

Assignment 3

Mandi Bluth

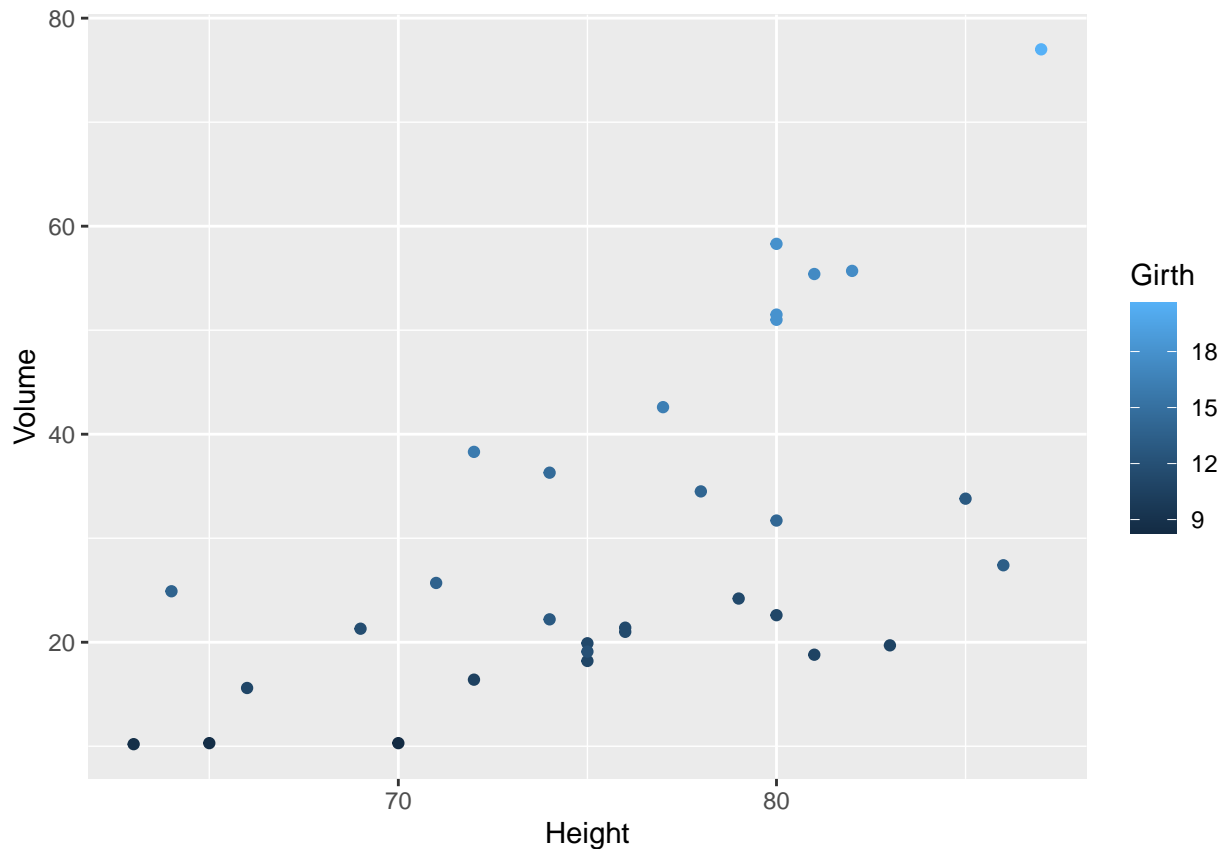
2024-09-04

Question 1

Examine the data set `trees`, which should already be pre-loaded. Look at the help file using `?trees` for more information about this data set. We wish to build a scatter plot that compares the height and girth of these cherry trees to the volume of lumber that was produced.

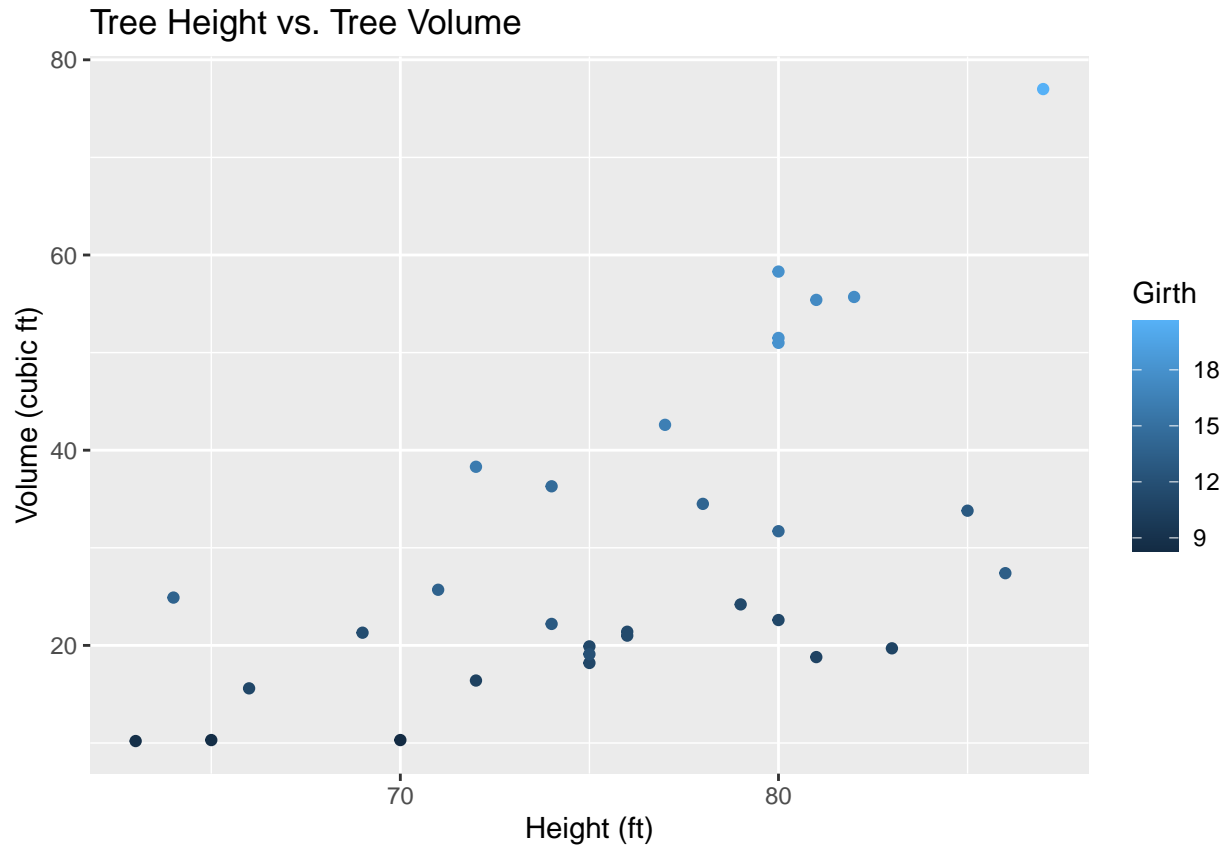
- a) Create a graph using `ggplot2` with Height on the x-axis, Volume on the y-axis, and Girth as the either the size of the data point or the color of the data point. Which do you think is a more intuitive representation?

```
data(trees)
ggplot(data=trees,
       mapping=aes(x=Height, y=Volume, color=Girth))+
geom_point()
```



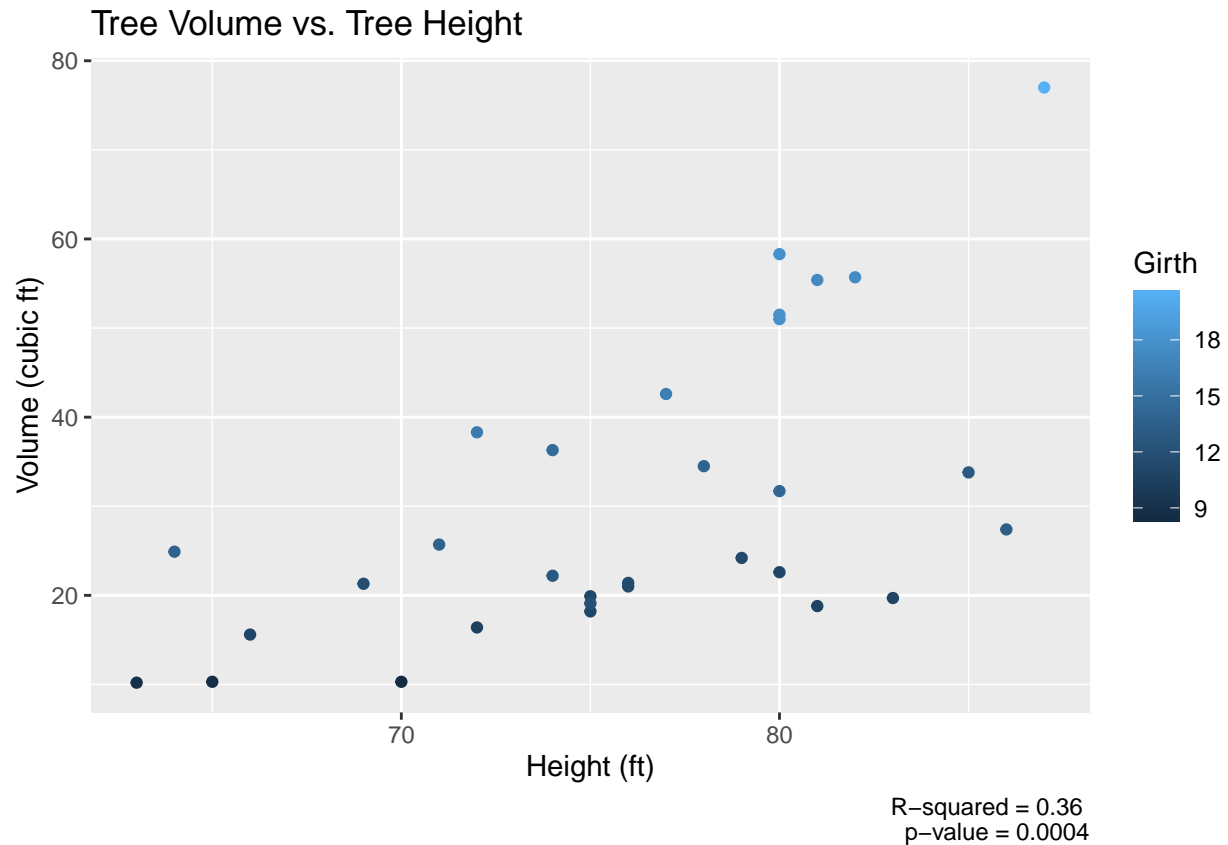
b) Add appropriate labels for the main title and the x and y axes.

```
data(trees)
ggplot(data=trees,
       mapping=aes(x=Height, y=Volume, color=Girth))+
geom_point()+
labs(x="Height (ft)", y="Volume (cubic ft)", title="Tree Height vs. Tree Volume")
```



c) The R-squared value for a regression through these points is 0.36 and the p-value for the statistical significance of height is 0.00038. Add text labels “R-squared = 0.36” and “p-value = 0.0004” somewhere on the graph.

```
ggplot(data=trees,
       mapping=aes(x=Height, y=Volume, color=Girth))+
geom_point()+
labs(x="Height (ft)", y="Volume (cubic ft)", title="Tree Volume vs. Tree Height", caption="R-squared = 0.36  
p-value = 0.0004")
```



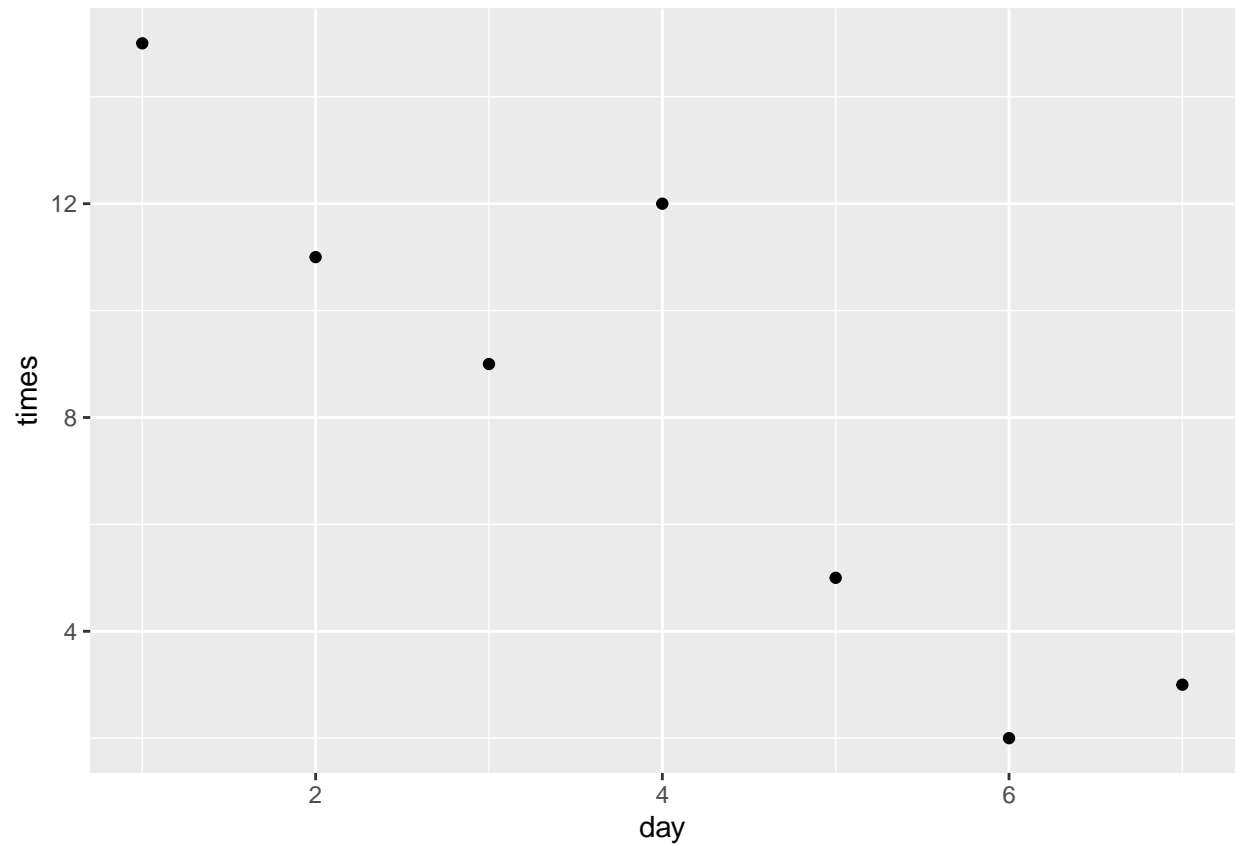
Question 2

Consider the following small data set that represents the number of times per day my wife played “Ring around the Rosy” with my daughter relative to the number of days since she has learned this game. The column `yhat` represents the best fitting line through the data, and `lwr` and `upr` represent a 95% confidence interval for the predicted value on that day. Because these questions ask you to produce several graphs and evaluate which is better and why, please include each graph and response with each sub-question.

```
Rosy <- data.frame(
  times = c(15, 11, 9, 12, 5, 2, 3),
  day    = 1:7,
  yhat   = c(14.36, 12.29, 10.21, 8.14, 6.07, 4.00, 1.93),
  lwr    = c( 9.54,  8.5,   7.22,  5.47,  3.08,  0.22, -2.89),
  upr    = c(19.18, 16.07, 13.2,  10.82,  9.06,  7.78,  6.75))
```

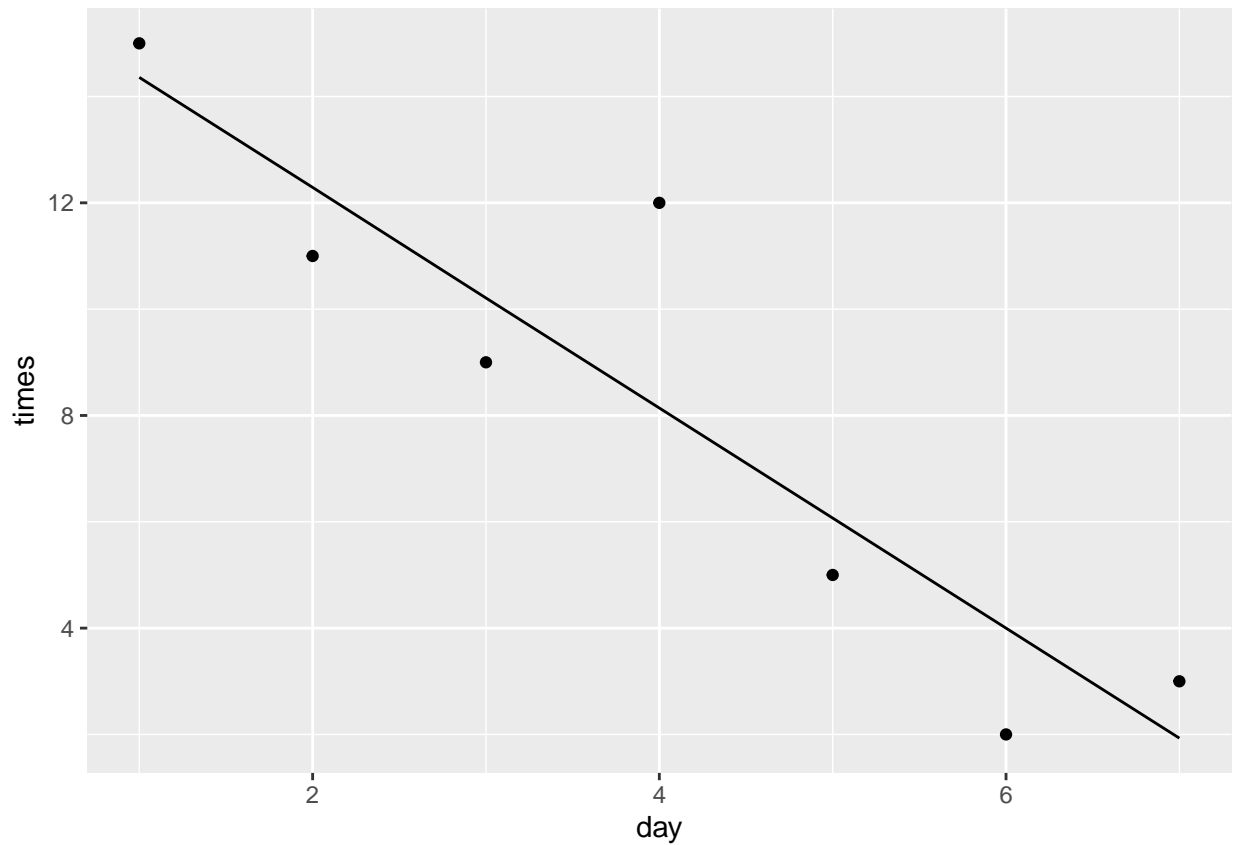
a) Using `ggplot()` and `geom_point()`, create a scatterplot with `day` along the x-axis and `times` along the y-axis.

```
ggplot(data=Rosy,
  mapping=aes(x=day, y=times))+
  geom_point()
```



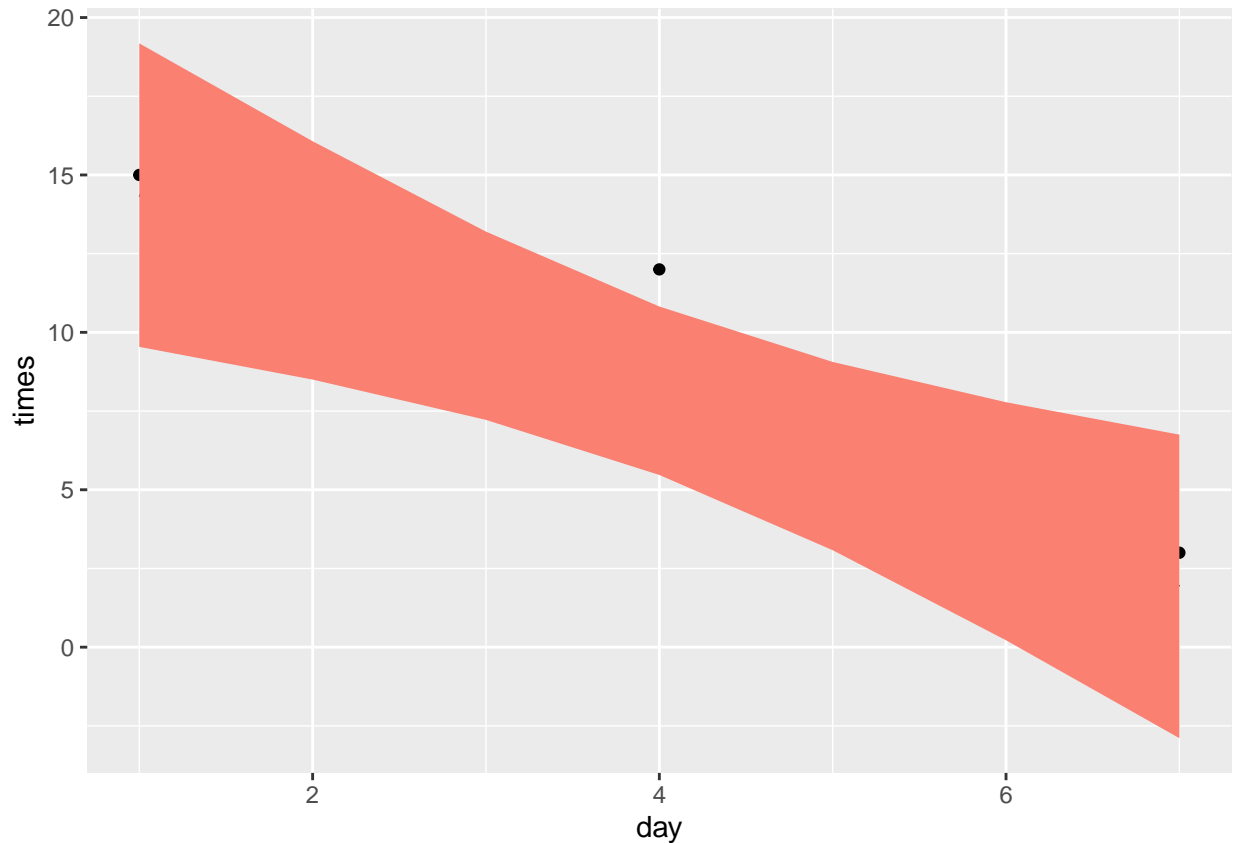
- b) Add a line to the graph where the x-values are the day values but now the y-values are the predicted values which we've called yhat. Notice that you have to set the aesthetic `y=times` for the points and `y=yhat` for the line. Because each `geom_` will accept an `aes()` command, you can specify the `y` attribute to be different for different layers of the graph.

```
ggplot(data=Rosy,  
       mapping=aes(x=day, y=times))+  
geom_point()+  
geom_line(aes(y=yhat))
```



- c) Add a ribbon that represents the confidence region of the regression line. The `geom_ribbon()` function requires an `x`, `ymin`, and `ymax` columns to be defined. For examples of using `geom_ribbon()` see the online documentation: https://ggplot2.tidyverse.org/reference/geom_ribbon.html.

```
ggplot(Rosy, aes(x=day)) +  
  geom_point(aes(y=times)) +  
  geom_line(aes(y=yhat)) +  
  geom_ribbon(aes(ymin=lwr, ymax=upr), fill='salmon')
```

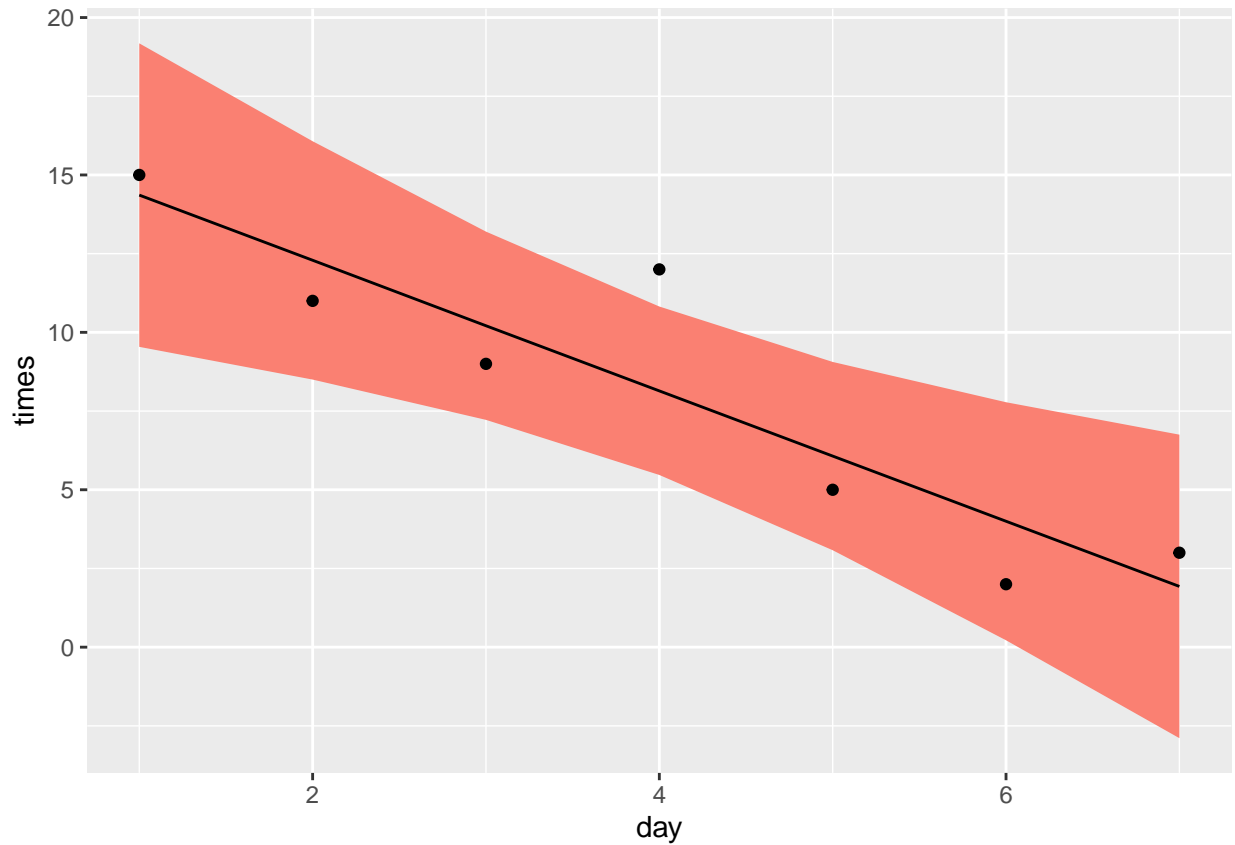


d) What happened when you added the ribbon? Did some points get hidden? If so, why?

The ribbon was added on top of the line and points so they are hidden, however, you can reorder the commands to make the points and lines visible again.

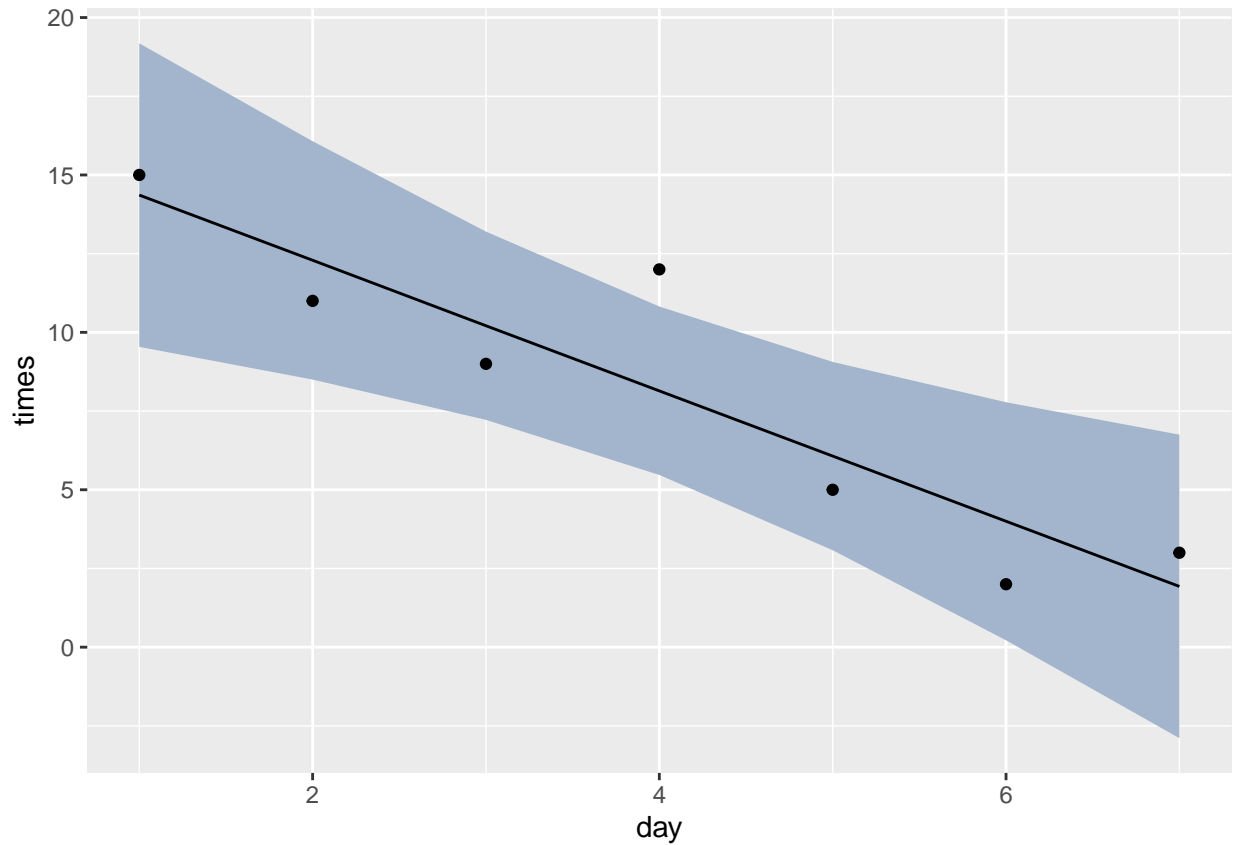
e) Reorder the statements that created the graph so that the ribbon is on the bottom and the data points are on top and the regression line is visible.

```
ggplot(Rosy, aes(x=day)) +
  geom_ribbon(aes(ymin=lwr, ymax=upr), fill='salmon') +
  geom_point(aes(y=times)) +
  geom_line(aes(y=yhat))
```



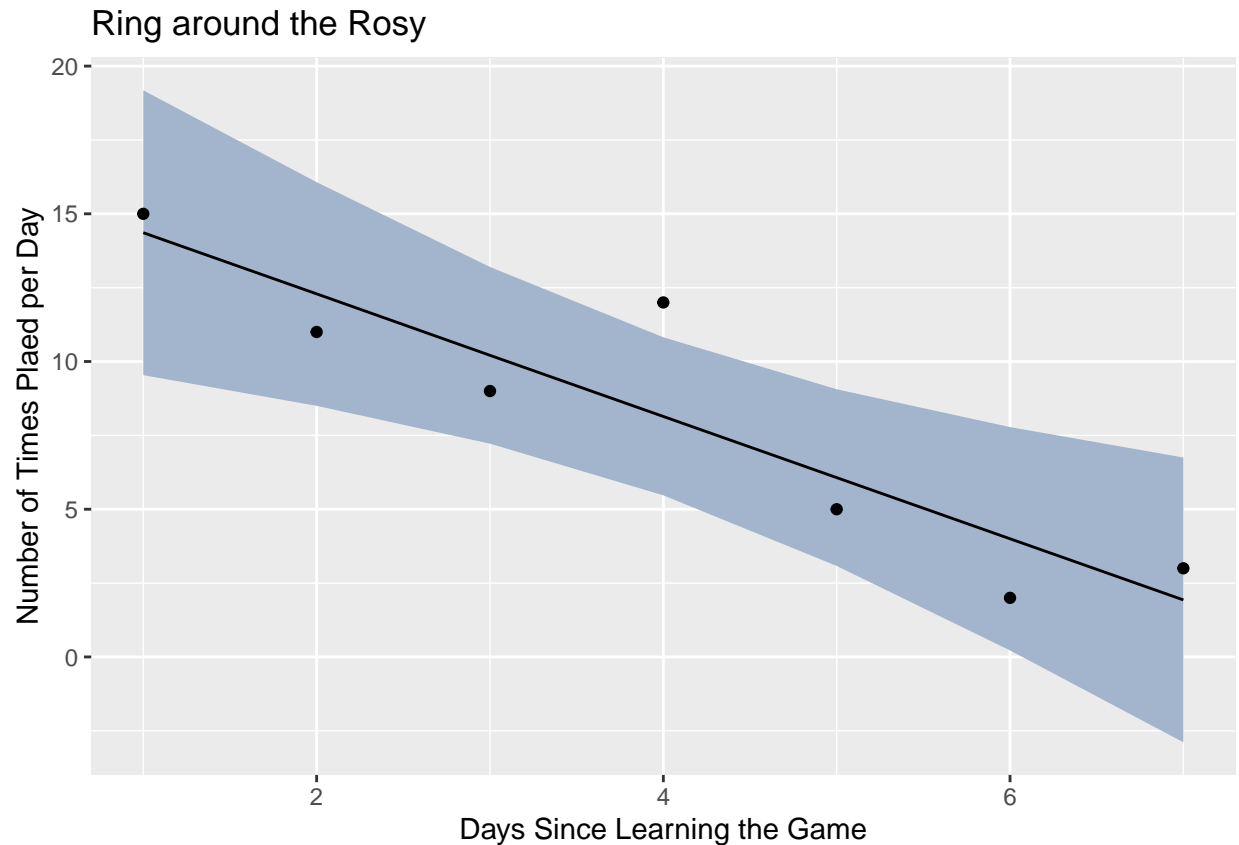
f) The color of the ribbon fill is ugly. Use Google to find a list of named colors available to ggplot2. For example, I googled “ggplot2 named colors” and found the following link: <http://sape.inf.usi.ch/quick-reference/ggplot2/colour>. Choose a color for the fill that is pleasing to you.

```
ggplot(Rosy, aes(x=day)) +
  geom_ribbon( aes(ymin=lwr, ymax=upr), fill='lightsteelblue3')+
  geom_point(aes(y=times)) +
  geom_line( aes(y=yhat))
```



g) Add labels for the x-axis and y-axis that are appropriate along with a main title.

```
ggplot(Rosy, aes(x=day)) +
  geom_ribbon( aes(ymin=lwr, ymax=upr), fill='lightsteelblue3') +
  geom_point(aes(y=times)) +
  geom_line( aes(y=yhat)) +
  labs(x="Days Since Learning the Game", y="Number of Times Plaed per Day", title = "Ring around the Ro")
```

Question 3

We'll next make some density plots that relate several factors towards the birth weight of a child. Because these questions ask you to produce several graphs and evaluate which is better and why, please include each graph and response with each sub-question.

- a) The MASS package contains a data set called `birthwt` which contains information about 189 babies and their mothers. In particular there are columns for the mother's race and smoking status during the pregnancy. Load the `birthwt` by either using the `data()` command or loading the MASS library.

```
data("birthwt")
```

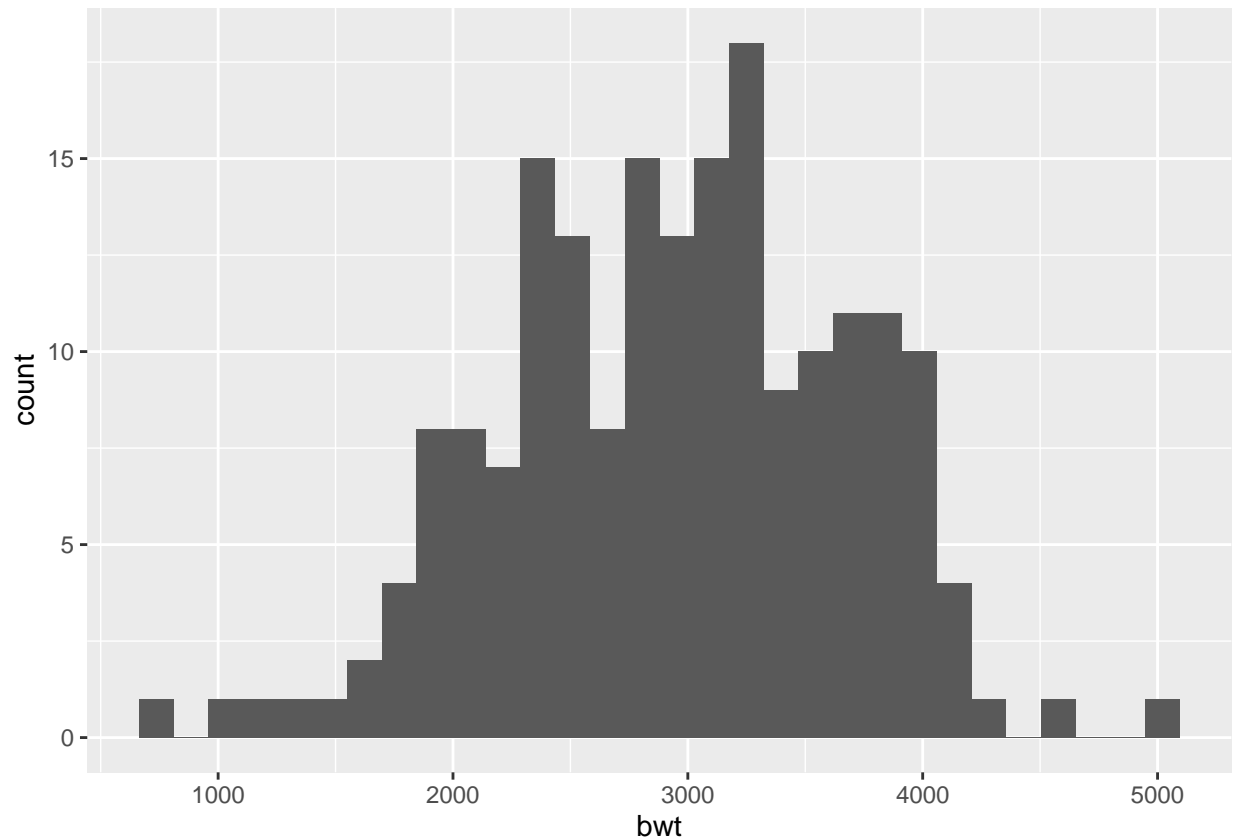
- b) Read the help file for the data set using `MASS::birthwt`. The covariates `race` and `smoke` are not stored in a user friendly manner. For example, smoking status is labeled using a 0 or a 1. Because it is not obvious which should represent that the mother smoked, we'll add better labels to the `race` and `smoke` variables. For more information about dealing with factors and their levels, see the Factors chapter in these notes.

```
library(tidyverse)
data('birthwt', package='MASS')
birthwt <- birthwt %>% mutate(
  race = factor(race, labels=c('White', 'Black', 'Other')),
  smoke = factor(smoke, labels=c('No Smoke', 'Smoke')))
```

c) Graph a histogram of the birth weights bwt using `ggplot(birthwt, aes(x=bwt)) + geom_histogram()`.

```
ggplot(data=birthwt,  
       mapping=aes(x=bwt))+  
geom_histogram()
```

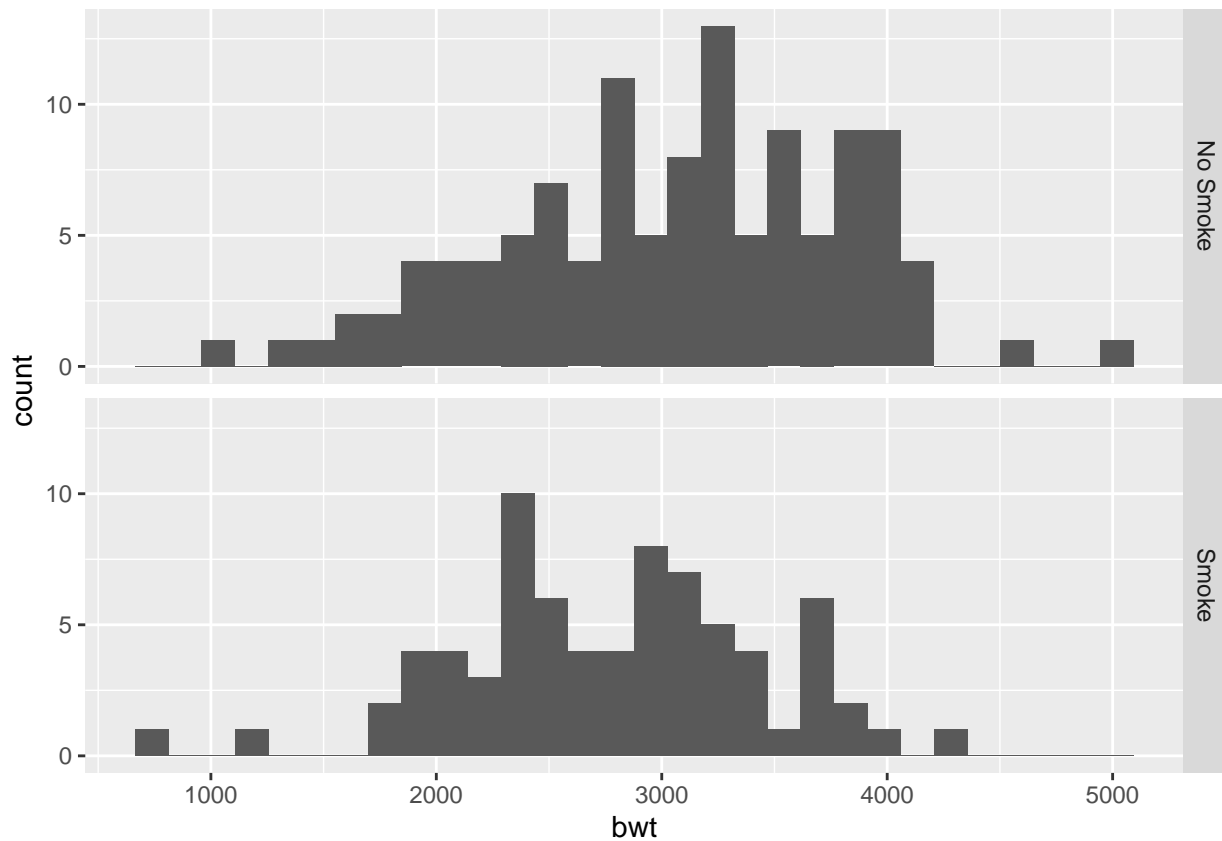
```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```



d) Make separate graphs that denote whether a mother smoked during pregnancy by appending `facet_grid()` command to your original graphing command.

```
ggplot(data=birthwt,  
       mapping=aes(x=bwt))+  
geom_histogram()+  
facet_grid( rows = vars(smoke))
```

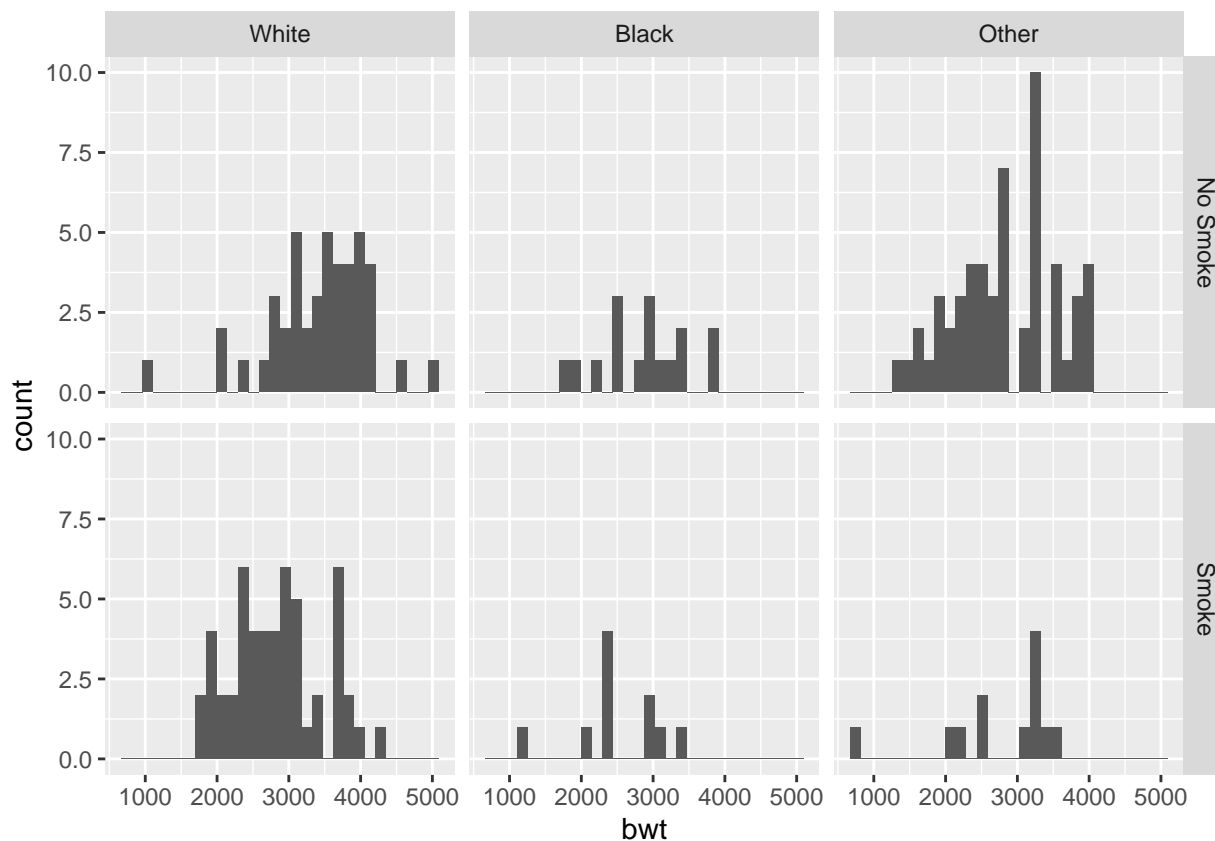
```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```



e) Perhaps race matters in relation to smoking. Make our grid of graphs vary with smoking status changing vertically, and race changing horizontally (that is the formula in `facet_grid()` should have smoking be the y variable and race as the x).

```
ggplot(data=birthwt,
       mapping=aes(x=bwt))+
geom_histogram()+
facet_grid( smoke ~ race)
```

```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```

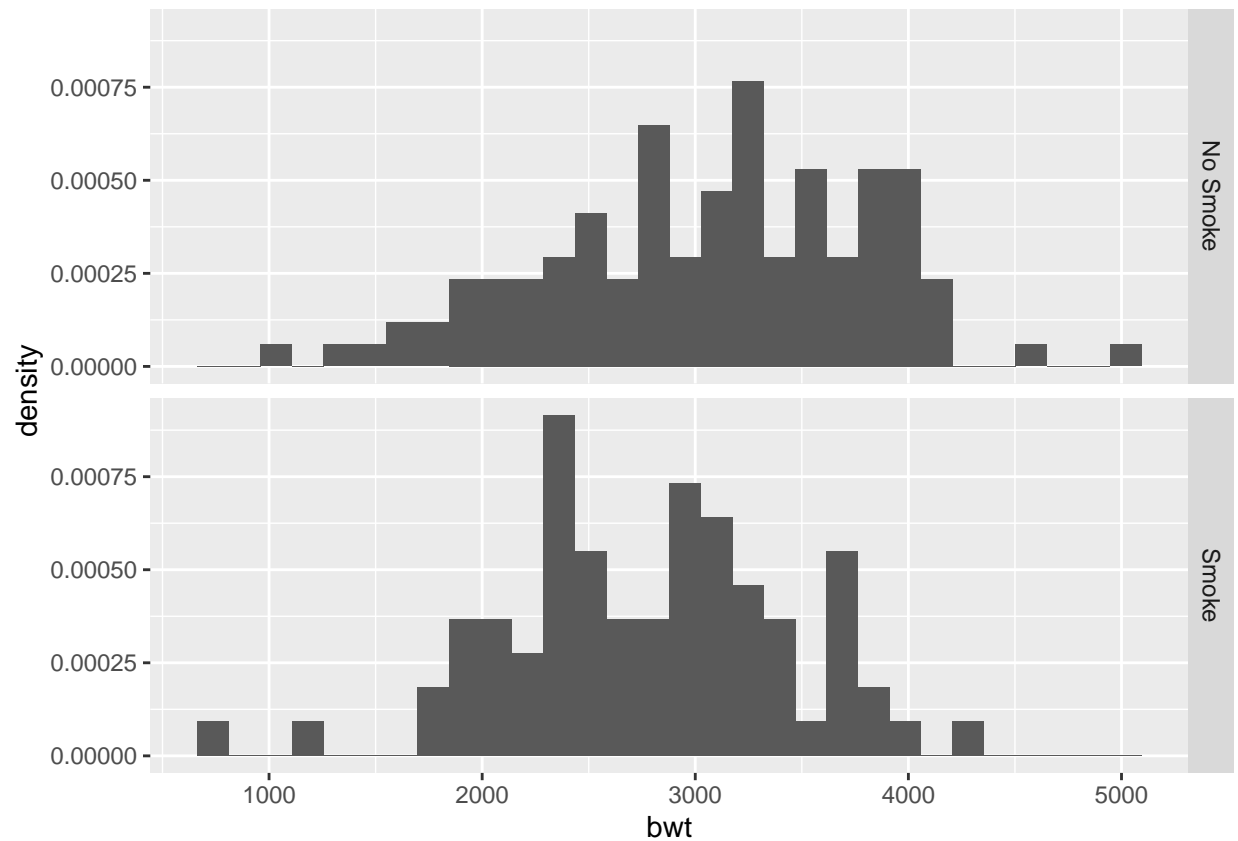


- f) Remove race from the facet grid, (so go back to the graph you had in part d). I'd like to next add an estimated density line to the graphs, but to do that, I need to first change the y-axis to be density (instead of counts), which we do by using `aes(y=..density..)` in the `ggplot()` aesthetics command.

```
ggplot(data=birthwt,
       mapping=aes(x=bwt, y=..density..))+
geom_histogram()+
facet_grid( rows = vars(smoke))
```

```
## Warning: The dot-dot notation ('..density..') was deprecated in ggplot2 3.4.0.
## i Please use 'after_stat(density)' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```

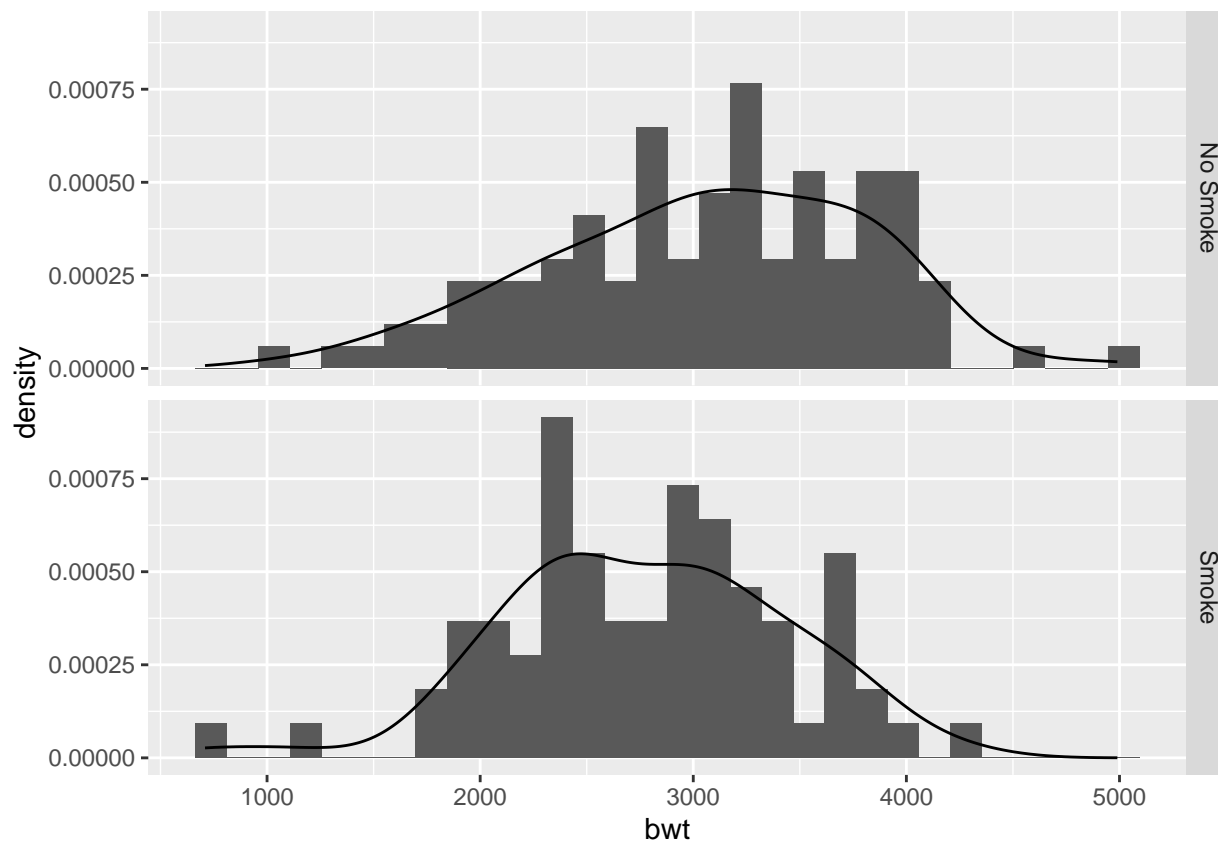
```
## 'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.
```



g) Next we can add the estimated smooth density using the `geom_density()` command.

```
ggplot(data=birthwt,
       mapping=aes(x=bwt, y=..density..))+
  geom_histogram()+
  facet_grid( rows = vars(smoke))+
  geom_density()
```

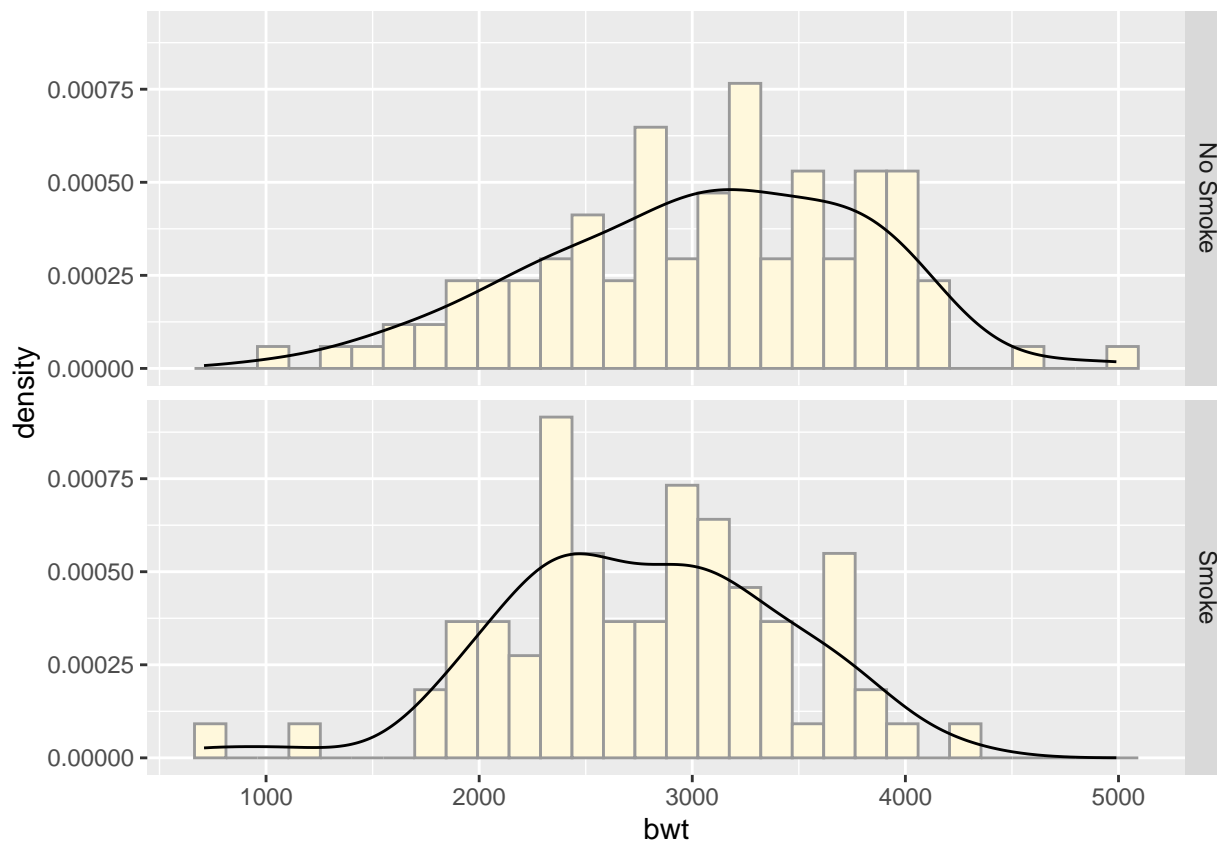
'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.



h) To really make this look nice, let's change the fill color of the histograms to be something less dark, let's use `fill='cornsilk'` and `color='grey60'`. To play with different colors that have names, check out the following: <https://www.datanovia.com/en/blog/awesome-list-of-657-r-color-names/>.

```
ggplot(data=birthwt,
       mapping=aes(x=bwt, y=..density..))+
  geom_histogram(fill='cornsilk', color='grey60')+
  facet_grid( rows = vars(smoke))+
  geom_density()
```

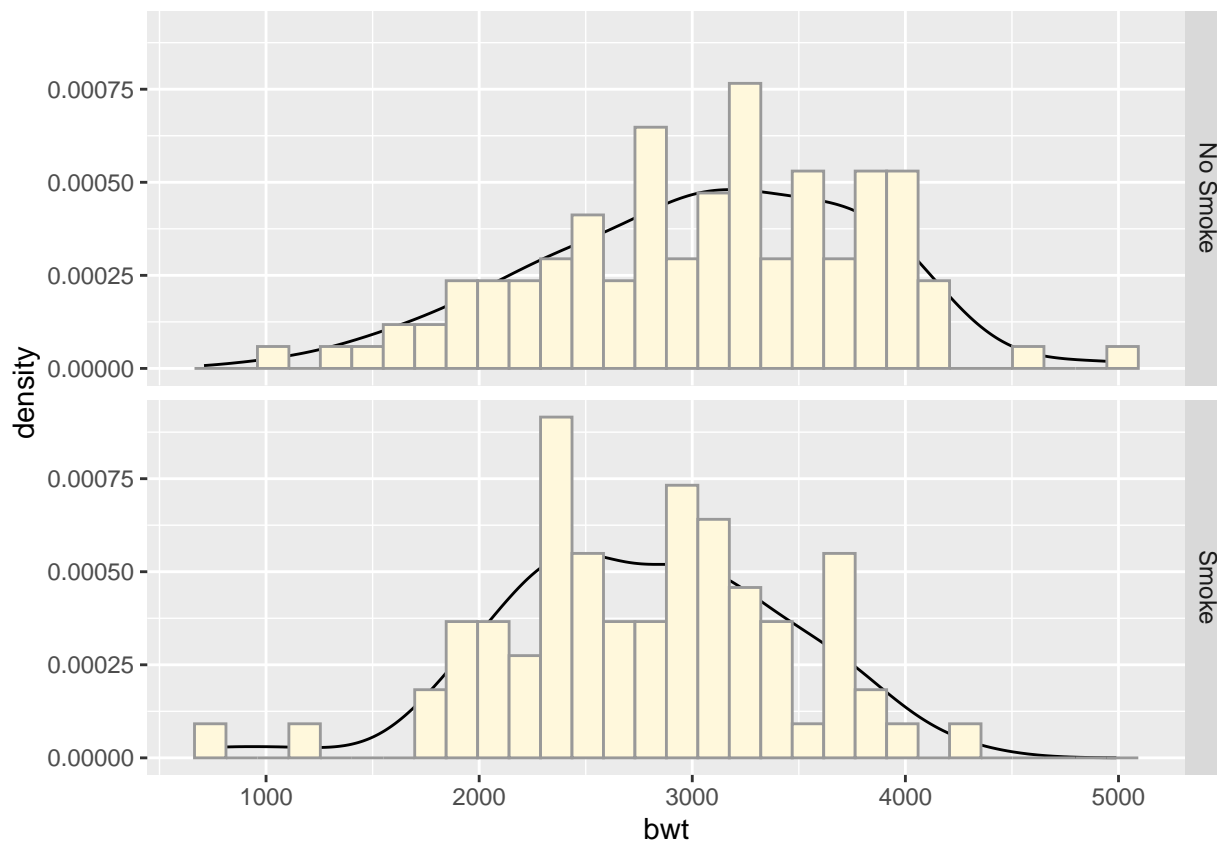
'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.



- i) Change the order in which the histogram and the density line are added to the plot. Does it matter and which do you prefer?

```
ggplot(data=birthwt,
       mapping=aes(x=bwt, y=..density..))+
  geom_density()+
  geom_histogram(fill='cornsilk', color='grey60')+
  facet_grid( rows = vars(smoke))
```

'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.

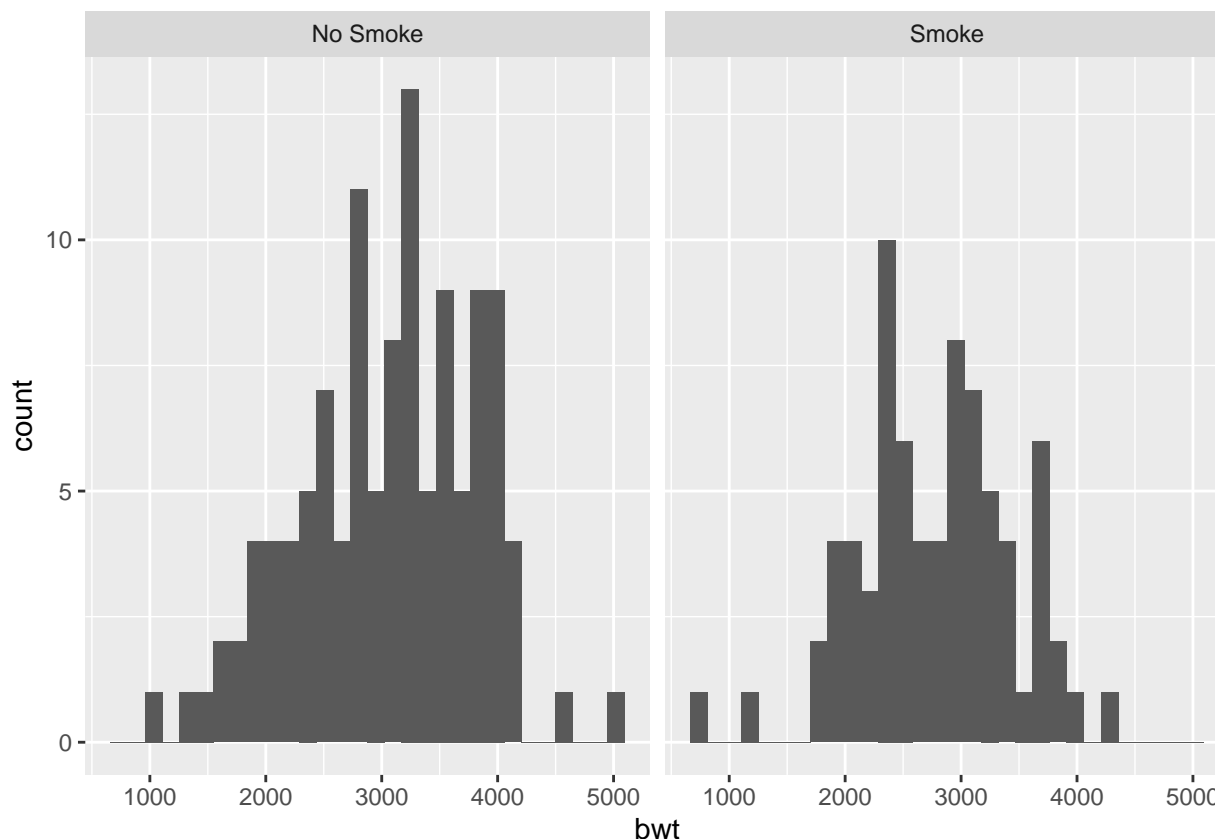


When you put the density line first, it gets a little hidden behind the bins. I think that having the line in front is more practical because the density line doesn't interfere with viewing the bins, however, the bins in front interferes with viewing the line.

- j) Finally consider if you should have the histograms side-by-side or one on top of the other (i.e. $\cdot \sim$ smoke or smoke $\sim \cdot$). Which do you think better displays the decrease in mean birth weight and why?

```
ggplot(data=birthwt,
       mapping=aes(x=bwt))+
geom_histogram()+
facet_grid( cols = vars(smoke))
```

'stat_bin()' using 'bins = 30'. Pick better value with 'binwidth'.



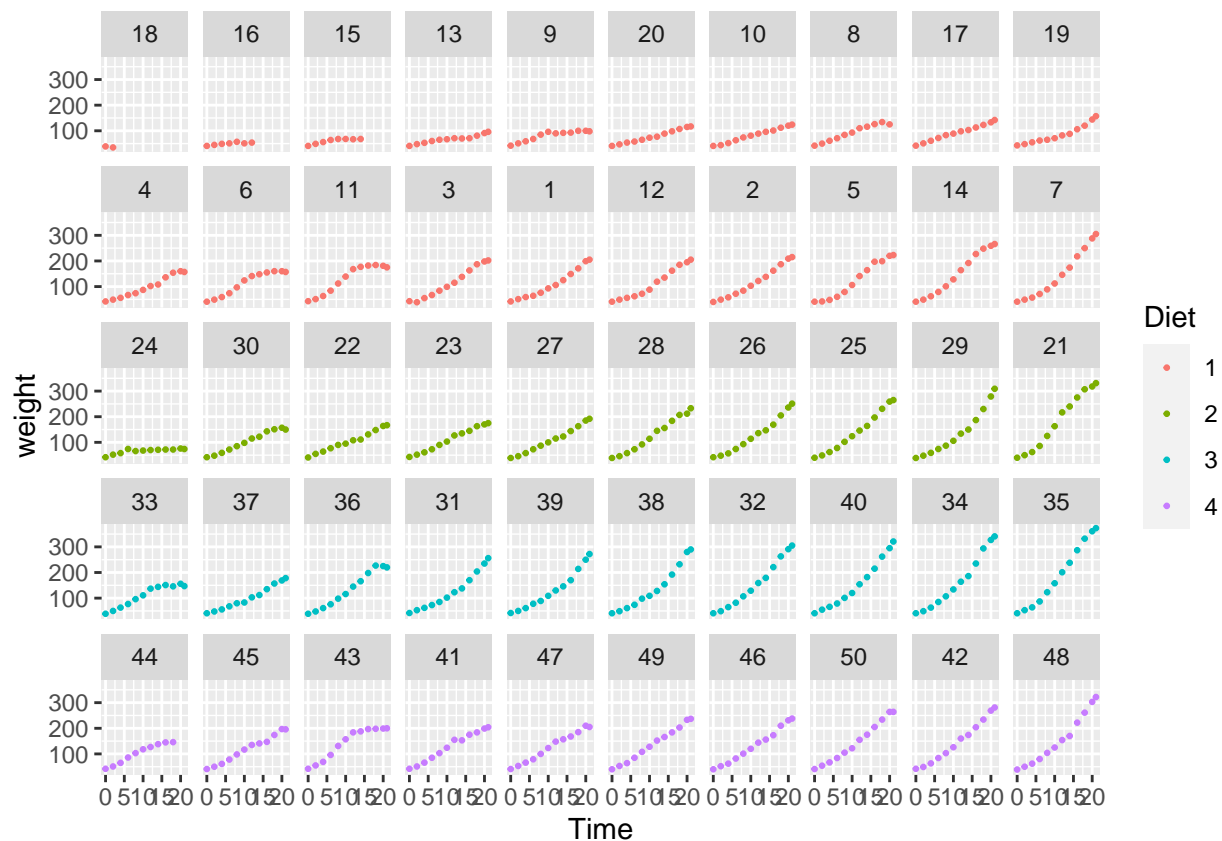
In this case, side-by-side is ideal for direct comparison of density, however, in other cases there may be limited horizontal space and a stacked graph may look more detailed.

Question 4

Load the data set ChickWeight, which comes pre-loaded in R, and get the background on the data set by reading the manual page ?ChickWeight. Because these questions ask you to produce several graphs and evaluate which is better and why, please include each graph and response with each sub-question.

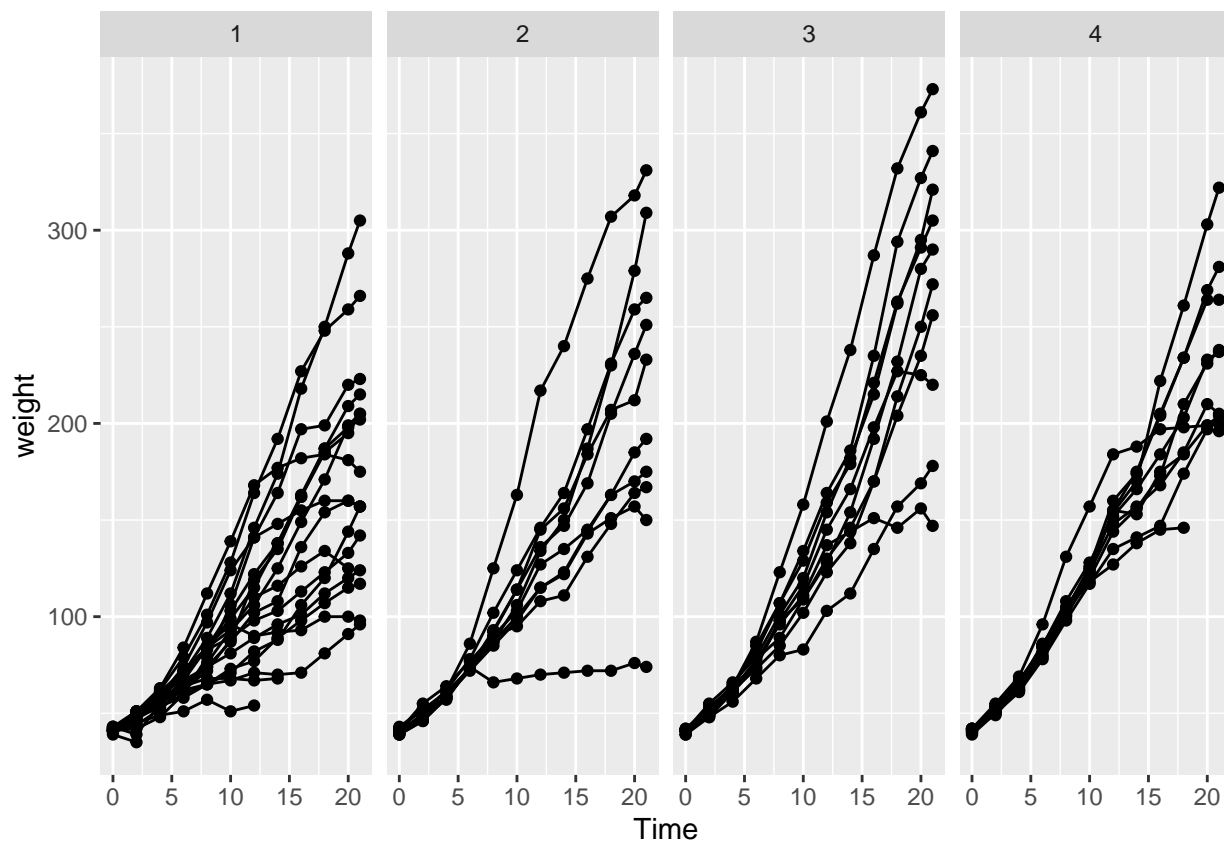
- Produce a separate scatter plot of weight vs age for each chick. Use color to distinguish the four different Diet treatments. Note, this question should produce 50 separate graphs! If the graphs are too squished you should consider how to arrange them so that the graphs wrap to a new row of graphs in the resulting output figure. The results are messy!

```
ggplot(data=ChickWeight,
       mapping=aes(x = Time, y = weight, color =Diet)) +
  geom_point(size = 0.5) +
  facet_wrap(~ Chick, ncol = 10)
```



b) We could examine these data by producing a scatter plot for each diet. Most of the code below is readable, but if we don't add the group aesthetic the lines would not connect the dots for each Chick but would instead connect the dots across different chicks.

```
ggplot(ChickWeight, aes(x=Time, y=weight, group=Chick )) +
  geom_point() + geom_line() +
  facet_grid( ~ Diet)
```



Notice in the code chunk above, if you copied from the online source code you must remove the `eval=FALSE` in the chunk header. This option allows the code to be displayed, but it won't be run and no plot will be produced in your final output document. So when you ask, why don't I see a plot?, I'll remind you of this statement!