Reproducible Research Example 1

Linda Wang

January 17, 2016

Synopsis

In this report we aim to study the impact of storm events by type across the United Sates. We obtained data on storm events from the U.S. National Climatic Data Center (specifically, the National Oceanic and Atmospheric Administration). In particular, we assume the audience to this analysis would be a government or municipal manager who might be responsible for preparing for severe weather events and will need to prioritize resources for different types of events. Therefore we focus on more recent, more complete data (as opposed to data decades ago which are incomplete) as it makes more sense in conducting cross-sectional comparisons and concurrent policy making.

From the data, we found that, on average across the U.S., when we remove the most extreme cases and look at both the average and aggregate impacts, hurricane, typhoon, flood and tornado are the major types that are the most disastrous.

Data Processing

Before this, download the data here, then put it in your working directory. We already know that the data set is very large, and it takes quite long for an average personal computer to read the entire file. It will help if we can eliminate certain part of that when loading the data, so a smaller amount of information will be read (hence taking a shorter time). First, we need to take a look at the first (couple of) row(s) to get an idea of each column. Here I am only reading the first row as a demonstration:

Reading the first row and getting a general idea

```
data_row1 <- read.csv("repdata-data-StormData.csv.bz2",header=TRUE,nrow=1)
data_row1
```

```
##   STATE__           BGN_DATE BGN_TIME TIME_ZONE COUNTY COUNTYNAME STATE
## 1       1 4/18/1950 0:00:00      130       CST     97     MOBILE    AL
##    EVTYPE BGN_RANGE BGN_AZI BGN_LOCATI END_DATE END_TIME COUNTY_END
## 1 TORNADO         0      NA         NA       NA       NA          0
##   COUNTYENDN END_RANGE END_AZI END_LOCATI LENGTH WIDTH F MAG FATALITIES
## 1         NA         0      NA         NA     14   100 3   0          0
##   INJURIES PROPDMG PROPDMGEXP CROPDMG CROPDMGEXP WFO STATEOFFIC ZONENAMES
## 1       15      25          K       0         NA  NA         NA        NA
##   LATITUDE LONGITUDE LATITUDE_E LONGITUDE_ REMARKS REFNUM
## 1     3040      8812       3051       8806      NA      1
```

Because we are interested in the questions "which types of events re most harmful with respect to population health" and "which types of events have the greatest economic consequences", many columns containing datailed geographical information would not be useful. Since the most important things we are interested in are the harmfullness to population health and the economic impact by type, we only need the beginning and ending dates, types, fatalities, injuries, the 4 columns that are relavent to damage estimates, and the remarks (in case further investigation into the supplemental comments is needed). To avoid counting the indecies of particular column names, we could do the following:

```
idx1 <- which(names(data_row1)=="BGN_DATE")
idx2 <- which(names(data_row1)=="EVTYPE")
idx3 <- which(names(data_row1)=="END_DATE")
idx4 <- which(names(data_row1)=="FATALITIES")
idx5 <- which(names(data_row1)=="INJURIES")
idx6 <- which(names(data_row1)=="PROPDMG")
idx7 <- which(names(data_row1)=="PROPDMGEXP")
idx8 <- which(names(data_row1)=="CROPDMG")
idx9 <- which(names(data_row1)=="CROPDMGEXP")
idx0 <- which(names(data_row1)=="REMARKS")
```

These indecies will help when we specify which columns to skip when loading the data.

Loading and processing the data

Now we can read the desired (part of the) data by skipping unnecessary variables, and get a preview of the data (with remarks omitted here)

```
data_select <- read.csv("repdata-data-StormData.csv.bz2",
                        colClasses=c(rep("NULL",idx1-
1),"character",rep("NULL",idx2-idx1-1),"character",
                                     rep("NULL",idx3-idx2-
1),"character",rep("NULL",idx4-idx3-1),"numeric","numeric",

"numeric","factor","numeric","factor",rep("NULL",idx0-idx9-
1),"character","NULL"),
                        header=TRUE,as.is=TRUE)

head(data_select[,1:9])

##             BGN_DATE  EVTYPE END_DATE FATALITIES INJURIES PROPDMG
## 1  4/18/1950 0:00:00 TORNADO                   0       15    25.0
## 2  4/18/1950 0:00:00 TORNADO                   0        0     2.5
## 3  2/20/1951 0:00:00 TORNADO                   0        2    25.0
## 4   6/8/1951 0:00:00 TORNADO                   0        2     2.5
## 5 11/15/1951 0:00:00 TORNADO                   0        2     2.5
## 6 11/15/1951 0:00:00 TORNADO                   0        6     2.5
##   PROPDMGEXP CROPDMG CROPDMGEXP
## 1          K       0
## 2          K       0
## 3          K       0
```

```
## 4          K       0
## 5          K       0
## 6          K       0
```

Converting the dates to "Date" type in R, we do the following by creating a new object with the same information:

```
data_select_m <- data_select
data_select_m$BGN_DATE <- as.Date(data_select_m$BGN_DATE,"%m/%d/%Y")
data_select_m$END_DATE <- as.Date(data_select_m$END_DATE,"%m/%d/%Y")
```

NOAA provides another documentation on the relative completeness of data (i.e. event types) on its official website. Therefore we know a more comprehensive analysis, especially if we would like to make cross-sectional comparisons, will be better constructed if we use the data where all types are included in the same interval. Also, to reduce the impact of inflation/deflation of U.S. dollars over time, evaluating economic consequences in more recent years tends to make more sense. Finally, as science and technology develops over time, earlier fatality/injury patterns might become out of date, and to study the impact on population health, it would also be more of our interest to look at later years. We know from the NOAA official documentation that data became more complete since Jan 1996, so we subset our data as:

```
data_select_m2 <- data_select_m[data_select_m$BGN_DATE>="1996-01-01",]
```

Next, to have an understanding of the "PROPDMGEXP" and "CROPDMGEXP" column, here are the summaries:

```
summary(data_select_m2$PROPDMGEXP)

##             -      ?      +      0      1      2      3      4      5
## 276185      0      0      0      1      0      0      0      0      0
##      6      7      8      B      h      H      K      m      M
##      0      0      0     32      0      0 369938      0   7374

summary(data_select_m2$CROPDMGEXP)

##             ?      0      2      B      k      K      m      M
## 373069      0      0      0      4      0 278686      0   1771
```

By reading through the documentation posted by NOAA (Page 12), we know the "K/k", "M/m", and "B/b" are the damage estimate factors ("multipliers") here (for "thousands", "millions" and "billions"). To make our analysis more accurate, it is better to only use the rows where we know for sure what the element in the multiplier column means, so to clean up the fuzziness, we remove the "unclear" rows from the data (by subsetting):

```
data_select_m3 <-
data_select_m2[grep("H|h|K|k|M|m|B|b",data_select_m2$PROPDMGEXP),]
data_select_m3 <-
data_select_m3[grep("H|h|K|k|M|m|B|b",data_select_m3$CROPDMGEXP),]
```

To get an idea of the type frequencies, we have:

```
library(dplyr)
type_freq <- count(data_select_m3,EVTYPE)
type_freq <- type_freq[order(type_freq$n),]
dim(type_freq)
```

```
## [1] 68  2
```

So we now have 68 event types, which is more than the 48 types given by NOAA.

```
head(type_freq)
```

```
## Source: local data frame [6 x 2]
##
##                     EVTYPE     n
##                      (chr) (int)
## 1 ASTRONOMICAL HIGH TIDE     1
## 2                      FOG     1
## 3                   FREEZE     1
## 4            Frost/Freeze     1
## 5             GUSTY WINDS     1
## 6   Heavy Rain/High Surf     1
```

Reading through the type names (whereas the above is for demonstration), by comparing them with NOAA's Directive 10-1605, we know some of them are not officially recognized types, we could either eliminate these unofficial categories or investigate a little and find a proper official name for them from the document (and this requires searching for key words to identify the names with the documentation and may take a while, but we will finally be able to clean up the data); we also notice that some of the type names are in lower cases, some in upper - this might be one cause too so we put all of them in uppercase:

```
data_select_m3$EVTYPE <- toupper(data_select_m3$EVTYPE)
data_select_m4 <- subset(data_select_m3,EVTYPE!="ASTRONOMICAL HIGH
TIDE"&EVTYPE!="HEAVY RAIN/HIGH SURF"&EVTYPE!="TSTM WIND/HAIL")

data_select_m4$EVTYPE[data_select_m4$EVTYPE=="FOG"]<-"DENSE FOG"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="GUSTY WINDS"]<-"Strong Wind"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="VOLCANIC ASHFALL"]<-"Volcanic
Ash"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="HEAVY SURF/HIGH SURF"]<-"High
Surf"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="TYPHOON"]<-"Hurricane/Typhoon"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="DRY MICROBURST"]<-"Thunderstorm
Wind"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="STORM SURGE"]<-"Storm
Surge/Tide"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="SMALL HAIL"]<-"HAIL"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="FREEZE"]<-"Frost/Freeze"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="WILD/FOREST FIRE"]<-"Wildfire"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="HURRICANE"]<-
"Hurricane/Typhoon"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="EXTREME COLD"]<-"Extreme
```

```
Cold/Wind Chill"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="URBAN/SML STREAM FLD"]<-"Heavy
Rain"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="TSTM WIND"]<-"Thunderstorm
Wind"
data_select_m4$EVTYPE[data_select_m4$EVTYPE=="LANDSLIDE"]<-"DEBRIS FLOW"

data_select_m4$EVTYPE <- toupper(data_select_m4$EVTYPE)
data_select_m5 <- subset(data_select_m4,EVTYPE!="RIVER FLOODING"&EVTYPE!="ICY
ROADS"&EVTYPE!="RIVER FLOOD")

data_select_m5$PROPDMGEXP <- toupper(data_select_m5$PROPDMGEXP)
data_select_m5$CROPDMGEXP <- toupper(data_select_m5$CROPDMGEXP)

head(data_select_m5[,1:9]) # Remarks not printed here to make it look nicer
```

```
##          BGN_DATE             EVTYPE   END_DATE FATALITIES INJURIES PROPDMG
## 248768 1996-01-06       WINTER STORM 1996-01-07          0        0     380
## 248818 1996-02-19 THUNDERSTORM WIND 1996-02-19          0       15     500
## 248825 1996-03-05               HAIL 1996-03-05          0        0       2
## 248826 1996-03-05               HAIL 1996-03-05          0        0       2
## 248827 1996-03-05               HAIL 1996-03-05          0        0      15
## 248829 1996-03-05               HAIL 1996-03-05          0        0       5
##        PROPDMGEXP CROPDMG CROPDMGEXP
## 248768          K      38          K
## 248818          K       0          K
## 248825          K       1          K
## 248826          K       2          K
## 248827          K      10          K
## 248829          K       2          K
```

In particular, we eliminated the "TSTM WIND/HAIL" type because the consequences of this type are not clearly divided between thunderstorm wind and hail, thus not meeting the NOAA standard; we also removed the "RIVER FLOODING", "RIVER FLOOD" and "ICY ROADS" types because the separation of the consequences into explicit official event types appears unclear even in the supplemental remarks. To see how the rows with full remarks can be checked, as an example, one can do:

```
data_select_m3[data_select_m3$EVTYPE=="ICY ROADS",]
```

We can count the type frequency again, which is now consistent with the NOAA directive:

```
type_freq_2 <- count(data_select_m5, EVTYPE)
dim(type_freq_2)
```

```
## [1] 48  2
```

Now, we have a clean dataset, where we want to compute the damage estimates explicitly:

```
data_select_m5$PROPVALUE <- rep(-1, nrow(data_select_m5))
data_select_m5$CROPVALUE <- rep(-1, nrow(data_select_m5))
```

```
testFunc <- function(n,m)
ifelse(m=="H",n*100,ifelse(m=="K",n*1000,ifelse(m=="M",n*1000000,n*1000000000
)))
PROPVALUE <-
as.vector(apply(data_select_m5[,c("PROPDMG","PROPDMGEXP")],1,function(x)
testFunc(as.numeric(x["PROPDMG"]),x["PROPDMGEXP"])))
CROPVALUE <-
as.vector(apply(data_select_m5[,c("CROPDMG","CROPDMGEXP")],1,function(x)
testFunc(as.numeric(x["CROPDMG"]),x["CROPDMGEXP"])))
data_select_m5$PROPVALUE <- PROPVALUE
data_select_m5$CROPVALUE <- CROPVALUE
```

To make later calculations easier, we make a data.table by extracting the needed columns from the cleaned-up data set:

```
library(data.table)
data_select_m6 <- data.table(data_select_m5[,c(1,2,4,5,11,12)])
```

One important thing about making inferences from data, especially when we are more interested in "what is the consequence of a typical/average event in certain category" as opposed to "what is the most extreme case of a particular type of disaster", is that we eliminate outliers. Focusing on the "typical or on-average impact" of an event type helps us compare all the categories and provides an better overview of what types to pay more attention to on a yearly, or decadely, basis. Because properpty damage and crop damage both have multipliers in the thousands, millions and billions, they are much more versatile than fatalities and injuries, so we want to eliminate the outliers in these two columns (i.e. rull out those points that are over 3 standard deviations from the mean - this would still keep at least 88.89% of the data by Chebyshev's inequality, still a pretty good cut-off); after that, we add a column to denote the years (as opposed to dates):

```
data_select_m7 <- data_select_m6[,.SD[abs(PROPVALUE-
mean(PROPVALUE))<3*sd(PROPVALUE) & abs(CROPVALUE-
mean(CROPVALUE))<3*sd(CROPVALUE)],by=EVTYPE]

library(dplyr)
data_select_m7 <- mutate(data_select_m7,YEAR=format(BGN_DATE,"%Y"))
```

It would be helpful if we can see whether a type has relatively large/small impact over a period of time (i.e. whether its consequences stay at a relatively stable level, or it only has a very "unusual, outstanding year" but then significantly falls back in other years); so we group our data by type and year:

```
group1 <- group_by(data_select_m7,EVTYPE,YEAR)
summary1 <- summarize(group1,

meanFATAL=mean(FATALITIES),meanINJUR=mean(INJURIES),meanPROPD=mean(PROPVALUE)
,meanCROPD=mean(CROPVALUE),

sumFATAL=sum(FATALITIES),sumINJUR=sum(INJURIES),sumPROPD=sum(PROPVALUE),sumCR
OPD=sum(CROPVALUE))
```

```
summary1$YEAR <- as.Date(paste(summary1$YEAR,rep("-01-
01",nrow(summary1)),sep=""),"%Y-%m-%d")

head(summary1)
```

```
##          EVTYPE       YEAR meanFATAL  meanINJUR  meanPROPD meanCROPD
## 1 WINTER STORM 1996-01-01 0.1666667 0.16666667  333000.00 17166.667
## 2 WINTER STORM 1997-01-01 0.0000000 0.00000000 1516000.00 30000.000
## 3 WINTER STORM 1998-01-01 0.4000000 2.10000000   36400.00  5800.000
## 4 WINTER STORM 1999-01-01 0.0000000 0.00000000       0.00     0.000
## 5 WINTER STORM 2000-01-01 0.0000000 0.00000000   67428.57     0.000
## 6 WINTER STORM 2001-01-01 0.0000000 0.03846154   25384.62  2307.692
##   sumFATAL sumINJUR sumPROPD sumCROPD
## 1        1        1  1998000   103000
## 2        0        0  6064000   120000
## 3        4       21   364000    58000
## 4        0        0        0        0
## 5        0        0   472000        0
## 6        0        1   660000    60000
```

Now the summary is ready for plotting.

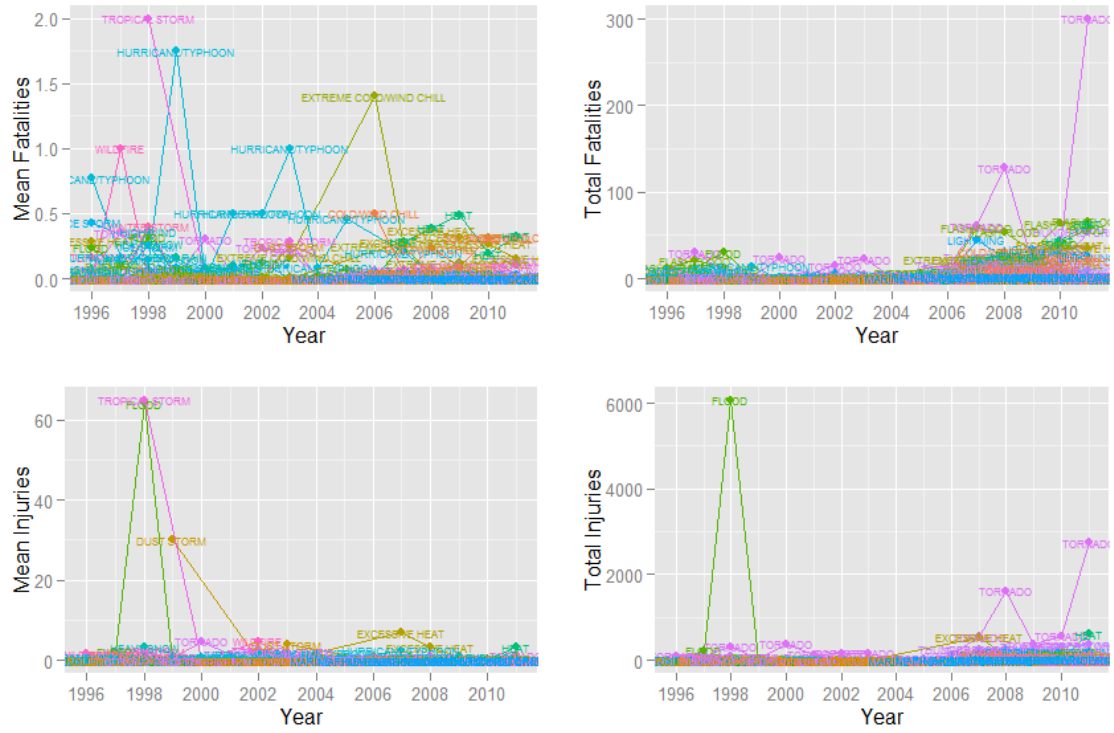
Results

Again, we are interested in "what is the consequence of a typical/average event in certain category on a yearly, or decadely, basis", and would like to make a comparison among types, it would be helpful to make line plots with years on the horizontal axis and variables that represent event impacts on the vertical axis. We include both the total and mean values:

```
library(ggplot2)
library(gridExtra)

p1 <-
ggplot(summary1,aes(x=YEAR,y=meanFATAL,color=EVTYPE))+geom_point()+geom_line(
)+xlab("Year")+ylab("Mean
Fatalities")+geom_text(aes(label=EVTYPE),size=2)+theme(legend.position="none"
)
p2 <-
ggplot(summary1,aes(x=YEAR,y=sumFATAL,color=EVTYPE))+geom_point()+geom_line()
+xlab("Year")+ylab("Total
Fatalities")+geom_text(aes(label=EVTYPE),size=2)+theme(legend.position="none"
)
p3 <-
ggplot(summary1,aes(x=YEAR,y=meanINJUR,color=EVTYPE))+geom_point()+geom_line(
)+xlab("Year")+ylab("Mean
Injuries")+geom_text(aes(label=EVTYPE),size=2)+theme(legend.position="none")
p4 <-
ggplot(summary1,aes(x=YEAR,y=sumINJUR,color=EVTYPE))+geom_point()+geom_line()
+xlab("Year")+ylab("Total
```

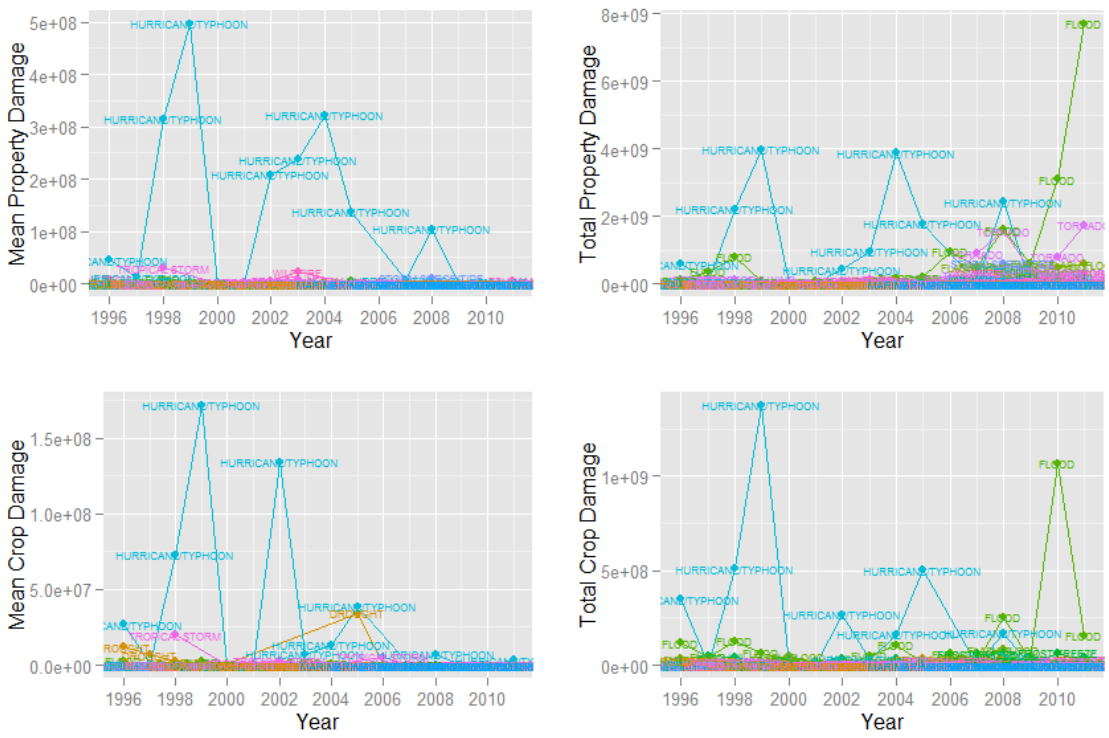
```
Injuries")+geom_text(aes(label=EVTYPE),size=2)+theme(legend.position="none")
grid.arrange(p1,p2,p3,p4,ncol=2)
```



From the panel we see, the "Mean Fatalities" plot looks quite "messy", the only type that constantly exhibits a relatively high mean fatality rate is "Hurricane/Typhoon" (whereas "Wildfire", "Tropical Storm" and "Extreme Cold/Wind Chill" only has a single peak then fall back, so these do not count); similarly, those stand out only once or twice over the years do not count in the mean injuries; and in the "Total Fatalities" and "Total Injuries" plots, "Tornado" is the only type that constantly stands out. So we conclude that: across the United States, "Tornado" and "Hurricane/Typhoon" are the two types most harmful with respect to population health.

```
p5 <-
ggplot(summary1,aes(x=YEAR,y=meanPROPD,color=EVTYPE))+geom_point()+geom_line(
)+xlab("Year")+ylab("Mean Property
Damage")+geom_text(aes(label=EVTYPE),size=2)+theme(legend.position="none")
p6 <-
ggplot(summary1,aes(x=YEAR,y=sumPROPD,color=EVTYPE))+geom_point()+geom_line()
+xlab("Year")+ylab("Total Property
Damage")+geom_text(aes(label=EVTYPE),size=2)+theme(legend.position="none")
p7 <-
ggplot(summary1,aes(x=YEAR,y=meanCROPD,color=EVTYPE))+geom_point()+geom_line(
)+xlab("Year")+ylab("Mean Crop
Damage")+geom_text(aes(label=EVTYPE),size=2)+theme(legend.position="none")
p8 <-
ggplot(summary1,aes(x=YEAR,y=sumCROPD,color=EVTYPE))+geom_point()+geom_line()
+xlab("Year")+ylab("Total Crop
```

```
Damage")+geom_text(aes(label=EVTYPE),size=2)+theme(legend.position="none")
grid.arrange(p5,p6,p7,p8,ncol=2)
```



From the panel we see, the plots are rather consistent, so we conclude that: across the United States, "Hurricane/Typhoon" and "Flood" are the two types with the greatest economic consequences.