STAT 8010 R Lab 4: Data Summary/Visualization III

Whitney Huang

August 31, 2020

Contents

Interquartile range (IQR)

```
data1 <- c(13, 18, 13, 14, 13, 16, 14, 21, 13)
IQR(data1, type = 1)
```

```
## [1] 3
```

```
data2 <- c(13, 18, 13, 14, 13, 16, 14, 210, 13)
IQR(data2, type = 1)
```

```
## [1] 3
```

Percentiles

```
#Q1
quantile(data1, 0.25, type = 1)
```

```
## 25%
##  13
```

```
#Q2 aka median
quantile(data1, 0.5, type = 1)
```

```
## 50%
##  14
```

```
#Q3
quantile(data1, 0.75, type = 1)
```

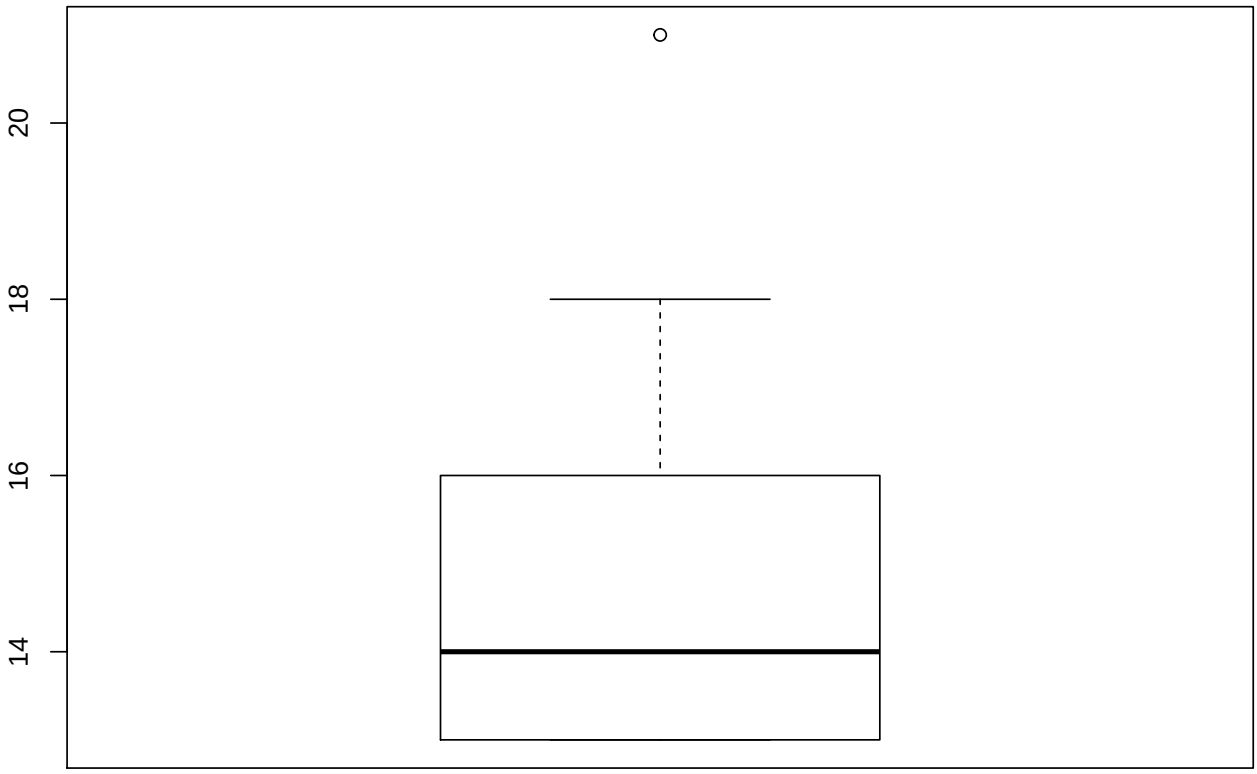
```
## 75%
##  16
```

```
data3 <- c(13, 18, 13, 14, 13, 16, 14, 21, 13, 9,
           27, 18, 25, 20, 6)
quantile(data3, c(0.35, 0.65), type = 1)
```

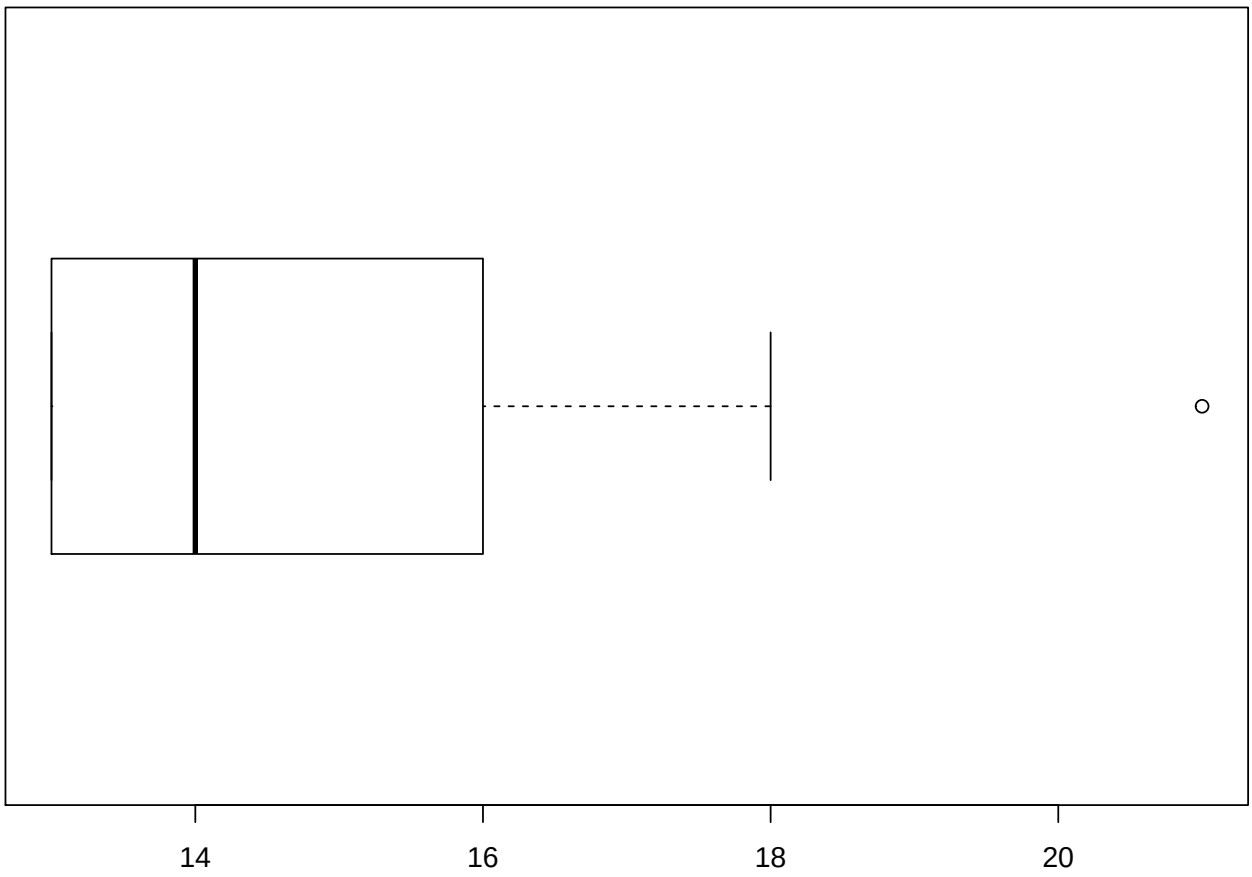
```
## 35% 65%
##  13  18
```

Boxplot

```
boxplot(data1)
```




```
boxplot(data1, horizontal = T)
```



Load the ORD flight dataset

```
url <- "https://whitneyhuang83.github.io/STAT8010/Data/flights.csv"
ORD <- read.csv(url, header = TRUE)
```

Let's take a look at the data

```
dim(ORD)
```

```
## [1] 12678     4
```

```
n <- dim(ORD)[1]
head(ORD)
```

```
##   month carrier origin arr_delay
## 1     1      UA    EWR        12
## 2     1      AA    LGA         8
## 3     1      AA    LGA        14
## 4     1      AA    LGA         4
## 5     1      UA    LGA        20
## 6     1      UA    EWR        21
```

2 way Frequency Table

```
tab3 <- table(ORD[, c("carrier", "origin")])
tab3
```

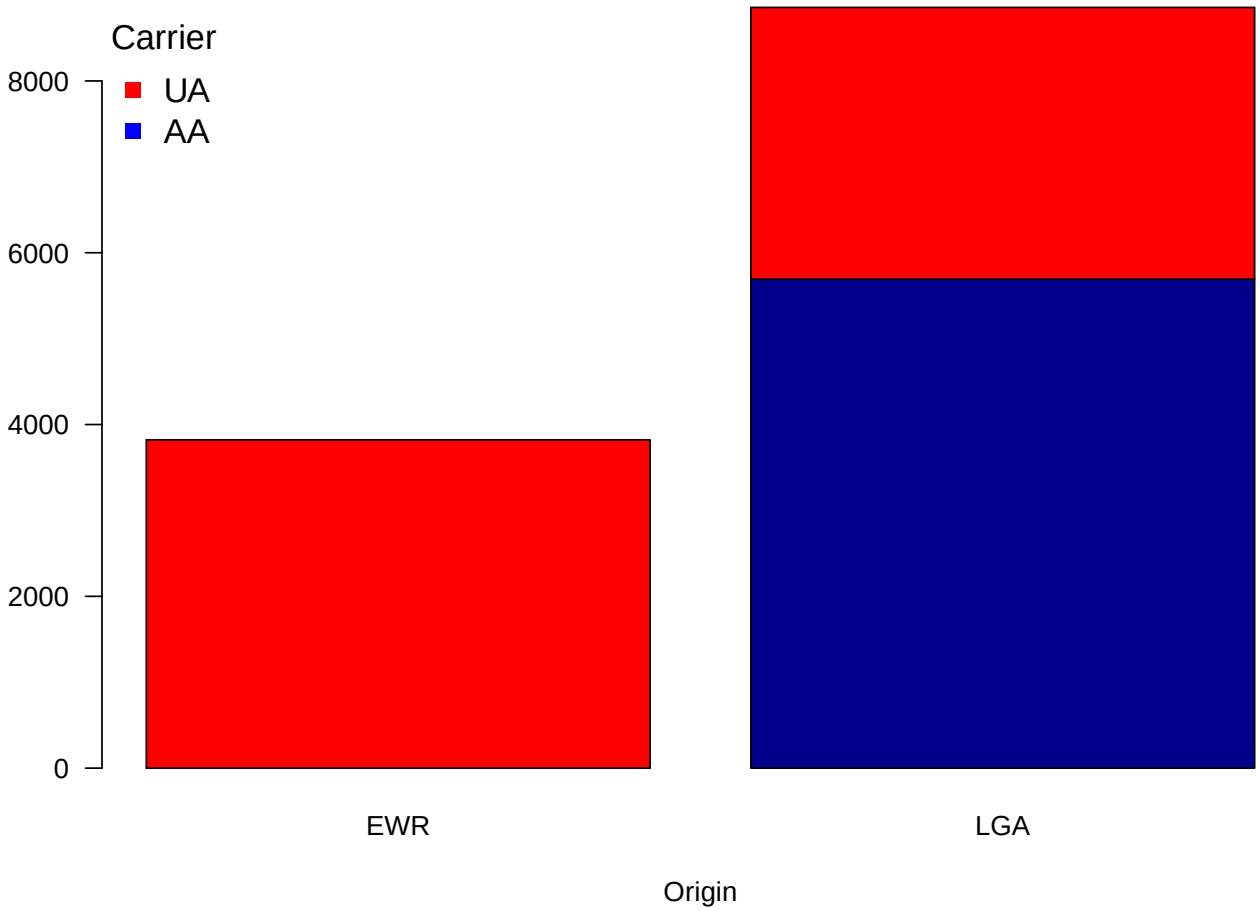
```
##        origin
## carrier  EWR  LGA
##      AA    0 5694
##      UA 3822 3162
```

```
tab4 <- table(ORD[, c("carrier", "origin")])/n
tab4
```

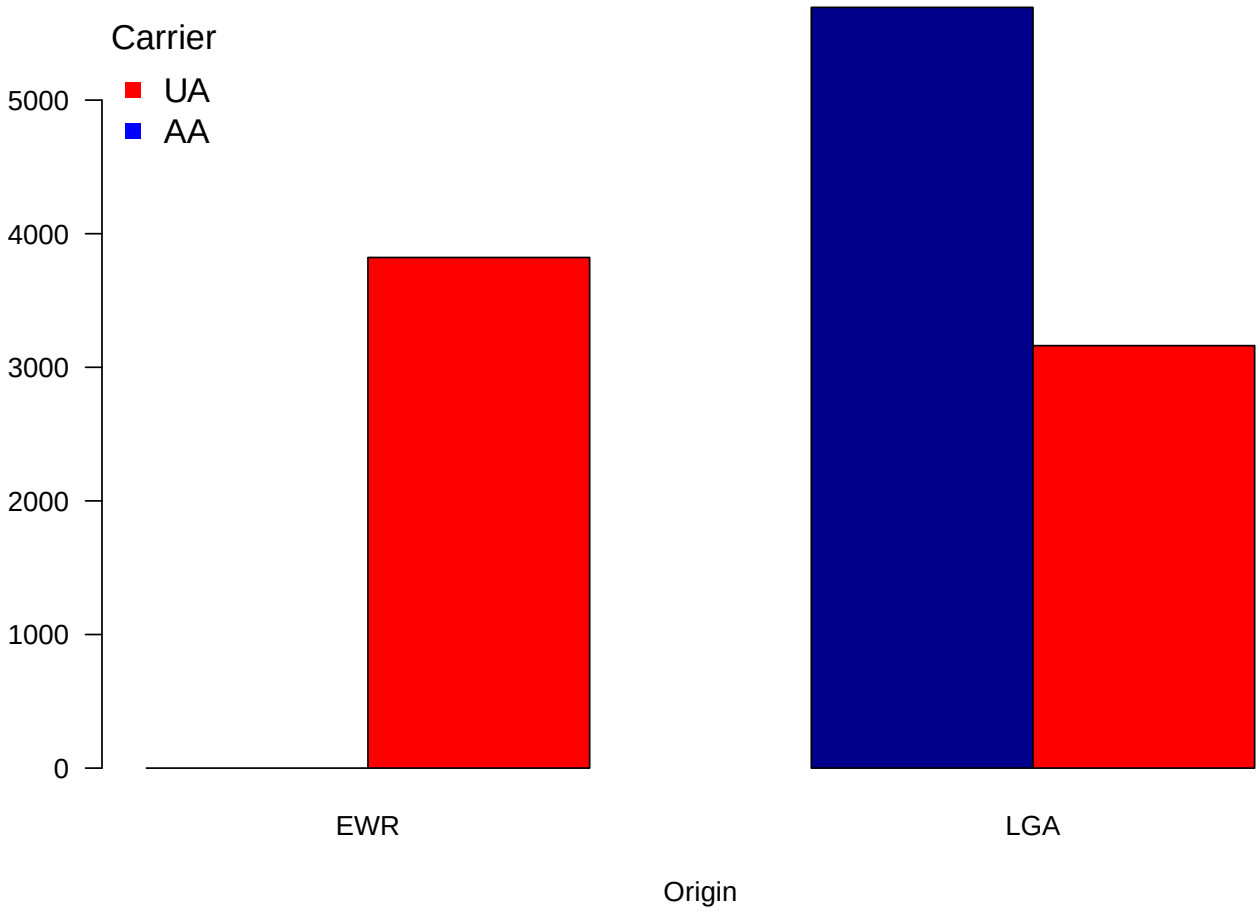
```
##        origin
## carrier       EWR       LGA
##      AA 0.0000000 0.4491245
##      UA 0.3014671 0.2494084
```

Stacked/dodged bar chart

```
## Stacked bar chart
barplot(tab3, xlab = "Origin", col = c("darkblue","red"), args.legend = list(x = "topleft"), las = 1)
legend("topleft", legend = c("UA", "AA"),
       pch = 15, col = c("red", "blue"), bty = "n", cex = 1.25, title = "Carrier")
```




```
## Dodged bar chart
barplot(tab3, xlab = "Origin", col = c("darkblue","red"), args.legend = list(x = "topleft"), las = 1, b
legend("topleft", legend = c("UA", "AA"),
       pch = 15, col = c("red", "blue"), bty = "n", cex = 1.25, title = "Carrier")
```



Qualitative vs Quantitative: Side by Side Boxplots

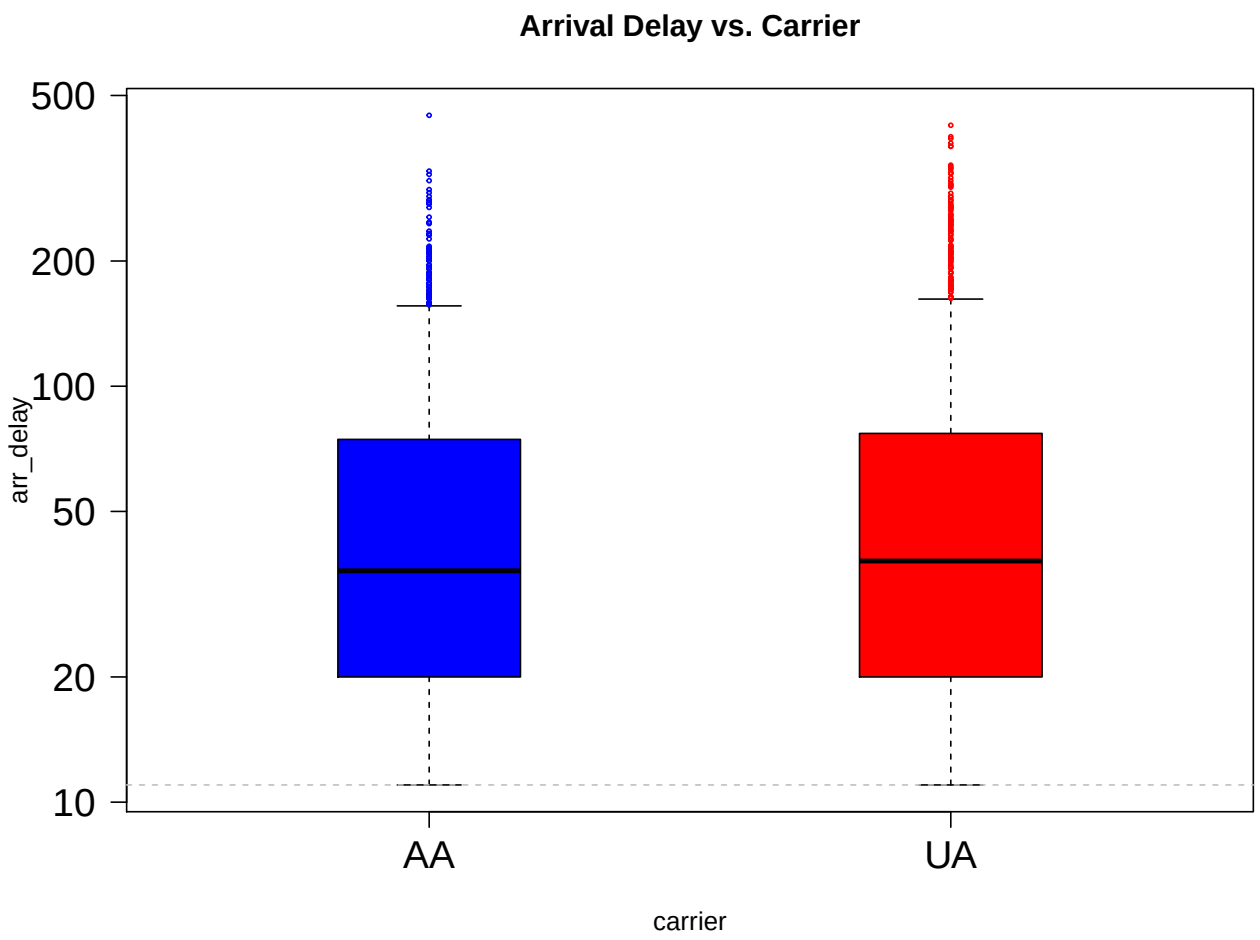
```
attach(ORD)
library(tidyverse)
```

```
## -- Attaching packages --------
```

```
## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
```

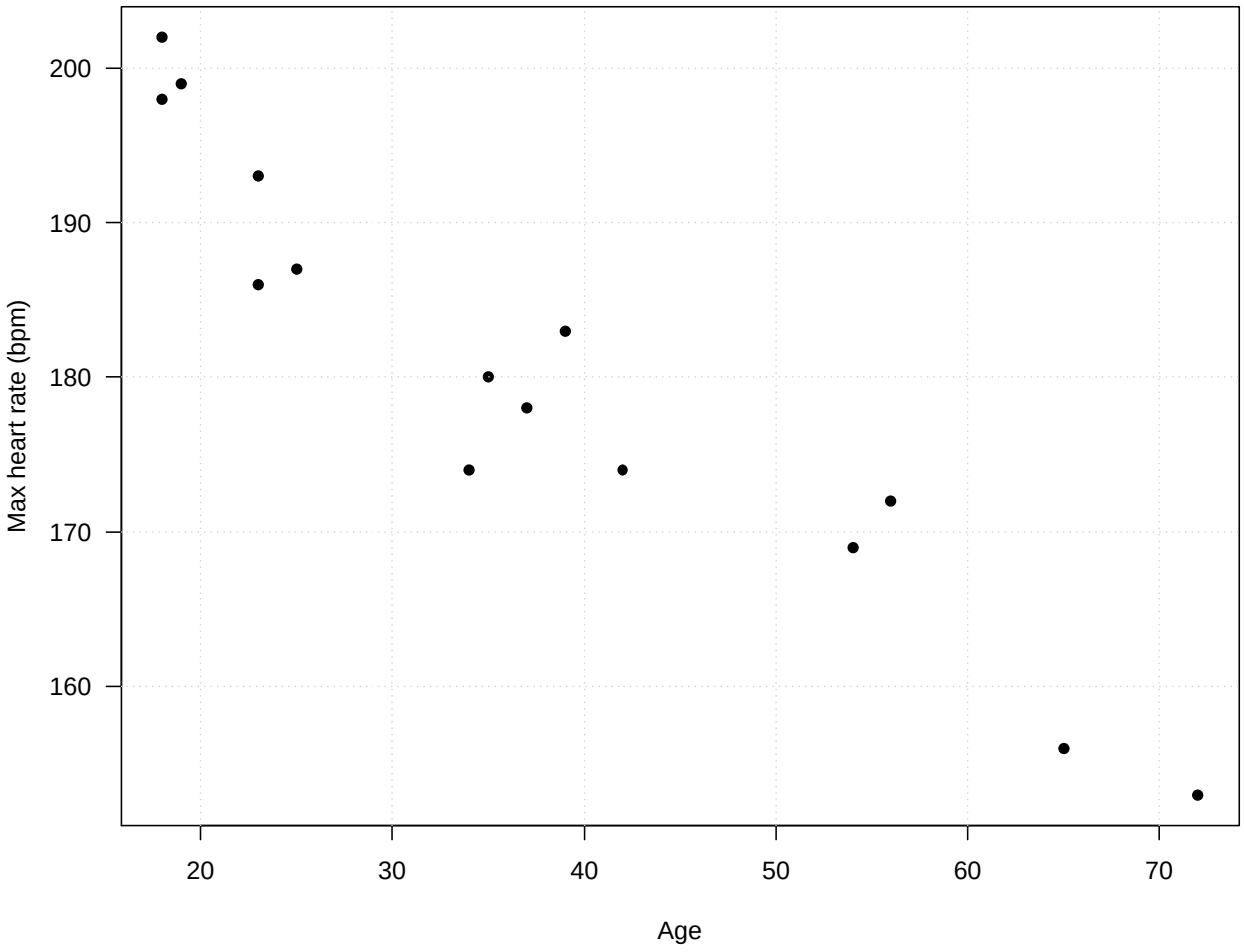
```
## -- Conflicts -----------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
boxplot(arr_delay ~ carrier, filter(ORD, arr_delay > 10), boxwex = 0.35,
        col = c("blue", "red"),
        staplewex = 0.35, outwex = 0.35,
        cex.axis = 1.5, las = 1, log = "y",
        outcol = c("blue", "red"),
        outcex = 0.35, main = "Arrival Delay vs. Carrier")
abline(h = 11, lty = 2, col = "gray")
```



Quantitative vs Quantitative: Scatter Plot

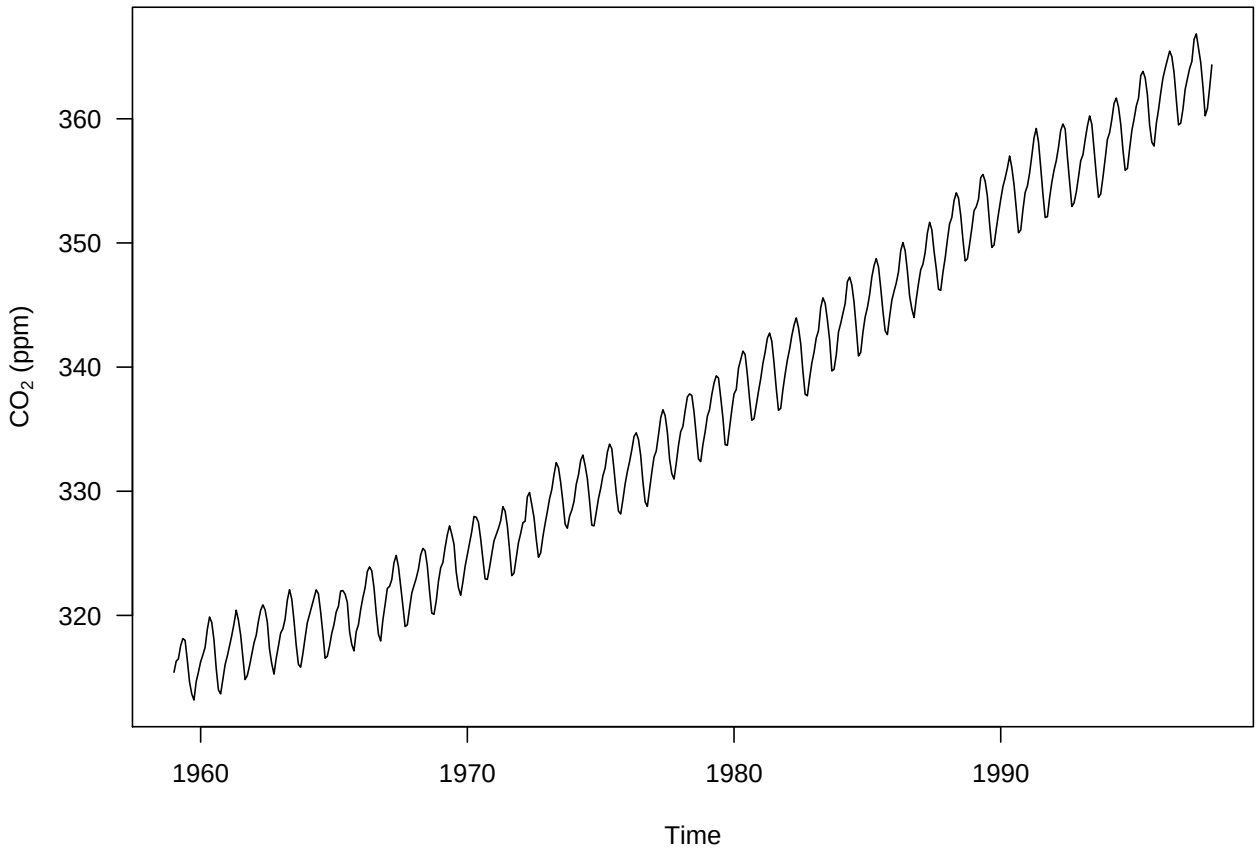
```
url <- "https://whitneyhuang83.github.io/STAT8010/Data/maxHeartRate.csv"
dat <- read.csv(url, header = TRUE)

par(las = 1, mar = c(4.1, 4.1, 1.1, 1.1))
plot(dat$Age, dat$MaxHeartRate, pch = 16, xlab = "Age", ylab = "Max heart rate (bpm)")
grid()
```



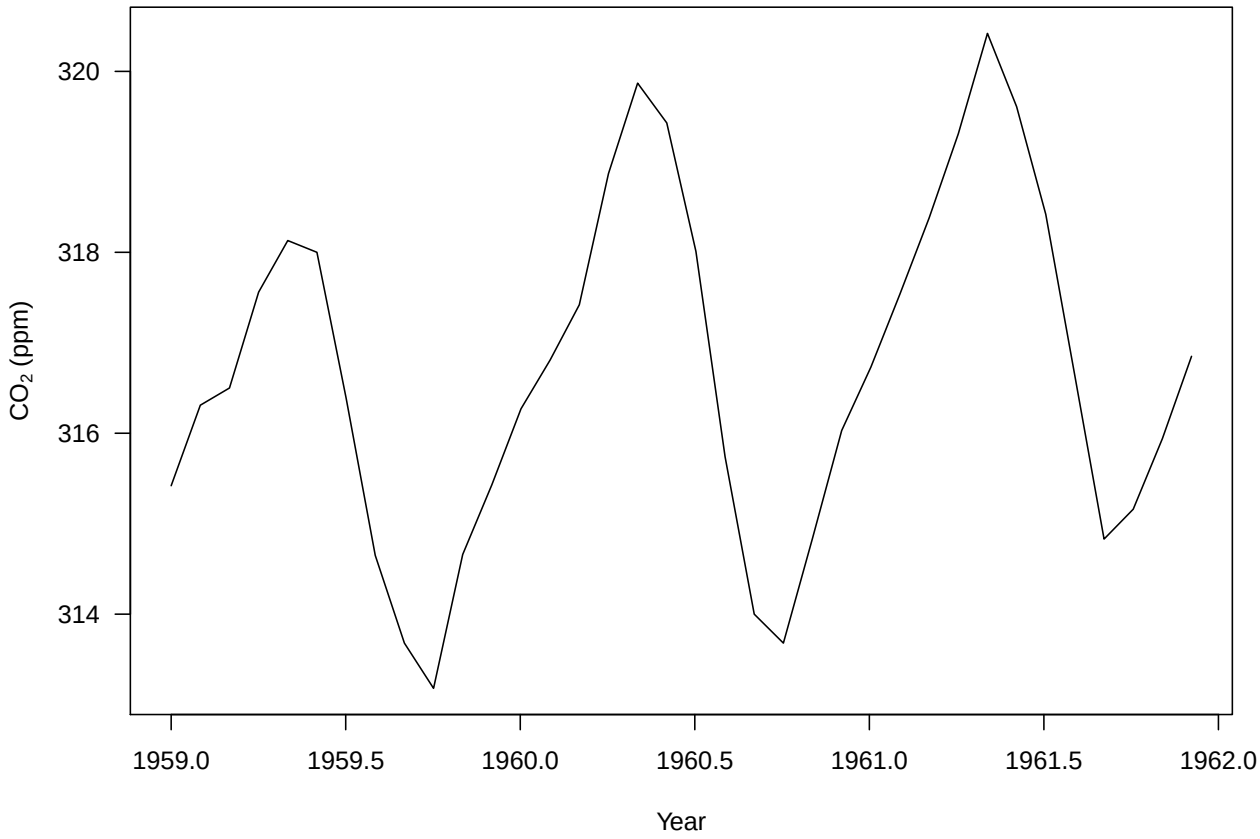
Visualizing Time Series Data: Mauna Loa Atmospheric CO2 Concentration

Atmospheric concentrations of CO_2 are expressed in parts per million (ppm) and reported in the preliminary 1997 SIO manometric mole fraction scale.

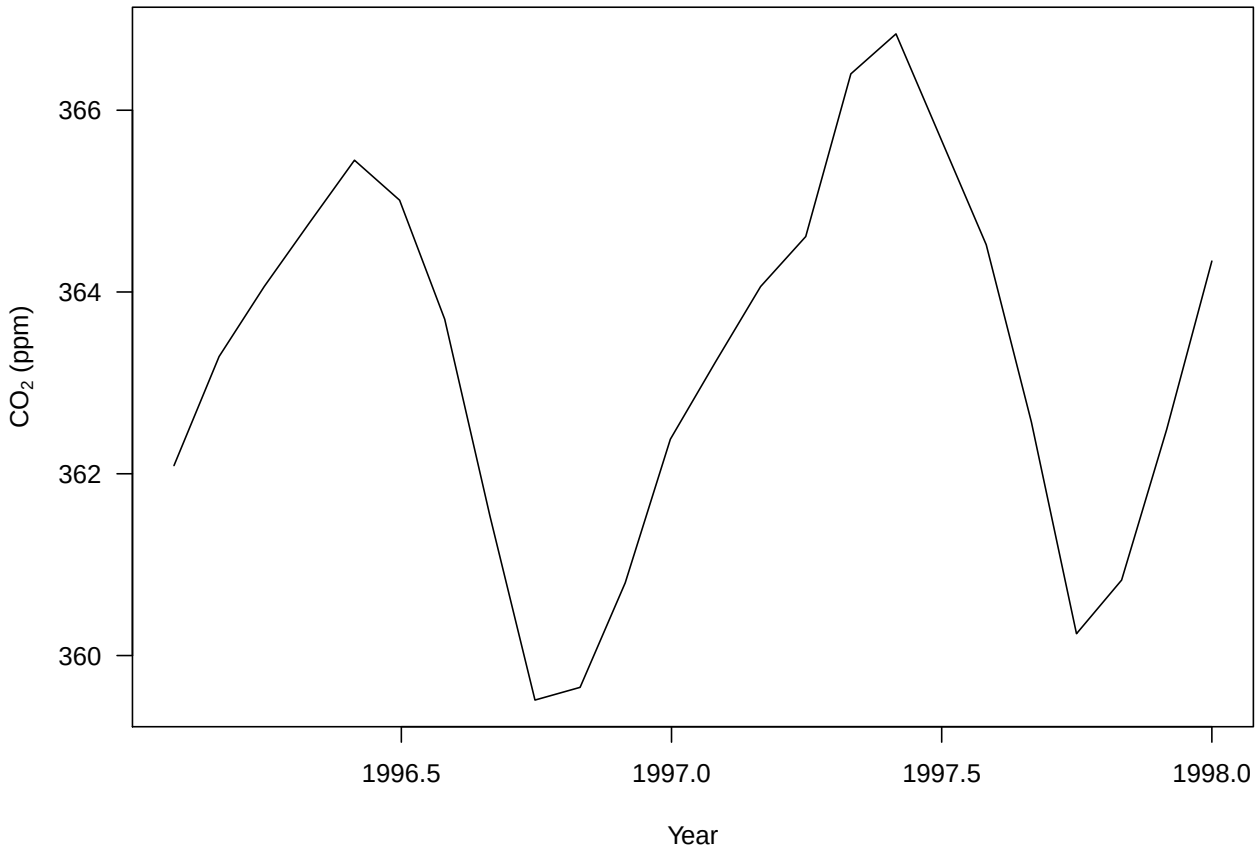
```
data("co2")
par(las = 1)
ts.plot(co2, ylab = expression(paste(CO[2], " (ppm)")))
```




```
time <- seq(1959, 1998, len = 468)
plot(time[1:36], co2[1:36], type = "l", xlab = "Year",
     ylab = expression(paste(CO[2], " (ppm)")),
     las = 1)
```




```
plot(time[445:468], co2[445:468], type = "l", xlab = "Year",
     ylab = expression(paste(CO[2], " (ppm)")),
     las = 1)
```



Visualizing Cross-Sectional Data

```
library(maps)
```

```
##
## Attaching package: 'maps'
```

```
## The following object is masked from 'package:purrr':
##
##     map
```

```
library(ggmap)
```

```
## Google's Terms of Service: https://cloud.google.com/maps-platform/terms/.
```

```
## Please cite ggmap if you use it! See citation("ggmap") for details.
```

```
data("USArrests")
USArrests$region <- tolower(row.names(USArrests))

statesMap <- map_data("state")
str(statesMap)
```

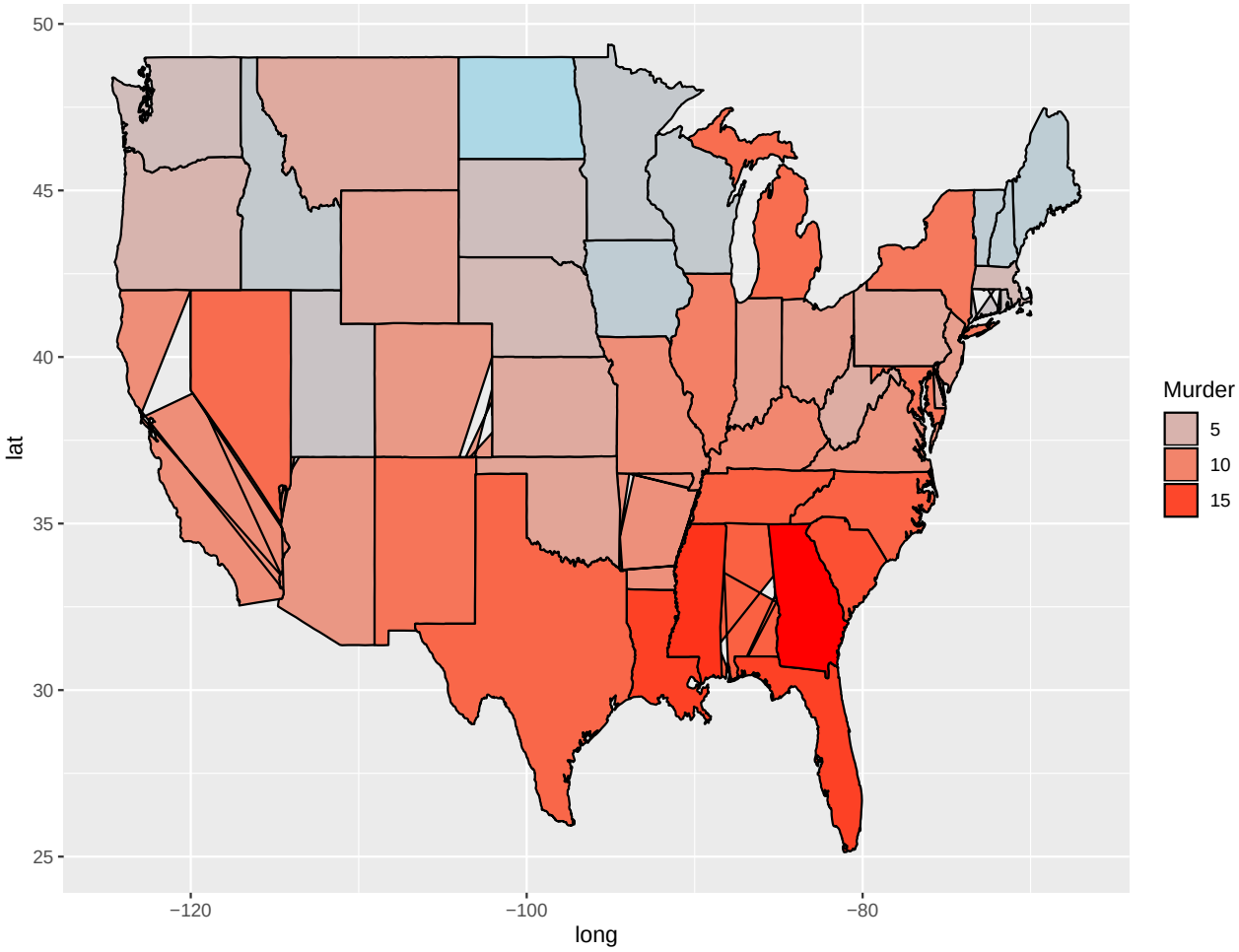
```
## 'data.frame':    15537 obs. of  6 variables:
##  $ long     : num  -87.5 -87.5 -87.5 -87.5 -87.6 ...
##  $ lat      : num  30.4 30.4 30.4 30.3 30.3 ...
##  $ group    : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ order    : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ region   : chr  "alabama" "alabama" "alabama" "alabama" ...
```

```
##  $ subregion: chr  NA NA NA NA ...
```

```
murderMap <- merge(statesMap, USArrests, by = "region")
str(murderMap)
```

```
## 'data.frame':    15527 obs. of  10 variables:
##  $ region   : chr  "alabama" "alabama" "alabama" "alabama" ...
##  $ long     : num  -87.5 -87.5 -88 -88 -88 ...
##  $ lat      : num  30.4 30.4 30.2 30.2 30.3 ...
##  $ group    : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ order    : int  1 2 13 14 15 3 4 5 6 7 ...
##  $ subregion: chr  NA NA NA NA ...
##  $ Murder   : num  13.2 13.2 13.2 13.2 13.2 13.2 13.2 13.2 13.2 13.2 ...
##  $ Assault  : int  236 236 236 236 236 236 236 236 236 236 ...
##  $ UrbanPop : int  58 58 58 58 58 58 58 58 58 58 ...
##  $ Rape     : num  21.2 21.2 21.2 21.2 21.2 21.2 21.2 21.2 21.2 21.2 ...
```

```
ggplot(murderMap, aes(x = long, y = lat, group = group, fill = Murder)) + geom_polygon(color = "black")
```



Visualizing Spatio-Temporal Data: ERA-Interim

The ERA-Interim is a global atmospheric reanalysis dataset. Reanalysis is an approach to produce spatially and temporally gridded datasets via data assimilation for climate monitoring and analysis.

```r
load("ERA_tmx_2010_JanFeb.RData")
library(fields)
```

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

```
## Loading required package: dotCall64
```

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

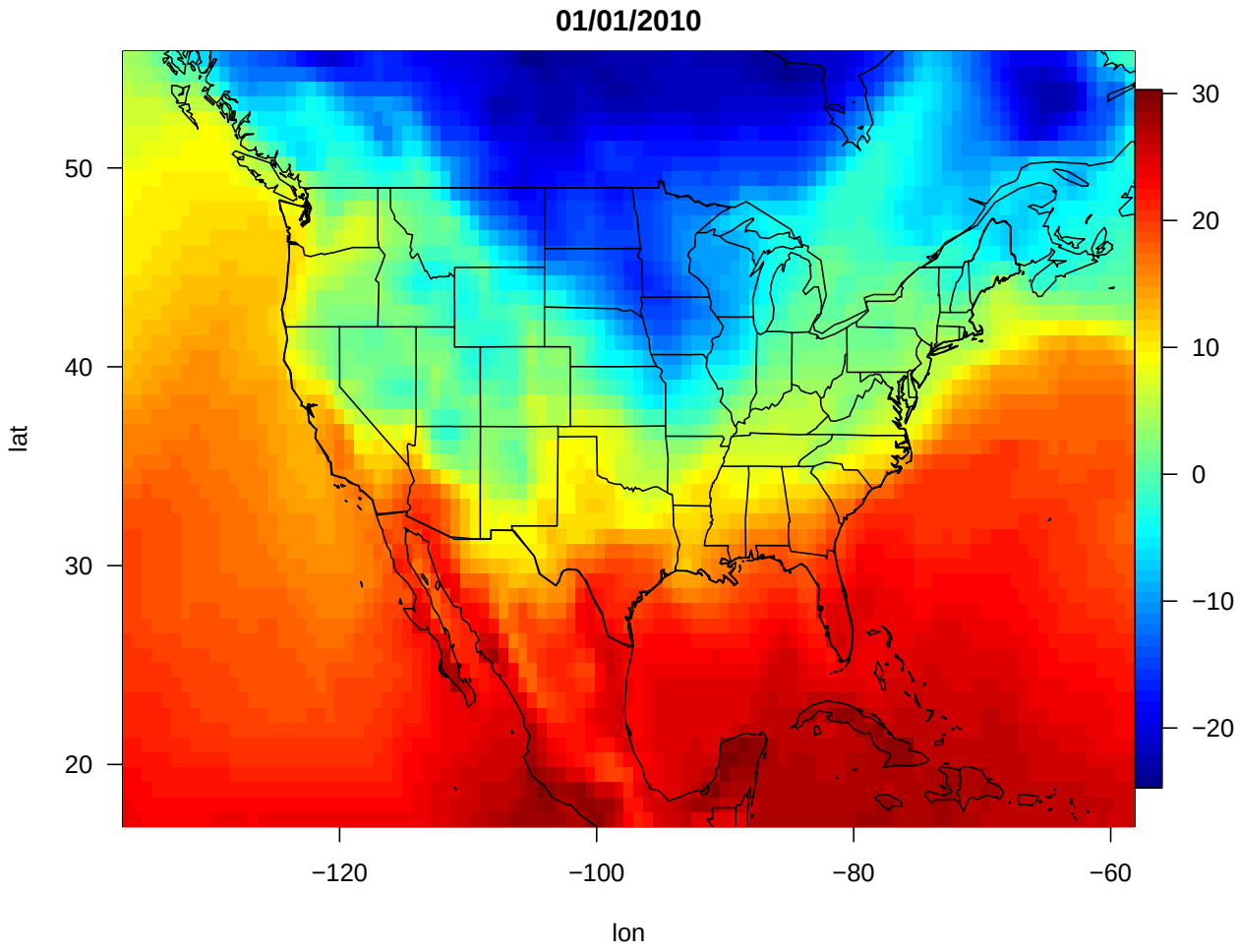
```
## Spam version 2.4-0 (2019-11-01) is loaded.
## Type 'help( Spam)' or 'demo( spam)' for a short introduction
## and overview of this package.
## Help for individual functions is also obtained by adding the
## suffix '.spam' to the function name, e.g. 'help( chol.spam)'.
```

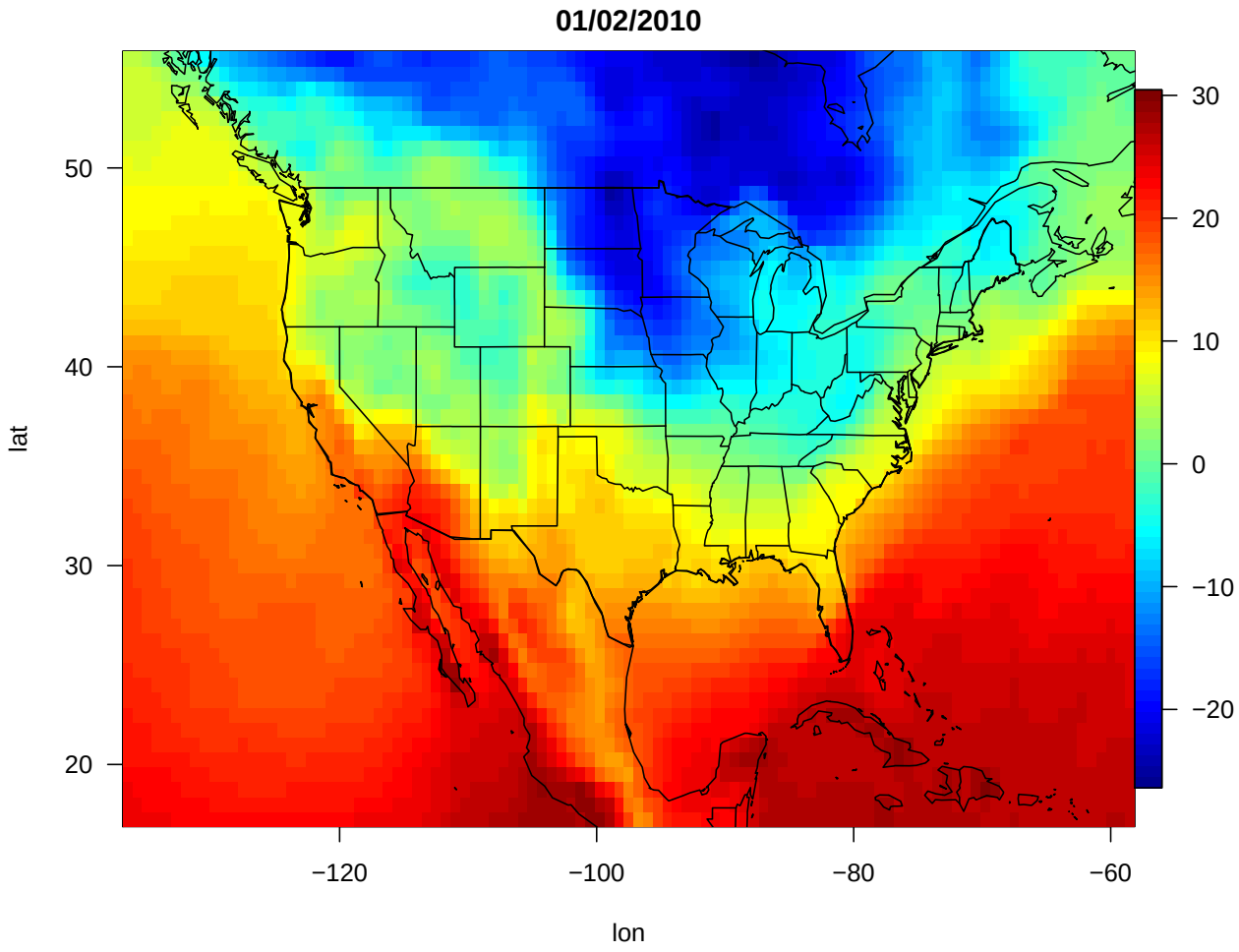
```
##
## Attaching package: 'spam'
```

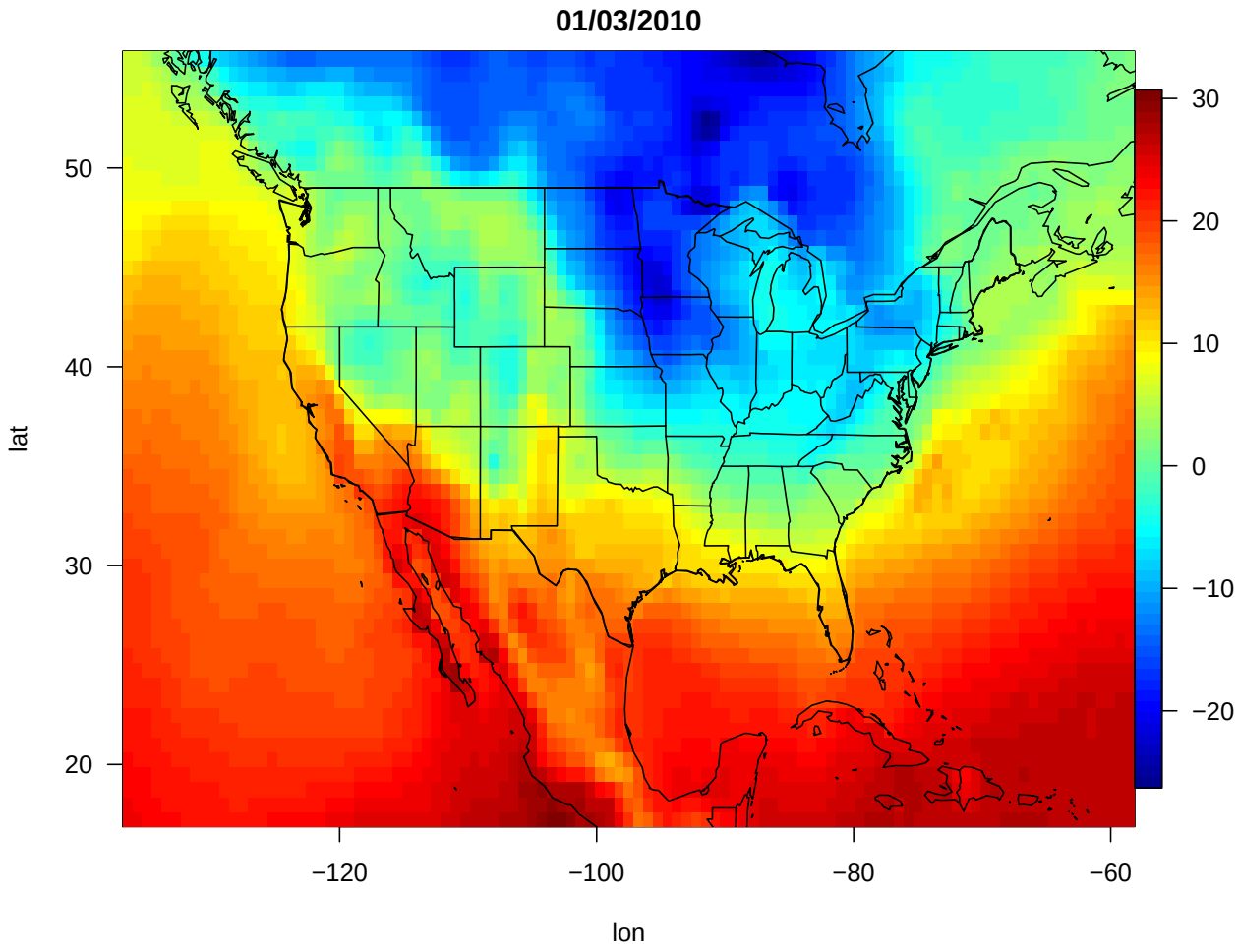
```
## The following objects are masked from 'package:base':
##
##     backsolve, forwardsolve
```

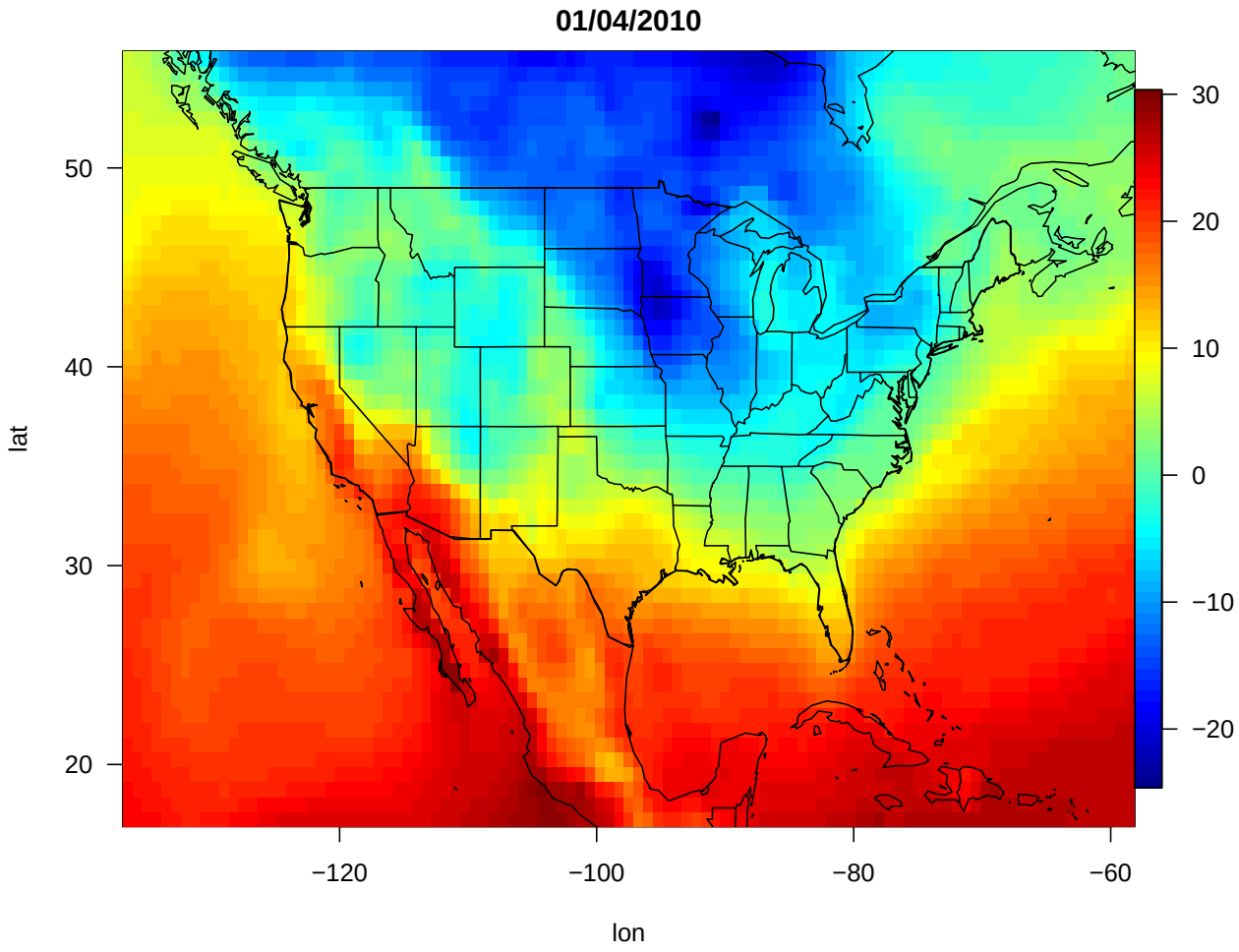
```
## See https://github.com/NCAR/Fields for
##  an extensive vignette, other supplements and source code
```

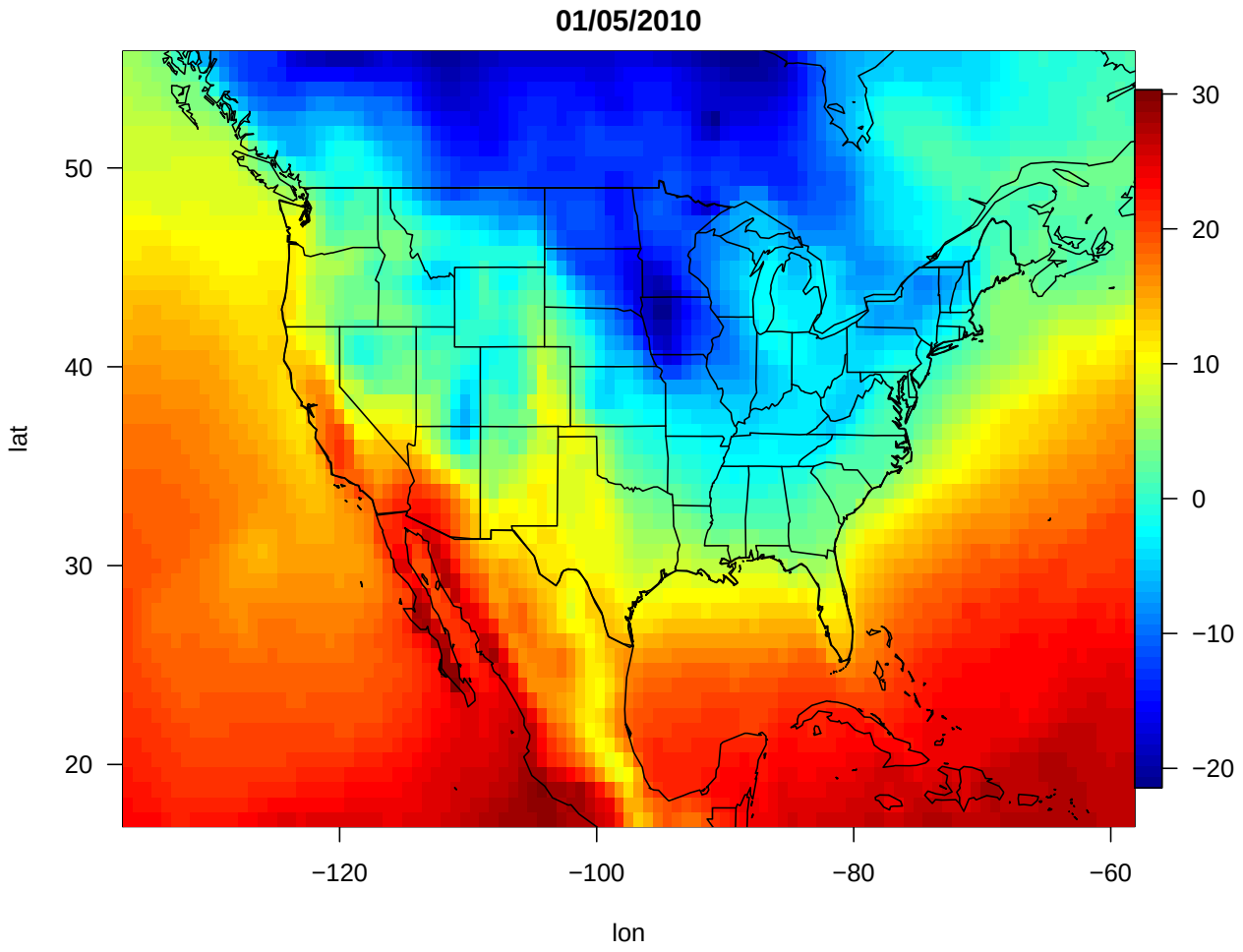
```r
par(mar = c(4.6, 4.1, 2.1, 0))
for (i in seq(1:5)){
        image.plot(lon, lat, tmx_dat[,, i], las = 1, main = format(day[i], "%m/%d/%Y"))
        map("state", xlim = range(lon), ylim = range(lat),
            add = T)
        map("world", xlim = range(lon), ylim = range(lat),
            add = T)
}
```


01/01/2010
50
40
30
20
lat
−120
−100
−80
−60
lon
30
20
10
0
−10
−20


01/02/2010
50
40
30
20
lat
−120
−100
−80
−60
lon
30
20
10
0
−10
−20


01/03/2010
50
40
30
20
lat
−120
−100
−80
−60
lon
30
20
10
0
−10
−20


01/04/2010
50
40
30
20
lat
−120
−100
−80
−60
lon
30
20
10
0
−10
−20




```
library(animation)
saveLatex({
  for (i in 1:58){
          image.plot(lon, lat, tmx_dat[,, i], las = 1, main = format(day[i], "%m/%d/%Y"),
                     zlim = range(tmx_dat))
          map("state", xlim = range(lon), ylim = range(lat), add = T)
          map("world", xlim = range(lon), ylim = range(lat), add = T)
  }
}, img.name = "ERA_Tmax", ani.opts = "controls,width=0.975\\textwidth",
latex.filename = ifelse(interactive(), "ERA_TMX_JanFeb.tex", ""),
interval = 0.5, nmax = 58, ani.dev = "pdf", ani.type = "pdf", ani.width = 8,
ani.height = 6, documentclass = paste("\\documentclass{article}",
                                      "\\usepackage[papersize={8in,6in},margin=0.1in]{geometry}",
                                      sep = "\n"))
```