hw2_sml

Group2

2023-10-11

Exercise 1

We start by focusing on the ridge expression. In the first part we want to center the response and the regressors:

$$\sum_{i=1}^{N}\left(y_i-\beta_0-\sum_{j=1}^{p}(x_{ij}-\overline{x}_j)\beta_j\right)=$$
$$\sum_{i=1}^{N}\left(y_i-\overline{y}-\beta_0-\sum_{j=1}^{p}\overline{x_j}\beta_j-\sum_{j=1}^{p}(x_{ij}-\overline{x}_j)\beta_j\right)=$$
$$\sum_{i=1}^{N}\left(y_i-\beta_0^c-\sum_{j=1}^{p}(x_{ij}-\overline{x}_j)\beta_j^c\right).$$

Hence, we have that:

$$\beta_0^c=\beta_0+\sum_{j=1}^{p}\overline{x_j}\beta_j-\overline{y}$$
$$\beta_j^c=\beta_j\ \ j=1,2,\ldots,p$$

As a final remark, this centering procedure consist in shifting all the variables x_j and the response y to have mean zero. As a result, only β_0^c which is the intercept, is going to change to be equal to 0, whereas the slope of the regression line β_j^c remain the same.

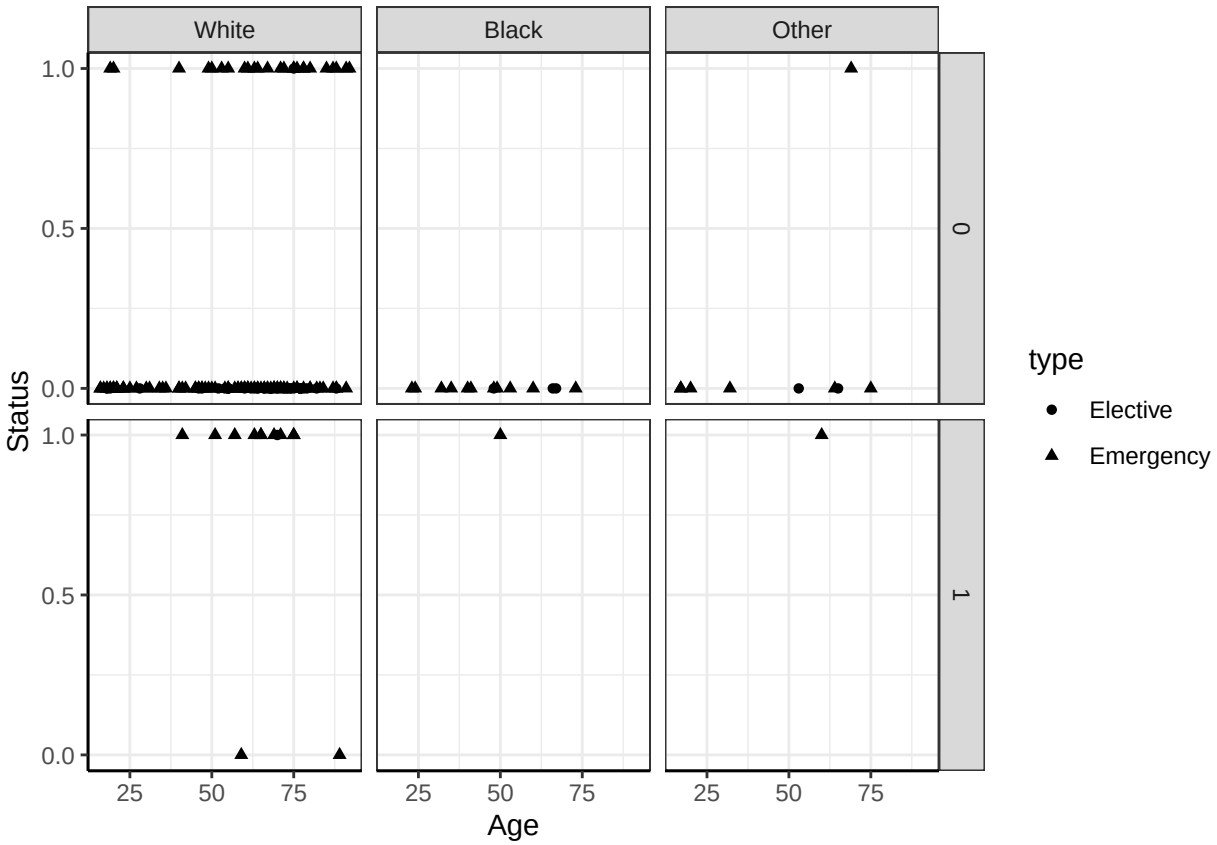
Exercise 9

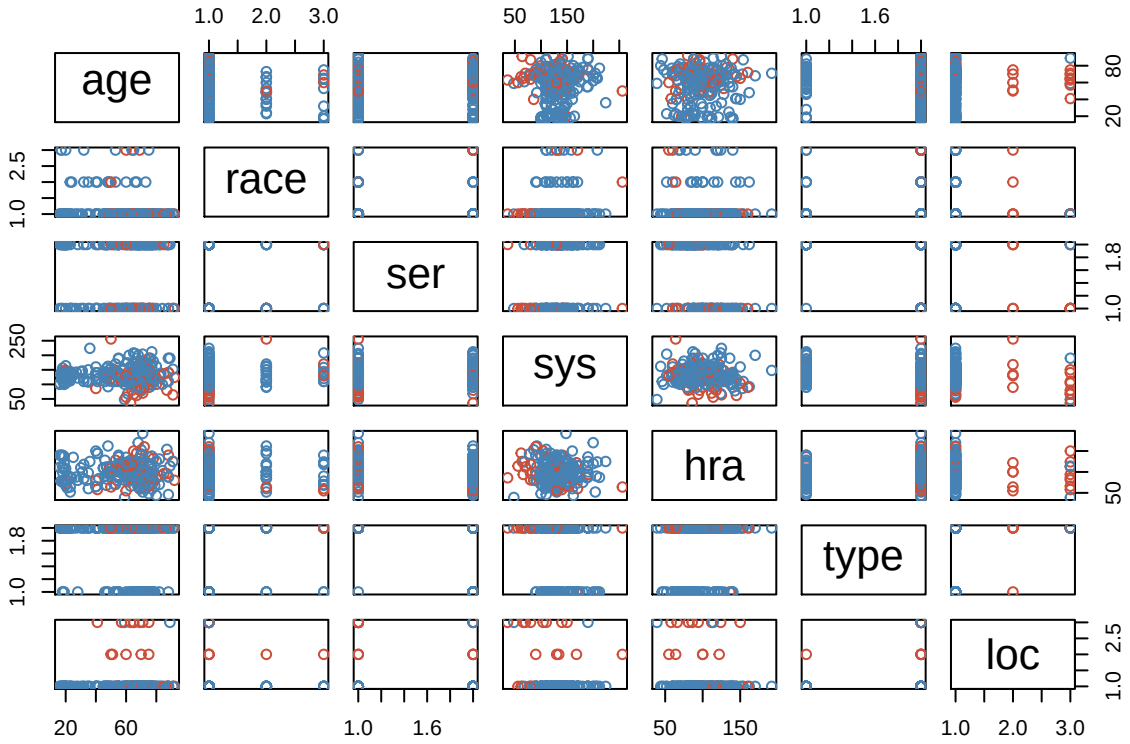
In this exercise, we delve deeper into common problems we might encounter when fitting a logistic regression model for the prediction of categorical data. The problem mentioned here consists of Complete or Quasi-Complete Separation, where the issue is not really connected to the model itself, but it is mainly due to "thin" data. This means, we might have a really skewed distribution in the response variable; in this case we observe only 20% of cases in which the patient has died, in a sample size of 200.

Point A

```r
#####################################################################
## A -  Fit a logistic regression model with all regressors
#####################################################################
## A.1 Short exploratory Data Analysis

# Dataframe
df <- icu

# Variables classes
#unlist(lapply(df, class))

# NAs
#unlist(lapply(df, function(x) sum(is.na(x))))

# Change variables of interest into numeric, binarize variables
df <- within(df,
             {
               # Response variable
               sta <- ifelse(sta == "Died", 1, 0)
               loc_bin <- ifelse(loc == "Nothing", 0, 1)
               race_bin <- ifelse(race == "White", 1, 0)
             })

## A.2 Model Fitting and summary. Note locComa and Stupor are both negative,
## but might have same pred power.


m1 <- glm(sta ~., family = binomial(link = "logit"), data = df[, 2:(length(df)-2)])
coef <- m1$coefficients
```

After running the lositic regression on all the variables we encounter the warning about fitted probabilities being numerically 0 or 1, which is a first sign of **separation**. To graphically illustrate the issue, we can look into the data with a grid plot:


White
Black
Other
1.0
0.5
0.0
Status
0
1
25
50
75
Age
type
Elective
Emergency



Also in the summary we that the coefficients for locStupor, Black race, Emergency and other regressors are huge, indicating the maximum likelihood estimates do not exist. We can also notice the standard errors of these coefficients to be particulary big in magnitude.

```
##
## Call:
## glm(formula = sta ~ ., family = binomial(link = "logit"), data = df[,
##     2:(length(df) - 2)])
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.50525  -0.53717  -0.17867  -0.00019   3.01708
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -5.548e+00  2.271e+00  -2.444  0.01454 *
## age            5.645e-02  1.848e-02   3.055  0.00225 **
## genderFemale  -7.215e-01  5.460e-01  -1.321  0.18639
## raceBlack     -1.617e+01  1.314e+03  -0.012  0.99018
## raceOther      5.829e-01  1.313e+00   0.444  0.65696
## serSurgical   -6.739e-01  6.289e-01  -1.071  0.28398
## canYes         3.483e+00  1.121e+00   3.106  0.00189 **
## crnYes         1.191e-01  8.449e-01   0.141  0.88786
## infYes        -1.081e-01  5.557e-01  -0.195  0.84573
## cprYes         1.032e+00  9.901e-01   1.043  0.29714
## sys           -2.084e-02  9.443e-03  -2.207  0.02732 *
```

```
## hra           -2.915e-03  1.032e-02  -0.282  0.77761
## preYes         1.279e+00  7.022e-01   1.822  0.06842 .
## typeEmergency  3.748e+00  1.342e+00   2.792  0.00523 **
## fraYes         1.649e+00  1.093e+00   1.509  0.13139
## po2<= 60      -6.765e-01  9.402e-01  -0.720  0.47179
## ph< 7.25       1.771e+00  1.212e+00   1.461  0.14410
## pco> 45       -2.084e+00  1.165e+00  -1.789  0.07361 .
## bic< 18       -2.623e-01  8.967e-01  -0.293  0.76985
## cre> 2.0       1.004e-01  1.131e+00   0.089  0.92925
## locStupor      3.771e+01  2.487e+03   0.015  0.98790
## locComa        3.458e+00  1.342e+00   2.578  0.00994 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 200.16  on 199  degrees of freedom
## Residual deviance: 112.17  on 178  degrees of freedom
## AIC: 156.17
##
## Number of Fisher Scoring iterations: 17

##     locStupor     raceBlack   (Intercept) typeEmergency        canYes
##  37.705271929  16.174546113   5.548262182   3.747879245   3.482602382
##       locComa       pco> 45      ph< 7.25        fraYes       preYes
##   3.458374688   2.083580267   1.770978482   1.649453062   1.279496861
##        cprYes  genderFemale      po2<= 60   serSurgical     raceOther
##   1.032231804   0.721456637   0.676510682   0.673862547   0.582918678
##       bic< 18        crnYes        infYes      cre> 2.0           age
##   0.262346172   0.119137902   0.108119860   0.100402678   0.056452457
##           sys           hra
##   0.020839745   0.002915162
```

Point B

So, as we observed in the previous point, we can absolutely make a case for a separation problem. Therefore, we start by pool stratified variable levels into fewer ones.

```
##     gender1     gender2       race1       race2       race3        ser1
##      "Male"    "Female"     "White"     "Black"     "Other"   "Medical"
##        ser2        can1        can2        crn1        crn2        inf1
##  "Surgical"        "No"       "Yes"        "No"       "Yes"        "No"
##        inf2        cpr1        cpr2        pre1        pre2       type1
##       "Yes"        "No"       "Yes"        "No"       "Yes"  "Elective"
##       type2        fra1        fra2        po21        po22         ph1
## "Emergency"        "No"       "Yes"      "> 60"     "<= 60"   ">= 7.25"
##         ph2        pco1        pco2        bic1        bic2        cre1
##    "< 7.25"     "<= 45"      "> 45"     ">= 18"      "< 18"    "<= 2.0"
##        cre2        loc1        loc2        loc3
##     "> 2.0"   "Nothing"    "Stupor"      "Coma"
```

From the listing above we see that only two factors, namely **race** and **loc** have such more than 2 levels, which were already binarized in the beginning. By refitting the full model, we see that pooling the levels together

somewhat improves the modelling, but we still are far from a satisfying results, given the large coefficients still, and standard errors. At this point, one could turn some of the categorical variables into continuous variables, or eliminate some variables. The second option will introduce some bias, but this might be the only thing to do given the sample that we have. We also note that with interaction terms, this problem would be even more accentuated, and we would not be able to determine whether such interactions would be relevant given the separation problem.

```
##
## Call:
## glm(formula = sta ~ ., family = binomial(link = "logit"), data = df2[,
##     2:length(df2)])
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.80396  -0.56064  -0.20440  -0.08635   2.97729
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -6.086979   2.260515  -2.693  0.00709 **
## age            0.056393   0.018624   3.028  0.00246 **
## genderFemale  -0.639725   0.531393  -1.204  0.22864
## serSurgical   -0.673522   0.601902  -1.119  0.26315
## canYes         3.107051   1.045846   2.971  0.00297 **
## crnYes        -0.035708   0.801647  -0.045  0.96447
## infYes        -0.204933   0.553191  -0.370  0.71104
## cprYes         1.053483   1.006614   1.047  0.29530
## sys           -0.015472   0.008497  -1.821  0.06864 .
## hra           -0.002769   0.009607  -0.288  0.77317
## preYes         1.131942   0.671450   1.686  0.09183 .
## typeEmergency  3.079583   1.081584   2.847  0.00441 **
## fraYes         1.411402   1.029705   1.371  0.17047
## po2<= 60       0.073822   0.857044   0.086  0.93136
## ph< 7.25       2.354078   1.208804   1.947  0.05148 .
## pco> 45       -3.018442   1.253448  -2.408  0.01604 *
## bic< 18       -0.709284   0.909777  -0.780  0.43561
## cre> 2.0       0.295143   1.116925   0.264  0.79159
## race_bin       0.565729   0.926828   0.610  0.54160
## loc_bin        5.232292   1.226303   4.267 1.98e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 200.16  on 199  degrees of freedom
## Residual deviance: 120.78  on 180  degrees of freedom
## AIC: 160.78
##
## Number of Fisher Scoring iterations: 6
```

Point D

Now we want to improve the base model and the separation problem by using stepwise procedures based on AIC and BIC. We fit backward stepwise model selection.

The first based on AIC gives out:

$$sta \sim age + can + sys + type + ph + pco + loc_b in$$
$$AIC = 144.4$$

For the backward stepwise model selection absed on BIC, we have:

$$sta \sim age + can + type + loc_b in$$
$$AIC = 165.63$$

```
s1_AIC<- step(m2, direction = "backward", k = 2)
n <- dim(df2)[1]
s1_BIC <- step(m2, direction = "backward", k = log(n))
```

Point E

Now that we have obtained our models, we want to compare the full model with the stepwise selected models by benchmarking their log-likelihoods ant the in-sample misclassification.

```
pred_AIC <- ifelse(s1_AIC$fitted.values > 0.5, 1, 0)
pred_BIC <- ifelse(s1_BIC$fitted.values > 0.5, 1, 0)
pred_full <- ifelse(m2$fitted.values > 0.5, 1, 0)

## Misclassification errors
mis_AIC <- mean(pred_AIC != df2$sta)
mis_BIC <- mean(pred_BIC != df2$sta)
mis_FULL <- mean(pred_full != df2$sta)
results_mis <- c(mis_AIC, mis_BIC, mis_FULL)

## Loglikelihoods
lik_AIC <- logLik(s1_AIC)
lik_BIC <- logLik(s1_BIC)
lik_FULL <- logLik(m2)
results_lik <- c(lik_AIC, lik_BIC, lik_FULL)

## Matrix
result_mat <- matrix(c(results_mis, results_lik), nrow = 2)
rownames(result_mat) <- c("Misclassification", "Log-Lik")
colnames(result_mat) <- c("AIC", "BIC", "FULL")
result_mat
```

```
##                     AIC    BIC      FULL
## Misclassification 0.13   0.11 -69.56732
## Log-Lik           0.14 -64.22 -60.38917
```

From these results, we can see that the full model performs better in both metrics. However we need to keep in mind that this is only true in-sample, and a thorough analysis should be carried out also out-of-sample in such cases. This is also expected, since by eliminating covariates, we increase bias to reduce possible variance out-of-sample.