Workshop 8 Solutions

Veerasak Kritsanapraphan

We’ll begin by loading all the packages we might need.

library(MASS)  
library(plyr)  
library(reshape) # You may need to install this one first!

##   
## Attaching package: 'reshape'

## The following objects are masked from 'package:plyr':  
##   
## rename, round\_any

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.5.2

require(moonBook)

## Loading required package: moonBook

require(webr)

## Loading required package: webr

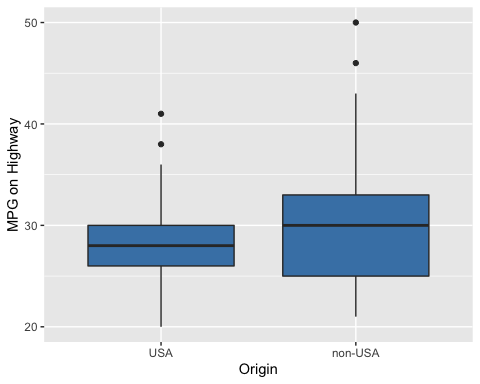
### Check Means of each group

aggregate(MPG.highway ~ Origin, data=Cars93, FUN= function(x) {  
 c(mean=mean(x), sd=sd(x))  
} )

## Origin MPG.highway.mean MPG.highway.sd  
## 1 USA 28.145833 4.151337  
## 2 non-USA 30.088889 6.247990

### Check Box Plot

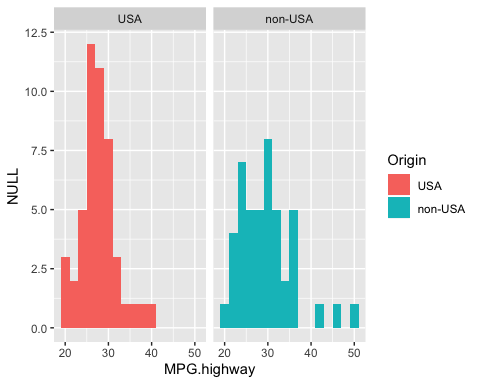
qplot(x = Origin, y = MPG.highway, geom="boxplot", data = Cars93,  
 xlab = "Origin", ylab="MPG on Highway", fill=I("steelblue"))



## Is the data normal?

**(a)** Construct histograms of MPG.highway, one plot for each Origin category.

qplot(x = MPG.highway, data = Cars93, facets = ~Origin, geom = "histogram", fill = Origin, binwidth = 2)

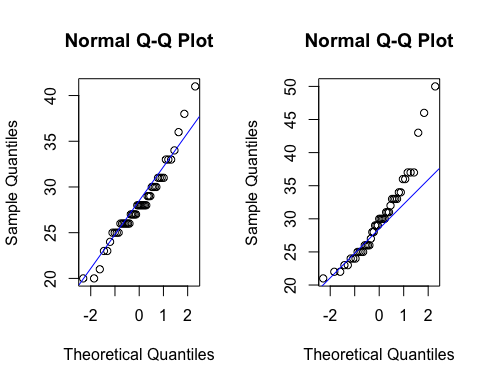


**(b)** Does the data look to be normally distributed?

The histograms don’t really look normally distributed, so we might be better off using the non-parametric test.

**(c)** Construct qqplots of MPG.highway, one plot for each Origin category. Overlay a line on each plot using with qqline() function.

par(mfrow = c(1,2))  
# USA cars  
with(Cars93, qqnorm(MPG.highway[Origin == "USA"]))  
with(Cars93, qqline(MPG.highway, col = "blue"))  
# Foreign cars  
with(Cars93, qqnorm(MPG.highway[Origin == "non-USA"]))  
with(Cars93, qqline(MPG.highway, col = "blue"))



**(d)** Does the data look to be normally distributed?

The non-USA MPG.highway data looks very far from normally distributed.

### Testing means between two groups

**(a)** Using the Cars93 data and the t.test() function, run a t-test to see if average MPG.highway is different between US and non-US vehicles.

Try doing this both using the formula style input and the x, y style input.

# Formula version  
mpg.t.test <- t.test(MPG.highway ~ Origin, data = Cars93)  
mpg.t.test

##   
## Welch Two Sample t-test  
##   
## data: MPG.highway by Origin  
## t = -1.7545, df = 75.802, p-value = 0.08339  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -4.1489029 0.2627918  
## sample estimates:  
## mean in group USA mean in group non-USA   
## 28.14583 30.08889

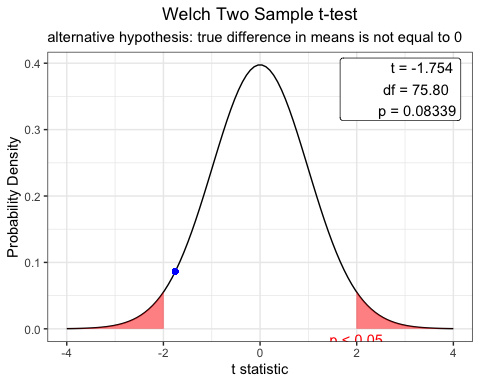
# x, y version

with(Cars93, t.test(x = MPG.highway[Origin == "USA"], y = MPG.highway[Origin == "non-USA"], alternative="greater"))

##   
## Welch Two Sample t-test  
##   
## data: MPG.highway[Origin == "USA"] and MPG.highway[Origin == "non-USA"]  
## t = -1.7545, df = 75.802, p-value = 0.9583  
## alternative hypothesis: true difference in means is greater than 0  
## 95 percent confidence interval:  
## -3.787251 Inf  
## sample estimates:  
## mean of x mean of y   
## 28.14583 30.08889

## Plot Normal Distribution

#install.packages("devtools")  
#devtools::install\_github("cardiomoon/webr")  
plot(mpg.t.test)



**(b)** What is the confidence interval for the difference?

mpg.t.test$conf.int

## [1] -4.1489029 0.2627918  
## attr(,"conf.level")  
## [1] 0.95

### If it is not normal distribution, we can use Wilcox Test

wilcox <- wilcox.test(MPG.highway ~ Origin, exact = FALSE, data=Cars93)  
wilcox

##   
## Wilcoxon rank sum test with continuity correction  
##   
## data: MPG.highway by Origin  
## W = 910, p-value = 0.1912  
## alternative hypothesis: true location shift is not equal to 0