# Esercizio 3 Modelli Statistici

## Potinga Marcelinio

## 2025-06-13

##Parte 1

**Importing the txt file**

```
data <-  read.csv("https://raw.githubusercontent.com/marcel0501/Esercizi-Mod-Stat/refs/heads/main/ANTRO
data$peso <- data$peso / 2.2 # Convert pounds to kg
```

**Stattistiche descrittive**

```
summary(data)
```

```
##      id_sogg           eta             peso            altez
##   Min.   :  1.00   Min.   :22.00   Min.   : 53.86   Min.   :162.6
##   1st Qu.: 65.75   1st Qu.:35.75   1st Qu.: 71.90   1st Qu.:173.4
##   Median :128.50   Median :43.00   Median : 80.06   Median :177.8
##   Mean   :127.74   Mean   :44.85   Mean   : 80.96   Mean   :178.6
##   3rd Qu.:190.25   3rd Qu.:54.00   3rd Qu.: 89.46   3rd Qu.:183.5
##   Max.   :252.00   Max.   :81.00   Max.   :119.43   Max.   :197.5
##      collo           torace           addom            anca
##   Min.   :31.10   Min.   : 79.30   Min.   : 69.40   Min.   : 85.00
##   1st Qu.:36.38   1st Qu.: 94.15   1st Qu.: 84.47   1st Qu.: 95.47
##   Median :38.00   Median : 99.60   Median : 90.95   Median : 99.30
##   Mean   :37.95   Mean   :100.67   Mean   : 92.31   Mean   : 99.66
##   3rd Qu.:39.42   3rd Qu.:105.30   3rd Qu.: 99.20   3rd Qu.:103.28
##   Max.   :43.90   Max.   :128.30   Max.   :126.20   Max.   :125.60
##      coscia          ginocch          caviglia         bicipite
##   Min.   :47.20   Min.   :33.00   Min.   :19.10   Min.   :24.80
##   1st Qu.:56.00   1st Qu.:36.90   1st Qu.:22.00   1st Qu.:30.20
##   Median :59.00   Median :38.45   Median :22.80   Median :32.00
##   Mean   :59.27   Mean   :38.54   Mean   :22.99   Mean   :32.22
##   3rd Qu.:62.30   3rd Qu.:39.90   3rd Qu.:24.00   3rd Qu.:34.33
##   Max.   :74.40   Max.   :46.00   Max.   :27.00   Max.   :39.10
##      avanbr           polso
##   Min.   :21.00   Min.   :15.80
##   1st Qu.:27.30   1st Qu.:17.60
##   Median :28.75   Median :18.30
##   Mean   :28.67   Mean   :18.22
##   3rd Qu.:30.00   3rd Qu.:18.80
##   Max.   :34.90   Max.   :21.40
```

**Modelli di regressione aventi X=bicipite e Y=peso con le specificazioni lineare-lineare, log-lineare, log-log, lineare log e quadratica.**

```
# Linear model
model_linear <- lm(peso ~ bicipite, data = data)
```

```r
# Log-linear model
model_log_linear <- lm(log(peso) ~ bicipite, data = data)
# Log-log model
model_log_log <- lm(log(peso) ~ log(bicipite), data = data)
# Linear-log model
model_linear_log <- lm(peso ~ log(bicipite), data = data)
# Quadratic model
model_quadratic <- lm(peso ~ bicipite + I(bicipite^2), data = data)
# Best model selection based on F-statistic
models <- list(
  linear = model_linear,
  log_linear = model_log_linear,
  log_log = model_log_log,
  linear_log = model_linear_log,
  quadratic = model_quadratic
)
best_model <- NULL
best_f_stat <- -Inf
for (model in models) {
  f_stat <- summary(model)$fstatistic[1]
  if (f_stat > best_f_stat) {
    best_f_stat <- f_stat
    best_model <- model
  }
}
# Display the best model
summary(best_model)
```

```
##
## Call:
## lm(formula = log(peso) ~ bicipite, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.39827 -0.05684  0.00154  0.06145  0.22972
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.069505   0.065864   46.60   <2e-16 ***
## bicipite    0.040756   0.002036   20.02   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09389 on 246 degrees of freedom
## Multiple R-squared:  0.6196, Adjusted R-squared:  0.618
## F-statistic: 400.7 on 1 and 246 DF,  p-value: < 2.2e-16
```
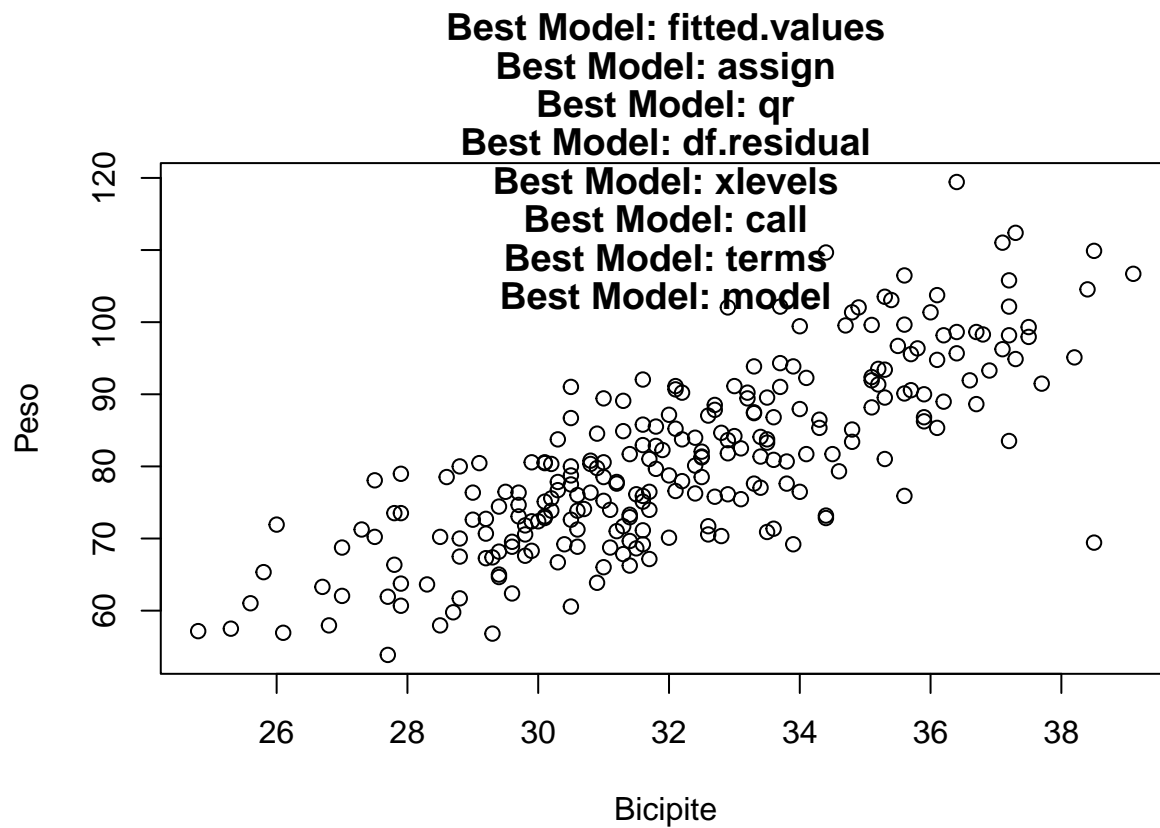
```r
# Plotting the best model
plot(data$bicipite, data$peso, main = paste("Best Model:", names(best_model)), xlab = "Bicipite", ylab =
abline(best_model, col = "red")
```
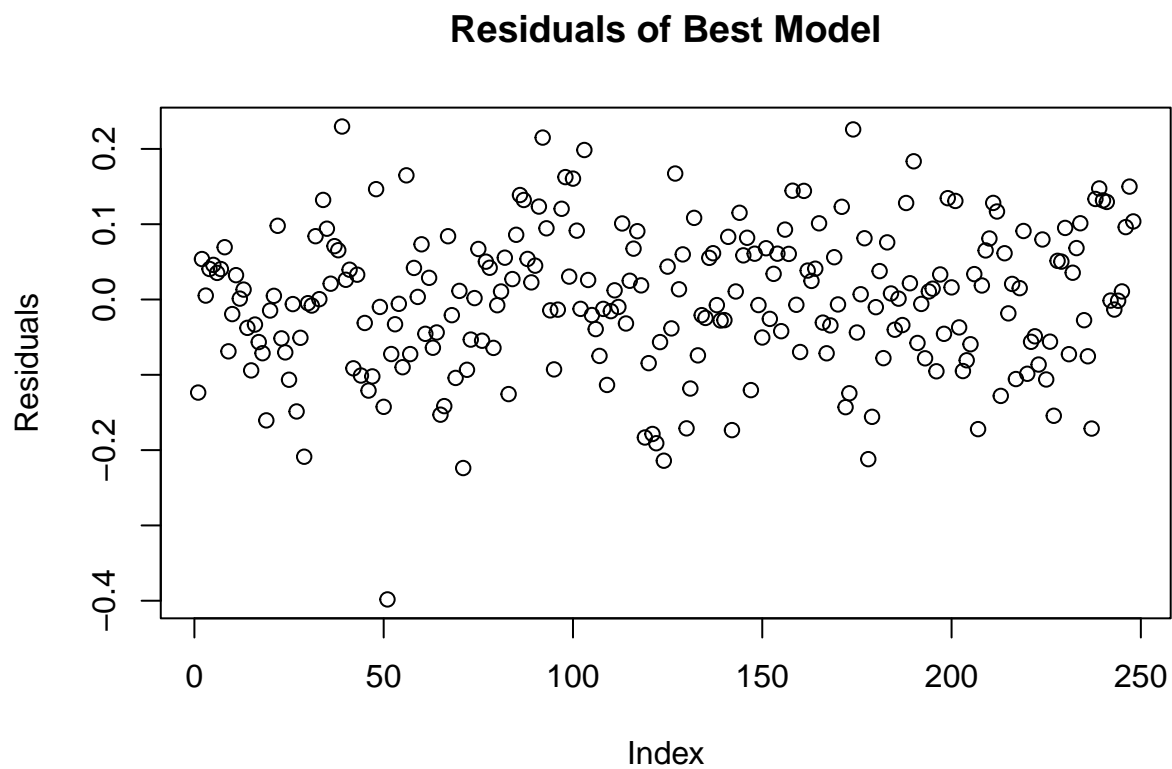
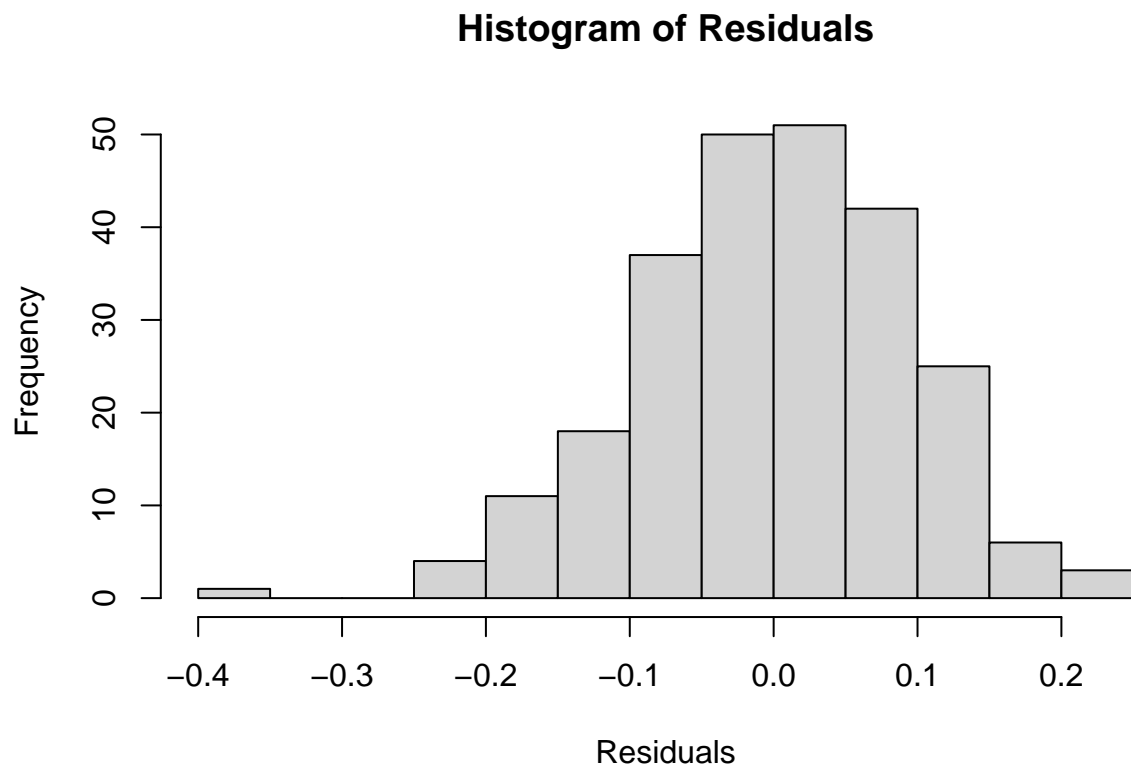**Best Model: fitted.values**
**Best Model: assign**
**Best Model: qr**
**Best Model: df.residual**
**Best Model: xlevels**
**Best Model: call**
**Best Model: terms**
**Best Model: model**



```r
# Residuals plot
plot(best_model$residuals, main = "Residuals of Best Model", ylab = "Residuals", xlab = "Index")
```
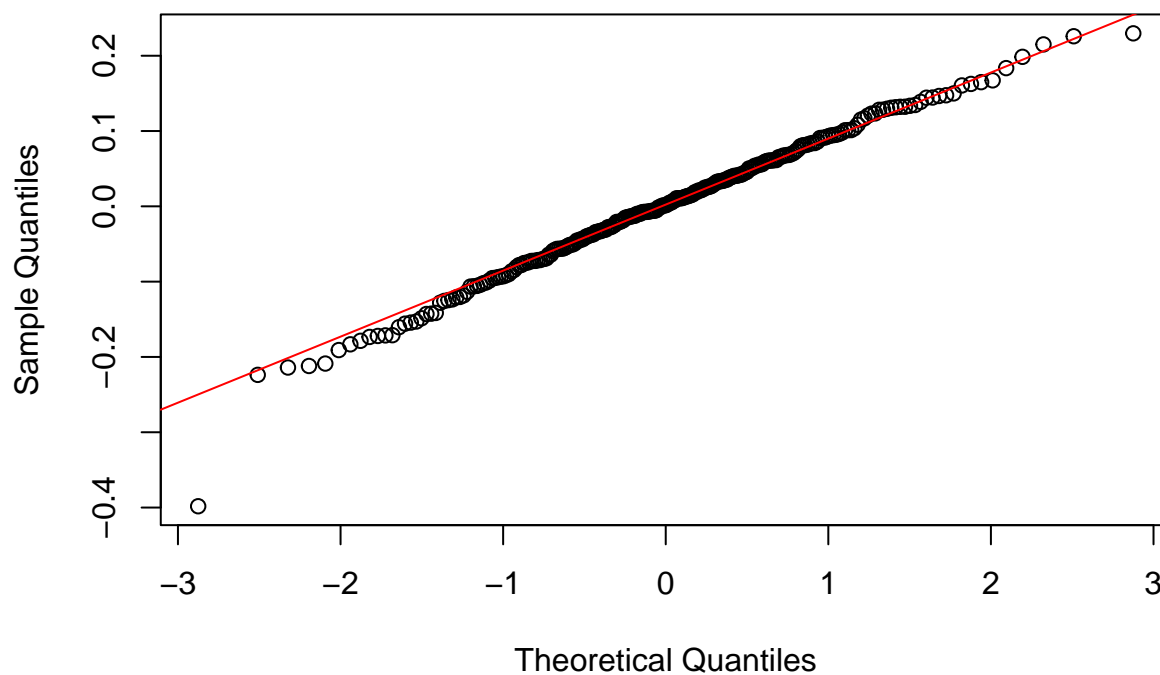
**Residuals of Best Model**

```r
# Histogram of residuals
hist(best_model$residuals, main = "Histogram of Residuals", xlab = "Residuals", breaks = 20)
```



**Histogram of Residuals**

```r
# QQ plot of residuals
qqnorm(best_model$residuals, main = "QQ Plot of Residuals")
qqline(best_model$residuals, col = "red")
```
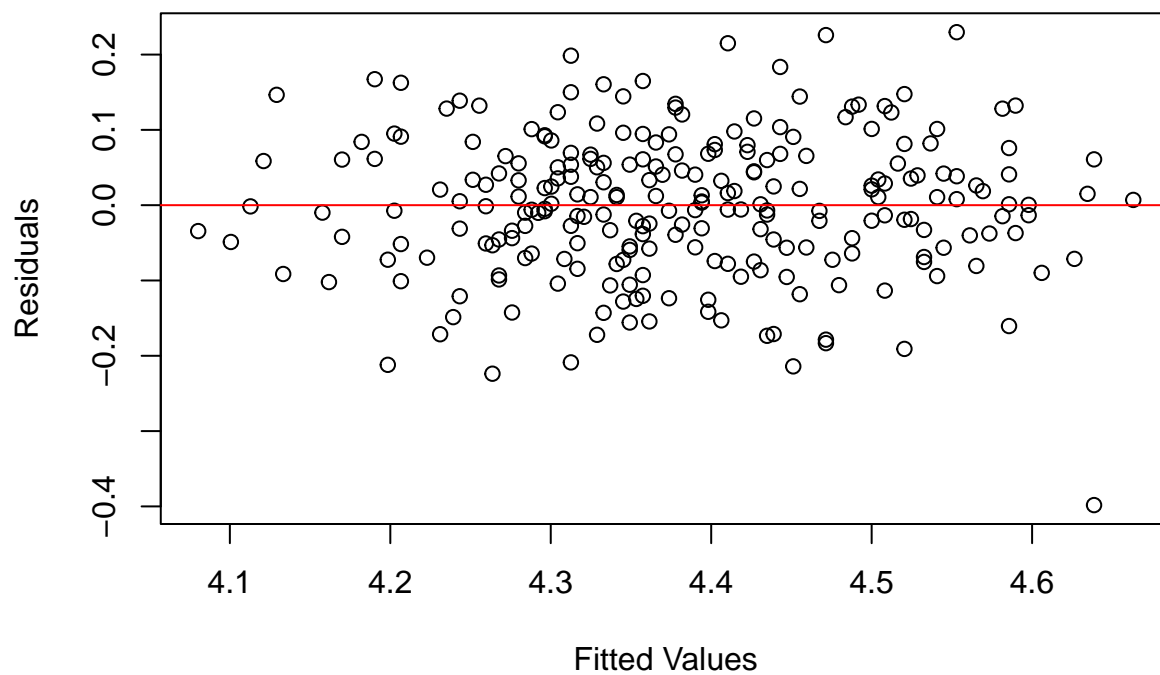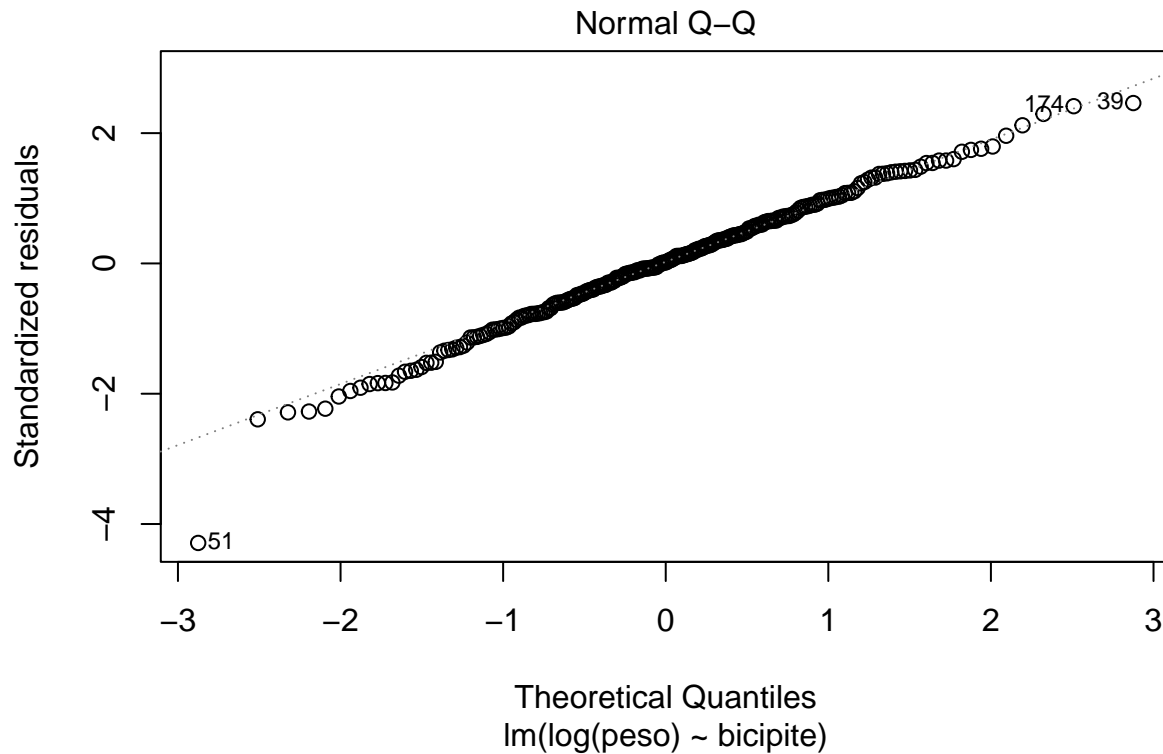
## QQ Plot of Residuals



```
#Residuals against fitted values
plot(best_model$fitted.values, best_model$residuals, main = "Residuals vs Fitted Values", xlab = "Fitte
abline(h = 0, col = "red")
```

## Residuals vs Fitted Values

```r
plot(best_model,which=2)
```



**Normal Q–Q**

Theoretical Quantiles
lm(log(peso) ~ bicipite)
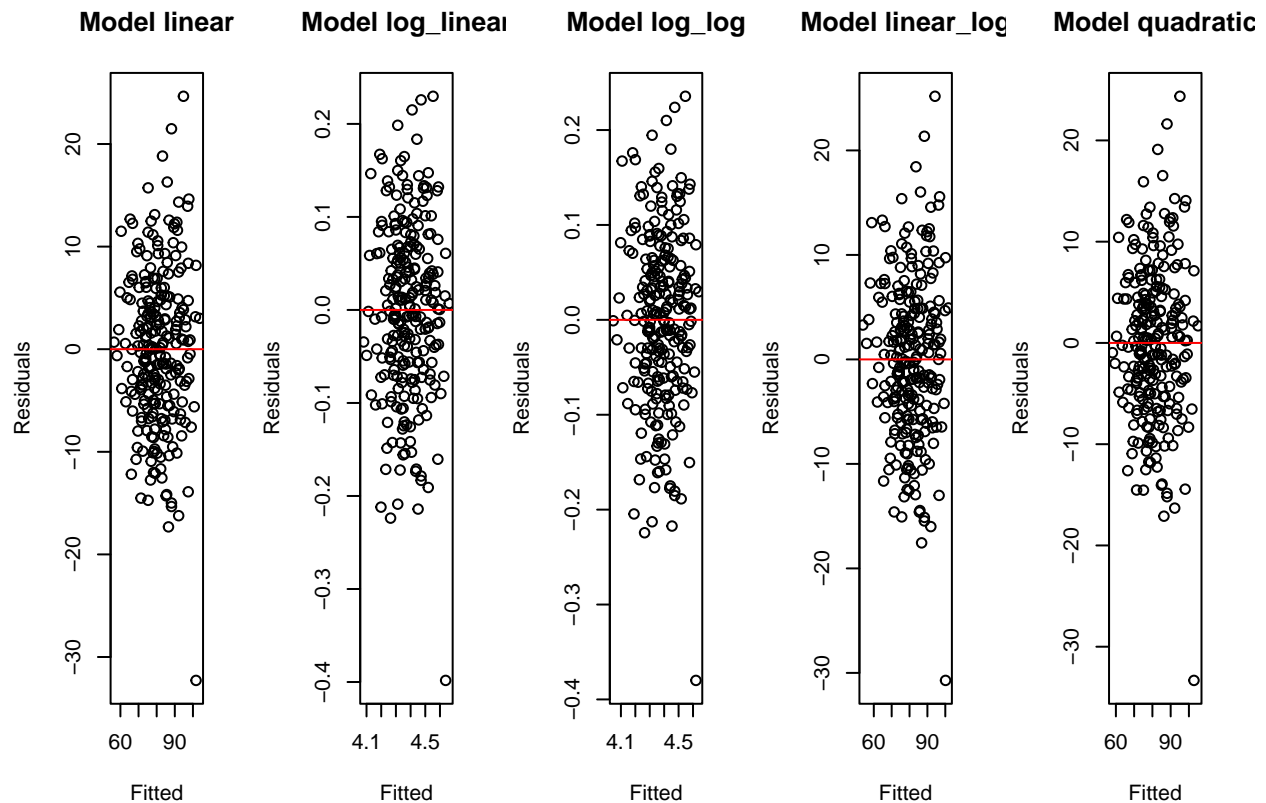
**Verifica eteroschedasticità per ogni modello**

```r
par(mfrow = c(1, length(models)))  # 1 row, N columns
# Plotting residuals vs fitted values for each model
for (i in seq_along(models)) {
  plot(fitted(models[[i]]), resid(models[[i]]),
       main = paste("Model", names(models)[i]),
       xlab = "Fitted", ylab = "Residuals")
  abline(h = 0, col = "red")
}
```
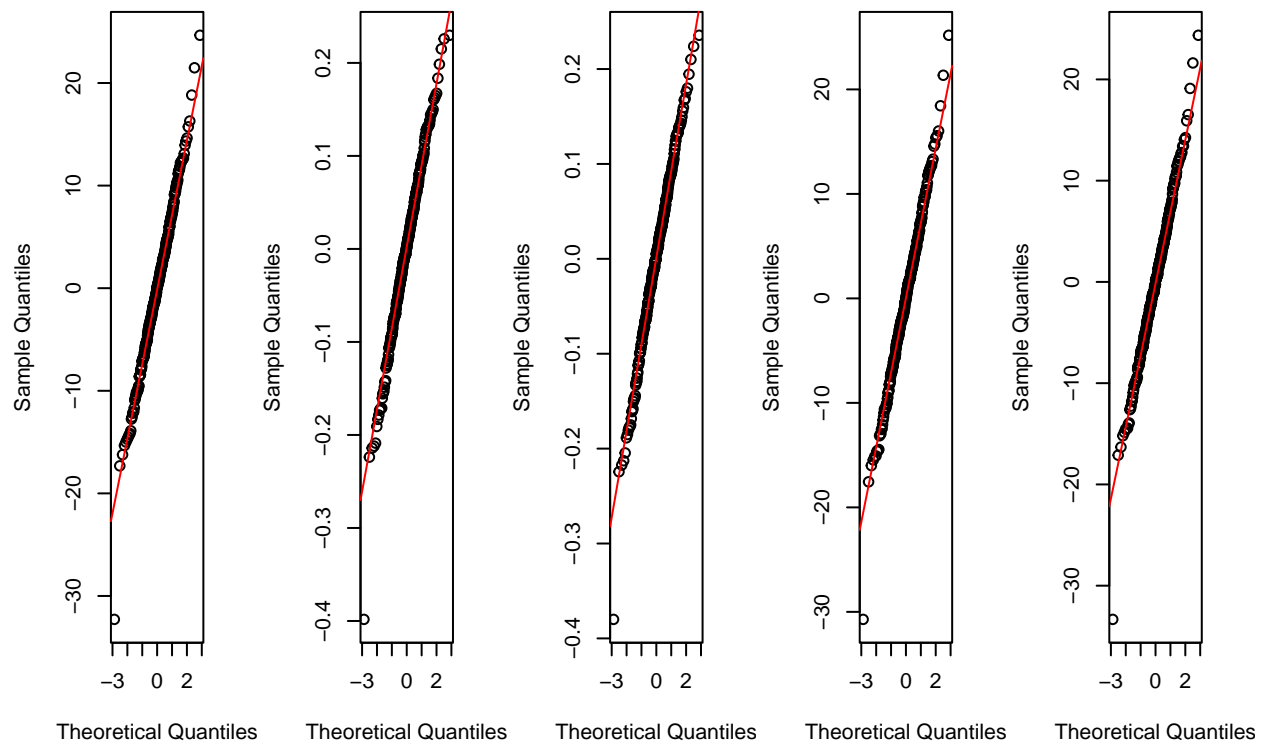
```r
# Resetting the plotting layout
par(mfrow = c(1, 1))
```

**Verifica normalità dei residui per ogni modello**

```r
par(mfrow = c(1, length(models)))  # 1 row, N columns
# QQ plot for each model
for (i in seq_along(models)) {
  qqnorm(resid(models[[i]]), main = paste("QQ Plot of Residuals -", names(models)[i]))
  qqline(resid(models[[i]]), col = "red")
}
```

**Q Plot of Residuals – lot of Residuals – kPlot of Residuals – lot of Residuals – liPlot of Residuals – q**

(Q-Q Plots of Residuals across five panels, each with "Sample Quantiles" on the y-axis and "Theoretical Quantiles" on the x-axis)

```
# Resetting the plotting layout
par(mfrow = c(1, 1))
```