Final Project

Ma.Xiaoran

2020/3/7

```r
library(MASS)
library(boot)
library(AER)
```

```
## Warning: package 'AER' was built under R version 3.6.3
```

```
## Loading required package: car
```

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

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

```
## The following object is masked from 'package:boot':
##
##     logit
```

```
## Loading required package: lmtest
```

```
## Warning: package 'lmtest' was built under R version 3.6.3
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: sandwich
```

```
## Loading required package: survival
```

```
##
## Attaching package: 'survival'
```

```
## The following object is masked from 'package:boot':
##
##     aml
```

```r
require(pscl)
```

```
## Loading required package: pscl
```

```
## Warning: package 'pscl' was built under R version 3.6.3
```

```
## Classes and Methods for R developed in the
## Political Science Computational Laboratory
## Department of Political Science
## Stanford University
## Simon Jackman
## hurdle and zeroinfl functions by Achim Zeileis
```

```r
library(aod)
```

```
## Warning: package 'aod' was built under R version 3.6.3

##
## Attaching package: 'aod'

## The following object is masked from 'package:survival':
##
##     rats
```

```r
library(faraway)
```

```
## Warning: package 'faraway' was built under R version 3.6.3

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

##
## Attaching package: 'faraway'

## The following objects are masked from 'package:aod':
##
##     rats, salmonella

## The following objects are masked from 'package:survival':
##
##     rats, solder

## The following objects are masked from 'package:car':
##
##     logit, vif

## The following objects are masked from 'package:boot':
##
##     logit, melanoma
```

```r
library(car)
```

```r
p <- read.table("pharmacist.txt",header = T)
str(p)
```

```
## 'data.frame':    500 obs. of  12 variables:
##  $ pc    : int  0 0 0 0 0 0 0 0 7 0 ...
##  $ sex   : int  1 1 0 0 0 0 0 0 1 1 ...
##  $ age   : num  0.19 0.72 0.47 0.27 0.19 0.72 0.62 0.37 0.27 0.57 ...
##  $ income: num  0.45 0.25 1.3 0.9 0.15 0.45 0.25 1.1 0.9 0.35 ...
##  $ lp    : int  1 0 1 1 0 0 0 1 1 0 ...
##  $ fp    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ fr    : int  0 1 0 0 0 1 1 0 0 1 ...
##  $ ill   : int  0 2 2 0 2 3 3 0 1 2 ...
##  $ ad    : int  0 14 0 0 0 0 0 0 3 0 ...
##  $ hs    : int  0 5 1 0 5 0 2 0 0 0 ...
##  $ ch1   : int  0 0 0 0 0 1 0 1 0 1 ...
##  $ ch2   : int  0 1 1 0 0 0 1 0 0 0 ...
```

```r
summary(p)
```

```
##        pc           sex              age              income      
##  Min.   :0.0   Min.   :0.000   Min.   :0.1900   Min.   :0.0000  
##  1st Qu.:0.0   1st Qu.:0.000   1st Qu.:0.2200   1st Qu.:0.2500  
##  Median :0.0   Median :1.000   Median :0.3200   Median :0.5500  
##  Mean   :0.3   Mean   :0.516   Mean   :0.4091   Mean   :0.5718  
##  3rd Qu.:0.0   3rd Qu.:1.000   3rd Qu.:0.6200   3rd Qu.:0.7500  
##  Max.   :8.0   Max.   :1.000   Max.   :0.7200   Max.   :1.5000  
##        lp              fp               fr              ill      
##  Min.   :0.000   Min.   :0.000   Min.   :0.000   Min.   :0.00  
##  1st Qu.:0.000   1st Qu.:0.000   1st Qu.:0.000   1st Qu.:0.00  
##  Median :0.000   Median :0.000   Median :0.000   Median :1.00  
##  Mean   :0.402   Mean   :0.048   Mean   :0.238   Mean   :1.42  
##  3rd Qu.:1.000   3rd Qu.:0.000   3rd Qu.:0.000   3rd Qu.:2.00  
##  Max.   :1.000   Max.   :1.000   Max.   :1.000   Max.   :5.00  
##        ad               hs              ch1             ch2       
##  Min.   : 0.000   Min.   : 0.000   Min.   :0.000   Min.   :0.000  
##  1st Qu.: 0.000   1st Qu.: 0.000   1st Qu.:0.000   1st Qu.:0.000  
##  Median : 0.000   Median : 0.000   Median :0.000   Median :0.000  
##  Mean   : 0.934   Mean   : 1.116   Mean   :0.412   Mean   :0.118  
##  3rd Qu.: 0.000   3rd Qu.: 1.000   3rd Qu.:1.000   3rd Qu.:0.000  
##  Max.   :14.000   Max.   :12.000   Max.   :1.000   Max.   :1.000  
```

```
## check data consistency
sum((p$lp+p$fp+p$fr)>1) # data is consistent
```

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

```
sum((p$ch1+p$ch2)>1) # data is consistent
```

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

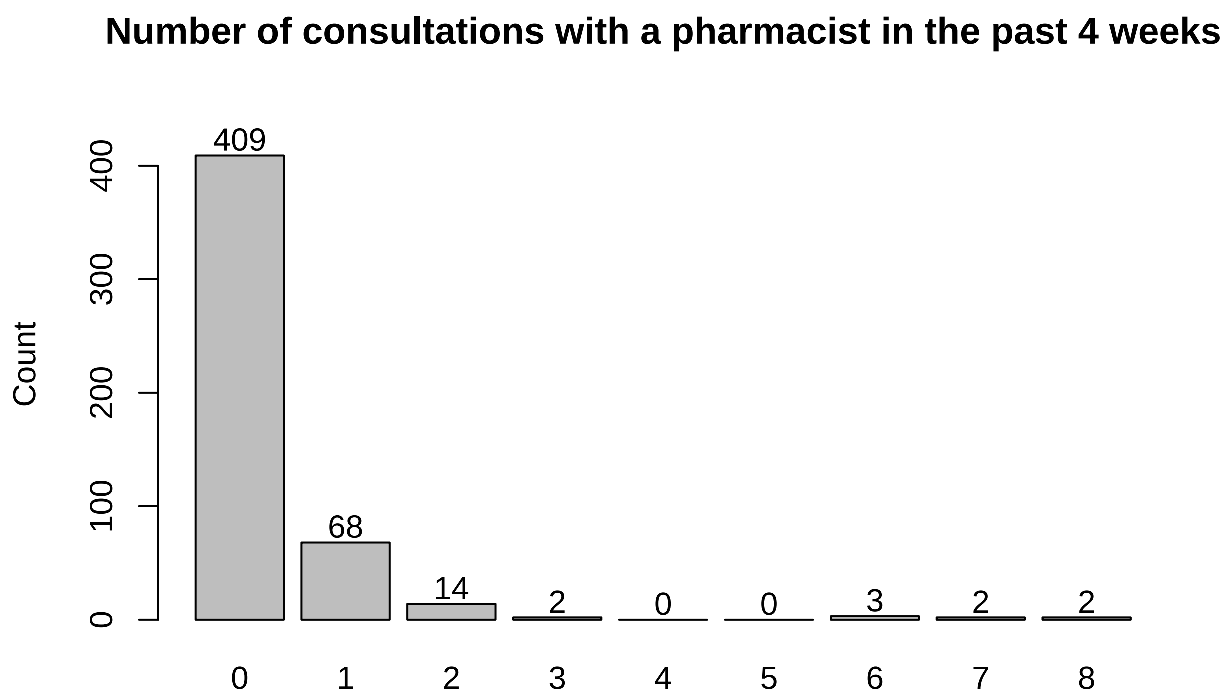
```
## combine lp,fp,fr variable and ch variables
p$insurance <- as.factor(ifelse(p$lp==1,1,ifelse(p$fp==1,2,ifelse(p$fr==1,3,4))))
p$ch <- as.factor(ifelse(p$ch1==1,1,ifelse(p$ch2==1,2,3)))


p2 <- p[,-c(5:7,11,12)]
str(p2)
```

```
## 'data.frame':    500 obs. of  9 variables:
##  $ pc       : int  0 0 0 0 0 0 0 0 7 0 ...
##  $ sex      : int  1 1 0 0 0 0 0 0 1 1 ...
##  $ age      : num  0.19 0.72 0.47 0.27 0.19 0.72 0.62 0.37 0.27 0.57 ...
##  $ income   : num  0.45 0.25 1.3 0.9 0.15 0.45 0.25 1.1 0.9 0.35 ...
##  $ ill      : int  0 2 2 0 2 3 3 0 1 2 ...
##  $ ad       : int  0 14 0 0 0 0 0 0 3 0 ...
##  $ hs       : int  0 5 1 0 5 0 2 0 0 0 ...
##  $ insurance: Factor w/ 4 levels "1","2","3","4": 1 3 1 1 4 3 3 1 1 3 ...
##  $ ch       : Factor w/ 3 levels "1","2","3": 3 2 2 3 3 1 2 1 3 1 ...
```

make a table: no missing value. The dependent variable is count; factor: sex, insurance, ch; count: pc, ill, ad; numeric: hs,age, income.

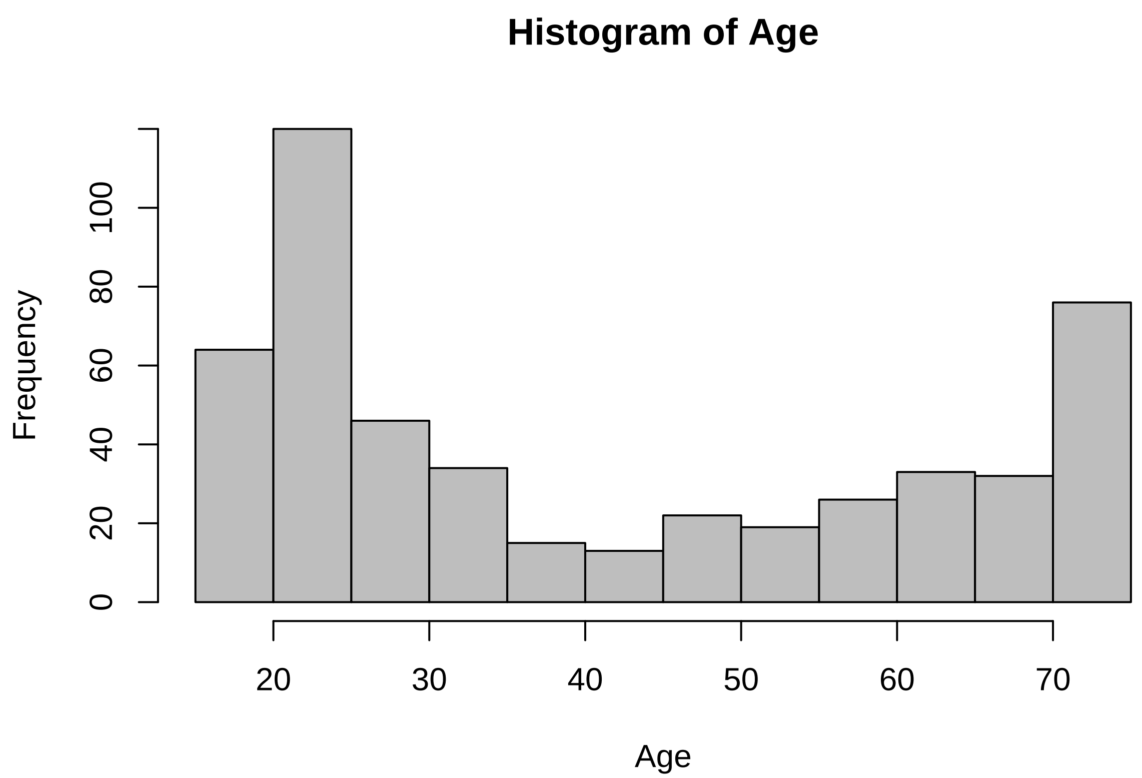
```
### EDA for each covariates
attach(p2)
newtable <- c(table(pc)[1:4],0,0,table(pc)[5:7])
names(newtable) <- c(0:8)
```

```r
x <- barplot(newtable,main = "Number of consultations with a pharmacist in the past 4 weeks",ylim=c(0,45
y <- newtable
text(x,y+14,labels=as.character(y))
```

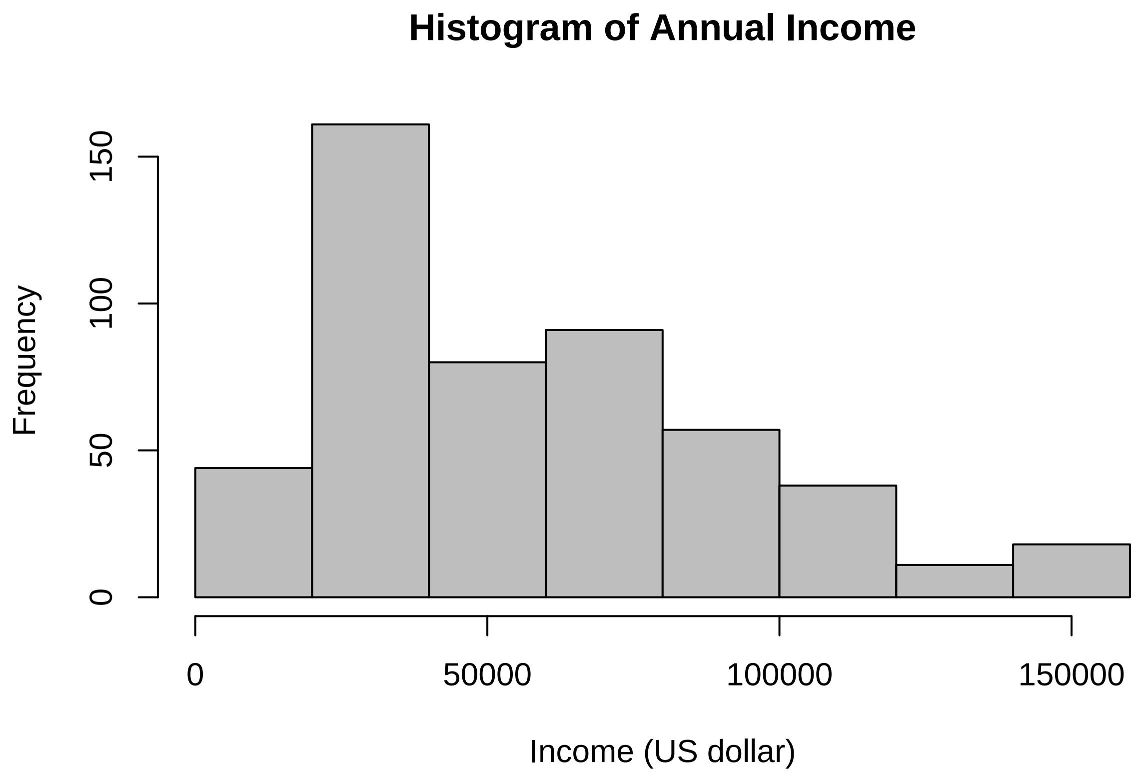



```r
x <- barplot(table(sex),main = "Sex",ylim=c(0,300), names.arg = c("male","female"),ylab="Count")
y <- table(sex)
text(x,y+14,labels=as.character(y))
```

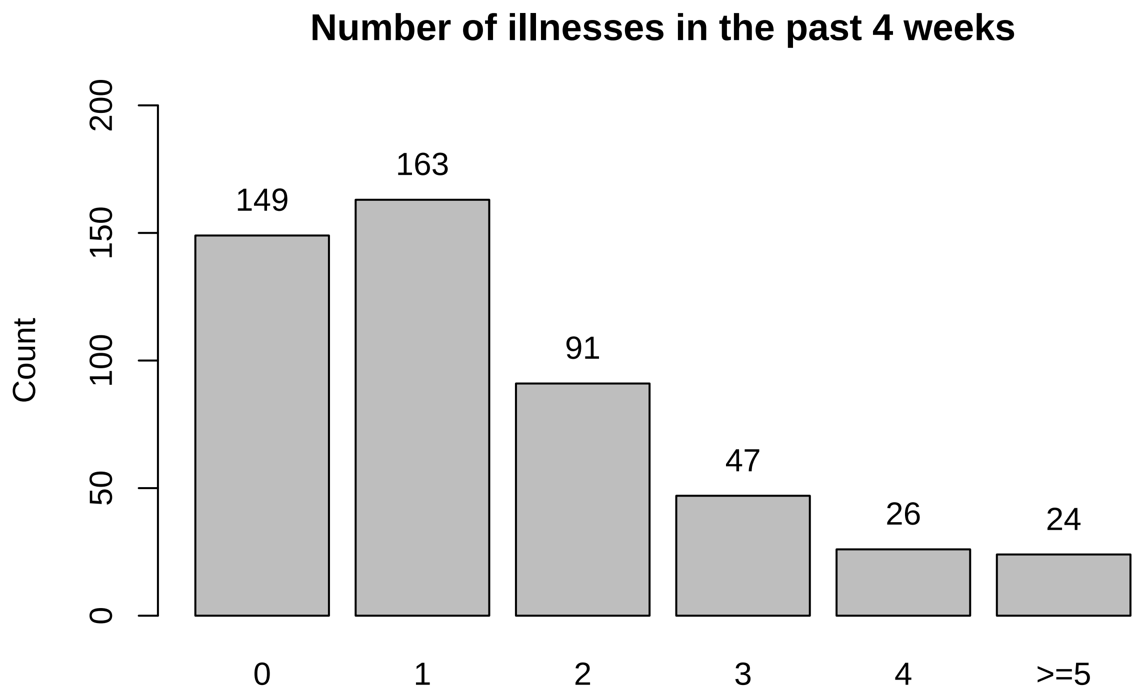

```
hist(age*100, col="gray", breaks = "Sturges", main="Histogram of Age",xlab = "Age")
```



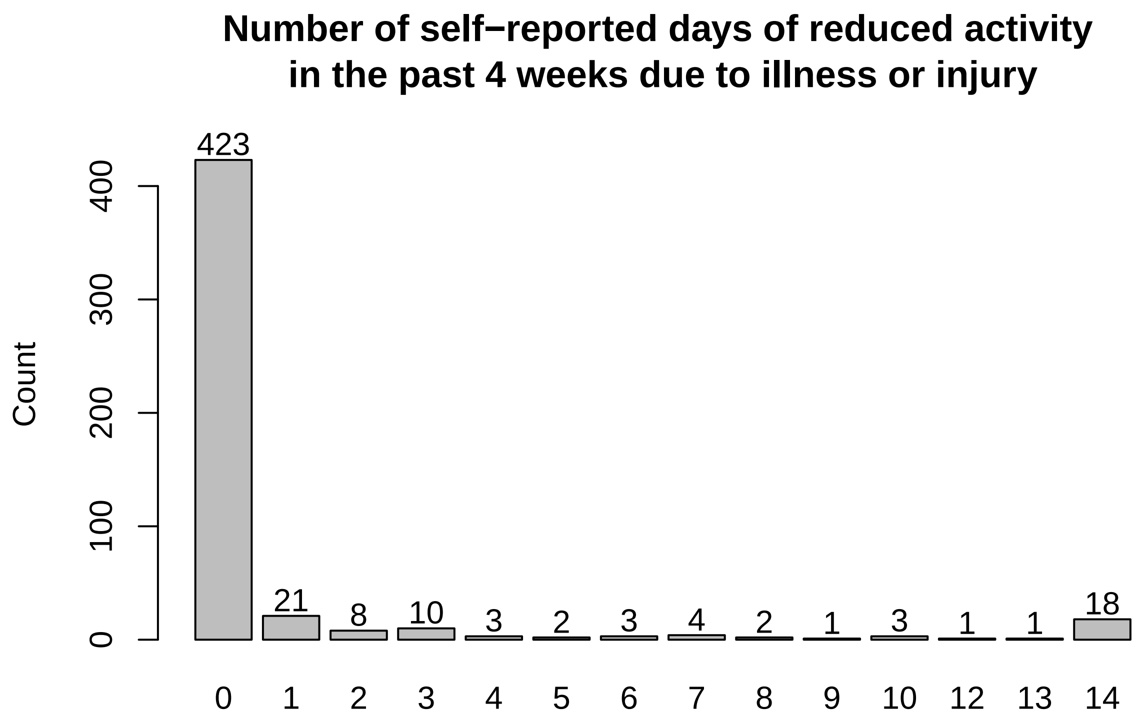

```
hist(income*100000, col="gray", breaks = "Sturges", main="Histogram of Annual Income",xlab = "Income (US
```



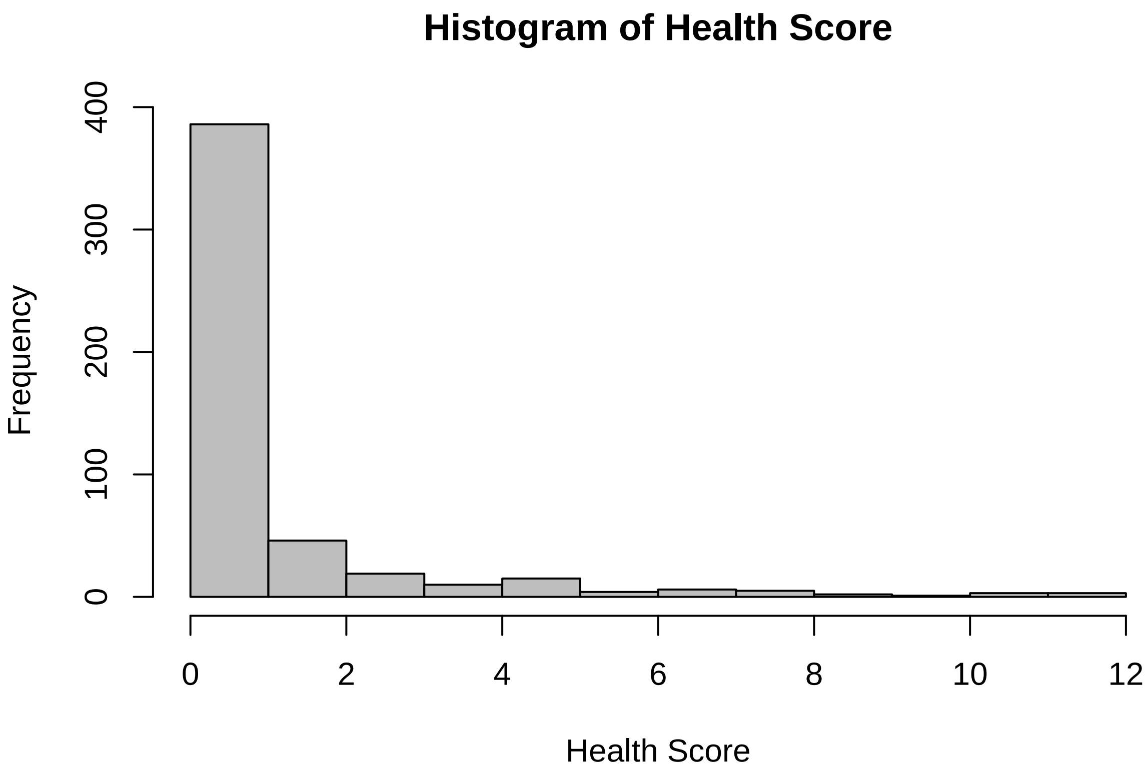

```
x <- barplot(table(ill),main = "Number of illnesses in the past 4 weeks",ylim=c(0,200),ylab="Count",name
y <- table(ill)
text(x,y+14,labels=paste(as.character(y)))
```



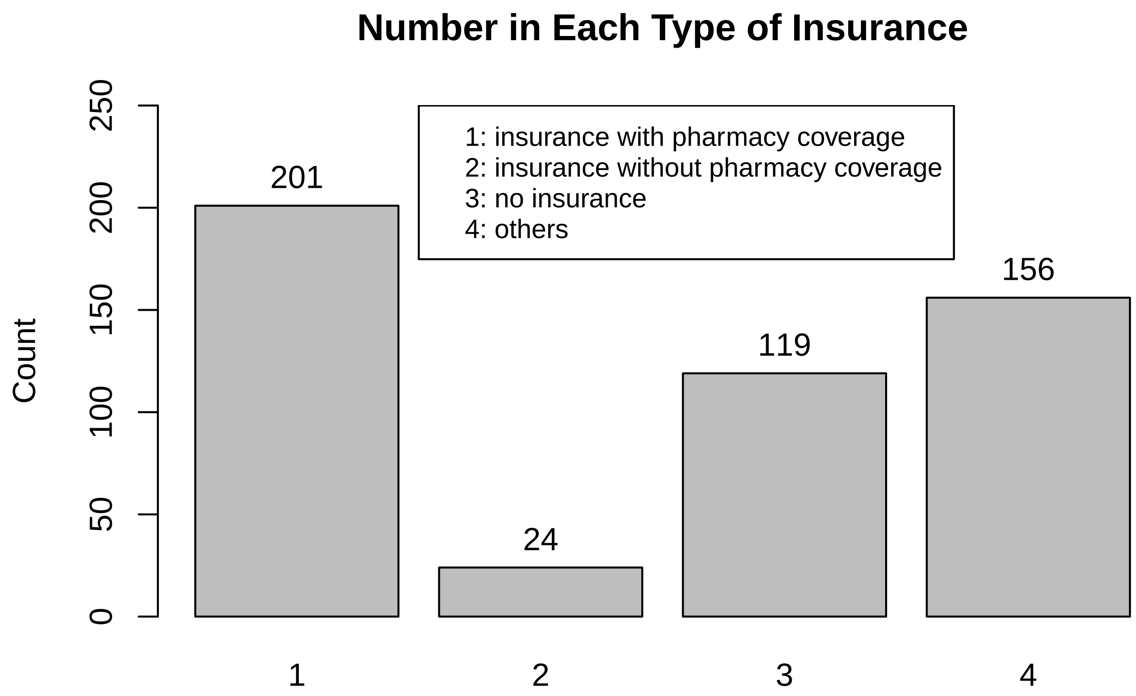

```
x <- barplot(table(ad),main = paste("Number of self-reported days of reduced activity","\nin the past 4
y <- table(ad)
text(x,y+14,labels=as.character(y))
```



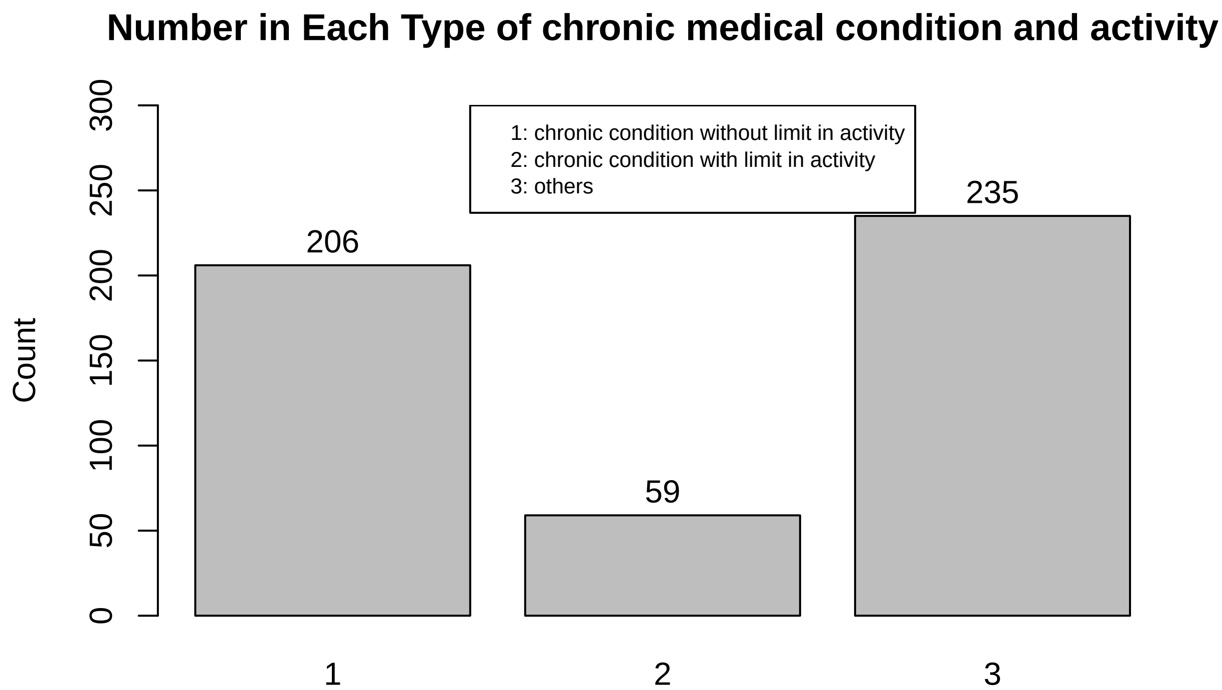

```r
hist(hs, col="gray", breaks = "Sturges", main="Histogram of Health Score",xlab = "Health Score")
```




```
x <- barplot(table(insurance),main = "Number in Each Type of Insurance",ylim=c(0,250),ylab="Count")
legend(1.3, 250, legend=c("1: insurance with pharmacy coverage","2: insurance without pharmacy coverage
y <- table(insurance)
text(x,y+14,labels=as.character(y))
```




```
x <- barplot(table(ch),main = "Number in Each Type of chronic medical condition and activity",ylab="Cou
legend(1.2, 300, legend=c("1: chronic condition without limit in activity","2: chronic condition with l
y <- table(ch)
text(x,y+14,labels=as.character(y))
```




```
detach(p2)
```

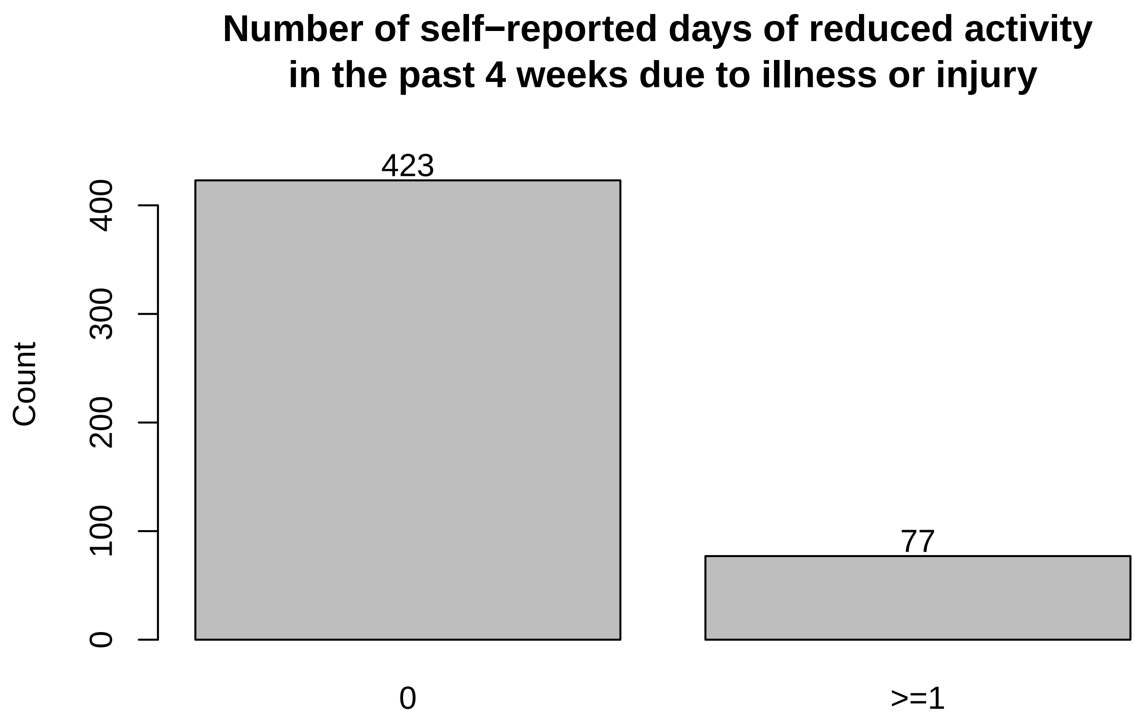
Change "Number of self-reported days of reduced activity in the past 4 weeks due to illness or injury" variable to be 0-1 where 0: no reduced activity and 1: reduced activity due to illness or injury.

'## Some of the covariates have too many levels (here "level" refers to both the level in categorical variable and count variable/discrete variable) and some levels have too few observations. I combined some of the levels in each independent variable. The reason is that I want to balance between Bias and Variance (in the sence of data science). In other words, if the model has too many explanatory variables, it tends to overfitting, capturing the noise along with the underlying pattern in data (low bias and high variance). If the the model has too few explanatory variables, it may be underfitting, unable to capture the underlying pattern of the data (high bias and low variance). Since our dataset is relatively small (500 observations), too many covariates may lead to overfitting.

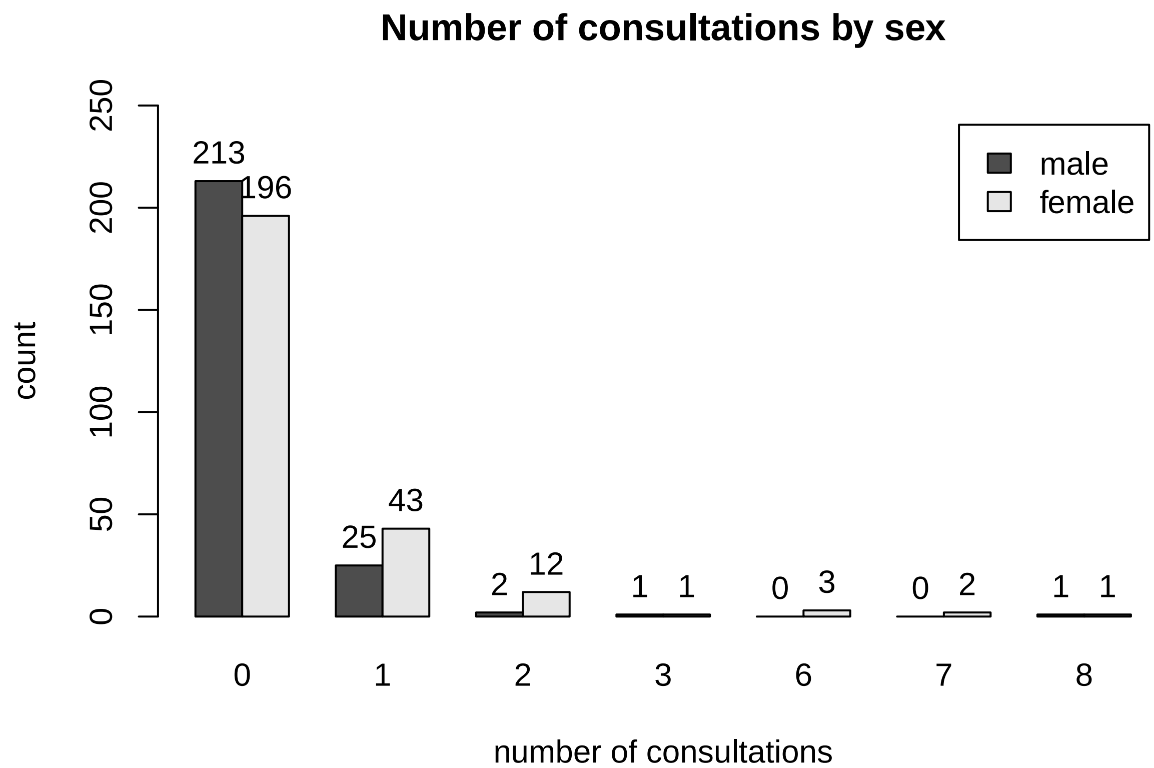
In light of covariates, we want each of them to contain as much information of dependent variable as possible. If a covariates is almost a constant, it merely explans the dependent variable. Therefore, for those levels which contains comparatively small number of observations, I combine them together.

In our data, I "remove" some levels of categorical data by combining some of them as one single level. It makes sense because they contain too few observations to be added as explanatory variable. '##

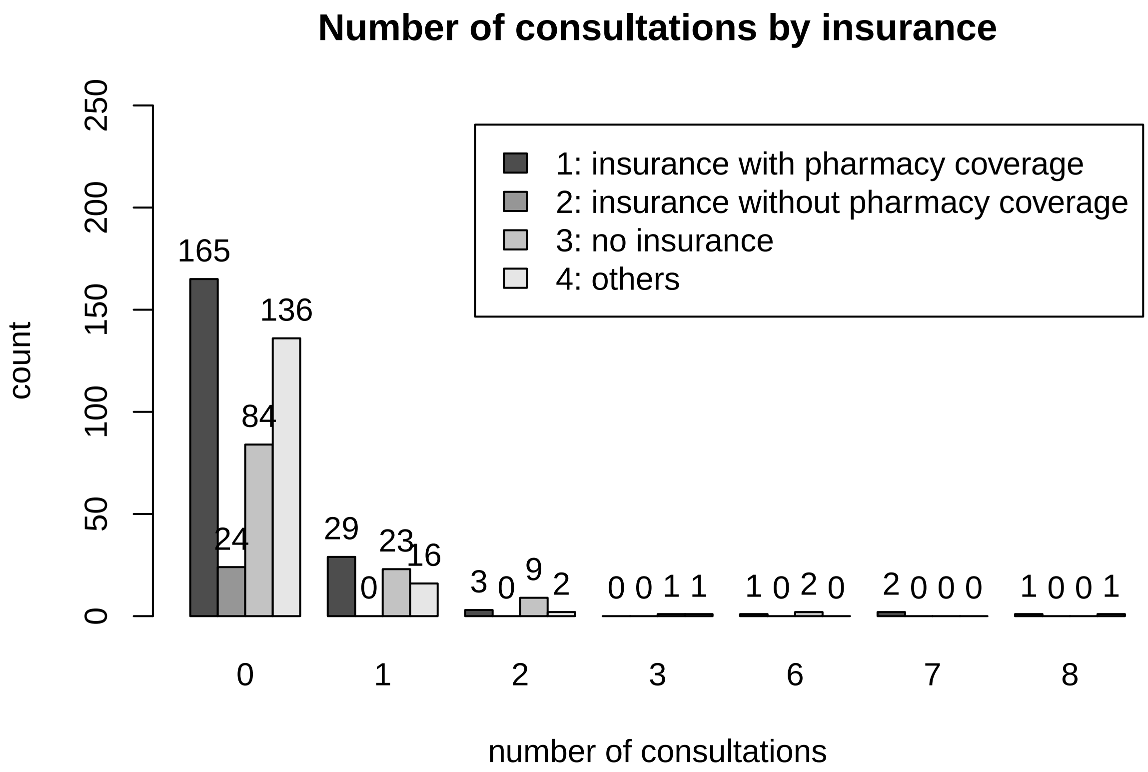
```
### change variables to 0-1
p3 <- p2
p3$ad <- ifelse(p3$ad==0,0,1)

x <- barplot(table(p3$ad),main = paste("Number of self-reported days of reduced activity","\nin the pas
y <- table(p3$ad)
text(x,y+14,labels=as.character(y))
```



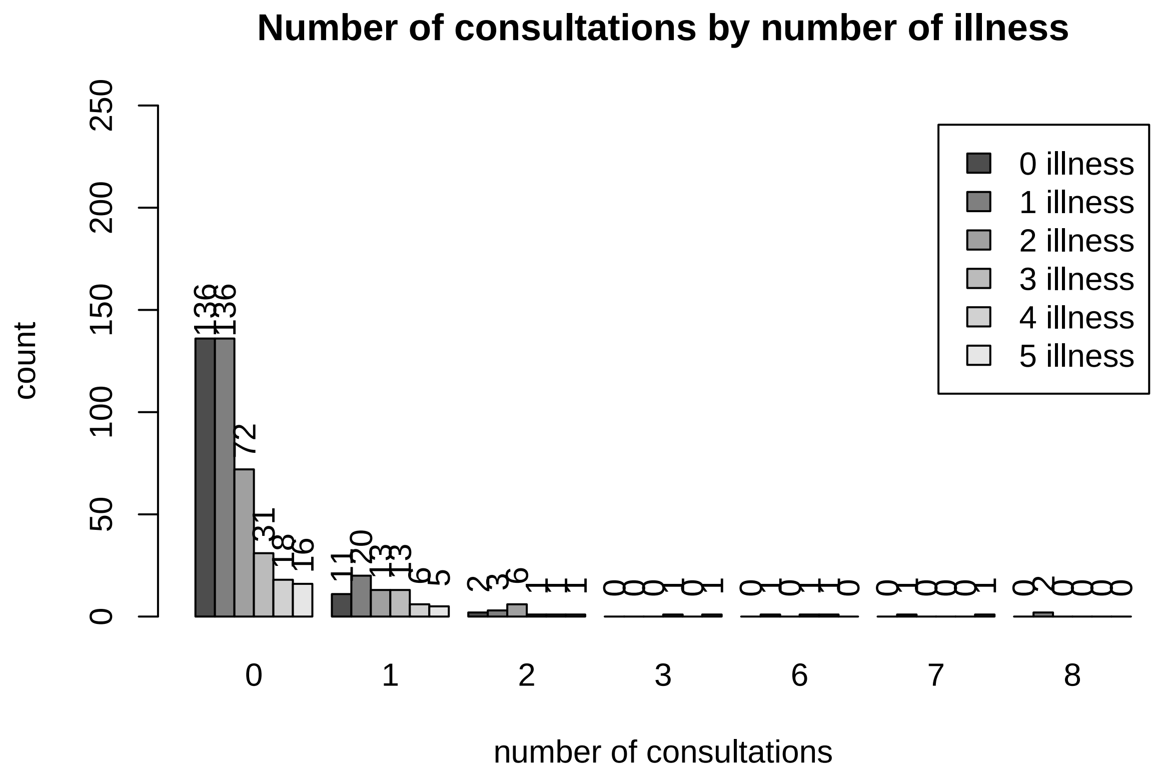

```
### EDA of covariates vs dependent variable

## sex
pc.sex <- table(p3$pc,p3$sex)
x <- barplot(t(pc.sex),beside = T, main="Number of consultations by sex",legend.text = c("male","female
text(x[1,],pc.sex[,1]+14,labels=as.character(pc.sex[,1]))
text(x[2,],pc.sex[,2]+14,labels=as.character(pc.sex[,2]))
```



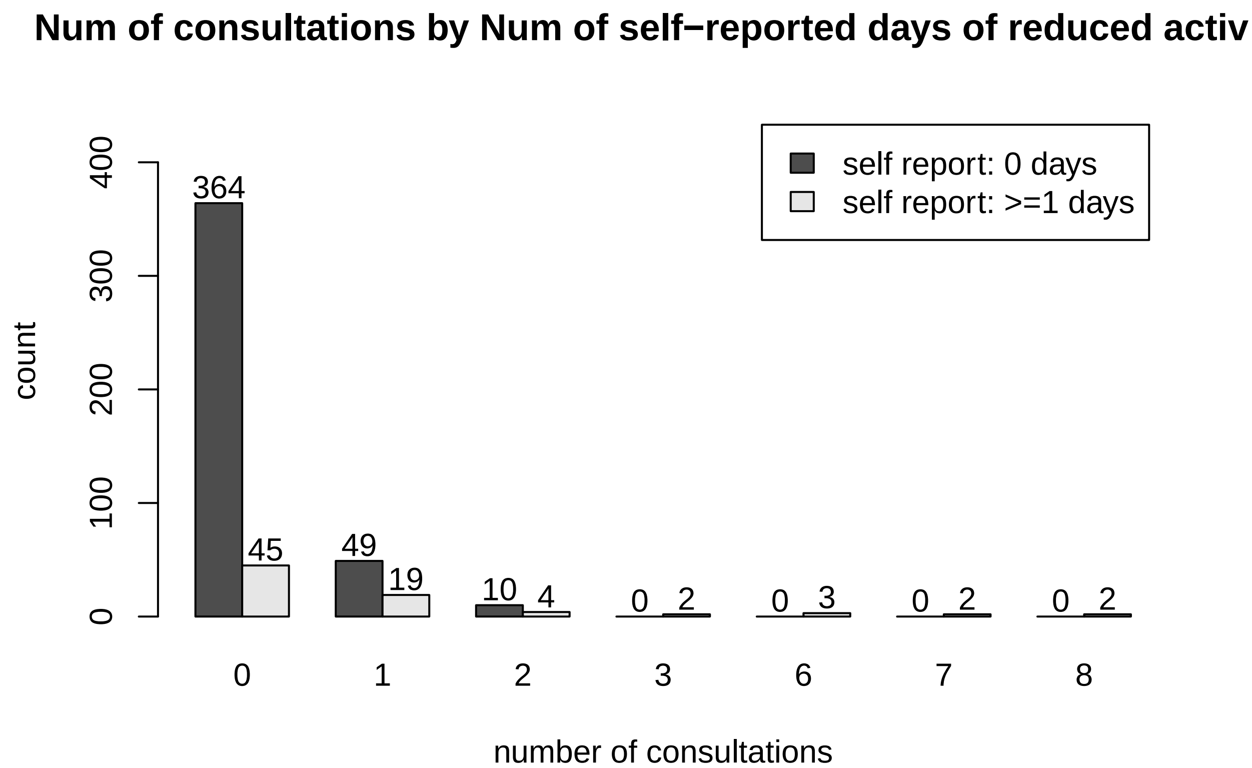

```
## insurance
pc.ins <- table(p3$pc,p3$insurance)
x <- barplot(t(pc.ins),beside = T, main="Number of consultations by insurance",legend.text = c("1: insu
text(x[1,],pc.ins[,1]+14,labels=as.character(pc.ins[,1]))
text(x[2,],pc.ins[,2]+14,labels=as.character(pc.ins[,2]))
text(x[3,],pc.ins[,3]+14,labels=as.character(pc.ins[,3]))
text(x[4,],pc.ins[,4]+14,labels=as.character(pc.ins[,4]))
```



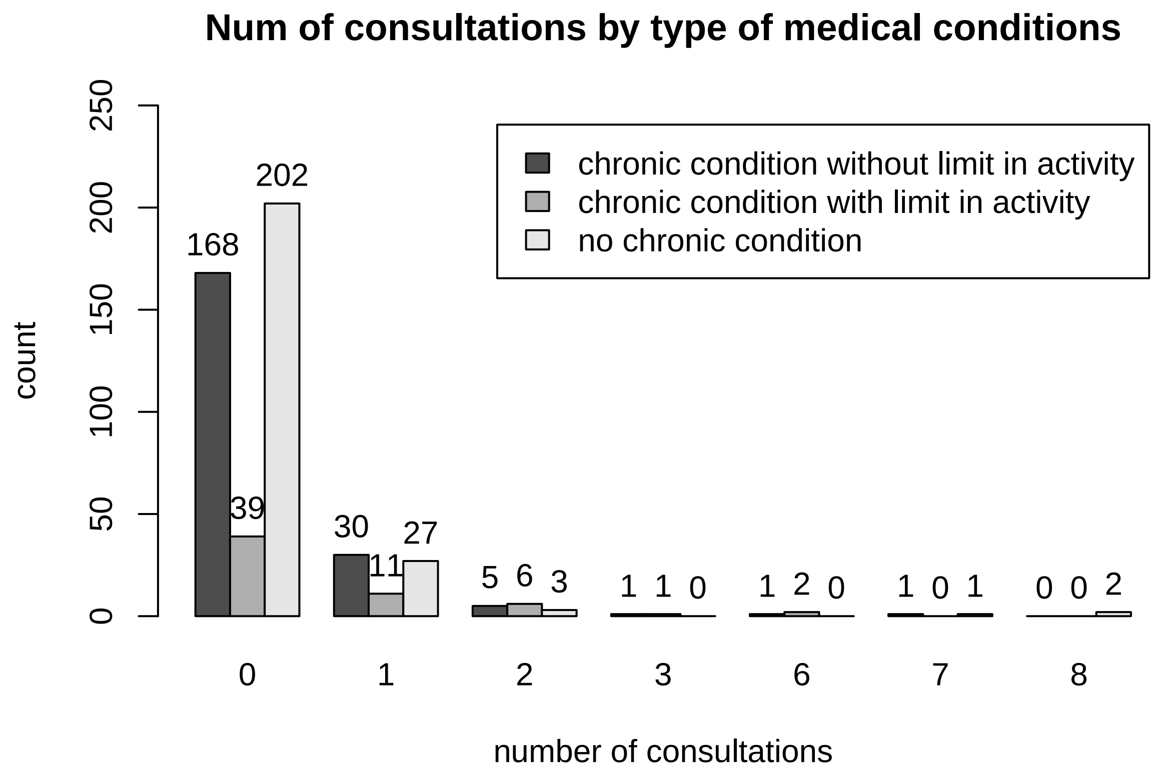

```
## illness
pc.ill <- table(p3$pc,p3$ill)
x <- barplot(t(pc.ill),beside = T, main="Number of consultations by number of illness",ylim=c(0,250),yla
text(x[1,],pc.ill[,1]+14,labels=as.character(pc.ill[,1]),srt=90)
text(x[2,],pc.ill[,2]+14,labels=as.character(pc.ill[,2]),srt=90)
text(x[3,],pc.ill[,3]+14,labels=as.character(pc.ill[,3]),srt=90)
text(x[4,],pc.ill[,4]+14,labels=as.character(pc.ill[,4]),srt=90)
text(x[5,],pc.ill[,5]+14,labels=as.character(pc.ill[,5]),srt=90)
text(x[6,],pc.ill[,6]+14,labels=as.character(pc.ill[,6]),srt=90)
```



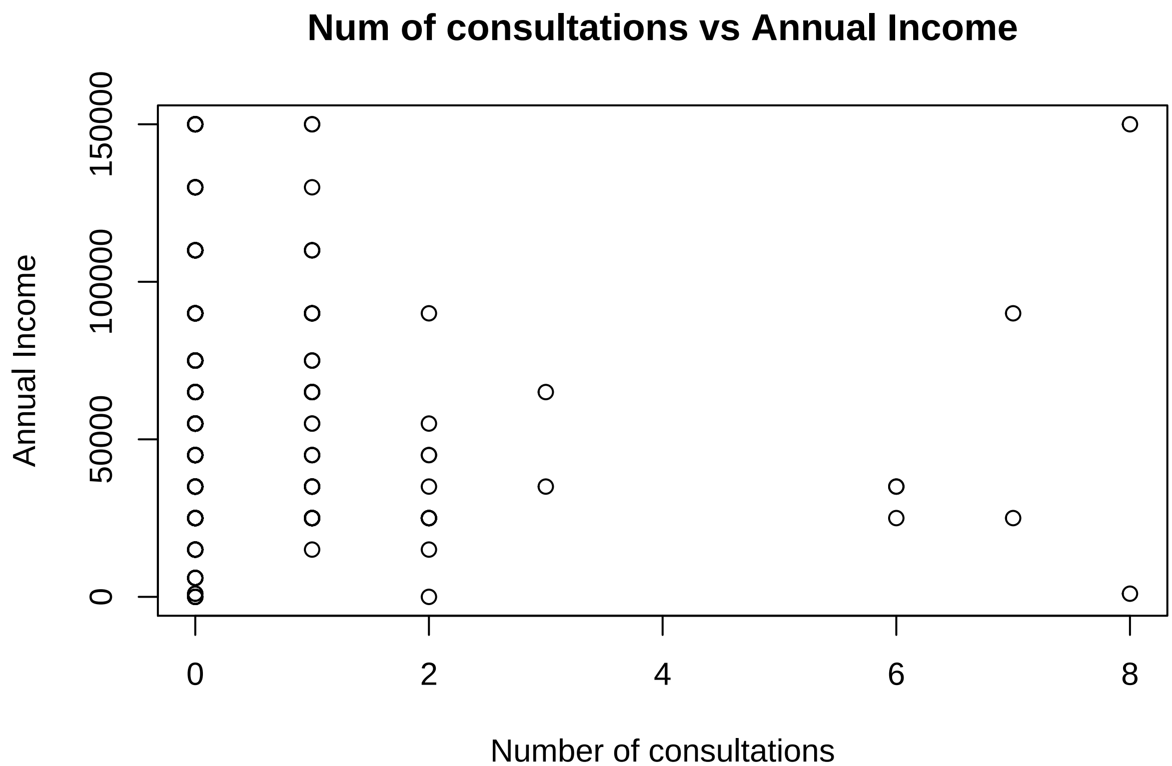

```
## self report days
pc.ad <- table(p3$pc,p3$ad)
x <- barplot(t(pc.ad),beside = T, main="Num of consultations by Num of self-reported days of reduced act
text(x[1,],pc.ad[,1]+14,labels=as.character(pc.ad[,1]))
text(x[2,],pc.ad[,2]+14,labels=as.character(pc.ad[,2]))
```



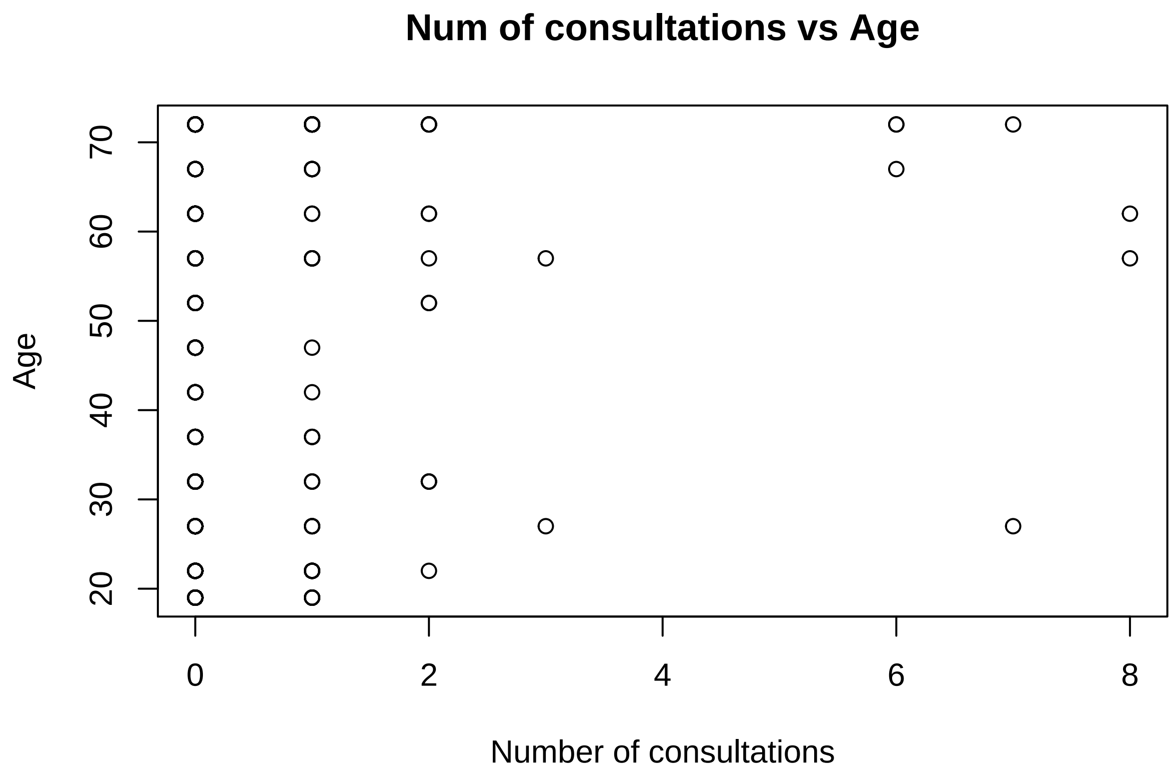

```
## ch
pc.ch <- table(p3$pc,p3$ch)
x <- barplot(t(pc.ch),beside = T, main="Num of consultations by type of medical conditions",legend.text
text(x[1,],pc.ch[,1]+14,labels=as.character(pc.ch[,1]))
text(x[2,],pc.ch[,2]+14,labels=as.character(pc.ch[,2]))
text(x[3,],pc.ch[,3]+14,labels=as.character(pc.ch[,3]))
```



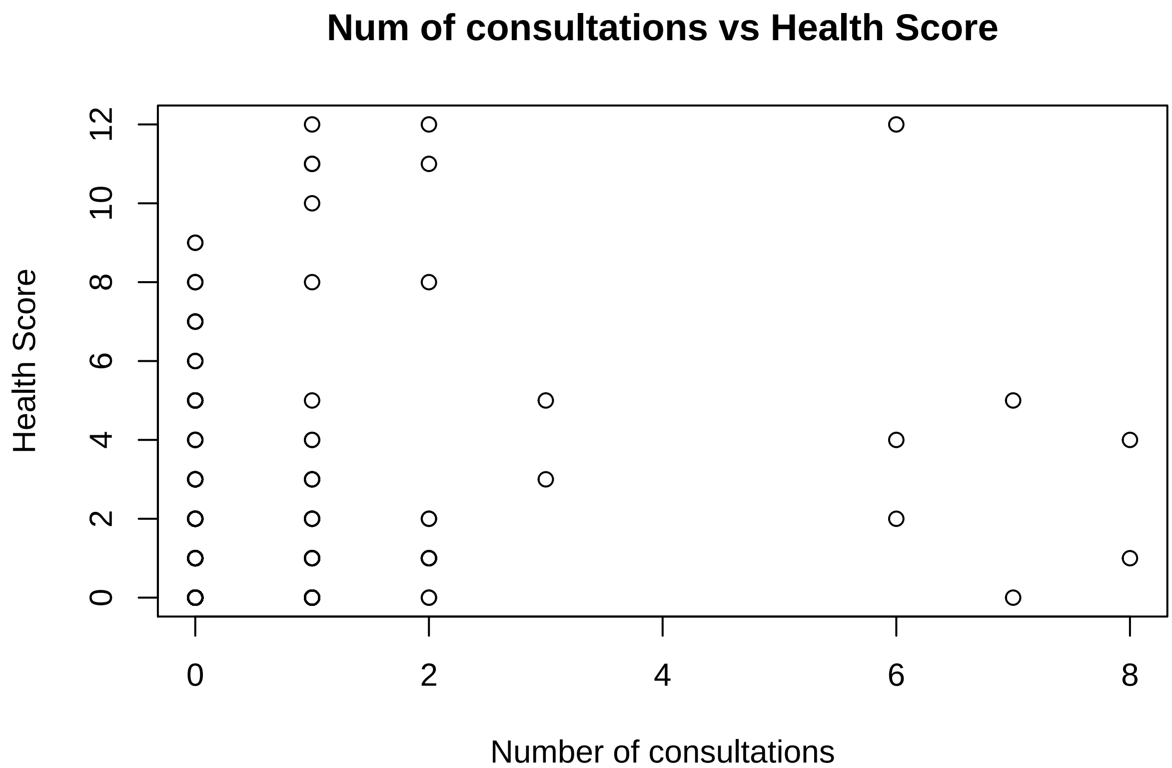

```
## income
plot(p3$pc,p3$income*100000, main="Num of consultations vs Annual Income", ylab="Annual Income",xlab="Nu
```




```
## age
plot(p3$pc,p3$age*100, main="Num of consultations vs Age", ylab="Age",xlab="Number of consultations")
```




```
## score
plot(p3$pc,p3$hs, main="Num of consultations vs Health Score", ylab="Health Score",xlab="Number of consu
```



vs sex: It can be seen that sex may have influence on our dependent variable since female tend to have more consultations than male.

vs insurance: It can be seen that the majority of those who did not have a consultation have insurance with pharmacy coverage. The number of consultations varies among these 4 groups of people which indicates that the type of insurance could be an influential covariate.

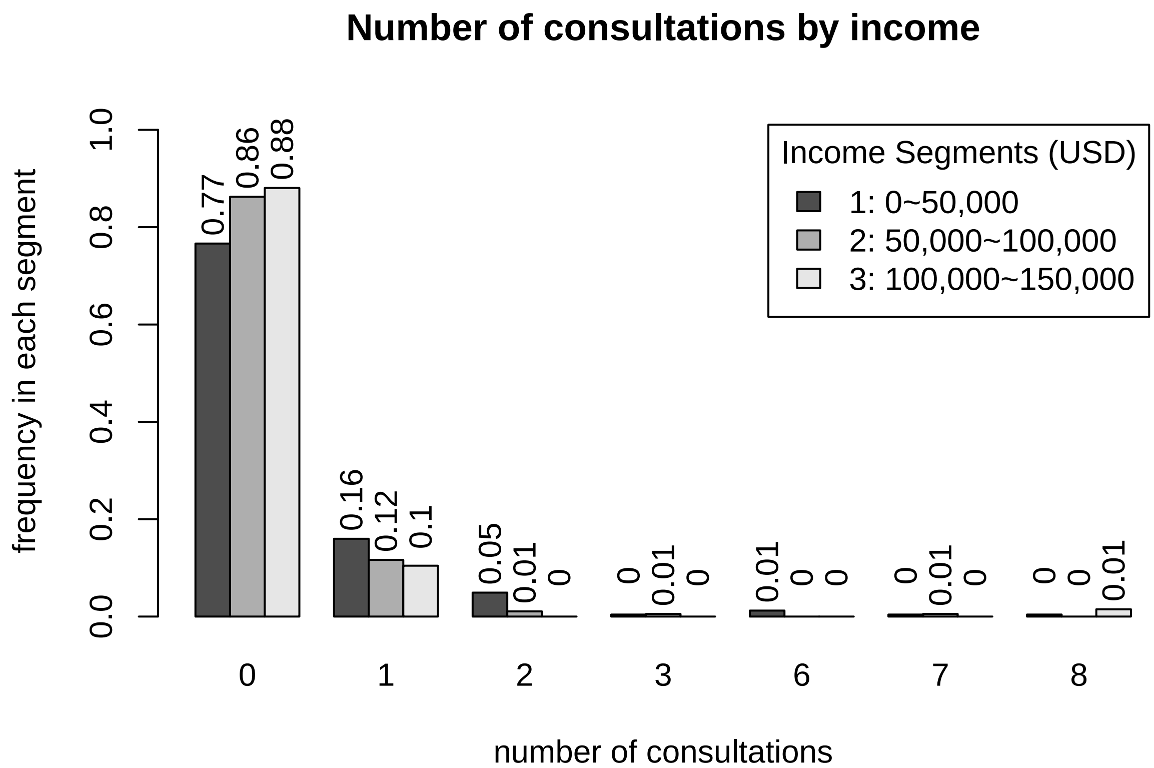
vs num of illness: Number of consultations also differs among different number of illness.

vs self report days: significant difference between 0 days and at least 1 days.

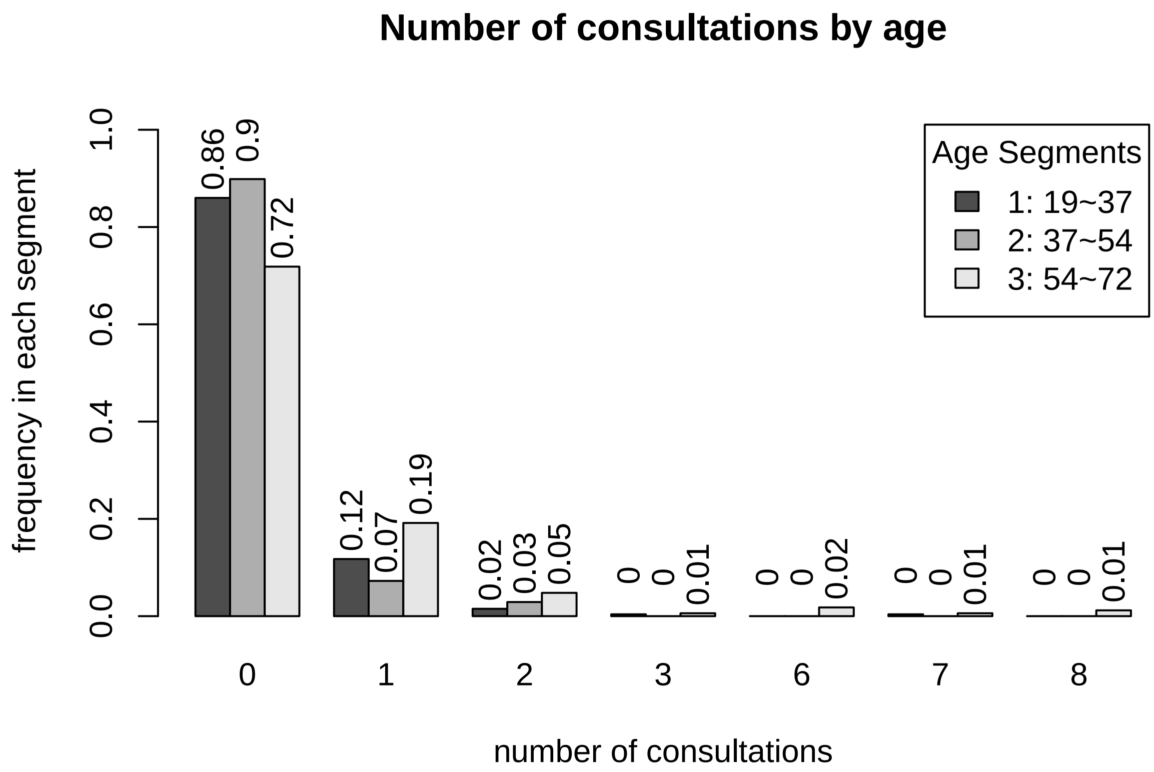
vs ch: similar pattern occurs at 0 consultations and 1 consultations.

For number of consultations vs continuous variables, it is hard to do EDA directly, rather, I split continuous variables into different chunks and do barplots.

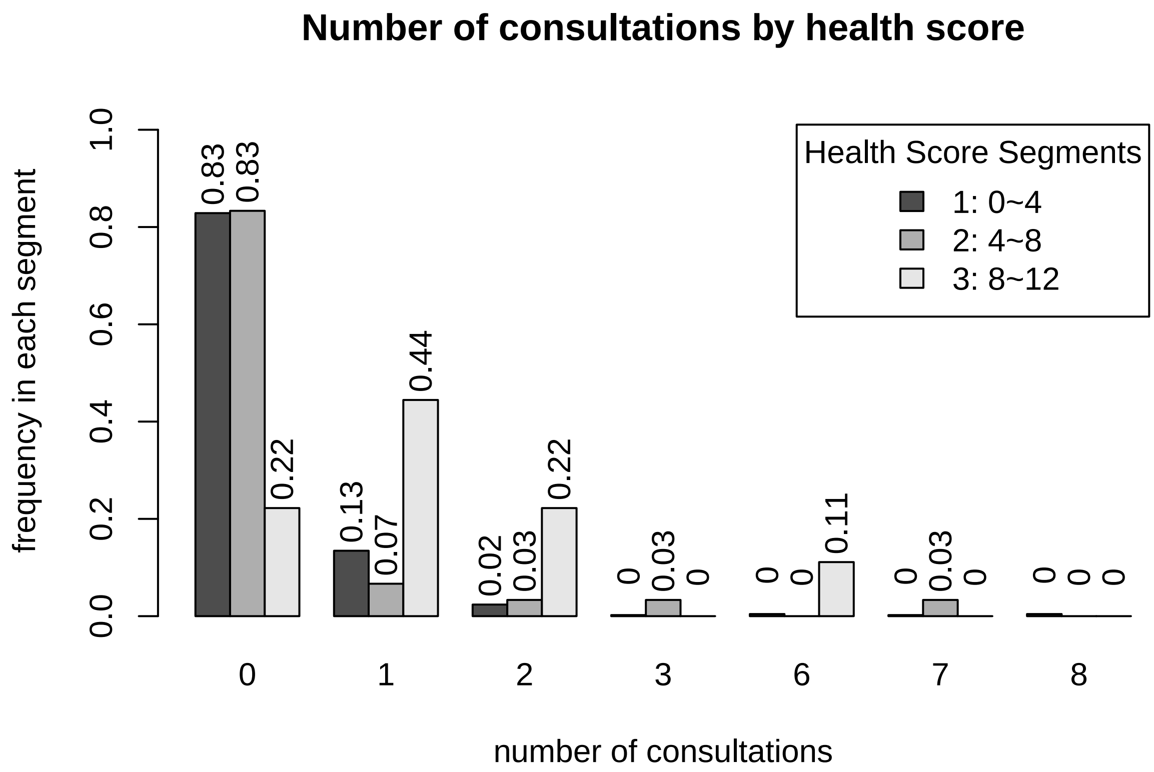
```
## split numerical values

## income
income <- cut(p3$income,3,labels = FALSE)
pc.inc <- table(p3$pc,income)
# transform count to ratio
for ( i in 1:ncol(pc.inc)){
  pc.inc[,i] <- pc.inc[,i]/sum(pc.inc[,i])
}
x <- barplot(t(pc.inc),beside = T, main="Number of consultations by income",legend.text = c("1: 0~50,00
text(x[1,],pc.inc[,1]+0.08,labels=as.character(round(pc.inc[,1],2)),srt=90)
text(x[2,],pc.inc[,2]+0.08,labels=as.character(round(pc.inc[,2],2)),srt=90)
text(x[3,],pc.inc[,3]+0.08,labels=as.character(round(pc.inc[,3],2)),srt=90)
```




```
## age
age <- cut(p3$age,3,labels = FALSE)
pc.age <- table(p3$pc,age)
# transform count to ratio
for ( i in 1:ncol(pc.age)){
  pc.age[,i] <- pc.age[,i]/sum(pc.age[,i])
}
x <- barplot(t(pc.age),beside = T, main="Number of consultations by age",legend.text = c("1: 19~37","2:
text(x[1,],pc.age[,1]+0.08,labels=as.character(round(pc.age[,1],2)),srt=90)
text(x[2,],pc.age[,2]+0.08,labels=as.character(round(pc.age[,2],2)),srt=90)
text(x[3,],pc.age[,3]+0.08,labels=as.character(round(pc.age[,3],2)),srt=90)
```




```
## score
score <- cut(p3$hs,3,labels = FALSE)
pc.sco <- table(p3$pc,score)
# transform count to ratio
for ( i in 1:ncol(pc.sco)){
  pc.sco[,i] <- pc.sco[,i]/sum(pc.sco[,i])
}
x <- barplot(t(pc.sco),beside = T, main="Number of consultations by health score",legend.text = c("1: 0
text(x[1,],pc.sco[,1]+0.08,labels=as.character(round(pc.sco[,1],2)),srt=90)
text(x[2,],pc.sco[,2]+0.08,labels=as.character(round(pc.sco[,2],2)),srt=90)
text(x[3,],pc.sco[,3]+0.08,labels=as.character(round(pc.sco[,3],2)),srt=90)
```



We can also find out possible relationships by comparing conditional mean and conditional standard deviation.

```
with (p3, tapply(pc,sex,function(x){
  paste("Mean is: ", round(mean(x),4), ", var is: ",round(var(x),4))
}))
```

```
##                                     0                                     1
## "Mean is:  0.1653 , var is:  0.4124" "Mean is:  0.4264 , var is:  1.2572"
```

```
with (p3, tapply(pc,ch,function(x){
  paste("Mean is: ", round(mean(x),4), ", var is: ",round(var(x),4))
}))
```

```
##                                     1                                     2
## "Mean is:  0.2718 , var is:  0.6282"  "Mean is:  0.6441 , var is:  1.578"
##                                     3
##  "Mean is:  0.2383 , var is:  0.866"
```

The table above shows the average numbers of consultation by different categorical variables and seems to suggest that number of each of them is a good candidate for predicting the number of consultation, our outcome variable, because the mean value of the outcome appears to vary by those covariates. The variances within each value of those categorical variables are higher than the means within each value. These are the conditional means and variances. These differences suggest that over-dispersion is present and that a Negative Binomial model would be appropriate.

From all the EDA above, it is hard to rule out any of those covariates. So let's fit the model with all the covariates included.

```r
### poisson diagnostics
diagFun <- function(fittedModel){

  # should be constant variance since deviance residuals have divided by V(mu)
  plot(residuals(fittedModel)~predict(fittedModel,type="link"),xlab=expression(hat(eta)),ylab="Deviance


  # not constant variance indicates overdispersion
  plot(residuals(fittedModel,type="response")~predict(fittedModel,type="link"),xlab=expression(hat(eta)

  # half normal plot
  halfnorm(residuals(fittedModel))
  #The half-normal plot of the (absolute value of the) residuals shown in Figure 5.3
  #shows no outliers.

  # mean is equal to the variance?
  plot(log(fitted(fittedModel)),log((p3$pc-fitted(fittedModel))^2),xlab=expression(hat(mu)),ylab=express
  abline(0,1)

  ### over-dispersion: est with pearson X square / df
  rp <- residuals(fittedModel, type = "pearson")
  rraw <- residuals(fittedModel, type = "response")
  phi <- sum(rp^2)/fittedModel$df.res

  ### over-dispersion: est with deviance / df
  phi2 <- fittedModel$deviance/fittedModel$df.res

  ### over-dispersion: dispersion test
  dispersiontest(fittedModel)

  ### Goodness of fit test
  g <- pchisq(fittedModel$deviance, df=fittedModel$df.residual, lower.tail=FALSE)

  ### return over-dispersion result
  result <- data.frame(estimated_phi_pearson=phi,estimated_phi_deviance=phi2,dispersion_test_p=dispersi
  #result <- data.frame(estimated_phi_pearson=phi,estimated_phi_deviance=phi2, Goodness=g)
  return(result)
}
```

Model Fitting

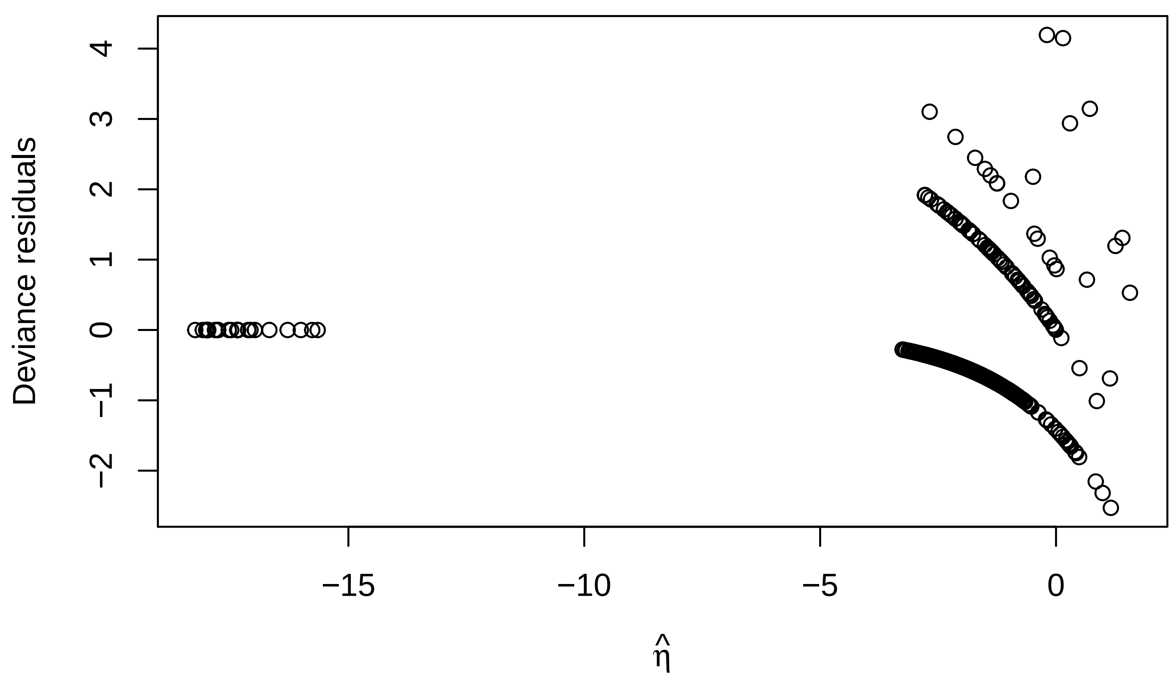
1. Fit on all covariates

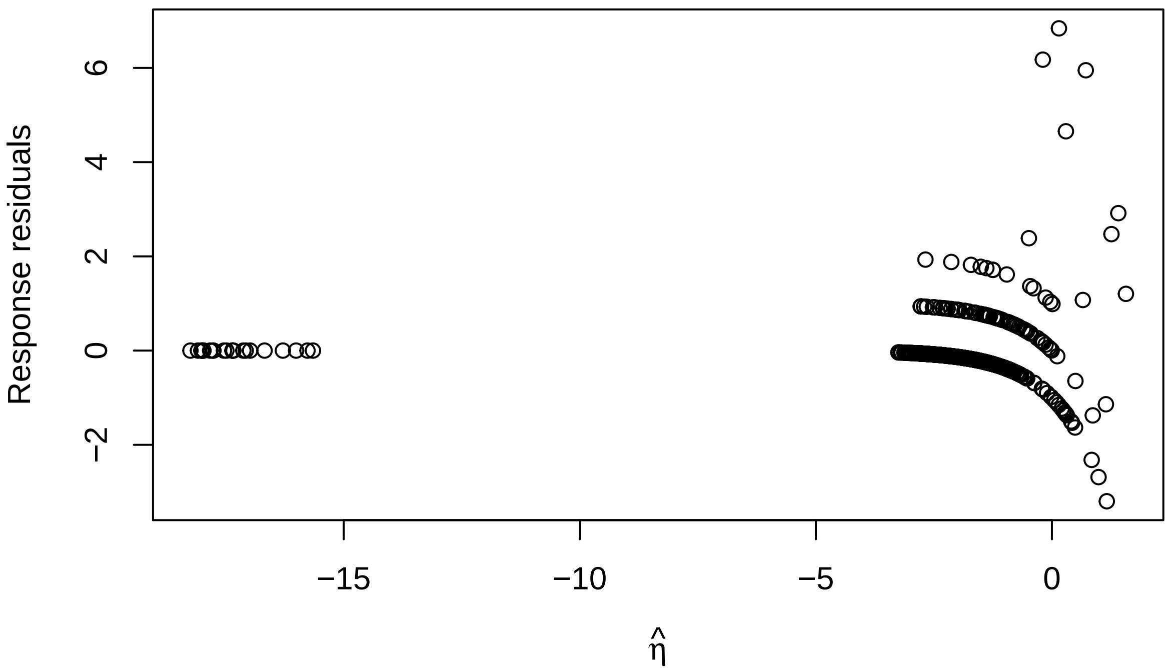
```r
fit1 <- glm(pc ~., family = poisson, data = p3)
summary(fit1)
```

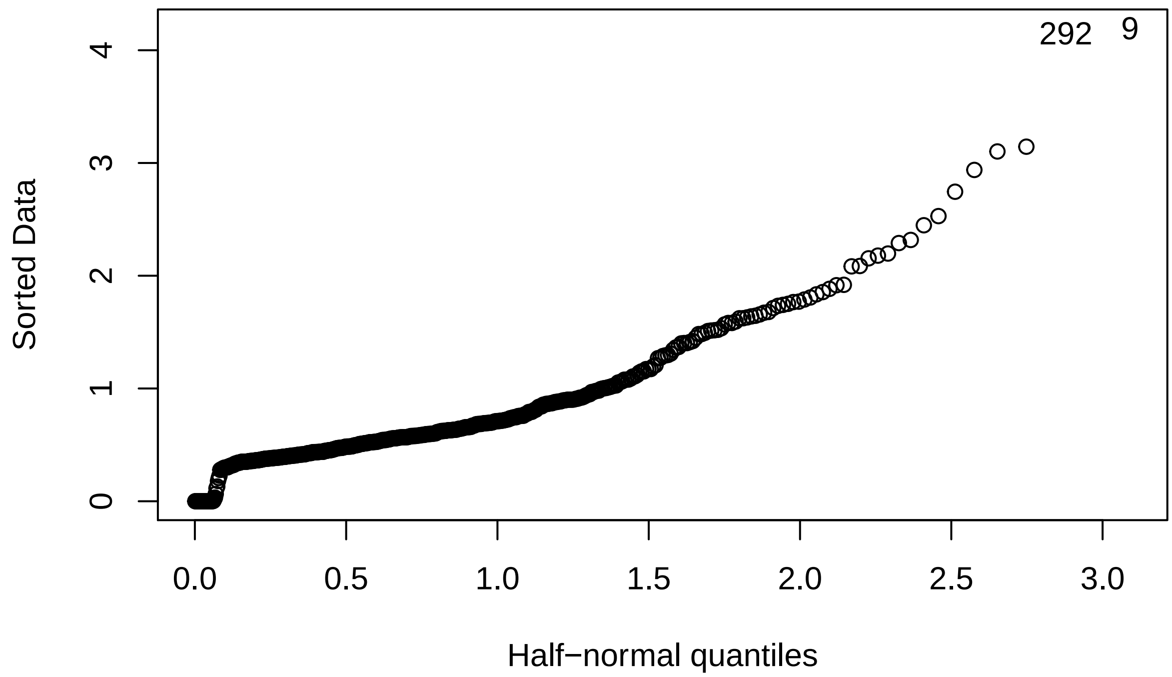
```
##
## Call:
## glm(formula = pc ~ ., family = poisson, data = p3)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5289  -0.6343  -0.4539  -0.3145   4.1940
##
## Coefficients:
```

```
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.76624    0.45381  -8.299  < 2e-16 ***
## sex           0.74260    0.20244   3.668 0.000244 ***
## age           2.26475    0.54056   4.190 2.79e-05 ***
## income        0.03457    0.27356   0.126 0.899438
## ill           0.15004    0.05726   2.620 0.008790 **
## ad            1.55615    0.17667   8.808  < 2e-16 ***
## hs            0.09696    0.02986   3.248 0.001164 **
## insurance2  -15.24404  607.18443  -0.025 0.979970
## insurance3   -0.51934    0.22285  -2.330 0.019784 *
## insurance4    0.04743    0.24347   0.195 0.845549
## ch2           0.40452    0.23472   1.723 0.084820 .
## ch3           0.48217    0.21136   2.281 0.022534 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 598.72  on 499  degrees of freedom
## Residual deviance: 393.33  on 488  degrees of freedom
## AIC: 622.37
##
## Number of Fisher Scoring iterations: 15
```

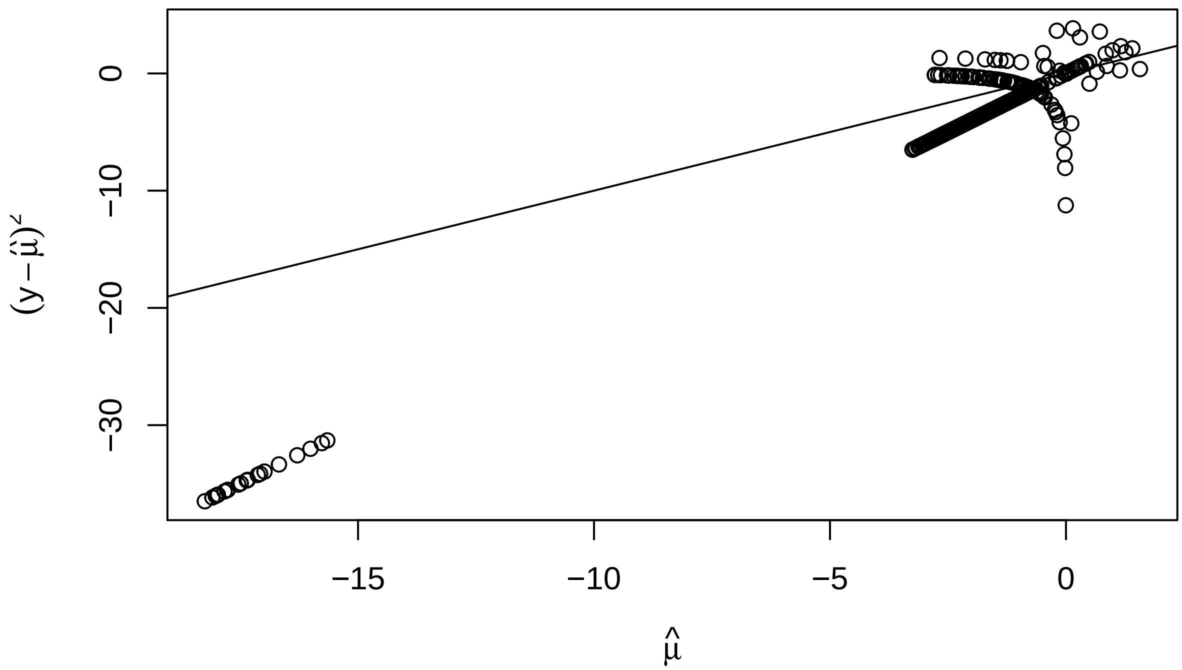
```r
diagFun(fit1)
```




Response residuals
6
4
2
0
−2
−15
−10
−5
0
$\hat{\eta}$



292
9
Sorted Data
0
1
2
3
4
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Half−normal quantiles




```
##   estimated_phi_pearson estimated_phi_deviance dispersion_test_p  Goodness
## 1               1.28926              0.8060125        0.01202283 0.9993813
```

```
outlierTest(fit1)
```

```
##     rstudent unadjusted p-value Bonferroni p
## 292 4.818766         1.4445e-06   0.00072225
## 9   4.474932         7.6436e-06   0.00382180
```

It's normal that the residual plot is hard to judge with many zeros

```
### simulation small mu
# df <- data.frame(x0=0.01,x1=seq(0,1,by = 0.01))
# df$mu <- df$x0+df$x1
# myvec <- rpois(nrow(df),df$mu)
# df$y <- myvec
# simfit <- glm(y~x1,data=df,family = "poisson")
# summary(simfit)
# diagFun(simfit)
```

```
### simulation large mu
# df <- data.frame(x0=2,x1=seq(0,10,by = 0.1))
# df$mu <- df$x0+df$x1
# myvec <- rpois(nrow(df),df$mu)
# df$y <- myvec
# simfit <- glm(y~x1,data=df,family = "poisson")
# summary(simfit)
# diagFun(simfit)
```

want to remove insignificant covariates. But before that, do tests to double check. Especially for categorical variable, one level insignificance does not necessarily indicate an overall insignificance.

Since our tests are approximated tests, I will use different tests results and combine their results to make the decision.

If the mean structure is specified correctly, the first plot should show no dependence of the size or sign of the residual on the value of the linear predictor. If the response is Poisson, then the squared raw residual should be on average equal to the mean (because it's an estimate for the variance, and variance equals mean for Poisson distribution).

```
## test income (insignificant)
wald.test(b=coef(fit1),Sigma = vcov(fit1),Terms = 4)
```

```
## Wald test:
## ----------
##
## Chi-squared test:
## X2 = 0.016, df = 1, P(> X2) = 0.9
```

```
fit1.2 <- update(fit1,pc ~ .-income)
anova(fit1.2,fit1,test="Chi")
```

```
## Analysis of Deviance Table
##
## Model 1: pc ~ sex + age + ill + ad + hs + insurance + ch
## Model 2: pc ~ sex + age + income + ill + ad + hs + insurance + ch
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       489     393.35
## 2       488     393.33  1 0.015928   0.8996
```

```
## test insurance (significant)
wald.test(b=coef(fit1),Sigma = vcov(fit1),Terms = 8:10) # only test 8 or 10 is exactly the same as in s
```

```
## Wald test:
## ----------
##
## Chi-squared test:
## X2 = 5.8, df = 3, P(> X2) = 0.12
```

```
fit1.2 <- update(fit1,pc ~ .-insurance)
anova(fit1.2,fit1,test="Chi")
```

```
## Analysis of Deviance Table
##
## Model 1: pc ~ sex + age + income + ill + ad + hs + ch
## Model 2: pc ~ sex + age + income + ill + ad + hs + insurance + ch
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       491     406.17
## 2       488     393.33  3   12.832 0.005015 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## test ch (significant)
wald.test(b=coef(fit1),Sigma = vcov(fit1),Terms = 11:12)
```

```
## Wald test:
## ----------
##
```

```
## Chi-squared test:
## X2 = 6.3, df = 2, P(> X2) = 0.043
```

```r
fit1.2 <- update(fit1,pc ~ .-ch)
anova(fit1.2,fit1,test="Chi")
```

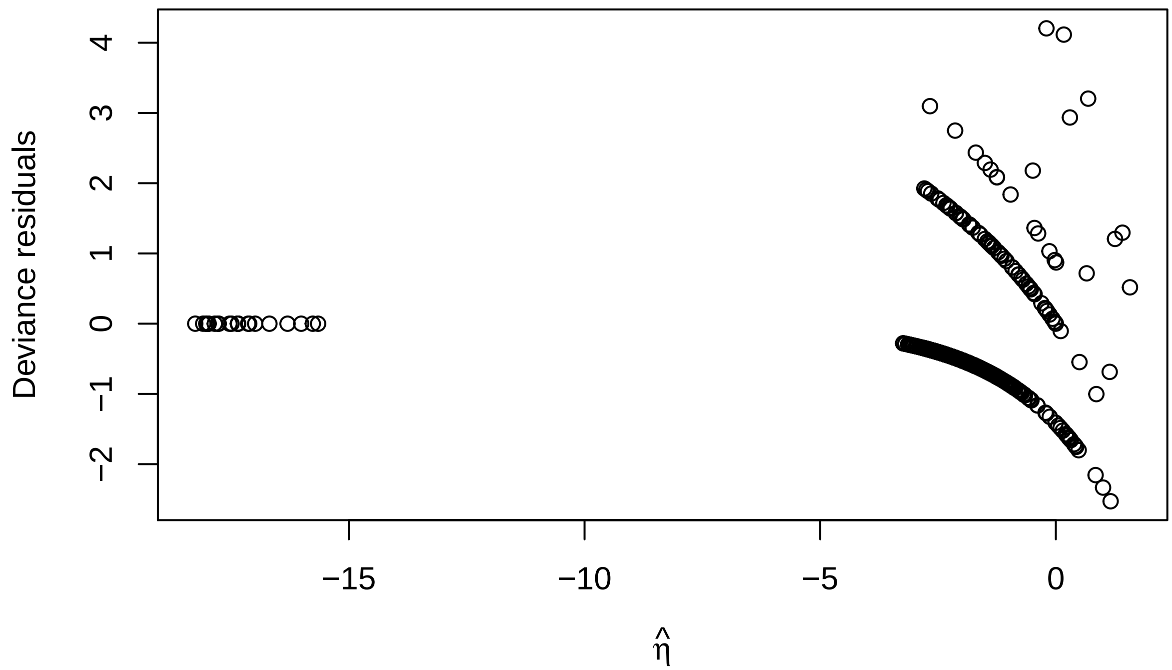
```
## Analysis of Deviance Table
##
## Model 1: pc ~ sex + age + income + ill + ad + hs + insurance
## Model 2: pc ~ sex + age + income + ill + ad + hs + insurance + ch
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       490     399.81
## 2       488     393.33  2   6.4788  0.03919 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

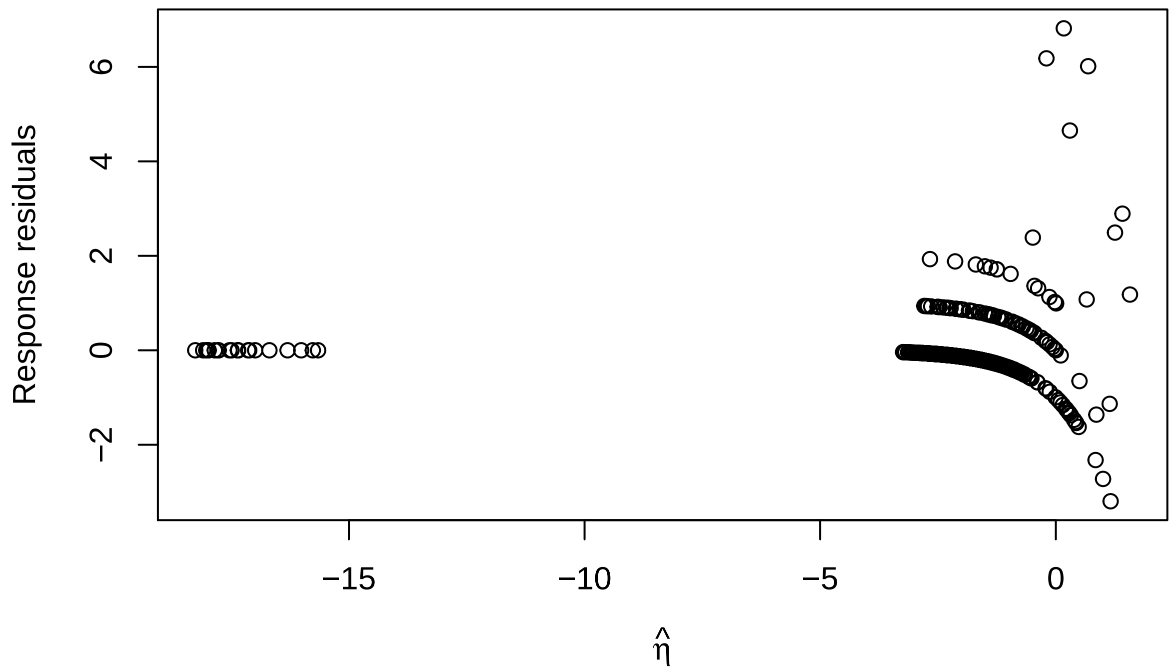
2. Remove income variable

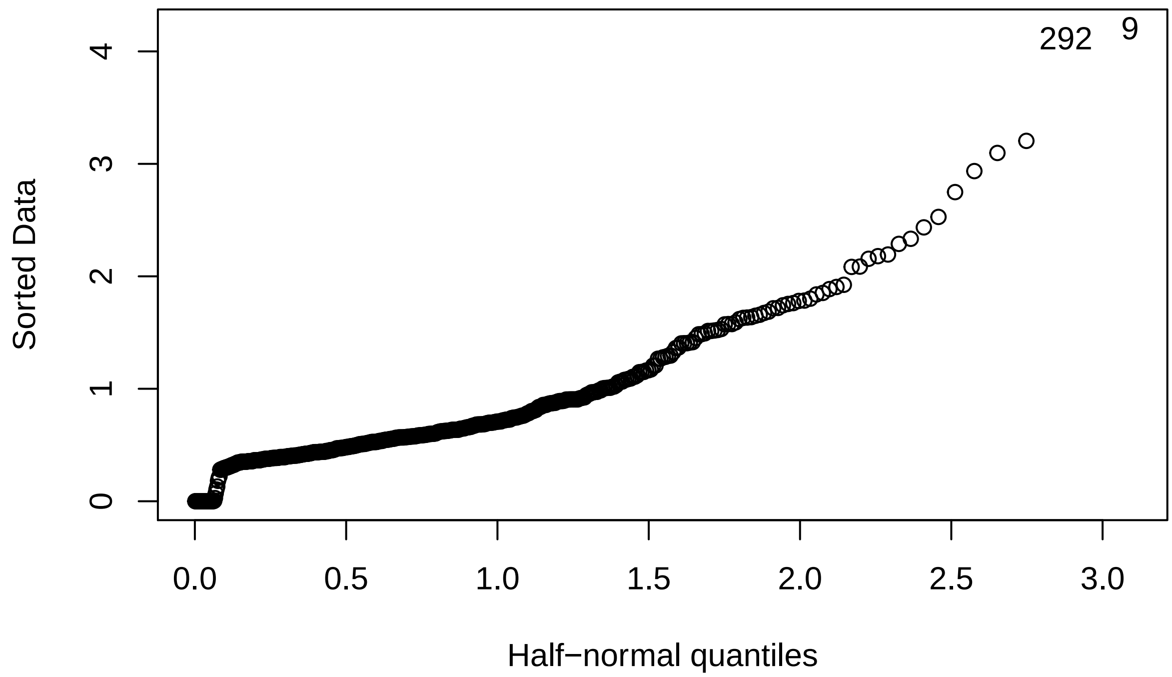
```r
fit2 <- update(fit1,pc~.-income)
summary(fit2)
```

```
##
## Call:
## glm(formula = pc ~ sex + age + ill + ad + hs + insurance + ch,
##     family = poisson, data = p3)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5281  -0.6352  -0.4546  -0.3135   4.2048
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.73491    0.37971  -9.836  < 2e-16 ***
## sex           0.73755    0.19841   3.717 0.000201 ***
## age           2.25635    0.53651   4.206  2.6e-05 ***
## ill           0.14933    0.05700   2.620 0.008798 **
## ad            1.55755    0.17635   8.832  < 2e-16 ***
## hs            0.09629    0.02939   3.277 0.001050 **
## insurance2  -15.26655  609.15579  -0.025 0.980006
## insurance3   -0.52723    0.21365  -2.468 0.013599 *
## insurance4    0.04426    0.24229   0.183 0.855051
## ch2           0.40552    0.23448   1.729 0.083722 .
## ch3           0.48124    0.21127   2.278 0.022739 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 598.72  on 499  degrees of freedom
## Residual deviance: 393.35  on 489  degrees of freedom
## AIC: 620.39
##
## Number of Fisher Scoring iterations: 15
```

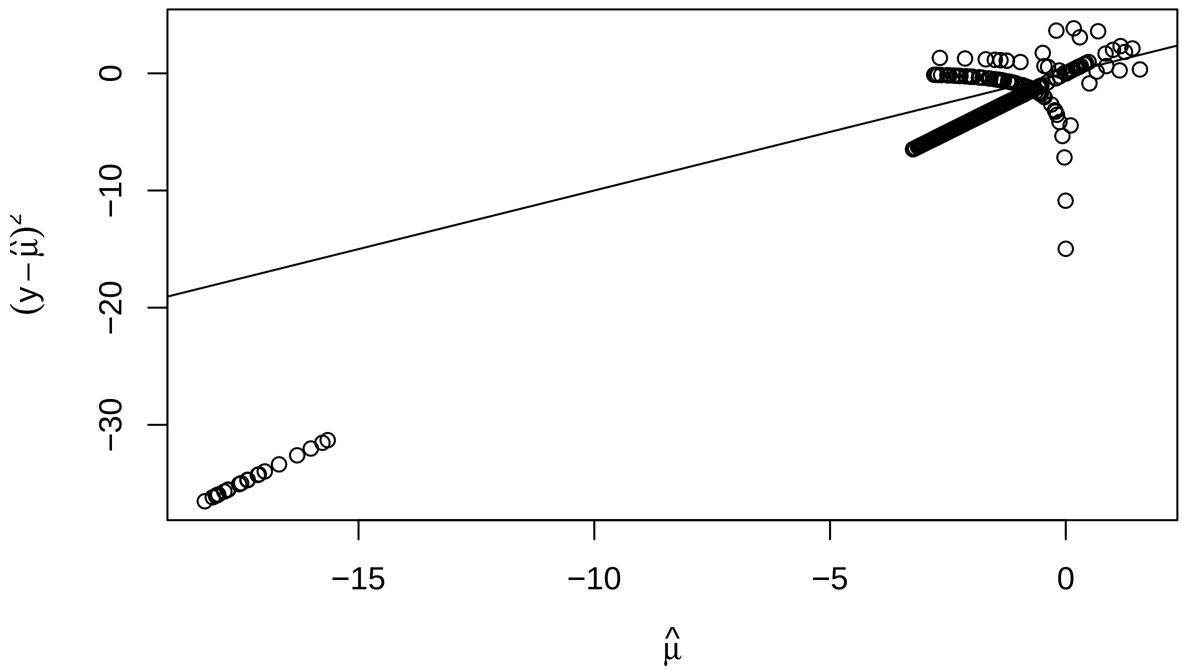
```r
diagFun(fit2)
```


Deviance residuals
4
3
2
1
0
−1
−2
−15
−10
−5
0
$\hat{\eta}$


Response residuals
6
4
2
0
−2
−15
−10
−5
0
$\hat{\eta}$



292
9
Sorted Data
0
1
2
3
4
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Half−normal quantiles




```
##   estimated_phi_pearson estimated_phi_deviance dispersion_test_p  Goodness
## 1              1.285742              0.8043968        0.01189975 0.9994482
```

Again, check significance of coefficients

```
## test insurance (significant)
wald.test(b=coef(fit2),Sigma = vcov(fit2),Terms = 7:9) # only test 8 or 10 is exactly the same as in su
```

```
## Wald test:
## ----------
##
## Chi-squared test:
## X2 = 6.4, df = 3, P(> X2) = 0.093
```

```
fit2.2 <- update(fit2,pc ~ .-insurance)
anova(fit2.2,fit2,test="Chi")
```

```
## Analysis of Deviance Table
##
## Model 1: pc ~ sex + age + ill + ad + hs + ch
## Model 2: pc ~ sex + age + ill + ad + hs + insurance + ch
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       492     407.64
## 2       489     393.35  3   14.289 0.002537 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
## test ch (significant)
wald.test(b=coef(fit2),Sigma = vcov(fit2),Terms = 10:11)
```

```
## Wald test:
## ----------
##
## Chi-squared test:
## X2 = 6.3, df = 2, P(> X2) = 0.043
```

```r
fit2.2 <- update(fit2,pc ~ .-ch)
anova(fit2.2,fit1,test="Chi")
```

```
## Analysis of Deviance Table
##
## Model 1: pc ~ sex + age + ill + ad + hs + insurance
## Model 2: pc ~ sex + age + income + ill + ad + hs + insurance + ch
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       491     399.82
## 2       488     393.33  3    6.488  0.09014 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

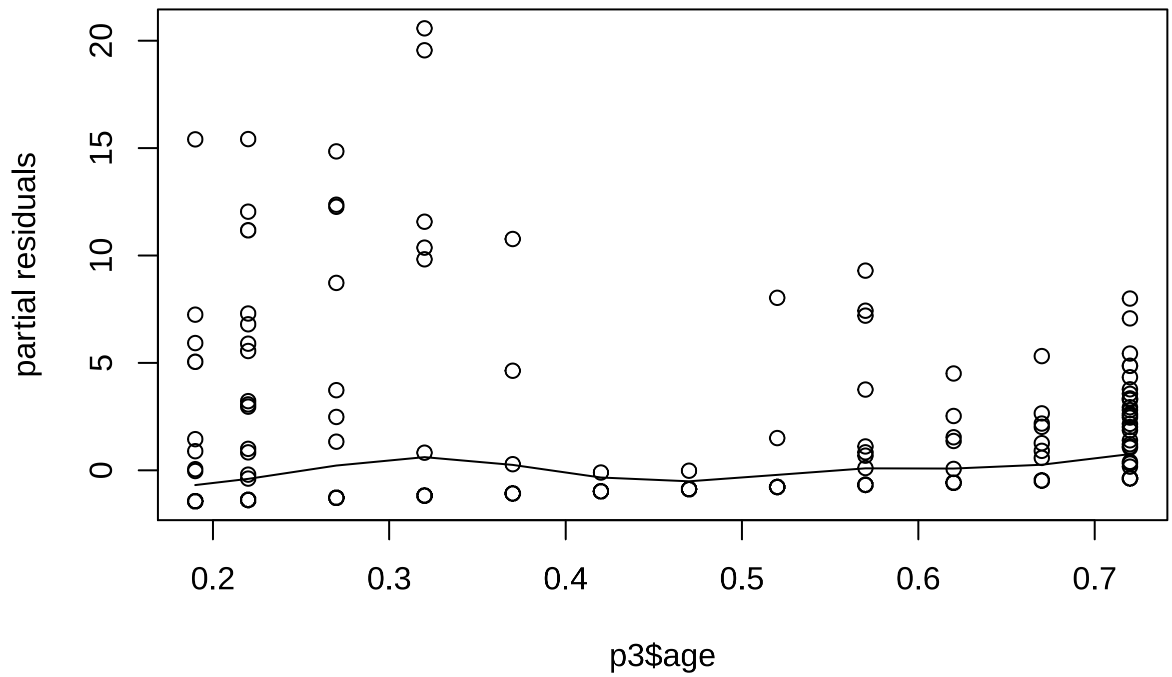
Then test model goodness-of-fit and possible overdispersion:

It is included in the diagnostics. It's good.

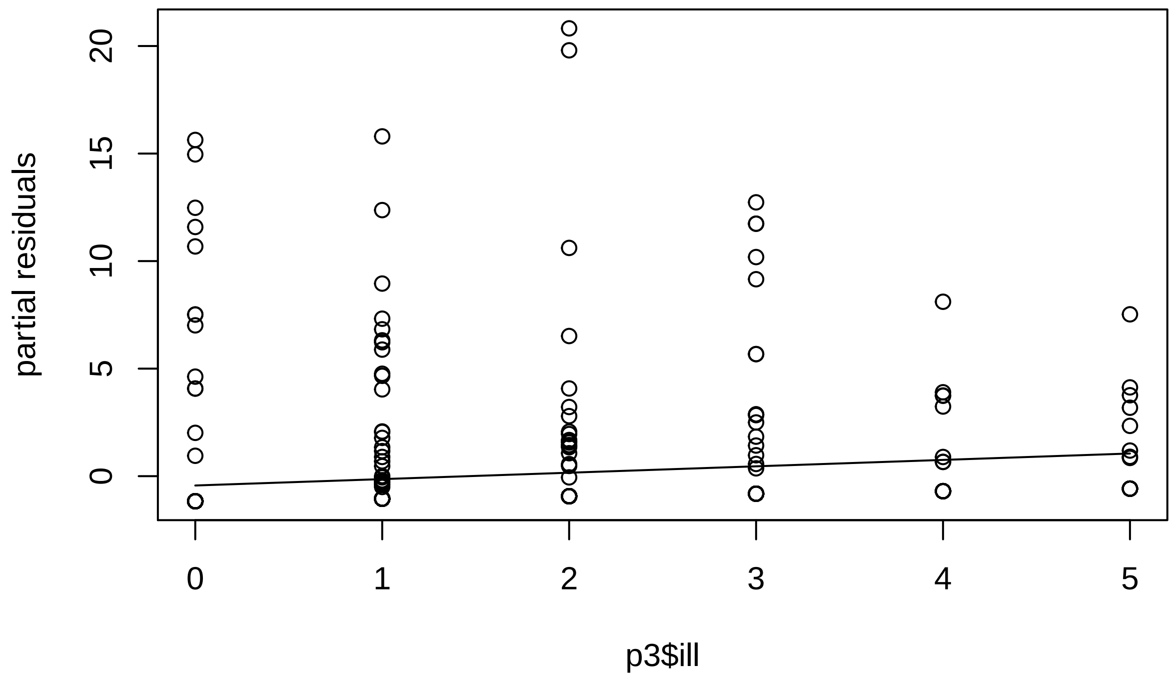
Notice that each time remove some covariates, the significance of other covariates may change a lot. May indicate interaction.

Do we need higher order terms?

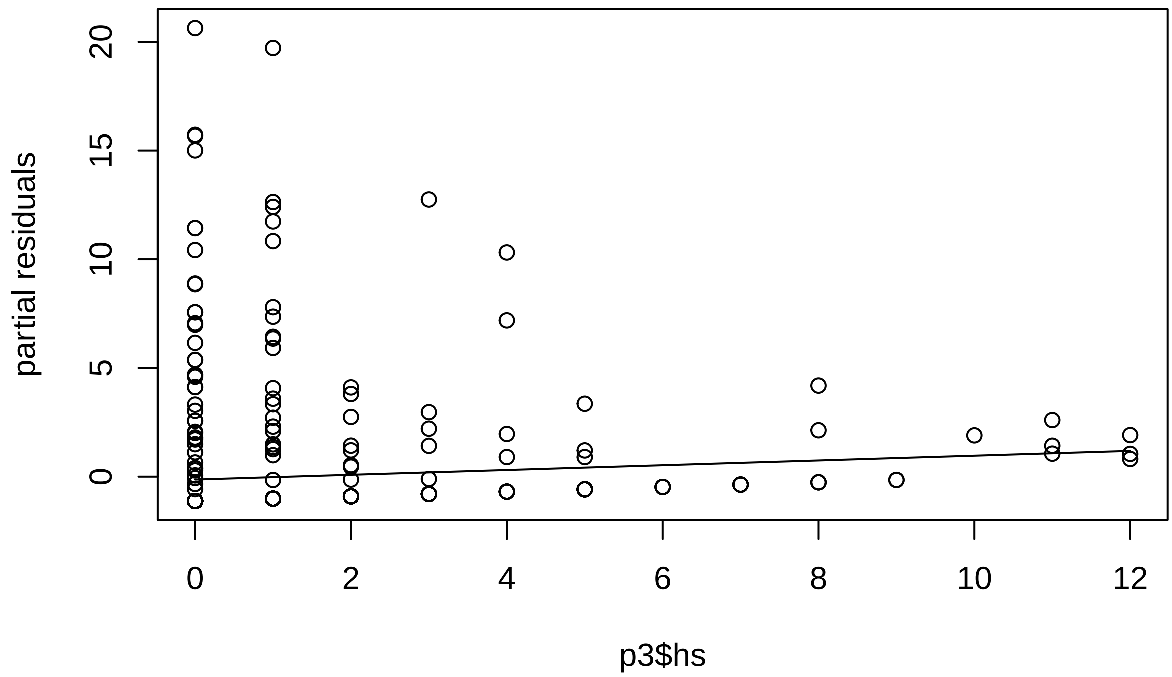
```r
# age
parresi <- residuals(fit2.2,type="partial")
plot(p3$age,parresi[,2],ylab="partial residuals")
lines(smooth.spline(p3$age,parresi[,2]))
```




```
#ill
parresi <- residuals(fit2.2,type="partial")
plot(p3$ill,parresi[,3],ylab="partial residuals")
lines(smooth.spline(p3$ill,parresi[,3]))
```




```
#hs
parresi <- residuals(fit2.2,type="partial")
plot(p3$hs,parresi[,5],ylab="partial residuals")
lines(smooth.spline(p3$hs,parresi[,5]))
```



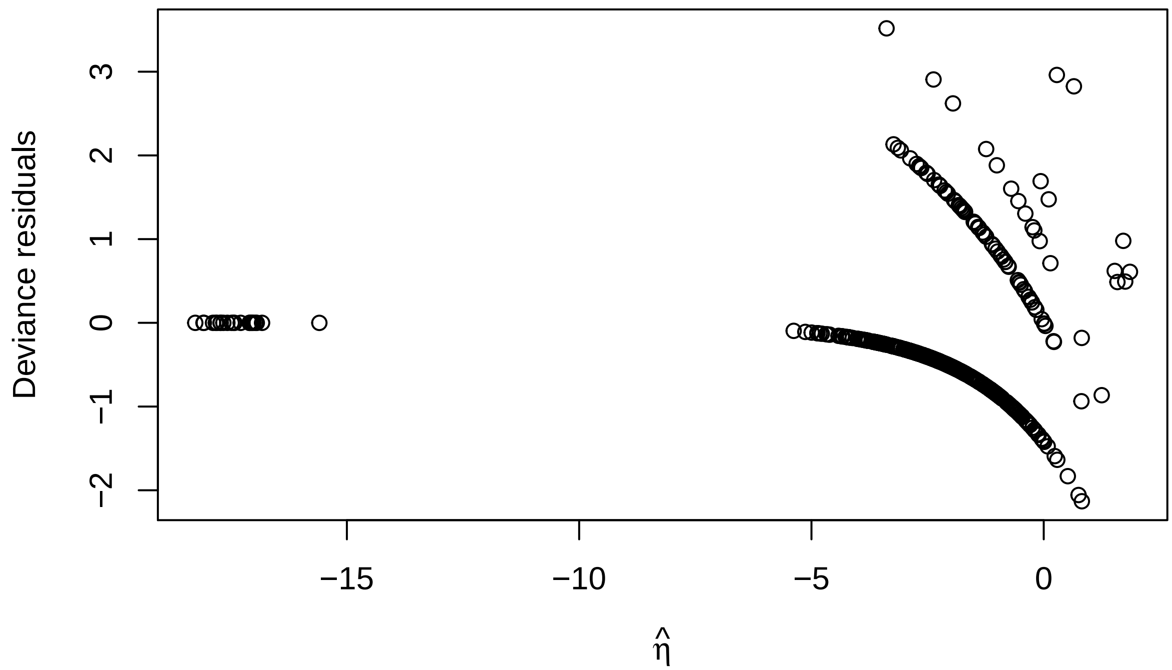
3. High order terms

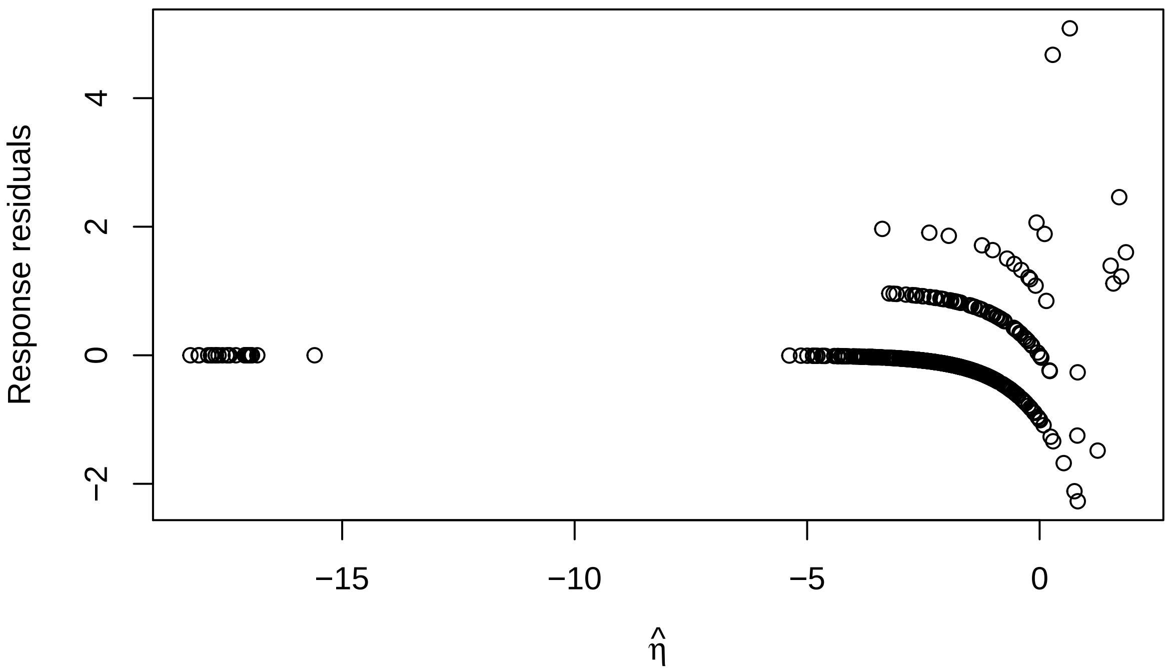
```
fit3 <- stepAIC(fit1,~.^2,trace=F)
summary(fit3)
```

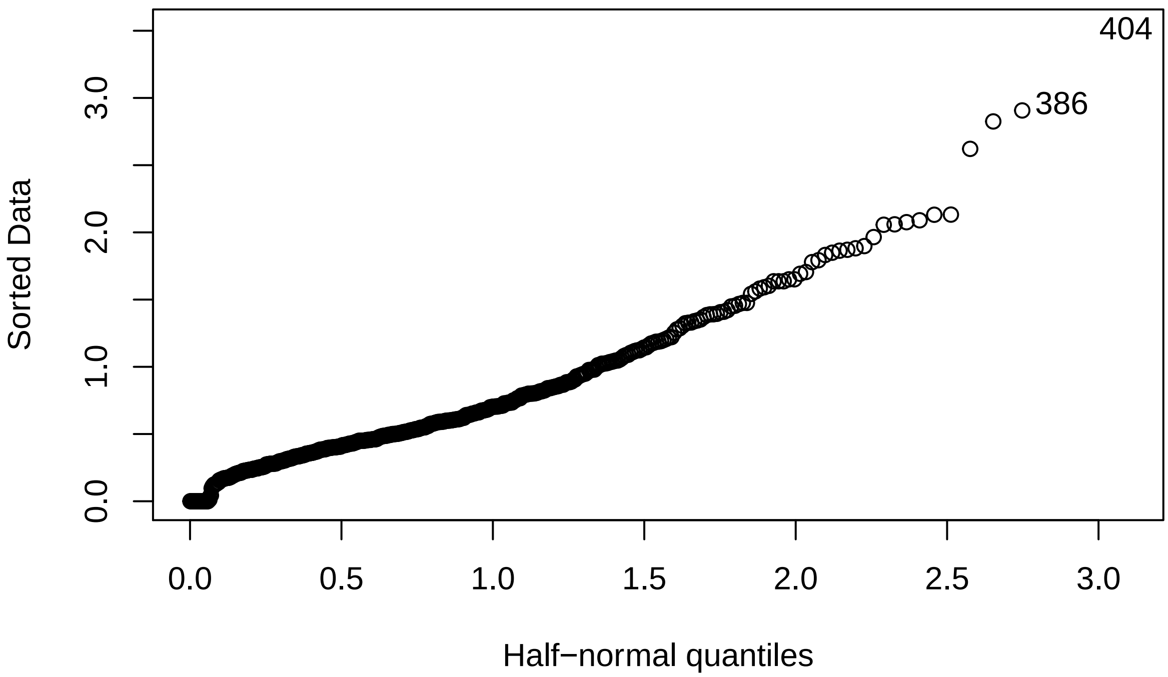
```
##
## Call:
## glm(formula = pc ~ sex + age + income + ill + ad + hs + insurance +
##     ch + income:ch + hs:insurance + ad:insurance + age:hs + ill:hs +
##     age:ad + ad:ch + ill:ch, family = poisson, data = p3)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1312  -0.6046  -0.3953  -0.1694   3.5177
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -2.060e+00  6.170e-01  -3.339 0.000840 ***
## sex              9.286e-01  2.225e-01   4.174 3.00e-05 ***
## age             -8.337e-01  9.149e-01  -0.911 0.362134
## income          -1.725e+00  6.213e-01  -2.777 0.005490 **
## ill              3.554e-01  1.070e-01   3.322 0.000893 ***
## ad               5.711e-01  6.656e-01   0.858 0.390895
## hs              -5.762e-02  1.489e-01  -0.387 0.698747
## insurance2      -1.503e+01  8.257e+02  -0.018 0.985478
## insurance3       5.935e-02  3.858e-01   0.154 0.877716
## insurance4      -9.909e-01  4.429e-01  -2.237 0.025276 *
```

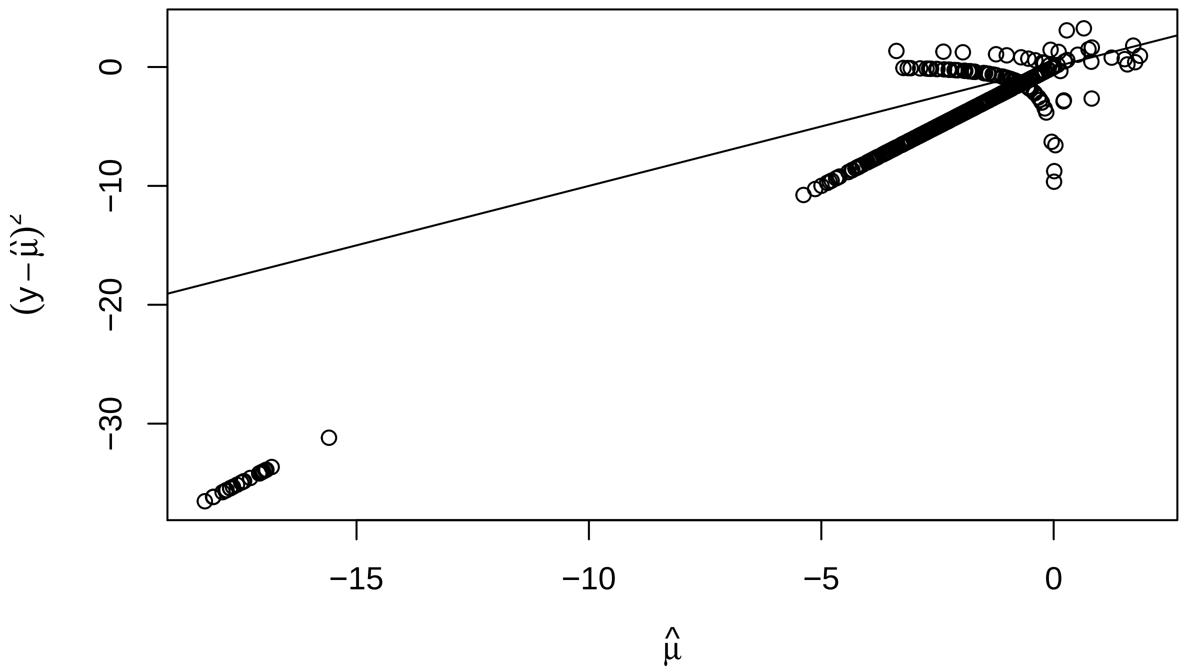
```
## ch2            2.778e-01  6.524e-01   0.426 0.670241
## ch3           -7.742e-01  5.335e-01  -1.451 0.146725
## income:ch2     1.012e+00  9.444e-01   1.072 0.283875
## income:ch3     2.587e+00  7.127e-01   3.629 0.000284 ***
## hs:insurance2  5.906e-02  3.230e+02   0.000 0.999854
## hs:insurance3 -2.600e-03  9.577e-02  -0.027 0.978344
## hs:insurance4  5.051e-01  1.249e-01   4.044 5.26e-05 ***
## ad:insurance2 -1.686e+00  2.404e+03  -0.001 0.999440
## ad:insurance3 -1.113e+00  4.865e-01  -2.287 0.022219 *
## ad:insurance4  6.978e-01  5.163e-01   1.352 0.176473
## age:hs         4.607e-01  2.430e-01   1.896 0.058010 .
## ill:hs        -5.895e-02  2.433e-02  -2.423 0.015404 *
## age:ad         2.322e+00  1.130e+00   2.055 0.039909 *
## ad:ch2        -5.090e-01  4.560e-01  -1.116 0.264353
## ad:ch3         8.163e-01  4.519e-01   1.806 0.070891 .
## ill:ch2       -4.812e-03  1.520e-01  -0.032 0.974739
## ill:ch3       -3.648e-01  1.958e-01  -1.863 0.062434 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 598.72  on 499  degrees of freedom
## Residual deviance: 324.70  on 473  degrees of freedom
## AIC: 583.74
##
## Number of Fisher Scoring iterations: 15
```

```
diagFun(fit3)
```


Deviance residuals
3
2
1
0
−1
−2
−15
−10
−5
0
η̂


Response residuals
−2
0
2
4
−15
−10
−5
0
$\hat{\eta}$


404
386
Sorted Data
0.0
1.0
2.0
3.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Half−normal quantiles



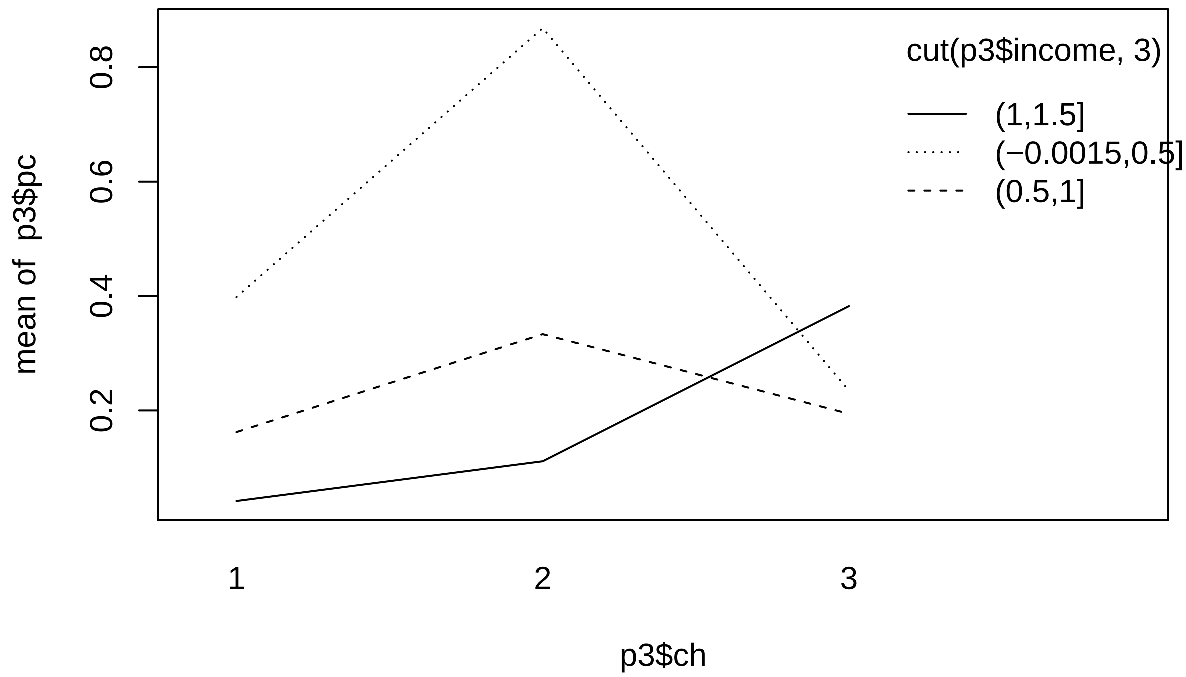

```
##   estimated_phi_pearson estimated_phi_deviance dispersion_test_p Goodness
## 1              1.296199              0.6864643         0.1222508        1
```

```r
outlierTest(fit3)
```

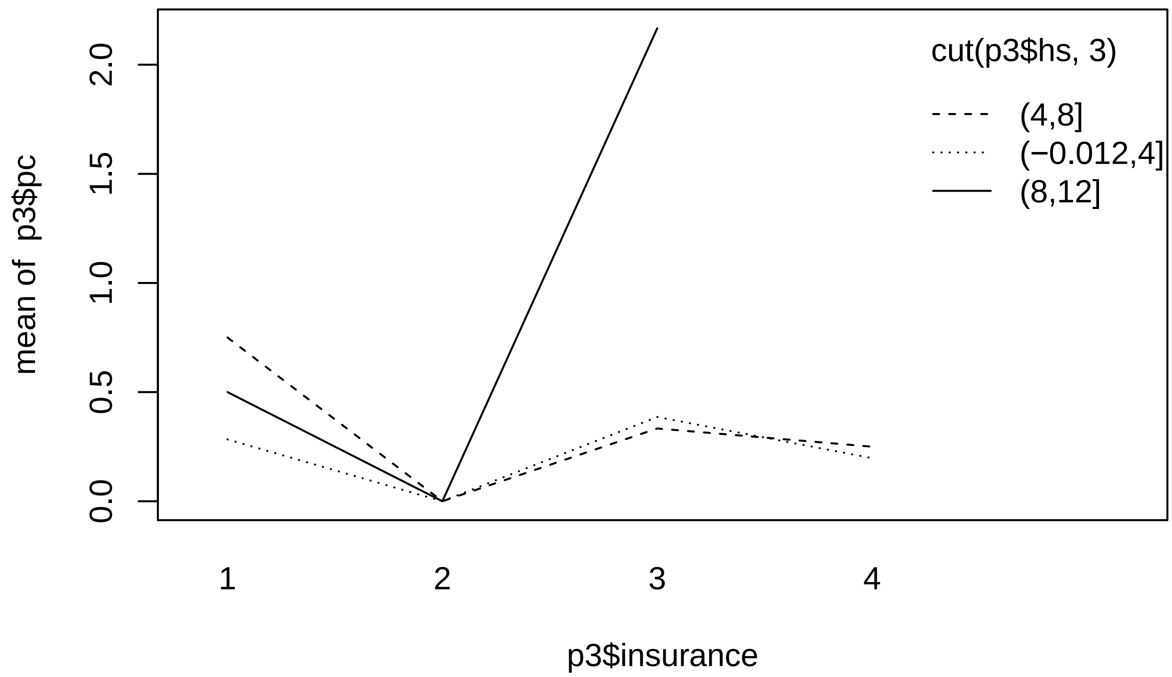
```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##     rstudent unadjusted p-value Bonferroni p
## 404 3.645069         0.00026732      0.13366
```

remove insignificant covariates

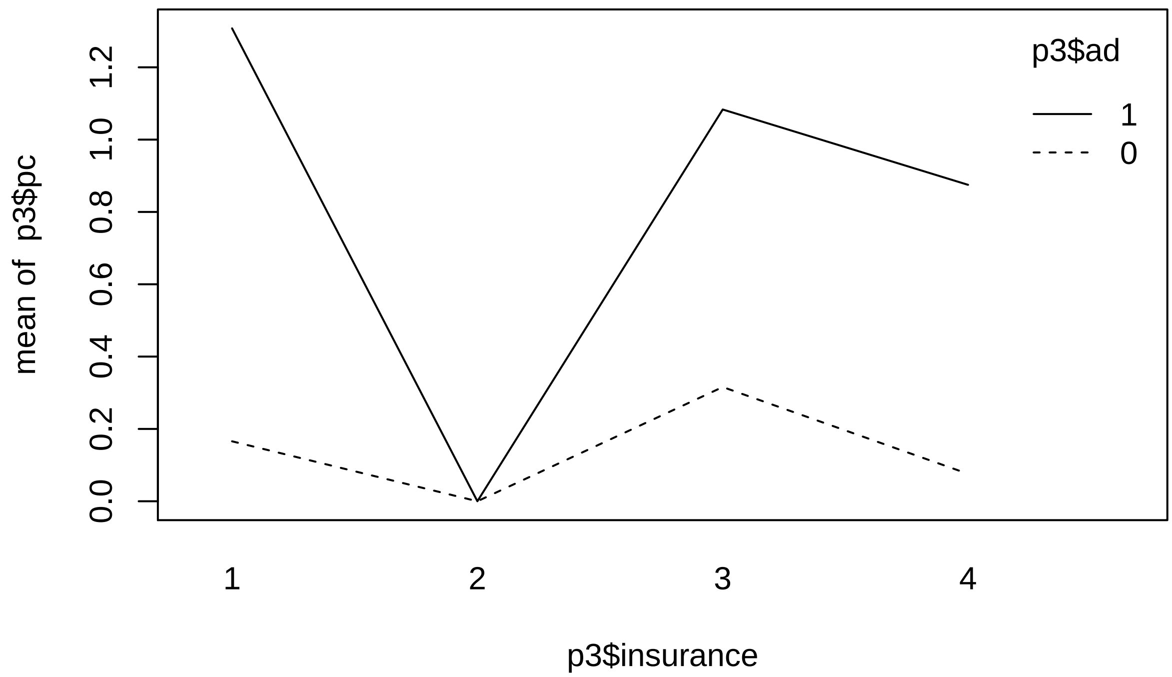
```r
## visualization
interaction.plot(p3$ch, cut(p3$income,3), p3$pc)        # yes
```



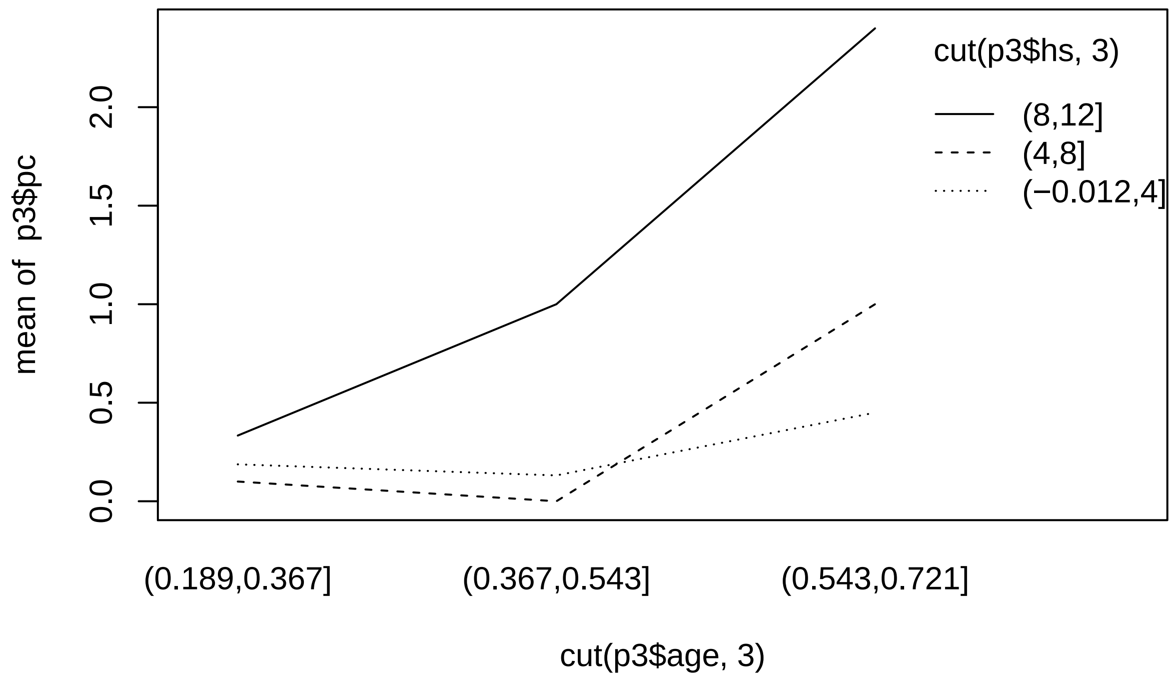

```r
interaction.plot(p3$insurance, cut(p3$hs,3), p3$pc)     # yes
```



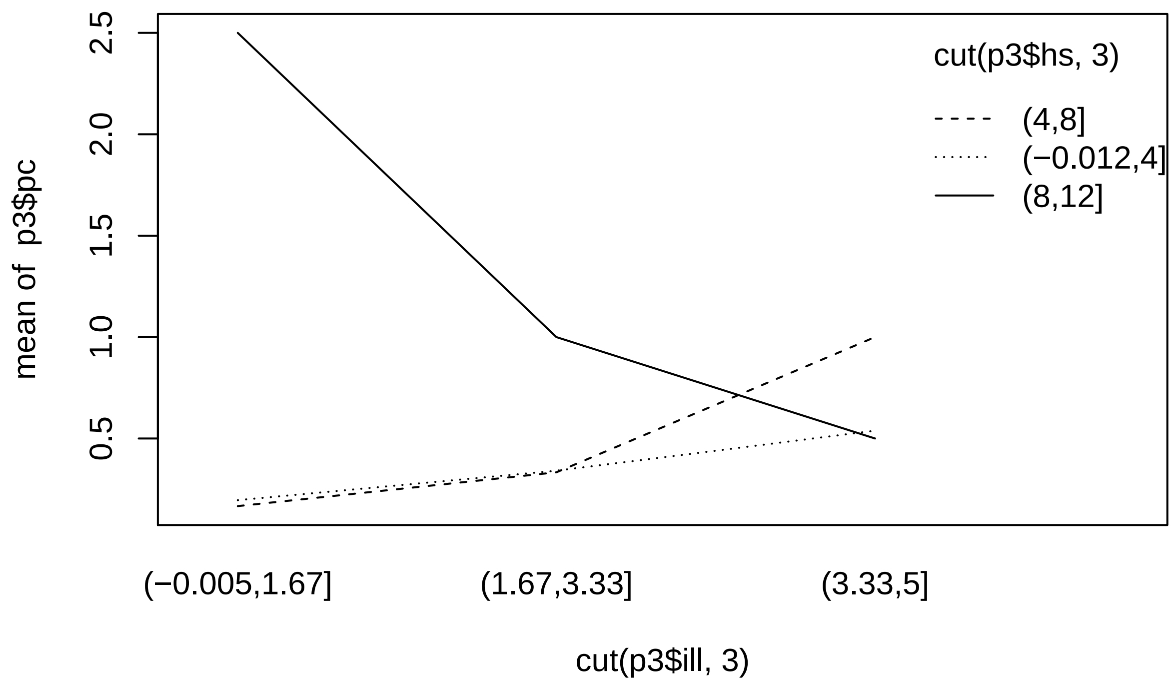

```r
interaction.plot(p3$insurance, p3$ad, p3$pc)            # maybe
```



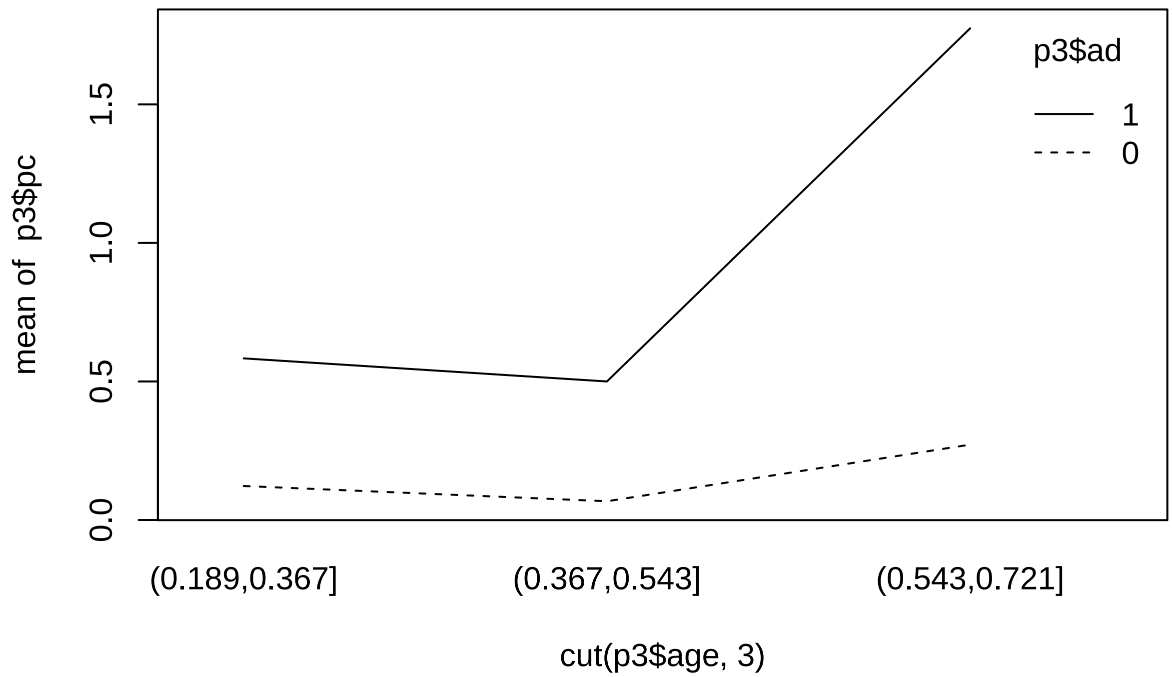

```
interaction.plot(cut(p3$age,3), cut(p3$hs,3), p3$pc)     # yes
```



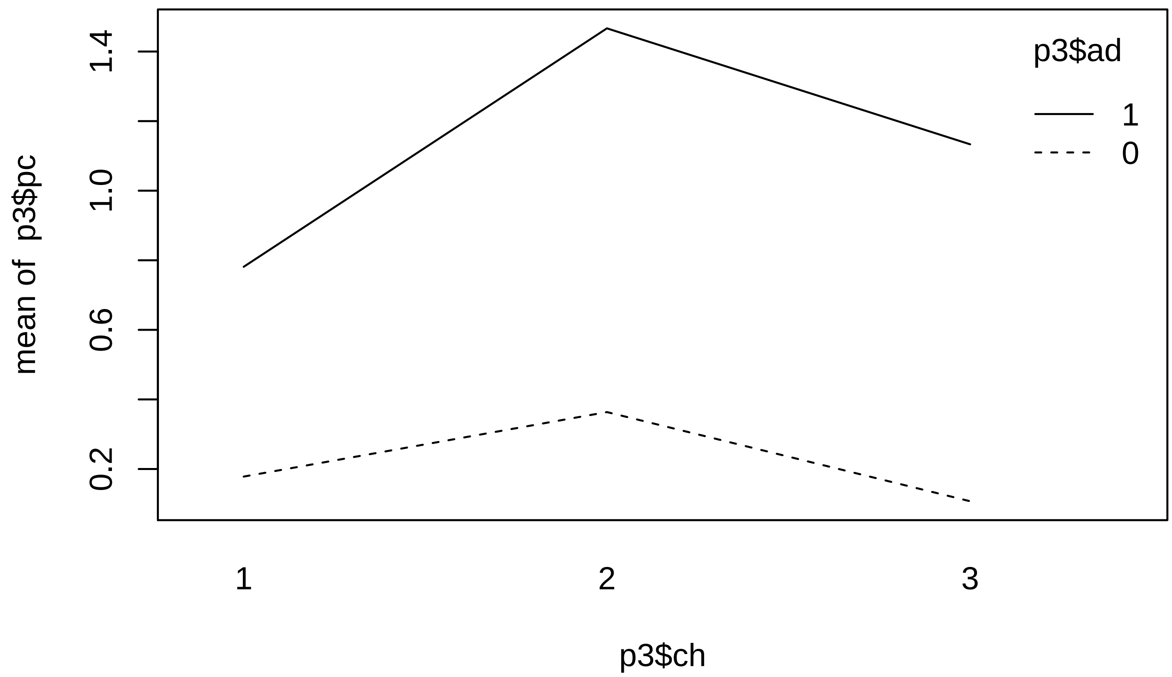

```
interaction.plot(cut(p3$ill,3), cut(p3$hs,3), p3$pc)     # yes
```



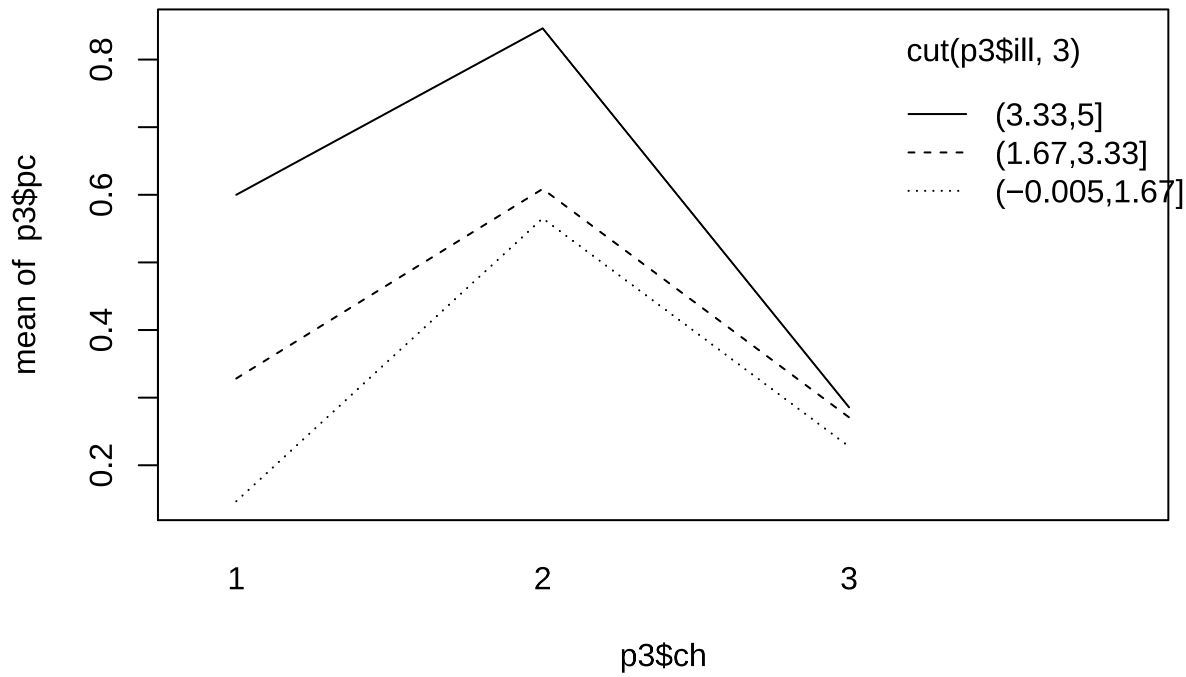

```
interaction.plot(cut(p3$age,3), p3$ad, p3$pc)          # no
```




```
interaction.plot(p3$ch, p3$ad, p3$pc)                    # no
```




```r
interaction.plot(p3$ch, cut(p3$ill,3), p3$pc)          # no
```




```r
# remove
fit3.2 <- update(fit3,~.-ad:ch-ill:ch-age:ad-ad:pc)
summary(fit3.2)
```

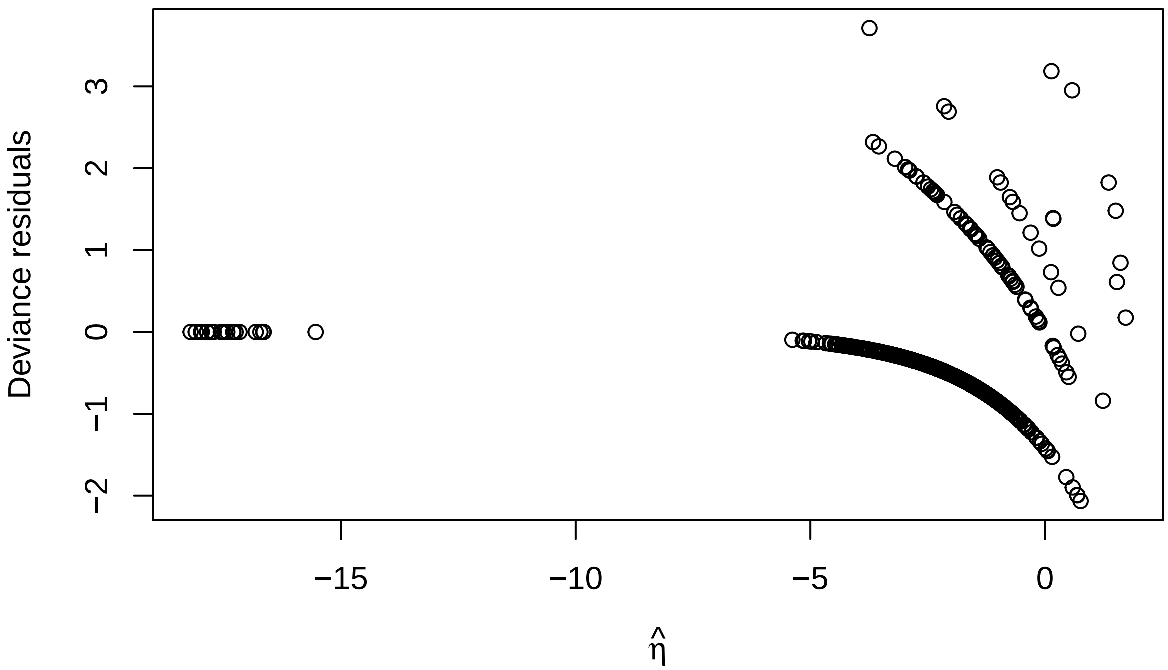
```
##
## Call:
## glm(formula = pc ~ sex + age + income + ill + ad + hs + insurance +
##     ch + income:ch + hs:insurance + ad:insurance + age:hs + ill:hs,
##     family = poisson, data = p3)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0656  -0.6498  -0.3875  -0.1660   3.7122
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -2.746e+00  5.413e-01  -5.072 3.93e-07 ***
## sex            9.801e-01  2.239e-01   4.378 1.20e-05 ***
## age            7.376e-01  7.234e-01   1.020 0.307887
## income        -1.655e+00  6.211e-01  -2.664 0.007719 **
## ill            2.464e-01  8.880e-02   2.775 0.005520 **
## ad             1.876e+00  2.741e-01   6.845 7.63e-12 ***
## hs            -1.455e-01  1.501e-01  -0.969 0.332364
## insurance2    -1.490e+01  8.174e+02  -0.018 0.985461
## insurance3    -7.919e-03  3.601e-01  -0.022 0.982455
## insurance4    -9.788e-01  4.570e-01  -2.142 0.032224 *
```

```
## ch2           -1.685e-03  4.404e-01  -0.004 0.996946
## ch3           -7.743e-01  4.150e-01  -1.866 0.062055 .
## income:ch2     1.122e+00  9.487e-01   1.183 0.236871
## income:ch3     2.540e+00  7.071e-01   3.591 0.000329 ***
## hs:insurance2  1.173e-01  3.374e+02   0.000 0.999723
## hs:insurance3 -6.112e-03  8.972e-02  -0.068 0.945688
## hs:insurance4  5.632e-01  1.257e-01   4.481 7.42e-06 ***
## ad:insurance2 -1.987e+00  2.521e+03  -0.001 0.999371
## ad:insurance3 -1.031e+00  4.073e-01  -2.531 0.011380 *
## ad:insurance4  6.000e-01  5.015e-01   1.197 0.231477
## age:hs         5.239e-01  2.356e-01   2.224 0.026172 *
## ill:hs        -4.040e-02  2.315e-02  -1.745 0.081020 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 598.72  on 499  degrees of freedom
## Residual deviance: 337.95  on 478  degrees of freedom
## AIC: 586.99
##
## Number of Fisher Scoring iterations: 15
```

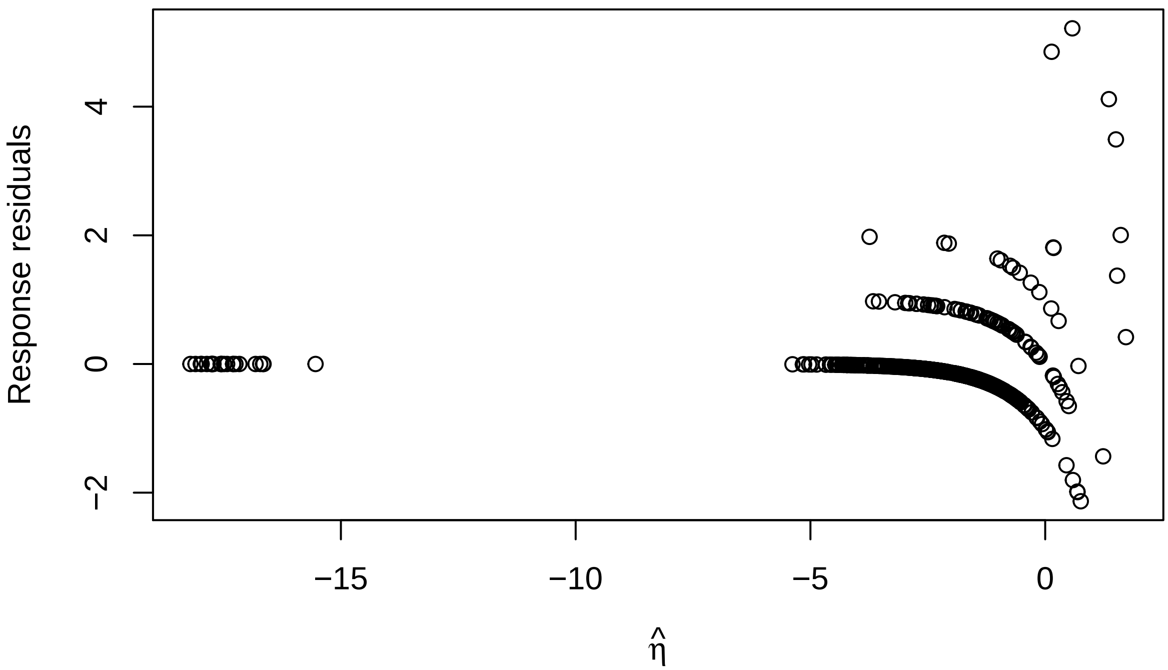
```r
# embedded test
anova(fit3,fit3.2,test="Chi")
```

```
## Analysis of Deviance Table
##
## Model 1: pc ~ sex + age + income + ill + ad + hs + insurance + ch + income:ch +
##     hs:insurance + ad:insurance + age:hs + ill:hs + age:ad +
##     ad:ch + ill:ch
## Model 2: pc ~ sex + age + income + ill + ad + hs + insurance + ch + income:ch +
##     hs:insurance + ad:insurance + age:hs + ill:hs
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       473     324.70
## 2       478     337.95 -5  -13.255   0.0211 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

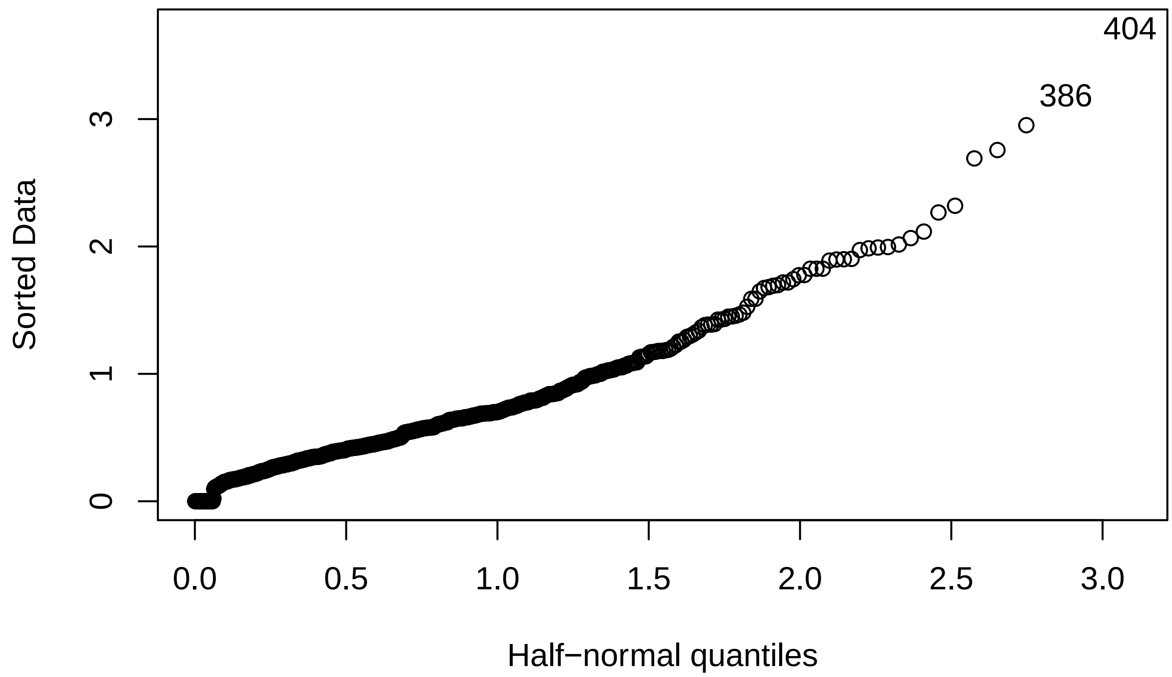
```r
diagFun(fit3.2)
```



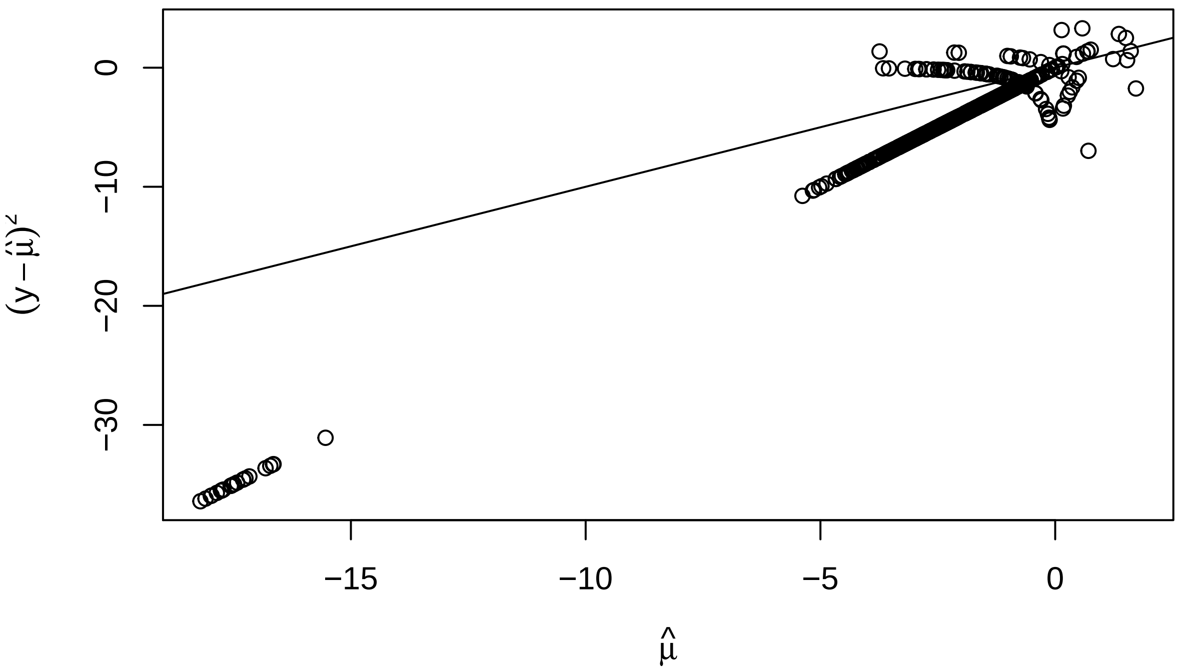
Deviance residuals
3
2
1
0
−1
−2
−15
−10
−5
0
$\hat{\eta}$



Response residuals
4
2
0
−2
−15
−10
−5
0
$\hat{\eta}$

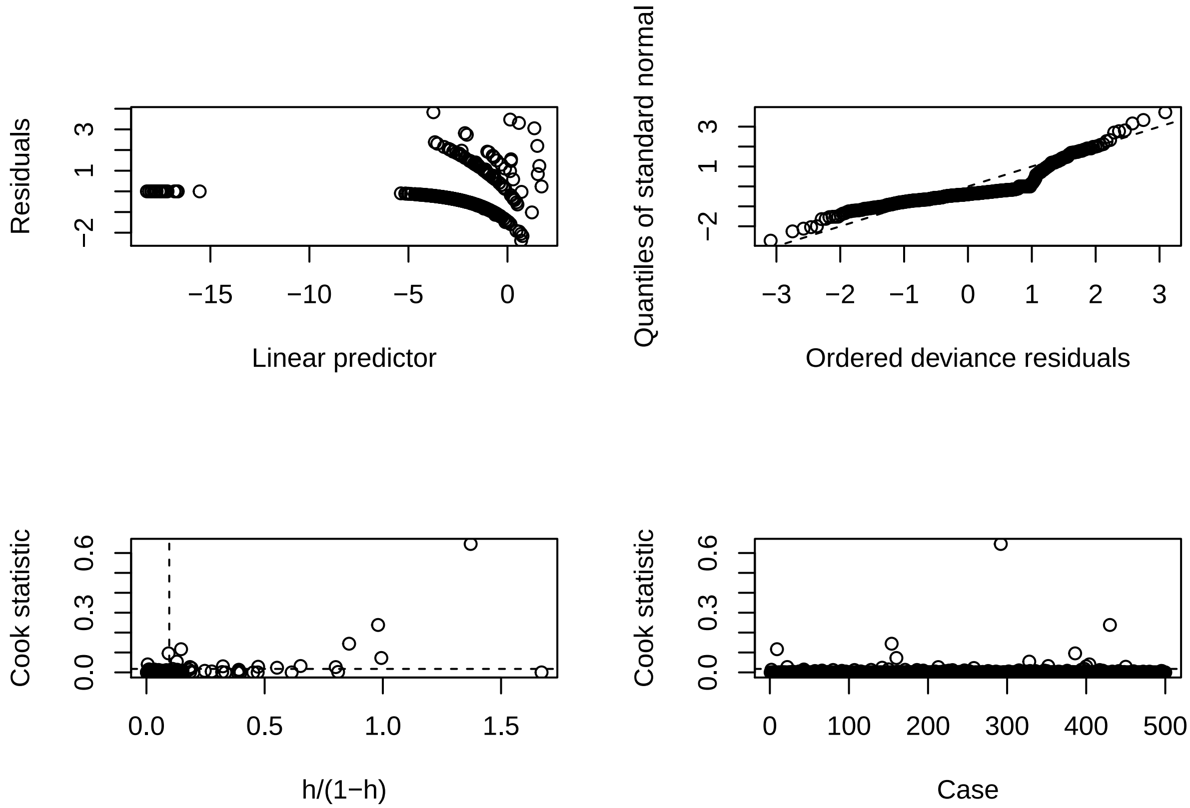


404
386
Sorted Data
0
1
2
3
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Half−normal quantiles




```
##   estimated_phi_pearson estimated_phi_deviance dispersion_test_p  Goodness
## 1              1.488056              0.7070147         0.1082517 0.9999998
```

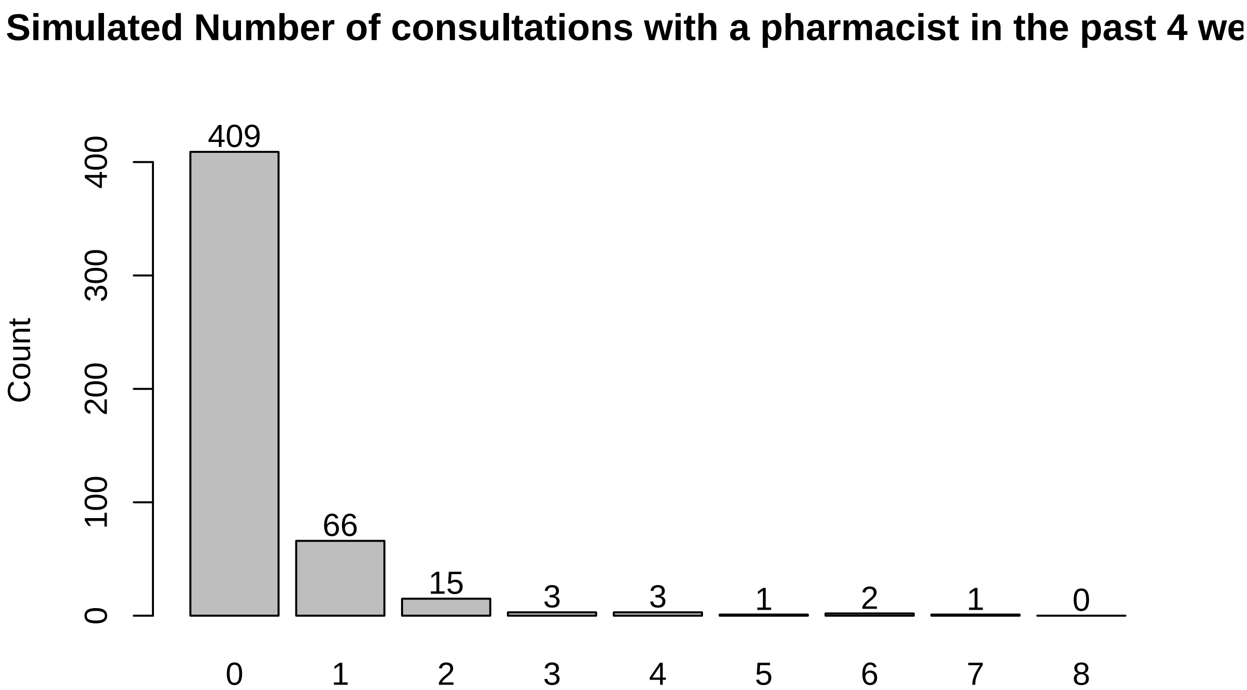
```
glm.diag.plots(fit3.2)
```



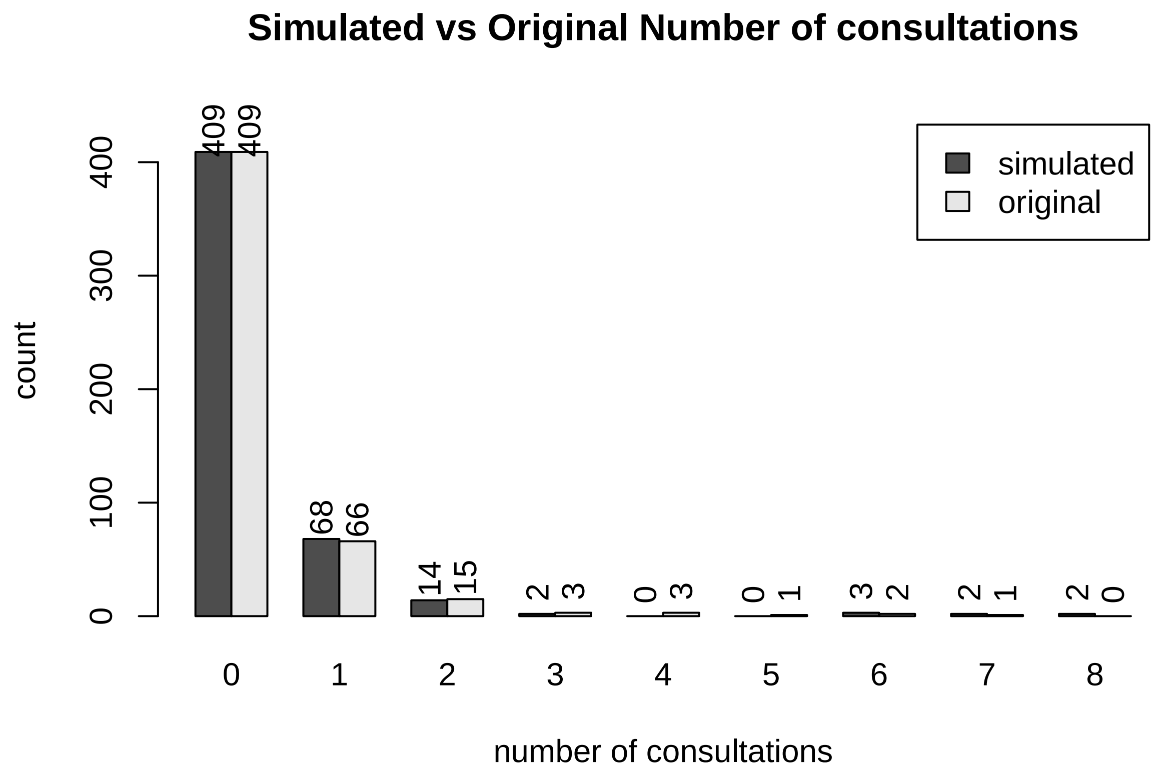

```r
outlierTest(fit3.2)
```

```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##     rstudent unadjusted p-value Bonferroni p
## 404 3.829043         0.00012864     0.064321
```

```r
# use simulation to see whether our model describe the data well
predmu <- predict(fit3.2,type="response")
set.seed(111)
simulate <- rpois(length(predmu),predmu)

newtable2 <- c(table(simulate),0)
names(newtable2) <- 0:8
x <- barplot(newtable2,main = "Simulated Number of consultations with a pharmacist in the past 4 weeks"
y <- newtable2
text(x,y+14,labels=as.character(y))
```




```
simulatedf <- data.frame(pc = c(p3$pc,simulate),sim = rep(c(0,1),each=500))
newtable3 <- table(simulatedf)

x <- barplot(t(newtable3),beside = T, main="Simulated vs Original Number of consultations",legend.text =
text(x[1,],newtable3[,1]+19,labels=as.character(newtable3[,1]),srt=90)
text(x[2,],newtable3[,2]+19,labels=as.character(newtable3[,2]),srt=90)
```

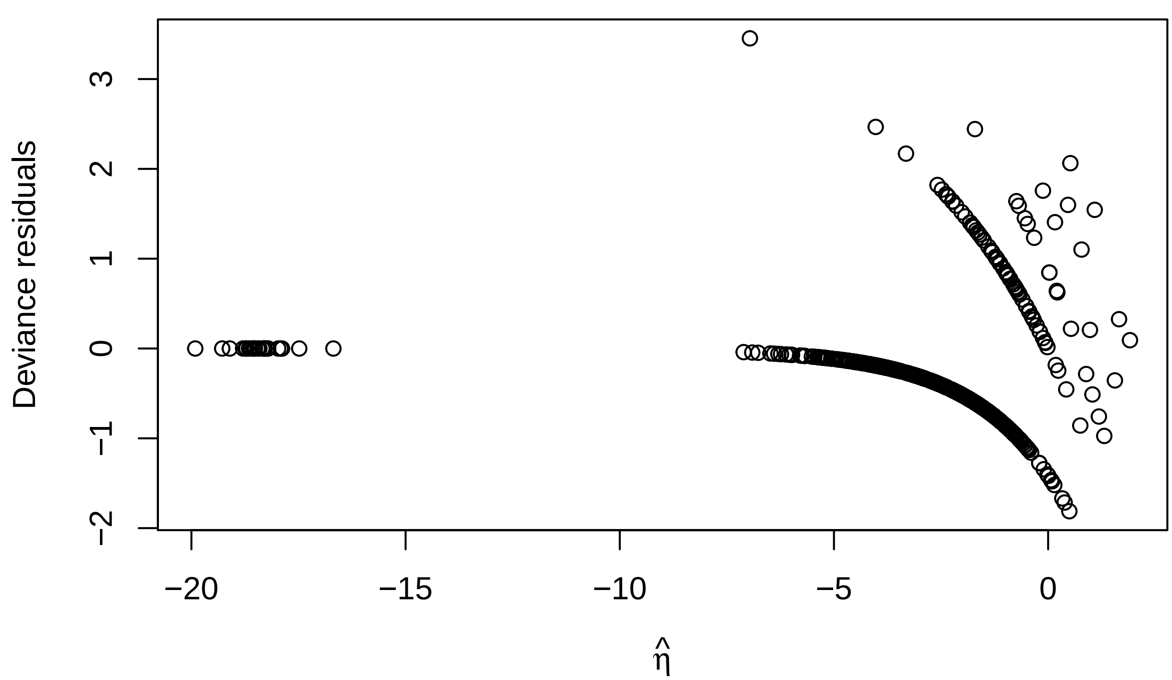


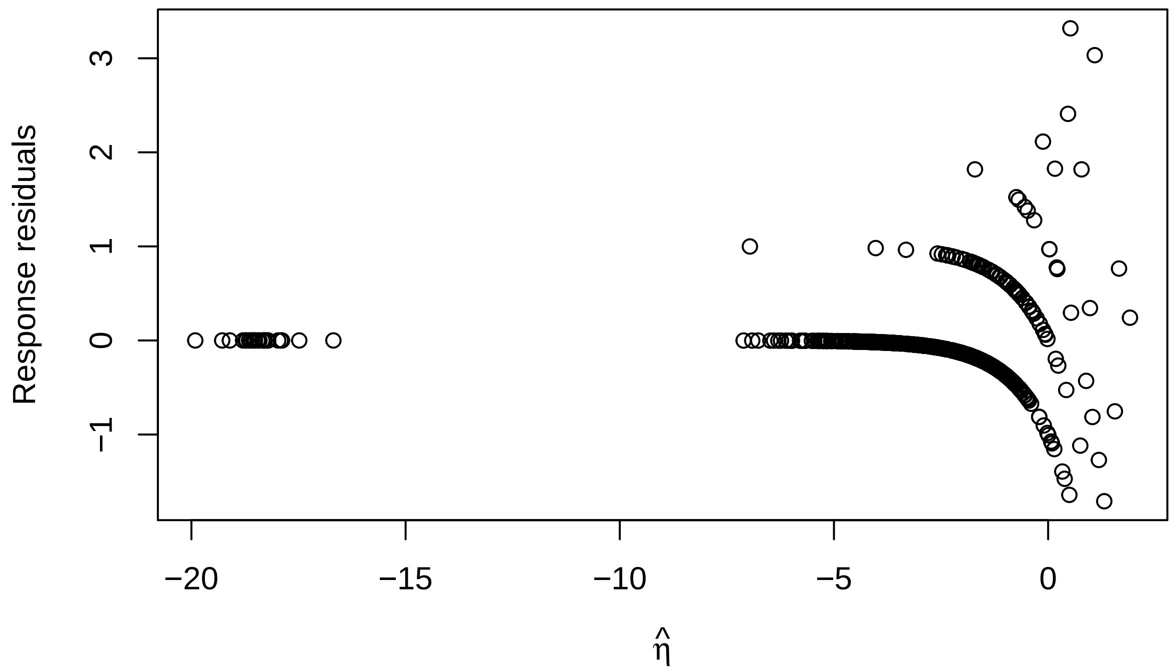

```r
# fit the same model and see the residual plots
p4 <- p3
p4$pc <- simulate
fitsim <- glm(pc ~ sex + age + income + ill + ad + hs + insurance + ch + income:ch + hs:insurance + ad:i
summary(fitsim)
```

```
##
## Call:
## glm(formula = pc ~ sex + age + income + ill + ad + hs + insurance +
##     ch + income:ch + hs:insurance + ad:insurance + age:hs + ill:hs,
##     family = "poisson", data = p4)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8120  -0.5599  -0.2474  -0.0972   3.4533
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -1.845e+00  5.723e-01  -3.224 0.001264 **
## sex              5.747e-01  2.220e-01   2.588 0.009649 **
## age              6.707e-01  8.042e-01   0.834 0.404300
## income          -3.213e+00  8.321e-01  -3.861 0.000113 ***
## ill              1.469e-01  9.394e-02   1.564 0.117781
## ad               1.930e+00  3.199e-01   6.034 1.60e-09 ***
## hs              -3.212e-01  1.734e-01  -1.853 0.063941 .
## insurance2      -1.601e+01  1.376e+03  -0.012 0.990717
```

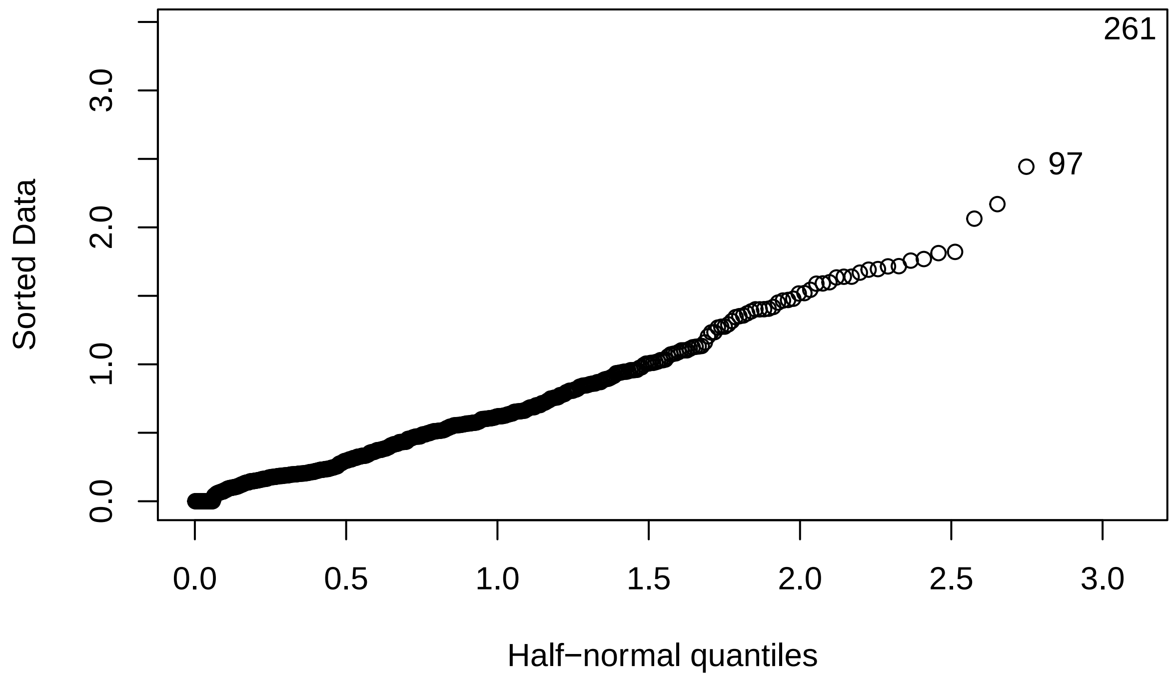
```
## insurance3     1.749e-01  3.764e-01   0.465 0.642102    
## insurance4    -1.877e+00  6.439e-01  -2.915 0.003553 ** 
## ch2           -4.702e-02  4.726e-01  -0.099 0.920744    
## ch3           -1.141e+00  4.405e-01  -2.589 0.009614 ** 
## income:ch2     1.553e+00  1.256e+00   1.236 0.216495    
## income:ch3     3.646e+00  9.257e-01   3.938 8.20e-05 ***
## hs:insurance2  2.350e-01  6.133e+02   0.000 0.999694    
## hs:insurance3  4.179e-02  1.029e-01   0.406 0.684571    
## hs:insurance4  6.746e-01  1.491e-01   4.524 6.05e-06 ***
## ad:insurance2 -2.626e+00  4.606e+03  -0.001 0.999545    
## ad:insurance3 -9.310e-01  4.198e-01  -2.218 0.026576 *  
## ad:insurance4  1.536e+00  6.632e-01   2.317 0.020509 *  
## age:hs         5.928e-01  2.645e-01   2.241 0.025017 *  
## ill:hs         2.133e-03  2.190e-02   0.097 0.922388    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for poisson family taken to be 1)
## 
##     Null deviance: 537.94  on 499  degrees of freedom
## Residual deviance: 244.60  on 478  degrees of freedom
## AIC: 493.18
## 
## Number of Fisher Scoring iterations: 16
```

```r
diagFun(fitsim)
```

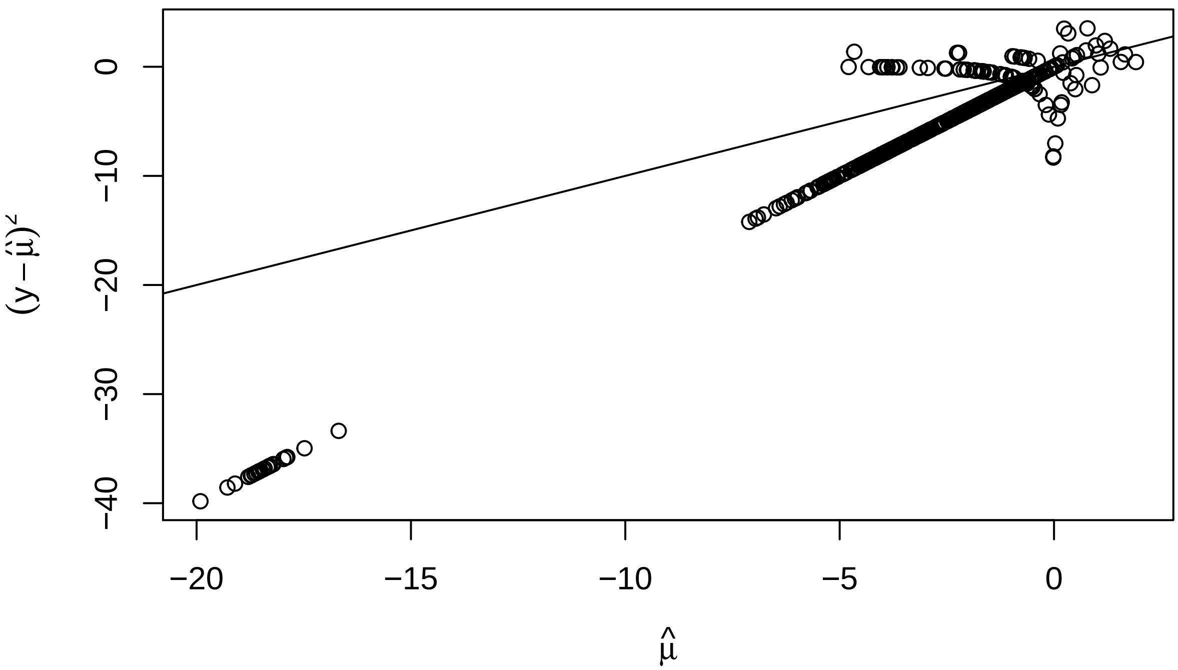




Response residuals
3
2
1
0
−1
−20
−15
−10
−5
0
η̂



261
97
Sorted Data
0.0
1.0
2.0
3.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Half−normal quantiles




```
##   estimated_phi_pearson estimated_phi_deviance dispersion_test_p Goodness
## 1              2.933664              0.5117131           0.98827        1
```

```r
outlierTest(fitsim)
```

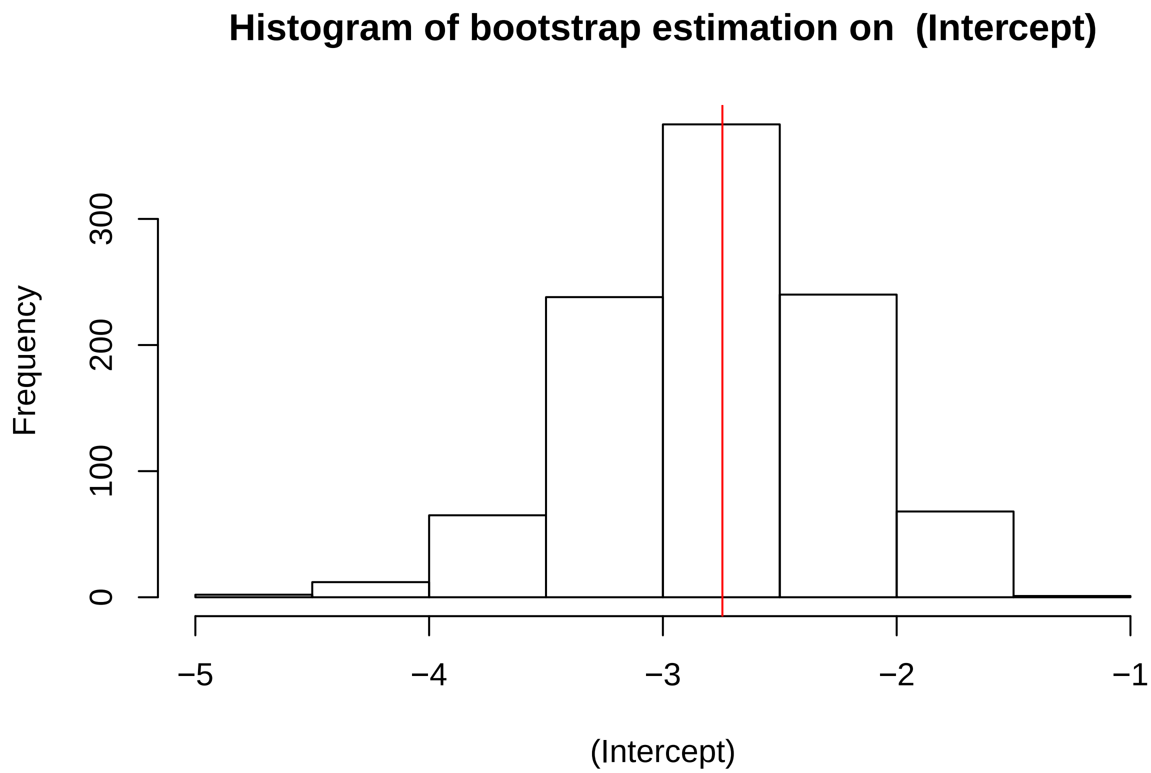
```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##     rstudent unadjusted p-value Bonferroni p
## 261 3.569361         0.00035785      0.17893
```

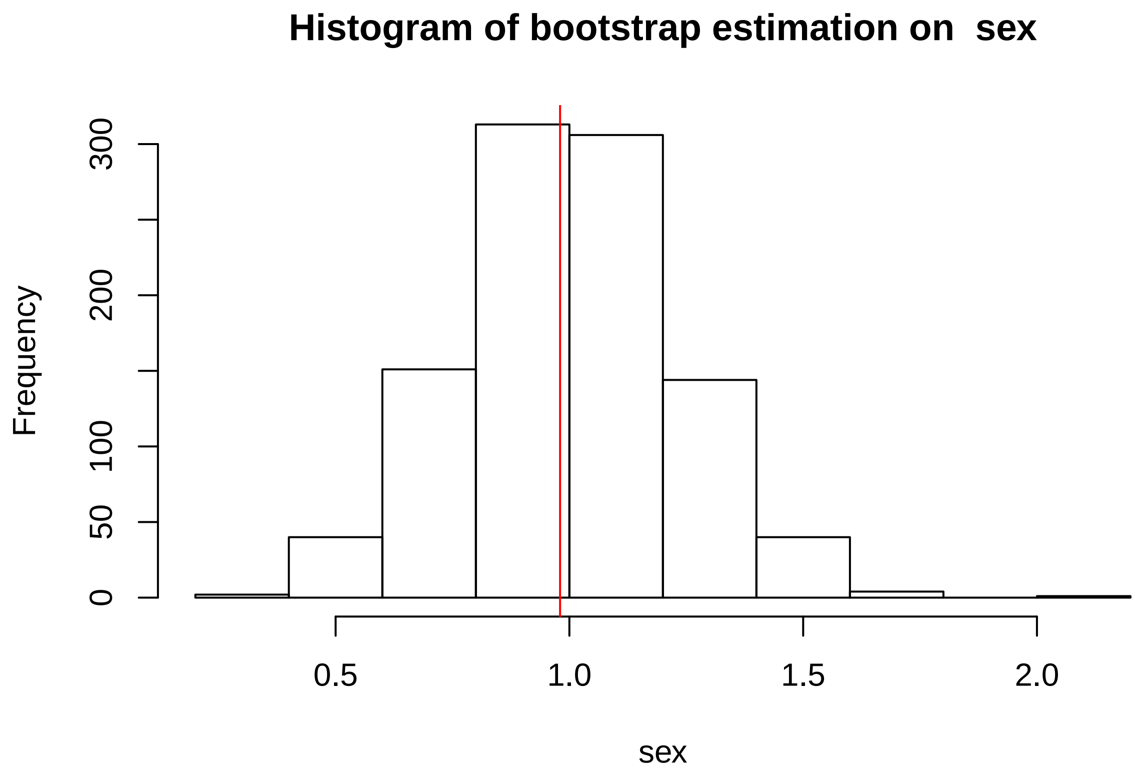
```r
# bootstrap regression

for (i in 1:1000){

  simulate <- rpois(length(predmu),predmu)
  psim <- p3
  psim$pc <- simulate
  fitsim <- glm(pc ~ sex + age + income + ill + ad + hs + insurance + ch + income:ch + hs:insurance + ad
  if (i == 1){
    result <- fitsim$coefficients
  }
  result <- rbind(result,fitsim$coefficients)

}

## find 95% quantile
qdf <- data.frame("0.95lower"=rep(NA,22),"0.95upper"=NA)
```

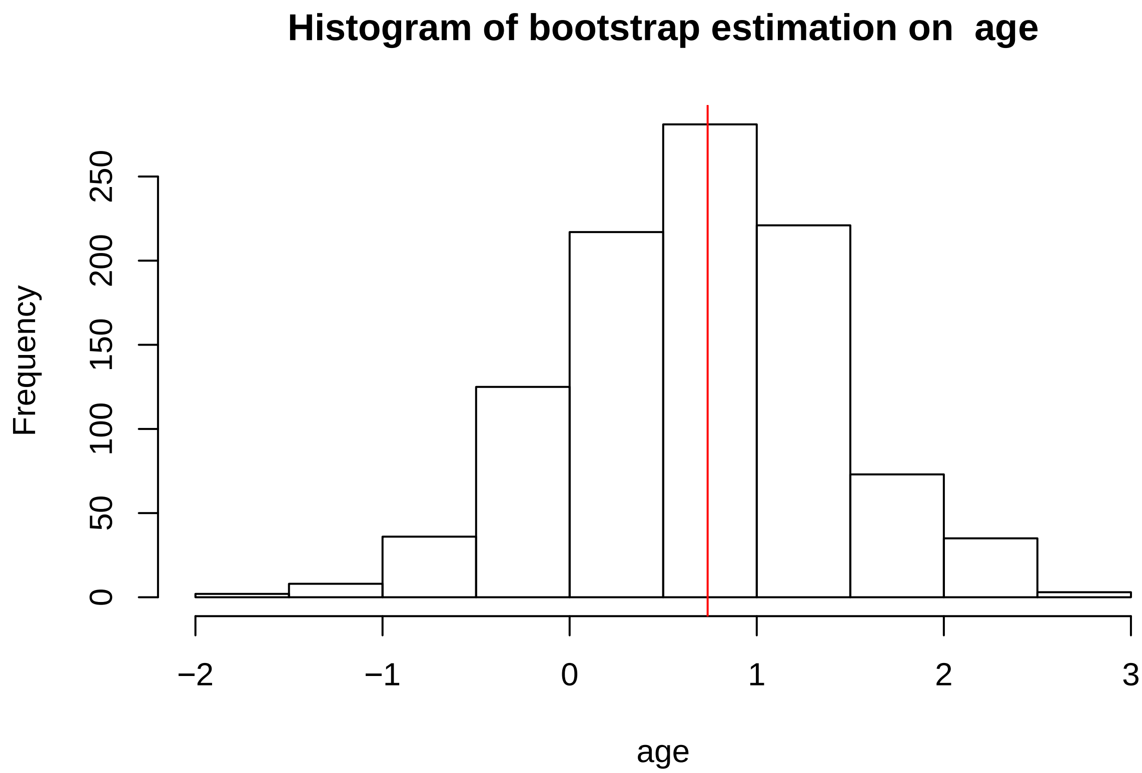
```r
for(i in 1:ncol(result)){
  q <- quantile(result[,i],probs=c(0.025,0.975))
  qdf[i,1] <- q[1]
  qdf[i,2] <- q[2]
}
qdf$est <- fit3.2$coefficients
qdf$`in` <- ifelse((qdf$est>qdf$X0.95lower)&(qdf$est<qdf$X0.95upper),"yes","no")
qdf
```

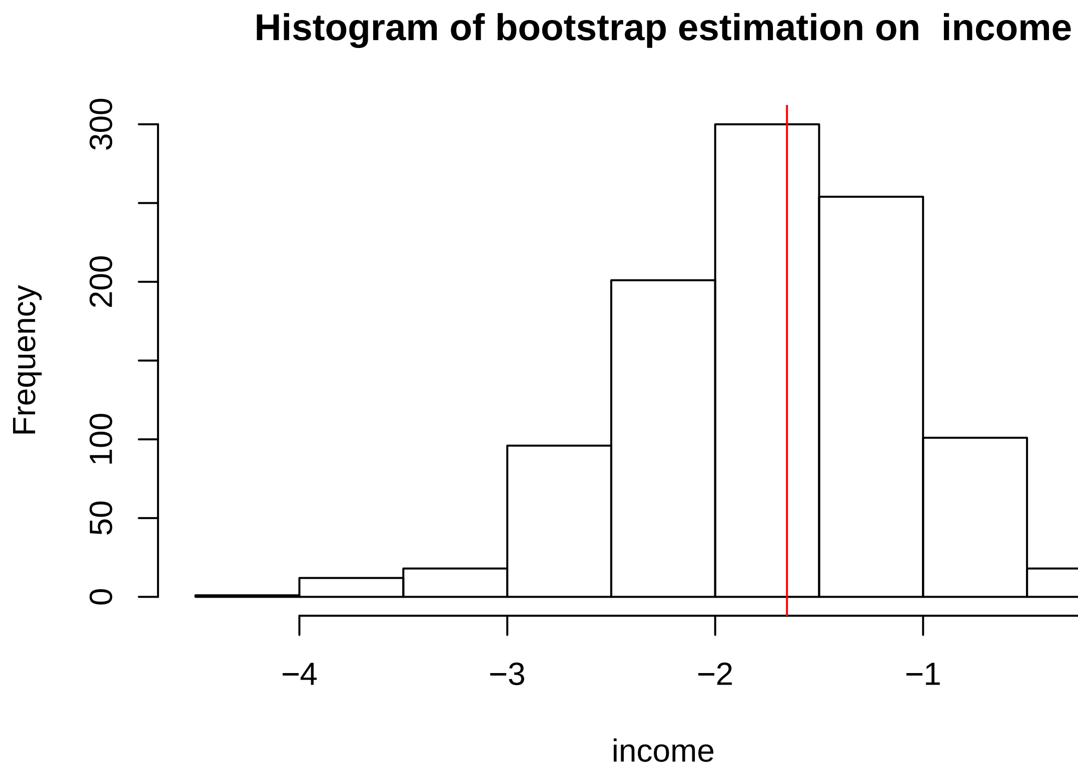
```
##      X0.95lower    X0.95upper          est  in
## 1   -3.88254694  -1.781878829  -2.745541122 yes
## 2    0.55157415   1.472433678   0.980090581 yes
## 3   -0.71725226   2.112184596   0.737616408 yes
## 4   -3.10456138  -0.550886770  -1.654705584 yes
## 5    0.07544049   0.424389961   0.246417419 yes
## 6    1.32309742   2.467779141   1.875981799 yes
## 7   -0.50896333   0.098629719  -0.145545578 yes
## 8  -16.28505752 -14.784376336 -14.895459621 yes
## 9   -0.72796478   0.713784624  -0.007918701 yes
## 10  -2.06598898  -0.174917559  -0.978788885 yes
## 11  -0.81765569   0.956234768  -0.001685487 yes
## 12  -1.62577800  -0.004226668  -0.774286868 yes
## 13  -1.45950222   2.816976844   1.122155093 yes
## 14   1.30298388   4.237751627   2.539531294 yes
## 15  -0.04483193   0.356857302   0.117315885 yes
## 16  -0.17606522   0.194722501  -0.006112167 yes
## 17   0.32707490   0.827194402   0.563174425 yes
## 18  -2.71857173  -1.354465577  -1.987315942 yes
## 19  -1.86433237  -0.199361258  -1.030806558 yes
## 20  -0.38522176   1.732887611   0.600047949 yes
## 21   0.09779776   1.090055218   0.523925034 yes
## 22  -0.09365768   0.005394905  -0.040395282 yes
```

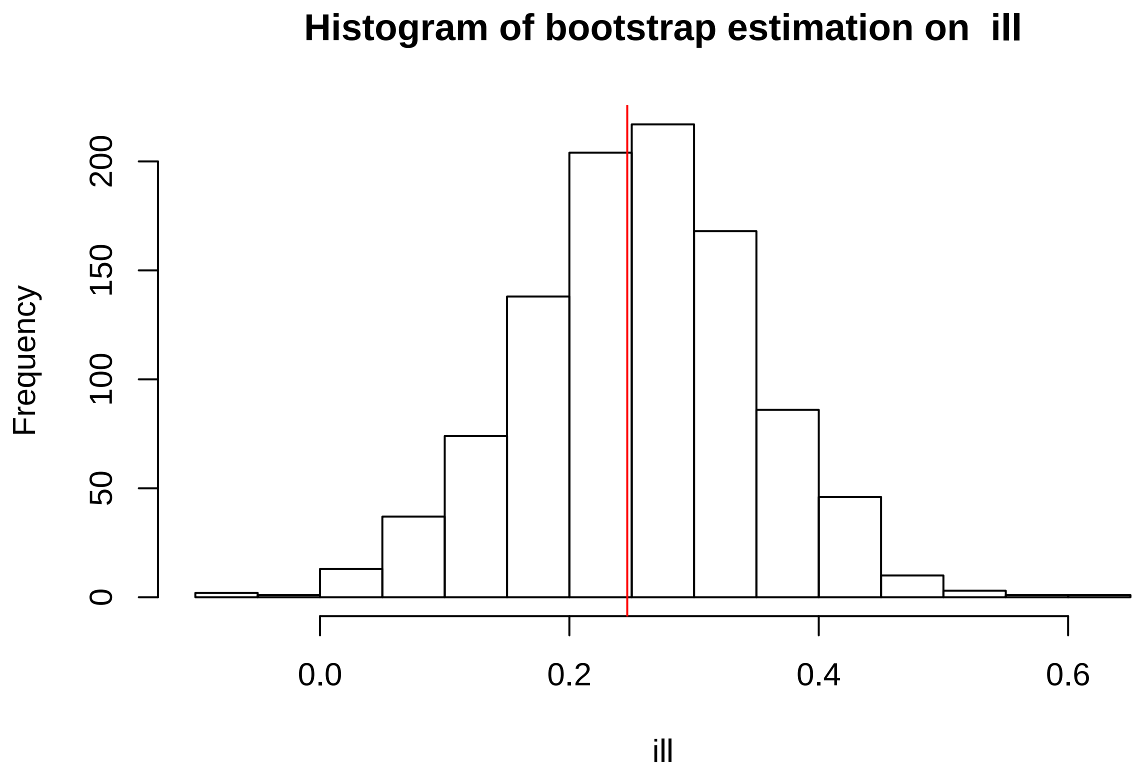
```r
## visualize
for (i in 1:ncol(result)){
  hist(result[,i],main = paste("Histogram of bootstrap estimation on ",colnames(result)[i]),xlab=colname
  abline(v = fit3.2$coefficients[i],col="red")
}
```

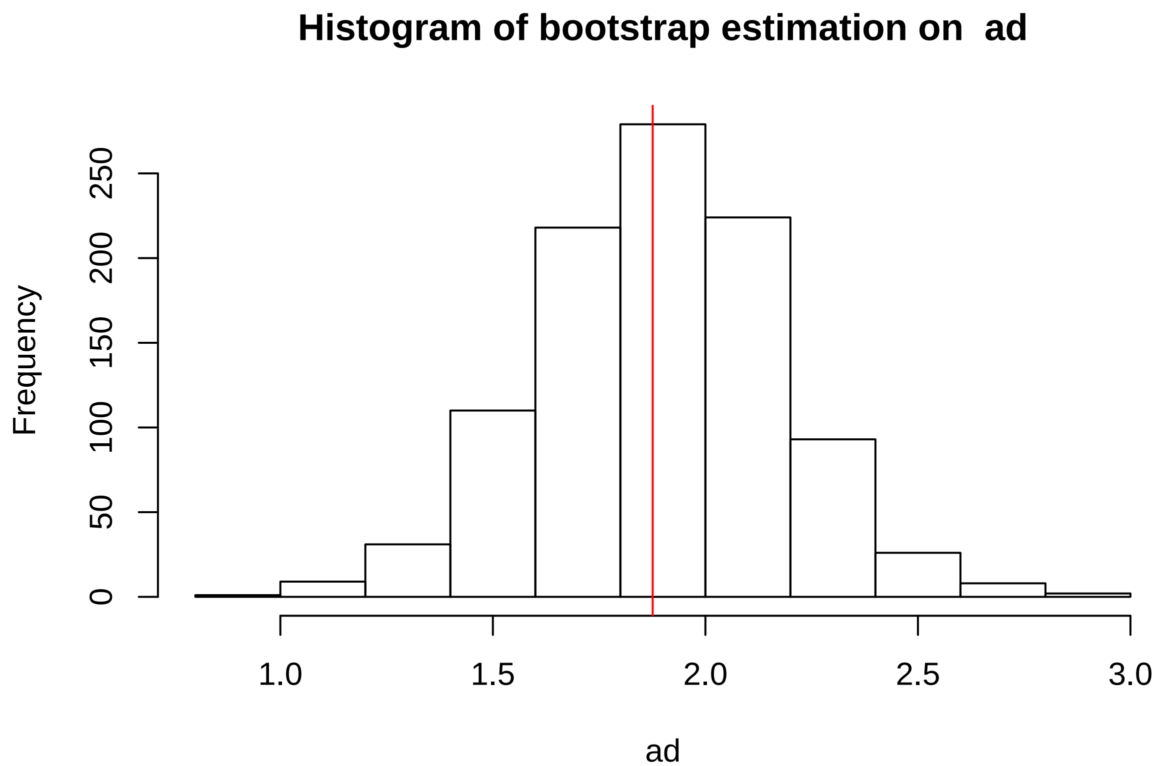

Histogram of bootstrap estimation on (Intercept)
Frequency
0
100
200
300
−5
−4
−3
−2
−1
(Intercept)


Histogram of bootstrap estimation on sex
Frequency
0
50
100
200
300
0.5
1.0
1.5
2.0
sex

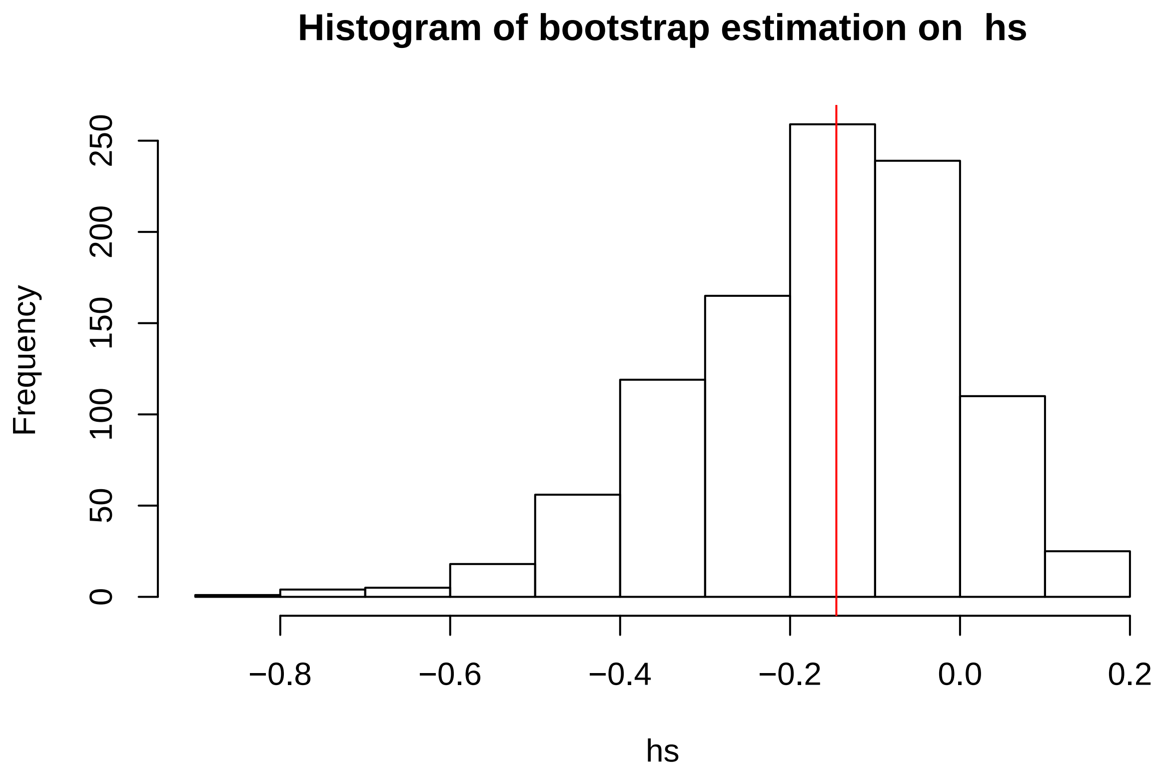

Histogram of bootstrap estimation on age
Frequency
0
50
100
150
200
250
−2
−1
0
1
2
3
age

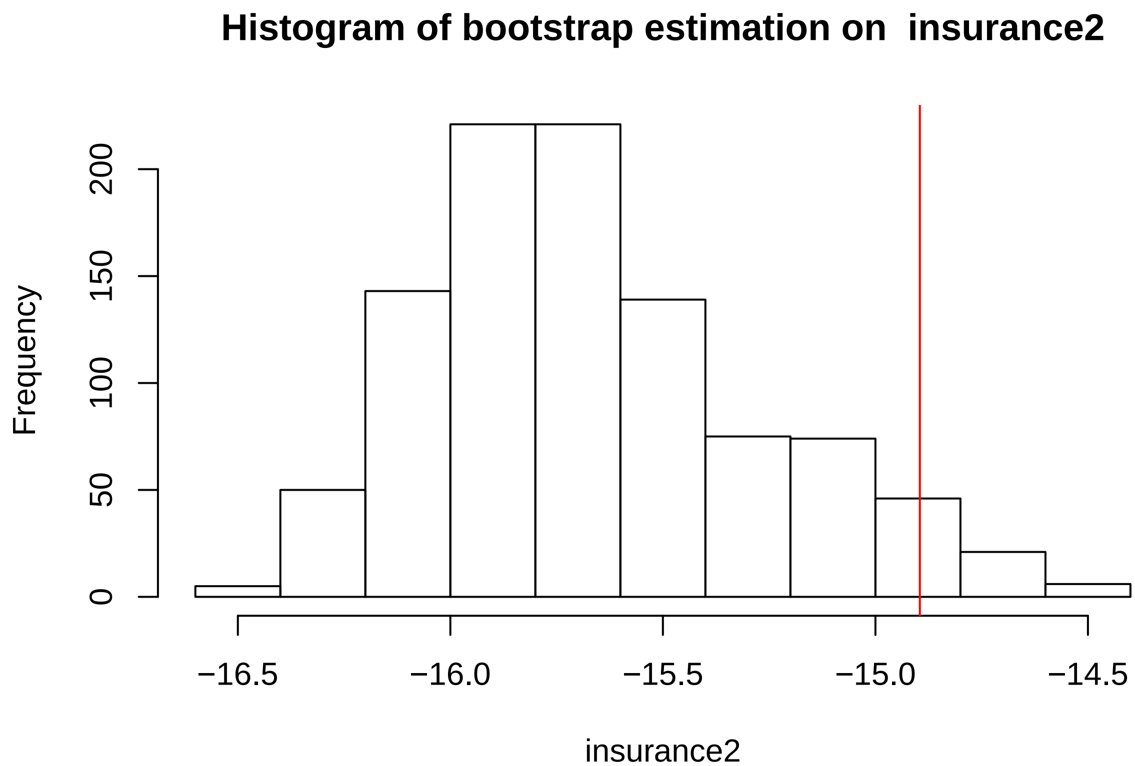

Histogram of bootstrap estimation on income
Frequency
0
50
100
200
300
−4
−3
−2
−1
0
income

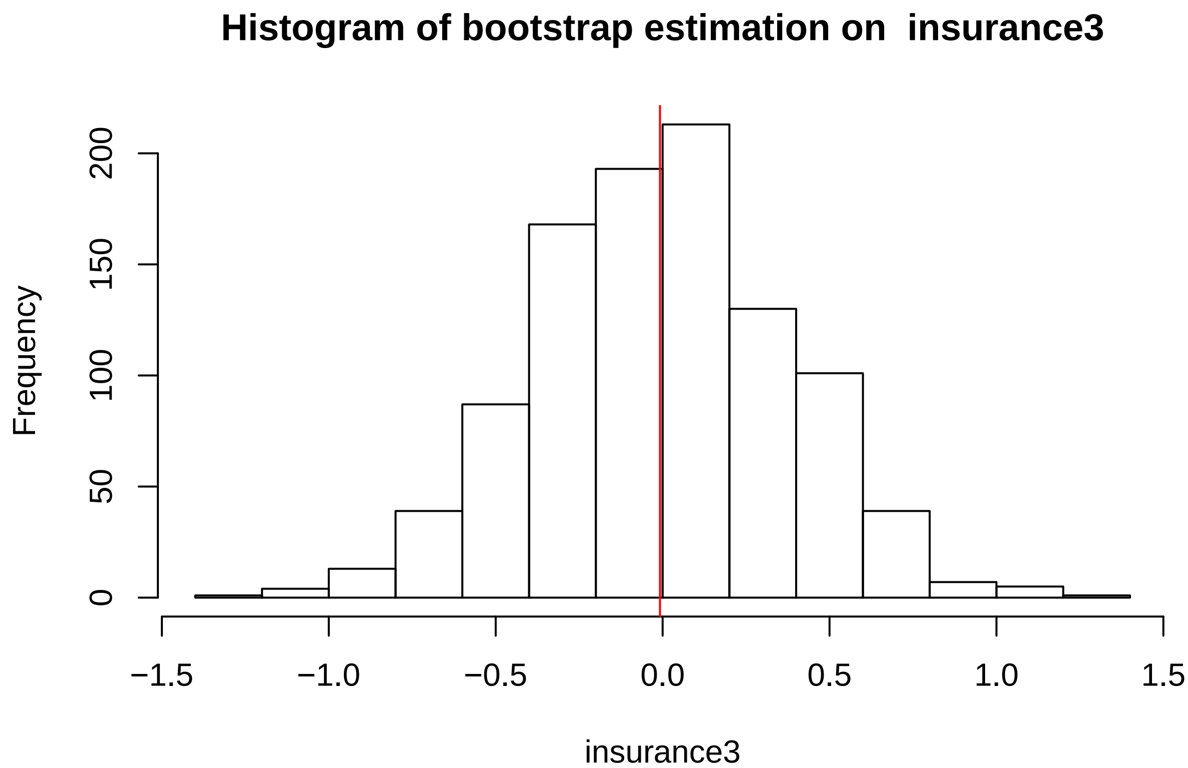

Histogram of bootstrap estimation on ill
Frequency
0
50
100
150
200
0.0
0.2
0.4
0.6
ill



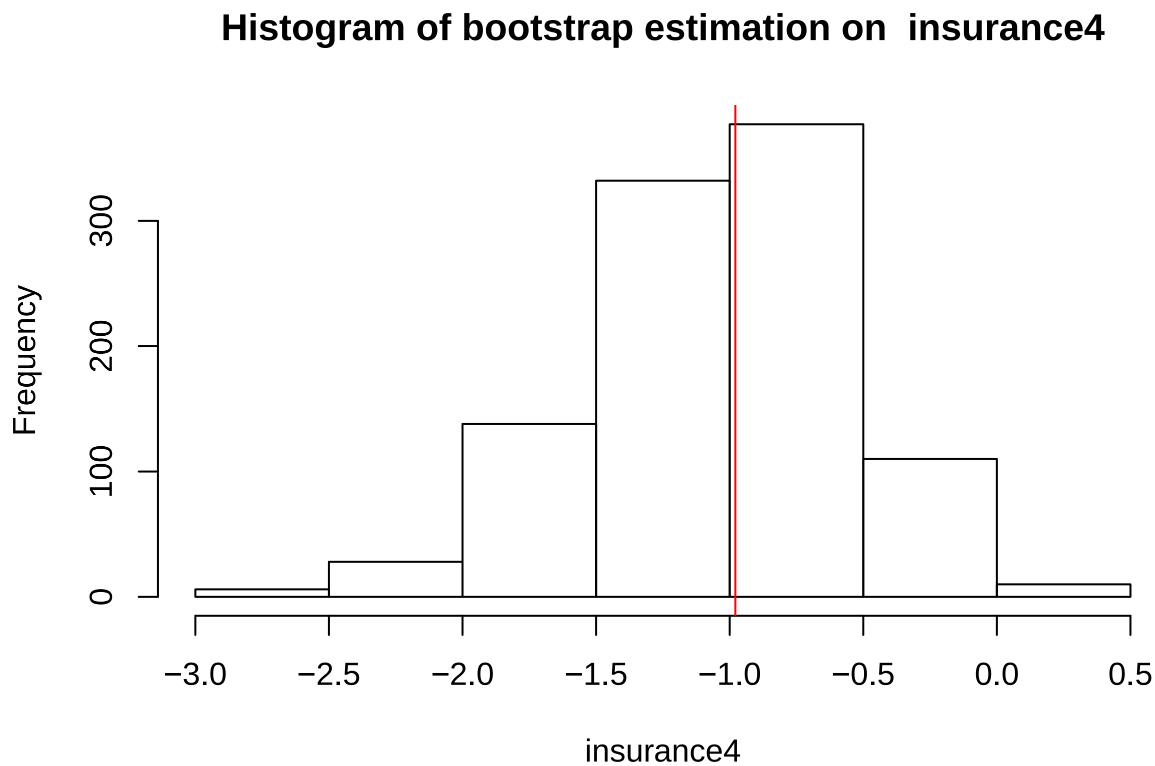
Histogram of bootstrap estimation on ad
Frequency
0
50
100
150
200
250
1.0
1.5
2.0
2.5
3.0
ad

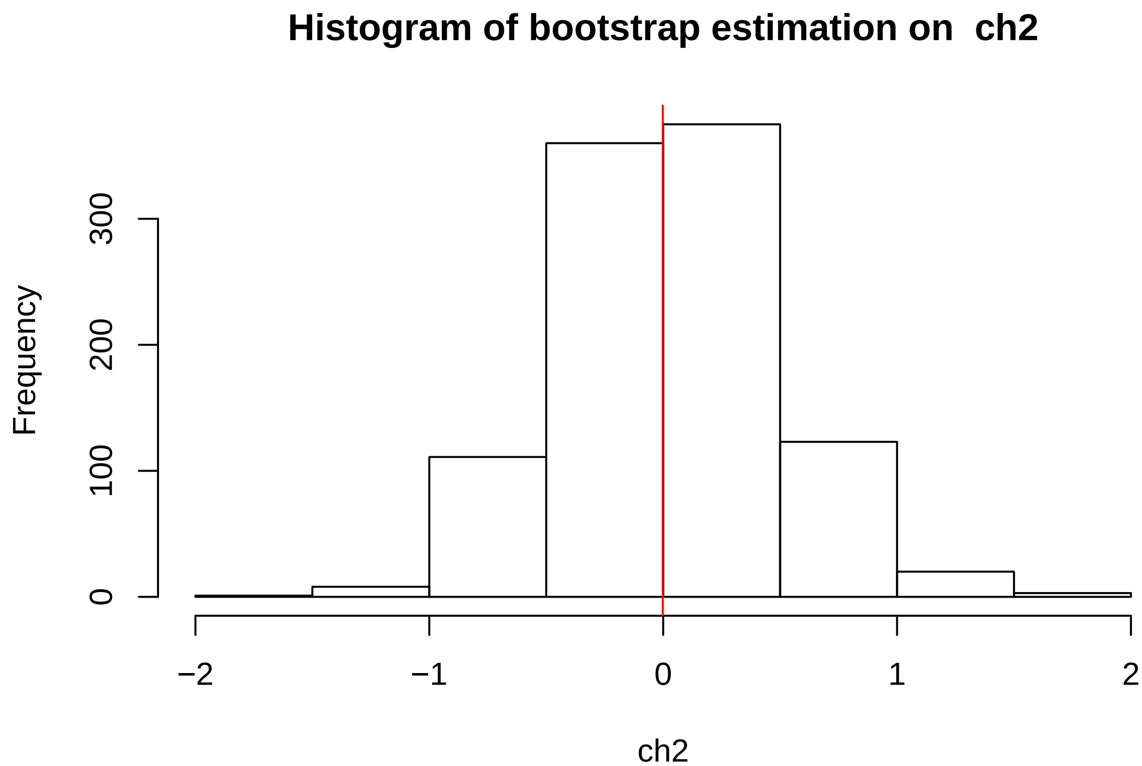

Histogram of bootstrap estimation on hs
Frequency
0
50
100
150
200
250
−0.8
−0.6
−0.4
−0.2
0.0
0.2
hs


Histogram of bootstrap estimation on insurance2
Frequency
0
50
100
150
200
−16.5
−16.0
−15.5
−15.0
−14.5
insurance2

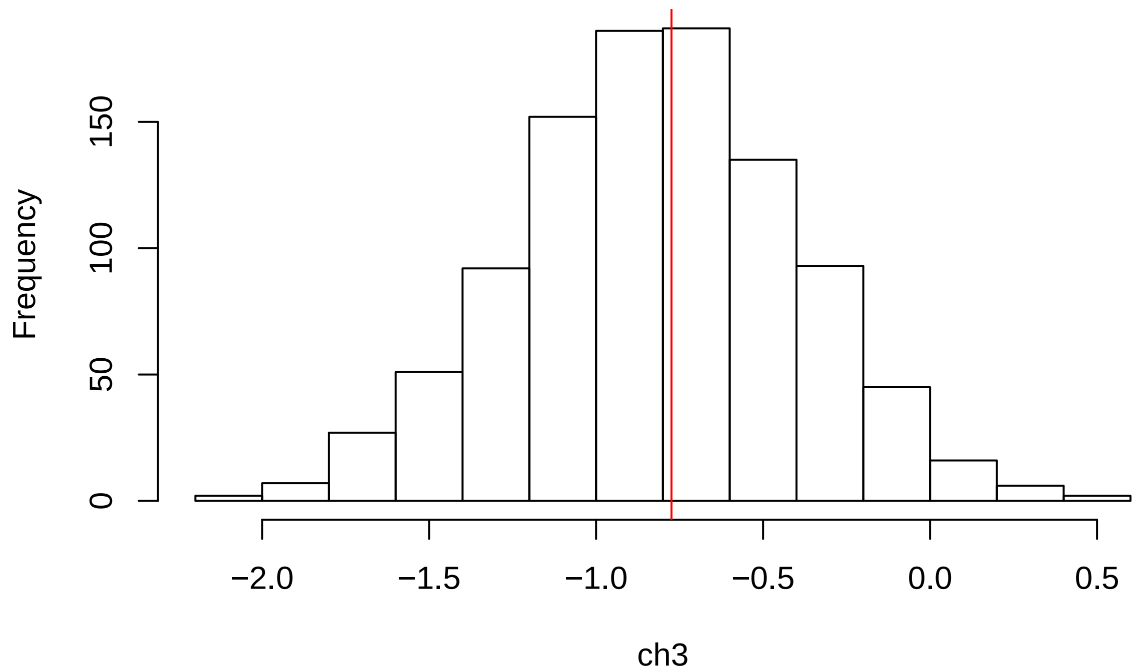


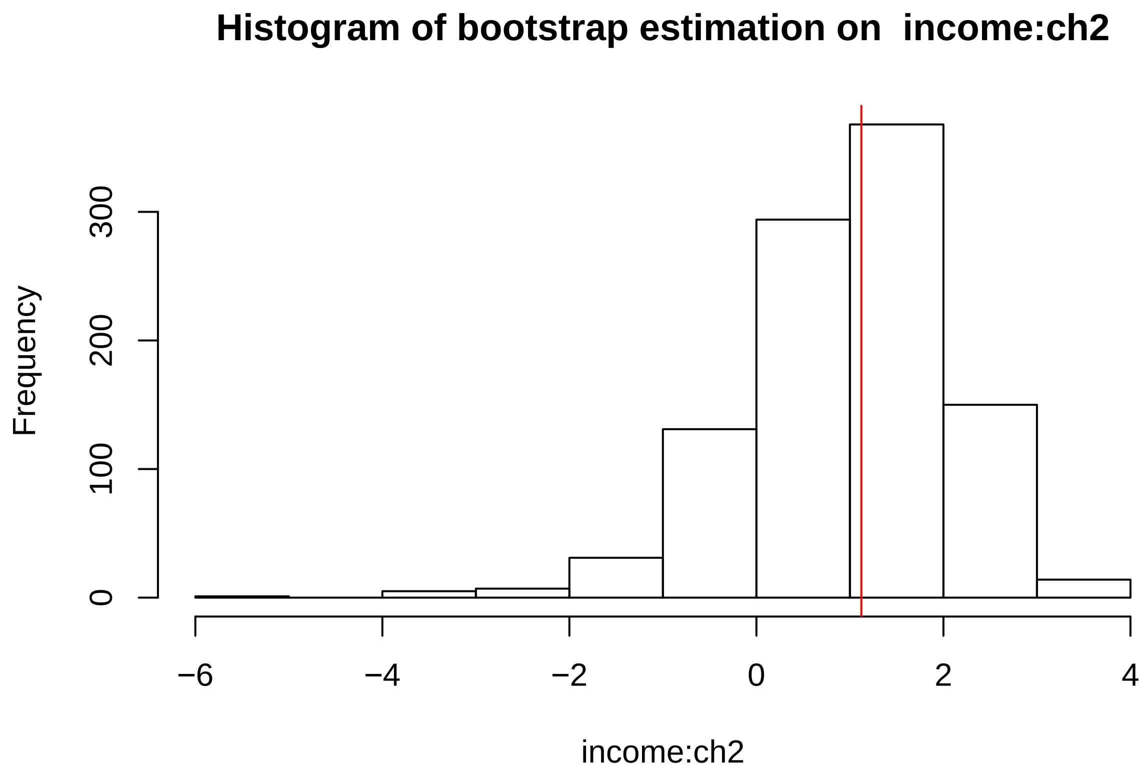
Histogram of bootstrap estimation on insurance3
Frequency
0
50
100
150
200
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
insurance3

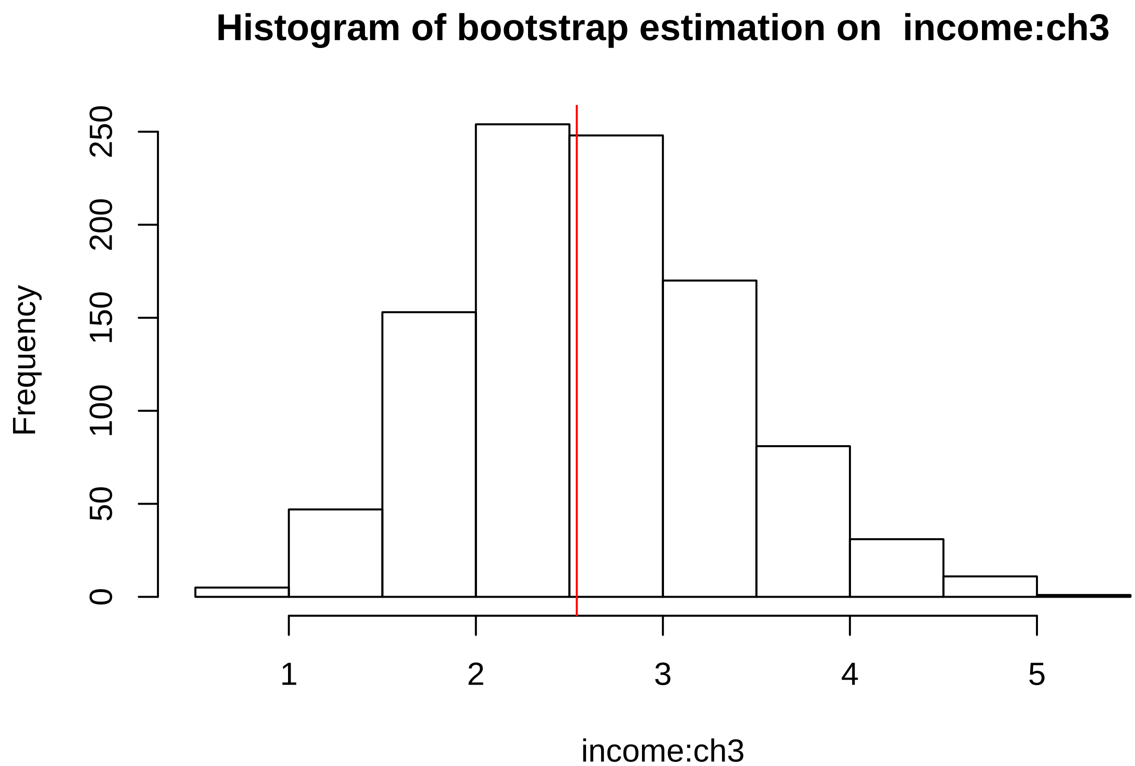

Histogram of bootstrap estimation on insurance4
Frequency
0
100
200
300
−3.0
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
0.5
insurance4

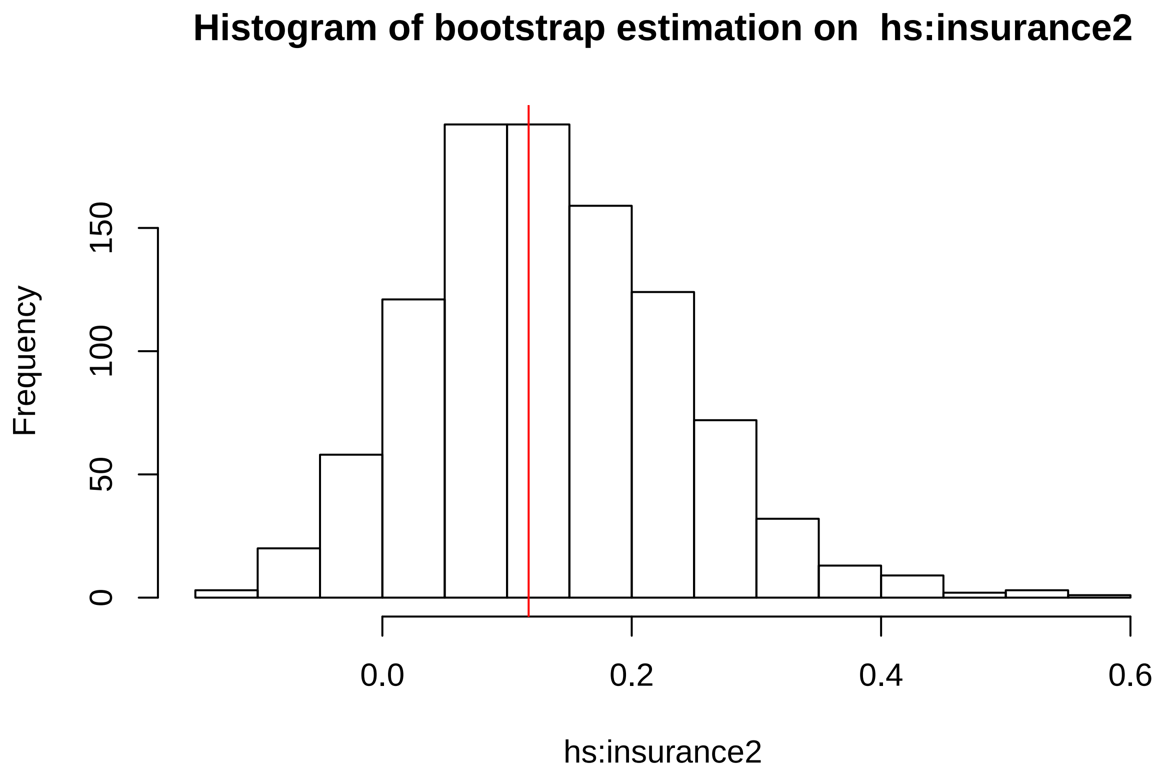

Histogram of bootstrap estimation on ch2
Frequency
0
100
200
300
−2
−1
0
1
2
ch2

Histogram of bootstrap estimation on ch3

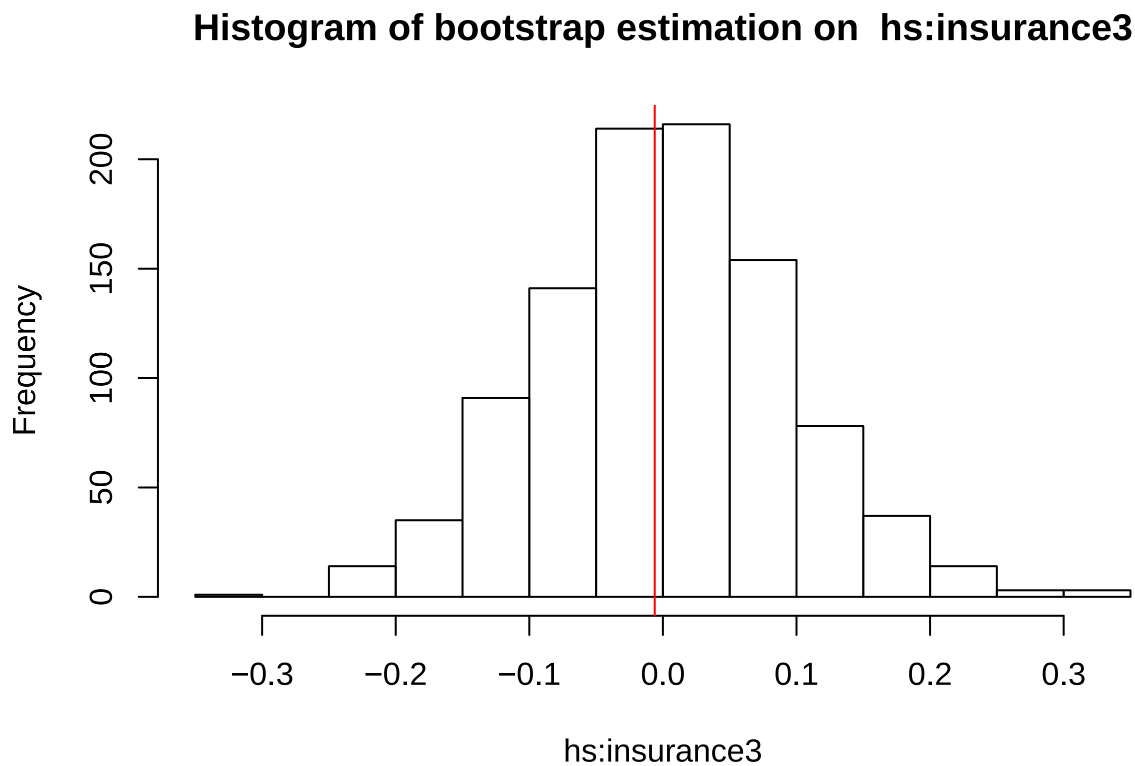



Histogram of bootstrap estimation on income:ch2
Frequency
0
100
200
300
−6
−4
−2
0
2
4
income:ch2

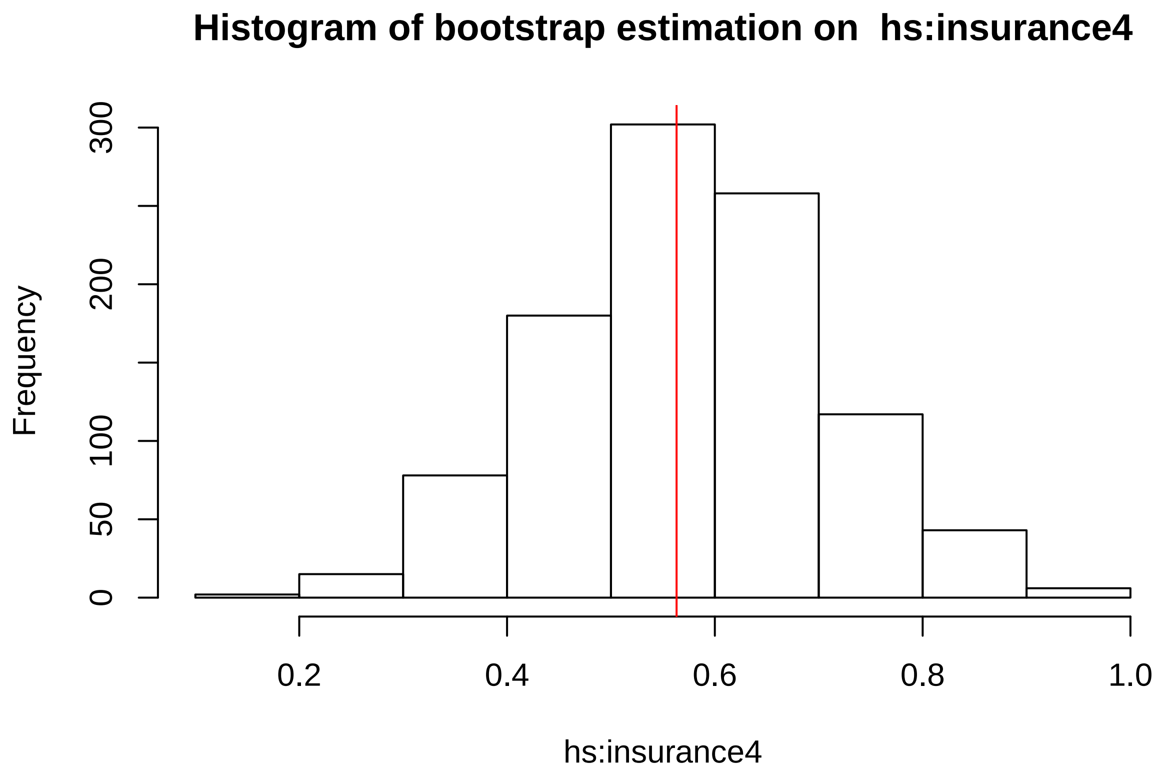

Histogram of bootstrap estimation on income:ch3
Frequency
0
50
100
150
200
250
1
2
3
4
5
income:ch3

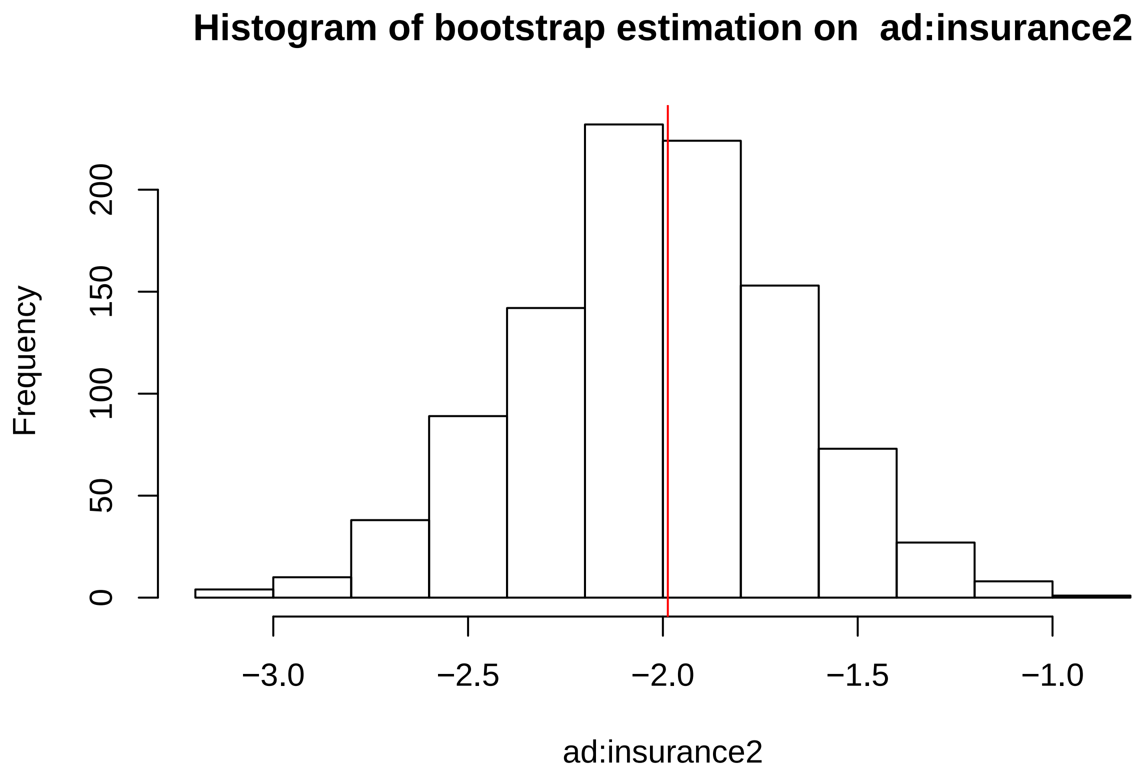


Histogram of bootstrap estimation on hs:insurance2
Frequency
0
50
100
150
0.0
0.2
0.4
0.6
hs:insurance2

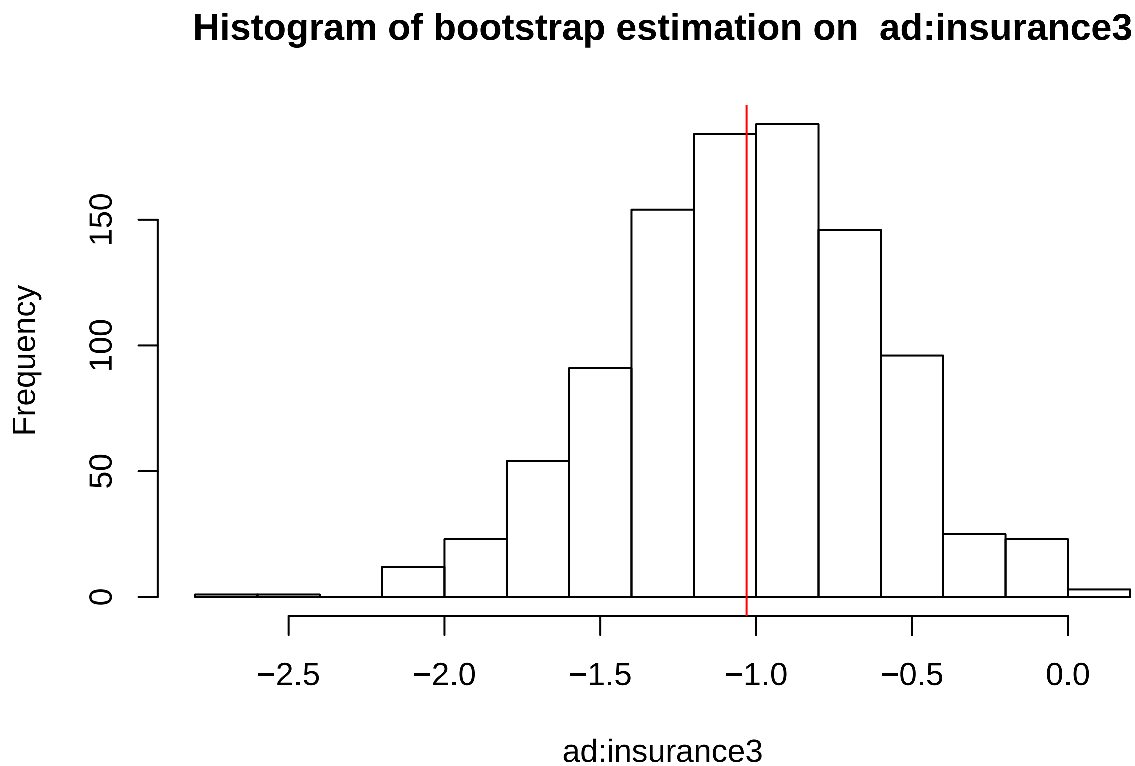


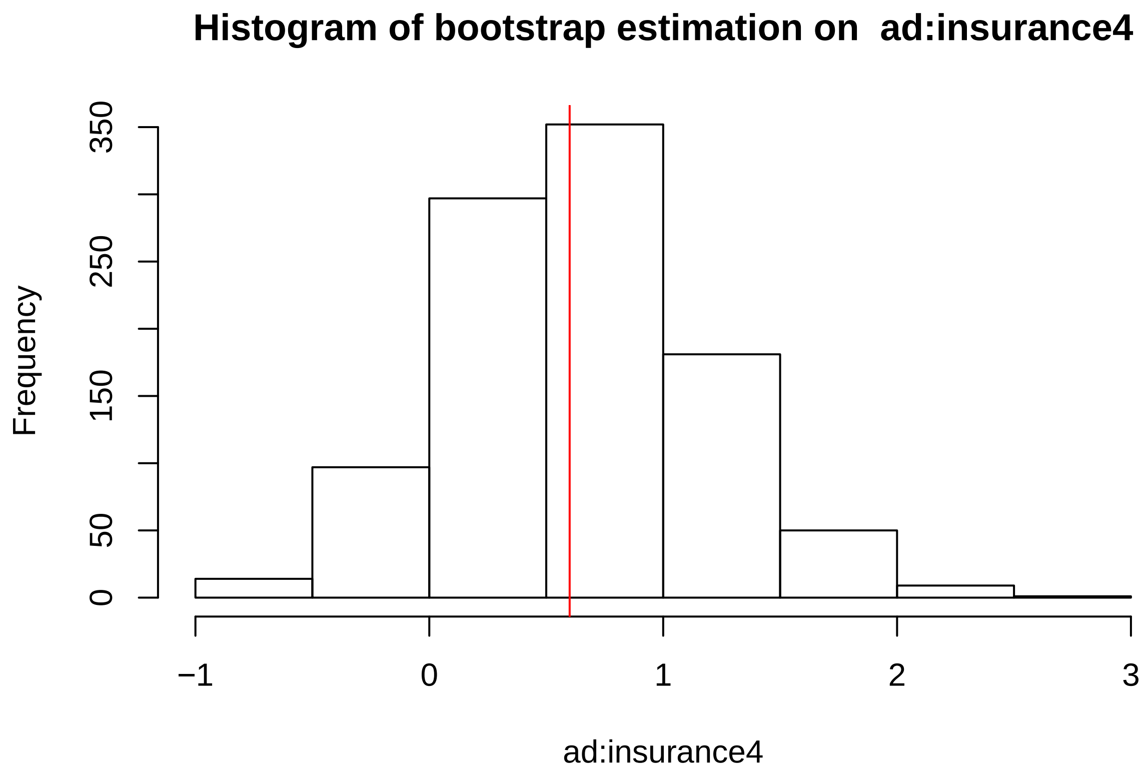
Histogram of bootstrap estimation on hs:insurance3
Frequency
0
50
100
150
200
−0.3
−0.2
−0.1
0.0
0.1
0.2
0.3
hs:insurance3

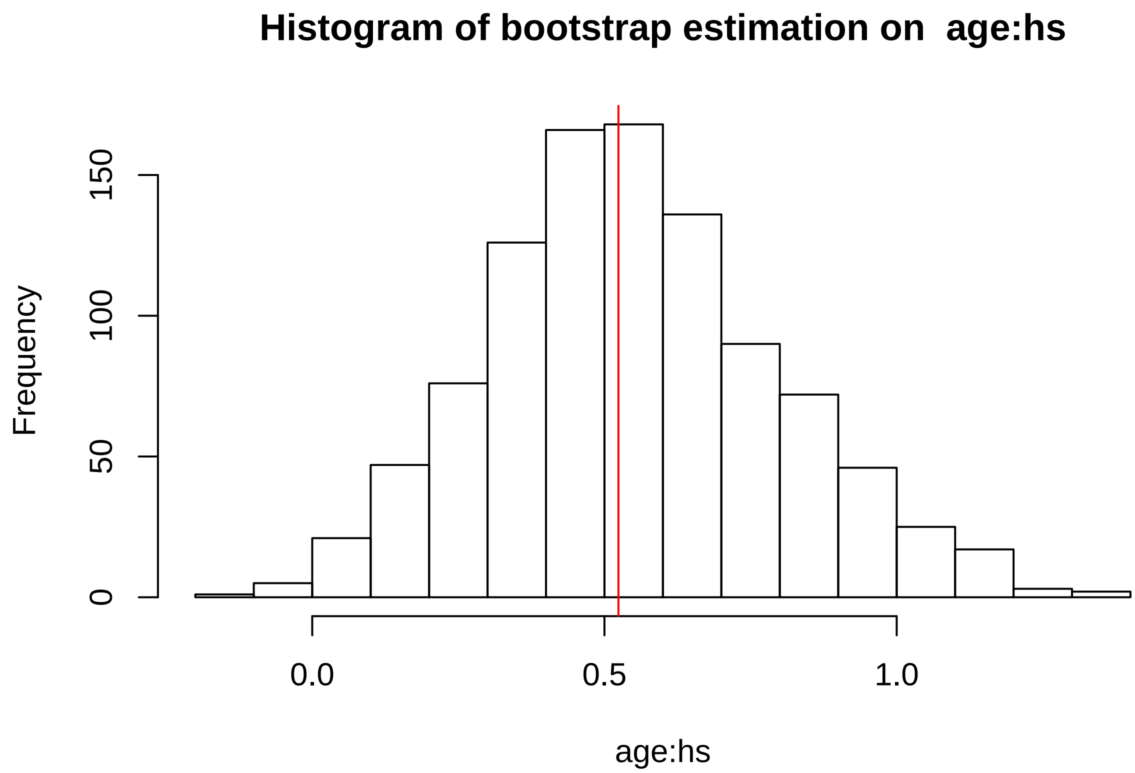

Histogram of bootstrap estimation on hs:insurance4
Frequency
0
50
100
200
300
0.2
0.4
0.6
0.8
1.0
hs:insurance4

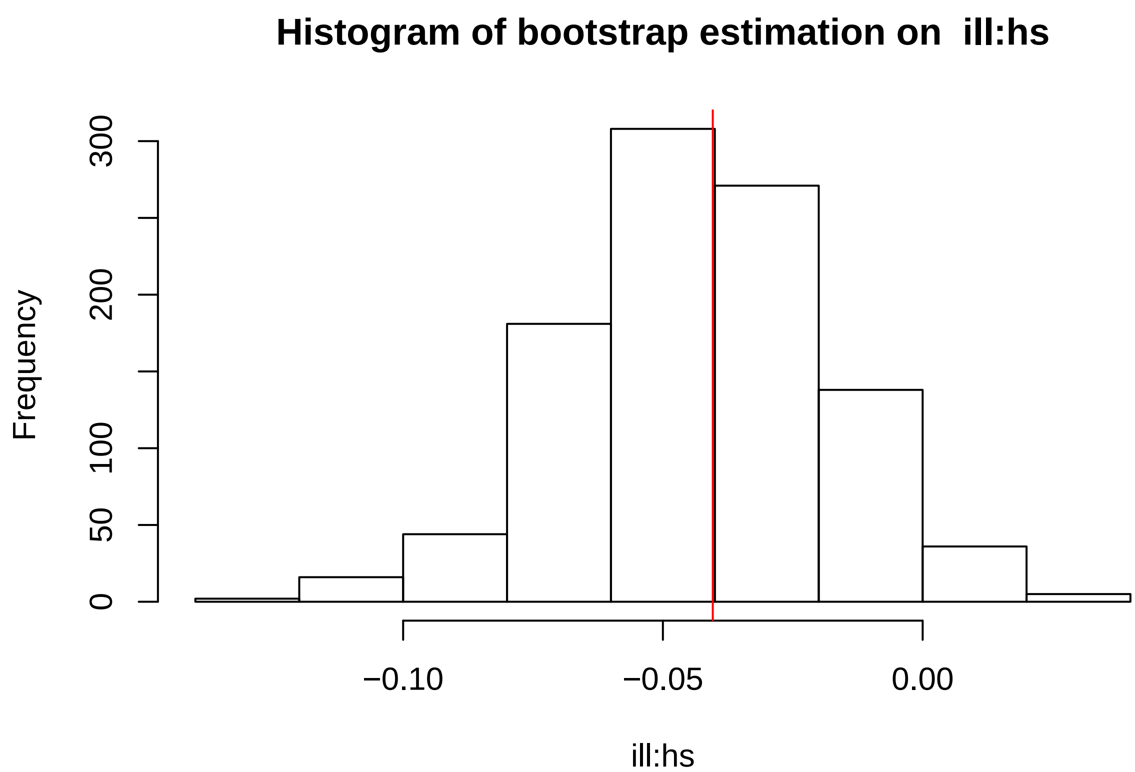


Histogram of bootstrap estimation on ad:insurance2
Frequency
0
50
100
150
200
−3.0
−2.5
−2.0
−1.5
−1.0
ad:insurance2


Histogram of bootstrap estimation on ad:insurance3
Frequency
0
50
100
150
−2.5
−2.0
−1.5
−1.0
−0.5
0.0
ad:insurance3


Histogram of bootstrap estimation on ad:insurance4
Frequency
0
50
150
250
350
−1
0
1
2
3
ad:insurance4


Histogram of bootstrap estimation on age:hs
Frequency
0
50
100
150
0.0
0.5
1.0
age:hs



fit3.2 is our final model

```r
#estimate the first person's probability that his/her number of pharmacist consultations equals 0,1,2,
# find out the mu for his/her poisson distribution
firstmu <- predict(fit3.2,type="response")[1]
data.frame(NumOfCon=0:8,Prob=round(dpois(0:8,firstmu),10))
```

```
##   NumOfCon         Prob
## 1        0 0.8735906156
## 2        1 0.1180600202
## 3        2 0.0079775172
## 4        3 0.0003593696
## 5        4 0.0000121416
## 6        5 0.0000003282
## 7        6 0.0000000074
## 8        7 0.0000000001
## 9        8 0.0000000000
```