Storm Data Analysis

Herizo RAJOELISON

1/24/2021

Contents

Analysys of Storm Data and the effects by type

This document constitutes a simple analysis of the StormData Weather events dataset and their impacts across the United States.

It seems that population health and the economy are impacted by approximately the same types of events but with varying degrees.

To the population health, the most harmful events are `TORNADO`, `EXCESSIVE HEAT` and `FLOOD`, making **51%** of the total.

On the other side, `FLOOD`, `HURRICANE (TYPHOON)` and `STORM SURGE/TIDE` have the most impacts to the economy, with **67%** of the total amount.

Data Processing

Data is loaded from the initial source and then uncompressed inside the data folder. Some checks are made to ensure that the destination folder exists.

```
library(dplyr)
library(R.utils)
library(lubridate)
library(scales)
library(knitr)


stormdata_url <- "https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2FStormData.csv.bz2"
stormdata_filename <- "repdata_data_StormData.csv"
stormdata_filepath <- file.path("./data/", stormdata_filename)
stormdata_archivename <- "repdata_data_StormData.csv.bz2"
stormdata_archivepath <- file.path("./data/", stormdata_archivename)
if (!file.exists(stormdata_archivepath)) {
  download.file(stormdata_url, stormdata_archivepath)
}
```

```
bunzip2(stormdata_archivepath, stormdata_filepath, remove=FALSE, skip=TRUE)
```

The data is read inside the *stormdata* variable.

```
stormdata <- read.csv(stormdata_filepath)

stormdata_nrows <- nrow(stormdata)
stormdata_ncols <- ncol(stormdata)
```

Before cleanup, the data contains **902297** columns and **37** rows.

```
column_labels <- colnames(stormdata)
print(column_labels)
```

```
##  [1] "STATE__"    "BGN_DATE"   "BGN_TIME"   "TIME_ZONE"  "COUNTY"
##  [6] "COUNTYNAME" "STATE"      "EVTYPE"     "BGN_RANGE"  "BGN_AZI"
## [11] "BGN_LOCATI" "END_DATE"   "END_TIME"   "COUNTY_END" "COUNTYENDN"
## [16] "END_RANGE"  "END_AZI"    "END_LOCATI" "LENGTH"     "WIDTH"
## [21] "F"          "MAG"        "FATALITIES" "INJURIES"   "PROPDMG"
## [26] "PROPDMGEXP" "CROPDMG"    "CROPDMGEXP" "WFO"        "STATEOFFIC"
## [31] "ZONENAMES"  "LATITUDE"   "LONGITUDE"  "LATITUDE_E" "LONGITUDE_"
## [36] "REMARKS"    "REFNUM"
```

Due to typos and too much variations on the input, there are more than **977** events against the **48** listed in the accompanying document. An attempt will be made to bring down those numbers by regrouping the events as much as possible into the official list.

Official list of events

The official list of events has been taken from the `repdata_data_StormData.pdf` and stored into a list.

```
# Official event types
official_event_types <- c("ASTRONOMICAL LOW TIDE", "AVALANCHE", "BLIZZARD",
                          "COASTAL FLOOD","COLD/WIND CHILL","DEBRIS FLOW",
                          "DENSE FOG", "DENSE SMOKE","DROUGHT",
                          "DUST DEVIL","DUST STORM", "EXCESSIVE HEAT",
                          "EXTREME COLD/WIND CHILL", "FLASH FLOOD", "FLOOD",
                          "FROST/FREEZE", "FUNNEL CLOUD","FREEZING FOG",
                          "HAIL", "HEAT", "HEAVY RAIN", "HEAVY SNOW",
                          "HIGH SURF", "HIGH WIND", "HURRICANE (TYPHOON)",
                          "ICE STORM", "LAKE-EFFECT SNOW", "LAKESHORE FLOOD",
                          "LIGHTNING", "MARINE HAIL", "MARINE HIGH WIND",
                          "MARINE STRONG WIND", "MARINE THUNDERSTORM WIND",
                          "RIP CURRENT", "SEICHE", "SLEET", "STORM SURGE/TIDE",
                          "STRONG WIND", "THUNDERSTORM WIND", "TORNADO",
                          "TROPICAL DEPRESSION", "TROPICAL STORM", "TSUNAMI",
                          "VOLCANIC ASH","WATERSPOUT", "WILDFIRE", "WINTER STORM",
                          "WINTER WEATHER")

print(official_event_types)
```

```
##  [1] "ASTRONOMICAL LOW TIDE"     "AVALANCHE"
##  [3] "BLIZZARD"                   "COASTAL FLOOD"
##  [5] "COLD/WIND CHILL"            "DEBRIS FLOW"
##  [7] "DENSE FOG"                  "DENSE SMOKE"
##  [9] "DROUGHT"                    "DUST DEVIL"
## [11] "DUST STORM"                 "EXCESSIVE HEAT"
## [13] "EXTREME COLD/WIND CHILL"    "FLASH FLOOD"
## [15] "FLOOD"                      "FROST/FREEZE"
## [17] "FUNNEL CLOUD"               "FREEZING FOG"
## [19] "HAIL"                       "HEAT"
## [21] "HEAVY RAIN"                 "HEAVY SNOW"
## [23] "HIGH SURF"                  "HIGH WIND"
## [25] "HURRICANE (TYPHOON)"        "ICE STORM"
## [27] "LAKE-EFFECT SNOW"           "LAKESHORE FLOOD"
## [29] "LIGHTNING"                  "MARINE HAIL"
## [31] "MARINE HIGH WIND"           "MARINE STRONG WIND"
## [33] "MARINE THUNDERSTORM WIND"   "RIP CURRENT"
## [35] "SEICHE"                     "SLEET"
## [37] "STORM SURGE/TIDE"           "STRONG WIND"
## [39] "THUNDERSTORM WIND"          "TORNADO"
## [41] "TROPICAL DEPRESSION"        "TROPICAL STORM"
## [43] "TSUNAMI"                    "VOLCANIC ASH"
## [45] "WATERSPOUT"                 "WILDFIRE"
## [47] "WINTER STORM"               "WINTER WEATHER"
```

Event types uniformization

For the grouping, a more manual approach has been chosen as it seems to allow finer control on the groups
and faster than using regular expressions. A list of similar event types based on the official events list is
created. Every element of the list contains a vector of the types thought to be similar enough. The first
element of each entry is the official name of the event.

A lookup table is built from this list allowing a replacement from the initial table's event types to the types
from the previous official list.

```r
replacement_list <- list(
  c("ASTRONOMICAL LOW TIDE"),
  c("AVALANCHE", "AVALANCE", "HEAVY SNOW/BLIZZARD/AVALANCHE"),
  c("BLIZZARD", "HIGH WIND/BLIZZARD", "HEAVY SNOW/BLIZZARD",
    "BLIZZARD AND EXTREME WIND CHIL", "BLIZZARD/HEAVY SNOW", "GROUND BLIZZARD",
    "BLIZZARD AND HEAVY SNOW", "BLIZZARD SUMMARY", "BLIZZARD WEATHER",
    "BLIZZARD/FREEZING RAIN", "BLIZZARD/HIGH WIND", "BLIZZARD/WINTER STORM",
    "HIGH WIND/ BLIZZARD", "HIGH WIND/BLIZZARD/FREEZING RA", "ICESTORM/BLIZZARD"
  ),
  c("COASTAL FLOOD", "COASTAL FLOODING", "COASTAL FLOODING/EROSION", "COASTAL/TIDAL FLOOD",
    "BEACH EROSION/COASTAL FLOOD", "COASTAL  FLOODING/EROSION", "COASTAL EROSION",
    "COASTALFLOOD", "HEAVY SURF COASTAL FLOODING", "HIGH WINDS/COASTAL FLOOD",
    "CSTL FLOODING/EROSION", "EROSION/CSTL FLOOD", "TIDAL FLOODING", "TIDAL FLOOD"
    ),
  c("COLD/WIND CHILL", "COLD", "HIGH WIND/LOW WIND CHILL", "HIGH WIND/WIND CHILL",
    "HIGH WIND/WIND CHILL/BLIZZARD", "WIND CHILL", "COLD WIND CHILL TEMPERATURES",
    "LOW WIND CHILL", "SNOW- HIGH WIND- WIND CHILL", "WIND CHILL/HIGH WIND"
    ),
  c("DEBRIS FLOW"),
```

```r
c("DENSE FOG", "FOG", "PATCHY DENSE FOG", "FOG AND COLD TEMPERATURES", "VOG"),
c("DENSE SMOKE", "SMOKE"),
c("DROUGHT", "UNSEASONABLY DRY", "DROUGHT/EXCESSIVE HEAT", "EXCESSIVE HEAT/DROUGHT",
  "HEAT DROUGHT", "HEAT WAVE DROUGHT", "HEAT/DROUGHT"
  ),
c("DUST DEVIL", "DUST DEVEL", "DUST DEVIL WATERSPOUT"),
c("DUST STORM", "BLOWING DUST", "SAHARAN DUST", "DUST STORM/HIGH WINDS",
  "DUSTSTORM", "HIGH WINDS DUST STORM"),
c("EXCESSIVE HEAT"),
c("EXTREME COLD/WIND CHILL", "EXTREME COLD", "RECORD COLD", "EXTREME WINDCHILL",
  "BITTER WIND CHILL TEMPERATURES", "BITTER WIND CHILL", "EXTREME WIND CHILL/BLOWING SNO",
  "EXTREME WIND CHILLS", "EXTREME WIND CHILL", "EXTREME WINDCHILL TEMPERATURES",
  "HIGH WINDS AND WIND CHILL", "UNSEASONABLY COLD", "PROLONG COLD", "UNUSUALLY COLD",
  "COLD TEMPERATURES", "COLD WEATHER", "EXTREME/RECORD COLD", "COLD WAVE",
  "COLD TEMPERATURE", "EXCESSIVE COLD", "COLD AND WET CONDITIONS", "COLD/WINDS",
  "EXTENDED COLD", "RECORD  COLD", "SEVERE COLD", "UNSEASONABLE COLD",
  "LOW TEMPERATURE", "RECORD LOW", "LOW TEMPERATURE RECORD",
  "BLOWING SNOW- EXTREME WIND CHI", "BLOWING SNOW/EXTREME WIND CHIL"
  ),
c("FLASH FLOOD", "FLASH FLOODING", "FLOOD/FLASH FLOOD", "FLASH FLOODS",
  "FLASH FLOODING/FLOOD", "FLASH FLOOD/FLOOD", "FLASH FLOOD FROM ICE JAMS",
  "FLOOD FLASH", "FLASH FLOOD - HEAVY RAIN", "FLASH FLOOD/ FLOOD", "FLOOD/FLASH",
  "FLOOD/FLASH FLOODING", "FLASH FLOOD LANDSLIDES", "FLASH FLOOD WINDS", "FLASH FLOOD/",
  "FLASH FLOOD/ STREET", "FLASH FLOOD/HEAVY RAIN", "FLASH FLOOD/LANDSLIDE",
  "FLASH FLOOODING", "FLOOD FLOOD/FLASH", "FLOOD/FLASH/FLOOD", "FLOOD/FLASHFLOOD",
  "LOCAL FLASH FLOOD"
  ),
c("FLOOD", "URBAN/SML STREAM FLD", "URBAN FLOOD", "URBAN FLOODING", "FLOODING",
  "STREET FLOOD", "RIVER FLOOD", "URBAN/SMALL STREAM FLOOD", "RIVER FLOODING",
  "LANDSLIDE/URBAN FLOOD", "URBAN FLOOD LANDSLIDE", "URBAN/SMALL STREAM",
  "URBAN AND SMALL STREAM FLOODIN", "SMALL STREAM/URBAN FLOOD",
  "URBAN/SMALL STREAM FLOODING", "URBAN AND SMALL STREAM",
  "URBAN AND SMALL STREAM FLOOD", "URBAN FLOODS", "URBAN/STREET FLOODING",
  "SMALL STREAM AND URBAN FLOOD", "URBAN AND SMALL", "URBAN SMALL STREAM FLOOD",
  "URBAN/SMALL", "URBAN/SMALL STREAM  FLOOD", "MUD SLIDES URBAN FLOODING",
  "SMALL STREAM AND URBAN FLOODIN", "SMALL STREAM URBAN FLOOD", "URBAN SMALL",
  "URBAN/SMALL FLOODING", "URBAN/SMALL STRM FLDG", "URBAN/SML STREAM FLDG",
  "SML STREAM FLD", "SMALL STREAM FLOOD", "FLOOD/RAIN/WINDS", "ICE JAM FLOODING",
  "SNOWMELT FLOODING", "MINOR FLOODING", "SMALL STREAM FLOODING", "FLOODS",
  "MAJOR FLOOD", "STREET FLOODING", "BEACH FLOOD", "RIVER AND STREAM FLOOD",
  "RURAL FLOOD", "BREAKUP FLOODING", "FLOOD WATCH/", "FLOOD/RAIN/WIND",
  "FLOOD/RIVER FLOOD", "HIGHWAY FLOODING", "ICE JAM FLOOD (MINOR",
  "LOCAL FLOOD", "MINOR FLOOD", "STREAM FLOODING"
  ),
c("FROST/FREEZE", "FREEZE", "FROST", "COLD AND FROST", "EARLY FROST",
  "RECORD COLD/FROST", "FIRST FROST", "FROST\\FREEZE", "DAMAGING FREEZE",
  "HARD FREEZE", "AGRICULTURAL FREEZE", "EARLY FREEZE", "LATE FREEZE"
  ),
c("FUNNEL CLOUD", "FUNNEL CLOUDS", "WATERSPOUT FUNNEL CLOUD", "FUNNEL CLOUD.",
  "WALL CLOUD/FUNNEL CLOUD", "ROTATING WALL CLOUD", "WALL CLOUD",
  "LARGE WALL CLOUD", "FUNNEL CLOUD/HAIL"
  ),
c("FREEZING FOG", "GLAZE", "GLAZE ICE", "ICE FOG"),
```

```
  c("HAIL", "SMALL HAIL", "HAIL 75", "HAIL 0.75", "HAIL 100", "HAIL 175",
    "NON SEVERE HAIL", "HAIL 1.00", "HAIL 1.75", "HAIL 275", "HAIL/WIND",
    "HAILSTORM", "HAIL 150", "HAIL 80", "HAIL DAMAGE", "HAIL/WINDS", "DEEP HAIL",
    "GUSTY WIND/HAIL", "HAIL 0.88", "HAIL 075", "HAIL 088", "HAIL 1.75)",
    "HAIL 125", "HAIL 200", "HAIL 225", "HAIL 450", "HAIL 88", "HAIL ALOFT",
    "HAIL FLOODING", "HAIL STORM", "HAIL(0.75)", "HAIL/ICY ROADS", "HAILSTORMS",
    "LATE SEASON HAIL", "WIND/HAIL"
    ),
  c("HEAT", "RECORD WARMTH", "UNSEASONABLY WARM", "RECORD HEAT", "HEAT WAVE",
    "EXTREME HEAT", "RECORD TEMPERATURE", "UNSEASONABLY WARM AND DRY",
    "UNSEASONABLY HOT", "UNUSUAL WARMTH", "RECORD/EXCESSIVE HEAT", "HEAT WAVES",
    "HEATBURST", "RECORD HEAT WAVE", "ABNORMAL WARMTH", "PROLONG WARMTH",
    "UNUSUALLY WARM", "UNSEASONABLY WARM YEAR", "UNSEASONABLY WARM/WET",
    "UNUSUAL/RECORD WARMTH", "RECORD WARM", "RECORD WARM TEMPS.",
    "UNSEASONABLY WARM & WET", "VERY WARM", "WARM DRY CONDITIONS", "WARM WEATHER",
    "HOT AND DRY","HOT SPELL", "DRY HOT WEATHER", "HOT PATTERN", "HOT WEATHER",
    "HOT/DRY PATTERN", "RECORD HIGH", "RECORD HIGH TEMPERATURE", "RECORD HIGH TEMPERATURES",
    "HIGH TEMPERATURE RECORD"
    ),
  c("HEAVY RAIN", "HEAVY RAINS", "HEAVY RAINS/FLOODING", "HEAVY RAIN AND WIND",
    "HEAVY RAIN/WIND", "HEAVY RAINFALL", "FLOOD & HEAVY RAIN", "HEAVY RAIN/FLOODING",
    "HEAVY RAIN/SEVERE WEATHER", "FLOODING/HEAVY RAIN",
    "HEAVY RAIN AND FLOOD", "HEAVY RAIN EFFECTS", "HEAVY RAIN; URBAN FLOOD WINDS;",
    "HEAVY RAIN/HIGH SURF", "HEAVY RAIN/LIGHTNING", "HEAVY RAIN/MUDSLIDES/FLOOD",
    "HEAVY RAIN/SMALL STREAM URBAN", "HEAVY RAIN/SNOW", "HEAVY RAIN/URBAN FLOOD",
    "HIGH WINDS HEAVY RAINS", "HIGH WINDS/HEAVY RAIN", "LIGHTNING AND HEAVY RAIN",
    "LIGHTNING/HEAVY RAIN", "LOCALLY HEAVY RAIN", "RAIN (HEAVY)", "RAIN",
    "RECORD RAINFALL", "MONTHLY RAINFALL", "EXCESSIVE RAIN", "RAIN/SNOW",
    "EXCESSIVE RAINFALL", "PROLONGED RAIN", "HVY RAIN", "RECORD LOW RAINFALL",
    "UNSEASONAL RAIN", "EARLY RAIN", "GUSTY WIND/HVY RAIN", "GUSTY WIND/RAIN",
    "RAIN AND WIND", "RAIN DAMAGE", "RAIN/WIND", "RAINSTORM", "RECORD/EXCESSIVE RAINFALL",
    "SNOW/RAIN", "TORRENTIAL RAIN", "TORRENTIAL RAINFALL", "SUMMARY OF JULY 11"
    ),
  c("HEAVY SNOW", "SNOW", "LIGHT SNOW", "MODERATE SNOWFALL", "HEAVY SNOW/FREEZING RAIN",
    "HEAVY SNOW   FREEZING RAIN", "HEAVY SNOW SQUALLS", "SNOW SQUALLS",
    "SNOW SQUALL", "HEAVY SNOW-SQUALLS", "HEAVY SNOW/SQUALLS", "HEAVY SNOW SQUALLS",
    "HEAVY SNOW-SQUALLS", "HEAVY SNOW/ICE", "HEAVY SNOW AND ICE",
    "HEAVY SNOW/SQUALLS", "SNOW AND HEAVY SNOW", "HEAVY SNOW & ICE",
    "HEAVY SNOW AND", "HEAVY SNOW AND STRONG WINDS", "HEAVY SNOW ANDBLOWING SNOW",
    "HEAVY SNOW SHOWER", "HEAVY SNOW/BLOWING SNOW", "HEAVY SNOW/HIGH",
    "HEAVY SNOW/SLEET", "HEAVY SNOW/WIND", "HEAVY SNOWPACK", "SNOW/HEAVY SNOW",
    "EXCESSIVE SNOW", "RECORD SNOW", "RECORD SNOWFALL", "RECORD WINTER SNOW",
    "NEAR RECORD SNOW", "RECORD MAY SNOW", "RECORD SNOW/COLD", "SNOWFALL RECORD",
    "HEAVY WET SNOW"
    ),
  c("HIGH SURF", "HEAVY SURF/HIGH SURF", "ASTRONOMICAL HIGH TIDE", "HEAVY SURF",
    "HIGH SURF ADVISORY", "ROUGH SURF", "HAZARDOUS SURF", "HEAVY SURF AND WIND",
    "HIGH SURF ADVISORIES"
    ),
  c("HIGH WIND", "HIGH WINDS", "WIND", "GUSTY WINDS", "GUSTY WIND", "GUSTY LAKE WIND",
    "HIGH WINDS/COLD", "HIGH WIND/HEAVY SNOW", "HIGH WINDS/SNOW", "HEAVY SNOW AND HIGH WINDS",
    "HIGH WIND (G40)", "HIGH WIND AND HIGH TIDES", "HIGH WIND DAMAGE",
    "HIGH WINDS 63", "HIGH WINDS 66", "HIGH WINDS 80", "SNOW/HIGH WINDS",
```

```r
    "HEAVY SNOW/HIGH WIND", "HEAVY SNOW/HIGH WINDS", "HEAVY SNOW/HIGH WINDS & FLOOD",
    "HEAVY SNOW/HIGH WINDS/FREEZING", "HIGH WIND 48", "HIGH WIND 63",
    "HIGH WIND 70", "HIGH WIND AND HEAVY SNOW", "HIGH WIND AND SEAS",
    "HIGH WIND/SEAS", "HIGH WINDS 55", "HIGH WINDS 57", "HIGH WINDS 58",
    "HIGH WINDS 67", "HIGH WINDS 73", "HIGH WINDS 76", "HIGH WINDS 82",
    "HIGH WINDS/", "HIGH WINDS/FLOODING", "HURRICANE OPAL/HIGH WINDS",
    "RECORD COLD AND HIGH WIND", "WINDS", "WIND DAMAGE"
    ),
  c("HURRICANE (TYPHOON)", "HURRICANE", "HURRICANE/TYPHOON", "TYPHOON",
    "HURRICANE OPAL", "HURRICANE ERIN", "HURRICANE-GENERATED SWELLS",
    "HURRICANE EDOUARD", "HURRICANE FELIX", "HURRICANE EMILY", "HURRICANE GORDON"
    ),
  c("ICE STORM", "ICE STORM/FLASH FLOOD", "SNOW/ICE STORM", "HEAVY SNOW AND ICE STORM",
    "HEAVY SNOW/ICE STORM", "GLAZE/ICE STORM", "ICE STORM AND SNOW", "SLEET/ICE STORM",
    "SNOW AND ICE STORM"),
  c("LAKE-EFFECT SNOW", "LAKE EFFECT SNOW", "HEAVY LAKE SNOW"),
  c("LAKESHORE FLOOD", "LAKE FLOOD"),
  c("LIGHTNING", "LIGHTING", "LIGHTNING  WAUSEON", "LIGHTNING AND WINDS",
    "LIGHTNING DAMAGE", "LIGHTNING FIRE", "LIGHTNING INJURY", "LIGHTNING."
    ),
  c("MARINE HAIL"),
  c("MARINE HIGH WIND"),
  c("MARINE STRONG WIND"),
  c("MARINE THUNDERSTORM WIND", "MARINE TSTM WIND"),
  c("RIP CURRENT", "RIP CURRENTS", "RIP CURRENTS/HEAVY SURF",
    "RIP CURRENTS HEAVY SURF"
    ),
  c("SEICHE"),
  c("SLEET", "SLEET STORM", "SNOW/SLEET", "FREEZING RAIN/SLEET",
    "FREEZING RAIN AND SLEET", "SNOW/SLEET/FREEZING RAIN", "SNOW AND SLEET",
    "LIGHT SNOW AND SLEET", "SLEET/FREEZING RAIN", "SLEET/SNOW",
    "FREEZING RAIN SLEET AND", "FREEZING RAIN SLEET AND LIGHT",
    "SLEET & FREEZING RAIN", "SLEET/RAIN/SNOW", "SNOW SLEET", "SNOW/RAIN/SLEET",
    "SNOW/SLEET/RAIN"
    ),
  c("STORM SURGE/TIDE", "STORM SURGE", "COASTAL STORM", "COASTAL SURGE", "COASTALSTORM"),
  c("STRONG WIND", "STRONG WINDS", "STRONG WIND GUST", "FLOOD/STRONG WIND",
    "ICE/STRONG WINDS"
    ),
  c("THUNDERSTORM WIND", "TSTM WIND/HAIL", "TSTM WIND", "THUNDERSTORM WINDS",
    "THUNDERSTORM WINDSS", "THUNDERTORM WINDS","THUNDERSTORM WINDS LIGHTNING",
    "THUNDERSTORMS WINDS", "THUNDERSTORMS WIND", "THUNDERSTORM WIND/LIGHTNING",
    "THUNDERTSORM WIND", "THUNDERSTORM WINDS/HAIL", "THUNDERSTORM", "THUNDERSTORM WINDS HAIL",
    "TSTM WIND (G45)", "SEVERE THUNDERSTORMS", "DRY MICROBURST", "MICROBURST",
    "TSTM WIND (G40)", "TSTM WINDS", "TSTM WIND 52", "TSTM HEAVY RAIN", "TSTM WIND 55",
    "TSTM WIND 51", "TSTM", "TSTM WIND  (G45)", "TSTM WIND (41)", "TSTM WIND (G35)",
    "TSTM WIND 40", "TSTM WIND 45", "TSTM WIND 50", "TSTM WIND 65)", "TSTM WIND AND LIGHTNING",
    "TSTM WIND DAMAGE", "TSTM WIND G45", "TSTM WIND G58", "TSTM WND", "TSTMW",
    "THUNDERSTORM  WINDS", "GUSTY THUNDERSTORM WINDS", "SEVERE THUNDERSTORM WINDS",
    "THUNDERSTORM WIND 60 MPH", "THUNDERSTORM WIND G50", "THUNDERSTORM WIND/ TREES",
    "THUNDERSTORMS", "GUSTY THUNDERSTORM WIND", "THUNDERSTORM WINDS.",
    "THUNDERSTORMW WINDS", "THUNDERSTORM DAMAGE", "THUNDERSTORM WIND 50",
    "THUNDERSTORM WIND G52", "THUNDERSTORM WIND G60", "THUNDERSTORM WINDS AND",
```

```
      "THUNDERSTORM WINDS FUNNEL CLOU", "THUNDERSTORM WINDS G", "THUNDERSTORM WINDS/ FLOOD",
      "THUNDERSTROM WINDS", "FLASH FLOODING/THUNDERSTORM WI", "LIGHTNING AND THUNDERSTORM WIN",
      "LIGHTNING THUNDERSTORM WINDS", "LIGHTNING THUNDERSTORM WINDSS",
      "THUNDERSNOW SHOWER", "THUNDERSTORM DAMAGE TO", "THUNDERSTORM HAIL", "THUNDERSTORM W INDS",
      "THUNDERSTORM WIND (G40)", "THUNDERSTORM WIND 52", "THUNDERSTORM WIND 56",
      "THUNDERSTORM WIND 59", "THUNDERSTORM WIND 59 MPH", "THUNDERSTORM WIND 59 MPH.",
      "THUNDERSTORM WIND 65 MPH", "THUNDERSTORM WIND 65MPH", "THUNDERSTORM WIND 69",
      "THUNDERSTORM WIND 98 MPH", "THUNDERSTORM WIND G51", "THUNDERSTORM WIND G55",
      "THUNDERSTORM WIND G61", "THUNDERSTORM WIND TREES", "THUNDERSTORM WIND.",
      "THUNDERSTORM WIND/ TREE", "THUNDERSTORM WIND/AWNING", "THUNDERSTORM WIND/HAIL",
      "THUNDERSTORM WINDS      LE CEN", "THUNDERSTORM WINDS 13", "THUNDERSTORM WINDS 2",
      "THUNDERSTORM WINDS 50", "THUNDERSTORM WINDS 52", "THUNDERSTORM WINDS 53",
      "THUNDERSTORM WINDS 60", "THUNDERSTORM WINDS 61", "THUNDERSTORM WINDS 62",
      "THUNDERSTORM WINDS 63 MPH", "THUNDERSTORM WINDS G60", "THUNDERSTORM WINDS HEAVY RAIN",
      "THUNDERSTORM WINDS SMALL STREA", "THUNDERSTORM WINDS URBAN FLOOD",
      "THUNDERSTORM WINDS/ HAIL", "THUNDERSTORM WINDS/FLASH FLOOD",
      "THUNDERSTORM WINDS/FLOODING", "THUNDERSTORM WINDS/FUNNEL CLOU",
      "THUNDERSTORM WINDS/HEAVY RAIN", "THUNDERSTORM WINDS53", "THUNDERSTORM WINDSHAIL",
      "THUNDERSTORM WINS", "THUNDERSTORMW", "THUNDERSTORMW 50", "THUNDERSTORMWINDS",
      "THUNDERSTROM WIND", "THUDERSTORM WINDS", "THUNDEERSTORM WINDS", "THUNDESTORM WINDS",
      "THUNDERESTORM WINDS", "THUNERSTORM WINDS", "DOWNBURST", "DOWNBURST WINDS",
      "GUSTNADO", "GUSTNADO AND", "WET MICROBURST", "DRY MICROBURST WINDS",
      "MICROBURST WINDS", "DRY MICROBURST 58", "DRY MICROBURST 50", "DRY MICROBURST 53",
      "DRY MICROBURST 61", "DRY MICROBURST 84", "DRY MIRCOBURST WINDS", "WET MICOBURST",
      "SUMMARY JAN 17", "SUMMARY OF MARCH 14", "SUMMARY OF MARCH 23", "SUMMARY OF MARCH 24",
      "SUMMARY OF APRIL 3RD", "SUMMARY OF APRIL 12", "SUMMARY OF APRIL 13",
      "SUMMARY OF APRIL 21", "SUMMARY AUGUST 11", "SUMMARY OF APRIL 27",
      "SUMMARY OF MAY 9-10", "SUMMARY OF MAY 10", "SUMMARY OF MAY 13",
      "SUMMARY OF MAY 14", "SUMMARY OF MAY 22 AM", "SUMMARY OF MAY 22 PM",
      "SUMMARY OF MAY 26 AM", "SUMMARY OF MAY 26 PM", "SUMMARY OF MAY 31 AM",
      "SUMMARY OF MAY 31 PM", "SUMMARY OF JUNE 3", "SUMMARY OF JUNE 4",
      "SUMMARY JUNE 5-6", "SUMMARY JUNE 6", "SUMMARY OF JUNE 11", "SUMMARY OF JUNE 12",
      "SUMMARY OF JUNE 13", "SUMMARY OF JUNE 15", "SUMMARY OF JUNE 16", "SUMMARY JUNE 18-19",
      "SUMMARY OF JUNE 23", "SUMMARY OF JUNE 24", "SUMMARY OF JUNE 30", "SUMMARY OF JULY 2",
      "SUMMARY OF JULY 3", "SUMMARY OF JULY 22", "SUMMARY JULY 23-24", "SUMMARY OF JULY 26",
      "SUMMARY OF JULY 29", "SUMMARY OF AUGUST 1", "SUMMARY AUGUST 2-3", "SUMMARY AUGUST 7",
      "SUMMARY AUGUST 9", "SUMMARY AUGUST 10", "SUMMARY AUGUST 17", "SUMMARY AUGUST 21",
      "SUMMARY AUGUST 28", "SUMMARY SEPTEMBER 4", "SUMMARY SEPTEMBER 20", "SUMMARY SEPTEMBER 23",
      "SUMMARY SEPT. 25-26", "SUMMARY: OCT. 20-21", "SUMMARY: OCTOBER 31", "SUMMARY: NOV. 6-7",
      "SUMMARY: NOV. 16", "SUMMARY OF MARCH 23", "SUMMARY OF APRIL 12", "SUMMARY OF APRIL 21",
      "SUMMARY OF MAY 22", "SUMMARY OF JUNE 6", "SUMMARY AUGUST 4", "SUMMARY OF JUNE 3",
      "SUMMARY OF JUNE 10", "SUMMARY OF JUNE 13", "SUMMARY OF JUNE 18", "SUMMARY AUGUST 10",
      "SUMMARY AUGUST 11", "SUMMARY SEPTEMBER 3", "SUMMARY: SEPT. 18", "SUMMARY SEPTEMBER 23",
      "SUMMARY: NOV. 16", "SUMMARY OF MARCH 24-25", "SUMMARY OF MARCH 27", "SUMMARY OF MARCH 29",
      "TUNDERSTORM WIND"
   ),
 c("TORNADO", "FUNNEL", "WATERSPOUTS", "SEVERE THUNDERSTORM", "TORNADO F0",
   "TORNADO F1", "TORNADO F2", "TORNADO F3", "TORNADOES", "COLD AIR TORNADO",
   "TORNADO DEBRIS", "TORNADOES, TSTM WIND, HAIL", "TORNADOS", "TORNDAO",
   "COLD AIR FUNNEL", "COLD AIR FUNNELS", "FUNNELS"
 ),
 c("TROPICAL DEPRESSION"),
 c("TROPICAL STORM", "TROPICAL STORM JERRY", "TROPICAL STORM DEAN",
```

```
      "TROPICAL STORM ALBERTO", "TROPICAL STORM GORDON"
      ),
    c("TSUNAMI", "LANDSLIDE", "LANDSLIDES"),
    c("VOLCANIC ASH", "VOLCANIC ASHFALL", "VOLCANIC ERUPTION", "VOLCANIC ASH PLUME"),
    c("WATERSPOUT", "WATERSPOUT-", "WATERSPOUT/", "WATER SPOUT", "WAYTERSPOUT",
      "WATERSPOUT/TORNADO", "WATERSPOUT-TORNADO", "WATERSPOUT/ TORNADO",
      "TORNADO/WATERSPOUT", "WATERSPOUT TORNADO"
      ),
    c("WILDFIRE", "WILD/FOREST FIRE", "WILDFIRES", "WILD FIRES", "BRUSH FIRE",
      "BRUSH FIRES", "FOREST FIRES", "GRASS FIRES", "WILD/FOREST FIRES"),
    c("WINTER STORM", "WINTER STORMS", "HEAVY SNOW/WINTER STORM",
      "WINTER STORM HIGH WINDS", "WINTER STORM/HIGH WIND", "WINTER STORM/HIGH WINDS"
      ),
    c("WINTER WEATHER", "WINTER WEATHER/MIX", "FREEZING RAIN", "LIGHT FREEZING RAIN",
      "SNOW FREEZING RAIN", "SNOW/FREEZING RAIN", "FREEZING RAIN/SNOW",
      "FREEZING RAIN AND SNOW", "WINTRY MIX", "WINTER WEATHER MIX", "WINTER MIX",
      "WINTERY MIX", "FREEZING DRIZZLE"
      )
)

lookup_table <- data.frame(EVTYPE=c(), NEWEVTYPE=c())

for (i in c(1:length(replacement_list))) {
  rows <- data.frame(EVTYPE=replacement_list[[i]], NEWEVTYPE=replacement_list[[i]][1])
  lookup_table <- rbind(lookup_table, rows)
}

print(head(lookup_table))
```

```
##                                EVTYPE            NEWEVTYPE
## 1          ASTRONOMICAL LOW TIDE ASTRONOMICAL LOW TIDE
## 2                      AVALANCHE            AVALANCHE
## 3                       AVALANCE            AVALANCHE
## 4 HEAVY SNOW/BLIZZARD/AVALANCHE            AVALANCHE
## 5                       BLIZZARD             BLIZZARD
## 6             HIGH WIND/BLIZZARD             BLIZZARD
```

Damage amounts uniformization

The same approach is used for the damage amounts. A conversion list for the exponents is created and used
to build lookup tables for those columns.

```
# Exponents table
exponents_list <- list(
  c("1e9", "B", "b"),
  c("1e6", "M", "m"),
  c("1e3", "K", "k"),
  c("1e2", "H", "h"),
  c("1e10", "10"),
  c("1e9", "9"),
  c("1e8", "8"),
  c("1e7", "7"),
```

```
  c("1e6", "6"),
  c("1e5", "5"),
  c("1e4", "4"),
  c("1e3", "3"),
  c("1e2", "2"),
  c("1e1", "1"),
  c("1e0", "0"),
  c("1", "+", "-", "?", "")
)

exponents_table <- data.frame(CURRENT=c(), NEW=c())
for (i in c(1:length(exponents_list))) {
  rows <- data.frame(CURRENT=exponents_list[[i]], NEW=exponents_list[[i]][1])
  exponents_table <- rbind(exponents_table, rows)
}

cropdmgexp_table <- exponents_table
names(cropdmgexp_table) <- c("CROPDMGEXP", "NEWCROPDMGEXP")

propdmgexp_table <- exponents_table
names(propdmgexp_table) <- c("PROPDMGEXP", "NEWPROPDMGEXP")

print(head(cropdmgexp_table))
```

```
##   CROPDMGEXP NEWCROPDMGEXP
## 1        1e9           1e9
## 2          B           1e9
## 3          b           1e9
## 4        1e6           1e6
## 5          M           1e6
## 6          m           1e6
```

```
print(head(propdmgexp_table))
```

```
##   PROPDMGEXP NEWPROPDMGEXP
## 1        1e9           1e9
## 2          B           1e9
## 3          b           1e9
## 4        1e6           1e6
## 5          M           1e6
## 6          m           1e6
```

Data transformation

From those tables, a new data frame is created by joining them with the initial stormdata. `left_joins` are used to match the lists between them. This allows the non uniform entries from the initial data to be replaced by the cleaned entries from the list.

To ease the matching, the initial event types are cleaned up by transforming them into upper case and removing the spaces before and after them.

New columns (`BGN_YEAR`, `CROPDMG_AMOUNT`, `PROPDMG_AMOUNT`, `POPULATION_AMOUNT`, `ECONOMIC_AMOUNT`) are created by using the numerical values obtained from the joins.

Only the columns `BGN_YEAR`, `BGN_DATE`, `BGN_TIME`, `COUNTYNAME`, `STATE`, `EVTYPE`, `INJURIES`, `FATALITIES`, `PROPDMG_AMOUNT`, `CROPDMG_AMOUNT`, `POPULATION_AMOUNT`, `ECONOMIC_AMOUNT` are selected for the need of the analysis. `distinct` is used to remove duplicate rows created by the joins, probably caused by character encoding problems.

The new data is grouped by the year of the events and their types.

```
clean_data <- stormdata %>%
  mutate(EVTYPE=trimws(toupper(EVTYPE))) %>%
  left_join(lookup_table, by=(c("EVTYPE" = "EVTYPE"))) %>%
  left_join(cropdmgexp_table, by=(c("CROPDMGEXP" = "CROPDMGEXP"))) %>%
  left_join(propdmgexp_table, by=(c("PROPDMGEXP" = "PROPDMGEXP"))) %>%
  mutate(EVTYPE = as.factor(if_else(!is.na(NEWEVTYPE), NEWEVTYPE, EVTYPE))) %>%
  mutate(BGN_YEAR = year(mdy_hms(BGN_DATE))) %>%
  mutate(CROPDMGEXP = as.numeric(if_else(!is.na(NEWCROPDMGEXP), NEWCROPDMGEXP, CROPDMGEXP))) %>%
  mutate(PROPDMGEXP = as.numeric(if_else(!is.na(NEWPROPDMGEXP), NEWPROPDMGEXP, PROPDMGEXP))) %>%
  mutate(CROPDMG_AMOUNT = CROPDMG * CROPDMGEXP) %>%
  mutate(PROPDMG_AMOUNT = PROPDMG * PROPDMGEXP) %>%
  mutate(POPULATION_AMOUNT = INJURIES + FATALITIES) %>%
  mutate(ECONOMIC_AMOUNT = CROPDMG_AMOUNT + PROPDMG_AMOUNT) %>%

  select(BGN_YEAR, BGN_DATE, BGN_TIME, COUNTYNAME, STATE, EVTYPE, INJURIES, FATALITIES,
         PROPDMG_AMOUNT, CROPDMG_AMOUNT,
         POPULATION_AMOUNT, ECONOMIC_AMOUNT, REFNUM) %>%
  distinct(REFNUM, .keep_all = TRUE) %>%
  group_by(BGN_YEAR, EVTYPE)

clean_data_nrows <- nrow(clean_data)
```

Cleaning method helper

A little drawback on using this method is that the list of event types from the data is too diverse and some are combined. To help, a list of non replaced types is built after each transformation. The list is ordered based on the numbers of observations for each type. This eases the iterative process by placing the most important types into the official types group.

```
remaining <- clean_data[!clean_data$EVTYPE %in% official_event_types,] %>%
  group_by(EVTYPE) %>%
  summarise(count=n()) %>%
  arrange(desc(count), .by_group = TRUE) %>%
  print
```

```
## # A tibble: 158 x 2
##    EVTYPE                count
##    <fct>                 <int>
## 1 ICE                       61
## 2 OTHER                     52
## 3 TEMPERATURE RECORD        43
## 4 MIXED PRECIPITATION       37
## 5 MONTHLY PRECIPITATION     36
## 6 SNOW AND ICE              34
## 7 ICY ROADS                 32
## 8 UNSEASONABLY WET          19
```

10

```
##  9 BLACK ICE                17
## 10 BLOWING SNOW             17
## # ... with 148 more rows
```

Final cleanup

The process is stopped when it becomes difficult to match the remaining event types against the official list.

The `TEMPERATURE RECORD` type for example contains positive record (`HEAT`) and negative records (`EXTREME COLD/WIND CHILL`). It also becomes necessary to look inside the `REMARKS` field for the match to be correct enough.

As a result, the rest of the data can be discarded as it is only a small proportion of the `clean_data` table. The rows with unofficial event types are filtered out from the final data.

```
clean_data <- clean_data[clean_data$EVTYPE %in% official_event_types,]

official_events_nrows <- nrow(clean_data)
```

Some analysis on the data show that there are years with only few recorded event types. Those years are also removed to mitigate the bias that they can induce.

```
# Event types by year
event_types_by_year <- clean_data %>%
  select(BGN_YEAR, EVTYPE) %>%
  unique %>%
  aggregate(EVTYPE ~ BGN_YEAR, data=., FUN = length)

types <- event_types_by_year$EVTYPE
names(types) <- event_types_by_year$BGN_YEAR

print(types)
```

```
## 1950 1951 1952 1953 1954 1955 1956 1957 1958 1959 1960 1961 1962 1963 1964 1965
##    1    1    1    1    1    3    3    3    3    3    3    3    3    3    3    3
## 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975 1976 1977 1978 1979 1980 1981
##    3    3    3    3    3    3    3    3    3    3    3    3    3    3    3    3
## 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997
##    3    3    3    3    3    3    3    3    3    3    3   34   37   37   35   38
## 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011
##   38   38   38   39   38   38   37   42   44   46   46   45   45   45
```

```
event_types_by_year <- event_types_by_year[event_types_by_year$EVTYPE > 3,]

types <- event_types_by_year$EVTYPE
names(types) <- event_types_by_year$BGN_YEAR

print(types)
```

```
## 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008
##   34   37   37   35   38   38   38   38   39   38   38   37   42   44   46   46
## 2009 2010 2011
##   45   45   45
```

```r
# Filter clean data
clean_data <- clean_data[clean_data$BGN_YEAR %in% event_types_by_year$BGN_YEAR,] %>%
  ungroup(BGN_YEAR)

filtered_events_nrows <- nrow(clean_data)
```

In the end, the remaining data contains **714005** rows, loosing **188292** rows from the initial data. The remaining rows are about **79%** of the initial data.

- **0** rows where lost during the first cleanup
- **733** rows did not match the official list
- **187559** rows where present in years with less than four observed event types.

Results

Population and Economic Impacts are computed from the cleaned data.

```r
# Impacts
population_amount <- sum(clean_data$POPULATION_AMOUNT)
economic_amount <- sum(clean_data$ECONOMIC_AMOUNT)

event_impacts <- clean_data %>%
  summarise(POPULATION_IMPACT = sum(POPULATION_AMOUNT),
            ECONOMIC_IMPACT = sum(ECONOMIC_AMOUNT)) %>%
  mutate(POPULATION_IMPACT_RATIO = POPULATION_IMPACT / population_amount,
         ECONOMIC_IMPACT_RATIO = ECONOMIC_IMPACT / economic_amount)

# Summarize population impact
population_impacts <- event_impacts %>%
  arrange(POPULATION_IMPACT) %>%
  .[, c("EVTYPE", "POPULATION_IMPACT", "POPULATION_IMPACT_RATIO")]

# Summarize economic impact
economic_impacts <- event_impacts %>%
  arrange(ECONOMIC_IMPACT) %>%
  .[, c("EVTYPE", "ECONOMIC_IMPACT", "ECONOMIC_IMPACT_RATIO")]
```

Population impact

```r
top10_population <- population_impacts %>%
  arrange(desc(POPULATION_IMPACT)) %>%
  head(10) %>%
  mutate(POPULATION_IMPACT = format(POPULATION_IMPACT, big.mark=","))
```

The table bellow shows the **r**nrow(top10_population)' most harmful events to population health based on fatalities and injuries.

The top 3 events are : `TORNADO`, `EXCESSIVE HEAT` and `FLOOD`. Those 3 events constitute the **51%** of the total impacts to population health.

```
top10_population %>%
  kable(col.names = c("Event", "Amount (count)", "Percentage"))
```

Event	Amount (count)	Percentage
TORNADO	24,974	0.3150459
EXCESSIVE HEAT	8,428	0.1063188
FLOOD	7,384	0.0931488
THUNDERSTORM WIND	6,663	0.0840534
LIGHTNING	6,048	0.0762952
HEAT	3,972	0.0501066
FLASH FLOOD	2,835	0.0357634
ICE STORM	2,081	0.0262517
HIGH WIND	1,932	0.0243721
WILDFIRE	1,698	0.0214202

The following figure shows the impact of each event type to population health. Past the top 3 events, there are few events that have about the same relative impact.

```
par(mar = c(6, 12, 2, 2))

barplot(height = population_impacts$POPULATION_IMPACT, names=population_impacts$EVTYPE,
        horiz = TRUE, las=1, cex.names = .5, cex.axis = .8, col="#69b3a2",
        main = "Population health impacts of each event type",
        xlab = "Health impact (count)")
```

Population health impacts of each event type

Health impact (count)
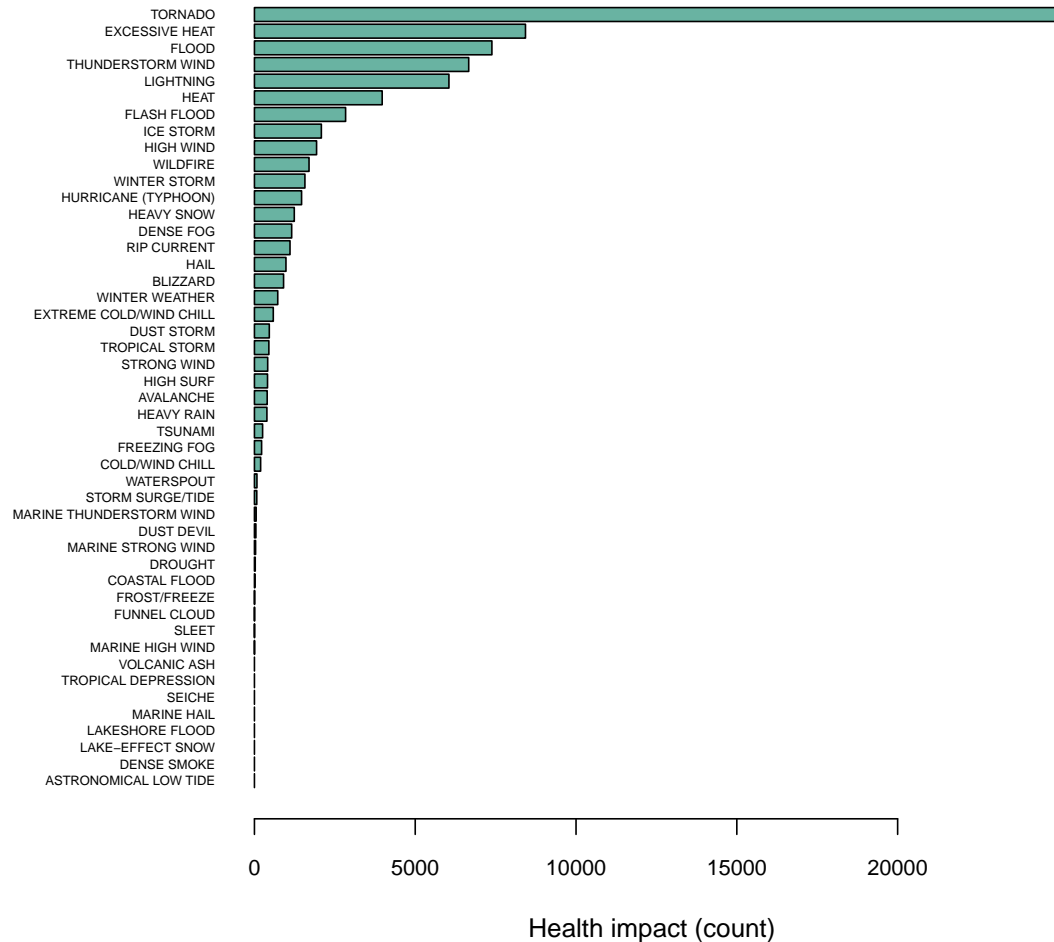
Economic impacts

```
top10_economic <- economic_impacts %>%
  arrange(desc(ECONOMIC_IMPACT)) %>%
  head(10) %>%
  mutate(ECONOMIC_IMPACT = format(ECONOMIC_IMPACT, big.mark=","))
```

The table bellow shows the 10 most harmful events to economy based on the caused damages.

The top 3 events are : `FLOOD`, `HURRICANE (TYPHOON)` and `STORM SURGE/TIDE`. These 3 events make the **67%** of the total impacts.

```
top10_economic %>%
  kable(col.names = c("Event", "Amount ($)", "Percentage"))
```

Event	Amount ($)	Percentage
FLOOD	161,063,280,427	0.3606800
HURRICANE (TYPHOON)	90,762,527,810	0.2032507
STORM SURGE/TIDE	47,966,129,000	0.1074138
TORNADO	29,576,975,376	0.0662337
FLASH FLOOD	19,121,509,246	0.0428201
HAIL	19,024,448,136	0.0426027
DROUGHT	15,018,927,780	0.0336329
THUNDERSTORM WIND	11,253,121,488	0.0251999
ICE STORM	8,968,141,360	0.0200830
WILDFIRE	8,899,910,130	0.0199302

Same as for the health impacts figure, the next figure shows the scale of economic impacts by event.

```
par(mar = c(6, 12, 2, 2))

barplot(height = economic_impacts$ECONOMIC_IMPACT, names=economic_impacts$EVTYPE,
        horiz = TRUE, las=1, cex.names = .5, cex.axis = .8, col="#69b3a2",
        main = "Economic impacts of each event type",
        xlab = "Amount ($)")
```

Economic impacts of each event type

Amount ($)