Heavy Weather: Most damaging types of storm events in the U.S., 1950-2011

Synopsis

For disaster-preparedness efforts at all different levels of government, it is always wise to be forewarned and forearmed. Examining past natural disasters can be critical in determining how to allocate our budgets on emergency preparedness. Data analysis can play an important role in these activities. In order to assist with triage and the most effective use of resources, we analyzed the U.S. National Oceanic and Atmospheric Administration's (NOAA) storm database to address two questions:

Q1. For the United States over the period 1950-2011, which storm event types have had the most severe economic consequences, where 'economic consequences' is taken to mean the sum of the estimated dollar amounts of property damage, and of damage to crops?

Q2. For the United States over the period 1950-2011, which storm event types have been the worst in harm to human health? The NOAA records the number of fatalities and number of injuries for a given event, so the measure used was the sum of these two figures.

Data Processing

1. Manual preliminaries: bz2'ed data file

In spite of the fact that R allows direct downloads from the Internet, I do not use this here. Since there is no download-progress dialog in R Studio, it is too difficult to know whether anything is happening. Therefore, I will assume that you have downloaded the following file which is compressed in bz2 format, from here (https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2FStormData.csv.bz2). The remainder of this Data Processing section does not contain any additional manual steps - since it is reproducible research, following this brief calibration, the script will take you all the way through without intervention.

So, please put the bz2 file in a designated working folder of your choice.

Run the *setwd* command from R Studio to set the working directory to the folder containing the StormData file.

```
# If you wish to fill out this line with your own path, please do it here.
# Otherwise, leave it commented and execute setwd from the console.
#setwd(".........")
```

The next line may take 5 minutes to complete. If you wish, uncompress the bz2 file yourself, and then modify the read.csv line so that it reads in a csv instead. There are no additional manual interventions for the remainder of the analysis.

Other recommendations: You may want to close and reopen R studio before reading in the data.

You may want to close other running applications before reading in the data.

```
all_storm_data <- read.csv("repdata_data_StormData.csv.bz2")
```

Everything beyond this point is a detailed narrative of how the data processing is carried out.

2. Other automated prerequisite steps

Load up several "cool" utility libraries ending in -r:

```
library("tidyr");library("stringr");library("plyr");library("dplyr");library("knitr")
```

```
## Warning: package 'tidyr' was built under R version 3.2.4
```

```
## Warning: package 'plyr' was built under R version 3.2.4
```

```
## Warning: package 'dplyr' was built under R version 3.2.4
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

And ggplot, too.

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.2.4
```

Set an option in order to shut off scientific notation. This is so that the large dollar amounts will be easier to read.

```
options(scipen=999)
```

3. Economic consequences: resolve the multiplier issue

For Question 1, the fields representing amounts of money need a transformation in order to permit arithmetic. The NOAA stores these figures in two fields. For example, for the figure $250,000, there woud be a "250" in the field PROPDMG, and a "K" in the field PROPDMGEXP, signifying thousands. For the figure $2,000,000, PROPDMG would contain a "2", while the PROPDMGEXP would contain "M". The data for crop damage works in a comparable fashion and is stored in CROPDMG and CROPDMGEXP. Therefore, the first series of steps is

to do the following:

convert certain established code letters into their common meaning.

"B" and "b" will be coded as one billion in a way that will permit arithmetic (1000000000).

"M" and "m" will be coded as one million in a way that will permit arithmetic (1000000).

"K" and 'k" will be coded as one thousand in a way that will permit arithmetic (1000)

Any other character in the two EXP fields will be converted into a "1" so that any damage amount for that incident will be assumed to have been that literal value. Once the multipliers are ready, multiply them by the coded numeric figures in PROPDMG and CROPDMG respectively. At this point, it is possible to use ddply and create one rollup for property damage amounts by event type, and one for crop damage amounts by event type.

```
ec_prop_KMB <- filter(all_storm_data, PROPDMGEXP %in% c('K','M','B','k','m','b'))
ec_prop_other <- filter(all_storm_data, !(PROPDMGEXP %in% c('K','M','B','k','m','b'))
)
ec_prop_K <- subset(ec_prop_KMB,PROPDMGEXP %in% c('k','K'))
ec_prop_M <- subset(ec_prop_KMB,PROPDMGEXP %in% c('m','M'))
ec_prop_B <- subset(ec_prop_KMB,PROPDMGEXP %in% c('b','B'))
ec_prop_K <- mutate(ec_prop_K,multiplier=1000)
ec_prop_M <- mutate(ec_prop_M,multiplier=1000000)
ec_prop_B <- mutate(ec_prop_B,multiplier=1000000000)
ec_prop_other <- mutate(ec_prop_other,multiplier=1)
ec_prop <- rbind(ec_prop_K,ec_prop_M,ec_prop_B,ec_prop_other)
ec_prop <- mutate(ec_prop,property_damage_in_dollars = PROPDMG * multiplier)
ec_prop_rollup <- ddply(ec_prop,.(EVTYPE),summarize,ec_prop=sum(property_damage_in_do
llars))

ec_crop_KMB <- filter(all_storm_data, CROPDMGEXP %in% c('K','M','B','k','m','b'))
ec_crop_other <- filter(all_storm_data, !(CROPDMGEXP %in% c('K','M','B','k','m','b'))
)
ec_crop_K <- subset(ec_crop_KMB,CROPDMGEXP %in% c('k','K'))
ec_crop_M <- subset(ec_crop_KMB,CROPDMGEXP %in% c('m','M'))
ec_crop_B <- subset(ec_crop_KMB,CROPDMGEXP %in% c('b','B'))
ec_crop_K <- mutate(ec_crop_K,multiplier=1000)
ec_crop_M <- mutate(ec_crop_M,multiplier=1000000)
ec_crop_B <- mutate(ec_crop_B,multiplier=1000000000)
ec_crop_other <- mutate(ec_crop_other,multiplier=1)
ec_crop <- rbind(ec_crop_K,ec_crop_M,ec_crop_B,ec_crop_other)
ec_crop <- mutate(ec_crop,crop_damage_in_dollars = CROPDMG * multiplier)
ec_crop_rollup <- ddply(ec_crop,.(EVTYPE),summarize,ec_crop=sum(crop_damage_in_dollar
s))
```

4. Economic consequences: Find the top ten worst events by a summed figure, then plot seperately

We have these two subgroups within the economic-consequences data. What if the list of worst event types varies slightly different between these two? The method used here is to sum the two figures and then make a cutoff afterwards. There's an arbitrary cutoff at the top ten event types (and the same is used below for population health). This is definitely enough to tell the story, since the question asks for the absolute worst. So

the next piece of analysis is to use merge on the two rollups with the event type field (EVTYPE) as the key. Often it is considered to be undesireable to use text strings as a key in joins, but because these two rollups came from the same original, these strings are reliable enough. Slight spelling variations are often a problem when joining on strings but that isn't a problem here.

```
ec_prop_crop <- merge(ec_prop_rollup,ec_crop_rollup,by="EVTYPE")
ec_prop_crop$sum_for_worst_events <- ec_prop_crop$ec_prop + ec_prop_crop$ec_crop
ec_prop_crop <- arrange(ec_prop_crop,desc(sum_for_worst_events))
ec_prop_crop <- head(ec_prop_crop,10)
ec_prop_crop <- arrange(ec_prop_crop,sum_for_worst_events)
```

4. Economic consequences: Prepare a bar plot that answers Q1

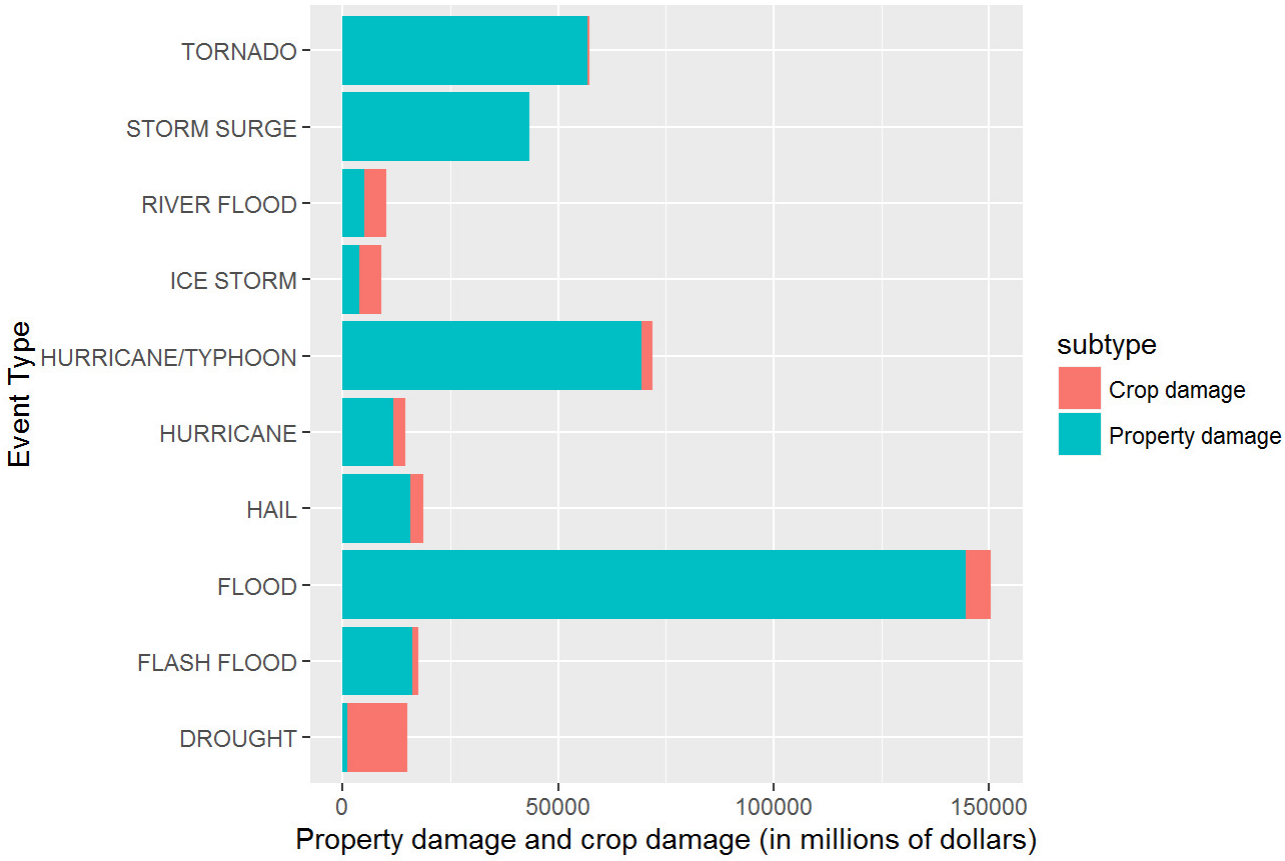
The rolled-up value will not be used for the plot itself, because that way, there can be a more granular result that shows the two buckets. Therefore, the merged table is now broken back out to a tall-and-skinny table with a 'subcategory' tag so that ggplot can generate the multicolored bars out of that. I didn't use "melt" but it's a similar idea. This section concludes with the plot which answers question 1: "Which storm event types have had the most severe economic consequences?"

```
ec_evtype_for_plot <- select(ec_prop_crop,1)
ec_prop_for_plot <- select(ec_prop_crop,2)
ec_crop_for_plot <- select(ec_prop_crop,3)


prop_data_for_plot <- cbind(ec_evtype_for_plot,ec_prop_for_plot)
prop_data_for_plot$subtype <- "Property damage"
crop_data_for_plot <- cbind(ec_evtype_for_plot,ec_crop_for_plot)
crop_data_for_plot$subtype <- "Crop damage"
prop_data_for_plot <- rename(prop_data_for_plot,amount = ec_prop)
crop_data_for_plot <- rename(crop_data_for_plot,amount = ec_crop)
data_for_ec_plot <- rbind(prop_data_for_plot,crop_data_for_plot)

# figures are divided by 1,000,000, and then the units on the plot are given in milli
ons of dollars
data_for_ec_plot$millions <- data_for_ec_plot$amount / 1000000
g <- ggplot(data_for_ec_plot,aes(EVTYPE,millions))
g + geom_bar(stat="identity",aes(fill=subtype))+coord_flip()+labs(title="Figure 1. Ec
onomic Consequences from Storm Event Types\nin the United States, 1950-2011", y="Prop
erty damage and crop damage (in millions of dollars)",x="Event Type")
```

Figure 1. Economic Consequences from Storm Event Types in the United States, 1950-2011



5. Population health: Using the same techniques as above, prepare a bar plot that answers Q2

This section will be a little more terse because the methods follow what is described above. To answer the second question, I used the fields INJURIES and FATALITIES, each of which are represented in numbers of people either hurt or killed, respectively. As described above, I have not delved into subtle questions about the relative weighting of injuries versus fatalities in determining the worst event types. I make an assumption that the very biggest and very worst natural disasters, tragically, involved both deaths and injuries. Therefore, these two different kinds of counts are added together in a comparable fashion to the method used above. I created rollups of total injuries and fatalities by event type. The two rollups are then merged (with EVTYPE as the key) so that a cutoff can be found based on the worst event types based on the sum of both subtypes. As above, the plot itself is done from the separate figures so that the plot can be a visual representation of both injuries and fatalities, separately, while also showing them together.

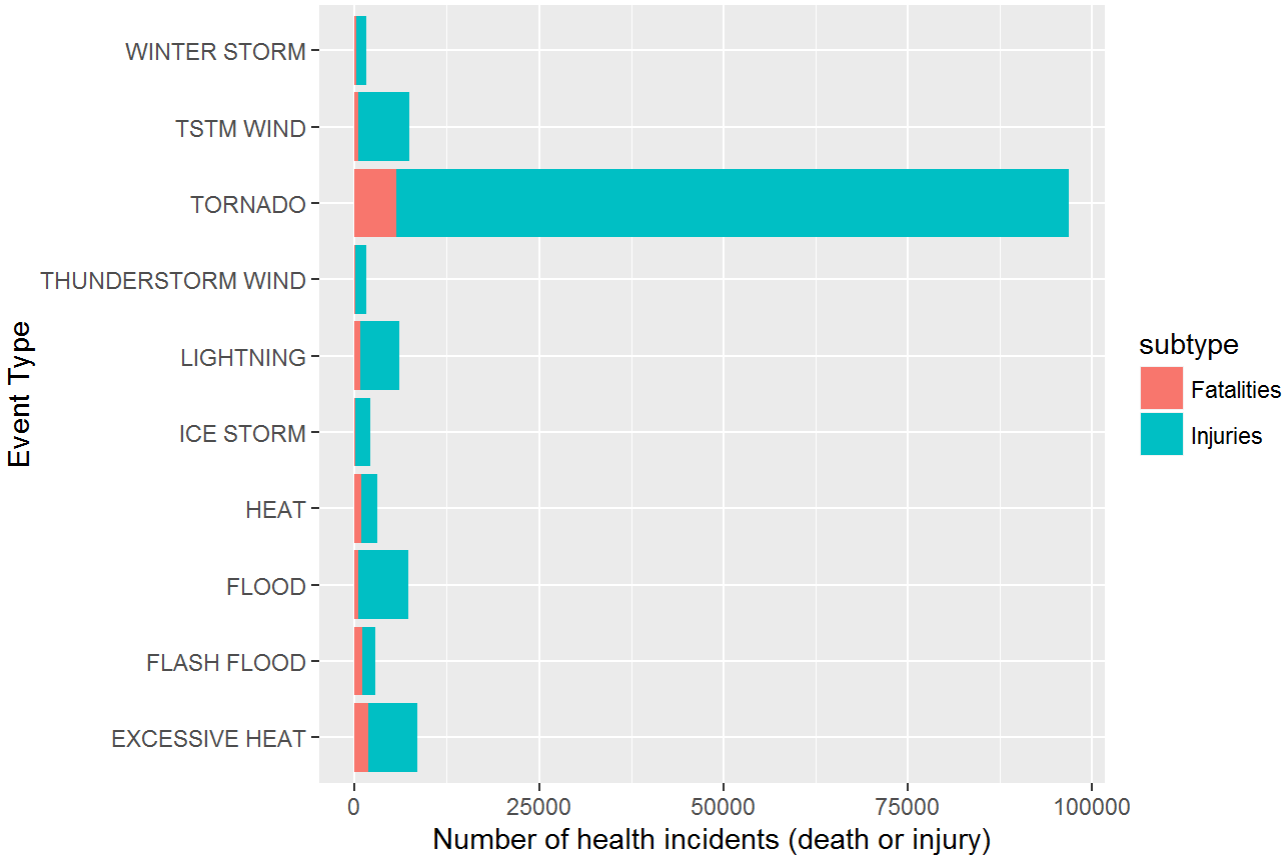
```
# Done with dollar amounts - now work on numbers of fatalities and injuries

ph_fatalities_rollup <- ddply(all_storm_data,.(EVTYPE),summarize,ph_f=sum(FATALITIES)
)
ph_injuries_rollup <- ddply(all_storm_data,.(EVTYPE),summarize,ph_i=sum(INJURIES))
ph_rollup <- merge(ph_fatalities_rollup,ph_injuries_rollup,by="EVTYPE")
ph_rollup$sum_for_worst_events <- ph_rollup$ph_f + ph_rollup$ph_i
ph_rollup <- arrange(ph_rollup,desc(sum_for_worst_events))
ph_rollup <- head(ph_rollup,10)
ph_rollup <- arrange(ph_rollup,sum_for_worst_events)
ph_event_type <- select(ph_rollup,1)
ph_fatalities <- select(ph_rollup,2)
ph_injuries <- select(ph_rollup,3)

fatalities_for_plot <- cbind(ph_event_type,ph_fatalities)
fatalities_for_plot$subtype <- "Fatalities"
injuries_for_plot <- cbind(ph_event_type,ph_injuries)
injuries_for_plot$subtype <- "Injuries"

fatalities_for_plot <- rename(fatalities_for_plot,count = ph_f)
injuries_for_plot <- rename(injuries_for_plot,count = ph_i)
data_for_ph_plot <- rbind(fatalities_for_plot,injuries_for_plot)
g <- ggplot(data_for_ph_plot,aes(EVTYPE,count))
g + geom_bar(stat="identity",aes(fill=subtype))+coord_flip()+labs(title="Figure 2. Po
pulation-health consequences from Storm Event Types,\nUnited States, 1950-2011", y="N
umber of health incidents (death or injury)",x="Event Type")
```

Figure 2. Population-health consequences from Storm Event Types, United States, 1950-2011



Results

Based on the plots, a picture emerges which answers the questions. Which storm event types have had the most severe economic consequences? Flooding has caused the most severe economic consequences. Hurricanes which were, for part of their existence, a typhoon and thus were coded as HURRICANE/TYPHOON, were also among the most severe. Tornadoes have been the third worst event type.

Which storm event types have been the worst, measured in harm to human health? Tornadoes vastly exceed any other type.

Based on past precedent as demonstrated in this analysis, it is likely that preparedness resources devoted to these event types will be borne out as a wise decision in the 21st century.