`qplot` R Graphics Cheat Sheet

David Gerard

2019-01-21

Abstract:

I reproduce some of the plots from Rstudio's ggplot2 cheat sheet using just the `qplot` function.

Before using `qplot` in a new R session, always first load the ggplot2 library.

```
library(ggplot2)
```

I use this dataset

```
data(mpg, package = "ggplot2")
```

General Considerations

The main options that I use are

- Options for `"geom"` argument:
  - `"point"`: Makes scatterplots.
  - `"line"`: Makes a line plot.
  - `"histogram"`: Makes a histogram.
  - `"boxplot"`: Makes a boxplot.
  - `"density"`: Makes the density plot.
  - `"bar"`: First tabulates frequencies of each value, then makes a barplot.
  - `"smooth"`: Fits a smooth line to a cloud of points and plots the output.
  - `"dotplot"`: Makes a dotplot.

`qplot` has other arguments that control the way the plot looks. You should read about these arguments. In particular, read carefully the help page `?qplot`. Useful ones are:

- `data`: Specify the dataframe that all variables belong to.
- `main`: This controls the title.
- `xlab`, `ylab`: These control the x and y axis labels.
- `color`: Controls the color of the lines/points.
- `fill`: Controls the color of areas (e.g. for histograms).
- `size`: Controls the size of points.
- `shape`: The shape of points (`"circle"`, `"square"`, `"triangle"`, etc...)
- `alpha`: Controls the level of transparency of points/lines/fills.
- `lwd`: Line width.
- `lty`: Line type (`"solid"`, `"dashed"`, `"dotted"`, etc...).
- `facets`: Split up the data into multiple plots.

If you want to make all points the same shape/size/color, you need to enclose the size/shape/color using the function `I()`.

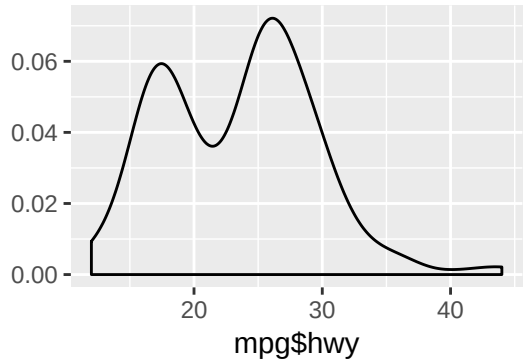
If a variable is being treated as continuous rather than categorical, you need to enclose that variable in a `factor()` function call.

One Variable

Continuous

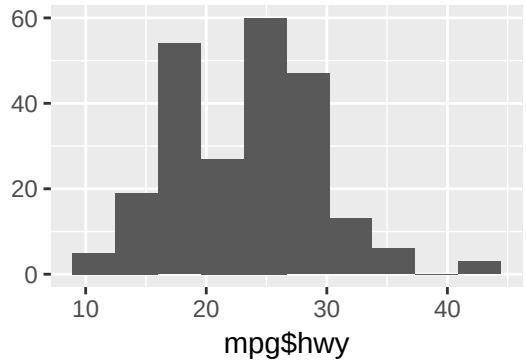
Density plot

```
qplot(x = mpg$hwy, geom = "density")
```



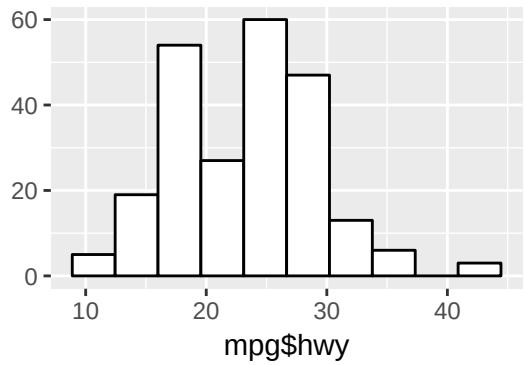
Histogram

```
qplot(mpg$hwy, geom = "histogram", bins = 10)
```



Make the bin lines black and the fill white.

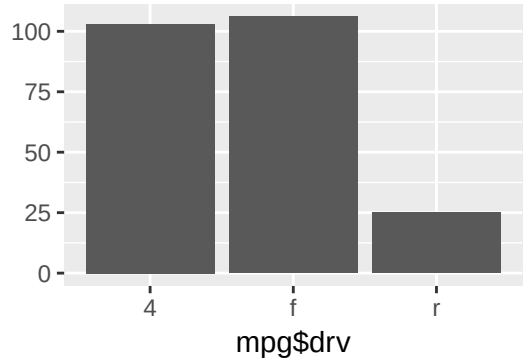
```
qplot(mpg$hwy, geom = "histogram", bins = 10, color = I("black"), fill = I("white"))
```



Discrete

Barplot

```
qplot(mpg$drv, geom = "bar")
```

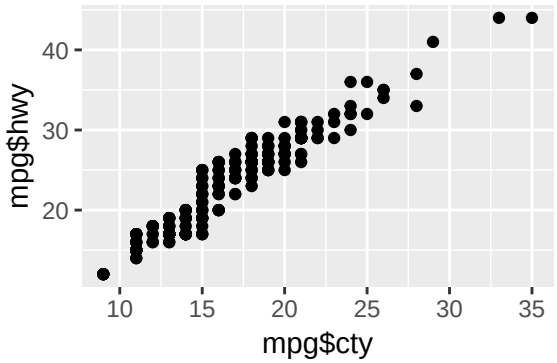


Two Variables

Continuous X, Continuous Y

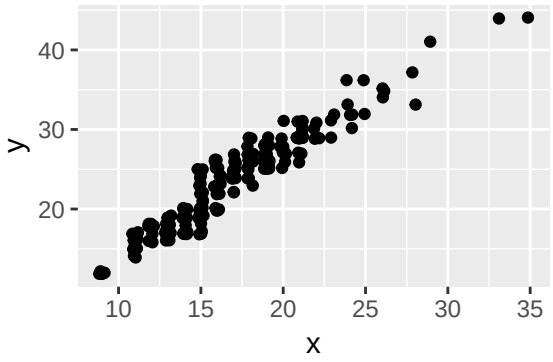
Scatterplot

```
qplot(mpg$cty, mpg$hwy, geom = "point")
```



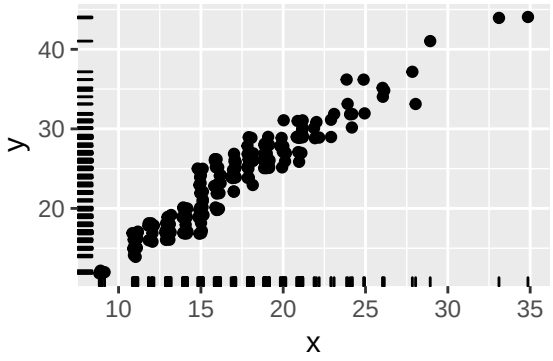
Jitter points to account for overlaying points.

```
x <- jitter(mpg$cty)
y <- jitter(mpg$hwy)
qplot(x, y, geom = "point")
```



Add a rug plot

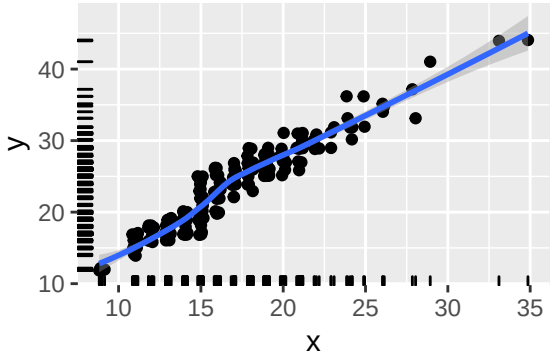
```
qplot(x, y, geom = "point") +
  geom_rug()
```



Add a Loess Smoother

```
qplot(x, y, geom = "point") +
  geom_rug() +
  geom_smooth()
```

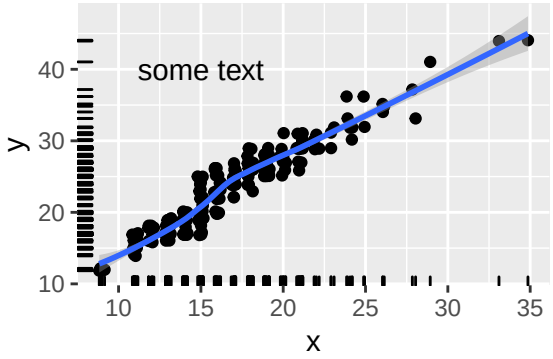
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Add text to a plot

```
qplot(x, y, geom = "point") +
  geom_rug() +
  geom_smooth() +
  annotate(geom = "text", x = 15, y = 40, label = "some text")
```

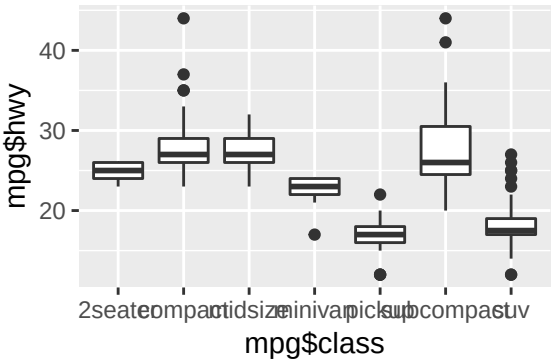
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Discrete X, Continuous Y

Boxplot

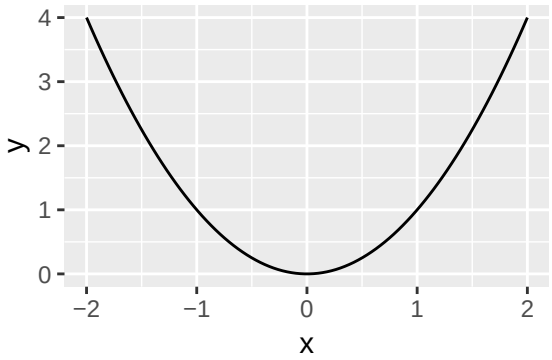
```
qplot(x = mpg$class, y = mpg$hwy, geom = "boxplot")
```



Continuous Function

Line plot

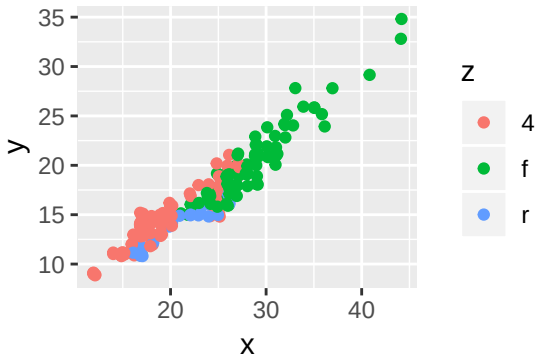
```
x <- seq(-2, 2, length = 100)
y <- x^2
qplot(x, y, geom = "line")
```



Color Coding and Legends

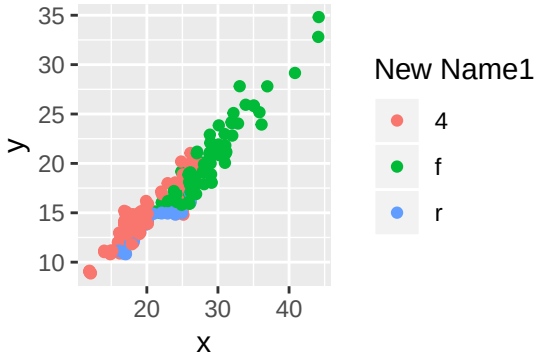
Color code a scatterplot by a categorical variable and add a legend.

```
x <- jitter(mpg$hwy)
y <- jitter(mpg$cty)
z <- factor(mpg$drv)
qplot(x, y, color = z)
```

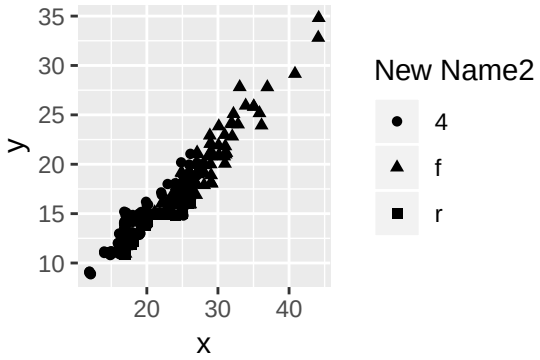


Changing a legend title

```
qplot(x, y, color = z) + scale_color_discrete(name = "New Name1")
```



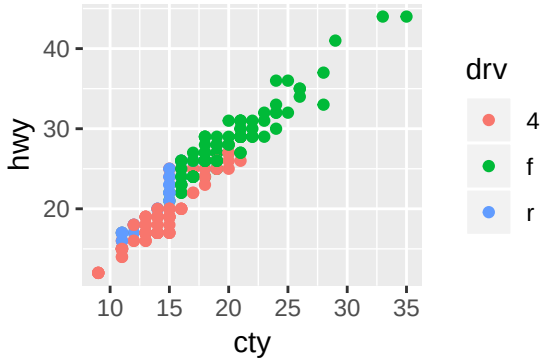

```
qplot(x, y, shape = z) + scale_shape_discrete(name = "New Name2")
```



The `data` argument

If all variables you are using in `qplot()` belong to the same dataframe, then you can specify the dataframe as the "data" argument and you don't need to use the "$" symbol.

```
qplot(cty, hwy, color = drv, data = mpg, geom = "point")
```

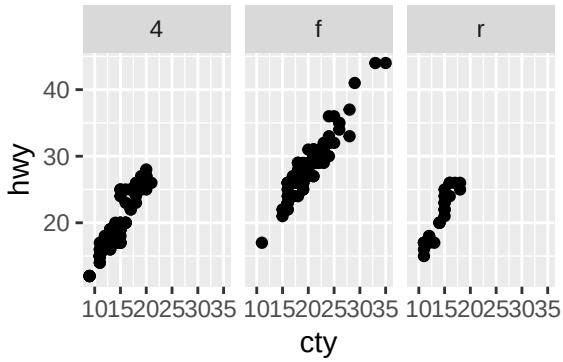


Faceting

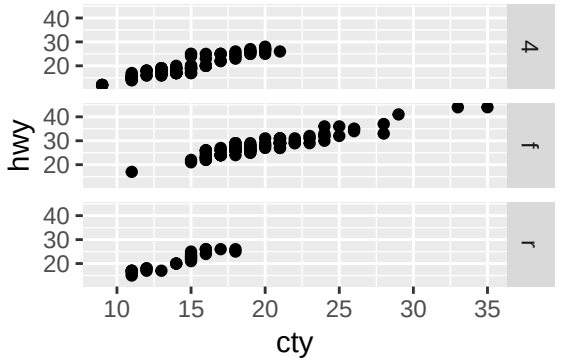
You can facet by a categorical variable using the `facets` argument.

The variable to the left of the tilde ("~") indexes the row facets, the variable to the right of the tilde indexes the column facets. Using a dot (".") in place of a variable means that there will only be one row/column facet.

```
qplot(cty, hwy, data = mpg, facets = . ~ drv, geom = "point")
```




```
qplot(cty, hwy, data = mpg, facets = drv ~ ., geom = "point")
```




```
qplot(cty, hwy, data = mpg, facets = fl ~ drv, geom = "point")
```