Announcements

- W4 due right now
- P4 out, due Friday
- Tracking of top 20 teams begins tonight

Survey

- More comprehensive discussion on Wednesday
- Immediate actions points:
  - Graduate / Undergraduate assessment
  - Slides++
  - Two review sessions: sent request to campus
  - Camera-man tracks all activity

Today

- Naïve Bayes
  - Inference
  - Parameter estimation
  - Generalization and overfitting
  - Smoothing
- General classification concepts
  - Confidences
  - Precision-Recall

Example Classification Tasks

- In classification, we predict labels y (classes) for inputs x
- Examples:
  - Spam detection (input: document, classes: spam / ham)
  - OCR (input: images, classes: characters)
  - Medical diagnosis (input: symptoms, classes: diseases)
  - Automatic essay grader (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!

Bayes Nets for Classification

- One method of classification:
  - Use a probabilistic model!
  - Features are observed random variables \( F_i \)
  - \( Y \) is the query variable
  - Use probabilistic inference to compute most likely \( Y \)
  \[
  y = \arg \max_y P(y | f_1 \ldots f_n) .
  \]
- You already know how to do this inference
A general Naïve Bayes model:

\[ P(Y, F_1 \ldots F_n) = \prod_{Y} P(F_i | Y) \prod_{Y} P(Y) \]

- We only specify how each feature depends on the class
- Total number of parameters is linear in \( n \)

Inference for Naïve Bayes

- Goal: compute posterior over causes
  - Step 1: get joint probability of causes and evidence
    \[ P(Y, f_1 \ldots f_n) = \frac{P(Y)}{P(f_1 \ldots f_n)} \]
    \[ P(f_1 \ldots f_n) \]
  - Step 2: get probability of evidence
  - Step 3: renormalize

What do we need in order to use Naïve Bayes?

- Inference (you know this part)
  - Start with a bunch of conditionals, \( P(Y) \) and the \( P(F_i | Y) \) tables
  - Use standard inference to compute \( P(Y|F_1 \ldots F_n) \)
  - 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: we’ll look at this now

Naïve Bayes for Digits

- Simple version:
  - One feature \( F_i \) for each grid position \(<i,j>\)
  - Possible 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.
    \( \text{1} \rightarrow (F_{1,0} = 0 \ F_{0,1} = 0 \ F_{0,2} = 0 \ F_{3,3} = 1 \ F_{0,4} = 0 \ldots F_{15,15} = 0) \)
  - 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?

Examples: CPTs
Parameter Estimation

- Estimating distribution of random variables like \( X \) or \( X | Y \)
  - **Empirically:** use training data
    - For each outcome \( x \), look at the empirical rate of that value:
      \[
      \hat{p}_n(x) = \frac{\text{count}(x)}{\text{total samples}}
      \]
    - This is the estimate that maximizes the likelihood of the data
  - **Elicitation:** ask a human!
    - Usually need domain experts, and sophisticated ways of eliciting probabilities (e.g. betting games)
    - Trouble calibrating

Naïve Bayes for Text

- **Bag-of-Words Naïve Bayes:**
  - Predict unknown class label (spam vs. ham)
  - Assume evidence features (e.g. the words) are independent
  - Warning: subtly different assumptions than before!

- **Generative model**
  \[
  P(Y, W_1, \ldots, W_n) = P(Y) \prod_i P(W_i \mid Y)
  \]
- **Tied distributions and bag-of-words**
  - Usually, each variable gets its own conditional probability distribution \( P(F \mid Y) \)
    - In a bag-of-words model
      - Each position is identically distributed
        - All positions share the same conditional probs \( P(W \mid C) \)
    - Why make this assumption?

Example: Spam Filtering

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

- **What are the parameters?**

- **Naïve Bayes for Text**
  - Data:
    - Collection of emails, labeled spam or ham
    - Note: someone has to hand label all this data!
  - Split into training, held-out, test sets

Spam Example

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

\[
\log \frac{P(\text{spam} \mid w) \cdot \text{spam} \cdot \text{ham}}{P(\text{ham} \mid w) \cdot \text{ham} \cdot \text{spam}} = \text{spam score}
\]

A Spam Filter

- **Naïve Bayes spam filter**
  - **Data:**
    - Collection of emails, labeled spam or ham
    - Note: someone has to hand label all this data!
  - **Elicitation:** ask a human!
    - Usually need domain experts, and sophisticated ways of eliciting probabilities (e.g. betting games)
    - Trouble calibrating

Spam Example

- **Model:**
  \[
  P(Y, W_1, \ldots, W_n) = P(Y) \prod_i P(W_i \mid Y)
  \]
- **What are the parameters?**

- **Naïve Bayes for Text**
  - Data:
    - Collection of emails, labeled spam or ham
    - Note: someone has to hand label all this data!
  - Split into training, held-out, test sets

Spam Example

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

\[
\log \frac{P(\text{spam} \mid w) \cdot \text{spam} \cdot \text{ham}}{P(\text{ham} \mid w) \cdot \text{ham} \cdot \text{spam}} = \text{spam score}
\]
Example: Overfitting

Posteriors determined by relative probabilities (odds ratios):

- $P(W|\text{ham})$ vs $P(W|\text{spam})$

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

What went wrong here?

Generalization and Overfitting

- 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 test it at test time
  - Unlikely that every occurrence of “minute” is 100% spam
  - 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)
  - 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

Estimation: Smoothing

- Problems with maximum likelihood estimates:
  - If I flip a coin once, and it’s heads, what’s the estimate for $P(\text{heads})$?
  - What if I flip 10 times with 8 heads?
  - What if I flip 10M times with 8M heads?

- Basic idea:
  - We have some prior expectation about parameters (here, the probability of heads)
  - Given little evidence, we should skew towards our prior
  - Given a lot of evidence, we should listen to the data

Estimation: Laplace Smoothing

- Laplace’s estimate:
  - Pretend you saw every outcome once more than you actually did
  - $P_{LAP}(X) = \frac{c(X) + 1}{N + |X|}$, $P_{ML}(X) =$
  - Can derive this as a MAP estimate with Dirichlet priors (see cs281a)
Estimation: Laplace Smoothing

- Laplace’ s estimate (extended):
  - Pretend you saw every outcome \( k \) extra times
    
    \[
    P_{\text{LAP}}(X) = \frac{c(x) + k}{N + k|X|}
    \]
  - What’ s Laplace with \( k = 0 \)?
  - \( k \) is the strength of the prior
  - Laplace for conditionals:
    - Smooth each condition
      
      \[
      P_{\text{LAP}}(x|y) = \frac{c(x,y) + k}{c(y) + k|X|}
      \]

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 \( P(X) \) from the data
  - Make sure the estimate of \( P(X|Y) \) isn’ t too different from \( P(X) \)
    
    \[
    P_{\text{LIN}}(x|y) = \alpha P(x|y) + (1.0 - \alpha) P(x)
    \]
  - What if \( \alpha = 0 \)?

Real NB: Smoothing

- For real classification problems, smoothing is critical
- New odds ratios:
  - \( \frac{P(W \mid \text{ham})}{P(W \mid \text{spam})} \)
  - \( \frac{P(W \mid \text{spam})}{P(W \mid \text{ham})} \)

<table>
<thead>
<tr>
<th></th>
<th>helvetica</th>
<th>verdana</th>
<th>seena</th>
<th>group</th>
<th>areas</th>
</tr>
</thead>
<tbody>
<tr>
<td>helvetica</td>
<td>11.4</td>
<td>28.6</td>
<td>10.8</td>
<td>10.2</td>
<td>8.3</td>
</tr>
<tr>
<td>verdana</td>
<td>28.6</td>
<td>10.8</td>
<td>10.2</td>
<td>8.3</td>
<td></td>
</tr>
<tr>
<td>seena</td>
<td>10.8</td>
<td>28.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>group</td>
<td>10.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>areas</td>
<td>8.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Do these make more sense?

Tuning on Held-Out Data

- Now we’ ve got two kinds of unknowns
  - Parameters: the probabilities \( P(Y|X), P(Y) \)
  - Hyperparameters, like the amount of smoothing to do: \( k, \alpha \)
- Where to learn?
  - Learn parameters from training data
  - Must 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

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 . . .
  - Errors, and 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
    - NB models do best when the features homogeneous
    - Next class we’ ll talk about classifiers which let you easily add arbitrary features more easily
Summary Naïve Bayes Classifier

- 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

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
  - 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?

Precision vs. Recall

- Let’s say we want to classify web pages as homepages or not
  - In a test set of 1K pages, there are 3 homepages
  - Our classifier says they are all non-homepages
  - 99.7 accuracy!
  - Need new measures for rare positive events
- Precision: fraction of guessed positives which were actually positive
- Recall: fraction of actual positives which were guessed as positive
- Say we guess 5 homepages, of which 2 were actually homepages
  - Precision: 2 correct / 5 guessed = 0.4
  - Recall: 2 correct / 3 true = 0.67
- Which is more important in customer support email automation?
- Which is more important in airport face recognition?

Precision/recall tradeoff

- Often, you can trade off precision and recall
- Only works well with weakly calibrated classifiers
- To summarize the tradeoff:
  - Break-even point: precision value when \( p = r \)
  - F-measure: harmonic mean of \( p \) and \( r \):
    \[
    F_1 = \frac{2}{\frac{1}{p} + \frac{1}{r}}
    \]