Second\_Assignment.R

ASUS

2022-01-09

###### SECOND ASSIGNMENTS  
  
## Andrea Scalenghe - 913359  
## Matteo Morabito - 918551  
  
#First of all we extract the dataset from the UsingR libray  
  
library(tidyverse)

## Warning: il pacchetto 'tidyverse' è stato creato con R versione 4.1.2

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5 v purrr 0.3.4  
## v tibble 3.1.6 v dplyr 1.0.7  
## v tidyr 1.1.4 v stringr 1.4.0  
## v readr 2.1.1 v forcats 0.5.1

## Warning: il pacchetto 'tibble' è stato creato con R versione 4.1.2

## Warning: il pacchetto 'readr' è stato creato con R versione 4.1.2

## Warning: il pacchetto 'forcats' è stato creato con R versione 4.1.2

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(UsingR)

## Warning: il pacchetto 'UsingR' è stato creato con R versione 4.1.2

## Caricamento del pacchetto richiesto: MASS

##   
## Caricamento pacchetto: 'MASS'

## Il seguente oggetto è mascherato da 'package:dplyr':  
##   
## select

## Caricamento del pacchetto richiesto: HistData

## Warning: il pacchetto 'HistData' è stato creato con R versione 4.1.2

## Caricamento del pacchetto richiesto: Hmisc

## Warning: il pacchetto 'Hmisc' è stato creato con R versione 4.1.2

## Caricamento del pacchetto richiesto: lattice

## Caricamento del pacchetto richiesto: survival

## Caricamento del pacchetto richiesto: Formula

##   
## Caricamento pacchetto: 'Hmisc'

## I seguenti oggetti sono mascherati da 'package:dplyr':  
##   
## src, summarize

## I seguenti oggetti sono mascherati da 'package:base':  
##   
## format.pval, units

##   
## Caricamento pacchetto: 'UsingR'

## Il seguente oggetto è mascherato da 'package:survival':  
##   
## cancer

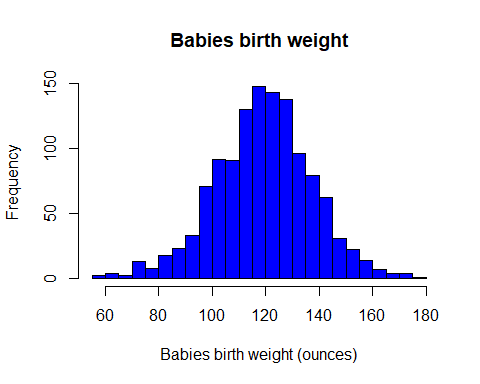
data(babies)  
attach(babies)  
str(babies)

## 'data.frame': 1236 obs. of 23 variables:  
## $ id : num 15 20 58 61 72 100 102 129 142 148 ...  
## $ pluralty : num 5 5 5 5 5 5 5 5 5 5 ...  
## $ outcome : num 1 1 1 1 1 1 1 1 1 1 ...  
## $ date : num 1411 1499 1576 1504 1425 ...  
## $ gestation: num 284 282 279 999 282 286 244 245 289 299 ...  
## $ sex : num 1 1 1 1 1 1 1 1 1 1 ...  
## $ wt : num 120 113 128 123 108 136 138 132 120 143 ...  
## $ parity : num 1 2 1 2 1 4 4 2 3 3 ...  
## $ race : num 8 0 0 0 0 0 7 7 0 0 ...  
## $ age : num 27 33 28 36 23 25 33 23 25 30 ...  
## $ ed : num 5 5 2 5 5 2 2 1 4 5 ...  
## $ ht : num 62 64 64 69 67 62 62 65 62 66 ...  
## $ wt1 : num 100 135 115 190 125 93 178 140 125 136 ...  
## $ drace : num 8 0 5 3 0 3 7 7 3 0 ...  
## $ dage : num 31 38 32 43 24 28 37 23 26 34 ...  
## $ ded : num 5 5 1 4 5 2 4 4 1 5 ...  
## $ dht : num 65 70 99 68 99 64 99 71 70 99 ...  
## $ dwt : num 110 148 999 197 999 130 999 192 180 999 ...  
## $ marital : num 1 1 1 1 1 1 1 1 0 1 ...  
## $ inc : num 1 4 2 8 1 4 98 2 2 2 ...  
## $ smoke : num 0 0 1 3 1 2 0 0 0 1 ...  
## $ time : num 0 0 1 5 1 2 0 0 0 1 ...  
## $ number : num 0 0 1 5 5 2 0 0 0 4 ...

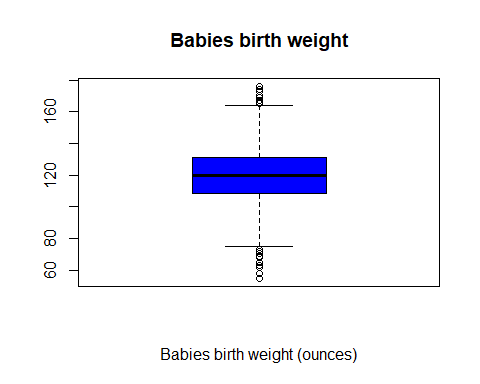
dim(babies)

## [1] 1236 23

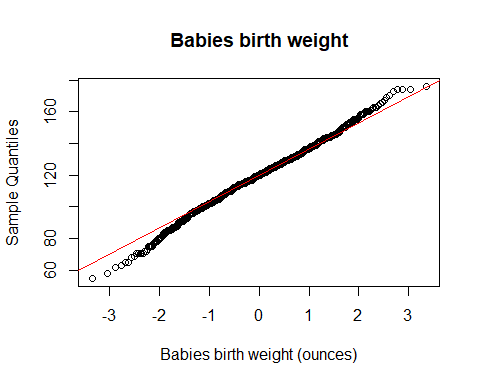
#Exercise 1  
  
#At first we plot some useful graphs for a complete descriptive analysis of wt  
  
hist(wt, breaks = 20, col="blue", main="Babies birth weight", xlab = "Babies birth weight (ounces)")



#We can immediately see that the data distributes similarly to a normal whit mean 120  
  
boxplot(wt, col="blue", main="Babies birth weight", xlab = "Babies birth weight (ounces)")



#Also the boxplot graph confirm what we saw on the histogram. In the end by   
#plotting the data in the Q-Q plot we can see that:  
  
qqnorm(wt, main="Babies birth weight", xlab = "Babies birth weight (ounces)")  
qqline(wt, col="red")



#This graphs states that, a part from the very end of the tails, data are  
#distribute like a normal distribution  
  
summary(wt)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 55.0 108.8 120.0 119.6 131.0 176.0

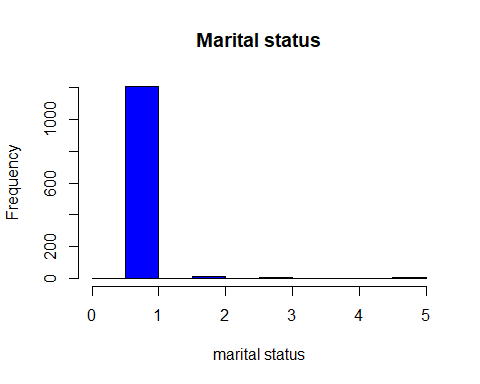
#The mean, as predicted, is around 120 ounces (as matter of facts 119.6 ounces).  
#The distribution of quantiles tells us that the first and third are equally   
#far from the mean. Considering the end of the tails the distribution has a slightly  
#bigger left tail.   
#We now test the normality  
  
shapiro.test(wt)

##   
## Shapiro-Wilk normality test  
##   
## data: wt  
## W = 0.99559, p-value = 0.001192

#The p-value is <0.05, which means that we refuse the normality. Although the shapiro  
#test gives more likely false results for large size samples, so we must take   
#in consideration the fact that the test is biased  
length(wt)

## [1] 1236

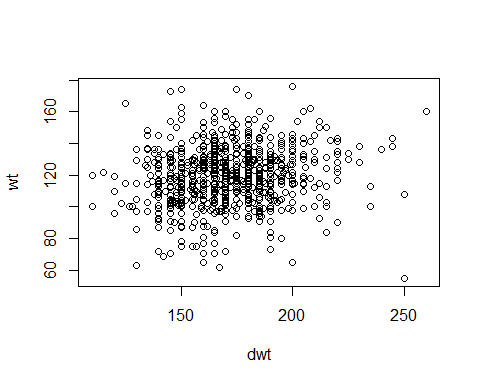
#For what concerns marital  
  
hist(marital, col="blue", main="Marital status", xlab = "marital status")



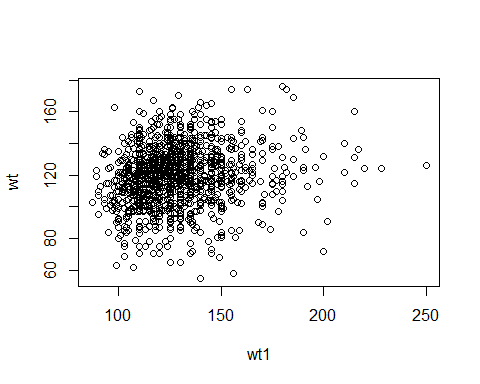
#It's clear from the histogram the almost the totality of marital status are 1:  
#most mother with babies are married  
  
summary(marital)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.000 1.000 1.000 1.038 1.000 5.000

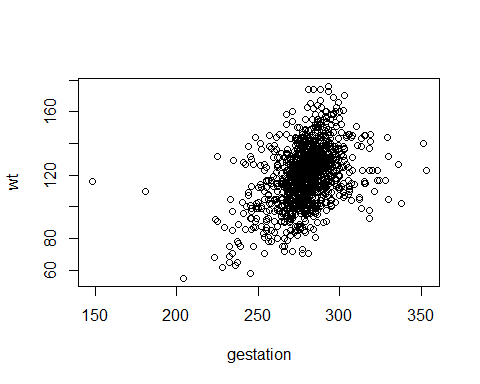
#In fact the summary tells us that the median, the mean and the first and third quantile   
#are all 1. It means that, as said, almost all the values are 1.  
  
  
#Exercise 2  
  
#We aim to find the best estimator for the babies weight. In order to do so we  
#compute a correlation test and look at the p-value. The two hypotheses are:  
#H0 : The factors are uncorrelated  
#H1 : The factors are correlated  
  
dwt[dwt==999]<-NA  
wt1[wt1==999]<-NA  
gestation[gestation==999]<-NA  
  
plot(dwt, wt)



plot(wt1, wt)



plot(gestation, wt)



#Linear correlation seems light in the first 2 plots, the stronger   
#one seems to be in the third  
  
cor.test(dwt,wt)

##   
## Pearson's product-moment correlation  
##   
## data: dwt and wt  
## t = 3.938, df = 735, p-value = 8.999e-05  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.07227869 0.21374478  
## sample estimates:  
## cor   
## 0.143746

cor.test(wt1,wt)

##   
## Pearson's product-moment correlation  
##   
## data: wt1 and wt  
## t = 5.396, df = 1198, p-value = 8.207e-08  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.09830554 0.20880768  
## sample estimates:  
## cor   
## 0.1540382

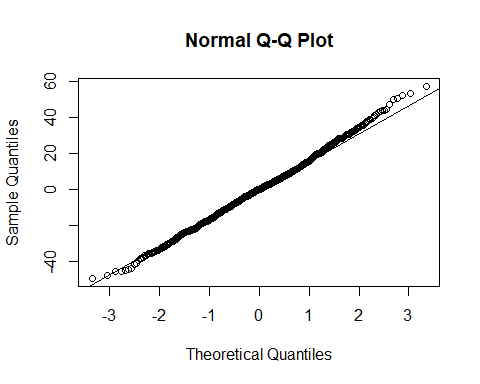
cor.test(gestation,wt)

##   
## Pearson's product-moment correlation  
##   
## data: gestation and wt  
## t = 15.609, df = 1221, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.3600303 0.4535398  
## sample estimates:  
## cor   
## 0.407854

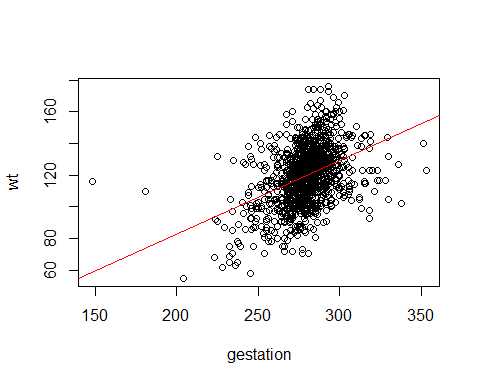
#The results obtained are:  
#p-value for dwt-wt correlation: 8.999e-05  
#p-value for wt1-wt correlation: 8.207e-08  
#p-value for gestation-wt correlation: 2.2e-16  
#Furthermore the higher cor values is the last one, let's check the linear  
#regression for wt and gestation  
  
  
model <- lm(wt~gestation)  
summary(model)

##   
## Call:  
## lm(formula = wt ~ gestation)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -49.394 -11.125 0.071 10.106 57.353   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -10.06418 8.32220 -1.209 0.227   
## gestation 0.46426 0.02974 15.609 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 16.66 on 1221 degrees of freedom  
## (13 osservazioni eliminate a causa di valori mancanti)  
## Multiple R-squared: 0.1663, Adjusted R-squared: 0.1657   
## F-statistic: 243.6 on 1 and 1221 DF, p-value: < 2.2e-16

residuals <- residuals(model)  
qqnorm(residuals)  
qqline(residuals)



#The model is good and residuals distributes almost perfectly as a normal  
  
plot(gestation, wt)  
abline(model, col = "red")



#We can state that the length of the gestation is the best estimator for babies   
#weight.  
  
#In order to perform a multivariate analysis, as taught in the article linked,  
#we use the library "Tidyverse" and the function "lm".  
  
model <- lm(wt~dwt+wt1+gestation)  
summary(model)

##   
## Call:  
## lm(formula = wt ~ dwt + wt1 + gestation)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -48.928 -10.381 -0.212 10.388 55.919   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -34.15495 12.06953 -2.830 0.004789 \*\*   
## dwt 0.09352 0.02807 3.331 0.000909 \*\*\*  
## wt1 0.10151 0.03071 3.305 0.000996 \*\*\*  
## gestation 0.44619 0.03911 11.408 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 16.56 on 708 degrees of freedom  
## (524 osservazioni eliminate a causa di valori mancanti)  
## Multiple R-squared: 0.188, Adjusted R-squared: 0.1845   
## F-statistic: 54.62 on 3 and 708 DF, p-value: < 2.2e-16

#The result is clear: p-value and F-statistic are not at all significant. Furthermore   
#the R-squared is far from 1, which means that the portion of variance explained  
#in the outcome is little.  
#We see also that the best predictor is confirmed to be the gestation.   
  
  
#Exercise 3  
  
parity1<-parity  
table(parity1)

## parity1  
## 0 1 2 3 4 5 6 7 8 9 10 11 13   
## 315 310 238 168 83 52 32 16 8 7 4 2 1

parity1[parity1>2]<-3  
as.factor(parity1)

## [1] 1 2 1 2 1 3 3 2 3 3 2 3 3 3 3 3 3 3 2 3 3 1 1 3 2 3 1 2 3 2 2 3 2 1 3 1 1  
## [38] 2 1 3 2 3 3 3 3 3 3 2 1 3 3 3 1 3 2 3 2 3 2 1 1 1 2 1 3 3 3 2 3 3 1 2 1 3  
## [75] 3 2 3 2 1 3 3 2 2 3 1 2 3 3 1 1 3 2 2 2 3 1 1 2 2 3 2 3 1 3 1 3 2 2 1 3 1  
## [112] 3 1 2 1 1 3 2 3 1 3 1 3 3 2 2 3 3 2 3 2 2 1 1 2 3 1 1 2 2 3 3 2 3 3 1 3 3  
## [149] 1 3 2 2 3 1 2 2 3 1 1 1 1 3 3 2 1 1 1 3 1 1 1 3 1 1 3 3 3 1 3 2 1 3 2 3 1  
## [186] 3 1 1 1 3 3 1 3 3 3 3 3 3 2 3 2 3 1 1 1 1 3 2 3 1 2 3 2 3 1 3 3 3 3 1 2 1  
## [223] 1 2 1 3 3 2 3 2 1 3 1 1 3 3 3 1 1 3 3 2 3 2 1 1 2 3 3 3 3 2 1 1 3 3 1 1 3  
## [260] 1 3 3 3 1 1 1 1 3 3 3 2 0 3 0 3 2 0 3 1 0 1 0 2 0 0 0 0 3 0 2 0 2 0 0 0 2  
## [297] 1 3 0 2 0 1 2 0 1 0 2 0 1 1 2 3 3 1 2 3 3 3 1 1 0 3 0 1 1 0 3 1 0 1 1 2 0  
## [334] 0 0 1 0 2 1 3 0 1 2 1 2 3 3 2 0 1 3 2 1 0 0 2 3 2 2 1 0 3 3 1 1 0 3 3 3 3  
## [371] 0 1 1 3 1 3 0 1 1 0 1 0 3 1 1 3 0 3 1 0 1 0 0 1 0 1 0 0 2 0 2 1 1 0 0 0 3  
## [408] 2 0 3 1 3 0 1 0 2 3 1 2 0 3 0 0 1 2 2 3 0 3 0 1 0 2 2 3 2 1 3 1 3 3 0 3 2  
## [445] 1 2 0 0 2 0 0 0 2 2 2 2 0 2 1 3 0 0 1 0 3 0 2 0 0 0 3 2 0 3 0 0 3 0 3 0 2  
## [482] 3 3 3 1 2 0 1 0 1 1 2 0 0 2 0 1 0 2 3 2 2 2 0 0 3 0 2 0 2 3 0 3 2 0 1 3 1  
## [519] 1 1 1 1 1 1 2 3 0 2 0 1 2 2 3 3 1 2 0 1 0 0 0 3 2 3 0 0 2 1 0 0 3 3 1 1 0  
## [556] 0 3 3 0 0 3 3 0 3 3 1 0 3 1 2 2 2 0 0 1 1 3 3 3 1 1 0 2 0 1 0 3 0 1 2 3 1  
## [593] 3 1 0 1 0 0 0 0 3 0 0 3 1 2 0 0 0 2 0 3 2 0 0 1 2 3 2 0 2 1 3 3 2 1 1 0 3  
## [630] 0 0 0 0 3 0 1 3 0 2 3 3 3 3 0 1 3 1 3 3 0 3 1 0 2 3 1 1 1 2 1 1 3 3 3 0 2  
## [667] 3 0 3 0 3 3 2 1 3 3 3 3 1 1 1 3 3 0 3 0 2 2 0 1 2 0 1 3 2 3 3 1 1 0 0 1 0  
## [704] 3 1 0 2 0 0 2 2 0 1 1 1 2 0 0 2 1 0 0 1 3 2 3 1 1 1 3 0 3 1 3 3 2 0 3 0 1  
## [741] 0 2 0 0 1 0 0 2 2 1 1 3 1 3 3 1 0 3 1 0 3 1 3 3 1 0 3 2 3 0 3 2 0 1 1 1 1  
## [778] 1 0 2 1 0 2 0 1 2 3 1 2 0 3 3 0 0 2 2 0 1 0 2 2 2 0 3 3 3 0 0 3 2 1 1 2 1  
## [815] 1 3 3 1 0 2 2 2 3 3 0 0 3 0 0 1 1 0 3 2 1 1 3 0 0 0 2 0 3 0 1 0 3 2 2 3 0  
## [852] 1 2 3 3 2 3 3 3 3 0 0 3 0 2 1 0 2 1 3 0 3 0 3 1 0 0 0 2 0 0 0 0 0 1 3 0 1  
## [889] 0 2 0 1 0 1 3 1 0 2 0 1 0 1 0 3 3 0 3 2 2 0 0 0 3 0 0 1 3 2 0 0 3 1 2 3 1  
## [926] 0 3 3 0 0 0 3 0 0 2 3 1 0 3 0 1 0 2 3 0 3 1 0 1 0 2 2 1 1 3 0 3 2 1 0 1 2  
## [963] 3 1 3 3 2 2 0 3 0 3 0 0 0 2 0 2 3 0 2 0 3 3 3 0 0 3 2 1 2 0 0 0 3 3 3 0 1  
## [1000] 0 1 3 1 1 0 0 0 2 2 2 0 2 1 1 2 0 3 1 3 1 3 2 3 3 1 1 1 0 2 3 1 0 1 2 2 3  
## [1037] 3 1 0 1 1 2 0 3 2 0 2 3 0 3 0 0 0 0 2 3 1 1 2 0 3 3 2 1 0 1 0 2 3 0 0 3 3  
## [1074] 0 2 1 1 3 0 0 0 1 3 0 3 3 3 0 0 0 1 0 3 0 3 3 1 2 0 3 3 1 0 3 0 1 2 0 2 3  
## [1111] 1 1 3 2 0 1 1 3 2 2 1 2 3 1 3 0 0 3 3 3 2 1 3 0 1 1 1 0 0 0 1 2 0 1 3 3 2  
## [1148] 3 0 3 2 3 1 0 1 3 1 3 1 1 2 1 3 2 0 0 3 3 3 2 2 0 1 1 0 1 3 2 0 1 2 3 1 1  
## [1185] 0 1 1 3 0 0 2 3 0 1 3 1 2 0 3 2 0 1 3 0 3 1 1 0 3 0 3 1 0 3 1 3 3 2 2 2 0  
## [1222] 0 0 0 3 0 0 2 1 2 3 0 1 3 0 3  
## Levels: 0 1 2 3

table(parity1)

## parity1  
## 0 1 2 3   
## 315 310 238 373

wta <- wt[parity==0]  
wtb <- wt[parity==1]  
wtc <- wt[parity==2]  
wtd <- wt[parity>2]  
summary(wta)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 63.0 107.0 118.0 118.1 128.0 176.0

summary(wtb)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 62.0 109.0 121.0 119.3 132.0 165.0

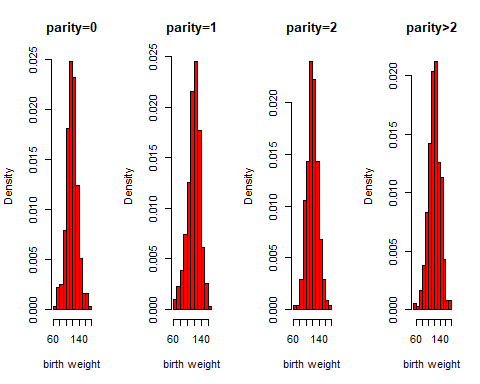
summary(wtc)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 69.0 110.0 120.0 120.0 130.8 174.0

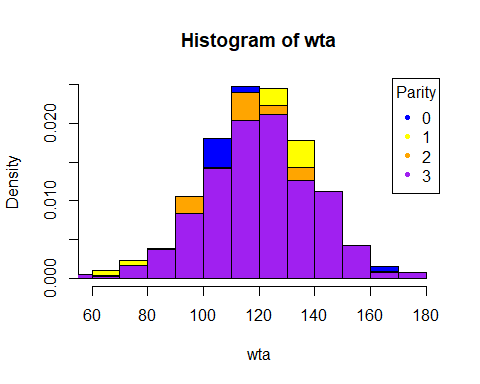
summary(wtd)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 55.0 108.0 121.0 120.8 133.0 174.0

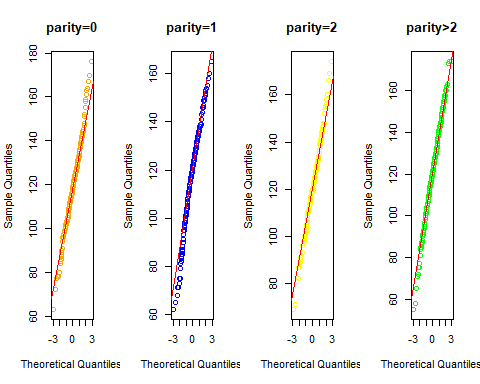
#From the summaries it is clear that the mean birth weight grows as parity grows  
#Although we can also see that the minumum decreases as the parity grows. This  
#means that the birth weight spreads as the parity grows.  
  
#Let's compare graphically   
par(mfrow=c(1,4))  
hist(wta, breaks=10, freq=F, col="red", main="parity=0", xlab = "birth weight" )  
hist(wtb, breaks=10, freq=F, col="red", main="parity=1", xlab = "birth weight" )  
hist(wtc, breaks=10, freq=F, col="red", main="parity=2", xlab = "birth weight" )  
hist(wtd, breaks=10, freq=F, col="red", main="parity>2", xlab = "birth weight" )



par(mfrow=c(1,1))  
hist(wta, freq=F, col="blue")  
hist(wtb, freq=F, col="yellow", add=T)  
hist(wtc, freq=F, col="orange", add=T)  
hist(wtd, freq=F, col="purple", add=T)  
legend(x="topright", legend = c(0,1,2,3), title = "Parity", pch=20,   
 col = c("blue","yellow","orange","purple"))



#We can see that all of the distribution are similar, although as the parity   
#grows the distribution spreads  
  
par(mfrow=c(1,4))  
qqnorm(wta, col="orange", main = "parity=0")  
qqline(wta, col="red")  
qqnorm(wtb, col="blue", main = "parity=1")  
qqline(wtb, col="red")  
qqnorm(wtc, col="yellow", main = "parity=2")  
qqline(wtc, col="red")  
qqnorm(wtd, col="green", main = "parity>2")  
qqline(wtd, col="red")



#From this graphs the samples seem to distribute as a normal.   
#We test the normality by the shapiro test   
  
shapiro.test(wta)

##   
## Shapiro-Wilk normality test  
##   
## data: wta  
## W = 0.99091, p-value = 0.04878

shapiro.test(wtb)

##   
## Shapiro-Wilk normality test  
##   
## data: wtb  
## W = 0.98009, p-value = 0.000265

shapiro.test(wtc)

##   
## Shapiro-Wilk normality test  
##   
## data: wtc  
## W = 0.99458, p-value = 0.5564

shapiro.test(wtd)

##   
## Shapiro-Wilk normality test  
##   
## data: wtd  
## W = 0.99574, p-value = 0.4101

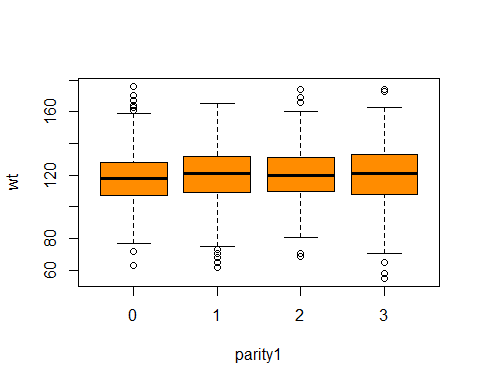
#p-values are not homogeneous and wt2 and wt3 result not normally distributed. We must keep in mind the the test is biased by the large size of the sample  
#Let's check the variances  
  
l <- list(wta,wtb,wtc,wtd)  
bartlett.test(l)

##   
## Bartlett test of homogeneity of variances  
##   
## data: l  
## Bartlett's K-squared = 8.8542, df = 3, p-value = 0.03129

#Variances aren't equal, therefore we can't perform the ANOVA test. Let's perform the Kruskal-Wallis test insted  
  
kruskal.test(wt~parity1)

##   
## Kruskal-Wallis rank sum test  
##   
## data: wt by parity1  
## Kruskal-Wallis chi-squared = 5.2752, df = 3, p-value = 0.1527

#The null hypotheses cannot not be refused: the medians of the three groups cannot be considered different at a 95% significance level  
  
par(mfrow=c(1,1))  
boxplot(wt~parity1, col = "darkorange")



#Exercise 4  
  
#Let's calculate the probability of each marital status:  
P0 <- sum(marital==0)/length(marital)  
P1 <- sum(marital==1)/length(marital)  
P2 <- sum(marital==2)/length(marital)  
P3 <- sum(marital==3)/length(marital)  
P5 <- sum(marital==5)/length(marital)  
  
#Then we simulate the samples   
  
n1 <- 1000  
n2 <- 10000  
n3 <- 100000  
n4 <- 1000000  
  
S1 <- sample(c(0,1,2,3,5), n1, replace = T, prob = c(P0, P1, P2, P3, P5))  
S2 <- sample(c(0,1,2,3,5), n2, replace = T, prob = c(P0, P1, P2, P3, P5))  
S3 <- sample(c(0,1,2,3,5), n3, replace = T, prob = c(P0, P1, P2, P3, P5))  
S4 <- sample(c(0,1,2,3,5), n4, replace = T, prob = c(P0, P1, P2, P3, P5))  
  
#We calculate now the relative frequency   
  
ftable1 <- table(S1)/n1  
ftable2 <- table(S2)/n2  
ftable3 <- table(S3)/n3  
ftable4 <- table(S4)/n4  
ftable <- table(marital)/length(marital)  
cbind(ftable1, ftable2, ftable3, ftable4, ftable)

## ftable1 ftable2 ftable3 ftable4 ftable  
## 0 0.002 0.0016 0.00163 0.001615 0.001618123  
## 1 0.977 0.9777 0.97627 0.977013 0.977346278  
## 2 0.013 0.0105 0.01231 0.012306 0.012135922  
## 3 0.004 0.0054 0.00468 0.004035 0.004045307  
## 5 0.004 0.0048 0.00511 0.005031 0.004854369

#It is clear from the table of contingency that the higher the number of sample   
#the better the estimation  
#Let's calculate the errors  
  
err1 <- ftable1 - ftable  
err2 <- ftable2 - ftable  
err3 <- ftable3 - ftable  
err4 <- ftable4 - ftable  
cbind(ftable, err1, err2, err3, err4)

## ftable err1 err2 err3 err4  
## 0 0.001618123 3.818770e-04 -1.812298e-05 1.187702e-05 -3.122977e-06  
## 1 0.977346278 -3.462783e-04 3.537217e-04 -1.076278e-03 -3.332783e-04  
## 2 0.012135922 8.640777e-04 -1.635922e-03 1.740777e-04 1.700777e-04  
## 3 0.004045307 -4.530744e-05 1.354693e-03 6.346926e-04 -1.030744e-05  
## 5 0.004854369 -8.543689e-04 -5.436893e-05 2.556311e-04 1.766311e-04

#From the table it is clear that from n=100000 the error is extremely low,   
#magnitude e-04   
  
#Let's consider the squared errors  
  
sum(err1\*err1)

## [1] 1.744368e-06

sum(err2\*err2)

## [1] 4.639837e-06

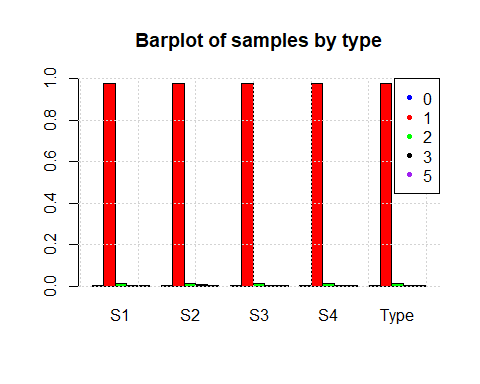
sum(err3\*err3)

## [1] 1.657001e-06

sum(err4\*err4)

## [1] 1.713154e-07

#It is clear the improvement, although there is less discrepance in the analysis  
#of SSE  
#We can also compare graphically the samples  
  
par(mfrow=c(1,1))  
barplot(cbind(ftable1, ftable2, ftable3, ftable4, ftable), beside = T,   
 names.arg = c("S1", "S2", "S3", "S4", "Type"), main = "Barplot of samples by type",   
 col=c("blue", "red", "green", "black", "purple"), ylim=c(0,1))  
grid()  
legend(x="topright", legend = c(0,1,2,3,5), pch=20, col=c("blue", "red", "green", "black", "purple"))



#We can see that, although error of low magnitude, the samples well approximate