Board Report

Leslie O'Bray

April 22, 2018

```
#####################################################################
###########  Regression Analysis - Board reviews  ###############
#####################################################################

library(vcd)
```

```
## Loading required package: grid
```

```
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
library(caTools)
library(ROCR)
```

```
## Loading required package: gplots
```

```
##
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(car)
library(caret)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
### Initialize data

load("/home/leslie/Desktop/StatsLab/snsf_data.RData")
source("/home/leslie/Desktop/StatsLab/stats-lab-snsf/Cleaning Functions.R")
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter

## The following object is masked from 'package:graphics':
##
##     layout

## The 'lsmeans' package is being deprecated.
## Users are encouraged to switch to 'emmeans'.
## See help('transition') for more information, including how
## to convert 'lsmeans' objects and scripts to work with 'emmeans'.

##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date

##
## Attaching package: 'plyr'

## The following object is masked from 'package:lubridate':
##
##     here

## The following objects are masked from 'package:plotly':
##
##     arrange, mutate, rename, summarise
source("/home/leslie/Desktop/StatsLab/stats-lab-snsf/Data for Regression.R")
```

Questions for Janine:

-For variable importance question, could use a random forest to assess variable importance? Or could standardize coefficients -Multiple testing problem??

Prepare the data and fit the full model:

```
board_data <- prepare_data_board_log_regression(final.apps, internal = final.internal, external = final
```

```
board_log_regression <- glm(board_data$IsApproved ~ Gender + Division + Age + IsContinuation + InstType
                            Ranking + OverallGrade + Gender:Division, family="binomial", data = board_
```

In fitting the regression, the summary shos that OverallGrade, Ranking, InstType and Age are all significant predictors.

```
(summary(board_log_regression))
```

```
##
## Call:
## glm(formula = board_data$IsApproved ~ Gender + Division + Age +
##     IsContinuation + InstType + AmountRequested + Ranking + OverallGrade +
##     Gender:Division, family = "binomial", data = board_data)
##
## Deviance Residuals:
```

```
##     Min       1Q   Median       3Q      Max
## -3.6139  -0.3908   0.0436   0.5734   2.6222
##
## Coefficients:
##                           Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -1.222e+01  8.979e-01 -13.605  < 2e-16 ***
## Genderf                1.390e-01  3.035e-01   0.458   0.6470
## DivisionDiv 2         -1.803e-01  2.648e-01  -0.681   0.4959
## DivisionDiv 3          2.834e-02  2.554e-01   0.111   0.9117
## Age                   -2.531e-02  1.055e-02  -2.400   0.0164 *
## IsContinuation1        4.185e-01  2.123e-01   1.972   0.0486 *
## InstTypeOther          2.207e-01  3.916e-01   0.564   0.5731
## InstTypeUAS/UTE       -8.982e-02  3.754e-01  -0.239   0.8109
## InstTypeUni           -2.842e-02  2.301e-01  -0.124   0.9017
## AmountRequested       -2.928e-07  3.108e-07  -0.942   0.3460
## Ranking                2.932e+00  1.699e-01  17.251  < 2e-16 ***
## OverallGrade           5.754e-01  1.317e-01   4.369 1.25e-05 ***
## Genderf:DivisionDiv 2 -3.778e-01  5.063e-01  -0.746   0.4556
## Genderf:DivisionDiv 3  1.897e-01  4.471e-01   0.424   0.6714
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2243.9  on 1622  degrees of freedom
## Residual deviance:  972.3  on 1609  degrees of freedom
## AIC: 1000.3
##
## Number of Fisher Scoring iterations: 6
```

In checking for correlation among coefficients, we only don't see any alarming values (VIF > 5).

```
(vif(board_log_regression))
```

```
##                     GVIF Df GVIF^(1/(2*Df))
## Gender          2.549886  1        1.596836
## Division        2.620963  2        1.272375
## Age             1.095397  1        1.046612
## IsContinuation  1.201700  1        1.096221
## InstType        1.559616  3        1.076886
## AmountRequested 1.111930  1        1.054481
## Ranking         1.123041  1        1.059736
## OverallGrade    1.118601  1        1.057640
## Gender:Division 3.180599  2        1.335449
```

Check VariableImportance. We see that Ranking is orders of magnitude more impactful than any of the other variables.

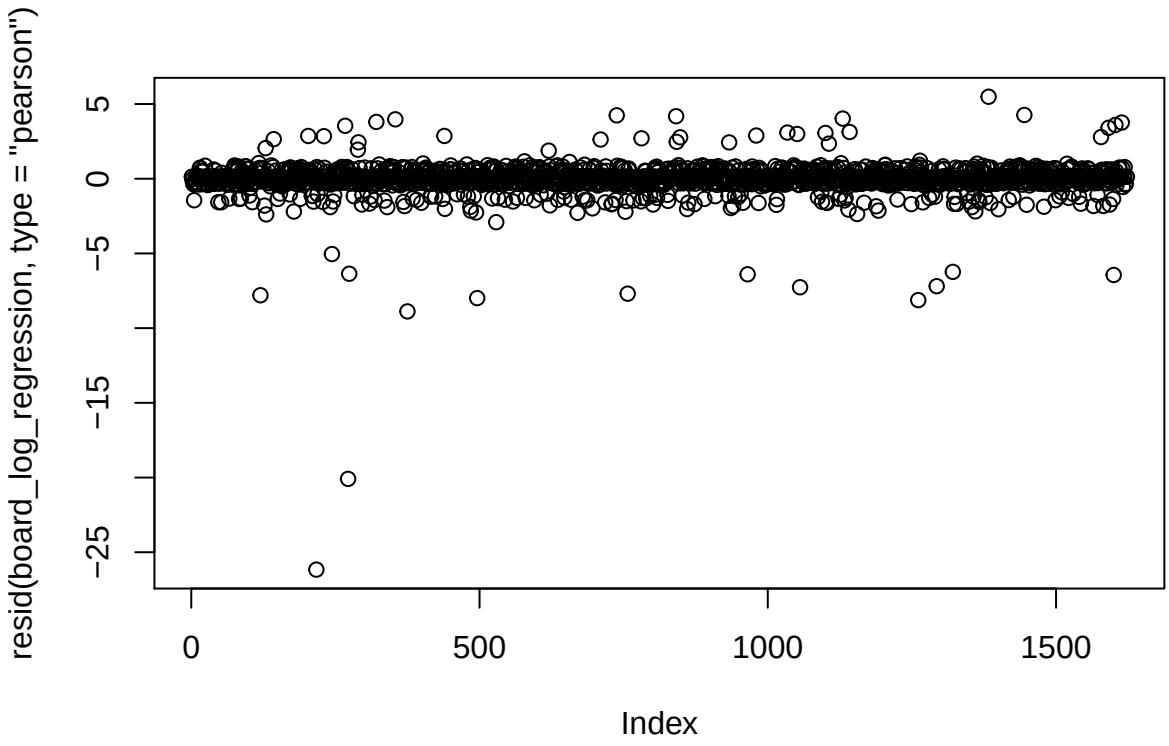
```
(board_variable_importance <- varImp(board_log_regression))
```

```
##                         Overall
## Genderf               0.4578914
## DivisionDiv 2         0.6810002
## DivisionDiv 3         0.1109545
## Age                   2.3997815
## IsContinuation1       1.9716641
```

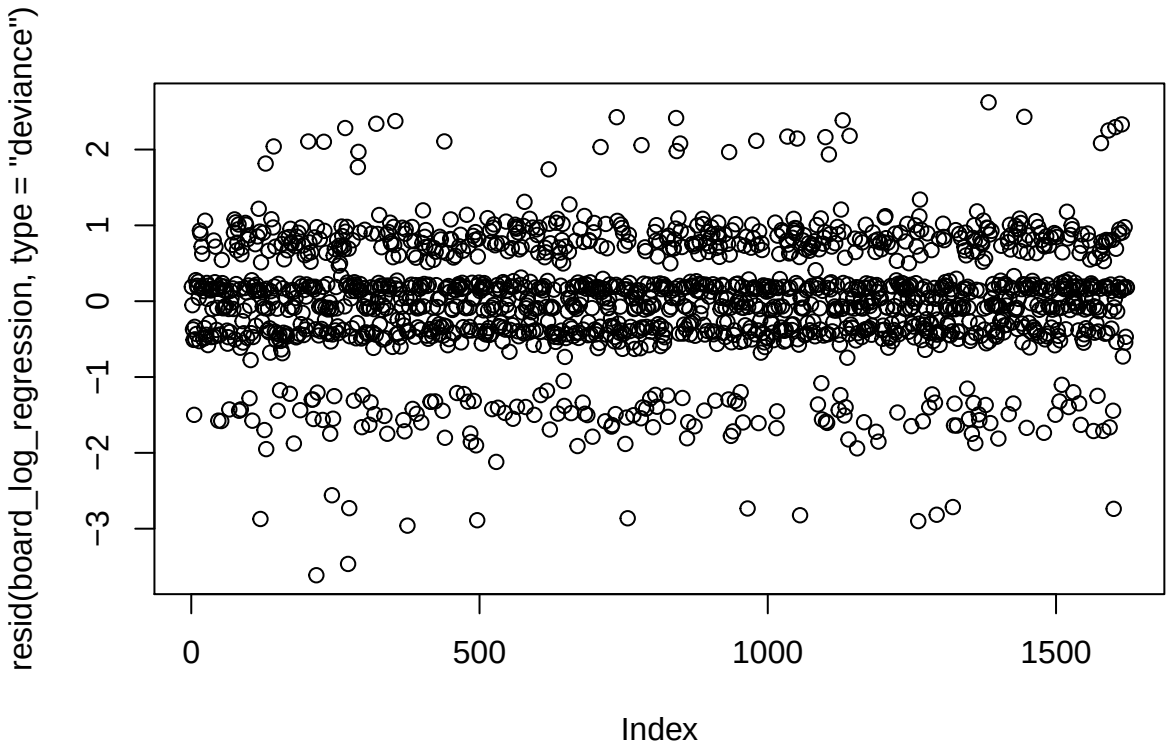
```
## InstTypeOther           0.5635433
## InstTypeUAS/UTE         0.2392943
## InstTypeUni             0.1235004
## AmountRequested         0.9422842
## Ranking                17.2511486
## OverallGrade            4.3689878
## Genderf:DivisionDiv 2   0.7461795
## Genderf:DivisionDiv 3   0.4242813
```

Check diagnostics: residuals:

```
plot(resid(board_log_regression, type="pearson"))
```




```
plot(resid(board_log_regression, type="deviance"))
```



Checking the residuals, they seem to be expectation 0, with a few outliers. However, in looking at the deviance, there appears to be some structure in the data. Should this indicate we need a new model fit?

Test out other models, to do a hierarchical model comparision:

```
board_log_regression2 <- glm(board_data$IsApproved ~ Gender + Division + Age + AmountRequested +
                             OverallGrade + Gender:Division, family="binomial", data = board_data)

board_log_regression1 <- glm(board_data$IsApproved ~ Gender + Division + Age + AmountRequested +
                               Ranking + Gender:Division, family="binomial", data = board_data)

board_log_regression4 <- glm(board_data$IsApproved ~ Gender + Ranking + OverallGrade, family="binomial"

anova(board_log_regression2, board_log_regression, test="Chisq") # significantly different than small m
```

```
## Analysis of Deviance Table
##
## Model 1: board_data$IsApproved ~ Gender + Division + Age + AmountRequested +
##     OverallGrade + Gender:Division
## Model 2: board_data$IsApproved ~ Gender + Division + Age + IsContinuation +
##     InstType + AmountRequested + Ranking + OverallGrade + Gender:Division
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1      1614     1651.8
## 2      1609      972.3  5   679.47 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(board_log_regression1, board_log_regression, test="Chisq") # significantly different than small m
```

```
## Analysis of Deviance Table
##
## Model 1: board_data$IsApproved ~ Gender + Division + Age + AmountRequested +
##     Ranking + Gender:Division
## Model 2: board_data$IsApproved ~ Gender + Division + Age + IsContinuation +
##     InstType + AmountRequested + Ranking + OverallGrade + Gender:Division
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1      1614     998.17
## 2      1609     972.30  5   25.871 9.452e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(board_log_regression4, board_log_regression, test="Chisq") # not significiantly different than sm
```

```
## Analysis of Deviance Table
##
## Model 1: board_data$IsApproved ~ Gender + Ranking + OverallGrade
## Model 2: board_data$IsApproved ~ Gender + Division + Age + IsContinuation +
##     InstType + AmountRequested + Ranking + OverallGrade + Gender:Division
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1      1619      985.3
## 2      1609      972.3 10   13.004   0.2234
```

```r
1-pchisq(board_log_regression1$dev-board_log_regression$dev, df=(board_log_regression1$df.res-board_log
```

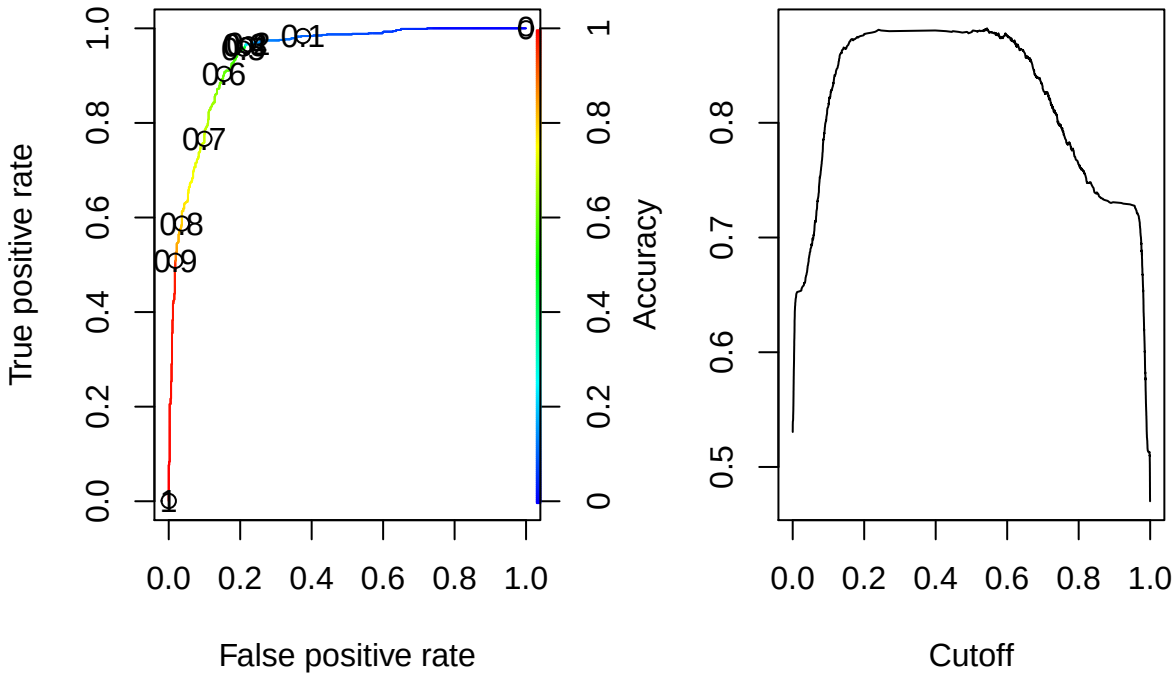
```
## [1] 9.451648e-05
```

Doing this model comparision, don't we have a big multiple testing problem? Should we avoid doing any kind of model comparison and instead 1) validate model fit, 2) assess performance, 3) compute variable importance?

Do CV to see if model fit is decent:

```r
########################################################################
#################### CV to assess model fit ############################
########################################################################


assess_board_model<- function(data=board_data, Div="All",
                      SplitRatio=0.8, cutoff=0.5 ){

  if (Div == "All"){
    final.data <- data
  }
  else {
    final.data<- subset(data,Division==Div, select = -(Division))
  }

  ### spliting the data into train and test set

  if (SplitRatio<1){
    split<-sample.split(final.data$IsApproved, SplitRatio = SplitRatio)
    Train<-subset(final.data, split=="TRUE")
    Test <-subset(final.data, split=="FALSE")
  } else {
```

```
    Test<-Train<-final.data
  }

  #### fitting the model

  # Cutoff
  cutoff <- cutoff

  # Optimize the model

  Model <- glm(Train$IsApproved ~ .-(ProjectID),data=Train,
               family="binomial")

  ### Testing the Treshold
  par(mfrow=c(1,2))
  predictor<-predict(Model, Test, type="response")
  ROCRPred<- prediction(predictor,Test$IsApproved)
  ROCRPerf<- performance(ROCRPred,"tpr","fpr")
  plot(ROCRPerf, colorize=TRUE, print.cutoffs.at=seq(0,1,by=0.1))

  ### with respect to accuracy
  ROCRACC<- performance(ROCRPred,"acc")
  plot(ROCRACC)

  ### Confusion Matrices
  AccTable<-table(ActualValue=Test$IsApproved,Prediction=predictor>=cutoff)
  accuracy<-(sum(diag(AccTable))/sum(AccTable))

  ### Return
  print(paste("Regresion for External Reviews.   ", "Division: ", Div))
  return(list(Model= summary(Model),
              #`Confidence Intervals`=confint(Model),
              `Confusion Matrix`=AccTable,
              `Percentage of data used for Training`=paste(SplitRatio*100,"%"),
              `Accuracy`=paste(round(accuracy,2)*100,"%")))
}


board.all <- assess_board_model(data=board_data, Div="All",
                        SplitRatio=1,cutoff=0.5)
```




```
## [1] "Regresion for External Reviews.    Division:  All"
```

```
board.all$Accuracy
```

```
## [1] "88 %"
```