

Base R Plots

Joey Miranda

9/16/2014

Purpose

Assignment 3: This document shows various strings of code with the corresponding graphs to get a better understanding of how to use the graphs with RMarkdown

Data

```
## Simulate some data

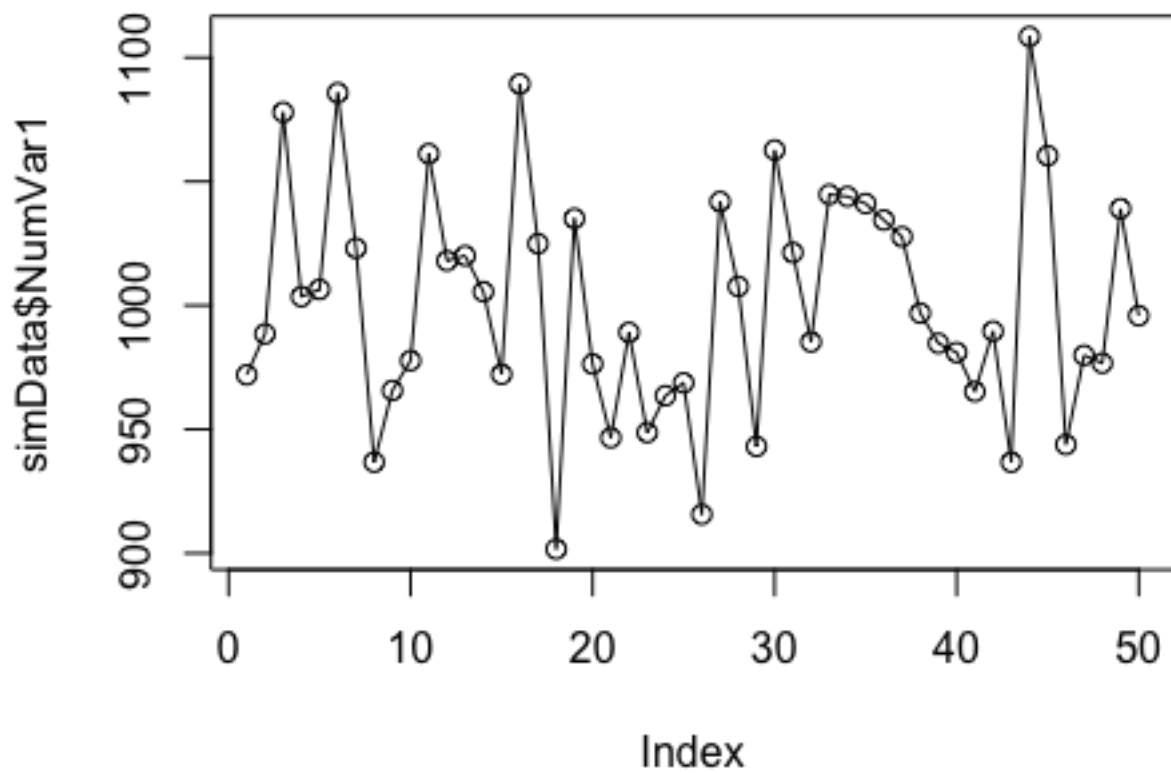
## 3 Factor Variables
FacVar1=as.factor(rep(c("level1","level2"),25))
FacVar2=as.factor(rep(c("levelA","levelB","levelC"),17)[-51])
FacVar3=as.factor(rep(c("levelI","levelII","levelIII","levelIV"),13)[-c(51:52)])

## 4 Numeric Variables
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

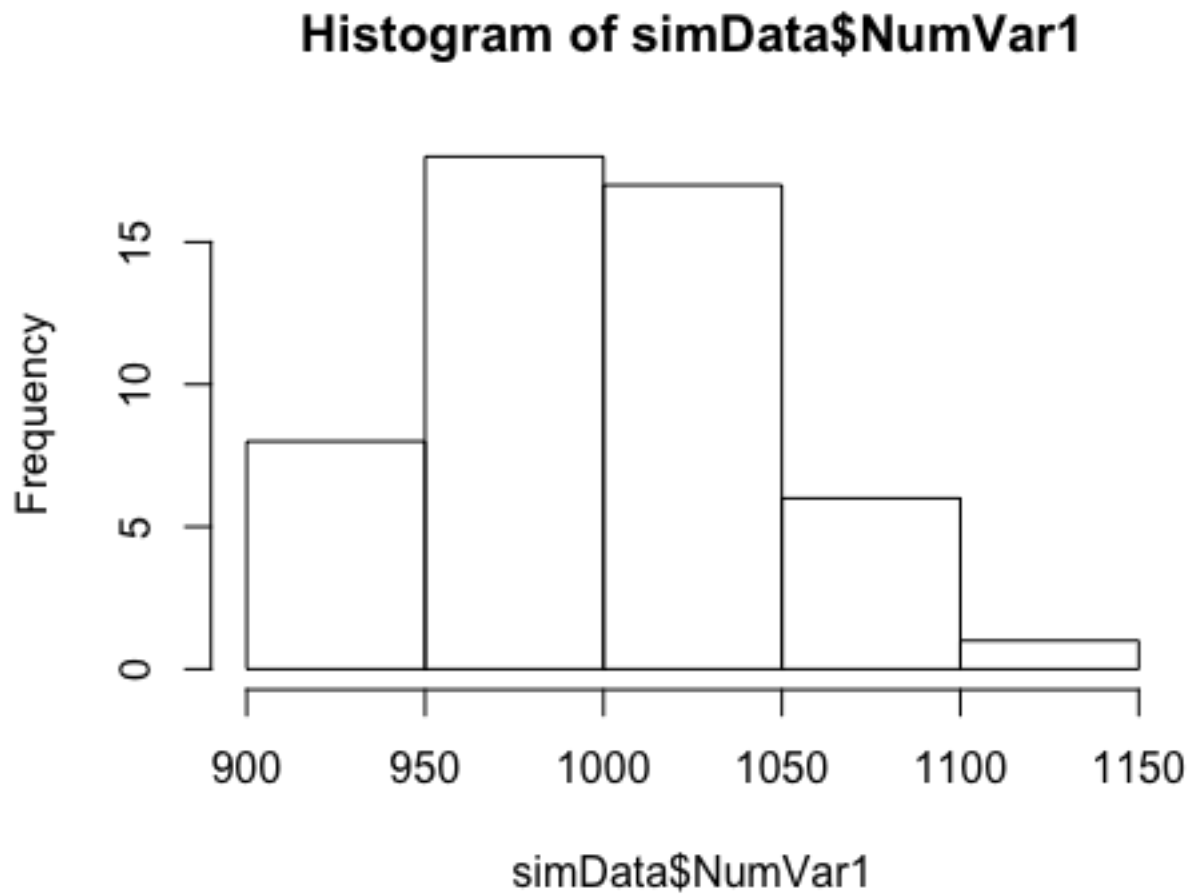
simData=data.frame(FacVar1,FacVar2,FacVar3,NumVar1,NumVar2,NumVar3,NumVar4)
```

Plotting one Variable: Numeric Variable

```
plot(simData$NumVar1,type="o") ## Index plot, 'type="o"' adds the line  
connecting each data point, 'plot("dataset")' is one of the most common  
of RMarkdown code for graphing a numeric variable dataset
```

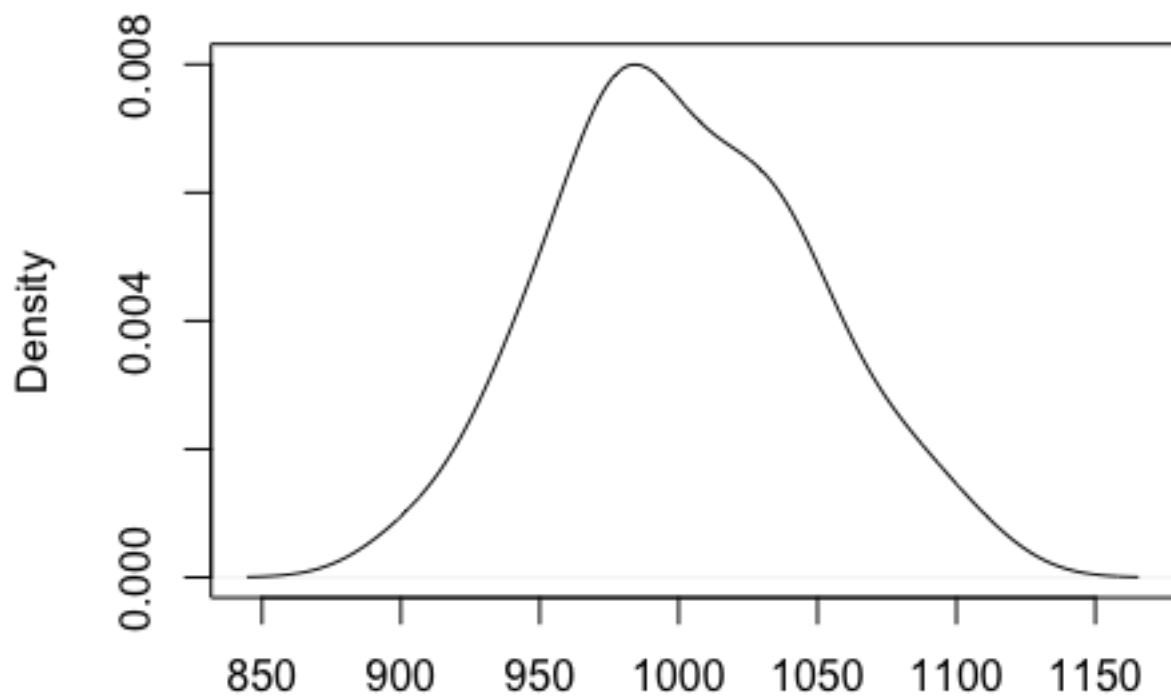


```
hist(simData$NumVar1) ## Histogram, replacing 'plot' with 'hist' using  
the same data set will create a Histogram instead of a data plot graph
```



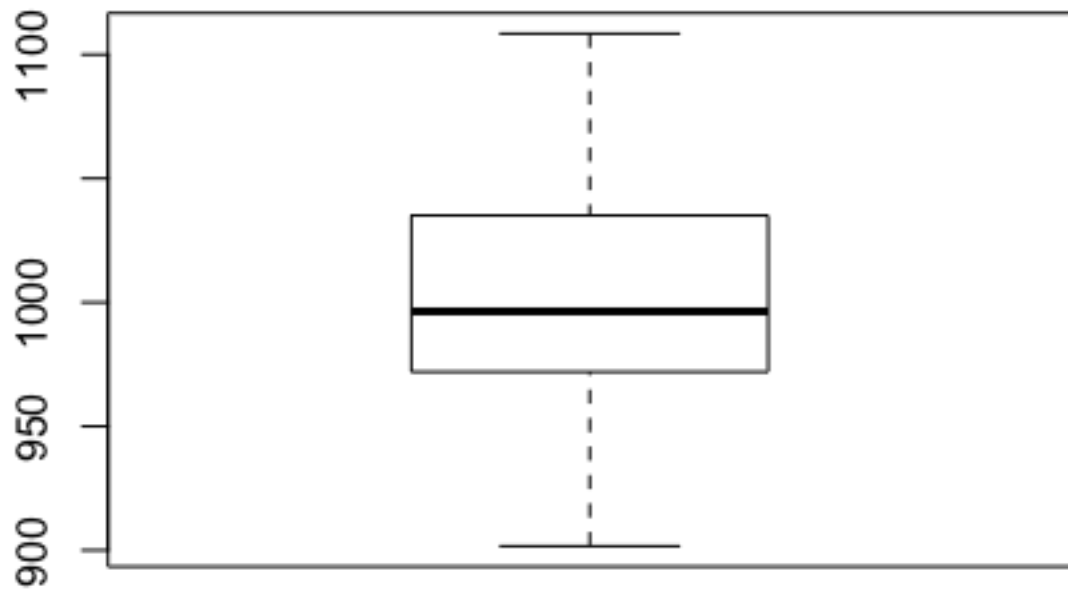
```
plot(density(simData$NumVar1)) ## Kernel density plot: adding  
'(density("dataset"))' to data plot will create a Kernel density  
plot, used to see a distribution of a variable
```

density.default(x = simData\$NumVar1)



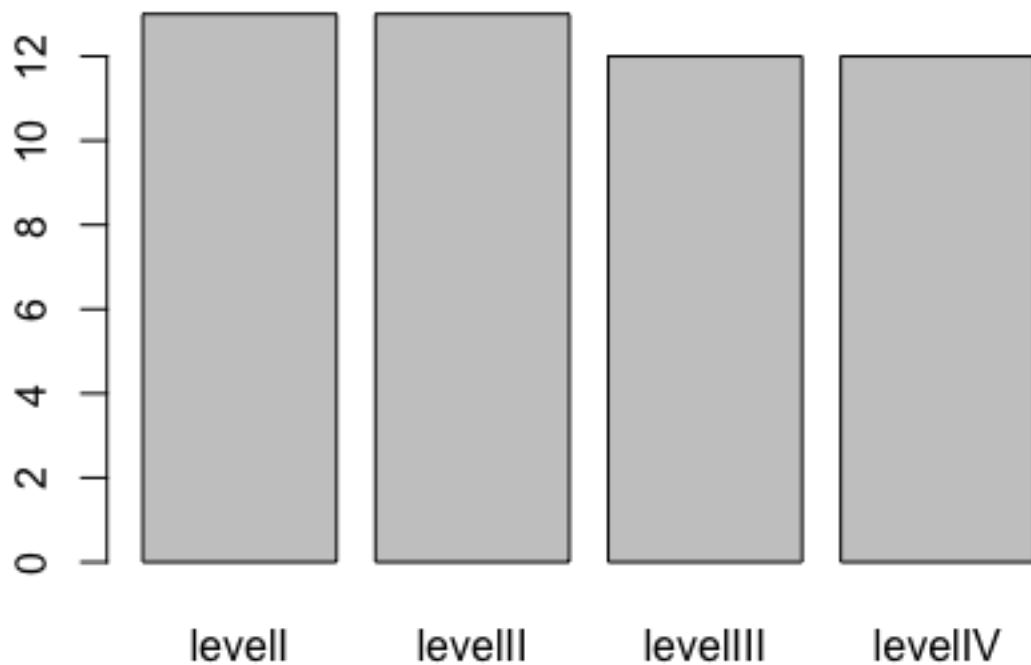
N = 50 Bandwidth = 19.05

```
boxplot(simData$NumVar1) ## box plot: adding 'box' outside of plot on a  
plot graph will create a box chart, used to visualize 25-75%  
distribution and also the remaining distribution extremes
```

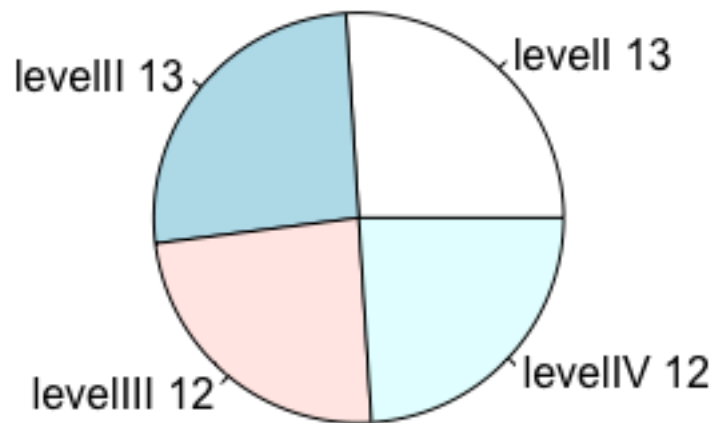


Plotting one variable: Factor variable

```
plot(simData$FacVar3) ## bar plot: stickig to the plot style but using  
a data set with factor variables will automatically create the most  
logical graph with is a bar plot
```

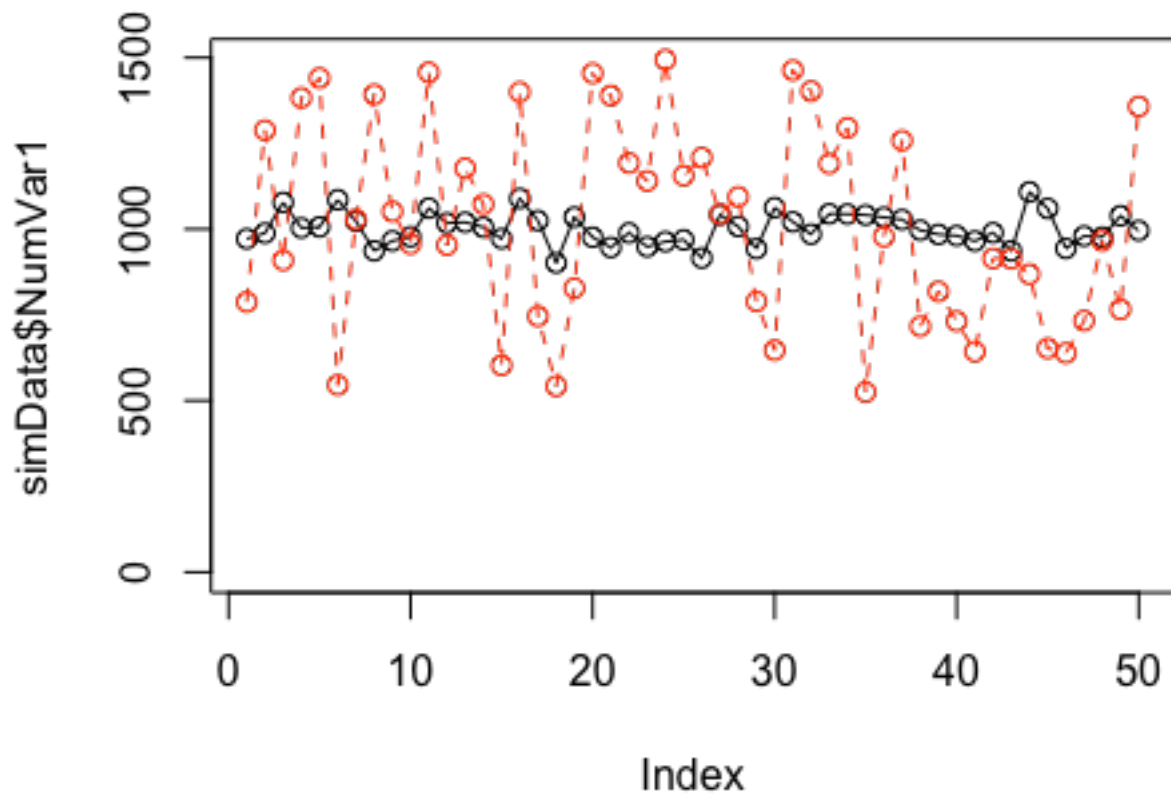


```
## pie chart - Not the best graph --- use with caution
counts=table(simData$FacVar3) ## get counts: this searches the dataset
for how many variables it is using
labs=paste(simData$FacVar3,counts)## create labels: this will take the
names of the variables and attach them to each variable from 'counts'
pie(counts,labels=labs) ## plot: this will plot the pie chart using the
variables from 'counts' and display the labels from 'labs'
```

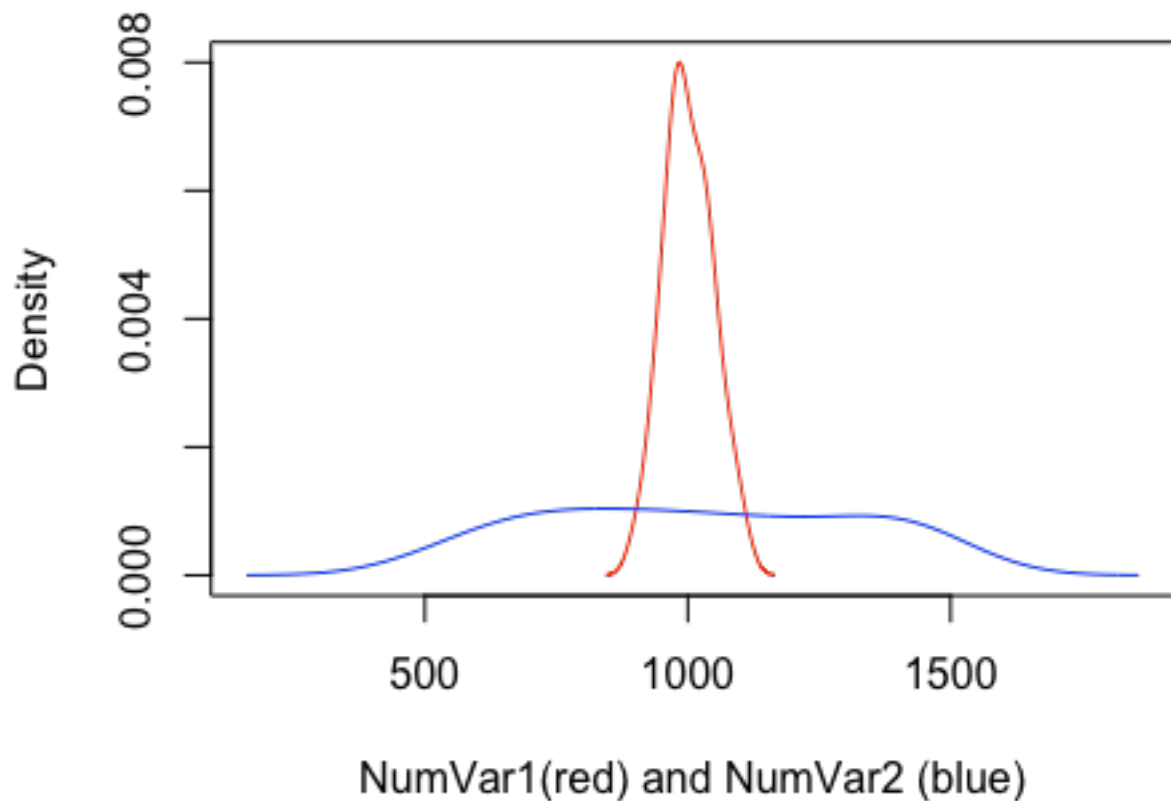


Two Variables: Two Numeric Variables

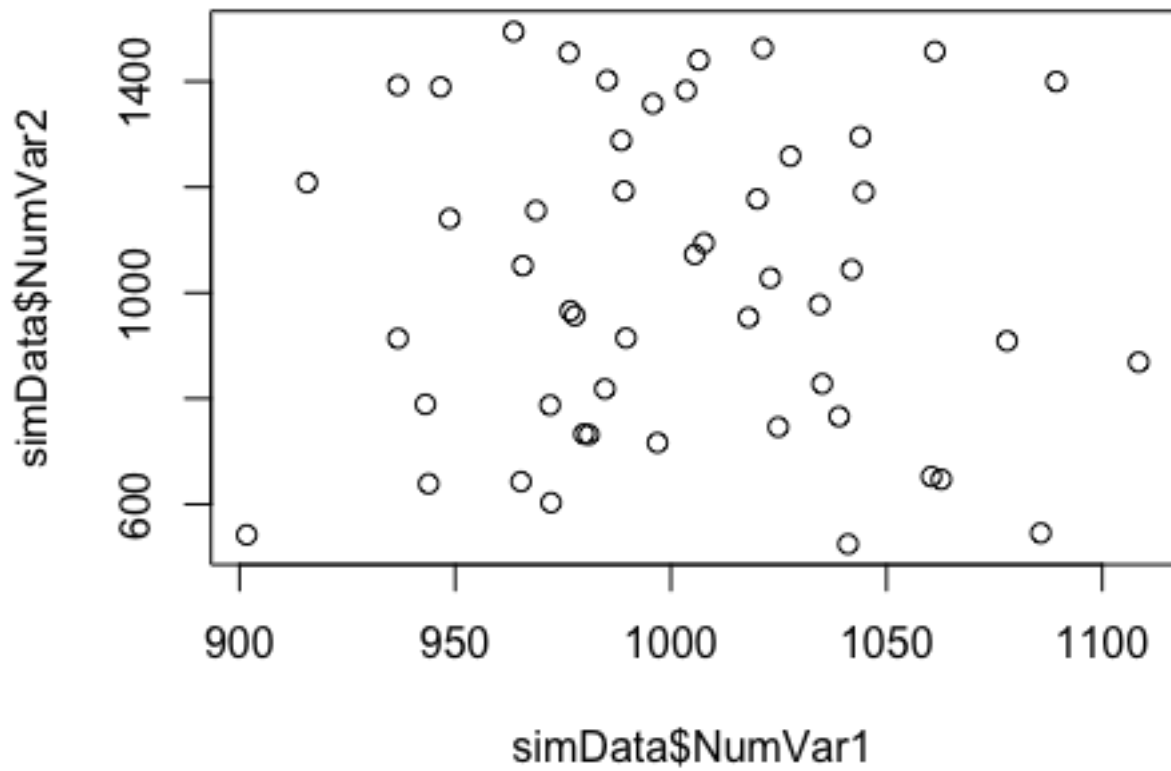
```
plot(simData$NumVar1,type="o",ylim=c(0,max(simData$NumVar1,simData$NumVar2)))## index plot with one variable: creates a normal lined plot chart but extends the y limit to fit another data set as well  
lines(simData$NumVar2,type="o",lty=2,col="red")## add another variable: adds the second dataset over the first one and colors it red
```




```
## Density plots
dv1=density(simData$NumVar1)
dv2=density(simData$NumVar2) ##these assign each density chart to a
variable 'dv1, dv2'
plot(range(dv1$x, dv2$x),range(dv1$y, dv2$y), type = "n", xlab =
"NumVar1(red) and NumVar2 (blue)",ylab = "Density") ##this plots each
density chart on one graph and identifies each line with a color in the
x axis and marks density on the y axis
lines(dv1, col = "red")
lines(dv2, col = "blue") ## these change each density plot to a
designated color
```



```
plot(simData$NumVar1,simData$NumVar2) ## scatterplots: this is a simple  
scatterplot, just use the plot function and use two numeric variable  
sets seperated with a comma
```



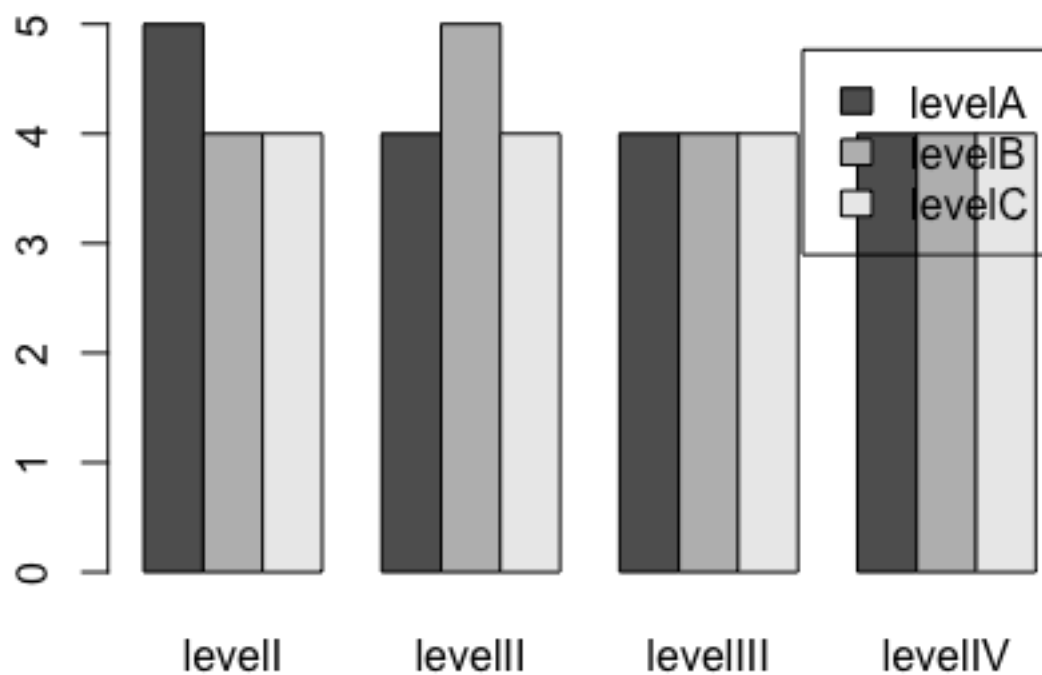
Two Variables: Two Factor Variables

```
plot(table(simData$FacVar2,simData$FacVar3)) ## Mosaic plot: using the  
plot function and adding 'table("datasets")' to it while using two  
factor variable sets seperated by a comma, this compares relationships  
among all the variables
```

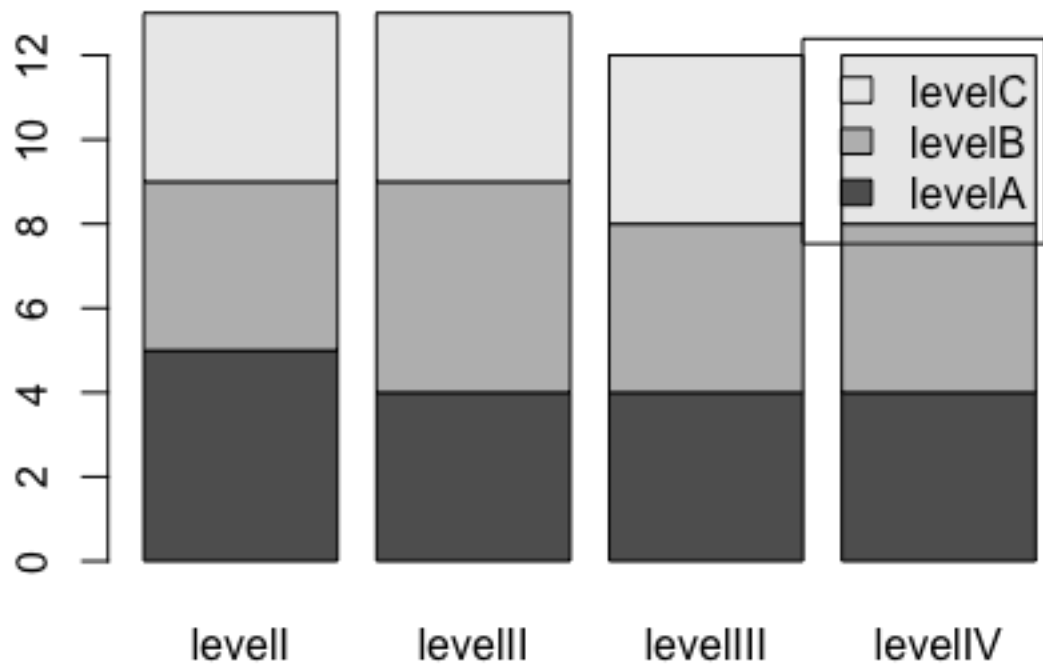
table(simData\$FacVar2, simData\$FacVar3)

	levelA	levelB	levelC
levelI			
levelII			
levelIII			
levelIV			

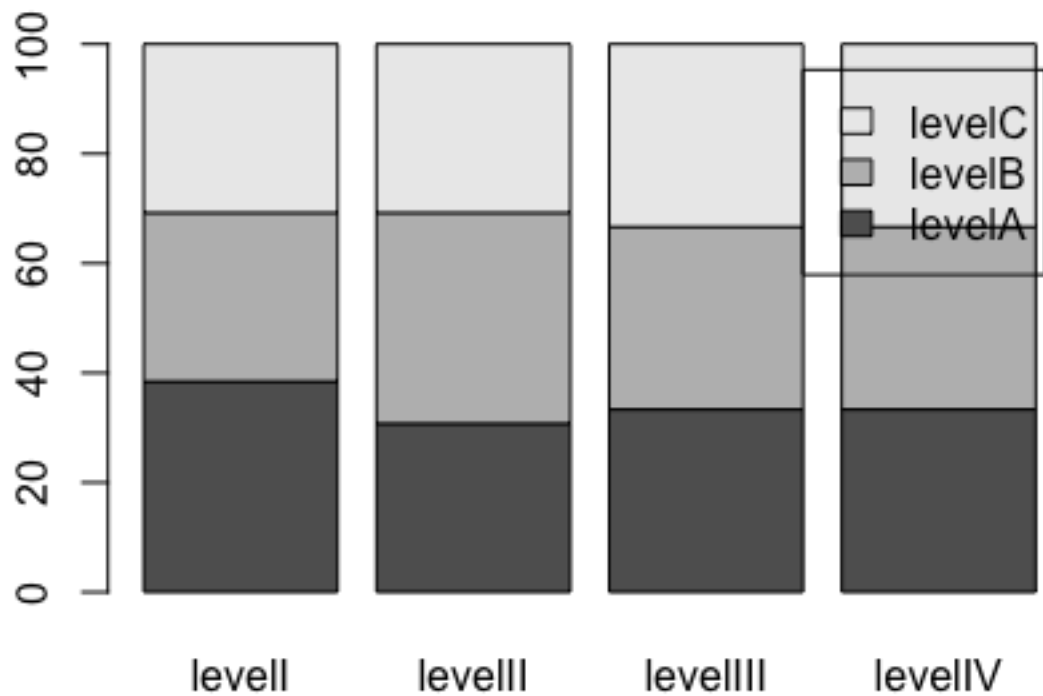
```
## barplots
bartable=table(simData$FacVar2,simData$FacVar3) ## get the cross tab:
this grabs the data sets identifies all the points, 'bartable' will be
used for the next as a variable for the next two graphs as well
barplot(bartable,beside=TRUE, legend=levels(unique(simData$FacVar2)))
## plot: this plots 'bartable' placing each variable in a set next to
each other('beside=TRUE') and creates a legend using the dataset to
identify each bar and relate it
```



```
barplot(bartable, legend=levels(unique(simData$FacVar2))) ## stacked:  
simial to the last barplot but it stacks the variables, note no  
'beside=TRUE' is the only difference
```

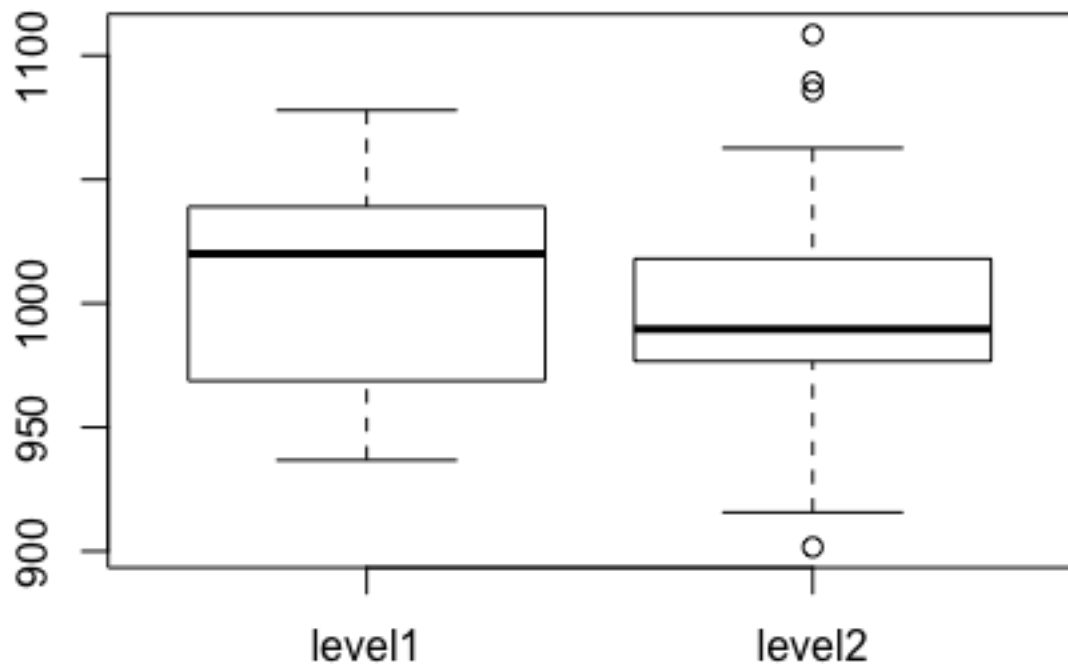


```
barplot(prop.table(bartable,2)*100,  
legend=levels(unique(simData$FacVar2))) ## stacked 100%: also similar  
to the last one but sets the variables out of 100%  
['prop.table(variable,index margin)']
```



Two Variables: One Factor and One Numeric

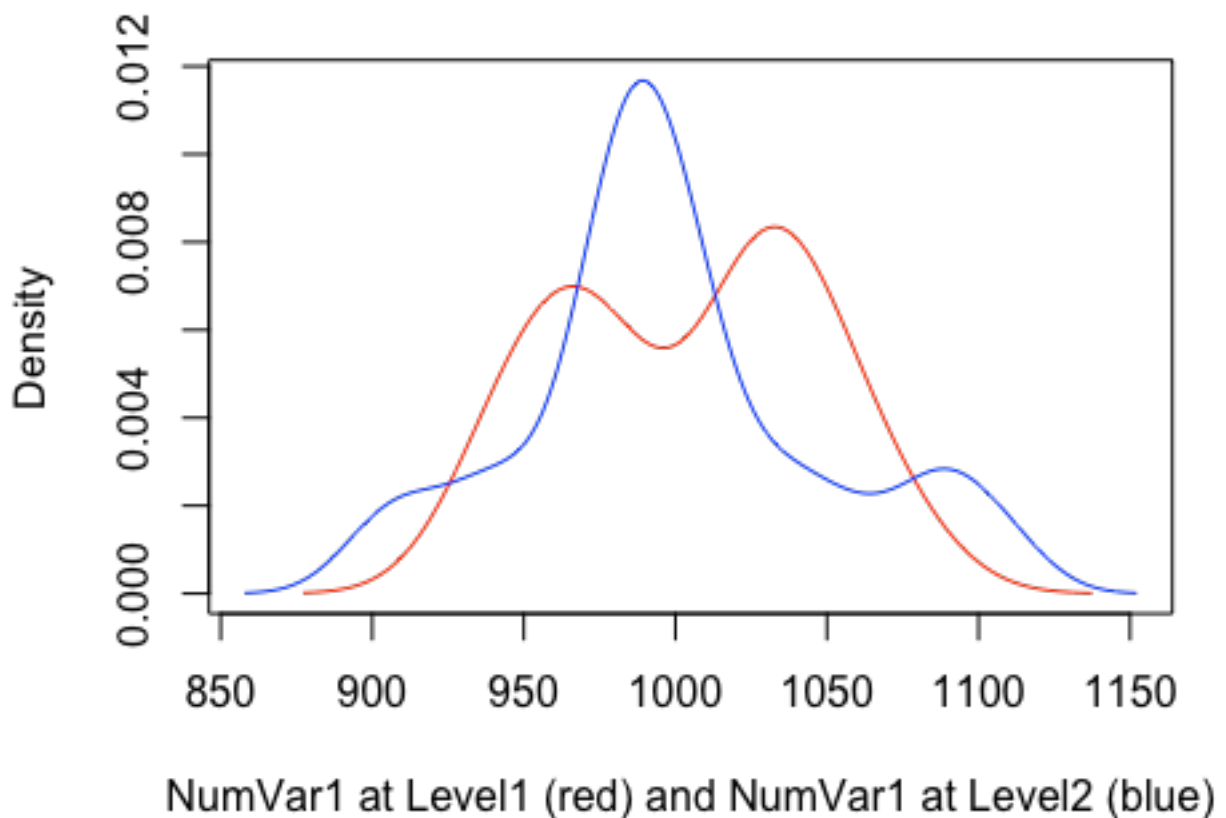
```
## Box plots for the numeric variable over the levels of the factor variable  
plot(simData$FacVar1,simData$NumVar1) ## simply using a data set of  
factor variables and numeric variables will create a box plot comparing  
each factor variable
```



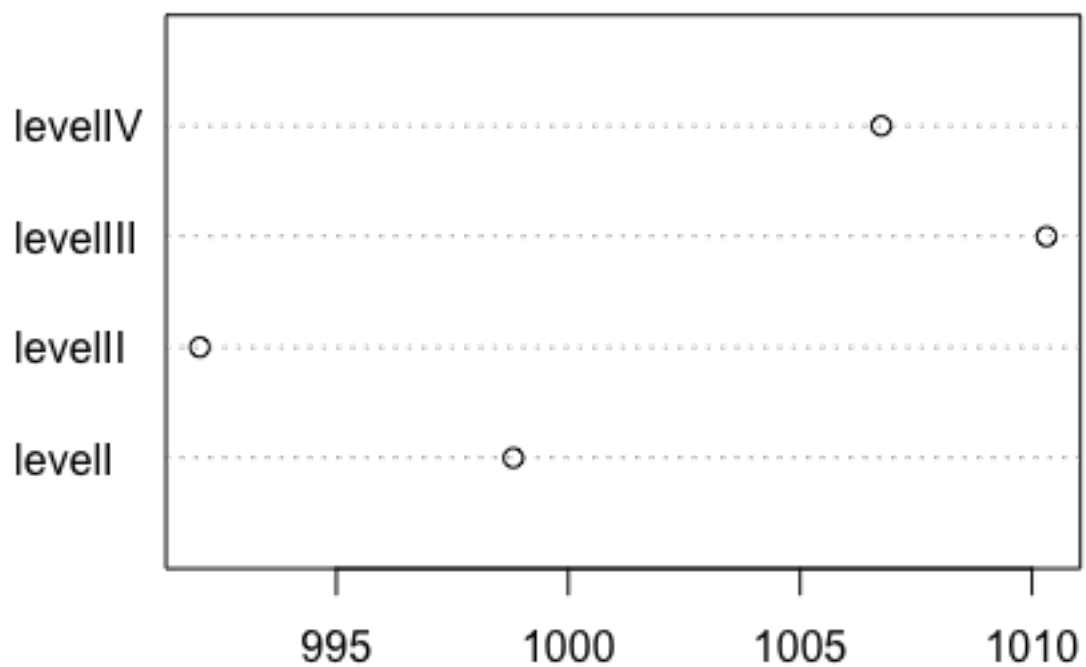
```
## density plot of numeric variables across multiple levels of the
factor variables
level1=simData[simData$FacVar1=="level1",]
level2=simData[simData$FacVar1=="level2",] ##assigns and names each
factor variable dataset to a certain name 'level1, level2'

dv3=density(level1$NumVar1)
dv4=density(level2$NumVar1) ##assigns a density plot from a numeric
variable set 'NumVar1' across both variable sets 'level 1,2'

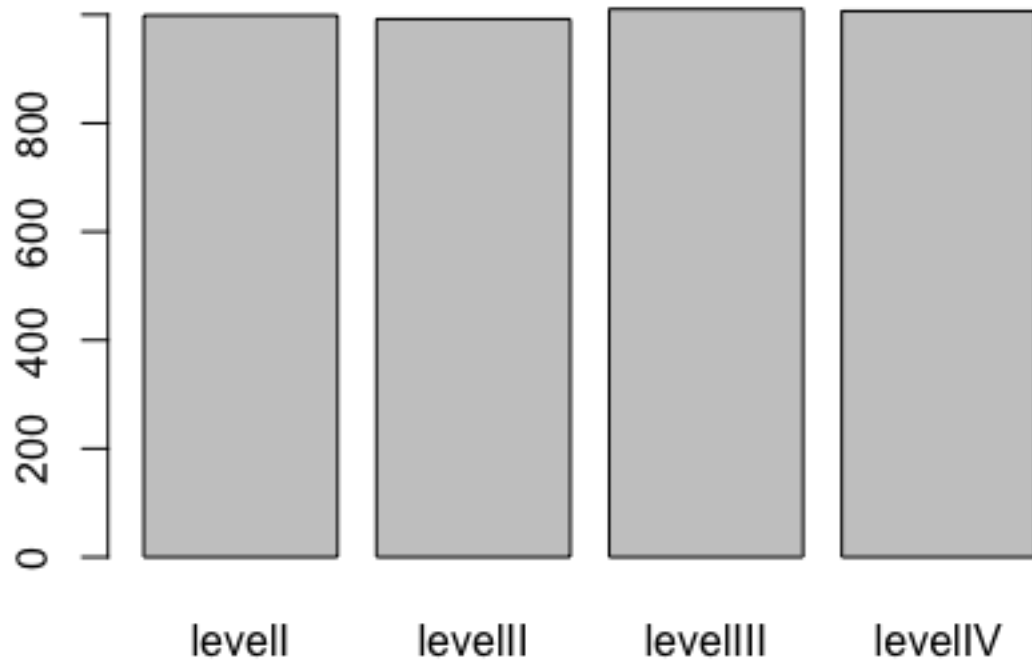
plot(range(dv3$x, dv4$x),range(dv3$y, dv4$y), type = "n", xlab =
"NumVar1 at Level1 (red) and NumVar1 at Level2 (blue)",ylab =
"Density")
lines(dv3, col = "red")
lines(dv4, col = "blue") ##plots the graph, and marks each axis and
colors each data line
```




```
## Mean of one numeric variable over levels of one factor variable  
meanagg=aggregate(simData$NumVar1, list(simData$FacVar3), mean) ##  
using 'aggregate(dataset)' will split the data into subsets, lists  
'list()' them according to each factor variable, and finds the mean  
'mean' of each numeric variable in relation to the factor variable  
  
dotchart(meanagg$x, labels=meanagg$Group.1) ## Dot Chart: plots the  
chart along with labeling the y axis with each factor variable
```



```
barplot(meanagg$x, names.arg=meanagg$Group.1)## Bar plot: using the same
variables 'meanagg$x, meanagg$Group.1' but instead of dots, it creates
a bar chart. note: this is not an appropriate method for displaying a
mean since the bars are too difficult to discern the differences
```

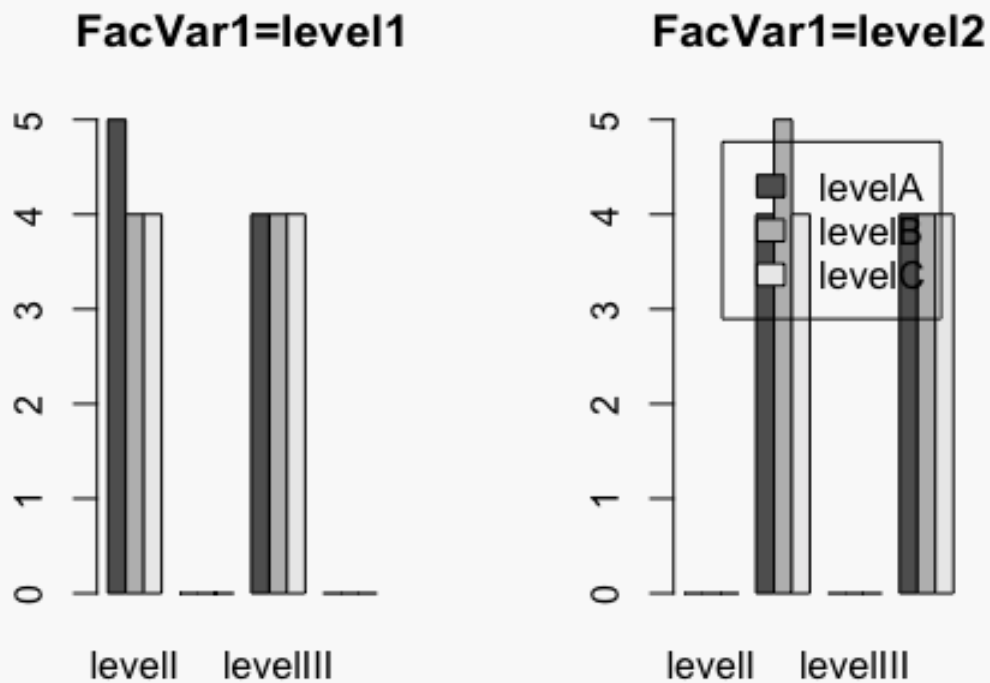


Three Variables: Three Factor Variables

```
par(mfrow=c(1,2)) ##this makes it so both graphs are placed next to each other to help compare easier
```

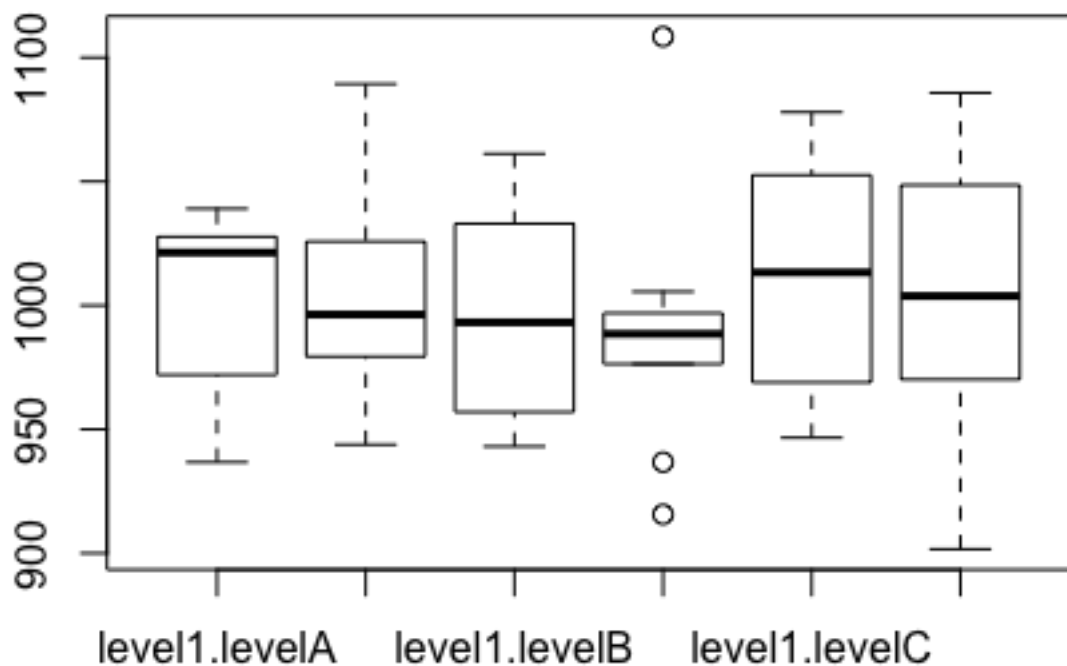
```
bar1table=table(level1$FacVar2,level1$FacVar3)  
barplot(bar1table,beside=TRUE, main="FacVar1=level1")
```

```
bar2table=table(level2$FacVar2,level2$FacVar3)  
barplot(bar2table,beside=TRUE, main="FacVar1=level2",  
legend=levels(unique(level2$FacVar2))) ##these create two barplots that display the levels or two different factor variables 'FacVar2,3' to another different factor variable 'level1,2'
```



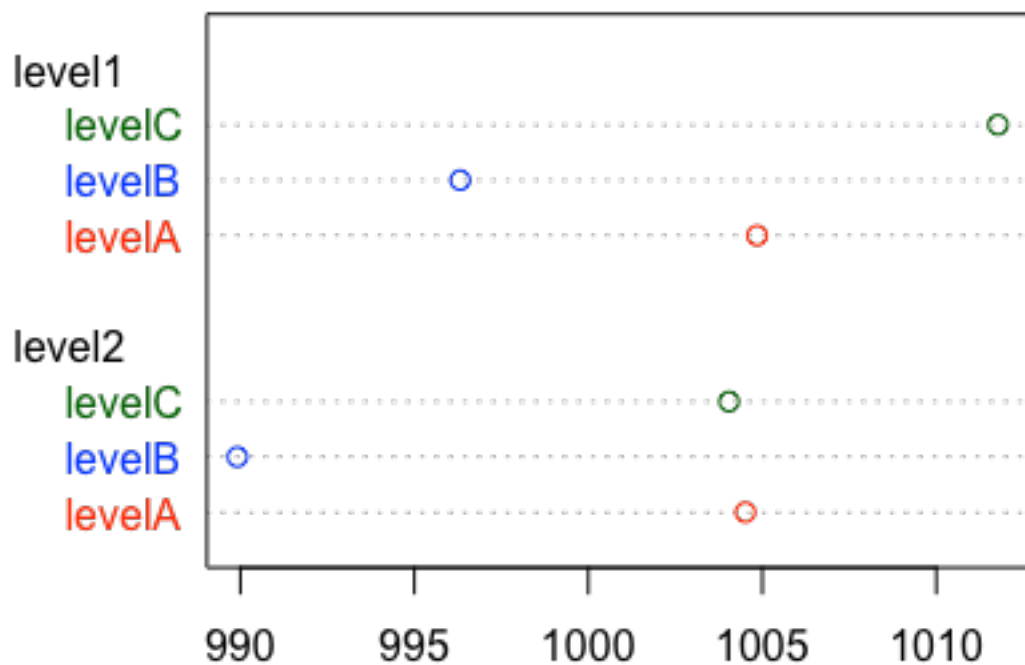
Three Variables: One Numeric and Two Factor Variables

```
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) ##creates a  
box plot of six boxes, each showing the relation of the numeric  
variable 'NumVar1' to both factor variables 'FacVar1,2','interaction'is  
used to find the interaction between the numeric variable and each  
factor variable
```

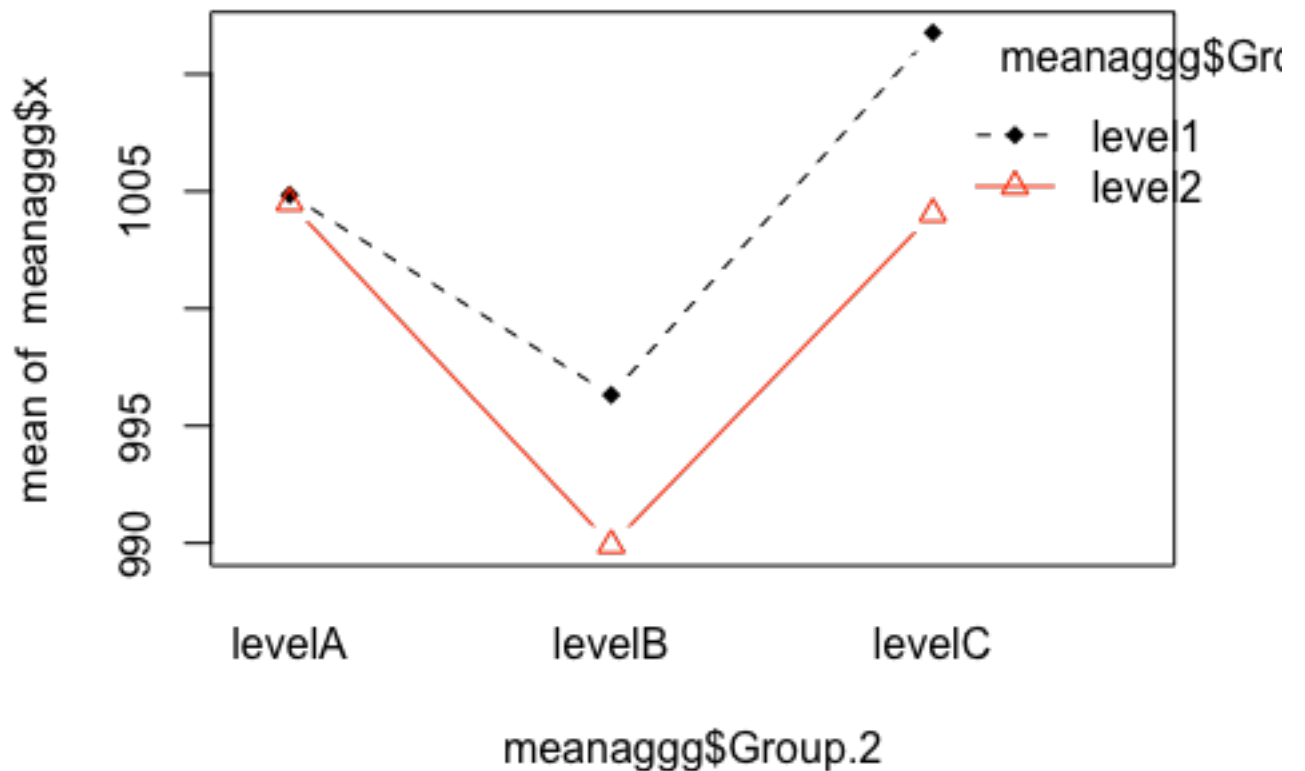


```
## Mean of 1 Numeric over levels of two factor vars
meanaggg=aggregate(simData$NumVar1,
  list(simData$FacVar1,simData$FacVar2), mean)
meanaggg=meanaggg[order(meanaggg$Group.1),]
meanaggg$color[meanaggg$Group.2=="levelA"] = "red"
meanaggg$color[meanaggg$Group.2=="levelB"] = "blue"
meanaggg$color[meanaggg$Group.2=="levelC"]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

dotchart(meanaggg$x,labels=meanaggg$Group.2,
  groups=meanaggg$Group.1,color=meanaggg$color) ## dotchart: similar to
the last dot chart displaying a mean, this one takes 'NumVar1' and
finds its mean in relation to both 'FacVar1,2' and places it on one
graph
```



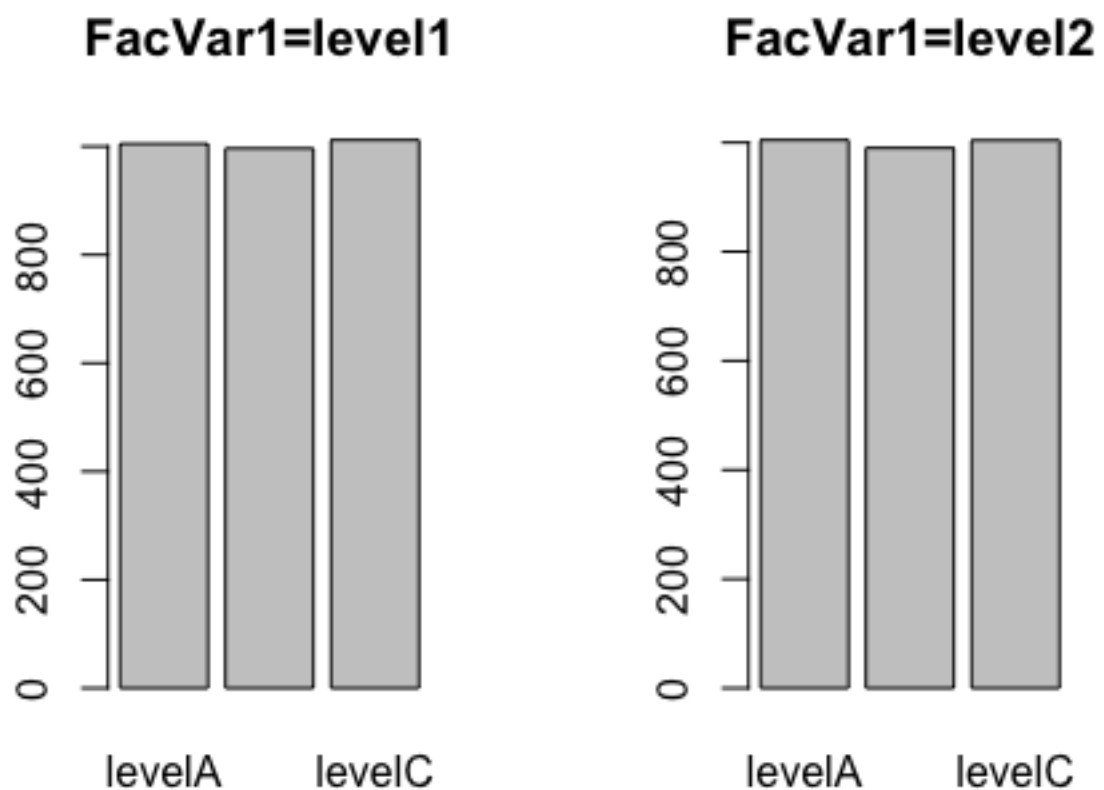
```
interaction.plot(meanaggg$Group.2,meanaggg$Group.1,meanaggg$x,type="b",
col=c(1:2),pch=c(18,24)) ## interaction plot - line plots of means:
creates a line chart that also relates the mean of a numeric variable
to two factor variables, and places the means at data points with a
line connecting each, it is color coded and each data point has a
different symbol based on which factor variable it is for
```



```
## a bar plot
par(mfrow=c(1,2))

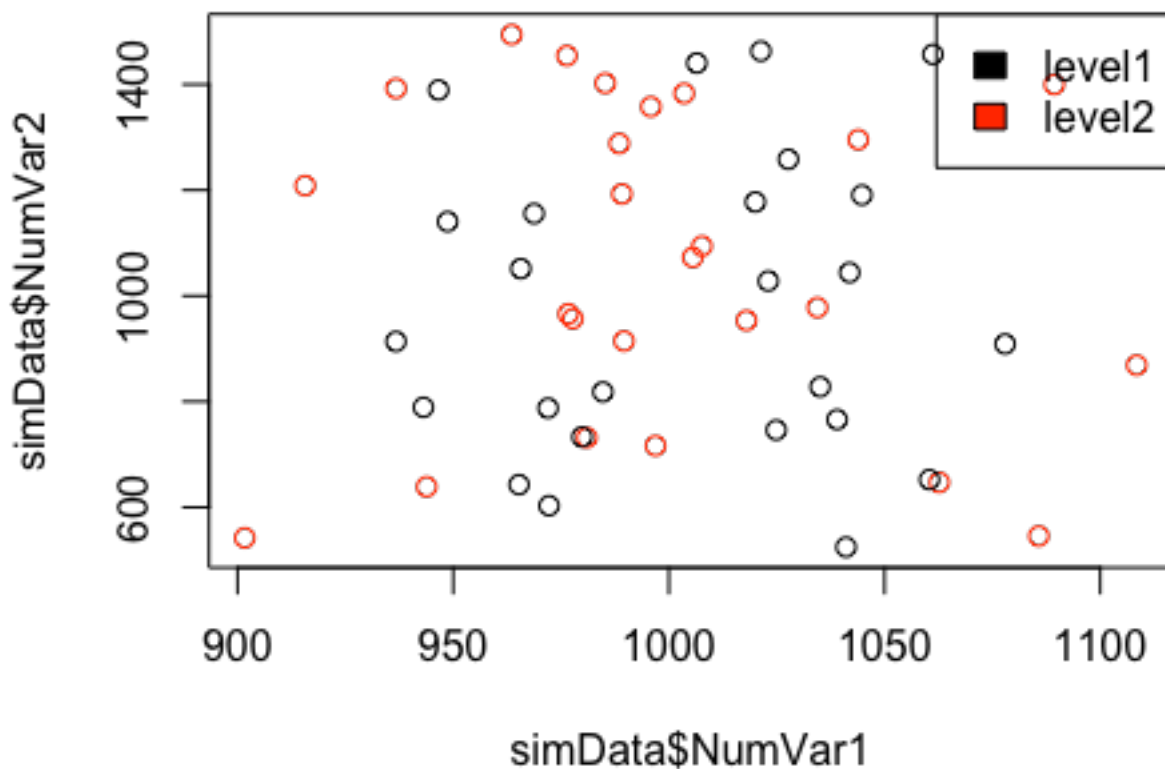
level1=meanaggg[meanaggg$Group.1=="level1",]
level2=meanaggg[meanaggg$Group.1=="level2",]

barplot(level1$x,names.arg=level1$Group.2, main="FacVar1=level1")
barplot(level2$x,names.arg=level2$Group.2, main="FacVar1=level2")
##very similar to the last few graphs and the numeric mean bar chart,
but uses one more factor variable, again note this is not ideal for
displaying means
```



Three Variables: Two Numeric and One Factor Variables

```
## 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) ##this
plots points from both numeric variables in relation to the factor
variable and each factor variable is color coded and uses the legend
identify each factor
```

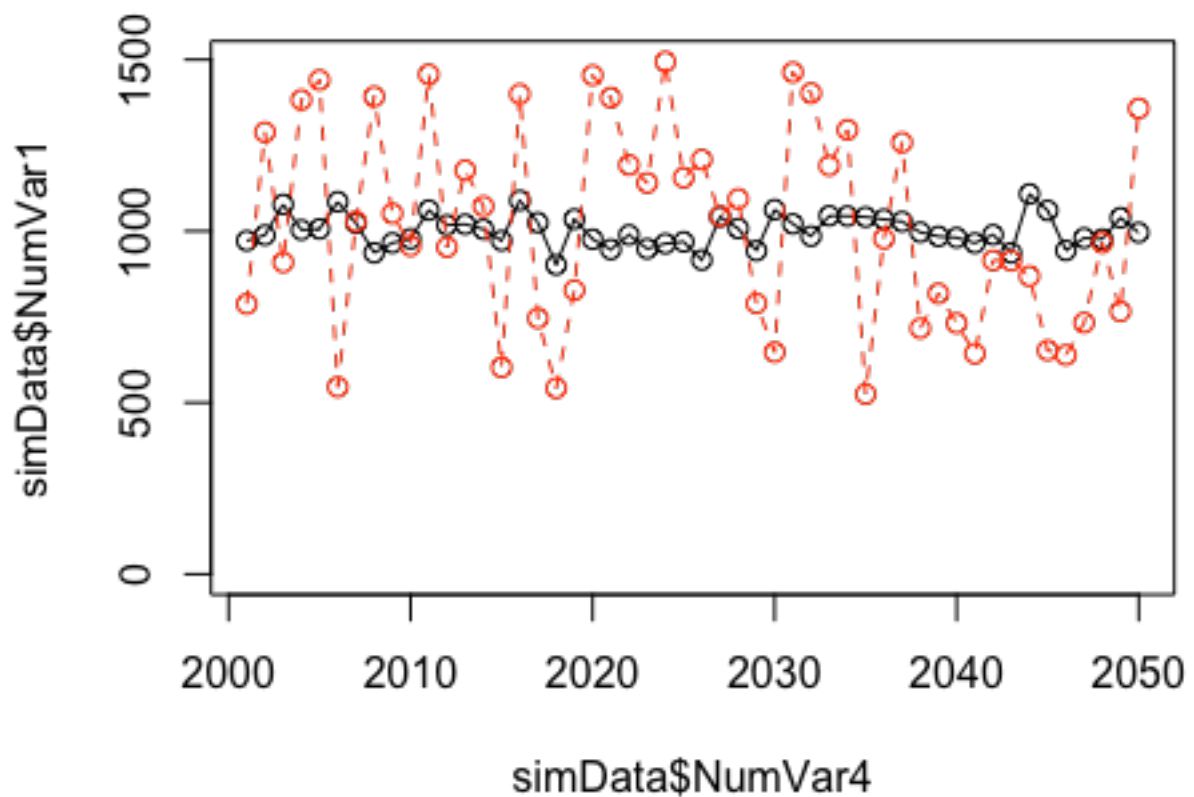


Three Variables: Three Numeric Variables

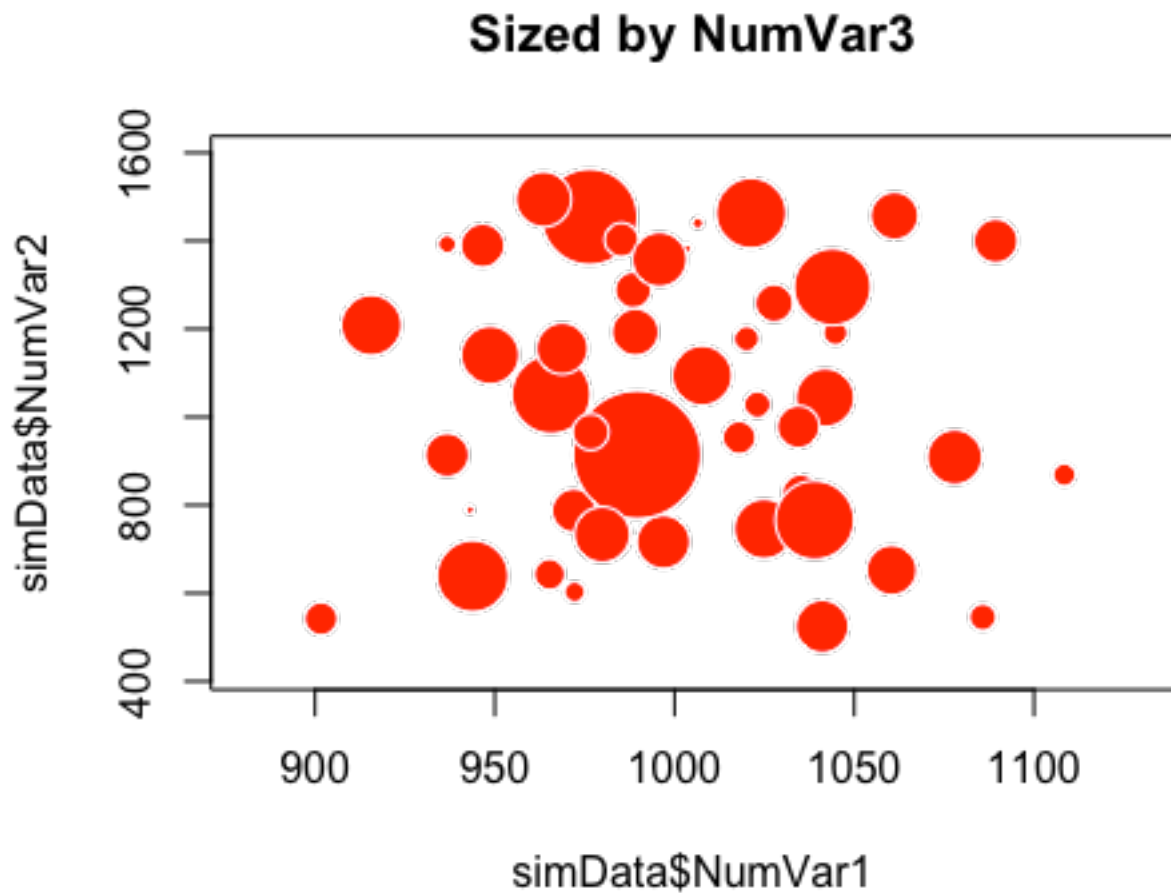
```
## NumVar4 is 2001 through 2050... possibly, a time variable - use that  
as the x-axis: this places two different numeric variables 'NumVar1,2'  
on a grid, each one is connected with a line and color coded, both  
variables are in relation to the third numeric variable 'NumVar4' which  
is places on the x axis
```

```
plot(simData$NumVar4,simData$NumVar1,type="o",ylim=c(0,max(simData$NumV  
ar1,simData$NumVar2)))## join dots with lines
```

```
lines(simData$NumVar4,simData$NumVar2,type="o",lty=2,col="red")## add  
another line
```



```
## Bubble plot - scatter plot of NumVar1 and NumVar2 with individual
observations sized by NumVar3: this places numeric data on a grid
relating two different numeric variables 'NumVar1,2', while identifying
with a third variable 'NumVar3'
radius <- sqrt( simData$NumVar3/ pi ) ##sets the radius of each
variable from NumVar3
symbols(simData$NumVar1,simData$NumVar2,circles=radius,
inches=.25,fg="white", bg="red", main="Sized by NumVar3")
```



Scatterplot Matrix of all Numeric Vars, colored by a Factor variable

```
pairs(simData[,4:7], col=simData$FacVar1) ##creates a scatter plot of every numeric variable compared to each numeric variable in relation to one factor variable 'FacVar1'
```

