Axes

David

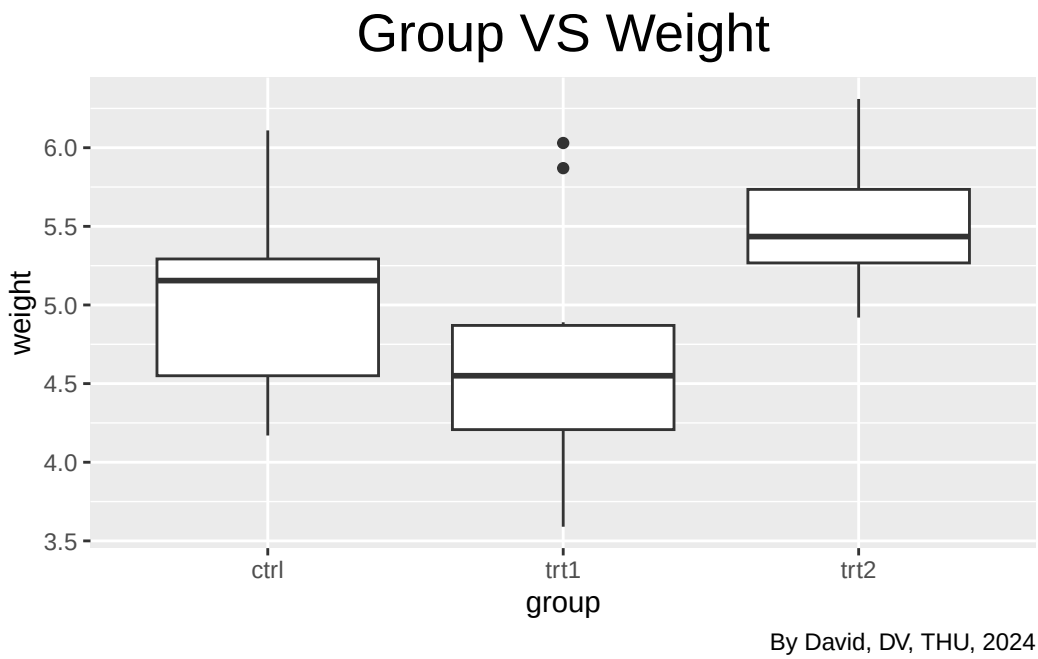
2024-11-25

Table of contents

1. Box plot

- Draw a boxplot using `PlantGrowth` dataset.
- x axis represents group, y axis represents `weight`.
- Use `geom_boxplot()` function to draw boxplot.
- `labs()` function is to add title, axis labels, and caption.
- Use `theme()` function to center the title.

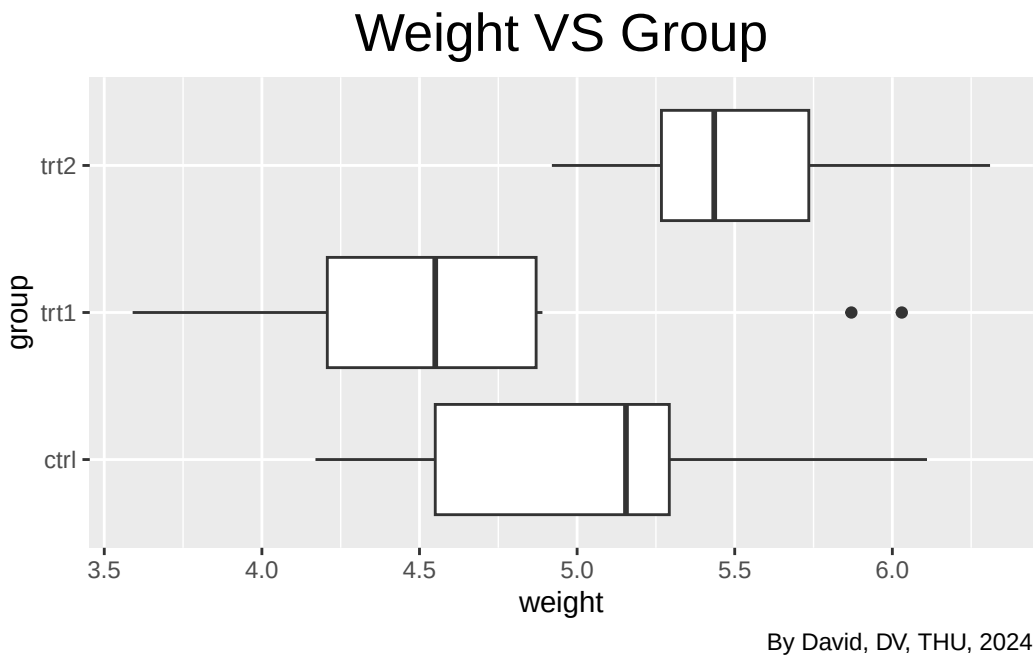
```r
library(ggplot2)
ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  labs(title = "Group VS Weight",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))
```



2. Swap x and y axes

- Use `coord_flip()` function to swap x and y axes.

```r
library(ggplot2)
ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  coord_flip()+
  labs(title = "Weight VS Group",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))
```



3. Arrange two plots side by side

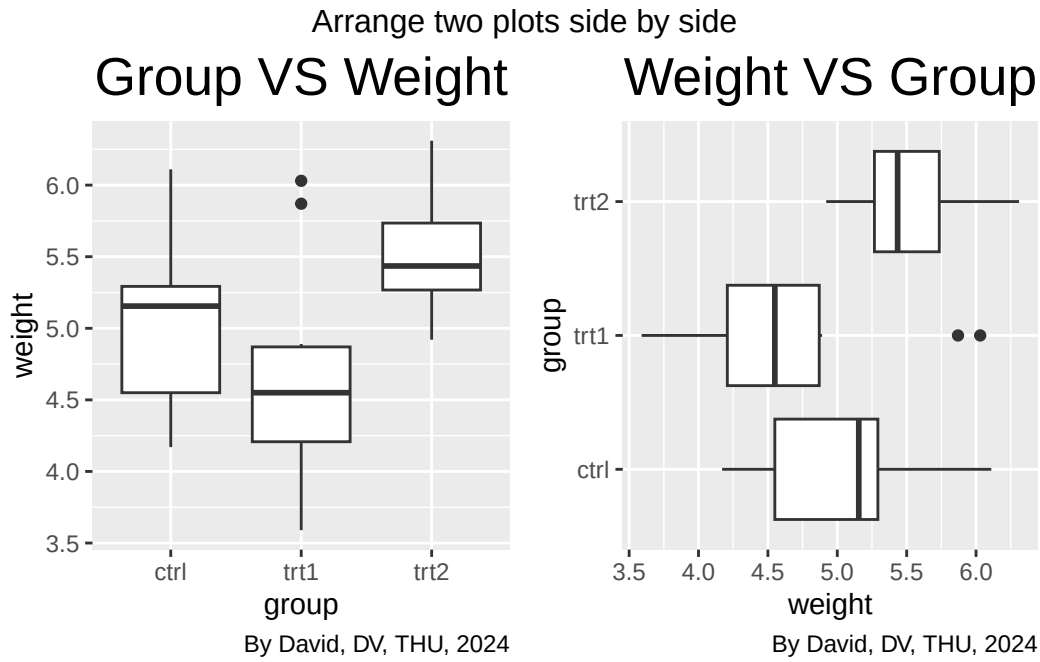
- Install `gridExtra` library.
- Prepare two plots: `p1, p2`.
- Use `grid.arrange()` function to arrange plots and add title.

```
library(ggplot2)
library(gridExtra)

p1 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  labs(title = "Group VS Weight",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))

p2 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  coord_flip()+
  labs(title = "Weight VS Group",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))
```

```
grid.arrange(p1, p2, ncol = 2,
             top = 'Arrange two plots side by side')
```



4. Setting the Range of a Continuous Axis

- `ylim()`: Sets the limits of the y-axis.

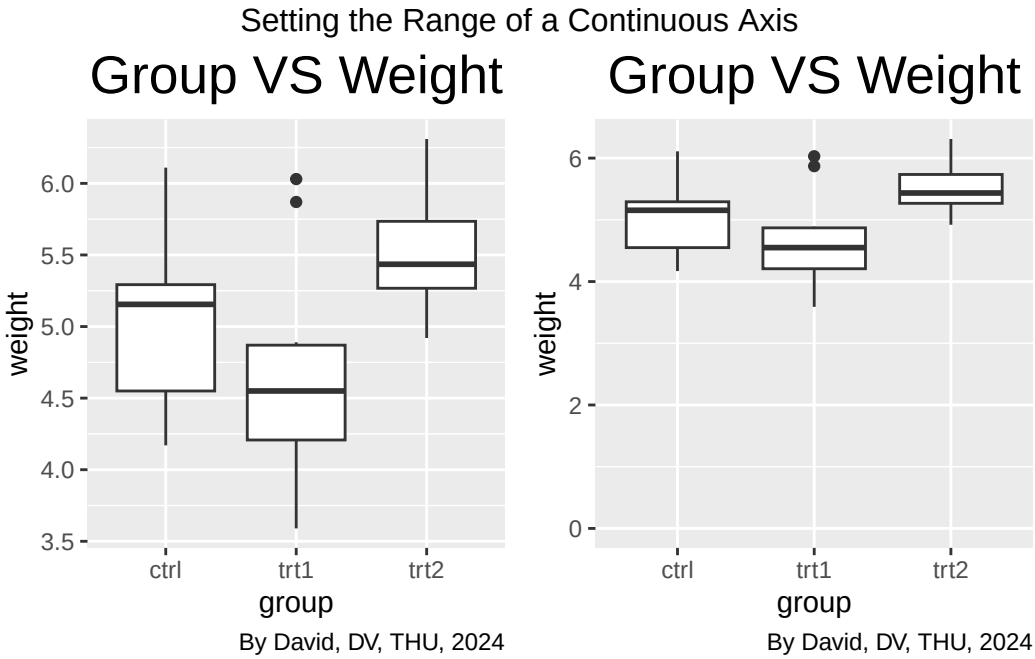
```
library(ggplot2)
library(gridExtra)

p1 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  labs(title = "Group VS Weight",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))

p2 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  labs(title = "Group VS Weight",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20)) +
```

```
  ylim(0, max(PlantGrowth$weight))

grid.arrange(p1, p2, ncol = 2,
             top = 'Setting the Range of a Continuous Axis')
```



5. Reversing a Continuous Axis

- `scale_y_reverse()` reverses the direction of the y-axis on a plot. So, what was previously the highest value on the y-axis becomes the lowest, and vice-versa.

```
library(ggplot2)
library(gridExtra)

p1 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  labs(title = 'Group vs Weight',
       x = 'Group',
       y = 'Weight',
       caption = 'By David, DV, THU, 2024') +
  theme(plot.title = element_text(hjust = 0.5))
```

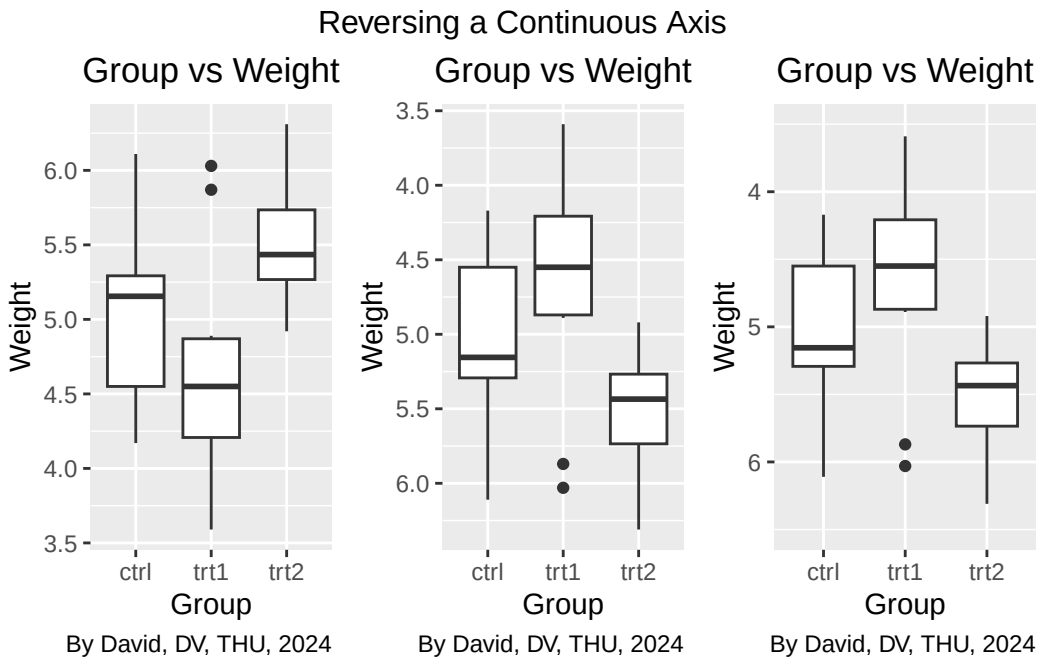
```
p2 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  scale_y_reverse() +
  labs(title = 'Group vs Weight',
       x = 'Group',
       y = 'Weight',
       caption = 'By David, DV, THU, 2024') +
  theme(plot.title = element_text(hjust = 0.5))

p3 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  scale_y_reverse() +
  ylim(6.5, 3.5) +
  labs(title = 'Group vs Weight',
       x = 'Group',
       y = 'Weight',
       caption = 'By David, DV, THU, 2024') +
  theme(plot.title = element_text(hjust = 0.5))
```

```
Scale for y is already present.
Adding another scale for y, which will replace the existing scale.
```

Scale for y is already present. Adding another scale for y, which will replace the existing scale.

```
grid.arrange(p1, p2, p3, ncol = 3,
             top = 'Reversing a Continuous Axis')
```



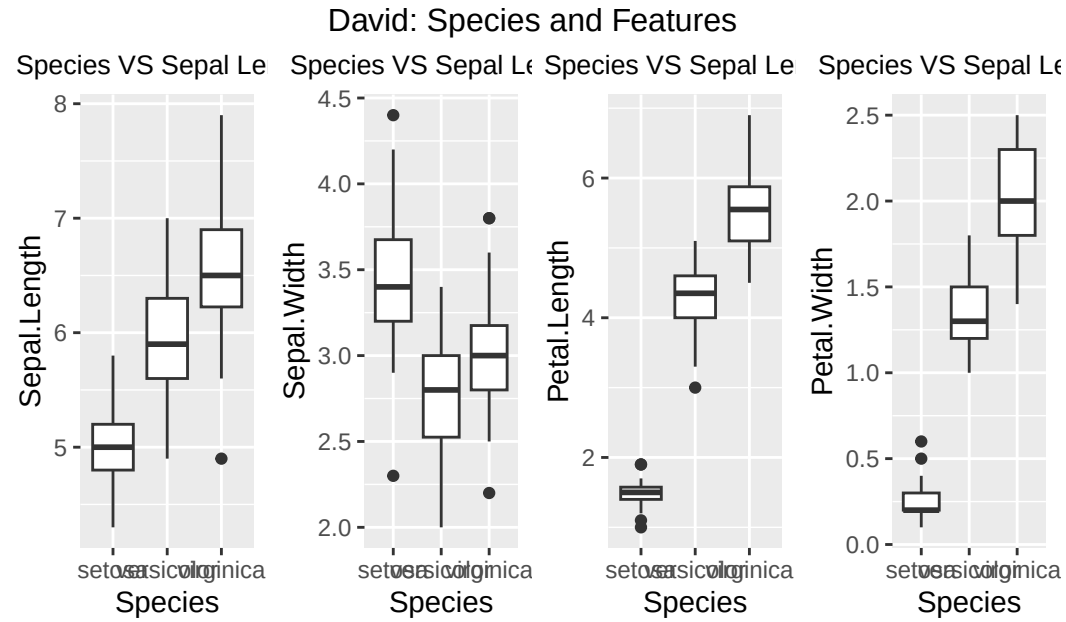
6. Practice: Multiple charts from Iris

- `grid.arrange()` function from the gridExtra package in R to arrange four plots (named plot1, plot2, plot3, and plot4) into a single figure.

```
plot1<-ggplot(iris, aes(x = Species, y = Sepal.Length)) +
  geom_boxplot() +
  labs(title = "Species VS Sepal Length",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 10))
plot2<-ggplot(iris, aes(x = Species, y = Sepal.Width)) +
  geom_boxplot() +
  labs(title = "Species VS Sepal Length",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 10))
plot3<-ggplot(iris, aes(x = Species, y = Petal.Length)) +
  geom_boxplot() +
  labs(title = "Species VS Sepal Length",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 10))
plot4<-ggplot(iris, aes(x = Species, y = Petal.Width)) +
  geom_boxplot() +
```

```
  labs(title = "Species VS Sepal Length",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 10))

grid.arrange(plot1, plot2, plot3, plot4, ncol=4,
             top = 'David: Species and Features')
```



7. Setting the Scaling Ratio of the X- and Y-Axes

- `coord_fixed()` is a function that fixes the aspect ratio of the plot. This means that the units on the x-axis and y-axis will be of equal length.

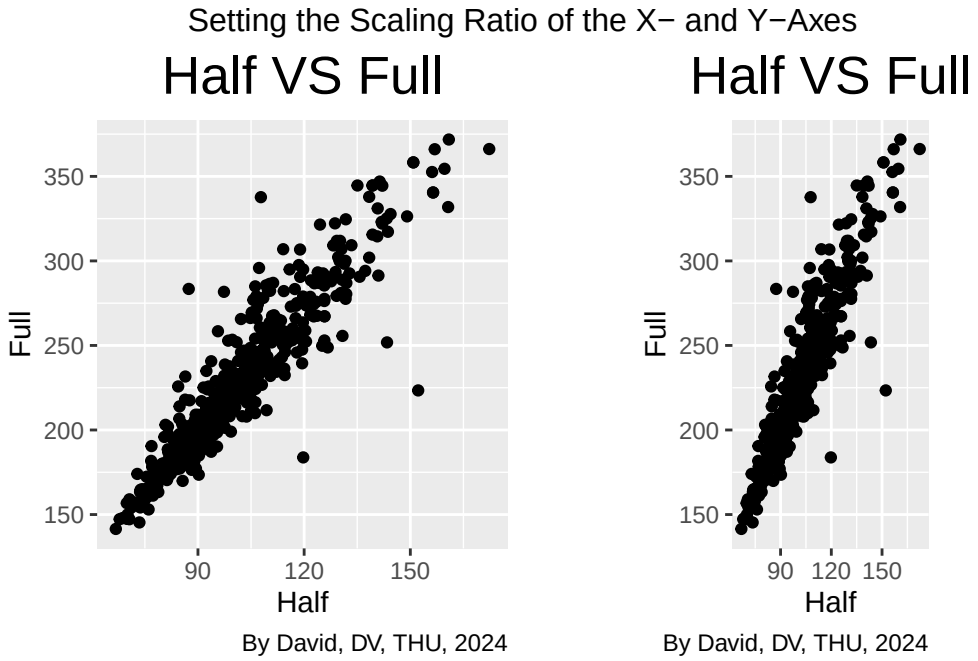
```
library(gcookbook)

plot1 <- ggplot(marathon, aes(x=Half,y=Full)) +
  geom_point() +
  labs(title = "Half VS Full",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))

plot2 <- ggplot(marathon, aes(x=Half,y=Full)) +
```

```r
  geom_point() +
  coord_fixed() +
  labs(title = "Half VS Full",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))

grid.arrange(plot1, plot2, ncol=2,
             top = 'Setting the Scaling Ratio of the X- and Y-Axes')
```



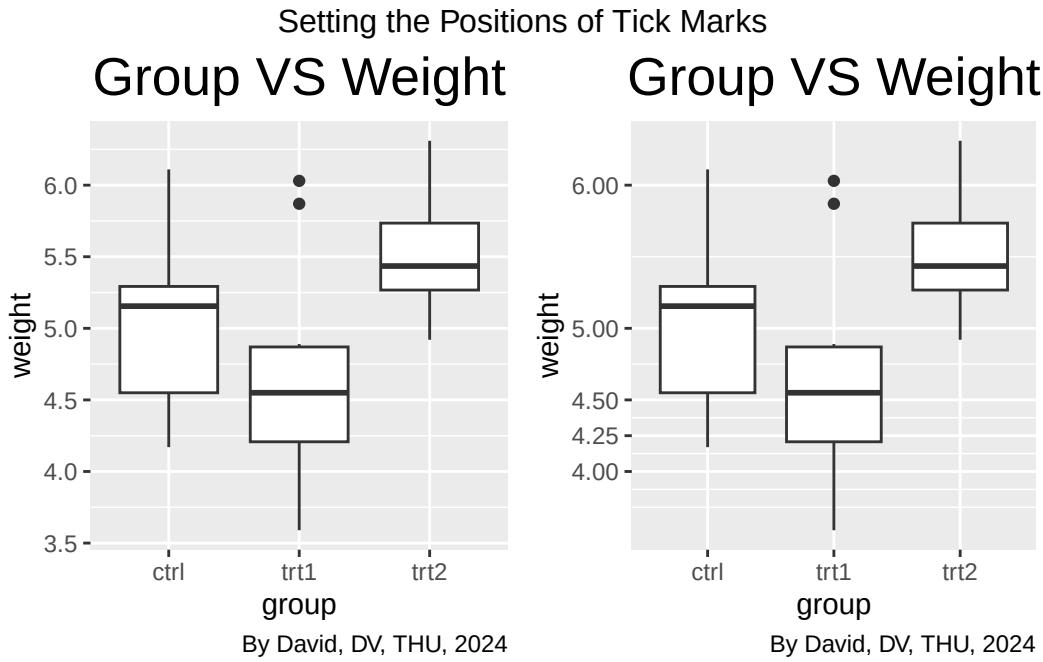
8. Setting the Positions of Tick Marks

- `scale_y_continuous()` is a function used to control the y-axis of a plot when the y-axis variable is continuous (meaning it can take on any value within a range, like temperature, weight, or time).

```r
p1 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  labs(title = "Group VS Weight",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))
```

```r
p2 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  scale_y_continuous(breaks = c(4, 4.25, 4.5, 5, 6, 8)) +
  labs(title = "Group VS Weight",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))

# Add title
grid.arrange(p1, p2, ncol = 2,
             top = 'Setting the Positions of Tick Marks')
```



9. Removing Tick Marks and Labels

- To remove the tick marks, use `theme(axis.ticks=element_blank())`. This will remove the tick marks on both axes.
- To remove the tick marks, the labels, and the grid lines, set breaks to `NULL`

```r
p1 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  labs(title = "Group VS Weight",
       caption = "By David, DV, THU, 2024") +
```

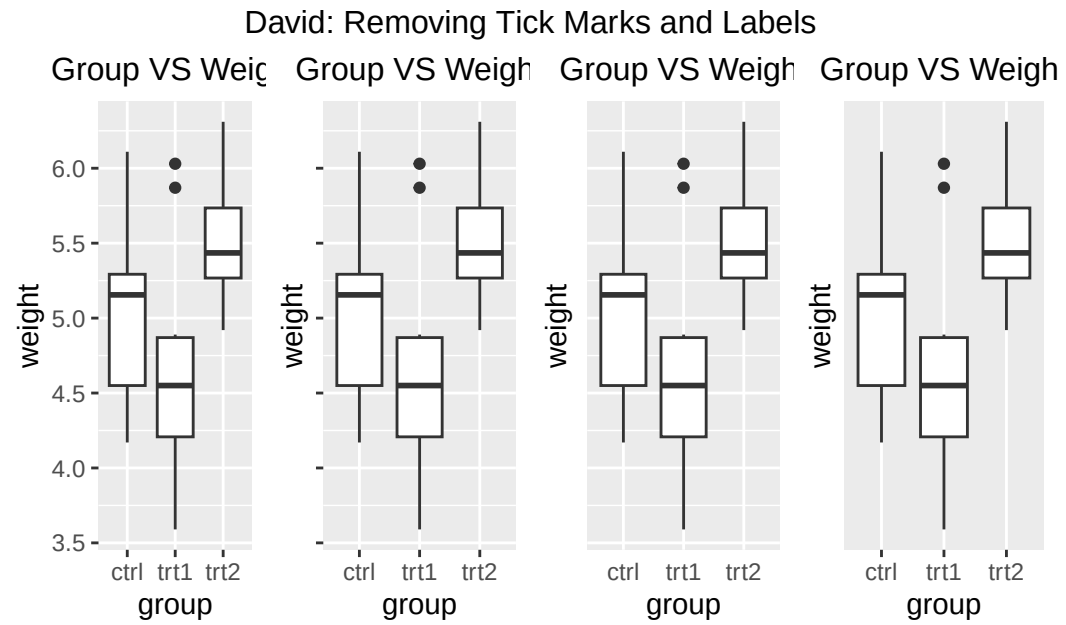
```
  theme(plot.title = element_text(hjust = 0.5, size = 12))

p2 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  theme(axis.text.y = element_blank()) +
  labs(title = "Group VS Weight",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 12))

p3 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  theme(axis.ticks = element_blank(), axis.text.y = element_blank()) +
  labs(title = "Group VS Weight",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 12))

p4 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  scale_y_continuous(breaks = NULL) +
  labs(title = "Group VS Weight",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 12))

grid.arrange(p1, p2, p3, p4, ncol = 4,
             top = 'David: Removing Tick Marks and Labels')
```



y David, DV, THU, 2024By David, DV, THU, 2024By David, DV, THU, 2024By David, DV, THU, 2024

10. Changing the Text of Tick Labels

- `breaks`: Specifying the locations of the tick marks and labels on the y-axis (breaks = seq(min, max, by = interval)). This gives you more control over the spacing and readability of the axis labels.
- `labels`: Customizing the text of the axis labels (labels = function(x) ...). This allows for formatting, unit specification, or abbreviation of the labels.

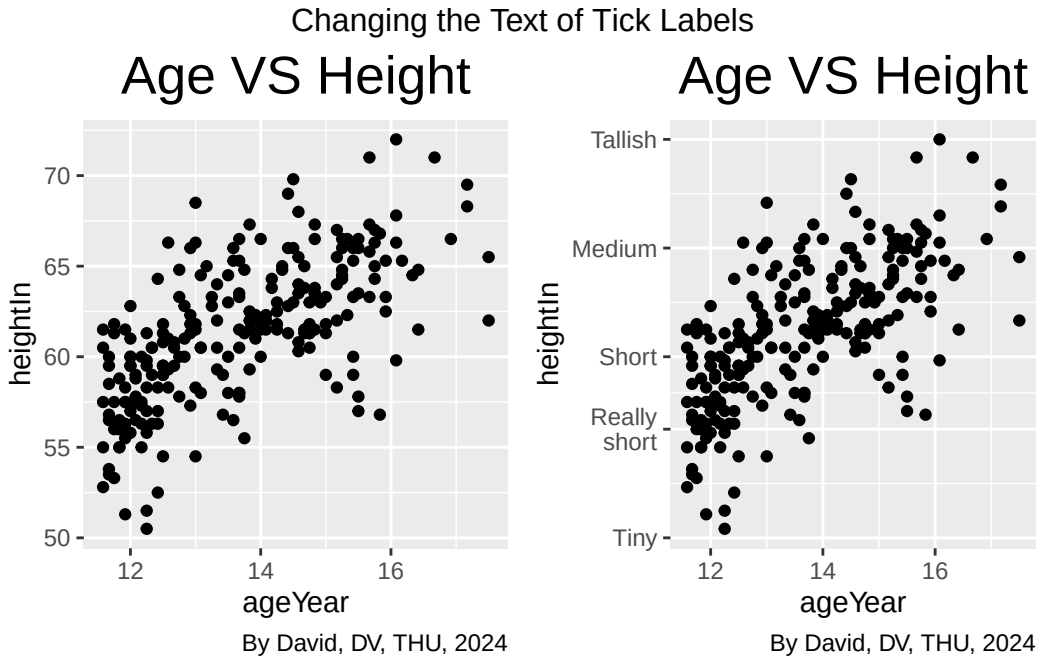
```
library(gcookbook)

p1 <- ggplot(heightweight, aes(x = ageYear, y = heightIn)) +
  geom_point() +
  labs(title = "Age VS Height",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))

p2 <- ggplot(heightweight, aes(x = ageYear, y = heightIn)) +
  geom_point() +
  scale_y_continuous(
    breaks = c(50, 56, 60, 66, 72),
    labels = c("Tiny", "Really\nshort", "Short", "Medium", "Tallish")
  ) +
```

```
  labs(title = "Age VS Height",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))

# Add title
grid.arrange(p1, p2, ncol = 2,
             top = 'Changing the Text of Tick Labels')
```



11. Lower 'Tallish' to include 6 people

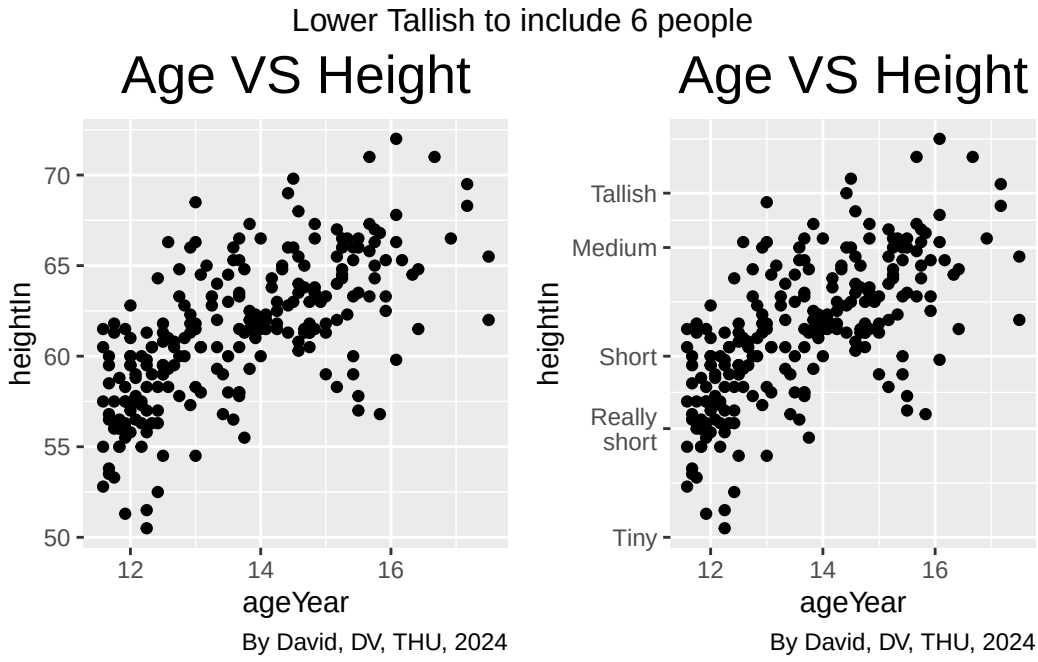
- Use `scale_y_continuous()` to change the text of the axis labels
- Use `breaks = c()` to break the axis labels
- Use `labels = c()` to add a name in axis labels

```
p1 <- ggplot(heightweight, aes(x = ageYear, y = heightIn)) +
  geom_point() +
  labs(title = "Age VS Height",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))

p2 <- ggplot(heightweight, aes(x = ageYear, y = heightIn)) +
```

```r
  geom_point() +
  scale_y_continuous(
    breaks = c(50, 56, 60, 66, 69),
    labels = c("Tiny", "Really\nshort", "Short", "Medium", "Tallish")
  ) +
  labs(title = "Age VS Height",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 20))

# Add title
grid.arrange(p1, p2, ncol = 2,
             top = 'Lower Tallish to include 6 people')
```

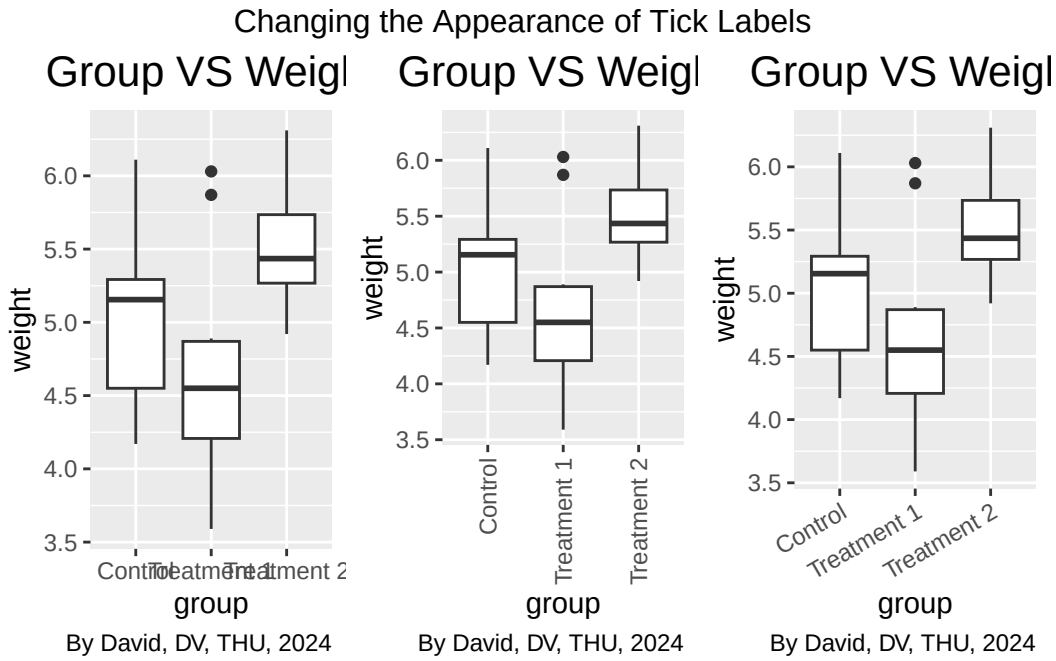


12. Changing the Appearance of Tick Labels

- `axis.text.x = element_text(...)`: This part specifically targets the text elements of the x-axis. element_text is a function that controls the formatting of text within the plot.
- `angle = 30`: This rotates the x-axis labels by 30 degrees. This is often useful when labels are long and overlapping.

```r
library(ggplot2)
library(gridExtra)
pg_plot <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot() +
  scale_x_discrete(
    breaks = c("ctrl", "trt1", "trt2"),
    labels = c("Control", "Treatment 1", "Treatment 2")
  ) +
  labs(title = "Group VS Weight",
       caption = "By David, DV, THU, 2024") +
  theme(plot.title = element_text(hjust = 0.5, size = 16))
p1 <- pg_plot
p2 <- pg_plot +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = .5))
p3 <- pg_plot +
  theme(axis.text.x = element_text(angle = 30, hjust = 1, vjust = 1))

grid.arrange(p1, p2, p3, ncol = 3,
             top = 'Changing the Appearance of Tick Labels')
```



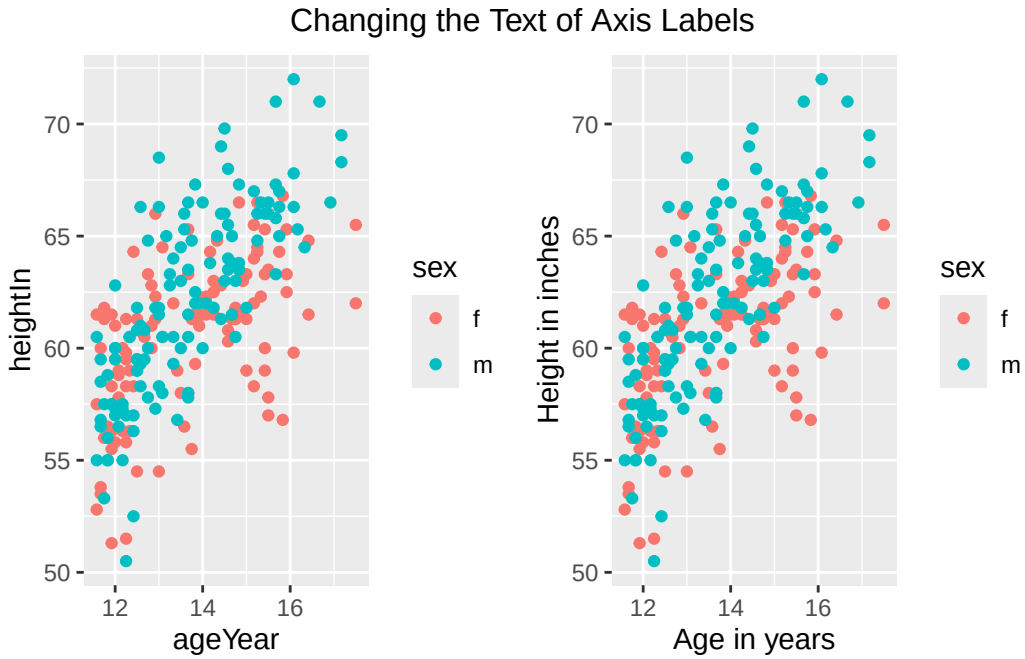
13. Changing the Text of Axis Labels

- Use `xlab()` or `ylab()` to change the text of the axis labels

```
hw_plot <- ggplot(heightweight, aes(x = ageYear, y = heightIn, colour = sex)) +
  geom_point()

p1 <- hw_plot
p2 <- hw_plot +
  xlab('Age in years') +
  ylab('Height in inches')

grid.arrange(p1, p2, ncol = 2,
             top = 'Changing the Text of Axis Labels')
```



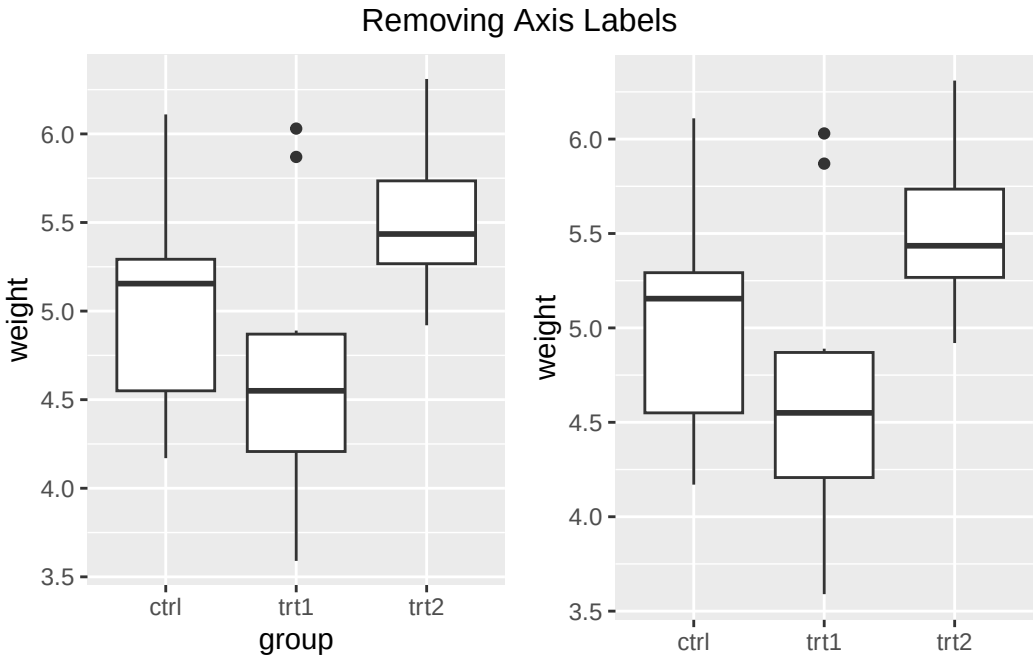
14. Removing Axis Labels

```
p1 <- ggplot(PlantGrowth, aes(x = group, y = weight)) +
  geom_boxplot()

p2 <- p1  +
```

```
  xlab(NULL)

grid.arrange(p1, p2, ncol = 2,
             top = 'Removing Axis Labels')
```



15. Changing the Appearance of Axis Labels.

- `axis.title.x = element_text(...)`: This specifies that we're modifying the x-axis title. element_text is used to set text properties.

```
p1 <- ggplot(heightweight, aes(x = ageYear, y = heightIn)) +
  geom_point()
p2 <- p1 +
  theme(axis.title.x = element_text(face = "italic", colour = "darkred",
                                    size = 14))
grid.arrange(p1, p2, ncol = 2,
             top = 'Changing the Appearance of Axis Labels')
```

Changing the Appearance of Axis Labels