Analisi dei residui nel modello di regressione lineare

Statistica Applicata
Corso di Laurea in Informatica

cristiano.varin@unive.it

Indice

1 Analisi dei residui

Illustriamo l'analisi dei residui con i dati `Prestige` contenuti in `car`[1]

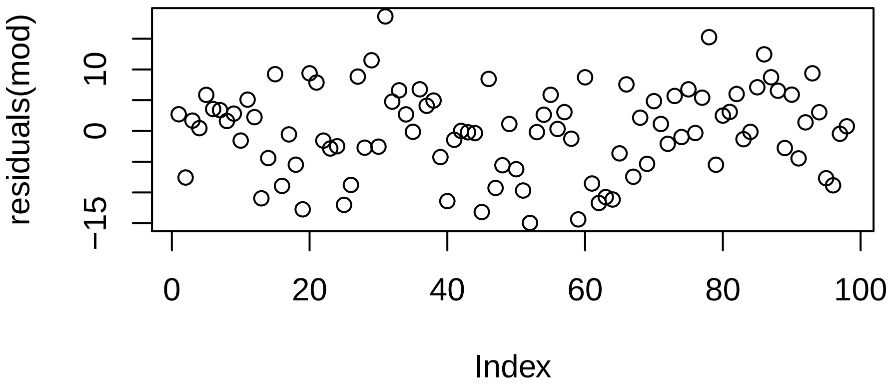
```
library(car)
data(Prestige)
mod <- lm(prestige~education+income+type, data=Prestige)
summary(mod)

##
## Call:
## lm(formula = prestige ~ education + income + type, data = Prestige)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -14.953  -4.449   0.168   5.057  18.632
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.622929   5.227525   -0.12     0.91
## education    3.673166   0.640502    5.73  1.2e-07 ***
```

[1] Basato su *Fox, J. and Weisberg, S. (2011). An R Companion to Applied Regression. Sage.*

```
## income        0.001013   0.000221    4.59  1.4e-05 ***
## typeprof      6.038971   3.866855    1.56     0.12
## typewc       -2.737231   2.513932   -1.09     0.28
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.09 on 93 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.835,Adjusted R-squared:  0.828
## F-statistic:  118 on 4 and 93 DF,  p-value: <2e-16
```

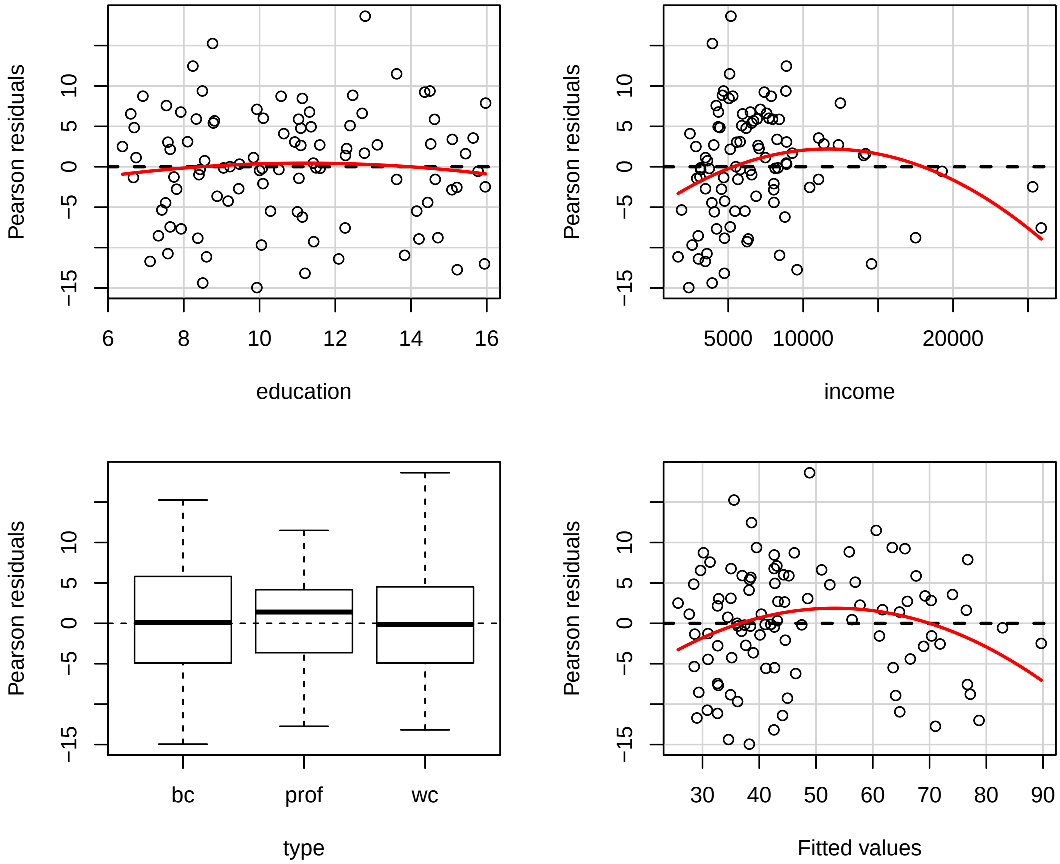
Iniziamo con un semplice grafico dei residui

```r
plot(residuals(mod))
```



Ora vediamo i grafici a dispersione dei residui rispetto ai predittori e ai valori stimati

```r
residualPlots(mod)
```



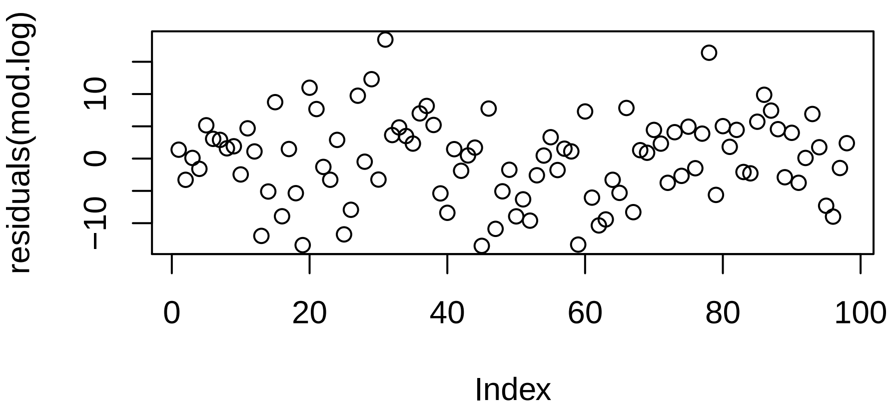

```
##              Test stat Pr(>|t|)
## education       -0.684    0.496
## income          -2.886    0.005
## type                NA       NA
## Tukey test      -2.610    0.009
```

Proviamo con una trasformazione logaritmica di `income` (perché?)

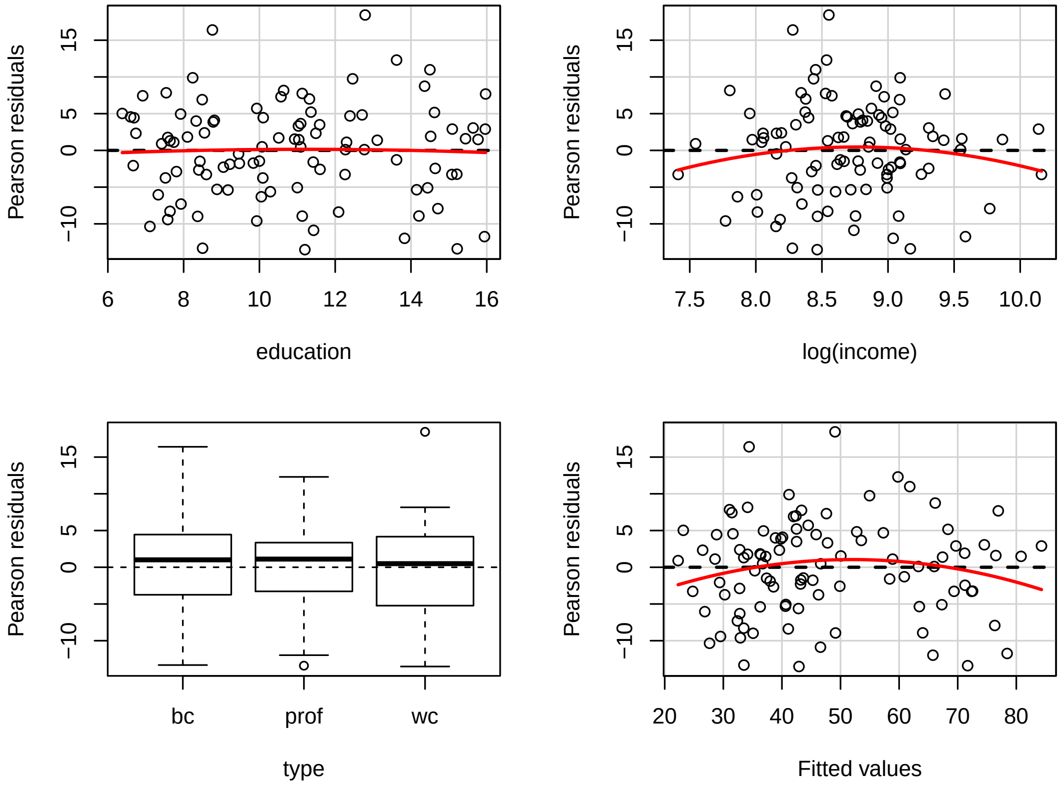
```
mod.log <- lm(prestige~education+log(income)+type, data=Prestige)
summary(mod.log)

##
## Call:
## lm(formula = prestige ~ education + log(income) + type, data = Prestige)
##
## Residuals:
```

```
##     Min      1Q Median      3Q     Max
## -13.51  -3.75   1.01   4.36  18.44
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -81.202     13.743   -5.91  5.6e-08 ***
## education      3.284      0.608    5.40  5.1e-07 ***
## log(income)   10.487      1.717    6.11  2.3e-08 ***
## typeprof       6.751      3.618    1.87    0.065 .
## typewc        -1.439      2.378   -0.61    0.546
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.64 on 93 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.855,Adjusted R-squared:  0.849
## F-statistic:  138 on 4 and 93 DF,  p-value: <2e-16
```

```r
plot(residuals(mod.log))
```



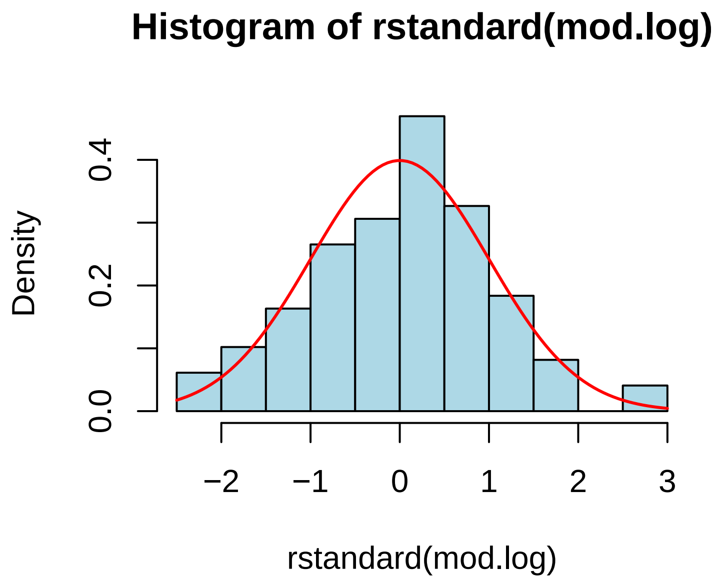

```r
residualPlots(mod.log)
```




```
##                  Test stat Pr(>|t|)
## education           -0.237    0.813
## log(income)         -1.044    0.299
## type                    NA       NA
## Tukey test          -1.446    0.148
```

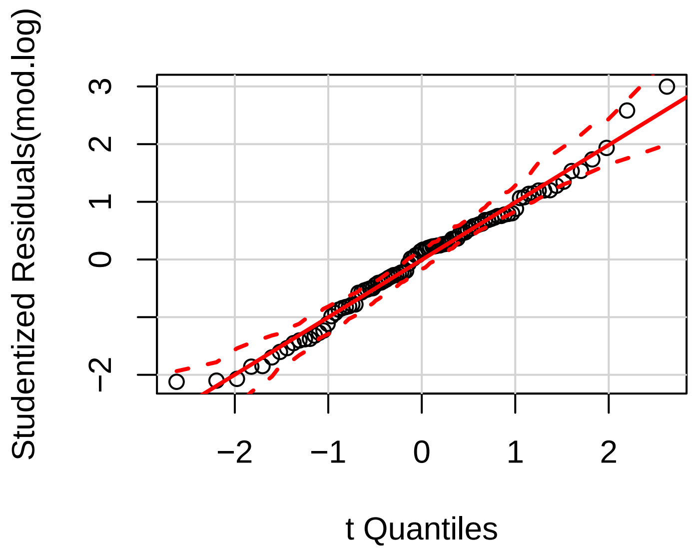
Per valutare l'assunzione di normalità degli errori, disegniamo l'istogramma dei residui standardizzati

```
hist( rstandard(mod.log), col="lightblue", freq=FALSE )
curve( dnorm(x), col="red", lwd=1.5, add=TRUE )
```



Uno strumento grafico più accurato è per valutare la normalità è il grafico quantile-quantile

```
qqPlot( mod.log )
```



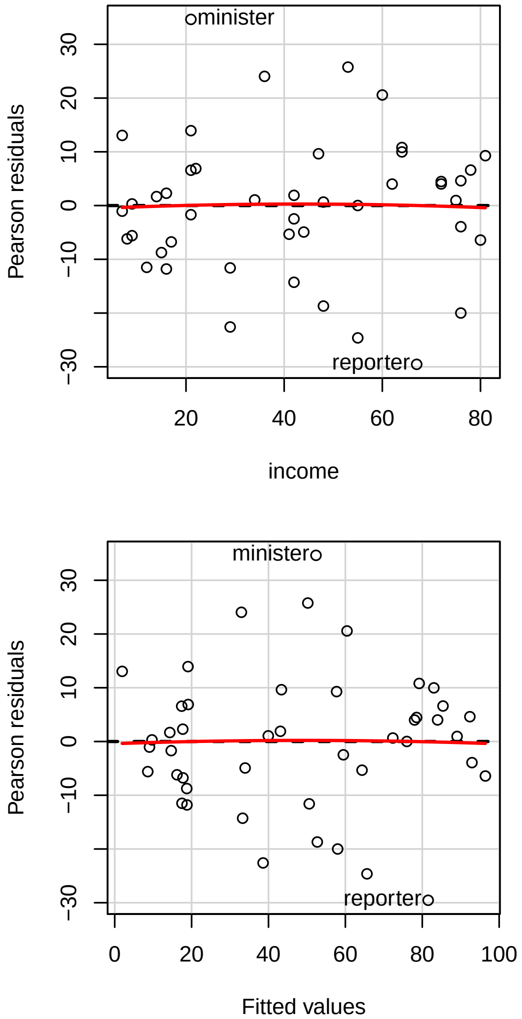
2 Outliers

Consideriamo il seguente modello con i dati `Duncan`

```
data(Duncan)
mod.duncan <- lm(prestige~income+education, data=Duncan)
summary(mod.duncan)

##
## Call:
## lm(formula = prestige ~ income + education, data = Duncan)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -29.54  -6.42   0.65   6.61  34.64
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -6.0647     4.2719   -1.42     0.16
## income        0.5987     0.1197    5.00  1.1e-05 ***
## education     0.5458     0.0983    5.56  1.7e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.4 on 42 degrees of freedom
## Multiple R-squared:  0.828,Adjusted R-squared:  0.82
## F-statistic:  101 on 2 and 42 DF,  p-value: <2e-16
```

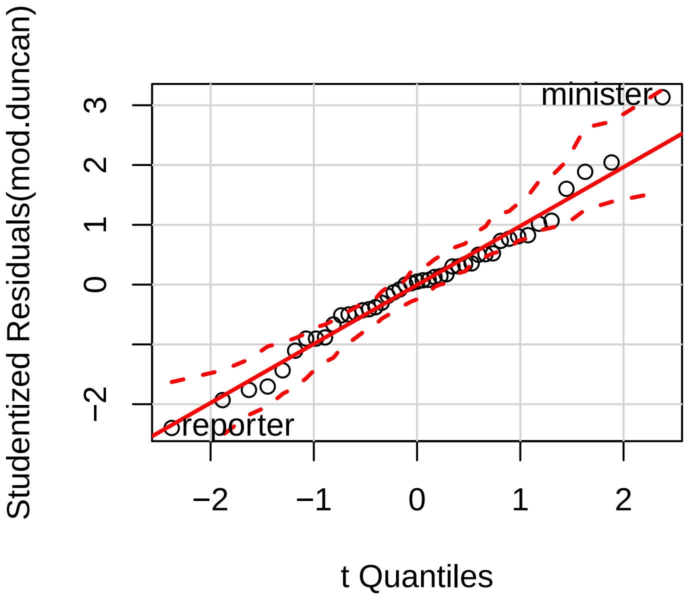
Controlliamo i residui

```
residualPlots(mod.duncan, id.n=2)
```




```
##              Test stat Pr(>|t|)
## income          -0.113    0.911
## education        0.672    0.505
## Tukey test      -0.081    0.935
```

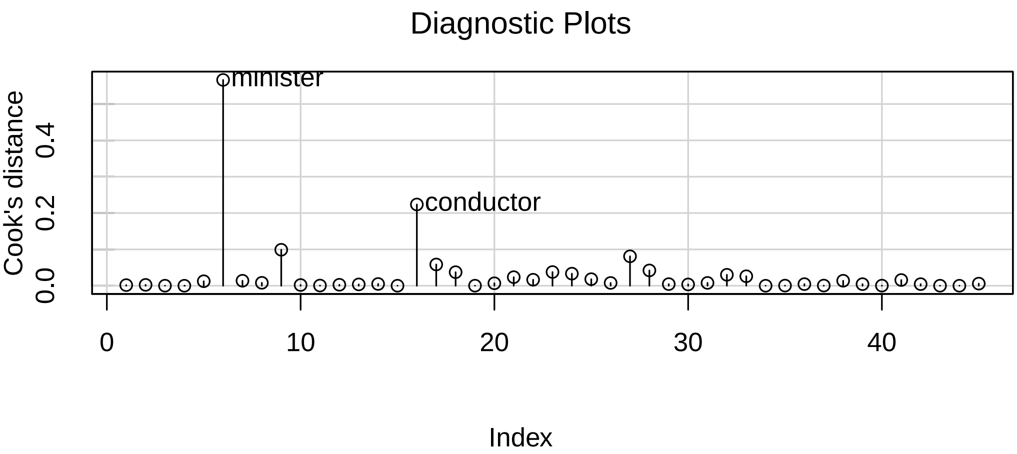
```
qqPlot(mod.duncan, id.n=2)
```




```
## reporter minister
##        1       45
```

Distanze di Cook

```
influenceIndexPlot(mod.duncan, vars="Cook", id.n=2, pch=21)
```

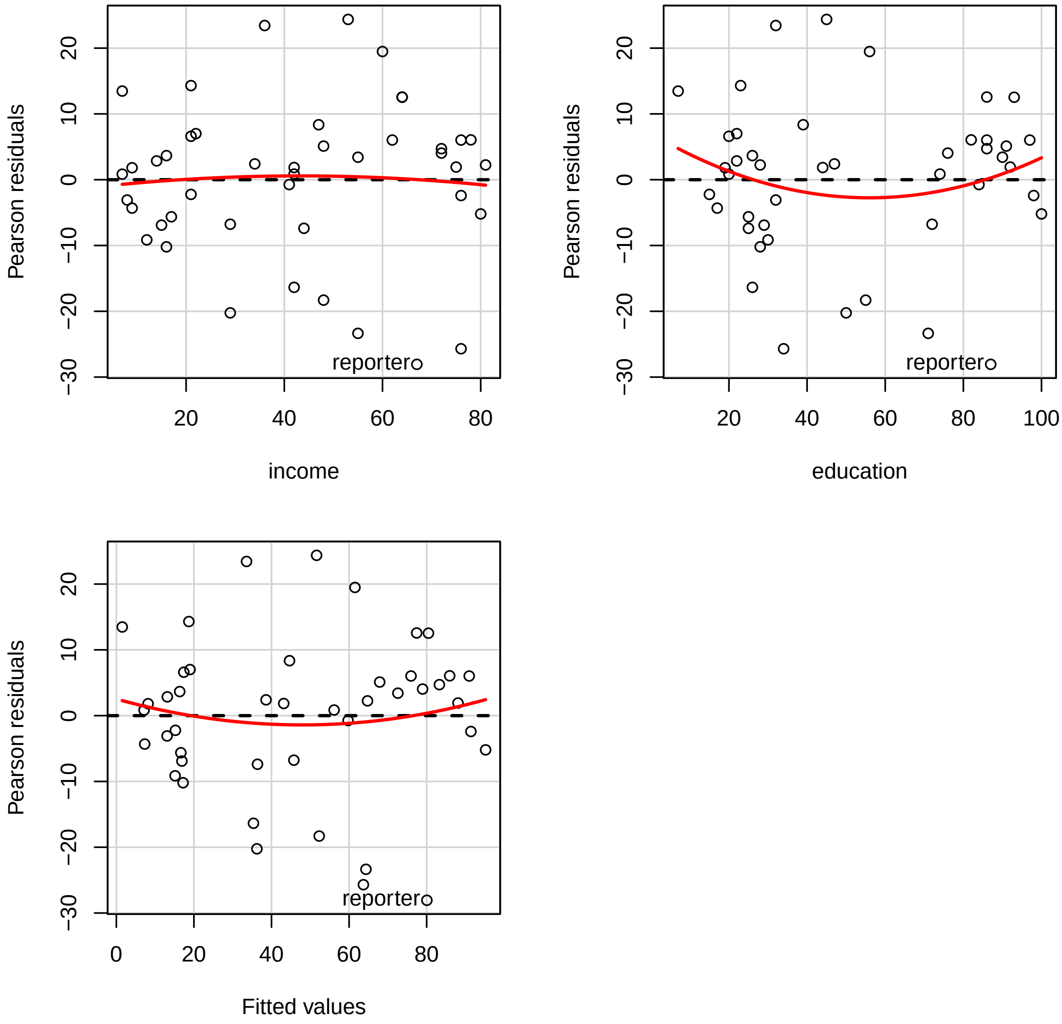


Ristimiamo il modello senza l'outlier `minister`

```
which( rownames(Duncan)=="minister" )

## [1] 6

mod.duncan2 <- lm(prestige~income+education, data=Duncan, subset=-6)
summary(mod.duncan2)

##
## Call:
## lm(formula = prestige ~ income + education, data = Duncan, subset = -6)
##
## Residuals:
##    Min      1Q Median      3Q     Max
## -28.06  -5.92   1.89   6.04  24.37
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -6.6275     3.8875   -1.70    0.096 .
## income        0.7316     0.1167    6.27  1.8e-07 ***
## education     0.4330     0.0963    4.50  5.6e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.2 on 41 degrees of freedom
## Multiple R-squared:  0.856,Adjusted R-squared:  0.849
## F-statistic:  122 on 2 and 41 DF,  p-value: <2e-16

compareCoefs(mod.duncan, mod.duncan2)

##
## Call:
## 1:"lm(formula = prestige ~ income + education, data = Duncan)"
## 2:"lm(formula = prestige ~ income + education, data = Duncan, subset = -6)"
##              Est. 1    SE 1  Est. 2    SE 2
## (Intercept) -6.0647  4.2719 -6.6275  3.8875
## income       0.5987  0.1197  0.7316  0.1167
## education    0.5458  0.0983  0.4330  0.0963
```

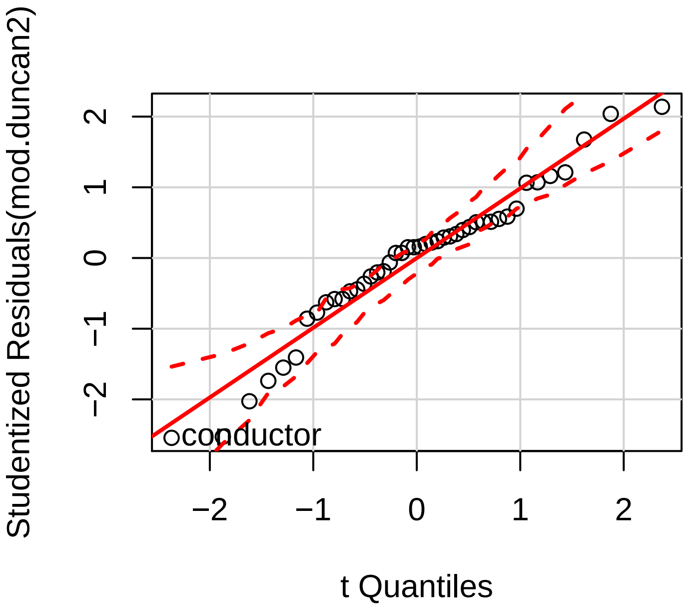
Controlliamo i nuovi residui

```
residualPlots(mod.duncan2, id.n=1)
```



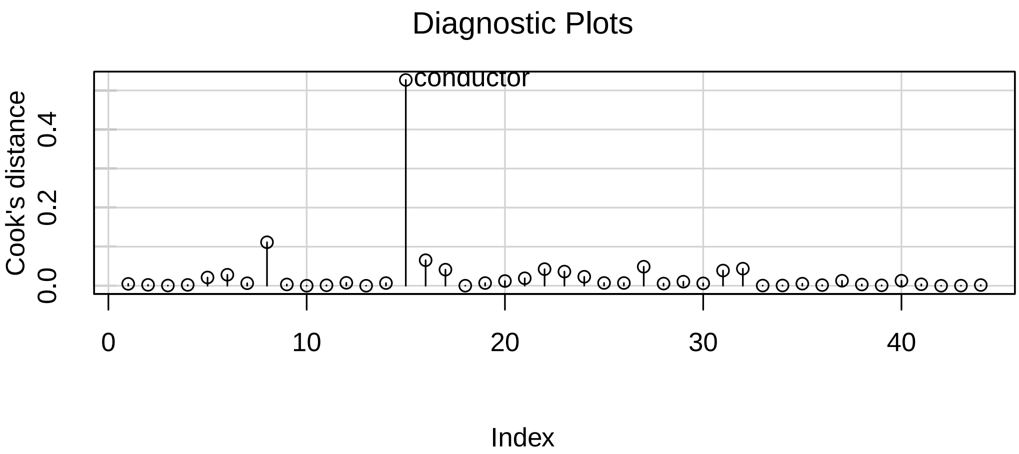

```
##                Test stat Pr(>|t|)
## income            -0.251    0.803
## education          0.952    0.347
## Tukey test         0.604    0.546
```

```
qqPlot(mod.duncan2, id.n=1)
```




```
## conductor
##         1
```

```
influenceIndexPlot(mod.duncan2, vars="Cook", id.n=1, pch=21)
```



Infine, guardiamo i test per la presenza di un outlier

```
outlierTest(mod.duncan)

##
## No Studentized residuals with Bonferonni p < 0.05
## Largest |rstudent|:
##          rstudent unadjusted p-value Bonferonni p
## minister    3.135           0.003177        0.143

outlierTest(mod.duncan2)

##
## No Studentized residuals with Bonferonni p < 0.05
## Largest |rstudent|:
##           rstudent unadjusted p-value Bonferonni p
## conductor   -2.543            0.01495       0.6577
```

Cosa si conclude?