

Severe Weather Events' Effects on Human Health and Local Economies

Synopsis

Using data from the National Oceanic and Atmospheric Administration (NOAA), this study assesses the overall impact of a variety of weather/natural disaster events upon the health and economic well being of the United States over the period from 1950 through 2011. As a result of the study, we determined that the greatest threat to population health are excessive heat events. From an economic impact perspective, droughts drive the greatest loss in economic value.

Data Processing

The following documents the steps to ingest, clean, and analyze the weather data provided by NOAA. This data catalogs various weather events from 1950 thorough late 2011, as well as measures defining the impact of each storm on human health/safety and economic damage.

Data Ingestion

NOAA's data is available for this study through the hosting platform cloudfront.net. Below, we note this resource and download the content directly. Noting the date and time of the download, we store the raw data after unzipping the file and pulling in the .csv file within.

```
library(R.utils)
library(lubridate)
URL <- "http://d396qusza40orc.cloudfront.net/repdata%2Fdata%2FStormData.csv.bz2"
download.time <- date()
download.file(URL, ".\\data\\StormData.csv.bz2")
bunzip2(".\\data\\StormData.csv.bz2", ".\\data\\StormData.csv")
raw.storm.data <- read.csv(".\\data\\StormData.csv")
```

Processing

In the next phase, we start with the raw data to focus in on and clean the variables and observations of interest to the purpose of the study. The code for this process is below, with the reasoning and light description of the approach following.

```
library(stringr)
library(lubridate)
## examine the variables present in the whole dataset
str(raw.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 ...
## $ 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",...: 135
13 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 ...
## $ EVTYPE : Factor w/ 985 levels " HIGH SURF ADVISORY",...: 834 834 834 834 834 834 834 834 8
34 834 ...
## $ 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 17 ...
## $ 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 ...
## $ 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/ 436774 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 ...

```

```

## subset the raw data to just the variables of interest
storm.data <- raw.storm.data[names(raw.storm.data) %in% c("BGN_DATE", "EVTYPE",
                                                         "FATALITIES", "INJURIES",
                                                         "PROPDMG", "PROPDMGEXP",
                                                         "CROPDMG", "CROPDMGEXP",
                                                         "REFNUM")]

## convert the date field to the proper format
## rename to simply DATE
storm.data$DATE <- mdy_hms(storm.data$BGN_DATE)
## note the discrepancy in the stated number (48) of EVTYPE categories and the
## number actually present
length(unique(storm.data$EVTYPE))

## [1] 985

## Execute rudimentary clean up steps to handle "easy cases" of lower case and
## leading white space
storm.data$EVTYPE <- toupper(storm.data$EVTYPE)
storm.data$EVTYPE <- str_trim(storm.data$EVTYPE)
## Convert disparate EVTYPE entries to logical groups:
## 1) initialize new variable
storm.data$EVTYPE.groups <- storm.data$EVTYPE
## 2) convert anything referencing cold or chill to simply "COLD"
storm.data$EVTYPE.groups[grepl("COLD", storm.data$EVTYPE.groups) == TRUE |
                         grepl("CHILL", storm.data$EVTYPE.groups) == TRUE |
                         grepl("FREEZE", storm.data$EVTYPE.groups) == TRUE |
                         grepl("FROST", storm.data$EVTYPE.groups) == TRUE |
                         grepl("RECORD LOW", storm.data$EVTYPE.groups) == TRUE |
                         grepl("COOL", storm.data$EVTYPE.groups) == TRUE |
                         grepl("HYPOTHERMIA", storm.data$EVTYPE.groups) == TRUE |
                         grepl("LOW [T][E][M][P]", storm.data$EVTYPE.groups) == TRUE] <- "COLD"
## 3) convert anything referencing wind to simply "WIND"
storm.data$EVTYPE.groups[grepl("[W][I]?[N][D]", storm.data$EVTYPE.groups) == TRUE |
                         grepl("DRY MICROBURST", storm.data$EVTYPE.groups) == TRUE |
                         grepl("TURBULENCE", storm.data$EVTYPE.groups) == TRUE] <- "WIND"
## 4) convert anything referencing rising water to simply "FLOOD"
storm.data$EVTYPE.groups[grepl("SURGE", storm.data$EVTYPE.groups) == TRUE |
                         grepl("FLOOD", storm.data$EVTYPE.groups) == TRUE |
                         grepl("WAVE", storm.data$EVTYPE.groups) == TRUE |
                         grepl("TIDE", storm.data$EVTYPE.groups) == TRUE |
                         grepl("TSUNAMI", storm.data$EVTYPE.groups) == TRUE |
                         grepl("SURF", storm.data$EVTYPE.groups) == TRUE |
                         grepl("FLD", storm.data$EVTYPE.groups) == TRUE |
                         grepl("SWELL", storm.data$EVTYPE.groups) == TRUE |
                         grepl("URBAN", storm.data$EVTYPE.groups) == TRUE |
                         grepl("WATER", storm.data$EVTYPE.groups) == TRUE |
                         grepl("STREAM", storm.data$EVTYPE.groups) == TRUE |

```

```

        grepl("DAM", storm.data$EVTYPE.groups) == TRUE] <- "FLOOD"

## 5) convert anything referencing hurricanes or tropical storms to simply "HURRICANE/TROP"
storm.data$EVTYPE.groups[grepl("HURRICANE", storm.data$EVTYPE.groups) == TRUE |
        grepl("TYPHOON", storm.data$EVTYPE.groups) == TRUE |
        grepl("FLOYD", storm.data$EVTYPE.groups) == TRUE |
        grepl("TROPICAL", storm.data$EVTYPE.groups) == TRUE |
        grepl("[T][S][T][M]", storm.data$EVTYPE.groups) == TRUE] <- "HURRICANE/TROP"

"

## 6) convert anything referencing tornados to simply "TORNADO"
storm.data$EVTYPE.groups[grepl("[N][A][D][O]", storm.data$EVTYPE.groups) == TRUE |
        grepl("[T][O][R][N]", storm.data$EVTYPE.groups) == TRUE |
        grepl("[S][P][O][U][T]", storm.data$EVTYPE.groups) == TRUE |
        grepl("FUNNEL", storm.data$EVTYPE.groups) == TRUE |
        grepl("WALL CLOUD", storm.data$EVTYPE.groups) == TRUE] <- "TORNADO"

## 7) convert anything referencing snow/ice to simply "WINTER PRECIP"
storm.data$EVTYPE.groups[grepl("[S][N][O][W]", storm.data$EVTYPE.groups) == TRUE |
        grepl("WINTRY", storm.data$EVTYPE.groups) == TRUE |
        grepl("[I][C][EY]", storm.data$EVTYPE.groups) == TRUE |
        grepl("BLIZZARD", storm.data$EVTYPE.groups) == TRUE |
        grepl("SLEET", storm.data$EVTYPE.groups) == TRUE |
        grepl("WINTER", storm.data$EVTYPE.groups) == TRUE |
        grepl("[F][R][E][E][Z]", storm.data$EVTYPE.groups) == TRUE |
        grepl("GLAZE", storm.data$EVTYPE.groups) == TRUE |
        grepl("[M][I][X]", storm.data$EVTYPE.groups) == TRUE] <- "WINTER PREC"

## 8) convert anything referencing rain to simply "RAIN"
storm.data$EVTYPE.groups[grepl("[R][A][I][N]", storm.data$EVTYPE.groups) == TRUE |
        grepl("SHOWER", storm.data$EVTYPE.groups) == TRUE |
        grepl("PRECIP", storm.data$EVTYPE.groups) == TRUE |
        grepl("[B][U][R][S][T]", storm.data$EVTYPE.groups) == TRUE |
        grepl("[W][E][T]", storm.data$EVTYPE.groups) == TRUE] <- "RAIN"

## 9) convert anything referencing hot or heat to simply "HEAT"
storm.data$EVTYPE.groups[grepl("HEAT", storm.data$EVTYPE.groups) == TRUE |
        grepl("HOT", storm.data$EVTYPE.groups) == TRUE |
        grepl("RECORD HIGH", storm.data$EVTYPE.groups) == TRUE |
        grepl("HIGH", storm.data$EVTYPE.groups) == TRUE |
        grepl("HYPERTHERMIA", storm.data$EVTYPE.groups) == TRUE |
        grepl("WARM", storm.data$EVTYPE.groups) == TRUE] <- "HEAT"

## 10) convert anything referencing hail to simply "HAIL"
storm.data$EVTYPE.groups[grepl("HAIL", storm.data$EVTYPE.groups) == TRUE] <- "HAIL"

## 11) convert anything referencing fire to simply "FIRE"
storm.data$EVTYPE.groups[grepl("[F][I][R][E]", storm.data$EVTYPE.groups) == TRUE] <- "FIRE"

## 12) convert anything referencing storms to simply "STORM"
storm.data$EVTYPE.groups[grepl("[S][T][O][R][M]", storm.data$EVTYPE.groups) == TRUE |
        grepl("LIGHTNING", storm.data$EVTYPE.groups) == TRUE |
        grepl("LIGHTING", storm.data$EVTYPE.groups) == TRUE |
        grepl("LIGNTNING", storm.data$EVTYPE.groups) == TRUE] <- "STORM"

## 13) convert anything referencing sliding/avalanche to simply "COLLAPSE"
storm.data$EVTYPE.groups[grepl("AVALANC[H]?E", storm.data$EVTYPE.groups) == TRUE |
        grepl("LANDSLUMP", storm.data$EVTYPE.groups) == TRUE |

```

```

      grepl("EROSI[O]?N", storm.data$EVTYPE.groups) == TRUE |
      grepl("ERUPTION", storm.data$EVTYPE.groups) == TRUE |
      grepl("[S][L][I][D][E]", storm.data$EVTYPE.groups) == TRUE] <- "COLLAPSE"
## 14) convert anything referencing ocean conditions to simply "MARINE CONDITIONS"
storm.data$EVTYPE.groups[grepl("MARINE", storm.data$EVTYPE.groups) == TRUE |
      grepl("CURRENT[S]?", storm.data$EVTYPE.groups) == TRUE |
      grepl("DROWNING", storm.data$EVTYPE.groups) == TRUE |
      grepl("RED FLAG", storm.data$EVTYPE.groups) == TRUE |
      grepl("SEAS", storm.data$EVTYPE.groups) == TRUE] <- "MARINE CONDITIONS"
## 15) convert anything referencing low visibility conditions to simply "LOW VIZ"
storm.data$EVTYPE.groups[grepl("FOG", storm.data$EVTYPE.groups) == TRUE |
      grepl("VOG", storm.data$EVTYPE.groups) == TRUE |
      grepl("[D][U][S][T]", storm.data$EVTYPE.groups) == TRUE |
      grepl("SMOKE", storm.data$EVTYPE.groups) == TRUE |
      grepl("NORTHERN LIGHTS", storm.data$EVTYPE.groups) == TRUE |
      grepl("ASH", storm.data$EVTYPE.groups) == TRUE] <- "LOW VIZ"
## 16) convert anything referencing dry conditions to simply "DROUGHT"
storm.data$EVTYPE.groups[grepl("[D][R][Y]", storm.data$EVTYPE.groups) == TRUE |
      grepl("DROUGHT", storm.data$EVTYPE.groups) == TRUE] <- "DROUGHT"
## 17) create object of valid event types based on 2-16 above and mark any
## other event left as "NA"
events <- c("COLD", "WIND", "FLOOD", "HURRICANE/TROP", "TORNADO", "WINTER PREC",
            "RAIN", "HEAT", "HAIL", "FIRE", "STORM", "COLLAPSE", "MARINE CONDITIONS",
            "LOW VIZ", "DROUGHT")

storm.data$EVTYPE.groups[(storm.data$EVTYPE.groups %in% events) == FALSE] <- "NA"
## with the recoding complete, convert the variable to a factor
storm.data$EVTYPE.groups <- as.factor(storm.data$EVTYPE.groups)
## examine the PROPDMGEXP/CROPDMGEXP to determine how this data is characterized
unique(storm.data$PROPDMGEXP)

```

```

## [1] K M B m + 0 5 6 ? 4 2 3 h 7 H - 1 8
## Levels: - ? + 0 1 2 3 4 5 6 7 8 B h H K m M

```

```

unique(storm.data$CROPDMGEXP)

```

```

## [1] M K m B ? 0 k 2
## Levels: ? 0 2 B k K m M

```

```

## convert combination of PROPDMG/CROPDGMG numeric and PROPDMGEXP/CROPDGMGEXP exponent
## to numerical values for further analysis:
storm.data$PROP.mult[storm.data$PROPDMGEXP %in% c("0", "+", "-", "")] <- 1
storm.data$PROP.mult[storm.data$PROPDMGEXP == "1"] <- 10
storm.data$PROP.mult[storm.data$PROPDMGEXP %in% c("[Hh]", "2")] <- 100
storm.data$PROP.mult[storm.data$PROPDMGEXP %in% c("[Kk]", "3")] <- 1000
storm.data$PROP.mult[storm.data$PROPDMGEXP == "4"] <- 10000
storm.data$PROP.mult[storm.data$PROPDMGEXP == "5"] <- 100000
storm.data$PROP.mult[storm.data$PROPDMGEXP %in% c("[Mm]", "6")] <- 1000000
storm.data$PROP.mult[storm.data$PROPDMGEXP == "7"] <- 10000000
storm.data$PROP.mult[storm.data$PROPDMGEXP == "8"] <- 100000000
storm.data$PROP.mult[storm.data$PROPDMGEXP == "[Bb]"] <- 1000000000
storm.data$PROP.mult[storm.data$PROPDMGEXP == "?"] <- "NA"
storm.data$CROP.mult[storm.data$CROPDGMGEXP %in% c("0", "")] <- 1
storm.data$CROP.mult[storm.data$CROPDGMGEXP %in% c("H", "h", "2")] <- 100
storm.data$CROP.mult[storm.data$CROPDGMGEXP %in% c("K", "k")] <- 1000
storm.data$CROP.mult[storm.data$CROPDGMGEXP %in% c("M", "m")] <- 1000000
storm.data$CROP.mult[storm.data$CROPDGMGEXP %in% c("B", "b")] <- 1000000000
storm.data$CROP.mult[storm.data$CROPDGMGEXP == "?"] <- "NA"

## convert these multipliers to numeric class
storm.data$PROP.mult <- as.numeric(storm.data$PROP.mult)
storm.data$CROP.mult <- as.numeric(storm.data$CROP.mult)

## take the product of the property/crop damage values and their multipliers to
## get one number for the total value of each
storm.data$prop.total <- storm.data$PROPDMG * storm.data$PROP.mult
storm.data$crop.total <- storm.data$CROPDMG * storm.data$CROP.mult

## add these two variables to get a total economic impact value
storm.data$econ.total <- storm.data$prop.total + storm.data$crop.total

## add the number of deaths and injuries to get a measure of impact on human health
storm.data$health.total <- storm.data$FATALITIES + storm.data$INJURIES

## create a new variable for the year of each event
storm.data$year <- year(storm.data$DATE)

## subset storm data to just variables of interest for further analysis
analysis.data <- subset(storm.data, select = c("year", "EVTYPE.groups",
                                             "health.total", "econ.total"))

## rename variables for simplicity
names(analysis.data) <- c("year", "event", "health.total", "econ.total")

## only consider the complete cases of data (i.e., no NAs)
analysis.data <- analysis.data[complete.cases(analysis.data), ]

```

Processing the data for this analysis involved a significant number of judgements as one might expect when dealing with a set spanning sixty one years and an untold number of contributors responsible for coalescing scattered, unstructured reports into summary content. Each challenge is documented above, but they can be summarized as:

Subsetting

Although the raw data contained a great deal of ancillary variables, for this study we are only concerned with the relationship between the storm events and their effects on human health and the economy. As such, the underlying goal behind each processing step is geared towards selection of only variables that will enable our analysis of that relationship.

Class Conversions

Ensuring the proper class for each variable is a simple, but notable step, throughout this processing path. Qualitative values are converted to factors, quantitative measures are treated as numerics, and dates are made into formal date classes as a general rule set.

Categorization

The most difficult portion of processing involved the decision-making in properly categorizing the various event types in this dataset. Although NOAA provides a set of forty eight possible codes for this variable, the actual data has many more due to a variety of reasons: typos, improper formatting, unique terms, or the wrong type of data such as the County in which the event occurred. Furthermore, the forty eight events from NOAA are not truly discrete and there are many overlaps in these codes.

To overcome these challenges, this analysis goes forward by converting the variety of provided event types into logical groupings of fifteen events based upon the author's knowledge of such conditions and reasonable judgement calls as defined in the code. Where the contributor's entry was so ambiguous such that it would be unreasonable to convert their input to a category, the event is considered missing and coded as NA.

Exponent Decisions

Similar to the challenges present with the provided event types, the corresponding exponent fields for the property and crop damage variables ran the gamut of inconsistent entries. The processing on this data sought to convert these entries into a numerical multiplier such that the product of this variable and the damage variables could be combined for both property and crops to represent the total economic impact of each event. Of course, the inconsistencies required judgement calls as to the intent of the contributor, but the reasoning is contained in the code and any cases where the value was so unclear as to make any further categorization untraceable to any logical reason were considered NA.

Removing NAs

Finally, after the processing steps to properly align the data, this study only considers complete observations for further analysis and generating conclusions.

Analysis

To address the question of which events have the greatest effect on human health and the economy, we'll examine three different analyses for conclusions:

- Summary statistics for each event type for each year
- The total impact of each event through the course of the data
- The average impact of each event, per event

Summary Statistics

To assess summary statistics, we'll first summarize the data by event and year.

```
library(reshape2)

## melt the data to set the course for summarizing
year.melt <- melt(analysis.data, id = c("year", "event"), na.rm = TRUE)

## recast data into the sum by event and year
year.cast <- dcast(year.melt, year + event ~ variable, sum)

year.cast <- year.cast[year.cast$event != "NA", ]
```

With this data, boxplots should help us determine which events have the greatest impact on an annual basis.

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

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

```
health.box <- ggplot(year.cast, aes(x = event, y = health.total)) +
  geom_boxplot() +
  ggtitle("Fatalaties and Injuries in Each Year") +
  theme(plot.title = element_text(lineheight = 0.8, face = "bold")) +
  labs(x = "Event", y = "Number of Fatalities and Injuries") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, lineheight = 0.7),
        axis.text.y = element_text(lineheight = 0.7))
econ.box <- ggplot(year.cast, aes(x = event, y = econ.total)) +
  geom_boxplot() +
  ggtitle("Total Damage in Each Year") +
  theme(plot.title = element_text(lineheight = 0.8, face="bold")) +
  labs(x = "Event", y = "Total Economic Impact [$USD]") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, lineheight = 0.7),
        axis.text.y = element_text(lineheight = 0.7))
grid.arrange(health.box, econ.box)
```

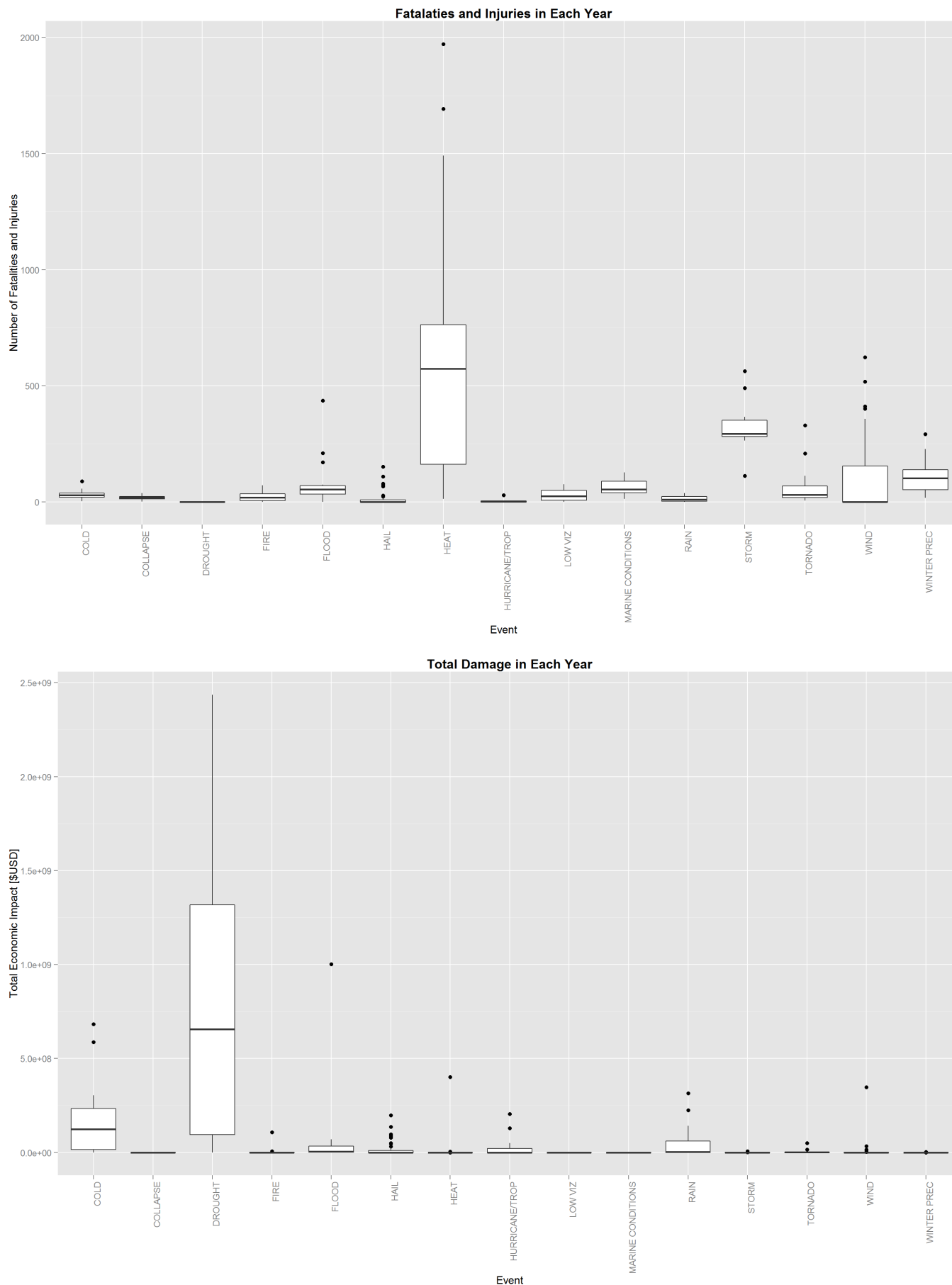



Fig. 1: Summary statistics for health and economic impact of each event type.

From this view, it would appear the greatest threat to human health and safety each year are heat events/excessively high temperatures. While major events like hurricanes may get the most notoriety for such effects, the more frequently occurring and less-feared heat wave has the greater effect.

With respect to economic impact, drought conditions cause the greatest amount of overall damage each year. Much of this value is likely due to crop loss, which while not as notable to the general public as property damage, is never-the-less a significant cost to society writ large.

Total Impacts

Next we examine, over the whole history of data, which events have left the greatest amount of damage to health and the economy in their wake.

Begin by summing the data by event over all years.

```
## recast data into the sum by event
total.cast <- dcast(year.melt, event ~ variable, sum)
total.cast <- total.cast[total.cast$event != "NA", ]
```

With this data, bar graphs should help us directly compare the effects of each event against each other.

```
health.bar <- ggplot(total.cast, aes(x = event, y = health.total)) +
  geom_bar(stat = "identity", alpha = 0.75) +
  ggtitle("Total Fatalities and Injuries [1950-2011]") +
  theme(plot.title = element_text(lineheight = 0.8, face = "bold")) +
  labs(x = "Event", y = "Number of Fatalities and Injuries") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, lineheight = 0.7),
        axis.text.y = element_text(lineheight = 0.7))
econ.bar <- ggplot(total.cast, aes(x = event, y = econ.total)) +
  geom_bar(stat = "identity", alpha = 0.75) +
  ggtitle("Total Economic Impact [1950-2011]") +
  theme(plot.title = element_text(lineheight = 0.8, face = "bold")) +
  labs(x = "Event", y = "Total Economic Impact [$USD]") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, lineheight = 0.7),
        axis.text.y = element_text(lineheight = 0.7))
grid.arrange(health.bar, econ.bar)
```

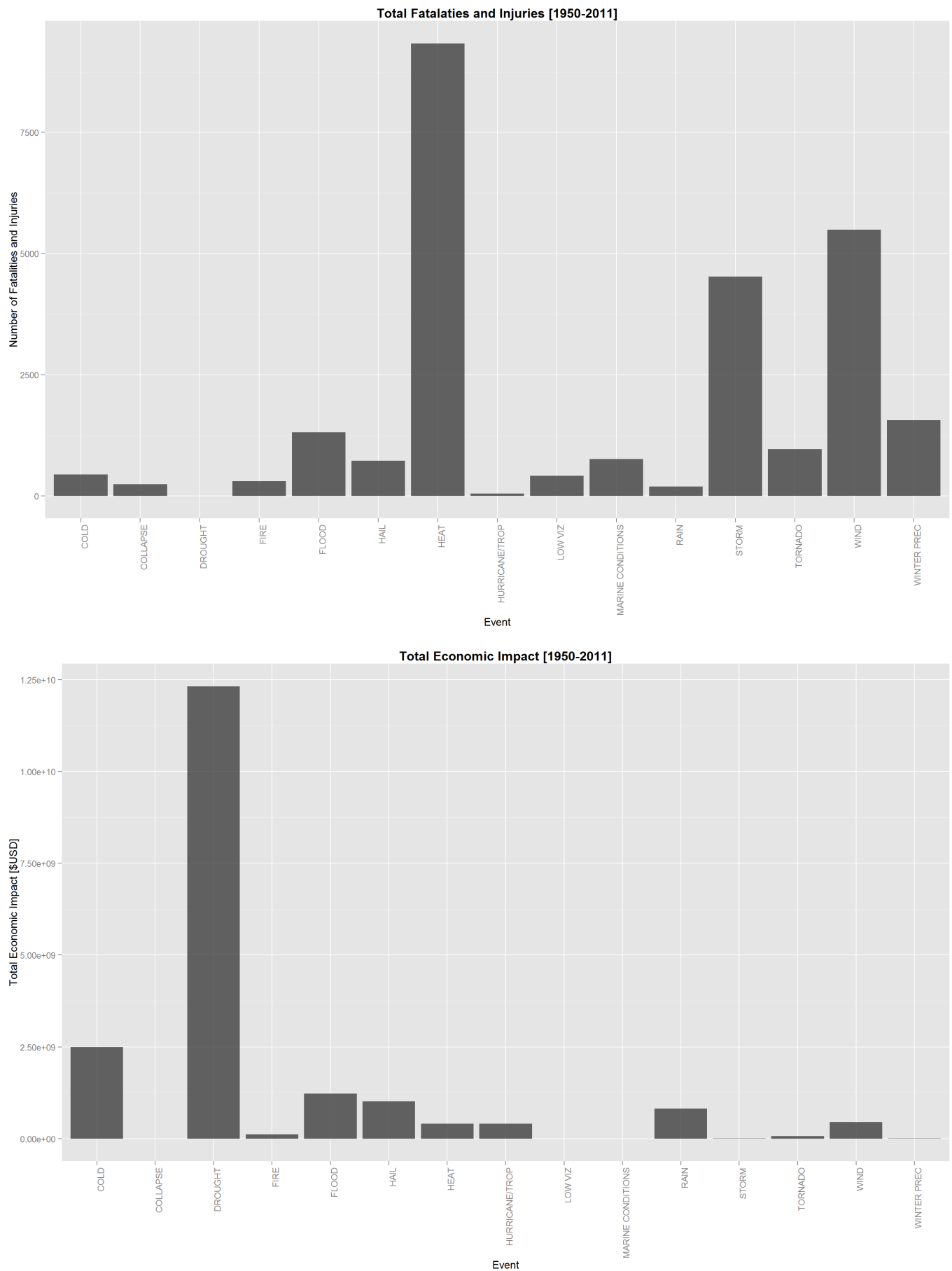


Fig. 2: Total health and economic impact of each event type.

Confirming the conclusions of our analysis of summary statistics by year, looking at the total effects of each event across all years of data allows us to note excessive heat and drought as the cause behind the greatest adverse effects on human health and the economy, respectively.

Average Impacts

Finally we examine this data for the events having the greatest effect on society when each event occurs. We'll assess the average casualty rate and economic impact based upon each instance of the event.

```
## recast data into the average by event
average.cast <- dcast(year.melt, event ~ variable, mean)

## count the total number of occurrences of each event and merge with the recast
## for later use in the plot
library(plyr)

event.count <- count(analysis.data$event)
names(event.count) <- c("event", "freq")
average.cast <- merge(average.cast, event.count, by = "event")
average.cast <- average.cast[average.cast$event != "NA", ]
```

To make sense of this data and provide a summary of the data to address the question of which event has the greatest impact to society as judged by effect on human health and the economy with consideration for how often the event has occurred in our data, we use a scatter plot to identify the greatest threatening event.

```
ggplot(average.cast, aes(x = health.total, y = econ.total)) +
  geom_point(aes(size = freq), colour = "blue", alpha = 0.75) +
  geom_text(aes(label = event), size = 3,
            position = position_jitter(width = 0.1, height = 0.1)) +
  scale_x_log10() +
  scale_y_log10() +
  ggtitle("Average Impact per Event [1950-2011]") +
  theme(plot.title = element_text(lineheight = 0.8, face = "bold")) +
  labs(x = "Log of Average Number of Fatalities and Injuries per Occurrence",
       y = "Log of Average Economic Impact per Occurrence [$USD]",
       size = "Count") +
  theme(axis.text.x = element_text(lineheight = 0.7),
        axis.text.y = element_text(lineheight = 0.7))
```

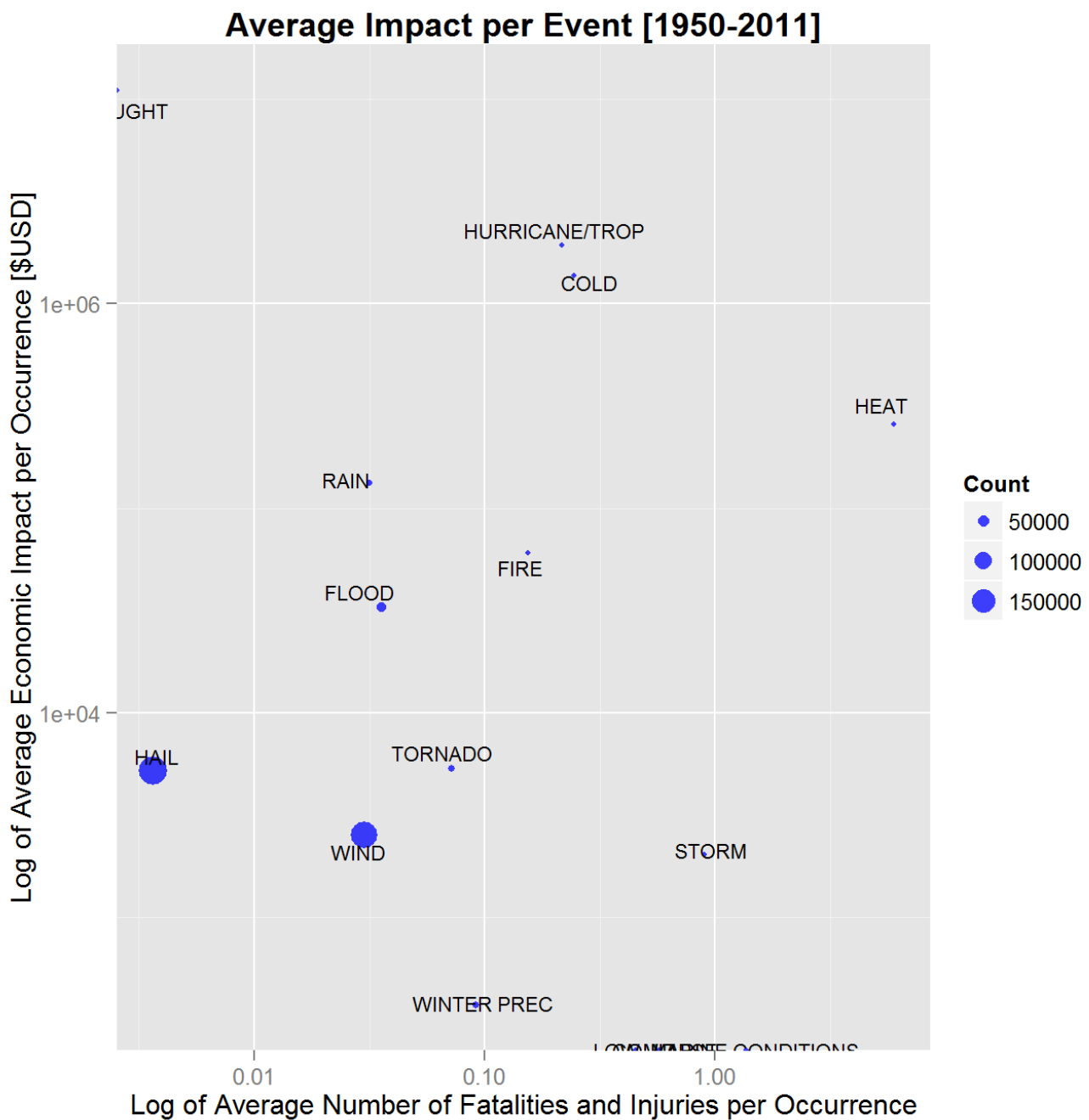


Fig. 3: Average health and economic impact of each event type, with dot size indicating frequency of occurrence.

With this view we have yet more evidence of the effects of various events on society as a whole. Hail and wind are the most frequently occurring, but have the lowest overall impact. While drought causes by far the most economic damage, it has minimal direct effect on human health. Heat causes the greatest number of casualties, but has less of an impact than the similar frequency of occurrence events of cold and hurricanes.

Results

As a conclusion of this analysis of NOAA data, we can address the question of the events having the greatest impact on society. Using the loss of life or injury as a measure, excessive heat has the greatest adverse effect on the United States in data collected from 1950 to 2011. Using that same data, the greatest threat to the economy is affiliated with drought conditions. Considering the data in aggregate, excessive heat is the most threatening event considering the combination of highest negative impact on human health and relatively high impact on the economy as compared to other events.

These conclusions are based upon the entire analysis presented here, including processing steps to categorize events. Errors or misjudgements in this process could have skewed these results, as well as errors in the original data itself. Independent evaluation of the processing steps would aid in validating the conclusions presented here.

For further study, this data could be used to for a variety of other interests including:

- Predictive studies on potential country-wide and local impacts from events (using geo data)
- Analysis on the year-over-year changes in frequency of occurrence of these events and a relation to the health and economic impact
- Similarly, the year-over-year analysis of health and economic impacts to assess the changing risk exposure of the United States to these events