Data Visualization using ggplot

Alice Kamau| Ken Mwai | Mark Otiende

7/5/2021

Learning objectives

- Produce boxplots, scatter plots and smoothed plots using ggplot.
- Describe what faceting is and apply faceting in ggplot.
- Modify the aesthetics of an existing ggplot plot (including axis labels and color).
- Build complex and customized plots from data in a data frame.

Building your plots iteratively

- Building plots with ggplot2 is typically an iterative process.
- We start by defining the dataset we'll use, lay out the axes, and choose a geom:
- Then, we start modifying this plot to extract more information from it.
- For instance, we can add transparency (alpha) to avoid overplotting:
- We can also add colors for all the points:
- Or to color each species in the plot differently, you could use a vector as an input to the argument color.
- ggplot2 will provide a different color corresponding to different values in the vector. Here is an example where we color with species_id:
- Load required package

```
library(tidyverse)
```

- Set the directory

```
setwd("/Users/akamau/Documents/I-StaR/Course 1/Day4/Presentation/")
```

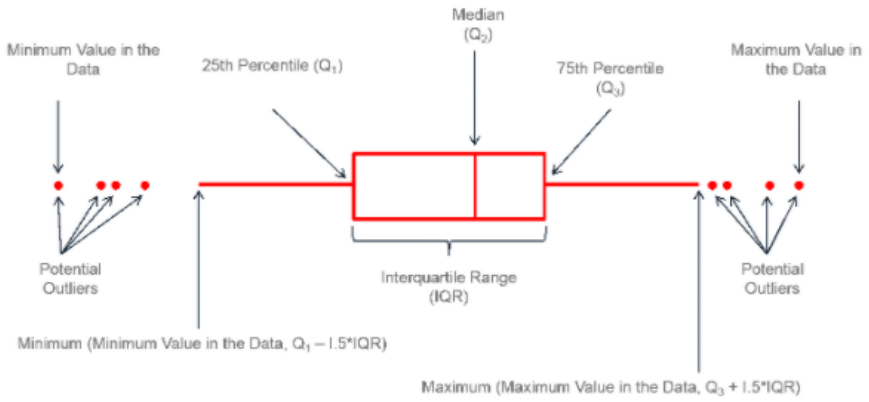
- Load the data

```
bw_df <- read.csv("Data/birthweight2.csv")
names(bw_df)
```

```
##  [1] "id"       "matage"   "ht"       "gestwks" "sex"      "bweight" "ethnic"
##  [8] "lbw"      "agegrp"   "lbw2"     "agegrp1"
```

Boxplot - for a categorical and continuous variable

- We will use boxplots to visualize the distribution of birth weight by gender:
- Boxplots provides a standardized way of displaying the distribution of data
- It attempts to provide a visual shape of the data distribution.
- This is based on some summary measures: min, 1^{st} quartile, median, 3^{rd} quartile, max
- Range, IQR, Outliers - 3 $\times$ IQR above 3^{rd} or below 1^{st} quartiles.



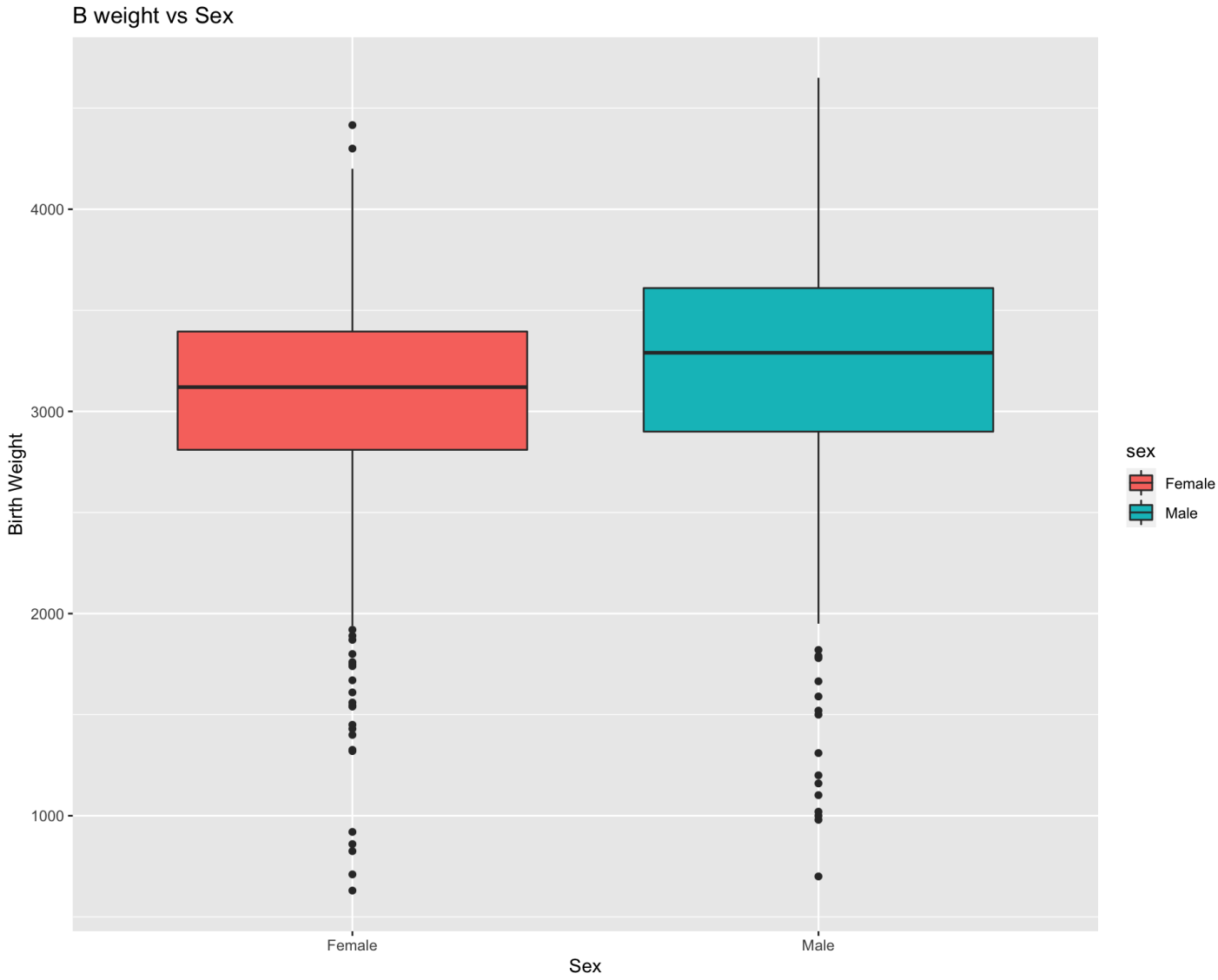
Lets do a Box plot?

- A box plot of bweight vs sex

```
ggplot(data = bw_df)
```

Adding aesthetics and labels

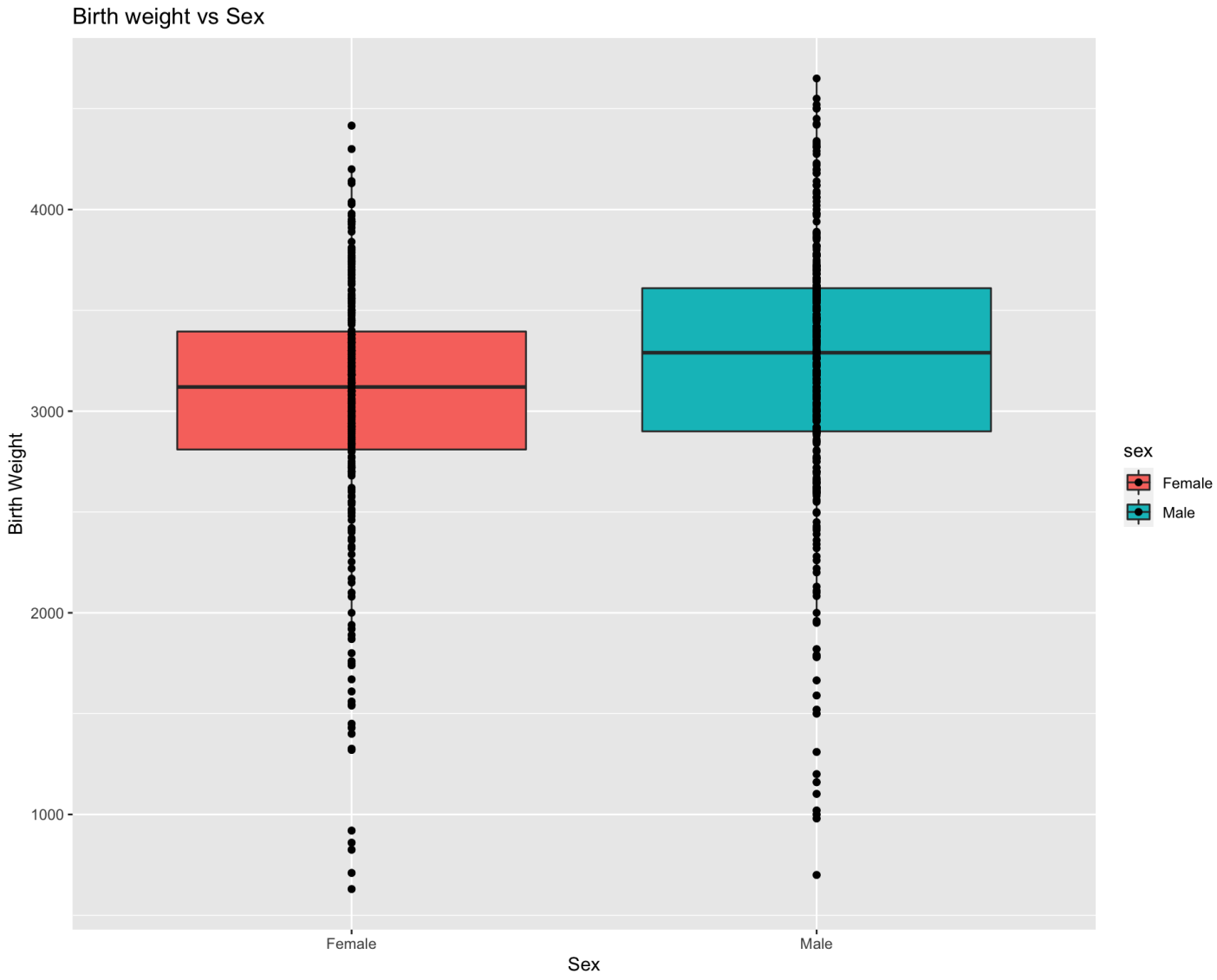
```
ggplot(data = bw_df) + geom_boxplot(aes(y = bweight, x = sex,
    fill = sex)) + ylab("Birth Weight") + xlab("Sex") + ggtitle("B weight vs Sex")
```



Box plot and add scatter

- By adding points to the boxplot, we can have a better idea of the number of measurements and of their distribution:

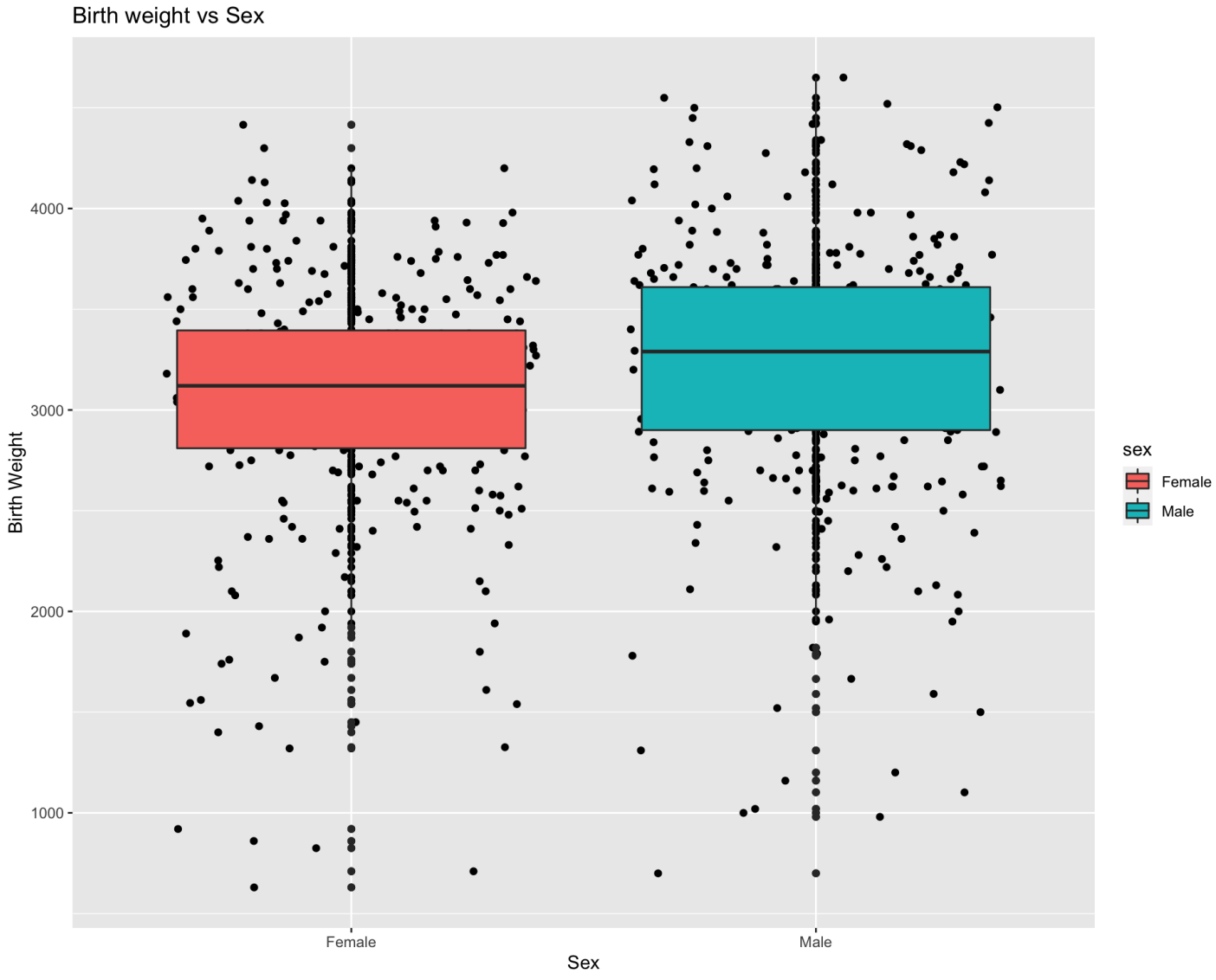
```
ggplot(data = bw_df, mapping = aes(y = bweight, x = sex, fill = sex)) +
    geom_boxplot() + geom_point() + ylab("Birth Weight") + xlab("Sex") +
    ggtitle("Birth weight vs Sex")
```



Box plot and add scatter points that are jittered

- We will jitter points to reduce overplotting
- Notice how the boxplot layer is behind the jitter layer? What do you need to change in the code to put the boxplot in front of the points such that it's not hidden?

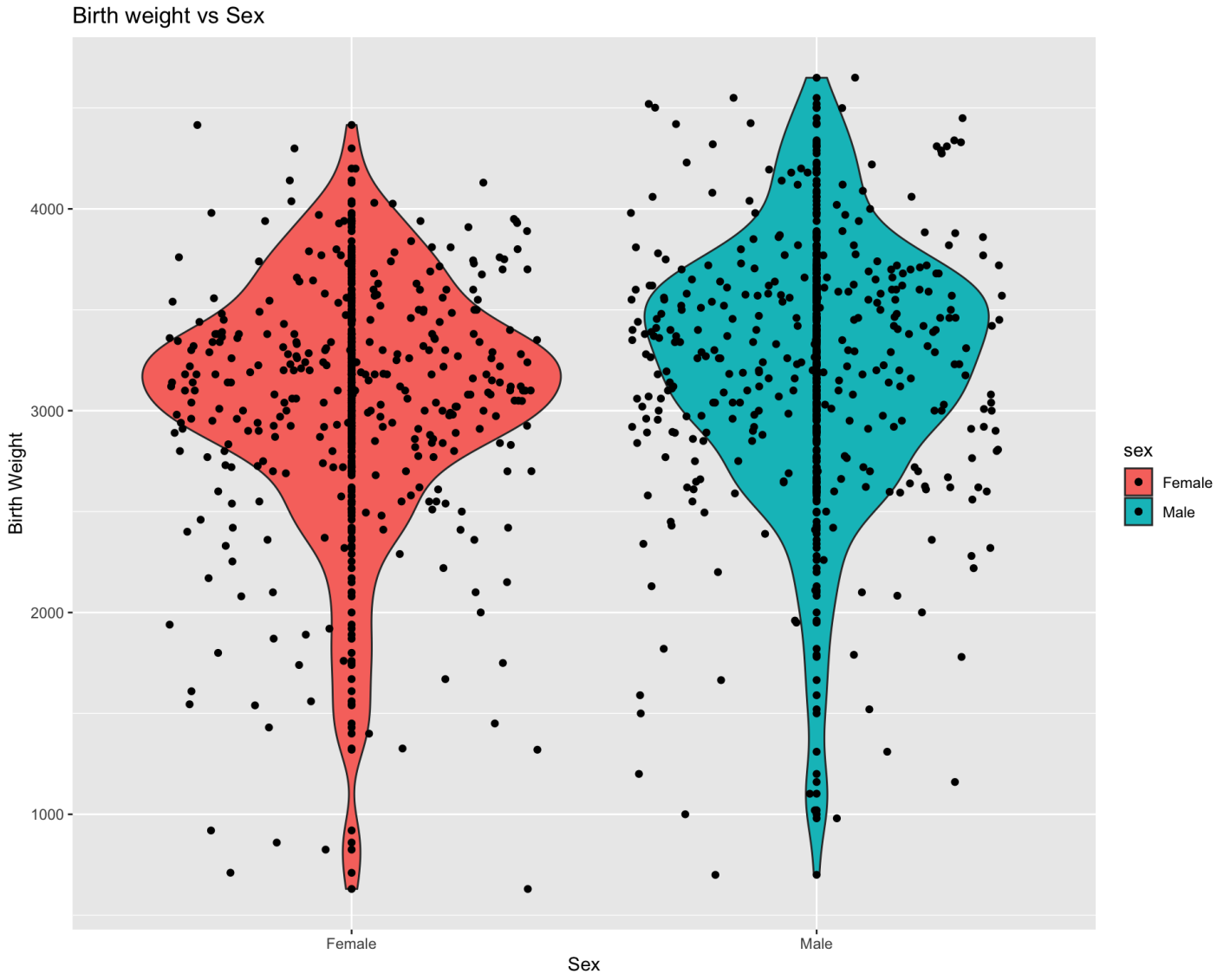
```
ggplot(data = bw_df, mapping = aes(y = bweight, x = sex, fill = sex)) +
    geom_point() + geom_jitter() + geom_boxplot() + ylab("Birth Weight") +
    xlab("Sex") + ggtitle("Birth weight vs Sex")
```



- Boxplots are useful summaries, but hide the shape of the distribution.
- For example, if there is a bimodal distribution, it would not be observed with a boxplot.
- An alternative to the boxplot is the violin plot (sometimes known as a beanplot), where the shape (of the density of points) is drawn.
- Replace the box plot with a violin plot; see geom_violin().

Violin plot

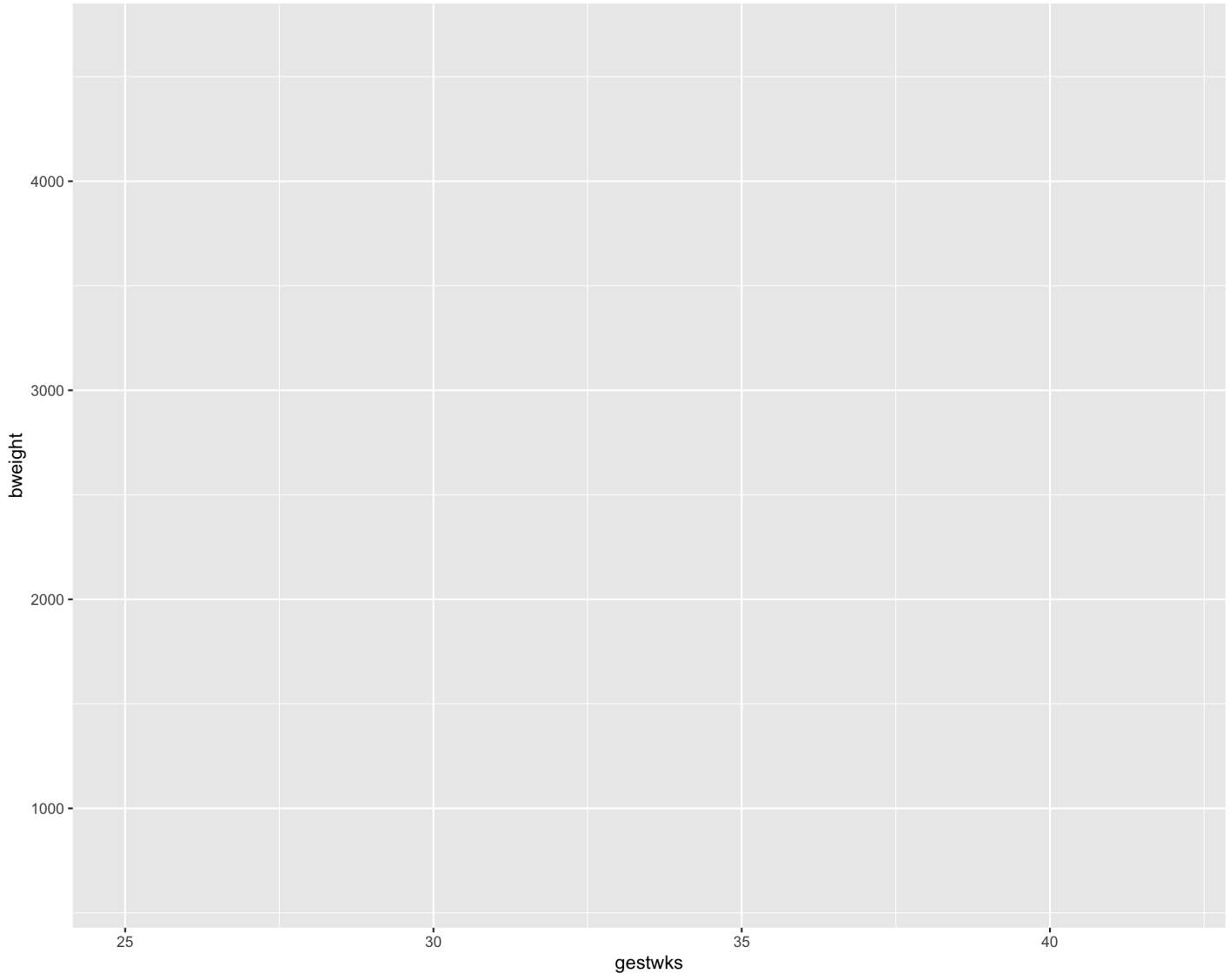
```
ggplot(data = bw_df, mapping = aes(y = bweight, x = sex, fill = sex)) +
    geom_violin() + geom_point() + geom_jitter() + ylab("Birth Weight") +
    xlab("Sex") + ggtitle("Birth weight vs Sex")
```



Scatter plot with ggplot2 - for two continuous variables

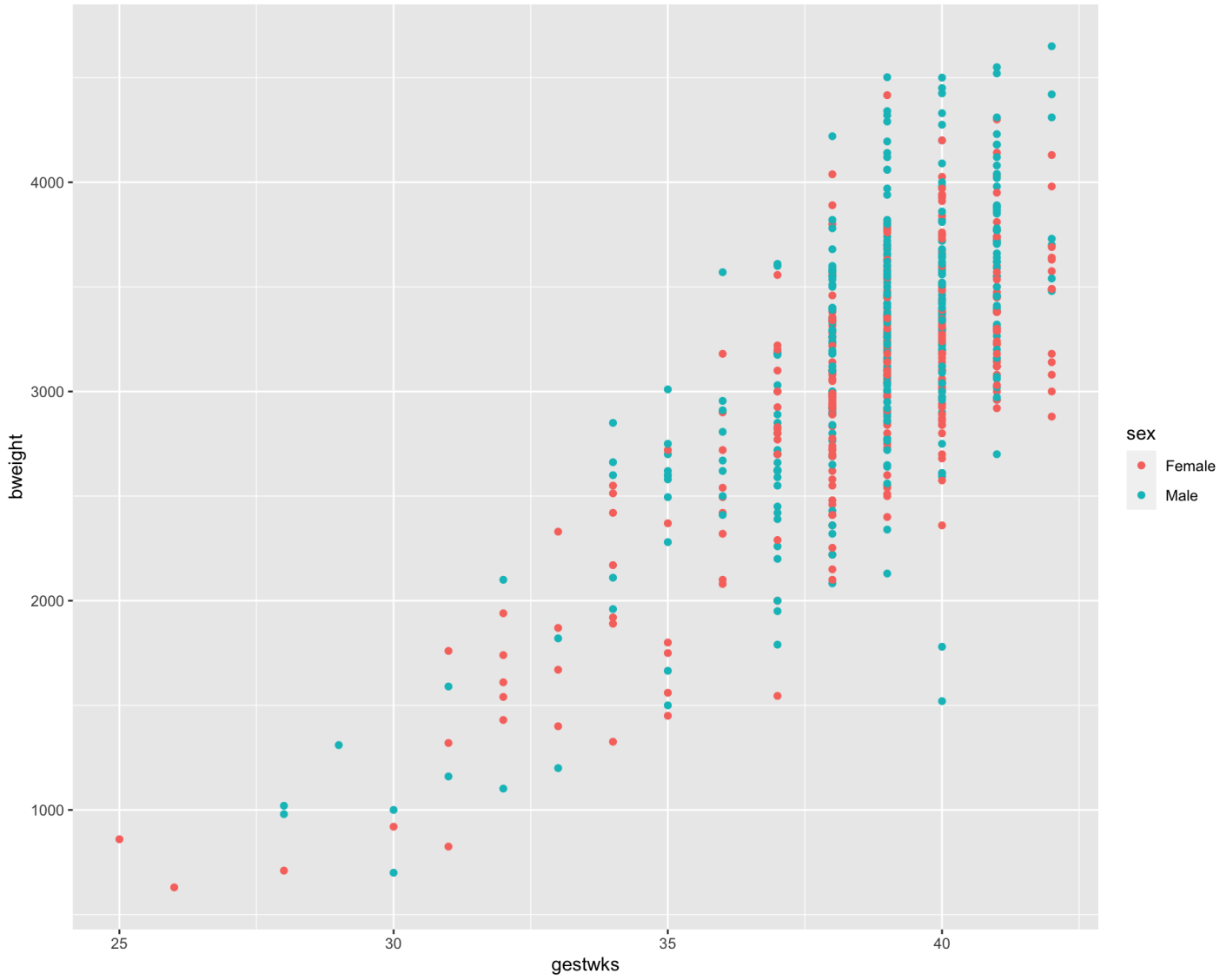
- We can build a plot sequentially to see how each grammatical layer changes the appearance
- Start with data and aesthetics

```
# Start with data and aesthetics
ggplot(data = bw_df, mapping = aes(x = gestwks, y = bweight,
    color = sex))
```



Add a point geom

```
# Start with data and aesthetics
ggplot(data = bw_df, mapping = aes(x = gestwks, y = bweight,
    color = sex)) + # Add a point geom
geom_point()
```

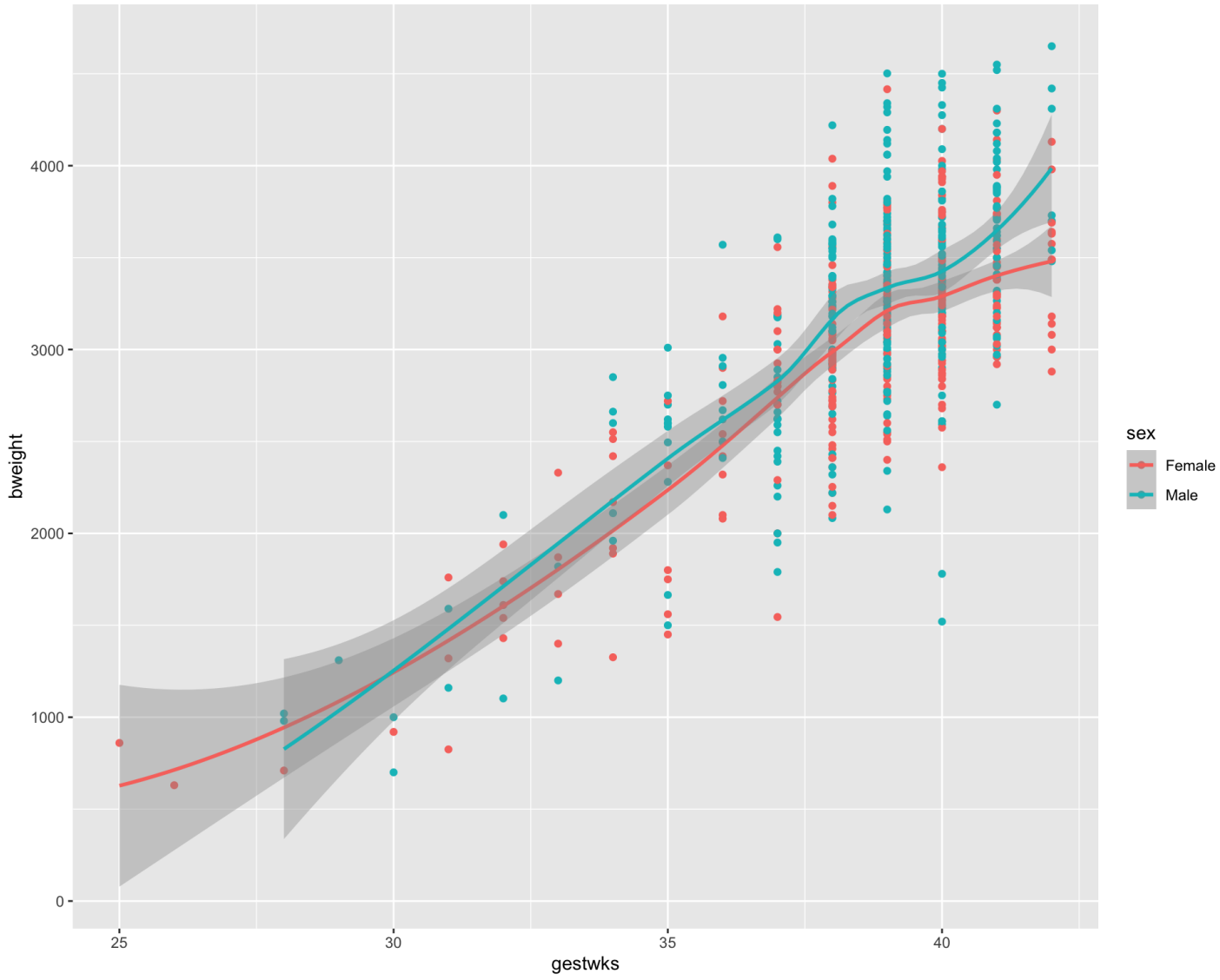


Add a smooth geom

- To add a regression line on a scatter plot, use the function geom_smooth()

```
# Start with data and aesthetics
ggplot(data = bw_df, mapping = aes(x = gestwks, y = bweight,
    color = sex)) + # Add a point geom
geom_point() + ## Add a smooth geom
geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

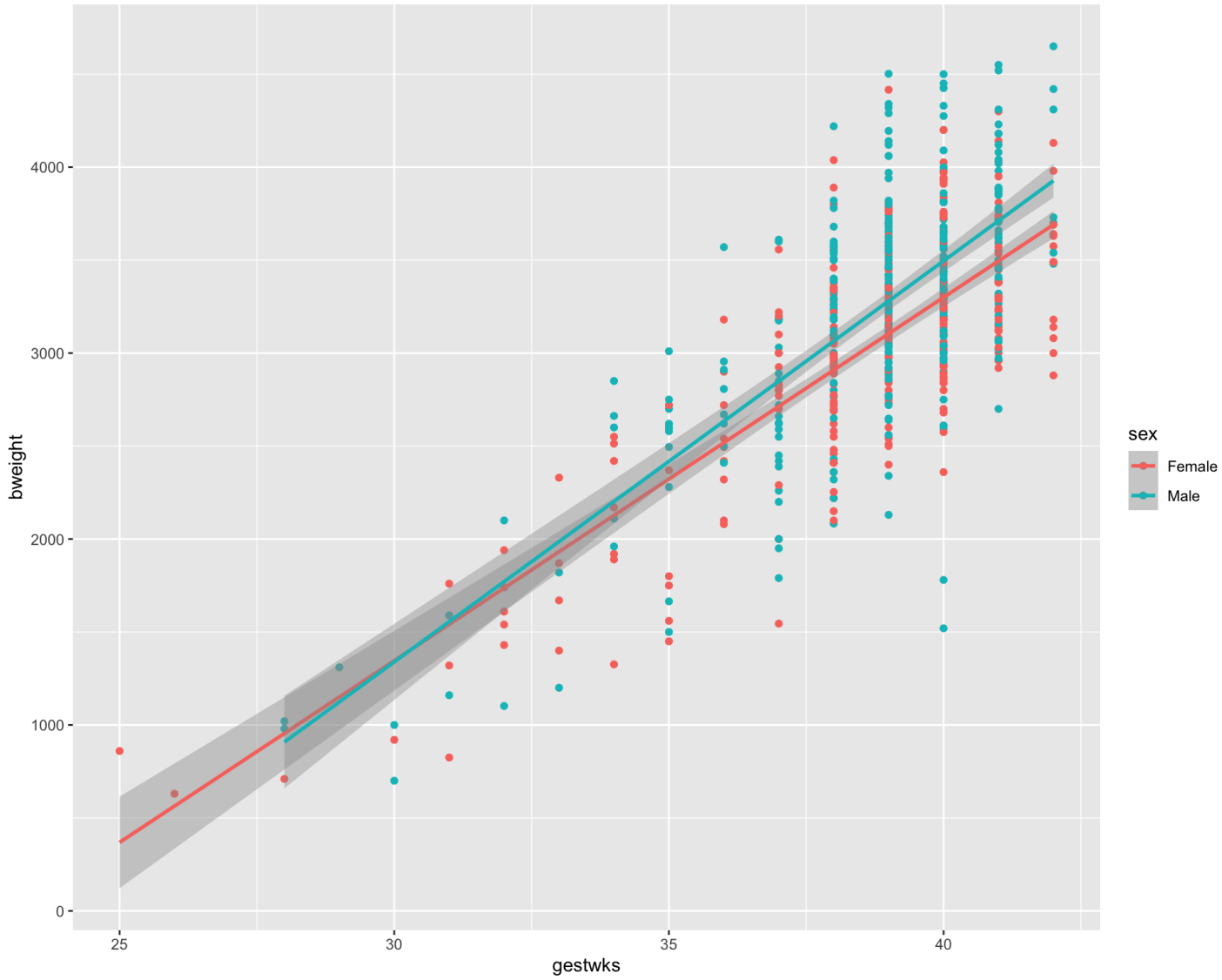


Make the smooth geom straight

- geom_smooth() is used in combination with the argument method = lm. lm stands for linear model.

```
# Start with data and aesthetics
ggplot(data = bw_df, mapping = aes(x = gestwks, y = bweight,
    color = sex)) + # Add a point geom
geom_point() + ## Add a smooth geom geom_smooth() + Make it straight
geom_smooth(method = "lm")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



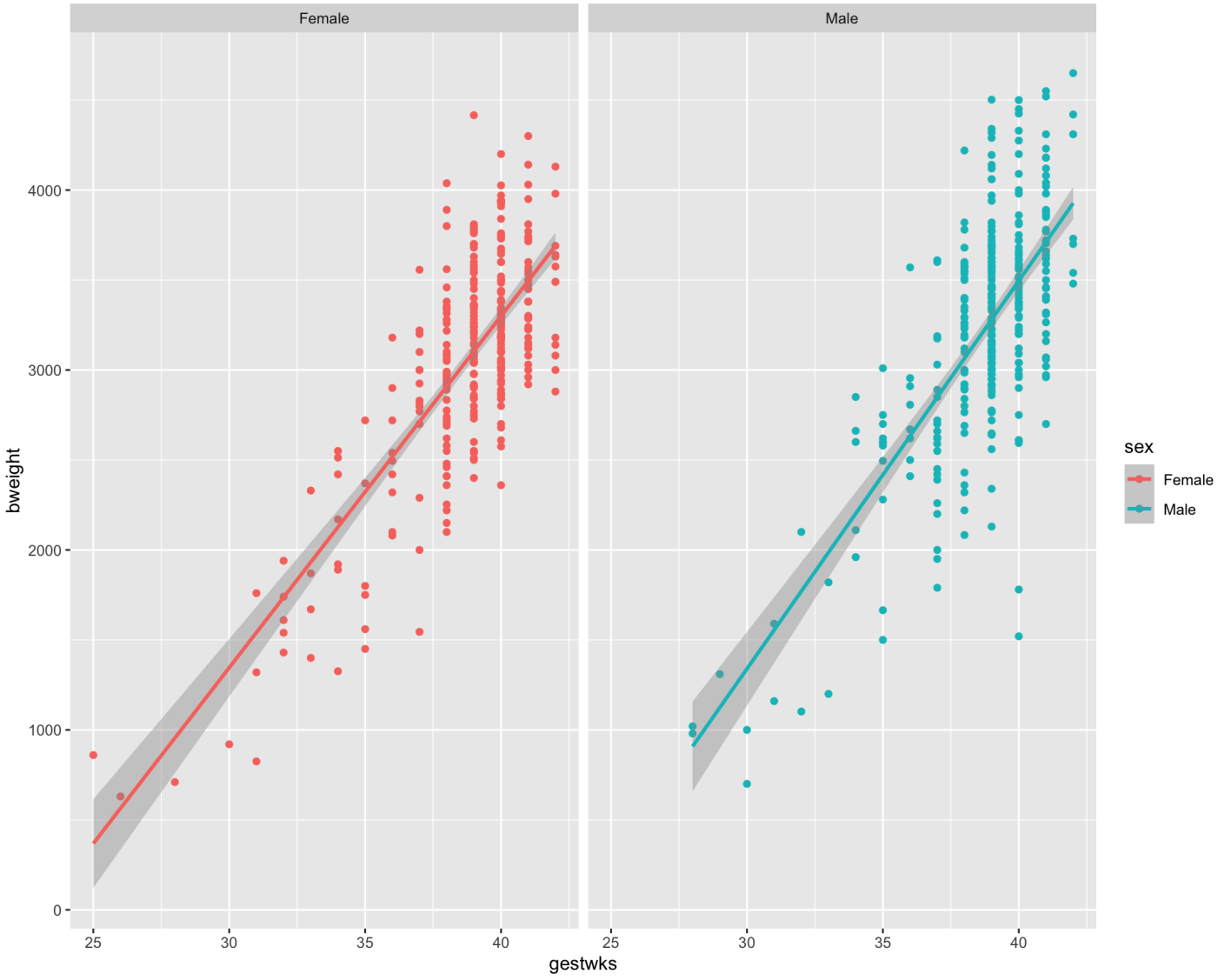
Faceting

- ggplot has a special technique called faceting that allows the user to split one plot into multiple plots based on a factor included in the dataset.
- We will use it to make a scatter plot of birth weight vs gestwks stratified by gender:
- Now we would like to split each plot by the sex of each individual measured.
- You can also organise the panels only by columns (or only by rows):

Facet by sex

```
# Start with data and aesthetics
ggplot(data = bw_df, mapping = aes(x = gestwks, y = bweight,
    color = sex)) + # Add a point geom
geom_point() + ## Add a smooth geom geom_smooth() + Make it straight
geom_smooth(method = "lm") + # Facet by sex
facet_wrap(vars(sex), ncol = 2)
```

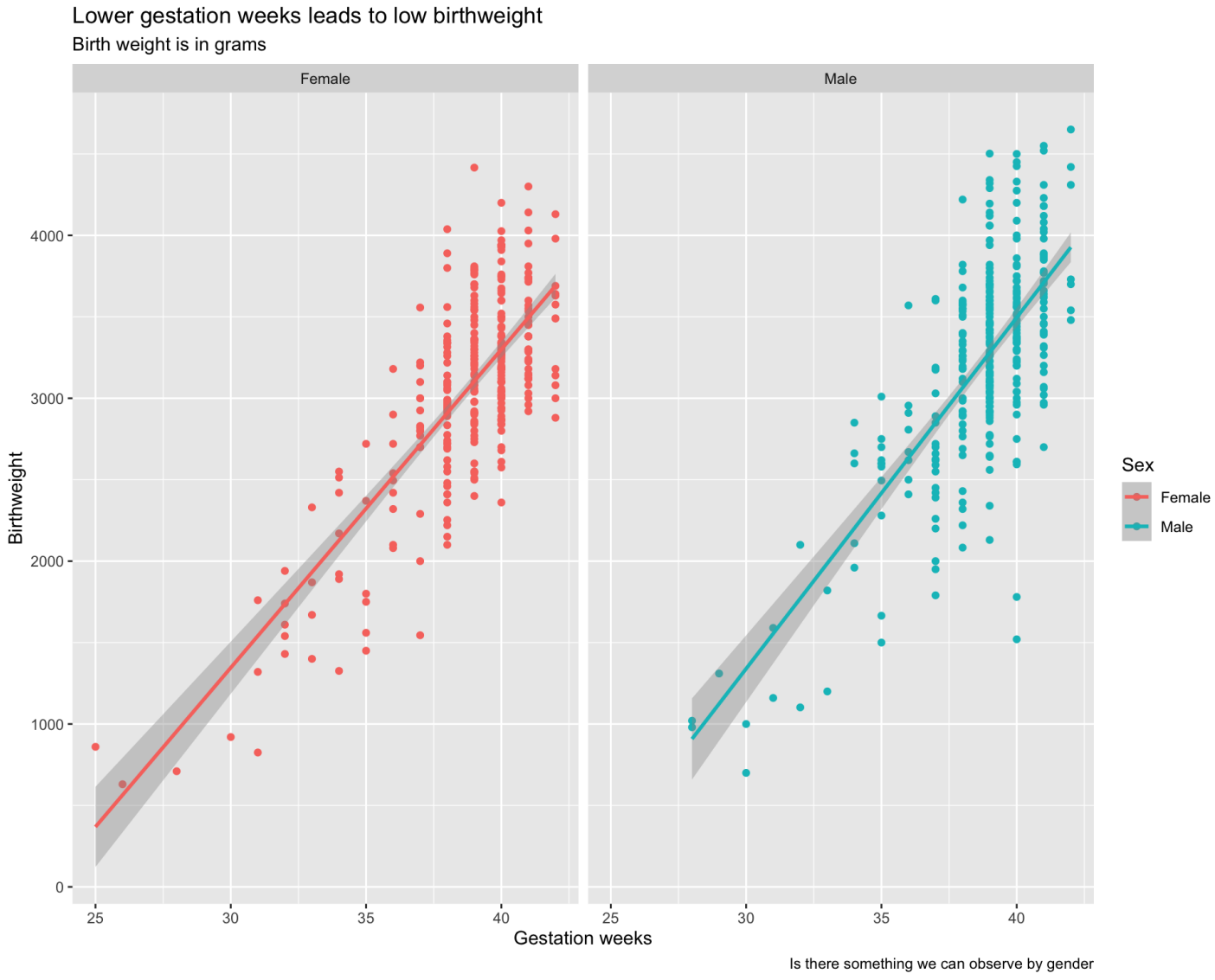
```
## `geom_smooth()` using formula 'y ~ x'
```



add labels

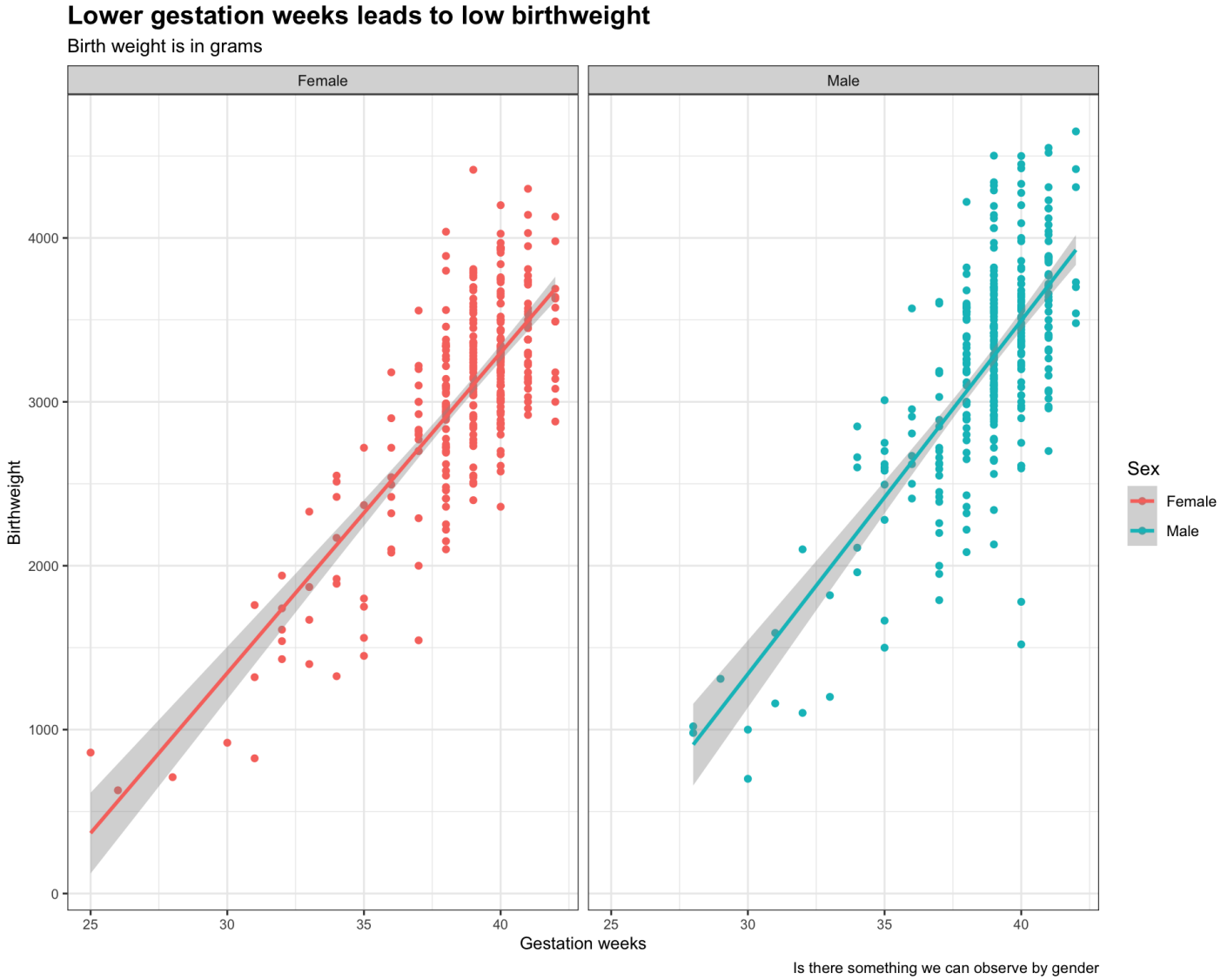
```
# Start with data and aesthetics
ggplot(data = bw_df, mapping = aes(x = gestwks, y = bweight,
    color = sex)) + # Add a point geom
geom_point() + ## Add a smooth geom geom_smooth() + Make it straight
geom_smooth(method = "lm") + # Facet by sex
facet_wrap(vars(sex), ncol = 2) + ## add labels
labs(x = "Gestation weeks", y = "Birthweight", color = "Sex",
    title = "Lower gestation weeks leads to low birthweight",
    subtitle = "Birth weight is in grams", caption = "Is there something we can observe by gender")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



adding ggplot2 themes

```
## `geom_smooth()` using formula 'y ~ x'
```



Exporting plots

```
my_plot2 <- ggplot(data = bw_df, mapping = aes(x = gestwks, y = bweight,
    color = sex)) + geom_point() + geom_smooth(method = "lm") +
    facet_wrap(vars(sex), ncol = 2) + labs(x = "Gestation weeks",
    y = "Birthweight", color = "Sex", title = "Lower gestation weeks leads to low birthweight",
    subtitle = "Birth weight is in grams", caption = "Is there something we can observe by gender") +
    theme_bw() + theme(plot.title = element_text(size = 15, face = "bold"),
    axis.text.x = element_text(size = 8), axis.text.y = element_text(size = 8),
    axis.title.x = element_text(size = 10), axis.title.y = element_text(size = 10)) +
    scale_color_discrete(name = "Sex")
ggsave("Output/Bweight.pdf", my_plot2, width = 15, height = 10)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

Useful link and resource with examples and code

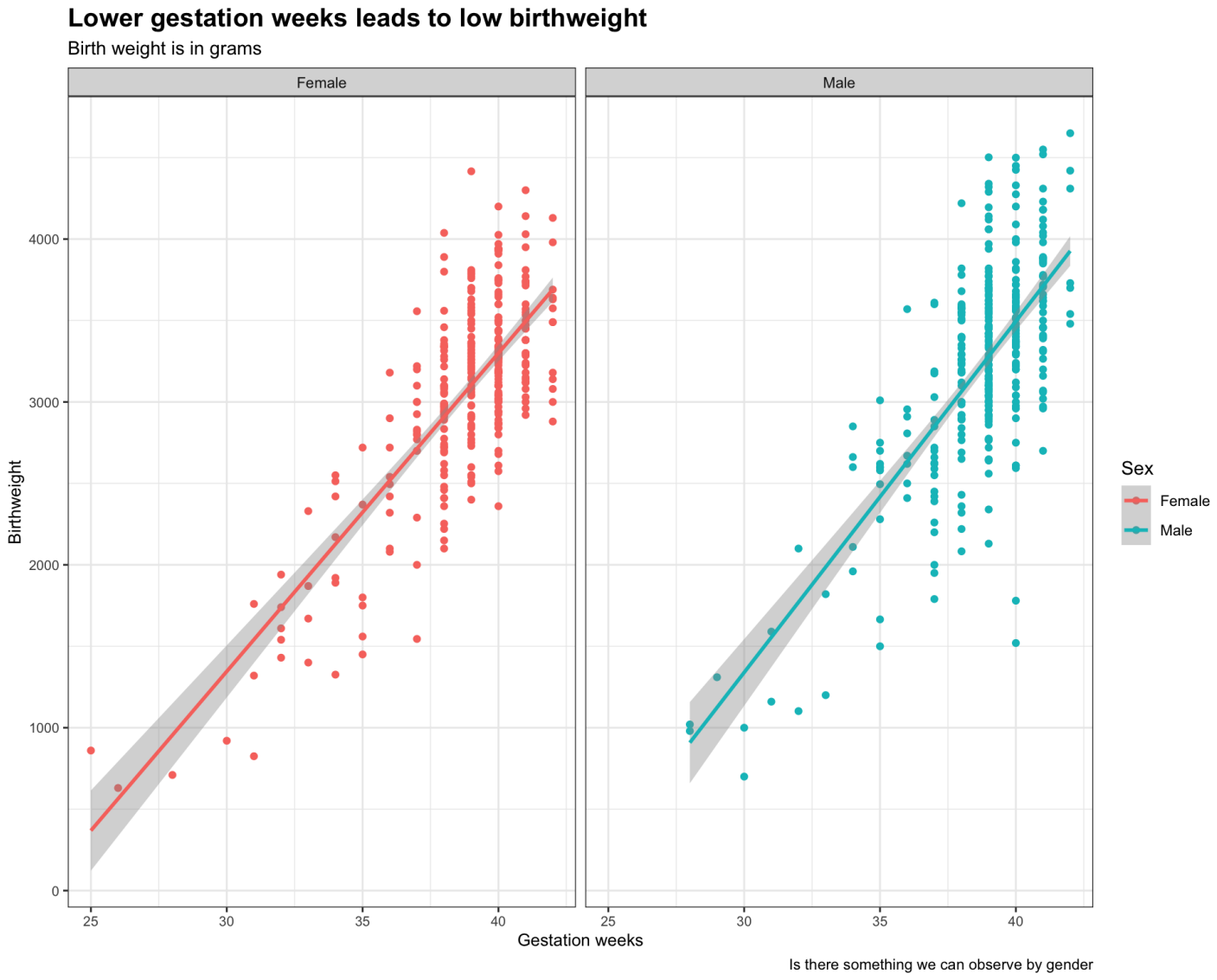
https://www.data-to-viz.com/

Break out session - Exercises

- Replicate the plot below and save

```
ggplot(data = bw_df, mapping = aes(x = gestwks, y = bweight,
    color = sex)) + geom_point() + geom_smooth(method = "lm") +
    facet_wrap(vars(sex), ncol = 2) + labs(x = "Gestation weeks",
    y = "Birthweight", color = "Sex", title = "Lower gestation weeks leads to low birthweight",
    subtitle = "Birth weight is in grams", caption = "Is there something we can observe by gender") +
    theme_bw() + theme(plot.title = element_text(size = 15, face = "bold"),
    axis.text.x = element_text(size = 8), axis.text.y = element_text(size = 8),
    axis.title.x = element_text(size = 10), axis.title.y = element_text(size = 10)) +
    scale_color_discrete(name = "Sex")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



- Can you make the shape of the points in the scatter plot to vary with ethnicity?
- add a scale shape attribute. Hint use: scale_shape_discrete(name="legend title")

- Instead of having multiple smoothing lines for each ethnic group, integrate them all under one line.

Solution

```
## `geom_smooth()` using formula 'y ~ x'
```