Research on Comparing Accuracy between Tree Method and Logistic Regression on Classification Question

Haoxing Chen, Audrey Hyke, Michael Liu, Melissa Wilson, Chenghan Yue

Executive Summary

Introduction

In order to correctly diagnose cancer, tumors must be analyzed to determine if they're malignant or benign so the appropriate treatment can be recommended. One potential way to differentiate between these two kinds of tumors is to image their cell nuclei. Since the nuclei of cancerous cells tend to be abnormally shaped, by quantifying their morphology the type of tumor can be classified based on the physical features of the nucleus.

Methods

A dataset consisting of 100 observations from prostate tumor samples were obtained from Kaggle. In addition to the patient ID and tumor type, it includes the characteristics radius, texture, perimeter, area, smoothness, compactness, symmetry and fractal dimension which describe cell nuclei properties. Logistic regression and classification trees were used to classify tumor type based on these eight characteristics. The performance of each method was compared to determine which method had higher accuracy.

Results

The logistic regression model correctly classified 88% of observations on the test set and used area and compactness in the final model. The classification tree correctly classified 92% of observations and used perimeter and smoothness in the pruned.

Conclusion

The classification tree method yielded higher prediction accuracy than logistic regression. Therefore, we recommend using a tree-based method when classifying tumor type with nuclei image processing.

Introduction

Prostate cancer represents the second most common cancer among American men (Alteri, 2020). Detecting prostate cancer involves first screening patients for tumors in the prostate and then performing analysis to determine if they are benign or malignant. Benign tumors do not contain cancerous cells, so they do not spread to other tissue. Accurate detection of tumors is critical for the successful treatment of patients. Prostate cancer and benign prostate hyplasia exhibit overlapping symptoms yet require different treatments. Correct diagnosis therefore typically involves imaging tissue samples and blood tests (Sadaf, 2017).

One method of classifying tumors consists of imaging cancer cells and analyzing the physical characteristics of the nuclei of tumor cells. The nuclei of cancer cells tend to be misshapen compared to healthy cells, which tend to be smooth. It is thought that this is caused by changes in expression level of different types of nuclear lamin, leading to higher concentrations of A-type lamins. Since these proteins make up the nuclear envelope and help give the nucleus its shape, these changes can compromise the structural integrity of the nucleus (Funkhouser et al., 2013). Image analysis of nuclei has been shown to be useful in distinguishing between benign and malignant tumors in breast tissue, especially when calculating symmetry and fractal dimension (Chan & Tuszynski, 2016; Yang, Zhang, Chen, Li, & Chen, 2009). Nuclei morphology has also been shown to be related to survival rates and success of cancer treatments (de Andrea, Petrilli, Jesus-Garcia, Bleggi-Torres, & Alves, 2011; Hsu et al., 2016; Nafe, Franz, Schlote, & Schneider, 2005; Wang et al., 2018).

Our research examines how different classification methods perform when using physical characteristics of the nuclei of tumor cells to differentiate prostate tumor types. This semester, our class studied a technique known as the tree method. According to *An Introduction to Statistical Learning*, "Tree method can be summarized that the set of splitting rules used to segment the predictor space" (James, Witten, Hastie, & Tibshirani, 2016). It can be used on both regression and classification problems. One will recall that logistic regression, an old method which we learned earlier, can also be used on both regression and classification problems. Therefore, we are interested in whether a tree or logistic regression model will exhibit higher accuracy, specifically on classification questions like the accurate diagnosis of prostate cancer. The goal of this study is to find out which method has higher accuracy when performing forecasting classification.

Data

The relevant data is compiled from Kaggle (Saifi, 2018). The data set consists of 100 observations and 10 variables: patient ID, diagnosis result, radius, texture, perimeter, area, smoothness, compactness, symmetry and fractal dimension. The characteristics were computed based on the techniques developed at the University of Wisconsin, Madison (Street, Wolberg, & Mangasarian, 1993). The diagnosis result is the outcome of interest consisting of benign or malignant tumor.

Descriptive Statistics

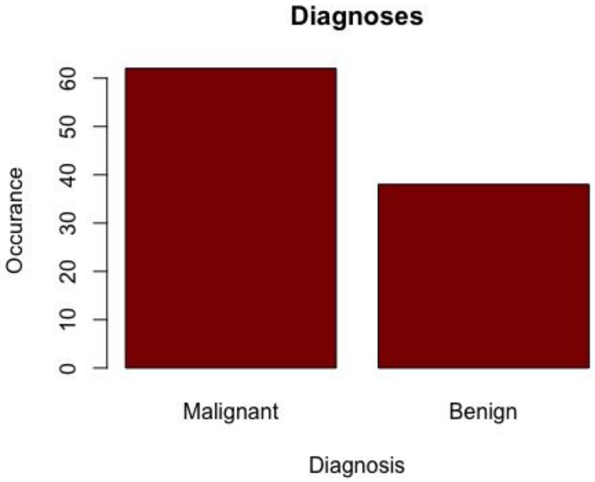


Figure 1: Bar chart for diagnoses frequencies.

A sample size of 100 patients was used in this study. 62% of the cases were malignant, and 38% of them were benign. There were 8 different variables accounted for in each patient. Figure 2 seems to imply that there are different distributions of perimeter, area, compactness, and symmetry based on if the diagnosis was malignant or benign. This means these should be the variables looked into for linear regression.

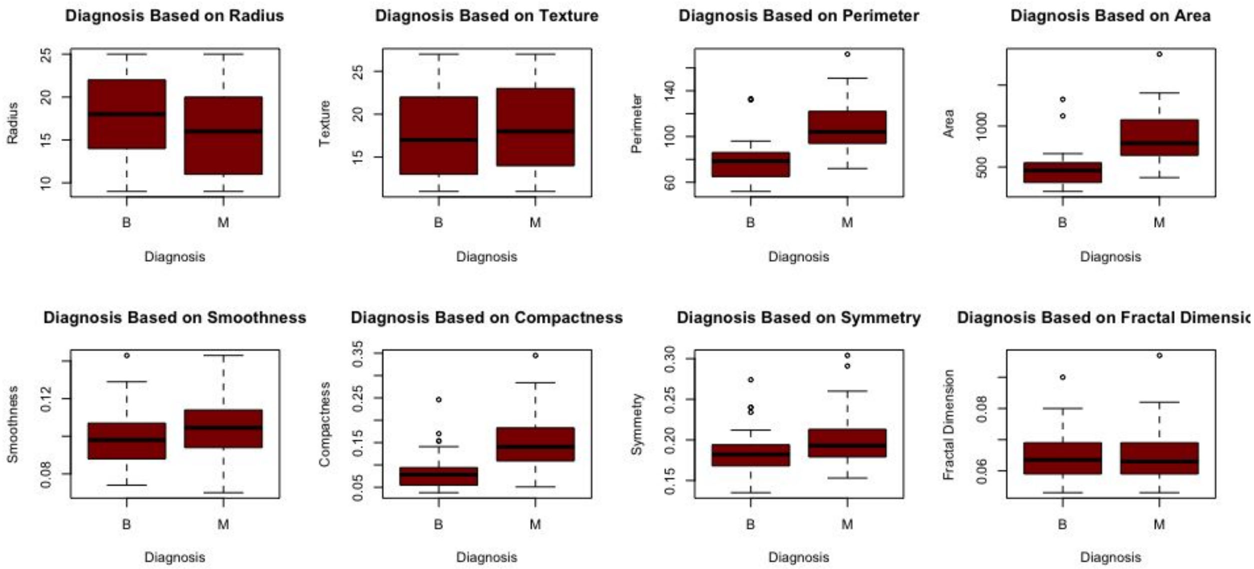


Figure 2: Diagnosis based on each variable.

3. Methodology

The variable we want to predict is diagnosis result. In this study, we split the data equally into training and test sets.

Tree Method

First, we fit the classification tree model on the data and plot the tree. The summary can be found in Code 1 in the appendix. The tree plot is shown below.

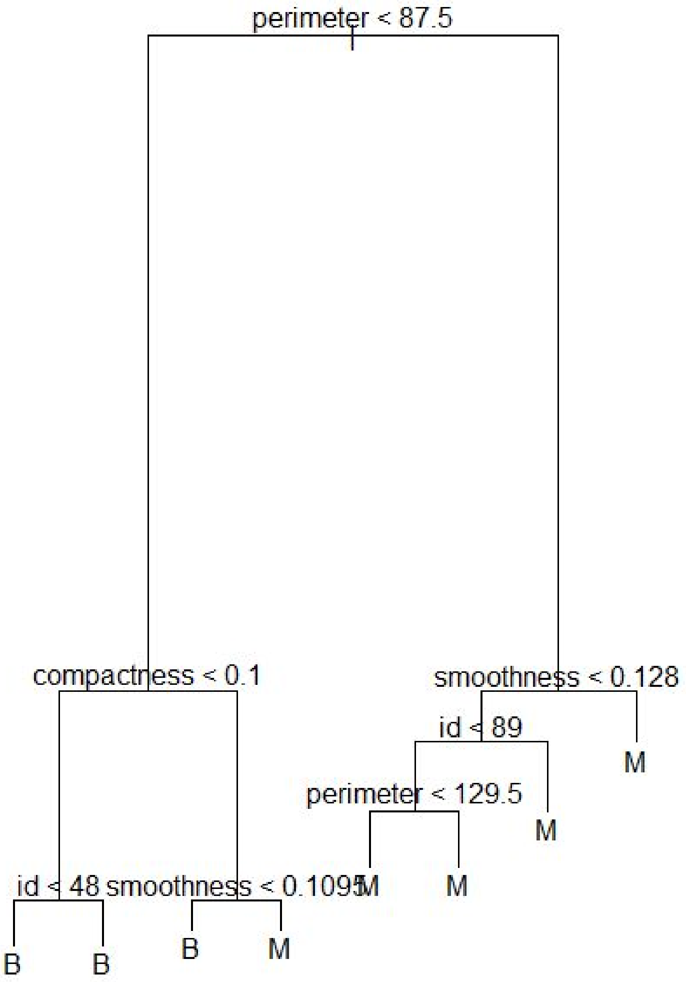


Figure 3. Unpruned tree plot.

According to the figure, we can find that the tree has eight branches and only four of the variables were used to build this tree. The tree method predicts the patient is benign for smaller smoothness (smoothness<0.1095), higher compactness (compactness <0.1), and smaller perimeter (perimeter <87.5). According to the table in Code 2, we can get the accuracy of the model, which is (14+32)/50 =0.92.

Since the tree was grown to full depth, it is possible that there are too many variables that can cause the model to have relatively high variance, low bias, and a higher probability of overfitting the data.

Therefore, we now use cross validation to determine the optimal level of tree complexity. This will help us decide whether pruning the tree will improve performance or not.

According to the cross validation, we can see that the lowest error rate corresponds to the tree with 4 subtrees. Therefore, we can prune the tree to make our tree better. The fixed tree is shown below.

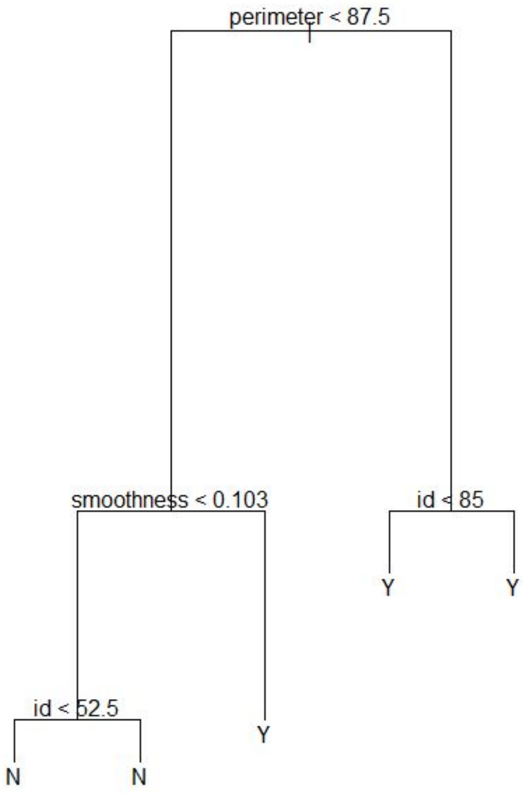


Figure 4: Results after pruning the tree.

According to Figure 4, we can see that the tree has 5 branches and only three variables (perimeter, id and smoothness) have been used to build this tree. Here N means benign, and Y means malignant. The tree method predicts the patient has a malignant tumor for higher smoothness (smoothness<0.103) and smaller perimeter (perimeter <87.5).

Then we are going to use the pruned tree on the test set. By using the pruned tree on the test set, according to the table in Code 3, we can get the accuracy of the model, which is also (14+32)/50 =0.92.

Logistic Regression

According to James, al.(2013), "logistic regression models the probability that Y belongs to a particular category". After setting the model, we can get the summary and the table in Code 4. According to the table, we can get the accuracy of the model, which is (15+29)/50 =0.88.

Results

As a result, the accuracy for tree method is 0.92 and the accuracy for logistic regression is 0.88. We can see that the tree method has a higher accuracy than logistic regression. Therefore, tree method is the most accurate model for predicting whether tumors are benign or malignant.

References

Alteri, R. (2020). Key Statistics for Prostate Cancer. Retrieved May 13, 2020, from https://www.cancer.org/cancer/prostate-cancer/about/key-statistics.html

Chan, A., & Tuszynski, J. A. (2016). Automatic prediction of tumour malignancy in breast cancer with fractal dimension. *Royal Society Open Science*, *3*(12). https://doi.org/10.1098/rsos.160558

de Andrea, C. E., Petrilli, A. S., Jesus-Garcia, R., Bleggi-Torres, L. F., & Alves, M. T. S. (2011). Large and round tumor nuclei in osteosarcoma: Good clinical outcome. *International Journal of Clinical and Experimental Pathology*, *4*(2), 169–174.

Funkhouser, C. M., Sknepnek, R., Shimi, T., Goldman, A. E., Goldman, R. D., & De La Cruz, M. O. (2013). Mechanical model of blebbing in nuclear lamin meshworks. *Proceedings of the National Academy of Sciences of the United States of America*, *110*(9), 3248–3253. https://doi.org/10.1073/pnas.1300215110

Hsu, C. Y. J., Wang, C. W., Kuo, C. C., Liang, H. K., Lai, S. F., Chen, Y. H., & Kuo, S. H. (2016). Tumor Compactness Improves the Pretreatment Volumetry-Based Prediction for Pathological Complete Response of Rectal Cancer After Neoadjuvant Concurrent Chemoradiation Therapy. *International Journal of Radiation Oncology*Biology*Physics*, *96*(2), E185. https://doi.org/10.1016/j.ijrobp.2016.06.1055

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2016). *An Introduction to Statistical Learning* (6th ed.). New York, NY: Springer. https://doi.org/10.1007/978-1-4614-7138-7

Nafe, R., Franz, K., Schlote, W., & Schneider, B. (2005). Morphology of tumor cell nuclei is significantly related with survival time of patients with glioblastomas. *Clinical Cancer Research*, *11*(6), 2141–2148. https://doi.org/10.1158/1078-0432.CCR-04-1198

Sadaf, A. (2017). Prostate carcinoma. Retrieved May 13, 2020, from https://www.cancertherapyadvisor.com/home/decision-support-in-medicine/hospital-medicine/prostate-carcinoma/

Saifi, S. (2018). Prostate Cancer. Retrieved May 13, 2020, from https://www.kaggle.com/sajidsaifi/prostate-cancer

Street, W. N., Wolberg, W. H., & Mangasarian, O. L. (1993). Nuclear feature extraction for breast tumor diagnosis. In R. S. Acharya & D. B. Goldgof (Eds.), *Biomedical Image Processing and Biomedical Visualization* (Vol. 1905, pp. 861–870). SPIE. https://doi.org/10.1117/12.148698

Wang, X., Barrera, C., Velu, P., Bera, K., Prasanna, P., Khunger, M., … Madabhushi, A. (2018). Computer extracted features of cancer nuclei from H&E stained tissues of tumor predicts response to nivolumab in non-small cell lung cancer. *Journal of Clinical Oncology*, *36*(15_suppl), 12061–12061. https://doi.org/10.1200/jco.2018.36.15_suppl.12061

Yang, W., Zhang, S., Chen, Y., Li, W., & Chen, Y. (2009). Shape symmetry analysis of breast tumors on ultrasound images. *Computers in Biology and Medicine*, *39*(3), 231–238. https://doi.org/10.1016/j.compbiomed.2008.12.007

Appendix

Code 1

```
## Classification tree:

## tree(formula = dat$diagnosis_result ~ . – test1, data = dat)

## Variables actually used in tree construction:

## [1] “perimeter”   “compactness” “smoothness”  “area”

## Number of terminal nodes:  9

## Residual mean deviance:  0.4224 = 38.44 / 91

## Misclassification error rate: 0.1 = 10 / 100
```

Code 2: Accuracy for unpruned tree

```
##            Diag.test

## tree.pred1  N  Y

##           N 14  1

##           Y  3 32
```

Code 3: Accuracy for pruned tree

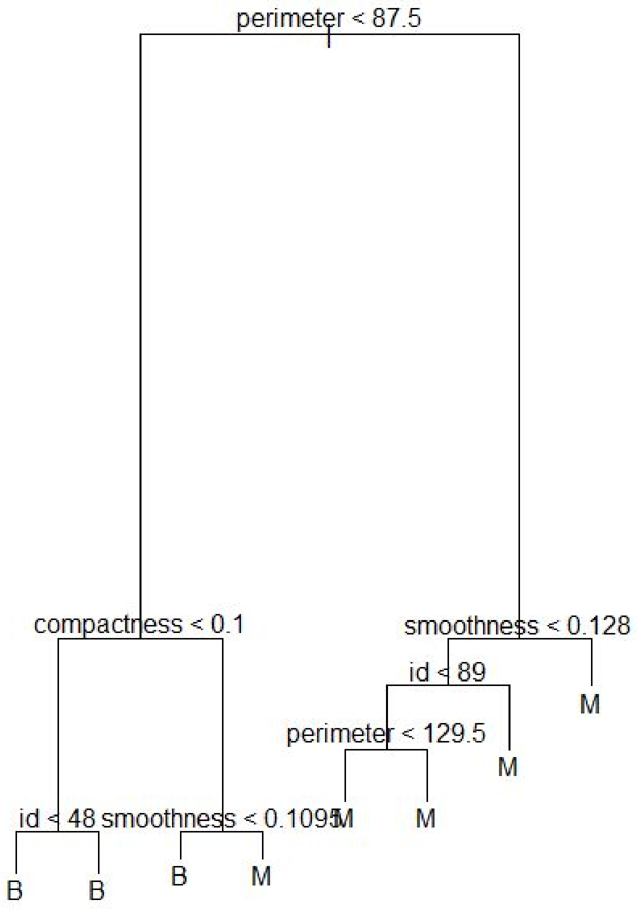
```
##            Diag.test
## tree.pred1  N  Y
##           N 14  1
##           Y  3 32
```

Code 4:

```
Call:
## glm(formula = test2 ~ . – diagnosis_result, family = binomial,
##     data = dattrain2)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.67730  -0.19614   0.01473   0.28961   2.35580
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept) -12.517532   4.182476  -2.993  0.00276 **
## area          0.016763   0.005976   2.805  0.00503 **
## compactness  29.408130  12.914609   2.277  0.02278 *
## ---
## Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 67.301  on 49  degrees of freedom
## Residual deviance: 23.780  on 47  degrees of freedom
## AIC: 29.78
##
## Number of Fisher Scoring iterations: 8
##         Diag.test2
## glm.pred  N  Y
##        Y  3 29
##       N  15  3
```

Whole R code

```
## classification decisioin tree method

set.seed(4052)

dat = read.csv("C:/Users/yuech/Downloads/stat 4052/Prostate_Cancer.csv",
header = T)
library(tree)

set.seed(4052)

test1=ifelse(dat$diagnosis_result=="B","N","Y")
dat=data.frame(dat ,test1)
tree.dat = tree(dat$diagnosis_result~.-test1, data = dat)
##summary
summary(tree.dat)

##
## Classification tree:
## tree(formula = dat$diagnosis_result ~ . - test1, data = dat)
## Variables actually used in tree construction:
## [1] "perimeter"   "compactness" "smoothness"  "area"
## Number of terminal nodes:  9
## Residual mean deviance:  0.4224 = 38.44 / 91
## Misclassification error rate: 0.1 = 10 / 100

##Detailed model structure
tree.dat

## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##  1) root 100 132.800 M ( 0.38000 0.62000 )
##    2) perimeter < 87.5 42  43.640 B ( 0.78571 0.21429 )
##       4) compactness < 0.1 29   8.700 B ( 0.96552 0.03448 )  
##         8) perimeter < 85.5 24   0.000 B ( 1.00000 0.00000 ) *
##         9) perimeter > 85.5 5   5.004 B ( 0.80000 0.20000 ) *
##       5) compactness > 0.1 13  17.320 M ( 0.38462 0.61538 )  
##        10) smoothness < 0.1095 7   9.561 B ( 0.57143 0.42857 ) *
##        11) smoothness > 0.1095 6   5.407 M ( 0.16667 0.83333 ) *
##     3) perimeter > 87.5 58  34.070 M ( 0.08621 0.91379 )  
##       6) smoothness < 0.128 53  23.060 M ( 0.05660 0.94340 )  
##        12) perimeter < 96.5 14  11.480 M ( 0.14286 0.85714 )  
##          24) area < 650 9   0.000 M ( 0.00000 1.00000 ) *
##          25) area > 650 5   6.730 M ( 0.40000 0.60000 ) *
##        13) perimeter > 96.5 39   9.301 M ( 0.02564 0.97436 )  
##          26) compactness < 0.2145 34   0.000 M ( 0.00000 1.00000 ) *
##          27) compactness > 0.2145 5   5.004 M ( 0.20000 0.80000 ) *
##       7) smoothness > 0.128 5   6.730 M ( 0.40000 0.60000 ) *

##plot
plot(tree.dat)
text(tree.dat,pretty=5)
```



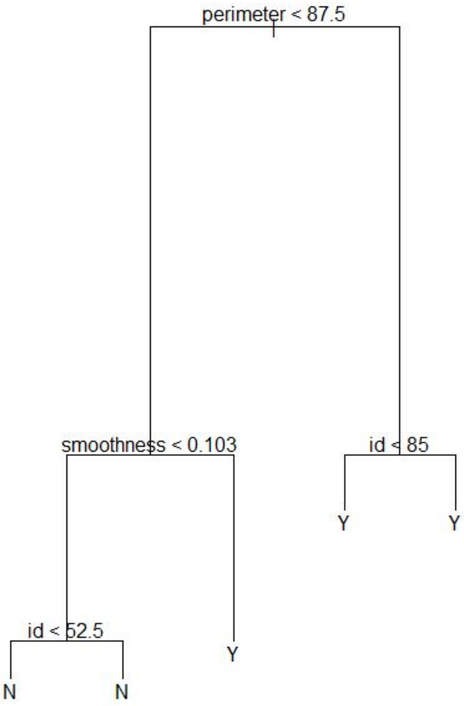
```
##accuracy
set.seed(4052)
train = sample (2:nrow(dat), 50)
dat.train=dat[train,]
dat.test=dat[train, ]
Diag.test=test1[train]
tree.dat2 =tree(test1~.-diagnosis_result , dat.train)
tree.pred1=predict(tree.dat2 ,dat.test ,type="class")
table(tree.pred1 ,Diag.test)
```

```
##            Diag.test
## tree.pred1  N  Y
##           N 14  1
##           Y  3 32
```

```
##prune

##prune test
cv.dat =cv.tree(tree.dat ,FUN=prune.misclass )
names(cv.dat)
```

```
[1] "size"   "dev"    "k"      "method"

> cv.dat

$size

[1] 8 4 3 2 1


$dev

[1] 20 20 19 21 39


$k

[1] -Inf    0    1    3   24


$method

[1] "misclass"


attr(,"class")

[1] "prune"         "tree.sequence"
```

```
##prune
prune.dat =prune.misclass (tree.dat2 ,best=4)

plot(prune.dat )
text(prune.dat ,pretty =0)
```



```
##accuracy
tree.pred2=predict(prune.dat, dat.test , type="class")
table(tree.pred2, Diag.test)

##            Diag.test
## tree.pred2  N  Y
##           N 14  1
##           Y  3 32

##Logistic Regression Classification
library(caret)

## Loading required package: lattice

## Loading required package: ggplot2

library(glmnet)
```

```
## Loading required package: Matrix

## Loaded glmnet 3.0-2

dat2 = read.csv("C:/Users/yuech/Downloads/stat 4052/Prostate_Cancer.csv",
header = T)
##remove predictors with cor less than 0.3
corVec=cor(as.numeric(dat$diagnosis_result),dat2[,2:ncol(dat2)])
notCor=which(abs(corVec)<0.3)
dat2=data.frame(diagnosis_result=dat2[,1],dat2[-c(1,notCor+1)])
##remove predictors with cor larger than 0.85
corMat=cor(dat2[,2:ncol(dat2)])
highCor=findCorrelation(corMat, cutoff=0.85)
dat2=data.frame(diagnosis_result=dat2[,1],dat2[-c(1,highCor+1)])
names(dat2)

## [1] "diagnosis_result" "area"             "compactness"

test2=ifelse(dat2$diagnosis_result=="B","N","Y")
dat2=data.frame(dat2 ,test2)
train2 = sample (2:nrow(dat2), 50)
dattrain2=dat2[train2,]
dattest2=dat2[-train2,]
Diag.test2=test2[-train2]
fit_glm = glm(test2~.-diagnosis_result,dattrain2,family=binomial)
summary(fit_glm)

##
## Call:
## glm(formula = test2 ~ . - diagnosis_result, family = binomial,
##     data = dattrain2)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.67730  -0.19614   0.01473   0.28961   2.35580
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept) -12.517532   4.182476  -2.993  0.00276 **
## area          0.016763   0.005976   2.805  0.00503 **
## compactness  29.408130  12.914609   2.277  0.02278 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 67.301  on 49  degrees of freedom
## Residual deviance: 23.780  on 47  degrees of freedom
## AIC: 29.78
```

```
##
## Number of Fisher Scoring iterations: 8
```

```
glm.probs=predict(fit_glm,dattest2, type="response")
glm.pred=rep("N",50)
glm.pred[glm.probs >.5]=" Y"
table(glm.pred ,Diag.test2)
```

```
##          Diag.test2
## glm.pred  N  Y
##        Y  3 29
##       N  15  3
```