Theoretical framework of Gaussian Naive Bayes

1 Introduction

Naive Bayes classifiers are among the simplest and most widely used probabilistic models for classification tasks. Their popularity stems from their simplicity and surprising effectiveness.

This small informal report aims to provide an accessible overview of the mathematical foundations of Naive Bayes classifiers. Specifically, we will begin by constructing discrete Naive Bayes classifiers and then, discuss how this approach might be adapted to work with continuous spaces. Finally, we will conclude with the presentation of the Gaussian Naive Bayes which is a specific continuous Naive Bayes classifier.

We hope that this report might be useful to better understand the principle of the Naive Bayes framework, thereby enabling more informed and effective applications of these models.

2 Discrete Naive Bayes

Let $\mathcal{X}$ represent a discrete sample space and $\mathcal{Y}$ a set of classes. Suppose we aim to build a classifier

$$h_\theta : \mathcal{X} \to \mathcal{Y}$$

that maps each observation $x \in \mathcal{X}$ to its associated class. The Bayesian approach consists of considering a classifier of the form

$$h_\theta(x) = \operatorname{argmax}_{y \in \mathcal{Y}} p(y|x)$$

Since the direct computation of $p(y|x)$ is generally infeasible, we will rewrite the previous expression using Bayes' theorem

$$\begin{aligned} h_\theta(x) &= \operatorname{argmax}_{y \in \mathcal{Y}} \; p(y|x) \\ &= \operatorname{argmax}_{y \in \mathcal{Y}} \; \frac{p(x|y)p(y)}{p(x)} \\ &= \operatorname{argmax}_{y \in \mathcal{Y}} \; p(x|y)p(y) \end{aligned}$$

While it may seem intuitive to construct a classifier by directly estimating $p(x|y)$ and $p(y)$ from a dataset, this approach often results in a non-functional model. Specifically, $p(x|y)$ will be zero for any (x, y) pair not represented in the dataset, rendering the model ineffective for such cases, and severely impacting its generalization capabilities

To overcome this limitation, Naive Bayes makes a *naive* hypothesis and assumes the independence of the feature of x given y. Under this assumption, we can write

$$p(x|y) = \prod_{i=1}^{k} p(x_i|y),$$

where $x = (x_1, \ldots, x_k)$ represents the features of x. Substituting this into the classifier expression yields

$$h_\theta(x) = \operatorname{argmax}_{y \in \mathcal{Y}} \; p(y) \prod_{i=1}^{k} p(x_i|y).$$

However, this formulation involves the multiplication of small probabilities, which can result in numerical instability. To mitigate this, a common approach consists of applying the logarithm to the expression. Since the logarithm is a monotonic function, it preserves the result of the argmax. Consequently, the final formulation of the model, representing the Naive Bayes classifier, becomes

$$h_\theta(x) = \text{argmax}_{y\in\mathcal{Y}} \ \log p(y) + \sum_{i=1}^{k} \log p(x_i|y).$$

3 Continuous Naive Bayes

The classifier constructed in the previous section performs surprisingly well in practice. However, it suffers from a significant limitation: the sample space $\mathcal{X}$ must be discrete. Otherwise the probabilities $p(x_i|y)$ will be zeros for every feature, making the model completely useless. However, in this section, we will demonstrate how the Naive Bayes approach can be extended to handle continuous feature spaces.

First, we encapsulate each x_i within an interval of the form $]x_i - \frac{\delta}{2}; x_i + \frac{\delta}{2}[$ with small $\delta > 0$ and we consider a classifier of the form

$$\text{argmax}_{y\in\mathcal{Y}} \ \log p(y) + \sum_{i=1}^{k} \log p\left( ]x_i - \frac{\delta}{2}; x_i + \frac{\delta}{2}[\ \middle| \ y \right).$$

Then, we assume that the random variables $X_i|Y$ follow a probability density function $f_{\theta_{i,y}}$. Remark that

$$\begin{aligned} p\left( ]x_i - \frac{\delta}{2}; x_i + \frac{\delta}{2}[\ \middle| \ y \right) &= \int_{x_i - \frac{\delta}{2}}^{x_i + \frac{\delta}{2}} f_{\theta_{i,y}}(x) dx \\ &\simeq \int_{x_i - \frac{\delta}{2}}^{x_i + \frac{\delta}{2}} f_{\theta_{i,y}}(x_i) dx \\ &= f_{\theta_{i,y}}(x_i)\delta \end{aligned}$$

Therefore, using this approximation we can write

$$\begin{aligned} h_\theta(x) &= \text{argmax}_{y\in\mathcal{Y}} \ \log p(y) + \sum_{i=1}^{k} \log f_{\theta_{i,y}}(x_i)\delta \\ &= \text{argmax}_{y\in\mathcal{Y}} \ \log p(y) + k\log\delta + \sum_{i=1}^{k} \log f_{\theta_{i,y}}(x_i) \\ &= \text{argmax}_{y\in\mathcal{Y}} \ \log p(y) + \sum_{i=1}^{k} \log f_{\theta_{i,y}}(x_i) \end{aligned}$$

which is the formulation of a Naive Bayes classifier for the continuous case.

4 Gaussian Naive Bayes

While the construction presented in the previous section is intuitive, it is not directly applicable in practice. Indeed, we still need to explicitly determine the distribution $f_{\theta_{i,y}}$. Although several approaches might be considered, here we will focus on one of the most widely used methods, which will result in the well-known Gaussian Naive Bayes classifier.

Specifically, we assume that $f_{\theta_{i,y}}$ follow a normal distribution $\mathcal{N}(\mu, \sigma^2)$, where μ denotes the mean of the distribution, and σ^2 the variance. As a common practice, we use the maximum likelihood estimator (mle) to approximate these parameters. Recall that the mle of μ and σ^2 denoted by μ_{mle} and σ^2_{mle} respectively, are given by

$$\mu_{mle} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

and

$$\sigma^2_{mle} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu_{mle})^2$$

where the summations are taken over all values of a specific feature corresponding to samples related to the class y.

Using these approximations, we are able to explicitly compute $f_{\theta_{i,y}}(x_i)$ and thus use the classifier

$$h_\theta(x) = \text{argmax}_{y\in\mathcal{Y}}\ \log p(y) + \sum_{i=1}^{k} \log f_{\theta_{i,y}}(x_i)$$