Estimating Probability Distributions ›

Derivation of Naive Bayes Classifier

The Naive Bayes classifier decomposes a single high dimensional probability estimation problem into many 1-dimensional estimation problems by assuming the features are conditionally independent given the class label. We want to estimate $\hat{\mathrm{P}}(y|\mathbf{x})$ with the Naive Bayes classifier. To do so, we will make use of **Bayes' Rule and the Naive Bayes assumption**.

☆ **Key Points**

First you need to calculate the probabilities of the labels given the observed data, using Bayes' Rule.

Then, you choose y such that this probability is maximized, considering each dimension of the feature vector independently.

Breaking Down Naive Bayes

You can estimate $\hat{\mathrm{P}}(y|\mathbf{x})$ by calculating $\mathrm{P}(y)$ and $\mathrm{P}(\mathbf{x}|y)$, since by Bayes' Rule:

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

Estimating $\mathrm{P}(y = c)$ (the probability that y takes on some value c) is simple - it is just the fraction of data points where $y = c$.

We can write this formally using an indicator variable I , which is 1 when the argument holds and 0 when it doesn't. If y takes on discrete binary values, such as 0 or 1, it simply counts how many times we observe each outcome:

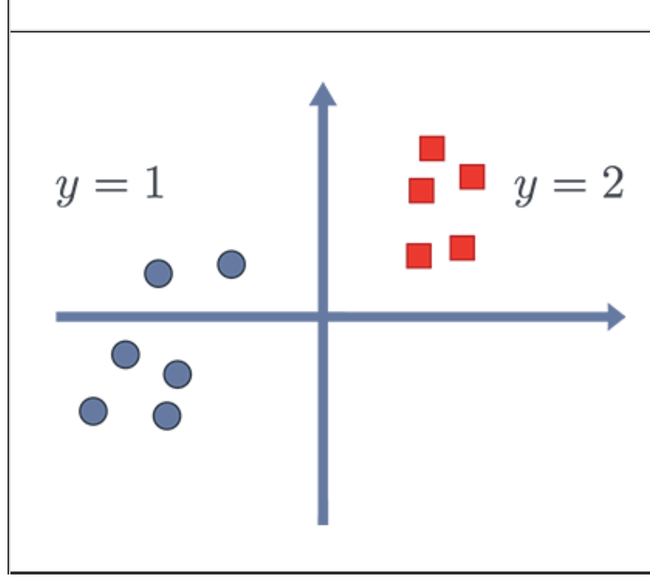
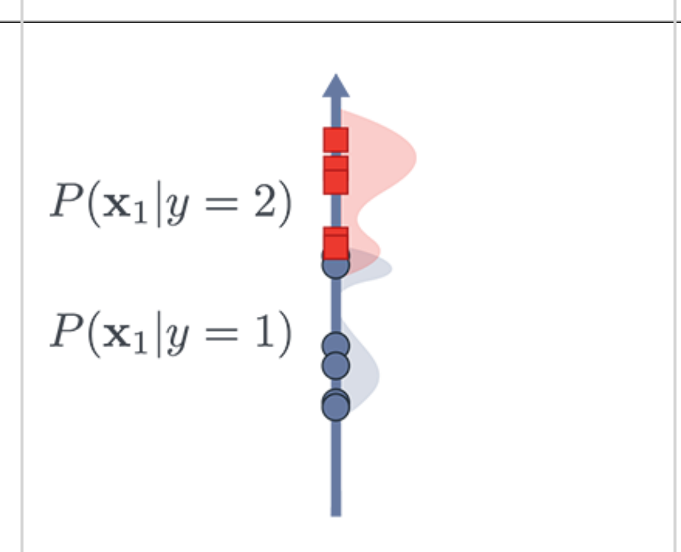
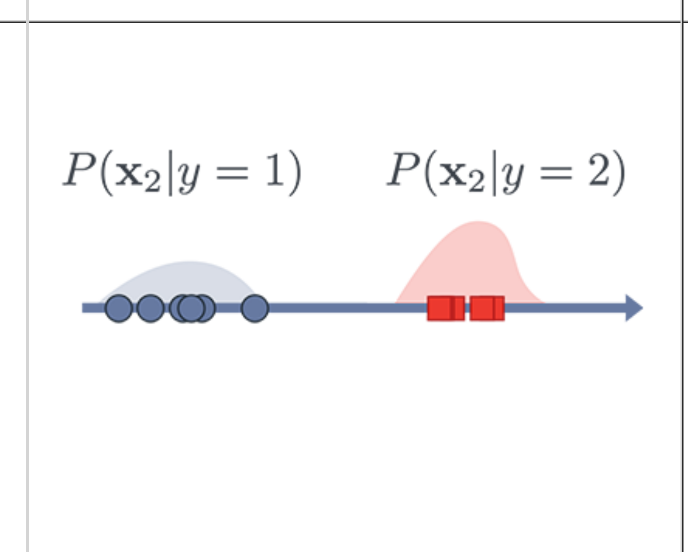
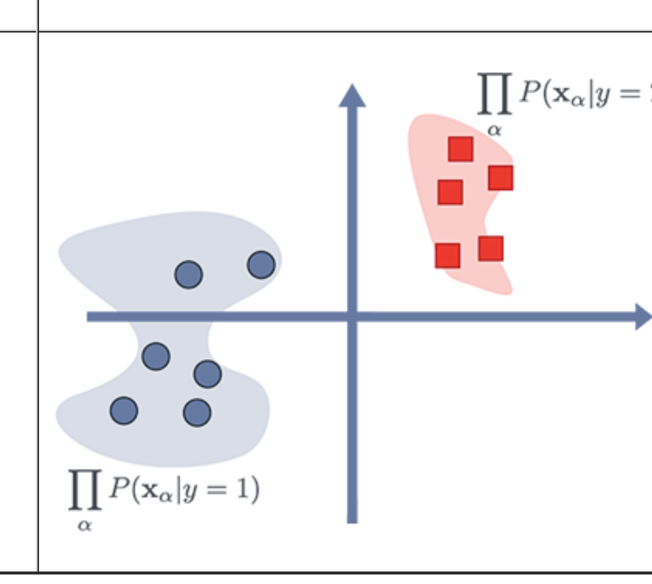
$$\mathrm{P}(y = c) = \frac{\sum_{i=1}^{n} I(y_i = c)}{n} = \hat{\pi}_c$$

Estimating $\mathrm{P}(\mathbf{x}|y)$, however, is not so simple. To do so, you will make use of the Naive Bayes assumption from earlier:

$$\mathrm{P}(\mathbf{x}|y) = \prod_{\alpha=1}^{d} \mathrm{P}(x_\alpha|y)$$

where $x_\alpha = [\mathbf{x}]_\alpha$ is the value for feature α.

Below, you'll see an illustration of how the Naive Bayes algorithm decomposes a multi-dimensional probability estimation problem (first image) into many 1-dimensional estimation problems. First, you estimate $\mathrm{P}(x_\alpha|y)$ independently in each dimension (center two images) and then obtain an estimate of the full data distribution by assuming conditional independence $\mathrm{P}(\mathbf{x}|y) = \prod_\alpha \mathrm{P}(x_\alpha|y)$ (rightmost image).

Original	Estimation of first dimension	Estimation of second dimension	Resulting data distribution
			

Derivation of Naive Bayes Classifier

Now, we can begin our derivation of the Naive Bayes classifier as follows:

$$\begin{aligned}
h(\mathbf{x}) &= \arg\max_y \mathrm{P}(y|\mathbf{x}) \\
&= \arg\max_y \frac{\mathrm{P}(\mathbf{x}|y)\mathrm{P}(y)}{\mathrm{P}(\mathbf{x})} && \text{(Bayes rule)} \\
&= \arg\max_y \mathrm{P}(\mathbf{x}|y)\mathrm{P}(y) && (\mathrm{P}(\mathbf{x}) \text{ does not depend on } y) \\
&= \arg\max_y \prod_{\alpha=1}^{d} \mathrm{P}(x_\alpha|y)\mathrm{P}(y) && \text{(by the Naive Bayes assumption)} \\
&= \arg\max_y \sum_{\alpha=1}^{d} \log(\mathrm{P}(x_\alpha|y)) + \log(\mathrm{P}(y)) && \text{(as log is a monotonic function)}
\end{aligned}$$

Estimating $\mathrm{P}(x_\alpha|y)$ is easy since we only need to consider one single dimension, so MLE should work well. Estimating $\mathrm{P}(y)$ is not affected by the assumption and can similarly done with straight-forward MLE.

◂ Previous

Next ▸