

# Visualization with ggplot2

*Irfan Kanat*

*11/04/2015*

R has long been known for its extensive visualization capabilities. The number of packages that handle visualizations are many, yet ggplot shines among them all. Today I will focus on ggplot and discuss plotting histograms and scatter plots with qplot. I will focus mostly on qplot() function, and discuss ggplot structure only briefly.

## Introducing the Dataset

[In this document we will analyze the Motor Trends data.](#) The dataset was compiled from 1974 issues of Motor Trends magazine and is included with R Base package.

Let us start with loading the dataset.

```
data(mtcars)
```

As we learned in the section on packages, you can query the documentation for almost anything. Including the datasets included in packages. The document includes descriptions of the variables.

```
?mtcars
```

Let us get a sense of the data.

```
# A summary of variables  
summary(mtcars)
```

```
##      mpg          cyl          disp          hp  
##  Min.   :10.40   Min.   :4.000   Min.    : 71.1   Min.    : 52.0  
## 1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8   1st Qu.: 96.5  
## Median :19.20   Median :6.000   Median :196.3   Median :123.0  
## Mean   :20.09   Mean    :6.188   Mean    :230.7   Mean    :146.7  
## 3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0   3rd Qu.:180.0  
## Max.   :33.90   Max.    :8.000   Max.    :472.0   Max.    :335.0  
##      drat          wt          qsec          vs  
##  Min.    :2.760   Min.    :1.513   Min.    :14.50   Min.    :0.0000  
## 1st Qu.:3.080   1st Qu.:2.581   1st Qu.:16.89   1st Qu.:0.0000  
## Median :3.695   Median :3.325   Median :17.71   Median :0.0000  
## Mean    :3.597   Mean    :3.217   Mean    :17.85   Mean    :0.4375  
## 3rd Qu.:3.920   3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.:1.0000  
## Max.    :4.930   Max.    :5.424   Max.    :22.90   Max.    :1.0000  
##      am          gear          carb  
##  Min.    :0.0000   Min.    :3.000   Min.    :1.000  
## 1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:2.000  
## Median :0.0000   Median :4.000   Median :2.000  
## Mean    :0.4062   Mean    :3.688   Mean    :2.812  
## 3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:4.000  
## Max.    :1.0000   Max.    :5.000   Max.    :8.000
```

```
# Correlation table for first 4 variables (due to space concerns)
cor(mtcars[,1:4])
```

```
##           mpg           cyl           disp           hp
## mpg    1.0000000 -0.8521620 -0.8475514 -0.7761684
## cyl   -0.8521620  1.0000000  0.9020329  0.8324475
## disp  -0.8475514  0.9020329  1.0000000  0.7909486
## hp    -0.7761684  0.8324475  0.7909486  1.0000000
```

```
# bivariate comparisons of categorical variables
table(mtcars[,c("am", "cyl")])
```

```
##      cyl
## am   4   6   8
##    0   3   4  12
##    1   8   3   2
```

```
# The histogram below should reflect these figures.
table(mtcars$gear)
```

```
##
##  3  4  5
## 15 12  5
```

## Plotting with qplot()

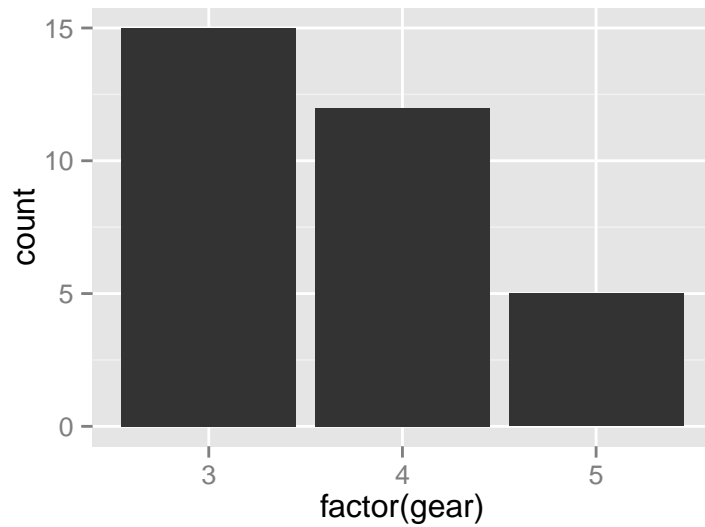
Now we can get to the fun part. qplot simplifies the ggplot functionality by automating most common tasks. We will use qplot for most common plots.

```
# Load the ggplot package
library(ggplot2)
# Review function syntax
?qplot
```

## Histogram

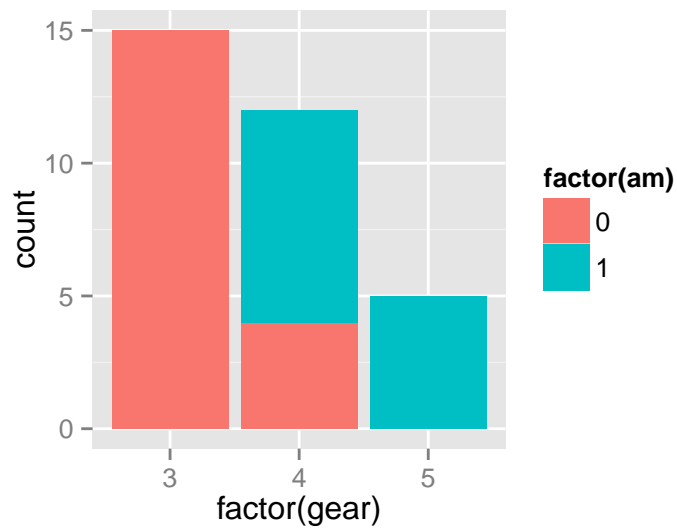
You would use a histogram when you are interested in frequencies of certain categories, like number of people with different eye colors.

```
# Let us report the number of cars with differing number of front gears
qplot(factor(gear), data=mtcars, geom="bar") # used factor to declare categorical
```



If we want to get fancy and want to report across two categorical variables we can color the bars based on another variable.

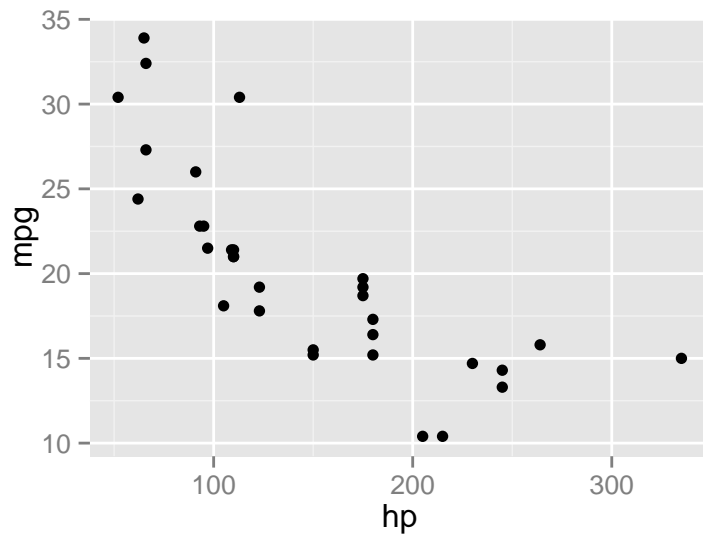
```
qplot(factor(gear), data=mtcars, fill=factor(am), geom="bar") # used factor to declare categorical
```



## Scatter Plots

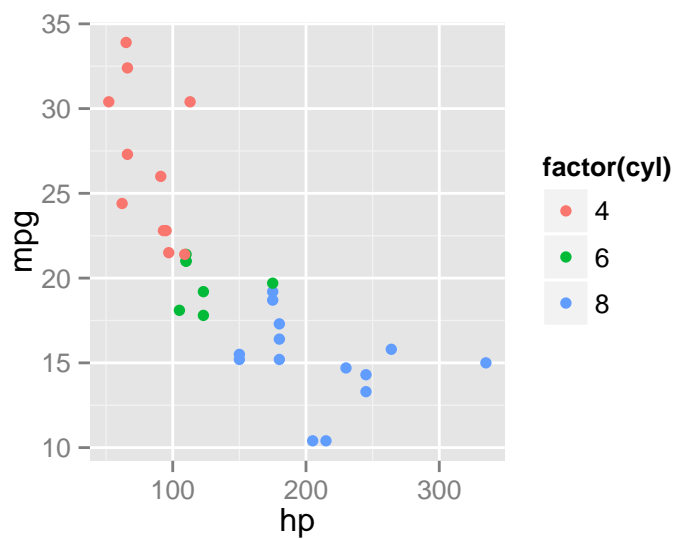
If you are interested in the relationship between two continuous variables, you can use scatter plots.

```
qplot(hp, mpg, data=mtcars)
```



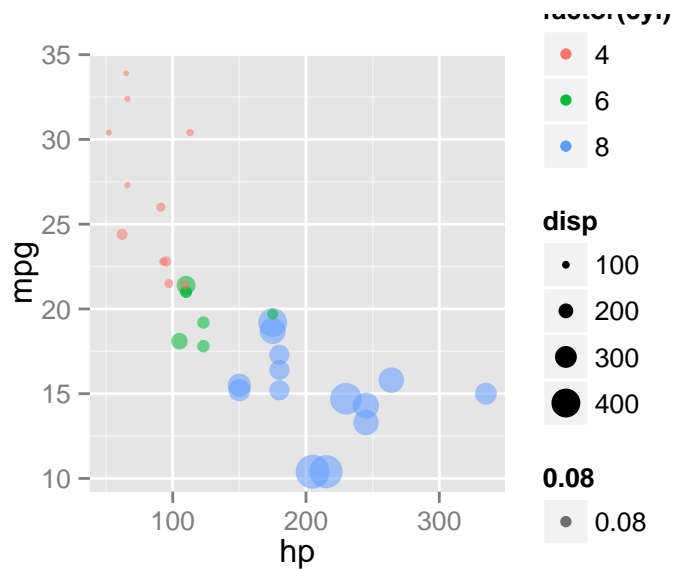
Let us impose an additional factor into the plot. Let us color the dots by the number of cylinders.

```
qplot(hp, mpg, data=mtcars, color=factor(cyl))
```



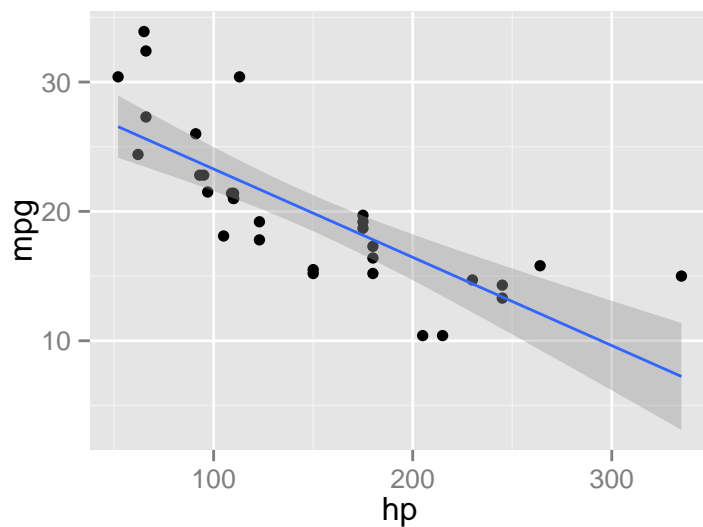
Size of dots dependent on a continuous variable (displacement).

```
qplot(hp, mpg, data=mtcars, color=factor(cyl), size=disp, alpha=.08)
```



Let us fit a regression line. This is where things start to get a bit ggplotty.

```
qplot(hp, mpg, data=mtcars) +  
  geom_smooth(method=lm, sd=F)
```



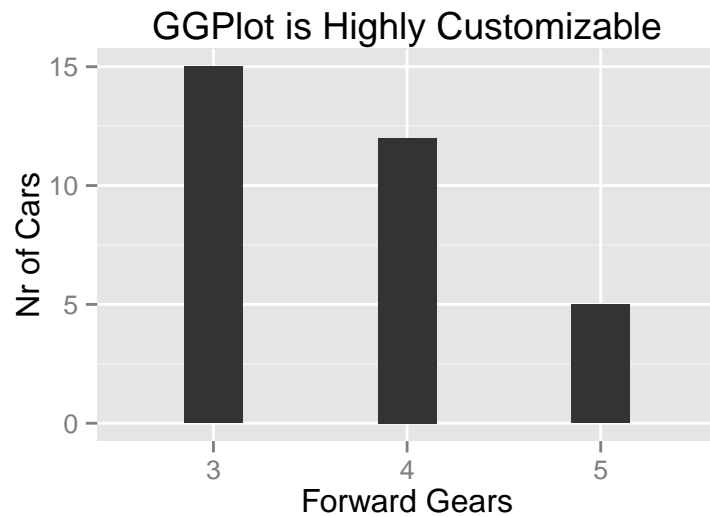
## ggplot

qplot provides a convenient command for plotting. While qplot would address 90% of your plotting needs, ggplot is way more than qplot, it is almost a different language just for plotting. The intricacies may be hard to learn and is clearly beyond the scope of this workshop. I am providing ggplot code below to achieve the same results as the qplot, so the attendees can get a sense of what ggplot is really about.

## Histogram

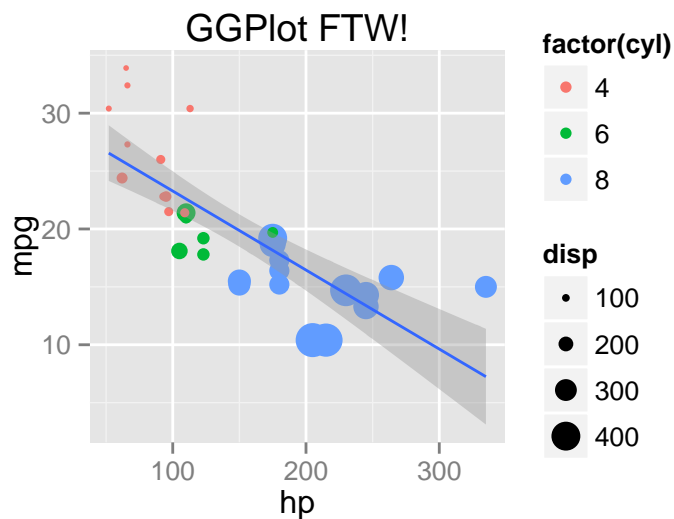
```
# Initialize the plot with variables of interest  
ggplot(mtcars, aes(factor(gear))) +
```

```
# Instruct ggplot to plot bars of width .3
geom_bar(stat = "bin", width=0.3) +
ggtitle('GGPlot is Highly Customizable') +
xlab('Forward Gears') +
ylab('Nr of Cars')
```



## Scatter Plot

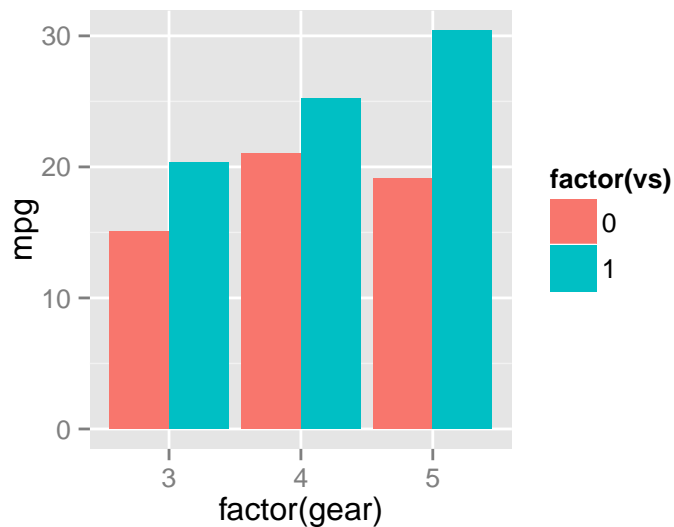
```
ggplot(mtcars, aes(x=hp, y=mpg)) +
  geom_point(aes(color=factor(cyl), size=disp)) + # For scatter plot
  geom_smooth(method=lm) + # Add a regression line
  ggtitle('GGPlot FTW!') # Add a title
```



## Bar Charts

You use bar charts when you want to visualize the relationship of a continuous variable over a categorical variable (eg. gender-height). Here I plot mean mpg over two categorical variables.

```
ggplot(mtcars, aes(x=factor(gear), y=mpg, fill=factor(vs)), color=factor(vs)) +
  stat_summary(fun.y=mean, position=position_dodge(), geom="bar")
```



## Bonus! World Map Visualizations

After having mingled that GDP data so much, I could not help but plot the results on a map. Here is how:  
Process data into a map.

```
# Load the Library
library(rworldmap) # Install if necessary
```

```
## Loading required package: sp
## ### Welcome to rworldmap ###
## For a short introduction type : vignette('rworldmap')
```

```
# Get the dataset
cData_Long<-read.csv("cDataLong.csv")

# Subset
cDataL2011 <- subset(cData_Long, time==2011)

# Get a taste
head(cDataL2011)
```

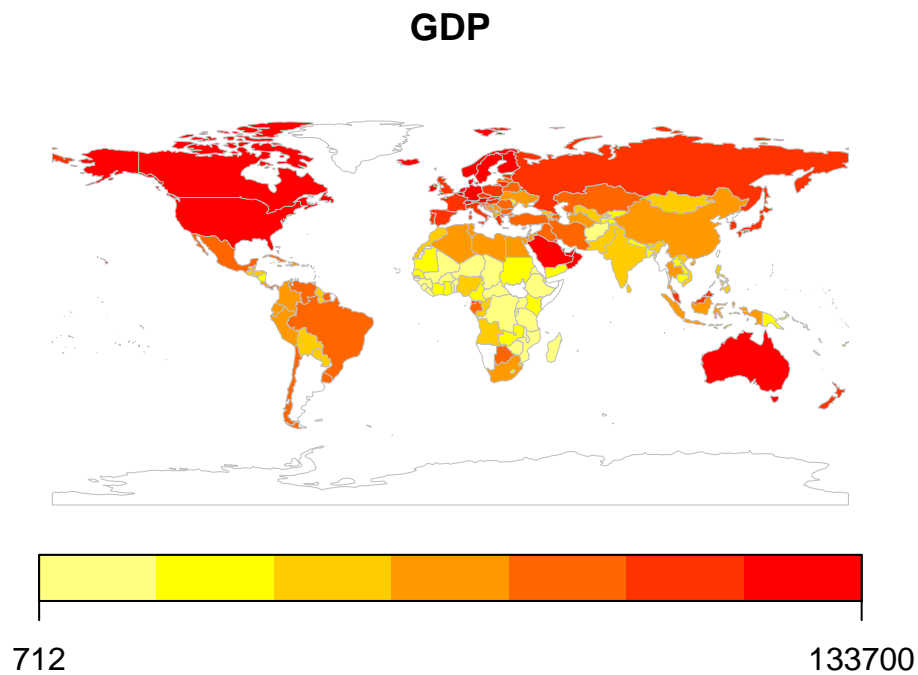
```
##   IS02 Continent time      GDP    logGDP   cumGDP   lagGDP
## 9    AE         AS 2011 56376.770 10.939812 754381.32 57379.972
## 19   AF         AS 2011 1695.153  7.435529  11970.69 1637.297
## 29   AG         AN 2011 19987.924  9.902884 200080.43 20567.359
## 39   AL         EU 2011  9897.180  9.200005  73030.75  9559.157
## 49   AM         AS 2011  6812.352  8.826493  54071.70  6507.914
## 59   AO         AF 2011  7094.084  8.867017  52753.28  7047.052
```

```
# Turn into map
cDataL2011<- joinCountryData2Map(cDataL2011, joinCode = "ISO2",
                                nameJoinColumn = 'ISO2')
```

```
## 187 codes from your data successfully matched countries in the map
## 0 codes from your data failed to match with a country code in the map
## 55 codes from the map weren't represented in your data
```

Plot the world map.

```
mapCountryData(cDataL2011, nameColumnToPlot = "GDP")
```

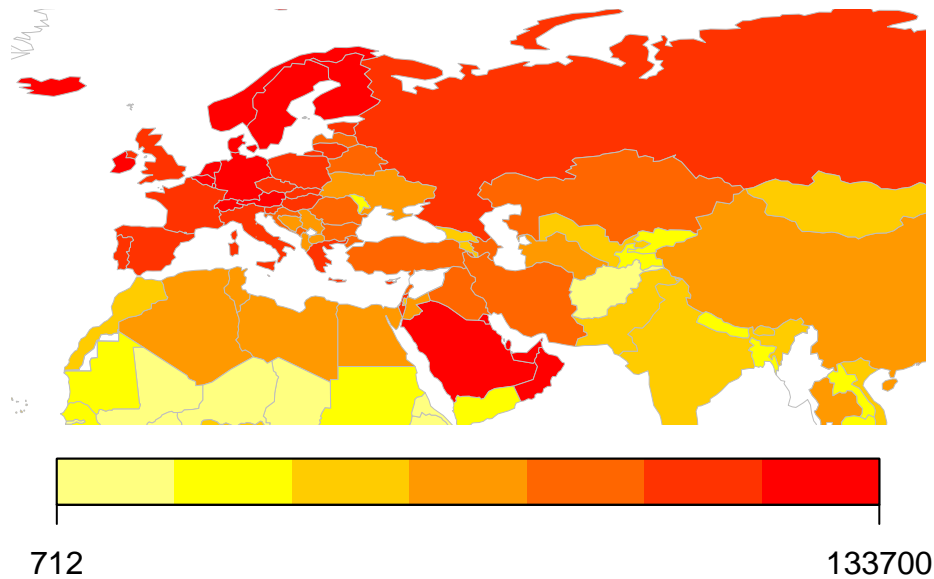


Focus on Eurasia.

```
mapCountryData(cDataL2011, nameColumnToPlot = "GDP", mapRegion = "eurasia")
```



## GDP



Plot a categorical variable as color and continuous as size.

```
mapDevice() # Initialize the map
mapBubbles(dF=cDataL2011, nameZSize="GDP", nameZColour="Continent", colourPalette="rainbow", oceanCol="")
```

## caret and Visualizations

We will cover caret a little in machine learning samples. This package provides convenient shortcuts to ggplot functionality. Simplifying most common plotting tasks in machine learning. Please [refer to the project page](#) for further reference.



How I Learned to Stop Worrying and Love the R Console by [Irfan E Kanat](#) is licensed under a [Creative Commons Attribution 4.0 International License](#). Based on a work at <http://github.com/iekanat/rworkshop>.