Heat, flood and tornados kill and maim the most people in the US, while flood and hurricanes cause the most damage

Phillip Rowe

8/23/2019

Synopsis

1. To help local and national US government officials respond to health and economic risks from weather events, we analyzed the data from the U.S. National Oceanic and Atmospheric Administration's (NOAA) storm database of events from 1951-2011, focusing on three dimensions: time, geographic location, and weather event type.
2. For time, the majority of data was collected between 1992 and 2011; thus, we broke our analysis into 5 periods of 4 years each, to smooth for weather events which had large impact but did not occur annually.

3. We believe a 4-year period is also probably more suitable to the budgeting and planning schedules of government officials than a longer period (such as a decade).
4. Geographic analysis was limited to four time zones: PST, MST, CST, EST.
5. Weather events were cleaned up (i.e., recategorized) and then filtered to include only twelve types of events, each of which accounted for more than 2% of at least one of the impacts (fatalities, injuries, property damage, or crop damage).
6. From this analysis, we found that 5 types of weather (heat, flood, wind, tornadoes, and lightning) accounted for more than 75% of fatalities and injuries in the 20 years analyzed, while flood and hurricanes were the main contributors to property damage.
7. Flood and drought were the main causes of crop damage.

8. In addition, a vast majority of fatalities and crop damage and injuries were concentrated in the Eastern Standard and Central Standard time zones, while the most property damage was suffered in the Central Standard and Pacific Standard time zones.
9. The Mountain Standard time zone, which was the least impacted across all risks, likely due to lower population density, had a large portion of property damage due to hail, unlike other regions.

Data Processing

Contents

1. **Load the data and pre-process data:**
2. **Analyze and limit data by cutoff year and timezone:**
3. **Clean/recategorize weather event types:**
4. **Melt and recast data:**

5. **Time period analysis:**
6. **Geographic (time zone) analysis:**

1. **Load the data and pre-process data:** filter the variables to be those relevant for our analysis, including formatting the bgn_date variable as a date, cleaning damage coding, and adding new variables for **year** and damage in millions of dollars (**propertydamage_M** and **cropdamage_M**).

```
library("dplyr")
library(lubridate)
library(ggplot2)
library(downloader)
library('stringr')
library('tibble')
library('reshape2')

#-----------------------------------------------------------------------------
-----------
#
# DATA PROCESSING
# 1. **Load the data and pre-process data:**
#
#-----------------------------------------------------------------------------
-----------

# uncomment following two lines for initial download
#url<-
'https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2FStormData.csv.bz2'
#sdownload(url,dest="storm_data.csv.bz2")

sdata<-read.csv('storm_data.csv.bz2')

FALSE Warning in scan(file = file, what = what, sep = sep, quote = quote, dec
=
FALSE dec, : EOF within quoted string

# Fields relevant for health: FATALITIES,INJURIES,
# Fields relevant for economic impact: PROPDMG,PROPDMGEXP,CROPDMG,CROPDMGEXP
# Fields relating to geography: STATE__,TIME_ZONE,STATE
# Other fields: EVTYPE,BGN_DATE
cols<-c('EVTYPE','BGN_DATE','FATALITIES','INJURIES','PROPDMG','PROPDMGEXP',
        'CROPDMG','CROPDMGEXP','TIME_ZONE','STATE__','STATE')

mydf<-select(sdata,cols)
names(mydf)<-tolower(names(mydf))
mydf$evtype<-tolower(mydf$evtype)
mydf$bgn_date<-gsub("\\s0:00:00?","",mydf$bgn_date)
mydf$bgn_date<-as.Date(mydf$bgn_date,"%m/%d/%Y")
```

```r
# remove rows where the coding for property damage is unclear; there are only 14 of these rows
# and they do not include any fatalities or injuries, so the impact on our analysis
# is likely low
prep_propdmgexp<-function(mydf) {
     mydf<-filter(mydf,!(propdmgexp%in%c("?","-","+")))
     mydf<-filter(mydf,!(cropdmgexp%in%c("?")))
     # change coding to be power of ten
     mydf$propdmgexp<-tolower(mydf$propdmgexp)
     mydf$propdmgexp[grep("k",mydf$propdmgexp)]<-3
     mydf$propdmgexp[grep("h",mydf$propdmgexp)]<-2
     mydf$propdmgexp[grep("b",mydf$propdmgexp)]<-9
     mydf$propdmgexp[grep("m",mydf$propdmgexp)]<-6
     mydf$propdmgexp<-as.numeric(mydf$propdmgexp)
     mydf$propdmgexp[is.na(mydf$propdmgexp)]<-0
     return(mydf)
}

prep_cropdmgexp<-function(mydf) {
     mydf$cropdmgexp<-tolower(mydf$cropdmgexp)
     mydf$cropdmgexp[grep("k",mydf$cropdmgexp)]<-3
     mydf$cropdmgexp[grep("b",mydf$cropdmgexp)]<-9
     mydf$cropdmgexp[grep("m",mydf$cropdmgexp)]<-6
     mydf$cropdmgexp<-as.numeric(mydf$cropdmgexp)
     mydf$cropdmgexp[is.na(mydf$cropdmgexp)]<-0
     return(mydf)
}

mydf<-prep_cropdmgexp(mydf) %>% prep_propdmgexp

FALSE Warning in prep_cropdmgexp(mydf): NAs introduced by coercion

# Add three new column variables: year, propdamage_M, cropdamage_M
# using cleaned up codes for damage
mydf$year<-(year(mydf$bgn_date))
mydf$propdamage_M<-with(mydf,propdmg*10^(propdmgexp-6))
mydf$cropdamage_M<-with(mydf,cropdmg*10^(cropdmgexp-6))
```

2. **Analyze and limit data by cutoff year and timezone:** we use the aggregate function to total the **propdamage_K**, **cropdamage_K**, **fatalities**, and **injuries** columns by year, and we can easily see by looking at the resulting dataframe **total_by_year** with the `View()` command that the last 19 years have the largest numbers in each risk, when manually sorted, and crop damage appears to have been untracked earlier. The year 1992 was smaller than quantities in other years, but we thought the 20 years from 1992-2011 was a pragmatic time frame to analyze.
   We also used the aggregate function to group the totals along the time zones. We added columns to the **total_by_tz** to see the percent contribution of each time zone to the totals and could see in the ordered dataframe that the top 3 time zones account for

well over 75% of the impact columns. For completeness, we included the MST timezone. Thus, we filtered the dataset down to four time zones (EST, CST, PST, MST) and years 1992 and beyond.

```
#-----------------------------------------------------------------------------
-----------
#
# DATA PROCESSING
#
# 2. **Analyze and limit data by cutoff year and timezone:**
#
#-----------------------------------------------------------------------------
-----------

premelt<-melt(data=mydf,id.vars=c("evtype","year","time_zone"),

measure.vars=c("fatalities","injuries","propdamage_M","cropdamage_M"))

# these were used in barplot code shown in the Appendix, not used in report
var<-c('health','health','economic','economic')
names(var)<-c('fatalities','injuries','propdamage_M','cropdamage_M')
premelt$impact<-var[premelt$variable]
premelt$damage<-rep("damage",length(premelt[,1]))

precast_year<-dcast(premelt,year~variable,fun=sum)
# sort and show variables by year; last 20 years are worth focusing on
head(precast_year[order(-precast_year$propdamage_M),],30)

##    year fatalities injuries propdamage_M cropdamage_M
## 57 2006        599     3368   121937.434     3534.239
## 56 2005        469     1834    96789.791     4035.202
## 55 2004        370     2426    25346.599     1452.178
## 44 1993        298     2149    16382.976     5603.604
## 59 2008        488     2703    15568.383     2209.793
## 49 1998        687    11177    11603.796     4507.685
## 46 1995       1489     4480    10697.868     2941.124
## 54 2003        443     2931    10254.548     1143.070
## 52 2001        469     2721    10027.044     1816.728
## 48 1997        601     3800     9558.060     1228.079
## 50 1999        908     5148     8721.227     3532.284
## 47 1996        542     2717     6086.815     1888.531
## 58 2007        421     2191     5788.934     1691.152
## 51 2000        477     2803     5621.428     3329.171
## 53 2002        498     3155     4100.882     1410.368
## 45 1994        344     4161     3778.211     5806.735
## 60 2009        173      766     3410.314      185.729
## 31 1980         28     1157     2152.814        0.000
## 35 1984        160     2858     2131.171        0.000
## 24 1973         89     2406     2063.067        0.000
## 25 1974        366     6824     1943.518        0.000
```

```
## 16 1965        301      5197     1762.499        0.000
## 40 1989         79      1675     1760.051        0.000
## 36 1985        112      1513     1608.097        0.000
## 41 1990         95      1825     1560.568        0.000
## 33 1982         64      1276     1429.218        0.000
## 43 1992         54      1754     1406.689        0.000
## 42 1991         73      1355     1152.296        0.000
## 39 1988         55      1030     1124.215        0.000
## 37 1986         51       915     1034.912        0.000
```

```
precast_tz<-dcast(premelt,time_zone~variable,fun=sum)
# sort and show variables by time zone; obviously 4 stand out
head(precast_tz[order(-precast_tz$propdamage_M),],10)
```

```
##    time_zone fatalities injuries propdamage_M cropdamage_M
## 7        CST       8365    99145 193117.69957  31075.00315
## 18       PST        820     4256 125686.73864   4075.19020
## 10       EST       3415    22386  67053.82827   9644.18510
## 16       MST        659     3511   6042.41989    558.13155
## 3        AST        169      147   2455.40262    616.63800
## 14       HST         42       83    827.50725      7.87663
## 20       SST         57      428    423.60700    107.43300
## 21       UNK          0        0    250.30053      0.00000
## 8        EDT          9      171    250.19925     10.29600
## 2        AKS          5        2     85.54291      0.00000
```

```
main_zones<-c('PST','EST','MST','CST')
data_recent_fourtz<-filter(mydf,year>=1992 & time_zone%in%main_zones)
```

3. **Clean/recategorize weather event types:** There were 985 unique event types in the original **sdata** dataframe, which was loaded raw without any filtering. After filtering the data down to the last 20 years and to four main time zones, there were still 820 unique event types. Through an iterative process of adding text recognition phrases and running them in the **prep_evtype()** function and looking at the largest contributing event types in the **total_by_type** aggregated dataframe, we reduced the unique event types to 313 before calculating which were the most significant event types. We chose the all events that accounted for more than 2% of the total, for each of the four categories of impact. We then did a union operation of all these factors to come up with 12 unique weather events, and then filtered the data to only include these rows.

```
#-----------------------------------------------------------------------------
-----------
#
# DATA PROCESSING
#
# 3. **Clean/recategorize weather event types:**
# we analyze a bit more to find the top event types per risk, which is the
hardest
# task given the messy recordkeeping
#-----------------------------------------------------------------------------
```

```
-----------

length(unique(sdata$EVTYPE))

## [1] 985

length(unique(data_recent_fourtz$evtype))

## [1] 820

# Run this first and search for poorly categorized/ redundant weather events
total_by_type_dirty<-
aggregate(cbind(propdamage_M,cropdamage_M,injuries,fatalities) ~ evtype,
                         data_recent_fourtz, sum)
head(total_by_type_dirty[order(-total_by_type_dirty$propdamage_M),],20)

##                        evtype propdamage_M cropdamage_M injuries
## 144                     flood   133240.158    4329.6124     6639
## 332         hurricane/typhoon    69033.100    2603.5008      922
## 539               storm surge    43298.589       0.0050       36
## 694                   tornado    16657.123     352.8632    17653
## 130               flash flood    13912.503    1106.9041     1533
## 200                      hail    11192.120    2497.9305      934
## 323                 hurricane     9898.998    2179.9300       44
## 707            tropical storm     7451.568     544.9950      337
## 810              winter storm     6606.942      26.3740     1316
## 475               river flood     5108.942    5029.4590        2
## 291                 high wind     5037.647     591.1793     1073
## 540         storm surge/tide     4598.446       0.0000        5
## 714                 tstm wind     4469.486     553.2373     3953
## 346                 ice storm     3930.391    5022.0235     1940
## 799                  wildfire     3858.896     281.3608      737
## 330            hurricane opal     3172.846      19.0000        1
## 797          wild/forest fire     2988.358     105.7802      531
## 623          thunderstorm wind     2810.443     238.3321      732
## 248 heavy rain/severe weather     2500.000       0.0000        0
## 649        thunderstorm winds     1749.662     184.9308      786
##     fatalities
## 144        351
## 332         63
## 539         13
## 694       1027
## 130        786
## 200         10
## 323         40
## 707         46
## 810        197
## 475          2
## 291        231
## 540         11
## 714        255
```

```
## 346         84
## 799         67
## 330          1
## 797         12
## 623         60
## 248          0
## 649         59
```

```
prep_evtype<-function(mydf) {
        # consolidate categories of weather
        mydf$evtype[grep("+tornado+",mydf$evtype)]<-"tornado"
        mydf$evtype[grep("+heavy rain+",mydf$evtype)]<-"heavy rain"
        mydf$evtype[grep("+heat+",mydf$evtype)]<-"heat"
        mydf$evtype[grep("+flood+",mydf$evtype)]<-"flood"
        mydf$evtype[grep("+rip current+",mydf$evtype)]<-"rip current"

        mydf$evtype[grep("+wind+",mydf$evtype)]<-"wind"
        mydf$evtype[grep("+cold+",mydf$evtype)]<-"cold/freeze"
        mydf$evtype[grep("+hurricane+",mydf$evtype)]<-"hurricane/tropical
storm"
        mydf$evtype[grep("+typhoon+",mydf$evtype)]<-"hurricane/tropical
storm"
        mydf$evtype[grep("+tropical storm+",mydf$evtype)]<-
"hurricane/tropical storm"

        mydf$evtype[grep("+snow+",mydf$evtype)]<-"snow"
        mydf$evtype[grep("+blizzard+",mydf$evtype)]<-"snow"
        mydf$evtype[grep("+ice storm+",mydf$evtype)]<-"snow"
        mydf$evtype[grep("+winter storm+",mydf$evtype)]<-"snow"
        mydf$evtype[grep("+freeze+",mydf$evtype)]<-"cold/freeze"
        mydf$evtype[grep("+frost+",mydf$evtype)]<-"cold/freeze"

        mydf$evtype[grep("+fire+",mydf$evtype)]<-"fire"
        mydf$evtype[grep("+high surf+",mydf$evtype)]<-"storm surge/high surf"
        mydf$evtype[grep("+storm surge+",mydf$evtype)]<-"storm surge/high
surf"
        mydf$evtype[grep("+hail+",mydf$evtype)]<-"hail"
        return(mydf)
}

df_type_clean<-prep_evtype(data_recent_fourtz)

total_by_type<-aggregate(cbind(propdamage_M,cropdamage_M,injuries,fatalities)
~ evtype,
                          df_type_clean, sum)
head(total_by_type[order(-total_by_type$propdamage_M),],30)
```

```
##                         evtype propdamage_M cropdamage_M injuries fatalities
## 59                       flood 153745.86920  10683.08210     8208       1204
## 97   hurricane/tropical storm  89876.58696   5501.08580     1353        165
```

```
## 184    storm surge/high surf  47911.19900       0.00500       138           81
## 259                  tornado  18313.06040     355.37047     17714         1055
## 306                     wind  15169.93625    1768.78454      7389          986
## 182                     snow  12152.14266    5295.59060      5158          526
## 79                      hail  11433.90071    2518.86347       944           10
## 57                      fire   7571.78323     388.15300      1420           82
## 85                heavy rain   3198.57006     764.17880       234           82
## 173      severe thunderstorm   1205.05000       0.00000         0            0
## 32                   drought   1040.99200   13935.09400         4            0
## 116                lightning    792.35906      11.49859      4749          739
## 110                landslide    253.52350      20.00000        47           35
## 23               cold/freeze    135.58040    3035.52650       282          230
## 286     urban/sml stream fld     57.81615       8.48810        73           28
## 81                      heat     19.09255     904.36928      8360         2937
## 308           winter weather     13.44770       0.00000       338           22
## 60                       fog     13.15550       0.00000       734           62
## 102                      ice     12.65500       0.00000       137            6
## 8     astronomical high tide      9.42500       0.00000         0            0
## 267                  tsunami      9.20300       0.00000         0            0
## 64             freezing rain      8.14650       0.00000        23            7
## 28                 dense fog      7.80700       0.00000       342           18
## 37             dry microburst      6.73260       0.01500        28            3
## 310       winter weather/mix      6.37200       0.00000        72           28
## 148                     rain      5.30000       0.25000         0            0
## 296               waterspout      4.23720       0.00000        29            3
## 48                dust storm      4.12100       3.10000       422           18
## 11                 avalanche      2.59600       0.00000       125          159
## 63              freezing fog      2.00000       0.00000         0            0
```

```
length(unique(df_type_clean$evtype))
```

```
## [1] 313
```

```
# now down to 313 unique event types
# let's see how many of these account for most of results

# - - - - - - - - - - - - - - -

total_by_type$pctprop<-with(total_by_type,
round(propdamage_M/sum(propdamage_M),2))
total_by_type$pctcrop<-with(total_by_type,
round(cropdamage_M/sum(cropdamage_M),2))
total_by_type$pctfat<-with(total_by_type,
round(fatalities/sum(fatalities),2))
total_by_type$pctinj<-with(total_by_type, round(injuries/sum(injuries),2))

crop_events<-filter(total_by_type,total_by_type$pctcrop>0.02)$evtype
prop_events<-filter(total_by_type,total_by_type$pctprop>0.02)$evtype
fat_events<-filter(total_by_type,total_by_type$pctfat>0.02)$evtype
inj_events<-filter(total_by_type,total_by_type$pctinj>0.02)$evtype
```

```
econ_risks<-union(fat_events,inj_events)
health_risks<-union(crop_events,prop_events)
risks<-union(econ_risks,health_risks)
risks

##  [1] "cold/freeze"              "flood"
##  [3] "heat"                     "lightning"
##  [5] "rip current"              "snow"
##  [7] "tornado"                  "wind"
##  [9] "drought"                  "hail"
## [11] "hurricane/tropical storm" "storm surge/high surf"

# filtering data down to these 12 risks

final_data<-filter(df_type_clean,evtype%in%risks)
length(unique(final_data$evtype))

## [1] 12
```

4. **Melt and recast data:**

At this point, it appeared prudent to break our 20 year period into five four-year periods, so as to average out spikes in the data from less regular weather events. We added a column **period**. annual spikes in the weather we still did not have a strictly tidy dataset, because the four important observation variables (fatalities, injuries, property damage, crop damage) were located in columns. Thus, we melted the data so that only one variable appeared per line, added a column specifying whether the observation was health or economic,and defined another column as **damage** so that body counts and dollar impact could be plotted along the same x-axis.

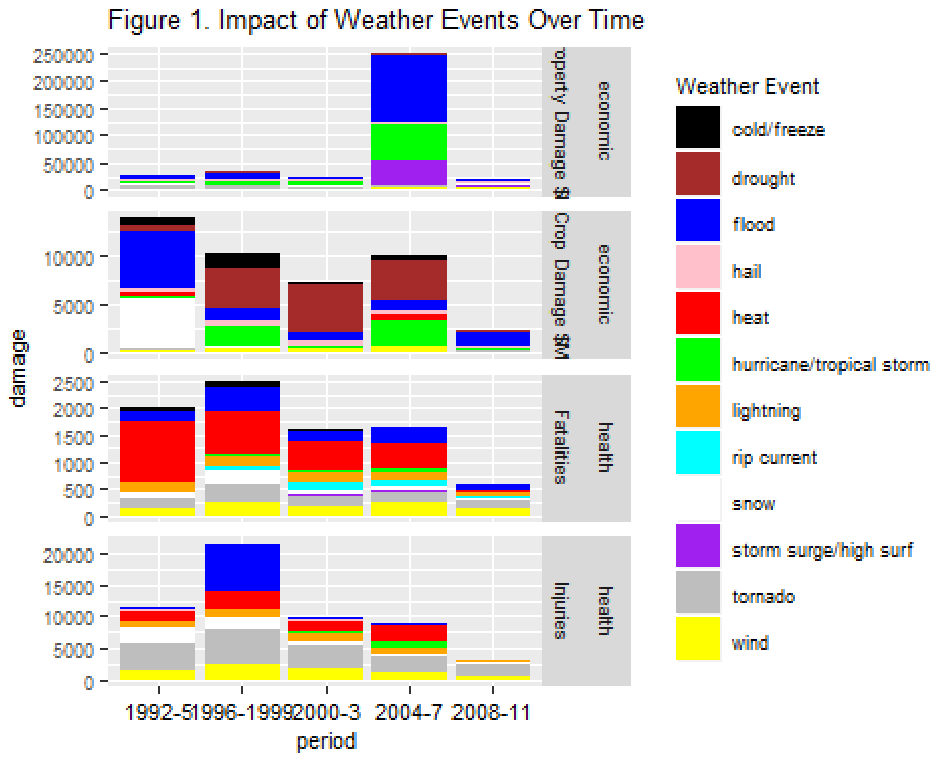
```
#-----------------------------------------------------------------------------
-----------
#
# DATA PROCESSING
# 4. **Melt and recasting data:**
#
#-----------------------------------------------------------------------------
-----------

myperiod<-c(rep('1992-5',4),rep('1996-1999',4),rep('2000-3',4),rep('2004-
7',4),rep('2008-11',4))
names(myperiod)<-c(1992:2011)

final_data$period<-myperiod[as.character(final_data$year)]


# first we want to melt the df into tidy date, with one observation per row
# and introduce the concept of health and economic cost
# health = of type fatalities or injuries and measured by body count
```

```r
# economic = of type property or crop damage and measured by USD Millions
# so we will have a long skinny dataframe where we have a year, timezone,
type of event,
# and only one type of the four costs above measured

melted<-melt(data=final_data,id.vars=c("evtype","period","time_zone"),

measure.vars=c("fatalities","injuries","propdamage_M","cropdamage_M"))

var<-c('health','health','economic','economic')
names(var)<-c('fatalities','injuries','propdamage_M','cropdamage_M')

melted$impact<-var[melted$variable]
melted$damage<-rep("damage",length(melted[,1]))

casted<-dcast(melted,evtype+time_zone+period+variable+impact~damage,fun=sum)
```

5. **Time period analysis:** We used the **ggplot/ geom_bar()** plots with free y-axis scales to plot the damage in 4 four barplots (four impact variables) over the five time periods (see Figure 1). We customized the color palette of fill colors so that each weather event was matched to a unique color that, when possible, made intuitive sense (e.g., flood was blue, snow was white). We set customized the y-labels to be more aesthetically pleasing.

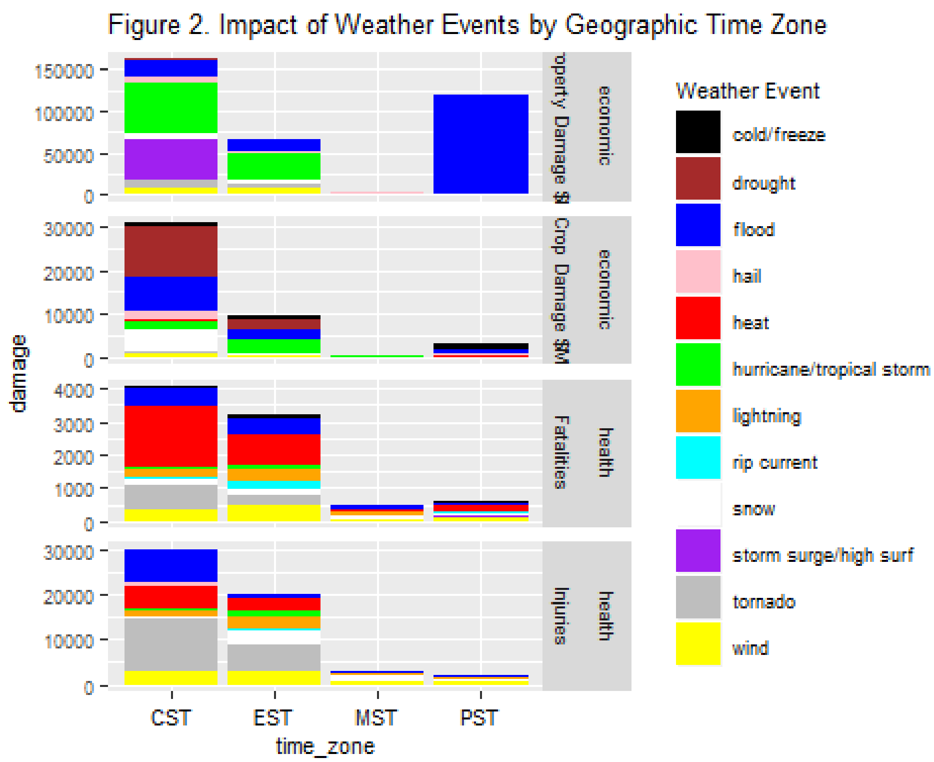
```r
#------------------------------------------------------------------------------
-----------
#
# DATA PROCESSING
# 4. Time period plot
#
# Now we set up 5 periods of 4 years each, covering the 20 years from 1992-
2011
# the named list myperiod acts as a dictionary for plotting functions for
mapping year to period
#------------------------------------------------------------------------------
-----------

prop_total = sum(subset(casted,variable=="propdamage_M")$damage)
crop_total = sum(subset(casted,variable=="cropdamage_M")$damage)
fat_total = sum(subset(casted,variable=="fatalities")$damage)
inj_total = sum(subset(casted,variable=="injuries")$damage)
print(paste("Property damage $M total 1992-2011 from 12 main events:",
prop_total))

## [1] "Property damage $M total 1992-2011 from 12 main events:
350590.81919571"

print(paste("Property damage $M total 1992-2011 from 12 main events:",
crop_total))

## [1] "Property damage $M total 1992-2011 from 12 main events: 44009.270351"
```

```r
print(paste('Property damage is ~',round(prop_total/crop_total,0),'x
larger'))

## [1] "Property damage is ~ 8 x larger"

print(paste("Total fatalities from 1992-2011 from 12 main events:",
fat_total))

## [1] "Total fatalities from 1992-2011 from 12 main events: 8357"

print(paste("Total injuries from 1992-2011 from 12 main events:", inj_total))

## [1] "Total injuries from 1992-2011 from 12 main events: 54690"

print(paste('Injury total is ~',round(inj_total/fat_total,0),'x larger'))

## [1] "Injury total is ~ 7 x larger"

mycolors= c('black','brown','blue', 'pink','red','green', 'orange','cyan',
            'white','purple','gray','yellow')

damage.labs<-c("Property Damage $M","Crop Damage $M","Fatalities","Injuries")
names(damage.labs)<-c("propdamage_M","cropdamage_M","fatalities","injuries")

theme_update(text = element_text(size=8))
 # for the main title

time_period_plot<-
ggplot(casted,aes(period,y=damage,fill=evtype,order=variable)) +
        geom_bar(stat='identity') + facet_grid(impact+variable
~.,scales="free_y",

labeller=labeller(variable=damage.labs))+
        theme(axis.text.x=element_text(size=8,colour='black',angle=0))+
        ggtitle('Figure 1. Impact of Weather Events Over Time')+
        labs(fill="Weather Event", scale_fill_manual(values = mycolors))
time_period_plot+ scale_fill_manual(values = mycolors)
```



6. **Geographic (time zone) analysis:** Because our analysis is directed to national and local government decisionmakers, it made sense to also include a geographic analysis of the data across the four impact variables. We used the stacked barplots with similar coloration scheme in Figure 2.

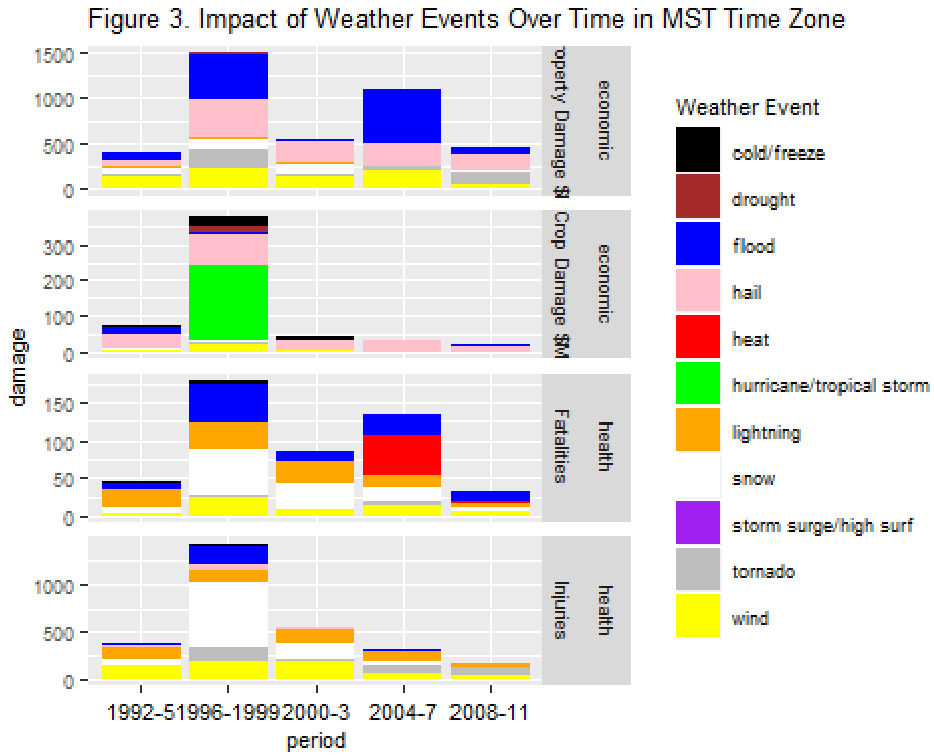
```
#-----------------------------------------------------------------------------
-----------
#
# DATA PROCESSING
# 5. Geographic plot (by time zone)
#
#-----------------------------------------------------------------------------
-----------

# TIME ZONE BY IMPACT
time_zone_plot<-
ggplot(casted,aes(time_zone,y=damage,fill=evtype,order=variable))+
        geom_bar(stat='identity')+
        facet_grid(impact+variable ~.,scales="free_y",
                   labeller=labeller(variable=damage.labs))+
        theme(axis.text.x=element_text(size=8,colour='black',angle=0))+
        ggtitle('Figure 2. Impact of Weather Events by Geographic Time
Zone')+
        labs(fill="Weather Event", scale_fill_manual(values = mycolors))
theme_update(text = element_text(size=8))

time_zone_plot+ scale_fill_manual(values = mycolors)
```



Because most damage in the US from weather occurs in the coastal and central time zones, the scale on Figures 1 and 2 obscure the damage that occurs in the Mountain Standard Time zone. Thus, we provide Figure 3 to inform local authorities in that time zone.

```
theme_update(text = element_text(size=8))

mycolors= c('black','brown','blue', 'pink','red','green', 'orange',
             'white','purple','gray','yellow')

damage.labs<-c("Property Damage $M","Crop Damage $M","Fatalities","Injuries")
names(damage.labs)<-c("propdamage_M","cropdamage_M","fatalities","injuries")
theme_update(text = element_text(size=8))

mst_plot<-
ggplot(subset(casted,time_zone=='MST'),aes(period,y=damage,fill=evtype,order=
variable)) +
        geom_bar(stat='identity') + facet_grid(impact+variable
~.,scales="free_y",

labeller=labeller(variable=damage.labs))+
        theme(axis.text.x=element_text(size=8,colour='black',angle=0))+
        ggtitle('Figure 3. Impact of Weather Events Over Time in MST Time
Zone')+
        labs(fill="Weather Event", scale_fill_manual(values = mycolors))
```

```r
# to actually change to my customized colors
mst_plot+ scale_fill_manual(values = mycolors)
```



Results

The events most harmful to population health and with greatest economic consequences are as follows:

Heat, flood, tornados were top 3 causes of death and injury, with wind and lightning also consistently big contributors

These five weather types were also the main causes of injury; however, tornados were a bigger component of injuries than heat or flooding. Event types that are most harmful to population health

Fatalities and injuries are the harmful effects to population health. For the time period, event types, and time zones analyzed, there was a total of more than 9,800 fatalities and more than 64,000 injuries. Unlike the concentration in a few event types for economic impact, the health impact was more evenly distributed among event types. In particular, heat, flood, tornados, wind and lightning were consistently the top causes of death, although the deathtoll from tornados grew over time while that from heat declined.

These five weather types were also the main causes of injury; however, tornados were a bigger component of injuries than heat or flooding.

In the MST timezone, much of which is at higher altitudes, wind, snow and lightning are the main risks to population health.

Flood and hurricanes/tropical storms accounted for most of property damage and second and third causes of crop damage,while drought was number 1 cause of crop damage

Economic consequences are measured by property damage and crop damage. As can be seen from the scales of Figures 1 and 2 and the calculated totals for 1992-2011 for the top 12 event types in the top four time zones, property damage in dollar terms was ~$374 billion, which is nearly 8x the cumulative crop damage of under $46 billion.

Flood and hurricanes/tropical storms accounted most of the property damage, concentrated in the 2004-2007 timeframe and in the EST, CST, and PST timezones. Drought was a major problem in the CST and EST time zones for crop damage. In the MST timezone, much of which is at higher altitudes, hail is the main cause of property and crop damage.