Naïve Bayes Classifier

Eric Xing
Lecture 3, January 23, 2006

Reading: Chap. 4 CB and handouts

Classification

- Representing data:

- Choosing hypothesis class

- Learning: \( h : X \mapsto Y \)
  - \( X \) – features
  - \( Y \) – target classes
Suppose you know the following...

- Classification-specific Dist.: \( P(\mathbf{X}|Y) \)
  \[
p(\mathbf{X} | Y = 1) = p_1(\mathbf{X}; \mu_1, \Sigma_1)
  \]
  \[
p(\mathbf{X} | Y = 2) = p_2(\mathbf{X}; \mu_2, \Sigma_2)
  \]

- Class prior (i.e., "weight"): \( P(Y) \)

- This is a generative model of the data!

Optimal classification

- **Theorem:** Bayes classifier is optimal!
  
  - That is
    
    \[
    \text{error}_{\text{true}}(h_{\text{Bayes}}) \leq \text{error}_{\text{true}}(h), \quad \forall h(\mathbf{x})
    \]

- **Proof:**
  
  - Recall last lecture:

  - How to learn a Bayes classifier?
Recall Bayes Rule

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

Which is shorthand for:

\[ (\forall i, j) P(Y = y_i | X = x_j) = \frac{P(X = x_j | Y = y_i)P(Y = y_i)}{P(x_j)} \]

Equivalently:

\[ (\forall i, j) P(Y = y_i | X = x_j) = \frac{P(X = x_j | Y = y_i)P(Y = y_i)}{\sum_k P(X = x_j | Y = y_k)P(Y = y_k)} \]

Recall Bayes Rule

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

Which is shorthand for:

\[ (\forall i, j) P(Y = y_i | X = x_j) = \frac{P(X = x_j | Y = y_i)P(Y = y_i)}{P(x_j)} \]

Common abbreviation:

\[ (\forall i, j) P(y_i|x_j) = \frac{P(x_j|y_i)P(y_i)}{\sum_k P(x_j|y_k)P(y_k)} \]
Learning Bayes Classifier

- Training data:

![Training data example](image)

- Learning = estimating $P(X|Y)$, and $P(Y)$

- Classification = using Bayes rule to calculate $P(Y | X_{\text{new}})$

How hard is it to learn the optimal classifier?

- How do we represent these? How many parameters?
  - Prior, $P(Y)$:
    - Suppose $Y$ is composed of $k$ classes

  - Likelihood, $P(X|Y)$:
    - Suppose $X$ is composed of $n$ binary features

- Complex model $\rightarrow$ High variance with limited data!!!
Naïve Bayes:

\[ P(X_1 \ldots X_n | Y) = \prod_{i} P(X_i | Y) \]

assuming that \( X_i \) and \( X_j \) are conditionally independent given \( Y \), for all \( i \neq j \)

Conditional Independence

- \( X \) is conditionally independent of \( Y \) given \( Z \), if the probability distribution governing \( X \) is independent of the value of \( Y \), given the value of \( Z \)

\[ (\forall i, j, k) P(X = i | Y = j, Z = k) = P(X = i | Z = k) \]

Which we often write

\[ P(X \mid Y, Z) = P(X \mid Z) \]

- e.g.,

\[ P(\text{Thunder} \mid \text{Rain, Lightning}) = P(\text{Thunder} \mid \text{Lightning}) \]

- Equivalent to:

\[ P(X, Y \mid Z) = P(X \mid Z)P(Y \mid Z) \]
Summary

- **Bayes classifier** is the best classifier which minimizes the probability of classification error.
- Nonparametric and parametric classifier
- A nonparametric classifier does not rely on any assumption concerning the structure of the underlying density function.
- A classifier becomes the **Bayes classifier** if the density estimates converge to the true densities
  - when an infinite number of samples are used
  - The resulting error is the **Bayes error**, the smallest achievable error given the underlying distributions.

The Naïve Bayes assumption

- Naïve Bayes assumption:
  - Features are independent given class:
    \[
P(X_1, X_2|Y) = P(X_1|X_2, Y)P(X_2|Y) = P(X_1|Y)P(X_2|Y)
    \]
  - More generally:
    \[
P(X_1 \ldots X_n|Y) = \prod_i P(X_i|Y)
    \]
- How many parameters now?
  - Suppose X is composed of n binary features
The Naïve Bayes Classifier

- Given:
  - Prior \( P(Y) \)
  - \( n \) conditionally independent features \( X \) given the class \( Y \)
  - For each \( X_i \), we have likelihood \( P(X_i|Y) \)

- Decision rule:
  \[
  y^* = h_{ND}(x) = \arg \max_y P(y)P(x_1, \ldots, x_n | y) \\
  = \arg \max_y P(y) \prod_i P(x_i | y)
  \]

- If assumption holds, NB is optimal classifier!

Naïve Bayes Algorithm

- Train Naïve Bayes (examples)
  - for each* value \( y_k \)
  - estimate \( \pi_k \equiv P(Y = y_k) \)
  - for each* value \( x_{ij} \) of each attribute \( X_i \)
  - estimate \( \theta_{ijk} \equiv P(X_i = x_{ij} | Y = y_k) \)

- Classify \( (X_{\text{new}}) \)
  \[
  Y_{\text{new}} \leftarrow \arg \max_{y_k} P(Y = y_k) \prod_i P(X_i = x_{ij} | Y = y_k)
  \]
  \[
  Y_{\text{new}} \leftarrow \arg \max_{y_k} \pi_k \prod_i \theta_{ijk}
  \]

* probabilities must sum to 1, so need estimate only \( n-1 \) parameters...
Learning NB: parameter estimation

- Maximum Likelihood Estimate (MLE): choose $\theta$ that maximizes probability of observed data $D$
  $$\hat{\theta} = \arg\max_{\theta} P(D|\theta)$$

- Maximum a Posteriori (MAP) estimate: choose $\theta$ that is most probable given prior probability and the data
  $$\hat{\theta} = \arg\max_{\theta} P(\theta|D)$$
  $$\hat{\theta} = \arg\max_{\theta} \frac{P(D|\theta)P(\theta)}{P(D)}$$

MLE for the parameters of NB

Discrete features:

- Maximum likelihood estimates (MLE’s): $\hat{\theta} = \arg\max_{\theta} P(D|\theta)$
- Given dataset
  - $\text{Count}(A=a, B=b) \leftarrow$ number of examples where $A=a$ and $B=b$
Subtleties of NB classifier 1 – Violating the NB assumption

- Often the $X_i$ are not really conditionally independent

- We use Naïve Bayes in many cases anyway, and it often works pretty well
  - often the right classification, even when not the right probability (see [Domingos&Pazzani, 1996])
  - But the resulting probabilities $P(Y|X_{new})$ are biased toward 1 or 0 (why?)

Subtleties of NB classifier 2 – Insufficient training data

- What if you never see a training instance where $X_{1000}=a$ when $Y=b$?
  - e.g., $Y=$(SpamEmail or not), $X_{1000}=\text{`Rolex'\}$
  - $P(X_{1000}=T \mid Y=T) = 0$

- Thus, no matter what the values $X_2,...,X_n$ take:
  - $P(Y=T \mid X_1,X_2,...,X_{1000}=T, \cdots, X_n) = 0$

- What now???
MAP for the parameters of NB

Discrete features:

- Maximum **a Posteriori** (MAP) estimate: (MAP’s):

  $$\hat{\theta} = \arg \max_{\theta} \frac{P(D|\theta)P(\theta)}{P(D)}$$

- Given prior:
  - Consider binary feature
  - $\theta$ is a Bernoulli rate

  $$P(\theta, \alpha_f, \alpha_r) = \frac{\Gamma(\alpha_f + \alpha_r)}{\Gamma(\alpha_f)\Gamma(\alpha_r)} \theta^{\alpha_f-1}(1-\theta)^{\alpha_r-1} = \frac{\theta^{\alpha_f-1}(1-\theta)^{\alpha_r-1}}{B(\alpha_f, \alpha_r)}$$

- Let $\beta_a = \text{Count}(X=a)$ ← number of examples where $X=a$

  $$P(\theta | D) = \frac{\theta^{\beta_f+\alpha_f-1}(1-\theta)^{\beta_r+\alpha_r-1}}{B(\beta_f+\alpha_f, \beta_r+\alpha_r)} \sim \text{Beta}(\beta_f+\alpha_f, \beta_r+\alpha_r)$$

Bayesian learning for NB parameters – a.k.a. smoothing

- **Posterior distribution of $\theta$**
  - Bernoulli:

    $$P(\theta | D) = \frac{\theta^{\beta_f+\alpha_f-1}(1-\theta)^{\beta_r+\alpha_r-1}}{B(\beta_f+\alpha_f, \beta_r+\alpha_r)} \sim \text{Beta}(\beta_f+\alpha_f, \beta_r+\alpha_r)$$

  - Multinomial

    $$P(\theta | D) = \frac{[\prod_{i=1}^{K} \theta_i^{\beta_i+\alpha_i-1}]}{D(\beta_1+\alpha_1, \ldots, \beta_K+\alpha_K)} \sim \text{Dirichlet}(\beta_1+\alpha_1, \ldots, \beta_K+\alpha_K)$$

- **MAP estimate**:

  $$\hat{\theta} = \arg \max_{\theta} P(\theta | D) =$$

  - Beta prior equivalent to extra thumbtack flips
  - As $N \rightarrow \infty$, prior is “forgotten”
  - But, for small sample size, prior is important!
MAP for the parameters of NB

- Dataset of $N$ examples
  - Let $\beta_{i,a,b} = \text{Count}(X_i=a, Y=b) \leftarrow$ number of examples where $X_i=a$ and $Y=b$
  - Let $\gamma_b = \text{Count}(Y=b)$

- Prior
  
  \[ Q(X_i|Y) \propto \text{Multinomial}(\alpha_{i,1}, \ldots, \alpha_{i,K}) \]  
  or Multinomial($\alpha/K$)  
  \[ Q(Y) \propto \text{Multinomial}(\tau_{i,1}, \ldots, \tau_{i,M}) \]  
  or Multinomial($\tau/M$)

- MAP estimate

\[ \hat{\pi}_k = \arg \max_{\pi_k} \prod_{k} P(Y = y_k; \pi_k) P(\pi_k|x) \]

\[ \hat{\theta}_{ij,k} = \arg \max_{\theta_{ij,k}} \prod_{j} P(X_i = x_{ij} | Y = y_k; \theta_{ij,k}) P(\theta_{ij,k}|x_{ij}) \]

- Now, even if you never observe a feature/class, posterior probability never zero

Text classification

- Classify e-mails
  - $Y = \{\text{Spam, NotSpam}\}$

- Classify news articles
  - $Y = \{\text{what is the topic of the article?}\}$

- Classify webpages
  - $Y = \{\text{Student, professor, project, …}\}$

- What about the features $X$?
  - The text!
Features $X$ are entire document – $X_i$ for $i^{th}$ word in article

Article from rec.sport.hockey

Path: cantaloupe.srv.cs.cmu.edu!das-news.harvard.<br>
From: xxx@yyy.zzz.edu (John Doe)<br>
Subject: Re: This year's biggest and worst (opini:<br>
Date: 5 Apr 93 09:53:39 GMT<br>

I can only comment on the Kings, but the most obvious candidate for pleasant surprise is Alex Zhitnik. He came highly touted as a defensive defenseman, but he's clearly much more than that. Great skater and hard shot (though wish he were more accurate). In fact, he pretty much allowed the Kings to trade away that huge defensive liability Paul Coffey. Kelly Hrudey is only the biggest disappointment if you thought he was any good to begin with. But, at best, he's only a mediocre goaltender. A better choice would be Tomas Sandstrom, though not through any fault of his own, but because some thugs in Toronto decided...

NB for Text classification

- $P(X|Y)$ is huge!!!
  - Article at least 1000 words, $X=\{X_1,...,X_{1000}\}$
  - $X_i$ represents $i^{th}$ word in document, i.e., the domain of $X_i$ is entire vocabulary, e.g., Webster Dictionary (or more), 10,000 words, etc.

- NB assumption helps a lot!!!
  - $P(X_i=x_i|Y=y)$ is just the probability of observing word $x_i$ in a document on topic $y$

$$h_{NB}(x) = \arg \max_y P(y) \prod_{i=1}^{\text{LengthDoc}} P(x_i|y)$$
Bag of words model

- Typical additional assumption – **Position in document doesn’t matter**: \( P(X_i=x_i|Y=y) = P(X_k=x_i|Y=y) \)
  - “Bag of words” model – order of words on the page ignored
  - Sounds really silly, but often works very well!

\[
P(y) \prod_{i=1}^{\text{LengthDoc}} P(x_i|y)
\]

When the lecture is over, remember to wake up the person sitting next to you in the lecture room.

---

Bag of words model

- Typical additional assumption – **Position in document doesn’t matter**: \( P(X_i=x_i|Y=y) = P(X_k=x_i|Y=y) \)
  - “Bag of words” model – order of words on the page ignored
  - Sounds really silly, but often works very well!

\[
P(y) \prod_{i=1}^{\text{LengthDoc}} P(x_i|y)
\]

In is lecture lecture next over person remember room sitting the the the to to up wake when you
Bag of words model

- Typical additional assumption – Position in document doesn’t matter: \( P(X_i=x_i|Y=y) = P(X_k=x_k|Y=y) \)
  - “Bag of words” model – order of words on the page ignored
  - Sounds really silly, but often works very well!

\[
P(y) \prod_{i=1}^{\text{LengthDoc}} P(x_i|y)
\]

in is lecture lecture next over person remember room sitting the the the to to up wake when you

Bag of Words Approach

<table>
<thead>
<tr>
<th>Word</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>aardvark</td>
<td>0</td>
</tr>
<tr>
<td>about</td>
<td>2</td>
</tr>
<tr>
<td>all</td>
<td>2</td>
</tr>
<tr>
<td>Africa</td>
<td>1</td>
</tr>
<tr>
<td>apple</td>
<td>0</td>
</tr>
<tr>
<td>anxious</td>
<td>0</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>gas</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>oil</td>
<td>1</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>Zaire</td>
<td>0</td>
</tr>
</tbody>
</table>
NB with Bag of Words for text classification

- **Learning phase:**
  - Prior $P(Y)$
    - Count how many documents you have from each topic (+ prior)
  - $P(X|Y)$
    - For each topic, count how many times you saw word in documents of this topic (+ prior)

- **Test phase:**
  - For each document
    - Use naive Bayes decision rule

$$h_{NB}(x) = \arg \max_y P(y) \prod_{i=1}^{\text{LengthDoc}} P(x_i|y)$$

---

**Twenty News Groups results**

Given 1000 training documents from each group
Learn to classify new documents according to which newsgroup it came from

- comp.graphics
- comp.os.ms-windows.misc
- comp.sys.ibm.pc.hardware
- comp.sys.mac.hardware
- comp.windows.x
- alt.atheism
- soc.religion.christian
- talk.religion.misc
- talk.politics.mideast
- talk.politics.misc
- talk.politics.guns
- rec.autos
- rec.motorcycles
- rec.sport.baseball
- rec.sport.hockey
- sci.space
- sci.crypt
- sci.electronics
- sci.med

Naive Bayes: 89% classification accuracy
Learning curve for Twenty News Groups

What if we have continuous $X_i$?

- Eg., character recognition: $X_i$ is $i^{th}$ pixel

- Gaussian Naïve Bayes (GNB):

$$P(X_i = x \mid Y = y_k) = \frac{1}{\sigma_{ik}\sqrt{2\pi}} \cdot e^{-\frac{(x-\mu_{ik})^2}{2\sigma_{ik}^2}}$$

Sometimes assume variance
- is independent of $Y$ (i.e., $\sigma_i$),
- or independent of $X_i$ (i.e., $\sigma_k$)
- or both (i.e., $\sigma$)
Estimating Parameters:
Y discrete, Xi continuous

- Maximum likelihood estimates:

\[
\mu_{ik} = \frac{1}{\sum_j \delta(Y^j = y_k)} \sum_j x_i^j \delta(Y^j = y_k)
\]

\[
\bar{\sigma}_{ik}^2 = \frac{1}{\sum_j \delta(Y^j = y_k)} \left( \sum_{i,j} (x_i^j - \mu_{ik})^2 \delta(Y^j = y_k) \right)
\]
Example: GNB for classifying mental states  [Mitchell et al.]

- 1 mm resolution
- ~2 images per sec.
- 15,000 voxels/image
- non-invasive, safe
- measures Blood Oxygen Level Dependent (BOLD) response

Brain scans can track activation with precision and sensitivity  [Mitchell et al.]
Gaussian Naïve Bayes: Learned $\mu_{\text{voxel,word}}$
$P(\text{BrainActivity} \mid \text{WordCategory} = \{\text{People,Animal}\})$ [Mitchell et al.]

Pairwise classification accuracy: 85%

Learned Bayes Models – Means for $P(\text{BrainActivity} \mid \text{WordCategory})$ [Mitchell et al.]
<table>
<thead>
<tr>
<th>What you need to know about Naïve Bayes</th>
</tr>
</thead>
<tbody>
<tr>
<td>● Optimal decision using Bayes Classifier</td>
</tr>
<tr>
<td>● Naïve Bayes classifier</td>
</tr>
<tr>
<td>● What’s the assumption</td>
</tr>
<tr>
<td>● Why we use it</td>
</tr>
<tr>
<td>● How do we learn it</td>
</tr>
<tr>
<td>● Why is Bayesian estimation important</td>
</tr>
<tr>
<td>● Text classification</td>
</tr>
<tr>
<td>● Bag of words model</td>
</tr>
<tr>
<td>● Gaussian NB</td>
</tr>
<tr>
<td>● Features are still conditionally independent</td>
</tr>
<tr>
<td>● Each feature has a Gaussian distribution given class</td>
</tr>
</tbody>
</table>