SEVERE WEATHER EVENTS ANALYSIS - CASUALTIES AND PROPERTY DAMAGE

by Bruno Berrehuel - 29th July 2016

Introduction

I apologize by advance for my poor English, as you will read I'm not a native English speaker so please be indulgent for "baby grammar problems".
Thank you for reading.

Synopsis

In order to evaluate which type of weather events are most harmful et have the most greatest economic consequences in the United States, the study uses a datafile from the National Climatic Data Center (NCDC) and from the National Weather Service, wich contains data from 1950 to november 2011.
The data processing is :

1. Download the file and store it in a stormData variable
2. Select the columns of interest for the questions with the DPLYR package : EVTYPE (type of weather event), FATALITIES (number of death), INJURIES (number of wunded), PROPDMG (amount of property damage) and PROPDMGEXP.
3. Remove events without any fatality, injury or damage.
4. Group event by more general type.

5. Process the amount of property damage to have all amount in billion dollars.
6. Finally plot the plots.

In further analysis we could check consequences by state and evolution since 1950, to verify if it's getting worse with global warming and growing population.

Results

Across the United States, which types of events are most harmful with respect to population health ?

The analysis shows that the 3 most harmful weather events are :

1. Tornado
2. Flood
3. Wind

Tornado is, with no doubt, the most harmful weather event, with **more than 5000 deaths and 90 000 injuries** since 1950.
Detailed results are available at the end of the document.

Across the United States, which types of events have the greatest economic consequences ?

The analysis shows that the 3 weather events with the greatest economic consequences are :

2. Flood
3. Hurricane
4. Tornado

Flood is, with no doubt, the weather event with the greatest economic consequence with more than **160 billion dollars of property damage** since 1950.
Detailed results are available at the end of the document.

Data Processing

Getting and storing data

Download the raw data directly from the web and decompress the file in order to read it with fread, much faster than read.csv :

```
if (!file.exists("stormData.csv")){
    download.file("https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2FStormData.csv.bz2",
                  destfile="stormData.csv.bz2")
    system("bunzip2 stormData.csv.bz2")
}
```

Store data in variable stormData with fread from data.table package and dplyr grammar from the dplyr package, then looking at data :

```
library(dplyr)
library(data.table)
```

```
stormData <- fread("stormData.csv")
```

```
##
Read 57.9% of 967216 rows
Read 85.8% of 967216 rows
Read 902297 rows and 37 (of 37) columns from 0.523 GB file in 00:00:04
```

```
str(stormData)
```

```
## Classes 'data.table' and 'data.frame':   902297 obs. of  37 variables:
##  $ STATE__   : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ BGN_DATE  : chr  "4/18/1950 0:00:00" "4/18/1950 0:00:00" "2/20/1951 0:00:00" "6/8/1951 0:00:00" .
##  $ BGN_TIME  : chr  "0130" "0145" "1600" "0900" ...
##  $ TIME_ZONE : chr  "CST" "CST" "CST" "CST" ...
##  $ COUNTY    : num  97 3 57 89 43 77 9 123 125 57 ...
##  $ COUNTYNAME: chr  "MOBILE" "BALDWIN" "FAYETTE" "MADISON" ...
##  $ STATE     : chr  "AL" "AL" "AL" "AL" ...
##  $ EVTYPE    : chr  "TORNADO" "TORNADO" "TORNADO" "TORNADO" ...
##  $ BGN_RANGE : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ BGN_AZI   : chr  "" "" "" "" ...
##  $ BGN_LOCATI: chr  "" "" "" "" ...
##  $ END_DATE  : chr  "" "" "" "" ...
##  $ END_TIME  : chr  "" "" "" "" ...
```

```
##  $ 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   : chr  "" "" "" "" ...
##  $ END_LOCATI: chr  "" "" "" "" ...
##  $ 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         : chr  "3" "2" "2" "2" ...
##  $ 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: chr  "K" "K" "K" "K" ...
##  $ CROPDMG   : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ CROPDMGEXP: chr  "" "" "" "" ...
##  $ WFO       : chr  "" "" "" "" ...
##  $ STATEOFFIC: chr  "" "" "" "" ...
##  $ ZONENAMES : chr  "" "" "" "" ...
##  $ 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   : chr  "" "" "" "" ...
##  $ REFNUM    : num  1 2 3 4 5 6 7 8 9 10 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

Cleaning and symplifying datas

Select the column of interest from the stormData variable (EVTYPE,FATALITIES,INJURIES,PROPDMG,PROPDMGEXP) and process the names for easier further work :

```
event <- stormData %>% select(EVTYPE,FATALITIES,INJURIES,PROPDMG,PROPDMGEXP)
event <- event %>% mutate(type = tolower(event$EVTYPE))
names(event) <- tolower(names(event))
event <- select(event, type, fatalities, injuries, propdmg, propdmgexp)
```

Filter the event with no consequences to reduce amount of data, filter the data with only magnitude (column propdmgexp) for further calculating, then looking of the number of the various event to see if it's usefull to group them :

```
event <- filter(event, grepl("[KMmB]",event$propdmgexp))
event <- event %>% filter(fatalities!=0 | injuries!=0 | propdmg!=0)
nb_event <- length(unique(event$type))
```

There is **370** types of event, we have to group them…

Now change the name of event to group them, according to the types in the Storm Data Documentation :

```
event$type[grep("avalanc",event$type)]                               <- "avalanche"
event$type[grep("blizzard",event$type)]                              <- "blizzard"
event$type[grep("glaze|cold|cool|freez|frost|hypo|low",event$type)] <- "cold"
event$type[grep("fog|smoke",event$type)]                             <- "dense fog"
event$type[grep("dust",event$type)]                                  <- "dust storm"
event$type[grep("floo|drown|rising|stream",event$type)]              <- "flood"
event$type[grep("funnel",event$type)]                                <- "funnel cloud"
```

```r
event$type[grep("hail",event$type)]                                    <- "hail"
event$type[grep("heat|hot|warm|hyper",event$type)]                     <- "heat"
event$type[grep("drought|dry|drie",event$type)]                        <- "drought"
event$type[grep("rain|showe|precip|wet",event$type)]                   <- "rain"
event$type[grep("snow|sleet",event$type)]                              <- "snow"
event$type[grep("surf|tide|surge|swell|wave",event$type)]              <- "tide/waves"
event$type[grep("wind|thunder|gust|wnd|tstm|turb",event$type)]         <- "wind"
event$type[grep("hurricane|typhoon",event$type)]                       <- "hurricane"
event$type[grep("ice|icy",event$type)]                                 <- "ice"
event$type[grep("torn|spout",event$type)]                              <- "tornado"
event$type[grep("tropical",event$type)]                                <- "tropical storm"
event$type[grep("volcanic",event$type)]                                <- "volcanic"
event$type[grep("fire",event$type)]                                    <- "fire"
event$type[grep("eros",event$type)]                                    <- "erosion"
event$type[grep("mud|lands|rock",event$type)]                          <- "landslide"
event$type[grep("lig",event$type)]                                     <- "lightning"
event$type[grep("sea|marine|water|curr",event$type)]                   <- "marine event"
event$type[grep("wint",event$type)]                                    <- "winter"
event$type[grep("urban",event$type)]                                   <- "urban"
event$type[grep("mix|other|seiche|county",event$type)]                 <- "other"
```

Now the event types are the following, which is more readable :

```r
sort(unique(event$type))
```

```
##  [1] "?"              "avalanche"      "blizzard"       "coastal storm"
##  [5] "cold"           "dam break"      "dense fog"      "downburst"
##  [9] "drought"        "dust storm"     "erosion"        "fire"
## [13] "flood"          "funnel cloud"   "hail"           "heat"
## [17] "hurricane"      "ice"            "landslide"      "lightning"
## [21] "marine event"   "microburst"     "other"          "rain"
## [25] "snow"           "tide/waves"     "tornado"        "tropical storm"
## [29] "tsunami"        "urban"          "volcanic"       "wind"
## [33] "winter"
```

The amount of property damages must converted in the same unit for comparison, so we chose to convert to billions dollars. The amount with K in propdmgexp is divided by 1000000, and the amount with M in propdmgexp is divided by 1000 to have the amount in billion dollars :

```r
kevent <- event %>% filter(propdmgexp=="K")
kevent <- kevent %>% mutate(damage=propdmg/1000000)
bevent <- event %>% filter(propdmgexp=="B")
bevent <- bevent %>% mutate(damage=propdmg)
mevent <- filter(event, grepl("[Mm]",event$propdmgexp))
mevent <- mevent %>% mutate(damage=propdmg/1000)
final_event <- rbind(mevent, kevent, bevent)
final_event <- final_event %>% select(type, fatalities, injuries, damage)
```

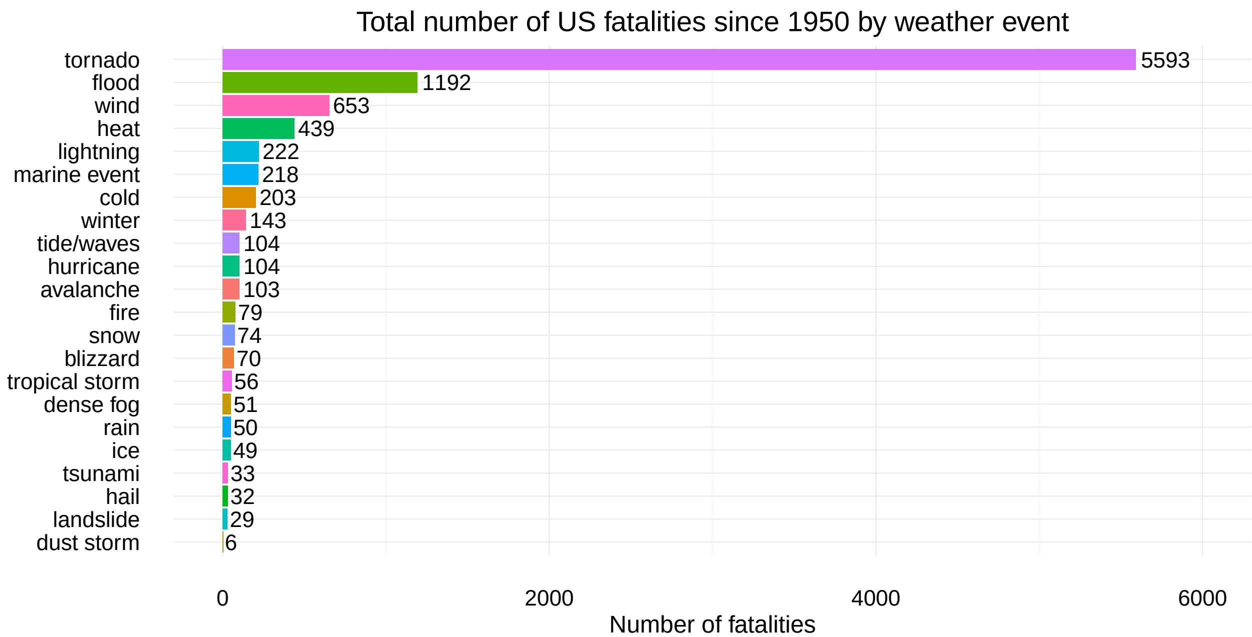
We can now group final_event by event type and calculate sum for each consequence :

```r
final_event <- group_by(final_event, type)
final_event <- summarize(final_event, sum(fatalities), sum(injuries), sum(damage))
names(final_event) <- c("type","fatalities","injuries","damages")
```

Ploting the results

fatalities ~ event type

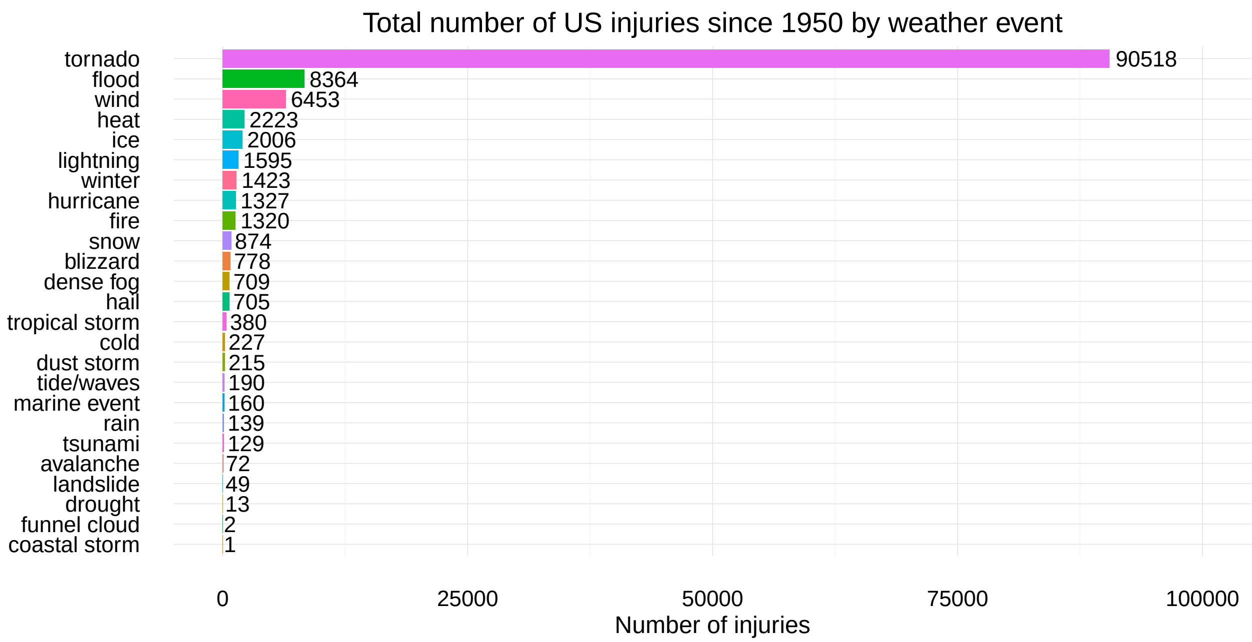
```
fatal <- final_event %>% filter(fatalities!="0") %>% arrange(desc(fatalities))
library(ggplot2)
ggplot(data=fatal, aes(reorder(type,fatalities),fatalities,fill=type)) +
       coord_flip() + theme_minimal() + geom_bar(stat="identity") +
       geom_text(aes(label=fatalities),hjust=-0.1) +
       theme(legend.position="none") + theme(axis.text=element_text(size=11)) +
       ylim(c(0,6000)) +
       labs(title="Total number of US fatalities since 1950 by weather event") +
       labs(y="Number of fatalities",x="")
```



Tornado is the most deadly event since 1950 in the US. Other spectaculary events like landslide or tsunami are the less deadly events.

injuries ~ event type

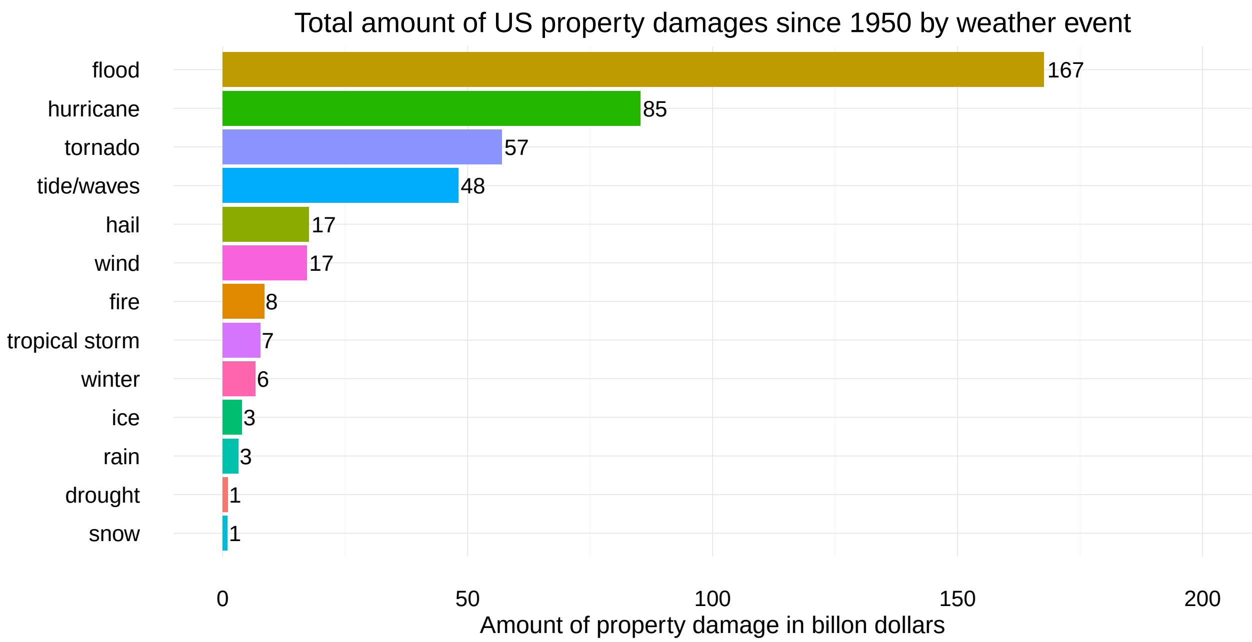
```
injury <- final_event %>% filter(injuries!="0") %>% arrange(desc(injuries))
library(ggplot2)
ggplot(data=injury, aes(reorder(type,injuries),injuries,fill=type)) +
       coord_flip() + theme_minimal() + geom_bar(stat="identity") +
       geom_text(aes(label=injuries),hjust=-0.1) +
       theme(legend.position="none") + theme(axis.text=element_text(size=11)) +
       ylim(c(0,100000)) +
       labs(title="Total number of US injuries since 1950 by weather event") +
       labs(y="Number of injuries",x="")
```



As for fatalities, tornado has the greatest number of injuried in the US since 1950.

property damages ~ event type

```
damage <- final_event %>% filter(damages>1) %>% arrange(desc(damages))
library(ggplot2)
ggplot(data=damage, aes(reorder(type,damages),damages,fill=type)) +
       coord_flip() + theme_minimal() + geom_bar(stat="identity") +
       geom_text(aes(label=floor(damages)),hjust=-0.1) +
       theme(legend.position="none") + theme(axis.text=element_text(size=11)) +
       ylim(c(0,200)) +
       labs(title="Total amount of US property damages since 1950 by weather event") +
       labs(y="Amount of property damage in billon dollars",x="")
```



Unlike the two other results, the weather event with the greatest consequence in property damage is flood.