Legends

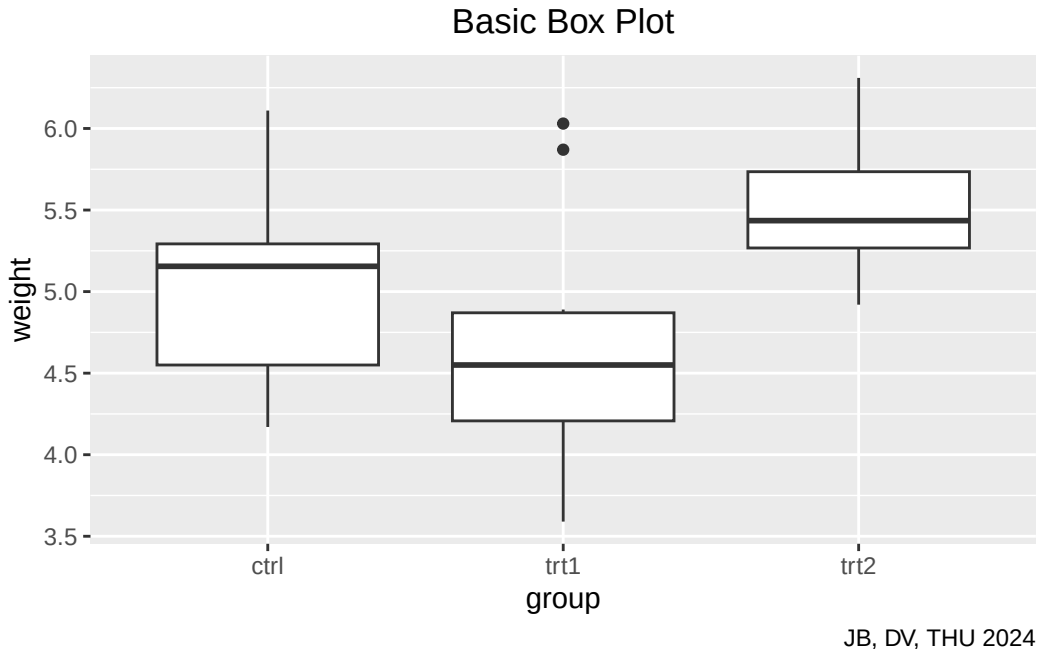
JB

Table of contents

1 Basic Box Plot

- `library(ggplot2)`: This loads the ggplot2 library.
- `ggplot(PlantGrowth, aes(x = group, y = weight))`:
  - Uses the built-in `PlantGrowth` dataset
  - Creates a plot with `group` on the x-axis and `weight` on the y-axis
  - `aes()` defines the aesthetic mappings of the plot
- `geom_boxplot()`‘: Adds a box plot geometry to visualize the distribution of weights across different groups
- `ggtitle('Basic Box Plot')`: Adds a title to the plot
- `theme(plot.title = element_text(hjust=0.5))`: Centers the title horizontally `(hjust=0.5)`

```
library(ggplot2)
ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  labs(title = 'Basic Box Plot', caption = 'JB, DV, THU 2024') +
  theme(plot.title = element_text(hjust=0.5))
```



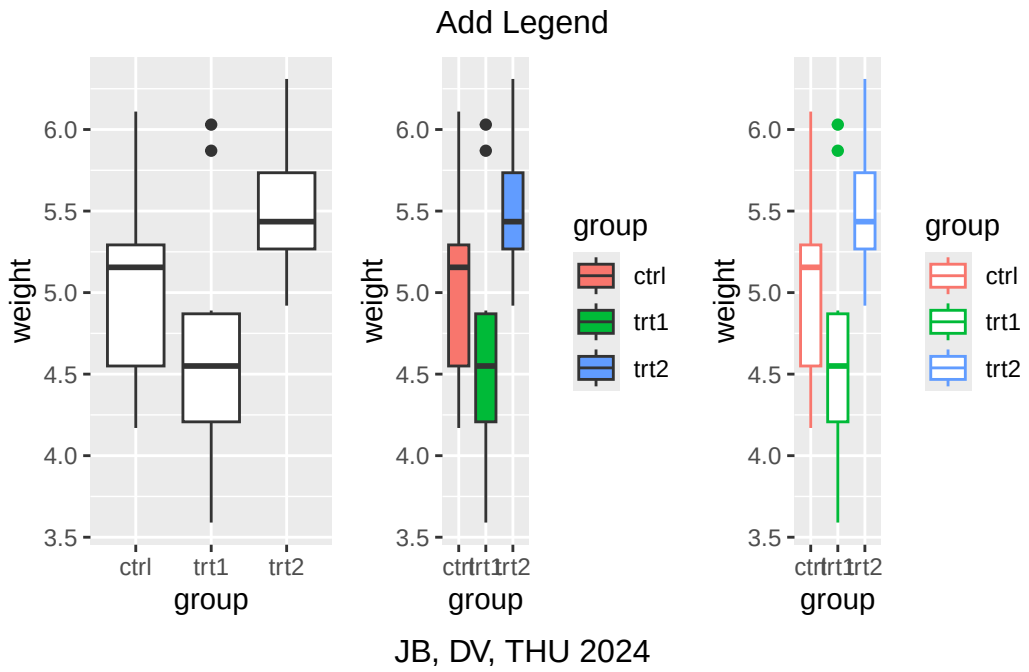
2 Add Legend

By adding `fill` or `color` options to `aes()`, a legend is created automatically on the right side of the plot.

- `library(gridExtra)`: This loads the `gridExtra` library to use `grid.arrange`
- `fill`: Fill in the `colors` inside the chart
- `color`: Change the `color` of the outlines of the chart
- `grid.arrange`: Arrange the order of the chart
- `top`: Add `text` on the top of the chart
- `bottom`: Add `text` on the bottom of the chart

```
library(gridExtra)

p1 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot()
p2 <- ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot()
p3 <- ggplot(PlantGrowth, aes(x = group, y = weight, color = group)) +
  geom_boxplot()
grid.arrange(p1,p2,p3,ncol=3,top='Add Legend', bottom = 'JB, DV, THU 2024')
```

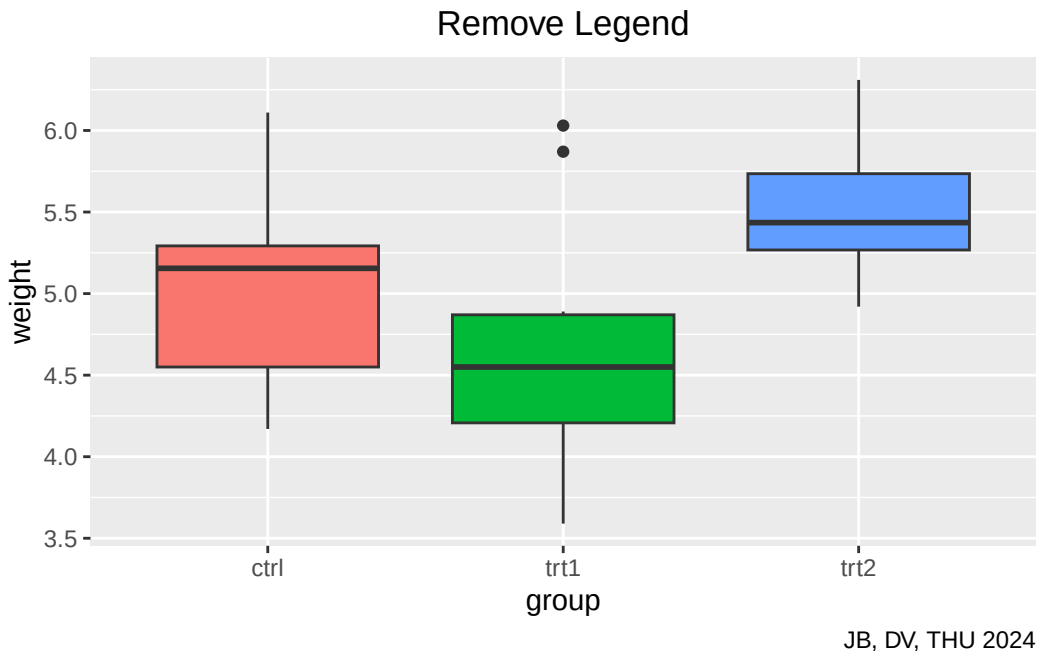


3 Remove Legend

The legend is removed by `guides(fill = FALSE)`

```
ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  guides(fill = FALSE) +
  ggtitle('Remove Legend') +
  labs(caption = 'JB, DV, THU 2024') +
  theme(plot.title = element_text(hjust=0.5))
```

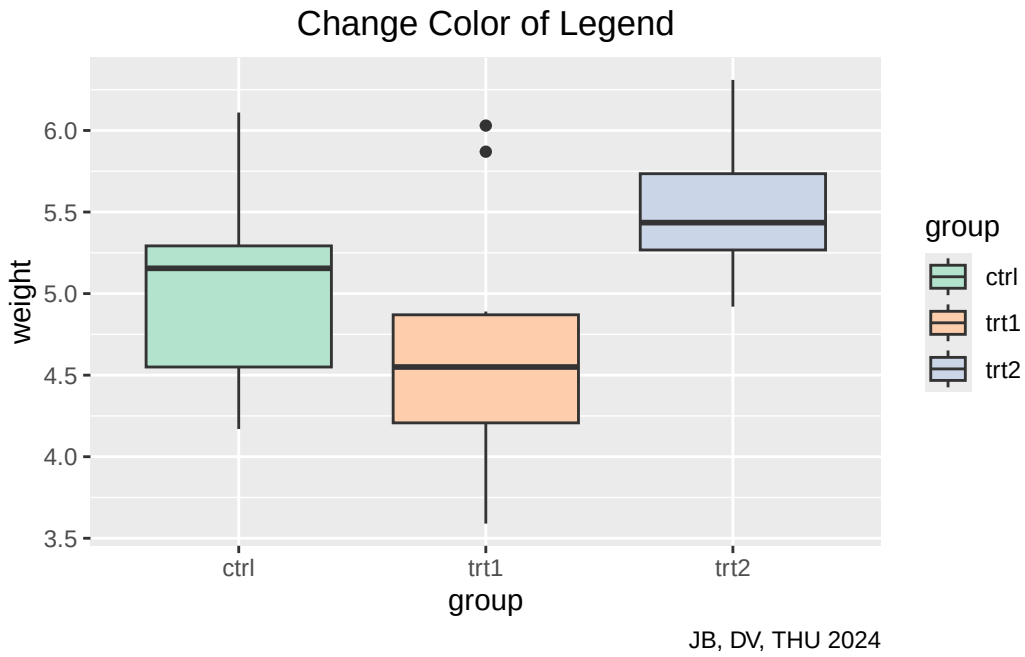
```
Warning: The `<scale>` argument of `guides()` cannot be `FALSE`. Use "none" instead as
of ggplot2 3.3.4.
```



4 Change Color of Legend

`scale_fill_brewer(palette = "Pastel2")` is used to change the color of the legend

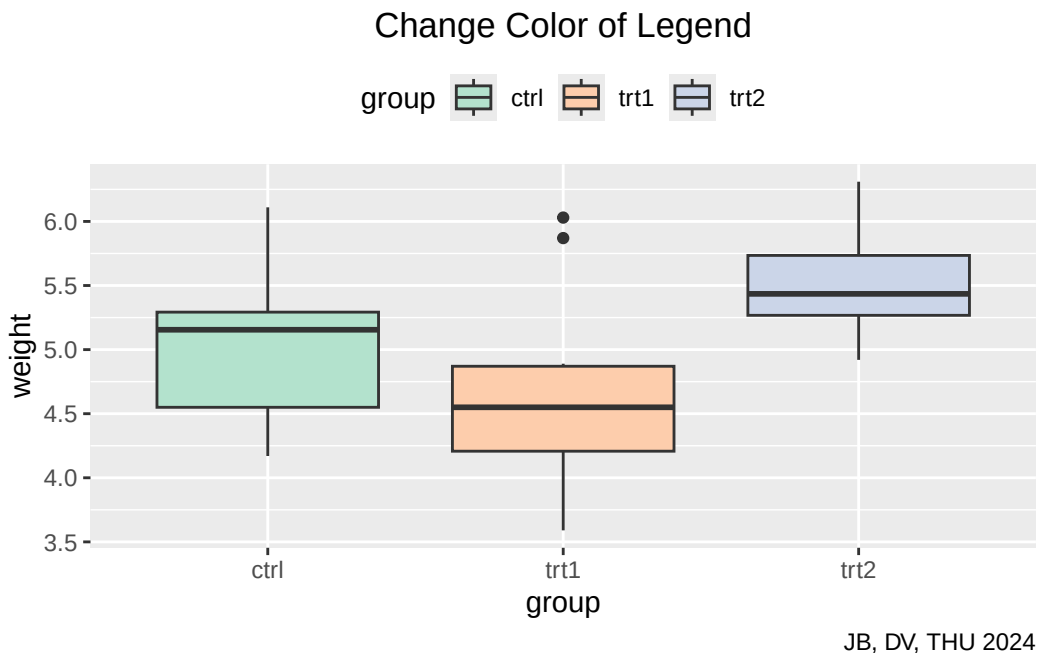
```
ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  scale_fill_brewer(palette = "Pastel2") +
  ggtitle('Change Color of Legend') +
  labs(caption = 'JB, DV, THU 2024') +
  theme(plot.title = element_text(hjust=0.5))
```



5 Change Position of Legend

`legend.postion` in `theme` is used to change the position of the legend

```
ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  scale_fill_brewer(palette = "Pastel2") +
  ggtitle('Change Color of Legend') +
  labs(caption = 'JB, DV, THU 2024') +
  theme(plot.title = element_text(hjust=0.5),
        legend.position = 'top')
```

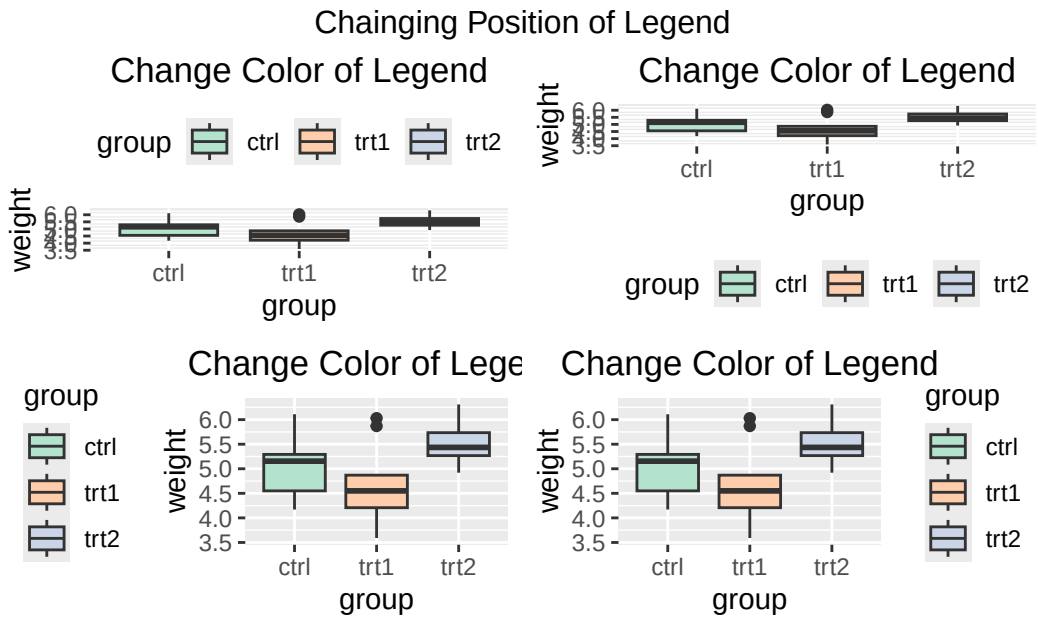


6 Practice

- Change legend position to top, bottom, left, and right
  - `legend.position = 'top'`: To change the position of the legend to `top`
  - `legend.position = 'bottom'`: To change the position of the legend to `bottom`
  - `legend.position = 'left'`: To change the position of the legend to `left`
  - `legend.position = 'right'`: To change the position of the legend to 'right"

```
p1 <- ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  scale_fill_brewer(palette = "Pastel2") +
  ggtitle('Change Color of Legend') +
  theme(plot.title = element_text(hjust=0.5),
        legend.position = 'top')
p2 <- p1 + theme(legend.position = 'bottom')
p3 <- p1 + theme(legend.position = 'left')
p4 <- p1 + theme(legend.position = 'right')

grid.arrange(p1, p2, p3, p4, ncol=2,
             top='Chainging Position of Legend', bottom = 'JB, DV, THU 2024')
```

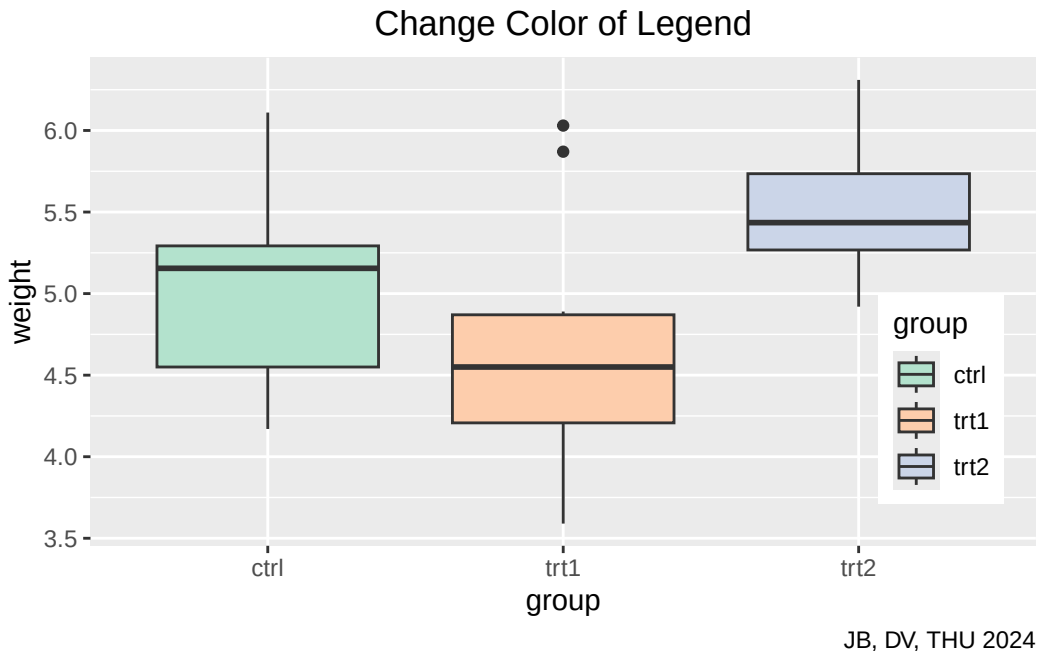


7 Change Position of Legend

- Using the `legend.position` line we are able to change the position the legend, and using the `c(x, y)` to set the position
  - the graph below shows the legend position in `c(.9, .3)`

```
ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  scale_fill_brewer(palette = "Pastel2") +
  ggtitle('Change Color of Legend') +
  labs(caption = 'JB, DV, THU 2024') +
  theme(plot.title = element_text(hjust=0.5),
        legend.position = c(.9, .3))
```

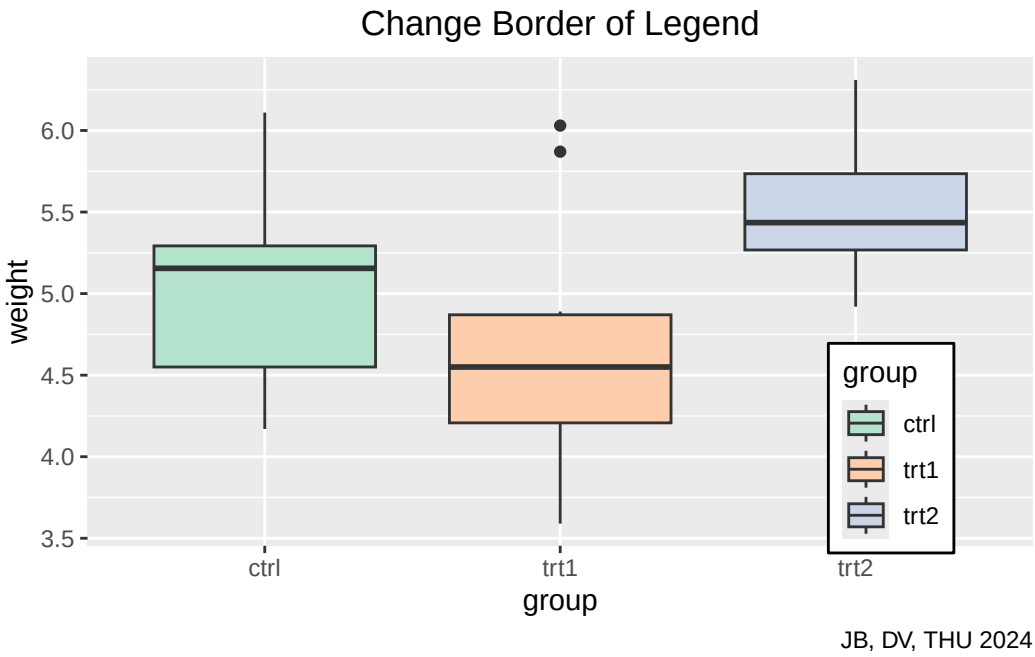
```
Warning: A numeric `legend.position` argument in `theme()` was deprecated in ggplot2
3.5.0.
i Please use the `legend.position.inside` argument of `theme()` instead.
```



8 Change Border of Legend

Using `legend.background` line, we are able to change the `color` of the border using `color`

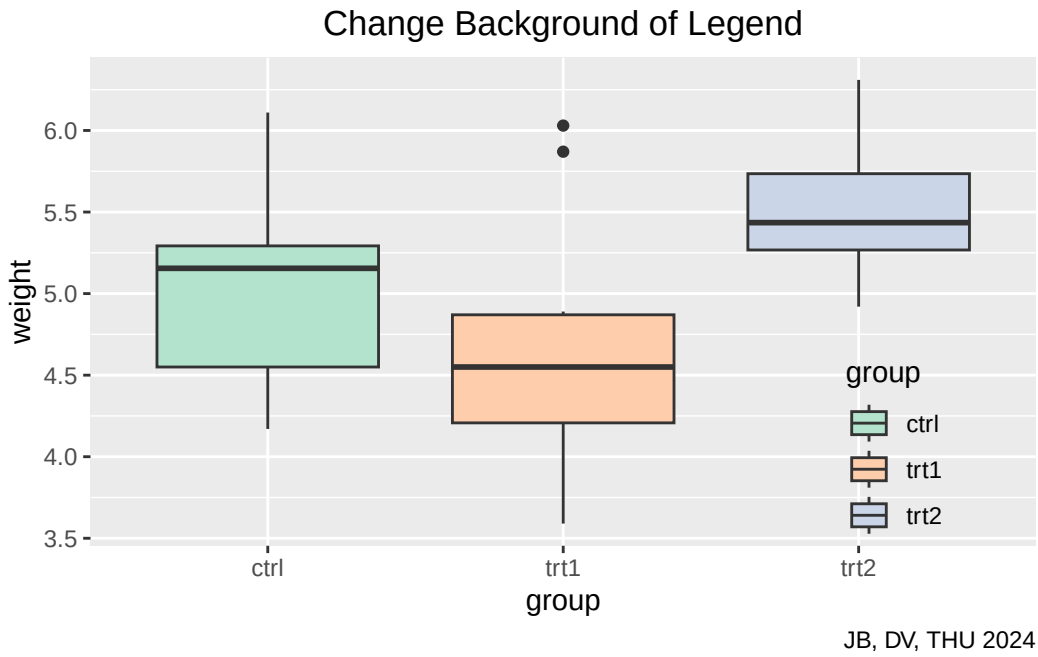
```
ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  scale_fill_brewer(palette = "Pastel2") +
  ggtitle('Change Border of Legend') +
  labs(caption = 'JB, DV, THU 2024') +
  theme(plot.title = element_text(hjust=0.5),
        legend.position = c(0.85, 0.2),
        legend.background = element_rect(fill = 'white',
                                         color = 'black'))
```



9 Change Background of Legend

As previously mentioned, by using `fill` we can change the `background` of the legend

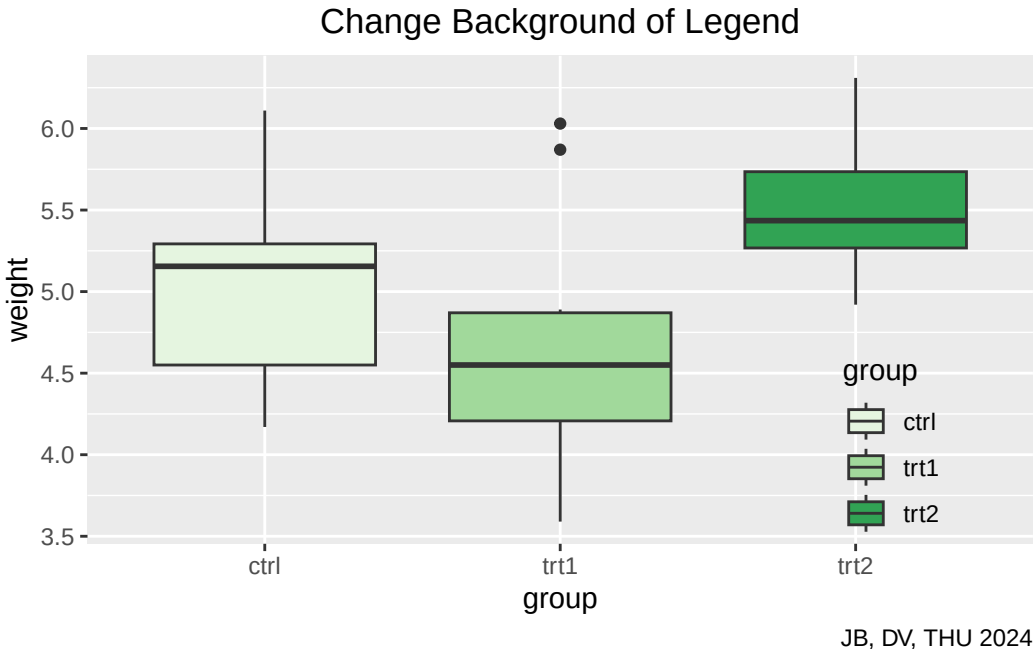
```
ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  scale_fill_brewer(palette = "Pastel2") +
  ggtitle('Change Background of Legend') +
  labs(caption = 'JB, DV, THU 2024') +
  theme(plot.title = element_text(hjust=0.5),
        legend.position = c(0.85, 0.2),
        legend.background = element_blank(),
        legend.key = element_blank())
```



Optionally, we can also change the colors using the Palette provided by from `ColorBrewer` provided in the `ggplot2`. Changing the `palette` will change the color of the chart

```
ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  scale_fill_brewer(palette = "Pastel6") +
  ggtitle('Change Background of Legend') +
  labs(caption = 'JB, DV, THU 2024') +
  theme(plot.title = element_text(hjust=0.5),
        legend.position = c(0.85, 0.2),
        legend.background = element_blank(),
        legend.key = element_blank())
```

```
Warning: Unknown palette: "Pastel6"
```



The `palette` provided are

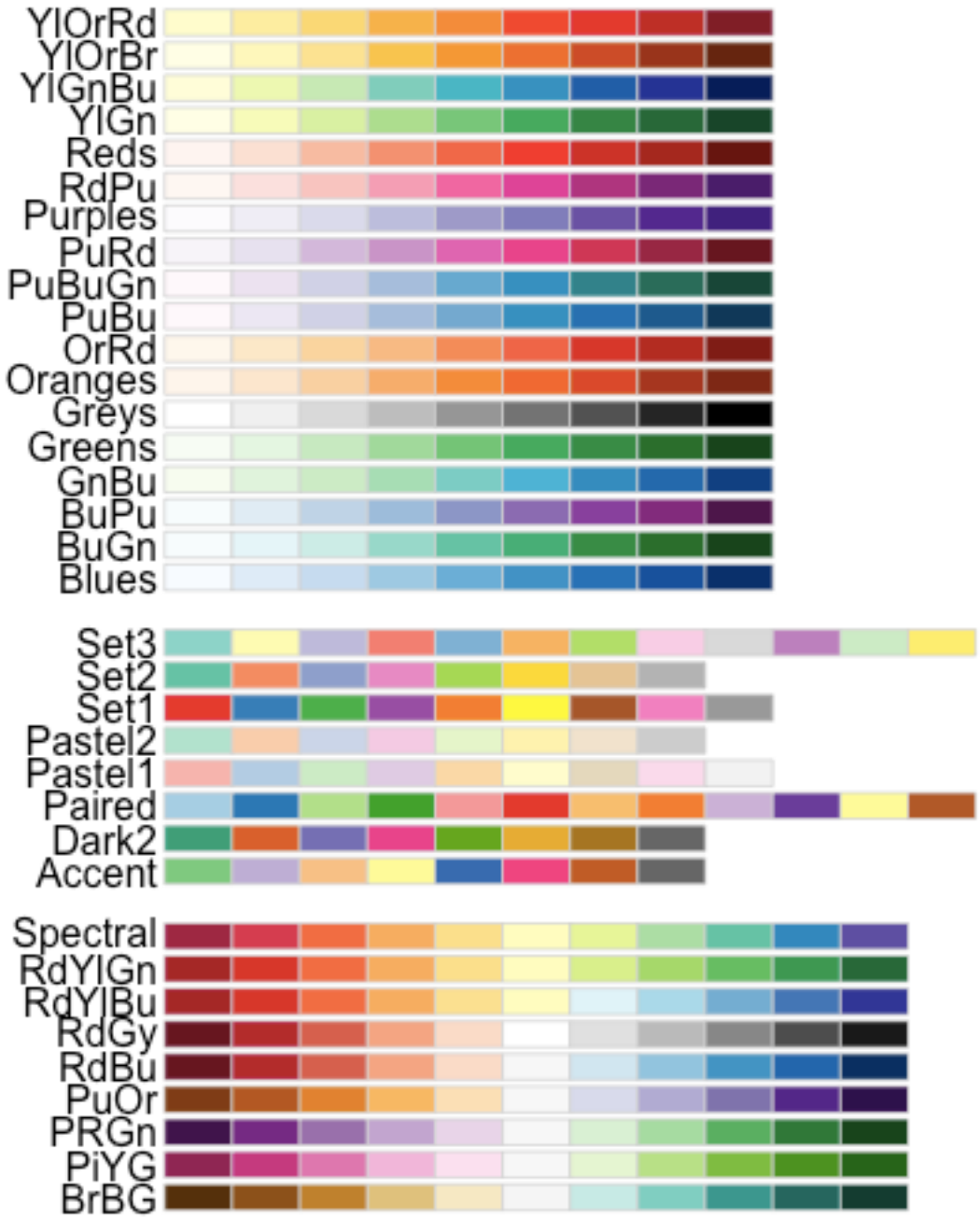


Figure 1: ggplot2 Palette

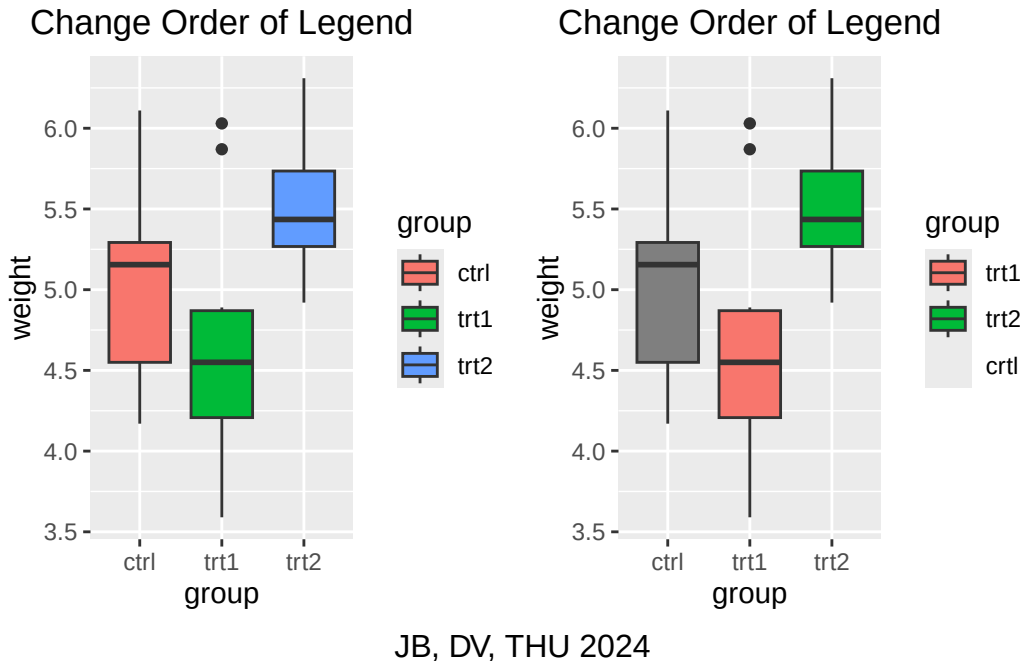
10 Change Order of Items in Legend

To change the order of the items in Legend, we can use `scale_fill_discrete` to add limits so that the items will be limited to the `order` that we want

```
p1 <- ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  ggtitle('Change Order of Legend') +
  theme(plot.title = element_text(hjust=0.5))

p2 <- p1 + scale_fill_discrete(limits = c('trt1', 'trt2', 'crtl'))

grid.arrange(p1, p2, ncol = 2, bottom = 'JB, DV, THU 2024')
```



11 Reverse Order of Items in Legend

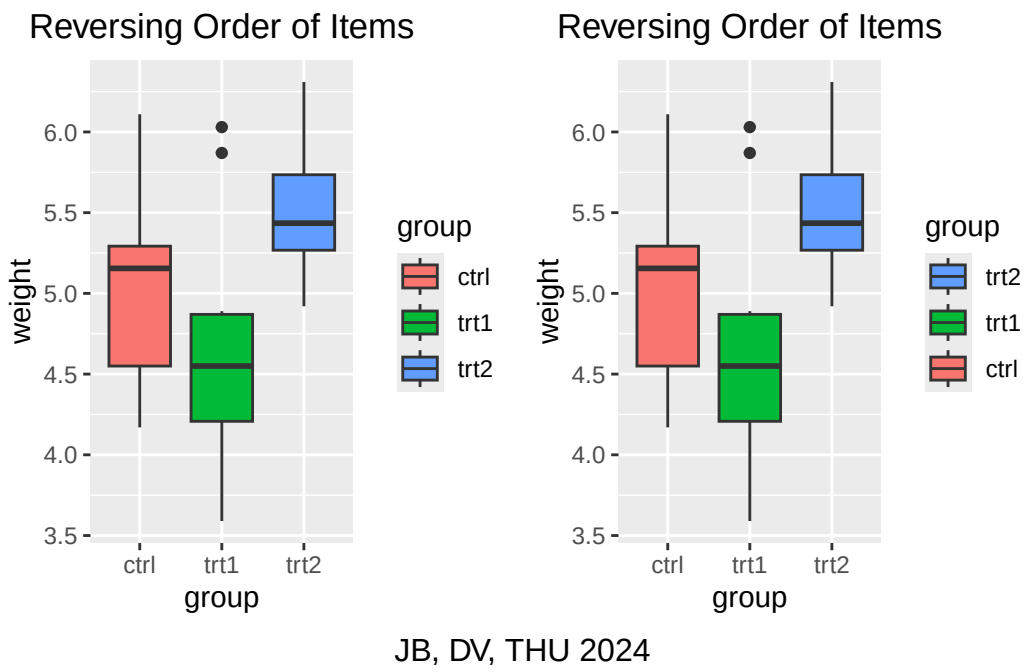
To reverse the order of Items in Legend we use the line `guide_legend(reverse = TRUE)`

```
p1 <- ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  ggtitle('Reversing Order of Items') +
```

```
  theme(plot.title = element_text(hjust=0.5))

p2 <- p1 + guides(fill = guide_legend(reverse = TRUE))

grid.arrange(p1, p2, ncol = 2, bottom = 'JB, DV, THU 2024')
```



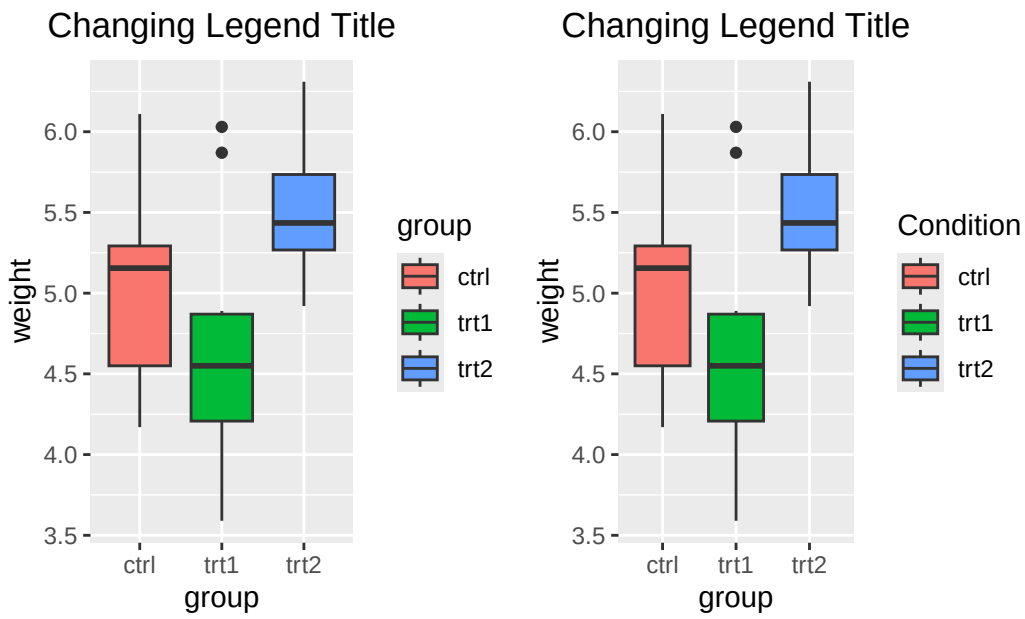
12 Change Legend Title

To change the Legend Title, we can use `plot.title` in `theme` and adding the `fill = 'Legend Title'`

```
p1 <- ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  ggtitle('Changing Legend Title') +
  theme(plot.title = element_text(hjust=0.5))

p2 <- p1 + labs(fill = 'Condition')

grid.arrange(p1, p2, ncol = 2, bottom = 'JB, DV, THU 2024')
```

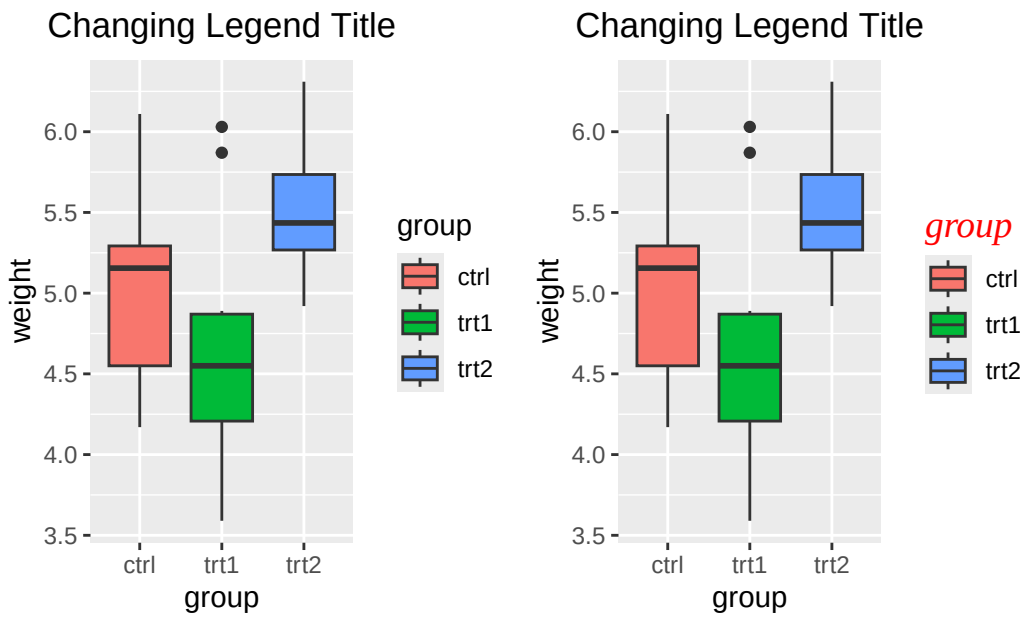


13 Change Appearance of Legend Title

- To change the appearance, we can add `legend.title` in the `theme` section and change the following:
  - `face`: To change style of the text (`bold`, `italic`, `underline`)
  - `family`: To change the Font Style
  - `color`: To change the `color` of the text
  - `size`: To change the `size` of the text

```
p1 <- ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  ggtitle('Changing Legend Title') +
  theme(plot.title = element_text(hjust=0.5))

p2 <- p1 + theme(legend.title = element_text(face = 'italic',
                                             family = 'Times',
                                             color = 'red',
                                             size = 14))
grid.arrange(p1, p2, ncol = 2, bottom = 'JB, DV, THU 2024')
```



JB, DV, THU 2024

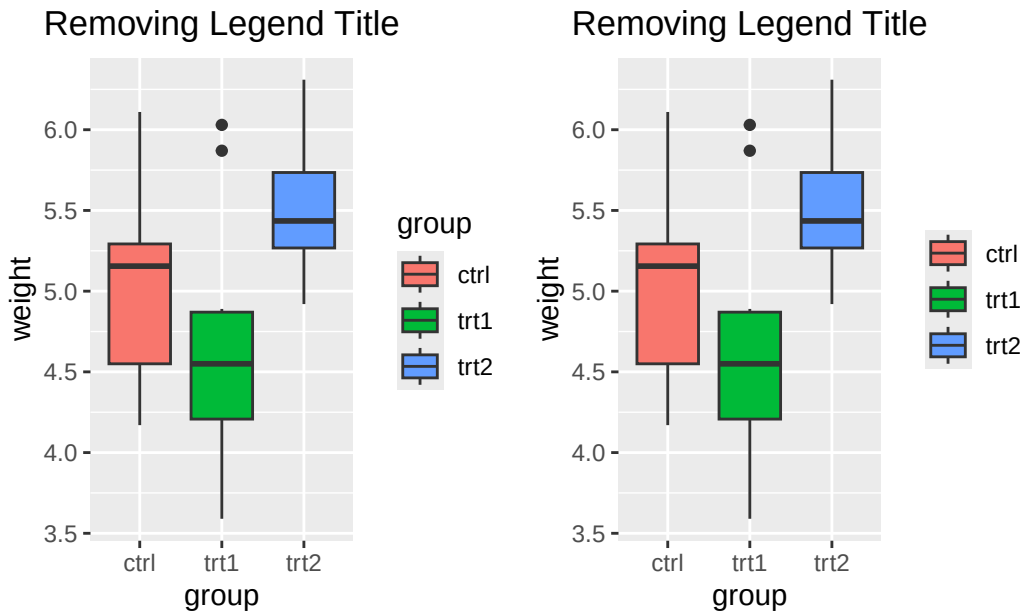
14 Remove Legend Title

`guide_legend(title = NULL` is used to remove the Legend Title

```
p1 <- ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  ggtitle('Removing Legend Title') +
  theme(plot.title = element_text(hjust=0.5))

p2 <- p1 + guides(fill = guide_legend(title = NULL))

grid.arrange(p1, p2, ncol = 2, bottom = 'JB, DV, THU 2024')
```



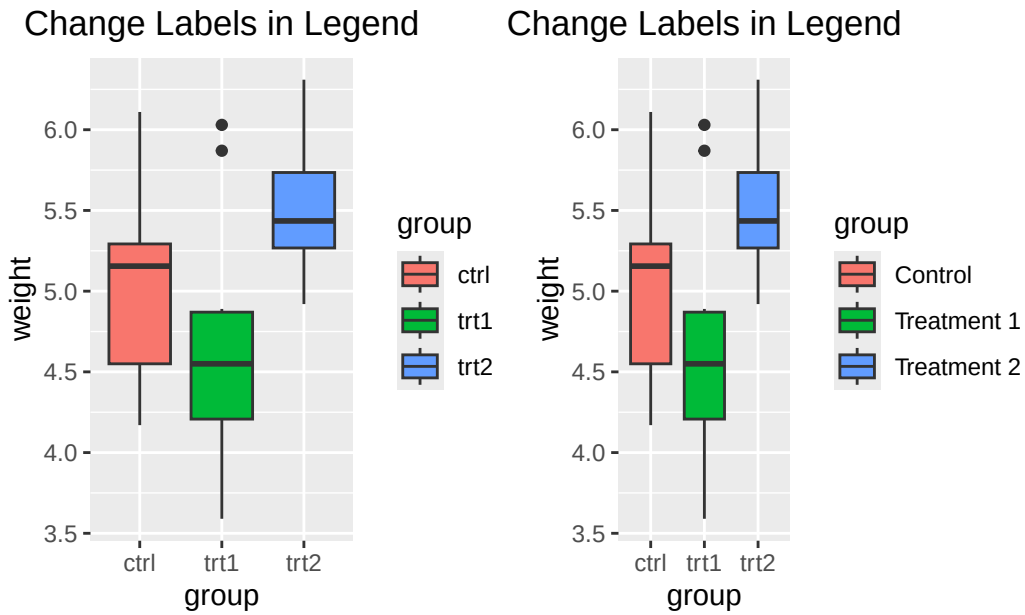
15 Change Labels in Legend

To change the Labels in the Legend, we first have to make the dataset using the `c('Legend 1', 'Legend 2', 'Legend 3')` line. Next, we can use `scale_fill_discrete(labels = c(x,y,z)` to change the labels inside the legend

```
p1 <- ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  ggtitle('Change Labels in Legend') +
  theme(plot.title = element_text(hjust=0.5))

p2 <- p1 +
  scale_fill_discrete(labels = c('Control',
                                 'Treatment 1',
                                 'Treatment 2'))

grid.arrange(p1, p2, ncol = 2, bottom = 'JB, DV, THU 2024')
```

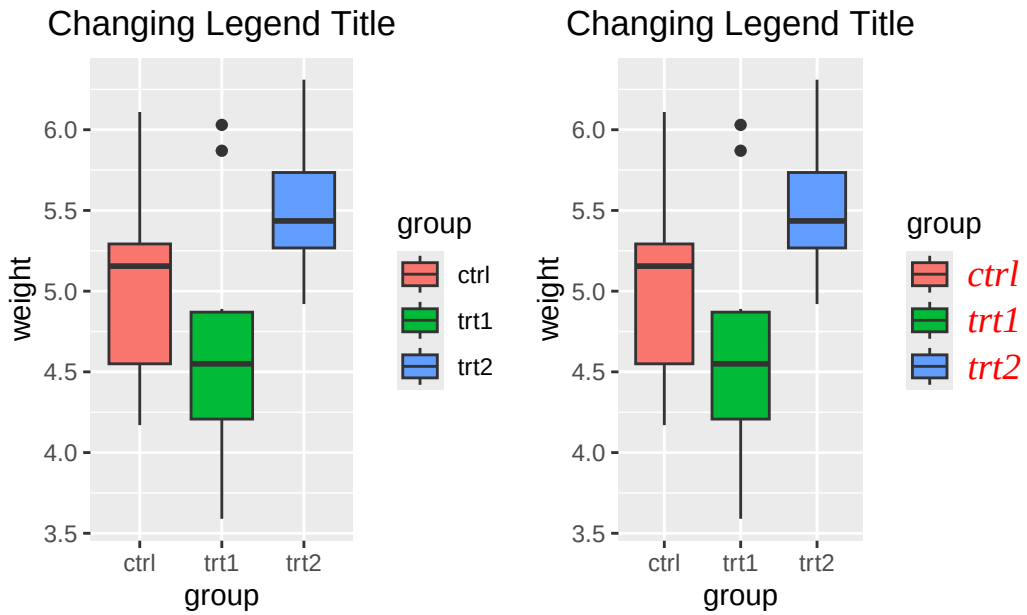


16 Change Apperarence of Legend Label

- Similarly to changing the appearance of the Legend Title, we use `theme` to change the appearance, with the additional line of `legend.text`
  - `face`: To change style of the text (`bold`, `italic`, `underline`)
  - `family`: To change the Font Style
  - `color`: To change the `color` of the text
  - `size`: To change the `size` of the text

```
p1 <- ggplot(PlantGrowth, aes(x = group, y = weight, fill = group)) +
  geom_boxplot() +
  ggtitle('Changing Legend Title') +
  theme(plot.title = element_text(hjust=0.5))

p2 <- p1 + theme(legend.text = element_text(face = 'italic',
                                            family = 'Times',
                                            color = 'red',
                                            size = 14))
grid.arrange(p1, p2, ncol = 2, bottom = 'JB, DV, THU 2024')
```

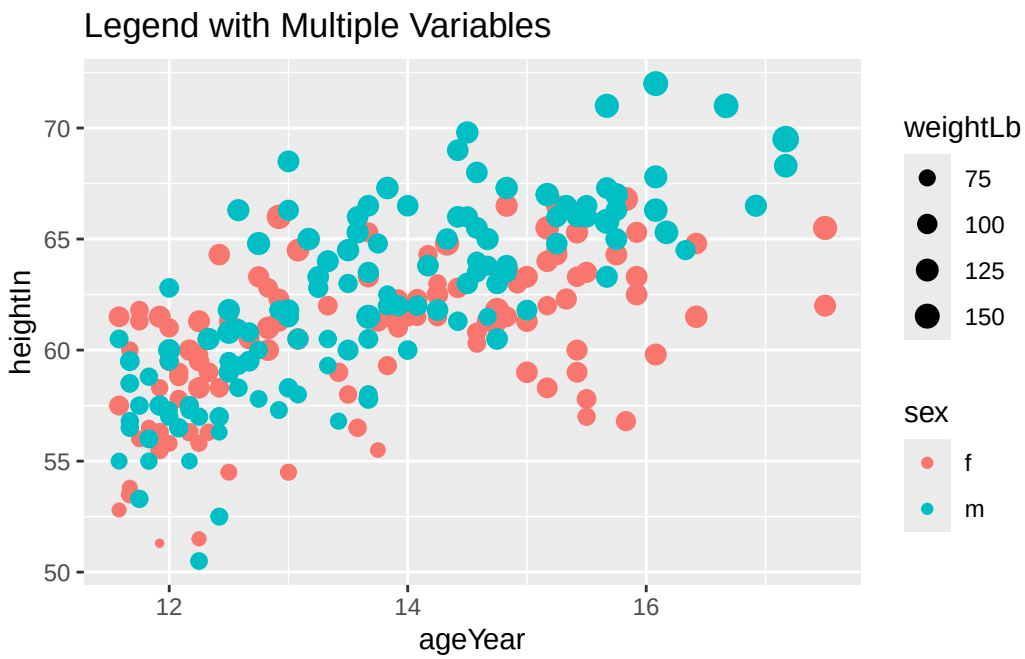


JB, DV, THU 2024

17 Legend with Multiple Variables

To add multiple variables in a chart, we can use add `aes` in `ggplot` to plot the dataset of `heightweight` and use `aes(size = weightLb)` to represent another variable that changed the size of the point accordingly

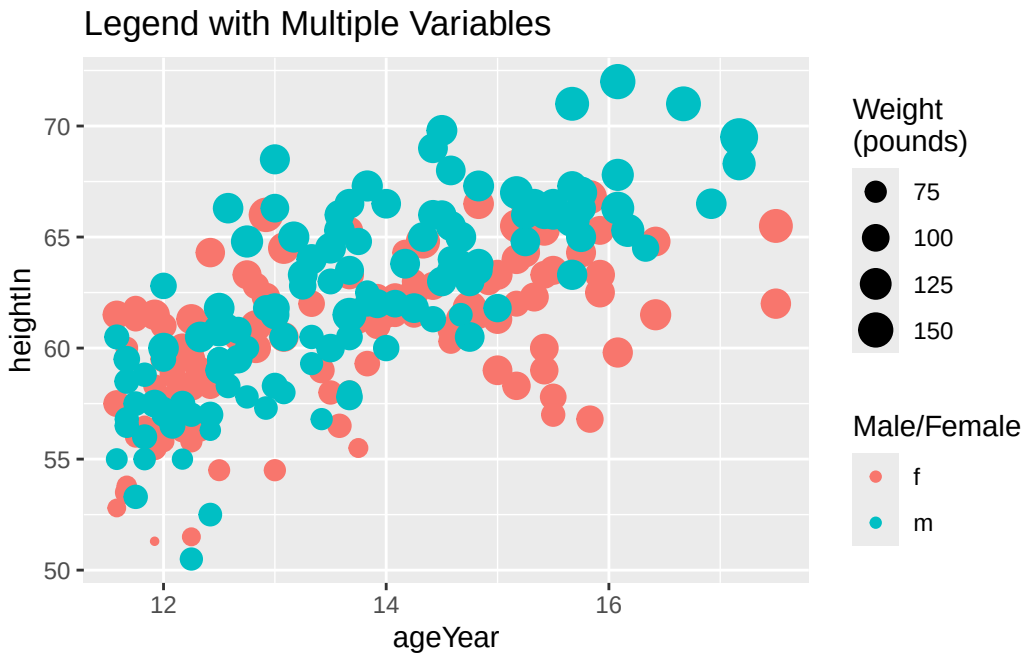
```
library(gcookbook)
ggplot(heightweight, aes(x = ageYear, y = heightIn, colour = sex)) +
  geom_point(aes(size = weightLb)) +
  scale_size_continuous(range = c(1, 4)) +
  ggtitle('Legend with Multiple Variables')
```



18 Resize the Point

Using the `size = weightLb` we can change the size of the point from `1` (being the smallest) to `4` being the largest in `geom_point`

```
library(gcookbook)
ggplot(heightweight, aes(x = ageYear, y = heightIn, colour = sex)) +
  geom_point(aes(size = weightLb)) +
  labs(colour = "Male/Female", size = "Weight\n(pounds)") +
  ggtitle('Legend with Multiple Variables')
```

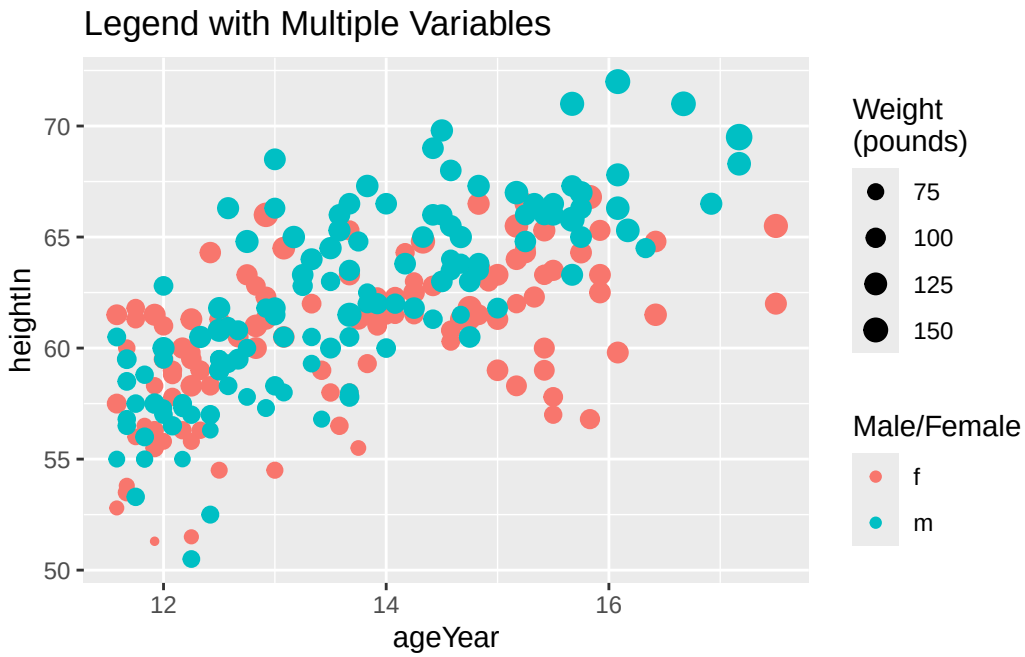


19 Change Legend Title

In the chart with multiple variable, the `color` represent `Male` or `Female` and the `size` represent the `Weight`.

In order to change the Legend Title, we use `labs` line and change the name by using of the `color` variable `colour = "Male/Female"` and `size` variable by adding `size = "Weight\n(pounds)"`

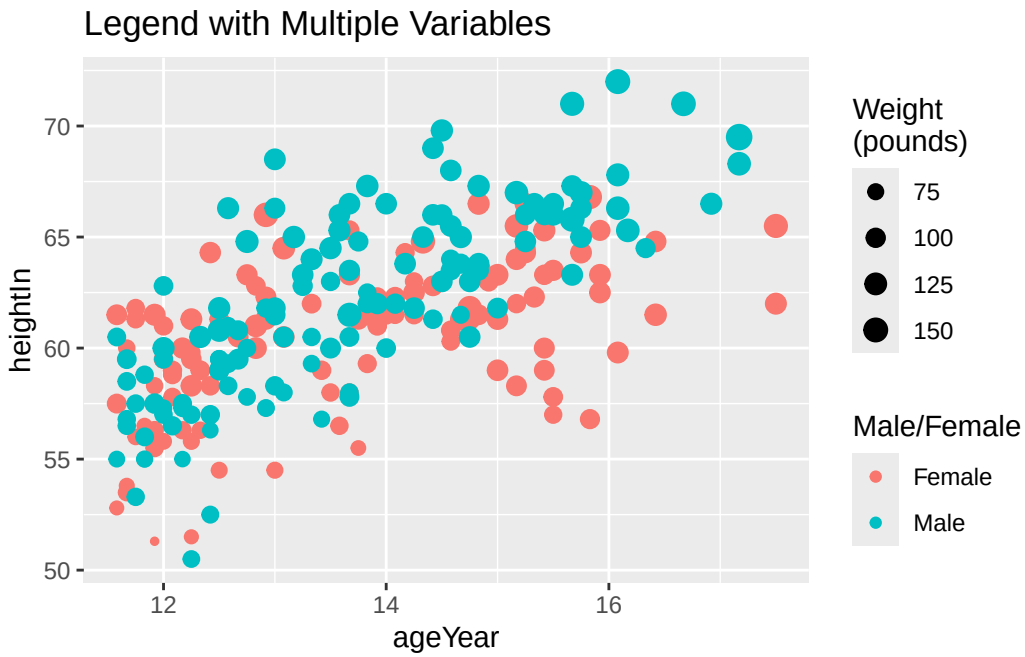
```
library(gcookbook)
ggplot(heightweight, aes(x = ageYear, y = heightIn, colour = sex)) +
  geom_point(aes(size = weightLb)) +
  scale_size_continuous(range = c(1, 4)) +
  labs(colour = "Male/Female", size = "Weight\n(pounds)") +
  ggtitle('Legend with Multiple Variables')
```



20 Change Labels in Legend

`scale_color_discrete(labels = c('Female','Male'))`: Using the `scale_color_discrete` we can change the `labels` in the `legend` to `Female` and `Male`

```
library(gcookbook)
ggplot(heightweight, aes(x = ageYear, y = heightIn, colour = sex)) +
  geom_point(aes(size = weightLb)) +
  scale_size_continuous(range = c(1, 4)) +
  labs(colour = "Male/Female", size = "Weight\n(pounds)") +
  scale_color_discrete(labels = c('Female','Male')) +
  ggtitle('Legend with Multiple Variables')
```


Legend with Multiple Variables
heightIn
70
65
60
55
50
12
14
16
ageYear
Weight (pounds)
75
100
125
150
Male/Female
Female
Male