STAT 46700/ CS 5900 Topics in Data Science Spring 2025

Lab 6

[Vaishak Balachandra]

Q.N. 1) Suzuki et al. (2006) measured sand grain size on 2828 beaches in Japan and observed the presence or absence of the burrowing wolf spider *Lycosa ishikariana* on each beach. Sand grain size is a measurement variable, and spider presence or absence is a nominal variable. Spider presence or absence is the dependent variable.

a) Fit a simple logistic regression for the subject data
b) Create a confusion matrix and find the accuracy rate of the classification.

```
> ################################################################################
> # Q1
> Q1 <- read.csv("Spider.csv")
> head(Q1)
  Grain.size Spiders
1      0.245  absent
2      0.247  absent
3      0.285 present
4      0.299 present
5      0.327 present
6      0.347 present
> dim(Q1)
[1] 28  2
> names(Q1)
[1] "Grain.size" "Spiders"
> attach(Q1)
>
> Q1$Spiders = as.numeric(Q1$Spiders == "present")
> head(Q1)
  Grain.size Spiders
1      0.245       0
2      0.247       0
3      0.285       1
4      0.299       1
5      0.327       1
6      0.347       1
>
>
> # a
> model1 <- glm(Q1$Spiders~Q1$Grain.size, family = "binomial")
> summary(model1)

Call:
glm(formula = Q1$Spiders ~ Q1$Grain.size, family = "binomial")

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)     -1.648      1.354  -1.217   0.2237
Q1$Grain.size    5.122      3.006   1.704   0.0884 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 35.165  on 27  degrees of freedom
Residual deviance: 30.632  on 26  degrees of freedom
```

```
AIC: 34.632

Number of Fisher Scoring iterations: 5

> cat("Logistic Fitted Model is:
+ Spiders = [1 + exp(1.648 - 5.122*Grain.Size)]^(-1)")
Logistic Fitted Model is:
Spiders = [1 + exp(1.648 - 5.122*Grain.Size)]^(-1)
> 
> table(Q1$Spiders)
 0  1 
 9 19 
> p = predict(model1, data = Q1, type = "response")
> p
        1         2         3         4         5         6         7         8         9        10 
       11 
0.4030327 0.4054996 0.4531423 0.4709625 0.5067803 0.5323432 0.5437994 0.5488769 0.5526784 0.5539443 
0.5964640 
       12        13        14        15        16        17        18        19        20        21 
       22 
0.5989270 0.6099472 0.6244662 0.6375823 0.6845733 0.7229707 0.7430108 0.7730416 0.7801496 0.8013220 
0.8347849 
       23        24        25        26        27        28 
0.8471102 0.9263250 0.9382579 0.9591598 0.9748715 0.9759763 
> pp = ifelse(p>0.4, 1, 0)
> pp
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 
 1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1 
> 
> 
> # b
> install.packages("caret")
package ‘caret’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
	C:\Users\PNW_checkout\AppData\Local\Temp\Rtmp0mjusd\downloaded_packages
> library(caret)
> confusionMatrix(data = factor(pp), reference = factor(Q1$Spiders), positive = "1")
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0  0  0
         1  9 19
                                          
               Accuracy : 0.6786          
                 95% CI : (0.4765, 0.8412)
    No Information Rate : 0.6786          
    P-Value [Acc > NIR] : 0.589064        
                                          
                  Kappa : 0               
                                          
 Mcnemar's Test P-Value : 0.007661        
                                          
            Sensitivity : 1.0000          
            Specificity : 0.0000          
         Pos Pred Value : 0.6786          
         Neg Pred Value :    NaN          
             Prevalence : 0.6786          
         Detection Rate : 0.6786          
   Detection Prevalence : 1.0000          
      Balanced Accuracy : 0.5000          
                                          
       'Positive' Class : 1               
                                          
> cat("Accuracy of the classification: 67.86%")
Accuracy of the classification: 67.86%
```

Q.N. 2) A real estate agent used information on 1115 houses. She wants to predict whether a house sold in the first 3 months it was on the market based on other variables. The variables available include:

Sold : 1 = Yes—the house sold within the first 3 months it was listed;
0 = No, it did not sell within 3 months.
Price: The price of the house as sold in 2002.
Living Area: The size of the living area of the house in square feet
Bedrooms : The number of bedrooms
Bathrooms : The number of bathrooms (a half bath is a toilet and sink only)
Age: Age of the house in years
Fireplaces: Number of fireplaces in the house

a) Fit a multiple logistic regression model to predict whether a house will sell within the first 3 months it's on the market based on Price ($), Living Area (sq ft), Bedrooms (#), Bathrooms (#), Fireplaces (#), and Age (years).
b) Create the confusion matrix and find the accuracy rate of the classification

```
> ################################################################################
> # Q2
> Q2 <- read.csv("home.csv")
> head(Q2)
  Living.Area Age  Price Bedrooms Bathrooms Fireplaces Sold
1        1680  31 196809        3       1.5          0    1
2        1442  27 200000        3       1.5          2    0
3        1785   1 199039        3       2.5          1    1
4        1480  19 165500        3       1.5          0    1
5        1845   0 214997        2       2.5          1    0
6        2822   1 365000        4       2.5          1    1
> dim(Q2)
[1] 1115    7
> names(Q2)
[1] "Living.Area" "Age"         "Price"       "Bedrooms"    "Bathrooms"   "Fireplaces"  "Sold"
> attach(Q2)
>
>
> # a
> model2 = glm(Sold~.,family= binomial , data = Q2)
> summary(model2)

Call:
glm(formula = Sold ~ ., family = binomial, data = Q2)

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.222e+00  3.826e-01  -8.422  < 2e-16 ***
Living.Area -1.444e-03  2.518e-04  -5.734  9.8e-09 ***
Age          4.900e-03  2.823e-03   1.736 0.082609 .
Price        1.693e-05  1.444e-06  11.719  < 2e-16 ***
Bedrooms     4.805e-01  1.366e-01   3.517 0.000436 ***
Bathrooms   -1.813e-01  1.829e-01  -0.991 0.321493
Fireplaces  -1.253e-01  1.633e-01  -0.767 0.442885
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1401.2  on 1114  degrees of freedom
Residual deviance: 1159.9  on 1108  degrees of freedom
```

```
AIC: 1173.9

Number of Fisher Scoring iterations: 4

> model2

Call:  glm(formula = Sold ~ ., family = binomial, data = Q2)

Coefficients:
(Intercept)  Living.Area          Age        Price     Bedrooms    Bathrooms   Fireplaces
 -3.222e+00   -1.444e-03    4.900e-03    1.693e-05    4.805e-01   -1.813e-01   -1.253e-01

Degrees of Freedom: 1114 Total (i.e. Null);  1108 Residual
Null Deviance:          1401
Residual Deviance: 1160      AIC: 1174
>
> cat("Multiple Logistic Fitted Model is:
+ Spiders = [1 + exp(3.222 + 0.001444*Living.Area - 0.0049*Age - 0.00001693*Price - 0.4805*Bedrooms
+ 0.01813*Bathrooms + 0.1253*Fireplaces)]^(-1)")
Multiple Logistic Fitted Model is:
Spiders = [1 + exp(3.222 + 0.001444*Living.Area - 0.0049*Age - 0.00001693*Price - 0.4805*Bedrooms +
0.01813*Bathrooms + 0.1253*Fireplaces)]^(-1)
>
>
> # b
> p1 = predict(model2, type = 'response')
> pp1 <- ifelse(p1 > 0.5, 1, 0)
> library(caret)
> confusionMatrix(data = factor(pp1), reference = factor(Q2$Sold), positive = "1")
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 687 225
         1  69 134

               Accuracy : 0.7363
                 95% CI : (0.7094, 0.762)
    No Information Rate : 0.678
    P-Value [Acc > NIR] : 1.314e-05

                  Kappa : 0.3183

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.3733
            Specificity : 0.9087
         Pos Pred Value : 0.6601
         Neg Pred Value : 0.7533
             Prevalence : 0.3220
         Detection Rate : 0.1202
   Detection Prevalence : 0.1821
      Balanced Accuracy : 0.6410

       'Positive' Class : 1

> cat("Accuracy of the model: 73.63%")
Accuracy of the model: 73.63%
```