ADA 442 HomeWork

Homework 2: Logistic Regression

Furkan

03/12/2021

Contents

1 Introduction

- The ultimate purpose of this research is to find a relation between 'Bl.cromatin' and 'Class' in the data set 'BreastCancer'.
- In Addition, comparison between different Logistic Regression Model over statistical facts has been made.

2 Methodology

- Since data set that chosen contains binary categorical variables , Logistic Regression has been used to create model.

3 Data Set

```
library(mlbench)
set.seed(73745) # for reproducible results
data(BreastCancer)
```

- I have used Breast Cancer data which contains 699 observations from mlbench library.
- Breast Cancer data frame has consisted of 11 variables and 1 target class.
- Since ID is irrelevant and Bare.nuclei has some empty values, have not been used.

4 Explaratory Data analysis

Brief information about the data set

```
head(BreastCancer)
```

```
##        Id Cl.thickness Cell.size Cell.shape Marg.adhesion Epith.c.size
## 1 1000025            5         1          1             1            2
## 2 1002945            5         4          4             5            7
## 3 1015425            3         1          1             1            2
## 4 1016277            6         8          8             1            3
## 5 1017023            4         1          1             3            2
## 6 1017122            8        10         10             8            7
##   Bare.nuclei Bl.cromatin Normal.nucleoli Mitoses     Class
## 1           1           3               1       1    benign
## 2          10           3               2       1    benign
## 3           2           3               1       1    benign
## 4           4           3               7       1    benign
## 5           1           3               1       1    benign
## 6          10           9               7       1 malignant
```

```
summary(BreastCancer)
```

```
##       Id             Cl.thickness   Cell.size     Cell.shape  Marg.adhesion
##  Length:699         1      :145   1      :384   1      :353   1      :407
##  Class :character   5      :130   10     : 67   2      : 59   2      : 58
##  Mode  :character   3      :108   3      : 52   10     : 58   3      : 58
##                     4      : 80   2      : 45   3      : 56   10     : 55
##                     10     : 69   4      : 40   4      : 44   4      : 33
##                     2      : 50   5      : 30   5      : 34   8      : 25
##                     (Other):117   (Other): 81   (Other): 95   (Other): 63
##   Epith.c.size  Bare.nuclei   Bl.cromatin  Normal.nucleoli    Mitoses
##  2      :386   1      :402   2      :166   1      :443     1      :579
##  3      : 72   10     :132   3      :165   10     : 61     2      : 35
##  4      : 48   2      : 30   1      :152   3      : 44     3      : 33
##  1      : 47   5      : 30   7      : 73   2      : 36     10     : 14
##  6      : 41   3      : 28   4      : 40   8      : 24     4      : 12
##  5      : 39   (Other): 61   5      : 34   6      : 22     7      :  9
```

```
##  (Other): 66   NA's   : 16   (Other): 69   (Other): 69     (Other): 17
##        Class
##  benign   :458
##  malignant:241
##
##
##
##
##
```

```
sum(is.na(BreastCancer$Bl.cromatin))
```

```
## [1] 0
```

- As seen above, there is no missing values under cell thickness variable.

5 Model Fit

- Since data is numeric, there is no extra effort has needed.

```
str(BreastCancer$Class)
```

```
##  Factor w/ 2 levels "benign","malignant": 1 1 1 1 1 2 1 1 1 1 ...
```

- Data set has been divided into two part as 80% and 20%.

```
sample.size <- floor(0.80 * nrow(BreastCancer)) # %80 for training, %20 for testing
train.index <- sample(seq_len(nrow(BreastCancer)), size = sample.size)
train <- BreastCancer[train.index, ]
test <- BreastCancer[-train.index, ]
```

##Logistic Regression Model Fitting

```
lm_BlCromatin <- glm(Class ~ Bl.cromatin, data = train, family = "binomial")
summary(lm_BlCromatin)
```

```
##
## Call:
## glm(formula = Class ~ Bl.cromatin, family = "binomial", data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2293  -0.3276  -0.1782   0.4172   2.8813
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -4.1352     0.7127  -5.802 6.56e-09 ***
## Bl.cromatin2     1.2369     0.8116   1.524 0.127516
## Bl.cromatin3     2.7489     0.7457   3.686 0.000228 ***
```

```
## Bl.cromatin4     5.7038     0.8658   6.588 4.47e-11 ***
## Bl.cromatin5     5.9677     0.8933   6.681 2.38e-11 ***
## Bl.cromatin6     5.9269     1.2941   4.580 4.65e-06 ***
## Bl.cromatin7     6.5331     0.8522   7.666 1.77e-14 ***
## Bl.cromatin8    22.7012  1390.6314   0.016 0.986976
## Bl.cromatin9    22.7012  2306.1011   0.010 0.992146
## Bl.cromatin10   22.7012  1809.0546   0.013 0.989988
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 713.91  on 558  degrees of freedom
## Residual deviance: 295.73  on 549  degrees of freedom
## AIC: 315.73
##
## Number of Fisher Scoring iterations: 17
```

- Smallest residual small error shows how well the model fits the data. Also, median should not be far from zero, and the minimum and maximum should be roughly equal in absolute value.
- As seen above, Bl.cromatin3, Bl.cromatin4, Bl.cromatin5, Bl.cromatin6 and Bl.cromatin7 values are significant.

```
predicted <- ifelse(predict(lm_BlCromatin, newdata = test, type = "response") > 0.5, "malignant", "beni
predicted <- as.factor(predicted)
```

5.1 Evaluate Model Performance

```
# Gather x variable, y variable and predicted y variable in the same data frame
compareDataFrame <- data.frame(Bl.cromatin = test$Bl.cromatin, Class = test$Class, PredictedClass = pre

library(caret)
```

```
## Loading required package: ggplot2

## Loading required package: lattice
```

```
confusionMatrix(compareDataFrame$Class, compareDataFrame$PredictedClass)
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction  benign malignant
##   benign        82         5
##   malignant     10        43
##
##                Accuracy : 0.8929
##                  95% CI : (0.8294, 0.9388)
##     No Information Rate : 0.6571
```

```
##     P-Value [Acc > NIR] : 1.207e-10
##
##                   Kappa : 0.768
##
##  Mcnemar's Test P-Value : 0.3017
##
##             Sensitivity : 0.8913
##             Specificity : 0.8958
##          Pos Pred Value : 0.9425
##          Neg Pred Value : 0.8113
##              Prevalence : 0.6571
##          Detection Rate : 0.5857
##    Detection Prevalence : 0.6214
##       Balanced Accuracy : 0.8936
##
##        'Positive' Class : benign
##
```

- What we can obtain from Accuracy is the ratio of all correct predictions out of Total Predictions. We have fairly high rate which is good.
- What we can obtain from Sensitivity is the ratio of True Positive out of Actual positive. We have fairly high rate which is good.
- What we can obtain from Specificity is the ratio of True Negative out of Actual Negative. We have fairly high rate which is good.
- What we can obtain from Precision is the ratio of True Positive out of Predicted Positive. We have fairly high rate which is good.

5.2 *Multiple Logistic Regression Model Fitting*

- For the logistic regression case, 'Bl.cromatin' and 'Cl.thickness' has chosen as variable for x.

```
ml_ClThicknessBlCromatin <- glm(Class ~ Bl.cromatin + Cl.thickness, data = train, family = binomial)
```

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

```
summary(ml_ClThicknessBlCromatin)
```

```
##
## Call:
## glm(formula = Class ~ Bl.cromatin + Cl.thickness, family = binomial,
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.3208  -0.1594  -0.0975   0.0000   2.9736
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)     6.988e-03  4.385e+02   0.000 0.999987
```

```
## Bl.cromatin2    9.374e-01  1.230e+00   0.762 0.446105
## Bl.cromatin3    2.769e+00  1.099e+00   2.519 0.011769 *
## Bl.cromatin4    5.458e+00  1.221e+00   4.471 7.78e-06 ***
## Bl.cromatin5    6.473e+00  1.375e+00   4.707 2.51e-06 ***
## Bl.cromatin6    5.516e+00  1.659e+00   3.324 0.000886 ***
## Bl.cromatin7    6.565e+00  1.208e+00   5.435 5.47e-08 ***
## Bl.cromatin8    2.395e+01  3.112e+03   0.008 0.993858
## Bl.cromatin9    2.446e+01  5.004e+03   0.005 0.996100
## Bl.cromatin10   2.393e+01  3.969e+03   0.006 0.995188
## Cl.thickness.L  2.264e+01  1.782e+03   0.013 0.989867
## Cl.thickness.Q  1.236e+01  1.178e+03   0.010 0.991626
## Cl.thickness.C  4.593e+00  1.023e+03   0.004 0.996416
## Cl.thickness^4 -3.147e+00  1.752e+03  -0.002 0.998566
## Cl.thickness^5 -4.593e+00  2.038e+03  -0.002 0.998201
## Cl.thickness^6 -5.273e+00  1.721e+03  -0.003 0.997556
## Cl.thickness^7 -3.638e+00  1.102e+03  -0.003 0.997366
## Cl.thickness^8 -2.820e+00  5.232e+02  -0.005 0.995700
## Cl.thickness^9 -2.315e+00  1.630e+02  -0.014 0.988672
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 713.91  on 558  degrees of freedom
## Residual deviance: 157.18  on 540  degrees of freedom
## AIC: 195.18
##
## Number of Fisher Scoring iterations: 19
```

```
predictedClass2 <- ifelse(predict(ml_ClThicknessBlCromatin, newdata = test, type = "response") > 0.5, "
predictedClass2 <- as.factor(predictedClass2)
```

```
compareDataFrame2 <- data.frame(Bl.cromatin = test$Bl.cromatin, Cl.thickness = test$Cl.thickness, Class

confusionMatrix(compareDataFrame2$Class, compareDataFrame2$PredictedClass2)
```

```
## Confusion Matrix and Statistics
##
##            Reference
## Prediction  benign malignant
##   benign        81         6
##   malignant      4        49
##
##                Accuracy : 0.9286
##                  95% CI : (0.8726, 0.9652)
##     No Information Rate : 0.6071
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.8493
##
##  Mcnemar's Test P-Value : 0.7518
##
##             Sensitivity : 0.9529
```

```
##             Specificity : 0.8909
##          Pos Pred Value : 0.9310
##          Neg Pred Value : 0.9245
##              Prevalence : 0.6071
##          Detection Rate : 0.5786
##    Detection Prevalence : 0.6214
##       Balanced Accuracy : 0.9219
##
##        'Positive' Class : benign
##
```

5.3 *Evaluate Multiple Logistic Regression Model Performance*

-We can use F1-score to determine which model has better performance. *F1-Score = (2 * precision * recall) / (precision + recall)*

- *For Logistic regression:*

```
F1LR <- (2 * 0.9375  * 0.9474) / (0.9375 + 0.9474)
F1LR
```

```
## [1] 0.942424
```

- *For Multiple Logistic regression:*

```
F1MLR <- (2 * 0.9583 * 0.9485) / (0.9583 + 0.9485)
F1MLR
```

```
## [1] 0.9533748
```

6 Conclusions

To conclude that, we can clearly see that if we have a consisted data, applying Multiple Regression is better than applying Logistic regression solely when we evaluate F-scores.

7 References

- Week 3 Lecture Slides
- https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62
- http://www.sthda.com/english/articles/36-classification-methods-essentials/151-logistic-regression-essentials-in-r/