

Chapter 2

Make a Plot by ggplot2

In the previous chapter, you have learned how to make a plot with just a click, without coding and applying various themes to the plot. In this chapter, I will give you a short introduction about the R package “ggplot2”.

2.1 The Grammar of Graphics

The term ggplot is an abbreviation of grammar of graphics plot. Leland Wilkinson (Adjunctive Professor of Computer Science, University of Illinois at Chicago) published a book entitled **The Grammar of Graphics** in 2005 and the R package ggplot2 is written by Hadley Wickham. The function ggplot() in ggplot2 package makes a plot using this grammar. Traditionally, the plots are classified into scatter plots, line graphs and bar graphs according to the appearance, but the ggplot2 is designed to work in a layered fashion.

2.1.1 The Components of Graphics

According to the grammar of graphics, the components of graphics are as follows.

- (1) Data: The data is what we want to visualize. You can use data.frame of R only in ggplot2.
- (2) Coordinate system(coord): A coord describes how data coordinates are mapped to the plane of the graphics. We normally use Cartesian coordinates (default), but a number of others are available, including polar coordinates and map projections.
- (3) Geoms: The geoms are the geometric objects that are drawn to represent the data such as points, lines, areas, polygons, etc.

- (4) Aesthetics: Aesthetics are visual properties of geoms such as x and y positions, colors, shapes, transparency, etc.
- (5) Scales: Scales map values in the data space to values in the aesthetic space whether it is color, size, or shape.
- (6) Statistical transformation(stats): The stats summarize data in many useful ways. Examples binning and counting to create a histogram and regression line for regression analysis.
- (7) Facets: How to break up the data into subsets and how to display those subsets as small multiples.

2.1.2 Steps for Making a Plot with ggplot2

- (1) Assign data: To make a plot with ggplot2, you have to declare the input data. frame.(e.g., data=acs).
- (2) Assign or set the aesthetics(aes): Assign a variable to aes or set the aes. You have to assign or set the x-axis variable. For example, you can make a histogram or density curve using the x-axis variable only. To make a scatter plot, you have to assign the y-axis variable together. You can assign a variable to color, fill, or size variable(e.g., color=sex) or set the variable(e.g., color = “black”).
- (3) Specify the geom(s): You can select various geoms layer by layer. For examples, you can select points, lines, areas, or polygons layer by layer.

Because the coordinate system and scales have default values, you can draw a plot without setting them. You can change the coordinate system or scales if needed. Stats and facets can be added as desired.

2.1.3 Example 1: Salaries of Professors

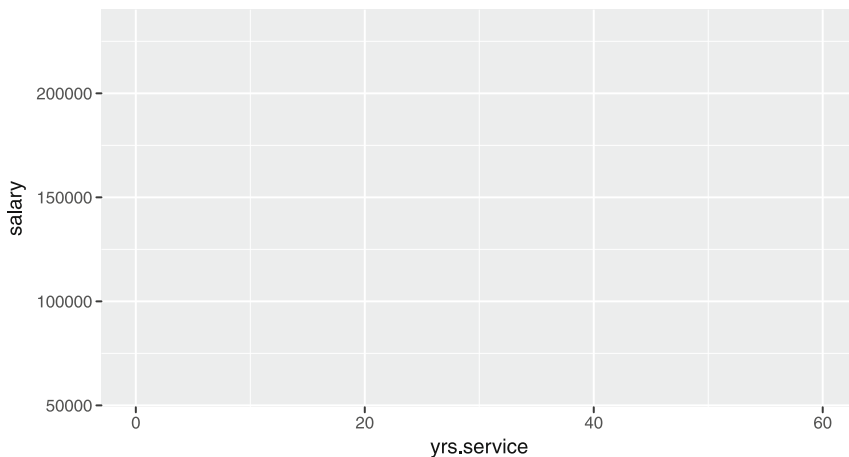
In the previous chapter, you have made a plot about salaries of professors. You can make a ggplot using the following R codes.

```
require(ggplot2)    # for use of ggplot()
require(car)        # for use of data Salaries

str(Salaries)      # See the structure of Salaries data

'data.frame':   397 obs. of  6 variables:
 $ rank          : Factor w/ 3 levels "AsstProf","AssocProf",...: 3 3 1 3 3 2 3 3 3 3 ...
 $ discipline    : Factor w/ 2 levels "A","B": 2 2 2 2 2 2 2 2 2 ...
 $ yrs.since.phd : int   19 20 4 45 40 6 30 45 21 18 ...
 $ yrs.service   : int   18 16 3 39 41 6 23 45 20 18 ...
 $ sex           : Factor w/ 2 levels "Female","Male": 2 2 2 2 2 2 2 2 1 ...
 $ salary        : int  139750 173200 79750 115000 141500 97000 175000 147765 119250 129000 ...

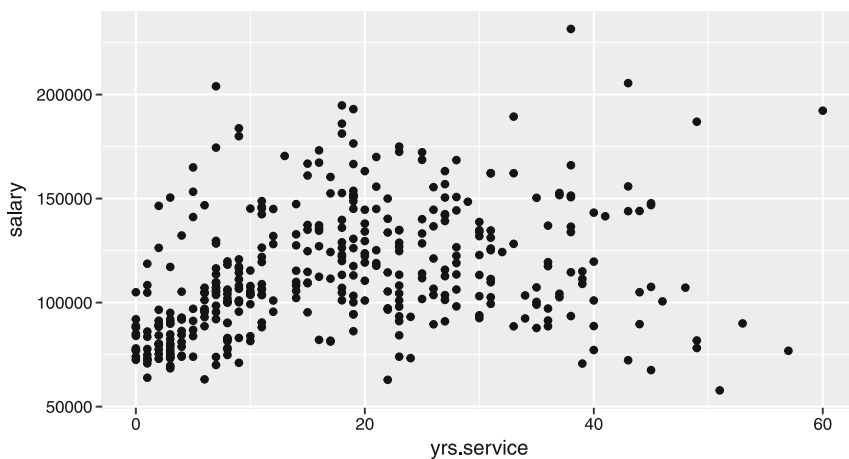
ggplot(data=Salaries, aes(x=yrs.service,y=salary)) # assign the data and variables
```



You can confirm that you have assigned the data and variables correctly.

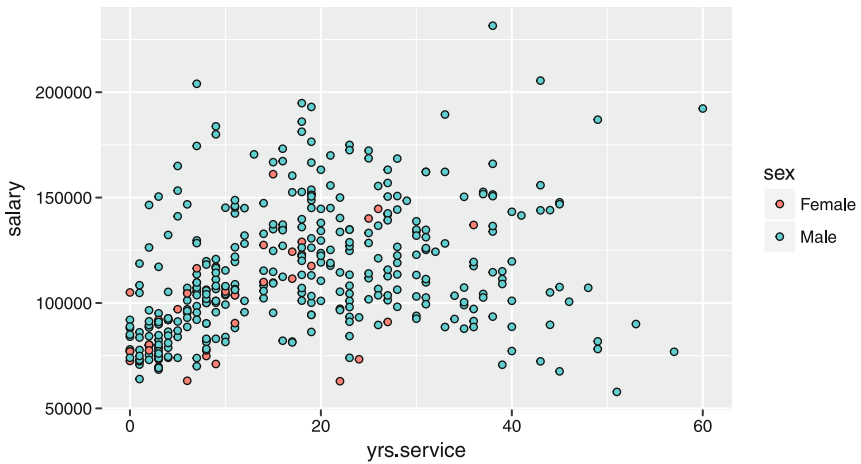
You can make a scatterplot with the following code.

```
p <- ggplot(data=Salaries, aes(x=yrs.service,y=salary))  
p + geom_point() # add scatterplot
```



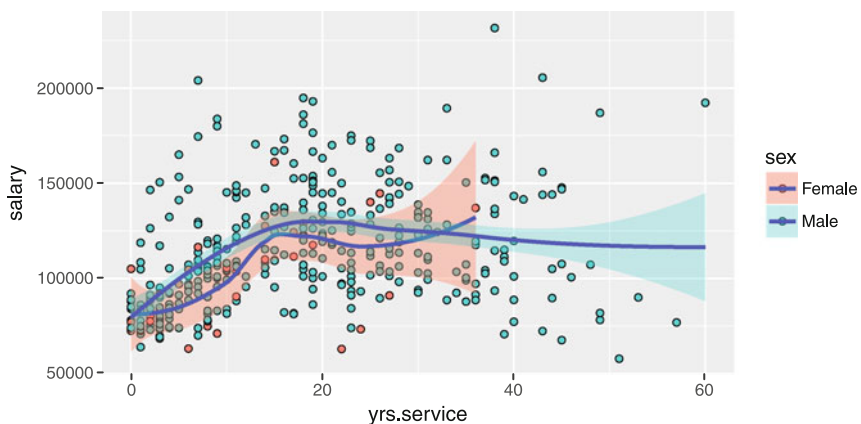
You can assign the sex to a fill variable.

```
p <- ggplot(data=Salaries, aes(x=yrs.service,y=salary,fill=sex)) # assign colour variable
p + geom_point(shape=21) # add scatterplot
```



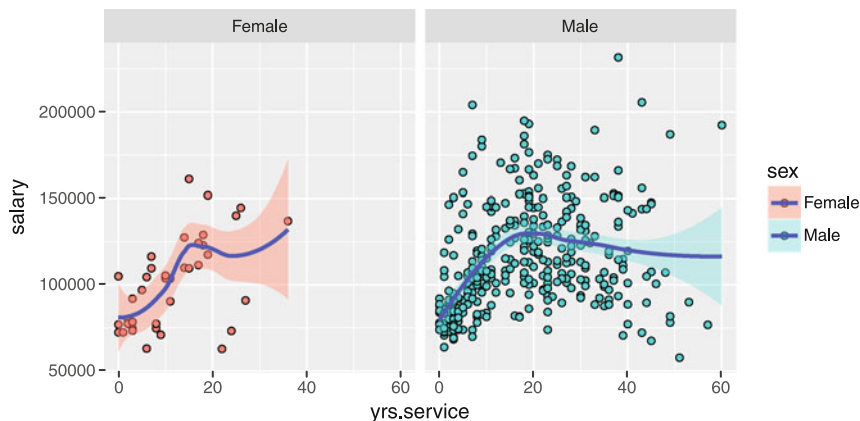
You can add regression lines layer by layer using `geom_smooth()` function. By default, LOESS regression lines are added. Because the sex is assigned as a color variable, two regression lines are added.

```
p + geom_point(shape=21) + # add scatterplot
  geom_smooth() # add regression lines
```



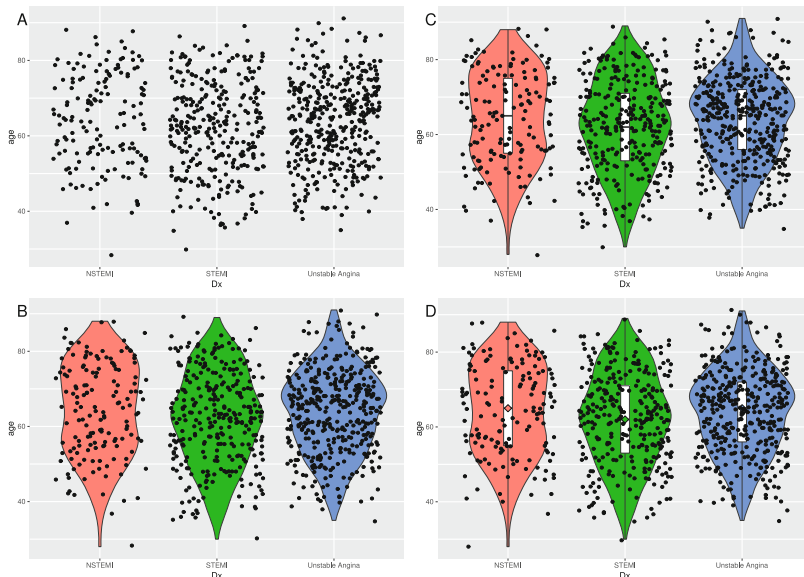
You can organize the data into subplots with facets.

```
p + geom_point(shape=21) +           # add scatterplot
  geom_smooth() +                     # add regression lines
  facet_grid(~sex)                   # faceted by sex in horizontal arranged subpanels
```



2.1.4 Example 2: The acs Data

The acs data included in the package moonBook is a dataset containing 857 patients with acute coronary syndrome(ACS). You can plot the age and diagnosis(Dx) as follows. Panel A is a scatter plot and panel B is a scatterplot with violin plot. Panel C is a scatterplot with violin plot and box plot. Panel D is a scatterplot with violin plot and box plot and has a median value using stat_summary() function.



I have taught R software and ggplot2 for many years. But most of my students failed to overcome the learning curve of ggplot2. So I have developed a shiny app named as “Learn ggplot2”. I developed this app for educational purpose. With this app, one can make a plot using ggplot2 without having to code each step and will become familiar with the ggplot2 code.

<http://www.springer.com/978-3-319-53018-5>

Learn ggplot2 Using Shiny App

Moon, K.-W.

2016, XVI, 351 p. 346 illus., 336 illus. in color.,

Softcover

ISBN: 978-3-319-53018-5