**Semana 4**

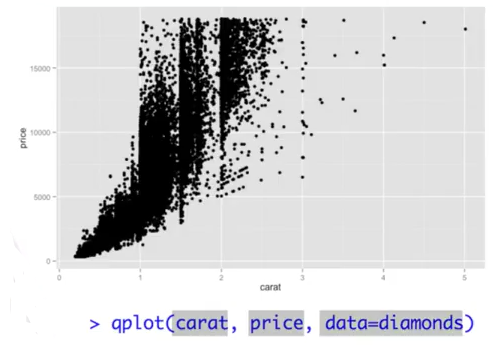
**19. Graficación con ggplot2**

Ahora aprenderemos el paquete ggplot2, que nos aporta gráficos de alto nivel donde el usuario no tiene que preocuparte por los detalles. En este, las leyendas o los tipos de fuente ya tienen valores por omisión, que son tan de buena calidad que pueden usarse directamente en publicaciones. Funciona con base en capas, en la que cada capa añade más información (en la primera se van viendo los datos crudos y luego se van añadiendo más datos). En vez de pasar mucho tiempo concentrado en los detalles del gráfico, aquí puedes ver cómo puedes ir añadiendo los datos e ir añadiendo más información de ellos.

Con la función *qplot()* se pueden hacer muchos gráficos y variar muchos atributos de los mismos. Con esta función, aprenderemos a hacer diferentes gráficos, y aprenderemos como mapear variables a diversos atributos estéticos. Para poder llevar a cabo los elementos, usaremos un conjunto de datos llamado *diamonds*, que es la información de más de 54000 diamantes.

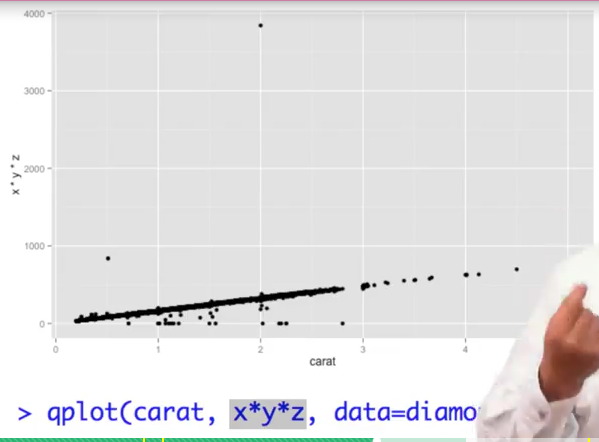


Se guarda en una nueva variable denominada *dchico*. Ahora podemos crear gráficas, recibiendo dos parámetros iniciales. Representa la gráfica de dispersión, aunque en este caso no necesitamos pasarle los vectores directamente.

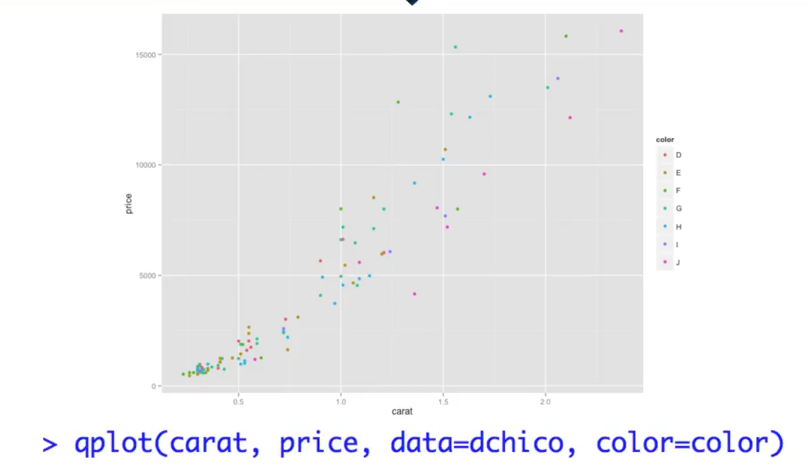


La diferencia con *plot()* es que en este caso no es necesario pasarle los vectores directamente, sino que solo necesitamos pasarle el nombre de las columnas y luego con *data* indicamos el archivo en el que se encuentran. Podemos ver que hay mucha saturación por la gran cantidad de datos, por lo que si utilizamos la muestra de *dchico* se muestra mejor. Hay cierta correlación exponencial y hay ciertos lugares donde se hacen estrías.

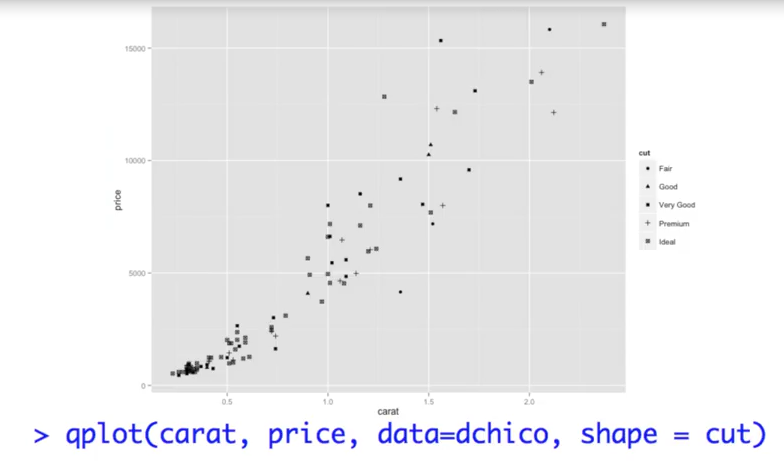
Podemos combinar los valores de una variable para representarlos juntos.



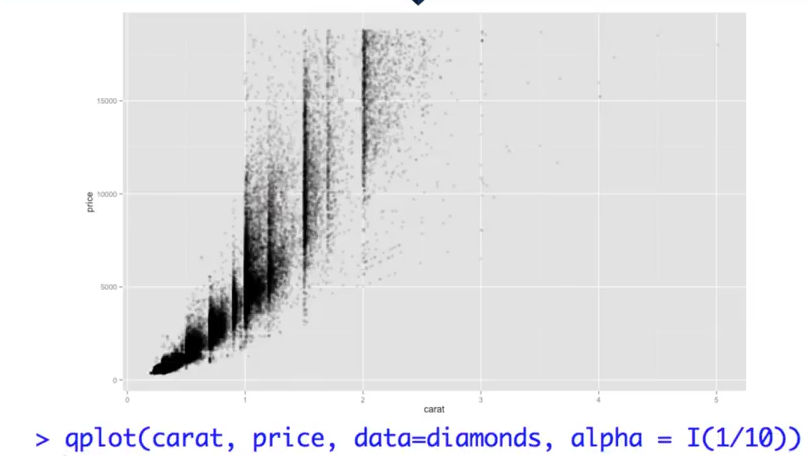
Los atributos estéticos como el color se pueden etiquetar con una variable.



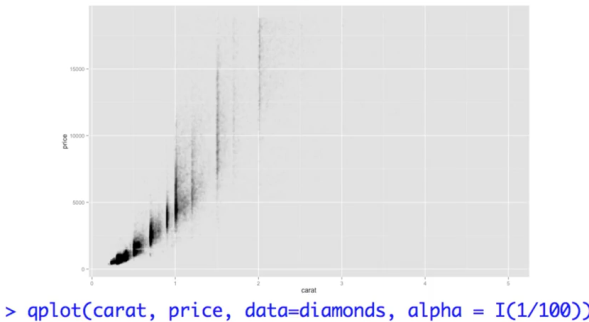
Otro atributo estético es *shape* que se refiere a la forma con la que pinta los puntos en la gráfica. Shape es bueno para representar características categóricas a gráficas.



El tercer atributo es alpha, que indica cuantos puntos se necesitan pintar para que podamos pintar un punto completamente opaco.



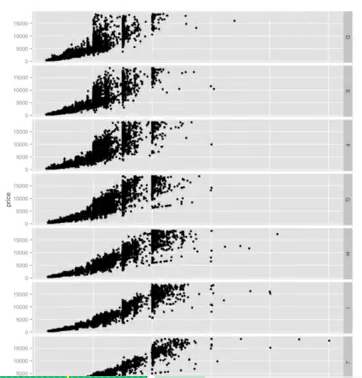
Si utilizamos 1/10 indicamos que queremos representar solo 1/10 de la profundidad del punto único. Es decir, necesitaríamos 10 puntos para que se pintara un solo punto. Si en lugar de 1/10 se representa con 1/100, se podría observar así:



El último atributo es *size* o tamaño, que en cada punto, dependiendo de la categoría que mapeemos a la variable representa un valor más grande o más pequeño.

El atributo *faceting* nos permite representar una tabla donde podemos subdividir en grupos.





Si queremos ver cómo se comparta la variable *carat* contra la *Price* y queremos el *faceting* por color va representando cada una de las tablas en función del color del diamante. Esto nos permite la comparación visual inmediata.

Los *geoms* o *geometrics objects* son una descripción de los objetos que queremos usar para representar. Con *smooth* podemos ver el tipo de gráfica que queremos. Otros *geoms* que vamos a ver sin *jitter* y *boxplots*, el primero organizado en gráficas y el segundo por gráficas de cajas. Otros dos *geoms* son *line* que nos dibuja una línea entre dos puntos, generando un gráfico que va de izquierda a derecha, usándose para representar series de tiempo, o *graph*, que une los valores en cualquier dirección.

Finalmente, podemos generar gráficas en una sola dimensión, como cuando queremos ver cómo se distribuye cierta variable en el conjunto de datos, como *histogram* en *geom=”histogram”*. También están *freqpoly y density*, que crean tablas de frecuencias y la densidad de datos.

* **Tarea de RStudio**

La función *View()* permite ver los elementos de un archivo que se pase en el ambiente.