

Lecture Series: Mastering Data Visualization using R

Session-2: Introduction to `ggplot2`

Jaynal Abedin

Twitter: @jaynal83

Lecture Series: Mastering Data Visualization using R

Session-2: Introduction to `ggplot2`

At the end of this session you will be able

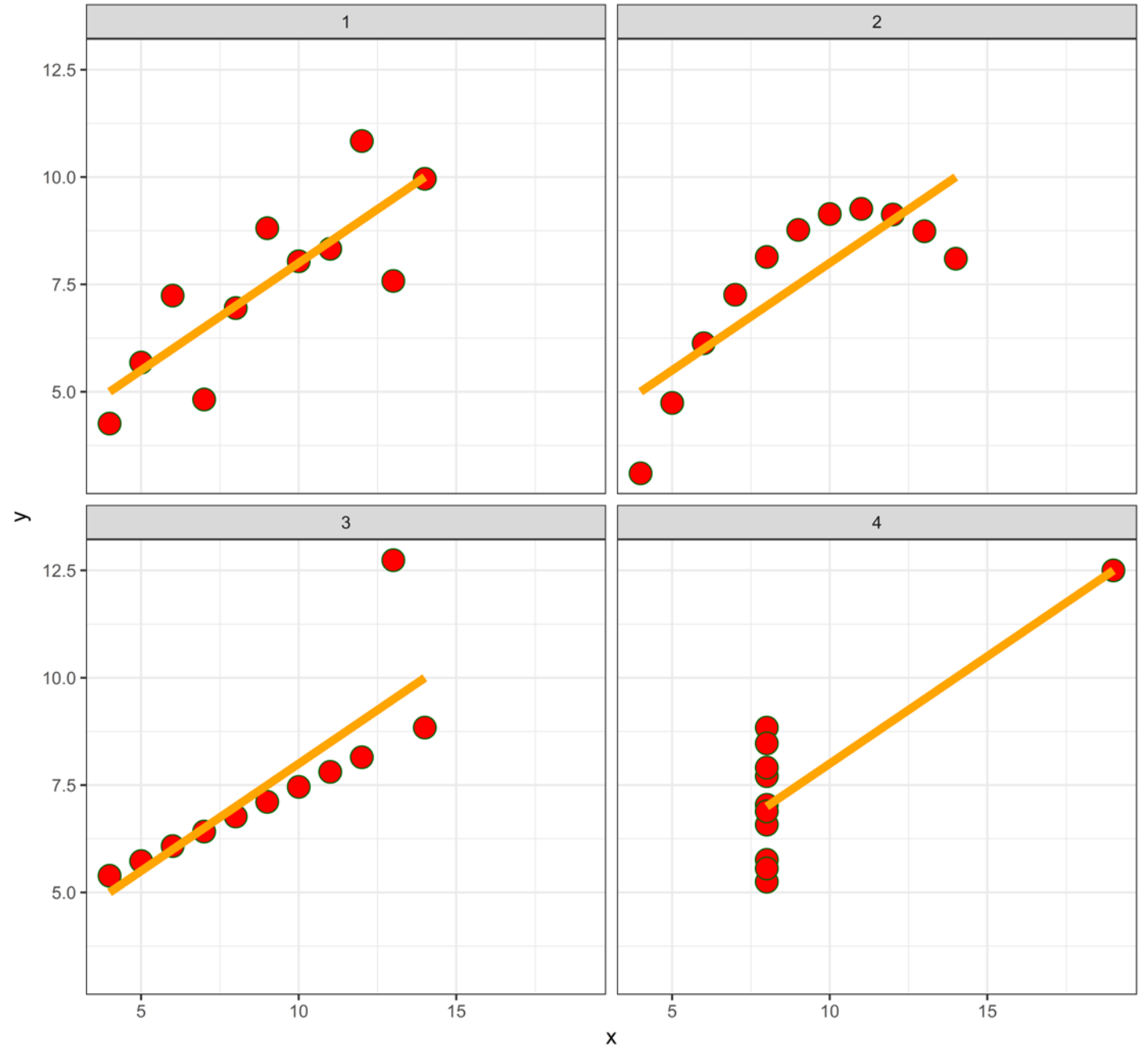
- to understand, the syntax structure of `ggplot2` library
- to understand aesthetic mapping and shape/geometry in `ggplot2`
- to create basic graphs using `ggplot2`

Start from the End

We will reproduce this graph using `ggplot2`

Anscombe Quartet

Scatter plot



What is ggplot2?

- A data visualization library in R; it is based on “grammar of graphics” that tries to talk parts (none of the bad parts) of “base” and “lattice” graphics
- This is an implementation of the book “Grammar of Graphics” by Leland Wilkinson, this library implemented by Hadley Wickham in 2005
- It defines structuring mathematic and aesthetic elements into a meaningful graph
- It uses layered concept of graphing to build component by component in each layer, rather than producing premade graphics
- Users can create their own visualization based on their concept and it is flexible enough to create any type of graphs from the data

What is ggplot2? (cont.)

- It breaks the data visualization into semantic components such as:
 - Data
 - Aesthetic mapping
 - Geometric object
 - Statistical transformations
 - Scales
 - Coordinate system
 - Position adjustments
 - Faceting

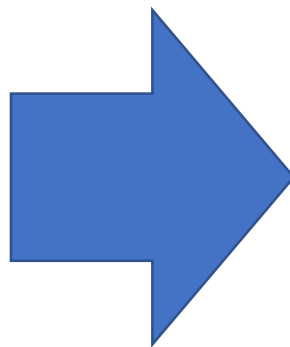
How ggplot2 works?

- It always take primary input as a dataframe, ggplot2 is not designed to take a vector input
- The necessary variables should be part of the dataframe
- User can add layer to enhance the visualization
- The first and mandatory function is `ggplot()`, without calling it you cannot create a plot in ggplot2 framework

```
ggplot(data = inputdataframe, aes(x = xaxis, y = yaxis))
```

Anscombe Quartet

I		II		III		IV	
x	y	x	y	x	y	x	y
10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89



File Home Insert Page Layout			
<div> <div>Cut Copy Paste Format Painter</div> <div>Clipboard</div> </div>			
Calibri B I U			
A1			
	A	B	C
1	x	y	group
2	10	8.04	1
3	8	6.95	1
4	13	7.58	1
5	9	8.81	1
6	11	8.33	1
7	14	9.96	1
8	6	7.24	1
9	4	4.26	1
10	12	10.84	1
11	7	4.82	1
12	5	5.68	1
13	10	9.14	2
14	8	8.14	2
15	13	8.74	2
16	9	8.77	2
17	11	9.26	2
18	14	8.1	2
19	6	6.13	2
20	4	3.1	2

Anscombe Quartet (scatter plot)

```
library(ggplot2)
```

```
xdat <- read.csv("anscombe_quartet_modified.csv")
```

```
> xdat <- read.csv("anscombe_quartet_modified.csv")
```

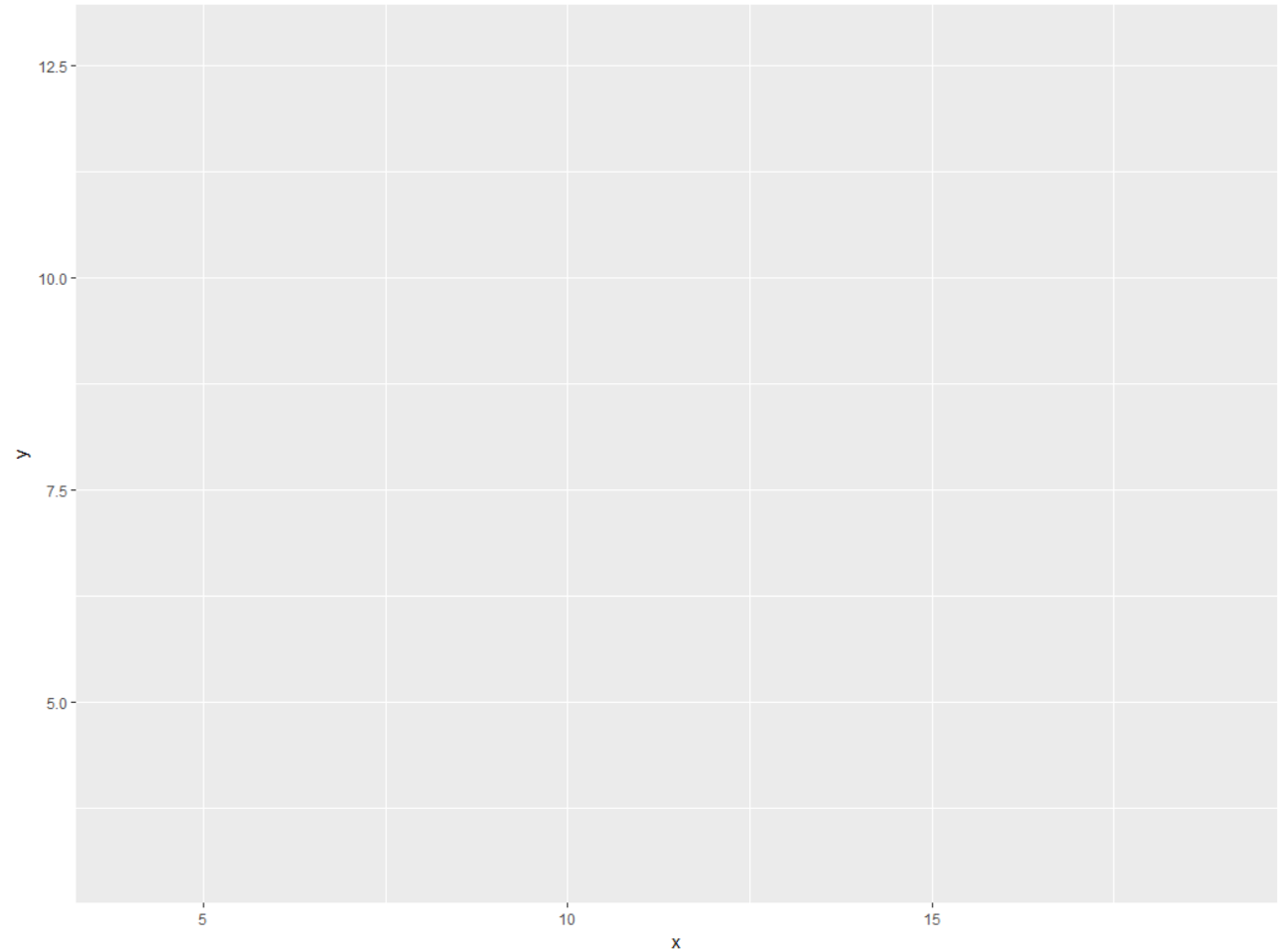
```
> head(xdat)
```

	x	y	group
1	10	8.04	1
2	8	6.95	1
3	13	7.58	1
4	9	8.81	1
5	11	8.33	1
6	14	9.96	1

```
> |
```

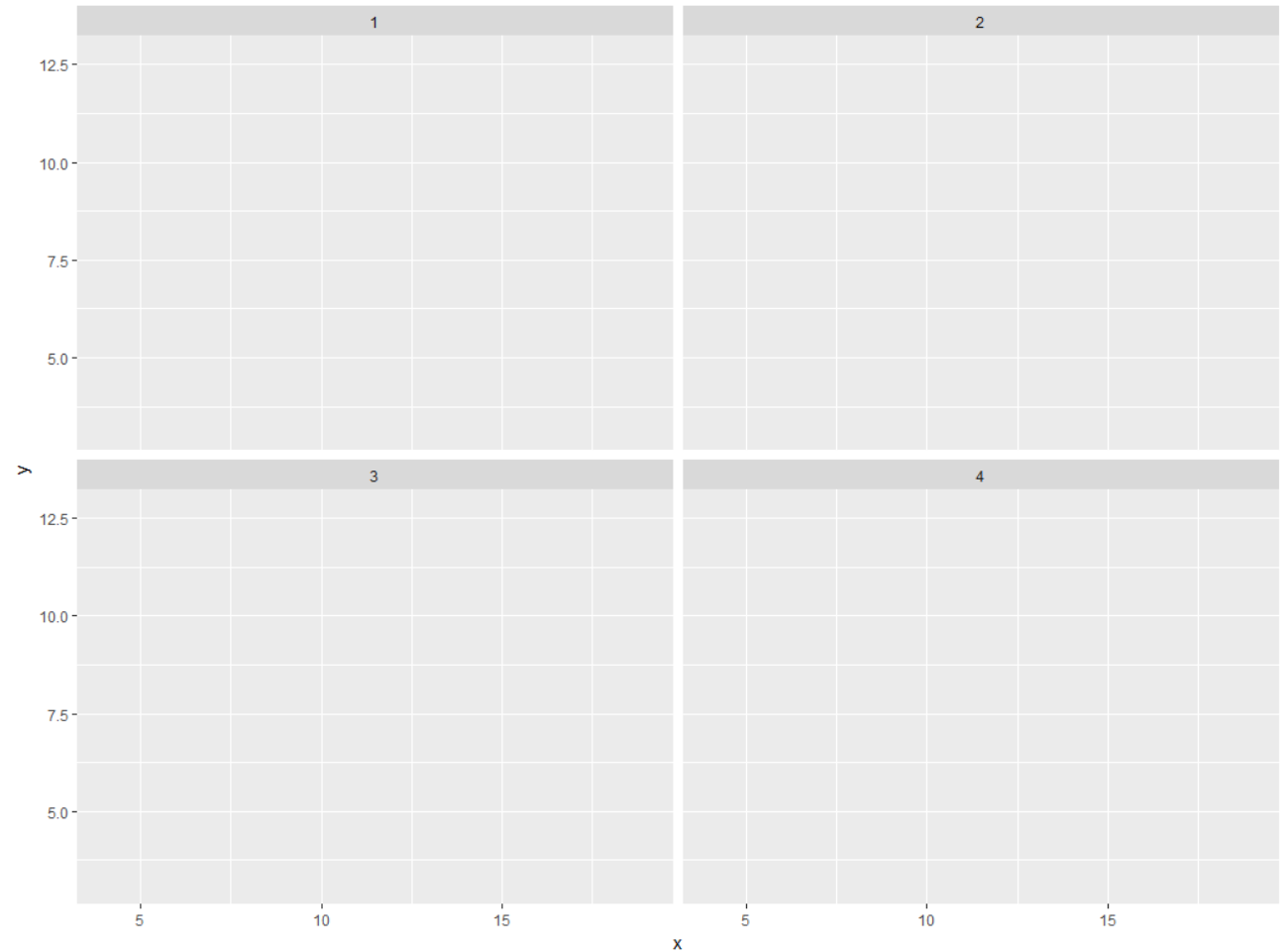

Step-1: Initialize ggplot & define aesthetic mapping

```
plot1 <- ggplot(data = xdat, aes(x=x, y=y))  
print(plot1)
```



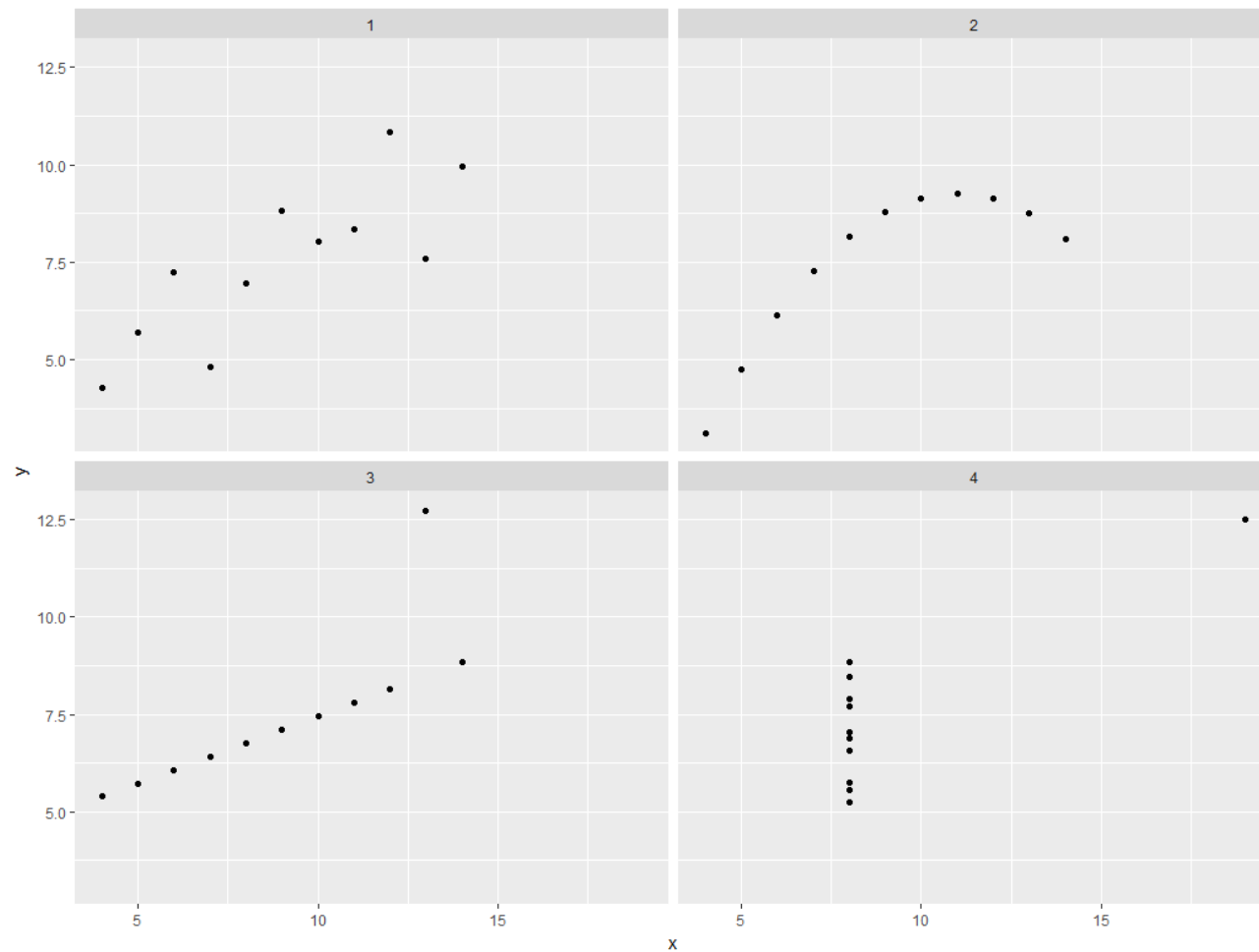
Step-2: Adding panel for each value of “group”

```
plot1 <- plot1 + facet_wrap(~group)  
print(plot1)
```



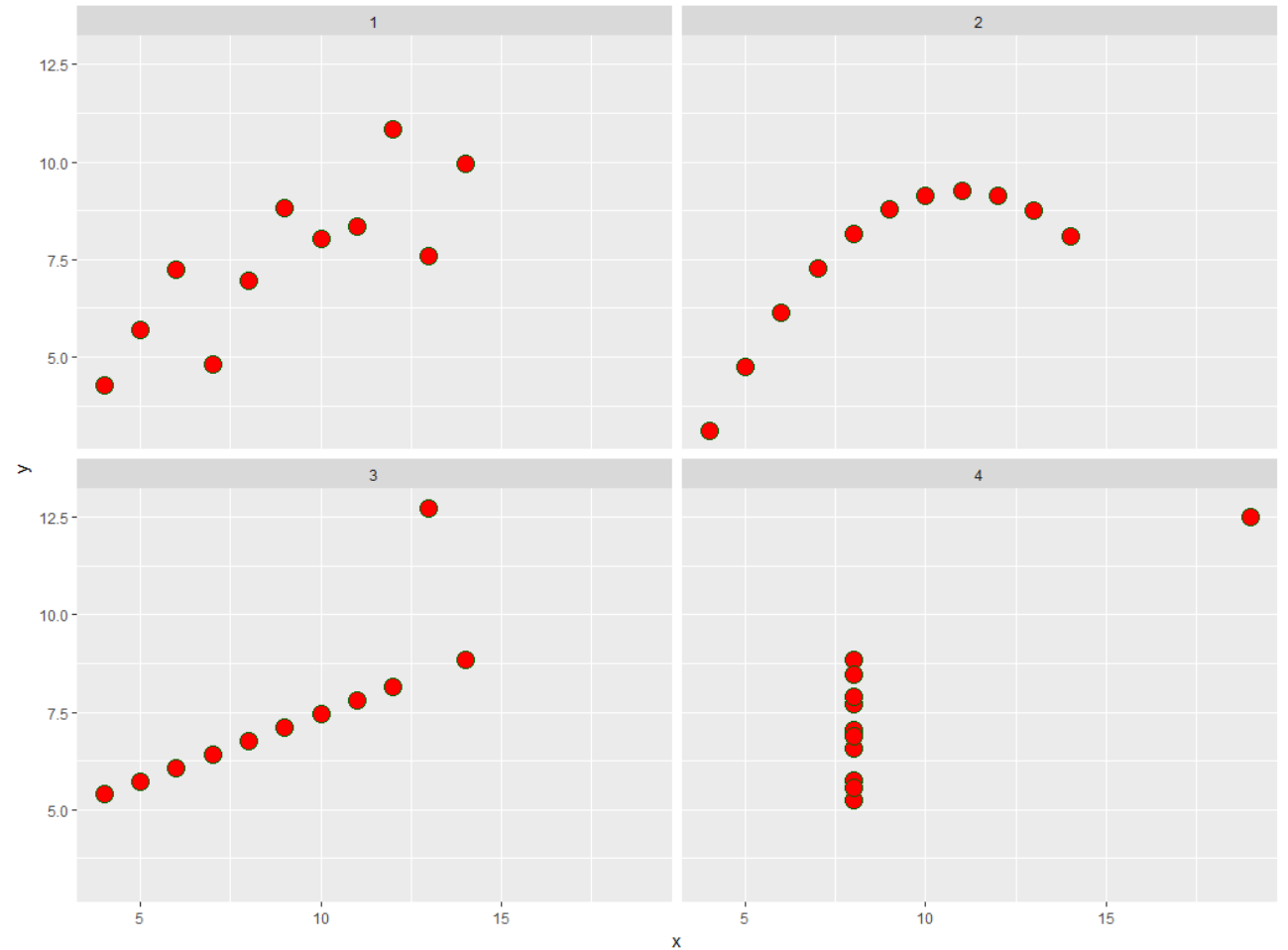
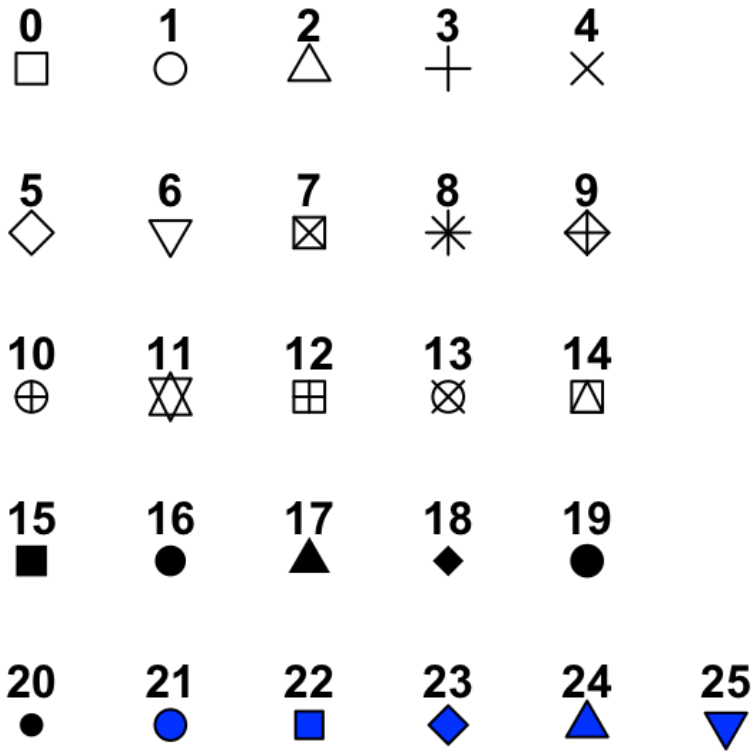
Step-3: Adding points

```
plot1 <- plot1 + geom_point()  
print(plot1)
```



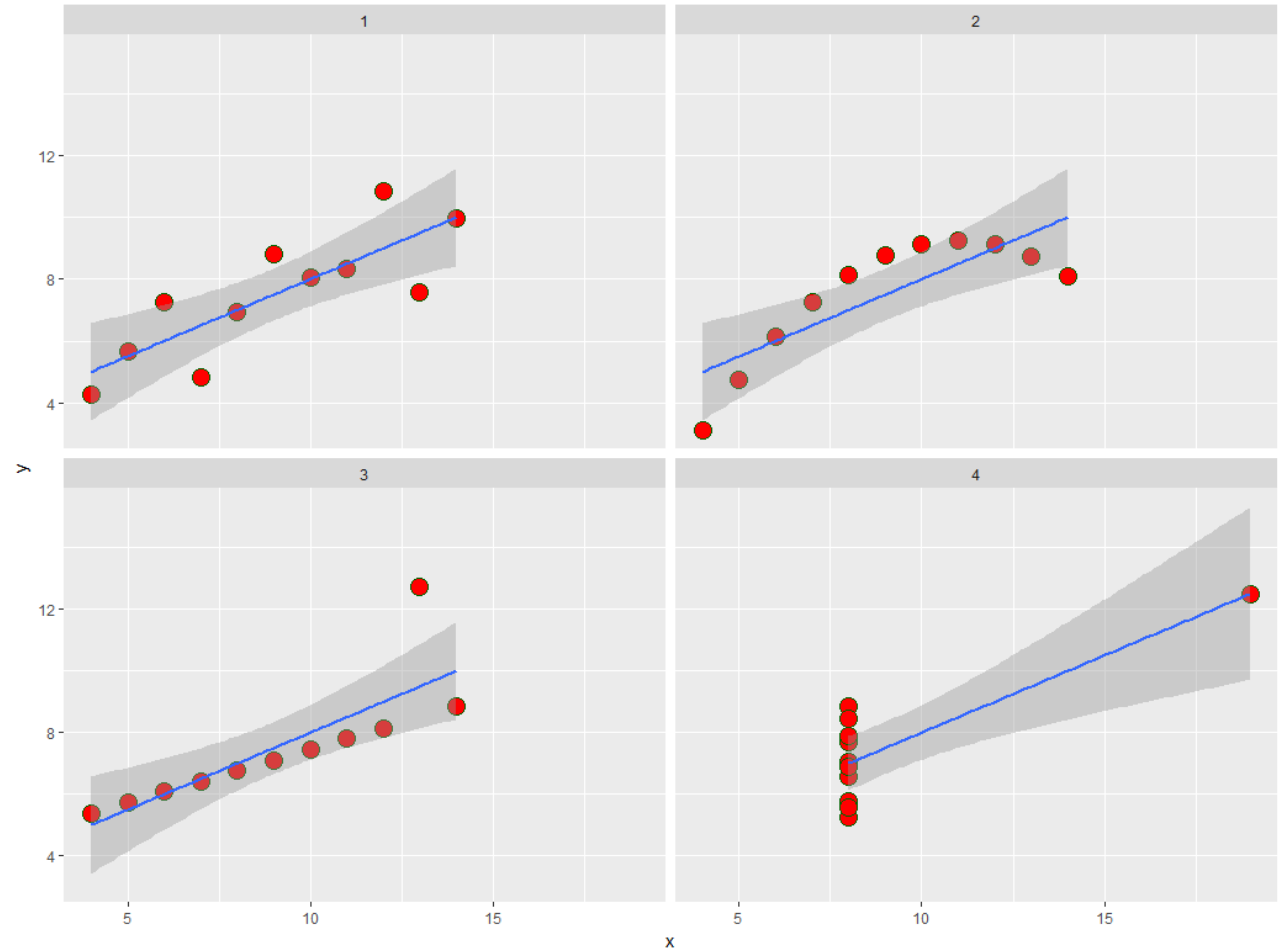
Step-4: Customize point properties

```
plot1 <- plot1 + geom_point(size = 5, fill = "red", color = "darkgreen", shape = 21)  
print(plot1)
```



Step-5: Adding a new layer with linear fitted line

```
plot1 <- plot1 + geom_smooth(method = "lm")  
print(plot1)
```



Step-6: Remove confidence interval of the line and customize it

```
plot1 <- plot1 + geom_smooth(method = "lm",  
                             se = F, linetype = 1, size = 2, col = "orange")  
print(plot1)
```

0. 'blank'

1. 'solid'

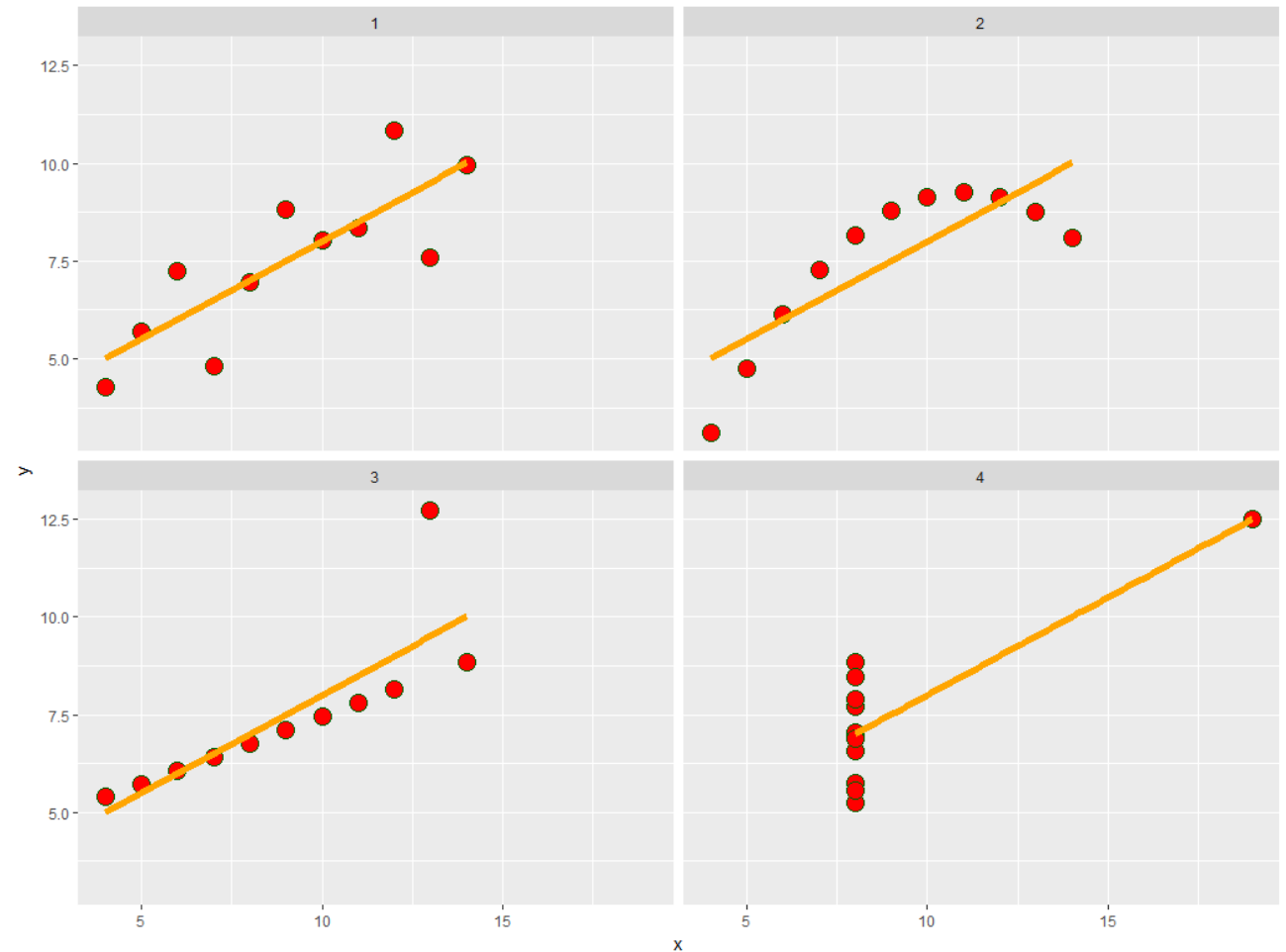
2. 'dashed'

3. 'dotted'

4. 'dotdash'

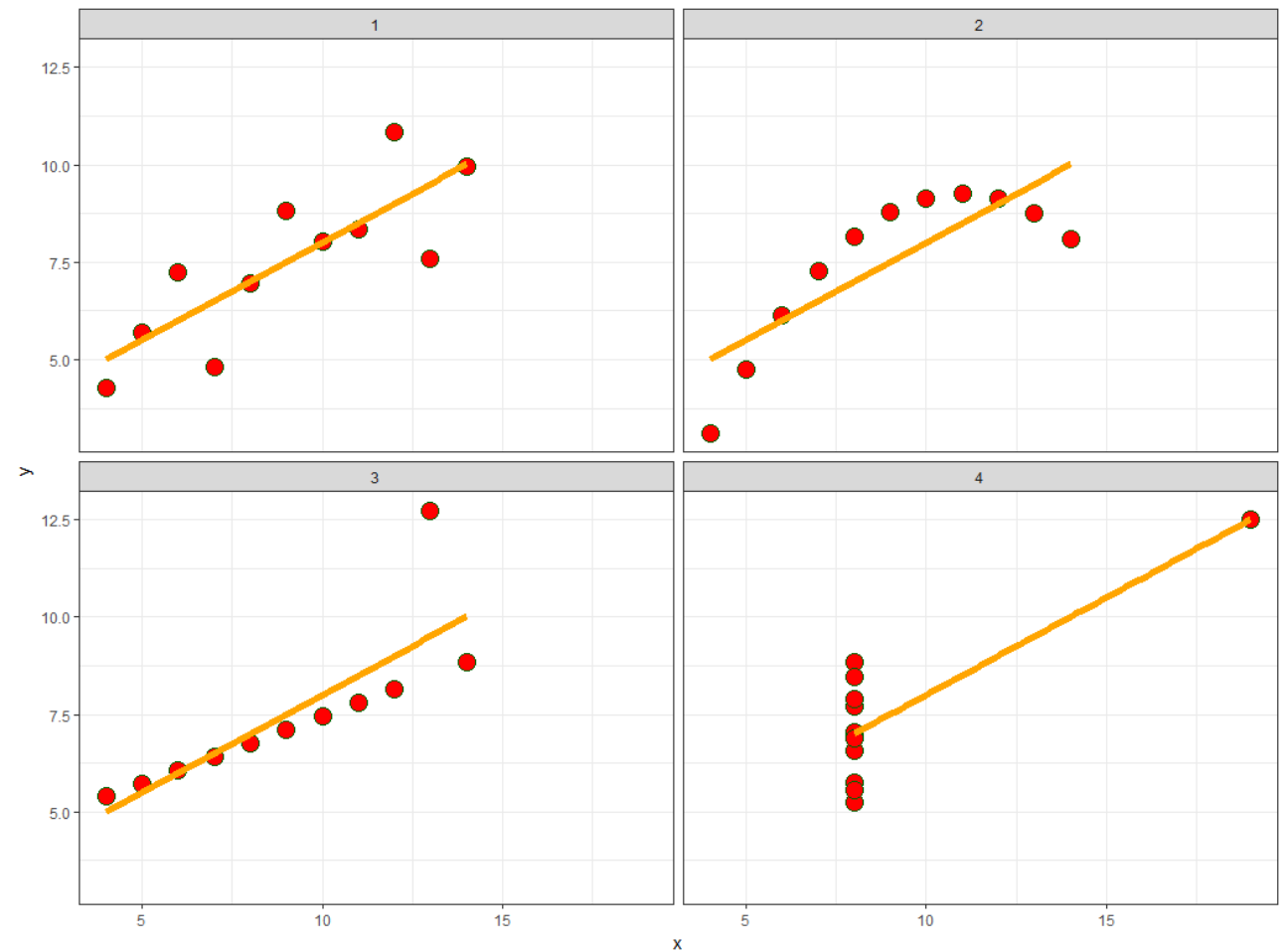
5. 'longdash'

6. 'twodash'



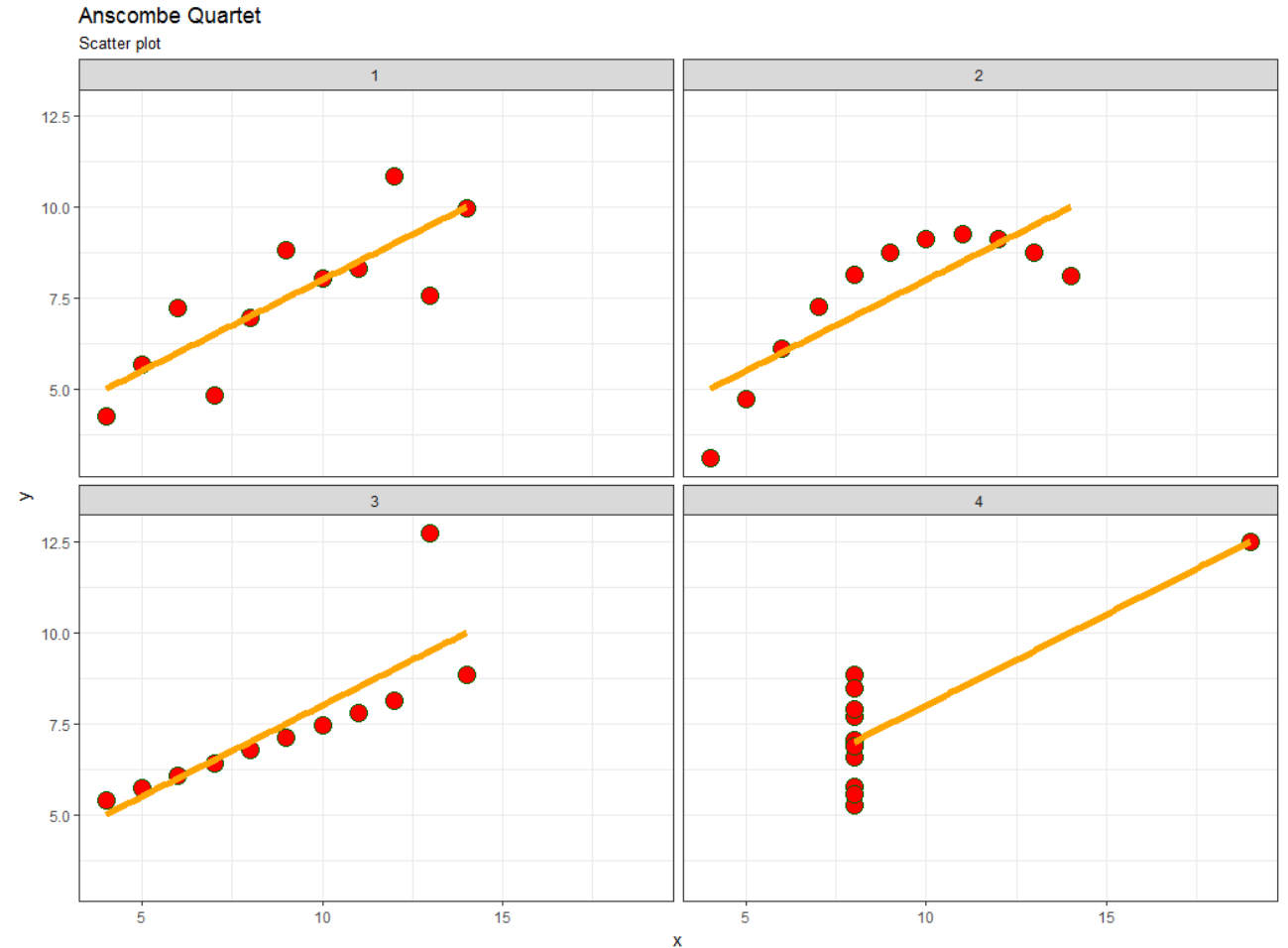
Step-7: Remove Background Colour

```
plot1 <- plot1 + theme_bw()  
print(plot1)
```



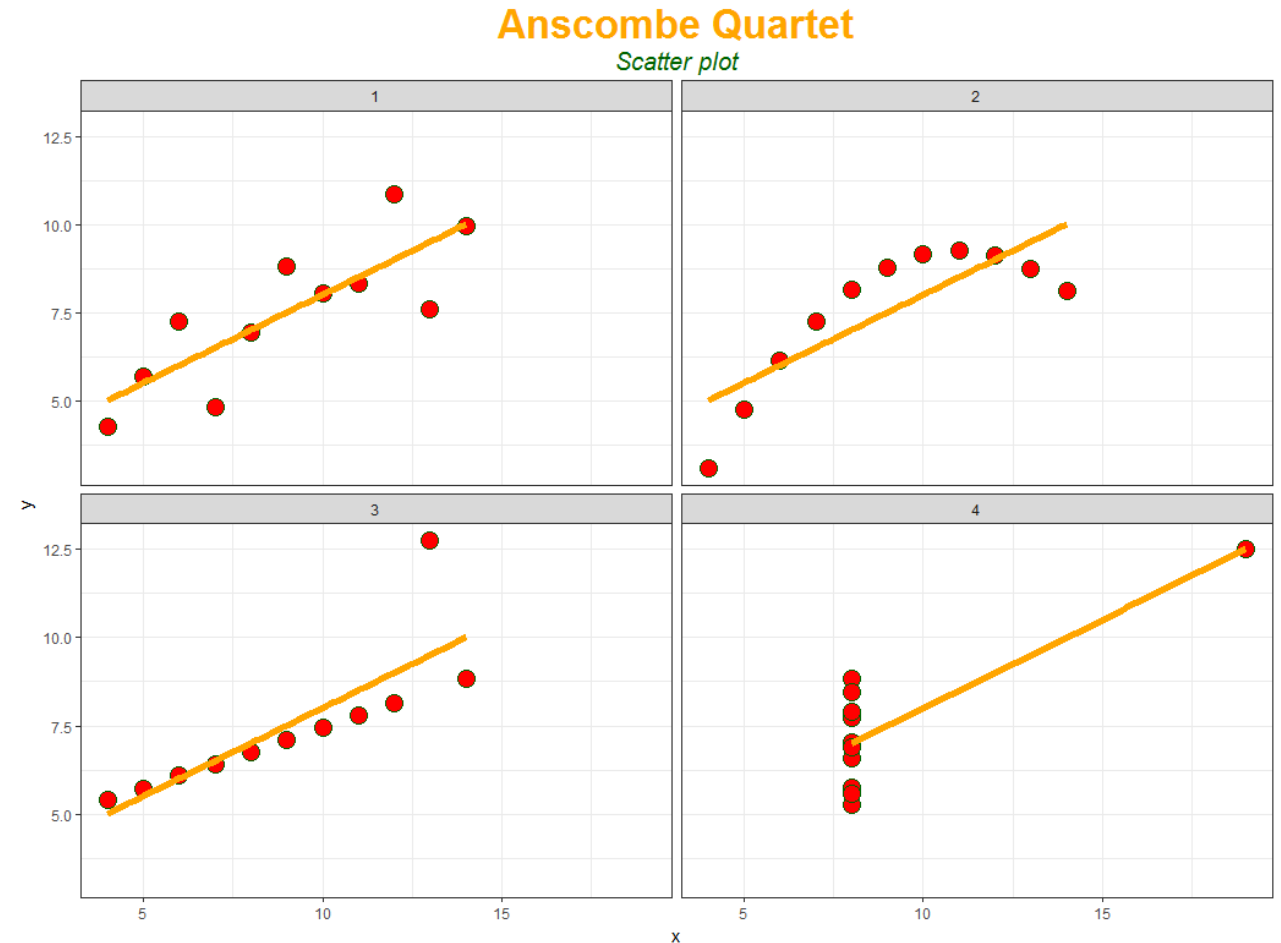
Step-8: Adding Plot Title and Sub-Title

```
plot1 <- plot1 + ggtitle(label = "Anscombe Quartet",  
                          subtitle = "Scatter plot")  
print(plot1)
```



Step-9: Customize Title and Sub-Title

```
plot1 <- plot1 + theme(  
  plot.title = element_text(  
    size = 25,  
    face = "bold",  
    color = "orange",  
    hjust = 0.5),  
  plot.subtitle =  
  element_text(  
    size = 15,  
    face = "italic",  
    colour = "darkgreen",  
    hjust = 0.5)  
)  
print(plot1)
```

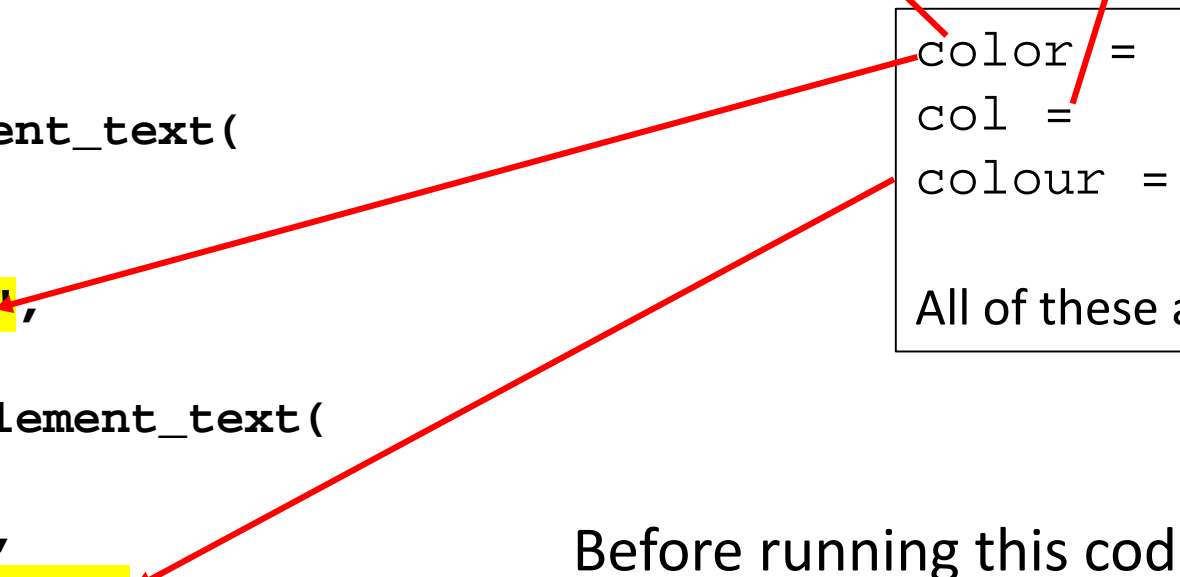


Step-10: Export as High Quality PNG Image

```
png("myplot.png",res = 1500, height = 8, width = 8, units = "in")  
print(plot1)  
dev.off()
```

- **res = 1500** is for high quality, you can use other number, large number for higher quality
- **height** and **width** is the output file or image size
- **units** represents the unit of height and width, inch/pixel etc

```
png("plot1.png", res = 1500, height = 8, width = 8, units = "in")
ggplot(data = xdat, aes(x=x, y=y))+
  facet_wrap(~group)+
  geom_point(size = 5, fill = "red", color = "darkgreen", shape = 21)+
  geom_smooth(method = "lm", se = F, linetype = 1, size = 2, col = "orange")+
  theme_bw()+
  ggtitle(
    label = "Anscombe Quartet",
    subtitle = "Scatter plot"
  )+
  theme(
    plot.title = element_text(
      size = 25,
      face = "bold",
      color = "orange",
      hjust = 0.5),
    plot.subtitle = element_text(
      size = 15,
      face = "italic",
      colour = "darkgreen",
      hjust = 0.5)
  )
dev.off()
```



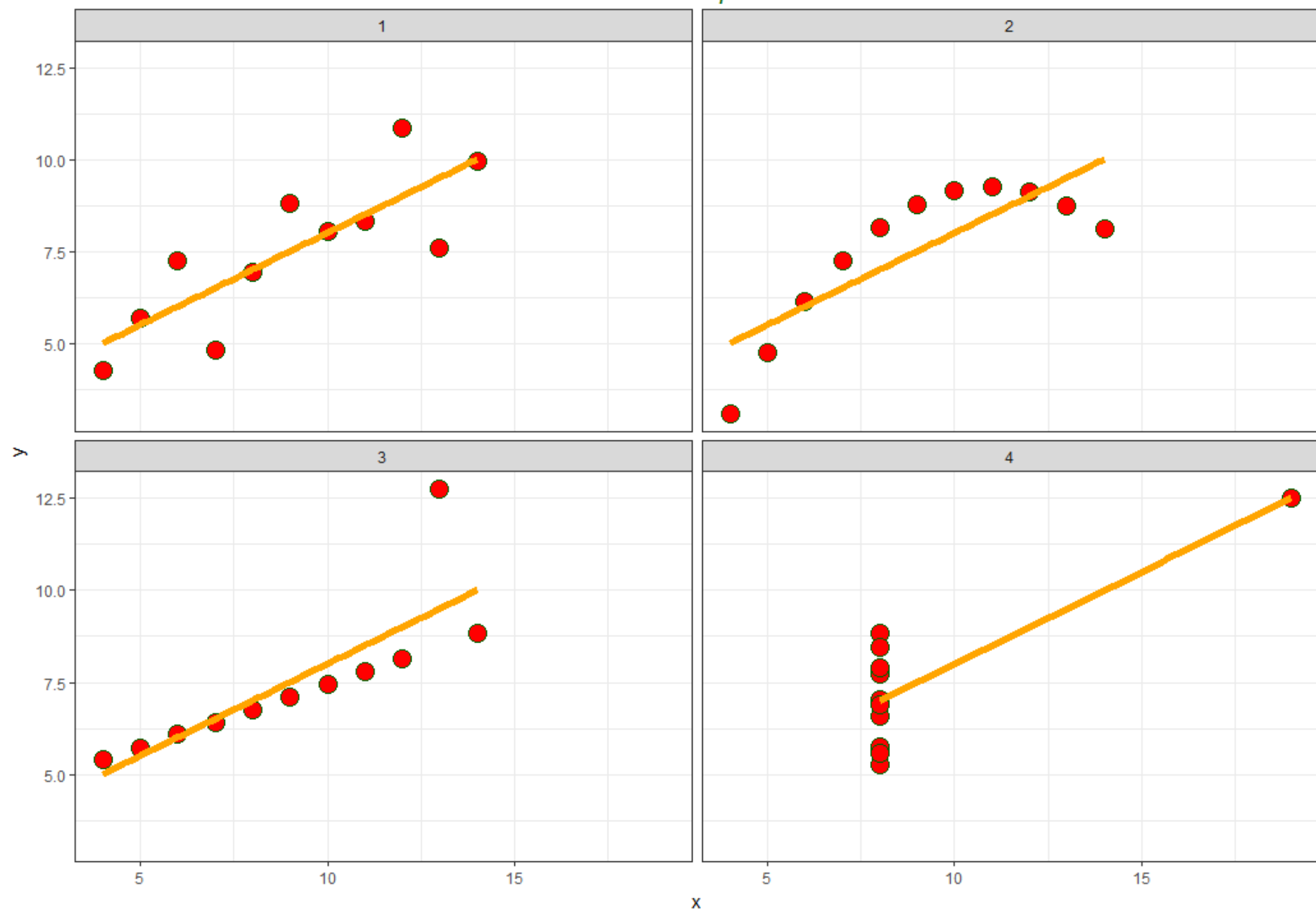
color =
col =
colour =

All of these are for same thing

Before running this code, make sure you imported the required data and stored in a dataframe (**xdat**)

Anscombe Quartet

Scatter plot



Materials to study further

- [Book]: ggplot2: Elegant Graphics for Data Analysis, Hadley Wickham, 2009

Lecture Series: Mastering Data Visualization using R

Session-**3**: Aesthetic Mapping and Geometric Objects in `ggplot2`

At the end of this session you will be able

- to understand aesthetic mapping and shape/geometry in `ggplot2`
- to work with `aes()` and `geom_*()` within `ggplot2` framework

Date: **21 Dec 2017**

Time: 10:00 pm Bangladesh

Duration: 30 min

