Severe Weather Top Events Requiring the Greatest Government and Municipality Resource Needs

Betsy Nash

December 13, 2017

Synopsis

The need for this review is to offer data-driven insights to assist the prioritization of government/municipal resources for different types of weather events. There are two questions posed:

1. Across the United States, which types of events (as indicated by 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?

Definitions:

a. Population health was defined as the aggregation of fatalaties and injuries, with the variable labeled TotalBodilyInjury

b. Economic consequences was defined as the aggregation of property damage and crop damage, with the variable labeled TotalPropertyDamage

Key Takeaways:

1. The US weather event most harmful to population health is tornados, followed by excessive heat, thunderstorm wind, flood, and lightning.
2. The US weather event causing the greatest economic consequence is flood, followed by hurricane/typhoon, tornado, storm surge, and hail to complete the top 5 events.

Note tornados and flood comprise the top 5 events in each view, health and property damage. These two weather events are worthy of a deeper look by governments and municipalities regarding resource deployment.

Data Processing Overall

Summary: Two datasets were created from the same source. One dataset for TotalBodilyInjury, one dataset for TotalPropertyDamage. The property damage dataset needed a conversion of the damage estimates to a first dollar basis. For example, the data contained values summarized as 2.5B or 100K, and in order to obtain a total these values needed to be converted into 2,500,000,000 and 100,000.

Data Source and imported dataset “StormDataUse”:

```r
library(R.utils)
##hide warnings and messages when loading library(package)
knitr::opts_chunk$set(warning=FALSE,message = FALSE)
Connection<-"https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2FStormData.csv.bz2"
destfile<-"StormData.csv.bz2"
download.file(Connection,destfile)
StormDataUse<-read.csv("StormData.csv.bz2")
```

The data and columns of interest were explored for possible data scrubbing & cleaning. Note page 12 of the storm data preparation document states the PROPEXP and CROPEXP columns represent exponent for the values in the PROPDMG and CROPDMG fields, respectively.

```r
library(dplyr)
#overall data summaries
dim(StormDataUse)
```

```
## [1] 902297     37
```

```r
names(StormDataUse)
```

```
##  [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"
```

```r
str(StormDataUse)
```

```
## '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 2
224 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/ 29601 levels "","5NM E OF MACKINAC BRIDGE TO PRESQUE ISLE
LT MI",..: 13513 1873 4598 10592 4372 10094 1973 23873 24418 4598 ...
##  $ 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 83
4 834 834 834 834 834 ...
##  $ BGN_RANGE : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ BGN_AZI   : Factor w/ 35 levels "","  N"," NW",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ BGN_LOCATI: Factor w/ 54429 levels "","- 1 N Albion",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ END_DATE  : Factor w/ 6663 levels "","1/1/1993 0:00:00",..: 1 1 1 1 1 1 1 1 1
1 ...
##  $ END_TIME  : Factor w/ 3647 levels ""," 0900CST",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ 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/ 24 levels "","E","ENE","ESE",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ END_LOCATI: Factor w/ 34506 levels "","- .5 NNW",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ 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/ 19 levels "","-","?","+",..: 17 17 17 17 17 17 17 17 17 1
7 ...
##  $ CROPDMG   : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ CROPDMGEXP: Factor w/ 9 levels "","?","0","2",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ WFO       : Factor w/ 542 levels ""," CI","$AC",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ STATEOFFIC: Factor w/ 250 levels "","ALABAMA, Central",..: 1 1 1 1 1 1 1 1 1
1 ...
##  $ ZONENAMES : Factor w/ 25112 levels
"","
                             "| __truncated__,..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ 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/ 436781 levels "","-2 at Deer Park\n",..: 1 1 1 1 1 1 1 1
1 1 ...
##  $ REFNUM    : num  1 2 3 4 5 6 7 8 9 10 ...
```

```
#head and tail were also reviewed, but not printed in the interest of report length
#head(StormDataUse,50)
#tail(StormDataUse,50)
#explore columns of interest
# look for na's
sum(is.na(StormDataUse$FATALITIES))
```

```
## [1] 0
```

```
sum(is.na(StormDataUse$INJURIES))
```

```
## [1] 0
```

```
sum(is.na(StormDataUse$PROPDMG))
```

```
## [1] 0
```

```
sum(is.na(StormDataUse$CROPDMG))
```

```
## [1] 0
```

```
# obtain totals for numeric columns of interest
sum(StormDataUse$FATALITIES)
```

```
## [1] 15145
```

```
sum(StormDataUse$INJURIES)
```

```
## [1] 140528
```

```
# note the following are exploring the column for errors prior to conversion from summ
arized units (2.5 of 2.5B) to first dollar (2,500,000,000)
formatC(round(sum(StormDataUse$PROPDMG),0), format = "d", big.mark=",")
```

```
## [1] "10,884,500"
```

```
formatC(round(sum(StormDataUse$CROPDMG),0), format = "d", big.mark=",")
```

```
## [1] "1,377,827"
```

```
#unique category values for non-numeric cols of interest
uniqueSD<-distinct(select(StormDataUse, STATE, EVTYPE))
sum(is.na(uniqueSD$EVTYPE))
```

```
## [1] 0
```

```
uniqueUnitsPD<-distinct(select(StormDataUse,PROPDMGEXP))
uniqueUnitsCROP<-distinct(select(StormDataUse,CROPDMGEXP))
```

The data needed for analysis was found to be clean except for the exponent units neeed for property damage estimates.

The next step is to create the datasets for TotalBodilyInjury and TotalPropertyDamage.

Data Processing - DataSet for Bodily Injury (Fatalities and Injuries)

The “StormDataUse” data just needs to be filtered for the columns of interest and aggregated for counts.

```
#table(StormDataUse$Cols of Interest)
myvars <- c("EVTYPE", "FATALITIES", "INJURIES")
StormDataSet<-StormDataUse[myvars]
StormDataSet <- mutate(StormDataSet, TotalBodilyInjury=FATALITIES+INJURIES)
#Test BIData
summary(StormDataSet)
```

```
##                 EVTYPE         FATALITIES          INJURIES        
##  HAIL             :288661   Min.   :  0.0000   Min.   :   0.0000  
##  TSTM WIND        :219940   1st Qu.:  0.0000   1st Qu.:   0.0000  
##  THUNDERSTORM WIND: 82563   Median :  0.0000   Median :   0.0000  
##  TORNADO          : 60652   Mean   :  0.0168   Mean   :   0.1557  
##  FLASH FLOOD      : 54277   3rd Qu.:  0.0000   3rd Qu.:   0.0000  
##  FLOOD            : 25326   Max.   :583.0000   Max.   :1700.0000  
##  (Other)          :170878                                         
##  TotalBodilyInjury 
##  Min.   :   0.0000  
##  1st Qu.:   0.0000  
##  Median :   0.0000  
##  Mean   :   0.1725  
##  3rd Qu.:   0.0000  
##  Max.   :1742.0000  
## 
```

```
#head and tail were also reviewed, but not printed in the interest of report length
#head(StormDataSet,50)
#tail(StormDataSet,50)

#aggregate by weather event for BI Injuries
Event <- group_by(StormDataSet, EVTYPE)
StormDataSum<-summarize(Event, TOTFATALITIES = sum(FATALITIES),TOTINJURIES = sum(INJUR
IES),
          TotalBodilyInjury = sum(TotalBodilyInjury))
#test aggregation
#head was also reviewed, but not printed in the interest of report length
#head(StormDataSum,50)
summary(StormDataSum)
```

```
##                     EVTYPE    TOTFATALITIES      TOTINJURIES
##     HIGH SURF ADVISORY:  1   Min.   :   0.00   Min.   :    0.0
##   COASTAL FLOOD       :  1   1st Qu.:   0.00   1st Qu.:    0.0
##   FLASH FLOOD         :  1   Median :   0.00   Median :    0.0
##   LIGHTNING           :  1   Mean   :  15.38   Mean   :  142.7
##   TSTM WIND           :  1   3rd Qu.:   0.00   3rd Qu.:    0.0
##   TSTM WIND (G45)     :  1   Max.   :5633.00   Max.   :91346.0
##  (Other)              :979
##  TotalBodilyInjury
##  Min.   :    0
##  1st Qu.:    0
##  Median :    0
##  Mean   :  158
##  3rd Qu.:    0
##  Max.   :96979
##
```

Now that the data is summarized, sort to identify those events that comprise at least 80% (solid majority) of the total bodily injury count.

```
#arrange rows to get top n for question 1
StormDataSumBI<-arrange(StormDataSum, desc(TotalBodilyInjury))
#Top5 BI Events for charting
Q1Data<-data.frame(StormDataSumBI[1:5,])
#Portion Top N represent (shoot for >= 80%)
BIPortion<-formatC(100*(sum(Q1Data$TotalBodilyInjury)/sum(StormDataSumBI$TotalBodilyIn
jury)))
```

The top 5 events for Bodily Injury comprise 81.05% of the total fatalities and injuries.

Data Processing - DataSet for Property Damage

The exponent units neeed for property damage estimates contained some unidentifiable characters. Examples: blank, ?, -, and +. The following code shows the numeric mapping of each unique item in the *EXP columns.

```
#convert units into numeric form, where blank = 1 and default = 0 where code not ident
ified below (ex: ?, -, +)
Units=function(x){
        m=0;
        if(x=="") {m=1}
                else if(x=="0"){m=1}
                else if(x=="1"){m=10}
                else if(x=="2"){m=100}
                else if(x=="3"){m=1000}
                else if(x=="4"){m=10000}
                else if(x=="5"){m=100000}
                else if(x=="6"){m=1000000}
                else if(x=="7"){m=10000000}
                else if(x=="8"){m=100000000}
                else if(x=="B") {m=1000000000}
                else if(x=="h") {m=100}
                else if(x=="H") {m=100}
                else if(x=="k") {m=1000}
                else if(x=="K") {m=1000}
                else if(x=="M" | x=="m") {m=1000000}
                else if(x=="M" | x=="m") {m=1000000}

        m}

#validate the table
#convert to fixed from scientific
options("scipen"=100, "digits"=10)
#create lookup table for PD
uniqueUnitsPDf<-unique(factor(StormDataUse$PROPDMGEXP))
TestPD<-data.frame(uniqueUnitsPDf,sapply(uniqueUnitsPDf,Units))
#rename (new name = old name)
TestPD<-rename(TestPD, PROPDMGEXP = uniqueUnitsPDf,Units = sapply.uniqueUnitsPDf..Unit
s.)
print(TestPD)
```

```
##    PROPDMGEXP      Units
## 1           K       1000
## 2           M    1000000
## 3                      1
## 4           B 1000000000
## 5           m    1000000
## 6           +          0
## 7           0          1
## 8           5     100000
## 9           6    1000000
## 10          ?          0
## 11          4      10000
## 12          2        100
## 13          3       1000
## 14          h        100
## 15          7   10000000
## 16          H        100
## 17          -          0
## 18          1         10
## 19          8  100000000
```

```
#now crop (scipen above carries forward)
uniqueUnitsCROPf=unique(factor(StormDataUse$CROPDMGEXP))
#lookup table for crop
TestCrop<-data.frame(uniqueUnitsCROPf,sapply(uniqueUnitsCROPf,Units))
TestCrop<-rename(TestCrop, CROPDMGEXP = uniqueUnitsCROPf, Units = sapply.uniqueUnitsCR
OPf..Units.)
print(TestCrop)
```

```
##   CROPDMGEXP      Units
## 1                     1
## 2          M    1000000
## 3          K       1000
## 4          m    1000000
## 5          B 1000000000
## 6          ?          0
## 7          0          1
## 8          k       1000
## 9          2        100
```

All exponent codes have been identified in the “Units” function. The next step is to create columns in the property damage dataset to represent estimated costs as the multiplier in PROPDMG & CROPDMG to the appropriate unit value above.

```
#make a copy of dataset for conversion work
StormDataUse2<-StormDataUse
PropUnits=sapply(StormDataUse2$PROPDMGEXP,Units)
CropUnits=sapply(StormDataUse2$CROPDMGEXP,Units)
EconCostsDataSet=data.frame(EVTYPE=StormDataUse2$EVTYPE,PROPDMG=StormDataUse2$PROPDMG,
                            PROPDMGEXP=StormDataUse2$PROPDMGEXP,PropCost=StormDataUse2
$PROPDMG*PropUnits,
                            CROPDMG=StormDataUse2$CROPDMG,CROPDMGEXP=StormDataUse2$CRO
PDMGEXP,
                            CropCost=StormDataUse2$CROPDMG*CropUnits)
#Test new dataset
#Summary was reviewed but not printed in the interest of report length
#summary(EconCostsDataSet)
head(EconCostsDataSet,25)
```

```
##     EVTYPE PROPDMG PROPDMGEXP PropCost CROPDMG CROPDMGEXP CropCost
## 1  TORNADO    25.0          K    25000       0                   0
## 2  TORNADO     2.5          K     2500       0                   0
## 3  TORNADO    25.0          K    25000       0                   0
## 4  TORNADO     2.5          K     2500       0                   0
## 5  TORNADO     2.5          K     2500       0                   0
## 6  TORNADO     2.5          K     2500       0                   0
## 7  TORNADO     2.5          K     2500       0                   0
## 8  TORNADO     2.5          K     2500       0                   0
## 9  TORNADO    25.0          K    25000       0                   0
## 10 TORNADO    25.0          K    25000       0                   0
## 11 TORNADO     2.5          M  2500000       0                   0
## 12 TORNADO     2.5          M  2500000       0                   0
## 13 TORNADO   250.0          K   250000       0                   0
## 14 TORNADO     0.0          K        0       0                   0
## 15 TORNADO    25.0          K    25000       0                   0
## 16 TORNADO    25.0          K    25000       0                   0
## 17 TORNADO    25.0          K    25000       0                   0
## 18 TORNADO    25.0          K    25000       0                   0
## 19 TORNADO    25.0          K    25000       0                   0
## 20 TORNADO    25.0          K    25000       0                   0
## 21 TORNADO    25.0          K    25000       0                   0
## 22 TORNADO     2.5          K     2500       0                   0
## 23 TORNADO     2.5          K     2500       0                   0
## 24 TORNADO    25.0          K    25000       0                   0
## 25 TORNADO    25.0          K    25000       0                   0
```

```
#Tail was reviewed but not printed in the interest of report length
#tail(EconCostsDataSet,25)
#find records to review with crop damages
head(filter(EconCostsDataSet, CROPDMG > 0))
```

```
##                      EVTYPE PROPDMG PROPDMGEXP  PropCost CROPDMG
## 1 HURRICANE OPAL/HIGH WINDS     0.1          B 100000000      10
## 2        THUNDERSTORM WINDS     5.0          M   5000000     500
## 3            HURRICANE ERIN    25.0          M  25000000       1
## 4            HURRICANE OPAL    48.0          M  48000000       4
## 5            HURRICANE OPAL    20.0          m  20000000      10
## 6        THUNDERSTORM WINDS    50.0          K     50000      50
##   CROPDMGEXP CropCost
## 1          M 10000000
## 2          K   500000
## 3          M  1000000
## 4          M  4000000
## 5          m 10000000
## 6          K    50000
```

The “EconCostsDataSet” data just needs a column to represent the sum of PropCost & CropCost and aggregated for total damages.

```
#table(StormDataUse$Cols of Interest)
EconCostsDataSet <- mutate(EconCostsDataSet, TotalPropertyDamage=PropCost+CropCost)
#Test new dataset
summary(EconCostsDataSet)
```

```
##                EVTYPE          PROPDMG          PROPDMGEXP    
##  HAIL             :288661   Min.   :   0.0000          :465934  
##  TSTM WIND        :219940   1st Qu.:   0.0000   K      :424665  
##  THUNDERSTORM WIND: 82563   Median :   0.0000   M      : 11330  
##  TORNADO          : 60652   Mean   :  12.0631   0      :   216  
##  FLASH FLOOD      : 54277   3rd Qu.:   0.5000   B      :    40  
##  FLOOD            : 25326   Max.   :5000.0000   5      :    28  
##  (Other)          :170878                       (Other):    84  
##     PropCost               CROPDMG           CROPDMGEXP    
##  Min.   :           0   Min.   :  0.000000          :618413  
##  1st Qu.:           0   1st Qu.:  0.000000   K      :281832  
##  Median :           0   Median :  0.000000   M      :  1994  
##  Mean   :      474594   Mean   :  1.527022   k      :    21  
##  3rd Qu.:         500   3rd Qu.:  0.000000   0      :    19  
##  Max.   :115000000000   Max.   :990.000000   B      :     9  
##                                              (Other):     9  
##     CropCost          TotalPropertyDamage 
##  Min.   :         0   Min.   :           0  
##  1st Qu.:         0   1st Qu.:           0  
##  Median :         0   Median :           0  
##  Mean   :     54421   Mean   :      529015  
##  3rd Qu.:         0   3rd Qu.:        1000  
##  Max.   :5000000000   Max.   :115032500000  
## 
```

```
#head was also reviewed, but not printed in the interest of report length
#head(EconCostsDataSet,50)
head(filter(EconCostsDataSet, CROPDMG > 0))
```

```
##                         EVTYPE PROPDMG PROPDMGEXP  PropCost CROPDMG
## 1 HURRICANE OPAL/HIGH WINDS     0.1          B 100000000      10
## 2        THUNDERSTORM WINDS     5.0          M   5000000     500
## 3            HURRICANE ERIN    25.0          M  25000000       1
## 4            HURRICANE OPAL    48.0          M  48000000       4
## 5            HURRICANE OPAL    20.0          m  20000000      10
## 6        THUNDERSTORM WINDS    50.0          K     50000      50
##   CROPDMGEXP CropCost TotalPropertyDamage
## 1          M 10000000           110000000
## 2          K   500000             5500000
## 3          M  1000000            26000000
## 4          M  4000000            52000000
## 5          m 10000000            30000000
## 6          K    50000              100000
```

```
#aggregate by event for PD Injuries
Event2 <- group_by(EconCostsDataSet, EVTYPE)
EconDataSum<-summarize(Event2, TOTPROPDMG = sum(PropCost),TOTCROPDMG = sum(CropCost),T
otalPropertyDamage = sum(TotalPropertyDamage))
#Test aggregation
#head was also reviewed, but not printed in the interest of report length
#head(EconDataSum,50)
summary(EconDataSum)
```

```
##                     EVTYPE      TOTPROPDMG            TOTCROPDMG
##     HIGH SURF ADVISORY:  1   Min.   :           0   Min.   :          0
##    COASTAL FLOOD       :  1   1st Qu.:           0   1st Qu.:          0
##    FLASH FLOOD         :  1   Median :           0   Median :          0
##    LIGHTNING           :  1   Mean   :   434746060   Mean   :   49851972
##    TSTM WIND           :  1   3rd Qu.:       51050   3rd Qu.:          0
##    TSTM WIND (G45)     :  1   Max.   :144657709807   Max.   :13972566000
##  (Other)              :979
##  TotalPropertyDamage
##  Min.   :           0
##  1st Qu.:           0
##  Median :           0
##  Mean   :   484598031
##  3rd Qu.:       85000
##  Max.   :150319678257
##
```

Now that the data is summarized, sort to identify those events that comprise at least 80% (solid majority) of the total property damage costs.

```
#arrange rows to get top n PD for question 2
EconDataSumPD<-arrange(EconDataSum, desc(TotalPropertyDamage))
#Top10 PD Events
Q2Data<-data.frame(EconDataSumPD[1:10,])
#Portion Top N represent (shoot for >= 80%)
PDPortion<-formatC(100*(sum(Q2Data$TotalPropertyDamage)/sum(EconDataSumPD$TotalPropert
yDamage)))
```

The top 10 events for PropertyDamage comprise 85.62% of the total property damage and crop damage estimates.

Now that the datasets for each bodily injury and property damage are complete, let’s take a look at the results.

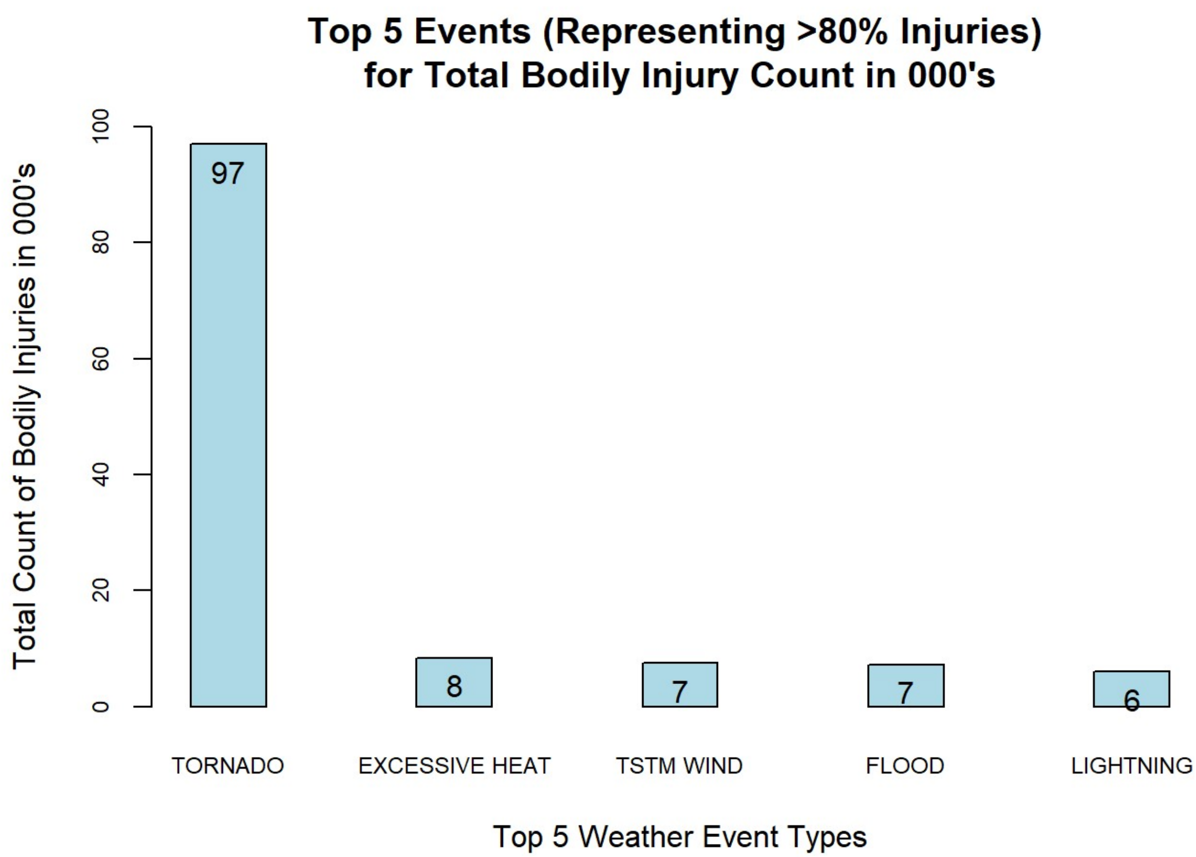
Results

The need is to prioritize government and municipal resources for different types of weather events.

Results - Question 1

Question 1) Across the United States, which types of events (as indicated by the EVTYPE variable) are most harmful with respect to population health? Health is defined as the summation of fatalities and injuries by weather event. The chart follows:

```
Q1Plot<-barplot(Q1Data$TotalBodilyInjury/1000,width=2,space=2,Q1Data$EVTYPE,
        cex.axis=0.75,cex.names=0.75,las=0,
        main="Top 5 Events (Representing >80% Injuries) \nfor Total Bodily Injury Coun
t in 000's",
        col="light blue",xlab="Top 5 Weather Event Types",
        ylab= "Total Count of Bodily Injuries in 000's",ylim=c(0,100))
text(Q1Plot, Q1Data$TotalBodilyInjury/1000,
     paste(round(Q1Data$TotalBodilyInjury/1000,0),sep=""),pos=1, cex=1, xpd=NA)
```

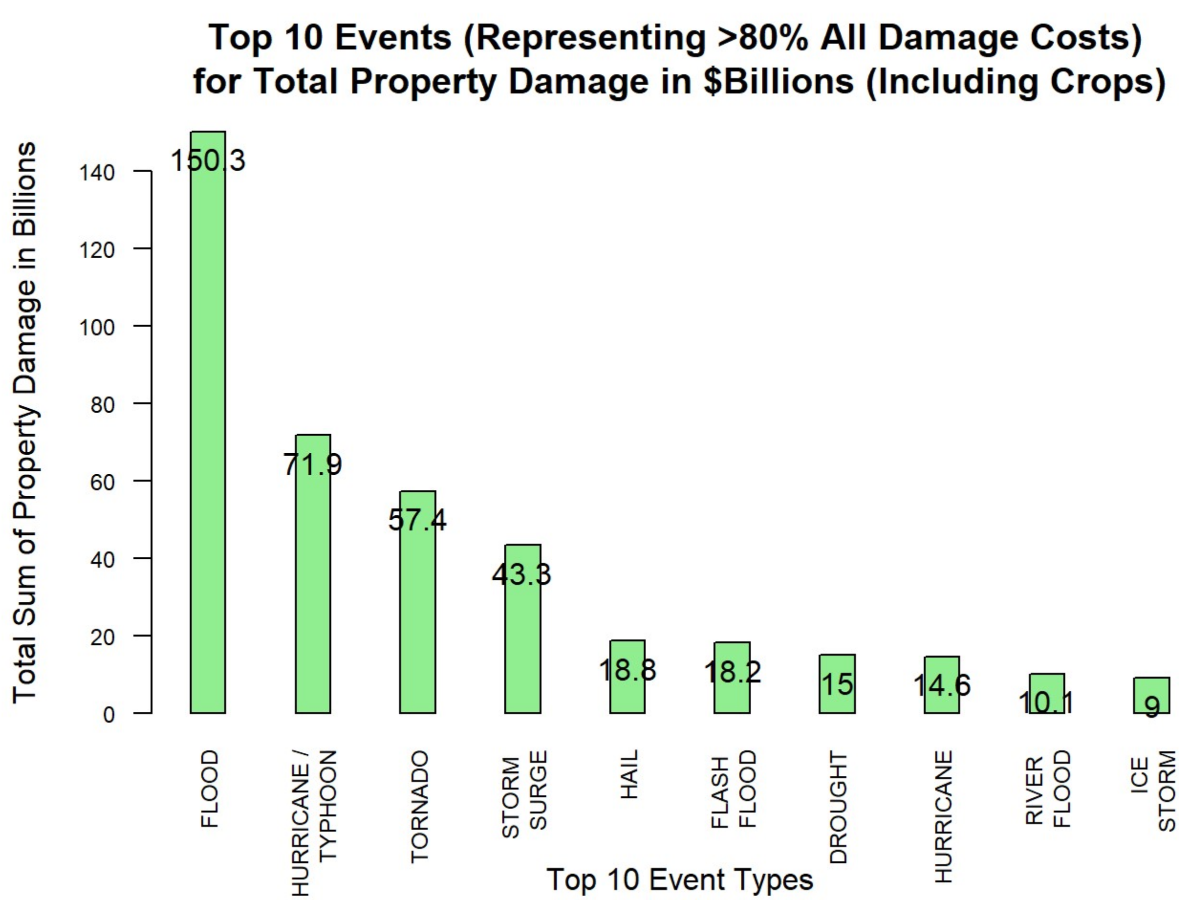


The result is tornados are the most severe event, followed by excessive heat, thunderstorm wind, flood, and lightning. These 5 events comprise 81.05% of all fatalities and injuries combined.

Results - Question 2

Question 2) Across the United States, which types of events have the greatest economic consequences? Economic consequences is defined as the aggregation of property damage and crop damage by weather event. The chart follows:

```
Q2labels<-c("FLOOD","HURRICANE /\nTYPHOON","TORNADO","STORM \nSURGE","HAIL",
        "FLASH \nFLOOD","DROUGHT","HURRICANE","RIVER \nFLOOD","ICE \nSTORM")
Q2Plot<-barplot(Q2Data$TotalPropertyDamage/1000000000,width=2,space=2,
        Q2Data$EVTYPE,cex.axis=0.75,cex.names=0.75,
        las=2,main="Top 10 Events (Representing >80% All Damage Costs) \nfor Total Pro
perty Damage in $Billions (Including Crops)",
        col="light green",names.arg=Q2labels,legend.text = NULL,
        ylab="Total Sum of Property Damage in Billions",ylim=c(0,150))
text(Q2Plot, Q2Data$TotalPropertyDamage/1000000000,
        paste(round(Q2Data$TotalPropertyDamage/1000000000,0.5),sep=""),
        pos=1, cex=1, xpd=NA)
#putting x-lab in margin so it doeasn't overlap
#margin=c(bottom,left,top,right) where default (5.1,4.1,4.1,2.1)
par(mar=c(7.1, 4.1, 4.1, 2.1))
mtext("Top 10 Event Types", 1, line=6, adj=0.5)
```



The result is floods are the most severe event, followed by hurricane/typhoon, tornado, storm surge, hail, flash flood, drought, hurricane, river flood, and ice storm. These 10 events comprise 85.62% of all property and crop damages combined.

Results - Overall Observation

Note tornados and flood comprise the top 5 events in each view, health and property damage. These two weather events are worthy of a deeper look by government/municipalities regarding resource deployment.

Challenge Results

This analysis is based on the entire dataset from years 1950 through November 2011. Event frequency and severity may have changed over time. An analysis over time is recommended to view for possible trends.