Data Cleaning Methodology for Case Study 1

Downloading and cleaning the data

Step 1: Install and load necessary packages:

To start, load downloader, repmis and RCurl. Other packages will be added as necessary.

library(repmis)

library(RCurl)

## Loading required package: bitops

library(downloader)

Download and create tidy data sets

The steps for creating tidy data are listed below.

First the data is saved as CSV files. Next, the CSV’s are loaded into a data frame. Once the data is in a data frame, remove missing rows and columns. Next, it is wise to rename the columns using meaningful names.

It is important to scan your columns and make sure the data types are in the format needed for the analysis you intend to run. Convert any character data types to numeric and factors as you see fit. In the analysis below, I converted both Economy and Rank to numeric fields. Notice how I removed the commas from the Economy attributes prior to converting the data. The commas were being factored into the data conversion and produced incorrect results when converting to numeric data.

Source URL’s are https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2FGDP.csv

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

download("https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2FGDP.csv", destfile="gdp.csv")

download("https://d396qusza40orc.cloudfront.net/getdata%2Fdata%2FEDSTATS\_Country.csv", destfile="educ.csv")

list.files()

## [1] "Analysis" "CaseStudy1.Rproj"

## [3] "CaseStudy1\_Output.html" "CaseStudy1\_Output.Rmd"

## [5] "educ.csv" "gdp.csv"

## [7] "Raw\_Data" "README.md"

# Load gdp data into gdpraw dataframe

gdpraw <-read.csv("gdp.csv", stringsAsFactors = FALSE, header = FALSE)

str(gdpraw)

## 'data.frame': 331 obs. of 10 variables:

## $ V1 : chr "" "" "" "" ...

## $ V2 : chr "Gross domestic product 2012" "" "" "Ranking" ...

## $ V3 : logi NA NA NA NA NA NA ...

## $ V4 : chr "" "" "" "Economy" ...

## $ V5 : chr "" "" "(millions of" "US dollars)" ...

## $ V6 : chr "" "" "" "" ...

## $ V7 : logi NA NA NA NA NA NA ...

## $ V8 : logi NA NA NA NA NA NA ...

## $ V9 : logi NA NA NA NA NA NA ...

## $ V10: logi NA NA NA NA NA NA ...

dim(gdpraw)

## [1] 331 10

head(gdpraw)

## V1 V2 V3 V4 V5 V6 V7 V8

## 1 Gross domestic product 2012 NA NA NA

## 2 NA NA NA

## 3 NA (millions of NA NA

## 4 Ranking NA Economy US dollars) NA NA

## 5 NA NA NA

## 6 USA 1 NA United States 16,244,600 NA NA

## V9 V10

## 1 NA NA

## 2 NA NA

## 3 NA NA

## 4 NA NA

## 5 NA NA

## 6 NA NA

tail(gdpraw)

## V1 V2 V3 V4 V5 V6 V7 V8 V9 V10

## 326 NA NA NA NA NA

## 327 NA NA NA NA NA

## 328 NA NA NA NA NA

## 329 NA NA NA NA NA

## 330 NA NA NA NA NA

## 331 NA NA NA NA NA

# Clean out missing data in both rows and columns

gdp <- gdpraw[6:195, c("V1","V2","V4","V5")]

head(gdp)

## V1 V2 V4 V5

## 6 USA 1 United States 16,244,600

## 7 CHN 2 China 8,227,103

## 8 JPN 3 Japan 5,959,718

## 9 DEU 4 Germany 3,428,131

## 10 FRA 5 France 2,612,878

## 11 GBR 6 United Kingdom 2,471,784

# Rename variables to be more meaningful

# Load dplyr

library(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

gdp <- rename(gdp, CountryCode=V1, Rank=V2, CountryFullName=V4, Economy=V5)

#Remove commas from Economy Column - These cause problems during Character to numeric conversion

gdp$Economy <- gsub(",", "", gdp$Economy)

#Convert Economy from Character to Numeric data type

gdp$Economy <- as.numeric(gdp$Economy)

#Convert Rank from Character to Numeric data type

gdp$Rank <- as.integer(gdp$Rank)

head(gdp)

## CountryCode Rank CountryFullName Economy

## 6 USA 1 United States 16244600

## 7 CHN 2 China 8227103

## 8 JPN 3 Japan 5959718

## 9 DEU 4 Germany 3428131

## 10 FRA 5 France 2612878

## 11 GBR 6 United Kingdom 2471784

Load education data into a data frame named educraw

The education data set has 234 rows and 31 columns. For our analysis, we do not need all 31 columns. To proceed with the analysis, the education data will be subset into a new data frame called educ that contains only 2 columns. One column contains the country code that will be used to merge the data set with the gdp data set. The other column, “Income.Group”, will be used to create bins of data further along in our analysis.

A note about NA attributes

Though the education data set has a plethora of NA’s, we will not remove them using the complete.cases or na.omit statements. By omiting the extraneous columns in the data frame, we have effectively removed NA values without removing valuable rows that will be needed to merge our data later on in our analysis.

educraw <-read.csv("educ.csv", stringsAsFactors = FALSE, header = TRUE)

str(educraw)

## 'data.frame': 234 obs. of 31 variables:

## $ CountryCode : chr "ABW" "ADO" "AFG" "AGO" ...

## $ Long.Name : chr "Aruba" "Principality of Andorra" "Islamic State of Afghanistan" "People's Republic of Angola" ...

## $ Income.Group : chr "High income: nonOECD" "High income: nonOECD" "Low income" "Lower middle income" ...

## $ Region : chr "Latin America & Caribbean" "Europe & Central Asia" "South Asia" "Sub-Saharan Africa" ...

## $ Lending.category : chr "" "" "IDA" "IDA" ...

## $ Other.groups : chr "" "" "HIPC" "" ...

## $ Currency.Unit : chr "Aruban florin" "Euro" "Afghan afghani" "Angolan kwanza" ...

## $ Latest.population.census : chr "2000" "Register based" "1979" "1970" ...

## $ Latest.household.survey : chr "" "" "MICS, 2003" "MICS, 2001, MIS, 2006/07" ...

## $ Special.Notes : chr "" "" "Fiscal year end: March 20; reporting period for national accounts data: FY." "" ...

## $ National.accounts.base.year : chr "1995" "" "2002/2003" "1997" ...

## $ National.accounts.reference.year : int NA NA NA NA 1996 NA NA 1996 NA NA ...

## $ System.of.National.Accounts : int NA NA NA NA 1993 NA 1993 1993 NA NA ...

## $ SNA.price.valuation : chr "" "" "VAB" "VAP" ...

## $ Alternative.conversion.factor : chr "" "" "" "1991-96" ...

## $ PPP.survey.year : int NA NA NA 2005 2005 NA 2005 2005 NA NA ...

## $ Balance.of.Payments.Manual.in.use : chr "" "" "" "BPM5" ...

## $ External.debt.Reporting.status : chr "" "" "Actual" "Actual" ...

## $ System.of.trade : chr "Special" "General" "General" "Special" ...

## $ Government.Accounting.concept : chr "" "" "Consolidated" "" ...

## $ IMF.data.dissemination.standard : chr "" "" "GDDS" "GDDS" ...

## $ Source.of.most.recent.Income.and.expenditure.data: chr "" "" "" "IHS, 2000" ...

## $ Vital.registration.complete : chr "" "Yes" "" "" ...

## $ Latest.agricultural.census : chr "" "" "" "1964-65" ...

## $ Latest.industrial.data : int NA NA NA NA 2005 NA 2001 NA NA NA ...

## $ Latest.trade.data : int 2008 2006 2008 1991 2008 2008 2008 2008 NA 2007 ...

## $ Latest.water.withdrawal.data : int NA NA 2000 2000 2000 2005 2000 2000 NA 1990 ...

## $ X2.alpha.code : chr "AW" "AD" "AF" "AO" ...

## $ WB.2.code : chr "AW" "AD" "AF" "AO" ...

## $ Table.Name : chr "Aruba" "Andorra" "Afghanistan" "Angola" ...

## $ Short.Name : chr "Aruba" "Andorra" "Afghanistan" "Angola" ...

dim(educraw)

## [1] 234 31

##Drop NA Values Problem - This removes all but 14 rows. NA's will be removed by omtting extraneous columns.

##educNoNA <- educraw[complete.cases(educraw),]

##head(educNoNA)

#New code to bring in only the pertinent variables for the study

educ <- educraw[, c("CountryCode","Income.Group")]

head(educ)

## CountryCode Income.Group

## 1 ABW High income: nonOECD

## 2 ADO High income: nonOECD

## 3 AFG Low income

## 4 AGO Lower middle income

## 5 ALB Upper middle income

## 6 ARE High income: nonOECD