CS 340 Lec. 12: Naive Bayes Classifiers

AD

February 2011

Classification

- We have training data $\left\{\mathbf{x}^k, y^k\right\}_{k=1}^N$.
- $\mathbf{x}$ corresponds to a vector of features.
- $Y \in \{1, 2, ..., C\}$ is a class label.
- **Aim**: Given $\left\{\mathbf{x}^k, y^k\right\}_{k=1}^N$, we want to learn a probabilistic model $p_{\mathbf{X},Y}(\mathbf{x}, y)$ to compute given a new input $\mathbf{x}$

$$p\left(Y = c \,|\, \mathbf{X} = \mathbf{x}\right) = p_{Y|\mathbf{X}}\left(c \,|\, \mathbf{x}\right).$$

- We will often use a non-rigorous notation: $p\left(y = c \,|\, \mathbf{x}\right)$.

Document Classification

- Assume you want to classify emails into 3 classes: $Y \in \{\text{spam,urgent,normal}\}$.
- We use a dictionary with d prespecified words and $\mathbf{X} = (X_1, ..., X_d)$ are binary features where

$$X_i = \mathbb{I}\,(\text{word } i \text{ is present in message})\,;$$

this is called a bag-of-words model.

- *Example*: Consider the following dictionary

	1	2	3	4	5	6	7
Words	John	Mary	sex	money	send	meeting	“unknown”

For the following sentence “John sent money to Mary after the meeting about money”, we obtain

$$\mathbf{x} = (1, 1, 0, 1, 0, 1, 1)\,.$$

Bayes Rule for Classifiers

- We have

$$p(y=c|\mathbf{x}) = \frac{p(\mathbf{x}|y=c)\,p(y=c)}{p(\mathbf{x})}$$

  where

$$p(\mathbf{x}) = \sum_{j=1}^{C} p(\mathbf{x}|y=j)\,p(y=j)$$

- $p(y=c|\mathbf{x})$ is the class posterior.
- $p(y=c)$ is the prior.
- $p(\mathbf{x}|y=c)$ is the class conditional distribution of the features.
- $p(\mathbf{x})$ is the (unconditional) distribution of the features.

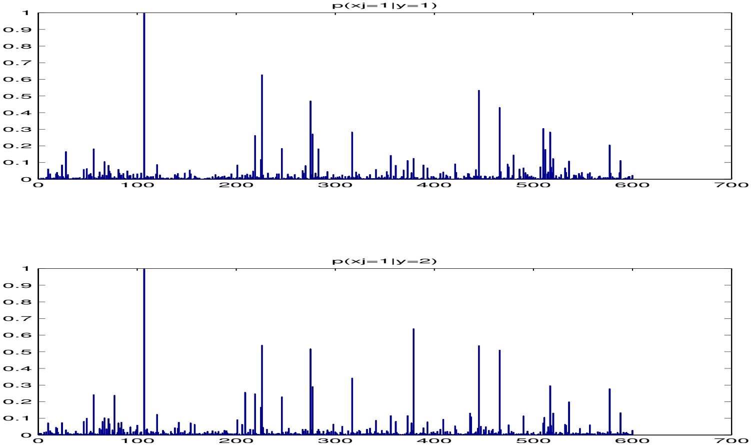
Naive Bayes Assumption

- What is the probability of generating a $d-$dimensional feature vector for each possible class $\{1, 2, ..., C\}$? It requires specifying $p\left(\mathbf{x}\middle|\, y = c\right)$.
- Naive Bayes assumes that

$$p\left(\mathbf{x}\middle|\, y = c\right) = \prod_{i=1}^{d} p\left(x_i\middle|\, y = c\right).$$

- E.g. proba of seeing "send" is assumed to be independent of seeing "money" given that we know this is a spam email.
- We can simply model $p\left(x_i\middle|\, y = c\right)$ using the Bernoulli distribution of parameter $\theta_{i,c} \in [0, 1]$; i.e.

$$\begin{aligned} p\left(x_i\middle|\, y = c\right) &= \theta_{i,c}^{\mathbb{I}(x_i=1)} \left(1 - \theta_{i,c}\right)^{\mathbb{I}(x_i=0)} \\ &= \theta_{i,c}^{x_i} \left(1 - \theta_{i,c}\right)^{1-x_i} \end{aligned}$$



Estimated class conditional densities $p\left(x_i = 1 \middle| y = c\right) = \widehat{\theta}_{i,c}$ for two document classes, corresponding to "X Windows" and "MS Windows". The spike corresponds to the word "subject" and we use $\widehat{\theta}_{i,c} = \sum_{k=1}^{N} \mathbb{I}\left(x_i^k = 1, y^k = c\right) / \sum_{k=1}^{N} \mathbb{I}\left(y^k = c\right)$.

Count Features for Document Classification

- Suppose now that we take

$$X_i = \text{Number of occurrences of word } i \text{ in message.}$$

- We have now $X_i \in \{0, 1, 2, ...\}$ so the Bernoulli distribution cannot be used to model $p\left(x_i \middle| y = c\right)$.
- We can use the Poisson distribution

$$p\left(x_i \middle| y = c\right) = \exp\left(-\theta_{i,c}\right) \frac{\theta_{i,c}^{x_i}}{x_i!}$$

where $\theta_{i,c}^{k} > 0$.
- We have $\mathbb{E}\left(X_i\right) = \mathbb{V}\left(X_i\right) = \theta_{i,c}$.
- We could estimate $\theta_{i,c}$ through $\widehat{\theta}_{i,c} = \sum_{k=1}^{N} x_i^k \mathbb{I}\left(y^k = c\right) / \sum_{k=1}^{N} \mathbb{I}\left(y^k = c\right)$.

Count Features for Document Classification

- An alternative model is

$$
\begin{aligned}
p\left(x_1, ..., x_d \right| y = c) &= \begin{pmatrix} & P & \\ x_1 \; x_2 & \cdots & x_d \end{pmatrix} \prod_{i=1}^{d} \theta_{i,c}^{x_i} \\
&= P! \prod_{i=1}^{d} \frac{\theta_{i,c}^{x_i}}{x_i!}
\end{aligned}
$$

  where $P = \sum_{i=1}^{d} x_i =$number of words in document, $\theta_{i,c} \geq 0$, $\sum_{i=1}^{d} \theta_{i,c} = 1$.

- This is a **multinomial distribution** of parameters $(\theta_{1,c}, \ldots, \theta_{d,c}, P)$.
- **Interpretation**: In class c, we have a population with $\theta_{i,c}\%$ of words i and $p\left(x_1, ..., x_d \right| y = c)$ is the probability of observing x_1 words 1, x_2 words 2,....,x_d words d.

- In this model we have $p\left(x_1, ..., x_d \middle| y = c\right) \neq \prod_{i=1}^{d} p\left(x_i \middle| y = c\right)$.
- We could estimate $\theta_{i,c}$ through

$$\widehat{\theta}_{i,c} = \frac{\sum_{k=1}^{N} \frac{x_i^k}{\left(\sum_{j=1}^{d} x_j^k\right)} \mathbb{I}\left(y^k = c\right)}{\sum_{k=1}^{N} \mathbb{I}\left(y^k = c\right)}$$

or through

$$\widehat{\theta}_{i,c} = \frac{\sum_{k=1:y^k=c}^{N} x_i^k}{\sum_{k=1:y^k=c}^{N} \left(\sum_{j=1}^{d} x_j^k\right)}.$$

- What is the "best" estimate intuitively?

Which Class-Conditional Density?

- For document classification, the multinomial model is found to work best. For sake of simplicity, we will mostly focus on the multivariate Bernoulli (binary features) model.
- We can easily handle features of different types; e.g. $x_1 \in \{0, 1\}$, $x_2 \in \mathbb{R}$, $x_3 \in \mathbb{R}^+$, $x_4 \in \{0, 1, 2, 3, \ldots\}$.
- We can use Gaussians, Gamma, Bernoulli etc.

Class Prior

- To encode $Y \in \{1, 2, ..., C\}$, we simply use

$$p(y) = \prod_{i=1}^{C} \pi_i^{\mathbb{I}(y=i)}.$$

- We can alternatively use C binary variables $(Y_1, Y_2, ..., Y_C) \in \{0, 1\}^C$ such that $\sum_{i=1}^{C} Y_i = 1$; i.e. $Y = 2 \Leftrightarrow (Y_1, Y_2, Y_3) = (0, 1, 0)$ for $C = 3$ so

$$p(y_1, ..., y_C) = \prod_{i=1}^{C} \pi_i^{y_i}$$

where $\pi_i \geq 0$, $\sum_{i=1}^{C} \pi_i = 1$. This is a multinomial distribution of parameters $(\pi_1, \ldots, \pi_C, 1)$ also known as a multinoulli distribution of parameters $(\pi_1, \ldots, \pi_C)$.

Class Posterior

- Bayes rule yields for the multivariate Bernoulli model

$$
\begin{aligned}
p(y=c|\mathbf{x}) &= \frac{p(y=c)\,p(\mathbf{x}|\,y=c)}{p(\mathbf{x})} \\
&= \frac{\pi_c \prod_{i=1}^{d} \theta_{i,c}^{\mathbb{I}(x_i=1)} \; (1-\theta_{i,c})^{\mathbb{I}(x_i=0)}}{p(\mathbf{x})}
\end{aligned}
$$

- In practice, numerator and denominator are very small, so need to use logs to avoid underflow; i.e.

$$
\begin{aligned}
\log p(y=c|\mathbf{x}) = \log \pi_c + \textstyle\sum_{i=1}^{d} \mathbb{I}(x_i=1) \log \theta_{i,c} \\
+\mathbb{I}(x_i=0) \log (1-\theta_{i,c}) - \log p(\mathbf{x})
\end{aligned}
$$

- How to compute the normalizing constant

$$
\log p(\mathbf{x}) = \log \left(\sum_{c=1}^{C} p(\mathbf{x}, y=c) \right) = \log \left(\sum_{c=1}^{C} \pi_c f_c \right)
$$

Log-sum-exp Trick

- Define

$$\begin{aligned}
\log p(\mathbf{x}) &= \log\left(\sum_{c=1}^{C}\pi_c f_c\right), \\
b_c &= \log \pi_c f_c = \log \pi_c + \log f_c \\
\log p(\mathbf{x}) &= \log\left(\sum_{c=1}^{C} e^{b_c}\right) = \log\left(\left(\sum_{c=1}^{C} e^{b_c}\right) e^{-B} e^{B}\right) \\
&= \log\left(\sum_{c=1}^{C} e^{b_c - B}\right) + B, \\
B &= \max_c \; b_c;
\end{aligned}$$

e.g.

$$\log\left(e^{-120} + e^{-121}\right) = \log\left(e^{-120}\left(e^{0} + e^{-1}\right)\right) = \log\left(1 + e^{-1}\right) - 120.$$

Missing Features

- Suppose the value of x_1 is unknown.
- We can still use the classifier, just drop the term $p(x_1|c)$. Indeed we have

$$
\begin{aligned}
p(y=c|x_{2:d}) &\propto \int p(y=c, x_{1:d})\, dx_1 \\
&= p(y=c) \int p(x_{1:d}|y=c)\, dx_1 \\
&= p(y=c) \int \prod_{i=1}^{d} p(x_i|y=c)\, dx_1 \\
&= p(y=c) \prod_{i=2}^{d} p(x_i|y=c)
\end{aligned}
$$

- This is a big advantage of generative classifiers which specify $p(\mathbf{x}|y=c)$ over discriminative classifiers which learn $p(y=c|\mathbf{x})$ directly.

Parameter Learning

- So far we have assumed that the parameter of $p(\mathbf{x}|\, y = c)$ and $p(y = c)$ are known.
- Obviously in practice, we are going to have to learn them from the training data $\left\{\mathbf{x}^k, y^k\right\}_{k=1}^{N}$.
- We have come up with intuitive estimates: e.g. for the multivariate Bernoulli model $p(\mathbf{x}|\, y = c)$ and $p(y = c)$ we took

$$\begin{aligned}
\widehat{\theta}_{i,c} &= \frac{\sum_{k=1}^{N} \mathbb{I}\left(x_i^k = 1, y^k = c\right)}{\sum_{k=1}^{N} \mathbb{I}\left(y^k = c\right)}, \\
\widehat{\pi}_c &= \frac{\sum_{k=1}^{N} \mathbb{I}\left(y^k = c\right)}{N}.
\end{aligned}$$

- Is there any rational for this? Can we do any better?