PreparingTempData

Amaryllis Adey

2024-06-13

# Goals

1. To create a single file document that has all the data for the pond temperature study together in one place
2. To ensure that this format is well setup for use in the analysis code that will be adapted from ’Exercise\_06\_StateSpace.Rmd

Code from the linear regression models Created by Luca Adelfio, Rachel Hughes, Amaryllis Adey, and Erik Curtis Adapted here to fit code from Alyssa

## Set up the packages to be used throughout this code

library(ggplot2) #for plotting  
library(plyr) #for data organization  
library(reshape2) #for data organization  
library(zoo) #for rmse, data interpolation

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

#library("zoo") #for interpolating to fill data gaps  
library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.5  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ lubridate 1.9.3 ✔ tibble 3.2.1  
## ✔ purrr 1.0.2 ✔ tidyr 1.3.1

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::arrange() masks plyr::arrange()  
## ✖ purrr::compact() masks plyr::compact()  
## ✖ dplyr::count() masks plyr::count()  
## ✖ dplyr::desc() masks plyr::desc()  
## ✖ dplyr::failwith() masks plyr::failwith()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::id() masks plyr::id()  
## ✖ dplyr::lag() masks stats::lag()  
## ✖ dplyr::mutate() masks plyr::mutate()  
## ✖ dplyr::rename() masks plyr::rename()  
## ✖ dplyr::summarise() masks plyr::summarise()  
## ✖ dplyr::summarize() masks plyr::summarize()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(ggpubr)#for arranging plots

##   
## Attaching package: 'ggpubr'  
##   
## The following object is masked from 'package:plyr':  
##   
## mutate

library(base)  
library(lubridate)  
library(ModelMetrics)

##   
## Attaching package: 'ModelMetrics'  
##   
## The following object is masked from 'package:base':  
##   
## kappa

library(dplyr)

## Set up the working directory

#remove all existing variables  
rm(list=ls())  
  
#Get working directory  
this.dir = dirname(rstudioapi::getActiveDocumentContext()$path)  
setwd(this.dir)  
  
plot.folder<- paste(this.dir,"/Plots/",sep = "") #address for exported plots

# Part 1 - Comparing Airport Data Between CRD and YF

## Load in the data and correct issues found initially

met1<-read.csv(paste(this.dir,"/DataFiles/ClimateData/2001\_2019\_CDV\_YAK/daily\_wx\_data.csv",sep=""),header=T, na.strings=-9999)  
#if file is in the iCloud drive and not downloaded, the line above will not work. Made sure file is downloaded.  
#also this needs to be updated through 2020  
met1$DATE=as.Date(met1$DATE, format="%m/%d/%Y")  
summary(met1)

## STATION NAME LATITUDE LONGITUDE   
## Length:24053 Length:24053 Min. :59.51 Min. :-145.8   
## Class :character Class :character 1st Qu.:59.51 1st Qu.:-145.8   
## Mode :character Mode :character Median :60.49 Median :-145.5   
## Mean :60.23 Mean :-143.9   
## 3rd Qu.:60.54 3rd Qu.:-139.7   
## Max. :60.56 Max. :-139.7   
##   
## ELEVATION DATE AWND PRCP   
## Min. : 3.000 Min. :2001-01-01 Min. : 0.000 Min. : 0.000   
## 1st Qu.: 7.600 1st Qu.:2006-07-29 1st Qu.: 0.900 1st Qu.: 0.000   
## Median : 9.400 Median :2011-05-29 Median : 1.600 Median : 1.300   
## Mean : 8.777 Mean :2011-03-04 Mean : 1.964 Mean : 9.153   
## 3rd Qu.:10.100 3rd Qu.:2016-01-10 3rd Qu.: 2.700 3rd Qu.: 10.400   
## Max. :25.300 Max. :2019-12-31 Max. :10.300 Max. :297.200   
## NA's :10237 NA's :412   
## SNOW SNWD TAVG TMAX   
## Min. : 0.000 Min. : 0.00 Min. :-17.800 Min. :-16.100   
## 1st Qu.: 0.000 1st Qu.: 0.00 1st Qu.: 1.700 1st Qu.: 4.400   
## Median : 0.000 Median : 0.00 Median : 5.700 Median : 8.900   
## Mean : 7.913 Mean : 91.55 Mean : 5.627 Mean : 8.883   
## 3rd Qu.: 0.000 3rd Qu.: 51.00 3rd Qu.: 10.825 3rd Qu.: 13.900   
## Max. :635.000 Max. :2438.00 Max. : 32.200 Max. : 32.200   
## NA's :9628 NA's :9575 NA's :18269 NA's :451   
## TMIN TSUN   
## Min. :-26.100 Min. :0   
## 1st Qu.: -2.200 1st Qu.:0   
## Median : 2.200 Median :0   
## Mean : 1.841 Mean :0   
## 3rd Qu.: 7.200 3rd Qu.:0   
## Max. : 20.000 Max. :0   
## NA's :459 NA's :22824

#fix erroneous Feb 28, 2012 data with known correct value  
met1[met1$DATE=="2012-02-28" & met1$STATION== "USW00026410" ,] #

## STATION NAME LATITUDE LONGITUDE ELEVATION DATE  
## 4076 USW00026410 CORDOVA AIRPORT, AK US 60.4888 -145.4511 9.4 2012-02-28  
## AWND PRCP SNOW SNWD TAVG TMAX TMIN TSUN  
## 4076 0.5 0.3 NA NA NA 31.1 -7.2 NA

met1$TMAX[met1$DATE=="2012-02-28" & met1$STATION== "USW00026410" ]<-2.2 #note this fix number comes from computing hrly data with mh.all code below LA 1/16/16  
met1[met1$DATE=="2012-02-28" & met1$STATION== "USW00026410" ,] #

## STATION NAME LATITUDE LONGITUDE ELEVATION DATE  
## 4076 USW00026410 CORDOVA AIRPORT, AK US 60.4888 -145.4511 9.4 2012-02-28  
## AWND PRCP SNOW SNWD TAVG TMAX TMIN TSUN  
## 4076 0.5 0.3 NA NA NA 2.2 -7.2 NA

## Find average temperature

met1$TAVG2<-(met1$TMAX+met1$TMIN)/2 #\*[This isn't the average though? - Erik]  
clim.period<-seq(as.Date("2001-01-01"),as.Date("2019-12-31"), by="day") #change to 1/1/1991 and 12/31/2021 when data are available  
clim.df<-data.frame("STATION"=rep(unique(met1$STATION),each=length(clim.period)),"DATE"=rep(clim.period, times= length(unique(met1$STATION))))  
met2<-merge(clim.df,met1, by=c("STATION","DATE"), all.x=T)  
met2$NAME<-as.character(met2$NAME)  
met2$NAME<-factor(substr(met2$NAME,1,nchar(met2$NAME)-7))  
summary(met2)

## STATION DATE NAME   
## Length:34695 Min. :2001-01-01 CORDOVA 14 ESE : 841   
## Class :character 1st Qu.:2005-10-01 CORDOVA AIRPORT: 6934   
## Mode :character Median :2010-07-02 CORDOVA NORTH : 5778   
## Mean :2010-07-02 CORDOVA WWTP : 3565   
## 3rd Qu.:2015-04-02 YAKUTAT AIRPORT: 6935   
## Max. :2019-12-31 NA's :10642   
##   
## LATITUDE LONGITUDE ELEVATION AWND   
## Min. :59.51 Min. :-145.8 Min. : 3.000 Min. : 0.000   
## 1st Qu.:59.51 1st Qu.:-145.8 1st Qu.: 7.600 1st Qu.: 0.900   
## Median :60.49 Median :-145.5 Median : 9.400 Median : 1.600   
## Mean :60.23 Mean :-143.9 Mean : 8.777 Mean : 1.964   
## 3rd Qu.:60.54 3rd Qu.:-139.7 3rd Qu.:10.100 3rd Qu.: 2.700   
## Max. :60.56 Max. :-139.7 Max. :25.300 Max. :10.300   
## NA's :10642 NA's :10642 NA's :10642 NA's :20879   
## PRCP SNOW SNWD TAVG   
## Min. : 0.000 Min. : 0.000 Min. : 0.00 Min. :-17.800   
## 1st Qu.: 0.000 1st Qu.: 0.000 1st Qu.: 0.00 1st Qu.: 1.700   
## Median : 1.300 Median : 0.000 Median : 0.00 Median : 5.700   
## Mean : 9.153 Mean : 7.913 Mean : 91.55 Mean : 5.627   
## 3rd Qu.: 10.400 3rd Qu.: 0.000 3rd Qu.: 51.00 3rd Qu.: 10.825   
## Max. :297.200 Max. :635.000 Max. :2438.00 Max. : 32.200   
## NA's :11054 NA's :20270 NA's :20217 NA's :28911   
## TMAX TMIN TSUN TAVG2   
## Min. :-16.100 Min. :-26.100 Min. :0 Min. :-20.550   
## 1st Qu.: 4.400 1st Qu.: -2.200 1st Qu.:0 1st Qu.: 1.150   
## Median : 8.900 Median : 2.200 Median :0 Median : 5.300   
## Mean : 8.882 Mean : 1.842 Mean :0 Mean : 5.361   
## 3rd Qu.: 13.900 3rd Qu.: 7.200 3rd Qu.:0 3rd Qu.: 10.550   
## Max. : 32.200 Max. : 20.000 Max. :0 Max. : 23.050   
## NA's :11093 NA's :11101 NA's :33466 NA's :11104

## Subset for period of water temperatures from CRD and YF

met2b<-subset(met2, NAME %in% c("CORDOVA AIRPORT","YAKUTAT AIRPORT") & DATE >= as.Date("2012-01-01"))  
met2b$NAME<-factor(met2b$NAME)  
range(met2b$DATE)

## [1] "2012-01-01" "2019-12-31"

#Plot Cordova vs. Yakutat air temps  
ggplot(data=met2b, aes(x=DATE))+  
 geom\_hline(yintercept=0,lwd=.5,lty=2)+  
 geom\_ribbon(aes(ymin=TMIN,ymax=TMAX), fill="azure3",alpha=.6)+  
 geom\_line(aes(y=TMAX), color="red",linewidth=1)+  
 geom\_line(aes(y=TMIN), color="blue",linewidth=1)+  
 geom\_line(aes(y=TAVG2), color="black",linewidth=1)+  
 xlab("Date")+  
 ylab(expression(paste("Air temperature ( ", degree ~C, " )")))+  
 facet\_wrap(~NAME,ncol=1)+  
 theme\_minimal()

#export plot  
ggsave("air\_temps.png", path=plot.folder, width=6.5, height=4, units="in",dpi=300)

## Calculating month-year averages for the weather data

LA wrote this for wkyr, RH changed to monyr –> AA calculating by monthly averages through time

met2b$monyr<-format(met2b$DATE, "%m-%Y") #add month-year column  
  
met3 <- met2b %>%  
 group\_by(NAME, monyr) %>%  
 summarise(  
 Amonavg = round(mean(TAVG2, na.rm = TRUE), 2),  
 Amonmax = round(max(TMAX, na.rm = TRUE), 2),  
 Amonmin = round(min(TMIN, na.rm = TRUE), 2)  
 ) %>%  
 ungroup()

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

#Regress Cordova vs. Yakutat   
lm1<-merge(subset(met2b, NAME == "CORDOVA AIRPORT", select=c(DATE,TAVG2)), subset(met2b, NAME == "YAKUTAT AIRPORT",select=c(DATE,TAVG2)), by="DATE", all=T)  
colnames(lm1)<-c("DATE","CORDOVA","YAKUTAT")  
summary(lm(YAKUTAT~CORDOVA,data=lm1))

##   
## Call:  
## lm(formula = YAKUTAT ~ CORDOVA, data = lm1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -13.1849 -1.0461 0.0702 1.1214 10.3719   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.158464 0.045027 25.73 <2e-16 \*\*\*  
## CORDOVA 0.897941 0.005528 162.45 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.944 on 2898 degrees of freedom  
## (22 observations deleted due to missingness)  
## Multiple R-squared: 0.901, Adjusted R-squared: 0.901   
## F-statistic: 2.639e+04 on 1 and 2898 DF, p-value: < 2.2e-16

ggplot(data=lm1, aes(x=CORDOVA,y=YAKUTAT))+  
 geom\_abline(slope=1, intercept=0, color="gray50", lty=2)+  
 geom\_point()+  
 geom\_smooth(method="lm", color="blue")+  
 xlab("Mean daily air temperature, Cordova Airport")+  
 ylab("Mean daily air temperature, Yakutat Airport")+  
 theme\_minimal()

## `geom\_smooth()` using formula = 'y ~ x'

## Warning: Removed 22 rows containing non-finite outside the scale range  
## (`stat\_smooth()`).

## Warning: Removed 22 rows containing missing values or values outside the scale range  
## (`geom\_point()`).

ggsave("YakCorReg.png", path=plot.folder, width=6.5, height=4, units="in",dpi=300)

## `geom\_smooth()` using formula = 'y ~ x'

## Warning: Removed 22 rows containing non-finite outside the scale range  
## (`stat\_smooth()`).  
## Removed 22 rows containing missing values or values outside the scale range  
## (`geom\_point()`).

#Air temperature stats Cordova vs. Yakutat  
lapply(unique(met2b$NAME), function(x) summary(met2b[met2b$NAME==x,]))

## [[1]]  
## STATION DATE NAME LATITUDE   
## Length:2918 Min. :2012-01-01 CORDOVA AIRPORT: 0 Min. :59.51   
## Class :character 1st Qu.:2013-12-30 YAKUTAT AIRPORT:2918 1st Qu.:59.51   
## Mode :character Median :2015-12-29 Median :59.51   
## Mean :2015-12-30 Mean :59.51   
## 3rd Qu.:2017-12-27 3rd Qu.:59.51   
## Max. :2019-12-31 Max. :59.51   
##   
## LONGITUDE ELEVATION AWND PRCP   
## Min. :-139.7 Min. :10.1 Min. :0.000 Min. : 0.000   
## 1st Qu.:-139.7 1st Qu.:10.1 1st Qu.:1.100 1st Qu.: 0.000   
## Median :-139.7 Median :10.1 Median :1.900 Median : 1.800   
## Mean :-139.7 Mean :10.1 Mean :2.145 Mean : 8.846   
## 3rd Qu.:-139.7 3rd Qu.:10.1 3rd Qu.:2.900 3rd Qu.: 11.400   
## Max. :-139.7 Max. :10.1 Max. :9.100 Max. :128.800   
## NA's :46 NA's :21   
## SNOW SNWD TAVG TMAX   
## Min. : 0.00 Min. : 0.0 Min. :-12.900 Min. :-10.600   
## 1st Qu.: 0.00 1st Qu.: 0.0 1st Qu.: 1.800 1st Qu.: 4.400   
## Median : 0.00 Median : 0.0 Median : 6.350 Median : 9.400   
## Mean : 7.95 Mean : 125.1 Mean : 6.126 Mean : 9.209   
## 3rd Qu.: 0.00 3rd Qu.: 76.0 3rd Qu.: 11.400 3rd Qu.: 13.900   
## Max. :343.00 Max. :2438.0 Max. : 19.900 Max. : 30.600   
## NA's :434 NA's :545 NA's :468 NA's :17   
## TMIN TSUN TAVG2 monyr   
## Min. :-23.900 Min. : NA Min. :-16.150 Length:2918   
## 1st Qu.: -2.100 1st Qu.: NA 1st Qu.: 1.400 Class :character   
## Median : 1.700 Median : NA Median : 5.550 Mode :character   
## Mean : 1.853 Mean :NaN Mean : 5.531   
## 3rd Qu.: 7.200 3rd Qu.: NA 3rd Qu.: 10.550   
## Max. : 15.000 Max. : NA Max. : 20.000   
## NA's :17 NA's :2918 NA's :17   
##   
## [[2]]  
## STATION DATE NAME LATITUDE   
## Length:2917 Min. :2012-01-01 CORDOVA AIRPORT:2917 Min. :60.49   
## Class :character 1st Qu.:2013-12-30 YAKUTAT AIRPORT: 0 1st Qu.:60.49   
## Mode :character Median :2015-12-29 Median :60.49   
## Mean :2015-12-29 Mean :60.49   
## 3rd Qu.:2017-12-27 3rd Qu.:60.49   
## Max. :2019-12-31 Max. :60.49   
##   
## LONGITUDE ELEVATION AWND PRCP   
## Min. :-145.5 Min. :9.4 Min. :0.00 Min. : 0.000   
## 1st Qu.:-145.5 1st Qu.:9.4 1st Qu.:0.90 1st Qu.: 0.000   
## Median :-145.5 Median :9.4 Median :1.50 Median : 1.000   
## Mean :-145.5 Mean :9.4 Mean :1.84 Mean : 6.402   
## 3rd Qu.:-145.5 3rd Qu.:9.4 3rd Qu.:2.50 3rd Qu.: 7.600   
## Max. :-145.5 Max. :9.4 Max. :9.50 Max. :133.600   
## NA's :4   
## SNOW SNWD TAVG TMAX   
## Min. : 0.00 Min. : 0.00 Min. : NA Min. :-14.300   
## 1st Qu.: 0.00 1st Qu.: 0.00 1st Qu.: NA 1st Qu.: 4.400   
## Median : 0.00 Median : 51.00 Median : NA Median : 8.900   
## Mean : 17.42 Mean : 75.22 Mean :NaN Mean : 9.047   
## 3rd Qu.: 18.00 3rd Qu.:127.00 3rd Qu.: NA 3rd Qu.: 13.900   
## Max. :145.00 Max. :305.00 Max. : NA Max. : 32.200   
## NA's :2839 NA's :2840 NA's :2917   
## TMIN TSUN TAVG2 monyr   
## Min. :-23.9000 Min. : NA Min. :-18.050 Length:2917   
## 1st Qu.: -3.8000 1st Qu.: NA 1st Qu.: 0.850 Class :character   
## Median : 1.1000 Median : NA Median : 5.250 Mode :character   
## Mean : 0.6283 Mean :NaN Mean : 4.838   
## 3rd Qu.: 6.1000 3rd Qu.: NA 3rd Qu.: 10.250   
## Max. : 13.3000 Max. : NA Max. : 20.250   
## NA's :2917

Here we found that there is a high correlation between the two airport datasets (I am unsure why we need this exactly??) But R2 = 0.901 which is great!

# Part 2 - Comparing Logger Data to the Airport Data

## Yakutat Forelands

### Pull in the Yakutat logger air temeprature from the ponds

getwd()

## [1] "/Users/amaryllisadey/Dropbox/SWEL/CopperRiverDelta/PondTemps/DataAnalysis/PondTempsAnalysis\_GIT"

dir1<-paste0(this.dir,"/DataFiles/PondData/Yakutat/Air/")  
yktdat<-list.files(dir1)  
  
#lapply(paste0(dir1,yktdat),function(x){ncol(read.csv(x, skip=1))}) #ID a pesky data.frame with too many columns! fixed now.  
  
sitekey<-c("MP1", "MP3","MP5","MP8","PL1","PL2","PL3","UBP1","UBP2", "UBP3","UBP4")  
yktdat

## [1] "MP1\_Air\_2012\_09\_26-2013\_06\_12\_10144206.csv"   
## [2] "MP1\_Air\_2013\_06\_27-2013\_10\_28\_10144206\_QAQC.csv"   
## [3] "MP1\_Air\_2013\_10\_28-2014\_06\_02\_10144206\_QAQC.csv"   
## [4] "MP1\_Air\_2014\_06\_02-2014\_09\_11\_10144206\_QAQC.csv"   
## [5] "MP1\_Air\_2016\_10\_13-2017\_05\_23\_10402259\_QAQC.csv"   
## [6] "MP1\_Air\_2018\_05\_25-2018\_10\_11\_10402261\_QAQC.csv"   
## [7] "MP1\_Air\_2018\_10\_11-2019\_06\_06\_10402261\_QAQC.csv"   
## [8] "MP1\_Air\_2019\_06\_06-2019\_10\_16\_10402261\_QAQC.csv"   
## [9] "MP3\_Air\_2012\_09\_25-2013\_06\_12\_10144200.csv"   
## [10] "MP3\_Air\_2013\_06\_27-2013\_10\_28\_10144200.csv"   
## [11] "MP3\_Air\_2013\_10\_28-2014\_06\_02\_10144200\_QAQC.csv"   
## [12] "MP3\_Air\_2014\_06\_02-2014\_09\_11\_10144200\_QAQC.csv"   
## [13] "MP3\_Air\_2018\_05\_25-2018\_10\_11\_11003081\_QAQC.csv"   
## [14] "MP3\_Air\_2018\_10\_11-2019\_06\_06\_11003081\_QAQC.csv"   
## [15] "MP3\_Air\_2019\_06\_06-2019\_10\_01\_11003081\_QAQC.csv"   
## [16] "MP5\_Air\_2012\_09\_25-2013\_06\_27\_10161551\_QAQC.csv"   
## [17] "MP5\_Air\_2013\_06\_27-2013\_10\_28\_10161551\_QAQC.csv"   
## [18] "MP5\_Air\_2013\_10\_28-2014\_06\_02\_10161551\_QAQC.csv"   
## [19] "MP5\_Air\_2014\_06\_02-2014\_09\_11\_10161551\_QAQC.csv"   
## [20] "MP5\_Air\_2016\_10\_04-2017\_05\_24\_10402255\_QAQC.csv"   
## [21] "MP5\_Air\_2018\_06\_21-2019\_03\_08\_10144204\_QAQC.csv"   
## [22] "MP5\_Air\_2019\_06\_06-2019\_10\_16\_10144204\_QAQC.csv"   
## [23] "MP8\_Air\_2012\_09\_24-2013\_06\_12\_9922072\_QAQC.csv"   
## [24] "MP8\_Air\_2013\_06\_27-2014\_03\_15\_9922072\_QAQC.csv"   
## [25] "MP8\_Air\_2014\_09\_11-2015\_06\_04\_10402266\_QAQC.csv"   
## [26] "MP8\_Air\_2015\_09\_30-2015\_10\_05\_10402266\_QAQC.csv"   
## [27] "MP8\_Air\_2016\_04\_14-2016\_04\_16\_10402266.csv"   
## [28] "MP8\_Air\_2018\_05\_25-2018\_10\_11\_10402255\_QAQC.csv"   
## [29] "MP8\_Air\_2018\_10\_11-2019\_06\_06\_10402255\_QAQC.csv"   
## [30] "PL1\_Air\_2012\_11\_08-2013\_10\_24\_10201439.csv"   
## [31] "PL1\_Air\_2013\_10\_24-2014\_06\_10\_10201439\_QAQC.csv"   
## [32] "PL1\_Air\_2014\_06\_10-2015\_06\_11\_10201439\_QAQC.csv"   
## [33] "PL1\_Air\_2015\_06\_11-2016\_05\_20\_10201439\_QAQC.csv"   
## [34] "PL1\_Air\_2018\_10\_17-2019\_06\_07\_10402263\_QAQC.csv"   
## [35] "PL1\_Air\_2019\_06\_07-2019\_10\_17\_10402263\_QAQC.csv"   
## [36] "PL2\_Air\_2012\_11\_08-2013\_10\_24\_10201436\_QAQC.csv"   
## [37] "PL2\_Air\_2013\_10\_24\_10-2014\_06\_10\_10201436\_QAQC.csv"  
## [38] "PL2\_Air\_2015\_06\_11-2016\_05\_20\_10201436\_QAQC.csv"   
## [39] "PL2\_Air\_2018\_10\_17-2019\_06\_07\_11003083\_QAQC.csv"   
## [40] "PL2\_Air\_2019\_06\_07-2019\_10\_17\_11003083\_QAQC.csv"   
## [41] "PL3\_Air\_2012\_11\_08-2013\_10\_24\_10201454\_QAQC.csv"   
## [42] "PL3\_Air\_2013\_10\_24-2014\_06\_10\_10201454\_QAQC.csv"   
## [43] "PL3\_Air\_2014\_06\_10-2015\_06\_11\_10201454\_QAQC.csv"   
## [44] "PL3\_Air\_2015\_05\_20-2016\_05\_20\_10201454\_QAQC.csv"   
## [45] "PL3\_Air\_2018\_10\_17-2019\_06\_07\_10402258\_QAQC.csv"   
## [46] "PL3\_Air\_2019\_06\_07-2019\_10\_17\_10402258\_QAQC.csv"   
## [47] "UBP1\_Air\_2018\_10\_16-2019\_05\_09\_10201439\_QAQC.csv"   
## [48] "UBP1\_Air\_2019\_05\_09-2019\_10\_28\_10201439\_QAQC.csv"   
## [49] "UBP2\_Air\_2012\_10\_25-2013\_06\_12\_10144202\_QAQC.csv"   
## [50] "UBP2\_Air\_2013\_06\_26-2013\_10\_22\_10144202\_QAQC.csv"   
## [51] "UBP2\_Air\_2013\_10\_22-2014\_06\_03\_10144202\_QAQC.csv"   
## [52] "UBP2\_Air\_2018\_10\_16-2019\_05\_09\_10402266\_QAQC.csv"   
## [53] "UBP3\_Air\_2012\_10\_25-2013\_03\_28\_10140253\_QAQC.csv"   
## [54] "UBP4\_Air\_2012\_10\_25\_2013\_06\_12\_10144205\_QAQC.csv"   
## [55] "UBP4\_Air\_2013\_06\_26\_2013\_10\_22\_10144205\_QAQC.csv"   
## [56] "UBP4\_Air\_2013\_10\_22-2014\_06\_03\_10144205\_QAQC.csv"   
## [57] "UBP4\_Air\_2018\_10\_29-2019\_05\_09\_10538947\_QAQC.csv"   
## [58] "UBP4\_Air\_2019\_05\_09-2019\_10\_30\_10538947\_QAQC.csv"

bysite1<-lapply(1:length(sitekey),function(x) grep(sitekey[x],yktdat))  
length(bysite1[[1]])

## [1] 8

yktlist<-lapply(paste0(dir1,yktdat),read.csv, skip=1, header=T,col.names=c("Date","Time","Temp"))  
len.list<-lapply(1:length(yktlist),function(x) nrow(yktlist[[x]]))  
  
df5<-do.call(rbind,yktlist)  
  
lenkey<-lapply(1:length(bysite1), function(x) rep(sitekey[x],length(bysite1[[x]])))  
lenkey2<-unlist(lenkey)  
  
df5$site<-unlist(lapply(1:length(lenkey2),function(x){   
 rep(lenkey2[x],times=as.numeric(len.list[[x]]))  
})) #add sitename column

### Format the Yakutat Air Temp Data

df5$Date<-as.Date(df5$Date,format="%m/%d/%Y") #format date  
  
df5$Region<-"YF"  
  
summary(df5)

## Date Time Temp site   
## Min. :2012-09-25 Length:378233 Min. :-21.691 Length:378233   
## 1st Qu.:2013-09-09 Class :character 1st Qu.: 0.121 Class :character   
## Median :2014-09-28 Mode :character Median : 4.311 Mode :character   
## Mean :2015-07-19 Mean : 4.720   
## 3rd Qu.:2017-03-22 3rd Qu.: 9.965   
## Max. :2019-10-30 Max. : 30.558   
## NA's :4 NA's :4   
## Region   
## Length:378233   
## Class :character   
## Mode :character   
##   
##   
##   
##

str(df5)

## 'data.frame': 378233 obs. of 5 variables:  
## $ Date : Date, format: "2012-09-26" "2012-09-26" ...  
## $ Time : chr "12:00" "13:00" "14:00" "15:00" ...  
## $ Temp : num 6.37 6.27 6.37 6.17 6.17 ...  
## $ site : chr "MP1" "MP1" "MP1" "MP1" ...  
## $ Region: chr "YF" "YF" "YF" "YF" ...

dim(df5)

## [1] 378233 5

library(dplyr)  
unloadNamespace("reshape2")  
unloadNamespace("plyr")  
  
# Check the unique combination of Date and site  
unique\_combinations <- df5 %>%  
 select(Date, site) %>%  
 distinct()  
  
# Count the number of unique combinations  
nrow(unique\_combinations)

## [1] 12021

df5a <- df5 %>%  
 group\_by(Date, site) %>%  
 summarise(  
 Tmean = round(mean(Temp, na.rm = TRUE), 2),  
 Tmax = ifelse(all(is.na(Temp)), NA\_real\_, round(max(Temp, na.rm = TRUE), 2)),  
 Tmin = ifelse(all(is.na(Temp)), NA\_real\_, round(min(Temp, na.rm = TRUE), 2)),  
 SD = round(sd(Temp, na.rm = TRUE), 2)  
 ) %>%  
 ungroup()

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

YF.period<-seq(min(df5$Date, na.rm=T),max(df5$Date, na.rm=T), by="day") #maximum study period across all sites  
YF.df<-data.frame("site"=rep(unique(df5a$site),each=length(YF.period)),"Date"=rep(YF.period, times= length(unique(df5a$site))))  
df5b<-merge(YF.df,df5a, by=c("Date","site"), all.x=T)  
library(reshape2)

##   
## Attaching package: 'reshape2'  
##   
## The following object is masked from 'package:tidyr':  
##   
## smiths

### Create a plot of the mean temp through time

ggplot(df5b, aes(x=Date, y= Tmean))+  
 geom\_line()+  
 ylab(expression(paste("Daily mean air temperature ( ", degree ~C, " )")))+  
 scale\_x\_date(date\_breaks="1 year", date\_labels = "%Y")+  
 facet\_wrap(~site, ncol=3)+  
 ggtitle("Yakutat site air temperatures")+  
 theme\_bw()+  
 theme(axis.text.x=element\_text(angle=50, vjust=1, hjust=1))

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_line()`).

#export plot  
ggsave("YK\_air\_temps.png", path=plot.folder, width=8.5, height=6, units="in",dpi=300)

## Warning: Removed 1 row containing missing values or values outside the scale range  
## (`geom\_line()`).

### Find 7 day non-rolling averages to compare to Yakutat Airport

df5b <- df5b %>%  
 mutate(monyr = format(Date, "%m-%Y"))  
  
library(dplyr)  
unloadNamespace("reshape2")  
unloadNamespace("plyr")  
  
df5c <- df5b %>%  
 group\_by(site, monyr) %>%  
 summarise(monavg = round(mean(Tmean, na.rm = TRUE), 2),  
 monmax = if(anyNA(Tmax)) NA else round(max(Tmax, na.rm = TRUE), 2),  
 monmin = if(anyNA(Tmin)) NA else round(min(Tmin, na.rm = TRUE), 2),  
 .groups = 'drop')  
  
met3 <- met2b %>%  
 group\_by(NAME, monyr) %>%  
 summarise(  
 Amonavg = round(mean(TAVG2, na.rm = TRUE), 2),  
 Amonmax = round(max(TMAX, na.rm = TRUE), 2),  
 Amonmin = round(min(TMIN, na.rm = TRUE), 2)  
 ) %>%  
 ungroup()

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

#merge Yakutat dat with weather dat  
df5d<-merge(df5c,subset(met3, NAME=="YAKUTAT AIRPORT"), by="monyr",all.x=T)  
df5d$Region<-"YF" #restore region column in prep for merge with Copper River data.  
  
#Air vs. airport  
ggplot(na.omit(df5d), aes(x=Amonavg, y=monavg))+  
 geom\_hline(yintercept = 0, linetype=2)+  
 geom\_vline(xintercept = 0,linetype=2)+  
 geom\_abline(slope=1,intercept = 0, color = "red", lty=3)+  
 geom\_point(color="darkgray")+  
 geom\_smooth(method="lm", color="black", se=F)+  
 ylab(expression(paste("Site air temperature ( ", degree ~C, " )")))+  
 xlab(expression(paste("Airport air temperature ( ", degree ~C, ")")))+  
 ggtitle("Yakutat air temperatures")+  
 facet\_wrap(~site,ncol=3)+  
 theme\_minimal()

## `geom\_smooth()` using formula = 'y ~ x'

#export plot  
ggsave("YK\_air\_lm\_reg.png", path=plot.folder, width=6, height=6, units="in",dpi=300)

## `geom\_smooth()` using formula = 'y ~ x'

### Find Regression fit and coefficients Fit linear models for air vs. air temps at Yakutat sites/airport

sites2<-c("MP1", "MP3","MP5","MP8","PL1","PL2","PL3","UBP1","UBP2", "UBP3","UBP4")  
  
lm.1.res<-lapply(sites2,function(x) try(lm(monavg~Amonavg,data=df5d, subset= site==x),silent=T)) #run LM   
lm.1.sum<-lapply(1:length(sites2),function(x) summary(lm.1.res[[x]]))  
lm.1.sum

## [[1]]  
##   
## Call:  
## lm(formula = monavg ~ Amonavg, data = df5d, subset = site ==   
## x)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.52899 -0.48949 0.00987 0.57007 1.84821   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.43458 0.14645 -2.967 0.00463 \*\*   
## Amonavg 0.98838 0.01849 53.444 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7678 on 49 degrees of freedom  
## (35 observations deleted due to missingness)  
## Multiple R-squared: 0.9831, Adjusted R-squared: 0.9828   
## F-statistic: 2856 on 1 and 49 DF, p-value: < 2.2e-16  
##   
##   
## [[2]]  
##   
## Call:  
## lm(formula = monavg ~ Amonavg, data = df5d, subset = site ==   
## x)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.3306 -0.5225 0.0182 0.3682 4.7842   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.08965 0.24001 0.374 0.711   
## Amonavg 0.96080 0.02844 33.784 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.086 on 41 degrees of freedom  
## (43 observations deleted due to missingness)  
## Multiple R-squared: 0.9653, Adjusted R-squared: 0.9645   
## F-statistic: 1141 on 1 and 41 DF, p-value: < 2.2e-16  
##   
##   
## [[3]]  
##   
## Call:  
## lm(formula = monavg ~ Amonavg, data = df5d, subset = site ==   
## x)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -5.8992 -0.9385 -0.4173 0.4227 11.1092   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.75747 0.47677 -1.589 0.119   
## Amonavg 1.11962 0.06001 18.658 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.476 on 46 degrees of freedom  
## (38 observations deleted due to missingness)  
## Multiple R-squared: 0.8833, Adjusted R-squared: 0.8807   
## F-statistic: 348.1 on 1 and 46 DF, p-value: < 2.2e-16  
##   
##   
## [[4]]  
##   
## Call:  
## lm(formula = monavg ~ Amonavg, data = df5d, subset = site ==   
## x)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.1769 -0.3897 0.1962 0.6051 1.4697   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.45924 0.17750 -2.587 0.0131 \*   
## Amonavg 0.94493 0.02448 38.594 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8781 on 44 degrees of freedom  
## (40 observations deleted due to missingness)  
## Multiple R-squared: 0.9713, Adjusted R-squared: 0.9707   
## F-statistic: 1489 on 1 and 44 DF, p-value: < 2.2e-16  
##   
##   
## [[5]]  
##   
## Call:  
## lm(formula = monavg ~ Amonavg, data = df5d, subset = site ==   
## x)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.6215 -0.3848 -0.1257 0.3344 1.8904   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.96798 0.14353 -6.744 1.07e-08 \*\*\*  
## Amonavg 1.08883 0.01903 57.228 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7631 on 54 degrees of freedom  
## (30 observations deleted due to missingness)  
## Multiple R-squared: 0.9838, Adjusted R-squared: 0.9835   
## F-statistic: 3275 on 1 and 54 DF, p-value: < 2.2e-16  
##   
##   
## [[6]]  
##   
## Call:  
## lm(formula = monavg ~ Amonavg, data = df5d, subset = site ==   
## x)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.6209 -0.4181 -0.2083 0.3462 2.0787   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.39011 0.15936 -2.448 0.0185 \*   
## Amonavg 1.06679 0.02085 51.171 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7745 on 43 degrees of freedom  
## (41 observations deleted due to missingness)  
## Multiple R-squared: 0.9838, Adjusted R-squared: 0.9835   
## F-statistic: 2618 on 1 and 43 DF, p-value: < 2.2e-16  
##   
##   
## [[7]]  
##   
## Call:  
## lm(formula = monavg ~ Amonavg, data = df5d, subset = site ==   
## x)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.00229 -0.29274 -0.05001 0.37567 1.62286   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.57404 0.13387 -4.288 7.49e-05 \*\*\*  
## Amonavg 1.08684 0.01774 61.248 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7118 on 54 degrees of freedom  
## (30 observations deleted due to missingness)  
## Multiple R-squared: 0.9858, Adjusted R-squared: 0.9855   
## F-statistic: 3751 on 1 and 54 DF, p-value: < 2.2e-16  
##   
##   
## [[8]]  
##   
## Call:  
## lm(formula = monavg ~ Amonavg, data = df5d, subset = site ==   
## x)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.85773 -0.04451 0.07360 0.17156 0.43013   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -1.11017 0.14483 -7.665 9.78e-06 \*\*\*  
## Amonavg 1.04172 0.01695 61.455 2.62e-15 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3312 on 11 degrees of freedom  
## (73 observations deleted due to missingness)  
## Multiple R-squared: 0.9971, Adjusted R-squared: 0.9968   
## F-statistic: 3777 on 1 and 11 DF, p-value: 2.621e-15  
##   
##   
## [[9]]  
##   
## Call:  
## lm(formula = monavg ~ Amonavg, data = df5d, subset = site ==   
## x)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -5.3853 -0.6600 0.2757 0.8713 1.5200   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.12292 0.30118 -0.408 0.686   
## Amonavg 0.96353 0.04805 20.053 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.34 on 27 degrees of freedom  
## (57 observations deleted due to missingness)  
## Multiple R-squared: 0.9371, Adjusted R-squared: 0.9348   
## F-statistic: 402.1 on 1 and 27 DF, p-value: < 2.2e-16  
##   
##   
## [[10]]  
##   
## Call:  
## lm(formula = monavg ~ Amonavg, data = df5d, subset = site ==   
## x)  
##   
## Residuals:  
## 5 84 165 714 794 876   
## 0.8164 1.8787 -0.1295 -2.3372 0.8834 -1.1119   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -0.7131 0.7082 -1.007 0.371  
## Amonavg 0.2044 0.3263 0.626 0.565  
##   
## Residual standard error: 1.71 on 4 degrees of freedom  
## (80 observations deleted due to missingness)  
## Multiple R-squared: 0.08932, Adjusted R-squared: -0.1383   
## F-statistic: 0.3923 on 1 and 4 DF, p-value: 0.565  
##   
##   
## [[11]]  
##   
## Call:  
## lm(formula = monavg ~ Amonavg, data = df5d, subset = site ==   
## x)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -4.5543 -0.2507 0.1904 0.7160 1.3175   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.20835 0.28554 0.73 0.471   
## Amonavg 0.92329 0.03841 24.04 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.284 on 32 degrees of freedom  
## (52 observations deleted due to missingness)  
## Multiple R-squared: 0.9475, Adjusted R-squared: 0.9459   
## F-statistic: 577.9 on 1 and 32 DF, p-value: < 2.2e-16

lm.1.pred<-lapply(sites2,function(x) try(predict(lm(monavg~Amonavg,data=df5d, subset= site==x)),silent=T))   
lm.1.obs<-lapply(sites2, function(x) na.omit(subset(df5d, subset= site==x, select = monavg )))  
  
  
#Create summary table for comparing models  
sumtab<-as.data.frame(matrix(NA,nrow=length(sites2),ncol=6))  
colnames(sumtab)<-c("site","lm1.slope","lm1.intercept","lm1.adj.r.sq","lm1.res.stan.error","lm1.rmse")  
  
sumtab[,1]<-sites2  
  
# get air freeze results for sumtab  
sumtab[,2]<-as.vector(unlist(lapply(1:length(sites2),function(x){  
 round(coef(lm.1.res[[x]])[2],1)  
}))) #slope  
sumtab[,3]<-as.vector(unlist(lapply(1:length(sites2),function(x){  
 round(coef(lm.1.res[[x]])[1],2)  
}))) #intercept  
  
sumtab[,4]<-as.vector(unlist(lapply(1:length(sites2),function(x){  
 round(summary(lm.1.res[[x]])$adj.r.squared,2)  
}))) #adj.r.sq  
  
sumtab[,5]<-as.vector(unlist(lapply(1:length(sites2),function(x){  
 round(summary(lm.1.res[[x]])$sigma,2)  
}))) #residual standard error  
  
sumtab[,6]<-round(unlist(lapply(1:length(sites2), function(x)   
 rmse(unlist(lm.1.pred[[x]]),unlist(lm.1.obs[[x]])))),3)  
  
print(sumtab)

## site lm1.slope lm1.intercept lm1.adj.r.sq lm1.res.stan.error lm1.rmse  
## 1 MP1 1.0 -0.43 0.98 0.77 0.753  
## 2 MP3 1.0 0.09 0.96 1.09 1.060  
## 3 MP5 1.1 -0.76 0.88 2.48 2.424  
## 4 MP8 0.9 -0.46 0.97 0.88 0.859  
## 5 PL1 1.1 -0.97 0.98 0.76 0.749  
## 6 PL2 1.1 -0.39 0.98 0.77 0.757  
## 7 PL3 1.1 -0.57 0.99 0.71 0.699  
## 8 UBP1 1.0 -1.11 1.00 0.33 0.305  
## 9 UBP2 1.0 -0.12 0.93 1.34 1.293  
## 10 UBP3 0.2 -0.71 -0.14 1.71 1.396  
## 11 UBP4 0.9 0.21 0.95 1.28 1.246

#Get needed info for air at pond vs. airport air temps now, as this step removes those files (repeat use of same data names below)  
rm(lm.1.obs,lm.1.pred, lm.1.res, lm.1.sum,sumtab)

### Here the files with the air temperature data are: df5d

head(df5d)

## monyr site monavg monmax monmin NAME Amonavg Amonmax Amonmin  
## 1 01-2013 MP1 -0.70 3.15 -12.66 YAKUTAT AIRPORT -0.31 5.6 -14.4  
## 2 01-2013 PL2 -0.98 6.67 -15.45 YAKUTAT AIRPORT -0.31 5.6 -14.4  
## 3 01-2013 MP3 -0.45 4.21 -11.92 YAKUTAT AIRPORT -0.31 5.6 -14.4  
## 4 01-2013 UBP2 0.05 5.45 -13.71 YAKUTAT AIRPORT -0.31 5.6 -14.4  
## 5 01-2013 UBP3 0.04 5.55 -13.87 YAKUTAT AIRPORT -0.31 5.6 -14.4  
## 6 01-2013 PL1 -1.35 6.47 -15.77 YAKUTAT AIRPORT -0.31 5.6 -14.4  
## Region  
## 1 YF  
## 2 YF  
## 3 YF  
## 4 YF  
## 5 YF  
## 6 YF

# Part 3 - Pulling in the Water Temperature Data

## Section 1 - Yakutat Forelands

### Check the working directory again

getwd()

## [1] "/Users/amaryllisadey/Dropbox/SWEL/CopperRiverDelta/PondTemps/DataAnalysis/PondTempsAnalysis\_GIT"

### Load in the water temperature data from the Yakutat Foreland

dir1<-paste0(this.dir,"/DataFiles/PondData/Yakutat/Water/")  
yktdat<-list.files(dir1)  
sitekey<-c("MP1","MP3","MP5","MP8","PL1","PL2","PL3","UBP1","UBP2", "UBP3","UBP4") #creates a vector of all of the site names  
yktdat

## [1] "MP1\_Surface\_2012\_08\_10-2013\_06\_27\_9922086\_QAQC.csv"   
## [2] "MP1\_Surface\_2013\_06\_27-2013\_10\_28\_9922086\_QAQC.csv"   
## [3] "MP1\_Surface\_2015\_09\_30-2016\_04\_14\_10144196\_QAQC.csv"   
## [4] "MP1\_Surface\_2016\_10\_13-2017\_05\_23\_10944566\_QAQC.csv"   
## [5] "MP1\_Surface\_2018\_05\_25-2018\_10\_11\_11001117\_QAQC.csv"   
## [6] "MP1\_Surface\_2018\_10\_11-2019\_06\_06\_11001117\_QAQC.csv"   
## [7] "MP1\_Surface\_2019\_06\_06-2019\_10\_16\_11001117\_QAQC.csv"   
## [8] "MP3 Surface Temp 2012\_08\_10 - 2013\_06\_27 10129015.csv"   
## [9] "MP3 Surface Temp 2013\_06\_27 - 2013\_10\_28 10129015.csv"   
## [10] "MP3 Surface Temp 2013\_10\_28 - 2014\_06\_02 10129015.csv"   
## [11] "MP3 Surface Temp 2014\_06\_02 - 2014\_09\_11 10129015.csv"   
## [12] "MP3\_Surface\_2018\_05\_25-2018\_10\_11\_10537664\_QAQC.csv"   
## [13] "MP3\_Surface\_2018\_10\_11-2019\_06\_06\_10537664\_QAQC.csv"   
## [14] "MP3\_Surface\_2019\_06\_06-2019\_10\_16\_10537664\_QAQC.csv"   
## [15] "MP5 Surface Temp 2012\_09\_25 - 2013\_06\_27 10144194.csv"   
## [16] "MP5 Surface Temp 2013\_06\_27 - 2013\_10\_28 10144194.csv"   
## [17] "MP5 Surface Temp 2013\_10\_28 - 2014\_06\_02 10144194.csv"   
## [18] "MP5 Surface Temp 2014\_06\_02 - 2014\_09\_11 10144194.csv"   
## [19] "MP5\_Surface\_2016\_10\_04-2017\_05\_24\_10377263\_QAQC.csv"   
## [20] "MP5\_Surface\_2019\_06\_06-2019\_10\_16\_10333059\_QAQC.csv"   
## [21] "MP8 Surface Temp 2012\_09\_25 - 2013\_06\_27 10144203.csv"   
## [22] "MP8 Surface Temp 2013\_06\_27 - 2013\_10\_28 10144203.csv"   
## [23] "MP8 Surface Temp 2013\_10\_28 - 2014\_06\_02 10144203.csv"   
## [24] "MP8 Surface Temp 2014\_06\_02 - 2014\_09\_11 10144203.csv"   
## [25] "MP8 Surface Temp 2015\_06\_04 - 2015\_10\_16 10144203\_weighted\_QAQC.csv"  
## [26] "MP8 Surface Temp 2015\_10\_20 - 2016\_04\_14 10144203.csv"   
## [27] "MP8\_Surface\_2018\_05\_25-2018\_10\_11\_10537665\_QAQC\_Question.csv"   
## [28] "MP8\_Surface\_2018\_10\_11-2019\_06\_06\_10537665\_QAQC.csv"   
## [29] "MP8\_Surface\_2019\_06\_06-2019\_10\_16\_10537665\_QAQC.csv"   
## [30] "PL1 Surface Temp 2013\_10\_24 - 2014\_06\_10 9775266.csv"   
## [31] "PL1\_Surface\_2015\_06\_11-2015\_06\_30\_10402264\_QAQC.csv"   
## [32] "PL1\_Surface\_2015\_06\_30-2016\_05\_20\_1021045\_QAQC.csv"   
## [33] "PL1\_Surface\_2018\_10\_17-2019\_06\_07\_11001118\_QAQC.csv"   
## [34] "PL1\_Surface\_2019\_06\_07-2019\_8\_27\_11001118\_QAQC.csv"   
## [35] "PL2 Surface Temp 2012\_08\_09 - 2013\_10\_24 9775266.csv"   
## [36] "PL2 Surface Temp 2013\_10\_24 - 2014\_06\_10 10402255.csv"   
## [37] "PL2\_Surface\_2018\_06\_21-2019\_06\_07\_11001123\_QAQC.csv"   
## [38] "PL2\_Surface\_2019\_06\_07-2019\_10\_17\_11001123\_QAQC.csv"   
## [39] "PL3 Surface Temp 2013\_10\_24 - 2014\_06\_10 10402265.csv"   
## [40] "PL3 Surface Temp 2014\_06\_10 - 2015\_06\_11 10402265.csv"   
## [41] "PL3\_Surface\_2012\_08\_09-2013\_10\_24\_1021045\_QAQC.csv"   
## [42] "PL3\_Surface\_2018\_10\_17-2019\_06\_07\_11001119.csv"   
## [43] "PL3\_Surface\_2019\_06\_07-2019\_10\_17\_11001119\_QAQC.csv"   
## [44] "UBP1 Surface Temp 2012\_08\_08 - 2013\_06\_26 10144208.csv"   
## [45] "UBP1 Surface Temp 2013\_06\_26 - 2013\_10\_22 10144208.csv"   
## [46] "UBP1 Surface Temp 2014\_09\_16 - 2015\_06\_04 10144208.csv"   
## [47] "UBP1\_Surface\_2018\_10\_16-2019\_05\_09\_10537669\_QAQC.csv"   
## [48] "UBP1\_Surface\_2019\_05\_09-2019\_10\_28\_10537669\_QAQC.csv"   
## [49] "UBP2\_Surface\_2012\_08\_08-2013\_06\_26\_10144199\_QAQC.csv"   
## [50] "UBP2\_Surface\_2013\_06\_26-2013\_10\_22\_10144199\_QAQC.csv"   
## [51] "UBP2\_Surface\_2013\_10\_22-2014\_06\_03\_10144199\_QAQC.csv"   
## [52] "UBP2\_Surface\_2014\_09\_16-2015\_06\_04\_10144199\_QAQC.csv"   
## [53] "UBP2\_Surface\_2015\_09\_30-2016\_05\_03\_10144199\_QAQC.csv"   
## [54] "UBP2\_Surface\_2016\_09\_23-2017\_05\_24\_11001117\_QAQC.csv"   
## [55] "UBP2\_Surface\_2018\_10\_16-2019\_05\_09\_10944564\_QAQC.csv"   
## [56] "UBP2\_Surface\_2019\_05\_09-2019\_10\_28\_10944564\_QAQC.csv"   
## [57] "UBP3 Surface Temp 2014\_09\_16 - 2015\_06\_04 10402258.csv"   
## [58] "UBP3\_Surface\_2012\_08\_08\_2013\_06\_26\_10144207\_QAQC.csv"   
## [59] "UBP3\_Surface\_2018\_10\_16-2019\_05\_09\_11001121\_QAQC.csv"   
## [60] "UBP3\_Surface\_2019\_05\_09-2019\_10\_28\_11001121\_QAQC.csv"   
## [61] "UBP4\_Surface\_2015\_09\_29-2015\_10\_18\_10402262\_QAQC.csv"   
## [62] "UBP4\_Surface\_2018\_10\_29-2019\_05\_09\_10944565\_QAQC.csv"   
## [63] "UBP4\_Surface\_2019\_05\_09-2019\_10\_30\_10944565\_QAQC.csv"

bysite1<-lapply(1:length(sitekey),function(x) grep(sitekey[x],yktdat))   
#Still working on lapply, but grep(pattern,x) searches for a pattern (sitekey) in a vector of character strings.  
#So for this, we are searching for the sitekey in yktdat  
length(bysite1[[1]])

## [1] 7

yktlist<-lapply(paste0(dir1,yktdat),read.csv, skip=1, header=T,col.names=c("Date","Time","Temp"))  
len.list<-lapply(1:length(yktlist),function(x) nrow(yktlist[[x]]))  
  
df1<-do.call(rbind,yktlist)  
  
yktlenkey<-lapply(1:length(bysite1), function(x) rep(sitekey[x],length(bysite1[[x]])))  
yktlenkey2<-unlist(yktlenkey)

### Format the dataframe

Include a date, site name column, YF for region

df1$site<-unlist(lapply(1:length(yktlenkey2),function(x){   
 rep(yktlenkey2[x],times=as.numeric(len.list[[x]]))  
})) #add sitename column  
  
  
df1$Date<-as.Date(df1$Date,format="%m/%d/%Y") #format date  
  
df1$Region<-"YF"  
  
summary(df1)

## Date Time Temp site   
## Min. :2012-08-08 Length:248122 Min. :-16.264 Length:248122   
## 1st Qu.:2013-12-16 Class :character 1st Qu.: 0.852 Class :character   
## Median :2016-03-12 Mode :character Median : 5.154 Mode :character   
## Mean :2016-06-21 Mean : 7.043   
## 3rd Qu.:2019-02-13 3rd Qu.: 13.112   
## Max. :2019-10-30 Max. : 26.879   
## Region   
## Length:248122   
## Class :character   
## Mode :character   
##   
##   
##

library(dplyr)  
unloadNamespace("reshape2")  
unloadNamespace("plyr")  
  
df1a <- df1 %>%  
 group\_by(Date, site) %>%  
 summarise(  
 Tmean = round(mean(Temp, na.rm = TRUE), 2),  
 Tmax = round(max(Temp, na.rm = TRUE), 2),  
 Tmin = round(min(Temp, na.rm = TRUE), 2),  
 SD = round(sd(Temp, na.rm = TRUE), 2)  
 )

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

library(reshape2)

##   
## Attaching package: 'reshape2'  
##   
## The following object is masked from 'package:tidyr':  
##   
## smiths

YF.period<-seq(min(df1$Date, na.rm=T),max(df1$Date, na.rm=T), by="day") #maximum study period across all sites  
YF.df<-data.frame("site"=rep(unique(df1a$site),each=length(YF.period)),"Date"=rep(YF.period, times= length(unique(df1a$site))))  
df1b<-merge(YF.df,df1a, by=c("Date","site"), all.x=T)  
  
df1b$Tmean[df1b$Tmean<0]<-0 #change mean values less than 0 to 0. Tmin will still reflect days when negative values were recorded, indicating air exposure in winter.

### Create a plot of the Yakutat Foreland Pond Temperatures through time

ggplot(df1b, aes(x=Date, y= Tmean))+  
 geom\_line()+  
 ylab(expression(paste("Daily mean water temperature ( ", degree ~C, " )")))+  
 scale\_x\_date(date\_breaks="1 year", date\_labels = "%Y")+  
 facet\_wrap(~site, ncol=3)+  
 theme\_minimal()+  
 ggtitle("Yakutat Foreland")+  
 theme(axis.text.x=element\_text(angle=50, vjust=1, hjust=1))

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_line()`).

#export plot  
ggsave("YF\_water\_temps.png", path=plot.folder, width=6, height=6, units="in",dpi=300)

## Warning: Removed 2 rows containing missing values or values outside the scale range  
## (`geom\_line()`).

## Section 2 - Copper River Delta

### Get the working directory info

getwd()

## [1] "/Users/amaryllisadey/Dropbox/SWEL/CopperRiverDelta/PondTemps/DataAnalysis/PondTempsAnalysis\_GIT"

### Pull in the data files

dir2<-paste0(this.dir,"/DataFiles/PondData/Cordova/Water/")  
  
crdat<-list.files(dir2)  
  
sitekeycr<-c("BeaverS", "Cabin","CanneryControl", "EyakS","RHM","Square","TDN", "TDS", "Wooded") #searching for these names  
crdat

## [1] "BeaverS\_CV\_10169389-8-13-13.csv"   
## [2] "BeaverS\_CV\_10201433-7-23-14.csv"   
## [3] "Cabin Lk Outlet Stream Surface Temp 2011\_09\_17 - 2012\_06\_29 10005377\_QAQC.csv"   
## [4] "Cabin Lk Outlet Stream Surface Temp 2012\_06\_29 - 2012\_10\_19 10005377\_QAQC.csv"   
## [5] "Cabin Lk Outlet Stream Surface Temp 2012\_10\_19 - 2013\_05\_31 10005377\_QAQC.csv"   
## [6] "Cabin Lk Outlet Stream Surface Temp 2013\_05\_31 - 2014\_06\_12 10005377\_QAQC\_akst.csv"  
## [7] "Cabin Lk Outlet Stream Surface Temp 2014\_06\_12 - 2015\_04\_30 10005377\_QAQC\_akst.csv"  
## [8] "Cabin Lk Outlet Stream Surface Temp 2015\_04\_30 - 2016\_02\_09 10506963\_QAQC\_akst.csv"  
## [9] "Cabin Lk Outlet Stream Surface Temp 2016\_02\_09 - 2016\_06\_03 10506963\_QAQC\_akst.csv"  
## [10] "Cabin Lk Outlet Stream Surface Temp 2016\_11\_27 - 2017\_11\_03 10506963\_QAQC.csv"   
## [11] "Cabin Lk Outlet Stream Surface Temp 2017\_11\_03 - 2019\_05\_31 10506963\_QAQC.csv"   
## [12] "CanneryControl surface temp top 2017\_05\_05 - 2017\_10\_20 10166531.csv"   
## [13] "CanneryControl\_06\_06\_2015-09\_18\_2015\_10163229.csv"   
## [14] "CanneryControl\_templight\_mid\_40cm\_20180608-20181026\_10129008.csv"   
## [15] "CanneryControl\_templight\_up\_50cm\_2019\_04\_23-2019\_10\_17\_20207524.csv"   
## [16] "CanneryControl\_templight\_WINTER\_30cm\_20181026-20190320\_10316646.csv"   
## [17] "CanneryControl\_uppercolumn\_20160609-20170505\_9995395.csv"   
## [18] "EyakS\_CV\_10166544-6-10-14.csv"   
## [19] "EyakS\_CV\_9995395-6-18-13.csv"   
## [20] "EyakSouth surface temp midcoluimn 2017\_06\_02 - 2017\_10\_20 10316650.csv"   
## [21] "EyakSouth\_templight\_mid\_30cm\_2018-06-08-2019-04-23\_10166532.csv"   
## [22] "EyakSouth\_templight\_mid\_30cm\_2019\_04\_23-2019\_10\_17\_10129009.csv"   
## [23] "RHM CV Surface Temp 2012\_07\_30 - 2013\_06\_20 10166539.csv"   
## [24] "RHM CV Surface Temp 2013\_06\_20 - 2013\_08\_05 10201447.csv"   
## [25] "RHM CV Surface Temp 2013\_08\_05 - 2014\_06\_23 10201453.csv"   
## [26] "RHM CV Surface Temp 2014\_06\_13 - 2014\_08\_05 10166541.csv"   
## [27] "RHM CV Surface Temp 2014\_08\_05 - 2016\_05\_02 10201433.csv"   
## [28] "RHM Surface Temp mid-column 2015\_05\_02 - 2016\_08\_26 10497803.csv"   
## [29] "RHM\_Surface\_2018\_07\_10-2019\_08\_27\_10506969\_QAQC.csv"   
## [30] "RHM\_surface\_temp\_sink\_2016\_08\_26 - 2018\_07\_10\_10506969\_QAQC.csv"   
## [31] "Square\_CV\_2012\_9775267-7-30-12.csv"   
## [32] "Square\_CV\_2013\_10166442-6-20-13.csv"   
## [33] "Square\_CV\_2014\_0201445-8-5-14.csv"   
## [34] "TDN Surface Temp 2011\_06\_21 - 2014\_06\_11 9796932\_QAQC.csv"   
## [35] "TDN\_CV\_10129009-6-17-13.csv"   
## [36] "TDN\_CV\_10166531-6-11-14.csv"   
## [37] "TDN\_CV\_9775266-7-31-12.csv"   
## [38] "TDS CV Surface Temp 2011\_09\_18 - 2012\_07\_31 9775268.csv"   
## [39] "TDS CV Surface Temp 2012\_07\_31 - 2013\_06\_17 10166541.csv"   
## [40] "TDS CV Surface Temp 2013\_06\_17 - 2013\_08\_07 10201449.csv"   
## [41] "TDS CV Surface Temp 2013\_08\_07 - 2014\_05\_07 10166442.csv"   
## [42] "TDS CV Surface Temp 2014\_06\_11-2014\_08\_07 10201437.csv"   
## [43] "TDS Surface Temp mid-column 2015\_05\_01 - 2016\_08\_26 10506970\_QAQC.csv"   
## [44] "TDS Surface Temp mid-column 2017\_06\_07 - 2018\_06\_25 10497804\_QAQC.csv"   
## [45] "TDS Surface Temp midcolumn 2016\_08\_26 - 2017\_06\_07 10497804\_QAQC.csv"   
## [46] "Wooded\_06\_10\_2015-09\_14\_2015\_10179156.csv"   
## [47] "Wooded\_20160607-20161029 10316631.csv"   
## [48] "Wooded\_20161029\_20170504\_10949702.csv"   
## [49] "Wooded\_20181102\_20191003\_10949702.csv"   
## [50] "Wooded\_templight\_24cm\_20170504\_20171018\_10201434.csv"   
## [51] "Wooded\_templight\_low\_47cm\_20180607-20181031\_10575150.csv"   
## [52] "Wooded\_templight\_mid\_40cm\_2018\_10\_31-2019\_04\_18\_10575150.csv"   
## [53] "Wooded\_templight\_mid\_40cm\_2019\_04\_18-2019\_10\_03\_10201433.csv"

bysite2<-lapply(1:length(sitekeycr),function(x) grep(sitekeycr[x],crdat)) #if this organizes the data by site   
length(bysite2) #used to be length(bysite[[1]]), but RH changed because this gives the length of bysite1, not the length of the first bysite1 (bysite[[1]])

## [1] 9

crlist<-lapply(paste0(dir2,crdat),read.csv, skip=1, header=T,col.names=c("Date","Time","Temp"),row.names=NULL)   
crlen.list<-lapply(1:length(crlist),function(x) nrow(crlist[[x]]))  
  
df2<-do.call(rbind,crlist)  
  
crlenkey<-lapply(1:length(bysite2), function(x) rep(sitekeycr[x],length(bysite2[[x]])))  
crlenkey2<-unlist(crlenkey)  
length(crlenkey2)

## [1] 53

### Format the dataframe

Include a date, site name column, YF for region

df2$site<-unlist(lapply(1:length(crlenkey2),function(x){   
 rep(crlenkey2[x],times=as.numeric(crlen.list[[x]]))  
})) #add sitename column  
  
df2$Date<-as.Date(df2$Date,format="%m/%d/%Y") #format date  
  
df2$Region<-"CR"  
  
library(plyr)

## ------------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)

## ------------------------------------------------------------------------------

##   
## Attaching package: 'plyr'

## The following object is masked from 'package:ggpubr':  
##   
## mutate

## The following objects are masked from 'package:dplyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize

## The following object is masked from 'package:purrr':  
##   
## compact

#Change site names to Carmella's codes for consistency with three character Yakutat codes  
df2$site<-mapvalues(df2$site, from = c("BeaverS", "Cabin","CanneryControl", "EyakS","RHM","Square","TDN", "TDS", "Wooded"),  
 to = c("BVS","CAB","CAN", "EYS","RHM","SQR","TIN","TIS","WDD"))  
unique(df2$site)

## [1] "BVS" "CAB" "CAN" "EYS" "RHM" "SQR" "TIN" "TIS" "WDD"

summary(df2)

## Date Time Temp site   
## Min. :2011-06-21 Length:450291 Min. :-11.487 Length:450291   
## 1st Qu.:2012-12-26 Class :character 1st Qu.: 1.112 Class :character   
## Median :2014-04-01 Mode :character Median : 4.623 Mode :character   
## Mean :2015-02-06 Mean : 6.820   
## 3rd Qu.:2017-04-19 3rd Qu.: 12.147   
## Max. :2019-10-17 Max. : 29.752   
## Region   
## Length:450291   
## Class :character   
## Mode :character   
##   
##   
##

unloadNamespace("reshape2")  
unloadNamespace("plyr")  
  
df2$Date<-as.Date(df2$Date,format="%m/%d/%Y") #format date  
  
df2$Region<-"CRD"  
  
df2a <- df2 %>%  
 group\_by(Date, site) %>%  
 summarise(  
 Tmean = round(mean(Temp, na.rm = TRUE), 2),  
 Tmax = round(max(Temp, na.rm = TRUE), 2),  
 Tmin = round(min(Temp, na.rm = TRUE), 2),  
 SD = round(sd(Temp, na.rm = TRUE), 2),  
 Region = unique(Region)  
 )

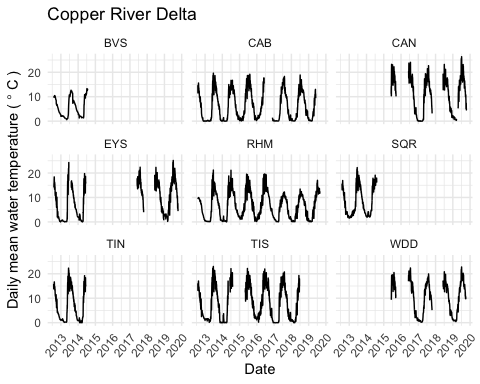
## `summarise()` has grouped output by 'Date'. You can override using the  
## `.groups` argument.

CR.period<-seq(min(df2$Date, na.rm=T),max(df2$Date, na.rm=T), by="day") #maximum study period across all sites  
CR.df<-data.frame("site"=rep(unique(df2a$site),each=length(CR.period)),"Date"=rep(CR.period, times= length(unique(df2a$site))))  
df2b<-merge(CR.df,df2a, by=c("Date","site"), all.x=T)  
  
df2b$Tmean[df2b$Tmean<0]<-0 #change mean values less than 0 to 0. Tmin will still reflect days when negative values were recorded, indicating air exposure in winter.  
df2b<-subset(df2b, Date >= as.Date("2012-08-01"))

### Create a plot of the Copper River Pond Temperatures through Time

ggplot(df2b, aes(x=Date, y= Tmean))+  
 geom\_line()+  
 ylab(expression(paste("Daily mean water temperature ( ", degree ~C, " )")))+  
 scale\_x\_date(date\_breaks="1 year", date\_labels = "%Y")+  
 facet\_wrap(~site, ncol=3)+  
 ggtitle("Copper River Delta")+  
 theme\_minimal()+  
 theme(axis.text.x=element\_text(angle=50, vjust=1, hjust=1))

## Warning: Removed 14 rows containing missing values or values outside the scale range  
## (`geom\_line()`).



#export plot  
ggsave("CR\_water\_temps.png", path=plot.folder, width=6, height=6, units="in",dpi=300)

## Warning: Removed 14 rows containing missing values or values outside the scale range  
## (`geom\_line()`).

# Part 4 - Finding Monthly Non-Rolling Averages for Water Temp

## Pulling in the data and adding a month-year column

#Yakutat  
df1a$monyr<-format(df1a$Date,"%m-%Y") #add month-year column  
  
df1c <- df1a %>%  
 group\_by(site, monyr) %>%  
 summarise(  
 monavg = round(mean(Tmean, na.rm = TRUE), 2),  
 monmax = round(max(Tmax, na.rm = TRUE), 2),  
 monmin = round(min(Tmin, na.rm = TRUE), 2)  
 ) %>%  
 ungroup()

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

#Cordova  
df2a$monyr<-format(df2a$Date,"%m-%Y") #add mon-year column  
  
df2c <- df2a %>%  
 group\_by(site, monyr) %>%  
 summarise(  
 monavg = round(mean(Tmean, na.rm = TRUE), 2),  
 monmax = round(max(Tmax, na.rm = TRUE), 2),  
 monmin = round(min(Tmin, na.rm = TRUE), 2)  
 ) %>%  
 ungroup()

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

## Merging the Weather Data with Each Region

#merge Yakutat dat with weather dat  
# Subset met3 where Airport is "YAKUTAT AIRPORT"  
YF\_subset\_met3 <- subset(met3, NAME == "YAKUTAT AIRPORT")  
  
# Perform the left join with subset\_met3 as the left data frame  
df1d <- merge(YF\_subset\_met3, df1c, by = "monyr", all.x = TRUE)  
df1d$Region<-"YF" #restore region column in prep for merge with Copper River data.  
  
# Subset met3 where Airport is "CORDOVA AIRPORT"  
CR\_subset\_met3 <- subset(met3, NAME == "CORDOVA AIRPORT")  
  
# Perform the left join with subset\_met3 as the left data frame  
df2d <- merge(CR\_subset\_met3, df2c, by = "monyr", all.x = TRUE)  
df2d$Region<-"CR" #restore region column in prep for merge with Copper River data.

## Merge the Yakutat Forelands and Copper River Delta Files

I think I can actually skip this and keep these files separate to forecast for CRD and YF separately

#Merge YF and CR dat  
#df3<-rbind(df1d, df2d)   
#df3$site<-factor(df3$site)  
#df3$NAME<-factor(df3$NAME)  
#df3$Region<-factor(df3$Region)  
#df3$monavg[df3$monavg<0]<-0

# Part 5 - Figuring out how to pivot this file to be wider instead of ‘longer’

## Section 1 - Yakutat Forelands

### Set up the file name and check the structure

YF\_Data <- df1d  
str(YF\_Data)

## 'data.frame': 479 obs. of 10 variables:  
## $ monyr : chr "01-2012" "01-2013" "01-2013" "01-2013" ...  
## $ NAME : Factor w/ 2 levels "CORDOVA AIRPORT",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ Amonavg: num -4.53 -0.31 -0.31 -0.31 -0.31 -0.31 -0.31 -0.31 -0.31 -0.31 ...  
## $ Amonmax: num 5.6 5.6 5.6 5.6 5.6 5.6 5.6 5.6 5.6 5.6 ...  
## $ Amonmin: num -23.9 -14.4 -14.4 -14.4 -14.4 -14.4 -14.4 -14.4 -14.4 -14.4 ...  
## $ site : chr NA "MP1" "MP3" "PL3" ...  
## $ monavg : num NA 0.12 0.65 -0.4 -0.4 -0.54 0.43 0.52 0.02 0.21 ...  
## $ monmax : num NA 0.23 0.78 -0.06 -0.06 0.67 0.67 0.67 0.12 0.23 ...  
## $ monmin : num NA -0.55 0.45 -3.12 -3.12 -8.45 0.23 0.45 -1.23 0.01 ...  
## $ Region : chr "YF" "YF" "YF" "YF" ...

## Need to make wider by site and change some column names

### Starting with changing column names to be clearer

YF\_AirPondData <- YF\_Data %>%  
 rename(  
 `Month-Year` = monyr,  
 Pond = site,  
 Water\_MonthAvg = monavg,  
 Water\_MonthMax = monmax,  
 Water\_MonthMin = monmin,  
 Airport = NAME,  
 Air\_MonthAvg = Amonavg,  
 Air\_MonthMax = Amonmax,  
 Air\_MonthMin = Amonmin  
 )  
  
str(YF\_AirPondData)

## 'data.frame': 479 obs. of 10 variables:  
## $ Month-Year : chr "01-2012" "01-2013" "01-2013" "01-2013" ...  
## $ Airport : Factor w/ 2 levels "CORDOVA AIRPORT",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ Air\_MonthAvg : num -4.53 -0.31 -0.31 -0.31 -0.31 -0.31 -0.31 -0.31 -0.31 -0.31 ...  
## $ Air\_MonthMax : num 5.6 5.6 5.6 5.6 5.6 5.6 5.6 5.6 5.6 5.6 ...  
## $ Air\_MonthMin : num -23.9 -14.4 -14.4 -14.4 -14.4 -14.4 -14.4 -14.4 -14.4 -14.4 ...  
## $ Pond : chr NA "MP1" "MP3" "PL3" ...  
## $ Water\_MonthAvg: num NA 0.12 0.65 -0.4 -0.4 -0.54 0.43 0.52 0.02 0.21 ...  
## $ Water\_MonthMax: num NA 0.23 0.78 -0.06 -0.06 0.67 0.67 0.67 0.12 0.23 ...  
## $ Water\_MonthMin: num NA -0.55 0.45 -3.12 -3.12 -8.45 0.23 0.45 -1.23 0.01 ...  
## $ Region : chr "YF" "YF" "YF" "YF" ...

### Pivot the dataframe wider to make a column for each ponds temepratures

YF\_PondAirTemp\_mean <- YF\_AirPondData %>%  
 pivot\_wider(  
 id\_cols = c("Month-Year", "Airport", "Air\_MonthAvg", "Air\_MonthMax", "Air\_MonthMin", "Region"),  
 names\_from = Pond,  
 values\_from = c(Water\_MonthAvg)  
 )  
  
YF\_PondAirTemp\_max <- YF\_AirPondData %>%  
 pivot\_wider(  
 id\_cols = c("Month-Year", "Airport", "Air\_MonthAvg", "Air\_MonthMax", "Air\_MonthMin", "Region"),  
 names\_from = Pond,  
 values\_from = c(Water\_MonthMax)  
 )  
  
YF\_PondAirTemp\_min <- YF\_AirPondData %>%  
 pivot\_wider(  
 id\_cols = c("Month-Year", "Airport", "Air\_MonthAvg", "Air\_MonthMax", "Air\_MonthMin", "Region"),  
 names\_from = Pond,  
 values\_from = c(Water\_MonthMin)  
 )

### Correct the Data formats and order these by date here

# 1. mean values  
str(YF\_PondAirTemp\_mean)

## tibble [96 × 18] (S3: tbl\_df/tbl/data.frame)  
## $ Month-Year : chr [1:96] "01-2012" "01-2013" "01-2014" "01-2015" ...  
## $ Airport : Factor w/ 2 levels "CORDOVA AIRPORT",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ Air\_MonthAvg: num [1:96] -4.53 -0.31 2.27 1.86 2.02 -2.33 0.32 -2 0.51 1.44 ...  
## $ Air\_MonthMax: num [1:96] 5.6 5.6 10 8.3 9.4 7.2 11.7 8.3 8.9 6.1 ...  
## $ Air\_MonthMin: num [1:96] -23.9 -14.4 -6.6 -11.6 -8.8 -16 -14.4 -16.6 -13.9 -5.6 ...  
## $ Region : chr [1:96] "YF" "YF" "YF" "YF" ...  
## $ NA : num [1:96] NA NA NA NA NA NA NA NA NA NA ...  
## $ MP1 : num [1:96] NA 0.12 NA NA -0.16 1.37 NA 1.1 NA 0.21 ...  
## $ MP3 : num [1:96] NA 0.65 0.16 NA NA NA NA 2.8 NA 0.61 ...  
## $ PL3 : num [1:96] NA -0.4 -0.21 2.77 NA NA NA 1.35 NA -0.27 ...  
## $ PL2 : num [1:96] NA -0.4 -0.17 NA NA NA NA -0.21 NA -0.27 ...  
## $ UBP3 : num [1:96] NA -0.54 NA 2.34 NA NA NA 3.13 NA 0.16 ...  
## $ MP5 : num [1:96] NA 0.43 1.44 NA NA 0.96 NA NA NA 0.38 ...  
## $ MP8 : num [1:96] NA 0.52 0.83 NA 1.4 NA NA 0.68 NA 0.36 ...  
## $ UBP1 : num [1:96] NA 0.02 NA 2.97 NA NA NA 0.78 NA 0.13 ...  
## $ UBP2 : num [1:96] NA 0.21 0.14 2.58 -0.23 1.81 NA 2.71 NA 0.15 ...  
## $ PL1 : num [1:96] NA NA -0.06 NA -0.51 NA NA 1.3 NA NA ...  
## $ UBP4 : num [1:96] NA NA NA NA NA NA NA 2.89 NA NA ...

test <- YF\_PondAirTemp\_mean  
 # pulling out the dates here  
 test$`Month-Year` <- as.yearmon(test$`Month-Year`, "%m-%Y")  
 test$`Month-Year` <- as.Date(as.yearmon(test$`Month-Year`, "%m-%Y"))  
 test <- test %>%  
 rename(Date = `Month-Year`)  
 # save over the Original File  
 YF\_PondAirTemp\_mean <- test  
 # Sort the dataframe by Date  
 YF\_PondAirTemp\_mean <- YF\_PondAirTemp\_mean[order(YF\_PondAirTemp\_mean$Date), ]  
 # Reset row names  
 rownames(YF\_PondAirTemp\_mean) <- NULL  
 # check the final file  
 head(YF\_PondAirTemp\_mean)

## # A tibble: 6 × 18  
## Date Airport Air\_MonthAvg Air\_MonthMax Air\_MonthMin Region `NA` MP1  
## <date> <fct> <dbl> <dbl> <dbl> <chr> <dbl> <dbl>  
## 1 2012-01-01 YAKUTAT … -4.53 5.6 -23.9 YF NA NA  
## 2 2012-02-01 YAKUTAT … 0.51 8.9 -13.9 YF NA NA  
## 3 2012-03-01 YAKUTAT … -1.39 10 -13.3 YF NA NA  
## 4 2012-04-01 YAKUTAT … 3.35 13.9 -7.2 YF NA NA  
## 5 2012-05-01 YAKUTAT … 5.56 13.9 -1.1 YF NA NA  
## 6 2012-06-01 YAKUTAT … 9.22 18.3 0 YF NA NA  
## # ℹ 10 more variables: MP3 <dbl>, PL3 <dbl>, PL2 <dbl>, UBP3 <dbl>, MP5 <dbl>,  
## # MP8 <dbl>, UBP1 <dbl>, UBP2 <dbl>, PL1 <dbl>, UBP4 <dbl>

# 2. max values  
str(YF\_PondAirTemp\_max)

## tibble [96 × 18] (S3: tbl\_df/tbl/data.frame)  
## $ Month-Year : chr [1:96] "01-2012" "01-2013" "01-2014" "01-2015" ...  
## $ Airport : Factor w/ 2 levels "CORDOVA AIRPORT",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ Air\_MonthAvg: num [1:96] -4.53 -0.31 2.27 1.86 2.02 -2.33 0.32 -2 0.51 1.44 ...  
## $ Air\_MonthMax: num [1:96] 5.6 5.6 10 8.3 9.4 7.2 11.7 8.3 8.9 6.1 ...  
## $ Air\_MonthMin: num [1:96] -23.9 -14.4 -6.6 -11.6 -8.8 -16 -14.4 -16.6 -13.9 -5.6 ...  
## $ Region : chr [1:96] "YF" "YF" "YF" "YF" ...  
## $ NA : num [1:96] NA NA NA NA NA NA NA NA NA NA ...  
## $ MP1 : num [1:96] NA 0.23 NA NA 0.56 2.26 NA 1.56 NA 0.23 ...  
## $ MP3 : num [1:96] NA 0.78 0.23 NA NA NA NA 3.46 NA 0.67 ...  
## $ PL3 : num [1:96] NA -0.06 0.89 3.47 NA NA NA 2.15 NA 2.98 ...  
## $ PL2 : num [1:96] NA -0.06 0.12 NA NA NA NA 0.66 NA 2.98 ...  
## $ UBP3 : num [1:96] NA 0.67 NA 5.04 NA NA NA 3.7 NA 2.62 ...  
## $ MP5 : num [1:96] NA 0.67 1.66 NA NA 1.32 NA NA NA 0.89 ...  
## $ MP8 : num [1:96] NA 0.67 1.98 NA 1.66 NA NA 1.53 NA 0.45 ...  
## $ UBP1 : num [1:96] NA 0.12 NA 4.1 NA NA NA 1.8 NA 0.23 ...  
## $ UBP2 : num [1:96] NA 0.23 0.67 2.94 0.67 2.64 NA 3.56 NA 0.23 ...  
## $ PL1 : num [1:96] NA NA 0.01 NA 0.6 NA NA 1.64 NA NA ...  
## $ UBP4 : num [1:96] NA NA NA NA NA NA NA 3.46 NA NA ...

test <- YF\_PondAirTemp\_max  
 # pulling out the dates here  
 test$`Month-Year` <- as.yearmon(test$`Month-Year`, "%m-%Y")  
 test$`Month-Year` <- as.Date(as.yearmon(test$`Month-Year`, "%m-%Y"))  
 test <- test %>%  
 rename(Date = `Month-Year`)  
 # save over the Original File  
 YF\_PondAirTemp\_max <- test  
 # Sort the dataframe by Date  
 YF\_PondAirTemp\_max <- YF\_PondAirTemp\_max[order(YF\_PondAirTemp\_max$Date), ]  
 # Reset row names  
 rownames(YF\_PondAirTemp\_max) <- NULL  
 # check the final file  
 head(YF\_PondAirTemp\_max)

## # A tibble: 6 × 18  
## Date Airport Air\_MonthAvg Air\_MonthMax Air\_MonthMin Region `NA` MP1  
## <date> <fct> <dbl> <dbl> <dbl> <chr> <dbl> <dbl>  
## 1 2012-01-01 YAKUTAT … -4.53 5.6 -23.9 YF NA NA  
## 2 2012-02-01 YAKUTAT … 0.51 8.9 -13.9 YF NA NA  
## 3 2012-03-01 YAKUTAT … -1.39 10 -13.3 YF NA NA  
## 4 2012-04-01 YAKUTAT … 3.35 13.9 -7.2 YF NA NA  
## 5 2012-05-01 YAKUTAT … 5.56 13.9 -1.1 YF NA NA  
## 6 2012-06-01 YAKUTAT … 9.22 18.3 0 YF NA NA  
## # ℹ 10 more variables: MP3 <dbl>, PL3 <dbl>, PL2 <dbl>, UBP3 <dbl>, MP5 <dbl>,  
## # MP8 <dbl>, UBP1 <dbl>, UBP2 <dbl>, PL1 <dbl>, UBP4 <dbl>

# 3. min values  
str(YF\_PondAirTemp\_min)

## tibble [96 × 18] (S3: tbl\_df/tbl/data.frame)  
## $ Month-Year : chr [1:96] "01-2012" "01-2013" "01-2014" "01-2015" ...  
## $ Airport : Factor w/ 2 levels "CORDOVA AIRPORT",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ Air\_MonthAvg: num [1:96] -4.53 -0.31 2.27 1.86 2.02 -2.33 0.32 -2 0.51 1.44 ...  
## $ Air\_MonthMax: num [1:96] 5.6 5.6 10 8.3 9.4 7.2 11.7 8.3 8.9 6.1 ...  
## $ Air\_MonthMin: num [1:96] -23.9 -14.4 -6.6 -11.6 -8.8 -16 -14.4 -16.6 -13.9 -5.6 ...  
## $ Region : chr [1:96] "YF" "YF" "YF" "YF" ...  
## $ NA : num [1:96] NA NA NA NA NA NA NA NA NA NA ...  
## $ MP1 : num [1:96] NA -0.55 NA NA -2.61 0.63 NA 0.82 NA 0.12 ...  
## $ MP3 : num [1:96] NA 0.45 0.12 NA NA NA NA 2.26 NA 0.34 ...  
## $ PL3 : num [1:96] NA -3.12 -6.75 1.87 NA ...  
## $ PL2 : num [1:96] NA -3.12 -4.88 NA NA ...  
## $ UBP3 : num [1:96] NA -8.45 NA 1.11 NA NA NA 2.69 NA -7.14 ...  
## $ MP5 : num [1:96] NA 0.23 1.11 NA NA 0.55 NA NA NA 0.23 ...  
## $ MP8 : num [1:96] NA 0.45 0.34 NA 1.22 NA NA 0.25 NA 0.23 ...  
## $ UBP1 : num [1:96] NA -1.23 NA 2.09 NA NA NA 0.05 NA -0.1 ...  
## $ UBP2 : num [1:96] NA 0.01 -3.32 2.3 -5.87 0.99 NA 1.51 NA -2.96 ...  
## $ PL1 : num [1:96] NA NA -3.55 NA -4.41 NA NA 0.14 NA NA ...  
## $ UBP4 : num [1:96] NA NA NA NA NA NA NA 0.85 NA NA ...

test <- YF\_PondAirTemp\_min  
 # pulling out the dates here  
 test$`Month-Year` <- as.yearmon(test$`Month-Year`, "%m-%Y")  
 test$`Month-Year` <- as.Date(as.yearmon(test$`Month-Year`, "%m-%Y"))  
 test <- test %>%  
 rename(Date = `Month-Year`)  
 # save over the Original File  
 YF\_PondAirTemp\_min <- test  
 # Sort the dataframe by Date  
 YF\_PondAirTemp\_min <- YF\_PondAirTemp\_min[order(YF\_PondAirTemp\_min$Date), ]  
 # Reset row names  
 rownames(YF\_PondAirTemp\_min) <- NULL  
 # check the final file  
 head(YF\_PondAirTemp\_min)

## # A tibble: 6 × 18  
## Date Airport Air\_MonthAvg Air\_MonthMax Air\_MonthMin Region `NA` MP1  
## <date> <fct> <dbl> <dbl> <dbl> <chr> <dbl> <dbl>  
## 1 2012-01-01 YAKUTAT … -4.53 5.6 -23.9 YF NA NA  
## 2 2012-02-01 YAKUTAT … 0.51 8.9 -13.9 YF NA NA  
## 3 2012-03-01 YAKUTAT … -1.39 10 -13.3 YF NA NA  
## 4 2012-04-01 YAKUTAT … 3.35 13.9 -7.2 YF NA NA  
## 5 2012-05-01 YAKUTAT … 5.56 13.9 -1.1 YF NA NA  
## 6 2012-06-01 YAKUTAT … 9.22 18.3 0 YF NA NA  
## # ℹ 10 more variables: MP3 <dbl>, PL3 <dbl>, PL2 <dbl>, UBP3 <dbl>, MP5 <dbl>,  
## # MP8 <dbl>, UBP1 <dbl>, UBP2 <dbl>, PL1 <dbl>, UBP4 <dbl>

## Section 2 - Copper River Delta

### Set up the file name and check the structure

CR\_Data <- df2d  
str(CR\_Data)

## 'data.frame': 450 obs. of 10 variables:  
## $ monyr : chr "01-2012" "01-2012" "01-2012" "01-2012" ...  
## $ NAME : Factor w/ 2 levels "CORDOVA AIRPORT",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Amonavg: num -8.22 -8.22 -8.22 -8.22 -0.43 -0.43 -0.43 -0.43 -0.43 -0.43 ...  
## $ Amonmax: num 4.4 4.4 4.4 4.4 8.3 8.3 8.3 8.3 8.3 8.3 ...  
## $ Amonmin: num -23.9 -23.9 -23.9 -23.9 -22.2 -22.2 -22.2 -22.2 -22.2 -22.2 ...  
## $ site : chr "TIN" "SQR" "TIS" "CAB" ...  
## $ monavg : num 0.07 0.53 0.32 -0.38 1.86 0.48 1.39 0.37 2.12 1.1 ...  
## $ monmax : num 0.91 1.07 0.93 0.14 2.41 0.67 1.66 0.56 2.41 1.66 ...  
## $ monmin : num -0.09 0.27 0.14 -2.31 1.55 0.23 0.89 0.23 1.98 0.72 ...  
## $ Region : chr "CR" "CR" "CR" "CR" ...

## Need to make wider by site and change some column names

### Starting with changing column names to be clearer

CR\_AirPondData <- CR\_Data %>%  
 rename(  
 `Month-Year` = monyr,  
 Pond = site,  
 Water\_MonthAvg = monavg,  
 Water\_MonthMax = monmax,  
 Water\_MonthMin = monmin,  
 Airport = NAME,  
 Air\_MonthAvg = Amonavg,  
 Air\_MonthMax = Amonmax,  
 Air\_MonthMin = Amonmin  
 )  
  
str(CR\_AirPondData)

## 'data.frame': 450 obs. of 10 variables:  
## $ Month-Year : chr "01-2012" "01-2012" "01-2012" "01-2012" ...  
## $ Airport : Factor w/ 2 levels "CORDOVA AIRPORT",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Air\_MonthAvg : num -8.22 -8.22 -8.22 -8.22 -0.43 -0.43 -0.43 -0.43 -0.43 -0.43 ...  
## $ Air\_MonthMax : num 4.4 4.4 4.4 4.4 8.3 8.3 8.3 8.3 8.3 8.3 ...  
## $ Air\_MonthMin : num -23.9 -23.9 -23.9 -23.9 -22.2 -22.2 -22.2 -22.2 -22.2 -22.2 ...  
## $ Pond : chr "TIN" "SQR" "TIS" "CAB" ...  
## $ Water\_MonthAvg: num 0.07 0.53 0.32 -0.38 1.86 0.48 1.39 0.37 2.12 1.1 ...  
## $ Water\_MonthMax: num 0.91 1.07 0.93 0.14 2.41 0.67 1.66 0.56 2.41 1.66 ...  
## $ Water\_MonthMin: num -0.09 0.27 0.14 -2.31 1.55 0.23 0.89 0.23 1.98 0.72 ...  
## $ Region : chr "CR" "CR" "CR" "CR" ...

### Pivot the dataframe wider to make a column for each ponds temepratures

CR\_PondAirTemp\_mean <- CR\_AirPondData %>%  
 pivot\_wider(  
 id\_cols = c("Month-Year", "Airport", "Air\_MonthAvg", "Air\_MonthMax", "Air\_MonthMin", "Region"),  
 names\_from = Pond,  
 values\_from = c(Water\_MonthAvg)  
 )  
  
CR\_PondAirTemp\_max <- CR\_AirPondData %>%  
 pivot\_wider(  
 id\_cols = c("Month-Year", "Airport", "Air\_MonthAvg", "Air\_MonthMax", "Air\_MonthMin", "Region"),  
 names\_from = Pond,  
 values\_from = c(Water\_MonthMax)  
 )  
  
CR\_PondAirTemp\_min <- CR\_AirPondData %>%  
 pivot\_wider(  
 id\_cols = c("Month-Year", "Airport", "Air\_MonthAvg", "Air\_MonthMax", "Air\_MonthMin", "Region"),  
 names\_from = Pond,  
 values\_from = c(Water\_MonthMin)  
 )

### Correct the Data formats and order these by date here

# 1. mean values  
str(CR\_PondAirTemp\_mean)

## tibble [96 × 16] (S3: tbl\_df/tbl/data.frame)  
## $ Month-Year : chr [1:96] "01-2012" "01-2013" "01-2014" "01-2015" ...  
## $ Airport : Factor w/ 2 levels "CORDOVA AIRPORT",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Air\_MonthAvg: num [1:96] -8.22 -0.43 2.34 -0.91 2.16 -4.73 -1.48 -2.62 -0.17 1.39 ...  
## $ Air\_MonthMax: num [1:96] 4.4 8.3 13.9 9.4 8.3 5 10.6 8.3 8.9 7.2 ...  
## $ Air\_MonthMin: num [1:96] -23.9 -22.2 -9.3 -14.3 -7.7 -19.3 -14.9 -18.8 -13.9 -6.1 ...  
## $ Region : chr [1:96] "CR" "CR" "CR" "CR" ...  
## $ TIN : num [1:96] 0.07 1.1 1 NA NA NA NA NA -0.06 1.01 ...  
## $ SQR : num [1:96] 0.53 1.86 2.29 NA NA NA NA NA 0.18 2.35 ...  
## $ TIS : num [1:96] 0.32 1.39 0.62 NA 2.02 -1.12 0.89 NA 0.3 1.37 ...  
## $ CAB : num [1:96] -0.38 0.04 0.51 0.67 1.23 0.07 0.38 0.43 0.05 0.16 ...  
## $ EYS : num [1:96] NA 0.48 1.82 NA NA NA NA 2.63 NA 0.44 ...  
## $ RHM : num [1:96] NA 0.37 0.67 1.57 1.44 1.07 0.37 0.33 NA 0.23 ...  
## $ BVS : num [1:96] NA 2.12 2.56 NA NA NA NA NA NA 2.06 ...  
## $ CAN : num [1:96] NA NA NA NA NA -0.56 NA 1.58 NA NA ...  
## $ WDD : num [1:96] NA NA NA NA NA 0.81 NA 1.06 NA NA ...  
## $ NA : num [1:96] NA NA NA NA NA NA NA NA NA NA ...

test <- CR\_PondAirTemp\_mean  
 # pulling out the dates here  
 test$`Month-Year` <- as.yearmon(test$`Month-Year`, "%m-%Y")  
 test$`Month-Year` <- as.Date(as.yearmon(test$`Month-Year`, "%m-%Y"))  
 test <- test %>%  
 rename(Date = `Month-Year`)  
 # save over the Original File  
 CR\_PondAirTemp\_mean <- test  
 # Sort the dataframe by Date  
 CR\_PondAirTemp\_mean <- CR\_PondAirTemp\_mean[order(CR\_PondAirTemp\_mean$Date), ]  
 # Reset row names  
 rownames(CR\_PondAirTemp\_mean) <- NULL  
 # check the final file  
 head(CR\_PondAirTemp\_mean)

## # A tibble: 6 × 16  
## Date Airport Air\_MonthAvg Air\_MonthMax Air\_MonthMin Region TIN SQR  
## <date> <fct> <dbl> <dbl> <dbl> <chr> <dbl> <dbl>  
## 1 2012-01-01 CORDOVA … -8.22 4.4 -23.9 CR 0.07 0.53  
## 2 2012-02-01 CORDOVA … -0.17 8.9 -13.9 CR -0.06 0.18  
## 3 2012-03-01 CORDOVA … -2.9 10 -16.1 CR 0 0.04  
## 4 2012-04-01 CORDOVA … 2.59 13.9 -8.9 CR -0.02 1.19  
## 5 2012-05-01 CORDOVA … 5.85 16.1 -2.8 CR 5.96 8.08  
## 6 2012-06-01 CORDOVA … 9.52 22.8 -0.6 CR 14.9 14.5   
## # ℹ 8 more variables: TIS <dbl>, CAB <dbl>, EYS <dbl>, RHM <dbl>, BVS <dbl>,  
## # CAN <dbl>, WDD <dbl>, `NA` <dbl>

# 2. max values  
str(CR\_PondAirTemp\_max)

## tibble [96 × 16] (S3: tbl\_df/tbl/data.frame)  
## $ Month-Year : chr [1:96] "01-2012" "01-2013" "01-2014" "01-2015" ...  
## $ Airport : Factor w/ 2 levels "CORDOVA AIRPORT",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Air\_MonthAvg: num [1:96] -8.22 -0.43 2.34 -0.91 2.16 -4.73 -1.48 -2.62 -0.17 1.39 ...  
## $ Air\_MonthMax: num [1:96] 4.4 8.3 13.9 9.4 8.3 5 10.6 8.3 8.9 7.2 ...  
## $ Air\_MonthMin: num [1:96] -23.9 -22.2 -9.3 -14.3 -7.7 -19.3 -14.9 -18.8 -13.9 -6.1 ...  
## $ Region : chr [1:96] "CR" "CR" "CR" "CR" ...  
## $ TIN : num [1:96] 0.91 1.66 4.83 NA NA NA NA NA 0.05 2.09 ...  
## $ SQR : num [1:96] 1.07 2.41 3.68 NA NA NA NA NA 0.47 3.58 ...  
## $ TIS : num [1:96] 0.93 1.66 4.73 NA 3.51 0.05 1.53 NA 0.41 1.76 ...  
## $ CAB : num [1:96] 0.14 0.16 1.72 1.43 2.45 0.19 1.18 1.45 0.41 0.52 ...  
## $ EYS : num [1:96] NA 0.67 3.26 NA NA NA NA 3.26 NA 0.89 ...  
## $ RHM : num [1:96] NA 0.56 2.84 1.98 4 1.7 1.07 1.4 NA 0.34 ...  
## $ BVS : num [1:96] NA 2.41 3.37 NA NA NA NA NA NA 2.19 ...  
## $ CAN : num [1:96] NA NA NA NA NA 0.23 NA 2.3 NA NA ...  
## $ WDD : num [1:96] NA NA NA NA NA 1.44 NA 1.66 NA NA ...  
## $ NA : num [1:96] NA NA NA NA NA NA NA NA NA NA ...

test <- CR\_PondAirTemp\_max  
 # pulling out the dates here  
 test$`Month-Year` <- as.yearmon(test$`Month-Year`, "%m-%Y")  
 test$`Month-Year` <- as.Date(as.yearmon(test$`Month-Year`, "%m-%Y"))  
 test <- test %>%  
 rename(Date = `Month-Year`)  
 # save over the Original File  
 CR\_PondAirTemp\_max <- test  
 # Sort the dataframe by Date  
 CR\_PondAirTemp\_max <- CR\_PondAirTemp\_max[order(CR\_PondAirTemp\_max$Date), ]  
 # Reset row names  
 rownames(CR\_PondAirTemp\_max) <- NULL  
 # check the final file  
 head(CR\_PondAirTemp\_max)

## # A tibble: 6 × 16  
## Date Airport Air\_MonthAvg Air\_MonthMax Air\_MonthMin Region TIN SQR  
## <date> <fct> <dbl> <dbl> <dbl> <chr> <dbl> <dbl>  
## 1 2012-01-01 CORDOVA … -8.22 4.4 -23.9 CR 0.91 1.07  
## 2 2012-02-01 CORDOVA … -0.17 8.9 -13.9 CR 0.05 0.47  
## 3 2012-03-01 CORDOVA … -2.9 10 -16.1 CR 0.02 0.11  
## 4 2012-04-01 CORDOVA … 2.59 13.9 -8.9 CR 0.14 4.17  
## 5 2012-05-01 CORDOVA … 5.85 16.1 -2.8 CR 16.4 15.6   
## 6 2012-06-01 CORDOVA … 9.52 22.8 -0.6 CR 21.9 20.8   
## # ℹ 8 more variables: TIS <dbl>, CAB <dbl>, EYS <dbl>, RHM <dbl>, BVS <dbl>,  
## # CAN <dbl>, WDD <dbl>, `NA` <dbl>

# 3. min values  
str(CR\_PondAirTemp\_min)

## tibble [96 × 16] (S3: tbl\_df/tbl/data.frame)  
## $ Month-Year : chr [1:96] "01-2012" "01-2013" "01-2014" "01-2015" ...  
## $ Airport : Factor w/ 2 levels "CORDOVA AIRPORT",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Air\_MonthAvg: num [1:96] -8.22 -0.43 2.34 -0.91 2.16 -4.73 -1.48 -2.62 -0.17 1.39 ...  
## $ Air\_MonthMax: num [1:96] 4.4 8.3 13.9 9.4 8.3 5 10.6 8.3 8.9 7.2 ...  
## $ Air\_MonthMin: num [1:96] -23.9 -22.2 -9.3 -14.3 -7.7 -19.3 -14.9 -18.8 -13.9 -6.1 ...  
## $ Region : chr [1:96] "CR" "CR" "CR" "CR" ...  
## $ TIN : num [1:96] -0.09 0.72 -3.79 NA NA NA NA NA -0.09 0.25 ...  
## $ SQR : num [1:96] 0.27 1.55 1.87 NA NA NA NA NA 0.08 1.76 ...  
## $ TIS : num [1:96] 0.14 0.89 -3.32 NA 0.85 -3.81 0.36 NA 0.14 0.23 ...  
## $ CAB : num [1:96] -2.31 -0.03 0 0.05 0.63 0.02 0.08 0.16 -2.31 -0.03 ...  
## $ EYS : num [1:96] NA 0.23 0.56 NA NA NA NA 0.56 NA 0.12 ...  
## $ RHM : num [1:96] NA 0.23 0.12 1 0.14 0.63 0.02 0.02 NA 0.12 ...  
## $ BVS : num [1:96] NA 1.98 0.56 NA NA NA NA NA NA 1.98 ...  
## $ CAN : num [1:96] NA NA NA NA NA -2.49 NA 0.34 NA NA ...  
## $ WDD : num [1:96] NA NA NA NA NA 0.12 NA 0.23 NA NA ...  
## $ NA : num [1:96] NA NA NA NA NA NA NA NA NA NA ...

test <- CR\_PondAirTemp\_min  
 # pulling out the dates here  
 test$`Month-Year` <- as.yearmon(test$`Month-Year`, "%m-%Y")  
 test$`Month-Year` <- as.Date(as.yearmon(test$`Month-Year`, "%m-%Y"))  
 test <- test %>%  
 rename(Date = `Month-Year`)  
 # save over the Original File  
 CR\_PondAirTemp\_min <- test  
 # Sort the dataframe by Date  
 CR\_PondAirTemp\_min <- CR\_PondAirTemp\_min[order(CR\_PondAirTemp\_min$Date), ]  
 # Reset row names  
 rownames(CR\_PondAirTemp\_min) <- NULL  
 # check the final file  
 head(CR\_PondAirTemp\_min)

## # A tibble: 6 × 16  
## Date Airport Air\_MonthAvg Air\_MonthMax Air\_MonthMin Region TIN SQR  
## <date> <fct> <dbl> <dbl> <dbl> <chr> <dbl> <dbl>  
## 1 2012-01-01 CORDOVA … -8.22 4.4 -23.9 CR -0.09 0.27  
## 2 2012-02-01 CORDOVA … -0.17 8.9 -13.9 CR -0.09 0.08  
## 3 2012-03-01 CORDOVA … -2.9 10 -16.1 CR -0.06 0.02  
## 4 2012-04-01 CORDOVA … 2.59 13.9 -8.9 CR -0.09 0.02  
## 5 2012-05-01 CORDOVA … 5.85 16.1 -2.8 CR -0.03 1.91  
## 6 2012-06-01 CORDOVA … 9.52 22.8 -0.6 CR 9.61 9.66  
## # ℹ 8 more variables: TIS <dbl>, CAB <dbl>, EYS <dbl>, RHM <dbl>, BVS <dbl>,  
## # CAN <dbl>, WDD <dbl>, `NA` <dbl>

# Part 6 - Adding in a column that is the std of the air temperature in the mean files

AW recommended doing that as std = (max - min)/4 AA increased the denominator here because the error for the observed data is way higher than for the forecastsed data

Copper River Delta

CR\_PondAirTemp\_mean

## # A tibble: 96 × 16  
## Date Airport Air\_MonthAvg Air\_MonthMax Air\_MonthMin Region TIN SQR  
## <date> <fct> <dbl> <dbl> <dbl> <chr> <dbl> <dbl>  
## 1 2012-01-01 CORDOVA… -8.22 4.4 -23.9 CR 0.07 0.53  
## 2 2012-02-01 CORDOVA… -0.17 8.9 -13.9 CR -0.06 0.18  
## 3 2012-03-01 CORDOVA… -2.9 10 -16.1 CR 0 0.04  
## 4 2012-04-01 CORDOVA… 2.59 13.9 -8.9 CR -0.02 1.19  
## 5 2012-05-01 CORDOVA… 5.85 16.1 -2.8 CR 5.96 8.08  
## 6 2012-06-01 CORDOVA… 9.52 22.8 -0.6 CR 14.9 14.5   
## 7 2012-07-01 CORDOVA… 10.7 20 5 CR 14.1 13.9   
## 8 2012-08-01 CORDOVA… 11.2 18.9 0.6 CR 14.7 14.4   
## 9 2012-09-01 CORDOVA… 8.78 17.2 -2.2 CR 10.1 10.0   
## 10 2012-10-01 CORDOVA… 1.5 12.8 -12.2 CR 5.97 5.62  
## # ℹ 86 more rows  
## # ℹ 8 more variables: TIS <dbl>, CAB <dbl>, EYS <dbl>, RHM <dbl>, BVS <dbl>,  
## # CAN <dbl>, WDD <dbl>, `NA` <dbl>

test <- CR\_PondAirTemp\_mean  
  
test$std <- (test$Air\_MonthMax - test$Air\_MonthMin)/10  
test$precision <- (1/(test$std^2))  
  
CR\_PondAirTemp\_mean <- test

Yakutat Forelands

YF\_PondAirTemp\_mean

## # A tibble: 96 × 18  
## Date Airport Air\_MonthAvg Air\_MonthMax Air\_MonthMin Region `NA` MP1  
## <date> <fct> <dbl> <dbl> <dbl> <chr> <dbl> <dbl>  
## 1 2012-01-01 YAKUTAT… -4.53 5.6 -23.9 YF NA NA   
## 2 2012-02-01 YAKUTAT… 0.51 8.9 -13.9 YF NA NA   
## 3 2012-03-01 YAKUTAT… -1.39 10 -13.3 YF NA NA   
## 4 2012-04-01 YAKUTAT… 3.35 13.9 -7.2 YF NA NA   
## 5 2012-05-01 YAKUTAT… 5.56 13.9 -1.1 YF NA NA   
## 6 2012-06-01 YAKUTAT… 9.22 18.3 0 YF NA NA   
## 7 2012-07-01 YAKUTAT… 11.2 18.9 3.9 YF NA NA   
## 8 2012-08-01 YAKUTAT… 11.4 19.4 0.6 YF NA 16.2   
## 9 2012-09-01 YAKUTAT… 9.16 16.1 0.6 YF NA 11.9   
## 10 2012-10-01 YAKUTAT… 2.79 11.1 -11.1 YF NA 6.62  
## # ℹ 86 more rows  
## # ℹ 10 more variables: MP3 <dbl>, PL3 <dbl>, PL2 <dbl>, UBP3 <dbl>, MP5 <dbl>,  
## # MP8 <dbl>, UBP1 <dbl>, UBP2 <dbl>, PL1 <dbl>, UBP4 <dbl>

test <- YF\_PondAirTemp\_mean  
  
test$std <- (test$Air\_MonthMax - test$Air\_MonthMin)/10  
test$precision <- (1/(test$std^2))  
  
YF\_PondAirTemp\_mean <- test

# Part 6 - Save the data files as .csv files to be loaded into other code here

write.csv(CR\_PondAirTemp\_min, file = "DataFiles/CR\_PondAirTemp\_min.csv")  
write.csv(CR\_PondAirTemp\_max, file = "DataFiles/CR\_PondAirTemp\_max.csv")  
write.csv(CR\_PondAirTemp\_mean, file = "DataFiles/CR\_PondAirTemp\_mean.csv")  
  
write.csv(YF\_PondAirTemp\_min, file = "DataFiles/YF\_PondAirTemp\_min.csv")  
write.csv(YF\_PondAirTemp\_max, file = "DataFiles/YF\_PondAirTemp\_max.csv")  
write.csv(YF\_PondAirTemp\_mean, file = "DataFiles/YF\_PondAirTemp\_mean.csv")