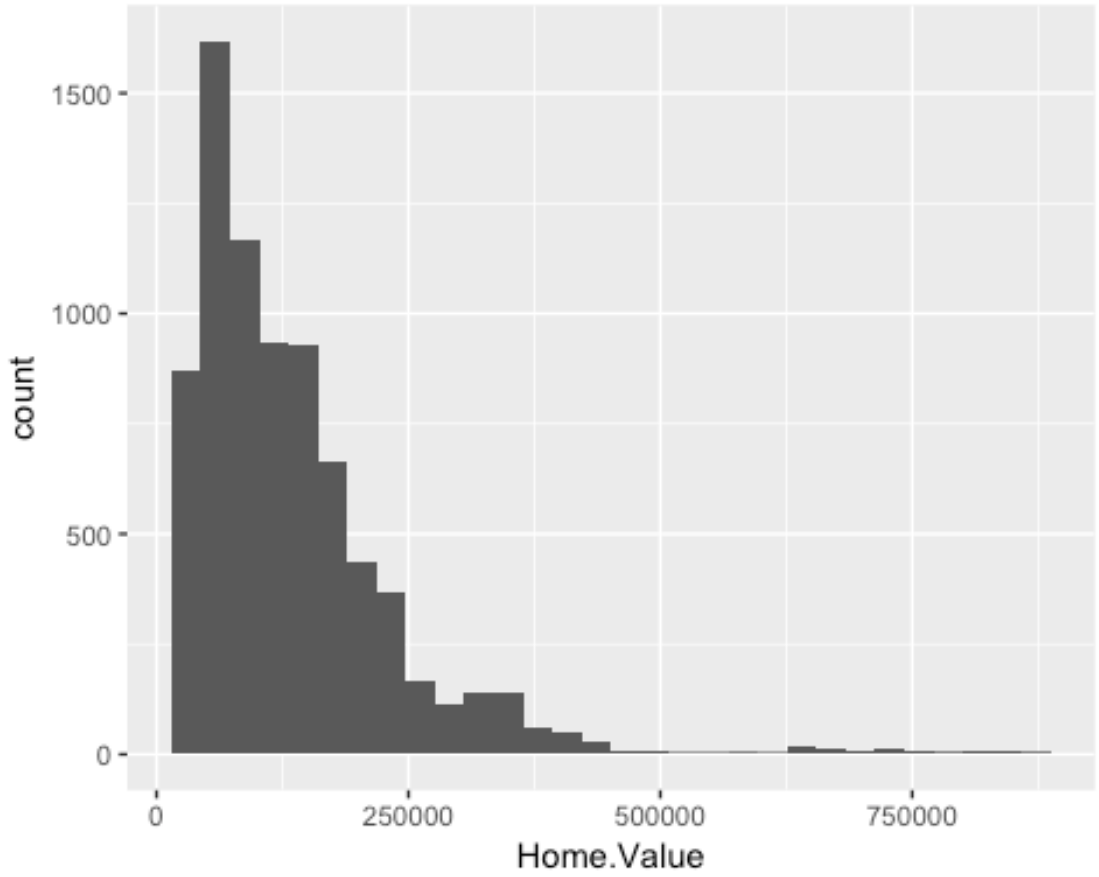
Data Visualization GGplot

```
# GGPLOT2
# A REPORT ON HOW PROFESSIONAL GRAPHICS GGPLOT WORKS IN R.
# Author: Jayendra Bhardwaj

#Lets have a look on Housing datasets.
setwd("/Users/jayendra/Desktop/DAV")
housing <- read.csv('landdata-states.csv')
head(housing[1:5])
```

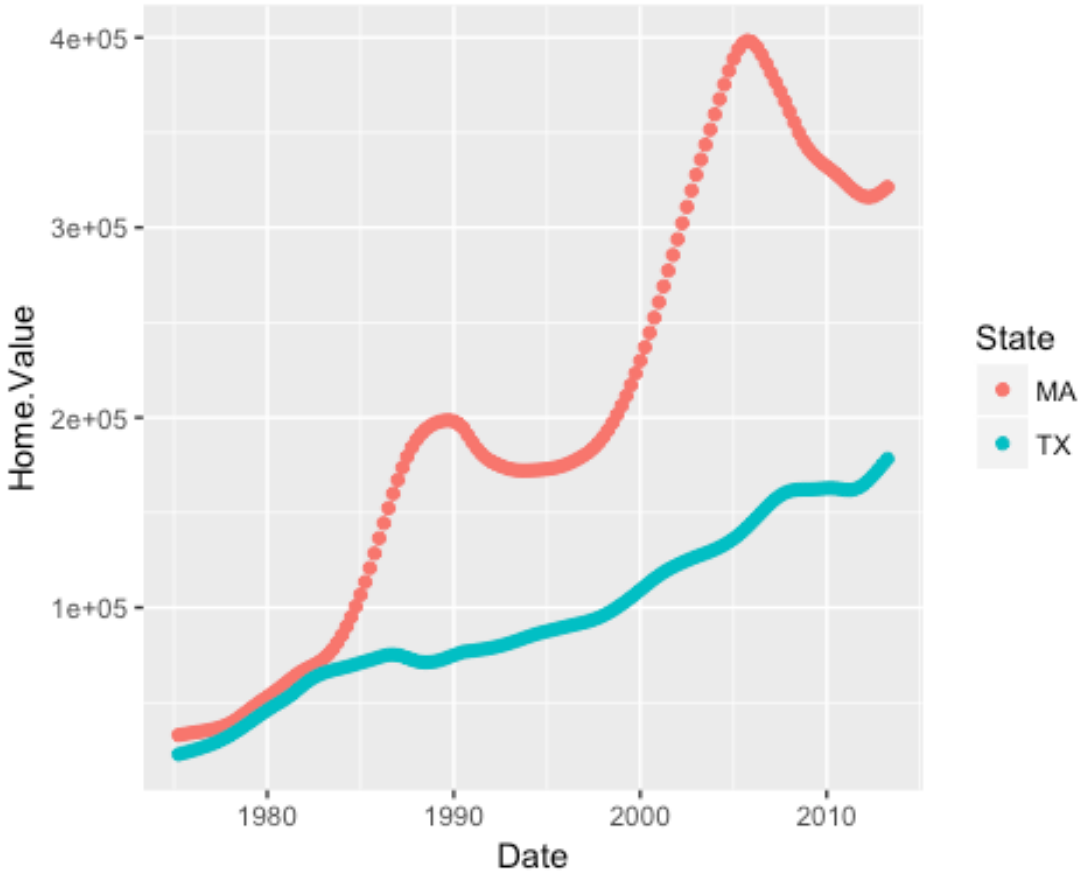
```
##   State region    Date Home.Value Structure.Cost
## 1    AK   West 2010.25     224952         160599
## 2    AK   West 2010.50     225511         160252
## 3    AK   West 2009.75     225820         163791
## 4    AK   West 2010.00     224994         161787
## 5    AK   West 2008.00     234590         155400
## 6    AK   West 2008.25     233714         157458
```

```
#GGPlot2 is a data exploration and visualisation package for the statistical
programming
#language R. ggplot2 is an implementation of Leland Wilkinson's Grammar of
Graphics.
#It can highly improve the quality and aesthetic of your graphs.It takes
#care of many of the complicated details that make plotting a hassle as well
as
#providing a powerful model of graphics that makes it easy to produce complex
#multi-layered graphics.

library(ggplot2)
ggplot(housing, aes(x = Home.Value)) +
  geom_histogram()
```

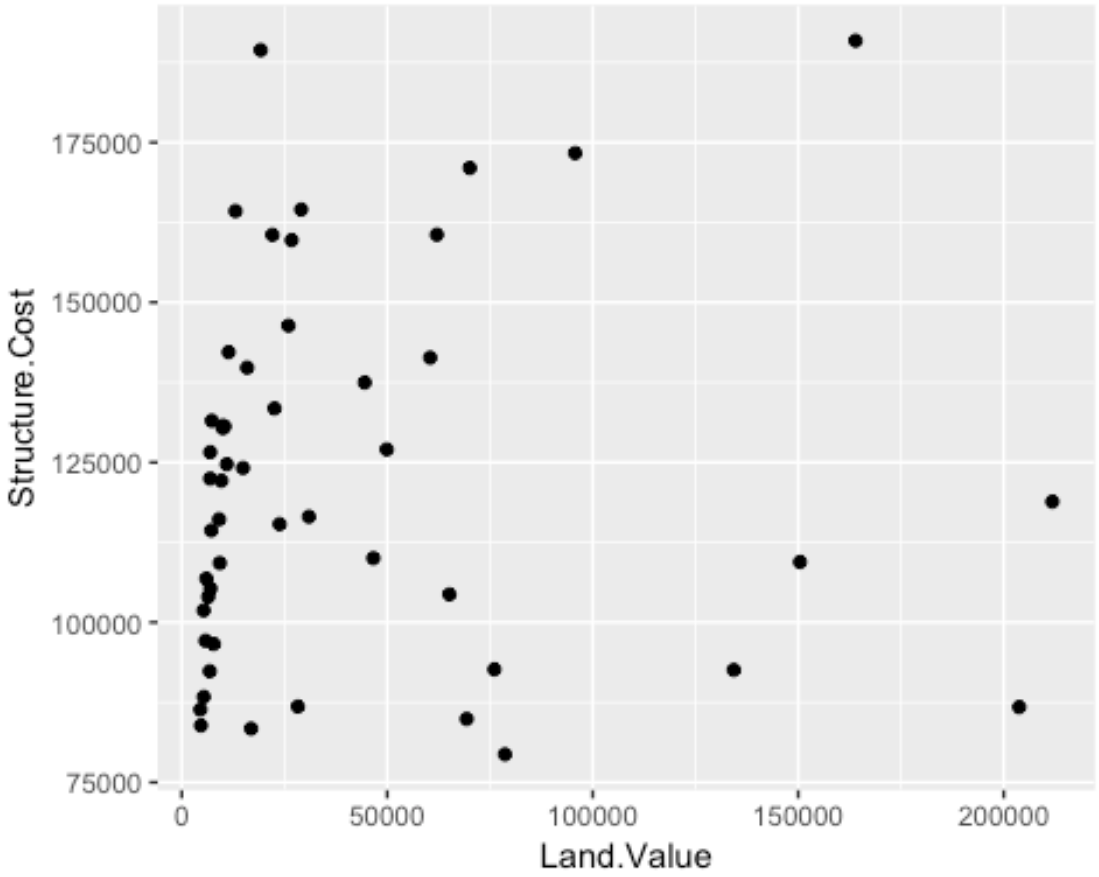
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



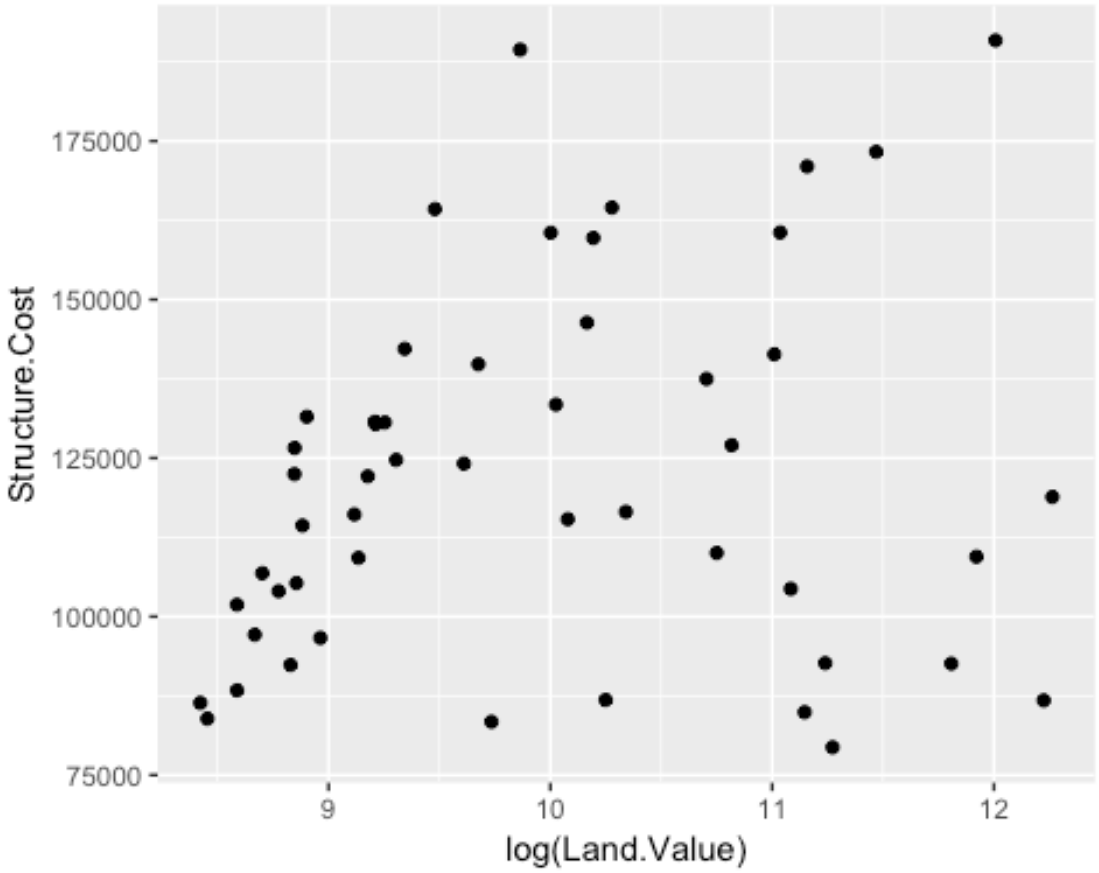

```
# This was an example of building an histogram.

#Now we will plot scatterplot

ggplot(subset(housing, State %in% c("MA", "TX")),
       aes(x=Date,
           y=Home.Value,
           color=State))+
  geom_point()
```




```
##   `ggplot2' wins!

# Geometric Objects And Aesthetics
# In ggplot land /aesthetic/ means "something you can see". Examples
# include: position (i.e., on the x and y axes),color ("outside" color), fill
("inside" color),
# shape (of points),linetype,size.
# Examples:- boxes,lines,pints.
# A plot must have at least one geom; there is no upper limit. You can
# add a geom to a plot using the `+' operator

hp2001Q1 <- subset(housing, Date == 2001.25)
ggplot(hp2001Q1,
       aes(y = Structure.Cost, x = Land.Value)) +
  geom_point()
```



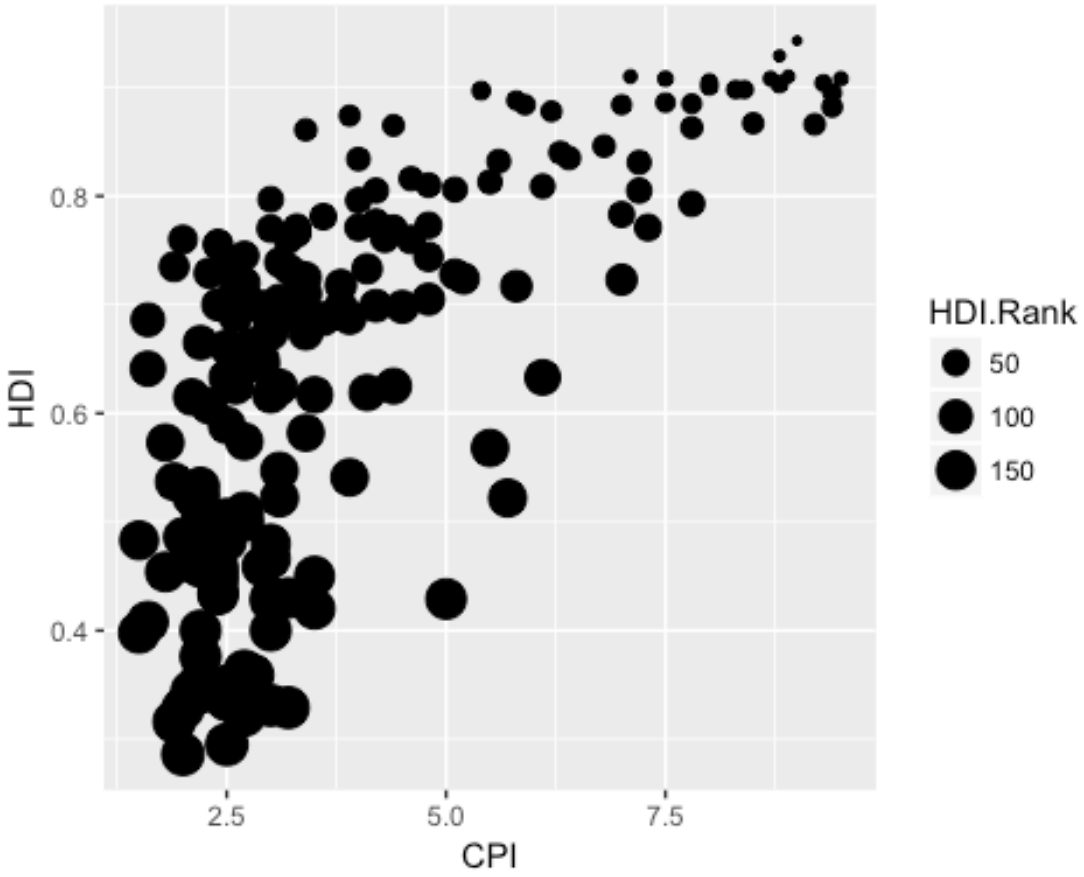

```
ggplot(hp2001Q1,
       aes(y = Structure.Cost, x = log(Land.Value))) +
  geom_point()
```



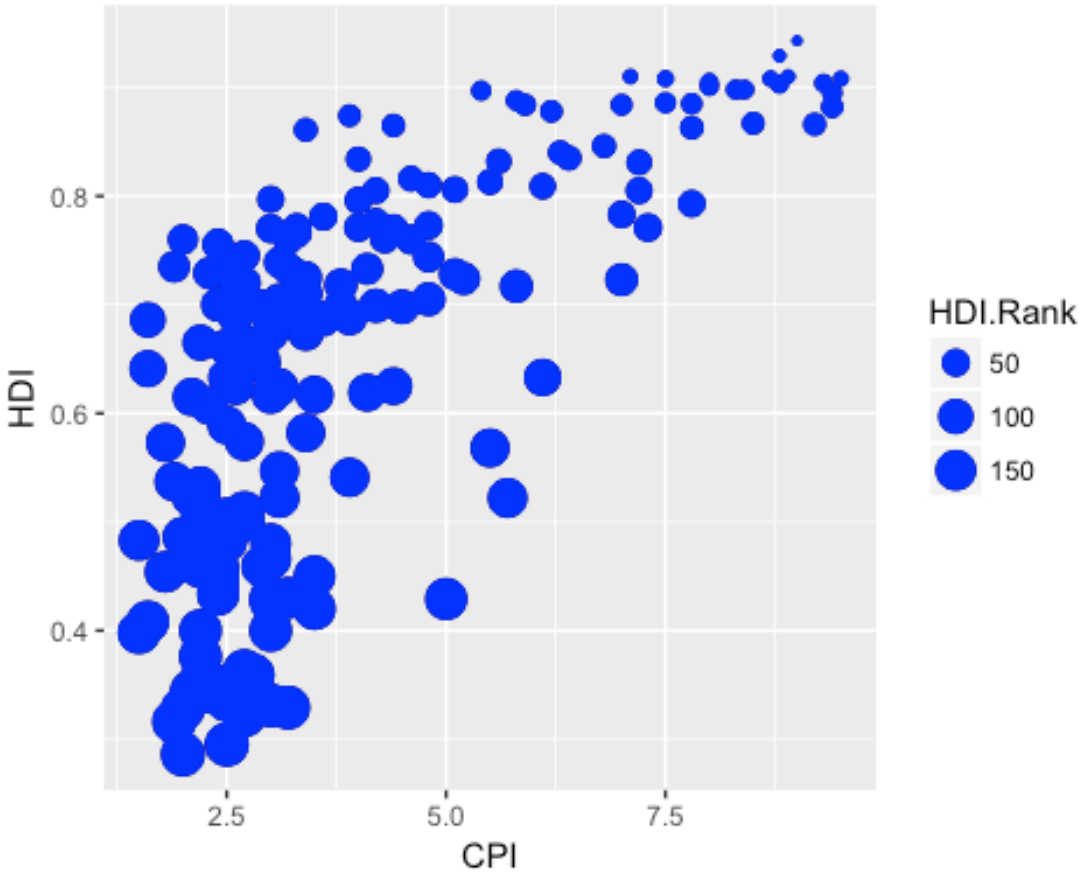

```
dat <- read.csv('EconomistData.csv')
head(dat)
```

```
##   X     Country HDI.Rank   HDI CPI            Region
## 1 1 Afghanistan      172 0.398 1.5      Asia Pacific
## 2 2     Albania       70 0.739 3.1 East EU Cemt Asia
## 3 3     Algeria       96 0.698 2.9              MENA
## 4 4      Angola      148 0.486 2.0               SSA
## 5 5   Argentina       45 0.797 3.0          Americas
## 6 6     Armenia       86 0.716 2.6 East EU Cemt Asia
```

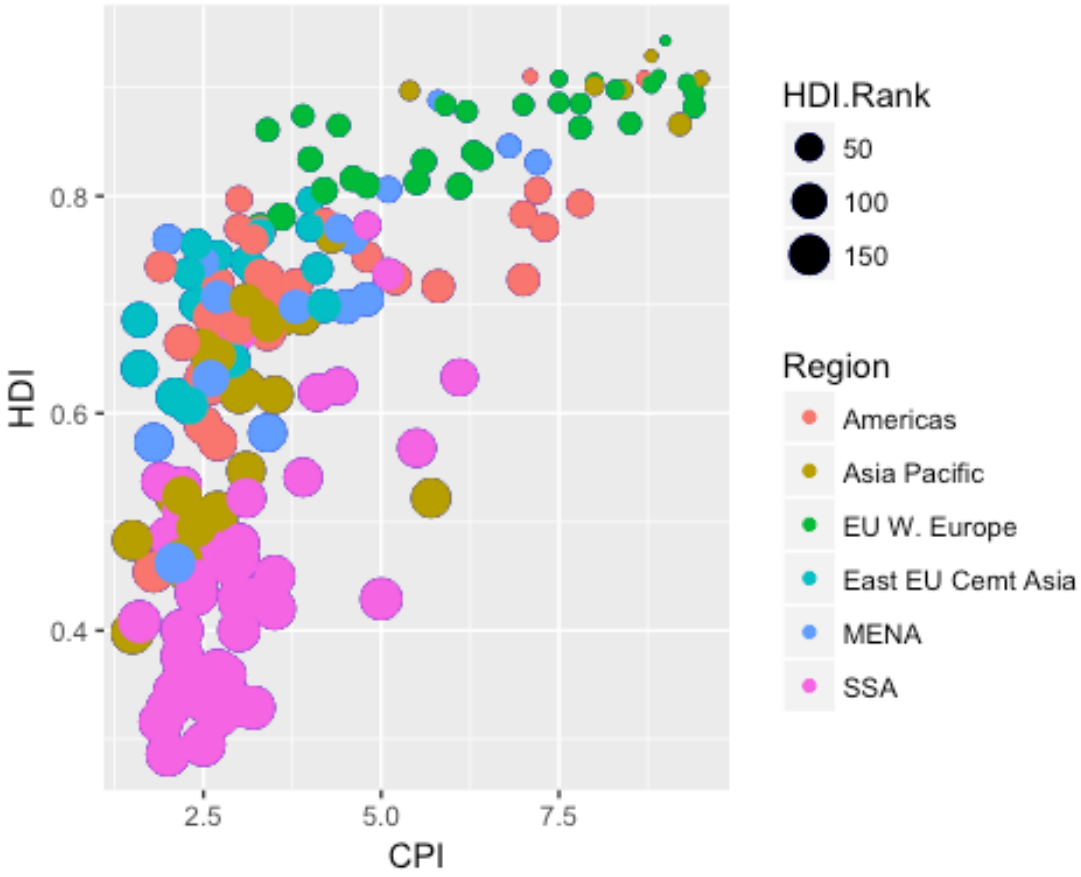
```
# Creating a scatterplot between CPI on x-axis and HDI on Y-axis.
P1 <-ggplot(dat, aes(x = CPI, y = HDI, size = HDI.Rank)) + geom_point()
P1
```



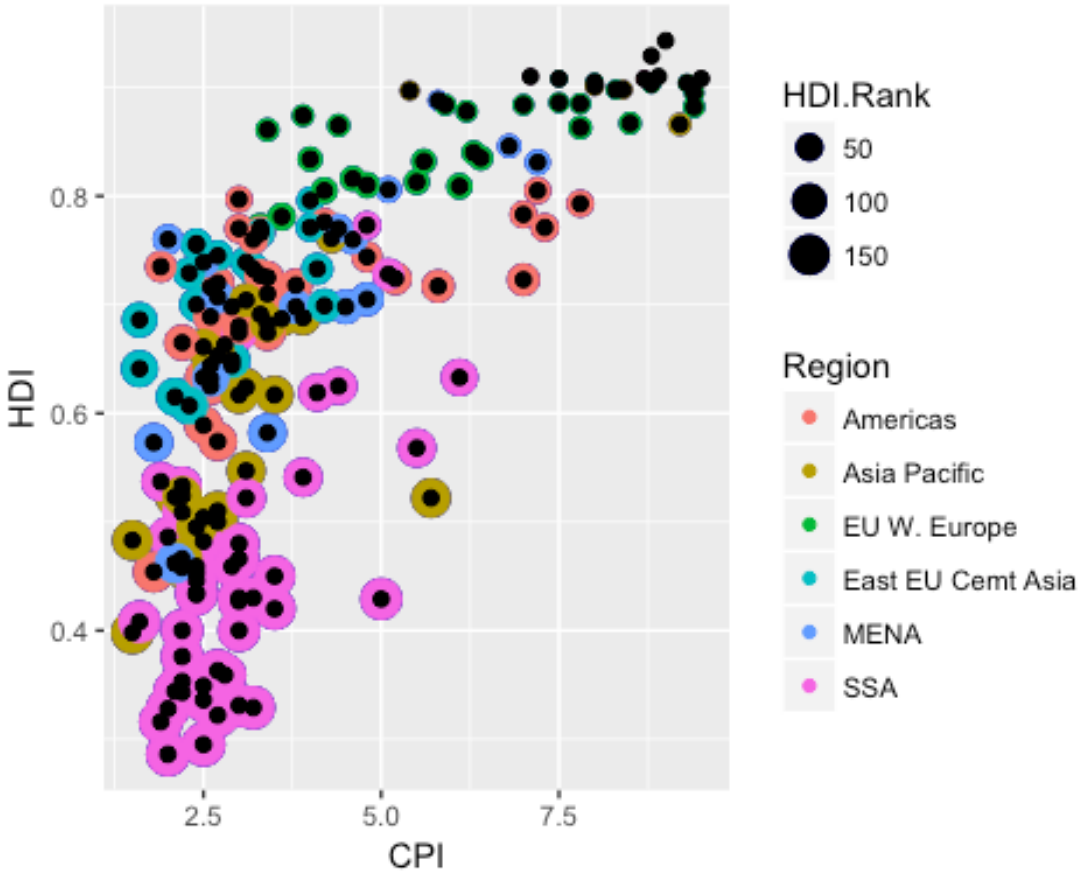

```
#  2. Color the points blue.

P2 <- P1+geom_point(colour ='blue')
P2
```



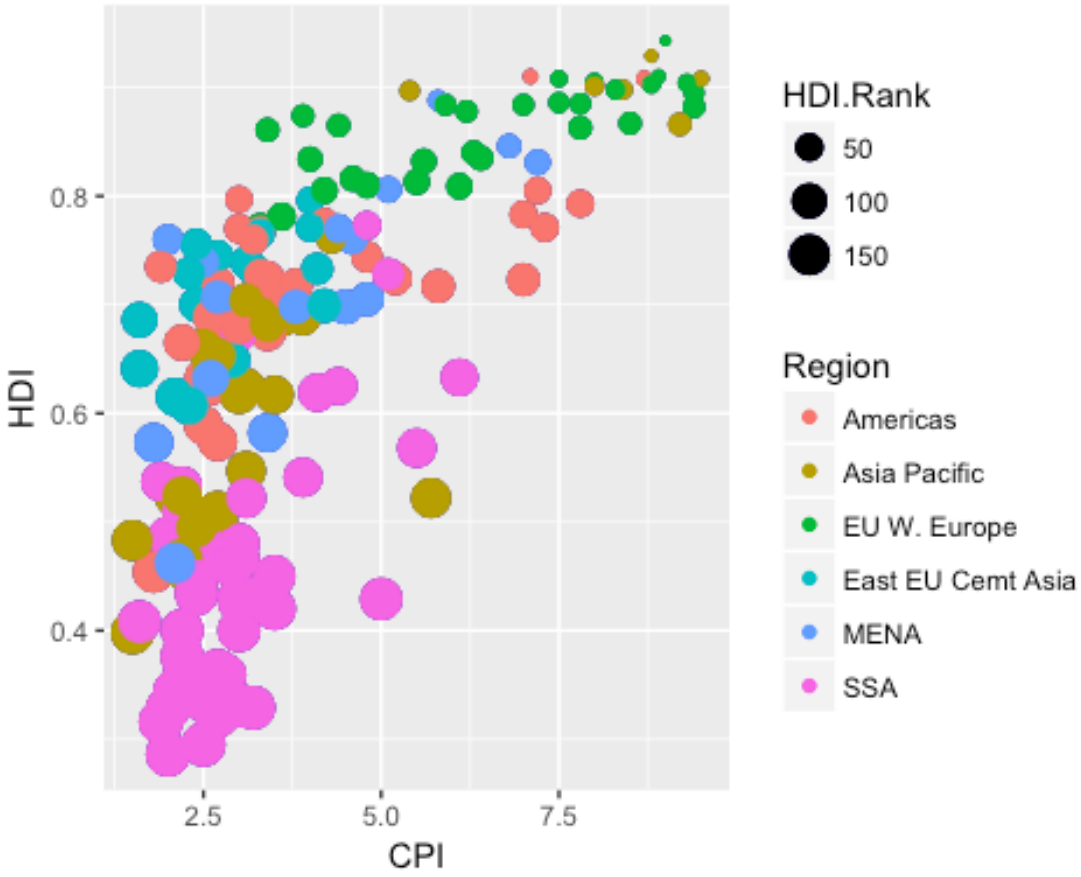

```
##   3. Map the color of the the points to Region.

P3 <- P2+geom_point(aes(color = Region))
P3
```



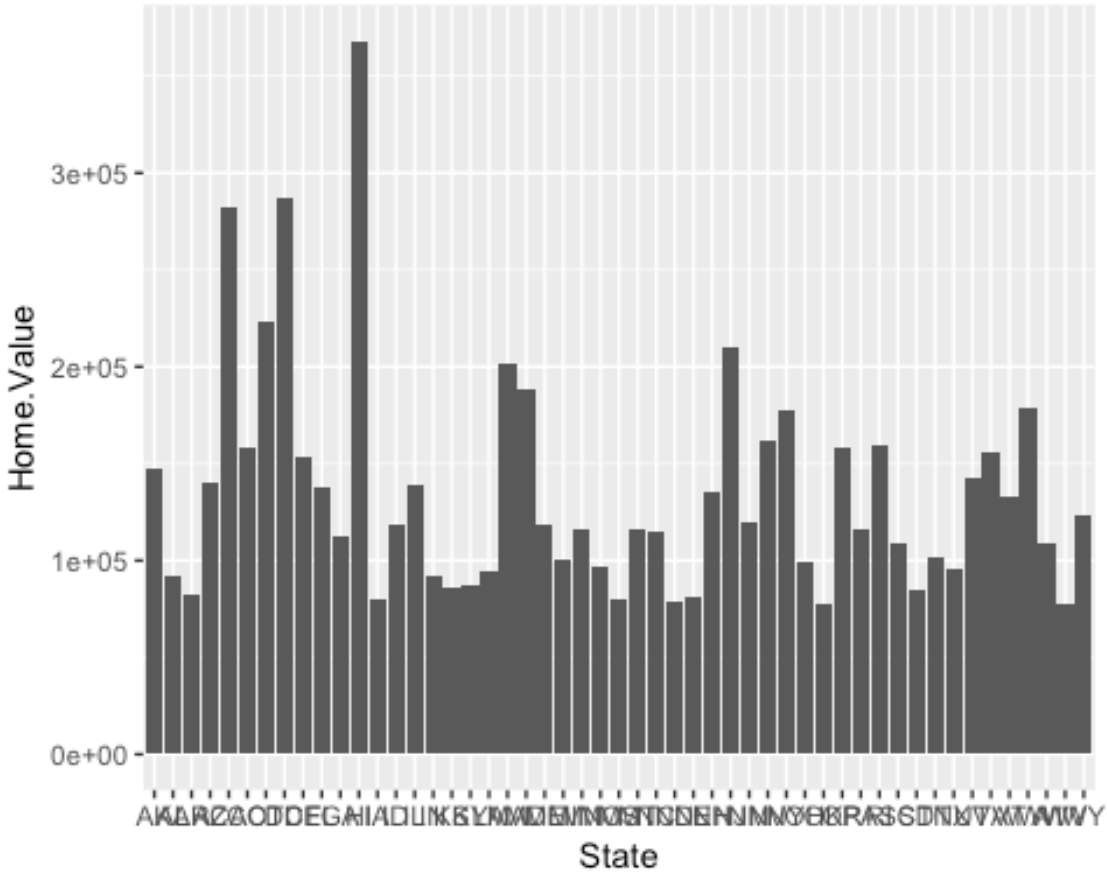

```
##   4. Make the points bigger by setting size to 2

P4 <- P3+geom_point(size=2)
P4
```



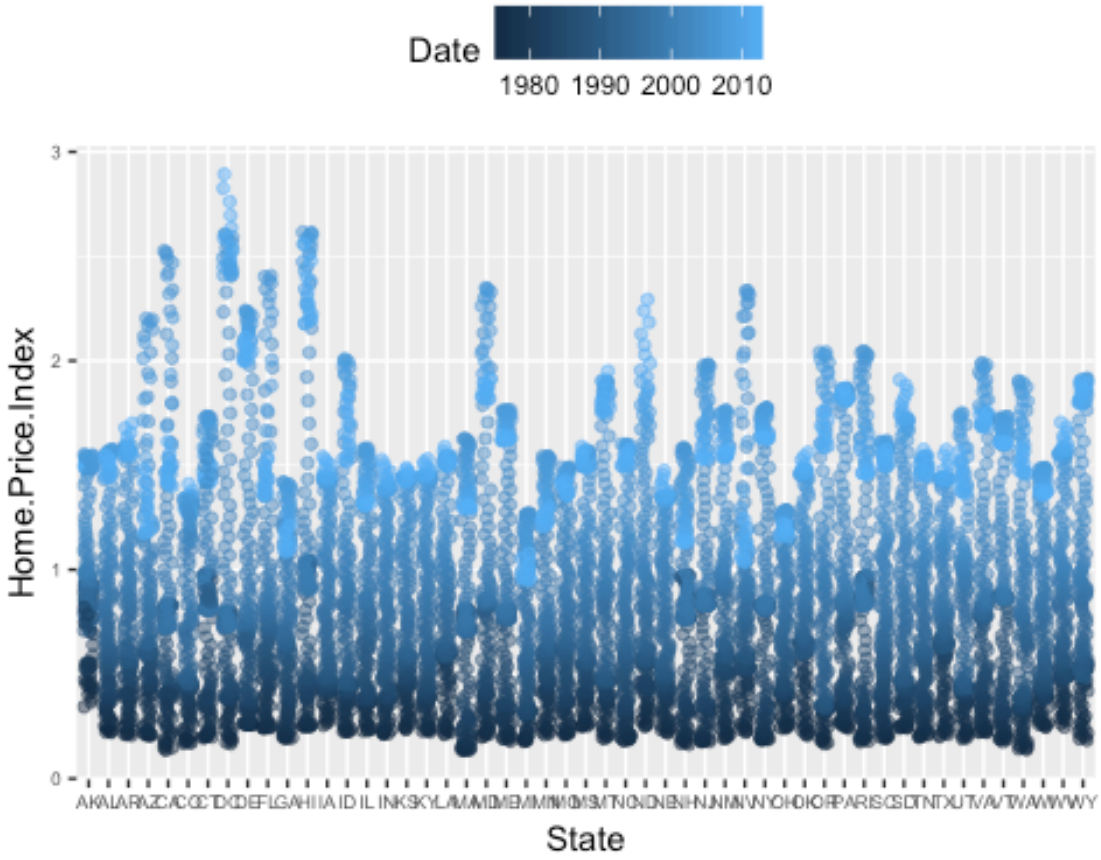

```
##   5. Map the size of the points to HDI.Rank

P4 <- P3+geom_point(aes(color = Region, size = HDI.Rank))
P4
```



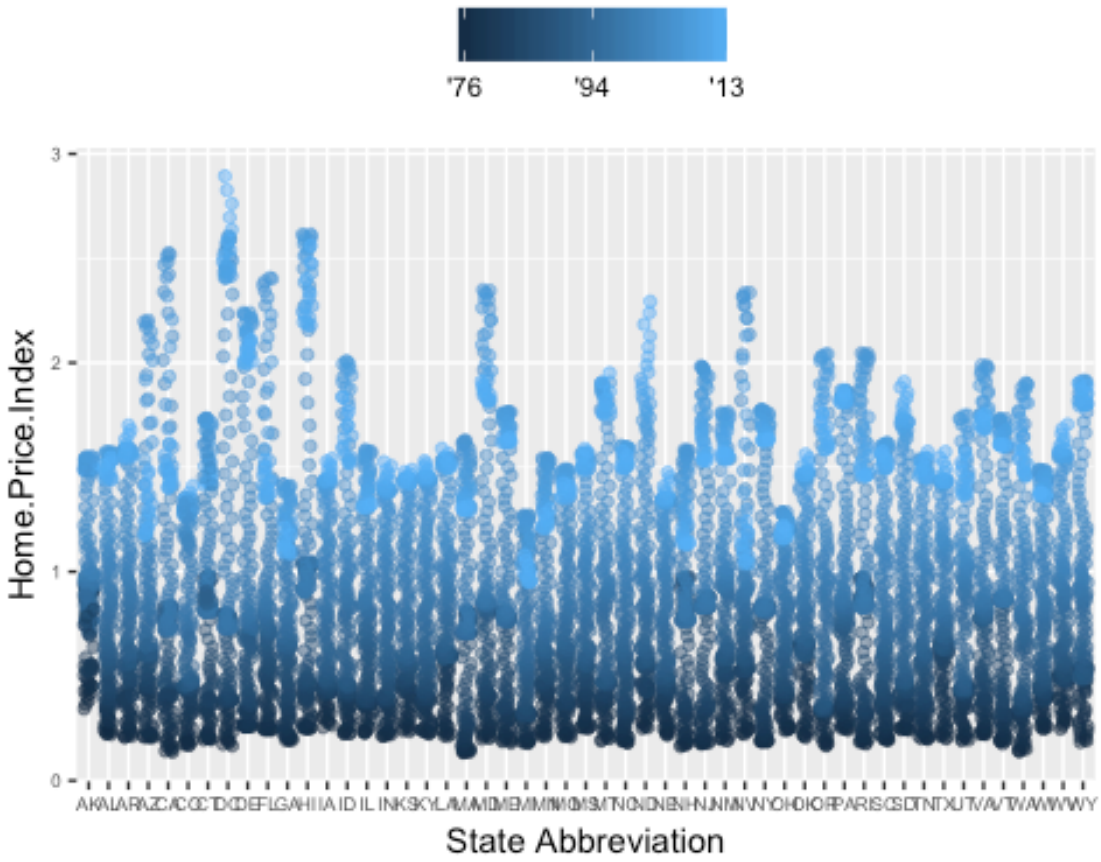

```
## Statistical Transformations

housing.sum <- aggregate(housing["Home.Value"], housing["State"], FUN=mean)
rbind(head(housing.sum), tail(housing.sum))

##    State Home.Value
## 1     AK  147385.14
## 2     AL   92545.22
## 3     AR   82076.84
## 4     AZ  140755.59
## 5     CA  282808.08
## 6     CO  158175.99
## 46    VA  155391.44
## 47    VT  132394.60
## 48    WA  178522.58
## 49    WI  108359.45
## 50    WV   77161.71
## 51    WY  122897.25

ggplot(housing.sum, aes(x=State, y=Home.Value)) +
  geom_bar(stat="identity")
```



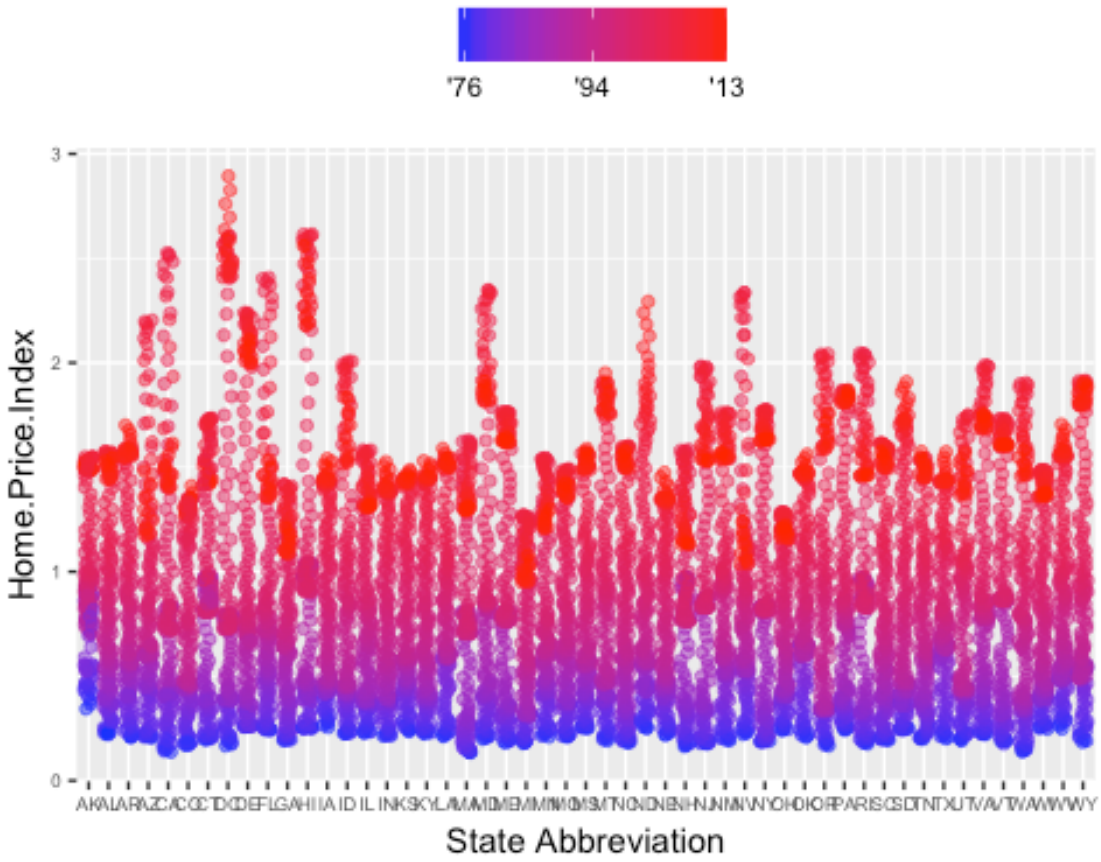

```
## Scale Modification Examples


#    Start by constructing a dotplot showing the distribution of home
#    values by Date and State.

p3 <- ggplot(housing,
             aes(x = State,
                 y = Home.Price.Index)) +
  theme(legend.position="top",
        axis.text=element_text(size = 6))
(p4 <- p3 + geom_point(aes(color = Date),
                       alpha = 0.5,
                       size = 1.5,
                       position = position_jitter(width = 0.25, height = 0)))
```



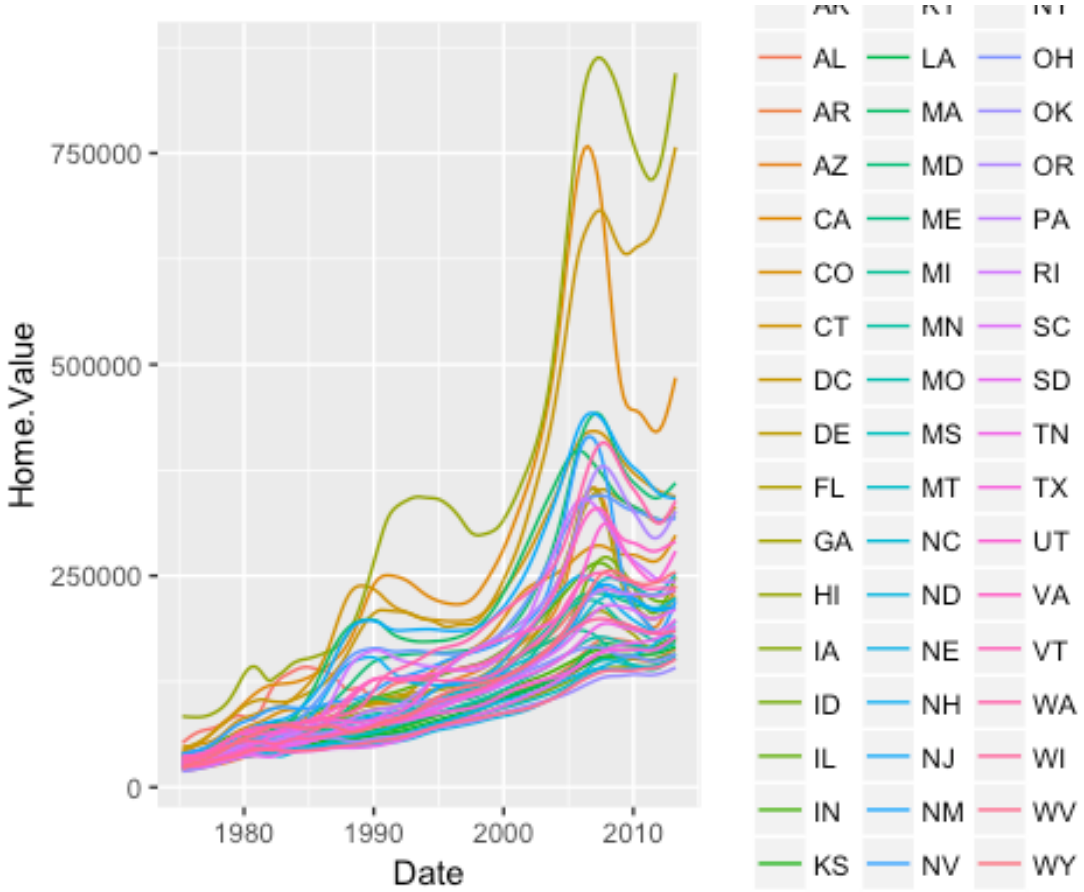

```
# Now modify the breaks for the x axis and color scales

p4 + scale_x_discrete(name="State Abbreviation") +
  scale_color_continuous(name="",
                         breaks = c(1976, 1994, 2013),
                         labels = c("'76", "'94", "'13"))
```




```
#   Next change the low and high values to blue and red:
p4 +
  scale_x_discrete(name="State Abbreviation") +
  scale_color_continuous(name="",
                         breaks = c(1976, 1994, 2013),
                         labels = c("'76", "'94", "'13"),
                         low = "blue", high = "red")
```




```
# Using different color scales

p5 <- ggplot(housing, aes(x = Date, y = Home.Value))
p5 + geom_line(aes(color = State))
```




```
#Reasons and Advantages we use GGPLOT2
#consistent underlying grammar of graphics (Wilkinson, 2005)
#very flexible
#theme system for polishing plot appearance
#many users, active mailing list
#plot specification at a high level of abstraction
#mature and complete graphics system
```