445-12

2023-10-30

Exercises

1. Convert the following to date or date/time objects.

   a) September 13, 2010.
   b) Sept 13, 2010.
   c) Sep 13, 2010.
   d) S 13, 2010. Comment on the month abbreviation needs. # only one month that starts with 'S'
   e) 07-Dec-1941.
   f) 1-5-1998. Comment on why you might be wrong. # month & day cannot be determined
   g) 21-5-1998. Comment on why you know you are correct. # 21 must be a day, there are only 12 months
   h) 2020-May-5 10:30 am
   i) 2020-May-5 10:30 am PDT (ex Seattle)
   j) 2020-May-5 10:30 am AST (ex Puerto Rico)

```
mdy('September 13, 2010', 'Sept 13, 2010', 'Sep 13, 2010', 'S 13, 2010') # a, b, c, d
```

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

```
dmy('07-Dec-1941', '1-5-1998', '21-5-1998') # e, f, g
```

```
## [1] "1941-12-07" "1998-05-01" "1998-05-21"
```

```
ymd_hm('2020-May-5 10:30 am') # h
```

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

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

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

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

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

2. Using just your date of birth (ex Sep 7, 1998) and today's date calculate the following *Write your code in a manner that the code will work on any date after you were born.*:

   a) Calculate the date of your 64th birthday.

```r
dob <- ymd('2002-06-14') # date of birth
dob + dyears(64) # 64 years after 'dob'
```

```
## [1] "2066-06-14"
```

```
b) Calculate your current age (in years). _Hint: Check your age is calculated correctly if your birthda
```

dob <- ymd(‘2023-10-30’) life <- as.period(dob %–% today()) life [1] “1d 0H 0M 0S”

dob <- ymd(‘2023-11-1’) life <- as.period(dob %–% today()) life [1] “-1d 0H 0M 0S”

```r
dob <- ymd('2002-06-14') # date of birth
life <- as.period(dob %--% today()) # period between birth and today
life
```

```
## [1] "21y 4m 17d 0H 0M 0S"
```

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

```r
bday22 <- dob + years(22) # 22 years since birth
bday22
```

```
## [1] "2024-06-14"
```

```
d) The number of _days_ until your next birthday.
```

```r
bday22 <- dob + years(22) # 22 years since birth
next_bday <- bday22 - today() # 22nd bday minus todays date
next_bday # days until next bday
```

```
## Time difference of 227 days
```

```
e) The number of _months_ and _days_ until your next birthday.
```

```r
bday22 <- dob + dyears(22) # 22 years since birth
next_bday <- bday22 - today() # 22nd bday minus todays date
```

```
## Warning: Incompatible methods ("-.POSIXt", "-.Date") for "-"
```

```r
next_bday # days until next bday
```

```
## [1] "2024-06-13 06:32:19 UTC"
```

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?

```
phone <- mdy_hm('May 8, 2015 3:00 pm', tz='US/Arizona') %>% # current AZ time
  with_tz(tzone='Pacific/Auckland') # convert to NZ time
phone
```

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

5. 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`.

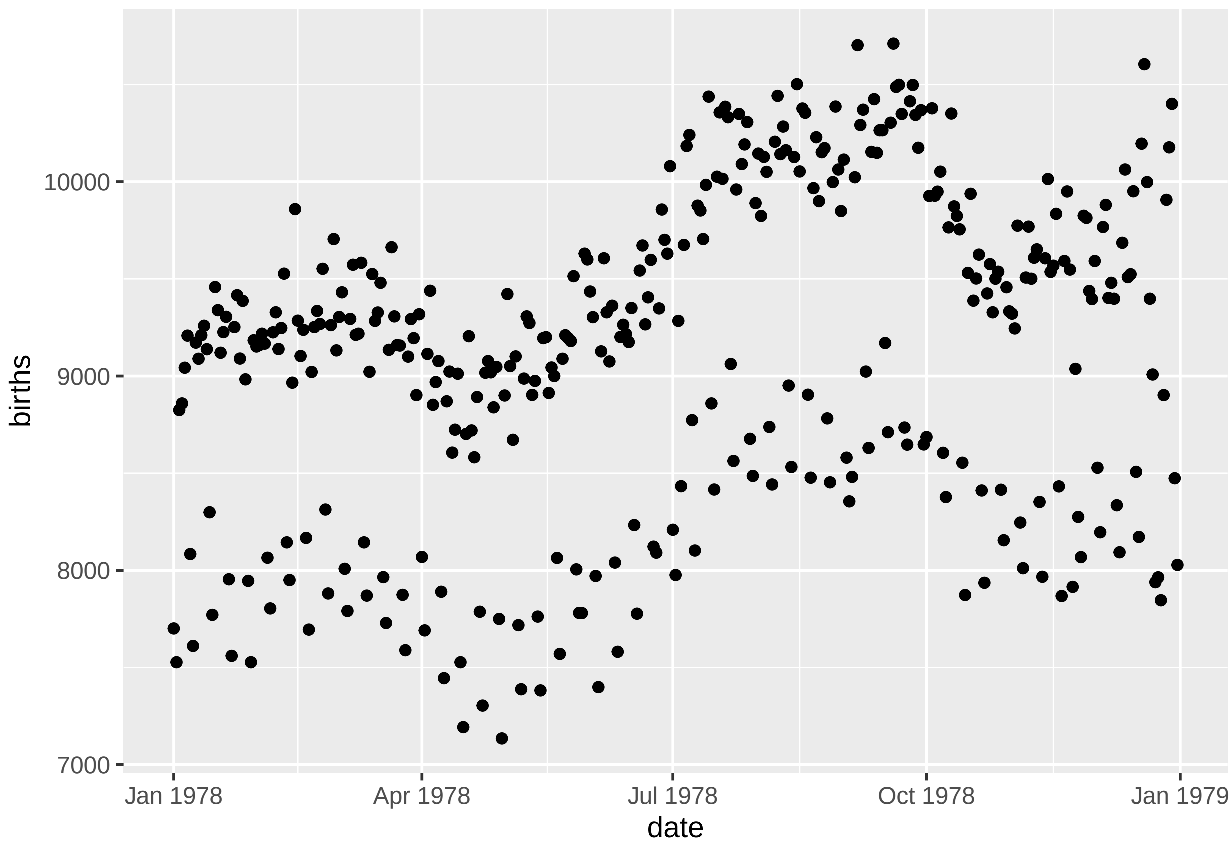
```
data('Births78') # load data
head(Births78 %>% select('date', 'births')) # only date & birth cols
```

```
##         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?
```

There is two different trend lines throughout the entire graph. This may be due to different days of the week or month when there are less births. Maybe people didn’t like to have births on Sunday for religious reasons.

```
ggplot(data=Births78, aes(x= date,y= births)) + geom_point() # date vs births
```

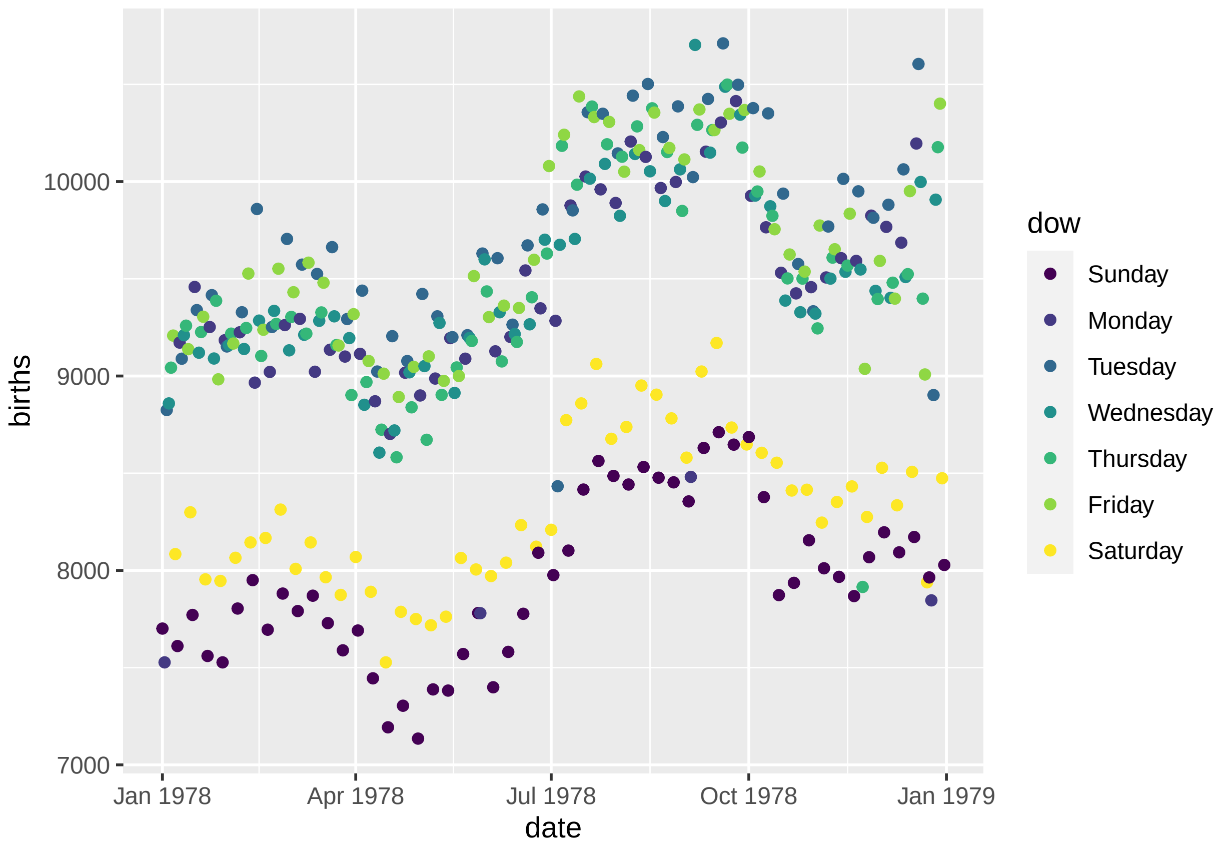


c. To test your assumption, we need to figure out the what day of the week each observation is. Use `dplyr::mutate` to add a new column named `dow` that is the day of the week (Monday, Tuesday, etc). This calculation will involve some function in the `lubridate` package and the `date` column.

```
Births78_dow <- Births78 %>% mutate(dow = wday(date, label=TRUE, abbr=FALSE))
# new 'dow' col by day
```

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

```
ggplot(data=Births78_dow, aes(x= date,y= births)) + geom_point(aes(color=dow))
```



```
# graph data on days of week by color
```