Lab Session 2 Part 1: Plotting and Linear Regression

Nicholas Link

2024-12-07

INSTALL AND LOAD PACKAGES

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(lubridate)
```

Find the current directory and then change to correct directory.

```
getwd()
```

LOAD AND VIEW DATA

Load the outpatient visits ".rds" file from lab 1.

```
outpatient <- readRDS('materials/session1/session1_data/example_outpatient.rds')
```

View the first six observations in the outaptient datasets.

```
head(outpatient)
```

```
## # A tibble: 6 x 2
##   date       outpatient_visits
##   <date>                 <dbl>
## 1 2016-01-01              4983
## 2 2016-02-01              5331
## 3 2016-03-01              6267
## 4 2016-04-01              6063
## 5 2016-05-01              5775
## 6 2016-06-01              4397
```

SUMMARIZING THE DATA

How many months are in the dataset?

```
nrow(outpatient)
```

```
## [1] 60
```

What is the date range in the dataset?

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

```
## # A tibble: 1 x 2
##   `min(date)` `max(date)`
##   <date>      <date>
## 1 2016-01-01  2020-12-01
```

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

```
outpatient %>%
  summarize(mean(outpatient_visits))
```

```
## # A tibble: 1 x 1
##   `mean(outpatient_visits)`
##                       <dbl>
## 1                     4681.
```

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

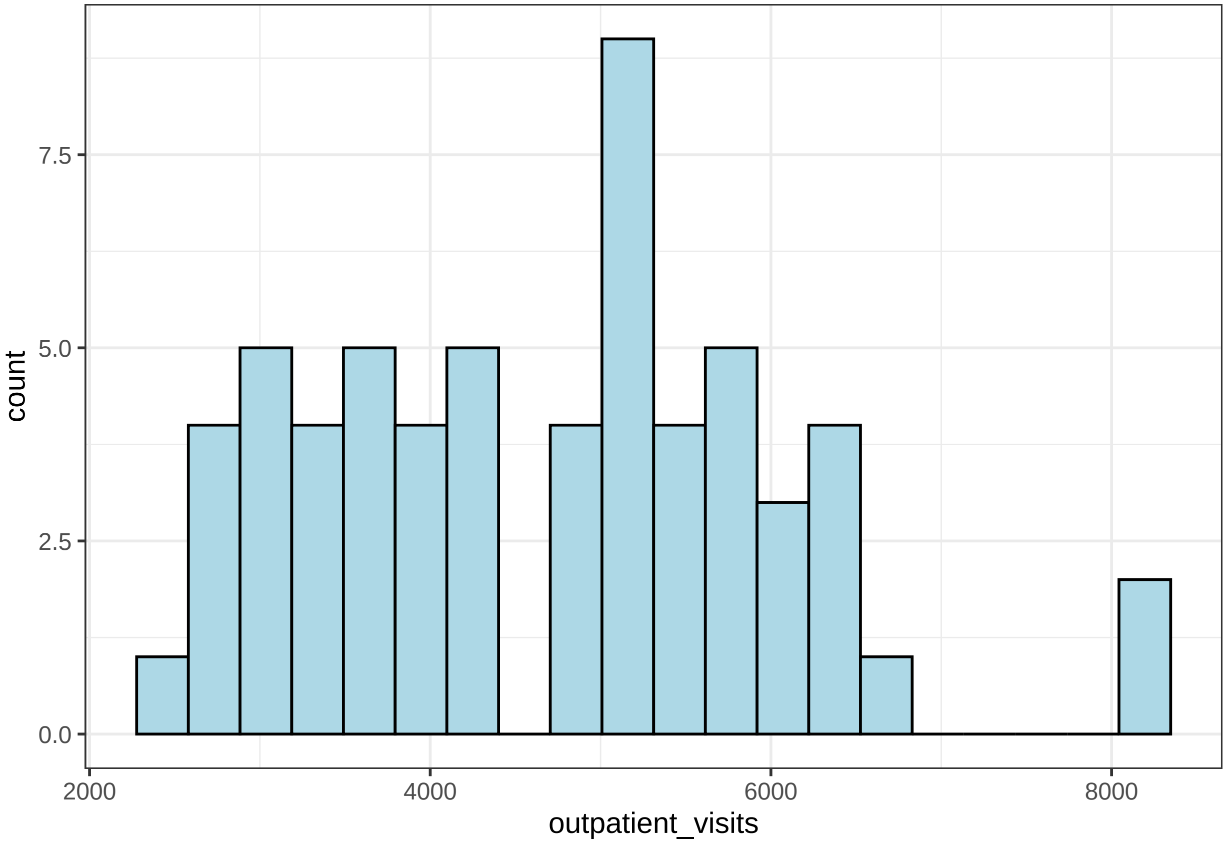
```
outpatient %>%
  summarize(min(outpatient_visits),
            max(outpatient_visits))
```

```
## # A tibble: 1 x 2
##   `min(outpatient_visits)` `max(outpatient_visits)`
##                      <dbl>                    <dbl>
## 1                     2559                     8326
```

VISUALIZING THE DATA

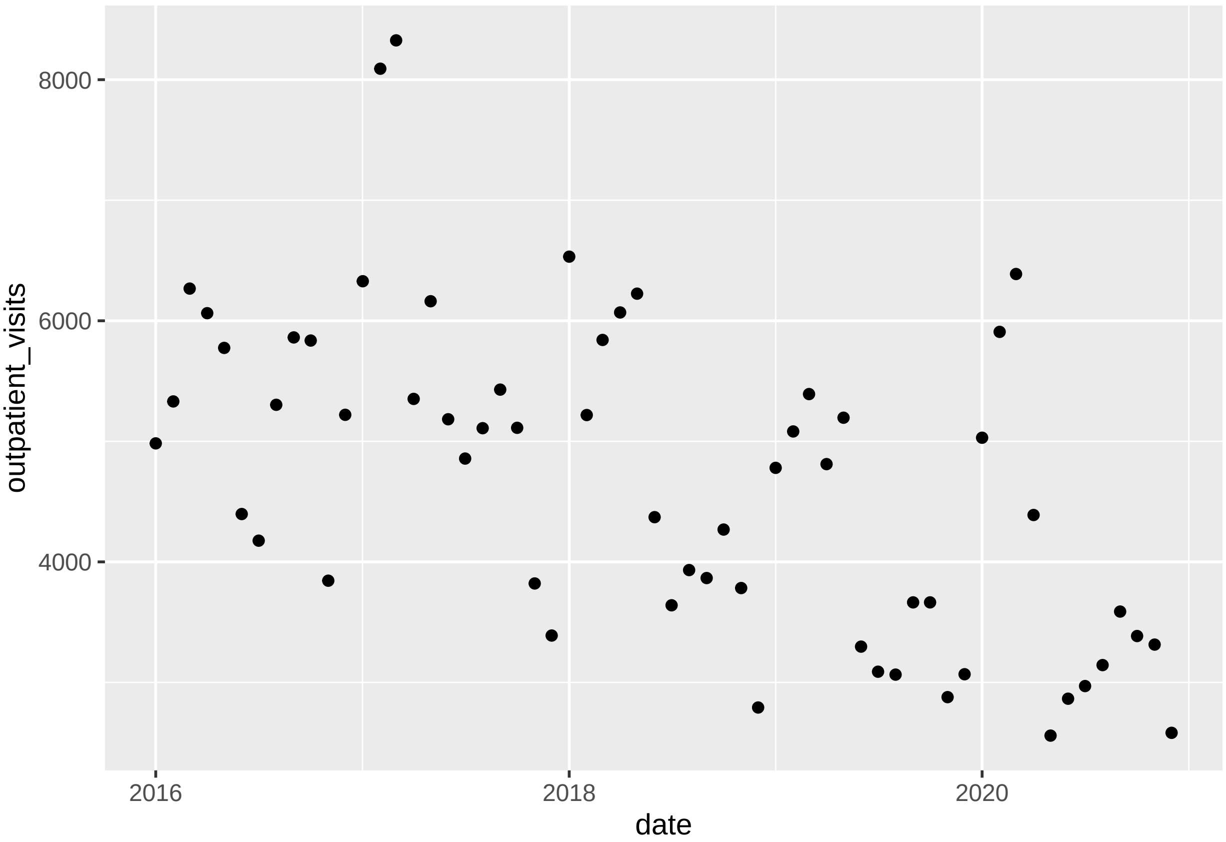
Create a histogram of the monthly outpatient visits outpatient_visits

```
ggplot(outpatient,aes(outpatient_visits)) +
  geom_histogram(color="black",fill="lightblue",bins = 20) +
  theme_bw()
```



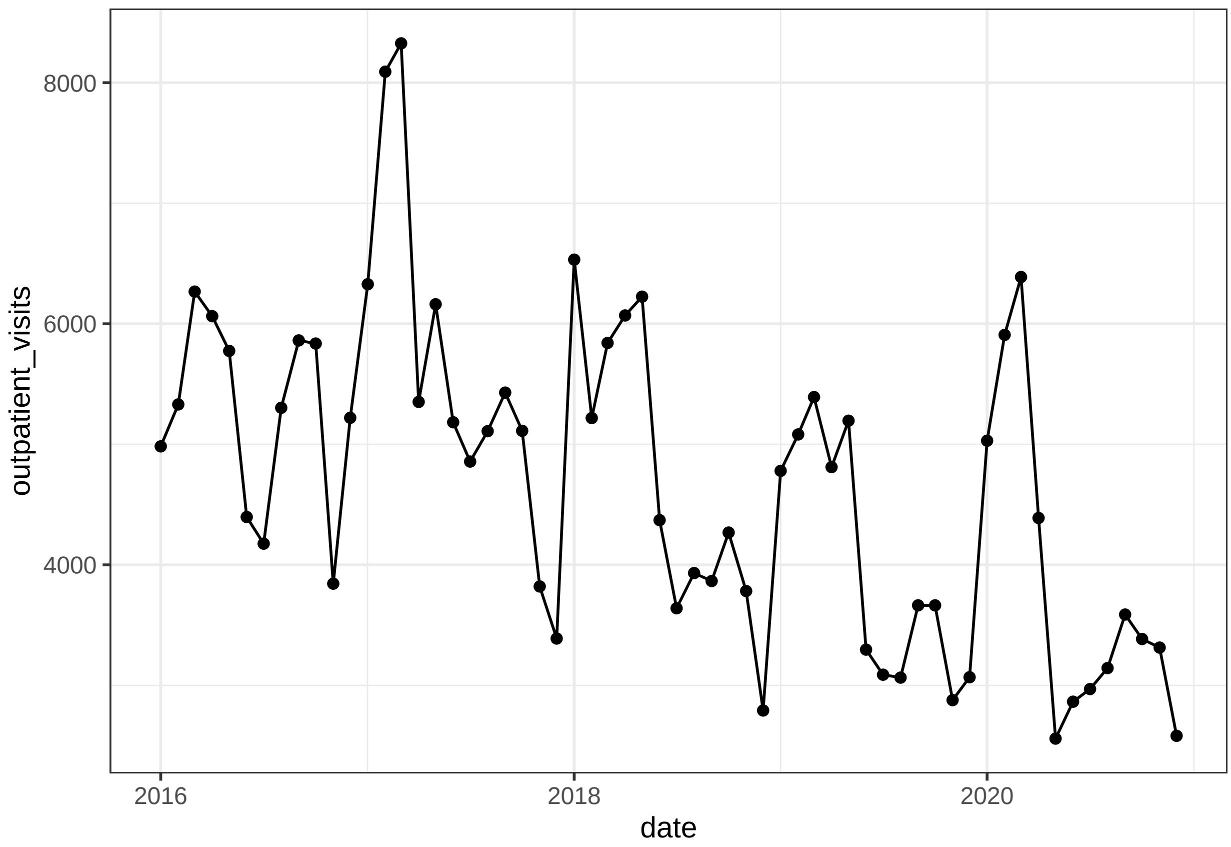
Create a scatter plot of the outpatient_visitss over time.

```
ggplot(outpatient,aes(x=date,y=outpatient_visits)) +
  geom_point()
```



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

```
ggplot(outpatient,aes(x=date,y=outpatient_visits)) +
  geom_point() +
  geom_line() +
  theme_bw()
```



ACTIVITY: Add aesthetics to the above plot by changing the various inputs.

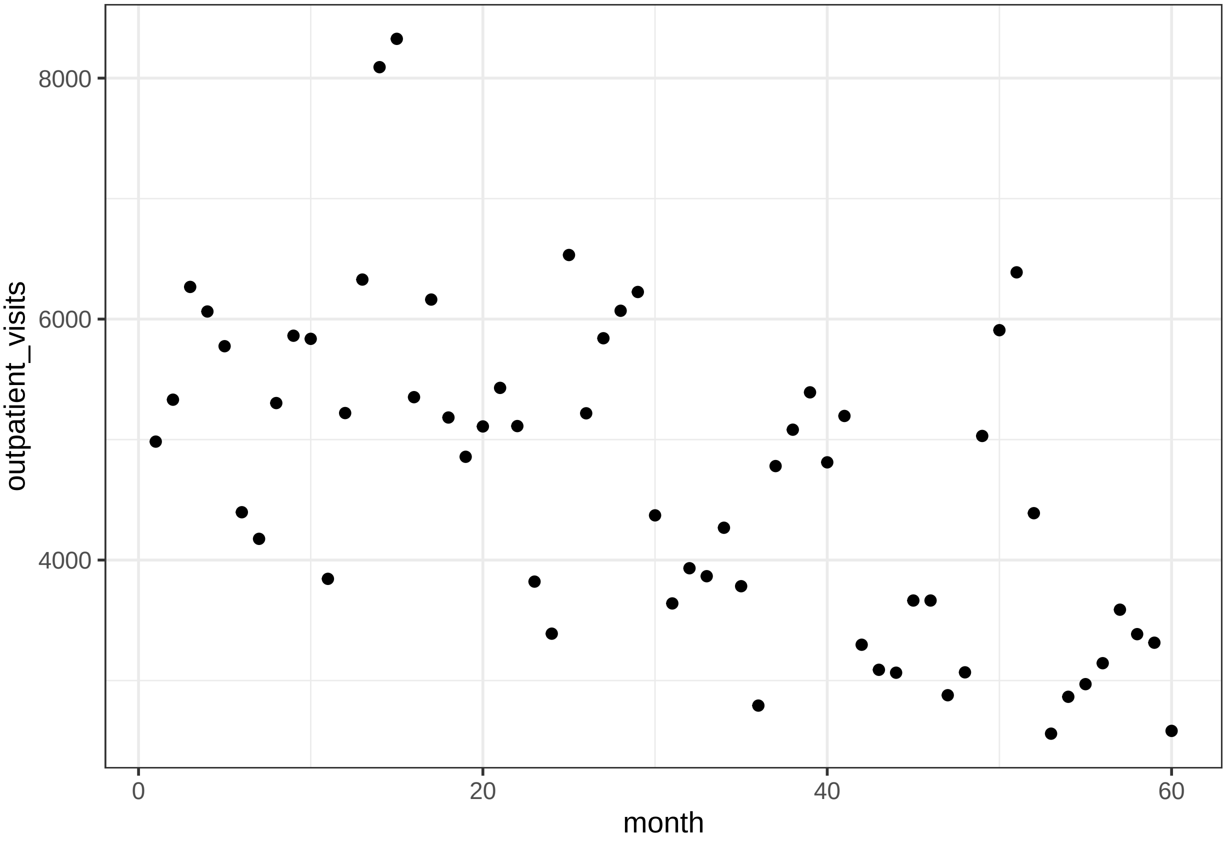
Linear regression and plotting output.

Create a new time variable for each month.

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

Plot the values of the data.

```
ggplot(outpatient.new,aes(x=month,y=outpatient_visits)) +
  geom_point() +
  theme_bw()
```

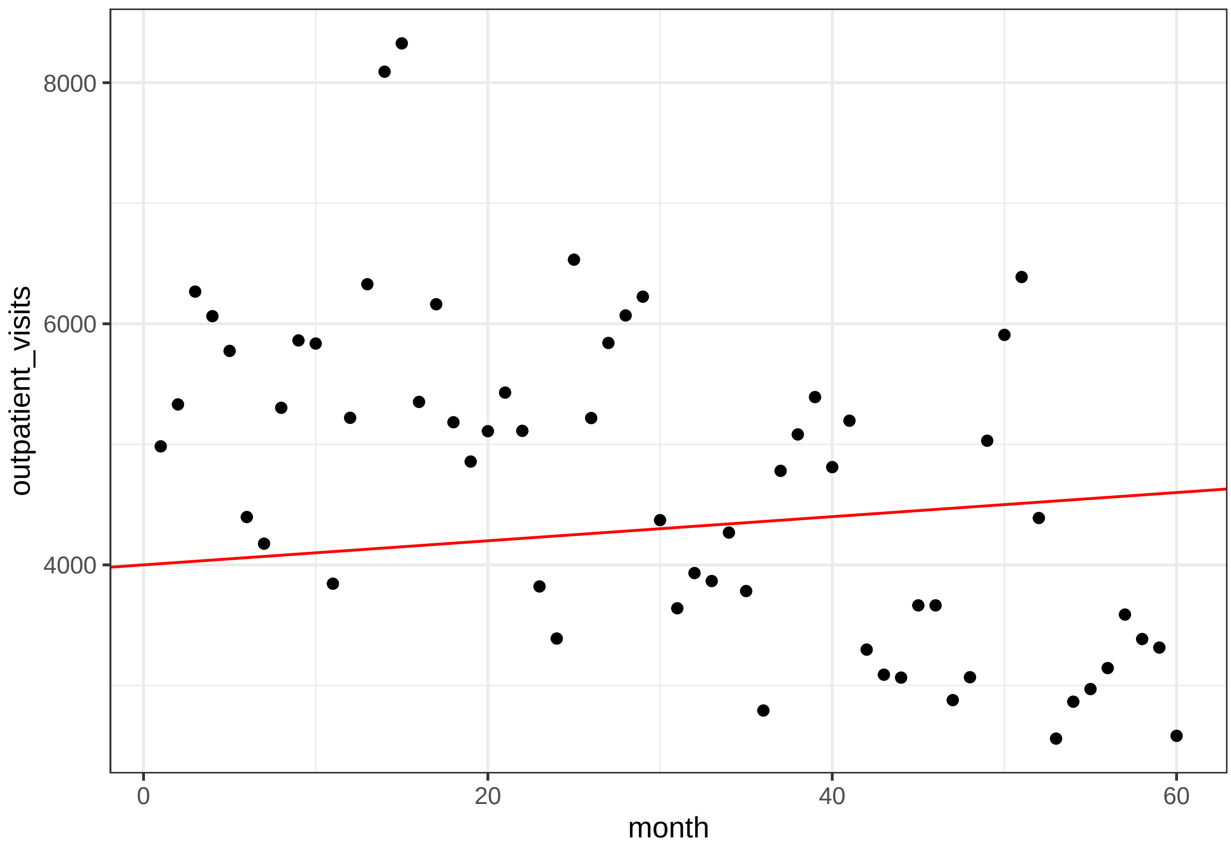


ACTIVITY:

(1) For the linear regression equation outpatient_visits = B0 + B1*month + e, write down your guess for B0 and B1 that best fit the data (no code)

(2) Using the geom_abline function, change the intercept and slope values to guess the best fitting line. How close are the intercept and slope values to what you guessed in (1)?

```r
ggplot(outpatient.new, aes(x=month,y=outpatient_visits)) +
  geom_point() +
  geom_abline(intercept = 4000, slope = 10, col = 'red') +
  theme_bw()
```



Fit a linear regression with an intercept and term for time. R automatically includes an intercept in the model.

```
fit.lm <- lm(outpatient_visits ~ month, data=outpatient.new)
```

Look at the coefficient results of the model. How close were your line coefficients to this one?

```
fit.lm
```

```
##
## Call:
## lm(formula = outpatient_visits ~ month, data = outpatient.new)
##
## Coefficients:
## (Intercept)        month
##     6042.08       -44.63
```

Look at the in-depth results of the model.

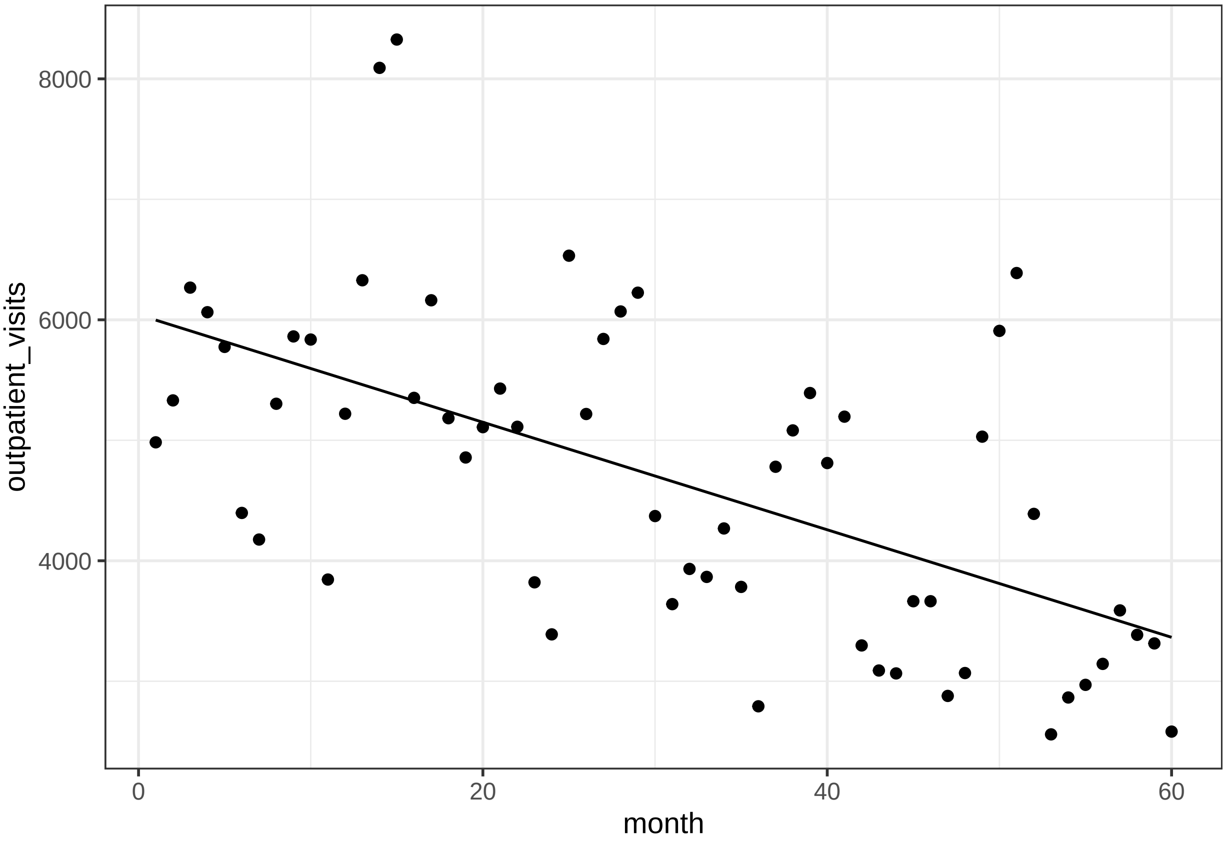
```
summary(fit.lm)
```

```
##
## Call:
## lm(formula = outpatient_visits ~ month, data = outpatient.new)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1707.20  -771.14   -82.02   582.30  2953.30
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6042.080    282.607  21.380  < 2e-16 ***
## month        -44.625      8.058  -5.538 7.75e-07 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 1081 on 58 degrees of freedom
## Multiple R-squared:  0.3459, Adjusted R-squared:  0.3346
## F-statistic: 30.67 on 1 and 58 DF,  p-value: 7.755e-07
```

Plot the fitted values from the above linear regression.

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



Using the regression formula outpatient = B0 + B1*month, predict what the outpatient value will be in month 61 (2021-01-01).

```r
predicted.point <- 6042.08-44.63*61
predicted.point
```

```
## [1] 3319.65
```

Now, run the model using “date” as the x variable instead of month.

```r
fit.lm.2 <- lm(outpatient_visits ~ date, data=outpatient.new)
```

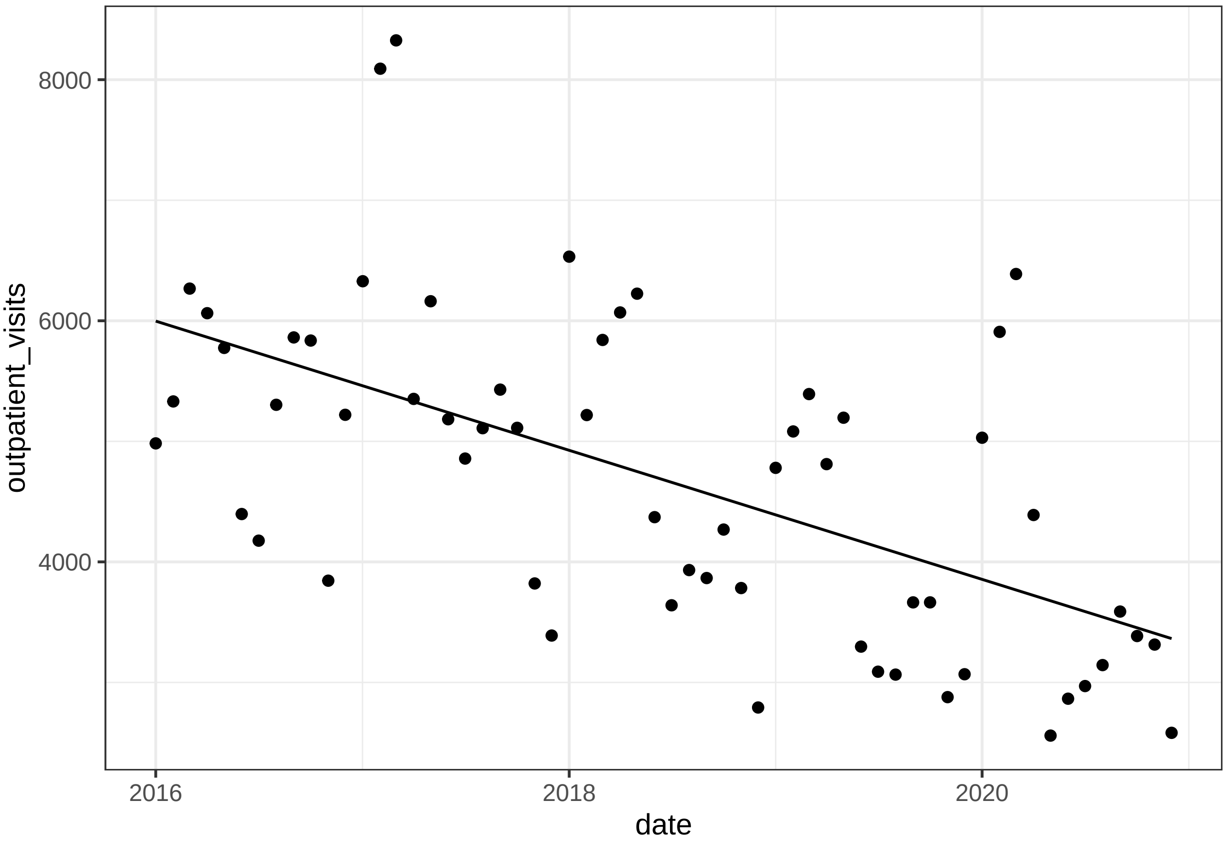
Print the results. Are the coefficients the same?

```r
summary(fit.lm.2)
```

```
## 
## Call:
## lm(formula = outpatient_visits ~ date, data = outpatient.new)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1706.01  -770.73   -80.94   581.49  2951.95 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 30632.9017  4687.8198   6.535 1.77e-08 ***
## date           -1.4663     0.2648  -5.538 7.75e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1081 on 58 degrees of freedom
## Multiple R-squared:  0.3459, Adjusted R-squared:  0.3346 
## F-statistic: 30.67 on 1 and 58 DF,  p-value: 7.75e-07
```

Plot the results. Does this look the same?

```r
ggplot(outpatient.new, aes(x = date, y = outpatient_visits)) +
  geom_point() +
  geom_line(aes(x = date, y = fit.lm.2$fitted.values)) +
  theme_bw()
```



Can you explain why the coefficients and the plots from these two models are the same/different?

The linear regression plots are the exact same. The best fitting line is the same in both because the data points are the same (visually, at least). However, the equation changes because the units of time (the x-axis) are different. B1 tells us the change in the outcome with one unit change of the x value, so when the units of the x value change, so does the value of B1. B0 shifted because in the second analysis, the time values did not start at 0, and B0 needed to account for this.