Lab Session 2: Plotting in R

Isabel Fulcher

3/09/2021

Set working directory

We will first set a working directory. Your working directory should be where your plan to save your R code and also where the example datasets have been stored. In the "Files" tab, navigate to this folder. Once you are there, select the "More" dropdown and select "Set As Working Directory".

Install and load R packages

You should have already installed the tidyverse package. Now, you will need to load the package into R. This will allow you to use the functionality of the package.

```
library(tidyverse)
library(lubridate)
```

Load and view data

Load the Facility B outpatient visits ".rds" file and save it as a data frame called "facility". An .rds file is an R object and is the best way to store data that will be used in R.

```
facility <- readRDS("session2_data/example_facility_B.rds")
```

View the first six observations in the facility dataset.

```
head(facility)
```

```
## # A tibble: 6 x 2
##    date        count
##    <date>      <dbl>
## 1 2016-01-01    557
## 2 2016-02-01    574
## 3 2016-03-01    542
## 4 2016-04-01    793
## 5 2016-05-01    605
## 6 2016-06-01    612
```

Summarizing the data

How many months are in the dataset?

```
nrow(facility)
```

```
## [1] 57
```

What is the date range in the dataset?

```
facility %>%
  summarize(min(date),
            max(date))
```

```
## # A tibble: 1 x 2
##    `min(date)` `max(date)`
```

```
##   <date>      <date>
## 1 2016-01-01  2020-09-01
```

What is the mean (average) number of monthly outpatient visits?

```
facility %>%
  summarize(mean(count))
```

```
## # A tibble: 1 x 1
##   `mean(count)`
##           <dbl>
## 1          539.
```

What is the maximum and minimum number of monthly outpatient visits?

```
facility %>%
  summarize(min(count),
            max(count))
```

```
## # A tibble: 1 x 2
##   `min(count)` `max(count)`
##          <dbl>        <dbl>
## 1          352          793
```
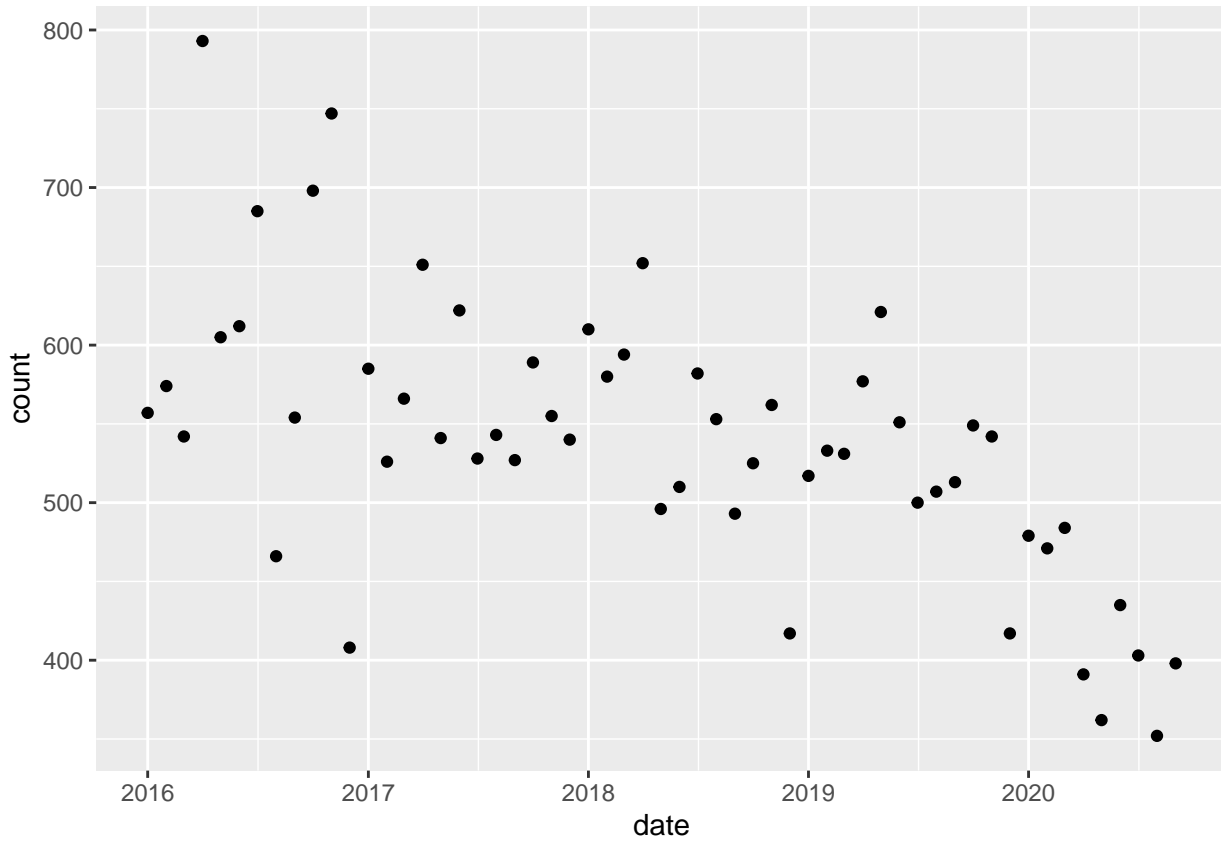
Visualizing the data

Create a histogram of the monthly outpatient visits counts.

```
ggplot(facility,aes(count)) +
  geom_histogram(color="black",fill="lightblue",bins = 20) +
  theme_bw()
```
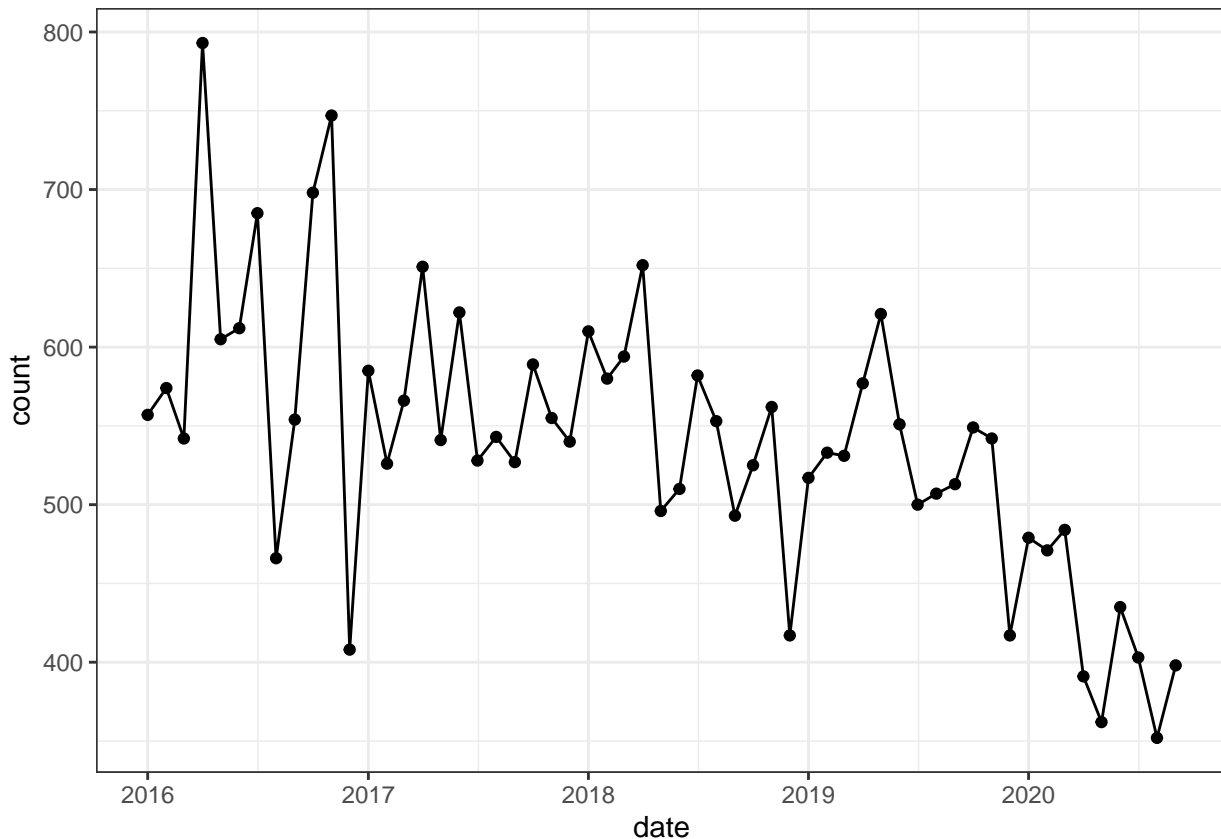
Create a scatter plot of the counts over time.

```
ggplot(facility,aes(x=date,y=count)) +
  geom_point()
```

In the above plot, connect the counts with a line.

```r
ggplot(facility,aes(x=date,y=count)) +
  geom_point() +
  geom_line() +
  theme_bw()
```

ACTIVITY: Add aesthetics to the above plot by changing the various inputs (see some options in code chunk below).

```
ggplot(facility,aes(x=date,y=count)) +
  geom_point(color="black",size=2,shape=16) +
  geom_line(color="black",size=.5,linetype="solid") +
  ylab("count") +
  xlab("date") +
  ggtitle("") +
  theme_bw()
```

Linear regression and plotting output

Create a new variable for each month.

```
facility %>%
  arrange(date) %>%
  mutate(month = 1:n()) -> facility.new
```

Fit a linear regression with an intercept and term for time.

```
fit.lm <- lm(count ~ month, data=facility.new)

summary(fit.lm)

##
## Call:
## lm(formula = count ~ month, data = facility.new)
##
```
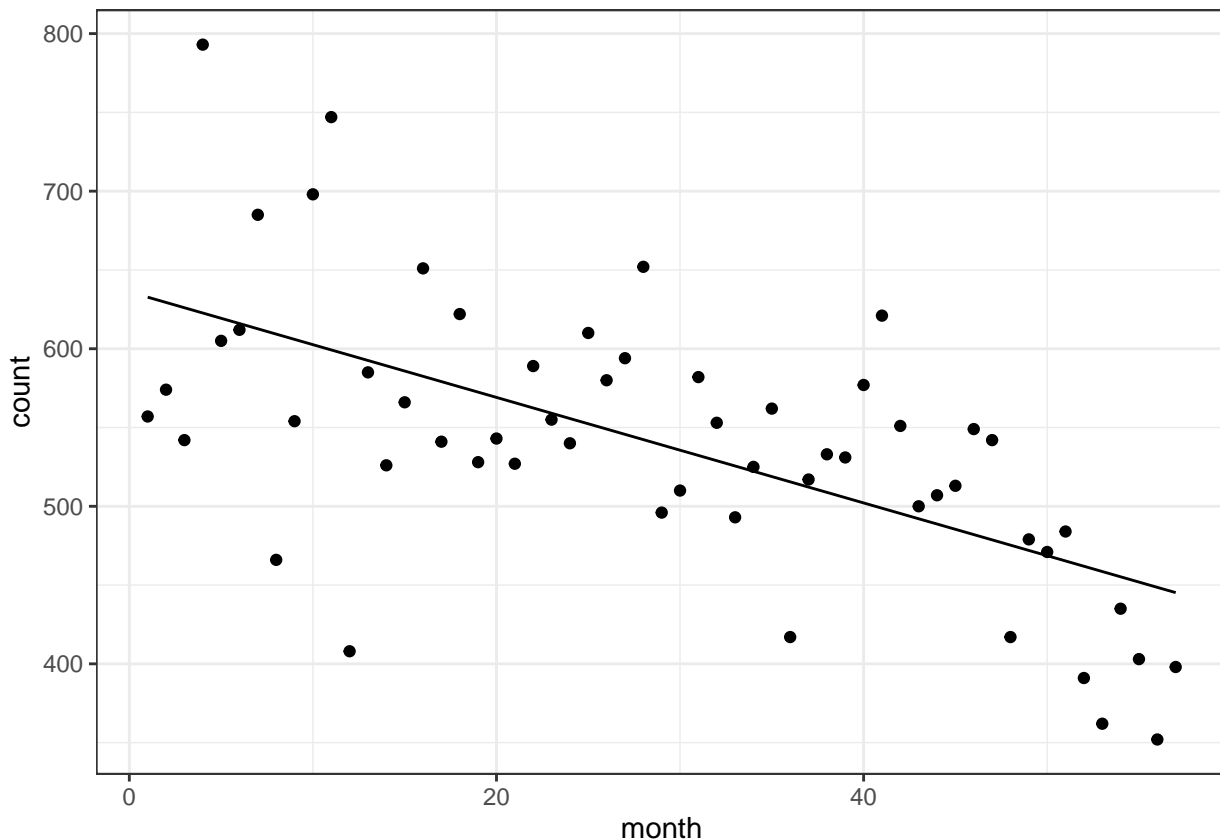
```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -187.891  -44.451    2.355   46.201  170.320
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 636.0739    18.5113  34.361  < 2e-16 ***
## month        -3.3486     0.5552  -6.031 1.43e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 68.96 on 55 degrees of freedom
## Multiple R-squared:  0.3981, Adjusted R-squared:  0.3872
## F-statistic: 36.38 on 1 and 55 DF,  p-value: 1.432e-07
```
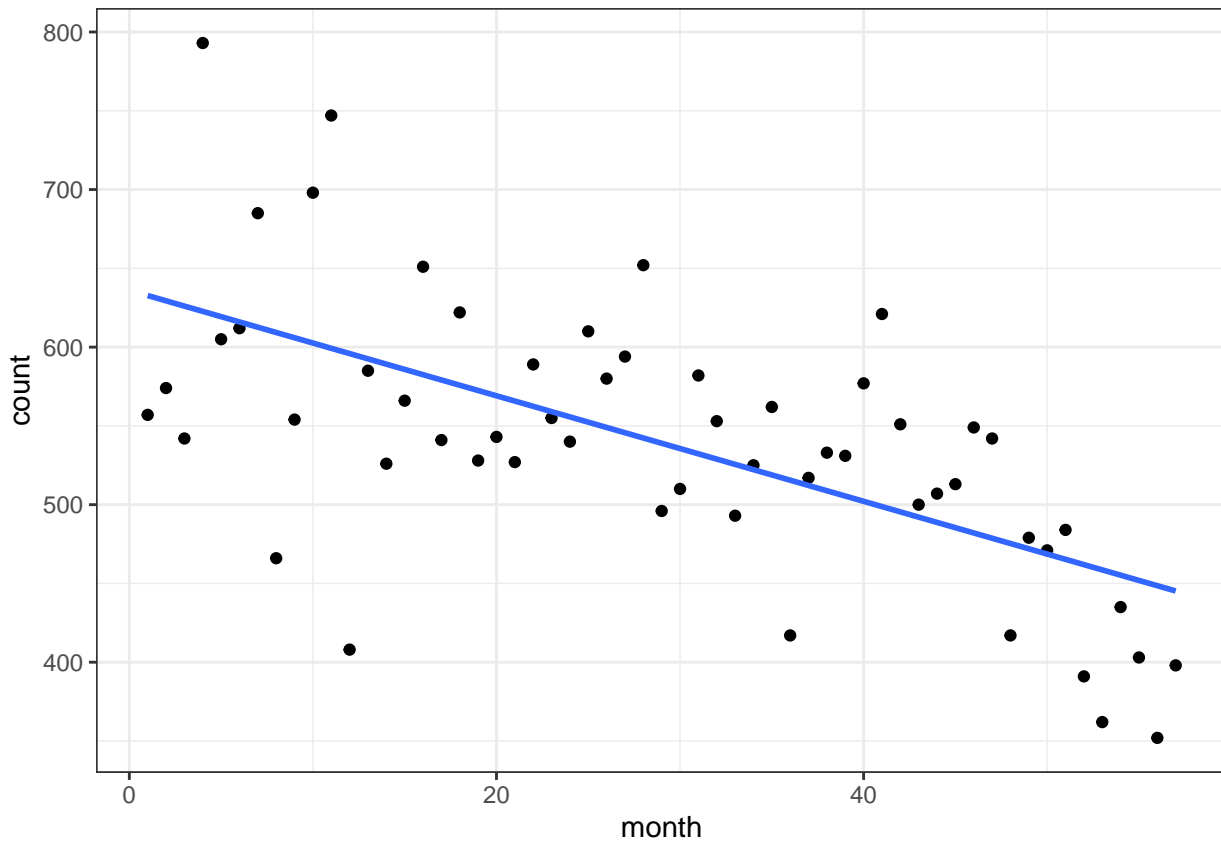
Plot the fitted values from the above linear regression.

```
ggplot(facility.new, aes(x = month, y = count)) +
  geom_point() +
  geom_line(aes(x = month, y = fit.lm$fitted.values)) +
  theme_bw()
```
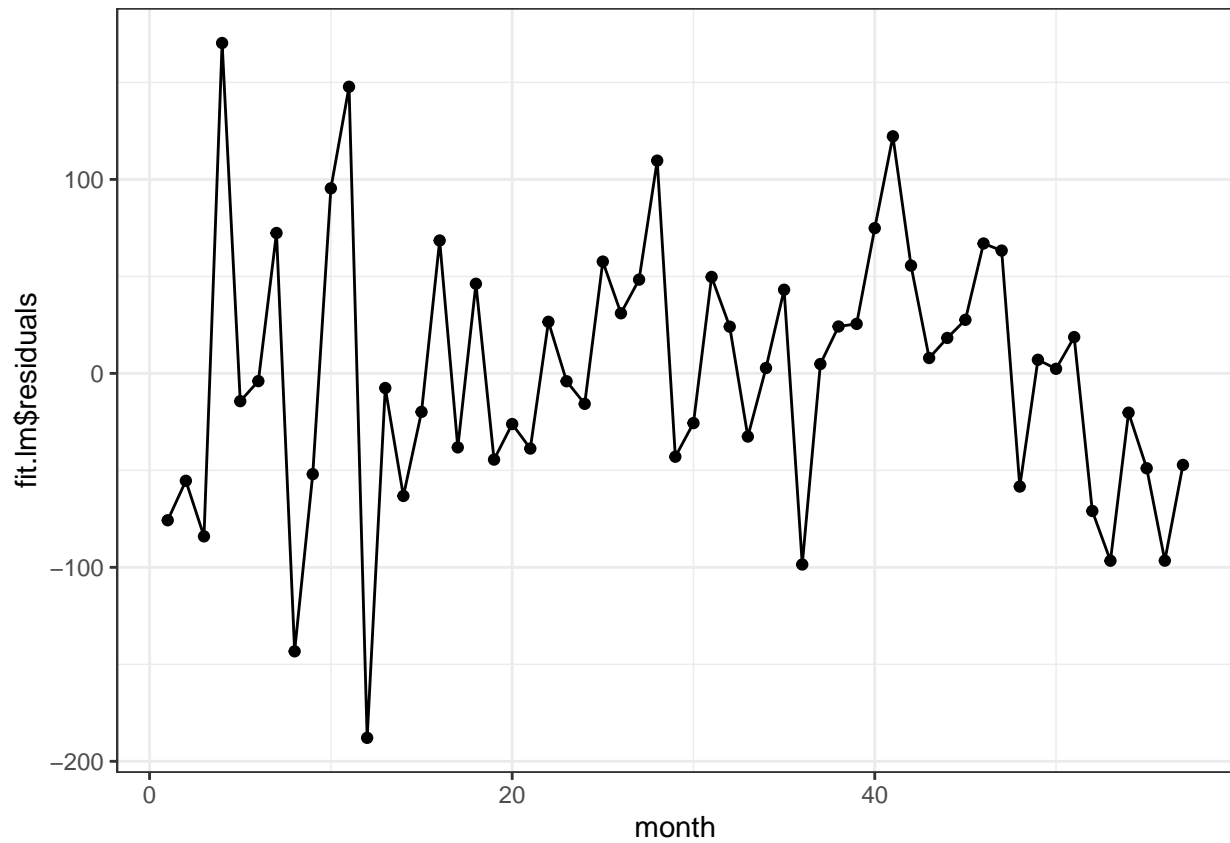

Optional: You can also fit a linear regression and plot at the same time!

```
ggplot(facility.new, aes(x = month, y = count)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  theme_bw()
```
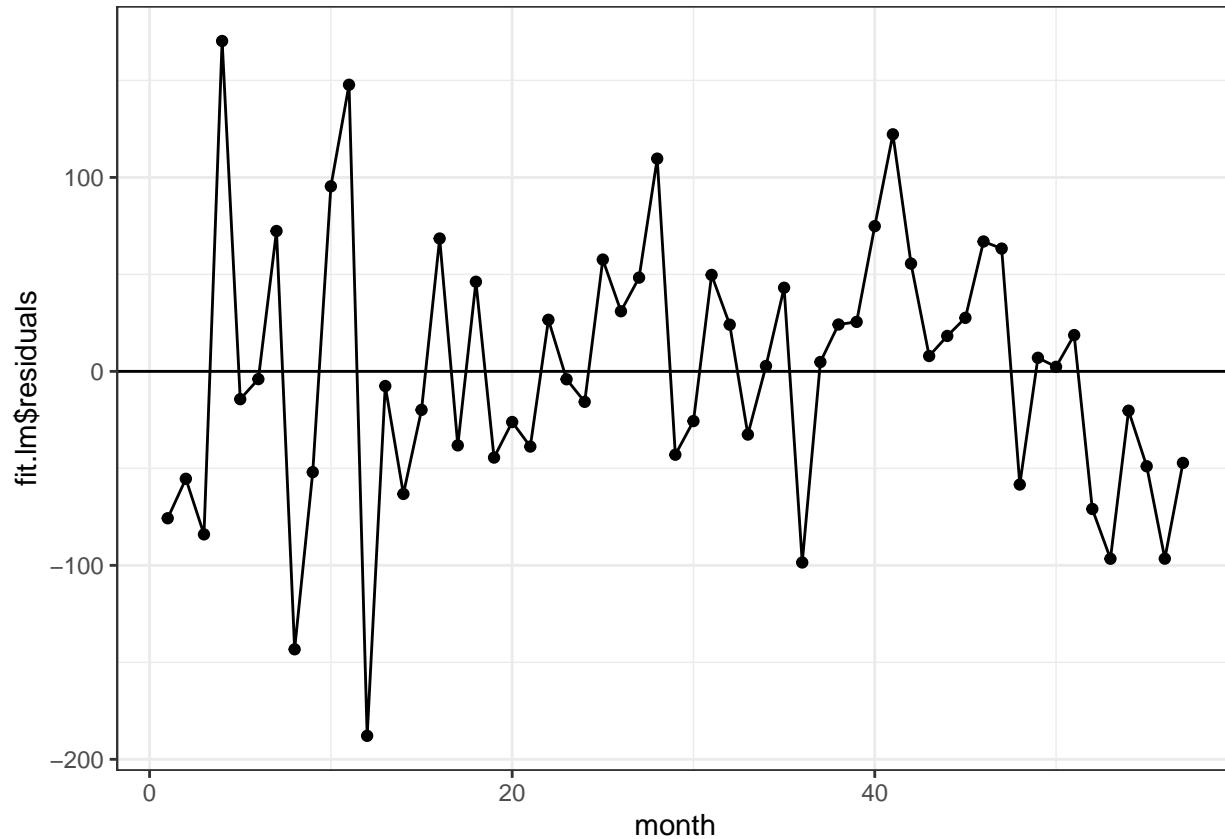
Plot the residuals from the above linear regression.

```
ggplot(facility.new, aes(x = month, y = fit.lm$residuals)) +
  geom_point() +
  geom_line() +
  theme_bw()
```

Add a horizontal line at zero.

```r
ggplot(facility.new, aes(x = month, y = fit.lm$residuals)) +
  geom_point() +
  geom_line() +
  geom_hline(yintercept=0) +
  theme_bw()
```

ACTIVITY

1. Fit a linear regression with a cubic term and plot the result. **Hint:** you will need three terms $month$, $month^2$, and $month^3$ in the model. You can do this by creating new terms in the dataset with the *mutate()* functon OR directly in the *lm()* function.

```
fit.lm3 <- lm(count ~ month + I(month^2) + I(month^3), data=facility.new)

summary(fit.lm3)

##
## Call:
## lm(formula = count ~ month + I(month^2) + I(month^3), data = facility.new)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -173.54  -41.91   -0.39   31.21  184.77
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 630.908471  37.405645  16.867   <2e-16 ***
## month        -6.631081   5.536793  -1.198    0.236
## I(month^2)    0.255800   0.220786   1.159    0.252
## I(month^3)   -0.003839   0.002504  -1.533    0.131
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 66.01 on 53 degrees of freedom
```
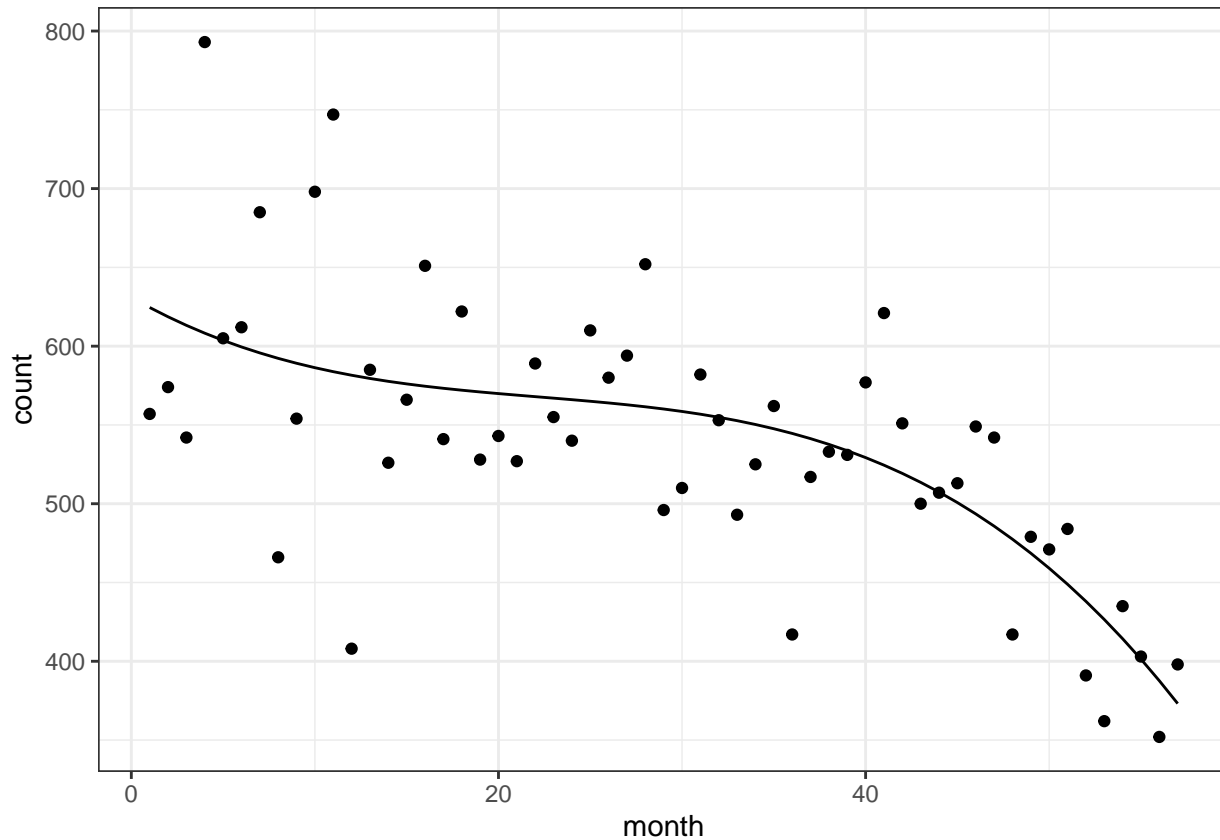
```
## Multiple R-squared:  0.4686, Adjusted R-squared:  0.4385
## F-statistic: 15.58 on 3 and 53 DF,  p-value: 2.18e-07
```

2. Plot the points with the fitted quadratic model.

```
ggplot(facility.new, aes(x = month, y = count)) +
  geom_point() +
  geom_line(aes(x = month, y = fit.lm3$fitted.values)) +
  theme_bw()
```


3. Plot the residuals with a horizontal line at zero. Is there evidence of residual correlation?

```
ggplot(facility.new, aes(x = month, y = fit.lm3$residuals)) +
  geom_point() +
  geom_line() +
  geom_hline(yintercept=0) +
  theme_bw()
```