TEMP temperature comparison

Eddie Wu

2023-07-26

Introduction

This .Rmd file is to show the results of water temperature comparison between the global model prediction and the tributary field observation for 11 Great Lake locations. (missing Saginaw data)

For all the graphs below, the black line is the mean weekly average temperature, the blue lines are the 95% confidence interval, the red dots are field observations by week.

```
library(ggplot2)
library(dplyr)
library(knitr)
```

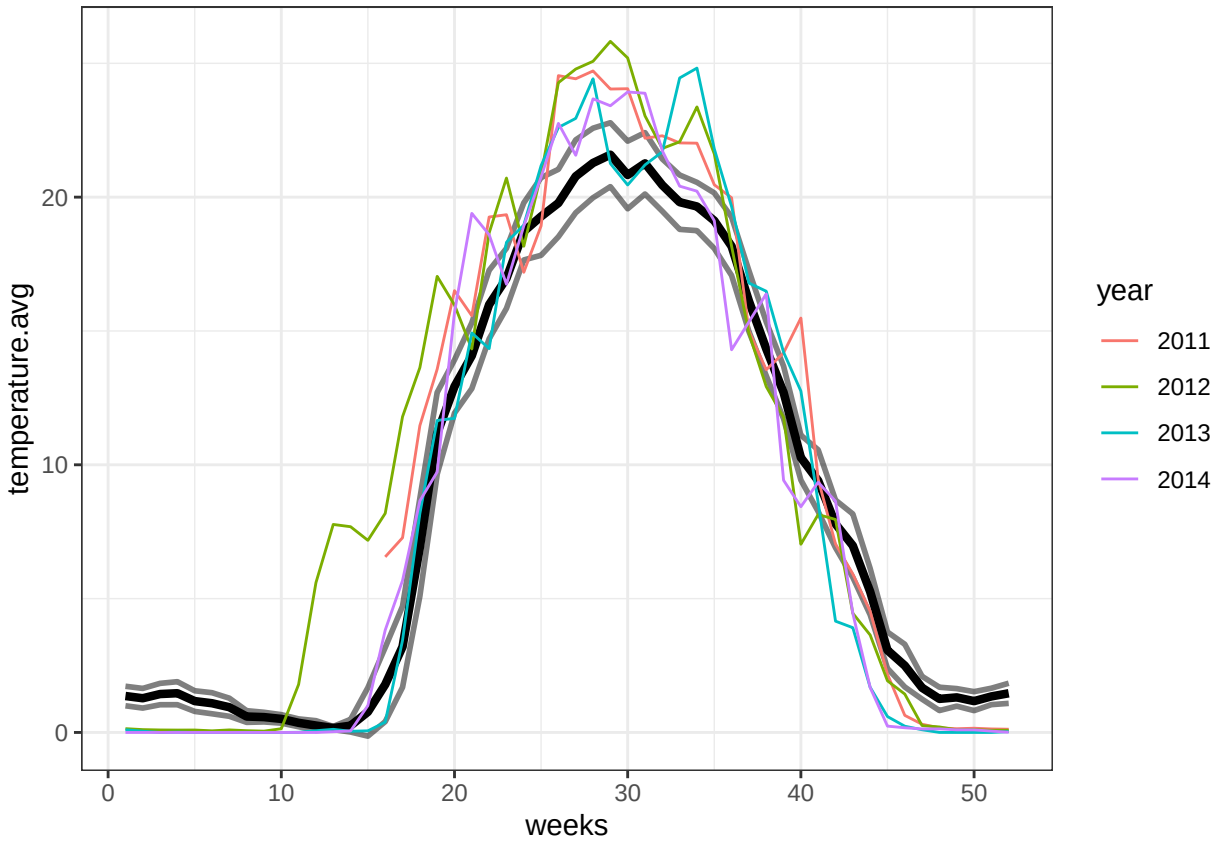
```
WT <- read.csv("water temperature clean/water_temperature_final_clean.csv",
               stringsAsFactors = TRUE)
WT$year <- as.factor(WT$year)
```

St. Louis River

```
# observed values
st.louis.field <- WT %>% filter(river == "stlouis")

# predicted values
st.louis.pred <- read.csv("water temperature clean/st_louis_model.csv")

## Plotting
ggplot(st.louis.pred, aes(x = weeks, y = temperature.avg))+
  geom_line(size = 1.5)+
  geom_line(aes(x = weeks, y = lower.CI), size = 1, alpha = 0.5)+
  geom_line(aes(x = weeks, y = upper.CI), size = 1, alpha = 0.5)+
  geom_line(st.louis.field, mapping = aes(x = weeks, y = temp, color = year))+
  scale_colour_hue()+
  theme_bw()
```




```
## Calculate the Root Mean Square Error (RMSE)
# Remove the observations from non-complete years
st.louis.field.c <- st.louis.field[st.louis.field$complete == 1,]

# Get the unique levels of the "year" variable
unique_year <- unique(st.louis.field.c$year)
unique_year
```

```
## [1] 2012 2013 2014
## 14 Levels: 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2011 2012 ... 2014
```

```
# Create a vector list to store the results
rmse_list <- vector("numeric", 0)

# Loop through each level and calculate the RMSE
for (year.number in unique_year) {
  sub <- subset(st.louis.field.c, year == year.number) # create subsets
  value <- sqrt(mean((sub$temp - st.louis.pred$temperature.avg)^2))
  rmse_list[year.number] <- value
}

rmse_list
```

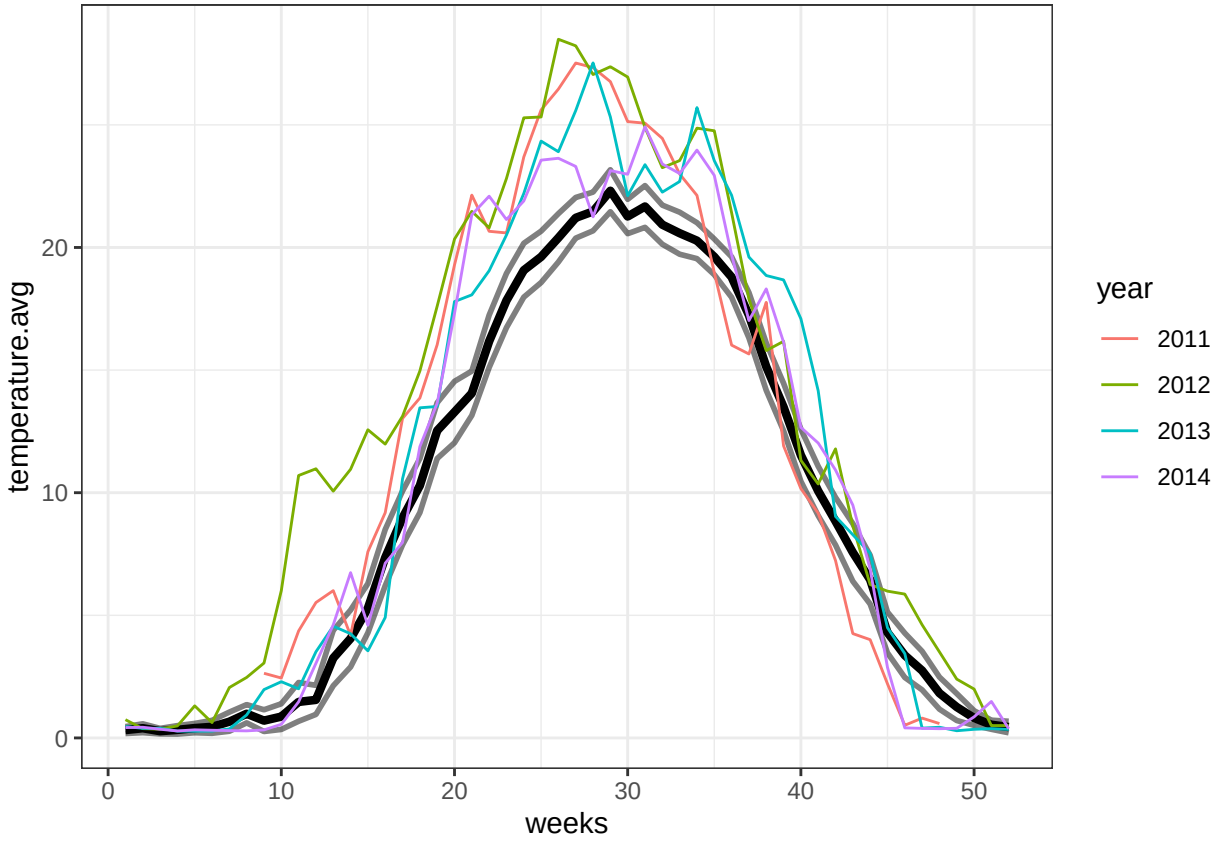
```
##     2012     2013     2014
## 3.332671 1.862503 1.898721
```

Fox River

```
# observed values
fox.field <- WT %>% filter(river == "fox")

# predicted values
fox.pred <- read.csv("water temperature clean/fox_model.csv")

## Plotting
ggplot(fox.pred, aes(x = weeks, y = temperature.avg))+
  geom_line(size = 1.5)+
  geom_line(aes(x = weeks, y = lower.CI), size = 1, alpha = 0.5)+
  geom_line(aes(x = weeks, y = upper.CI), size = 1, alpha = 0.5)+
  geom_line(fox.field, mapping = aes(x = weeks, y = temp, color = year))+
  theme_bw()
```




```
## Calculate the Root Mean Square Error (RMSE)
# Remove the observations from non-complete years
fox.field.c <- fox.field[fox.field$complete == 1,]

# Get the unique levels of the "year" variable
unique_year <- unique(fox.field.c$year)
unique_year
```

```
## [1] 2012 2013 2014
## 14 Levels: 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2011 2012 ... 2014
```

```r
# Create a vector list to store the results
rmse_list <- vector("numeric", 0)

# Loop through each level and calculate the RMSE
for (year.number in unique_year) {
  sub <- subset(fox.field.c, year == year.number) # create subsets
  value <- sqrt(mean((sub$temp - fox.pred$temperature.avg)^2))
  rmse_list[year.number] <- value
}

rmse_list
```

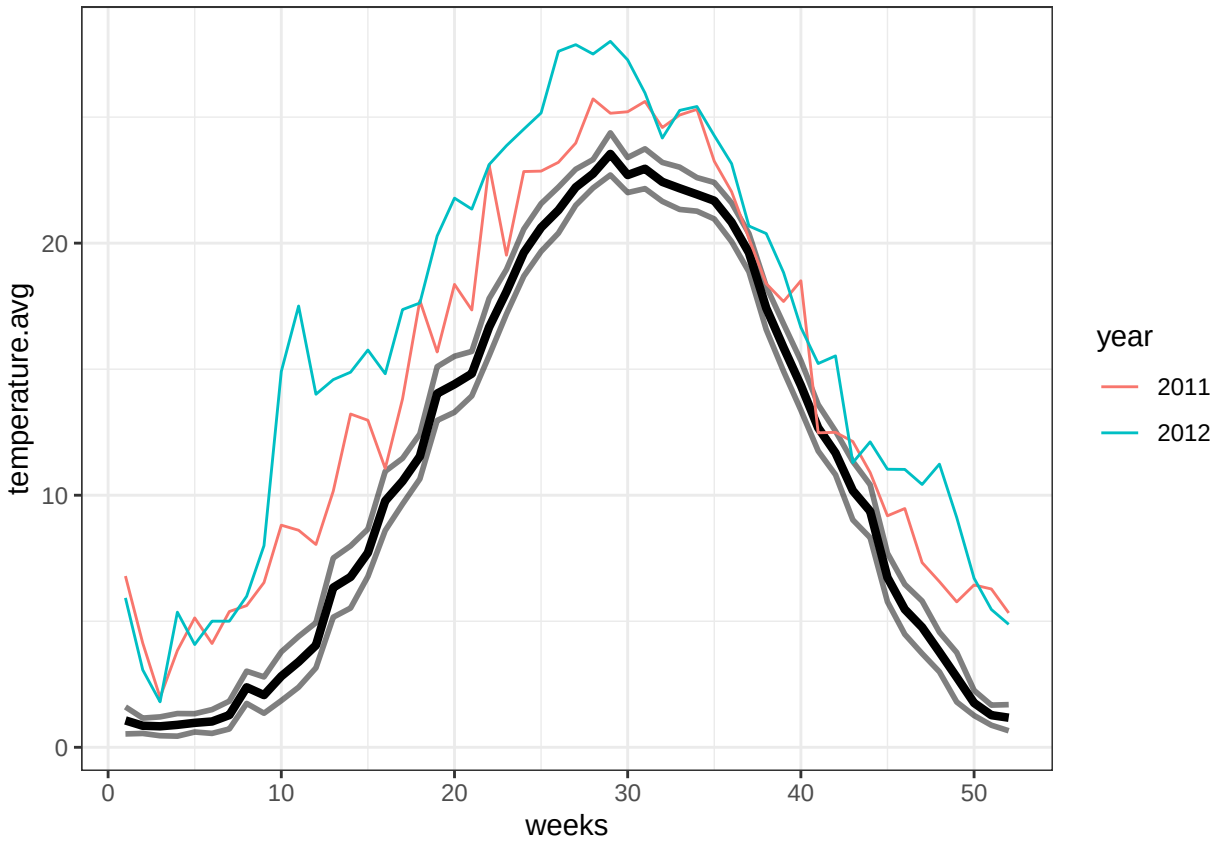
```
##     2012     2013     2014
## 4.352846 2.643812 2.293065
```

Portage-Burns Waterways

```r
# observed values
pb.field <- WT %>% filter(river == "portage")

# predicted values
pb.pred <- read.csv("water temperature clean/pb_model.csv")

## Plotting
ggplot(pb.pred, aes(x = weeks, y = temperature.avg))+
  geom_line(size = 1.5)+
  geom_line(aes(x = weeks, y = lower.CI), size = 1, alpha = 0.5)+
  geom_line(aes(x = weeks, y = upper.CI), size = 1, alpha = 0.5)+
  geom_line(pb.field, mapping = aes(x = weeks, y = temp, color = year))+
  theme_bw()
```




```
## Calculate the Root Mean Square Error (RMSE)
#' All the years are complete for pb

# Get the unique levels of the "year" variable
unique_year <- unique(pb.field$year)
unique_year
```

```
## [1] 2011 2012
## 14 Levels: 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2011 2012 ... 2014
```

```
# Create a vector list to store the results
rmse_list <- vector("numeric", 0)

# Loop through each level and calculate the RMSE
for (year.number in unique_year) {
  sub <- subset(pb.field, year == year.number) # create subsets
  value <- sqrt(mean((sub$temp - pb.pred$temperature.avg)^2))
  rmse_list[year.number] <- value
}

rmse_list
```

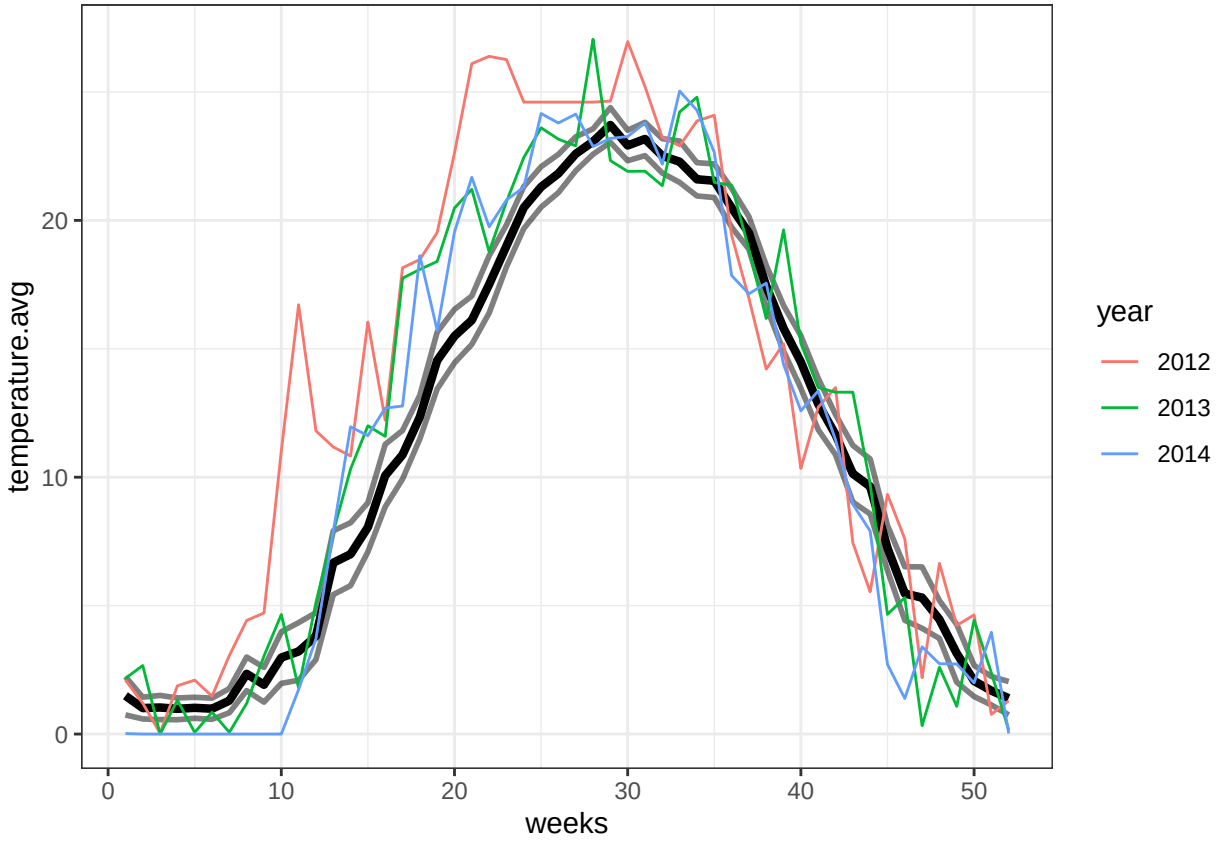
```
##     2011     2012
## 3.462018 5.569847
```

Vermilion River

```
# observed values
vermilion.field <- WT %>% filter(river == "vermilion")

# predicted values
vermilion.pred <- read.csv("water temperature clean/vermilion_model.csv")

## Plotting
ggplot(vermilion.pred, aes(x = weeks, y = temperature.avg))+
  geom_line(size = 1.5)+
  geom_line(aes(x = weeks, y = lower.CI), size = 1, alpha = 0.5)+
  geom_line(aes(x = weeks, y = upper.CI), size = 1, alpha = 0.5)+
  geom_line(vermilion.field, mapping = aes(x = weeks, y = temp, color = year))+
  theme_bw()
```




```
## Calculate the Root Mean Square Error (RMSE)
#' All years complete for vermilion

# Get the unique levels of the "year" variable
unique_year <- unique(vermilion.field$year)
unique_year
```

```
## [1] 2012 2013 2014
## 14 Levels: 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2011 2012 ... 2014
```

```r
# Create a vector list to store the results
rmse_list <- vector("numeric", 0)

# Loop through each level and calculate the RMSE
for (year.number in unique_year) {
  sub <- subset(vermilion.field, year == year.number) # create subsets
  value <- sqrt(mean((sub$temp - vermilion.pred$temperature.avg)^2))
  rmse_list[year.number] <- value
}

rmse_list
```

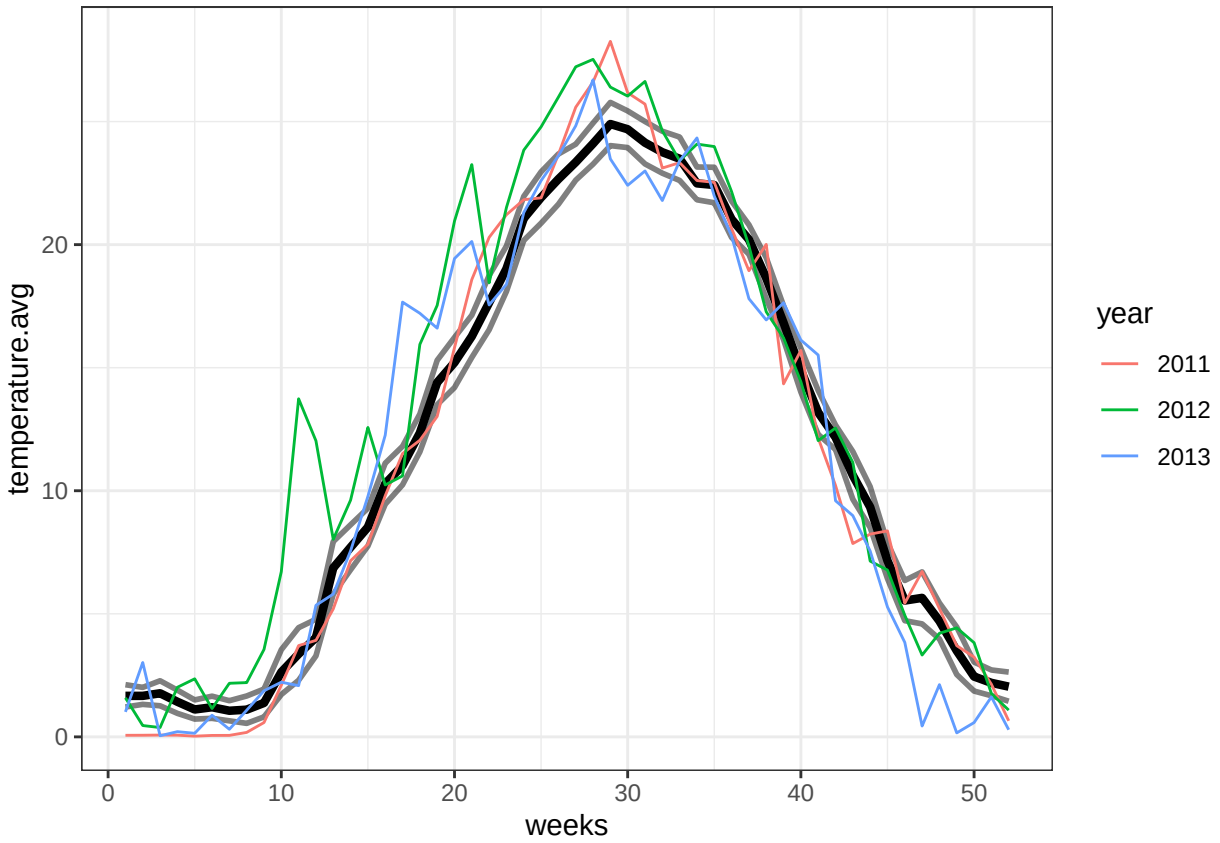
```
##     2012     2013     2014
## 4.388509 2.487505 2.324450
```

Genesee River

```r
# observed values
genesee.field <- WT %>% filter(river == "genesee")

# predicted values
genesee.pred <- read.csv("water temperature clean/genesee_model.csv")

## Plotting
ggplot(genesee.pred, aes(x = weeks, y = temperature.avg))+
  geom_line(size = 1.5)+
  geom_line(aes(x = weeks, y = lower.CI), size = 1, alpha = 0.5)+
  geom_line(aes(x = weeks, y = upper.CI), size = 1, alpha = 0.5)+
  geom_line(genesee.field, mapping = aes(x = weeks, y = temp, color = year))+
  theme_bw()
```




```
## Calculate the Root Mean Square Error (RMSE)
#' All years complete for genesee

# Get the unique levels of the "year" variable
unique_year <- unique(genesee.field$year)
unique_year
```

```
## [1] 2011 2012 2013
## 14 Levels: 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2011 2012 ... 2014
```

```
# Create a vector list to store the results
rmse_list <- vector("numeric", 0)

# Loop through each level and calculate the RMSE
for (year.number in unique_year) {
  sub <- subset(genesee.field, year == year.number) # create subsets
  value <- sqrt(mean((sub$temp - genesee.pred$temperature.avg)^2))
  rmse_list[year.number] <- value
}

rmse_list
```

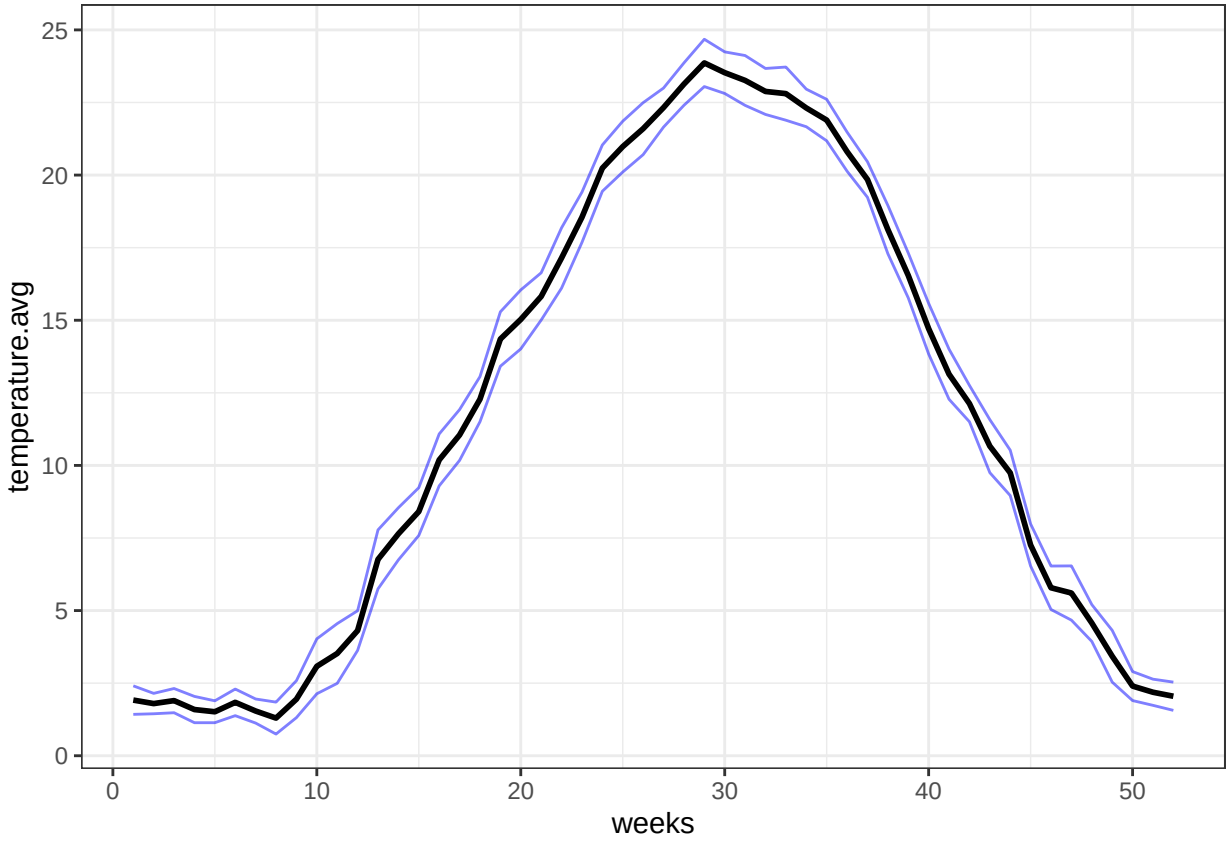
```
##     2011     2012     2013
## 1.371171 2.854283 2.140610
```

Big Creek River

```
# observed values
big.creek.field <- WT %>% filter(river == "bigcreek")

# predicted values
big.creek.pred <- read.csv("water temperature clean/bigcreek_model.csv")

## Plotting
ggplot(big.creek.pred, aes(x = weeks, y = temperature.avg))+
  geom_line(size = 1)+
  geom_line(aes(x = weeks, y = lower.CI), color = "blue", alpha = 0.5)+
  geom_line(aes(x = weeks, y = upper.CI), color = "blue", alpha = 0.5)+
  geom_point(big.creek.field, mapping = aes(x = weeks, y = temp), color = "red")+
  theme_bw()
```




```
## Calculate the Root Mean Square Error (RMSE)
RMSE.big.creek <-
  sqrt(mean((big.creek.field$temp - big.creek.pred$temperature.avg)^2))
RMSE.big.creek
```

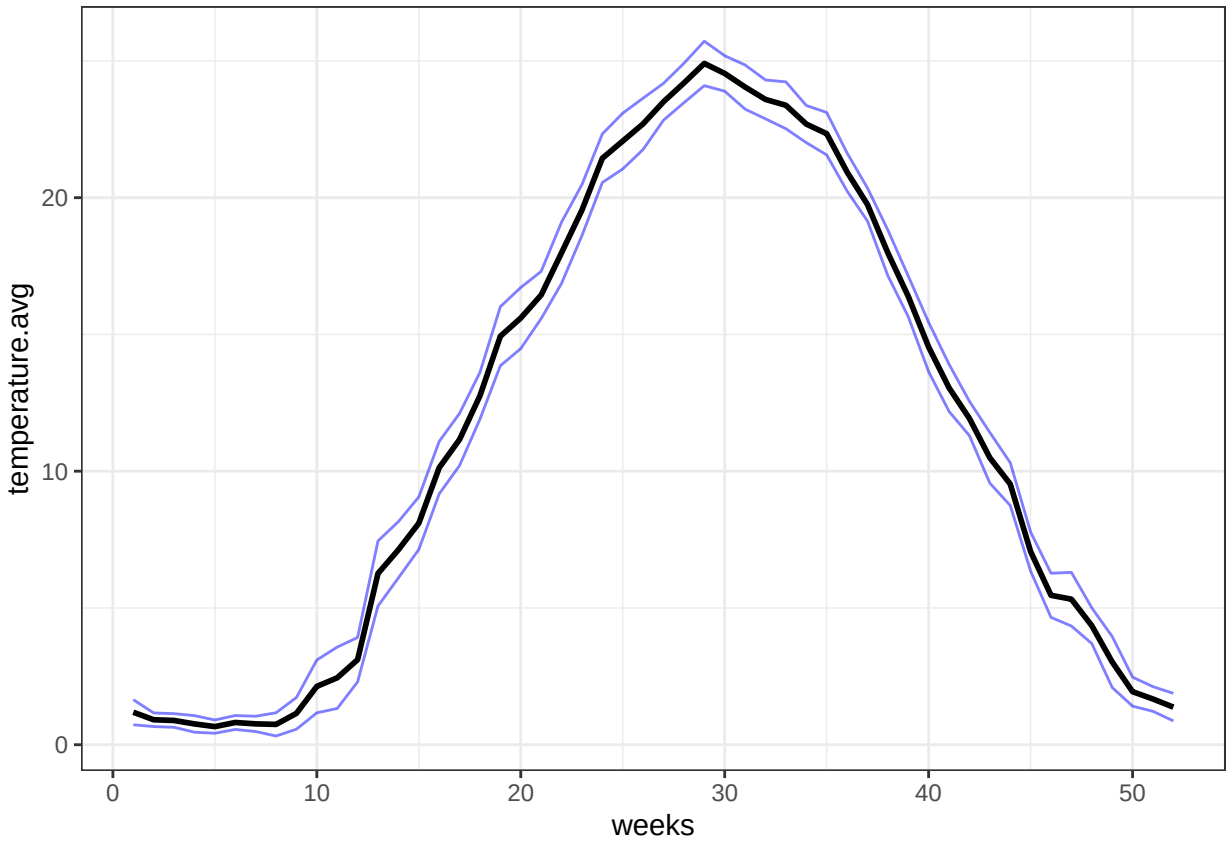
```
## [1] NaN
```

Big Otter River

```r
# observed values
big.otter.field <- WT %>% filter(river == "bigotter")

# predicted values
big.otter.pred <- read.csv("water temperature clean/bigotter_model.csv")

## Plotting
ggplot(big.otter.pred, aes(x = weeks, y = temperature.avg))+
  geom_line(size = 1)+
  geom_line(aes(x = weeks, y = lower.CI), color = "blue", alpha = 0.5)+
  geom_line(aes(x = weeks, y = upper.CI), color = "blue", alpha = 0.5)+
  geom_point(big.otter.field, mapping = aes(x = weeks, y = temp), color = "red")+
  theme_bw()
```




```r
## Calculate the Root Mean Square Error (RMSE)
RMSE.big.otter <-
  sqrt(mean((big.otter.field$temp - big.otter.pred$temperature.avg)^2))
RMSE.big.otter
```

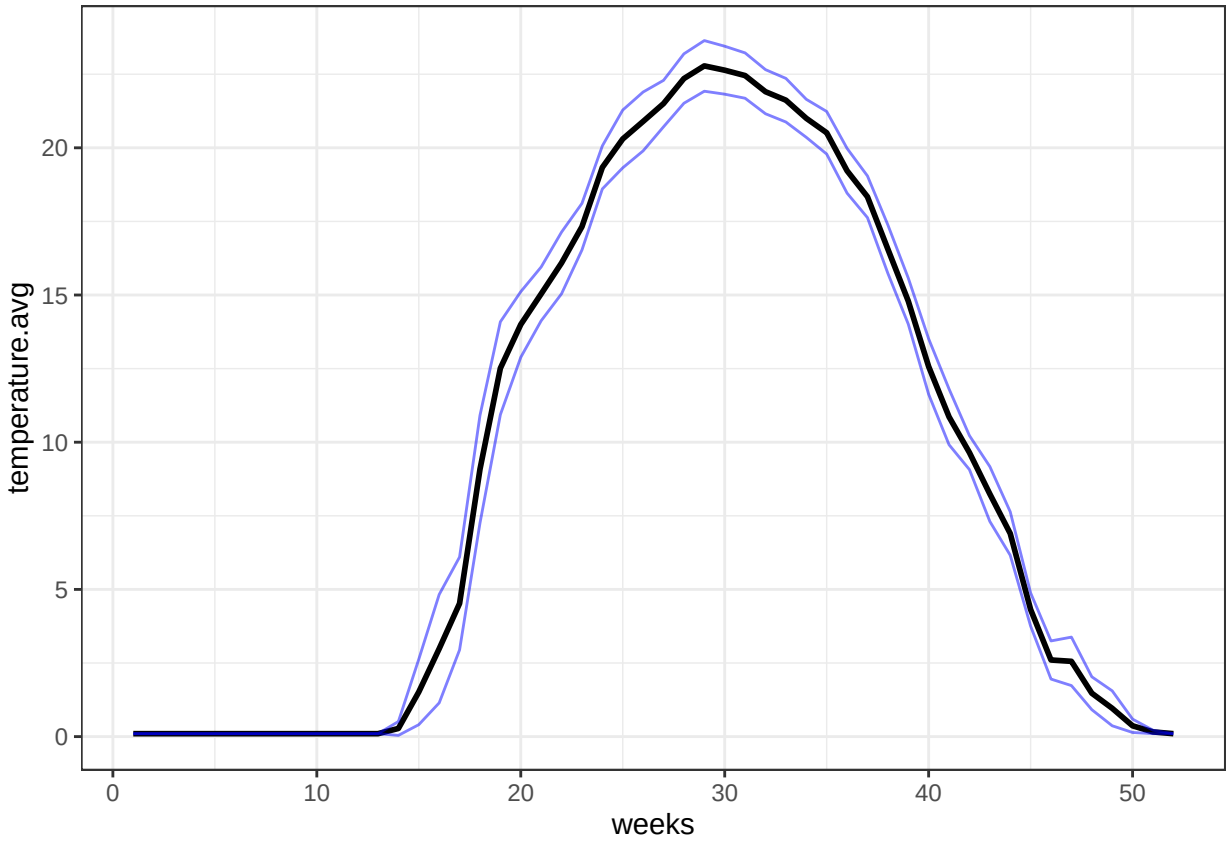
```
## [1] NaN
```

Still River

```r
# observed values
still.field <- WT %>% filter(river == "still")
```

```
# predicted values
still.pred <- read.csv("water temperature clean/still_model.csv")

## Plotting
ggplot(still.pred, aes(x = weeks, y = temperature.avg))+
  geom_line(size = 1)+
  geom_line(aes(x = weeks, y = lower.CI), color = "blue", alpha = 0.5)+
  geom_line(aes(x = weeks, y = upper.CI), color = "blue", alpha = 0.5)+
  geom_point(still.field, mapping = aes(x = weeks, y = temp), color = "red")+
  theme_bw()
```




```
## Calculate the Root Mean Square Error (RMSE)
RMSE.still <- sqrt(mean((still.field$temp - still.pred$temperature.avg)^2))
RMSE.still
```

```
## [1] NaN
```

Mississagi River

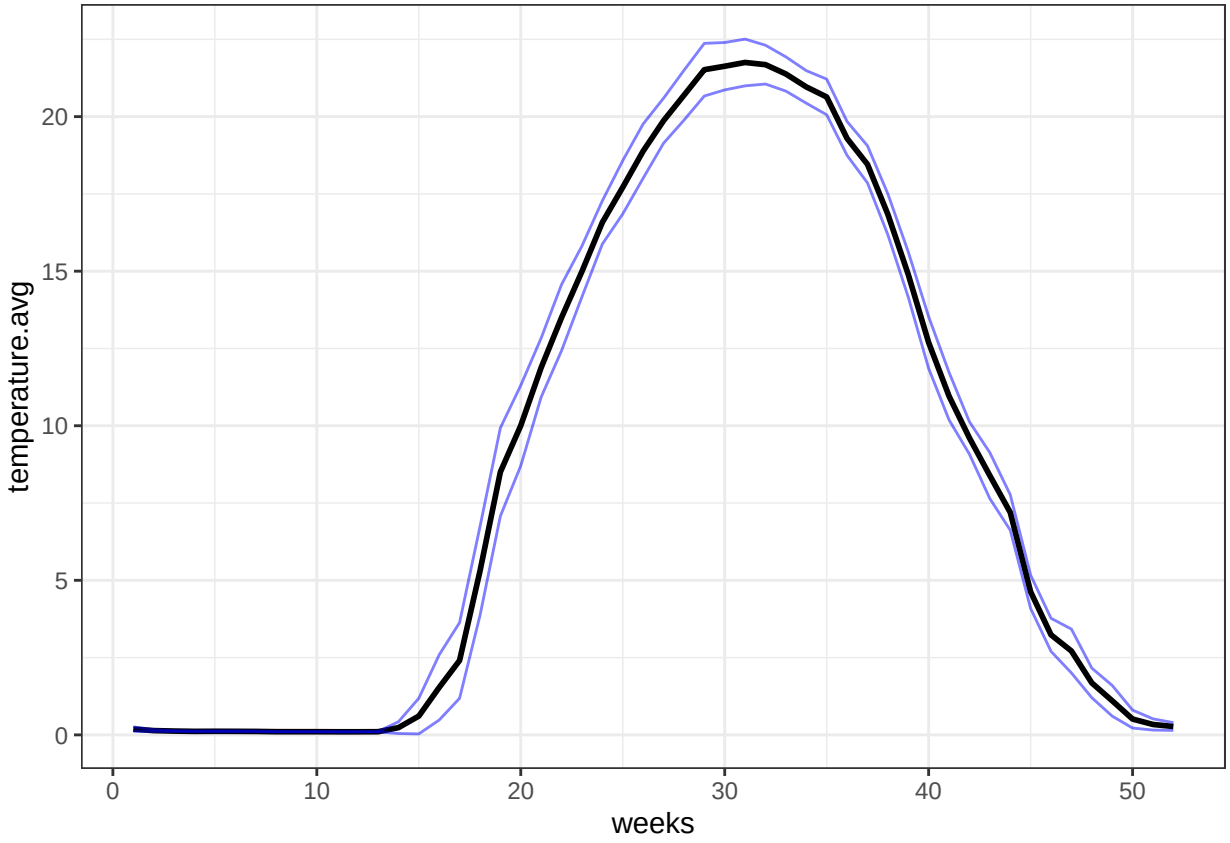
```
# observed values
mississagi.field <- WT %>% filter(river == "mississagi")
str(mississagi.field)
```

```
## 'data.frame':    0 obs. of  5 variables:
##  $ river   : Factor w/ 6 levels "big creek","fox",..:
##  $ year    : Factor w/ 14 levels "2000","2001",..:
```

```
##  $ weeks   : int
##  $ temp    : num
##  $ complete: int
```

```
# predicted values
mississagi.pred <- read.csv("water temperature clean/mississagi_model.csv")


## Plotting
ggplot(mississagi.pred, aes(x = weeks, y = temperature.avg))+
  geom_line(size = 1)+
  geom_line(aes(x = weeks, y = lower.CI), color = "blue", alpha = 0.5)+
  geom_line(aes(x = weeks, y = upper.CI), color = "blue", alpha = 0.5)+
  geom_point(mississagi.field, mapping = aes(x = weeks, y = temp), color = "red")+
  theme_bw()
```




```
## Calculate the Root Mean Square Error (RMSE)
RMSE.mississagi <-
  sqrt(mean((mississagi.field$temp - mississagi.pred$temperature.avg)^2))
RMSE.mississagi
```

```
## [1] NaN
```

Nipigon River

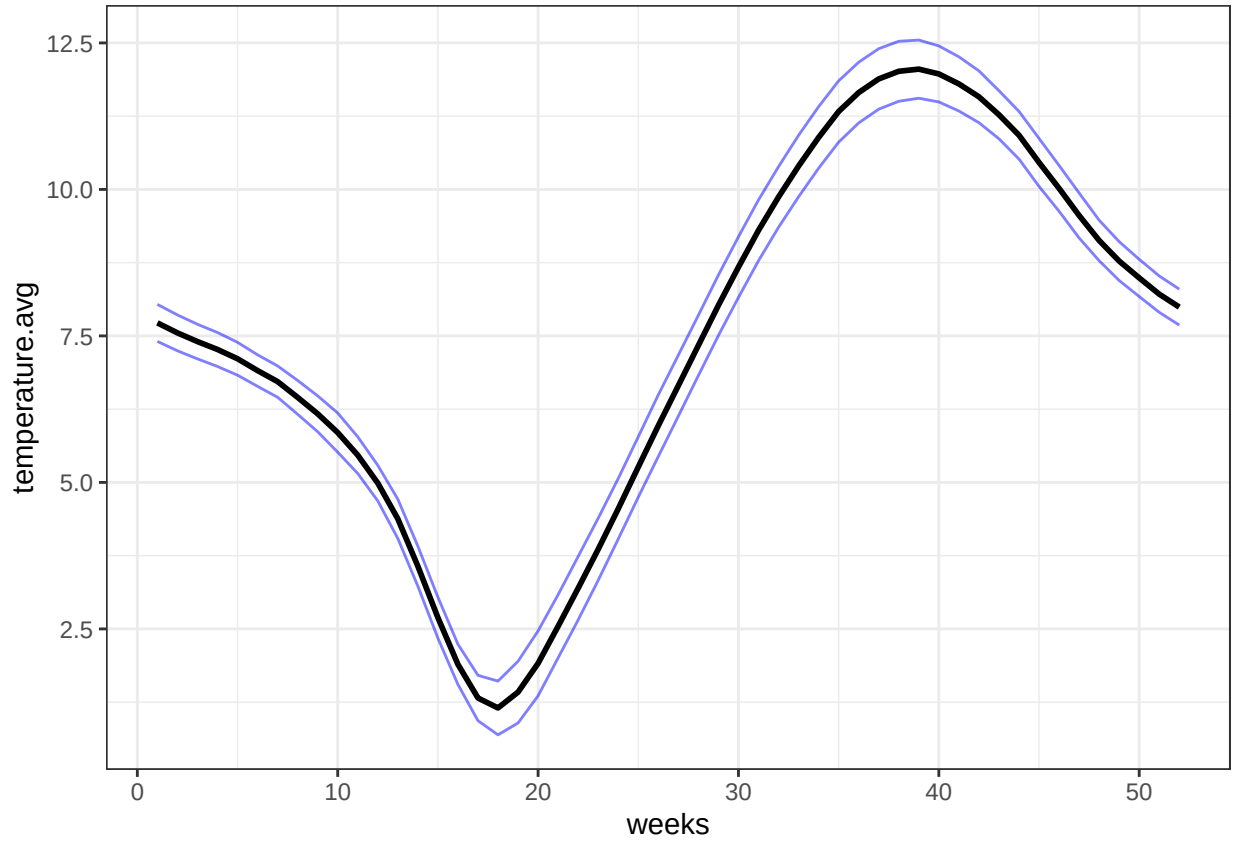
```
# observed values
nipigon.field <- WT %>% filter(river == "nipigon")
```

```
str(nipigon.field)
```

```
## 'data.frame':    0 obs. of  5 variables:
##  $ river   : Factor w/ 6 levels "big creek","fox",..:
##  $ year    : Factor w/ 14 levels "2000","2001",..:
##  $ weeks   : int
##  $ temp    : num
##  $ complete: int
```

```
# predicted values
nipigon.pred <- read.csv("water temperature clean/nipigon_model.csv")

## Plotting
ggplot(nipigon.pred, aes(x = weeks, y = temperature.avg))+
  geom_line(size = 1)+
  geom_line(aes(x = weeks, y = lower.CI), color = "blue", alpha = 0.5)+
  geom_line(aes(x = weeks, y = upper.CI), color = "blue", alpha = 0.5)+
  geom_point(nipigon.field, mapping = aes(x = weeks, y = temp), color = "red")+
  theme_bw()
```




```
## Calculate the Root Mean Square Error (RMSE)
RMSE.nipigon <- sqrt(mean((nipigon.field$temp - nipigon.pred$temperature.avg)^2))
RMSE.nipigon
```

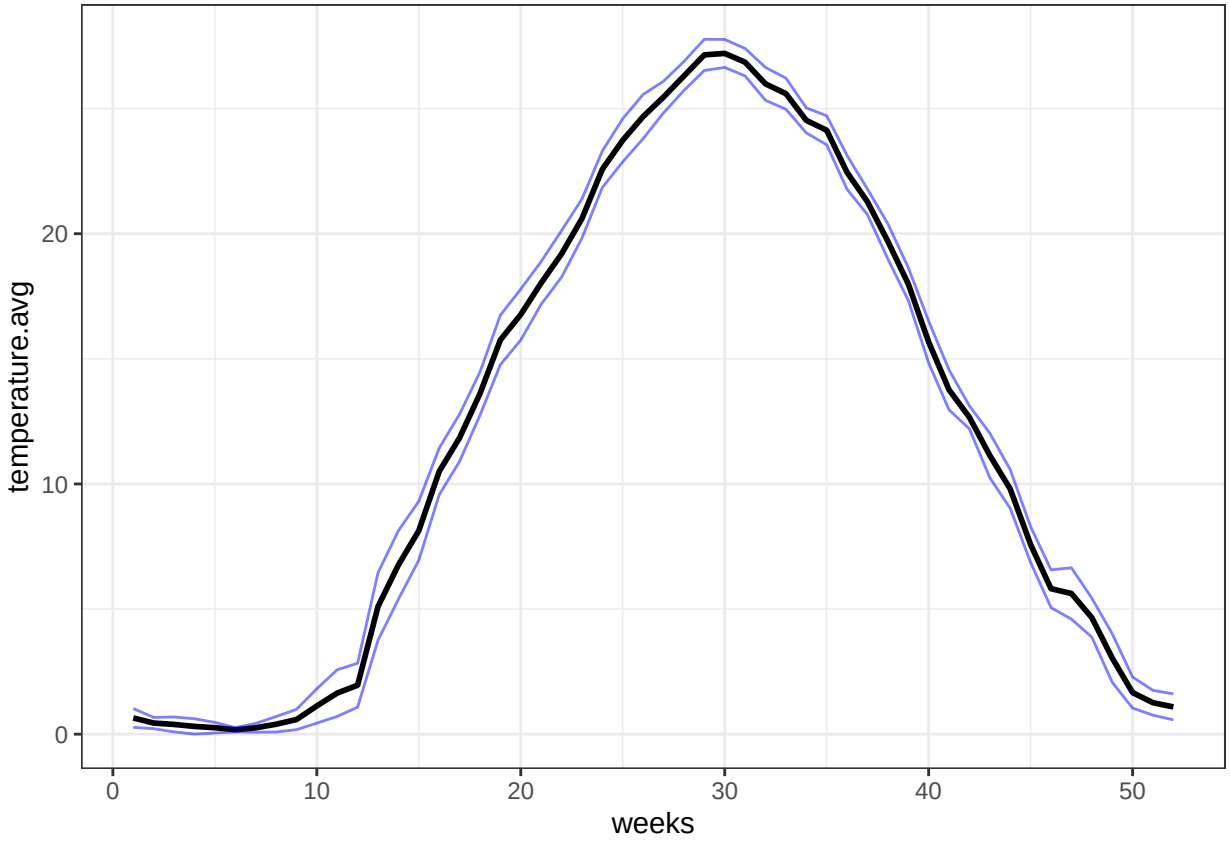
```
## [1] NaN
```

Humber River

```
# observed values
humber.field <- WT %>% filter(river == "humber")

# predicted values
humber.pred <- read.csv("water temperature clean/humber_model.csv")

## Plotting
ggplot(humber.pred, aes(x = weeks, y = temperature.avg))+
  geom_line(size = 1)+
  geom_line(aes(x = weeks, y = lower.CI), color = "blue", alpha = 0.5)+
  geom_line(aes(x = weeks, y = upper.CI), color = "blue", alpha = 0.5)+
  geom_point(humber.field, mapping = aes(x = weeks, y = temp), color = "red")+
  theme_bw()
```




```
## Calculate the Root Mean Square Error (RMSE)
RMSE.humber <- sqrt(mean((humber.field$temp - humber.pred$temperature.avg)^2))
RMSE.humber
```

```
## [1] NaN
```