Quickplot

- The `qplot()` function is supposed to make the same sort of plots as `ggplot()`, but with simpler syntax.
- In practice, for more complex plots, it is probably more straightforward to use `ggplot()`.
- Many of the arguments supplied to `plot()` (e.g. main, xlab, ylab etc) can be supplied to `qplot()`.

Motor Trend Car Road Tests

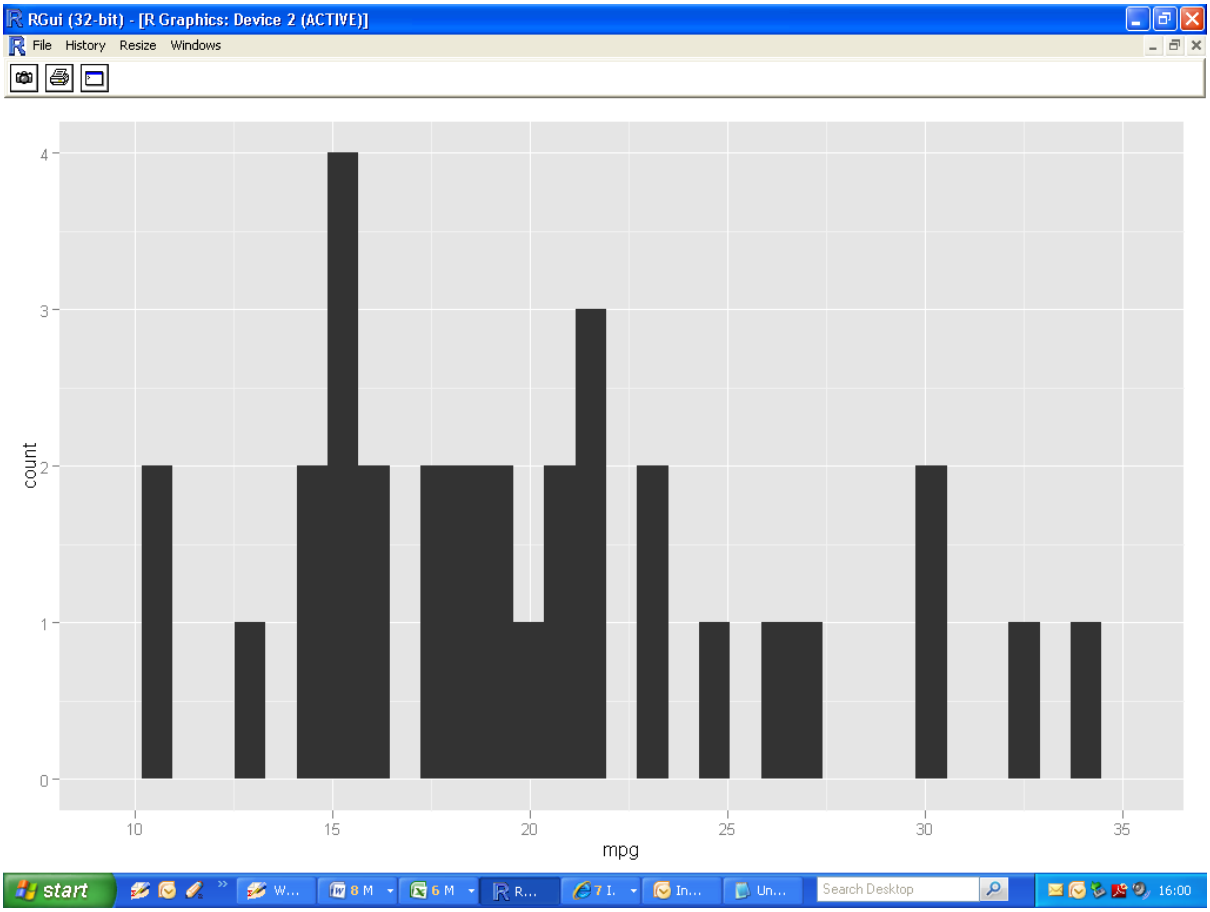
This data set was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973–74 models).

(Source: Henderson and Velleman (1981), Building multiple regression models interactively. *Biometrics*, 37, 391–411)

Simple histogram using quickplot

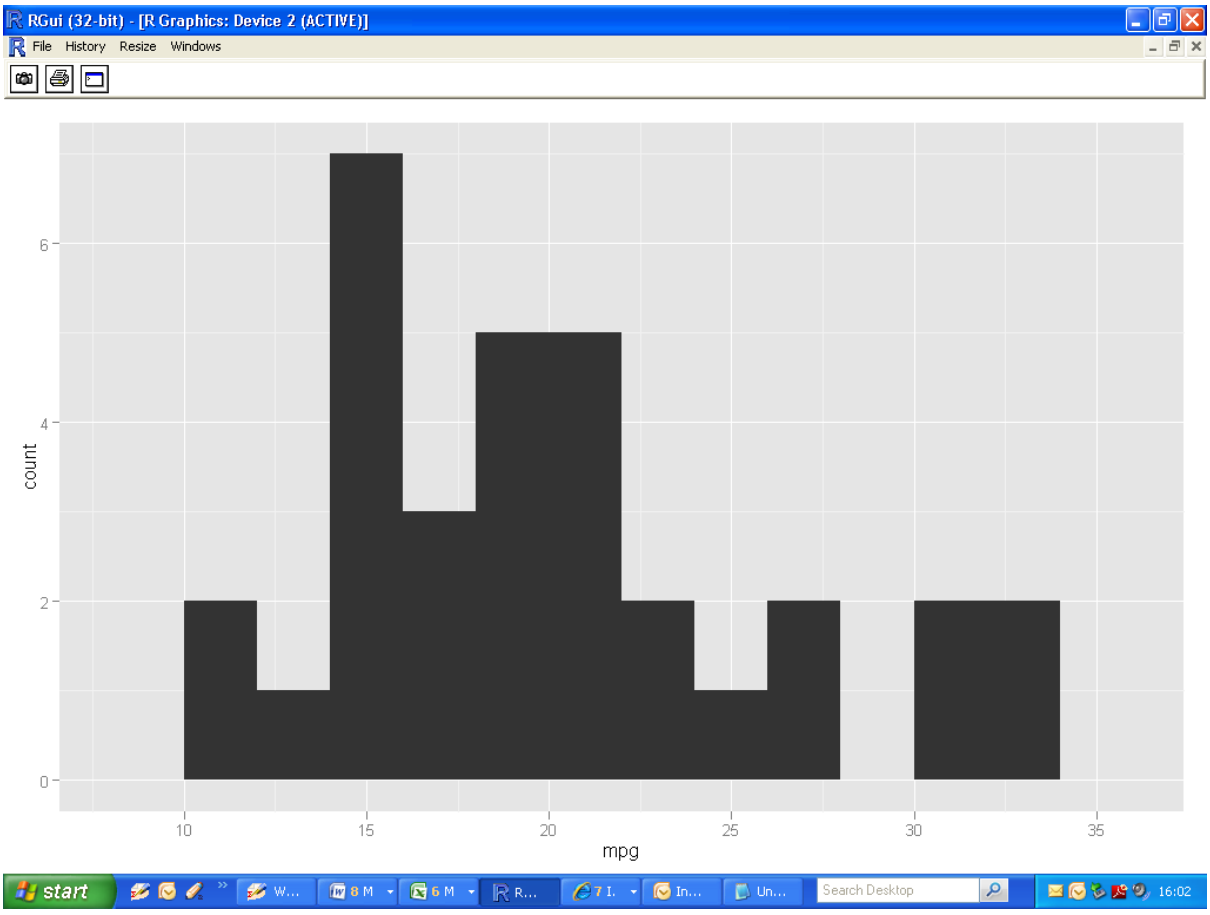
The basic syntax of the command is

```
qplot(x.var,data=dataset.name)
```



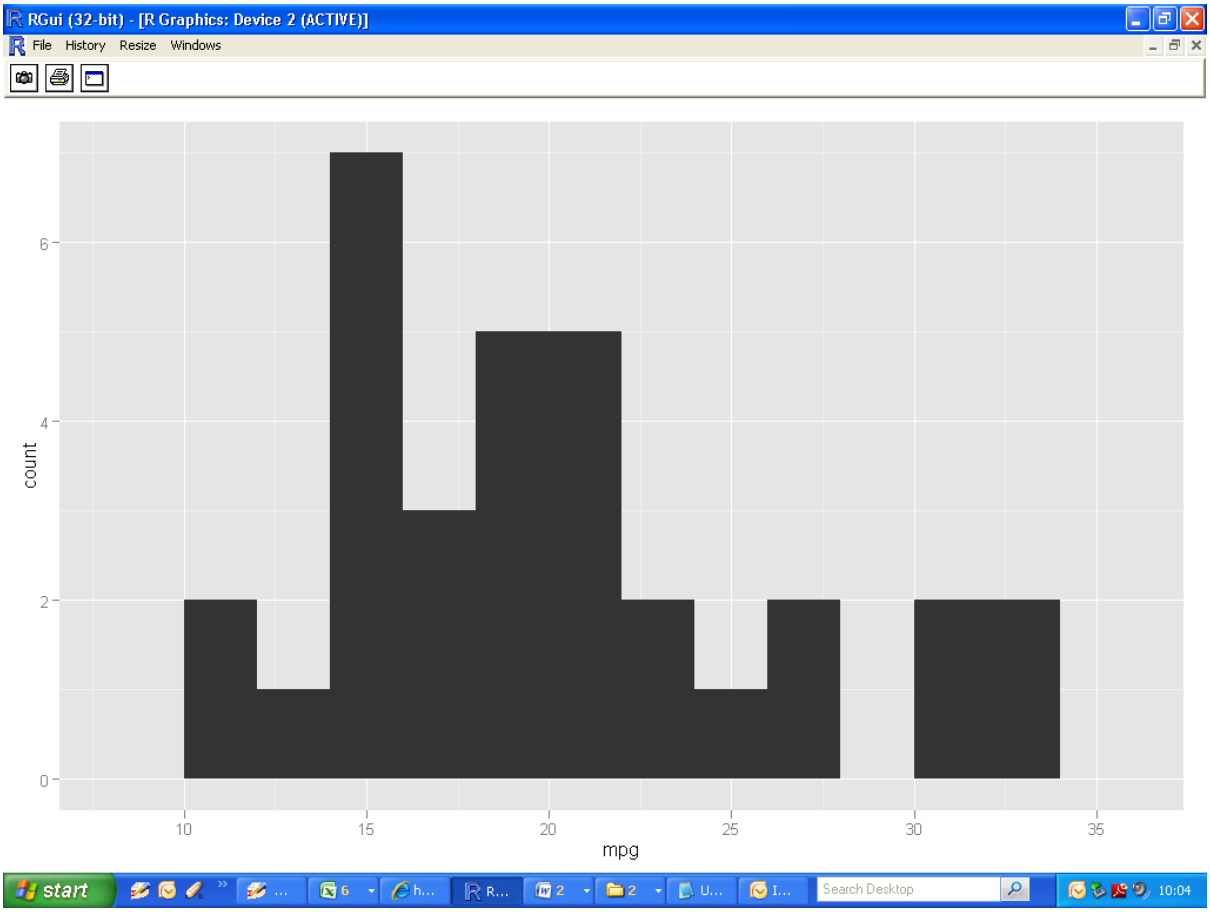
Adjusting bin-width

```
>
>qplot(mpg,data=mtcars, binwidth=2)
>
```



Histogram geom

```
> qplot(mpg,data=mtcars, geom=c("histogram"),
binwidth=2)
```



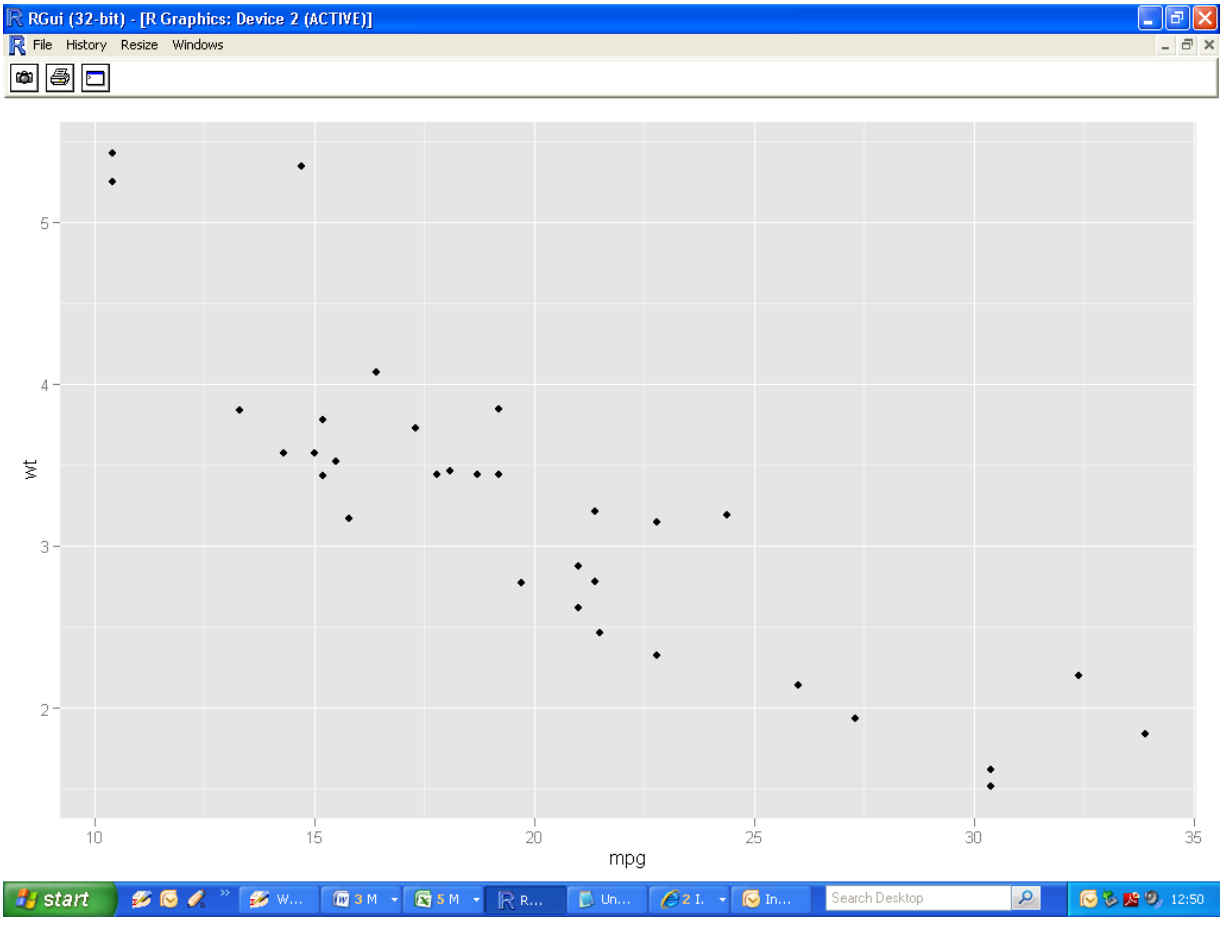
Simple scatterplot using quickplot

The basic syntax of the command is

```
qplot(x.var,y.var,data="dataset.name")
```

- Dataset name : mtcars
- X variable: Miles Per Gallon (mpg)
- Y variable: Weight (wt)

```
>
> qplot(mpg,wt,data=mtcars)
>
```



Immediately noticeable:

- Glyph is little black dot

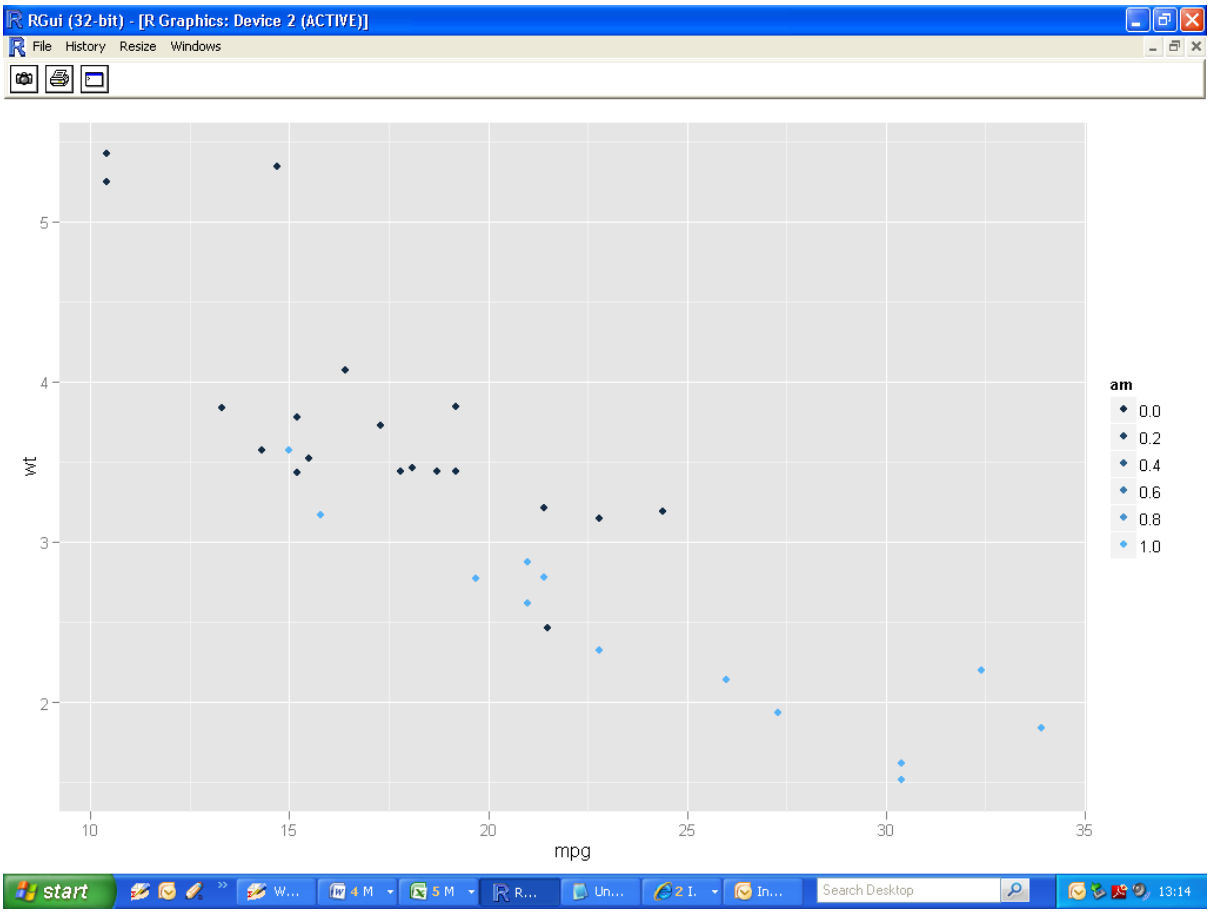
- Grey background with white gridlines (This is called a “theme”. More later)

Subsetting

Those familiar with the mtcars dataset would be aware of other (categorical) variables. For example

- cyl : number of cylinders (4, 6 or 8?)
- vs : (binary variable)
- am : Transmission (0 = automatic, 1 = manual)

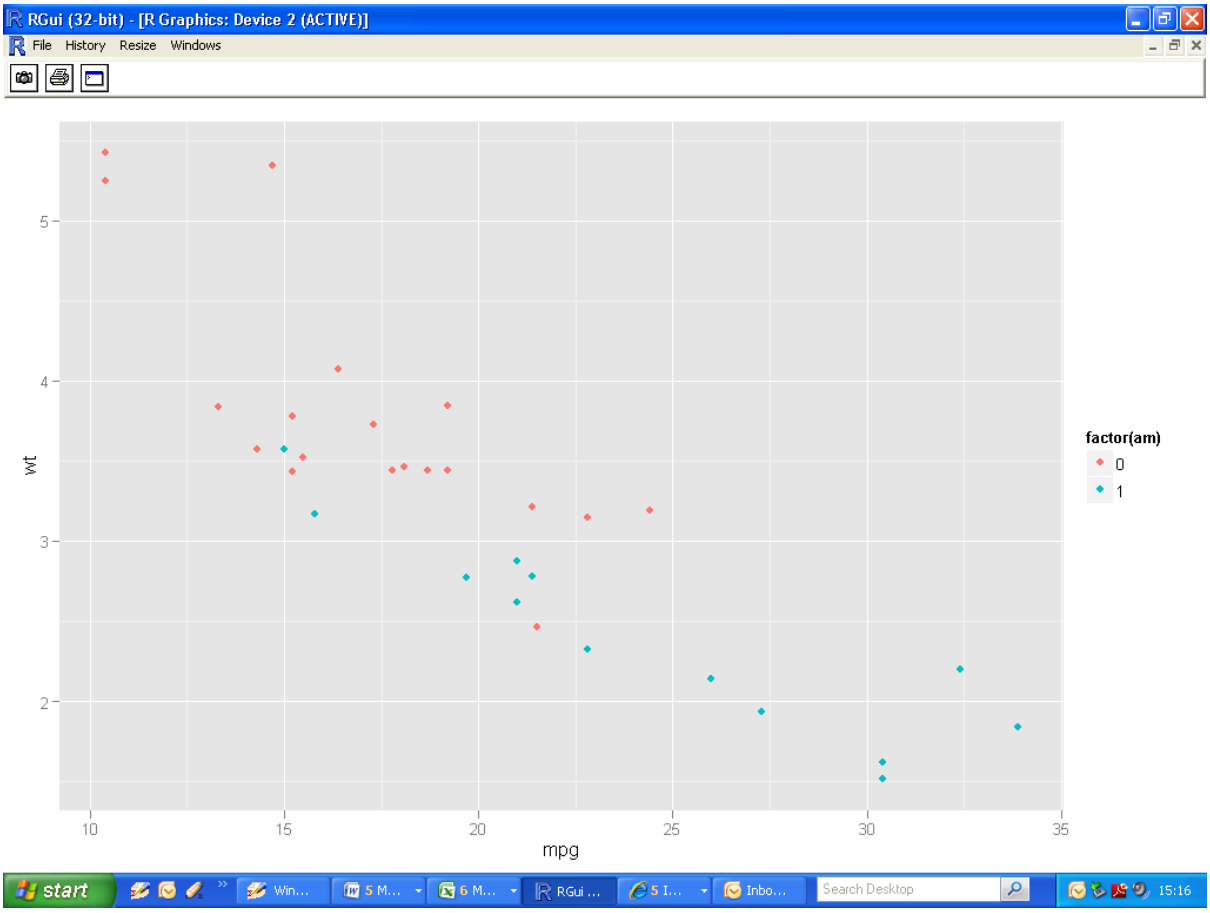
```
>
>qplot(mpg,wt,data=mtcars, colour=am)
>
```



- Divide the scatterplot into two clusters (automatic and transmission). Useful visual aid.

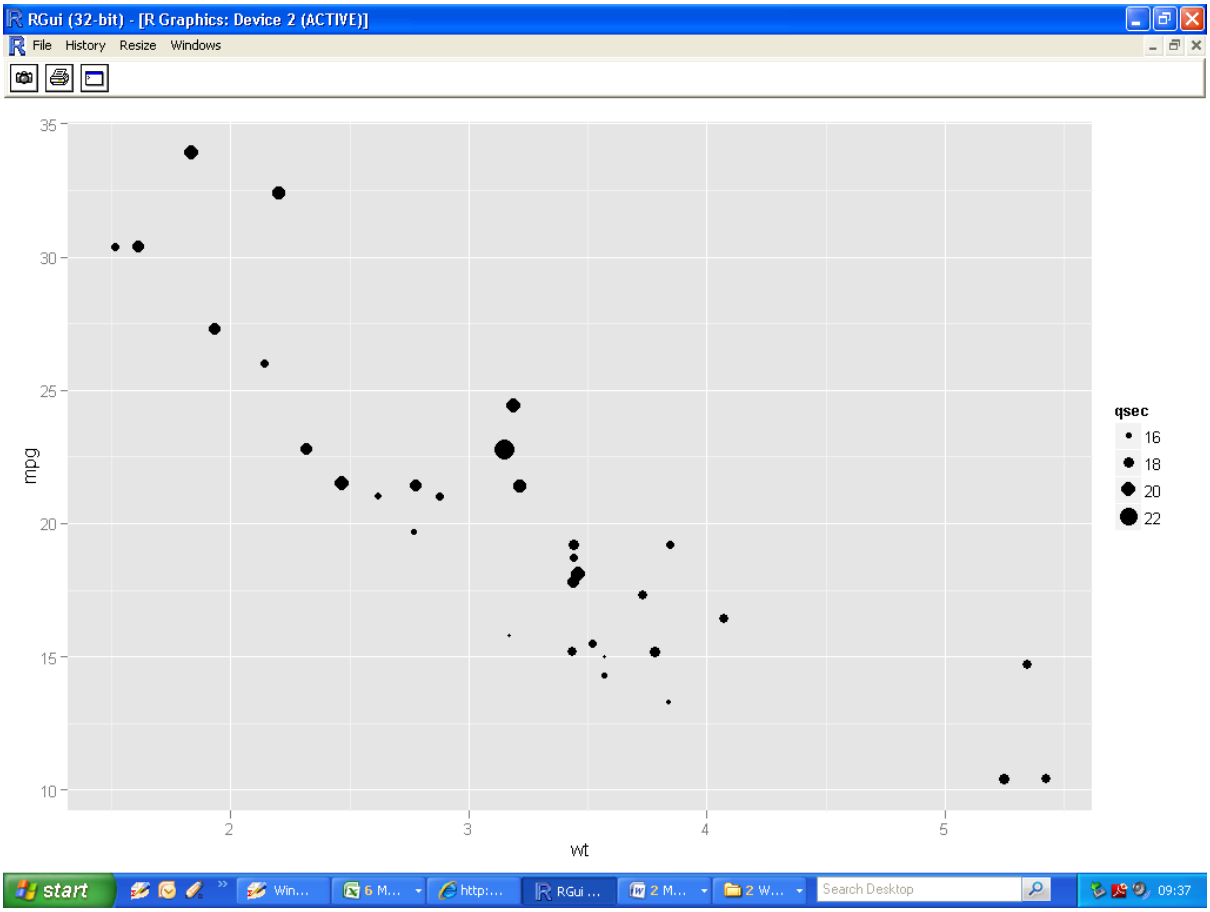
- However, notice the legend – we have 6 categories for the variable “am” (which is in fact binary). Need to fix this; use the function `factor()`.

```
>
>qplot(mpg,wt,data=mtcars, colour=factor(am))
>
```



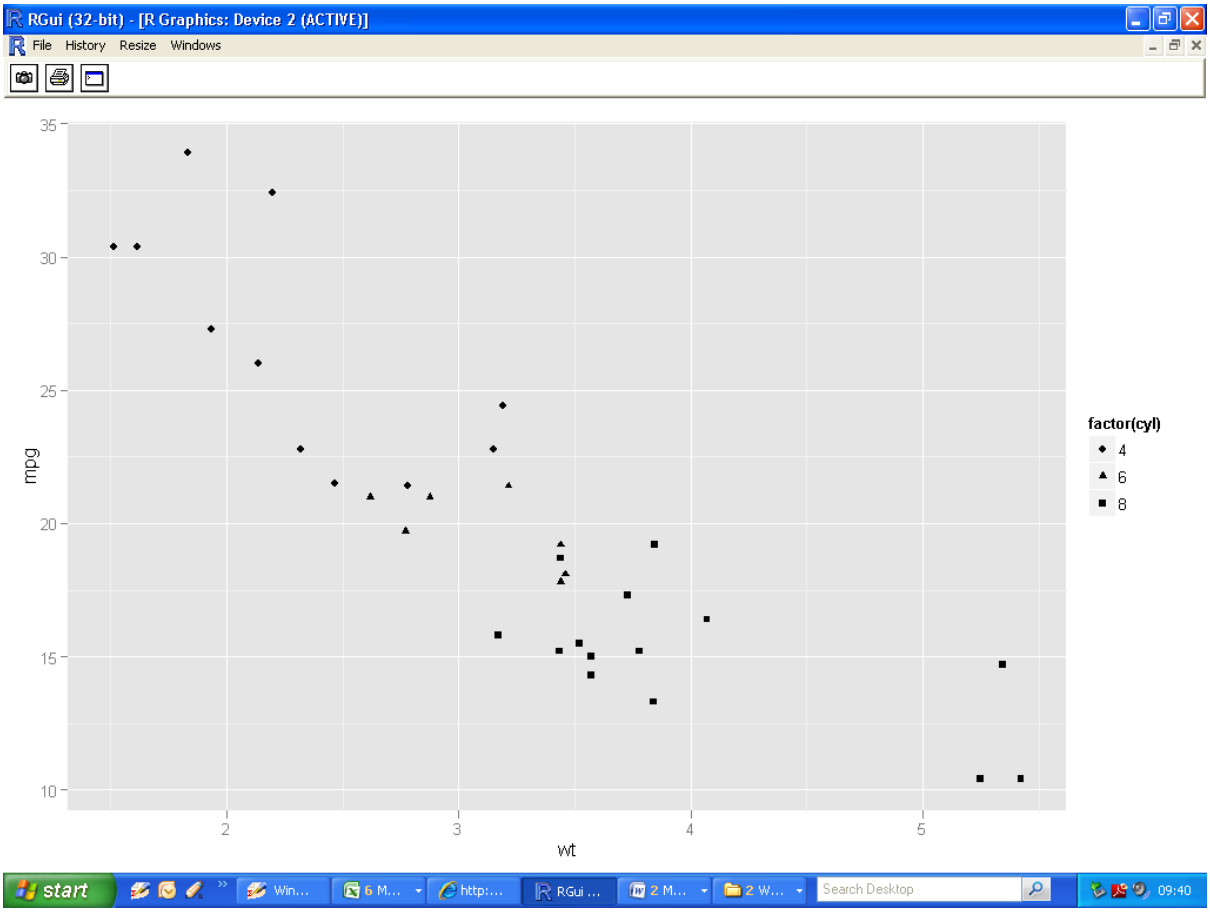
Using different aesthetic mappings : size

```
> qplot(wt, mpg, data=mtcars, size=qsec)
>
```



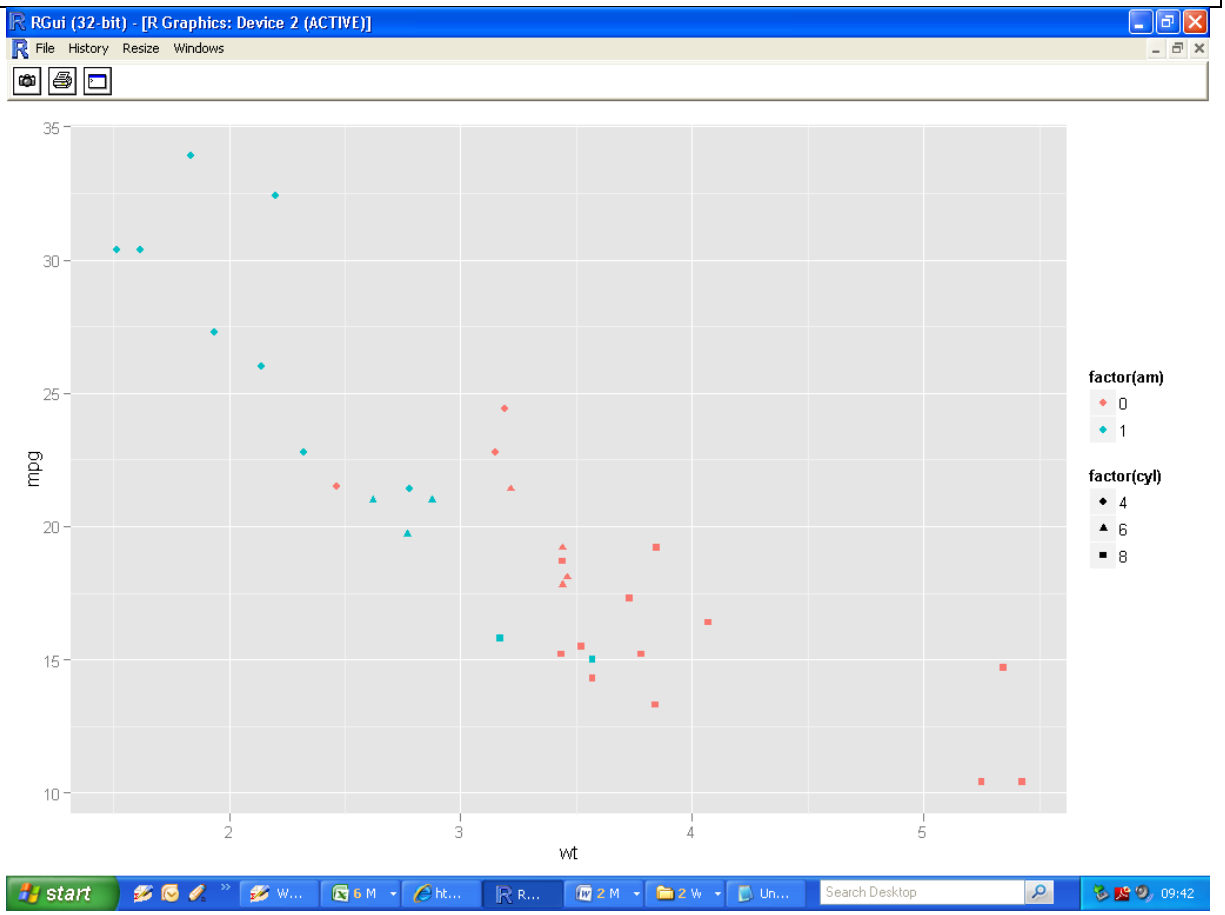
Using different aesthetic mappings : shape

```
> qplot(wt, mpg, data=mtcars, shape=factor(cyl))
>
```



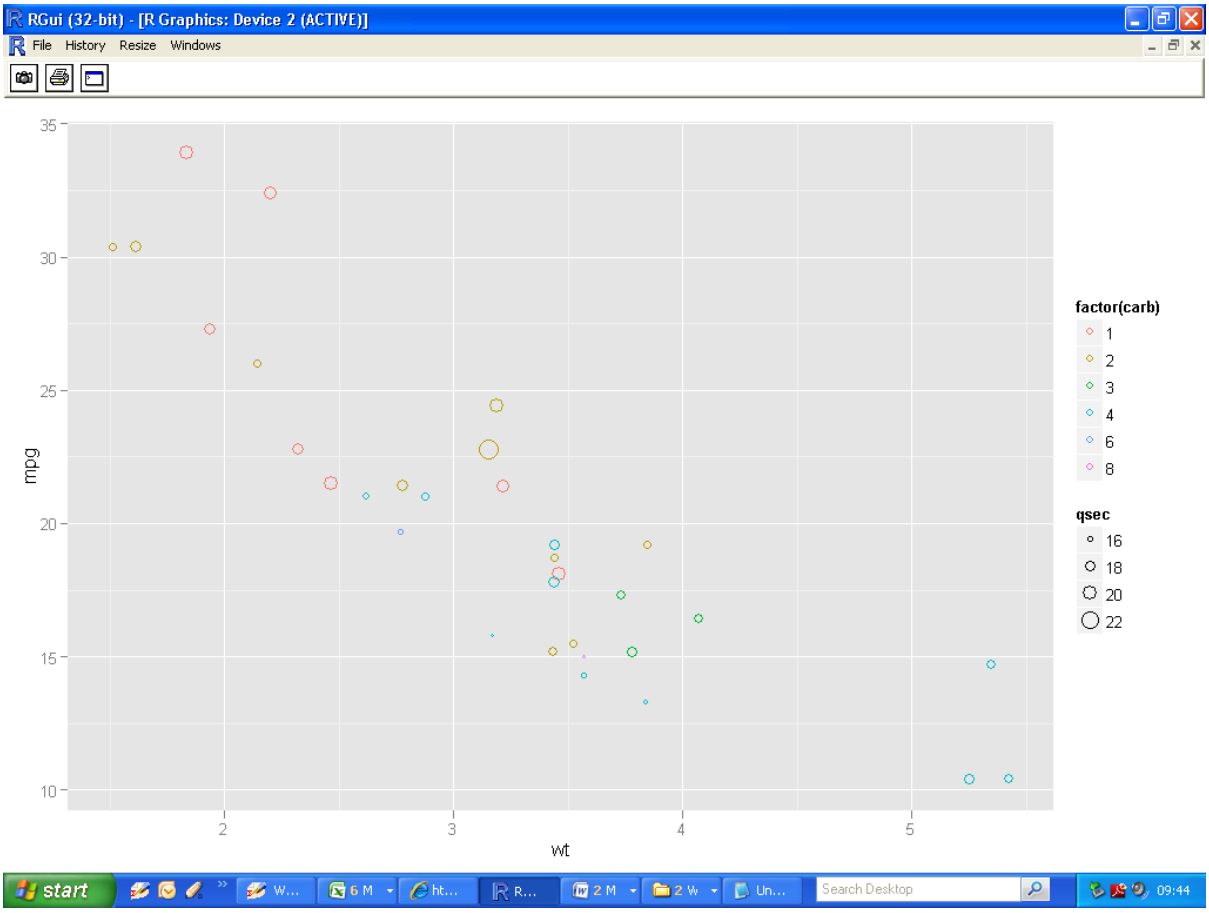
Using different aesthetic mappings : combinations

```
> qplot(wt, mpg, data=mtcars, colour=factor(am),
shape=factor(cyl))
>
```



Using different aesthetic mappings : hollow glyphs

```
> qplot(wt, mpg, data=mtcars, size=qsec,
colour=factor(carb), shape=I(1))
>
```

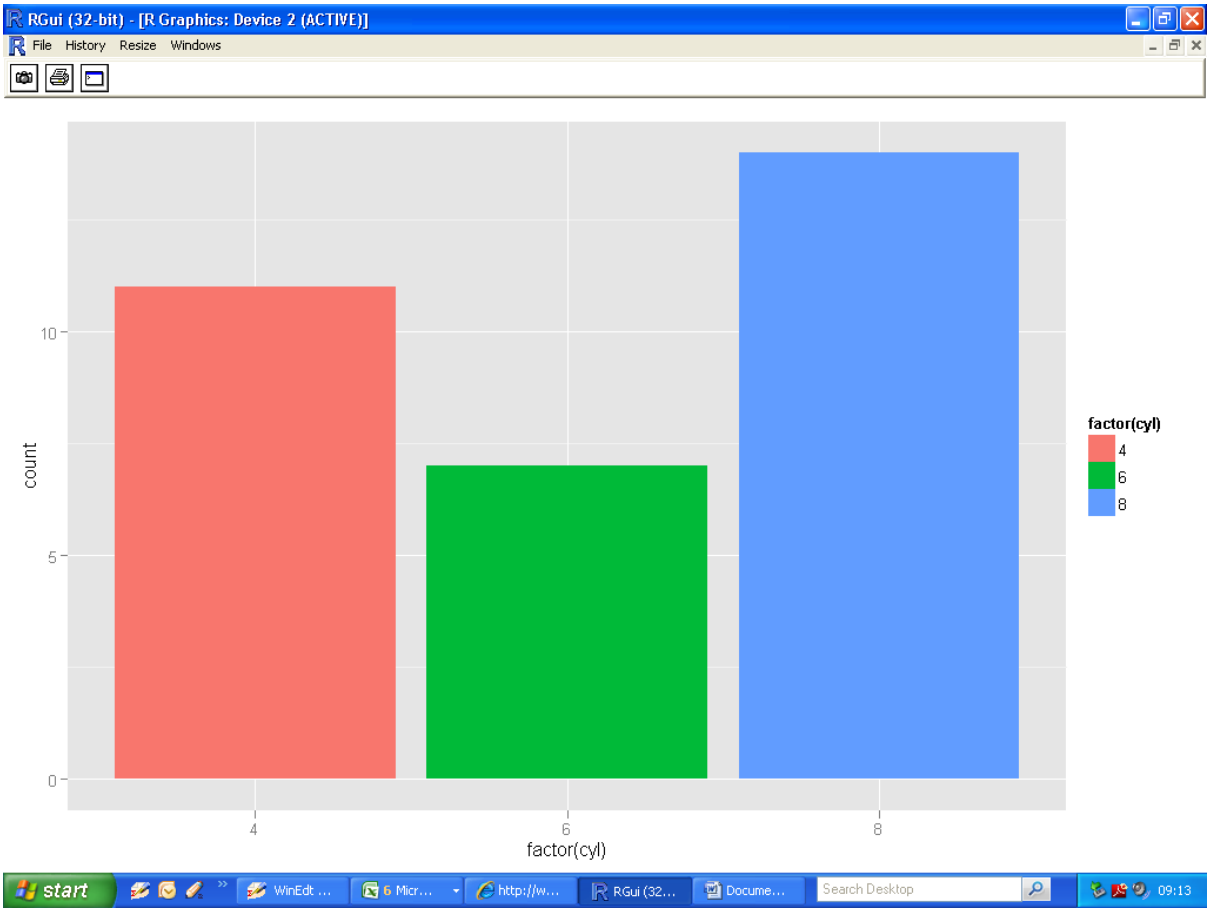


Barplots and Histograms

```
>qplot(factor(cyl), data=mtcars, geom="bar")
```

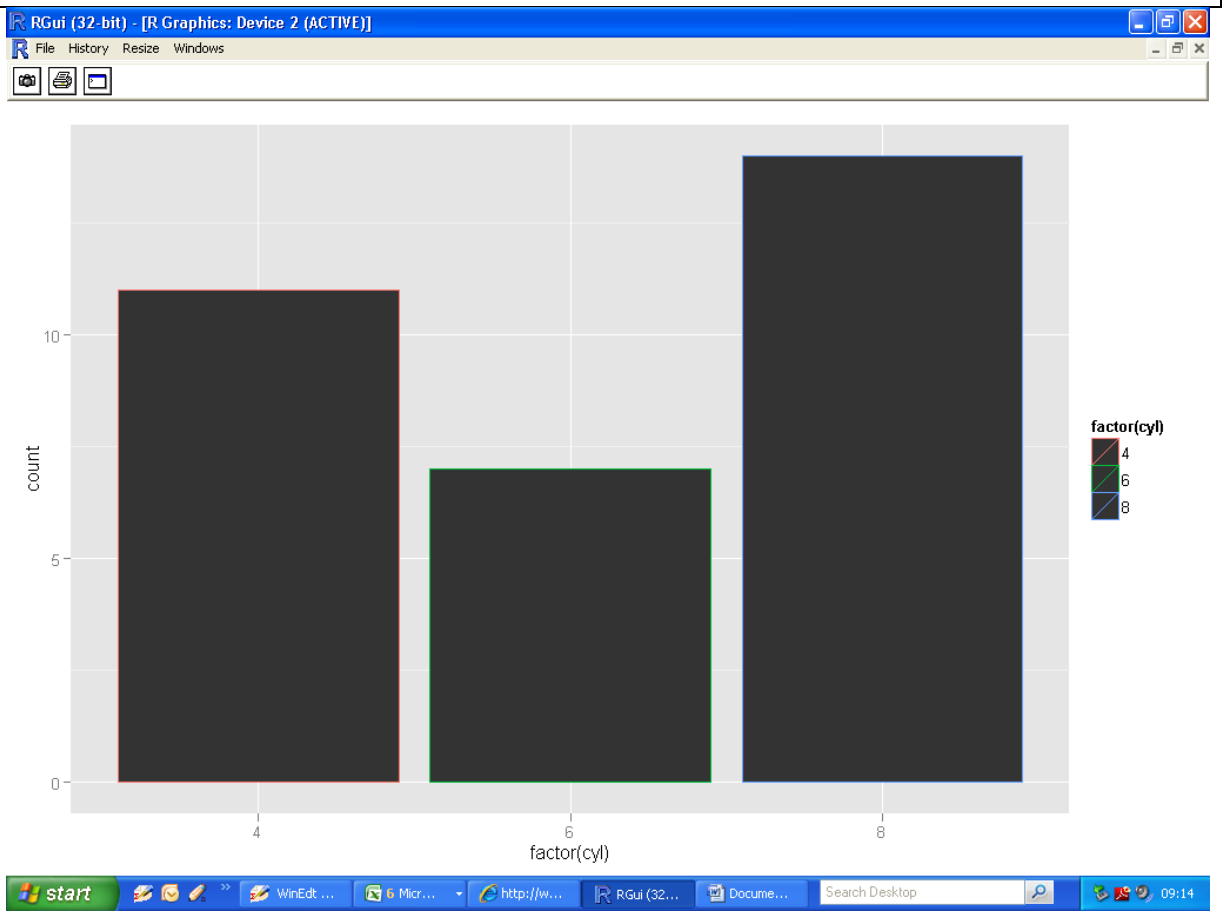
Barplots: Fill argument

```
>qplot(factor(cyl), data=mtcars, geom="bar", fill=factor(cyl))
>
```



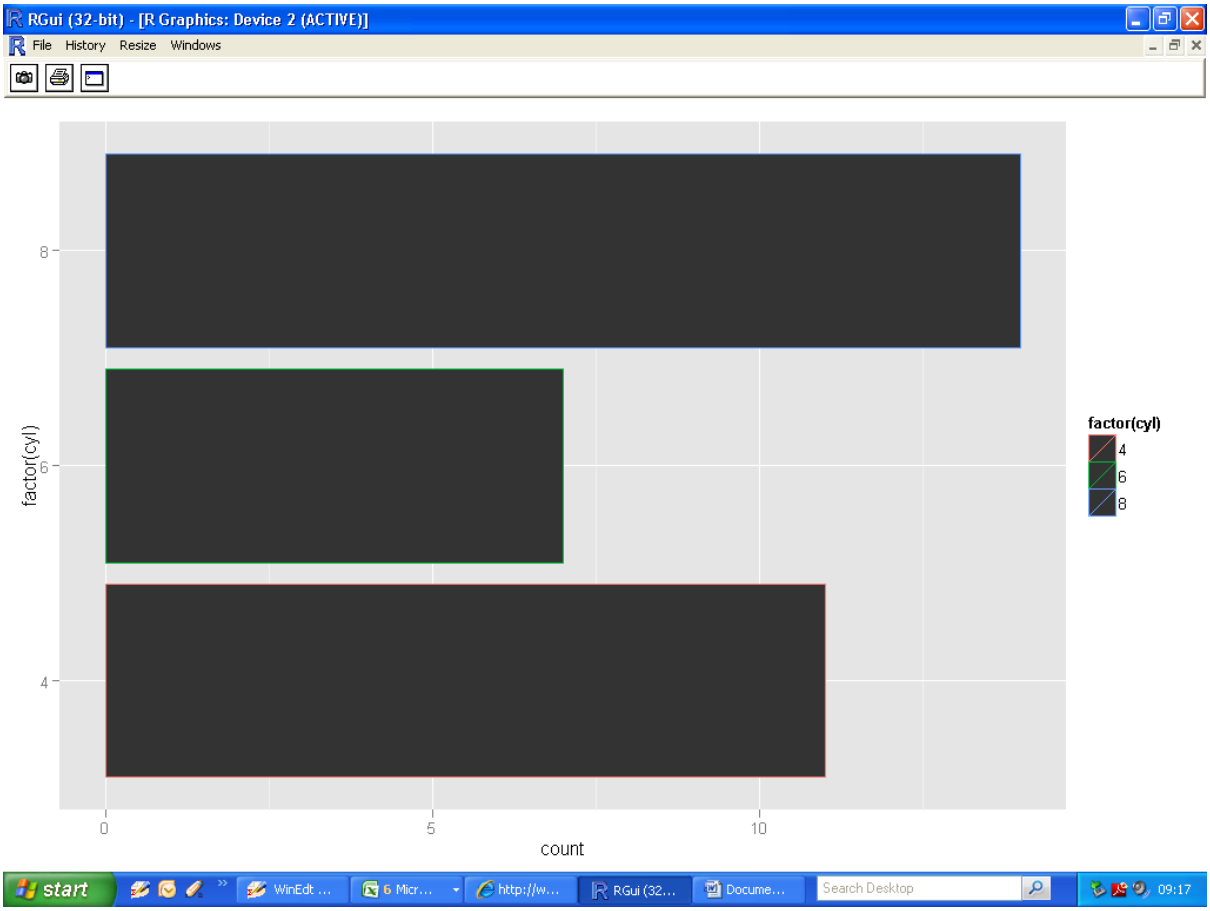
Barplots: Colour argument (border colouring)

```
> qplot(factor(cyl), data=mtcars, geom="bar", colour=factor(cyl))
>
```



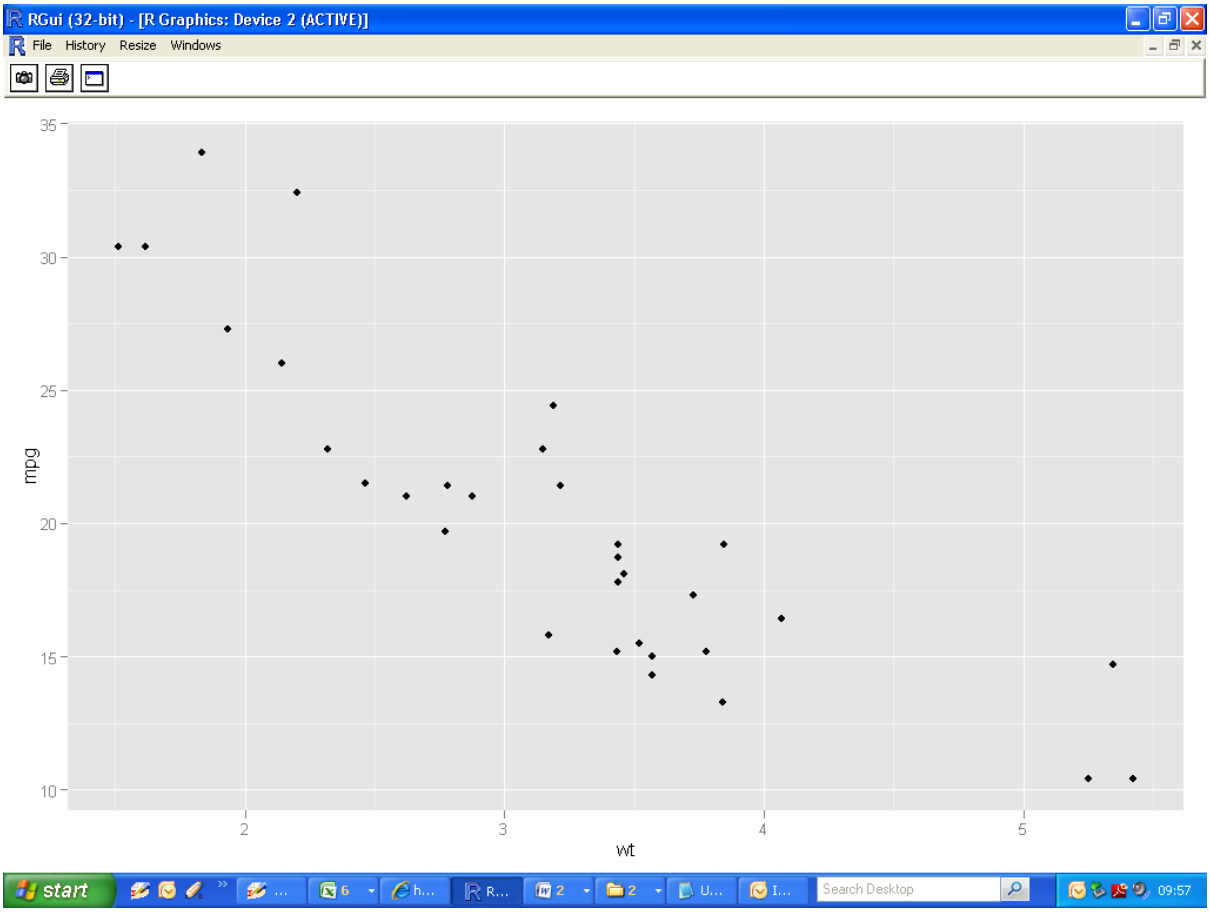
Flipping a plot onto its side

```
> qplot(factor(cyl), data=mtcars, geom="bar" ,colour=factor(cyl))
+ coord_flip()
>
```



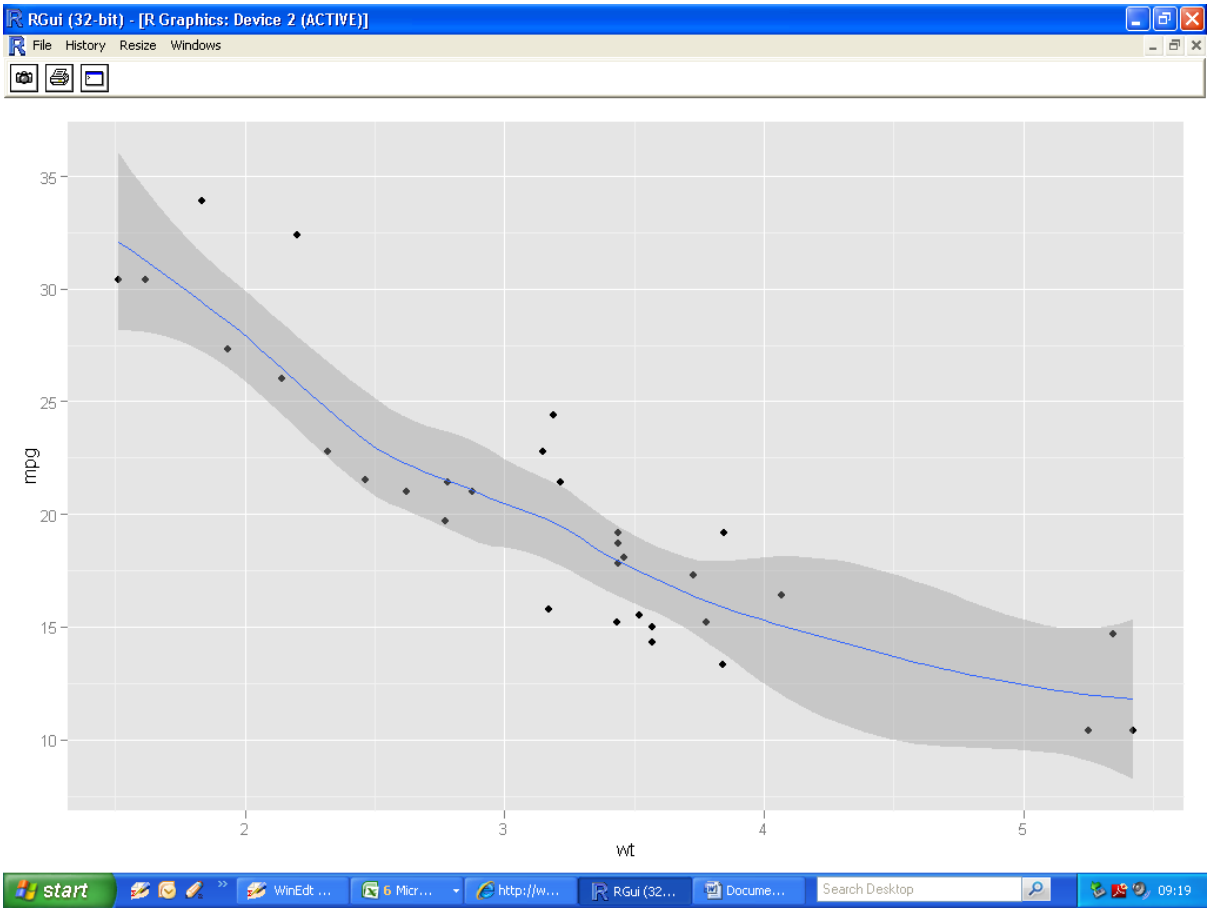
Scatterplots (using Geoms)

```
>qplot(wt, mpg, data=mtcars, geom="point")
>
```



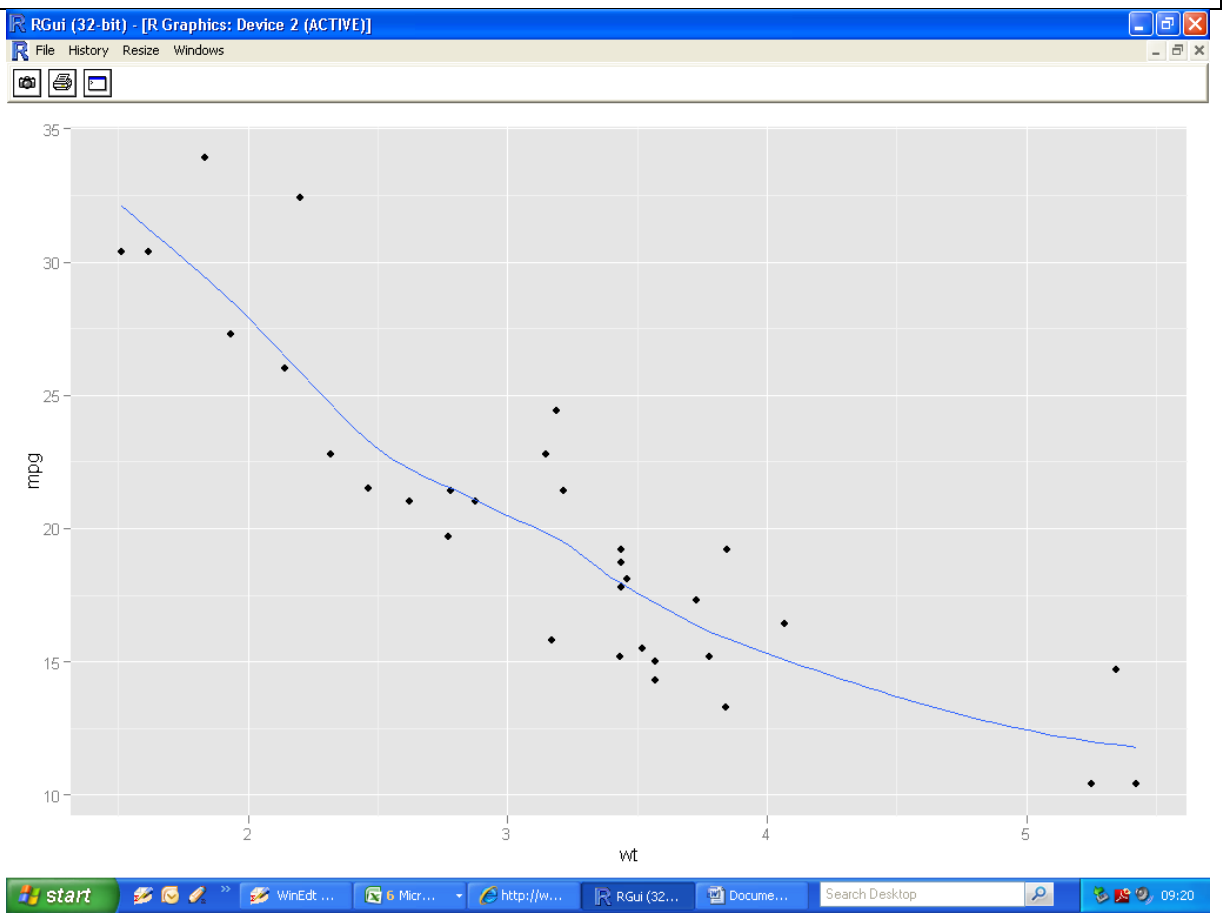
Smoothing on a Scatterplot (loess smoothing)

```
> qplot(wt, mpg, data=mtcars,
  geom=c("point", "smooth"))
>
```



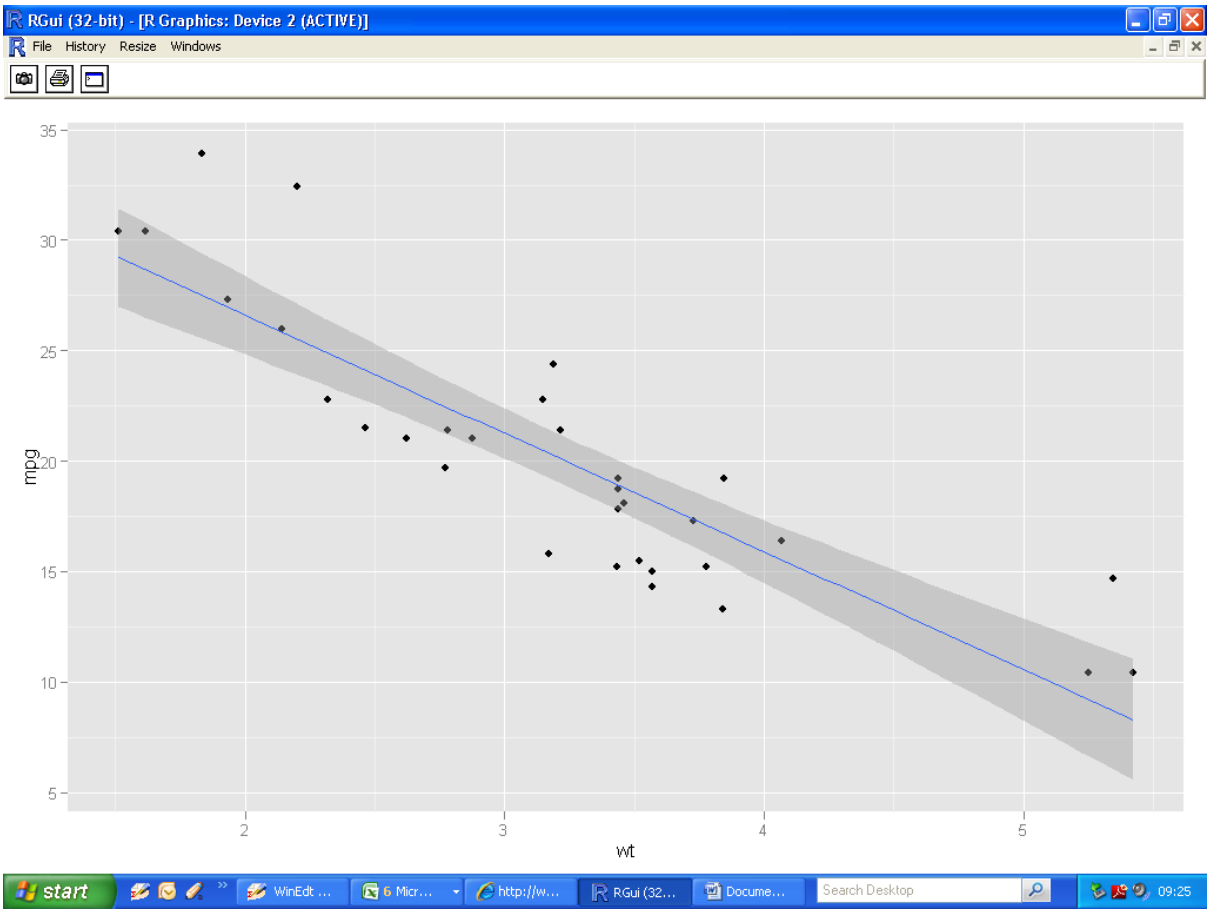
Removing the standard error from the plot

```
> qplot(wt, mpg, data=mtcars,
 geom=c("point", "smooth"),
 se=FALSE)
>
```



Linear Modelling

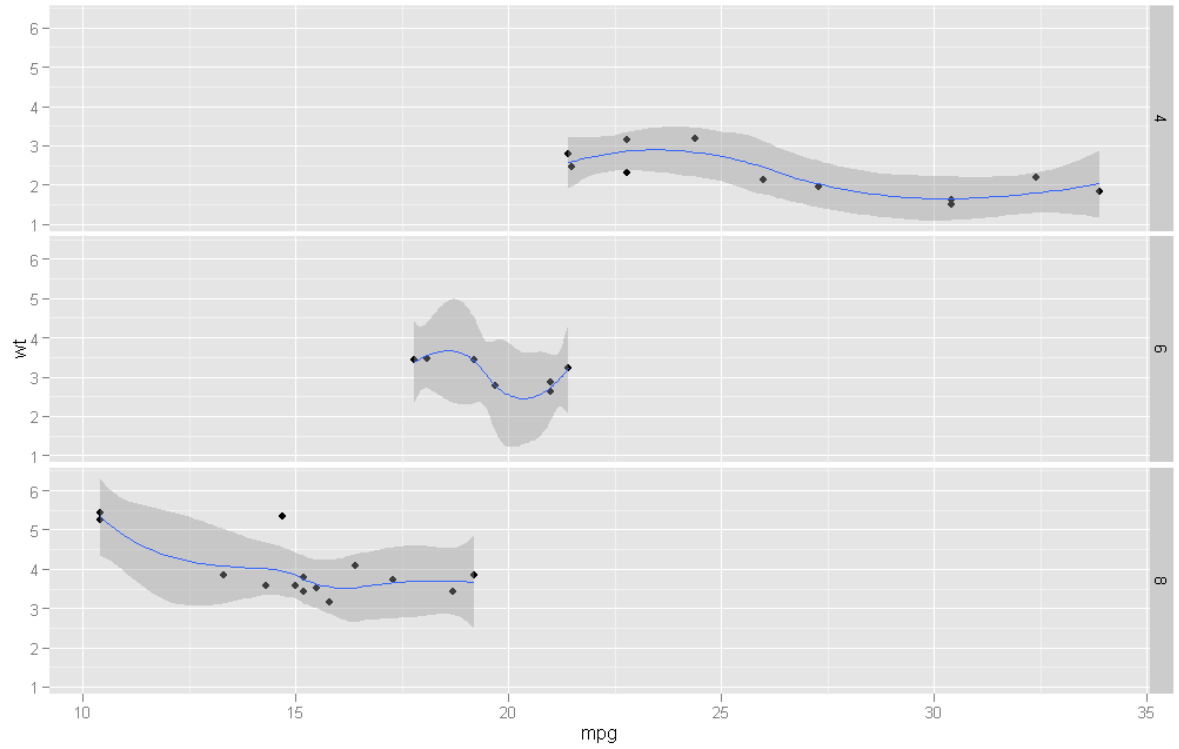
```
> qplot(wt, mpg, data=mtcars,
  geom=c("point", "smooth"),
  method="lm")
>
```



Facetting

- **Split into three subplots for each level of cylinder**

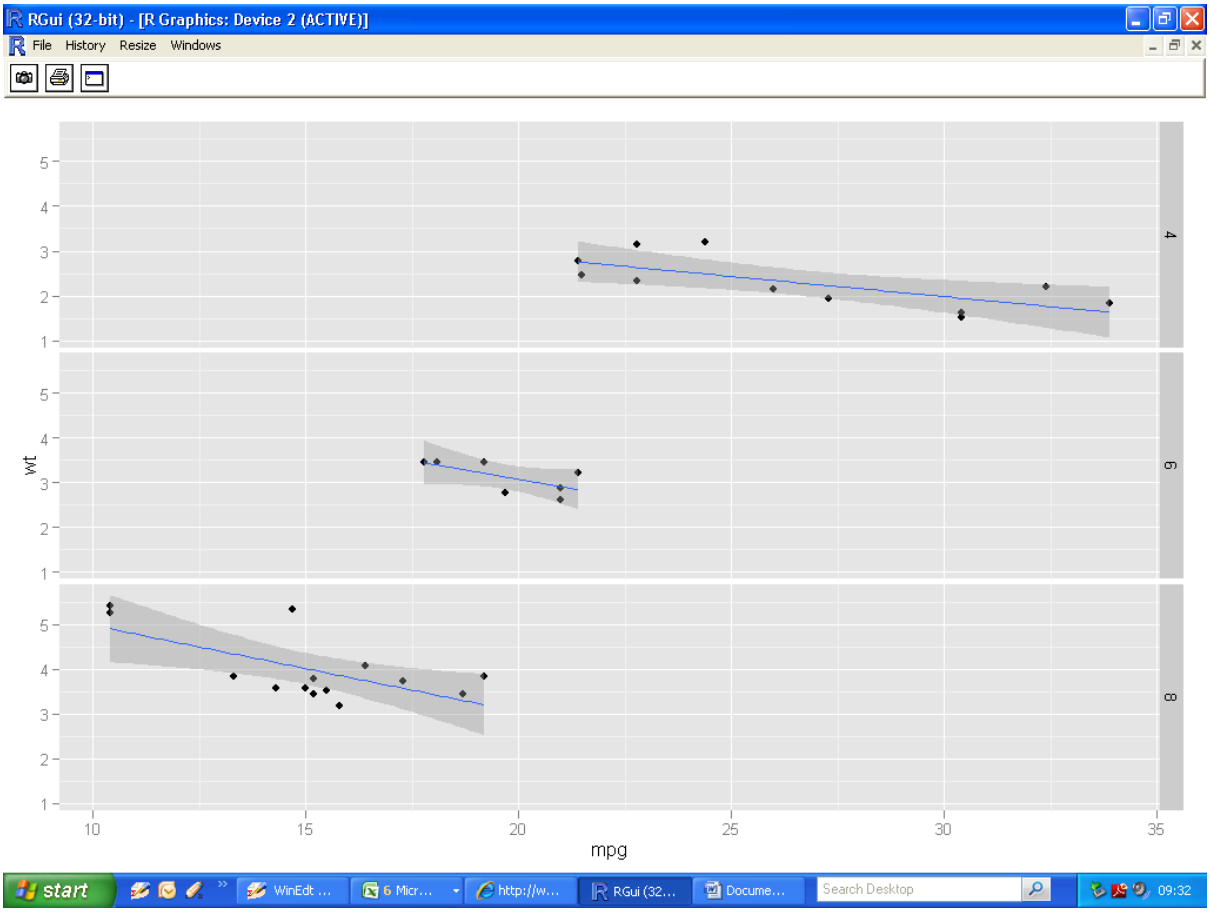
```
> qplot(mpg, wt, data=mtcars, facets= cyl~.,
geom=c("point", "smooth"))
```



- **Three categories of cylinder : 4, 6 and 8**

Facetting (example 2)

```
> qplot(mpg, wt, data=mtcars,
  facets=cyl~.,
  geom=c("point", "smooth"),
  method="lm")
>
```



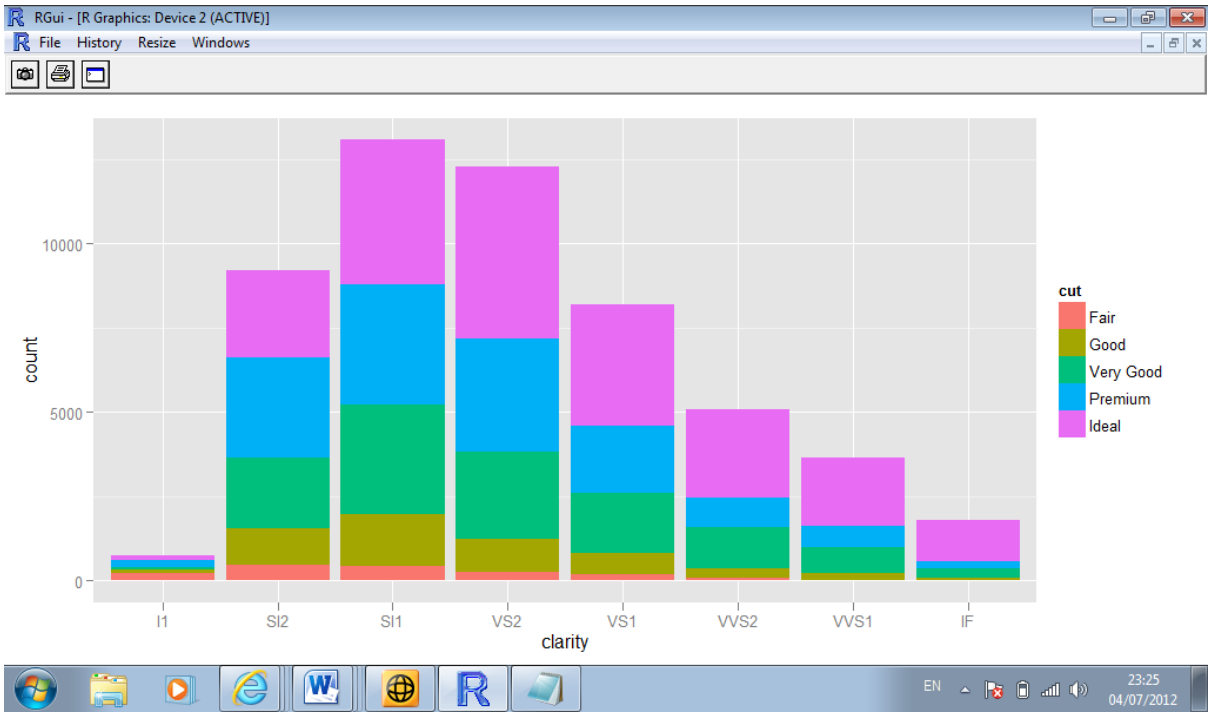
Diamonds data set

```
> head(diamonds)
  carat       cut color clarity depth table price    x    y    z
1  0.23     Ideal     E     SI2  61.5    55   326 3.95 3.98 2.43
2  0.21   Premium     E     SI1  59.8    61   326 3.89 3.84 2.31
3  0.23      Good     E     VS1  56.9    65   327 4.05 4.07 2.31
4  0.29   Premium     I     VS2  62.4    58   334 4.20 4.23 2.63
5  0.31      Good     J     SI2  63.3    58   335 4.34 4.35 2.75
6  0.24 Very Good     J    VVS2  62.8    57   336 3.94 3.96 2.48
```

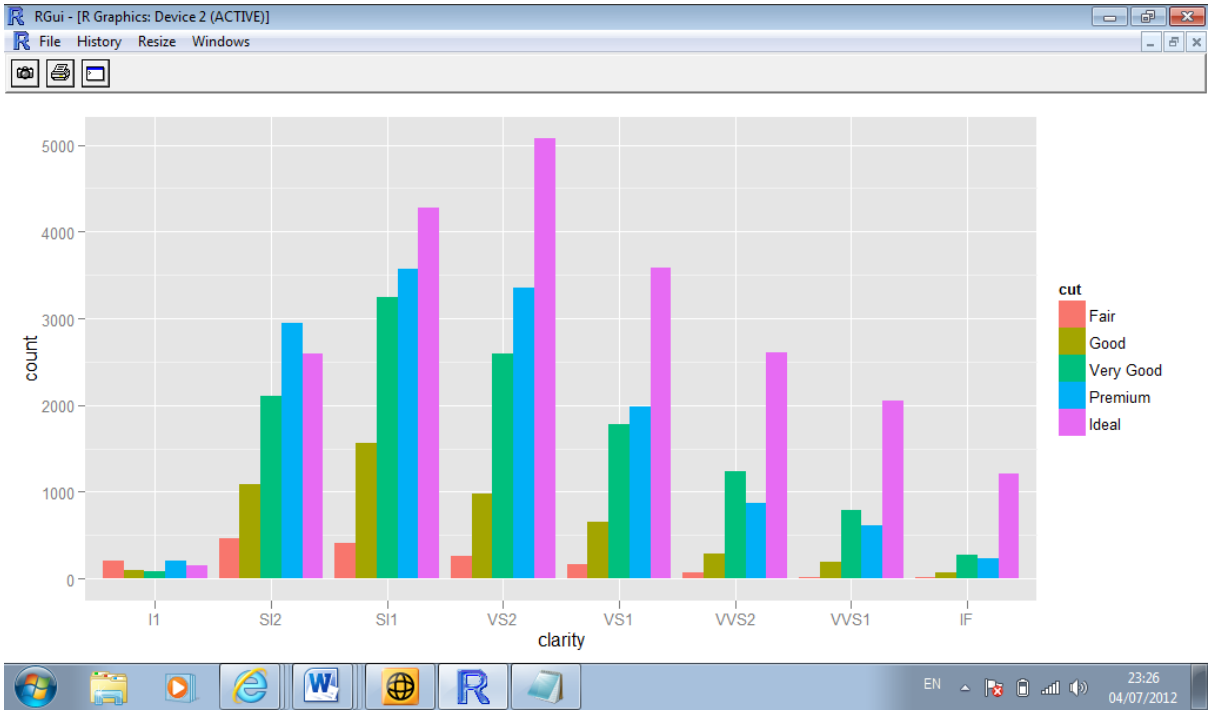
Different display of bar plots

```
qplot(clarity, data=diamonds, geom="bar", fill=cut,
position="stack")

qplot(clarity, data=diamonds, geom="bar", fill=cut,
position="dodge")

qplot(clarity, data=diamonds, geom="bar", fill=cut,
position="fill")

qplot(clarity, data=diamonds, geom="bar", fill=cut,
position="identity")
```

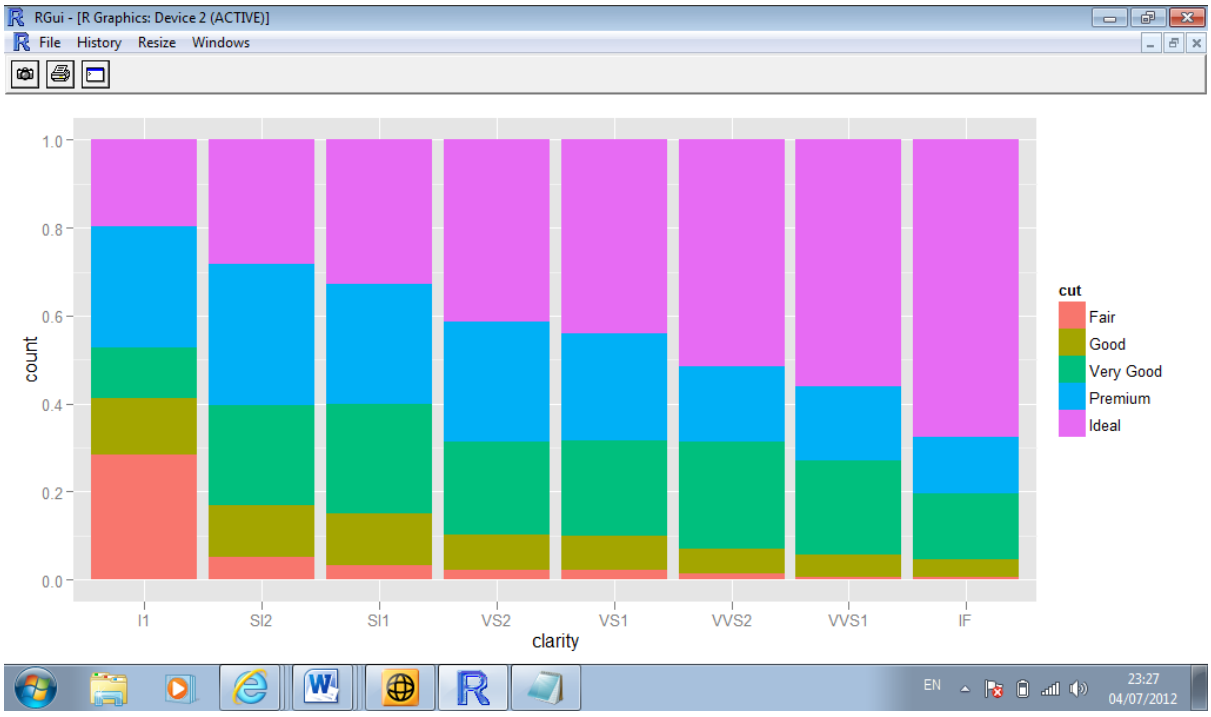
1. Stack



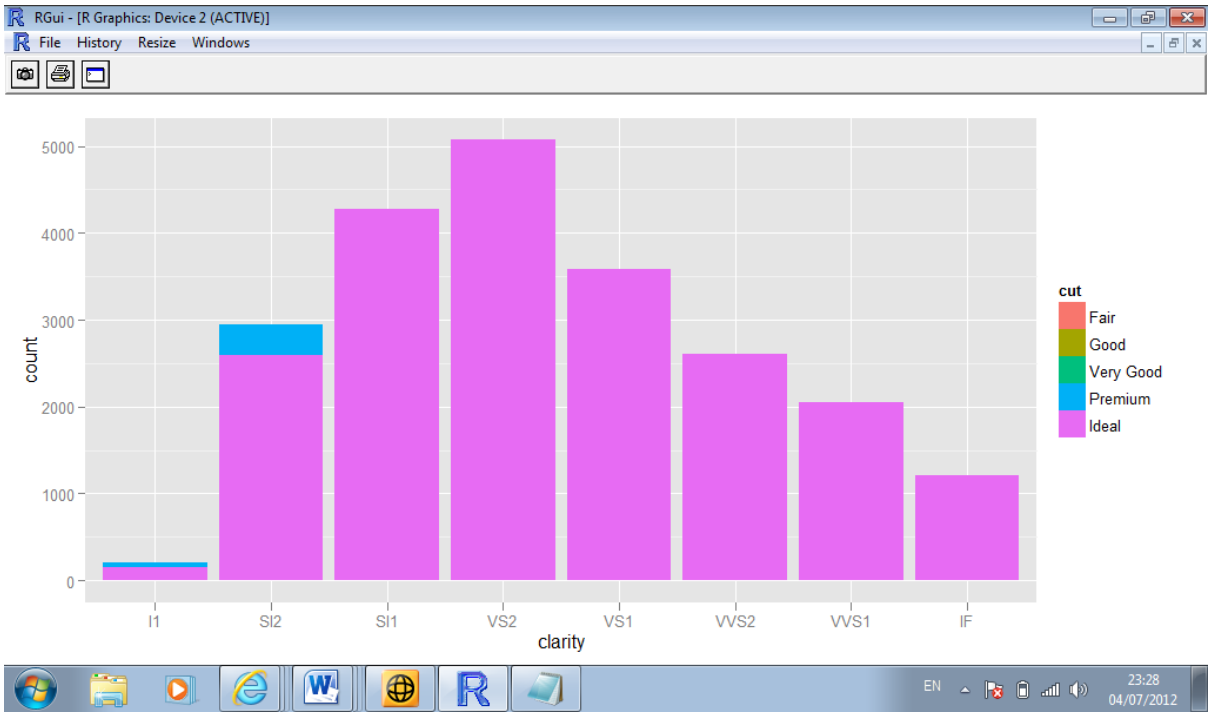
2. Dodge



3. Fill



4. Identity

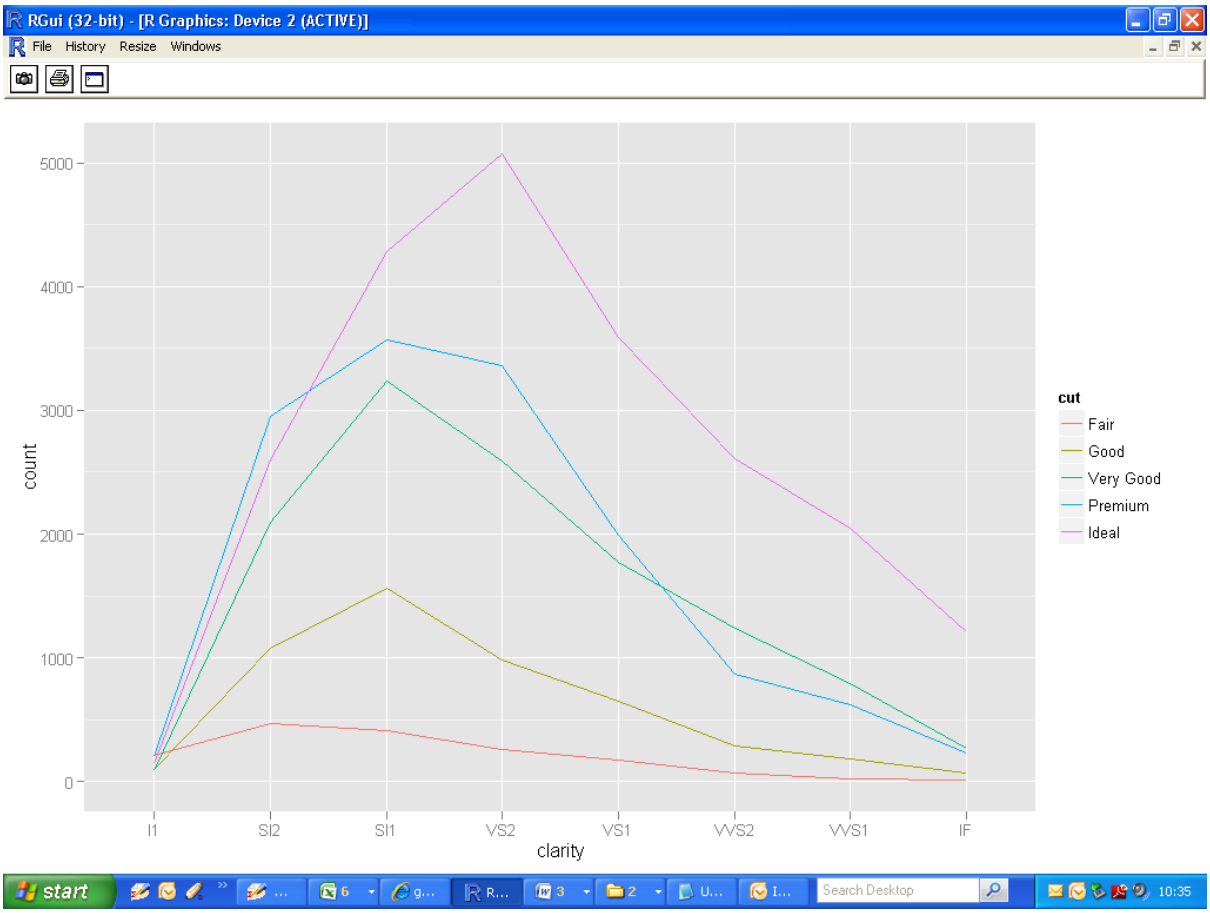

RGui - [R Graphics: Device 2 (ACTIVE)]
File History Resize Windows
5000
4000
3000
2000
1000
0
count
I1
SI2
SI1
VS2
VS1
VVS2
VVS1
IF
clarity
cut
Fair
Good
Very Good
Premium
Ideal
EN
23:28
04/07/2012

Frequency Polygons

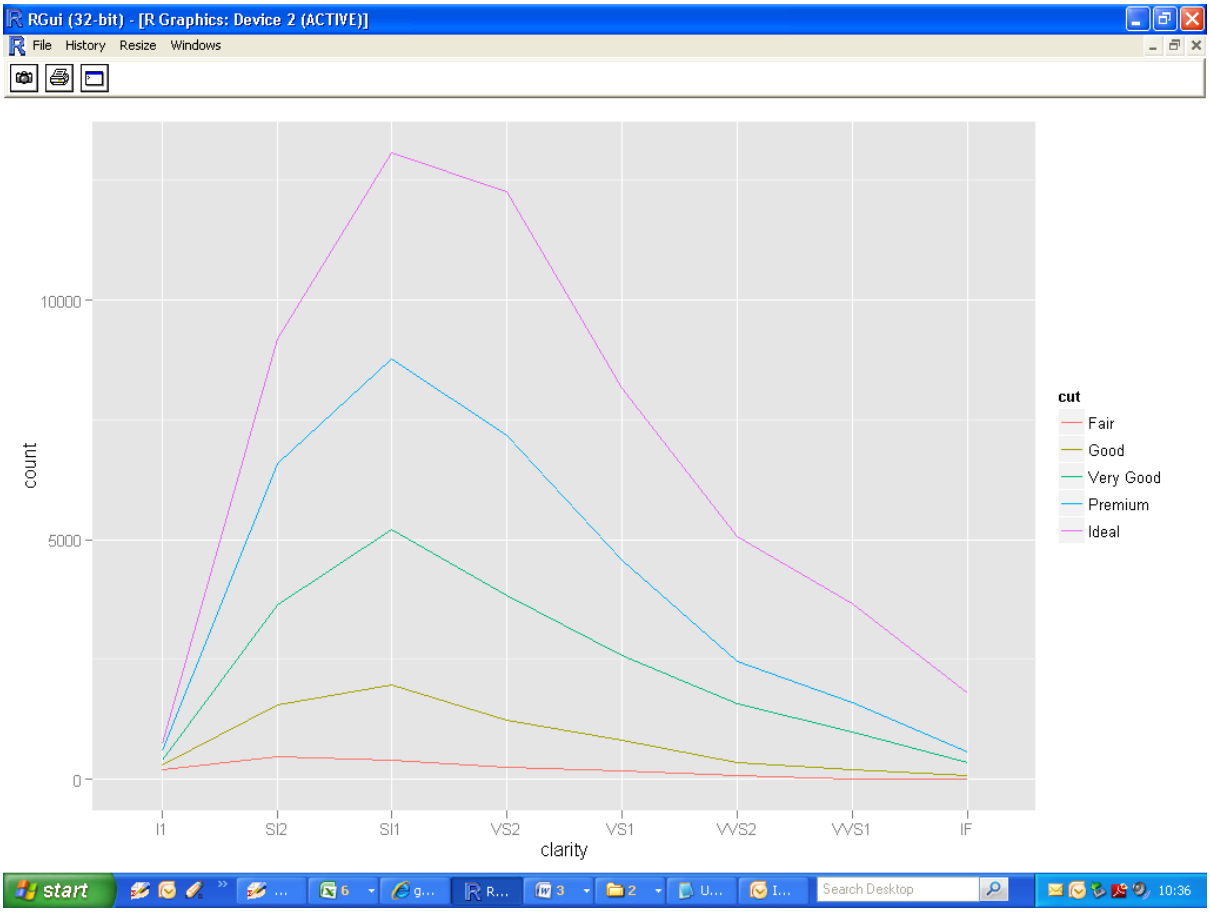
```
qplot(clarity, data=diamonds, geom="freqpoly",
group=cut, colour=cut, position="identity")

qplot(clarity, data=diamonds, geom="freqpoly",
group=cut, colour=cut, position="stack")
```

Contingency table

```
> table(diamonds$cut,diamonds$clarity)

              I1   SI2   SI1   VS2   VS1 VVS2 VVS1    IF
  Fair       210   466   408   261   170   69   17     9
  Good        96  1081  1560   978   648  286  186    71
  Very Good   84  2100  3240  2591  1775 1235  789   268
  Premium    205  2949  3575  3357  1989  870  616   230
  Ideal      146  2598  4282  5071  3589 2606 2047  1212
```

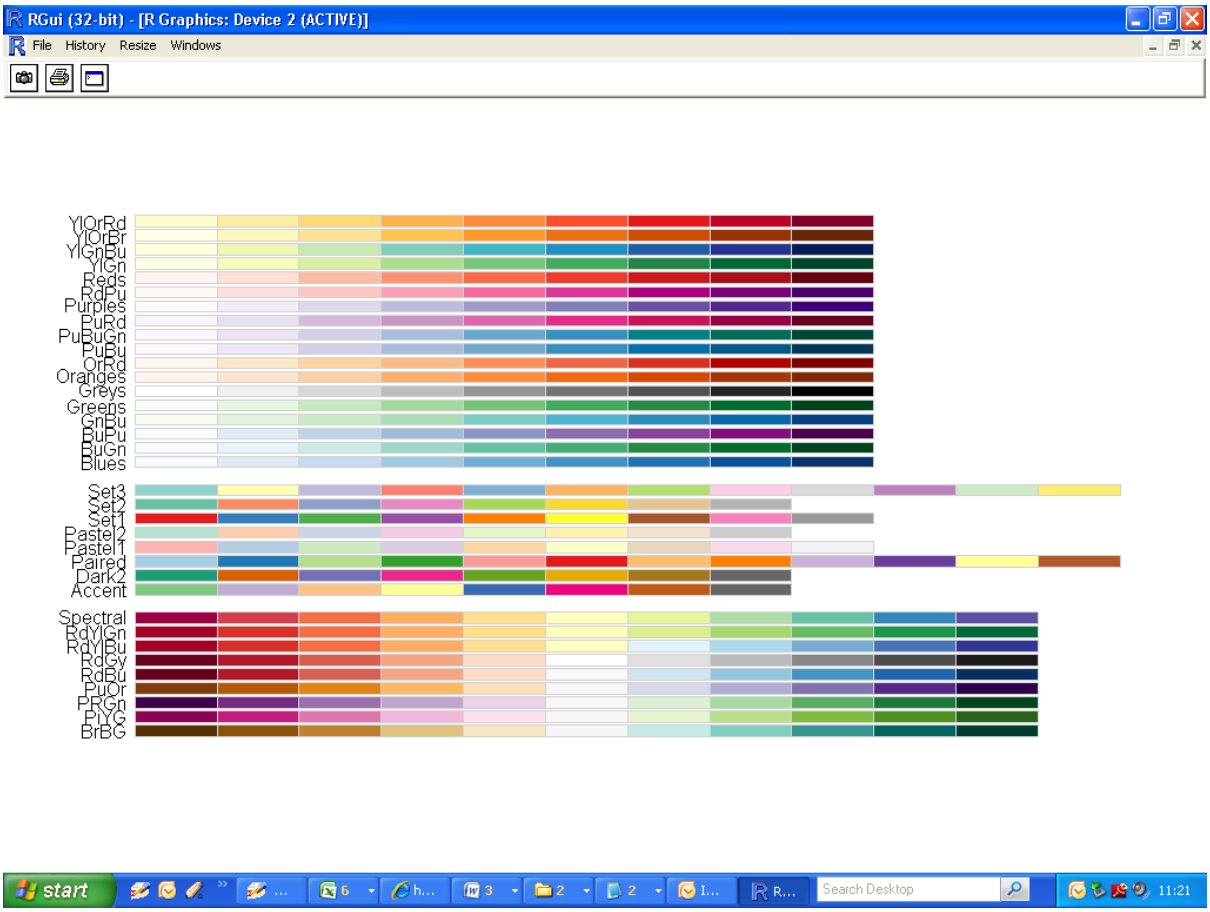

RGui (32-bit) - [R Graphics: Device 2 (ACTIVE)]
File History Resize Windows
5000
4000
3000
2000
1000
0
count
I1
SI2
SI1
VS2
VS1
VVS2
VVS1
IF
clarity
cut
Fair
Good
Very Good
Premium
Ideal
start
Search Desktop
10:35

Dublin R : ggplot2 Workshop (5th July 2012)



Palettes

```
>RColorBrewer::display.brewer.all()
```



```
>QPlot + scale_fill_brewer(palette="Oranges")
```