STAT 456 Homework 3

Hailey Lee

02/29/2024

Instructions:

1. Please use R to finish all the questions below. Although in some simple cases, you may obtain the solution directly without using R, you still need to provide the corresponding R code.
2. You are liable for missing points if you don't include output;
3. Whenever possible, please run saved variables, so our TA knows if your code goes the right way and assigns partial credits even if your final answer is wrong.
4. Please submit your solutions in .rmd file and .pdf file compiled via the R markdown through Blackboard.

1. **State-level COVID-19 Data.** Recall the The dataset `state.long` from the package `IDDA` in Homework 2. `state.long` contains the following variables:

```
# install the IDDA package from github
library(devtools)
```

```
## Loading required package: usethis
```

```
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   3.4.4     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
```

```
library(dplyr)
devtools::install_github('FIRST-Data-Lab/IDDA')
```

```
## Skipping install of 'IDDA' from a github remote, the SHA1 (7439336f) has not changed
##   Use `force = TRUE` to force installation
```

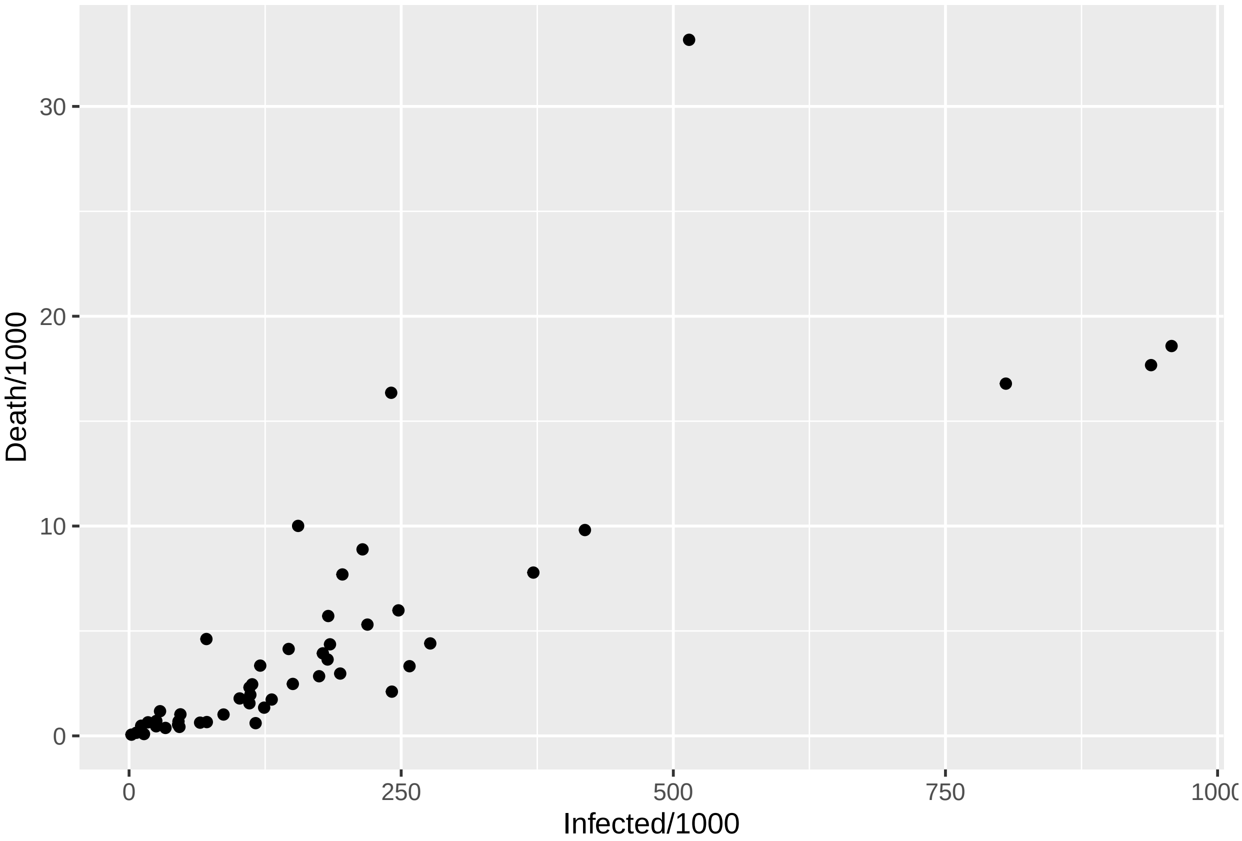
```
# load objects in I.county into my workspace
library(IDDA)
data(state.long)
```

- `State`: name of state matched with ID.
- `Region`: region of a state.
- `Division`: division of a state.
- `pop`: population of a state.
- `DATE`: date that the data is reported.
- `Infected`: the cumulative infected count of a state.
- `Death`: the cumulative death count of a state

Make a scatter plot using `IDDA::state.long` on 2020-11-01 according to the following.

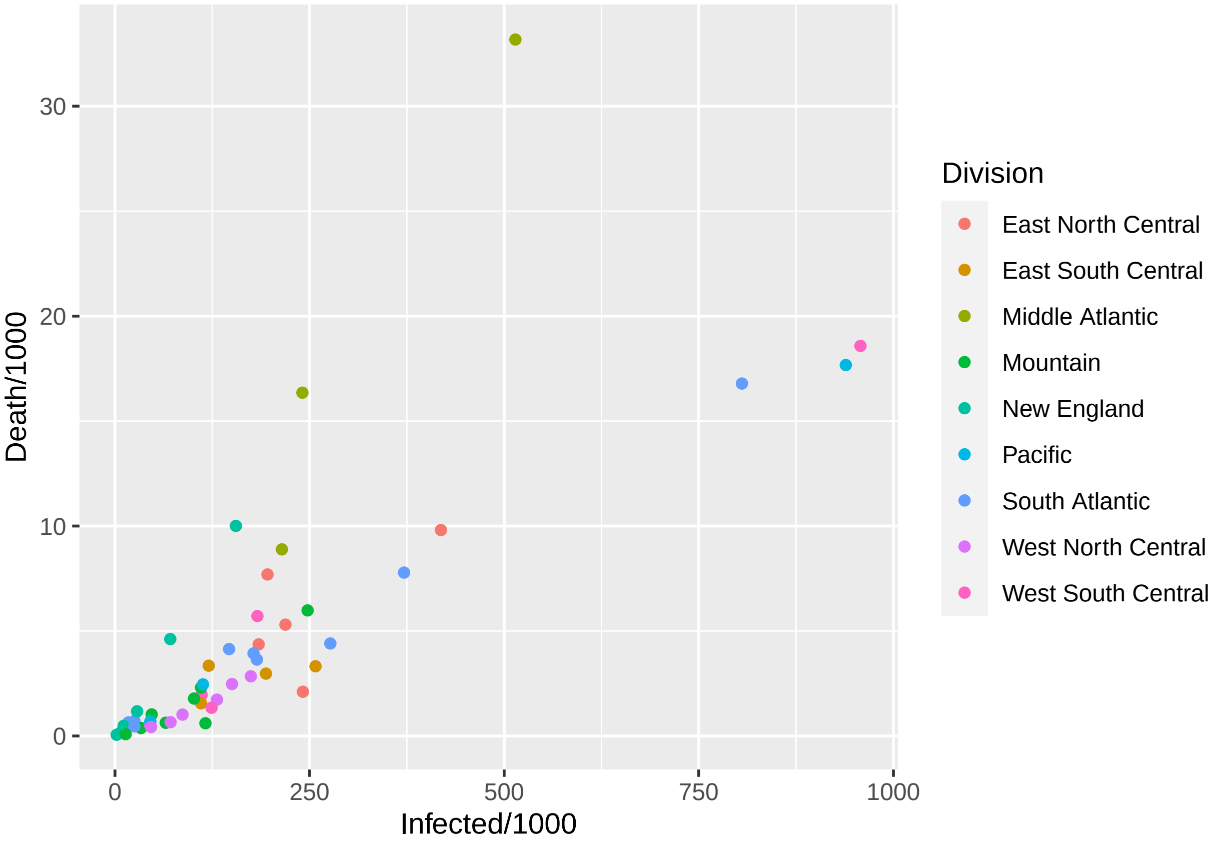
(a). Create a scatter plot. Treat `Infected/1000` as x-axis, and `Death/1000` as y-axis.

```
IDDA::state.long %>%
  filter(DATE == "2020-11-01") %>%
  ggplot(aes(x = Infected/1000, y = Death/1000)) +
  geom_point() +
  labs(x = "Infected/1000",
       y = "Death/1000")
```



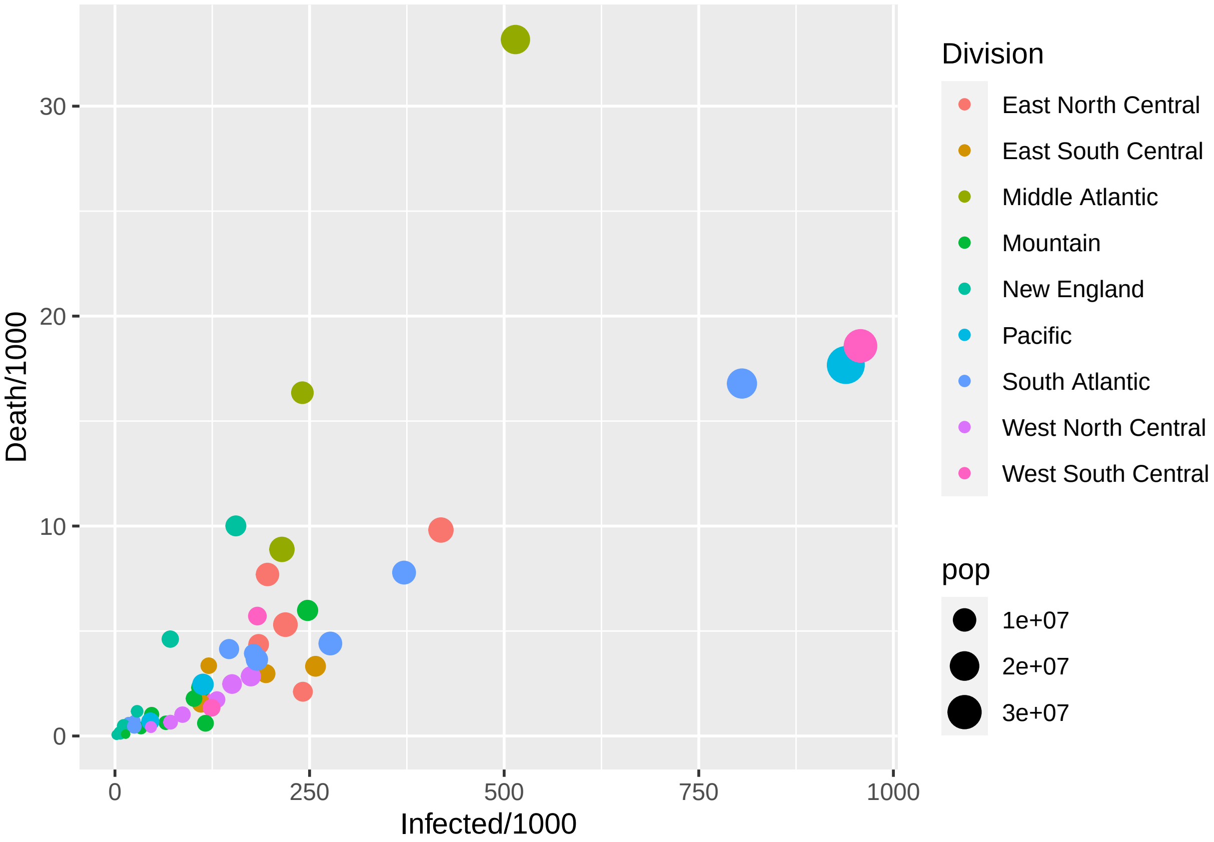
(b). Color the points according to `Division`. Hint: use `aes(color = )`.

```
IDDA::state.long %>%
  filter(DATE == "2020-11-01") %>%
  ggplot(aes(x = Infected/1000, y = Death/1000, color = Division)) +
  geom_point() +
  labs(x = "Infected/1000",
       y = "Death/1000")
```



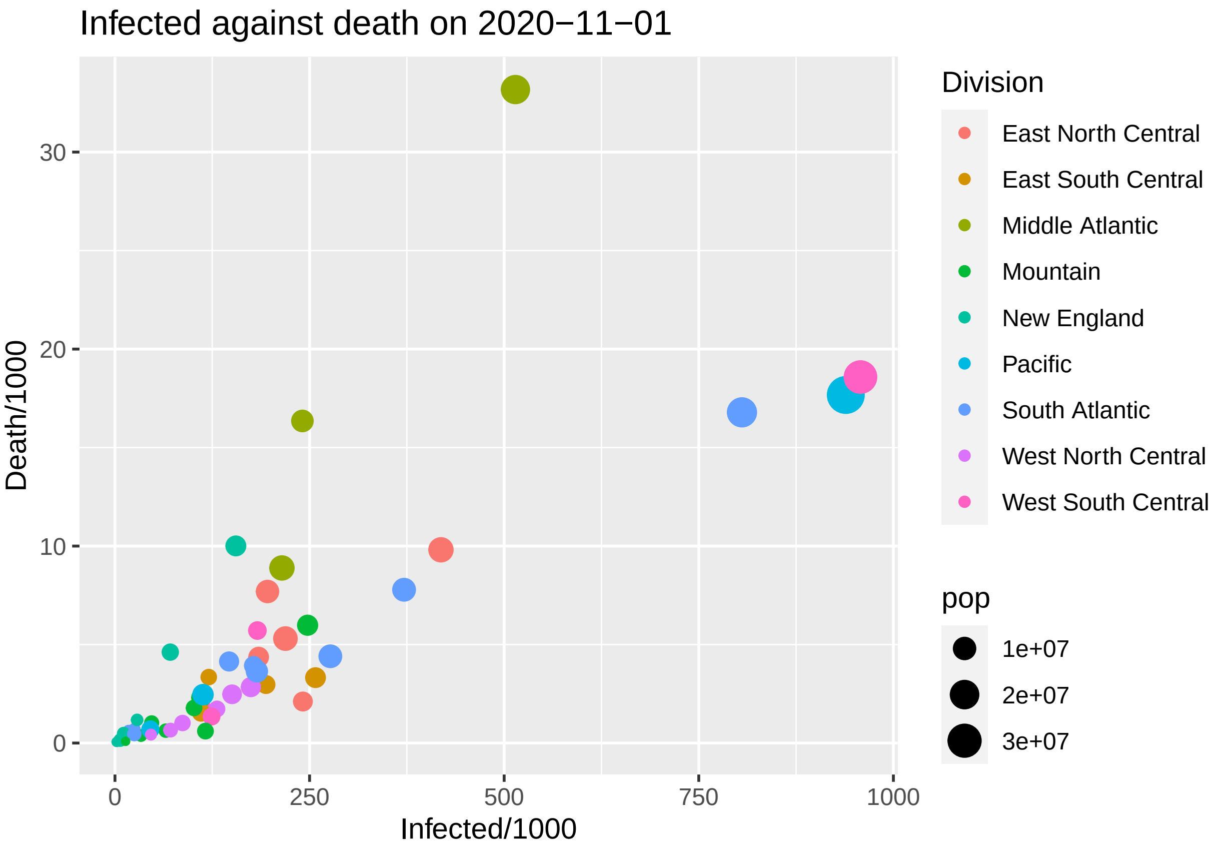
(c). Change the size of the points to be proportional to population. Hint: use `aes(size = )`.

```
IDDA::state.long %>%
  filter(DATE == "2020-11-01") %>%
  ggplot(aes(x = Infected/1000, y = Death/1000, color = Division, size = pop)) +
  geom_point() +
  labs(x = "Infected/1000",
       y = "Death/1000")
```



(d). Change the title of the figure as ‘Infected against death on 2020-11-01’.

```
IDDA::state.long %>%
  filter(DATE == "2020-11-01") %>%
  ggplot(aes(x = Infected/1000, y = Death/1000, color = Division, size = pop)) +
  geom_point() +
  labs(title = "Infected against death on 2020-11-01",
       x = "Infected/1000",
       y = "Death/1000")
```



2. **Time at the table.** Does how long young children remain at the lunch table help to predict how much they eat. Here are the data on 20 toddlers observed over several months at a nursery school. `Time` is the average number of minutes a child spent at the table when lunch was served. `Calories` is the average number of calories the child consumed during lunch, calculated from careful observation of what the child ate each day.

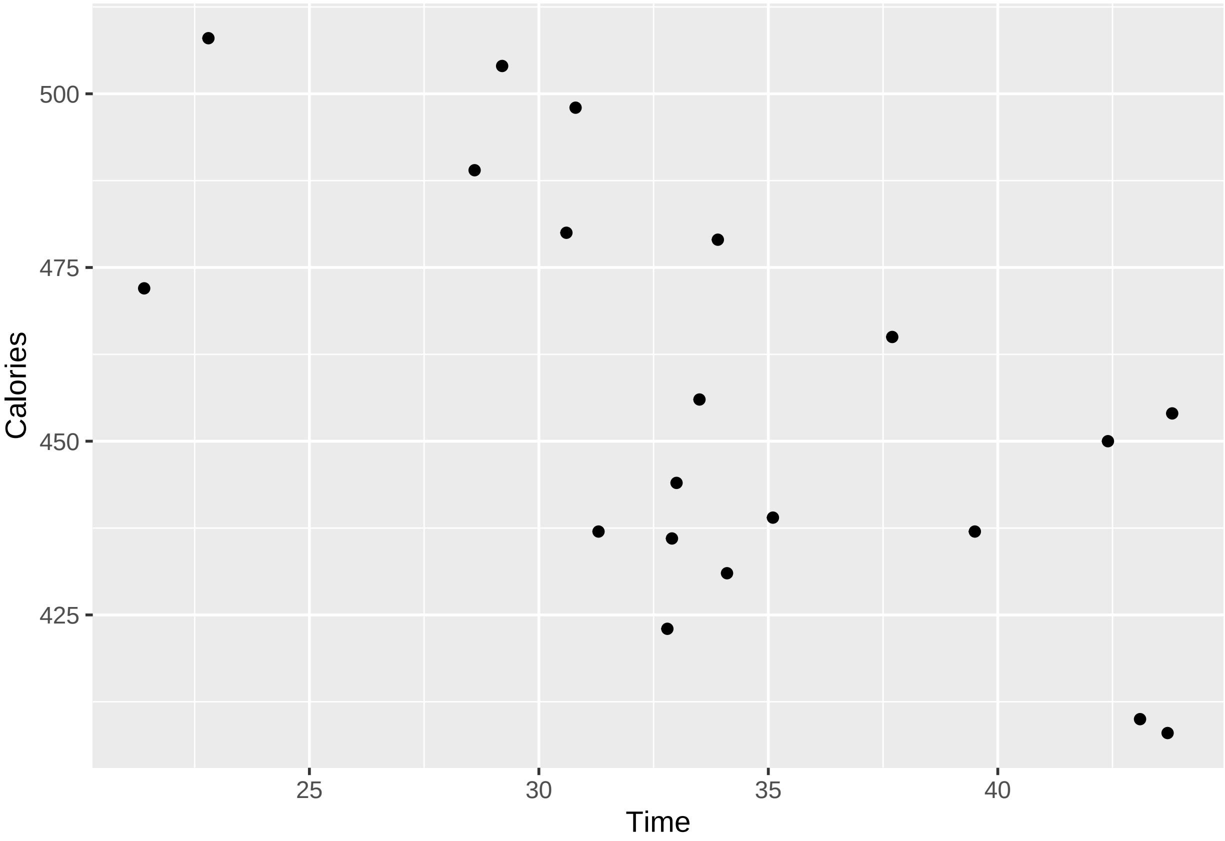
Child	Time	Calories
1	21.4	472
2	30.8	498
3	37.7	465
4	33.5	456
5	32.8	423
6	39.5	437
7	22.8	508
8	34.1	431
9	33.9	479
10	43.8	454
11	42.4	450
12	43.1	410
13	29.2	504
14	31.3	437

Child	Time	Calories
15	28.6	489
16	32.9	436
17	30.6	480
18	35.1	439
19	33.0	444
20	43.7	408

(a). Draw a scatter-plot.

```
lunch_table <- data.frame(
  child = 1:20,
  Time = c(21.4, 30.8, 37.7, 33.5, 32.8, 39.5, 22.8, 34.1, 33.9, 43.8,
           42.4, 43.1, 29.2, 31.3, 28.6, 32.9, 30.6, 35.1, 33.0, 43.7),
  Calories = c(472, 498, 465, 456, 423, 437, 508, 431, 479, 454,
               450, 410, 504, 437, 489, 436, 480, 439, 444, 408)
)

lunch_table %>%
  ggplot(aes(x = Time, y = Calories)) +
  geom_point()
```



(b). Report the Pearson correlation between `Calories` and `Time`.

```r
cor.test(lunch_table$Time, lunch_table$Calories, method="pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  lunch_table$Time and lunch_table$Calories
## t = -3.6208, df = 18, p-value = 0.001954
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.8480643 -0.2899357
## sample estimates:
##        cor
## -0.6491667
```

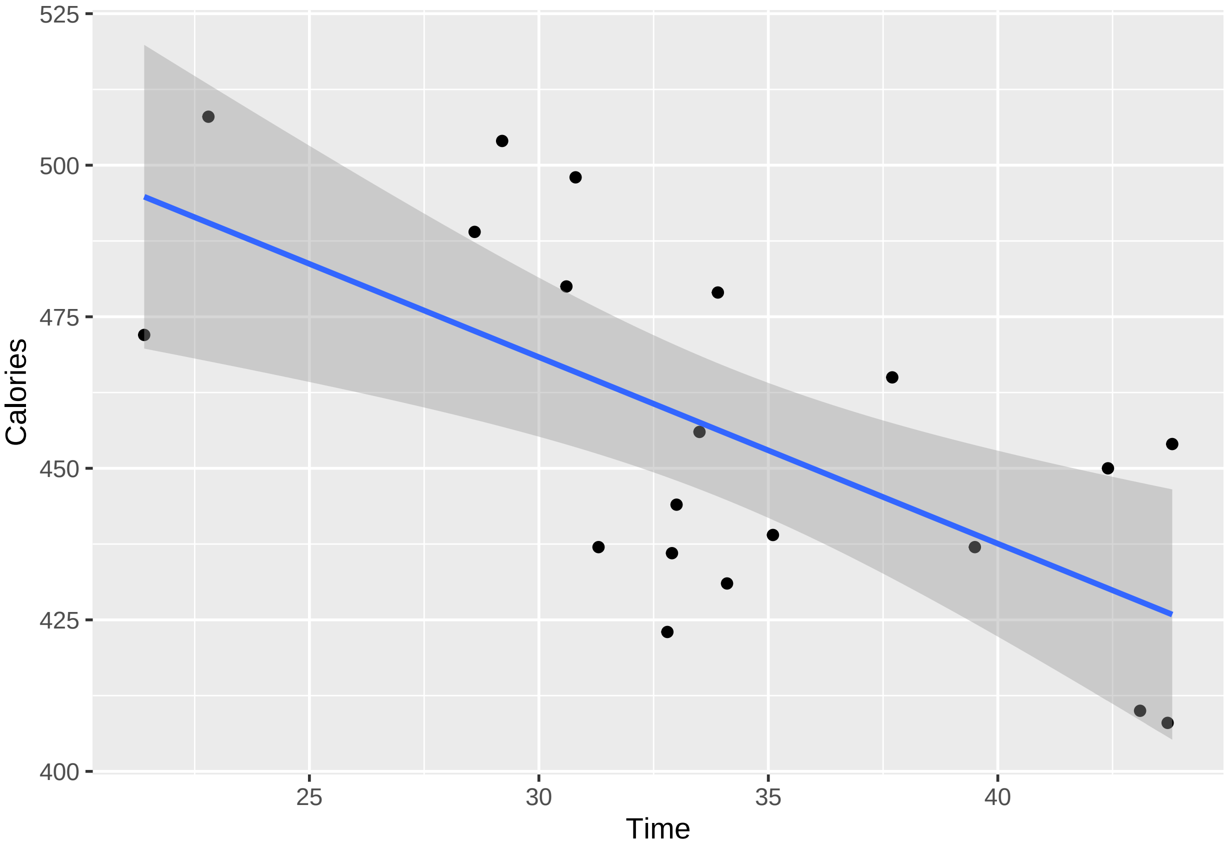
(c). Fit a linear regression model between `Calories` and `Time`. Note that `Calories` is the dependent variable. Add the fitted line to the scatter-plot. Write down the estimated value of slope and intercept. Based on your work, describe the direction, form and strength of the relationship.

```r
lm(Calories ~ Time, data = lunch_table)
```

```
##
## Call:
## lm(formula = Calories ~ Time, data = lunch_table)
##
## Coefficients:
## (Intercept)         Time
##     560.651       -3.077
```

```r
lunch_table %>%
  ggplot(aes(x = Time, y = Calories)) +
  geom_point() +
  geom_smooth(method = "lm")
```

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



Estimated value of the intercept is 560.651, and estimated value of the slope is -3.077. Based on the estimated value of the intercept and slope, the direction is negative, the form is linear, and the points appear to follow a single stream.

3. **Ball Thrower.** A child throws a tennis ball straight up in the air. The table below shows the height of the ball (measured in feet from the ground) at $n = 7$ times, measured in seconds, where Time= 0 corresponds to the time at which the ball was released.

Time	Height
0.0	8
0.5	48
1.0	72
1.5	80
2.0	72
2.5	48
3.0	8

(a). Report the Pearson correlation between X and Y, and test if it is significant.

```r
tennis <- data.frame(
  Time = c(0.0, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0),
  Height = c(8, 48, 72, 80, 72, 48, 8)
)

cor.test(tennis$Time, tennis$Height, method = "pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  tennis$Time and tennis$Height
## t = 0, df = 5, p-value = 1
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7530581  0.7530581
## sample estimates:
## cor
##   0
```

Since the p-value is greater than the significance level, 0.005, we fail to reject the null hypothesis. There is not statistically significant evidence to suggest a correlation between time and height.

(b). If one fits the above data by the linear regression model:

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i,$$

where Y is Height and X is Time, obtain the least-squares estimates of $\widehat{\beta}_0$ and $\widehat{\beta}_1$.

```
coefficients(lm(Height ~ Time, data = tennis))
```

```
##  (Intercept)         Time
## 4.800000e+01 4.667014e-15
```

(c). What is the R^2 for the model in part (a). Discuss your results.

```
summary(lm(Height ~ Time, data = tennis))
```

```
##
## Call:
## lm(formula = Height ~ Time, data = tennis)
##
## Residuals:
##          1          2          3          4          5          6          7
## -4.000e+01  2.346e-15  2.400e+01  3.200e+01  2.400e+01 -4.760e-15 -4.000e+01
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.800e+01  2.234e+01   2.148   0.0844 .
## Time        4.667e-15  1.239e+01   0.000   1.0000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32.79 on 5 degrees of freedom
## Multiple R-squared:  6.574e-32,  Adjusted R-squared:   -0.2
## F-statistic: 3.287e-31 on 1 and 5 DF,  p-value: 1
```

The value of R-squared is 6.5743e-32. The points are almost not explained by the regression at all.