Homework 1

David Maimon

2025-09-11

```r
#Set Up
library(lterdatasampler)
icecover <- ntl_icecover
airtemp <- ntl_airtemp

library(tidyverse)
```

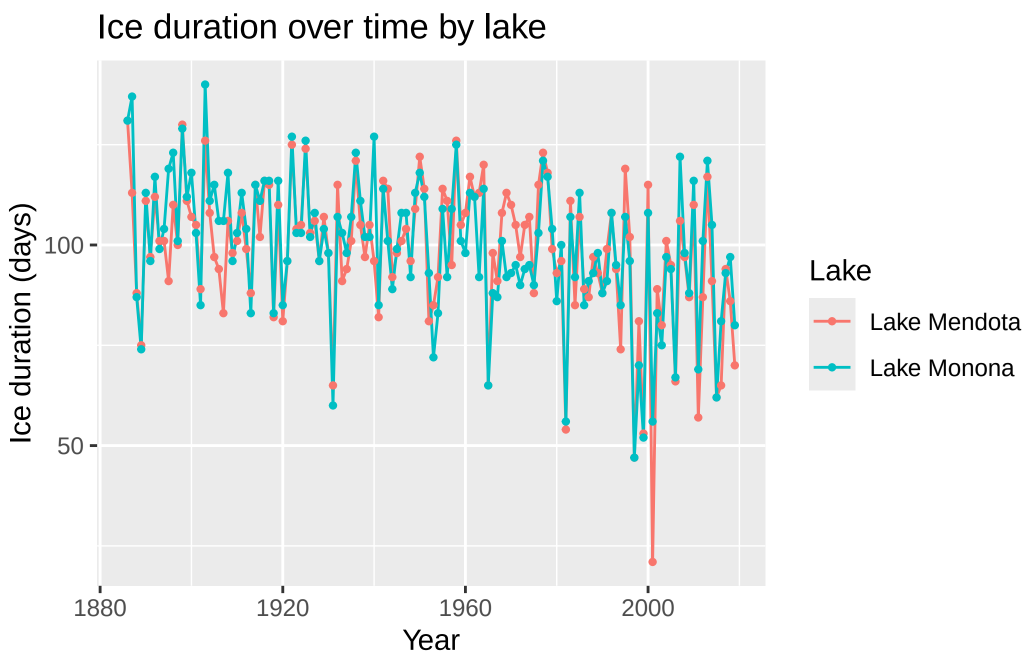
```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.2     v tibble    3.2.1
v lubridate 1.9.4     v tidyr     1.3.1
v purrr     1.0.4
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
```

```r
library(ggplot2)

#Ensuring data is within the correct year range
icecover <- icecover |>
  filter(year >= 1886, year <= 2019)

airtemp <- airtemp |>
  filter(year >= 1886, year <= 2019)

glimpse(icecover)
```

```
Rows: 268
Columns: 5
$ lakeid       <fct> Lake Mendota, Lake Mendota, Lake Mendota, Lake Mendota, L~
$ ice_on       <date> 1886-12-05, 1887-12-24, 1889-01-02, 1890-01-14, 1890-12-~
$ ice_off      <date> 1887-04-15, 1888-04-15, 1889-03-31, 1890-03-30, 1891-04-~
$ ice_duration <dbl> 131, 113, 88, 75, 111, 97, 112, 101, 101, 91, 110, 100, 1~
$ year         <dbl> 1886, 1887, 1888, 1889, 1890, 1891, 1892, 1893, 1894, 189~
```

Exercise 6

```r
##Part a
ggplot(icecover, aes(year, ice_duration, color = lakeid)) +
  geom_line() +
  geom_point(size = 0.8) +
  labs(x = "Year", y = "Ice duration (days)", color = "Lake",
       title = "Ice duration over time by lake")
```



b. Averaging reduces lake-specific noise and gives a single series per year. It can hide differences between lakes. If both lakes have similar trends, then the average will be a good summary, but otherwise, the average will hide a lot of lake-specific variation. Using the average is simpler, but if the lakes differ significantly, then averaging will hide meaningful patterns.

Exercise 7

```r
#a
airtemp_year <- airtemp |>
  group_by(year) |>
  summarise(air_temp_avg = mean(ave_air_temp_adjusted, na.rm = TRUE), .groups = "drop")
airtemp_year
```

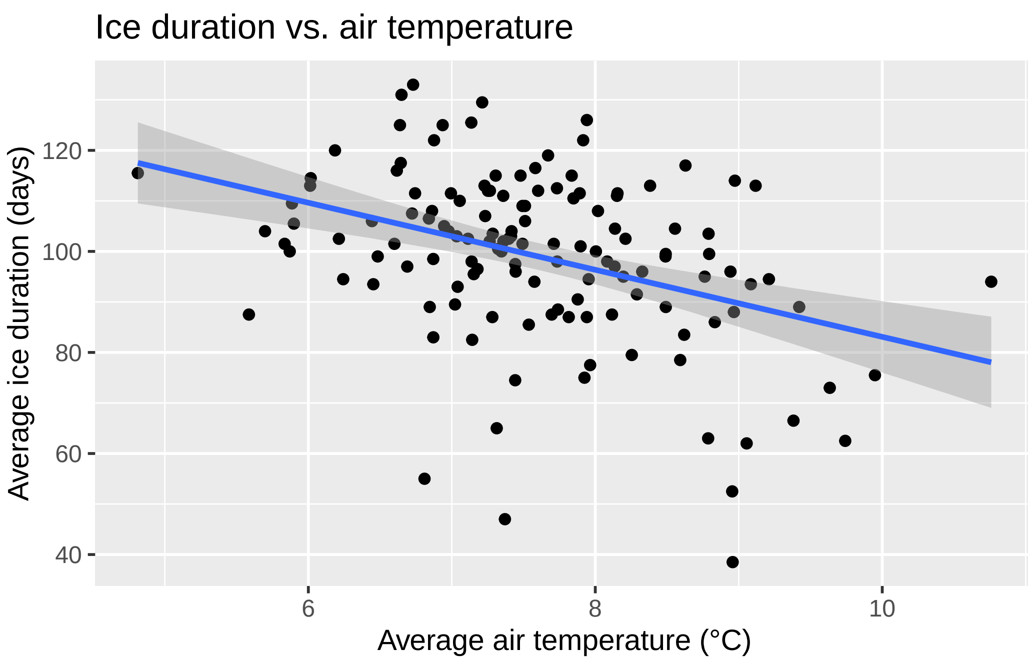
```
# A tibble: 134 x 2
    year air_temp_avg
   <dbl>        <dbl>
 1  1886         6.65
 2  1887         6.64
 3  1888         5.59
 4  1889         7.44
 5  1890         7.25
 6  1891         7.18
 7  1892         6.02
 8  1893         5.87
 9  1894         8.21
10  1895         6.95
# i 124 more rows
```

```r
analysis_df <- icecover |>
  filter(!is.na(ice_duration)) |>
  group_by(year) |>
  filter(n_distinct(lakeid) == 2) |>
  summarise(ice_duration_avg = mean(ice_duration), .groups = "drop") |>
  inner_join(airtemp_year, by = "year")

dim(analysis_df)
```

```
[1] 134   3
```

```r
#b
ggplot(analysis_df, aes(air_temp_avg, ice_duration_avg)) +
  geom_point() +
  geom_smooth(method = "lm", se = TRUE) +
  labs(x = "Average air temperature (°C)",
       y = "Average ice duration (days)",
       title = "Ice duration vs. air temperature")
```

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



The relationship look roughly linear and negative, since the spread of points around the slope line look fairly balanced, without strong curvature or other pattern. For this reason, yes, a linear model is a reasonable choice.

Exercise 8

1. y : 134 x 1
2. X: 134 x 2
3. B: 2 x 1
4. e: 134 x 1

```r
#b
y <- as.matrix(analysis_df$ice_duration_avg)
X <- cbind(Intercept = 1, Temp = analysis_df$air_temp_avg) |> as.matrix()
beta_hat <- solve(t(X) %*% X) %*% t(X) %*% y
beta_hat
```

```
                [,1]
Intercept 149.453781
Temp       -6.636303
```

```r
#c
fit <- lm(ice_duration_avg ~ air_temp_avg, data = analysis_df)
round(coef(fit), 3)
```

```
 (Intercept) air_temp_avg 
     149.454       -6.636 
```

```r
summary(fit)
```

```

Call:
lm(formula = ice_duration_avg ~ air_temp_avg, data = analysis_df)

Residuals:
    Min      1Q  Median      3Q     Max 
-53.543  -8.027   1.707  10.438  29.248 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)   149.454     10.535  14.187  < 2e-16 ***
air_temp_avg   -6.636      1.376  -4.824 3.82e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 15.58 on 132 degrees of freedom
Multiple R-squared:  0.1499,	Adjusted R-squared:  0.1434 
F-statistic: 23.27 on 1 and 132 DF,  p-value: 3.818e-06
```

Exercise 9

```
#a
r2 <- summary(fit)$r.squared
r2
```

```
[1] 0.1498813
```

This means that only 14.98% of the variation in the average ice duration is explained by the average air temperature.

```
#b
rmse <- sqrt(mean(residuals(fit)^2))
rmse
```

```
[1] 15.4623
```

The RMSE is about 15 days, meaning that on average our predictions for ice duration are off by about 2 weeks. From the the ice_duration vs. air_temp plot, we could see that typical ice durations range from 80 to 140 days, so the prediction error is fairly large in context, suggesting that while air temperature is related to ice duration, it is not the only factor.

c. Since the R^2 is low, and the RMSE is relatively high, we can say that the model fit is not great (model's predictions are imprecise).

Exercise 10

a. The slope of -6.6363 indicates that, on average, for every one degree celsius increase in average air temperate, the average_ice_duration for the 2 lakes decreases by 6.636 days.

```
tail(analysis_df)
```

```
# A tibble: 6 x 3
   year ice_duration_avg air_temp_avg
  <dbl>            <dbl>        <dbl>
1  2014               98         7.14
2  2015               62         9.06
3  2016               73         9.63
```

4	2017	93.5	9.08
5	2018	91.5	8.29
6	2019	75	7.92

b. Using our regression formula, we have that
Y-hat = 148.110 - 6.182 * (7.925) = 99.1177

So, the predicted ice duration was approximately 99 days.

The residual is calculated as (e = Y - Y-hat). This means e = (75 - 99.1177) = -24.1177 days.