Naïve Bayes classification in R

Zhongheng Zhang

Department of Critical Care Medicine, Jinhua Municipal Central Hospital, Jinhua Hospital of Zhejiang University, Jinhua 321000, China
Correspondence to: Zhongheng Zhang, MMed. 351#, Mingyue Road, Jinhua 321000, China. Email: zh_zhang1984@hotmail.com.

Author's introduction: Zhongheng Zhang, MMed. Department of Critical Care Medicine, Jinhua Municipal Central Hospital, Jinhua Hospital of Zhejiang University. Dr. Zhongheng Zhang is a fellow physician of the Jinhua Municipal Central Hospital. He graduated from School of Medicine, Zhejiang University in 2009, receiving Master Degree. He has published more than 35 academic papers (science citation indexed) that have been cited for over 200 times. He has been appointed as reviewer for 10 journals, including *Journal of Cardiovascular Medicine*, *Hemodialysis International*, *Journal of Translational Medicine*, *Critical Care*, *International Journal of Clinical Practice*, *Journal of Critical Care*. His major research interests include hemodynamic monitoring in sepsis and septic shock, delirium, and outcome study for critically ill patients. He is experienced in data management and statistical analysis by using R and STATA, big data exploration, systematic review and meta-analysis.

Zhongheng Zhang, MMed.

Abstract: Naïve Bayes classification is a kind of simple probabilistic classification methods based on Bayes' theorem with the assumption of independence between features. The model is trained on training dataset to make predictions by predict() function. This article introduces two functions naiveBayes() and train() for the performance of Naïve Bayes classification.

Keywords: Machine learning; R; naïve Bayes; classification; average accuracy; kappa

Submitted Jan 25, 2016. Accepted for publication Feb 24, 2016.
doi: 10.21037/atm.2016.03.38
View this article at: http://dx.doi.org/10.21037/atm.2016.03.38

Introduction to naïve Bayes classification

Bayes' theorem can be used to make prediction based on prior knowledge and current evidence (1). With accumulating evidence, the prediction is changed. In technical terms, the prediction is the posterior probability that investigators are interested in. The prior knowledge is termed prior probability that reflects the most probable guess on the outcome without additional evidence. The current evidence is expressed as likelihood that reflects the probability of a predictor given a certain outcome. The training dataset is used to derive likelihood (2,3). Bayes' theorem is formally expressed by the following equation.

$$P(A|B)=\frac{P(B|A)\times P(A)}{P(B)} \qquad [1]$$

where P(A) and P(B) are probability of events A and B without regarding each other. P(A|B) is the probability of A conditional on B and P(B|A) is the probability of B conditional on A. In naïve Bayes classification, A is categorical outcome events and B is a series of predictors. The word "naïve" indicates that the predictors are independent on each other conditional on the same outcome value. Therefore $P(b1,b2,b3\mid A)$ can be written as $P(b1\mid A)\times P(b2\mid A)\times P(b3\mid A)$, which makes the calculation process much easier.

I will use an example to illustrate how the naïve Bayes classification works. The example of sepsis diagnosis is employed and the algorithm is simplified. Suppose there are two predictors of sepsis, namely, the respiratory rate and mental status. Septic patients are defined as fast respiratory rate and altered mental status (4-6). The likelihood table is shown in *Table 1*. The table is obtained from training dataset. In Bayes' theorem terms, the likelihood of fast respiratory rate given sepsis is 15/20=0.75, and the likelihood of altered mental status given non-sepsis is 3/80=0.0375. Suppose we have a patient with slow respiratory rate and altered mental status, and we want to make a classification of this patient to either sepsis or non-sepsis.

The prior probabilities of sepsis and non-sepsis are:

P(sepsis)=20/100=0.2

P(non-sepsis)=80/100=0.8

The probabilities of likelihood are:

P(fast RR|sepsis)=15/20=0.75

P(slow RR|sepsis)=5/20=0.25

P(fast RR|non-sepsis)=5/80=0.0625

P(slow RR|non-sepsis)=75/80=0.9375

P(altered mental status| sepsis)=17/20=0.85

P(normal mental status | sepsis)=3/20=0.15

P(altered mental status|non-sepsis)=3/80=0.0375

P(normal mental status |non-sepsis)=77/80=0.9625

Table 1 Likelihood table to make a diagnosis of sepsis

Likelihood	Respiratory rate		Mental status		
	Fast	Slow	Altered	Normal	Total
Sepsis	15/20	5/20	17/20	3/20	20
Non-sepsis	5/80	75/80	3/80	77/80	80
Total	20/100	80/100	20/100	80/100	100

By applying the maximum a posteriori classification rule (7,8), only the numerator of Bayes' equation needs to be calculated. The denominators of each classification are the same. The likelihood of sepsis given slow respiratory rate and altered mental status are:

$$\begin{aligned}&P(sepsis|slow\,RR\cap altered\ mental\ status)\\&=\frac{P(slow|RR\ sepsis)\times P(altered\ mental\ status\ sepsis)\times P(sepsis)}{P(slow\,RR)\times P(altered\ mental\ status)}\\&=\frac{0.25\times 0.85\times 0.2}{P}=\frac{0.0425}{P}\end{aligned} \qquad [2]$$

The probability of non-sepsis can be calculated in a similar fashion:

$$\begin{aligned}&P(non-sepsis|slow\,RR\ altered\ mental\ status)\\&=\frac{P(slow|RR\ non-sepsis)\times P(altered\ mental\ status\ non-sepsis)\times P(non-sepsis)}{P(slow\,RR)\times P(altered\ mental\ status)}\\&=\frac{0.9375\times 0.0375\times 0.8}{P}=\frac{0.028125}{P}\end{aligned} \qquad [3]$$

Since the likelihood of sepsis is greater than non-sepsis (0.0425>0.028125), we classify it into sepsis.

Working example

We employed the Titanic dataset to illustrate how naïve Bayes classification can be performed in R.

```
> data(Titanic)
> str(Titanic)
 table [1:4, 1:2, 1:2, 1:2] 0 0 35 0 0 0 17 0 118 154 ...
 - attr(*, "dimnames")=List of 4
  ..$ Class   : chr [1:4] "1st" "2nd" "3rd" "Crew"
  ..$ Sex     : chr [1:2] "Male" "Female"
  ..$ Age     : chr [1:2] "Child" "Adult"
  ..$ Survived: chr [1:2] "No" "Yes"
```

The dataset is a 4-dimensional array resulting from cross-tabulating 2,201 observations on 4 variables. Because the NaiveBayes() function can pass both data frame and tables, I would like to convert the 4-dimensional array into a data frame with each row represents a passenger. This is also the format with which original data are collected.

```
> countsToCases <- function(x, countcol = "Freq") {
   # Get the row indices to pull from x
   idx <- rep.int(seq_len(nrow(x)), x[[countcol]])
   # Drop count column
   x[[countcol]] <- NULL
   # Get the rows from x
   x[idx, ]
}
> caseTita<-countsToCases(as.data.frame(Titanic))
> head(caseTita)
          Class     Sex      Age      Survived
3         3rd       Male     Child    No
3.1       3rd       Male     Child    No
3.2       3rd       Male     Child    No
3.3       3rd       Male     Child    No
3.4       3rd       Male     Child    No
3.5       3rd       Male     Child    No
> nrow(caseTita)
[1] 2201
```

The as.data.frame() function converts the array into a data fame with each row representing the unique combinations of all variables. Then the custom-made function countsToCases() (http://www.cookbook-r.com) is employed to expand rows with more than one observations. The new dataset *caseTita* contains 2201 rows and four columns, which is exactly what we want.

Naïve Bayes classification with e1071 package

The e1071 package contains a function named naiveBayes() which is helpful in performing Bayes classification (9). The function is able to receive categorical data and contingency table as input. It returns an object of class "naiveBayes". This object can be passed to predict() to predict outcomes of unlabeled subjects.

```
> install.packages("e1071")
> library(e1071)
```

```
> model <- naiveBayes(Survived ~ ., data = caseTita)
> predict(model, caseTita[sample(1:2201,10,replace=FAL
SE),])
 [1] No  No  No  No  No  No  No  No  No  Yes
Levels: No Yes
```

In the above example, a training model is created by using naiveBayes() function. The model is used to predict the survival status of a random sample of ten passengers. Here we used sample() function to select 10 passengers without replacement. The predicted survival status of them is "No" for nine passengers and "yes" for the last one. If you want to examine the conditional a-posterior probabilities, a character value "raw" should be assigned to the type argument.

```
> predict(model, caseTita[sample(1:2201,10,replace=FALSE
),],type="raw")
               No              Yes
[1,]           0.72478200      0.2752180
[2,]           0.72478200      0.2752180
[3,]           0.84661708      0.1533829
[4,]           0.09927006      0.9007299
[5,]           0.85522172      0.1447783
[6,]           0.72478200      0.2752180
[7,]           0.84661708      0.1533829
[8,]           0.85522172      0.1447783
[9,]           0.52792424      0.4720758
[10,]          0.52792424      0.4720758
```

The naiveBayes() function receives contingency table as well. The example below is also presented in the help file of the naiveBayes() function.

```
> m <- naiveBayes(Survived ~ ., data = Titanic)
> m
Naive Bayes Classifier for Discrete Predictors
Call:
naiveBayes.formula(formula = Survived ~ ., data = Titanic)
A-priori probabilities:
Survived
No                        Yes
0.676965                  0.323035

Conditional probabilities:
             Class
```

```
Survived 1st        2nd        3rd        Crew
No       0.08187919 0.11208054 0.35436242 0.45167785
Yes      0.28551336 0.16596343 0.25035162 0.29817159

         Sex
Survived Male       Female
No       0.91543624 0.08456376
Yes      0.51617440 0.48382560

         Age
Survived Child      Adult
No       0.03489933 0.96510067
Yes      0.08016878 0.91983122
```

The a-priori probabilities are prior probability in Bayes' theorem. That is, how frequently each level of class occurs in the training dataset. The rationale underlying the prior probability is that if a level is rare, it is unlikely that such level will occur in the test dataset. In other words, the prediction of an outcome is not only influenced by the predictors, but also by the prevalence of the outcome. Conditional probabilities are calculated for each variable. It is actually the likelihood table as shown in *Table 1*. For example, the likelihood of male given survival P(Male|Survived) equals to 0.51617440. Similarly, the predict() function can be applied for new passengers with predictors of age, sex and class.

Naïve Bayes classification with caret package

The caret package contains train() function which is helpful in setting up a grid of tuning parameters for a number of classification and regression routines, fits each model and calculates a resampling based performance measure. Let's first install and load the package.

```
> install.packages("caret")
> library(caret)
```

Then the data frame caseTita should be split into the predictor data frame and outcome vector. Remember to convert the outcome variable into a vector instead of a data frame. The later will result in error message.

```
> x<-caseTita[,-4]
> y<-caseTita$Survived
```

The model is trained by using train() function.

```
> model1 <- train(x,y,'nb',trControl=trainControl(method='c
v',number=10))
> model1
Naive Bayes
2201 samples
  3 predictor
  2 classes: 'No', 'Yes'
No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 1981, 1981, 1980, 1981, 1981, 1981,
...
Resampling results across tuning parameters:
usekernel Accuracy  Kappa     Accuracy SD Kappa SD
FALSE     0.7782826 0.4441458 0.01614583  0.04959141
TRUE      0.7782826 0.4441458 0.01614583  0.04959141
Tuning parameter 'fL' was held constant at a value of 0
Accuracy was used to select the optimal model using the
largest value.
The final values used for the model were fL = 0 and
usekernel = FALSE.
```

The first argument is a data frame where samples are in rows and features are in columns. The second argument is a vector containing outcomes for each sample. 'nb' is a string specifying that the classification model is naïve Bayes. The trainController argument tells the trainer to use cross-validation ('cv') with 10 folds. Specifically, the original dataset is randomly divided into 10 equal sized subsamples. Of the 10 subsamples, 9 subsamples are used as training data, and the remaining one subsample is used as the validation data. The cross-validation process is then repeated for 10 times, with each of the 10 subsamples used once as the validation data. The process results in 10 estimates which then are averaged (or otherwise combined) to produce a single estimation (10,11). The output shows a kappa of 0.4, which is not very good. It doesn't matter since this is only an illustration example. The next task is to use the model for prediction.

```
> predict(model1$finalModel,caseTita[sample(1:2201,10,replac
e=FALSE),])$class
12.316 11.225 29.116 11.298 27.19 29.35 28.159 12.78 12.331 11.2
No     No     Yes    No     No    Yes   No     No    No     No
Levels: No Yes
```

The 10 subjects are randomly selected from the original dataset. The first line of the output indicates the row names of the subjects. The second line indicates the survival status by prediction with the model. To compare the predicted results to observed results, a confusion matrix can be useful.

```
> table(predict(model1$finalModel,x)$class,y)
              y
              No        Yes
No            1364      362
Yes           126       349
```

This time the whole dataset was used. It is obvious that the error rate is not low as indicated by off-diagonal numbers.

Summary

The article introduces some basic ideas behind the naïve Bayes classification. It is a sample method in machine learning methods but can be useful in some instances. The training is easy and fast that just requires considering each predictors in each class separately. There are two packages *e1071* and *caret* for the performance of naïve Bayes classification. Key parameters within these packages are introduced.

Acknowledgements

None.

Footnote

Conflicts of Interest: The author has no conflicts of interest to declare.

Cite this article as: Zhang Z. Naïve Bayes classification in R. Ann Transl Med 2016;4(12):241. doi: 10.21037/atm.2016.03.38

References

1. Efron B. Mathematics. Bayes' theorem in the 21st century. Science 2013;340:1177-8.
2. Medow MA, Lucey CR. A qualitative approach to Bayes' theorem. Evid Based Med 2011;16:163-7.
3. López Puga J, Krzywinski M, Altman N. Points of significance: Bayes' theorem. Nat Methods 2015;12:277-8.
4. Zhang Z, Chen L, Ni H. Antipyretic therapy in critically ill patients with sepsis: an interaction with body temperature. PLoS One 2015;10:e0121919.
5. Cohen J, Vincent JL, Adhikari NK, et al. Sepsis: a roadmap for future research. Lancet Infect Dis 2015;15:581-614.
6. Drewry AM, Hotchkiss RS1. Sepsis: Revising definitions of sepsis. Nat Rev Nephrol 2015;11:326-8.
7. Murphy KP. Machine Learning: A Probabilistic Perspective (Adaptive Computation and Machine Learning series). 1st ed. London: The MIT Press; 2012:1.
8. Kononenko I. Machine learning for medical diagnosis: history, state of the art and perspective. Artif Intell Med 2001;23:89-109.
9. Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien [R package e1071 version 1.6-7]. Comprehensive R Archive Network (CRAN); 2014. Available online: https://CRAN.R-project.org/package=e1071
10. Hadorn DC, Draper D, Rogers WH, et al. Cross-validation performance of mortality prediction models. Stat Med 1992;11:475-89.
11. Schumacher M, Holländer N, Sauerbrei W. Resampling and cross-validation techniques: a tool to reduce bias caused by model building? Stat Med 1997;16:2813-27.