Naïve Bayes Classifier

Boris Velichkov

Naïve Bayes Classifier

- Machine Learning
  - Supervised Learning
    - Classification
  - Global Learning
  - Model-Based Learning
  - Eager Learning

Probabilistic Model

- Abstractly, ***Naïve Bayes*** is a conditional probability model: given a problem instance to be classified, represented by a vector $\boldsymbol{x} = (x_1, \ldots, x_n)$ representing some ***n*** features (independent variables), it assigns to this instance probabilities

$$p(C_k \mid x_1, \ldots, x_n)$$

for each of **K** possible outcomes or *classes* $\boldsymbol{C_K}$.

Probabilistic Model

- The problem with the above formulation is that if the number of features ***n*** is large or if a feature can take on a large number of values, then basing such a model on probability tables is infeasible. We therefore reformulate the model to make it more tractable. Using Bayes' theorem, the conditional probability can be decomposed as

$$p(C_k \mid \mathbf{x}) = \frac{p(C_k)\, p(\mathbf{x} \mid C_k)}{p(\mathbf{x})}$$

Probabilistic Model

- In plain English, using Bayesian probability terminology, the above equation can be written as

$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

Probabilistic Model

- In practice, there is interest only in the numerator of that fraction, because the denominator does not depend on ***C*** and the values of the features $\boldsymbol{x_i}$ are given, so that the denominator is effectively constant. The numerator is equivalent to the ***joint probability*** model

$$p(C_k, x_1, \dots, x_n)$$

Probabilistic Model

- which can be rewritten as follows, using the ***chain rule*** for repeated applications of the definition of ***conditional probability***:

$$
\begin{aligned}
p(C_k, x_1, \dots, x_n) &= p(x_1, \dots, x_n, C_k) \\
&= p(x_1 \mid x_2, \dots, x_n, C_k)\, p(x_2, \dots, x_n, C_k) \\
&= p(x_1 \mid x_2, \dots, x_n, C_k)\, p(x_2 \mid x_3, \dots, x_n, C_k)\, p(x_3, \dots, x_n, C_k) \\
&= \cdots \\
&= p(x_1 \mid x_2, \dots, x_n, C_k)\, p(x_2 \mid x_3, \dots, x_n, C_k) \cdots p(x_{n-1} \mid x_n, C_k)\, p(x_n \mid C_k)\, p(C_k)
\end{aligned}
$$

Probabilistic Model

- Now the "naïve" conditional independence assumptions come into play: assume that all features in ***x*** are mutually independent, conditional on the category $\boldsymbol{C_K}$. Under this assumption,

$$p(x_i \mid x_{i+1}, \ldots, x_n, C_k) = p(x_i \mid C_k)$$

Probabilistic Model

- Thus, the joint model can be expressed as

$$
\begin{aligned}
p(C_k \mid x_1, \dots, x_n) &\propto p(C_k, x_1, \dots, x_n) \\
&\propto p(C_k)\, p(x_1 \mid C_k)\, p(x_2 \mid C_k)\, p(x_3 \mid C_k)\, \cdots \\
&\propto p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)\,,
\end{aligned}
$$

where $\propto$ denotes proportionality.

Probabilistic Model

- This means that under the above independence assumptions, the conditional distribution over the class variable ***C*** is:

$$p(C_k \mid x_1, \ldots, x_n) = \frac{1}{Z} p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)$$

where the evidence $Z = p(\mathbf{x}) = \sum_k p(C_k)\, p(\mathbf{x} \mid C_k)$

is a scaling factor dependent only on $x_1, \ldots, x_n$, that is, a constant if the values of the feature variables are known.

Constructing a Classifier from the Probability Model

The discussion so far has derived the independent feature model, that is, the naïve Bayes probability model. The naïve Bayes classifier combines this model with a decision rule. One common rule is to pick the hypothesis that is most probable; this is known as the maximum a posteriori or MAP decision rule. The corresponding classifier, a Bayes classifier, is the function that assigns a class label $\hat{y} = C_k$ for some k as follows:

$$\hat{y} = \underset{k \in \{1, \dots, K\}}{\operatorname{argmax}} \ p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)$$

Example

http://shatterline.com/blog/2013/09/12/not-so-naive-classification-with-the-naive-bayes-classifier/

Zero Probability Problem

?

Laplace Smoothing

In statistics, Laplace Smoothing is a technique to smooth categorical data. Laplace Smoothing is introduced to solve the problem of zero probability. By applying this method, prior probability and conditional probability can be written as:

$$p_\lambda(C_k) = p_\lambda(Y = C_k) = \frac{\sum_{i=1}^{N} I(y_i = C_k) + \lambda}{N + K\lambda}$$

$$p(x_1 = a_j | y = C_k) = \frac{\sum_{i=1}^{N} I(x_{1i} = a_j, y_i = C_k) + \lambda}{\sum_{i=1}^{N} I(y_i = C_k) + A\lambda}$$

K denotes the number of different values in y and A denotes the number of different values in aj. Usually lambda in the formula equals to 1.

Log Probabilities

$$\log(ab) = \log(a) + \log(b)$$

$$\log(P(\text{class }_i | \,\mathbf{data})) \propto \log(P(\text{class}_i)) + \sum_j \log(P(\text{data}_j | \text{class}_i))$$

Gaussian Naïve Bayes

$$p(x = v \mid C_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}}\, e^{-\frac{(v-\mu_k)^2}{2\sigma_k^2}}$$