Part C : Quickplot

- The quickplot function (“ `qplot()` ”) function is supposed to make the same sort of plots as `ggplot()`, the main function offered by the `ggplot2` package, but with simpler syntax.
    - We will look at `ggplot()` later on.
- However, in practice, for more complex plots, it is probably more straightforward to use `ggplot()`.
- Many of the arguments supplied to R’s plotting command `plot()` (e.g. main, xlab, ylab etc) can be supplied to `qplot()`, and processed the same way.

- Motor Trend Car Road Tests

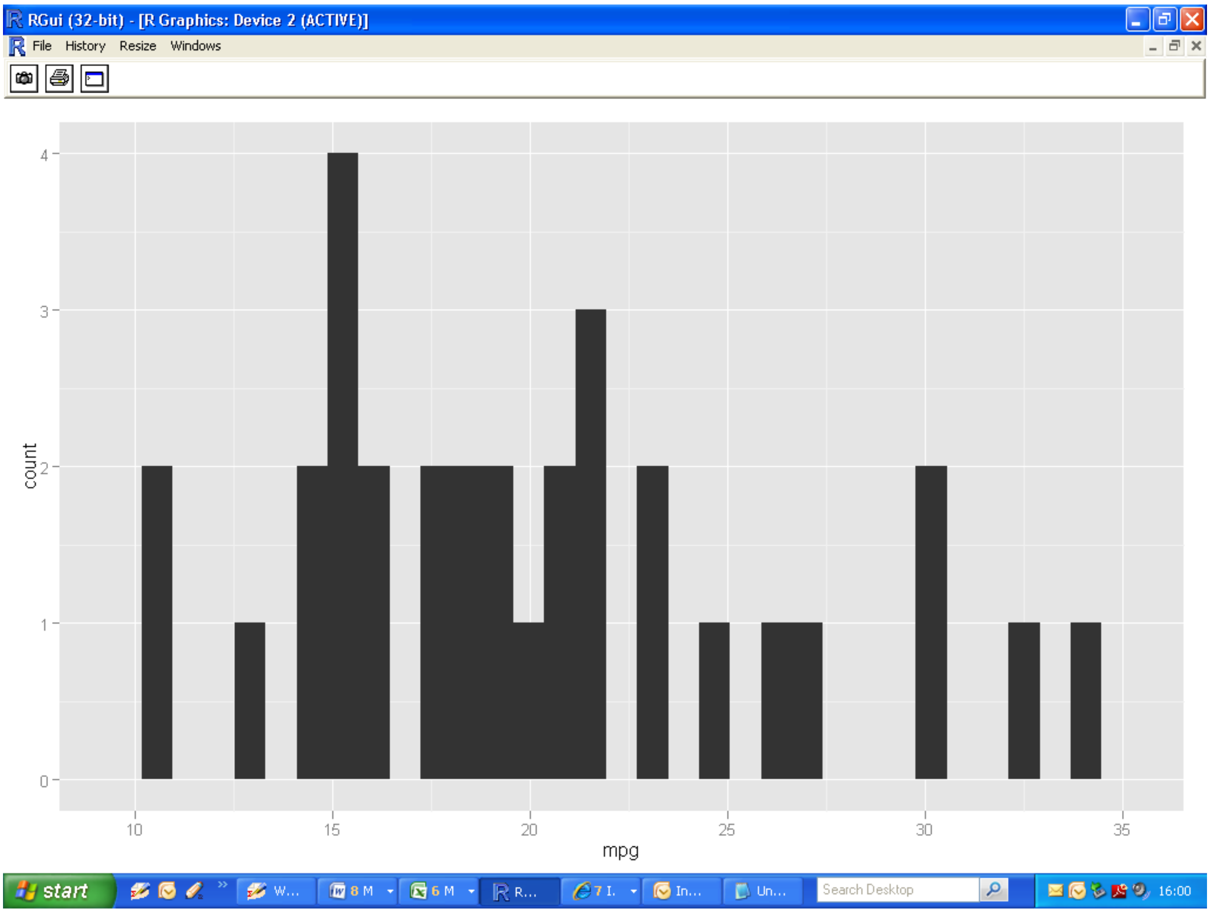
This data set was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973–74 models).

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

Simple histogram using quickplot

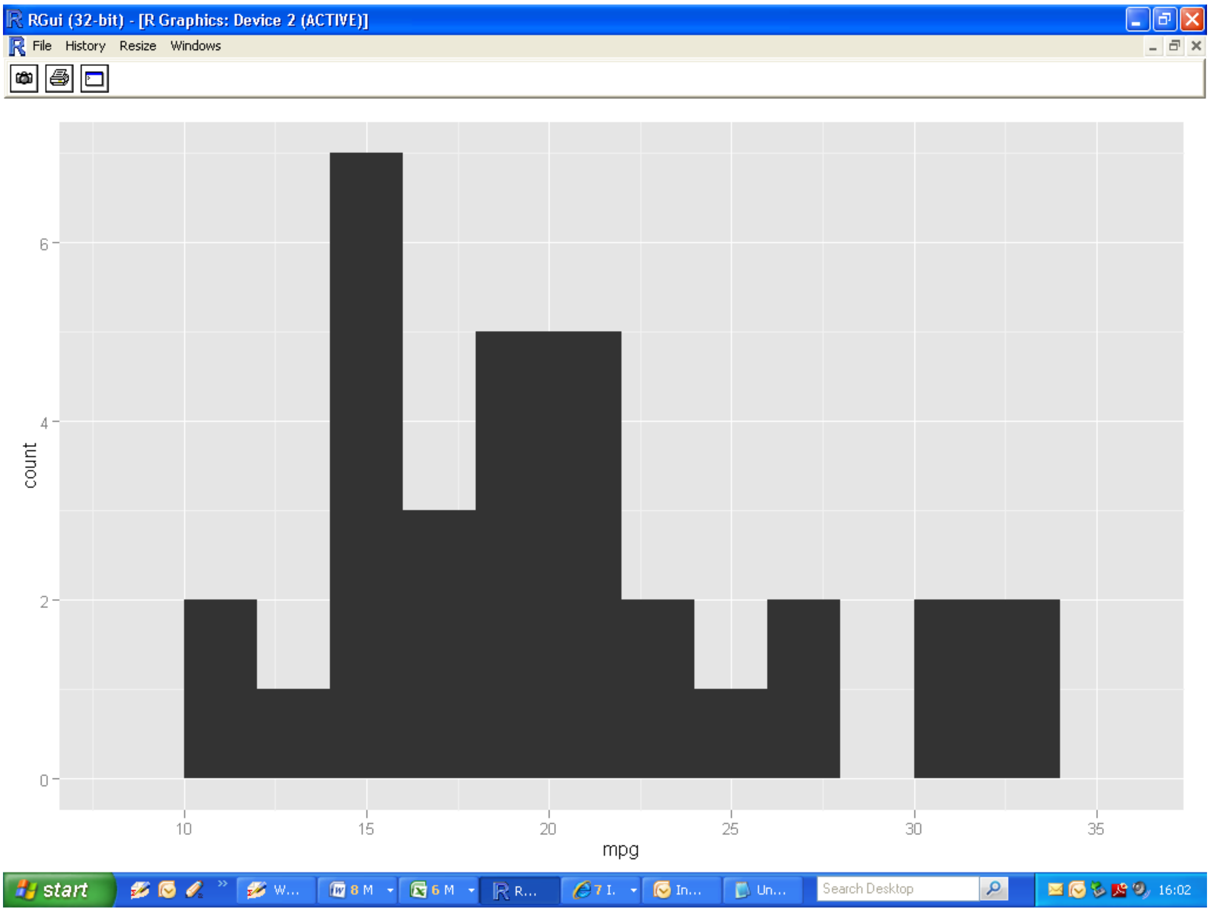
The basic syntax of the command is

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


RGui (32-bit) - [R Graphics: Device 2 (ACTIVE)]
File History Resize Windows
count
0
1
2
3
4
10
15
20
25
30
35
mpg
start
Search Desktop
16:00

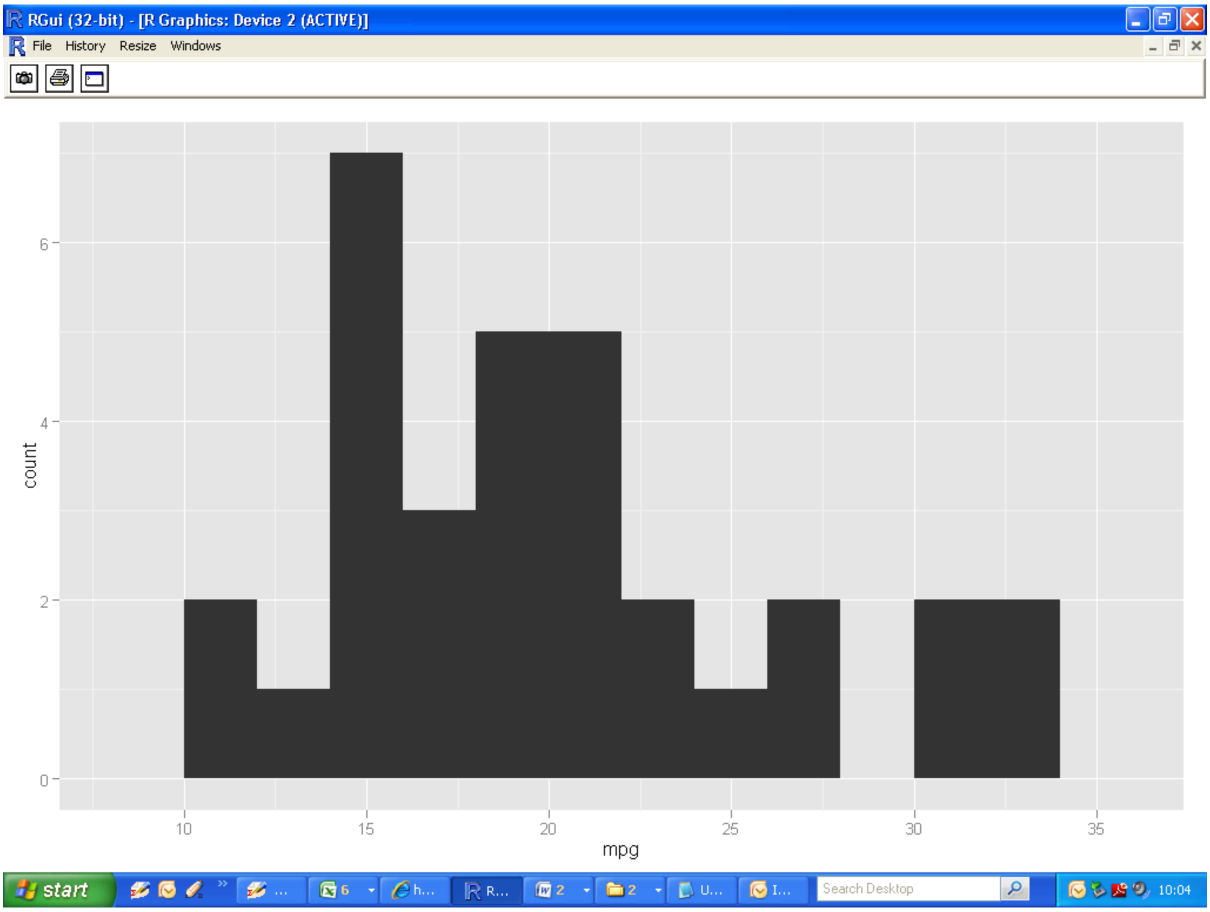
Adjusting bin-width

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



Histogram geom

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


RGui (32-bit) - [R Graphics: Device 2 (ACTIVE)]
File History Resize Windows
count
0
2
4
6
10
15
20
25
30
35
mpg
start
Search Desktop
10:04

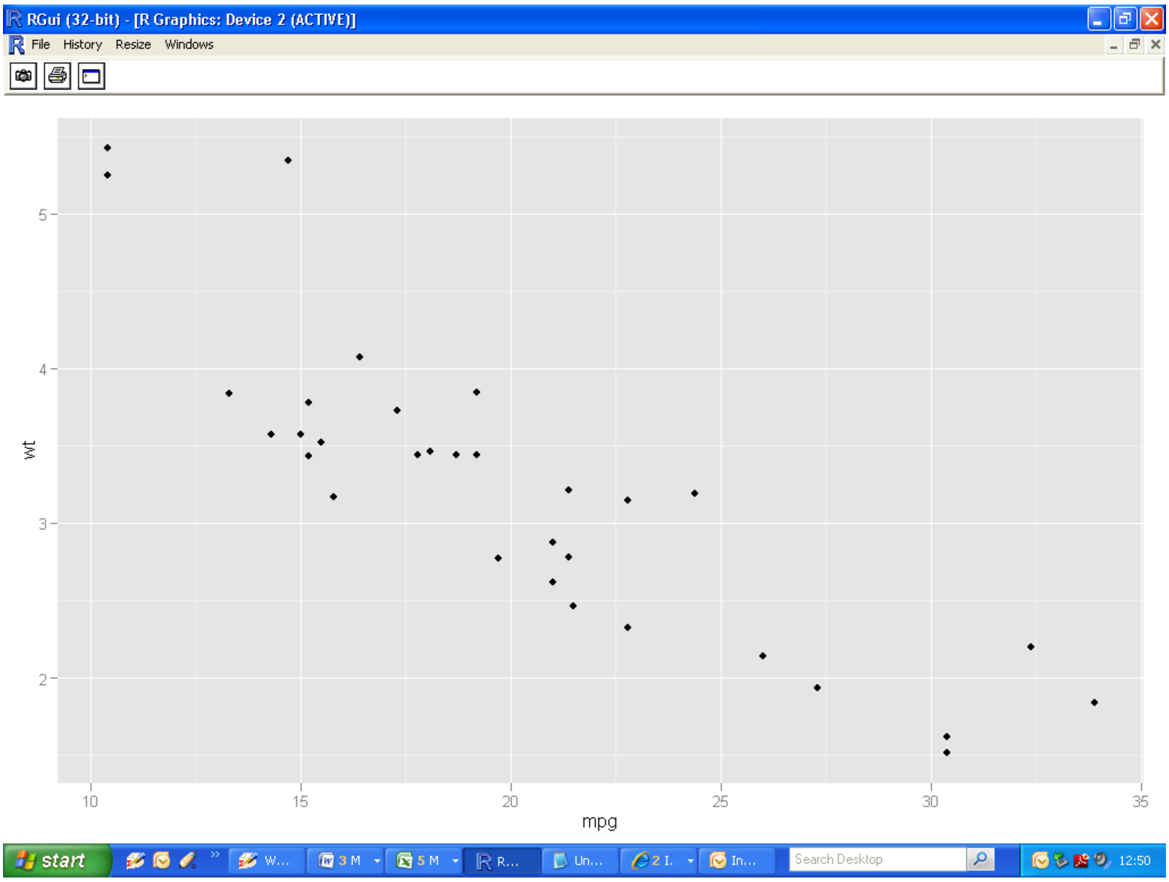
Simple scatterplot using quickplot

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)
>
```


RGui (32-bit) - [R Graphics: Device 2 (ACTIVE)]
File History Resize Windows
wt
5
4
3
2
10
15
20
25
30
35
mpg
start
Search Desktop
12:50

Immediately noticeable:

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

Subsetting

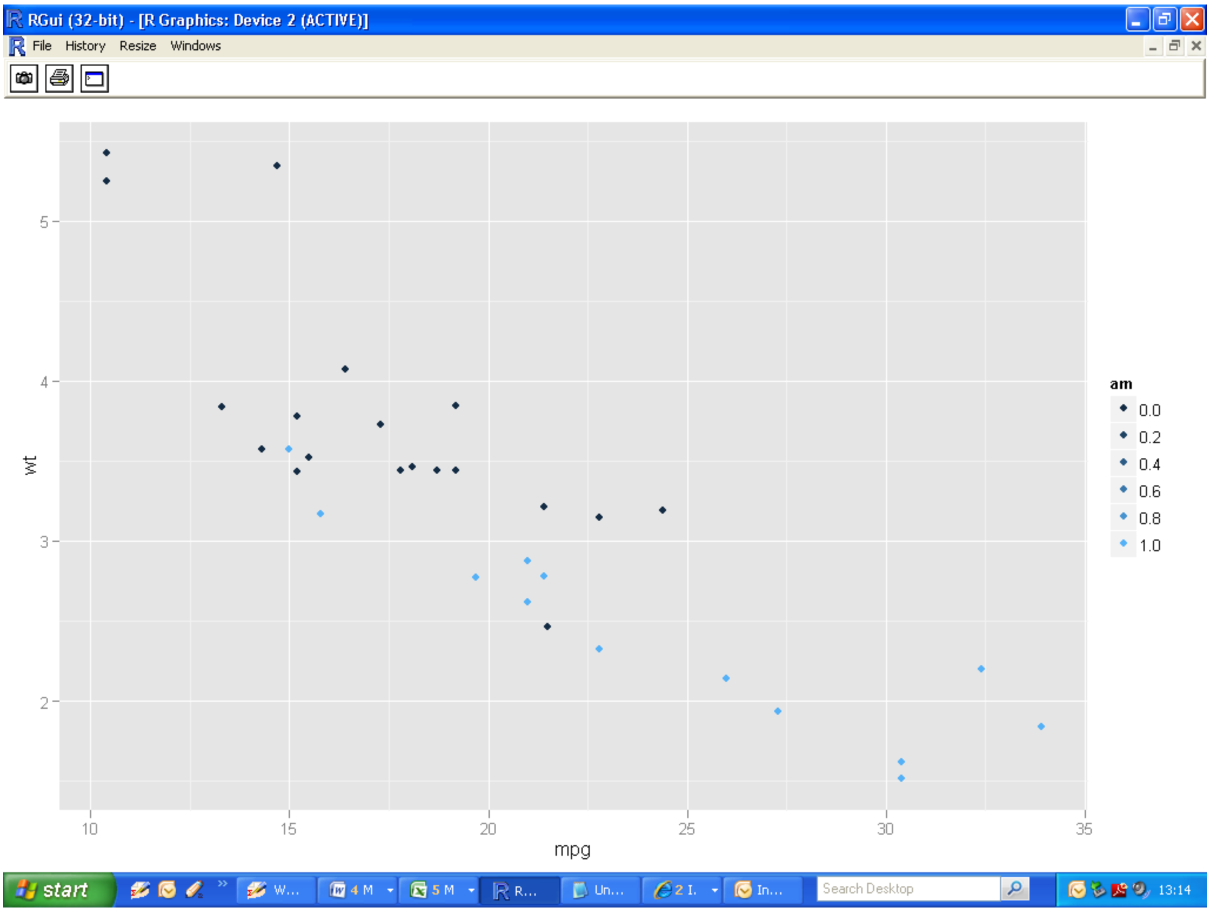
Those familiar with the `mtcars` dataset would be aware of other (categorical) variables. For example

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

First we will subset by colour, using the ***am*** variable

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

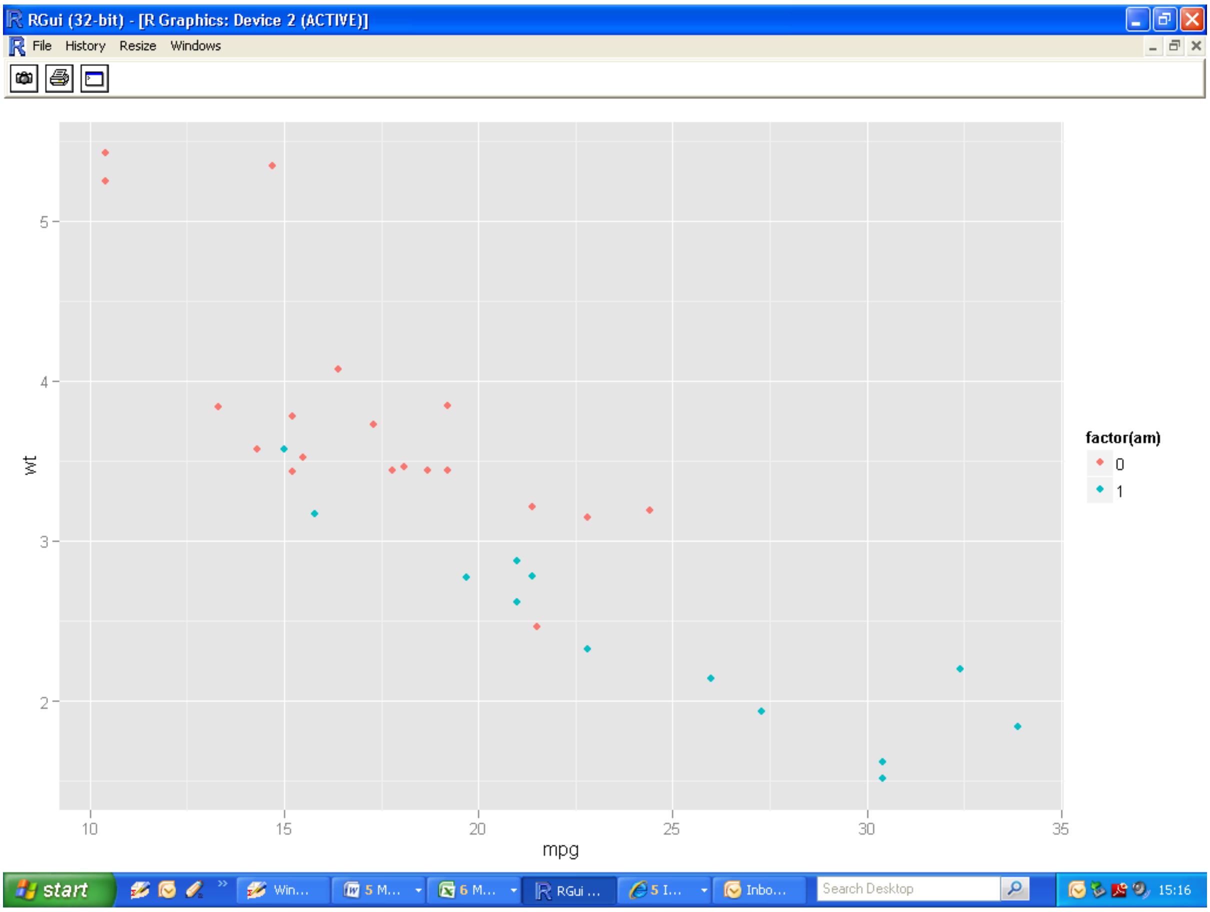
We can now use the plot to make some deductions about whether or not being automatic or manual has a bearing.



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

- However, notice the legend – we have 6 “levels” for the variable ***am*** (which is in fact binary, so it makes no sense).
- We need to fix this; use the function `factor()`.

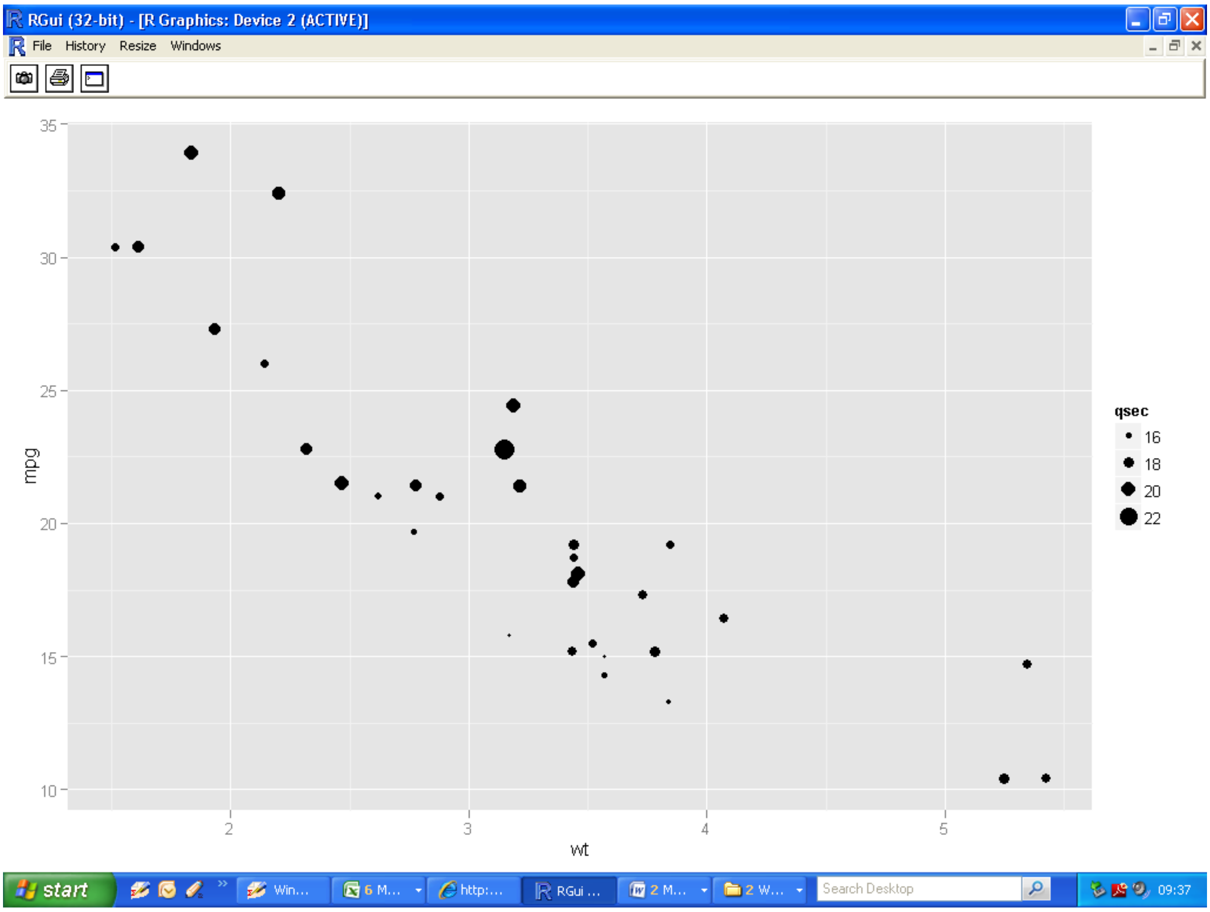
```
>
>qplot(mpg,wt,data=mtcars, colour=factor(am))
>
```



Using different aesthetic mappings - size

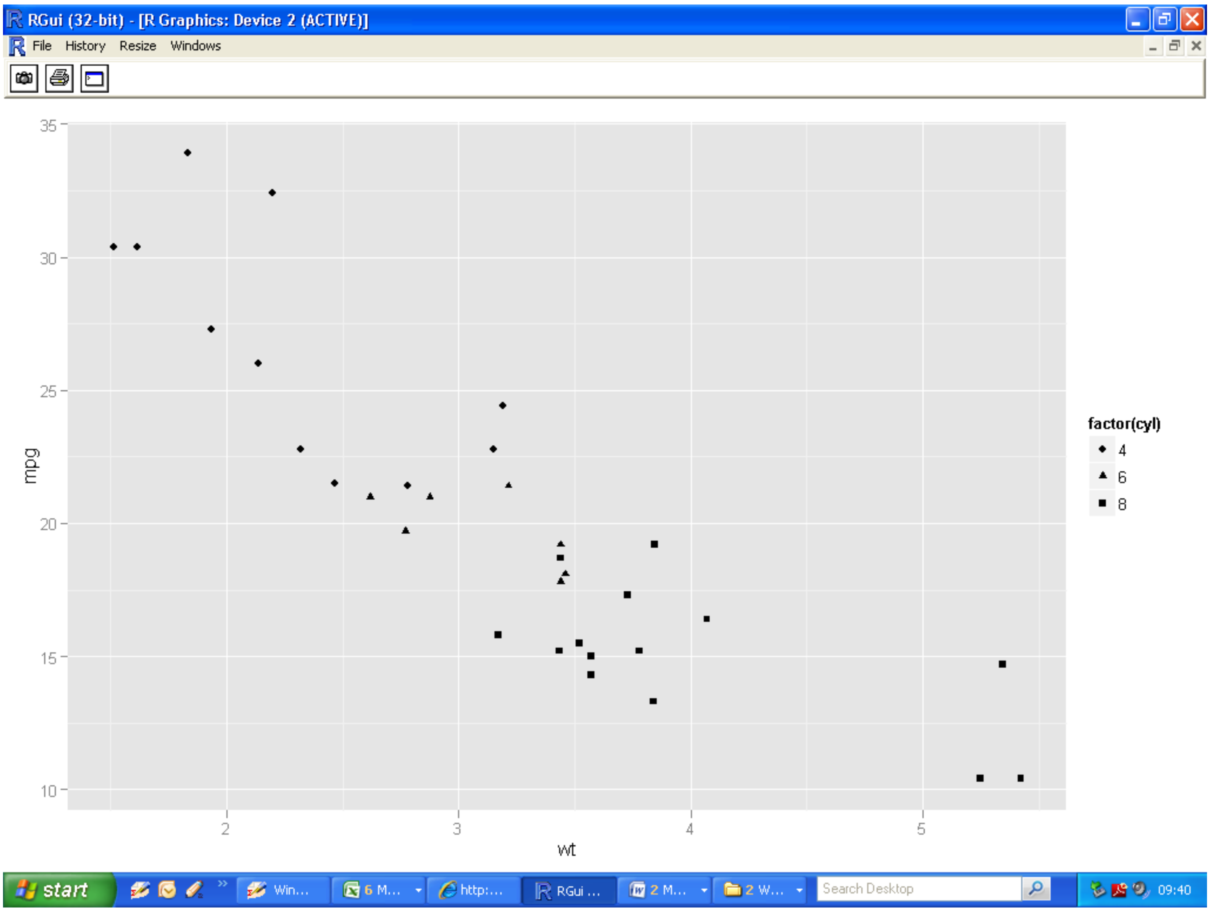
We can scale the glyph in terms of size, depending on which level of a variable they belong to. We will use a different variable this time.

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



Using different aesthetic mappings - shape

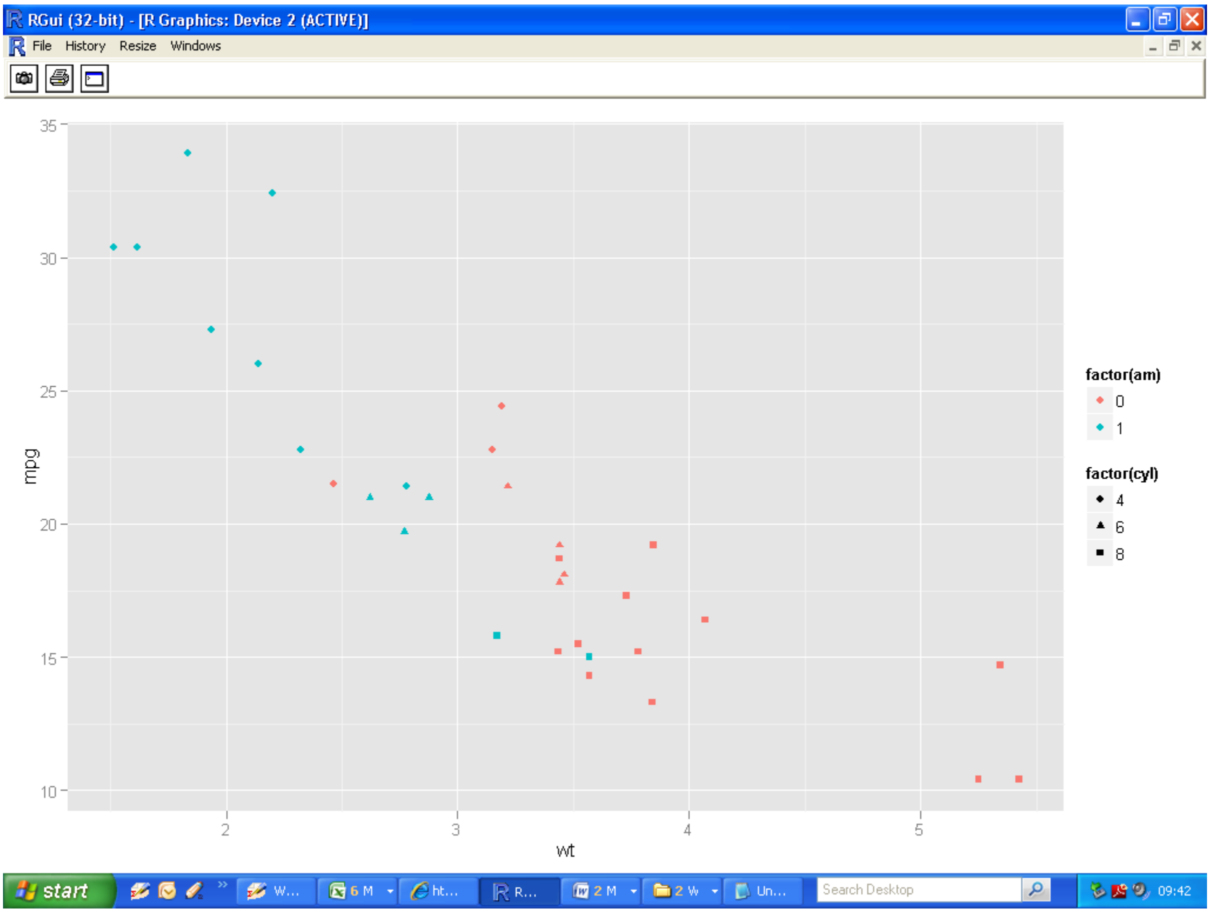
```
> qplot(wt, mpg, data=mtcars, shape=factor(cyl))
>
```



Using different aesthetic mappings : combinations

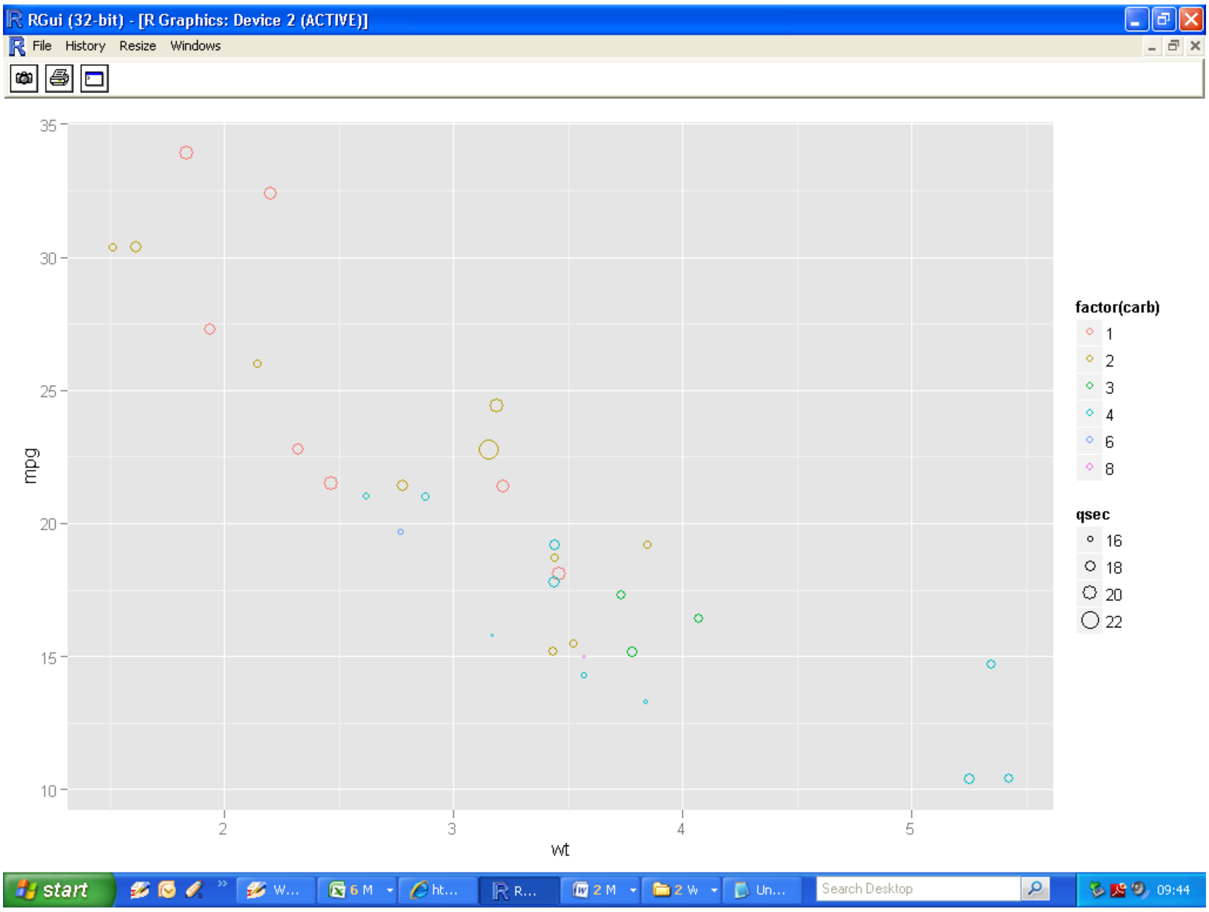
```
> qplot(wt, mpg, data=mtcars, colour=factor(am),
shape=factor(cyl))
>
```

This particular combination is arguably overly complicated, but it is good to know that plots can be enhanced in such a way.


RGui (32-bit) - [R Graphics: Device 2 (ACTIVE)]
File History Resize Windows
35
30
25
20
15
10
mpg
2
3
4
5
wt
factor(am)
0
1
factor(cyl)
4
6
8
start
Search Desktop
09:42

Using different aesthetic mappings : hollow glyphs

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


RGui (32-bit) - [R Graphics: Device 2 (ACTIVE)]
File History Resize Windows
35
30
25
20
15
10
mpg
2
3
4
5
wt
factor(carb)
1
2
3
4
6
8
qsec
16
18
20
22
start
Search Desktop
09:44