NAIVE BAYES MODEL
Classification and Prediction

Lê Hồng Phương

<phuonglh@gmail.com>
Vietnam National University of Hanoi
Hanoi University of Science

August 2014
Outline

1 Introduction
   - Generative versus Discriminative Models

2 Naive Bayes Model
   - Maximum Likelihood Estimation
   - Classification Rule
   - Smoothing

3 Examples
   - Sportive Activity
   - Text Classification

4 Exercises
Recap: Classification and Regression

- **Classification problem:** classify each object $\mathbf{x} \in \mathcal{X}$ into one class $y \in \mathcal{Y} = \{1, 2, \ldots, K\}$.
- **Vector representation:** $\mathbf{x} = (x_1, x_2, \ldots, x_D)$.
  - We say that $\mathbf{x}$ has $D$ features or attributes.
  - Each component $x_j$ is called a feature value or an attribute value of $\mathbf{x}$.
  - $x_j$ can be discrete or continuous ($x_j \in \mathbb{R}$).
- **Supervised learning:** given a dataset of $N$ samples $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$, we need to build a good model/hypothesis $h(\cdot)$ to predict $y$ given $\mathbf{x}$.
- **Notation:** $\mathbf{x}_i$ is the $i$-th sample of the dataset, $x_{ij}$ is the $j$-th feature value of $\mathbf{x}_i$.
  - If $h(\mathbf{x})$ is discrete (finite), we have a **classification** problem.
  - If $h(\mathbf{x})$ is continuous (infinite) we have a **regression** problem.
Generative versus Discriminative Models

- A model is said **generative** if it can be used to generate the dataset given some hidden parameter $\theta \Rightarrow P(x, y; \theta)$.
- Bayes’s rule:
  \[
P(x, y; \theta) = P(x | y; \theta)P(y; \theta).
  \]
- So, in a generative model, we need to model the **likelihood distribution** $P(x | y)$ and the **prior distribution** $P(y)$.
- Next, we compute the **posterior distribution**:
  \[
P(y | x; \theta) = \frac{P(x | y; \theta)P(y; \theta)}{P(x; \theta)}.
  \]
- The classification rule for an object $x$ is:
  \[
  \arg \max_{y \in \mathcal{Y}} P(y | x; \theta) = \arg \max_{y \in \mathcal{Y}} P(x | y; \theta)P(y; \theta)
  \]
since $P(x; \theta)$ does not depend on $y$. 
In contrast, a **discriminative** model *directly* specifies the conditional probability of a class $y$ given $x \Rightarrow P(y|x; \theta)$.

Therefore, a discriminative model cannot generate the dataset.

The graphical representation of generative and discriminative models:

- **Generative model**
  - $Y$ → $X$ → $\theta$
- **Discriminative model**
  - $Y$ → $X$ → $\theta$

It has been shown that, in general, discriminative models give better performance than generative ones when trained on large datasets.
Generative Models

- **Discrete features:**
  - Naive Bayes model \((x_j \in \{0, 1\})\)
  - Multinominal Bayes model \((x_j \in \{0, 1, 2, \ldots, n\})\)

- **Continuous features:**
  - Gaussian Bayes model
  - GDA – Gaussian Discriminant Analysis

Some other generative models:

- HMM – Hidden Markov Model
- LDA – Latent Dirichlet Allocation
- Gaussian Mixture Model
Assumption: All features are binary ($x_j \in \{0, 1\}$) and independent.

We have

$$P(x, y; \theta) = P(x | y; \theta)P(y; \theta)$$

$$= \prod_{j=1}^{D} P(x_j | y; \theta)P(y; \theta).$$

Parameters of the model:

$$\theta_k = P(y = k), \forall k = 1, 2, \ldots, K$$

$$\theta_{j|k} = P(x_j = 1 | y = k), \forall j = 1, 2, \ldots, D; \forall k = 1, 2, \ldots, K$$

Note that $\theta_K = 1 - \sum_{k=1}^{K-1} \theta_k$, the model has $(K - 1) + DK$ parameters.
Maximum Likelihood Estimation

- The likelihood of a dataset of $N$ samples $(x_1, y_1), \ldots, (x_N, y_N)$ is

\[
L(\theta) = \prod_{i=1}^{N} P(x_i, y_i) = \prod_{i=1}^{N} \left( \prod_{j=1}^{D} P(x_j | y; \theta) P(y; \theta) \right) \\
= \prod_{i=1}^{N} \left( \prod_{j=1}^{D} \theta_{j|k} \theta_k \right).
\]

- Maximum likelihood estimation:

\[
\hat{\theta}_k = \frac{\sum_{i=1}^{N} \delta(y_i = k)}{N},
\]

\[
\hat{\theta}_{j|k} = \frac{\sum_{i=1}^{N} \delta(x_{ij} = 1, y_i = k)}{\sum_{i=1}^{N} \delta(y_i = k)},
\]

where $\delta(\cdot)$ is the identity function.
Classification Rule

- Given an object \( x \), its class is determined as

\[
y := \hat{k} = \arg\max_{k=1,2,...,K} P(y = k | x)
\]

\[
= \arg\max_k \prod_{j=1}^D \theta_{j|k} \theta_k.
\]

- If the loga function is used, we have a linear classifier:

\[
y := \hat{k} = \arg\max_{k=1,...,K} \left( \sum_{j=1}^D \log \theta_{j|k} + \log \theta_k \right).
\]
Smoothing

- We need to smooth the model to take into account the case that \( \theta_{j|k} = 0 \).
- If \( \theta_{j|k} = 0, \forall k = 1, 2, \ldots, K \) then
  \[
P(x) = \sum_{k=1}^{K} \left( \theta_k \prod_{j=1}^{D} \theta_{j|k} \right) = 0.
  \]
- So we have
  \[
P(y = k|x) = \frac{0}{0}, \quad \forall k = 1, 2, \ldots, K.
  \]
  \[\Rightarrow \text{we cannot construct a classification rule for } x.\]
Laplace smoothing:

\[
\hat{\theta}_{j|k} = \frac{\sum_{i=1}^{N} \delta(x_{ij} = 1, y_i = k) + \alpha}{\sum_{i=1}^{N} \delta(y_i = k) + K \alpha},
\]

where \( \alpha \) is a smoothing term (predefined or estimated by cross-validation).
In order to predict whether a person will play an outdoor sportive activity or not, the following weather information is used:

- ngày nắng hay mưa
- nhiệt độ nóng hay mát
- độ ẩm cao hay bình thường
- trời có gió hay không
Note that all attributes take binary values. Each object has 4 attributes ($D = 4$):

\[
\begin{align*}
    x_1 & \in \{\text{nắng, mưa}\} \equiv \{1, 0\} \\
    x_2 & \in \{\text{nóng, mát}\} \equiv \{1, 0\} \\
    x_3 & \in \{\text{cao, bình thường}\} \equiv \{1, 0\} \\
    x_4 & \in \{\text{đúng, sai}\} \equiv \{1, 0\}
\end{align*}
\]

Each object is classified into one class $y \in \{\text{có, không}\}$ ($K = 2$).
The training data is as follows \((N = 8)\):

<table>
<thead>
<tr>
<th>Ngoài trời (x_1)</th>
<th>Nhiệt độ (x_2)</th>
<th>Độ ẩm (x_3)</th>
<th>Có gió (x_4)</th>
<th>Chơi (y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>nắng</td>
<td>nóng</td>
<td>cao</td>
<td>sai</td>
<td>không</td>
</tr>
<tr>
<td>nắng</td>
<td>nóng</td>
<td>cao</td>
<td>đúng</td>
<td>không</td>
</tr>
<tr>
<td>mưa</td>
<td>nóng</td>
<td>cao</td>
<td>sai</td>
<td>có</td>
</tr>
<tr>
<td>mưa</td>
<td>mát</td>
<td>cao</td>
<td>đúng</td>
<td>có</td>
</tr>
<tr>
<td>mưa</td>
<td>mát</td>
<td>bình thường</td>
<td>sai</td>
<td>có</td>
</tr>
<tr>
<td>mưa</td>
<td>mát</td>
<td>bình thường</td>
<td>đúng</td>
<td>không</td>
</tr>
<tr>
<td>nắng</td>
<td>mát</td>
<td>bình thường</td>
<td>sai</td>
<td>có</td>
</tr>
<tr>
<td>mưa</td>
<td>nóng</td>
<td>cao</td>
<td>đúng</td>
<td>không</td>
</tr>
</tbody>
</table>

Estimate the parameters of a NB model and predict the chance of sportive activity of a day if it is defined as:

\[ \mathbf{x} = (\text{trời nắng, mát, độ ẩm cao, sai}) \]
Ta có xác suất để thời tiết là nắng nếu biết người đó có chơi thể thao hay không là:

$$\hat{\theta}_1|\text{có} = \frac{\sum_{i=1}^{8} \delta(x_{i1} = \text{nắng}, y_i = \text{có})}{\sum_{i=1}^{8} \delta(y_i = \text{có})} = \frac{1}{4} = 0.25$$

$$\hat{\theta}_1|\text{không} = \frac{\sum_{i=1}^{8} \delta(x_{i1} = \text{nắng}, y_i = \text{không})}{\sum_{i=1}^{8} \delta(y_i = \text{không})} = \frac{2}{4} = 0.5$$

Tương tự, xác suất để nhiệt độ là nóng nếu biết người đó có chơi hay không là:

$$\hat{\theta}_2|\text{có} = \frac{\sum_{i=1}^{8} \delta(x_{i2} = \text{nóng}, y_i = \text{có})}{\sum_{i=1}^{8} \delta(y_i = \text{có})} = \frac{1}{4} = 0.25$$

$$\hat{\theta}_2|\text{không} = \frac{\sum_{i=1}^{8} \delta(x_{i2} = \text{nóng}, y_i = \text{không})}{\sum_{i=1}^{8} \delta(y_i = \text{không})} = \frac{3}{4} = 0.75$$
Xác suất để độ ẩm là cao nếu biết người đó có chơi hay không là:

\[
\hat{\theta}_{3|c0} = \frac{\sum_{i=1}^{8} \delta(x_{i3} = \text{cao}, y_{i} = \text{c0})}{\sum_{i=1}^{8} \delta(y_{i} = \text{c0})} = \frac{2}{4} = 0.5
\]

\[
\hat{\theta}_{3|k0} = \frac{\sum_{i=1}^{8} \delta(x_{i3} = \text{cao}, y_{i} = \text{k0})}{\sum_{i=1}^{8} \delta(y_{i} = \text{k0})} = \frac{3}{4} = 0.75
\]

Xác suất để trời có gió nếu biết người đó có chơi hay không là:

\[
\hat{\theta}_{4|c0} = \frac{\sum_{i=1}^{8} \delta(x_{i4} = \text{dung}, y_{i} = \text{c0})}{\sum_{i=1}^{8} \delta(y_{i} = \text{c0})} = \frac{1}{4} = 0.25
\]

\[
\hat{\theta}_{4|k0} = \frac{\sum_{i=1}^{8} \delta(x_{i4} = \text{dung}, y_{i} = \text{k0})}{\sum_{i=1}^{8} \delta(y_{i} = \text{k0})} = \frac{3}{4} = 0.75
\]
Tóm lại ta có giá trị của các tham số $\theta_{j|k}$ như trong bảng sau:

| $\theta_{j|k}$ | có | không |
|----------------|----|-------|
| nắng           | 0.25 | 0.5   |
| nóng           | 0.25 | 0.75  |
| cao           | 0.5  | 0.75  |
| dùng          | 0.25 | 0.75  |
Ta có, với $x = (nắng, mát, cao, sai)$:

\[
P(y = có | x) = \frac{[0.25 \times (1 - 0.25) \times 0.5 \times (1 - 0.25)]^{\frac{1}{2}}}{P(x)} = \frac{0.0703125}{2P(x)}
\]

\[
P(y = không | x) = \frac{[0.5 \times (1 - 0.75) \times 0.75 \times (1 - 0.5)]^{\frac{1}{2}}}{P(x)} = \frac{0.046875}{2P(x)}.
\]

Từ đó, nhãn của $x$ được dự báo là $y = “có”$ vì

\[
P(y = có | x) > P(y = không | x).
\]
Text Classification

- Assigning subjects, categories, topics, or genres
- Spam detection
- Authorship identification
- Age/gender identification
- Language identification
- Sentiment analysis
- ...
Classification Methods: Hand-Coded Rules

- Rules based on combinations of words or other features.
  - spam: black-list addresses OR (“dollar” AND “have been selected”)
- Accuracy can be high if rules are carefully refined by expert.
- But building and maintaining these rules is expensive.
**Training:**

- A fixed set of classes: \( \mathcal{Y} = \{1, 2, \ldots, K\} \).
- A training set of \( N \) hand-labelled documents:
  \[
  (x_1, y_1), \ldots, (x_N, y_N)
  \]
- Output: a learned classifier \( h : \mathcal{X} \rightarrow \mathcal{Y} \).
Any kind of classifiers:

- **Naive Bayes**
- Logistic regression
- **Support vector machines**
- $k$-nearest neighbors
- ...
Consider a Naive Bayes model (multivariate Bernoulli version) for spam classification:

- Vocabulary: 'secret', 'offer', 'low', 'price', 'valued', 'customer', 'today', 'dollar', 'million', 'sports', 'is', 'for', 'play', 'healthy', 'pizza'.

- Example spam messages: “million dollar offer”, “secret offer today”, “secret is secret”

- Example normal messages: “low price for valued customer”, “play secret sports today”, “sports is healthy”, “low price pizza”.

Give the MLEs for

\[ \theta_{\text{spam}}, \theta_{\text{secret}|\text{spam}}, \theta_{\text{secret}|\text{non-spam}}, \theta_{\text{sports}|\text{non-spam}}, \theta_{\text{dollar}|\text{spam}}. \]
Exercises

- Implement the Naive Bayes model using a programming language of your choice.
  - If you implement the model in Java, you can reuse the architecture of an abstract classifier and some utility classes developed by your instructor.
- Details are found in a separate presentation.