STA_445_Assignment_6

Sofia Mendoza

03/26/2024

libraries

```
library(lubridate)
library(mosaicData)
library(dplyr)
library(ggplot2)
```

Problem 1

Convert the following to date or date/time objects.

a. September 13, 2010.

```
date_a <- mdy("September 13, 2010")
date_a
```

```
## [1] "2010-09-13"
```

b. Sept 13, 2010.

```
date <- "Sept 13, 2010"
date_correct_b <- gsub("Sept", "Sep", date)
date_b <- mdy(date_correct_b)
date_b
```

```
## [1] "2010-09-13"
```

c. Sep 13, 2010.

```
date_c <- mdy("Sep 13, 2010")
date_c
```

```
## [1] "2010-09-13"
```

d. S 13, 2010. Comment on the month abbreviation needs.

```
date <- "S 13, 2010"
date_correct_d <- gsub("S", "Sep", date)
date_d <- mdy(date_correct_b)
date_d
```

```
## [1] "2010-09-13"
```

e. 07-Dec-1941.

```
date_e <-  dmy("07-Dec-1941")
date_e
```

```
## [1] "1941-12-07"
```

f. 1-5-1998. Comment on why you might be wrong.

```
date_f <- mdy("1-5-1998")
date_f
```

```
## [1] "1998-01-05"
```

I could be wrong because we don't know if the 1 or the 5 represent either the day or the month so it could be printed viscervsa

g. 21-5-1998. Comment on why you know you are correct.

```
date_g <- dmy("21-5-1998")
date_g
```

```
## [1] "1998-05-21"
```

h. 2020-May-5 10:30 am

```
date_h <- ymd_hm("2020-May-5 10:30")
date_h
```

```
## [1] "2020-05-05 10:30:00 UTC"
```

i. 2020-May-5 10:30 am PDT (ex Seattle)

```
date_i <-ymd_hm ("2020-May-5 10:30", tz = "America/Los_Angeles")
date_i
```

```
## [1] "2020-05-05 10:30:00 PDT"
```

j. 2020-May-5 10:30 am AST (ex Puerto Rico)

```
date_j <- ymd_hm("2020-May-5 10:30", tz = "America/Puerto_Rico")
date_j
```

```
## [1] "2020-05-05 10:30:00 AST"
```

Problem 2

Using just your date of birth (ex Sep 7, 1998) and today's date calculate the following:

a. Calculate the date of your 64th birthday.

```
birthday <- ymd("2004-04-02")
today <- ymd("2024-03-28")

birthday_64 <- birthday + years(64)
birthday_64
```

```
## [1] "2068-04-02"
```

b. Calculate your current age (in years).

```
current <- interval(birthday, today) %>% as.period() %>%year()
current
```

```
## [1] 19
```

c. Using your result in part (b), calculate the date of your next birthday.

```
nextbday <- birthday+years(current+1)
nextbday
```

```
## [1] "2024-04-02"
```

d. The number of *days* until your next birthday. CHECK

```
days <- difftime(nextbday, today, units = "days")
days
```

```
## Time difference of 5 days
```

e. The number of *months* and *days* until your next birthday.

```
paste(as.period(interval(today, nextbday))$month, "months and", as.period(interval(today, nextbday))$day
```

```
## [1] "0 months and 5 days"
```

Problem 3

Suppose you have arranged for a phone call to be at 3 pm on May 8, 2015 at Arizona time. However, the recipient will be in Auckland, NZ. What time will it be there?

```r
arizona_time <- with_tz(ymd_hms("2015-05-08 15:00:00", tz = "America/Phoenix"), tzone = "America/Phoeni
auckland_time <- with_tz(arizona_time, tzone = "Pacific/Auckland")
auckland_time
```

```
## [1] "2015-05-09 10:00:00 NZST"
```

Problem 4

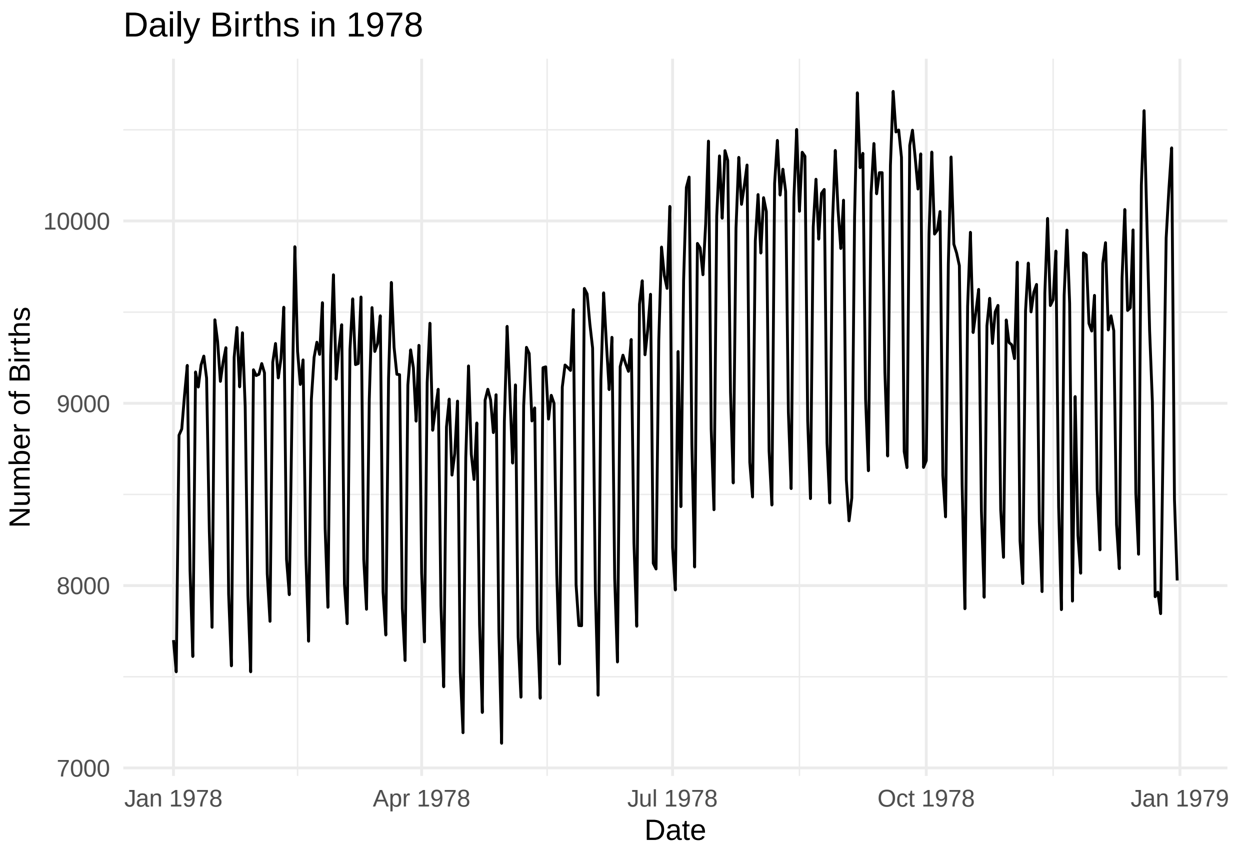
It turns out there is some interesting periodicity regarding the number of births on particular days of the year.

a. Using the `mosaicData` package, load the data set `Births78` which records the number of children born on each day in the United States in 1978. Because this problem is intended to show how to calculate the information using the `date`, remove all the columns *except* `date` and `births`.

```r
data(Births78)
Births78_cleaned <- Births78 %>% select(date, births)
```

b. Graph the number of `births` vs the `date` with date on the x-axis. What stands out to you? Why do you think we have this trend?

```r
ggplot(Births78_cleaned, aes(x = date, y = births)) +
  geom_line() +
  theme_minimal() +
  labs(title = "Daily Births in 1978", x = "Date", y = "Number of Births")
```



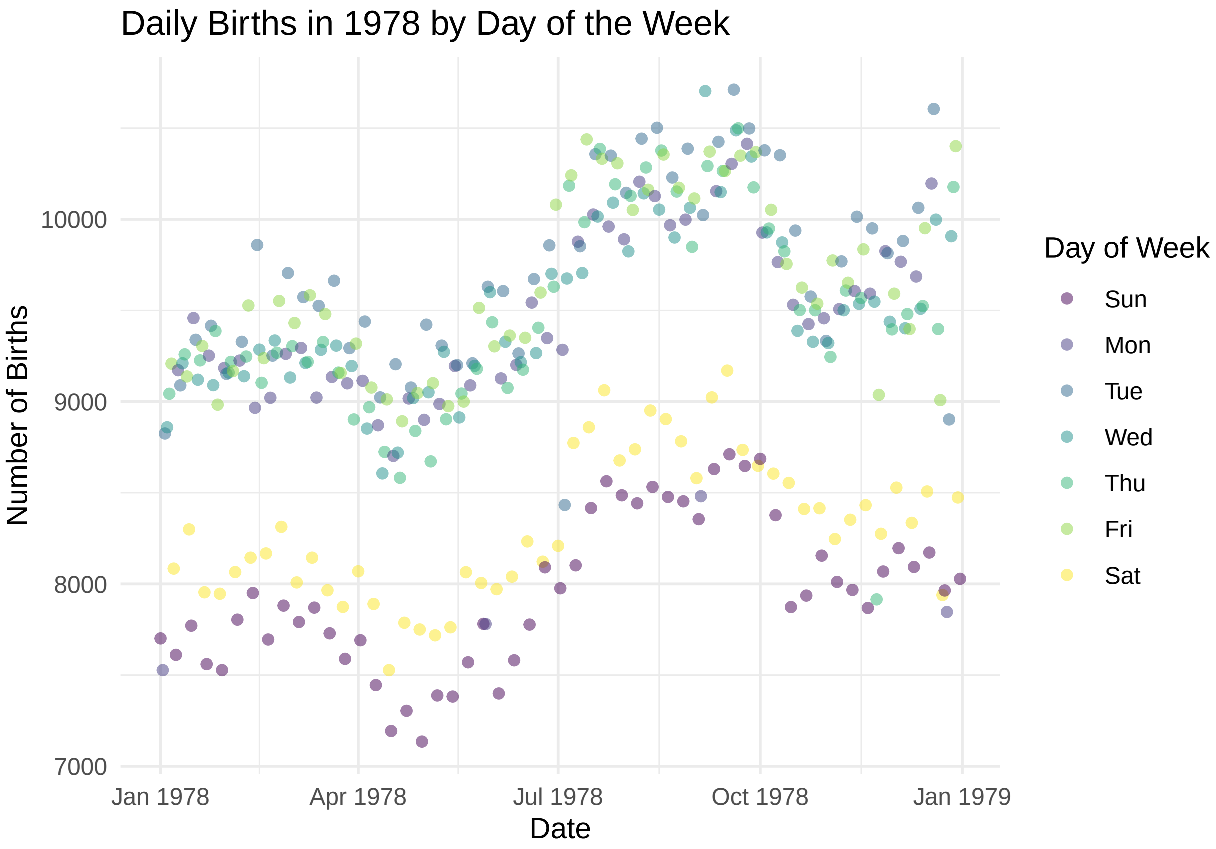
we can tell that as the time goes by the births increase but we can also tell that during summer and fall is when there are the most births of the year.

c. To test your assumption, we need to figure out the what day of the week each observation is. Use `dplyr::mutate` to add a new column named `dow` that is the day of the week (Monday, Tuesday, etc). This calculation will involve some function in the `lubridate` package and the `date` column.

```r
Births78_cleaned <- Births78_cleaned %>%
  mutate(dow = wday(date, label = TRUE))
```

d. Plot the data with the point color being determined by the day of the week variable.

```r
ggplot(Births78_cleaned, aes(x = date, y = births, color = dow)) +
  geom_point(alpha = 0.5) +
  theme_minimal() +
  labs(title = "Daily Births in 1978 by Day of the Week", x = "Date", y = "Number of Births", color = "
```


Daily Births in 1978 by Day of the Week
Number of Births
Jan 1978
Apr 1978
Jul 1978
Oct 1978
Jan 1979
Date
Day of Week
Sun
Mon
Tue
Wed
Thu
Fri
Sat