Data Analysis of the NOAA Storm Dataset

Jon Kinsey

Thu Jan 1 01:59:13 2015

```
# Here, we explore the NOAA Storm Database and answer some basic questions
# about severe weather events. The NOAA 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. The events in the database start in the year 1950 and end
# in November 2011. The data is downloaded from the National Weather Service.
# See http://www.rpubs.com/mariuszgil/noaa-storm-data
# Here we attempt to answer the following:
#   1. Across the United States, which types
#      of events (as indicated in the EVTYPE variable) are most harmful with respect
#      to population health?
#   2. Across the United States, which types of events have the greatest
#      economic consequences?
#
# Only need 7 variables which are related to these questions:
#
# EVTYPE as a measure of event type (e.g. tornado, flood, etc.)
# FATALITIES as a measure of harm to human health
# INJURIES as a measure of harm to human health
# PROPDMG as a measure of property damage and hence economic damage in USD
# PROPDMGEXP as a measure of magnitude of property damage (e.g. thousands,
#  millions USD, etc.)
# CROPDMG as a measure of crop damage and hence economic damage in USD
# CROPDMGEXP as a measure of magnitude of crop damage (e.g. thousands,
#  millions USD, etc.)
#
# The dataset came from a comma-separated-value file, compressed via the
# bz2 algorithm. File is located in CloudFront CDN service on the URL
# https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2FStormData.csv.bz2.
#
library(xtable)
library(plyr)
#
# load the data
# url <- "https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2FStormData.csv.bz2"
setwd("/Users/Jon/Desktop/R-Projects")
storm <- read.csv("repdata-data-StormData.csv",  header = TRUE)
#
names(storm)
```

```
##  [1] "STATE__"    "BGN_DATE"   "BGN_TIME"   "TIME_ZONE"  "COUNTY"
##  [6] "COUNTYNAME" "STATE"      "EVTYPE"     "BGN_RANGE"  "BGN_AZI"
## [11] "BGN_LOCATI" "END_DATE"   "END_TIME"   "COUNTY_END" "COUNTYENDN"
## [16] "END_RANGE"  "END_AZI"    "END_LOCATI" "LENGTH"     "WIDTH"
## [21] "F"          "MAG"        "FATALITIES" "INJURIES"   "PROPDMG"
## [26] "PROPDMGEXP" "CROPDMG"    "CROPDMGEXP" "WFO"        "STATEOFFIC"
## [31] "ZONENAMES"  "LATITUDE"   "LONGITUDE"  "LATITUDE_E" "LONGITUDE_"
```

```
## [36] "REMARKS"    "REFNUM"
```

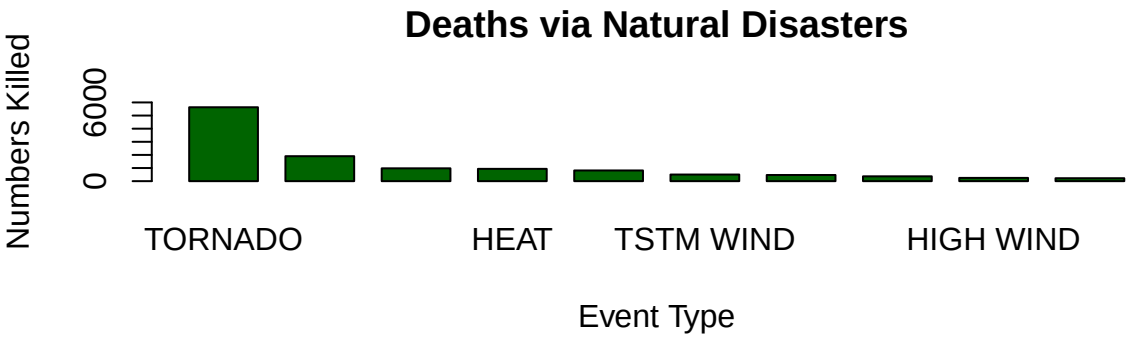
```
my_data <- data.frame(storm$EVTYPE, storm$FATALITIES, storm$INJURIES,
                      storm$PROPDMG, storm$PROPDMGEXP, storm$CROPDMG,
                      storm$CROPDMGEXP)
#
# We first analyze the public health aspect, among which two import variables
# are analyzed, fatalities and those injured. We group all the
# events based on the event type and sort them from mostly deadly to least.
#
# Top 10 fatality/injury event(classed by different type) :
#
cdata <- ddply(my_data, c("storm.EVTYPE"), summarise,
               sum_fatal  = sum(storm.FATALITIES, na.rm=TRUE),
               sum_injury = sum(storm.INJURIES, na.rm=TRUE))
colnames(cdata) = c("event.type", "fatality.total", "injury.total")
cdata.sorted1 <- cdata[order(-cdata$fatality.total),]
top10.fatality <- cdata.sorted1[1:10,]
cdata.sorted2 <- cdata[order(-cdata$injury.total),]
top10.injury <- cdata.sorted2[1:10,]
print(xtable(top10.fatality))
```

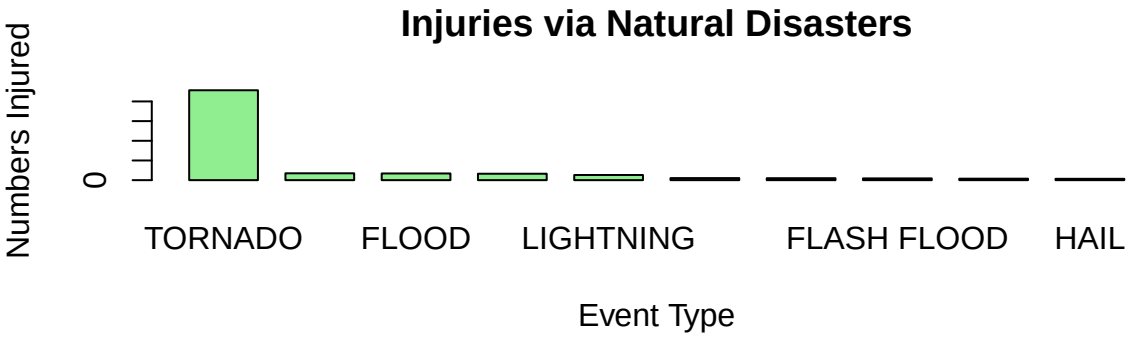
```
## % latex table generated in R 3.1.2 by xtable 1.7-3 package
## % Thu Jan  1 02:01:13 2015
## \begin{table}[ht]
## \centering
## \begin{tabular}{rlrr}
##   \hline
##  & event.type & fatality.total & injury.total \\
##   \hline
## 834 & TORNADO & 5633.00 & 91346.00 \\
##   130 & EXCESSIVE HEAT & 1903.00 & 6525.00 \\
##   153 & FLASH FLOOD & 978.00 & 1777.00 \\
##   275 & HEAT & 937.00 & 2100.00 \\
##   464 & LIGHTNING & 816.00 & 5230.00 \\
##   856 & TSTM WIND & 504.00 & 6957.00 \\
##   170 & FLOOD & 470.00 & 6789.00 \\
##   585 & RIP CURRENT & 368.00 & 232.00 \\
##   359 & HIGH WIND & 248.00 & 1137.00 \\
##   19 & AVALANCHE & 224.00 & 170.00 \\
##    \hline
## \end{tabular}
## \end{table}
```

```
#
# Next, we analyze the data according to natural disasters with the greatest
# economic impact. We combine the property damage as well as the crop damage.
# The trick part here is that they are measured on different scales or magnitudes.
# So we need to transform them onto the same scale. The following
# preprocessing is done. For those with a missing value, we give it 0 and for those
# with - or +, we give them 1 in scale. For the other, we give them according
# to their short form. For example, k to 10 to power 3, m to 10 to power 6 and B
# to 10 to power 9 etc.
```

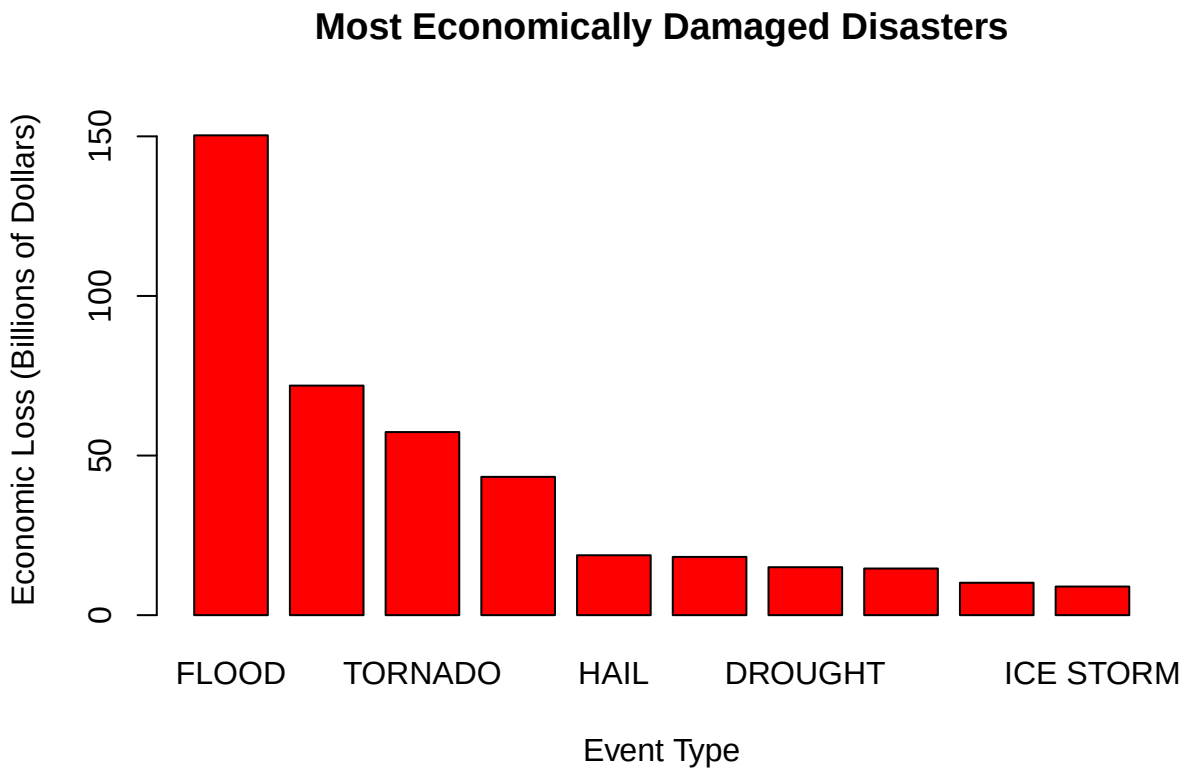
```
#
# Top disasters with high economic loss :
#table(my_data$storm.PROPDMGEXP)
#table(my_data$storm.CROPDMGEXP)
my_data$storm.PROPDMGEXP <- as.character(my_data$storm.PROPDMGEXP)
my_data$storm.CROPDMGEXP <- as.character(my_data$storm.CROPDMGEXP)
#
my_data$storm.PROPDMGEXP[(my_data$storm.PROPDMGEXP == "")] <- 0
my_data$storm.PROPDMGEXP[(my_data$storm.PROPDMGEXP == "+") | (my_data$storm.PROPDMGEXP ==
                                                                "-") | (my_data$storm.PROPDMG
my_data$storm.PROPDMGEXP[(my_data$storm.PROPDMGEXP == "h") | (my_data$storm.PROPDMGEXP ==
                                                                "H")] <- 2
my_data$storm.PROPDMGEXP[(my_data$storm.PROPDMGEXP == "k") | (my_data$storm.PROPDMGEXP ==
                                                                "K")] <- 3
my_data$storm.PROPDMGEXP[(my_data$storm.PROPDMGEXP == "m") | (my_data$storm.PROPDMGEXP ==
                                                                "M")] <- 6
my_data$storm.PROPDMGEXP[(my_data$storm.PROPDMGEXP == "B")] <- 9

my_data$storm.CROPDMGEXP[(my_data$storm.CROPDMGEXP == "")] <- 0
my_data$storm.CROPDMGEXP[(my_data$storm.CROPDMGEXP == "+") | (my_data$storm.CROPDMGEXP ==
                                                                "-") | (my_data$storm.CROPDMG
my_data$storm.CROPDMGEXP[(my_data$storm.CROPDMGEXP == "h") | (my_data$storm.CROPDMGEXP ==
                                                                "H")] <- 2
my_data$storm.CROPDMGEXP[(my_data$storm.CROPDMGEXP == "k") | (my_data$storm.CROPDMGEXP ==
                                                                "K")] <- 3
my_data$storm.CROPDMGEXP[(my_data$storm.CROPDMGEXP == "m") | (my_data$storm.CROPDMGEXP ==
                                                                "M")] <- 6
my_data$storm.CROPDMGEXP[(my_data$storm.CROPDMGEXP == "B")] <- 9
#
# convert to integer for computation
my_data$storm.PROPDMGEXP <- as.integer(my_data$storm.PROPDMGEXP)
my_data$storm.CROPDMGEXP <- as.integer(my_data$storm.CROPDMGEXP)
#
# The same analysis applies here where we group by event type and sort the loss
# from greatest to least.
my_data$damage_total <- my_data$storm.PROPDMG * 10 ^ my_data$storm.PROPDMGEXP +
  my_data$storm.CROPDMG * 10 ^ my_data$storm.CROPDMGEXP
ddata <- ddply(my_data, c("storm.EVTYPE"), summarise,
               sum_damage_total  = sum(damage_total, na.rm=TRUE))
colnames(ddata) = c("event.type", "damage_total")
ddata.sorted <- ddata[order(-ddata$damage_total),]
topddata.damage <- ddata.sorted[1:10,]
#
# Regarding the most deadly and injurying disasters, we make two plots together.
# As we can see, tornado, heat, and flood, lightning are among the top 10.
par(mfrow=c(2,1))
bar1<-barplot(top10.fatality$fatality.total, names.arg = top10.fatality$event.type,
              ylim=c(0,6000),space = 0.4,  xlab="Event Type", ylab="Numbers Killed",
              main="Deaths via Natural Disasters", col="darkgreen")
bar2<-barplot(top10.injury$injury.total, names.arg = top10.injury$event.type,
              ylim=c(0,80000),space = 0.4,  xlab="Event Type", ylab="Numbers Injured",
              main="Injuries via Natural Disasters", col="lightgreen")
```






```
#
# We find that tornadoes are the most harmful event with respect to population health.
#
par(mfrow=c(1,1))
# Plotting the economic loss in billions of dollars we see that
# flood, hurricane/typhoon and then tornado are the top 3 disasters.
bar3<-barplot(height = topddata.damage$damage_total/1.e9, names.arg = topddata.damage$event.type,
              ylim=c(0,160),space = 0.3,  xlab="Event Type", ylab="Economic Loss (Billions of Dollars)"
              main="Most Economically Damaged Disasters", col="red")
#
# Here is another way to do the analysis and plot the results using ggplot2
#
library(ggplot2)
```




```
data <- read.csv("repdata-data-StormData.csv",  header = TRUE)
# Count number of missing values
nmissing <- function(x) sum(is.na(x))
# Apply to every column in a data frame
colwise(nmissing)(data)
```

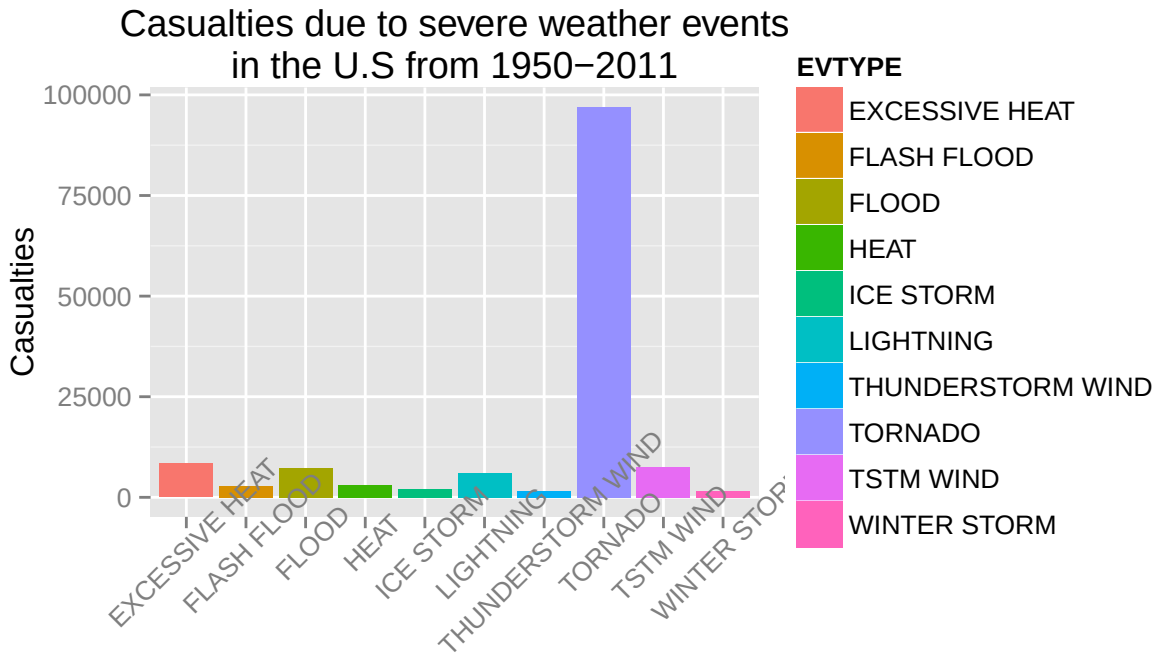
```
##   STATE__ BGN_DATE BGN_TIME TIME_ZONE COUNTY COUNTYNAME STATE EVTYPE
## 1       0        0        0         0      0          0     0      0
##   BGN_RANGE BGN_AZI BGN_LOCATI END_DATE END_TIME COUNTY_END COUNTYENDN
## 1         0       0          0        0        0          0     902297
##   END_RANGE END_AZI END_LOCATI LENGTH WIDTH      F MAG FATALITIES INJURIES
## 1         0       0          0      0     0 843563   0          0        0
##   PROPDMG PROPDMGEXP CROPDMG CROPDMGEXP WFO STATEOFFIC ZONENAMES LATITUDE
## 1       0          0       0          0   0          0         0       47
##   LONGITUDE LATITUDE_E LONGITUDE_ REMARKS REFNUM
## 1         0         40          0       0      0
```

```
#
# Sum the FATALITIES and INJURIES by EVTYPE, and get the top 10 harmful types.
DeathInjury <- ddply(data, .(EVTYPE), summarize, TotalHarm = sum(FATALITIES + INJURIES))
DeathInjury <- DeathInjury[order(DeathInjury$TotalHarm, decreasing = T), ]
TopHarm <- DeathInjury[1:10, ]
#
# Sum the PROPDMG by EVTYPE and PROPDMGEXP. Then calculate real property
# damage by accounting PROPDMGEXP. Finally, sum the new property damage
# data by EVTYPE
prop <- ddply(data, .(EVTYPE, PROPDMGEXP), summarize, PROPDMG = sum(PROPDMG))
prop <- mutate(prop, PropertyDamage = ifelse(toupper(PROPDMGEXP) =='K', PROPDMG*1000, ifelse(toupper(PR
```

```
prop <- subset(prop, select = c("EVTYPE", "PropertyDamage"))
prop.total <- ddply(prop, .(EVTYPE), summarize, TotalPropDamage = sum(PropertyDamage))
#
# Sum the CROPDMG by EVTYPE and CROPDMGEXP. Then calculate real crop damage by
# accounting CROPDMGEXP. Last step, sum the new crop damage data by EVTYPE.
crop <- ddply(data, .(EVTYPE, CROPDMGEXP), summarize, CROPDMG = sum(CROPDMG))
crop <- mutate(crop, CropDamage = ifelse(toupper(CROPDMGEXP) =='K', CROPDMG*1000, ifelse(toupper(CROPDM
crop <- subset(crop, select = c("EVTYPE", "CropDamage"))
crop.total <- ddply(crop, .(EVTYPE), summarize, TotalCropDamage = sum(CropDamage))
#
# Now, merge the property and crop damage data, and select the top ten damage.
damage <- merge(prop.total, crop.total, by="EVTYPE")
damage <- mutate(damage, TotalDamage = TotalPropDamage + TotalCropDamage)
damage <- damage[order(damage$TotalDamage, decreasing = T), ]
TopDamage <- damage[1:10, ]
#
# Here is the result of top 10 harmful type based on the sum of casualties :
TopHarm
```

```
##                EVTYPE TotalHarm
## 834           TORNADO     96979
## 130    EXCESSIVE HEAT      8428
## 856         TSTM WIND      7461
## 170             FLOOD      7259
## 464         LIGHTNING      6046
## 275              HEAT      3037
## 153       FLASH FLOOD      2755
## 427         ICE STORM      2064
## 760 THUNDERSTORM WIND      1621
## 972      WINTER STORM      1527
```

```
p <- qplot(EVTYPE, TotalHarm, data = TopHarm, stat='identity',geom = "bar",
           fill= EVTYPE,xlab="Top 10 weather events",ylab="Casualties",
           main="Casualties due to severe weather events\nin the U.S from 1950-2011")
p + theme(axis.text.x = element_text(angle = 45))
```

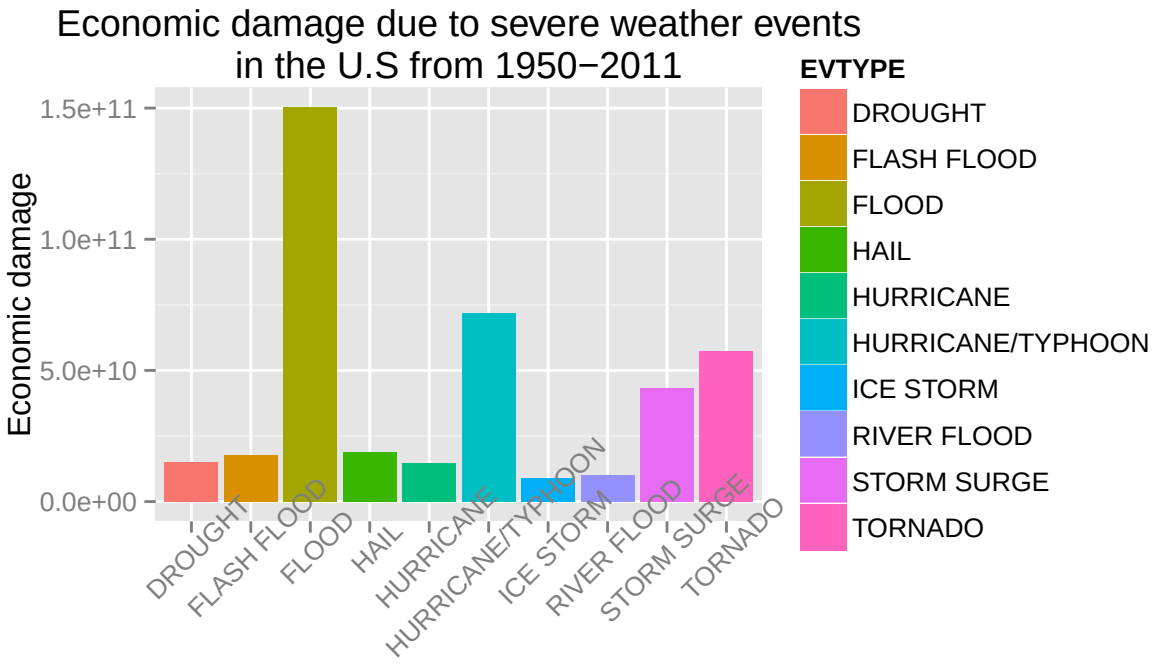


Top 10 weather events

```
#
# Here is the result of top 10 harmful type based on the sum of damages :
TopDamage
```

```
##                EVTYPE TotalPropDamage TotalCropDamage  TotalDamage
## 170             FLOOD    144657709807      5661968450 150319678257
## 411 HURRICANE/TYPHOON     69305840000      2607872800  71913712800
## 834           TORNADO     56937160779       414953270  57352114049
## 670       STORM SURGE     43323536000            5000  43323541000
## 244              HAIL     15732267543      3025954473  18758222016
## 153       FLASH FLOOD     16140812067      1421317100  17562129167
## 95            DROUGHT      1046106000     13972566000  15018672000
## 402         HURRICANE     11868319010      2741910000  14610229010
## 590       RIVER FLOOD      5118945500      5029459000  10148404500
## 427         ICE STORM      3944927860      5022113500   8967041360
```

```
p <- qplot(EVTYPE, TotalDamage, data = TopDamage, stat='identity',geom = "bar",
           fill= EVTYPE,xlab="Top 10 weather events",ylab="Economic damage",
           main="Economic damage due to severe weather events\nin the U.S from 1950-2011")
p + theme(axis.text.x = element_text(angle = 45))
```



Top 10 weather events