Base R Replication

Sam Eckhardt

9/16/2014

# Purpose

I am creating this document as a guide to you, newcomer. Welcome to R. I am your guide, Alfonzo. To start, each of the shaded areas of the paper, is what I have entered into RStudio to make the resulting graph. Anything with a "#" in front of it within those shaded areas are notes, notes that you can make to help yourself. They are not necessary for the code, but when dealing with a huge string of code, it will be very helpful to you.

# Data

First, we will start with creating some data. We will have 3 Factor, or Qualitative, variables, and 4 Numeric, or Quantitiative, variables.

## Simulate some data. This is all randomly generated data from R.   
  
## 3 Factor Variables  
FacVar1=as.factor(rep(c("level1","level2"),25))  
FacVar2=as.factor(rep(c("levelA","levelB","levelC"),17)[-51])  
FacVar3=as.factor(rep(c("levelI","levelII","levelIII","levelIV"),13)[-c(51:52)])  
  
## 4 Numeric Variables.   
##R uses a seed to randomly generate data. The seed for the data we are using is seed 123.   
## If you use seed 123, you will always get this exact random data.   
## if you use no seed, R creates a seed off of the clock of your computer.   
## Good luck reproducing it.   
set.seed(123)  
NumVar1=round(rnorm(n=50,mean=1000,sd=50),digits=2) ## Normal distribution  
set.seed(123)  
NumVar2=round(runif(n=50,min=500,max=1500),digits=2) ## Uniform distribution  
set.seed(123)  
NumVar3=round(rexp(n=50,rate=.001)) ## Exponential distribution  
NumVar4=2001:2050  
  
simData=data.frame(FacVar1,FacVar2,FacVar3,NumVar1,NumVar2,NumVar3,NumVar4)

Now that we have our randomly generated data, lets mess with it.

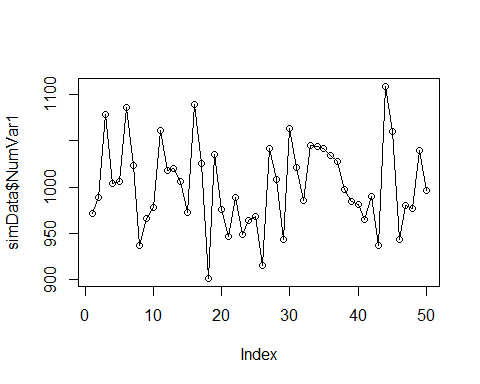
# Plotting a Single Numeric Variable

While plotting a single variable may seem a bit pointless, there is lots of information we can recieve from the following graphs.

## Index

An Index plot is just plotting the points generated from R, using the number of points as the X axis. the first point is the first number generated, the second point number 2, etc. This can be helpful by quickly seeing if there is any autocorrelation in your data, or any data that is basing itself off of data from the previous period. There will be an obvious pattern if you have Autocorrelation.

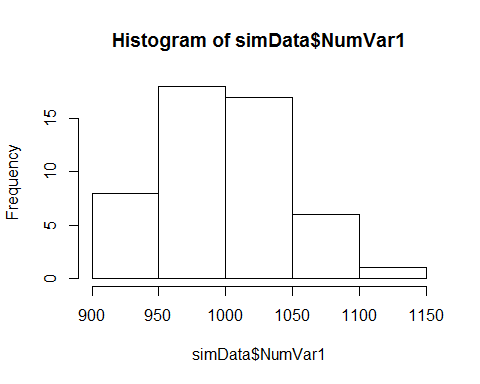
plot(simData$NumVar1,type="o") ## Index plot

 Here, we use NumVar1. We could use any of the 4 variables, and it would make a graph of any of those variables. Type o is a certain type of chart, specifically it adds the line to the graph. There are several types of plots; you can find them by typing "?plot" into the console.

## Histogram

Histograms are to Quantitative Data what Bar Charts are to Qualitative Data.

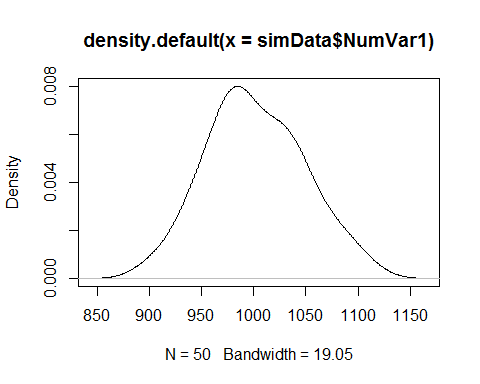
hist(simData$NumVar1) ## Histogram



## Kernel Density Plot

Density Plots show the density of the distribution to each individual number. for example, at ~1000, there are .008 points of data. Since there are only 50 points of data, and a range of over 300 in the data, the density is quite low; if we had 500, or 5000 data points, the density would increase.

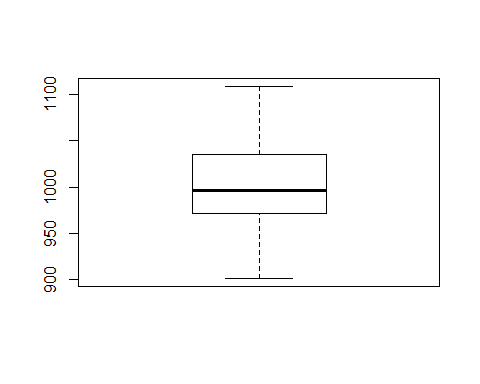
plot(density(simData$NumVar1)) ## Kernel density plot



## Box Plot

A Box Plot is a plot that shows the distribution of the data along the range of numbers. From the bottom line to the bottom of the box, that is the lowest quartile of points. From the bottom of the box to the thick black line is the second quartile, and the Black Line is the median. Black line to top of box is the third quartile, and from top of box to top line is the fourth quartile.  
Occasionally, there will be dots above (below) your top (bottom) line. These are outliers, and R determines those as outliers mathematically (also known as magic).

boxplot(simData$NumVar1) ## box plot

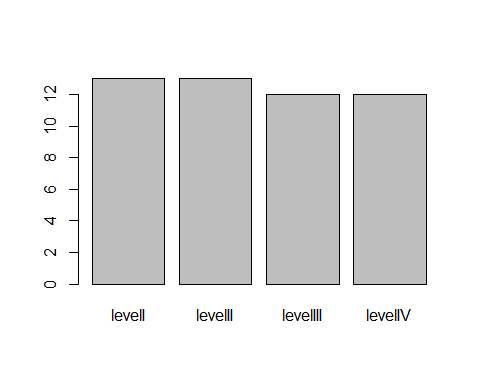


# Plotting a Single Factor Variable

## Bar Plot

This bar plot shows the number of times each level shows up in Factor Variable 3.

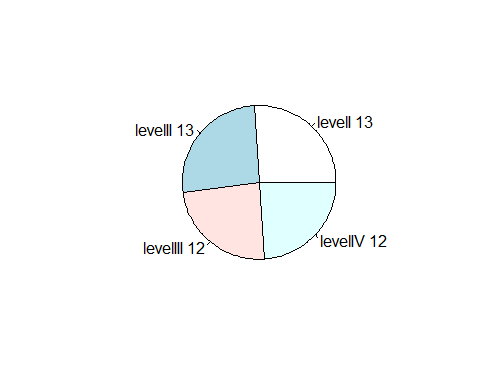
plot(simData$FacVar3) ## Bar Plot



## Pie Chart

This shows the exact same information as the chart above. However, since the human brain is much better at picturing length instead of area, it is harder to determin which slice of the pie is biggest. This terrible malady of the human brain also leads to countless fights as children over who got the biggest slice of Caramel Apple Pie.

## The only good pie is apple. pie chart - Not the best graph --- use with caution  
counts=table(simData$FacVar3) ## get counts  
labs=paste(simData$FacVar3,counts)## create labels  
pie(counts,labels=labs) ## plot



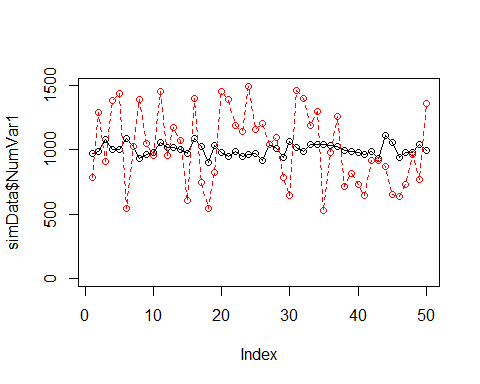
# Two Variables: Numeric

Comparing two variables is a major reason for graphs over a data set; it is much easier to comprehend the relationship between two variables this way.

## Index

Just like last time, we are plotting the points generated by R, corresponding to the number in which it was generated. The black line is the same as the first line, NumVar1; the red line is NUmVar2

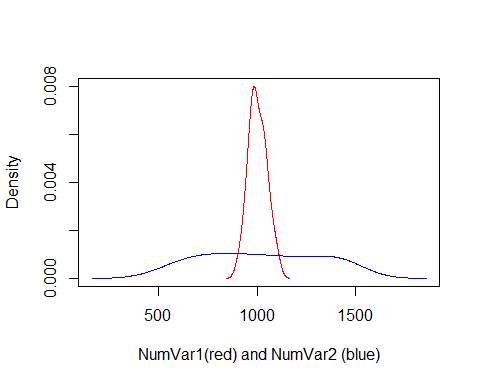
plot(simData$NumVar1,type="o",ylim=c(0,max(simData$NumVar1,simData$NumVar2)))## index plot with one variable  
lines(simData$NumVar2,type="o",lty=2,col="red")## add another variable

 As you can see, what looked like major swings in NumVar1 in the single variable graph looks like minor swings in the second graph compared to NumVar2. This brings up a major point in data visualization. Instead of NumVar2 being a range from ~500 to ~1500, what if it were a range from ~500 to ~15000? Then the line for NumVar2 would basically be a solid black line along the bottom of the graph.  
This shows the importance of having similar ranges in your graphs. if NumVar2 had values up to 15000, and was on the same graph as NumVar1, it would make the data for NumVar1, which may be vastly important, seem insignificant to NumVar2, or indeterminant in that graph. It is important that the data is able to be read, even if it means making 2 graphs instead of one.

## Density Plots

This is repeating what we did in the single variable density plot, but adding again a second variable.

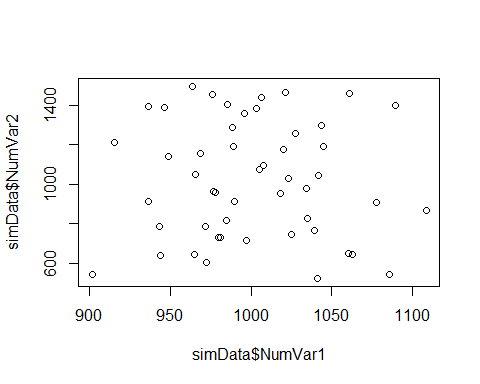
## Let's draw density plots : https://stat.ethz.ch/pipermail/r-help/2006-August/111865.html  
dv1=density(simData$NumVar1)   
dv2=density(simData$NumVar2)  
plot(range(dv1$x, dv2$x),range(dv1$y, dv2$y), type = "n", xlab = "NumVar1(red) and NumVar2 (blue)",  
 ylab = "Density")  
lines(dv1, col = "red")  
lines(dv2, col = "blue")



## Scatterplots

Scatterplots are the main graph used in two numeric variable graphs. Since each of these points has two different numbers for each variable, we plot one of them, NumVar1, on the X axis, and the other, NumVar2, on the Y axis. This type of graph is helpful to see if there is any pattern to the data, or any correlation to the data.  
Since this is two randomly generated data sets, there should be no correlation.

## scatterplots  
plot(simData$NumVar1,simData$NumVar2)



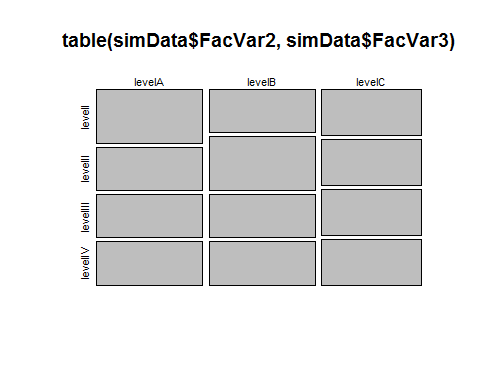
# Two Variables: Factor

These graphs show the relationship between two Qualitative sets of data.

## Mosaic

This type of graph shows the relationship between each level of FacVar2 with each level of FacVar3. Since the relationship is quantified in terms of the total area of the boxes, it is difficult to see which box is bigger than any other box.

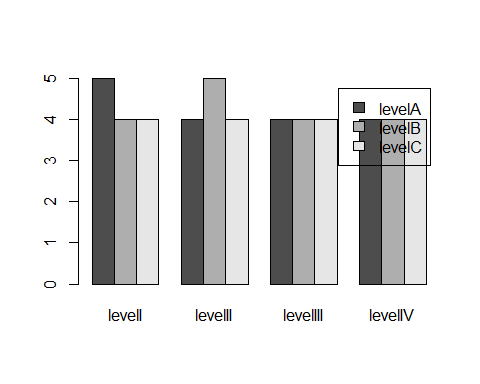
## Mosaic plot  
plot(table(simData$FacVar2,simData$FacVar3))



## Bar Plots

Bar plots are by far the best way to show a relationship between two Qualitative variables. In this bar plot, FacVar2 is shown on the X axis, FacVar3 is each individual bar, and the Y axis is the number of FacVar2's correlates to a variable in FacVar3

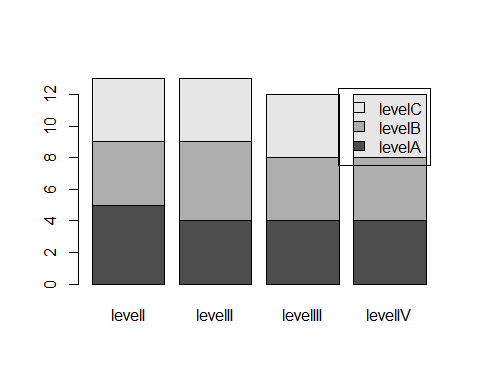
## barplots  
bartable=table(simData$FacVar2,simData$FacVar3) ## get the cross tab  
barplot(bartable,beside=TRUE, legend=levels(unique(simData$FacVar2))) ## plot



## Stacked Bar Plots

Stacked bar plots show the same information as normal bar plots, but stacked. This is a great way to visualize both which set of FacVar2 has the most points, as well as the composition of FacVar3 within FacVar2. However, it is not as descriptive, and it is somewhat hard to determine what portion is larger or smaller than another, so it should not be used to show specific data without labels.

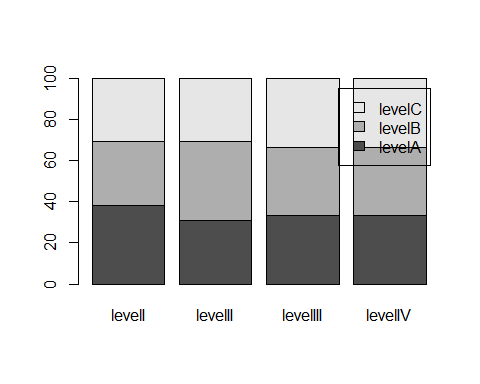
barplot(bartable, legend=levels(unique(simData$FacVar2))) ## stacked



## Stacked Percentage

This stacked bar plot shows what percentage of FacVar3 variables correlate to FacVar2 variables. Without labels, this is a difficult graph to decipher specific percentages, so should not be used to show specific data.

barplot(prop.table(bartable,2)\*100, legend=levels(unique(simData$FacVar2))) ## stacked 100%



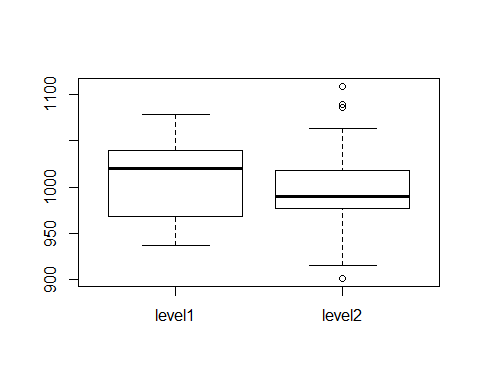
# Two Variables: One Factor, One Numeric

Finding correlation between quantitative and qualitative data is a very important part of any study. For example, any study using both men and women will want to have some sort of distinction in their findings if there is a difference in results between the two. An easy way to determine so is by plotting the quantitative data against the qualitative.

## Box Plots

This box plot uses the same ideas as the one in the single numeric variable. Now, it uses the range of results in NumVar1 as the Y axis, and uses the variables from FacVar1 on the X axis.

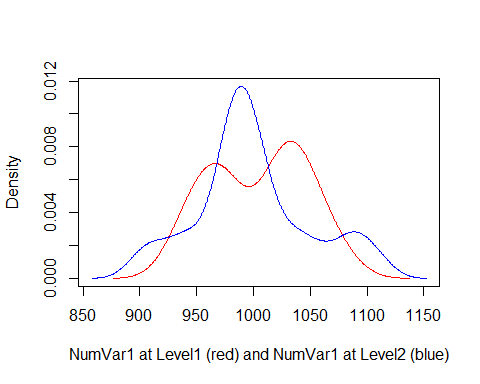
## Box plots for the numeric var over the levels of the factor var  
plot(simData$FacVar1,simData$NumVar1)

 As you can see, there are now outliers! Level 2 has one low outlier, and three high outliers.

## Density Chart

The Density Chart shown here is the same as the one shown in the one numeric variable chart; now, it splits the data between level one and level 2, shown in red and blue respectively.

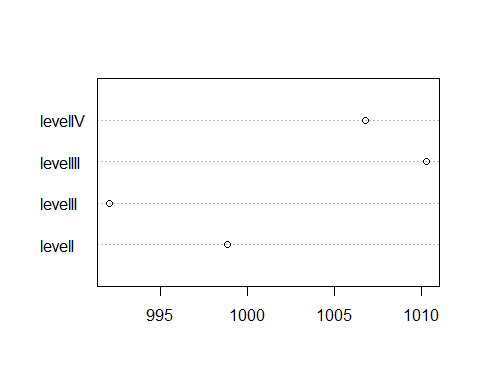
## density plot of numeric var across multiple levels of the factor var  
level1=simData[simData$FacVar1=="level1",]   
level2=simData[simData$FacVar1=="level2",]  
  
dv3=density(level1$NumVar1)  
dv4=density(level2$NumVar1)  
  
plot(range(dv3$x, dv4$x),range(dv3$y, dv4$y), type = "n", xlab = "NumVar1 at Level1 (red) and NumVar1 at Level2 (blue)",ylab = "Density")  
lines(dv3, col = "red")  
lines(dv4, col = "blue")



## Mean Value of Numeric Over Factor

This chart shows what the mean value of the numeric variable is in each of the factor variables.

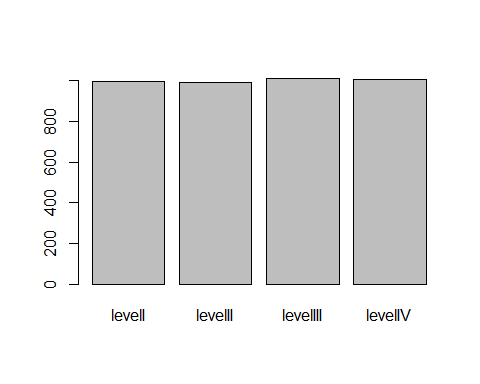
## Mean of one numeric var over levels of one factor var  
meanagg=aggregate(simData$NumVar1, list(simData$FacVar3), mean)  
  
dotchart(meanagg$x,labels=meanagg$Group.1) ## Dot Chart



## Bar Plot

This graph shows the same data as the previous one, but in bar graph form.

barplot(meanagg$x,names.arg=meanagg$Group.1)## Bar plot



## Question: Is a bar plot even appropriate when displaying a mean--- a point?

Excellent question, hypothetical student in the front row! Looking at these two graphs, which one is easier to get information from? Which is more accurate? Which level in the Bar Graph has the highest mean? The lowest?  
Bar Graphs are great for many things; this is not one of them. Using Tufte's theory, use as little ink as possible that is not necessary. With that philosophy, it is much better to plot the points instead of using the bar graphs.

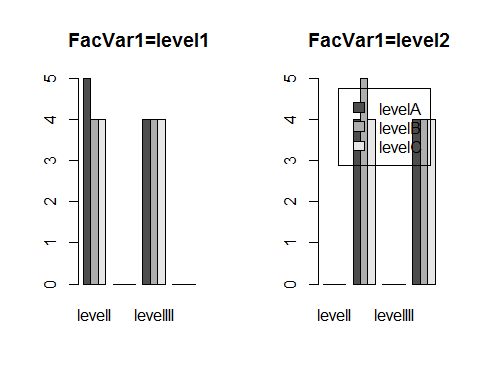
# Three Variables: Three Factor

The more the merrier as far as I'm concerned!

## Double Bar

Double bar plot has 2 bar plots, with FacVar3 as the X axis on both graphs, the bars themselves as facVar2, and each individual graph is a variable in FacVar1. Each bar is the number of points in both FacVar 3 and FacVar 2, in either FacVar level 1 or 2.

par(mfrow=c(1,2))  
  
bar1table=table(level1$FacVar2,level1$FacVar3)  
barplot(bar1table,beside=TRUE, main="FacVar1=level1")  
  
bar2table=table(level2$FacVar2,level2$FacVar3)  
barplot(bar2table,beside=TRUE, main="FacVar1=level2", legend=levels(unique(level2$FacVar2)))



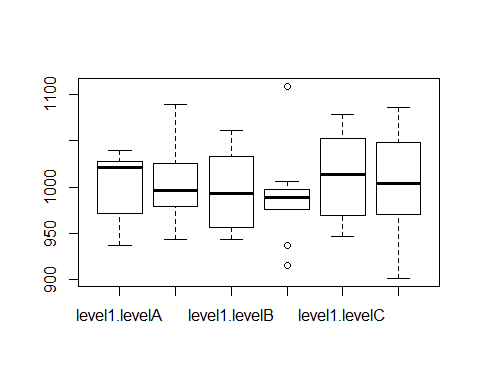
# Three Variables: Two Factor, One Numeric

This allows us to compare two factor variables in relation to one numeric. For example, test scores among men and women, who have either blonde or brown hair.

## Box Plot

Same set up as before, but now each box correlates to a different factor variable in each of the numeric variables.

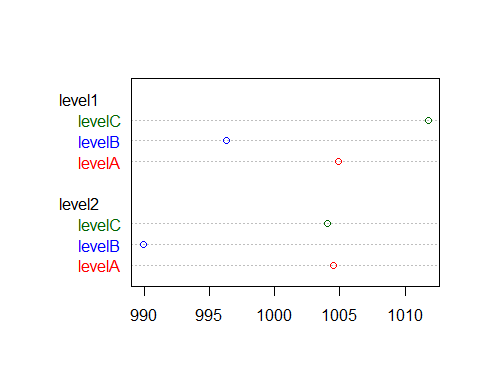
par(mfrow=c(1,1))  
## boxplot of NumVar1 over an interaction of 6 levels of the combination of FacVar1 and FacVar2  
boxplot(NumVar1~interaction(FacVar1,FacVar2),data=simData)



## Mean Plot

Mean plot, but now we have 2 factor variables, so 2 different graphs of means here in one graph.

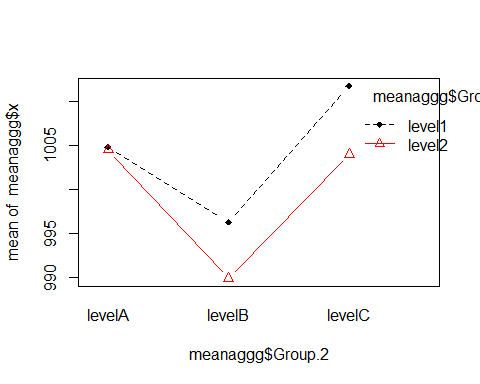
## Mean of 1 Numeric over levels of two factor vars  
meanaggg=aggregate(simData$NumVar1, list(simData$FacVar1,simData$FacVar2), mean)  
meanaggg=meanaggg[order(meanaggg$Group.1),]  
meanaggg$color[meanaggg$Group.2=="levelA"] = "red"  
meanaggg$color[meanaggg$Group.2=="levelB"] = "blue"  
meanaggg$color[meanaggg$Group.2=="levelC"] = "darkgreen"   
  
dotchart(meanaggg$x,labels=meanaggg$Group.2, groups=meanaggg$Group.1,color=meanaggg$color) ## dotchart



## Line Plot of Means

This is a line plot of the means, between each Factor variable sorted with each numeric variable.

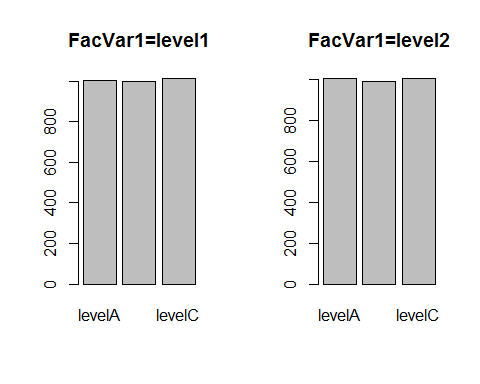
interaction.plot(meanaggg$Group.2,meanaggg$Group.1,meanaggg$x,type="b", col=c(1:2),pch=c(18,24)) ## interaction plot - line plots of means



## Dual Bar Plot

Once again, the means in a bar plot set up. Not nearly as informative as the mean chart or mean line chart.

## some a bar plot  
par(mfrow=c(1,2))  
  
level1=meanaggg[meanaggg$Group.1=="level1",]  
level2=meanaggg[meanaggg$Group.1=="level2",]  
  
barplot(level1$x,names.arg=level1$Group.2, main="FacVar1=level1")  
barplot(level2$x,names.arg=level2$Group.2, main="FacVar1=level2")



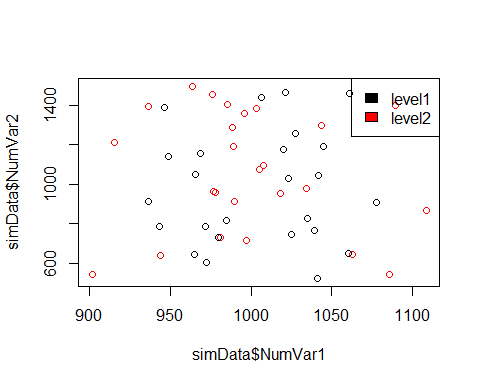
# Three Variables: Two Numeric, One Factor

We like numbers, lets compare them with one factor!

## Scatterplot

This scatterplot between NumVar1 and NumVar2 is the same graph as we saw earlier; now, we have two different colors showing the corresponding FacVar1 data to each point.

## Scatter plot with color identifying the factor variable  
par(mfrow=c(1,1))  
plot(simData$NumVar1,simData$NumVar2, col=simData$FacVar1)  
legend("topright",levels(simData$FacVar1),fill=simData$FacVar1)

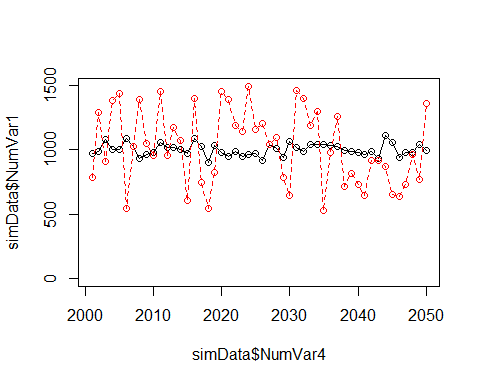


# Three Variables: Three Numeric

## Time Series

We are now using the graph we have in the two numeric variable section under index, and we have put NumVar4, or a time series set, against it. Now, we have a year by year change in both NumVar1 and 2.

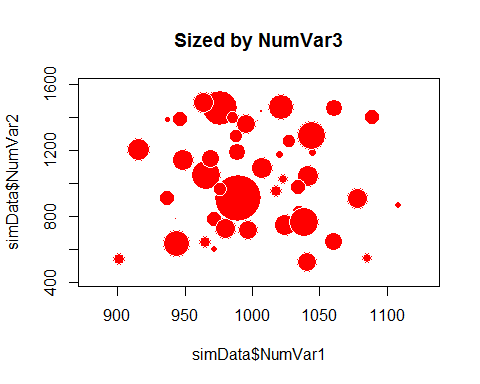
## NumVar4 is 2001 through 2050... possibly, a time variable - use that as the x-axis  
plot(simData$NumVar4,simData$NumVar1,type="o",ylim=c(0,max(simData$NumVar1,simData$NumVar2)))## join dots with lines  
  
lines(simData$NumVar4,simData$NumVar2,type="o",lty=2,col="red")## add another line



## Bubble Plot

A bubble plot works by having the scatterplot of NumVar1 and 2, then adding the corresponding data from NumVar3 and changing the size of each of the points in the set. This makes it so we can relate the information from NumVar3 in the scatterplot correlation from NumVar1 and 2.

## Bubble plot - scatter plot of NumVar1 and NumVar2 with individual observations sized by NumVar3  
# http://flowingdata.com/2010/11/23/how-to-make-bubble-charts/  
radius <- sqrt( simData$NumVar3/ pi )  
symbols(simData$NumVar1,simData$NumVar2,circles=radius, inches=.25,fg="white", bg="red", main="Sized by NumVar3")



## Matrix Of Scatterplots

This is a matrix of all of the scatterplots between two of the numerical variables we have. It looks pretty cool, but unless you are comparing all of the graph in a particular row or column, it is hard to relate one scatterplot to the next.

pairs(simData[,4:7], col=simData$FacVar1)

