ggplot2 Replication

Aaron Grenz

Abstract This is a graphing manual for ggplot graphs.

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

R-Markdown language:

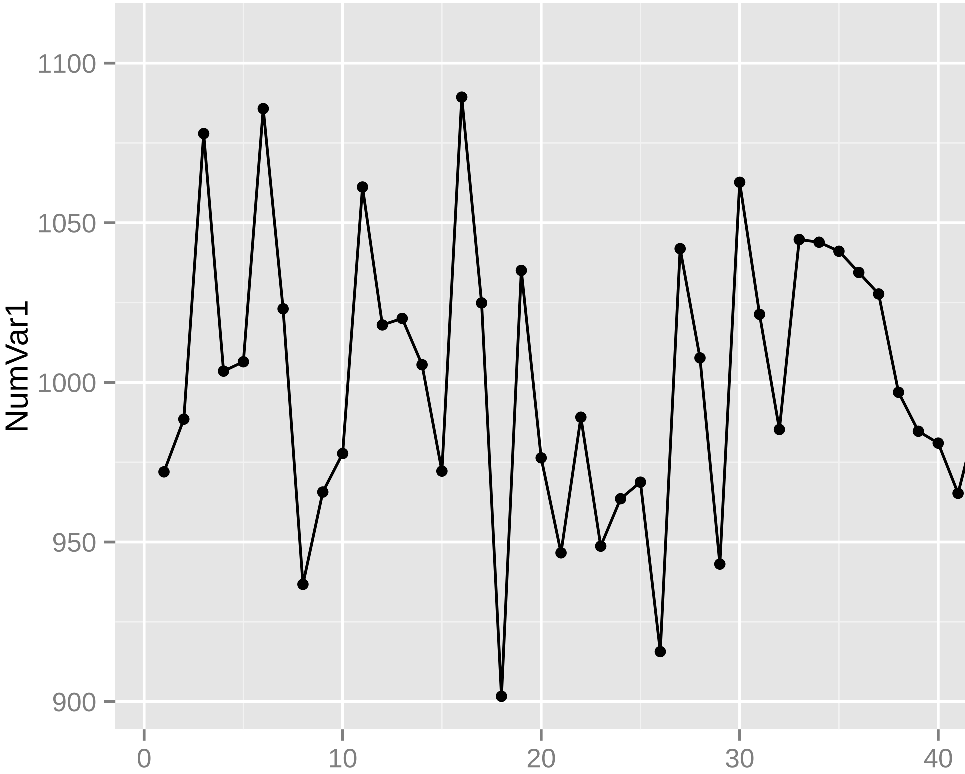
```r
## Simulate some data

## 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)
```

```r
library(ggplot2)
library(reshape2)
```

Initialize the libraries used for this page. What it means:

The libraries are add-on packages that must be “called” in order to be used.

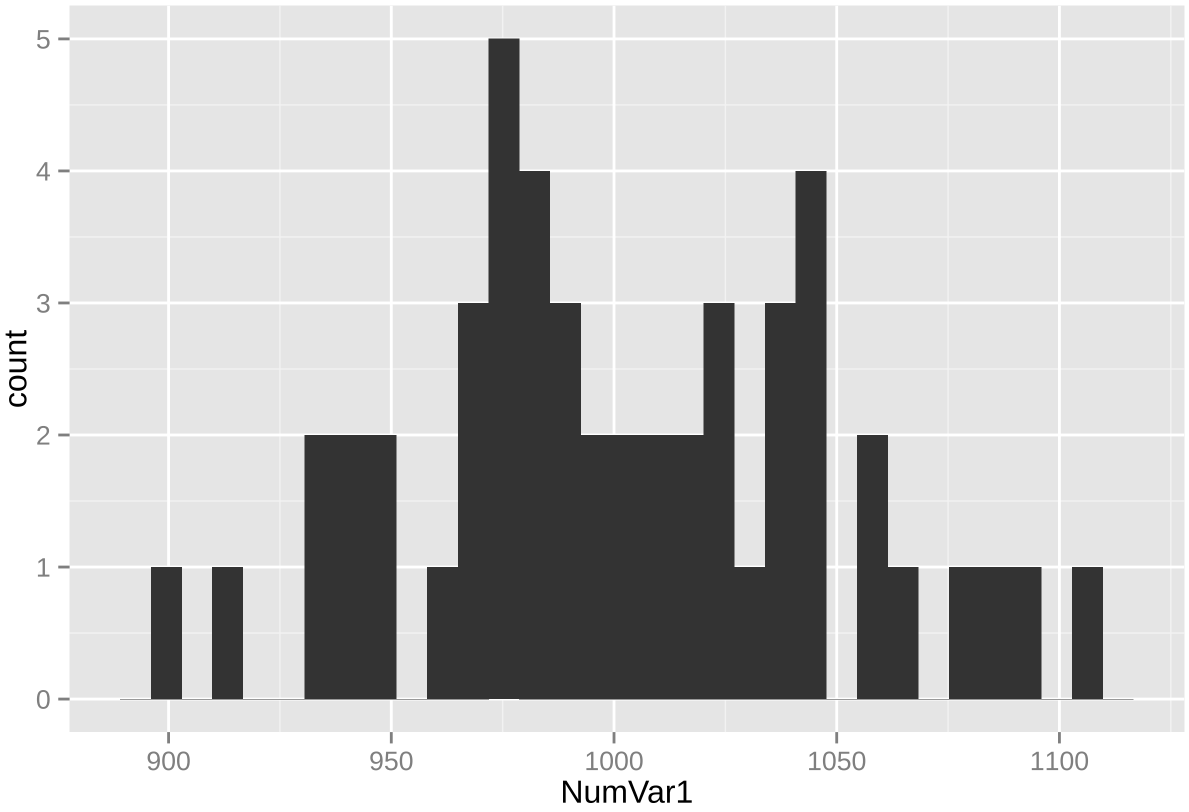
```r
ggplot(simData,aes(y=NumVar1,x=1:nrow(simData),group="NumVar1"))+geom_point()+geom_line()+ xlab("") ## I
```



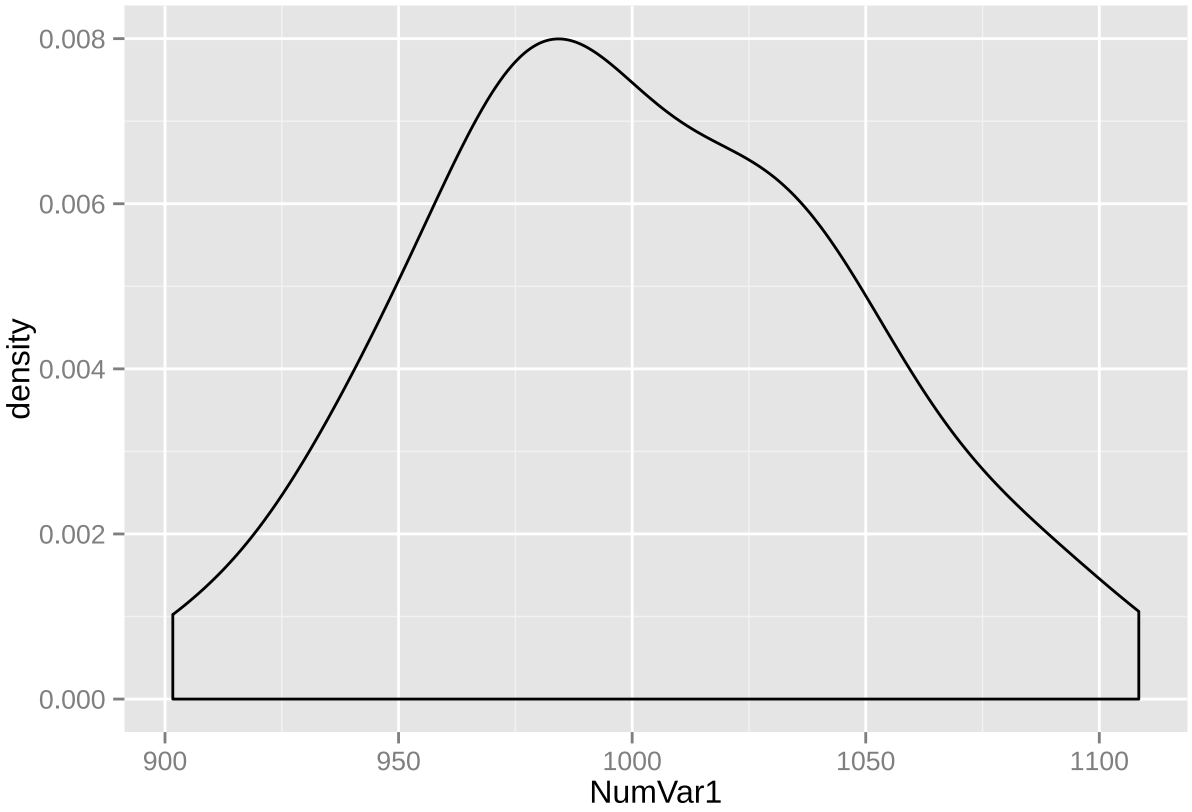
One Variable: Numeric Variable

```
ggplot(simData,aes(x=NumVar1))+geom_histogram() ## histogram
```

```
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
```



```
ggplot(simData,aes(x=NumVar1))+geom_density() ## Kernel density plot
```



What it means:

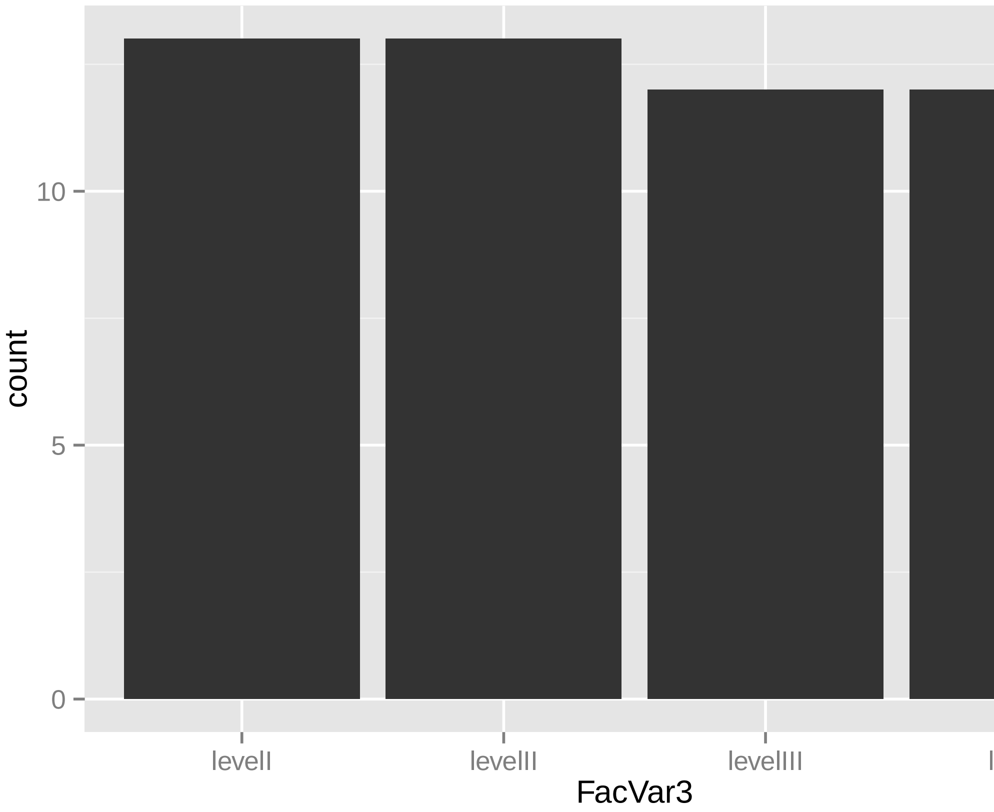
ggplot's standard notation is this: ggplot(data table, aesthetic generators)+ type of graph For example, "simData" is the data table including the three factor variables and four numeric variables.

"aes" represents the aesthetic generators of the graph. These can include characteristics such as line thickness, line color, data point shape, data point size, fill color, labels, and so on. This portion of the code controls what the look of the graph.

The"aes" section also holds the values of both the x and y variables (for relavent graphs). Depending on the type of graph you specify, these variables will need to be defined in this notation group.

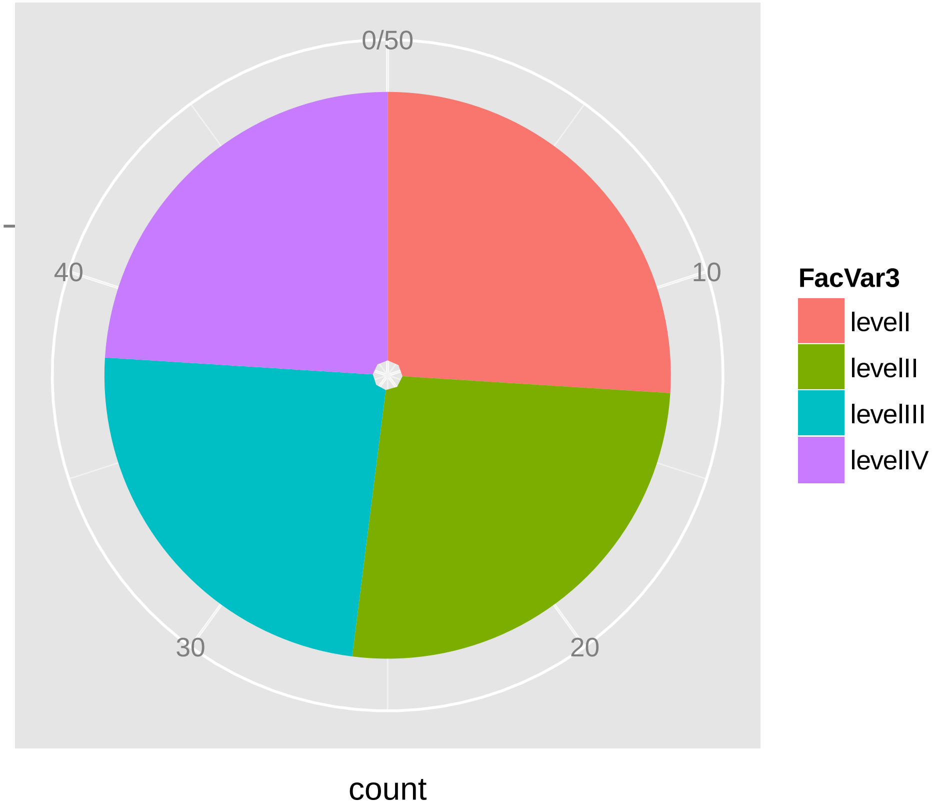
The last portion of the notation includes a plus symbol and the text "geom_". This is the portion of the code that establishes what type of graph will be produced.

```
## barplot
ggplot(simData,aes(x=FacVar3))+geom_bar()
```

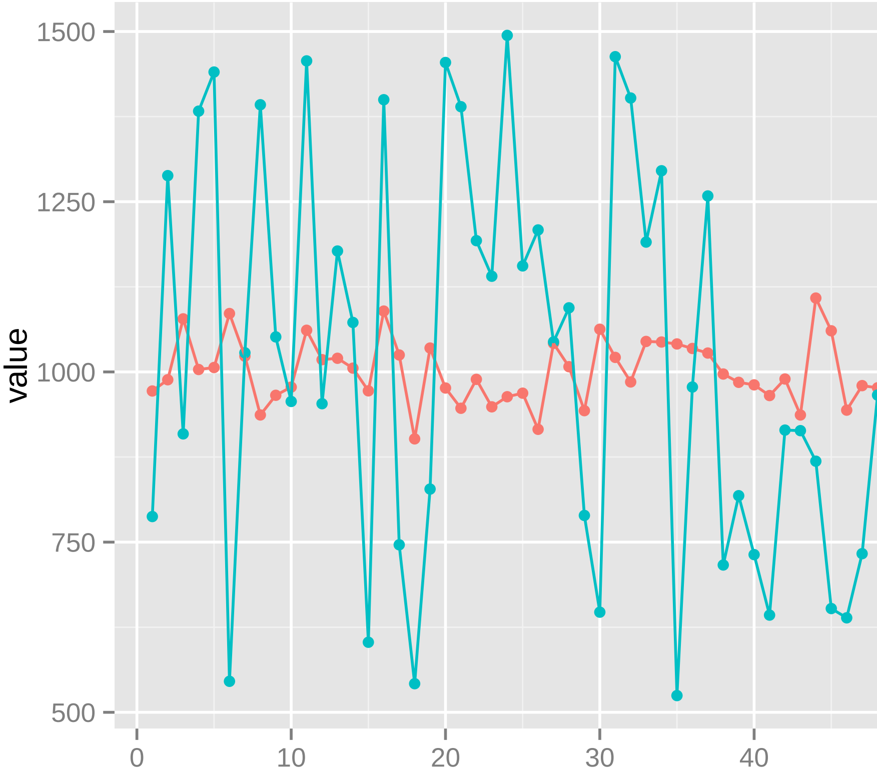


One Variable: Factor Variable

```
## pie chart - Not the best graph --- use with caution
ggplot(simData,aes(x = factor(""), fill=FacVar3, label=FacVar3))+geom_bar()+ coord_polar(theta = "y")  +
```

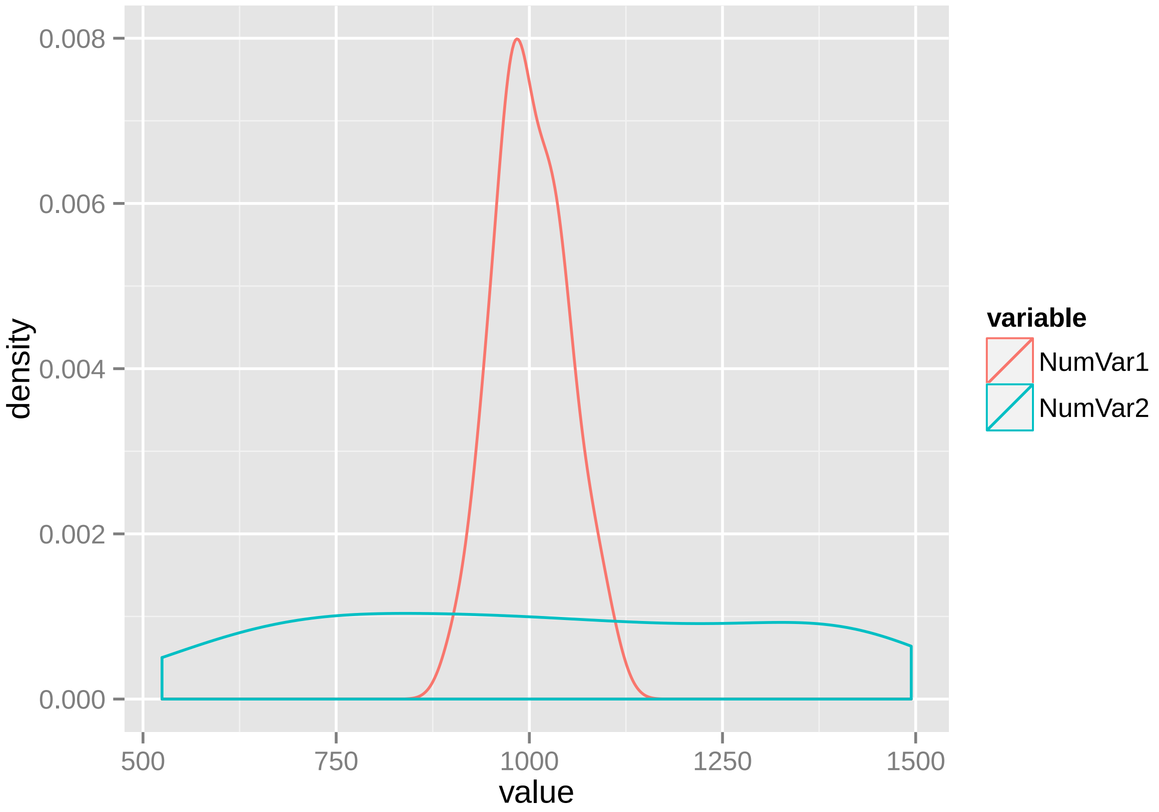


```
simtmp=simData[,c(4:5)] ## 4th and 5th columns are NumVar1 and NumVar2
simtmp$index=1:nrow(simtmp)
simtmpmelt=melt(simtmp,id=c("index"))

## line plots with observation number as index
ggplot(simtmpmelt,aes(y=value,x=index,color=variable))+geom_point()+geom_line()+xlab("")
```

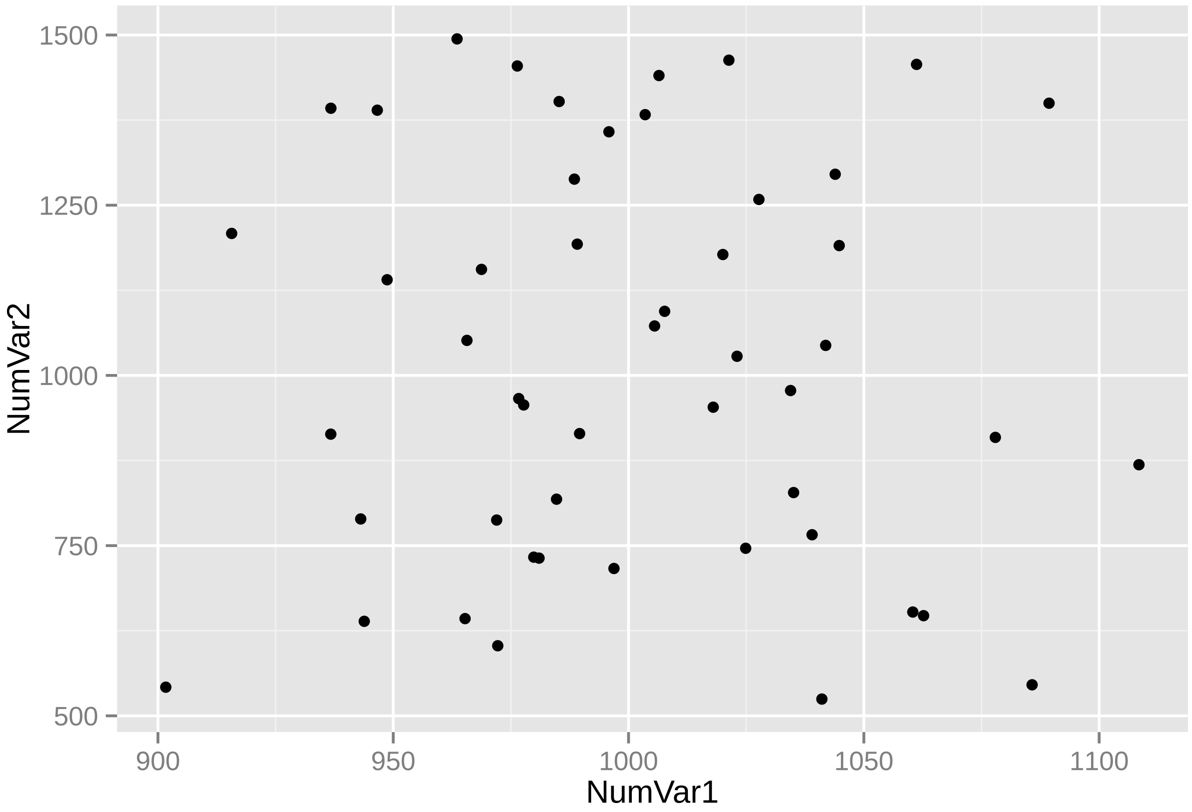


Two Variables: Two Numeric Variables

```
## Let's draw density functions for NumVar1 & NumVar2
ggplot(simtmpmelt,aes(x=value,color=variable))+geom_density()
```



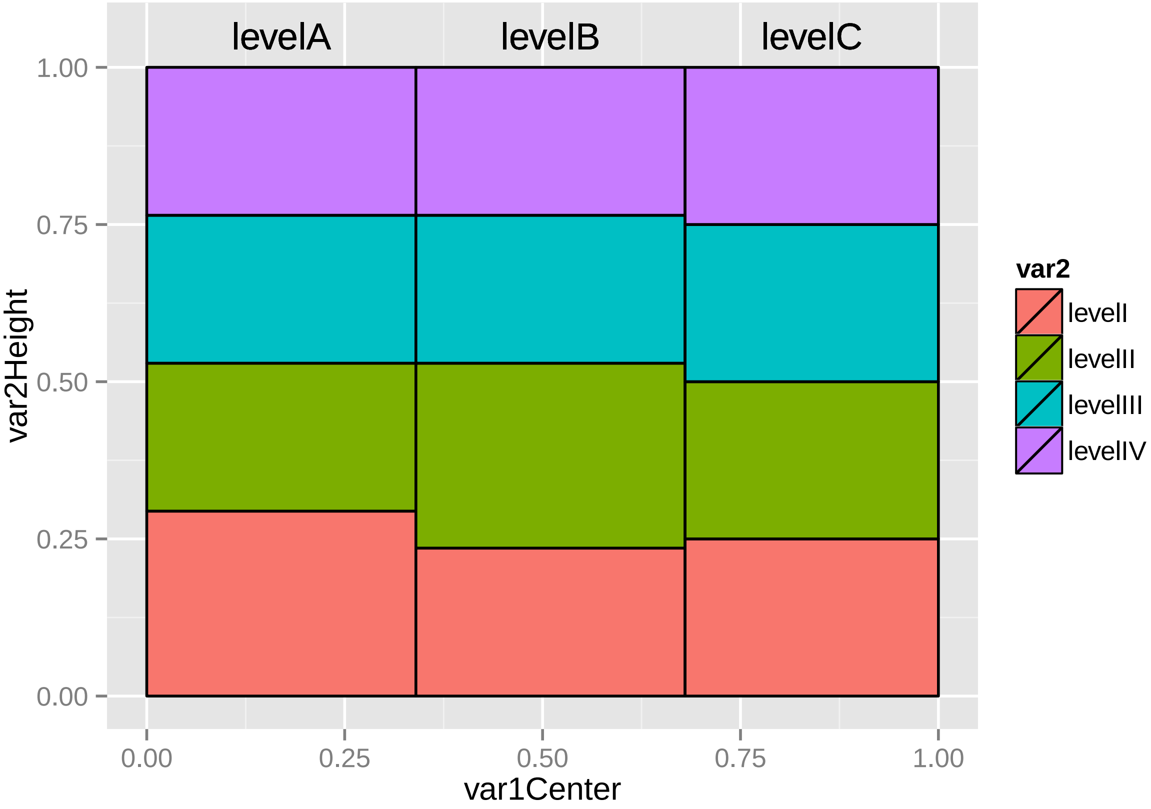
```
## scatter plot
ggplot(simData,aes(x=NumVar1,y=NumVar2))+geom_point()
```



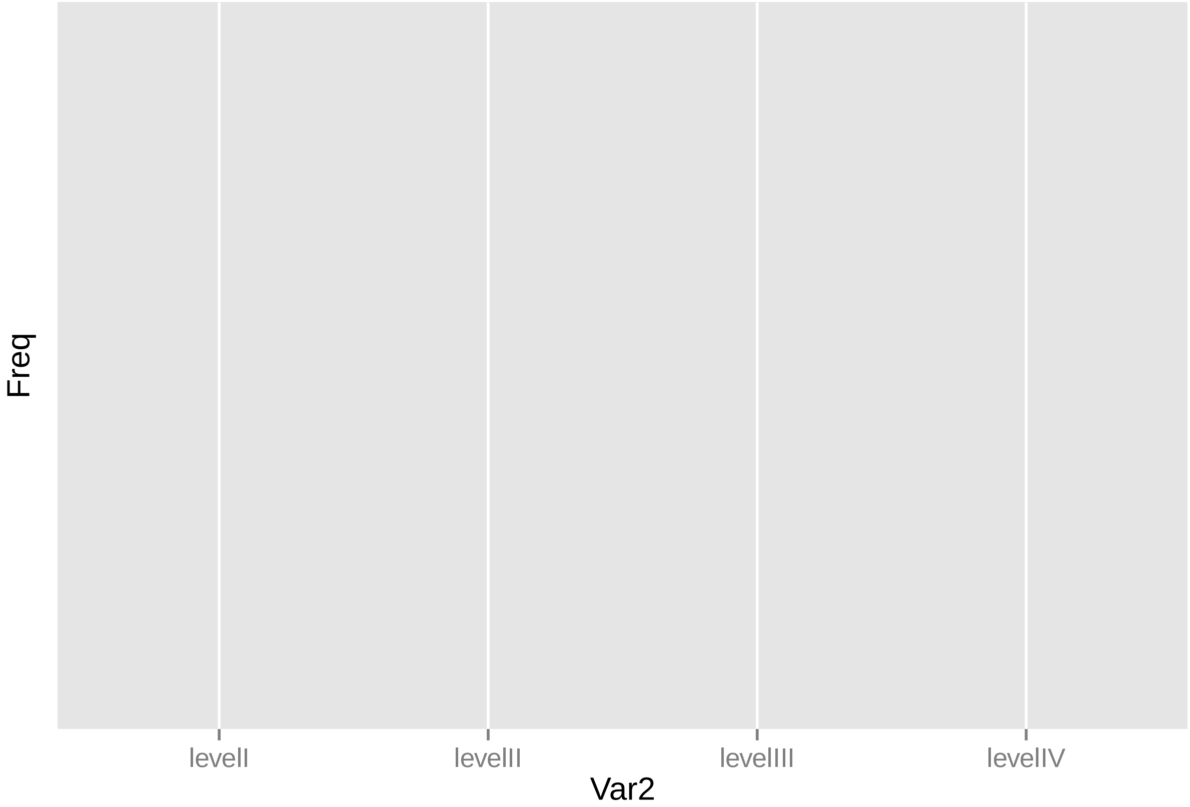
```
## Mosaic plot: ggMMplot function
ggMMplot <- function(var1, var2){
  require(ggplot2)
  levVar1 <- length(levels(var1))
  levVar2 <- length(levels(var2))

  jointTable <- prop.table(table(var1, var2))
  plotData <- as.data.frame(jointTable)
  plotData$marginVar1 <- prop.table(table(var1))
  plotData$var2Height <- plotData$Freq / plotData$marginVar1
  plotData$var1Center <- c(0, cumsum(plotData$marginVar1)[1:levVar1 -1]) +
    plotData$marginVar1 / 2

  ggplot(plotData, aes(var1Center, var2Height)) +
    geom_bar(stat = "identity", aes(width = marginVar1, fill = var2), col = "Black") +
    geom_text(aes(label = as.character(var1), x = var1Center, y = 1.05))
}
ggMMplot(simData$FacVar2, simData$FacVar3)
```

Two Variables: Two Factor Variables

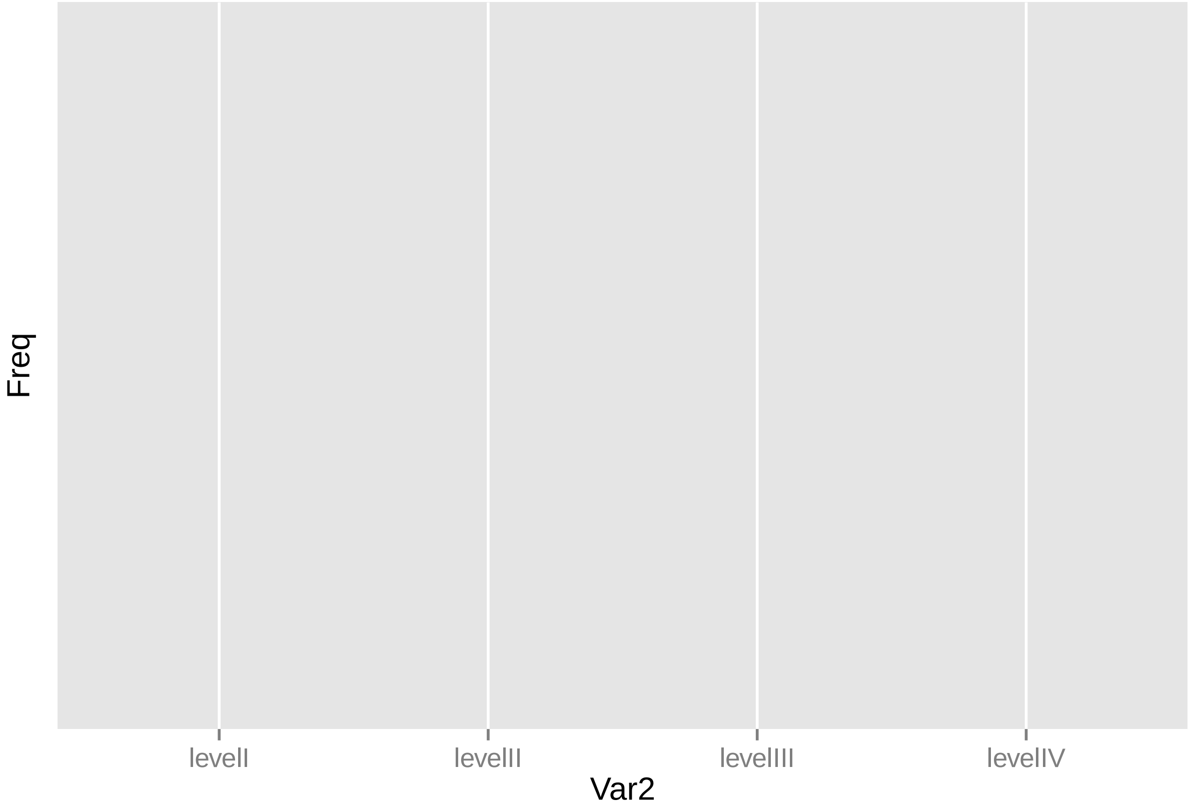
```
## Warning: position_stack requires constant width: output may be incorrect
```



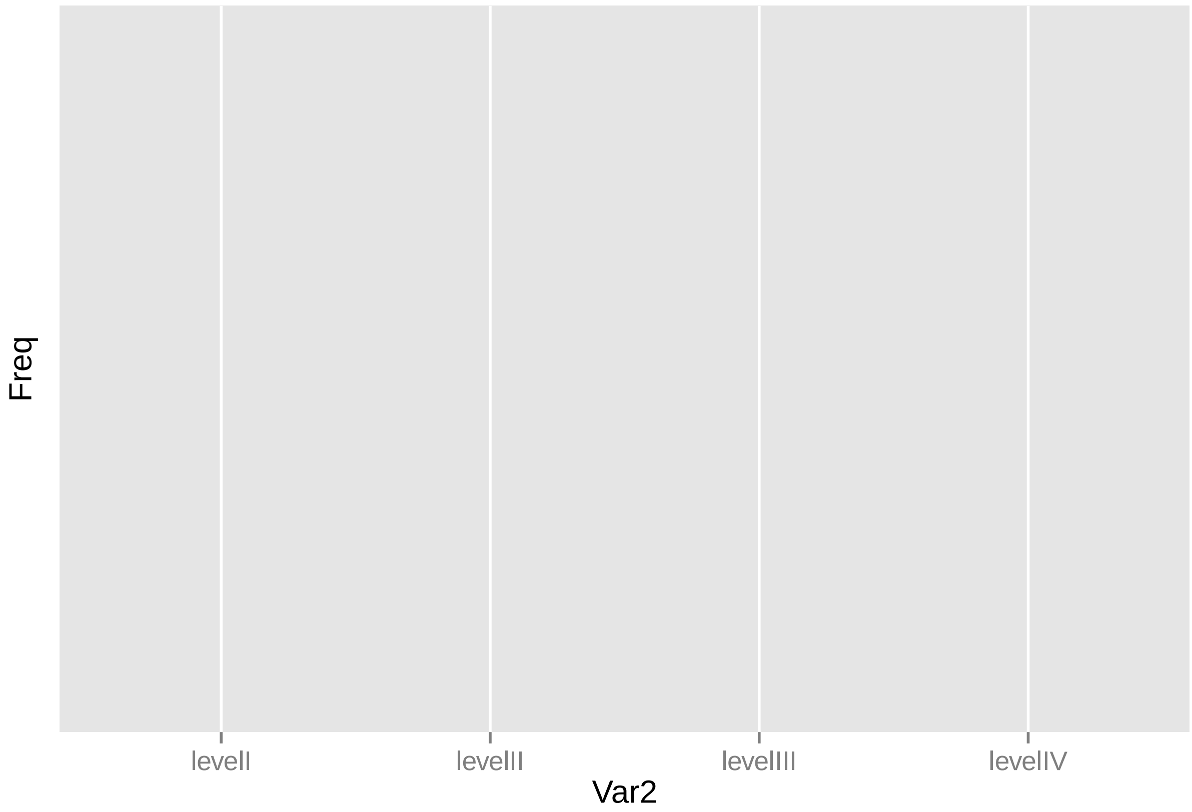
```
## barplots
bartabledat = as.data.frame(table(simData$FacVar2, simData$FacVar3)) ## get the cross tab
ggplot(bartabledat,aes(x=Var2,y=Freq,fill=Var1))+geom_bar(position="dodge") ## plot
```



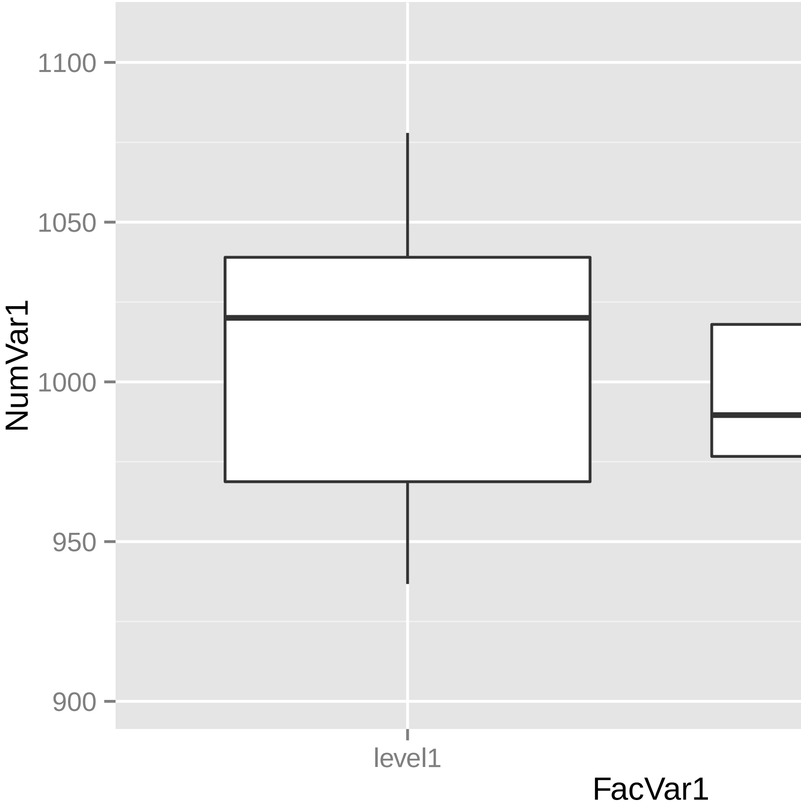
```
ggplot(bartabledat,aes(x=Var2,y=Freq,fill=Var1))+geom_bar() ## stacked
```



```
bartableprop =as.data.frame(prop.table(table(simData$FacVar2, simData$FacVar3),2)*100)
ggplot(bartableprop,aes(x=Var2,y=Freq,fill=Var1))+geom_bar() ## Stacked 100%
```

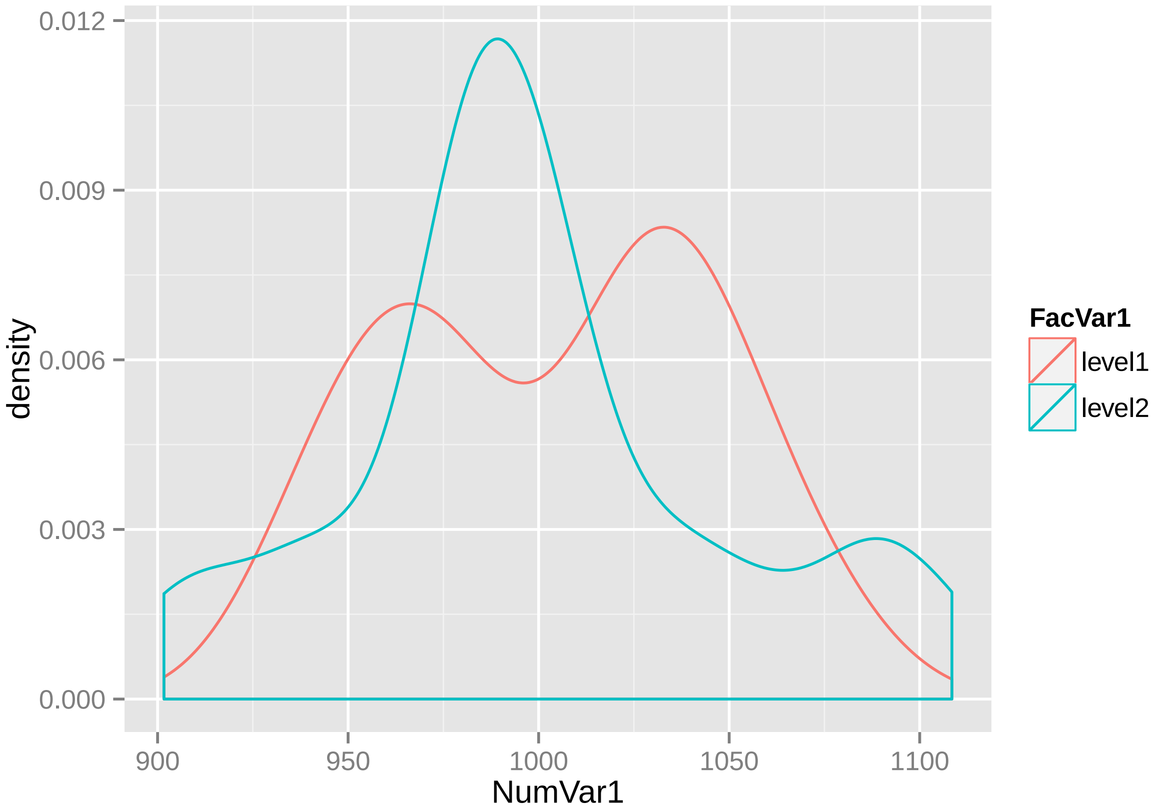


```
## Box plots for the numeric var over the levels of the factor var
ggplot(simData,aes(x=FacVar1,y=NumVar1))+geom_boxplot()
```

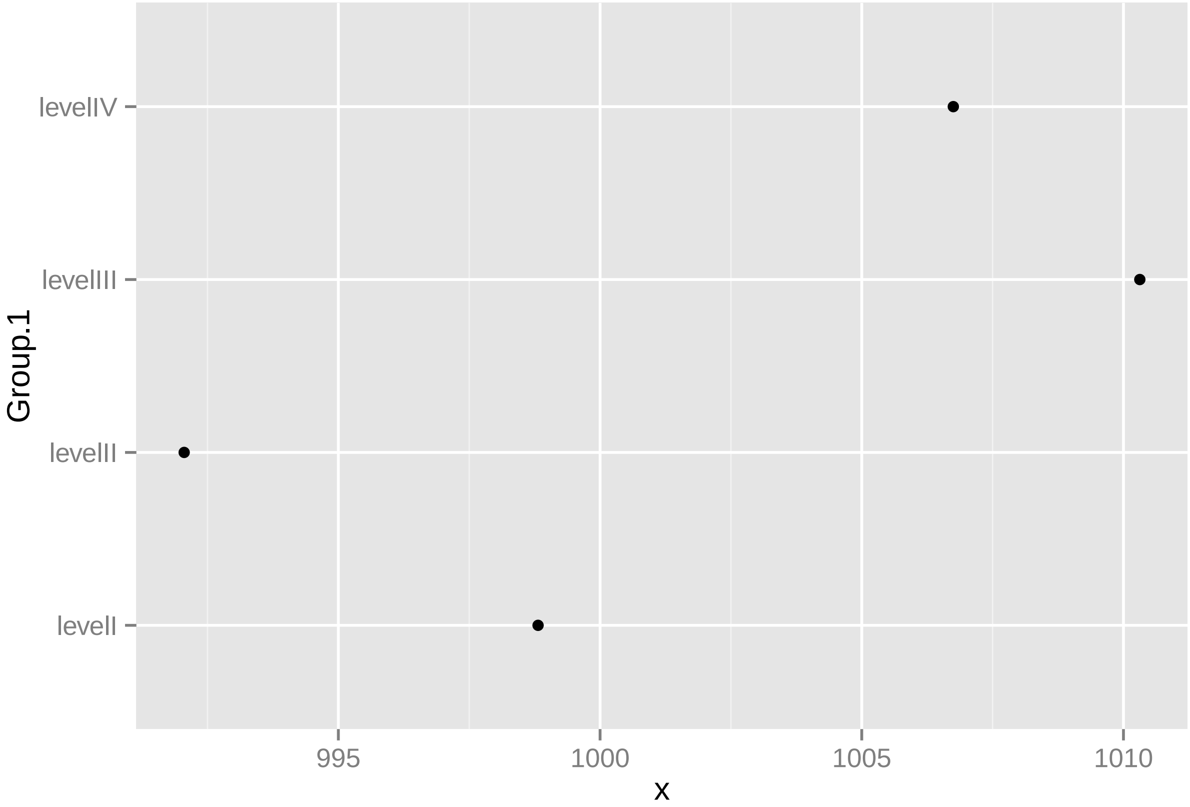


Two Variables: One Factor and One Numeric

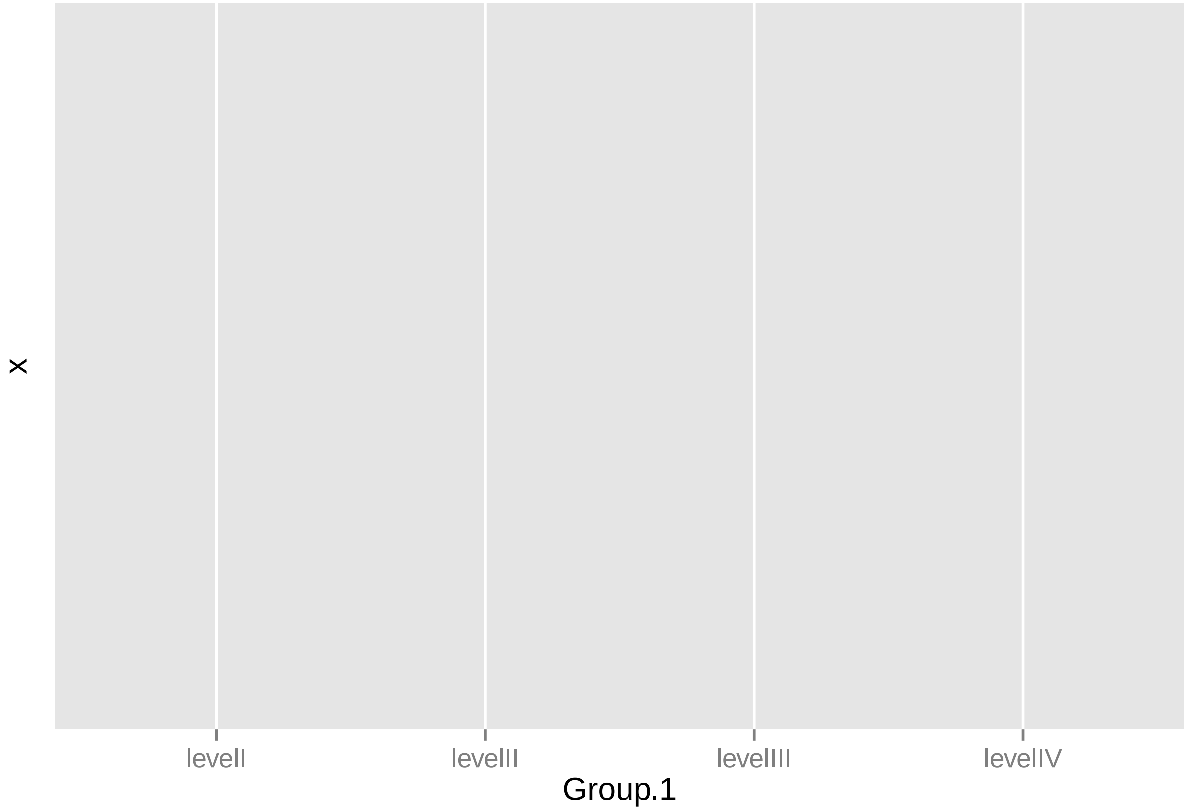
```
## density plot of numeric var across multiple levels of the factor var
ggplot(simData,aes(x=NumVar1,color=FacVar1))+geom_density()
```



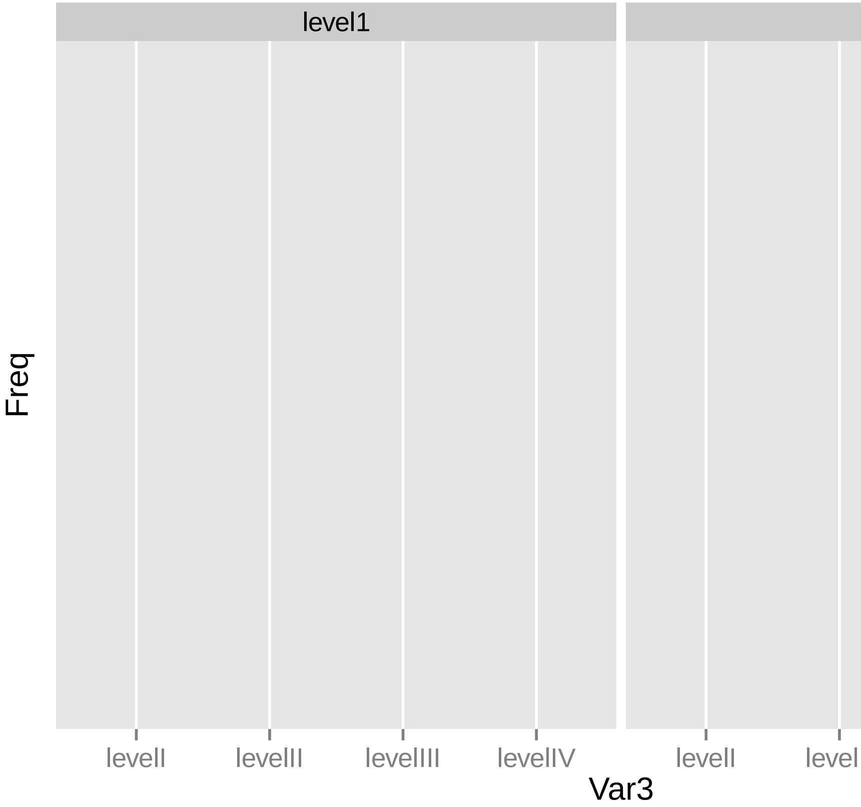
```
## Mean of one numeric var over levels of one factor var
meanagg = aggregate(simData$NumVar1, list(simData$FacVar3), mean)
ggplot(meanagg,aes(x=Group.1,y=x))+geom_point()+coord_flip() ## Dot Chart equivalent
```



```
ggplot(meanagg,aes(x=Group.1,y=x))+geom_bar() ## Bar plot
```

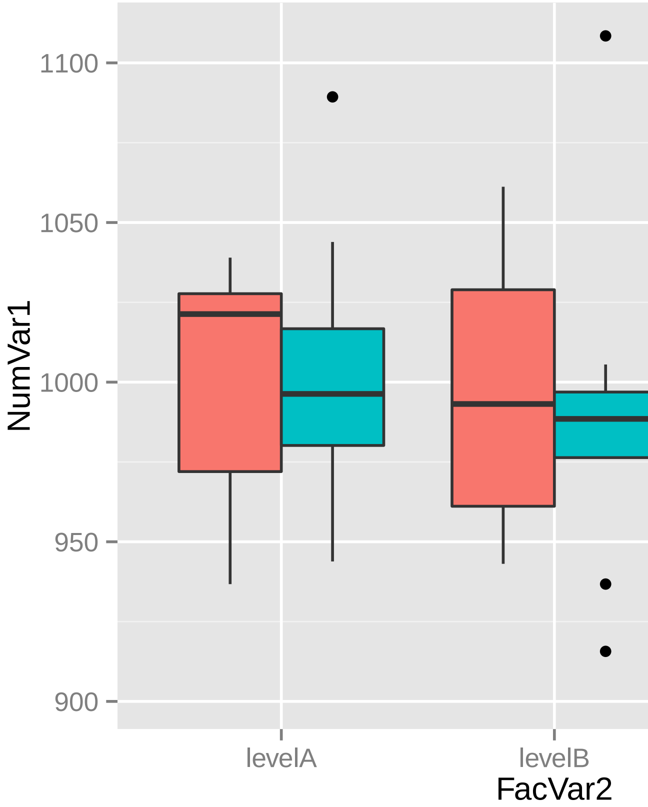


```
Threebartable = as.data.frame(table(simData$FacVar1, simData$FacVar2, simData$FacVar3)) ## CrossTab
ggplot(Threebartable,aes(x=Var3,y=Freq,fill=Var2))+geom_bar(position="dodge")+facet_wrap(~Var1) ## Bar
```



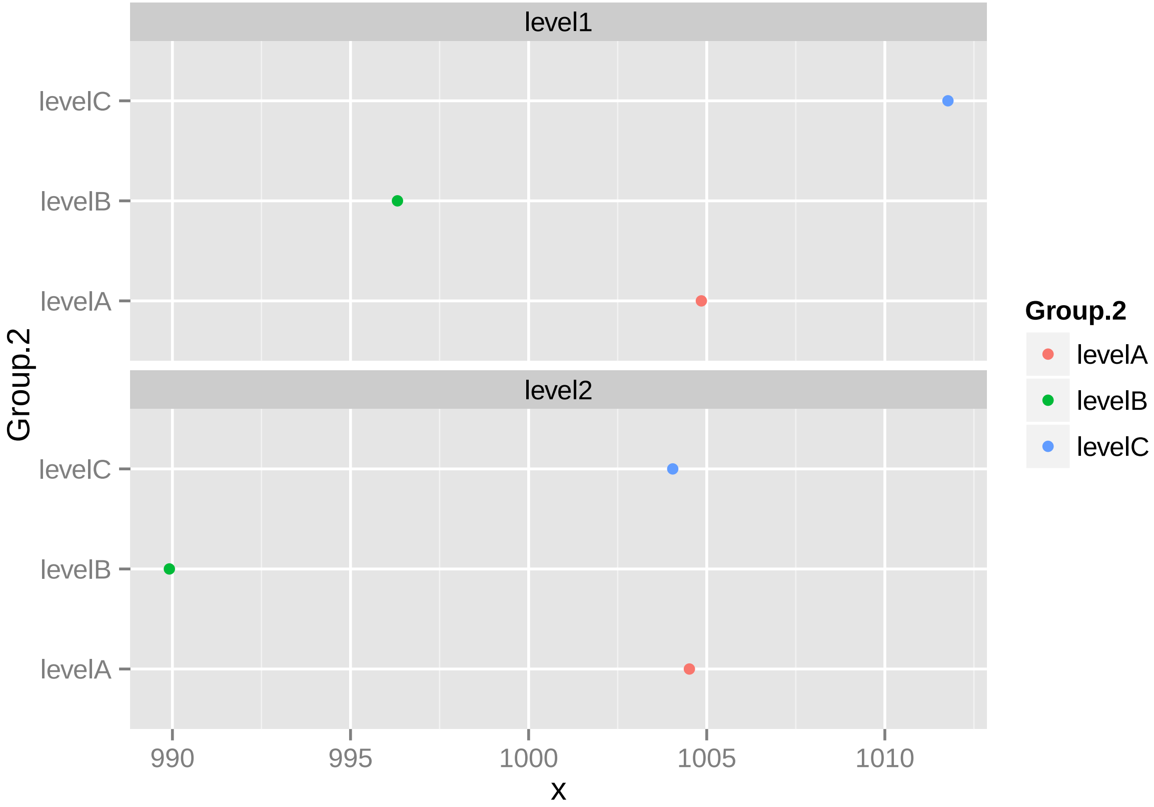
Three Variables: Three Factor Variables

```
## boxplot of NumVar1 over an interaction of 6 levels of the combination of FacVar1 and FacVar2
ggplot(simData,aes(x=FacVar2,y=NumVar1, fill=FacVar1))+geom_boxplot()
```

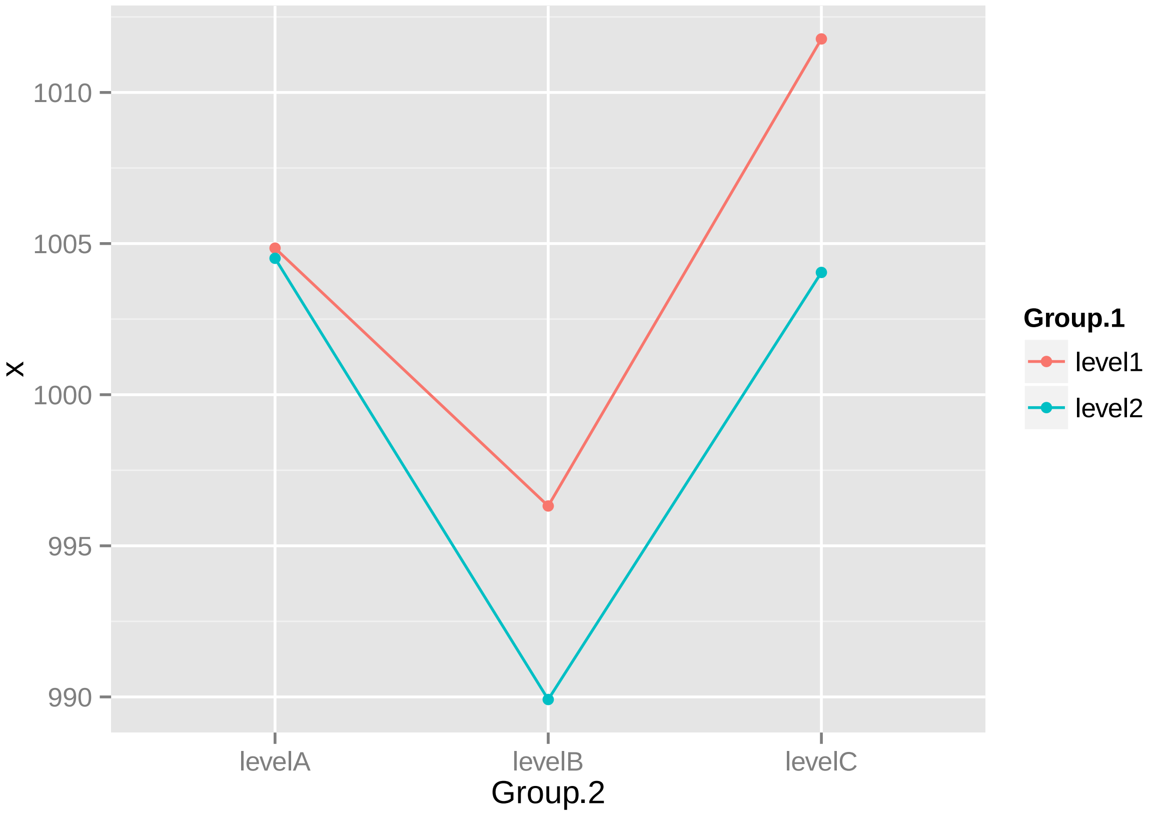


Three Variables: One Numeric and Two Factor Variables

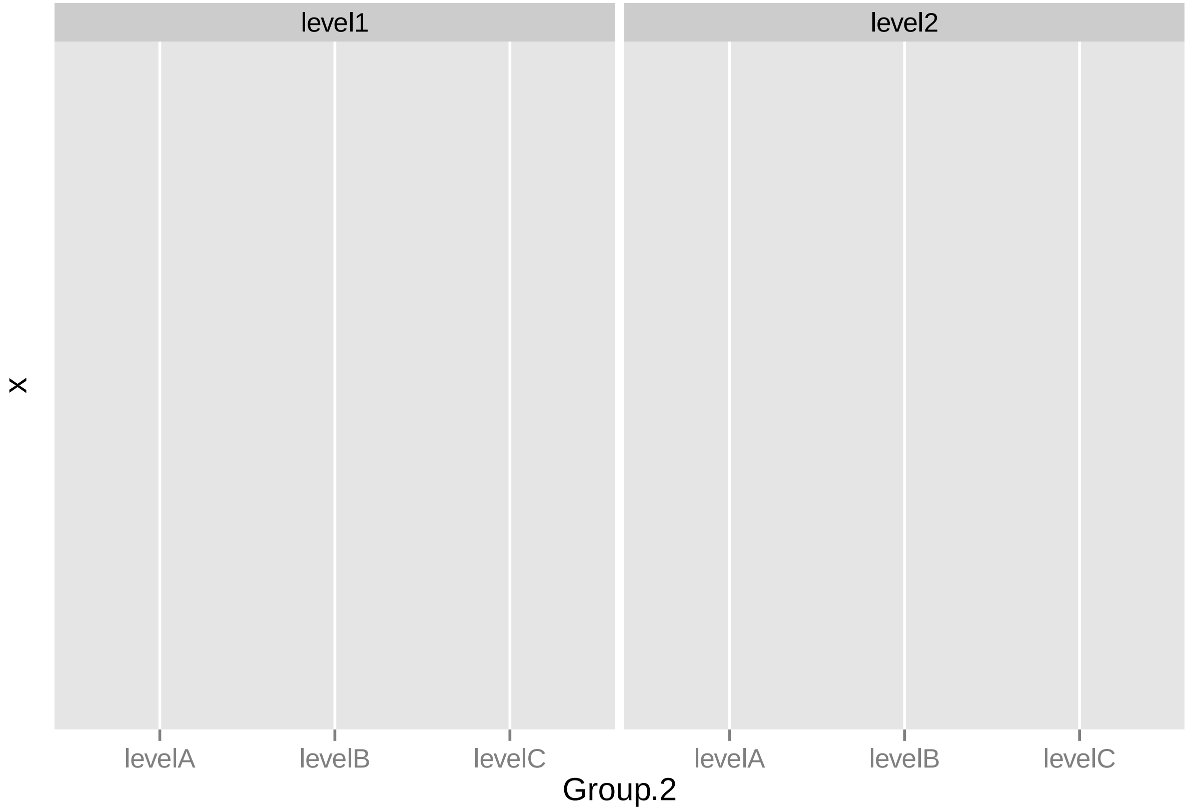
```
## Mean of 1 Numeric over levels of two factor vars
meanaggg = aggregate(simData$NumVar1, list(simData$FacVar1, simData$FacVar2), mean)
## Dot Chart equivalent
ggplot(meanaggg,aes(x=Group.2,y=x,color=Group.2))+geom_point()+coord_flip()+facet_wrap(~Group.1, ncol=1)
```



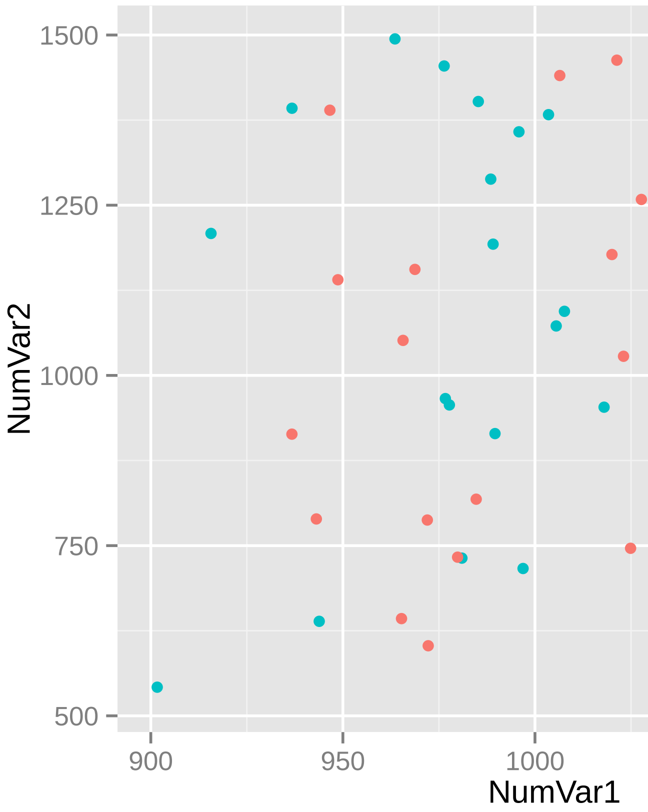
```
## Interaction chart - line chart
ggplot(meanaggg,aes(x=Group.2,y=x,color=Group.1, group=Group.1))+geom_point()+geom_line()
```



```
## And bar plot
ggplot(meanaggg,aes(x=Group.2,y=x))+geom_bar()+facet_wrap(~Group.1)
```

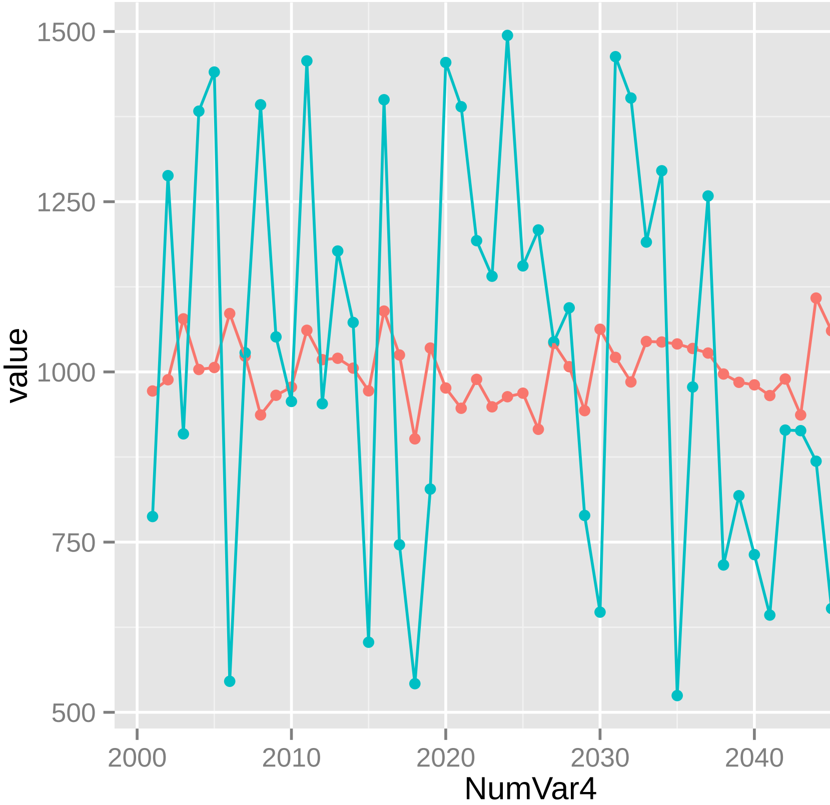


```
## Scatter plot with color identifying the factor variable
ggplot(simData,aes(x=NumVar1,y=NumVar2,color=FacVar1))+geom_point()
```



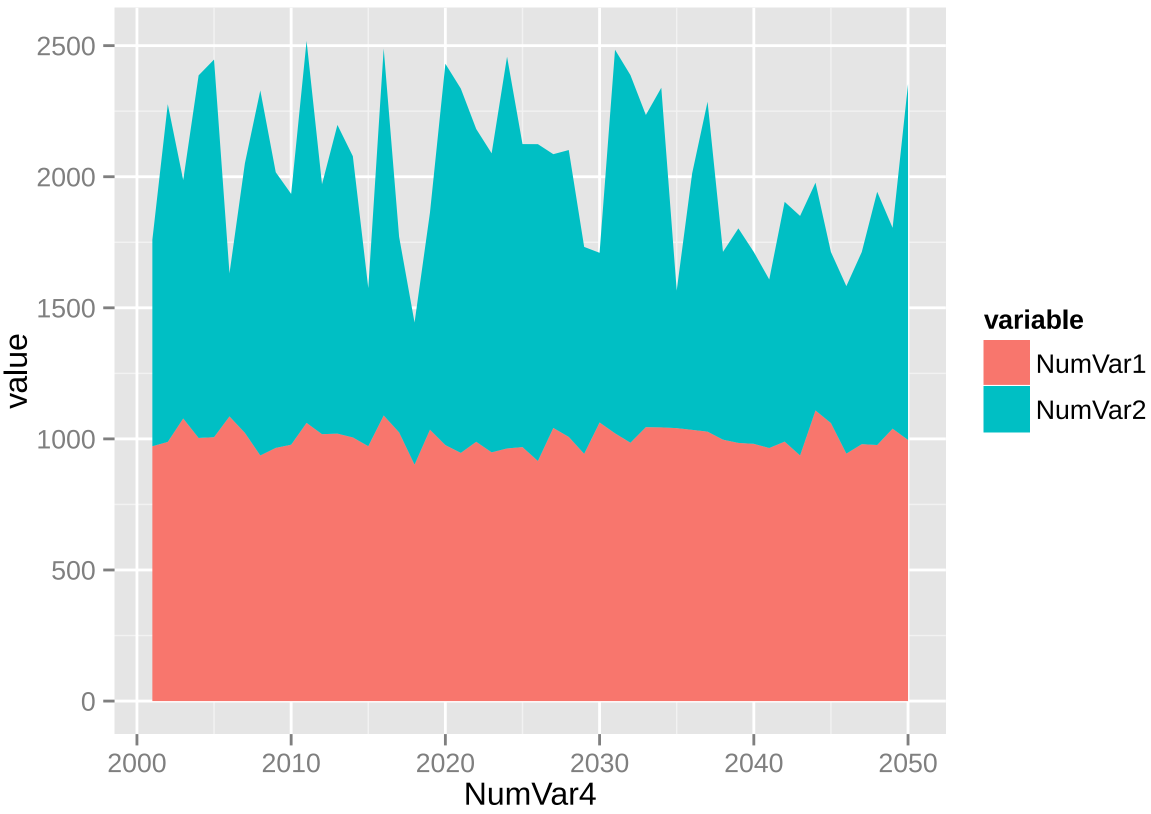
Three Variables: Two Numeric and One Factor Variables

```
## NumVar4 is 2001 through 2050... possibly, a time variable - use that as the x-axis
simtmpp=simData[,c(4,5,7)]
simtmppmelt=melt(simtmpp,id=c("NumVar4"))
ggplot(simtmppmelt,aes(x=NumVar4,y=value,color=variable,group=variable))+geom_point()+geom_line()
```

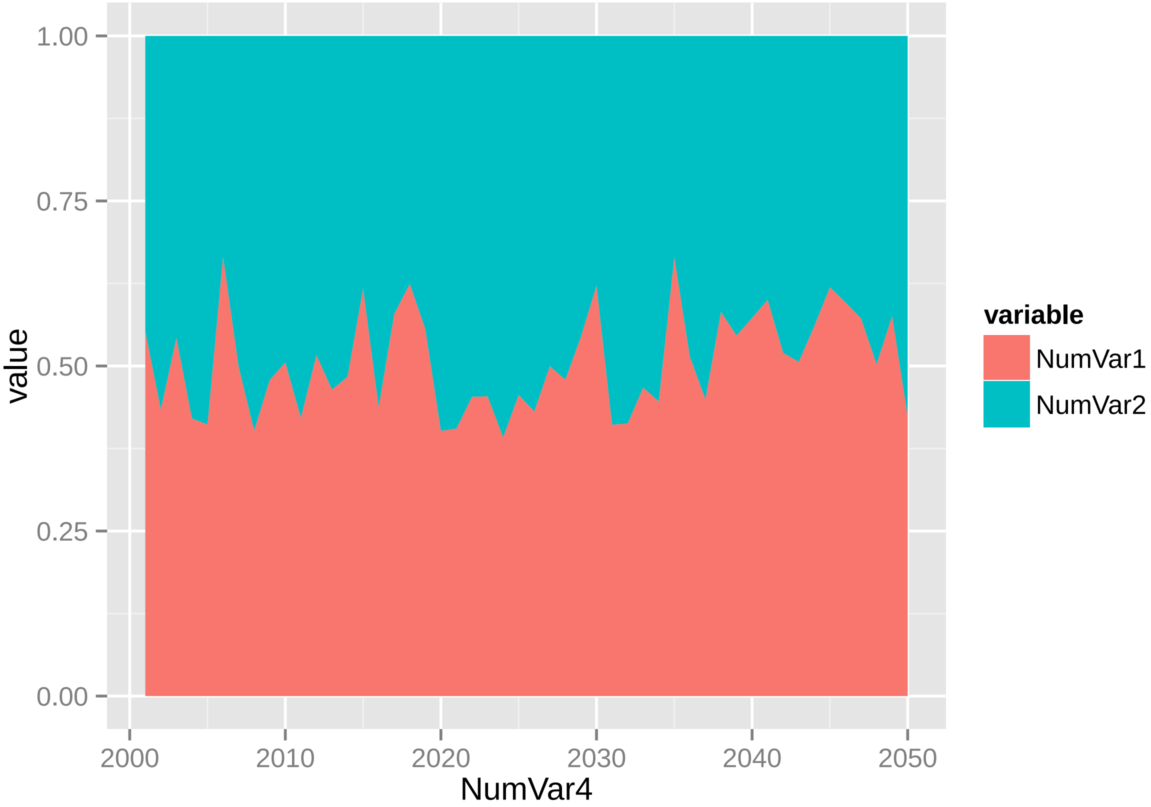


Three Variables: Three Numeric Variables

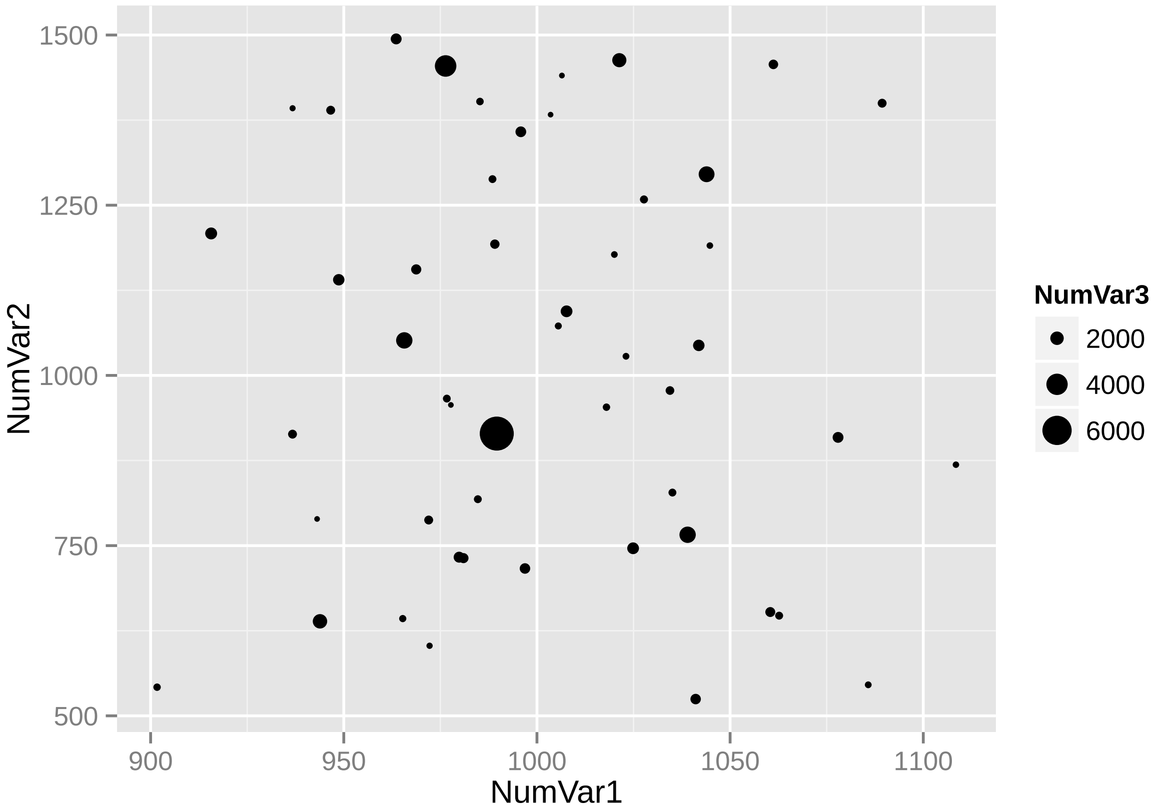
```
## Extra: Stacked Area Graph
ggplot(simtmppmelt,aes(x=NumVar4,y=value,fill=variable))+geom_area(position="stack")
```



```
## Extra: 100% stacked area graph
ggplot(simtmppmelt,aes(x=NumVar4,y=value,fill=variable))+geom_area(position="fill")
```

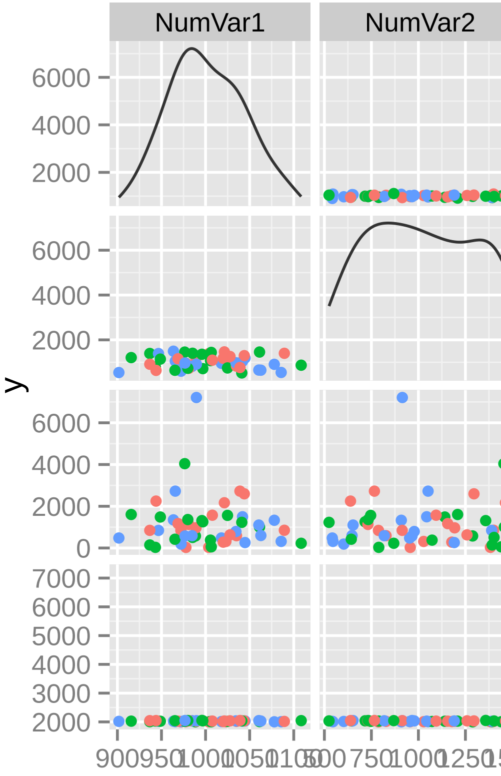


```
## ## Bubble plot - scatter plot of NumVar1 and NumVar2 with individual observations sized by NumVar3
ggplot(simData,aes(x=NumVar1,y=NumVar2,size=NumVar3))+geom_point()
```



```
#Thanks to Gaston Sanchez for the function: http://gastonsanchez.wordpress.com/2012/08/27/scatterplot-m
 makePairs <- function(data)
{
  grid <- expand.grid(x = 1:ncol(data), y = 1:ncol(data))
  grid <- subset(grid, x != y)
  all <- do.call("rbind", lapply(1:nrow(grid), function(i) {
    xcol <- grid[i, "x"]
    ycol <- grid[i, "y"]
    data.frame(xvar = names(data)[ycol], yvar = names(data)[xcol],
               x = data[, xcol], y = data[, ycol], data)
  }))
  all$xvar <- factor(all$xvar, levels = names(data))
  all$yvar <- factor(all$yvar, levels = names(data))
  densities <- do.call("rbind", lapply(1:ncol(data), function(i) {
    data.frame(xvar = names(data)[i], yvar = names(data)[i], x = data[, i])
  }))
  list(all=all, densities=densities)
}

## expanding numeric columns for pairs plot
gg1 = makePairs(simData[,4:7])

## new data frame
```

```
simDatabig = data.frame(gg1$all,simData[,1:3])

## pairs plot
ggplot(simDatabig, aes_string(x = "x", y = "y")) +
  facet_grid(xvar ~ yvar, scales = "free") +
  geom_point(aes(colour=FacVar2), na.rm = TRUE) +
  stat_density(aes(x = x, y = ..scaled.. * diff(range(x)) + min(x)),
               data = gg1$densities, position = "identity",
               colour = "grey20", geom = "line")
```



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

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/