R Notebook

The following is your first chunk to start with. Remember, you can add chunks using the menu above (Insert -> R) or using the keyboard shortcut Ctrl+Alt+I. A good practice is to use different code chunks to answer different questions. You can delete this comment if you like.

Other useful keyboard shortcuts include Alt- for the assignment operator, and Ctrl+Shift+M for the pipe operator. You can delete these reminders if you don't want them in your report.

```
setwd("C:/") #Don't forget to set your working directory before you start!

library("tidyverse")

## Warning: package 'tidyverse' was built under R version 3.6.2

## -- Attaching packages ---------------------------------------------------
------------------------------------------------------------------------
---- tidyverse 1.3.0 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## Warning: package 'ggplot2' was built under R version 3.6.1

## Warning: package 'tibble' was built under R version 3.6.2

## Warning: package 'tidyr' was built under R version 3.6.2

## Warning: package 'readr' was built under R version 3.6.2

## Warning: package 'purrr' was built under R version 3.6.2

## Warning: package 'dplyr' was built under R version 3.6.1

## Warning: package 'forcats' was built under R version 3.6.2

## -- Conflicts ------------------------------------------------------------
------------------------------------------------------------------------ t
idyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library("tidymodels")

## Warning: package 'tidymodels' was built under R version 3.6.2
```

```
## -- Attaching packages ---------------------------------------------------
-------------------------------------------------------------------------
--- tidymodels 0.0.3 --

## v broom     0.5.4     v recipes   0.1.9
## v dials     0.0.4     v rsample   0.0.5
## v infer     0.5.1     v yardstick 0.0.4
## v parsnip   0.0.5

## Warning: package 'dials' was built under R version 3.6.2

## Warning: package 'infer' was built under R version 3.6.2

## Warning: package 'parsnip' was built under R version 3.6.2

## Warning: package 'recipes' was built under R version 3.6.2

## Warning: package 'rsample' was built under R version 3.6.2

## Warning: package 'yardstick' was built under R version 3.6.2

## -- Conflicts ------------------------------------------------------------
------------------------------------------------------------------------ ti
dymodels_conflicts() --
## x scales::discard() masks purrr::discard()
## x dplyr::filter()   masks stats::filter()
## x recipes::fixed()  masks stringr::fixed()
## x dplyr::lag()      masks stats::lag()
## x dials::margin()   masks ggplot2::margin()
## x yardstick::spec() masks readr::spec()
## x recipes::step()   masks stats::step()

library("plotly")

## Warning: package 'plotly' was built under R version 3.6.2

##
## 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

library("skimr")
```

```
## Warning: package 'skimr' was built under R version 3.6.2
```

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

```
## Warning: package 'caret' was built under R version 3.6.2
```

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

```
##
## Attaching package: 'caret'
```

```
## The following objects are masked from 'package:yardstick':
##
##     precision, recall
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
dff= read_csv("lab3FraminghamHeart.csv")
```

```
## Parsed with column specification:
## cols(
##   gender = col_double(),
##   age = col_double(),
##   education = col_double(),
##   currentSmoker = col_double(),
##   cigsPerDay = col_double(),
##   BPMeds = col_double(),
##   prevalentStroke = col_double(),
##   prevalentHyp = col_double(),
##   diabetes = col_double(),
##   totChol = col_double(),
##   sysBP = col_double(),
##   diaBP = col_double(),
##   BMI = col_double(),
##   heartRate = col_double(),
##   glucose = col_double(),
##   TenYearCHD = col_double()
## )
```

```
colsToFactor <- c('gender', 'education', 'currentSmoker', 'BPMeds', 'prevalen
tStroke', 'prevalentHyp', 'diabetes')
```

```
dff <- dff %>%
  mutate_at(colsToFactor, ~factor(.))
```

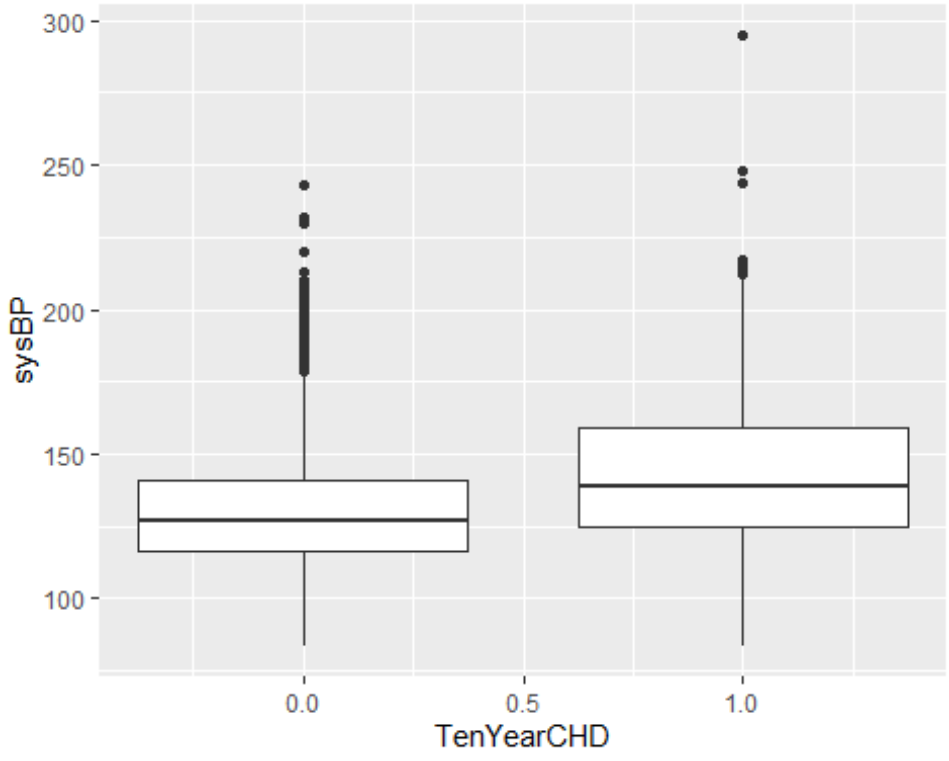
```
str(dff)
```

```
## Classes 'spec_tbl_df', 'tbl_df', 'tbl' and 'data.frame': 3658 obs. of  16
variables:
##  $ gender         : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 1 1 2 2 ...
```

```
##  $ age            : num  39 46 48 61 46 43 63 45 52 43 ...
##  $ education      : Factor w/ 4 levels "1","2","3","4": 4 2 1 3 3 2 1 2 1
1 ...
##  $ currentSmoker  : Factor w/ 2 levels "0","1": 1 1 2 2 2 1 1 2 1 2 ...
##  $ cigsPerDay     : num  0 0 20 30 23 0 0 20 0 30 ...
##  $ BPMeds         : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ prevalentStroke: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ prevalentHyp   : Factor w/ 2 levels "0","1": 1 1 1 2 1 2 1 1 2 2 ...
##  $ diabetes       : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ totChol        : num  195 250 245 225 285 228 205 313 260 225 ...
##  $ sysBP          : num  106 121 128 150 130 ...
##  $ diaBP          : num  70 81 80 95 84 110 71 71 89 107 ...
##  $ BMI            : num  27 28.7 25.3 28.6 23.1 ...
##  $ heartRate      : num  80 95 75 65 85 77 60 79 76 93 ...
##  $ glucose        : num  77 76 70 103 85 99 85 78 79 88 ...
##  $ TenYearCHD     : num  0 0 0 1 0 0 1 0 0 0 ...
```

Question 1) #sysBP Boxplot

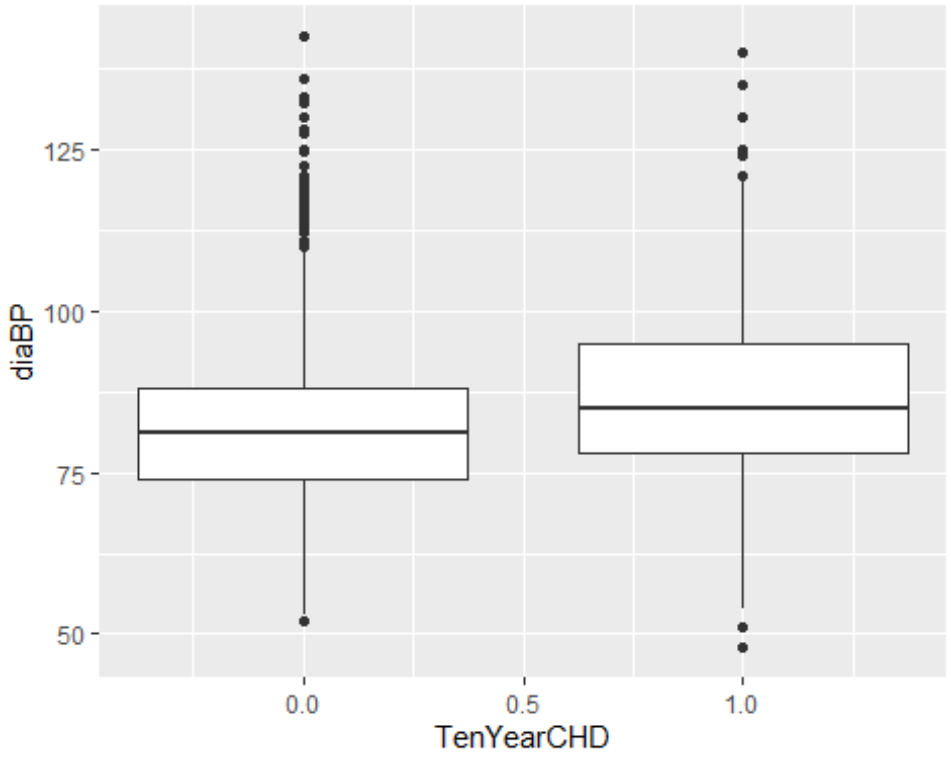
```
plot1 <- dff %>%
  ggplot(aes(x= TenYearCHD, y=sysBP, group= TenYearCHD)) +
  geom_boxplot()

plot1
```



#diaBP Boxplot

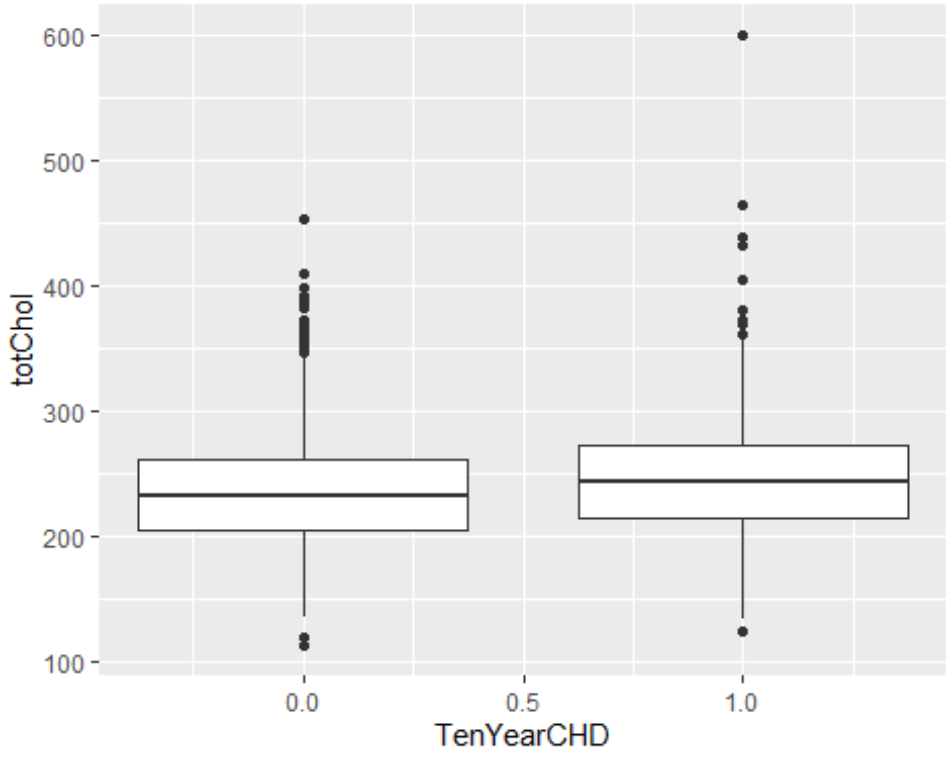
```
plot2 <- dff %>%
  ggplot(aes(x= TenYearCHD, y=diaBP, group= TenYearCHD)) +
```

```
  geom_boxplot()

plot2
```



#totChol Boxplot

```
plot3 <- dff %>%
  ggplot(aes(x= TenYearCHD, y=totChol, group= TenYearCHD)) +
  geom_boxplot()


plot3
```



Question 2) (i)

```
set.seed(123)

dffTrain <- dff %>%  sample_frac(0.7)
dffTest <- dplyr::setdiff(dff,dffTrain)
```

Question 2) (ii)

```
#Gender
dffTrain %>% group_by(gender) %>%
  tally() %>%
  mutate(pct = 100*n/sum(n))
```

```
## # A tibble: 2 x 3
##   gender     n   pct
##   <fct>  <int> <dbl>
## 1 0       1419  55.4
## 2 1       1142  44.6
```

```
#Gender
dffTest %>% group_by(gender) %>%
  tally() %>%
  mutate(pct = 100*n/sum(n))
```

```
## # A tibble: 2 x 3
##   gender     n   pct
##   <fct>  <int> <dbl>
```

```
## 1 0        616  56.2
## 2 1        481  43.8
```

```
#AgeGroup
dffTrain %>% group_by( ageGroup=cut_interval(age, length=10)) %>%
  tally() %>%
  mutate(pct = 100*n/sum(n))
```

```
## # A tibble: 4 x 3
##   ageGroup     n   pct
##   <fct>    <int> <dbl>
## 1 [30,40]    467  18.2
## 2 (40,50]    973  38.0
## 3 (50,60]    772  30.1
## 4 (60,70]    349  13.6
```

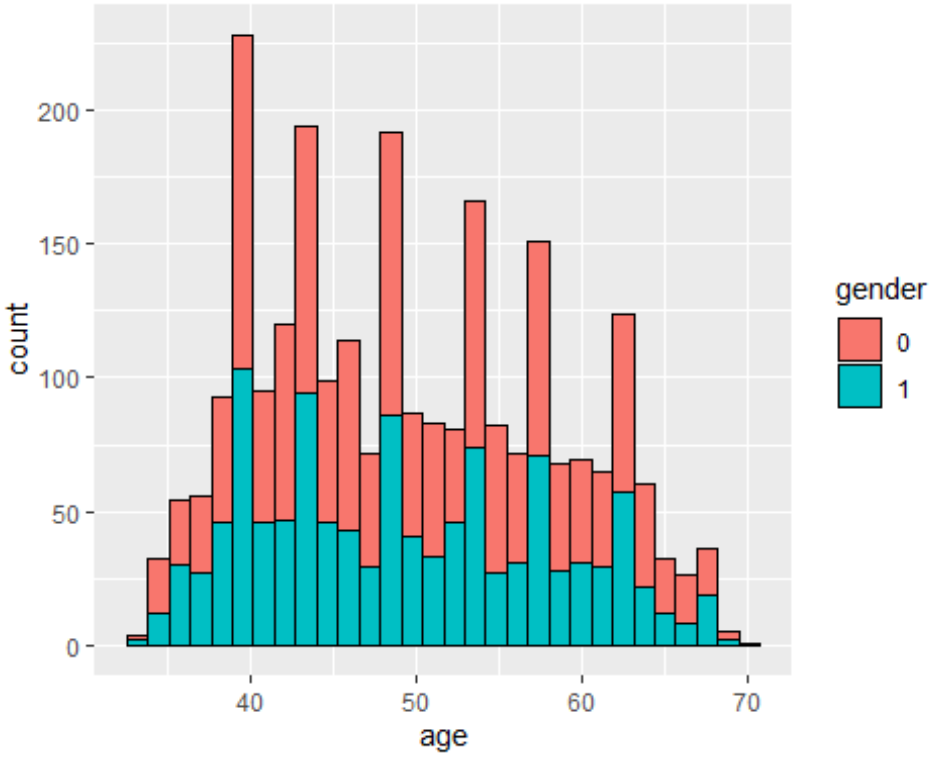
```
#AgeGroup
dffTest %>% group_by( ageGroup=cut_interval(age, length=10)) %>%
  tally() %>%
  mutate(pct = 100*n/sum(n))
```

```
## # A tibble: 4 x 3
##   ageGroup     n   pct
##   <fct>    <int> <dbl>
## 1 [30,40]    181  16.5
## 2 (40,50]    421  38.4
## 3 (50,60]    346  31.5
## 4 (60,70]    149  13.6
```

#Histogram

```
plot4 <- dffTrain %>%
  ggplot(aes(x=age, fill=gender)) +
  geom_histogram(color='black')

plot4
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Question 3)

```
fitLPM <- lm(TenYearCHD ~., data= dffTrain)
summary(fitLPM)

##
## Call:
## lm(formula = TenYearCHD ~ ., data = dffTrain)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.69588 -0.18760 -0.09864 -0.00854  1.06563
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -0.5193243  0.0939086  -5.530 3.53e-08 ***
## gender1           0.0402834  0.0149552   2.694  0.00711 **
## age               0.0073056  0.0009204   7.938 3.06e-15 ***
## education2       -0.0114841  0.0167200  -0.687  0.49224
## education3       -0.0345910  0.0196551  -1.760  0.07854 .
## education4       -0.0259428  0.0230652  -1.125  0.26080
## currentSmoker1    0.0143681  0.0216179   0.665  0.50634
## cigsPerDay        0.0018669  0.0009316   2.004  0.04519 *
## BPMeds1           0.0184297  0.0434995   0.424  0.67184
## prevalentStroke1  0.2099878  0.0983542   2.135  0.03285 *
## prevalentHyp1     0.0448001  0.0208879   2.145  0.03206 *
## diabetes1         0.0204464  0.0513727   0.398  0.69066
```

```
## totChol           0.0002882  0.0001590   1.813  0.07000 .
## sysBP             0.0023876  0.0005798   4.118 3.95e-05 ***
## diaBP            -0.0016597  0.0009716  -1.708  0.08770 .
## BMI               0.0007242  0.0018265   0.397  0.69175
## heartRate        -0.0013046  0.0005843  -2.233  0.02566 *
## glucose           0.0011775  0.0003608   3.264  0.00111 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3388 on 2543 degrees of freedom
## Multiple R-squared:  0.1077, Adjusted R-squared:  0.1017
## F-statistic: 18.05 on 17 and 2543 DF,  p-value: < 2.2e-16
```

By logic: currentSmoker and cigsPerDay are collinear can if cigsPerDay >0 then person is a smoker

By analysis: Using VIF

```
car::vif(fitLPM)
```

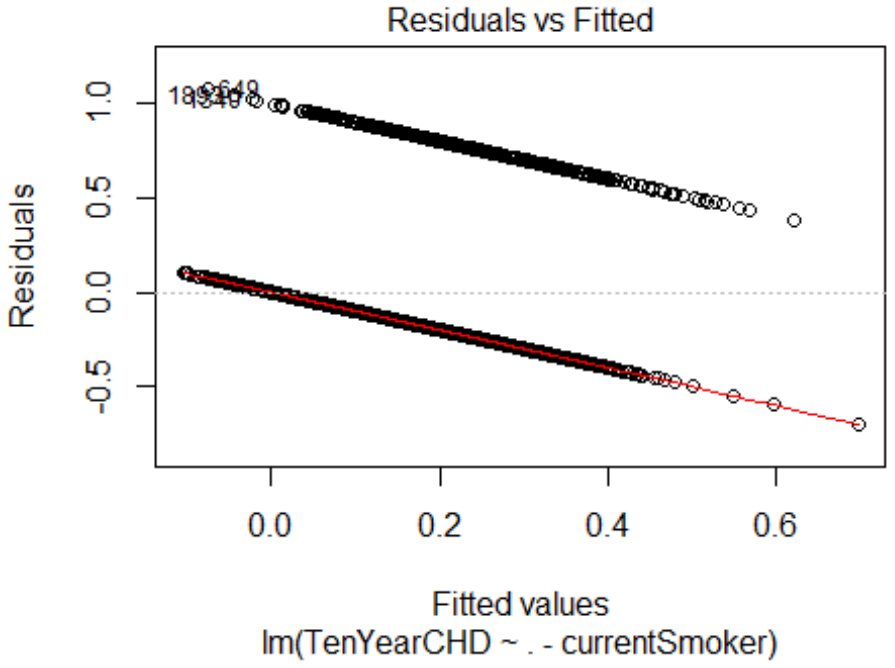
```
## Registered S3 methods overwritten by 'car':
##   method                           from
##   influence.merMod                 lme4
##   cooks.distance.influence.merMod lme4
##   dfbeta.influence.merMod          lme4
##   dfbetas.influence.merMod         lme4

##                      GVIF Df GVIF^(1/(2*Df))
## gender           1.232950  1        1.110383
## age              1.398367  1        1.182526
## education        1.139817  3        1.022051
## currentSmoker    2.604754  1        1.613925
## cigsPerDay       2.762784  1        1.662163
## BPMeds           1.106826  1        1.052058
## prevalentStroke 1.006585  1        1.003287
## prevalentHyp     2.057398  1        1.434363
## diabetes         1.630615  1        1.276956
## totChol          1.106930  1        1.052107
## sysBP            3.777158  1        1.943491
## diaBP            2.997947  1        1.731458
## BMI              1.227604  1        1.107973
## heartRate        1.095878  1        1.046842
## glucose          1.645722  1        1.282857
```

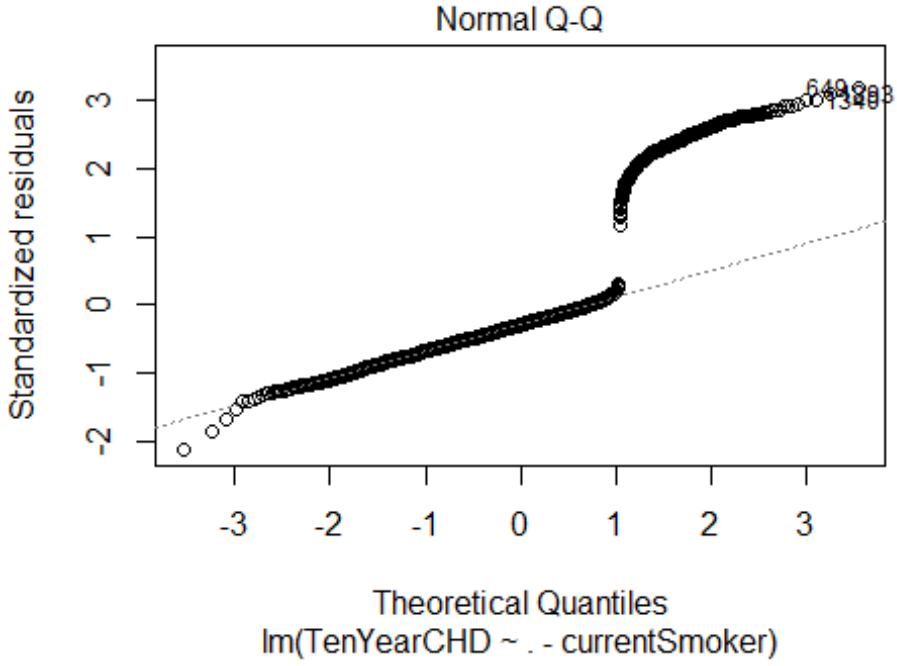
#Correct model will not have both cigsPerDay and currentSmoker variables. # The new model can be made now

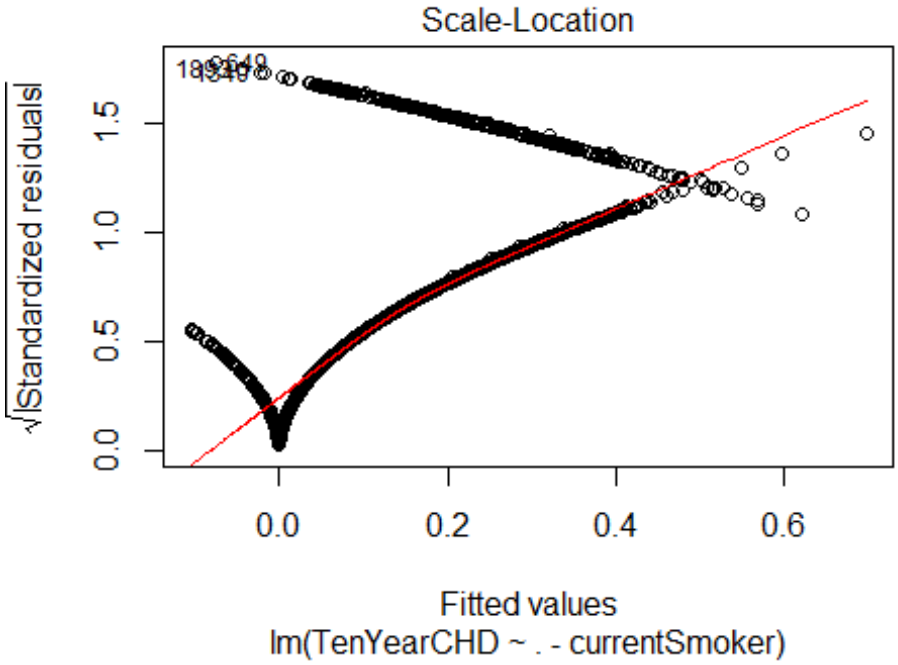
```r
newfitLPM <- lm(TenYearCHD ~. -currentSmoker, data= dffTrain)
summary(newfitLPM)
```

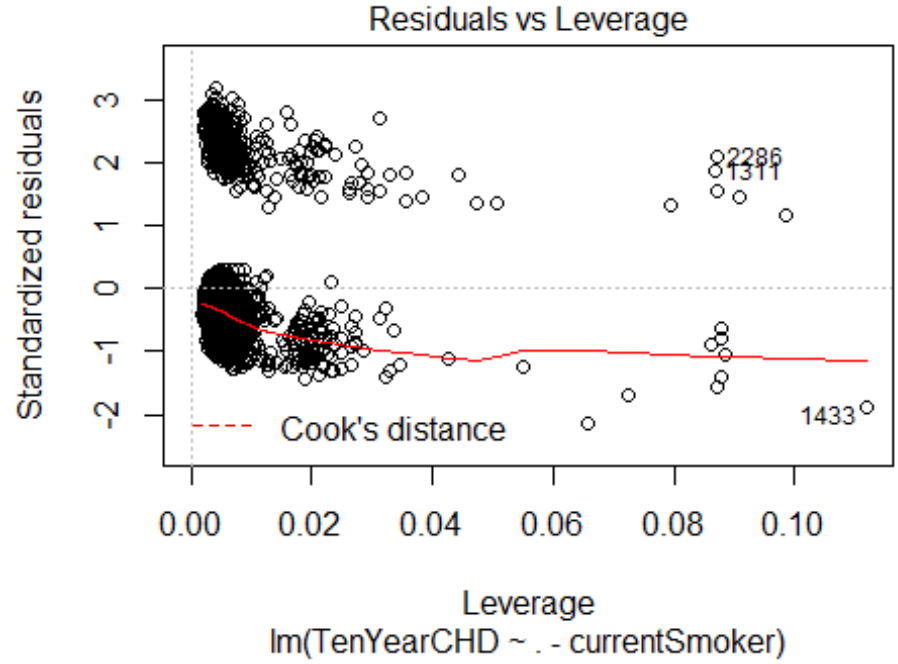
```
##
## Call:
## lm(formula = TenYearCHD ~ . - currentSmoker, data = dffTrain)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.69721 -0.18848 -0.09967 -0.00937  1.07518
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -0.5092583  0.0926691  -5.495 4.28e-08 ***
## gender1           0.0396262  0.0149208   2.656 0.007962 **
## age               0.0072591  0.0009176   7.911 3.78e-15 ***
## education2       -0.0113009  0.0167159  -0.676 0.499067
## education3       -0.0346151  0.0196529  -1.761 0.078304 .
## education4       -0.0260964  0.0230615  -1.132 0.257909
## cigsPerDay        0.0023323  0.0006145   3.795 0.000151 ***
## BPMeds1           0.0185984  0.0434940   0.428 0.668972
## prevalentStroke1  0.2097097  0.0983425   2.132 0.033066 *
## prevalentHyp1     0.0448426  0.0208855   2.147 0.031882 *
## diabetes1         0.0203925  0.0513670   0.397 0.691403
## totChol           0.0002875  0.0001590   1.809 0.070633 .
## sysBP             0.0023882  0.0005798   4.119 3.92e-05 ***
## diaBP            -0.0016833  0.0009708  -1.734 0.083051 .
## BMI               0.0006191  0.0018194   0.340 0.733670
## heartRate        -0.0013019  0.0005843  -2.228 0.025944 *
## glucose           0.0011752  0.0003607   3.258 0.001138 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3388 on 2544 degrees of freedom
## Multiple R-squared:  0.1075, Adjusted R-squared:  0.1019
## F-statistic: 19.16 on 16 and 2544 DF,  p-value: < 2.2e-16
```

```r
plot(newfitLPM)
```


Residuals vs Fitted
Residuals
1.0
0.5
0.0
-0.5
0.0
0.2
0.4
0.6
Fitted values
lm(TenYearCHD ~ . - currentSmoker)


Normal Q-Q
Standardized residuals
3
2
1
0
-1
-2
-3
-2
-1
0
1
2
3
Theoretical Quantiles
lm(TenYearCHD ~ . - currentSmoker)





Question 4)

```
resultsLPM <-
    lm( TenYearCHD ~. -currentSmoker, data= dffTrain ) %>%
```

```
    predict(., dffTest) %>%
    bind_cols(dffTest, predictedProb=.) %>%
    mutate(predictedClass = ifelse(predictedProb > 0.5, 1, 0))
resultsLPM
```

```
## # A tibble: 1,097 x 18
##    gender   age education currentSmoker cigsPerDay BPMeds prevalentStroke
##    <fct>  <dbl> <fct>     <fct>              <dbl> <fct>  <fct>
##  1 1         48 1         1                     20 0      0
##  2 0         43 2         0                      0 0      0
##  3 0         43 2         0                      0 0      0
##  4 0         41 3         0                      0 1      0
##  5 0         52 3         1                     20 0      0
##  6 0         61 3         0                      0 0      0
##  7 1         46 1         1                     20 0      0
##  8 0         63 2         1                     40 0      0
##  9 0         62 1         0                      0 0      0
## 10 1         49 1         1                      2 0      0
## # ... with 1,087 more rows, and 11 more variables: prevalentHyp <fct>,
## #   diabetes <fct>, totChol <dbl>, sysBP <dbl>, diaBP <dbl>, BMI <dbl>,
## #   heartRate <dbl>, glucose <dbl>, TenYearCHD <dbl>, predictedProb <dbl>,
## #   predictedClass <dbl>
```

```
#TenYearCHD in Test data
dffTest %>% group_by(TenYearCHD ) %>%
  tally() %>%
  mutate(pct = 100*n/sum(n))
```

```
## # A tibble: 2 x 3
##   TenYearCHD     n   pct
##        <dbl> <int> <dbl>
## 1          0   925  84.3
## 2          1   172  15.7
```

```
#TenYearCHD in resultsLPM
resultsLPM %>% group_by(predictedClass ) %>%
  tally() %>%
  mutate(pct = 100*n/sum(n))
```

```
## # A tibble: 2 x 3
##   predictedClass     n    pct
##            <dbl> <int>  <dbl>
## 1              0  1087 99.1
## 2              1    10  0.912
```

```
#Factoring TenYearCHD in training and test datasets

colsToFactor <- c('TenYearCHD')

dffTrain  <- dffTrain  %>%
```

```
  mutate_at(colsToFactor, ~factor(.))
dffTrain
```

```
## # A tibble: 2,561 x 16
##    gender   age education currentSmoker cigsPerDay BPMeds prevalentStroke
##    <fct>  <dbl> <fct>     <fct>              <dbl> <fct>  <fct>
##  1 0         63 3         0                      0 0      0
##  2 1         43 4         1                     25 0      0
##  3 1         53 4         0                      0 0      0
##  4 0         64 2         1                      9 0      0
##  5 0         57 2         0                      0 0      0
##  6 1         40 4         1                     25 0      0
##  7 0         55 2         0                      0 0      0
##  8 0         57 2         0                      0 0      0
##  9 1         62 1         1                     30 0      0
## 10 0         60 1         0                      0 0      0
## # ... with 2,551 more rows, and 9 more variables: prevalentHyp <fct>,
## #   diabetes <fct>, totChol <dbl>, sysBP <dbl>, diaBP <dbl>, BMI <dbl>,
## #   heartRate <dbl>, glucose <dbl>, TenYearCHD <fct>
```

```
dffTest   <- dffTest   %>%
  mutate_at(colsToFactor, ~factor(.))
dffTest
```

```
## # A tibble: 1,097 x 16
##    gender   age education currentSmoker cigsPerDay BPMeds prevalentStroke
##    <fct>  <dbl> <fct>     <fct>              <dbl> <fct>  <fct>
##  1 1         48 1         1                     20 0      0
##  2 0         43 2         0                      0 0      0
##  3 0         43 2         0                      0 0      0
##  4 0         41 3         0                      0 1      0
##  5 0         52 3         1                     20 0      0
##  6 0         61 3         0                      0 0      0
##  7 1         46 1         1                     20 0      0
##  8 0         63 2         1                     40 0      0
##  9 0         62 1         0                      0 0      0
## 10 1         49 1         1                      2 0      0
## # ... with 1,087 more rows, and 9 more variables: prevalentHyp <fct>,
## #   diabetes <fct>, totChol <dbl>, sysBP <dbl>, diaBP <dbl>, BMI <dbl>,
## #   heartRate <dbl>, glucose <dbl>, TenYearCHD <fct>
```

Question 5)

```
fitGLM <- glm(TenYearCHD ~. -currentSmoker, family = binomial(), data= dffTra
in)
summary(fitGLM)
```

```
##
## Call:
## glm(formula = TenYearCHD ~ . - currentSmoker, family = binomial(),
##     data = dffTrain)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8022  -0.5882  -0.4071  -0.2738   2.8363
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -7.927497   0.846875  -9.361  < 2e-16 ***
## gender1           0.422202   0.133313   3.167 0.001540 **
## age               0.066797   0.008110   8.237  < 2e-16 ***
## education2       -0.079672   0.146967  -0.542 0.587743
## education3       -0.329631   0.183167  -1.800 0.071921 .
## education4       -0.236143   0.213615  -1.105 0.268960
## cigsPerDay        0.020000   0.005146   3.886 0.000102 ***
## BPMeds1          -0.002423   0.294477  -0.008 0.993434
## prevalentStroke1  1.152421   0.659094   1.748 0.080379 .
## prevalentHyp1     0.338398   0.166699   2.030 0.042358 *
## diabetes1        -0.005002   0.374594  -0.013 0.989345
## totChol           0.003606   0.001338   2.696 0.007017 **
## sysBP             0.014442   0.004495   3.213 0.001315 **
## diaBP            -0.007077   0.007813  -0.906 0.365014
## BMI               0.011682   0.015070   0.775 0.438211
## heartRate        -0.011470   0.005157  -2.224 0.026137 *
## glucose           0.007397   0.002634   2.808 0.004983 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2168.1  on 2560  degrees of freedom
## Residual deviance: 1894.3  on 2544  degrees of freedom
## AIC: 1928.3
##
## Number of Fisher Scoring iterations: 5
```

```
exp(coef(fitGLM))
```

```
##      (Intercept)          gender1              age       education2
##     0.0003606879     1.5253171095     1.0690784440     0.9234189417
##       education3       education4       cigsPerDay          BPMeds1
##     0.7191887265     0.7896676736     1.0202012574     0.9975796686
## prevalentStroke1    prevalentHyp1        diabetes1          totChol
##     3.1658488040     1.4026980839     0.9950101842     1.0036127972
##            sysBP            diaBP              BMI        heartRate
##     1.0145465769     0.9929479273     1.0117507851     0.9885958031
##          glucose
##     1.0074239785
```

Question 6)

```r
#predictedClass will need to be defined as a factor

resultsLog  <-
    glm(TenYearCHD ~. -currentSmoker, family = binomial(), data= dffTrain ) %
>%
    predict(dffTest, type= 'response') %>%
    bind_cols(dffTest, predictedProb=.) %>%
    mutate(predictedClass = as.factor(ifelse(predictedProb > 0.5, 1, 0)))
resultsLog
```

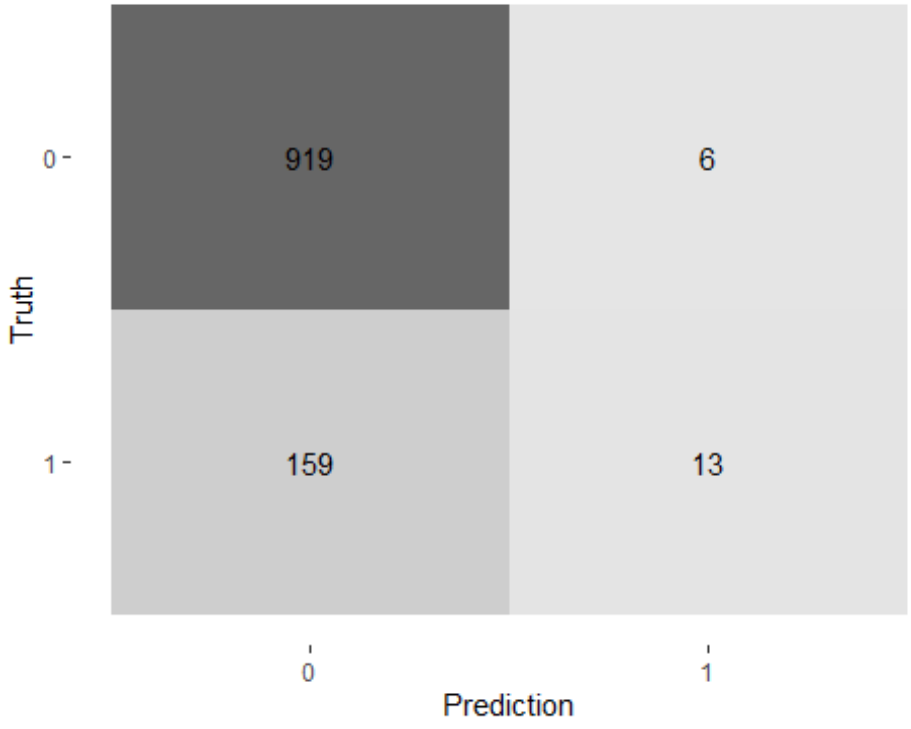
```
## # A tibble: 1,097 x 18
##    gender   age education currentSmoker cigsPerDay BPMeds prevalentStroke
##    <fct>  <dbl> <fct>     <fct>               <dbl> <fct>  <fct>
##  1 1         48 1         1                      20 0      0
##  2 0         43 2         0                       0 0      0
##  3 0         43 2         0                       0 0      0
##  4 0         41 3         0                       0 1      0
##  5 0         52 3         1                      20 0      0
##  6 0         61 3         0                       0 0      0
##  7 1         46 1         1                      20 0      0
##  8 0         63 2         1                      40 0      0
##  9 0         62 1         0                       0 0      0
## 10 1         49 1         1                       2 0      0
## # ... with 1,087 more rows, and 11 more variables: prevalentHyp <fct>,
## #   diabetes <fct>, totChol <dbl>, sysBP <dbl>, diaBP <dbl>, BMI <dbl>,
## #   heartRate <dbl>, glucose <dbl>, TenYearCHD <fct>, predictedProb <dbl>,
## #   predictedClass <fct>
```

```r
resultsLog %>% group_by(predictedClass ) %>%
  tally() %>%
  mutate(pct = 100*n/sum(n))
```

```
## # A tibble: 2 x 3
##   predictedClass     n   pct
##   <fct>          <int> <dbl>
## 1 0               1078 98.3
## 2 1                 19  1.73
```

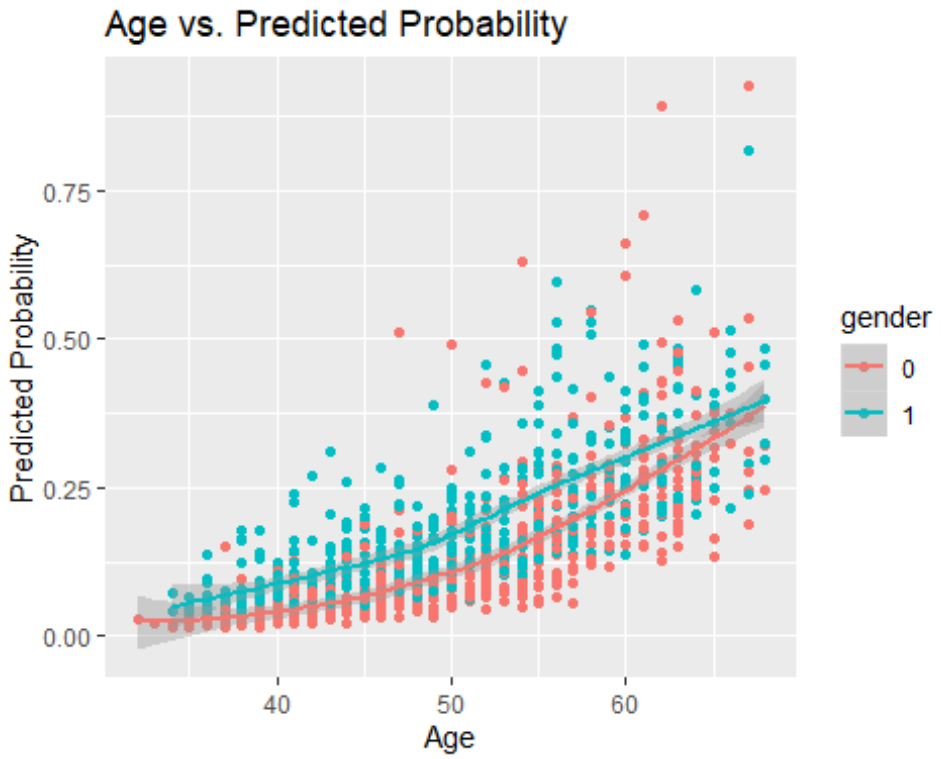
Question 7)

```r
resultsLog %>%
  conf_mat(estimate = predictedClass, truth =TenYearCHD) %>%
  autoplot(type = 'heatmap')
```

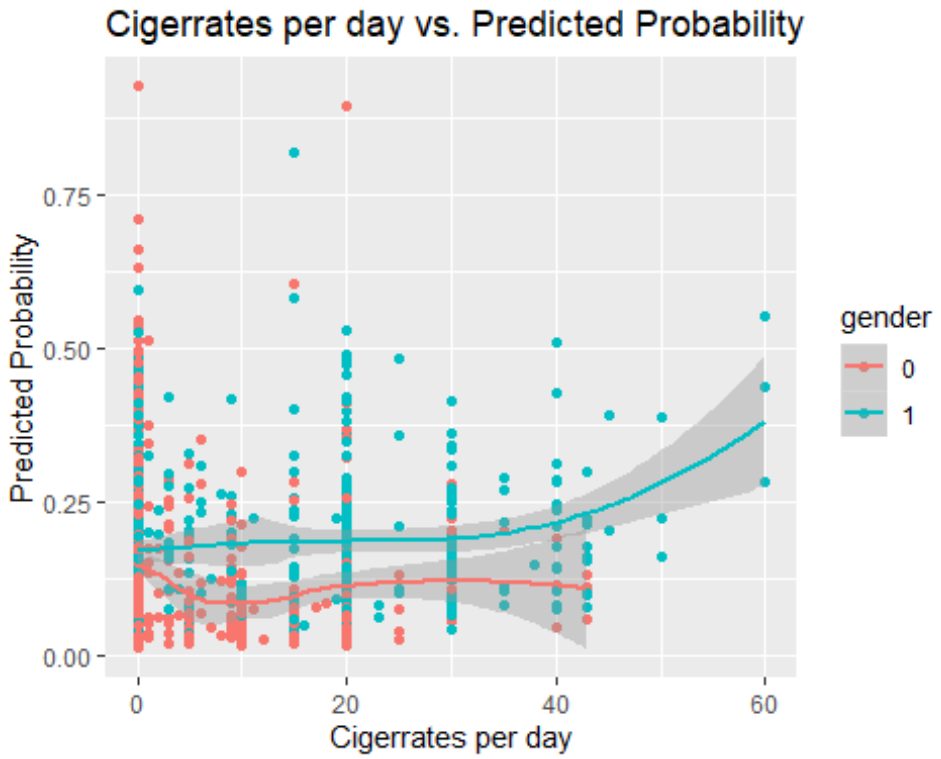


Question 8)

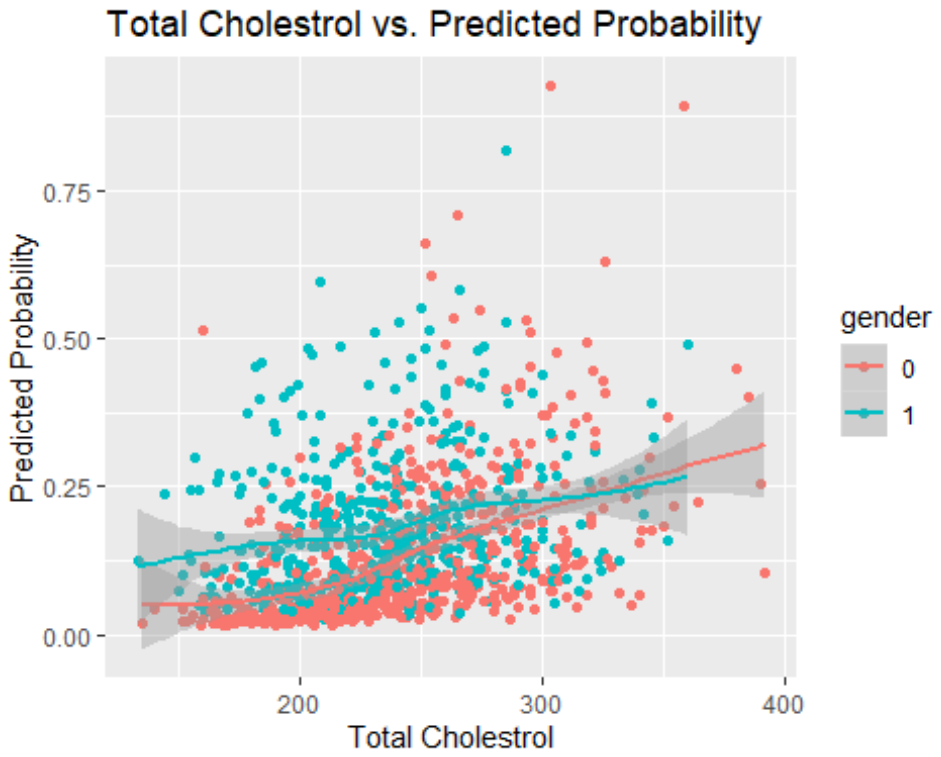
```
#Age vs predictedClass
plot5 <- resultsLog %>%
  ggplot(aes(x= age, y=predictedProb, color=gender)) +
  geom_point() +
  geom_smooth()+
  labs(title= "Age vs. Predicted Probability", x= "Age", y= "Predicted Probab
ility")

plot5

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



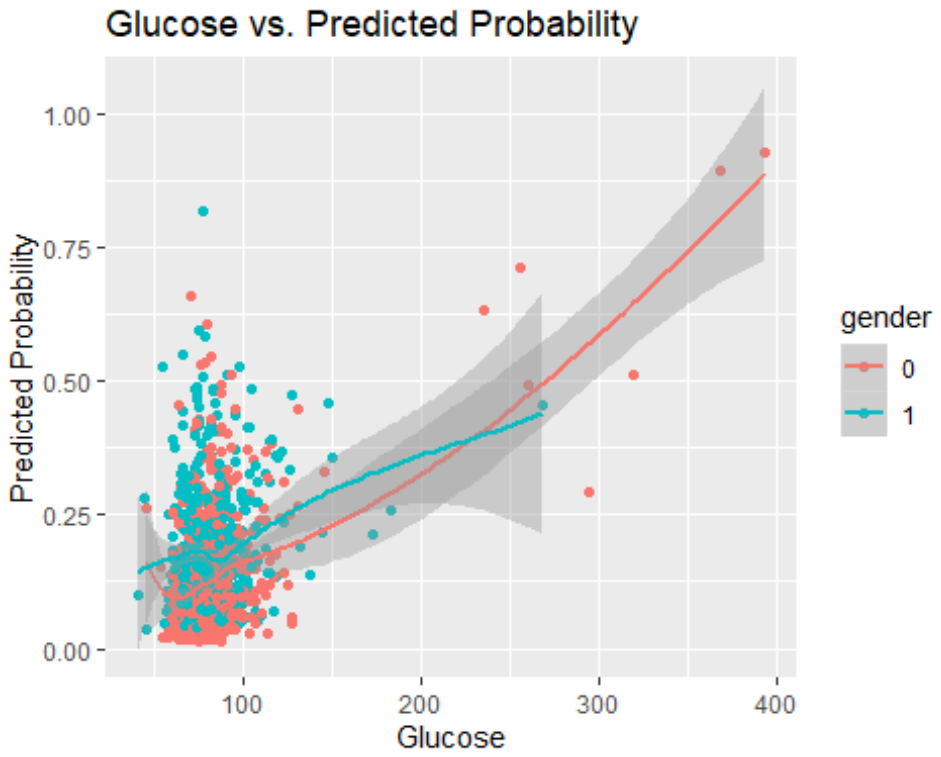

```
#cigsPerDay vs predictedClass
plot6 <- resultsLog %>%
  ggplot(aes(x= cigsPerDay, y=predictedProb, color=gender)) +
  geom_point() +
  geom_smooth() +
  labs(title= "Cigerrates per day vs. Predicted Probability", x= "Cigerrates
per day", y= "Predicted Probability")

plot6

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```




```
#totChol vs predictedClass
plot7 <- resultsLog %>%
  ggplot(aes(x= totChol, y=predictedProb, color=gender)) +
  geom_point() +
  geom_smooth() +
  labs(title= "Total Cholestrol vs. Predicted Probability", x= "Total Cholest
rol", y= "Predicted Probability")

plot7

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```




```
#glucose vs predictedClass
plot8 <- resultsLog %>%
  ggplot(aes(x= glucose, y=predictedProb,color=gender)) +
  geom_point() +
  geom_smooth() +
  labs(title= "Glucose vs. Predicted Probability", x= "Glucose", y= "Predicte
d Probability")

plot8

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Question 9)

```
library(e1071)

## Warning: package 'e1071' was built under R version 3.6.2

resultsLogCaret  <-
    train(TenYearCHD ~. -currentSmoker, family = 'binomial', data= dffTrain,
method= 'glm' ) %>%
    predict(dffTest, type= 'raw') %>%
    bind_cols(dffTest, predictedClass=.)

resultsLogCaret

## # A tibble: 1,097 x 17
##    gender   age education currentSmoker cigsPerDay BPMeds prevalentStroke
##    <fct>  <dbl> <fct>     <fct>               <dbl> <fct>  <fct>
##  1 1         48 1         1                      20 0      0
##  2 0         43 2         0                       0 0      0
##  3 0         43 2         0                       0 0      0
##  4 0         41 3         0                       0 1      0
##  5 0         52 3         1                      20 0      0
##  6 0         61 3         0                       0 0      0
##  7 1         46 1         1                      20 0      0
##  8 0         63 2         1                      40 0      0
##  9 0         62 1         0                       0 0      0
## 10 1         49 1         1                       2 0      0
```

```
## # ... with 1,087 more rows, and 10 more variables: prevalentHyp <fct>,
## #   diabetes <fct>, totChol <dbl>, sysBP <dbl>, diaBP <dbl>, BMI <dbl>,
## #   heartRate <dbl>, glucose <dbl>, TenYearCHD <fct>, predictedClass <fct>
```

```
resultsLogCaret %>%
  xtabs(~predictedClass+TenYearCHD, .) %>%
  confusionMatrix(positive = '1')
```

```
## Confusion Matrix and Statistics
##
##               TenYearCHD
## predictedClass   0   1
##              0 919 159
##              1   6  13
##
##                Accuracy : 0.8496
##                  95% CI : (0.827, 0.8702)
##     No Information Rate : 0.8432
##     P-Value [Acc > NIR] : 0.297
##
##                   Kappa : 0.1083
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.07558
##             Specificity : 0.99351
##          Pos Pred Value : 0.68421
##          Neg Pred Value : 0.85250
##              Prevalence : 0.15679
##          Detection Rate : 0.01185
##    Detection Prevalence : 0.01732
##       Balanced Accuracy : 0.53455
##
##        'Positive' Class : 1
##
```

Question 10)

```
dff1= read_csv("lab3BancoPortugal.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   job = col_character(),
##   marital = col_character(),
##   education = col_character(),
##   default = col_character(),
##   housing = col_character(),
##   loan = col_character(),
##   contact = col_character(),
##   month = col_character(),
```

```
##   day_of_week = col_character(),
##   poutcome = col_character(),
##   agegroup = col_character()
## )

## See spec(...) for full column specifications.

str(dff1)

## Classes 'spec_tbl_df', 'tbl_df', 'tbl' and 'data.frame': 30488 obs. of  23
variables:
##  $ age           : num  56 37 40 56 59 24 25 25 29 57 ...
##  $ job           : chr  "housemaid" "services" "admin." "services" ...
##  $ marital       : chr  "married" "married" "married" "married" ...
##  $ education     : chr  "basic.4y" "high.school" "basic.6y" "high.school"
...
##  $ default       : chr  "no" "no" "no" "no" ...
##  $ housing       : chr  "no" "yes" "no" "no" ...
##  $ loan          : chr  "no" "no" "no" "yes" ...
##  $ contact       : chr  "telephone" "telephone" "telephone" "telephone" ..
.
##  $ month         : chr  "may" "may" "may" "may" ...
##  $ day_of_week   : chr  "mon" "mon" "mon" "mon" ...
##  $ duration      : num  261 226 151 307 139 380 50 222 137 293 ...
##  $ campaign      : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ pdays         : num  999 999 999 999 999 999 999 999 999 999 ...
##  $ previous      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ poutcome      : chr  "nonexistent" "nonexistent" "nonexistent" "nonexis
tent" ...
##  $ emp.var.rate  : num  1.1 1.1 1.1 1.1 1.1 1.1 1.1 1.1 1.1 1.1 ...
##  $ cons.price.idx: num  94 94 94 94 94 ...
##  $ cons.conf.idx : num  -36.4 -36.4 -36.4 -36.4 -36.4 -36.4 -36.4 -36.4 -3
6.4 -36.4 ...
##  $ euribor3m     : num  4.86 4.86 4.86 4.86 4.86 ...
##  $ nr.employed   : num  5191 5191 5191 5191 5191 ...
##  $ openedAccount : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ agegroup      : chr  "Adults" "Adults" "Adults" "Adults" ...
##  $ newcustomer   : num  1 1 1 1 1 1 1 1 1 1 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   age = col_double(),
##   ..   job = col_character(),
##   ..   marital = col_character(),
##   ..   education = col_character(),
##   ..   default = col_character(),
##   ..   housing = col_character(),
##   ..   loan = col_character(),
##   ..   contact = col_character(),
##   ..   month = col_character(),
##   ..   day_of_week = col_character(),
```

```
##   ..   duration = col_double(),
##   ..   campaign = col_double(),
##   ..   pdays = col_double(),
##   ..   previous = col_double(),
##   ..   poutcome = col_character(),
##   ..   emp.var.rate = col_double(),
##   ..   cons.price.idx = col_double(),
##   ..   cons.conf.idx = col_double(),
##   ..   euribor3m = col_double(),
##   ..   nr.employed = col_double(),
##   ..   openedAccount = col_double(),
##   ..   agegroup = col_character(),
##   ..   newcustomer = col_double()
##   .. )

#Converting categorical variables to Factors
colsToFactor <- c('openedAccount', 'newcustomer', 'agegroup', 'job', 'marital
', 'education', 'default', 'housing', 'loan', 'contact', 'month', 'day_of_wee
k', 'poutcome')
dff1 <- dff1 %>%
  mutate_at(colsToFactor, ~factor(.))
str(dff1)

## Classes 'spec_tbl_df', 'tbl_df', 'tbl' and 'data.frame': 30488 obs. of  23
variables:
##  $ age           : num  56 37 40 56 59 24 25 25 29 57 ...
##  $ job           : Factor w/ 11 levels "admin.","blue-collar",..: 4 8 1 8
1 10 8 8 2 4 ...
##  $ marital       : Factor w/ 3 levels "divorced","married",..: 2 2 2 2 2 3
3 3 3 1 ...
##  $ education     : Factor w/ 7 levels "basic.4y","basic.6y",..: 1 4 2 4 6
6 4 4 4 1 ...
##  $ default       : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ housing       : Factor w/ 2 levels "no","yes": 1 2 1 1 1 2 2 2 1 2 ...
##  $ loan          : Factor w/ 2 levels "no","yes": 1 1 1 2 1 1 1 1 2 1 ...
##  $ contact       : Factor w/ 2 levels "cellular","telephone": 2 2 2 2 2 2
2 2 2 2 ...
##  $ month         : Factor w/ 10 levels "apr","aug","dec",..: 7 7 7 7 7 7 7
7 7 7 ...
##  $ day_of_week   : Factor w/ 5 levels "fri","mon","thu",..: 2 2 2 2 2 2 2
2 2 2 ...
##  $ duration      : num  261 226 151 307 139 380 50 222 137 293 ...
##  $ campaign      : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ pdays         : num  999 999 999 999 999 999 999 999 999 999 ...
##  $ previous      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ poutcome      : Factor w/ 3 levels "failure","nonexistent",..: 2 2 2 2
2 2 2 2 2 2 ...
##  $ emp.var.rate  : num  1.1 1.1 1.1 1.1 1.1 1.1 1.1 1.1 1.1 1.1 ...
##  $ cons.price.idx: num  94 94 94 94 94 ...
##  $ cons.conf.idx : num  -36.4 -36.4 -36.4 -36.4 -36.4 -36.4 -36.4 -36.4 -3
```

```
6.4 -36.4 ...
##  $ euribor3m     : num  4.86 4.86 4.86 4.86 4.86 ...
##  $ nr.employed   : num  5191 5191 5191 5191 5191 ...
##  $ openedAccount : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ agegroup      : Factor w/ 4 levels "Adults","Senior Citizens",..: 1 1 1
1 1 4 4 4 4 1 ...
##  $ newcustomer   : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...

#Splitting into train and test datasets
set.seed(123)

dff1Train <- dff1 %>%  sample_frac(0.7)
dff1Test <- dplyr::setdiff(dff1,dff1Train)

# Model 1: glm model
#Using all variables except duration

bancoDflog <- glm(openedAccount~. -(duration),family='binomial',data=dff1Trai
n)
summary(bancoDflog)

##
## Call:
## glm(formula = openedAccount ~ . - (duration), family = "binomial",
##     data = dff1Train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0409  -0.4175  -0.3284  -0.2630   2.8873
##
## Coefficients: (1 not defined because of singularities)
##                                Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  -2.423e+02  4.388e+01  -5.522 3.36e-08 ***
## age                           6.870e-03  4.383e-03   1.568 0.116983
## jobblue-collar               -9.972e-02  9.473e-02  -1.053 0.292485
## jobentrepreneur              -8.855e-02  1.456e-01  -0.608 0.543120
## jobhousemaid                 -7.142e-02  1.794e-01  -0.398 0.690467
## jobmanagement                 7.193e-02  9.618e-02   0.748 0.454509
## jobretired                    1.003e-01  1.478e-01   0.679 0.497300
## jobself-employed             -5.474e-04  1.316e-01  -0.004 0.996680
## jobservices                  -1.197e-01  1.008e-01  -1.188 0.234910
## jobstudent                    3.339e-01  1.372e-01   2.435 0.014910 *
## jobtechnician                 4.060e-02  8.035e-02   0.505 0.613400
## jobunemployed                 2.781e-02  1.447e-01   0.192 0.847575
## maritalmarried                1.481e-04  7.979e-02   0.002 0.998519
## maritalsingle                 2.612e-02  9.055e-02   0.288 0.773022
## educationbasic.6y             2.198e-02  1.493e-01   0.147 0.882976
## educationbasic.9y            -1.291e-01  1.157e-01  -1.117 0.264160
## educationhigh.school          2.498e-02  1.103e-01   0.226 0.820820
## educationilliterate           1.084e+00  8.651e-01   1.253 0.210085
```

```
## educationprofessional.course  7.099e-02  1.196e-01   0.594 0.552650
## educationuniversity.degree    1.377e-01  1.105e-01   1.246 0.212633
## defaultyes                   -7.749e+00  1.195e+02  -0.065 0.948286
## housingyes                   -2.869e-02  4.737e-02  -0.606 0.544752
## loanyes                      -2.648e-02  6.501e-02  -0.407 0.683772
## contacttelephone             -7.416e-01  8.654e-02  -8.570  < 2e-16 ***
## monthaug                      3.617e-01  1.394e-01   2.595 0.009462 **
## monthdec                      4.742e-01  2.500e-01   1.896 0.057899 .
## monthjul                     -2.649e-02  1.108e-01  -0.239 0.811005
## monthjun                     -7.482e-01  1.422e-01  -5.260 1.44e-07 ***
## monthmar                      1.371e+00  1.658e-01   8.273  < 2e-16 ***
## monthmay                     -4.613e-01  9.372e-02  -4.922 8.57e-07 ***
## monthnov                     -4.741e-01  1.396e-01  -3.397 0.000681 ***
## monthoct                      8.348e-02  1.797e-01   0.464 0.642328
## monthsep                      3.100e-01  2.103e-01   1.474 0.140585
## day_of_weekmon               -1.545e-01  7.805e-02  -1.979 0.047761 *
## day_of_weekthu                1.137e-01  7.508e-02   1.514 0.130051
## day_of_weektue                2.111e-01  7.697e-02   2.743 0.006088 **
## day_of_weekwed                3.045e-01  7.598e-02   4.007 6.15e-05 ***
## campaign                     -4.514e-02  1.265e-02  -3.569 0.000359 ***
## pdays                        -9.467e-04  2.668e-04  -3.548 0.000388 ***
## previous                     -7.466e-02  7.224e-02  -1.033 0.301374
## poutcomenonexistent           4.376e-01  1.119e-01   3.912 9.14e-05 ***
## poutcomesuccess               8.064e-01  2.610e-01   3.090 0.002000 **
## emp.var.rate                 -1.449e+00  1.586e-01  -9.138  < 2e-16 ***
## cons.price.idx                2.126e+00  2.871e-01   7.407 1.30e-13 ***
## cons.conf.idx                 2.938e-02  9.061e-03   3.242 0.001186 **
## euribor3m                     9.927e-02  1.555e-01   0.638 0.523251
## nr.employed                   8.211e-03  3.618e-03   2.269 0.023252 *
## agegroupSenior Citizens       8.333e-02  1.569e-01   0.531 0.595384
## agegroupTeenagers             9.259e-01  4.837e-01   1.914 0.055618 .
## agegroupYoung Adults          2.271e-01  7.896e-02   2.877 0.004018 **
## newcustomer1                         NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 16149  on 21341  degrees of freedom
## Residual deviance: 12642  on 21292  degrees of freedom
## AIC: 12742
##
## Number of Fisher Scoring iterations: 9

#Model 1: Caret
#Using all variables except duration

bancoDflogCaret  <-
    train(openedAccount ~. -duration, family = 'binomial', data= dff1Train, m
ethod= 'glm' ) %>%
```

```r
    predict(dff1Test, type= 'raw') %>%
    bind_cols(dff1Test, predictedClass=.)
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
```

```
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading
```

```
bancoDflogCaret %>%
  xtabs(~predictedClass+openedAccount, .) %>%
  confusionMatrix(positive = '1')
```

```
## Confusion Matrix and Statistics
##
##               openedAccount
## predictedClass    0    1
##              0 7833  871
##              1  136  302
##
##                Accuracy : 0.8898
##                  95% CI : (0.8833, 0.8962)
##     No Information Rate : 0.8717
##     P-Value [Acc > NIR] : 6.372e-08
##
##                   Kappa : 0.328
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.25746
##             Specificity : 0.98293
##          Pos Pred Value : 0.68950
##          Neg Pred Value : 0.89993
##              Prevalence : 0.12831
##          Detection Rate : 0.03303
##    Detection Prevalence : 0.04791
##       Balanced Accuracy : 0.62020
##
##        'Positive' Class : 1
##
```

```
# Model 2: caret model
#Applying domain knowledge and statistical analysis
bancoDflogCaret1  <-
    train(openedAccount ~. -(duration + marital + euribor3m + newcustomer + c
ontact+ education + loan + day_of_week), family = 'binomial', data= dff1Train
, method= 'glm' ) %>%
    predict(dff1Test, type= 'raw') %>%
    bind_cols(dff1Test, predictedClass=.)
```

```
## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type
== :
## prediction from a rank-deficient fit may be misleading
```

```
bancoDflogCaret1 %>%
  xtabs(~predictedClass+openedAccount, .) %>%
  confusionMatrix(positive = '1')

## Confusion Matrix and Statistics
##
##               openedAccount
## predictedClass    0    1
##              0 7837  891
##              1  132  282
##
##                Accuracy : 0.8881
##                  95% CI : (0.8815, 0.8945)
##     No Information Rate : 0.8717
##     P-Value [Acc > NIR] : 9.625e-07
##
##                   Kappa : 0.3091
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.24041
##             Specificity : 0.98344
##          Pos Pred Value : 0.68116
##          Neg Pred Value : 0.89791
##              Prevalence : 0.12831
##          Detection Rate : 0.03085
##    Detection Prevalence : 0.04529
##       Balanced Accuracy : 0.61192
##
##        'Positive' Class : 1
##

# Model 3: caret model
#Using an irrelevant variaable

bancoDflogCaret2  <-
    train(openedAccount ~ marital, family = 'binomial', data= dff1Train, meth
od= 'glm' ) %>%
    predict(dff1Test, type= 'raw') %>%
    bind_cols(dff1Test, predictedClass=.)

bancoDflogCaret2 %>%
  xtabs(~predictedClass+openedAccount, .) %>%
  confusionMatrix(positive = '1')

## Confusion Matrix and Statistics
##
##               openedAccount
## predictedClass    0    1
##              0 7969 1173
```

```
##              1    0    0
##
##                Accuracy : 0.8717
##                  95% CI : (0.8647, 0.8785)
##     No Information Rate : 0.8717
##     P-Value [Acc > NIR] : 0.5078
##
##                   Kappa : 0
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.0000
##             Specificity : 1.0000
##          Pos Pred Value :    NaN
##          Neg Pred Value : 0.8717
##              Prevalence : 0.1283
##          Detection Rate : 0.0000
##    Detection Prevalence : 0.0000
##       Balanced Accuracy : 0.5000
##
##        'Positive' Class : 1
##
```