Economic and Health Impact of Severe Weather in the United States

Introduction

This is an analysis focussing on the impacts of severe weather conditions in the United States. The project uses the U.S. National Oceanic and Atmospheric Administration's (NOAA) storm database to study the effect of severe weather on economic and health of the people in the US.

The analysis is performed by 2 categories:

1. Population health with data related to injurues and fatalaties.
2. Economic consequence with data related to property and crop damage.

The data is collected from the year 1950 and has records until 2011, although records in earlier years are fewer compared to more recent data.

> As you scroll down the analysis, there will be code blocks in R language which was used for this analysis.

Data Processing

The 1st step of this analysis is to read in the data. R language has many inbuilt libraries which will aid us in this analysis. So we import them in order to use the libraries.

```
library(plyr)
library(ggplot2)
library(gridExtra)
library(grid)
```

Reading in the data. (Assuming the data is in working directory)

```
data <- read.csv("repdata-data-StormData.csv.bz2")
```

The data is now imported into a variable called data which stores it in tabular format (rows and columns).

Since the analysis concerns only with economic and health impact lets take subset of data we require for the analysis into a new variable.

```
requiredData <- data[,c("EVTYPE", "FATALITIES", "INJURIES", "PROPDMG","PROPDMGEXP","C
ROPDMG","CROPDMGEXP")]
```

The data we require is ready. But the data is not suitable for analysis. The data for “PROPDMGEXP” and “CROPDMGEXP” are characters representing their exponential numbers. These need to be substituted to proceed with the analysis.

For this we write a small function which will do the above.

```
getExpValue <- function(ch){
  if (ch %in% c("h", "H")) return(2)
  else if (ch %in% c("k", "K")) return(3)
  else if (ch %in% c("m", "M")) return(6)
  else if (ch %in% c("b", "B")) return(9)
  else return(0)
}
```

Now we use this function and add 2 new columns to dataset. These column will contain the corresponding numeric values of exponents.

```
property <- sapply(requiredData$PROPDMGEXP, FUN=getExpValue)
requiredData$propertyDamage <- requiredData$PROPDMG * (10 ** property)
crop <- sapply(requiredData$CROPDMGEXP, FUN=getExpValue)
requiredData$cropDamage <- requiredData$CROPDMG * (10 ** crop)
```

Now the dataset is read to proceed with the analysis.

Data Analysis

Now lets begin the required analysis.

Population Health Consequence (PHC)

The fatalities and injuries data is available in the dataset. We sum the total of those 2 columns as per each event type.

```
PHCperEvent <- ddply(requiredData, .(EVTYPE), summarize,fatalities = sum(FATALITIES),
injuries = sum(INJURIES))
fatalData <- PHCperEvent[order(PHCperEvent$fatalities, decreasing = T), ]
injuryData <- PHCperEvent[order(PHCperEvent$injuries, decreasing = T), ]
```

Economic Consequence (EC)

The economic consequence data analysis is done by using the two new columns we generated in last section - propertyDamage and cropDamage

```
ECperEvent <- ddply(requiredData, .(EVTYPE), summarize,propertyDamage = sum(propertyD
amage), cropDamage = sum(cropDamage))
propertyDamageSumPerEvent <- ECperEvent[order(ECperEvent$propertyDamage, decreasing =
T), ]
cropDamageSumPerEvent <- ECperEvent[order(ECperEvent$cropDamage, decreasing = T), ]
```

Result

Now that the data has been processed and analyzed, lets look at the results.

Population health results

The top events which caused maximum number of injuries:

```
head(injuryData[, c("EVTYPE", "injuries")])
```

```
##             EVTYPE injuries
## 834        TORNADO    91346
## 856      TSTM WIND     6957
## 170          FLOOD     6789
## 130 EXCESSIVE HEAT     6525
## 464      LIGHTNING     5230
## 275           HEAT     2100
```

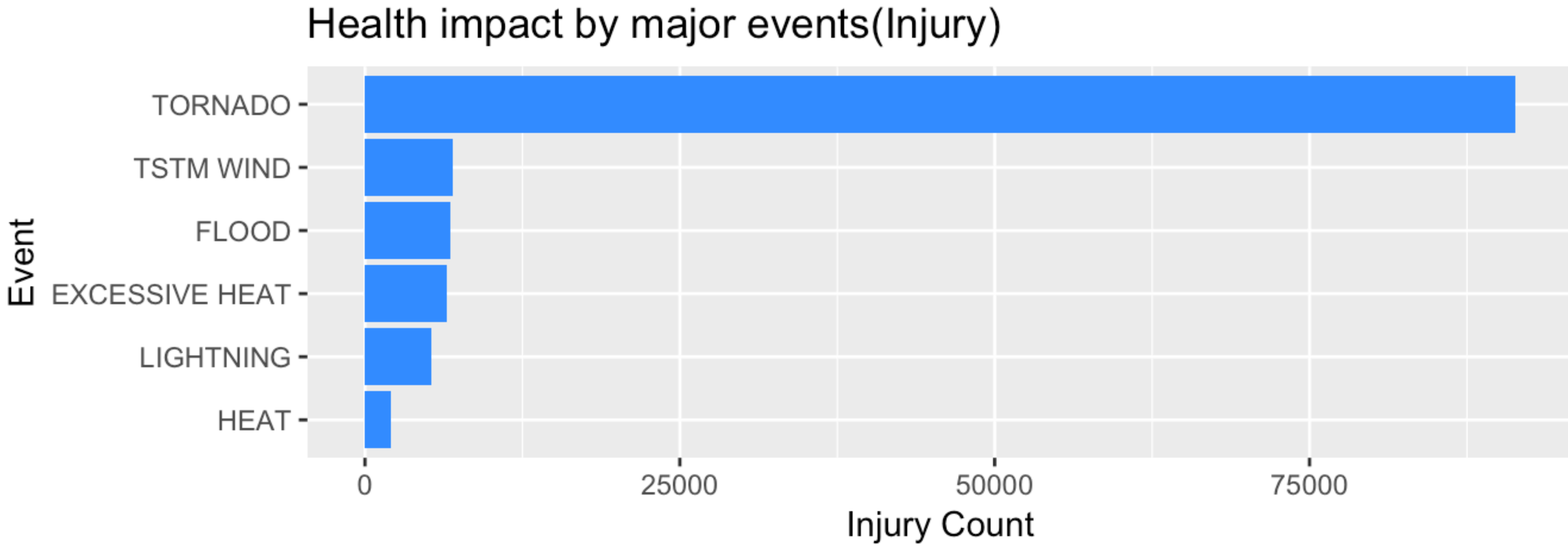
The top events which caused maximum number of fatalities:

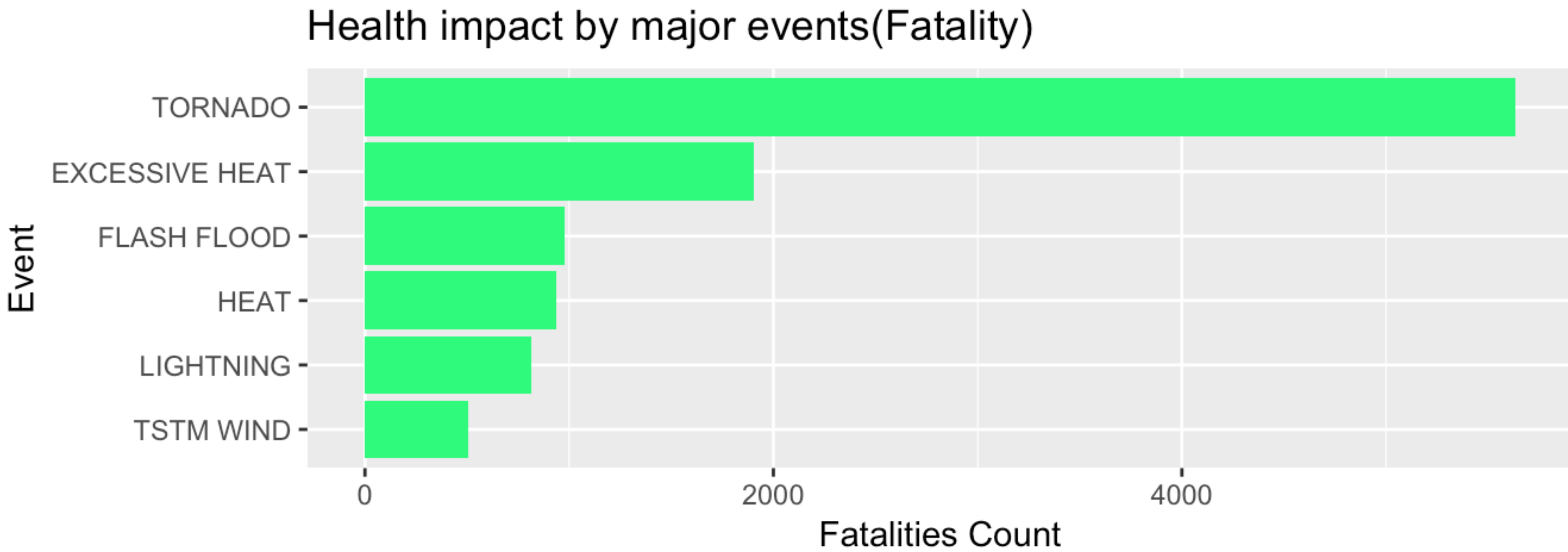
```
head(fatalData[, c("EVTYPE", "fatalities")])
```

```
##             EVTYPE fatalities
## 834        TORNADO       5633
## 130 EXCESSIVE HEAT       1903
## 153    FLASH FLOOD        978
## 275           HEAT        937
## 464      LIGHTNING        816
## 856      TSTM WIND        504
```

The data can be viewed better when plotted in graphs. The plot of injury and fatality count data is plotted against the event type.

```
plotInjury <- ggplot(data=head(injuryData), aes(x=reorder(EVTYPE, injuries), y=injuri
es)) + geom_bar(fill="#338DFF",stat="identity")  + coord_flip() +
ylab("Injury Count") + xlab("Event") +
ggtitle("Health impact by major events(Injury)")
plotFatality <- ggplot(data=head(fatalData), aes(x=reorder(EVTYPE, fatalities), y=fat
alities)) +
geom_bar(fill="#33FF7D",stat="identity") + coord_flip() +
ylab("Fatalities Count") + xlab("Event") +
ggtitle("Health impact by major events(Fatality)")
grid.arrange(plotInjury, plotFatality, nrow=2)
```





Financial Damage results

The top events which caused the maximum amount in property damage:

```
head(propertyDamageSumPerEvent[, c("EVTYPE", "propertyDamage")])
```

```
##                EVTYPE propertyDamage
## 170             FLOOD   144657709807
## 411 HURRICANE/TYPHOON    69305840000
## 834           TORNADO    56937160779
## 670       STORM SURGE    43323536000
## 153       FLASH FLOOD    16140812067
## 244              HAIL    15732267543
```

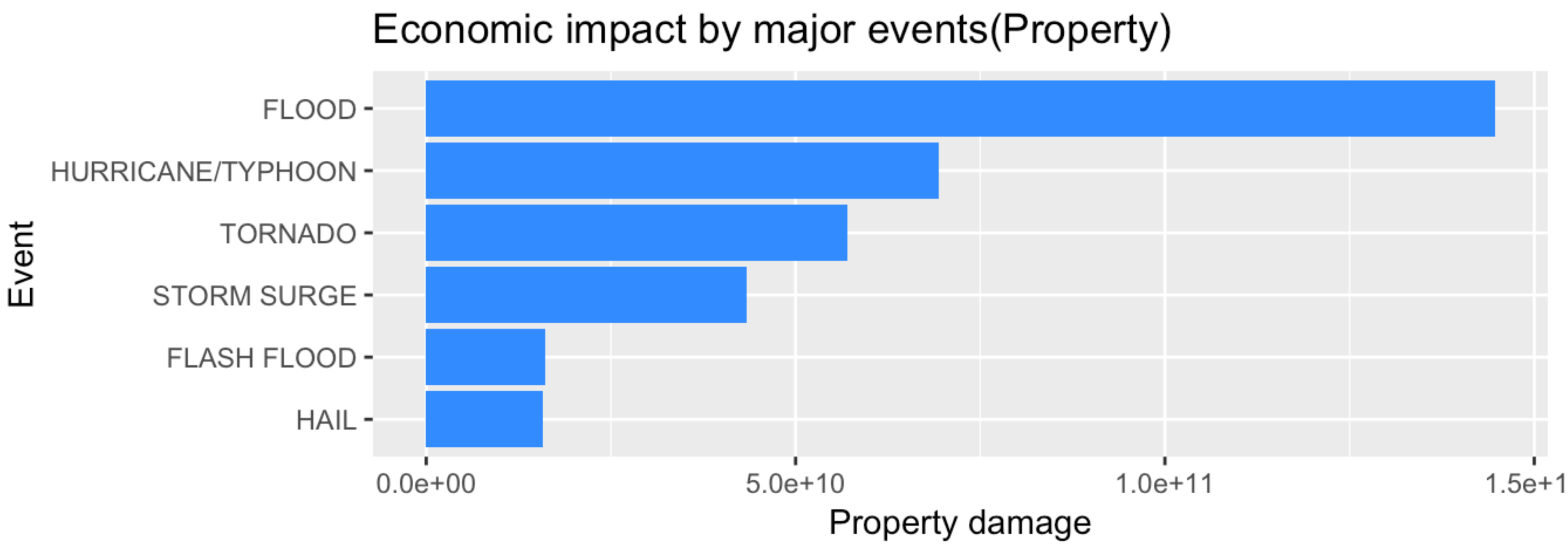
The top events which caused the maximum amount in crop damage:

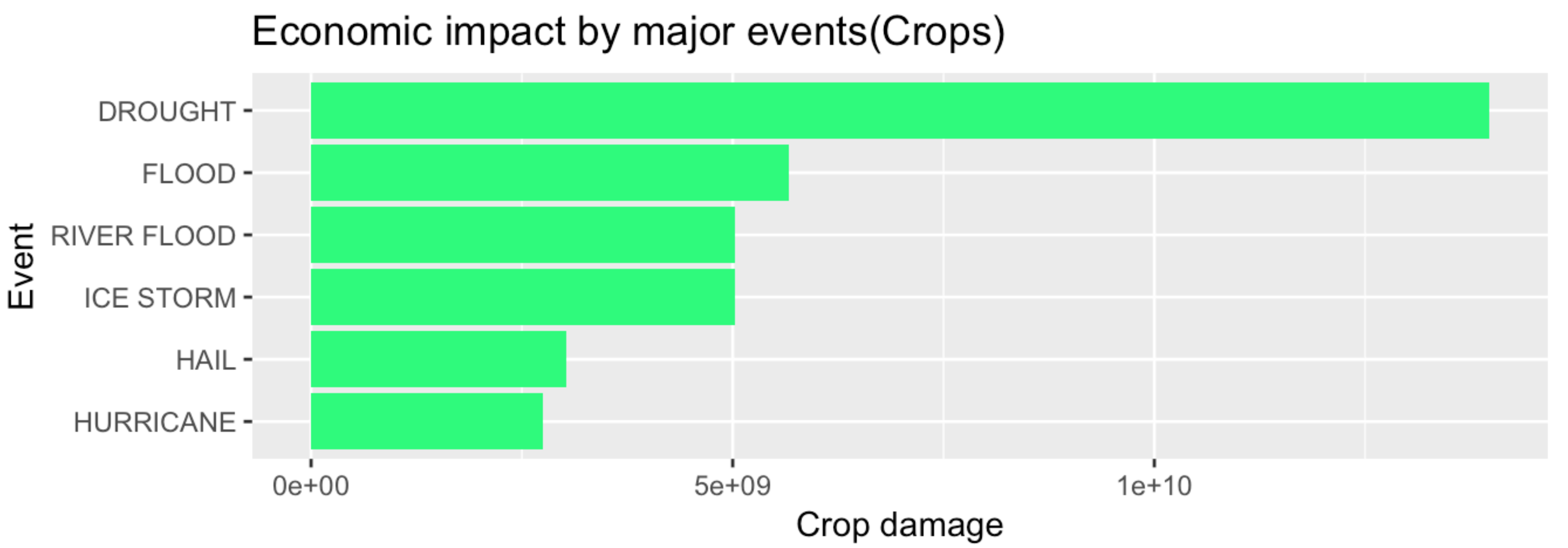
```
head(cropDamageSumPerEvent[, c("EVTYPE", "cropDamage")])
```

```
##          EVTYPE  cropDamage
## 95      DROUGHT 13972566000
## 170       FLOOD  5661968450
## 590 RIVER FLOOD  5029459000
## 427   ICE STORM  5022113500
## 244        HAIL  3025954473
## 402   HURRICANE  2741910000
```

The data can be viewed better when plotted in graphs.

```
plotPropDamage <- ggplot(data=head(propertyDamageSumPerEvent), aes(x=reorder(EVTYPE,
propertyDamage), y=propertyDamage)) +
geom_bar(fill="#338DFF", stat = "identity") + coord_flip() +
xlab("Event") + ylab("Property damage") +
ggtitle("Economic impact by major events(Property)")
plotCropDamage <- ggplot(data=head(cropDamageSumPerEvent), aes(x=reorder(EVTYPE, crop
Damage), y=cropDamage)) +
geom_bar(fill="#33FF7D", stat="identity") + coord_flip() +
xlab("Event") + ylab("Crop damage")+
ggtitle("Economic impact by major events(Crops)")
grid.arrange(plotPropDamage, plotCropDamage, nrow = 2)
```





As seen above in the graphs data can be better vizualized when they are in form of plots.