STA_445_Assignment_6

Mason Nabbefeld

2024-03-28

Problem 1

Convert the following to date or date/time objects.

a. September 13, 2010.

```
mdy("September 13, 2010")
```

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

b. Sept 13, 2010.

```
mdy("Sept 13, 2010")
```

```
## Warning: All formats failed to parse. No formats found.
```

```
## [1] NA
```

R doesn't like this abbreviation for September.

c. Sep 13, 2010.

```
mdy("Sep 13, 2010")
```

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

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

```
mdy("S 13, 2010")
```

```
## Warning: All formats failed to parse. No formats found.
```

```
## [1] NA
```

R doesn't like this abbreviation for September, Sep seems to be the only acceptable one.

e. 07-Dec-1941.

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

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

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

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

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

We don't know if the 1 is the month or the day, and same with 5.

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

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

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

We know for a fact that 5 is the month now because there are not 21 months.

h. 2020-May-5 10:30 am

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

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

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

```
ymd_hm("2020-May-5 10:30 am", tz='US/Pacific')
```

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

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

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

```
## [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.

```r
dob <- make_date(2003, 10, 23)
today <- today()
dob + years(64)
```

```
## [1] "2067-10-23"
```

b. Calculate your current age (in years).

```r
age = dob%--%today
timePeriod <- as.period(age)
year(timePeriod)
```

```
## [1] 20
```

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

```r
nextBday <- make_date(year(today), month(dob), day(dob))
nextBday
```

```
## [1] "2024-10-23"
```

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

```r
daysTilNext = today%--%nextBday
timePeriod <- as.period(daysTilNext, "days")
timePeriod
```

```
## [1] "209d 0H 0M 0S"
```

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

```r
timePeriod <- as.period(daysTilNext, "month", "days")
timePeriod
```

```
## [1] "6m 25d 0H 0M 0S"
```

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
phoneCallAZ <- ymd_hm("2015-May-8 3:00 pm", tz='US/Arizona')
with_tz(phoneCallAZ, tz='Pacific/Auckland')
```

```
## [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.

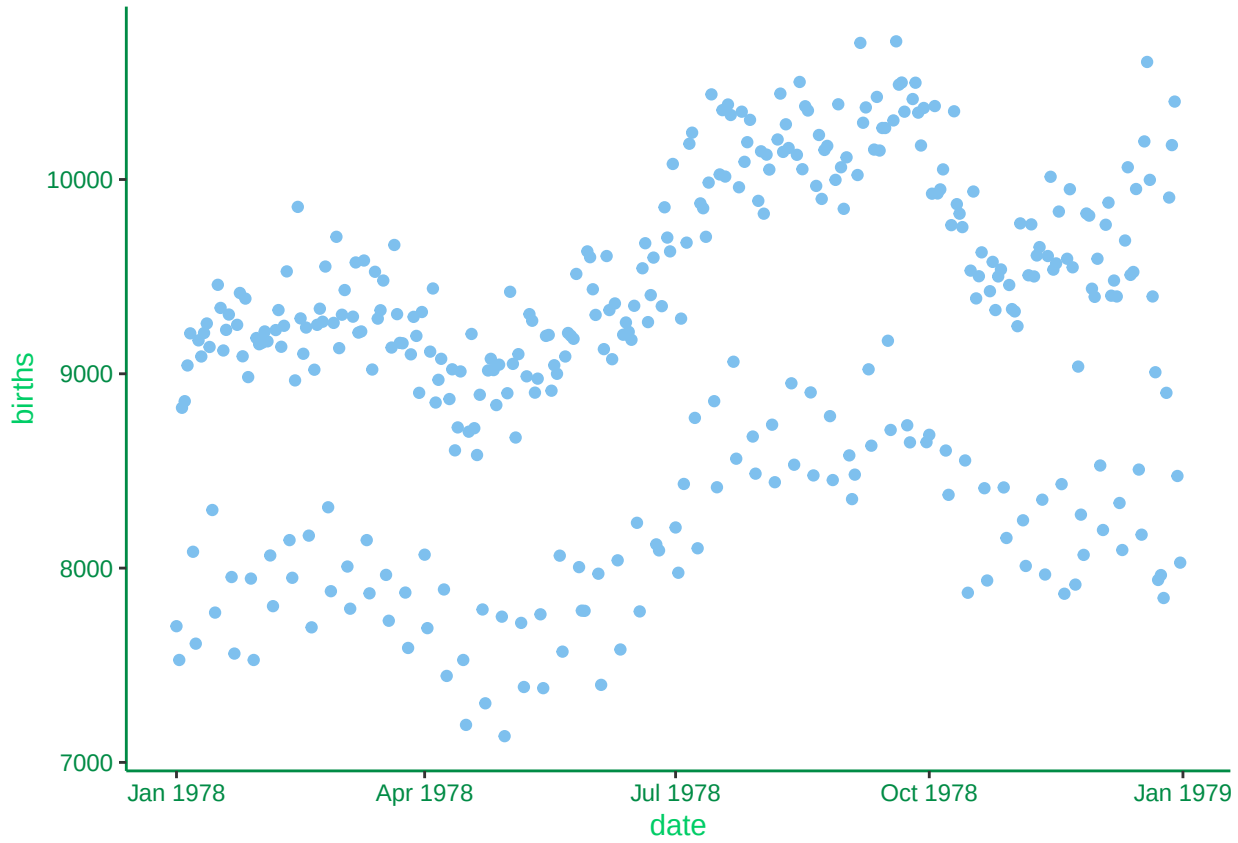
a. Using the `mosaicData` package, load the data set `Births78` which records the number of children born on each day in the United States in 1978. Because this problem is intended to show how to calculate the information using the `date`, remove all the columns *except* `date` and `births`.

```
library(mosaicData)
my.births <- Births78[c(1,2)]
head(my.births)
```

```
##         date births
## 1 1978-01-01   7701
## 2 1978-01-02   7527
## 3 1978-01-03   8825
## 4 1978-01-04   8859
## 5 1978-01-05   9043
## 6 1978-01-06   9208
```

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?

```
ggplot(my.births,
       aes(date,births))+
  geom_point(color = "skyblue2")+
  theme_classic()+
  theme(axis.title = element_text(colour="springgreen3"),
        axis.text = element_text(colour="springgreen4"),
        axis.line = element_line(colour="springgreen4"))
```



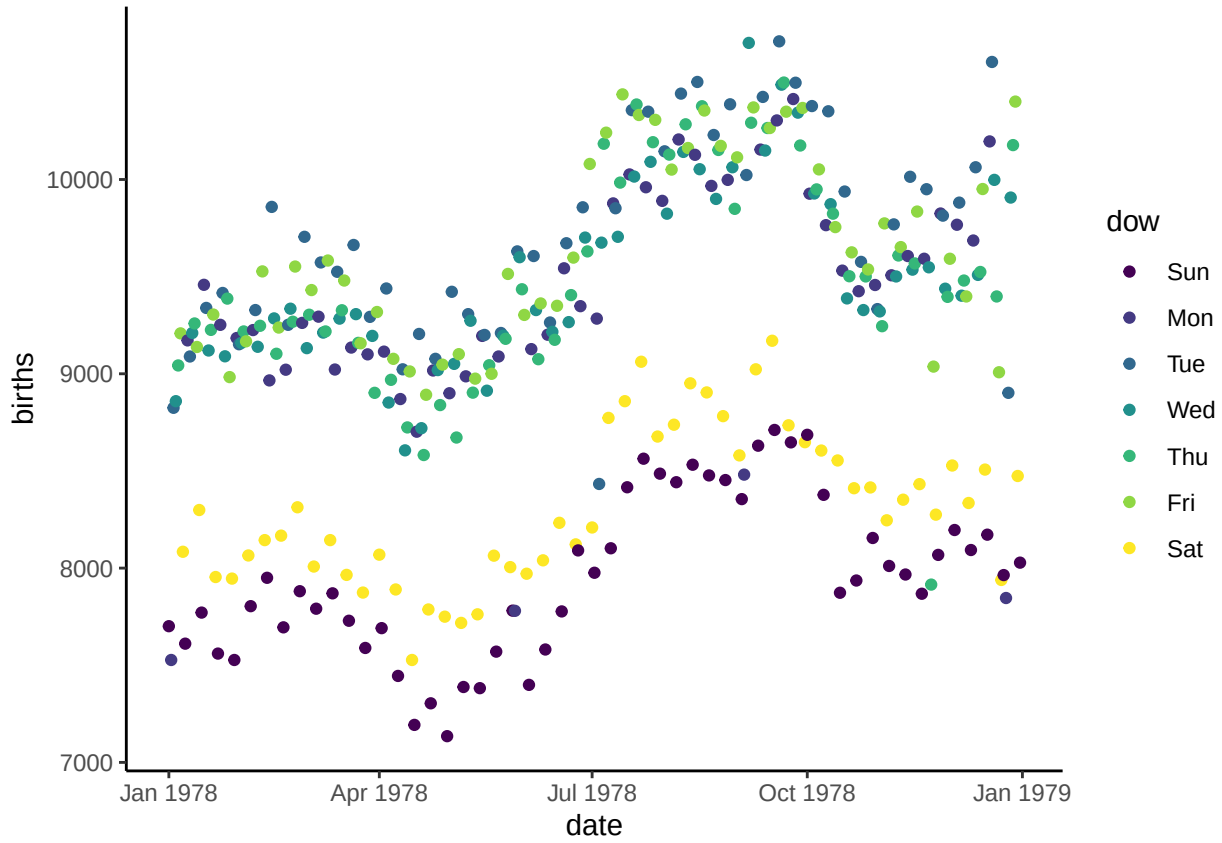
There are 2 distinct groupings of births, I think this is because people dont often have babies on weekends.

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.

```
my.births <- mutate(my.births, dow = wday(my.births$date, label = TRUE))
```

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

```
ggplot(my.births,
       aes(date,births, color = dow))+
  geom_point()+
  theme_classic()
```



My assumption was correct, the lowest days that people have babies are on Sunday and Saturday.