Exam_Racinskis_pr20015

Pēteris Račinskis pr20015

6/9/2021

1. uzdevums

Piezīme: eksāmenā laika nav daudz, tāpēc visa tā pati copy & paste diagnostika normalitātei, dispersijas vienmērīgumam un autokorelācijai nav atkārtota pēc katra regresijas modeļa.

Bibliotēku un datu ielāde:

```
df <- read.delim("poverty.txt")
summary(df)
attach(df)
library(psych)
library(robustbase)
library(histogram)
library(nortest)
library(boot)
library(e1071)
```

Palīgfunkciju definīcija:

```
general_lreg <- function(vec1,vec2,degree=1,plot=F,print=F,names=c("","")) {
  fit<-lm(vec2~poly(vec1,degree,raw=T))
  if(plot){
    plot(vec1,vec2,xlab=names[1],ylab=names[2])
    x <- seq(min(vec1),max(vec1),length.out = length(vec1))
    f <- predict(fit, newdata = data.frame(vec1 = x))
    lines(x,f,col="red",lwd=2)
  }
  if(print){
    print(paste("R-squared:",summary(fit)$r.squared))
  }
  fit
}
cb_skewness <- function(x, i){
  skewness(x[i])
}
```

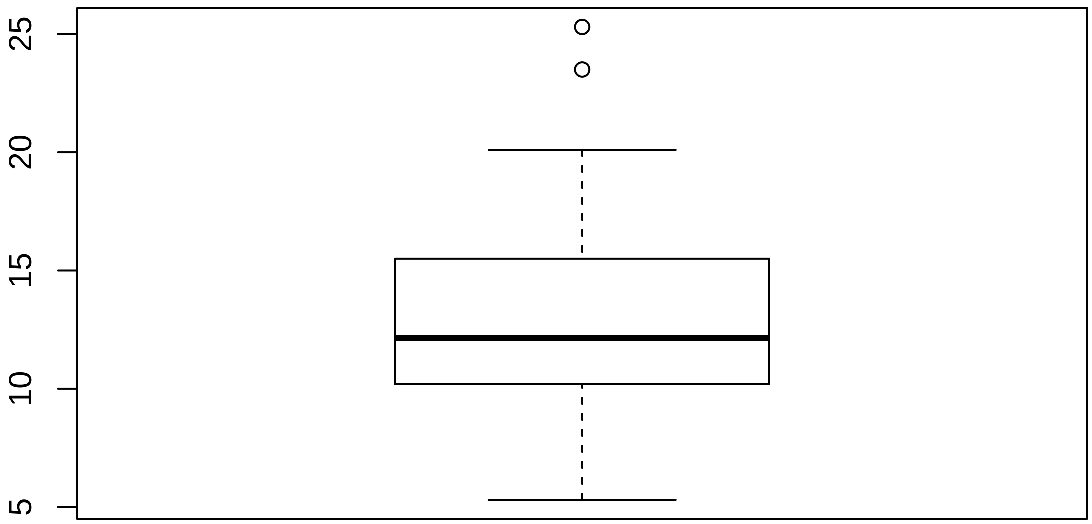
Dati: viena kolonna ar štatu nosaukumiem, 5 skaitlisku datu kolonnas.

1.1 Aprakstošās statistikas:

```
describe(PovPct)
```

```
##    vars  n  mean   sd median trimmed  mad min  max range skew kurtosis   se
## X1    1 50 12.94 4.13  12.15   12.54 3.78 5.3 25.3    20 0.89     0.58 0.58
```

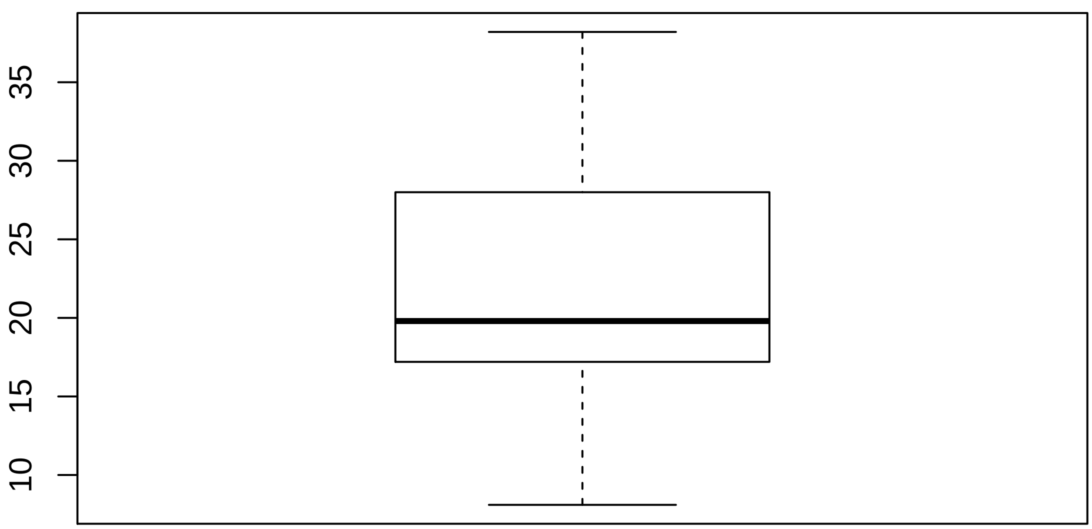
```r
boxplot(PovPct)
```




```r
describe(Brth15to17)
```

```
##    vars  n  mean   sd median trimmed  mad min  max range skew kurtosis   se
## X1    1 50 21.83 7.45   19.8   21.42 5.93 8.1 38.2  30.1 0.52    -0.56 1.05
```

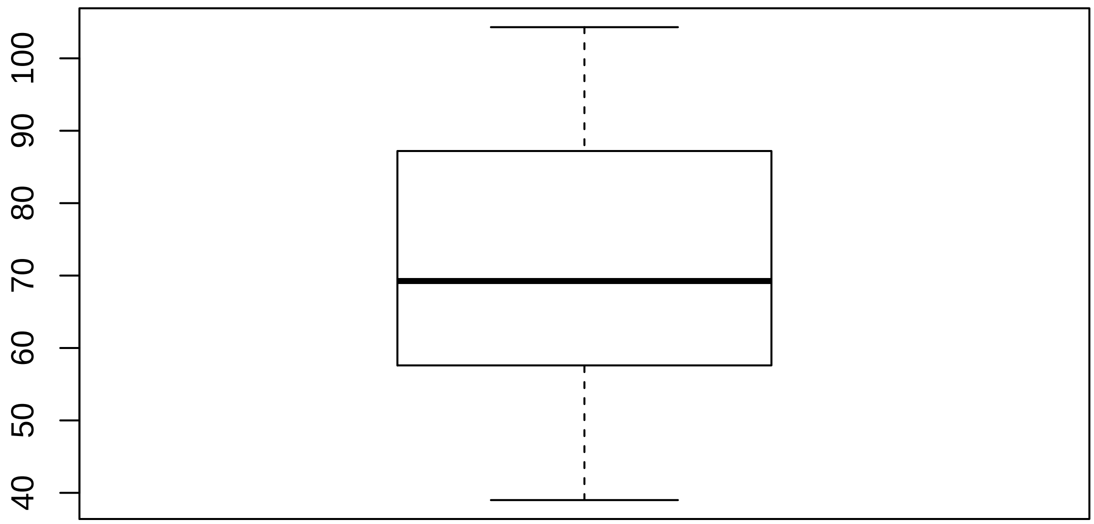
```r
boxplot(Brth15to17)
```




```r
describe(Brth18to19)
```

```
##    vars  n  mean    sd median trimmed   mad min   max range skew kurtosis   se
## X1    1 50 71.43 18.69  69.25   71.17 19.27  39 104.3  65.3 0.13    -1.07 2.64
```

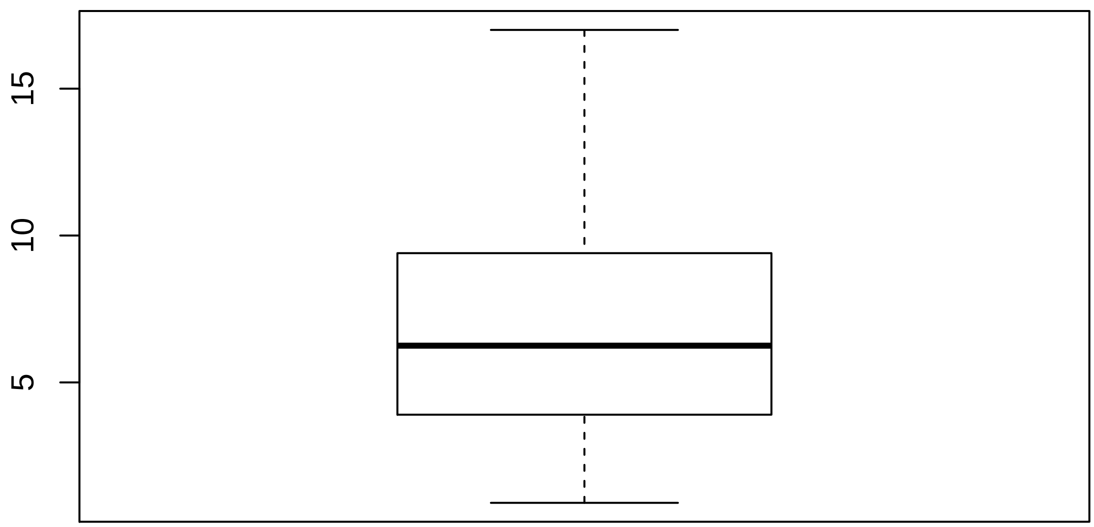
```r
boxplot(Brth18to19)
```




```
describe(ViolCrime)
```

```
##    vars  n mean   sd median trimmed  mad min max range skew kurtosis   se
## X1    1 50 6.71 3.62   6.25    6.55 4.08 0.9  17  16.1 0.42    -0.48 0.51
```

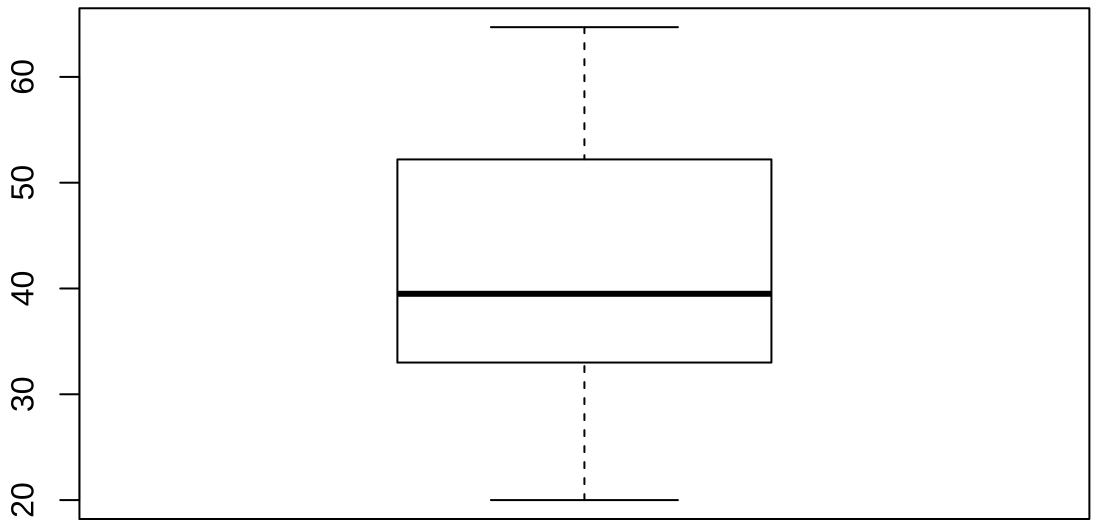
```
boxplot(ViolCrime)
```




```
describe(TeenBrth)
```

```
##    vars  n  mean    sd median trimmed  mad min  max range skew kurtosis   se
## X1    1 50 41.71 11.82   39.5   41.35 10.9  20 64.7  44.7 0.26    -0.92 1.67
```

```
boxplot(TeenBrth)
```



1.2 Robusti novērtētāji pret vidējo vērtību:

```
mean(PovPct)
```

```
## [1] 12.94
```

```
huberM(PovPct)$mu
```

```
## [1] 12.63372
```

```
median(PovPct)
```

```
## [1] 12.15
```

```
mean(Brth15to17)
```

```
## [1] 21.832
```

```
huberM(Brth15to17)$mu
```

```
## [1] 21.32322
```

```
median(Brth15to17)
```

```
## [1] 19.8
```

```
mean(Brth18to19)
```

```
## [1] 71.43
```

```
huberM(Brth18to19)$mu
```

```
## [1] 71.33912
```

```
median(Brth18to19)
```

```
## [1] 69.25
```

```
mean(ViolCrime)
```

```
## [1] 6.712
```

```
huberM(ViolCrime)$mu
```

```
## [1] 6.623571
```

```
median(ViolCrime)
```

```
## [1] 6.25
```

```
mean(TeenBrth)
```

```
## [1] 41.706
```

```
huberM(TeenBrth)$mu
```

```
## [1] 41.36206
```

```
median(TeenBrth)
```

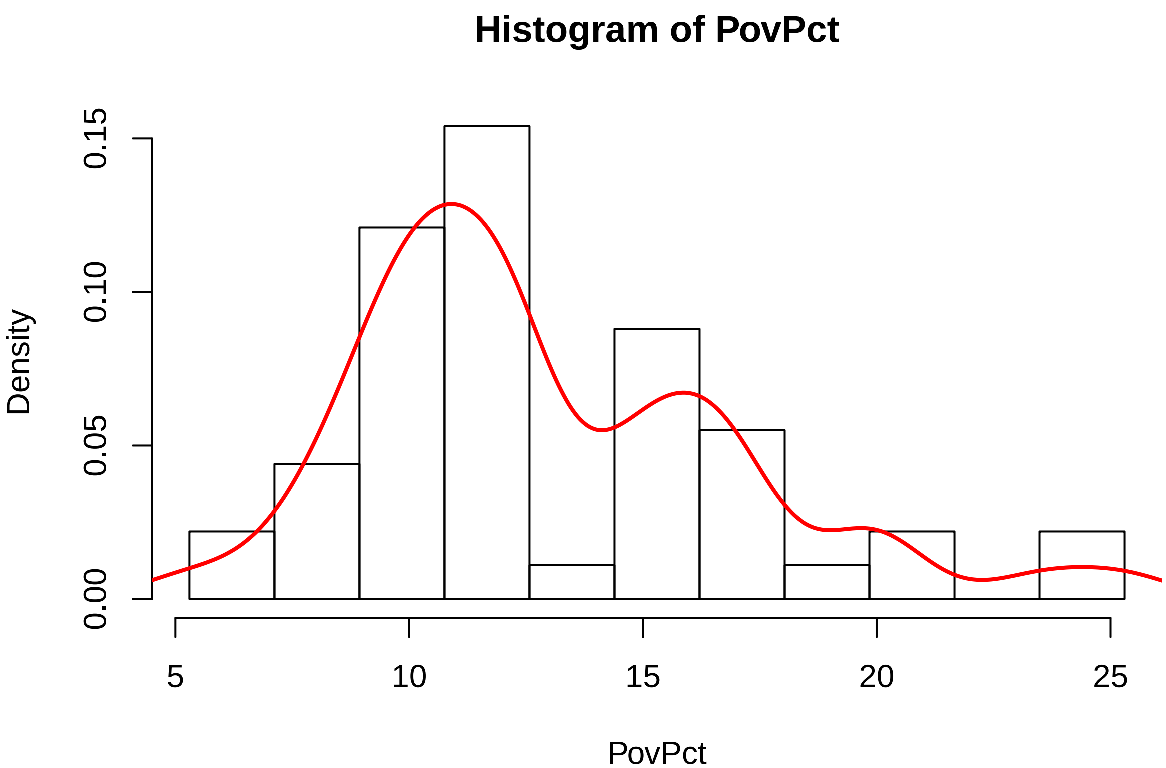
```
## [1] 39.5
```

Secinājums: Hubera M-statistika visur tuvu seko vidējai vērtībai, bet mediāna vietām ir diezgan nobīdīta. Tas skaidrojams ar izlēcēju klātbūtni vai sadalījumu asimetriskumu.

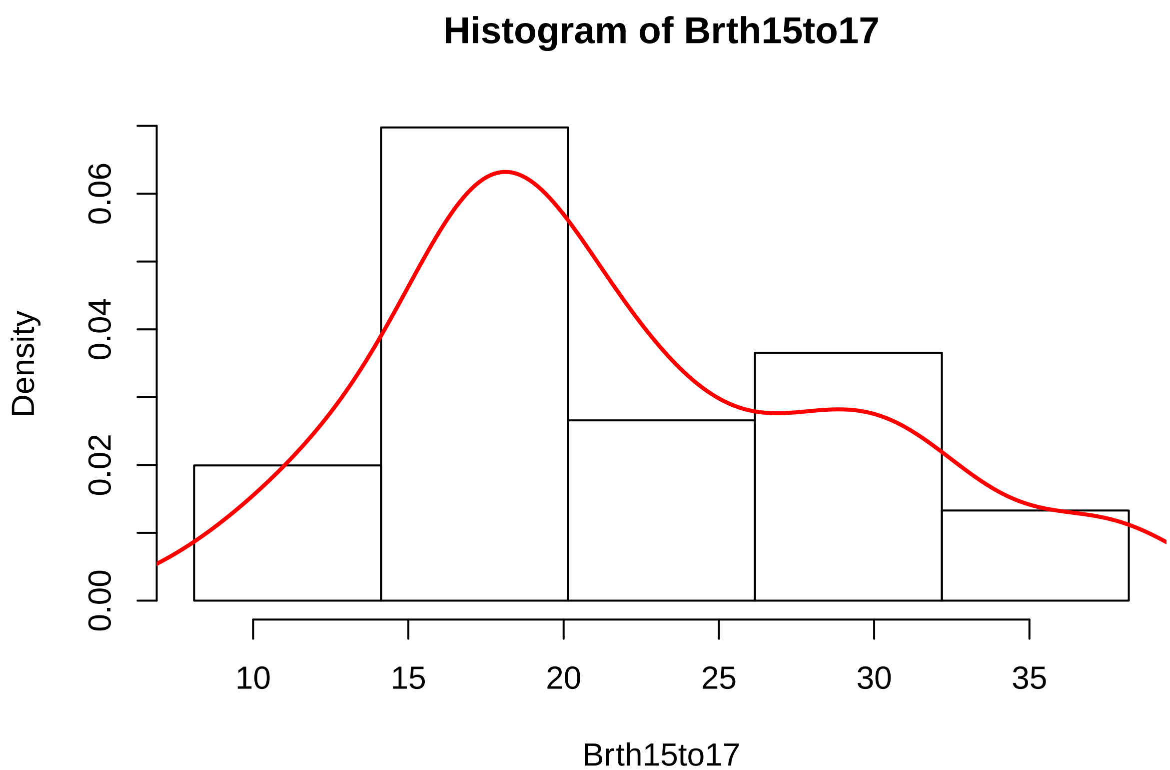
1.3 Histogrammas.

Binu platums iegūts ar krosvalidācijas metodi un netiek mainīts, pat ja rezultējošā histogramma ir pilnīgi bezjēdzīga:

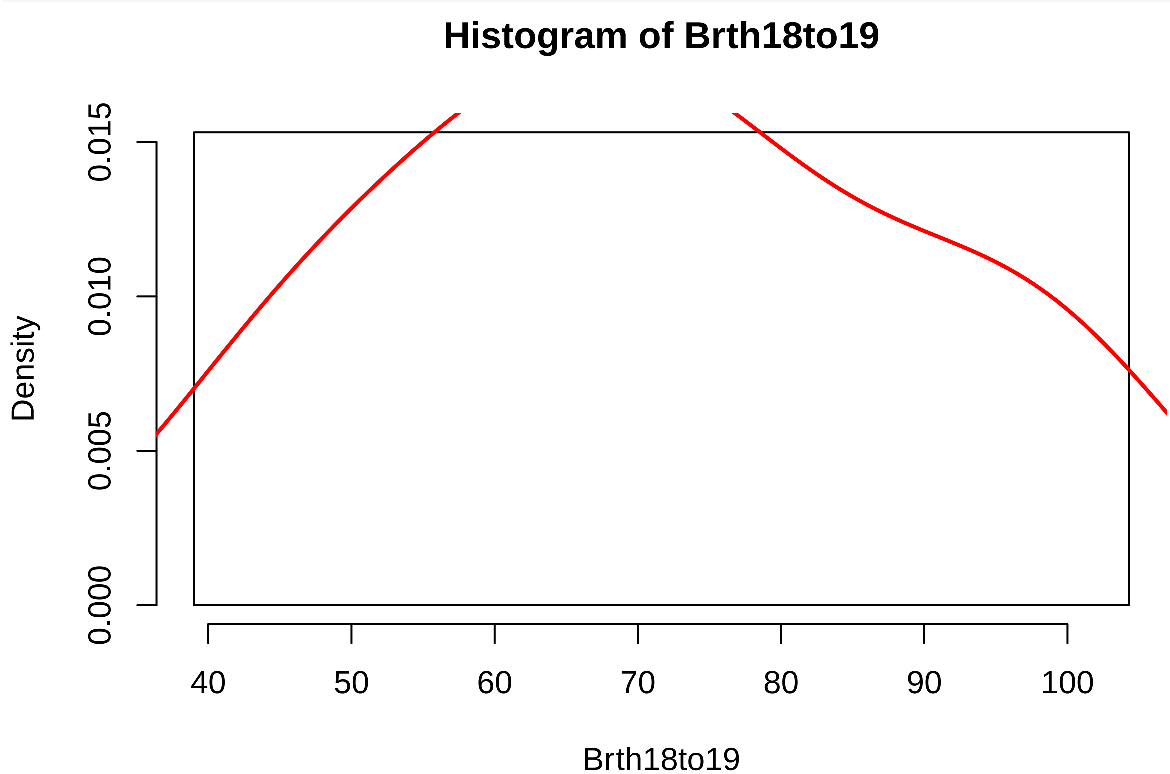
```
hh<-histogram(PovPct,type="regular",penalty="cv")
lines(density(PovPct,bw="ucv"),col="red",lwd=2)
```



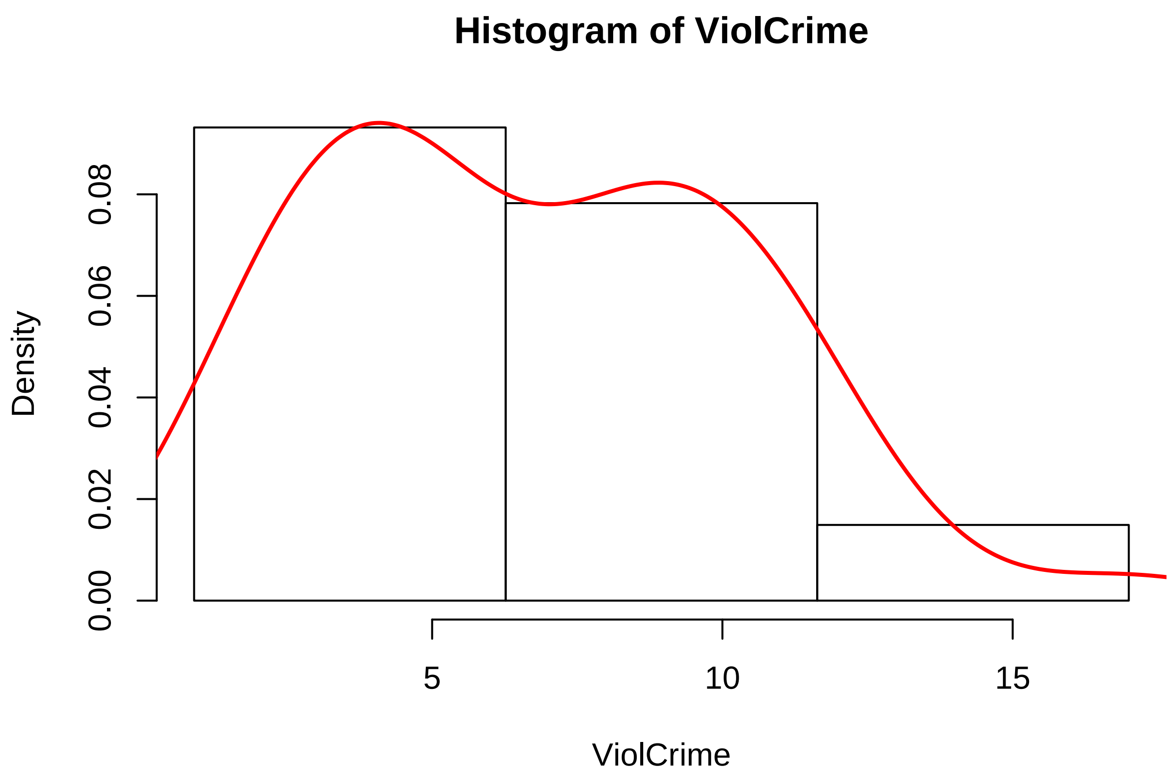

```
hh<-histogram(Brth15to17,type="regular",penalty="cv")
lines(density(Brth15to17,bw="ucv"),col="red",lwd=2)
```



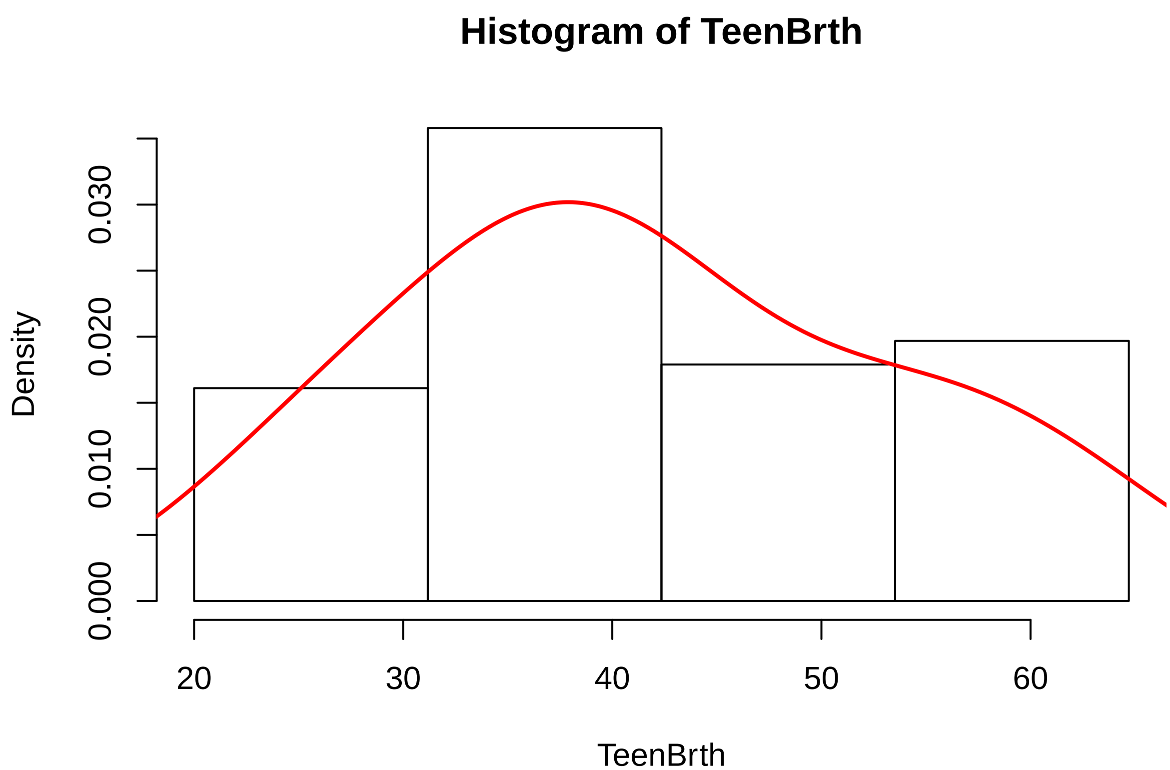

```
hh<-histogram(Brth18to19,type="regular",penalty="cv")
lines(density(Brth18to19,bw="ucv"),col="red",lwd=2)
```




```
hh<-histogram(ViolCrime,type="regular",penalty="cv")
lines(density(ViolCrime,bw="ucv"),col="red",lwd=2)
```




```r
hh<-histogram(TeenBrth,type="regular",penalty="cv")
lines(density(TeenBrth,bw="ucv"),col="red",lwd=2)
```



1.4 Vai skaitlisko datu kolonnas ir normāli sadalītas?

```
(lillie.test(PovPct)$p > 0.05)
```

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

```
(lillie.test(Brth15to17)$p > 0.05)
```

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

```
(lillie.test(Brth18to19)$p > 0.05)
```

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

```
(lillie.test(ViolCrime)$p > 0.05)
```

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

```
(lillie.test(TeenBrth)$p > 0.05)
```

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

1.5 Asimetrijas koeficienta novērtējums ar butstrapa metodi

```
N<-10000
bobj_skew <- boot(PovPct,statistic=cb_skewness,R=N)
boot.ci(bobj_skew)
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 10000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = bobj_skew)
##
## Intervals :
## Level      Normal              Basic
## 95%   ( 0.3307,  1.5563 )   ( 0.3341,  1.5761 )
##
## Level     Percentile            BCa
## 95%   ( 0.1992,  1.4412 )   ( 0.3069,  1.5569 )
## Calculations and Intervals on Original Scale
```

```
bobj_skew <- boot(Brth15to17,statistic=cb_skewness,R=N)
boot.ci(bobj_skew)
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 10000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = bobj_skew)
##
## Intervals :
## Level      Normal              Basic
## 95%   ( 0.0752,  0.9601 )   ( 0.0556,  0.9460 )
##
## Level     Percentile            BCa
## 95%   ( 0.0883,  0.9787 )   ( 0.0917,  0.9798 )
## Calculations and Intervals on Original Scale
```

```r
bobj_skew <- boot(Brth18to19,statistic=cb_skewness,R=N)
boot.ci(bobj_skew)
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 10000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = bobj_skew)
##
## Intervals :
## Level      Normal              Basic
## 95%   (-0.2383,  0.4855 )   (-0.2445,  0.4837 )
##
## Level     Percentile            BCa
## 95%   (-0.2301,  0.4980 )   (-0.2356,  0.4909 )
## Calculations and Intervals on Original Scale
```

```r
bobj_skew <- boot(ViolCrime,statistic=cb_skewness,R=N)
boot.ci(bobj_skew)
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 10000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = bobj_skew)
##
## Intervals :
## Level      Normal              Basic
## 95%   (-0.1060,  1.0381 )   (-0.1050,  1.0120 )
##
## Level     Percentile            BCa
## 95%   (-0.1789,  0.9382 )   (-0.0856,  1.0695 )
## Calculations and Intervals on Original Scale
```

```r
bobj_skew <- boot(TeenBrth,statistic=cb_skewness,R=N)
boot.ci(bobj_skew)
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 10000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = bobj_skew)
##
## Intervals :
## Level      Normal              Basic
## 95%   (-0.1315,  0.6432 )   (-0.1419,  0.6335 )
##
## Level     Percentile            BCa
## 95%   (-0.1200,  0.6555 )   (-0.1221,  0.6507 )
## Calculations and Intervals on Original Scale
```

1.6 Korelācijas koeficienti:

```r
df1 <- df[c("PovPct","Brth15to17","Brth18to19","ViolCrime","TeenBrth")]
cor(df1)[1,]
```

```
##     PovPct Brth15to17 Brth18to19  ViolCrime   TeenBrth
##  1.0000000  0.6988130  0.6269856  0.5156478  0.6731801
```

```
cor(df1, method="spearman")[1,]
```

```
##     PovPct Brth15to17 Brth18to19  ViolCrime   TeenBrth
##  1.0000000  0.5676092  0.5691432  0.4435261  0.5787894
```

```
# is cor significant?
x<-cor.test(PovPct,Brth15to17)
(x$p.value < 0.05)
```

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

```
x$conf.int[1:2]
```

```
## [1] 0.5220014 0.8180411
```

```
x<-cor.test(PovPct,Brth18to19)
(x$p.value < 0.05)
```

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

```
x$conf.int[1:2]
```

```
## [1] 0.4223457 0.7708113
```

```
x<-cor.test(PovPct,ViolCrime)
(x$p.value < 0.05)
```

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

```
x$conf.int[1:2]
```

```
## [1] 0.2770673 0.6943376
```

```
x<-cor.test(PovPct,TeenBrth)
(x$p.value < 0.05)
```

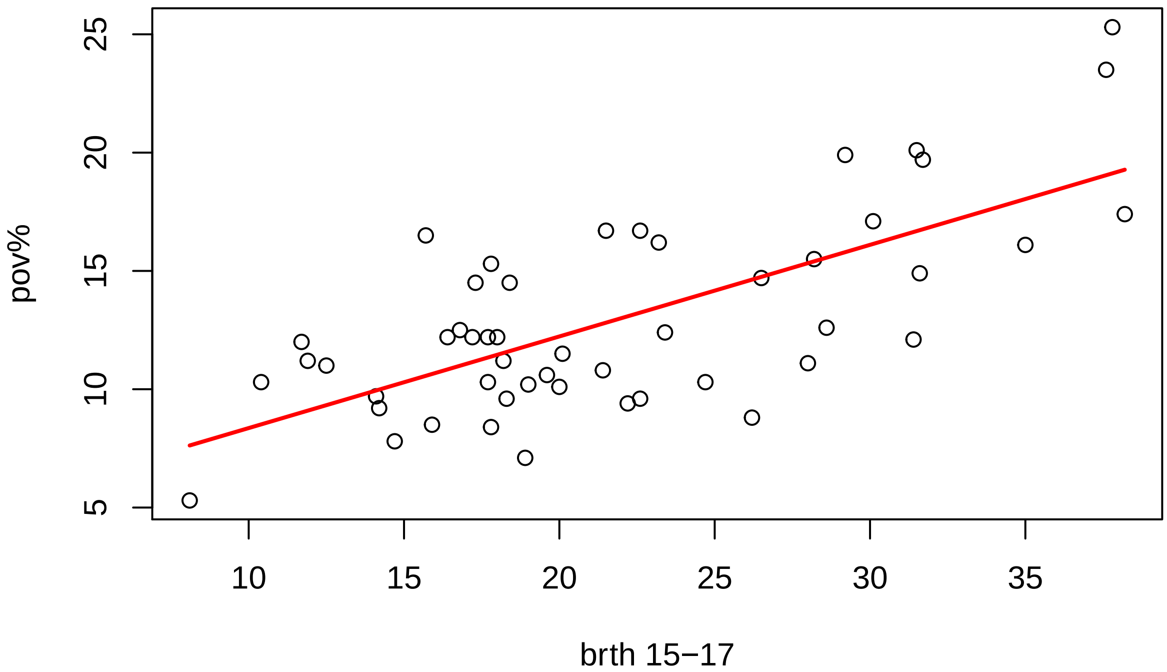
```
## [1] TRUE
```

```
x$conf.int[1:2]
```

```
## [1] 0.4858746 0.8013688
```

1.7 Lineārās regresijas:

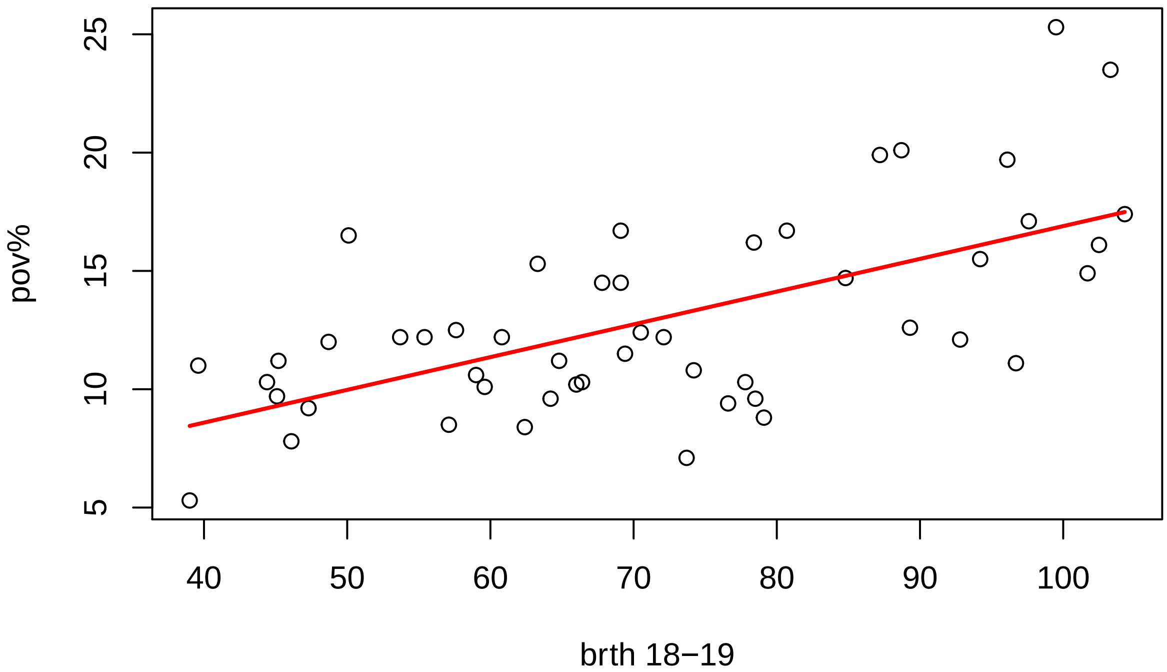
```
fit<-general_lreg(Brth15to17,PovPct,plot=T,names=c("brth 15-17","pov%"))
```




```
summary(fit)
```

```
##
## Call:
## lm(formula = vec2 ~ poly(vec1, degree, raw = T))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.8312 -2.0912 -0.2901  2.5741  6.1775
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   4.4871     1.3181   3.404  0.00135 **
## poly(vec1, degree, raw = T)   0.3872     0.0572   6.768 1.67e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.982 on 48 degrees of freedom
## Multiple R-squared:  0.4883, Adjusted R-squared:  0.4777
## F-statistic: 45.81 on 1 and 48 DF,  p-value: 1.666e-08
```

```
fit<-general_lreg(Brth18to19,PovPct,plot=T,names=c("brth 18-19","pov%"))
```




```
summary(fit)
```

```
##
## Call:
## lm(formula = vec2 ~ poly(vec1, degree, raw = T))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.1542 -2.3119 -0.4056  2.0195  8.4746
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   3.05279    1.83169   1.667    0.102
## poly(vec1, degree, raw = T)  0.13842    0.02482   5.576 1.11e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.248 on 48 degrees of freedom
## Multiple R-squared:  0.3931, Adjusted R-squared:  0.3805
## F-statistic: 31.09 on 1 and 48 DF,  p-value: 1.106e-06
```

1.8 Daudzfaktoru lineārās regresijas:

```
fit_multi<-lm(PovPct~Brth15to17+Brth18to19, data=df1)
summary(fit_multi)
```

```
##
## Call:
## lm(formula = PovPct ~ Brth15to17 + Brth18to19, data = df1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.1177 -2.2548 -0.3315  2.5948  5.2562
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.43963    1.95904   3.287  0.00192 **
## Brth15to17   0.63235    0.19178   3.297  0.00186 **
## Brth18to19  -0.10227    0.07642  -1.338  0.18724
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.958 on 47 degrees of freedom
## Multiple R-squared:  0.5071, Adjusted R-squared:  0.4861
## F-statistic: 24.18 on 2 and 47 DF,  p-value: 6.017e-08
```

```r
fit_multi<-lm(PovPct~., data=df1)
summary(fit_multi)
```

```
##
## Call:
## lm(formula = PovPct ~ ., data = df1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.1641 -2.0823  0.1841  1.6865  5.7184
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.5262     1.8294   3.567  0.00087 ***
## Brth15to17   -0.6910     0.4807  -1.437  0.15750
## Brth18to19   -0.9706     0.2959  -3.281  0.00200 **
## ViolCrime     0.1197     0.1697   0.705  0.48427
## TeenBrth      2.1587     0.7143   3.022  0.00413 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.756 on 45 degrees of freedom
## Multiple R-squared:  0.5904, Adjusted R-squared:  0.554
## F-statistic: 16.22 on 4 and 45 DF,  p-value: 2.704e-08
```

1.9 Robustās regresijas dzimstības rādītājiem atsevišķi:

```r
fit<-lmrob(PovPct~Brth15to17,data=df1)
summary(fit)
```

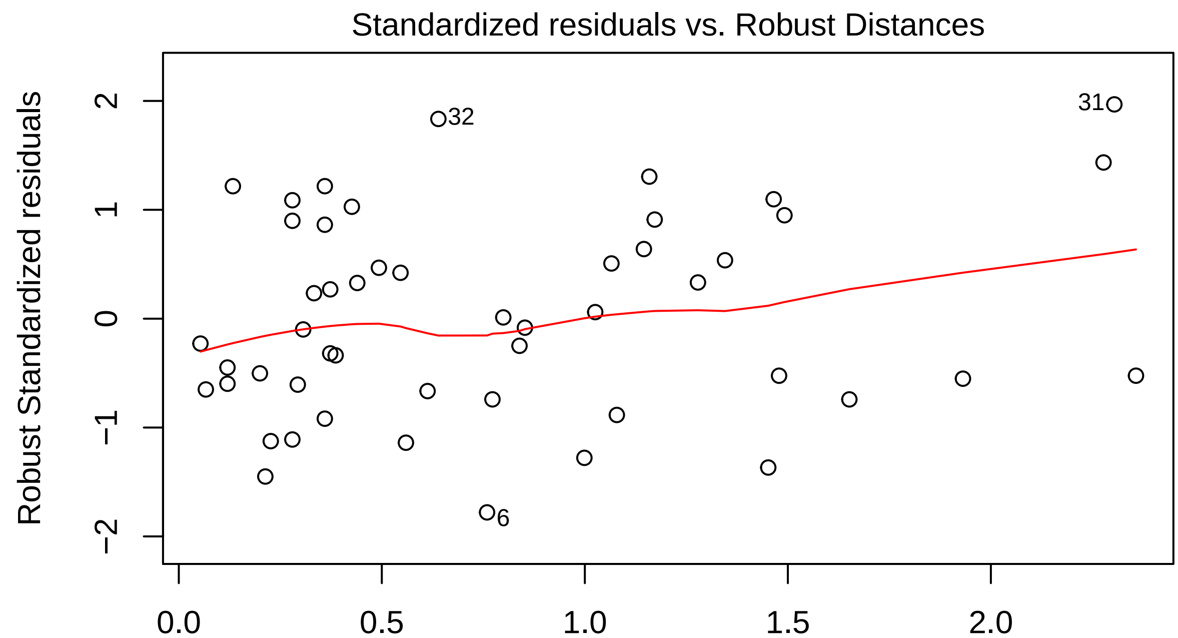
```
##
## Call:
## lmrob(formula = PovPct ~ Brth15to17, data = df1)
##  \--> method = "MM"
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7486 -2.0652 -0.2934  2.6069  6.3594
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.6285     1.4615   3.167  0.00268 **
## Brth15to17    0.3786     0.0736   5.144 4.92e-06 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Robust residual standard error: 3.231
## Multiple R-squared:  0.4477, Adjusted R-squared:  0.4362
## Convergence in 12 IRWLS iterations
##
## Robustness weights:
##  4 weights are ~= 1. The remaining 46 ones are summarized as
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.6781  0.8888  0.9506  0.9223  0.9770  0.9952
## Algorithmic parameters:
##        tuning.chi                bb        tuning.psi        refine.tol
##         1.548e+00         5.000e-01         4.685e+00         1.000e-07
##           rel.tol         scale.tol         solve.tol       eps.outlier
##         1.000e-07         1.000e-10         1.000e-07         2.000e-03
##             eps.x warn.limit.reject warn.limit.meanrw
##         6.949e-11         5.000e-01         5.000e-01
##      nResample         max.it       best.r.s       k.fast.s          k.max
##            500             50              2              1            200
##    maxit.scale      trace.lev            mts     compute.rd fast.s.large.n
##            200              0           1000              0           2000
##                   psi           subsampling                   cov
##            "bisquare"         "nonsingular"         ".vcov.avar1"
## compute.outlier.stats
##                  "SM"
## seed : int(0)
```

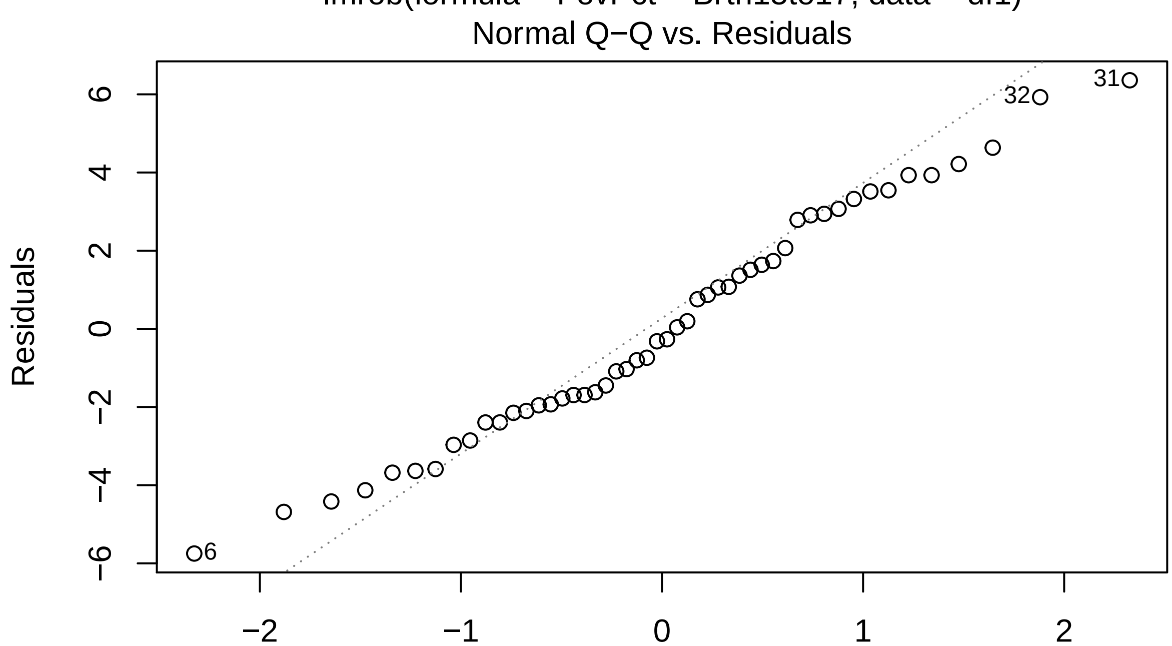
```
plot(fit)
```

```
## recomputing robust Mahalanobis distances
```

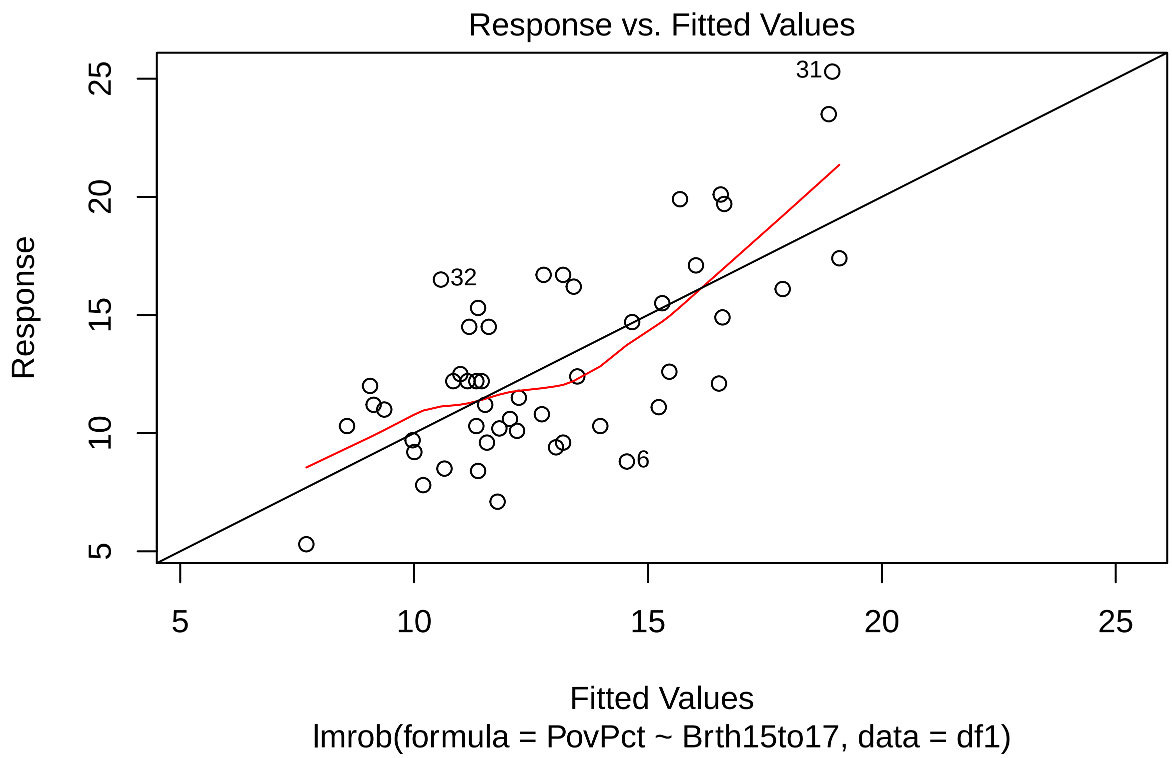
```
## saving the robust distances 'MD' as part of 'fit'
```

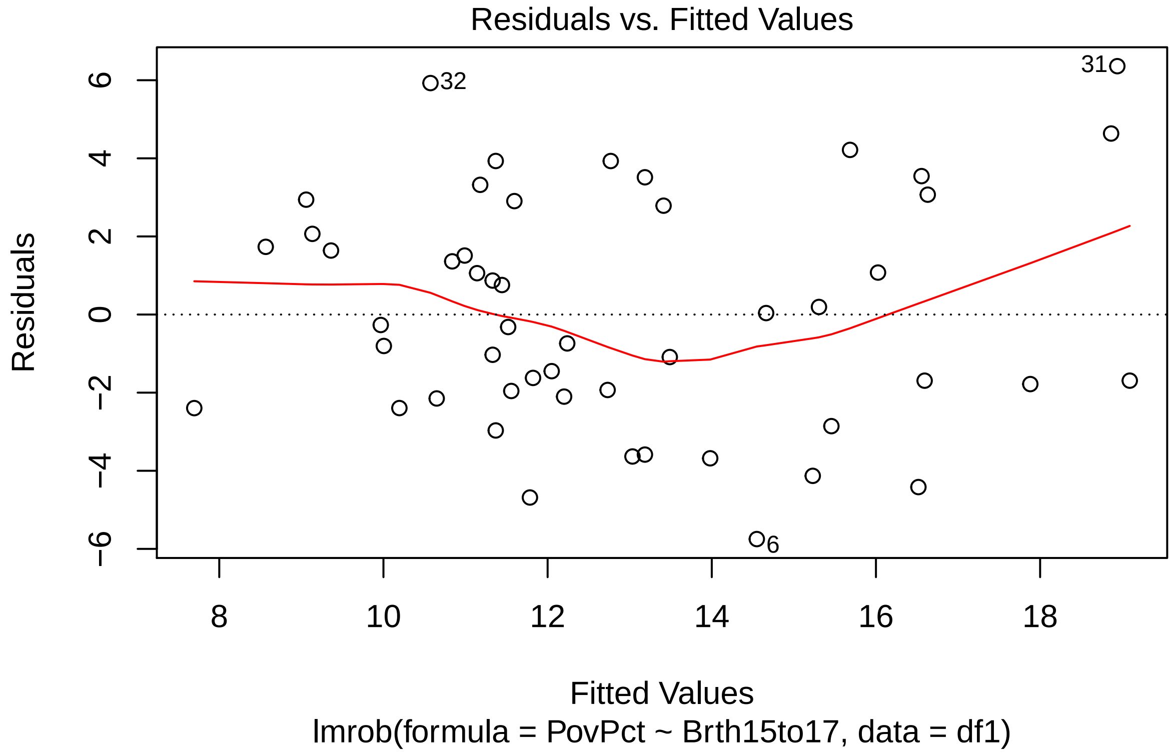


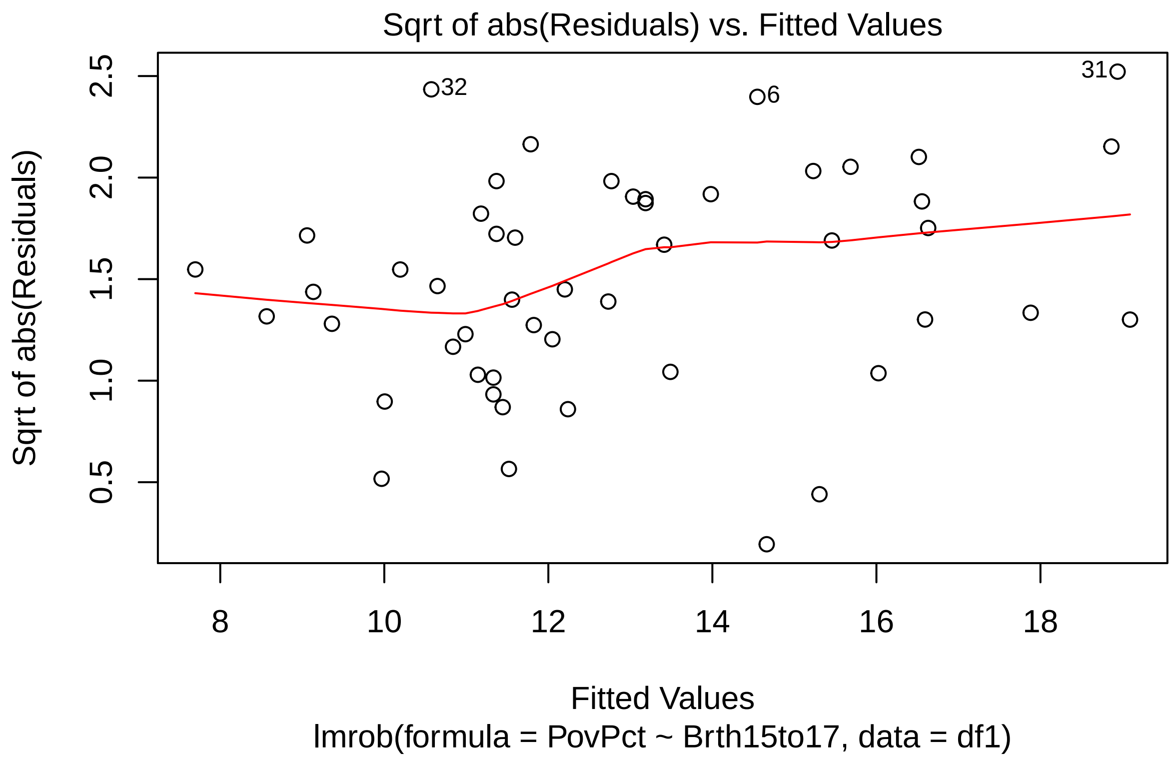
Robust Distances
lmrob(formula = PovPct ~ Brth15to17, data = df1)



Theoretical Quantiles
lmrob(formula = PovPct ~ Brth15to17, data = df1)


Response vs. Fitted Values
Response
Fitted Values
lmrob(formula = PovPct ~ Brth15to17, data = df1)
31
32
6
5
10
15
20
25


Residuals vs. Fitted Values
Residuals
Fitted Values
lmrob(formula = PovPct ~ Brth15to17, data = df1)
31
32
6
8
10
12
14
16
18
−6
−4
−2
0
2
4
6



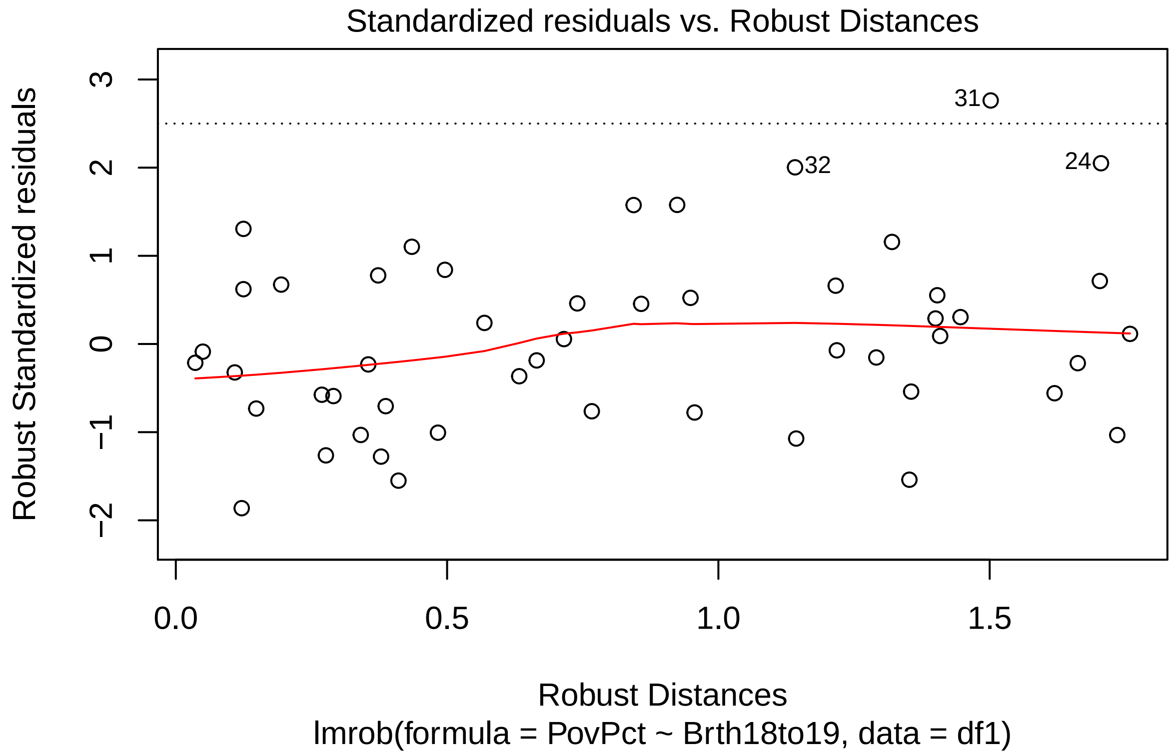

```r
fit<-lmrob(PovPct~Brth18to19,data=df1)
summary(fit)
```

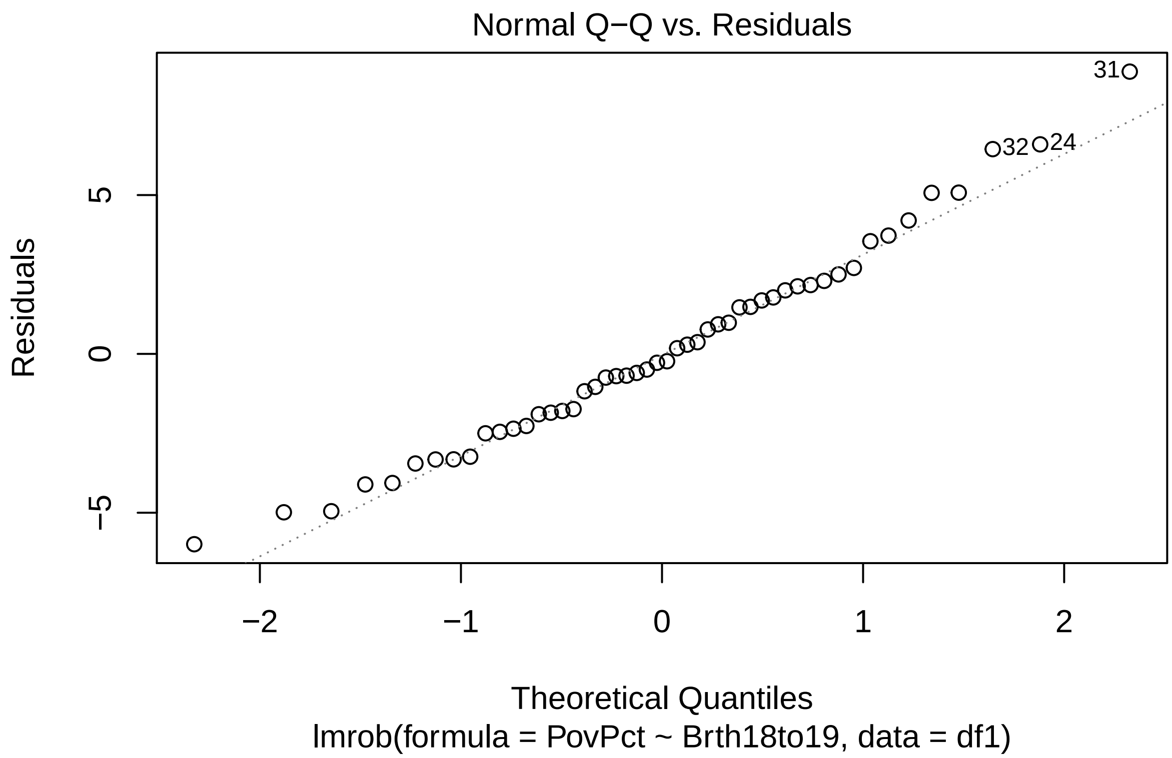
```
##
## Call:
## lmrob(formula = PovPct ~ Brth18to19, data = df1)
##  \--> method = "MM"
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.9918 -2.1765 -0.2555  2.0955  8.8863
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.60240    1.82845   1.970   0.0546 .
## Brth18to19   0.12876    0.02851   4.515 4.11e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Robust residual standard error: 3.217
## Multiple R-squared:  0.3548, Adjusted R-squared:  0.3413
## Convergence in 13 IRWLS iterations
##
## Robustness weights:
##  4 weights are ~= 1. The remaining 46 ones are summarized as
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.4257  0.8843  0.9546  0.9093  0.9811  0.9988
## Algorithmic parameters:
##        tuning.chi                bb        tuning.psi        refine.tol
##         1.548e+00         5.000e-01         4.685e+00         1.000e-07
##           rel.tol         scale.tol         solve.tol       eps.outlier
```

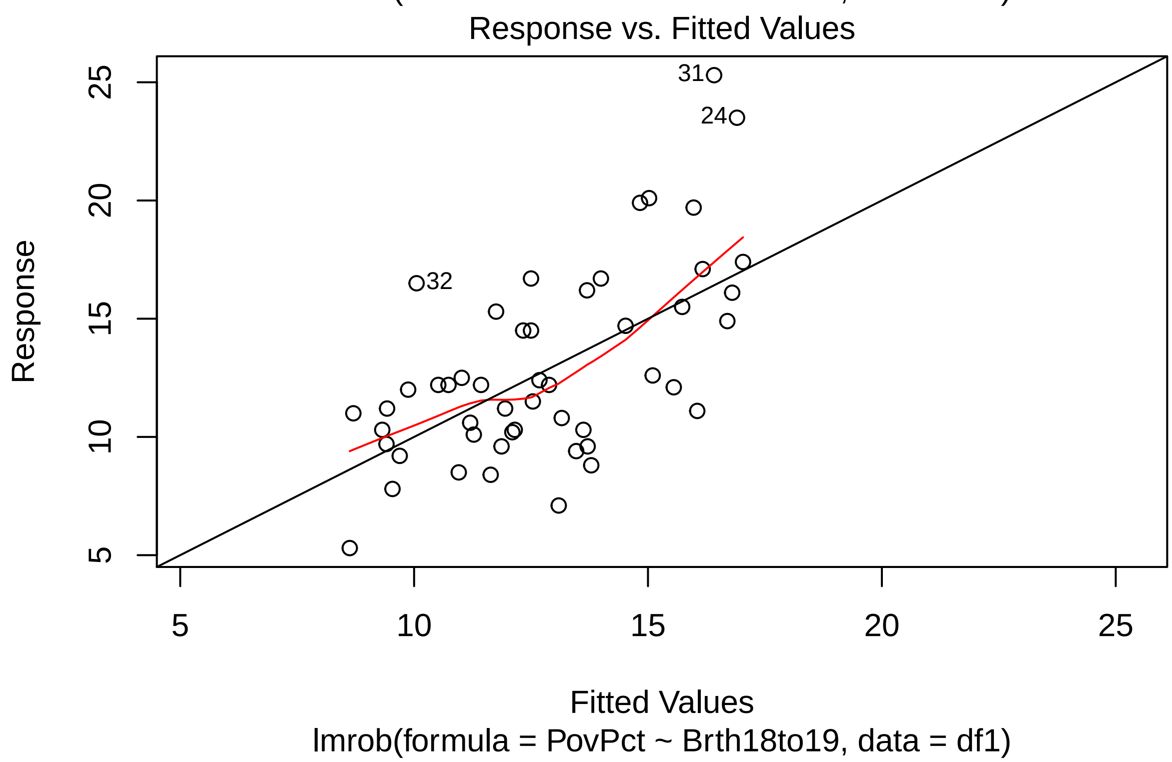
```
##         1.000e-07         1.000e-10         1.000e-07         2.000e-03
##             eps.x warn.limit.reject warn.limit.meanrw
##         1.897e-10         5.000e-01         5.000e-01
##       nResample            max.it       best.r.s       k.fast.s           k.max
##             500                50              2              1             200
##     maxit.scale         trace.lev            mts     compute.rd fast.s.large.n
##             200                 0           1000              0            2000
##                   psi           subsampling                    cov
##            "bisquare"         "nonsingular"          ".vcov.avar1"
## compute.outlier.stats
##                  "SM"
## seed : int(0)
```

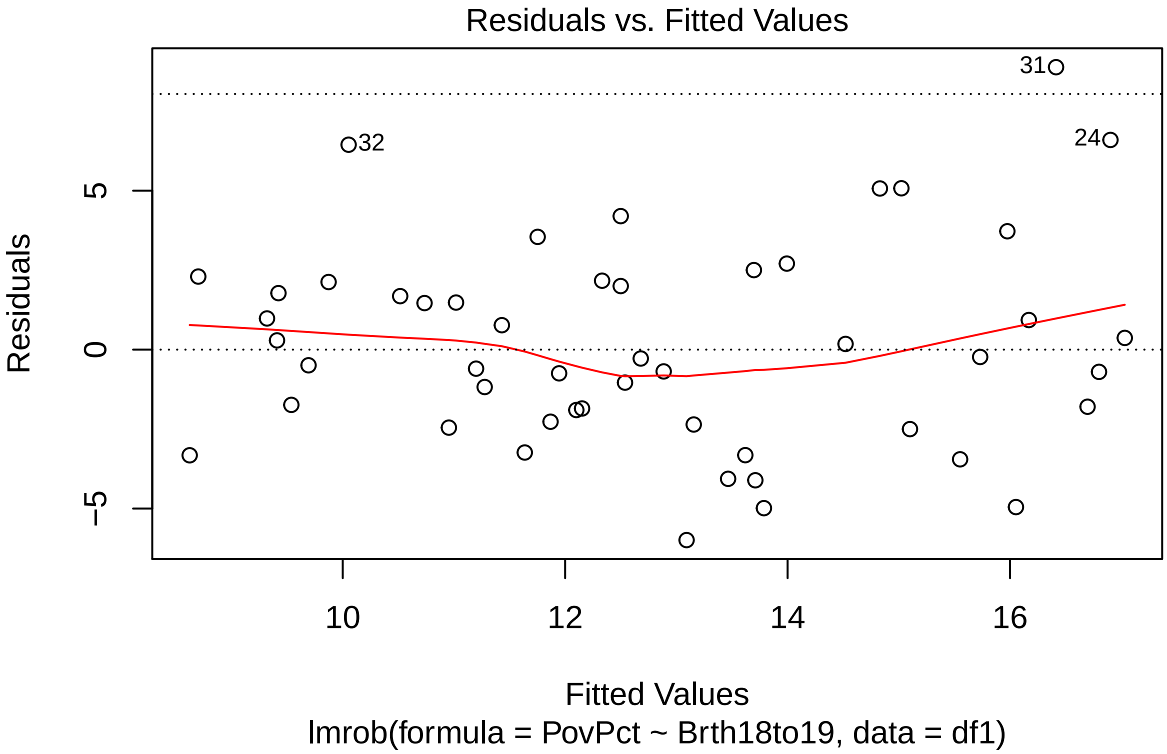
```
plot(fit)
```

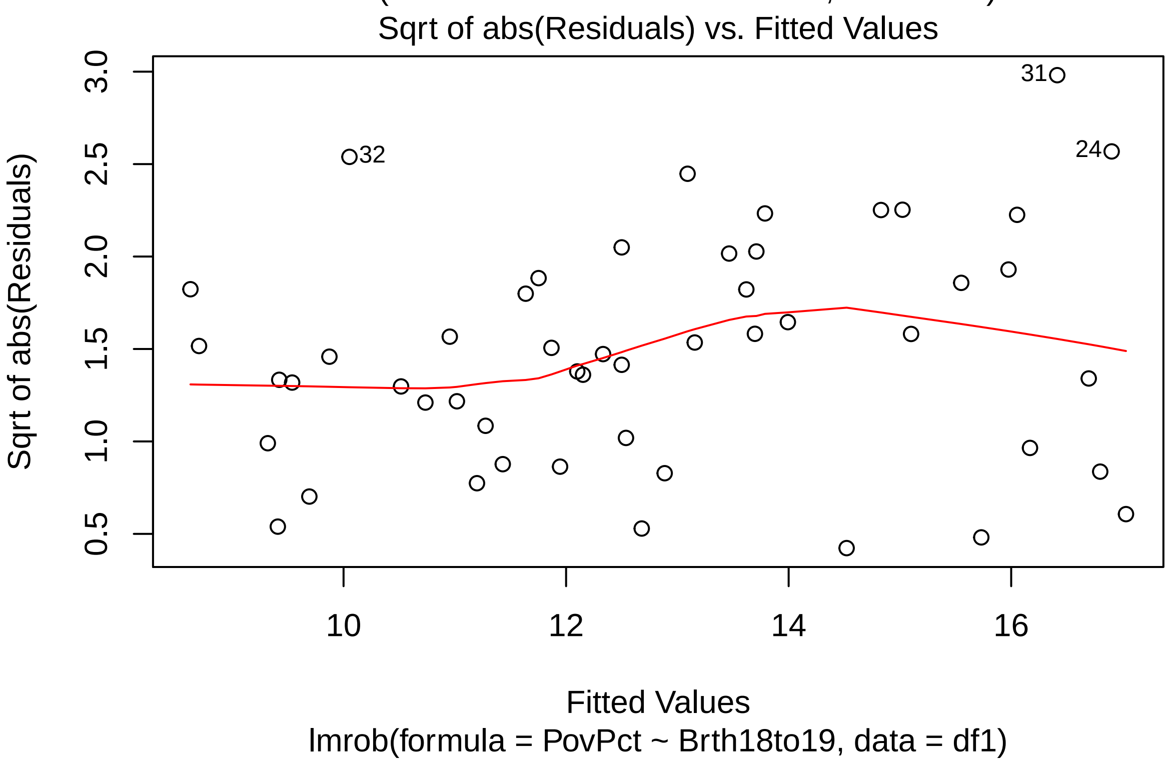
```
## recomputing robust Mahalanobis distances
## saving the robust distances 'MD' as part of 'fit'
```




Normal Q−Q vs. Residuals
Residuals
Theoretical Quantiles
lmrob(formula = PovPct ~ Brth18to19, data = df1)
31
32
24
−5
0
5
−2
−1
0
1
2


Response vs. Fitted Values
Response
Fitted Values
lmrob(formula = PovPct ~ Brth18to19, data = df1)
31
24
32
5
10
15
20
25
5
10
15
20
25





Secinājums: robustās regresijas šajā gadījumā sniedz sliktākus determinācijas koeficientus un atrod 2-3 izlēcējus katrā gadījumā.

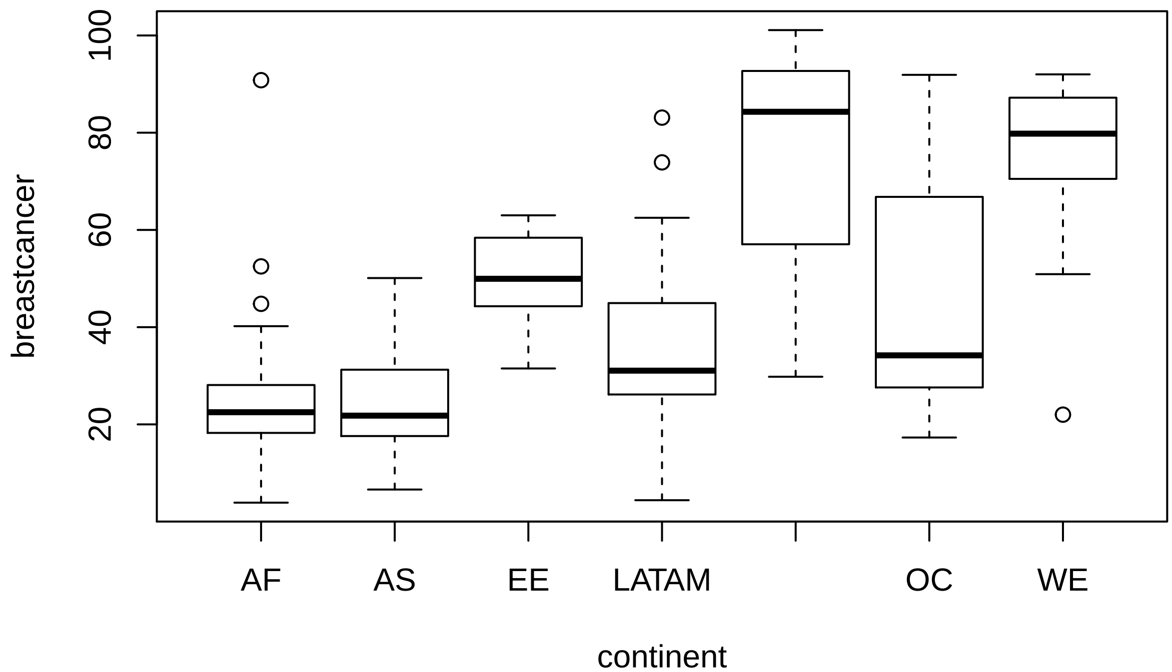
2. uzdevums

Dati un bibliotēkas:

```
library(rstatix)
df <- read.csv("gapC.csv")
attach(df)
```

2.1 Kastu grafiki pa kontinentiem:

```
boxplot(breastcancer~continent)
```



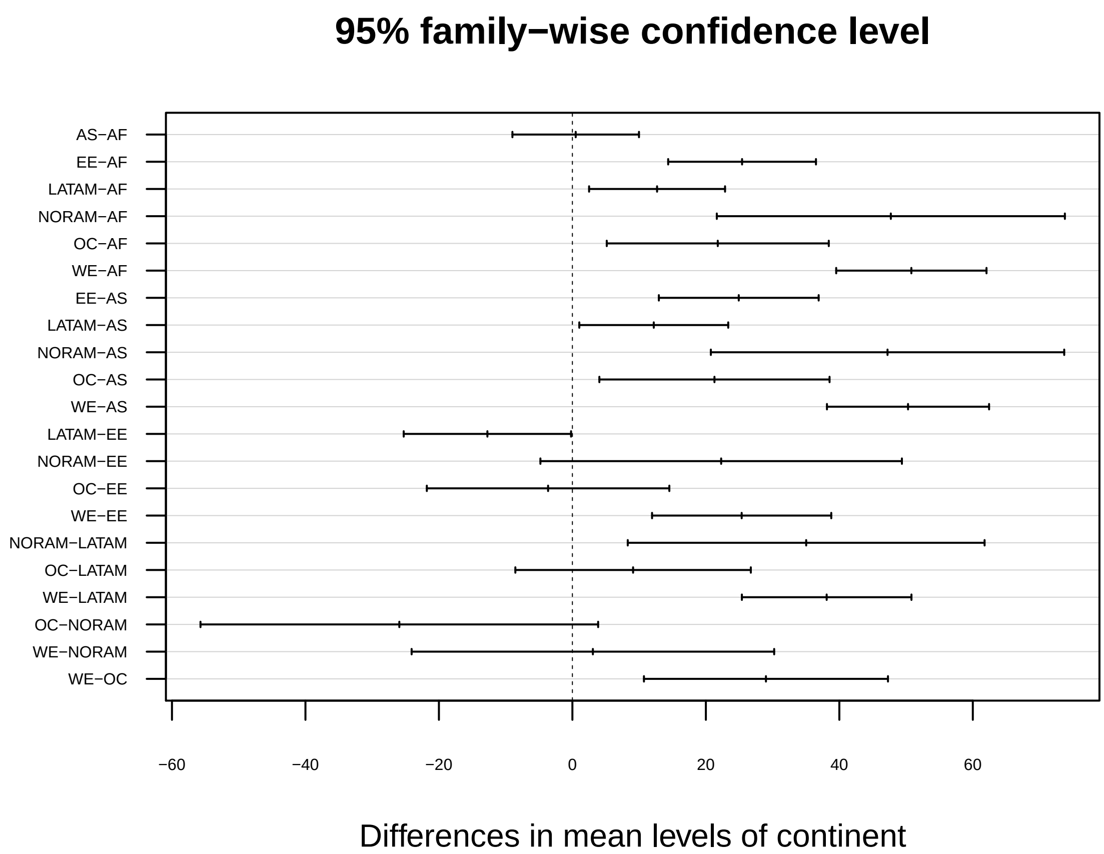
2.2 ANOVA pa kontinentiem - noraida vidējo vērtību vienādību:

```
fit<-aov(breastcancer~continent)
summary(fit)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## continent     6  52531    8755   40.28 <2e-16 ***
## Residuals   166  36083     217
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

2.3 Post-Hoc tests - atrod grupas, kuru vidējās vērtības varētu būt vienādas:

```
f<-TukeyHSD(fit)
op <- par(mar= c(4,5,3,3) + 0.1, cex.axis=0.5)
plot(f,las=1)
```




```
par(op)
f
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = breastcancer ~ continent)
##
## $continent
##                   diff        lwr        upr     p adj
## AS-AF        0.4953571  -8.986848  9.9775626 0.9999987
## EE-AF       25.4248377  14.352007 36.4976680 0.0000000
## LATAM-AF    12.6875000   2.501977 22.8730225 0.0050172
## NORAM-AF    47.7172619  21.638434 73.7960896 0.0000035
## OC-AF       21.7839286   5.151040 38.4168172 0.0025337
## WE-AF       50.7886905  39.528172 62.0492093 0.0000000
## EE-AS       24.9294805  12.956321 36.9026399 0.0000001
## LATAM-AS    12.1921429   1.034462 23.3498237 0.0223712
## NORAM-AS    47.2219048  20.748253 73.6955563 0.0000067
## OC-AS       21.2885714   4.043225 38.5339174 0.0056343
## WE-AS       50.2933333  38.146389 62.4402777 0.0000000
## LATAM-EE   -12.7373377 -25.274849 -0.1998261 0.0437993
## NORAM-EE    22.2924242  -4.791696 49.3765447 0.1822328
## OC-EE       -3.6409091 -21.809489 14.5276712 0.9967979
## WE-EE       25.3638528  11.938369 38.7893364 0.0000015
## NORAM-LATAM 35.0297619   8.296134 61.7633901 0.0025162
## OC-LATAM     9.0964286  -8.545414 26.7382711 0.7208506
```

```
## WE-LATAM     38.1011905  25.397612 50.8047690 0.0000000
## OC-NORAM    -25.9333333 -55.725866  3.8591991 0.1332198
## WE-NORAM      3.0714286 -24.089965 30.2328219 0.9998787
## WE-OC        29.0047619  10.721189 47.2883344 0.0000943
```

2.4 Neparametriskie ekvivalenti:

```
library(rstatix)
kruskal.test(breastcancer~continent)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  breastcancer by continent
## Kruskal-Wallis chi-squared = 91.536, df = 6, p-value < 2.2e-16
```

```
dunn_test(df, breastcancer~continent)
```

```
## # A tibble: 21 x 9
##    .y.         group1 group2    n1    n2 statistic        p    p.adj p.adj.signif
##  * <chr>       <chr>  <chr>  <int> <int>     <dbl>    <dbl>    <dbl> <chr>
##  1 breastcan~ AF     AS        56    35     0.262 7.93e- 1 1.00e+ 0 ns
##  2 breastcan~ AF     EE        56    22     5.91  3.32e- 9 6.31e- 8 ****
##  3 breastcan~ AF     LATAM     56    28     3.52  4.36e- 4 6.98e- 3 **
##  4 breastcan~ AF     NORAM     56     3     2.88  3.99e- 3 5.58e- 2 ns
##  5 breastcan~ AF     OC        56     8     2.71  6.81e- 3 8.17e- 2 ns
##  6 breastcan~ AF     WE        56    21     7.58  3.54e-14 7.44e-13 ****
##  7 breastcan~ AS     EE        35    22     5.26  1.42e- 7 2.56e- 6 ****
##  8 breastcan~ AS     LATAM     35    28     2.99  2.81e- 3 4.21e- 2 *
##  9 breastcan~ AS     NORAM     35     3     2.74  6.10e- 3 7.93e- 2 ns
## 10 breastcan~ AS     OC        35     8     2.47  1.37e- 2 1.50e- 1 ns
## # ... with 11 more rows
```

2.5 Wilkoksona zīmju-rangu tests - alternatīva metode:

```
pairwise.wilcox.test(breastcancer,continent,p.adjust.method="BH")
```

```
##
##  Pairwise comparisons using Wilcoxon rank sum test
##
## data:  breastcancer and continent
##
##       AF      AS      EE      LATAM   NORAM  OC
## AS    0.7619  -       -       -       -      -
## EE    7.5e-09 6.4e-08 -       -       -      -
## LATAM 5.0e-05 0.0012  0.0010  -       -      -
## NORAM 0.0219  0.0371  0.4429  0.1522  -      -
## OC    0.0128  0.0219  0.2183  0.5858  0.4413 -
## WE    7.5e-09 2.0e-08 3.4e-05 3.3e-06 0.8053 0.0371
##
## P value adjustment method: BH
```

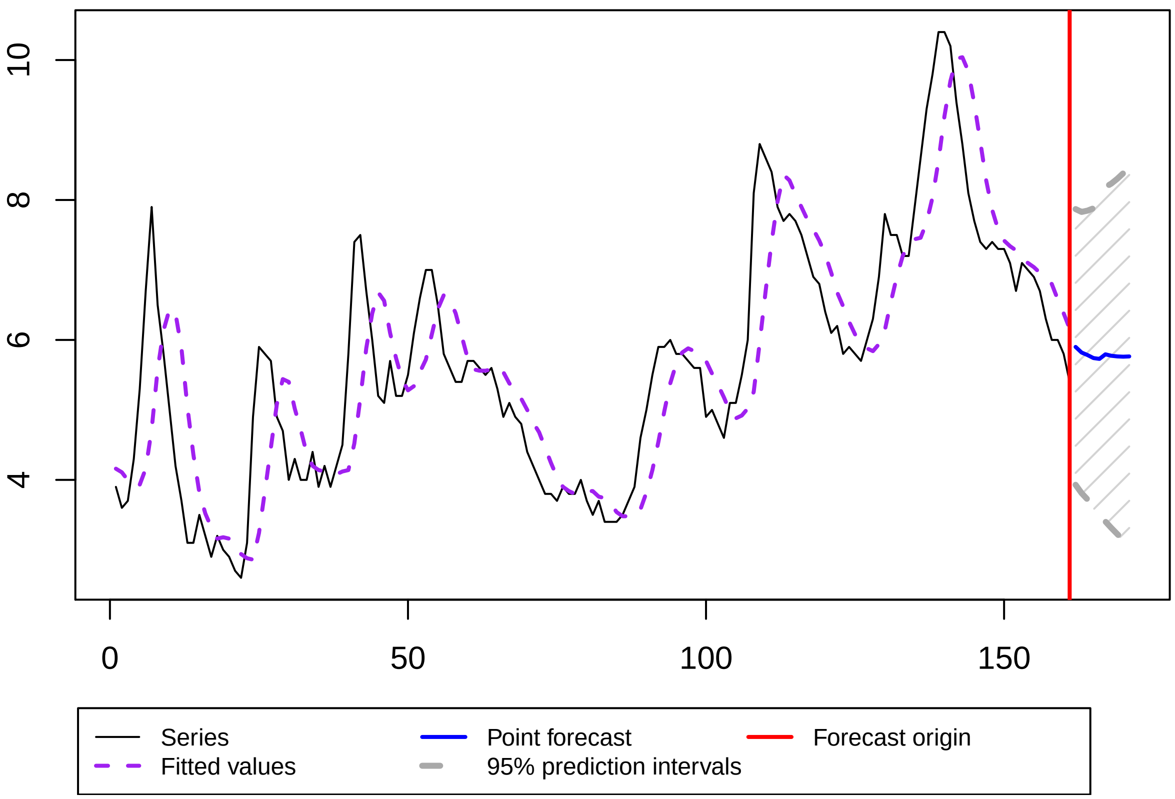
3. uzdevums

Datu ielāde, bibliotēkas:

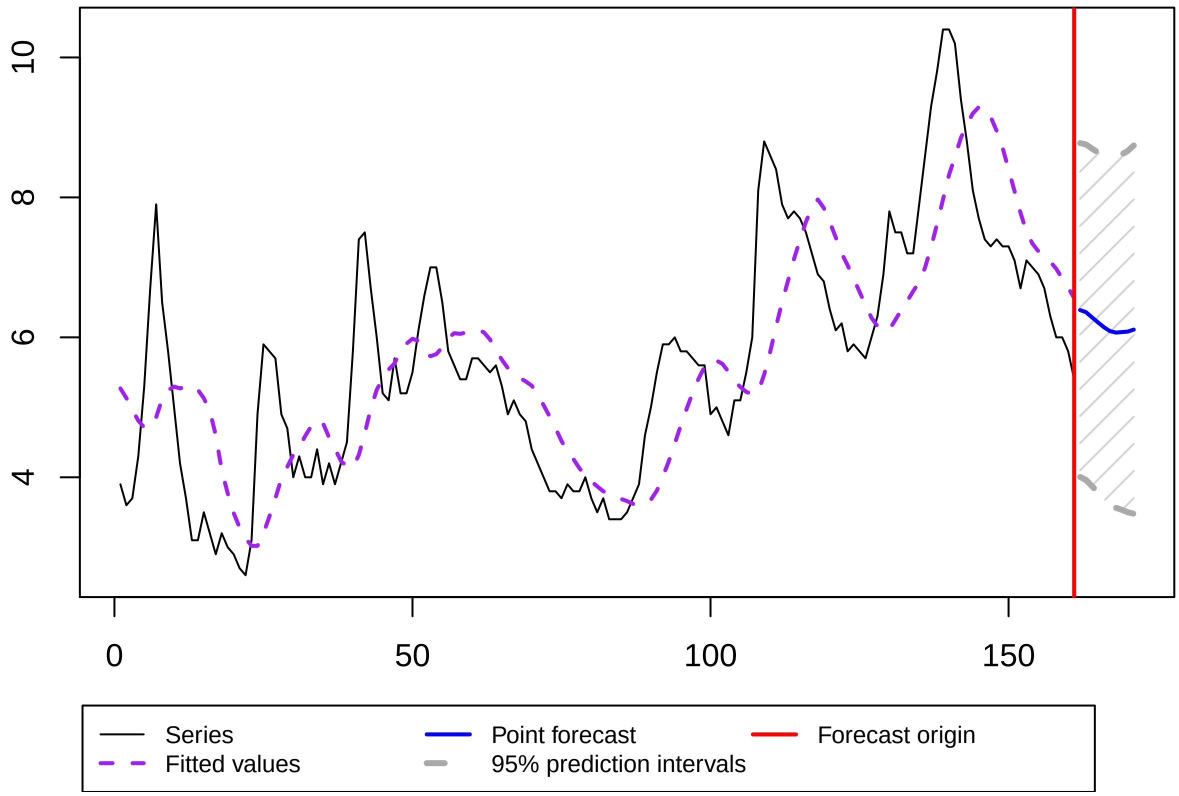
```
library(astsa)
library(smooth)
library(np)
df <- as.data.frame(econ5)
attach(df)
```

3.1 Gludināšana ar slīdošo vidējo, prognozes:

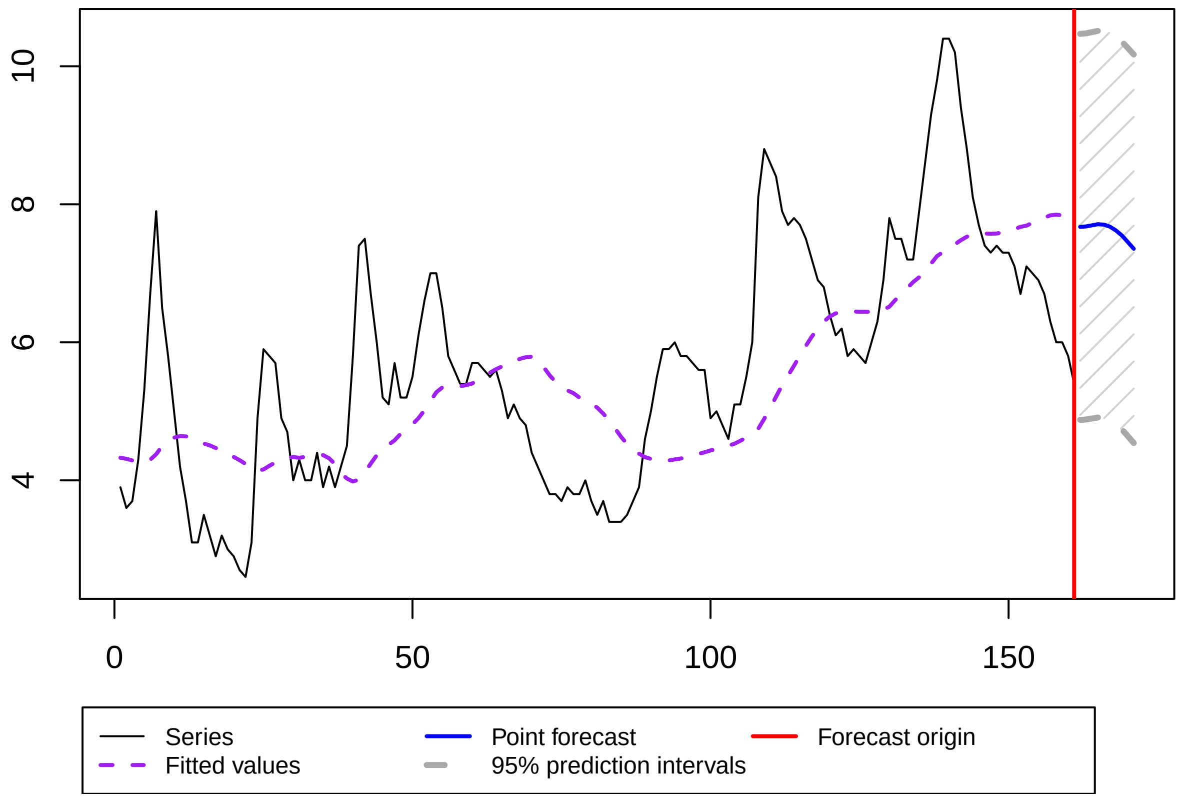
```
md<-sma(unemp,h=10,order=5)
plot(forecast(md))
```




```
md<-sma(unemp,h=10,order=10)
plot(forecast(md))
```




```
md<-sma(unemp,h=10,order=30)
plot(forecast(md))
```

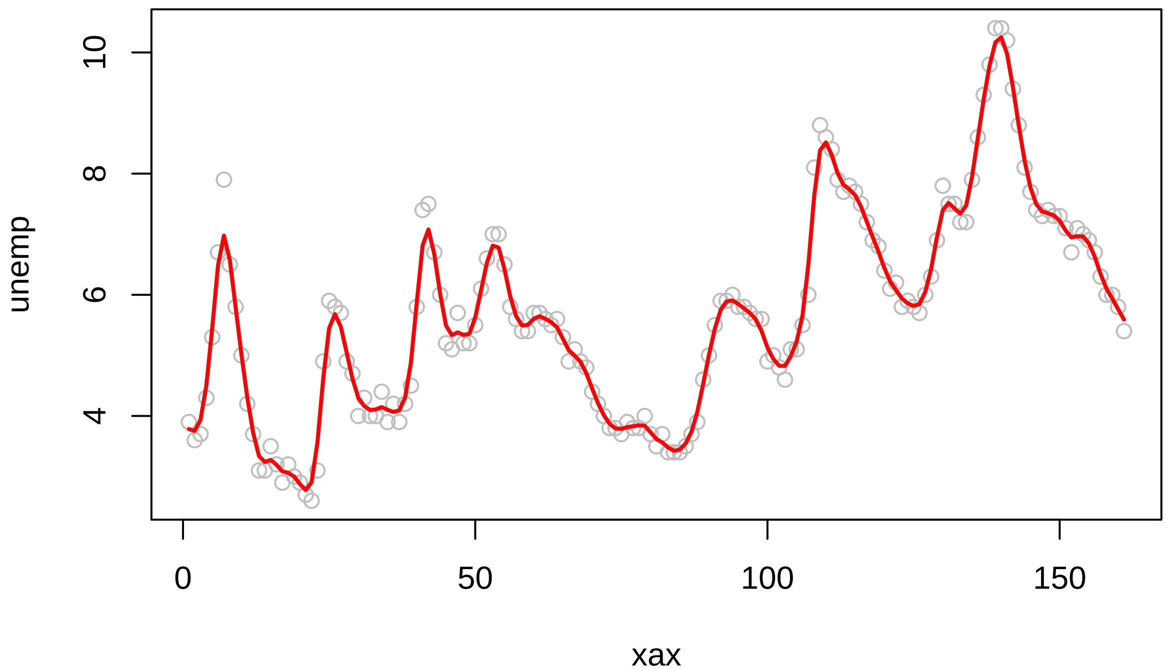


Kā redzams, lietojot mazākus logus, modelis tuvāk seko datiem intervālā, taču zaudē spēju paredzēt sakarības ilgākos laika periodos.

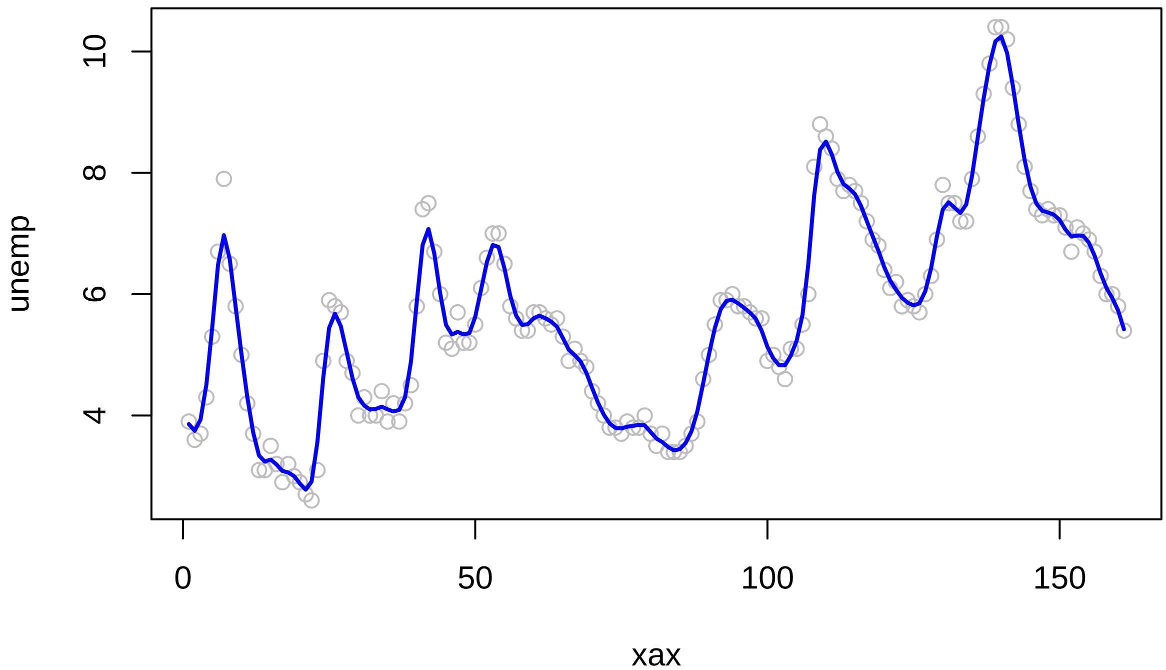
3.2 Gludināšana ar neparametriskajām regresijām

Nadaraya-Watson (local constant) vs local linear:

```
xax<-seq(1:length(unemp))
bwnw <- npregbw(unemp~xax,regtype="lc",bwmethod="cv.aic")
bwll <- npregbw(unemp~xax,regtype="ll",bwmethod="cv.aic")
nnw <- npreg(bwnw,residuals=T)
nll <- npreg(bwll,residuals=T)
plot(xax,unemp,col="grey")
lines(nnw$mean,col="red",lwd=2)
```

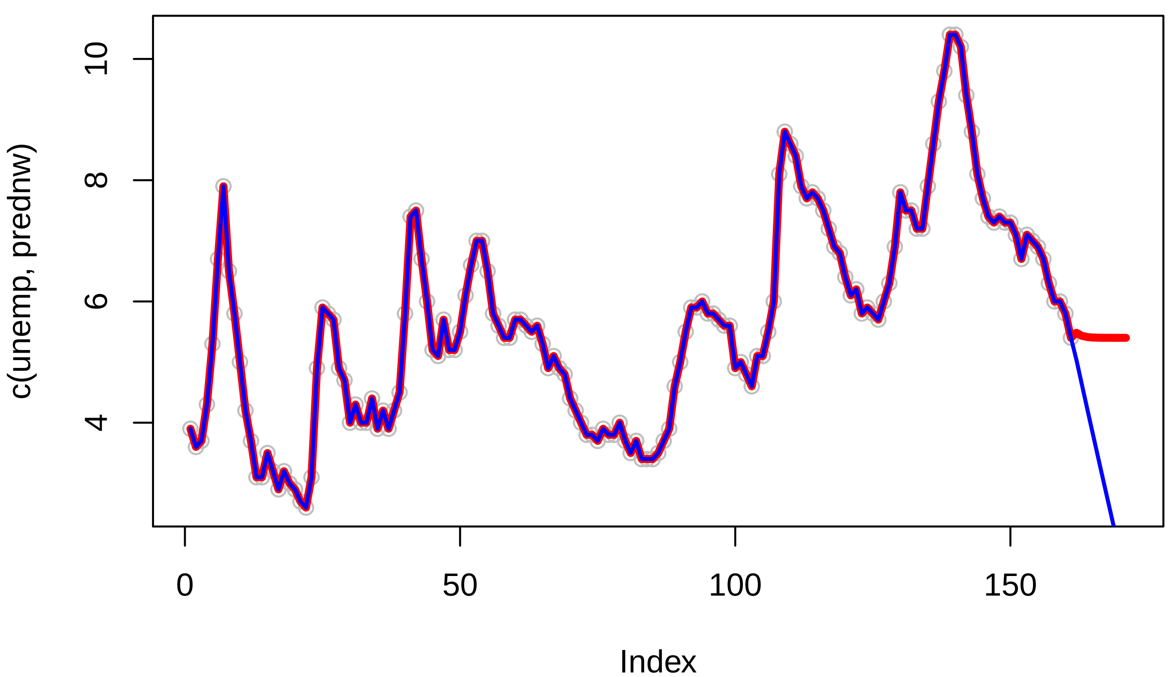



```
plot(xax,unemp,col="grey")
lines(nll$mean,col="blue",lwd=2)
```



Regresijas modeļi definīcijas intervālā izskatās gandrīz identiski, ar nelielām atšķirībām abos galos. Šo atšķirību nozīme kļūst acīmredzama, veicot ekstrapolāciju:

```
flen<-10
forecast <- seq(length(unemp)+1,length(unemp)+flen)
predll<-predict(nll, newdata=data.frame(xax=forecast))
prednw<-predict(nnw, newdata=data.frame(xax=forecast))
plot(c(unemp,prednw),col="white")
points(unemp,col="grey")
lines(c(unemp,prednw),col="red",lwd=4)
lines(c(unemp,predll),col="blue",lwd=2)
```



Kā redzams, lokāli lineārais modelis lielāku svaru liek uz lejupejošās tendences saglabāšanu netālu no datu kopas beigām, savukārt NW modelis nenovirzās tālu no datu vidējās vērtības.