NOAA Storm Database Analysis on Impact of Storm Events on Health and Economic Damage

Sai Thu Ya Aung

8/14/2021

Synopsis

This report is submitted by Sai Thu Ya Aung for Reproducible Research course.

The report is about analysis on NOAA database that how storms and certain weather events can damage public health and economy. The impact on public healh is analysed with fatalities and injuries counts, and that of on economy is analysed based on property and crop damage.

The results of the analysis contain 10 weather events that cuased most damaging on public health and economy.

Data Processing

Load required packages

```
library(ggplot2)
library(gridExtra)
```

Download and load the NOAA data

```
if(!file.exists("./dataset")){dir.create("./dataset")}
downloadUrl <- "https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2FStormData.csv.bz2"
download.file(downloadUrl, destfile = "./dataset/StormData.csv.bz2")

storm_data <- read.table("./dataset/StormData.csv.bz2", sep = ",", header = TRUE, na.strings = "")

str(storm_data)
```

```
## 'data.frame':    902297 obs. of  37 variables:
##  $ STATE__   : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ BGN_DATE  : Factor w/ 16335 levels "1/1/1966 0:00:00",..: 6523 6523 4242 11116 2224 2224 2260 383 3980 3980
...
##  $ BGN_TIME  : Factor w/ 3608 levels "00:00:00 AM",..: 272 287 2705 1683 2584 3186 242 1683 3186 3186 ...
##  $ TIME_ZONE : Factor w/ 22 levels "ADT","AKS","AST",..: 7 7 7 7 7 7 7 7 7 7 ...
##  $ COUNTY    : num  97 3 57 89 43 77 9 123 125 57 ...
##  $ COUNTYNAME: Factor w/ 29600 levels "5NM E OF MACKINAC BRIDGE TO PRESQUE ISLE LT MI",..: 13512 1872 4597 105
91 4371 10093 1972 23872 24417 4597 ...
##  $ STATE     : Factor w/ 72 levels "AK","AL","AM",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ EVTYPE    : Factor w/ 985 levels "   HIGH SURF ADVISORY",..: 834 834 834 834 834 834 834 834 834 834 ...
##  $ BGN_RANGE : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ BGN_AZI   : Factor w/ 34 levels "  N"," NW","E",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ BGN_LOCATI: Factor w/ 54428 levels " Christiansburg",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ END_DATE  : Factor w/ 6662 levels "1/1/1993 0:00:00",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ END_TIME  : Factor w/ 3646 levels " 0900CST"," 200CST",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ COUNTY_END: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ COUNTYENDN: logi  NA NA NA NA NA NA ...
##  $ END_RANGE : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ END_AZI   : Factor w/ 23 levels "E","ENE","ESE",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ END_LOCATI: Factor w/ 34505 levels " CANTON"," TULIA",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ LENGTH    : num  14 2 0.1 0 0 1.5 1.5 0 3.3 2.3 ...
##  $ WIDTH     : num  100 150 123 100 150 177 33 33 100 100 ...
##  $ F         : int  3 2 2 2 2 2 2 1 3 3 ...
##  $ MAG       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ FATALITIES: num  0 0 0 0 0 0 0 0 1 0 ...
##  $ INJURIES  : num  15 0 2 2 2 6 1 0 14 0 ...
##  $ PROPDMG   : num  25 2.5 25 2.5 2.5 2.5 2.5 2.5 25 25 ...
##  $ PROPDMGEXP: Factor w/ 18 levels "-","?","+","0",..: 16 16 16 16 16 16 16 16 16 16 ...
##  $ CROPDMG   : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ CROPDMGEXP: Factor w/ 8 levels "?","0","2","B",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ WFO       : Factor w/ 541 levels " CI","%SD","$AC",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ STATEOFFIC: Factor w/ 249 levels "ALABAMA, Central",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ ZONENAMES : Factor w/ 25111 levels "
"| __truncated__,..: NA NA NA NA NA NA NA NA NA NA ...
##  $ LATITUDE  : num  3040 3042 3340 3458 3412 ...
##  $ LONGITUDE : num  8812 8755 8742 8626 8642 ...
##  $ LATITUDE_E: num  3051 0 0 0 0 ...
##  $ LONGITUDE_: num  8806 0 0 0 0 ...
##  $ REMARKS   : Factor w/ 436780 levels "\t","\t\t","\t\t\t\t",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ REFNUM    : num  1 2 3 4 5 6 7 8 9 10 ...
```

Create new storm data with only required variables

- `EVTYPE` : Event type
- `FATALITIES` : Number of fatalities
- `INJURIES` : Number of injuries
- `PROPDMG` : Property damage count
- `PROPDMGEXP` : Property damage unit
- `CROPDMG` : Crop damage count
- `CROPDMGEXP` : Crop damage unit

```
new_storm_data <- storm_data[, c("EVTYPE", "FATALITIES", "INJURIES", "PROPDMG", "PROPDMGEXP", "CROPDMG", "CROPDMG
EXP")]
```

Check is there any NA

```
apply(new_storm_data, 2, function(x) any(is.na(x)))
```

```
##     EVTYPE FATALITIES   INJURIES    PROPDMG PROPDMGEXP    CROPDMG CROPDMGEXP
##      FALSE      FALSE      FALSE      FALSE       TRUE      FALSE       TRUE
```

Handle NAs and convert crop and property damage unit

Unit indicator	Description	Replaced by
NA	Undefined	0
h/H	Hundred	100
K	Kilo	1000
m/M	Million	1,000,000
B	Billion	1,000,000,000
1, 2, …, 8	Decimal Unit	Left unchanged
0	Unclear	0
+	Unclear	0
-	Unclear	0

In this case, we left out NAs and unclear units by replacing with zero and work only with accurate data.

Create a new column with converted unit for crop damage

```
for(i in 1:nrow(new_storm_data)) {
        index_unit <- 0
        if(!is.na(new_storm_data$CROPDMGEXP[i])) {
                temp <- tolower(new_storm_data$CROPDMGEXP[i])
                if (grepl("h", temp) == TRUE) {
                        index_unit <- 100
                } else if (grepl("k", temp) == TRUE) {
                        index_unit <- 1000
                } else if (grepl("m", temp) == TRUE) {
                        index_unit <- 1000000
                } else if (grepl("b", temp) == TRUE) {
                        index_unit <- 1000000000
                } else if (grepl("^[+-]", temp) == TRUE) {
                        index_unit <- 0
                } else {
                        index_unit <- strtoi(temp)
                }
        }
        new_storm_data$crop_dmg_unit[i] <- index_unit
}
```

Create a new column with converted unit for property damage

```
for(i in 1:nrow(new_storm_data)) {
        index_unit <- 0
        if(!is.na(new_storm_data$PROPDMGEXP[i])) {
                temp <- tolower(new_storm_data$PROPDMGEXP[i])
                if (grepl("h", temp) == TRUE) {
                        index_unit <- 100
                } else if (grepl("k", temp) == TRUE) {
                        index_unit <- 1000
                } else if (grepl("m", temp) == TRUE) {
                        index_unit <- 1000000
                } else if (grepl("b", temp) == TRUE) {
                        index_unit <- 1000000000
                } else if (grepl("^[+-]", temp) == TRUE) {
                        index_unit <- 0
                } else {
                        index_unit <- strtoi(temp)
                }
        }
        new_storm_data$prop_dmg_unit[i] <- index_unit
}
```

Check data again with new columns

```
str(new_storm_data)
```

```
## 'data.frame':    902297 obs. of  9 variables:
##  $ EVTYPE       : Factor w/ 985 levels "   HIGH SURF ADVISORY",..: 834 834 834 834 834 834 834 834 834 834 ...
##  $ FATALITIES   : num  0 0 0 0 0 0 0 0 1 0 ...
##  $ INJURIES     : num  15 0 2 2 2 6 1 0 14 0 ...
##  $ PROPDMG      : num  25 2.5 25 2.5 2.5 2.5 2.5 2.5 25 25 ...
##  $ PROPDMGEXP   : Factor w/ 18 levels "-","?","+","0",..: 16 16 16 16 16 16 16 16 16 16 ...
##  $ CROPDMG      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ CROPDMGEXP   : Factor w/ 8 levels "?","0","2","B",..: NA NA NA NA NA NA NA NA NA NA ...
##  $ crop_dmg_unit: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ prop_dmg_unit: num  1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 ...
```

```
head(new_storm_data)
```

```
##    EVTYPE FATALITIES INJURIES PROPDMG PROPDMGEXP CROPDMG CROPDMGEXP
## 1 TORNADO          0       15    25.0          K       0       <NA>
## 2 TORNADO          0        0     2.5          K       0       <NA>
## 3 TORNADO          0        2    25.0          K       0       <NA>
## 4 TORNADO          0        2     2.5          K       0       <NA>
## 5 TORNADO          0        2     2.5          K       0       <NA>
## 6 TORNADO          0        6     2.5          K       0       <NA>
##   crop_dmg_unit prop_dmg_unit
## 1             0          1000
## 2             0          1000
## 3             0          1000
## 4             0          1000
## 5             0          1000
## 6             0          1000
```

Results

Get top 10 most harmful storm events on fatalities

```
storm_fatalities <- aggregate(FATALITIES ~ EVTYPE, data = new_storm_data, FUN = sum)
storm_fatalities <- storm_fatalities[order(-storm_fatalities$FATALITIES),]

new_storm_fatalities <- storm_fatalities[1:10, ]

new_storm_fatalities
```

```
##             EVTYPE FATALITIES
## 834        TORNADO       5633
## 130 EXCESSIVE HEAT       1903
## 153    FLASH FLOOD        978
## 275           HEAT        937
## 464      LIGHTNING        816
## 856      TSTM WIND        504
## 170          FLOOD        470
## 585    RIP CURRENT        368
## 359      HIGH WIND        248
## 19       AVALANCHE        224
```

Tornado and excess heat most of the fatalities.

Get top 10 most harmful storm events on injuries

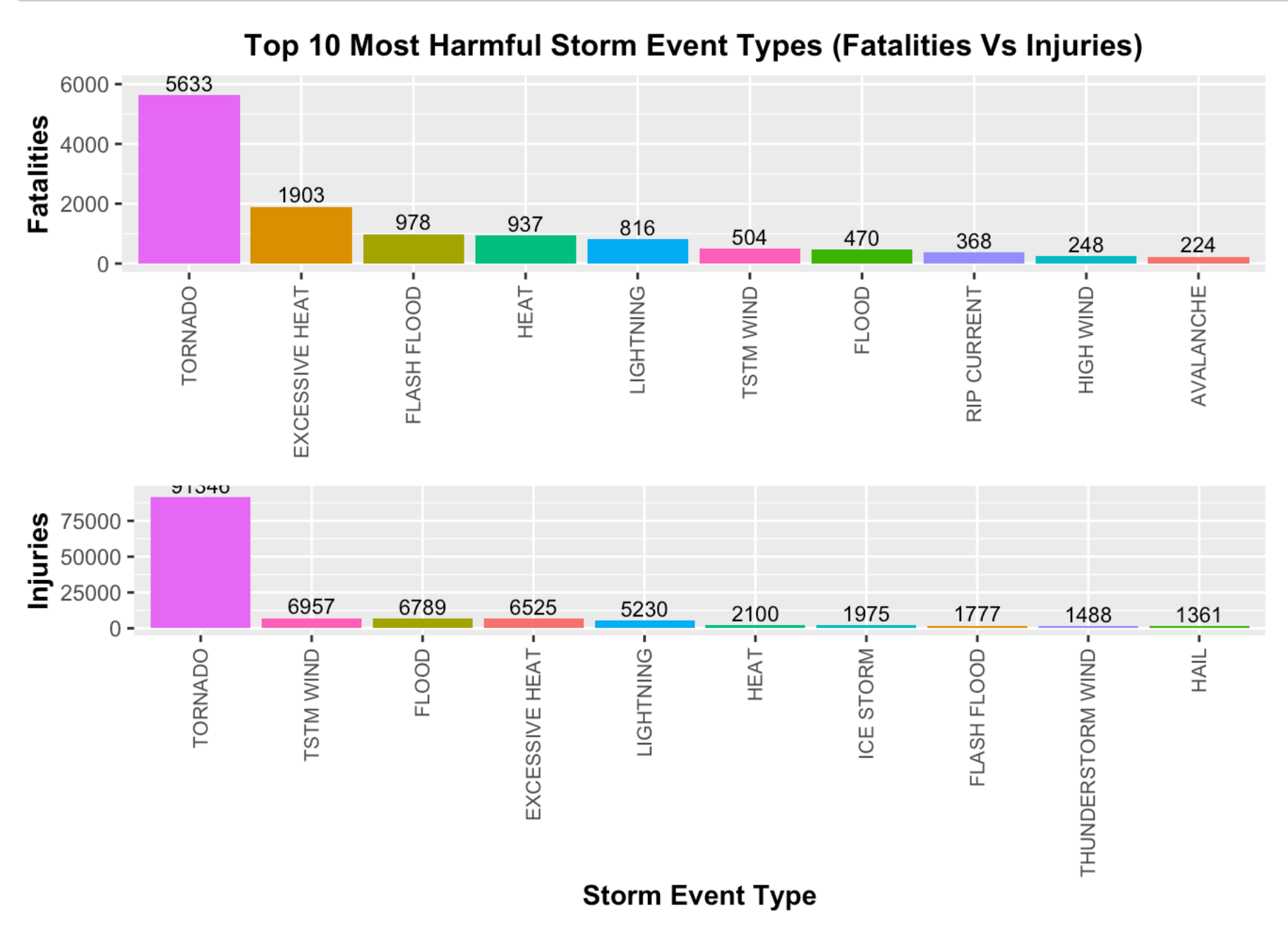
```
storm_injuries <- aggregate(INJURIES ~ EVTYPE, data = new_storm_data, FUN = sum)
storm_injuries <- storm_injuries[order(-storm_injuries$INJURIES),]

new_storm_injuries <- storm_injuries[1:10,]

new_storm_injuries
```

```
##                EVTYPE INJURIES
## 834           TORNADO    91346
## 856         TSTM WIND     6957
## 170             FLOOD     6789
## 130    EXCESSIVE HEAT     6525
## 464         LIGHTNING     5230
## 275              HEAT     2100
## 427         ICE STORM     1975
## 153       FLASH FLOOD     1777
## 760 THUNDERSTORM WIND     1488
## 244              HAIL     1361
```

Tornado caused siginificant higher injuries than the rest of weather events.

```
plot1 <- ggplot(new_storm_fatalities, aes(x = reorder(EVTYPE, -FATALITIES), y = FATALITIES, fill = EVTYPE)) +
        geom_bar(stat = "identity") +
        theme(legend.position = "none") +
        theme(plot.title = element_text(hjust = 0.5, face = "bold", size = 12)) +
        theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1, size =8)) +
        theme(axis.title.x = element_blank()) +
        theme(axis.title.y = element_text(face = "bold")) +
        geom_text(label = new_storm_fatalities$FATALITIES, vjust = -.3, size =3) +
        ylab("Fatalities") +
        ggtitle("Top 10 Most Harmful Storm Event Types (Fatalities Vs Injuries)") +
        scale_y_continuous(limits = c(0, 6000))
```

```
plot2 <- ggplot(new_storm_injuries, aes(x = reorder(EVTYPE, -INJURIES), y = INJURIES, fill = EVTYPE)) +
        geom_bar(stat = "identity") +
        theme(legend.position = "none") +
        theme(plot.title = element_text(hjust = 0.5, face = "bold", size = 12)) +
        theme(axis.title.x = element_text(face = "bold")) +
        theme(axis.title.y = element_text(face = "bold")) +
        geom_text(label = new_storm_injuries$INJURIES, vjust = -.3, size =3) +
        theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1, size =8)) +
        xlab("Storm Event Type") +
        ylab("Injuries") +
        scale_y_continuous(limits = c(0, 95000))

grid.arrange(plot1, plot2, nrow =2)
```



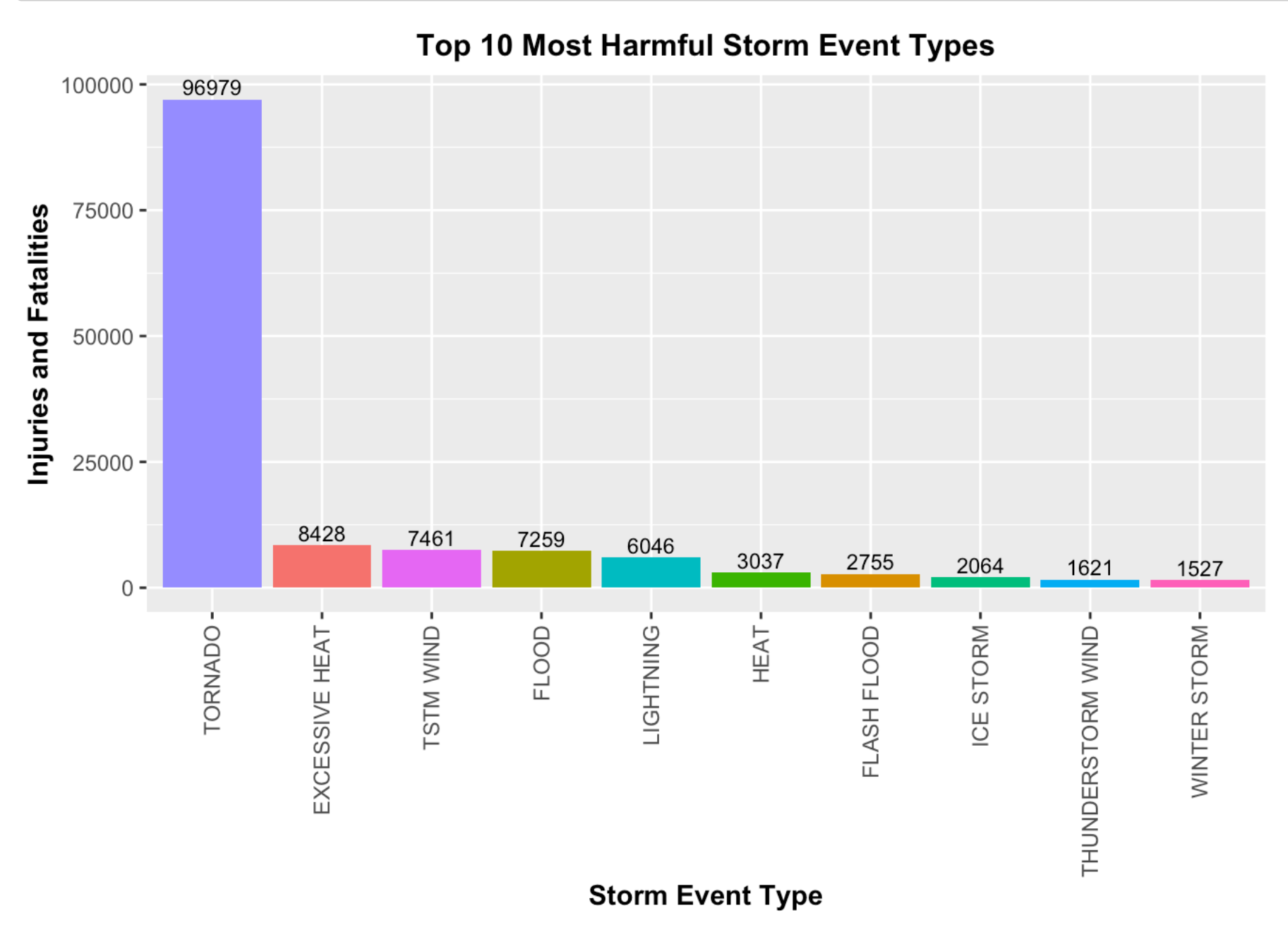
Get top 10 most harmful storm events on fatalities and injuries combined

```
new_storm_data$INJU_FATAL <- new_storm_data$INJURIES + new_storm_data$FATALITIES

storm_injuries_fatalities <- aggregate(INJU_FATAL ~ EVTYPE, data = new_storm_data, FUN = sum)
storm_injuries_fatalities <- storm_injuries_fatalities[order(-storm_injuries_fatalities$INJU_FATAL),]

new_storm_injuries_fatalities <- storm_injuries_fatalities[1:10, ]

new_storm_injuries_fatalities
```

```
##                EVTYPE INJU_FATAL
## 834           TORNADO      96979
## 130    EXCESSIVE HEAT       8428
## 856         TSTM WIND       7461
## 170             FLOOD       7259
## 464         LIGHTNING       6046
## 275              HEAT       3037
## 153       FLASH FLOOD       2755
## 427         ICE STORM       2064
## 760 THUNDERSTORM WIND       1621
## 972      WINTER STORM       1527
```

By combining fatalites and injuries count, tornado is the most damaging weather event.

```
ggplot(new_storm_injuries_fatalities, aes(x = reorder(EVTYPE, -INJU_FATAL), y = INJU_FATAL, fill = EVTYPE)) +
        geom_bar(stat = "identity") +
        theme(legend.position = "none") +
        theme(plot.title = element_text(hjust = 0.5, face = "bold", size = 12)) +
        theme(axis.title.x = element_text(face = "bold")) +
        theme(axis.title.y = element_text(face = "bold")) +
        geom_text(label = new_storm_injuries_fatalities$INJU_FATAL, vjust = -.3, size =3) +
        theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
        xlab("Storm Event Type") +
        ylab("Injuries and Fatalities") +
        ggtitle("Top 10 Most Harmful Storm Event Types")
```



Get top 10 most damaging storm events on properties

```
storm_prop <- aggregate((PROPDMG * prop_dmg_unit) ~ EVTYPE, data = new_storm_data, FUN = sum)
names(storm_prop) <- c("EVTYPE", "total_prop")
storm_prop <- storm_prop[order(-storm_prop$total_prop),]

new_storm_prop <- storm_prop[1:10,]

new_storm_prop
```

```
##                EVTYPE   total_prop
## 170             FLOOD 144657709800
## 411 HURRICANE/TYPHOON  69305840000
## 834           TORNADO  56937160991
## 670       STORM SURGE  43323536000
## 153       FLASH FLOOD  16140812087
## 244              HAIL  15732267370
## 402         HURRICANE  11868319010
## 848    TROPICAL STORM   7703890550
## 972      WINTER STORM   6688497250
## 359         HIGH WIND   5270046260
```

Flood caused highest damage on properties.

Get top 10 most damaging storm events on crops

```
storm_crop <- aggregate((PROPDMG * crop_dmg_unit) ~ EVTYPE, data = new_storm_data, FUN = sum)
names(storm_crop) <- c("EVTYPE", "total_crop")
storm_crop <- storm_crop[order(-storm_crop$total_crop),]

new_storm_crop <- storm_crop[1:10,]

new_storm_crop
```

```
##                EVTYPE   total_crop
## 426         ICE STORM 500228721840
## 243              HAIL  43401421640
## 169             FLOOD  29493285700
## 153       FLASH FLOOD  18002019760
## 855         TSTM WIND  15847431330
## 833           TORNADO  10145532010
## 410 HURRICANE/TYPHOON   9243299510
## 759 THUNDERSTORM WIND   7277529970
## 589       RIVER FLOOD   6223490000
## 358         HIGH WIND   4309796400
```

Ice storm caused significant damage on crops compare with other storm events.

```
plot3 <- ggplot(new_storm_prop, aes(x = reorder(EVTYPE, -total_prop), y = total_prop, fill = EVTYPE)) +
        geom_bar(stat = "identity") +
        theme(legend.position = "none") +
        theme(plot.title = element_text(hjust = 0.5, face = "bold", size = 12)) +
        theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1, size =6)) +
        theme(axis.title.x = element_blank()) +
        theme(axis.title.y = element_text(face = "bold")) +
        geom_text(label = paste(round(new_storm_prop$total_prop/1000000000, 2), "B"), vjust = -.3, size =3) +
        ylab("Property Damage") +
        ggtitle("Top 10 Most Damaging Storm Event Types (Properties and Crops)") +
        scale_y_continuous(limits = c(0, 154657709800))
```

```
plot4 <- ggplot(new_storm_crop, aes(x = reorder(EVTYPE, -total_crop), y = total_crop, fill = EVTYPE)) +
        geom_bar(stat = "identity") +
        theme(legend.position = "none") +
        theme(plot.title = element_text(hjust = 0.5, face = "bold", size = 12)) +
        theme(axis.title.x = element_text(face = "bold")) +
        theme(axis.title.y = element_text(face = "bold")) +
        geom_text(label = paste(round(new_storm_crop$total_crop/1000000000, 2), "B"), vjust = -.3, size =3) +
        theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1, size =6)) +
        xlab("Storm Event Type") +
        ylab("Crop Damage") +
        scale_y_continuous(limits = c(0, 530228721840))

grid.arrange(plot3, plot4, nrow =2)
```