```r
library(ggplot2)
#
# Session times for websites A and B
times1 <- read.csv('web_page_data.csv')
times1[,2] <- times1[,2] * 100
str(times1)
```

```
## 'data.frame':    36 obs. of  2 variables:
##  $ Page: Factor w/ 2 levels "Page A","Page B": 1 2 1 2 1 2 1 2 1 2 ...
##  $ Time: num  21 253 35 71 67 85 211 246 132 149 ...
```

```r
head(times1)
```

```
##     Page Time
## 1 Page A   21
## 2 Page B  253
## 3 Page A   35
## 4 Page B   71
## 5 Page A   67
## 6 Page B   85
```

```r
# number of session times for each website
table(times1$Page)
```

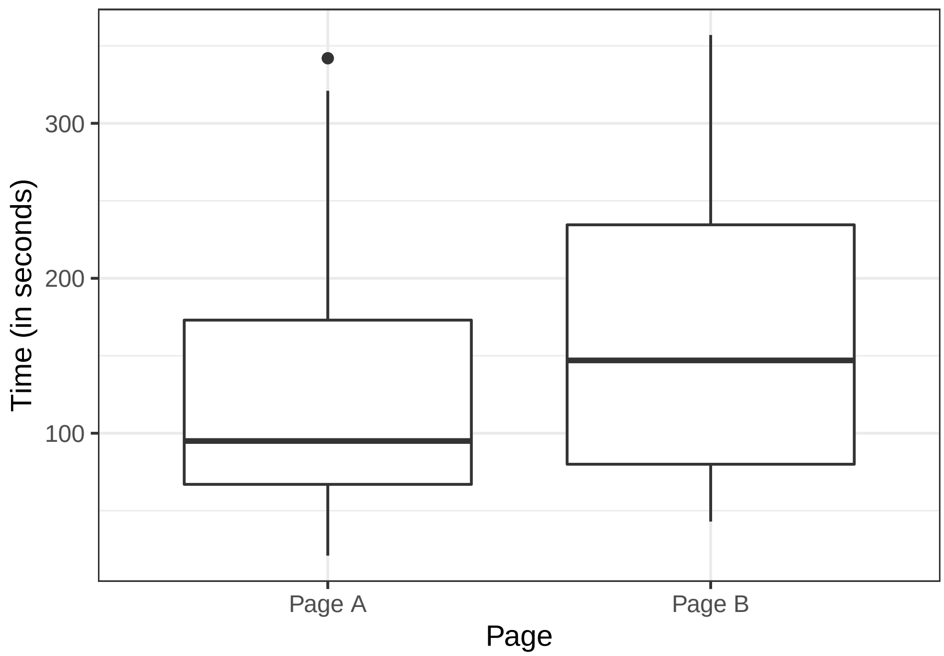
```
##
## Page A Page B
##     21     15
```

```r
# average session time for each website
aux = tapply(times1$Time,times1$Page,mean)
aux
```

```
##   Page A   Page B
## 126.3333 162.0000
```

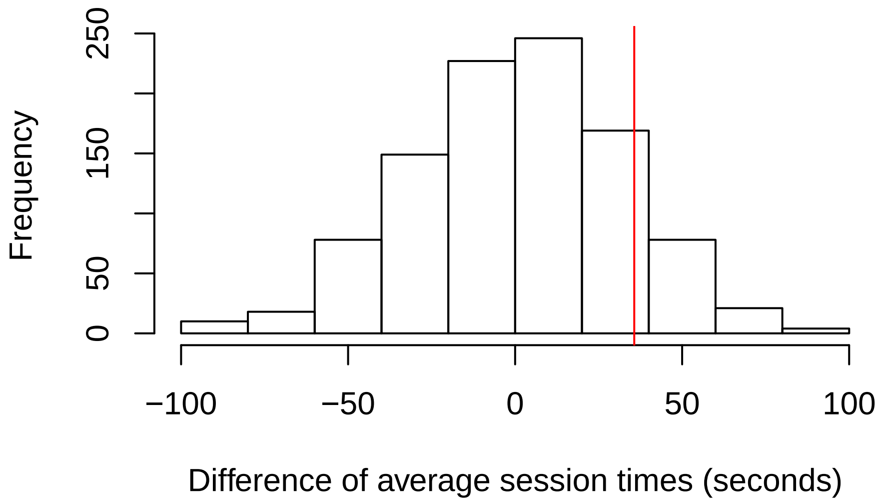
```r
obs_diff = aux[2]-aux[1]

# boxplot comparing the distribution of session times
ggplot(times1, aes(x=Page, y=Time)) +
  geom_boxplot() +
  labs(y='Time (in seconds)') +
  theme_bw()
```




```r
#
# how likely is this difference to be observed if average session times are equal?
#
# PERMUTATION TEST
#
# randomly group 21 and 15 times, then find difference of their average times
# repeat many times finding the distribution of differences of average times
# if obs_diff is out of the range of the distribution
# then conclude that the true difference of average session times is not zero
# and therefore one webpage results in longer session times, on average.
#
# note: setdiff(x,y) collects those elements in x but not in y
#
# x: vector of numeric values
# n1: group A
# n2: group B
#
# find difference (mean of group B values - mean of group A values)
#
function1 <- function(x, n1, n2)
{
  n <- n1 + n2
  idx_b <- sample(1:n, n1)
  idx_a <- setdiff(1:n, idx_b)
  mean_diff <- mean(x[idx_b]) - mean(x[idx_a])
  return(mean_diff)
}

# dist of 1000 differences between session times of two groups randomly chosen
set.seed(1)
differences <- rep(0, 1000)
times = times1$Time
for(i in 1:1000) differences[i] = function1(times, 21, 15)
hist(differences, xlab='Difference of average session times (seconds)', main='')
abline(v = obs_diff, col='red')
```




```r
# proportion of TRUE in a sequence of logical obs
aux = c(TRUE,TRUE,TRUE,FALSE,TRUE,TRUE)
aux
```

```
## [1]  TRUE  TRUE  TRUE FALSE  TRUE  TRUE
```

```r
sum(aux)
```

```
## [1] 5
```

```r
mean(aux)
```

```
## [1] 0.8333333
```

```r
# proportion of session times beyond redline (p-value approximation)
mean(differences > obs_diff)
```

```
## [1] 0.138
```

```r
#
# t-test (p-value)
t.test(Time ~ Page, data=times1, alternative='less' )
```

```
##
##  Welch Two Sample t-test
##
## data:  Time by Page
## t = -1.0983, df = 27.693, p-value = 0.1408
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##      -Inf 19.59674
## sample estimates:
## mean in group Page A mean in group Page B
##             126.3333             162.0000
```

```r
#
# p-value agrees with that of permutation test
#
test1 = t.test(Time ~ Page, data=times1, alternative='less' )
str(test1)
```

```
## List of 10
##  $ statistic  : Named num -1.1
##   ..- attr(*, "names")= chr "t"
##  $ parameter  : Named num 27.7
```

```
##   ..- attr(*, "names")= chr "df"
##  $ p.value    : num 0.141
##  $ conf.int   : num [1:2] -Inf 19.6
##   ..- attr(*, "conf.level")= num 0.95
##  $ estimate   : Named num [1:2] 126 162
##   ..- attr(*, "names")= chr [1:2] "mean in group Page A" "mean in group Page B"
##  $ null.value : Named num 0
##   ..- attr(*, "names")= chr "difference in means"
##  $ stderr     : num 32.5
##  $ alternative: chr "less"
##  $ method     : chr "Welch Two Sample t-test"
##  $ data.name  : chr "Time by Page"
##  - attr(*, "class")= chr "htest"
```

```
test1$p.value
```

```
## [1] 0.1407622
```

```
#
#
# PROPORTIONS
#
# p1, p2 population proportion of bath soap sales with designs 1 and 2
# First design is better if p1 > p2

d0 = read.csv("Xm13-09.csv",header=T)
head(d0)
```

```
##   Supermarket1 Supermarket2
## 1         4255         4255
## 2         4255         9077
## 3         4255         8855
## 4         8855         9077
## 5         7118         8855
## 6         9077         4255
```

```
#
# table shows scanner codes for different brands
# focus on 9077
tail(d0)
```

```
##      Supermarket1 Supermarket2
## 1033           NA         3745
## 1034           NA         8855
## 1035           NA         3745
## 1036           NA         8855
## 1037           NA         9077
## 1038           NA         8855
```

```
# number of sales in the Supermarkets is different

# rename cols
names(d0) = c("s1","s2")
dim(d0)
```

```
## [1] 1038    2
```

```
# soap brands purchased at Supermarket 1
table(d0$s1)
```

```
##
```

```
## 3745 4255 7118 8855 9077
##   99  218  163  244  180
```

```r
# 180 bath soaps sold in Supermarket 1
#
# soap brands purchased at Supermarket 2
table(d0$s2)
```

```
##
## 3745 4255 7118 8855 9077
##  134  228  218  303  155
```

```r
# 155 bath soaps sold in Supermarket 2
#
x = c(180,155)

# number of purchases at each Supermarket
sum(table(d0$s1))
```

```
## [1] 904
```

```r
sum(table(d0$s2))
```

```
## [1] 1038
```

```r
# number of soaps from all brands sold in 1-week
n1 = sum(table(d0$s1))
n2 = sum(table(d0$s2))
n = c(n1,n2)

# number of bath soaps sold from other brands
others = n-x

# table of counts
d2 = rbind(x,others,n)
rownames(d2) = c("brand","others","total")
colnames(d2) = c("Super1","Super2")
d2
```

```
##        Super1 Super2
## brand     180    155
## others    724    883
## total     904   1038
```

```r
# sample proportions
p1 = 180/904
p1
```

```
## [1] 0.199115
```

```r
p2 = 155/1038
p2
```

```
## [1] 0.1493256
```

```r
obs_diff = p1 - p2
obs_diff
```

```
## [1] 0.04978942
```

```r
#
# Is this difference due to different designs or to random chance?
```

```
# ignore total sales row
d1 = d2[1:2,]
d1
```

```
##        Super1 Super2
## brand     180    155
## others    724    883
```

```
# sum by row
apply(d1,1,sum)
```

```
##  brand others
##    335   1607
```

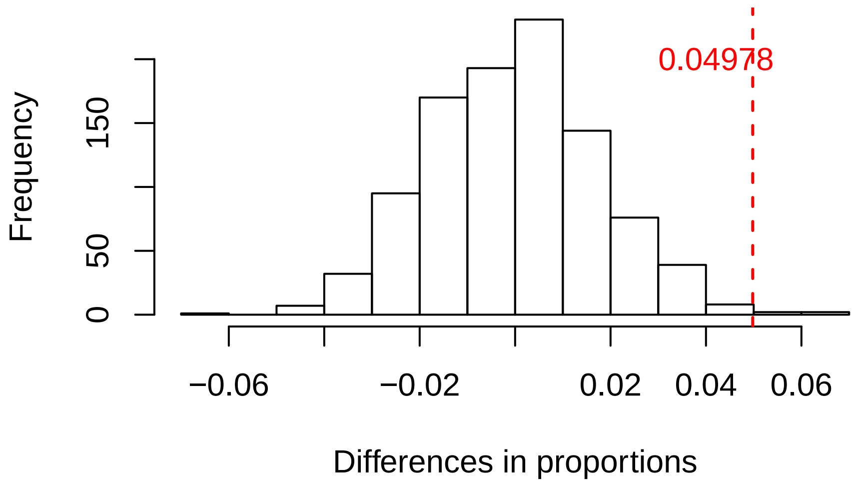
```
# total soaps sold
sum(apply(d1,1,sum))
```

```
## [1] 1942
```

```
#
# create a vector of 335 ones, 1607 zeros
vector1 <- c(rep(0, 1607), rep(1, 335))
#
# one to represent a transaction from our brand
# zero for a transaction from other brand
#
# total number of sales by SuperMarket
d2[3,]
```

```
## Super1 Super2
##    904   1038
```

```
#
# PERMUTATION TEST
#
# For Supermarket 1, select sample of 904 sales,
# record number of ones in this sample
# find proportion of ones in this sample
# For Supermarket 2,
# count number of ones in the remaining 1038 rows
# find proportion of ones in these remaining rows
#
# find difference between the 2 proportions
# repeat 1000 times
#
differences <- rep(0, 1000)
#
set.seed(1)
for(i in 1:1000) differences[i] = function1(vector1,904,1038)
hist(differences, xlab='Differences in proportions', main='')
abline(v = obs_diff, lty=2, lwd=1.5,col='red')
text("0.04978", x=0.03,y=200,adj=0,col='red')
```




```
# fraction beyond redline (approx p-value)
mean(differences > obs_diff)
```

```
## [1] 0.004
```

```
#
# test on two (or more) proportions
#
x
```

```
## [1] 180 155
```

```
n
```

```
## [1]  904 1038
```

```
prop.test(x,n, alternative="greater")
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  x out of n
## X-squared = 8.0461, df = 1, p-value = 0.00228
## alternative hypothesis: greater
## 95 percent confidence interval:
##  0.02032291 1.00000000
## sample estimates:
##    prop 1    prop 2
## 0.1991150 0.1493256
```

```
#
# p-values from random sampling and test on proportions agree
#
# p-value is smaller than alpha
# reject Ho: p1 = p2
# conclude p1 > p2
# Company should use design 1
#
# get p-value alone
#
prop.test(x,n, alternative="greater")$p.value
```

```
## [1] 0.002280073
```