Analyze a car dataset with visualization

Merlin Shi

10/31/2017

Introduction

The topic of this post is data visualization with ggplot. As taught in lecture, we may use ggplot to plot various kinds of graphs, such as scatterplot, line graph, and fitting curves. However, there is much more that ggplot provides for us. In this post, we are going to explore more functionality of ggplot package along with the intrinsic relationship of columns of a specific dataset, mtcars. We will try scatterplot, faceting layout, boxplot, and pie chart.

examples

```
library(ggplot2)
```

We use the built-in dataset, mtcars, which comprises various aspects of car models, to investigate the relationship between automobile design and performance.

First, let's load the dataset and take a preview.

```
# Loading
dat <- mtcars
# Print the first few rows
head(dat)
```

```
##                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

The data frame contains 32 observations on 11 variables. By analyze the 11 numeric parameters (all in double) of 32 models of car, we can get a general sense of the influence of design on performances and the tradeoff between performances.

data cleaning

We can cast variable \(cyl\) (number of cylinders) from a vector of double to a factor, because data in this column are always integer. Later, we will see the benefit of casting to factor.

```
# dat$cyl <- as.character(dat$cyl)   # An alternative to explicitly express discreteness
dat$cyl <- factor(dat$cyl)
head(dat)
```

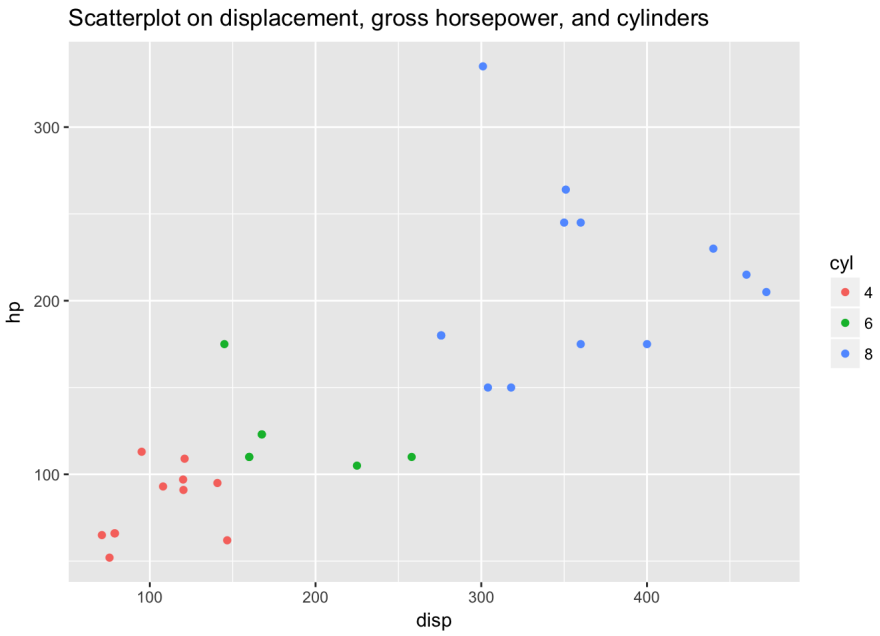
```
##                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
# typeof(dat$cyl) is "integer", class(dat$cyl) is "factor"
```

Relation between displacement, gross horsepower, and number of cylinders

As a warmup, we want to explore the relationship among the selected three variables. We take \(disp\) and \(hp\) as x- and y-axis, and use \(cyl\) to color the scatterpoints. Empirically, more cylinders lead to higher displacement and higher horsepower.

```
ggplot(data = dat, aes(x = disp, y = hp)) +
  geom_point(aes(colour = cyl)) +
  labs(title = "Scatterplot on displacement, gross horsepower, and cylinders")
```



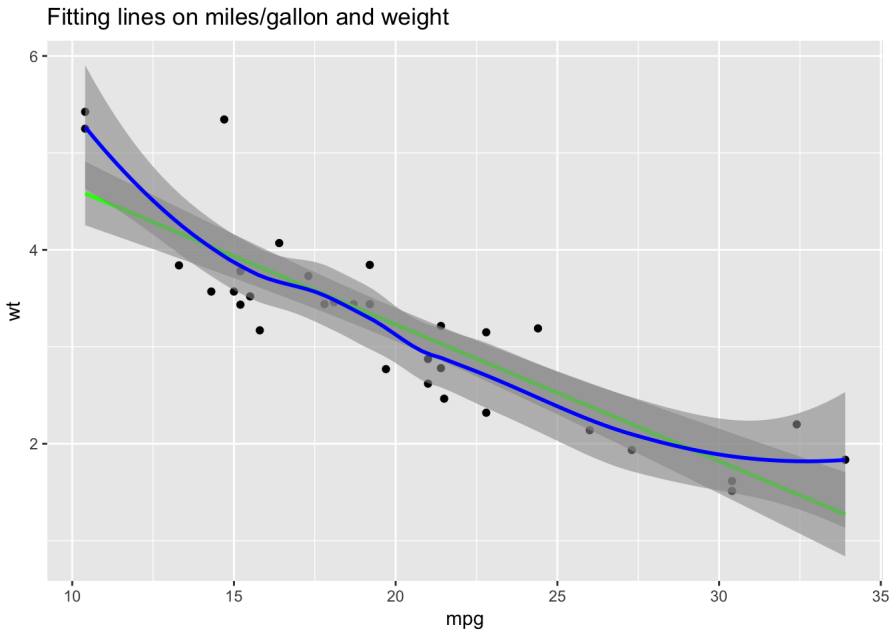
Since \(cyl\) is in factor, the legend in the side shows 3 discrete colors rather than a color gradient.

The visualization verifies our thoughts. Generally, the displacement of a 4-cylinder car ranges from 50 to 150 cu.in.; that of a 6-cylinder car ranges from 150 to 270 cu.in; and that of a 8-cylinder car is 270 cu.in. and above. There are clear distinctions between cars of different number of cylinders. On the other hand, number of cylinders and gross horsepower are positively correlated, but a car with less cylinders can have horsepower greater than a car with more cylinders.

Relation between fuel consumption (mpg) and weight

Next, we would like to explore the relation between fuel consumption and weight. From our experience, we assume that these two indices are negatively correlated. It would be useful if we can predict the other variable when we only have one.

```
ggplot(data = dat, aes(x = mpg, y = wt)) +
  geom_point() +
  geom_smooth(method = 'glm', color = 'green', alpha = 0.6) +
  geom_smooth(method = 'loess', color = 'blue', alpha = 0.6) +
  labs(title = "Fitting lines on miles/gallon and weight")
```



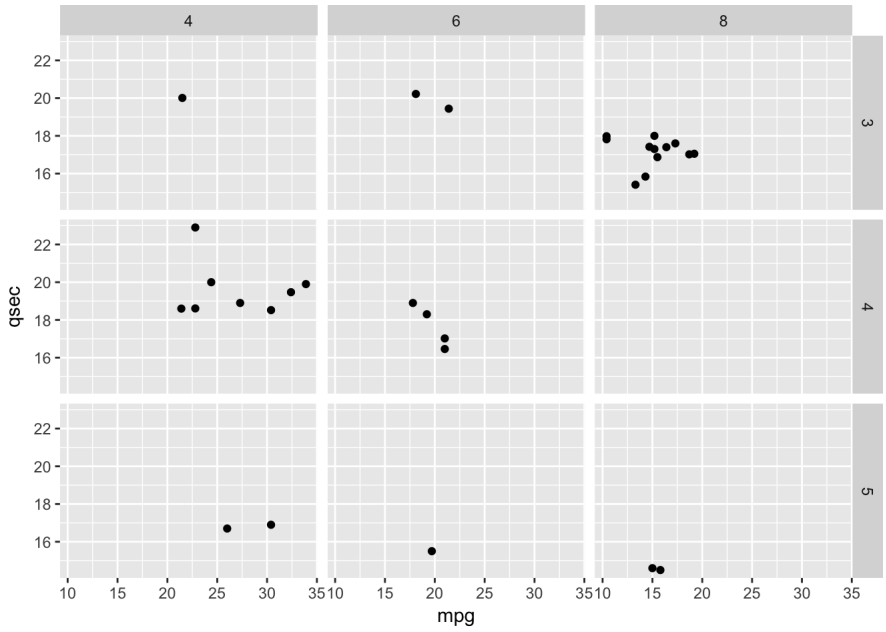
From the plot above, data show that the two variables are indeed negatively correlated. We use two different approaches to fit scatter points. The green line assumes a linear underlying model, whereas the blue line adopts loess method. In term of the dissimilarity to the true unknown underlying model, the green line is of larger bias and the blue line is of larger variance. With more data to come (larger number of car models; larger number of cars per model), the variance can be reduced, due to the Law of Large Numbers.

Performance and design

We want to know how the performance (mpg and qsec) is influenced by design (number of forward gears, and number of cylinders). Since there are so many variables involved, we can't fit all these into the 2-dimensional plane of the post. Fortunately, ggplot provides a faceting tool to divide a plot into several subplots. We can exploit this tool to analyze the connection between performance and design.

The design features are discrete, so it is reasonable to use them as faceting parameters.

```
ggplot(data = dat, aes(x = mpg, y = qsec)) +
  geom_point() +
  facet_grid(gear ~ cyl)
```

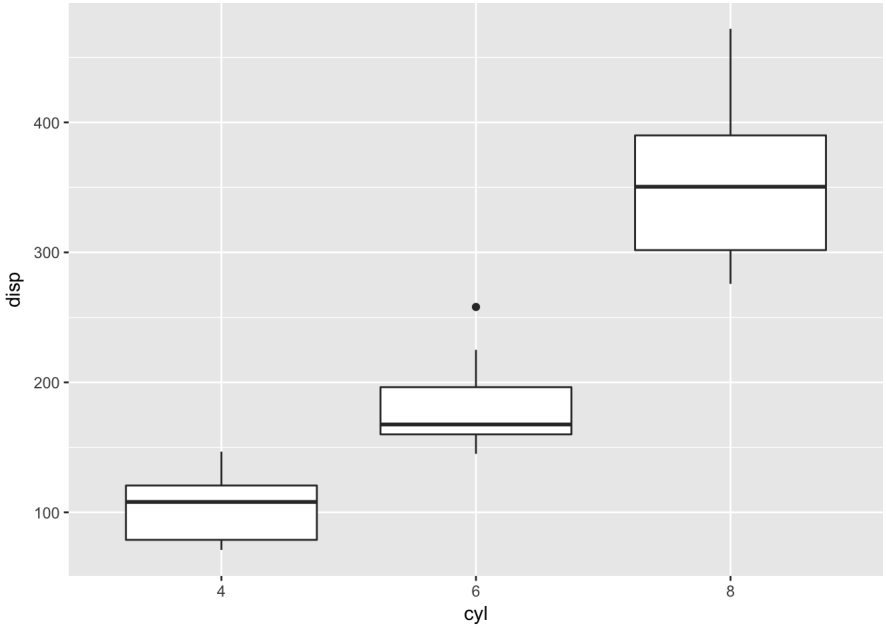


Here we experience the issue that the data is limited. We only have 32 observations. The quantity is even smaller when we divide them into non-overlapping subplots. The general spirit is there, but we need more data to do a more comprehensive research.

More ggplot features

When X is discrete and Y is continuous, a better choice of graph is boxplot, which is easy to implement in ggplot.

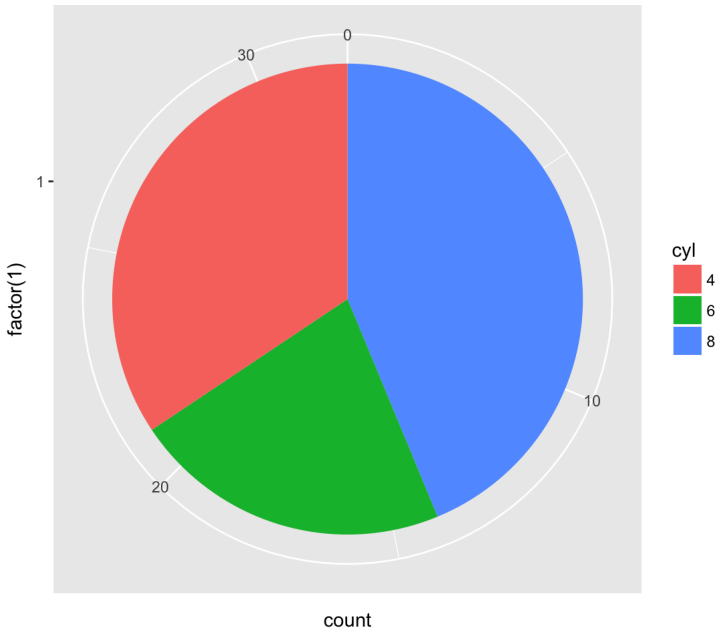
```
ggplot(data = dat, aes(x = cyl, y = disp)) +
  geom_boxplot()
```



From this graph, the corresponding displacement to number of cylinders have great distinction.

Use polar coordinate system to show the distribution of number of cylinders in the dataset.

```
ggplot(data = dat, aes(x = factor(1), fill = cyl)) +
  geom_bar(width = 1) +
  coord_polar(theta = "y")
```



conclusions

- The central idea of ggplot is that each graph is built from a dataset, a set of geoms, and a coordinate system. By configuring these components, the author can plot for almost any purpose. In this way, ggplot provides a high degree of freedom.
- ggplot is designed to to added layer by layer, and therefore easy to code. Whenever you need a new feature (scatter points, lines, facets), you can just write another line with a preceding “+” to instruct ggplot to do the task.

references

https://flowingdata.com/2016/03/22/comparing-ggplot2-and-r-base-graphics/

http://ggplot2.tidyverse.org/reference/geom_boxplot.html

https://medium.com/optima-blog/using-polar-coordinates-for-better-visualization-1d337b6c9dec

http://ggplot2.tidyverse.org/reference/coord_polar.html

https://www.rstudio.com/wp-content/uploads/2015/03/ggplot2-cheatsheet.pdf

https://www3.nd.edu/~steve/computing_with_data/11_geom_examples/ggplot_examples.html

https://thepracticalr.wordpress.com/tag/ggplot2/