

N. Umpqua Temperature Analysis

Summer-run Steelhead Ecology and Conservation in the North Umpqua Basin: Genetic Insights and Management Opportunities, (Pepping et al., 2026)

J.Hart

2025-08-23

Water Logger IDs:

21679180 - Steamboat Above Big Bend Cr.

15170014 - Big Bend at mouth

21679175 - Steamboat below Big Bend Cr.

15170003 - Steamboat above Canton Cr.

21679177 - Upper Canton Cr.

21433141 - N. Umpqua above Steamboat Cr.

```
# Load packages
```

```
library(tidyverse)
library(readxl)
library(janitor)
library(writexl)
library(readr)
#library(tinytex)
```

```
# Read in 2024 temp logger data summaries
```

```
daily_water_master <- read_xlsx("2024 Logger Sites/Water/daily_water_master.xlsx")
daily_air_master <- read_xlsx("2024 Logger Sites/Air/daily_air_master.xlsx")
```

15110003 - Fish Cr. at mouth

Steamboat Above Big Bend Cr.

```
# Logger ID: 21679180 (WSC logger data)

# Read in 2024 WSC data and filter for summer season
Steam_AboveBigBend <- daily_water_master %>%
  filter(logger_id == 21679180 & longdate > '2024-06-19' & longdate <= '2024-09-15')

# 2024 max temp
Steam_AboveBigBend_MaxT <- max(Steam_AboveBigBend$max_temp)
print(paste("Max Temp C:", Steam_AboveBigBend_MaxT))

## [1] "Max Temp C: 25.27"

# 2024 number of days above 20C
Steam_AboveBigBend_DaysOver20 <- sum(Steam_AboveBigBend$max_temp > 20)
print(paste("# days above 20C (2024):", Steam_AboveBigBend_DaysOver20))

## [1] "# days above 20C (2024): 56"

### Mean, Min, Max Temp Plot

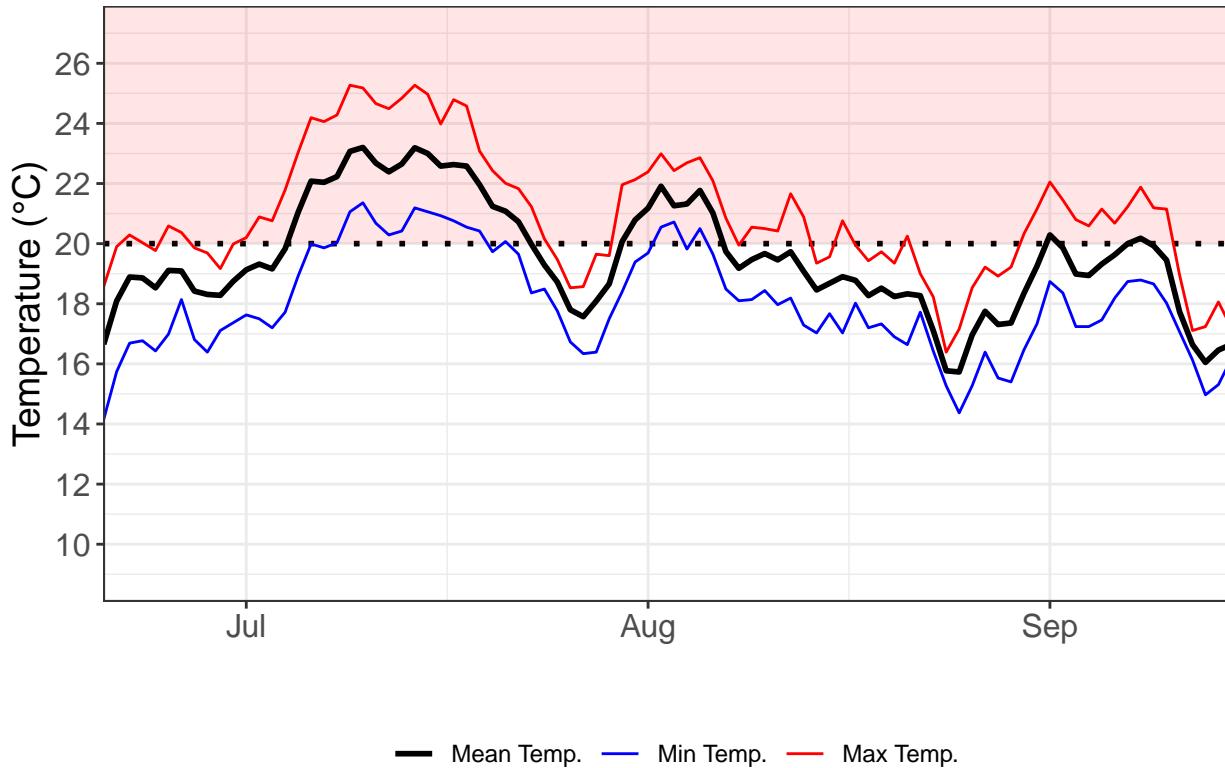
# Plot data frame
temp <- Steam_AboveBigBend

x=as.Date(temp$longdate)
y_lower = 20
y_upper = Inf

# Create temperature plot
plot_21679180 <- ggplot() +
  geom_hline(yintercept = 20, linetype = "dotted", linewidth = 1) +
  geom_ribbon(aes(x=x,ymin = y_lower, ymax = y_upper), fill = "red1", alpha = 0.10) +
  geom_line(data=temp, aes(x=as.Date(longdate),y=mean_temp, color="mean_temp"), linewidth = 1) +
  geom_line(data=temp, aes(x=as.Date(longdate),y=min_temp, color = "min_temp"), linewidth = 0.5) +
  geom_line(data=temp, aes(x=as.Date(longdate), y=max_temp, color= "max_temp"), linewidth = 0.5) +
  scale_colour_manual("", 
    breaks = c("mean_temp", "min_temp", "max_temp"),
    values = c("black", "blue", "red"),
    labels = c("Mean Temp.", "Min Temp.", "Max Temp."))
  +
  scale_x_date(date_breaks = "1 month", date_labels = "%b", expand = c(0,0)) +
  scale_y_continuous(limits = c(9,27),breaks = seq(10,26,by= 2)) +
  theme_bw() +
  labs(title= "Steamboat above Big Bend Cr.",
    y="Temperature (\u00b0C)",
    x="") +
  theme(plot.subtitle = element_text(size = 10),
    plot.title = element_text(size= 14, face="bold"),
    axis.text=element_text(size= 12),
    axis.title=element_text(size= 14),
    legend.position = "bottom")
```

```
rm(temp)
plot_21679180
```

Steamboat above Big Bend Cr.



```
# Export plot to file
# ggsave(plot_21679180, file="Figure Plots/Temp Plots/Temp Plot_SteamboatAboveBigBend.png", height = 5,
```

```
## Air/Water Regression Plot

# Prepare air logger data for regression
air <- daily_air_master %>%
  filter(logger_id == "21679162") %>% # N. Umpqua at Steamboat air logger
  mutate(longdate = make_date(year,month,day)) %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(air_mean = mean_temp)

# Prepare individual water logger data for regression
water <- daily_water_master %>%
  filter(logger_id == '21679180') %>%
  mutate(longdate = make_date(year,month,day)) %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(water_max = max_temp)

# Join air/water data frames for linear regression
```

```

airwater_join <- water %>%
  left_join(air, by = "longdate")

# Create equation formula for regression equation
lm_eqn <- function(df){
  m <- lm(water_max ~ air_mean, df);
  eq <- substitute(italic(y) == a + b %.% italic(x)*", "~~italic(r)^2~~="~r2,
    list(a = format(unname(coef(m)[1]), digits = 3),
        b = format(unname(coef(m)[2]), digits = 3),
        r2 = format(summary(m)$adj.r.squared, digits = 3)))
  as.character(as.expression(eq));
}

# Calculate Pearson correlation coefficient
correlation <- cor(airwater_join$water_max, airwater_join$air_mean, method = 'pearson')
print(paste("Pearson R:", correlation))

## [1] "Pearson R: 0.86044468252824"

# Linear model fit summary
fit <- lm(water_max ~ air_mean, data = airwater_join)
summary(fit)

##
## Call:
## lm(formula = water_max ~ air_mean, data = airwater_join)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.83245 -0.85138 -0.00655  0.64290  2.44061 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 7.97981   0.83619   9.543 3.86e-15 ***
## air_mean     0.64498   0.04119  15.660 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.077 on 86 degrees of freedom
## Multiple R-squared:  0.7404, Adjusted R-squared:  0.7373 
## F-statistic: 245.2 on 1 and 86 DF,  p-value: < 2.2e-16

# Create regression plot
reg_plot_21679180 <- ggplot(airwater_join, aes(x= air_mean, y = water_max)) +
  geom_point(shape= 21, color='black', fill= 'deepskyblue3', size = 2) +
  geom_smooth(method = "lm", se = TRUE, colour = 'black') +
  theme_light() +
  labs(title="Steamboat above Big Bend Cr. (Air - N. Umpqua at Steamboat)",
       subtitle = "Max Water Temp ~ Mean Air Temp",
       x="Daily Mean Air temperature (\u00B0C)",
       y="Daily Max Water temperature (\u00B0C)") +
  geom_text(x = 16, y = 24, label = lm_eqn(airwater_join), parse = TRUE) +

```

```

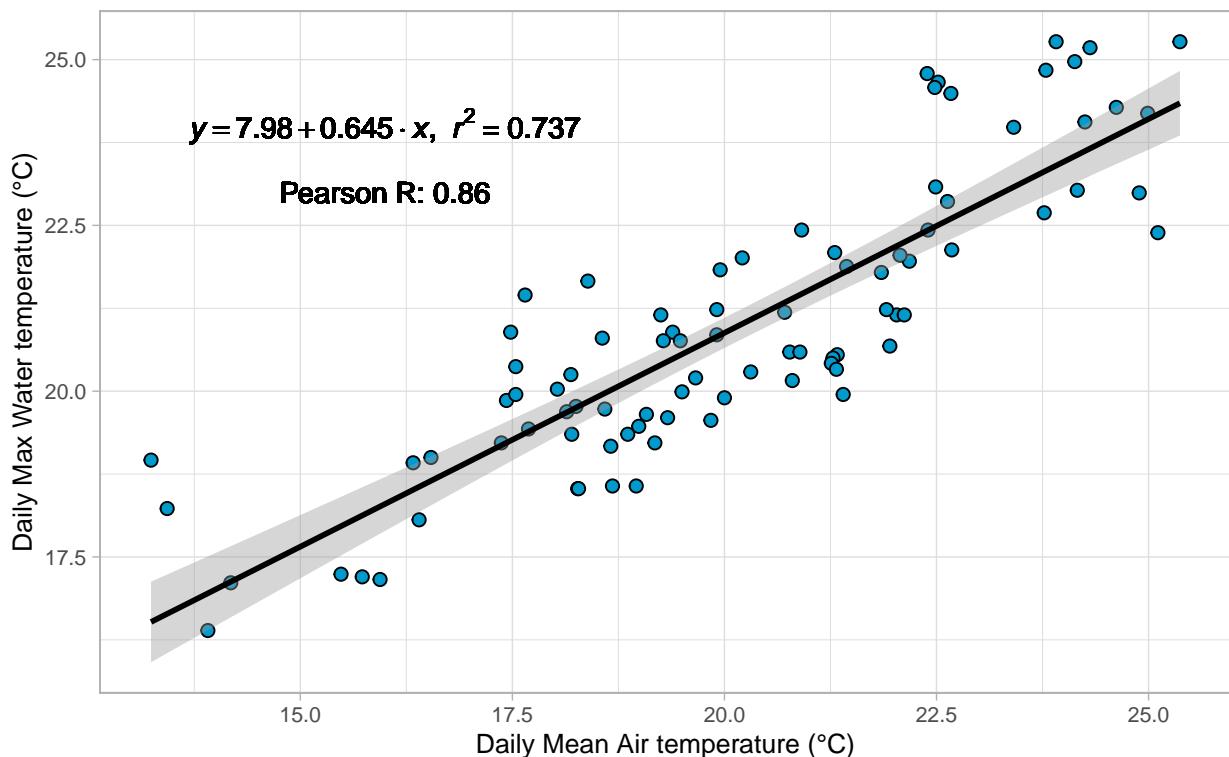
geom_text(x = 16, y = 23, label = paste("Pearson R:", round(correlation,3))) +
  theme(plot.title = element_text(size= 12, face="bold"),
    plot.subtitle = element_text(size = 11),
    axis.text=element_text(size= 8),
    axis.title=element_text(size= 10))

reg_plot_21679180

```

Steamboat above Big Bend Cr. (Air – N. Umpqua at Steamboat)

Max Water Temp ~ Mean Air Temp



```

# Export plot to file
# ggsave(reg_plot_21679180, file="Figure Plots/Regression Plots/regression_plot_SteamboatAboveBigBend.p

#-----
## Climate Scenario (3.5C air temp warming)

# Create vector of new air temps under 3.5C warming scenario
new_temps <- data.frame(air_mean = airwater_join$air_mean + 3.5)

# Use lm to predict new water temps using updated air temps
watertemp_pred <- predict(fit, newdata = new_temps)

# Estimate number of days water temp above 20C given future climate scenario
daysover20_future <- data.frame(daysover20 = sum(watertemp_pred > 20))
print(paste("Predicted # days above 20C (+3.5C future climate):", {daysover20_future$daysover20}))

```

```

## [1] "Predicted # days above 20C (+3.5C future climate): 84"

rm(new_temps)
rm(watertemp_pred)

```

Big Bend Cr. at mouth

```

# Logger ID: 15170014 (USGS hydro station data)

# Read in Big Bend data, clean, and summarize daily mean, min, max temp statistics

list_of_files <- list.files(path = "USB Files/Big Bend",
                           recursive = TRUE,
                           pattern = "\\.csv$",
                           full.names = TRUE)

BigBend <- readr::read_csv(list_of_files, id = "file_name") %>%
  clean_names() %>%
  mutate(site = "Big Bend Cr. at mouth") %>%
  mutate(temp_c = (temp - 32) * (5/9)) %>% #temp in F, need to convert to C
  mutate(logger_id = "15170014") %>%
  mutate(longdate = as.Date(date,format = "%m/%d/%Y")) %>%
  separate(longdate,
           into = c('year', 'month', 'day'),
           sep= '-',
           remove = FALSE) %>%
  mutate(month_day = format(longdate, "%m-%d")) %>%
  group_by(longdate, month_day, year, logger_id) %>%
  summarise(mean_temp = round(mean(temp_c), digits= 2),
            min_temp = round(min(temp_c), digits= 2),
            max_temp = round(max(temp_c), digits= 2))

## Calculate other temperature statistics

# Max temp (2024)
BigBend_maxt_2024 <- BigBend %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  group_by(logger_id) %>%
  summarise(maxt = max(max_temp))

print(paste("Max Temp C (2024):", BigBend_maxt_2024$maxt))

```

```

## [1] "Max Temp C (2024): 20.51"

```

```

# Days >20C (2024)
BigBend_Days20_2024 <- BigBend %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  filter(max_temp > 20) %>%
  group_by(logger_id) %>%
  summarise(days_over20 = n())

```

```

print(paste("# Days above 20C (2024):", BigBend_Days20_2024$days_over20))

## [1] "# Days above 20C (2024): 5"

# Max temp (10-yr Avg.)
BigBend_maxt_10yrAvg <- BigBend %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  group_by(logger_id, year) %>%
  summarise(maxt = max(max_temp)) %>%
  summarise(maxt_avg = mean(maxt))

print(paste("Max Temp C (10-yr Avg.):", BigBend_maxt_10yrAvg$maxt_avg))

## [1] "Max Temp C (10-yr Avg.): 18.687"

# Days >20C (10-yr Avg.)
BigBend_Days20_10yrAvg <- BigBend %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  filter(max_temp > 20) %>%
  group_by(logger_id, year) %>%
  summarise(days_over20 = n()) %>%
  summarise(days_over20_avg = sum(days_over20) / 10)

print(paste("# Days above 20C (10-yr Avg.):", BigBend_Days20_10yrAvg$days_over20_avg))

## [1] "# Days above 20C (10-yr Avg.): 0.6"

# Max Temp (Drought)
BigBend_maxt_drought <- BigBend %>%
  filter(year == '2021' | year == '2022') %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  group_by(logger_id) %>%
  summarise(maxt = max(max_temp))

print(paste("Max Temp C (Drought yrs.):", BigBend_maxt_drought$maxt))

## [1] "Max Temp C (Drought yrs.): 20.29"

# Days >20C (Drought)
BigBend_Days20_drought <- BigBend %>%
  filter(year == '2021' | year == '2022') %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  filter(max_temp > 20) %>%
  group_by(logger_id, year) %>%
  summarise(days_over20 = n()) %>%
  summarise(days_over20_avg = sum(days_over20) / 2)

print(paste("# Days above 20C (Drought yrs.):", BigBend_Days20_drought$days_over20_avg))

## [1] "# Days above 20C (Drought yrs.): 0.5"

```

```

## Mean, Min, Max Temp Plot

# Data frame for plot
temp <- BigBend %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15')

x $\leftarrow$ as.Date(temp$longdate)
y_lower = 20
y_upper = Inf

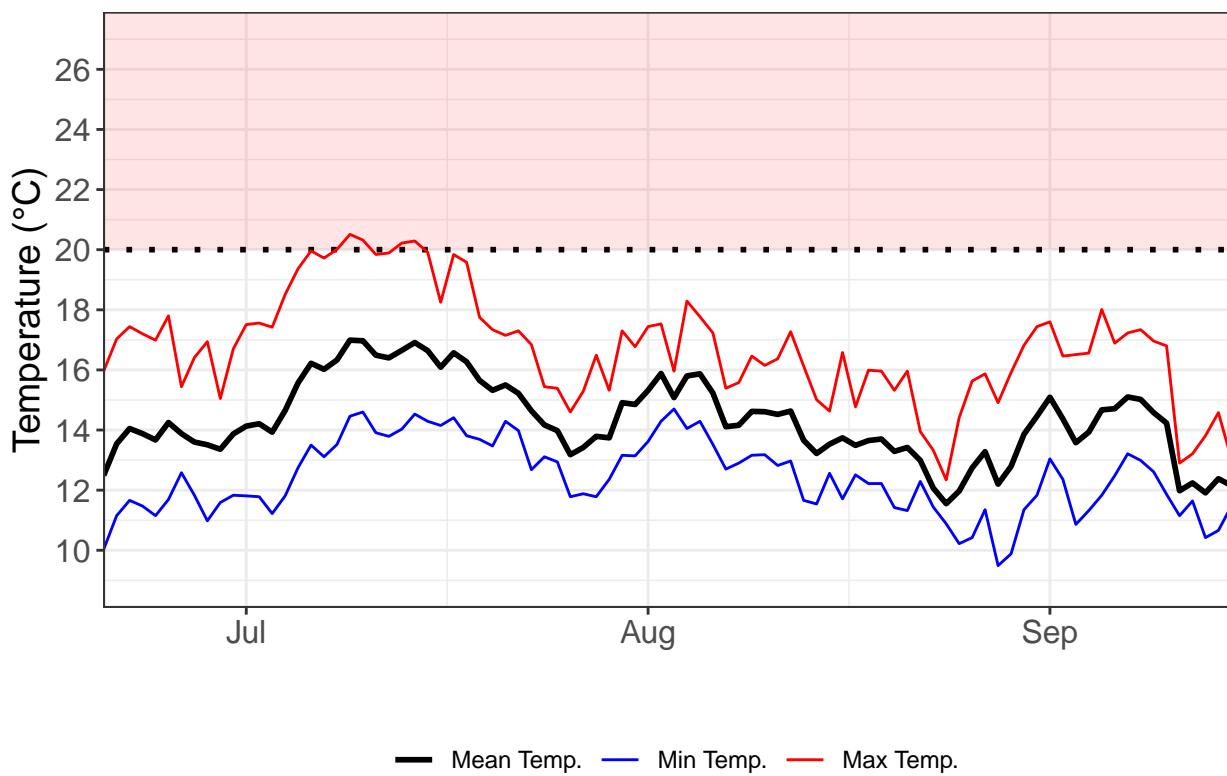
# Create temp plot
plot_15170014 <- ggplot() +
  geom_hline(yintercept = 20, linetype = "dotted", linewidth = 1) +
  geom_ribbon(aes(x=x,ymin = y_lower, ymax = y_upper), fill = "red1", alpha = 0.10) +
  geom_line(data=temp, aes(x=as.Date(longdate),y=mean_temp, color="mean_temp"), linewidth = 1) +
  geom_line(data=temp, aes(x=as.Date(longdate),y=min_temp, color = "min_temp"), linewidth = 0.5) +
  geom_line(data=temp, aes(x=as.Date(longdate), y=max_temp, color= "max_temp"), linewidth = 0.5) +
  scale_colour_manual("", 
    breaks = c("mean_temp", "min_temp", "max_temp"),
    values = c("black", "blue", "red"),
    labels = c("Mean Temp.", "Min Temp.", "Max Temp."))
  scale_x_date(date_breaks = "1 month", date_labels = "%b", expand = c(0,0)) +
  scale_y_continuous(limits = c(9,27),breaks = seq(10,26,by= 2)) +
  theme_bw() +
  labs(title= "Big Bend Cr. at mouth", y="Temperature (\u00B0C)", x="")
  theme(plot.subtitle = element_text(size = 10),
        plot.title = element_text(size= 14, face="bold"),
        axis.text=element_text(size= 12),
        axis.title=element_text(size= 14),
        legend.position = "bottom")

rm(temp)

plot_15170014

```

Big Bend Cr. at mouth



```
# Export plot to file
# ggsave(plot_15170014, file="Figure Plots/Temp Plots/Temp Plot_BigBend.png", height = 5, width = 8) #For R

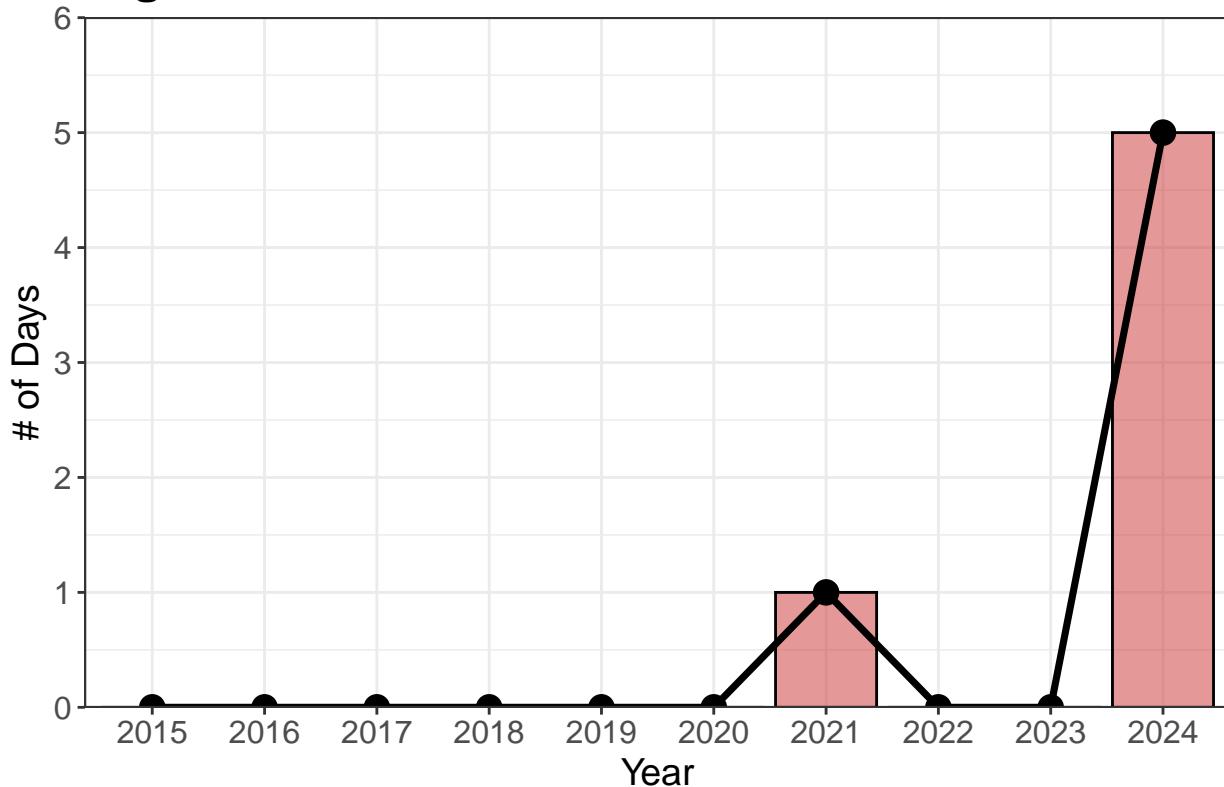
# -----
## Number of days above 20C plot

temp <- BigBend %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  group_by(year) %>%
  summarize(conditional_count = sum(max_temp > 20))

BigBend_Days20_Plot <- ggplot(data=temp) +
  geom_col(aes(x=as.factor(year),y=conditional_count), fill="brown3", color="black",alpha=0.5) +
  geom_path(aes(x=as.factor(year),y=conditional_count, group = 1), size=1.25) +
  geom_point(aes(x=as.factor(year),y=conditional_count), size =4) +
  scale_y_continuous(expand = c(0,0), limits = c(0,6)) +
  theme_bw() +
  labs(title= "Big Bend Cr. at mouth", y="# of Days", x="Year") +
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14),
        plot.subtitle = element_text(size = 10),
        plot.title = element_text(size=18, face="bold"),
        legend.position = "none")
```

```
rm(temp)
BigBend_Days20_Plot
```

Big Bend Cr. at mouth



```
# Export plot to file
# ggsave(BigBend_Days20_Plot, file="Figure Plots/Temp Plots/DaysOver20C_BigBend.png", height = 4, width = 6)

## Air/Water Regression Plot

# Prepare air logger data for regression
air <- daily_air_master %>%
  filter(logger_id == "21679162") %>% # N. Umpqua at Steamboat air logger
  mutate(longdate = make_date(year,month,day)) %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(air_mean = mean_temp)

# Prepare individual water logger data for regression
water <- BigBend %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(water_max = max_temp)

# Join air/water data frames for linear regression
airwater_join <- water %>%
  left_join(air, by = "longdate")
```

```

# Create equation formula for regression equation
lm_eqn <- function(df){
  m <- lm(water_max ~ air_mean, df);
  eq <- substitute(italic(y) == a + b %.% italic(x)*", "~~italic(r)^2~"=~r2,
    list(a = format(unname(coef(m)[1]), digits = 3),
        b = format(unname(coef(m)[2]), digits = 3),
        r2 = format(summary(m)$adj.r.squared, digits = 3)))
  as.character(as.expression(eq));
}

# Calculate Pearson correlation coefficient
correlation <- cor(airwater_join$water_max, airwater_join$air_mean, method = 'pearson')
print(paste("Pearson R:", correlation))

## [1] "Pearson R: 0.857961419366988"

# Linear model fit summary
fit <- lm(water_max ~ air_mean, data = airwater_join)
summary(fit)

## 
## Call:
## lm(formula = water_max ~ air_mean, data = airwater_join)
## 
## Residuals:
##       Min     1Q     Median      3Q      Max 
## -2.1525 -0.6612  0.0207  0.6623  1.7880 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 5.37096   0.74242   7.234 1.82e-10 ***
## air_mean    0.56637   0.03657  15.488 < 2e-16 ***
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.9558 on 86 degrees of freedom
## Multiple R-squared:  0.7361, Adjusted R-squared:  0.733 
## F-statistic: 239.9 on 1 and 86 DF,  p-value: < 2.2e-16

# Create regression plot
reg_plot_15170014 <- ggplot(airwater_join, aes(x=air_mean, y = water_max)) +
  geom_point(shape= 21, color='black', fill= 'deepskyblue3', size = 2) +
  geom_smooth(method = "lm", se = TRUE, colour = 'black') +
  theme_light() +
  labs(title="Big Bend Cr. at mouth (Air - N. Umpqua at Steamboat)",
       subtitle = "Max Water Temp ~ Mean Air Temp",
       x="Daily Mean Air temperature (\u00b0C)",
       y="Daily Max Water temperature (\u00b0C)") +
  geom_text(x = 16, y = 19, label = lm_eqn(airwater_join), parse = TRUE) +
  geom_text(x = 16, y = 18, label = paste("Pearson R:", round(correlation,3))) +
  theme(plot.title = element_text(size=12, face="bold"),
        plot.subtitle = element_text(size = 11),

```

```

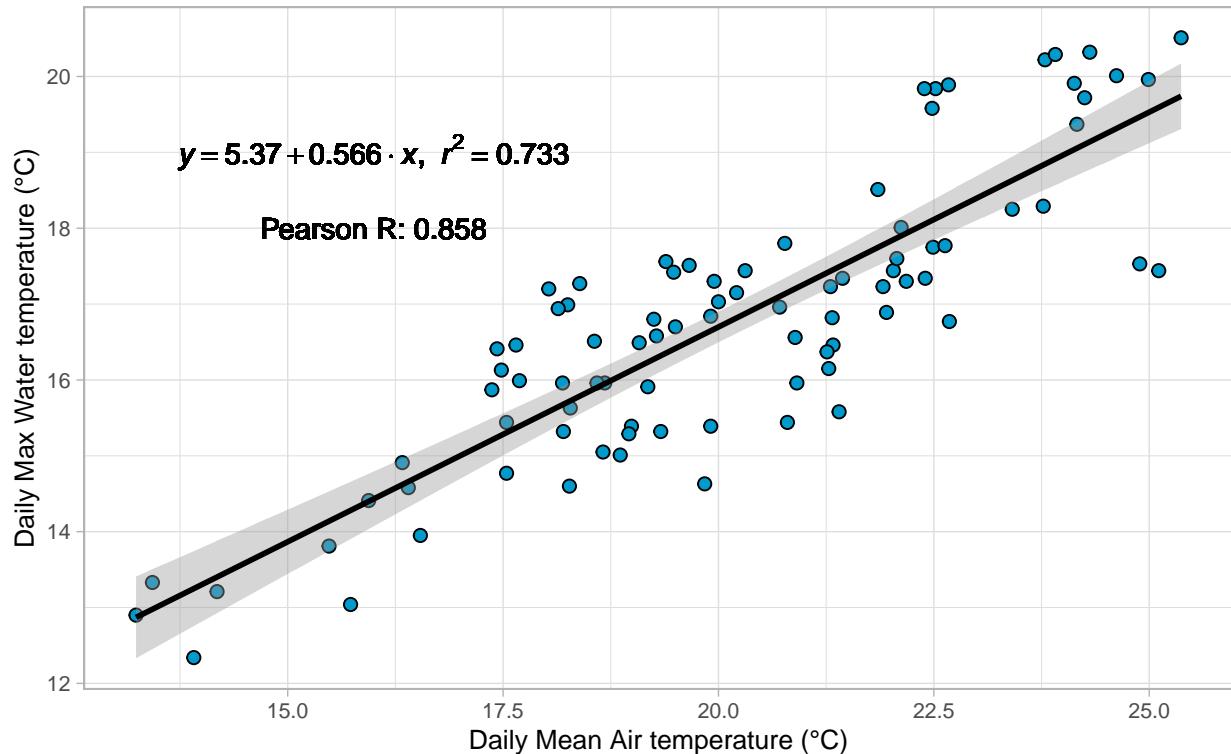
axis.text=element_text(size=8),
axis.title=element_text(size=10))

reg_plot_15170014

```

Big Bend Cr. at mouth (Air – N. Umpqua at Steamboat)

Max Water Temp ~ Mean Air Temp



```

# Export plot to file
# ggsave(reg_plot_15170014, file="Figure Plots/Regression Plots/regression_plot_BigBend.png", height = .)

#-----
## Climate Scenario (3.5C air temp warming)

# Create vector of new air temps under 3.5C warming scenario
new_temps <- data.frame(air_mean = airwater_join$air_mean + 3.5)

# Use lm to predict new water temps using updated air temps
watertemp_pred <- predict(fit, newdata = new_temps)

# Estimate number of days water temp above 20C given future climate scenario
daysover20_future <- data.frame(daysover20 = sum(watertemp_pred > 20))
print(paste("Predicted # days above 20C (+3.5C future climate):", {daysover20_future$daysover20}))

## [1] "Predicted # days above 20C (+3.5C future climate): 21"

```

```
rm(new_temps)
rm(watertemp_pred)
```

Steamboat Below Big Bend Cr.

```
# Logger ID: 21679175 (WSC logger data)

# Read in 2024 WSC data and filter for summer season
Steam_BelowBigBend <- daily_water_master %>%
  filter(logger_id == 21679175 & longdate > '2024-06-19' & longdate <= '2024-09-15')

# 2024 Max Temp
Steam_BelowBigBend_maxt <- max(Steam_BelowBigBend$max_temp)
print(paste("Max Temp C (2024):", Steam_BelowBigBend_maxt))

## [1] "Max Temp C (2024): 22.91"

# 2024 # Days above 20C
Steam_BelowBigBend_Days20 <- sum(Steam_BelowBigBend$max_temp > 20)
print(paste("# Days above 20C:", Steam_BelowBigBend_Days20))

## [1] "# Days above 20C: 17"

### Mean, Min, Max Temp Plot

# Plot data frame
temp <- Steam_BelowBigBend

x<as.Date(temp$longdate)
y_lower = 20
y_upper = Inf

# Create temp plot
plot_21679175 <- ggplot() +
  geom_hline(yintercept = 20, linetype = "dotted", linewidth= 1) +
  geom_ribbon(aes(x=x,ymin = y_lower, ymax = y_upper), fill = "red1", alpha = 0.10) +
  geom_line(data=temp, aes(x=as.Date(longdate),y=mean_temp, color="mean_temp"), linewidth = 1) +
  geom_line(data=temp, aes(x=as.Date(longdate),y=min_temp, color = "min_temp"), linewidth = 0.5) +
  geom_line(data=temp, aes(x=as.Date(longdate), y=max_temp, color= "max_temp"), linewidth = 0.5) +
  scale_colour_manual("",

                     breaks = c("mean_temp", "min_temp", "max_temp"),
                     values = c("black", "blue", "red"),
                     labels = c("Mean Temp.", "Min Temp.", "Max Temp."))
  +
  scale_x_date(date_breaks = "1 month", date_labels = "%b", expand = c(0,0)) +
  scale_y_continuous(limits = c(9,27),breaks = seq(10,26,by= 2)) +
  theme_bw() +
  labs(title= "Steamboat below Big Bend Cr.", y="Temperature (\u00B0C)", x="")
  +
  theme(plot.subtitle = element_text(size = 10),
        plot.title = element_text(size=14, face="bold"))
```

```

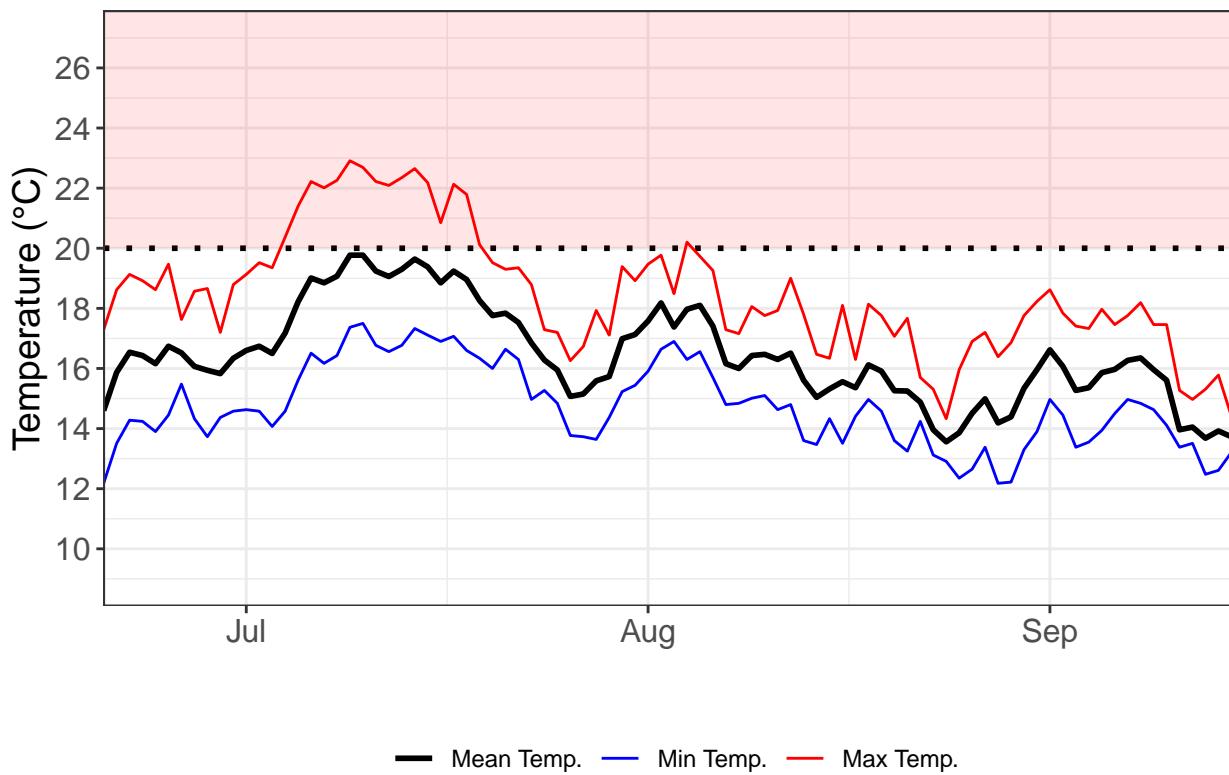
axis.text=element_text(size=12),
axis.title=element_text(size=14),
legend.position = "bottom"

rm(temp)

plot_21679175

```

Steamboat below Big Bend Cr.



```

# Export plot to file
ggsave(plot_21679175, file="Figure Plots/Temp Plots/Temp Plot_SteamboatBelowBigBend.png", height = 5, width = 10)

## Air/Water Regression Plot

# Prepare air logger data for regression
air <- daily_air_master %>%
  filter(logger_id == "21679162") %>% # N. Umpqua at Steamboat air logger
  mutate(longdate = make_date(year,month,day)) %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(air_mean = mean_temp)

# Prepare individual water logger data for regression
water <- daily_water_master %>%
  filter(logger_id == '21679175') %>%
  mutate(longdate = make_date(year,month,day)) %>%

```

```

filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(water_max = max_temp)

# Join air/water data frames for linear regression
airwater_join <- water %>%
  left_join(air, by = "longdate")

# Create equation formula for regression equation

lm_eqn <- function(df){
  m <- lm(water_max ~ air_mean, df);
  eq <- substitute(italic(y) == a + b %.% italic(x)*", "~~italic(r)^2~"="~r2,
    list(a = format(unname(coef(m)[1]), digits = 3),
        b = format(unname(coef(m)[2]), digits = 3),
        r2 = format(summary(m)$adj.r.squared, digits = 3)))
  as.character(as.expression(eq));
}

# Calculate Pearson correlation coefficient
correlation <- cor(airwater_join$water_max, airwater_join$air_mean, method = 'pearson')
print(paste("Pearson R:", correlation))

## [1] "Pearson R: 0.826012131295225"

fit <- lm(water_max ~ air_mean, data = airwater_join)
summary(fit)

##
## Call:
## lm(formula = water_max ~ air_mean, data = airwater_join)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -2.13682 -1.05882  0.06288  0.91008  2.27836 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 6.31835   0.90372  6.991 5.51e-10 ***
## air_mean     0.60494   0.04451 13.590  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.163 on 86 degrees of freedom
## Multiple R-squared:  0.6823, Adjusted R-squared:  0.6786 
## F-statistic: 184.7 on 1 and 86 DF,  p-value: < 2.2e-16

# Create regression plot

reg_plot_21679175 <- ggplot(airwater_join, aes(x=air_mean, y = water_max)) +
  geom_point(shape= 21, color='black', fill= 'deepskyblue3', size = 2) +
  geom_smooth(method = "lm", se = TRUE, colour = 'black') +
  theme_light() +

```

```

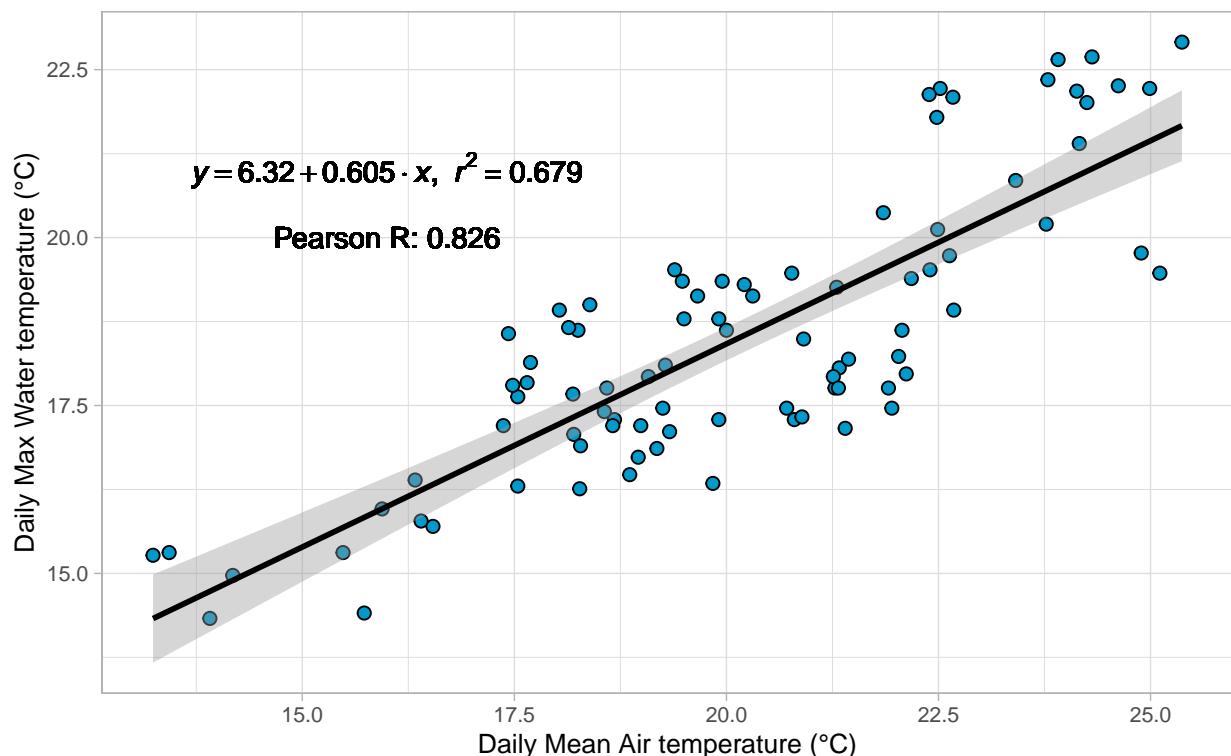
  labs(title="Steamboat below Big Bend Cr. (Air - N. Umpqua at Steamboat)",
       subtitle = "Max Water Temp ~ Mean Air Temp",
       x="Daily Mean Air temperature (\u00b0C)",
       y="Daily Max Water temperature (\u00b0C)") +
  geom_text(x = 16, y = 21, label = lm_eqn(airwater_join), parse = TRUE) +
  geom_text(x = 16, y = 20, label = paste("Pearson R:", round(correlation,3))) +
  theme(plot.title = element_text(size=12, face="bold"),
        plot.subtitle = element_text(size = 11),
        axis.text=element_text(size=8),
        axis.title=element_text(size=10))

reg_plot_21679175

```

Steamboat below Big Bend Cr. (Air – N. Umpqua at Steamboat)

Max Water Temp ~ Mean Air Temp



```

# Export plot to file
# ggsave(reg_plot_21679175, file="Figure Plots/Regression Plots/regression_plot_SteamboatBelowBigBend.p
#-----

## Climate Scenario (3.5C air temp warming)

# Create vector of new air temps under 3.5C warming scenario
new_temps <- data.frame(air_mean = airwater_join$air_mean + 3.5)

# Use lm to predict new water temps using updated air temps
watertemp_pred <- predict(fit, newdata = new_temps)

```

```

# Estimate number of days water temp above 20C given future climate scenario
daysover20_future <- data_frame(daysover20 = sum(watertemp_pred > 20))
print(paste("Predicted # days above 20C (+3.5C future climate):", {daysover20_future$daysover20}))

## [1] "Predicted # days above 20C (+3.5C future climate): 55"

rm(new_temps)
rm(watertemp_pred)

```

Steamboat Above Canton Cr.

```

# Logger ID: 15170003 (USGS hydro station data)

## Read in Steamboat Above data, clean, and summarize daily mean, min, max temp statistics

list_of_files <- list.files(path = "USB Files/SteamAboveCanton",
                           recursive = TRUE,
                           pattern = "\\.csv$",
                           full.names = TRUE)

Steamboat_AboveCant <- readr::read_csv(list_of_files, id = "file_name") %>%
  clean_names() %>%
  mutate(site = "Steamboat above Canton Cr.") %>%
  mutate(temp_c = (temp - 32) * (5/9)) %>% #temp in F, need to convert to C
  mutate(logger_id = "15170003") %>%
  mutate(longdate = as.Date(date,format = "%m/%d/%Y")) %>%
  separate(longdate,
           into = c('year', 'month', 'day'),
           sep= '-',
           remove = FALSE) %>%
  mutate(month_day = format(longdate, "%m-%d")) %>%
  group_by(longdate, month_day, year, logger_id) %>%
  summarise(mean_temp = round(mean(temp_c), digits=2),
            min_temp = round(min(temp_c), digits=2),
            max_temp = round(max(temp_c), digits=2))

## Calculate other temperature statistics

# Max Temp (2024)
SteamboatCant_maxt_2024 <- Steamboat_AboveCant %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  group_by(logger_id) %>%
  summarise(maxt = max(max_temp))

print(paste("Max Temp C (2024):", SteamboatCant_maxt_2024$maxt))

## [1] "Max Temp C (2024): 26.21"

```

```

# Days >20C (2024)
SteamboatCant_Days20_2024 <- Steamboat_AboveCant %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  filter(max_temp > 20) %>%
  group_by(logger_id) %>%
  summarise(days_over20 = n())

print(paste("# Days above 20C (2024):", SteamboatCant_Days20_2024$days_over20))

```

[1] "# Days above 20C (2024): 71"

```

# Max Temp (10-yr Avg.)
SteamboatCant_maxt_10yrAvg <- Steamboat_AboveCant %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  group_by(logger_id,year) %>%
  summarise(maxt = max(max_temp)) %>%
  summarise(maxt_avg = mean(maxt))

print(paste("Max Temp C (10-yr Avg.):", SteamboatCant_maxt_10yrAvg$maxt_avg))

```

[1] "Max Temp C (10-yr Avg.): 26.234"

```

# Days >20C (10-yr Avg.)
SteamboatCant_Days20_10yrAvg <- Steamboat_AboveCant %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  filter(max_temp > 20) %>%
  group_by(logger_id,year) %>%
  summarise(days_over20 = n()) %>%
  summarise(days_over20_avg = sum(days_over20) / 10)

print(paste("# Days above 20C (10-yr Avg.):", SteamboatCant_Days20_10yrAvg$days_over20_avg))

```

[1] "# Days above 20C (10-yr Avg.): 64.2"

```

# Max Temp (Drought)
SteamboatCant_maxt_drought <- Steamboat_AboveCant %>%
  filter(year == '2021' | year == '2022') %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  group_by(logger_id) %>%
  summarise(maxt = max(max_temp))

print(paste("Max Temp C (Drought yrs):", SteamboatCant_maxt_drought$maxt))

```

[1] "Max Temp C (Drought yrs): 28.87"

```

# Days >20C (Drought)
SteamboatCant_Days20_drought <- Steamboat_AboveCant %>%
  filter(year == '2021' | year == '2022') %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  filter(max_temp > 20) %>%

```

```

group_by(logger_id,year) %>%
  summarise(days_over20 = n()) %>%
  summarise(days_over20_avg = sum(days_over20) / 2)

print(paste("# Days above 20C (Drought yrs):", SteamboatCant_Days20_drought$days_over20_avg))

## [1] "# Days above 20C (Drought yrs): 60.5"

## Mean, Min, Max Temp Plot

# Data frame for plot
temp <- Steamboat_AboveCant %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15')

x=as.Date(temp$longdate)
y_lower = 20
y_upper = Inf

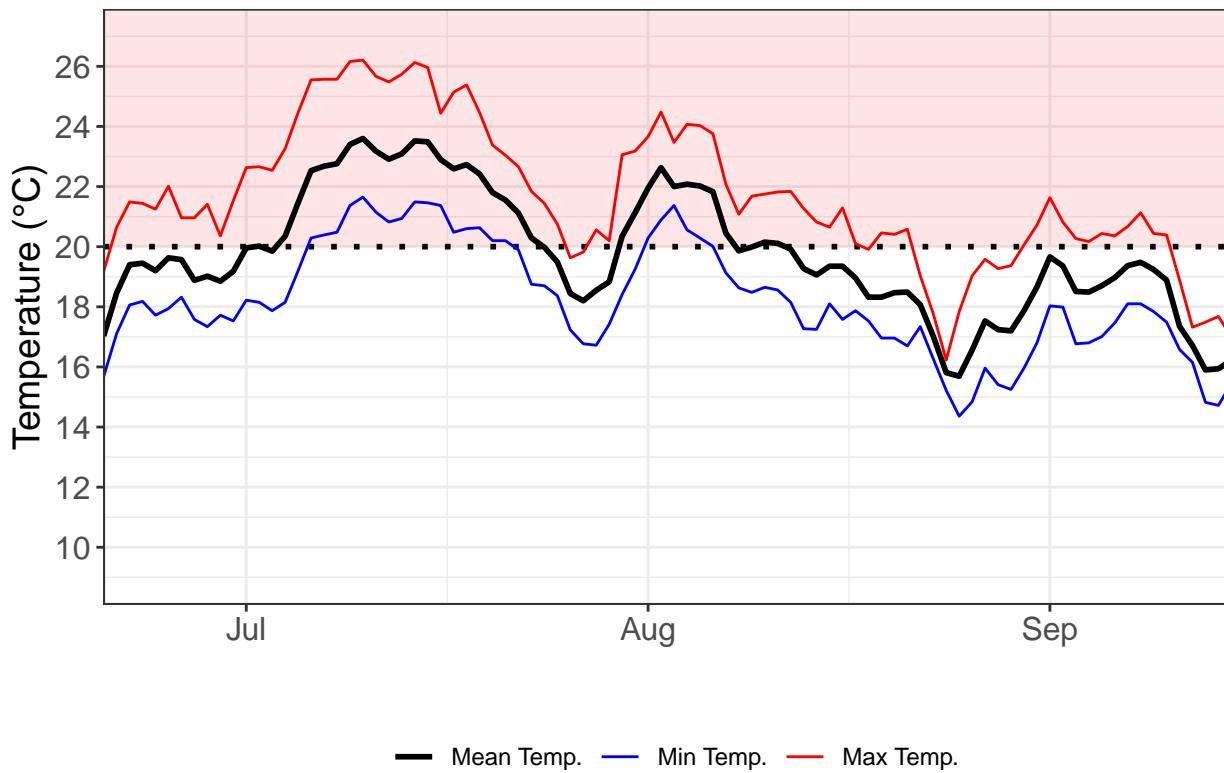
# Create temp plot
plot_15170003 <- ggplot() +
  geom_hline(yintercept = 20, linetype = "dotted", linewidth= 1) +
  geom_ribbon(aes(x=x,ymin = y_lower, ymax = y_upper), fill = "red1", alpha = 0.10) +
  geom_line(data=temp, aes(x=longdate,y=mean_temp, color="mean_temp"), linewidth =1) +
  geom_line(data=temp, aes(x=longdate,y=min_temp, color = "min_temp"), linewidth =0.5) +
  geom_line(data=temp, aes(x=longdate, y=max_temp), color= "max_temp"), linewidth =0.5) +
  scale_colour_manual("", 
    breaks = c("mean_temp", "min_temp", "max_temp"),
    values = c("black", "blue", "red"),
    labels = c("Mean Temp.", "Min Temp.", "Max Temp."))
  scale_x_date(date_breaks = "1 month", date_labels = "%b", expand = c(0,0)) +
  scale_y_continuous(limits = c(9,27),breaks = seq(10,26,by=2)) +
  theme_bw() +
  labs(title= "Steamboat above Canton Cr.", y="Temperature (\u00B0C)", x="")
  theme(plot.subtitle = element_text(size = 10),
  plot.title = element_text(size=14, face="bold"),
  axis.text=element_text(size=12),
  axis.title=element_text(size=14),
  legend.position = "bottom")

rm(temp)

plot_15170003

```

Steamboat above Canton Cr.



```
# Export plot to file
ggsave(plot_15170003, file="Figure Plots/Temp Plots/Temp Plot_SteamboatAboveCant.png", height = 5, width = 10)

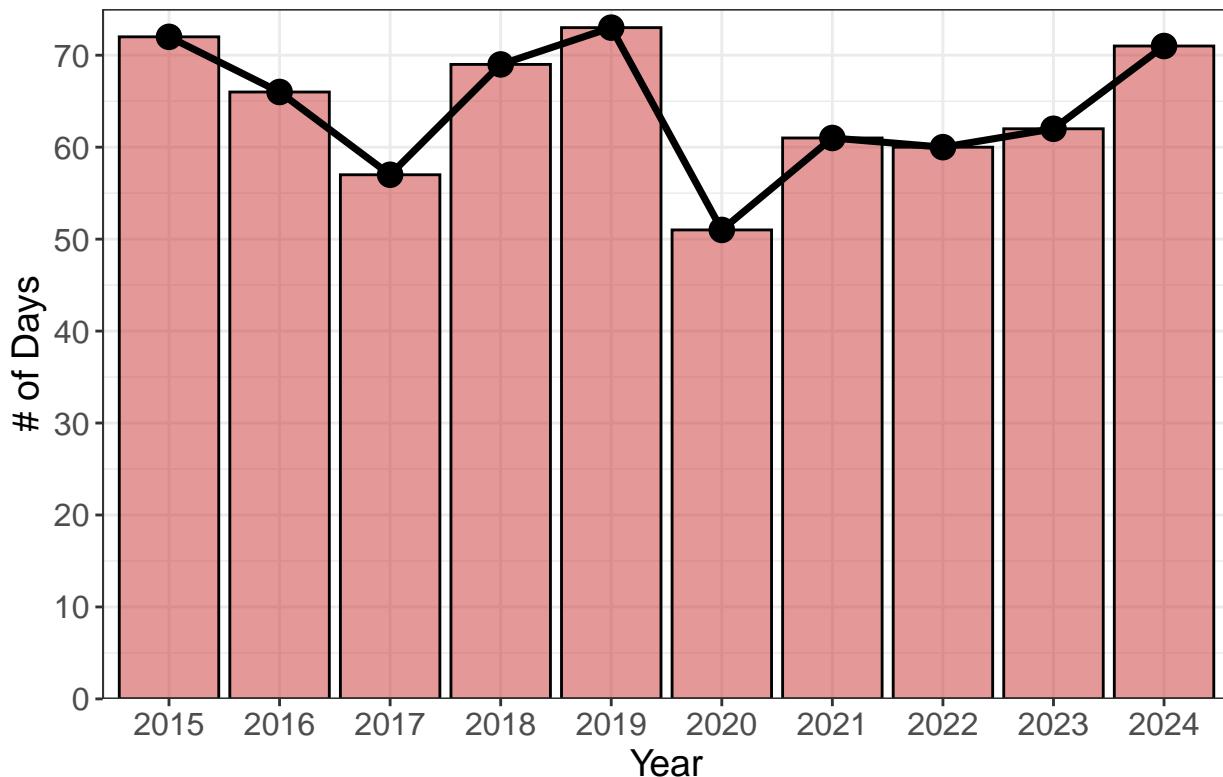
#-----#
## Number of days above 20C Plot

temp <- Steamboat_AboveCant %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  group_by(year) %>%
  summarize(conditional_count = sum(max_temp > 20))

SteamboatCant_Days20_Plot <- ggplot(data=temp) +
  geom_col(aes(x=as.factor(year),y=conditional_count), fill="brown3", color="black",alpha=0.5) +
  geom_path(aes(x=as.factor(year),y=conditional_count, group = 1), size=1.25) +
  geom_point(aes(x=as.factor(year),y=conditional_count), size =4) +
  scale_y_continuous(expand = c(0,0), limits = c(0,75), breaks = seq(0,70,by=10)) +
  theme_bw() +
  theme(legend.position = "none") +
  labs(title= "Steamboat above Canton Cr.", y="# of Days", x="Year") +
  theme(plot.subtitle = element_text(size = 10),
        plot.title = element_text(size=18, face="bold")) +
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14))

SteamboatCant_Days20_Plot
```

Steamboat above Canton Cr.



```

# Export plot to file
ggsave(SteamboatCant_Days20_Plot, file="Figure Plots/Temp Plots/DaysOver20C_SteamboatAboveCant.png", he

## Air/Water Regression Plot

# Prepare air logger data for regression
air <- daily_air_master %>%
  filter(logger_id == "21679162") %>% # N. Umpqua at Steamboat air logger
  mutate(longdate = make_date(year,month,day)) %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(air_mean = mean_temp)

# Prepare individual water logger data for regression
water <- Steamboat_AboveCant %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(water_max = max_temp)

# Join air/water data frames for linear regression
airwater_join <- water %>%
  left_join(air, by = "longdate")

# Create equation formula for regression equation

lm_eqn <- function(df){
  m <- lm(water_max ~ air_mean, df);
  eq <- substitute(italic(y) == a + b %.% italic(x)^"2" = "r2",

```

```

        list(a = format(unname(coef(m)[1]), digits = 3),
              b = format(unname(coef(m)[2]), digits = 3),
              r2 = format(summary(m)$adj.r.squared, digits = 3)))
    as.character(as.expression(eq));
}

# Calculate Pearson correlation coefficient
correlation <- cor(airwater_join$water_max, airwater_join$air_mean, method = 'pearson')
print(paste("Pearson R:",correlation))

## [1] "Pearson R: 0.86049344691036"

fit <- lm(water_max ~ air_mean, data = airwater_join)
summary(fit)

##
## Call:
## lm(formula = water_max ~ air_mean, data = airwater_join)
##
## Residuals:
##       Min     1Q   Median     3Q    Max 
## -2.68809 -0.81058 -0.01364  1.07469  2.25092 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 7.03627   0.94293   7.462 6.39e-11 ***
## air_mean    0.72748   0.04644  15.663 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 

## 
## Residual standard error: 1.214 on 86 degrees of freedom
## Multiple R-squared:  0.7404, Adjusted R-squared:  0.7374 
## F-statistic: 245.3 on 1 and 86 DF,  p-value: < 2.2e-16

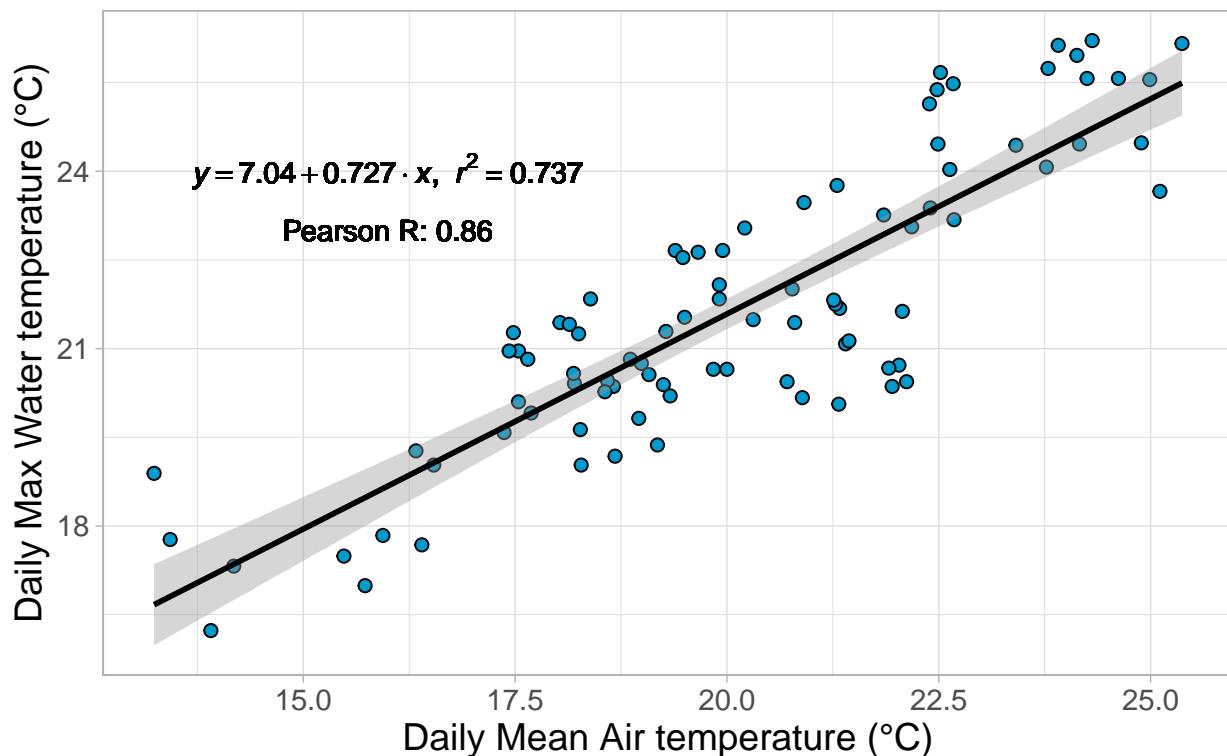
# Create regression plot
reg_plot_15170003 <- ggplot(airwater_join, aes(x=air_mean, y = water_max)) +
  geom_point(shape= 21, color='black', fill= 'deepskyblue3', size = 2) +
  geom_smooth(method = "lm", se = TRUE, colour = 'black') +
  theme_light() +
  labs(title="Steamboat above Canton Cr. (Air - N. Umpqua at Steamboat)",
       subtitle = "Max Water Temp ~ Mean Air Temp",
       x="Daily Mean Air temperature (\u00b0C)",
       y="Daily Max Water temperature (\u00b0C)") +
  geom_text(x = 16, y = 24, label = lm_eqn(airwater_join), parse = TRUE) +
  geom_text(x = 16, y = 23, label = paste("Pearson R:", round(correlation,3))) +
  theme(plot.title = element_text(size=12, face="bold"),
        plot.subtitle = element_text(size = 11),
        axis.text=element_text(size=11),
        axis.title=element_text(size=14))

reg_plot_15170003

```

Steamboat above Canton Cr. (Air – N. Umpqua at Steamboat)

Max Water Temp ~ Mean Air Temp



```
# Export plot to file
# ggsave(reg_plot_15170003, file="Figure Plots/Regression Plots/regression_plot_SteamboatAboveCanton.pdf")
#-----#
## Climate Scenario (3.5C air temp warming)

# Create vector of new air temps under 3.5C warming scenario
new_temps <- data.frame(air_mean = airwater_join$air_mean + 3.5)

# Use lm to predict new water temps using updated air temps
watertemp_pred <- predict(fit, newdata = new_temps)

# Estimate number of days water temp above 20C given future climate scenario
daysover20_future <- data.frame(daysover20 = sum(watertemp_pred > 20))
print(paste("Predicted # days above 20C (+3.5C future climate):", daysover20_future$daysover20))

## [1] "Predicted # days above 20C (+3.5C future climate): 84"

rm(new_temps)
rm(watertemp_pred)
```

Upper Canton Cr.

```
# Logger ID: 21679177 (WSC logger data)

# Read in 2024 WSC data and filter for summer season
UpperCanton <- daily_water_master %>%
  filter(logger_id == 21679177 & longdate > '2024-06-19' & longdate <= '2024-09-15')

# 2024 Max Temp
UpperCanton_maxt <- max(UpperCanton$max_temp)
print(paste("Max Temp C (2024):", UpperCanton_maxt))

## [1] "Max Temp C (2024): 22.52"

# 2024 # Days above 20C
UpperCanton_Days20 <- sum(UpperCanton$max_temp > 20)
print(paste("# Days above 20C (2024):", UpperCanton_Days20))

## [1] "# Days above 20C (2024): 23"

### Mean, Min, Max Temp Plot

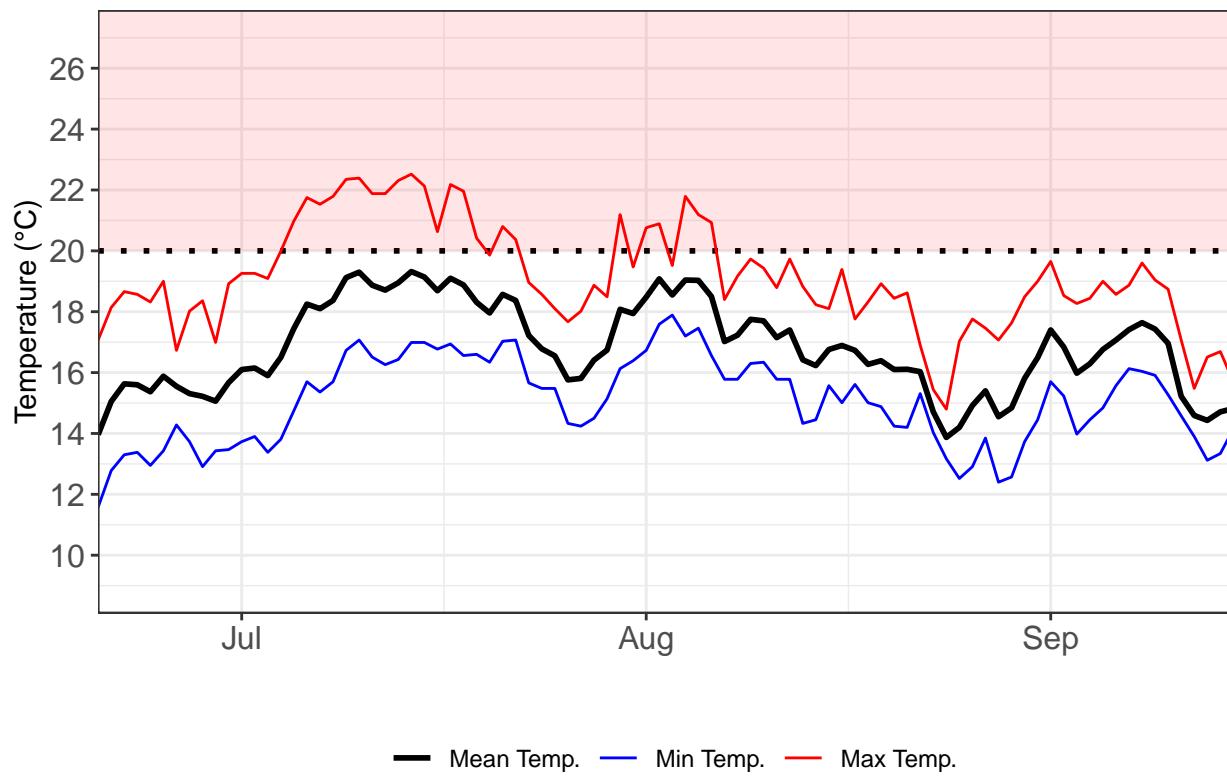
# Plot data frame
temp <- UpperCanton

x=as.Date(temp$longdate)
y_lower = 20
y_upper = Inf

# Create temp plot
plot_21679177 <- ggplot() +
  geom_hline(yintercept = 20, linetype = "dotted", linewidth= 1) +
  geom_ribbon(aes(x=x,ymin = y_lower, ymax = y_upper), fill = "red1", alpha = 0.10) +
  geom_line(data=temp, aes(x=as.Date(longdate),y=mean_temp, color="mean_temp"), linewidth = 1) +
  geom_line(data=temp, aes(x=as.Date(longdate),y=min_temp, color = "min_temp"), linewidth = 0.5) +
  geom_line(data=temp, aes(x=as.Date(longdate), y=max_temp, color= "max_temp"), linewidth = 0.5) +
  scale_colour_manual("", 
                      breaks = c("mean_temp", "min_temp", "max_temp"),
                      values = c("black", "blue", "red"),
                      labels = c("Mean Temp.", "Min Temp.", "Max Temp."))
  scale_x_date(date_breaks = "1 month", date_labels = "%b", expand = c(0,0)) +
  scale_y_continuous(limits = c(9,27),breaks = seq(10,26,by= 2)) +
  theme_bw() +
  labs(title= "Upper Canton Cr.", y="Temperature (\u00B0C)", x="")
  theme(plot.subtitle = element_text(size = 10),
        plot.title = element_text(size=14, face="bold"),
        axis.text=element_text(size=12),
        xis.title=element_text(size=14),
        legend.position = "bottom")
```

```
rm(temp)
plot_21679177
```

Upper Canton Cr.



```
# Export plot to file
# ggsave(plot_21679177, file="Figure Plots/Temp Plots/Temp Plot_UpperCanton.png", height = 5, width = 8)

## Air/Water Regression Plot

# Prepare air logger data for regression
air <- daily_air_master %>%
  filter(logger_id == "21679162") %>% # N. Umpqua at Steamboat air logger
  mutate(longdate = make_date(year,month,day)) %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(air_mean = mean_temp)

# Prepare individual water logger data for regression
water <- daily_water_master %>%
  filter(logger_id == '21679177') %>%
  mutate(longdate = make_date(year,month,day)) %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(water_max = max_temp)

# Join air/water data frames for linear regression
airwater_join <- water %>%
```

```

left_join(air, by = "longdate")

# Create equation formula for regression equation

lm_eqn <- function(df){
  m <- lm(water_max ~ air_mean, df);
  eq <- substitute(italic(y) == a + b %.% italic(x)*", "~~italic(r)^2~~="~r2,
    list(a = format(unname(coef(m)[1]), digits = 3),
        b = format(unname(coef(m)[2]), digits = 3),
        r2 = format(summary(m)$adj.r.squared, digits = 3)))
  as.character(as.expression(eq));
}

# Calculate Pearson correlation coefficient
correlation <- cor(airwater_join$water_max, airwater_join$air_mean, method = 'pearson')
print(paste("Pearson R:", correlation))

## [1] "Pearson R: 0.883615218955373"

# Summary of linear model fit
fit <- lm(water_max ~ air_mean, data = airwater_join)
summary(fit)

##
## Call:
## lm(formula = water_max ~ air_mean, data = airwater_join)
##
## Residuals:
##       Min     1Q   Median     3Q    Max 
## -1.56767 -0.55731 -0.04173  0.55559  1.79778 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 7.93773   0.64476  12.31   <2e-16 ***
## air_mean    0.55581   0.03176  17.50   <2e-16 ***
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
## 
## Residual standard error: 0.8301 on 86 degrees of freedom
## Multiple R-squared:  0.7808, Adjusted R-squared:  0.7782 
## F-statistic: 306.3 on 1 and 86 DF,  p-value: < 2.2e-16

# Create regression plot
reg_plot_21679177 <- ggplot(airwater_join, aes(x=air_mean, y = water_max)) +
  geom_point(shape= 21, color='black', fill= 'deepskyblue3', size = 2) +
  geom_smooth(method = "lm", se = TRUE, colour = 'black') +
  theme_light() +
  labs(title="Upper Canton Cr. (Air - N. Umpqua at Steamboat)",
       subtitle = "Max Water Temp ~ Mean Air Temp",
       x="Daily Mean Air temperature (\u00b0C)",
       y="Daily Max Water temperature (\u00b0C)") +
  geom_text(x = 16, y = 21, label = lm_eqn(airwater_join), parse = TRUE) +

```

```

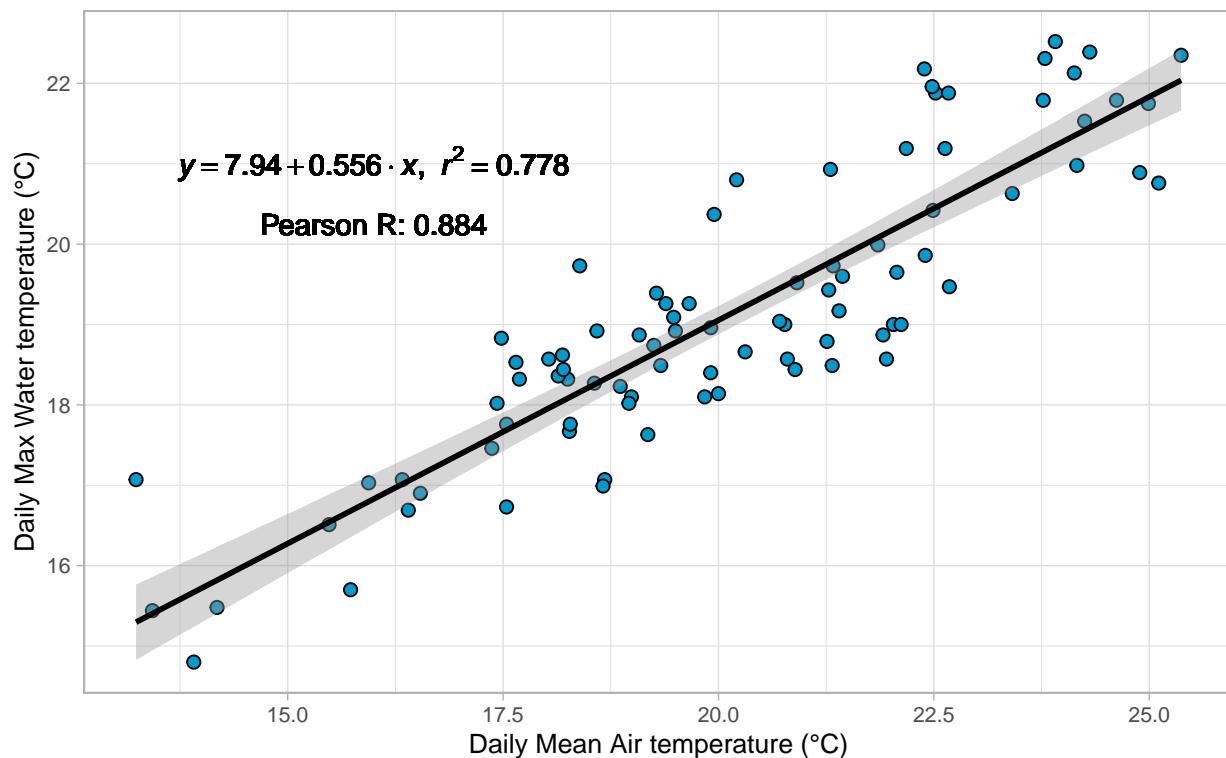
geom_text(x = 16, y = 20.25, label = paste("Pearson R:", round(correlation,3))) +
  theme(plot.title = element_text(size=12, face="bold"),
        plot.subtitle = element_text(size = 11),
        axis.text=element_text(size=8),
        axis.title=element_text(size=10))

reg_plot_21679177

```

Upper Canton Cr. (Air – N. Umpqua at Steamboat)

Max Water Temp ~ Mean Air Temp



```

# Export plot to file
# ggsave(reg_plot_21679177, file="Figure Plots/Regression Plots/regression_plot_UpperCanton.png", height=5, width=5)

## Climate Scenario (3.5C air temp warming)

# Create vector of new air temps under 3.5C warming scenario
new_temps <- data.frame(air_mean = airwater_join$air_mean + 3.5)

# Use lm to predict new water temps using updated air temps
watertemp_pred <- predict(fit, newdata = new_temps)

# Estimate number of days water temp above 20C given future climate scenario
daysover20_future <- data.frame(daysover20 = sum(watertemp_pred > 20))
print(paste("Predicted # days above 20C (+3.5C future climate):", {daysover20_future$daysover20}))

```

```
## [1] "Predicted # days above 20C (+3.5C future climate): 67"
```

```
rm(new_temps)
rm(watertemp_pred)
```

N. Umpqua above Steamboat

```
# Logger ID: 21433141 (WSC logger data)

# Read in 2024 WSC data and filter for summer season
NUmpqua_AboveSteam <- daily_water_master %>%
  filter(logger_id == 21433141 & longdate > '2024-06-19' & longdate <= '2024-09-15')

# 2024 Max Temp
NUmpqua_AboveSteam_maxt <- max(NUmpqua_AboveSteam$max_temp)
print(paste("Max Temp C (2024):", NUmpqua_AboveSteam_maxt))
```

```
## [1] "Max Temp C (2024): 19.39"
```

```
# 2024 # Days above 20C
NUmpqua_AboveSteam_Days20 <- sum(NUmpqua_AboveSteam$max_temp > 20)
print(paste("# Days above 20C:", NUmpqua_AboveSteam_Days20))
```

```
## [1] "# Days above 20C: 0"
```

```
## Mean, Min, Max Temp Plot

# Plot data frame
temp <- NUmpqua_AboveSteam

x(as.Date(temp$longdate)
y_lower = 20
y_upper = Inf

# Create temp plot
plot_21433141 <- ggplot() +
  geom_hline(yintercept = 20, linetype = "dotted", linewidth = 1) +
  geom_ribbon(aes(x=x,ymin = y_lower, ymax = y_upper), fill = "red1", alpha = 0.10) +
  geom_line(data=temp, aes(x=as.Date(longdate),y=mean_temp, color="mean_temp"), linewidth = 1) +
  geom_line(data=temp, aes(x=as.Date(longdate),y=min_temp, color = "min_temp"), linewidth = 0.5) +
  geom_line(data=temp, aes(x=as.Date(longdate), y=max_temp, color= "max_temp"), linewidth = 0.5) +
  scale_colour_manual("", 
    breaks = c("mean_temp", "min_temp", "max_temp"),
    values = c("black", "blue", "red"),
    labels = c("Mean Temp.", "Min Temp.", "Max Temp."))
  scale_x_date(date_breaks = "1 month", date_labels = "%b", expand = c(0,0)) +
  scale_y_continuous(limits = c(9,27),breaks = seq(10,26,by=2)) +
  theme_bw()
```

```

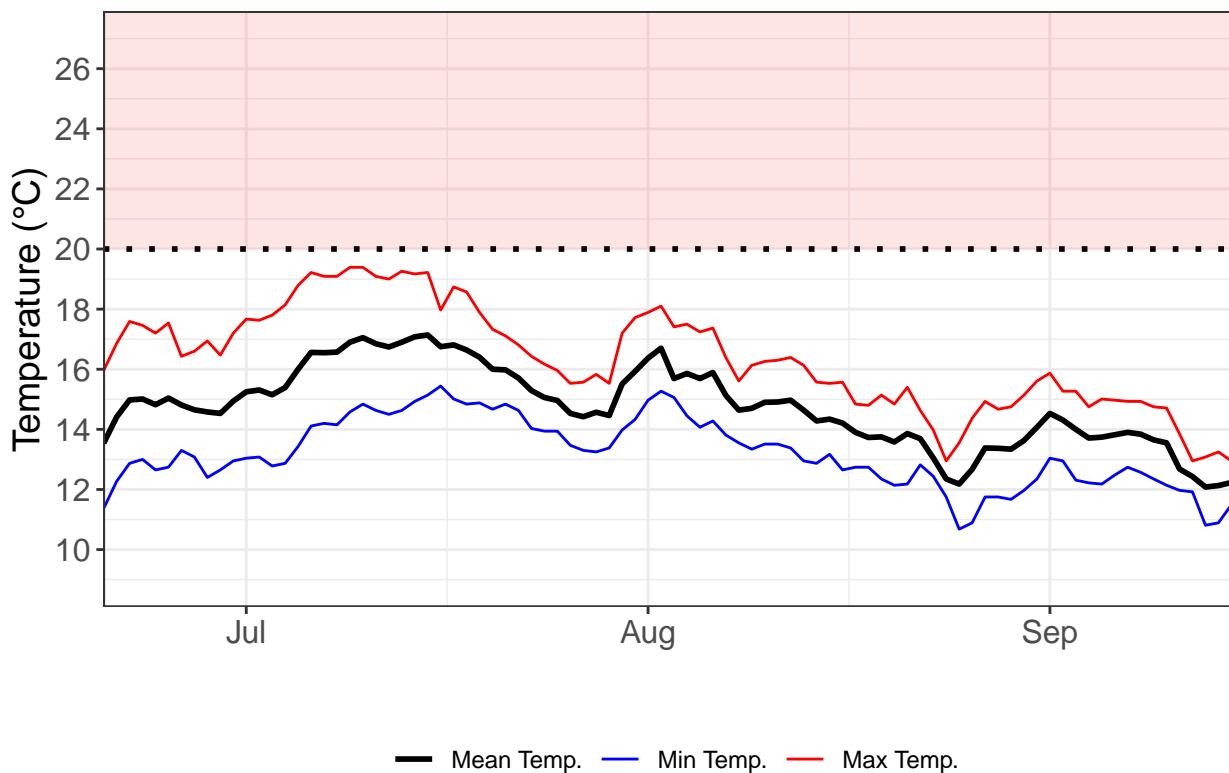
  labs(title= "N. Umpqua above Steamboat Cr.", y="Temperature (\u00b0C)", x="") +
  theme(plot.subtitle = element_text(size = 10),
        plot.title = element_text(size=14, face="bold"),
        axis.text=element_text(size=12),
        axis.title=element_text(size=14),
        legend.position = "bottom")

rm(temp)

plot_21433141

```

N. Umpqua above Steamboat Cr.



```

# Export plot to file
# ggsave(plot_21433141, file="Figure Plots/Temp Plots/Temp Plot_NUmpqua_AboveSteam.png", height = 5, width = 10)

## Air/Water Regression Plot

# Prepare air logger data for regression
air <- daily_air_master %>%
  filter(logger_id == "21679162") %>% # N. Umpqua at Steamboat air logger
  mutate(longdate = make_date(year,month,day)) %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(air_mean = mean_temp)

# Prepare individual water logger data for regression
water <- daily_water_master %>%

```

```

filter(logger_id == '21433141') %>%
  mutate(longdate = make_date(year,month,day)) %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(water_max = max_temp)

# Join air/water data frames for linear regression
airwater_join <- water %>%
  left_join(air, by = "longdate")

# Create equation formula for regression equation

lm_eqn <- function(df){
  m <- lm(water_max ~ air_mean, df);
  eq <- substitute(italic(y) == a + b %.% italic(x)*",",~~italic(r)^2~"=~r2,
    list(a = format(unname(coef(m)[1]), digits = 3),
        b = format(unname(coef(m)[2]), digits = 3),
        r2 = format(summary(m)$adj.r.squared, digits = 3)))
  as.character(as.expression(eq));
}

# Calculate Pearson correlation coefficient
correlation <- cor(airwater_join$water_max, airwater_join$air_mean, method = 'pearson')
print(paste("Pearson R:",correlation))

```

```
## [1] "Pearson R: 0.78598791083731"
```

```

# Summary linear model fit
fit <- lm(water_max ~ air_mean, data = airwater_join)
summary(fit)

```

```

##
## Call:
## lm(formula = water_max ~ air_mean, data = airwater_join)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.31992 -0.61825  0.03484  0.83068  2.09444
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6.70608   0.82706  8.108 3.19e-12 ***
## air_mean     0.48028   0.04074 11.790  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.065 on 86 degrees of freedom
## Multiple R-squared:  0.6178, Adjusted R-squared:  0.6133
## F-statistic: 139 on 1 and 86 DF,  p-value: < 2.2e-16

```

```

# Create regression plot
reg_plot_21433141 <- ggplot(airwater_join, aes(x=air_mean, y = water_max)) +
  geom_point(shape= 21, color='black', fill= 'deepskyblue3', size = 2) +

```

```

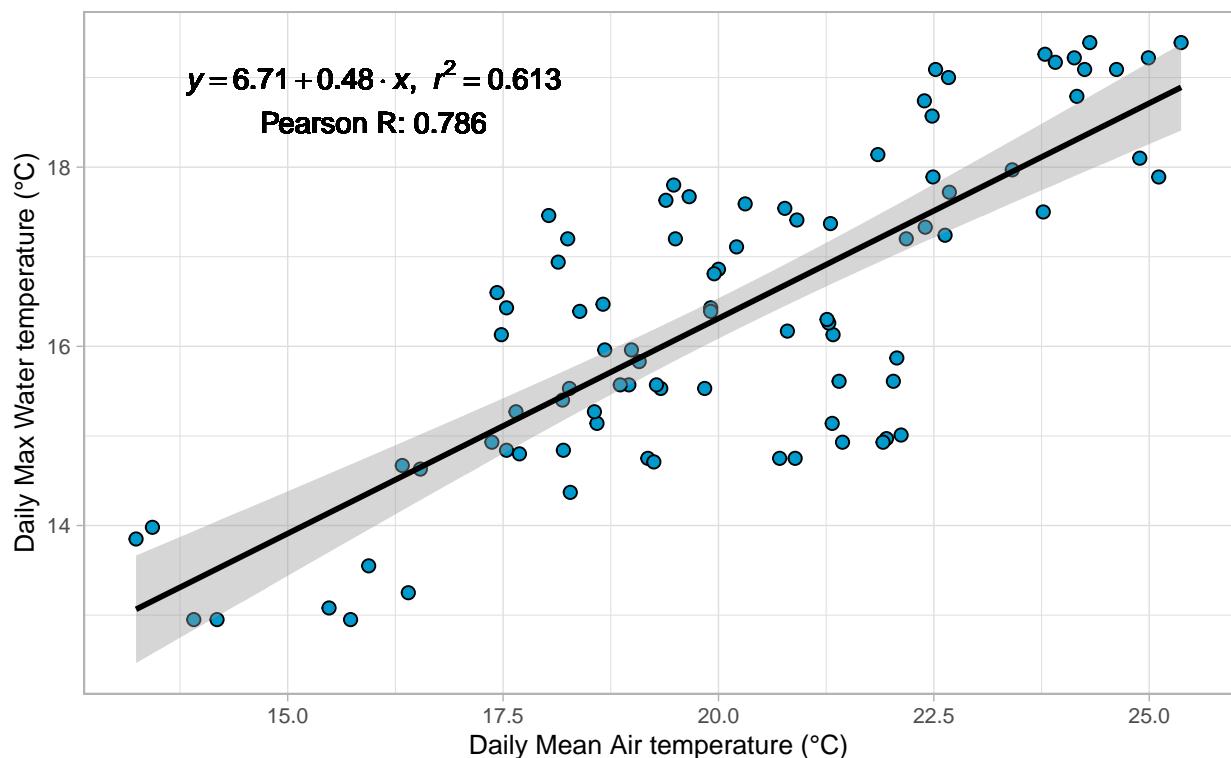
geom_smooth(method = "lm", se = TRUE, colour = 'black') +
theme_light() +
labs(title="North Umpqua above Steamboat Cr. (Air - N. Umpqua at Steamboat)",
  subtitle = "Max Water Temp ~ Mean Air Temp",
  x="Daily Mean Air temperature (\u00B0C)",
  y="Daily Max Water temperature (\u00B0C)") +
geom_text(x = 16, y = 19, label = lm_eqn(airwater_join), parse = TRUE) +
geom_text(x = 16, y = 18.5, label = paste("Pearson R:", round(correlation,3))) +
theme(plot.title = element_text(size=12, face="bold"),
  plot.subtitle = element_text(size = 11),
  axis.text=element_text(size=8),
  axis.title=element_text(size=10))

reg_plot_21433141

```

North Umpqua above Steamboat Cr. (Air – N. Umpqua at Steamboat)

Max Water Temp ~ Mean Air Temp



```

#Export plot to file
#ggsave(reg_plot_21433141, file="Figure Plots/Regression Plots/regression_plot_NUmpqua_AboveSteam.png",
#-----#
## Climate Scenario (3.5C air temp warming)

# Create vector of new air temps under 3.5C warming scenario
new_temps <- data.frame(air_mean = airwater_join$air_mean + 3.5)

```

```

# Use lm to predict new water temps using updated air temps
watertemp_pred <- predict(fit, newdata = new_temps)

# Estimate number of days water temp above 20C given future climate scenario
daysover20_future <- data_frame(daysover20 = sum(watertemp_pred > 20))
print(paste("Predicted # days above 20C (+3.5C future climate):", daysover20_future$daysover20))

## [1] "Predicted # days above 20C (+3.5C future climate): 7"

rm(new_temps)
rm(watertemp_pred)

```

Fish Cr. at mouth

```

# Logger ID: 15110003 (USGS hydro station data)

## Read in Big Bend data, clean, and summarize daily mean, min, max temp statistics

list_of_files <- list.files(path = "USB Files/Fish",
                            recursive = TRUE,
                            pattern = "\\.csv$",
                            full.names = TRUE)

FishCr <- readr::read_csv(list_of_files, id = "file_name") %>%
  clean_names() %>%
  mutate(site = "Fish Cr. at mouth") %>%
  mutate(temp_c = (temp - 32) * (5/9)) %>% #temp in F, need to convert to C
  mutate(logger_id = "15110003") %>%
  mutate(longdate = as.Date(date, format = "%m/%d/%Y")) %>%
  separate(longdate,
           into = c('year', 'month', 'day'),
           sep= '-',
           remove = FALSE) %>%
  mutate(month_day = format(longdate, "%m-%d")) %>%
  group_by(longdate, month_day, year, logger_id) %>%
  summarise(mean_temp = round(mean(temp_c), digits=2),
            min_temp = round(min(temp_c), digits=2),
            max_temp = round(max(temp_c), digits=2))

## Calculate other temperature statistics

# Max Temp (2024)
FishCr_maxt_2024 <- FishCr %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  group_by(logger_id) %>%
  summarise(maxt = max(max_temp))

print(paste("Max Temp C (2024):", FishCr_maxt_2024$maxt))

## [1] "Max Temp C (2024): 18.24"

```

```

# Days >20C (2024)
FishCr_Days20_2024 <- FishCr %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  filter(max_temp > 20) %>%
  group_by(logger_id) %>%
  summarise(days_over20 = n())

print(paste("# Days above 20C (2024):", FishCr_Days20_2024$days_over20))

```

```
## [1] "# Days above 20C (2024): "
```

```

# Max Temp (10-yr Avg.)
FishCr_maxt_10yrAvg <- FishCr %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  group_by(logger_id,year) %>%
  summarise(maxt = max(max_temp)) %>%
  summarise(maxt_avg = mean(maxt))

```

```
print(paste("Max Temp C (10-yr Avg.):", FishCr_maxt_10yrAvg$maxt_avg))
```

```
## [1] "Max Temp C (10-yr Avg.): 19.965"
```

```

# Days >20C (10-yr Avg.)
FishCr_Days20_10yrAvg <- FishCr %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  filter(max_temp > 20) %>%
  group_by(year) %>%
  summarise(days_over20 = n()) %>%
  summarise(days_over20_avg = sum(days_over20) / 10)

```

```
print(paste("# Days above 20C (10-yr Avg.):", FishCr_Days20_10yrAvg$days_over20_avg))
```

```
## [1] "# Days above 20C (10-yr Avg.): 1.8"
```

```

# Max Temp (Drought)
FishCr_maxt_drought <- FishCr %>%
  filter(year == '2021' | year == '2022') %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  group_by(logger_id) %>%
  summarise(maxt = max(max_temp))

```

```
print(paste("Max Temp C (Drought yrs):", FishCr_maxt_drought$maxt))
```

```
## [1] "Max Temp C (Drought yrs): 20.22"
```

```

# Days >20C (Drought)
FishCr_Days20_drought <- FishCr %>%
  filter(year == '2021' | year == '2022') %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  filter(max_temp > 20) %>%

```

```

group_by(logger_id,year) %>%
  summarise(days_over20 = n()) %>%
  summarise(days_over20_avg = sum(days_over20) / 2)

print(paste("# Days above 20C (Drought yrs):", FishCr_Days20_drought$days_over20_avg))

## [1] "# Days above 20C (Drought yrs): 1"

## Mean, Min, Max Temp Plot

# Data frame for plot
temp <- FishCr %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15')

x<-as.Date(temp$longdate)
y_lower = 20
y_upper = Inf

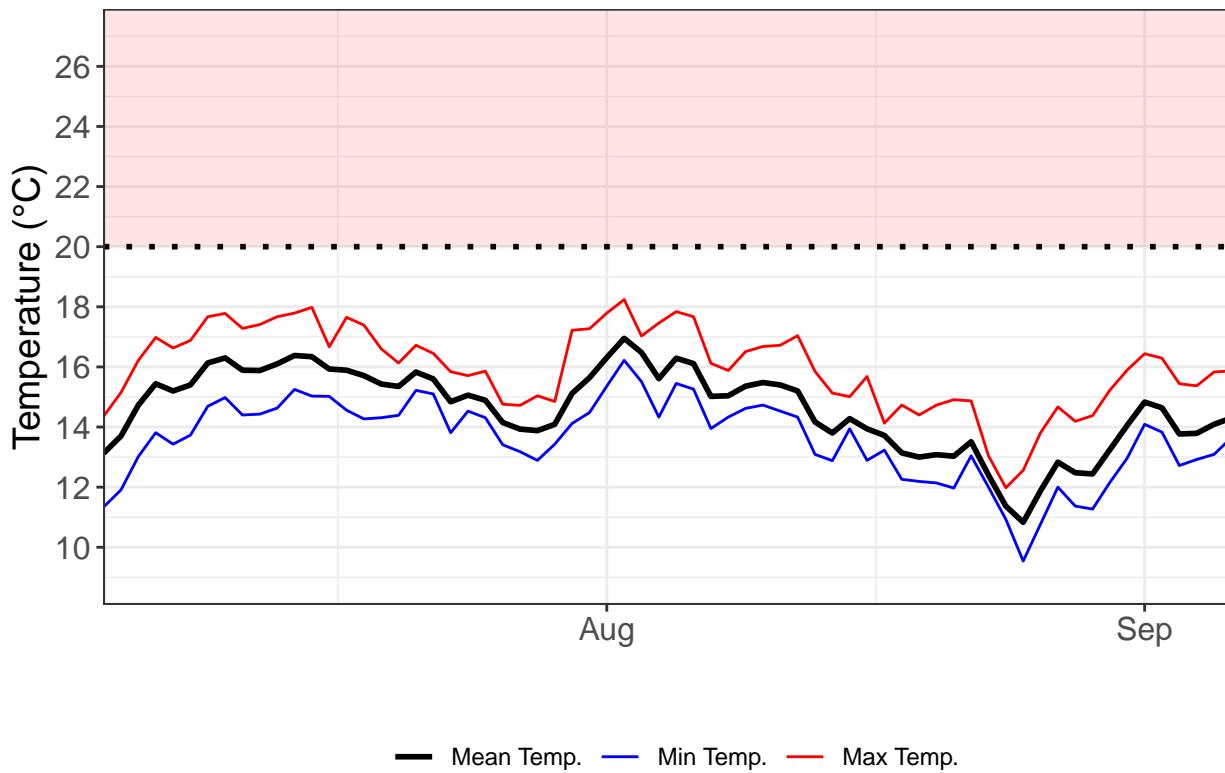
# Create temp plot
plot_15110003 <- ggplot() +
  geom_hline(yintercept = 20, linetype = "dotted", linewidth= 1) +
  geom_ribbon(aes(x=x,ymin = y_lower, ymax = y_upper), fill = "red1", alpha = 0.10) +
  geom_line(data=temp, aes(x=longdate,y=mean_temp, color="mean_temp"), linewidth = 1) +
  geom_line(data=temp, aes(x=longdate,y=min_temp, color = "min_temp"), linewidth = 0.5) +
  geom_line(data=temp, aes(x=longdate, y=max_temp), color= "max_temp"), linewidth = 0.5) +
  scale_colour_manual("", 
    breaks = c("mean_temp", "min_temp", "max_temp"),
    values = c("black", "blue", "red"),
    labels = c("Mean Temp.", "Min Temp.", "Max Temp."))
  scale_x_date(date_breaks = "1 month", date_labels = "%b", expand = c(0,0)) +
  scale_y_continuous(limits = c(9,27),breaks = seq(10,26,by=2)) +
  theme_bw() +
  labs(title= "Fish Cr. at mouth", y="Temperature (\u00b0C)", x="")
  theme(plot.subtitle = element_text(size = 10),
    plot.title = element_text(size=14, face="bold"),
    axis.text=element_text(size=12),
    axis.title=element_text(size=14),
    legend.position = "bottom")

rm(temp)

plot_15110003

```

Fish Cr. at mouth



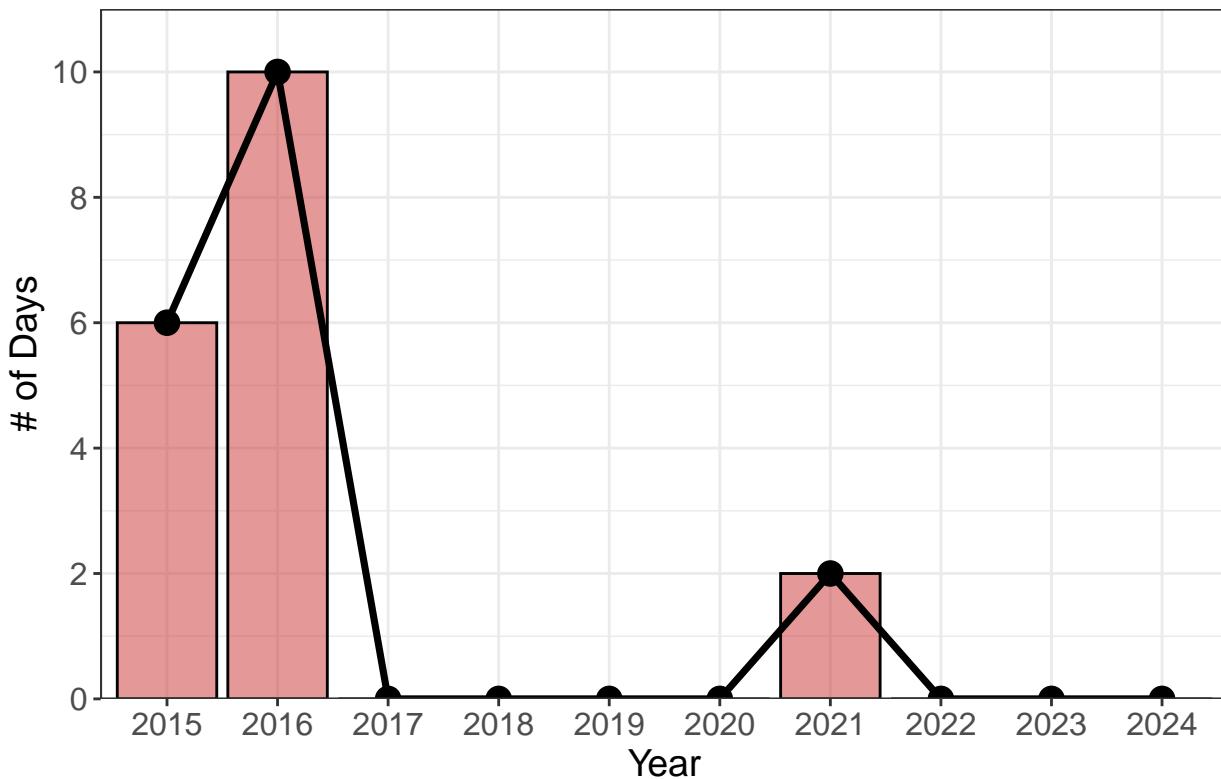
```
# Export plot to file
# ggsave(plot_15110003, file="Figure Plots/Temp Plots/Temp Plot_FishCreek.png", height = 5, width = 8) +
#
## Number of days above 20C Plot

temp <- FishCr %>%
  filter(month_day > '06-19' & month_day <= '09-15') %>%
  group_by(year) %>%
  summarize(conditional_count = sum(max_temp > 20))

FishCr_Days20_Plot <- ggplot(data=temp) +
  geom_col(aes(x=as.factor(year),y=conditional_count), fill="brown3", color="black",alpha=0.5) +
  geom_path(aes(x=as.factor(year),y=conditional_count, group = 1), size=1.25) +
  geom_point(aes(x=as.factor(year),y=conditional_count), size =4) +
  scale_y_continuous(expand = c(0,0), limits = c(0,11), breaks = seq(0,10,by=2)) +
  theme_bw() +
  theme(legend.position = "none") +
  labs(title= "Fish Cr. at mouth", y="# of Days", x="Year") +
  theme(plot.subtitle = element_text(size = 10),
        plot.title = element_text(size=18, face="bold")) +
  theme(axis.text=element_text(size=12),
        axis.title=element_text(size=14))

FishCr_Days20_Plot
```

Fish Cr. at mouth



```
#ggsave(FishCr_Days20_Plot, file="Figure Plots/Temp Plots/DaysOver20C_FishCreek.png", height = 5, width = 10)
```

```
## Air/Water Regression Plot

# Prepare air logger data for regression
air <- daily_air_master %>%
  filter(logger_id == "21433138") %>% # Copeland Cr. at mouth air logger
  mutate(longdate = make_date(year,month,day)) %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(air_mean = mean_temp)

# Prepare individual water logger data for regression
water <- FishCr %>%
  filter(longdate > '2024-06-19' & longdate <= '2024-09-15') %>%
  mutate(water_max = max_temp)

# Join air/water data frames for linear regression
airwater_join <- water %>%
  left_join(air, by = "longdate")

# Create equation formula for regression equation

lm_eqn <- function(df){
  m <- lm(water_max ~ air_mean, df);
  eq <- substitute(italic(y) == a + b %.% italic(x)*",", "~~italic(r)^2~"="~r2,
```

```

        list(a = format(unname(coef(m)[1]), digits = 3),
              b = format(unname(coef(m)[2]), digits = 3),
              r2 = format(summary(m)$adj.r.squared, digits = 3)))
    as.character(as.expression(eq));
}

# Calculate Pearson correlation coefficient
correlation <- cor(airwater_join$water_max, airwater_join$air_mean, method = 'pearson')
print(paste("Pearson R:",correlation))

## [1] "Pearson R: 0.887893235248857"

# Summary linear model fit
fit <- lm(water_max ~ air_mean, data = airwater_join)
summary(fit)

##
## Call:
## lm(formula = water_max ~ air_mean, data = airwater_join)
##
## Residuals:
##       Min     1Q   Median     3Q    Max 
## -1.6264 -0.4055  0.1052  0.3724  1.7061 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 7.89711   0.52886  14.93   <2e-16 ***
## air_mean     0.40069   0.02595  15.44   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6411 on 64 degrees of freedom
## Multiple R-squared:  0.7884, Adjusted R-squared:  0.785 
## F-statistic: 238.4 on 1 and 64 DF,  p-value: < 2.2e-16

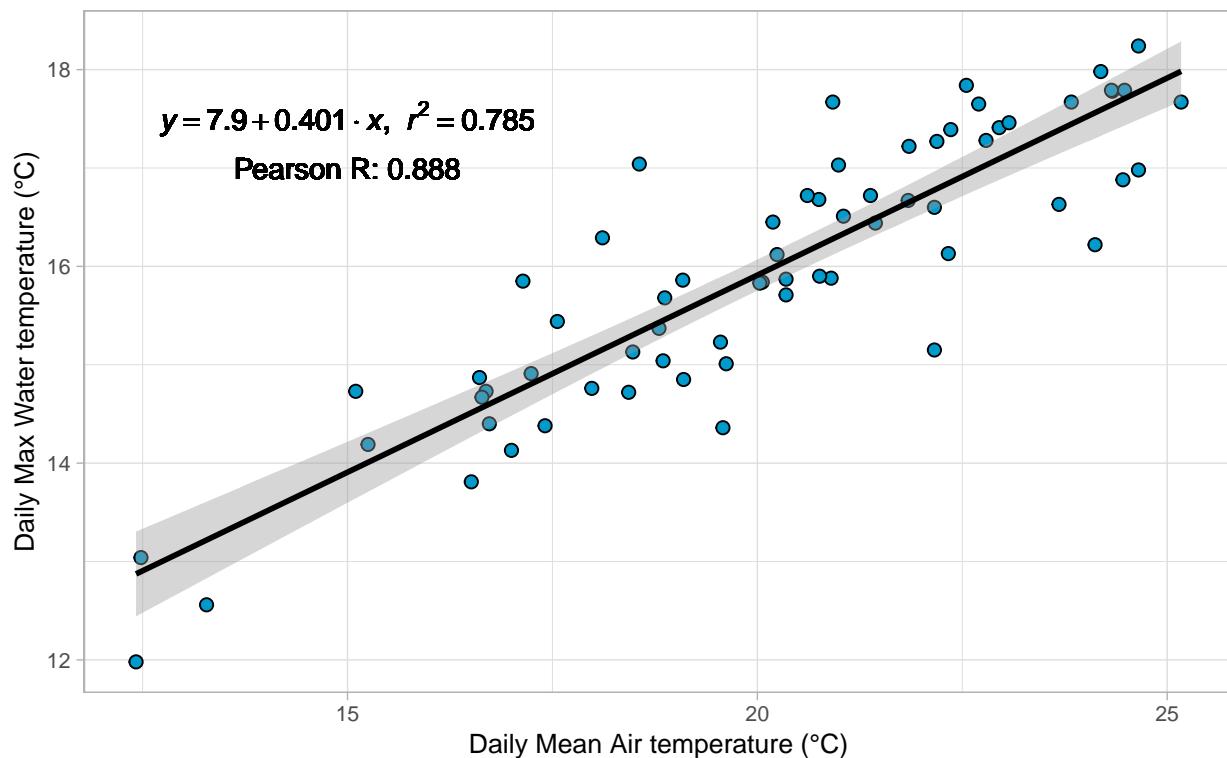
# Create regression plot
reg_plot_15110003 <- ggplot(airwater_join, aes(x=air_mean, y = water_max)) +
  geom_point(shape= 21, color='black', fill= 'deepskyblue3', size = 2) +
  geom_smooth(method = "lm", se = TRUE, colour = 'black') +
  theme_light() +
  labs(title="Fish Cr. at mouth (Air - Copeland Cr. at mouth)",
       subtitle = "Max Water Temp ~ Mean Air Temp",
       x="Daily Mean Air temperature (\u00B0C)",
       y="Daily Max Water temperature (\u00B0C)") +
  geom_text(x = 15, y = 17.5, label = lm_eqn(airwater_join), parse = TRUE) +
  geom_text(x = 15, y = 17, label = paste("Pearson R:", round(correlation,3))) +
  theme(plot.title = element_text(size=12, face="bold"),
        plot.subtitle = element_text(size = 11),
        axis.text=element_text(size=8),
        axis.title=element_text(size=10))

reg_plot_15110003

```

Fish Cr. at mouth (Air – Copeland Cr. at mouth)

Max Water Temp ~ Mean Air Temp



```
#Export plot to file
#ggsave(reg_plot_15110003, file="Figure Plots/Regression Plots/regression_plot_FishCreek.png", height =
#-----
## Climate Scenario (3.5C air temp warming)

# Create vector of new air temps under 3.5C warming scenario
new_temps <- data.frame(air_mean = airwater_join$air_mean + 3.5)

# Use lm to predict new water temps using updated air temps
watertemp_pred <- predict(fit, newdata = new_temps)

# Estimate number of days water temp above 20C given future climate scenario
daysover20_future <- data.frame(daysover20 = sum(watertemp_pred > 20))
print(paste("Predicted # days above 20C (+3.5C future climate):", {daysover20_future$daysover20}))

## [1] "Predicted # days above 20C (+3.5C future climate): 0"

rm(new_temps)
rm(watertemp_pred)
```