

Lesson 07

Christopher A. Swenson (chris@csvenson.com)

11/27/2021

IQ and physical characteristics (confidence and prediction intervals)

Load the iqsize data. Fit a multiple linear regression model of PIQ on Brain and Height. Calculate a 95% confidence interval for mean PIQ at Brain=90, Height=70. Calculate a 95% confidence interval for mean PIQ at Brain=79, Height=62. Calculate a 95% prediction interval for individual PIQ at Brain=90, Height=70.

```
iqsize <- read.table("./Data/iqsize.txt", header=T)
attach(iqsize)
```

```
model <- lm(PIQ ~ Brain + Height)
```

```
predict(model, interval="confidence", se.fit=T,
        newdata=data.frame(Brain=90, Height=70))
```

```
## $fit
##      fit      lwr      upr
## 1 105.6391 98.23722 113.041
##
## $se.fit
## [1] 3.646064
##
## $df
## [1] 35
##
## $residual.scale
## [1] 19.50957
```

```
# $fit
#      fit      lwr      upr
# 1 105.6391 98.23722 113.041
#
# $se.fit
# [1] 3.646064
```

```
predict(model, interval="confidence", se.fit=T,
        newdata=data.frame(Brain=79, Height=62))
```

```
## $fit
##      fit      lwr      upr
## 1 104.8114 91.41796 118.2049
##
## $se.fit
## [1] 6.597407
##
```

```
## $df
## [1] 35
##
## $residual.scale
## [1] 19.50957

# $fit
#      fit      lwr      upr
# 1 104.8114 91.41796 118.2049
#
# $se.fit
# [1] 6.597407

predict(model, interval="prediction",
        newdata=data.frame(Brain=90, Height=70))

##      fit      lwr      upr
## 1 105.6391 65.34688 145.9314

# $fit
#      fit      lwr      upr
# 1 105.6391 65.34688 145.9314

detach(iqsize)
```

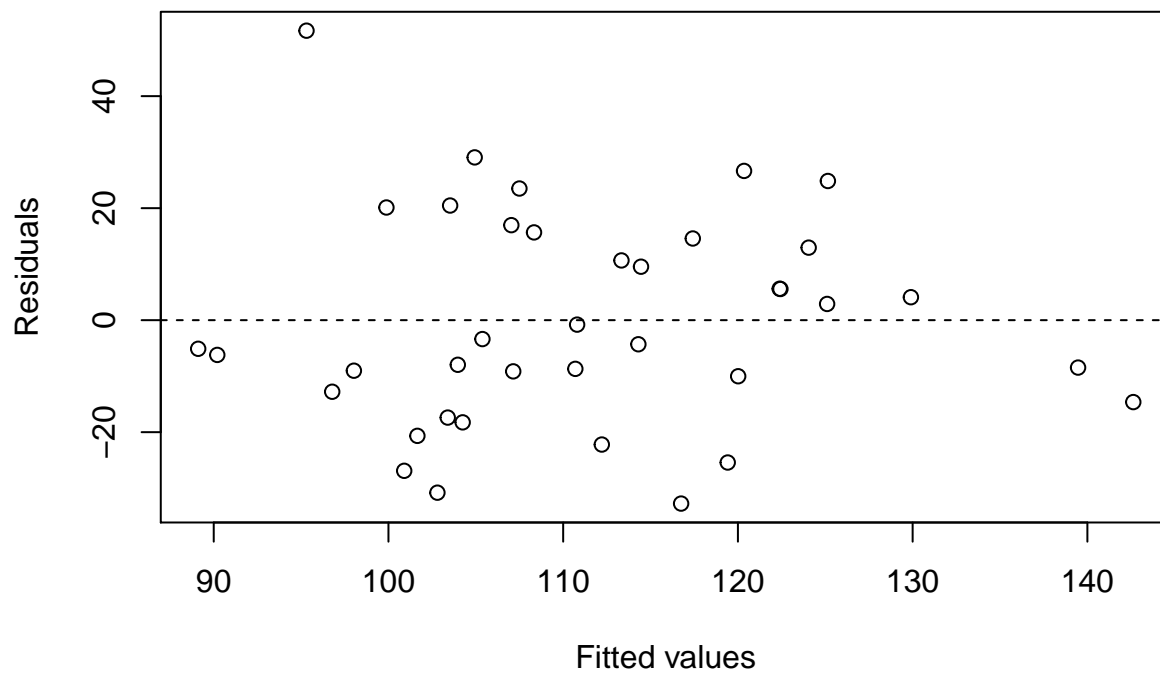
IQ and physical characteristics (residual plots and normality tests)

Load the iqsize data. Fit a multiple linear regression model of PIQ on Brain and Height. Display residual plot with fitted (predicted) values on the horizontal axis. Display residual plot with Brain on the horizontal axis. Display residual plot with Height on the horizontal axis. Display histogram of the residuals. Display normal probability plot of the residuals and add a diagonal line to the plot. The argument “datax” determines which way round to plot the axes (false by default, which plots the data on the vertical axis, or true, which plots the data on the horizontal axis). Display residual plot with Weight on the horizontal axis. Load the nortest package to access normality tests: Anderson-Darling Shapiro-Wilk (Ryan-Joiner unavailable in R) Lilliefors (Kolmogorov-Smirnov)

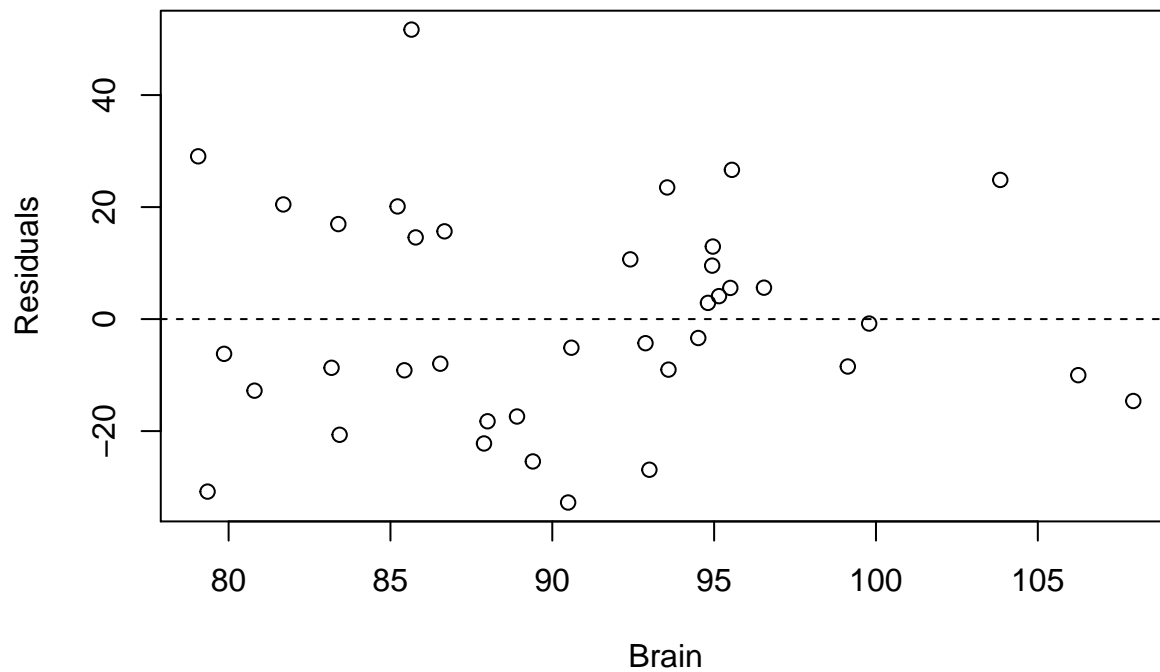
```
iqsize <- read.table("./Data/iqsize.txt", header=T)
attach(iqsize)

model <- lm(PIQ ~ Brain + Height)

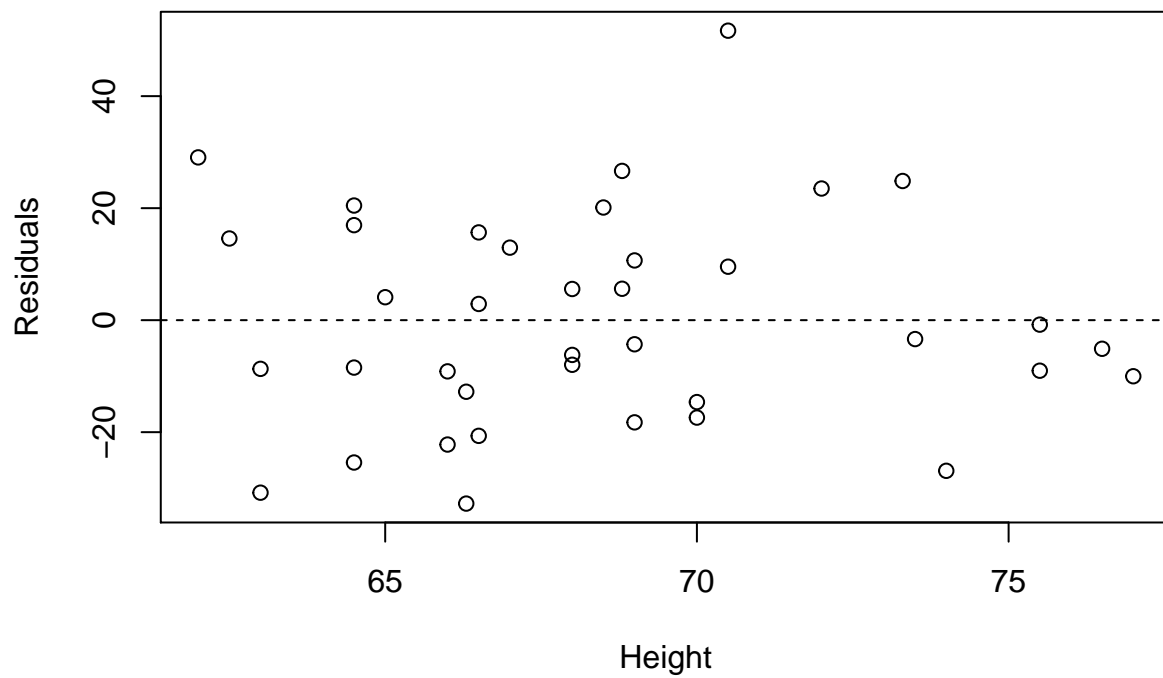
plot(x=fitted(model), y=residuals(model),
     xlab="Fitted values", ylab="Residuals",
     panel.last = abline(h=0, lty=2))
```



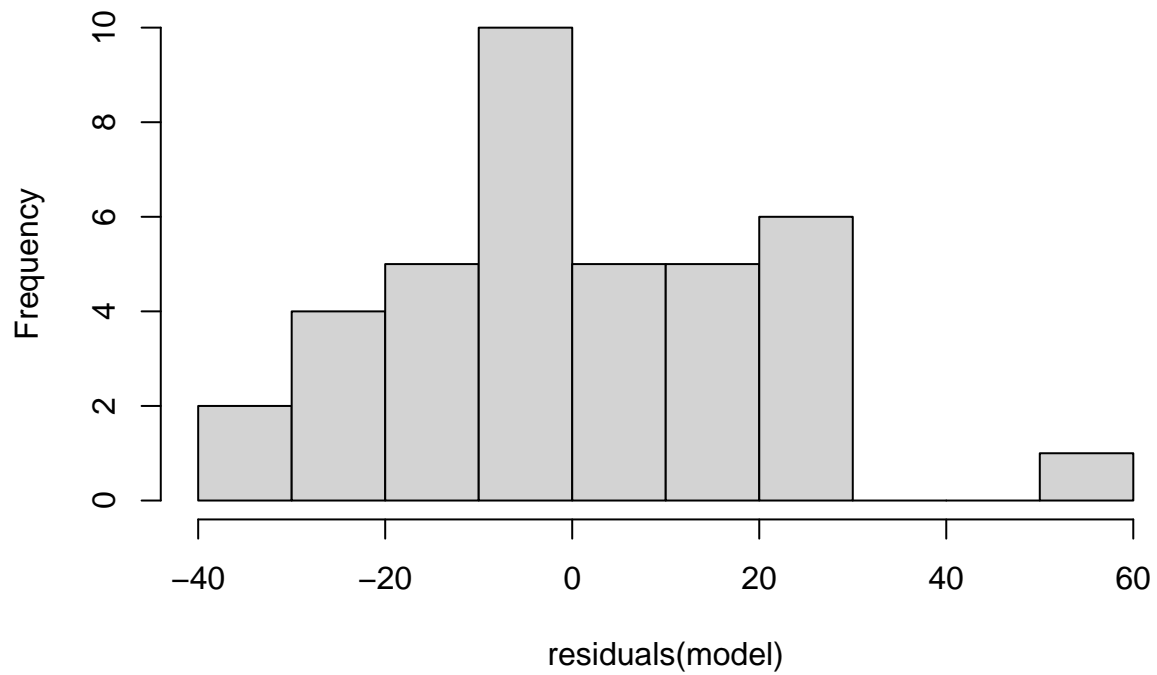
```
plot(x=Brain, y=residuals(model),
     ylab="Residuals",
     panel.last = abline(h=0, lty=2))
```



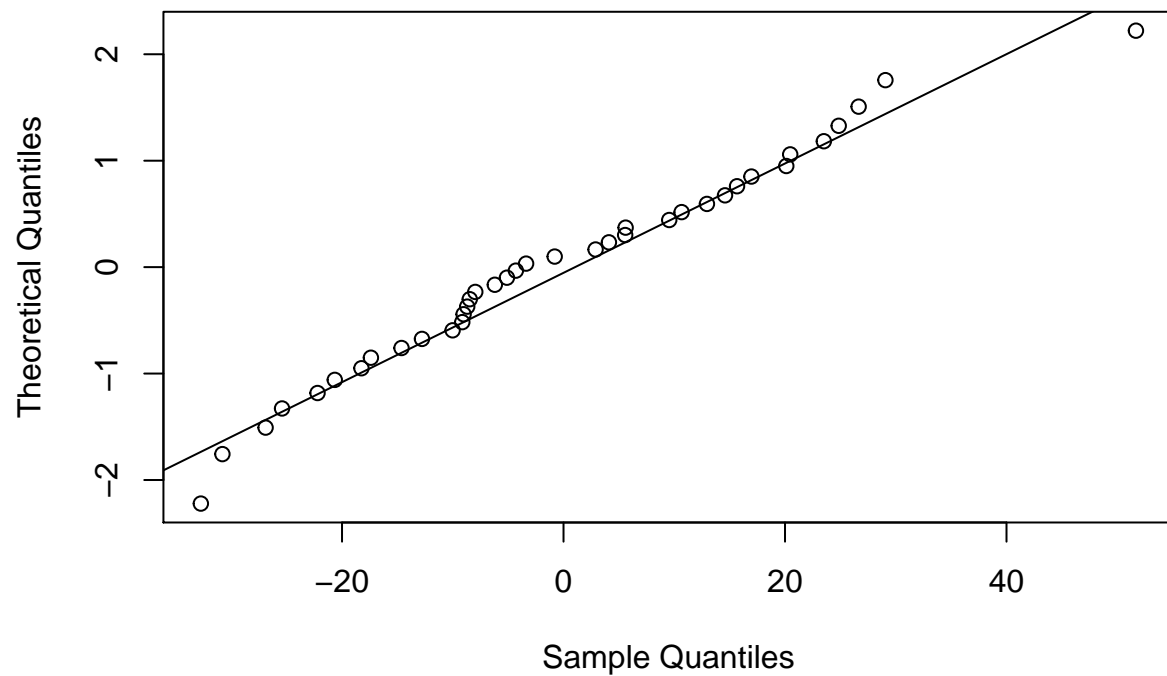
```
plot(x=Height, y=residuals(model),
     ylab="Residuals",
     panel.last = abline(h=0, lty=2))
```



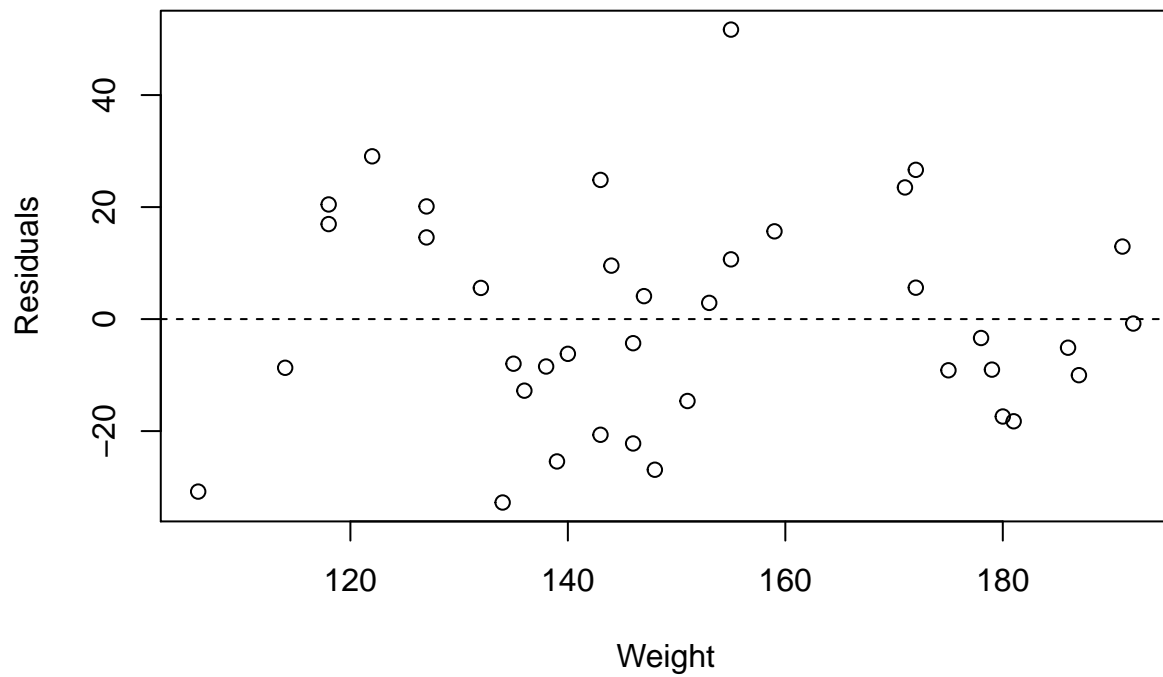
```
hist(residuals(model), main="")
```



```
qqnorm(residuals(model), main="", datax=TRUE)
qqline(residuals(model), datax=TRUE)
```



```
plot(x=Weight, y=residuals(model),
     ylab="Residuals",
     panel.last = abline(h=0, lty=2))
```



```
library(nortest)
ad.test(residuals(model)) # A = 0.2621, p-value = 0.6857
```

```
##
## Anderson-Darling normality test
##
## data: residuals(model)
```

```
## A = 0.26206, p-value = 0.6857
shapiro.test(residuals(model)) # W = 0.976, p-value = 0.5764

##
## Shapiro-Wilk normality test
##
## data: residuals(model)
## W = 0.976, p-value = 0.5764
lillie.test(residuals(model)) # D = 0.097, p-value = 0.4897

##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: residuals(model)
## D = 0.096962, p-value = 0.4897
detach(iqsize)
```

Toluca refrigerator parts (tests for constant error variance)

Load the Toluca data. Fit a simple linear regression model of WorkHours on LotSize. Create lotgroup factor variable splitting the sample into two groups. Load the car package to access tests for constant error variance: Modified Levene (Brown-Forsythe) Cook-Weisberg score (Breusch-Pagan)

```
toluca <- read.table("./Data/toluca.txt", header=T)
attach(toluca)

model <- lm(WorkHours ~ LotSize)
lotgroup <- factor(LotSize<=70)

library(car)

## Loading required package: carData
leveneTest(residuals(model), group=lotgroup)

## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value Pr(>F)
## group 1  1.7331  0.201
##      23
# Levene's Test for Homogeneity of Variance (center = median)
#      Df F value Pr(>F)
# group 1  1.7331  0.201
#      23

ncvTest(model)

## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.8209192, Df = 1, p = 0.36491
# Non-constant Variance Score Test
# Variance formula: ~ fitted.values
# Chisquare = 0.8209192 Df = 1 p = 0.3649116

detach(toluca)
```