getwd() setwd("C:/RR/C5") dir()

My github page

Synopsis

In this assignment, we analyzed the most harmful nature events using the data of natural events from he U.S. National Oceanic and Atmospheric Administration's (NOAA) storm database. We first read the data and clean up some event types by looking into the cookbook. Then we aggregate the fatality, body injury, property damag, and crop damage by using the aggregate function according to different event types. With data processing and analyzing, we summarized the most harmful events to human health and the events have strongest damage to property and crop by table and figures. The results are tornado, thunderstrom wind, flood, excessive heat are the most harmful events to human health while while flood, hurricane, tornado, storm surge and hail have the most economic consequences.

Data Processing

Here we prepare data, download source file if it's not exsists, check its structure and make sure it loaded correctly:

```
URL <- "https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2FStormData.csv.bz2"
URL
```

```
## [1] "https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2FStormData.csv.bz2"
```

```
  if(!file.exists("repdata-data-StormData.csv.bz2")){
    res <- tryCatch(download.file(URL,
                                  destfile="./repdata-data-StormData.csv.bz2",
                                  method="auto"),
                    error=function(e) 1)
    if(dat!=1) load("./repdata-data-StormData.csv.bz2")
}


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

str(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 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",..: 13513 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 2 ...
##  $ EVTYPE    : Factor w/ 985 levels "   HIGH SURF ADVISORY",..: 834 834 834 834 834 834 834 834 834 834 ...
##  $ 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 1 ...
##  $ WFO       : Factor w/ 542 levels "","$AC",","$AC","$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/ 436781 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 ...
```

```
class(data)
```

```
## [1] "data.frame"
```

First we clean our data and analyze Fatality category doing aggregation to decrease number of records to be revised and cleaned:

```
trim <- function( x ) {  gsub("(^[[:space:]]+|[[:space:]]+$)", "", x)    }

datax <- data[,c("FATALITIES","INJURIES","EVTYPE","PROPDMG","PROPDMGEXP", "CROPDMGEXP", "CROPDMG")]
datax$EVTYPE <- trim(datax$EVTYPE)
datax$EVTYPE <- tolower(datax$EVTYPE)
head(datax)
```

```
##   FATALITIES INJURIES  EVTYPE PROPDMG PROPDMGEXP CROPDMGEXP CROPDMG
## 1          0       15 tornado    25.0          K                 0
## 2          0        0 tornado     2.5          K                 0
## 3          0        2 tornado    25.0          K                 0
## 4          0        2 tornado     2.5          K                 0
## 5          0        2 tornado     2.5          K                 0
## 6          0        6 tornado     2.5          K                 0
```

```
fatal <- aggregate(FATALITIES ~ EVTYPE, data = datax, sum)
fatalx <- fatal[fatal$FATALITIES > 0, ]
fatalx <- fatalx[grep('[A-Za-z]+', fatalx$EVTYPE),]
head(fatalx)
```

```
##           EVTYPE FATALITIES
## 10      avalance          1
## 11     avalanche        224
## 19     black ice          1
## 20      blizzard        101
## 32  blowing snow          2
## 41 coastal flood          3
```

After examining *nrow* intermediate results in *fatalx* and comparing them with table 2.1.1 in coocbook we noticed that some categories in source document spelled/abbreviated differently so we will take necesseary steps to remap those like specified in document doing manual checking and assighing right Category by intellegent guessing. We put extra column for this for easy tracking which later can be dropped: We convert our dataset to *data.table* to utilize all feature for wild matching

```
library(data.table)
fatalx$TYPENEW <- fatalx$EVTYPE
fatalt <- data.table(fatalx)
class(fatalt)
```

```
## [1] "data.table" "data.frame"
```

```
head(fatalx)
```

```
##           EVTYPE FATALITIES       TYPENEW
## 10      avalance          1      avalance
```

```
## 11     avalanche        224     avalanche
## 19     black ice          1     black ice
## 20      blizzard        101      blizzard
## 32  blowing snow          2  blowing snow
## 41 coastal flood          3 coastal flood
```

```
fatalt[fatalt$EVTYPE  %like% "blizzard",]$TYPENEW = "blizzard"


dup1 <- c("thunderstorm wind","thunderstorm wind (g40)","thunderstorm wind g52","thunderstorm winds","thundertorm winds","tstm wind","tstm wind (g35)","tstm wind/hail", "coastal storm","coastalstorm")
dup2 <- c("wild fires", "wild/forest fire","wildfire")
dup3 <- c("tropical storm","tropical storm gordon")
dup4 <- c("wind","winds", "wnd", "tstm wind (g45)")
dup5 <- c("winter storm","winter storms")
dup6 <- c("urban and small stream floodin","urban/sml stream fld")
dup7 <- c("coastal flood","beach erosion/coastal flood","coastal  flooding/erosion","coastal flooding","coastal flooding/erosion","coastal/tidal flood","coastalflood","cstl flooding/erosion","erosion/cstl flood","beach flood","breakup flooding")
dup8 <- c("avalance","avalanche")
dup9 <- c("cold/wind chill","cold/winds")
dup10<- c("high wind","high wind and seas","high wind/seas","high winds","high winds/snow")
dup11<- c("winter storm","winter storm high winds","winter storms")
dup12<- c("winter weather","winter weather/mix)")


fatalt[fatalt$EVTYPE  %in% dup1, ]$TYPENEW = "thunderstorm wind"
fatalt[fatalt$EVTYPE  %in% dup2, ]$TYPENEW = "wildfire"
fatalt[fatalt$EVTYPE  %in% dup3, ]$TYPENEW = "tropical storm"
fatalt[fatalt$EVTYPE  %in% dup4, ]$TYPENEW = "strong wind"
fatalt[fatalt$EVTYPE  %in% dup5, ]$TYPENEW = "winter storm"
fatalt[fatalt$EVTYPE  %in% dup6, ]$TYPENEW = "flood"

fatalt[fatalt$EVTYPE  %in% dup7, ]$TYPENEW = "coastal flood"
fatalt[fatalt$EVTYPE  %in% dup8, ]$TYPENEW = "avalanche"
fatalt[fatalt$EVTYPE  %in% dup9, ]$TYPENEW = "cold/wind chill"
fatalt[fatalt$EVTYPE  %in% dup10, ]$TYPENEW = "high wind"
fatalt[fatalt$EVTYPE  %in% dup11, ]$TYPENEW = "winter storm"
fatalt[fatalt$EVTYPE  %in% dup12, ]$TYPENEW = "winter weather"
```

After this cleaning we do another round of summary to see if *EVTYPE* are correct, total number of EVTYPE in coockbook = 48. We go thru several iteration of this step putting more translation in above steps to get right results in sorted order:

```
nrow(fatalt)
```

```
## [1] 160
```

```
fatalt <- aggregate(FATALITIES ~ TYPENEW, data = fatalt, sum)
nrow(fatalt)
```

```
## [1] 135
```

```
max(fatal$FATALITIES)
```

```
## [1] 5633
```

```
mean(fatal$FATALITIES)
```

```
## [1] 17.01685
```

```
fatalSort <- fatalt[order(fatalt$FATALITIES, decreasing = TRUE), ]
head(fatalSort)
```

```
##               TYPENEW FATALITIES
## 120           tornado       5633
## 18     excessive heat       1903
## 26        flash flood        978
## 48               heat        937
## 81          lightning        816
## 119 thunderstorm wind        714
```

Now we do same processign for Injuries: First do aggregate to decrease number of records to be revised and clean/remap column names

```
injury <- aggregate(INJURIES ~ EVTYPE, data = datax, sum)
injuryx <- injury[injury$INJURIES > 0, ]
injuryx <- injuryx[grep('[A-Za-z]+', injuryx$EVTYPE),]
head(injuryx)
```

```
##           EVTYPE INJURIES
## 11     avalanche      170
## 19     black ice       24
## 20      blizzard      805
## 32  blowing snow       14
## 37    brush fire        2
## 41 coastal flood        2
```

```
injuryt <- data.table(injuryx)

injuryt[injuryt$EVTYPE  %in% dup1, ]$EVTYPE = "thunderstorm wind"
injuryt[injuryt$EVTYPE  %in% dup2, ]$EVTYPE = "wildfire"
injuryt[injuryt$EVTYPE  %in% dup3, ]$EVTYPE = "tropical storm"
injuryt[injuryt$EVTYPE  %in% dup4, ]$EVTYPE = "strong wind"
injuryt[injuryt$EVTYPE  %in% dup5, ]$EVTYPE = "winter storm"
injuryt[injuryt$EVTYPE  %in% dup6, ]$EVTYPE = "flood"

injuryt[injuryt$EVTYPE  %in% dup7, ]$EVTYPE = "coastal flood"
injuryt[injuryt$EVTYPE  %in% dup8, ]$EVTYPE = "avalanche"
injuryt[injuryt$EVTYPE  %in% dup9, ]$EVTYPE = "cold/wind chill"
injuryt[injuryt$EVTYPE  %in% dup10, ]$EVTYPE = "high wind"
injuryt[injuryt$EVTYPE  %in% dup11, ]$EVTYPE = "winter storm"
injuryt[injuryt$EVTYPE  %in% dup12, ]$EVTYPE = "winter weather"
head(injuryt)
```

```
##           EVTYPE INJURIES
## 1:     avalanche      170
## 2:     black ice       24
## 3:      blizzard      805
## 4:  blowing snow       14
## 5:    brush fire        2
## 6: coastal flood        2
```

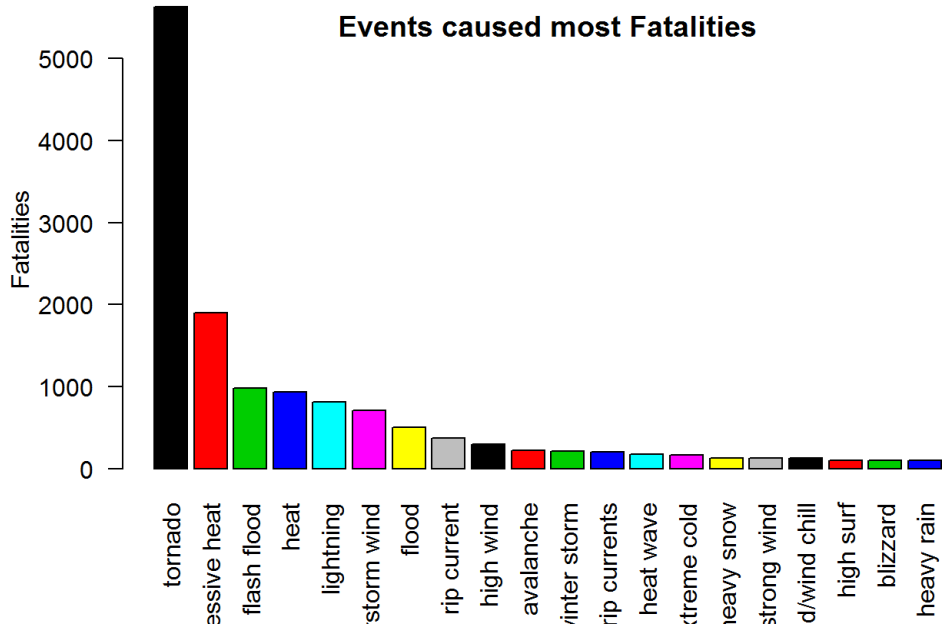
```
injurySort <- injuryt[order(injuryt$INJURIES, decreasing = TRUE), ]
head(injurySort)
```

```
##               EVTYPE INJURIES
## 1:           tornado    91346
## 2: thunderstorm wind     6957
## 3:             flood     6789
## 4:    excessive heat     6525
## 5:         lightning     5230
## 6:              heat     2100
```

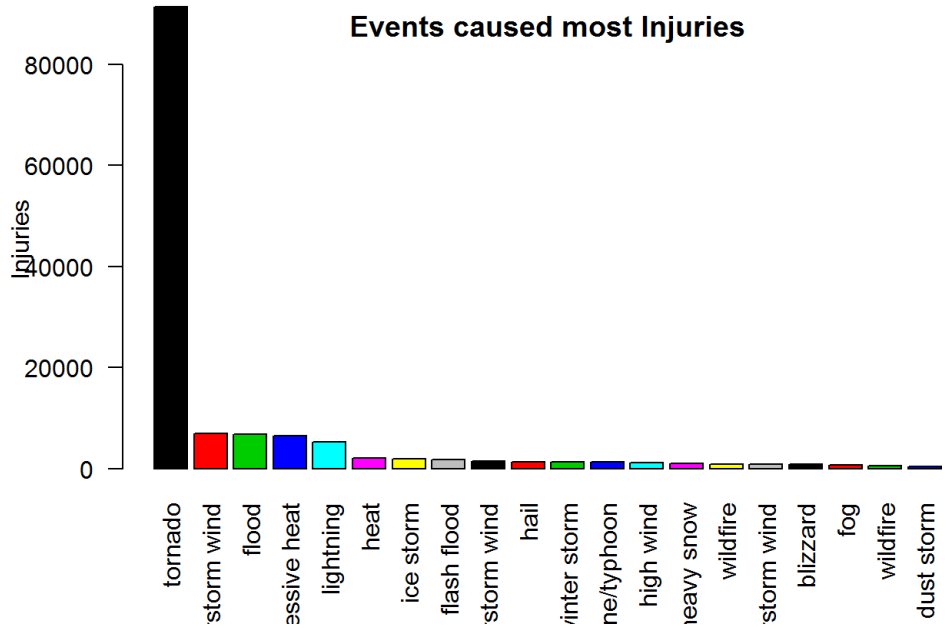
Display results

Now we going to visualize our results using basic plots for total numbers side by side. For Fatalities and Injuries, note that Tornado is main reason for both:

```
barplot (height = fatalSort$FATALITIES[1:20], names.arg = fatalSort$TYPENEW[1:20], las = 2, col = 1:20)
title (main = "Events caused most Fatalities", line=-1)
title (ylab = "Fatalities")
```




```
barplot (height = injurySort$INJURIES[1:20], names.arg = injurySort$EVTYPE[1:20], las = 2,  col = 1:20)
title (main = "Events caused most Injuries", line=-1)
title (ylab = "Injuries")
```



Now we examine and display damage effect on Property and Crop sector by each of events in top 20.

```
head(datax)
```

```
##   FATALITIES INJURIES  EVTYPE PROPDMG PROPDMGEXP CROPDMGEXP CROPDMG
## 1          0       15 tornado    25.0          K                 0
## 2          0        0 tornado     2.5          K                 0
## 3          0        2 tornado    25.0          K                 0
## 4          0        2 tornado     2.5          K                 0
## 5          0        2 tornado     2.5          K                 0
## 6          0        6 tornado     2.5          K                 0
```

```
unique(datax$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(datax$CROPDMGEXP)
```

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

```
symbol <- c("", "+", "-", "?", 0:9, "h", "H", "k", "K", "m", "M", "b", "B");
factor <- c(rep(0,4), 0:9, 2, 2, 3, 3, 6, 6, 9, 9)
df <- data.frame (symbol, factor)


datax$damage.prop <- datax$PROPDMG*10^df[match(datax$PROPDMGEXP,df$symbol),2]
datax$damage.crop <- datax$CROPDMG*10^df[match(datax$CROPDMGEXP,df$symbol),2]
datax$damage <- datax$damage.prop + datax$damage.crop

sum(is.na(datax$CROPDMG & datax$PROPDMG))
```

```
## [1] 0
```

```
datax[is.na(datax$CROPDMG | datax$PROPDMG)] <- 0
head(datax)
```

```
##   FATALITIES INJURIES  EVTYPE PROPDMG PROPDMGEXP CROPDMGEXP CROPDMG
## 1          0       15 tornado    25.0          K                 0
## 2          0        0 tornado     2.5          K                 0
## 3          0        2 tornado    25.0          K                 0
## 4          0        2 tornado     2.5          K                 0
## 5          0        2 tornado     2.5          K                 0
## 6          0        6 tornado     2.5          K                 0
##   damage.prop damage.crop damage
## 1       25000           0  25000
## 2        2500           0   2500
## 3       25000           0  25000
## 4        2500           0   2500
## 5        2500           0   2500
## 6        2500           0   2500
```

```
damage <- aggregate (damage~EVTYPE, datax, sum);

damage$bilion <- round((damage$damage / 1000000000),digits=1);
damage <- damage [order(damage$bilion, decreasing=TRUE),]
head(damage)
```

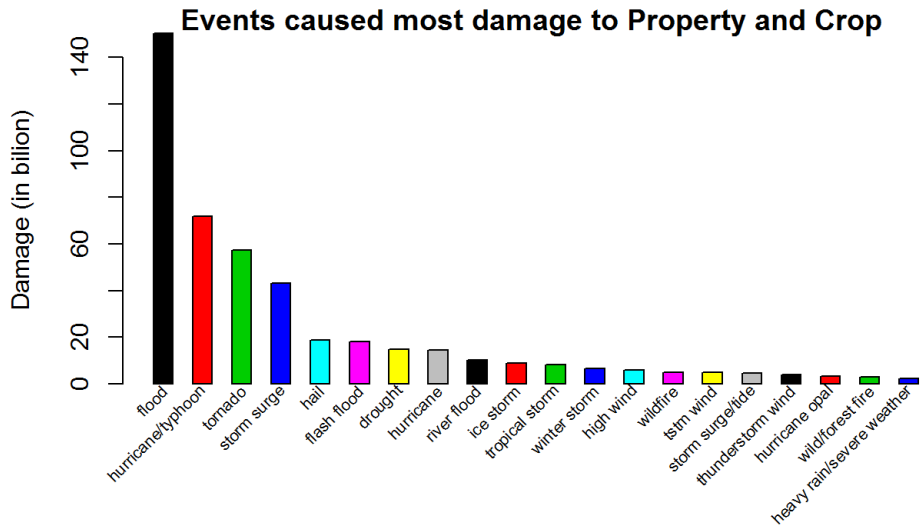
```
##                EVTYPE       damage bilion
## 146             flood 150319678257  150.3
## 364 hurricane/typhoon  71913712800   71.9
```

```
## 750           tornado  57362333947   57.4
## 591       storm surge  43323541000   43.3
## 204              hail  18761221986   18.8
## 130       flash flood  18244041079   18.2
```

```
labels <- damage$EVTYPE[1:20]
par(mar=c(12, 6, 1, 1))
end_point <- 39.5

par(mar=c(12, 6, 1, 1))
mp <-
  barplot (height = damage$bilion[1:20], main = "Events caused most damage to Property and Crop ",
               xlab = "", space=1, col = 1:20)
title (ylab = "Damage (in bilion)")

text(seq(1.5,end_point,by=2), par("usr")[3]-0.25,
     srt = 45, adj= 1, xpd = TRUE,
     labels = labels, cex=0.65)
```



To see exact breakdown numbers for Properties and Crop and for training purposes we display top 10 events using following calculations:

```
damage2 <- aggregate (cbind(damage.prop, damage.crop)~EVTYPE, datax, sum);
damage2 <- damage2[grep('[A-Za-z]+', damage2$EVTYPE),]
str(damage2)
```

```
## 'data.frame':    889 obs. of  3 variables:
##  $ EVTYPE     : chr  "abnormal warmth" "abnormally dry" "abnormally wet" "accumulated snowfall" ...
##  $ damage.prop: num  0 0 0 0 0 ...
##  $ damage.crop: num  0 0 0 0 28820000 ...
```

```
head(damage2)
```

```
##                EVTYPE damage.prop damage.crop
## 2      abnormal warmth           0           0
## 3       abnormally dry           0           0
## 4       abnormally wet           0           0
## 5 accumulated snowfall           0           0
## 6  agricultural freeze           0    28820000
## 7        apache county        5000           0
```

```r
# 10 events caused most Property damage
prop10 <- head(damage2[order(damage2$damage.prop, decreasing = TRUE), ],10)
prop10$Index <- 1:nrow(prop10)
prop10[,c(4,1,2,3)]
```

```
##     Index            EVTYPE  damage.prop damage.crop
## 146     1             flood 144657709807  5661968450
## 364     2 hurricane/typhoon  69305840000  2607872800
## 750     3           tornado  56947380677   414953270
## 591     4       storm surge  43323536000        5000
## 130     5       flash flood  16822723979  1421317100
## 204     6              hail  15735267513  3025954473
## 355     7         hurricane  11868319010  2741910000
## 764     8    tropical storm   7703890550   678346000
## 880     9      winter storm   6688497251    26944000
## 312    10         high wind   5270046295   638571300
```

```r
# 10 events caused most Crop damage
crop10 <- head(damage2[order(damage2$damage.crop, decreasing = TRUE), ],10)
crop10$Index <- 1:nrow(crop10)
crop10[,c(4,1,3,2)]
```

```
##     Index            EVTYPE damage.crop  damage.prop
## 76      1           drought 13972566000   1046106000
## 146     2             flood  5661968450 144657709807
## 521     3       river flood  5029459000   5118945500
## 379     4         ice storm  5022113500   3944927860
## 204     5              hail  3025954473  15735267513
## 355     6         hurricane  2741910000  11868319010
## 364     7 hurricane/typhoon  2607872800  69305840000
## 130     8       flash flood  1421317100  16822723979
## 117     9      extreme cold  1312973000     67737400
## 179    10      frost/freeze  1094186000     10480000
```