CS 331: Artificial Intelligence
Naïve Bayes

Thanks to Andrew Moore for some course material

Naïve Bayes

- A special type of Bayesian network
- Makes a conditional independence assumption
- Typically used for classification

Classification

Suppose you are trying to classify situations that determine whether or not Canvas will be down. You’ve come up with the following list of variables (which are all Boolean):

<table>
<thead>
<tr>
<th>Monday</th>
<th>Is a Monday</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assn</td>
<td>CS331 assignment due</td>
</tr>
<tr>
<td>Grades</td>
<td>CS331 instructor needs to enter grades</td>
</tr>
<tr>
<td>Win</td>
<td>The Beavers won the football game</td>
</tr>
</tbody>
</table>

We also have a Boolean variable called CD which stands for “Canvas down”

These entries in the CD column are called “class labels”

Classification

These are called features or attributes

This is called the “class” variable (because we’re trying to classify it)

You create a dataset out of your past experience. This is called “training data”.

You now have 2 new situations and you would like to predict if Canvas will go down. This is called “test data”.

Naïve Bayes Structure

Notice the conditional independence assumption: The features are conditionally independent given the class variable.
Naïve Bayes Parameters

\[ P(CD) = ? \]

\[ P(M \mid CD) = ? \]
\[ P(A \mid CD) = ? \]
\[ P(G \mid CD) = ? \]
\[ P(W \mid CD) = ? \]

How do you get these parameters from the training data?

\[ P(\text{CD}) = \frac{\text{false}}{\text{true}} \]
\[ P(M \mid \text{CD}) = \frac{\text{false}}{\text{true}} \]
\[ P(A \mid \text{CD}) = \frac{\text{false}}{\text{true}} \]
\[ P(G \mid \text{CD}) = \frac{\text{false}}{\text{true}} \]
\[ P(W \mid \text{CD}) = \frac{\text{false}}{\text{true}} \]

Inference in Naïve Bayes

\[ P(CD \mid M, A, G, W) \]
\[ = \frac{P(M, A, G, W \mid CD)P(CD)}{P(M, A, G, W)} \]
\[ = \alpha P(M, A, G, W \mid CD)P(CD) \]
\[ = \alpha P(CD)P(M \mid CD)P(A \mid CD)P(G \mid CD)P(W \mid CD) \]

\[ \text{By Bayes Rule} \]
\[ \text{Treat denominator as constant} \]
\[ \text{From conditional independence} \]

Prediction

- Suppose you are now in a day when M=true, A=true, G=true, W=true.
- You need to predict if CD=true or CD=false.
- We will use the notation that CD=true is equivalent to cd and CD=false is equivalent to \( \neg \text{cd} \).
Naïve Bayes Classifier

$$y^{\text{predict}} = \arg \max_v P(Y = v \mid X = u)$$

$$y^{\text{predict}} = \arg \max_v \frac{P(Y = v, X = u)}{P(X = u)}$$

$$y^{\text{predict}} = \arg \max_v \frac{P(X = u \mid Y = v) P(Y = v)}{P(X = u)}$$

$$y^{\text{predict}} = \arg \max_v P(X = u \mid Y = v) P(Y = v)$$

Because of the structure of the Bayes Net
Technical Point #1

- The probabilities $P(X_j = u_j \mid Y = v)$ can sometimes be really small.
- This can result in numerical instability since floating point numbers are not represented exactly on any computer architecture.
- To get around this, use the log of the last line in the previous slide i.e. 

$$Y^{\text{predict}} = \arg\max_v \left\{ \log(P(Y = v)) + \sum_{j=1}^{M} \log(P(X_j = u_j \mid Y = v)) \right\}$$

Technical Point #2

- When estimating parameters, what happens if you don’t have any records that match a certain combination of features?
- For example, in our training data, we didn’t have $M=false, A=false, G=false, W=false$.
- This means that $P(X_j = u_j \mid Y = v)$ in the formula below will be 0 and the entire expression will be 0.

$$P(Y = v) \prod_{j=1}^{M} P(X_j = u_j \mid Y = v)$$

Uniform Dirichlet Priors

Let $N_j$ be the number of values that $X_j$ can take on.

$$P(X_j = u_j \mid Y = v) = \frac{(#\text{records with } X_j = u_j \text{ and } Y = v) + 1}{(#\text{records with } Y = v) + N_j}$$

What happens when you have no records with $Y = v$?

$$P(X_j = u_j \mid Y = v) = \frac{1}{N_j}$$

This means that each value of $X_j$ is equally likely in the absence of data. If you have a lot of data, it dominates the $1/N_j$ value. We call this trick a “uniform Dirichlet prior”.

Example

<table>
<thead>
<tr>
<th>Monday</th>
<th>Assn</th>
<th>Grades</th>
<th>Win</th>
<th>CD</th>
</tr>
</thead>
<tbody>
<tr>
<td>true</td>
<td>true</td>
<td>true</td>
<td>false</td>
<td>true</td>
</tr>
<tr>
<td>false</td>
<td>false</td>
<td>true</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>true</td>
<td>false</td>
<td>false</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>false</td>
<td>false</td>
<td>true</td>
<td>false</td>
<td>true</td>
</tr>
<tr>
<td>true</td>
<td>false</td>
<td>true</td>
<td>false</td>
<td>true</td>
</tr>
<tr>
<td>true</td>
<td>false</td>
<td>false</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>true</td>
<td>true</td>
<td>false</td>
<td>true</td>
<td>false</td>
</tr>
</tbody>
</table>

Compute $P(M \mid CD)$ using uniform Dirichlet priors

**CW: Practice**

<table>
<thead>
<tr>
<th>Monday</th>
<th>Assn</th>
<th>Grades</th>
<th>Win</th>
<th>CD</th>
</tr>
</thead>
<tbody>
<tr>
<td>true</td>
<td>true</td>
<td>true</td>
<td>false</td>
<td>true</td>
</tr>
<tr>
<td>false</td>
<td>false</td>
<td>true</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>true</td>
<td>false</td>
<td>false</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>false</td>
<td>false</td>
<td>true</td>
<td>false</td>
<td>true</td>
</tr>
<tr>
<td>true</td>
<td>false</td>
<td>true</td>
<td>false</td>
<td>true</td>
</tr>
<tr>
<td>true</td>
<td>false</td>
<td>true</td>
<td>false</td>
<td>true</td>
</tr>
</tbody>
</table>

Compute $P(W=\text{true} \mid CD=\text{true})$ using uniform Dirichlet priors

**Programming Assignment #3**

You will classify text into two classes.

There are two files:

1. Training data: trainingSet.txt
2. Testing data: testSet.txt
Programming Assignment #3

Two parts to this assignment:
1. Pre-processing step
2. Classification step

1. Preprocessing Step

- Recall that naïve Bayes has the structure shown to the right
- The nodes correspond to random variables, which are the features or attributes in the data
- What are the features in the documents?
- Note: a “document” in our assignment is a Yelp review to be classified as positive or negative

The Vocabulary

- The features of the documents will be the presence/absence of words in the vocabulary
- The vocabulary is the list of words that are known to the classifier
- Ideally, the vocabulary would be all the words in the English language
- For this assignment, you will form the vocabulary using all the words in the training data

Bag of Words

Suppose you have the following documents:

<table>
<thead>
<tr>
<th>Training Data</th>
<th>Class Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>This is an excellent laptop</td>
<td>Class 1</td>
</tr>
<tr>
<td>No, this is not sarcasm</td>
<td>Class 0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Test Data</th>
<th>Class Label</th>
</tr>
</thead>
<tbody>
<tr>
<td>Excellent Laptop =P</td>
<td>Class 1</td>
</tr>
</tbody>
</table>

The vocabulary will be:

this, is, an, excellent, laptop, no, not, sarcasm

Bag of Words

Vocab: this, is, an, excellent, laptop, no, not, sarcasm

Keep this in alphabetical order to help with debugging

Vocab: an, excellent, is, laptop, no, not, sarcasm, this

Training data

Next, convert your training and test data into features

| Training Data | | Class Label |
|---------------|-----------------|
| an | excellent | is | laptop | no | not | sarcasm | this | Class Label |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |

| Test Data | | Class Label |
|-----------|-----------------|
| an | excellent | is | laptop | no | not | sarcasm | this | Class Label |
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |

You will output the training data in feature form, with the features alphabetized (we will grade you on this output).
2. Classification Step (Training Phase)

- Your naïve Bayes classifier now looks something like the above
- You still need to fill in the conditional probability tables in each node
- This is done in the training phase (as described on slides 9 and 10)
- Remember to use the uniform Dirichlet prior trick (see slide 21)

2. Classification Step (Testing Phase)

- Load the featurized test data
- For each document in the test data, predict its class label
- This requires computing:
  \[ P(\text{Class label} | \text{Words in document}) \]

Suppose you have the following test instance:

\[
P(\text{Class} = 1 | \text{an} = 0, \text{excellent} = 1, \text{is} = 0, \text{laptop} = 1, \text{no} = 0, \text{not} = 0, \text{sarcasm} = 0, \text{this} = 0) = \alpha \cdot P(\text{Class} = 1) \cdot P(\text{an} = 0 | \text{Class} = 1) \cdot P(\text{excellent} = 1 | \text{Class} = 1) \cdot P(\text{is} = 0 | \text{Class} = 1) \cdot P(\text{laptop} = 1 | \text{Class} = 1) \cdot P(\text{no} = 0 | \text{Class} = 1) \cdot P(\text{not} = 0 | \text{Class} = 1) \cdot P(\text{sarcasm} = 0 | \text{Class} = 1) \cdot P(\text{this} = 0 | \text{Class} = 1)
\]

Note: Use \( P(\text{Word} = 1 | \text{Class}) \) if you have a 1 for the word. Otherwise use \( P(\text{Word} = 0 | \text{Class}) \)

Then compute the following:

\[
P(\text{Class} = 0 | \text{an} = 0, \text{excellent} = 1, \text{is} = 0, \text{laptop} = 1, \text{no} = 0, \text{not} = 0, \text{sarcasm} = 0, \text{this} = 0) = \alpha \cdot P(\text{Class} = 0) \cdot P(\text{an} = 0 | \text{Class} = 0) \cdot P(\text{excellent} = 1 | \text{Class} = 0) \cdot P(\text{is} = 0 | \text{Class} = 0) \cdot P(\text{laptop} = 1 | \text{Class} = 0) \cdot P(\text{no} = 0 | \text{Class} = 0) \cdot P(\text{not} = 0 | \text{Class} = 0) \cdot P(\text{sarcasm} = 0 | \text{Class} = 0) \cdot P(\text{this} = 0 | \text{Class} = 0)
\]

Predict \( \text{Class} = 1 \) otherwise predict \( \text{Class} = 0 \)
Results

There are two sets of results we require:

1. Results #1:
   - Use trainingSet.txt for the training phase
   - Use trainingSet.txt for the testing phase
   - Report accuracy

2. Results #2:
   - Use trainingSet.txt for the training phase
   - Use testSet.txt for the testing phase
   - Report accuracy

What You Should Know

- How to learn the parameters for a Naïve Bayes model
- How to make predictions with a Naïve Bayes model
- How to implement a Naïve Bayes Model