Reproducible Research - Project - 2

Gabeesh Mishra

10/1/2019

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

summary(cars)

## speed dist   
## Min. : 4.0 Min. : 2.00   
## 1st Qu.:12.0 1st Qu.: 26.00   
## Median :15.0 Median : 36.00   
## Mean :15.4 Mean : 42.98   
## 3rd Qu.:19.0 3rd Qu.: 56.00   
## Max. :25.0 Max. :120.00

## 1: Synopsis

The goal of the assignment is to explore the NOAA Storm Database and explore the effects of severe weather events on both population and economy.The database covers the time period between 1950 and November 2011.

The following analysis investigates which types of severe weather events are most harmful on:

1. Health (injuries and fatalities)
2. Property and crops (economic consequences)

Information on the Data: [Documentation](https://d396qusza40orc.cloudfront.net/repdata%2Fpeer2_doc%2Fpd01016005curr.pdf)

## 2: Data Processing

### 2.1: Data Loading

Download the raw data file and extract the data into a dataframe.Then convert to a data.table

setwd("C:/Data Science - John Hopkins/Data/5-Reproducible Research")  
getwd()

## [1] "C:/Data Science - John Hopkins/Data/5-Reproducible Research"

library("data.table")  
library("ggplot2")  
  
stormDF <- read.csv("C:/Data Science - John Hopkins/Data/5-Reproducible Research/repdata\_data\_StormData.csv")  
# Converting data.frame to data.table  
stormDT <- as.data.table(stormDF)

### 2.2: Examining Column Names

colnames(stormDT)

## [1] "STATE\_\_" "BGN\_DATE" "BGN\_TIME" "TIME\_ZONE" "COUNTY"   
## [6] "COUNTYNAME" "STATE" "EVTYPE" "BGN\_RANGE" "BGN\_AZI"   
## [11] "BGN\_LOCATI" "END\_DATE" "END\_TIME" "COUNTY\_END" "COUNTYENDN"  
## [16] "END\_RANGE" "END\_AZI" "END\_LOCATI" "LENGTH" "WIDTH"   
## [21] "F" "MAG" "FATALITIES" "INJURIES" "PROPDMG"   
## [26] "PROPDMGEXP" "CROPDMG" "CROPDMGEXP" "WFO" "STATEOFFIC"  
## [31] "ZONENAMES" "LATITUDE" "LONGITUDE" "LATITUDE\_E" "LONGITUDE\_"  
## [36] "REMARKS" "REFNUM"

### 2.3: Data Subsetting

Subset the dataset on the parameters of interest. Basically, we remove the columns we don’t need for clarity.

# Finding columns to remove  
cols2Remove <- colnames(stormDT[, !c("EVTYPE"  
 , "FATALITIES"  
 , "INJURIES"  
 , "PROPDMG"  
 , "PROPDMGEXP"  
 , "CROPDMG"  
 , "CROPDMGEXP")])  
# Removing columns  
stormDT[, c(cols2Remove) := NULL]  
# Only use data where fatalities or injuries occurred.   
stormDT <- stormDT[(EVTYPE != "?" &   
 (INJURIES > 0 | FATALITIES > 0 | PROPDMG > 0 | CROPDMG > 0)), c("EVTYPE"  
 , "FATALITIES"  
 , "INJURIES"  
 , "PROPDMG"  
 , "PROPDMGEXP"  
 , "CROPDMG"  
 , "CROPDMGEXP") ]

### 2.4: Converting Exponent Columns into Actual Exponents instead of (-,+, H, K, etc)

Making the PROPDMGEXP and CROPDMGEXP columns cleaner so they can be used to calculate property and crop cost.

# Change all damage exponents to uppercase.  
cols <- c("PROPDMGEXP", "CROPDMGEXP")  
stormDT[, (cols) := c(lapply(.SD, toupper)), .SDcols = cols]  
# Map property damage alphanumeric exponents to numeric values.  
propDmgKey <- c("\"\"" = 10^0,  
 "-" = 10^0,   
 "+" = 10^0,  
 "0" = 10^0,  
 "1" = 10^1,  
 "2" = 10^2,  
 "3" = 10^3,  
 "4" = 10^4,  
 "5" = 10^5,  
 "6" = 10^6,  
 "7" = 10^7,  
 "8" = 10^8,  
 "9" = 10^9,  
 "H" = 10^2,  
 "K" = 10^3,  
 "M" = 10^6,  
 "B" = 10^9)  
# Map crop damage alphanumeric exponents to numeric values  
cropDmgKey <- c("\"\"" = 10^0,  
 "?" = 10^0,   
 "0" = 10^0,  
 "K" = 10^3,  
 "M" = 10^6,  
 "B" = 10^9)  
stormDT[, PROPDMGEXP := propDmgKey[as.character(stormDT[,PROPDMGEXP])]]  
stormDT[is.na(PROPDMGEXP), PROPDMGEXP := 10^0 ]  
stormDT[, CROPDMGEXP := cropDmgKey[as.character(stormDT[,CROPDMGEXP])] ]  
stormDT[is.na(CROPDMGEXP), CROPDMGEXP := 10^0 ]

### 2.5: Making Economic Cost Columns

stormDT <- stormDT[, .(EVTYPE, FATALITIES, INJURIES, PROPDMG, PROPDMGEXP, propCost = PROPDMG \* PROPDMGEXP, CROPDMG, CROPDMGEXP, cropCost = CROPDMG \* CROPDMGEXP)]

### 2.6: Calcuating Total Property and Crop Cost

totalCostDT <- stormDT[, .(propCost = sum(propCost), cropCost = sum(cropCost), Total\_Cost = sum(propCost) + sum(cropCost)), by = .(EVTYPE)]  
totalCostDT <- totalCostDT[order(-Total\_Cost), ]  
totalCostDT <- totalCostDT[1:10, ]  
head(totalCostDT, 5)

## EVTYPE propCost cropCost Total\_Cost  
## 1: FLOOD 144657709807 5661968450 150319678257  
## 2: HURRICANE/TYPHOON 69305840000 2607872800 71913712800  
## 3: TORNADO 56947380677 414953270 57362333947  
## 4: STORM SURGE 43323536000 5000 43323541000  
## 5: HAIL 15735267513 3025954473 18761221986

### 2.7: Calcuating Total Fatalities and Injuries

totalInjuriesDT <- stormDT[, .(FATALITIES = sum(FATALITIES), INJURIES = sum(INJURIES), totals = sum(FATALITIES) + sum(INJURIES)), by = .(EVTYPE)]  
totalInjuriesDT <- totalInjuriesDT[order(-FATALITIES), ]  
totalInjuriesDT <- totalInjuriesDT[1:10, ]  
head(totalInjuriesDT, 5)

## EVTYPE FATALITIES INJURIES totals  
## 1: TORNADO 5633 91346 96979  
## 2: EXCESSIVE HEAT 1903 6525 8428  
## 3: FLASH FLOOD 978 1777 2755  
## 4: HEAT 937 2100 3037  
## 5: LIGHTNING 816 5230 6046

## 3: Results

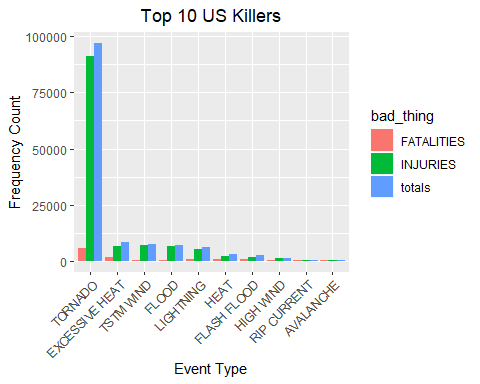
### 3.1: Events that are Most Harmful to Population Health

Melting data.table so that it is easier to put in bar graph format

bad\_stuff <- melt(totalInjuriesDT, id.vars="EVTYPE", variable.name = "bad\_thing")  
head(bad\_stuff, 5)

## EVTYPE bad\_thing value  
## 1: TORNADO FATALITIES 5633  
## 2: EXCESSIVE HEAT FATALITIES 1903  
## 3: FLASH FLOOD FATALITIES 978  
## 4: HEAT FATALITIES 937  
## 5: LIGHTNING FATALITIES 816

# Create chart  
healthChart <- ggplot(bad\_stuff, aes(x=reorder(EVTYPE, -value), y=value))  
# Plot data as bar chart  
healthChart = healthChart + geom\_bar(stat="identity", aes(fill=bad\_thing), position="dodge")  
# Format y-axis scale and set y-axis label  
healthChart = healthChart + ylab("Frequency Count")   
# Set x-axis label  
healthChart = healthChart + xlab("Event Type")   
# Rotate x-axis tick labels   
healthChart = healthChart + theme(axis.text.x = element\_text(angle=45, hjust=1))  
# Set chart title and center it  
healthChart = healthChart + ggtitle("Top 10 US Killers") + theme(plot.title = element\_text(hjust = 0.5))  
healthChart



### 3.2: Events that have the Greatest Economic Consequences

Melting data.table so that it is easier to put in bar graph format

econ\_consequences <- melt(totalCostDT, id.vars="EVTYPE", variable.name = "Damage\_Type")  
head(econ\_consequences, 5)

## EVTYPE Damage\_Type value  
## 1: FLOOD propCost 144657709807  
## 2: HURRICANE/TYPHOON propCost 69305840000  
## 3: TORNADO propCost 56947380677  
## 4: STORM SURGE propCost 43323536000  
## 5: HAIL propCost 15735267513

# Create chart  
econChart <- ggplot(econ\_consequences, aes(x=reorder(EVTYPE, -value), y=value))  
# Plot data as bar chart  
econChart = econChart + geom\_bar(stat="identity", aes(fill=Damage\_Type), position="dodge")  
# Format y-axis scale and set y-axis label  
econChart = econChart + ylab("Cost (dollars)")   
# Set x-axis label  
econChart = econChart + xlab("Event Type")   
# Rotate x-axis tick labels   
econChart = econChart + theme(axis.text.x = element\_text(angle=45, hjust=1))  
# Set chart title and center it  
econChart = econChart + ggtitle("Top 10 US Storm Events causing Economic Consequences") + theme(plot.title = element\_text(hjust = 0.5))  
econChart

