15-381: Artificial Intelligence

Naïve Bayesian classifier
Where we are

- **Density Estimator**: Probability
  - Inputs: Today
  - Today

- **Classifier**: Predict category
  - Inputs: Today
  - Today

- **Regressor**: Predict real no.
  - Inputs: Later
  - Later

- **Predictor**: Today
  - Later
Classification

• Assume we want to teach a computer to distinguish between cats and dogs …
Classification

• Assume we want to teach a computer to distinguish between cats and dogs …

Several steps:
1. feature transformation
2. Model / classifier specification
3. Model / classifier estimation (with regularization)
4. feature selection
Classification

- Assume we want to teach a computer to distinguish between cats and dogs ...

Several steps:
1. feature transformation
2. Model / classifier specification
3. Model / classifier estimation (with regularization)
4. feature selection

How do we encode the picture? A collection of pixels? Do we use the entire image or a subset? …
Classification

• Assume we want to teach a computer to distinguish between cats and dogs …

Several steps:
1. feature transformation
2. Model / classifier specification
3. Model / classifier estimation (with regularization)
4. feature selection

What type of classifier should we use?
Classification

• Assume we want to teach a computer to distinguish between cats and dogs …

Several steps:
1. feature transformation
2. Model / classifier specification
3. Model / classifier estimation (with regularization)
4. feature selection

How do we learn the parameters of our classifier? Do we have enough examples to learn a good model?
Classification

• Assume we want to teach a computer to distinguish between cats and dogs …

Several steps:
1. feature transformation
2. Model / classifier specification
3. Model / classifier estimation (with regularization)
4. feature selection

Do we really need all the features? Can we use a smaller number and still achieve the same (or better) results?
Types of classifiers

• We can divide the large variety of classification approaches into roughly two main types

1. Generative:
   - build a generative statistical model
   - e.g., Bayesian networks

2. Discriminative
   - directly estimate a decision rule/boundary
   - e.g., decision tree
Types of classifiers

• We can divide the large variety of classification approaches into roughly two main types

  1. Generative:
     - build a generative statistical model
     - e.g., Bayesian networks

  2. Discriminative
     - directly estimate a decision rule/boundary
     - e.g., decision trees
Text classification

- Text classification (information retrieval):
  - A few labeled documents: $D_l = \{(x_1, y_1), \ldots, (x_n, y_n)\}$
  - Many unlabeled data points: $D_t = \{(x_{n+1}), \ldots, (x_{n+m})\}$

- Problem formulations:
  - Train with $D_l$
  - Classify all the unlabeled examples in $D_t$
  (there are also other ways to formulate this problem)

- Several steps:
  1. Feature transformation
  2. Model/classifier specification
  3. Model/classifier estimation (with regularization)
  4. Feature selection
Feature transformation

• We can construct $m$ (about 10,000) indicator features (basis functions) $\{\phi_i(x)\}$ for whether a word appears in the document
  - $\phi_i(x) = 1$, if word $i$ appears in document $x$; zero otherwise
• $\Phi(x) = [\phi_1(x), \ldots, \phi_m(x)]^T$ is the resulting feature vector
• For notational simplicity we will replace each document $x$ with a fixed length vector $\Phi = [\phi_1, \ldots, \phi_m]^T$, where $\phi_i = \phi_i(x)$. 
Example

Dictionary

• Washington
• Congress
...
54. McCain
55. Obama
56. Nader

\[ \phi_{54} = \phi_{54}(x) = 1 \]
\[ \phi_{55} = \phi_{55}(x) = 1 \]
\[ \phi_{56} = \phi_{56}(x) = 0 \]

Assume we would like to classify documents as election related or not.

End of Battle Centers on Turf Bush Carried

In Denver on Sunday, Senator Barack Obama appeared at a rally at Civic Center Park that drew tens of thousands of people.

By Adam Nagourney and Jeff Zeleny
Published: October 20, 2008

Senator John McCain and Senator Barack Obama are heading into the final week of the presidential campaign planning to spend nearly all their time in states that President Bush won last time, testimony to the increasingly dire position of Mr. McCain and his party as Election...
Selecting a classification model: Bayes Classifier

• We can determine which class a document belongs to by finding the class that maximizes the likelihood of the label (class) given the observed features:

$$\hat{y} = \arg \max_v p(y = v | \Phi)$$

• This can be simplified by using Bayes rule:

$$\hat{y} = \arg \max_v p(y = v | \Phi)$$

$$= \arg \max_v \frac{p(\Phi | y = v) p(y = v)}{p(\Phi)}$$

$$= \arg \max_v p(\Phi | y = v) p(y = v)$$

Our feature vector

Easier to compute (part of our generative model)
Naïve Bayes Classifier

• In order to find the class of the document we would like to find $v$ such that:

$$\hat{y} = \arg \max_v \ p(y = v \mid \Phi)$$

$$= \arg \max_v \ \frac{p(\Phi \mid y = v)p(y = v)}{p(\Phi)}$$

$$= \arg \max_v \ p(\Phi \mid y = v)p(y = v)$$

• If we assume that within each class of documents, the presence/absence of each word is independent of other words then we can use a “Naïve Bayes” model over documents and labels:

$$p(\Phi \mid y)p(y) = \prod_i p(\phi_i \mid y, i) \ p(y)$$

Binary value (whether or not word $i$ appeared)

Specific model for feature $i$
Example: Classifying Election (E) or Sports (S)

Assume we have

\[ \phi_{\text{bush}} = 1 \quad \text{P}(1 \mid E, \text{bush}) = 0.8, \, \text{P}(1 \mid S, \text{bush}) = 0.1 \]
\[ \phi_{\text{obama}} = 1 \quad \text{P}(1 \mid E, \text{obama}) = 0.9, \, \text{P}(1 \mid S, \text{obama}) = 0.05 \]
\[ \phi_{\text{mccain}} = 1 \quad \text{P}(1 \mid E, \text{mccain}) = 0.9, \, \text{P}(1 \mid S, \text{bush}) = 0.05 \]
\[ \phi_{\text{football}} = 0 \quad \text{P}(1 \mid E, \text{football}) = 0.1, \, \text{P}(1 \mid S, \text{football}) = 0.7 \]

\[
\text{P}(Y = E \mid 1,1,1,0) = 0.8 \times 0.9 \times 0.9 \times 0.9 \times \text{P}(E) \\
\text{P}(Y = S \mid 1,1,1,0) = 0.1 \times 0.05 \times 0.05 \times 0.3 \times \text{P}(S)
\]

Assuming equal priors (\(\text{P}(E) = \text{P}(S) = 0.5\)) we get:

\[
\text{P}(Y = E \mid 1,1,1,0) = 0.5832 \\
\text{P}(Y = S \mid 1,1,1,0) = 0.000075
\]
Determining the parameters

- If we have a two class problem then \(y \in \{0, 1\}\).
- Thus, we can write the individual conditional probabilities compactly as
  \[
p(\phi_i \mid y, i) = \prod_i \theta_{i|y}^{\phi_i} (1 - \theta_{i|y})^{1-\phi_i}
  \]
  where \(\theta_{i|y}\) is the conditional probability that \(\phi_i = 1\) given the class \(y\).
- We can find the parameters that maximize the likelihood of our data in a similar way to what we did in the density estimation case.
Parameter estimation

- The log likelihood can be written as:
  \[
  l(\Phi, y) = \sum_y N_i(1, y) \log(\theta_{i|y}) + N_i(0, y) \log(1 - \theta_{i|y})
  \]

- Taking the derivative w.r.t. \( \theta_{i|y} \) gives:
  \[
  \frac{\partial l(\Phi, y)}{\partial \theta_{i|y}} = \frac{N_i(1, y)}{\theta_{i|y}} - \frac{N_i(0, y)}{1 - \theta_{i|y}}
  \]
  \[
  \Rightarrow \hat{\theta}_{i|y} = \frac{N_i(1, y)}{N_i(1, y) + N_i(0, y)}
  \]

Log likelihood of training data (feature \( i \) + label)

Note: we do this independently for each feature \( i \)

\( N_i(1, y) = \# \) of documents containing word \( i \) and labeled \( y \)

\( N_i(0, y) = \# \) of documents without word \( i \) and labeled \( y \)
Specific models for Naïve Bayes classifiers: Binomial and Gaussian

- So far we assumed a binomial distribution
  - the features are generated by drawing $y_i \sim \text{Binomial}(p_1)$
- Another very useful assumption is the Gaussian generative model
- In this model we assume that the $i$’th record in the database is created using the following algorithm
  - Generate the features from a Gaussian PDF that depends on the value of $y_i$
    $$x_i \sim N(m_i, S_i).$$
MLE for Gaussian Naïve Bayes Classifier

• We need to estimate one global value and two values for each feature in each class:

• Estimate the prior probability for each class (global parameter)

• For each feature
  - Estimate its mean for each of the two classes
  - Estimate its variance for each of the two classes
Gaussian Bayes Classification

\[ P(y = v \mid x) = \frac{p(x \mid y = v)P(y = v)}{p(x)} \]

Prior for each class

\[ P(y = v \mid x) = p_v \prod_i \frac{1}{(2\pi)^{1/2}\sigma_{i,v}} \exp \left[ -\frac{(x_i - \mu_{i,v})^2}{2\sigma_{i,v}^2} \right] \]

How do we deal with that?

We compute the probability for both classes and select the class with the higher probability as our predicted class.
Here is a dataset

<table>
<thead>
<tr>
<th>age</th>
<th>employment</th>
<th>education</th>
<th>marital</th>
<th>...</th>
<th>job</th>
<th>relation</th>
<th>race</th>
<th>gender</th>
<th>hours</th>
<th>country</th>
<th>wealth</th>
</tr>
</thead>
<tbody>
<tr>
<td>39</td>
<td>State_gov</td>
<td>Bachelors</td>
<td>Never_married</td>
<td>...</td>
<td>Adm_clerical</td>
<td>Not_in_family</td>
<td>White</td>
<td>Male</td>
<td>40</td>
<td>United_States</td>
<td>poor</td>
</tr>
<tr>
<td>51</td>
<td>Self_emp_nc</td>
<td>Bachelors</td>
<td>Married</td>
<td>...</td>
<td>Exec_manager</td>
<td>Husband</td>
<td>White</td>
<td>Male</td>
<td>13</td>
<td>United_States</td>
<td>poor</td>
</tr>
<tr>
<td>39</td>
<td>Private</td>
<td>HS_grad</td>
<td>Divorced</td>
<td>...</td>
<td>Handlers_clerical</td>
<td>Not_in_family</td>
<td>White</td>
<td>Male</td>
<td>40</td>
<td>United_States</td>
<td>poor</td>
</tr>
<tr>
<td>54</td>
<td>Private</td>
<td>11th</td>
<td>Married</td>
<td>...</td>
<td>Handlers_clerical</td>
<td>Husband</td>
<td>Black</td>
<td>Male</td>
<td>40</td>
<td>United_States</td>
<td>poor</td>
</tr>
<tr>
<td>28</td>
<td>Private</td>
<td>Bachelors</td>
<td>Married</td>
<td>...</td>
<td>Prof_specialist</td>
<td>Wife</td>
<td>Black</td>
<td>Female</td>
<td>40</td>
<td>Cuba</td>
<td>poor</td>
</tr>
<tr>
<td>38</td>
<td>Private</td>
<td>Masters</td>
<td>Married</td>
<td>...</td>
<td>Exec_manager</td>
<td>Wife</td>
<td>White</td>
<td>Female</td>
<td>40</td>
<td>United_States</td>
<td>poor</td>
</tr>
<tr>
<td>50</td>
<td>Private</td>
<td>9th</td>
<td>Married_spouse</td>
<td>...</td>
<td>Other_services</td>
<td>Not_in_family</td>
<td>Black</td>
<td>Female</td>
<td>16</td>
<td>Jamaica</td>
<td>poor</td>
</tr>
<tr>
<td>52</td>
<td>Self_emp_nc</td>
<td>HS_grad</td>
<td>Married</td>
<td>...</td>
<td>Exec_manager</td>
<td>Husband</td>
<td>White</td>
<td>Male</td>
<td>45</td>
<td>United_States</td>
<td>rich</td>
</tr>
<tr>
<td>31</td>
<td>Private</td>
<td>Masters</td>
<td>Never_married</td>
<td>...</td>
<td>Prof_specialist</td>
<td>Not_in_family</td>
<td>White</td>
<td>Female</td>
<td>50</td>
<td>United_States</td>
<td>rich</td>
</tr>
<tr>
<td>42</td>
<td>Private</td>
<td>Bachelors</td>
<td>Married</td>
<td>...</td>
<td>Exec_manager</td>
<td>Husband</td>
<td>White</td>
<td>Male</td>
<td>40</td>
<td>United_States</td>
<td>rich</td>
</tr>
<tr>
<td>37</td>
<td>Private</td>
<td>Some_college</td>
<td>Married</td>
<td>...</td>
<td>Exec_manager</td>
<td>Husband</td>
<td>Black</td>
<td>Male</td>
<td>80</td>
<td>United_States</td>
<td>rich</td>
</tr>
<tr>
<td>30</td>
<td>State_gov</td>
<td>Bachelors</td>
<td>Married</td>
<td>...</td>
<td>Prof_specialist</td>
<td>Husband</td>
<td>Asian</td>
<td>Male</td>
<td>40</td>
<td>India</td>
<td>rich</td>
</tr>
<tr>
<td>24</td>
<td>Private</td>
<td>Bachelors</td>
<td>Never_married</td>
<td>...</td>
<td>Adm_clerical</td>
<td>Own_child</td>
<td>White</td>
<td>Female</td>
<td>30</td>
<td>United_States</td>
<td>poor</td>
</tr>
<tr>
<td>33</td>
<td>Private</td>
<td>Assoc_acd</td>
<td>Married</td>
<td>...</td>
<td>Sales</td>
<td>Not_in_family</td>
<td>Black</td>
<td>Male</td>
<td>50</td>
<td>United_States</td>
<td>poor</td>
</tr>
<tr>
<td>41</td>
<td>Private</td>
<td>Assoc_voc</td>
<td>Married</td>
<td>...</td>
<td>Craft_repair</td>
<td>Husband</td>
<td>Asian</td>
<td>Male</td>
<td>40</td>
<td>*MissingValue</td>
<td>rich</td>
</tr>
<tr>
<td>34</td>
<td>Private</td>
<td>7th_8th</td>
<td>Married</td>
<td>...</td>
<td>Transport_worker</td>
<td>Husband</td>
<td>Amer_Indian</td>
<td>Male</td>
<td>45</td>
<td>Mexico</td>
<td>poor</td>
</tr>
<tr>
<td>26</td>
<td>Self_emp_nc</td>
<td>HS_grad</td>
<td>Never_married</td>
<td>...</td>
<td>Farming_fish</td>
<td>Own_child</td>
<td>White</td>
<td>Male</td>
<td>35</td>
<td>United_States</td>
<td>poor</td>
</tr>
<tr>
<td>33</td>
<td>Private</td>
<td>HS_grad</td>
<td>Never_married</td>
<td>...</td>
<td>Machine_operator</td>
<td>Unmarried</td>
<td>White</td>
<td>Male</td>
<td>40</td>
<td>United_States</td>
<td>poor</td>
</tr>
<tr>
<td>38</td>
<td>Private</td>
<td>11th</td>
<td>Married</td>
<td>...</td>
<td>Sales</td>
<td>Husband</td>
<td>White</td>
<td>Male</td>
<td>50</td>
<td>United_States</td>
<td>poor</td>
</tr>
<tr>
<td>44</td>
<td>Self_emp_nc</td>
<td>Masters</td>
<td>Divorced</td>
<td>...</td>
<td>Exec_manager</td>
<td>Unmarried</td>
<td>White</td>
<td>Female</td>
<td>45</td>
<td>United_States</td>
<td>rich</td>
</tr>
<tr>
<td>41</td>
<td>Private</td>
<td>Doctorate</td>
<td>Married</td>
<td>...</td>
<td>Prof_specialist</td>
<td>Husband</td>
<td>White</td>
<td>Male</td>
<td>60</td>
<td>United_States</td>
<td>rich</td>
</tr>
</tbody>
</table>

48,000 records, 16 attributes [Kohavi 1995]
Predicting wealth from age

Distribution for the first class (poor)

Distribution for the second class (rich)

Prior distribution for wealth = poor: prior = 0.760718

Prior distribution for wealth = rich: prior = 0.239282

Distribution of age for poor: mean = 37.374, cov = 198.935

Distribution of age for rich: mean = 44.7727, cov = 111.618
Predicting wealth from age

wealth = poor
(prior = 0.760718)
1  mean  cov
   age  37.374  198.935

density
0.025
0.015
0.01
0.005

wealth = rich
(prior = 0.239282)
1  mean  cov
   age  44.7727  111.618

density
0.037
0.029
0.021
0.013

wealth values: poor rich
prob
1
0.6
0.2

age
20 30 40 50 60 70 80 90
0.2
Wealth from hours worked

- Wealth = Poor
  - Prior: 0.760718
  - Mean: 38.84
  - Covariance: 152.692

- Wealth = Rich
  - Prior: 0.239282
  - Mean: 45.4529
  - Covariance: 123.014

Wealth values:
- Poor
- Rich

Probability distribution for hours worked:
- Poor: Blue
- Rich: Red

Hours worked range from 10 to 70.
Wealth from years of education

wealth = poor
(prior = 0.760718)

1  mean  cov
edunum  9.59849  5.94225

density

wealth = rich
(prior = 0.239282)

1  mean  cov
edunum  11.6028  5.6769

density

wealth values:  poor  rich

prob

<table>
<thead>
<tr>
<th>1</th>
<th>0.6</th>
<th>0.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>11</td>
<td>13</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

edunum
age, hours $\rightarrow$ wealth

wealth = poor
(prior = 0.760718)

1

<table>
<thead>
<tr>
<th></th>
<th>mean</th>
<th>cov</th>
</tr>
</thead>
<tbody>
<tr>
<td>age</td>
<td>37.374</td>
<td>198.935</td>
</tr>
<tr>
<td>hours_worked</td>
<td>38.84</td>
<td>8.70283</td>
</tr>
</tbody>
</table>

wealth = rich
(prior = 0.239282)

1

<table>
<thead>
<tr>
<th></th>
<th>mean</th>
<th>cov</th>
</tr>
</thead>
<tbody>
<tr>
<td>age</td>
<td>44.7727</td>
<td>111.618</td>
</tr>
<tr>
<td>hours_worked</td>
<td>45.4529</td>
<td>-14.1565</td>
</tr>
</tbody>
</table>

hours_worked
age, hours → wealth

Wealth = poor
(prior = 0.760718)

1
<table>
<thead>
<tr>
<th></th>
<th>mean</th>
<th>cov</th>
</tr>
</thead>
<tbody>
<tr>
<td>age</td>
<td>37.374</td>
<td>198.935</td>
</tr>
<tr>
<td>hours_worked</td>
<td>38.84</td>
<td>128.28</td>
</tr>
</tbody>
</table>

Wealth = rich
(prior = 0.239282)

1
<table>
<thead>
<tr>
<th></th>
<th>mean</th>
<th>cov</th>
</tr>
</thead>
<tbody>
<tr>
<td>age</td>
<td>44.7727</td>
<td>111.618</td>
</tr>
<tr>
<td>hours_worked</td>
<td>45.4558</td>
<td>112.365</td>
</tr>
</tbody>
</table>
Having 2 inputs instead of one helps in two ways:
Having 2 inputs instead of one helps in two ways:
1. Combining evidence from two 1d Gaussians
2. Off-diagonal covariance distinguishes class “shape”
age, edunum $\rightarrow$ wealth

wealth = poor

(prior = 0.760718)

\[
\begin{array}{ccc}
1 & \text{mean} & \text{cov} \\
\text{age} & 37.374 & 198.935 & -1.94765 \\
\text{edunum} & 9.59849 & -1.94765 & 5.94225 \\
\end{array}
\]

wealth = rich

(prior = 0.239282)

\[
\begin{array}{ccc}
1 & \text{mean} & \text{cov} \\
\text{age} & 44.7727 & 111.618 & -0.557852 \\
\text{edunum} & 11.6028 & -0.557852 & 5.6769 \\
\end{array}
\]
age, edunum $\rightarrow$ wealth

wealth = poor
(prior = 0.760718)

<table>
<thead>
<tr>
<th></th>
<th>mean</th>
<th>cov</th>
</tr>
</thead>
<tbody>
<tr>
<td>age</td>
<td>37.374</td>
<td>-1.94765</td>
</tr>
<tr>
<td>edunum</td>
<td>9.59849</td>
<td>5.94225</td>
</tr>
</tbody>
</table>

wealth = rich
(prior = 0.239282)

<table>
<thead>
<tr>
<th></th>
<th>mean</th>
<th>cov</th>
</tr>
</thead>
<tbody>
<tr>
<td>age</td>
<td>44.7727</td>
<td>111.618</td>
</tr>
<tr>
<td>edunum</td>
<td>11.6028</td>
<td>-0.557852</td>
</tr>
</tbody>
</table>

wealh values: poor rich
# Accuracy

<table>
<thead>
<tr>
<th>Name</th>
<th>Model</th>
<th>Parameters</th>
<th>FracRight</th>
</tr>
</thead>
<tbody>
<tr>
<td>age+hours</td>
<td>bayesclass</td>
<td>density=joint submodel=gauss, gausstype=general</td>
<td>0.760452 +/- 0.00319521</td>
</tr>
<tr>
<td>age+hours+edunum</td>
<td>bayesclass</td>
<td>density=joint submodel=gauss, gausstype=general</td>
<td>0.796513 +/- 0.00542432</td>
</tr>
<tr>
<td>a+h+e+capgain</td>
<td>bayesclass</td>
<td>density=joint submodel=gauss, gausstype=general</td>
<td>0.793518 +/- 0.00319241</td>
</tr>
<tr>
<td>a+h+e+c+taxweight</td>
<td>bayesclass</td>
<td>density=joint submodel=gauss, gausstype=general</td>
<td>0.793477 +/- 0.00321524</td>
</tr>
</tbody>
</table>