Assignment 2: Physical Properties of Lakes

Walker Grimshaw

OVERVIEW

This exercise accompanies the lessons in Hydrologic Data Analysis on the physical properties of lakes.

Directions

1. Change "Student Name" on line 3 (above) with your name.
2. Work through the steps, **creating code and output** that fulfill each instruction.
3. Be sure to **answer the questions** in this assignment document.
4. When you have completed the assignment, **Knit** the text and code into a single PDF file.
5. After Knitting, submit the completed exercise (PDF file) to the dropbox in Sakai. Add your last name into the file name (e.g., "Salk_A02_LakePhysical.Rmd") prior to submission.

The completed exercise is due on 11 September 2019 at 9:00 am.

Setup

1. Verify your working directory is set to the R project file,
2. Load the tidyverse, lubridate, and cowplot packages
3. Import the NTL-LTER physical lake dataset and set the date column to the date format
4. Set your ggplot theme (can be theme_classic or something else)

```
# keep warnings from appearing in knitted pdf
knitr::opts_chunk$set(warning = FALSE)

getwd()
```

```
## [1] "C:/Users/walke/OneDrive/Documents/Duke/Courses/Fall_2019/Hydrologic_Data_Analysis/Assignments"
```

```
## successfully verified working directory

## load packages
library(tidyverse)
library(lubridate)
library(cowplot)

## Import NTL-LTER dataset, but must use ../Data because of knitting directory
NTL.Physical <- read.csv("../Data/Raw/NTL-LTER_Lake_ChemistryPhysics_Raw.csv",
                         header = T)

## reformat date column as date
NTL.Physical$sampledate <- as.Date(NTL.Physical$sampledate, "%m/%d/%y")

## create ggplot theme and set as default
WalkersTheme <- theme_bw(base_size = 12) + theme(legend.position = "top")
theme_set(WalkersTheme)
```

Creating and analyzing lake temperature profiles

Single lake, multiple dates

5. Choose either Peter or Tuesday Lake. Create a new data frame that wrangles the full data frame so that it only includes that lake during two different years (one year from the early part of the dataset and one year from the late part of the dataset).

```
## Tuesday lake
Tuesday.data <- NTL.Physical %>%
  filter(lakename == "Tuesday Lake")

## use table() to summarize the number of data points from each year
table(Tuesday.data$year4)
```

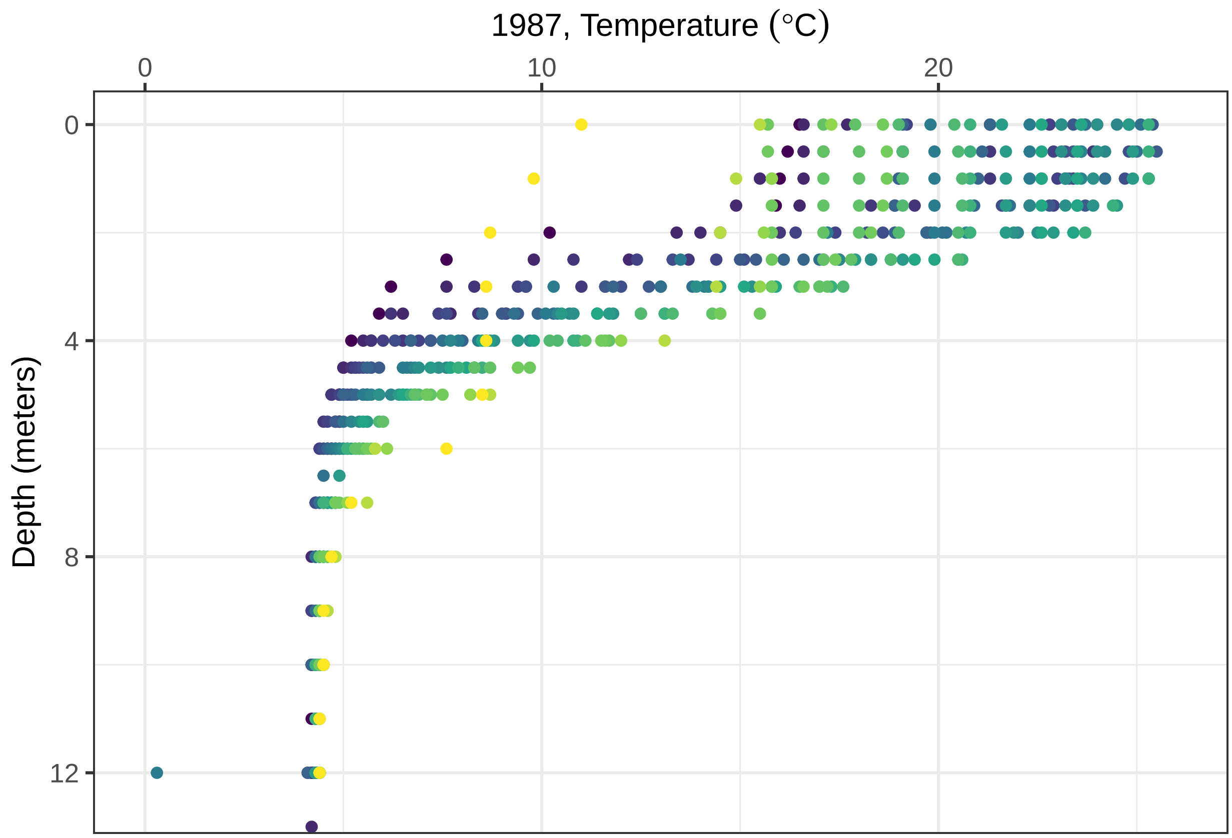
```
##
## 1984 1985 1986 1987 1988 1989 1990 1991 1993 1994 1995 1996 1997 1998 1999
##  254  310  255  663  355  304  356  329  172  168  154   88  286   88  330
## 2000 2002 2007 2012 2013 2014 2015 2016
##   88  385   18  257  295  305  360  287
```

```
## use 1987 as early year and 2015 as late year for large number of data points
Tuesday.early.late <- filter(Tuesday.data, year4 %in% c(1987, 2015))
```

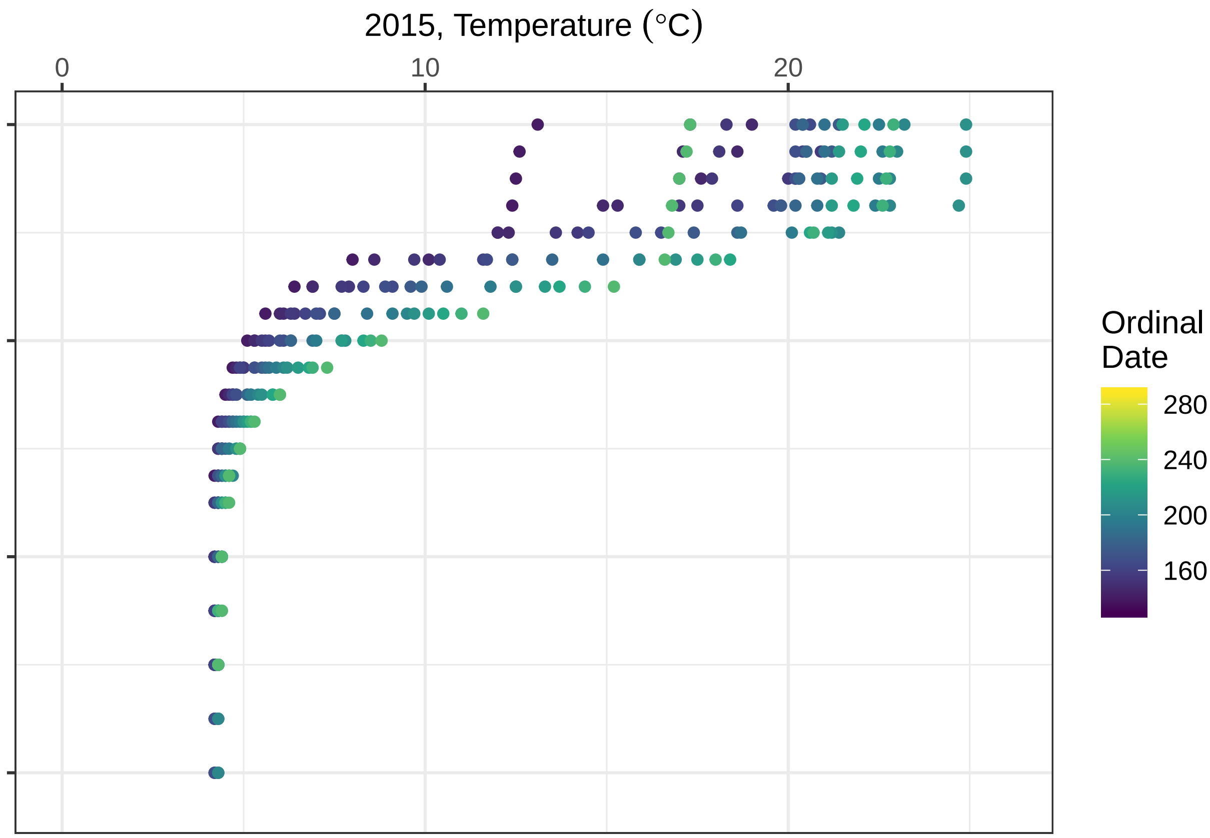
6. Create three graphs: (1) temperature profiles for the early year, (2) temperature profiles for the late year, and (3) a `plot_grid` of the two graphs together. Choose `geom_point` and color your points by date.

Remember to edit your graphs so they follow good data visualization practices.

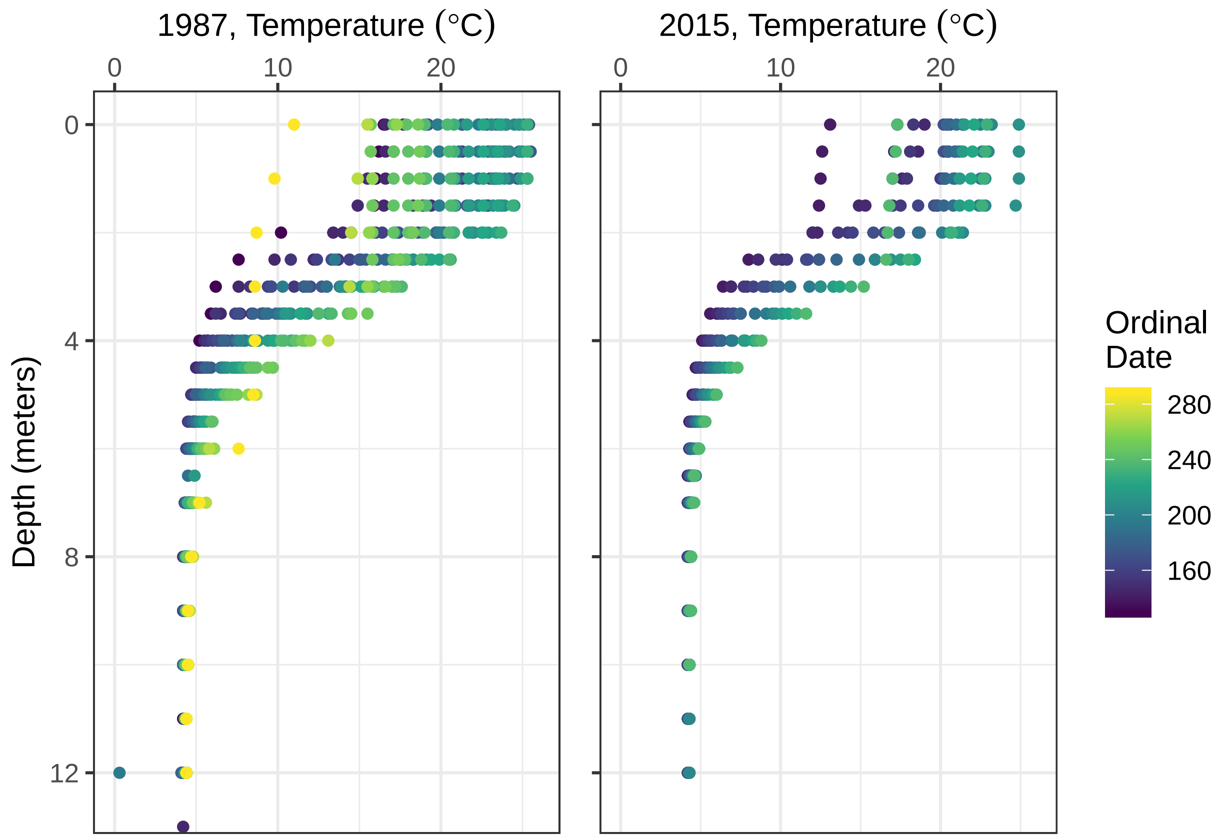
```
## 1987 temperature profile
profile.1987 <- ggplot(data = filter(Tuesday.early.late, year4 == 1987),
                       aes(x = temperature_C, y = depth, color = daynum)) +
  geom_point() +
  scale_y_reverse() +
  scale_x_continuous(position = "top") +
  theme(legend.position = "none") +
  labs(y = "Depth (meters)",
       x = expression("1987, Temperature "(degree*C)),
       color = "Ordinal\nDate") +
  scale_color_viridis_c(limits = c(min(Tuesday.early.late$daynum),
                                   max(Tuesday.early.late$daynum))) +
  coord_cartesian(xlim = c(0, 26), ylim = c(0, 12.5))
print(profile.1987)
```




```r
## 2015 temperature profile
profile.2015 <- ggplot(data = filter(Tuesday.early.late, year4 == 2015),
                       aes(x = temperature_C, y = depth, color = daynum)) +
  geom_point() +
  scale_y_reverse() +
  scale_x_continuous(position = "top") +
  theme(legend.position = "right",
        axis.text.y = element_blank(), axis.title.y = element_blank()) +
  labs(y = "Depth (meters)",
       x = expression("2015, Temperature "(degree*C)),
       color = "Ordinal\nDate") +
  scale_color_viridis_c(limits = c(min(Tuesday.early.late$daynum),
                                   max(Tuesday.early.late$daynum))) +
  coord_cartesian(xlim = c(0, 26), ylim = c(0, 12.5))
print(profile.2015)
```




```r
## Tuesday 1987 and 2015 profiles in one grid plot
profiles.temp <- plot_grid(profile.1987, profile.2015,
                           nrow = 1, rel_widths = c(1, 1.15))
print(profiles.temp)
```



7. Interpret the stratification patterns in your graphs in light of seasonal trends. In addition, do you see differences between the two years?

   In 1987, the epilimnion appears to heat up throughout the year, until approximately ordinal date 200, near the middle of July. After this point, the epilimnion cools down, and at the latest sample date, around the middle of October, the lake is nearly mell-mixed throughout. The hypolimnion remains at nearly the same temperature and depth throughout the sample period. In 2015, the temperature patterns remain similar despite fewer sample points. The epilimnion is however cooler at the first sample point of 2015 than the first sample point of 1987. The maximum temperature of each year is similar, just above 25 degrees, and occurs in mid July.

Multiple lakes, single date

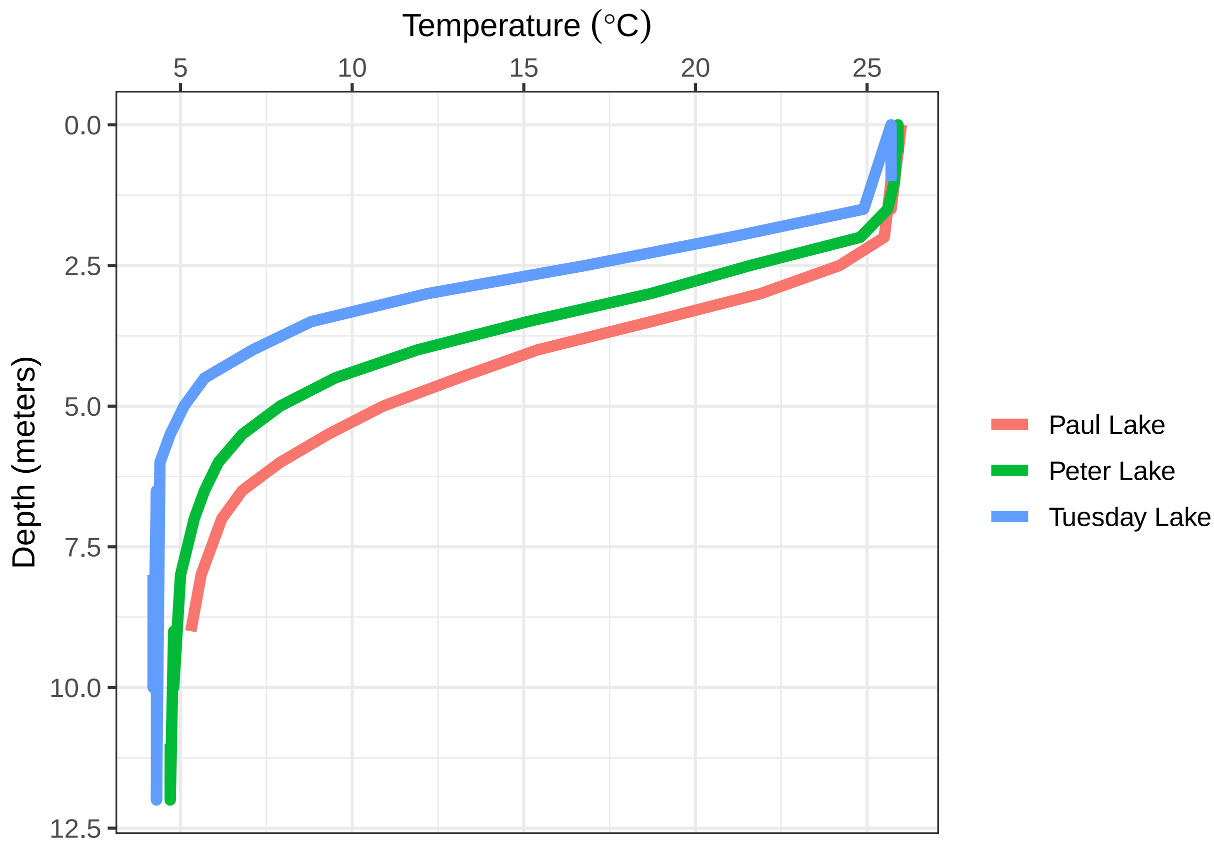
8. On July 25, 26, and 27 in 2016, all three lakes (Peter, Paul, and Tuesday) were sampled. Wrangle your data frame to include just these three dates.

```
## Only the three dates of all three lakes
NTL.July <- NTL.Physical %>%
  filter(lakename %in% c("Peter Lake", "Paul Lake", "Tuesday Lake") &
           (sampledate == "2016-07-25" |
              sampledate == "2016-07-26" |
              sampledate == "2016-07-27"))
```

9. Plot a profile line graph of temperature by depth, one line per lake. Each lake can be designated by a separate color.

```
ggplot(NTL.July, aes(x = temperature_C, y = depth, color = lakename)) +
  geom_line(size = 2) +
```

```r
  scale_y_reverse() +
  scale_x_continuous(position = "top") +
  theme(legend.position = "right") +
  labs(y = "Depth (meters)",
       x = expression("Temperature "(degree*C)),
       color = NULL)
```



10. What is the depth range of the epilimnion in each lake? The thermocline? The hypolimnion?

    For Tuesday Lake, the epilimnion extends to approximately 1.3 m, the thermocline extends from 1.3 m to 5.5 m, and the hypolimnion extends below 5.5 m. For Peter Lake, the epilimnion extends to approximately 1.5 m, the thermocline extends from 1.5 m to 6 m, and the hypolimnion extends below 6 m. For Paul Lake, the epilimnion extends to approximately 2 m, the thermocline extends from 2 m to 7 m, and the hypolimnion extends below 7 m.

Trends in surface temperatures over time.

11. Run the same analyses we ran in class to determine if surface lake temperatures for a given month have increased over time ("Long-term change in temperature" section of day 4 lesson in its entirety), this time for either Peter or Tuesday Lake.

```r
## use Tuesday data across all years
Tuesday.summer <- Tuesday.data %>%
  ## add month column
  mutate(Month = month(sampledate)) %>%
  ## filter to surface and months 5-8
  filter(depth == 0 & Month %in% c(5:8))
```

```r
TuesdayMay <- filter(Tuesday.summer, Month == 5)
TuesdayJune <- filter(Tuesday.summer, Month == 6)
TuesdayJuly <- filter(Tuesday.summer, Month == 7)
TuesdayAugust <- filter(Tuesday.summer, Month == 8)

## linear regression for each month separately
Mayregression <- lm(data = TuesdayMay, temperature_C ~ year4)
Juneregression <- lm(data = TuesdayJune, temperature_C ~ year4)
Julyregression <- lm(data = TuesdayJuly, temperature_C ~ year4)
Augustregression <- lm(data = TuesdayAugust, temperature_C ~ year4)

## output of linear regression
summary(Mayregression) # not significant
```

```
##
## Call:
## lm(formula = temperature_C ~ year4, data = TuesdayMay)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.6223 -1.4411  0.0314  1.5604  5.2216
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -27.15303   73.73032  -0.368    0.715
## year4         0.02196    0.03689   0.595    0.556
##
## Residual standard error: 2.522 on 32 degrees of freedom
## Multiple R-squared:  0.01095,    Adjusted R-squared:  -0.01995
## F-statistic: 0.3544 on 1 and 32 DF,  p-value: 0.5558
```

```r
summary(Juneregression) # not significant
```

```
##
## Call:
## lm(formula = temperature_C ~ year4, data = TuesdayJune)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.0339 -1.5343 -0.0279  1.9180  6.7676
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 21.1373460 50.7897026   0.416    0.678
## year4       -0.0002531  0.0254253  -0.010    0.992
##
## Residual standard error: 2.621 on 80 degrees of freedom
## Multiple R-squared:  1.239e-06,  Adjusted R-squared:  -0.0125
## F-statistic: 9.912e-05 on 1 and 80 DF,  p-value: 0.9921
```

```r
summary(Julyregression) # significant, 0.036 degrees per year
```

```
##
## Call:
## lm(formula = temperature_C ~ year4, data = TuesdayJuly)
```

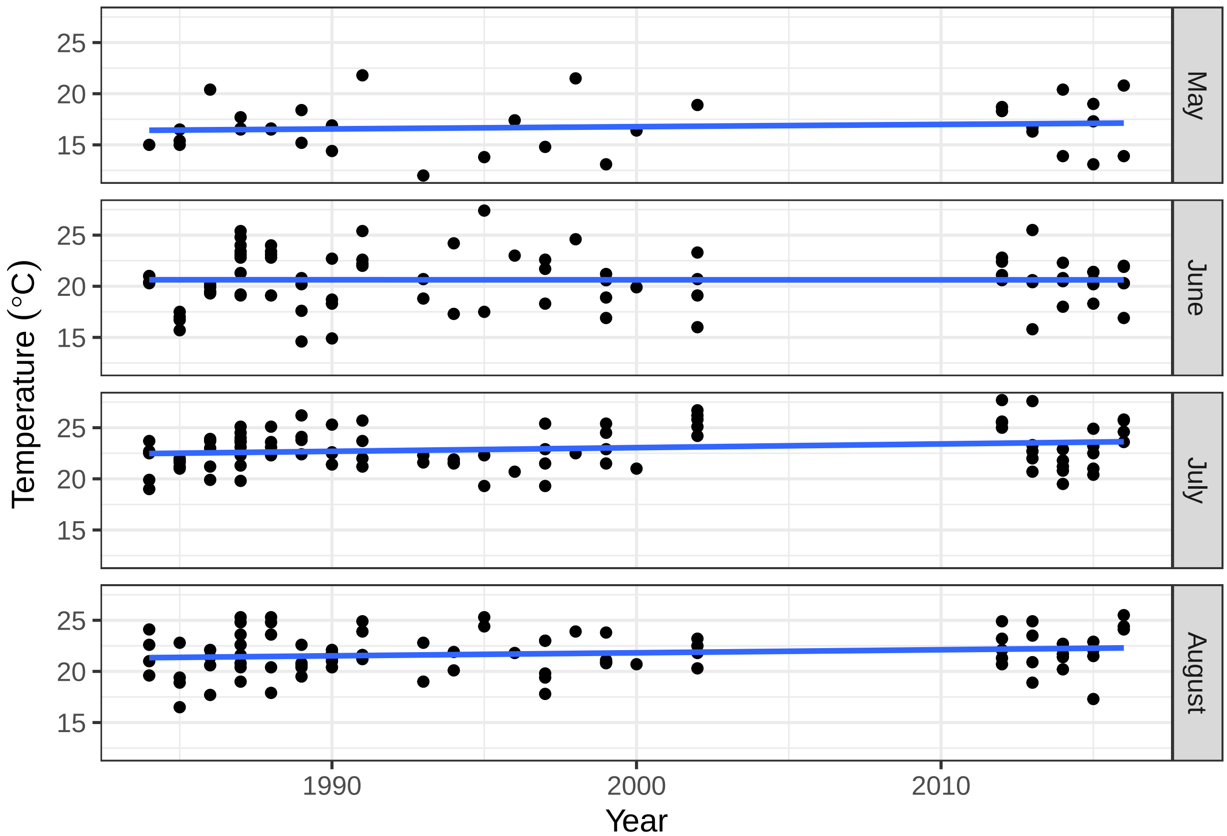
```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.0561 -1.3275 -0.2047  1.4031  4.2161
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -49.18776   37.36614  -1.316   0.1916
## year4         0.03612    0.01871   1.931   0.0569 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.953 on 84 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.04248,    Adjusted R-squared:  0.03109
## F-statistic: 3.727 on 1 and 84 DF,  p-value: 0.05691
```

```r
summary(Augustregression) # not significant
```

```
##
## Call:
## lm(formula = temperature_C ~ year4, data = TuesdayAugust)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.9656 -1.1055 -0.0787  1.2820  3.8677
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -37.70343   41.36954  -0.911    0.365
## year4         0.02976    0.02072   1.436    0.155
##
## Residual standard error: 2.025 on 81 degrees of freedom
## Multiple R-squared:  0.02484,    Adjusted R-squared:  0.0128
## F-statistic: 2.063 on 1 and 81 DF,  p-value: 0.1547
```

```r
## faceted plot
## create month labels vector
month.labs <- c("May", "June", "July", "August")
names(month.labs) <- c(5:8)

ggplot(Tuesday.summer, aes(x = year4, y = temperature_C)) +
  geom_point() +
  facet_grid(rows = vars(Month),
             labeller = labeller(Month = month.labs)) +
  geom_smooth(method = lm, se = FALSE) +
  labs(y = expression("Temperature "(degree*C)),
       x = "Year")
```



12. How do your results compare to those we found in class for Paul Lake? Do similar trends exist for both lakes?

    Only July showed a statistically significant increase in surface temperature over the course of the study for Tuesday Lake. This increase was 0.036 degrees C per year, or 1.15 degrees from 1984 to 2016. Paul Lake had a greater increase in temperature for July, at approximately 0.06 degrees per year, and experienced a significant increase in temperature in August as well.