3: Physical Properties of Lakes

Hydrologic Data Analysis | Kateri Salk

Fall 2019

Lesson Objectives

1. Investigate the concepts of lake stratification and mixing by analyzing monitoring data
2. Apply data analytics skills to applied questions about physical properties of lakes
3. Communicate findings with peers through oral, visual, and written modes

Opening Discussion

What is a lake? How does this differ from a stream, river, or wetland?

What are the physical properties of lakes?

Session Set Up

```
# Check working directory (should be project file location)
getwd()
```

```
## [1] "/Users/katerisalk/Box Sync/Courses/Hydrologic Data Analysis/Lessons"
```

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

# Load data
NTLdata <- read.csv("../Data/Raw/NTL-LTER_Lake_ChemistryPhysics_Raw.csv")

# set ggplot theme
theme_set(theme_classic())
```

Data Wrangling and Exploration

Investigate structure of dataset

Common steps/checks for data wrangling:

- Date formatting
- Addressing NAs
- Subsetting and filtering

```
# Is the date column perceived as a date by R?
class(NTLdata$sampledate)
```

```
## [1] "factor"
```

```
NTLdata$sampledate <- as.Date(NTLdata$sampledate, "%m/%d/%y")
class(NTLdata$sampledate)
```

```
## [1] "Date"
```

```r
# What does this column look like now?

# Remove rows that contain NAs in temperature column
dim(NTLdata)
```

```
## [1] 38614    11
```

```r
NTLdata <- NTLdata %>%
  drop_na(temperature_C)
dim(NTLdata)
```

```
## [1] 34756    11
```

```r
# How many observations are there for each lake?
summary(NTLdata$lakename)
```

```
## Central Long Lake     Crampton Lake    East Long Lake  Hummingbird Lake
##               443              1108              3550               378
##         Paul Lake        Peter Lake      Tuesday Lake         Ward Lake
##              9253             10189              5503               527
##    West Long Lake
##              3805
```

```r
# Let's choose the three lakes with data
NTLdata <- NTLdata %>%
  filter(lakename %in% c("Paul Lake", "Peter Lake", "Tuesday Lake"))

# What is another way to use the filter command to get to the same result?
NTLdata <- NTLdata %>%
  filter(lakename == "Paul Lake" | lakename == "Peter Lake" |
           lakename == "Tuesday Lake")

# Make three data frames, one for each lake
Pauldata <- filter(NTLdata, lakename == "Paul Lake")
Peterdata <- filter(NTLdata, lakename == "Peter Lake")
Tuesdaydata <- filter(NTLdata, lakename == "Tuesday Lake")

# How long did the monitoring last?
min(Pauldata$sampledate)
```

```
## [1] "1984-05-27"
```

```r
max(Pauldata$sampledate)
```

```
## [1] "2016-08-16"
```

```r
min(Peterdata$sampledate)
```

```
## [1] "1984-05-28"
```

```r
max(Peterdata$sampledate)
```

```
## [1] "2016-08-15"
```

```r
min(Tuesdaydata$sampledate)
```

```
## [1] "1984-05-29"
```

```r
max(Tuesdaydata$sampledate)
```

```
## [1] "2016-08-17"
```

```r
# Which depths are sampled in each lake?
unique(Pauldata$depth)
```

```
##  [1]  0.0  1.0  2.0  3.0  4.0  5.0  6.0  7.0  8.0  9.0 10.0 11.0 12.0  0.5
## [15]  1.5  2.5  3.5 20.0  4.5  5.5  6.5  7.5 11.5 13.0  8.5  9.5
```

```r
unique(Peterdata$depth)
```

```
##  [1]  0.00  1.00  2.00  3.00  4.00  5.00  6.00  7.00  8.00  9.00 10.00
## [12] 11.00 12.00 13.00 14.00 15.00 16.00 17.00  0.50  1.50  2.50  3.50
## [23]  4.50  5.50  6.50  7.50  8.50  0.75
```

```r
unique(Tuesdaydata$depth)
```

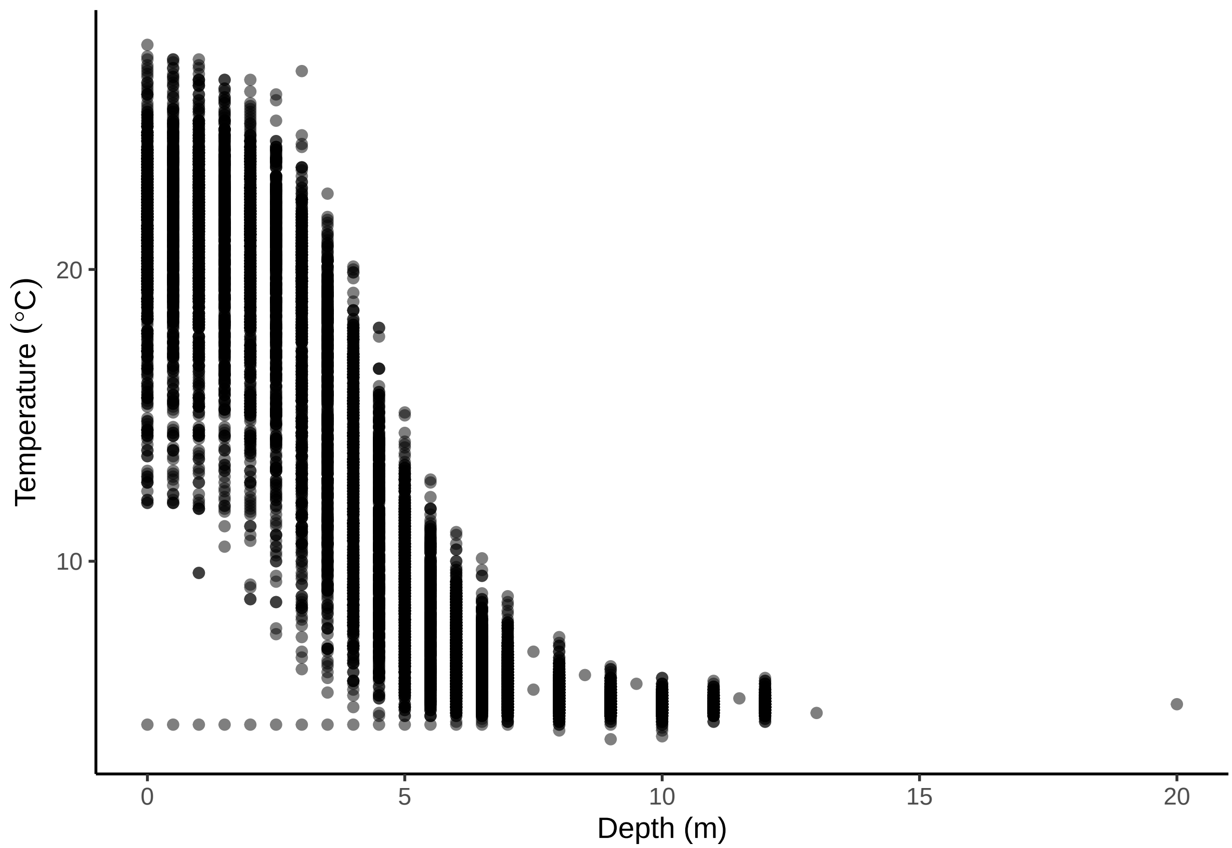
```
##  [1]  0.0  1.0  1.5  2.0  3.0  4.0  5.0  6.0  7.0  8.0  9.0 10.0 12.0  0.5
## [15]  2.5  3.5 14.0 16.0 15.0 13.0 11.0  4.5  5.5  6.5
```

```r
# Why didn't we use the "summary" function here?
```

Exploratory data visualization

Let's make a plot of temperatures by depth. There are a lot of points, so adding a 50 % transparency to the points helps us see where points are densely clustered together.

```r
TempvsDepth <-
  ggplot(Pauldata, aes(x = depth, y = temperature_C)) +
  geom_point(alpha = 0.5) +
  labs(y = expression("Temperature "(degree*C)), x = "Depth (m)")
print(TempvsDepth)
```



How do temperatures at the surface compare to temperatures at the mid-depths and at the bottom?

Let's make a few data frames that include measurements from specific depths. We will choose dates only from the 2000s for ease of interpretation.

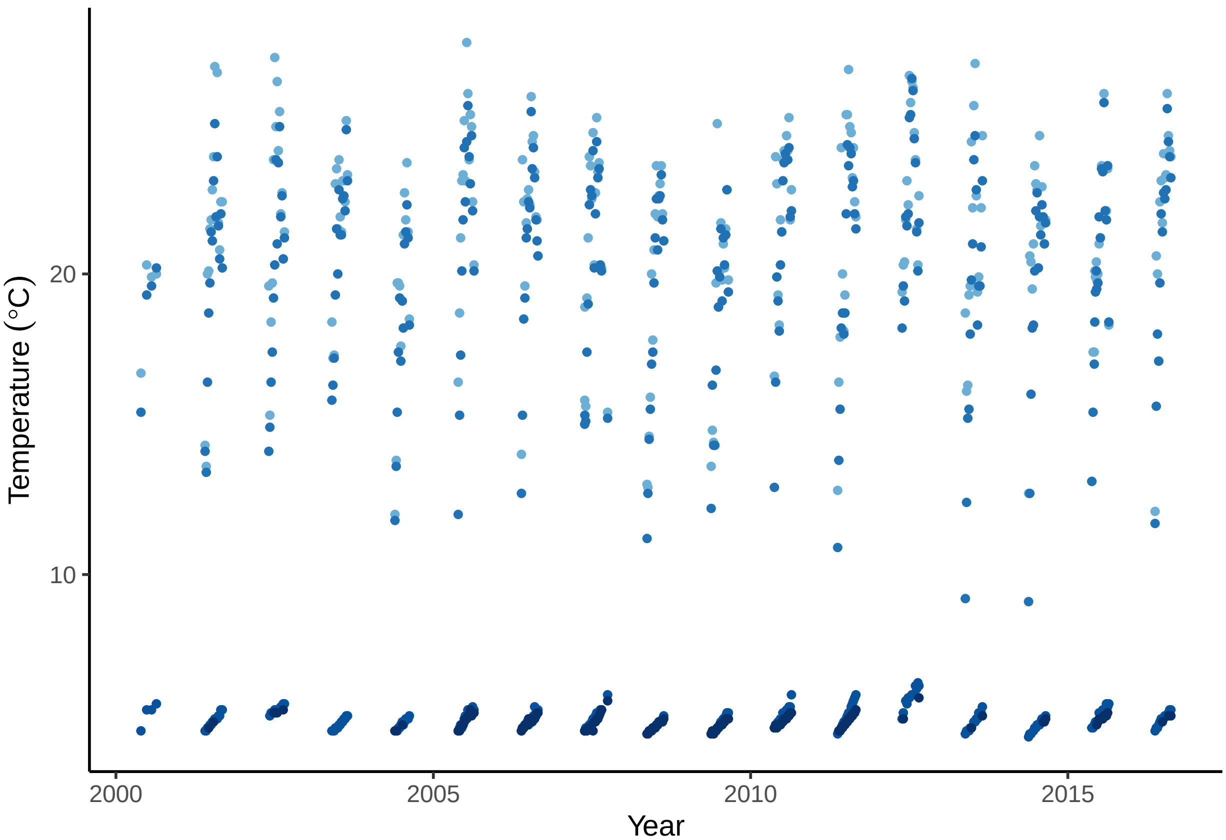
```
Pauldata2000s <- filter(Pauldata, year4 > 1999)
Pauldata.surface <- filter(Pauldata2000s, depth == 0)
Pauldata.2m <- filter(Pauldata2000s, depth == 2)
Pauldata.3m <- filter(Pauldata2000s, depth == 3)
Pauldata.4m <- filter(Pauldata2000s, depth == 4)
Pauldata.5m <- filter(Pauldata2000s, depth == 5)
Pauldata.6m <- filter(Pauldata2000s, depth == 6)
Pauldata.7m <- filter(Pauldata2000s, depth == 7)
Pauldata.9m <- filter(Pauldata2000s, depth == 9)
Pauldata.bottom <- filter(Pauldata2000s, depth == 12)
```

Now let's plot a few of the data frames on the same ggplot. How do temperatures at the surface compare to those at 2 m depth? How do temperatures at the bottom compare to those at 9 m depth?

Here we are highlighting depths considered the **epilimnion** and **hypolimnion**.

```
Tempplot.mixed <-
  ggplot(Pauldata.surface, aes(x = sampledate, y = temperature_C)) +
  geom_point(color = "#6baed6", size = 1) +
  geom_point(data = Pauldata.2m, aes(x = sampledate, y = temperature_C),
             color = "#2171b5", size = 1) +
  geom_point(data = Pauldata.9m, aes(x = sampledate, y = temperature_C),
             color = "#08519c", size = 1) +
```

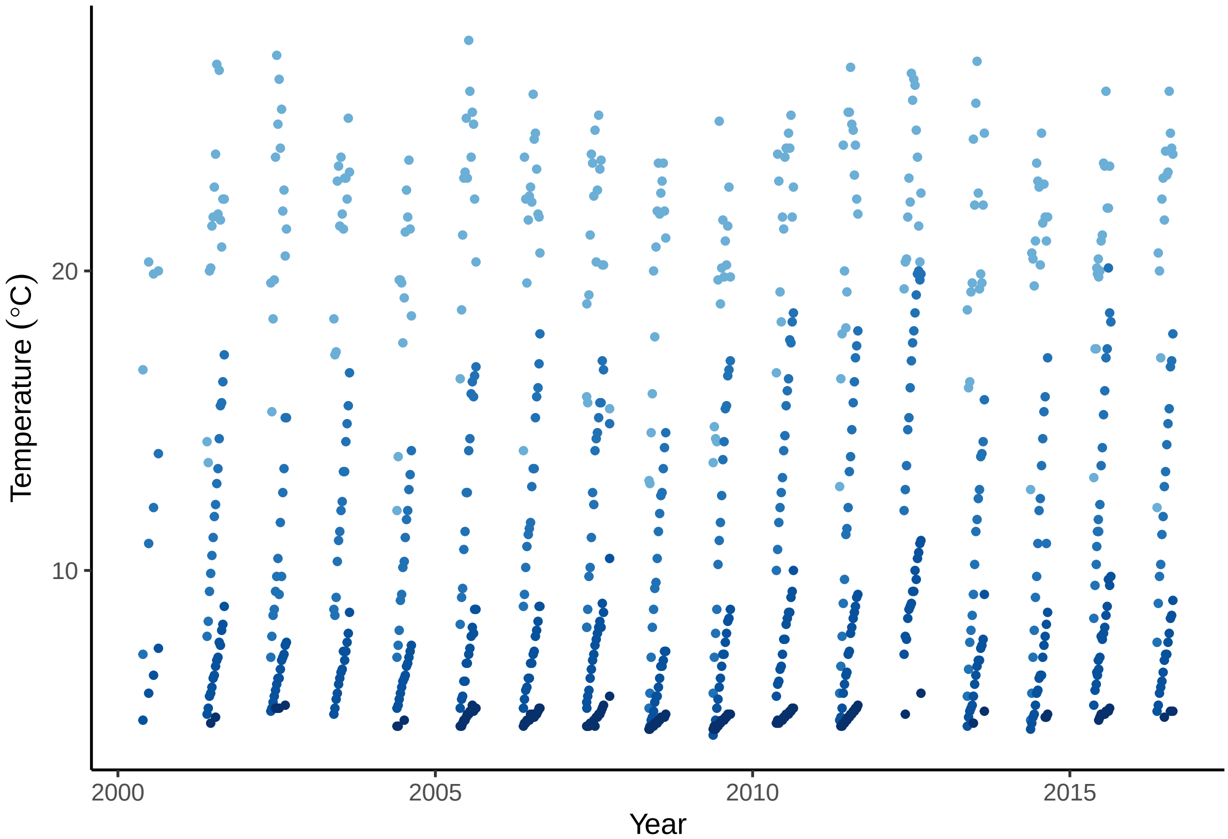
```
  geom_point(data = Pauldata.bottom, aes(x = sampledate, y = temperature_C),
             color = "#08306b", size = 1) +
  labs(y = expression("Temperature "(degree*C)), x = "Year")
print(Tempplot.mixed)
```



Now let's take out the 2 m and 9 m points and add in the 4 m and 6 m points. How do these compare to the surface and bottom temperatures?

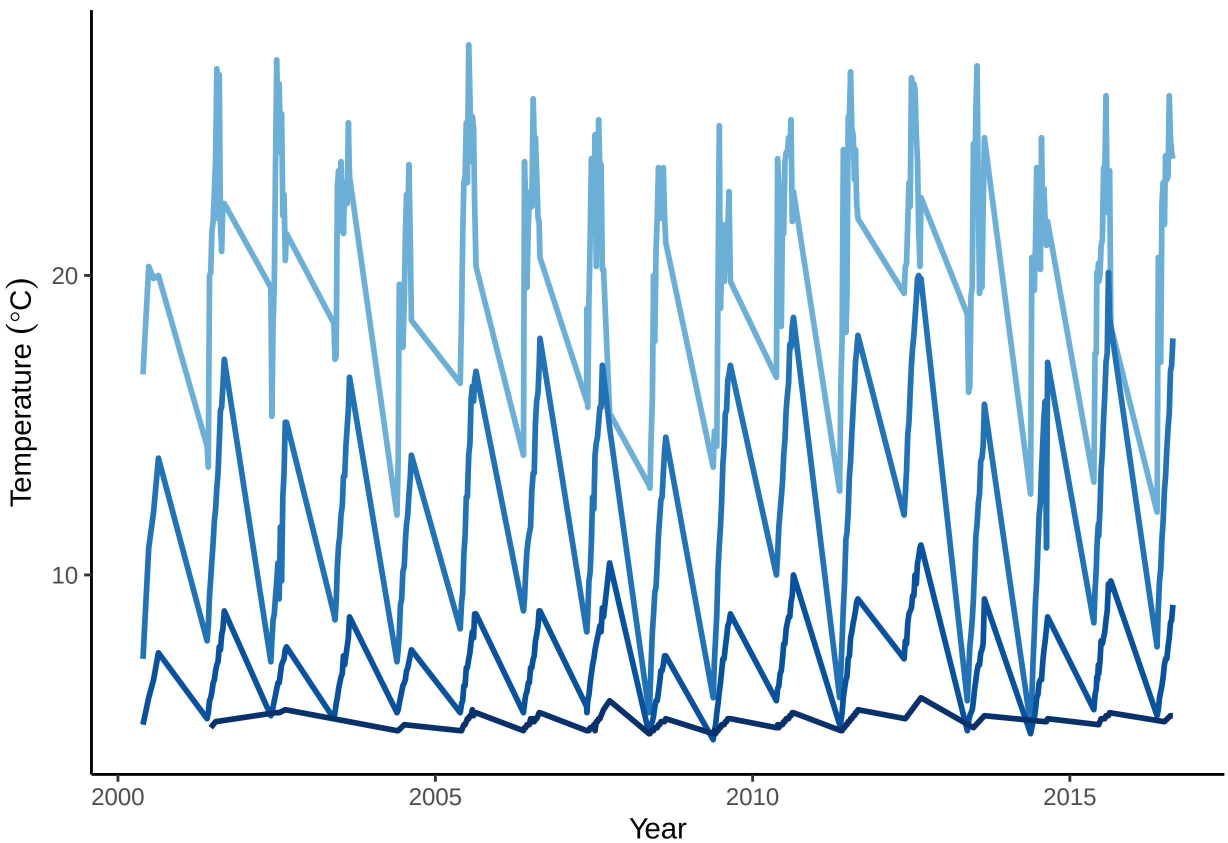
Here are are highlighting depths considered the **metalimnion** or **thermocline**.

```
Tempplot.thermocline <-
  ggplot(Pauldata.surface, aes(x = sampledate, y = temperature_C)) +
  geom_point(color = "#6baed6", size = 1) +
  geom_point(data = Pauldata.4m, aes(x = sampledate, y = temperature_C),
             color = "#2171b5", size = 1) +
  geom_point(data = Pauldata.6m, aes(x = sampledate, y = temperature_C),
             color = "#08519c", size = 1) +
  geom_point(data = Pauldata.bottom, aes(x = sampledate, y = temperature_C),
             color = "#08306b", size = 1) +
  labs(y = expression("Temperature "(degree*C)), x = "Year")
print(Tempplot.thermocline)
```



Why can't we use a line graph here?

```
Tempplot.line <-
  ggplot(Pauldata.surface, aes(x = sampledate, y = temperature_C)) +
  geom_line(color = "#6baed6", size = 1) +
  geom_line(data = Pauldata.4m, aes(x = sampledate, y = temperature_C),
             color = "#2171b5", size = 1) +
  geom_line(data = Pauldata.6m, aes(x = sampledate, y = temperature_C),
             color = "#08519c", size = 1) +
  geom_line(data = Pauldata.bottom, aes(x = sampledate, y = temperature_C),
             color = "#08306b", size = 1) +
  labs(y = expression("Temperature "(degree*C)), x = "Year")
print(Tempplot.line)
```

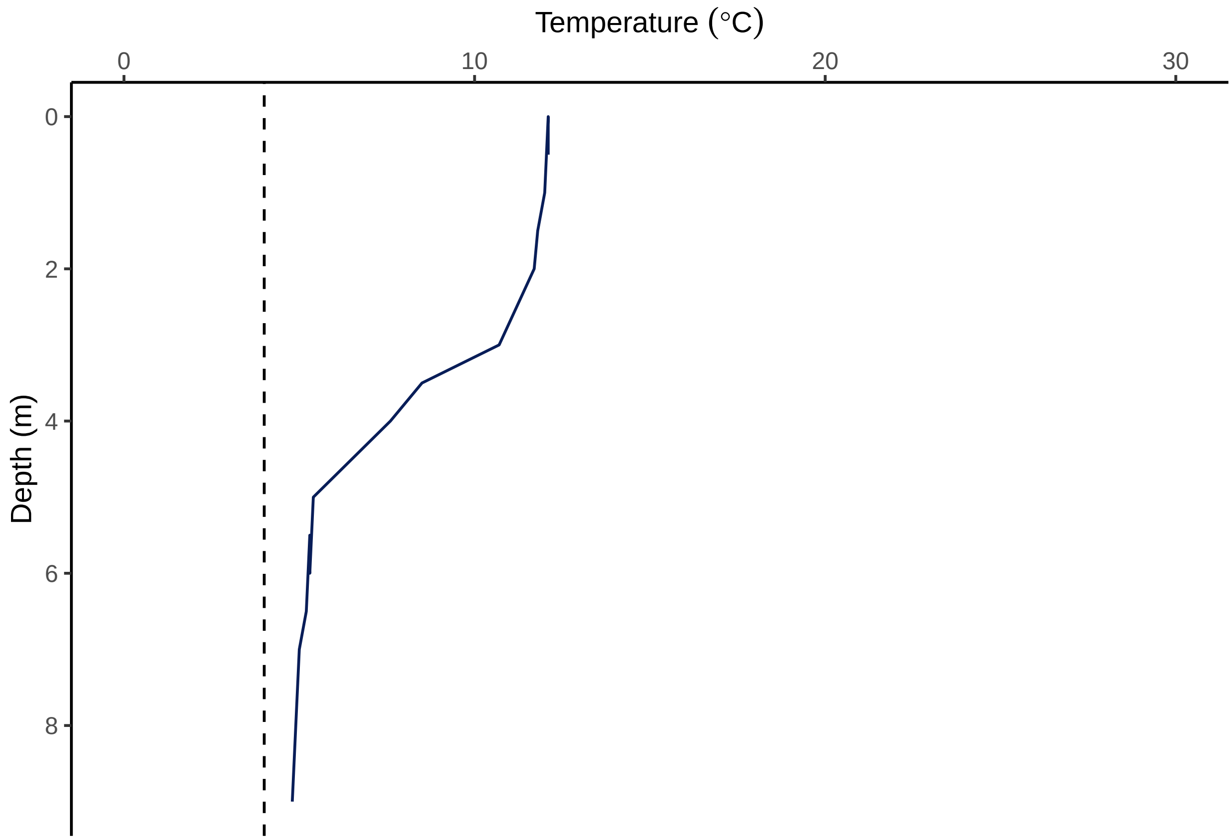


Data Visualization and Analysis

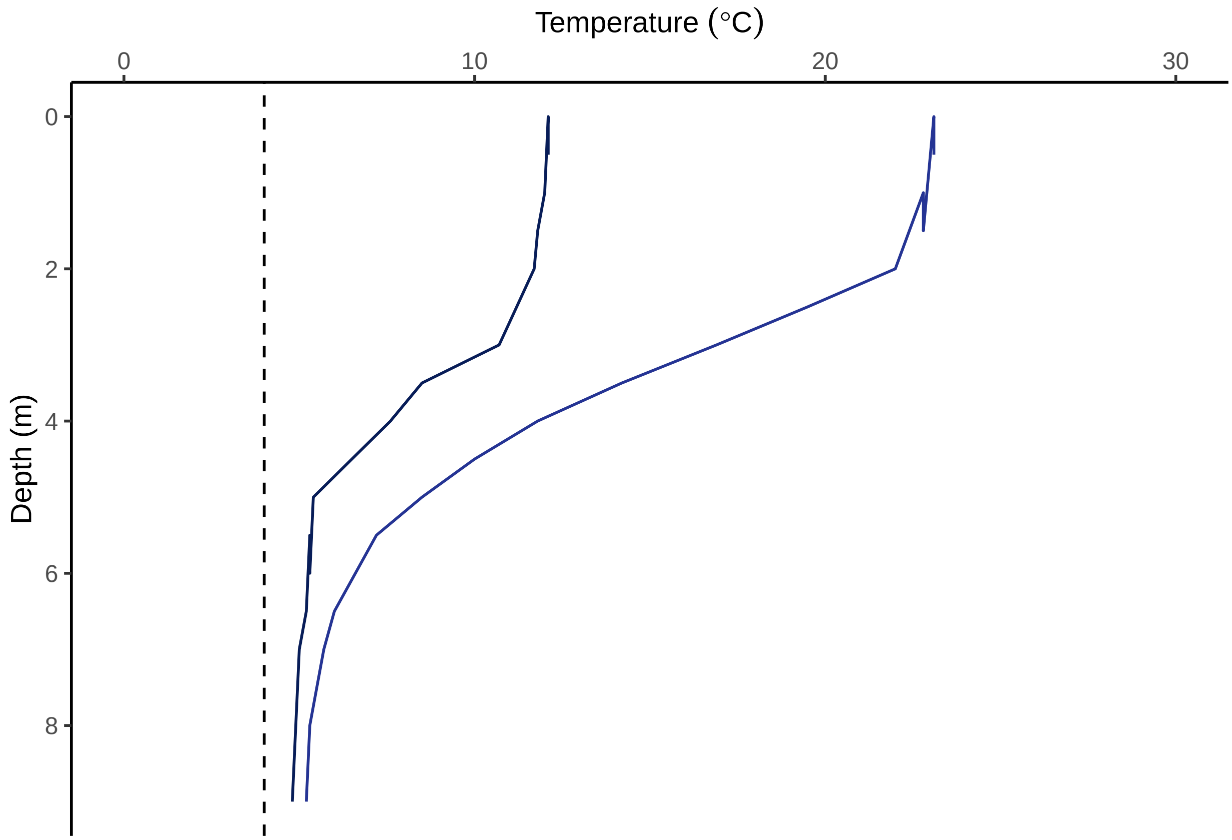
Creating profile graphs

The field of **limnology**, the study of inland waters, uses a unique graph format to display relationships of variables by depth in a lake (the field of oceanography uses the same convention). Depth is placed on the y-axis in reverse order and the other variable(s) are placed on the x-axis. In this manner, the graph appears as if a cross section were taken from that point in the lake, with the surface at the top of the graph.

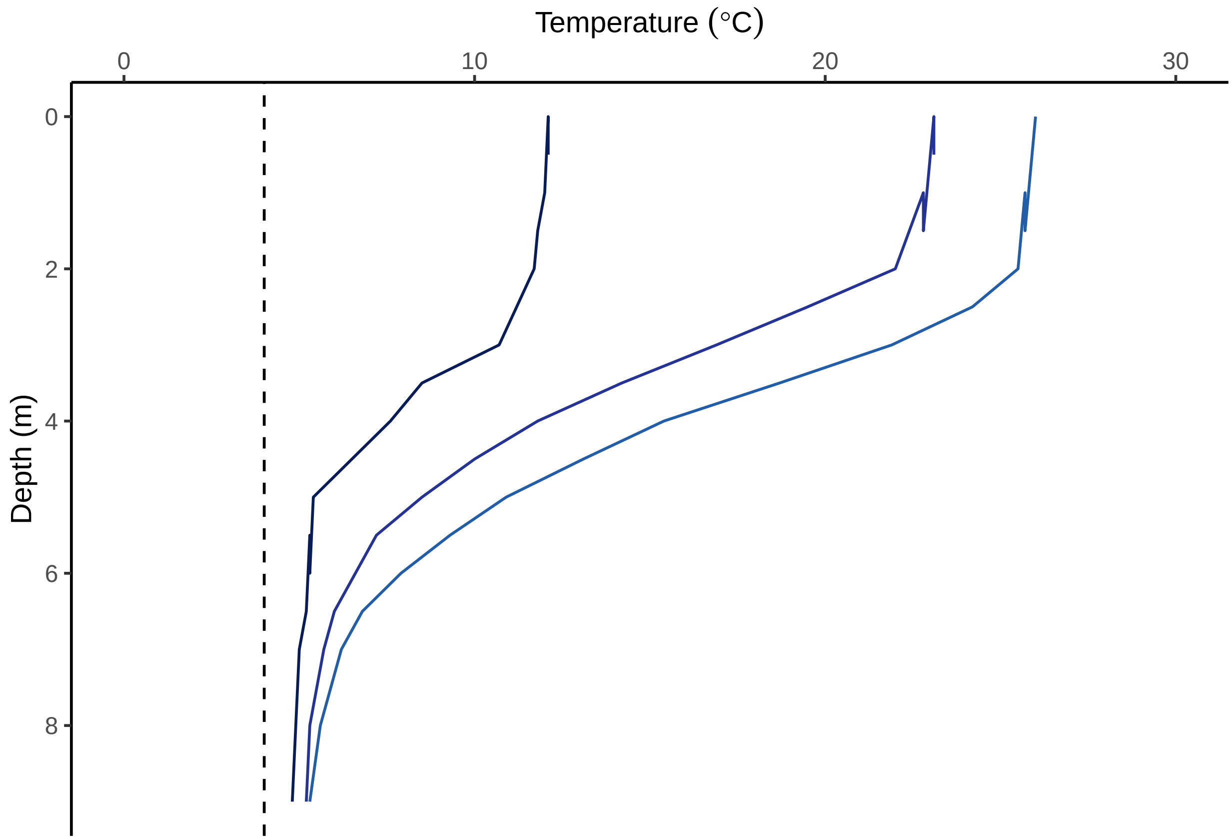
```
Pauldata.2016 <- filter(Pauldata, year4 == 2016)
Pauldata.May2016 <- filter(Pauldata, sampledate == "2016-05-17")
Pauldata.June2016 <- filter(Pauldata, sampledate == "2016-06-21")
Pauldata.July2016 <- filter(Pauldata, sampledate == "2016-07-26")
Pauldata.Aug2016 <- filter(Pauldata, sampledate == "2016-08-16")

TempprofileMay2016 <-
  ggplot(Pauldata.May2016, aes(x = temperature_C, y = depth)) +
  geom_line(color = "#081d58") +
  geom_vline(xintercept = 4, lty = 2) +
  scale_y_reverse(breaks = c(0, 2, 4, 6, 8, 10, 12)) +
  scale_x_continuous(position = "top", limits = c(0, 30)) +
  labs(x = expression("Temperature "(degree*C)), y = "Depth (m)")
print(TempprofileMay2016)
```



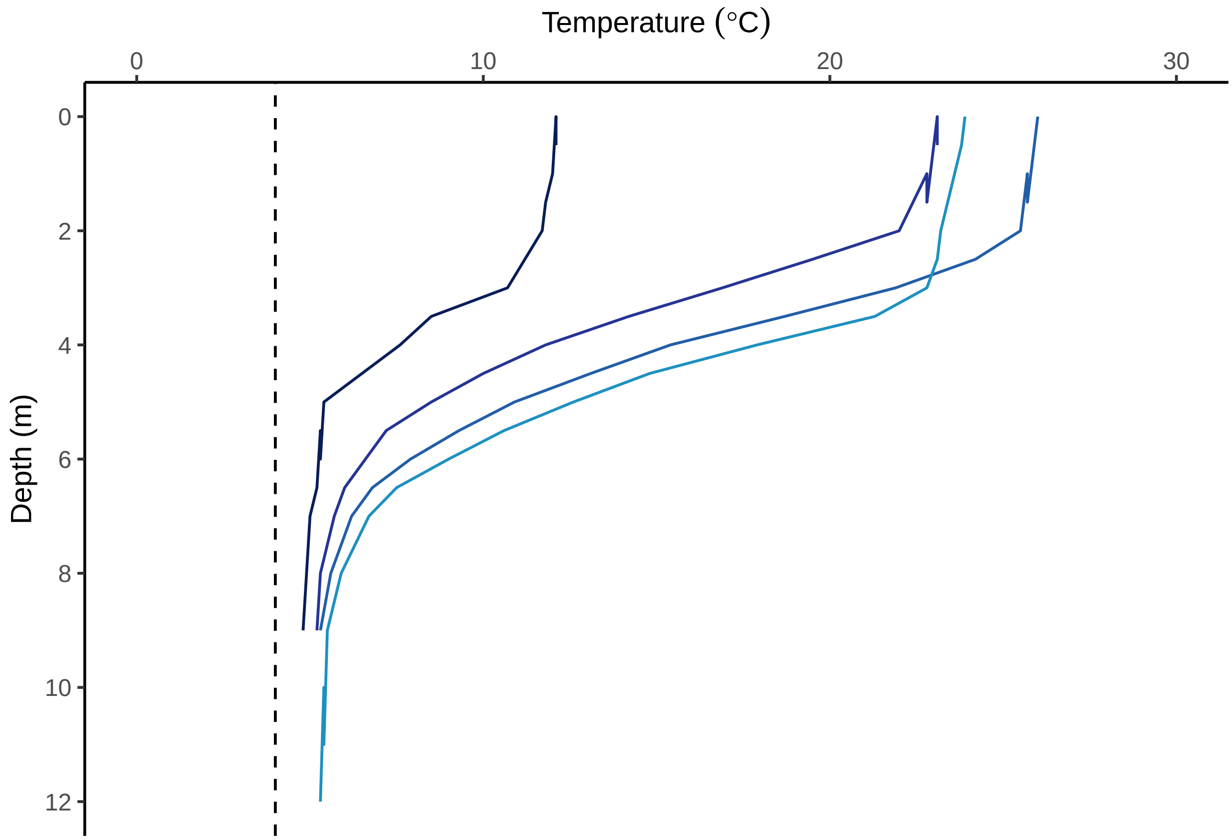

```r
TempprofileJune2016 <-
  TempprofileMay2016 +
  geom_line(data = Pauldata.June2016, aes(x = temperature_C, y = depth),
            color = "#253494")
print(TempprofileJune2016)
```




```
TempprofileJuly2016 <-
  TempprofileJune2016 +
  geom_line(data = Pauldata.July2016, aes(x = temperature_C, y = depth),
            color = "#225ea8")
print(TempprofileJuly2016)
```




```
TempprofileAug2016 <-
  TempprofileJuly2016 +
  geom_line(data = Pauldata.Aug2016, aes(x = temperature_C, y = depth),
            color = "#1d91c0")
print(TempprofileAug2016)
```



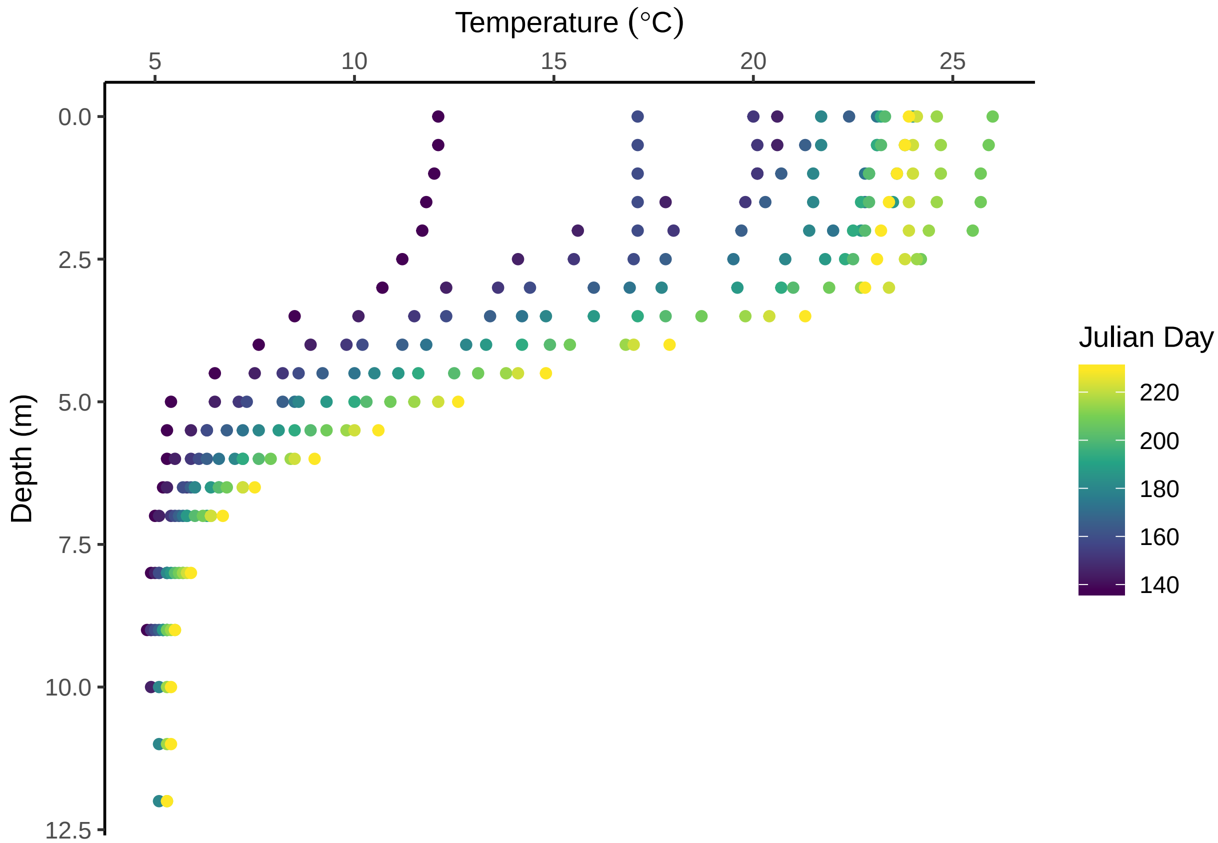
In some places, the lines are not consistently smooth. What's going on here? How does this differ from your expectations of what the data should look like?

If you were to explain this graph to someone who didn't know anything about lakes, how would you describe it? Write your answer below.

Mixing and stratification

Let's visualize all of the sampled dates in 2016 at once.

```
Tempprofiles2016 <-
  ggplot(Pauldata.2016, aes(x = temperature_C, y = depth, color = daynum)) +
  geom_point() +
  scale_y_reverse() +
  scale_x_continuous(position = "top") +
  scale_color_viridis_c() +
  labs(x = expression("Temperature "(degree*C)), y = "Depth (m)",
       color = "Julian Day")
print(Tempprofiles2016)
```



Paul Lake, like many lakes in temperate climates, is **dimictic**. What does this mean? What evidence do you see of this phenomenon in the graph? If you had data from the rest of the year, what would you expect to see?

Let's put your prediction to the test. In 1993, the lake was sampled in November. Wrangle your data to capture this date and create a profile graph of that date.

Why does mixing occur in the spring and fall? What are the mechanisms that make this possible?

Closing Discussion

What are the main concepts you learned about the physical properties of lakes today? What was the evidence for these concepts in the dataset?