Machine Learning Using Python

Lesson 3: Naive Bayes

Marcel Scharth

The University of Sydney Business School

Lesson 3: Naive Bayes

1. Introduction
2. Naïve Bayes classifier
3. Model evaluation for binary classification
4. Technical note: Bayes' Rule

Introduction

Classification

Consider the following business decision making scenarios.

1. Should we invest resources in acquiring and retaining a customer?
2. Should we offer a mortgage to a credit applicant?
3. Should we invest more resources to train an employee?
4. Should we place a bid to a sponsor an online search?
5. Should we investigate a transaction for possible fraud?

Classification

All these scenarios involve a **classification task**.

1. Do we predict that the customer will be profitable?
2. Do we predict that the applicant will repay the mortgage in full?
3. Do we predict that the employee will stay in the company?
4. Do we predict that the user will click on the ad and make a purchase?
5. Do we flag the transaction?

Classification

In classification, the response variable Y is **qualitative** or **categorical** that takes values in a finite unordered set $\mathcal{Y} = \{1, \ldots, C\}$, where C is the number of classes. Our task is to predict which class a subject belongs to based on input variables.

A **classifier** $\widehat{Y}(\boldsymbol{x})$ is a mapping from the input vector $\boldsymbol{x}$ to $\{1, \ldots, C\}$. A classifier is a prediction rule that assigns the subject to one of the classes, given the observed values of the predictors.

Application: Sentiment Analysis

Sentiment analysis is the problem of classifying text documents according to the sentiment expressed by their authors (for example, positive or negative)

A simple approach is to represent each document as a vector of binary variables, where each element records whether a word is present in the document or not. Hence, $x_{ij} = 1$ if word j appears in document i, and $x_{ij} = 0$ otherwise.

This is called a **bag of words** model.

Application: Sentiment Analysis

Sentiment analysis is the problem of classifying text documents according to the sentiment expressed by their authors (for example, positive or negative).

A simple approach is to represent each document as a vector of binary variables, where each element records whether a word is present in the document or not. Hence, $x_{ij} = 1$ if word j appears in document i, and $x_{ij} = 0$ otherwise.

This is called a **bag of words** model.

Application: Sentiment Analysis

In our application, the response variable will be sentiment $= \{\text{positive}, \text{negative}\}$. We can code the response as

$$Y = \begin{cases} 1 & \text{if positive,} \\ 0 & \text{if negative.} \end{cases}$$

Probabilistic classification

We will mostly rely on probabilistic models for classification. For binary responses, our classifier is

$$\widehat{y}(\boldsymbol{x}) = \begin{cases} 1 \text{ if } P(Y = 1|X = \boldsymbol{x}) > \tau. \\ 0 \text{ if } P(Y = 1|X = \boldsymbol{x}) \leq \tau. \end{cases}$$

where $P(Y = 1|X = \boldsymbol{x})$ is the probability that $Y = 1$ conditional on the predictors (to be learned), and τ is a threshold specified at the problem formulation stage.

Binary classifier

In this lesson we use $\tau = 0.5$, which means that our goal will be to minimise the number of misclassified test cases.

$$\widehat{y}(\boldsymbol{x}) = \begin{cases} 1 \text{ if } P(Y = 1 | X = \boldsymbol{x}) > 0.5. \\ 0 \text{ if } P(Y = 1 | X = \boldsymbol{x}) \leq 0.5. \end{cases}$$

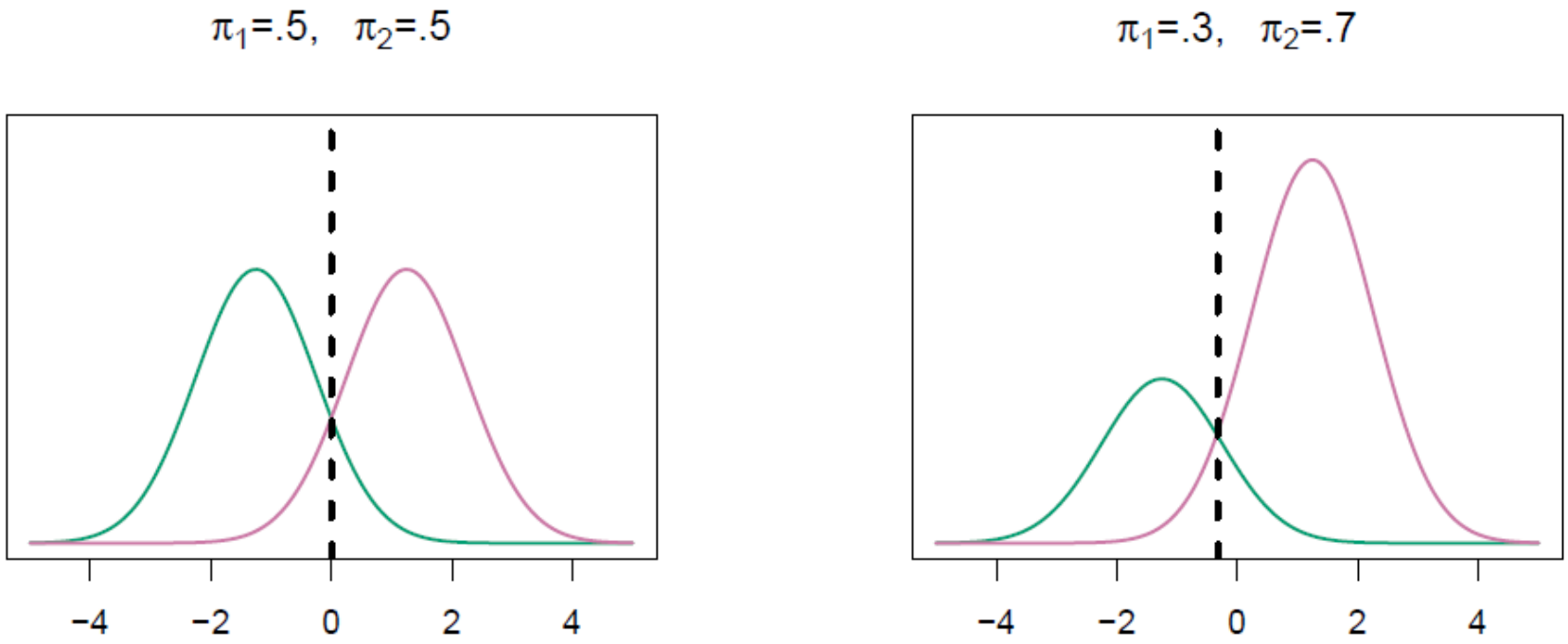
Example: Sentiment Analysis

- Given the contents of a message, what is the probability that it is positive or negative?
- If the probability that the message is positive is higher than 50%, then we classify it as a positive. Otherwise, we classify it as a negative.

Naïve Bayes classifier

Generative classifiers

Consider the case with one normally distributed predictor and two classes. We classify to the highest input density, taking the prior into account.



Generative classifiers

A **generative classifier** is a model that specifies how to generate the data given the **class conditional densities** $p(\boldsymbol{x}|y=c)$ and the (prior) class probabilities $p(y=c)$. This is a model for the joint distribution $p(y, \boldsymbol{x})$.

We compute the conditional probabilities for classification using Bayes' theorem,

$$p(y=c|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|y=c)p(y=c)}{\sum_{c' \in \mathcal{Y}} p(\boldsymbol{x}|y=c')p(y=c')}.$$

Naïve Bayes classifier (key concept)

The **Naïve Bayes classifier** (NBC) is a simple generative model based on the assumption that the predictors are **conditionally independent** given the class label.

The class conditional density the becomes

$$p(\boldsymbol{x}|y=c) = \prod_{j=1}^{p} p(x_j|y=c).$$

Naïve Bayes classifier

- The method is "naive" because we do not think that the features are in fact conditionally independent.
- The simplicity of the NBC method makes it relatively immune to overfitting, which is useful for applications where the number of features is large.
- The assumption of conditional independence makes it easy to mix and match different predictor types.

Naïve Bayes classifier

- Despite being based on an assumption that is not true, the Naïve Bayes classifier often performs very well in practice compared to more complex alternatives,

- The reason is again the bias-variance trade-off: while the assumption of class-conditional independence may lead to highly biased probabilities, the simplifications brought by it may lead to substantial savings in variance.

Model evaluation for binary classification

Classification outcomes

We adopt the following standard terminology.

		Classification	
		$\widehat{Y} = 0$	$\widehat{Y} = 1$
Actual	$Y = 0$	True negative	False positive
	$Y = 1$	False negative	True positive

Confusion matrix (key concept)

A **confusion matrix** counts the number of true negatives, false positives, false negatives, and true positives for the test data.

		Classification		
		$\widehat{Y} = 0$	$\widehat{Y} = 1$	Total
Actual	$Y = 0$	True negatives (TN)	False positives (FP)	N
	$Y = 1$	False negatives (FN)	True positives (TP)	P
	Total	Negative predictions	Positive predictions	

Sensitivity and specificity (key concepts)

The **sensitivity**, recall, or true positive rate is

$$P(\widehat{Y} = 1|Y = 1) = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{\text{True positives}}{\text{Actual positives}}.$$

The **specificity** is

$$P(\widehat{Y} = 0|Y = 0) = \frac{\text{TN}}{\text{TN} + \text{FP}} = \frac{\text{True negatives}}{\text{Actual negatives}}.$$

Trade-off between sensitivity and specificity (key concept)

- There is a trade-off between sensitivity and specificity, since a classifier can always obtain maximum sensitivity (specificity) by setting $\tau = 0$ ($\tau = 1$) and classifying all cases as positive (negative).

- Equivalently, this is a trade-off between sensitivity and achieving a lower false positive rate.

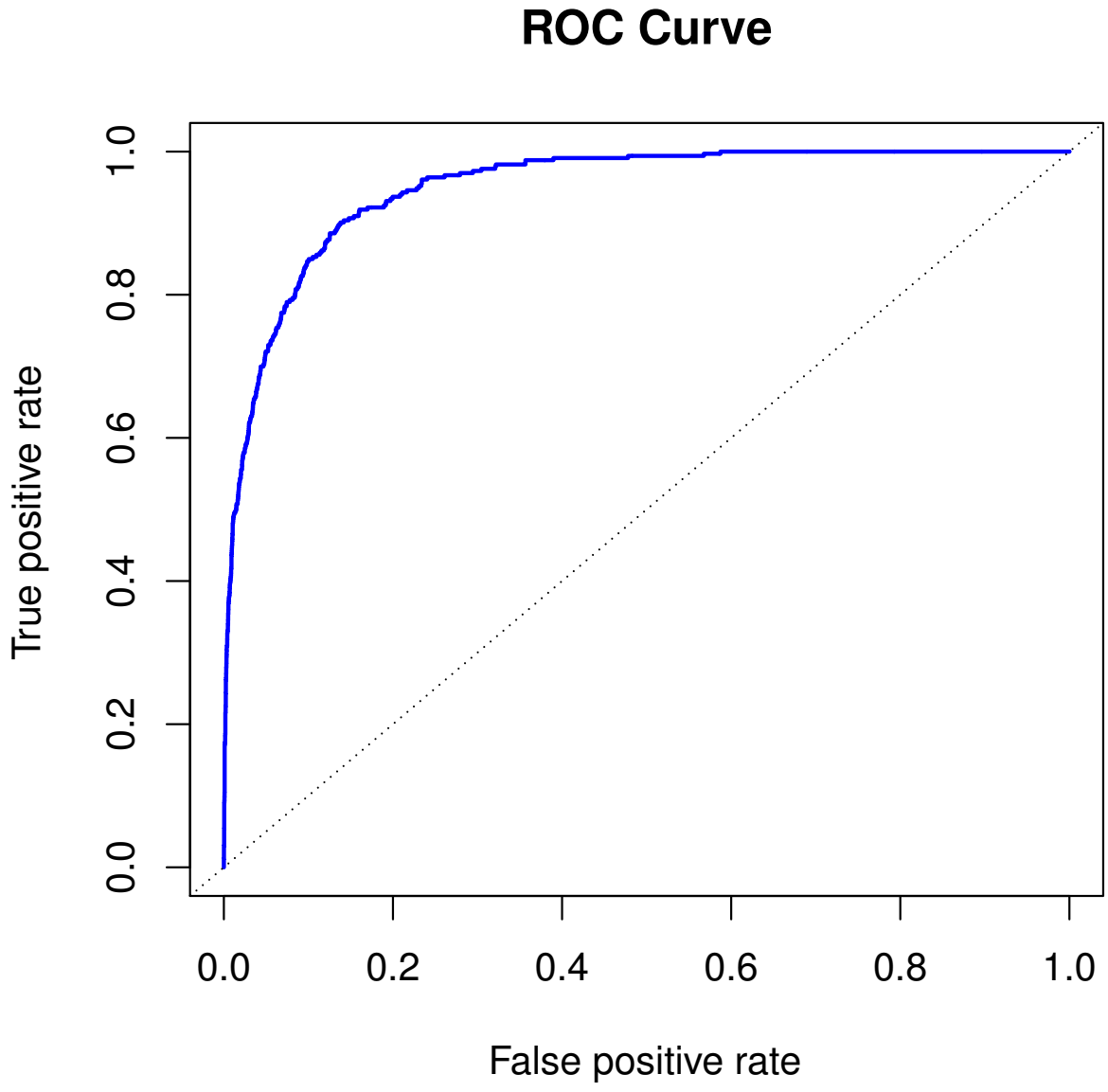
ROC curve (key concept)

A **receiver operating characteristic** or **ROC** curve plots the sensitivity against specificity or the false positive rate for a range of threshold values τ.

We can read the the ROC plot as telling us the false positive rate that we need to accept to obtain a given level of sensitivity.

We often summarise the quality of ROC curve as a single number using the **area under the curve** or **AUC**. Higher AUC scores are better, with a maximum of one.

ROC curve


ROC Curve
True positive rate
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0

Imbalanced classes

Many classification scenarios (such as fraud detection) concern rare events, leading to a very large proportion of negatives in the data.

In this situation we say that the classes are highly **imbalanced**.

The specificity is not very informative for these problems, as it will tend to be high regardless of the quality of the classifier (nearly all transactions are legitimate and classified as such).

Precision (key concept)

In the imbalanced scenario, we are usually more interested in the proportion of detections that are actually positive. We define the **precision** as

$$P(Y = 1|\widehat{Y} = 1) = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{\text{True positives}}{\text{Positive classifications}}$$

Precision recall curve

A **precision recall curve** plots the precision against the recall (sensitivity) as we vary the threshold τ. The mean precision (averaging over recall values) approximates the area under the precision recall curve.

Technical note: Bayes' Rule

Bayes' rule

Let X and Y be two discrete random variables. We state the **Bayes' rule** or **Bayes' theorem** as

$$P(Y = y|X = x) = \frac{P(Y = y, X = x)}{P(X = x)}$$

$$= \frac{P(X = x|Y = y)P(Y = y)}{\sum_{y' \in \mathcal{Y}} P(X = x|Y = y')P(Y = y')}$$

Example: medical test

A mammogram is a medical test for breast cancer. Suppose that the test has a **sensitivity** of 80%, which means that if a woman has cancer, the test will return positive with probability 0.8,

$$P(X = 1|Y = 1) = 0.8.$$

In case of a positive test result, what is the probability that a woman has cancer?

Example: medical test

Using Bayes' theorem, the conditional probability $P(Y = 1|X = 1)$ is

$$\frac{P(X = 1|Y = 1)P(Y = 1)}{P(X = 1|Y = 1)P(Y = 1) + P(X = 1|Y = 0)P(Y = 0)}$$

This equation tells us that in order to calculate the desired probability, we also need to know the **prevalence** of breast cancer $P(Y = 1)$ and the **false positive rate** $P(X = 1|Y = 0)$.

Ignoring the prevalence is a logical mistake known as the **base rate fallacy**.

Example: medical test

Suppose that $P(Y = 1) = 0.004$ and $P(X = 1|Y = 0) = 0.1$.

$$P(Y = 1|X = 1) = \frac{0.8 \times 0.004}{0.8 \times 0.004 + 0.1 \times 0.996} = 0.031$$