A12.Zunqiu.Wang

Zunqiu Wang

11/30/2021

Q1

```
library(openintro)
```

```
## Loading required package: airports
```

```
## Loading required package: cherryblossom
```

```
## Loading required package: usdata
```

```
library(psych)
library(ggplot2)
```

```
##
## Attaching package: 'ggplot2'
```

```
## The following objects are masked from 'package:psych':
##
##     %+%, alpha
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1
## v purrr   0.3.4
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x ggplot2::%+%()   masks psych::%+%()
## x ggplot2::alpha() masks psych::alpha()
## x dplyr::filter()  masks stats::filter()
## x dplyr::lag()     masks stats::lag()
```

```
library(dplyr)
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
```

```
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
## The following object is masked from 'package:openintro':
##
##     tips
```

```
library(mlr)
```

```
## Loading required package: ParamHelpers
```

```
## Warning message: 'mlr' is in 'maintenance-only' mode since July 2019.
## Future development will only happen in 'mlr3'
## (<https://mlr3.mlr-org.com>). Due to the focus on 'mlr3' there might be
## uncaught bugs meanwhile in {mlr} - please consider switching.
```

```
##
## Attaching package: 'mlr'
```

```
## The following object is masked from 'package:openintro':
##
##     bac
```

```
library(caret)
```

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

```
##
## Attaching package: 'lattice'
```

```
## The following objects are masked from 'package:openintro':
##
##     ethanol, lsegments
```

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

```
## The following object is masked from 'package:mlr':
##
##     train
```

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

```
## The following object is masked from 'package:openintro':
##
##     dotPlot
```

```
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

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

```
## The following objects are masked from 'package:openintro':
##
##     housing, mammals
```

```
library(leaps)
library(UpSetR)
```

```
##
## Attaching package: 'UpSetR'
```

```
## The following object is masked from 'package:lattice':
##
##     histogram
```

```
library(naniar)
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(glmnet)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
## Loaded glmnet 4.1-2
```

```
library(e1071)
```

```
##
## Attaching package: 'e1071'
```

```
## The following object is masked from 'package:mlr':
##
##     impute
```

```
data("ncbirths")
```

```
ncbirths
```

```
## # A tibble: 1,000 x 13
##     fage  mage mature      weeks premie    visits marital gained weight lowbirthweight
##    <int> <int> <fct>       <int> <fct>      <int> <fct>    <int>  <dbl> <fct>
##  1    NA    13 younger mom    39 full term     10 not ma~     38   7.63 not low
##  2    NA    14 younger mom    42 full term     15 not ma~     20   7.88 not low
##  3    19    15 younger mom    37 full term     11 not ma~     38   6.63 not low
##  4    21    15 younger mom    41 full term      6 not ma~     34   8    not low
##  5    NA    15 younger mom    39 full term      9 not ma~     27   6.38 not low
##  6    NA    15 younger mom    38 full term     19 not ma~     22   5.38 low
##  7    18    15 younger mom    37 full term     12 not ma~     76   8.44 not low
##  8    17    15 younger mom    35 premie         5 not ma~     15   4.69 low
##  9    NA    16 younger mom    38 full term      9 not ma~     NA   8.81 not low
## 10    20    16 younger mom    37 full term     13 not ma~     52   6.94 not low
## # ... with 990 more rows, and 3 more variables: gender <fct>, habit <fct>,
## #   whitemom <fct>
```

```
# check variable type
str(ncbirths)
```

```
## tibble [1,000 x 13] (S3: tbl_df/tbl/data.frame)
##  $ fage          : int [1:1000] NA NA 19 21 NA NA 18 17 NA 20 ...
##  $ mage          : int [1:1000] 13 14 15 15 15 15 15 15 16 16 ...
##  $ mature        : Factor w/ 2 levels "mature mom","younger mom": 2 2 2 2 2 2 2 2 2 2 ...
##  $ weeks         : int [1:1000] 39 42 37 41 39 38 37 35 38 37 ...
##  $ premie        : Factor w/ 2 levels "full term","premie": 1 1 1 1 1 1 1 2 1 1 ...
```

```
##  $ visits        : int [1:1000] 10 15 11 6 9 19 12 5 9 13 ...
##  $ marital       : Factor w/ 2 levels "not married",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ gained        : int [1:1000] 38 20 38 34 27 22 76 15 NA 52 ...
##  $ weight        : num [1:1000] 7.63 7.88 6.63 8 6.38 5.38 8.44 4.69 8.81 6.94 ...
##  $ lowbirthweight: Factor w/ 2 levels "low","not low": 2 2 2 2 2 1 2 1 2 2 ...
##  $ gender        : Factor w/ 2 levels "female","male": 2 2 1 2 1 2 2 2 2 1 ...
##  $ habit         : Factor w/ 2 levels "nonsmoker","smoker": 1 1 1 1 1 1 1 1 1 1 ...
##  $ whitemom      : Factor w/ 2 levels "not white","white": 1 1 2 2 1 1 1 1 2 2 ...
```

```
summary(ncbirths)
```

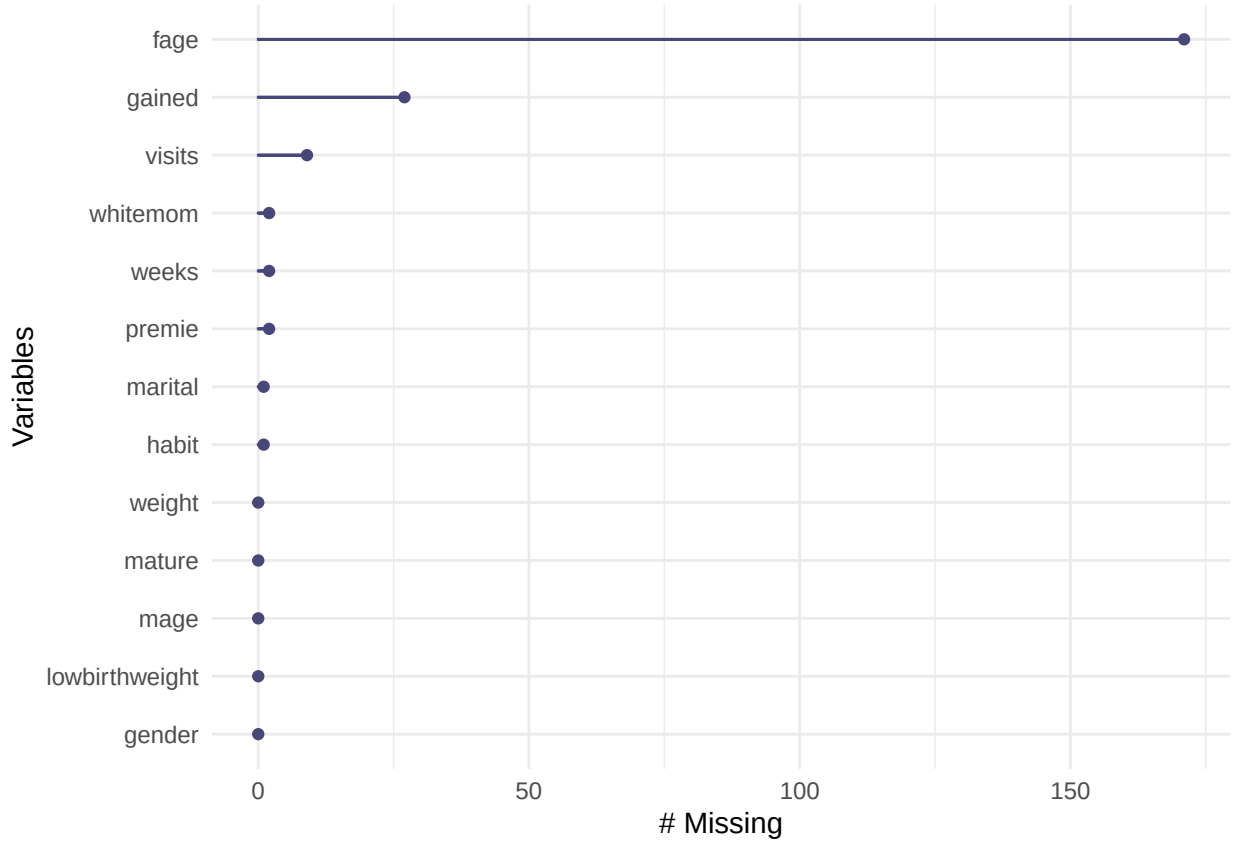
```
##       fage            mage            mature        weeks            premie
##  Min.   :14.00   Min.   :13   mature mom :133   Min.   :20.00   full term:846
##  1st Qu.:25.00   1st Qu.:22   younger mom:867   1st Qu.:37.00   premie   :152
##  Median :30.00   Median :27                     Median :39.00   NA's     :  2
##  Mean   :30.26   Mean   :27                     Mean   :38.33
##  3rd Qu.:35.00   3rd Qu.:32                     3rd Qu.:40.00
##  Max.   :55.00   Max.   :50                     Max.   :45.00
##  NA's   :171                                    NA's   :2
##      visits           marital        gained          weight
##  Min.   : 0.0   not married:386   Min.   : 0.00   Min.   : 1.000
##  1st Qu.:10.0   married    :613   1st Qu.:20.00   1st Qu.: 6.380
##  Median :12.0   NA's       :  1   Median :30.00   Median : 7.310
##  Mean   :12.1                     Mean   :30.33   Mean   : 7.101
##  3rd Qu.:15.0                     3rd Qu.:38.00   3rd Qu.: 8.060
##  Max.   :30.0                     Max.   :85.00   Max.   :11.750
##  NA's   :9                        NA's   :27
##  lowbirthweight    gender          habit          whitemom
##  low    :111    female:503   nonsmoker:873   not white:284
##  not low:889    male  :497   smoker   :126   white    :714
##                              NA's     :  1   NA's     :  2
##
##
##
##
```

```
#check if NA
colSums(is.na(ncbirths))
```

```
##           fage           mage         mature          weeks         premie
##            171              0              0              2              2
##         visits        marital         gained         weight lowbirthweight
##              9              1             27              0              0
##         gender          habit       whitemom
##              0              1              2
```

```
gg_miss_var(ncbirths)
```

```
## Warning: It is deprecated to specify `guide = FALSE` to remove a guide. Please
## use `guide = "none"` instead.
```




```
# impute NA
## impute numerical var with median
ncbirths <- ncbirths %>%
  mutate_if(is.numeric, function(x) ifelse(is.na(x), median(x, na.rm = T), x))

## impute categorical var with mode
get_mode <- function(x){
  uniq_val <- unique(x)
  count_unique <- tabulate(match(x, uniq_val))
  uniq_val[which.max(count_unique)]
}

###premie
ncbirths$premie[is.na(ncbirths$premie)] <- get_mode(ncbirths$premie)


### marital
ncbirths$marital[is.na(ncbirths$marital)] <- get_mode(ncbirths$marital)

### habit
ncbirths$habit[is.na(ncbirths$habit)] <- get_mode(ncbirths$habit)

### whitemom
ncbirths$whitemom[is.na(ncbirths$whitemom)] <- get_mode(ncbirths$whitemom)

str(ncbirths)
```

```
## tibble [1,000 x 13] (S3: tbl_df/tbl/data.frame)
##  $ fage          : int [1:1000] 30 30 19 21 30 30 18 17 30 20 ...
##  $ mage          : int [1:1000] 13 14 15 15 15 15 15 15 16 16 ...
##  $ mature        : Factor w/ 2 levels "mature mom","younger mom": 2 2 2 2 2 2 2 2 2 2 ...
##  $ weeks         : num [1:1000] 39 42 37 41 39 38 37 35 38 37 ...
##  $ premie        : Factor w/ 2 levels "full term","premie": 1 1 1 1 1 1 1 2 1 1 ...
##  $ visits        : int [1:1000] 10 15 11 6 9 19 12 5 9 13 ...
##  $ marital       : Factor w/ 2 levels "not married",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ gained        : int [1:1000] 38 20 38 34 27 22 76 15 30 52 ...
##  $ weight        : num [1:1000] 7.63 7.88 6.63 8 6.38 5.38 8.44 4.69 8.81 6.94 ...
##  $ lowbirthweight: Factor w/ 2 levels "low","not low": 2 2 2 2 2 1 2 1 2 2 ...
##  $ gender        : Factor w/ 2 levels "female","male": 2 2 1 2 1 2 2 2 2 1 ...
##  $ habit         : Factor w/ 2 levels "nonsmoker","smoker": 1 1 1 1 1 1 1 1 1 1 ...
##  $ whitemom      : Factor w/ 2 levels "not white","white": 1 1 2 2 1 1 1 1 2 2 ...
```

```
summary(ncbirths)
```

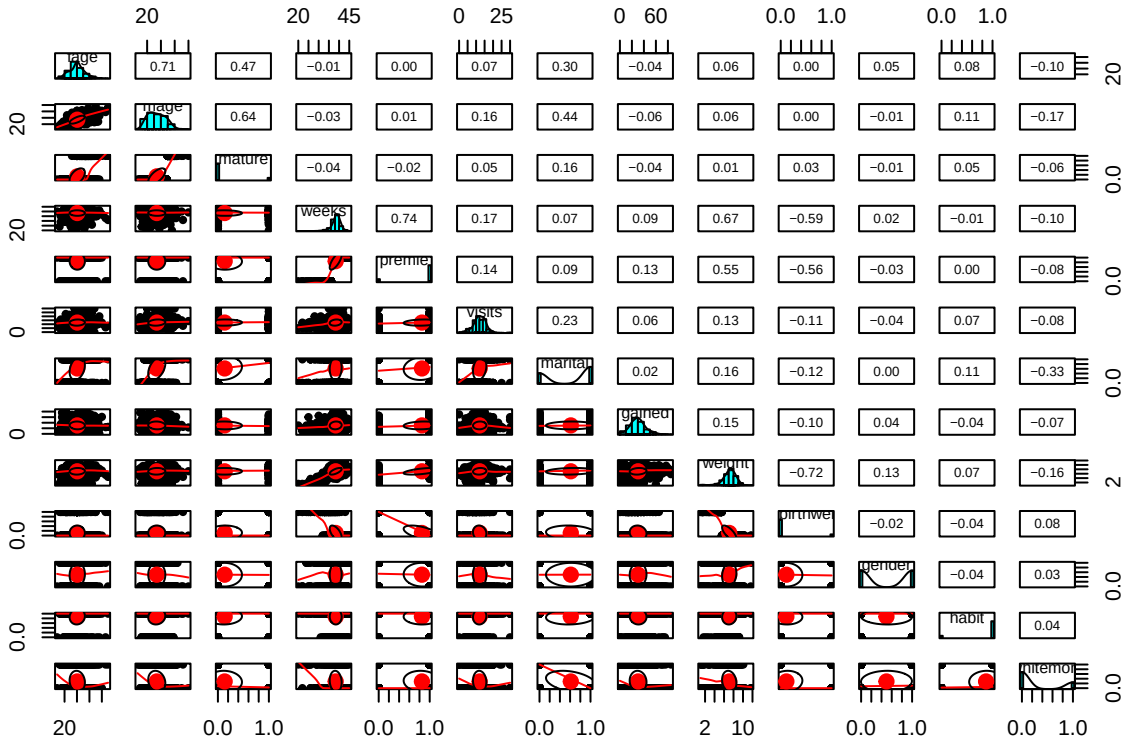
```
##       fage            mage            mature        weeks             premie
##  Min.   :14.00   Min.   :13   mature mom :133   Min.   :20.00   full term:848
##  1st Qu.:26.00   1st Qu.:22   younger mom:867   1st Qu.:37.00   premie   :152
##  Median :30.00   Median :27                     Median :39.00
##  Mean   :30.21   Mean   :27                     Mean   :38.34
##  3rd Qu.:34.00   3rd Qu.:32                     3rd Qu.:40.00
##  Max.   :55.00   Max.   :50                     Max.   :45.00
##      visits           marital        gained          weight
##  Min.   : 0.0   not married:386   Min.   : 0.00   Min.   : 1.000
##  1st Qu.:10.0   married    :614   1st Qu.:21.00   1st Qu.: 6.380
##  Median :12.0                     Median :30.00   Median : 7.310
##  Mean   :12.1                     Mean   :30.32   Mean   : 7.101
##  3rd Qu.:15.0                     3rd Qu.:38.00   3rd Qu.: 8.060
##  Max.   :30.0                     Max.   :85.00   Max.   :11.750
##  lowbirthweight    gender          habit          whitemom
##  low    :111    female:503   nonsmoker:874   not white:284
##  not low:889    male  :497   smoker   :126   white    :716
##
##
##
##
```

```
#check if NA again
colSums(is.na(ncbirths))
```

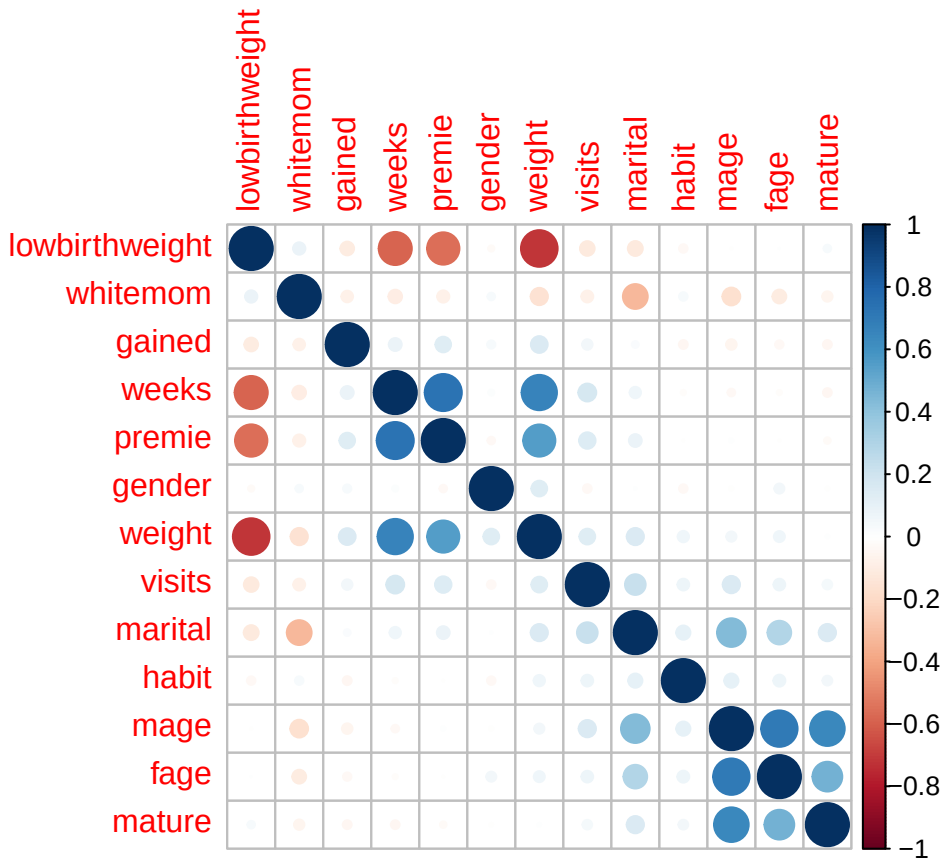
```
##           fage           mage         mature          weeks         premie
##              0              0              0              0              0
##         visits        marital         gained         weight lowbirthweight
##              0              0              0              0              0
##         gender          habit       whitemom
##              0              0              0
```

```
# convert catogorical var to binary var
ncbirths$mature <- ifelse(ncbirths$mature == "mature mom",1,0)
ncbirths$premie <- ifelse(ncbirths$premie == "full term",1,0)
ncbirths$marital <- ifelse(ncbirths$marital == "married",1,0)
ncbirths$lowbirthweight <- ifelse(ncbirths$lowbirthweight == "low",1,0)
ncbirths$gender <- ifelse(ncbirths$gender == "male",1,0)
ncbirths$habit <- ifelse(ncbirths$habit == "nonsmoker",1,0)
```

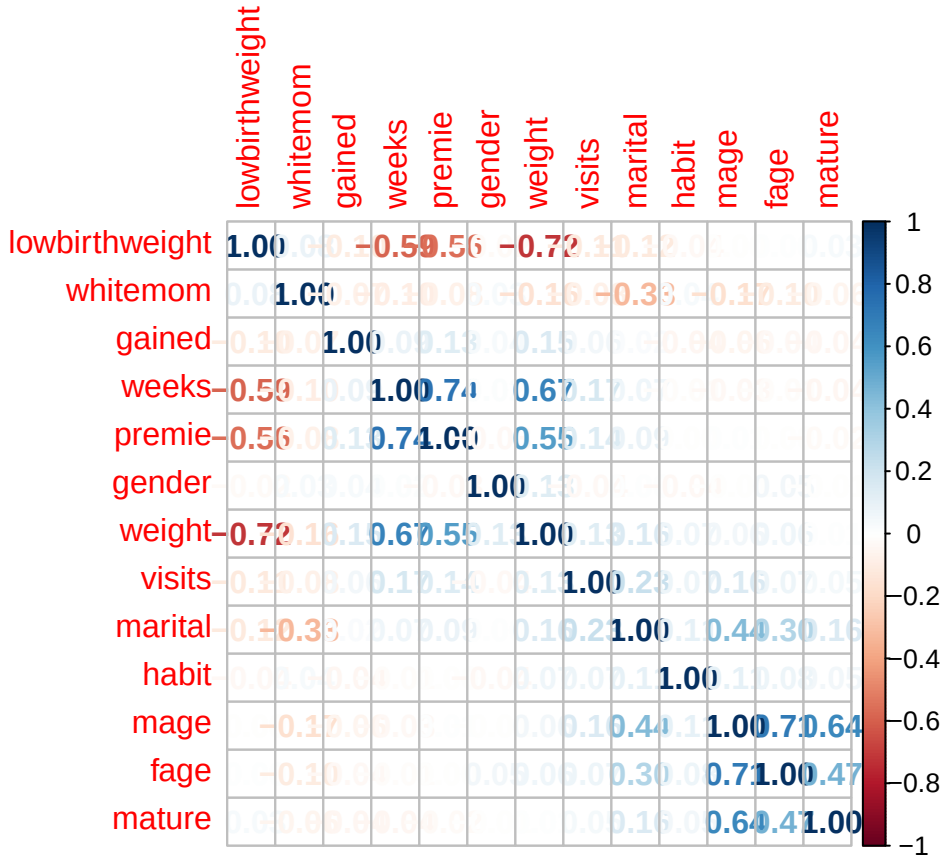
```
ncbirths$whitemom <- ifelse(ncbirths$whitemom == "not white",1,0)

# cor matrix
pairs.panels(ncbirths)
```




```
correlations <- cor(ncbirths)
corrplot(correlations, method="circle", order = "AOE")
```




```
corrplot(correlations, method="number", order = "AOE")
```



Q1

a

```
x <- as.matrix(ncbirths[, -9])
y <- ncbirths$weight

set.seed(123)
index <- sample(1:nrow(x),size=nrow(x)*0.5,replace = FALSE) #random selection of 50% data.

#train set and test set
trainx <- x[index,] # 50% training data
trainy <- y[index]
testx <- x[-index,] # remaining 50% test data
testy <- y[-index]

#ridge,lasso,elastic net model with cv
ridge.fit <- cv.glmnet(trainx, trainy, alpha=0)
lasso.fit <- cv.glmnet(trainx, trainy, alpha=1)
elnet.fit <- cv.glmnet(trainx, trainy, alpha=0.5)
```

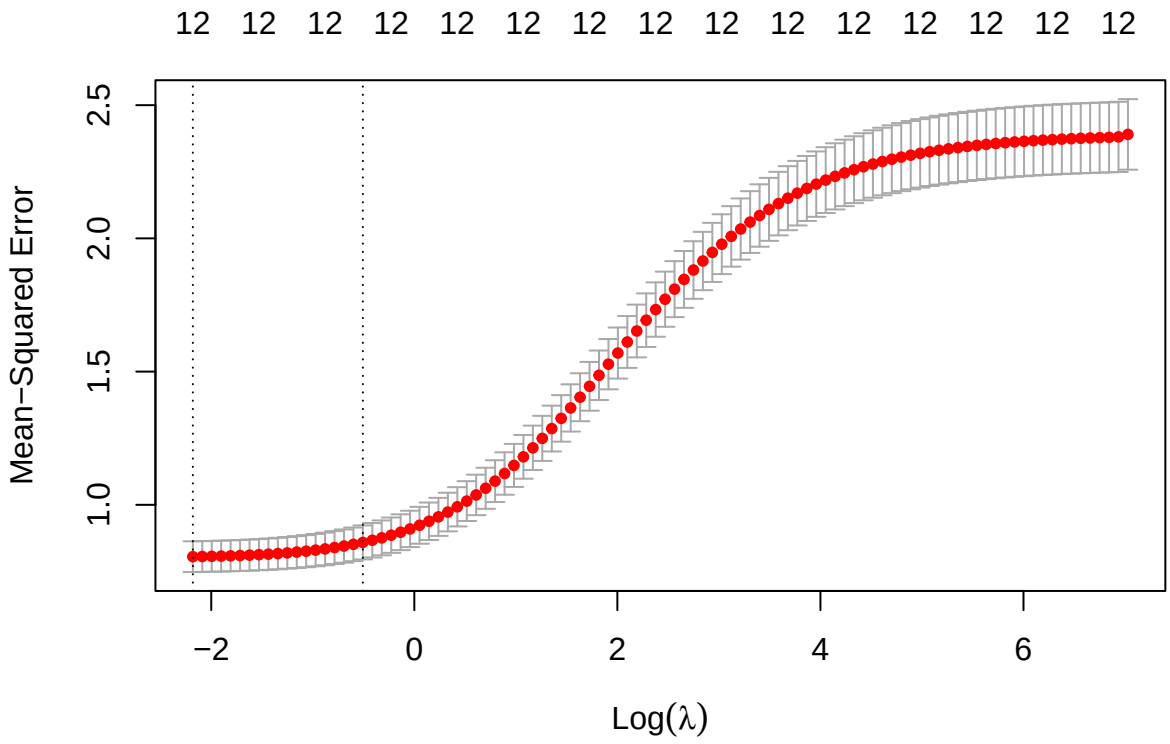
b

```
# comparison for lowest/best mse
ridge.fit
```

```
##
## Call:  cv.glmnet(x = trainx, y = trainy, alpha = 0)
```

```
##
## Measure: Mean-Squared Error
##
##     Lambda Index Measure      SE Nonzero
## min 0.1129   100  0.8053 0.05768      12
## 1se 0.6024    82  0.8588 0.06377      12
```

```
plot(ridge.fit)
```




```
min(ridge.fit$cvm)# lowest error for ridge
```

```
## [1] 0.8053494
```

```
predict(ridge.fit, type="coefficients",s=ridge.fit$lambda.min) # all predictors
```

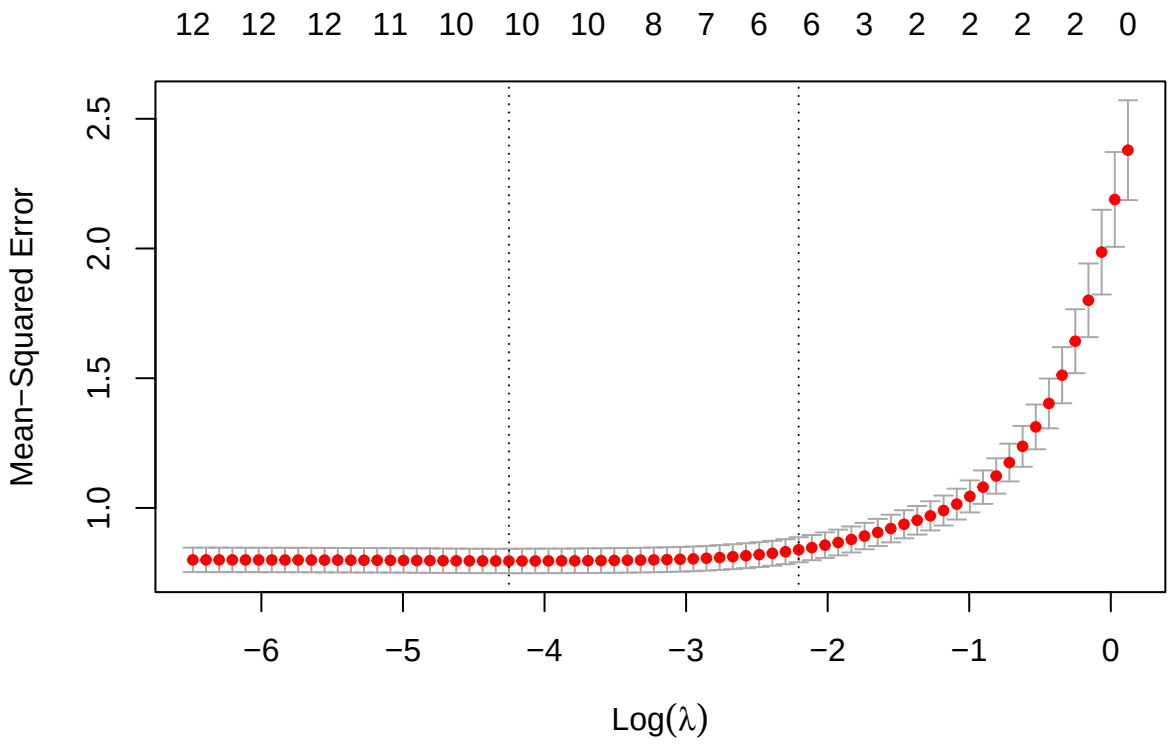
```
## 13 x 1 sparse Matrix of class "dgCMatrix"
##                              s1
## (Intercept)    -0.388071436
## fage            0.012531886
## mage           -0.002245040
## mature          0.084641840
## weeks           0.182063423
## premie          0.122708963
## visits         -0.006968339
## marital         0.142759209
## gained          0.004902728
## lowbirthweight -2.204789608
## gender          0.314573123
```

```
## habit           0.146230048
## whitemom       -0.381225579
```

```
lasso.fit # find min lmbda and its lowest MSE
```

```
##
## Call:  cv.glmnet(x = trainx, y = trainy, alpha = 1)
##
## Measure: Mean-Squared Error
##
##      Lambda Index Measure      SE Nonzero
## min 0.01424    48  0.7956 0.04709      10
## 1se 0.11029    26  0.8388 0.04831       6
```

```
plot(lasso.fit)
```




```
min(lasso.fit$cvm) # lowest error for lasso
```

```
## [1] 0.7956483
```

```
predict(lasso.fit, type="coefficients",s=lasso.fit$lambda.min) # 10 predictors
```

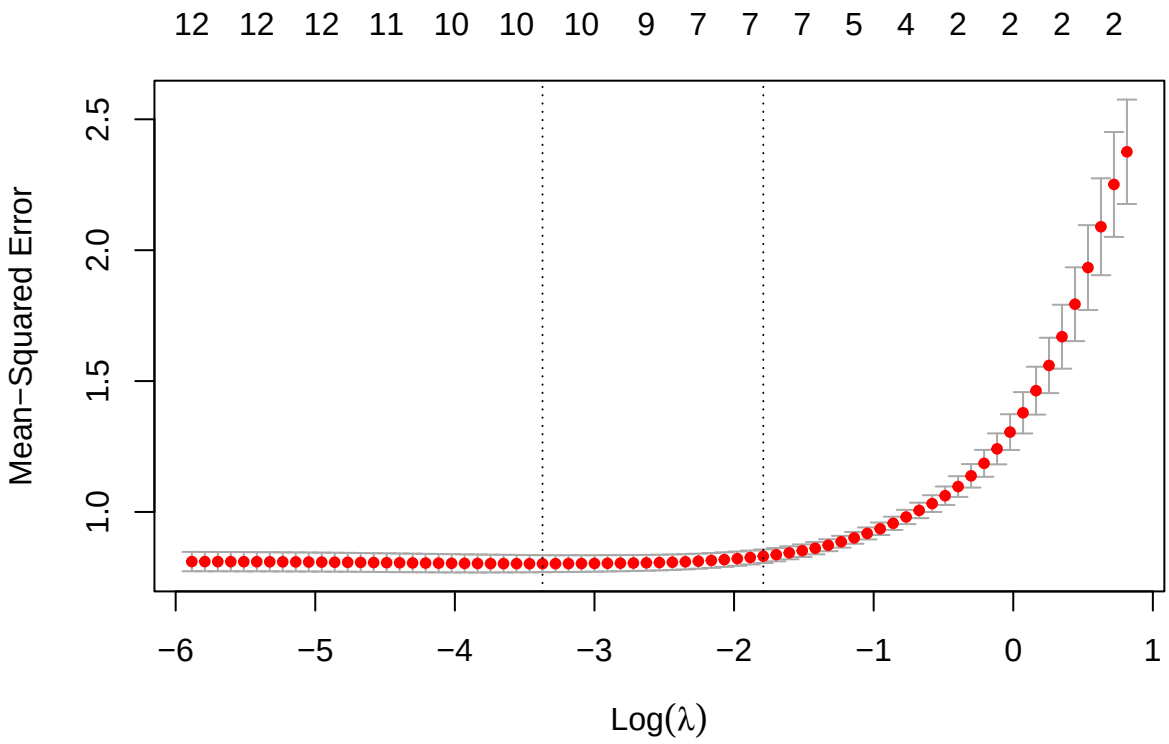
```
## 13 x 1 sparse Matrix of class "dgCMatrix"
##                              s1
## (Intercept)    -0.684762816
## fage            0.011666347
## mage            .
## mature          0.044177678
## weeks           0.193135446
```

```
## premie           .
## visits         -0.003838889
## marital         0.111631801
## gained          0.004166878
## lowbirthweight -2.331748254
## gender          0.304674079
## habit           0.111724964
## whitemom       -0.378687972
```

```
elnet.fit # find min lmbda and its lowest MSE
```

```
##
## Call:  cv.glmnet(x = trainx, y = trainy, alpha = 0.5)
##
## Measure: Mean-Squared Error
##
##      Lambda Index Measure      SE Nonzero
## min 0.03432    46  0.8024 0.03247      10
## 1se 0.16686    29  0.8308 0.02592       6
```

```
plot(elnet.fit)
```




```
min(elnet.fit$cvm) # lowest error for elastic net
```

```
## [1] 0.8024171
```

```
predict(elnet.fit, type="coefficients",s=elnet.fit$lambda.min) # 10 predictors
```

```
## 13 x 1 sparse Matrix of class "dgCMatrix"
```

```
##                         s1
## (Intercept)    -0.603913667
## fage            0.011293993
## mage            .
## mature          0.038652056
## weeks           0.191404069
## premie          .
## visits         -0.002491621
## marital         0.109439351
## gained          0.003950556
## lowbirthweight -2.307109481
## gender          0.296245171
## habit           0.098624029
## whitemom       -0.372207046
```

```
# predict for out of sample
## since lasso model gives lowest error, i will use it for testing out of sample
yhat.l <- predict(lasso.fit$glmnet.fit, s=lasso.fit$lambda.min, testx)
```

c

```
# reuse trainx,trainy,testx,testy

lmout <- lm(trainy ~ trainx)
yhat.r <- cbind(1,testx) %*% lmout$coefficients

#MSE for multiple regression
mean((testy - yhat.r)^2)
```

```
## [1] 0.8967002
```

```
# MSE for lasso regression
sum((testy - yhat.l)^2)/nrow(testx)
```

```
## [1] 0.8906549
```

```
## MSE from lasso regression model is a little lower than multiple regression model

#different a lot?
tss <- sum((testy - mean(testy))^2)
sse.reg <- sum((testy - yhat.r)^2)
sse.las <- sum((testy - yhat.l)^2)
r2.r <- (tss - sse.reg) / tss
r2.l <- (tss - sse.las) / tss
r2.r
```

```
## [1] 0.5862495
```

```
r2.l
```

```
## [1] 0.5890389
```

```
## R^2 like between two methods didnt differ a lot
```

d

```
#coef for elastic net model
predict(elnet.fit, type="coefficients",s=elnet.fit$lambda.min)
```

```
## 13 x 1 sparse Matrix of class "dgCMatrix"
```

```
##                         s1
## (Intercept)    -0.603913667
## fage            0.011293993
## mage            .
## mature          0.038652056
## weeks           0.191404069
## premie          .
## visits         -0.002491621
## marital         0.109439351
## gained          0.003950556
## lowbirthweight -2.307109481
## gender          0.296245171
## habit           0.098624029
## whitemom       -0.372207046
```

```
#coef for multiple regression model
coef(lmout)
```

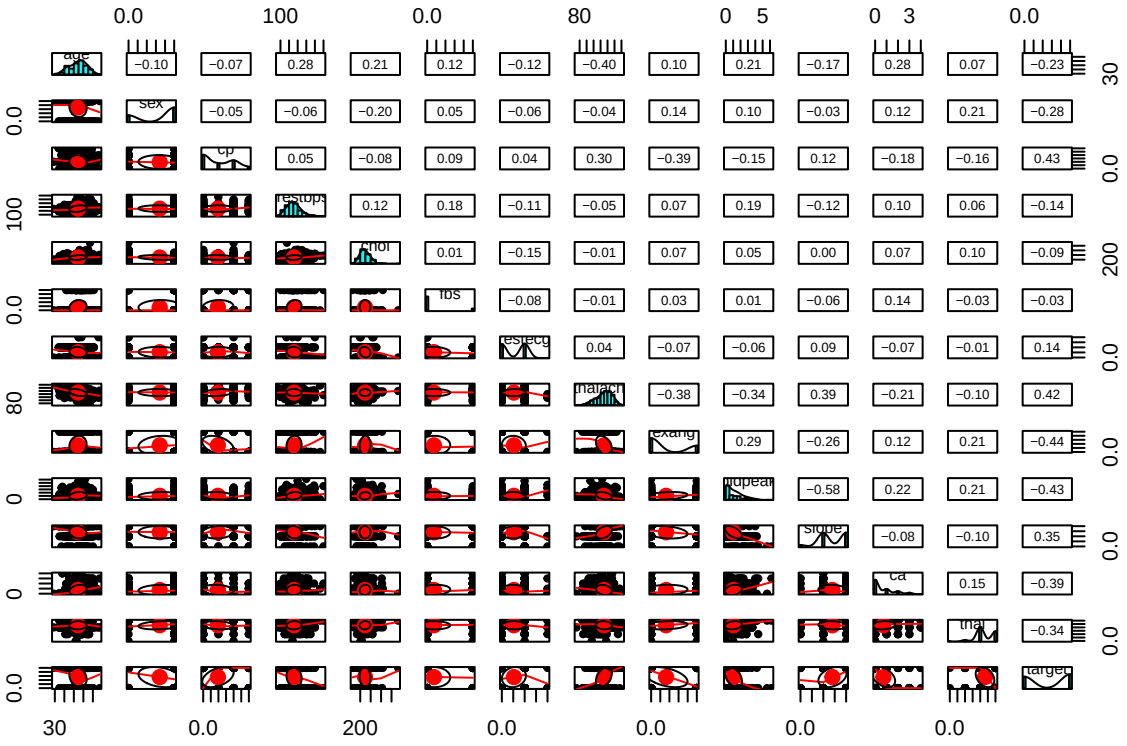
```
##          (Intercept)           trainxfage           trainxmage
##         -1.061911213          0.015045098         -0.004750372
##         trainxmature          trainxweeks         trainxpremie
##          0.111725561          0.204188846         -0.103823828
##         trainxvisits        trainxmarital         trainxgained
##         -0.008068464          0.144899456          0.005387025
## trainxlowbirthweight         trainxgender          trainxhabit
##         -2.373014901          0.328479922          0.170918753
##       trainxwhitemom
##         -0.394337225
```

Observation from coef of 2 models tells a fact that elastic model performs shrinkage on coef of mage and premie to 0 and thereby feature reduction.

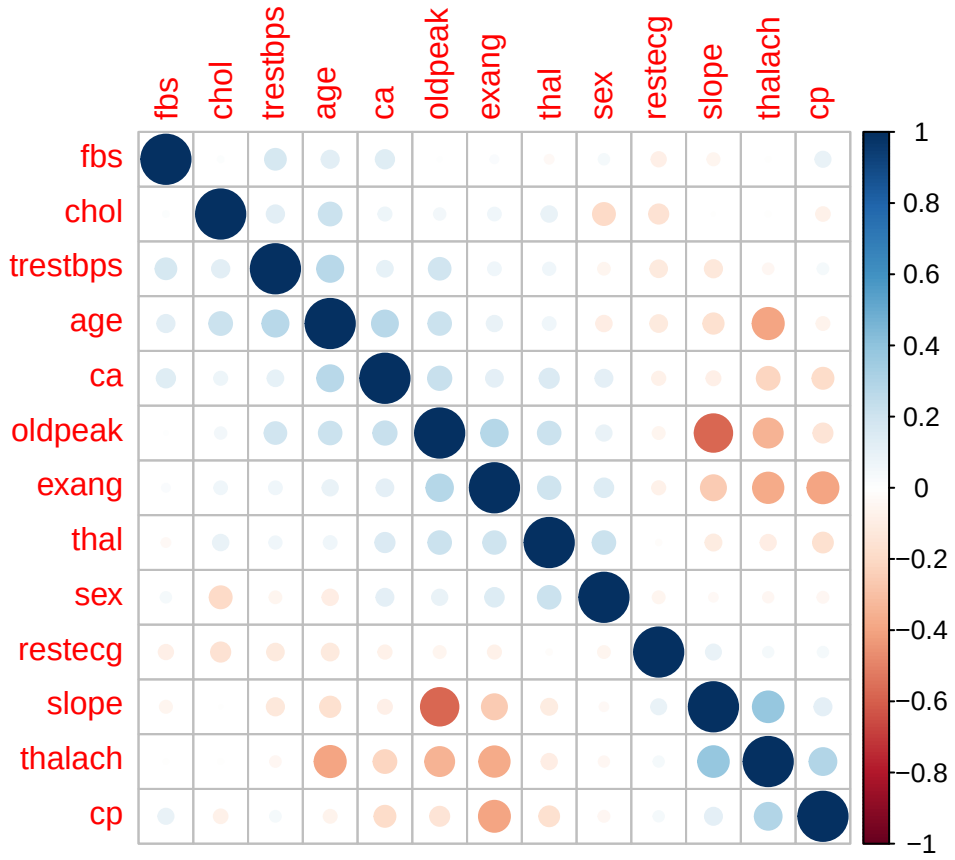
Q2

a

```
heart <- read.csv("heart.csv")
heart <- na.omit(heart)

#visualize correlation
pairs.panels(heart)
```




```
correlations <- cor(heart[,-14])
corrplot(correlations, method="circle", order = "AOE")
```




```
# first 13 variables are predictors for 14th variable, which stands for if presence of heart disease.
heart$target <- factor(heart$target)

set.seed(4321)
index.h <- sample(1:nrow(x),size=nrow(x)*0.5,replace = FALSE) #random selection of 50% data.

#train set and test set
train.h <- heart[index.h,] # 50% training data
test.h <- heart[-index.h,] # remaining 50% test data

# tune linaer kernel
costvalues <- 10^seq(-3,2,1)
svm.l <- tune(svm, target ~., data=train.h, ranges=list(cost=costvalues), kernel="linear")
svm.l
```

```
##
## Parameter tuning of 'svm':
##
## - sampling method: 10-fold cross validation
##
## - best parameters:
##  cost
##   0.1
##
## - best performance: 0.1697249
```

```r
# best cost = 0.01
svm.l$best.model
```

```
##
## Call:
## best.tune(method = svm, train.x = target ~ ., data = train.h, ranges = list(cost = costvalues),
##     kernel = "linear")
##
##
## Parameters:
##    SVM-Type:  C-classification
##  SVM-Kernel:  linear
##        cost:  0.1
##
## Number of Support Vectors:  63
```

```r
# linear kernel in smaple accuracy
yhat.in.l <- predict(svm.l$best.model, train.h)
sum(yhat.in.l == train.h$target)/length(train.h$target)
```

```
## Warning in `==.default`(yhat.in.l, train.h$target): longer object length is not
## a multiple of shorter object length
```

```
## Warning in is.na(e1) | is.na(e2): longer object length is not a multiple of
## shorter object length
```

```
## [1] NA
```

```r
# tune radial kernel
costvalues <- 10^seq(-3,2,1)
svm.r <- tune(svm, target ~., data=train.h, ranges=list(cost=costvalues), kernel="radial")

# best cost = 0.1
svm.r$best.model
```

```
##
## Call:
## best.tune(method = svm, train.x = target ~ ., data = train.h, ranges = list(cost = costvalues),
##     kernel = "radial")
##
##
## Parameters:
##    SVM-Type:  C-classification
##  SVM-Kernel:  radial
##        cost:  1
##
## Number of Support Vectors:  91
```

```r
# radial kernel in smaple accuracy
yhat.in.r <- predict(svm.r$best.model)
sum(yhat.in.r == train.h$target)/length(train.h$target)
```

```
## Warning in `==.default`(yhat.in.r, train.h$target): longer object length is not
## a multiple of shorter object length

## Warning in `==.default`(yhat.in.r, train.h$target): longer object length is not
## a multiple of shorter object length
```

```
## [1] NA
```

Thus, radial kernel outperforms linear kernel for in sample accuracy of 0.847682 with a cost=0.1 compared to 0.8278146 with cost=0.01 from linear kernel.

b

```
# choose radial kernel predict with test, accuracy since it has higher in sample accuracy
yhat.out.r <- predict(svm.r$best.model, test.h)
table(yhat.out.r, test.h$target)
```

```
##
## yhat.out.r  0  1
##          0 54  6
##          1 27 76
```

```
sum(yhat.out.r == test.h$target)/length(test.h$target)
```

```
## [1] 0.797546
```

Compared to in sample accuracy, out of sample accuracy is lower, which is expected