R Notebook

Imports

```r
library("dplyr")
```

```
## Warning: replacing previous import 'lifecycle::last_warnings' by
## 'rlang::last_warnings' when loading 'pillar'
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library("DataExplorer")
```

```
## Warning: package 'DataExplorer' was built under R version 4.1.3
```

```r
library("caret")
```

```
## Warning: package 'caret' was built under R version 4.1.3
```

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

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

```r
library("randomForest")
```

```
## Warning: package 'randomForest' was built under R version 4.1.3
```

```
## randomForest 4.7-1.1
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

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

## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library("Hmisc")
```

```
## Warning: package 'Hmisc' was built under R version 4.1.3

## Loading required package: survival

##
## Attaching package: 'survival'

## The following object is masked from 'package:caret':
##
##     cluster

## Loading required package: Formula

##
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:dplyr':
##
##     src, summarize

## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```
library("car")
```

```
## Warning: package 'car' was built under R version 4.1.3

## Loading required package: carData

## Warning: package 'carData' was built under R version 4.1.3

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode
```

```r
library("glmnet")
```

```
## Warning: package 'glmnet' was built under R version 4.1.3

## Loading required package: Matrix

## Warning: package 'Matrix' was built under R version 4.1.3

## Loaded glmnet 4.1-4
```

```r
library("pROC")
```

```
## Warning: package 'pROC' was built under R version 4.1.3

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

##
## Attaching package: 'pROC'

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

```r
library("gbm")
```

```
## Warning: package 'gbm' was built under R version 4.1.3

## Loaded gbm 2.1.8.1
```

Setting Up

```r
df_train <- read.csv("joint_numeric.csv", header = TRUE , na.strings = c("na", "NA"),
                     stringsAsFactors = FALSE, sep = ",")

df_test <- read.csv("joint_test_numeric.csv", header = TRUE , na.strings = c("na", "NA"),
                     stringsAsFactors = FALSE, sep = ",")
```

```r
df_test[is.na(df_test)] = 0

df_train$X<-NULL
df_test$X<-NULL
df_train$fraudulent<-as.factor(df_train$fraudulent)
df_test$fraudulent<-as.factor(df_test$fraudulent)
df_train$department_n_first_personp<-NULL
#colnames(df_train)[colSums(is.na(df_train)) > 0]
#colnames(df_test)[colSums(is.na(df_test)) > 0]
```

```r
df_train = subset(df_train, select = c('has_company_logo', 'has_questions', 'fraudulent', 'has_departmen

df_test = subset(df_test, select = c('has_company_logo', 'has_questions', 'fraudulent', 'has_department

##IMPORTANTTTT
train_df <- df_train %>% select_if(function(col) length(unique(col))>1)
test_df <- df_test %>% select_if(function(col) length(unique(col))>1)
```

Original Logistic Regression Model

```r
# Log reg with everything
set.seed(123)
fraud_glm0 <- glm(fraudulent~., family=binomial, data=df_train)
```

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

```r
#summary(fraud_glm0)

#originally got this warning
#Warning: glm.fit: algorithm did not converge
#Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

# I found out it was due to ... Singularity means that your predictor variables are linearly dependent,
```

Correlation Plots

```r
without_df = subset(train_df, select = -c(fraudulent))
#sum(is.na(train_df))

res2 <- rcorr(as.matrix(train_df))

corrdisp <- cor(without_df, method="s")

# find indices of highly correlated attributes
highlycorrelated <- findCorrelation(corrdisp, cutoff= 0.98)
#highlycorrelated

#count(highlycorrelated) --> 69 (with 0.5)
#without_df[highlycorrelated]

edit2_df = subset(train_df, select = -c(highlycorrelated))
edit3_df = subset(test_df, select= -c(highlycorrelated))
```

Revised Logistic

```r
# from the correlation plot I noticed the below columns has a perfect colinearity so I got rid of one o
set.seed(123)

#This one surpasses all warnings after ridding of all perfect colinearity
fraud_glm5 <- glm(fraudulent~., family=binomial, data=edit2_df)
#summary(fraud_glm5)
```

```r
# Revising model before testing
fraud_glm6 = glm(fraudulent~
                   . - region_cat_SW - has_industry -department_sent_vader
                - department_n_second_personp -company_profile_n_chars
                - company_profile_n_hashtags -company_profile_n_uq_words
                -company_profile_n_first_person -company_profile_n_prepositions
                -department_n_charsperword -description_n_nonasciis -description_sent_afinn
                -description_sent_vader -description_n_first_person -description_n_first_personp
                -requirements_n_hashtags -requirements_sent_vader -requirements_n_second_personp
                - benefits_sent_bing -benefits_sent_syuzhet -industry_top -employment_type_Full.time
                -title_customer -title_teacher -title_assistant -required_experience_Entry.level
                -required_experience_Mid.Senior.level -fn_Finance -fn_Production
                - required_education_Professional - has_department ,
                 data = edit2_df, family = binomial)
#summary(fraud_glm6)
```

Analysis of Model

```r
predictTrain = predict(fraud_glm6, type = "response")
table(edit2_df$fraudulent, predictTrain >= 0.5)
```

```
##
##     FALSE  TRUE
##   0 13518    86
##   1   476   224
```

```r
accuracy = (244 + 13518) / nrow(edit2_df)
sensitivity = 244 / (244 + 476)
specificity = 13518 / (13518 + 86)
#sensitivity
#specificity

cat("accuracy: ", accuracy)
```

```
## accuracy:  0.9621085
```

```r
threshold=0.5
predicted_values<-ifelse(predict(fraud_glm6,type="response")>threshold,1,0)
actual_values<-fraud_glm6$y
conf_matrix<-table(predicted_values,actual_values)
#conf_matrix
#sensitivity(conf_matrix)
#specificity(conf_matrix)
```

Applying Test

```r
predictTest = predict(fraud_glm6, type = "response", newdata = edit3_df)

# no preference over error t = 0.5
edit3_df$fraudulent = as.numeric(predictTest >= 0.5)
table(edit3_df$fraudulent)
```

```
##
##    0    1
## 3508   68
```

```r
predicted_probabilities <- predict(fraud_glm5,
                           newdata=edit2_df,
                           type="response")

class_preds <- ifelse(predicted_probabilities >= 0.5, 1, 0)

# Make a table of predictions vs. actual
result_table <- table(class_preds,
                      edit2_df$fraudulent)

#result_table

confusionMatrix(data = factor(class_preds),
                reference = factor(edit2_df$fraudulent),
                positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction     0     1
##          0 13508   447
##          1    96   253
##
##                Accuracy : 0.962
##                  95% CI : (0.9588, 0.9651)
##     No Information Rate : 0.9511
##     P-Value [Acc > NIR] : 1.519e-10
##
##                   Kappa : 0.4649
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.36143
##             Specificity : 0.99294
##          Pos Pred Value : 0.72493
##          Neg Pred Value : 0.96797
##              Prevalence : 0.04894
##          Detection Rate : 0.01769
##    Detection Prevalence : 0.02440
##       Balanced Accuracy : 0.67719
##
##        'Positive' Class : 1
##
```