CS 6140: Machine Learning **Spring 2015**
College of Computer and Information Science
Northeastern University
Lecture 2 **January, 26**
Instructor: Bilal Ahmed **Scribe: Bilal Ahmed & Yu Wen**

The Naive Bayes Model

1 Introduction

The Naive Bayes model is another kind of classifier, which is very effective in practical usage. This model is often used as a classifier for distinguishing the category of text data such as predicting whether an email is spam or not. The nave Bayes classifier is a generative model, which is different from logistic regression model – a discriminant model.

2 Background Knowledge

Here we introduce some basic concept we are going to use later.

Bayes' theorem :

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Bayes' theorem can be viewed as the transformation of formula $P(A|B)P(B) = P(AB) = P(B|A)P(A)$. By using the notation we've used in previous lectures, we can denote our classification problem as followings,

$$P(y|\mathbf{x}) = \frac{P(\mathbf{x}|y)P(y)}{P(\mathbf{x})}$$

Usually, $P(y|\mathbf{x})$ is called the **posterior**. $P(\mathbf{x}|y)$ is called the **likelihood**. $P(y)$ is called the **prior**. It will make sense for these terminologies if we think like this: for any given data or example $\mathbf{x}$, $P(y)$ is the probability of y being equal to some value, which is not related to the $\mathbf{x}$ we observed, thus, we can treat it like a prior knowledge and call it the prior. $P(\mathbf{x}|y)$ is the probability of observing a certain data $\mathbf{x}$ given a label is observed, thus, we can see it as how likely we can observe a certain data if we know the label. $P(y|\mathbf{x}))$ can be viewed as the probability of a certain data is classified as y after we've known the prior and the likelihood, which makes the name "posterior" reasonable.

chain rule :
The chain rule is a formula to calculate the joint probability of n events $A_n, A_{n-1}, ..., A_1$ as followings,

$$P(A_n, A_{n-1}, ..., A_1) = P(A_n|A_{n-1}A_{n-2})...A_1)P(A_{n-1}|A_{n-2}A_{n-3})...A_1)...P(A_2|A_1)$$

3 Build the Model

As we've discussed in the previous lectures, we have the design matrix $\mathbf{X}$ and corresponding labels $\mathbf{y}$. Here we suppose, for the i^{th} example, $y_i \in \{C_1, C_2, ...C_j\}$. Our task is to find the optimal value of y_i^* to classify y_i such that the posterior is maximum. Then the Naive Bayes classifier can be described as followings,

$$y_i^* = \arg\max_{C_k} P(y_i = C_k|\mathbf{x}_i) \quad (\text{for } k = 1, 2, ..., j)$$

According to the Bayes theorem, we have

$$y_i^* = \arg\max_{C_k} \frac{P(\mathbf{x}_i|y_i = C_k)P(y_i = C_k)}{P(\mathbf{x}_i)}$$

According to the definition of marginal probability distribution $P(\mathbf{x}_i) = \sum_{k=1}^{j} P(\mathbf{x}_i)P(y_i = C_k)$, then we have

$$y_i^* = \arg\max_{C_k} \frac{P(\mathbf{x}_i|y_i = C_k)P(y_i = C_k)}{\sum_{k=1}^{j} P(\mathbf{x}_i|y_i = C_k)}$$

$$= \arg\max_{C_k} P(\mathbf{x}_i|y_i = C_k)P(y_i = C_k)$$

Here, we assume y_i follows a multinomial distribution.

4 Train the Model

The training of Naive Bayes is to find the optimal parameter to ensure we get the maximum posterior probability.

$$\begin{aligned} y_i^* &= \arg\max_{C_k} P(\mathbf{x}_i|y_i = C_k)P(y_i = C_k) \\ &= \arg\max_{C_k} P(\mathbf{x}_{im}, \mathbf{x}_{im-1}, ..., \mathbf{x}_{i1}|y_i = C_k)P(y_i = C_k) \\ &= \arg\max_{C_k} P(\mathbf{x}_{im}|\mathbf{x}_{im-1}, ..., \mathbf{x}_{i1}, y_i = C_k)P(\mathbf{x}_{im-1}|\mathbf{x}_{im-2}, ..., \mathbf{x}_{i1}, y_i = C_k)P(\mathbf{x}_{i1}|y_i = C_k)P(y_i = C_k) \end{aligned}$$

The last line was derived according to the chain rule. If we want to compute above probability, the computational cost is very expensive.Thus, we assume that $\mathbf{x}_{im}, \mathbf{x}_{im-1}, ..., \mathbf{x}_{i1}$ are **conditionally independent**, which means given the condition that $y_i = C_k$, $\mathbf{x}_{im}, \mathbf{x}_{im-1}, ..., \mathbf{x}_{i1}$ are independent. This assumption could make the computation much simple, such that

$$\begin{aligned} y_i^* &= \arg\max_{C_k} P(\mathbf{x}_{im}, \mathbf{x}_{im-1}, ..., \mathbf{x}_{i1}|y_i = C_k)P(y_i = C_k) \\ &= \arg\max_{C_k} P(y_i = C_k) \prod_{l=1}^{m} P(\mathbf{x}_{il}|y_i = C_k) \end{aligned}$$

Then the parameter we are going to estimate from the given data becomes the likelihood $P(\mathbf{x}_{il}|y_i = C_k)$ for $l = 1...m$ and the prior $P(y_i = C_k)$
To achieve the maximum posterior

$$P(y_i = C_k) = \frac{\text{count of } y_i = C_k}{N}$$

$$P(\mathbf{x}_{il}|y_i = C_k) = \frac{\text{count of both } x_{il} \text{ and } y_i = C_k}{\text{count of } y_i = C_k}$$

Here we gave the derivation of above conclusion. Similarly we first get the data likelihood function

$$\begin{aligned} L &= \prod_{i=1}^{N} P(\mathbf{x}_i, y_i) \\ &= \prod_{i=1}^{N} (P(y_i) \prod_{l=1}^{m} P(\mathbf{x}_{il}|y_i)) \end{aligned}$$

Then, we use the logarithm operation

$$\begin{aligned} l &= ln(L) \\ &= \sum_{i=1}^{N} ln(P(y_i) \prod_{l=1}^{m} P(\mathbf{x}_{il}|y_i) \\ &= \sum_{i=1}^{N} \sum_{l=1}^{m} ln(P(\mathbf{x}_{il}|y_i)) + \sum_{i=1}^{N} ln(P(y_i)) \end{aligned}$$

Here we assume that y_i follows a multinominal distribution, and notice that y_i is not related to the left part of above equation, we can rewrite the right part of the above equation as

$$\begin{aligned} \sum_{i=1}^{N} ln(P(y_i)) &= \sum_{i=1}^{N} ln(\prod_{k=1}^{j} \mathbb{1}(y_i = C_k) P(y_i = C_k)) \\ &= \sum_{i=1}^{N} \sum_{k=1}^{j} \mathbb{1}(y_i = C_k) ln(P(y_i = C_k)) \end{aligned}$$

Thus, the estimate of y_i can be formed as

$$\begin{aligned} max \sum_{i=1}^{N} \sum_{k=1}^{j} \mathbb{1}(y_i = C_k) ln(P(y_i = C_k)) \\ s.t. \sum_{k=1}^{j} P(y_i = C_k) = 1 \end{aligned}$$

By using the Lagrange multipliers to solve the problem , we can get the Lagrange function J. Thus we can rewrite the optimizing problem as

$$max \quad J = \sum_{i=1}^{N} \sum_{k=1}^{j} \mathbb{1}(y_i = C_k) ln(P(y_i = C_k)) + \lambda(\sum_{k=1}^{j} P(y_i = C_k) - 1)$$

,where λ is the Lagrange multiplier.
Clearly, J is a function regards to $P(y_i = C_1), P(y_i = C_2), ..., P(y_i = C_j)$ and λ. By taking the partial derivative of J with respect to λ ,we have

$$\lambda = -N$$

By taking the partial derivative of J with respect to $P(y_i = C_k)$ (for $k = 1, 2...j$) and set it to 0, we have

$$\sum_{i=1}^{N} \frac{\mathbb{1}(y_i = C_k)}{P(y_i = C_k)} + \lambda = 0$$

Thus, we have

$$P(y_i = C_k) = \frac{\text{count of } y_i = C_k}{N}$$

5 Advantages and Disadvantages

This conditionally independence is somewhat naive since it may not hold true in reality. For example, if we treate each word in an email as a feature, each word is not independent given the reality that this document is classified as spam. This is the reason why Naive Bayes is called "naive".

However, in practical application, this model is very effective.In addition, the training process of the Naive Bayes didn't involve the process of using an iterative method, which makes the training very fast.

6 Reference

1. Jiawei Han, Data Mining Concepts and Techniques, 3rd