StatR 101: Fall 2012

Homework 3

Rod Doe

Saturday, October 13, 2012

# Median function:

## Implementation:

Median <- function(x)

{

result = NA

# Clean the source vector of NA values.

xc <- sort(x[which(!is.na(x))])

# If any clean data remain, continue to analyze.

if (length(xc) > 0)

{

# Need this point in either even or odd data count.

# This selects 3 in either (1,2,3,4,5) or (1,2,3,4).

m1 = xc[(length(xc) %/% 2) + 1]

result = m1 # Assume length is odd.

# Is length even?

if ((length(xc) %% 2) == 0)

{

# This selects 2 in (1,2,3,4)

m2 = xc[(length(xc) %/% 2)]

result = mean(c(m1, m2))

}

}

return(result)

}

Test cases:

## > xed = c(NA, 10, NA, 20, NA, 30, NA, 40)

## > Median(xed)

## [1] 25

## > xod = c(xed, 50)

## > Median(xod)

## [1] 30

## > xna = c(NA,NA,NA,NA)

## > Median(xna)

## [1] NA

> dfCountry = read.csv("Country.csv")

> names(dfCountry)

[1] "Country" "Country2" "Area" "Literacy" "Population"

[6] "GDP" "Water" "Birthrate" "Continent"

> xd = dfCountry$Birthrate

> median(xd)

[1] NA

> median(xd[which(!is.na(xd))])

[1] 18.09

> Median(xd[which(!is.na(xd))])

[1] 18.09

> Median(xd)

[1] 18.09

# Skew Function

## Implementation

Skew <- function(x)

{

result = NA

# Clean the source vector of NA values.

xc <- x[which(!is.na(x))]

# If any clean data remain, continue to analyze.

if (length(xc) > 0)

{

xbar = mean(xc)

numerator = (sum((xc - xbar)^3)) / length(xc)

denominator = sqrt(var(xc))^3

result = numerator / denominator

}

return(result)

}

This implementation exploits the variance relationship in denominator. The R variance uses n-1 in lieu of n, so while the magnitude may be affected slightly, the sign is correct and the resultant value can be used for comparison. Anything I can delegate to R is one less opportunity for an error to creep into new code.

## Test results

> dfCountry = read.csv("Country.csv")

> xna = c(NA,NA,NA,NA)

> Skew(xna)

[1] NA

> Skew(dfCountry$GDP)

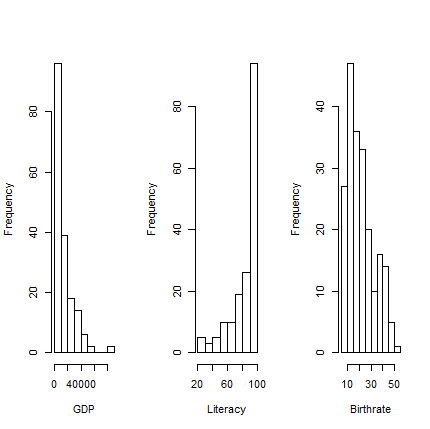
[1] 1.854636

> Skew(dfCountry$Literacy)

[1] -1.405475

> Skew(dfCountry$Birthrate)

[1] 0.7432042



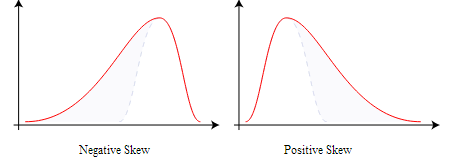
Negative skew means skew to the left, and positive skew means skew to the right. My function yields:

* A positive number for the clearly left-skewed GDP.
* A negative number for the clearly right-skewed Literacy.
* A positive number for the left-skewed Birthrate.

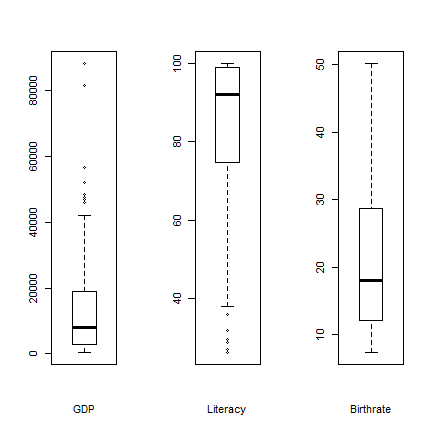
The boxplot depiction of these data is more informative, and matches the results of my Skew function. Here is the Wikipedia definition of skew:

*Qualitatively, a negative skew indicates that the tail on the left side of the probability density function is longer than the right side and the bulk of the values (possibly including the median) lie to the right of the mean. A positive skew indicates that the tail on the right side is longer than the left side and the bulk of the values lie to the left of the mean. A zero value indicates that the values are relatively evenly distributed on both sides of the mean, typically (but not necessarily) implying a symmetric distribution.*

Also from the Wikipedia article:

**

Here are boxplots of the GDP, Literacy, and Birthrate from the country data.



Repeating my Skew function results:

> Skew(dfCountry$GDP)

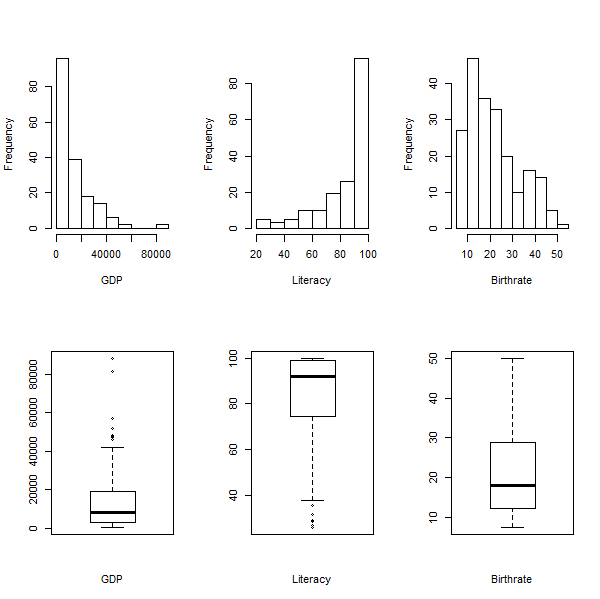
[1] 1.854636

> Skew(dfCountry$Literacy)

[1] -1.405475

> Skew(dfCountry$Birthrate)

[1] 0.7432042



> par(mfrow=c(2,3))

> hist(dfCountry$GDP, main="", xlab="GDP")

> hist(dfCountry$Literacy, main="", xlab="Literacy")

> hist(dfCountry$Birthrate, main="", xlab="Birthrate")

> boxplot(dfCountry$GDP, main="", xlab="GDP")

> boxplot(dfCountry$Literacy, main="", xlab="Literacy")

> boxplot(dfCountry$Birthrate, main="", xlab="Birthrate")

# GDP, Literacy, and Birth Rate by Continent

Africa = c(Skew(dfCountry$GDP[dfCountry$Continent == 'Africa']), Skew(dfCountry$Literacy[dfCountry$Continent == 'Africa']), Skew(dfCountry$Birthrate[dfCountry$Continent == 'Africa']) )

> Africa

[1] 1.9840605 -0.4471456 -0.6071048

> names(Africa) = c('GDP', 'Literacy', 'Birthrate')

> Africa

GDP Literacy Birthrate

1.9840605 -0.4471456 -0.6071048

> Asia = c(Skew(dfCountry$GDP[dfCountry$Continent == 'Asia']), Skew(dfCountry$Literacy[dfCountry$Continent == 'Asia']), Skew(dfCountry$Birthrate[dfCountry$Continent == 'Asia']) )

> Europe = c(Skew(dfCountry$GDP[dfCountry$Continent == 'Europe']), Skew(dfCountry$Literacy[dfCountry$Continent == 'Europe']), Skew(dfCountry$Birthrate[dfCountry$Continent == 'Europe']) )

> Europe

[1] 1.168448 -2.996606 1.237022

> NorthAmerica = c(Skew(dfCountry$GDP[dfCountry$Continent == 'North America']), Skew(dfCountry$Literacy[dfCountry$Continent == 'North America']), Skew(dfCountry$Birthrate[dfCountry$Continent == 'North America']) )

> Oceania = c(Skew(dfCountry$GDP[dfCountry$Continent == 'Oceania']), Skew(dfCountry$Literacy[dfCountry$Continent == 'Oceania']), Skew(dfCountry$Birthrate[dfCountry$Continent == 'Oceania']) )

> Oceania

[1] 0.77610163 -0.81064154 0.08399888

> SouthAmerica = c(Skew(dfCountry$GDP[dfCountry$Continent == 'South America']), Skew(dfCountry$Literacy[dfCountry$Continent == 'South America']), Skew(dfCountry$Birthrate[dfCountry$Continent == 'South America']) )

> SouthAmerica

[1] 0.1234674 0.1612707 0.7929967

> World = c(Skew(dfCountry$GDP), Skew(dfCountry$Literacy), Skew(dfCountry$Birthrate) )

> dfByContinent = rbind(Africa, Asia, Europe, NorthAmerica, Oceania, SouthAmerica, World)

### Skew of GDP, Literacy, and Birth Rate by Continent

> dfByContinent

GDP Literacy Birthrate

Africa 1.9840605 -0.4471456 -0.60710481

Asia 1.7112893 -1.2918465 0.46129432

Europe 1.1684477 -2.9966058 1.23702249

NorthAmerica 1.2909156 -0.9506784 0.73122855

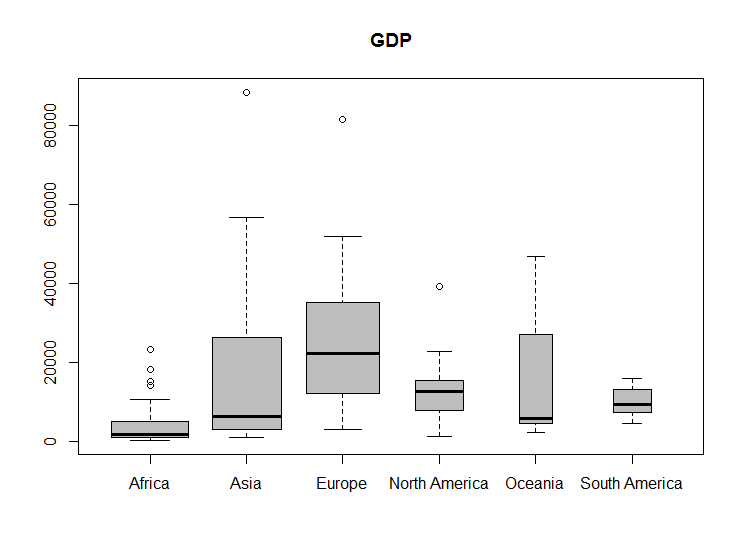
Oceania 0.7761016 -0.8106415 0.08399888

SouthAmerica 0.1234674 0.1612707 0.79299668

World 1.8546359 -1.4054746 0.74320418

GDP as a function of continent

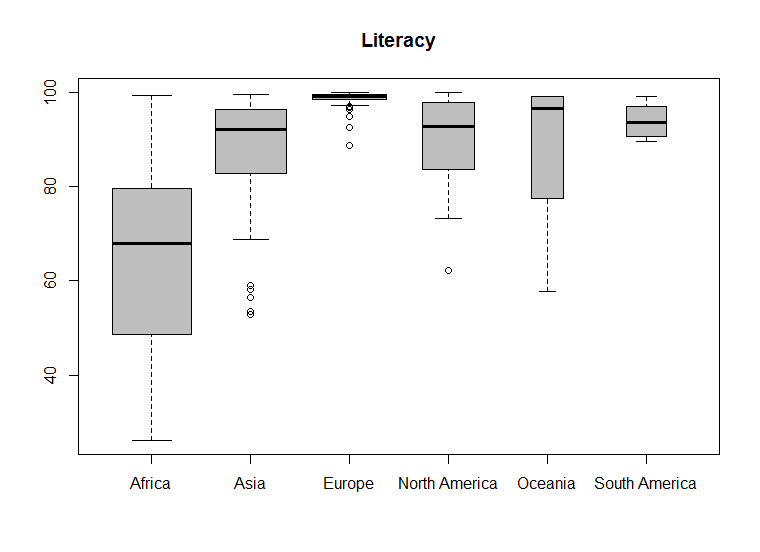
aggregate(C$GDP, list(C$Continent), Skew)

boxplot(GDP~Continent, data=C, varwidth=TRUE, main="GDP", col="grey")

## Literacy as a function of continent

aggregate(C$Literacy, list(C$Continent), Skew)

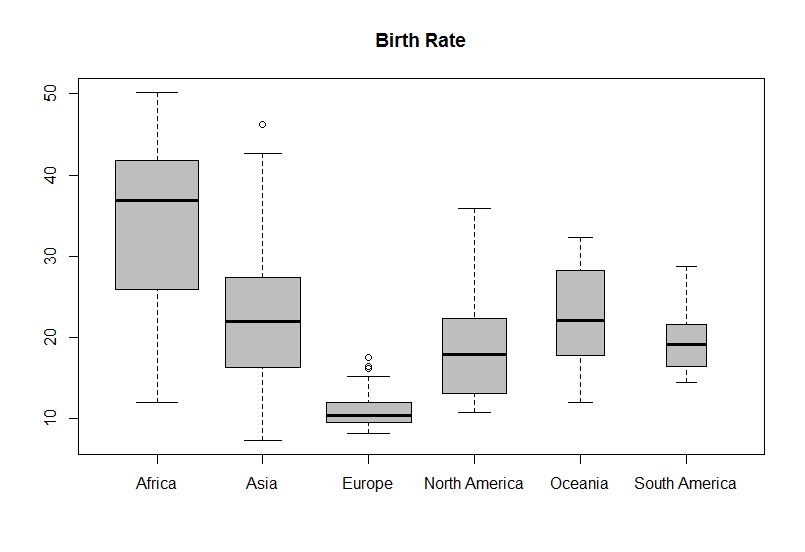
boxplot(Literacy~Continent, data=C, varwidth=TRUE, col="grey", main=Literacy)



## Birth rate as a function of continent

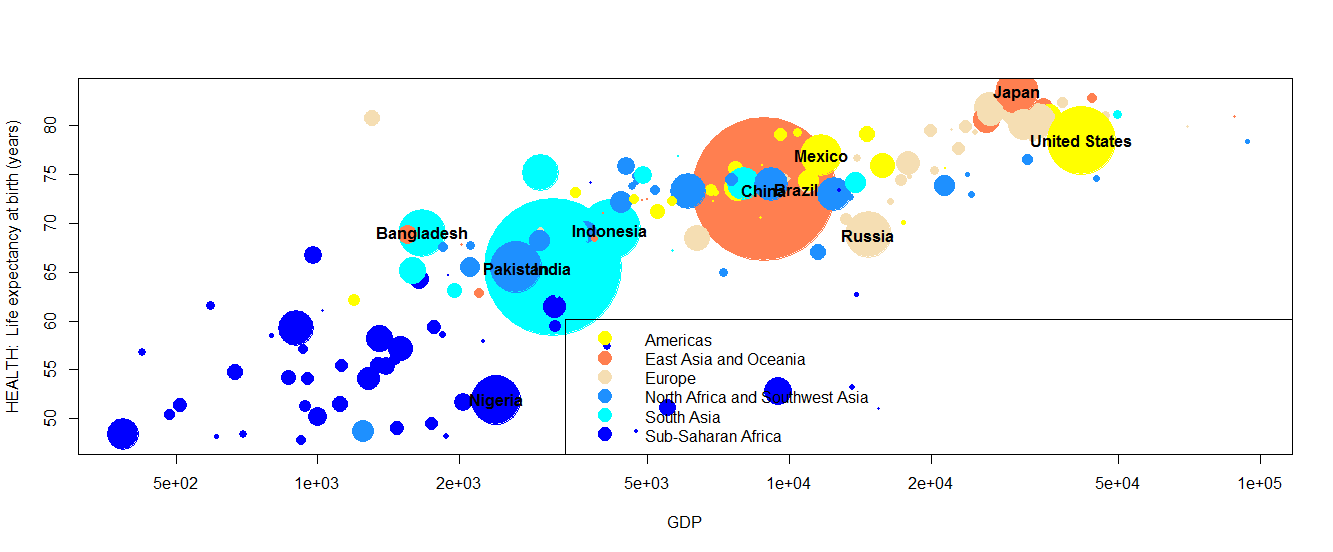
aggregate(C$Birthrate, list(C$Continent), Skew)

boxplot(Birthrate~Continent, data=C, varwidth=TRUE, col="grey", main=”Birth Rate”)



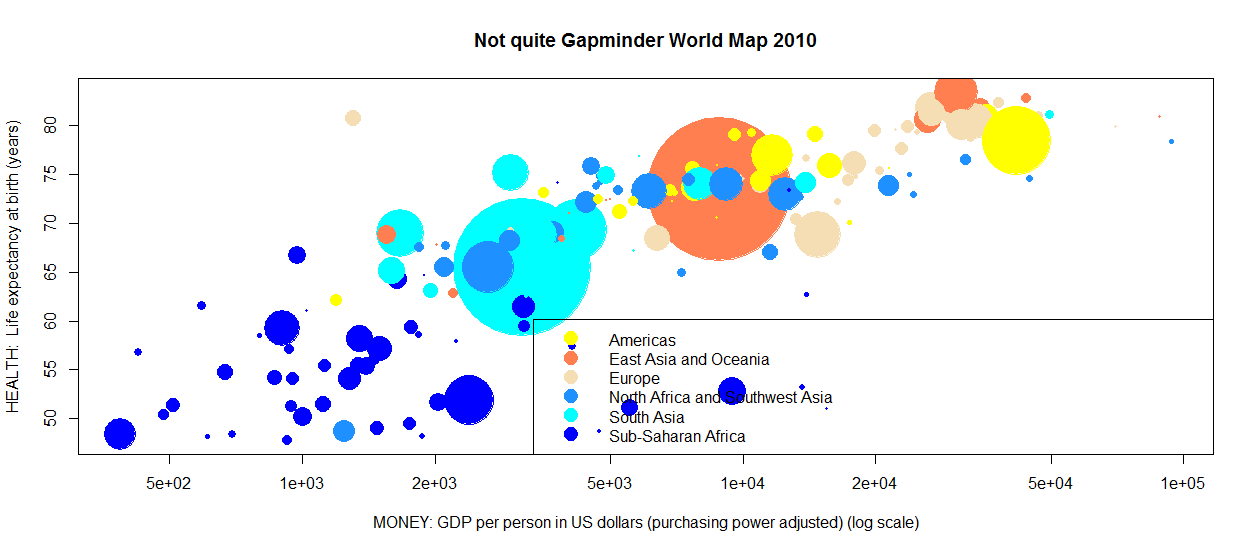
# Copy Gapminder World Map 2010

## The first iteration – hits the high points



This version labels the large countries. I don’t like the size of the legend, and can’t find a way to change it. It also lacks xlab, ylab, and main labels.

## The second iteration – adds xlab, ylab, and main labels.



Addition of xlab, ylab, and main wipes out the large country labels.

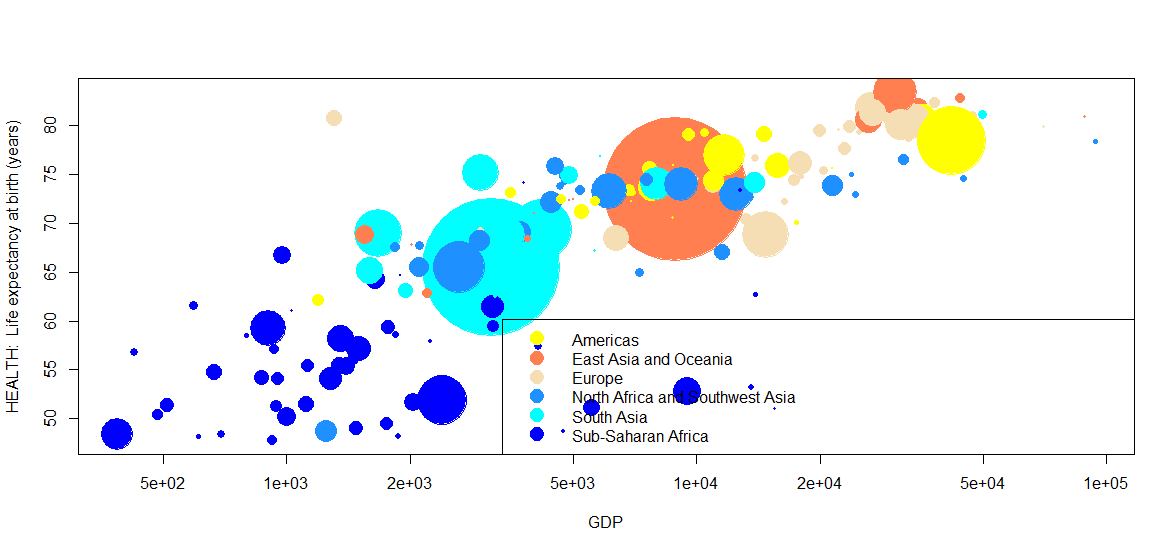
cols = c("yellow", "coral", "wheat", "dodgerblue", "cyan", "blue")

plot(GDP, LifeExpectancy, log="x", main="Not quite Gapminder World Map 2010", xlab="MONEY: GDP per person in US dollars (purchasing power adjusted) (log scale)", ylab="HEALTH: Life expectancy at birth (years)",cex=sqrt(Population/max(Population, na.rm=TRUE))\*20, pch=19, col=cols[match(Region2, levels(Region2))])

legend("bottomright", legend=levels(Region2), col=cols, pch=19, pt.cex=2)

text(labels = Country[Population > 10^8], x = GDP[Population > 10^8], y = LifeExpectancy[Population > 10^8], font=2)

## The third iteration – try to label all countries



This didn’t work either.

## Summary:

* I need a good graphics reference for R. (The help(plot) page does not cover pch codes.)
* I have R In a Nutshell on my Kindle, but the Kindle sucks for reference usage.
* I ordered a paper copy of the new edition of R In a Nutshell, which ships this week.

## Question:

I find that the way that seems to work best for editing large graphics is to have the R code in a text editor then copy from the editor to the R console. Does that sound about right?