Naive Bayes classification

Christos Dimitrakakis

December 4, 2015

1 Introduction

One of the most important methods in machine learning and statistics is that of Bayesian inference. This is the most fundamental method of drawing conclusions from data and explicit prior assumptions. In Bayesian inference, prior assumptions are represented as a probabilities on a space of hypothesis. Each hypothesis is seen as a probabilistic model of all possible data that we can see.

2 The basics of Bayesian inference

Frequently, we want to draw conclusions from data. However, the conclusions are never solely inferred from data, but also depend on prior assumptions about reality.

Example 2.1. John claims he has psychic powers and can predict a series of coin tosses. We oblige, and throw a coin 8 times. John predicts 8 out of 8 coin tosses. The probability of him doing so by chance is 2^{-8}. If he was a medium, as he claims, then his probability of achieving the feat would be 1. Should we believe John?

Example 2.2. Traces of DNA are found at a murder scene. We perform a DNA test against a database of 10^4 citizens registered to be living in the area. We know that the probability of a false positive (that is, the test finding a match by mistake) is 10^{-6}. If there is a match in the database, does that mean that the citizen was at the scene of the crime?

Answering these questions requires us to clearly define what are the possible hypotheses we wish to consider. Taking the first example, we can define two:

1. hypothesis M, that John is a medium.
2. hypothesis $\neg M$, that John is not a medium.

We can also define a probability model for the number of successful predictions that John would make in either case.

Let x_t be 0 if John makes an incorrect prediction at time t and $x_t = 1$ if he makes a correct prediction. John's claim that he can predict our tosses perfectly means that for a sequence of tosses $\boldsymbol{x} = x_1, \ldots, x_n$,

$$\mathbb{P}(\boldsymbol{x} \mid M) = \begin{cases} 1, & x_t = 1 \forall t \in [n] \\ 0, & \exists t \in [n] : x_t = 0. \end{cases}$$

That is, the probability of perfectly correct predictions is 1, and that of one or more incorrect prediction is 0. For the other model, we can assume that all draws are independently and identically distributed from a fair coin. Consequently, no matter what John's predictions are, we have that:

$$\mathbb{P}(\boldsymbol{x} \mid \neg M) = 2^{-n}.$$

So, for the given example, as stated, we have the following facts:

- If John makes one or more mistakes, then $\mathbb{P}(\boldsymbol{x} \mid M) = 0$ and $\mathbb{P}(\boldsymbol{x} \mid \neg M) = 2^{-n}$. Thus, we should perhaps say that then John is not a medium
- If John makes no mistakes at all, then

$$\mathbb{P}(\boldsymbol{x} \mid M) = 1, \qquad \mathbb{P}(\boldsymbol{x} \mid \neg M) = 2^{-n}. \tag{2.1}$$

Does that mean that we must conclude that John is a medium? What if $n = 1$? What if $n = 100$? In fact, our conclusion should somehow depend on the strength of the evidence. Should it also not depend on how likely we think that a medium exists?

It is this latter idea that we'll try and exploit. We'd like to combine the weight of the evidence, with the weight of our prior beliefs about reality. To do this, we first recall the definition of conditional probability.

2.1 Conditional probability and Bayesian inference

Let A and B be two events. Let $P(A)$ be the probability of event A and $P(B)$ the probability of event B. We can think of the probability of an event as the *relative size* of the event in the space of probabilities.[1] The probability of both events A and B happening at the same time is denoted by $P(A \cap B)$. This amounts to measuring the size of the space by the intersection of A and B. The basic probability laws are the following.

Axioms of probability

1. The probability of the certain event is $P(\Omega) = 1$

[1] More formally, probability is a measure; a function similar to volume, area, length and mass.

2. The probability of the impossible event is $P(\emptyset) = 0$
3. The probability of any event A is $1 \leq P(A) \geq 0$.
4. If A, B are disjoint, i.e. $A \cap B = \emptyset$, meaning that they cannot happen at the same time, then

$$P(A \cup B) = P(A) + P(B)$$

Sometimes we would like to calculate the probability of some event A happening given that we know that some other event B has happened. For this we need to first define the idea of conditional probability.

Definition 2.1 (Conditional probability)**.** The probability of A happening if we know that B has happened is defined to be:

$$P(A \mid B) \triangleq \frac{P(A \cap B)}{P(B)}.$$

Here, the probability measure of any event A given B is defined to be the probability of the intersection of of the events divided by the second event. We can rewrite this definition as follows, by using the definition for $P(B \mid A)$

Bayes's theorem

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}.$$

Now let us apply this idea to our specific problem. This allows us to calculate the probability of John being a medium, given the data:

$$\mathbb{P}(M \mid \boldsymbol{x}) = \frac{\mathbb{P}(\boldsymbol{x} \mid M)\,\mathbb{P}(M)}{\mathbb{P}(\boldsymbol{x})},$$

where

$$\mathbb{P}(\boldsymbol{x}) = \mathbb{P}(\boldsymbol{x} \mid M)\,\mathbb{P}(M) + \mathbb{P}(\boldsymbol{x} \mid \neg M)\,\mathbb{P}(\neg M).$$

The only thing left to specify is $\mathbb{P}(M)$, the probability that John is a medium before seeing the data. This is our subjective prior belief that mediums exist and that John is one of them.

More generally, we can think of Bayesian inference as follows:

- We start with a set of mutually exclusive hypotheses $H = \{M_1, \ldots, M_k\}$.
- Each hypothesis M is represented by a specific probabilistic model for any possible data $\boldsymbol{x}$, that is $\mathbb{P}(\boldsymbol{x} \mid M)$.

- For each hypothesis, we have a prior probability $\mathbb{P}(M)$ that it is correct.
- After observing the data, we can calculate a posterior probability that the hypothesis is correct:

$$\mathbb{P}(M \mid \boldsymbol{x}) = \frac{\mathbb{P}(\boldsymbol{x} \mid M)\,\mathbb{P}(M)}{\sum_{i=1}^{k} \mathbb{P}(\boldsymbol{x} \mid M_i)\,\mathbb{P}(M_i)}.$$

Combining the prior belief with evidence is key in this procedure. Our posterior belief can then be used as a new prior belief when we get more evidence.

3 Naive Bayes classifiers

One special case of this idea is in classification, when each hypothesis corresponds to a specific class. Then, given a new example vector of data $\boldsymbol{x}$, we would like to calculate the probability of different classes C given the data, $\mathbb{P}(C \mid \boldsymbol{x})$.

From Bayes's theorem, we see that we can write this as

$$\mathbb{P}(C \mid \boldsymbol{x}) = \frac{\mathbb{P}(\boldsymbol{x} \mid C)\,\mathbb{P}(C)}{\sum_i \mathbb{P}(\boldsymbol{x} \mid C_i)\,\mathbb{P}(C_i)}$$

for any class C. This directly gives us a method for classifying new data, as long as we have a way to obtain $\mathbb{P}(\boldsymbol{x} \mid C)$ and $\mathbb{P}(C)$.

Calculating the prior probability of classes

A simple method is to simply count the number of times each class appears in the training data $\mathcal{D}_T = ((x_t, y_t))_{t=1}^{T}$. Then we can set

$$\mathbb{P}(C) = 1/T \sum_{t=1}^{T} \mathbb{I}\{y_t = C\}$$

The Naive Bayes classifier uses the following model for observations, where observations are independent of each other given the class. Thus, for example the result of three different tests for lung cancer (stethoscope, radiography and biopsy) only depend on whether you have cancer, and not on each other.

Probability model for observations

$$\mathbb{P}(\boldsymbol{x} \mid C) = \mathbb{P}(x(1), \ldots, x(n) \mid C) = \prod_{k=1}^{n} \mathbb{P}(x(k) \mid C).$$

There are two different types of models we can have, one of which is mostly useful for continuous attributes and the other for discrete attributes. In the first, we just need to count the number of times each feature takes different values in different classes.

Discrete attribute model.
Here we simply count the average number of times that the attribute k had the value i when the label was C. This is in fact analogous to the conditional probability definition.

$$\mathbb{P}(x(k) = i \mid C) = \frac{\sum_{t=1}^{T} \mathbb{I}\{x_t(k) = i \wedge y_t = C\}}{\sum_{t=1}^{T} \mathbb{I}\{y_t = C\}} = \frac{N_k(i, C)}{N(C)},$$

where $N_k(i, C)$ is the numb l l .er of examples in class C whose k-th attribute has the value i, and $N(C)$ is the number of examples in class C.

Sometimes we need to be able to deal with cases where there are no examples at all of one class. In that case, that class would have probability zero. To get around this problem, we add "fake observations" to our data. This is called *Laplace smoothing*.

Remark 3.1. In Laplace smoothing with constant λ, our probability model is

$$\mathbb{P}(x(k) = i \mid C) = \frac{\sum_{t=1}^{T} \mathbb{I}\{x_t(k) = i \wedge y_t = C\} + \lambda}{\sum_{t=1}^{T} \mathbb{I}\{y_t = C\} + n_k\lambda} = \frac{N_k(i, C) + \lambda}{N(C) + n_k\lambda}.$$

where n_k is the number of values that the k-th attribute can take. This is necessary, because we want $\sum_{i=1}^{n_k} \mathbb{P}(x(k) = i \mid C) = 1$. (You can check that this is indeed the case as a simple exercise).

Continuous attribute model.
Here we can use a Gaussian model for each continuous dimension.

$$\mathbb{P}(x(k) = v \mid C) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{(v-\mu)^2}{\sigma^2}},$$

where μ and σ are the mean and variance of the Gaussian, typically calculated from the training data as:

$$\mu = \frac{\sum_{t=1}^{T} x_t(k)\, \mathbb{I}\{y_t = C\}}{\sum_{t=1}^{T} \mathbb{I}\{y_t = C\}},$$

i.e. μ is the mean of the k-th attribute when the label is C and

$$\sigma = \frac{\sum_{t=1}^{T} [x_t(k) - \mu]^2\, \mathbb{I}\{y_t = C\}}{\sum_{t=1}^{T} \mathbb{I}\{y_t = C\}},$$

i.e. σ is the variance of the k-th attribute when the label is C. Sometimes we can just fix σ to a constant value, i.e. $\sigma = 1$.

Estimates versus true probabilities

Remember that the probabilities we get from this calculation are only *estimates*. We do not really know the probabilities of each observation given the classes: we are only estimating them from the data. It is also possible that our assumption about the independence of features is completely wrong.

Exercise 1. This is an exercise to get you familiar with the NaiveBayes implementation in the `e1071` library.

- First, open R and install the library `e1071` by doing

  ```
  > install.packages('e1071', dependencies = TRUE)
  ```

- Then load the library:

  ```
  \library('e1071')
  ```

- Then, check the documentation either by going to `http://www.inside-r.org/packages/cran/e1071/docs/naivebayes` or by simply typing

  ```
  > ?naiveBayes
  ```

  in R.

then type the commands in the tutorial. Some explanations are given below.

The following line creates a Naive Bayes model predicting the `Class` variable from all the other variables.

```
1 model <- naiveBayes(Class ~ ., data = Training)
```

There are two ways to predict new data given our model. The first method gives us the labels as outputs

```
1 class.predictions <- predict(model, Holdout )
```

The second method gives us the class probabilities as outputs

```
prob.predictions <- predict(model, Holdout, type = "
    raw") #
```

We can create a contingency table from our class predictions

```
table(class.predictions, Y$Class)
```

More precisely, the command

```
 A <- table(x, y)
```

gives a matrix, whose entry A_{ij} is equal to the number of times that $x_t = i$ and $y_t = j$. Consequently, the sum of terms in the diagonal is the number of correctly classified examples and the sum of the remaining terms it that of the incorrectly classified examples.

Exercise 2. The purpose of this exercise is to explore the effect of the Laplace smoothing parameter λ on Naive Bayes classification. For this exercise use the package DNA:

```
data(DNA, package = "mlbench")
```

The class labels are stored in the column `Class`. Use

- Data points 1–1593 for training (store it in a variable called `Training`)
- Data points 1594–2124 for holdout (store it in a variable called `Holdout`)
- Data point 2125–3186 for testing (store it in a variable called `Testing`)

Then do the following:

1. Use a loop to go through the parameters $\lambda \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ and for each value
   - Train a model using

     ```
     model <- naiveBayes(Class ~ ., data = Training,
         laplace = lambda)
     ```

   - Measure the classification error of the model on the Holdout data using `predict` and do a plot for all different lambda, using a `for` loop.
   - Save the plot in a PDF file using the `pdf` command.
   - Find the best value (with lowest error) for λ on the Holdout set.

- Test the accuracy of the model with the best λ on the Test set.

Then submit the following items *with a private message on Piazza, using the tag [hw4].*

1. Your R code, in a single R file named `MyNameBayes.R` which I should be able to directly run with `source("MyNameBayes.R")`; to produce:
   (a) The hold out classification error for the different lambdas, in a plot. You can generate PDF plots with the `pdf` command (see also my example: `knnExample.R`)
   (b) The value of the best λ, and the resulting testing error classification.
2. An answer to the following questions:
   (a) How does the testing error compare to the training and holdout error? Why do you think this is the case?
   (b) What do you think are the advantages and disadvanages of Naive Bayes over KNN and decision trees?