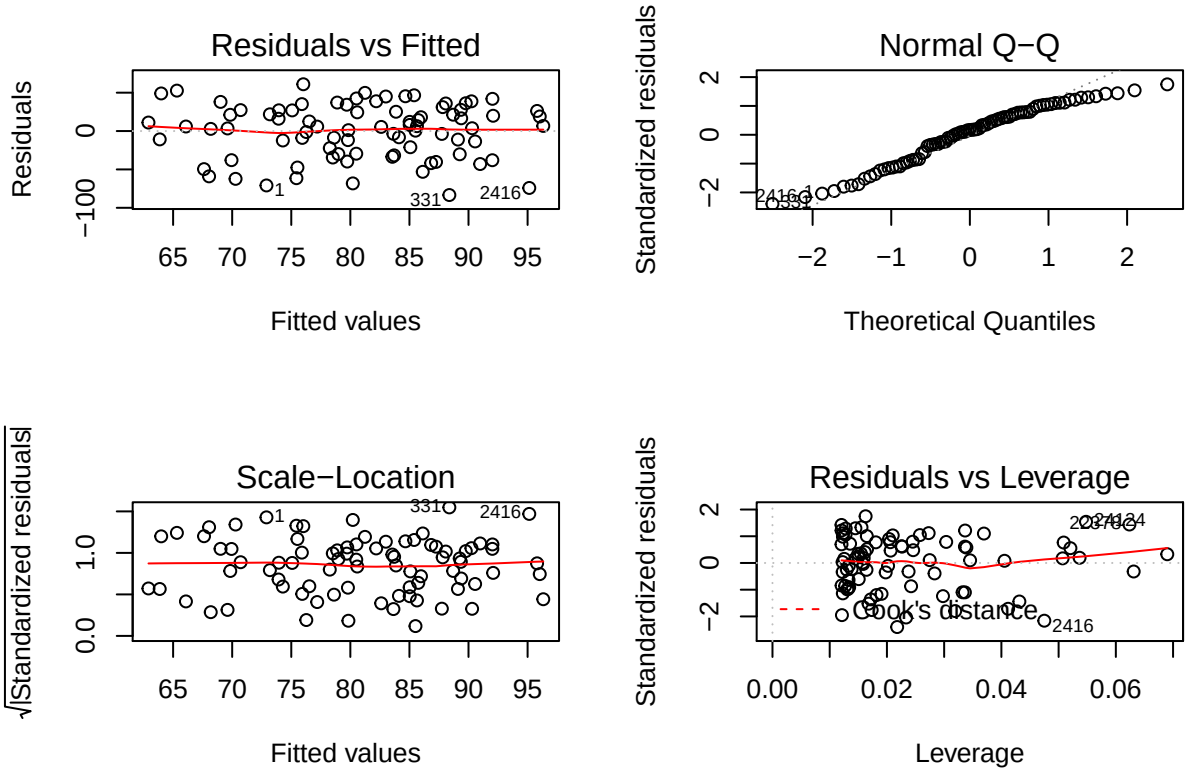
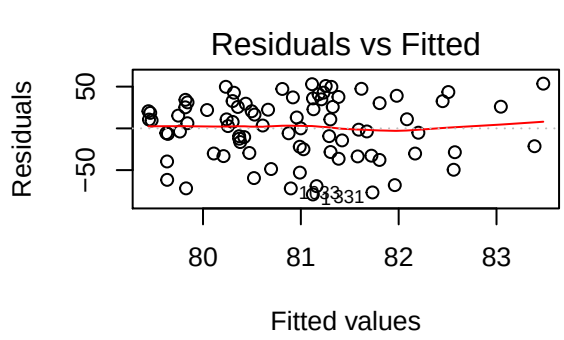
R Notebook

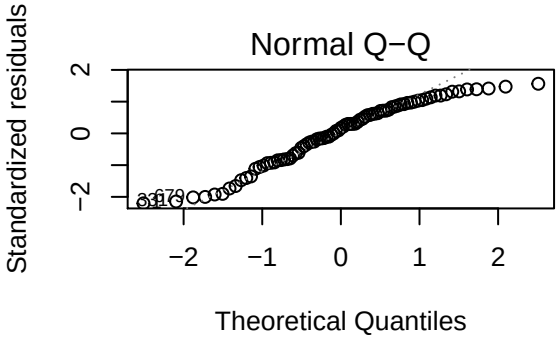
This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

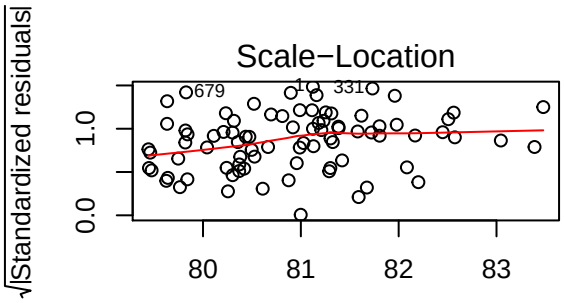
Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*.

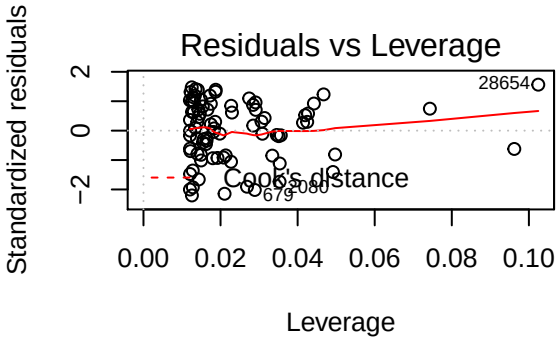
```
oj <- read.table("oj.csv",header=TRUE,sep=",")
library(lattice)
par(mfrow = c(2,2))
df <- unique(oj[c(1,7:17)])
var_list<- names(df)[2:12]
models <- lapply(var_list,
          function(x)
            {
             mod = lm(substitute(store ~ i, list(i = as.name(x))), data = df)
            })
par(mfrow = c(2,2))
invisible(lapply(models,plot))
```

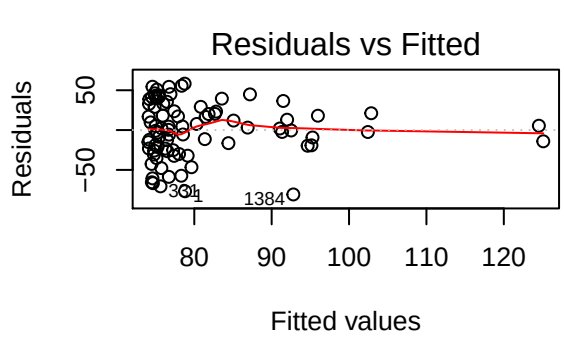


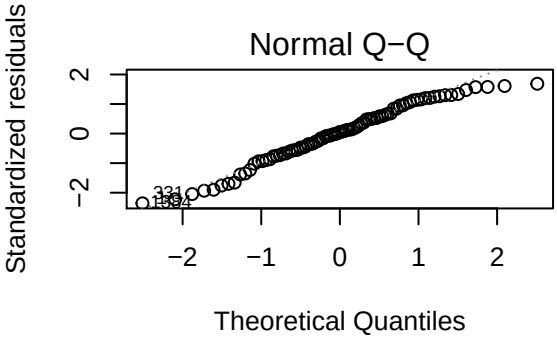

Residuals vs Fitted
Residuals
Fitted values
80
81
82
83
50
−50
679
1
331

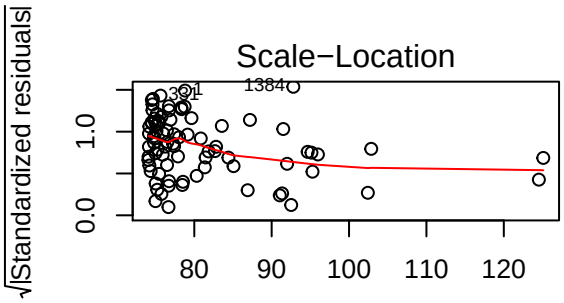

Normal Q−Q
Standardized residuals
Theoretical Quantiles
−2
−1
0
1
2
679

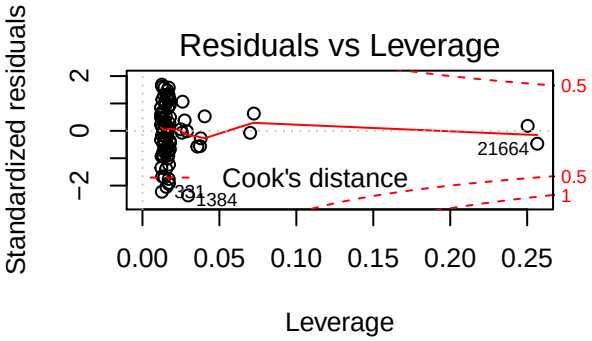

Scale−Location
√|Standardized residuals|
Fitted values
80
81
82
83
0.0
1.0
679
1
331

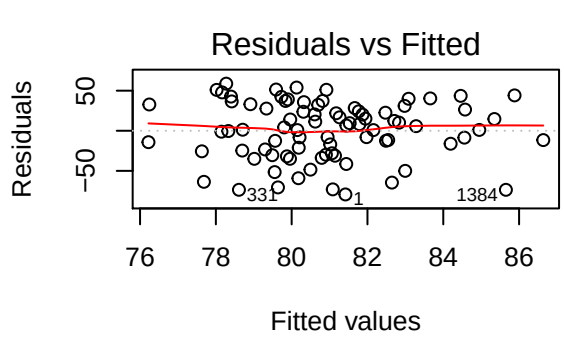

Residuals vs Leverage
Standardized residuals
Leverage
Cook's distance
0.00
0.02
0.04
0.06
0.08
0.10
−2
0
2
28654
679

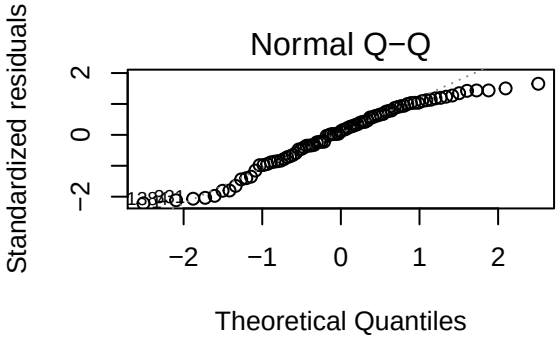

Residuals vs Fitted
Residuals
Fitted values
1384

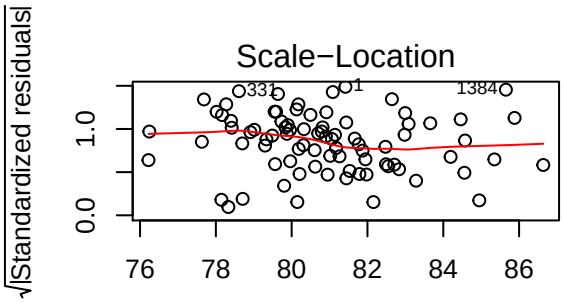

Normal Q−Q
Standardized residuals
Theoretical Quantiles

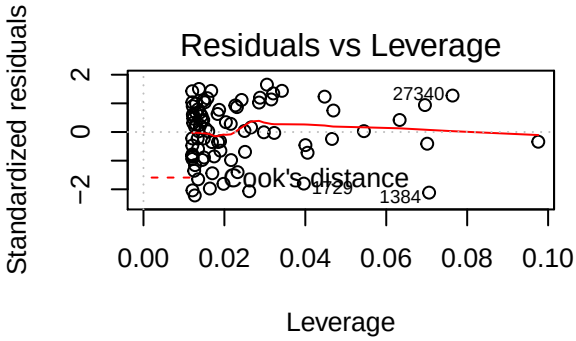

Scale−Location
√|Standardized residuals|
Fitted values
1384

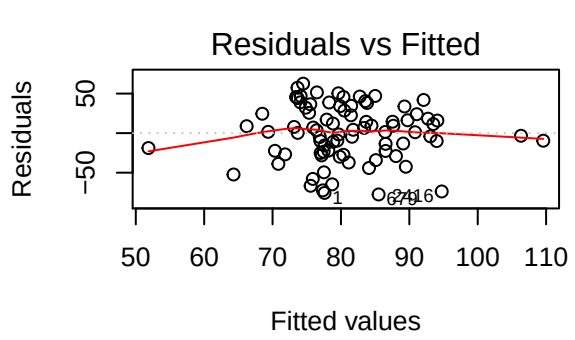

Residuals vs Leverage
Standardized residuals
Cook's distance
Leverage
21664
1384

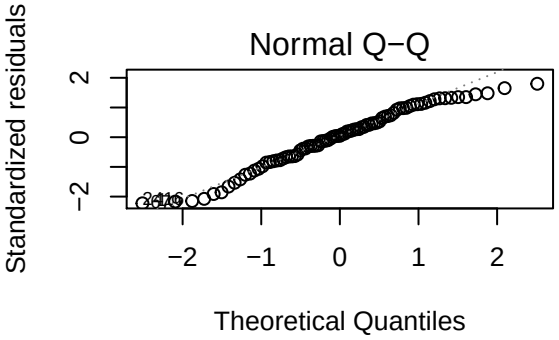

Residuals vs Fitted
Residuals
Fitted values
331
1
1384

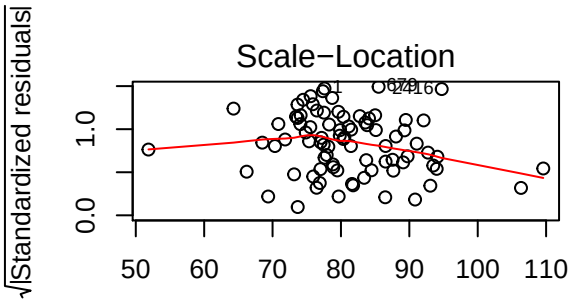

Normal Q−Q
Standardized residuals
Theoretical Quantiles

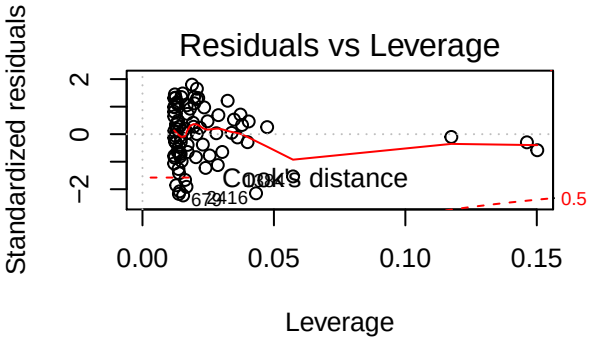

Scale−Location
√|Standardized residuals|
Fitted values
331
1
1384

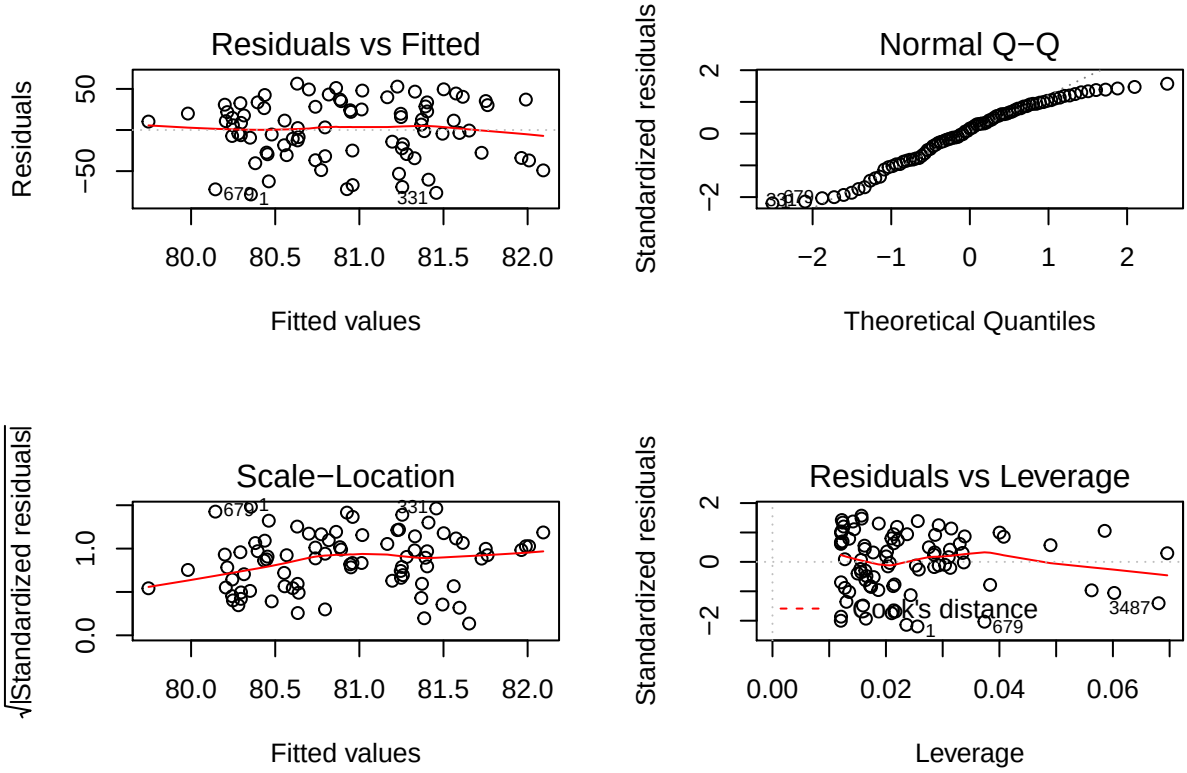

Residuals vs Leverage
Standardized residuals
Leverage
Cook's distance
27340
1729
1384

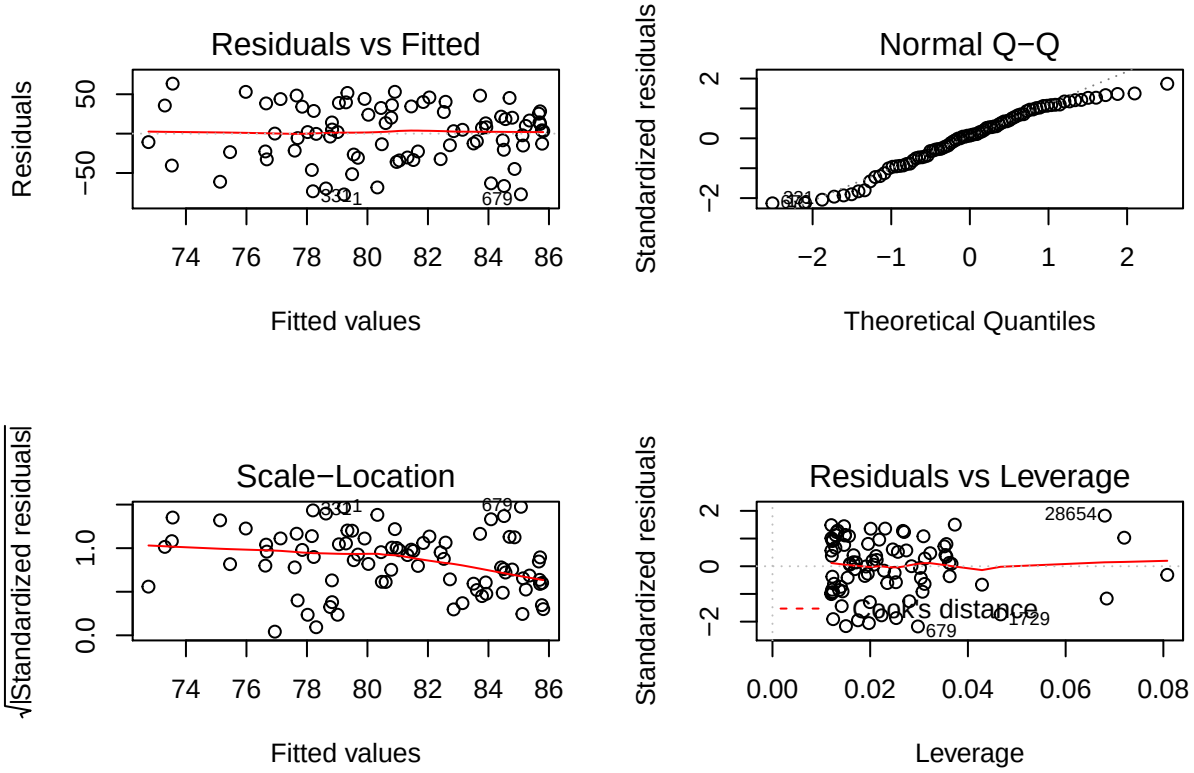

Residuals vs Fitted
Residuals
Fitted values
50
60
70
80
90
100
110
−50
50

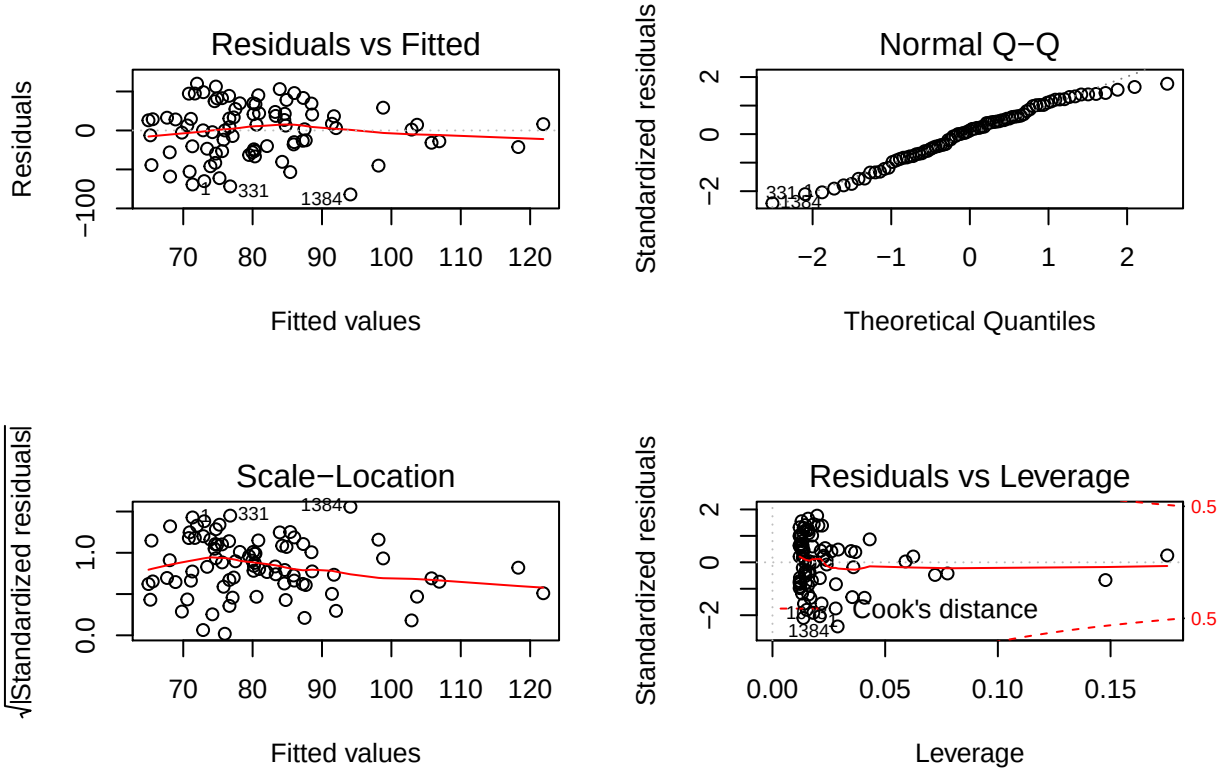

Normal Q−Q
Standardized residuals
Theoretical Quantiles
−2
−1
0
1
2

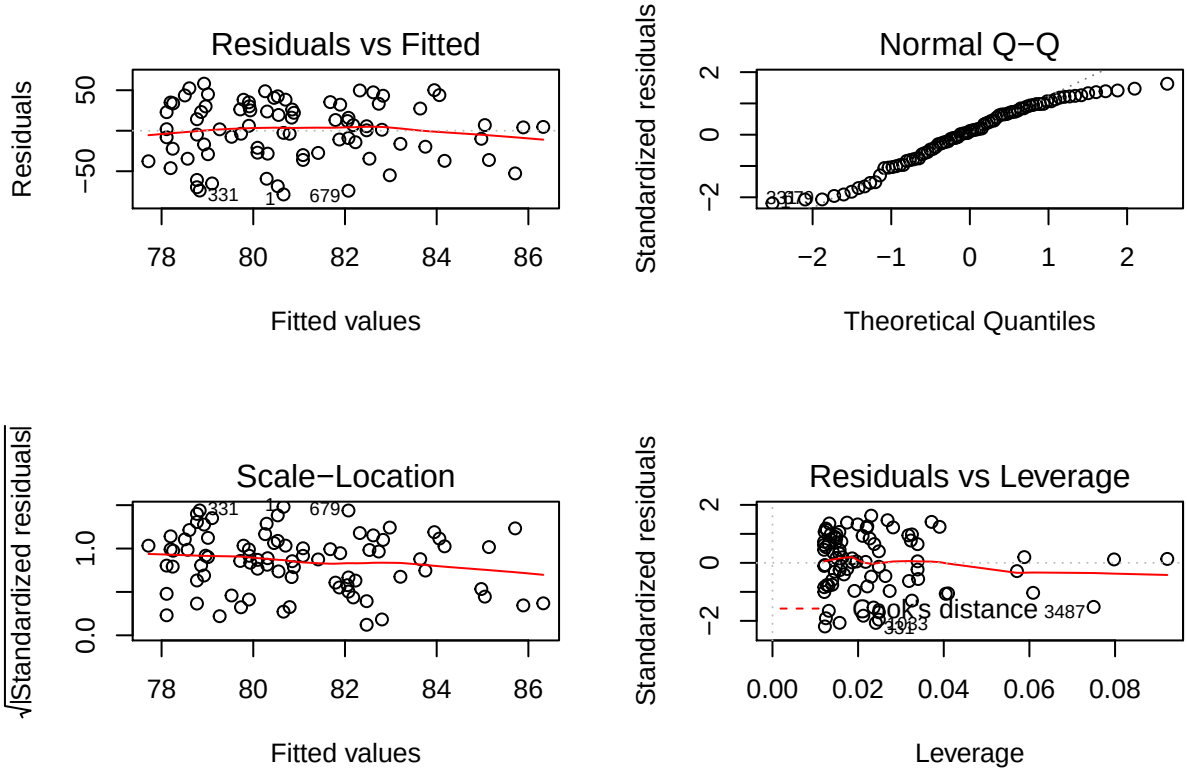

Scale−Location
√|Standardized residuals|
Fitted values
0.0
1.0
50
60
70
80
90
100
110

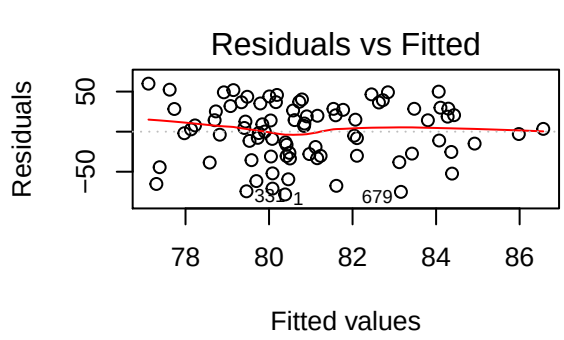

Residuals vs Leverage
Standardized residuals
Leverage
Cook's distance
0.5
−2
0
2
0.00
0.05
0.10
0.15

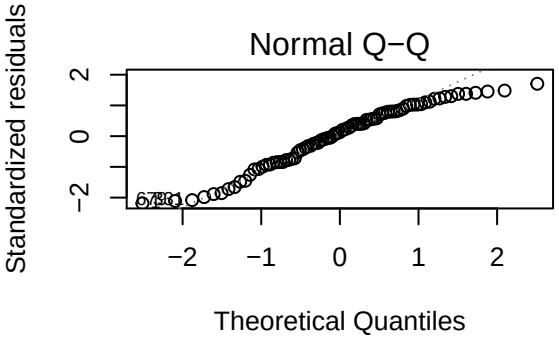

Residuals vs Fitted
Residuals
Fitted values
Normal Q−Q
Standardized residuals
Theoretical Quantiles
Scale−Location
√|Standardized residuals|
Fitted values
Residuals vs Leverage
Standardized residuals
Cook's distance
Leverage

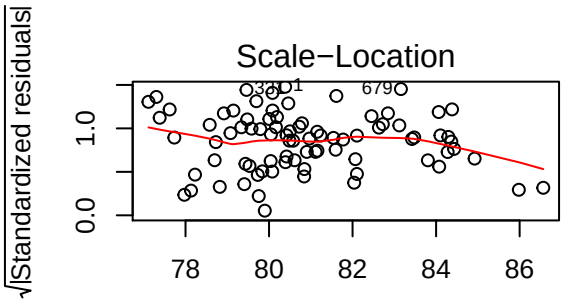

Residuals vs Fitted
Residuals
Fitted values
Normal Q−Q
Standardized residuals
Theoretical Quantiles
Scale−Location
√|Standardized residuals|
Fitted values
Residuals vs Leverage
Standardized residuals
Cook's distance
Leverage

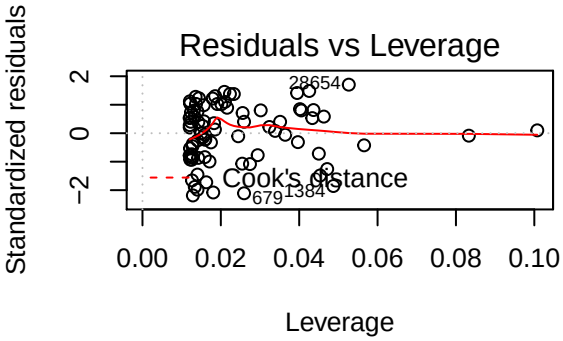

Residuals vs Fitted
Residuals
Fitted values
Normal Q−Q
Standardized residuals
Theoretical Quantiles
Scale−Location
√|Standardized residuals|
Fitted values
Residuals vs Leverage
Standardized residuals
Cook's distance
Leverage

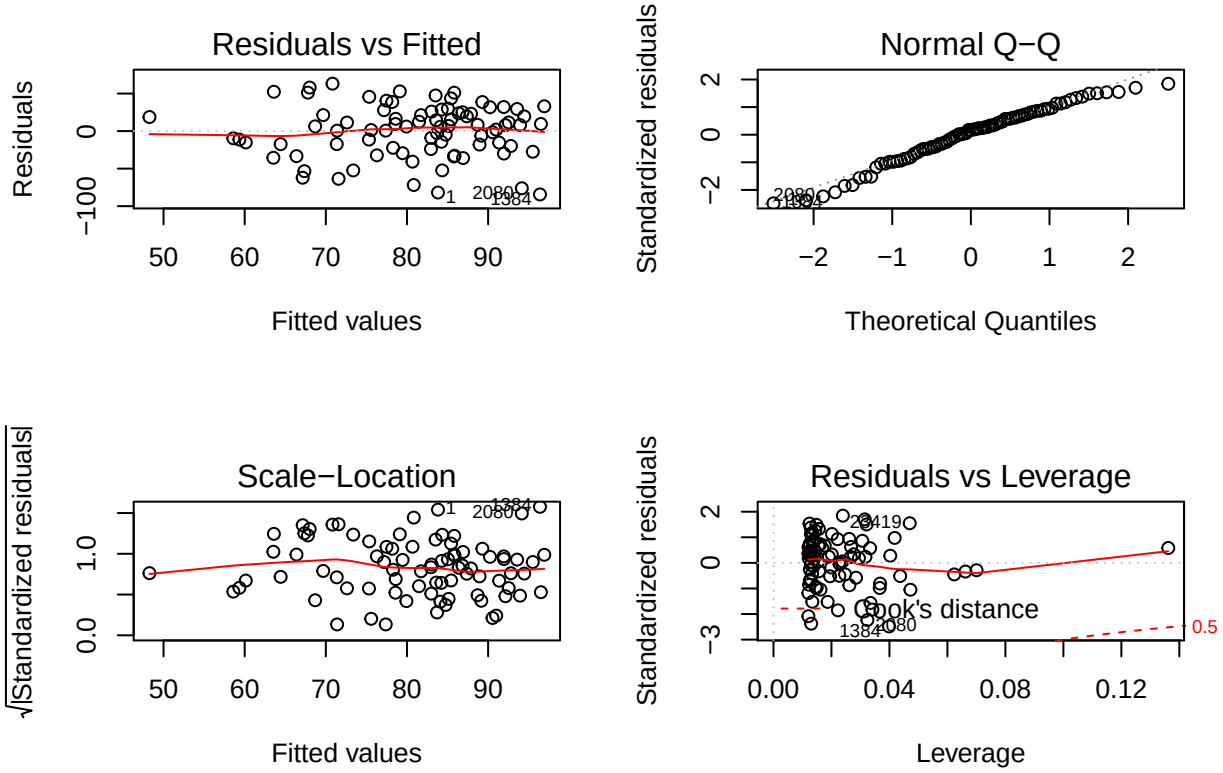

Residuals vs Fitted
Residuals
Fitted values
331
1
679
Normal Q−Q
Standardized residuals
Theoretical Quantiles
Scale−Location
√|Standardized residuals|
Fitted values
Residuals vs Leverage
Standardized residuals
Cook's distance
Leverage
3487


Residuals vs Fitted
Residuals
Fitted values
78
80
82
84
86
−50
50
679


Normal Q−Q
Standardized residuals
Theoretical Quantiles
−2
−1
0
1
2


Scale−Location
√|Standardized residuals|
Fitted values
78
80
82
84
86
0.0
1.0
679


Residuals vs Leverage
Standardized residuals
Leverage
Cook's distance
28654
679
1384
0.00
0.02
0.04
0.06
0.08
0.10




```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.3.2
```

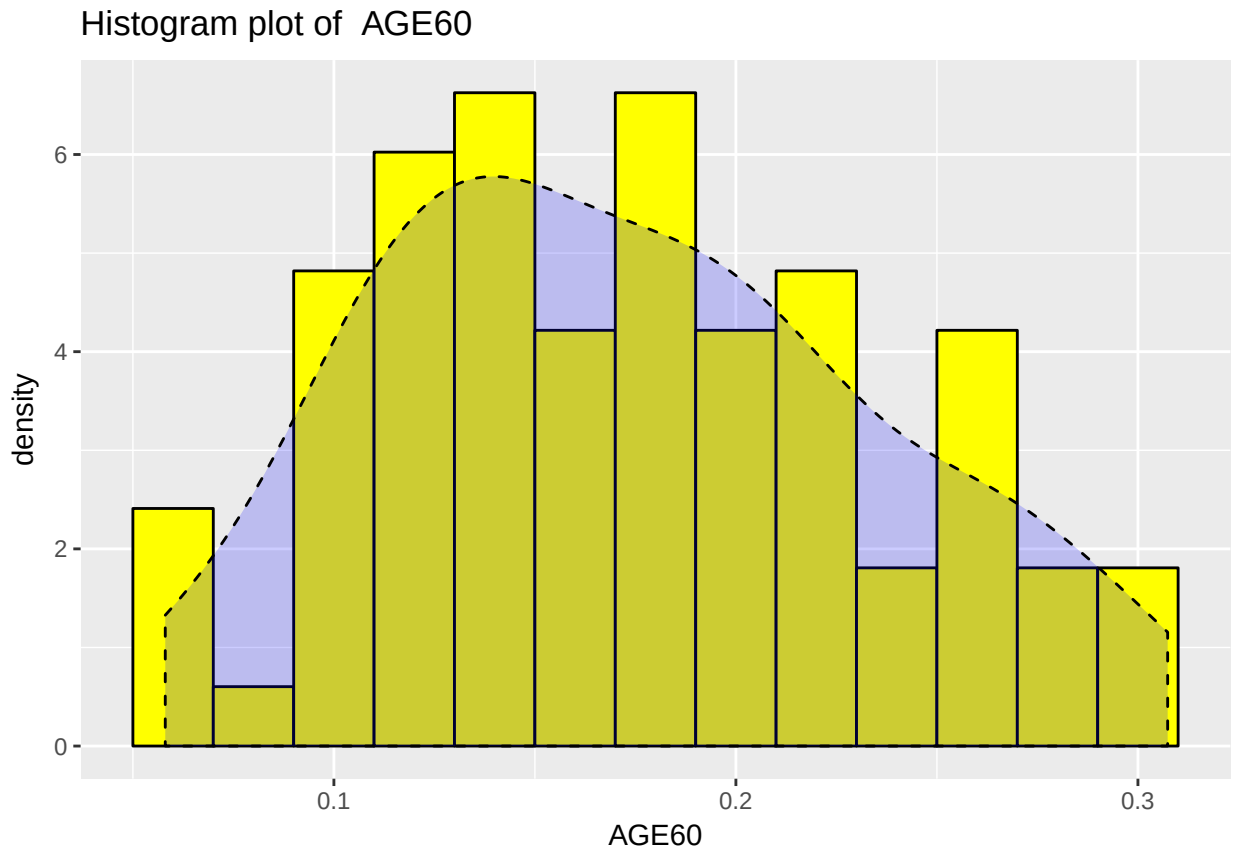
```
df <- unique(oj[c(1,7:17)])
var_list<- names(df)[2:12]
plots <- function(i){
                     ggplot(df, aes(x = df[i])) +
                     geom_histogram(aes(y = ..density..),colour = "black",
                                    fill = "yellow",binwidth = 0.02) +
                     geom_density(alpha = 0.2, fill = "blue",linetype = "dashed") +
                     geom_vline(aes(xintercept=mean(df[i], na.rm = TRUE)),
                                color = "black",linetype = "dashed", size = 1) +
                     xlab(names(df)[i]) +
                     ggtitle(paste("Histogram plot of ",names(df)[i]))}
par(mfrow = c(2,2))
lapply(2:12,FUN = plots)
```

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

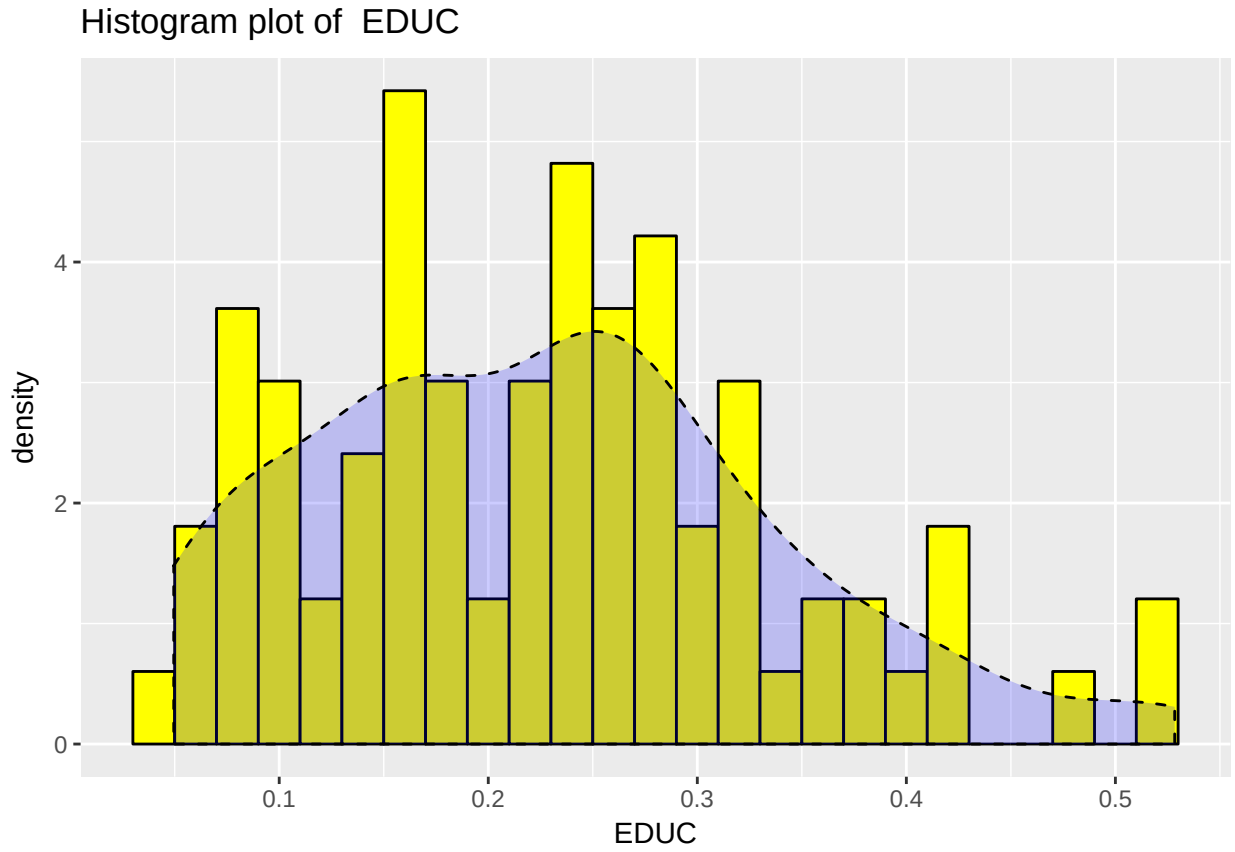
```
## Don't know how to automatically pick scale for object of type data.frame. Defaulting to continuous.
```

```
## Warning in mean.default(df[i], na.rm = TRUE): argument is not numeric or
## logical: returning NA
```

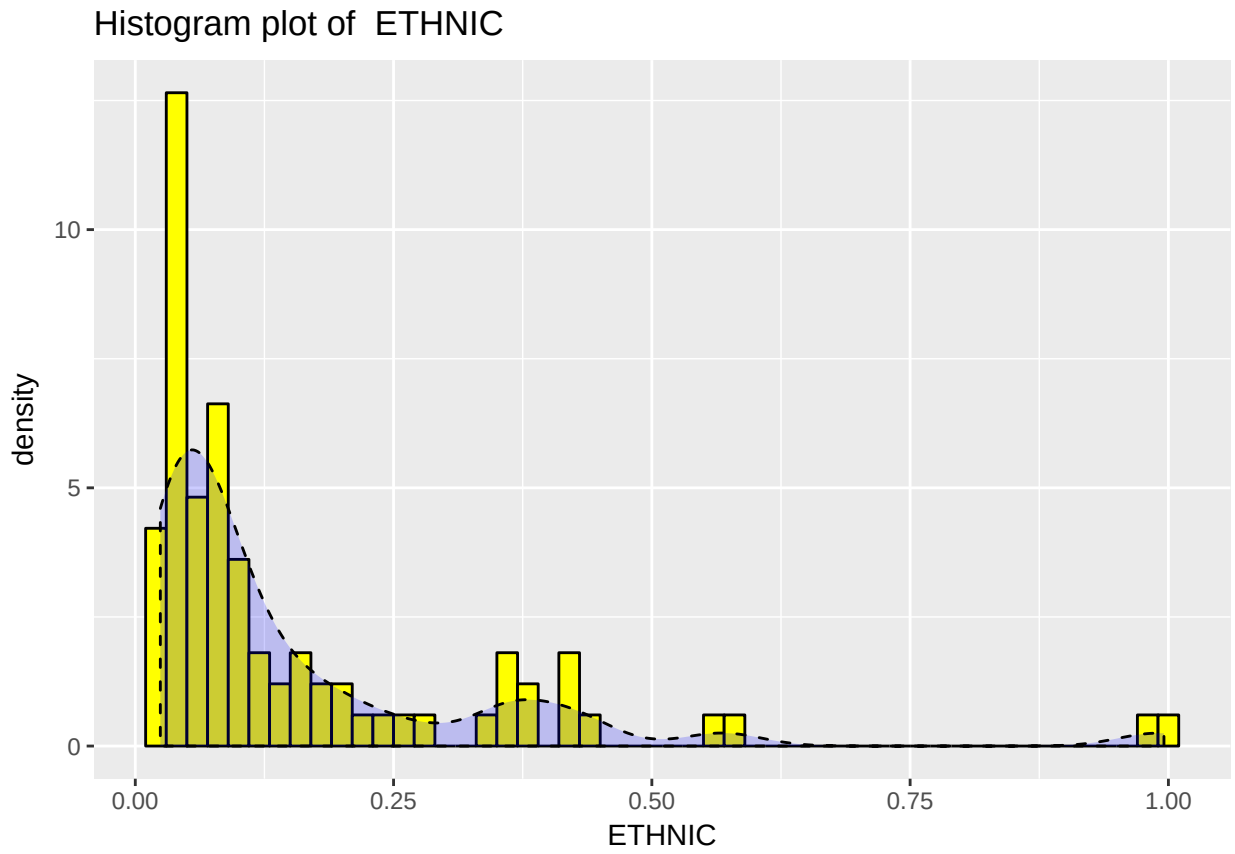
```
## Warning: Removed 83 rows containing missing values (geom_vline).
```



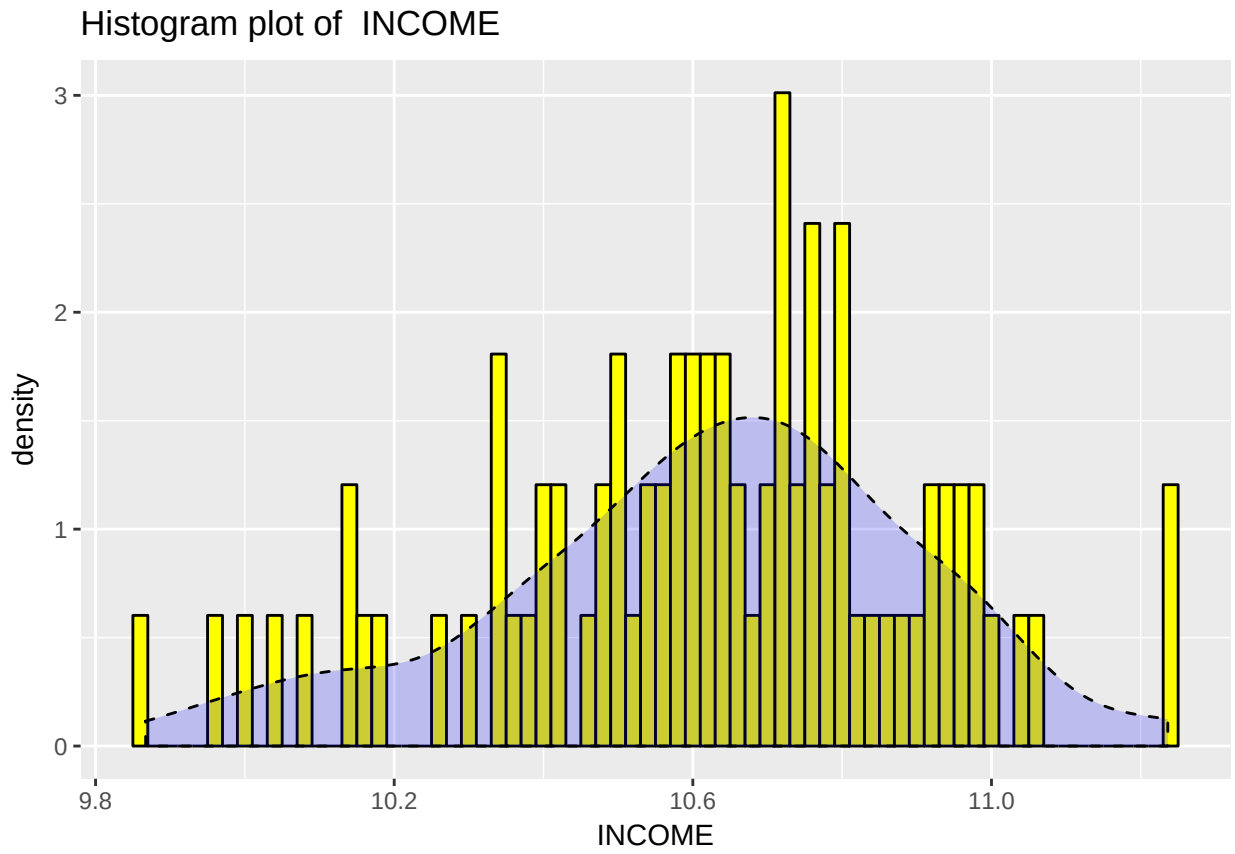

```
##
## [[2]]

## Don't know how to automatically pick scale for object of type data.frame. Defaulting to continuous.

## Warning in mean.default(df[i], na.rm = TRUE): argument is not numeric or
## logical: returning NA

## Warning in mean.default(df[i], na.rm = TRUE): Removed 83 rows containing
## missing values (geom_vline).
```



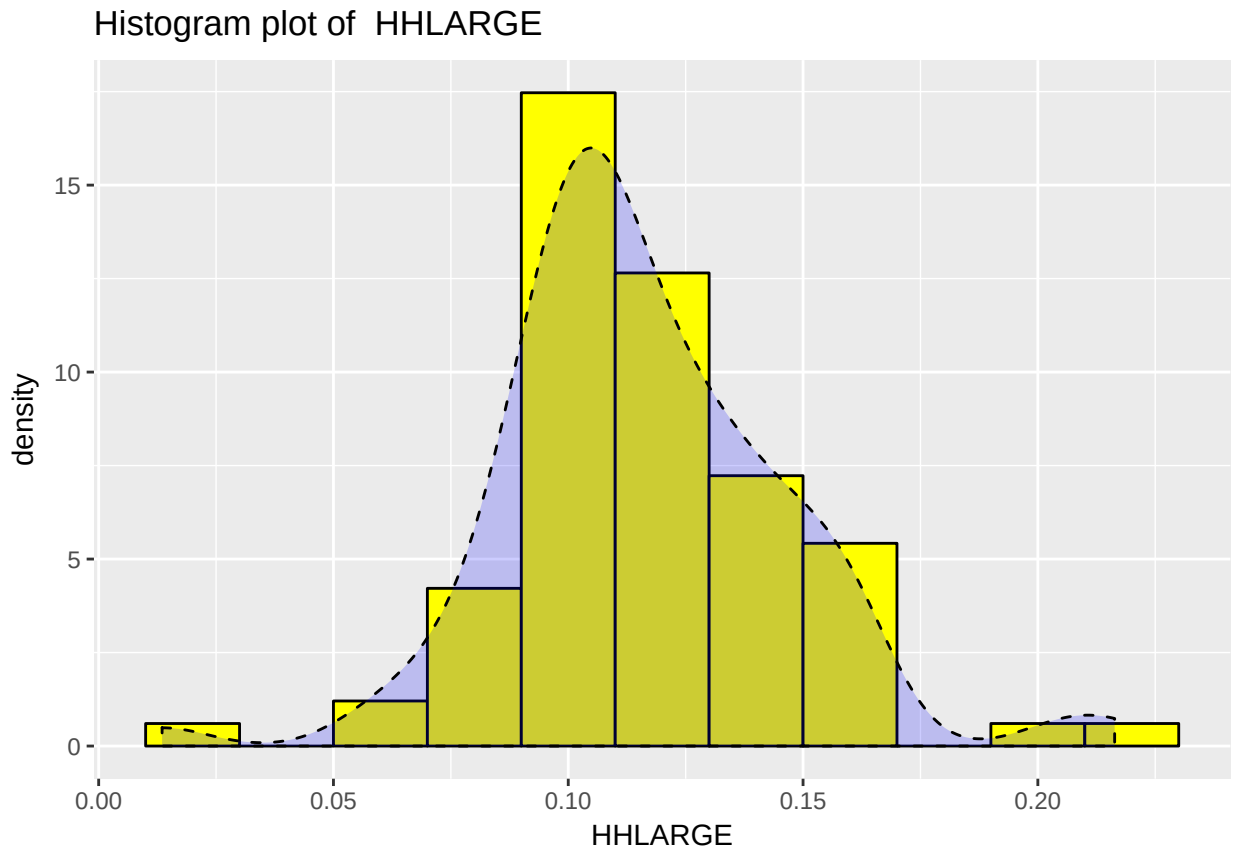

```
##
## [[3]]

## Don't know how to automatically pick scale for object of type data.frame. Defaulting to continuous.

## Warning in mean.default(df[i], na.rm = TRUE): argument is not numeric or
## logical: returning NA

## Warning in mean.default(df[i], na.rm = TRUE): Removed 83 rows containing
## missing values (geom_vline).
```



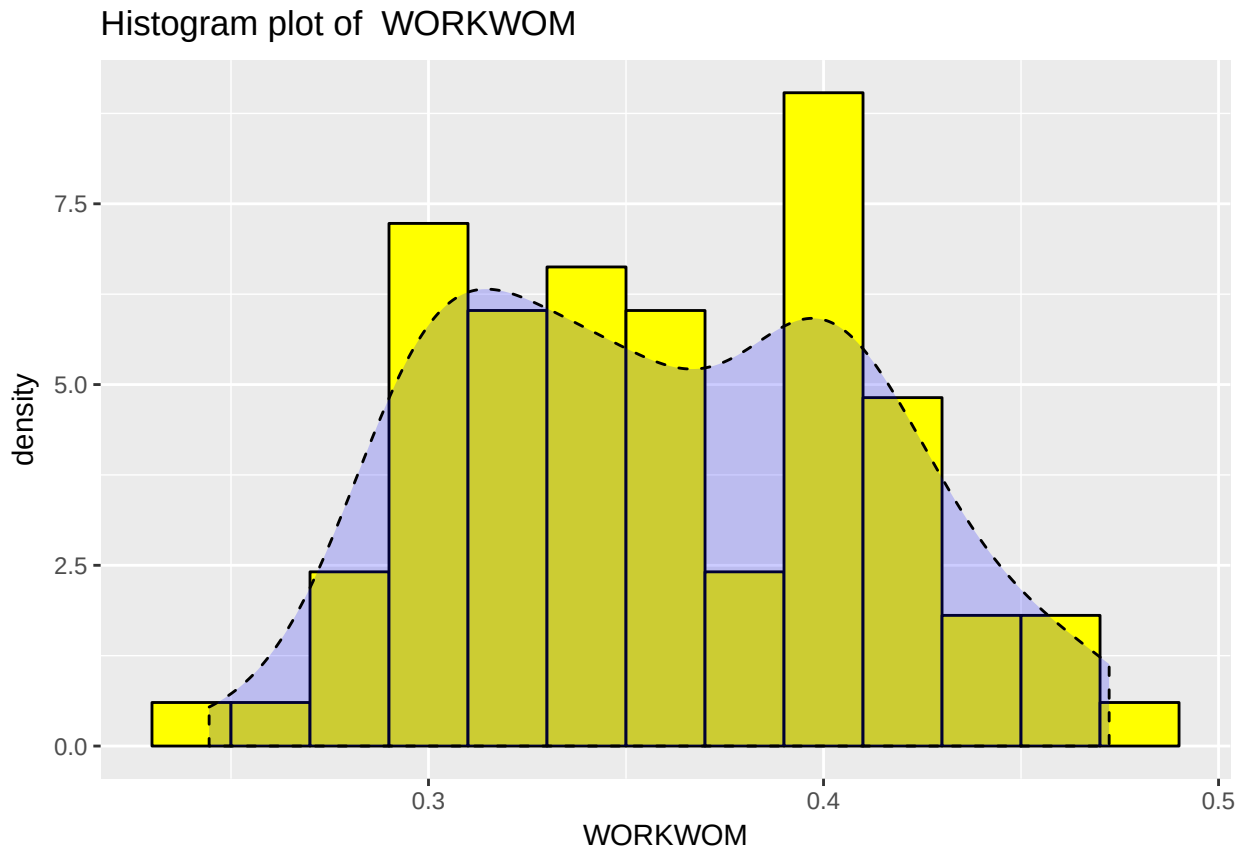

```
##
## [[4]]

## Don't know how to automatically pick scale for object of type data.frame. Defaulting to continuous.

## Warning in mean.default(df[i], na.rm = TRUE): argument is not numeric or
## logical: returning NA

## Warning in mean.default(df[i], na.rm = TRUE): Removed 83 rows containing
## missing values (geom_vline).
```



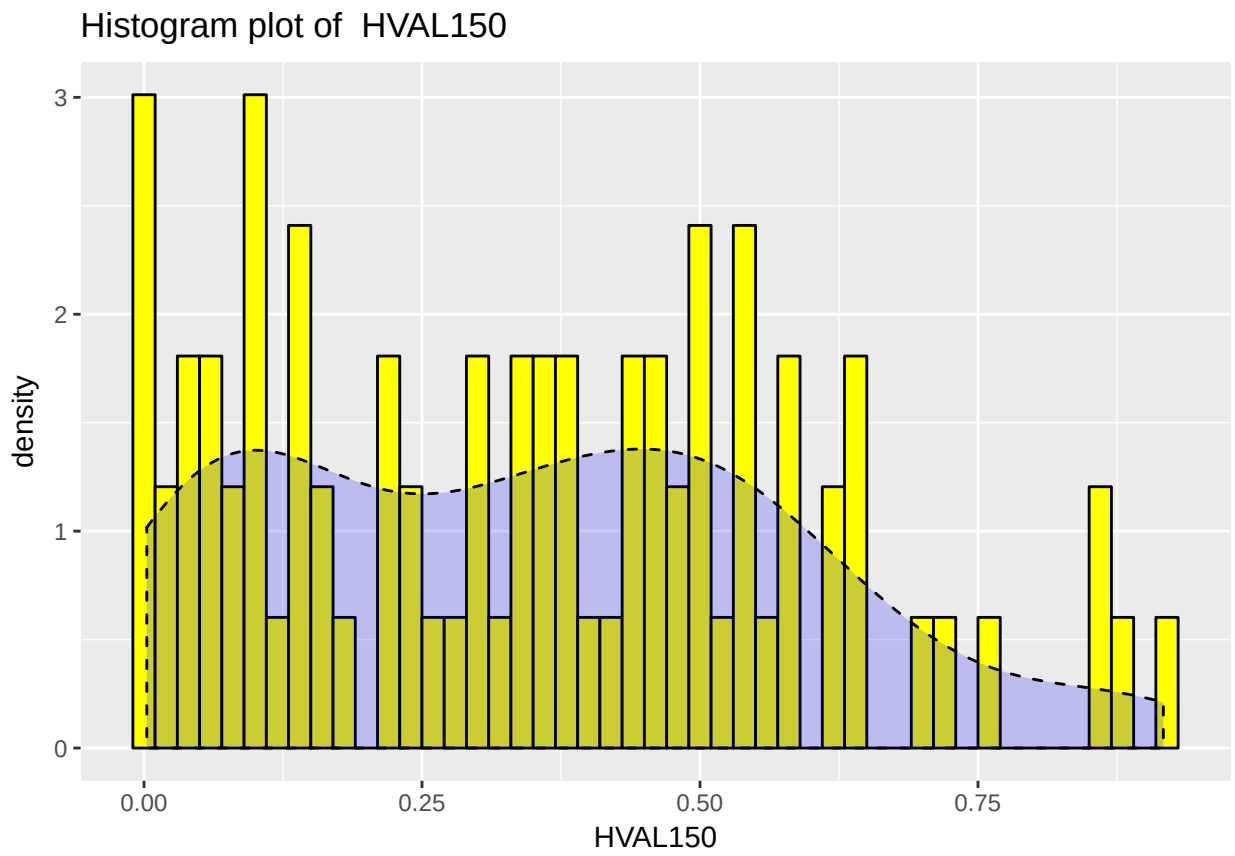

```
##
## [[5]]

## Don't know how to automatically pick scale for object of type data.frame. Defaulting to continuous.

## Warning in mean.default(df[i], na.rm = TRUE): argument is not numeric or
## logical: returning NA

## Warning in mean.default(df[i], na.rm = TRUE): Removed 83 rows containing
## missing values (geom_vline).
```



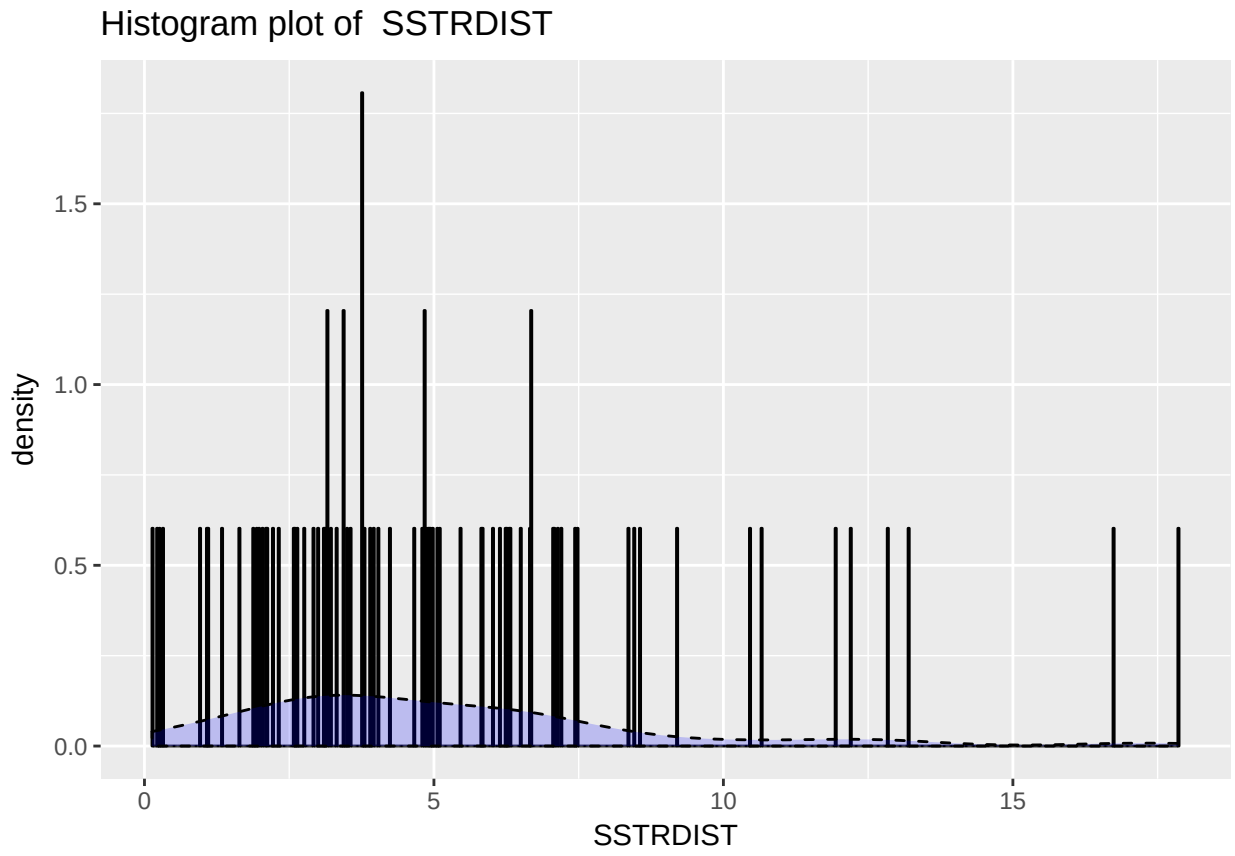

```
##
## [[6]]

## Don't know how to automatically pick scale for object of type data.frame. Defaulting to continuous.

## Warning in mean.default(df[i], na.rm = TRUE): argument is not numeric or
## logical: returning NA

## Warning in mean.default(df[i], na.rm = TRUE): Removed 83 rows containing
## missing values (geom_vline).
```



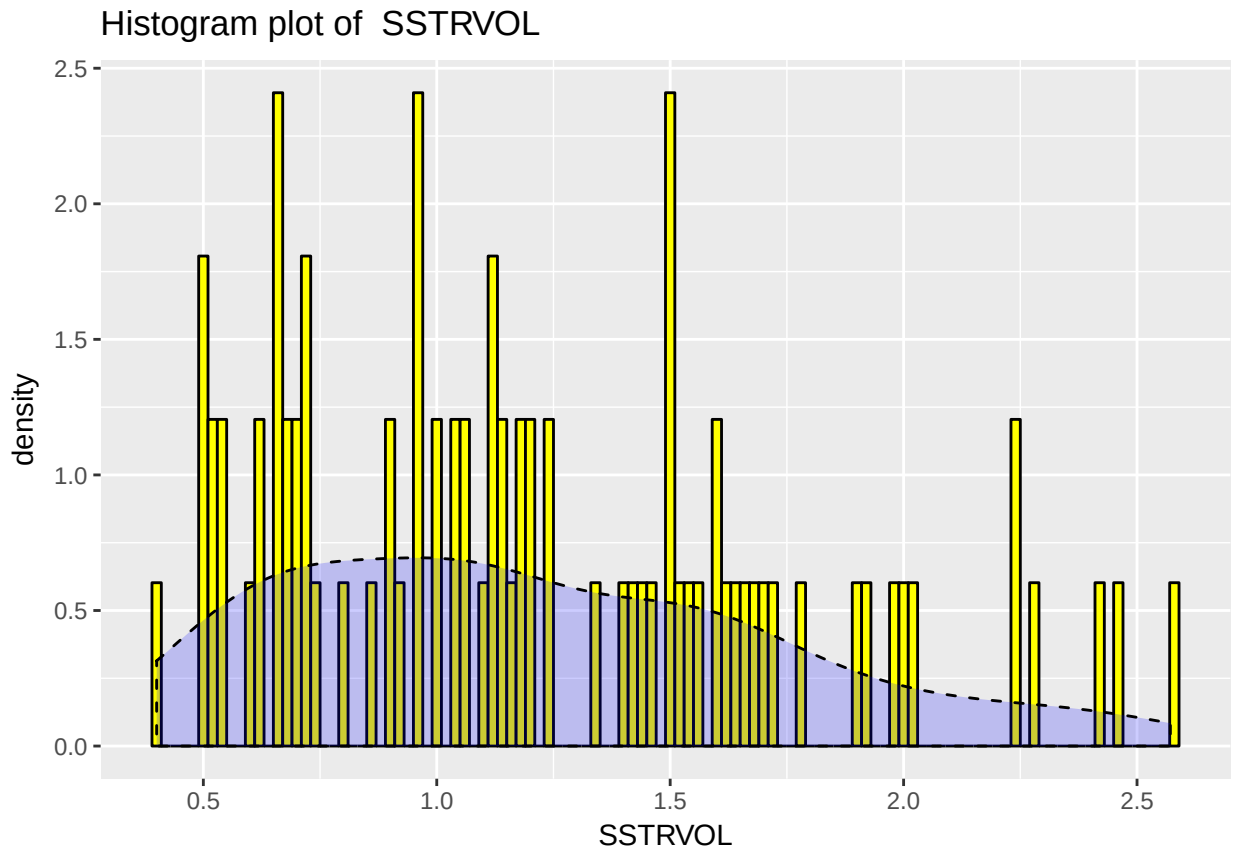

```
##
## [[7]]

## Don't know how to automatically pick scale for object of type data.frame. Defaulting to continuous.

## Warning in mean.default(df[i], na.rm = TRUE): argument is not numeric or
## logical: returning NA

## Warning in mean.default(df[i], na.rm = TRUE): Removed 83 rows containing
## missing values (geom_vline).
```



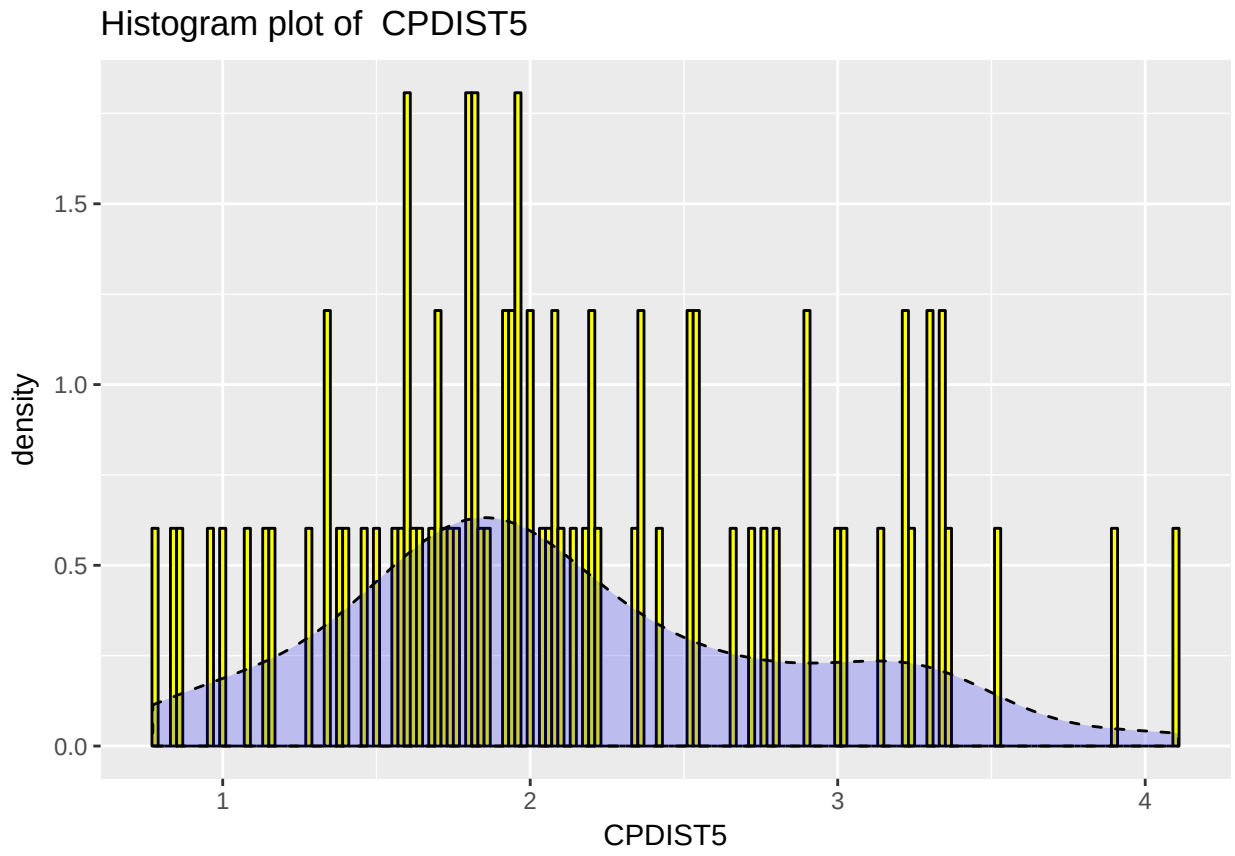

```
##
## [[8]]

## Don't know how to automatically pick scale for object of type data.frame. Defaulting to continuous.

## Warning in mean.default(df[i], na.rm = TRUE): argument is not numeric or
## logical: returning NA

## Warning in mean.default(df[i], na.rm = TRUE): Removed 83 rows containing
## missing values (geom_vline).
```



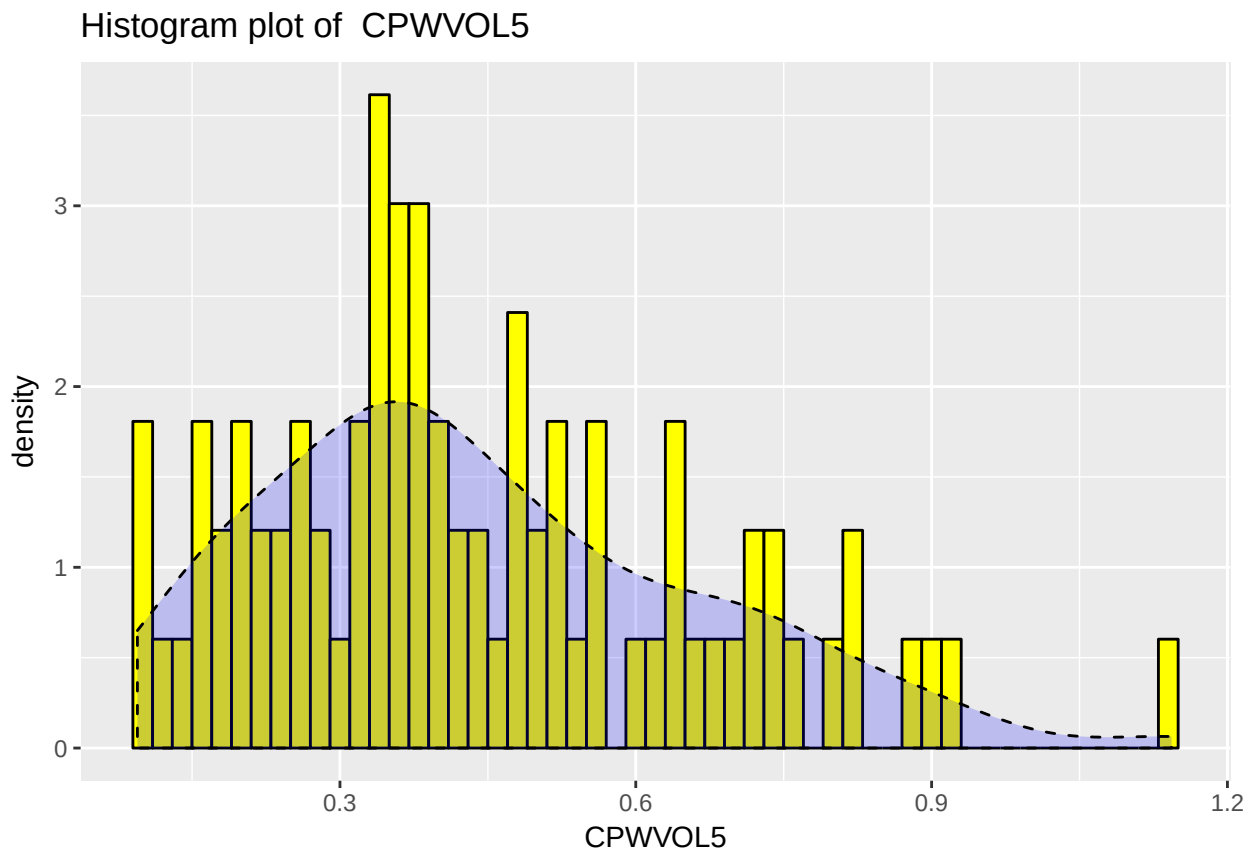

```
##
## [[9]]

## Don't know how to automatically pick scale for object of type data.frame. Defaulting to continuous.

## Warning in mean.default(df[i], na.rm = TRUE): argument is not numeric or
## logical: returning NA

## Warning in mean.default(df[i], na.rm = TRUE): Removed 83 rows containing
## missing values (geom_vline).
```




```
##
## [[10]]

## Don't know how to automatically pick scale for object of type data.frame. Defaulting to continuous.

## Warning in mean.default(df[i], na.rm = TRUE): argument is not numeric or
## logical: returning NA

## Warning in mean.default(df[i], na.rm = TRUE): Removed 83 rows containing
## missing values (geom_vline).
```



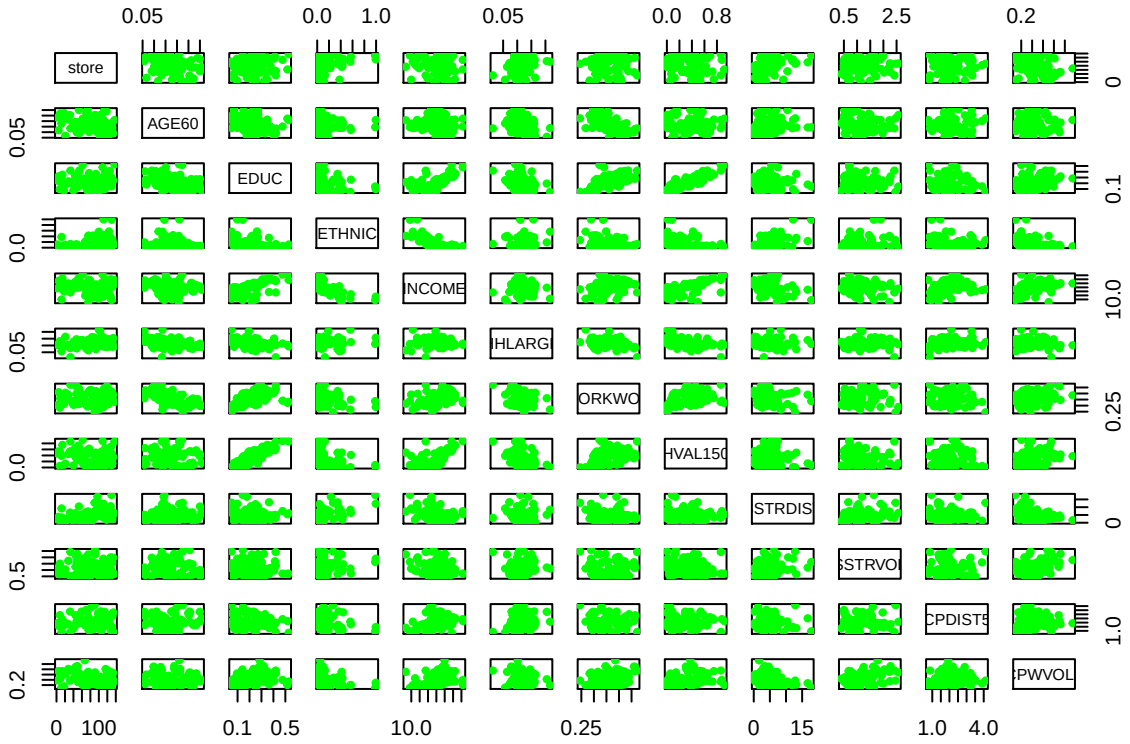

```
##
## [[11]]

## Don't know how to automatically pick scale for object of type data.frame. Defaulting to continuous.

## Warning in mean.default(df[i], na.rm = TRUE): argument is not numeric or
## logical: returning NA

## Warning in mean.default(df[i], na.rm = TRUE): Removed 83 rows containing
## missing values (geom_vline).
```



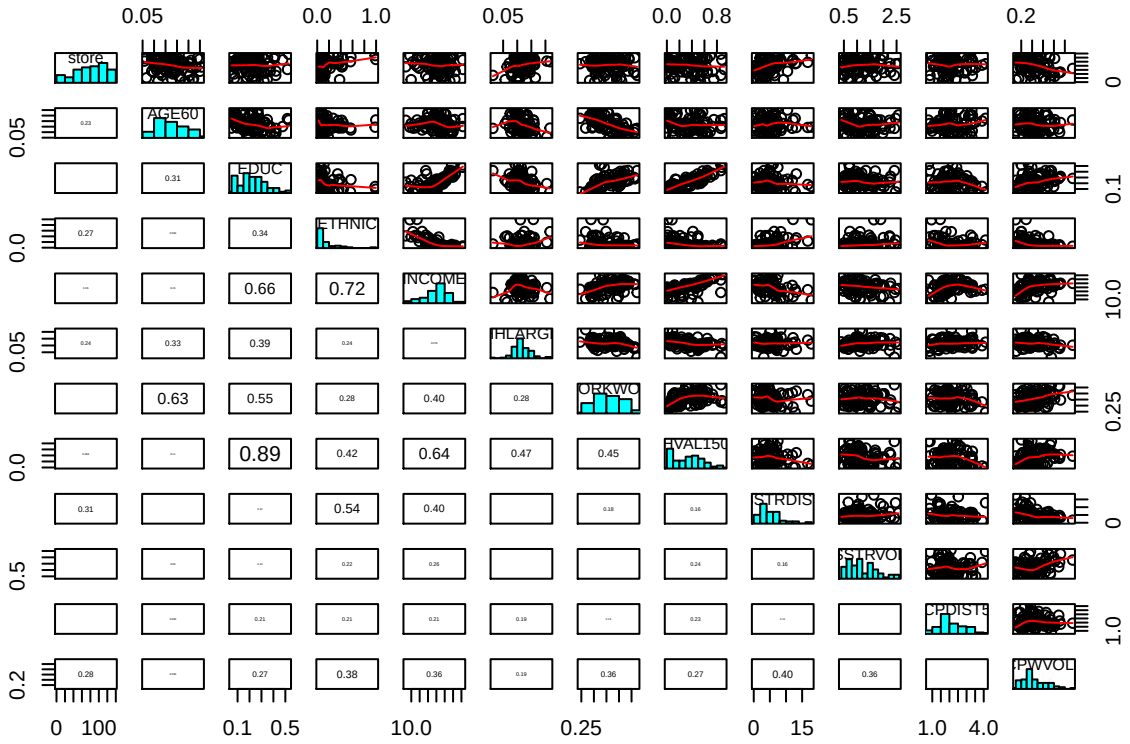

```
library(s20x)
```

```
## Warning: package 's20x' was built under R version 3.3.2
```

```
pairs(df,col="green",pch = 20)
```


store
AGE60
EDUC
ETHNIC
NCOME
HLARG
ORKWO
HVAL150
STRDIS
STRVOL
CPDIST5
PWVOL

```
pairs20x(df)
```




```
library(stargazer)
```

```
## Warning: package 'stargazer' was built under R version 3.3.2
```

```
##
## Please cite as:
##  Hlavac, Marek (2015). stargazer: Well-Formatted Regression and Summary Statistics Tables.
##  R package version 5.2. http://CRAN.R-project.org/package=stargazer
```

```
library(doBy)
```

```
## Warning: package 'doBy' was built under R version 3.3.2
```

```
stargazer(df, type = "text", median = TRUE, digits = 2)
```

```
##
## ================================================
## Statistic N  Mean  St. Dev.  Min  Median  Max
## ------------------------------------------------
## store     83 80.93  35.93     2     86    137
## AGE60     83 0.17    0.06   0.06   0.17  0.31
## EDUC      83 0.23    0.11   0.05   0.23  0.53
## ETHNIC    83 0.15    0.19   0.02   0.07  1.00
## INCOME    83 10.62   0.28   9.87  10.64  11.24
## HHLARGE   83 0.12    0.03   0.01   0.11  0.22
## WORKWOM   83 0.36    0.05   0.24   0.36  0.47
## HVAL150   83 0.34    0.24   0.003  0.35  0.92
```

```
## SSTRDIST  83 5.10    3.49   0.13   4.65  17.86
## SSTRVOL   83 1.21    0.53   0.40   1.12  2.57
## CPDIST5   83 2.12    0.74   0.77   1.96  4.11
## CPWVOL5   83 0.44    0.22   0.09   0.38  1.14
## ------------------------------------------------
```

```
summaryBy(AGE60 ~ store, data = oj, FUN = c(mean), na.rm =TRUE)
```

```
##    store AGE60.mean
## 1      2 0.23286473
## 2      5 0.11736803
## 3      8 0.25239404
## 4      9 0.26911902
## 5     12 0.17834141
## 6     14 0.21394927
## 7     18 0.27231337
## 8     21 0.06689646
## 9     28 0.21330879
## 10    32 0.25495303
## 11    33 0.13416997
## 12    40 0.18185180
## 13    44 0.19098278
## 14    45 0.12885735
## 15    47 0.12579830
## 16    48 0.09792196
## 17    49 0.18747319
## 18    50 0.15335738
## 19    51 0.17615972
## 20    52 0.15224119
## 21    53 0.30027868
## 22    54 0.09022228
## 23    56 0.19288855
## 24    59 0.11081891
## 25    62 0.22253426
## 26    64 0.14199202
## 27    67 0.21027298
## 28    68 0.18141776
## 29    70 0.19023580
## 30    71 0.26807087
## 31    72 0.28372769
## 32    73 0.25745078
## 33    74 0.30739786
## 34    75 0.20769949
## 35    76 0.14919242
## 36    77 0.10110045
## 37    78 0.11194799
## 38    80 0.15269126
## 39    81 0.18111894
## 40    83 0.20083469
## 41    84 0.12210000
## 42    86 0.13875637
## 43    88 0.16041421
## 44    89 0.20581136
## 45    90 0.22521957
## 46    91 0.25573061
```

```
## 47    92 0.13782763
## 48    93 0.14239019
## 49    94 0.10300220
## 50    95 0.23071750
## 51    97 0.14243323
## 52    98 0.24920053
## 53   100 0.13699514
## 54   101 0.22503522
## 55   102 0.21662623
## 56   103 0.05805397
## 57   104 0.13528637
## 58   105 0.17554213
## 59   106 0.10988735
## 60   107 0.26186745
## 61   109 0.15105566
## 62   110 0.11495669
## 63   111 0.21051284
## 64   112 0.08972372
## 65   113 0.29935255
## 66   114 0.18217330
## 67   115 0.06028005
## 68   116 0.18817339
## 69   117 0.11010273
## 70   118 0.28944238
## 71   119 0.12157496
## 72   121 0.16358133
## 73   122 0.06195391
## 74   123 0.17604094
## 75   124 0.11962581
## 76   126 0.10700227
## 77   128 0.15748519
## 78   129 0.10341287
## 79   130 0.14511731
## 80   131 0.17065481
## 81   132 0.13961735
## 82   134 0.09015265
## 83   137 0.20960245
```

```
t.test(df$AGE60,df$EDUC)
```

```
##
##  Welch Two Sample t-test
##
## data:  df$AGE60 and df$EDUC
## t = -3.7769, df = 128.79, p-value = 0.0002415
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.08046517 -0.02514245
## sample estimates:
## mean of x mean of y
## 0.1729724 0.2257762
```

```
reg <- lm(df$store~df$AGE60+df$INCOME+df$EDUC+df$ETHNIC+df$HVAL150+df$HHLARGE+df$WORKWOM+df$SSTRDIST+df$
summary(reg)
```

```
##
## Call:
## lm(formula = df$store ~ df$AGE60 + df$INCOME + df$EDUC + df$ETHNIC +
##     df$HVAL150 + df$HHLARGE + df$WORKWOM + df$SSTRDIST + df$SSTRVOL +
##     df$CPDIST5 + df$CPWVOL5)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -73.087 -22.128  -0.087  22.716  64.968
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -208.779    310.956  -0.671   0.5041
## df$AGE60     -38.375    120.740  -0.318   0.7515
## df$INCOME     21.789     31.483   0.692   0.4911
## df$EDUC       77.623     95.245   0.815   0.4178
## df$ETHNIC     35.161     35.236   0.998   0.3217
## df$HVAL150   -25.203     39.299  -0.641   0.5234
## df$HHLARGE   161.804    217.625   0.743   0.4596
## df$WORKWOM    65.330    138.795   0.471   0.6393
## df$SSTRDIST    1.759      1.379   1.275   0.2064
## df$SSTRVOL     8.738      9.196   0.950   0.3452
## df$CPDIST5     4.716      5.900   0.799   0.4268
## df$CPWVOL5   -47.787     24.259  -1.970   0.0528 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 33.17 on 71 degrees of freedom
## Multiple R-squared:  0.2623, Adjusted R-squared:  0.148
## F-statistic: 2.295 on 11 and 71 DF,  p-value: 0.01816
```