Thoracic Surgery Data Week 10

Gillian Tatreau

2022-11-05

i.

```
##
## Call:
## glm(formula = Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6 + PRE7 +
##     PRE8 + PRE9 + PRE10 + PRE11 + PRE14 + PRE17 + PRE19 + PRE25 +
##     PRE30, family = binomial(), data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6381  -0.4663  -0.3781  -0.2602   2.4216
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.670e+01  2.400e+03  -0.007   0.9944
## AGE         -6.649e-03  2.331e-02  -0.285   0.7754
## DGNDGN2      1.463e+01  2.400e+03   0.006   0.9951
## DGNDGN3      1.394e+01  2.400e+03   0.006   0.9954
## DGNDGN4      1.449e+01  2.400e+03   0.006   0.9952
## DGNDGN5      1.647e+01  2.400e+03   0.007   0.9945
## DGNDGN6      1.752e-01  2.666e+03   0.000   0.9999
## DGNDGN8      1.211e+00  3.393e+03   0.000   0.9997
## PRE4        -1.644e-01  2.254e-01  -0.729   0.4658
## PRE5        -2.399e-02  1.838e-02  -1.305   0.1918
## PRE6PRZ1    -3.973e-01  6.192e-01  -0.642   0.5211
## PRE6PRZ2     2.651e-01  9.163e-01   0.289   0.7724
## PRE7T        1.186e+00  6.275e-01   1.890   0.0588 .
## PRE8T        1.567e-01  4.788e-01   0.327   0.7434
## PRE9T        1.259e+00  6.166e-01   2.042   0.0412 *
## PRE10T       3.862e-01  5.616e-01   0.688   0.4917
## PRE11T       4.140e-01  4.960e-01   0.835   0.4039
## PRE14OC12    2.013e-01  4.099e-01   0.491   0.6233
## PRE14OC13    1.448e+00  6.923e-01   2.092   0.0365 *
## PRE14OC14    1.436e+00  7.195e-01   1.996   0.0459 *
## PRE17T       1.125e+00  5.433e-01   2.071   0.0383 *
## PRE19T      -1.423e+01  1.678e+03  -0.008   0.9932
## PRE25T      -8.600e-01  1.423e+00  -0.604   0.5455
## PRE30T       1.036e+00  6.014e-01   1.722   0.0850 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 282.44  on 358  degrees of freedom
## Residual deviance: 235.58  on 335  degrees of freedom
## AIC: 283.58
##
## Number of Fisher Scoring iterations: 15
```

ii.

According to the summary, the variables that had the greatest affect on survival rate (those with a p-value of less than 0.25) were PRE14, PRE9, PRE17, PRE30, PRE4, PRE5, PRE6, and AGE. Therefore, the model that would be the most accurate would include just those variables in the order from most significant to least significant p-values.

```
##
## Call:
## glm(formula = Risk1Yr ~ PRE14 + PRE9 + PRE17 + PRE30 + PRE4 +
##     PRE5 + PRE6 + AGE, family = binomial(), data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3651  -0.4839  -0.4479  -0.3071   2.4637
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.9170690  1.7799238  -1.639  0.10124
## PRE14OC12    0.1659823  0.3789319   0.438  0.66137
## PRE14OC13    1.6975992  0.6323254   2.685  0.00726 **
## PRE14OC14    1.4514988  0.6705051   2.165  0.03040 *
## PRE9T        0.8610332  0.5772843   1.492  0.13582
## PRE17T       1.1870907  0.5128368   2.315  0.02063 *
## PRE30T       0.9341346  0.5773826   1.618  0.10569
## PRE4        -0.1128181  0.2091913  -0.539  0.58968
## PRE5        -0.0123289  0.0179174  -0.688  0.49139
## PRE6PRZ1     0.0724134  0.4302748   0.168  0.86635
## PRE6PRZ2     1.0728396  0.6566144   1.634  0.10228
## AGE          0.0003303  0.0219707   0.015  0.98801
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 282.44  on 358  degrees of freedom
## Residual deviance: 254.80  on 347  degrees of freedom
## AIC: 278.8
##
## Number of Fisher Scoring iterations: 5
```

iii.

The model was able to predict the correct value for the training data with approximately an 85% accuracy. The model was able to predict the correct value for the testing data with approximately an 82% accuracy.

```
# accuracy for training data
res <- predict(model17, train, type = "response")
res
```

```
##          1          2          3          4          5          7         10
## 0.31200750 0.09913556 0.09725925 0.03383355 0.23629607 0.25534870 0.10146487
##         11         12         13         14         15         17         18
## 0.04527424 0.09423544 0.09486656 0.52026032 0.10683634 0.11195141 0.10420658
##         19         20         21         22         24         27         28
## 0.11553011 0.08989032 0.08719013 0.09920908 0.08269891 0.09160390 0.09035003
##         29         30         31         32         34         35         36
## 0.09265044 0.07943968 0.30238303 0.03896621 0.15569914 0.04077509 0.09537070
##         37         38         39         41         44         45         46
## 0.11989216 0.25814081 0.07754025 0.08657803 0.10634830 0.10345659 0.09376140
##         47         48         49         51         52         53         54
## 0.09689852 0.11458555 0.22028553 0.04921705 0.08513721 0.43715768 0.10201523
##         55         56         58         61         62         63         64
## 0.10184657 0.10695692 0.24593829 0.33320618 0.25222186 0.07120008 0.10279855
##         65         66         68         69         70         71         72
## 0.10769972 0.04662780 0.10215747 0.11320845 0.10996591 0.03720880 0.27444282
##         73         75         78         79         80         81         82
## 0.08353597 0.07931019 0.10799265 0.11997616 0.04641936 0.10192108 0.33595550
##         83         85         86         87         88         89         90
## 0.10817885 0.08920286 0.10127733 0.07746174 0.24007217 0.22987039 0.11955513
##         92         95         96         97         98         99        100
## 0.09935661 0.27984555 0.07932751 0.11421132 0.22139624 0.10357666 0.19471770
##        102        103        104        105        106        107        109
## 0.31832697 0.09562968 0.04267581 0.04028213 0.03121714 0.11590382 0.03594152
##        112        113        114        115        116        117        119
## 0.26471233 0.04730902 0.07332835 0.08717199 0.26185318 0.11175292 0.08981770
##        120        121        122        123        124        126        129
## 0.11626279 0.03591753 0.10379451 0.58236937 0.08590708 0.11259270 0.43673820
##        130        131        132        133        134        136        137
## 0.07013320 0.09115178 0.12126805 0.36313210 0.09925631 0.09558687 0.34135659
##        138        139        140        141        143        146        147
## 0.33264661 0.11643498 0.04230406 0.10408551 0.03031245 0.09903671 0.03551105
##        148        149        150        151        153        154        155
## 0.11119122 0.10295795 0.09150095 0.04506516 0.04808129 0.10475633 0.10774195
##        156        157        158        160        163        164        165
## 0.14286285 0.56748446 0.11346514 0.10335202 0.10798059 0.04976065 0.30833638
##        166        167        168        170        171        172        173
## 0.22201987 0.09114274 0.10020807 0.36224210 0.09959773 0.27659349 0.22251080
##        174        175        177        180        181        182        183
## 0.09843127 0.11251326 0.54091510 0.25127103 0.10242595 0.08718755 0.09321615
##        184        185        187        188        189        190        191
## 0.11072492 0.03887111 0.09191740 0.09276727 0.10346978 0.08332055 0.10819003
##        192        194        197        198        199        200        201
## 0.09153307 0.09329143 0.11203725 0.04619439 0.08969263 0.10135718 0.11736672
##        202        204        205        206        207        208        209
```

```
## 0.09066214 0.09509705 0.04255790 0.11202473 0.07970411 0.09923803 0.09049444
##        211        214        215        216        217        218        219
## 0.07210399 0.29395010 0.08877913 0.10621289 0.10783014 0.09331978 0.08499671
##        221        222        223        224        225        226        228
## 0.36729080 0.09568401 0.19011890 0.08171362 0.08854777 0.29421959 0.11388555
##        231        232        233        234        235        236        238
## 0.16652402 0.11085603 0.10240009 0.11472442 0.10086754 0.09636690 0.08641441
##        239        240        241        242        243        245        248
## 0.09099218 0.10405968 0.07604091 0.08393487 0.24980984 0.08900259 0.09124429
##        249        250        251        252        253        255        256
## 0.09246676 0.11594195 0.10105522 0.22167947 0.10066896 0.08935513 0.03964043
##        257        258        259        260        262        265        266
## 0.09699285 0.08922537 0.08770753 0.09619972 0.12541177 0.09917900 0.10365090
##        267        268        269        270        272        273        274
## 0.10115288 0.59814862 0.45905581 0.08238535 0.08913542 0.04838765 0.47207264
##        275        276        277        279        282        283        284
## 0.09830645 0.12657058 0.11002101 0.03234516 0.03900209 0.04545800 0.11859008
##        285        286        287        289        290        291        292
## 0.09544818 0.10124580 0.08883014 0.31081239 0.10467632 0.09736609 0.26341535
##        293        294        296        299        300        301        302
## 0.09451577 0.09784470 0.10210305 0.19693367 0.09929812 0.09628269 0.04208183
##        303        304        306        307        308        309        310
## 0.33598910 0.09672604 0.11354938 0.10685208 0.11202661 0.10190130 0.10642174
##        311        313        316        317        318        319        320
## 0.03503170 0.10831085 0.10492599 0.04316753 0.23800139 0.10116835 0.04516123
##        321        323        324        325        326        327        328
## 0.25544459 0.09749763 0.26219451 0.07337419 0.02885392 0.10795011 0.23237538
##        330        333        334        335        336        337        338
## 0.04819980 0.09275548 0.04054280 0.09866496 0.10134786 0.10283882 0.10723184
##        340        341        342        343        344        345        347
## 0.11069611 0.08057657 0.11836073 0.10116074 0.09494250 0.10224516 0.08956282
##        350        351        352        353        354        355        357
## 0.01801674 0.11450204 0.09556337 0.04594753 0.16492524 0.08188836 0.22946895
##        358        359        360        361        362        364        367
## 0.11545072 0.11715579 0.04408758 0.11447376 0.09583975 0.11994226 0.09921474
##        368        369        370        371        372        374        375
## 0.30609818 0.10061821 0.09405214 0.10442369 0.04669237 0.60613561 0.11720776
##        376        377        378        379        381        384        385
## 0.08672433 0.08441744 0.11274452 0.09320689 0.09653765 0.04224559 0.05193744
##        386        387        388        389        391        392        393
## 0.30108582 0.17222204 0.09008786 0.39448100 0.11132231 0.12454636 0.34883844
##        394        395        396        398        401        402        403
## 0.10146421 0.08526815 0.25149995 0.09261835 0.03930007 0.04052570 0.11025903
##        404        405        406        408        409        410        411
## 0.09852524 0.10342982 0.03517123 0.10646861 0.28637474 0.09213895 0.11617439
##        412        413        415        418        419        420        421
## 0.31665452 0.03542749 0.11253509 0.08609524 0.04537126 0.26478378 0.07579452
##        422        423        425        426        427        428        429
## 0.34397804 0.10689912 0.11195953 0.10703093 0.37741881 0.03760918 0.10888417
##        430        432        435        436        437        438        439
## 0.58131510 0.11218478 0.09386902 0.09525318 0.19028981 0.09699426 0.01643382
##        440        442        443        444        445        446        447
## 0.10168311 0.03638983 0.09233883 0.08920429 0.04715258 0.09189699 0.03317873
##        449        452        453        454        455        456        457
```

```
## 0.10379795 0.10937762 0.36974565 0.09658611 0.08632373 0.11159832 0.11814538
##        459        460        461        462        463        464        466
## 0.03652345 0.03728711 0.03798955 0.05306514 0.24130478 0.27821250 0.34164226
##        469        470
## 0.12343664 0.08498240
```

```r
confmatrix <- table(Actual_Value = train$Risk1Yr, Predicted_Value = res > 0.5)
confmatrix
```

```
##             Predicted_Value
## Actual_Value FALSE TRUE
##            F   305    6
##            T    47    1
```

```r
(confmatrix[[1,1]] + confmatrix[[2,2]]) / sum(confmatrix)
```

```
## [1] 0.8523677
```

```r
# accuracy for testing data
res <- predict(model17, test, type = "response")
res
```

```
##          6          8          9         16         23         25         26
## 0.04185542 0.09284068 0.21811456 0.09084696 0.11071714 0.03153122 0.03878658
##         33         40         42         43         50         57         59
## 0.14533959 0.07902225 0.10713657 0.10737265 0.03571900 0.08116728 0.08114655
##         60         67         74         76         77         84         91
## 0.09305596 0.03935069 0.02458991 0.29659143 0.10792753 0.12061853 0.10444631
##         93         94        101        108        110        111        118
## 0.10326689 0.06220992 0.09122850 0.10294680 0.22206201 0.08659067 0.29173429
##        125        127        128        135        142        144        145
## 0.10592810 0.08035493 0.28144941 0.08988940 0.09672035 0.11399966 0.20945607
##        152        159        161        162        169        176        178
## 0.08615843 0.10905084 0.04399905 0.08960643 0.25726814 0.20153399 0.10942567
##        179        186        193        195        196        203        210
## 0.09369599 0.09178929 0.03892086 0.08352280 0.11320658 0.33471293 0.11278043
##        212        213        220        227        229        230        237
## 0.10323199 0.20887106 0.09057088 0.11035763 0.03966121 0.29745169 0.10877588
##        244        246        247        254        261        263        264
## 0.08052663 0.08455330 0.07762795 0.10013428 0.14487058 0.08959469 0.03652948
##        271        278        280        281        288        295        297
## 0.11643881 0.27963469 0.08416252 0.10604563 0.11058348 0.26422320 0.11620737
##        298        305        312        314        315        322        329
## 0.19821282 0.08066362 0.04977540 0.11532870 0.12163590 0.09192842 0.12298937
##        331        332        339        346        348        349        356
## 0.04452421 0.08337262 0.07936249 0.54463041 0.33066724 0.08197131 0.10147015
##        363        365        366        373        380        382        383
## 0.28761904 0.23603505 0.11423907 0.09497530 0.09816652 0.07056676 0.04451364
##        390        397        399        400        407        414        416
## 0.28806618 0.04389059 0.09540061 0.09822669 0.08979677 0.11607315 0.03525326
##        417        424        431        433        434        441        448
## 0.24681334 0.03405375 0.09399607 0.12059037 0.08278331 0.10991536 0.09366994
##        450        451        458        465        467        468
## 0.11198591 0.10941467 0.07469647 0.36333041 0.08125261 0.18265056
```

```
confmatrix <- table(Actual_Value = test$Risk1Yr, Predicted_Value = res > 0.5)
confmatrix
```

```
##              Predicted_Value
## Actual_Value FALSE TRUE
##            F    88    1
##            T    22    0
```

```
(confmatrix[[1,1]] + confmatrix[[2,2]]) / sum(confmatrix)
```

```
## [1] 0.7927928
```

Code Appendix

```
knitr::opts_chunk$set(echo = TRUE)
library(foreign)
library(caTools)

setwd("/Users/gillian/Documents/Bellevue Grad Program/Fall 2022/DSC520/DSC520 Repo")

file_name <- "/Users/gillian/Documents/Bellevue Grad Program/Fall 2022/DSC520/DSC520 Repo/ThoraricSurge
surgery <- read.arff(file_name)
head(surgery)

split <- sample.split(surgery, SplitRatio = 0.8)
split
train <- subset(surgery, split == "TRUE")
test <- subset(surgery, split == "FALSE")

colSums(is.na(surgery))

model1 <- glm(Risk1Yr ~ AGE, data = train, family = binomial())
model2 <- glm(Risk1Yr ~ AGE + DGN, data = train, family = binomial())
model3 <- glm(Risk1Yr ~ AGE + DGN + PRE4, data = train, family = binomial())
model4 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5, data = train, family = binomial())
model5 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6, data = train, family = binomial())
model6 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6 + PRE7, data = train, family = binomial())
model7 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8, data = train, family = binomial()
model8 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 + PRE9, data = train, family = bin
model9 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 + PRE9 + PRE10, data = train, famil
model10 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 + PRE9 + PRE10 + PRE11, data = tr
model11 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 + PRE9 + PRE10 + PRE11 + PRE14, d
model12 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 + PRE9 + PRE10 + PRE11 + PRE14 + 
model13 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 + PRE9 + PRE10 + PRE11 + PRE14 + 
model14 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 + PRE9 + PRE10 + PRE11 + PRE14 + 
model15 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 + PRE9 + PRE10 + PRE11 + PRE14 + 
model16 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 + PRE9 + PRE10 + PRE11 + PRE14 + 

summary(model1)
summary(model2)
```

```
summary(model3)
summary(model4)
summary(model5)
summary(model6)
summary(model7)
summary(model8)
summary(model9)
summary(model10)
summary(model11)
summary(model12)
summary(model13)
summary(model14)
summary(model15)
summary(model16)

model17 <- glm(Risk1Yr ~ PRE14 + PRE9 + PRE17 + PRE30 + PRE4 + PRE5 + PRE6 + AGE, data = train, family =

# compare model 1 and model 15
modelChi1 <- model1$deviance - model15$deviance
chidf1 <- model1$df.residual - model15$df.residual
chisq.prob1 <- 1 - pchisq(modelChi1, chidf1)
modelChi1; chidf1; chisq.prob1

# compare model 1 and model 17
modelChi2 <- model1$deviance - model17$deviance
chidf2 <- model1$df.residual - model17$df.residual
chisq.prob2 <- 1 - pchisq(modelChi2, chidf2)
modelChi2; chidf2; chisq.prob2

# compare model 17 and model 15
modelChi3 <- model17$deviance - model15$deviance
chidf3 <- model17$df.residual - model15$df.residual
chisq.prob3 <- 1 - pchisq(modelChi3, chidf3)
modelChi3; chidf3; chisq.prob3
model15 <- glm(Risk1Yr ~ AGE + DGN + PRE4 + PRE5 + PRE6 + PRE7 + PRE8 + PRE9 + PRE10 + PRE11 + PRE14 + P
summary(model15)
model17 <- glm(Risk1Yr ~ PRE14 + PRE9 + PRE17 + PRE30 + PRE4 + PRE5 + PRE6 + AGE, data = train, family =
summary(model17)
# accuracy for training data
res <- predict(model17, train, type = "response")
res

confmatrix <- table(Actual_Value = train$Risk1Yr, Predicted_Value = res > 0.5)
confmatrix

(confmatrix[[1,1]] + confmatrix[[2,2]]) / sum(confmatrix)

# accuracy for testing data
res <- predict(model17, test, type = "response")
res
confmatrix <- table(Actual_Value = test$Risk1Yr, Predicted_Value = res > 0.5)
confmatrix
```

```
(confmatrix[[1,1]] + confmatrix[[2,2]]) / sum(confmatrix)
```