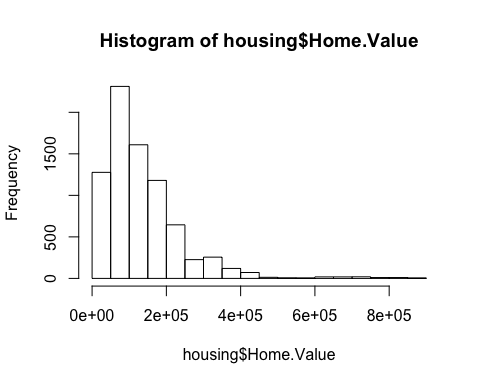
ggplot2 tutorial

## ggplot2 tutorial from harvard.edu  
## http://tutorials.iq.harvard.edu/R/Rgraphics/Rgraphics.html#orgheadline19  
############## 1. Geometric Objects and Aesthetics ###############  
housing = read.csv("dataSets/landdata-states.csv")  
head(housing[1:5])

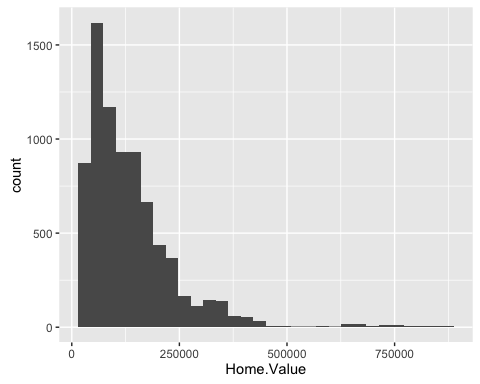
## State region Date Home.Value Structure.Cost  
## 1 AK West 20101 224952 160599  
## 2 AK West 20102 225511 160252  
## 3 AK West 20093 225820 163791  
## 4 AK West 20094 224994 161787  
## 5 AK West 20074 234590 155400  
## 6 AK West 20081 233714 157458

# Base graphics histogram   
hist(housing$Home.Value)  
# ggplot2 histogram  
library(ggplot2)

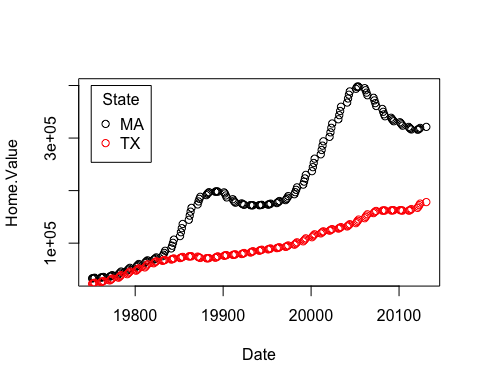


ggplot(housing, aes(x = Home.Value)) +   
 geom\_histogram()

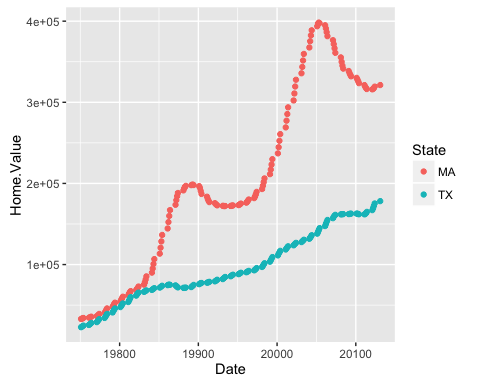
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



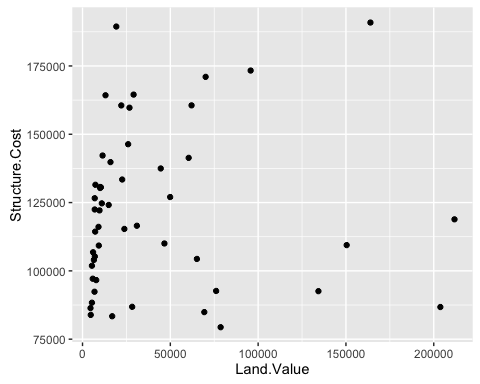
# Base color scatter plot  
plot(Home.Value ~ Date,data = subset(housing, State == "MA"))  
points(Home.Value ~ Date, col="red", data = subset(housing, State == "TX"))  
legend(19750, 400000, c("MA", "TX"), title="State", col=c("black", "red"), pch=c(1, 1))



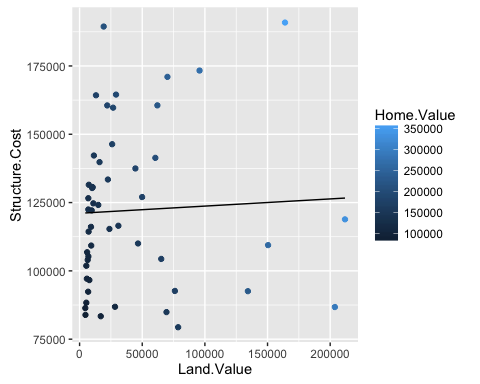
# ggplot2 color scatter plot  
ggplot(subset(housing, State %in% c("MA","TX")), aes(x = Date, y = Home.Value, color = State)) +  
 geom\_point()



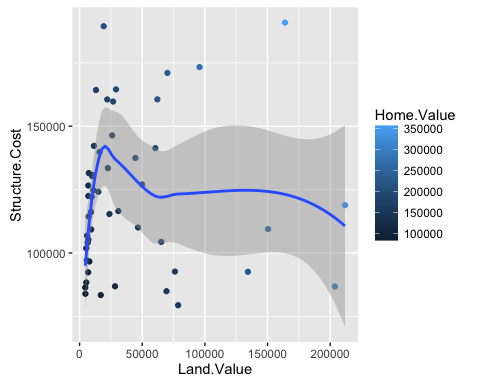
####################################################################################  
# Points(Scatterplot)  
hp2001Q1 = subset(housing, Date == 20011)  
ggplot(hp2001Q1, aes(x = Land.Value, y = Structure.Cost)) +  
 geom\_point()



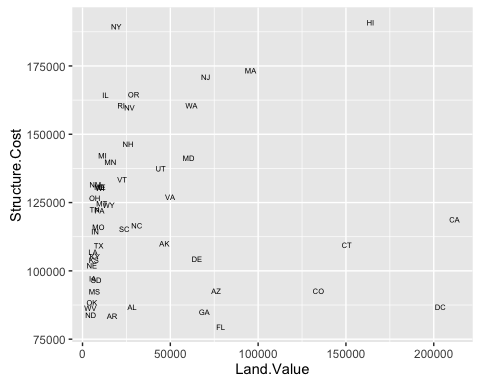
# Lines(Prediction Line)  
hp2001Q1$pred.SC = predict(lm(Structure.Cost ~ Land.Value, data = hp2001Q1))  
p1 = ggplot(hp2001Q1,aes(x = Land.Value, y = Structure.Cost))  
p1 + geom\_point(aes(color = Home.Value)) +  
 geom\_line(aes(y = pred.SC))



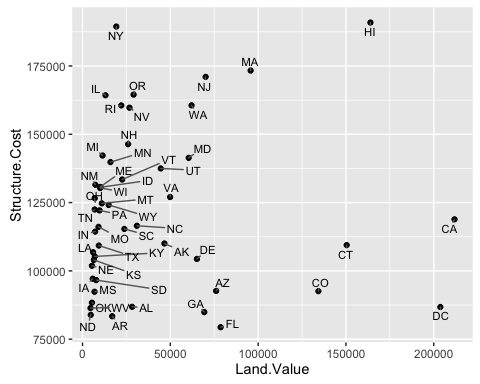
# Smoothers (model = lm)  
p1 + geom\_point(aes(color = Home.Value)) +  
 geom\_smooth()



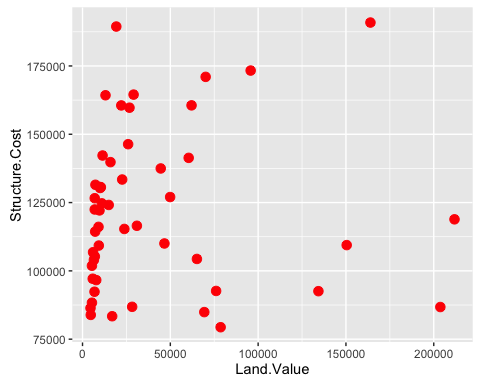
# Text(Lable Points)  
p1 + geom\_text(aes(label = State), size = 2)



# Text repel  
library("ggrepel")  
p1 + geom\_point() +  
 geom\_text\_repel(aes(label = State), size = 3)

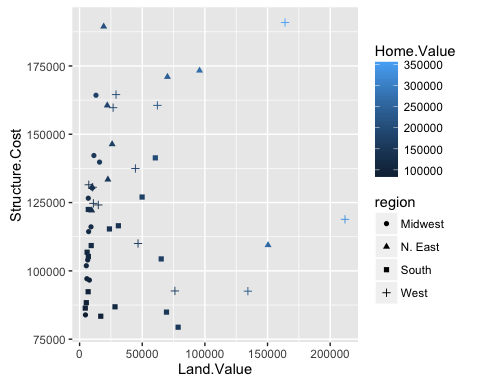


#Aesthetic Mapping vs Assignment  
p1 + geom\_point(color = "red",size = 3)

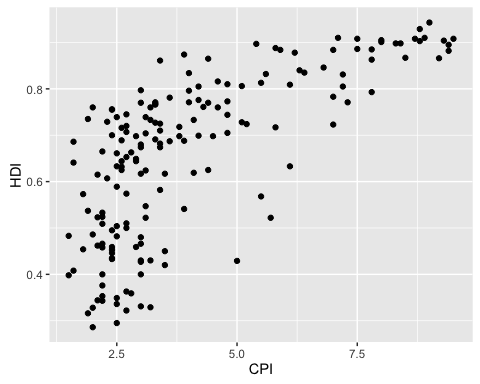


# Mapping variables tp other Aesthetics  
p1 + geom\_point(aes(color = Home.Value, shape = region))

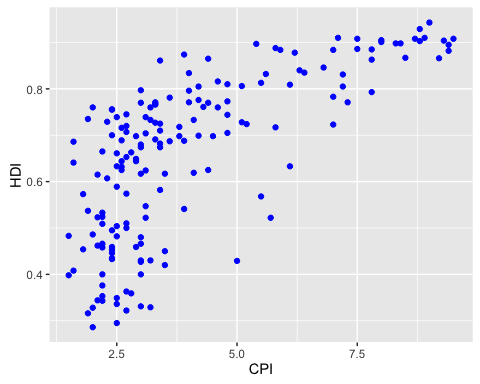
## Warning: Removed 1 rows containing missing values (geom\_point).



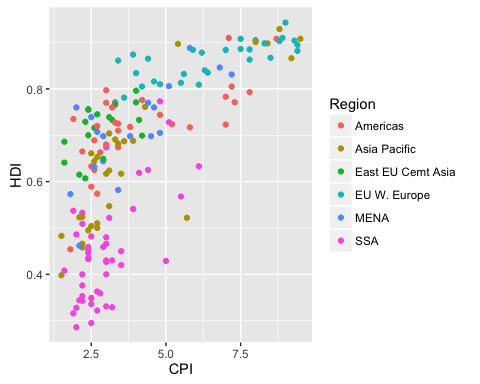
## Exercise 1  
# These data consist of Human Development Index and Corruption Perception Index scores for several countries  
# 1. Create a scatter plot with CPI on the x axis and HDI on the y axis  
dat <- read.csv("dataSets/EconomistData.csv")  
e1 = ggplot(dat, aes(x = CPI, y = HDI))  
e1 + geom\_point()



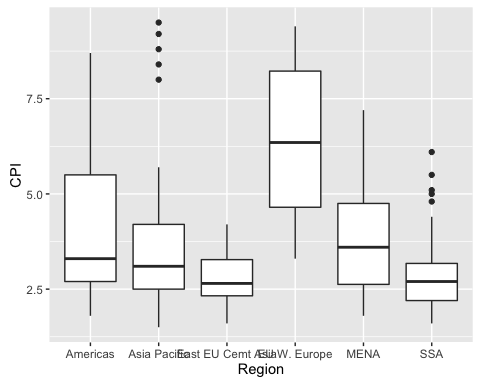
# 2. Color the points in the previous plot blue  
e1 + geom\_point(col = "blue")



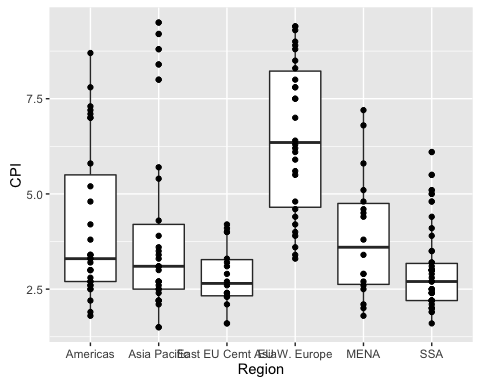
# 3. Color the points in the previous plot according to Region  
e1 + geom\_point(aes(col = Region))



# 4. Create boxplots of CPI by Region  
ggplot(dat, aes(x = Region, y = CPI)) + geom\_boxplot()

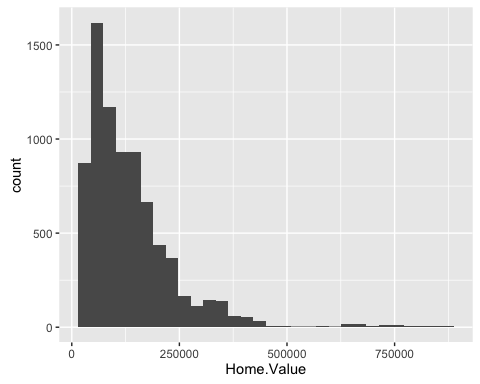


# 5. Overlay points on top of the boxplots  
ggplot(dat, aes(x = Region, y = CPI)) + geom\_boxplot() + geom\_point()

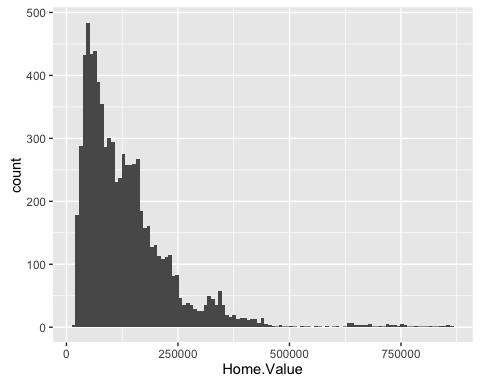


############## 2. Statistical Transformations ###############  
# Default histogram of Home.Value  
p2 = ggplot(housing, aes(x = Home.Value))  
p2 + geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



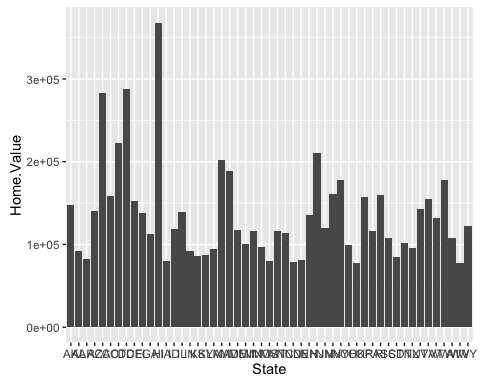
# Change the binwidth  
p2 + geom\_histogram(stat = "bin", binwidth = 8000)



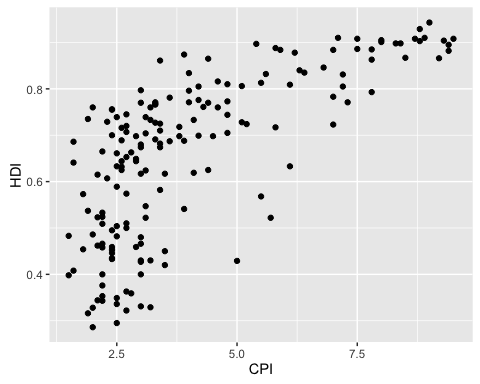
# Changing the statistical transformation  
housing.sum = aggregate(x = housing["Home.Value"], by = housing["State"], FUN = mean)  
rbind(head(housing.sum), tail(housing.sum))

## 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

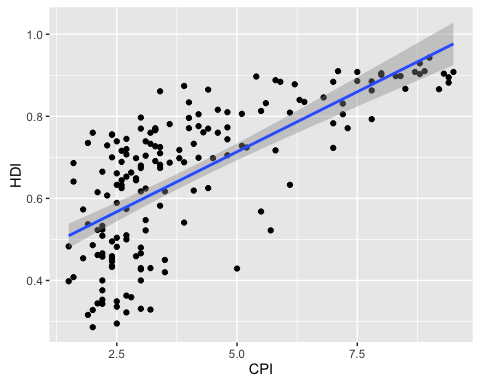
ggplot(housing.sum, aes(x = State, y = Home.Value)) + geom\_bar(stat = "identity")



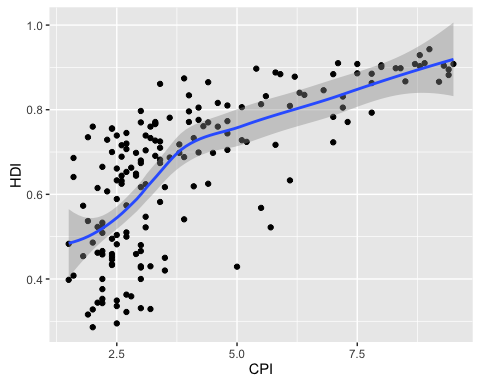
## Exercise 2  
# 1. Re-create a scatter plot with CPI on the x axis anf HDI on the y axis  
e2 = ggplot(dat, aes(x = CPI, y = HDI))  
e2 + geom\_point()



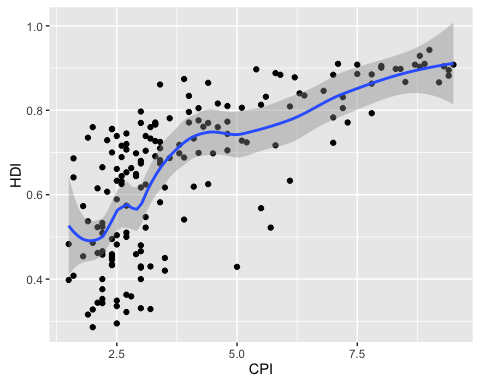
# 2. Overlay a smoothing line on the top of the scatter plot using the lm method  
e2 + geom\_point() + geom\_smooth(method = "lm")



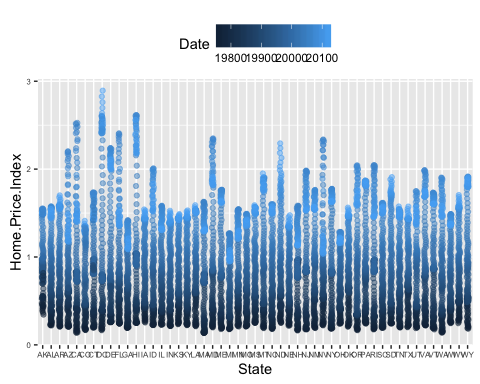
# 3. Overlay a smoothing line on top of the scatter plot using the default method  
e2 + geom\_point() + geom\_smooth()



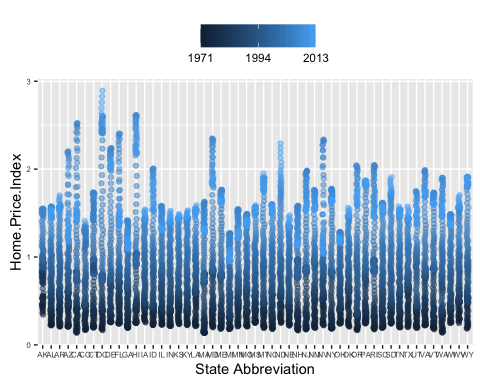
# 4. Overlay a smoothing line on top of the scatter plot using the default loess method, but make it less smooth  
e2 + geom\_point() + geom\_smooth(span = 0.4)



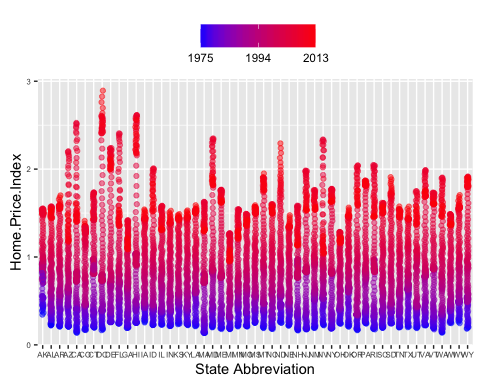
############## 3. Scales ###############  
# Scale modification examples  
# Start by constructing a dotplot showing the distribution of home valuus by Date and State  
p3 = ggplot(housing, aes(x = State, y = Home.Price.Index)) +   
 theme(legend.position = "top", axis.text = element\_text(size = 6))  
p3 + geom\_point(aes(color = Date), alpha = 0.5, size = 1.5, position = position\_jitter(width = 0.25, height = 0))



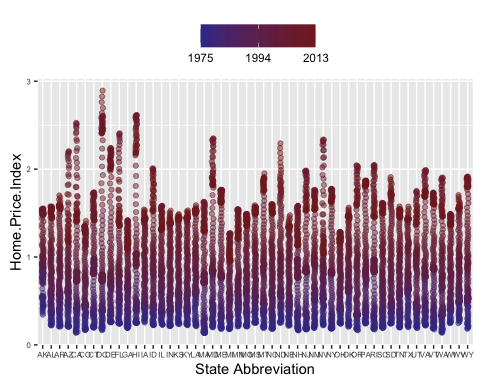
# Modify the breaks and labels for the x axis and color scales  
p4 = p3 + geom\_point(aes(color = Date), alpha = 0.5, size = 1.5, position = position\_jitter(width = 0.25, height = 0))  
p4 + scale\_x\_discrete(name = "State Abbreviation") +   
 scale\_color\_continuous(name = "",  
 breaks = c(19751,19941,20131),  
 labels = c(1971,1994,2013))



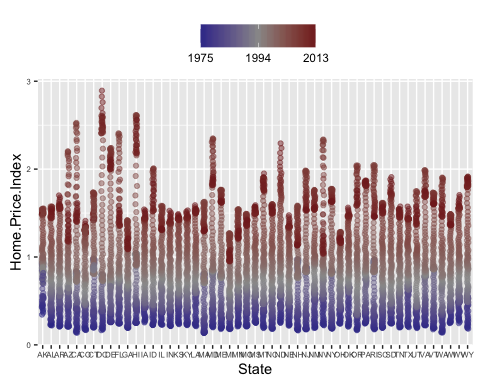
# Change the low and high values to blue and red  
p4 + scale\_x\_discrete(name = "State Abbreviation") +  
 scale\_color\_continuous(name = "",  
 breaks = c(19751,19941,20131),  
 labels = c(1975,1994,2013),  
 low = "blue", high = "red")



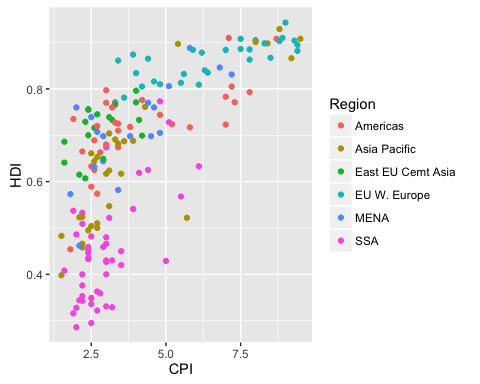
library("scales")  
p4 + scale\_x\_discrete(name = "State Abbreviation") +  
 scale\_color\_continuous(name = "",  
 breaks = c(19751,19941,20131),  
 labels = c(1975,1994,2013),  
 low = muted("blue"), high = muted("red"))



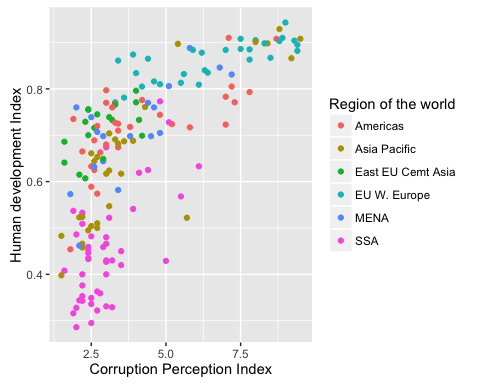
# Using different color scales  
p4 + scale\_color\_gradient2(name = "",  
 breaks = c(19751,19941,20131),  
 labels = c(1975,1994,2013),  
 low = muted("blue"), high = muted("red"), mid = "gray60", midpoint = 19941)



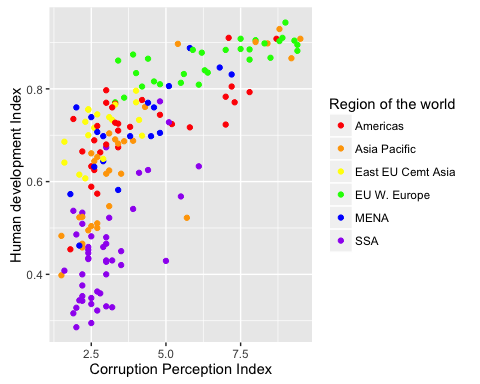
## Exercise 3  
# 1. Create a scatter plot with CPI on the x axis and HDI on the y axis. Color the points to indicate region  
e3 = ggplot(dat, aes(x = CPI, y = HDI, color = Region))  
e3 + geom\_point()



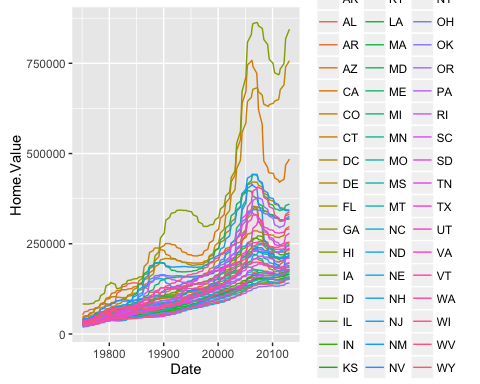
# 2. Modify the x,y, and color scales so that they have more easily-understood names  
# (e.g., spell out "Human developent Index instead of "HDI")  
e3 + geom\_point() + scale\_x\_continuous(name = "Corruption Perception Index") +   
 scale\_y\_continuous(name = "Human development Index") +  
 scale\_color\_discrete(name = "Region of the world")



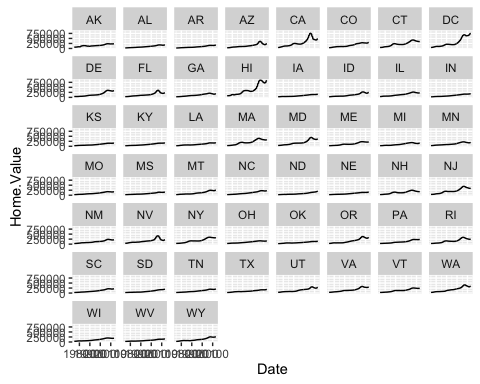
# 3. Modify the color scale to use specific values of your choosing  
e3 + geom\_point() + scale\_x\_continuous(name = "Corruption Perception Index") +   
 scale\_y\_continuous(name = "Human development Index") +  
 scale\_color\_manual(name = "Region of the world",  
 values = c("red","orange","yellow","green","blue","purple"))



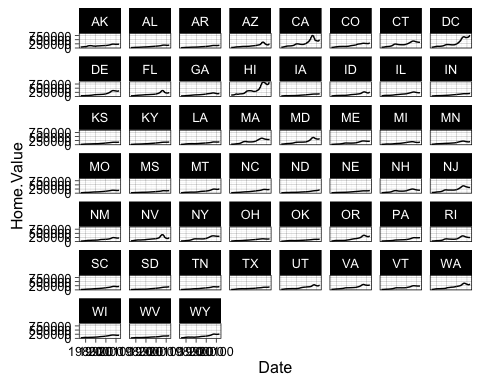
############## 4. Faceting ###############  
p5 = ggplot(housing, aes(x = Date, y = Home.Value))  
p5 + geom\_line(aes(color = State))



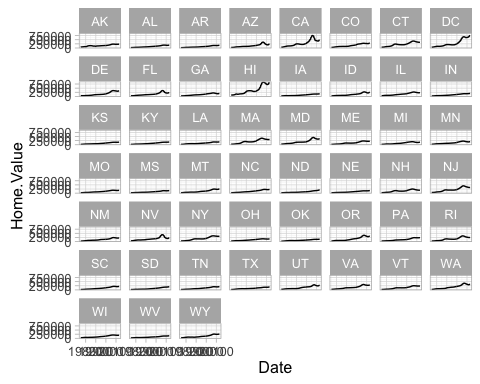
# Plot by faceting by state rather than mapping state to color  
p5 + geom\_line() +   
 facet\_wrap(~ State, ncol = 8)



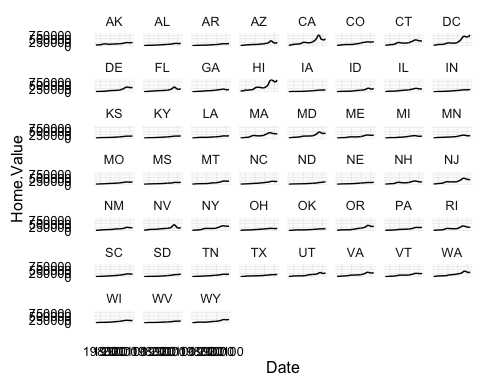
############## 5. Themes ###############  
p5 = p5 + geom\_line() +   
 facet\_wrap(~ State, ncol = 8)  
p5 + theme\_linedraw()



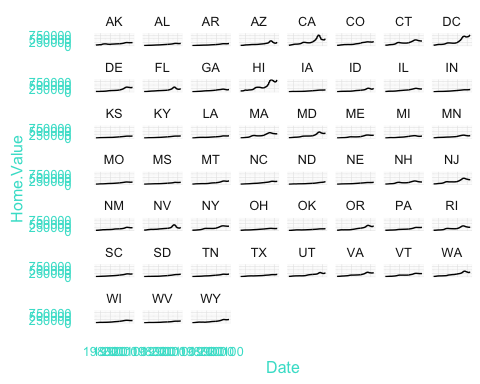
p5 + theme\_light()



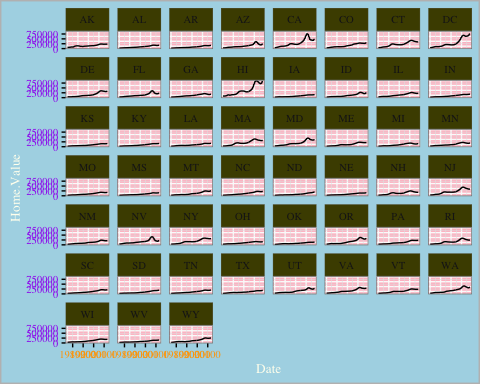
p5 + theme\_minimal()



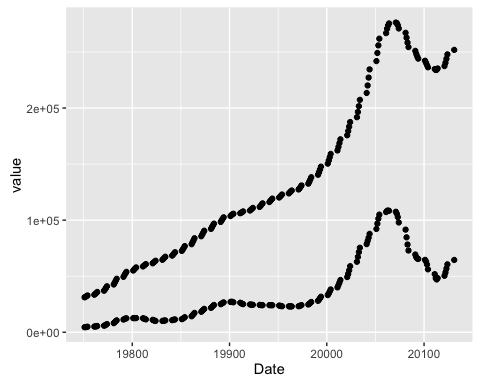
# Overriding theme defaults  
p5 + theme\_minimal() +  
 theme(text = element\_text(color = "turquoise"))



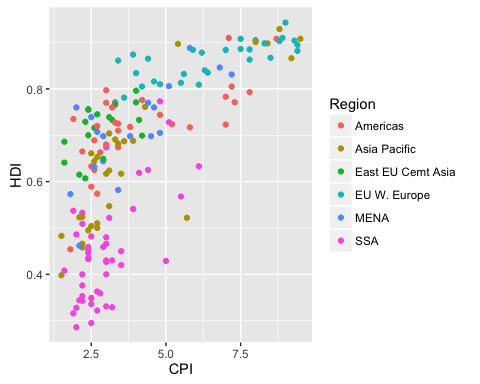
# Creating and saving new themes  
theme\_new = theme\_bw() +  
 theme(plot.background = element\_rect(size = 1, color = "grey", fill = "lightblue"),  
 text = element\_text(size = 10, family = "serif", color = "ivory"),  
 axis.text.x = element\_text(color = "orange"),  
 axis.text.y = element\_text(color = "purple"),  
 panel.background = element\_rect(fill = "pink"),  
 strip.background = element\_rect(fill = muted("yellow")))  
p5 + theme\_new



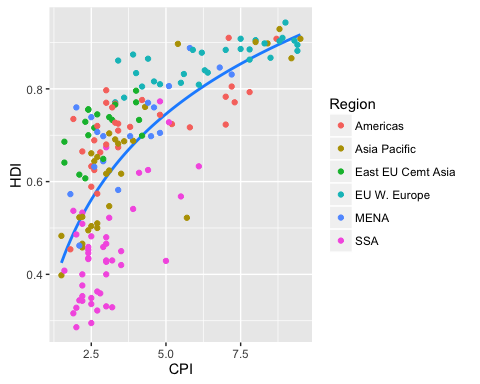
# Map Aesthetic to different columns  
library(tidyr)  
housing.byyear = aggregate(cbind(Home.Value, Land.Value) ~ Date, data = housing, mean)  
home.land.byyear = gather(housing.byyear,  
 value = "value",  
 key = "type",  
 Home.Value, Land.Value)  
ggplot(home.land.byyear, aes(x = Date, y = value)) +  
 geom\_point()



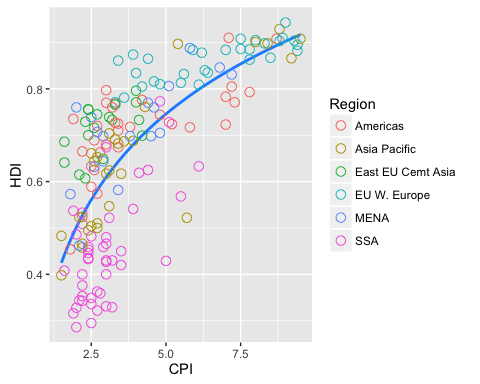
## Challenge problem  
data = read.csv("dataSets/EconomistData.csv")  
# Basic graph  
c1 = ggplot(data,aes(x = CPI, y = HDI, color = Region))  
c1 + geom\_point()



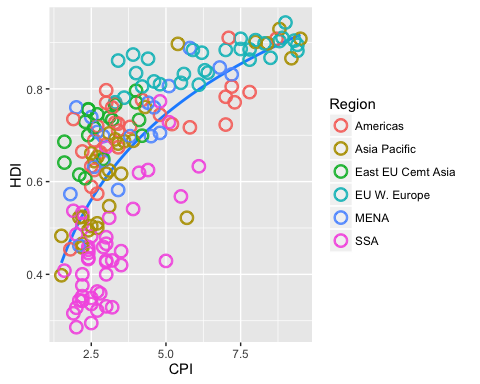
# 1. Add a trend line  
c2 = c1 + geom\_smooth(aes(group = 1),  
 method = "lm",  
 formula = y ~ log(x),  
 se = FALSE,  
 color = "dodgerblue")  
c2 + geom\_point()



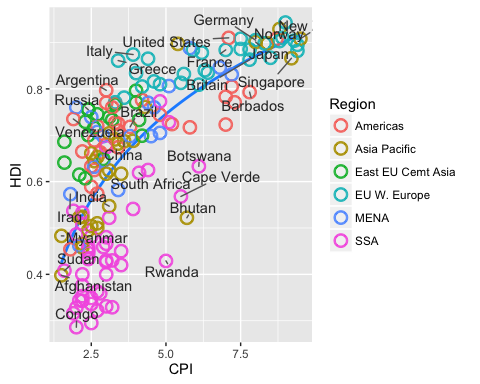
# Comments: group = 1 fits a single line of best fit   
  
# 2. Change the point shape to open circle  
c2 + geom\_point(shape = 1, size = 3)



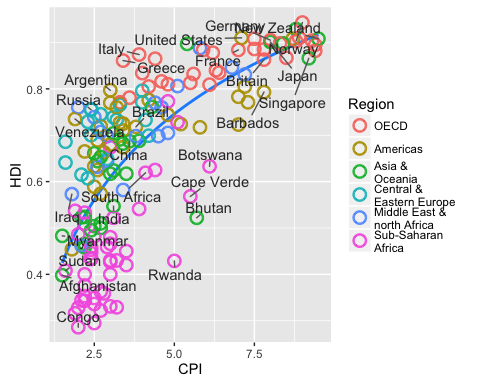
# Multiple point layers of slightly different size  
c3 = c2 + geom\_point(size = 4.5, shape = 1) +  
 geom\_point(size = 4, shape = 1) +  
 geom\_point(size = 3.5, shape = 1)  
c3



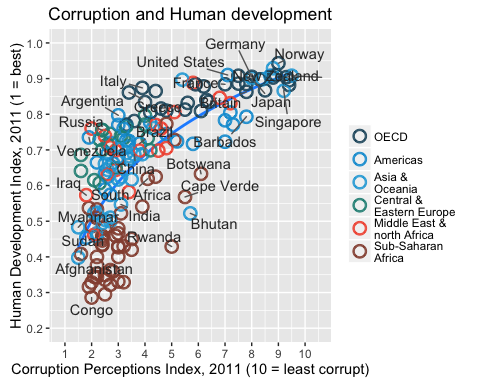
# 3. Label select points  
pointsToLabel = c("Russia", "Venezuela", "Iraq", "Myanmar", "Sudan",  
 "Afghanistan", "Congo", "Greece", "Argentina", "Brazil",  
 "India", "Italy", "China", "South Africa", "Spane",  
 "Botswana", "Cape Verde", "Bhutan", "Rwanda", "France",  
 "United States", "Germany", "Britain", "Barbados", "Norway", "Japan",  
 "New Zealand", "Singapore")  
library("ggrepel")  
c4 = c3 + geom\_text\_repel(aes(label = Country),  
 color = "gray20",  
 data = subset(data, Country %in% pointsToLabel),  
 force = 10)  
c4



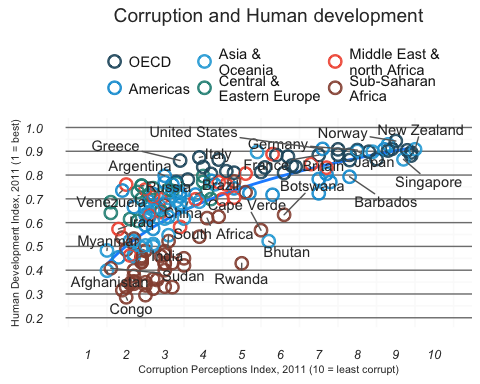
# Comments: Force of replusion between overlapping labels. Default is 1  
  
# 4. change the order and labels of Region  
# To change the region labels and order, we need to use the factor function  
data$Region = factor(data$Region,   
 levels = c("EU W. Europe","Americas","Asia Pacific",  
 "East EU Cemt Asia","MENA","SSA"),  
 labels = c("OECD","Americas","Asia &\nOceania",  
 "Central &\nEastern Europe","Middle East &\nnorth Africa","Sub-Saharan\nAfrica"))  
c4$data = data  
c4



# 5. Add title and format axes  
library("grid")  
c5 = c4 + scale\_x\_continuous(name = "Corruption Perceptions Index, 2011 (10 = least corrupt)",  
 limits = c(0.9,10.5),  
 breaks = 1:10) +  
 scale\_y\_continuous(name = "Human Development Index, 2011 (1 = best)",  
 limits = c(0.2,1.0),  
 breaks = seq(0.2,1.0,by = 0.1)) +  
 scale\_color\_manual(name = "",  
 values = c("#24576D","#099DD7","#28AADC","#248E84","#F2583F","#96503F")) +  
 ggtitle("Corruption and Human development")  
c5



# 6. Theme tweaks  
c6 = c5 + theme\_minimal() + # start with the minimal theme and add what we need  
 theme(text = element\_text(color = "gray20"),  
 legend.position = c("top"),  
 legend.direction = "horizontal",  
 legend.justification = 0.1,  
 legend.text = element\_text(size = 11, color = "gray10"),  
 axis.text = element\_text(face = "italic"),  
 axis.title.x = element\_text(size = 8, vjust = -1), # move title away from axis  
 axis.title.y = element\_text(size = 8, vjust = 2),  
 axis.ticks.y = element\_blank(),  
 axis.line = element\_line(color = "gray40",size = 0.5),  
 axis.line.y = element\_blank(),  
 panel.grid.major = element\_line(color = "gray50",size = 0.5),  
 panel.grid.major.x = element\_blank()  
 )  
c6



# 7. Add model R^2 and source note  
mr2 = summary(lm(HDI ~ log(CPI), data = data))$r.squared  
library(grid)  
png(file = "images/econScatter10.png", width = 800, height = 600)  
c6  
grid.text("Sources: Transparency International; UN Human Development Report",  
 x = 0.02, y = 0.02, just = "left", draw = TRUE)   
grid.segments(x0 = 0.81, x1 = 0.825,  
 y0 = 0.90, y1 = 0.90,  
 gp = gpar(col = "red"),  
 draw = TRUE)  
grid.text(paste0("R² = ",  
 as.integer(mr2\*100),  
 "%"),  
 x = 0.835, y = 0.90,  
 gp = gpar(col = "gray20"),  
 draw = TRUE,  
 just = "left")  
dev.off()

## quartz\_off\_screen   
## 2