

- 1 Introduction and Project Goals
- 2 Methods Overview
- 3 Takeaways from Vermont Lakes
- 4 Conclusions
- 5 Bibliography
- 6 Code Guide

# Vermont's Sentinel Lakes - Preliminary Results from 5 Years of Data

Katelynn Warner, PhD Candidate, Vermont Limnology Lab,  
University of Vermont

January 31, 2023

## 1 Introduction and Project Goals

Climate change is warming lake water temperatures globally, though the extent of warming is much more dramatic in northern, temperate, ice-covered lakes (O'Reilly et al., 2015). Specifically, lakes that experience ice-cover are experiencing increases in their surface water at a rate that is two times faster than ambient air temperatures. Such increases in water temperature can drive shifts in the thermal structures in lakes, which, in turn, can disrupt lake-wide physical and biological processes (Adrian et al., 2009; O'Reilly et al., 2015).

While it is evident that climate change is warming lakes, the effect of increases in water temperatures will vary depending on lake morphology, mixing patterns, and surface area. For instance, smaller lakes (<0.5 km<sup>2</sup>) experience increasing near surface water temperatures due to climate change, but experience variable trends on bottom water temperatures (Winslow et al., 2015). Warming in small lakes tends to be slower than in larger lakes (Richardson et al., 2017; Winslow et al., 2015) because smaller, inland lakes are often more sheltered and have a smaller fetch, so wind mixing may be less frequent. With less mixing, the warming surface waters don't integrate as often with the bottom waters, resulting in slower overall water temperature warming. Similarly, dimictic lakes warm more slowly than polymictic lakes because stratification is stronger and mixing events are less frequent. Therefore, understanding the underlying morphology and mixing patterns of lakes will help us understand how they may respond to climate warming.

Lakes act as sentinels for climate change because they are sensitive to environmental change, and several physical, chemical and biological factors respond to climate forcing (Adrian et al., 2009). Changes to lakes' thermal structure, stratification and mixing patterns can be indicative of warming air temperatures and changes in wind patterns and precipitation. However, it is difficult to separate climate change signals in lakes from other anthropogenic disturbances (eutrophication, acidification, etc.) due to the potential confounding synergistic or antagonistic effects (Adrian et al., 2009; Christensen et al., 2006; Folt et al., 1999; Wilkinson et al., 2020). Eutrophication can increase phytoplankton abundance, decreasing availability

of light in the surface waters. Similarly, acidification decreases dissolved organic content in lakes, affecting the light transparency. Changes in light transparency into a lake will shift heating potential at different depths, affecting lake stratification. These different alterations to lakes can skew our perception of climate change on aquatic systems (Carpenter 2007). Most studies in lakes occur in popular, large lakes with high recreational value (Downing, 2008; Winslow et al., 2015b). Yet, less work is conducted in small, minimally disturbed lakes. Using minimally disturbed lakes as sentinels for climate change, we can begin to tease apart the effects of climate from synergistic or antagonist effects with other anthropogenic disturbances.

The regional monitoring network (RMN) is a collaboration with the United States Environmental Protection Agency (USEPA) and regional, state, local and tribe partners that focuses on increasing monitoring efforts in freshwater inland lakes in the upper Midwest and Northeastern portion of the United States (Stamp, 2018). The Vermont Department of Conservation, as a participant in the regional monitoring network, is currently monitoring five sentinel lakes (lakes that experience minimal anthropogenic disturbances) across the state of Vermont. The long-term goal of the Regional Monitoring Network project is to determine the regional effects of climate change on lakes. The Vermont dataset currently has a maximum of five years of data in it, therefore, long-term trend analysis is not possible (Wilkinson et al., 2020). However, this project will begin to assess interannual variability in thermal patterns within three lakes. The key research aims for this project are to:

- 1. Investigate baseline thermal regimes and structures in three lakes with five years of data across the state of Vermont**
- 2. Assess the interannual variability in each lakes' thermal structure, including: surface water temperatures, length of thermal stratification and ice cover, stratification strength, and thermocline depth**
- 3. Identify if trends exist in thermal regimes that may be a result of climate change**
- 4. Create a step-by-step guide to provide to partners within the Regional Monitoring Network to use to help process their RMN data**

## 1.0.1 Holland Pond



*Photo of Holland Pond Buoy. Photo Credit: Kellie Merrell.*

Holland Pond is located in Holland, VT, has a surface area of 1.3 km<sup>2</sup> (329 acres), lies at 1430 ft in elevation and has a max depth of 39 feet. The basin to lake ratio is 13. Much of the watershed and lakeshore is in the 9,040 acre Bill Sladyk Wildlife Management Area. The VT DEC has been monitoring Spring total phosphorus (TP) levels in Holland pond since 1981, and spring secchi transparency since 1982. The mean long-term spring TP is 9 ug/L and the mean secchi transparency is 3.2m. Though minimally disturbed, the lake has a highly significantly increasing phosphorus trend, and while oligotrophic, the lake has become mesotrophic in recent years. Holland Pond has been monitored for 5 years with the high frequency buoy.

## 1.0.2 Long Pond



*Photo of Long Pond. Photo credit: Kellie Merrell*

Long Pond is located in Greensboro, VT within a 767 acre natural area protected by the Nature Conservancy. Long Pond has a surface area of 0.4 km<sup>2</sup> (99 acres), lies at 1590 ft elevation, and has a max depth of 33ft. The basin to lake area ratio is 19. The VT DEC has been monitoring spring total phosphorus levels and secchi transparency since 1991. The mean spring TP is 16.5 ug/L and the mean secchi transparency is 3.4m. Long Pond is mesotrophic and has a significantly increasing phosphorus trend.



## 1.0.3 High Pond



*Photo of High Pond Buoy. Photo Credit: Kellie Merrell*

High Pond is located in Sudbury, VT and is located in a 3000 acre natural area protected by the Nature Conservancy. High pond has a surface area of 0.07 km<sup>2</sup> (17.5 acres), lies at 1033 ft in elevation and has a max depth of 56 feet. The basin to lake ratio is 110. The VT DEC has been monitoring Spring total phosphorus levels in high pond , and spring secchi transparency since 1994. The mean long-term spring TP is 22.9 ug/L and the secchi transparency is 5.3m. Though minimally disturbed and protected within a natural area, the lake has a highly significantly increasing phosphorus trend, and while still mesotrophic, it's becoming more eutrophic.

## 2 Methods Overview

Each lake is currently being monitored with Onset continuous sensors. In lakes that have motorboat access, the buoys were deployed 1m below the water's surface to prevent damage to the buoys. Temperature sensors were placed every meter and record lake temperature every hour. The thermal profiles are downloaded every twice a year, in late spring and fall. Additionally, each lake has a dissolved oxygen sensor and water level sensor that is located one meter from the bottom of the lake, and an air temperature and pressure sensor located on shore. Data files were loaded into Hoboware software and exported as .csv files. All files were then run through QA/QC protocols through the Continuous Data QC Shiny App (<https://tetrattech-wtr-wne.shinyapps.io/ContDataQC/>) designed by Tetra Tech. Below is a table of the QC

metric used in this analysis. Upon further investigation in R, values that were evidently incorrect and associated with days the buoys were pulled and removed from the water to download files, were manually removed.

*Thresholds, Quick Reference*

Analyte	Flag	Gross	Spike	Rate of Change	Flat Line
Temperature, Water (deg C)	Fail	> 30 deg C or < -2 deg C	>= 5 deg C (+/-)	NA	> 30 consecutive measurements within 0.01 units of one another
Temperature, Water (deg C)	Suspect	> 25 deg C or < -0.1 deg C	>= 3 deg C (+/-)	>= 3 standard deviations within 25 hours	> 20 consecutive measurements within 0.01 units of one another
Dissolved Oxygen (mg/L)	Fail	> 20 mg.L or < 1 mg.L	>= 10 mg.L (+/-)	NA	> 60 consecutive measurements within 0.01 units of one another
Dissolved Oxygen (mg/L)	Suspect	> 18 mg.L or < 2 mg.L	>= 5 mg.L (+/-)	>= 3 standard deviations within 25 hours	> 30 consecutive measurements within 0.01 units of one another

**Table 1:** *Different thresholds used to QA/QC high frequency data collected.*

To help determine trends that may be occurring as a result of climate change, anomalies were calculated. In climate change studies temperature anomalies are more important in understanding change over time than absolute temperature values. A temperature anomaly is defined as the difference in temperature from a baseline temperature. If a temperature anomaly is positive, then the temperature is warmer than average for that period of time. If the temperature anomaly is negative, then the temperature for that period is colder than average. Briefly, daily averages were calculated for each variable being analyzed. Then, each daily average over the length of the sampling period was averaged together to get a “three-year average,” or baseline measurement. The daily average for that variable was subtracted from the three-year daily average in order to determine whether or not the daily value was higher or lower than the average.

For example: If there were three consecutive years of data being collected for surface water temperature, all values for each individual day would be averaged to obtain the daily average (ex: June 15, 2019; June 15, 2020; and June 15, 2021). Then, if I wanted to know how much warmer or cooler June 15th was each year than the 3 year average, I would take Temp (June 15, 2019) - Temp (Average June 15th) and so on.

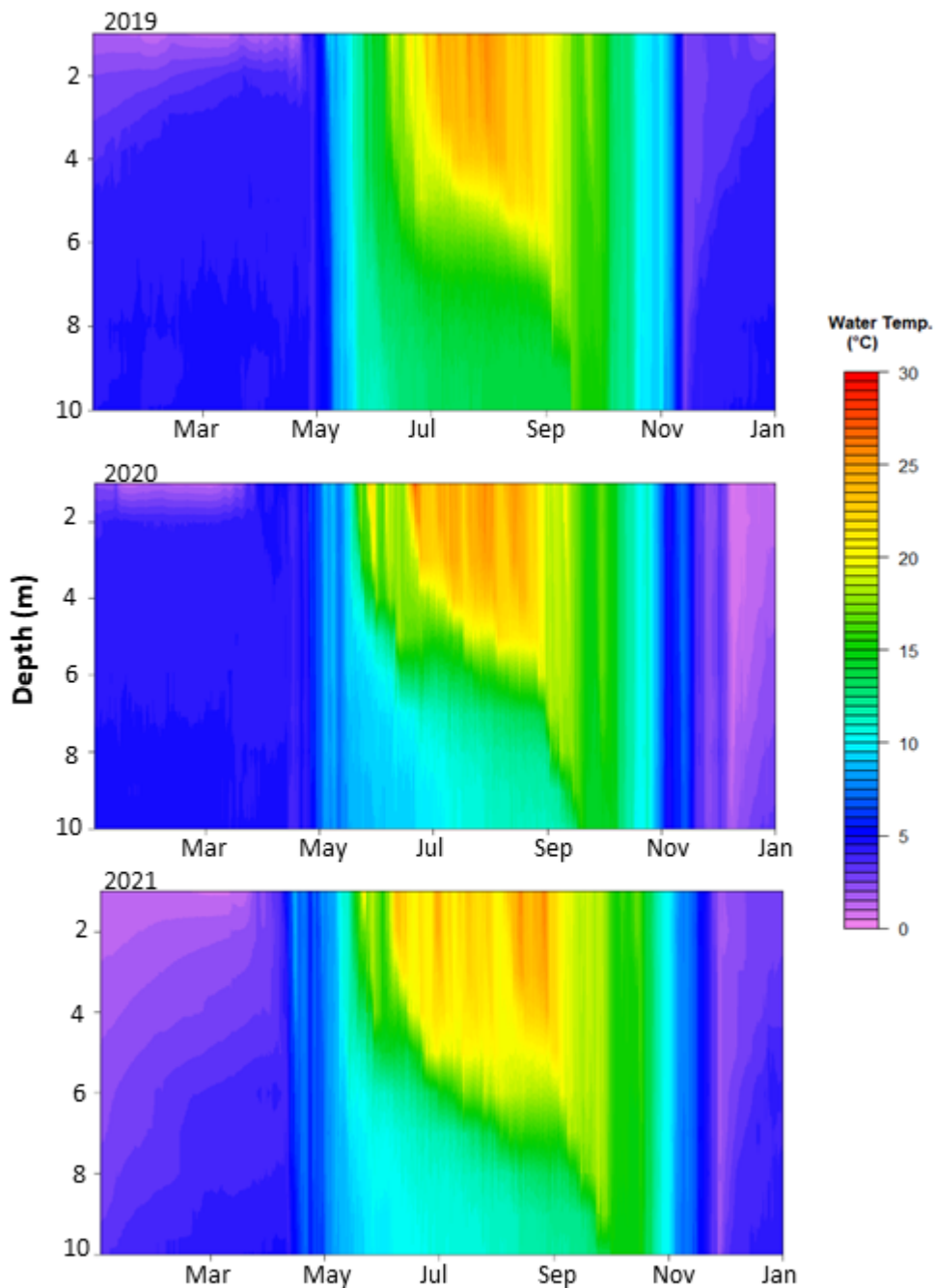
### 3 Takeaways from Vermont Lakes

The next portion of this report will go over the results of the data analysis after the continuous data was run through the quality control procedure in ContDataQC.

# 3.1 Thermal Regimes in each Lake

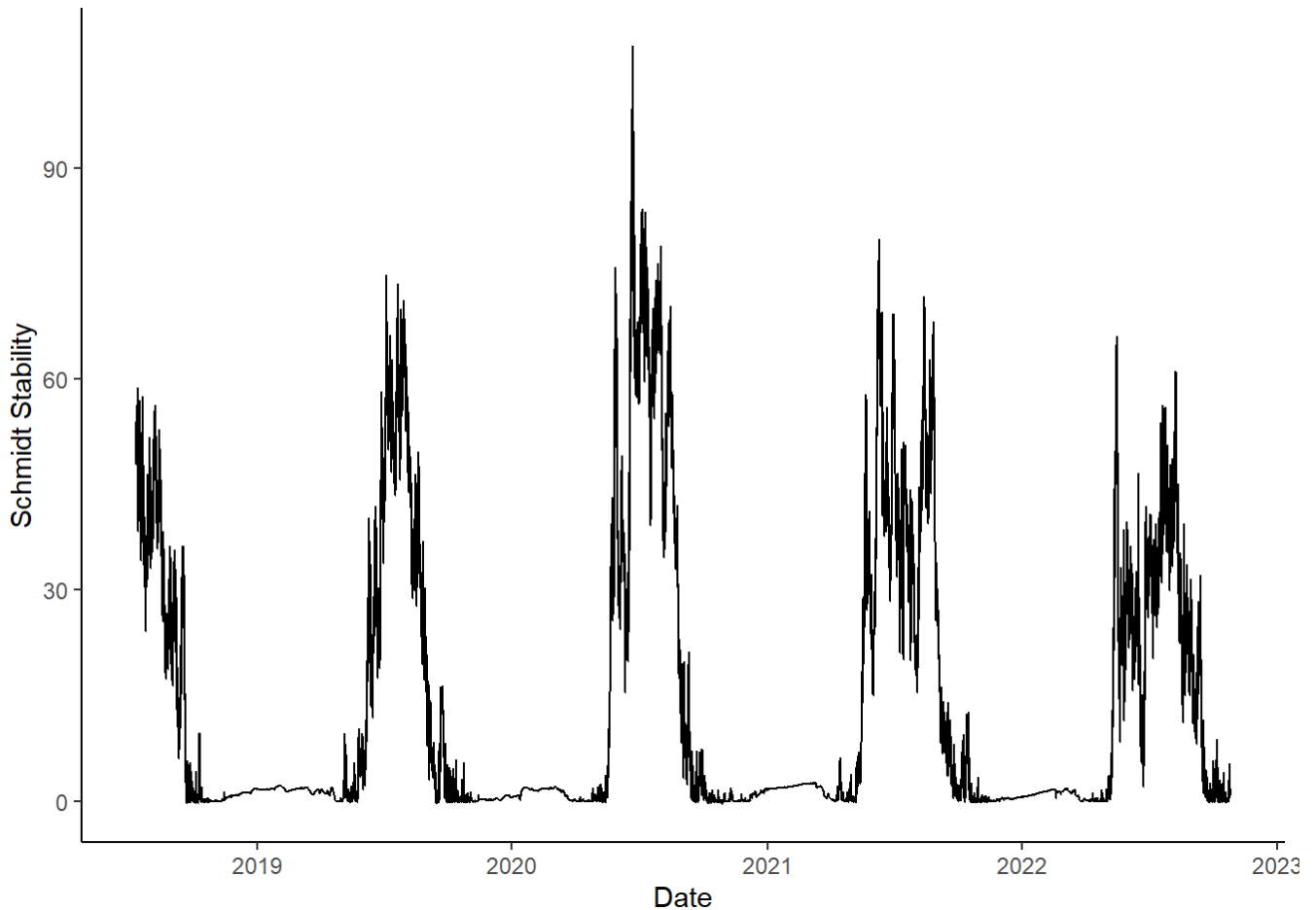
## 3.1.1 Holland

Holland Pond is a dimictic lake that mixes fully once in the spring and once in the fall. Thermal profiles measured in Holland Pond to date show that the lake has been getting warmer in the spring and the fall, which leads to increased duration of the stratified season (**Figure 1**). Spring turnover occurred in late April or early May, and fall turnover occurred in mid to late September, however, there was usually a period of re-stratification around October.



**Figure 1.** Thermal Heatmaps for the three full years of data collection in Holland Pond.

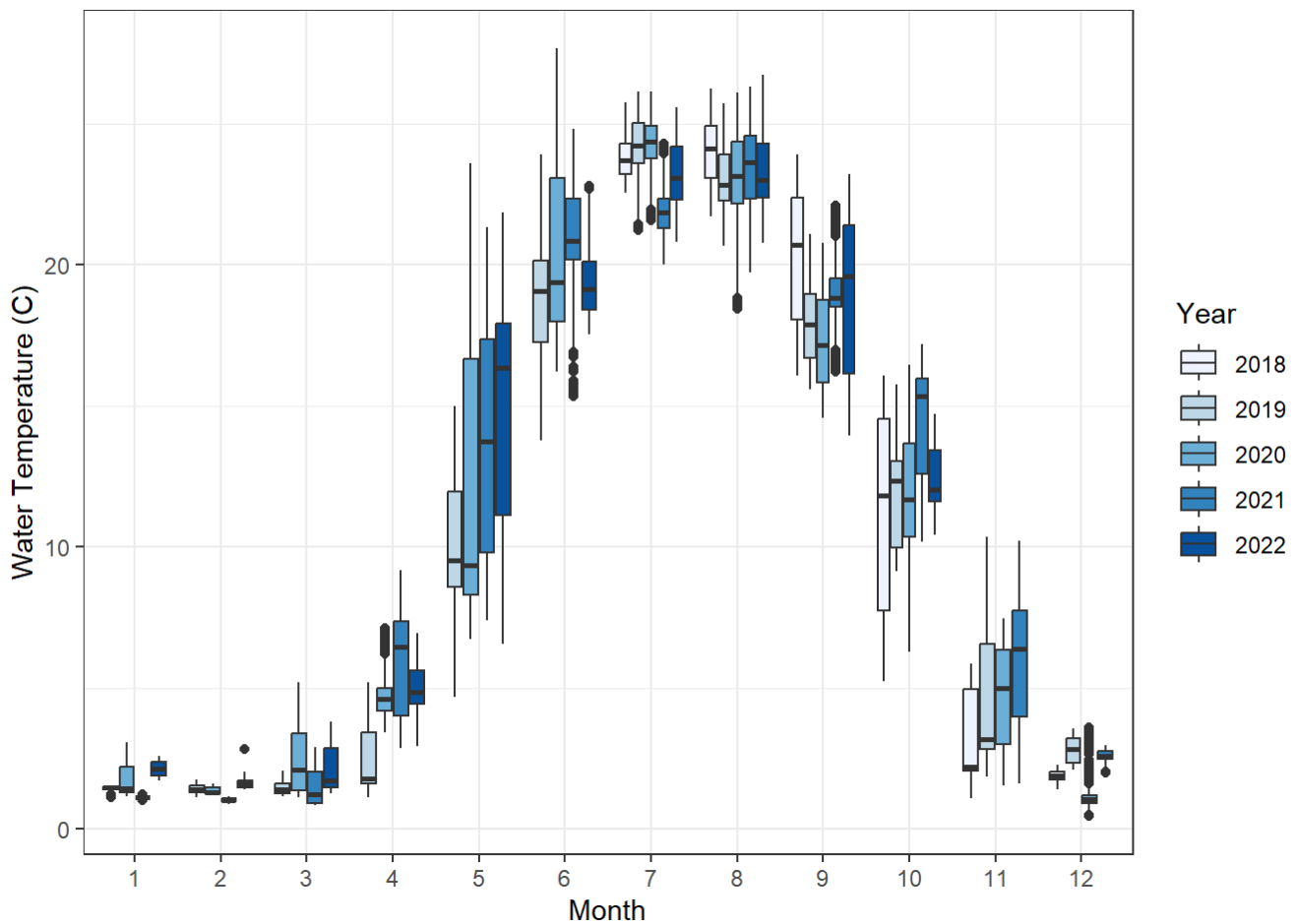
The schmidt stability index of a lake measures how much wind energy is required to fully mix a water column. If the value is high, then the lake is stratified, and it would require a lot of energy to mix the water column fully. If the schmidt stability is low, the lake is either unstratified, or weakly stratified, meaning that only a little bit of wind energy is required to mix the lake.



**Figure 2.** *Holland Pond's Schmidt Stability over time.*

In Holland Pond, we see an interesting pattern in Schmidt Stability occurring (**Figure 2**). As we would expect, schmidt stability is highest in the summer period, and relatively low in the winter period. In Holland Pond, we see a trend of increasing stability in the summer until 2021, where schmidt stability begins to decrease. It appears that over the sampling period Holland Pond became more weakly stratified in the summer seasons. Conversely, it may be that the summer of 2020 was an anomaly, where there was a strong difference in temperatures. The colorful thermal heat maps (**Figure 1**) show that in 2019 and 2021, water temperatures seem to be warmer deeper in the water column, which could account for the differences in schmidt stability.



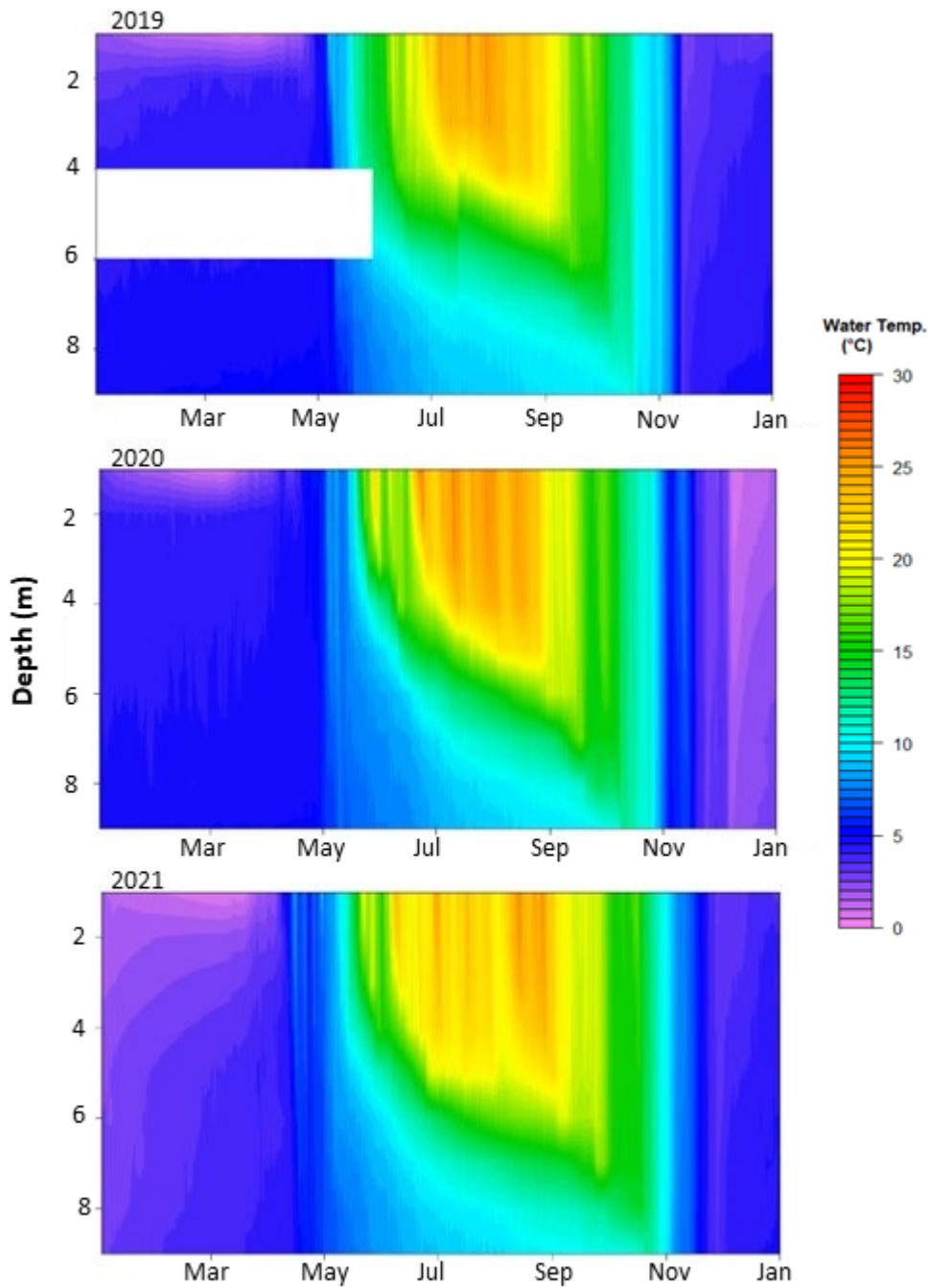


**Figure 3.** Monthly variability in surface water temperatures in Holland Pond. Boxplot colors refer to data from different years.

Monthly variability in surface water temperature as measured at 1m below the surface in Holland Pond experience relatively little variability in the winter and summer months, and more variability in the spring and fall (**Figure 3**). Interestingly, the sharpest increases in surface water warming occur in May and November. This is resulting in the longer periods of summer stratification depicted in the thermal heat maps.

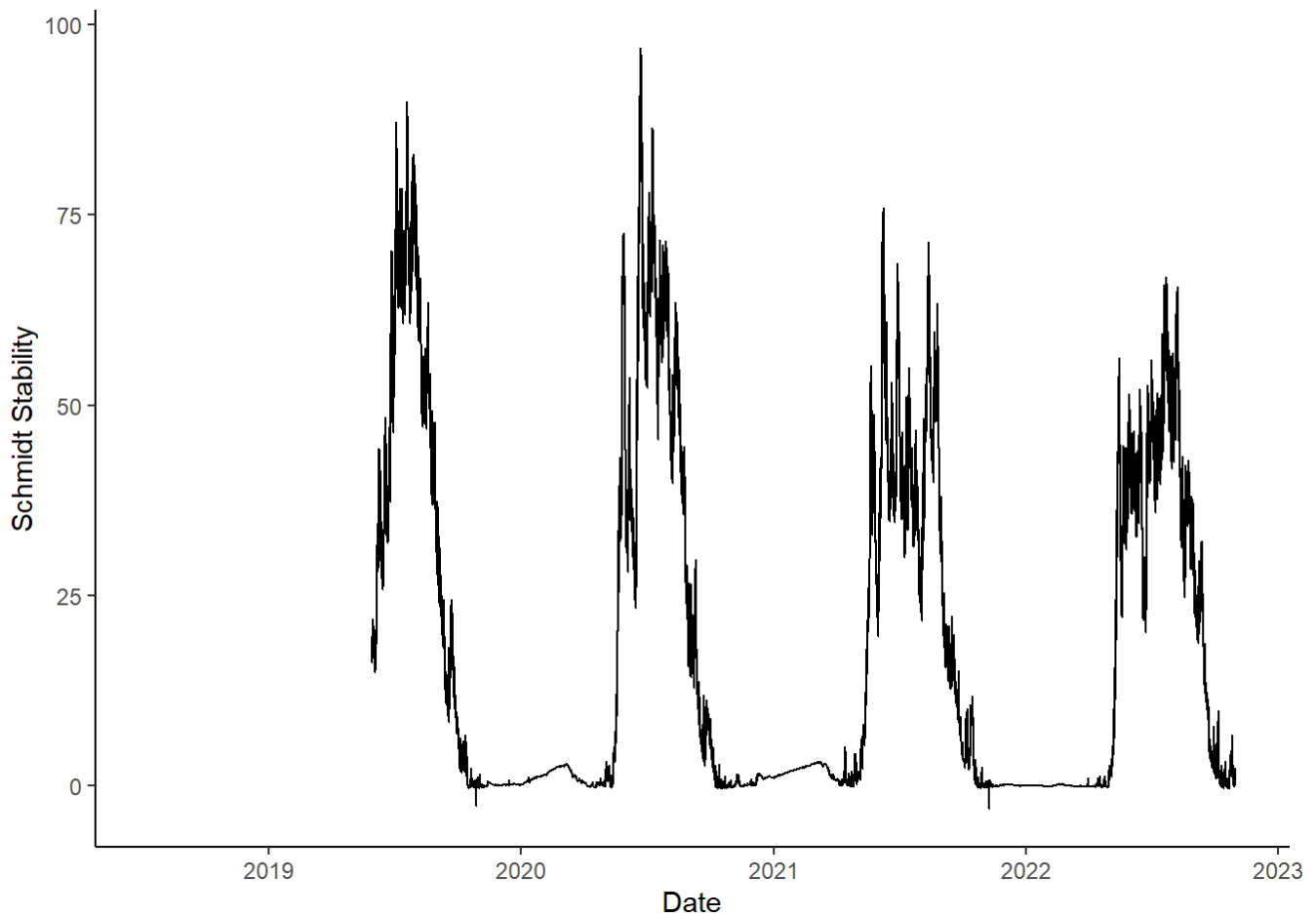
### 3.1.2 Long

Long Pond is also a dimictic lake that mixes fully once in the spring and once in the fall. Thermal profiles show that the lake is getting warmer in the spring and the fall, which leads to increased duration of the stratified season (**Figure 4**). Similar to what was observe in Holland Pond, spring turnover occurred in late April or early May, and fall turnover occurred in mid to late September, with a period of re-stratification around October.



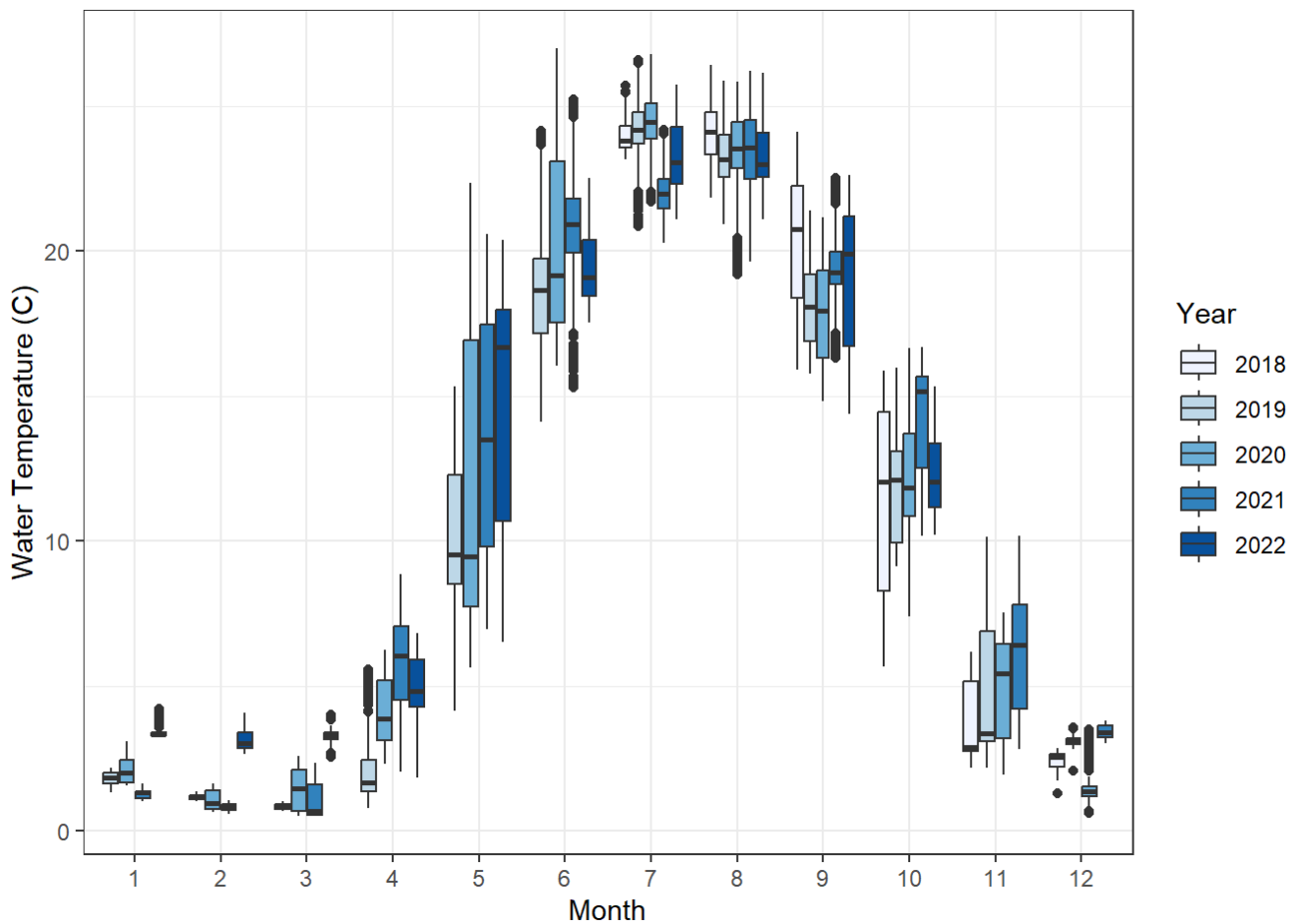
**Figure 4.** Thermal Heatmaps for the three full years of data collection in Long Pond. The white rectangle represents missing data at 5m.

A different pattern occurred with Schmidt Stability in Long Pond (**Figure 5**). The summer stability seems to be decreasing, but the period of stability is getting longer. This means that the maximum stratification appears weakened over the monitoring period, but the length of stratification increased. Note here that the schmidt stability could not be calculated for Long Pond for the summer of 2018 due to the missing values at 5m. This could be interpolated in future work to avoid this issue.



**Figure 5.** *Long Pond's Schmidt Stability over time.*

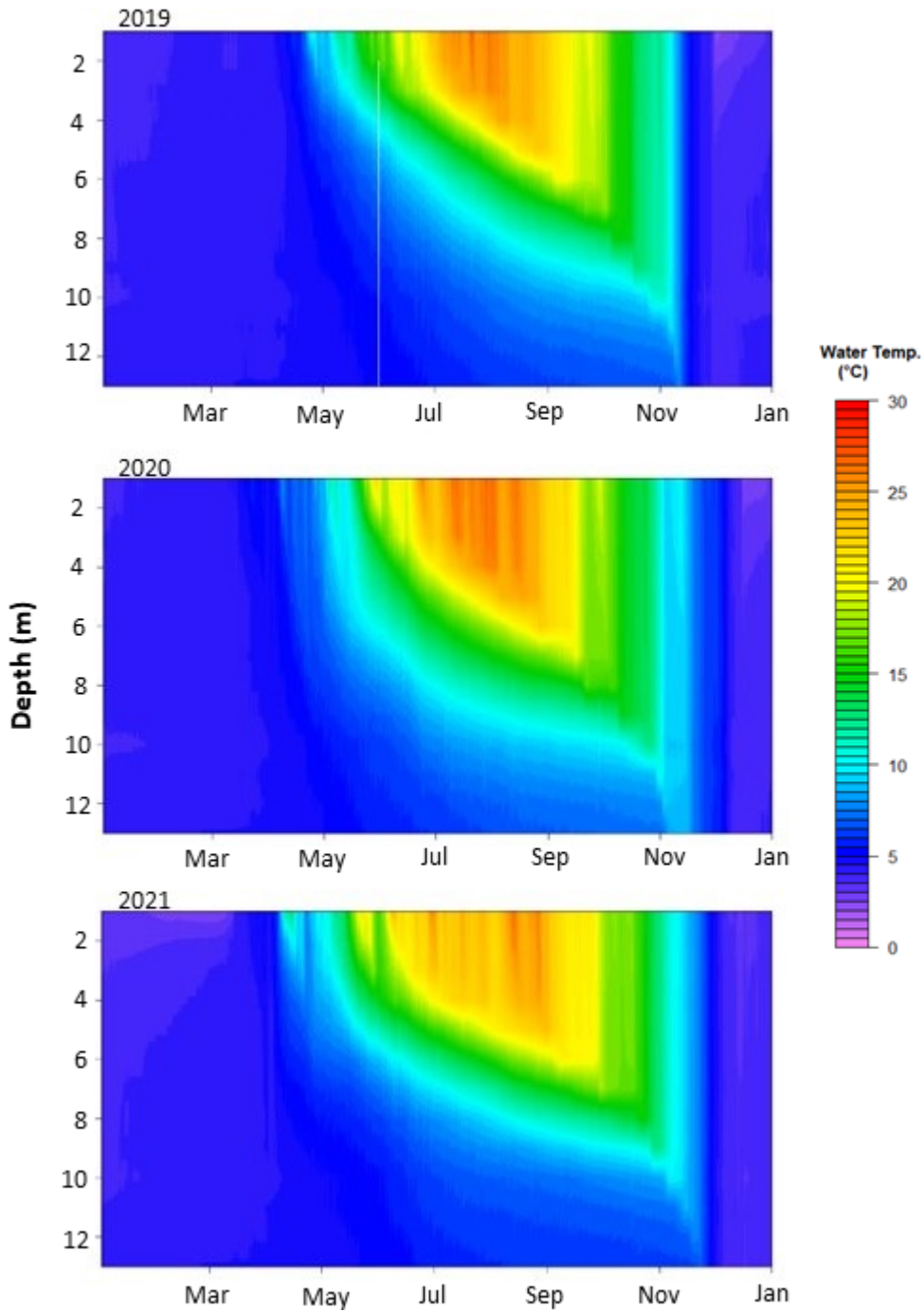
Monthly variability in surface water temperature in Long Pond are shown in **Figure 6**. The surface readings come from 0.75m below the surface. Similar to what was observed in Holland Pond, Long Pond's surface water temperatures experienced relatively little variability in the winter and summer months, and more variability in the spring and fall. Interestingly, just like in Holland Pond, the sharpest increases in surface water warming in Long Pond occurred in May and November. This resulted in the longer periods of summer stratification in the thermal heat maps.



**Figure 6.** Monthly variability in surface water temperatures in Long Pond. Boxplot colors refer to data from different years.

### 3.1.3 High

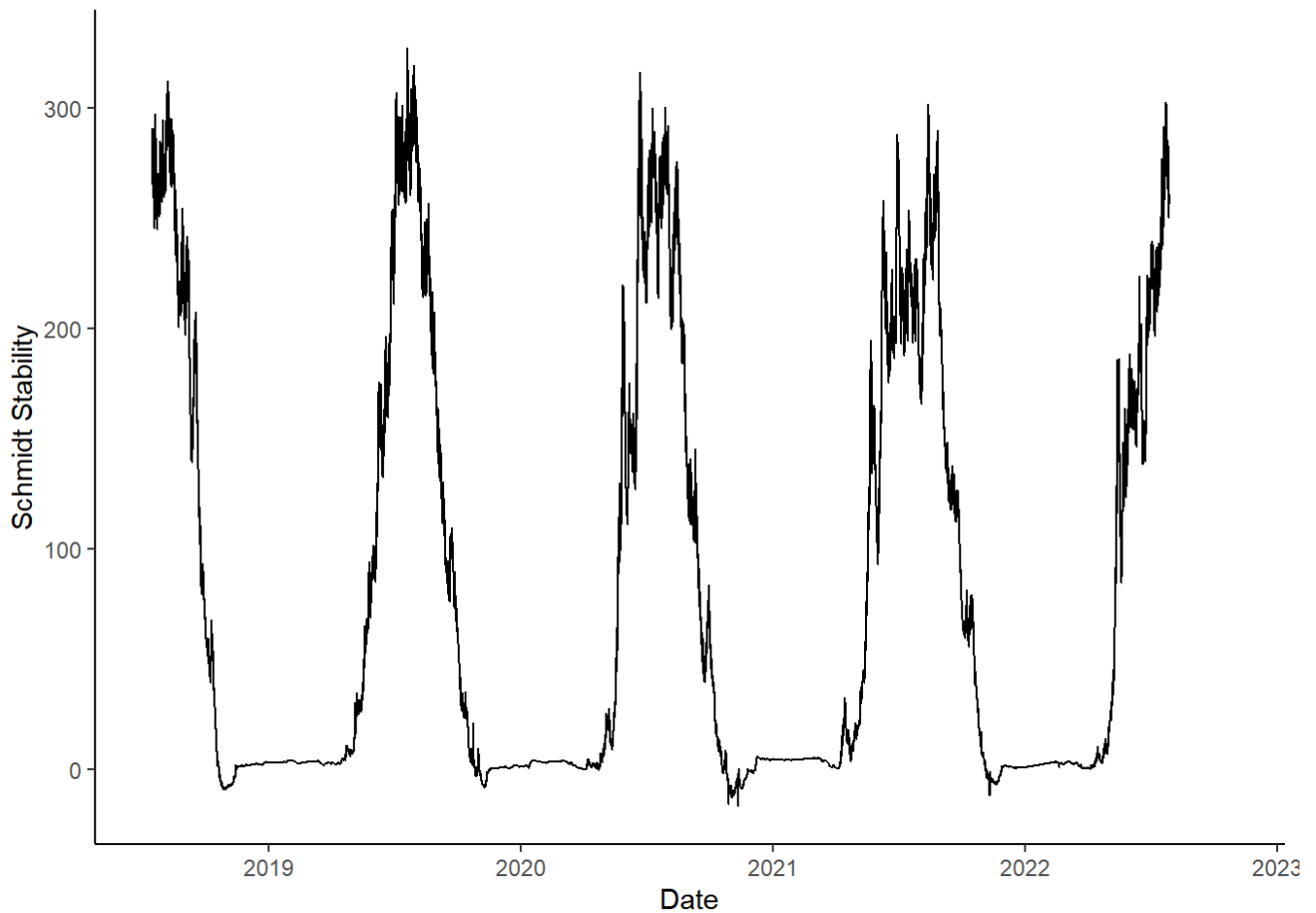
High Pond is also a dimictic lake that mixes fully once in the spring and once in the fall. High Pond is the deepest of out the three lakes, it is located at the lowest elevation, is generally more protected from wind, and is the most southern lake. As such, it exhibits a much longer period of stratification than the other two. The thermal profiles in High Pond show variability in the depth of heating, where it appears that in 2020 the water was warmer at depth than it was during the other two years (**Figure 7**). Similar to Holland and Long, High Pond is getting warmer in the spring and the fall, which led to increased duration of the stratified season. However, High Pond saw increasingly longer periods of partial stratification in the spring after ice off. Spring turnover occurred in April, but didn't shift later in the season as was evident in the other lakes. High Pond's water column did warm a lot faster after ice off. Fall turnover occurred much later in the season in High Pond, and didn't fully mix until November, and in some cases, even December. Often times, inverse stratification started to set in before the lake fully mixed.



**Figure 7.** Thermal Heatmaps for the three full years of data collection in High Pond. White lines refer to missing data.

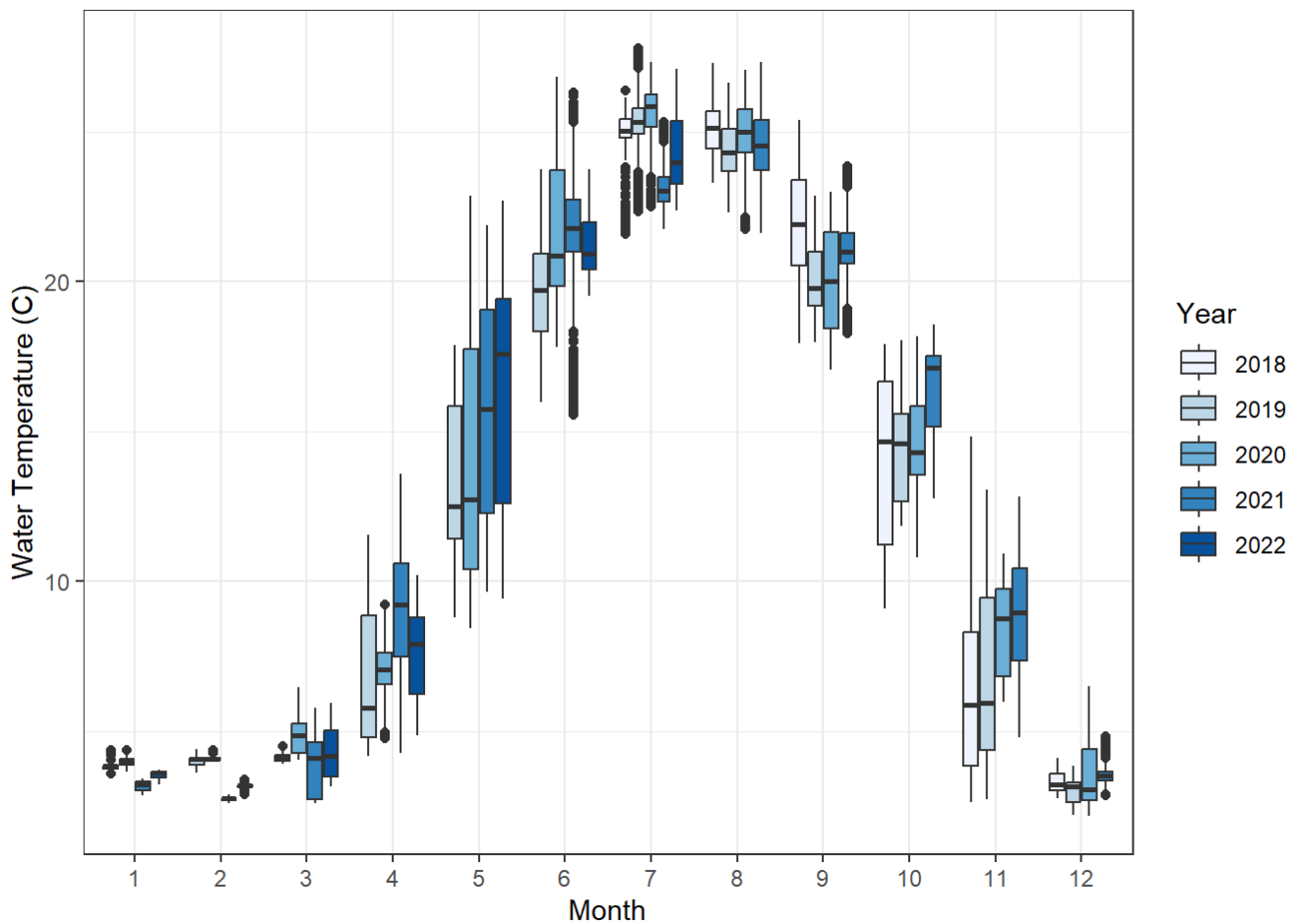
Comparatively, High Pond's water column was more stable over the time monitored, and the strength of stratification didn't fluctuate as much as it did on the other two lakes (**Figure 8**). However, in recent years, the period of stratification appears to be increasing over time, similar to what was observe on Holland and Long Ponds.





**Figure 8.** *High Pond's Schmidt Stability over time.*

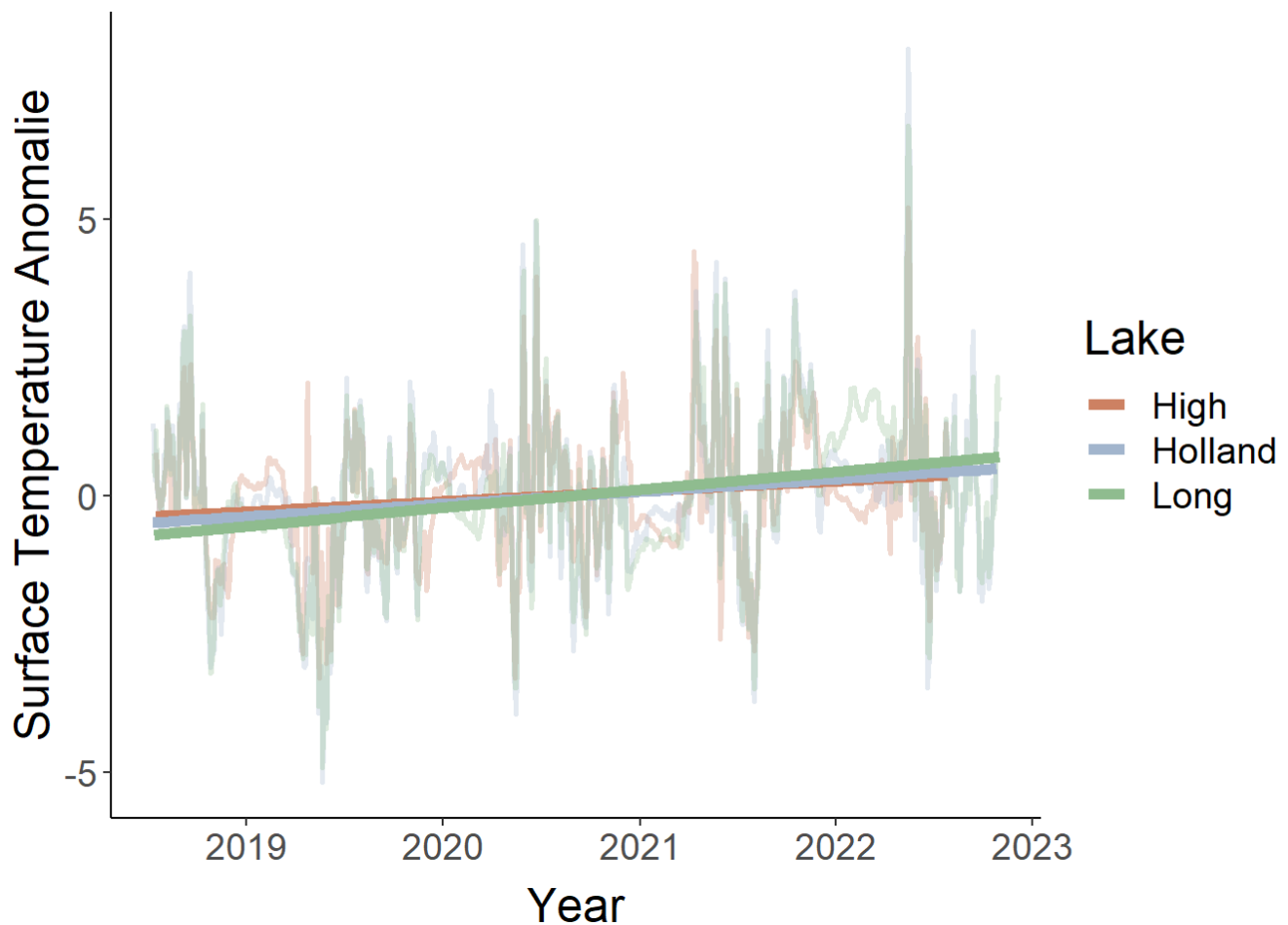
High Pond experiences the most stability in surface water temperatures during the summer and winter. The surface readings come from 1m below the surface (**Figure 9**). The most variability occurs in fall and spring. Additionally, surface water warming occurred the most in the same months as the other two lakes - May and November, accounting for those longer periods of stratification.



**Figure 9.** Monthly variability in surface water temperatures in High Pond. Boxplot colors refer to data from different years.

## 3.2 Insights from trends across all lakes

To understand how Vermont lakes are potentially being impacted from climate stressors, I wanted to look at how many days per year lakes remained stratified, how long their ice periods lasted, and compare the rate of surface water warming to air temperature warming.

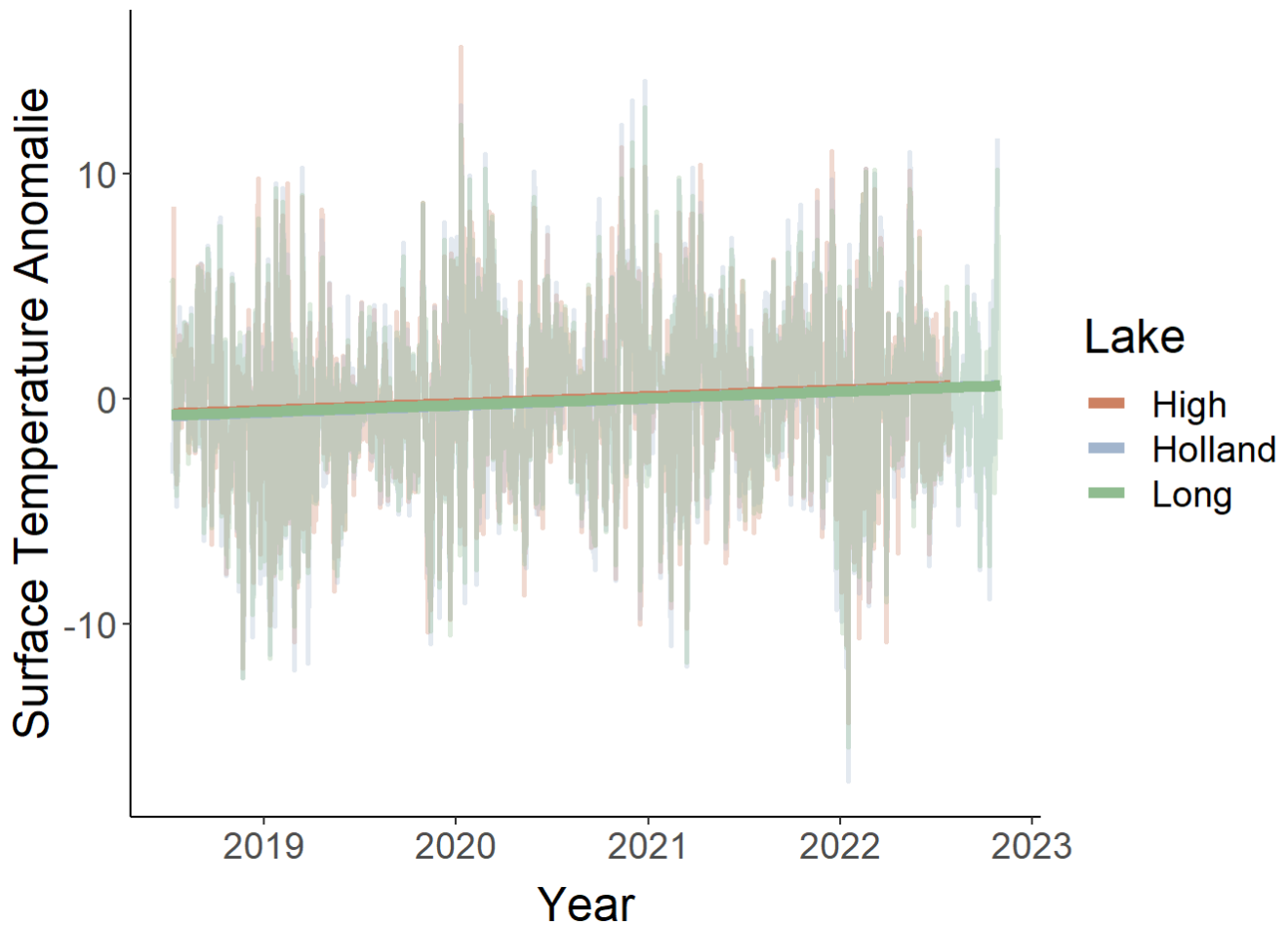


**Figure 10.** *Temperature anomalies for surface waters in all three lakes over the three years of continuous data collection.*

From looking at individual surface water temperatures through the thermal heat maps and surface water boxplots, we see that each lake in this study has been getting warmer over the past few years in both the spring and the fall. However, the surface temperature anomalies plot above shows how many degrees Celsius each day is compared to the 5 year average. After graphing all three lakes together and applying a simple linear model, on average, all of the lakes have experienced warming surface waters over the time monitored (**Figure 10**).

Also note here that Holland and Long Pond, which are located in the northeastern portion of Vermont have very similar trends, and High Pond, which is more southern, has a less pronounced slope. This means that there may be some latitudinal relationship between warming in our lakes in Vermont.

### 3.2.1 Air Temperature Trends

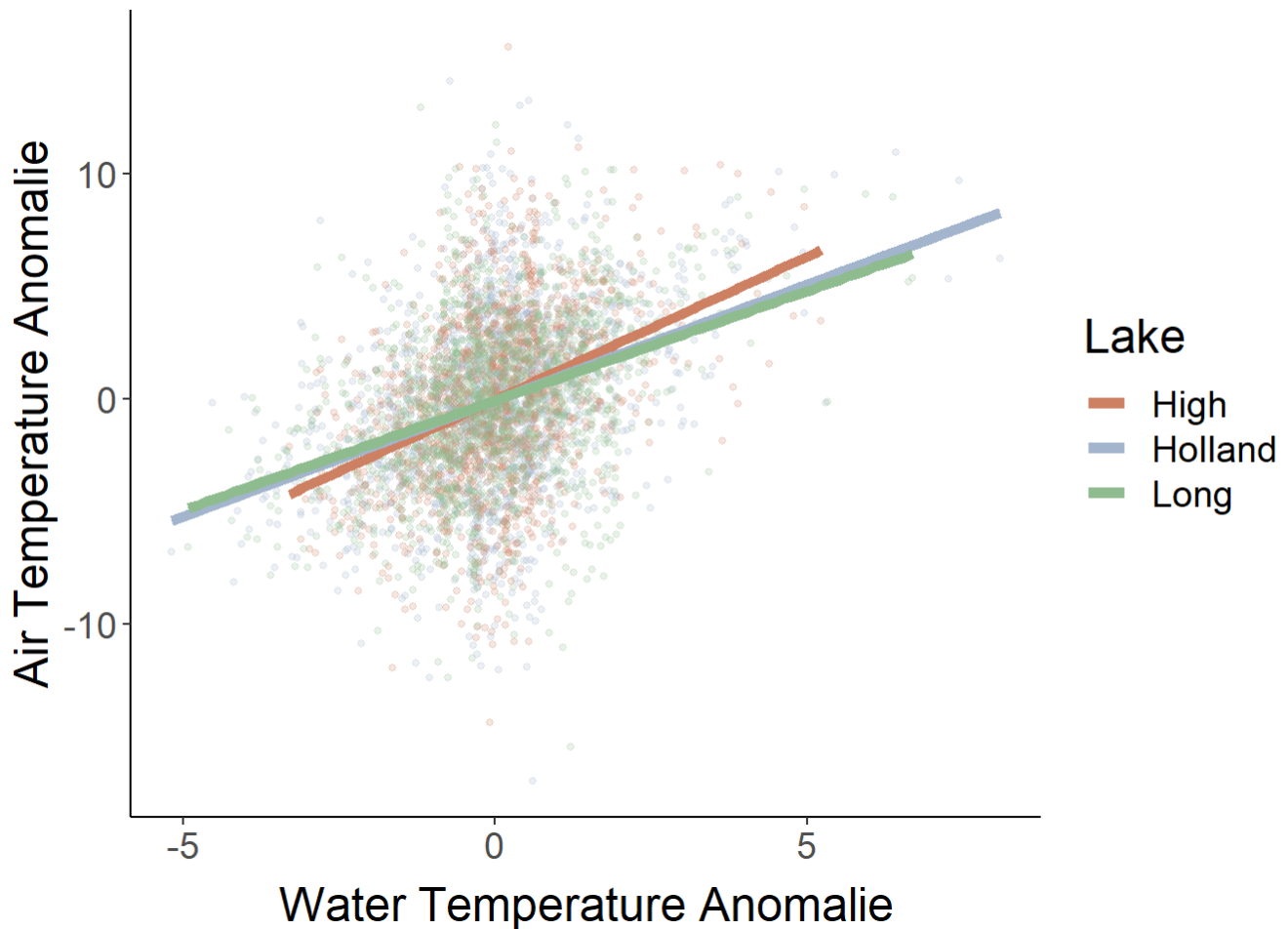


**Figure 11.** *Temperature anomalies in air temperature at all three lakes. Solid lines represent linear trends.*

Because there is only 3 full consecutive years of water temperature data collected on these lakes in Vermont, we can't necessarily conclude climate change is the primary driver of the increase we are seeing. However, there are longer term air temperature data sets from Vermont. By exploring trends in air temperature and surface water temperature, we can get an idea as to how well trends in water temperature track with air temperatures over time, and can infer the impact of climate change on lake water temperatures.

Here we can see that just like the surface waters in all of the lakes, after applying a linear model, the air temperatures are also increasing over time (**Figure 11**). However, there is much more noise in the measured air temperature data as compared to the water temperature data, as would be expected.

### 3.2.2 Air Temperature vs. Surface Water Temperature.

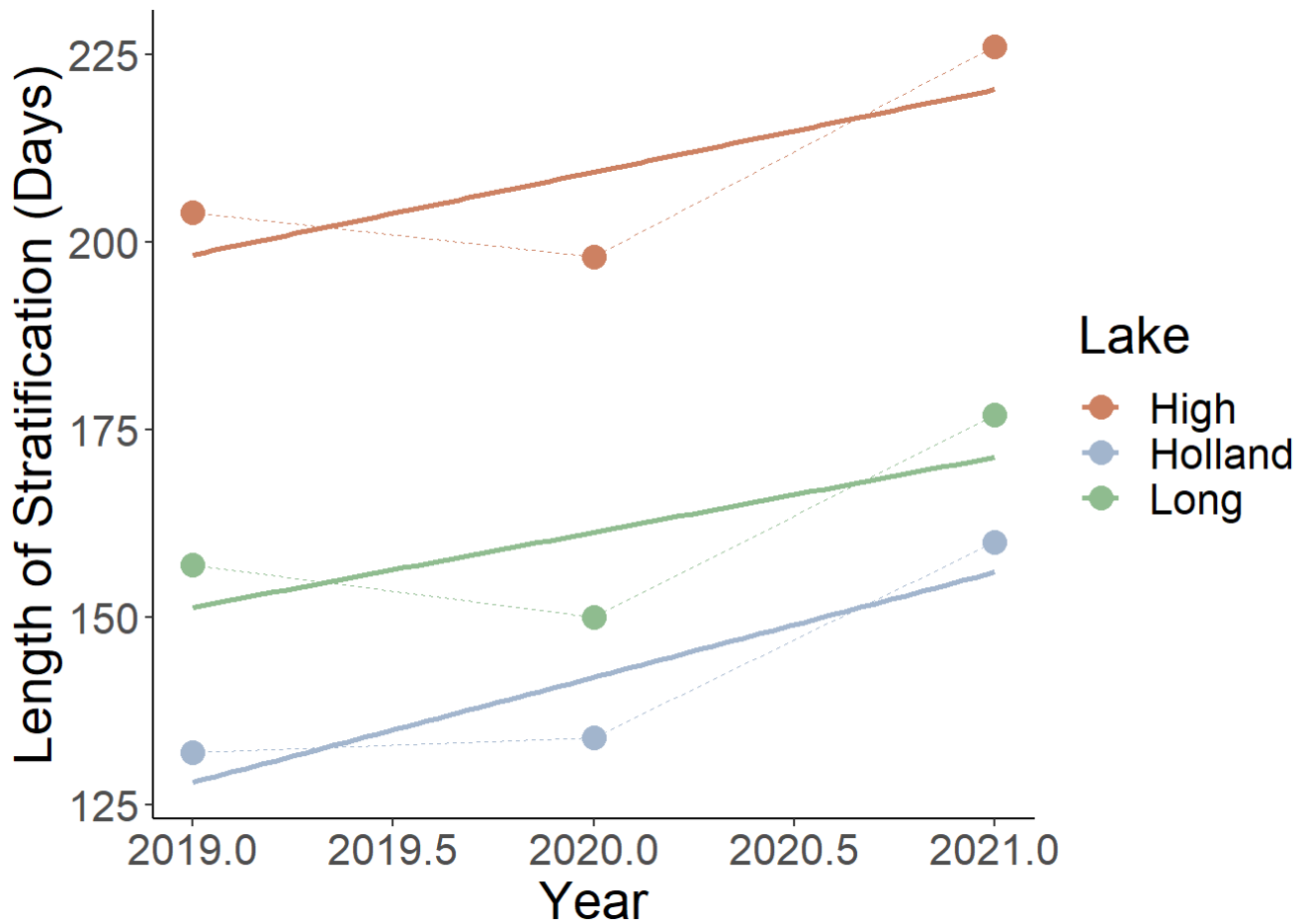


**Figure 12.** Relationship between air temperature anomalies and water temperature anomalies at all three lakes. Solid lines represent linear trends.

After running a linear model on the relationship between air temperature and surface temperatures (**Figure 12**), there is a positive relationship, suggesting that if air temperatures continue to increase as a result of climate change, surface water temperatures are likely to follow suit.



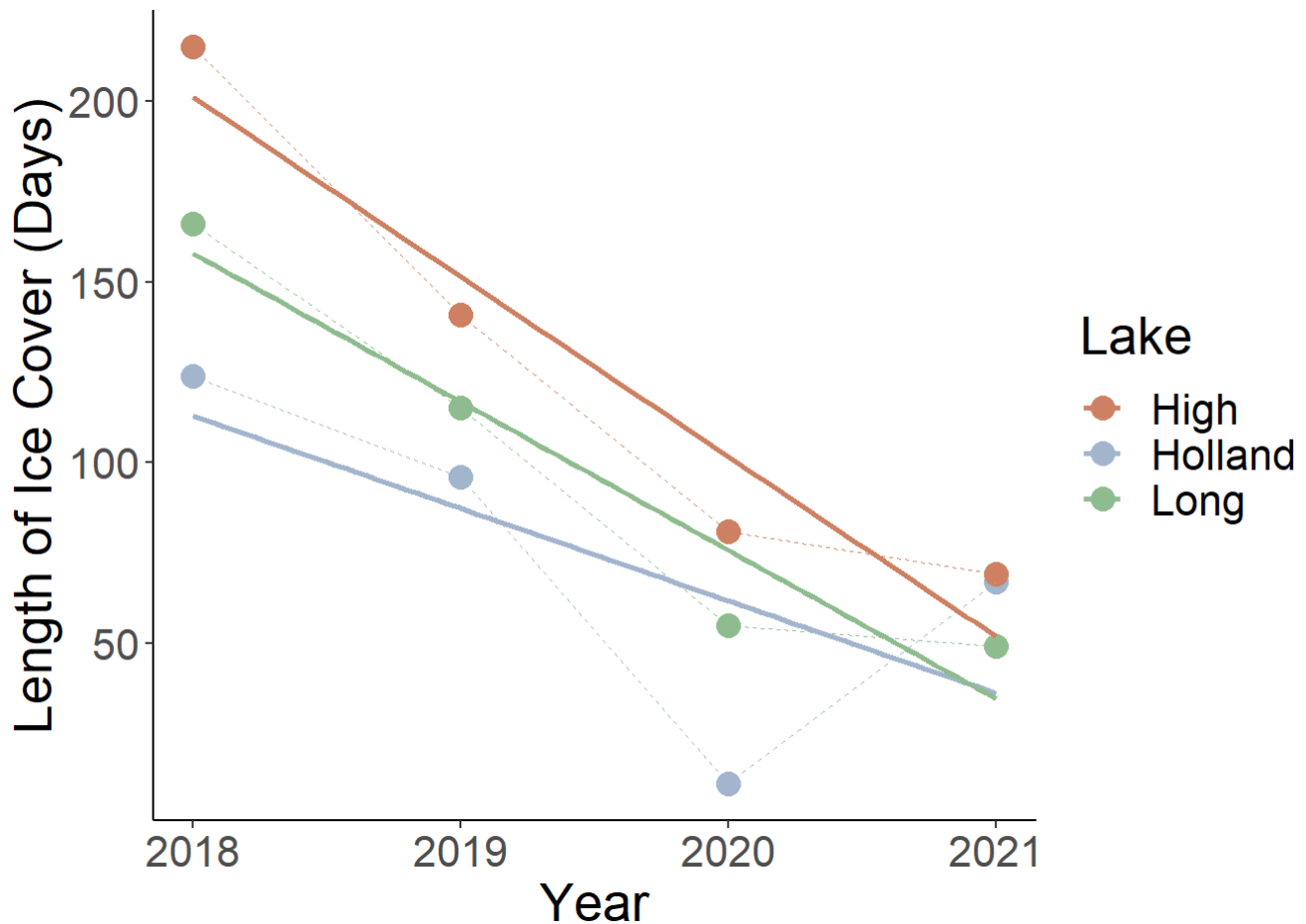
### 3.2.3 Length of Stratification



**Figure 13.** Length of summer stratification in each lake over time for each year where there was full season of collected data. Solid lines represent linear trends.

All three lakes experienced similar patterns in length of summer stratification over time (**Figure 13**). High Pond, which is the deepest lake and the most southern lake, experiences longer periods of stratification than the other two lakes. Both High Pond and Long Pond experienced a decrease in the stratification period in 2020, while Holland had a longer period. All lakes had a sharp increase in stratification days from 2020 to 2021.

### 3.2.4 Length of the Ice Period

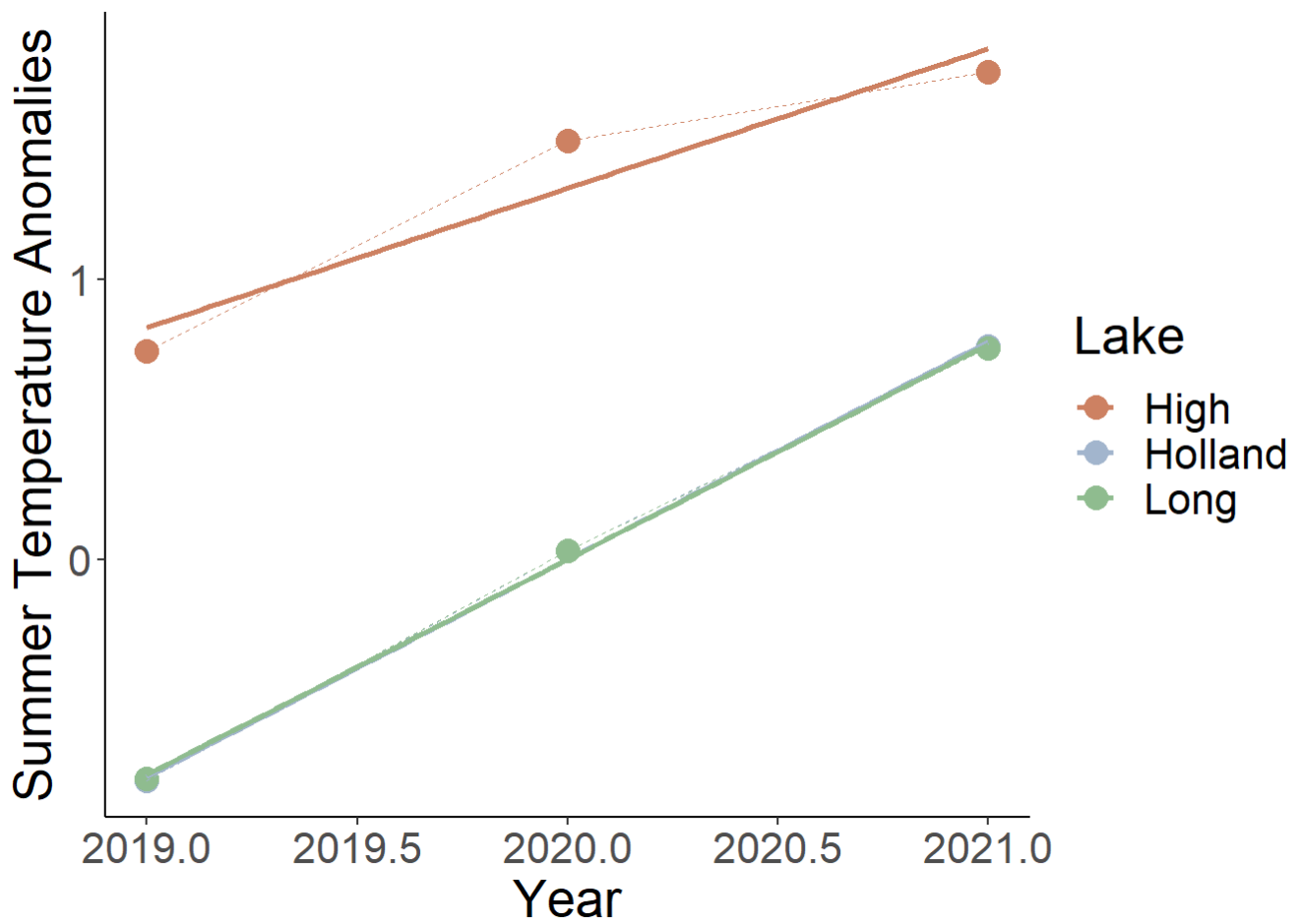


**Figure 14.** Length of ice cover in each lake where there was a full sampling period for the winter season. Ice cover was determined by counting days where inverse stratification occurred (at least 1 degree Celcius difference between 1 meter). Solid lines represent linear trends.

When looking at the length of ice cover, or the duration of time where there is inverse winter stratification, all of the lakes displayed similar patterns, with the exception of Holland Pond, which had a longer duration of inverse stratification this past winter (**Figure 14**). In general, it appears that the length of ice cover decreased in all of the lakes over the period monitored.

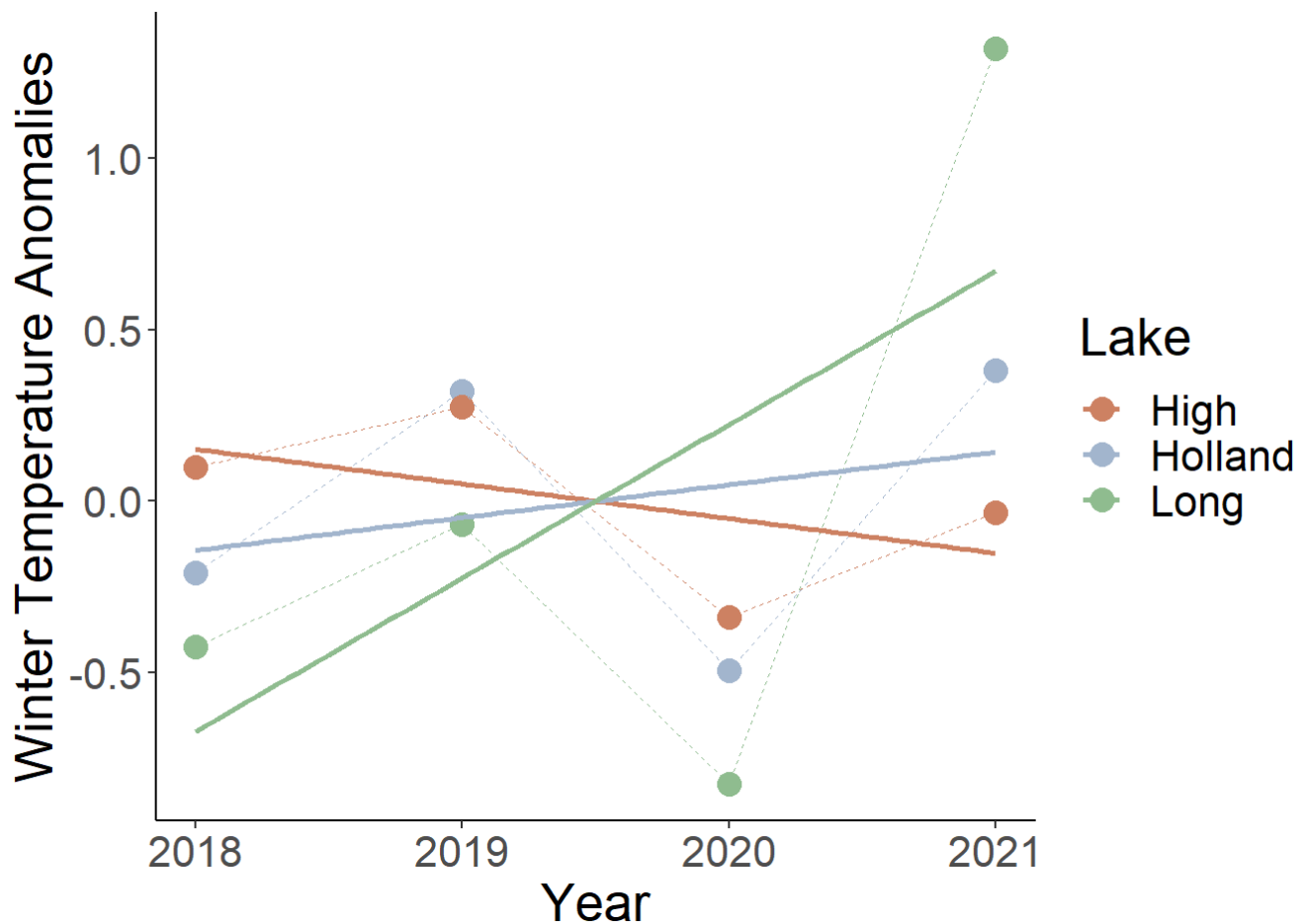
### 3.2.5 Surface Temperature in Winter vs Summer.

Because there is seasonality in the rate of warming in Vermont Lakes, I wanted to look at the rates of warming in the summer vs. the winter periods.



**Figure 15.** Summer season surface water temperature anomalies for all three lakes over time. Solid lines represent linear trends.

Holland and Long have nearly the exact rates of warming in the summer period, while High Pond, which is southern, is warming more in the summer than the other two lakes (**Figure 15**).



**Figure 16.** Winter season surface water temperature anomalies over time for each lake. Solid lines represent linear trends.

Winter warming periods are more variable across the three lakes than the summer (**Figure 16**). However, Long Pond experiences the largest variability in winter temperature compared to High Pond and Holland Pond. While there is only four full winter periods measured, we can utilize the trends in surface water temperatures as signals of climate change. Specifically, although the degrees of warming and cooling may differ from lake to lake, each lake experiences the same trends in temperature anomalies over time in the winter period.

## 4 Conclusions

After running these analyses on these three minimally disturbed lakes in Vermont, there is evidence that the surface waters of these lakes are warming in conjunction with rising air temperatures. Surface waters are warming the most in May and November, which creates longer periods of stratification, and shorter winter periods. With more data gathered on temperature dynamics and mixing regimes in Vermont lakes over time, we can start to parse out whether or not this trend is a result of climate change, or if it is occurring as a result of natural variation. There is evidence of synchrony in thermal patterns across lakes in the state of Vermont, which suggest we can use water temperatures as signals for climate change. Additionally, by pooling together data from other lakes in the Lake Regional Monitoring Network, we can better understand how lakes in the region are responding to climate change.

## 5 Bibliography

Adrian, R., O'reilly, C. M., Zagarese, H., Baines, S. B., Hessen, D. O., Keller, W., Livingstone, D. M., Sommaruga, R., Straile, D., Donk, E. van, Weyhenmeyer, G. A., Winder, M., & Muir, J. (2009). Lakes as sentinels of climate change. *Limnology and Oceanography*, 54(6), 2283–2297.

(<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2854826/>)

Christensen, M. R., Graham, M. D., Vinebrooke, R. D., Findlay, D. L., Paterson, M. J., & Turner, M. A. (2006). Multiple anthropogenic stressors cause ecological surprises in boreal lakes. *Global Change Biology*, 12(12), 2316–2322. (<https://doi.org/10.1111/j.1365-2486.2006.01257.x>)

Downing, J. A. (2008). Emerging global role of small lakes and ponds: little things mean a lot. *Limnetica*, 29(1), 9–24. Folt, C. L., Chen, C. Y., Moore, M. v., & Burnaford, J. (1999). Synergism and antagonism among multiple stressors. *Limnology and Oceanography*, 44(3 II), 864–877 ([https://doi.org/10.4319/LO.1999.44.3\\_PART\\_2.0864](https://doi.org/10.4319/LO.1999.44.3_PART_2.0864))

Folt, C. L., Chen, C. Y., Moore, M. v., & Burnaford, J. (1999). Synergism and antagonism among multiple stressors. *Limnology and Oceanography*, 44(3 II), 864–877. ([https://doi.org/10.4319/LO.1999.44.3\\_PART\\_2.0864](https://doi.org/10.4319/LO.1999.44.3_PART_2.0864))

O'Reilly, C. M., Sharma, S., Gray, D. K., Hampton, S. E., Read, J. S., Rowley, R. J., Schneider, P., Lenters, J. D., McIntyre, P. B., Kraemer, B. M., Weyhenmeyer, G. A., Straile, D., Dong, B., Adrian, R., Allan, M. G., Anneville, O., Arvola, L., Austin, J., Bailey, J. L., ... Zhang, G. (2015). Rapid and highly variable warming of lake surface waters around the globe. *Geophysical Research Letters*, 42(24), 10,773–10,781. (<https://doi.org/10.1002/2015GL066235>)

Richardson, D. C., Melles, S. J., Pilla, R. M., Hetherington, A. L., Knoll, L. B., Williamson, C. E., Kraemer, B. M., Jackson, J. R., Long, E. C., Moore, K., Rudstam, L. G., Rusak, J. A., Saros, J. E., Sharma, S., Strock, K. E., Weathers, K. C., & Wigdahl-Perry, C. R. (2017). Transparency, Geomorphology and Mixing Regime Explain Variability in Trends in Lake Temperature and Stratification across Northeastern North America (1975–2014). *Water* 2017, Vol. 9, Page 442, 9(6), 442. (<https://doi.org/10.3390/W9060442>)

Stamp, Jen. (2018) Regional Monitoring Network for Lakes. *Lakeline Magazine*. (<https://www.nalms.org/wp-content/uploads/2018/09/38-2-8.pdf>)

Wilkinson, G. M., Walter, J., Fleck, R., & Pace, M. L. (2020). Beyond the trends: The need to understand multiannual dynamics in aquatic ecosystems. *Limnology and Oceanography Letters*, 5(4), 281–286. (<https://doi.org/10.1002/lol2.10153>)

Winslow, L. A., Read, J. S., Hansen, G. J. A., & Hanson, P. C. (2015). Small lakes show muted climate change signal in deepwater temperatures. *Geophysical Research Letters*, 42(2), 355–361. (<https://doi.org/10.1002/2014GL062325>)

## 6 Code Guide

This guide will go over the step-by-step data analysis approach I used for each lake, and how I combined all of the lakes in order to assess trends in lakes across Vermont.

### 6.1 Quality Control and Assurance in R

After taking the .csv files from the hoboware software and running them through the Continuous Data QC Shiny App (<https://tetrattech-wtr-wne.shinyapps.io/ContDataQC/>), further visualization and QC metrics were required before analyzing the data. Along with this step-by-step coding guide, I provided a powerpoint that walks you through the metrics I took with the QC process using the shiny app developed by tetra tech.



## 6.1.1 Libraries and Loading Data

For this analysis, I used four primary packages:

```
library(rLakeAnalyzer)
library(tidyverse)
library(lubridate)
library(readxl)
library(DT)
```

In the file folder format, the aggregated data will be found in the Data3\_Aggregate Folder.

```
setwd("C:/Users/kwarn/Desktop/Sentinel_Lakes")

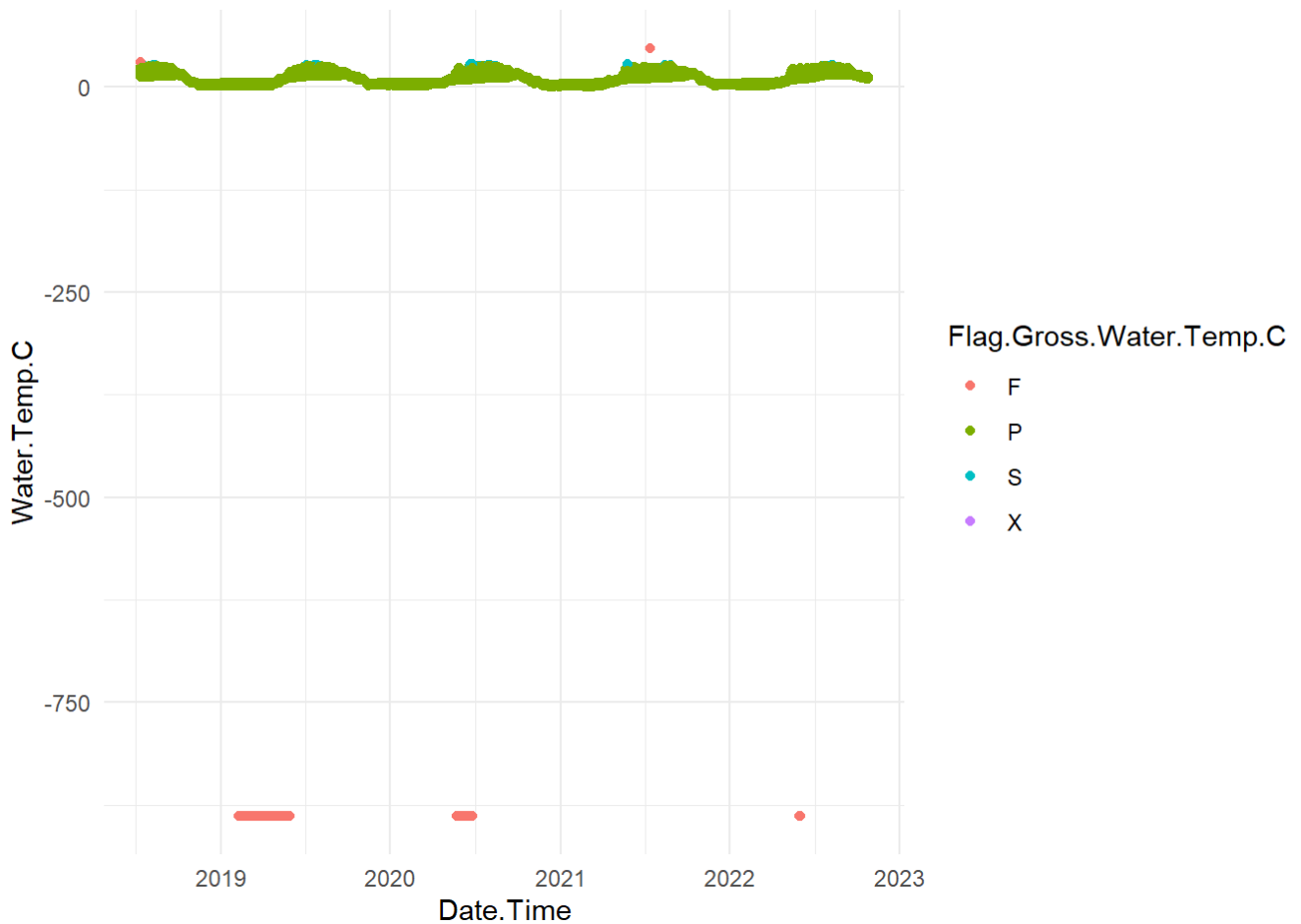
Data <- read_csv("./Holland/Data3_Aggregate/DATA_QC_HOLLA--01M_Water_00626_91024.csv")
```

As you can see in this dataframe there are many parameters and several different flagged QC columns. The next bit of code will be an example of how I went through and ran QC checks on the data.

## 6.1.2 Quality Control for Water Temperature

```
Data %>%
  ggplot(aes(x=Date.Time, y=Water.Temp.C, color=Flag.Gross.Water.Temp.C)) +
  geom_point() +
  theme_minimal()
```

```
## Warning: Removed 9 rows containing missing values (`geom_point()`).
```

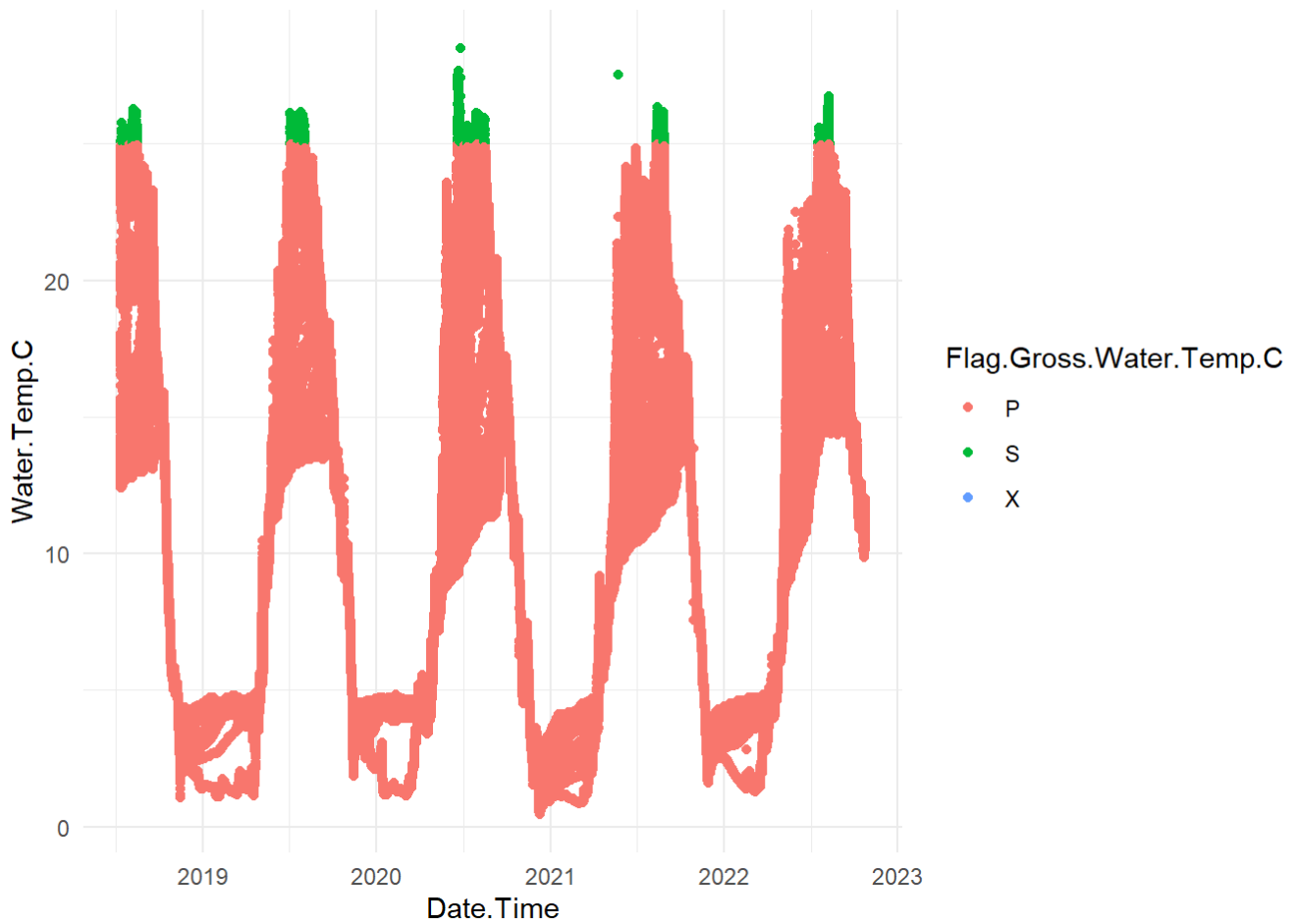


When looking at the temperature values, we can see that one of the arrays must have malfunctioned, so I will manually remove those values by filtering out the failed flags for the Gross Water Temperature.

```
Data <- Data %>%
  filter(Flag.Gross.Water.Temp.C != "F")

Data %>%
  ggplot(aes(x=Date.Time, y=Water.Temp.C, color=Flag.Gross.Water.Temp.C)) +
  geom_point() +
  theme_minimal()
```

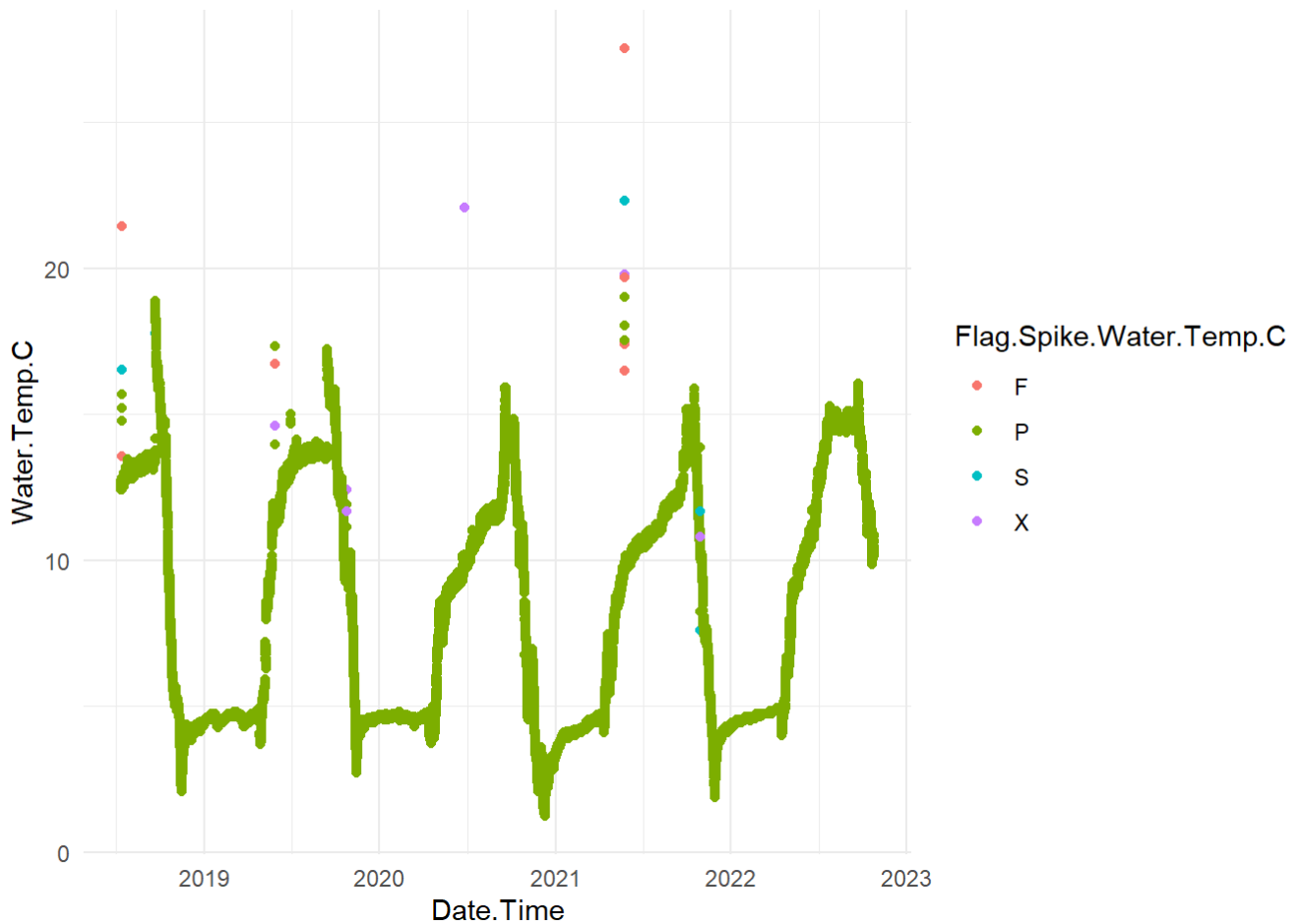
```
## Warning: Removed 9 rows containing missing values (`geom_point()`).
```



Next I will look at the spike test.

```
Data %>%
  filter(SiteID == "HOLLA--10M") %>%
  ggplot(aes(x=Date.Time, y=Water.Temp.C, color=Flag.Spike.Water.Temp.C)) +
  geom_point() +
  theme_minimal()
```

```
## Warning: Removed 9 rows containing missing values (`geom_point()`).
```



This is where I think Tim Martin's QC method would be better because you can manually choose points and change their flag value. I went through and removed them using a string of filtering and mutating, which is signified in the code below, but it was time intensive. You also might be able to change the results via changing spike thresholds or having a list of days where sampling events occurred.

Briefly, I went through and identified the days where the buoys were pulled and visualized each of those days at each depth. I then created the chunk of code below to remove the values that were incorrect.

```

Data <- Data %>%
  mutate(Remove = if_else(Date == "2018-07-13" & Water.Temp.C > 13 &
    SiteID == "HOLLA--10M", "Yes", "No"),
    Remove = if_else(Date == "2018-07-13" & Water.Temp.C > 13 &
    SiteID == "HOLLA--09M", "Yes", Remove),
    Remove = if_else(Date == "2018-07-13" & Water.Temp.C > 15 &
    SiteID == "HOLLA--08M", "Yes", Remove),
    Remove = if_else(Date == "2018-07-13" & Water.Temp.C > 13 &
    SiteID == "HOLLA--07M", "Yes", Remove),
    Remove = if_else(Date == "2018-07-13" & Water.Temp.C > 13 &
    SiteID == "HOLLA--07M", "Yes", Remove),
    Remove = if_else(Date == "2018-07-13" & Water.Temp.C > 20 &
    SiteID == "HOLLA--06M", "Yes", Remove),
    Remove = if_else(Date == "2019-05-29" & Water.Temp.C > 13 &
    SiteID == "HOLLA--10M", "Yes", Remove),
    Remove = if_else(Date == "2019-05-29" & Water.Temp.C > 13 &
    SiteID == "HOLLA--09M", "Yes", Remove),
    Remove = if_else(Date == "2019-05-29" & Water.Temp.C > 15 &
    SiteID == "HOLLA--08M", "Yes", Remove),
    Remove = if_else(Date == "2019-05-29" & Water.Temp.C > 13 &
    SiteID == "HOLLA--07M", "Yes", Remove),
    Remove = if_else(Date == "2019-05-29" & Water.Temp.C > 15 &
    SiteID == "HOLLA--06M", "Yes", Remove),
    Remove = if_else(Date == "2019-06-29" & Water.Temp.C > 13 &
    SiteID=="HOLLA--10M", "Yes", Remove),
    Remove = if_else(Date == "2019-06-29" & Water.Temp.C > 15 &
    SiteID=="HOLLA--09M", "Yes", Remove),
    Remove = if_else(Date == "2019-06-29" & Water.Temp.C > 15 &
    SiteID=="HOLLA--08M", "Yes", Remove),
    Remove = if_else(Date == "2019-06-29" & Water.Temp.C > 13 &
    SiteID=="HOLLA--07M", "Yes", Remove),
    Remove = if_else(Date == "2019-06-29" & Water.Temp.C > 15 &
    SiteID=="HOLLA--06M", "Yes", Remove),
    Remove = if_else(Date == "2020-06-26" & Water.Temp.C > 13 &
    SiteID=="HOLLA--10M", "Yes", Remove),
    Remove = if_else(Date == "2020-06-26" & Water.Temp.C > 13 &
    SiteID=="HOLLA--09M", "Yes", Remove),
    Remove = if_else(Date == "2020-06-26" & Water.Temp.C > 13 &
    SiteID=="HOLLA--08M", "Yes", Remove),
    Remove = if_else(Date == "2020-06-26" & Water.Temp.C > 13 &
    SiteID=="HOLLA--07M", "Yes", Remove),
    Remove = if_else(Date == "2020-06-26" & Water.Temp.C > 15 &
    SiteID=="HOLLA--06M", "Yes", Remove),
    Remove = if_else(Date == "2020-06-26" & Water.Temp.C > 20 &
    SiteID=="HOLLA--05M", "Yes", Remove),
    Remove = if_else(Date == "2020-06-26" & Water.Temp.C > 25 &
    SiteID=="HOLLA--04M", "Yes", Remove),
    Remove = if_else(Date == "2021-05-24" & Water.Temp.C > 15&
    SiteID=="HOLLA--10M", "Yes", Remove),
    Remove = if_else(Date == "2021-05-24" & Water.Temp.C > 13 &
    SiteID=="HOLLA--09M", "Yes", Remove),
    Remove = if_else(Date == "2021-05-24" & Water.Temp.C > 15 &

```

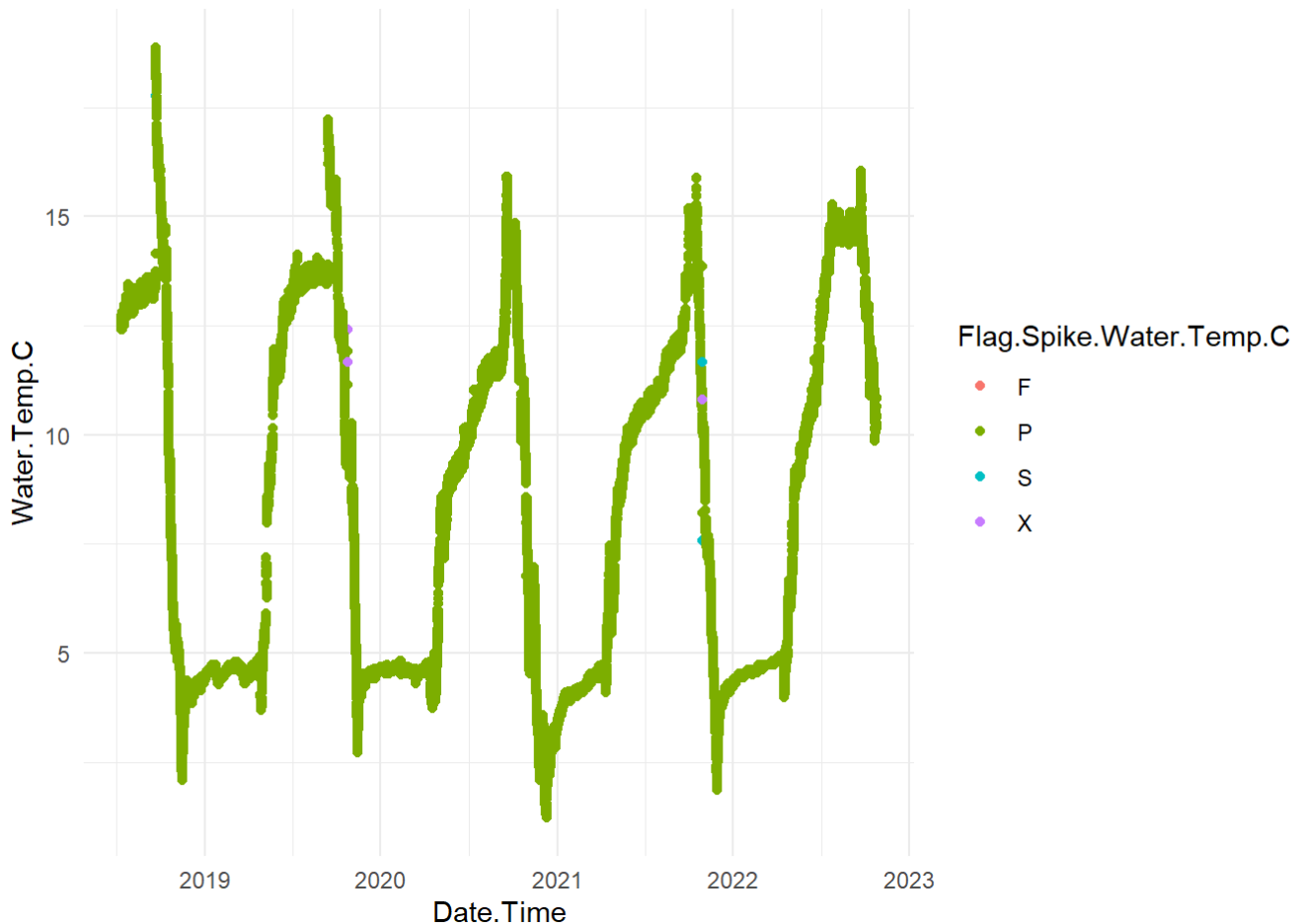
```

SiteID=="HOLLA--08M", "Yes", Remove),
Remove = if_else(Date == "2021-05-24" & Water.Temp.C > 13 &
  SiteID=="HOLLA--07M", "Yes", Remove),
Remove = if_else(Date == "2021-05-24" & Water.Temp.C > 15 &
  SiteID=="HOLLA--06M", "Yes", Remove),
Remove = if_else(Date == "2022-05-31" & Water.Temp.C > 13 &
  SiteID=="HOLLA--10M", "Yes", Remove),
Remove = if_else(Date == "2022-05-31" & Water.Temp.C > 13 &
  SiteID=="HOLLA--09M", "Yes", Remove),
Remove = if_else(Date == "2022-05-31" & Water.Temp.C > 17 &
  SiteID=="HOLLA--08M", "Yes", Remove),
Remove = if_else(Date == "2022-05-31" & Water.Temp.C > 13 &
  SiteID=="HOLLA--07M", "Yes", Remove),
Remove = if_else(Date == "2022-05-31" & Water.Temp.C > 15 &
  SiteID=="HOLLA--06M", "Yes", Remove),
Remove = if_else(Date == "2022-05-31" & Water.Temp.C > 22 &
  SiteID=="HOLLA--04M", "Yes", Remove)) %>%
filter(Remove == "No")

```

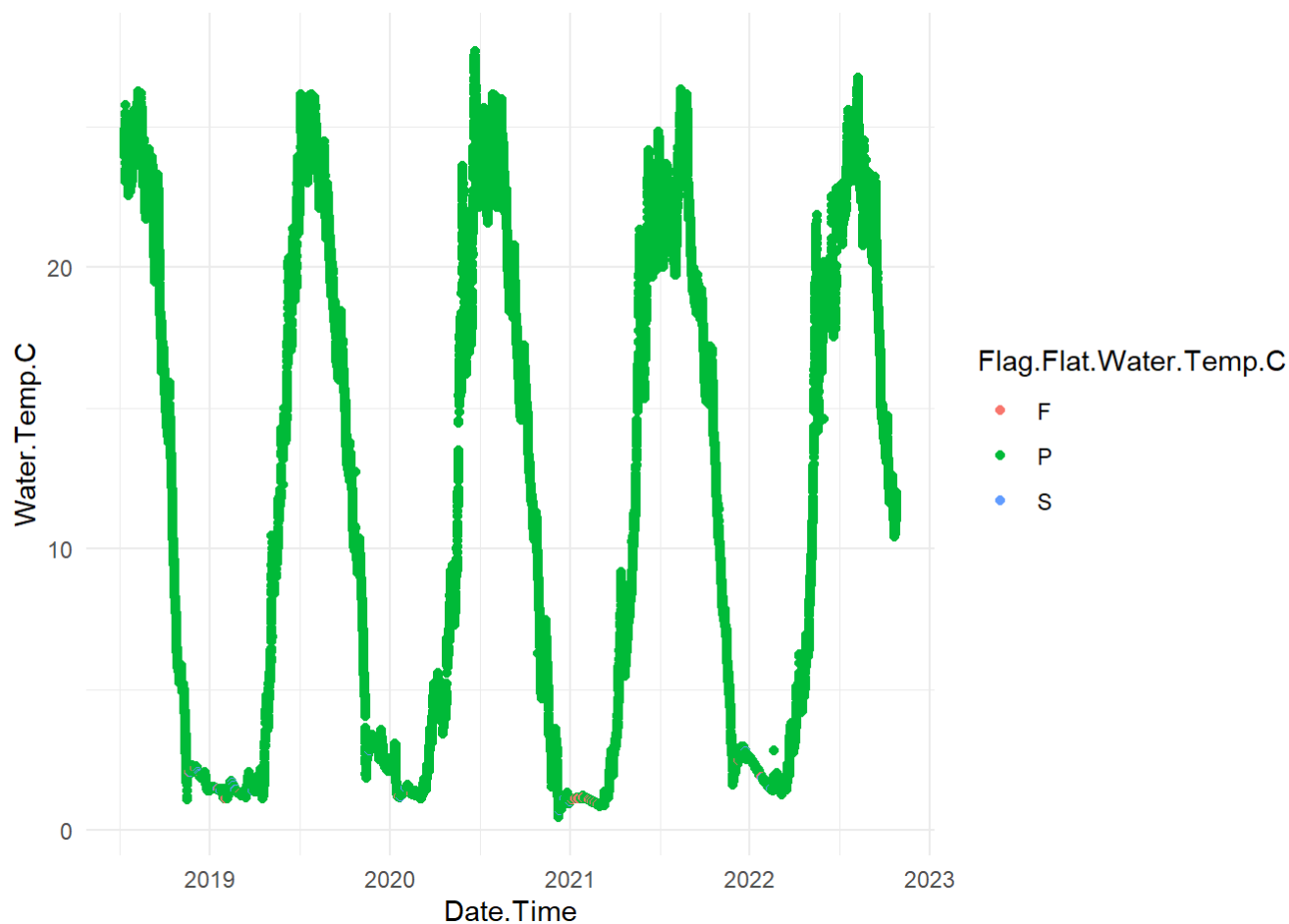
The figure below is the 10m mark after I cleaned up the spike test flags.

```
## Warning: Removed 4 rows containing missing values (`geom_point()`).
```



## 6.1.3 Flat Line Test

```
Data %>%
  filter(SiteID == "HOLLA--01M") %>%
  ggplot(aes(x=Date.Time, y=Water.Temp.C, color=Flag.Flat.Water.Temp.C)) +
  geom_point() +
  theme_minimal()
```



Notice that there are no evident flat line flags in the 1m. But it is good to check through each layer in the lake and remove any that may not be a result of ice cover.

## 6.2 Thermal Profiles

### 6.2.1 Creating a matrix for rlakeanalyzer format

rlakeanalyzer has many useful functions for exploring thermal mixing regimes. To explore the different thermal structures of each lake, we will use rlakeanalyzer's heatmap function.

However, it requires you to have a matrix with a specific format.

*(Explain section of format)*

```
Data_1 <- Data %>%  
  filter(SiteID == "HOLLA--01M") %>%  
  mutate(datetime = Date.Time) %>%  
  mutate(wtr_1.0 = Water.Temp.C) %>%  
  select(datetime, wtr_1.0)
```

```
Data_2 <- Data %>%  
  filter(SiteID == "HOLLA--02M") %>%  
  mutate(datetime = Date.Time) %>%  
  mutate(wtr_2.0 = Water.Temp.C) %>%  
  select(datetime, wtr_2.0)
```

```
Data_3 <- Data %>%  
  filter(SiteID == "HOLLA--03M") %>%  
  mutate(datetime = Date.Time) %>%  
  mutate(wtr_3.0 = Water.Temp.C) %>%  
  select(datetime, wtr_3.0)
```

```
Data_4 <- Data %>%  
  filter(SiteID == "HOLLA--04M") %>%  
  mutate(datetime = Date.Time) %>%  
  mutate(wtr_4.0 = Water.Temp.C) %>%  
  select(datetime, wtr_4.0)
```

```
Data_5 <- Data %>%  
  filter(SiteID == "HOLLA--05M") %>%  
  mutate(datetime = Date.Time) %>%  
  mutate(wtr_5.0 = Water.Temp.C) %>%  
  select(datetime, wtr_5.0)
```

```
Data_6 <- Data %>%  
  filter(SiteID == "HOLLA--06M") %>%  
  mutate(datetime = Date.Time) %>%  
  mutate(wtr_6.0 = Water.Temp.C) %>%  
  select(datetime, wtr_6.0)
```

```
Data_7 <- Data %>%  
  filter(SiteID == "HOLLA--07M") %>%  
  mutate(datetime = Date.Time) %>%  
  mutate(wtr_7.0 = Water.Temp.C) %>%  
  select(datetime, wtr_7.0)
```

```
Data_8 <- Data %>%  
  filter(SiteID == "HOLLA--08M") %>%  
  mutate(datetime = Date.Time) %>%  
  mutate(wtr_8.0 = Water.Temp.C) %>%  
  select(datetime, wtr_8.0)
```

```
Data_9 <- Data %>%  
  filter(SiteID == "HOLLA--09M") %>%  
  mutate(datetime = Date.Time) %>%  
  mutate(wtr_9.0 = Water.Temp.C) %>%
```



```

select(datetime, wtr_9.0)

Data_10 <- Data %>%
  filter(SiteID == "HOLLA--10M") %>%
  mutate(datetime = Date.Time) %>%
  mutate(wtr_10.0 = Water.Temp.C) %>%
  select(datetime, wtr_10.0)

### Join the dataframes together

Data_ts <- left_join(Data_1, Data_2, by="datetime") %>%
  left_join(., Data_3) %>%
  left_join(., Data_4) %>%
  left_join(., Data_5) %>%
  left_join(., Data_6) %>%
  left_join(., Data_7) %>%
  left_join(., Data_8) %>%
  left_join(., Data_9) %>%
  left_join(., Data_10) %>%
  na.omit()

# Write a .wtr file the entire dataset
write.table(Data_ts, file = "./Holland/Data5_R_Files/HOLLA.wtr", sep = "\t",
            row.names = FALSE)

# Separate each full year into it's own data frame and write .wtr files

Data19 <- Data_ts %>%
  filter(datetime >= "2019-01-01 00:00:00" & datetime <= "2019-12-31 23:00:00")

write.table(Data19, file = "./Holland/Data5_R_Files/HOLLA19.wtr", sep = "\t",
            row.names = FALSE)

Data20 <- Data_ts %>%
  filter(datetime >= "2020-01-01 00:00:00" & datetime <= "2020-12-31 23:00:00")

write.table(Data20, file = "./Holland/Data5_R_Files/HOLLA20.wtr", sep = "\t",
            row.names = FALSE)

Data21 <- Data_ts %>%
  filter(datetime >= "2021-01-01 00:00:00" & datetime <= "2021-12-31 23:00:00")

write.table(Data21, file = "./Holland/Data5_R_Files/HOLLA21.wtr", sep = "\t",
            row.names = FALSE)

Data22 <- Data_ts %>%
  filter(datetime >= "2022-01-01 00:00:00" & datetime <= "2022-12-31 23:00:00")

write.table(Data22, file = "./Holland/Data5_R_Files/HOLLA22.wtr", sep = "\t",
            row.names = FALSE)

```

## 6.2.2 Water Temperature Heat Maps

After you write the required .wtr file, I load it back in to ensure it's in the proper format for rlakeanalyzer.

```
ts = load.ts("./Holland/Data5_R_Files/HOLLA.wtr")

attach(ts)

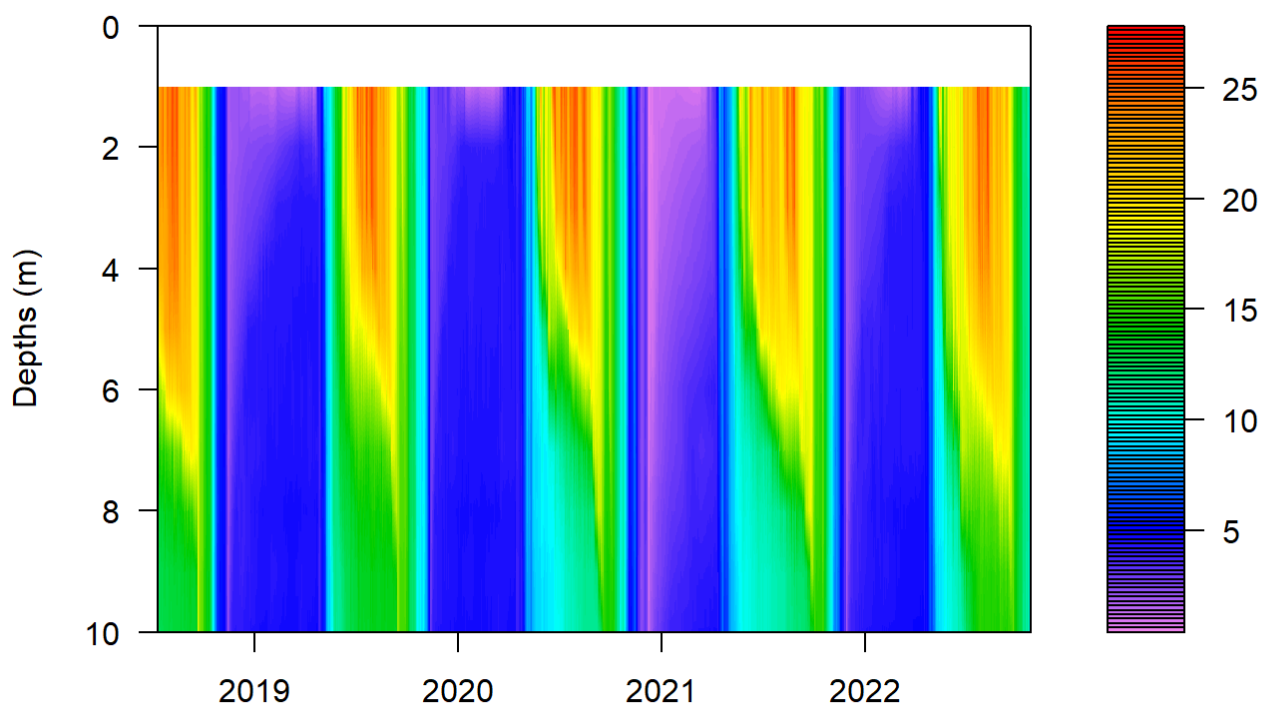
ts <- ts[order(datetime),]

detach(ts)

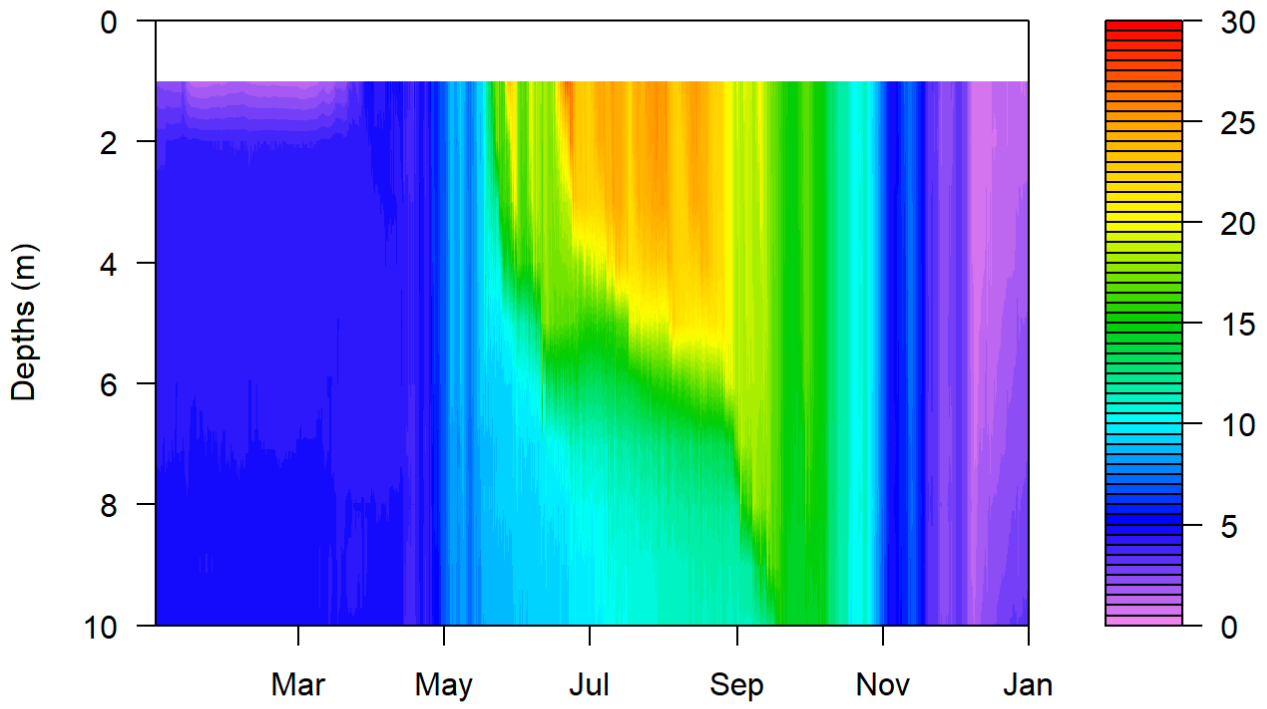
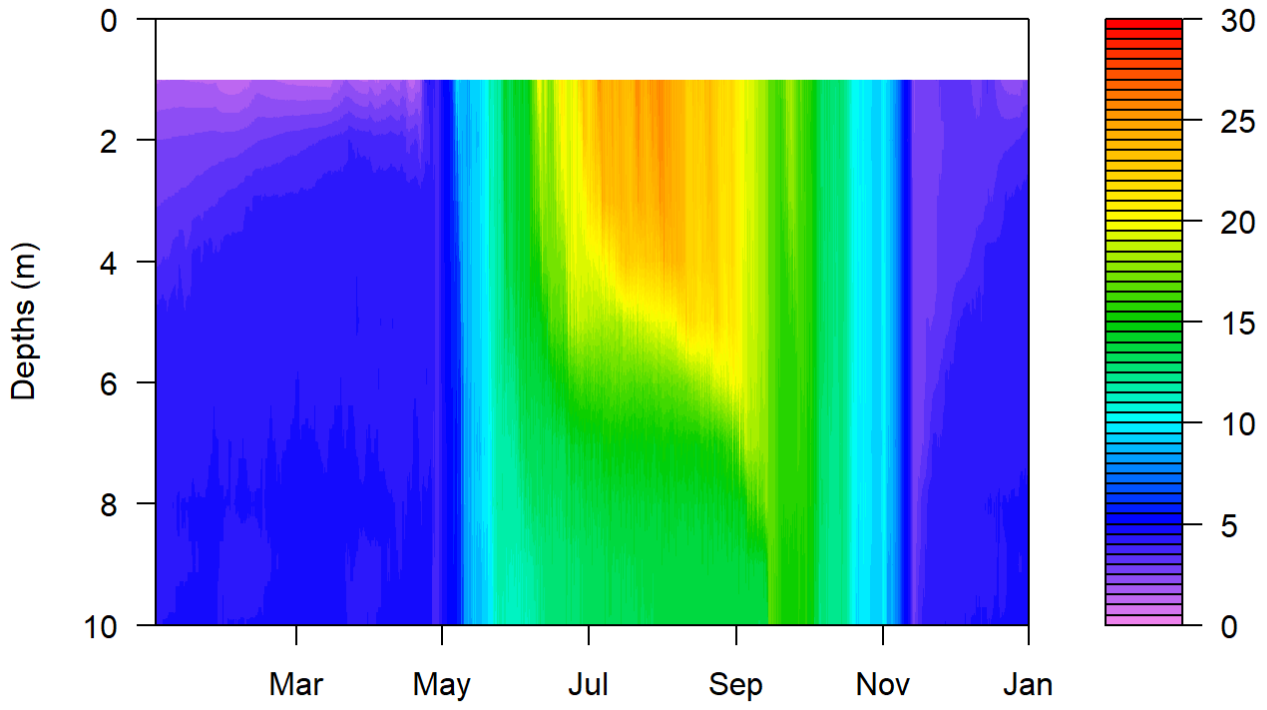
# Here we are going to remove any weird duplicates in the timeseries
ts <- ts[!duplicated(ts$datetime), ]

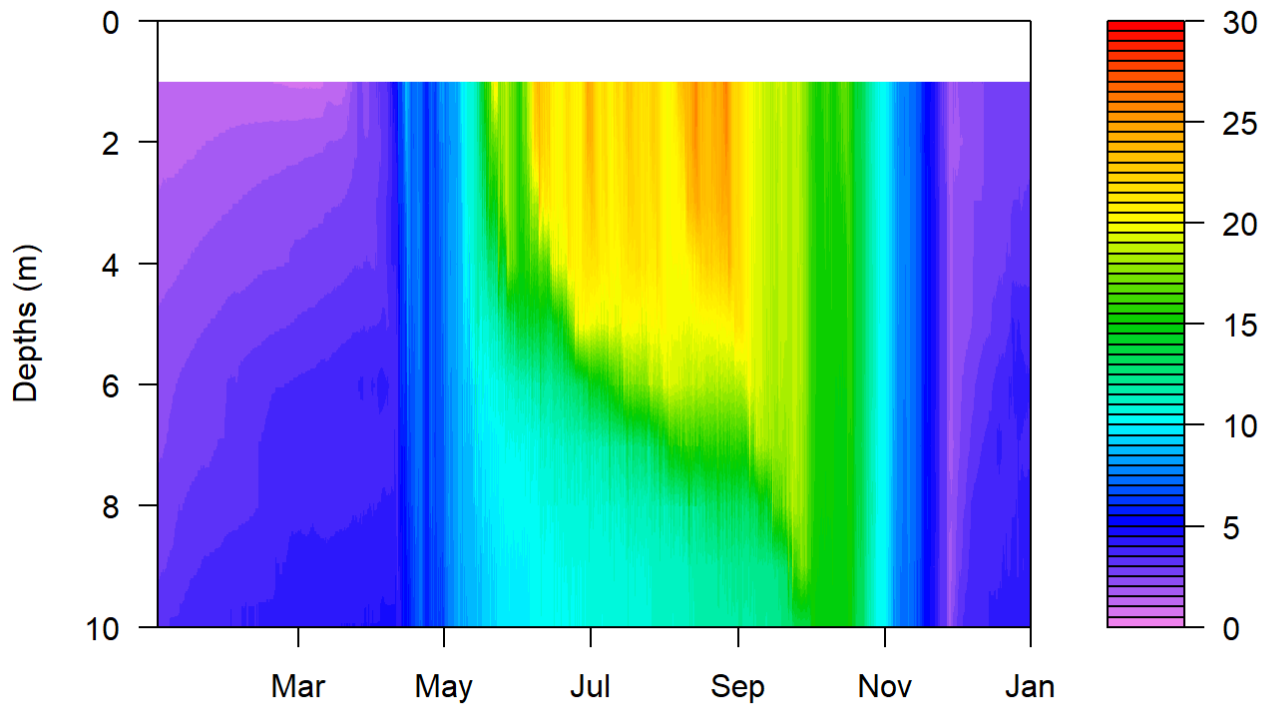
# plot the time series!

wtr.heat.map(ts)
```



From this figure, we can determine that Holland Pond is a dimictic lake, meaning that it has two full mixing events. One that occurs in the spring after ice off, and one that occurs in the fall. However, we can start to look at differences in stratification when we layer these figures on top of one another by year.





We can start to see that stratification starts to begin earlier every year, and that stratification seems to be occurring later into the fall season.

## 6.3 Schmidt Stability

Another metric to look at thermal regimes in lakes is to look at how strong stratification is from year to year. Schmidt stability is a metric that calculates how much wind energy would be required to fully mix the water column. Therefore, higher values indicate that more wind energy is required to mix the water column, meaning that the lake is likely stratified.

In order to calculate schmidt stability, we need to have both bathymetric data and the water temperature matrix. Specifically, rlakeanalyzer requires depths and the areas of each of those depth slices. Similar to the heat map matrix with the .wtr file, bathymetric data needs to have a specific .bth file.

## 6.3.1 Creating the .bth file

```
# reading in appropriate bathymetry data to use for schmidt stability.

Holl_bathy <- read.csv("./Holland/Data5_R_Files/HOLLA_hypso.csv") %>%
  mutate(depths = Contour_depth..m.) %>%
  mutate(areas = Area) %>%
  select(depths, areas)

# writing a .bth file that is required for rlakeanalyzer

write.table(Holl_bathy, file = "./Holland/Data5_R_Files/Holla_Bath.bth", sep = "\t",
            row.names = FALSE)

# reading in the file you just created

Holl_bathy <- load.bathy("./Holland/Data5_R_Files/Holla_Bath.bth")
```

## 6.3.2 Calculating Schmidt Stability

After loading in the .bth file, we can reload the .wtr file and use these two files to run that schmidt.stability function in the rlakeanalyzer package.

```
ts = load.ts("./Holland/Data5_R_Files/HOLLA.wtr")

Holl_Schmidt <- ts.schmidt.stability(ts, Holl_bathy)

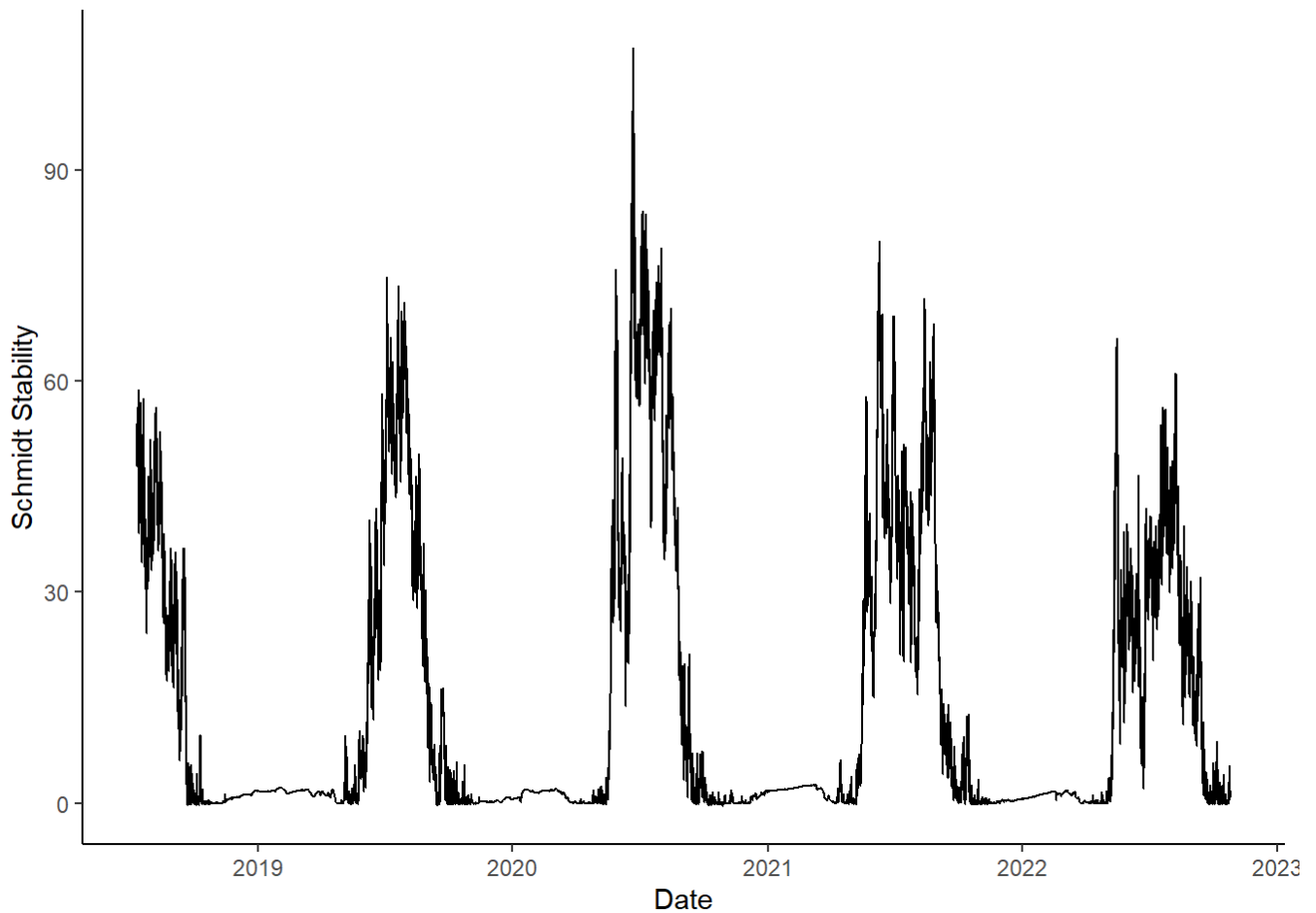
head(Holl_Schmidt)
```

```
##           datetime schmidt.stability
## 1 2018-07-11 19:00:00          53.98091
## 2 2018-07-11 19:00:00          53.98348
## 3 2018-07-11 19:00:00          53.97817
## 4 2018-07-11 20:00:00          52.59499
## 5 2018-07-11 20:00:00          52.59326
## 6 2018-07-11 20:00:00          52.58857
```

## 6.3.3 Visualizing Schmidt Stability

After examining the Schmidt stability dataframe, we can very easily visualize the data using ggplot.

```
Holl_Schmidt %>%
  ggplot(aes(datetime, schmidt.stability)) +
  geom_line() +
  theme_classic() +
  ylab("Schmidt Stability") +
  xlab("Date")
```



### 6.3.4 Trends in Schmidt Stability over time

Looking at this type of information on it's own can be hard to digest and to understand. However, we can use "anomalies" to understand how schmidt stability compares from year to year, and to to the overall average.

I use anomalies throughout this project to guage how trends deviate from the 4(ish) year average from when the loggers were deployed in 2018 to when the data was last downloaded in 2022.

Below is how I create those anomalies.

```

#### Schmidt Stability Trends ####

Holla_year.ss <- Holl_Schmidt %>%
  mutate(Month = month(datetime),
         Day = as.numeric(day(datetime)),
         Year = year(datetime)) %>%
  group_by(Month, Day) %>%
  summarise(Yearly.ss=mean(schmidt.stability))

Holla_day.ss <- Holl_Schmidt %>%
  mutate(Month = month(datetime),
         Day = as.numeric(day(datetime)),
         Year = year(datetime)) %>%
  group_by(Month, Day, Year) %>%
  summarise(Daily.ss=mean(schmidt.stability))

Holl_ss_trend <- left_join(Holla_year.ss, Holla_day.ss, by=c("Month","Day"))

Holl_ss_trend <- Holl_ss_trend %>%
  mutate(SS.Diff = Daily.ss-Yearly.ss) %>%
  mutate(Year1 = 2000) %>%
  mutate(Date = make_date(Year1, Month, Day)) %>%
  mutate(RealDate = make_date(Year, Month, Day))

write.csv(Holl_ss_trend, "./Holland/Data5_R_Files/Holl_SS_Trend.csv")

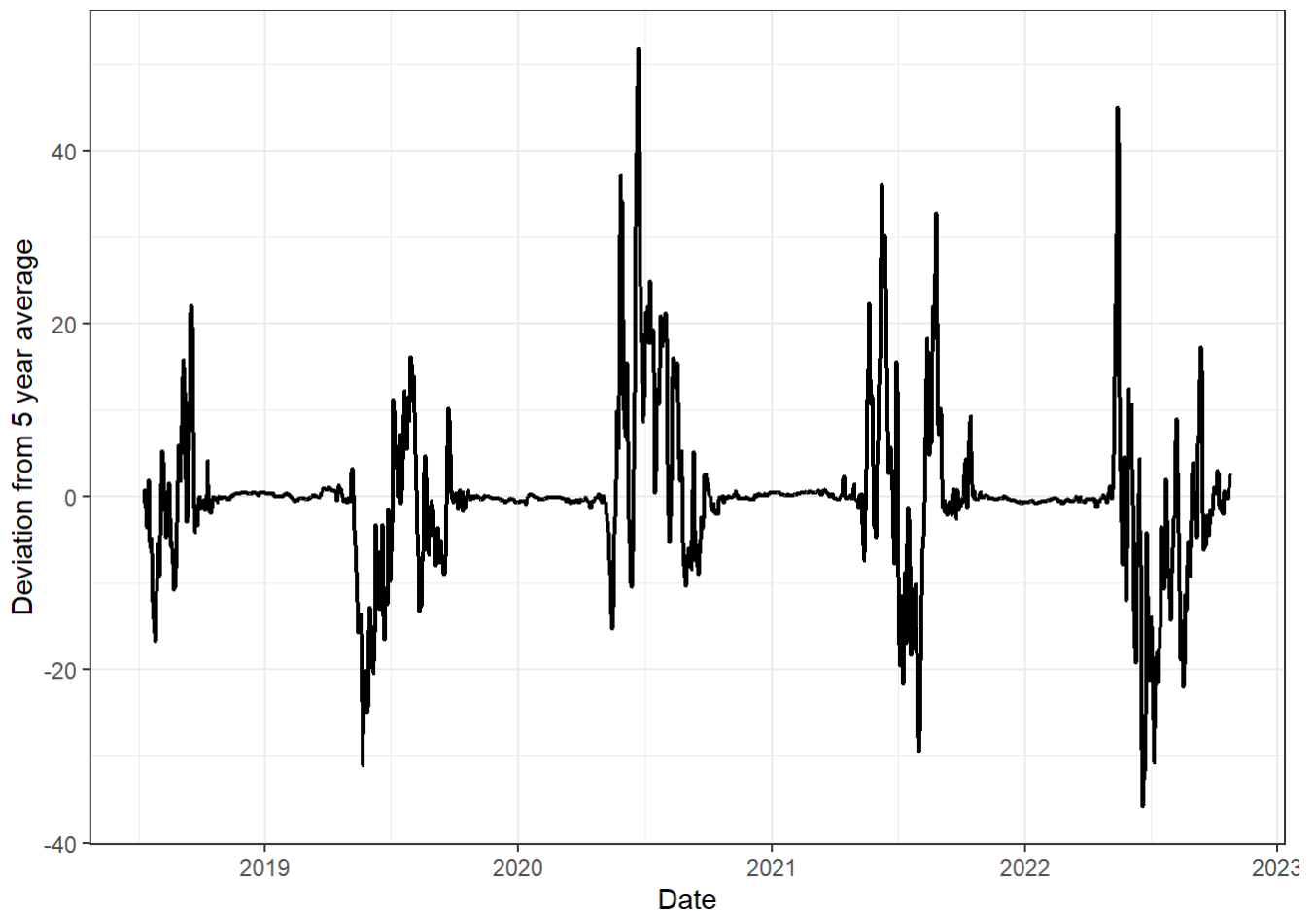
```

After you create the schmidt stability anomalies, you can graph how the daily schmidt stability compares to the daily average across the whole sampling period.

```

Holl_ss_trend %>%
  ggplot(aes(x=RealDate, y=SS.Diff))+
  geom_line(size=0.75) +
  theme_bw() +
  ylab("Deviation from 5 year average") +
  xlab("Date")

```



## 6.4 Thermocline Depth

Next we can use the `rlakeanalyzer` package to identify what depth the thermocline exists at every hour that temperature profiles are logged at. The thermocline depth function only requires that temperature matrix we created earlier (`ts`).

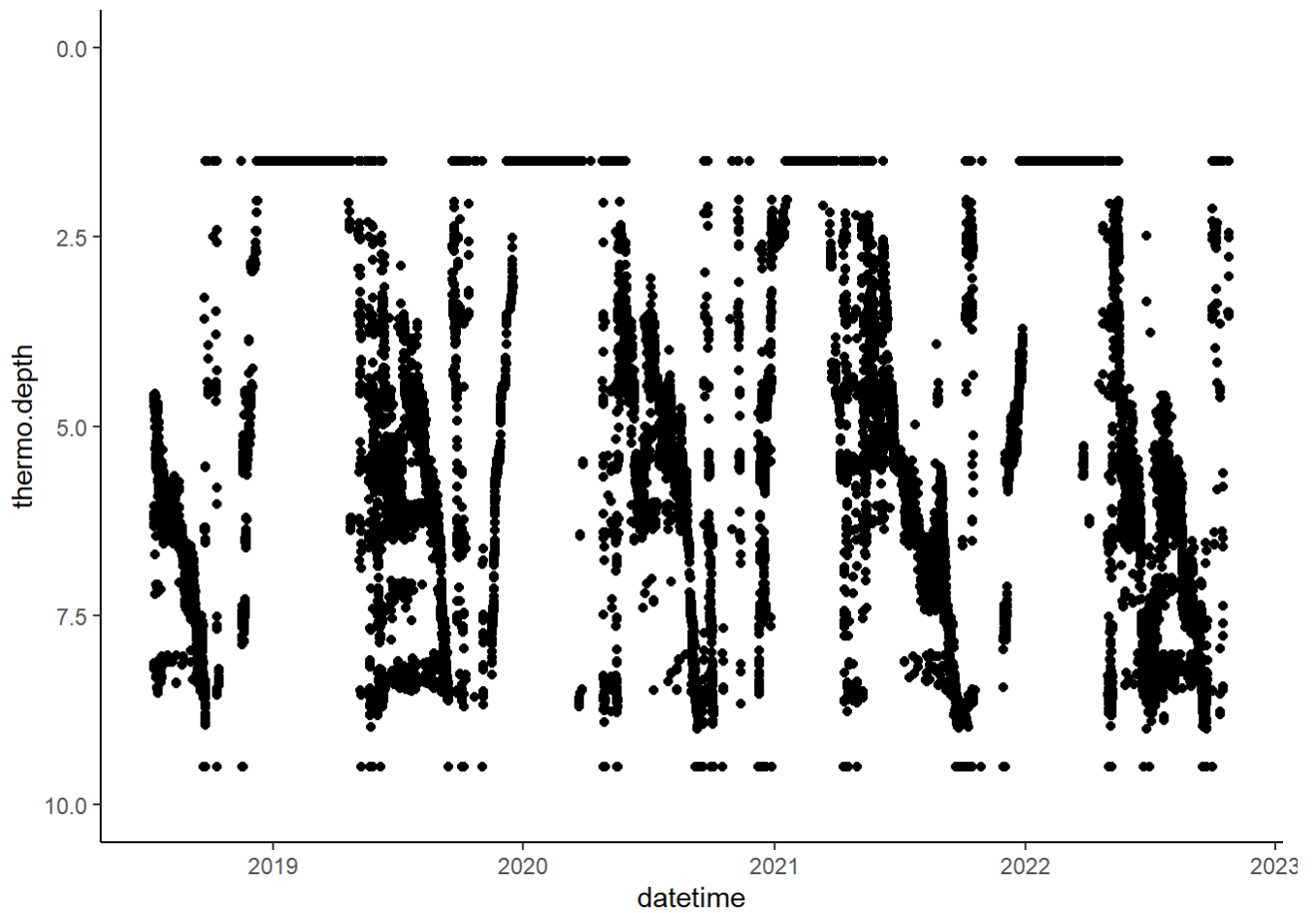
```
therm <- ts.thermo.depth(ts)

therm <- therm %>%
  mutate(Month = month(datetime),
         Year = year(datetime),
         Day = day(datetime))

therm %>%
  ggplot(aes(x=datetime, y=thermo.depth)) +
  geom_point() +
  theme_classic() +
  ylim(10,0)
```

```
## Warning: Removed 23056 rows containing missing values (`geom_point()`).
```





### 6.4.1 Trends in thermocline depth

If you wanted to know how thermocline depths compare to one another over time, then we can create those same anomalies as we did for schmidt stability.

```

Holl_year.therm <- therm %>%
  select(Month, Day, Year, thermo.depth) %>%
  group_by(Month, Day) %>%
  summarize(Yearly.Therm = mean(thermo.depth))

Holl_daily.therm <- therm %>%
  select(Month, Day, Year, thermo.depth) %>%
  group_by(Month, Day, Year) %>%
  summarize(Daily.Therm = mean(thermo.depth))

Holl_therm_trend <- left_join(Holl_daily.therm, Holl_year.therm,
                             by=c("Month", "Day"))

Holl_therm_trend <- Holl_therm_trend %>%
  mutate(Therm.Diff = Daily.Therm-Yearly.Therm) %>%
  mutate(Year1 = 2000) %>%
  mutate(Date = make_date(Year1, Month, Day)) %>%
  mutate(RealDate = make_date(Year, Month, Day))

write.csv(Holl_therm_trend, "./Holland/Data5_R_Files/Holl_Therm_Trend.csv")

##### Line graph of differences. Are we getting warmer over time?

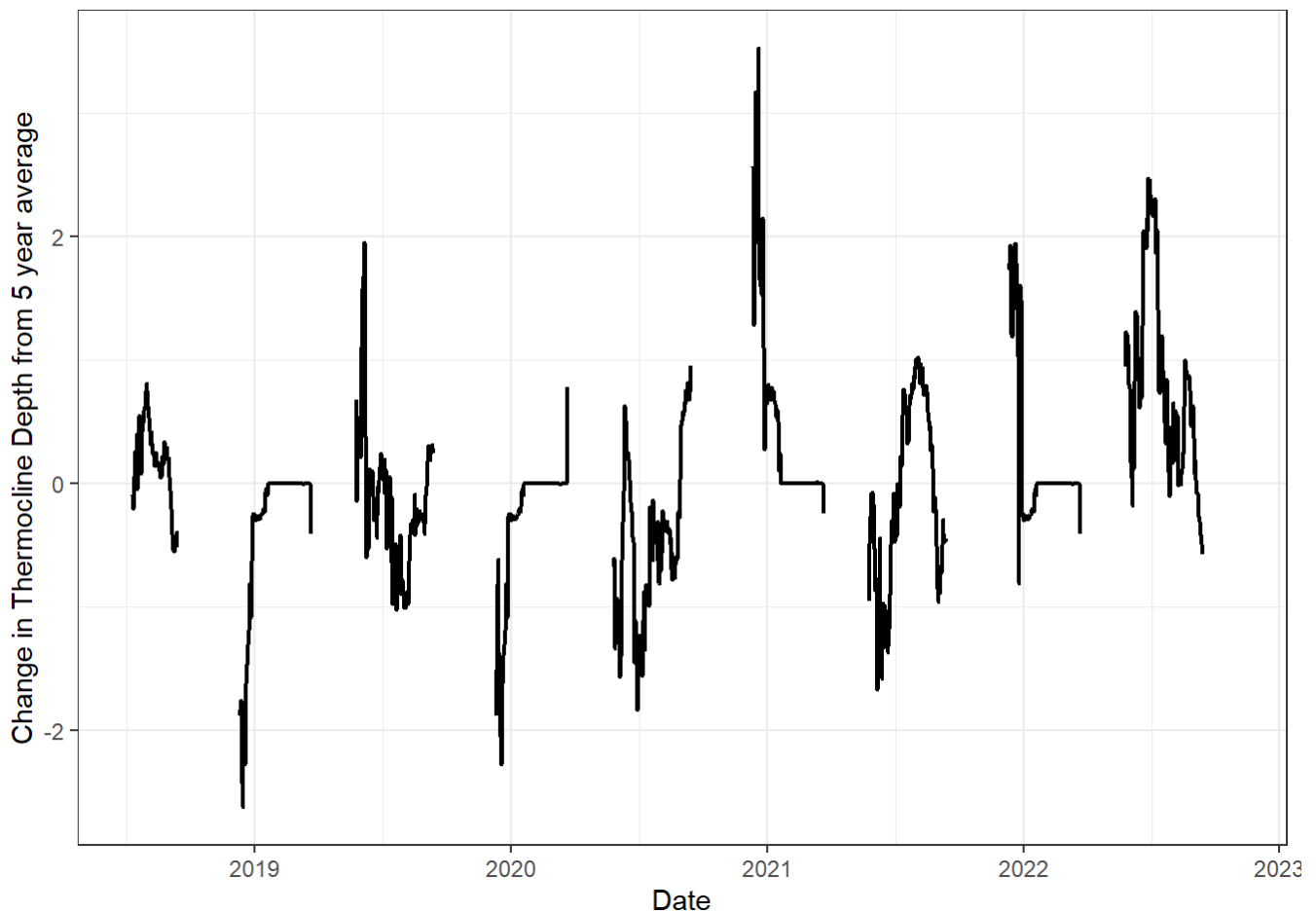
Holl_therm_trend %>%
  ggplot(aes(x=RealDate, y=Therm.Diff))+
  geom_line(size=0.75) +
  theme_bw() +
  ylab("Change in Thermocline Depth from 5 year average") +
  xlab("Date")

```

```

## Warning: Removed 42 rows containing missing values (`geom_line()`).

```



## 6.5 Onset of Stratification

It is hypothesized that lakes are going to experience longer duration of summer stratification as a result of climate change. The next chunk of code will investigate how long Holland Pond remained stratified each year to determine if stratification is increasing over time.

For this analysis, I am deeming a lake as "stratified" if there is a  $\geq 1\text{C}$  change in temperature per meter in the water column.

### 6.5.1 Formatting Data

Because the depth columns are currently labeled as "HOLLA-01M", etc. I want to change these to be numeric depths instead of a character string. I also will be creating a column for the julian day, which will help count how many days experienced stratification. Additionally, there are multiple sensors that read water temperature at 10m, so I only selected the temperature logger from the 10m water temperature hobo file.

```
Data <- Data %>%
  select(Date.Time, SiteID, Water.Temp.C, Month, Day, Year,
         MonthDay, Water.LoggerID, Air.Temp.C) %>%
  mutate(Depth = if_else(SiteID == "HOLLA--01M", 1, 0),
         Depth = if_else(SiteID == "HOLLA--02M", 2, Depth),
         Depth = if_else(SiteID == "HOLLA--03M", 3, Depth),
         Depth = if_else(SiteID == "HOLLA--04M", 4, Depth),
         Depth = if_else(SiteID == "HOLLA--05M", 5, Depth),
         Depth = if_else(SiteID == "HOLLA--06M", 6, Depth),
         Depth = if_else(SiteID == "HOLLA--07M", 7, Depth),
         Depth = if_else(SiteID == "HOLLA--08M", 8, Depth),
         Depth = if_else(SiteID == "HOLLA--09M", 9, Depth),
         Depth = if_else(SiteID == "HOLLA--10M", 10, Depth)) %>%
  mutate(jday = yday(Date.Time)) %>%
  mutate(Keep = if_else(Depth == 10, "No", "Yes")) %>%
  mutate(Keep = if_else(Keep == "No" & Water.LoggerID == 20313145,
                       "Yes", Keep)) %>%
  filter(Keep == "Yes")
```

## 6.5.2 Calculating Stratification Length

```
Holl19 <- Data %>%
  filter(Year == "2019") %>%
  select(Month, Day, Year, Date.Time, jday,
         Water.Temp.C, Depth, Water.LoggerID) %>%
  arrange(Date.Time) %>%
  group_by(Date.Time) %>%
  mutate(next.temp = lead(Water.Temp.C)) %>%
  mutate(Temp.dif = next.temp - Water.Temp.C) %>%
  filter(Temp.dif <= -1) %>%
  group_by(jday) %>%
  count(jday) %>%
  nrow()

Holl20 <- Data %>%
  filter(Year == "2020") %>%
  select(Month, Day, Year, Date.Time, jday, Water.Temp.C, Depth) %>%
  arrange(Date.Time) %>%
  group_by(Date.Time) %>%
  mutate(next.temp = lead(Water.Temp.C)) %>%
  mutate(Temp.dif = next.temp - Water.Temp.C) %>%
  filter(Temp.dif <= -1) %>%
  group_by(jday) %>%
  count(jday) %>%
  nrow()

Holl21 <- Data %>%
  filter(Year == "2021") %>%
  select(Month, Day, Year,
         Date.Time, jday, Water.Temp.C, Depth) %>%
  group_by(Date.Time) %>%
  arrange(Date.Time) %>%
  mutate(next.temp = lead(Water.Temp.C)) %>%
  mutate(Temp.dif = next.temp - Water.Temp.C) %>%
  filter(Temp.dif <= -1) %>%
  group_by(jday) %>%
  count(jday) %>%
  nrow()
```

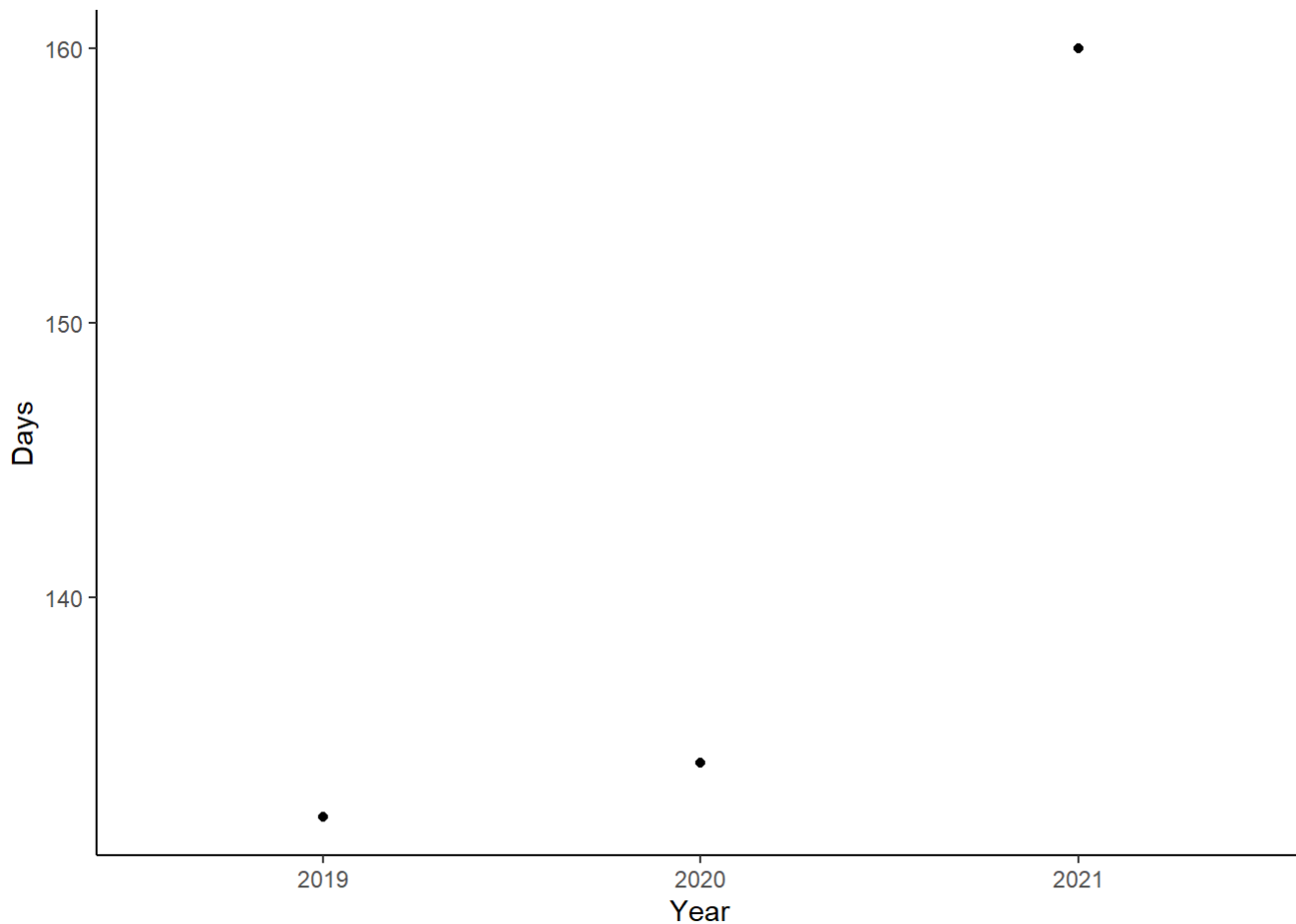
## 6.5.3 Creating a Dataframe

```
Holl_Strat <- rbind(Holl19, Holl20, Holl21) %>%
  as.data.frame() %>%
  rename(Days = V1) %>%
  mutate(Lake = "Holland") %>%
  rownames_to_column(var="Year") %>%
  mutate(Year = str_replace(Year, "Holl", "20"))

write.csv(Holl_Strat, "./Holland/Data5_R_Files/Holl_Strat.csv")
```

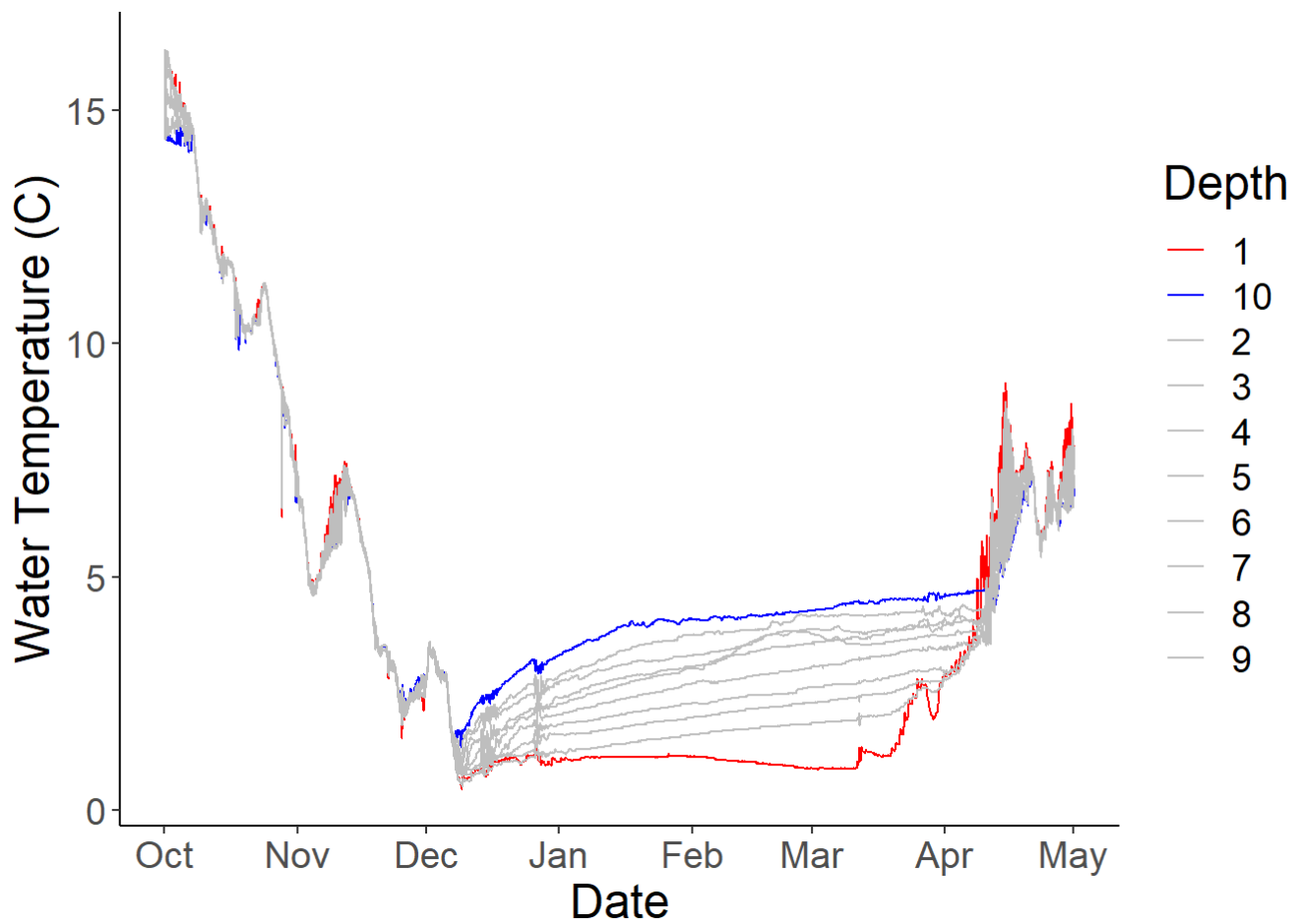
## 6.5.4 Visualize Stratification Length

```
Holl_Strat %>%  
  ggplot(aes(x=Year, y=Days)) +  
  geom_point() +  
  theme_classic()
```



## 6.6 Length of Ice Periods

Like in the summer, lakes often experience stratification in the winter period as a result of ice cover. When ice cover is present, water temperatures tend to be cooler at the surface of the water than at the bottom. This figure represents this inverse stratification, where the blue line is 10m, and the red line is 1m.



To calculate the period of “ice cover,” I calculated how many days in a given winter period experienced this inverse stratification, where the surface water was cooler than the layer below it by 1 degree.

## 6.6.1 Calculating length of Ice Periods

```
Holl_Ice18 <- Data %>%
  filter(Date.Time >= "2018-10-01 00:00:00" &
         Date.Time <= "2019-06-01 00:00:00") %>%
  select(Month, Day, Year, Date.Time, jday, Water.Temp.C, Depth) %>%
  arrange(Date.Time) %>%
  group_by(Date.Time) %>%
  mutate(next.temp = lead(Water.Temp.C)) %>%
  mutate(Temp.dif = next.temp - Water.Temp.C) %>%
  filter(Temp.dif >= 1) %>%
  group_by(jday) %>%
  count(jday) %>%
  nrow()
```

```
Holl_Ice19 <- Data %>%
  filter(Date.Time >= "2019-10-01 00:00:00" &
         Date.Time <= "2020-06-01 00:00:00") %>%
  select(Month, Day, Year, Date.Time, jday, Water.Temp.C, Depth) %>%
  arrange(Date.Time) %>%
  group_by(Date.Time) %>%
  mutate(next.temp = lead(Water.Temp.C)) %>%
  mutate(Temp.dif = next.temp - Water.Temp.C) %>%
  filter(Temp.dif >= 1) %>%
  group_by(jday) %>%
  count(jday) %>%
  nrow()
```

```
Holl_Ice20 <- Data %>%
  filter(Date.Time >= "2020-10-01 00:00:00" &
         Date.Time <= "2021-06-01 00:00:00") %>%
  select(Month, Day, Year, Date.Time, jday, Water.Temp.C, Depth) %>%
  arrange(Date.Time) %>%
  group_by(Date.Time) %>%
  mutate(next.temp = lead(Water.Temp.C)) %>%
  mutate(Temp.dif = next.temp - Water.Temp.C) %>%
  filter(Temp.dif >= 1) %>%
  group_by(jday) %>%
  count(jday) %>%
  nrow()
```

```
Holl_Ice21 <- Data %>%
  filter(Date.Time >= "2021-10-01 00:00:00" &
         Date.Time <= "2022-06-01 00:00:00") %>%
  select(Month, Day, Year, Date.Time, jday, Water.Temp.C, Depth) %>%
  arrange(Date.Time) %>%
  group_by(Date.Time) %>%
  mutate(next.temp = lead(Water.Temp.C)) %>%
  mutate(Temp.dif = next.temp - Water.Temp.C) %>%
  filter(Temp.dif >= 1) %>%
  group_by(jday) %>%
```



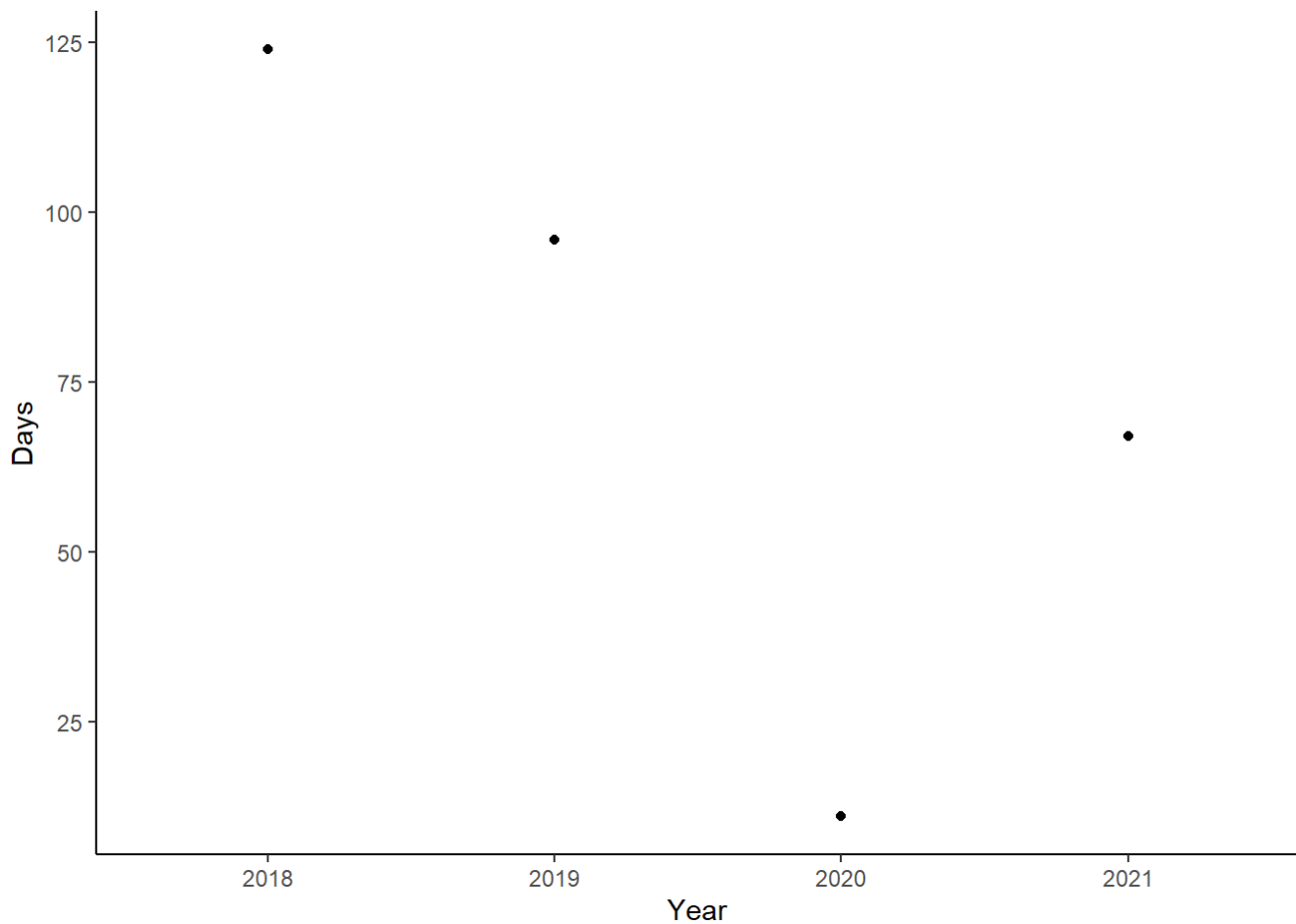
```
count(jday) %>%  
nrow()
```

## 6.6.2 Creating a dataframe

```
Holl_Ice <- rbind(Holl_Ice18, Holl_Ice19, Holl_Ice20, Holl_Ice21) %>%  
  as.data.frame() %>%  
  rename(Days = V1) %>%  
  mutate(Lake = "Holland") %>%  
  rownames_to_column(var="Year") %>%  
  mutate(Year = str_replace(Year, "Holl_Ice", "20"))  
  
write.csv(Holl_Ice, "./Holland/Data5_R_Files/Holl_Ice.csv")
```

## 6.6.3 Visualizing ice period trends

```
Holl_Ice %>%  
  ggplot(aes(x=Year, y=Days)) +  
  geom_point() +  
  theme_classic()
```



## 6.7 Surface Water Trends

It is well documented in the literature that epilimnetic waters are increasing in temperature as a result of climate change. Northern temperature lakes are included, however, it is seen that their surface waters are increases two times faster than that of air temperatures. The next bit of code will be exploring if that pattern is occurring in Holland Pond.

### 6.7.1 Epilimnion Trends over 5 years

```
epi_year.temp <- Data %>%
  select(MonthDay, Year, Water.Temp.C, SiteID) %>%
  filter(SiteID == "HOLLA--01M") %>%
  group_by(MonthDay, SiteID) %>%
  summarize(Yearly.Temp = mean(Water.Temp.C))
```

```
## `summarise()` has grouped output by 'MonthDay'. You can override using the
## `.groups` argument.
```

```
epi_day.temp <- Data %>%
  select(MonthDay, Month, Day, Year, Water.Temp.C, SiteID) %>%
  filter(SiteID == "HOLLA--01M") %>%
  group_by(Month, Day, MonthDay, Year) %>%
  summarize(Daily.Temp = mean(Water.Temp.C))
```

```
## `summarise()` has grouped output by 'Month', 'Day', 'MonthDay'. You can
## override using the `.groups` argument.
```

```
Holl_epi_trend <- left_join(epi_day.temp, epi_year.temp, by="MonthDay")

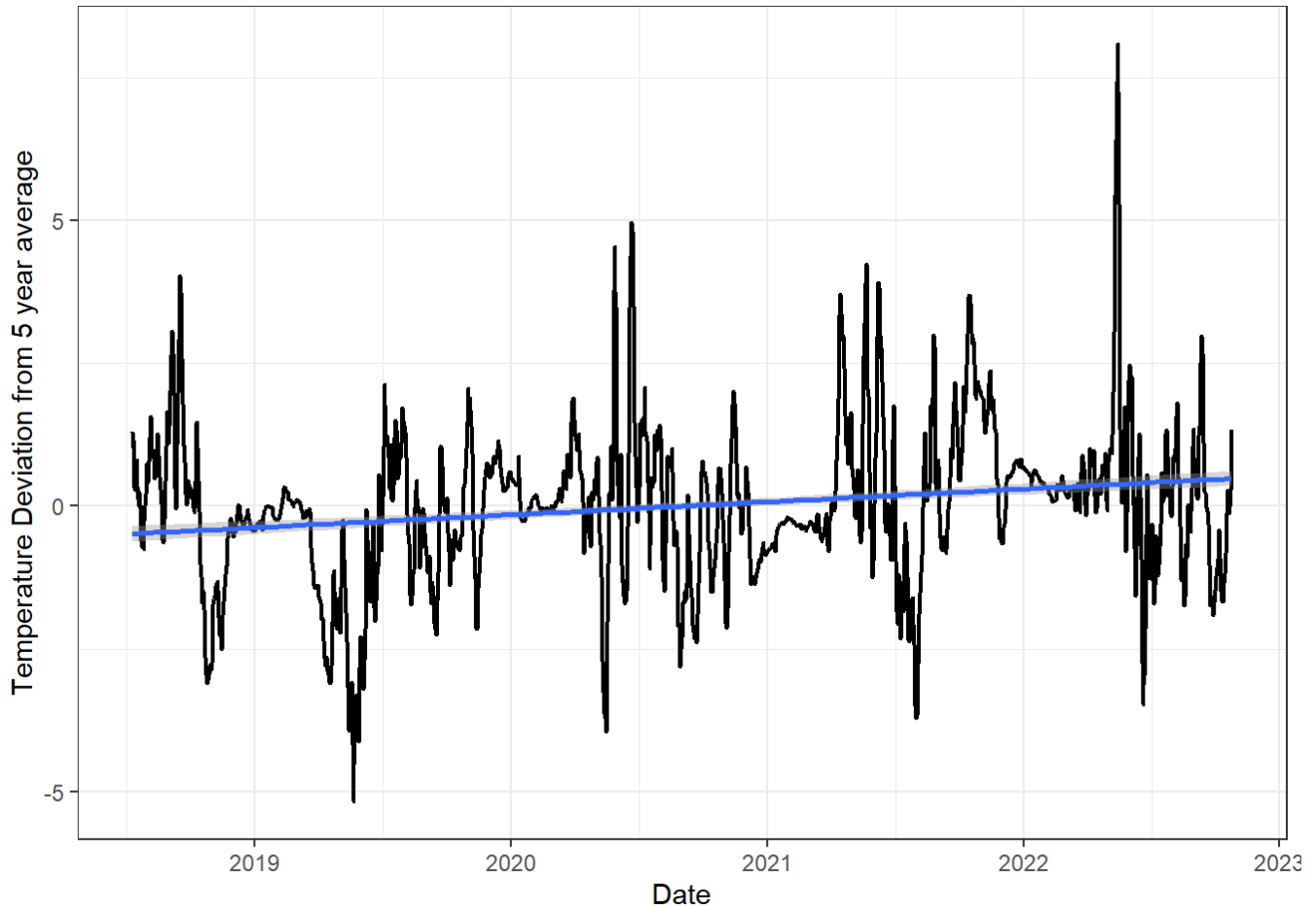
Holl_epi_trend <- Holl_epi_trend %>%
  mutate(Temp.Diff = Daily.Temp-Yearly.Temp) %>%
  mutate(Year1 = 2000) %>%
  mutate(Date = make_date(Year1, Month, Day)) %>%
  mutate(RealDate = make_date(Year, Month, Day))

write.csv(Holl_epi_trend, "./Holland/Data5_R_Files/Holl_Epi_Trend.csv")
```

### 6.7.2 Visualize the trend

```
Holl_epi_trend %>%
  ggplot(aes(x=RealDate, y=Temp.Diff))+
  geom_line(size=0.75) +
  geom_smooth(method="lm") +
  theme_bw() +
  ylab("Temperature Deviation from 5 year average") +
  xlab("Date")
```

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

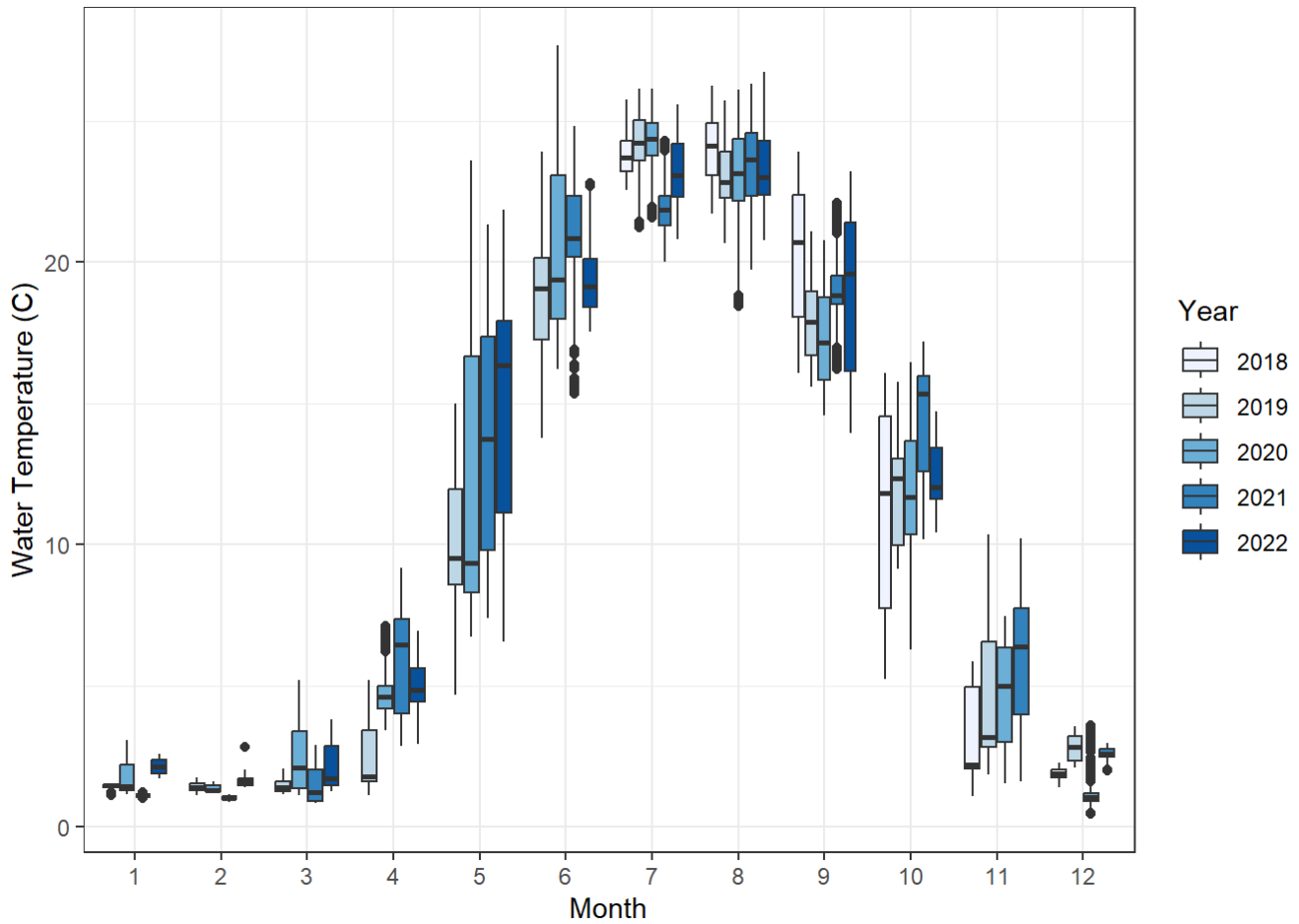


Looking at this figure, we can see that Holland Pond is getting warmer than the five year daily average over time. However, the linear model doesn't take into account seasonality.

### 6.7.3 Where are temperatures warming the most?

To try to determine where the most warming was occurring, I created a boxplot of all temperatures for a given month and compared each year.

```
Data %>%
  filter(SiteID == "HOLLA--01M") %>%
  mutate(Year = as.character(Year)) %>%
  mutate(Month = as.character(Month)) %>%
  mutate(Month = factor(Month, c("1", "2", "3", "4", "5",
                                "6", "7", "8", "9", "10",
                                "11", "12"))) %>%
  ggplot(aes(x=Month, y=Water.Temp.C, fill=Year)) +
  geom_boxplot() +
  theme_bw() +
  ylab("Water Temperature (C)") +
  scale_fill_manual(values=c("#eff3ff", "#bdd7e7", "#6baed6",
                             "#3182bd", "#08519c"))
```



This figure gives us a bit more information. We can see here that the months of May and November are responsible for the greatest amount of warming over time. This makes sense, because we saw in our temperature heat maps that stratification was starting earlier and lasting longer. Additionally, we saw that the number of days that stratification was occurring was increasing over time.

## 6.7.4 Seasonal Trends in Surface water temperatures

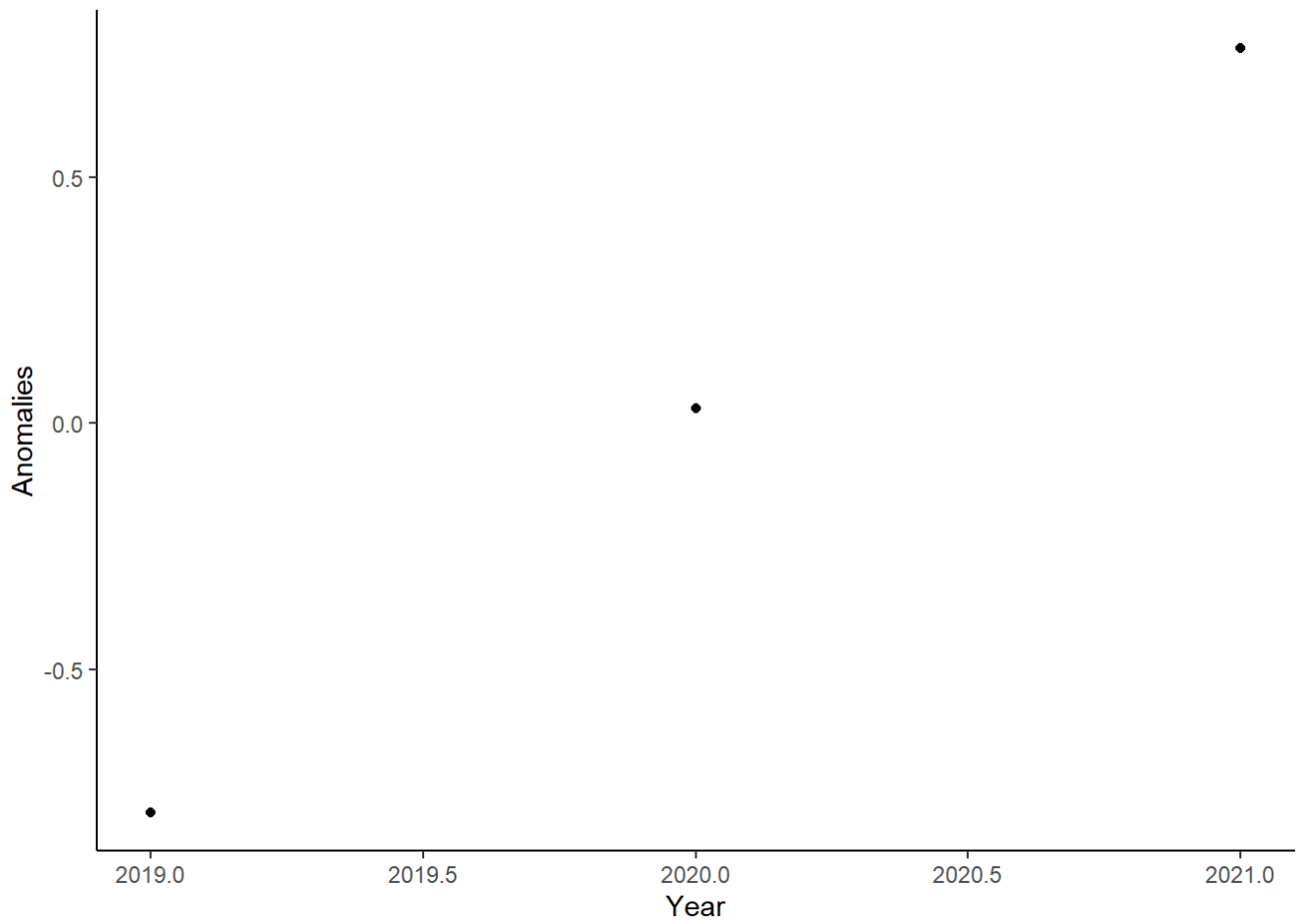
### 6.7.5 Summer

```
Holl_epi.summ.avg <- Data %>%
  filter(Year != 2022 & Year != 2018) %>%
  filter(Month >= 4 & Month < 12) %>%
  select(MonthDay, Year, Water.Temp.C, SiteID) %>%
  filter(SiteID == "HOLLA--01M") %>%
  summarize(Summ.Avg.Temp = mean(Water.Temp.C))

Holl_epi.summ.avg <- Holl_epi.summ.avg$Summ.Avg.Temp

Holl_epi.summ.temp <- Data %>%
  filter(Year != 2022 & Year != 2018) %>%
  filter(Month >= 4 & Month < 12) %>%
  select(MonthDay, Month, Day, Year, Water.Temp.C, SiteID) %>%
  filter(SiteID == "HOLLA--01M") %>%
  group_by(Year) %>%
  summarize(Yearly.Summ.Temp = mean(Water.Temp.C)) %>%
  ungroup() %>%
  mutate(Avg.Temp = Holl_epi.summ.avg) %>%
  mutate(Anomalies = (Yearly.Summ.Temp-Avg.Temp))

Holl_epi.summ.temp %>%
  ggplot(aes(x=Year, y=Anomalies)) +
  geom_point()+
  theme_classic()
```



```
write.csv(Holl_epi.summ.temp, "./Holland/Data5_R_Files/Holl_epi.summ.temp.csv")
```

From the figure above, we can tell that our summer temperatures are warming over time.

## 6.7.6 Winter

```
#### Epi Winter Temperature Anomalies ####

Holl_epi.wint.avg.18 <- Data %>%
  filter(Date.Time >= "2018-12-01 00:00:00" &
         Date.Time < "2019-04-01 00:00:00") %>%
  select(MonthDay, Year, Water.Temp.C, SiteID) %>%
  filter(SiteID == "HOLLA--01M") %>%
  summarize(Wint.Avg.Temp = mean(Water.Temp.C))%>%
  mutate(Year = 2018)

Holl_epi.wint.avg.19 <- Data %>%
  filter(Date.Time >= "2019-12-01 00:00:00" &
         Date.Time < "2020-04-01 00:00:00") %>%
  select(MonthDay, Year, Water.Temp.C, SiteID) %>%
  filter(SiteID == "HOLLA--01M") %>%
  summarize(Wint.Avg.Temp = mean(Water.Temp.C))%>%
  mutate(Year = 2019)

Holl_epi.wint.avg.20 <- Data %>%
  filter(Date.Time >= "2020-12-01 00:00:00" &
         Date.Time < "2021-04-01 00:00:00") %>%
  select(MonthDay, Year, Water.Temp.C, SiteID) %>%
  filter(SiteID == "HOLLA--01M") %>%
  summarize(Wint.Avg.Temp = mean(Water.Temp.C))%>%
  mutate(Year = 2020)

Holl_epi.wint.avg.21 <- Data %>%
  filter(Date.Time >= "2021-12-01 00:00:00" &
         Date.Time < "2022-04-01 00:00:00") %>%
  select(MonthDay, Year, Water.Temp.C, SiteID) %>%
  filter(SiteID == "HOLLA--01M") %>%
  summarize(Wint.Avg.Temp = mean(Water.Temp.C)) %>%
  mutate(Year = 2021)

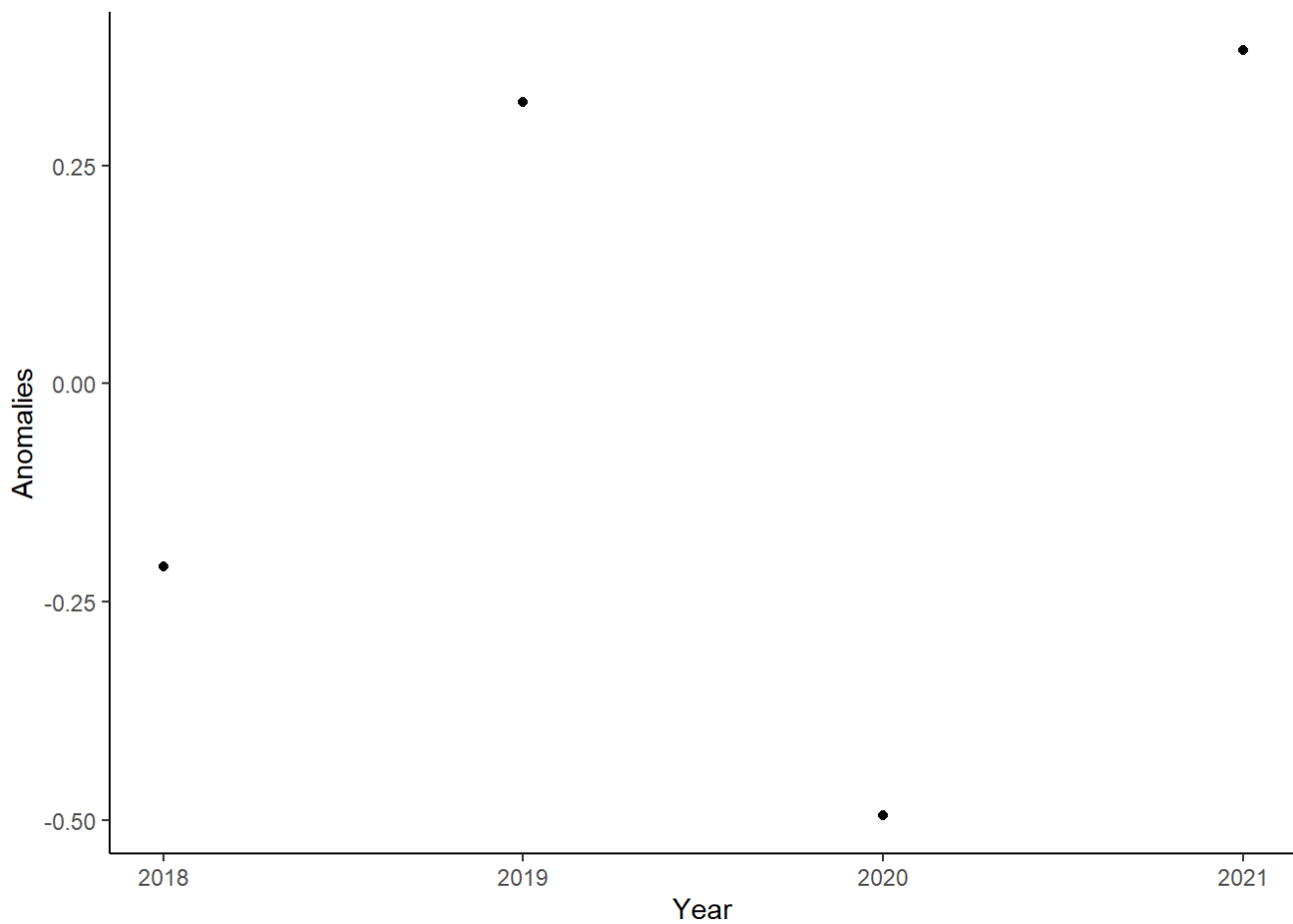
Holl_epi.wint <- rbind(Holl_epi.wint.avg.18,
                      Holl_epi.wint.avg.19,
                      Holl_epi.wint.avg.20,
                      Holl_epi.wint.avg.21)

Holl_epi.wint.avg <- Holl_epi.wint %>%
  summarise(Wint.Avg.Temp = mean(Wint.Avg.Temp))

Holl_epi.wint.avg <- Holl_epi.wint.avg$Wint.Avg.Temp

Holl_epi.wint.temp <- Holl_epi.wint %>%
  mutate(Avg.Temp = Holl_epi.wint.avg) %>%
  mutate(Anomalies = (Wint.Avg.Temp-Avg.Temp))
```

```
Holl_epi.wint.temp %>%  
  ggplot(aes(x=Year, y=Anomalies)) +  
  geom_point()+  
  theme_classic()
```



```
write.csv(Holl_epi.wint.temp, "./Holland/Data5_R_Files/Holl_epi.wint.temp.csv")
```

Here we can see that winter temperatures in Holland Pond are variable and oscillate from being warm and cold.

## 6.8 Air Temperature Trends

While we only have three to four years of data and can't necessarily determine whether or not the trends we are seeing are due to climate change, or if they are just smaller trends in a longer more variable trend over time, we can look at changes in air temperature to determine if a relationship exists between air temperature and water temperature.

### 6.8.1 Air Temperature Anomalies.

```
Data <- read_csv("./Holland/Data3_Aggregate/DATA_QC_HOLLA--01M_Water_00626_91024.csv")
```



```
## Rows: 438774 Columns: 44
## — Column specification —————
## Delimiter: ","
## chr (20): SiteID, Logger.Deployment, Flag.Water.Temp.C, Flag.Gross.Water.Te...
## dbl (14): Water.Temp.C, Water.LoggerID, Water.RowID, Month, Day, Year, Mont...
## lgl (6): Flag.Date.Time, Comment.MOD.Date.Time, Comment.MOD.Water.Temp.C, ...
## dtm (2): Date.Time, RAW.Date.Time
## date (1): Date
## time (1): Time
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
air_year.temp <- Data %>%
  select(MonthDay, Year, Air.Temp.C, SiteID, Flag.Air.Temp.C) %>%
  na.omit(Air.Temp.C) %>%
  filter(Flag.Air.Temp.C == "P") %>%
  group_by(MonthDay, SiteID) %>%
  summarize(Yearly.Temp = mean(Air.Temp.C))
```

```
## `summarise()` has grouped output by 'MonthDay'. You can override using the
## `.groups` argument.
```

```
air_day.temp <- Data %>%
  select(MonthDay, Month, Day, Year, Air.Temp.C, SiteID, Flag.Air.Temp.C) %>%
  na.omit(Air.Temp.C) %>%
  filter(Flag.Air.Temp.C == "P") %>%
  group_by(Month, Day, MonthDay, Year) %>%
  summarize(Daily.Temp = mean(Air.Temp.C))
```

```
## `summarise()` has grouped output by 'Month', 'Day', 'MonthDay'. You can
## override using the `.groups` argument.
```

```
air_trend <- left_join(air_day.temp, air_year.temp, by="MonthDay")

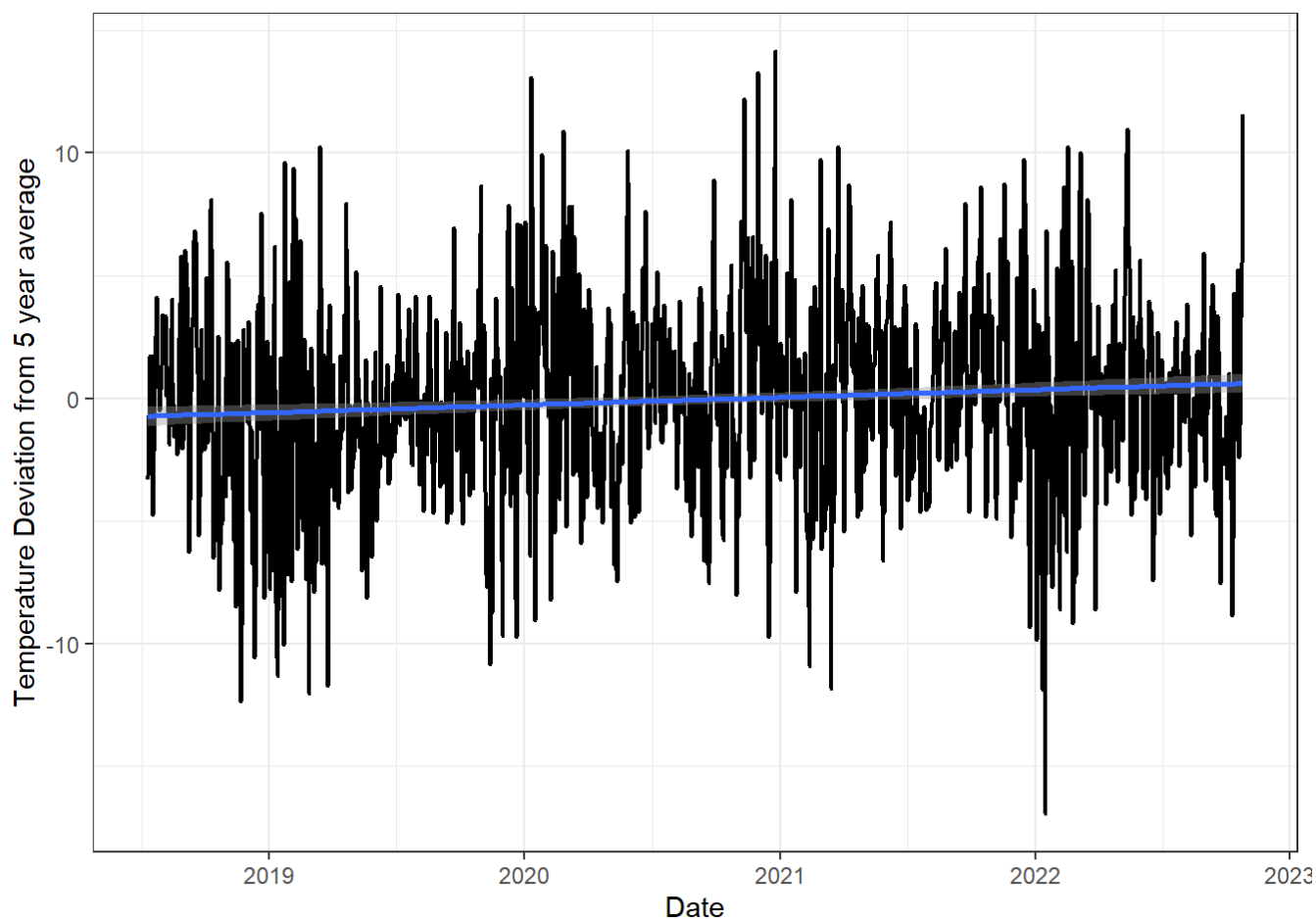
Holl_air_trend <- air_trend %>%
  mutate(Temp.Diff.Air = Daily.Temp-Yearly.Temp) %>%
  mutate(Year1 = 2000) %>%
  mutate(Date = make_date(Year1, Month, Day)) %>%
  mutate(RealDate = make_date(Year, Month, Day))

write.csv(Holl_air_trend, "./Holland/Data5_R_Files/Holl_Air_Trend.csv")
```

## 6.8.2 Visualize Trends

```
Holl_air_trend %>%  
  ggplot(aes(x=RealDate, y=Temp.Diff.Air))+  
  geom_line(size=0.75) +  
  geom_smooth(method="lm") +  
  theme_bw() +  
  ylab("Temperature Deviation from 5 year average") +  
  xlab("Date")
```

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



## 6.9 Dissolved Oxygen

Vermont has Dissolved Oxygen sensors hung 1 meter off the bottom of each lake, however the data needs to be QC'd like the temperature data was.

### 6.9.1 Create a new data frame for dissolved oxygen

```
Data <- read_csv("../Holland/Data3_Aggregate/DATA_QC_HOLLA--01M_Water_00626_91024.csv")
```

```
## Rows: 438774 Columns: 44
## — Column specification —————
## Delimiter: ","
## chr (20): SiteID, Logger.Deployment, Flag.Water.Temp.C, Flag.Gross.Water.Te...
## dbl (14): Water.Temp.C, Water.LoggerID, Water.RowID, Month, Day, Year, Mont...
## lgl (6): Flag.Date.Time, Comment.MOD.Date.Time, Comment.MOD.Water.Temp.C, ...
## dtm (2): Date.Time, RAW.Date.Time
## date (1): Date
## time (1): Time
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

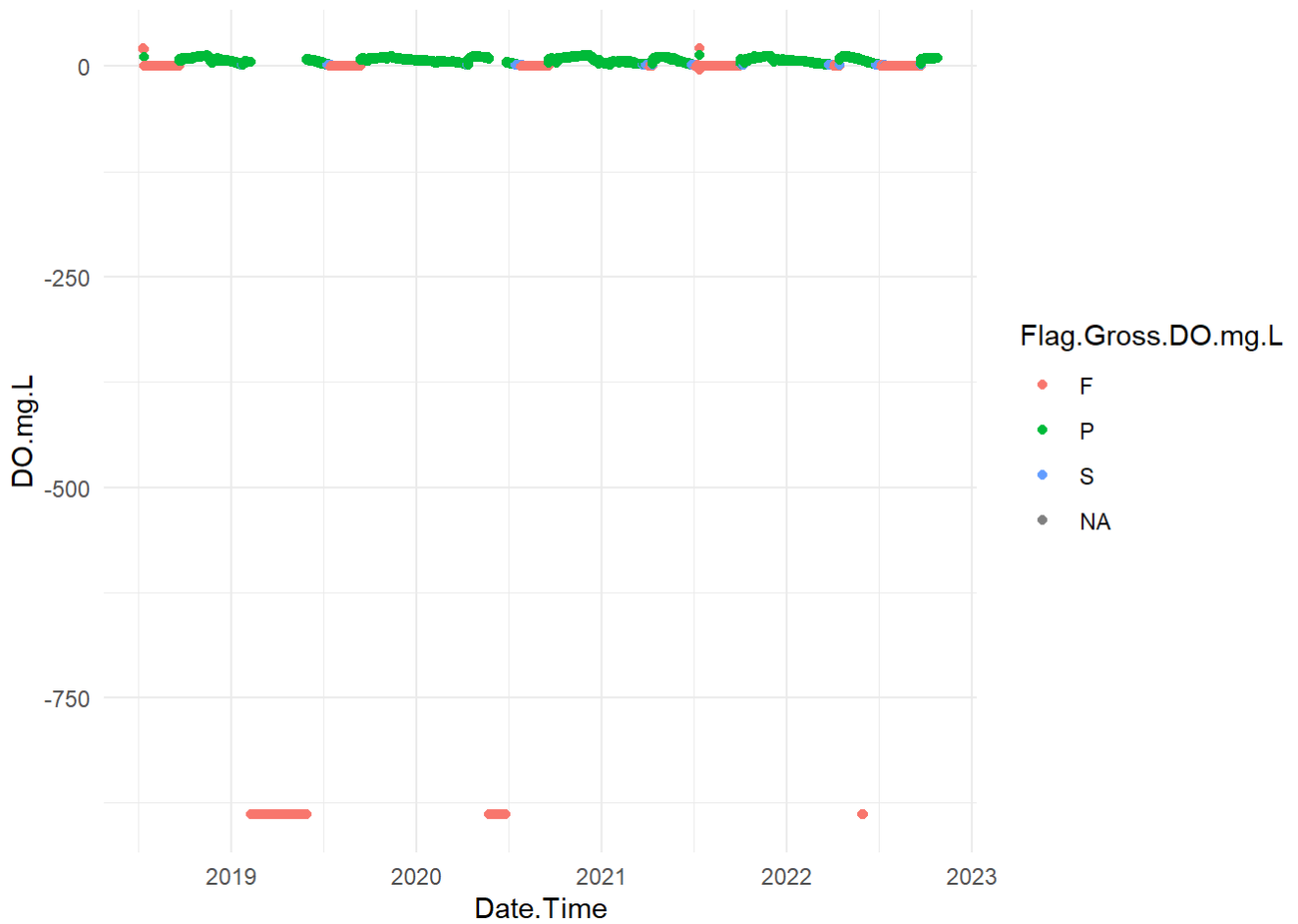
```
DO <- Data %>%
  select(Date, Date.Time, Month, Day, Year, MonthDay, SiteID, DO.mg.L, Flag.DO.mg.L, Flag.
RoC.DO.mg.L,
         Flag.Flat.DO.mg.L, Flag.Gross.DO.mg.L, Flag.Spike.DO.mg.L) %>%
  mutate(jday = yday(Date.Time))
```

## 6.9.2 QA/QC

## 6.9.3 Gross Values

```
DO %>%
  ggplot(aes(x=Date.Time, y=DO.mg.L, color=Flag.Gross.DO.mg.L)) +
  geom_point() +
  theme_minimal()
```

```
## Warning: Removed 401177 rows containing missing values (`geom_point()`).
```

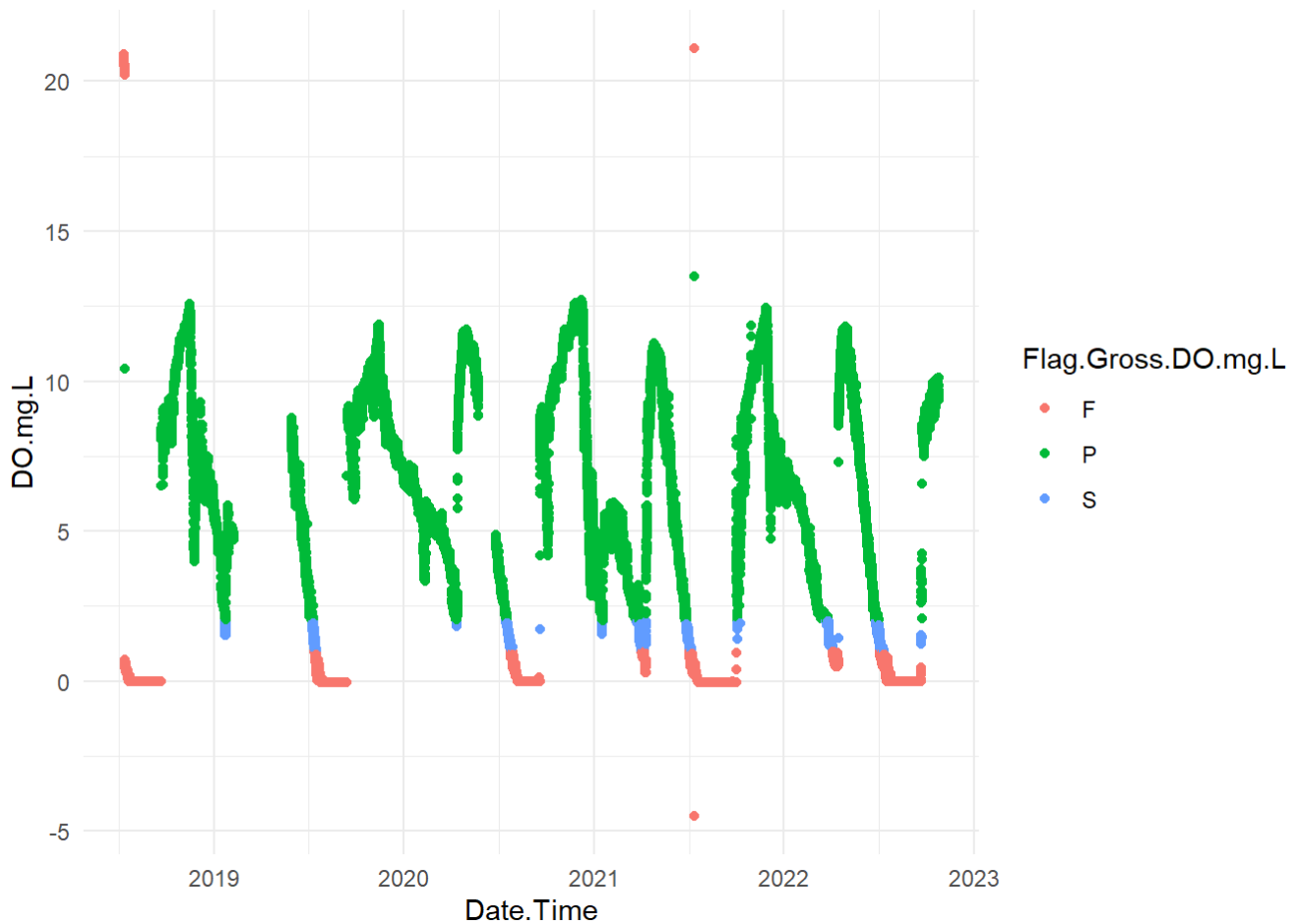


From this figure we can see that there were periods of time where the dissolved oxygen arrays stopped working. We will manually remove those negative values.

```
DO <- DO %>%
  filter(DO.mg.L > -250)
```

We can now recheck those values...

```
DO %>%
  ggplot(aes(x=Date.Time, y=DO.mg.L, color=Flag.Gross.DO.mg.L)) +
  geom_point() +
  theme_minimal()
```

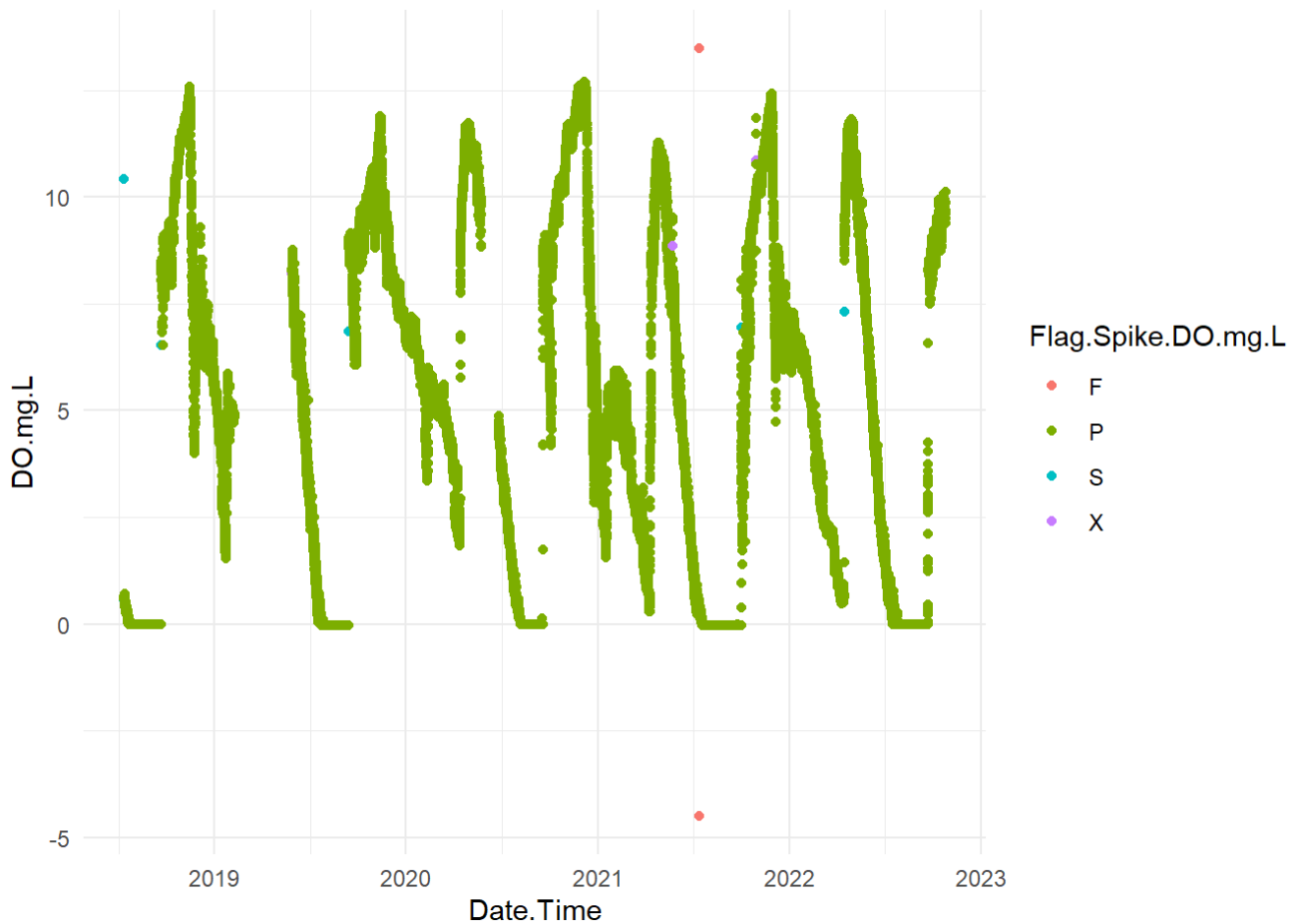


We can see that during deployment of the buoys, there were some inaccurate readings. Let's remove those. The rest of the values that are suspect and fail flags are okay for lakes. They often go hypoxic.

```
DO <- DO %>%
  filter(DO.mg.L < 14)
```

## 6.9.4 Spike Test

```
DO %>%
  ggplot(aes(x=Date.Time, y=DO.mg.L, color=Flag.Spike.DO.mg.L)) +
  geom_point() +
  theme_minimal()
```

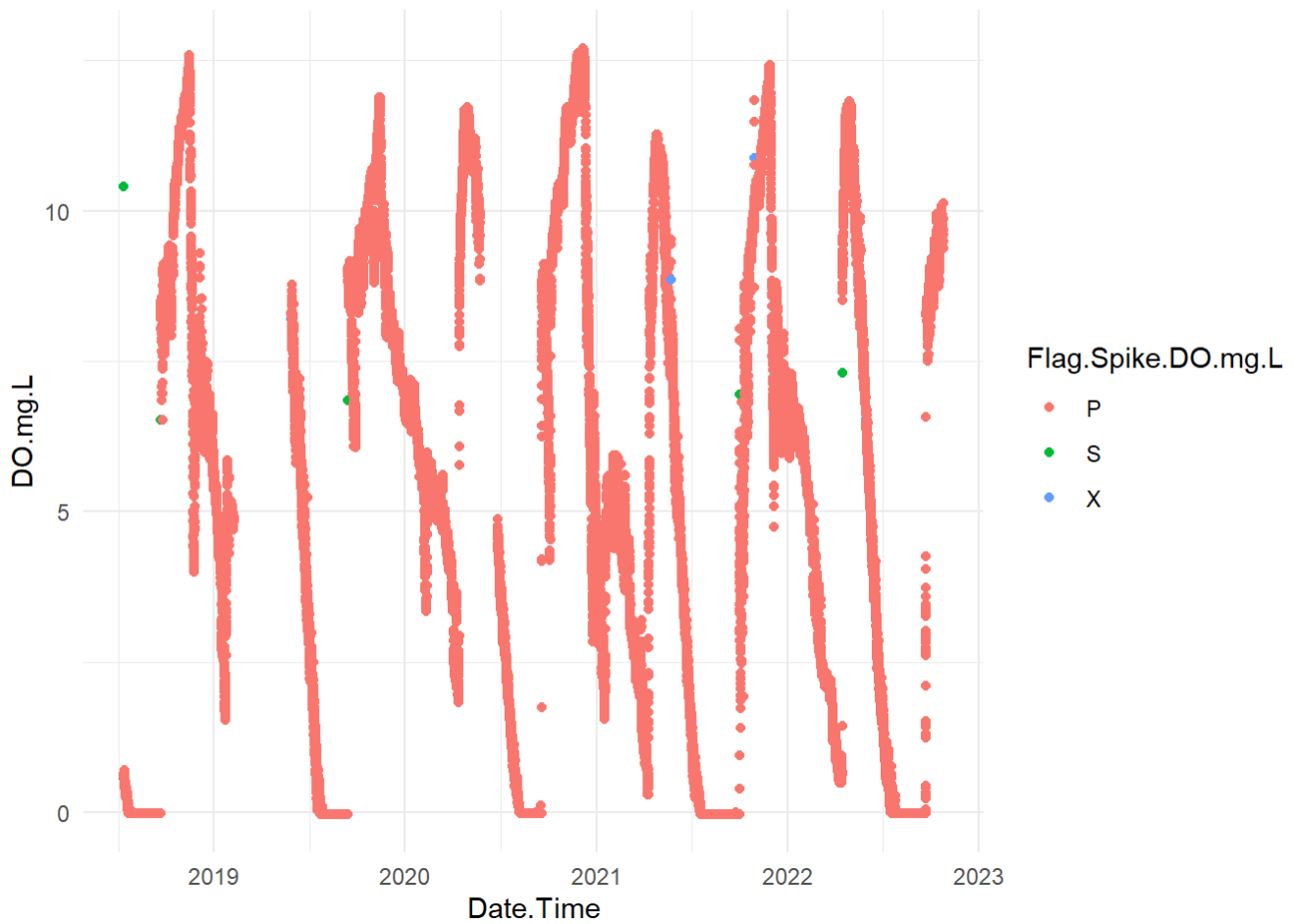


The fail flag can be removed since it is likely occurring during a sampling day where the buoy was pulled and data was downloaded.

```
DO <- DO %>%
  filter(Flag.Spike.DO.mg.L != "F")
```

Now we can check to make sure that value was removed..

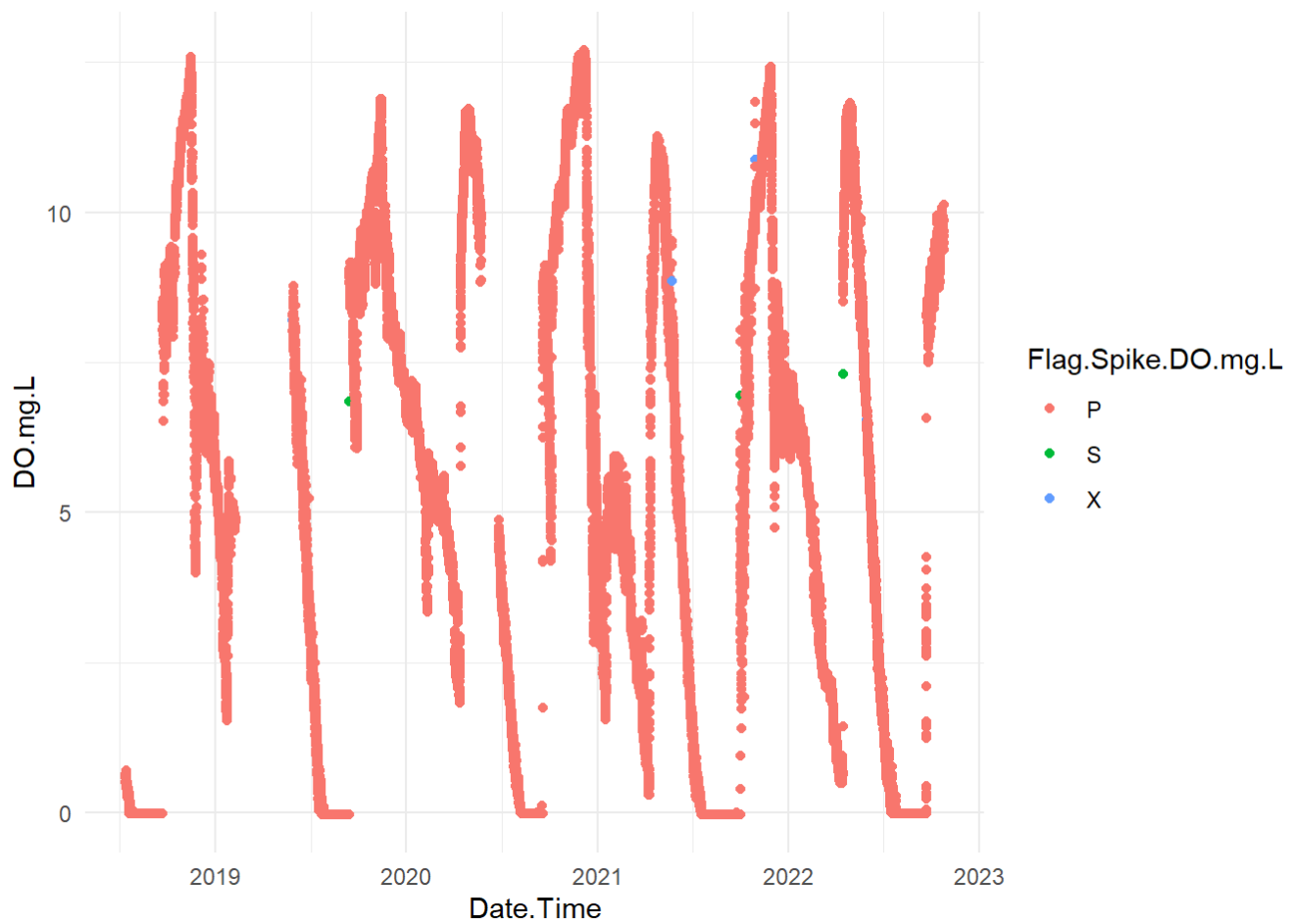
```
DO %>%
  ggplot(aes(x=Date.Time, y=DO.mg.L, color=Flag.Spike.DO.mg.L)) +
  geom_point() +
  theme_minimal()
```



I want to remove that flagged value in 2018 that is labeled as suspect.

```
DO <- DO %>%
  mutate(Remove = if_else(Year == 2018 & Flag.Spike.DO.mg.L == "S", "Yes", "No")) %>%
  filter(Remove == "No")

DO %>%
  ggplot(aes(x=Date.Time, y=DO.mg.L, color=Flag.Spike.DO.mg.L)) +
  geom_point() +
  theme_minimal()
```



## 6.9.5 Rate of Change

```
DO %>%
  ggplot(aes(x=Date.Time, y=DO.mg.L, color=Flag.RoC.DO.mg.L)) +
  geom_point() +
  theme_minimal()
```





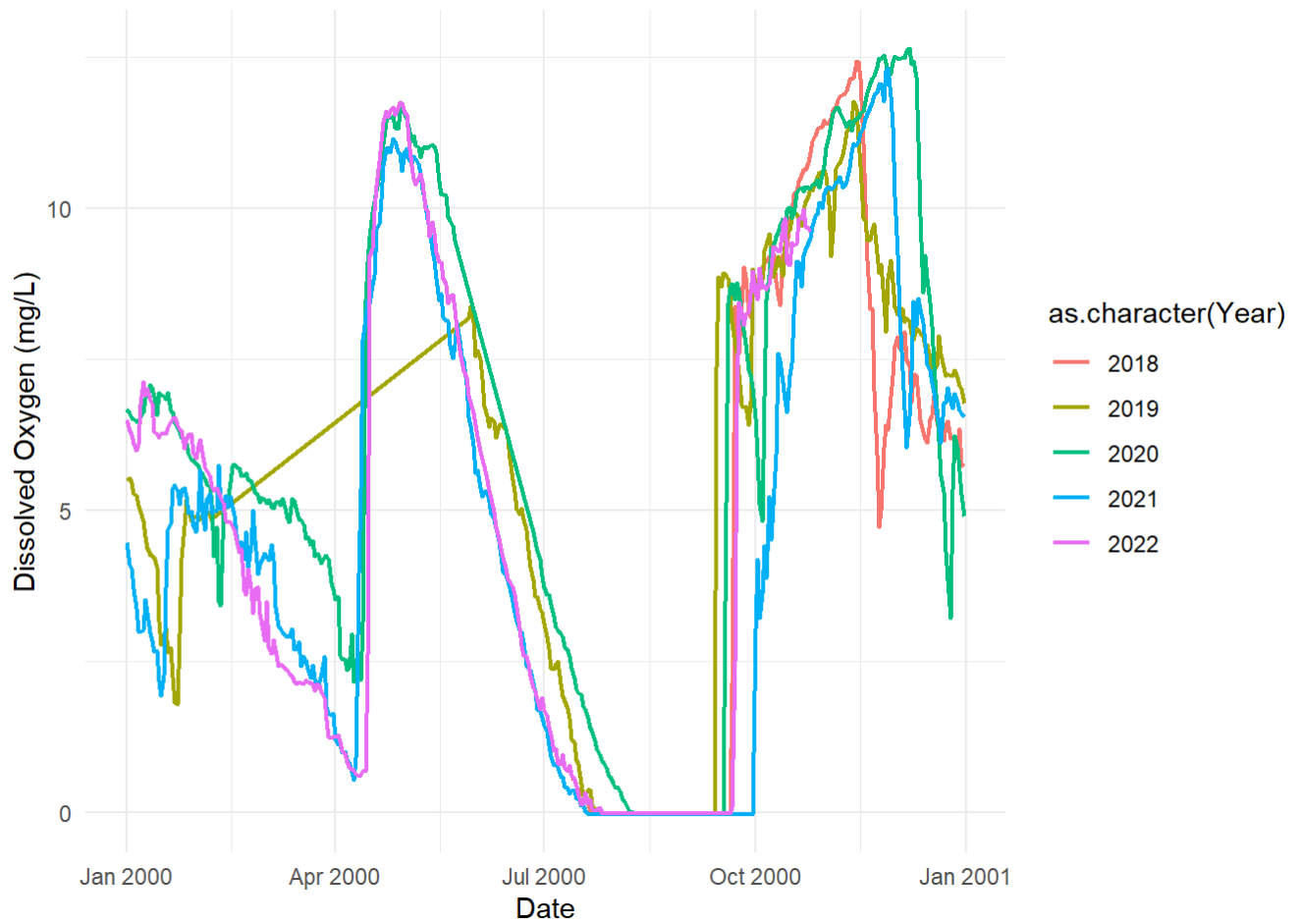
Everything here looks alright.

## 6.9.6 Yearly Dissolved Oxygen Patterns

Now that the data is Qc'd, I am going to create a plot where we plot every year as its own line to try to understand periods of anoxia...when is anoxia setting in, how long does it last, etc.

```
DO %>%
  mutate(Year1 = 2000) %>%
  mutate(FakeDate = make_date(Year1, Month, Day)) %>%
  group_by(Month, Day, Year, FakeDate) %>%
  summarize(Avg_DO = mean(DO.mg.L)) %>%
  ggplot(aes(x=FakeDate, y=Avg_DO, color=as.character(Year))) +
  geom_line(linewidth=0.75) +
  theme_minimal() +
  xlab("Date")+
  ylab("Dissolved Oxygen (mg/L)")
```

```
## `summarise()` has grouped output by 'Month', 'Day', 'Year'. You can override
## using the `.groups` argument.
```



We can see that there are differences in timing of anoxia from year to year. Like stratification and ice periods we can count how many days in a year the lake would be considered “anoxic” by counting how many days dissolved oxygen was below 2 mg/L.

```

D019 <- DO %>%
  filter(Year == "2019") %>%
  select(Month, Day, Year, Date.Time, jday,
         DO.mg.L) %>%
  filter(DO.mg.L <= 2) %>%
  group_by(jday) %>%
  count(jday) %>%
  nrow()

D020 <- DO %>%
  filter(Year == "2020") %>%
  select(Month, Day, Year, Date.Time, jday,
         DO.mg.L) %>%
  filter(DO.mg.L <= 2) %>%
  group_by(jday) %>%
  count(jday) %>%
  nrow()

D021 <- DO %>%
  filter(Year == "2021") %>%
  select(Month, Day, Year, Date.Time, jday,
         DO.mg.L) %>%
  filter(DO.mg.L <= 2) %>%
  group_by(jday) %>%
  count(jday) %>%
  nrow()

#### Create a dataframe

Holl_DO <- rbind(D019, D020, D021) %>%
  as.data.frame() %>%
  rename(Days = V1) %>%
  mutate(Lake = "Holland") %>%
  rownames_to_column(var="Year") %>%
  mutate(Year = str_replace(Year, "D0", "20"))

write.csv(Holl_DO, "./Holland/Data5_R_Files/Holl_DO.csv")

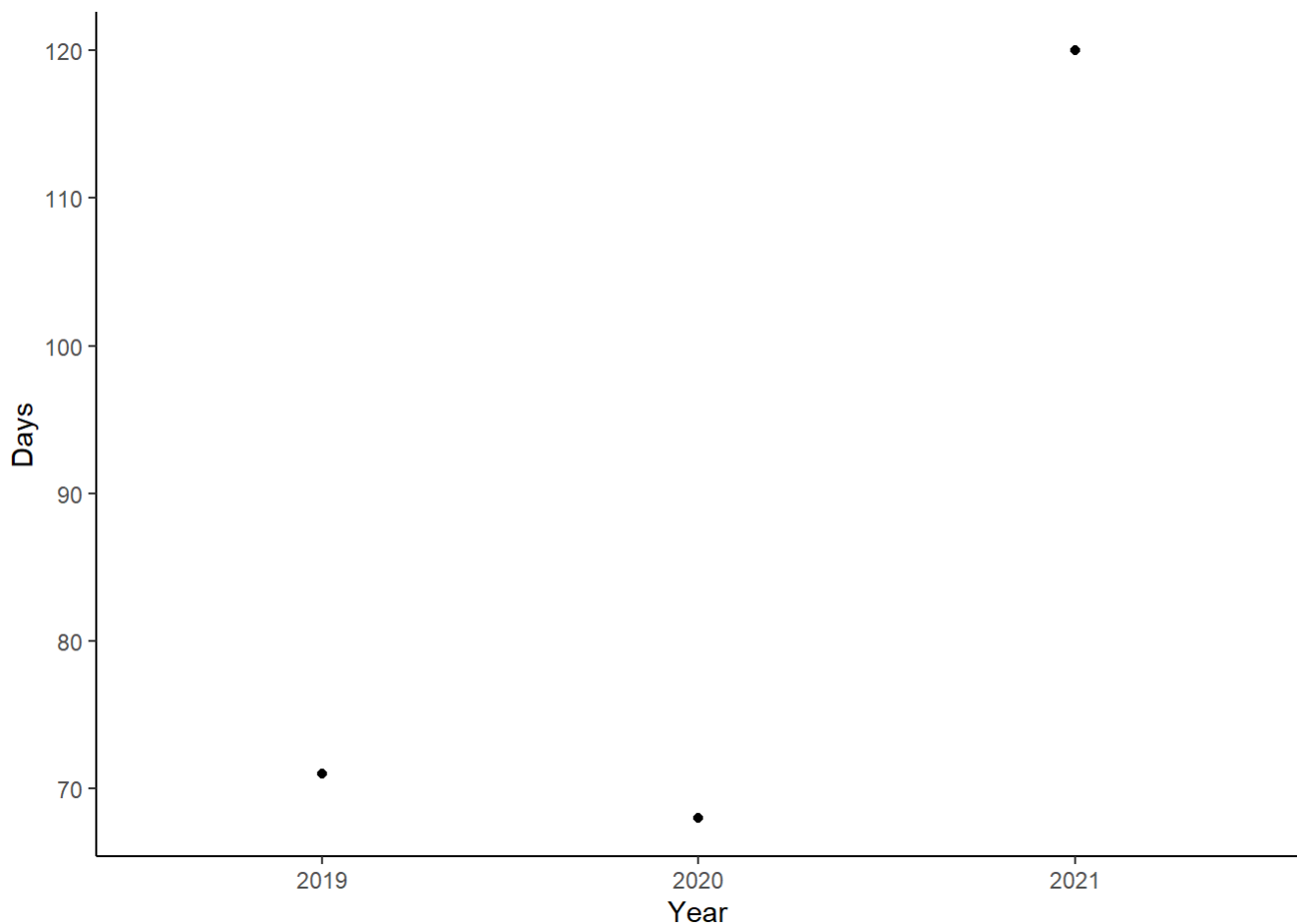
```

We can visualize these trends the same way we did earlier.

```

Holl_DO %>%
  ggplot(aes(x=Year, y=Days)) +
  geom_point() +
  theme_classic()

```



Note that these results aren't accurate unless you have full days of DO data. We don't have that for the first few years of Vermont data.

## 6.10 Water Level Analysis

Cleaning up the Water Level was very similar to cleaning up the water temperature and DO Data. For several of my lakes, the Water Level Data did not join together properly with the ContDataQC Shiny App, so I provided code that would load and join these data frames together into one larger frame.

```
setwd("./Holland/Data1_RAW/")

WaterLevel <- list.files(pattern = "HOLLA--10M_AW*") %>%
  map_df(~read_csv(.))

str(WaterLevel)
```

```

## spc_tbl_ [37,612 × 11] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ SiteID      : chr [1:37612] "HOLLA--10M" "HOLLA--10M" "HOLLA--10M" "HOLLA--10M"
...
## $ Date.Time   : POSIXct[1:37612], format: "2018-07-11 19:00:00" "2018-07-11 20:00:0
0" ...
## $ Air.Temp.C  : num [1:37612] 21.5 17.8 15.5 13.7 12.2 ...
## $ Air.LoggerID : num [1:37612] 20158113 20158113 20158113 20158113 20158113 ...
## $ Air.RowID   : num [1:37612] 1 2 3 4 5 6 7 8 9 10 ...
## $ Water.Temp.C : num [1:37612] 12.6 12.5 12.7 12.5 12.6 ...
## $ Water.LoggerID : num [1:37612] 20158133 20158133 20158133 20158133 20158133 ...
## $ Water.RowID  : num [1:37612] 1 2 3 4 5 6 7 8 9 10 ...
## $ Air.BP.psi   : num [1:37612] 725 725 725 725 726 ...
## $ Water.P.psi  : num [1:37612] 28.2 28.2 28.3 28.3 28.3 ...
## $ Sensor.Depth.ft: num [1:37612] 10 10 10 10 10 ...
## - attr(*, "spec")=
## .. cols(
## ..   SiteID = col_character(),
## ..   Date.Time = col_datetime(format = ""),
## ..   Air.Temp.C = col_double(),
## ..   Air.LoggerID = col_double(),
## ..   Air.RowID = col_double(),
## ..   Water.Temp.C = col_double(),
## ..   Water.LoggerID = col_double(),
## ..   Water.RowID = col_double(),
## ..   Air.BP.psi = col_double(),
## ..   Water.P.psi = col_double(),
## ..   Sensor.Depth.ft = col_double()
## .. )
## - attr(*, "problems")=<externalptr>

```

Using `list.files()`, you can load all files that have the “AW” tag into R, which you can pipe directly into the `map_df()` function. The `map_df()` function will load all of the .csv files and join them together for you.

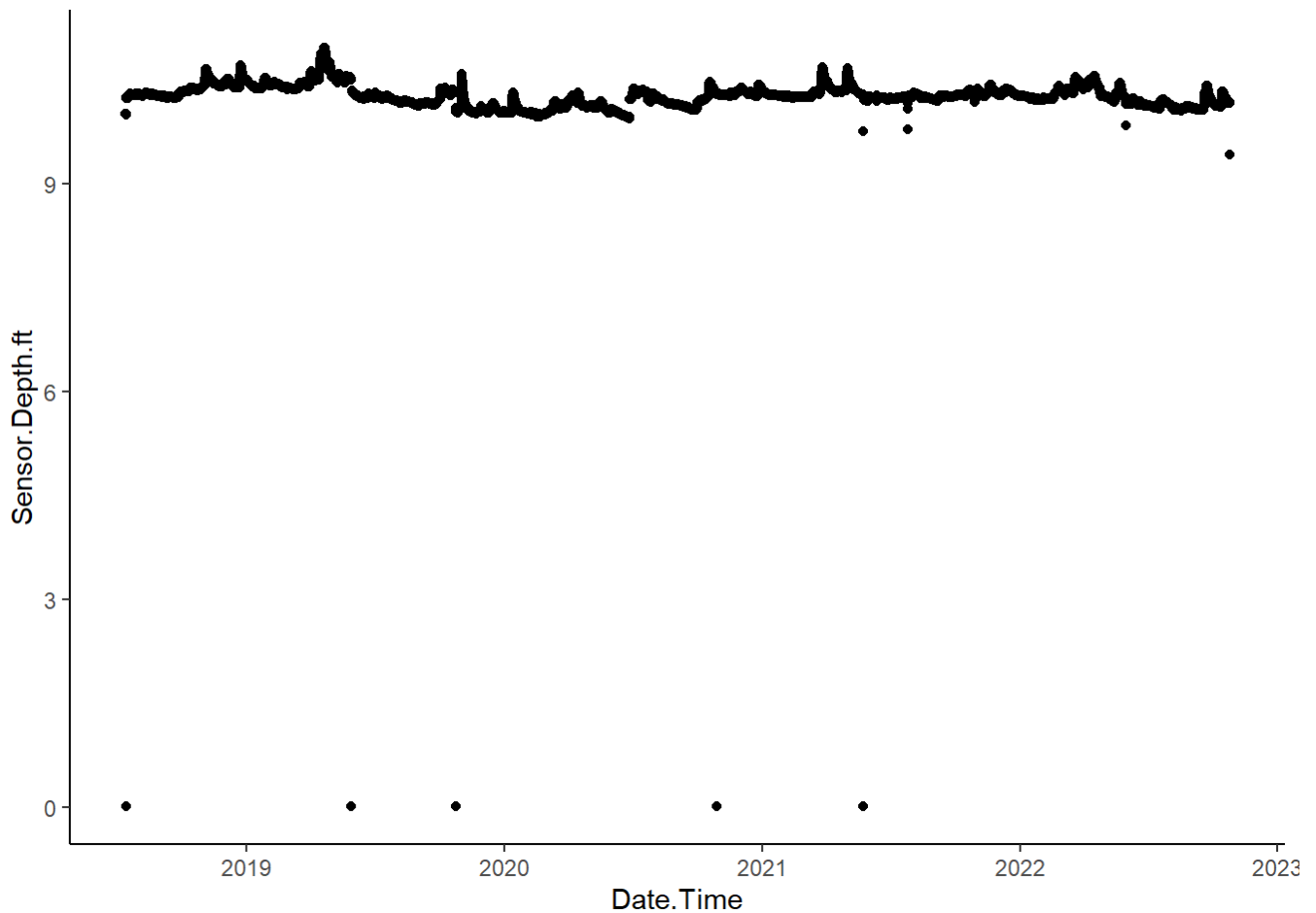
Following loading the data frame in, let’s get an idea as to what the data look like:

```

WaterLevel %>%
  ggplot(aes(x=Date.Time, y=Sensor.Depth.ft)) +
  geom_point() +
  theme_classic()

```

```
## Warning: Removed 17 rows containing missing values (`geom_point()`).
```



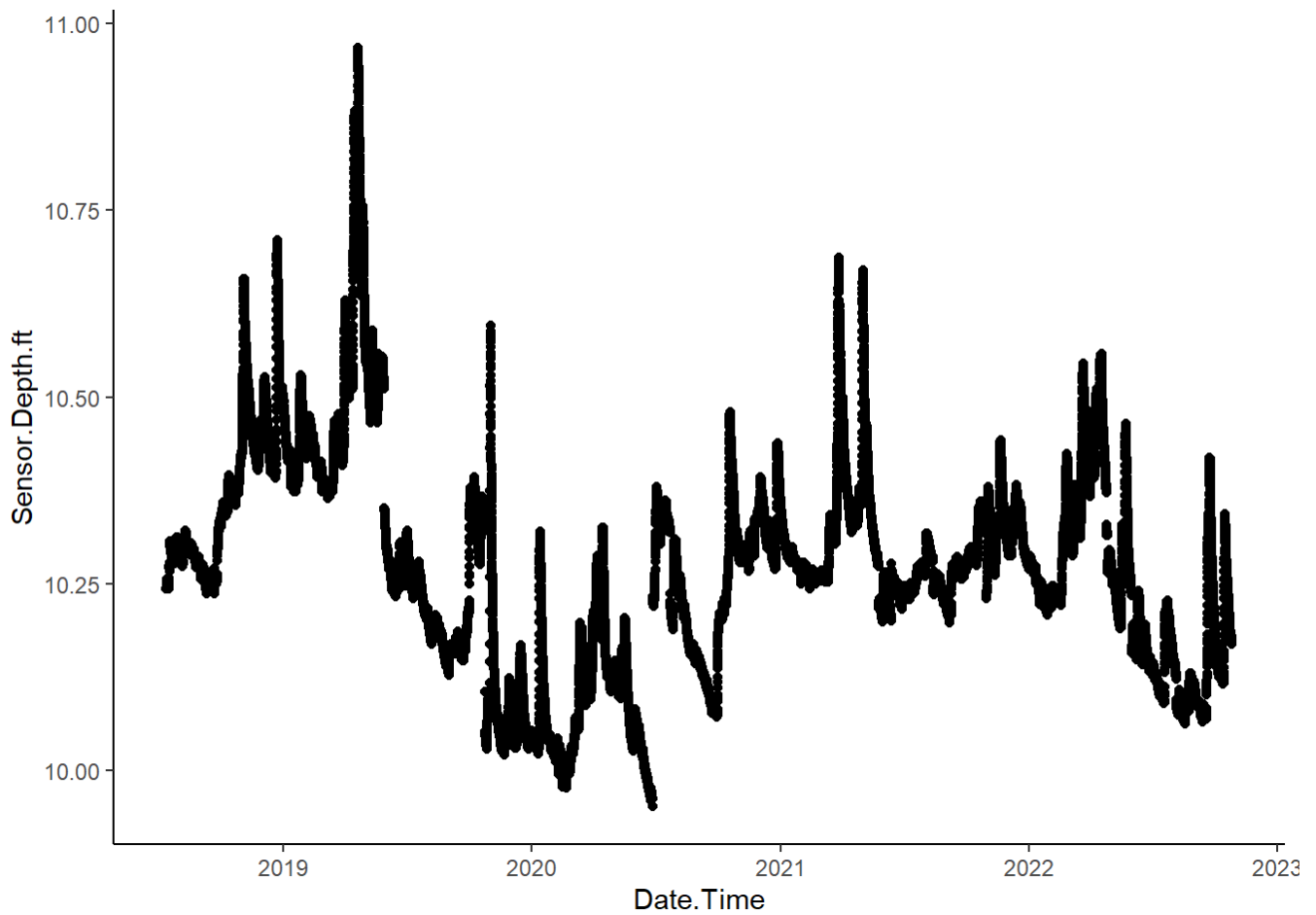
As you can see, there are some points that seem to be incorrect. The code below was designed after using ggplot and filter functions to identify years and sensor depths that are appropriate to use in eliminating the data points.

```
WaterLevel <- WaterLevel %>%
  filter(Sensor.Depth.ft > 9) %>%
  mutate(Year = year(Date.Time),
         Month = month(Date.Time)) %>%
  mutate(Remove = if_else(Year == 2018 & Sensor.Depth.ft < 10.2, "Yes", "No")) %>%
  mutate(Remove = if_else(Year == 2021 & Sensor.Depth.ft < 10.19, "Yes", Remove)) %>%
  mutate(Remove = if_else(Year == 2022 & Sensor.Depth.ft < 10, "Yes", Remove)) %>%
  filter(Remove == "No")

write.csv(WaterLevel, "./Holland/Data5_R_Files/HollandPond_WaterLevel.csv")
```

After cleaning up those data points, we can visualize Water level again:

```
WaterLevel %>%
  ggplot(aes(x=Date.Time, y=Sensor.Depth.ft)) +
  geom_point()+
  theme_classic()
```



## 6.11 Combining analysis from multiple lakes

After running this analysis on several different lakes and downloading the respective .csv files, you can combine dataframes in order to visualize how lakes are changing compared to one another.

For the sake of this analysis, you will see that I have three separate lakes to work with from the Vermont dataset.

## 6.11.1 Read in all .csv files

```
Holl_Strat <- read.csv("../Holland/Data5_R_Files/Holl_Strat.csv")
Long_Strat <- read.csv("../Long/Data5 - R_Files/Long_Strat.csv")
High_Strat <- read.csv("../High/Data5 - R_Files/High_Strat.csv")

## Load Ice off Data

Holl_Ice <- read.csv("../Holland/Data5_R_Files/Holl_Ice.csv")
Long_Ice <- read.csv("../Long/Data5 - R_Files/Long_Ice.csv")
High_Ice <- read.csv("../High/Data5 - R_Files/High_Ice.csv")

## Read in Epi Temp Anomalies

Holl_Epi_Summ <- read.csv("../Holland/Data5_R_Files/Holl_epi.summ.temp.csv") %>%
  mutate(Lake = "Holland")
High_Epi_Summ <- read.csv("../High/Data5 - R_Files/High_epi.summ.temp.csv") %>%
  mutate(Lake = "High")
Long_Epi_Summ <- read.csv("../Long/Data5 - R_Files/Long_epi.summ.temp.csv") %>%
  mutate(Lake="Long")

Holl_Epi_Wint <- read.csv("../Holland/Data5_R_Files/Holl_epi.wint.temp.csv") %>%
  mutate(Lake="Holland")
High_Epi_Wint <- read.csv("../High/Data5 - R_Files/High_epi.wint.temp.csv") %>%
  mutate(Lake = "High")
Long_Epi_Wint <- read.csv("../Long/Data5 - R_Files/Long_epi.wint.temp.csv") %>%
  mutate(Lake="Long")

# Reading in over all air trends and surface water temperature trends
Holl_epi_trend <- read_csv("../Holland/Data5_R_Files/Holl_Epi_Trend.csv")
High_epi_trend <- read_csv("../High/Data5 - R_Files/High_Epi_Trend.csv")
Long_epi_trend <- read_csv("../Long/Data5 - R_Files/Long_Epi_Trend.csv")

Holl_air_trend <- read_csv("../Holland/Data5_R_Files/Holl_Air_Trend.csv")
High_air_trend <- read_csv("../High/Data5 - R_Files/High_Air_Trend.csv")
Long_air_trend <- read_csv("../Long/Data5 - R_Files/Long_Air_Trend.csv")
```

After reading in all the different .csv files we created, I join together the dataframes, and make sure that there is a lake column in each dataframe.



```

##### Join data frames together

## Surface water temperatures

All_Epi <- full_join(Holl_epi_trend, High_epi_trend) %>%
  full_join(., Long_epi_trend)

All_Epi <- All_Epi %>%
  mutate(Lake = if_else(SiteID == "HOLLA--01M", "Holland", "Other"),
         Lake = if_else(SiteID == "HIGHS--02M", "High", Lake),
         Lake = if_else(SiteID == "LONGG--01M", "Long", Lake))

## Stratification Length

All_Strat <- full_join(Holl_Strat, Long_Strat, by=c("Lake", "Days", "Year")) %>%
  full_join(., High_Strat, by=c("Lake", "Days", "Year"))

## Length of ice cover

All_Ice <- full_join(Holl_Ice, Long_Ice, High_Ice, by=c("Lake", "Days", "Year")) %>%
  full_join(., High_Ice, by=c("Lake", "Days", "Year"))

## Summer epilimnion temperatures

Epi_Summ <- full_join(Holl_Epi_Summ, High_Epi_Summ, by=c("Year", "Lake", "Anomalies")) %>%
  full_join(., Long_Epi_Summ, by=c("Year", "Lake", "Anomalies")) %>%
  select(Year, Lake, Anomalies)

## Winter epilimnion temperatures

Epi_Wint <- full_join(Holl_Epi_Wint, High_Epi_Wint, by=c("Year", "Lake", "Anomalies")) %>%
  full_join(., Long_Epi_Wint, by=c("Year", "Lake", "Anomalies")) %>%
  select(Year, Lake, Anomalies)

## Air Trend Anomalies

All_Air <- full_join(Holl_air_trend, High_air_trend) %>%
  full_join(., Long_air_trend)

All_Air <- All_Air %>%
  mutate(Lake = if_else(SiteID == "HOLLA--10M", "Holland", "Other"),
         Lake = if_else(SiteID == "HIGHS--14M", "High", Lake),
         Lake = if_else(SiteID == "LONGG--09M", "Long", Lake))

# Join Epilimnion and air trend dataframes into one.

All_AirEpi <- full_join(All_Air, All_Epi, by=c("Lake", "RealDate"))

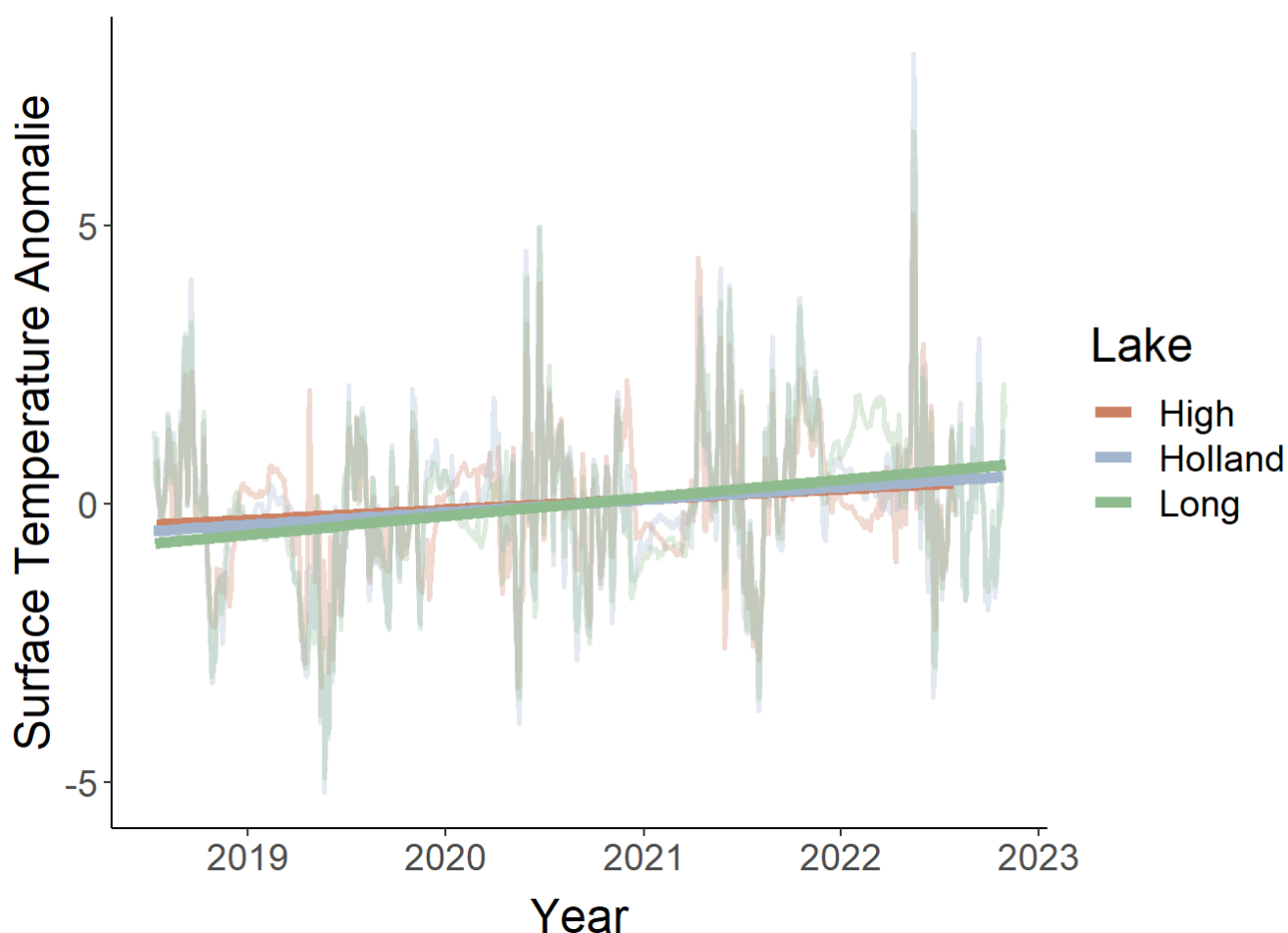
```

## 6.11.2 Making the figures!

After all of the files have been loaded in and the dataframes joined together, we can start to visualize the trends across all of the lakes. The following chunks of code are all going to be very similar.

### 6.11.3 Surface Water Trends

```
All_Epi %>%  
  ggplot(aes(x=RealDate, y=Temp.Diff, color=Lake)) +  
  geom_line(alpha=0.3,size=1) +  
  scale_colour_manual(values=c("lightsalmon3", "lightsteelblue3", "darkseagreen"))+  
  geom_smooth(method="lm", se=FALSE, size=2) +  
  ylab("Surface Temperature Anomalie") +  
  xlab("Year") +  
  theme_classic() +  
  theme(text = element_text(size = 18),  
        axis.title.x = element_text(vjust = -0.35))
```

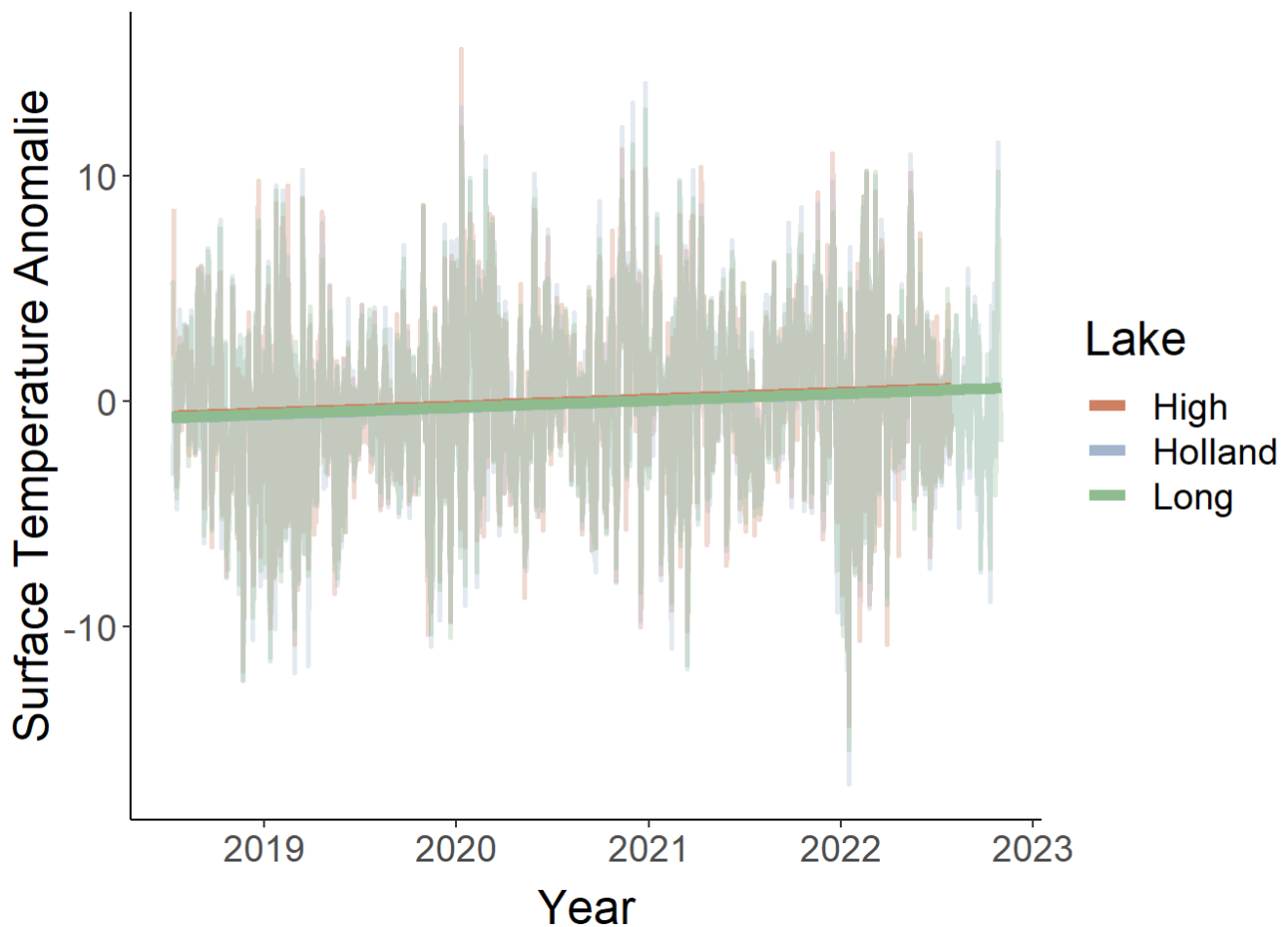


After graphing all three lakes together and applying a simple linear model via `geom_smooth`, we can see that on average, all of our lakes have experience warming surface waters over time. However, there tends to be some seasonality to this. Future work could run time series decompositions on this data when more years are added. However, I will dive into how we can explore this further by separating the winter temperatures and the summer temperature trends.

Also note here that Holland and Long Pond, which are located in the northeastern portion of Vermont have very similar trends, and High Pond, which is more southern, has a less pronounced slope. This means that there is likely some longitudinal relationship between warming in our lakes in Vermont.

## 6.11.4 Air Temperature Trends

```
All_Air %>%
  ggplot(aes(x=RealDate, y=Temp.Diff.Air, color=Lake)) +
  geom_line(alpha=0.3, size=1) +
  scale_colour_manual(values=c("lightsalmon3", "lightsteelblue3", "darkseagreen"))+
  geom_smooth(method="lm", se=FALSE, size=2) +
  ylab("Surface Temperature Anomalie") +
  xlab("Year") +
  theme_classic() +
  theme(text = element_text(size = 18),
        axis.title.x = element_text(vjust = -0.35))
```



While we only have 3-4 years of data collected on our lakes in Vermont so conclusions of climate can't necessarily be reached, we do have the ability to track the changes in air temperature and look at the relationship between surface water and air temperature.

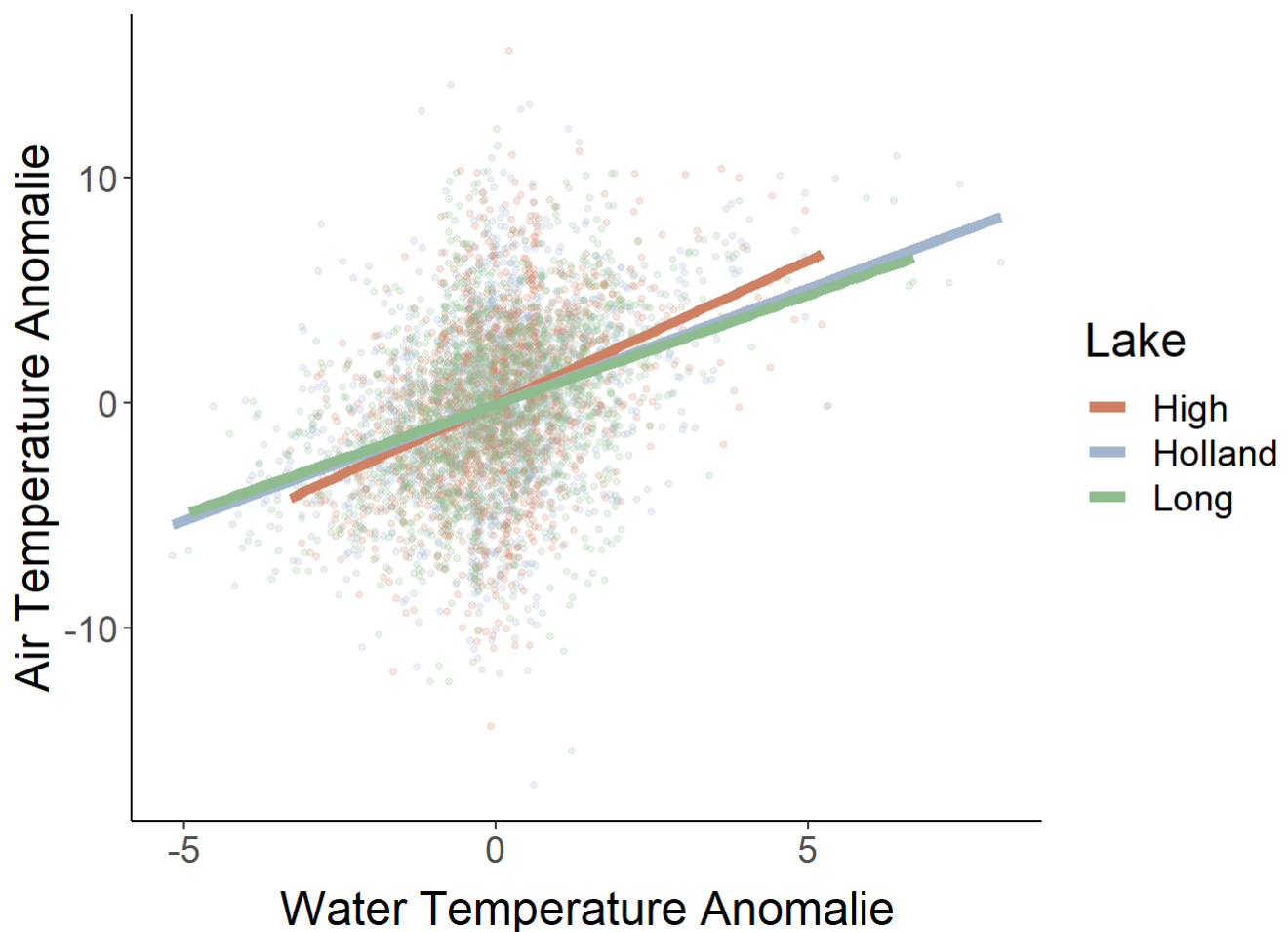
Here we can see that just like the surface waters in all of the lakes, after applying a linear model, the air temperatures are also increasing over time, however, there is much more noise in the data.

## 6.11.5 Air Temperature vs. Surface Water Temperature.

```
All_AirEpi %>%  
  ggplot(aes(x=Temp.Diff, y=Temp.Diff.Air, color=Lake)) +  
  geom_point(alpha=0.2,size=1) +  
  scale_colour_manual(values=c("lightsalmon3", "lightsteelblue3", "darkseagreen"))+  
  geom_smooth(method="lm", se=FALSE, size=2) +  
  ylab("Air Temperature Anomalie") +  
  xlab("Water Temperature Anomalie") +  
  theme_classic() +  
  theme(text = element_text(size = 18),  
        axis.title.x = element_text(vjust = -0.35))
```

```
## Warning: Removed 6 rows containing non-finite values (`stat_smooth()`).
```

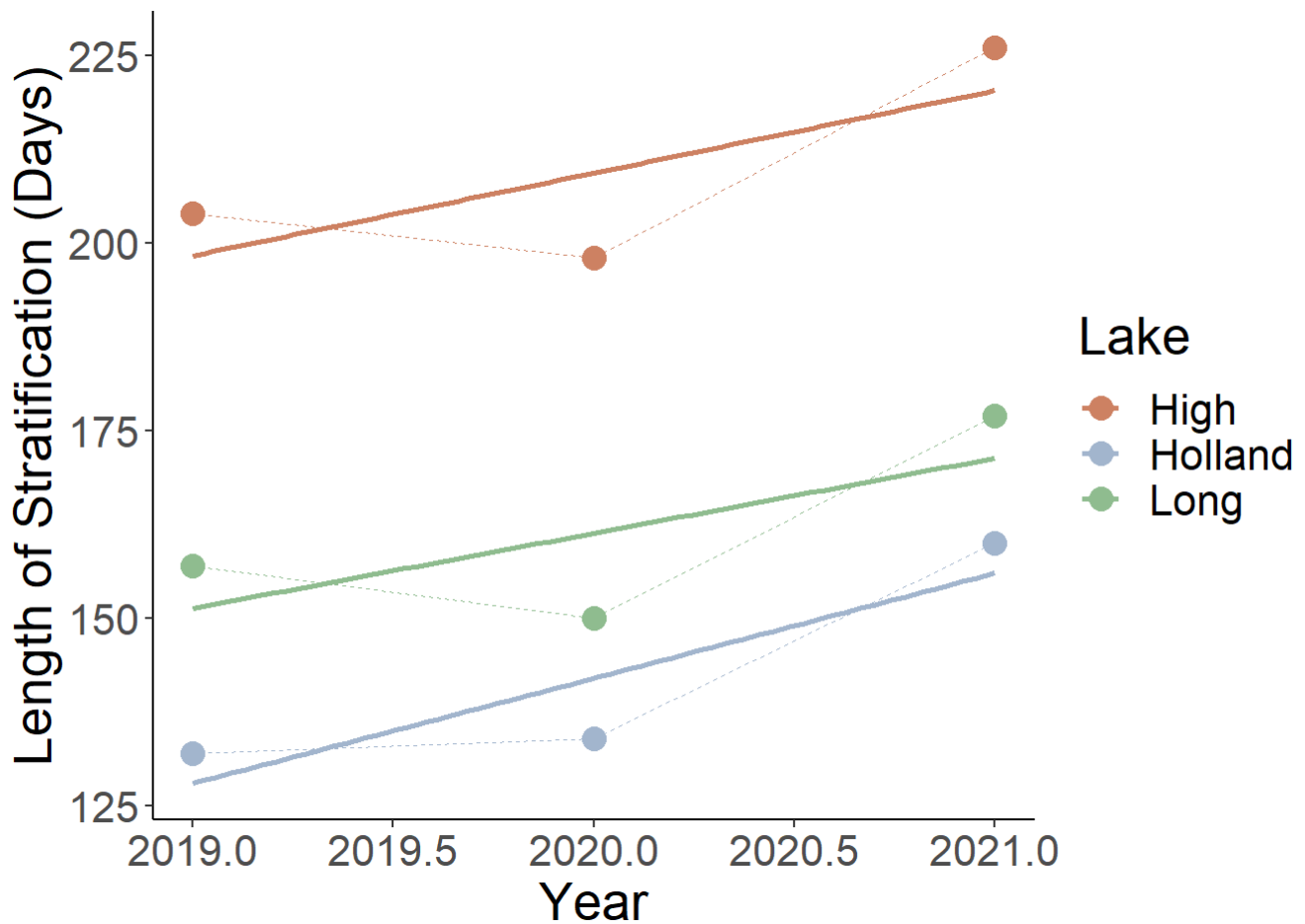
```
## Warning: Removed 6 rows containing missing values (`geom_point()`).
```



After running a simple linear model on the relationship between air temperature and surface temperatures, we again see a positive relationship, suggesting that if we continue to have increased air temperatures as a result of climate change, we can expect to see continued increases in lake surface water temperatures.

## 6.11.6 Length of Stratification

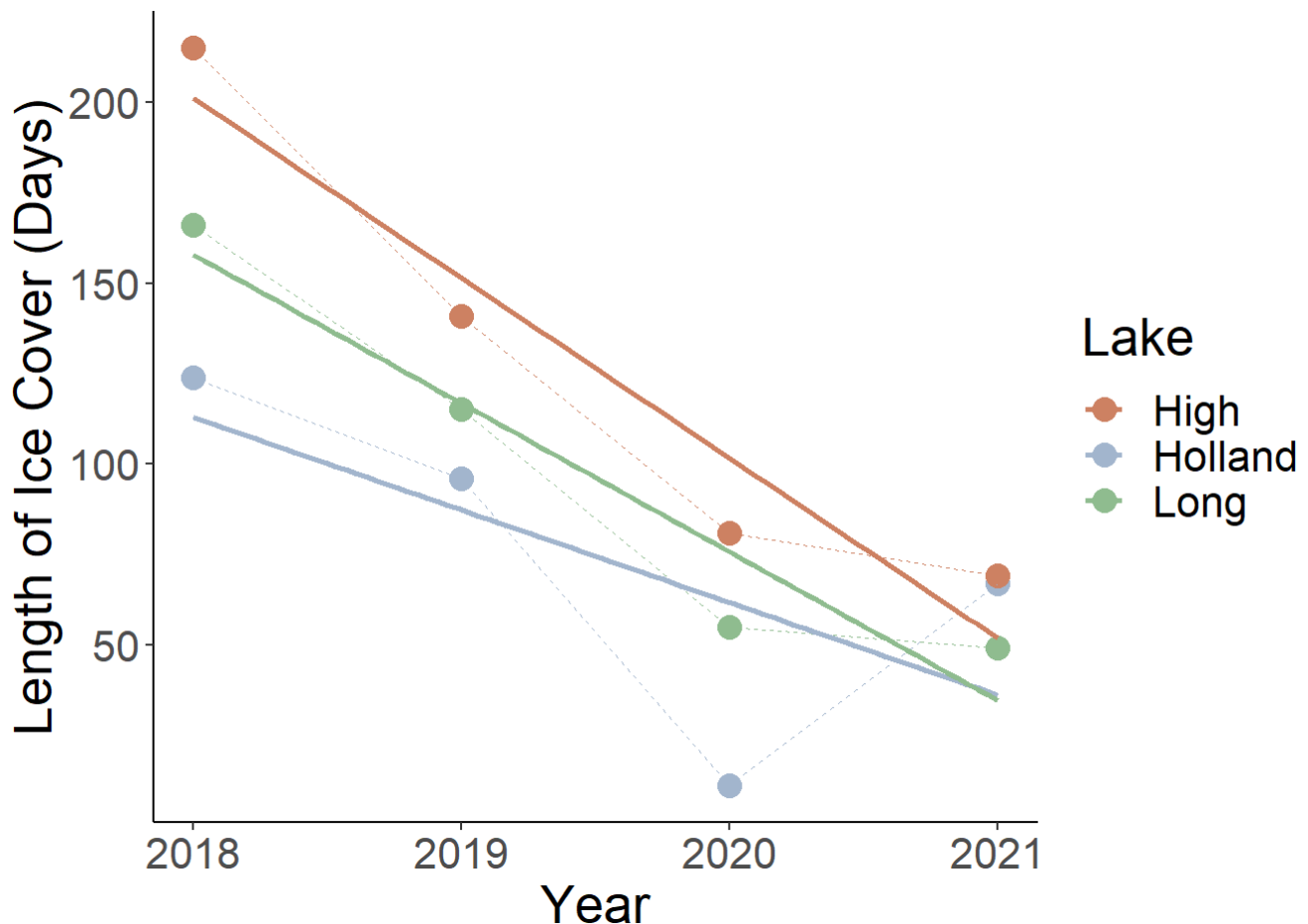
```
All_Strat %>%
  ggplot(aes(x=Year, y=Days, group=Lake, color=Lake)) +
  geom_point(size=4) +
  geom_line(size=0.3, linetype=2) +
  geom_smooth(method = "lm", se=FALSE) +
  scale_color_manual(values=c("lightsalmon3", "lightsteelblue3", "darkseagreen"))+
  theme_classic()+
  theme(text = element_text(size = 20)) +
  ylab("Length of Stratification (Days)") +
  xlab("Year")
```



When we look at the changes in lengths of stratification, we can see that all three lakes seem to experience the same pattern.

## 6.11.7 Length of the Ice Period

```
All_Ice %>%
  ggplot(aes(x=Year, y=Days, group=Lake, color=Lake)) +
  geom_point(size=4) +
  geom_line(size=0.3, linetype=2) +
  geom_smooth(method = "lm", se=FALSE) +
  scale_color_manual(values=c("lightsalmon3", "lightsteelblue3", "darkseagreen"))+
  theme_classic()+
  theme(text = element_text(size = 20)) +
  ylab("Length of Ice Cover (Days)") +
  xlab("Year")
```



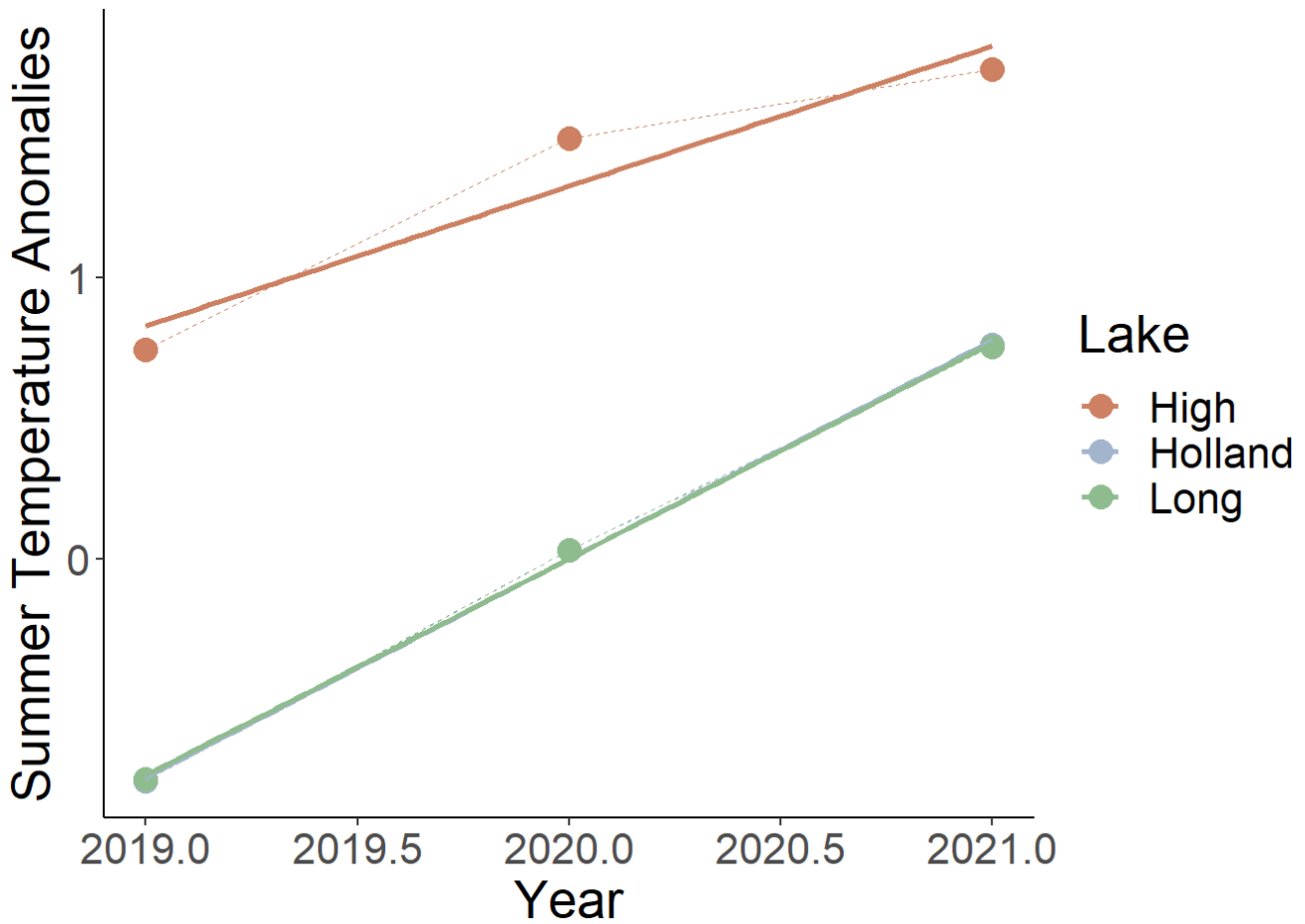
When looking at the length of ice cover, or the duration of time where there is inverse winter stratification, we can see that again all of the lakes display similar patterns, with the exception of Holland Pond, which had a longer duration of inverse stratification this past winter. In general, it appears that the length of ice cover is decreasing in all of the lakes.

## 6.11.8 Surface Temperature in Winter vs Summer.

Earlier we saw that there was seasonality in the trends of air temperature anomalies. I then decided to take the ice free and ice cover periods, and look at the surface water temperatures to see if their trends varied, or if we were seeing more warming in either period vs the other.

Let's start by looking at the summer.

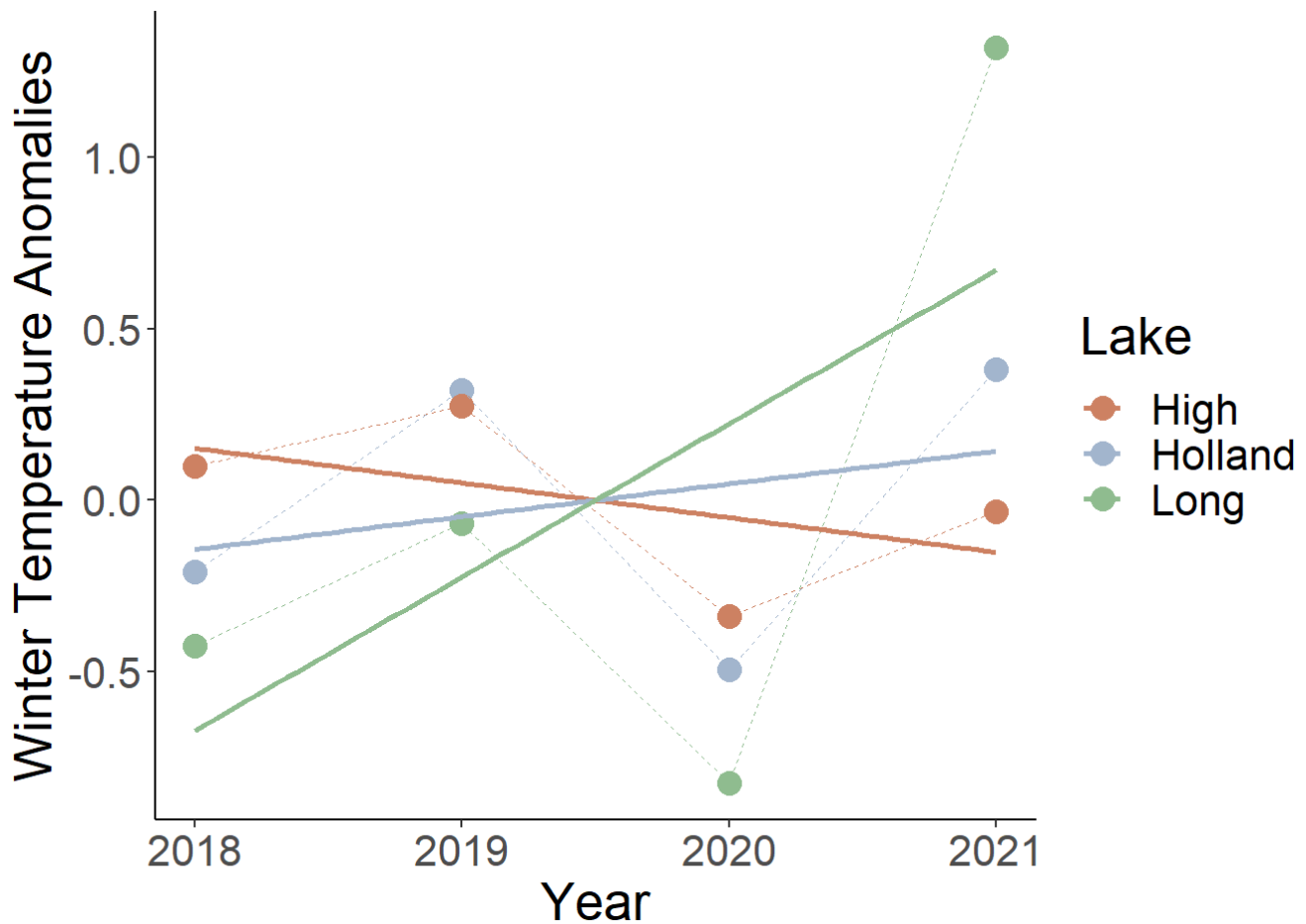
```
Epi_Summ %>%
  ggplot(aes(x=Year, y=Anomalies, group=Lake, color=Lake)) +
  geom_point(size=4) +
  geom_line(size=0.3, linetype=2) +
  geom_smooth(method = "lm", se=FALSE) +
  scale_color_manual(values=c("lightsalmon3", "lightsteelblue3", "darkseagreen"))+
  theme_classic()+
  theme(text = element_text(size = 20)) +
  ylab("Summer Temperature Anomalies") +
  xlab("Year")
```



We can see from this figure that Holland and Long have nearly the exact rates of warming, and High Pond, which is souther, seems to be warming more in the summer than the other two lakes.

Now what about the Winter?

```
Epi_Wint %>%
  ggplot(aes(x=Year, y=Anomalies, group=Lake, color=Lake)) +
  geom_point(size=4) +
  geom_line(size=0.3, linetype=2) +
  geom_smooth(method = "lm", se=FALSE) +
  scale_color_manual(values=c("lightsalmon3", "lightsteelblue3", "darkseagreen"))+
  theme_classic()+
  theme(text = element_text(size = 20)) +
  ylab("Winter Temperature Anomalies") +
  xlab("Year")
```



Here we see much more variable results. Long Pond is warming the most, while High Pond is actually getting colder. Holland Pond seems changing the least. In general we can see that this past winter, all lakes experienced much warmer water temperatures than the average.

## 6.12 Summary

In this guide we:

1. Ran QA/QC Checks on the thermal profile data from the RMN Project in a Vermont Lake
2. Learned how to make thermal heat maps, calculate schmidt stability and thermocline depth, create 'anomalies' to understand trends in lakes, and calculated stratification length and ice period length
3. Also learned how to combine dataframes in order to visualize trends across multiple lakes.



I hope this guide can be helpful in developing code to process data from the RMN Project!