DSC640 WEEK 9 & 10

Cindy Herrera

2/7/2020

5.2 Exercises: Heat Maps, Spatial Charts, And Contour Charts

TYou need to submit 3 heat maps, 3 spatial charts and 3 contour charts using Tableau or PowerBI, Python and R using the data below (or your own datasets). You can also use D3. You can choose which library to use in Python or R, documentation is provided to help you decide and as you start to play around in the libraries, you will decide which you prefer.

You may also download them directly from this link: Exercise 5.2 Datasets https://content.bellevue.edu/cst/dsc/640/datasets/ex5-2.zip

```
library(pheatmap)
library(ggplot2)
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

```
library(contoureR)
```

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

```
library(ContourFunctions)
library(maps)
library(meltt)
library(reshape2)
library(scales)
library(plyr)
```

```
##
## Attaching package: 'plyr'
```

```
## The following object is masked from 'package:maps':
##
##     ozone
```

```
## The following objects are masked from 'package:plotly':
```

```
##
##     arrange, mutate, rename, summarise
```

```
library(rvest)
```

```
## Loading required package: xml2
```

```
library(tidyr)
```

```
##
## Attaching package: 'tidyr'
```

```
## The following object is masked from 'package:reshape2':
##
##     smiths
```

```
library(scatterplot3d)
library(corrgram)
```

```
## Registered S3 method overwritten by 'seriation':
##   method         from
##   reorder.hclust gclus
```

```
##
## Attaching package: 'corrgram'
```

```
## The following object is masked from 'package:plyr':
##
##     baseball
```

```
library(mapproj)
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.
```

```
##
## Attaching package: 'ggmap'
```

```
## The following object is masked from 'package:plotly':
##
##     wind
```

```
costcos <- read.csv("~/Desktop/00 data640/ex5-2/costcos-geocoded.csv")
str(costcos)
```

```
## 'data.frame':    417 obs. of  7 variables:
##  $ Address  : Factor w/ 416 levels "1 Industrial Lane",..: 49 255 387 316 243 276 269 268 341 124 ..
##  $ City     : Factor w/ 369 levels "Albany","Albuquerque",..: 139 137 210 147 7 7 256 330 330 117 ..
##  $ State    : Factor w/ 40 levels "Alabama","Alaska",..: 1 1 1 2 2 2 3 3 3 3 ...
##  $ Code     : Factor w/ 40 levels "AK","AL","AZ",..: 2 2 2 1 1 1 3 3 3 3 ...
##  $ Zip.Code : Factor w/ 415 levels "01089-4672","01923-1014",..: 115 114 116 415 414 413 239 238 237
##  $ Latitude : num  34.7 33.4 32.4 58.4 61.1 ...
##  $ Longitude: num  -86.6 -86.8 -86.2 -134.5 -149.9 ...
```

```
nba <- read.csv("~/Desktop/00 data640/ex5-2/ppg2008.csv", header = TRUE)
str(nba)
```

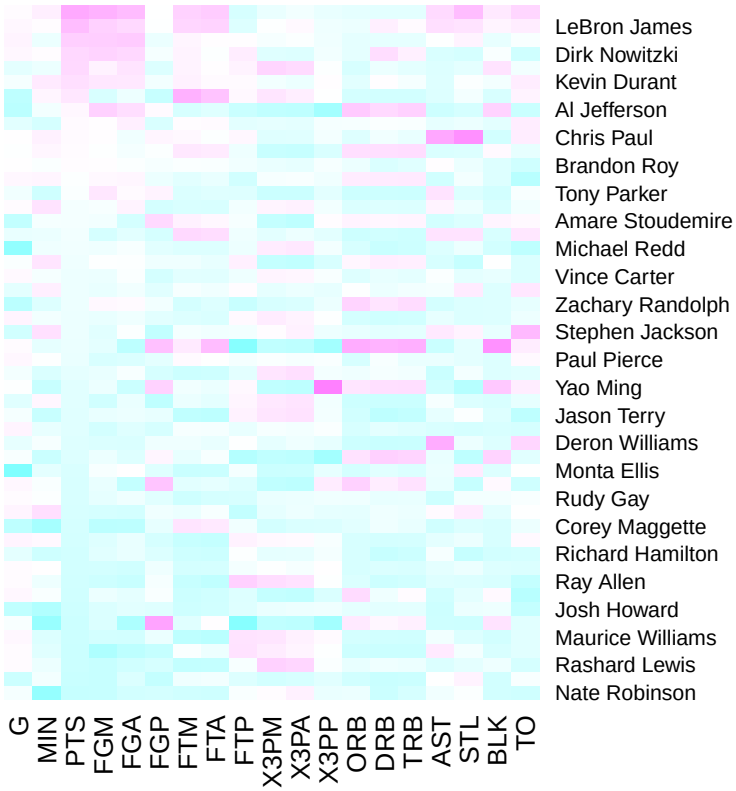
```
## 'data.frame':    50 obs. of  21 variables:
##  $ Name: Factor w/ 50 levels "Al Harrington ",..: 21 31 29 19 15 27 28 2 13 9 ...
##  $ G   : int  79 81 82 81 67 74 51 50 78 66 ...
```

```
##  $ MIN : num  38.6 37.7 36.2 37.7 36.2 39 38.2 36.6 38.5 34.5 ...
##  $ PTS : num  30.2 28.4 26.8 25.9 25.8 25.3 24.6 23.1 22.8 22.8 ...
##  $ FGM : num  10.8 9.7 9.8 9.6 8.5 8.9 6.7 9.7 8.1 8.1 ...
##  $ FGA : num  22 19.9 20.9 20 19.1 18.8 15.9 19.5 16.1 18.3 ...
##  $ FGP : num  0.491 0.489 0.467 0.479 0.447 0.476 0.42 0.497 0.503 0.443 ...
##  $ FTM : num  7.5 7.3 5.9 6 6 6.1 9 3.7 5.8 5.6 ...
##  $ FTA : num  9.8 9.4 6.9 6.7 6.9 7.1 10.3 5 6.7 7.1 ...
##  $ FTP : num  0.765 0.78 0.856 0.89 0.878 0.863 0.867 0.738 0.868 0.793 ...
##  $ X3PM: num  1.1 1.6 1.4 0.8 2.7 1.3 2.3 0 0.8 1 ...
##  $ X3PA: num  3.5 4.7 4.1 2.1 6.7 3.1 5.4 0.1 2.3 2.6 ...
##  $ X3PP: num  0.317 0.344 0.351 0.359 0.404 0.422 0.415 0 0.364 0.371 ...
##  $ ORB : num  1.1 1.3 1.1 1.1 0.7 1 0.6 3.4 0.9 1.6 ...
##  $ DRB : num  3.9 6.3 4.1 7.3 4.4 5.5 3 7.5 4.7 5.2 ...
##  $ TRB : num  5 7.6 5.2 8.4 5.1 6.5 3.6 11 5.5 6.8 ...
##  $ AST : num  7.5 7.2 4.9 2.4 2.7 2.8 2.7 1.6 11 3.4 ...
##  $ STL : num  2.2 1.7 1.5 0.8 1 1.3 1.2 0.8 2.8 1.1 ...
##  $ BLK : num  1.3 1.1 0.5 0.8 1.4 0.7 0.2 1.7 0.1 0.4 ...
##  $ TO  : num  3.4 3 2.6 1.9 2.5 3 2.9 1.8 3 3 ...
##  $ PF  : num  2.3 1.7 2.3 2.2 3.1 1.8 2.3 2.8 2.7 3 ...
```

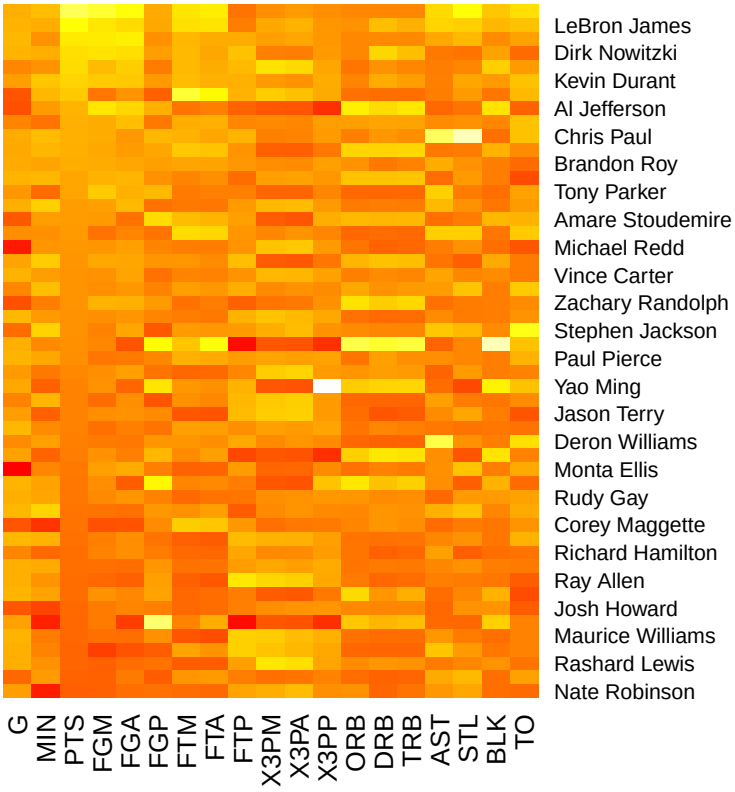
```
nba <- nba[order(nba$PTS),]
```

```
row.names(nba) <- nba$Name
nba <- nba[,2:20]
nba_matrix <- data.matrix(nba)
```

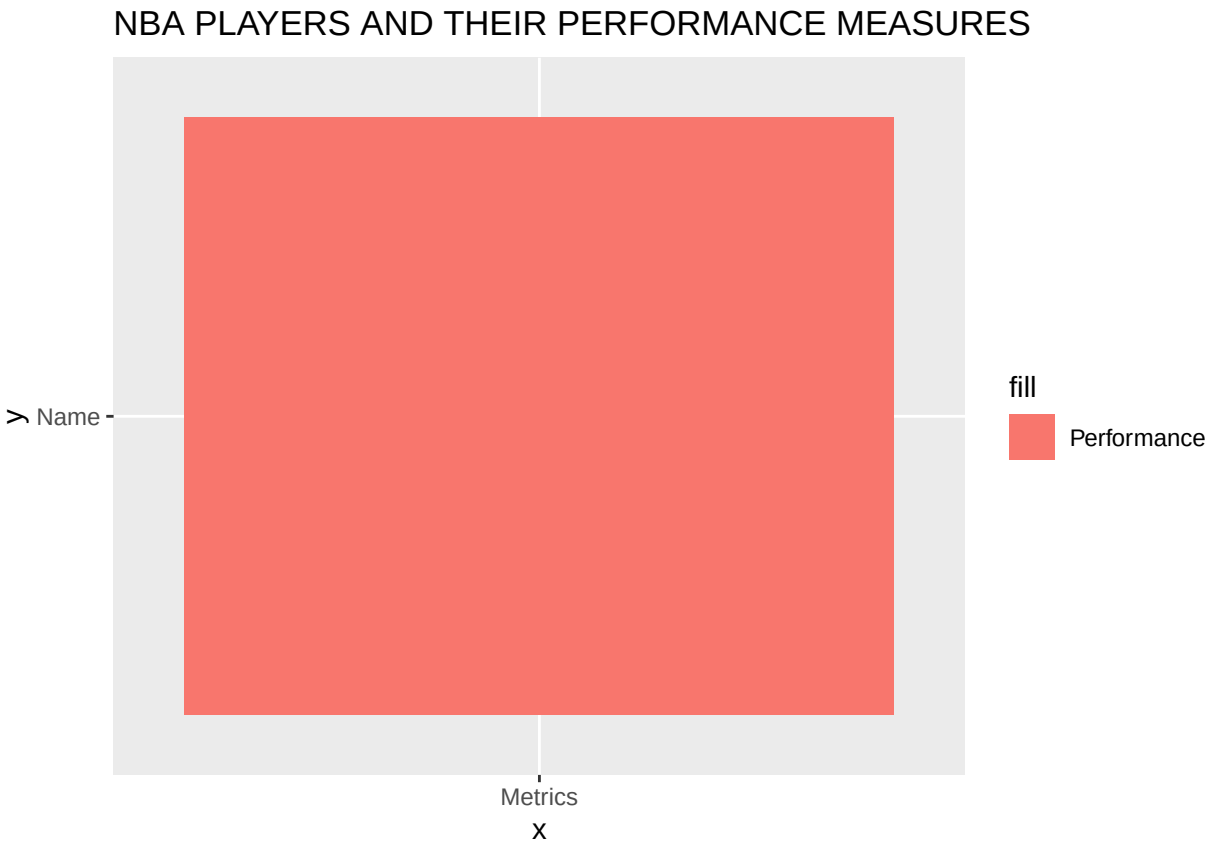
```
nba_heatmap <- heatmap(nba_matrix, Rowv = NA, Colv = NA, col = cm.colors(256), scale = "column", margins
```



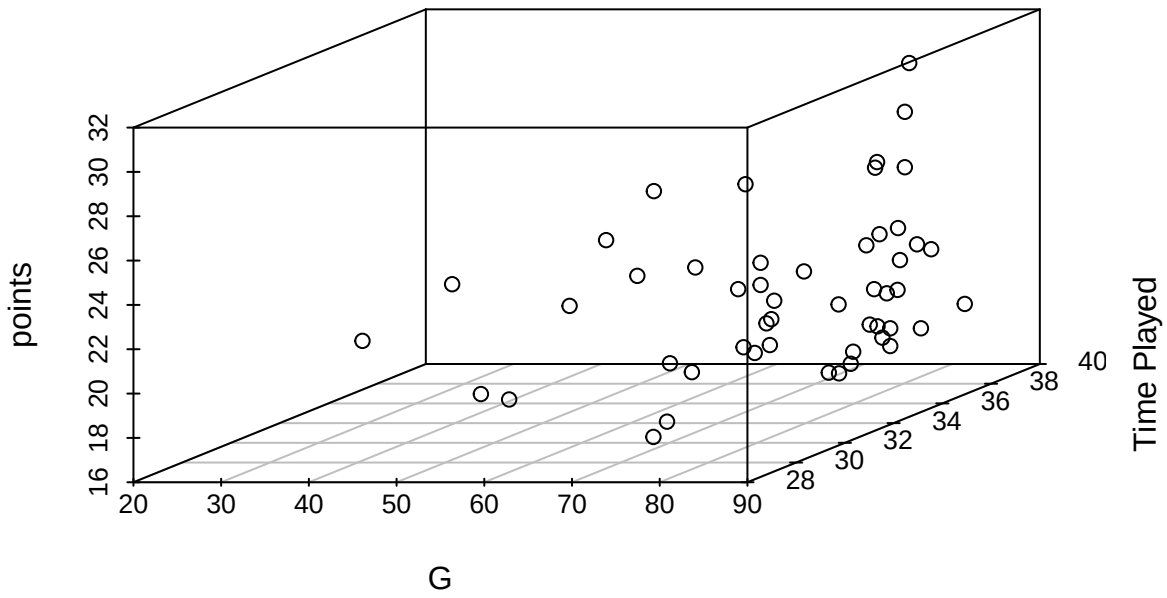
```
nba_heatmap <- heatmap(nba_matrix, Rowv = NA, Colv = NA, col = heat.colors(256), scale = "column", margi
```



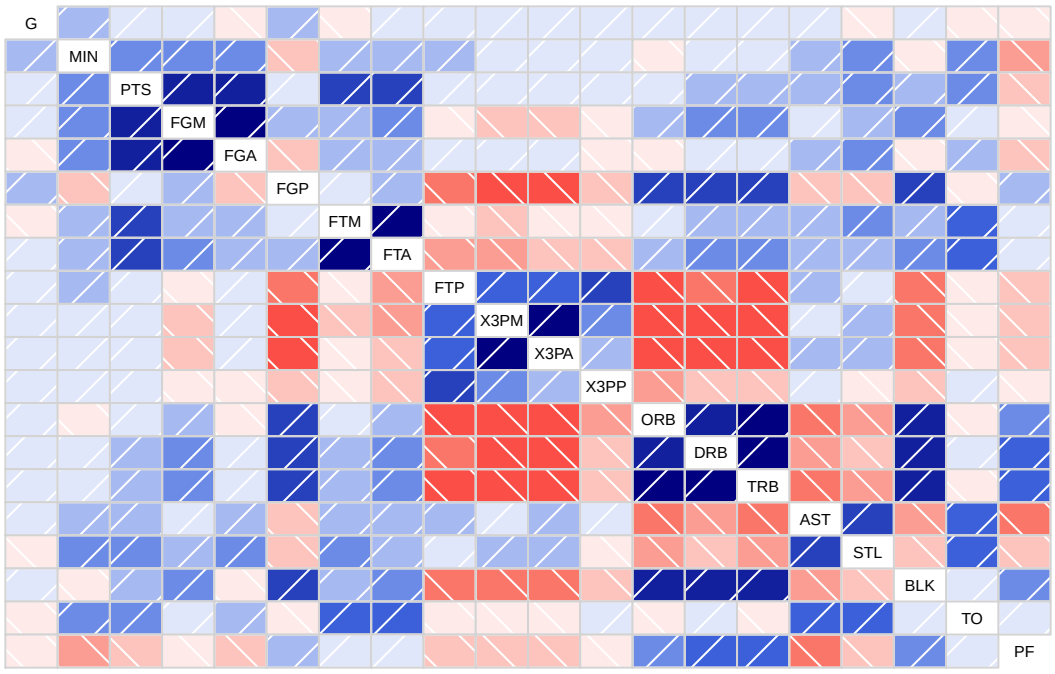

```
NBA <- read.csv("http://datasets.flowingdata.com/ppg2008.csv",sep = ",")

NBAlong <- NBA %>% gather(key = Metrics, value = Performance, G:PF)
## Joining all the metrics into a single var]

ggplot(data = NBAlong,aes(x = 'Metrics', y = 'Name')) +
  geom_tile(aes(fill = 'Performance')) +
  ggtitle("NBA PLAYERS AND THEIR PERFORMANCE MEASURES")
```




```
scatterplot3d(NBA$G,NBA$MIN,NBA$PTS,axis = T, grid = T, box = T, zlab = "points",
              ylab = "Time Played", xlab = "G")
```

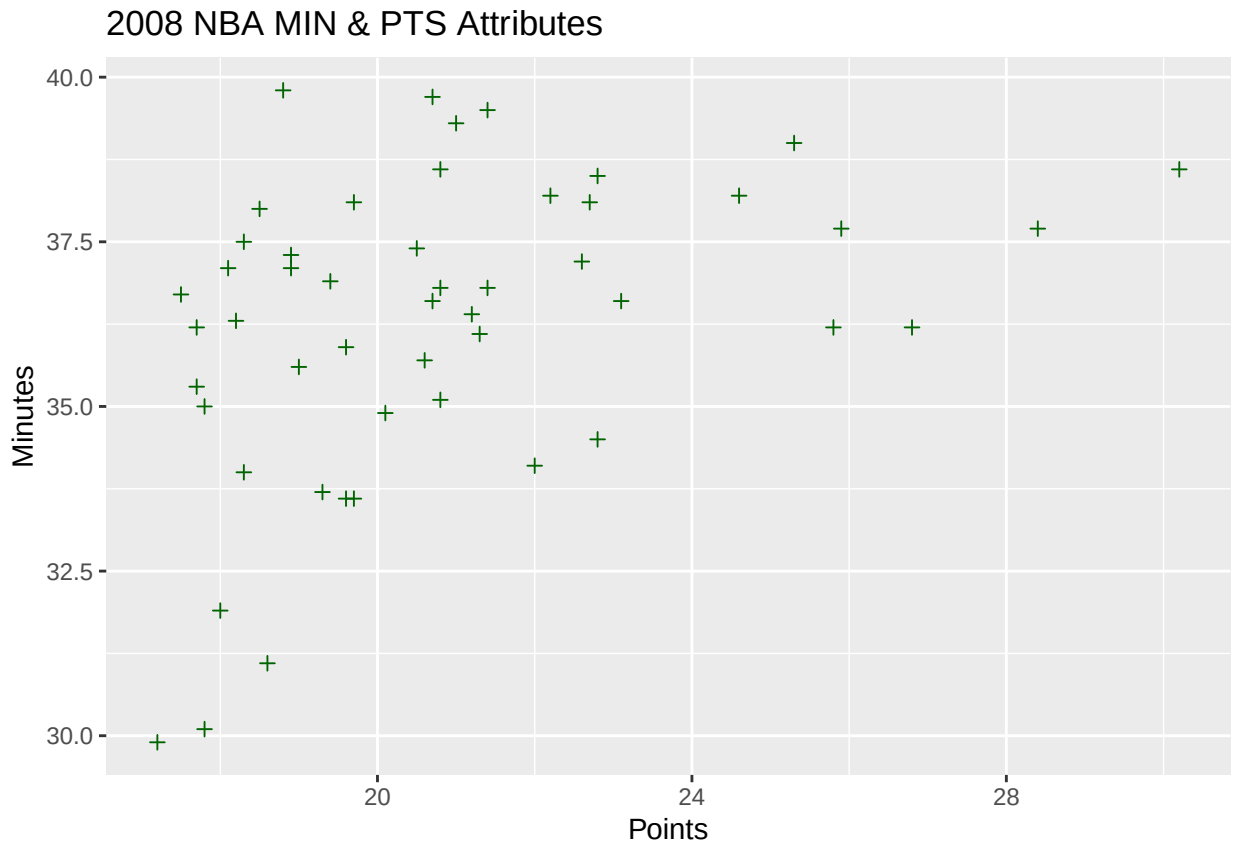



```
corrgram(NBA)
```

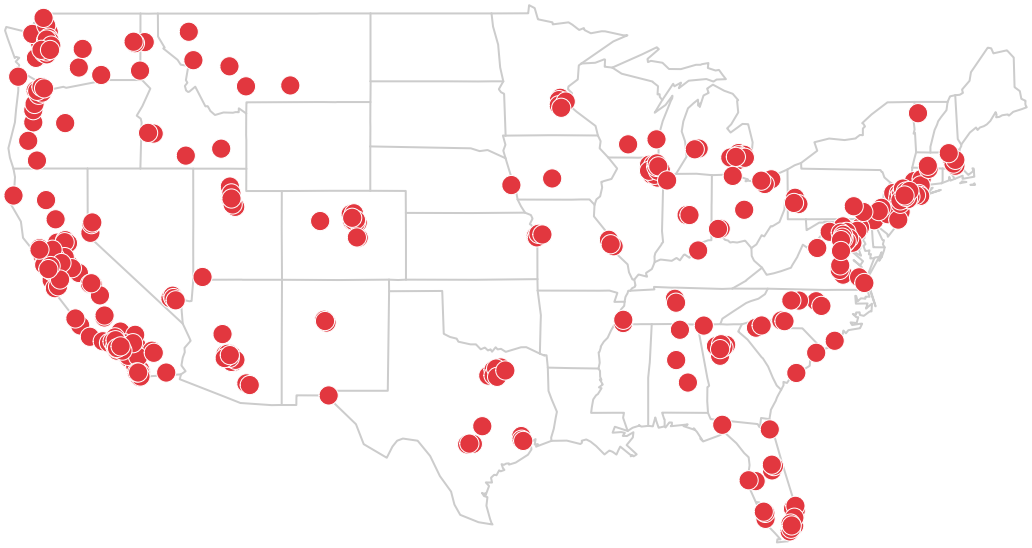


Just for practice...

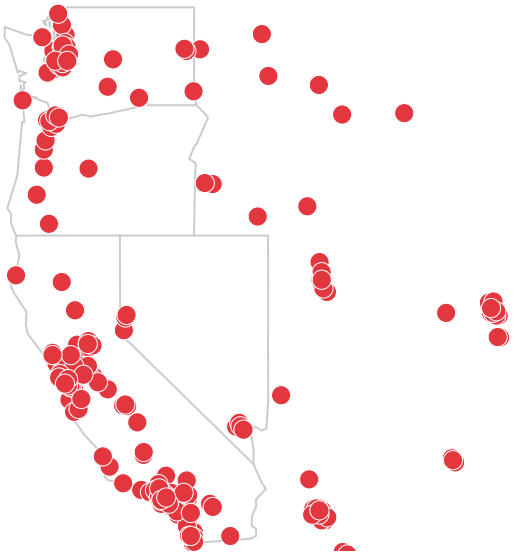
```
ggplot(data = nba) +
  geom_point(mapping = aes( x = nba$PTS, y = nba$MIN), col = "dark green",
             size = 1.5, pch = 3) +
  labs(title = "2008 NBA MIN & PTS Attributes", y = "Minutes", x = "Points")
```




```
map(database = 'state', col = '#cccccc')
symbols(costcos$Longitude, costcos$Latitude, bg = '#e2373f', fg = '#ffffff',
        lwd = 0.5, circles = rep(1,nrow(costcos)), inches = 0.05, add = TRUE)
```



```
map(database = 'state',region = c('California','Nevada','Oregon','Washington'),
    col = '#cccccc')
symbols(costcos$Longitude, costcos$Latitude, bg = '#e2373f', fg = '#ffffff',
        lwd = 0.5, circles = rep(1,nrow(costcos)), inches = 0.05, add = TRUE)
```



```
par(mfrow = c(2, 1))
map("usa")
```

```
cs <- count(costcos, "State")
head(cs)
```

```
##         State freq
## 1     Alabama    3
## 2      Alaska    3
## 3     Arizona   17
## 4  California  115
## 5    Colorado   12
## 6 Connecticut    5
```

```
data(costcos)
```

```
## Warning in data(costcos): data set 'costcos' not found
```

```
data(cs)
```

```
## Warning in data(cs): data set 'cs' not found
```

```
colors = c("#F1EEF6", "#D4B9DA", "#C994C7", "#DF65B0", "#DD1C77",
    "#980043")
head(cs)
```

```
##         State freq
## 1     Alabama    3
## 2      Alaska    3
## 3     Arizona   17
## 4  California  115
## 5    Colorado   12
```

```
## 6 Connecticut    5
```

```
head(costcos)
```

```
##                     Address       City   State Code   Zip.Code Latitude
## 1  1205 N. Memorial Parkway Huntsville Alabama   AL 35801-5930 34.74309
## 2      3650 Galleria Circle     Hoover Alabama   AL 35244-2346 33.37765
## 3    8251 Eastchase Parkway Montgomery Alabama   AL      36117 32.36389
## 4 5225 Commercial Boulevard     Juneau  Alaska   AK 99801-7210 58.35920
## 5      330 West Dimond Blvd  Anchorage  Alaska   AK 99515-1950 61.14327
## 6          4125 DeBarr Road  Anchorage  Alaska   AK 99508-3115 61.21081
##    Longitude
## 1  -86.60096
## 2  -86.81242
## 3  -86.15088
## 4 -134.48300
## 5 -149.88422
## 6 -149.80434
```

Stores per State

Here using mapproj displaying the number of stores per state by using the counts created. Adding them to display by contour shading

```
colorsmatched <- cs$colorBuckets[match(costcos$State, cs$State)]

map("state", col = colors[colorsmatched], fill = TRUE, resolution = 0,
    lty = 0, projection = "polyconic")

map("state", col = "white", fill = FALSE, add = TRUE, lty = 1, lwd = 0.2,
    projection = "polyconic")
title("Number of Costco Stores Per State")

leg.txt <- c("<4", "4-8", "8-12", "12-16", "16-20", ">20")
legend("topright", leg.txt, horiz = TRUE, fill = colors)
```

Number of Costco Stores Per State