Knowledge Discovery in Data: Naïve Bayes

Overview

Naïve Bayes methodology refers to a *probabilistic approach* to information discovery in data. The idea behind the **Naïve Bayes** technique is applicable to the following problems studied in this course:

1. Classification/Supervised Learning.
2. Text Classification.

Naïve Bayesian Classification

Recall the nature of the *classification problem*:

Classification Problem. Given a (training) dataset $D = (A_1, \ldots, A_n, C)$, construct a **classification/prediction function** that correctly predicts the class label C for every record in D.

One way of predicting the *class label* $c(d)$ for a record $d \in D$ is to **estimate the probabilities** $Pr(c(d) = c_1), Pr(c(d) = c_2), \ldots, Pr(c(d) = c_k)$, and **pick as the prediction the class with the highest probability**.

Note: In general, we *do not need to know* the exact probabilities, it **suffices to know their order (ranks)**.

Naïve Bayes is a method of estimating the probabilities (ranks) of a record belonging to each class.

Classifiers and Observations. Imagine that a classifier you are building is a black box with an input, which can observe a record from a dataset D, and an output, on which *you can observe* the class label $c \in dom(C)$[1]

Notation. **Naïve Bayes** method encodes $Pr(c(d) = c_i)$ as the conditional probability of observing c_i in the output when d is observed in the input:

$$Pr(d = c_i) = Pr(c_i|d).$$

$d = (a_1, \ldots, a_n)$. Therefore,

$$Pr(c_i|d) = Pr(c_i|A_1 = a_1, A_2 = a_2, \ldots, A_n = a_n).$$

By definition of **conditional probability**:

$$Pr(X|Y) = \frac{Pr(X \wedge Y)}{Pr(Y)}.$$

Bayes Theorem for conditional probabilities:

$$Pr(X|Y) \cdot Pr(Y) = Pr(Y|X) \cdot Pr(X).$$

When $Pr(Y) \neq 0$ and $Pr(X) \neq 0$, Bayes Theorem can be rewritten as:

$$Pr(X|Y) = \frac{Pr(Y|X) \cdot Pr(X)}{Pr(Y)}.$$

Naïve Bayes Step 1: The Bayes Part (Applying Bayes Theorem): According to Bayes Theorem, the conditional probability $Pr(c_i|d)$ of observing an object of category c_i given values in vector d can be represented as:

$$Pr(c_i|d) = \frac{Pr(d|c_i) \cdot Pr(c_i)}{Pr(d)},$$

or

$$Pr(c_i|d) = Pr(c_i|A_1 = a_1, \ldots A_n = a_n) = \frac{Pr(A_1 = a_1, \ldots, A_n = a_n|c_i) \cdot Pr(c_i)}{Pr(A_1 = a_1, \ldots A_n = a_n)}.$$

Here,

- $P(d|c_i) = Pr(A_1 = a_1, \ldots, A_n = a_n|c_i)$ is the probability of observing vector d given that the classifier recognized the vector as belonging to class c_i. (i.e., the probability of observing d among all vectors of class c_i).
- $P(c_i)$ is the probability of observing a vector that belongs to class c_i (textsfprior probability of class c_i).
- $P(d)$ is the probability of observing vector d in the input. (prior probability of vector d).

[1] See the Classification/Supervised Learning lecture notes for the notation.

What's the point? We have just reduced the problem of estimating $Pr(c_i|d)$ to the problem of estimating $P(d|c_i)$ (as well as $Pr(c_i)$ and $Pr(d)$).

Naïve Bayes Step 2: Simplification. We need to estimate three probabilities: $Pr(d|c_i), Pr(c_i)$ and $Pr(d)$. We observe the following:

1. **Estimating $Pr(c_i)$.** The probability of observing a vector from class c_i can be estimated as the fraction of the training set D that belongs to this class.

   In other words, let $D_i = \{d \in D | c(d) = c_i\}$. Then,

   $$Pr(c_i) = \frac{|D_i|}{|D|}.$$

2. **Dealing with $Pr(d)$.** Vector d, when it occurs in the training set can be associated with one of k classes. This means that we can represent $Pr(d)$ as the sum of conditional probabilities of observing d given that a specific class $c_1, \ldots, c_k$ has been observed:

   $$Pr(d) = Pr(d|c_1) \cdot Pr(c_1) + \ldots Pr(d|c_k) \cdot Pr(c_k) = \sum_{i=1}^{k} Pr(d|c_i) \cdot Pr(c_i).$$

   We observe that $Pr(d)$ remains constant among $Pr(c_1|d)$, ..., $Pr(c_k|d)$:

   $$Pr(c_1|d) = \frac{Pr(d|c_1) \cdot Pr(c_1)}{Pr(d)}; Pr(c_2|d) = \frac{Pr(d|c_2) \cdot Pr(c_2)}{Pr(d)}; \ldots; Pr(c_k|d) = \frac{Pr(d|c_k) \cdot Pr(c_k)}{Pr(d)}.$$

   Therefore, ***if we are interested solely in* ranking *the probability estimates*, we can ignore $Pr(d)$ and concentrate on estimating**

   $$Pr(d|c_i) \cdot Pr(c_i).$$

Naïve Bayes: Step 3: the Naïve Part (estimating conditional probability): We now need to estimate

$$Pr(d|c_i) = Pr(A_1 = a_1, \ldots, A_n = a_n|c_i).$$

The **naïve** part of the method comes from the **conditional independence assumption.**

Conditional Independence Assumption. Two random variables X and Y are said to be **conditionally independent** iff for all $x \in dom(X), y \in dom(Y)$,

$$Pr(X = x|Y = y) = Pr(X = x),$$

i.e., if the probability of observing any value of X is not affected by having had observed a specific value of Y.

The corrolary to the conditional independence property is that the joint probability of two random variables is equal to the product of their marginal probabilities:

$$Pr(X = x \wedge Y = y) = Pr(X = x) \cdot Pr(Y = y).$$

Estimating $Pr(d|c_i)$ using conditional independence assumption. We apply **conditional independence assumption** to

$$Pr(d|c_i) = Pr(A_1 = a_1, \ldots, A_n = a_n|c_i).$$

That is, we assume that each attribute variable A_i is conditionally independent of $A_1, \ldots, A_{i-1}, A_{i+1}, \ldots, A_n$ given C, the class variable:

$$Pr(A_i = a_i|A_1 = a_1, \ldots A_{i-1} = a_{i-1}, A_{i+1} = a_{i+1}, \ldots, A_n, C = c_j) = Pr(A_i = a_i|C = c_j).$$

Therefore:

$$Pr(A_1 = a_1, \ldots, A_n = a_n|c_i) = Pr(A_1 = a_1|c_i) \cdot \ldots \cdot Pr(A_n = a_n|c_i) = \prod_{j=1}^{n} Pr(A_j = a_j|c_i).$$

$Pr(A_j = a_j|c_i)$ is the probability of observing a record with $A_j = a_j$ in class c_i.

What's the point? We have now reduced the problem of estimating the conditional probability $Pr(c_i|d)$ to the problem of estimating the family of probabilities $Pr(A_j = a_j|c_i)$.

Naïve Bayes Step 4: Estimating Probabilities. We estimate $Pr(A_j = a_j|c_i)$ as follows.

Let $D_i = \{d \in D|c(d) = c_i\}$ be the set of all records of class c_i. Let $D_{ij} = \{d \in D_i|d.A_j = a_j\}$ be the set of all records of class c_i with a_j as the value of the attribute A_j. Then, our estimate for $Pr(A_j = a_j|c_i)$ is

$$Pr(A_j = a_j|c_i) = \frac{|D_{ij}|}{|D_i|},$$

i.e., the percentage of records in class c_i that have a_j as the value of A_j.

Naïve Bayes Step 5: Predict the class . Compute estimates

$$Pr(d|c_1) \cdot Pr(c_1), \ldots, Pr(d|c_k) \cdot Pr(c_k),$$

using Steps 1–4 from above.

Predict:

$$c(d) = arg \max_{i=1,\ldots,k} (Pr(d|c_i) \cdot Pr(c_i)).$$

Combining it all together

Naïve Bayes for classification:

1. Use Bayes Theorem.

$$Pr(c_i|d) = Pr(c_i|A_1 = a_1, \ldots A_n = a_n) = \frac{Pr(A_1 = a_1, \ldots, A_n = a_n|c_i) \cdot Pr(c_i)}{Pr(A_1 = a_1, \ldots A_n = a_n)}.$$

2. Simplify.

$$Pr(c_i|A_1 = a_1, \ldots A_n = a_n) \sim Pr(A_1 = a_1, \ldots, A_n = a_n|c_i) \cdot Pr(c_i).$$

3. Apply Independence Assumption.

$$Pr(c_i|A_1 = a_1, \ldots A_n = a_n) \sim Pr(A_1 = a_1|c_i) \cdot \ldots \cdot Pr(A_n = a_n|c_i) \cdot Pr(c_i) = Pr(c_i) \cdot \prod_{j=1}^{n} Pr(A_j = a_j|c_i).$$

4. Estimate Probabilities.

$$Pr(c_i) = \frac{|D_i|}{|D|}.$$

$$Pr(A_j = a_j|c_i) = \frac{|D_{ij}|}{|D_i|}.$$

$$Pr(c_i|A_1 = a_1, \ldots A_n = a_n) \sim \frac{|D_i|}{|D|} \cdot \prod_{j=1}^{n} \frac{|D_{ij}|}{|D_i|} = \frac{|D_{i1}| \cdot \ldots \cdot |D_{in}|}{|D| \cdot |D_i|^{n-1}}.$$

5. Predict.

$$c(d) = arg \max_{i=1,\ldots,k} (Pr(d|c_i) \cdot Pr(c_i)) = arg \max_{i=1,\ldots,k} \frac{|D_{i1}| \cdot \ldots \cdot |D_{in}|}{|D| \cdot |D_i|^{n-1}}.$$

Naïve Bayes for Text Classification

Generative Models for Text Classification

Text Classification. The problem of determining/predicting a caterogy label for text documents from a document collection in a presence of training set.

Bag of words. One of traditional representations of a text document, **a bag of words** is collection of all **words** or **terms** found in a document. Multiple occurences of the same term are rpesented by the term frequency associated with it.

Generative model. A name for a class of classification models. A **generative model** assumes that there is a **parametric probability distribution** associated with the **process of text generation** for *each category* of text documents.

Different categories have different parameters associated with their probability distributions. Training data is used to estimate the parameters.

Classification is performed by estimating the **posterior probability** of a text document being generated from each probability distribution, and selecting the category/class label which maximizes it.

Two assumptions:

1. The text documents are generated by a **mixture model**.
2. There is a one-to-one correspondence between **mixture components** and **class labels**.

Mixture models. A **mixture model** represents the probability distribution as a **mixture** of probability distributions for a number of **components**.

Let $Y = \{Y_1, \ldots, Y_k\}$ be the k components of a mixture model (each is a random variable). Assumption 2 from above states that each component Y_i corresponds to a class label c_i from the list of class labels $C = \{c_1, \ldots c_k\}$.

Let Θ_i be all the parameters of Y_i. Let

$$\Theta = (\Theta_1, \Theta_2, \ldots, \Theta_k).$$

let ϕ_i be the **weight** of component i in the mixture model. The weights obey the

$$\sum_{i=1}^{k} \phi_i = 1$$

equality.

ϕ_i is a **class prior probability**, i.e., the probability of a randomly selected documents to belong to class c_i:

$$\phi_i = Pr(c_i|\Theta).$$

Text generation via mixture model. Let $M = \langle Y = (Y_1, \ldots, Y_k); \Theta = (\Theta_1, \ldots, \Theta_k); \phi = (\phi_1, \ldots, \phi_k)\rangle$ be a generative mixture model for a collection of documents $D = \{d_1, \ldots d_N\}$ with k classes $C = \{c_1, \ldots, c_k\}$.

A document d_j is generated by M as follows:

1. The model selects a mixture component i using $\phi = (\phi_1, \ldots, \phi_k)$ as the probability distribution.
2. Document d_j is generated by the Y_i component of the model using the probability distribution $Pr(d_j|c_i, \Theta_i)$.

Posterior Probability. Given a document d_j, the probability that it was generated via a given mixture model is

$$Pr(d_j|\Theta) = \sum_{i=1}^{k} Pr(c_i|\Theta) \cdot Pr(d_j|c_i; \Theta_i) = \sum_{i=1}^{k} \phi_i \cdot Pr(d_j|c_i; \Theta_i).$$

Naïve Bayes Generative Text Classification Model

Assumptions.

1. Each document is represented as a **bag of words**.
2. Each document is generated by generating each word in it. Each word is **generated independently of other words**.
3. The probability of generating a word **does not depend** on its position in the document. (which is why we reduce the document to a bag of words).

Document generation. With this assumption, each term is drawn from a **multinomial distribution** over the set $V = \{t_1, \ldots, t_M\}$ of words (terms) found in the document collection.

A document of length n is generated by n independent term generations.

Parameters. A multinomial distribution X over the vocabulary V has **parameters** $p_1, \ldots, p_M$, where $p_i = Pr(X = p_i) = Pr(t_i|c;\Theta)$ (i.e., the probability to generate term t_i for a given class c).

Additionally, $n = |d|$, the length of a document is a parameter for the generative model.

Probability of document generation. Let $d_i \in D$ be a document. For $t_s \in V$, let $freq(t_s, d_i)$ be the number of occurrences of t_s in d_i. The probability of d_i being generated by the jth component of the Naïve Bayes generative model (i.e., the probability that $Class(d_i) = c_j$) can be formulated as follows:

$$Pr(d_i|c_j;\Theta) = Pr(|d_i|)\cdot|d_i|!\cdot\prod_{s=1}^{M}\frac{Pr(t_s|c_i;\Theta)^{freq(t_s,d_i)}}{freq(t_s,d_i)!};$$

$$\sum_{s=1}^{M} freq(t_s,d_i) = |d_i|;$$

$$\sum_{s=1}^{M} Pr(t_s|c_j;\Theta) = 1.$$

Parameter Estimation. Let T be a training set of documents, and $T_1, \ldots T_k$ be the subsets of T of documents with class labels $c_1, \ldots, c_k$ respectively. Let $|T| = N_T$.

As $\hat{\Theta}$ we denote the estimate for the parameters $\Theta = (\Theta_1, \ldots, \Theta_k)$. We estimate $Pr(t_s|c_i;\hat{\Theta})$ as follows:

$$Pr(t_s|c_i,\hat{\Theta}) = \frac{\sum_{j=1}^{N} freq(t_s,d_j)\cdot Pr(c_i|d_j)}{\sum_{t\in V}\sum_{j=1}^{N} freq(t,d_j)\cdot Pr(c_i|d_j)}.$$

Here $Pr(c_i|d_j) = 1$ if $d_j \in T_i$ and $Pr(c_i|d_j) = 0$ otherwise.

Lidstone smoothing. Some words may occur in documents from catagory c_i, but may be absent in training set documents. To account for this possibility, the parameter estimation formula from above is augmented by using a small parameter λ:

$$Pr(t_s|c_i, \hat{\Theta}) = \frac{\lambda + \sum_{j=1}^{N_T} freq(t_s, d_j) \cdot Pr(c_i|d_j)}{\lambda \cdot M + \sum_{t \in V} \sum_{j=1}^{N_T} freq(t, d_j) \cdot Pr(c_i|d_j)}.$$

Whem $\lambda = 1$, this is known as **Laplace smoothing**. A typical value for text classification/Information retrieval is $\lambda = 0.5$.

Estimating class prior probabilitites. Component weights ϕ_i, a.k.a., class prior probabilities can be estiamted from the training set as follows:

$$\phi_i = Pr(c_i|\hat{\Theta}) = \frac{\sum_{j=1}^{N} Pr(c_i|d_j)}{N_T} = \frac{|T_i|}{N_T}.$$

Classification. The probability of a document d_j belonging to class c_i, $Pr(c_i|d_j;\hat{\Theta})$ is estimated as follows:

$$Pr(c_i|d_j;\hat{\Theta}) = \frac{Pr(c_i|\hat{\Theta}) \cdot Pr(d_j|c_i;\hat{\Theta})}{Pr(d_j|\hat{\Theta})} =$$

$$= \frac{Pr(c_i|\hat{\Theta}) \cdot \prod_{t \in d_j} freq(t, d_j) \cdot Pr(t|c_i;\hat{\Theta})}{\sum_{r=1} k \left(Pr(c_r|\hat{\Theta}) \cdot \prod_{t \in d_j} freq(t, d_j) \cdot Pr(t|c_r;\hat{\Theta}) \right)}.$$

Then,

$$class(d_j) = arg \max_{c_i \in C} Pr(c_i|d_j;\hat{\Theta}).$$