Lab exercises: beginning to work with data: distributions, visualization exercises using ggplot2 package

Thilanka Munasinghe

Data Analytics

ITWS-4600/ITWS-6600/MATP-4450/CSCI-4960

Group 1, Lab 1- part 2, January 26th, 2024

Reminder: files

https://rpi.box.com/s/4rxtho71ko160uprwkubm6rlcwrc42e6

- And some directories under this link
  - **please search before you ask**
- This is where the files for assignments, lab exercises are
  - data and code fragments…

Quantile-Quantile (Q-Q) Plot

- qqplot() function produces a quantile-quantile (Q-Q) plot, also called a probability plot.
- A *quantile-quantile (Q-Q) plot*, also called a *probability plot*, is a plot of the observed order statistics from a random sample (the empirical quantiles) against their (estimated) mean or median values based on an assumed distribution, or against the empirical quantiles of another set of data (Wilk and Gnanadesikan, 1968).
- **Q-Q plots are used to assess whether data come from a particular distribution, or whether two datasets have the same parent distribution**.

- **If the distributions have the same shape (but not necessarily the same location or scale parameters), then the plot will fall roughly on a straight line.**
- **If the distributions are exactly the same, then the plot will fall roughly on the straight line y=x**.

Read the Q-Q Plot documentation:
https://www.rdocumentation.org/packages/EnvStats/versions/2.3.1/topics/qqPlot

Exercise 1: fitting a distribution beyond histograms

- Cumulative density function

  > plot(ecdf(EPI), do.points=FALSE, verticals=TRUE)

- Quantile-Quantile
- help("qqnorm") # read the RStudio documentation for qqnorm

  > par(pty="s")

  > qqnorm(EPI); qqline(EPI)

- Make a Q-Q plot against the generating distribution by: x<-seq(30,95,1)

  > qqplot(qt(ppoints(250), df = 5), x, xlab = "Q-Q plot for t dsn")

  > qqline(x)

Read the QQ plot Documentation:
https://www.rdocumentation.org/packages/EnvStats/versions/2.3.1/topics/qqPlot

Exercise 1 code…

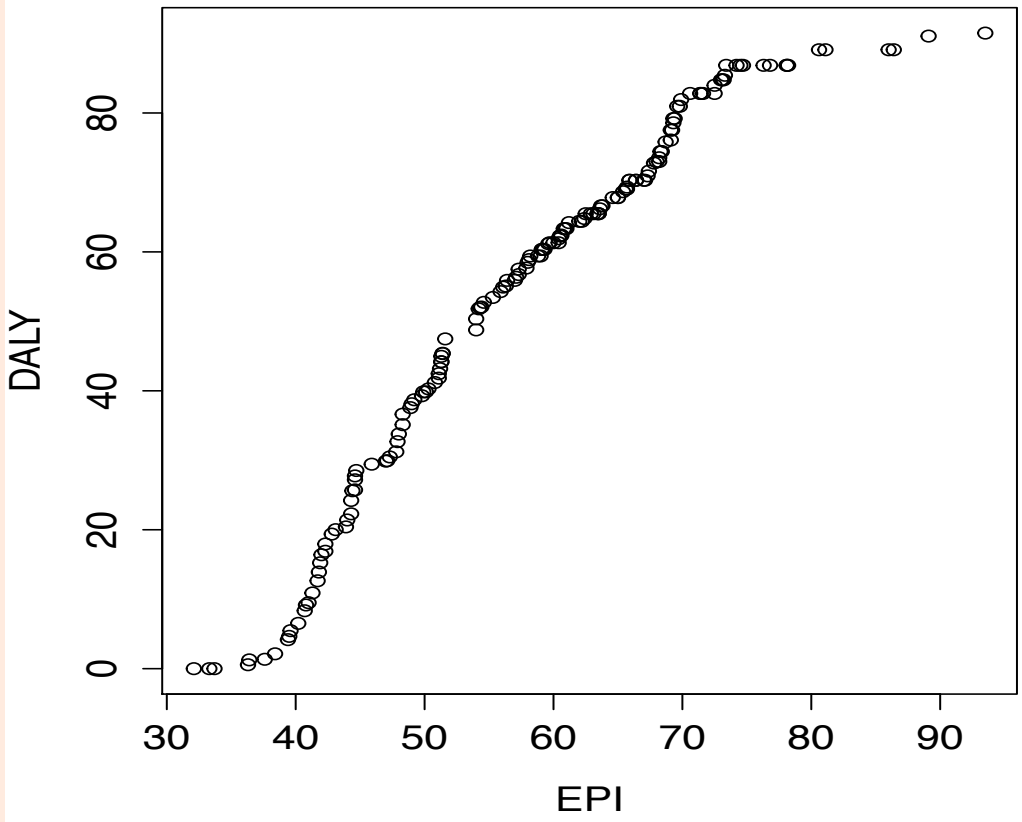
```
plot(ecdf(EPI_data$EPI),do.points=FALSE,verticals = TRUE)
plot(ecdf(EPI_data$EPI),do.points=TRUE,verticals = TRUE) # points are visible on the plot.
par(pty="s")
help("qqnorm") # read the RStudio documentation for qqnorm
help("qqplot") # read the RStudio documentation for qqplot
qqnorm(EPI_data$EPI)
qqline(EPI_data$EPI) # adding the line on the Q-Q plot
x <- seq(30,95,1)
x
x2 <-seq(30,95,2)
x2
x2 <-seq(30,96,2)
x2
qqplot(qt(ppoints(250),df=5),x, xlab = "Q-Q plot")
qqline(x)
```

Exercise 1: fitting a distribution

- Your exercise: do the same exploration and fitting for another 2 variables in the EPI_data, i.e. primary variables (DALY, WATER_H, …)

- Try fitting other distributions – i.e. as ecdf or qq-

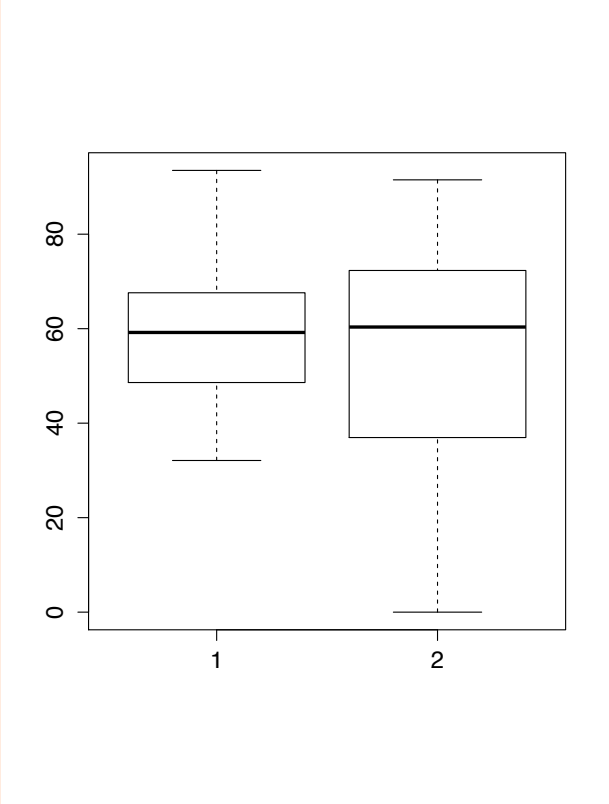
qqplot(EPI,DALY)



Read the QQ Documentation:
https://www.rdocumentation.org/packages/EnvStats/versions/2.3.1/topics/qqPlot

Comparing distributions

```
boxplot(EPI_data$EPI,EPI_data$DALY)
```



But there is more

- Your exercise – intercompare: EPI, ENVHEALTH, ECOSYSTEM, DALY, AIR_H, WATER_H, AIR_EWATER_E, BIODIVERSITY ** (subject to possible filtering…)

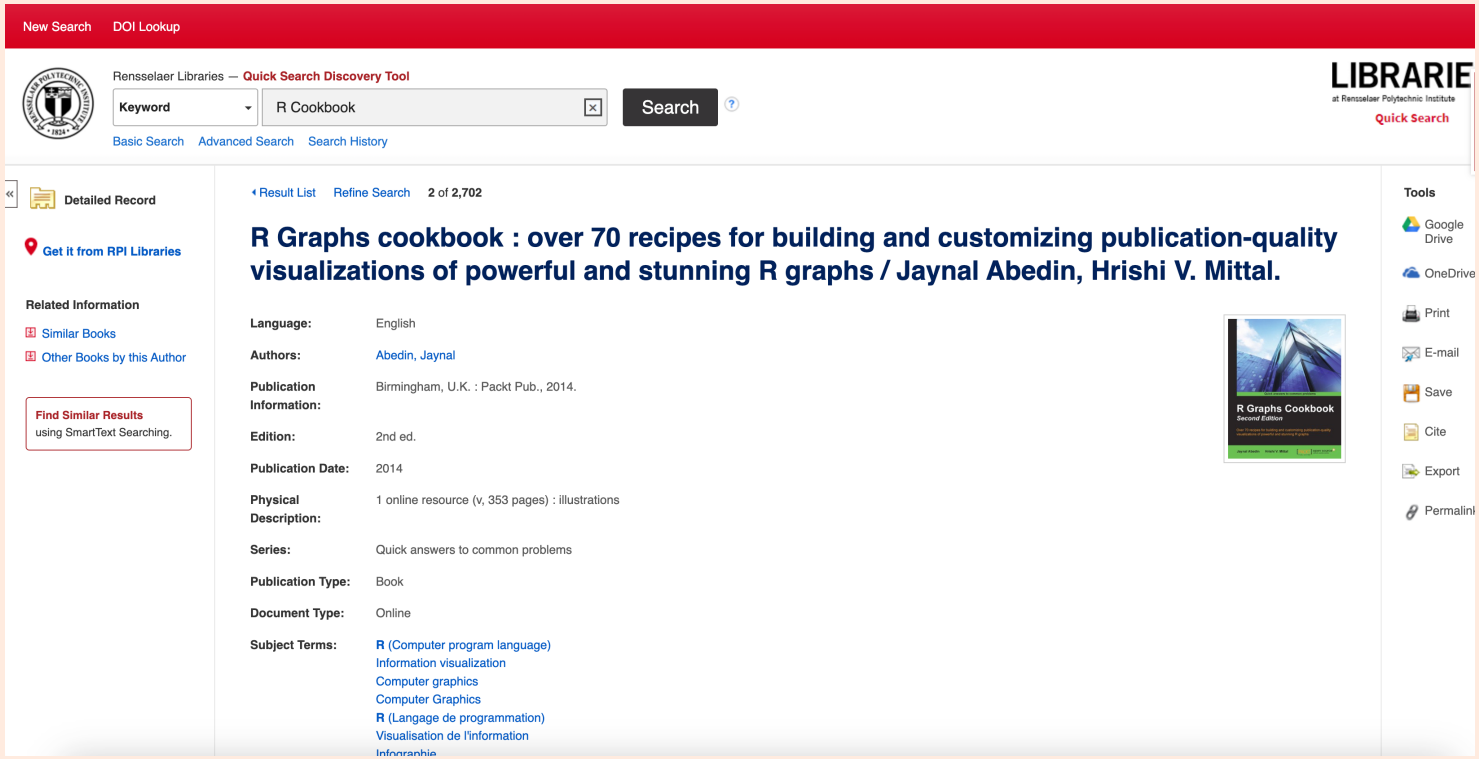
- Note 2010 and 2016 datasets….
- Environmental Performance Index (EPI) Datasets are from: https://sedac.ciesin.columbia.edu/data/collection/epi

Outliers in Regression

- How does an outlier influence the Least Square line?
- In general, outliers are the points that fall away from the cloud of points.
- Two types –
  - Leverage Points : Outliers that fall horizontally away from the center of the cloud of points but don't influence the slope of the regression line are called leverage points.
  - Influential Points : Outliers that actually influence the slope of the regression line are called influential points.

Exercises from R Cookbook

- The online version of this book is available at RPI library (RCS login required).
- ***R Graphs cookbook : over 70 recipes for building and customizing publication-quality visualizations of powerful and stunning R graphs / Jaynal Abedin, Hrishi V. Mittal.***
- Work on the Chapter 1-2 examples from this book. Some important parts are given/shown in below slides.



In-Class Work: ggplot examples

These code snippets will not *run-as-is* you need to read and work on the remaining part of the Chapter 2 of R Graphics Cookbook

```
# Creating Plots
# Chapter 2 -- R Graphics Cookbook.
plot(mtcars$wt,mtcars$mpg)
library(ggplot2)
qplot(mtcars$wt,mtcars$mpg)
qplot(wt,mpg,data = mtcars)
ggplot(mtcars,aes(x=wt,y=mpg))+ geom_point()
plot(pressure$temperature,pressure$pressure, type = "l")
points(pressure$temperature,pressure$pressure)

lines(pressure$temperature,pressure$pressure/2, col="red")
points(pressure$temperature,pressure$pressure/2, col="blue")
library(ggplot2)
qplot(pressure$temperature,pressure$pressure, geom="line")
qplot(temperature,pressure, data = pressure, geom = "line")
ggplot(pressure, aes(x=temperature,y=pressure)) + geom_line() + geom_point()
ggplot(pressure, aes(x=temperature, y=pressure))+ geom_line() + geom_point()
```

Creating Bar graphs

- **These code snippets will not *run-as-is* you need to read and work on the remaining part of the Chapter 2 of R Graphics Cookbook**

```
# Creating Bar graphs
barplot(BOD$demand, names.arg = BOD$Time)
table(mtcars$cyl)
barplot(table(mtcars$cyl)) # generate a table of counts.
qplot(mtcars$cyl) # cyl is continous here
qplot(factor(mtcars$cyl)) # treat cyl as discrete
# Bar graph of counts
qplot(factor(cyl),data = mtcars)
ggplot(mtcars,aes(x=factor(cyl))) + geom_bar()
```

Creating Histograms using ggplot

- **These code snippets will not *run-as-is* you need to read and work on the remaining part of the Chapter 2 of R Graphics Cookbook**

```
# Creating Histogram
# View the distribution of one-dimentional data with a histogram.
hist(mtcars$mpg)
hist(mtcars$mpg, breaks = 10) # specify apprioximate number of bins with breaks.
hist(mtcars$mpg, breaks = 5)
hist(mtcars$mpg, breaks = 12)
qplot(mpg, data = mtcars, binwidth=4)
ggplot(mtcars,aes(x=mpg)) + geom_histogram(binwidth = 4)
ggplot(mtcars, aes(x=mpg)) + geom_histogram(binwidth = 5)
```

Creating Box-plots using ggplot

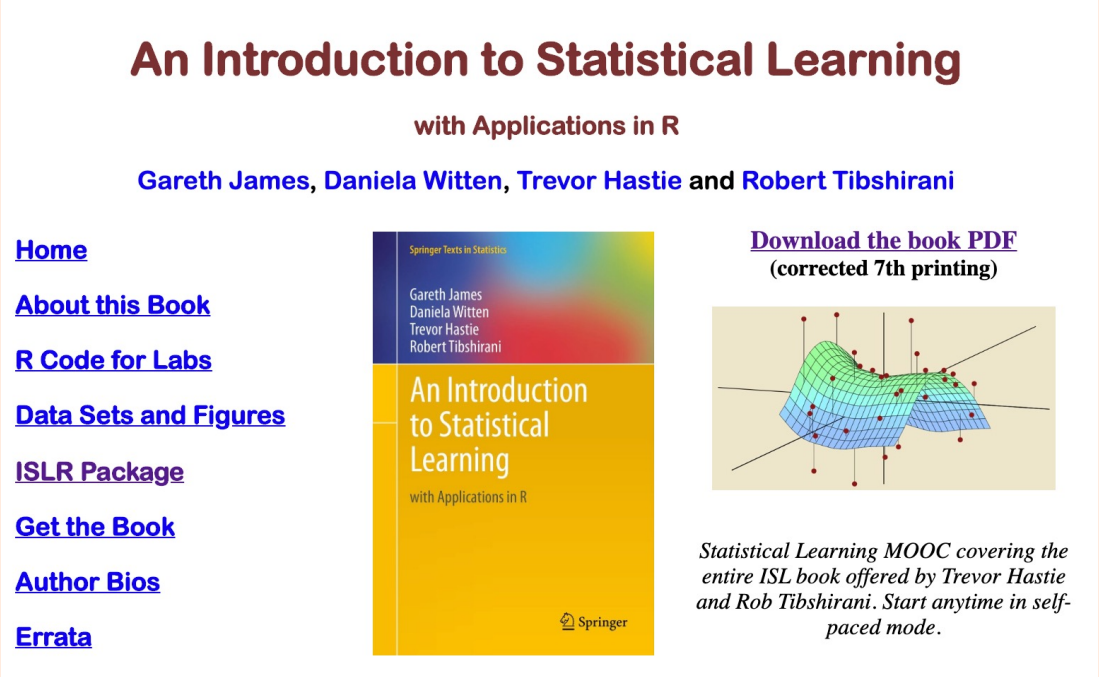
```
# Creating Box-plot
plot(ToothGrowth$supp, ToothGrowth$len) # using plot() function and pass it a factor of x-values and a vecctor of y-values.
#Formula Syntax
boxplot(len ~ supp, data = ToothGrowth) # if the tow vectors are in the same dataframe, you can use the formula syntax. With
# this syntax you can combine two variables on the x-axis.
# put interaction  of two variables on x-axis
boxplot(len ~ supp + dose, data = ToothGrowth)
# with ggplot2 you can get the same results above.
library(ggplot2)
qplot(ToothGrowth$supp, ToothGrowth$len, geom = "boxplot")
# if the two vectors are in the same dataframe, you can use the following syntax
qplot(supp, len, data = ToothGrowth, geom = "boxplot")
# in ggplot2, the above is equvalent to:
ggplot(ToothGrowth, aes(x=supp, y=len)) + geom_boxplot()
# Using three seperate vectors
qplot(interaction(ToothGrowth$supp, ToothGrowth$dose), ToothGrowth$len, geom = "boxplot")
# You can write the samething above, get the columns from the dataframe
qplot(interaction(supp, dose), len, data = ToothGrowth, geom = "boxplot")
# Using ggplot() you can do the samething and it is equivalent to:
ggplot(ToothGrowth, aes(x=interaction(supp, dose), y=len)) + geom_boxplot()
#Plotting a function curve
```

Visualization exercise

- Additional ggplot2 library examples are available on LMS. Please see the "**visualization exercise**: ggplot and bar graphs" R code snippet on LMS
- After you finish the visualization exercise, make sure to push your code to GitHub.
- ***TA will check your code on GitHub today for this visualization exercises before end of the class!***
- ***Make sure to push your code to GitHub!***

Textbook

- Introduction to Statistical Learning with Applications in R ~ 7th Edition
- https://www.statlearning.com/



- Push your Lab1_part2 code to your Github repository.
- Share your GitHub repo URL with the TA if you have not shared it.
- **Project dataset search: This is a reminder for you to look/search for the datasets that you will be working for the class project.**
- **Read: Chapter 3 – Introduction to Statistical Learning with Applications in R, 7th Edition**