R Notebook

```
library("skimr")
library("plotly")
```

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

```
##
## 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("tidymodels")
```

```
## Registered S3 method overwritten by 'xts':
##   method     from
##   as.zoo.xts zoo
```

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

```
## v broom     0.5.3     v purrr     0.3.3
## v dials     0.0.4     v recipes   0.1.9
## v dplyr     0.8.3     v rsample   0.0.5
## v infer     0.5.1     v tibble    2.1.3
## v parsnip   0.0.5     v yardstick 0.0.4
```

```
## -- Conflicts ------------------------------------------
tidymodels_conflicts() --
## x purrr::discard()    masks scales::discard()
## x dplyr::filter()     masks plotly::filter(), stats::filter()
## x dplyr::lag()        masks stats::lag()
## x dials::margin()     masks ggplot2::margin()
## x recipes::step()     masks stats::step()
## x recipes::yj_trans() masks scales::yj_trans()
```

```
library("tidyverse")
```

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

```
## v readr   1.3.1     v forcats 0.4.0
## v stringr 1.4.0

## -- Conflicts ---------------------------------------
tidyverse_conflicts() --
## x readr::col_factor() masks scales::col_factor()
## x purrr::discard()    masks scales::discard()
## x dplyr::filter()     masks plotly::filter(), stats::filter()
## x stringr::fixed()    masks recipes::fixed()
## x dplyr::lag()        masks stats::lag()
## x dials::margin()     masks ggplot2::margin()
## x readr::spec()       masks yardstick::spec()
```

```
library("lubridate")
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

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

```
## 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
```

```
library("e1071")
```

```
dff <- read.csv("lab3FraminghamHeart.csv")
str(dff)
```

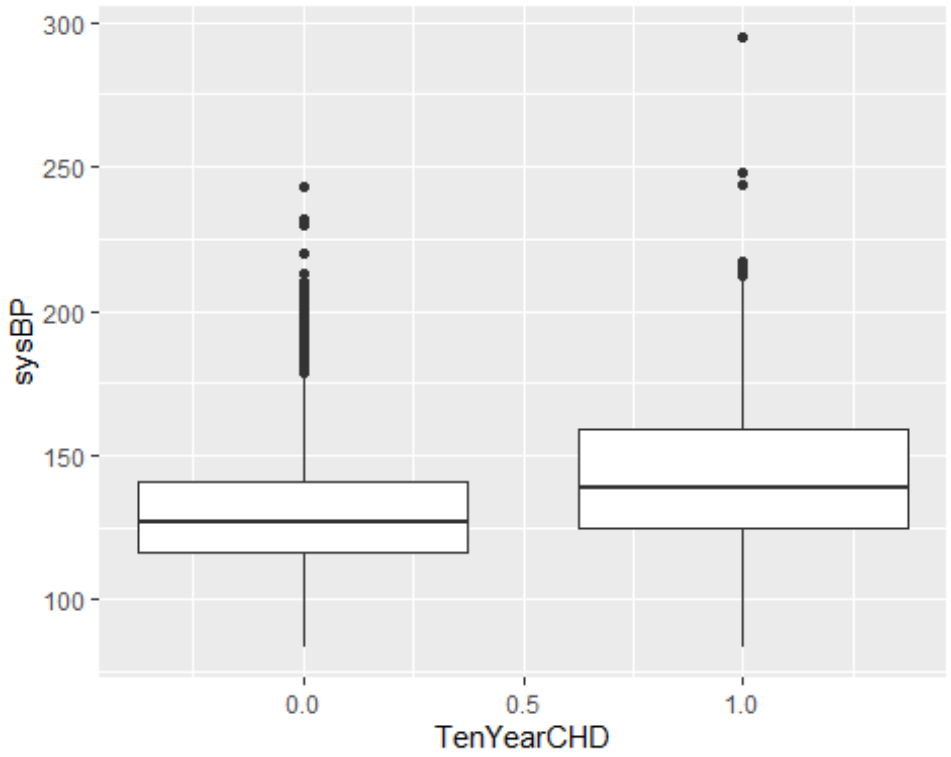
```
## 'data.frame':    3658 obs. of  16 variables:
##  $ gender         : int  1 0 1 0 0 0 0 0 1 1 ...
##  $ age            : int  39 46 48 61 46 43 63 45 52 43 ...
##  $ education      : int  4 2 1 3 3 2 1 2 1 1 ...
##  $ currentSmoker  : int  0 0 1 1 1 0 0 1 0 1 ...
##  $ cigsPerDay     : int  0 0 20 30 23 0 0 20 0 30 ...
##  $ BPMeds         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ prevalentStroke: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ prevalentHyp   : int  0 0 0 1 0 1 0 0 1 1 ...
##  $ diabetes       : int  0 0 0 0 0 0 0 0 0 0 ...
```

```
##  $ totChol        : int  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      : int  80 95 75 65 85 77 60 79 76 93 ...
##  $ glucose        : int  77 76 70 103 85 99 85 78 79 88 ...
##  $ TenYearCHD     : int  0 0 0 1 0 0 1 0 0 0 ...
```

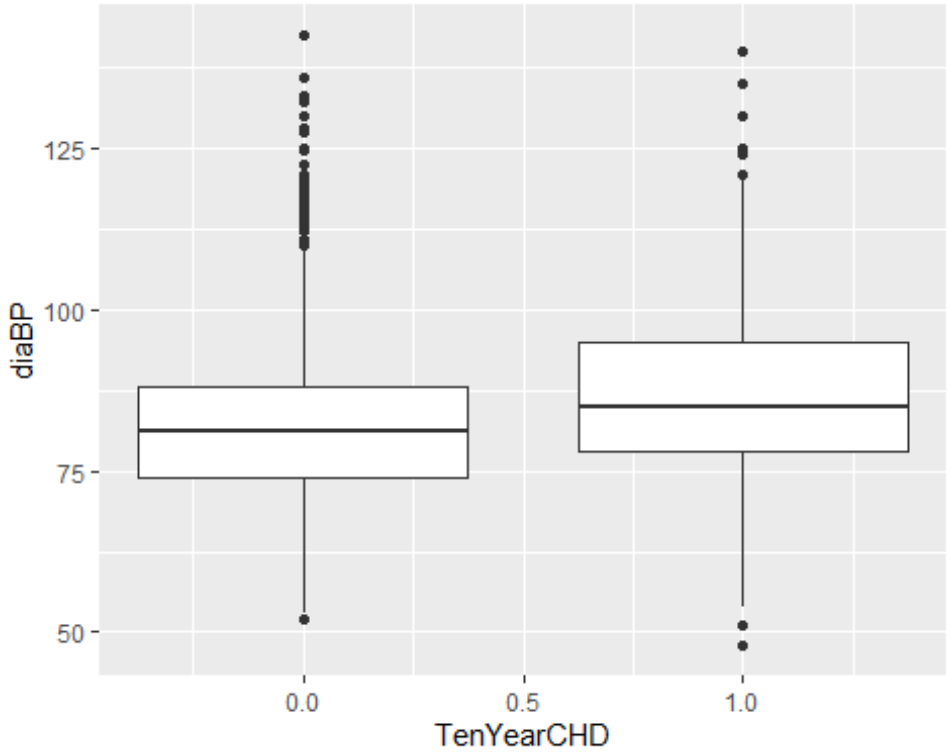
```
colsToFactor <- c('gender', 'education', 'currentSmoker', 'BPMeds',
'prevalentStroke', 'prevalentHyp', 'diabetes')

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

str(dff)
```

```
## '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            : int  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     : int  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        : int  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      : int  80 95 75 65 85 77 60 79 76 93 ...
##  $ glucose        : int  77 76 70 103 85 99 85 78 79 88 ...
##  $ TenYearCHD     : int  0 0 0 1 0 0 1 0 0 0 ...
```

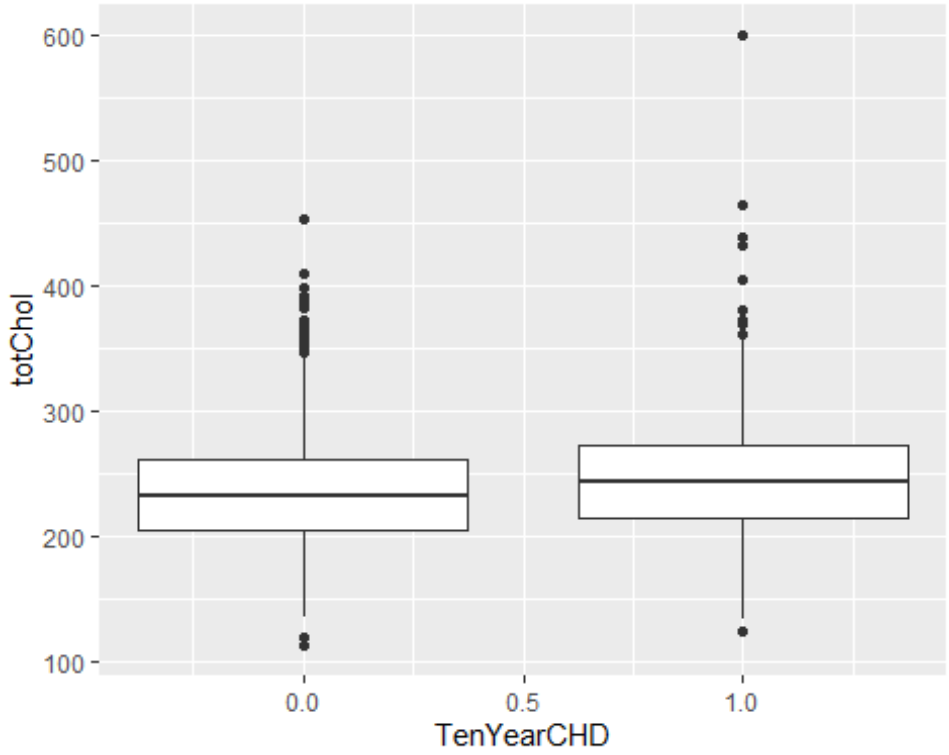
```
plot <-
  ggplot(aes(x=TenYearCHD, y=sysBP,
group=TenYearCHD),data=dff)+geom_boxplot()

#ggplotly(plot)
plot
```




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

 #ggplotly(plot2)
plot2
```




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


 #ggplotly(plot3)
 plot3
```




```
set.seed(123)

dffTrain <-
  dff %>% sample_frac(0.7)

dffTest <-
  dplyr::setdiff(dff, dffTrain)

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

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
```

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

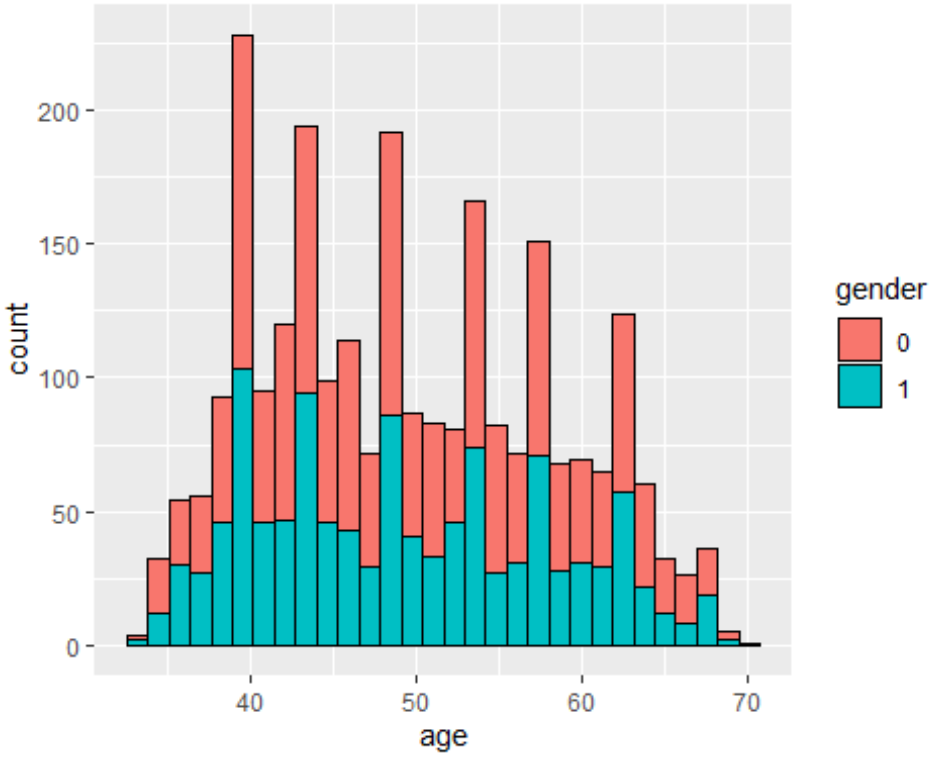
```
## # A tibble: 4 x 3
##   ageGroup     n   pct
##   <fct>    <int> <dbl>
## 1 [30,40]    467  18.2
## 2 (40,50]    973  38.0
## 3 (50,60]    772  30.1
## 4 (60,70]    349  13.6
```

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

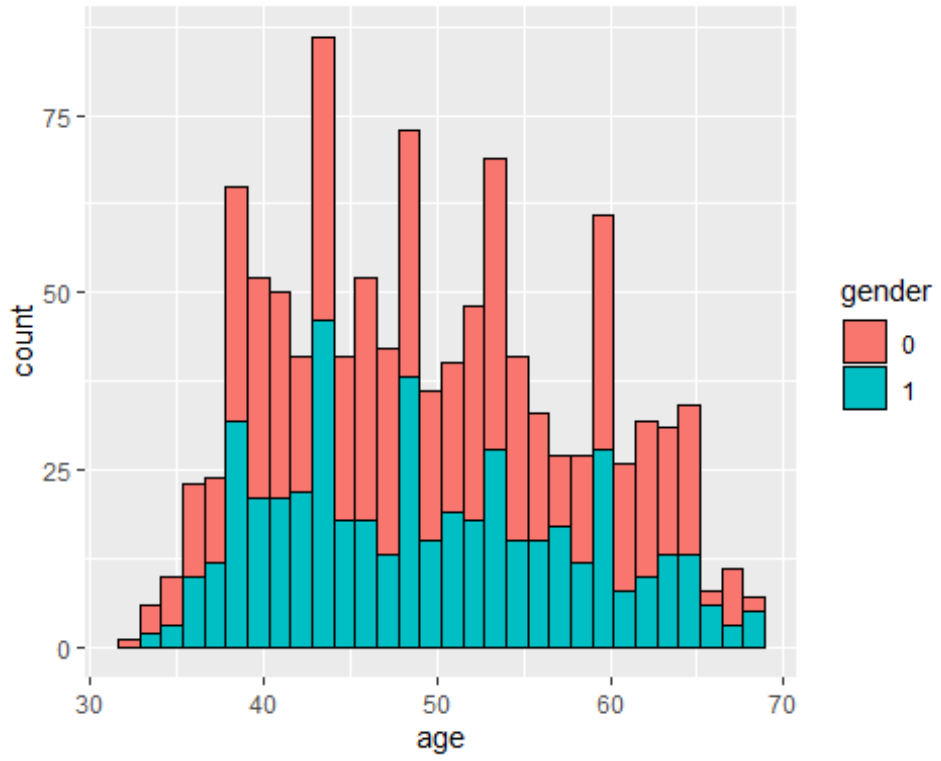
```
## # A tibble: 4 x 3
##   ageGroup     n   pct
##   <fct>    <int> <dbl>
## 1 [30,40]    181  16.5
## 2 (40,50]    421  38.4
## 3 (50,60]    346  31.5
## 4 (60,70]    149  13.6
```

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

#ggplotly(plot4)
plot4
```

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




```
plot5 <-
  ggplot(aes(x=age, fill=gender), data=dffTest)+geom_histogram(color='black')

#ggplotly(plot5)
plot5

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



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
```

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

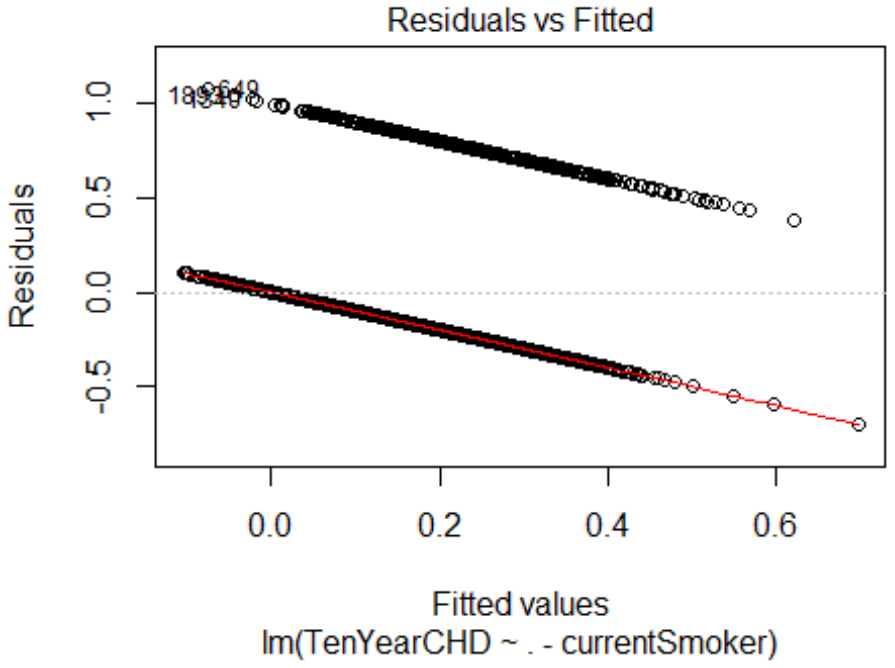
```
## Registered S3 methods overwritten by 'car':
##   method                          from
##   influence.merMod                lme4
##   cooks.distance.influence.merMod lme4
##   dfbeta.influence.merMod         lme4
##   dfbetas.influence.merMod        lme4
```

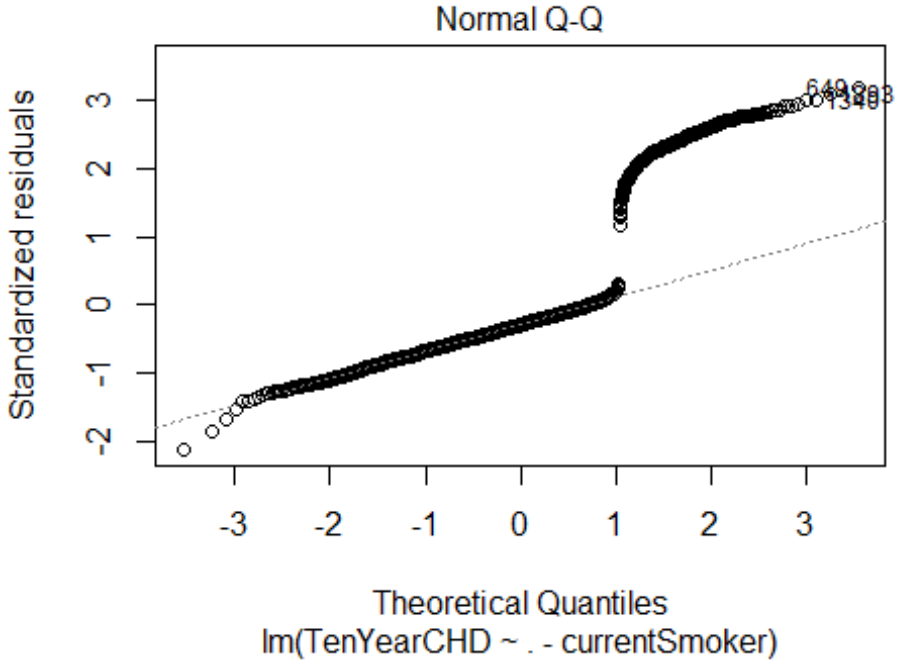
```
##                      GVIF Df GVIF^(1/(2*Df))
## gender           1.232950  1        1.110383
## age              1.398367  1        1.182526
## education        1.139817  3        1.022051
## currentSmoker    2.604754  1        1.613925
## cigsPerDay       2.762784  1        1.662163
## BPMeds           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
```

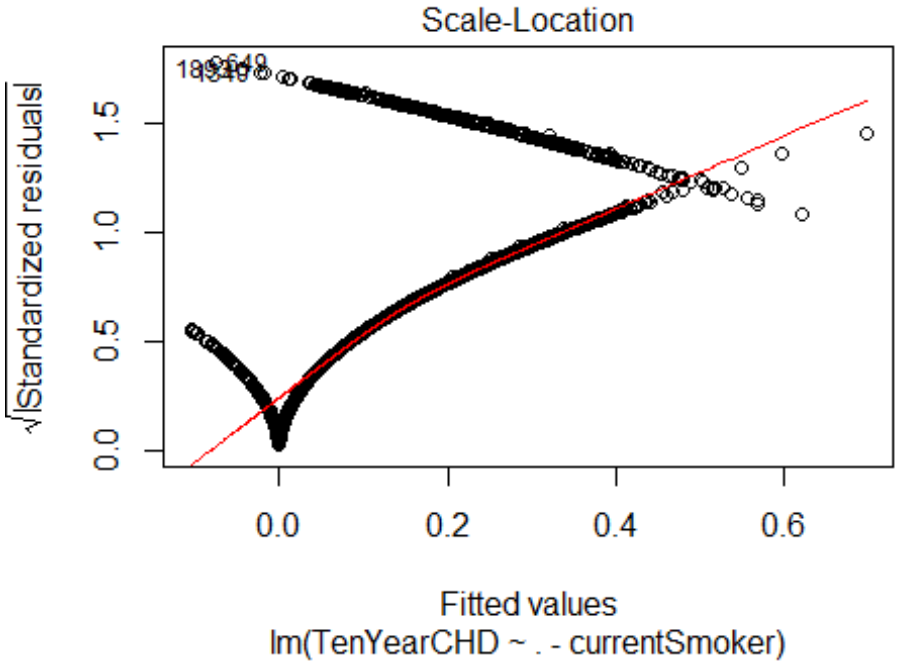
```
newfitLPM <-
  lm(TenYearCHD ~. -currentSmoker, data= dffTrain)

summary(newfitLPM)
```

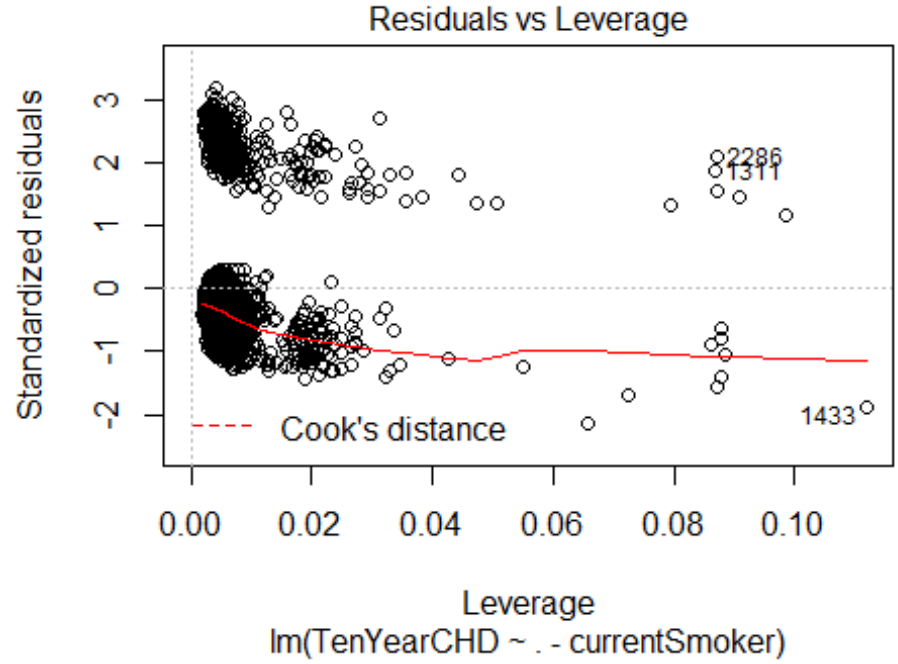
```
##
## Call:
## lm(formula = TenYearCHD ~ . - currentSmoker, data = dffTrain)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.69721 -0.18848 -0.09967 -0.00937  1.07518
##
```

```
## 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

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)






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

```
    mutate(predictedClass = ifelse(predictedProb > 0.5, 1, 0))
#resultsLPM

dffTest %>%
  group_by(TenYearCHD ) %>%
  tally() %>%
  mutate(pct = 100*n/sum(n))

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

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

colsToFactor <- c('TenYearCHD')

dffTrain  <-
  dffTrain%>%
  mutate_at(colsToFactor, ~factor(.))
#dffTrain

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

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

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

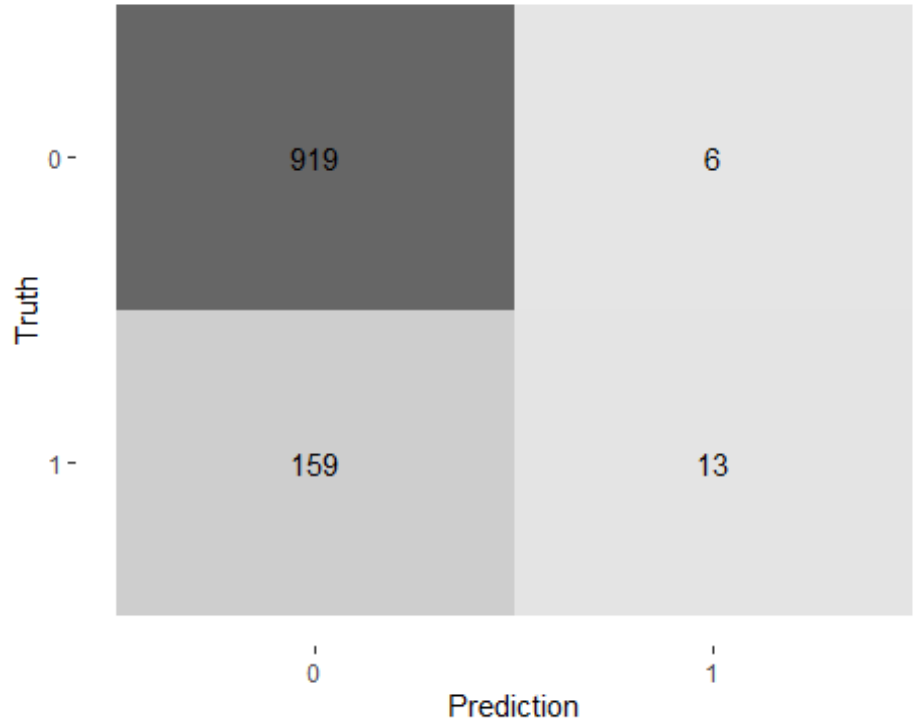
```
##      (Intercept)          gender1              age        education2
##     0.0003606879     1.5253171095     1.0690784440     0.9234189417
##        education3       education4       cigsPerDay          BPMeds1
##     0.7191887265     0.7896676736     1.0202012574     0.9975796686
## prevalentStroke1    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
```

```r
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
```

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

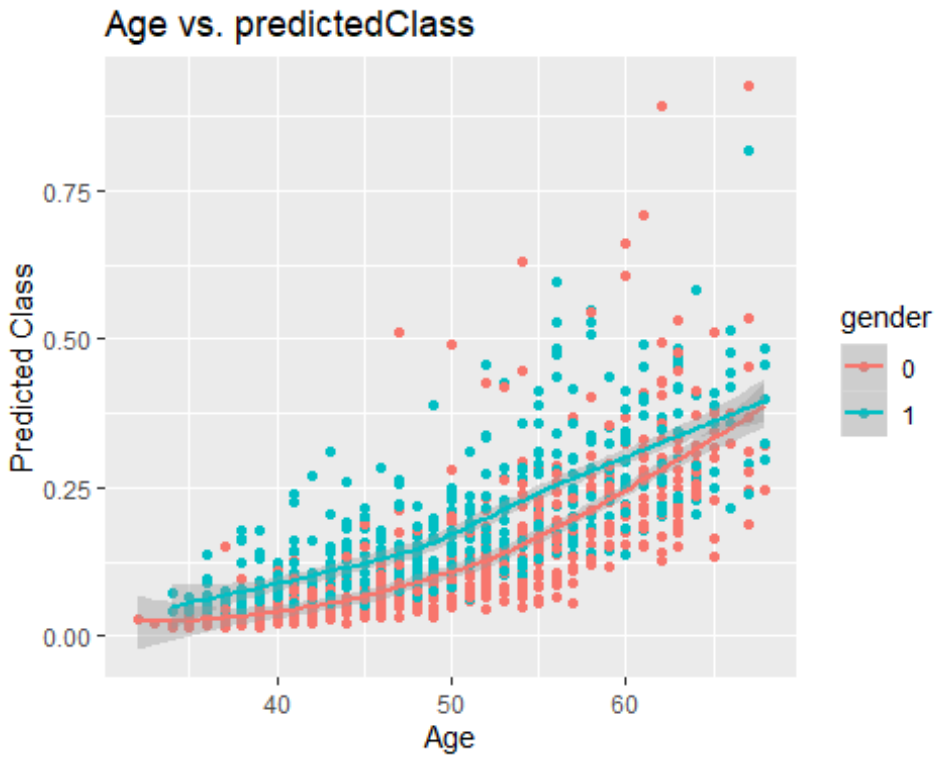
```
## # A tibble: 2 x 3
##   predictedClass     n   pct
##   <fct>          <int> <dbl>
## 1 0               1078 98.3
## 2 1                 19  1.73
```

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

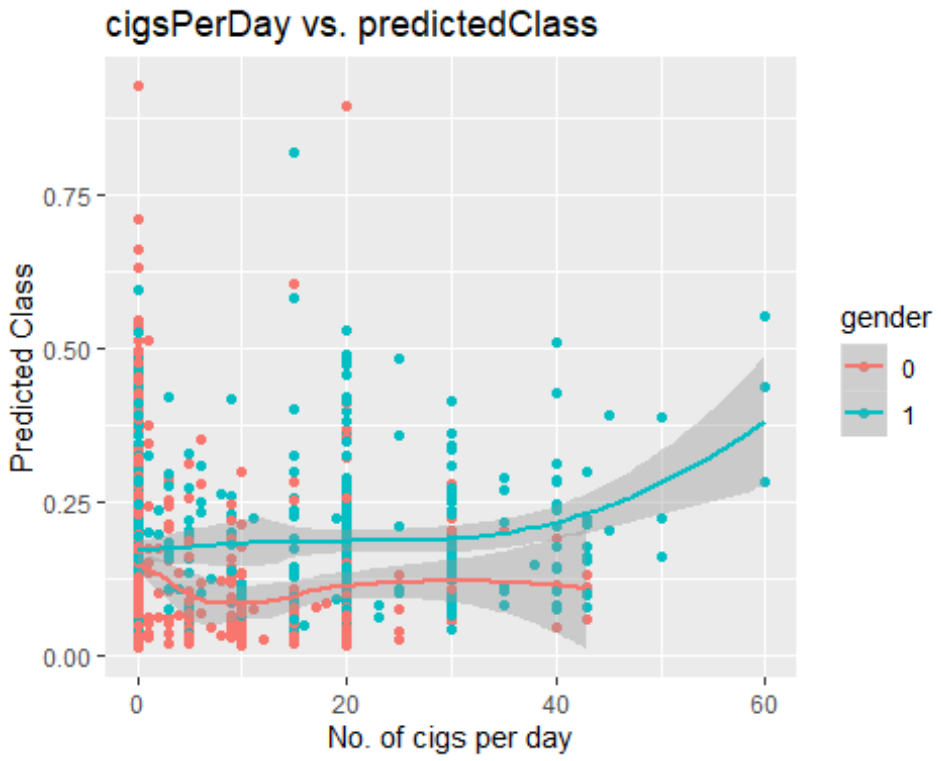



```
plot6 <-
  ggplot(aes(x= age, y=predictedProb, color=gender),
data=resultsLog)+geom_point()+geom_smooth()+labs(title="Age vs.
predictedClass", x="Age", y="Predicted Class")

plot6
```

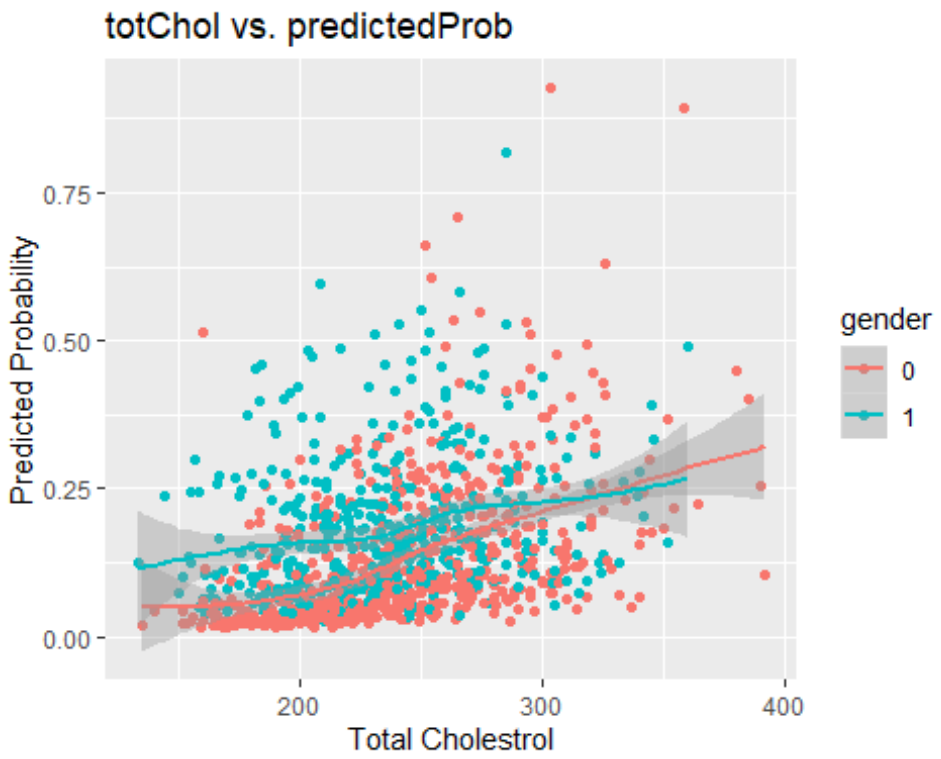
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



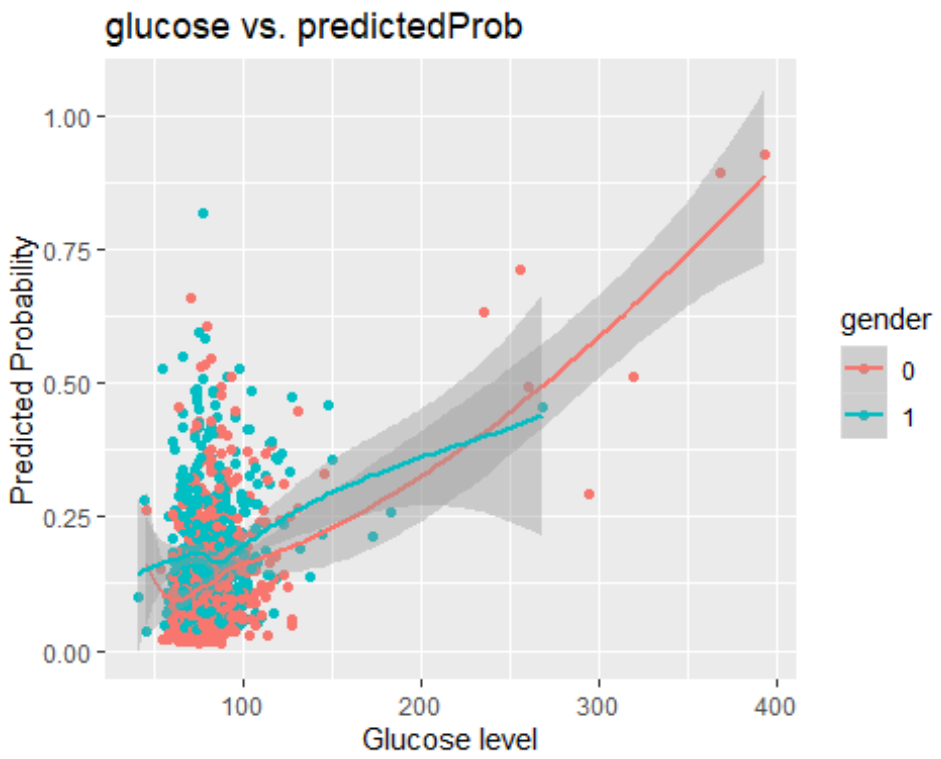

```
plot7 <-
  ggplot(aes(x= cigsPerDay, y=predictedProb, color=gender),
data=resultsLog)+geom_point()+geom_smooth()+labs(title="cigsPerDay vs.
predictedClass", x="No. of cigs per day", y="Predicted Class")

plot7

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




```
plot8 <-
  ggplot(aes(x= totChol, y=predictedProb, color=gender),
data=resultsLog)+geom_point()+geom_smooth()+labs(title="totChol vs.
predictedProb", x="Total Cholestrol", y="Predicted Probability")

plot8

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




```
plot9 <-
  ggplot(aes(x= glucose, y=predictedProb,color=gender),
data=resultsLog)+geom_point()+geom_smooth()+labs(title="glucose vs.
predictedProb", x="Glucose level", y="Predicted Probability")

plot9

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




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

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
##
```

```
dfBanco <- 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.
```

```
skim(dfBanco)
```

Data summary

Name	dfBanco
Number of rows	30488
Number of columns	23

Column type frequency:

character	11
numeric	12

Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
job	0	1	6	13	0	11	0
marital	0	1	6	8	0	3	0
education	0	1	8	19	0	7	0
default	0	1	2	3	0	2	0
housing	0	1	2	3	0	2	0
loan	0	1	2	3	0	2	0
contact	0	1	8	9	0	2	0
month	0	1	3	3	0	10	0
day_of_week	0	1	3	3	0	5	0
poutcome	0	1	7	11	0	3	0
agegroup	0	1	6	15	0	4	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
age	0	1	39.03	10.33	17.00	31.00	37.00	45.00	95.00	
duration	0	1	259.48	261.71	0.00	103.00	181.00	321.00	4918.00	
campaign	0	1	2.52	2.72	1.00	1.00	2.00	3.00	43.00	
pdays	0	1	956.33	201.37	0.00	999.00	999.00	999.00	999.00	
previous	0	1	0.19	0.52	0.00	0.00	0.00	0.00	7.00	
emp.var.rate	0	1	-0.07	1.61	-3.40	-1.80	1.10	1.40	1.40	
cons.price.idx	0	1	93.52	0.59	92.20	93.08	93.44	93.99	94.77	
cons.conf.idx	0	1	-40.60	4.79	-50.80	-42.70	-41.80	-36.40	-26.90	
euribor3m	0	1	3.46	1.78	0.63	1.31	4.86	4.96	5.04	
nr.employed	0	1	5160.81	75.16	4963.60	5099.10	5191.00	5228.10	5228.10	
openedAc	0	1	0.13	0.33	0.00	0.00	0.00	0.00	1.00	

count										▁▁
newcustomer	0	1	0.85	0.36	0.00	1.00	1.00	1.00	1.00	▂▁▁▁▇

```
colsToFactorBank <- c('openedAccount', 'newcustomer', 'default', 'housing',
'loan')

dfBanco <- dfBanco %>%
  mutate_at(colsToFactorBank, ~factor(.))

str(dfBanco)

## 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       : 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       : 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"
"nonexistent" ...
##  $ 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 -
36.4 -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      : chr  "Adults" "Adults" "Adults" "Adults" ...
##  $ newcustomer   : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...

set.seed(123)

dfTrainBanco <- dfBanco %>% sample_frac(0.7)

dfTestBanco <- dplyr::setdiff(dfBanco, dfTrainBanco)

bancoDflogit <-
  glm(openedAccount~. -(duration),family='binomial',data=dfTestBanco)
```

```r
summary(bancoDflogit)
```

```
##
## Call:
## glm(formula = openedAccount ~ . - (duration), family = "binomial",
##     data = dfTestBanco)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2651  -0.4044  -0.3241  -0.2654   2.8866
##
## Coefficients: (1 not defined because of singularities)
##                                Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  -3.169e+02  6.773e+01  -4.679 2.88e-06 ***
## age                          -1.680e-03  6.898e-03  -0.244  0.80757
## jobblue-collar               -2.454e-01  1.451e-01  -1.692  0.09071 .
## jobentrepreneur              -8.622e-02  2.146e-01  -0.402  0.68778
## jobhousemaid                  7.953e-02  2.599e-01   0.306  0.75963
## jobmanagement                -2.662e-01  1.543e-01  -1.725  0.08455 .
## jobretired                    8.730e-02  2.243e-01   0.389  0.69705
## jobself-employed             -1.442e-01  2.029e-01  -0.711  0.47732
## jobservices                  -2.323e-01  1.527e-01  -1.521  0.12819
## jobstudent                    1.087e-01  2.213e-01   0.491  0.62319
## jobtechnician                -8.129e-02  1.241e-01  -0.655  0.51250
## jobunemployed                -7.586e-02  2.260e-01  -0.336  0.73715
## maritalmarried                6.297e-02  1.207e-01   0.522  0.60193
## maritalsingle                -2.897e-02  1.381e-01  -0.210  0.83393
## educationbasic.6y             5.447e-01  2.370e-01   2.298  0.02154 *
## educationbasic.9y             3.858e-01  1.873e-01   2.059  0.03946 *
## educationhigh.school          3.516e-01  1.817e-01   1.935  0.05297 .
## educationilliterate           1.412e+00  1.442e+00   0.979  0.32746
## educationprofessional.course  2.198e-01  1.958e-01   1.122  0.26172
## educationuniversity.degree    3.168e-01  1.814e-01   1.746  0.08083 .
## defaultyes                   -9.326e+00  2.295e+02  -0.041  0.96758
## housingyes                   -4.499e-02  7.332e-02  -0.614  0.53946
## loanyes                       9.764e-03  1.014e-01   0.096  0.92328
## contacttelephone             -9.220e-01  1.398e-01  -6.596 4.22e-11 ***
## monthaug                      6.422e-01  2.195e-01   2.926  0.00343 **
## monthdec                      8.929e-02  3.778e-01   0.236  0.81316
## monthjul                     -1.045e-02  1.733e-01  -0.060  0.95191
## monthjun                     -8.328e-01  2.247e-01  -3.707  0.00021 ***
## monthmar                      2.068e+00  2.696e-01   7.670 1.72e-14 ***
## monthmay                     -2.816e-01  1.462e-01  -1.926  0.05410 .
## monthnov                     -6.099e-01  2.168e-01  -2.814  0.00490 **
## monthoct                      1.485e-01  2.681e-01   0.554  0.57962
## monthsep                      3.018e-01  3.167e-01   0.953  0.34060
## day_of_weekmon               -1.685e-01  1.180e-01  -1.428  0.15335
## day_of_weekthu                2.073e-01  1.154e-01   1.797  0.07228 .
## day_of_weektue               -1.169e-01  1.206e-01  -0.970  0.33214
```

```
## day_of_weekwed                6.629e-02  1.186e-01   0.559  0.57613
## campaign                     -2.950e-02  1.821e-02  -1.620  0.10527
## pdays                        -1.240e-03  3.911e-04  -3.171  0.00152 **
## previous                     -3.613e-02  1.120e-01  -0.323  0.74703
## poutcomenonexistent           5.130e-01  1.759e-01   2.917  0.00354 **
## poutcomesuccess               8.688e-01  3.830e-01   2.269  0.02329 *
## emp.var.rate                 -2.036e+00  2.452e-01  -8.303  < 2e-16 ***
## cons.price.idx                2.770e+00  4.447e-01   6.229 4.70e-10 ***
## cons.conf.idx                 2.951e-02  1.438e-02   2.052  0.04020 *
## euribor3m                     3.850e-01  2.378e-01   1.619  0.10542
## nr.employed                   1.090e-02  5.563e-03   1.960  0.05004 .
## agegroupSenior Citizens       3.350e-01  2.452e-01   1.366  0.17184
## agegroupTeenagers            -1.459e+00  8.601e-01  -1.696  0.08985 .
## agegroupYoung Adults          2.877e-02  1.196e-01   0.240  0.80998
## 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: 7005.7  on 9141  degrees of freedom
## Residual deviance: 5304.9  on 9092  degrees of freedom
## AIC: 5404.9
##
## Number of Fisher Scoring iterations: 11
```

```
bancoDfCaret  <-
    train(openedAccount ~. -(duration), family = 'binomial', data=
dfTrainBanco, method= 'glm' ) %>%
    predict(dfTestBanco, type= 'raw') %>%
    bind_cols(dfTestBanco, 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
```

```r
#bancoDfCaret

bancoDfCaret %>%
  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
##
```

```
bancoDfCaret1  <-
    train(openedAccount ~. -(duration + marital + agegroup + housing + loan +
day_of_week + euribor3m + newcustomer + contact), family = 'binomial', data=
dfTrainBanco, method= 'glm' ) %>%
    predict(dfTestBanco, type= 'raw') %>%
    bind_cols(dfTestBanco, 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
```

```
#bancoDfCaret1

bancoDfCaret1 %>%
  xtabs(~predictedClass+openedAccount, .) %>%
  confusionMatrix(positive = '1')
```

```
## Confusion Matrix and Statistics
##
##                 openedAccount
## predictedClass    0    1
##               0 7841  886
##               1  128  287
##
##                Accuracy : 0.8891
##                  95% CI : (0.8825, 0.8955)
##     No Information Rate : 0.8717
##     P-Value [Acc > NIR] : 2.162e-07
##
##                   Kappa : 0.3156
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.24467
##             Specificity : 0.98394
```

```
##          Pos Pred Value : 0.69157
##          Neg Pred Value : 0.89848
##              Prevalence : 0.12831
##          Detection Rate : 0.03139
##    Detection Prevalence : 0.04539
##       Balanced Accuracy : 0.61430
##
##        'Positive' Class : 1
##
```

```
bancoDfCaret2  <-
train(openedAccount ~ marital , family = 'binomial', data= dfTrainBanco,
method= 'glm' ) %>%
    predict(dfTestBanco, type= 'raw') %>%
    bind_cols(dfTestBanco, predictedClass=.)

#bancoDfCaret2

bancoDfCaret2 %>%
  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
##
```