week_7.R
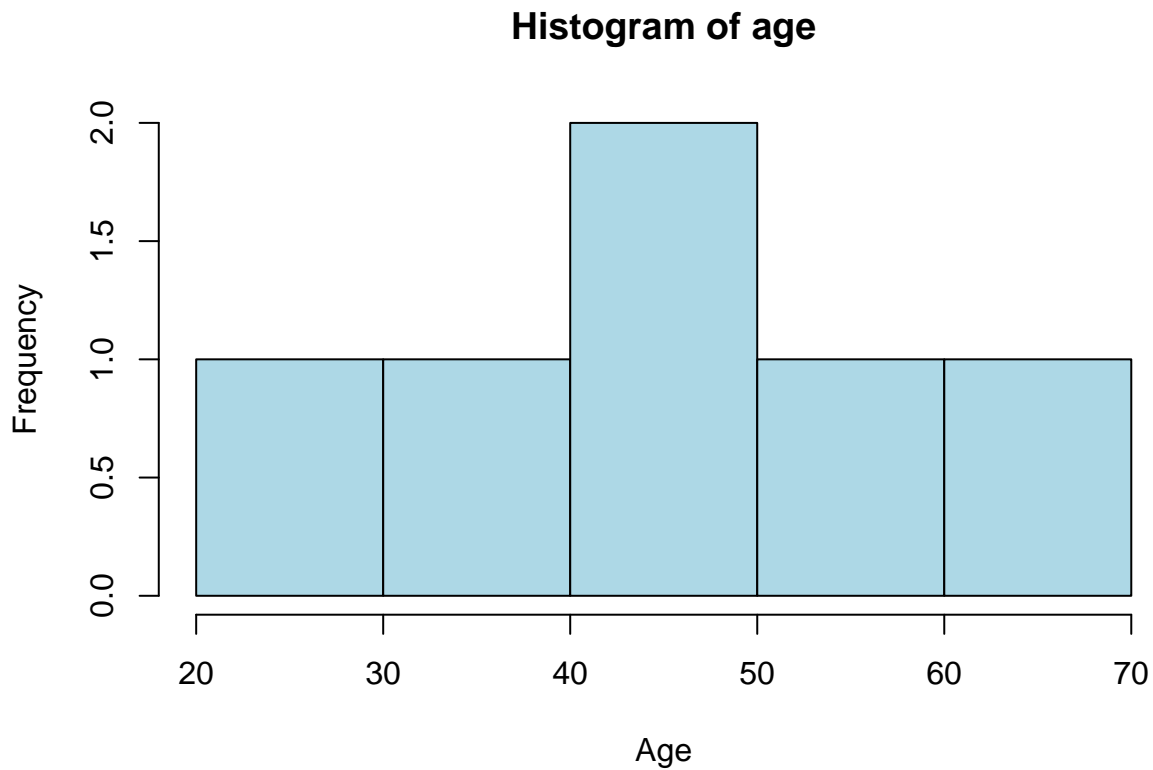
joebrew

Thu Feb 19 16:36:25 2015

```r
setwd('/home/joebrew/Documents/uf/phc7065')
dat <- read.csv('week_7_data.csv')

hist(dat$age, main = "Histogram of age",
     xlab = "Age", col = "lightblue")
```
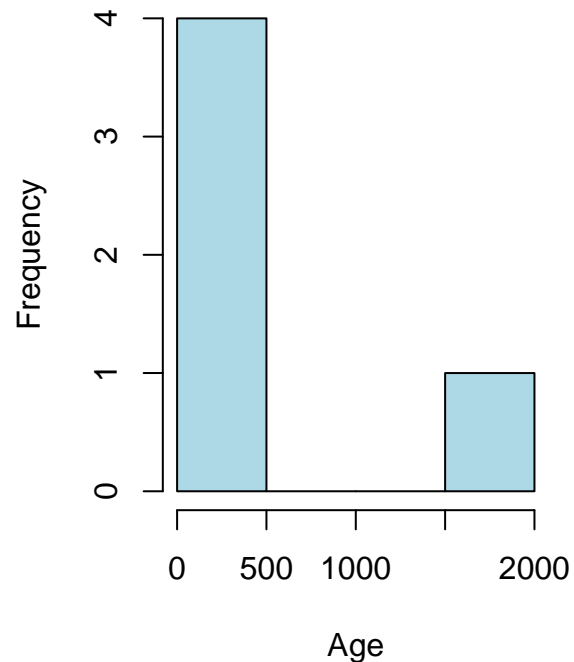

Histogram of age

```r
par(mfrow = c(1,2))
hist(dat$donations2014, main = "Histogram of 2014 donations",
     xlab = "Age", col = "lightblue")

hist(dat$donations2015, main = "Histogram of 2015 donations",
     xlab = "Age", col = "lightblue")
```

Histogram of 2014 donations
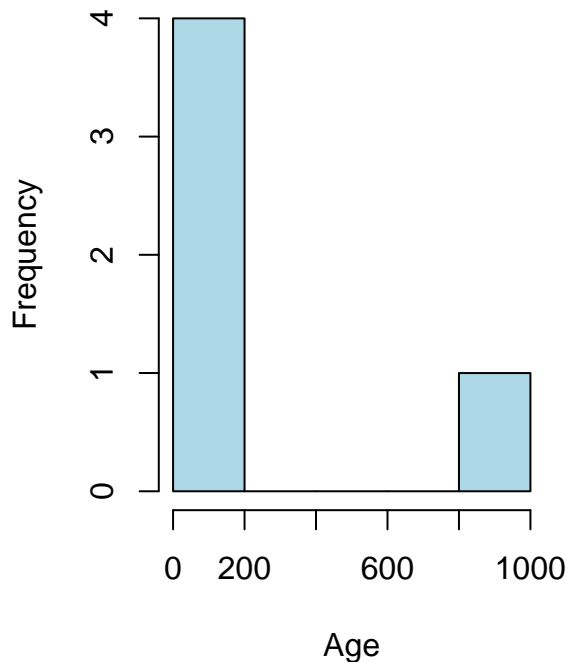
Histogram of 2015 donations


```r
par(mfrow = c(1,1))

x <- data.frame(donations2014 = dat$donations2014,
                donations2015 = dat$donations2015)
scatterBarNorm <- function(x, dcol="blue", lhist=20, num.dnorm=5*lhist, ...){
  x <- x[which(!is.na(x[,1]) & !is.na(x[,2])),]
  ## check input
  stopifnot(ncol(x)==2)
  ## set up layout and graphical parameters
  layMat <- matrix(c(2,0,1,3), ncol=2, byrow=TRUE)
  layout(layMat, widths=c(5/7, 2/7), heights=c(2/7, 5/7))
  ospc <- 0.5 # outer space
  pext <- 4 # par extension down and to the left
  bspc <- 1 # space between scatter plot and bar plots
  par. <- par(mar=c(pext, pext, bspc, bspc),
              oma=rep(ospc, 4)) # plot parameters
  ## scatter plot
  plot(x, xlim=range(x[,1]), ylim=range(x[,2]), ...)
  ## 3) determine barplot and height parameter
  ## histogram (for barplot-ting the density)
  xhist <- hist(x[,1], plot=FALSE, breaks = 20)
#               breaks=seq(from=min(x[,1]), to=max(x[,1]),
#                                     length.out=lhist))
  yhist <- hist(x[,2], plot=FALSE, breaks = 20)
#               breaks=seq(from=min(x[,2]), to=max(x[,2]),
#                                     length.out=lhist)) # note: this uses probability=TRUE
  ## determine the plot range and all the things needed for the barplots and lines
  xx <- seq(min(x[,1]), max(x[,1]), length.out=num.dnorm) # evaluation points for the overlaid density
  xy <- dnorm(xx, mean=mean(x[,1]), sd=sd(x[,1])) # density points
```

```
  yx <- seq(min(x[,2]), max(x[,2]), length.out=num.dnorm)
  yy <- dnorm(yx, mean=mean(x[,2]), sd=sd(x[,2]))
  ## barplot and line for x (top)
  par(mar=c(0, pext, 0, 0))
  barplot(xhist$density, axes=FALSE, ylim=c(0, max(xhist$density, xy)),
          space=0, col = adjustcolor("red", alpha.f = 0.6)) # barplot
  lines(seq(from=0, to=lhist-1, length.out=num.dnorm), xy, col="red") # line
  ## barplot and line for y (right)
  par(mar=c(pext, 0, 0, 0))
  barplot(yhist$density, axes=FALSE, xlim=c(0, max(yhist$density, yy)),
          space=0,col = adjustcolor("blue", alpha.f = 0.6), horiz=TRUE) # barplot
  lines(yy, seq(from=0, to=lhist-1, length.out=num.dnorm), col="blue") # line
  ## restore parameters
  par(par.)
}

require(mvtnorm)
```
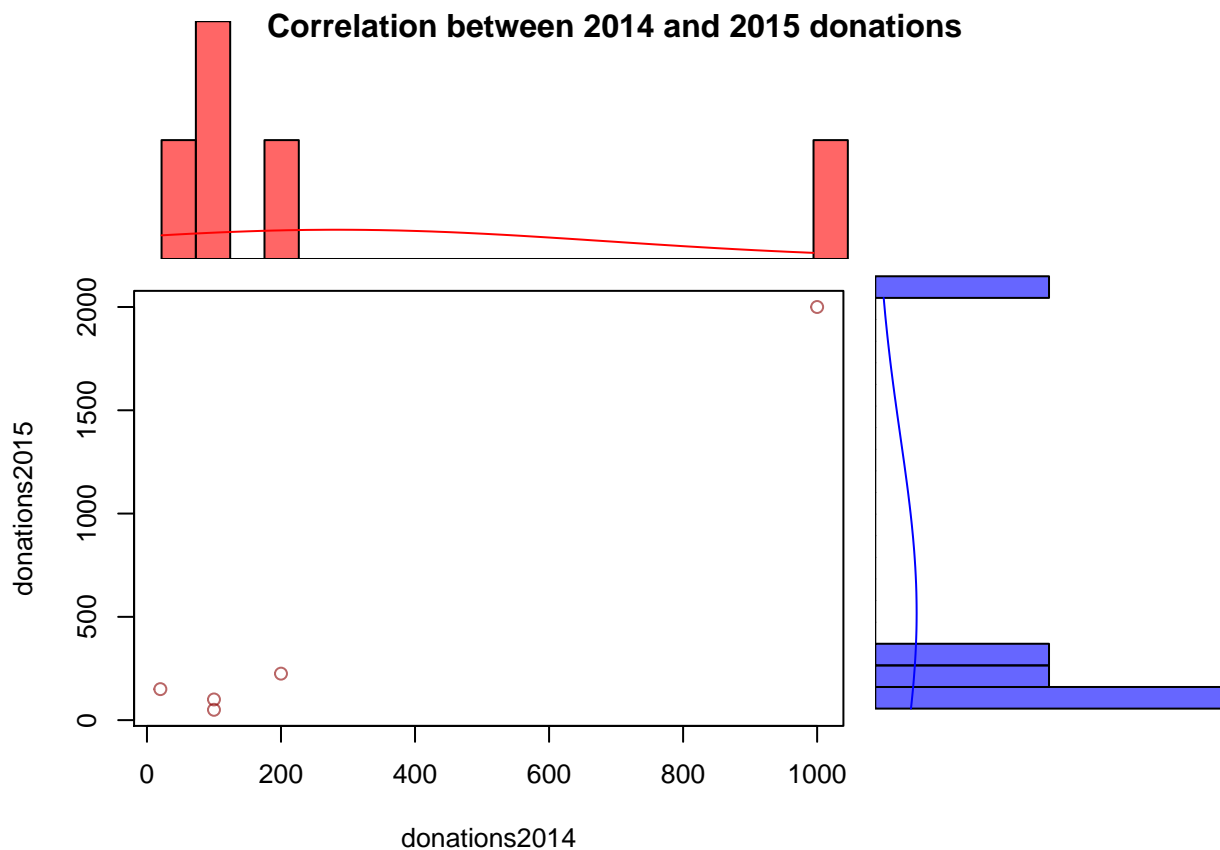
```
## Loading required package: mvtnorm
```

```
scatterBarNorm(x, col = adjustcolor("darkred", alpha.f = 0.6))
title(main = "Correlation between 2014 and 2015 donations", outer = TRUE, line = -1)

library(ggmap)
```

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

```
dat$country <- gsub("[.]| of ", "", dat$country)
temp <- geocode(as.character(paste0(dat$address, ", ",dat$country)))
```

```
## Information from URL : http://maps.googleapis.com/maps/api/geocode/json?address=Webster+Groves,+St.+
## Google Maps API Terms of Service : http://developers.google.com/maps/terms
## Information from URL : http://maps.googleapis.com/maps/api/geocode/json?address=Saint+Germain,+Tunis
## Google Maps API Terms of Service : http://developers.google.com/maps/terms
## Information from URL : http://maps.googleapis.com/maps/api/geocode/json?address=Maryland+Heights,+Ka
## Google Maps API Terms of Service : http://developers.google.com/maps/terms
## Information from URL : http://maps.googleapis.com/maps/api/geocode/json?address=Coon+Rapids,+Ohio,++
## Google Maps API Terms of Service : http://developers.google.com/maps/terms
## Information from URL : http://maps.googleapis.com/maps/api/geocode/json?address=O'Fallon,+Illinois,+
## Google Maps API Terms of Service : http://developers.google.com/maps/terms
## Information from URL : http://maps.googleapis.com/maps/api/geocode/json?address=+,++Germany&sensor=fa
## Google Maps API Terms of Service : http://developers.google.com/maps/terms
```

```
library(maps)
par(mar = c(0,0,0,0))
par(oma = c(0,0,0,0))
par(mfrow = c(1,1))
map("world", fill = TRUE, col = "grey", border = FALSE)
points(temp$lon, temp$lat, col = adjustcolor("darkred", alpha.f = 0.8),
       pch = 17, cex = 2)
title(main = "Location of observations in the dataset")
```


Location of observations in the dataset