Data Analytics and Machine Learning PS3

Jiaming Huang, Hogun Kim, Yichu Li, An Yang

2019/4/19

Question 1: Predicting Default

```
library(foreign)
library(ggplot2)
library(data.table)
setwd("/Users/jiaminghuang/Downloads")

loan <- as.data.table(read.dta("LendingClub_LoanStats3a_v12.dta"))
```

a. We will use the column "loan_status" as the indicator for whether the loan was paid or there was a default.

(i) Drop all rows where "loan_status" is not equal to either "Fully Paid" or "Charged Off."
Define the new variable Default as 1 (or TRUE) if "loan_status" is equal to "Charged Off", and 0 (or FALSE) otherwise.

```
new_loan <- loan[(loan_status == "Fully Paid") | (loan_status == "Charged Off")]
new_loan[,Default := ifelse(loan_status == "Fully Paid",0,1)]
unique(new_loan$loan_status)
```

```
## [1] "Fully Paid"  "Charged Off"
```

```
unique(new_loan$Default)
```

```
## [1] 0 1
```

(ii) Report the average default rate in the sample (number of defaults divided by total number of loans)

```
average_default_rate <- sum(new_loan$Default)/nrow(new_loan)
average_default_rate
```

```
## [1] 0.143535
```

b. LendingClub gives a "grade" to each borrower, designed as a score of each borrowers creditworthiness. The best grade is "A", the worst grade is "G".

(i) Using the glm function, run a logistic regression of the Default variable on the grade.
Report and explain the regression output. I.e., what is the interpretation of the coefficients? Do the numbers 'make sense'.

```
out = glm(Default~grade, family = "binomial",data = new_loan)
summary(out)
```

```
##
## Call:
## glm(formula = Default ~ grade, family = "binomial", data = new_loan)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.8827  -0.6077  -0.5053  -0.3511   2.3736
##
```

```
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.75542    0.04203  -65.56   <2e-16 ***
## gradeB       0.76143    0.05061   15.04   <2e-16 ***
## gradeC       1.15967    0.05153   22.50   <2e-16 ***
## gradeD       1.46001    0.05381   27.13   <2e-16 ***
## gradeE       1.69834    0.06030   28.17   <2e-16 ***
## gradeF       1.97319    0.07933   24.87   <2e-16 ***
## gradeG       2.01395    0.12800   15.73   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 32423  on 39411  degrees of freedom
## Residual deviance: 30914  on 39405  degrees of freedom
## AIC: 30928
##
## Number of Fisher Scoring iterations: 5
```

Explaination:

From the report of logistic regression:

a. The Intercept number -2.75542 means when a loan is tagged as grade A, the probability of default is 0.05978128.
b. The coefficients of all grades dummy variables are positive, which means lower the grade, larger the probability of default.
c. The z stat value of each grade dummy variables are pretty large, which means each grade is significant.
d. From the Null deviance and Residual deviance, we can tell that our model has lower residual deviance, which means our model is predictive.
e. The Fisher Scoring iterations means the number of guesses it took for computer.

(ii) Construct and report a test of whether the model performs better than the null model where only "beta0", and no conditioning information, is present in the logistic model.

```
test_stat = out$null.deviance-out$deviance
test_stat
```

```
## [1] 1508.097
```

```
k = out$df.null-out$df.residual
pvalue=1-pchisq(test_stat,df = k)
pvalue
```

```
## [1] 0
```

We construct the test a. to see if our model's residual deviance is smaller than the null deviance. b. and the chisq test for the null hypothesis that all betas equal to zero.

From the result, we can see that residual deviance is smaller than null deviance by 1508, also the p-value for the chisq test is 0 which means we reject the null hypothesis.

(iii) Construct the lift table and the ROC curve for this model. Explain the interpretion of the numbers in the lift table and the lines and axis in the ROC curve. Does the model perform better than a random guess?

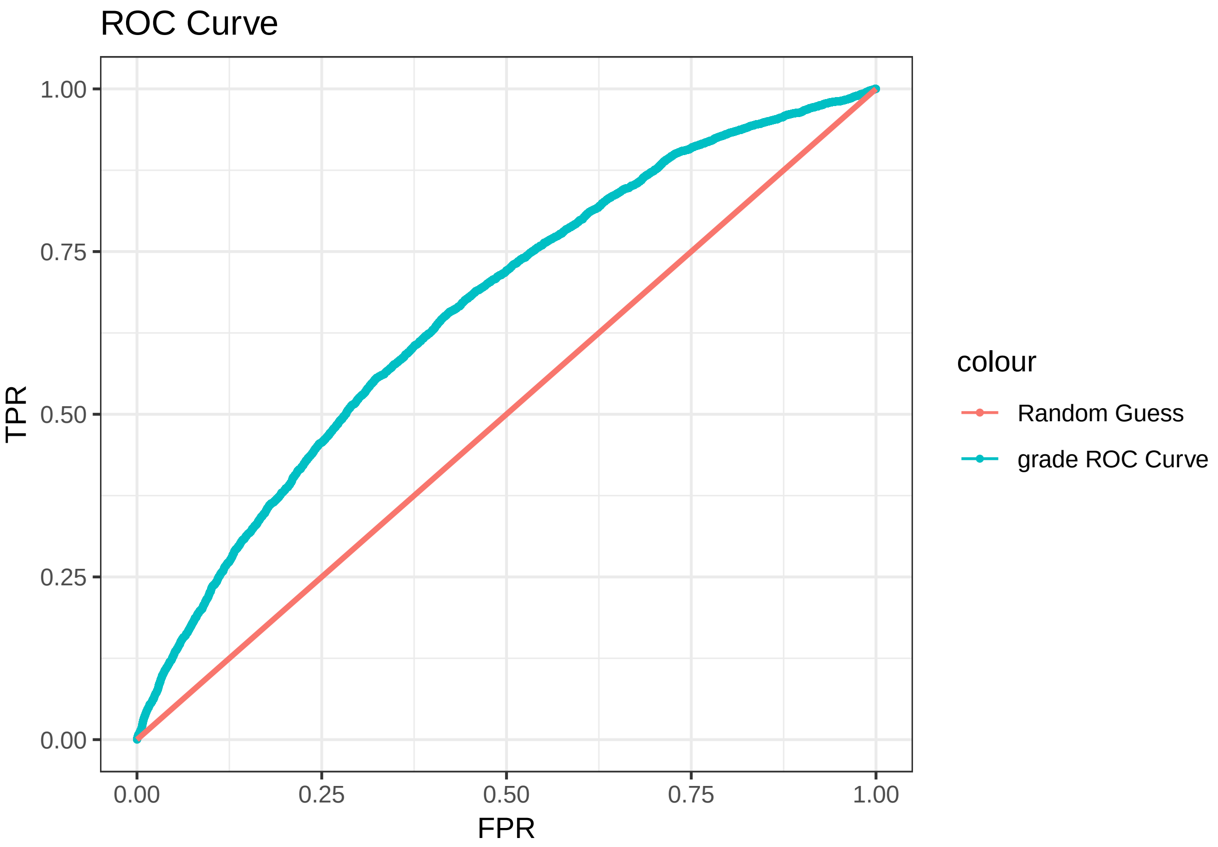
```
#lift table
phat=predict(out,type = "response")
```

```
deciles=cut(phat,breaks = quantile(phat,probs = c(seq(from=0,to=1,by=1/3))),include.lowest = TRUE)
deciles=as.numeric(deciles)
df = data.frame(deciles = deciles, phat = phat, default = new_loan$Default)
lift = aggregate(df,by=list(deciles),FUN = "mean",data = df)
lift = lift[,c(2,4)]
lift[,3]=lift[,2]/mean(new_loan$Default)
names(lift)=c("decile","Mean Response","Lift Factor")
lift
```

```
##   decile Mean Response Lift Factor
## 1      1    0.09235104   0.6434045
## 2      2    0.16857712   1.1744673
## 3      3    0.24214004   1.6869760
```

```
#ROC curve
simple_roc <- function(labels,scores)
{
  labels<-labels[order(scores,decreasing = TRUE)]
  data.frame(
    TPR=cumsum(labels)/sum(labels),
    FPR=cumsum(!labels)/sum(!labels),
    labels
  )
}

glm_simple_roc <- simple_roc(new_loan$Default == 1,phat)
TPR = glm_simple_roc$TPR
FPR = glm_simple_roc$FPR
qplot(FPR,TPR,xlab = "FPR",ylab = "TPR",col="grade ROC Curve",main = "ROC Curve",size = I(0.75))+
  geom_segment(aes(x=0,xend = 1,y=0,yend=1,size=I(1),col="Random Guess"))+
  theme_bw()
```



From the lift table, we see that the lift factor is increasing by decile, which means our model performance is better than random guess.

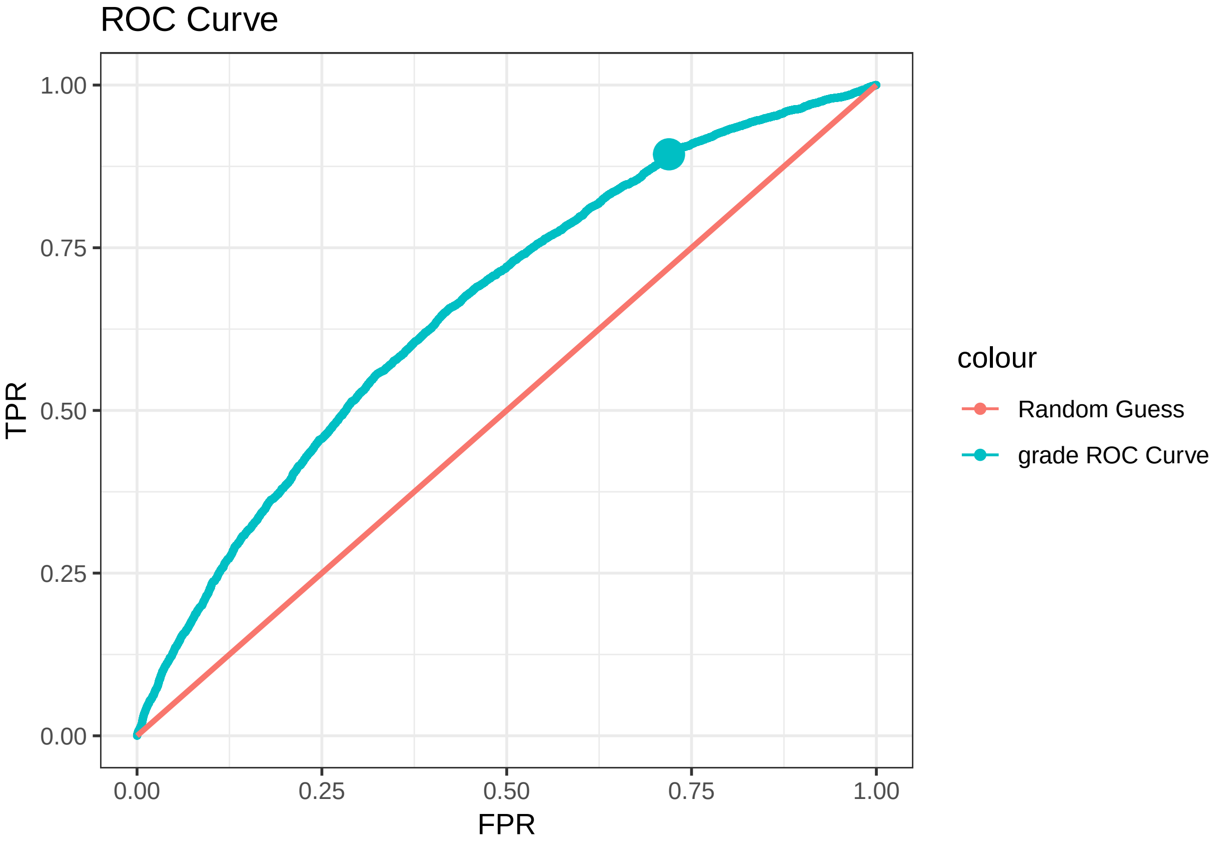
Also, as the ROC curve is above the 45 degree line, it proves the same conclusion.

(iv) Assume that each loan is for 100, and that you make a 1 profit if there is no default, but lose $10 if there is a default (both given in present value terms to keep things easy). Using data from the ROC curve (True Positive Rate and False Positive Rate) along with the average rate of default (total number of defaults divided by total number of loans), what is the cutoff default probability you should use as your decision criterion to maximize profits? Plot the corresponding point on the ROC curve.

```r
FPRs = c()
TPRs = c()
count = 1
profit = c()
P = average_default_rate
N = 1-P
for (g in sort(unique(new_loan$grade))) {
  TPRs[count] = sum(new_loan[grade >= g]$Default)/sum(new_loan$Default)
  FPRs[count] = (nrow(new_loan[grade >= g])-sum(new_loan[grade >= g]$Default))/(nrow(new_loan)-sum(new_
  profit[count] = N*(1-FPRs[count])*1+P*(1-TPRs[count])*(-10)
  count = count + 1
}
result <- data.table(
  grade = sort(unique(new_loan$grade)),
  TPR = TPRs,
  FPR = FPRs,
  profit = profit
)
result
```

```
##    grade        TPR         FPR       profit
## 1:     A 1.00000000 1.000000000  0.000000000
## 2:     B 0.89358317 0.719508221  0.087486045
## 3:     C 0.64079901 0.408354318 -0.008855171
## 4:     D 0.40162630 0.210665087 -0.182837714
## 5:     E 0.20222733 0.088609095 -0.364508272
## 6:     F 0.07477462 0.027136720 -0.494798539
## 7:     G 0.01785399 0.006280551 -0.558636963
```

```
qplot(FPR,TPR,xlab = "FPR",ylab = "TPR",col="grade ROC Curve",main = "ROC Curve",size = I(0.75))+
  geom_segment(aes(x=0,xend = 1,y=0,yend=1,size=I(1),col="Random Guess"))+
  geom_point(aes(x = 0.719508221,y = 0.89358317, size = I(5)))+
  theme_bw()
```



From the table above, we see that the maximum profit is when we set the cutoff at grade B, and the FPR is 0.719508221, the TPR is 0.89358317. The maximum profit is 0.087486045, which is pointed out on the ROC curve.

c. Next, we will see if it is possible to do better than the internal “grade”???variable, using other information about the borrower and the loan as provided by LendingClub.

(i) First, consider a logistic regression model that uses only loan amount (loan_amnt) and annual income (annual_inc) as explantory variables. Report the regression results.
Show the lift table, comparing to the ‘grade’???model from a. Plot the ROC curves of both the ‘grade’???model and the altnerative model. Which model performs better?

```
out2 = glm(Default~loan_amnt+annual_inc, family = "binomial",data = new_loan)
summary(out2)
```

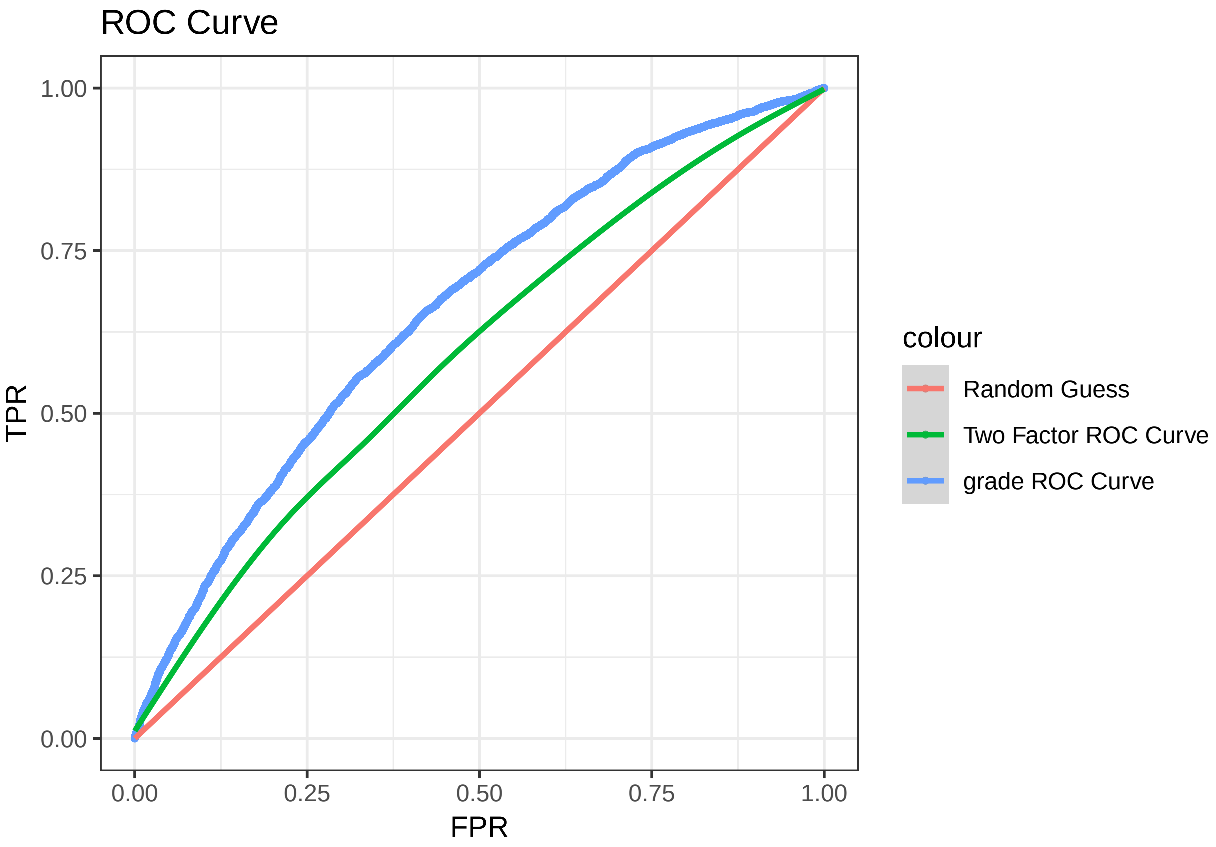
```
##
## Call:
## glm(formula = Default ~ loan_amnt + annual_inc, family = "binomial",
##     data = new_loan)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.8525  -0.5832  -0.5393  -0.4766   4.4804
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.725e+00  3.213e-02  -53.71   <2e-16 ***
## loan_amnt    3.484e-05  2.081e-06   16.74   <2e-16 ***
## annual_inc  -7.089e-06  4.663e-07  -15.20   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 32423  on 39411  degrees of freedom
## Residual deviance: 32027  on 39409  degrees of freedom
## AIC: 32033
##
## Number of Fisher Scoring iterations: 5
```

```
#lift table
phat2=predict(out2,type = "response")
deciles2=cut(phat2,breaks = quantile(phat2,probs = c(seq(from=0,to=1,by=0.1))),include.lowest = TRUE)
deciles2=as.numeric(deciles2)
df2 = data.frame(deciles2 = deciles2, phat2 = phat2, default2 = new_loan$Default)
lift2 = aggregate(df2,by=list(deciles2),FUN = "mean",data = df2)
lift2 = lift2[,c(2,4)]
lift2[,3]=lift2[,2]/mean(new_loan$Default)
names(lift2)=c("decile","Mean Response","Lift Factor")
lift2
```

```
##    decile Mean Response Lift Factor
## 1       1    0.08846641   0.6163405
## 2       2    0.10462164   0.7288930
## 3       3    0.11218030   0.7815538
## 4       4    0.12636664   0.8803893
## 5       5    0.13270743   0.9245652
## 6       6    0.14260340   0.9935098
## 7       7    0.15343647   1.0689832
## 8       8    0.14978421   1.0435381
## 9       9    0.20248668   1.4107133
## 10     10    0.22272958   1.5517444
```

```
#ROC Curve
glm_simple_roc2 <- simple_roc(new_loan$Default == 1,phat2)
TPR2 = glm_simple_roc2$TPR
FPR2 = glm_simple_roc2$FPR
```

```
qplot(FPR,TPR,xlab = "FPR",ylab = "TPR",col="grade ROC Curve",main = "ROC Curve",size = I(0.75))+
  geom_segment(aes(x=0,xend = 1,y=0,yend=1,size=I(1),col="Random Guess"))+
  geom_smooth(aes(x = FPR2,y=TPR2, col = "Two Factor ROC Curve"))+
  theme_bw()
```



From the decile 7 and 8 from the lift table of two factors and the ROC Curve, we can tell that the grade model performs better.

(ii) Now, include also information from the loan itself. In particular, include the maturity of the loan (term) and the interest rate (int_rate) in the logistic regression. Report the output. How does R handle the term???variable? In particular, what is the interpretation of the regression coefficient? Again show the lift table and ROC curve relative to the original 'grade' model. Now, which model is better? What is the likely explanation for why this new model performs better/worse?

```
out3 = glm(Default~loan_amnt+annual_inc+term+int_rate, family = "binomial",data = new_loan)
summary(out3)
```

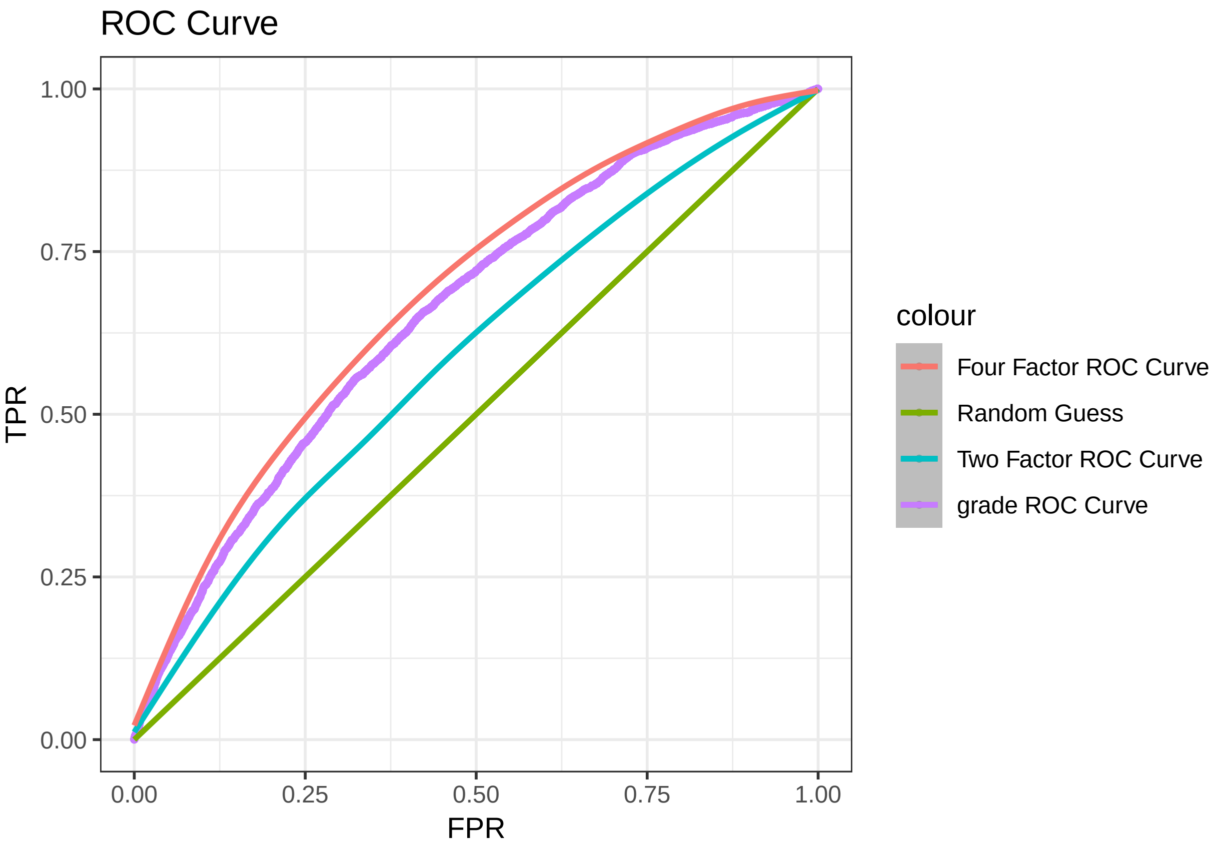
```
##
## Call:
## glm(formula = Default ~ loan_amnt + annual_inc + term + int_rate,
##     family = "binomial", data = new_loan)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2520  -0.5868  -0.4694  -0.3598   4.1684
##
## Coefficients:
```

```
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -3.266e+00  6.055e-02 -53.942   <2e-16 ***
## loan_amnt       1.176e-06  2.311e-06   0.509    0.611
## annual_inc     -6.117e-06  4.643e-07 -13.173   <2e-16 ***
## term 60 months  4.538e-01  3.564e-02  12.732   <2e-16 ***
## int_rate        1.349e+01  4.560e-01  29.575   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 32423  on 39411  degrees of freedom
## Residual deviance: 30418  on 39407  degrees of freedom
## AIC: 30428
##
## Number of Fisher Scoring iterations: 5
```

```
#lift table
phat3=predict(out3,type = "response")
deciles3=cut(phat3,breaks = quantile(phat3,probs = c(seq(from=0,to=1,by=0.1))),include.lowest = TRUE)
deciles3=as.numeric(deciles3)
df3 = data.frame(deciles3 = deciles3, phat3 = phat3, default3 = new_loan$Default)
lift3 = aggregate(df3,by=list(deciles3),FUN = "mean",data = df3)
lift3 = lift3[,c(2,4)]
lift3[,3]=lift3[,2]/mean(new_loan$Default)
names(lift3)=c("decile","Mean Response","Lift Factor")
lift3
```

```
##    decile Mean Response Lift Factor
## 1       1    0.03652968   0.2545002
## 2       2    0.06368942   0.4437206
## 3       3    0.08043644   0.5603961
## 4       4    0.09895965   0.6894463
## 5       5    0.11646790   0.8114253
## 6       6    0.14514083   1.0111880
## 7       7    0.15782796   1.0995785
## 8       8    0.18751586   1.3064124
## 9       9    0.23572697   1.6422965
## 10     10    0.31303907   2.1809255
```

```
#ROC Curve
glm_simple_roc3 <- simple_roc(new_loan$Default == 1,phat3)
TPR3 = glm_simple_roc3$TPR
FPR3 = glm_simple_roc3$FPR
qplot(FPR,TPR,xlab = "FPR",ylab = "TPR",col="grade ROC Curve",main = "ROC Curve",size = I(0.75))+
  geom_segment(aes(x=0,xend = 1,y=0,yend=1,size=I(1),col="Random Guess"))+
  geom_smooth(aes(x = FPR2,y=TPR2, col = "Two Factor ROC Curve"))+
  geom_smooth(aes(x = FPR3,y=TPR3, col = "Four Factor ROC Curve"))+
  theme_bw()
```



As we can see from the lift table and ROC Curve, the model including term factor and interest rate factor performs better than two factors model and grade model.

The coefficient of the term factor informs us that longer the term larger the probability of default.

The possible explaination for this result is that somehow the term and interest rate factors are related to grade, as longer the term and higher the interet rate, worse the loan and in result, lower the grade. Since term and interest rate, along with grade are all factors of loan itself. Then when we add more factors, the four factors model will perform better than the grade ones.

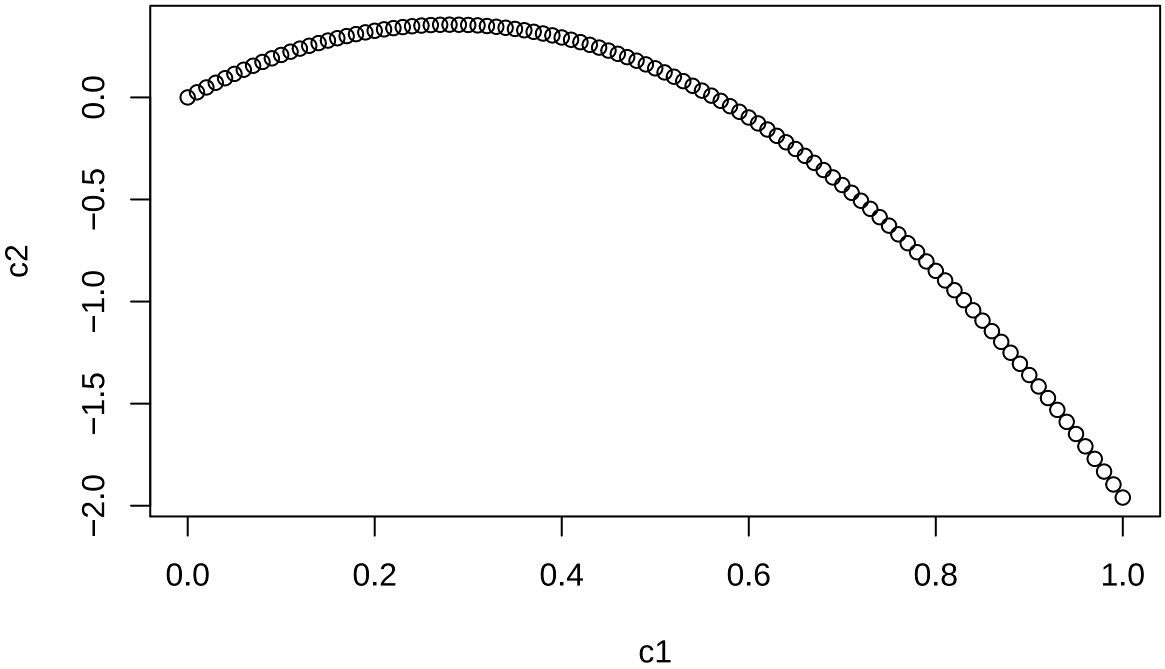
(iii) Create the squared of the interest rate and add this variable to the last model. Is the coefficient on this variable significant? Please give an intuition for what the coefficients on both int_rate and its squared value imply for the relationship between defaults and the interest rate.

```
new_loan[,int_rate2 := int_rate^2]
out4 = glm(Default~loan_amnt+annual_inc+term+int_rate+int_rate2, family = "binomial",data = new_loan)
summary(out4)
```

```
##
## Call:
## glm(formula = Default ~ loan_amnt + annual_inc + term + int_rate +
##     int_rate2, family = "binomial", data = new_loan)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0836  -0.5992  -0.4734  -0.3400   4.1124
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -4.035e+00  1.667e-01 -24.201  < 2e-16 ***
## loan_amnt       1.934e-06  2.307e-06   0.838    0.402    
## annual_inc     -5.982e-06  4.635e-07 -12.905  < 2e-16 ***
## term 60 months  4.680e-01  3.548e-02  13.190  < 2e-16 ***
## int_rate        2.553e+01  2.458e+00  10.385  < 2e-16 ***
## int_rate2      -4.494e+01  8.985e+00  -5.002 5.67e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 32423  on 39411  degrees of freedom
## Residual deviance: 30393  on 39406  degrees of freedom
## AIC: 30405
## 
## Number of Fisher Scoring iterations: 5
```

Initially, from the z value, we can tell that both coefficients are 99.9% significant. The intuition behind is that as shown in the graph below, when interest rate is small, higher the interst rate, higher the default probability. However, as interest rate becomes pretty large, the larger the interst rate, lower the default probability. I think that is because, when interst rate is low, default probability is mainly correlated with credit. Then higher the interest rate, lower the credit. However when interest rate becomes large, default probability is mainly correlated with profitability, then higher the interest rate, higher the profitability and lower the default probability.

```
c1 = seq(0,1,0.01)
c2 = 2.53*c1-4.49*c1^2
plot(x = c1, y = c2)
```


c2
0.0
−0.5
−1.0
−1.5
−2.0
0.0
0.2
0.4
0.6
0.8
1.0
c1