

# Cherry Blossom Predictions

```
# Loading packages and reading data
library(tidyverse)

-- Attaching packages ----- tidyverse 1.3.1 --

v ggplot2 3.3.5      v purrr   0.3.4
v tibble  3.1.6      v dplyr   1.0.8
v tidyr   1.1.4      v stringr 1.4.0
v readr   2.1.1      v forcats 0.5.1

-- Conflicts ----- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()     masks stats::lag()

library(geosphere)

cherry <- read.csv("data/washingtondc.csv") %>%
  bind_rows(read.csv("data/liestal.csv")) %>%
  bind_rows(read.csv("data/kyoto.csv")) %>%
  bind_rows(read.csv("data/vancouver.csv"))

# Collect photoperiod and calculate cumulative photoperiod days
for (i in 1:nrow(cherry)) {
  total_day_length_hours <- 0
  for (j in 1:cherry$bloom_doy[i]) {
    total_day_length_hours <- total_day_length_hours + daylength(cherry$lat[i], j)}
  cherry$sum_day_length_hours[i] <- total_day_length_hours}
```

```
# Calculate cut off threshold for prediction
threshold <- mean(cherry$sum_day_length_hours)
print(paste("Average threshold calculated:", threshold))
```

```
[1] "Average threshold calculated: 1145.8736895029"
```

```
# Function to calculate bloom date by using calculated threshold

calculate_bloom_doy <- function(latitude, threshold) {
  total_day_length_hours <- 0
  day_of_year <- 1
  while (total_day_length_hours < threshold) {
    total_day_length_hours <- total_day_length_hours + daylength(latitude, day_of_year)
    day_of_year <- day_of_year + 1}
  return(day_of_year)}

# Calculate MAE/RMSE
predictions <- sapply(cherry$lat, calculate_bloom_doy, threshold = threshold)
actual <- cherry$bloom_doy
MAE <- mean(abs(predictions - actual))
RMSE <- sqrt(mean((predictions - actual)^2))
print(paste("Mean Absolute Error:", MAE))
```

```
[1] "Mean Absolute Error: 6.47383177570093"
```

```
print(paste("Root Mean Squared Error:", RMSE))
```

```
[1] "Root Mean Squared Error: 8.52182464775607"
```

```
# Calculate submission predictions
location <- c("washingtondc", "lietal", "kyoto", "vancouver", "newyorkcity")
final_predictions <- sapply(c(38.8853, 47.4814, 35.0120, 49.2237, 40.73040),
  calculate_bloom_doy, threshold = threshold)
final_predictions_df <- data.frame(location, final_predictions)
colnames(final_predictions_df) <- c("location", "prediction")
write.csv(final_predictions_df, "data/predictions.csv", row.names = F)
```

```

# Figure 1 graph creation
x_values <- seq(0, 8, by = 0.1)

y_linear <- x_values
y_square <- x_values^2
y_cubic <- x_values^3

plot(x_values, y_linear, type = "l", col = "black", ylim = c(0, 10), xlim = c(0, 3),
     xlab = "Time", ylab = "Temperature", main = "")
lines(x_values, y_square, col = "blue")
lines(x_values, y_cubic, col = "green")
segments(x0 = 1.5, y0 = 0, x1 = 3, y1 = 0, col = "red", lwd = 2)
text(x = 2.25, y = 0.4, labels = "Highly Dependent Area", col = "red", cex = 0.8)
legend("topleft", legend = c("Quetlet's GDD", "Cumulative GDD", "Temperature"),
     col = c("green", "blue", "black"), lty = 1)

```

