HW5 – Perceptrons and Neural Nets has been released. When do you want it to be due?
- Tuesday August 6
- Tuesday August 13

The final project is due Wednesday August 7. If you’re doing the group R2D2 project, you should give a 5 minute presentation to describe what you did and show off your robots in action.

The final exam is in-class on Friday August 9. It will be mainly multiple choice questions similar to the quizzes, and it will cover all of the assigned readings.
Bayes’ Nets - Wrapup

Read AIMA
Chapter 14 "Probabilistic Reasoning"
(Sections 14.1, 14.2 and 14.4)
Conditional Independence

- X and Y are independent if
  \[ \forall x, y \quad P(x, y) = P(x)P(y) \quad \rightarrow \quad X \perp Y \]

- X and Y are conditionally independent given Z
  \[ \forall x, y, z \quad P(x, y|z) = P(x|z)P(y|z) \quad \rightarrow \quad X \perp Y|Z \]

- (Conditional) independence is a property of a distribution

- Example: \( Alarm \perp Fire|Smoke \)
Unconditional (absolute) independence very rare, and it doesn’t help us make inferences about other variables.

*Conditional independence* is our most basic and robust form of knowledge about uncertain environments.

\( X \perp Y \mid Z \)

\[ \forall x, y, z : P(x, y \mid z) = P(x \mid z)P(y \mid z) \]

or, equivalently, if and only if

\[ \forall x, y, z : P(x \mid z, y) = P(x \mid z) \]
Bayes Nets: Assumptions

- Assumptions we are required to make to define the Bayes net when given the graph:
  \[ P(x_i|x_1 \cdots x_{i-1}) = P(x_i|\text{parents}(X_i)) \]

- Beyond the “chain rule → Bayes net” conditional independence assumptions
  - There are often additional conditional independences
  - They can be read off the graph

- Important for modeling: understand assumptions made when choosing a Bayes net graph
D-separation: Outline
D-separation: Outline

- Study independence properties for triples
- Analyze complex cases in terms of member triples
- D-separation: a condition / algorithm for answering such queries
Recap: Causal Chains

- This configuration is a "causal chain"

- Guaranteed X independent of Z? **No!**
  - One example set of CPTs for which X is not independent of Z is sufficient to show this independence is not guaranteed.
  - Example:
    - Low pressure causes rain causes traffic, high pressure causes no rain causes no traffic
  - In numbers:
    \[
    P(+y | +x) = 1, \quad P(-y | -x) = 1, \\
    P(+z | +y) = 1, \quad P(-z | -y) = 1
    \]
Recap: Causal Chains

- This configuration is a “causal chain”

- Guaranteed X independent of Z given Y?

\[
P(z|x, y) = \frac{P(x, y, z)}{P(x, y)} = \frac{P(x)P(y|x)P(z|y)}{P(x)P(y|x)} = P(z|y)
\]

Yes!

- Evidence along the chain “blocks” the influence

\[
P(x, y, z) = P(x)P(y|x)P(z|y)
\]
Recap: Common Cause

- This configuration is a “common cause”

\[ P(x, y, z) = P(y)P(x|y)P(z|y) \]

- Guaranteed X independent of Z? No!

- One example set of CPTs for which X is not independent of Z is sufficient to show this independence is not guaranteed.

- Example:
- Project due causes both forums busy and office hours to be full

- In numbers:

\[ P(+x | +y) = 1, P(-x | -y) = 1, \]
\[ P(+z | +y) = 1, P(-z | -y) = 1 \]
Recap: Common Cause

- This configuration is a “common cause”

\[ P(x, y, z) = P(y)P(x|y)P(z|y) \]

- Guaranteed X and Z independent given Y?

\[
\begin{align*}
P(z|x, y) &= \frac{P(x, y, z)}{P(x, y)} \\
&= \frac{P(y)P(x|y)P(z|y)}{P(y)P(x|y)} \\
&= P(z|y)
\end{align*}
\]

Yes!

- Observing the cause blocks influence between effects.
Recap: Common Effect

- Last configuration: two causes of one effect (v-structures)

- Are X and Y independent?
  - Yes: the ballgame and the rain cause traffic, but they are not correlated
  - Still need to prove they must be (try it!)

- Are X and Y independent given Z?
  - No: seeing traffic puts the rain and the ballgame in competition as explanation.

- This is backwards from the other cases
  - Observing an effect activates influence between possible causes.
Are you variables in a BN independent?

- General question: in a given BN, are two variables independent (given some evidence)?

- Solution: analyze the graph

- Any complex example can be broken into repetitions of the three canonical cases
Recipe: shade evidence nodes, look for paths in the resulting graph

Attempt 1: if two nodes are connected by an undirected path not blocked by a shaded node, they are conditionally independent

Almost works, but not quite
  - Where does it break?
  - Answer: the v-structure at T doesn’t count as a link in a path unless “active”
Question: Are X and Y conditionally independent given evidence variables \{Z\}?

- Yes, if X and Y “d-separated” by Z
- Consider all (undirected) paths from X to Y
- No active paths = independence!

A path is active if each triple is active:

- Causal chain $A \rightarrow B \rightarrow C$ where B is unobserved (either direction)
- Common cause $A \leftarrow B \rightarrow C$ where B is unobserved
- Common effect (aka v-structure)
  - $A \rightarrow B \leftarrow C$ where B or one of its descendants is observed

All it takes to block a path is a single inactive segment
D-Separation

- **Query:** \( X_i \perp \!\!\!\!\!\!\perp X_j | \{X_{k_1}, \ldots, X_{k_n}\} \) ?

- **Check all (undirected!) paths between** \( X_i \) **and** \( X_j \)
  - If one or more active, then independence not guaranteed
    \( X_i \perp \!\!\!\!\!\!\perp X_j | \{X_{k_1}, \ldots, X_{k_n}\} \)
  - Otherwise (i.e. if all paths are inactive), then independence is guaranteed
    \( X_i \perp \!\!\!\!\!\!\perp X_j | \{X_{k_1}, \ldots, X_{k_n}\} \)
Example

\[ R \perp B \]
\[ R \perp B | T \]
\[ R \perp B | T' \]

Yes
Example

$L \perp T' | T$ Yes
$L \perp B$ Yes
$L \perp B | T$
$L \perp B | T'$
$L \perp B | T, R$ Yes
Structure Implications

- Given a Bayes net structure, can run d-separation algorithm to build a complete list of conditional independences that are necessarily true of the form

\[ X_i \perp X_j \mid \{X_{k_1}, \ldots, X_{k_n}\} \]

- This list determines the set of probability distributions that can be represented
Computing All Independences

Compute **ALL THE INDEPENDENCES**!
Inference

- Inference: calculating some useful quantity from a joint probability distribution

- Examples:
  - Posterior probability
    
    $$P(Q|E_1 = e_1, \ldots E_k = e_k)$$
  - Most likely explanation:
    
    $$\arg\max_q P(Q = q|E_1 = e_1 \ldots)$$
Inference by Enumeration

- **General case:**
  - **Evidence variables:** \( E_1 \ldots E_k = e_1 \ldots e_k \)
  - **Query* variable:** \( Q \)
  - **Hidden variables:** \( H_1 \ldots H_r \)

\[ \{ X_1, X_2, \ldots, X_n \} \quad \text{All variables} \]

- **We want:**
  \[ P(Q | e_1 \ldots e_k) \]

- **Step 1:** Select the entries consistent with the evidence

- **Step 2:** Sum out \( H \) to get joint of Query and evidence

\[ P(Q, e_1 \ldots e_k) = \sum_{h_1 \ldots h_r} P(Q, h_1 \ldots h_r, e_1 \ldots e_k) \]

\[ \text{Z} = \sum_q P(Q, e_1 \ldots e_k) \]

\[ P(Q | e_1 \ldots e_k) = \frac{1}{Z} P(Q, e_1 \ldots e_k) \]

- **Step 3:** Normalize

* Works fine with multiple query variables, too
Inference by Enumeration in Bayes’ Net

- Given unlimited time, inference in BNs is easy
- Reminder of inference by enumeration by example:

\[
P(B \mid +j, +m) \propto_B P(B, +j, +m) \\
= \sum_{e,a} P(B, e, a, +j, +m) \\
= \sum_{e,a} P(B)P(e)P(a \mid B, e)P(+j \mid a)P(+m \mid a)
\]

\[
= P(B)P(+e)P(+a \mid B, +e)P(+j \mid + a)P(+m \mid + a) + P(B)P(+e)P(-a \mid B, +e)P(+j \mid - a)P(+m \mid - a) \\
P(B)P(-e)P(+a \mid B, -e)P(+j \mid + a)P(+m \mid + a) + P(B)P(-e)P(-a \mid B, -e)P(+j \mid - a)P(+m \mid - a)
\]
Inference by Enumeration?

\[ P(\text{Antilock} | \text{observed variables}) = ? \]
Inference by Enumeration vs. Variable Elimination

- Why is inference by enumeration so slow?
  - You join up the whole joint distribution before you sum out the hidden variables

- Advanced technique: Variable Elimination
  - Interleave joining and marginalizing
  - Still NP-hard, but usually much faster than inference by enumeration
  - See the textbook for a description.
Naïve Bayes

Agent Testing Today!

Slides courtesy of Dan Klein and Pieter Abbeel --- University of California, Berkeley

[These slides were created by Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley. All CS188 materials are available at http://ai.berkeley.edu.]
Naïve Bayes

Read AIMA
Section 20.1
Machine Learning

- Up until now: how use a model to make optimal decisions

- Machine learning: how to acquire a model from data / experience
  - Learning parameters (e.g. probabilities)
  - Learning structure (e.g. BN graphs)
  - Learning hidden concepts (e.g. clustering)

- Today: model-based classification with Naive Bayes
Classification
Example: Spam Filter

- **Input:** an email
- **Output:** spam/ham

**Setup:**
- Get a large collection of example emails, each labeled "spam" or "ham"
- Note: someone has to hand label all this data!
- Want to learn to predict labels of new, future emails

**Features:** The attributes used to make the ham/spam decision
- Words: FREE!
- Text Patterns: $dd, CAPS
- Non-text: SenderInContacts
- ...
Example: Digit Recognition

- **Input:** images / pixel grids
- **Output:** a digit 0-9

**Setup:**
- Get a large collection of example images, each labeled with a digit
- Note: someone has to hand label all this data!
- Want to learn to predict labels of new, future digit images

**Features:** The attributes used to make the digit decision
- Pixels: (6,8)=ON
- Shape Patterns: NumComponents, AspectRatio, NumLoops
- …

0
1
2
??
Other Classification Tasks

- Classification: given inputs $x$, predict labels $y$

- Examples:
  - Spam detection (input: document, classes: spam / ham)
  - OCR (input: images, classes: characters)
  - Medical diagnosis (input: symptoms, classes: diseases)
  - Automatic essay grading (input: document, classes: grades)
  - Fraud detection (input: account activity, classes: fraud / no fraud)
  - Customer service email routing
  - ... many more

- Classification is an important commercial technology!
Model-Based Classification
Model-Based Classification

- Model-based approach
  - Build a model (e.g. Bayes’ net) where both the label and features are random variables
  - Instantiate any observed features
  - Query for the distribution of the label conditioned on the features

- Challenges
  - What structure should the BN have?
  - How should we learn its parameters?
Naïve Bayes for Digits

- Naïve Bayes: Assume all features are independent effects of the label

- Simple digit recognition version:
  - One feature (variable) $F_{ij}$ for each grid position $<i,j>$
  - Feature values are on / off, based on whether intensity is more or less than 0.5 in underlying image
  - Each input maps to a feature vector, e.g.
    \[
    \begin{pmatrix} 1 \end{pmatrix} \rightarrow \langle F_{0,0} = 0, F_{0,1} = 0, F_{0,2} = 1, F_{0,3} = 1, F_{0,4} = 0, \ldots, F_{15,15} = 0 \rangle
    \]
  - Here: lots of features, each is binary valued

- Naïve Bayes model:
  \[
  P(Y|F_{0,0} \ldots F_{15,15}) \propto P(Y) \prod_{i,j} P(F_{i,j}|Y)
  \]

- What do we need to learn?
A general Naïve Bayes model:

\[ P(Y, F_1 \ldots F_n) = P(Y) \prod_i P(F_i|Y) \]

- We only have to specify how each feature depends on the class
- Total number of parameters is \textit{linear} in number of features
- Model is very simplistic, but often works anyway
Inference for Naïve Bayes

- **Goal:** compute posterior distribution over label variable $Y$
  - **Step 1:** get joint probability of label and evidence for each label

$$P(Y, f_1 \ldots f_n) = \begin{bmatrix}
P(y_1, f_1 \ldots f_n) \\
P(y_2, f_1 \ldots f_n) \\
\vdots \\
P(y_k, f_1 \ldots f_n)
\end{bmatrix} + \begin{bmatrix}
P(y_1) \prod_i P(f_i|y_1) \\
P(y_2) \prod_i P(f_i|y_2) \\
\vdots \\
P(y_k) \prod_i P(f_i|y_k)
\end{bmatrix}
\frac{P(f_1 \ldots f_n)}{P(Y|f_1 \ldots f_n)}$$

- **Step 2:** sum to get probability of evidence
- **Step 3:** normalize by dividing Step 1 by Step 2
General Naïve Bayes

What do we need in order to use Naïve Bayes?

- Inference method (we just saw this part)
  - Start with a bunch of probabilities: $P(Y)$ and the $P(F_i|Y)$ tables
  - Use standard inference to compute $P(Y|F_1...F_n)$
  - Nothing new here

- Estimates of local conditional probability tables
  - $P(Y)$, the prior over labels
  - $P(F_i|Y)$ for each feature (evidence variable)
  - These probabilities are collectively called the parameters of the model and denoted by $\theta$
  - Up until now, we assumed these appeared by magic, but...
  - ...they typically come from training data counts: we’ll look at this soon
Example: Conditional Probabilities

$P(Y)$

<table>
<thead>
<tr>
<th></th>
<th>0.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.1</td>
</tr>
<tr>
<td>2</td>
<td>0.1</td>
</tr>
<tr>
<td>3</td>
<td>0.1</td>
</tr>
<tr>
<td>4</td>
<td>0.1</td>
</tr>
<tr>
<td>5</td>
<td>0.1</td>
</tr>
<tr>
<td>6</td>
<td>0.1</td>
</tr>
<tr>
<td>7</td>
<td>0.1</td>
</tr>
<tr>
<td>8</td>
<td>0.1</td>
</tr>
<tr>
<td>9</td>
<td>0.1</td>
</tr>
<tr>
<td>0</td>
<td>0.1</td>
</tr>
</tbody>
</table>

$P(F_{3,1} = on|Y)$

<table>
<thead>
<tr>
<th></th>
<th>0.01</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.01</td>
</tr>
<tr>
<td>2</td>
<td>0.05</td>
</tr>
<tr>
<td>3</td>
<td>0.05</td>
</tr>
<tr>
<td>4</td>
<td>0.30</td>
</tr>
<tr>
<td>5</td>
<td>0.80</td>
</tr>
<tr>
<td>6</td>
<td>0.90</td>
</tr>
<tr>
<td>7</td>
<td>0.05</td>
</tr>
<tr>
<td>8</td>
<td>0.60</td>
</tr>
<tr>
<td>9</td>
<td>0.50</td>
</tr>
<tr>
<td>0</td>
<td>0.80</td>
</tr>
</tbody>
</table>

$P(F_{5,5} = on|Y)$

<table>
<thead>
<tr>
<th></th>
<th>0.05</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.05</td>
</tr>
<tr>
<td>2</td>
<td>0.01</td>
</tr>
<tr>
<td>3</td>
<td>0.90</td>
</tr>
<tr>
<td>4</td>
<td>0.80</td>
</tr>
<tr>
<td>5</td>
<td>0.90</td>
</tr>
<tr>
<td>6</td>
<td>0.90</td>
</tr>
<tr>
<td>7</td>
<td>0.25</td>
</tr>
<tr>
<td>8</td>
<td>0.85</td>
</tr>
<tr>
<td>9</td>
<td>0.60</td>
</tr>
<tr>
<td>0</td>
<td>0.80</td>
</tr>
</tbody>
</table>
A Spam Filter

- Naïve Bayes spam filter
  - Data:
    - Collection of emails, labeled spam or ham
    - Note: someone has to hand label all this data!
    - Split into training, held-out, test sets
  - Classifiers
    - Learn on the training set
    - (Tune it on a held-out set)
    - Test it on new emails

Dear Sir.

First, I must solicit your confidence in this transaction, this is by virtue of its nature as being utterly confidencial and top secret. …

TO BE REMOVED FROM FUTURE MAILINGS, SIMPLY REPLY TO THIS MESSAGE AND PUT "REMOVE" IN THE SUBJECT.

99 MILLION EMAIL ADDRESSES FOR ONLY $99

Ok, I know this is blatantly OT but I'm beginning to go insane. Had an old Dell Dimension XPS sitting in the corner and decided to put it to use, I know it was working pre being stuck in the corner, but when I plugged it in, hit the power nothing happened.
Naïve Bayes for Text

- **Bag-of-words Naïve Bayes:**
  - Features: \( W_i \) is the word at position \( i \)
  - As before: predict label conditioned on feature variables (spam vs. ham)
  - As before: assume features are conditionally independent given label
  - New: each \( W_i \) is identically distributed

- **Generative model:**
  \[
P(Y, W_1 \ldots W_n) = P(Y) \prod_i P(W_i|Y)
\]

- **“Tied” distributions and bag-of-words**
  - Usually, each variable gets its own conditional probability distribution \( P(F|Y) \)
  - In a bag-of-words model
    - Each position is identically distributed
    - All positions share the same conditional probs \( P(W|Y) \)
    - Why make this assumption?
  - Called “bag-of-words” because model is insensitive to word order or reordering
Example: Spam Filtering

- **Model:**
  \[ P(Y, W_1 \ldots W_n) = P(Y) \prod_i P(W_i|Y) \]

- **What are the parameters?**

  \[
  \begin{array}{|c|c|}
  \hline
  P(Y) & P(W|\text{spam}) & P(W|\text{ham}) \\
  \hline
  \text{ham : 0.66} & \text{the : 0.0156} & \text{the : 0.0210} \\
  \text{spam : 0.33} & \text{to : 0.0153} & \text{to : 0.0133} \\
  & \text{and : 0.0115} & \text{of : 0.0119} \\
  & \text{of : 0.0095} & \text{2002: 0.0110} \\
  & \text{you : 0.0093} & \text{with: 0.0110} \\
  & \text{a : 0.0086} & \text{from: 0.0108} \\
  & \text{with: 0.0080} & \text{and : 0.0107} \\
  & \text{from: 0.0075} & \text{a : 0.0105} \\
  & \text{...} & \text{...} \\
  \hline
  \end{array}
  \]

- **Where do these tables come from?**
Spam Example

| Word    | P(w|spam) | P(w|ham) | Tot Spam | Tot Ham |
|---------|-----------|---------|----------|---------|
| (prior) | 0.33333   | 0.66666 | -1.1     | -0.4    |

P(spam | w) = 98.9
Training and Testing

1. 
   
   
   
   
   

2. 
   Practice Exam

3. 
   Final Exam!
Important Concepts

- **Data:** labeled instances, e.g., emails marked spam/ham
  - Training set
  - Held out set
  - Test set

- **Features:** attribute-value pairs which characterize each x

- **Experimentation cycle**
  - Learn parameters (e.g., model probabilities) on training set
  - (Tune hyperparameters on held-out set)
  - Compute accuracy of test set
  - Very important: never “peek” at the test set!

- **Evaluation**
  - Accuracy: fraction of instances predicted correctly

- **Overfitting and generalization**
  - Want a classifier which does well on test data
  - Overfitting: fitting the training data very closely, but not generalizing well
  - We’ll investigate overfitting and generalization formally in a few lectures
Generalization and Overfitting
Overfitting

Degree 15 polynomial
Example: Overfitting

\[ P(\text{features}, C = 2) \]

\[ P(C = 2) = 0.1 \]

\[ P(\text{on}|C = 2) = 0.8 \]

\[ P(\text{off}|C = 2) = 0.1 \]

\[ P(\text{on}|C = 2) = 0.01 \]

\[ P(\text{features}, C = 3) \]

\[ P(C = 3) = 0.1 \]

\[ P(\text{on}|C = 3) = 0.8 \]

\[ P(\text{on}|C = 3) = 0.9 \]

\[ P(\text{off}|C = 3) = 0.7 \]

\[ P(\text{on}|C = 3) = 0.0 \]

2 wins!!
Example: Overfitting

- Postiors determined by *relative* probabilities (odds ratios):

\[
\frac{P(W|\text{ham})}{P(W|\text{spam})} \quad \frac{P(W|\text{spam})}{P(W|\text{ham})}
\]

- south-west : inf
- nation : inf
- morally : inf
- nicely : inf
- extent : inf
- seriously : inf

- screens : inf
- minute : inf
- guaranteed : inf
- $205.00 : inf
- delivery : inf
- signature : inf

What went wrong here?
Relative frequency parameters will **overfit** the training data!
- Just because we never saw a 3 with pixel (15,15) on during training doesn’t mean we won’t see it at test time
- Unlikely that every occurrence of “minute” is 100% spam
- Unlikely that every occurrence of “seriously” is 100% ham
- What about all the words that don’t occur in the training set at all?
- In general, we can’t go around giving unseen events zero probability

As an extreme case, imagine using the entire email as the only feature
- Would get the training data perfect (if deterministic labeling)
- Wouldn’t *generalize* at all
- Just making the bag-of-words assumption gives us some generalization, but isn’t enough

To generalize better: we need to smooth or *regularize* the estimates
Parameter Estimation
Parameter Estimation

- Estimating the distribution of a random variable
- *Elicitation*: ask a human (why is this hard?)
- *Empirically*: use training data (learning!)
  - E.g.: for each outcome $x$, look at the *empirical rate* of that value:
    \[ P_{\text{ML}}(x) = \frac{\text{count}(x)}{\text{total samples}} \]
    \[ P_{\text{ML}}(r) = \frac{2}{3} \]
  - This is the estimate that maximizes the *likelihood of the data*

\[ L(x, \theta) = \prod_i P_{\theta}(x_i) \]
Relative frequencies are the maximum likelihood estimates

\[ \theta_{ML} = \arg \max_{\theta} P(X|\theta) = \arg \max_{\theta} \prod_i P_\theta(X_i) \]

\[ P_{ML}(x) = \frac{\text{count}(x)}{\text{total samples}} \]
Unseen Events
Laplace’s estimate:

- Pretend you saw every outcome once more than you actually did

\[
PLAP(x) = \frac{c(x) + 1}{\sum_x [c(x) + 1]}
\]

\[
= \frac{c(x) + 1}{N + |X|}
\]

- Can derive this estimate with Dirichlet priors
Laplace Smoothing

- **Laplace’s estimate (extended):**
  - Pretend you saw every outcome $k$ extra times

  $$P_{LAP,k}(x) = \frac{c(x) + k}{N + k|X|}$$

  - What’s Laplace with $k = 0$?
  - $k$ is the strength of the prior

- **Laplace for conditionals:**
  - Smooth each condition independently:

  $$P_{LAP,k}(x|y) = \frac{c(x, y) + k}{c(y) + k|X|}$$

  - $P_{LAP,0}(X) =$
  - $P_{LAP,1}(X) =$
  - $P_{LAP,100}(X) =$
Estimation: Linear Interpolation*

- In practice, Laplace often performs poorly for $P(X|Y)$:
  - When $|X|$ is very large
  - When $|Y|$ is very large

- Another option: linear interpolation
  - Also get the empirical $P(X)$ from the data
  - Make sure the estimate of $P(X|Y)$ isn’t too different from the empirical $P(X)$

$$P_{LIN}(x|y) = \alpha \hat{P}(x|y) + (1.0 - \alpha) \hat{P}(x)$$

- What if $\alpha$ is 0? 1?

- For even better ways to estimate parameters, take CIS 530 next semester. 😊
Real NB: Smoothing

- For real classification problems, smoothing is critical
- New odds ratios:

\[
\frac{P(W|\text{ham})}{P(W|\text{spam})} \quad \frac{P(W|\text{spam})}{P(W|\text{ham})}
\]

<table>
<thead>
<tr>
<th>helvetica</th>
<th>11.4</th>
<th>verdana</th>
<th>28.8</th>
</tr>
</thead>
<tbody>
<tr>
<td>seems</td>
<td>10.8</td>
<td>Credit</td>
<td>28.4</td>
</tr>
<tr>
<td>group</td>
<td>10.2</td>
<td>ORDER</td>
<td>27.2</td>
</tr>
<tr>
<td>ago</td>
<td>8.4</td>
<td>&lt;FONT&gt;</td>
<td>26.9</td>
</tr>
<tr>
<td>areas</td>
<td>8.3</td>
<td>money</td>
<td>26.5</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

Do these make more sense?
Now we’ve got two kinds of unknowns
- Parameters: the probabilities $P(X|Y), P(Y)$
- Hyperparameters: e.g. the amount / type of smoothing to do, $k, \alpha$

What should we learn where?
- Learn parameters from training data
- Tune hyperparameters on different data
  - Why?
- For each value of the hyperparameters, train and test on the held-out data
- Choose the best value and do a final test on the test data
Features:

- 4 Wheels!
- Larger than a Breadbox
- Made of Metal
- 100,000-mile drivetrain warranty

*Batteries Not Included*
Errors, and What to Do

Examples of errors

Dear GlobalSCAPE Customer,

GlobalSCAPE has partnered with ScanSoft to offer you the latest version of OmniPage Pro, for just $99.99* — the regular list price is $499! The most common question we've received about this offer is — Is this genuine? We would like to assure you that this offer is authorized by ScanSoft, is genuine and valid. You can get the . . .

. . . To receive your $30 Amazon.com promotional certificate, click through to
  http://www.amazon.com/apparel
and see the prominent link for the $30 offer. All details are there. We hope you enjoyed receiving this message. However, if you'd rather not receive future e-mails announcing new store launches, please click . . .
What to Do About Errors?

- Need more features—words aren’t enough!
  - Have you emailed the sender before?
  - Have 1K other people just gotten the same email?
  - Is the sending information consistent?
  - Is the email in ALL CAPS?
  - Do inline URLs point where they say they point?
  - Does the email address you by (your) name?

- Can add these information sources as new variables in the NB model

- Next class we’ll talk about classifiers which let you easily add arbitrary features more easily
Baselines

- **First step: get a baseline**
  - Baselines are very simple “straw man” procedures
  - Help determine how hard the task is
  - Help know what a “good” accuracy is

- **Weak baseline: most frequent label classifier**
  - Gives all test instances whatever label was most common in the training set
  - E.g. for spam filtering, might label everything as ham
  - Accuracy might be very high if the problem is skewed
  - E.g. calling everything “ham” gets 66%, so a classifier that gets 70% isn’t very good...

- For real research, usually use previous work as a (strong) baseline
Confidences from a Classifier

- **The confidence of a probabilistic classifier:**
  - Posterior over the top label
  
  $\text{confidence}(x) = \max_y P(y|x)$

  - Represents how sure the classifier is of the classification
  - Any probabilistic model will have confidences
  - No guarantee confidence is correct

- **Calibration**
  - Weak calibration: higher confidences mean higher accuracy
  - Strong calibration: confidence predicts accuracy rate
  - What’s the value of calibration?
Summary

- Bayes rule lets us do diagnostic queries with causal probabilities
- The naïve Bayes assumption takes all features to be independent given the class label
- We can build classifiers out of a naïve Bayes model using training data
- Smoothing estimates is important in real systems
- Classifier confidences are useful, when you can get them
Next Time: Perceptron!