Case Study 1: Visualising the data

Richard Wilkinson

This document is created using the rmarkdown package in Rstudio. This is an easy way of combining R code and output, with text. You can download the .Rmd file from moodle. We are making use of colour plots, and so this is best viewed on a computer, rather than in your black and white lecture handout.

Visualising datasets before fitting any models can be extremely useful. It allows us to see obvious patterns and suggests models and transformations.

R has a very good basic plotting function. Here, however, we use an R package (ggplot2) that provides additional functionality that produces beautiful plots. The first time you use this package you will need to install it (if not using the University computers)

```
install.packages('ggplot2')
```

A nice introduction to this package is provided at http://www.r-bloggers.com/basic-introduction-to-ggplot2/

Plotting the data

The data

We will look at a large dataset consisting of prices and other attributes of 54,000 diamonds. The variables are

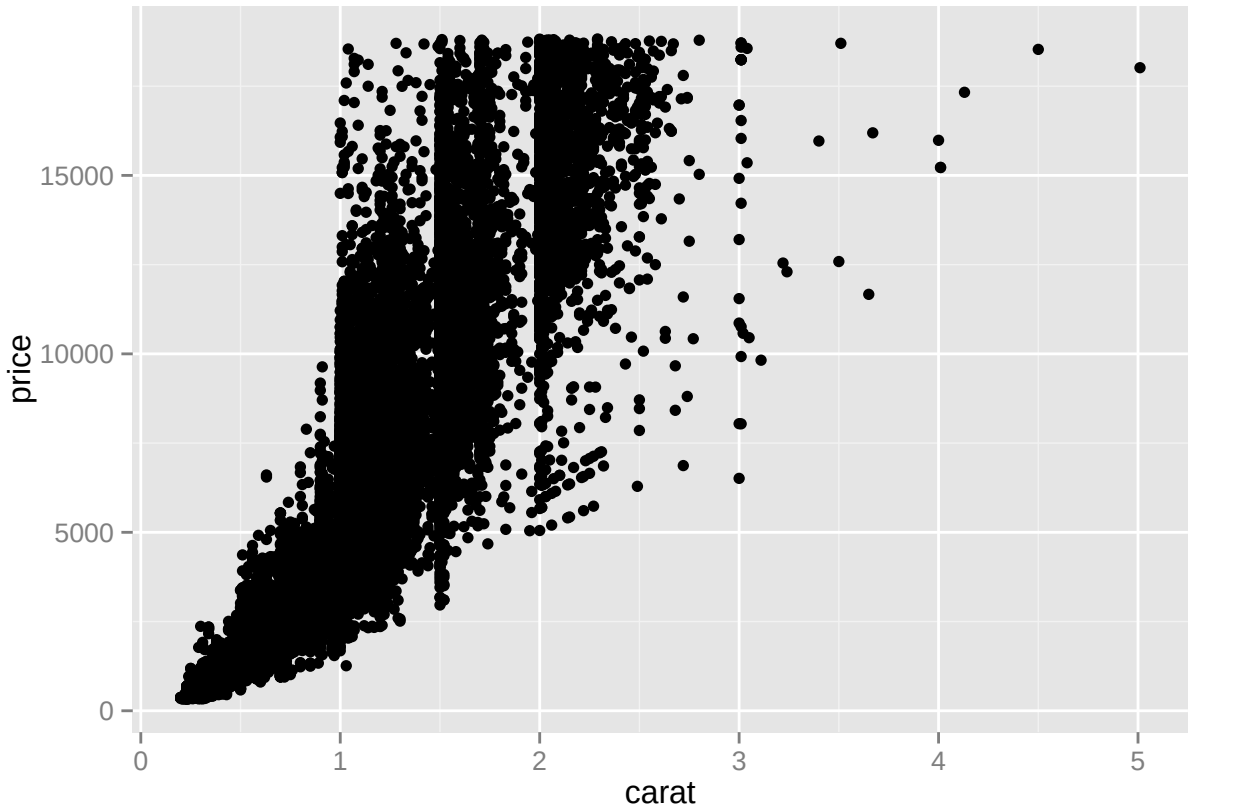
- price: in US dollars
- carat: weight of the diamond
- cut: quality of the cut (Fair, Good, Very Good, Premium, Ideal)
- colour: diamond colour, from J (worst) to D (best)
- clarity: a measurement of how clear the diamond is (I1 (worst), SI1, SI2, VS1, VS2, VVS1, VVS2, IF (best))
- x: length in mm (0–10.74)
- y: width in mm (0–58.9)
- z: depth in mm (0–31.8)
- depth: total depth percentage = z / mean(x, y) = 2 * z / (x + y) (43–79)
- table: width of top of diamond relative to widest point (43–95)

```
library(ggplot2)
str(diamonds)
```

```
## 'data.frame':    53940 obs. of  10 variables:
##  $ carat  : num  0.23 0.21 0.23 0.29 0.31 0.24 0.24 0.26 0.22 0.23 ...
##  $ cut    : Ord.factor w/ 5 levels "Fair"<"Good"<..: 5 4 2 4 2 3 3 3 1 3 ...
##  $ color  : Ord.factor w/ 7 levels "D"<"E"<"F"<"G"<..: 2 2 2 6 7 7 6 5 2 5 ...
##  $ clarity: Ord.factor w/ 8 levels "I1"<"SI2"<"SI1"<..: 2 3 5 4 2 6 7 3 4 5 ...
##  $ depth  : num  61.5 59.8 56.9 62.4 63.3 62.8 62.3 61.9 65.1 59.4 ...
##  $ table  : num  55 61 65 58 58 57 57 55 61 61 ...
##  $ price  : int  326 326 327 334 335 336 336 337 337 338 ...
##  $ x      : num  3.95 3.89 4.05 4.2 4.34 3.94 3.95 4.07 3.87 4 ...
##  $ y      : num  3.98 3.84 4.07 4.23 4.35 3.96 3.98 4.11 3.78 4.05 ...
##  $ z      : num  2.43 2.31 2.31 2.63 2.75 2.48 2.47 2.53 2.49 2.39 ...
```

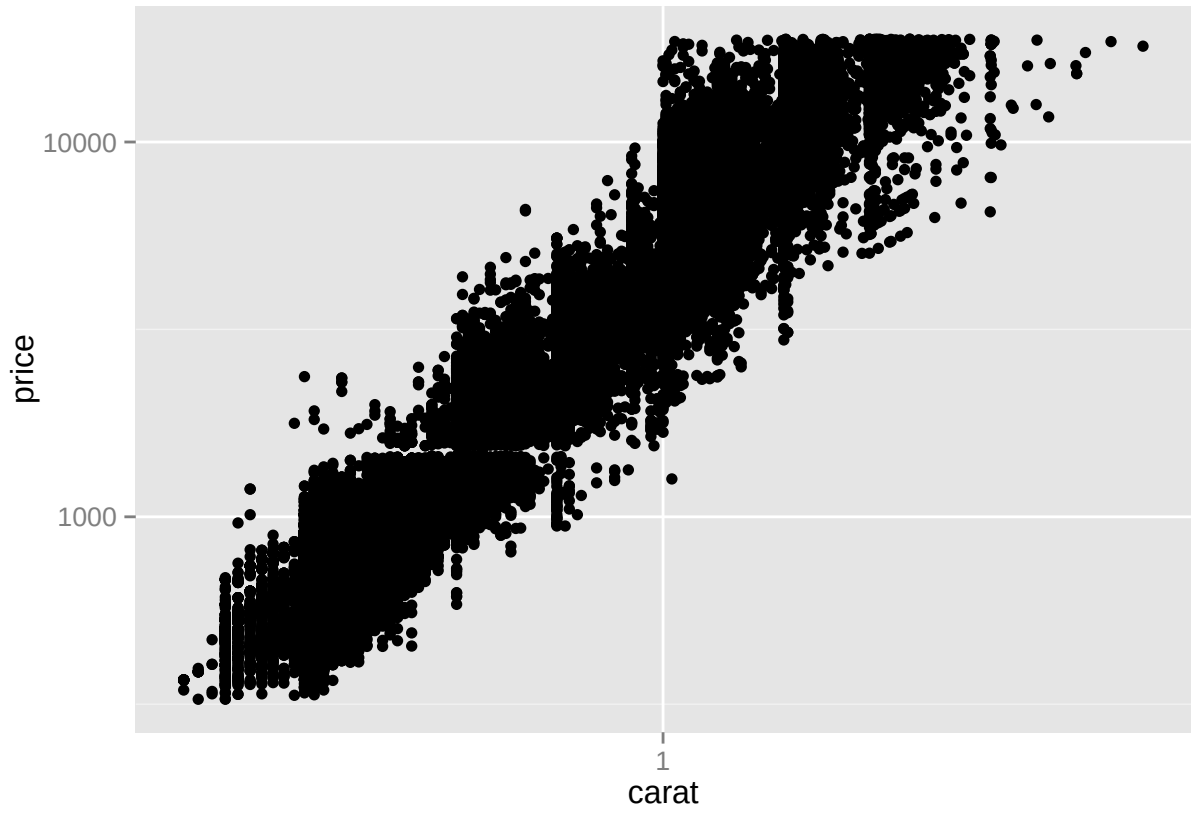
Lets start by looking at the effect of the carat on price.

```
qplot(carat, price, data = diamonds)
```



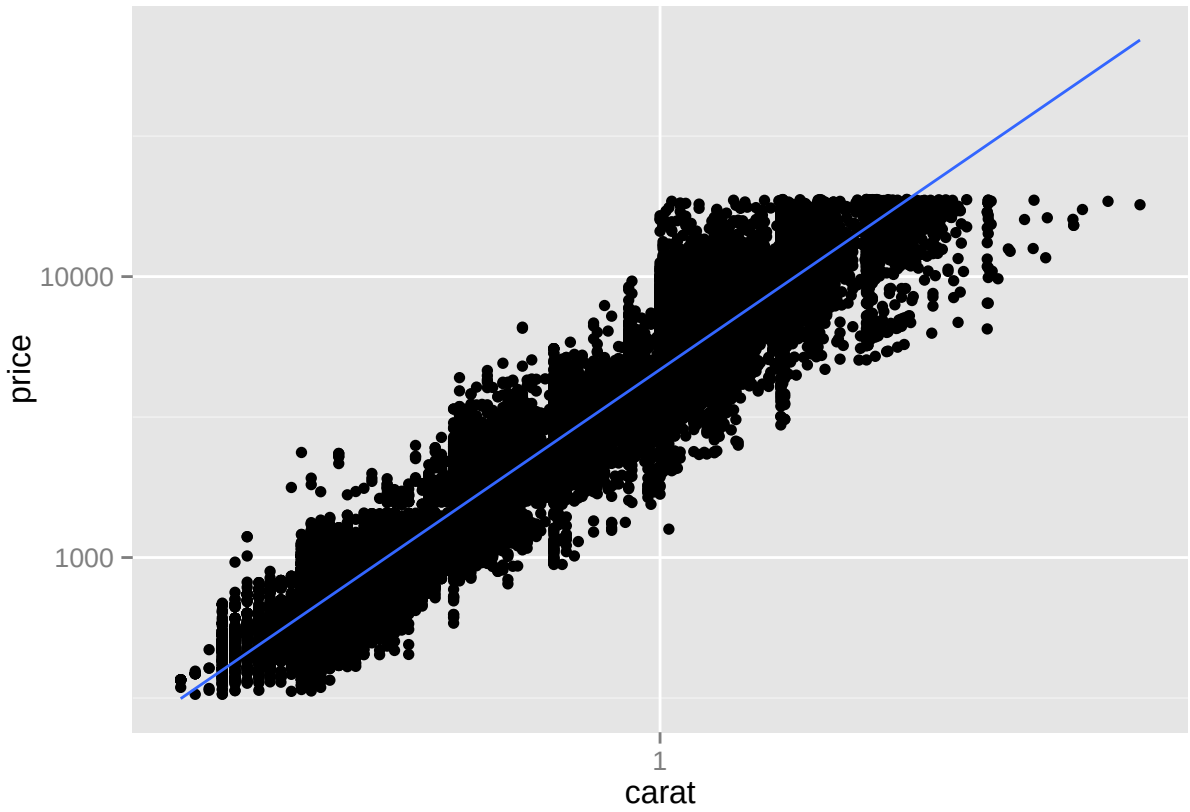
We see the obvious trend that bigger diamonds are generally worth more. It looks like a transformation to x and y would improve this plot. As price is constrained to be positive, lets try taking logs.

```
qplot(carat, price, data = diamonds, log='xy')
```



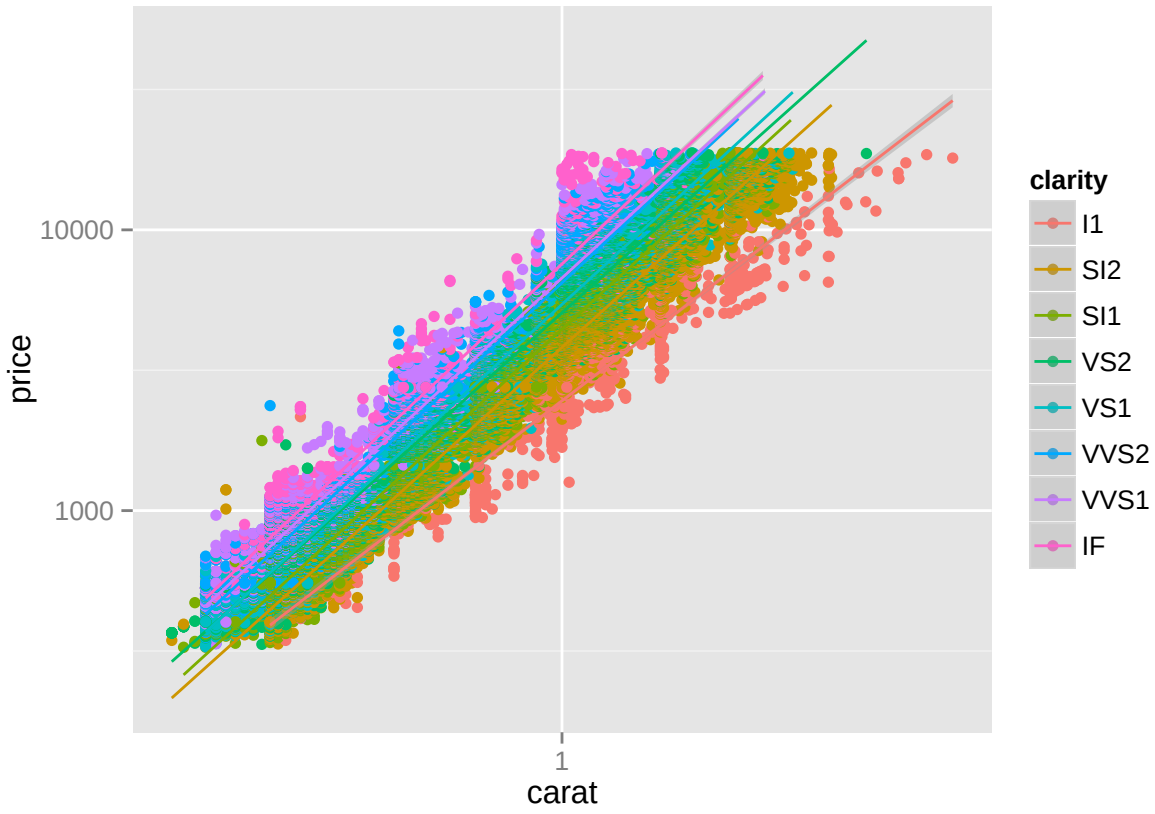
We can add the fitted regression line:

```
qplot(carat, price, data = diamonds, log='xy', geom=c('point','smooth'), method='lm')
```



We can add more information by colouring the points by their clarity (note that clarity is a factor with 8 levels)

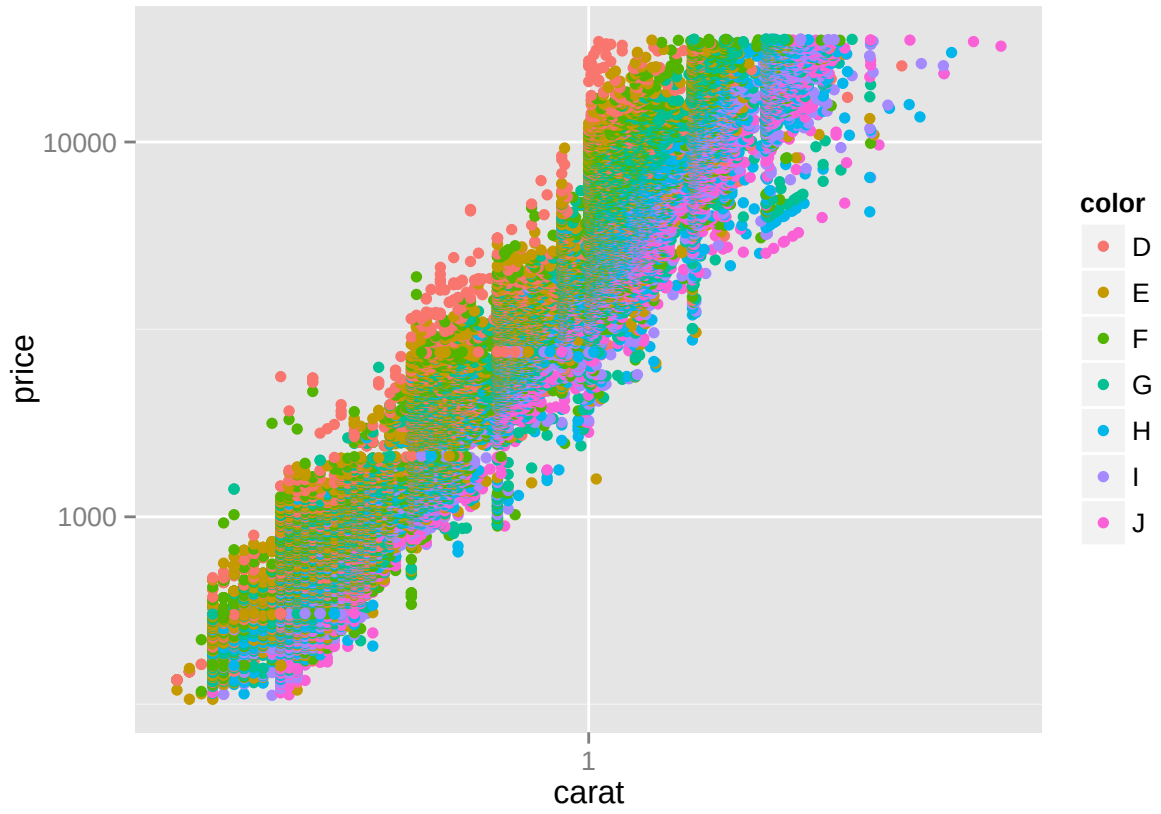
```
qplot(carat, price, data = diamonds,log='xy', geom=c('point','smooth'), method='lm', colour=clarity )
```



The clearer diamonds are worth more!

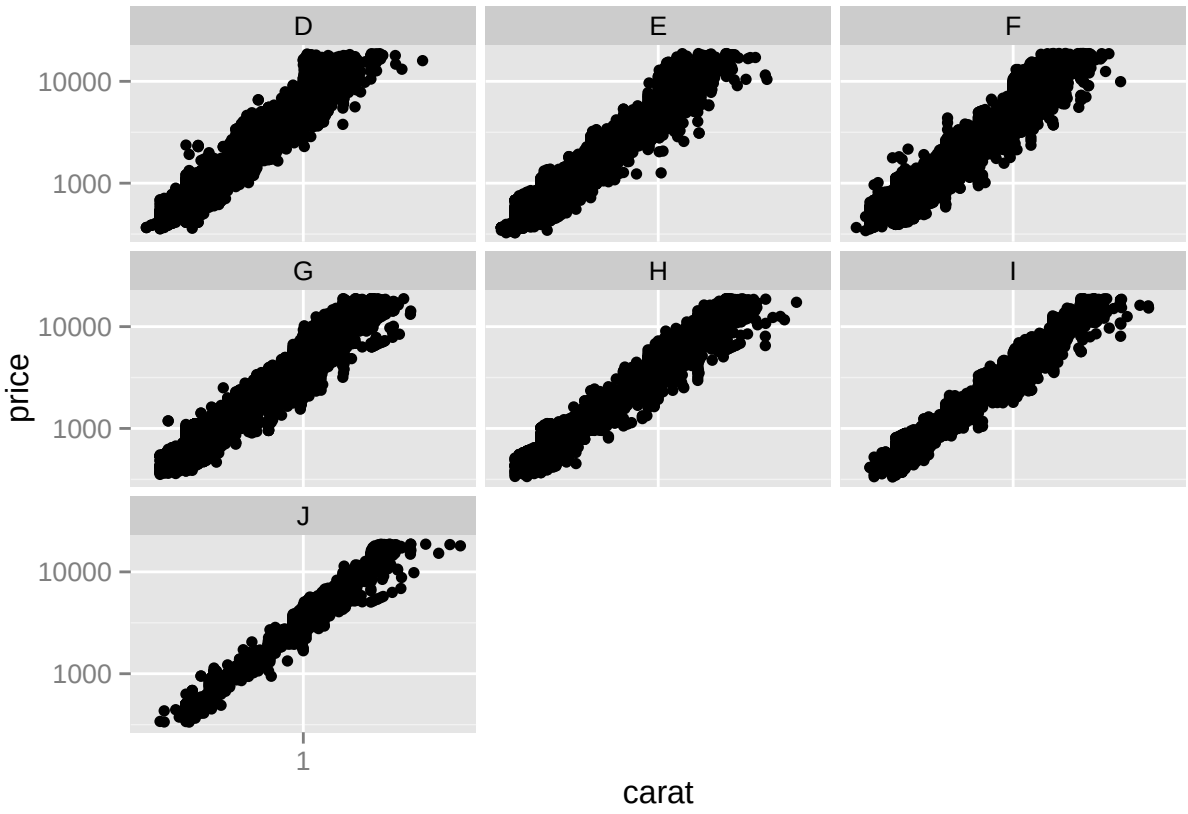
The cut quality also has an effect

```
qplot(carat, price, data = diamonds, colour=color, log='xy')
```



This is less clear, so lets use the facets feature of qplot

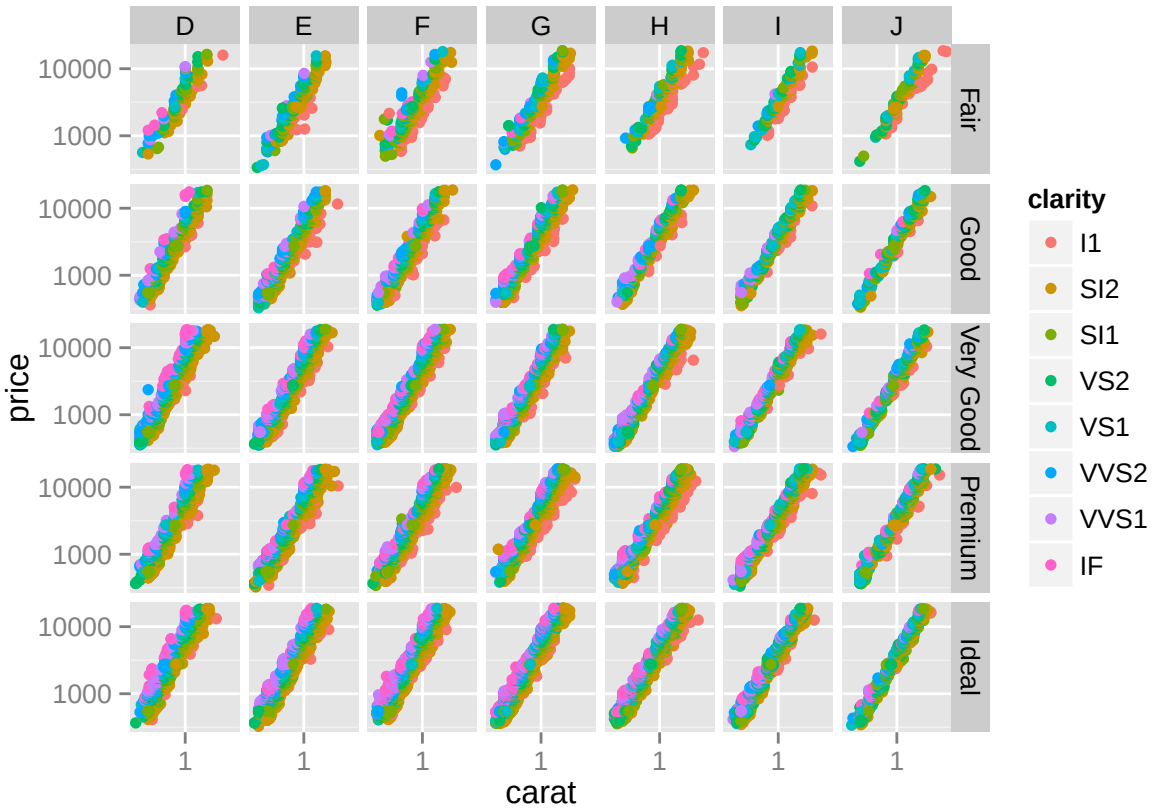
```
qplot(carat, price, data = diamonds, log='xy', facets = ~color)
```



which hasn't particularly helped.

Note that it is possible to include too much information on a single plot:

```
qplot(carat, price, data = diamonds, log='xy', facets = cut~color, color=clarity)
```



Here we've plotted log(price) against log(carat), coloured the points by their clarity, and then given a separate plot for each different diamond colour and cut quality. I don't personally find this very useful, but until you try, you never know. Data visualisation is essentially a case of trial and error

- try lots of different plots, discarding most of them.

Some however, will be useful

- it is these you want to report to your client/include in coursework etc.