Generative Models for Classification

CS6780 – Advanced Machine Learning
Spring 2019

Thorsten Joachims
Cornell University

Reading:
Murphy 3.5, 4.1, 4.2, 8.6.1
Generative vs. Conditional vs. ERM

- **Empirical Risk Minimization**
  - Find $h = \arg\min_{h \in H} Err_S(h)$ s.t. overfitting control
  - **Pro:** directly estimate decision rule
  - **Con:** need to commit to loss, input, and output before training

- **Discriminative Conditional Model**
  - Find $P(Y|X)$, then derive $h(x)$ via Bayes rule
  - **Pro:** not yet committed to loss during training
  - **Con:** need to commit to input and output before training; learning conditional distribution is harder than learning decision rule

- **Generative Model**
  - Find $P(X,Y)$, then derive $h(x)$ via Bayes rule
  - **Pro:** not yet committed to loss, input, or output during training; often computationally easy
  - **Con:** Needs to model dependencies in $X$
Bayes Decision Rule

• Assumption:
  – learning task $P(X,Y) = P(Y|X) \ P(X)$ is known

• Question:
  – Given instance $x$, how should it be classified to minimize prediction error?

• Bayes Decision Rule:

  $$h_{bayes}(\tilde{x}) = \arg\max_{y \in Y} [P(Y = y | X = \tilde{x})]$$
Example: Modeling Flu Patients

- **Data:**

<table>
<thead>
<tr>
<th>fever (h,l,n)</th>
<th>cough (y,n)</th>
<th>pukes (y,n)</th>
<th>flu?</th>
</tr>
</thead>
<tbody>
<tr>
<td>high</td>
<td>yes</td>
<td>no</td>
<td>1</td>
</tr>
<tr>
<td>high</td>
<td>no</td>
<td>yes</td>
<td>1</td>
</tr>
<tr>
<td>low</td>
<td>yes</td>
<td>no</td>
<td>-1</td>
</tr>
<tr>
<td>low</td>
<td>yes</td>
<td>yes</td>
<td>1</td>
</tr>
</tbody>
</table>

- **Approach:** One model for flu, one for not-flu.
Bayes Theorem

• It is possible to “switch” conditioning according to the following rule

• Given any two random variables $X$ and $Y$, it holds that

\[ P(Y = y | X = x) = \frac{P(X = x | Y = y)P(Y = y)}{P(X = x)} \]

• Note that

\[ P(X = x) = \sum_{y \in Y} P(X = x | Y = y)P(Y = y) \]
Naïve Bayes’ Classifier (Multivariate)

• Model for each class

\[
P(X = \bar{x} | Y = +1) = \prod_{i=1}^{N} P(X_i = x_i | Y = +1)
\]
\[
P(X = \bar{x} | Y = -1) = \prod_{i=1}^{N} P(X_i = x_i | Y = -1)
\]

• Prior probabilities

\[
P(Y = +1), P(Y = -1)
\]

• Classification rule:

\[
h_{naive}(\bar{x}) = \arg\max_{y \in \{+1,-1\}} \left\{ P(Y = y) \prod_{i=1}^{N} P(X_i = x_i | Y = y) \right\}
\]
Estimating the Parameters of NB

- Count frequencies in training data
  - \( n \): number of training examples
  - \( n_+ / n_- \): number of pos/neg examples
  - \#(X_i=x_i, y)\): number of times feature \( X_i \) takes value \( x_i \) for examples in class \( y \)
  - \( |X_i| \): number of values attribute \( X_i \) can take
- Estimating \( P(Y) \)
  - Fraction of positive / negative examples in training data
    \[
    \hat{P}(Y = +1) = \frac{n_+}{n} \quad \hat{P}(Y = -1) = \frac{n_-}{n}
    \]
- Estimating \( P(X|Y) \)
  - Maximum Likelihood Estimate
    \[
    \hat{P}(X_i = x_i| Y = y) = \frac{\#(X_i = x_i, y)}{n_y}
    \]
  - Smoothing with Laplace estimate
    \[
    \hat{P}(X_i = x_i| Y = y) = \frac{\#(X_i = x_i, y) + 1}{n_y + |X_i|}
    \]

<table>
<thead>
<tr>
<th>fever (h,l,n)</th>
<th>cough (y,n)</th>
<th>pukes (y,n)</th>
<th>flu?</th>
</tr>
</thead>
<tbody>
<tr>
<td>high</td>
<td>yes</td>
<td>no</td>
<td>1</td>
</tr>
<tr>
<td>high</td>
<td>no</td>
<td>yes</td>
<td>1</td>
</tr>
<tr>
<td>low</td>
<td>yes</td>
<td>no</td>
<td>-1</td>
</tr>
<tr>
<td>low</td>
<td>yes</td>
<td>yes</td>
<td>1</td>
</tr>
<tr>
<td>high</td>
<td>no</td>
<td>yes</td>
<td>???</td>
</tr>
</tbody>
</table>
Linear Discriminant Analysis

- Spherical Gaussian model with unit variance for each class
  \[ P(X = \hat{x}|Y = +1) \sim \exp \left( -\frac{1}{2} (\hat{x} - \mu_+)^2 \right) \]
  \[ P(X = \hat{x}|Y = -1) \sim \exp \left( -\frac{1}{2} (\hat{x} - \mu_-)^2 \right) \]

- Prior probabilities
  \[ P(Y = +1), P(Y = -1) \]

- Classification rule
  \[ h_{LDA}(\hat{x}) = \arg\max_{y \in \{+1,-1\}} \left\{ P(Y = y) \exp \left( -\frac{1}{2} (\hat{x} - \mu_y)^2 \right) \right\} \]
  \[ = \arg\max_{y \in \{+1,-1\}} \left\{ \log(P(Y = y)) - \frac{1}{2} (\hat{x} - \mu_y)^2 \right\} \]
Estimating the Parameters of LDA

- Count frequencies in training data
  - \((\tilde{x}_1, \tilde{y}_1), \ldots, (\tilde{x}_n, \tilde{y}_n) \sim P(X, Y)\): training data
  - \(n\): number of training examples
  - \(n_+ / n_-\): number of positive/negative training examples

- Estimating \(P(Y)\)
  - Fraction of pos / neg examples in training data
    \[
    \hat{P}(Y = +1) = \frac{n_+}{n} \quad \hat{P}(Y = -1) = \frac{n_-}{n}
    \]

- Estimating class means
  \[
  \hat{\mu}_+ = \frac{1}{n_+} \sum_{\{i: y_i = 1\}} \tilde{x}_i \quad \hat{\mu}_- = \frac{1}{n_-} \sum_{\{i: y_i = -1\}} \tilde{x}_i
  \]
Naïve Bayes Classifier (Multinomial)

- **Application:** Text classification \((x = (w_1, ..., w_l)\) sequence)

<table>
<thead>
<tr>
<th>Text</th>
<th>CS?</th>
</tr>
</thead>
<tbody>
<tr>
<td>((The,\ art,\ of,\ Programming))</td>
<td>+1</td>
</tr>
<tr>
<td>((Introduction,\ to,\ Calculus))</td>
<td>-1</td>
</tr>
<tr>
<td>((Introduction,\ to,\ Complexity,\ Theory))</td>
<td>+1</td>
</tr>
<tr>
<td>((Introduction,\ to,\ Programming))</td>
<td>??</td>
</tr>
</tbody>
</table>

- **Assumption**

\[
P(X = x|Y = +1) = \prod_{i=1}^{l} P(W = w_i|Y = +1)
\]

\[
P(X = x|Y = -1) = \prod_{i=1}^{l} P(W = w_i|Y = -1)
\]

- **Classification Rule**

\[
h_{naive}(x) = \arg\max_{y\in\{+1,-1\}} \left\{ P(Y = y) \prod_{i=1}^{l} P(W = w_i|Y = y) \right\}
\]
Estimating the Parameters of Multinomial Naïve Bayes

- Count frequencies in training data
  - $n$: number of training examples
  - $n_+/n_-$: number of pos/neg examples
  - #(W=w, y): number of times word $w$ occurs in examples of class $y$
  - $l_+/l_-$: total number of words in pos/neg examples
  - |V|: size of vocabulary

- Estimating $P(Y)$
  $$\hat{P}(Y = +1) = \frac{n_+}{n} \quad \hat{P}(Y = -1) = \frac{n_-}{n}$$

- Estimating $P(X|Y)$ (smoothing with Laplace estimate):
  $$\hat{P}(W = w | Y = y) = \frac{#(W = w, y) + 1}{l_y + |V|}$$

<table>
<thead>
<tr>
<th>text</th>
<th>CS?</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1 = (The, art, of, Programming)$</td>
<td>+1</td>
</tr>
<tr>
<td>$x_2 = (Introduction, to, Calculus)$</td>
<td>-1</td>
</tr>
<tr>
<td>$x_3 = (Introduction, to, Complexity, Theory)$</td>
<td>+1</td>
</tr>
<tr>
<td>$x_4 = (Introduction, to, Programming)$</td>
<td>??</td>
</tr>
</tbody>
</table>