Reproducible research: Peer Graded Assingment 2

2023-03-29

1: Synopsis The basic goal of this assignment is to explore the NOAA Storm Database and answer some basic questions about severe weather events. You must use the database to answer the questions below and show the code for your entire analysis. Your analysis can consist of tables, figures, or other summaries. You may use any R package you want to support your analysis.

Questions Your data analysis must address the following questions:

1) Across the United States, which types of events (as indicated in the EVTYPE variable) are most harmful with respect to population health?

2) Across the United States, which types of events have the greatest economic consequences?

2: Data Processing

Extract the data into a dataframe.Then convert to a data.table

```
library("data.table")
library("ggplot2")
setwd("/home/mike/Documents/Learning/ds/")

storm_df <- read.csv("repdata_data_StormData.csv")

# Converting data.frame to data.table
storm_dt <- as.data.table(storm_df)
colnames(storm_df)
```

```
##  [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"
```

Remove unnessessary columns

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

```
                                                                    , "FATALITIES"
                                                                    , "INJURIES"
                                                                    , "PROPDMG"
                                                                    , "PROPDMGEXP"
                                                                    , "CROPDMG"
                                                                    , "CROPDMGEXP") ]
```

Making PRPDMGEXP and CROPDMGEXP numerical values, to calculate losses.

```
# Change all damage exponents to uppercase just in case.
cols <- c("PROPDMGEXP", "CROPDMGEXP")
storm_dt[,  (cols) := c(lapply(.SD, toupper)), .SDcols = cols]

# Map alphanumeric exponent codes
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)

storm_dt[, PROPDMGEXP := propDmgKey[as.character(storm_dt[,PROPDMGEXP])]]
storm_dt[is.na(PROPDMGEXP), PROPDMGEXP := 10^0 ]

storm_dt[, CROPDMGEXP := cropDmgKey[as.character(storm_dt[,CROPDMGEXP])] ]
storm_dt[is.na(CROPDMGEXP), CROPDMGEXP := 10^0 ]
```

Calculating losses and adding them to the table.

```
storm_dt <- storm_dt[, .(EVTYPE, FATALITIES, INJURIES, PROPDMG, PROPDMGEXP, propTotal = PROPDMG * PROPDM
```

Calculating Total Properties and Crop costs

```
cost_dt <- storm_dt[, .(propCost = sum(propTotal), cropCost = sum(cropTotal),
total_cost = sum(propTotal) + sum(cropTotal)), by = .(EVTYPE)]

cost_dt <- cost_dt[order(-total_cost), ]
```

```
cost_dt <- cost_dt[1:10, ]

head(cost_dt, 5)
```

```
##               EVTYPE     propCost   cropCost   total_cost
## 1:             FLOOD 144657709807 5661968450 150319678257
## 2: HURRICANE/TYPHOON  69305840000 2607872800  71913712800
## 3:           TORNADO  56947380676  414953270  57362333946
## 4:       STORM SURGE  43323536000       5000  43323541000
## 5:              HAIL  15735267513 3025954473  18761221986
```

Calculating human losses

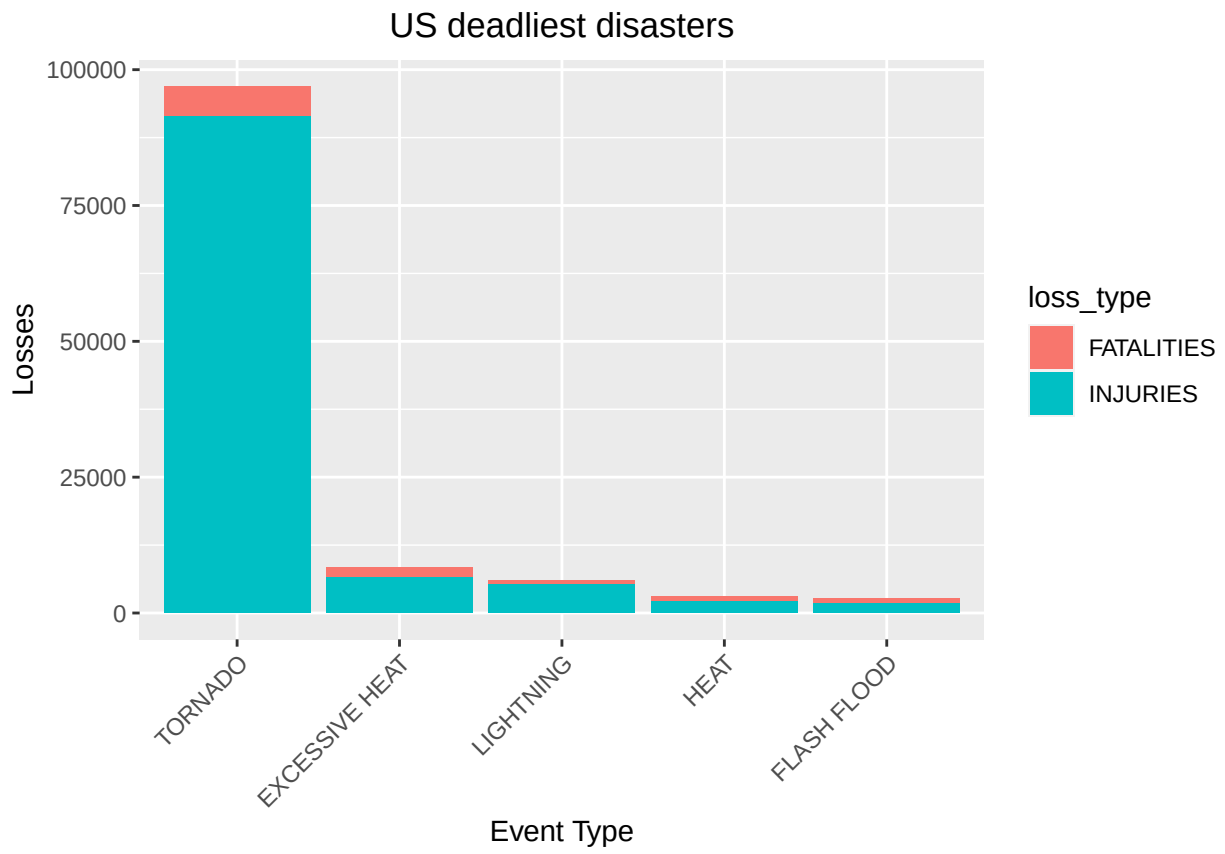
```
total_human_dt <- storm_dt[, .(FATALITIES=sum(FATALITIES), INJURIES=sum(INJURIES), TOTAL = sum(FATALITI

total_human_dt <- total_human_dt[order(-FATALITIES)]
total_human_dt <- total_human_dt[1:5]
head(total_human_dt)
```

```
##            EVTYPE FATALITIES INJURIES TOTAL
## 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

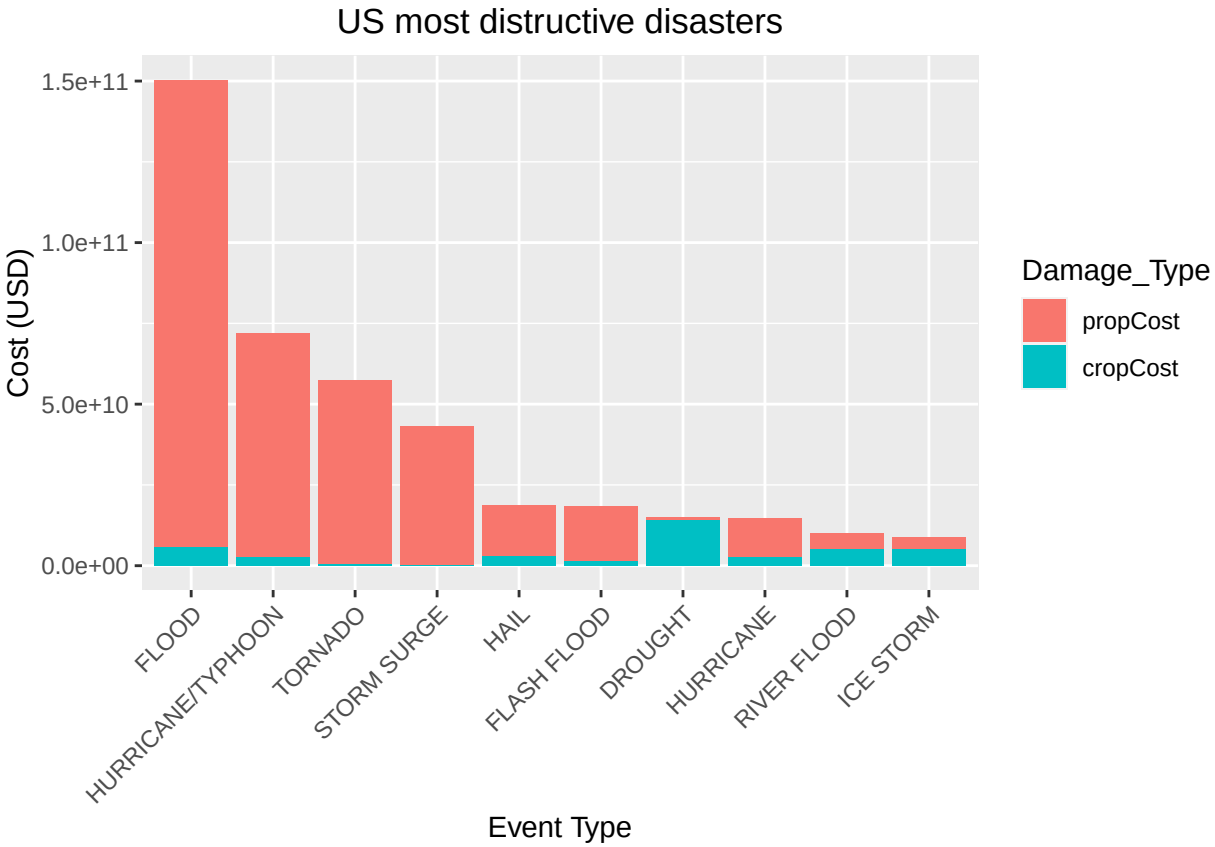
```
losses <- melt(total_human_dt[, .(EVTYPE, FATALITIES, INJURIES)],
               id.vars="EVTYPE", variable.name = "loss_type")
healthChart <- ggplot(losses, aes(x=reorder(EVTYPE, -value), y=value))
healthChart = healthChart + geom_bar(stat="identity", aes(fill=loss_type))
healthChart = healthChart + ylab("Losses")
healthChart = healthChart + xlab("Event Type")
healthChart = healthChart + theme(axis.text.x = element_text(angle=45, hjust=1))
healthChart = healthChart + ggtitle("US deadliest disasters") + theme(plot.title = element_text(hjust =
healthChart
```



3.2: Events that have the Greatest Economic Consequences

```
econ <- melt(cost_dt[,.(EVTYPE,propCost, cropCost)], id.vars="EVTYPE", variable.name = "Damage_Type")
econChart <- ggplot(econ, aes(x=reorder(EVTYPE, -value), y=value))
econChart = econChart + geom_bar(stat="identity", aes(fill=Damage_Type),)
econChart = econChart + ylab("Cost (USD)")
econChart = econChart + xlab("Event Type")
econChart = econChart + theme(axis.text.x = element_text(angle=45, hjust=1))
econChart = econChart + ggtitle("US most distructive disasters") + theme(plot.title = element_text(hjus

econChart
```


US most distructive disasters
1.5e+11
1.0e+11
5.0e+10
0.0e+00
Cost (USD)
FLOOD
HURRICANE/TYPHOON
TORNADO
STORM SURGE
HAIL
FLASH FLOOD
DROUGHT
HURRICANE
RIVER FLOOD
ICE STORM
Event Type
Damage_Type
propCost
cropCost