Date Analytics HW3

Jiaqi Li

April 21, 2019

Question 1

```
options(warn = -1)
library(ggplot2)
library(data.table)
library(dplyr)
```

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

```
## The following objects are masked from 'package:data.table':
##
##     between, first, last
```

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

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

```
library(haven)
```

part a

(i)

```
#############################################################################
#                                 (a)                                       #
#############################################################################
#------------------------------(i)----------------------------------#
data = read_dta("LendingClub_LoanStats3a_v12.dta") %>% as.data.table()
data = data[loan_status == "Fully Paid" | loan_status == "Charged Off",]
data[,Default := ifelse(loan_status == "Charged Off",1,0)]
```

(ii)

```
#------------------------------(ii)----------------------------------#
Def_rate = mean(data$Default)
Def_rate
```

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

The average default rate is abour 14.35%.

part b

(i)

```
#########################################################################
#                                (b)                                    #
#########################################################################
#------------------------------(i)---------------------------------#
reg = glm(Default~grade,family = "binomial",data=data)
summary(reg)
```

```
##
## Call:
## glm(formula = Default ~ grade, family = "binomial", data = data)
##
## 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
```

Based on the regression, all the coefficients are significant different than zero. Only the intercept is negative, which means only lending money to borrowers of grade A could decrease the overall default rate while lending money to anyone who is not in grade A level will increase the default rate. Lending money to borrowers in grade G, which is the lowest grade, will largely increase the default rate. These coefficients make sense since the lower the creditworthiness of the borrowers, the higher the probability they will default.

(ii)

```
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

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

```
Anova(reg)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Default
##       LR Chisq Df Pr(>Chisq)
## grade   1508.1  6  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The Chi-square is 1508.1 and F-stat is about 0, which means the F test shows that the model perform much better than the null model.

(iii)

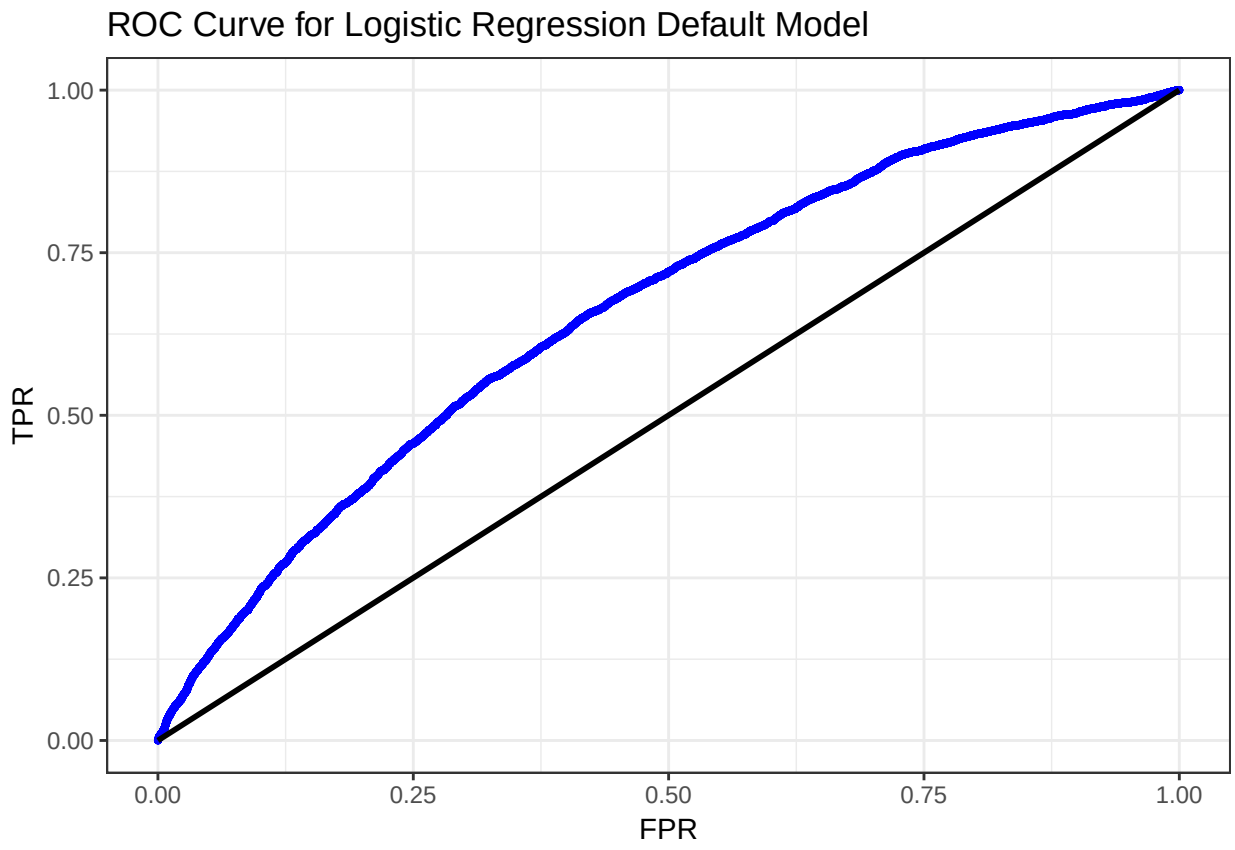
```
#-----------------------------(iii)--------------------------------#
phat_temp = jitter(predict(reg,type="response"))
deciles = cut(phat_temp,breaks = quantile(phat_temp,probs=c(seq(from=0,to=1,by=0.1))),
              include.lowest = TRUE)
deciles = as.numeric(deciles)
df = data.frame(deciles=deciles,phat=phat_temp,default=data$Default)
lift = aggregate(df,by=list(deciles),FUN="mean",data=df)
lift = lift[,c(2,4)]
lift[,3] = lift[,2]/mean(data$Default)
names(lift) = c("decile","Mean Response","Lift Factor")
lift
```

```
##    decile Mean Response Lift Factor
## 1       1    0.05834602   0.4064934
## 2       2    0.05658462   0.3942219
## 3       3    0.08779498   0.6116626
## 4       4    0.12052778   0.8397103
## 5       5    0.12661761   0.8821377
## 6       6    0.13245369   0.9227974
## 7       7    0.16188785   1.1278635
## 8       8    0.19360568   1.3488399
## 9       9    0.22100990   1.5397635
## 10     10    0.27650939   1.9264253
```

From the table, we can see that the lift factro is in a decreasing order in general. Only the 5th observation is smaller than the previous one, which is negligible. Such a monotonic drecreasing pattern indicates that our model is good model. The mean response rate for high fitted probabilities are much greater than for low fitted probabilities.

```
simple_roc <- function(labels, scores){
  labels <- labels[order(scores, decreasing=TRUE)]
  data.frame(TPR=cumsum(labels)/sum(labels),
             FPR=cumsum(!labels)/sum(!labels),
             labels)
}
phat = predict(reg,type="response")
glm_simple_roc <- simple_roc(data$Default=="1", phat)
TPR <- glm_simple_roc$TPR
FPR <- glm_simple_roc$FPR

qplot(FPR,TPR,xlab="FPR",ylab="TPR",col=I("blue"),
      main="ROC Curve for Logistic Regression Default Model",
      size=I(0.75)) +
  geom_segment(aes(x = 0, xend = 1, y = 0, yend = 1), size=I(1.0)) +
  theme_bw()
```



TPR is the True Positive Rate where True positive means predicted default is the true default. FPR is False Positive Rate where False Positive is the predicted default is not the true default.

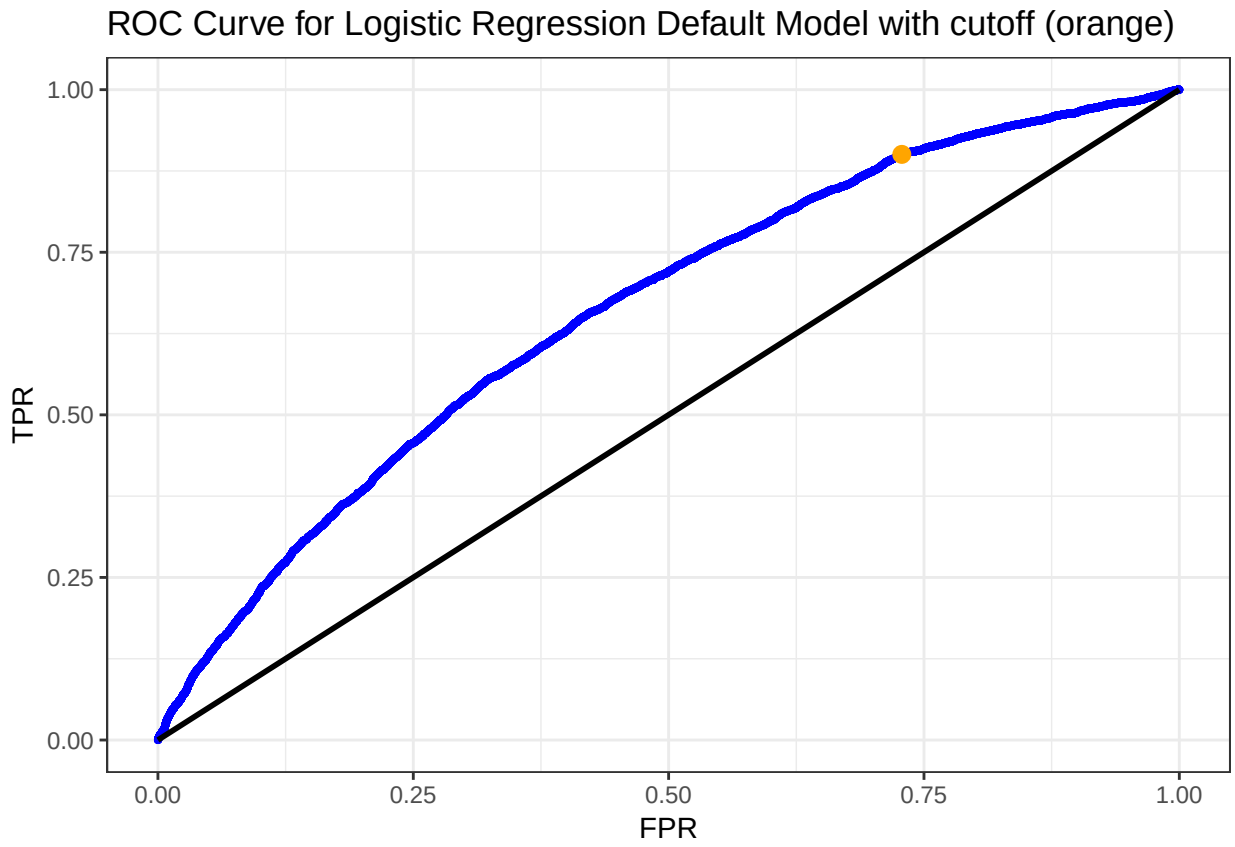
The random guess is the diagnal line in the above plot while the ROC curve, which is a measure of how informative a given model is, is the blue line. Based on the plot, we can see that the ROC is not so far from the random guess, which may indicate that the model is not very informative but the model does a better job than the random guess.

(iiii)

```
#-----------------------------(iiii)-------------------------------#
P = sum(data$Default)
N = -sum(data$Default-1)
TN = N*(1-FPR)
FN = P*(1-TPR)
temp1 = cbind(profit = TN*1-FN*10,phat = sort(phat,decreasing = T)) %>%
  as.data.frame()
temp2 = cbind(TPR,FPR) %>% as.data.frame()
temp = cbind(temp1,temp2)
cutoff = temp[which(temp$profit == max(temp$profit)),]
cutoff
```

```
##      profit       phat       TPR       FPR
## 1480   3541 0.05978153 0.9004773 0.7283069
```

```
qplot(FPR,TPR,xlab="FPR",ylab="TPR",col=I("blue"),
      main="ROC Curve for Logistic Regression Default Model with cutoff (orange)",
      size=I(0.75)) +
  geom_segment(aes(x = 0, xend = 1, y = 0, yend = 1), size=I(1.0)) +
  geom_point(aes(x=cutoff$FPR,y=cutoff$TPR),shape = 16,
             color = I("orange"), size = 3) +
  theme_bw()
```




```
default_prob = cutoff$phat
default_prob
```

```
## [1] 0.05978153
```

In this senario, the cutoff default probability I should use as my decision criterion to maximize profits is 5.978%. The cutoff point is shown on the ROC curve above.

part c

(i)

```
#############################################################################
#                                  (c)                                      #
#############################################################################
#-------------------------------(i)-----------------------------------#
reg_ci = glm(Default~loan_amnt+annual_inc,family = "binomial",data = data)
summary(reg_ci)
```

```
##
## Call:
## glm(formula = Default ~ loan_amnt + annual_inc, family = "binomial",
##     data = data)
##
```

```
## 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
```

Based on the summary above, all the variables are significant. Higher loan amount could increase the default rate and higher annual income would decrease the default rate based on the coefficient signs.

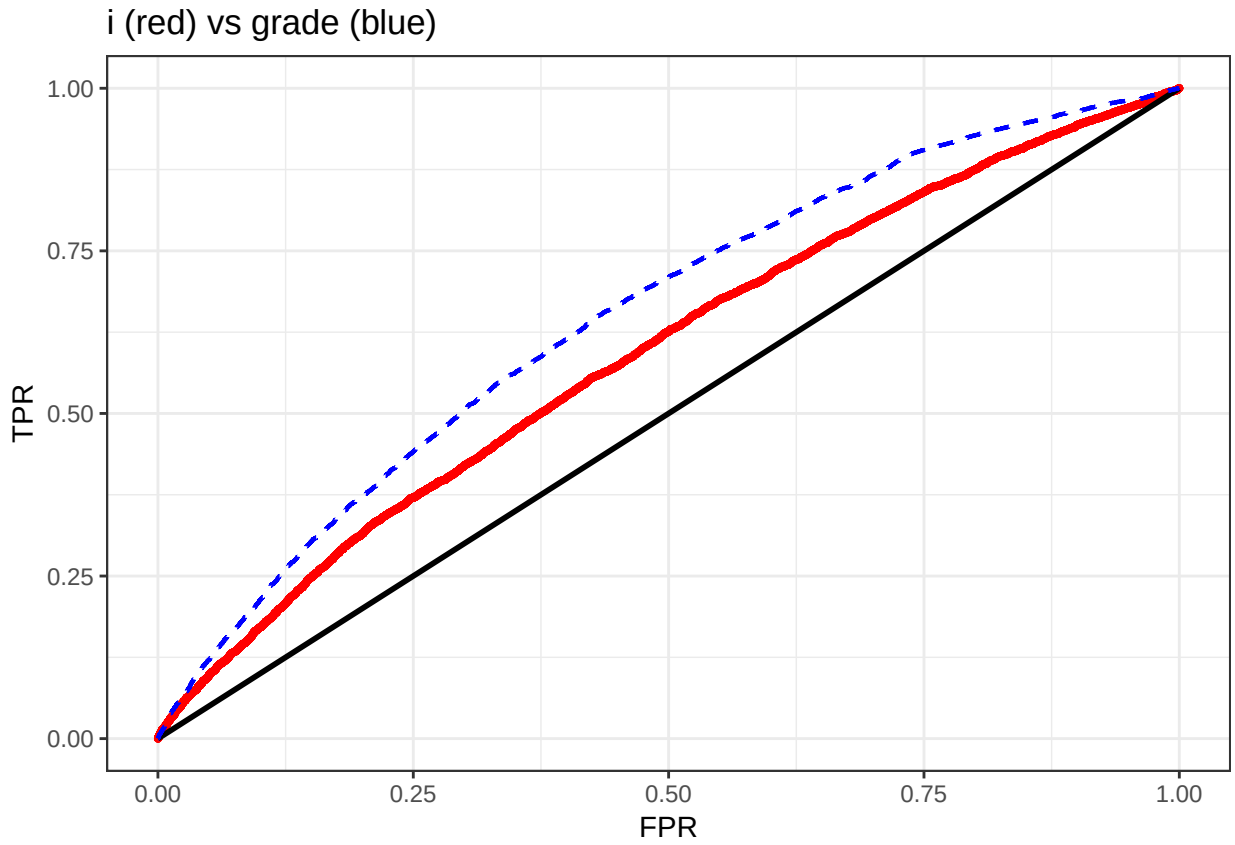
```
phat_ci = jitter(predict(reg_ci,type="response"))
deciles_ci = cut(phat_ci,breaks = quantile(phat_ci,probs=c(seq(from=0,to=1,by=0.1))),
              include.lowest = TRUE)
deciles_ci = as.numeric(deciles_ci)
df_ci = data.frame(deciles=deciles_ci,phat=phat_ci,default=data$Default)
lift_ci = aggregate(df_ci,by=list(deciles_ci),FUN="mean",data=df_ci)
lift_ci = lift_ci[,c(2,4)]
lift_ci[,3] = lift_ci[,2]/mean(data$Default)
names(lift_ci) = c("decile","Mean Response","Lift Factor")
lift_ci
```

```
##    decile Mean Response Lift Factor
## 1       1    0.08828006   0.6150422
## 2       2    0.10479574   0.7301060
## 3       3    0.11190053   0.7796047
## 4       4    0.12661761   0.8821377
## 5       5    0.13296118   0.9263330
## 6       6    0.14234966   0.9917420
## 7       7    0.15351434   1.0695257
## 8       8    0.14970820   1.0430086
## 9       9    0.20248668   1.4107133
## 10     10    0.22272958   1.5517444
```

```
phat_ci = predict(reg_ci,type="response")
glm_simple_roc_ci <- simple_roc(data$Default=="1", phat_ci)
TPR_ci <- glm_simple_roc_ci$TPR
FPR_ci <- glm_simple_roc_ci$FPR

qplot(FPR_ci,TPR_ci,xlab="FPR",ylab="TPR",col=I("red"),
      main="i (red) vs grade (blue)",
      size=I(0.75)) +
```

```
  geom_segment(aes(x = 0, xend = 1, y = 0, yend = 1), size=I(1.0)) +
  geom_line(aes(y=TPR),color=I("blue"),size=I(0.75),linetype = 2) +
  theme_bw()
```



Based on the lift table, the lift factors show a monotonic decreasing order with a very little noise in general, which means the model is a fairly good fit. However, when comparing with the model considering only the grade, this model using only loan amount and annual income is less informative because it is under the model considering only grade.

(ii)

```
#-----------------------------(ii)---------------------------------#
reg_cii = glm(Default~loan_amnt+annual_inc+term+int_rate,family = "binomial",data = data)
summary(reg_cii)
```

```
##
## Call:
## glm(formula = Default ~ loan_amnt + annual_inc + term + int_rate,
##     family = "binomial", data = data)
##
## 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
```

Based on the summary above, we can see that both the coefficients of maturity of the loan and the interest rate are positive, which means that both of these 2 variables could increase the default probability. By intuition, the longer maturity the loan take and the higher the interest rate, the higher the default probability coud be. Adding these 2 variables makes loan amount variable not significant anymore.

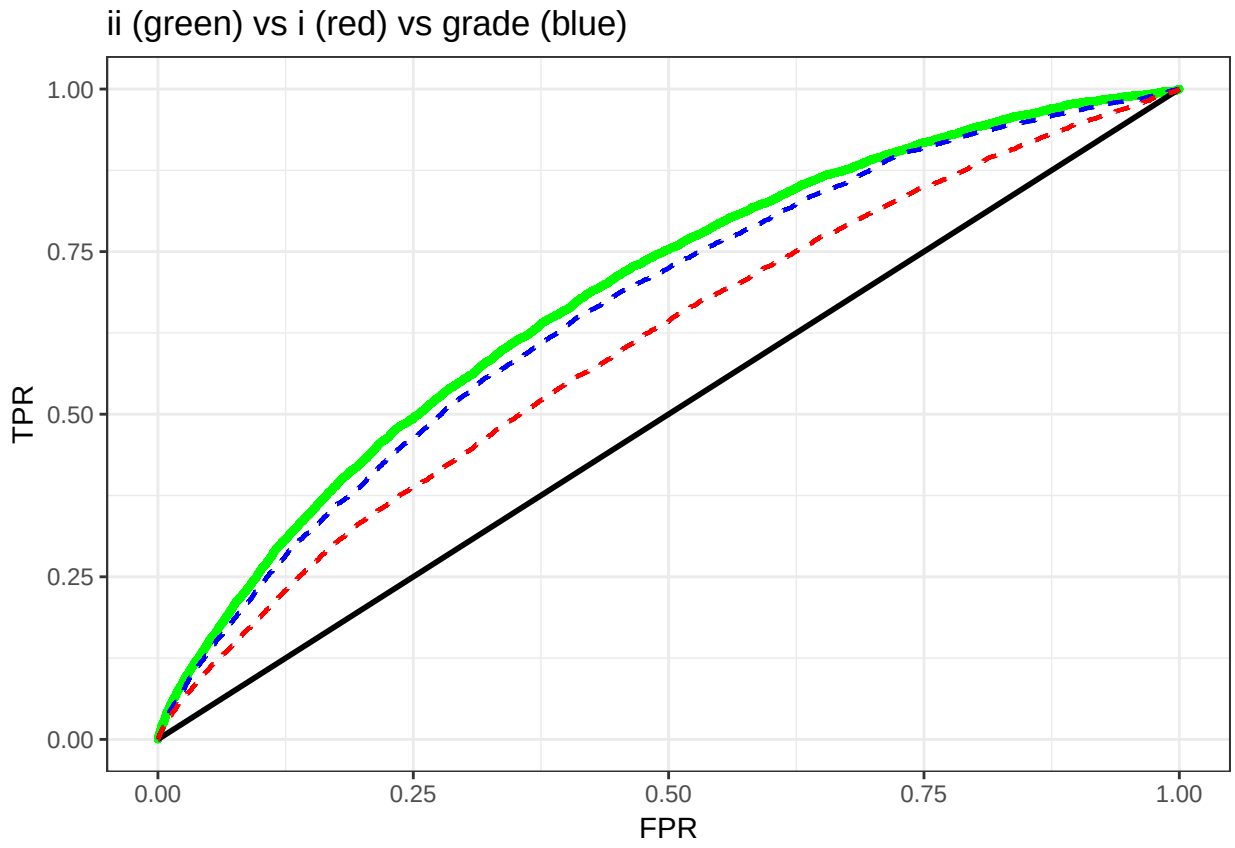
```
phat_cii = jitter(predict(reg_cii,type="response"))
deciles_cii = cut(phat_cii,breaks = quantile(phat_cii,probs=c(seq(from=0,to=1,by=0.1))),
                  include.lowest = TRUE)
deciles_cii = as.numeric(deciles_cii)
df_cii = data.frame(deciles=deciles_cii,phat=phat_cii,default=data$Default)
lift_cii = aggregate(df_cii,by=list(deciles_cii),FUN="mean",data=df_cii)
lift_cii = lift_cii[,c(2,4)]
lift_cii[,3] = lift_cii[,2]/mean(data$Default)
names(lift_cii) = c("decile","Mean Response","Lift Factor")
lift_cii
```

```
##    decile Mean Response Lift Factor
## 1       1    0.03652968   0.2545002
## 2       2    0.06368942   0.4437206
## 3       3    0.08018269   0.5586283
## 4       4    0.09921340   0.6912141
## 5       5    0.11646790   0.8114253
## 6       6    0.14488708   1.0094201
## 7       7    0.15808171   1.1013463
## 8       8    0.18751586   1.3064124
## 9       9    0.23598072   1.6440643
## 10     10    0.31278539   2.1791582
```

```
phat_cii = predict(reg_cii,type="response")
glm_simple_roc_cii <- simple_roc(data$Default=="1", phat_cii)
TPR_cii <- glm_simple_roc_cii$TPR
FPR_cii <- glm_simple_roc_cii$FPR

qplot(FPR_cii,TPR_cii,xlab="FPR",ylab="TPR",col=I("green"),
      main="ii (green) vs i (red) vs grade (blue)",
```

```
    size=I(0.75)) +
  geom_segment(aes(x = 0, xend = 1, y = 0, yend = 1), size=I(1.0)) +
  geom_line(aes(y=TPR),color=I("blue"),size=I(0.75),linetype = 2) +
  geom_line(aes(y=TPR_ci),color=I("red"),size=I(0.75),linetype = 2) +
  theme_bw()
```



Based on the lift table, the lift factors show a monotonic decreasing order, which means the model is a pretty good fit. When comparing with the model considering only the grade, this model using the loan amount, the annual income, the maturity of the loan, and the interest rate is more informative because it is above the model considering only grade. The reason could be that the default rate depends heavily on maturity and the interest rate. Longer maturity indicates higher volatility and higher interest rate indicates higher interest of the loan, which makes the loan harder to be paid in full.

(iii)

```
#----------------------------(iii)---------------------------------#
data[,int_rate_2 := int_rate^2]
reg_ciii = glm(Default~loan_amnt+annual_inc+term+int_rate+int_rate_2,family = "binomial",data = data)
summary(reg_ciii)
```

```
##
## Call:
## glm(formula = Default ~ loan_amnt + annual_inc + term + int_rate +
```

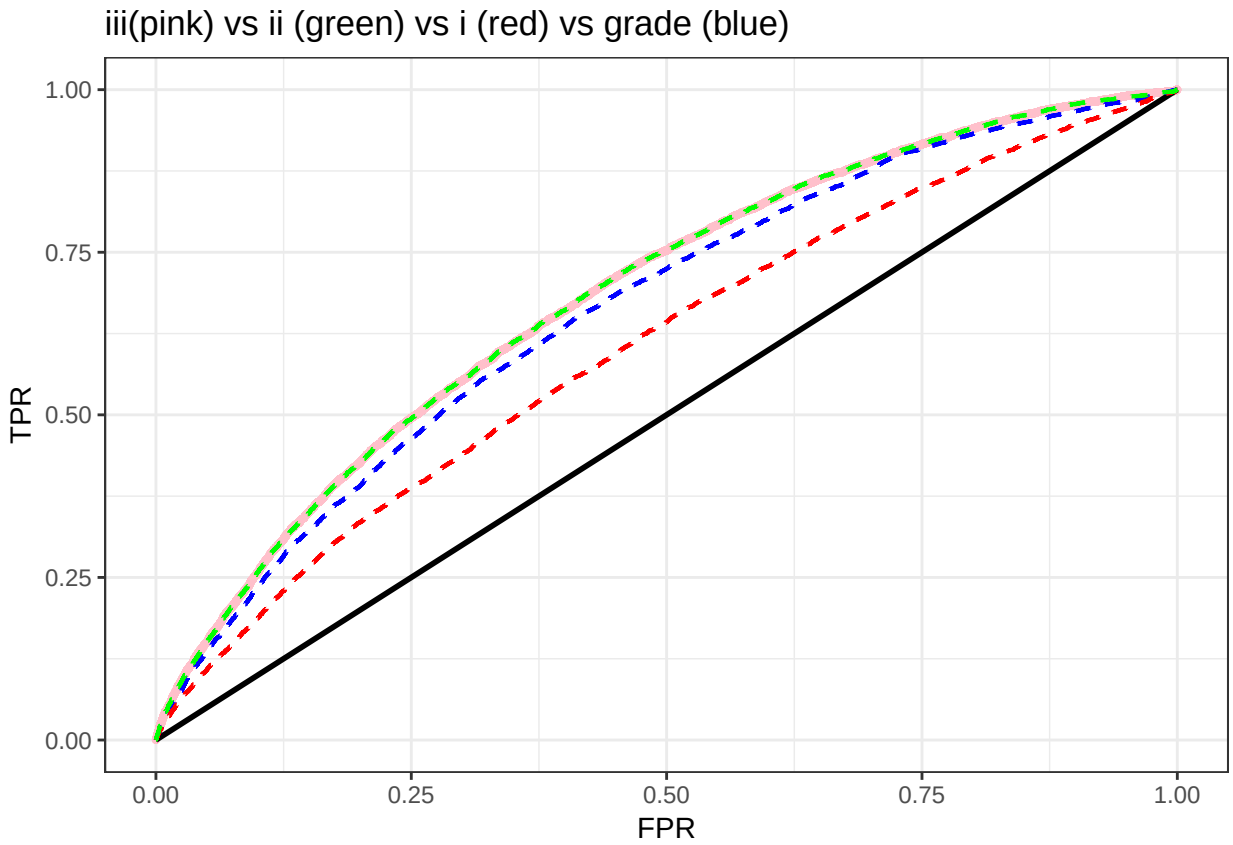
```
##     int_rate_2, family = "binomial", data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0836  -0.5992  -0.4734  -0.3400   4.1124
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -4.035e+00  1.667e-01 -24.201  < 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_rate_2     -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
```

```
#qplot(int_rate, color = I("red"), data=data)
#qplot(int_rate_2,color = I("blue"), data=data)
```

Based on the summary above, the variable of square of interest rate is significant due to its very small p-value. The higher the square of interest rate, the lower the default probability is due to the negative sign of the coefficient of the square of interest rate. The higher the interest rate, the higher the default probability is due to the positive sign of the coefficient of the interest rate. The relationship is that when the interest rate is extremely high, as the interest rate keeps increasing, the interest rate does not have much effect anymore overall becasue the square interest rate is also very high, which cancels some effects of the interest rate; when the interest rate is extremely low, as the interest rate keeps increasing, the interest rate will have more and more effects while the square interest rate does not have much effect because the square of a very small number (smaller than 1) is very small.

```
phat_ciii = predict(reg_ciii,type="response")
phat_ciii = jitter(predict(reg_ciii,type="response"))
glm_simple_roc_ciii <- simple_roc(data$Default=="1", phat_ciii)
TPR_ciii <- glm_simple_roc_ciii$TPR
FPR_ciii <- glm_simple_roc_ciii$FPR

qplot(FPR_ciii,TPR_ciii,xlab="FPR",ylab="TPR",col=I("pink"),
      main="iii(pink) vs ii (green) vs i (red) vs grade (blue)",
      size=I(0.75)) +
  geom_segment(aes(x = 0, xend = 1, y = 0, yend = 1), size=I(1.0)) +
  geom_line(aes(y=TPR),color=I("blue"),size=I(0.75),linetype = 2) +
  geom_line(aes(y=TPR_ci),color=I("red"),size=I(0.75),linetype = 2) +
  geom_line(aes(y=TPR_cii),color=I("green"),size=I(0.75),linetype = 2) +
  theme_bw()
```



Based on the plot, adding the square of the interest rate does not make any difference to the ROC curve.