Course Project 2

Erik Rehnberg Steeb

7/19/2020

```
```r
knitr::opts_chunk$set(echo = TRUE, eval = TRUE)

library(knitr)
library(dplyr)

##
## 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

library(magrittr)
library(stringr)
```

Synopsis

There are a wide variety of severe weather events that impact the United States on any given year. Most of these do not cause much, if any, damage in the form of either direct economic damage or public health impacts, as measured by reported injuries and fatalities. Others, however, cause billions of dollars and can cost thousands of lives. Preparing exclusively for either the modal no damage scenario or the extreme outlier events, however, are not effective strategies. This analysis takes a set of data from the National Weather Service and analyzes it with an eye towards which events cause the worst damage, both to public health and to the economy.

Data Processing

Initial, the hope was that, given the provenance from the NWS, the dataset for this project was tidy and did require much work in order to be analayzed, despite the overall size of the file. This was, of course, not actually the case. Initial reading in of the data shows 985 different event types, many of which are simple variations on each other. The most common ones were in the pattern "THUNDERSTORM [wind speed]", "Summary of [date]," and "Hurricane/Tropical Storm [storm name]." Some simple regular expressions are used in the processing code in order to standardize all named storm to storm type, keep all thunderstorms together,
```

and keep summary of dates as a single category. This isn't perfect, but it removes almost 200 seperate EVTYPES. I have also shifted everything to UPPERCASE because of some inconsistent capitalization in the original data.

My initial code is shown below, first simply reading in the file from the working directory. Then, I set the event type variable, EVTYPE, to factors, which helps with aggregation.

Finally, the code multiplies the two damage columns (PROPDMG and CROPDMG) by their respective exponent columns to properly display the magnitude of each set of damage. K results in 1000x; M means 1,000,000; B, 1,000,000,000. All non-standard letters are converted to a multiple of 1, all non-standard numbers are left as-is.

```
# Read and process data assuming repdata_data_StormData.csv.bz2 is in your wd

data <- read.csv("repdata_data_StormData.csv.bz2")

# Cleaning event types

test2 <- data$EVTYPE %>%
  toupper()

test2[grep("THUNDERSTORM*", test2)] <- "THUNDERSTORM"
test2[grep("TSTM",test2)] <- "THUNDERSTORM"
test2[grep("SUMMARY.*", test2)] <- "SUMMARY"
test2[grep("TROPICAL STORM", test2)] <- "TROPICAL STORM"
test2[grep("HURRICANE", test2)] <- "HURRICANE"

data$EVTYPE <- test2

# Convert *EXP columns to numeric multiples
prop_dmg_exp <- data$PROPDMGEXP

prop_dmg_exp[grep("K", prop_dmg_exp, ignore.case = TRUE)] <- 1000
prop_dmg_exp[grep("B", prop_dmg_exp, ignore.case = TRUE)] <- 1000000
prop_dmg_exp[grep("M", prop_dmg_exp, ignore.case = TRUE)] <- 1000000000
prop_dmg_exp[grep("[^0-9]", prop_dmg_exp)] <- 1

prop_dmg_exp <- as.numeric(prop_dmg_exp)
prop_dmg_exp[is.na(prop_dmg_exp)] <- 0

data$PROPDMGEXP <- prop_dmg_exp

crop_dmg_exp <- data$CROPDMGEXP

crop_dmg_exp[grep("K", crop_dmg_exp)] <- 1000
crop_dmg_exp[grep("K", crop_dmg_exp)] <- 1000000
crop_dmg_exp[grep("K", crop_dmg_exp)] <- 1000000000
crop_dmg_exp[grep("[^0-9]", crop_dmg_exp)] <- 1

crop_dmg_exp <- as.numeric(crop_dmg_exp)
crop_dmg_exp[is.na(crop_dmg_exp)] <- 0

data$CROPDMGEXP <- crop_dmg_exp

# Data selection, sets EVTYPE to factor, remove data variable to free up 500mb of RAM
```

```
data$EVTYPE <- as.factor(data$EVTYPE)
proc_data_injuries <- data%>%
  dplyr::select(EVTYPE, FATALITIES, INJURIES)%>%
  dplyr::group_by(EVTYPE)
proc_data_damage <- data%>%
  select(EVTYPE, PROPDMG, PROPDMGEXP,
                CROPDMG, CROPDMGEXP)%>%
  dplyr::group_by(EVTYPE)%>%
  mutate(Act.Crop.Dmg = CROPDMG * CROPDMGEXP,
         Act.Prop.Dmg = PROPDMG * PROPDMGEXP
  )
rm(data)
```

Most Harmful Events to PubliC Health

Over the course of the dataset, we see the following headline numbers of deaths and injuries:

```
n_fatalities <- as.numeric(sum(proc_data_injuries$FATALITIES))
n_injuries <- as.numeric(sum(proc_data_injuries$INJURIES))

print(paste("Over all the events listed in the dataset, there were",
      n_fatalities,
      "deaths and",
      n_injuries,
      "injuries.",
      sep = " "))
```

```
## [1] "Over all the events listed in the dataset, there were 15145 deaths and 140528 injuries."
```

```
no_death_injuries <- filter(proc_data_injuries, FATALITIES == 0 & INJURIES == 0)

with_deaths <- filter(proc_data_injuries, FATALITIES != 0)
with_injuries <- filter(proc_data_injuries, INJURIES != 0)
with_both <- filter(proc_data_injuries, FATALITIES != 0 & INJURIES !=0)

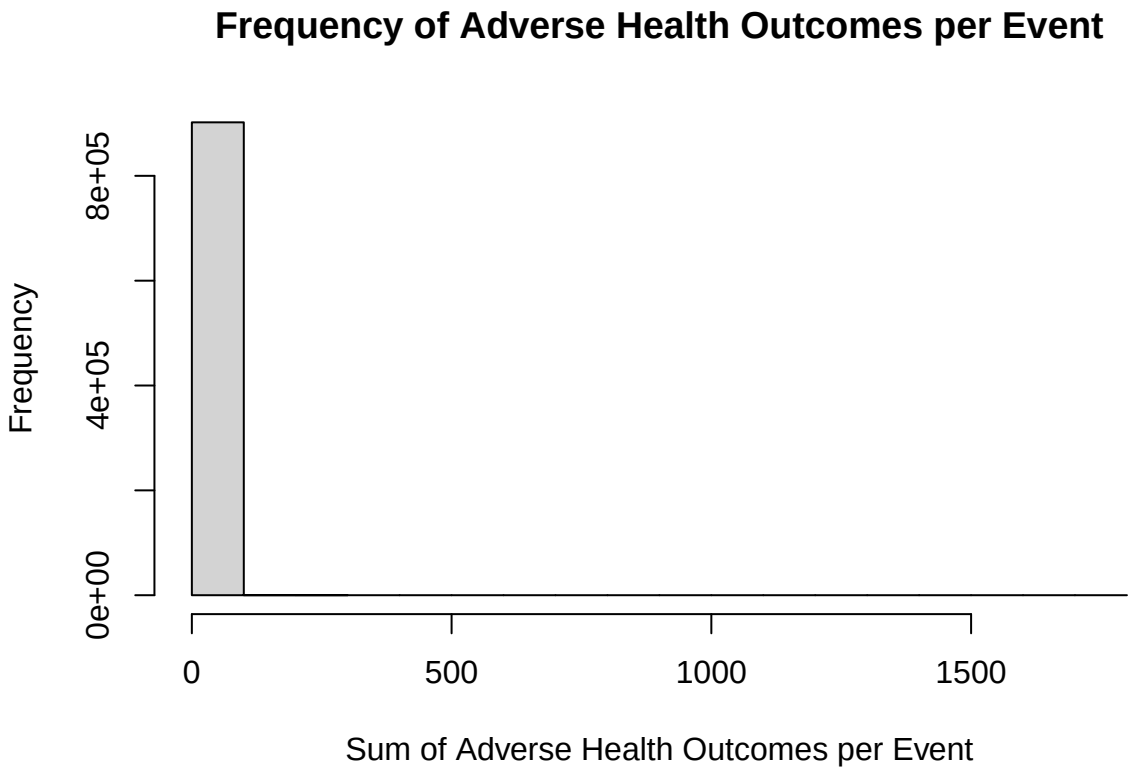
ev_w_death <- nrow(with_deaths)
ev_w_inj <- nrow(with_injuries)
ev_w_both <- nrow(with_both)


print(paste0("Of the events in the database, the vast majority, over ",
             trunc((100 * (nrow(no_death_injuries)/nrow(proc_data_injuries)))),
             "%, had neither injuries nor deaths associated with them."))
```

```
## [1] "Of the events in the database, the vast majority, over 97%, had neither injuries nor deaths asso
```

There were 6974 events with deaths associated, 17604 events with injuries associated, and 2649 events with both injuries and deaths associated. Fortunately, the vast majority of events, as highlighted by the plot below, had no adverse health effects associated with them:

```r
proc_data_injuries$Adverse <- rowSums(proc_data_injuries[,2:3])

hist(proc_data_injuries$Adverse,xlab = "Sum of Adverse Health Outcomes per Event", main = "Frequency of
```



Both fatalities and injuries cause damage to public health, although clearly not at the same scale. Typically, injuries can be compared to outright fatalities by looking at a measure called Disability-Adjusted Life Years (DALY). Unfortunately, this dataset has no information on either injury severity or number of life-years lost per fatality. Assuming that twenty injuries have the same negative impact on public health as a single fatality is most likely not a particularly good estimate, but without any other information or guidance from the NWS or the course assignment,

```r
# Aggregate numbers by EVTYPE

agg_list <- as.list("FATALITIES", "INJURIES")

fat_agg_by_evtype <- aggregate(proc_data_injuries$FATALITIES, by = list(proc_data_injuries$EVTYPE), sum
colnames(fat_agg_by_evtype) <- c("EVTYPE", "Fatalities")
inj_agg_by_evtype <- aggregate(proc_data_injuries$INJURIES, by = list(proc_data_injuries$EVTYPE), sum)
colnames(inj_agg_by_evtype) <- c("EVTYPE", "Injuries")

df <- cbind(inj_agg_by_evtype, fat_agg_by_evtype$Fatalities)
colnames(df) <- c("EVTYPE", "Injuries", "Fatalities")

eval_df <- df%>%
    mutate(Adj.Injuries = df$Injuries /20)
```

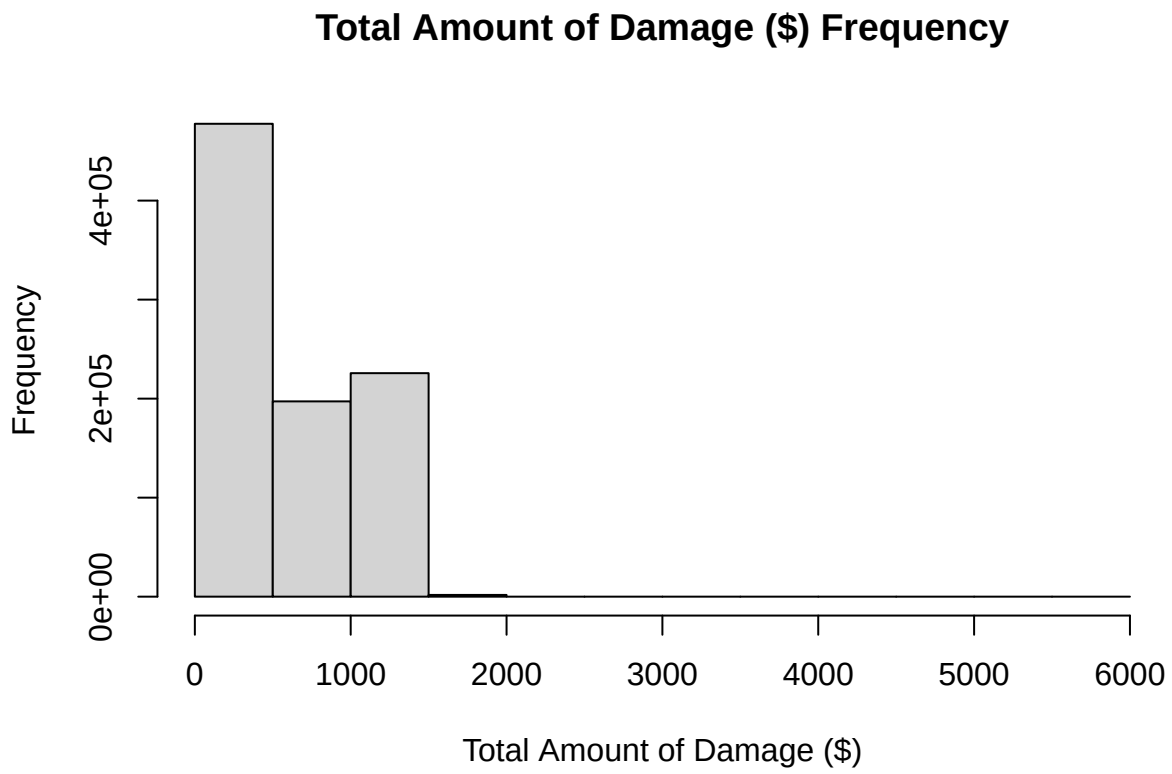
```
eval_df$Sum.Adverse <-  rowSums(eval_df[,3:4])


worst_event <- eval_df[which.max(eval_df$Sum.Adverse),]
worst_event_name <- tolower(as.character(worst_event[1,1]))
```

Using our definitions above, the worst event type is tornado, which led to 9.1346×10^4 injuries, 5633 fatalities, and a sum of adjusted injuries and fatalities of 4567.3.

Most Economically Devastating Events

Once again, there are two distinct types of economic damage - property and crop damage. These are, however, both denominated as a cash figure, so they are very easy to compare. All analysis going forward looks at the sum of damages. Similarly to deaths, damages are very heavily clustered around 0, with long tails for more extreme or damaging events. Since this skew can overweight specific, rare, but highly dangerous events, this analysis will focus on the mean economic damage per event, including ones for which no damage was reported.

```
proc_data_damage$Combined.Damage <- rowSums(proc_data_damage[,2:3])
hist(proc_data_damage$Combined.Damage, xlab = "Total Amount of Damage ($)", main = "Total Amount of Dama
```




```
df <- aggregate(proc_data_damage$Combined.Damage, by = list(proc_data_damage$EVTYPE), mean)

colnames(df) <- c("EVTYPE", "Combined.Damage")
```

```
most_damage <- df[which.max(df$Combined.Damage),]
most_damage_evtype <- tolower(as.character(most_damage[1,1]))
most_damage_num <- trunc(as.numeric(most_damage[1,2]))
```

The most economically damaging type of event is coastal erosion, which causes an average of $1766 each occurance. ##