Laboratory Exercise Week 3

Ryan Estes | 06

09/06/2017

*Directions*:

* Write your R code inside the code chunks after each question.
* Write your answer comments after the # sign.
* To generate the word document output, click the button Knit and wait for the word document to appear.
* RStudio will prompt you (only once) to install the knitr package.
* Submit your completed laboratory exercise using Blackboard's Turnitin feature. Your Turnitin upload link is found on your Blackboard Course shell under the Laboratory folder.

For this exercise, you will need to use the package mosaic to find numerical and graphical summaries.

# if mosaic package is not installed, then run install.packages("mosaic")   
# in your console: note that you need to this only one time  
library(mosaic) # load the package to use its functions

## Loading required package: dplyr

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

## Loading required package: lattice

## Loading required package: ggformula

## Loading required package: ggplot2

##   
## New to ggformula? Try the tutorials:   
## learnr::run\_tutorial("introduction", package = "ggformula")  
## learnr::run\_tutorial("refining", package = "ggformula")

## Loading required package: mosaicData

## Loading required package: Matrix

##   
## The 'mosaic' package masks several functions from core packages in order to add   
## additional features. The original behavior of these functions should not be affected by this.  
##   
## Note: If you use the Matrix package, be sure to load it BEFORE loading mosaic.

##   
## Attaching package: 'mosaic'

## The following object is masked from 'package:Matrix':  
##   
## mean

## The following objects are masked from 'package:dplyr':  
##   
## count, do, tally

## The following objects are masked from 'package:stats':  
##   
## binom.test, cor, cor.test, cov, fivenum, IQR, median,  
## prop.test, quantile, sd, t.test, var

## The following objects are masked from 'package:base':  
##   
## max, mean, min, prod, range, sample, sum

1. Recall the iris data set from last week's exercise. The iris data set is already pre-loaded in R - look at the help file using ?iris for more information on this data set.
   1. Check the structure of the data using the function str(iris).
   2. Find the average (or mean) measurement of the variable Sepal.Length. Do this in two ways as described in the lesson.
   3. Find the average Sepal.Length for the different flower Species. Give a brief comment on the averages.
   4. Repeat (ii) and (iii) but use the summary standard deviation sd() which describes the spread of the variable.
   5. Describe the shape of the variable Sepal.Length by creating a histogram using histogram(). Write your description outside the code chunk.
   6. Compare the Sepal.Length of the three species of flowers by creating a side-by-side boxplot using bwplot(). Write your description outside the code chunk.

### Code chunk

# Insert your code for this question after this line  
# i)  
str(iris)

## 'data.frame': 150 obs. of 5 variables:  
## $ Sepal.Length: num 5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...  
## $ Sepal.Width : num 3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...  
## $ Petal.Length: num 1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...  
## $ Petal.Width : num 0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...  
## $ Species : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...

# ii)  
mean(iris$Sepal.Length)

## [1] 5.843333

#mean(~ Sepal.Length, data = iris)  
  
# iii)  
# iv)  
# v)  
library(plyr)

## -------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)

## -------------------------------------------------------------------------

##   
## Attaching package: 'plyr'

## The following object is masked from 'package:mosaic':  
##   
## count

## The following objects are masked from 'package:dplyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize

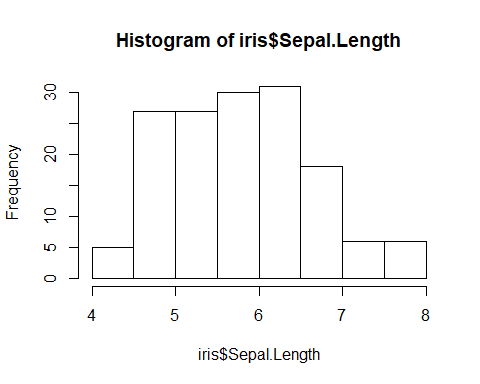
ddply(iris, ~Species,summarise,mean=mean(Sepal.Length),sd=sd(Sepal.Length))

## Species mean sd  
## 1 setosa 5.006 0.3524897  
## 2 versicolor 5.936 0.5161711  
## 3 virginica 6.588 0.6358796

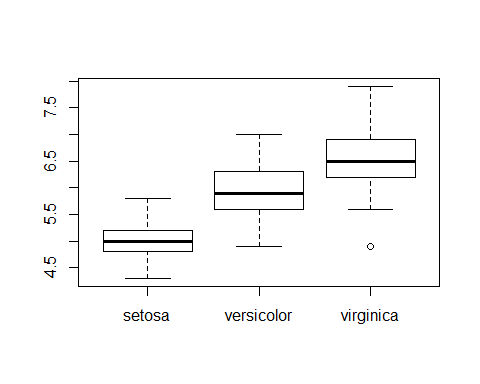
#mean(Species ~ Sepal.Length, data = iris)  
  
# Mean value of Sepal.Length of Setosa < Versicolor < Virginica  
  
# iv)   
sd(iris$Sepal.Length)

## [1] 0.8280661

#sd(~ Sepal.Length, data = iris)  
  
# v)  
hist(iris$Sepal.Length)



# vi)  
boxplot(Sepal.Length ~ Species, data = iris)



# last R code line

Species mean sd

1 setosa 5.006 0.3524897 2 versicolor 5.936 0.5161711 3 virginica 6.588 0.6358796

1. The histogram looks similar to a bell curve with most of the Lengths between 4.5-7. The most Lengths lie between 6-6.5.
2. The Boxplot description:

Setosa: Min: 4.2 1st Quartile: 4.8 Median: 5.0 3rd Quartile: 5.2 Max: 5.8

Versicolor: Min: 4.9 1st Quartile: 5.7 Median: 6.0 3rd Quartile: 6.4 Max: 7.1

Virginica: Min: 5.7 1st Quartile: 6.3 Median: 6.6 3rd Quartile: 7.0 Max: 7.9

1. The data set MLB-TeamBatting-S16.csv contains MLB Team Batting Data for selected variables. Load the data set from the given url using the code below. This data set was obtained from [Baseball Reference](https://www.baseball-reference.com/leagues/MLB/2016-standard-batting.shtml).
   * Tm - Team
   * Lg - League: American League (AL), National League (NL)
   * BatAge - Battersâ average age
   * RPG - Runs Scored Per Game
   * G - Games Played or Pitched
   * AB - At Bats
   * R - Runs Scored/Allowed
   * H - Hits/Hits Allowed
   * HR - Home Runs Hit/Allowed
   * RBI - Runs Batted In
   * SO - Strikeouts
   * BA - Hits/At Bats
   * SH - Sacrifice Hits (Sacrifice Bunts)
   * SF - Sacrifice Flies
   1. Find the average measurement for the following variables BatAge, RPG, R, H and BA.
   2. Create dotplot's or histogram's for each variable in (i).
   3. Using your own words, describe the distribution of each variable in (i). Write your answer outside the code chunk.
   4. Find the average and the standard deviation of the variables RPG, H and BA for each league.
   5. Describe any differences or similarities between the leagues. Write your comment outside the code chunk.

### Code chunk

# load the data set  
mlb16.data <- read.csv("https://raw.githubusercontent.com/jpailden/rstatlab/master/data/MLB-TeamBatting-S16.csv")  
str(mlb16.data) # check structure

## 'data.frame': 30 obs. of 14 variables:  
## $ Tm : Factor w/ 30 levels "ARI","ATL","BAL",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ Lg : Factor w/ 2 levels "AL","NL": 2 2 1 1 2 1 2 1 2 1 ...  
## $ BatAge: num 26.7 28.9 28.4 28.5 27.4 28.3 27.8 28.9 27.8 29.8 ...  
## $ RPG : num 4.64 4.03 4.59 5.42 4.99 4.23 4.42 4.83 5.22 4.66 ...  
## $ G : int 162 161 162 162 162 162 162 161 162 161 ...  
## $ AB : int 5665 5514 5524 5670 5503 5550 5487 5484 5614 5526 ...  
## $ R : int 752 649 744 878 808 686 716 777 845 750 ...  
## $ H : int 1479 1404 1413 1598 1409 1428 1403 1435 1544 1476 ...  
## $ HR : int 190 122 253 208 199 168 164 185 204 211 ...  
## $ RBI : int 709 615 710 836 767 656 678 733 805 719 ...  
## $ SO : int 1427 1240 1324 1160 1339 1285 1284 1246 1330 1303 ...  
## $ BA : num 0.261 0.255 0.256 0.282 0.256 0.257 0.256 0.262 0.275 0.267 ...  
## $ SH : int 43 64 17 8 42 29 58 31 54 17 ...  
## $ SF : int 38 52 36 40 37 44 44 60 34 38 ...

head(mlb16.data) # show first six rows

## Tm Lg BatAge RPG G AB R H HR RBI SO BA SH SF  
## 1 ARI NL 26.7 4.64 162 5665 752 1479 190 709 1427 0.261 43 38  
## 2 ATL NL 28.9 4.03 161 5514 649 1404 122 615 1240 0.255 64 52  
## 3 BAL AL 28.4 4.59 162 5524 744 1413 253 710 1324 0.256 17 36  
## 4 BOS AL 28.5 5.42 162 5670 878 1598 208 836 1160 0.282 8 40  
## 5 CHC NL 27.4 4.99 162 5503 808 1409 199 767 1339 0.256 42 37  
## 6 CHW AL 28.3 4.23 162 5550 686 1428 168 656 1285 0.257 29 44

# i)  
mean(mlb16.data$BatAge)

## [1] 28.43

mean(mlb16.data$RPG)

## [1] 4.478333

mean(mlb16.data$R)

## [1] 724.8

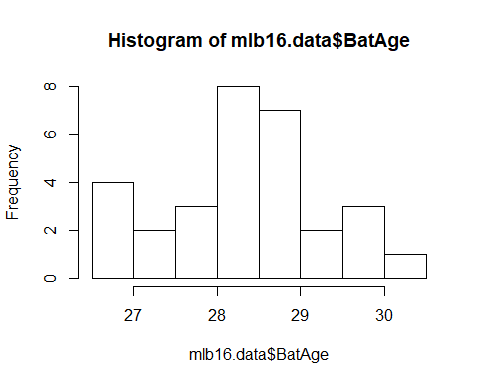
mean(mlb16.data$H)

## [1] 1409.2

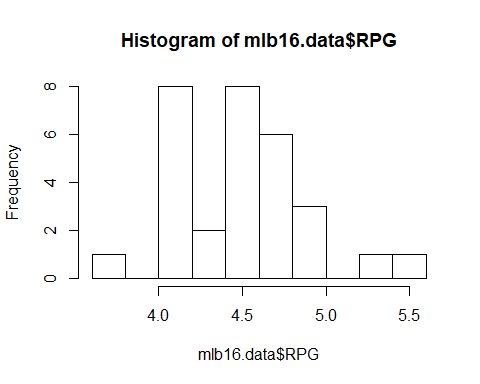
mean(mlb16.data$BA)

## [1] 0.2553

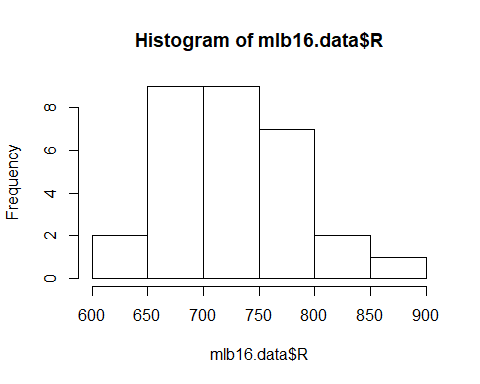
# ii)  
hist(mlb16.data$BatAge)



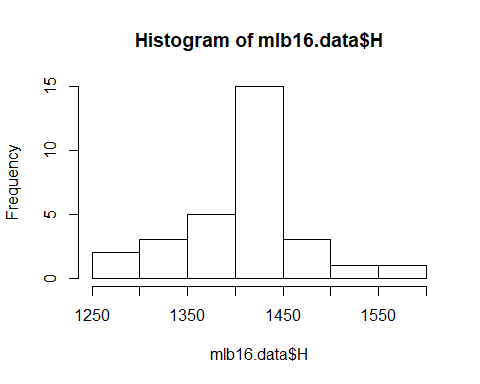
hist(mlb16.data$RPG)



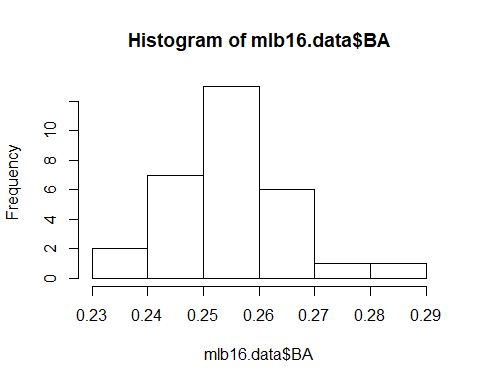
hist(mlb16.data$R)



hist(mlb16.data$H)



hist(mlb16.data$BA)



# iii)  
# Below  
  
# iv)  
  
library(plyr)  
ddply(mlb16.data, ~Lg,summarise,mean=mean(BatAge),sd=sd(BatAge))

## Lg mean sd  
## 1 AL 28.64667 1.0676186  
## 2 NL 28.21333 0.8458864

ddply(mlb16.data, ~Lg,summarise,mean=mean(RPG),sd=sd(RPG))

## Lg mean sd  
## 1 AL 4.519333 0.3533607  
## 2 NL 4.437333 0.3914345

ddply(mlb16.data, ~Lg,summarise,mean=mean(R),sd=sd(R))

## Lg mean sd  
## 1 AL 731.3333 56.83016  
## 2 NL 718.2667 63.89105

ddply(mlb16.data, ~Lg,summarise,mean=mean(H),sd=sd(H))

## Lg mean sd  
## 1 AL 1419.933 64.49858  
## 2 NL 1398.467 71.57301

ddply(mlb16.data, ~Lg,summarise,mean=mean(BA),sd=sd(BA))

## Lg mean sd  
## 1 AL 0.2568667 0.009869626  
## 2 NL 0.2537333 0.009837731

# last R code line

#ddply(mlb16.data, ~Lg,summarise,mean=mean(BatAge),sd=sd(BatAge))

Lg mean sd 1 AL 28.64667 1.0676186 2 NL 28.21333 0.8458864

The AL had a higher age but skewed more based on standard deviation.

#ddply(mlb16.data, ~Lg,summarise,mean=mean(RPG),sd=sd(RPG))

Lg mean sd 1 AL 4.519333 0.3533607 2 NL 4.437333 0.3914345

THE AL had, on average, more runs per game than the NL as well as a lower sd.

#ddply(mlb16.data, ~Lg,summarise,mean=mean(R),sd=sd(R))

Lg mean sd 1 AL 731.3333 56.83016 2 NL 718.2667 63.89105

The AL had more runs per game than the NL as well as a lower sd.

#ddply(mlb16.data, ~Lg,summarise,mean=mean(H),sd=sd(H))

Lg mean sd 1 AL 1419.933 64.49858 2 NL 1398.467 71.57301

The AL had more hits per than the NL as well as a lower sd.

#ddply(mlb16.data, ~Lg,summarise,mean=mean(BA),sd=sd(BA))

Lg mean sd 1 AL 0.2568667 0.009869626 2 NL 0.2537333 0.009837731

The AL had a slightly better batting average than the NL, but they are basically the same. .257 vs .254

#mean(mlb16.data$BatAge)

[1] 28.43

The average batter's age in the MLB for 2016 was about 28 and a half years old.

#mean(mlb16.data$RPG)

[1] 4.478333

The average runs per game in the MLB for 2016 was a little under 5.

#mean(mlb16.data$R)

[1] 724.8

The average runs scored per team in the MLB for 2016 was about 725 for the season.

#mean(mlb16.data$H)

[1] 1409.2

The average hits per team in the MLB for 2016 was about 1409 for the season.

#mean(mlb16.data$BA)

[1] 0.2553

The average batting average per team in the MLB for 2016 was about .255 (or a hit 26% of the time) for the season.