Naive Bayes with the Breast Cancer data

Karen Mazidi

In this notebook we compare Naive Bayes and logistic regression on the breast cancer data in package mlbench.

Load the data

The breast cancer data is in the mlbench package. There are 669 observations with 11 columns. Column 1 is an ID that will be ignored later, columns 2-10 are factors specifying information gleaned from bioposies. The final column is the label: benign or malignant. The class distribution is 458 benign to 241 malignant, about 64% benign to 36% malignant.

```
library(mlbench)
data(BreastCancer)
str(BreastCancer)
```

```
## 'data.frame':    699 obs. of  11 variables:
##  $ Id             : chr  "1000025" "1002945" "1015425" "1016277" ...
##  $ Cl.thickness   : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 5 5 3 6 4 8 1 2 2 4 ...
##  $ Cell.size      : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 1 4 1 8 1 10 1 1 1 2 ...
##  $ Cell.shape     : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 1 4 1 8 1 10 1 2 1 1 ...
##  $ Marg.adhesion  : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 1 5 1 1 3 8 1 1 1 1 ...
##  $ Epith.c.size   : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 2 7 2 3 2 7 2 2 2 2 ...
##  $ Bare.nuclei    : Factor w/ 10 levels "1","2","3","4",..: 1 10 2 4 1 10 10 1 1 1 ...
##  $ Bl.cromatin    : Factor w/ 10 levels "1","2","3","4",..: 3 3 3 3 3 9 3 3 1 2 ...
##  $ Normal.nucleoli: Factor w/ 10 levels "1","2","3","4",..: 1 2 1 7 1 7 1 1 1 1 ...
##  $ Mitoses        : Factor w/ 9 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 5 1 ...
##  $ Class          : Factor w/ 2 levels "benign","malignant": 1 1 1 1 1 2 1 1 1 1 ...
```

```
summary(BreastCancer$Class)
```

```
##    benign malignant
##       458       241
```

Divide data into train, test

First remove the Id column, then divide into 80% train, 20% test.

```
set.seed(1234)
df <- BreastCancer[,-1] # remove ID
i <- sample(1:nrow(df), 0.8*nrow(df), replace=FALSE)
train <- df[i,]
test <- df[-i,]
```

logistic regression

Build a logistic regression model.

```r
glm1 <- glm(Class~., data=train, family=binomial)
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```r
summary(glm1)
```

```
##
## Call:
## glm(formula = Class ~ ., family = binomial, data = train)
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -4.099e-05  -2.100e-08  -2.100e-08   2.100e-08   4.278e-05
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)            6.385 165538.648   0.000    1.000
## Cl.thickness.L        54.219 151485.142   0.000    1.000
## Cl.thickness.Q        19.629 199108.155   0.000    1.000
## Cl.thickness.C        -4.280 107373.266   0.000    1.000
## Cl.thickness^4        -7.838 151342.702   0.000    1.000
## Cl.thickness^5         9.077 148364.155   0.000    1.000
## Cl.thickness^6         5.817 220832.958   0.000    1.000
## Cl.thickness^7        -5.236 160772.983   0.000    1.000
## Cl.thickness^8       -27.742 104519.771   0.000    1.000
## Cl.thickness^9         1.945  82671.331   0.000    1.000
## Cell.size.L          -19.072 299266.140   0.000    1.000
## Cell.size.Q          -18.416 203837.115   0.000    1.000
## Cell.size.C            4.598 240915.106   0.000    1.000
## Cell.size^4           33.112 313352.825   0.000    1.000
## Cell.size^5           30.762 139842.679   0.000    1.000
## Cell.size^6           17.643 185171.691   0.000    1.000
## Cell.size^7           14.657 233742.894   0.000    1.000
## Cell.size^8           35.910 196991.493   0.000    1.000
## Cell.size^9           25.312 222420.621   0.000    1.000
## Cell.shape.L          65.177 278651.322   0.000    1.000
## Cell.shape.Q          11.994 209955.628   0.000    1.000
## Cell.shape.C         -12.236 221931.624   0.000    1.000
## Cell.shape^4         -50.374 394995.528   0.000    1.000
## Cell.shape^5         -71.246 297663.230   0.000    1.000
## Cell.shape^6         -35.971 167293.078   0.000    1.000
## Cell.shape^7         -19.564 127980.759   0.000    1.000
## Cell.shape^8           8.277 139567.678   0.000    1.000
## Cell.shape^9         -11.813 229864.717   0.000    1.000
## Marg.adhesion.L       22.082 310523.466   0.000    1.000
## Marg.adhesion.Q        6.683 221485.925   0.000    1.000
## Marg.adhesion.C        9.599 404255.572   0.000    1.000
## Marg.adhesion^4      -25.000 302006.356   0.000    1.000
## Marg.adhesion^5      -25.933 217459.738   0.000    1.000
## Marg.adhesion^6        2.511 240574.406   0.000    1.000
## Marg.adhesion^7       15.349 257923.317   0.000    1.000
## Marg.adhesion^8        4.714 234309.207   0.000    1.000
## Marg.adhesion^9       42.449 135949.401   0.000    1.000
```

```
## Epith.c.size.L        39.687 315288.607   0.000    1.000
## Epith.c.size.Q         5.045 230840.676   0.000    1.000
## Epith.c.size.C       -32.798 197765.956   0.000    1.000
## Epith.c.size^4        -9.349 302469.279   0.000    1.000
## Epith.c.size^5       -38.673 422276.808   0.000    1.000
## Epith.c.size^6       -44.869 438676.621   0.000    1.000
## Epith.c.size^7       -28.939 250143.258   0.000    1.000
## Epith.c.size^8         7.701 219323.587   0.000    1.000
## Epith.c.size^9        -2.188 194706.971   0.000    1.000
## Bare.nuclei2          -2.531 220207.064   0.000    1.000
## Bare.nuclei3          17.034  86709.030   0.000    1.000
## Bare.nuclei4          27.799 101840.725   0.000    1.000
## Bare.nuclei5          41.710  28255.501   0.001    0.999
## Bare.nuclei6          55.160 261690.910   0.000    1.000
## Bare.nuclei7          -6.101 384591.603   0.000    1.000
## Bare.nuclei8          29.658 229499.910   0.000    1.000
## Bare.nuclei9          23.882 272442.586   0.000    1.000
## Bare.nuclei10         32.931 142336.804   0.000    1.000
## Bl.cromatin2           4.500 185587.959   0.000    1.000
## Bl.cromatin3          18.073 125580.283   0.000    1.000
## Bl.cromatin4          57.662 120071.762   0.000    1.000
## Bl.cromatin5          13.001 119608.454   0.000    1.000
## Bl.cromatin6           6.836 212062.515   0.000    1.000
## Bl.cromatin7          22.880 126201.274   0.000    1.000
## Bl.cromatin8           4.750 301582.646   0.000    1.000
## Bl.cromatin9          23.598 282337.549   0.000    1.000
## Bl.cromatin10          5.519 346407.750   0.000    1.000
## Normal.nucleoli2      -7.712 150911.764   0.000    1.000
## Normal.nucleoli3      25.460 157715.498   0.000    1.000
## Normal.nucleoli4      -2.431 145578.568   0.000    1.000
## Normal.nucleoli5      -1.970 266915.549   0.000    1.000
## Normal.nucleoli6      17.058 136763.610   0.000    1.000
## Normal.nucleoli7     -59.555 149577.941   0.000    1.000
## Normal.nucleoli8     -22.763 150761.724   0.000    1.000
## Normal.nucleoli9      31.634 329999.584   0.000    1.000
## Normal.nucleoli10     31.950 257595.942   0.000    1.000
## Mitoses2               2.583  95687.028   0.000    1.000
## Mitoses3              16.032 267200.002   0.000    1.000
## Mitoses4              25.354 239744.839   0.000    1.000
## Mitoses5              -1.884 366428.154   0.000    1.000
## Mitoses6             -44.471 942832.613   0.000    1.000
## Mitoses7             -30.182 244876.140   0.000    1.000
## Mitoses8               6.504 391179.580   0.000    1.000
## Mitoses10             13.149 445847.800   0.000    1.000
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 7.0934e+02  on 545  degrees of freedom
## Residual deviance: 1.6846e-08  on 465  degrees of freedom
##   (13 observations deleted due to missingness)
## AIC: 162
##
## Number of Fisher Scoring iterations: 25
```

Test

Evaluate on the test data. The logistic regression model gets 91% accuracy.

```
probs1 <- predict(glm1, newdata=test, type="response")
pred1 <- ifelse(probs1>0.5, 2, 1)
print(table(pred1, test$Class))
```

```
##
## pred1 benign malignant
##     1     86         7
##     2      5        39
```

```
acc1 <- mean(pred1==as.integer(test$Class), na.rm=TRUE)
acc1
```

```
## [1] 0.9124088
```

Examine the results using the caret package.

```
library(caret)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
confusionMatrix(factor(pred1), factor(as.integer(test$Class)))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  1  2
##          1 86  7
##          2  5 39
##
##                Accuracy : 0.9124
##                  95% CI : (0.852, 0.9539)
##     No Information Rate : 0.6642
##     P-Value [Acc > NIR] : 8.633e-12
##
##                   Kappa : 0.8015
##
##  Mcnemar's Test P-Value : 0.7728
##
##             Sensitivity : 0.9451
##             Specificity : 0.8478
##          Pos Pred Value : 0.9247
##          Neg Pred Value : 0.8864
##              Prevalence : 0.6642
##          Detection Rate : 0.6277
##    Detection Prevalence : 0.6788
##       Balanced Accuracy : 0.8964
##
##        'Positive' Class : 1
##
```

Build a Naive Bayes classifier

Use the same test and train data for comparison.

```r
library(e1071)
#nb1 <- naiveBayes(train[,-10], train[,10])
nb1 <- naiveBayes(Class~., data=train)
summary(nb1)
```

```
##           Length Class  Mode
## apriori   2      table  numeric
## tables    9      -none- list
## levels    2      -none- character
## isnumeric 9      -none- logical
## call      4      -none- call
```

```r
nb1
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##    benign malignant
## 0.6529517 0.3470483
##
## Conditional probabilities:
##            Cl.thickness
## Y                     1           2           3           4           5
##   benign    0.334246575 0.098630137 0.200000000 0.131506849 0.194520548
##   malignant 0.015463918 0.015463918 0.056701031 0.056701031 0.164948454
##            Cl.thickness
## Y                     6           7           8           9          10
##   benign    0.035616438 0.002739726 0.002739726 0.000000000 0.000000000
##   malignant 0.061855670 0.092783505 0.170103093 0.061855670 0.304123711
##
##            Cell.size
## Y                     1           2           3           4           5
##   benign    0.835616438 0.082191781 0.057534247 0.010958904 0.000000000
##   malignant 0.010309278 0.036082474 0.113402062 0.139175258 0.118556701
##            Cell.size
## Y                     6           7           8           9          10
##   benign    0.005479452 0.002739726 0.002739726 0.002739726 0.000000000
##   malignant 0.103092784 0.082474227 0.113402062 0.015463918 0.268041237
##
##            Cell.shape
## Y                     1           2           3           4           5
##   benign    0.761643836 0.117808219 0.082191781 0.021917808 0.002739726
##   malignant 0.010309278 0.025773196 0.097938144 0.154639175 0.118556701
##            Cell.shape
## Y                     6           7           8           9          10
##   benign    0.005479452 0.005479452 0.002739726 0.000000000 0.000000000
##   malignant 0.103092784 0.108247423 0.108247423 0.036082474 0.237113402
##
##            Marg.adhesion
## Y                     1           2           3           4           5
```

```
##   benign    0.827397260 0.076712329 0.063013699 0.010958904 0.005479452
##   malignant 0.149484536 0.097938144 0.103092784 0.118556701 0.072164948
##             Marg.adhesion
## Y                       6           7           8           9          10
##   benign    0.010958904 0.000000000 0.000000000 0.002739726 0.002739726
##   malignant 0.072164948 0.051546392 0.082474227 0.020618557 0.231958763
##
##             Epith.c.size
## Y                       1           2           3           4           5
##   benign    0.112328767 0.786301370 0.063013699 0.016438356 0.010958904
##   malignant 0.005154639 0.092783505 0.201030928 0.139175258 0.149484536
##             Epith.c.size
## Y                       6           7           8           9          10
##   benign    0.002739726 0.005479452 0.002739726 0.000000000 0.000000000
##   malignant 0.154639175 0.036082474 0.082474227 0.010309278 0.128865979
##
##             Bare.nuclei
## Y                       1           2           3           4           5
##   benign    0.866855524 0.048158640 0.036827195 0.014164306 0.022662890
##   malignant 0.046632124 0.025906736 0.072538860 0.056994819 0.072538860
##             Bare.nuclei
## Y                       6           7           8           9          10
##   benign    0.000000000 0.000000000 0.005665722 0.000000000 0.005665722
##   malignant 0.020725389 0.031088083 0.088082902 0.036269430 0.549222798
##
##             Bl.cromatin
## Y                       1           2           3           4           5
##   benign    0.339726027 0.358904110 0.265753425 0.013698630 0.008219178
##   malignant 0.005154639 0.030927835 0.154639175 0.118556701 0.134020619
##             Bl.cromatin
## Y                       6           7           8           9          10
##   benign    0.002739726 0.010958904 0.000000000 0.000000000 0.000000000
##   malignant 0.036082474 0.278350515 0.108247423 0.046391753 0.087628866
##
##             Normal.nucleoli
## Y                       1           2           3           4           5
##   benign    0.893150685 0.057534247 0.021917808 0.002739726 0.002739726
##   malignant 0.175257732 0.030927835 0.139175258 0.077319588 0.077319588
##             Normal.nucleoli
## Y                       6           7           8           9          10
##   benign    0.005479452 0.005479452 0.008219178 0.002739726 0.000000000
##   malignant 0.072164948 0.046391753 0.077319588 0.041237113 0.262886598
##
##             Mitoses
## Y                       1           2           3           4           5
##   benign    0.978082192 0.013698630 0.002739726 0.000000000 0.000000000
##   malignant 0.525773196 0.108247423 0.134020619 0.051546392 0.025773196
##             Mitoses
## Y                       6           7           8          10
##   benign    0.000000000 0.002739726 0.002739726 0.000000000
##   malignant 0.015463918 0.041237113 0.030927835 0.067010309
```

Evaluate on the test data

The Naive Bayes model gets 96% accuracy.

```
pred2 <- predict(nb1, newdata=test[,-10], type="class")
table(pred2, test$Class)
```

```
##
## pred2       benign malignant
##   benign        87         0
##   malignant      6        47
```

```
acc2 <- mean(pred2==test$Class)
acc2
```

```
## [1] 0.9571429
```

Evaluate the results with the caret package.

```
confusionMatrix(pred2, test$Class, positive="malignant")
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction  benign malignant
##   benign        87         0
##   malignant      6        47
##
##                Accuracy : 0.9571
##                  95% CI : (0.9091, 0.9841)
##     No Information Rate : 0.6643
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.9069
##
##  Mcnemar's Test P-Value : 0.04123
##
##             Sensitivity : 1.0000
##             Specificity : 0.9355
##          Pos Pred Value : 0.8868
##          Neg Pred Value : 1.0000
##              Prevalence : 0.3357
##          Detection Rate : 0.3357
##    Detection Prevalence : 0.3786
##       Balanced Accuracy : 0.9677
##
##        'Positive' Class : malignant
##
```

Set cut-off points for predictors

```
df2 <- df[, c(2:3, 10)]
df2$Cell.size <- as.factor(ifelse(df$Cell.size > 5, 1, 0))
df2$Cell.shape <- as.factor(ifelse(df$Cell.shape > 5, 1, 0))
str(df2)
```

```
## 'data.frame':    699 obs. of  3 variables:
```

```
##  $ Cell.size : Factor w/ 2 levels "0","1": 1 1 1 2 1 2 1 1 1 1 ...
##  $ Cell.shape: Factor w/ 2 levels "0","1": 1 1 1 2 1 2 1 1 1 1 ...
##  $ Class     : Factor w/ 2 levels "benign","malignant": 1 1 1 1 1 2 1 1 1 1 ...
```

```
train2 <- df2[i,]
test2 <- df2[-i,]
```

Re-run logistic regression

```
glm2 <- glm(Class~., data=train2, family=binomial)
summary(glm2)
```

```
##
## Call:
## glm(formula = Class ~ ., family = binomial, data = train2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0635  -0.5672  -0.5672   0.1357   1.9527
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.7457     0.1368 -12.764  < 2e-16 ***
## Cell.size1    3.1586     0.5123   6.166 7.02e-10 ***
## Cell.shape1   3.2705     0.5084   6.433 1.25e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 721.78  on 558  degrees of freedom
## Residual deviance: 403.81  on 556  degrees of freedom
## AIC: 409.81
##
## Number of Fisher Scoring iterations: 6
```

```
probs3 <- predict(glm2, newdata=test2, type="response")
pred3 <- ifelse(probs3>0.5, 2, 1)
print(table(pred3, test$Class))
```

```
##
## pred3 benign malignant
##     1     92        10
##     2      1        37
```

```
acc3 <- mean(pred3==as.integer(test2$Class), na.rm=TRUE)
acc3
```

```
## [1] 0.9214286
```

Re-run naive Bayes

```
nb2 <- naiveBayes(Class~., data=train2)
nb2
```

```
##
```

```
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##    benign malignant
## 0.6529517 0.3470483
##
## Conditional probabilities:
##            Cell.size
## Y                   0          1
##   benign    0.98630137 0.01369863
##   malignant 0.41752577 0.58247423
##
##            Cell.shape
## Y                   0          1
##   benign    0.98630137 0.01369863
##   malignant 0.40721649 0.59278351
```

```
pred4 <- predict(nb2, newdata=test2, type="class")
table(pred4, test$Class)
```

```
##
## pred4       benign malignant
##   benign        92        10
##   malignant      1        37
```

```
acc4 <- mean(pred4==test$Class)
acc4
```

```
## [1] 0.9214286
```