midterm_project

Hao Qin,Xinyi Wang,Qixuan Zhang,Shiyu Zhang

10/11/2018

R Markdown

Baseball Data Scrape from Web

```
##Separate Baseball stats URL
url1 <- "https://www.baseball-reference.com/teams/BOS/"
url2 <- "-schedule-scores.shtml"
years <- c(2012:2017)
urls <- str_c(url1, years, url2, sep = "")
filenames <- str_c("baseball", years, sep = "")

##Run through each URL to get "Attendence","home",and "Date"
for (i in 1:length(urls)) {
  read_url <- read_html(urls[i])
  file = read_url %>%
         html_table(fill=TRUE)%>%
         .[[1]]
  suppressMessages(
    assign(filenames[i], file)
  )

  colnames(file)[1] <-"YYYY"
  colnames(file)[5] <-"home"
  file = file[!str_detect(file$YYYY,"Gm#"),]
  file[,1] = years[i]

  if(i == 1){
    baseball <- file
  }
  else{
    baseball <- rbind.data.frame(baseball, file)
  }

}
##Clean and modify baseball dataset(change "Date" format,change column names, etc.)
baseball = baseball[!str_detect(baseball$home,"@"),]
baseball$Date = str_c(baseball$Date, baseball$YYYY, sep = ",")
baseball$Date = str_replace(baseball$Date, " \\(.*\\)", "")
baseball$Date = as.Date(baseball$Date,format="%a, %b %d,%Y")
baseball$Attendance = gsub(",","",baseball$Attendance)
baseball$Attendance = as.numeric(as.character(baseball$Attendance))
colnames(baseball)[2] = "DATE"
colnames(baseball)[3] = "NA"
```

Weather Scrape from Web

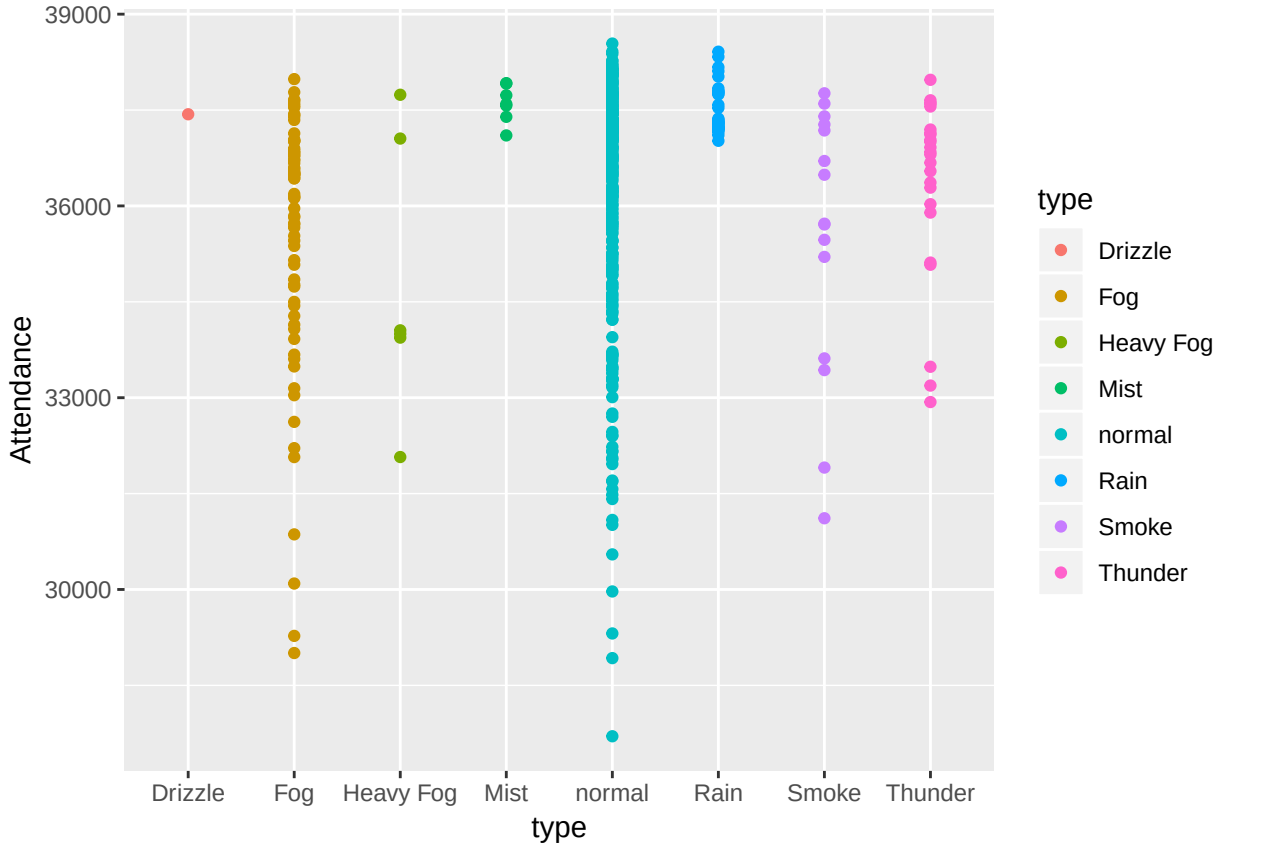
```
##Donloaded "weather.csv" from noaa.gov,read csv file
weather = read.csv("weather.csv",header=TRUE)

##Clean and modify weather dataset(change "Date" format,remove columns with all NA, etc.)
weather$DATE = as.Date(weather$DATE,format="%Y-%m-%d")
weather = weather[,colSums(is.na(weather)) < nrow(weather)]

##Build two vector contains all weather type code and corresponding actual type(information are from noa
typelist = c("Fog","Heavy Fog","Thunder","Ice Pellets","Hail","Glaze","Smoke","Blowing","Mist","Drizzle
         "Unknow Source of Preipitation","Ice Fog")
type_code = c("WT01","WT02","WT03","WT04","WT05","WT06","WT08","WT09","WT13",
              "WT14","WT15","WT16","WT17","WT18","WT19","WT22")
weather$type<-NA

for (i in 1:length(typelist)) {
  colnames(weather)[which(colnames(weather)==type_code[i])] = typelist[i]
}

weather[is.null(weather)] <- NA

##Run through all types to get the weather of a certain day, add that to the "type" column
for (m in 1:dim(weather)[1]) {
  t<-0
  for (n in 1:length(typelist)) {
    if (is.null(weather[m,typelist[n]])) {
      weather[m,typelist[n]] = NA
    }
    if (!is.na(weather[m,typelist[n]])) {
      weather[m,"type"] =  typelist[n]
      t<-t+1
    }
  }
  if(t==0)
    weather[m,"type"] = "normal"
}

##Select only columns that matters to this project
weather = weather[,c("DATE","TMAX","TMIN","type")]
x = cbind(weather$TMAX,weather$TMIN)
weather$tavg = apply(x,1,mean)
```

Join Baseball and Weather

```
##Join baseball and weather dataset by "DATE"
baseball_weather = inner_join(baseball,weather,by="DATE")
```

Red Sox home game Attendence vs. Weather Plot

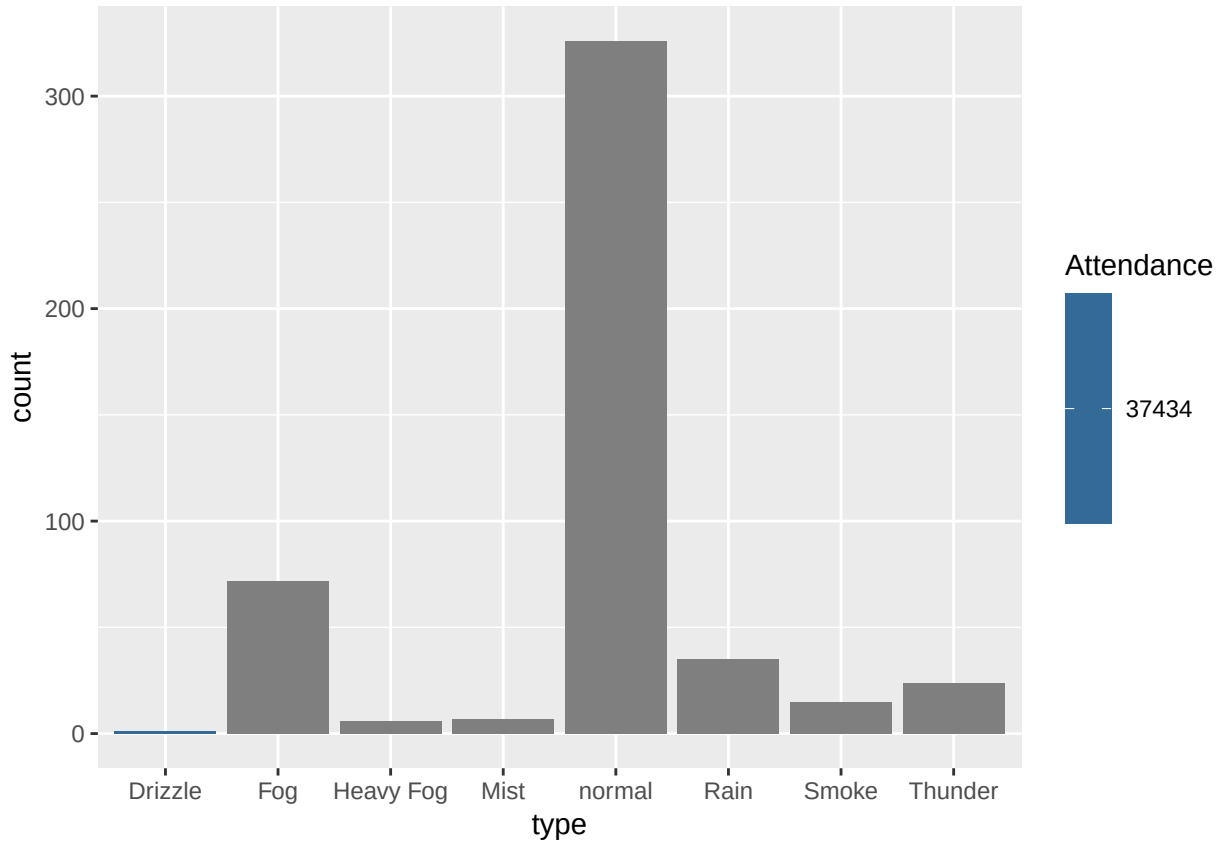
```
# View(baseball_weather)
# names(baseball_weather)
# summary(baseball_weather)
#scatter plot,boxplot and histogram plot of weather and baseball
b1<-ggplot(data=baseball_weather)+geom_point(aes(x=type,y=Attendance,col=type))
b1
```



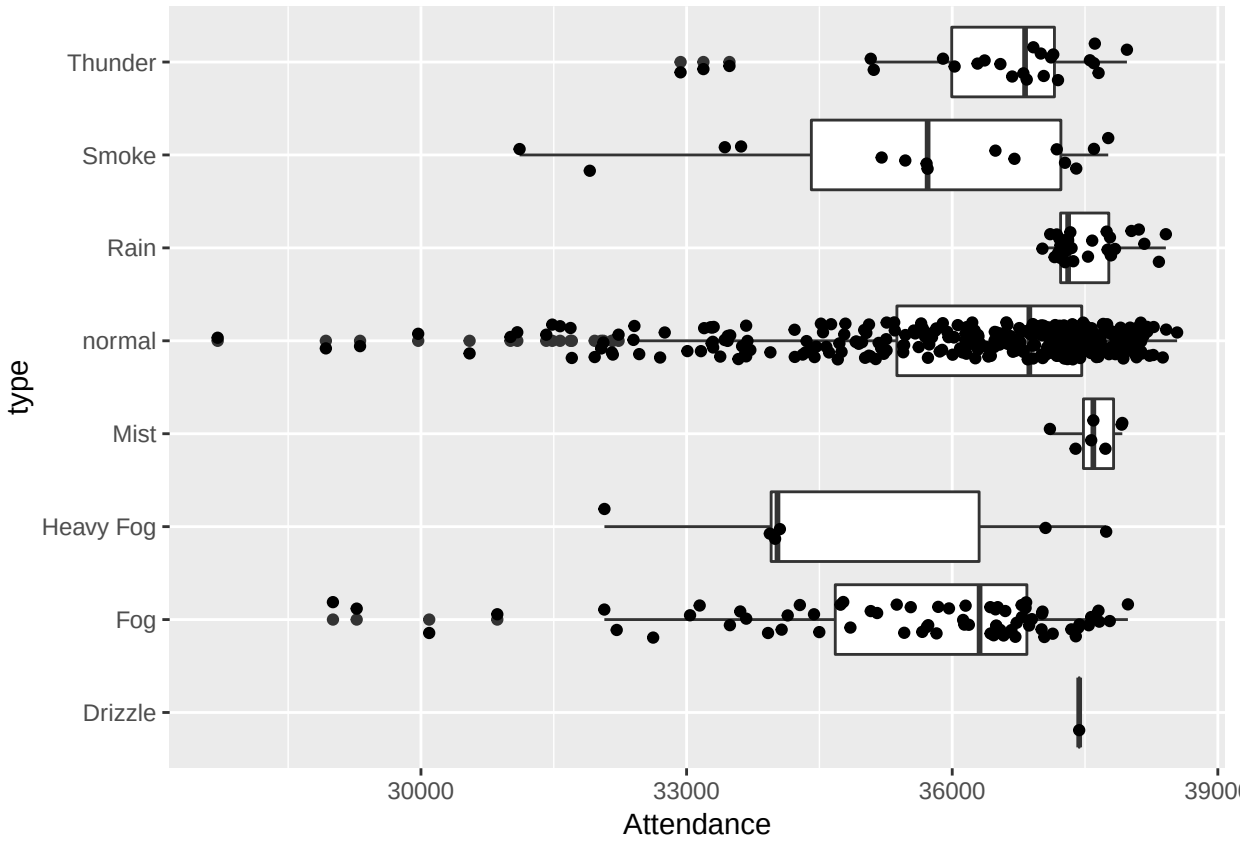

```
b11 <- ggplot(data=baseball_weather,aes(x=type,fill=Attendance))+
  geom_histogram(stat="count",position = "identity")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

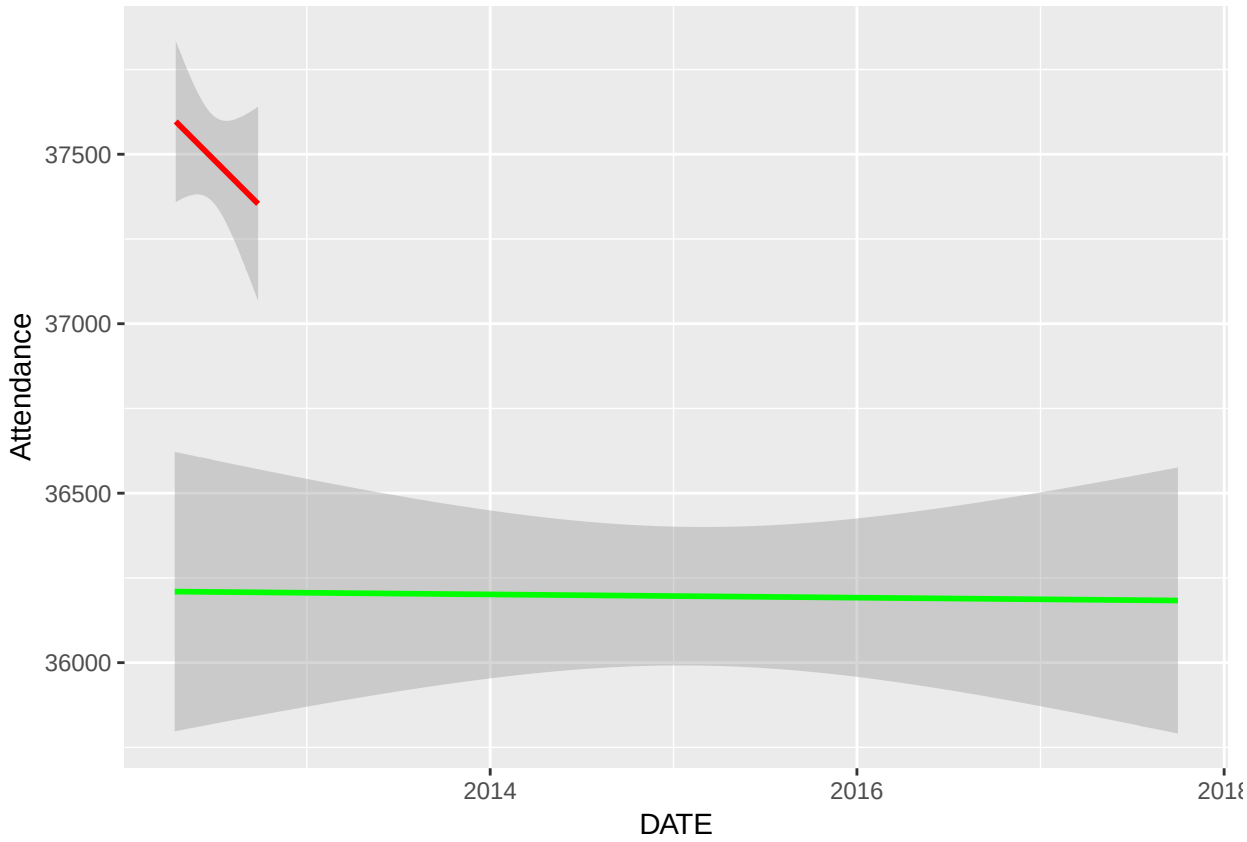
```
b11
```



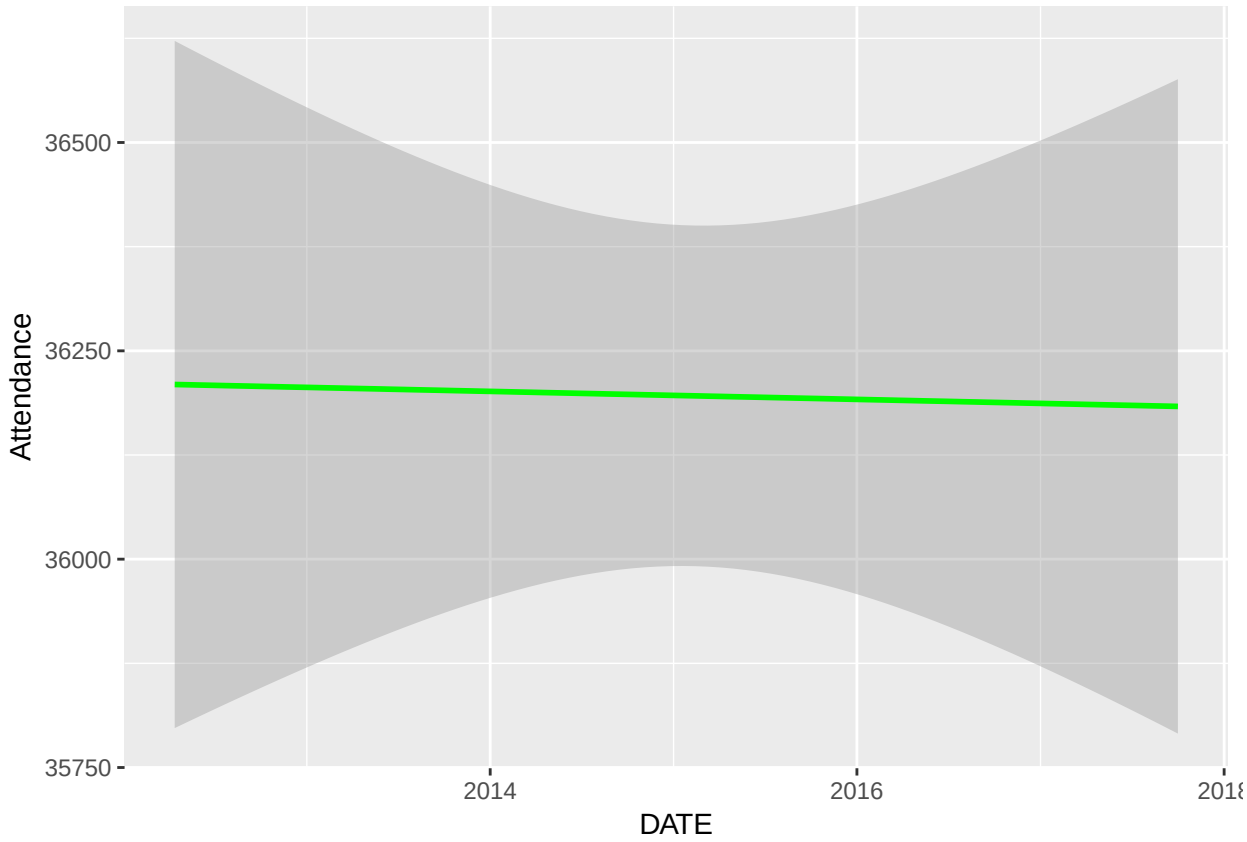

```
b12<-ggplot(data=baseball_weather,aes(x=type,y=Attendance))+
        geom_boxplot(fill="white")+
        geom_jitter(width = 0.2)+coord_flip()
b12
```



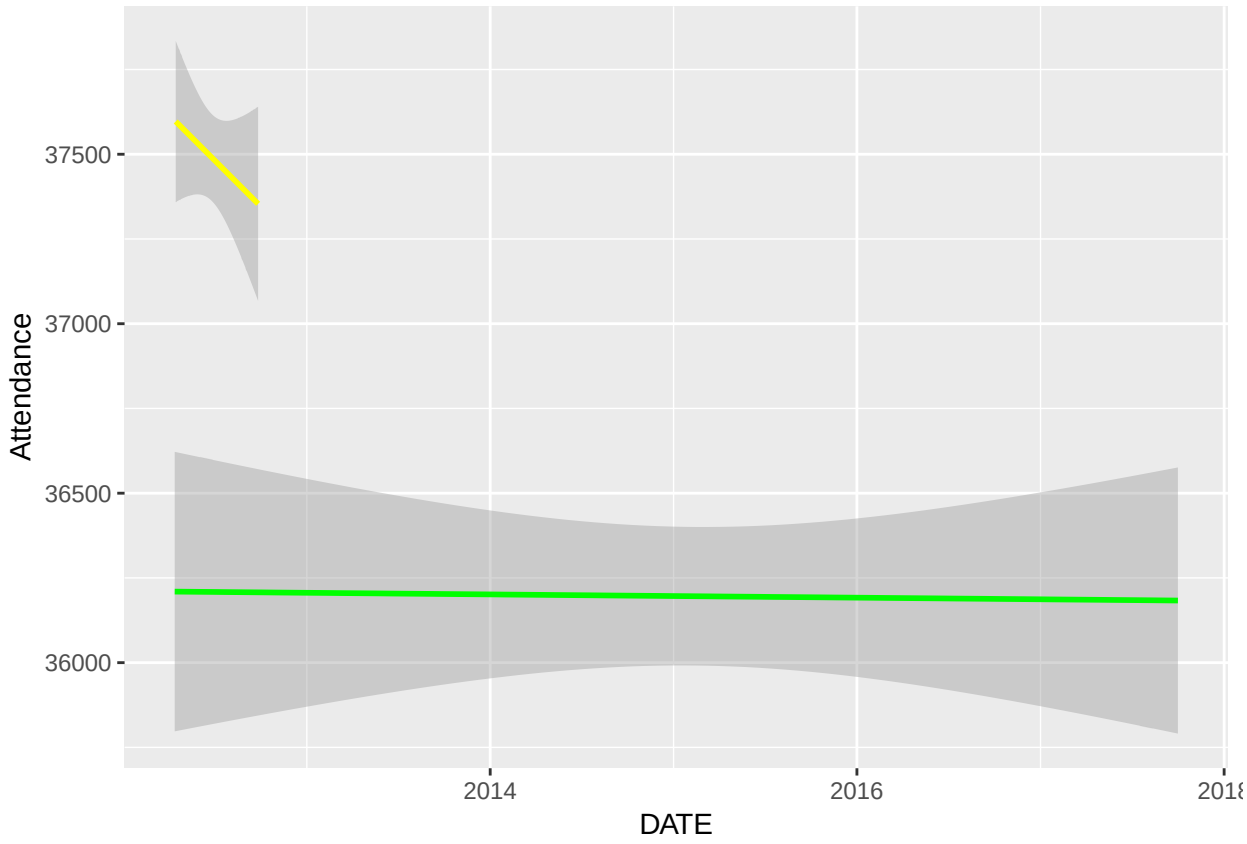

```
#Compared the difference betwwen normal and rain
normal<- baseball_weather %>% filter(type=="normal")
#View(normal)
Rain<- baseball_weather %>% filter(type=="Rain")
snow<- baseball_weather %>% filter(type=="Snow")
b2<-ggplot()+geom_smooth(data=Rain,aes(x=DATE,y=Attendance),color="red",method="lm")+
  geom_smooth(data=normal,aes(x=DATE,y=Attendance),color="green",method="lm")
b2
```



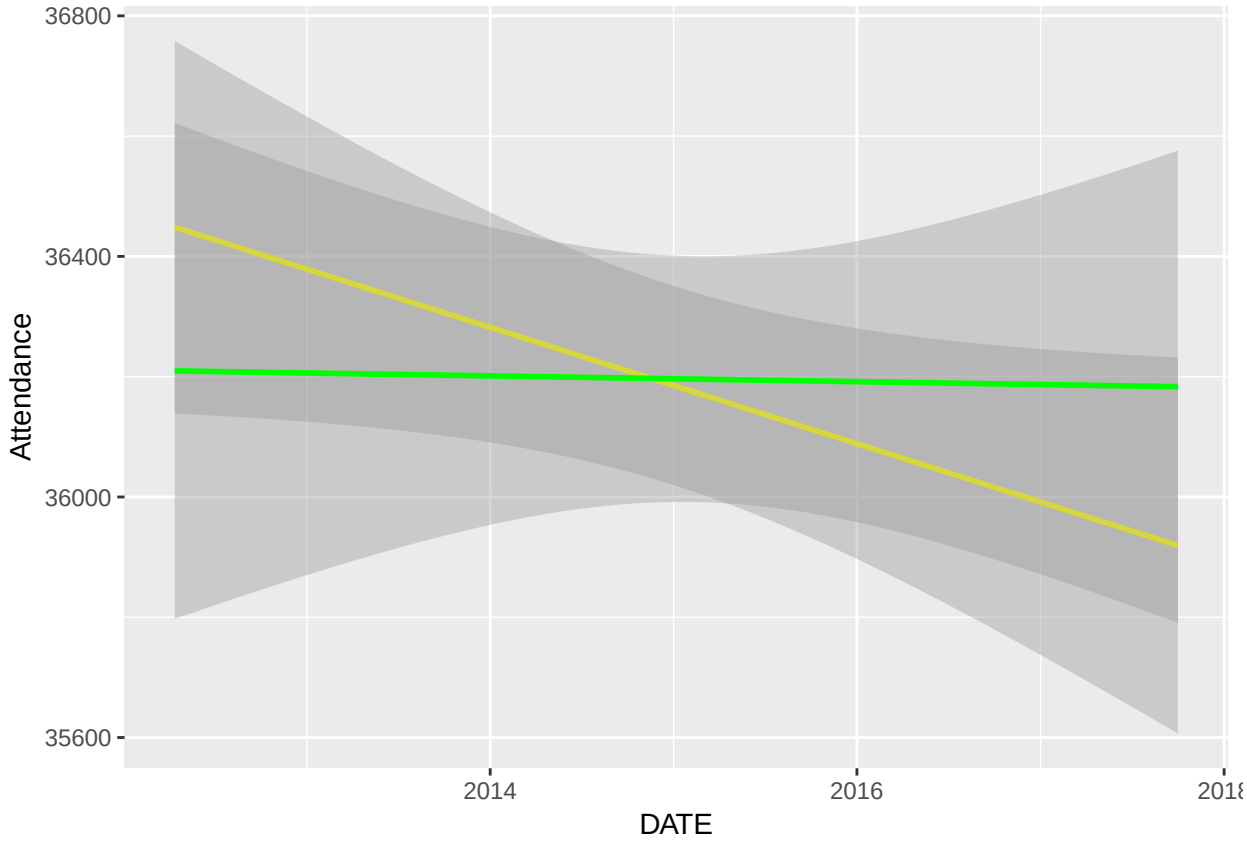

```
#Compared the difference between normal and snow
b3<-ggplot()+geom_smooth(data=snow,aes(x=DATE,y=Attendance),color="balck",method="lm")+
  geom_smooth(data=normal,aes(x=DATE,y=Attendance),color="green",method="lm")
b3
```




```
#Compared the difference between normal,snow and rain
b4<-ggplot()+geom_smooth(data=snow,aes(x=DATE,y=Attendance),color="red",method="lm")+
  geom_smooth(data=normal,aes(x=DATE,y=Attendance),color="green",method="lm")+
  geom_smooth(data=Rain,aes(x=DATE,y=Attendance),color="yellow",method="lm")
b4
```




```
# compare the difference between fog and normal
fog<-baseball_weather %>% group_by("fog")
# View(fog)
b5<-ggplot()+geom_smooth(data=fog,aes(x=DATE,y=Attendance),color="yellow",method="lm")+
  geom_smooth(data=normal,aes(x=DATE,y=Attendance),color="green",method="lm")
b5
```



Basketball Scrape from Web

```
url3 = "http://www.espn.com/nba/team/schedule/_/name/bos/season/"
url4 = "/seasontype/2"
years = c(2012:2018)

urls = str_c(url3,years,url4,sep = "")
filenames <- str_c("basketball", years, sep = "")
for (i in 1:length(urls)) {
  read_url <- read_html(urls[i])
  file = read_url %>%
         html_nodes('.ml4 a') %>%
         html_attrs() %>%
         map(1) %>%
         unlist()

  date = read_url %>%
         html_nodes('.Table2__even~ .Table2__even+ .Table2__even .Table2__td:nth-child(1) span') %>%
         html_text()

  home = read_url %>%
         html_nodes('.pr2:nth-child(1)') %>%
         html_text()

  YYYY = NA
```

```
  ##Delete cancelled games & unwanted 'home'
  if(years[i]==2013){
    date<-date[-which(date=="Tue, Apr 16")]
    home = home[-c(81)]
  }
  else if(years[i]==2016){
    date<-date[-which(date=="Sat, Jan 23")]
    home = home[-c(45)]
  }
  ##Remove blank 'home'
  if(home[1]==""){
    home = home[-1]
  }

  suppressMessages(
    assign(filenames[i], cbind(as.data.frame(file,stringsAsFactors=FALSE),date,home,YYYY))
  )

}

##Remove games not in 2012-2017 and putting same year together
basketball2012 = rbind(basketball2012[5:66,],basketball2013[1:30,])
basketball2012$YYYY = "2012"
basketball2013 = rbind(basketball2013[31:81,],basketball2014[1:31,])
basketball2013$YYYY = "2013"
basketball2014 = rbind(basketball2014[32:82,],basketball2015[1:29,])
basketball2014$YYYY = "2014"
basketball2015 = rbind(basketball2015[30:82,],basketball2016[1:32,])
basketball2015$YYYY = "2015"
basketball2016 = rbind(basketball2016[33:82,],basketball2017[1:34,])
basketball2016$YYYY = "2016"
basketball2017 = rbind(basketball2017[35:82,],basketball2018[1:40,])
basketball2017$YYYY = "2017"

##Combine all years together and clean
basketball = rbind.data.frame(basketball2012,basketball2013,basketball2014,basketball2015,basketball201
basketball = basketball[!str_detect(basketball$home,"@"),]
basketball$file = gsub(".*=","",basketball$file)

##Read Attendence
url5 = "http://www.espn.com/nba/game?gameId="
gameID = basketball$file
urls = str_c(url5,gameID,sep = "")

get_A<-function(urls){
  read_html(urls) %>%
  html_nodes('div[class=\"game-info-note capacity\"]') %>%
  html_text() ->try
  try_A<-str_trim(unlist(str_split(try[1],c(":"))[1])[2])
  A<-as.numeric(str_c(unlist(str_split(try_A,","))[1],unlist(str_split(try_A,","))[2]))
  return(A)
}
```

```
Attend<-data.frame()
for(i in 1:length(urls)){
  Attend<-rbind(Attend,get_A(urls[i]))
}

##Clean and modify basketball dataset
basketball = cbind(basketball,Attend)
colnames(basketball)[5] <-"Attendence"
colnames(basketball)[1] <-"gameID"
colnames(basketball)[2] <-"DATE"
basketball$DATE = str_c(basketball$DATE, basketball$YYYY, sep = ",")
basketball$DATE = as.Date(basketball$DATE,format="%a, %b %d,%Y")
```

Join Basketball and Weather

```
##Join basketball and weather dataset by "DATE"
basketball_weather = inner_join(basketball,weather,by="DATE")
```

Celtics home game Attendence vs. Weather Plot

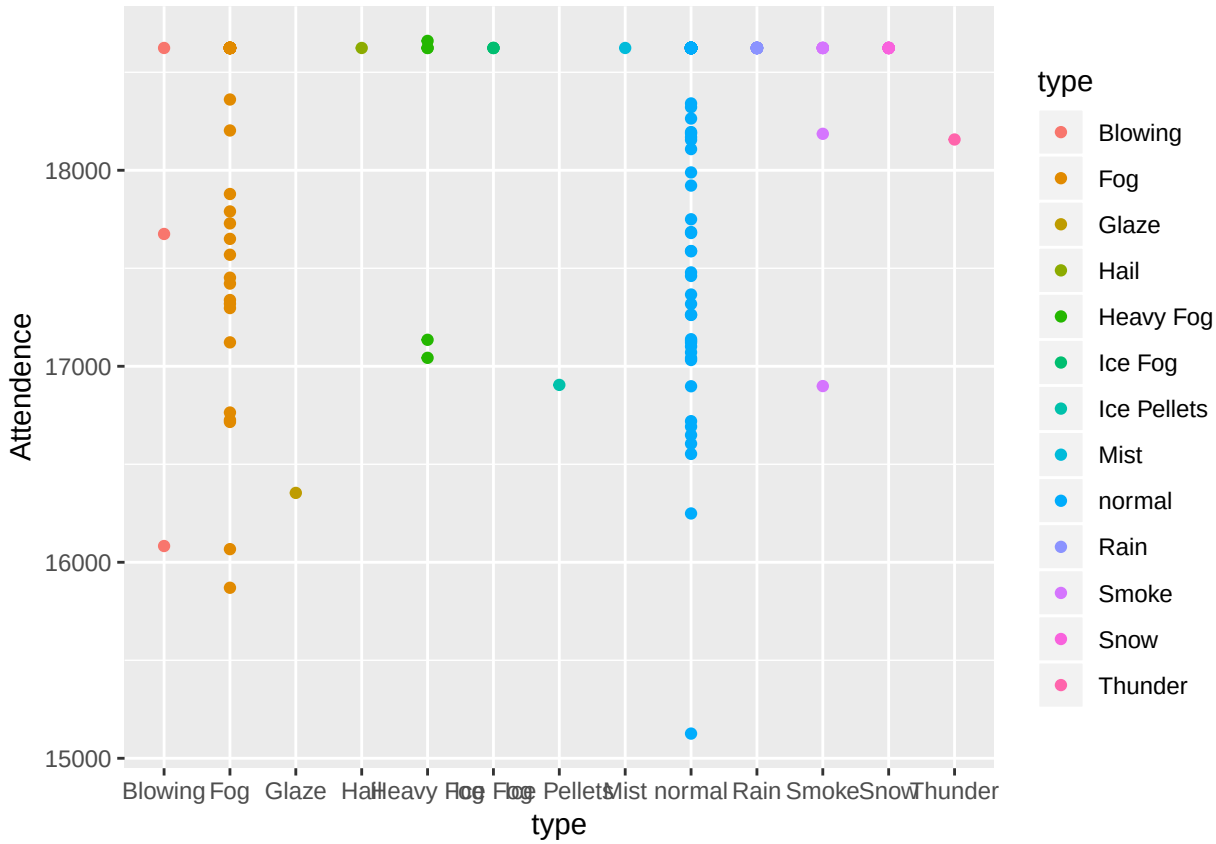
```
# View(basketball_weather)
names(basketball_weather)
```

```
## [1] "gameID"     "DATE"       "home"       "YYYY"       "Attendence"
## [6] "TMAX"       "TMIN"       "type"       "tavg"
```

```
summary(basketball_weather$type)
```

```
##    Length     Class      Mode
##       258 character character
```

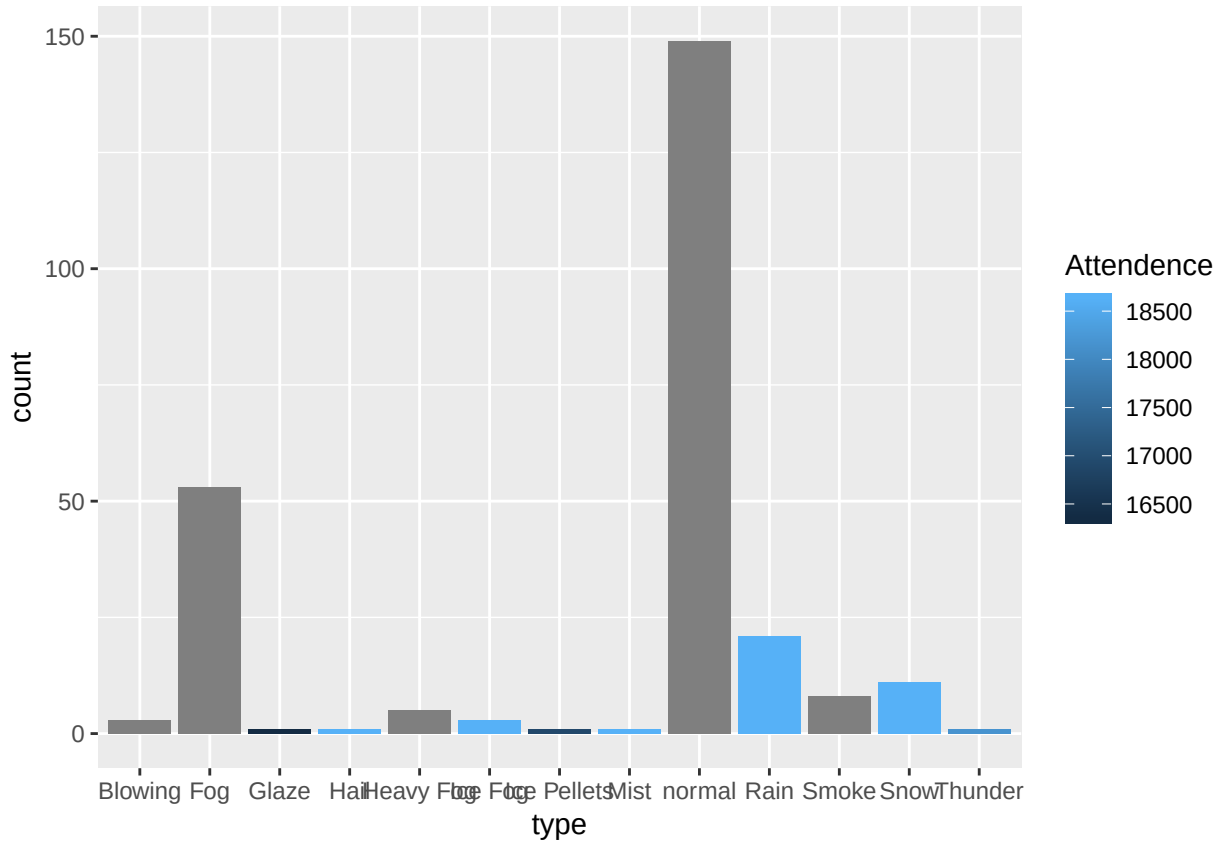
```
#scatter plot and histogram plot of weather and basketball
g1<-ggplot(data=basketball_weather)+geom_point(aes(x=type,y=Attendence,col=type))
g1
```



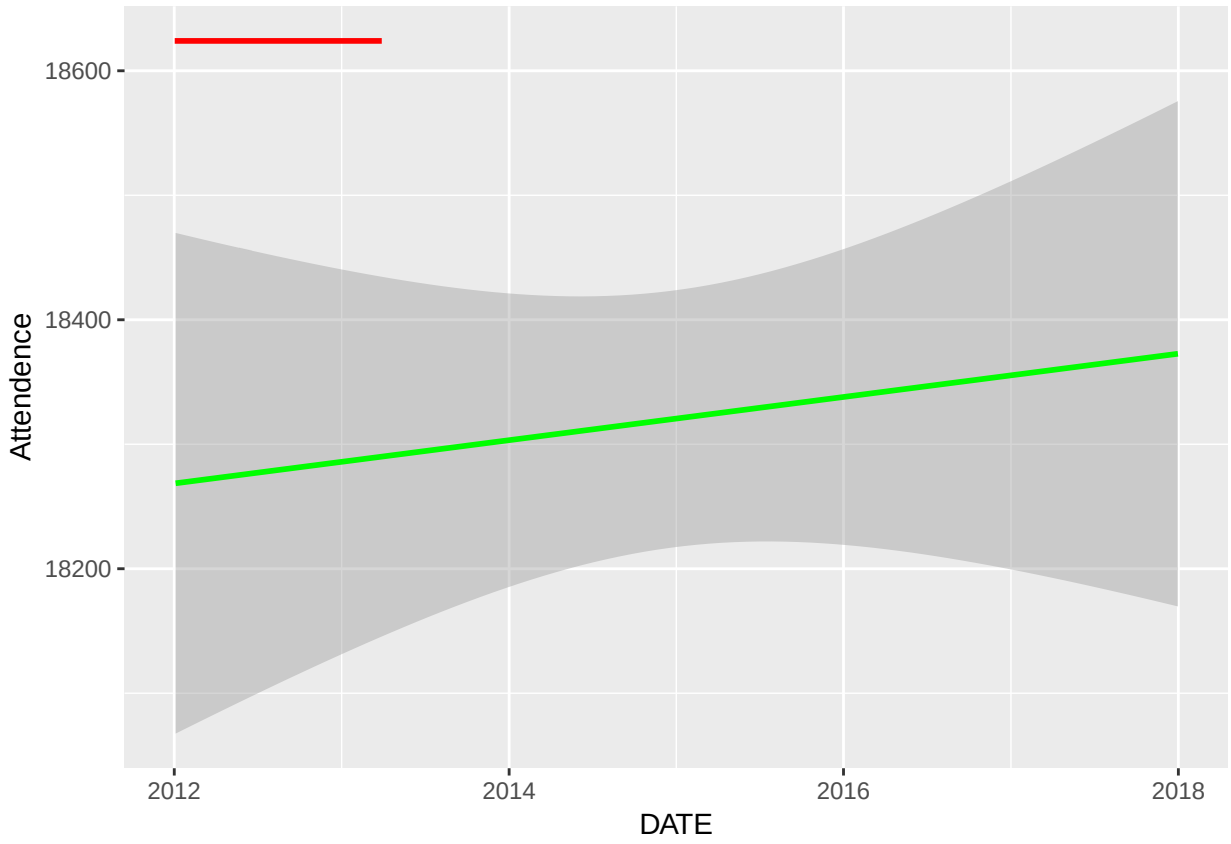

```
g11 <- ggplot(data=basketball_weather,aes(x=type,fill=Attendence))+
  geom_histogram(stat="count",position = "identity")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

```
g11
```

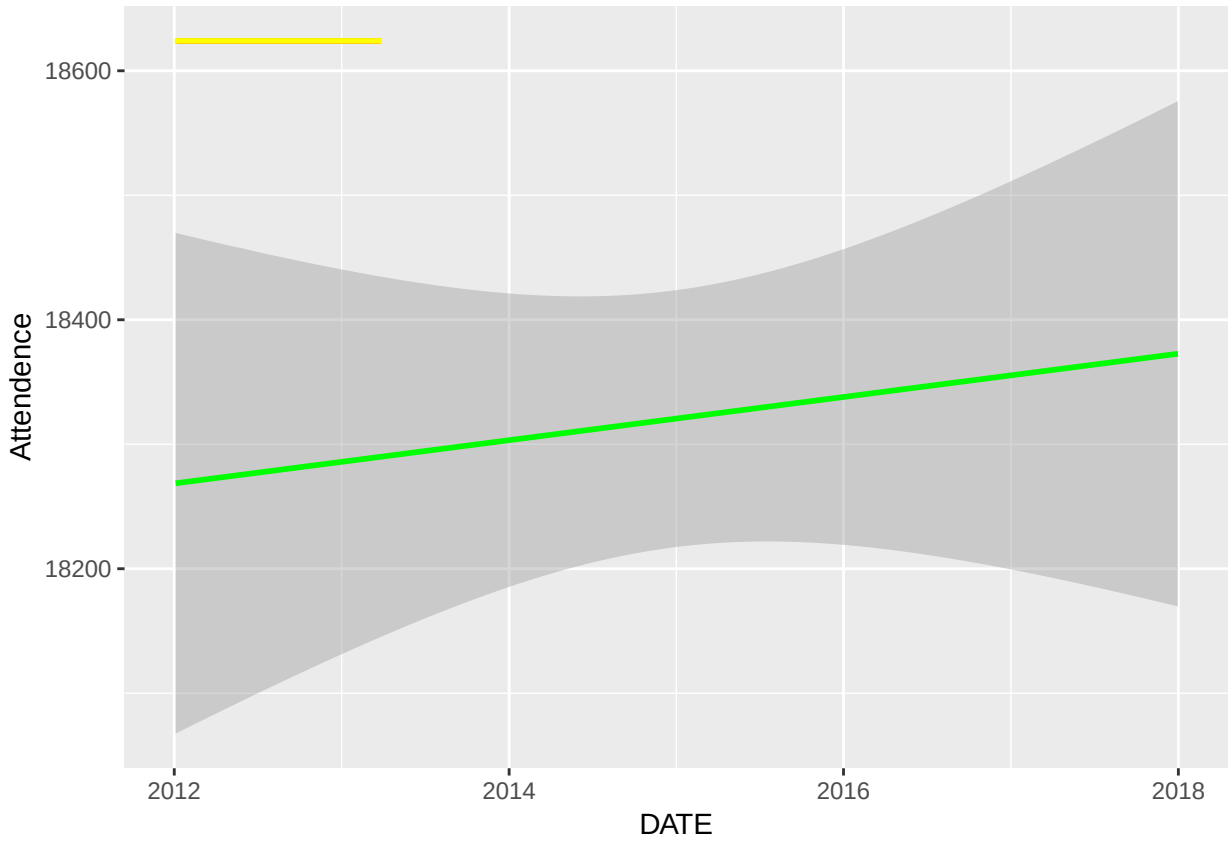



```
#Compared the difference betwwen normal and rain
normal<- basketball_weather %>% filter(type=="normal")
# View(normal)
Rain<- basketball_weather %>% filter(type=="Rain")
snow<- basketball_weather %>% filter(type=="Snow")
g2<-ggplot()+geom_smooth(data=Rain,aes(x=DATE,y=Attendence),color="red",method="lm")+
  geom_smooth(data=normal,aes(x=DATE,y=Attendence),color="green",method="lm")
g2
```

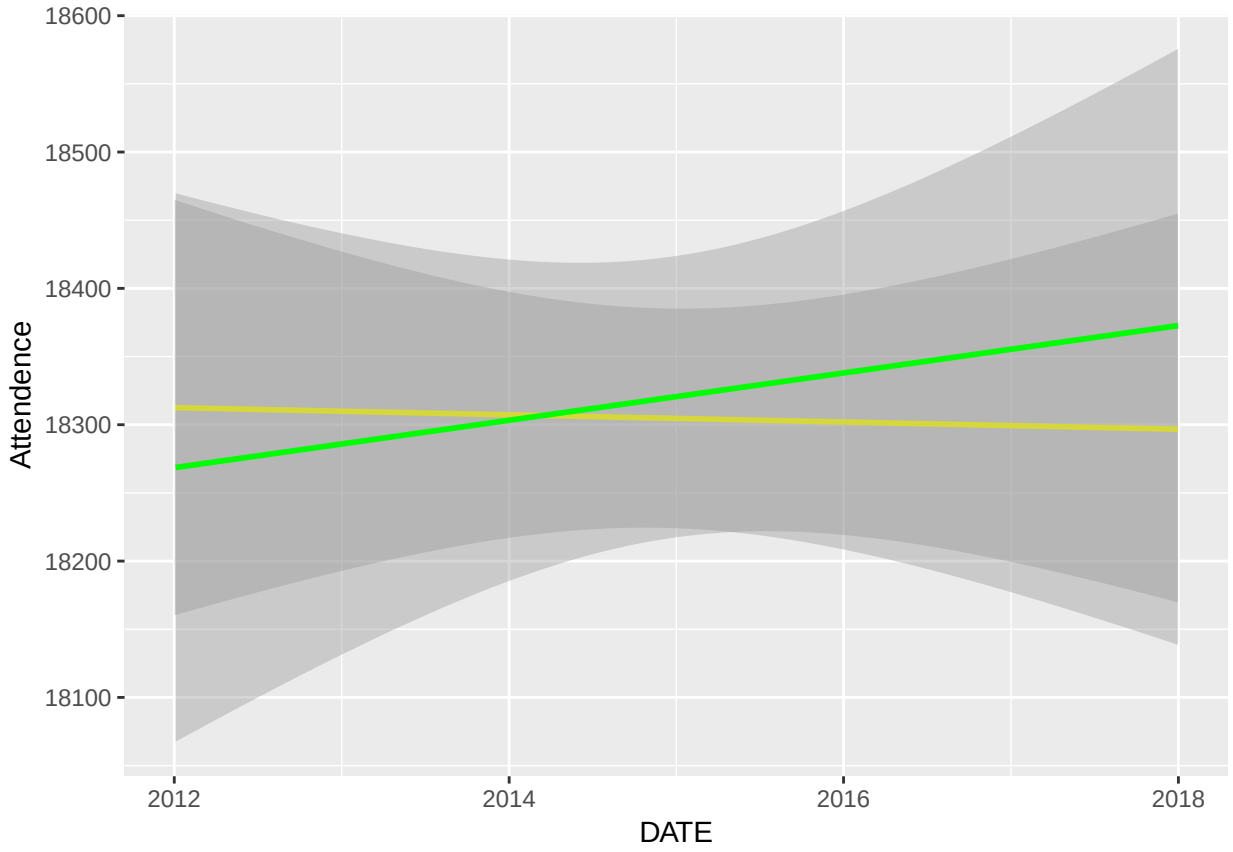



```
#Compared the difference between normal and snow
g3<-ggplot()+geom_smooth(data=snow,aes(x=DATE,y=Attendence),color="red",method="lm")+
  geom_smooth(data=normal,aes(x=DATE,y=Attendence),color="green",method="lm")
g3
```


```
#Compared the difference between normal,snow and rain
g4<-ggplot()+geom_smooth(data=snow,aes(x=DATE,y=Attendence),color="red",method="lm")+
  geom_smooth(data=normal,aes(x=DATE,y=Attendence),color="green",method="lm")+
  geom_smooth(data=Rain,aes(x=DATE,y=Attendence),color="yellow",method="lm")
g4
```




```
# compare the difference between fog and normal
fog<-basketball_weather %>% group_by("fog")
# View(fog)
g5<-ggplot()+geom_smooth(data=fog,aes(x=DATE,y=Attendence),color="yellow",method="lm")+
  geom_smooth(data=normal,aes(x=DATE,y=Attendence),color="green",method="lm")
g5
```



Shiny

```
library(shiny)
library(dplyr)
library(datasets)
a = select(baseball_weather,"YYYY","Attendance","type")
a = a %>%
  group_by(YYYY,type) %>%
  summarise(Attendance = sum(Attendance))
a = as.data.frame(a)

b = select(basketball_weather,"YYYY","Attendence","type")
b$YYYY = as.integer(b$YYYY)
b = b %>%
  group_by(YYYY,type) %>%
  summarise(Attendence = sum(Attendence))
b = as.data.frame(b)

ui <- fluidPage(
  # Give the page a title
  titlePanel("Weather vs. Attendence"),
  # Generate a row with a sidebar
  sidebarLayout(
    # Define the sidebar with one input
    sidebarPanel(
```

```
      selectInput("YYYY", "Year:",
                  choices=c(2012:2017)),
      hr()
    ),
    # Create a spot for the barplot
    mainPanel(
      "main panel",
      fluidRow(
        splitLayout(cellWidths = c("50%", "50%"), plotOutput("BarPlot1"), plotOutput("BarPlot2"))
              )
    )

  )
)

server <- function(input, output) {
output$BarPlot1 <- renderPlot({
    # Render a barplot
    barplot(a[which(a$YYYY==input$YYYY),"Attendance"],
            main="Baseball(Red Sox)",
            names.arg=a[which(a$YYYY==input$YYYY),"type"],
            col=terrain.colors(length(a[which(a$YYYY==input$YYYY),"Attendance"])),
            ylab="Attendance",
            xlab="Weather Condition")
  })
output$BarPlot2 <-  renderPlot({
  barplot(b[which(b$YYYY==input$YYYY),"Attendence"],
            main="Basketball(Celtics)",
            names.arg=b[which(b$YYYY==input$YYYY),"type"],
            col=cm.colors(length(b[which(b$YYYY==input$YYYY),"Attendence"])),
            ylab="Attendance",
            xlab="Weather Condition")
  })
}

shinyApp(ui = ui, server = server)
```

Shiny applications not supported in static R Markdown documents