c \)
  • \( d \) belongs to any class for which \( y_c \) returns true

More Than Two Classes: Sets of binary classifiers

• One-of or multinomial classification
  • Classes are mutually exclusive: each document in exactly one class

• For each class \( c \in C \)
  • Build a classifier \( y_c \) to distinguish \( c \) from all other classes \( c' \in C \)

• Given test doc \( d \),
  • Evaluate it for membership in each class using each \( y_c \)
  • \( d \) belongs to the one class with maximum score
Evaluation: Classic Reuters-21578 Data Set

- Most (over)used data set, 21,578 docs (each 90 types, 200 tokens)
- 9603 training, 3299 test articles (ModApte/Lewis split)
- 118 categories
  - An article can be in more than one category
  - Learn 118 binary category distinctions
- Average document (with at least one category) has 1.24 classes
- Only about 10 out of 118 categories are large

Common categories (#train, #test)
- Earn (2877, 1087)
- Acquisitions (1650, 179)
- Money-fx (538, 179)
- Grain (433, 149)
- Crude (389, 189)
- Trade (369,119)
- Interest (347, 131)
- Ship (197, 89)
- Wheat (212, 71)
- Corn (182, 56)

Reuters Text Categorization data set (Reuters-21578) document

```xml
<REUTERS TOPICS="YES" LEWISSPLIT="TRAIN" CGISPLIT="TRAINING-SET" OLDID="12981"
NEWID="798">  
</REUTERS>

<DATE> 2-MAR-1987 16:51:43.42</DATE>

<TOPICS><D>livestock</D><D>hog</D></TOPICS>

<TITLE>AMERICAN PORK CONGRESS KICKS OFF TOMORROW</TITLE>

<DATELINE> CHICAGO, March 2 - </DATELINE><BODY>The American Pork Congress kicks off tomorrow, March 3, in Indianapolis with 160 of the nations pork producers from 44 member states determining industry positions on a number of issues, according to the National Pork Producers Council, NPPC.

Delegates to the three day Congress will be considering 26 resolutions concerning various issues, including the future direction of farm policy and the tax law as it applies to the agriculture sector. The delegates will also debate whether to endorse concepts of a national PRV (pseudorabies virus) control and eradication program, the NPPC said.

A large trade show, in conjunction with the congress, will feature the latest in technology in all areas of the industry, the NPPC added. Reuter</BODY> &amp;#3;</TEXT></REUTERS>
Confusion matrix c

- For each pair of classes $<c_1, c_2>$ how many documents from $c_1$ were incorrectly assigned to $c_2$?
  - $c_{3,2}$: 90 wheat documents incorrectly assigned to poultry

<table>
<thead>
<tr>
<th>Docs in test set</th>
<th>Assigned UK</th>
<th>Assigned poultry</th>
<th>Assigned wheat</th>
<th>Assigned coffee</th>
<th>Assigned interest</th>
<th>Assigned trade</th>
</tr>
</thead>
<tbody>
<tr>
<td>True UK</td>
<td>95</td>
<td>1</td>
<td>13</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>True poultry</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>True wheat</td>
<td>10</td>
<td>90</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>True coffee</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>34</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>True interest</td>
<td>-</td>
<td>1</td>
<td>2</td>
<td>13</td>
<td>26</td>
<td>5</td>
</tr>
<tr>
<td>True trade</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>14</td>
<td>5</td>
<td>10</td>
</tr>
</tbody>
</table>

Per class evaluation measures

Recall:
Fraction of docs in class $i$ classified correctly:

$$\frac{C_{ii}}{\sum_{j} C_{ij}}$$

Precision:
Fraction of docs assigned class $i$ that are actually about class $i$:

$$\frac{C_{ii}}{\sum_{j} C_{ji}}$$

Accuracy: (1 - error rate)
Fraction of docs classified correctly:

$$\frac{\sum_{i} C_{ii}}{\sum_{j} \sum_{i} C_{ij}}$$
Micro- vs. Macro-Averaging

- If we have more than one class, how do we combine multiple performance measures into one quantity?
- **Macroaveraging**: Compute performance for each class, then average.
- **Microaveraging**: Collect decisions for all classes, compute contingency table, evaluate.

Micro- vs. Macro-Averaging: Example

<table>
<thead>
<tr>
<th>Class 1</th>
<th>Truth: yes</th>
<th>Truth: no</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classifier: yes</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Classifier: no</td>
<td>10</td>
<td>970</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class 2</th>
<th>Truth: yes</th>
<th>Truth: no</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classifier: yes</td>
<td>90</td>
<td>10</td>
</tr>
<tr>
<td>Classifier: no</td>
<td>10</td>
<td>890</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Micro Ave. Table</th>
<th>Truth: yes</th>
<th>Truth: no</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classifier: yes</td>
<td>100</td>
<td>20</td>
</tr>
<tr>
<td>Classifier: no</td>
<td>20</td>
<td>1860</td>
</tr>
</tbody>
</table>

- Macroaveraged precision:
- Microaveraged precision:
- Microaveraged score is dominated by score on common classes
Micro- vs. Macro-Averaging: Example

<table>
<thead>
<tr>
<th>Class 1</th>
<th>Class 2</th>
<th>Micro Ave. Table</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Truth: yes</td>
<td>Truth: no</td>
</tr>
<tr>
<td>Classifier: yes</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Classifier: no</td>
<td>10</td>
<td>970</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>890</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>1860</td>
</tr>
</tbody>
</table>

- Macroaveraged precision: \((0.5 + 0.9)/2 = 0.7\)
- Microaveraged precision: \(100/120 = .83\)
- Microaveraged score is dominated by score on common classes

Development Test Sets and Cross-validation

- Metric: P/R/F1 or Accuracy
- Unseen test set
  - avoid overfitting (‘tuning to the test set’)
  - more conservative estimate of performance
- Cross-validation over multiple splits
  - Handle sampling errors from different datasets
  - Pool results over each split
  - Compute pooled dev set performance
Statistical Significance

- Suppose we have two classifiers, classify1 and classify2.
- Is classify1 better? The “null hypothesis,” denoted H0, is that it isn't. But if Accuracy1 >> Accuracy2 (or whatever your evaluation metric is instead of accuracy) we are tempted to believe otherwise.
- How much larger must A1 be than A2 to reject H0?
- Frequentist view: how (im)probable is the observed difference, given H0 = true?

Text Classification

Practical Issues
The Real World

- Gee, I’m building a text classifier for real, now!
- What should I do?

No training data?
Manually written rules

If (wheat or grain) and not (whole or bread) then
Categorize as grain

- Need careful crafting
  - Human tuning on development data
  - Time-consuming: 2 days per class
Very little data?

- Use Naive Bayes
  - On Discriminative vs. Generative classifiers: A comparison of logistic regression and naive Bayes (Ng and Jordan 2002 NIPS)
- Get more labeled data
  - Find clever ways to get humans to label data for you
- Try semi-supervised machine learning methods

A reasonable amount of data?

- Try more clever classifiers
A huge amount of data?

- Can achieve high accuracy!
- At a cost (high train or test time for some methods)
- So Naive Bayes can come back into its own again!

Accuracy as a function of data size

- With enough data
  - Classifier may not matter
Underflow Prevention: log space

- Multiplying lots of probabilities can result in floating-point underflow.
- Since $\log(xy) = \log(x) + \log(y)$
  - Better to sum logs of probabilities instead of multiplying probabilities.
- Class with highest un-normalized log probability score is still most probable.

$$c_{NB} = \text{argmax}_{c_j \in C} \log P(c_j) + \sum_{i \in \text{positions}} \log P(x_i | c_j)$$

- Model is now just max of sum of weights

How to tweak performance

- Domain-specific features and weights: very important in real performance
- Sometimes need to collapse terms:
  - Part numbers, chemical formulas, ...
  - But stemming generally doesn’t help
- Upweighting: Counting a word as if it occurred twice:
  - title words (Cohen & Singer 1996)
  - first sentence of each paragraph (Murata, 1999)
  - In sentences that contain title words (Ko et al, 2002)