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

Thilanka Munasinghe

Data Analytics

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

Group 1, Lab 1- part 2, September 19th, 2023

Reminder: files

https://rpi.box.com/s/lp28bxs8xk26ow80unnkiax916afibfn

- 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…

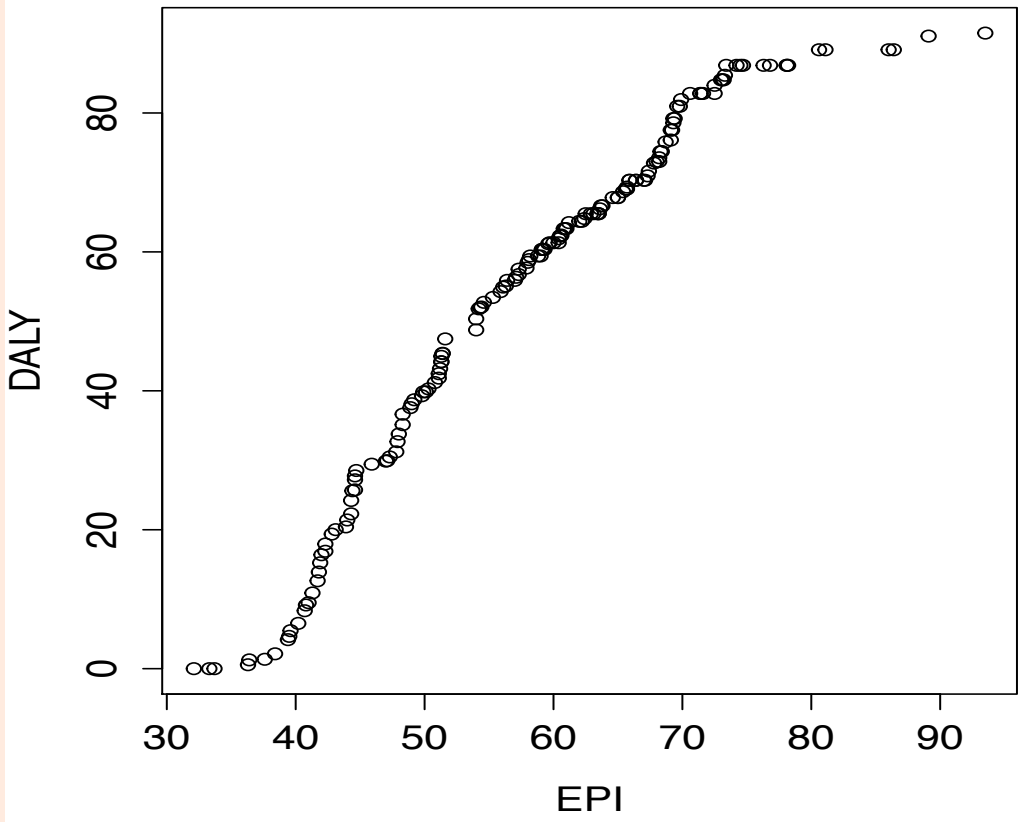
```
plot(ecdf(EPI_data$EPI),do.points=FALSE,verticals = TRUE)
plot(ecdf(EPI_data$EPI),do.points=TRUE,verticals = TRUE) # points are visible on the plot.
par(pty="s")
help("qqnorm") # read the RStudio documentation for qqnorm
help("qqplot") # read the RStudio documentation for qqplot
qqnorm(EPI_data$EPI)
qqline(EPI_data$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-

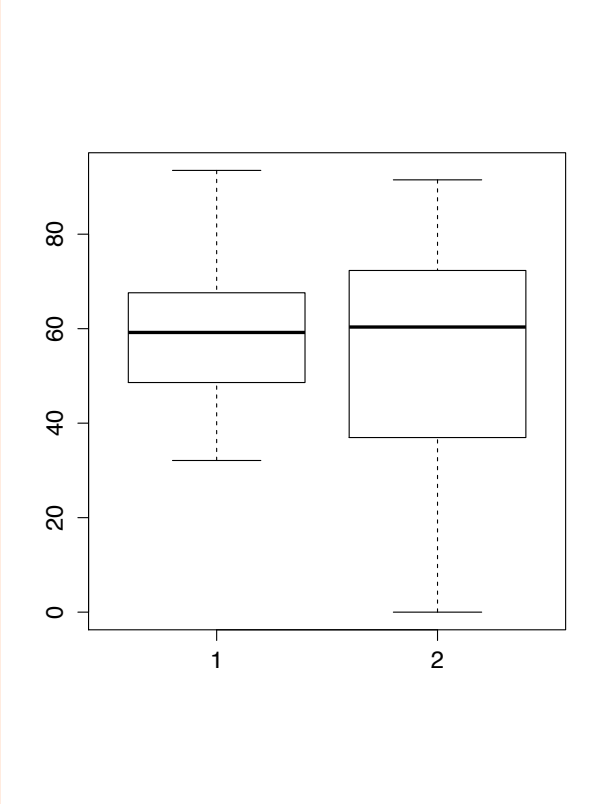
qqplot(EPI,DALY)



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

Input/Output

- Input: inputs go by different names,

  input: ***predictors***, ***independent variables***, ***features***, sometimes just variables

  $X = (x_1, x_2, \ldots, x_p)$

- Output: The output variable called the ***response*** **or** ***dependent variable***, typically denoted by Y

- Suppose that we observe Quantitative response Y, and *p* different predictors,

  $x_1, x_2, \ldots x_p$.
- We assume some relationship between Y and X =($x_1, x_2, \ldots x_p$) , which can be written as:

$$Y = f(x) + \varepsilon$$

f is an unknown function of x

random error term, which is independent of x

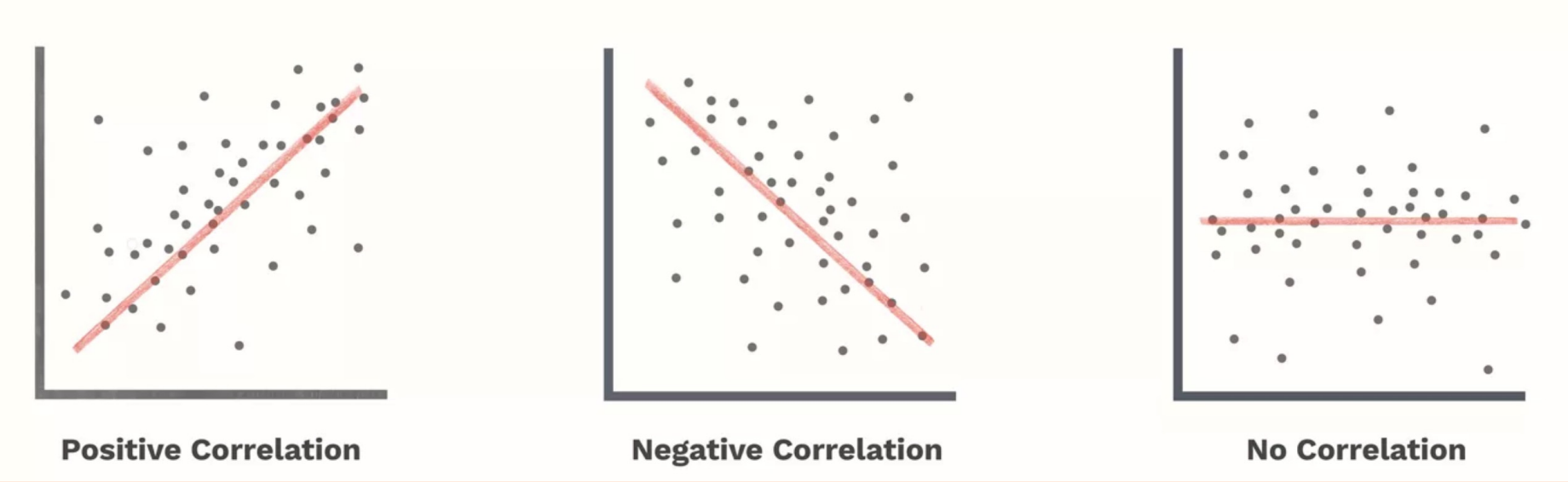
Correlation

- One measure of the strength of the association between two numerical variables is correlation.
- Correlation describes the strength of the linear association between two variables.
- Correlation coefficient is between -1 and +1
- -1 indicates a perfect negative linear association and +1 indicates a perfect positive linear association. The correlation coefficient of 0, indicates that there is no linear relationship in the two variables. -0.1 and +0.1, indicates no linear relationship *or a very weak* linear relationship

- Correlation coefficient is sensitive to outliers.
- Correlation coefficient is unitless.

Reference(s): https://www.investopedia.com/terms/c/correlationcoefficient.asp
https://www.investopedia.com/ask/answers/032515/what-does-it-mean-if-correlation-coefficient-positive-negative-or-zero.asp

Correlation…

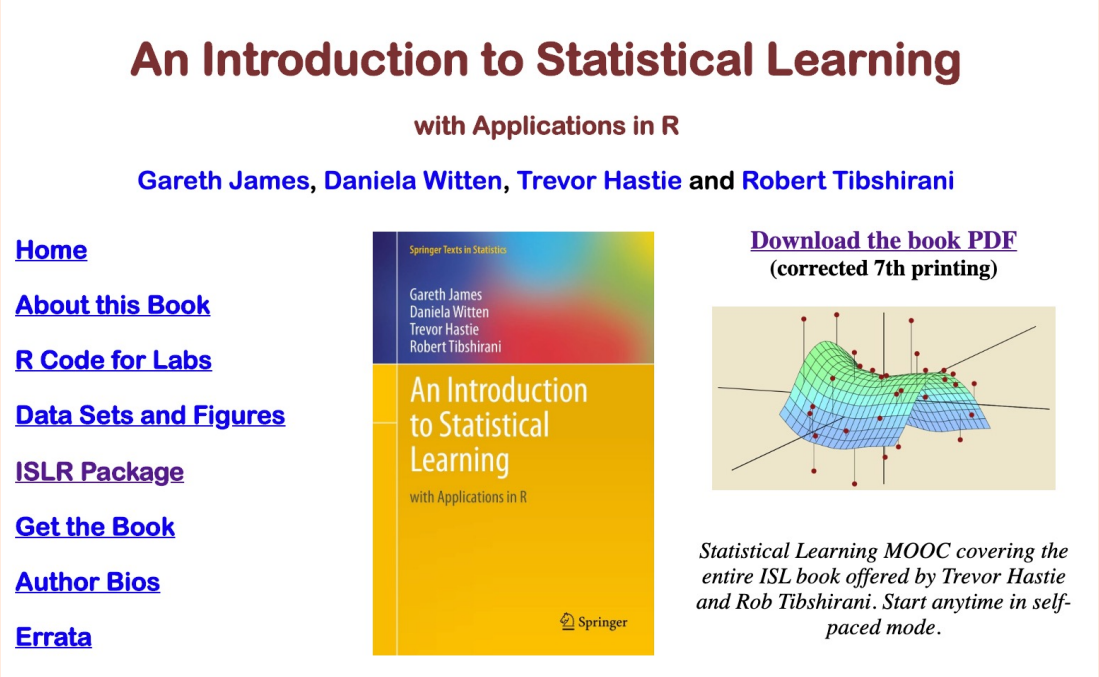


Image/Photo Credit: https://www.investopedia.com/ask/answers/032515/what-does-it-mean-if-correlation-coefficient-positive-negative-or-zero.asp

Reference(s): https://www.investopedia.com/terms/c/correlationcoefficient.asp
https://www.investopedia.com/ask/answers/032515/what-does-it-mean-if-correlation-coefficient-positive-negative-or-zero.asp

Textbook

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



Residuals …

- The residual is defined as the difference between the observed value and the predicted value.(Difference between the observed value and the predicted value of the response variable for a given data point).

$e_i = yi - \hat{y}_i$ represents the i^{th} residual,

this is the difference between the i^{th} observed response value and the i^{th} response value that is predicted by the linear model.

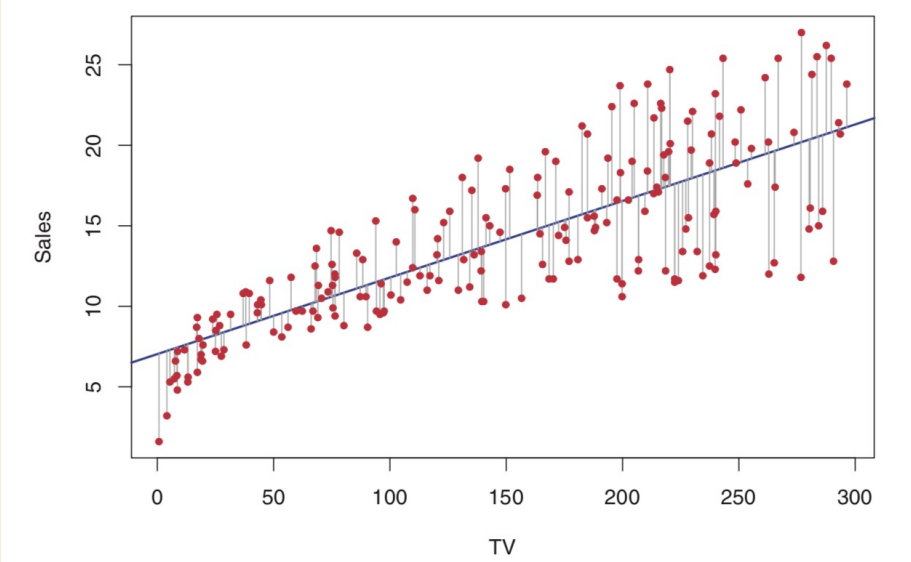


Image Credit: Introduction to Statistical Learning with Applications in R, 7th Edition, Chapter 3 – Linear Regression

Reference: Introduction to Statistical Learning with Applications in R, 7th Edition, Chapter 3 - Linear Regression

Best line?

- How do we measure the best line?
- There are two options:

Option 1: Minimize the sum of magnitudes(absolute values) of the residuals

$$|e_1| + |e_2| + |e_3| + \ldots + |en|$$

OR

Option 2: Minimize the sum of squared residuals

$$\text{RSS} = e_1^2 + e_2^2 + e_3^2 + \ldots + e_n^2$$

$e_i = y_i - \hat{y}_i$ represents the i^{th} residual

Homework: read Chapter 3 of Introduction to Statistical Learning with Applications in R, 7th Edition

Reference: Introduction to Statistical Learning with Applications in R, 7th Edition, Chapter 3 - Linear Regression

Least Squares Line

- Most commonly used is the *Least Squares approach*
- Least Squares approach choses $\hat{\beta}_0$ and $\hat{\beta}_1$ to minimize the RSS

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$$

Predicted response

Intercept

Slope

Explanatory variable

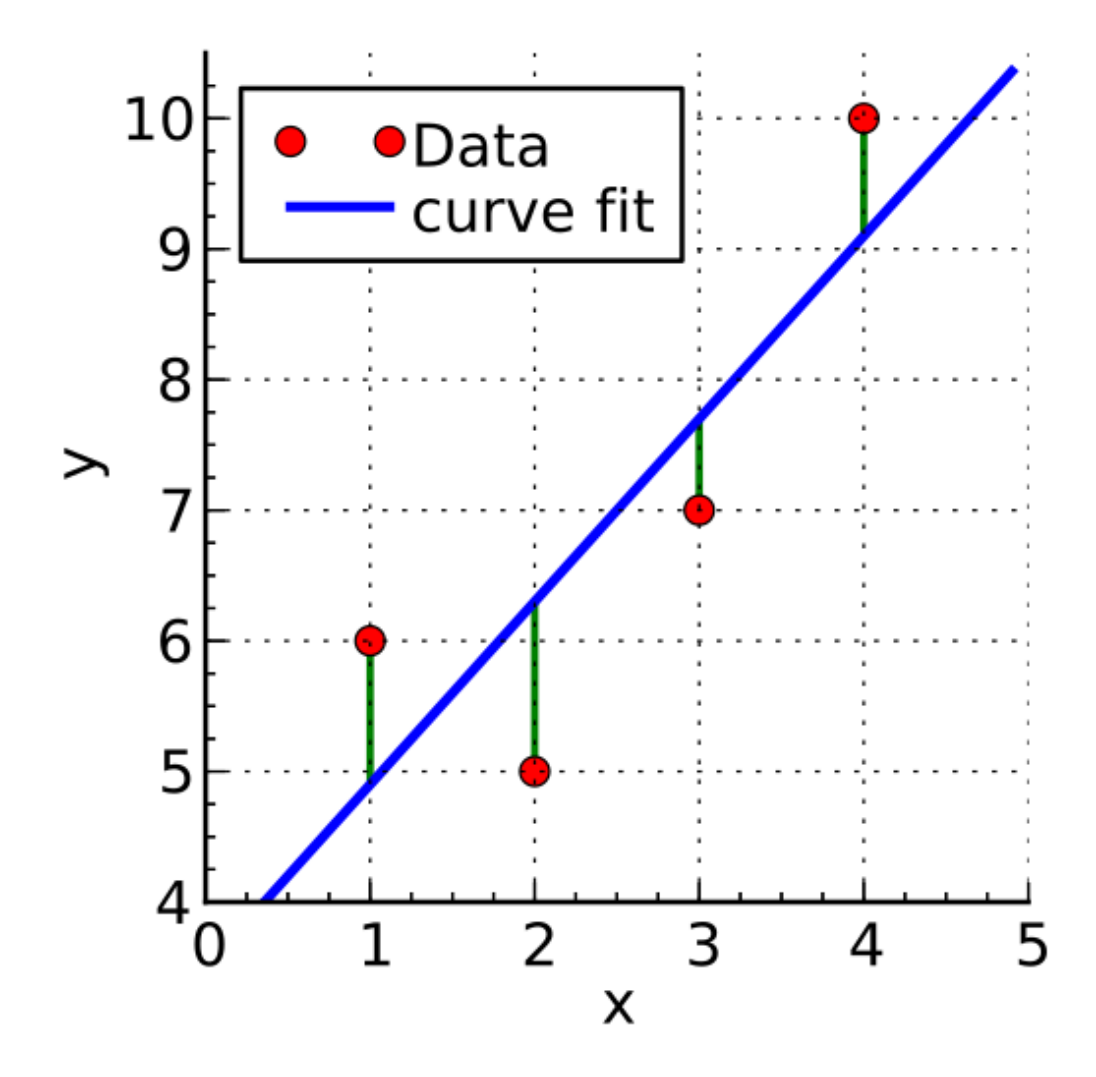
- $\hat{y}$ = Predicted value of the response variable
- x = Explanatory variable (x)
- $\hat{\beta}_0$ = Intercept
- $\hat{\beta}_1$ = Slope

Reference: Introduction to Statistical Learning with Applications in R, 7th Edition, Chapter 3 - Linear Regression

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.

regression…



Linear basis and least-squares constraints

```
> multivariate <-
read.csv("~/Documents/teaching/DataAnalytics/
data/multivariate.csv")
> attach(multivariate)
> mm<-lm(Homeowners~Immigrant)
> mm
```

dependent variable

independent variable

```
Call:
lm(formula = Homeowners ~ Immigrants)

Coefficients:
(Intercept)   Immigrants
     107495        -6657
```

	City	Income	Population	Immigrants	Homeowners	area
1	A	35460	20066	10.62	12081	50108
2	B	27038	70526	11.53	56518	141209
3	C	40337	99433	10.26	92478	199498
4	D	44833	27118	10.75	2920	54752
5	E	34590	86892	14.47	25331	173937
6	F	31205	34764	11.62	12764	70091
7	G	47102	41286	12.67	5044	82834

Multivariate.csv dataset

Dataset “multivariate.csv” is available at:

https://rpi.box.com/s/m2fs5xa27e1uaeso3pwct8x0shc7qrjy

```
# Multivariate Regression
multivariate <-read.csv("~/Downloads/multivariate.csv")
head(multivariate)
attach(multivariate)
help(lm)
mm <-lm(Homeowners~Immigrant)
mm # mm here is a R object.
summary(mm)$coef # The output above shows the estimate of the regression beta coefficients (column Estimate) and
# their significance levels (column Pr(>|t|).
# The intercept is 107494.898 and the coefficient of Immigrant variable is -6656.839.
# The estimated regression equation can be written as follow:
# Homeowners = 107494.898 + (-6656.839)*Immigrant
# We can rewrite it as:
# Homeowners = 107494.898 - 6656.839*Immigrant.
```

```
plot(Homeowners~Immigrant)
help(abline)
abline(mm)
abline(mm,col=2,lwd=3)
# Using this formula, for each new value in Immigrant, you can predict the value for Homeowners.
# As an example:
# For Immigrant value = 0, we will get: Homeowners = 107494.898 - 6656.839*0 = 107494.898
# for Immigrant value = 20, we will get: Homeowners = 107494.898 - 6656.839*20 = -25641.88
# Predictions can be easily made using the R function predict().
# In the following example, we predict Homeowners for two Immigrant values: 0 and 20.
# you can pass the 0 and 20 values as a concatenated list for Immigrants as follows:
newImmigrantdata <- data.frame(Immigrant = c(0,  20))
mm %>% predict(newImmigrantdata)

abline(mm)
abline(mm,col=3,lwd=3) # line color = green, line width = 3
attributes(mm)
mm$coefficients
```

In-Class Work: ggplot examples

```
# 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

```
# 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

```
# 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 exersize, make sure to push your code to github.

- 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**