Data Processing

The data for this assignment come in the form of a comma-separated-value file compressed via the bzip2 algorithm to reduce its size. You can download the file from the course web site:

Read the data in

```
# first clean the environment and setup the working directory

rm(list= ls())

setwd("c:/repres-assignmet2")


# now download file

if (!file.exists("StormData.csv.bz2")) {

    fileURL <-
'https://d396qusza40orc.cloudfront.net/repdata%2Fdata%2FStormData.csv.bz2'

    download.file(fileURL, destfile='StormData.csv.bz2', method = 'curl')

}

noaaDF <- read.csv(bzfile('StormData.csv.bz2'),header=TRUE, stringsAsFactors
= FALSE)
```

load the various needed packages

```
# laod libraries for tidying - not all will be used in all this weeks
assignment

require(dplyr)
```

```
## Loading required package: 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
```

```
require(tidyr)
```

```
## Loading required package: tidyr
```

```
require(lubridate)
```

```
## Loading required package: lubridate
##
## Attaching package: 'lubridate'
## The following object is masked from 'package:base':
##
##     date
```

```
require(ggplot2)
```

```
## Loading required package: ggplot2
```

preliminary analysis

First a summary of the NU.S. National Oceanic and Atmospheric Administration's (NOAA) storm database:

```
summary(noaaDF)
```

```
##     STATE__        BGN_DATE           BGN_TIME          TIME_ZONE
##  Min.   : 1.0   Length:902297      Length:902297      Length:902297
##  1st Qu.:19.0   Class :character   Class :character   Class :character
##  Median :30.0   Mode  :character   Mode  :character   Mode  :character
##  Mean   :31.2
##  3rd Qu.:45.0
##  Max.   :95.0
##
```

```
##      COUNTY       COUNTYNAME           STATE              EVTYPE
##  Min.   :  0.0   Length:902297      Length:902297      Length:902297
##  1st Qu.: 31.0   Class :character   Class :character   Class :character
##  Median : 75.0   Mode  :character   Mode  :character   Mode  :character
##  Mean   :100.6
##  3rd Qu.:131.0
##  Max.   :873.0
##
##    BGN_RANGE          BGN_AZI           BGN_LOCATI
##  Min.   :   0.000   Length:902297      Length:902297
##  1st Qu.:   0.000   Class :character   Class :character
##  Median :   0.000   Mode  :character   Mode  :character
##  Mean   :   1.484
##  3rd Qu.:   1.000
##  Max.   :3749.000
##
##    END_DATE           END_TIME           COUNTY_END COUNTYENDN
##  Length:902297      Length:902297      Min.   :0    Mode:logical
##  Class :character   Class :character   1st Qu.:0    NA's:902297
##  Mode  :character   Mode  :character   Median :0
##                                        Mean   :0
##                                        3rd Qu.:0
##                                        Max.   :0
##
##    END_RANGE          END_AZI           END_LOCATI
##  Min.   :  0.0000   Length:902297      Length:902297
##  1st Qu.:  0.0000   Class :character   Class :character
##  Median :  0.0000   Mode  :character   Mode  :character
##  Mean   :  0.9862
##  3rd Qu.:  0.0000
##  Max.   :925.0000
##
```

```
##      LENGTH              WIDTH                F               MAG
##  Min.   :   0.0000   Min.   :   0.000   Min.   :0.0      Min.   :    0.0
##  1st Qu.:   0.0000   1st Qu.:   0.000   1st Qu.:0.0      1st Qu.:    0.0
##  Median :   0.0000   Median :   0.000   Median :1.0      Median :   50.0
##  Mean   :   0.2301   Mean   :   7.503   Mean   :0.9      Mean   :   46.9
##  3rd Qu.:   0.0000   3rd Qu.:   0.000   3rd Qu.:1.0      3rd Qu.:   75.0
##  Max.   :2315.0000   Max.   :4400.000   Max.   :5.0      Max.   :22000.0
##                                         NA's   :843563
##    FATALITIES          INJURIES            PROPDMG
##  Min.   :  0.0000   Min.   :   0.0000   Min.   :   0.00
##  1st Qu.:  0.0000   1st Qu.:   0.0000   1st Qu.:   0.00
##  Median :  0.0000   Median :   0.0000   Median :   0.00
##  Mean   :  0.0168   Mean   :   0.1557   Mean   :  12.06
##  3rd Qu.:  0.0000   3rd Qu.:   0.0000   3rd Qu.:   0.50
##  Max.   :583.0000   Max.   :1700.0000   Max.   :5000.00
##
##   PROPDMGEXP           CROPDMG         CROPDMGEXP
##  Length:902297      Min.   :  0.000   Length:902297
##  Class :character   1st Qu.:  0.000   Class :character
##  Mode  :character   Median :  0.000   Mode  :character
##                     Mean   :  1.527
##                     3rd Qu.:  0.000
##                     Max.   :990.000
##
##      WFO              STATEOFFIC         ZONENAMES            LATITUDE
##  Length:902297      Length:902297      Length:902297      Min.   :   0
##  Class :character   Class :character   Class :character   1st Qu.:2802
##  Mode  :character   Mode  :character   Mode  :character   Median :3540
##                                                           Mean   :2875
##                                                           3rd Qu.:4019
##                                                           Max.   :9706
##                                                           NA's   :47
```

```
##    LONGITUDE        LATITUDE_E     LONGITUDE_       REMARKS
##  Min.   :-14451   Min.   :   0   Min.   :-14455   Length:902297
##  1st Qu.:  7247   1st Qu.:   0   1st Qu.:     0   Class :character
##  Median :  8707   Median :   0   Median :     0   Mode  :character
##  Mean   :  6940   Mean   :1452   Mean   :  3509
##  3rd Qu.:  9605   3rd Qu.:3549   3rd Qu.:  8735
##  Max.   : 17124   Max.   :9706   Max.   :106220
##                   NA's   :40
##      REFNUM
##  Min.   :     1
##  1st Qu.:225575
##  Median :451149
##  Mean   :451149
##  3rd Qu.:676723
##  Max.   :902297
##
```

Next the structure of the Data Frame:

```
str(noaaDF)
```

```
## 'data.frame':    902297 obs. of  37 variables:
##  $ STATE__   : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ BGN_DATE  : chr  "4/18/1950 0:00:00" "4/18/1950 0:00:00" "2/20/1951
0:00:00" "6/8/1951 0:00:00" ...
##  $ BGN_TIME  : chr  "0130" "0145" "1600" "0900" ...
##  $ TIME_ZONE : chr  "CST" "CST" "CST" "CST" ...
##  $ COUNTY    : num  97 3 57 89 43 77 9 123 125 57 ...
##  $ COUNTYNAME: chr  "MOBILE" "BALDWIN" "FAYETTE" "MADISON" ...
##  $ STATE     : chr  "AL" "AL" "AL" "AL" ...
##  $ EVTYPE    : chr  "TORNADO" "TORNADO" "TORNADO" "TORNADO" ...
##  $ BGN_RANGE : num  0 0 0 0 0 0 0 0 0 0 ...
```

```
##  $ BGN_AZI   : chr  "" "" "" "" ...
##  $ BGN_LOCATI: chr  "" "" "" "" ...
##  $ END_DATE  : chr  "" "" "" "" ...
##  $ END_TIME  : chr  "" "" "" "" ...
##  $ 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   : chr  "" "" "" "" ...
##  $ END_LOCATI: chr  "" "" "" "" ...
##  $ 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: chr  "K" "K" "K" "K" ...
##  $ CROPDMG   : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ CROPDMGEXP: chr  "" "" "" "" ...
##  $ WFO       : chr  "" "" "" "" ...
##  $ STATEOFFIC: chr  "" "" "" "" ...
##  $ ZONENAMES : chr  "" "" "" "" ...
##  $ 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   : chr  "" "" "" "" ...
##  $ REFNUM    : num  1 2 3 4 5 6 7 8 9 10 ...
```

Results

1: address the question of which types of events are most harmful to population health

Calculate the fatalities and injuries seperately

The fatalities:

```
totFatalities <- aggregate(noaaDF$FATALITIES, by = list(noaaDF$EVTYPE),
"sum")

names(totFatalities) <- c("Event", "Fatalities")

totFatalitiesSorted <- totFatalities[order(-totFatalities$Fatalities),
][1:20, ]

totFatalitiesSorted
```

```
##                       Event Fatalities
## 834                 TORNADO       5633
## 130          EXCESSIVE HEAT       1903
## 153             FLASH FLOOD        978
## 275                    HEAT        937
## 464               LIGHTNING        816
## 856               TSTM WIND        504
## 170                   FLOOD        470
## 585             RIP CURRENT        368
## 359               HIGH WIND        248
## 19                AVALANCHE        224
## 972            WINTER STORM        206
## 586            RIP CURRENTS        204
## 278               HEAT WAVE        172
## 140            EXTREME COLD        160
## 760       THUNDERSTORM WIND        133
## 310              HEAVY SNOW        127
## 141 EXTREME COLD/WIND CHILL        125
```

```
## 676             STRONG WIND        103
## 30                 BLIZZARD        101
## 350               HIGH SURF        101
```

The injuries:

```
totInjuries <- aggregate(noaaDF$INJURIES, by = list(noaaDF$EVTYPE), "sum")
names(totInjuries) <- c("Event", "Injuries")
totInjuriesSorted <- totInjuries[order(-totInjuries$Injuries), ][1:20, ]
totInjuriesSorted
##                  Event Injuries
## 834            TORNADO    91346
## 856          TSTM WIND     6957
## 170              FLOOD     6789
## 130     EXCESSIVE HEAT     6525
## 464          LIGHTNING     5230
## 275               HEAT     2100
## 427          ICE STORM     1975
## 153        FLASH FLOOD     1777
## 760  THUNDERSTORM WIND     1488
## 244               HAIL     1361
## 972       WINTER STORM     1321
## 411  HURRICANE/TYPHOON     1275
## 359          HIGH WIND     1137
## 310         HEAVY SNOW     1021
## 957           WILDFIRE      911
## 786 THUNDERSTORM WINDS      908
## 30            BLIZZARD      805
## 188                FOG      734
## 955   WILD/FOREST FIRE      545
```

```
## 117          DUST STORM      440
```

Finally plot both the fatalities and injuries in a single plot:

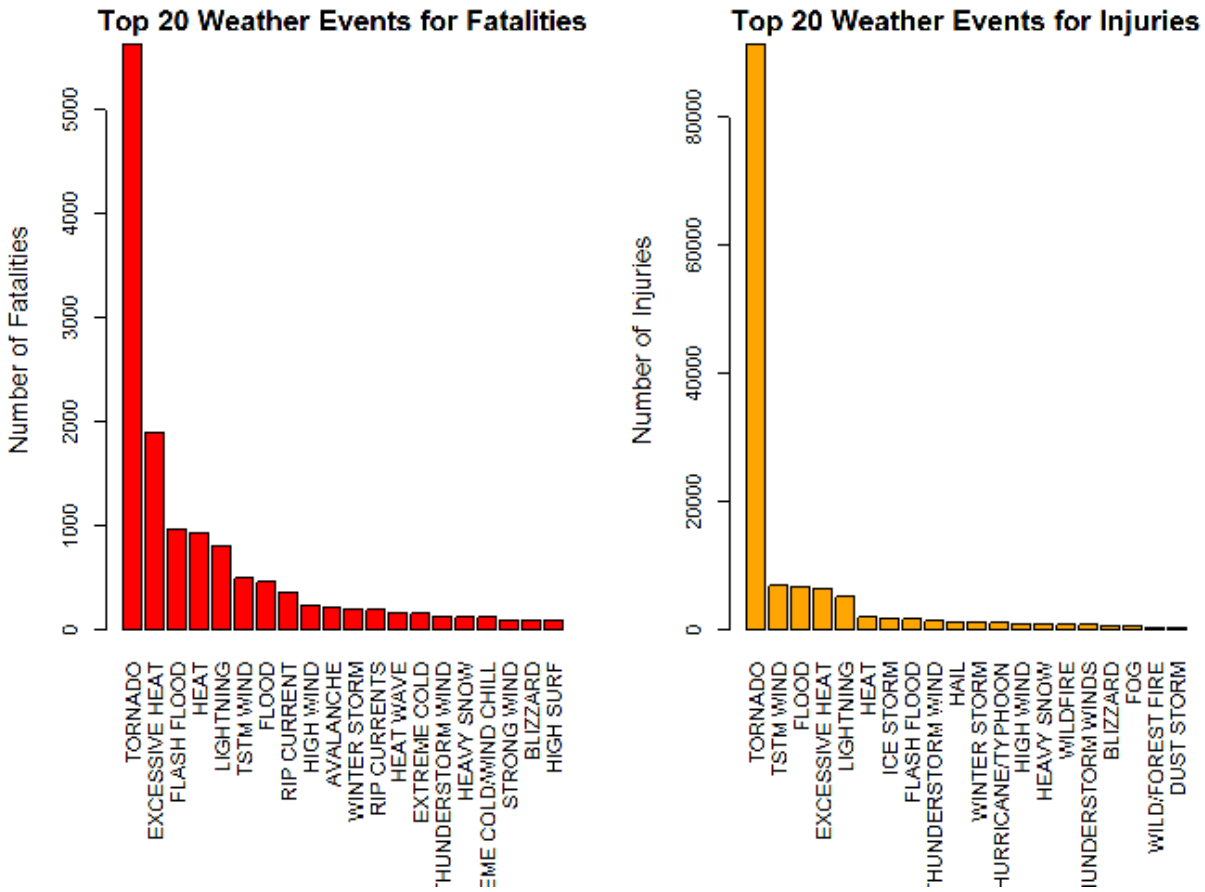
```
par(mfrow = c(1, 2), mar = c(10, 4, 2, 2), las = 3, cex = 0.7, cex.main =
1.4, cex.lab = 1.2)

barplot(totFatalitiesSorted$Fatalities, names.arg =
totFatalitiesSorted$Event, col = 'red',

        main = 'Top 20 Weather Events for Fatalities', ylab = 'Number of
Fatalities')

barplot(totInjuriesSorted$Injuries, names.arg = totInjuriesSorted$Event, col
= 'orange',

        main = 'Top 20 Weather Events for Injuries', ylab = 'Number of
Injuries')
```



address the question of which types of events have the greatest economic consequences

Calculate the cost of property and crop damages seperately

The property:

```
totProperty <- aggregate(noaaDF$PROPDMG, by = list(noaaDF$EVTYPE), "sum")
names(totProperty) <- c("Event", "Property")
totPropertySorted <- totProperty[order(-totProperty$Property), ][1:20, ]
totPropertySorted
```

```
##                    Event   Property
## 834              TORNADO 3212258.16
## 153          FLASH FLOOD 1420124.59
## 856            TSTM WIND 1335965.61
## 170                FLOOD  899938.48
## 760    THUNDERSTORM WIND  876844.17
## 244                 HAIL  688693.38
## 464            LIGHTNING  603351.78
## 786   THUNDERSTORM WINDS  446293.18
## 359            HIGH WIND  324731.56
## 972         WINTER STORM  132720.59
## 310           HEAVY SNOW  122251.99
## 957             WILDFIRE   84459.34
## 427            ICE STORM   66000.67
## 676          STRONG WIND   62993.81
## 376           HIGH WINDS   55625.00
```

```
## 290           HEAVY RAIN   50842.14
## 848       TROPICAL STORM   48423.68
## 955     WILD/FOREST FIRE   39344.95
## 164       FLASH FLOODING   28497.15
## 919 URBAN/SML STREAM FLD   26051.94
```

The crop:

```
totCrop <- aggregate(noaaDF$CROPDMG, by = list(noaaDF$EVTYPE), "sum")
names(totCrop) <- c("Event", "Crop")
totCropSorted <- totCrop[order(-totCrop$Crop), ][1:20, ]
totCropSorted
```

```
##                  Event      Crop
## 244               HAIL 579596.28
## 153        FLASH FLOOD 179200.46
## 170              FLOOD 168037.88
## 856          TSTM WIND 109202.60
## 834            TORNADO 100018.52
## 760  THUNDERSTORM WIND  66791.45
## 95             DROUGHT  33898.62
## 786 THUNDERSTORM WINDS  18684.93
## 359          HIGH WIND  17283.21
## 290         HEAVY RAIN  11122.80
## 212       FROST/FREEZE   7034.14
## 140       EXTREME COLD   6121.14
## 848     TROPICAL STORM   5899.12
## 402           HURRICANE   5339.31
## 164     FLASH FLOODING   5126.05
## 411  HURRICANE/TYPHOON   4798.48
## 957           WILDFIRE   4364.20
## 873     TSTM WIND/HAIL   4356.65
```

```
## 955   WILD/FOREST FIRE   4189.54
## 464           LIGHTNING   3580.61
```

Next plot both the cost of property and crop damages in a single plot:

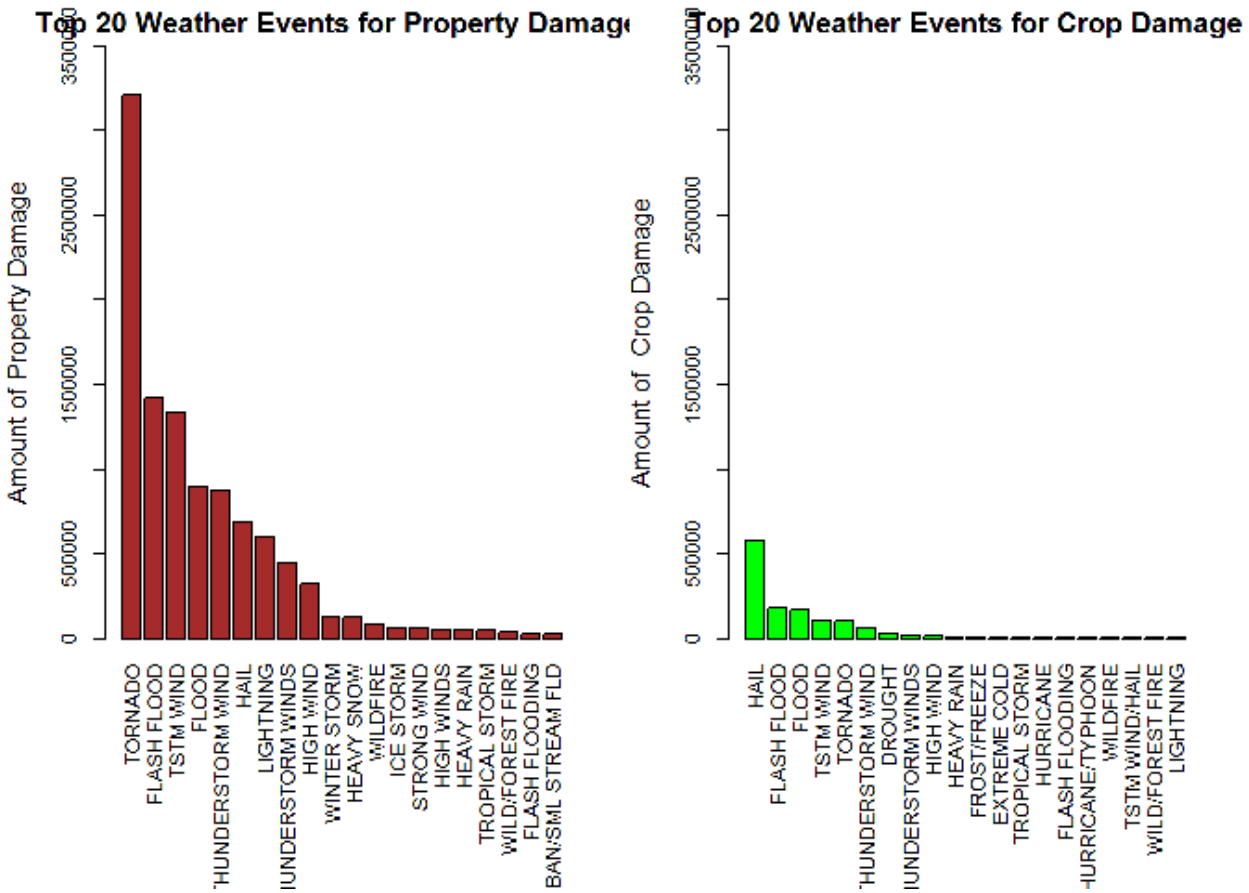
```
par(mfrow = c(1, 2), mar = c(10, 4, 2, 2), las = 3, cex = 0.7, cex.main =
1.4, cex.lab = 1.2)

barplot(totPropertySorted$Property, names.arg = totPropertySorted$Event, col
= 'Brown',

        main = 'Top 20 Weather Events for Property Damage ', ylab = 'Amount
of Property Damage', ylim = c(0, 3500000))

barplot(totCropSorted$Crop, names.arg = totCropSorted$Event, col = 'Green',

        main = 'Top 20 Weather Events for Crop Damage', ylab = 'Amount of
Crop Damage', ylim = c(0, 3500000))
```



Finally the totl damage by adding both costs (property and crop damage)

```
totTotalCost <- aggregate(noaaDF$CROPDMG+noaaDF$PROPDMG, by =
list(noaaDF$EVTYPE), "sum")

names(totTotalCost) <- c("Event", "TotalCost")

totTotalCostSorted <- totTotalCost[order(-totTotalCost$TotalCost), ][1:20, ]

totTotalCostSorted
```

```
##                  Event  TotalCost
## 834            TORNADO 3312276.68
## 153        FLASH FLOOD 1599325.05
## 856          TSTM WIND 1445168.21
## 244               HAIL 1268289.66
## 170              FLOOD 1067976.36
## 760  THUNDERSTORM WIND  943635.62
## 464          LIGHTNING  606932.39
## 786 THUNDERSTORM WINDS  464978.11
## 359          HIGH WIND  342014.77
## 972       WINTER STORM  134699.58
## 310         HEAVY SNOW  124417.71
## 957           WILDFIRE   88823.54
## 427          ICE STORM   67689.62
## 676        STRONG WIND   64610.71
## 290         HEAVY RAIN   61964.94
## 376         HIGH WINDS   57384.60
## 848     TROPICAL STORM   54322.80
## 955   WILD/FOREST FIRE   43534.49
## 95              DROUGHT   37997.67
## 164     FLASH FLOODING   33623.20
```

And a single plot

```
par(mfrow = c(1,1), mar = c(10, 4, 2, 2), las = 3, cex = 0.7, cex.main = 1.4,
cex.lab = 1.2)

barplot(totTotalCostSorted$TotalCost, names.arg = totTotalCostSorted$Event,
col = 'Black',

        main = 'Top 20 Weather Events for total Damage ', ylab = 'Amount of
total Damage', ylim = c(0, 3500000))
```