Analysis of Damage due to Weather Events

Ian Gordon

2015-02-17

Synopsis

Using the National Oceanic and Atmospheric Administration (NOAA) National Weather Service storm data, we will analyze storm records to determine which types of weather events are responsible for the greatest number of fatalities and injuries, as well as crop and property damage. The data identifies major weather events as recorded from 1950 to 2011. Knowing which types of weather events result in the most damage (to both people and property) can be useful in planning responses to such events when they occur.

From the the dataset details at http://www.ncdc.noaa.gov/stormevents/details.jsp, we see that the data for 1950 to 1954 consists of tornado events only, from 1955 to 1992, only tornado, thunderstorm wind, and hail events were recorded, and all event types recorded from 1966. As such, this analysis will focus only on weather events from 1996 and later. This range of events is most useful for planning purposes as it is the most complete.

For this range of years, we can see that tornadoes were responsible for the vast majority of injuries,and heat and tornados for most fatalities, while floods by far account for most property damage with drought responsible for most crop damage.

Data Processing

The data set will be loaded and preprocessed to exclude unneeded columns and to reduce the data to a set containing just the columns that will be used in this analysis. We then exclude observations that lie outside the range of years that we are interested in, namely observations prior to 1996.

One problem with the data is the inconsistency in the names of the weather events. Some events appear multiple times, with names in different forms or spellings. For example, there are a number of different entries that relate to thunderstorms - some have obvious spelling mistakes, some have extra spaces, some contain details about wind speed. In a lot of cases, such a typo appears only once in the data, so cleaning up that event name will have no significant impact on the analysis. Using TSTM WIND and its variations as an example, we see 219940 observations of TSTM WIND and 39 of TSTM WIND (G45) and 10 of TSTM (G40). If we clean these event names so that they can be counted with the TSTM WIND observations, the impact is truly negligable. Furthermore, changing the event names could, in some cases, also change their meanings. For example, response to a flood may be different from the response required for a flash flood. Rather than try to standardize all names (a complex and potentially error-prone operation), I have closen to standardize only those that are amongst the most common (meaning that after identifying the most common weather events, I went back and cleaned up only those names).

The CSV file containing the data is assumed to be in the current directory.

Libraries

```
library(ggplot2)
library(gridExtra) # Supports multiple plots in one figure.
```

```
## Loading required package: grid
```

```
library(scales) # Allows improved formatting of labels in plot axis.
```

```
rawData <- read.csv("repdata_data_StormData.csv", header=TRUE, stringsAsFactors=FALSE)

# Separate the year from the date and store in a new column. This will make the task of restricting the
# data to the desired range of years easier.
# See http://r.789695.n4.nabble.com/Year-and-Month-extraction-from-Date-object-td904011.html for an
# explanation of how this works.
rawData$year <- as.numeric(format(as.Date(rawData$BGN_DATE, format="%m/%d/%Y %H:%M:%S"), "%Y"))
byYearData <- subset(rawData, year >= 1996)

# Next, extract only the columns that are needed for this analysis.
desiredColumns <- c("EVTYPE", "FATALITIES", "INJURIES", "PROPDMG", "PROPDMGEXP", "CROPDMG", "CROPDMGEXP
data <- byYearData[desiredColumns]
```

The next step is to try to standardize the event type names of the most commom weather events.

```
data$EVTYPE <- toupper(data$EVTYPE)

data$EVTYPE[data$EVTYPE == "TSTM WIND"] <- "THUNDERSTORM WIND"
data$EVTYPE[data$EVTYPE == "RIP CURRENTS"] <- "RIP CURRENT"
data$EVTYPE[data$EVTYPE == "HURRICANE/TYPHOON"] <- "HURRICANE"
```

Finally, we need to cleanup the property and crop damage estimates, converting them to a standard format that we can use. The EXP field in the original data does not seem to be clearly defined. As such, I will focus on the more obvious values (K (thousands), M (millions), and B (billions)). Other values will be assumed to have no significant impact on the other values.

```
normalizeCost <- function(damageAmount, exp) {
    options(scipen=999) # Avoid scientific notation
    if (exp == "K") {
        damageAmount <- damageAmount * 1000
    } else if (exp == "M") {
        damageAmount <- damageAmount * 1000000
    } else if (exp == "B") {
        damageAmount <- damageAmount * 1000000000
    } else {
        damageAmount
    }
}

# Add new columns to hold the cleaned values.
data$PropDamageAmount <- mapply(normalizeCost, data$PROPDMG, data$PROPDMGEXP)
data$CropDamageAmount <- mapply(normalizeCost, data$CROPDMG, data$CROPDMGEXP)
```

Results

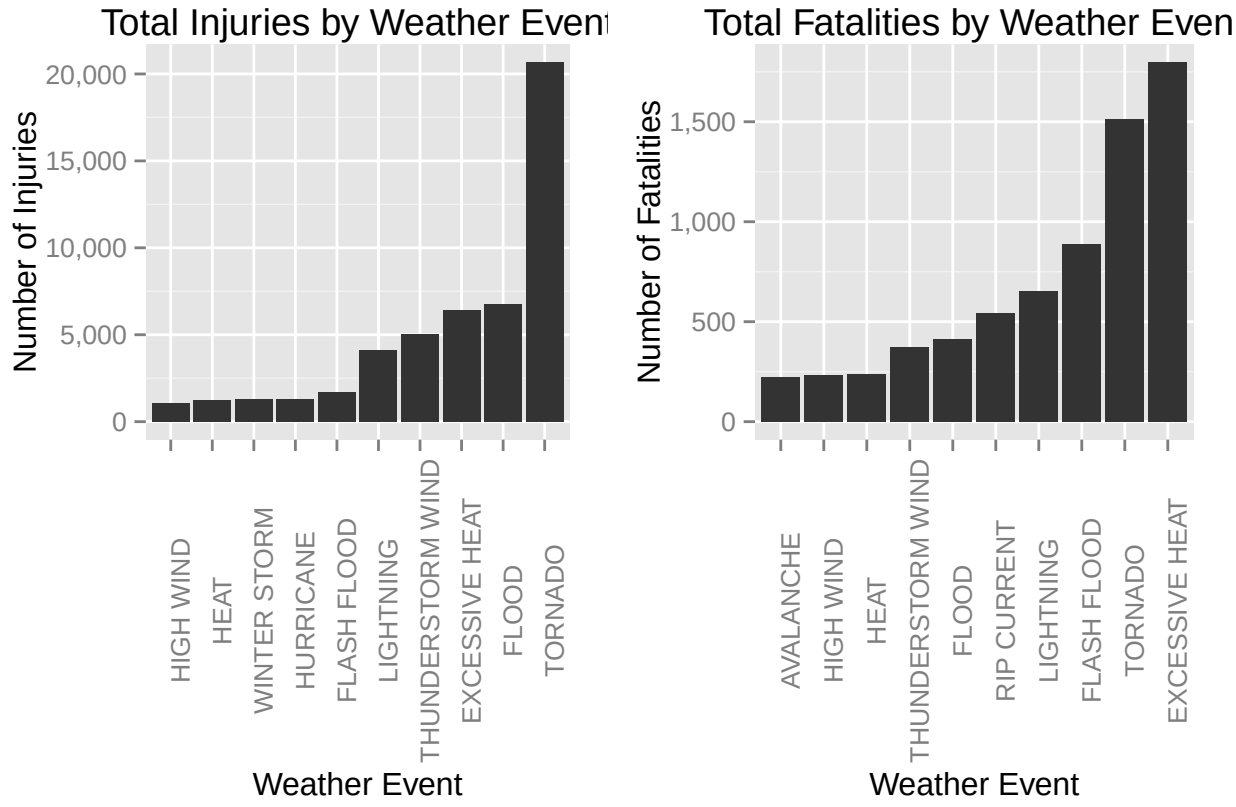
Which weather events are responsible for the greatest numbers of injuries and fatalities and which are responsible for the greatest amounts of property and crop damage?

To answer this question, we first separate the data related to injuries and fatalities, and to property and crop damage. Results will be shown in figures, each containing a pair of graphs. As the ordering of events within

each category differs, this seems a cleaner way of presenting the results, although the reader must remain conscious of the differences in scale of each plot. (And this fits the assignment specifications as each figure is allowed to contain multiple plots.)

Fatalities and Injuries

```
# For each category, find the sums for each type of weather event that relate that category.
# Order the results from highest to lowest.
# Extract only the top ten events to be plotted.
fatalities <- aggregate(FATALITIES ~ EVTYPE, data=data, FUN=sum)
fatalities <- fatalities[order(fatalities$FATALITIES, decreasing=TRUE),]
fatalities <- fatalities[1:10,] # Limit to the 10 most dangerous events

injuries <- aggregate(INJURIES ~ EVTYPE, data=data, FUN=sum)
injuries <- injuries[order(injuries$INJURIES, decreasing=TRUE),]
injuries <- injuries[1:10,] # Limit to the 10 most dangerous events

# See http://rstudio-pubs-static.s3.amazonaws.com/7433_4537ea5073dc4162950abb715f513469.html for an
# explanation of how to change the default ordering of the x-axis to avoid the default alphabetic
# ordering.
injuries$EVTYPE <- factor(injuries$EVTYPE, levels=injuries$EVTYPE[order(injuries$INJURIES)])
fatalities$EVTYPE <- factor(fatalities$EVTYPE, levels=fatalities$EVTYPE[order(fatalities$FATALITIES)])

# Plot side-by-side bar graphs to give a good indication of the effects of total injuries and fatalitie
p1 <- ggplot(data=injuries, aes(x=EVTYPE, y=INJURIES)) + geom_bar(stat="identity") +
    ggtitle("Total Injuries by Weather Event") + xlab("Weather Event") +
    theme(axis.text.x = element_text(angle=90)) + ylab("Number of Injuries") + scale_y_continuous(labels

p2 <- ggplot(data=fatalities, aes(x=EVTYPE, y=FATALITIES)) + geom_bar(stat="identity") +
    ggtitle("Total Fatalities by Weather Event") + xlab("Weather Event") +
    theme(axis.text.x = element_text(angle=90)) + ylab("Number of Fatalities") + scale_y_continuous(labe

# Plot multiple graphs in figure for fatalities and injuries.
grid.arrange(p1, p2, ncol=2)
```



Crop and Property Damage

```
propertyDamage <- aggregate(PropDamageAmount ~ EVTYPE, data=data, FUN=sum)
propertyDamage <- propertyDamage[order(propertyDamage$PropDamageAmount, decreasing=TRUE),]
propertyDamage <- propertyDamage[1:10,] # Limit to the 10 most dangerous events

cropDamage <- aggregate(CropDamageAmount ~ EVTYPE, data=data, FUN=sum)
cropDamage <- cropDamage[order(cropDamage$CropDamageAmount, decreasing=TRUE),]
cropDamage <- cropDamage[1:10,]

# See http://rstudio-pubs-static.s3.amazonaws.com/7433_4537ea5073dc4162950abb715f513469.html for an
# explanation of how to change the default ordering of the x-axis to avoid the default alphabetic
# ordering.
propertyDamage$EVTYPE <- factor(propertyDamage$EVTYPE, levels=propertyDamage$EVTYPE[order(propertyDamag
cropDamage$EVTYPE <- factor(cropDamage$EVTYPE, levels=cropDamage$EVTYPE[order(cropDamage$CropDamageAmou

# Plot side-by-side bar graphs to give a good indication of the effects of total injuries and fatalitie
# Scale the dollar amounts to multiples of one million to enhance readability.
# Use the theme() function to rotate the event names 90 degrees to make them more readable.
# Also, for more readability, the dollar values on the y-axis will be formatted with commas.
# See http://stackoverflow.com/questions/11610377/how-do-i-change-the-formatting-of-numbers-on-an-axis-
p1 <- ggplot(data=propertyDamage, aes(x=EVTYPE, y=PropDamageAmount / 1000000)) + geom_bar(stat="identit
    ggtitle("Total Property Damage") + xlab("Weather Event") +
    theme(axis.text.x = element_text(angle=90)) + ylab("Damage (in million dollars)") + scale_y_continu
```

```
p2 <- ggplot(data=cropDamage, aes(x=EVTYPE, y=CropDamageAmount / 1000000)) + geom_bar(stat="identity") +
    ggtitle("Total Crop Damage") + xlab("Weather Event") +
    theme(axis.text.x = element_text(angle=90)) + ylab("Damage (in million dollars)") + scale_y_continu

# Plot multiple graphs in figure for property and crop damage.
grid.arrange(p1, p2, ncol=2)
```