Reproducible Research Project 2

Dhruba

September 13, 2017

Reproducible Research Project 2

Health and Economic Impact of Weather Events in the US

Storms and other severe weather events can cause both public health and economic problems for communities and municipalities. Many severe events can result in fatalities, injuries, and property damage, and preventing such outcomes to the extent possible is a key concern.

This project involves exploring the U.S. National Oceanic and Atmospheric Administration's (NOAA) storm database. This database tracks characteristics of major storms and weather events in the United States, including when and where they occur, as well as estimates of any fatalities, injuries, and property damage.

Synopsis

Tornadoes followed by excessive heat are the most dangerous event wrt to population health. On the economic front flash floods and thunderstorm wreck havoc. Crops are majorly damaged by drought, flood, hails.

Loading File

```
library(data.table)
storm <- fread('StormData.csv', header = T, sep = ',')
```

```
##
Read 3.1% of 967216 rows
Read 22.7% of 967216 rows
Read 36.2% of 967216 rows
Read 48.6% of 967216 rows
Read 55.8% of 967216 rows
```

```
Read 68.2% of 967216 rows
Read 73.4% of 967216 rows
Read 75.5% of 967216 rows
Read 77.5% of 967216 rows
Read 79.6% of 967216 rows
Read 83.7% of 967216 rows
Read 88.9% of 967216 rows
Read 902297 rows and 37 (of 37) columns from 0.523 GB file in 00:00:17
```

```
# number of unique event types
length(unique(storm$EVTYPE))
```

```
## [1] 985
```

```
# translate all letters to lowercase
event_types <- tolower(storm$EVTYPE)
# replace all punct. characters with a space
event_types <- gsub("[[:blank:][:punct:]+]", " ", event_types)
length(unique(event_types))
```

```
## [1] 874
```

```
# update the data frame
storm$EVTYPE <- event_types
```

Dangerous Events wrt Population Health

The number of casualties are aggregated event wise.

```
library(plyr)
```

```
## Warning: package 'plyr' was built under R version 3.4.1
```

```
casualties <- ddply(storm, .(EVTYPE), summarize,
                    fatalities = sum(FATALITIES),
                    injuries = sum(INJURIES))

# Find events that caused most death and injury
fatal_events <- head(casualties[order(casualties$fatalities, decreasing = T), ], 10)
injury_events <- head(casualties[order(casualties$injuries, decreasing = T), ], 10)
```

```
fatal_events[, c("EVTYPE", "fatalities")]
```

```
injury_events[, c("EVTYPE", "injuries")]
```

```
##                EVTYPE injuries
## 741           tornado    91346
## 762         tstm wind     6957
## 154             flood     6789
## 116    excessive heat     6525
## 410         lightning     5230
## 240              heat     2100
## 382         ice storm     1975
## 138       flash flood     1777
## 671 thunderstorm wind     1488
## 209              hail     1361
```

Economic damages

The available property damage and crop damage reportings/estimates were used to estimate economic damages

```
exp_transform <- function(e) {
  # h -> hundred, k -> thousand, m -> million, b -> billion
  if (e %in% c('h', 'H'))
    return(2)
  else if (e %in% c('k', 'K'))
    return(3)
  else if (e %in% c('m', 'M'))
    return(6)
  else if (e %in% c('b', 'B'))
    return(9)
  else if (!is.na(as.numeric(e))) # if a digit
    return(as.numeric(e))
  else if (e %in% c('', '-', '?', '+'))
    return(0)
  else {
    stop("Invalid exponent value.")
  }
}
```

```
prop_dmg_exp <- sapply(storm$PROPDMGEXP, FUN=exp_transform)
```

```
## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion
```

```
## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion
```

```
storm$prop_dmg <- storm$PROPDMG * (10 ** prop_dmg_exp)
crop_dmg_exp <- sapply(storm$CROPDMGEXP, FUN=exp_transform)
```

```
## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion

## Warning in FUN(X[[i]], ...): NAs introduced by coercion
```

```
storm$crop_dmg <- storm$CROPDMG * (10 ** crop_dmg_exp)
```

```
library(plyr)
econ_loss <- ddply(storm, .(EVTYPE), summarize,
                   prop_dmg = sum(prop_dmg),
                   crop_dmg = sum(crop_dmg))

# filter out events that caused no economic loss
econ_loss <- econ_loss[(econ_loss$prop_dmg > 0 | econ_loss$crop_dmg > 0), ]
prop_dmg_events <- head(econ_loss[order(econ_loss$prop_dmg, decreasing = T), ], 10)
crop_dmg_events <- head(econ_loss[order(econ_loss$crop_dmg, decreasing = T), ], 10)
```

```
prop_dmg_events[, c("EVTYPE", "prop_dmg")]
```

```
##               EVTYPE     prop_dmg
## 154            flood 144657709807
```

```
## 366 hurricane typhoon  69305840000
## 741           tornado  56947380677
## 585       storm surge  43323536000
## 138       flash flood  16822673979
## 209              hail  15735267513
## 357         hurricane  11868319010
## 755    tropical storm   7703890550
## 866      winter storm   6688497251
## 314         high wind   5270046295
```

```
crop_dmg_events[, c("EVTYPE", "crop_dmg")]
```

```
##                EVTYPE    crop_dmg
## 84            drought 13972566000
## 154             flood  5661968450
## 519       river flood  5029459000
## 382         ice storm  5022113500
## 209              hail  3025954473
## 357         hurricane  2741910000
## 366 hurricane typhoon  2607872800
## 138       flash flood  1421317100
## 125      extreme cold  1312973000
## 185      frost freeze  1094186000
```

RESULTS

Impact on Population Health

Top dangerous weather event types.

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.4.1
```

```
# Set the levels in order
p1 <- ggplot(data=fatal_events,
             aes(x=reorder(EVTYPE, fatalities), y=fatalities, fill=fatalities)) +
  geom_bar(stat="identity") +
  coord_flip() +
  ylab("Total number of fatalities") +
  xlab("Event type") +
  theme(legend.position="none")

p2 <- ggplot(data=injury_events,
             aes(x=reorder(EVTYPE, injuries), y=injuries, fill=injuries)) +
  geom_bar(stat="identity") +
  coord_flip() +
  ylab("Total number of injuries") +
```

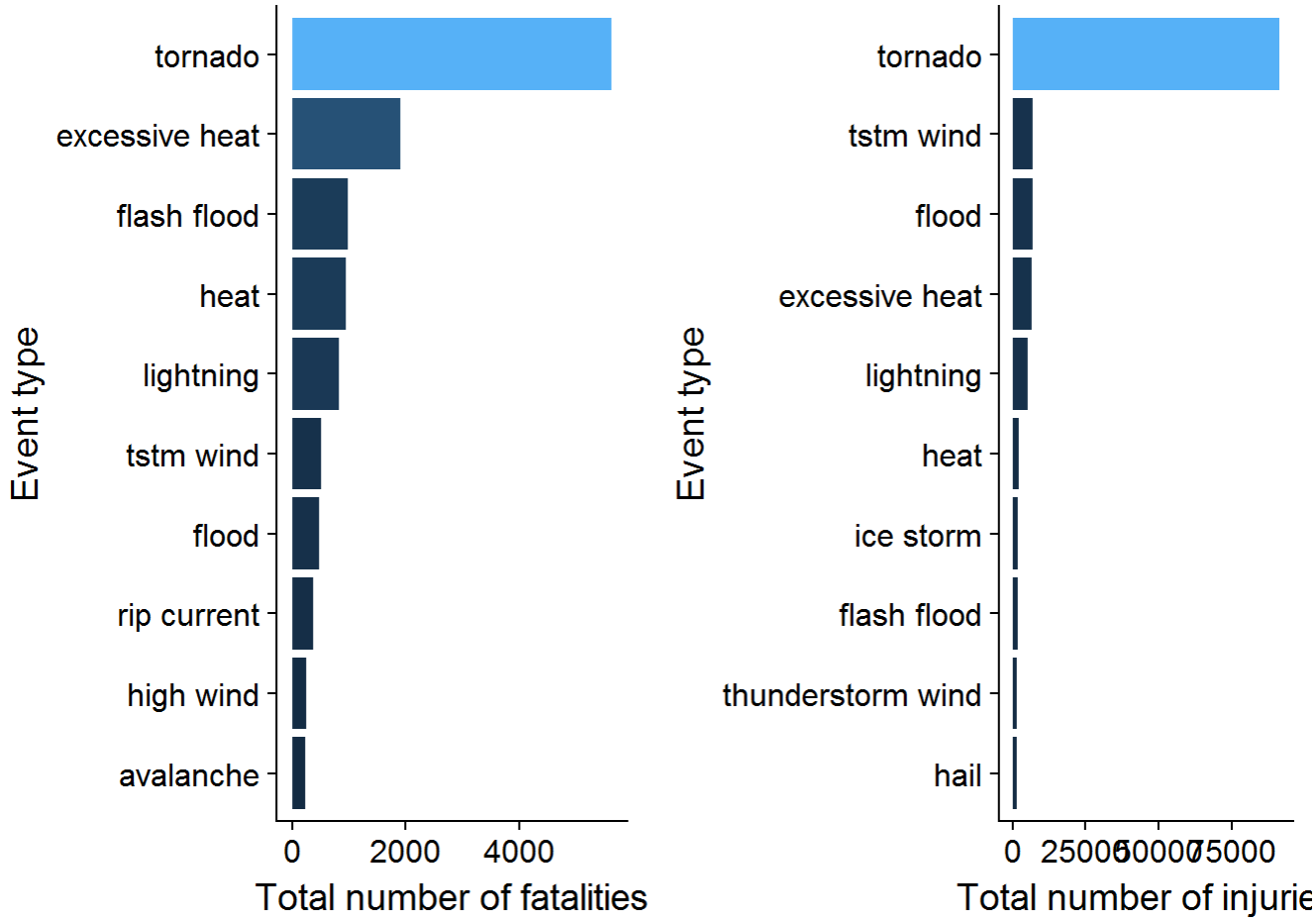
```
  xlab("Event type") +
  theme(legend.position="none")


library(cowplot)
```

```
## Warning: package 'cowplot' was built under R version 3.4.1
```

```
##
## Attaching package: 'cowplot'
```

```
## The following object is masked from 'package:ggplot2':
##
##     ggsave
```

```
plot_grid(p1, p2, align='h')
```



Tornadoes, Excessive Heat and Flash Floods are the most dangerous events.

Economic Impact

Plots shows the most damage-prone disasters from 1950s

```
library(ggplot2)

# Set the levels in order
p1 <- ggplot(data=prop_dmg_events,
             aes(x=reorder(EVTYPE, prop_dmg), y=log10(prop_dmg), fill=prop_dmg )) +
  geom_bar(stat="identity") +
  coord_flip() +
  xlab("Event type") +
  ylab("Property damage in dollars (log-scale)") +
  theme(legend.position="none")

p2 <- ggplot(data=crop_dmg_events,
             aes(x=reorder(EVTYPE, crop_dmg), y=crop_dmg, fill=crop_dmg)) +
  geom_bar(stat="identity") +
  coord_flip() +
  xlab("Event type") +
  ylab("Crop damage in dollars") +
  theme(legend.position="none")

library(cowplot)
plot_grid(p1, p2, align='h')
```