Capstone Intial Results

John Arnold

November 7, 2018

```
#Data load

df<-
data.frame(read.csv("C:/Users/jatosan/Desktop/CAPSTONE/semifinal/secondtry/2500_column_gone_2016Q1.csv"))

#Data Cleaning and Preparation
#Removing "current" rows
df<-subset(df,!(df$loan_status=="Current"))

#Coerce lates, grace periods and charge offs to 1 and fully paid to 0

#Get loan status col name index
grep("loan_status", colnames(df))

## [1] 12

#change class of loan status to numeric to coerce loan status attributes

gracep<-which(df$loan_status=="In Grace Period")
late1<-which(df$loan_status=="Late (16-30 days)")
late2<-which(df$loan_status=="Late (31-120 days)")
default<-which(df$loan_status=="Charged Off")

loanstat1<-c(gracep,late1,late2,default)

fpaid<-which(df$loan_status=="Fully Paid")

loanstat0<-fpaid

#change class of loan status to numeric to coerce
df$loan_status<-as.numeric(df$loan_status)
  #Coercing
  df[loanstat1,12]<-1
  df[loanstat0,12]<-0


#Remove percentage sign in df columns (int_rate, revol_util) and change to numeric for correlation matrix

int_rateclean<-as.numeric(gsub("%","",df$int_rate))
```

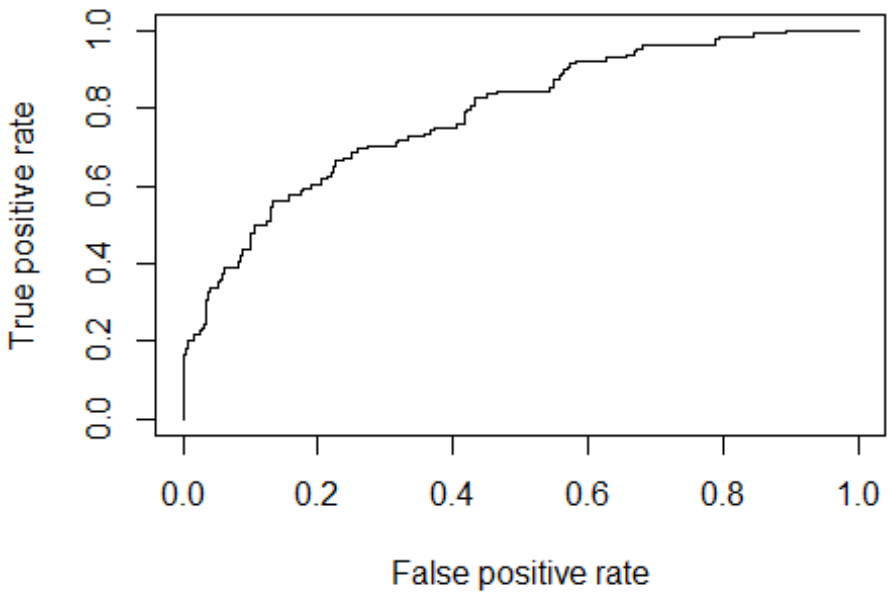
```r
df$int_rate<-int_rateclean

revol_utilcleann<-as.numeric(gsub("%","",df$revol_util))
df$revol_util<-revol_utilcleann

#Changing emp_length to numeric values

emp_lengthc1<-gsub(" years | year|s|+","",df$emp_length)
emp_lengthc2<-sub("10+","10",emp_lengthc1,fixed = TRUE)
emp_lengthc3<-sub("< 1","0",emp_lengthc2,fixed = TRUE)
emp_lengthc4<-sub("n/a","0",emp_lengthc3,fixed = TRUE)
emp_lengthc5<-as.numeric(emp_lengthc4)

df$emp_length<-emp_lengthc5

#Data Cleaning and Preparation continued
#sum(is.na(df$emp_title))
#sum(is.na(df$mths_since_last_delinq)) #646
#sum(is.na(df$mths_since_last_record)) #1134
#sum(is.na(df$mths_since_last_major_derog)) #992
#sum(is.na(df$mths_since_recent_bc_dlq)) #1051
#sum(is.na(df$mths_since_recent_inq)) #110
#sum(is.na(df$mths_since_recent_revol_delinq))#894
#sum(is.na(df$il_util)) #194

#Removing the columns with a sizable proportion of the rows are NAs

df<-subset(df ,select=-
c(mths_since_last_delinq,mths_since_last_record,mths_since_last_major_derog,m
ths_since_last_major_derog,mths_since_recent_bc_dlq,

mths_since_recent_inq,mths_since_recent_revol_delinq,revol_util,mths_since_rc
nt_il,il_util,all_util,bc_open_to_buy,bc_util,mo_sin_old_il_acct,mths_since_r
ecent_bc,percent_bc_gt_75,last_credit_pull_d))


#Removing columns full of 0s

length(which(df$collections_12_mths_ex_med==0)) #1367

## [1] 1367

length(which(df$num_tl_90g_dpd_24m==0)) #1325

## [1] 1325

df<-subset(df,select=-c(collections_12_mths_ex_med,num_tl_90g_dpd_24m))

df<-subset(df,select=-
```

```
c(emp_title,earliest_cr_line,grade,sub_grade,purpose,addr_state,verification_
status))

#Logistic Regression Modelling

glm.lr<-glm(df$loan_status ~.,data=df,family = binomial,maxit=100)

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

vim<-varImp(glm.lr)
impr<-which((varImp(glm.lr)>0.5))


t1<-apply(vim,2,function(x) x[order(-x)])
t1

##                               Overall
## tot_hi_cred_lim            4.05262350
## tot_cur_bal                3.79407180
## total_il_high_credit_limit 3.27648549
## total_rev_hi_lim           3.08209871
## open_acc_6m                2.78473341
## term 60 months             2.29090931
## dti                        2.26190275
## num_il_tl                  2.17788010
## num_op_rev_tl              1.82202045
## open_act_il                1.81723043
## int_rate                   1.81031448
## revol_bal                  1.77497292
## total_acc                  1.71890876
## num_rev_accts              1.67142248
## total_bal_il               1.58933020
## mort_acc                   1.53905150
## open_acc                   1.38736224
## num_tl_op_past_12m         1.26896910
## num_bc_tl                  1.26352392
## home_ownershipOWN          1.23850545
## mo_sin_rcnt_rev_tl_op      1.21708862
## initial_list_statusw       1.18456016
## open_rv_12m                1.16348306
## delinq_2yrs                1.15159567
## avg_cur_bal                1.11024994
## home_ownershipRENT         1.08996412
## installment                1.05932391
## open_il_12m                0.88198946
## max_bal_bc                 0.72996867
## inq_last_6mths             0.67160571
## num_bc_sats                0.65752154
## pct_tl_nvr_dlq             0.53953788
## mo_sin_old_rev_tl_op       0.52449216
```

```
## loan_amnt                   0.49092166
## num_sats                    0.48391385
## pub_rec_bankruptcies        0.44627656
## pub_rec                     0.41694515
## num_actv_bc_tl              0.41480107
## num_actv_rev_tl             0.41085990
## inq_last_12m                0.39753949
## total_cu_tl                 0.35839992
## open_il_24m                 0.32814589
## acc_open_past_24mths        0.32096082
## emp_length                  0.26714819
## num_accts_ever_120_pd       0.25578847
## annual_inc                  0.24942979
## mo_sin_rcnt_tl              0.12248878
## total_bc_limit              0.10112516
## tot_coll_amt                0.08598267
## num_rev_tl_bal_gt_0         0.07103166
## total_bal_ex_mort           0.06980736
## open_rv_24m                 0.05337691
## out_prncp                   0.03637960
## inq_fi                      0.03444153
## tax_liens                   0.01842961
```

```
#Training and test data indexing

trainindex <- sample(1:nrow(df), 0.7 * nrow(df))

train.set <- df[trainindex,]
test.set  <- df[-trainindex,]

#Prediction and accuracy

glm.pred<-predict(glm.lr,newdata=test.set,type='response')
glm.pred<-ifelse(glm.pred>0.5,1,0)

misclasserr<-mean(glm.pred ==test.set$loan_status)

print(paste('Accuracy',1-misclasserr))

## [1] "Accuracy 0.235154394299287"

glm<-glm.pred

#ROC curve, report on accuracies of models.


rocp<-predict(glm.lr,newdata=test.set,type="response")
pr<-prediction(rocp,test.set$loan_status)
```

```
perf <- performance(pr, measure = "tpr", x.measure = "fpr")

plot(perf)
```




```
auc <- performance(pr, measure = "auc")
auc <- auc@y.values[[1]]
auc

## [1] 0.7859695
```