R-Markdown Graphs/Charts

Aaron Grenz

Abstract This is a graphing manual for r-markdown plots.

Data Let us begin by simulating our sample data of 3 factor variables and 4 numeric variables.

R-Markdown language:

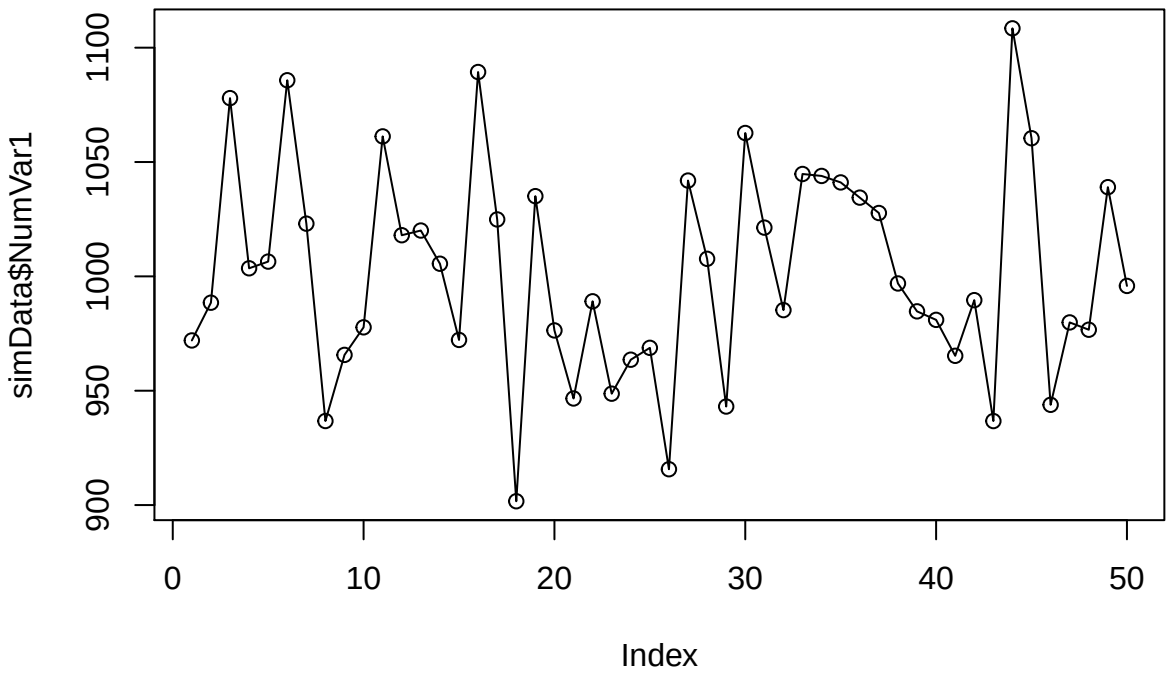
```
## Simulate some data

## 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)
```

What it means:

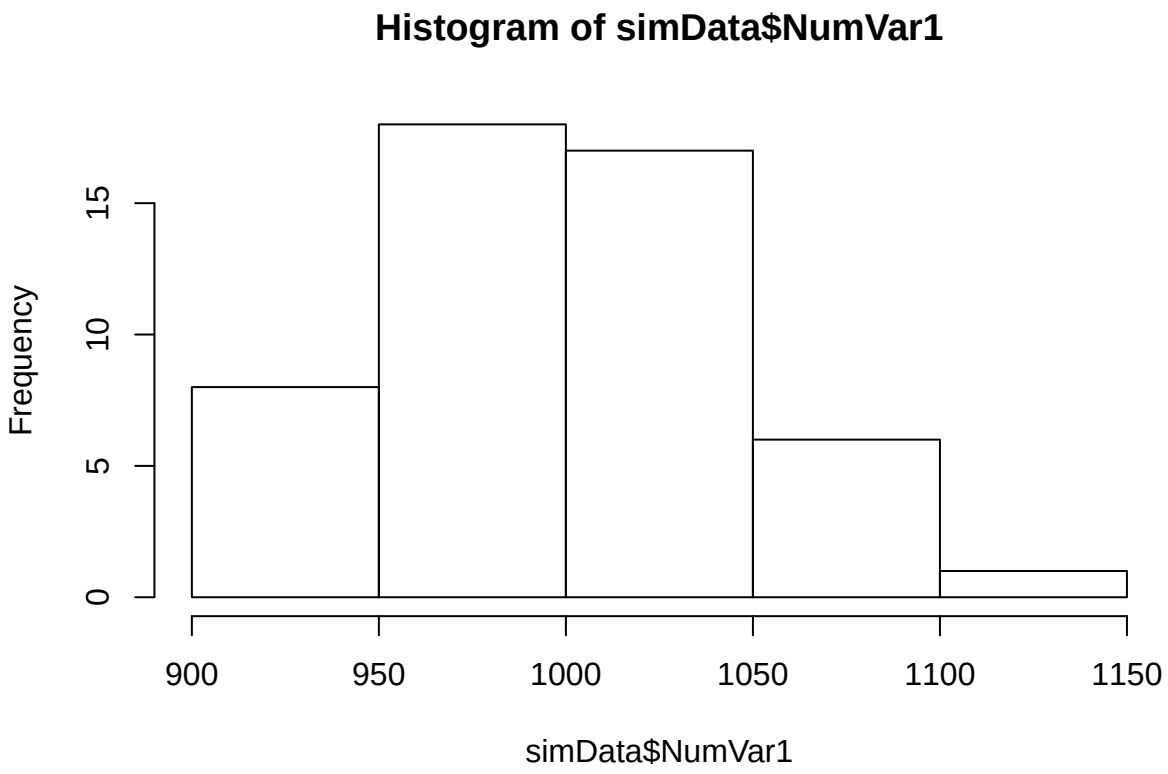
FacVar1, FacVar2, and FacVar3 are factor variables. NumVar1, NumVar2, NumVar3, and NumVar4 are numeric variables. simData represents a variable that combines the factor and numeric variables into a matrix type table.

One Variable: Numeric Variable R-Markdown language:

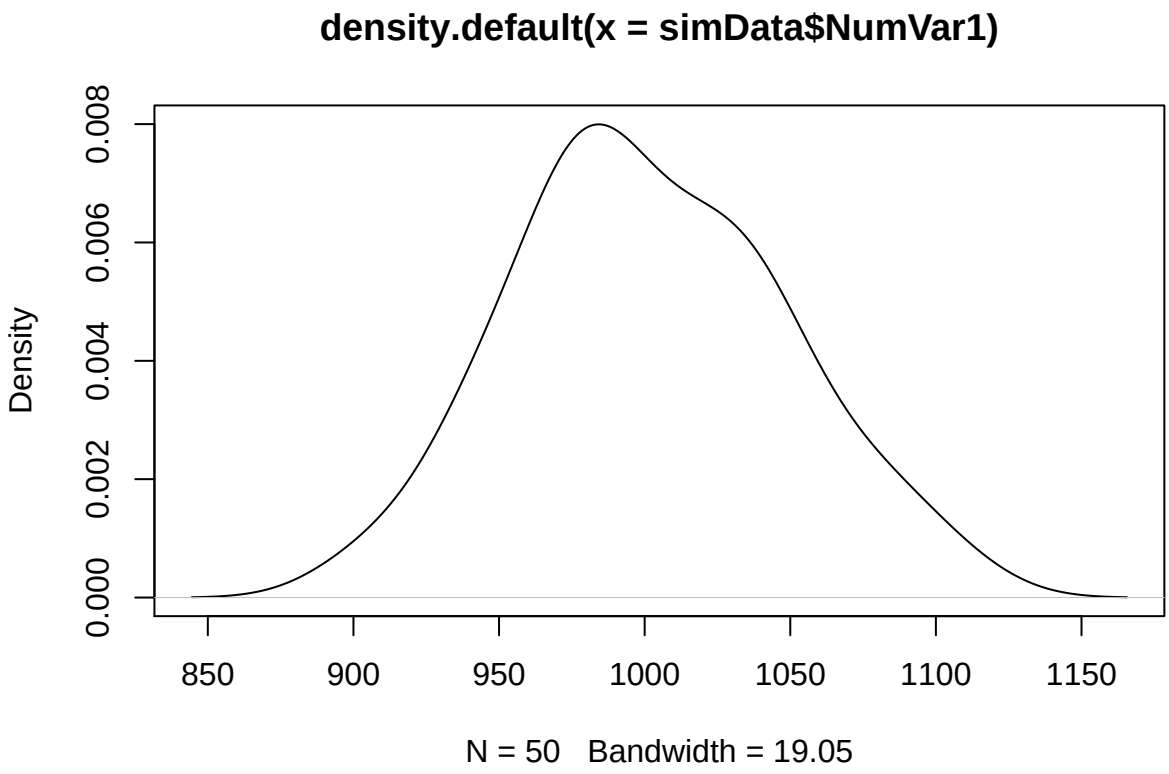
```
plot(simData$NumVar1,type="o") ## Index plot
```



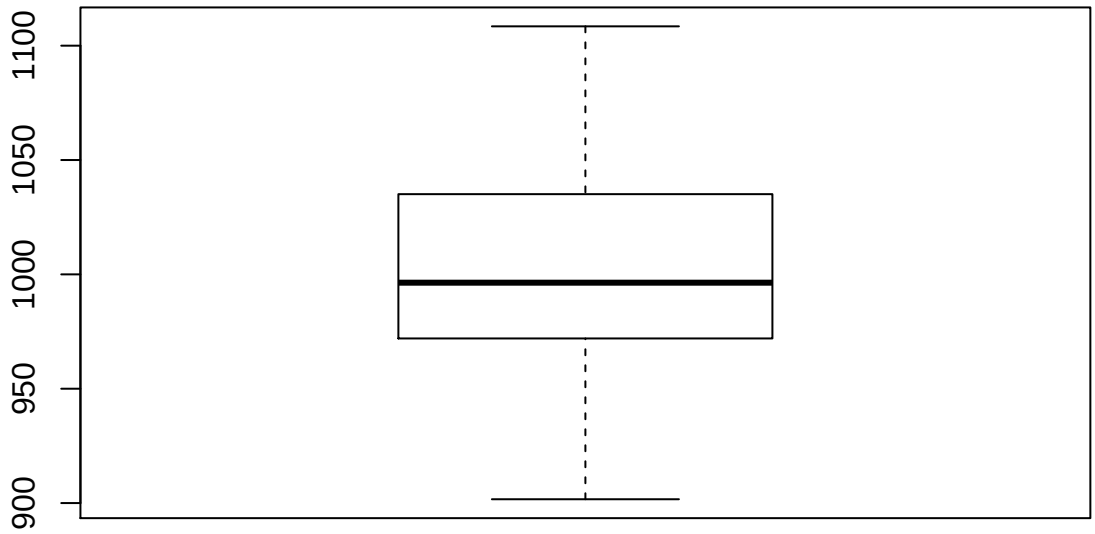

```
hist(simData$NumVar1) ## histogram
```




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




```
boxplot(simData$NumVar1) ## box plot
```



What it means:

"plot" represents the generic function for plotting of R objects.

Notation for "plot": plot(x,y,...) where "..." are the arguments to be passed to methods, such as what type of plot will be used. In this example, type="o" is an index plot.

"hist" represents the generic function for computing histograms.

Notation for "hist": hist(x,...)

"density" is a generic function for computing kernel density estimates.

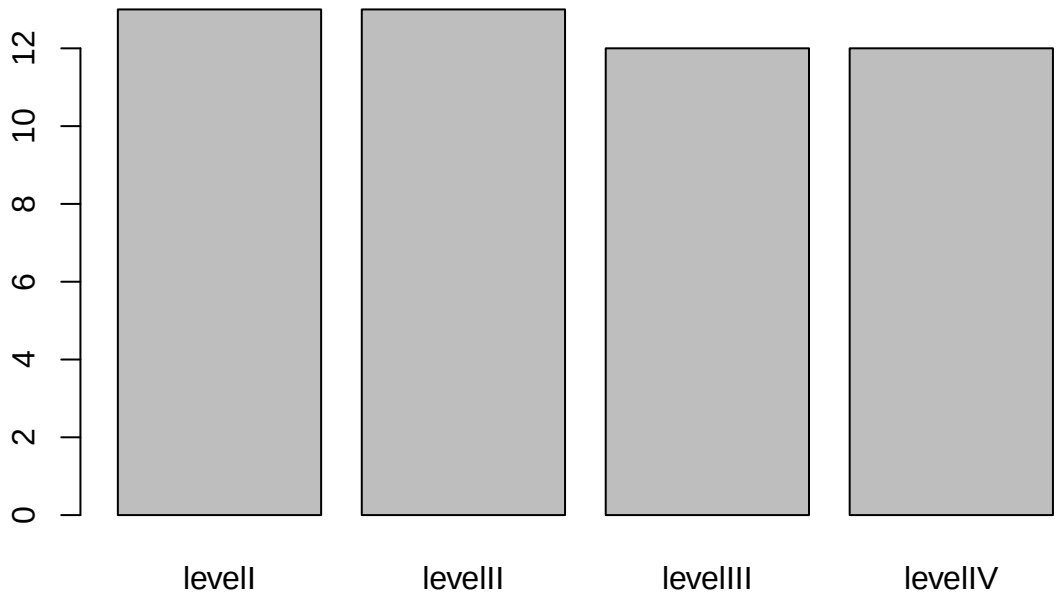
Notation for "density": density(x,...)

"boxplot" produces a box-and-whisker plot of the given data set.

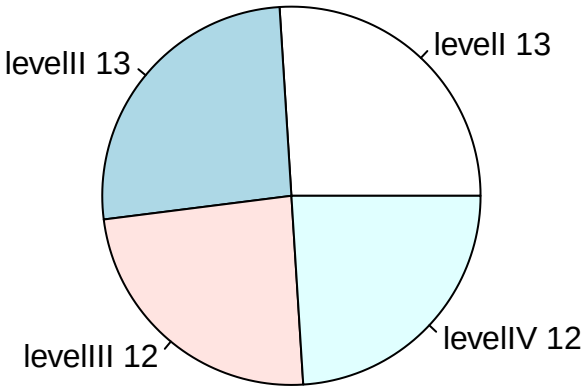
notation for "boxplot": boxplot(x,...)

One Variable: Factor Variable R-Markdown language:

```
plot(simData$FacVar3) ## bar plot
```




```
## 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
```



What it means:

"counts" represents a function to count the number of fields in the data set.

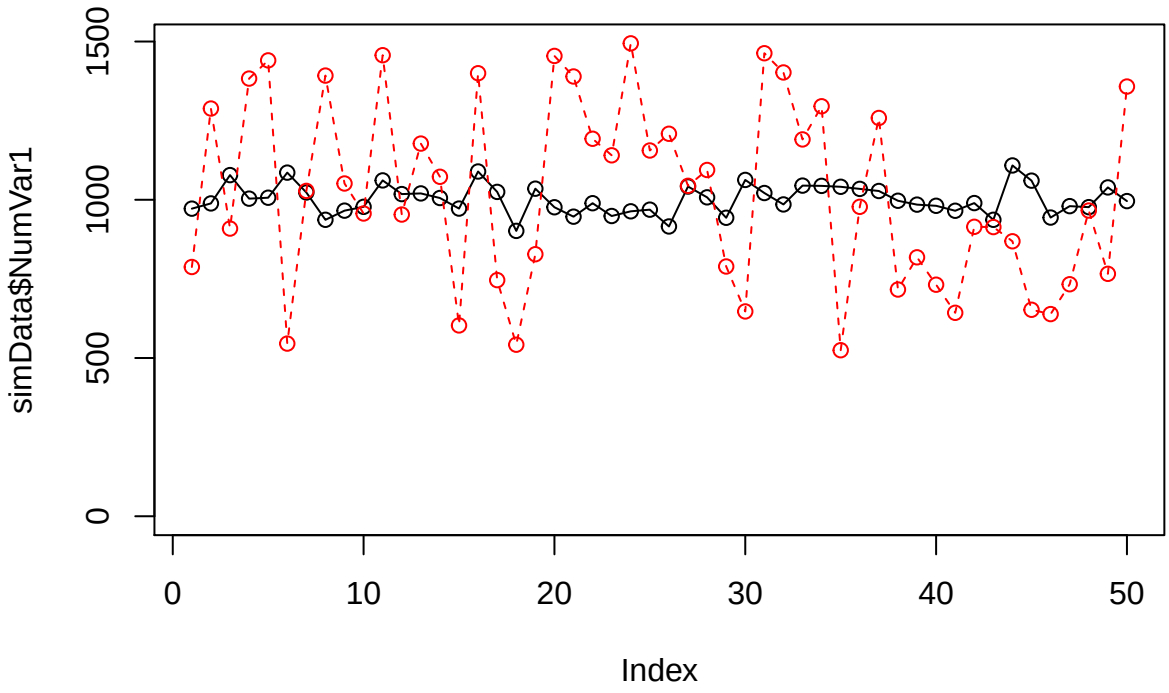
"labs" represents the variable for the labels of the data set.

"pie" is a function to draw a pie chart.

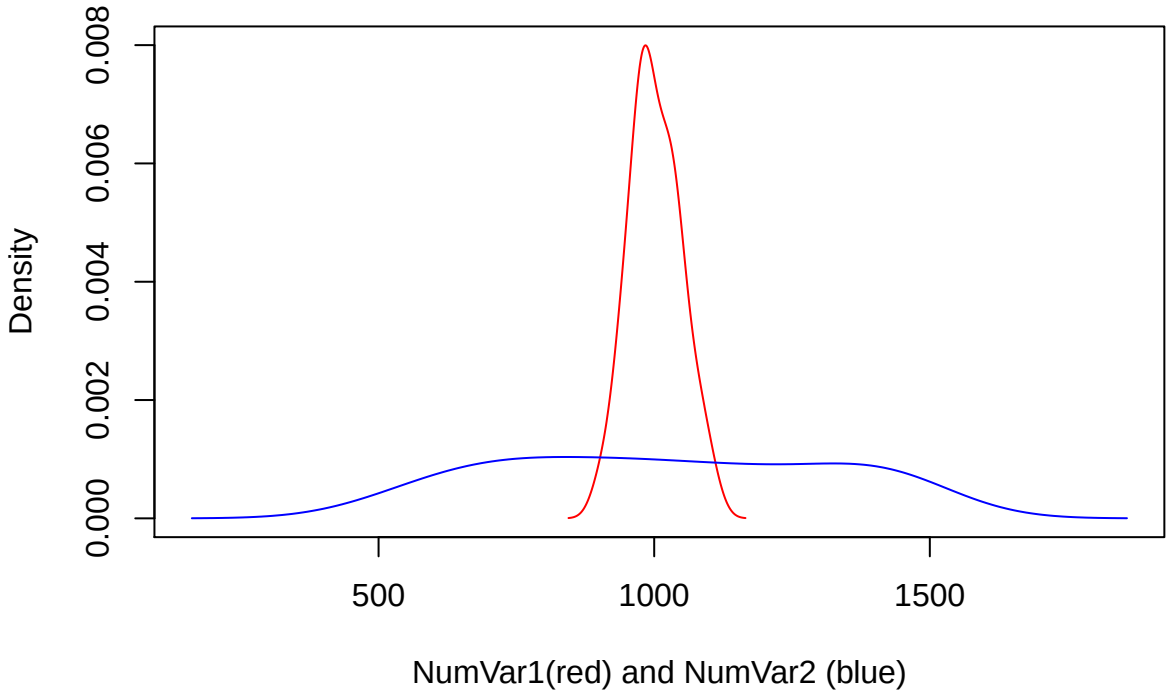
Notation for "pie": pie(x, labels,...) where "..." represents numerous other fields to choose from.

Two Variables: Two Numeric Variables R-Markdown language:

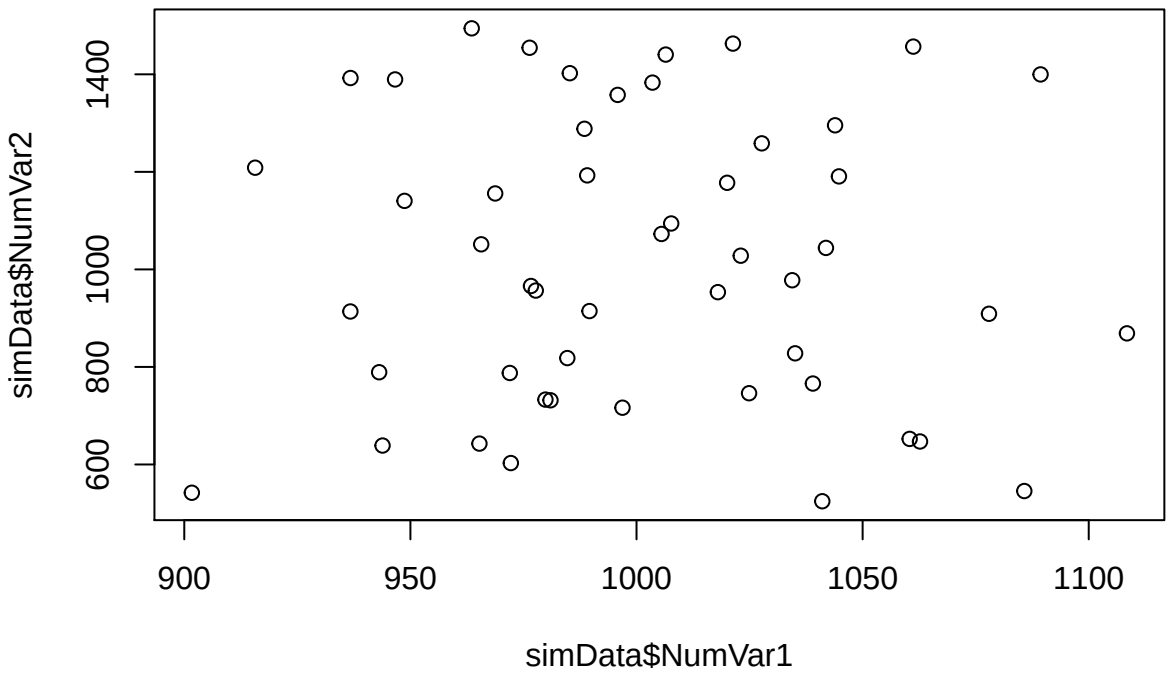
```
plot(simData$NumVar1,type="o",ylim=c(0,max(simData$NumVar1,simData$NumVar2)))## index plot with one vari
lines(simData$NumVar2,type="o",lty=2,col="red")## add another variable
```




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



What it means:

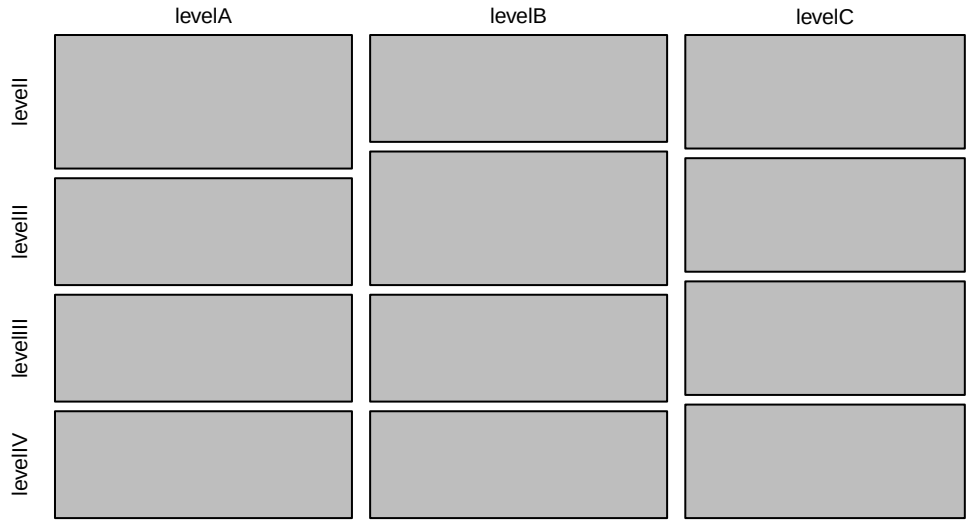
"ylim" represents a function for the world coordinates of a graphic window. In this example, the y-coordinate maximum is taken from the largest value between the two data sets, NumVar1 and NumVar2.

"lines" is a generic function that takes coordinates given in various ways and joining the corresponding points with line segments. This is one way to add another variable to a graph and to change the color of the line segment.

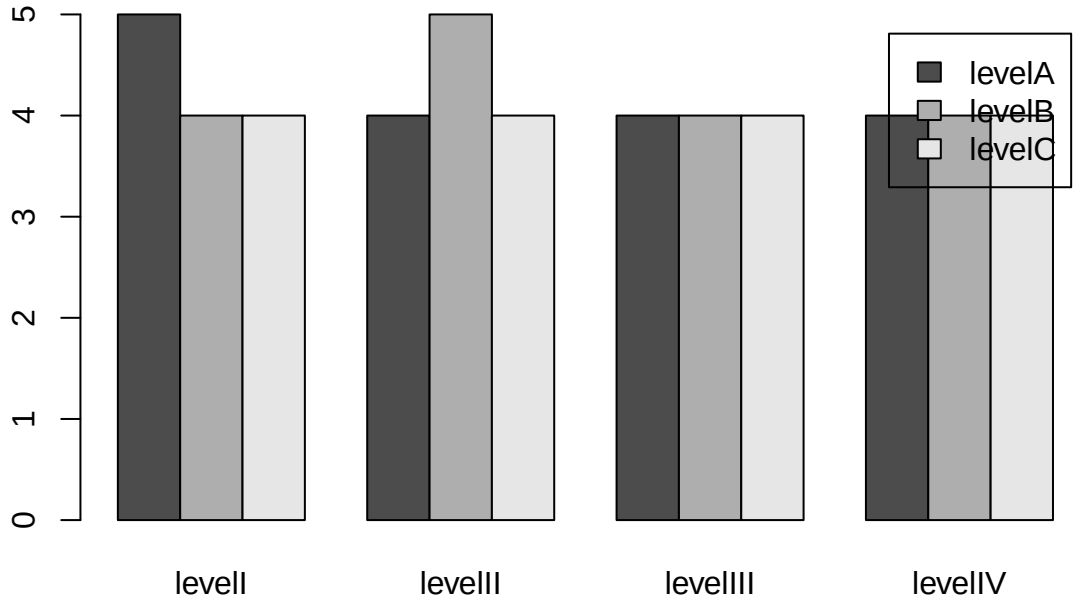
To create a scatter plot, simply use the "plot" function and input a variable for both the x and y notation sections.

Two Variables: Two Factor Variables R-Markdown language:

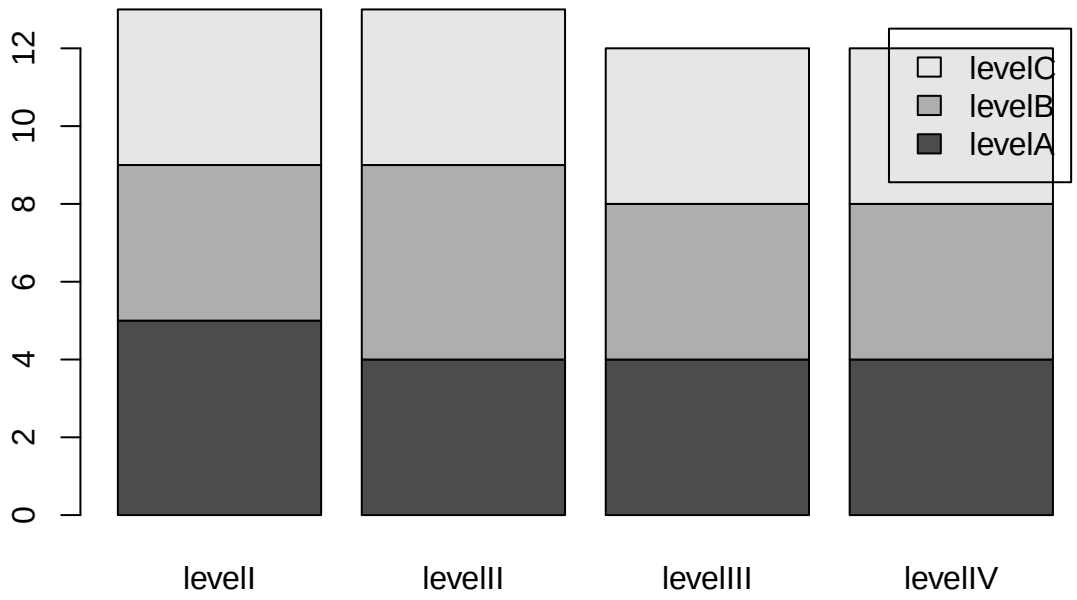
```
## Mosaic plot
plot(table(simData$FacVar2,simData$FacVar3))
```



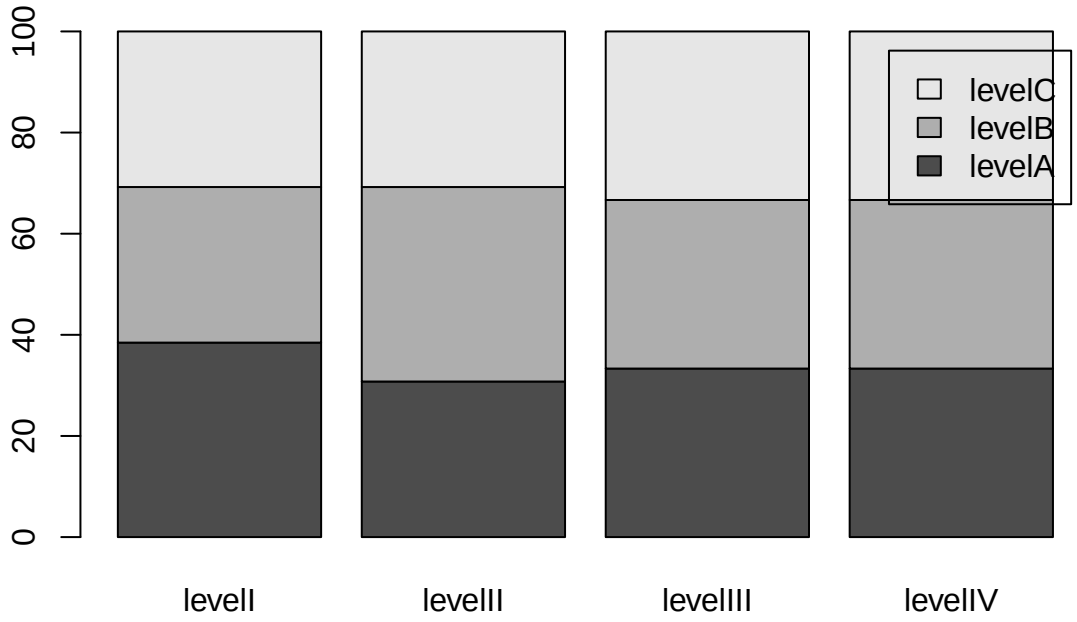

```
## barplots
bartable=table(simData$FacVar2,simData$FacVar3) ## get the cross tab
barplot(bartable,beside=TRUE, legend=levels(unique(simData$FacVar2))) ## plot
```




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




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



What it means:

To create a Mosaic plot, which is a graphic image of qualitative variables, use the "plot" function with the "table" function inside.

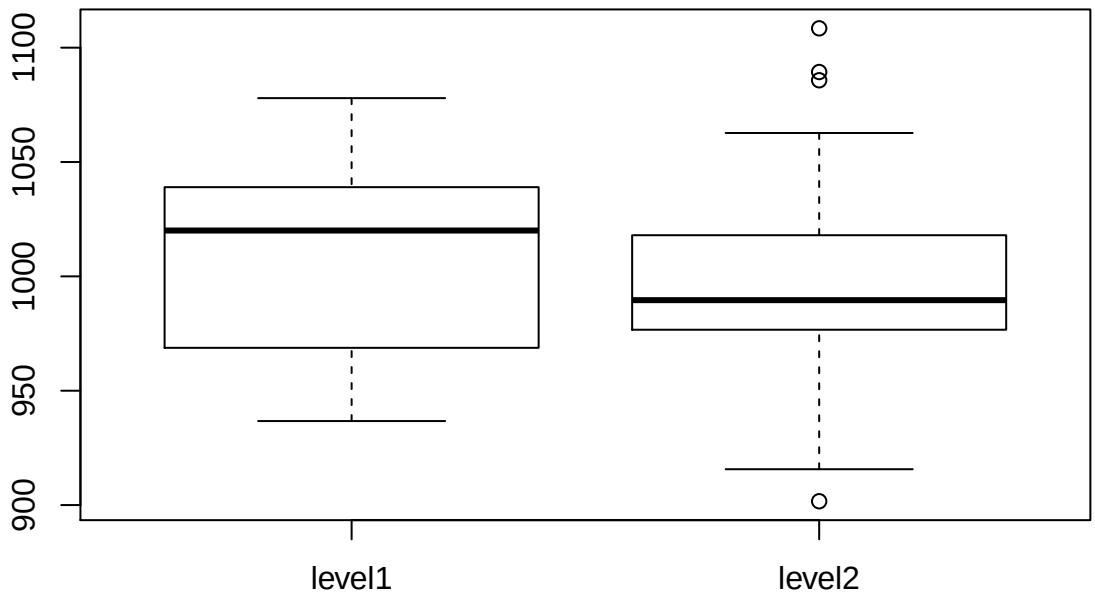
Barplots are created in a similar manner. Set a variable equal to the "table" function and use this variable inside the "barplot" function.

"barplot" creates a bar plot with vertical or horizontal bars.

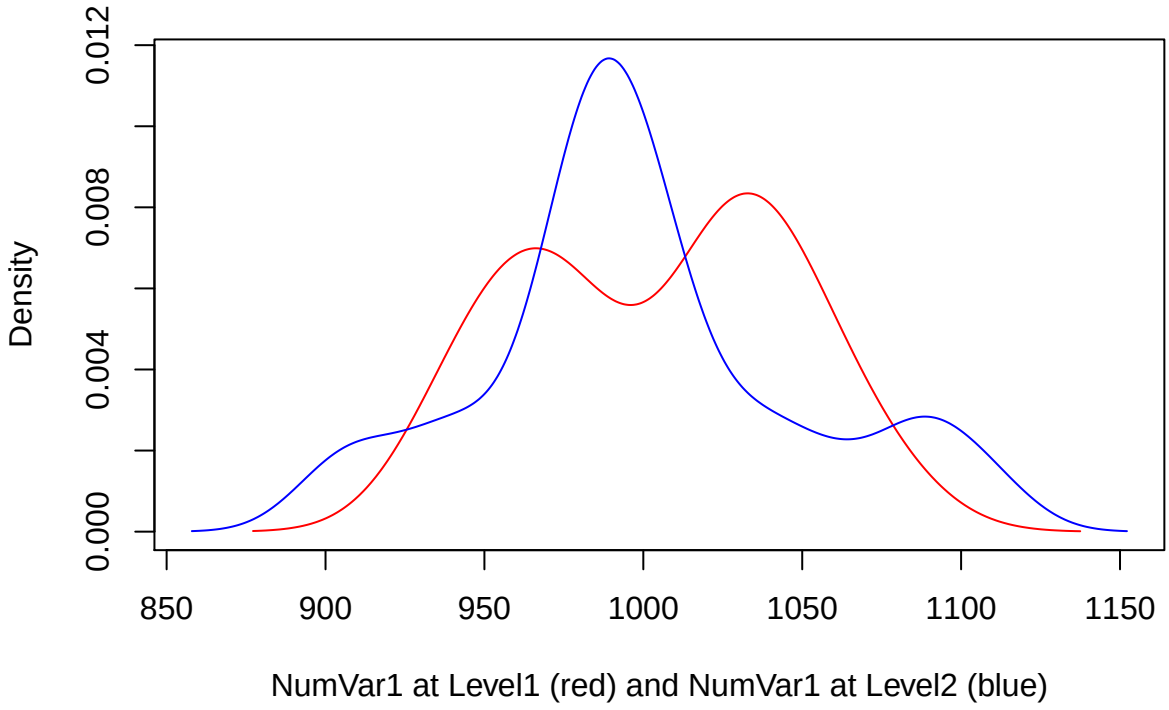
Notation for "barplot": barplot(height,...)

Two Variables: One Factor and One Numeric R-Markdown language:

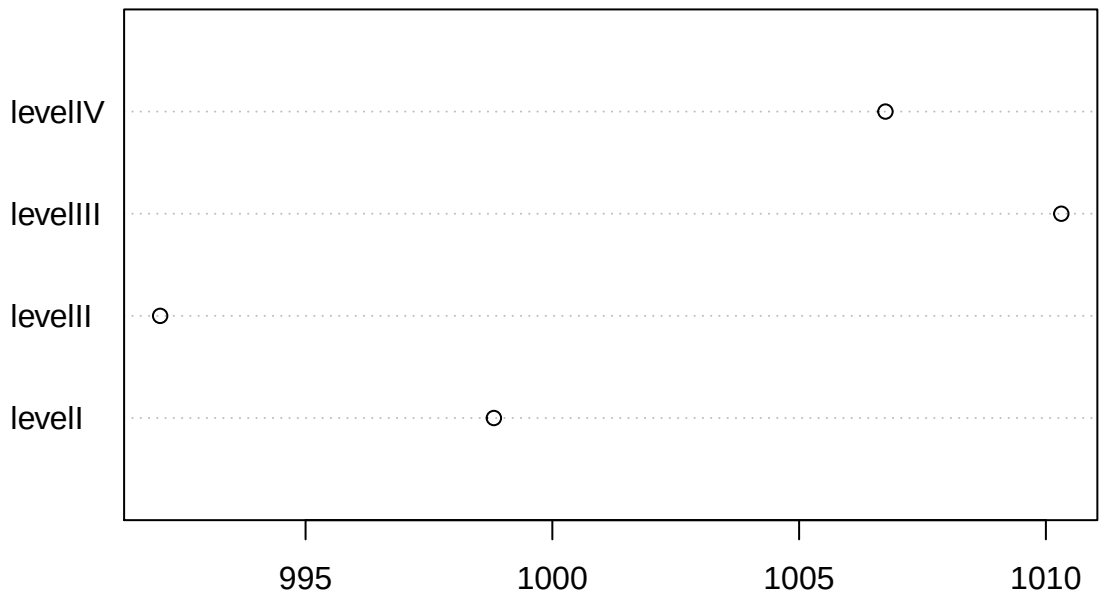
```
## Box plots for the numeric var over the levels of the factor var
plot(simData$FacVar1,simData$NumVar1)
```



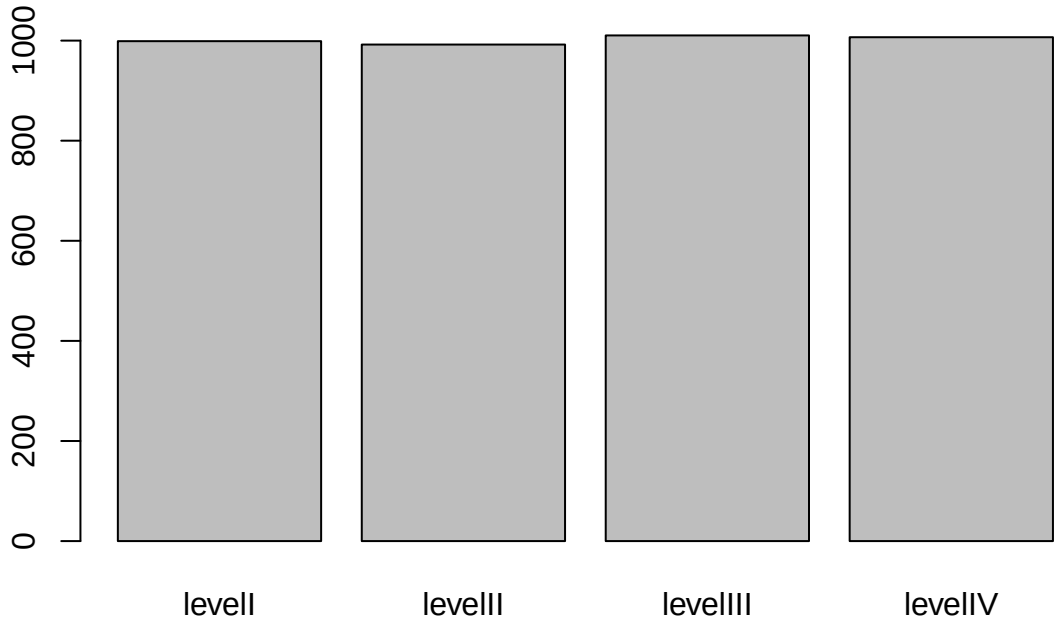

```
## density plot of numeric var across multiple levels of the factor var
level1=simData[simData$FacVar1=="level1",]
level2=simData[simData$FacVar1=="level2",]

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 a
lines(dv3, col = "red")
lines(dv4, col = "blue")
```




```
## 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
```




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




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

What it means:

"aggregate" splits the data into subsets, computes summary statistics for each, and returns the result in a convenient form.

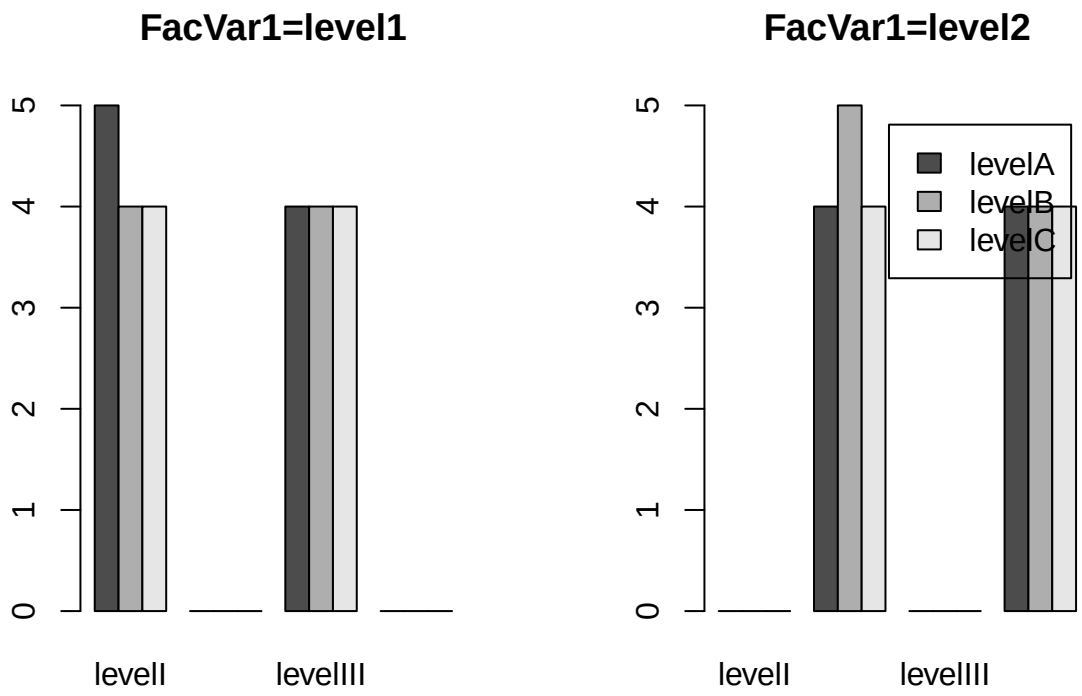
Notation for "aggregate": aggregate(x,...)

"dotchart" draws a Cleveland dot plot.

Notation for "dotchart": dotchart(x, labels,...)

Three Variables: Three Factor Variables R-Markdown language:

```
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)))
```



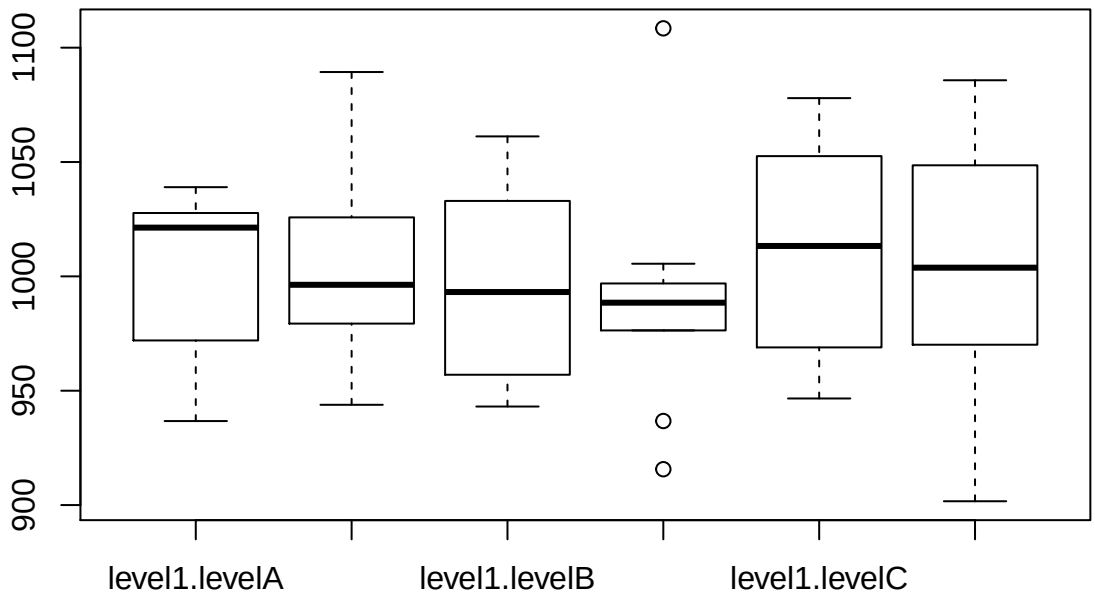
What it means:

"par" is used to set paramters by specifying them as arguments.

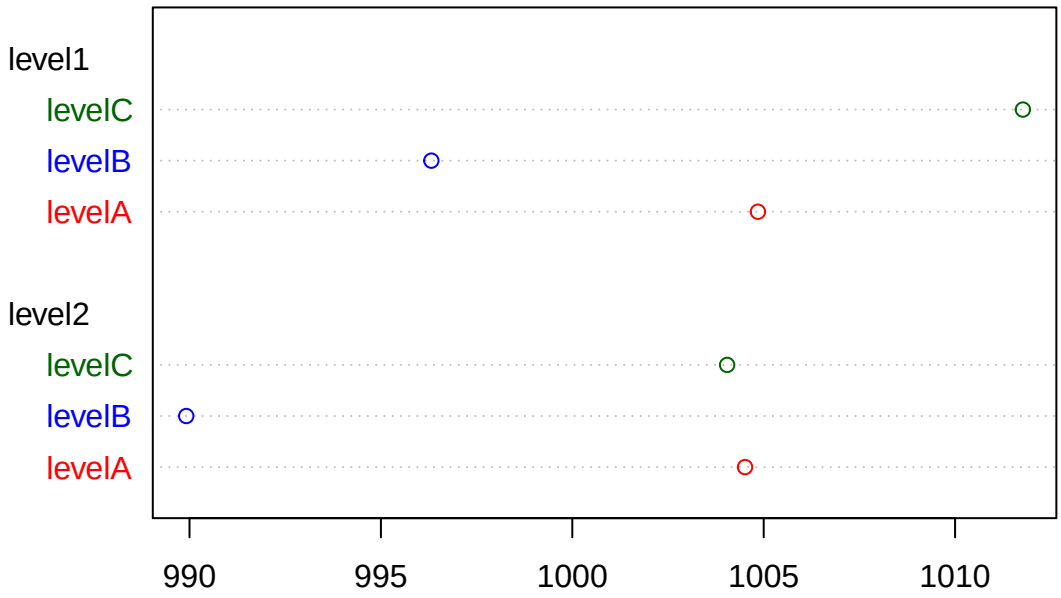
Notation of "par": par(. . . ,no.readonly = FALSE)

Three Variables: One Numeric and Two Factor Variables R-Markdown language:

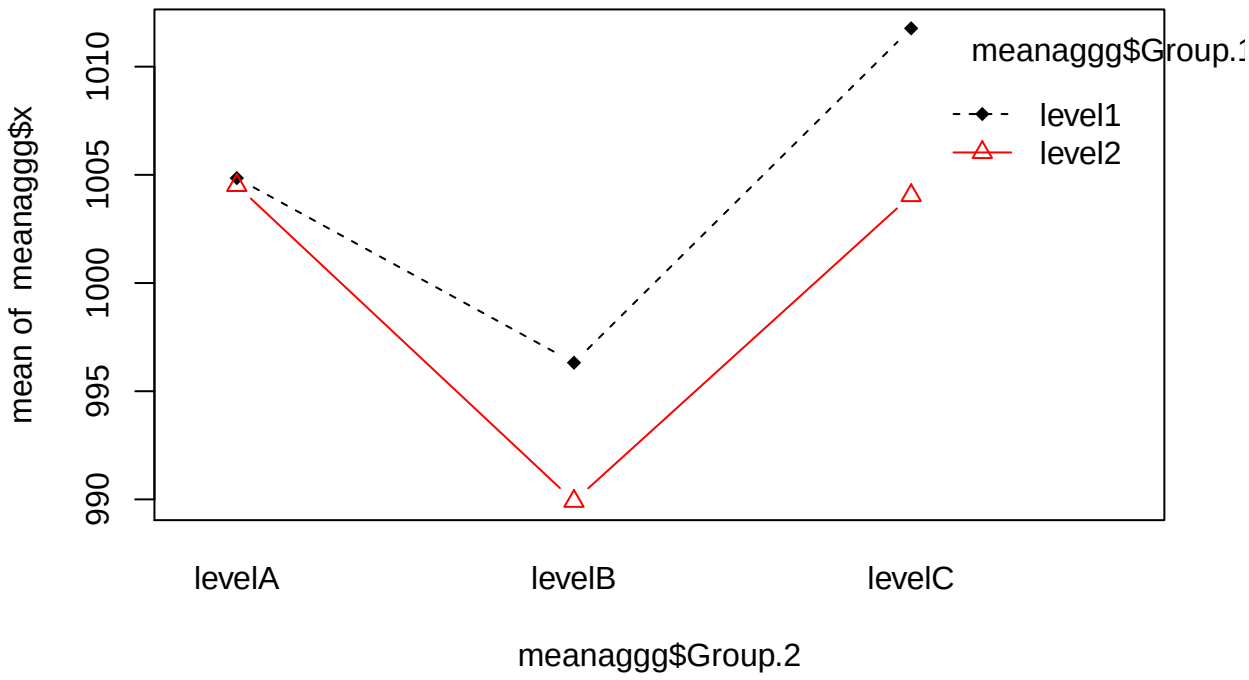
```
par(mfrow=c(1,1))
## boxplot of NumVar1 over an interaction of 6 levels of the combination of FacVar1 and FacVar2
boxplot(NumVar1~interaction(FacVar1,FacVar2),data=simData)
```



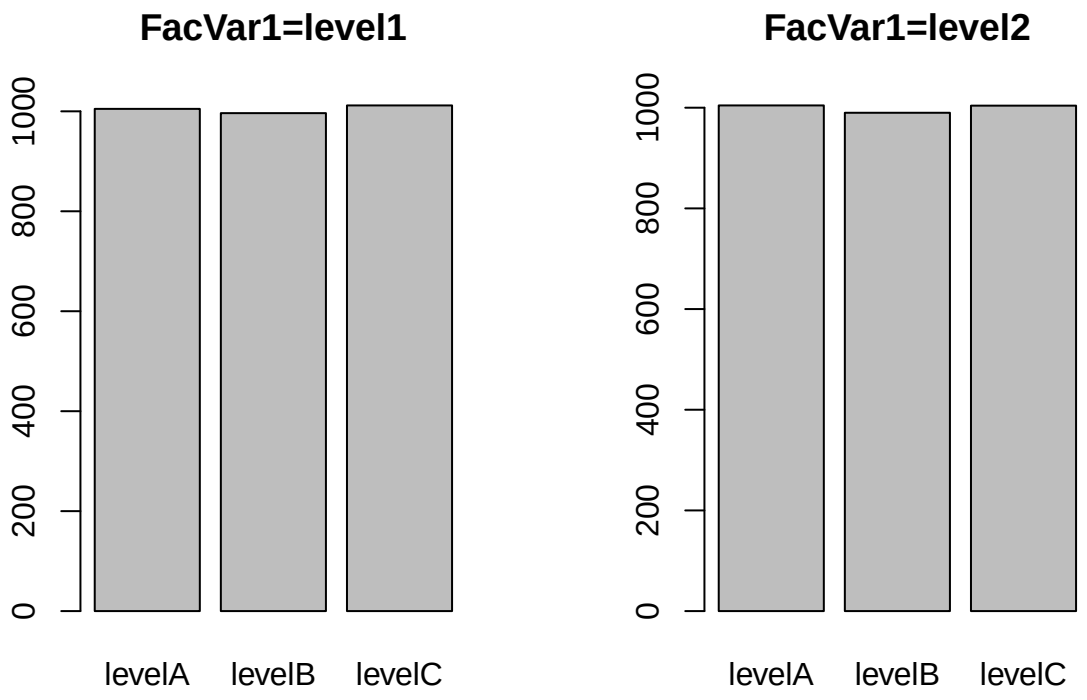

```
## Mean of 1 Numeric over levels of two factor vars
meanaggg=aggregate(simData$NumVar1, list(simData$FacVar1,simData$FacVar2), mean)
meanaggg=meanaggg[order(meanaggg$Group.1),]
meanaggg$color[meanaggg$Group.2=="levelA"] = "red"
meanaggg$color[meanaggg$Group.2=="levelB"] = "blue"
meanaggg$color[meanaggg$Group.2=="levelC"]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

dotchart(meanaggg$x,labels=meanaggg$Group.2, groups=meanaggg$Group.1,color=meanaggg$color) ## dotchart
```




```
interaction.plot(meanaggg$Group.2,meanaggg$Group.1,meanaggg$x,type="b", col=c(1:2),pch=c(18,24)) ## int
```



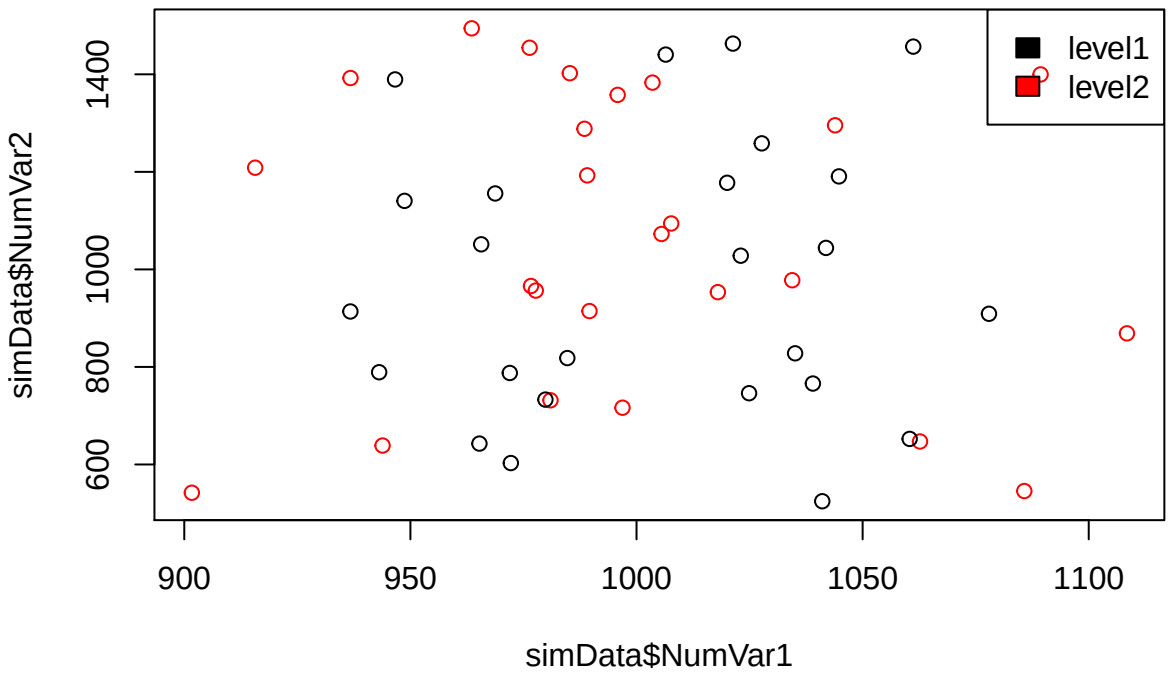

```
## some a bar plot
par(mfrow=c(1,2))

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")
```



What it means:

Three Variables: Two Numeric and One Factor Variables R-Markdown language:

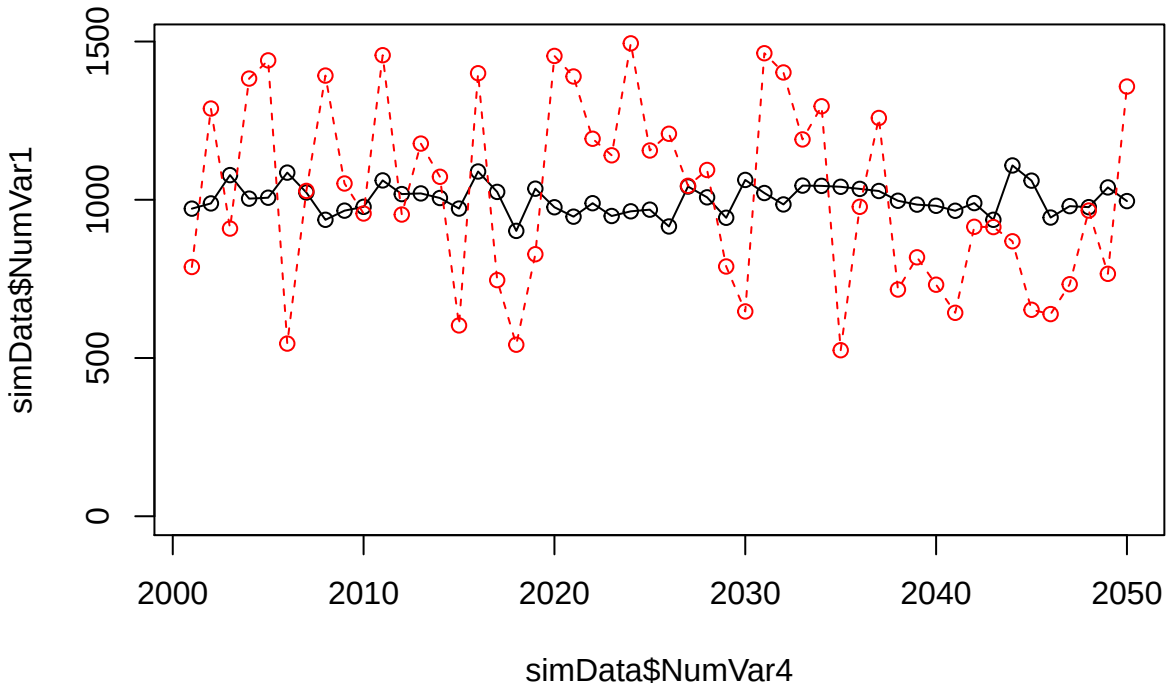
```
## Scatter plot with color identifying the factor variable
par(mfrow=c(1,1))
plot(simData$NumVar1,simData$NumVar2, col=simData$FacVar1)
legend("topright",levels(simData$FacVar1),fill=simData$FacVar1)
```



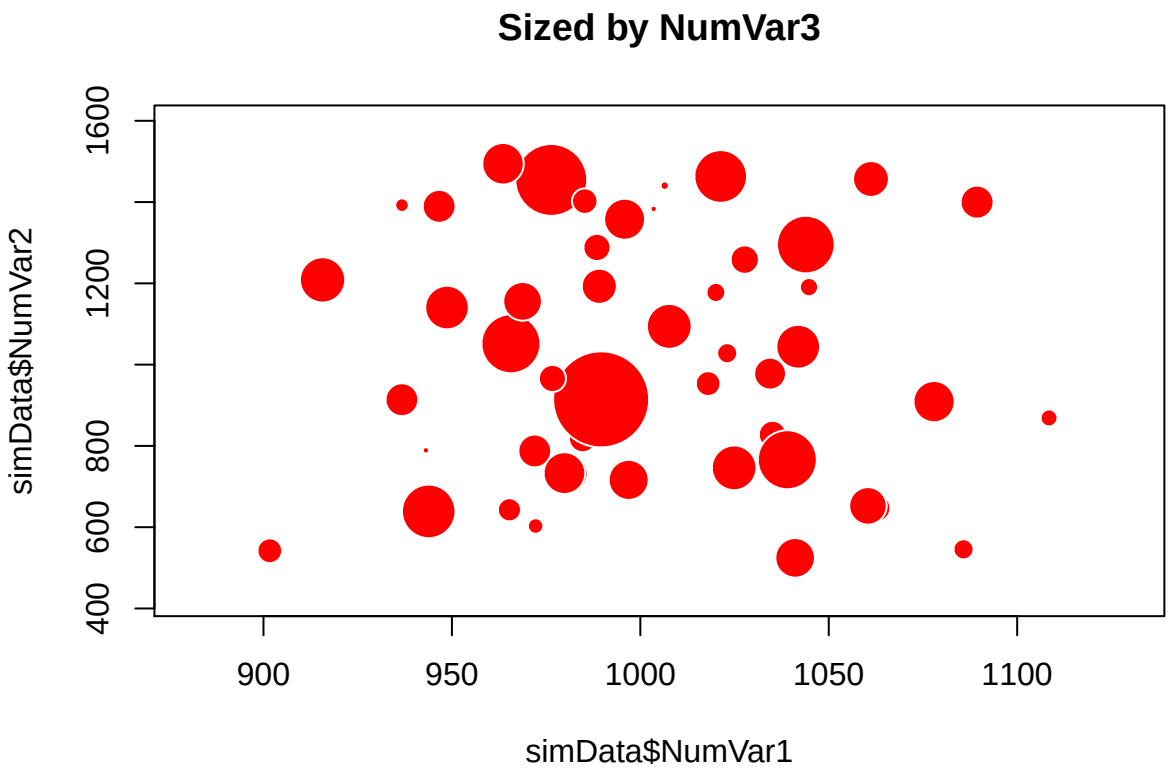
What it means:

Three Variables: Three Numeric Variables R-Markdown language:

```
## 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 do

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



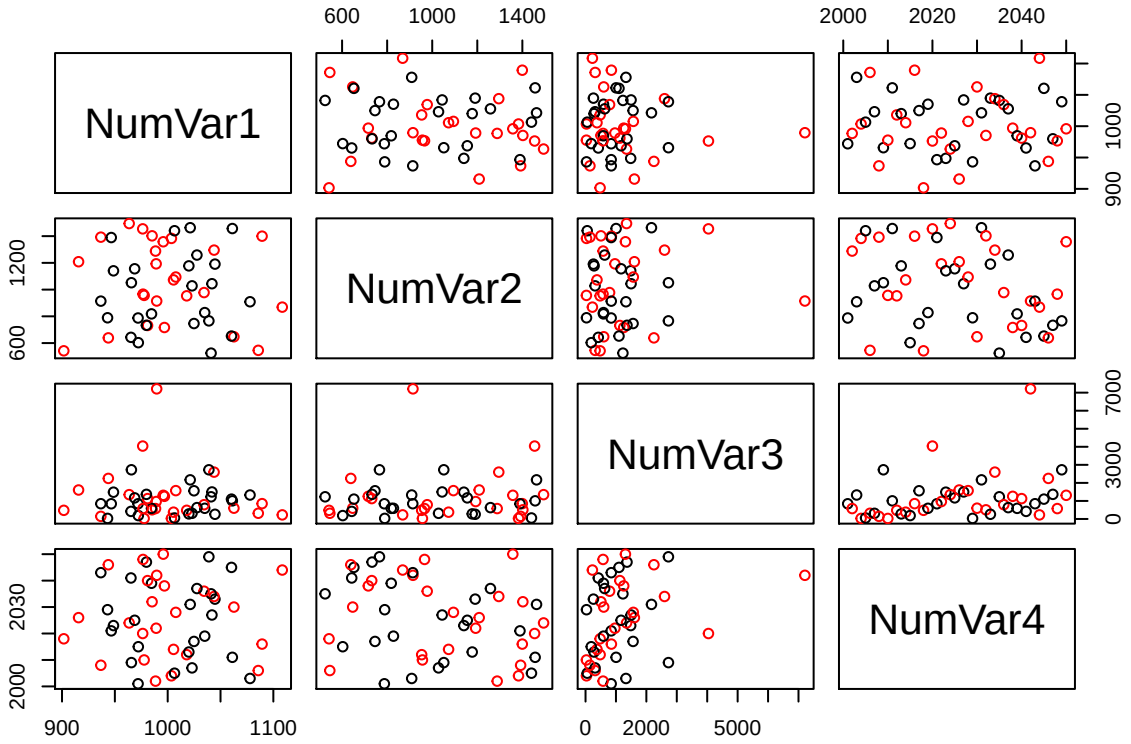

```
## Bubble plot - scatter plot of NumVar1 and NumVar2 with individual observations sized by NumVar3
# http://flowingdata.com/2010/11/23/how-to-make-bubble-charts/
radius <- sqrt( simData$NumVar3/ pi )
symbols(simData$NumVar1,simData$NumVar2,circles=radius, inches=.25,fg="white", bg="red", main="Sized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



What it means:

Scatterplot Matrix of all Numeric Vars, colored by a Factor variable R-Markdown language:

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



What it means:

References Besides the link from flowingdata.com referred to in the context of the bubble plot, additional websites were used as references. http://www.harding.edu/fmccown/r/ http://www.statmethods.net/