Jason Cessna

3/21/2017

**Case Study One: An Analysis on the Gross Domestic Product and Income of Countries in 2012**

**Introduction**:

This is a statistical analysis conducted on the Gross Domestic Product of countries in 2012 in direct relation to Education Data we have received which denotes the income level of countries. We will combine these data together to compare the relationship between the ranking of countries in terms of their GDP (gross domestic product) and their income grouping levels. To perform this analysis we will use R. The data files we will be using for analysis can be found here:

Gross Domestic Product of 2012:

https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2FGDP.csv

Eudcational Data:

https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2FEDSTATS\_Country.csv

**Methods**:

We will start our analysis by installing the proper packages into R (ggplot2, dplyr, and doBy). We will need these for our statistical analysis and visualization of our data.

Then we download the data.

Now we need to clean our data to make sure there is no unncessary blank spaces that would interfere with our analysis. We are also preparing to merge the two data files on their short code (or country code) so we have all relevant information for each country provided in both documents.

Now that our data is clean, we need to merge the two data frames together. In order to do so, we must make sure that the numerical data from our gross domestic product data frame is converted properly so the statistical analysis can be executed properly after the merge.

Now that our data is merged, we can perform statistical analysis on it. The questions we are looking to answer are as follows:

**Results**:

Merge the data based on the country shortcode. How many of the IDs match?

Since we already merged the data based on the country shortcode, we are just looking to discover how many matches we had in total (ie: how many countries have valid data for our analysis?)

Out of the 190 country codes in our Gross Domestic Product data and the 234 country codes in the Educational Data, we are able to match and merge on 189 total country codes.

Sort the data frame in ascending order by GDP (so United States is last). What is the 13th

country in the resulting data frame?

The country with the 13th-lowest generated domestic product is St. Kitts and Nevis

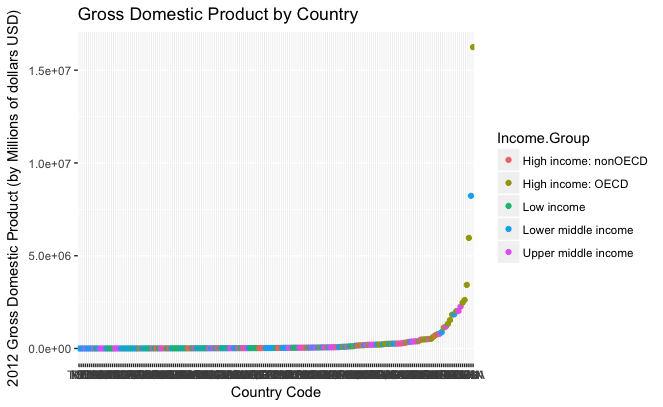
What are the average GDP rankings for the "High income: OECD" and "High income:

nonOECD" groups?

The average (or mean) of the ranking of countries based on their GDP in the High Income: OECD income group is 32.97, and the mean of the rankings of countries based on their GDP in the High Income: nonOECD income group is 91.91.

Show the distribution of GDP value for all the countries and color plots by income group. Use

ggplot2 to create your plot.



Provide summary statistics of GDP by income groups.

Cut the GDP ranking into 5 separate quantile groups. Make a table versus Income.Group.

How many countries are Lower middle income but among the 38 nations with highest

GDP?

There are 16 countries in both the "lower middle income"" income group and the first quantile (top 38 countries) for Gross Domestic Product produced in 2012. This begs the question: how are these countries utilyzing their internal resources in terms of the betterment of their economy? What sorts of resources are produced to make up their GDP? What is the pay which the residents within that country receive while working to produce these resources (if any)? Our data alone cannot answer these questions, but we do have grounds for further investigation based on our results.

**Discussion**:

The analysis we have conducted on these data has showed us a qualitative comparison between countries based on their income groups, rankings, and gross domestic product. We did not find that there is a constant correlation between a high GDP and high income within countries and raises more questions that we can perform additional analysis on if we can obtain more data (ie: products produced, services sold, mean wages for citizens in those jobs, and mean populations for each country) like the correlation between population's wages of a certain industry and the overall profits which the country receives on them.

We also see (in our plot) that only 3 countries have a GDP over 5 trillion in 2012, one of which had an income that places that country in the lower-middle income group. Less than 25% of countries had a GDP of above 1 trillion USD in 2012, and their incomes vary tremendously as well.

With both of these observations, we can conclude that there is not a direct correlation between the gross domestic product of a country and their overall income.

**Appendix**:

Question 1:

##### QUESTION ONE:Merge the data based on the country shortcode. How many of the IDs match?

##### I USED THE NAME "CountryCode" TO DEFINE THE FIRST ROW OF "grossDomProduct" DATA FRAME

##### IN ORDER TO MATCH THE 'SHORT CODE' FROM BOTH DATA FRAMES AND MERGE ONTO IT

## CHECK INITIAL LENGTH BEFORE MERGE IN grossDomProduct

length(grossDomProduct$CountryCode)

## RESULTS:

## [1] 190

## CHECK INITIAL LENGTH BEFORE MERGE IN educationData

length(educationData$CountryCode)

## RESULTS:

## [1] 234

## CHECK LENGTH POST MERGE

length((productAndEducation$CountryCode))

## RESULTS:

## [1] 189

#### THE ANSWER TO QUESTION ONE IS OUT OF THE 190 COUNTRY CODE IDs IN grossDomProduct

#### AND THE 234 COUNTRY CODE IDs IN educationData, THERE ARE 189 MATCHES

Question 2:

#### QUESTION TWO: Sort the data frame in ascending order by GDP (so United States is last).

#### What is the 13th country in the resulting data frame?

## SORT MERGED DATA BY GROSS DOMESTIC PRODUCT ASCENDING

productAndEducation<-productAndEducation[order(productAndEducation$GrossDomesticProduct),]

## CHECK TO MAKE SURE SORT OPERATION WAS SUCCSESSFUL

productAndEducation$GrossDomesticProduct

## RESULTS:

## [1] 40 175 182 228 263 326 472 480 596 684

## [11] 713 767 767 787 822 917 1008 1129 1134 1239

## [21] 1293 1493 1734 1780 1827 2184 2222 2448 2472 2584

## [31] 2851 3092 3744 3796 3814 3908 4199 4225 4264 4373

## [41] 5012 5474 5632 6075 6445 6475 6773 6972 7103 7253

## [51] 7557 7843 8149 8722 9418 9613 9802 9951 9975 10271

## [61] 10308 10441 10486 10507 12648 12887 13072 13579 13678 14038

## [71] 14046 14244 14504 14755 15654 15747 16954 17204 17466 17697

## [81] 18377 18434 18963 19881 20497 20678 22390 22767 23320 23864

## [91] 24680 25322 25502 27035 28242 28373 29044 31015 35164 35646

## [101] 36253 37489 40697 40711 41605 42344 42945 43582 45104 45279

## [111] 45662 49920 50234 50972 51113 55178 58769 59047 59228 59423

## [121] 63267 66605 68234 69972 73672 84040 91149 95982 101496 114147

## [131] 116355 124600 155820 160913 167347 171476 176309 192711 196446 203521

## [141] 203790 205789 210280 210771 212274 225143 247546 249099 250182 258217

## [151] 262597 262832 263259 269869 274701 305033 314887 348595 365966 369606

## [161] 381286 384313 394708 475502 483262 489795 499667 514060 523806 631173

## [171] 711050 770555 789257 878043 1129598 1178126 1322965 1532408 1821424 1841710

## [181] 2014670 2014775 2252664 2471784 2612878 3428131 5959718 8227103 16244600

## GET THE 13TH COUNTRY NAME IN THE DATA FRAME

productAndEducation$LongName[13]

## RESULT:

## [1] "St. Kitts and Nevis"

#### THE ANSWER TO QUESTION TWO IS 'St. Kitts and Nevis'

Question 3:

#### QUESTION THREE: What are the average GDP rankings for the "High income: OECD"

#### and "High income:nonOECD" groups?

## FIND THE MEAN OF RANKS IN THE DATA FRAME IF THE INCOME GROUP EQUALS 'HIGH INCOME: OECD'

mean(productAndEducation$Ranking[productAndEducation$Income.Group == "High income: OECD"])

## RESULTS:

## [1] 32.96667

## FIND THE MEAN OF RANKS IN THE DATA FRAME IF THE INCOME GROUP EQUALS 'HIGH INCOME: nonOECD'

mean(productAndEducation$Ranking[productAndEducation$Income.Group == "High income: nonOECD"])

## RESULTS:

## [1] 91.91304

#### THE ANSWER TO QUESTION THREE IS:

#### High income: OECD: 32.97

#### High income: nonOECD: 91.91

Question 4:

#### QUESTION 4: Show the distribution of GDP value for all the countries and color plots

#### by income group. Use ggplot2 to create your plot.

## LOAD LIBRARIES

library(ggplot2)

library(scales)

## CREATE ELEMENT TO HOLD ALL INCOME GROUP NAMES

allIncomeGroups <- unique(productAndEducation[c("Income.Group")])

## CHECK DATA FRAME

allIncomeGroups

## RESULTS:

## Income.Group

## 173 Lower middle income

## 137 Upper middle income

## 39 Low income

## 1 High income: nonOECD

## 82 High income: OECD

## CREATE GG PLOT

gdpPlot <- ggplot(data = productAndEducation, aes(reorder(CountryCode,GrossDomesticProduct), y=GrossDomesticProduct))

## MAKE SURE THE RESULTS ARE IN MILLIONS OF DOLLARS AND BYPASS SCIENTIFIC NOTATION FOR LEGIBILITY

gdpPlot + geom\_point(aes(color = Income.Group)) + labs(title="Gross Domestic Product by Country", y="2012 Gross Domestic Product (by Millions of dollars USD)", x= "Country Code") + scale\_y\_continuous(labels = comma, breaks=c(1000000,5000000,10000000,15000000))

## RESULTS:

Question 5:

#### QUESTION 5: Provide summary statistics of GDP by income groups.

## LOAD LIBRARIES

library(dplyr)

library(doBy)

## BEGIN SUMMARY STATISTICS

tapply(productAndEducation$GrossDomesticProduct, productAndEducation$Income.Group, summary)

## RESULTS IN ANSWER BELOW

## GET SPECIFIC STATISTICS

summaryBy(GrossDomesticProduct ~ Income.Group, data = productAndEducation, FUN = function(x) { c(mean = mean(x), stdev = sd(x), sum=sum(x), count=length(x))})

## RESULTS IN ANSWER BELOW:

#### ANSWER TO QUESTION 5:

#### SUMMARY STATISTICS FOR EACH INCOME GROUP:

#### $`High income: nonOECD`

#### Min. 1st Qu. Median Mean 3rd Qu. Max.

#### 2584 12840 28370 104300 131200 711000

#### $`High income: OECD`

#### Min. 1st Qu. Median Mean 3rd Qu. Max.

#### 13580 211100 486500 1484000 1480000 16240000

#### $`Low income`

#### Min. 1st Qu. Median Mean 3rd Qu. Max.

#### 596 3814 7843 14410 17200 116400

#### $`Lower middle income`

#### Min. 1st Qu. Median Mean 3rd Qu. Max.

#### 40 2549 24270 256700 81450 8227000

#### $`Upper middle income`

#### Min. 1st Qu. Median Mean 3rd Qu. Max.

#### 228 9613 42940 231800 205800 2253000

####

#### SPECIFIC STATISTICS FOR THE GROUPS:

#### INCOME GROUP MEAN STANDARD DEVIATION SUM TOTAL(count of countries in group)

#### High income: nonOECD 104349.83 165334.45 2400046 23

#### High income: OECD 1483917.13 3070463.52 44517514 30

#### Low income 14410.78 20473.09 533199 37

#### Lower middle income 256663.48 1139619.92 13859828 54

#### Upper middle income 231847.84 476872.04 10433153 45

Question 6:

#### QUESTION 6: Cut the GDP ranking into 5 separate quantile groups.

#### Make a table versus Income.Group. How many countries are Lower middle

#### income but among the 38 nations with highest GDP?

## CREATE 5 QUANTILES IN A NEW DATAFRAME BASED ON GROSS DOMESTIC PRODUCT AND INCLUDE INCOME GROUPS

GDPE <- data.frame(ntile(productAndEducation$GrossDomesticProduct,n=5),productAndEducation$GrossDomesticProduct,productAndEducation$Income.Group)

## NAME DATA FRAME COLUMNS FOR DISPLAY

names(GDPE)<-c("Quantile","GrossDomesticProduct","IncomeGroup")

##CREATE A TABLE WITH QUANTILES VS INCOME GROUPS

xtabs(~GDPE$Quantile+GDPE$IncomeGroup)

## RESULTS:

## GDPE$IncomeGroup

## GDPE$Quantile High income: nonOECD High income: OECD Low income Lower middle income

## 1 0 2 0 11 16

## 2 0 4 1 16 9

## 3 0 8 1 9 11

## 4 0 5 10 1 14

## 5 0 4 18 0 4

## GDPE$IncomeGroup

## GDPE$Quantile Upper middle income

## 1 9

## 2 8

## 3 9

## 4 8

## 5 11

#### ANSWER TO QUESITON SIX: There are 16 countries in the lower middle income

#### but the first quantile (top 38 countries) for Gross Domestic Product

Install Packages:

## INSTALL ggplot2 FOR GGPLOT2 AND SCALES AND dplyr AND doBy FOR STATISTICAL ANALYSIS

install.packages('ggplot2',repos = "http://cran.us.r-project.org")

install.packages('doBy',repos = "http://cran.us.r-project.org")

install.packages('dplyr',repos = "http://cran.us.r-project.org")

Download data:

#IMPORT FILES FROM URL AND READ THEM INTO R AS DATA FRAMES

grossDomProduct<-read.csv('https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2FGDP.csv', header=FALSE, sep=",", stringsAsFactors=FALSE, skip=5)

educationData<-read.csv('https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2FEDSTATS\_Country.csv', header=TRUE, sep=",", stringsAsFactors=TRUE)

Clean Data:

## CLEANING THE DATA

## COUNT ALL 'NA'S IN CSVs

countNA <- function(x) {

count <- vector()

for (name in names(x)) {

count<-append(count, sapply(x[name], function(y) sum(length(which(is.na(y)))))) #sums number of Nas for each column

}

return(count)

}

## COUNT '..' IN CSVs

countElipse<-function(x){

count <- vector()

for (name in names(x)) {

count<-append(count, sapply(x[name], function(y) sum(length(which(is.element('..', y))))))

}

return(count)

}

## COUNT BLANK ROWS IN COLUMNS

countBlank<-function(x){

count <- vector()

for (name in names(x)) {

count<-append(count, sapply(x[name], function(y) sum(length(which(is.element('', y))))))

}

return(count)

}

## FUNCTION TO READ EACH PREVIOUS FUNCTION FOR EASE OF DIAGNOSTICS

checkBadValues <- function(x) {

countNA <- countNA(x)

countBlank<-countBlank(x)

countElipse<-countElipse(x)

return(data.frame(countNA, countBlank, countElipse))

}

## RUN FUNCTION

checkBadValues(grossDomProduct)

## RESULTS:

## countNA countBlank countElipse

## V1 0 1 0

## V2 0 1 0

## V3 326 0 0

## V4 0 1 0

## V5 0 1 1

## V6 0 1 0

## V7 326 0 0

## V8 326 0 0

## V9 326 0 0

## V10 326 0 0

##

## WE NEED TO GET RID OF V3, V6, V7, V8, AND V9

## DELETE SPECIFIED COLUMNS FROM DATA FRAME

grossDomProduct$V3 <- NULL

grossDomProduct$V6 <- NULL

grossDomProduct$V7 <- NULL

grossDomProduct$V8 <- NULL

grossDomProduct$V9 <- NULL

grossDomProduct$V10 <- NULL

## RENAME COLUMNS FROM grossDomProduct TO MATCH THOSE IN educationData AND OTHER RELEVANT NAMES

names(grossDomProduct)[1] <- "CountryCode"

names(grossDomProduct)[2] <- "Ranking"

names(grossDomProduct)[3] <- "LongName"

names(grossDomProduct)[4] <- "GrossDomesticProduct"

## CHECK VALUES FOR educationData

checkBadValues(educationData)

## RESULTS:

## CountryCode 0 0 0

## Long.Name 0 0 0

## Income.Group 0 1 0

## Region 0 1 0

## Lending.category 0 1 0

## Other.groups 0 1 0

## Currency.Unit 0 1 0

## Latest.population.census 0 1 0

## Latest.household.survey 0 1 0

## Special.Notes 0 1 0

## National.accounts.base.year 0 1 0

## National.accounts.reference.year 197 0 0

## System.of.National.Accounts 149 0 0

## SNA.price.valuation 0 1 0

## Alternative.conversion.factor 0 1 0

## PPP.survey.year 89 0 0

## Balance.of.Payments.Manual.in.use 0 1 0

## External.debt.Reporting.status 0 1 0

## System.of.trade 0 1 0

## Government.Accounting.concept 0 1 0

## IMF.data.dissemination.standard 0 1 0

## Source.of.most.recent.Income.and.expenditure.data 0 1 0

## Vital.registration.complete 0 1 0

## Latest.agricultural.census 0 1 0

## Latest.industrial.data 139 0 0

## Latest.trade.data 46 0 0

## Latest.water.withdrawal.data 82 0 0

## X2.alpha.code 1 1 0

## WB.2.code 1 1 0

## Table.Name 0 0 0

## Short.Name 0 0 0

## KEEP ONLY COLUMNS WE NEED:

educationData<-educationData[c(1,2,3,31)]

## CHECK TO MAKE SURE PROPER COLUMNS ARE STILL AVAILABLE:

names(educationData)

## RESULTS:

## [1] "CountryCode" "Long.Name" "Income.Group" "Short.Name"

## SECOND CHECK FOR VALID COLUMNS AND ROWS

checkBadValues(grossDomProduct)

## RESULTS:

## countNA countBlank countElipse

## CountryCode 0 1 0

## Ranking 0 1 0

## LongName 0 1 0

## GrossDomesticProduct 0 1 1

## BLANKS AND ELIPSE MAY BE THE RESULT OF EXCESS ROWS

checkBadValues(educationData)

## RESULTS:

## countNA countBlank countElipse

## CountryCode 0 0 0

## Long.Name 0 0 0

## Income.Group 0 1 0

## Short.Name 0 0 0

## BLANK ROW IN Income.Group IS OK AS IT IS NON-NUMERIC

## CLIP EXCESS ROWS IN grossDomProduct

grossDomProduct <- grossDomProduct[-c(191:330),]

## FINAL CHECK FOR CLEAN DATA

checkBadValues(grossDomProduct)

## RESULTS:

## countNA countBlank countElipse

## CountryCode 0 0 0

## Ranking 0 0 0

## LongName 0 0 0

## GrossDomesticProduct 0 0 0

## DATA HAS BEEN CLEANED!

Merge Data:

## PREP THE DATA FOR THE MERGE AND MERGE THE DATA

## CHECK CAPABILITY FOR CONVERTING THE 'Ranking' ROW IN grossDomProduct FOR

## NUMERIC CONVERSION

suppressWarnings(as.numeric(grossDomProduct$Ranking))

## RESULTS:

## [1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23

## [24] 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46

## [47] 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69

## [70] 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92

## [93] 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115

## [116] 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138

## [139] 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161

## [162] 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 178 180 181 182 183 184

## [185] 185 186 187 188 189 190

## ALL ROWS READY FOR CONVERSION

## CONVERT RANKINGS

grossDomProduct$Ranking <- as.numeric(grossDomProduct$Ranking)

## CHECK FOR INCOMPATIBLE ROWS IN 'GrossDomesticProduct' IN grossDomProduct

suppressWarnings(as.numeric(grossDomProduct$GrossDomesticProduct))

## RESULTS:

## [1] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA

## [24] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA

## [47] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA

## [70] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA

## [93] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA

## [116] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA

## [139] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA

## [162] NA NA NA NA NA NA NA NA NA NA NA NA NA 917 822 787 767 767 713 684 596 480 472

## [185] 326 263 228 182 175 40

## DIAGNOSE PROBLEM WITH THE COLUMN

head(grossDomProduct$GrossDomesticProduct)

## RESULTS:

## [1] " 16,244,600 " " 8,227,103 " " 5,959,718 " " 3,428,131 " " 2,612,878 " " 2,471,784 "

## WE HAVE COMMAS AND SPACES THAT NEED TO BE SUBSTITUDED TO NULL BEFORE WE CAN CONVERT

## CHECK FOR INCOMPATIBLE ROWS IF WE CONVERT COMMA TO NULL

suppressWarnings(as.numeric(",","",grossDomProduct$GrossDomesticProduct))

## RESULTS:

## [1] NA

## COMMA ALONE ISN'T ENOUGH FOR CONVERSION, NEED TO SUBSTITUDE BOTH COMMAS AND SPACES TO NULL

## SUB COMMAS TO NULL

grossDomProduct$GrossDomesticProduct<-gsub(',','',grossDomProduct$GrossDomesticProduct)

## SUB BLANK CHARACTERS TO NULL

grossDomProduct$GrossDomesticProduct<-gsub(' ','',grossDomProduct$GrossDomesticProduct)

## CONFIRM SUBSTITUTION

head(grossDomProduct$GrossDomesticProduct)

## RESULTS:

## [1] "16244600" "8227103" "5959718" "3428131" "2612878" "2471784"

## SECOND CHECK FOR INCOMPATIBLE ROWS IN GROSS DOMESTIC PRODUCT

suppressWarnings(as.numeric(grossDomProduct$GrossDomesticProduct))

## RESULTS:

## [1] 16244600 8227103 5959718 3428131 2612878 2471784 2252664 2014775 2014670 1841710

## [11] 1821424 1532408 1322965 1178126 1129598 878043 789257 770555 711050 631173

## [21] 523806 514060 499667 489795 483262 475502 394708 384313 381286 369606

## [31] 365966 348595 314887 305033 274701 269869 263259 262832 262597 258217

## [41] 250182 249099 247546 225143 212274 210771 210280 205789 203790 203521

## [51] 196446 192711 176309 171476 167347 160913 155820 124600 116355 114147

## [61] 101496 95982 91149 84040 73672 69972 68234 66605 63267 59423

## [71] 59228 59047 58769 55178 51113 50972 50234 49920 45662 45279

## [81] 45104 43582 42945 42344 41605 40711 40697 37489 36253 35646

## [91] 35164 31015 29044 28373 28242 27035 25502 25322 24680 23864

## [101] 23320 22767 22390 20678 20497 19881 18963 18434 18377 17697

## [111] 17466 17204 16954 15747 15654 14755 14504 14244 14046 14038

## [121] 13678 13579 13072 12887 12648 10507 10486 10441 10308 10271

## [131] 10220 9975 9951 9802 9613 9418 8722 8149 7843 7557

## [141] 7253 7103 6972 6773 6475 6445 6075 5632 5474 5012

## [151] 4373 4264 4225 4199 3908 3814 3796 3744 3092 2851

## [161] 2584 2472 2448 2222 2184 1827 1780 1734 1493 1293

## [171] 1239 1134 1129 1008 917 822 787 767 767 713

## [181] 684 596 480 472 326 263 228 182 175 40

## ALL ROWS READY FOR CONVERSION

## CONVERT TO NUMERIC

grossDomProduct$GrossDomesticProduct<-(as.numeric(grossDomProduct$GrossDomesticProduct))

## FINAL CHECK FOR CLEAN AND NUMERIC DATA BEFORE MERGE

head(grossDomProduct$GrossDomesticProduct)

## RESULTS:

## [1] "16244600" "8227103" "5959718" "3428131" "2612878" "2471784"

checkBadValues(grossDomProduct)

## RESULTS:

## countNA countBlank countElipse

## CountryCode 0 0 0

## Ranking 0 0 0

## LongName 0 0 0

## GrossDomesticProduct 0 0 0

## CONVERSIONS SUCCESSFUL! READY FOR MERGE

## MERGE GROSS DOMESTIC PRODUCT AND EDUCATION DATA

productAndEducation <- merge(grossDomProduct, educationData, by="CountryCode", all=FALSE)

## CHECK FOR ERRORS IN MERGED DATA AND GET A SUMMARY

summary(productAndEducation)

str(productAndEducation)

checkBadValues(productAndEducation)

## RESULTS:

## countNA countBlank countElipse

## CountryCode 0 0 0

## Ranking 0 0 0

## LongName 0 0 0

## GrossDomesticProduct 0 0 0

## Long.Name 0 0 0

## Income.Group 0 0 0

## Short.Name 0 0 0

## DATA HAS SUCCESSFULLY MERGED WITH THE MEANS TO CONTINUE ANALYSIS