

STA_445_Assignment_6

Nicholas Larson

base::Sys.Date()

```
library(tidyverse)
library(lubridate)
```

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.

```
as.Date('Sept 13, 2010', format='%b %d, %Y')
```

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

This one doesn't work because R does not recognize 'Sept' as a month, only its three character form or its full form.

c. Sep 13, 2010.

```
as.Date('Sep 13, 2010', format='%b %d, %Y')
```

```
## [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
```

In order to be read correctly, the month needs to be either fully complete or 3 characters long, not 1, not 4.

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"
```

I cannot be sure whether the '1' or the '5' is the month or day as they both make sense.

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

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

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

I know this is the true answer because there is not 21 months in the year, so it must be the day.

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/Virgin")
```

```
## [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.

```
bday <- mdy('02042003')  
bday + years(64)
```

```
## [1] "2067-02-04"
```

b. Calculate your current age (in years).

```
bday <- mdy('02042003')  
exactbday <- as.period(bday %--% base::Sys.Date())  
year(exactbday)
```

```
## [1] 21
```

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

```
nbday <- bday + years(22)  
nbday
```

```
## [1] "2025-02-04"
```

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

```
dt1l <- as.period(base::Sys.Date() %--% nbday)  
as.period(dt1l, unit='days')
```

```
## [1] "313d 9H 0M 0S"
```

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

```
dt1l <- as.period(base::Sys.Date() %--% nbday)  
as.period(dt1l, unit='months')
```

```
## [1] "10m 9d 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?

```
mdy_h('May 8, 2015 3pm', tz='US/Arizona') %>% with_tz("NZ")
```

```
## [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.

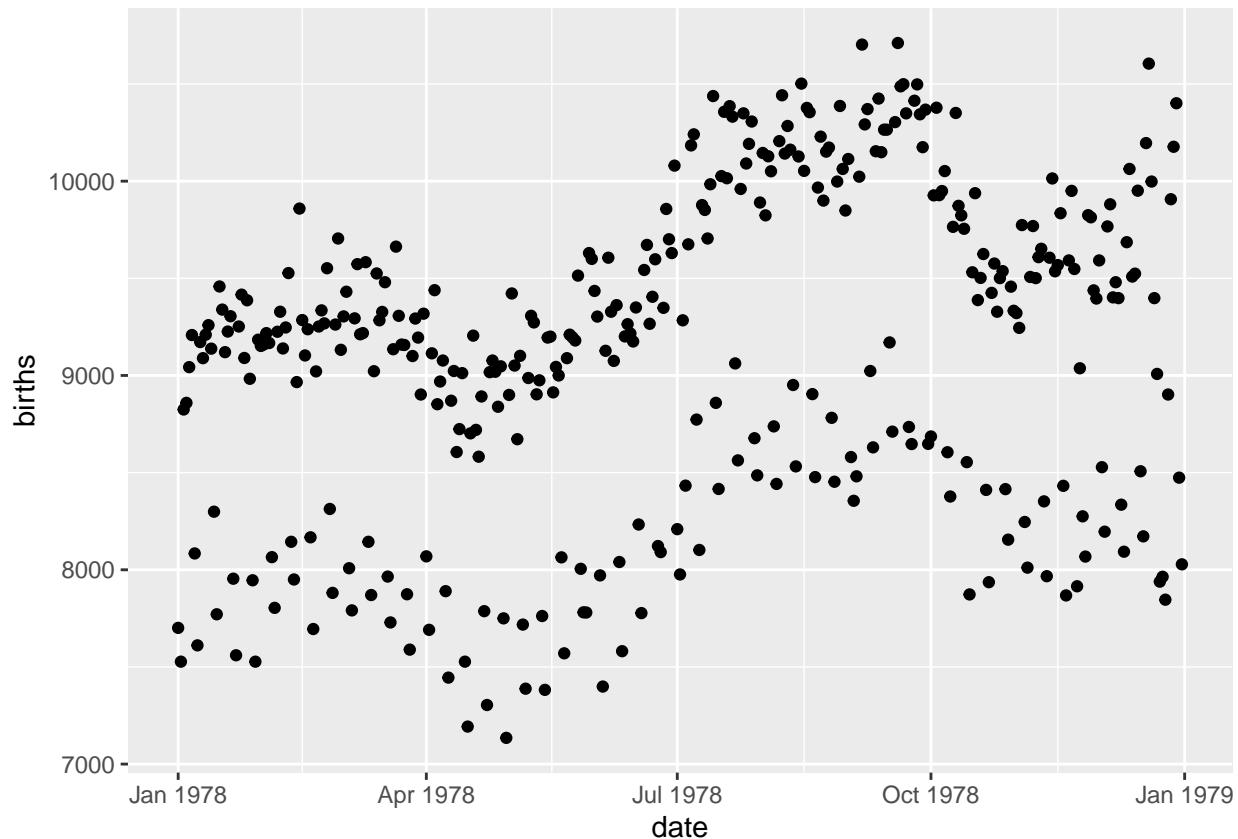
- a. Using the `mosaicData` package, load the data set `Births78` which records the number of children born on each day in the United States in 1978. Because this problem is intended to show how to calculate the information using the `date`, remove all the columns *except* `date` and `births`.

```
library(mosaicData)
data("Births78")
Births78 <- Births78 %>% select(date, births)
head(Births78)
```

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



I am not sure why that could be, but the next problem talks about days of the week, so that probably has something to do with it.

- 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 <- Births78 %>% mutate(
  dow = format(date, '%A')
)
head(Births78)
```

```
##      date births    dow
## 1 1978-01-01  7701  Sunday
## 2 1978-01-02  7527  Monday
## 3 1978-01-03  8825  Tuesday
## 4 1978-01-04  8859 Wednesday
## 5 1978-01-05  9043 Thursday
## 6 1978-01-06  9208  Friday
```

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

```
ggplot(data=Births78, aes(x=date, y=births)) + geom_point(aes(color=dow))
```

