Exploring the NOAA Storm Database

Jeferson Santos Monteiro

12 de abril de 2019

Synopsis

The purpose of the report is to demonstrate the most severe weather events that have caused the worst consequences on the healthy and economical population in the United States from 1950 to 2011. Based on the National Oceanic and Atmospheric Administration (NOAA) Storm Data. In the results of the analysis it was identified that the hurricane is the most harmful event for health, for the economy the floods cause more negative consequences.

Data Processing

The **R** was used with the `package: dplyr` , for the analysis. The code is reported below with the respective outputs. The code for producing this document was written in **R Markdown**.

```
library(dplyr)                      #Tool for data frame
```

```
## Warning: package 'dplyr' was built under R version 3.5.3
```

```
## 
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
## 
##     filter, lag
```

```
## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union
```

Configuration of the system

```
sessionInfo()
```

```
## R version 3.5.1 (2018-07-02)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 17134)
## 
## Matrix products: default
## 
## locale:
## [1] LC_COLLATE=Portuguese_Brazil.1252  LC_CTYPE=Portuguese_Brazil.1252   
## [3] LC_MONETARY=Portuguese_Brazil.1252 LC_NUMERIC=C                      
## [5] LC_TIME=Portuguese_Brazil.1252    
## 
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base     
## 
## other attached packages:
## [1] dplyr_0.8.0.1
## 
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.1       crayon_1.3.4     digest_0.6.18    assertthat_0.2.1
##  [5] R6_2.4.0         magrittr_1.5     evaluate_0.13    pillar_1.3.1    
##  [9] rlang_0.3.4      stringi_1.4.3    rmarkdown_1.12   tools_3.5.1     
## [13] stringr_1.4.0    glue_1.3.1       purrr_0.3.2      xfun_0.6        
## [17] yaml_2.2.0       compiler_3.5.1   pkgconfig_2.0.2  htmltools_0.3.6 
## [21] tidyselect_0.2.5 knitr_1.22       tibble_2.1.1
```

Loading and processing Data

Data are stored in the standard *comma-separated-value* format, compress with *bzip2* algorithm.

```
storm.data <-read.csv("repdata_data_StormData.csv.bz2",header=TRUE)
```

Selecting data

After loading the data, a table is created to view the information.

They become names in *compatibility names* for R and for applications which do not allow *underline* in names.

```
storm <- tbl_df(storm.data)                               # create table for data

names(storm) <- make.names(names(storm), allow_ = FALSE)   # compatibility names

print(storm)                                              # read the data
```

```
## # A tibble: 902,297 x 37
##    STATE.. BGN.DATE BGN.TIME TIME.ZONE COUNTY COUNTYNAME STATE EVTYPE
##      <dbl> <fct>    <fct>    <fct>      <dbl> <fct>      <fct> <fct>
##  1       1 4/18/19~ 0130     CST           97 MOBILE     AL    TORNA~
##  2       1 4/18/19~ 0145     CST            3 BALDWIN    AL    TORNA~
##  3       1 2/20/19~ 1600     CST           57 FAYETTE    AL    TORNA~
##  4       1 6/8/195~ 0900     CST           89 MADISON    AL    TORNA~
##  5       1 11/15/1~ 1500     CST           43 CULLMAN    AL    TORNA~
##  6       1 11/15/1~ 2000     CST           77 LAUDERDALE AL    TORNA~
##  7       1 11/16/1~ 0100     CST            9 BLOUNT     AL    TORNA~
##  8       1 1/22/19~ 0900     CST          123 TALLAPOOSA AL    TORNA~
##  9       1 2/13/19~ 2000     CST          125 TUSCALOOSA AL    TORNA~
## 10       1 2/13/19~ 2000     CST           57 FAYETTE    AL    TORNA~
## # ... with 902,287 more rows, and 29 more variables: BGN.RANGE <dbl>,
## #   BGN.AZI <fct>, BGN.LOCATI <fct>, END.DATE <fct>, END.TIME <fct>,
## #   COUNTY.END <dbl>, COUNTYENDN <lgl>, END.RANGE <dbl>, END.AZI <fct>,
## #   END.LOCATI <fct>, LENGTH <dbl>, WIDTH <dbl>, F <int>, MAG <dbl>,
## #   FATALITIES <dbl>, INJURIES <dbl>, PROPDMG <dbl>, PROPDMGEXP <fct>,
## #   CROPDMG <dbl>, CROPDMGEXP <fct>, WFO <fct>, STATEOFFIC <fct>,
## #   ZONENAMES <fct>, LATITUDE <dbl>, LONGITUDE <dbl>, LATITUDE.E <dbl>,
## #   LONGITUDE. <dbl>, REMARKS <fct>, REFNUM <dbl>
```

There were 902297 total observations with 37 variables.

The variables interested in are the **type of event** (`EVTYPE`), **fatalities** (`FATALITIES`) and **injuries** (`INJURIES`) and those describing the **ammount of damage** (all fields including `DMG`). Extract the variables and print a sample of ten cases to watch them togheter.

```
use.storm <- storm %>%                                          # from strom

        select(EVTYPE, FATALITIES, INJURIES,                    # select explicit variables

               contains("DMG"))                                 # and the ones containing "DMG"

set.seed(1304)                                                  # set seed

sample_n(use.storm,10)                                          # print a random sample of 10 rows
```

```
## # A tibble: 10 x 7
##    EVTYPE         FATALITIES INJURIES PROPDMG PROPDMGEXP CROPDMG CROPDMGEXP
##    <fct>               <dbl>    <dbl>   <dbl> <fct>        <dbl> <fct>
##  1 TORNADO                 0        1   250   K                0 ""
##  2 TSTM WIND               0        0     2   K                0 ""
##  3 FLASH FLOOD             0        0     0   ""               0 ""
##  4 THUNDERSTORM ~          0        0     8   K                0 K
##  5 HAIL                    0        0     0   ""               0 ""
##  6 TORNADO                 0        0     2.5 M                0 ""
##  7 LIGHTNING               0        0    30   K                0 ""
##  8 WATERSPOUT              0        0     0   K                0 K
##  9 FLASH FLOOD             0        0     0   K                0 K
## 10 TSTM WIND               0        0     0   ""               0 ""
```

Results

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

To reach the objective, only the variables related to the type of event, fatalities and injuries are selected. Subsequently the total amount of both events is considered, then the sum of the deaths and injuries is considered and each event is classified according to what happened, deaths, injuries and the sum of both. Then the first ten events are shown, of total injuries and deaths.

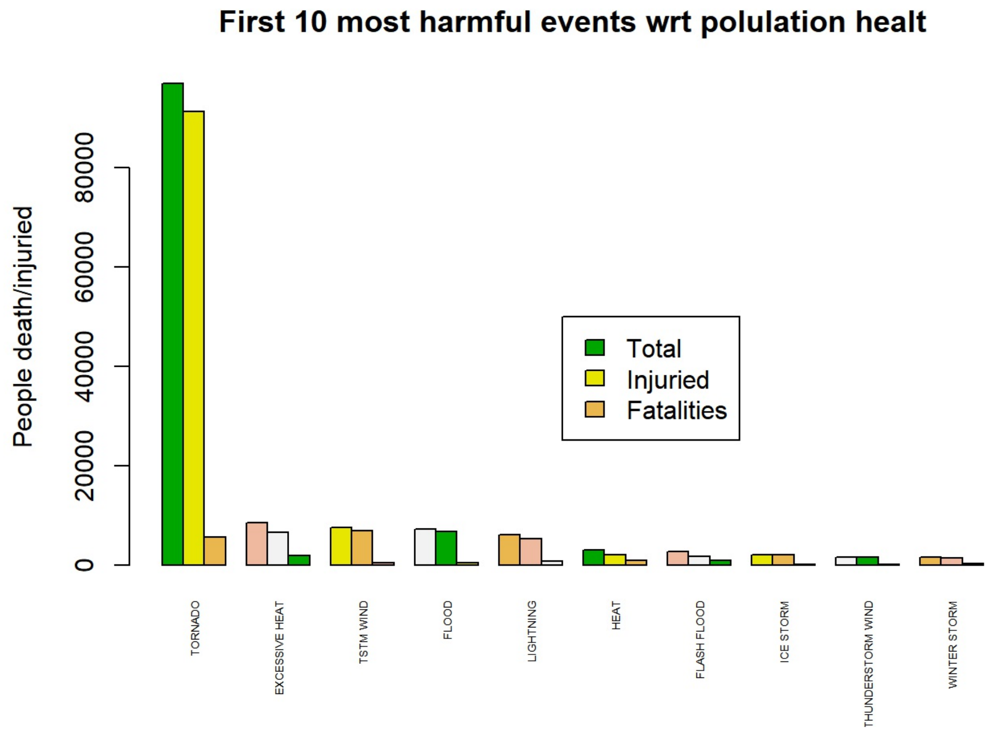
```
health.storm <- use.storm %>%
                select(EVTYPE, FATALITIES, INJURIES) %>%
                group_by(EVTYPE) %>%
                summarise_each(funs(sum)) %>%
                mutate(TOT.HARMFUL=FATALITIES + INJURIES,
                       RK.FAT=dense_rank(desc(FATALITIES)),
                       RK.INJ=dense_rank(desc(INJURIES)),
                       RK.TOT=dense_rank(desc(TOT.HARMFUL))) %>%
                arrange(desc(TOT.HARMFUL),
                        desc(FATALITIES),
                        desc(INJURIES))
```

```
## Warning: funs() is soft deprecated as of dplyr 0.8.0
## please use list() instead
##
## # Before:
## funs(name = f(.)
##
## # After:
## list(name = ~f(.))
## This warning is displayed once per session.
```

```
health.storm
```

```
## # A tibble: 985 x 7
##    EVTYPE            FATALITIES INJURIES TOT.HARMFUL RK.FAT RK.INJ RK.TOT
##    <fct>                  <dbl>    <dbl>       <dbl>  <int>  <int>  <int>
##  1 TORNADO                 5633    91346       96979      1      1      1
##  2 EXCESSIVE HEAT          1903     6525        8428      2      4      2
##  3 TSTM WIND                504     6957        7461      6      2      3
##  4 FLOOD                    470     6789        7259      7      3      4
##  5 LIGHTNING                816     5230        6046      5      5      5
##  6 HEAT                     937     2100        3037      4      6      6
##  7 FLASH FLOOD              978     1777        2755      3      8      7
##  8 ICE STORM                 89     1975        2064     23      7      8
##  9 THUNDERSTORM WIND        133     1488        1621     15      9      9
## 10 WINTER STORM             206     1321        1527     11     11     10
## # ... with 975 more rows
```

```
barplot(t(as.matrix(health.storm[1:10,4:2])),
        main = "First 10 most harmful events wrt polulation healt",
        names.arg = health.storm$EVTYPE[1:10],
        las=3,
        cex.names = 0.45,
        ylab = "People death/injuried",
        beside = TRUE,
        col = terrain.colors(5))
legend(20,50000,c("Total", "Injuried", "Fatalities"),
       fill = terrain.colors(5))
```



It is obvious considered cases, only fatalities, injuries or the sum of both: __*tornado is the most harmful event wrt population healt__.

Which types of events have the greatest economic consequences?

In order to find the event with the greatest economic consequences, first select only the variable of the data set referring to the type of event and those that report the damages. With greater detail, property damage and crop damage are calculated by calculating the amount of damage caused by events. Subsequently, the events are classified for the types of damages and for the total of them. Following is the first ten events that classify the total economic consequences.

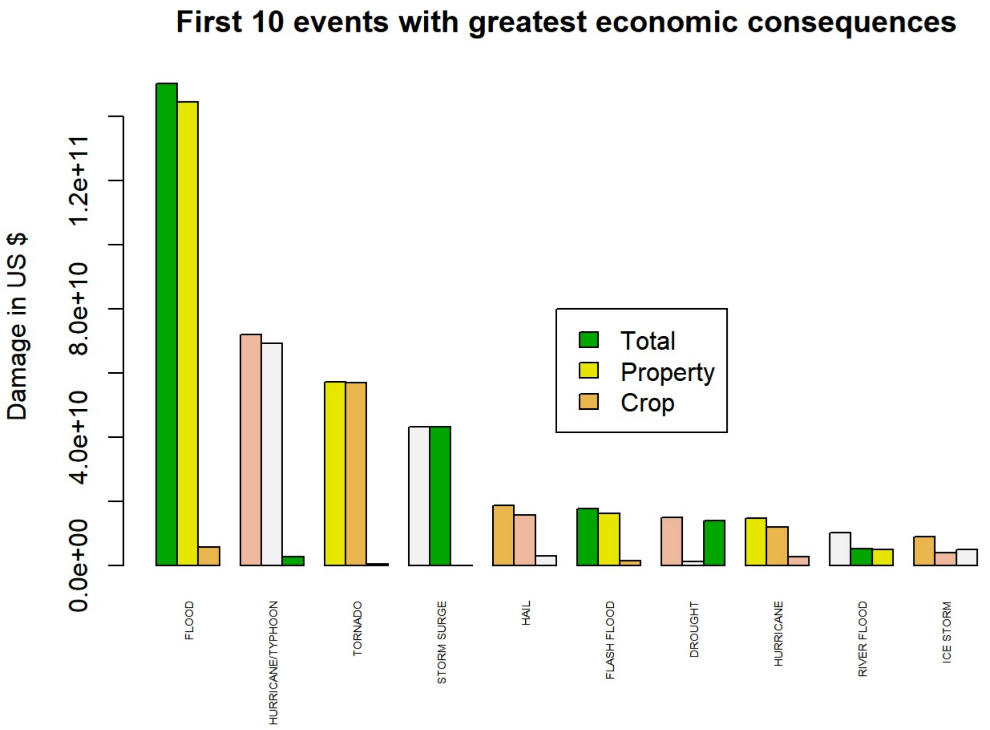
```
PROP.storm <- use.storm %>%
                select(EVTYPE, starts_with("PROP")) %>%
                group_by(EVTYPE, PROPDMGEXP) %>%
                summarize(DAMAGE.SET=sum(PROPDMG)) %>%
                mutate(
                        PROPDAMAGE=ifelse(PROPDMGEXP=="K",
                                     DAMAGE.SET*(10^3),
                                ifelse(PROPDMGEXP=="M",
                                       DAMAGE.SET*(10^6),
                                ifelse(PROPDMGEXP=="B",
                                       DAMAGE.SET*(10^9),
                                DAMAGE.SET)))) %>%
                summarise(TOTPROPDMG=sum(PROPDAMAGE))
CROP.storm <- use.storm %>%
                select(EVTYPE, starts_with("CROP")) %>%
                group_by(EVTYPE, CROPDMGEXP) %>%
                summarize(DAMAGE.SET=sum(CROPDMG)) %>%
                mutate(CROPDAMAGE=ifelse(CROPDMGEXP=="K",
                                     DAMAGE.SET*(10^3),
                                ifelse(CROPDMGEXP=="M",
                                       DAMAGE.SET*(10^6),
                                ifelse(CROPDMGEXP=="B",
                                       DAMAGE.SET*(10^9),
                                DAMAGE.SET)))) %>%
                summarise(TOTCROPDMG=sum(CROPDAMAGE))
DMG.storm <- full_join(PROP.storm,CROP.storm) %>%
                mutate(TOTDMG=TOTPROPDMG + TOTCROPDMG,
                       RK.PROP=dense_rank(desc(TOTPROPDMG)),
                       RK.CROP=dense_rank(desc(TOTCROPDMG)),
                       RK.DMG=dense_rank(desc(TOTDMG))) %>%
                arrange(desc(TOTDMG))
```

```
## Joining, by = "EVTYPE"
```

```
DMG.storm %>% print(width = Inf)
```

```
## # A tibble: 985 x 7
##    EVTYPE               TOTPROPDMG  TOTCROPDMG        TOTDMG RK.PROP
##    <fct>                     <dbl>       <dbl>         <dbl>   <int>
##  1 FLOOD             144657709807   5661968450 150319678257        1
##  2 HURRICANE/TYPHOON  69305840000   2607872800  71913712800        2
##  3 TORNADO            56925660790.   414953270  57340614060.       3
##  4 STORM SURGE        43323536000         5000  43323541000        4
##  5 HAIL               15727367053.  3025537890  18752904943.       6
##  6 FLASH FLOOD        16140812067.  1421317100  17562129167.       5
##  7 DROUGHT             1046106000  13972566000  15018672000       23
##  8 HURRICANE          11868319010   2741910000  14610229010        7
##  9 RIVER FLOOD         5118945500   5029459000  10148404500       11
## 10 ICE STORM           3944927860   5022113500   8967041360       15
##    RK.CROP RK.DMG
##      <int>  <int>
##  1       2      1
##  2       7      2
##  3      17      3
##  4      92      4
##  5       5      5
##  6       8      6
##  7       1      7
##  8       6      8
##  9       3      9
## 10       4     10
## # ... with 975 more rows
```

```
barplot(t(as.matrix(DMG.storm[1:10,c(4,2,3)])),
        main = "First 10 events with greatest economic consequences",
        names.arg = DMG.storm$EVTYPE[1:10],
        las=3,
        cex.names = 0.45,
        ylab = "Damage in US $",
        beside = TRUE,
        col = terrain.colors(5))
legend(20,80000000000,c("Total", "Property", "Crop"),
       fill = terrain.colors(5))
```



This is one events that clearly is the worst one. In this case it is not the tornado, but the *Flood is the events with the greatest economic consequences*.