MA615

2024-09-28

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

#a Your first exercise is to read in the data for all the years from 1985 to 2023. As discussedin class, you don't want to do this manually and will need to figure out a way to do itprogrammatically. We've given you a skeleton of how to do this for data for one year below.Your task is to adapt this to reading in multiple datasets from all the years in question. Thisexample code is meant to be a guide and if you think of a better way to read the data in, gofor it.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(readr)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
file_root <- "https://www.ndbc.noaa.gov/view_text_file.php?filename=44013h"
year <- "2023"
tail <- ".txt.gz&dir=data/historical/stdmet/"
path <- paste0(file_root, year, tail)
header <- read_lines(path, n_max = 1)
buoy <- read_table(path, skip = 2, col_names = c("YY", "MM", "DD", "hh", "mm", "WDIR", "WSPD","GST","WV
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   YY = col_double(),
##   MM = col_character(),
##   DD = col_character(),
##   hh = col_character(),
##   mm = col_character(),
##   WDIR = col_double(),
##   WSPD = col_double(),
##   GST = col_double(),
##   WVHT = col_double(),
##   DPD = col_double(),
##   APD = col_double(),
##   MWD = col_double(),
##   PRES = col_double(),
##   ATMP = col_double(),
##   WTMP = col_double(),
##   DEWP = col_double(),
##   VIS = col_double(),
##   TIDE = col_double()
## )
```

```
## Warning: 48050 parsing failures.
## row col   expected     actual
##   1  -- 18 columns 19 columns 'https://www.ndbc.noaa.gov/view_text_file.php?filename=44013h2023.txt.g
##   2  -- 18 columns 19 columns 'https://www.ndbc.noaa.gov/view_text_file.php?filename=44013h2023.txt.g
##   3  -- 18 columns 19 columns 'https://www.ndbc.noaa.gov/view_text_file.php?filename=44013h2023.txt.g
##   4  -- 18 columns 19 columns 'https://www.ndbc.noaa.gov/view_text_file.php?filename=44013h2023.txt.g
##   5  -- 18 columns 19 columns 'https://www.ndbc.noaa.gov/view_text_file.php?filename=44013h2023.txt.g
## ... ... .......... .......... ......................................................................
## See problems(...) for more details.
```

```r
buoy <- buoy %>%
  mutate(Year = as.integer(YY),
    Month = as.integer(MM),
    Day = as.integer(DD),
    Hour = as.integer(hh),
    Minute = as.integer(mm),
    Date = make_datetime(Year, Month, Day, Hour, Minute))
head(buoy)
```

```
## # A tibble: 6 x 24
##      YY MM    DD    hh    mm     WDIR  WSPD   GST  WVHT   DPD   APD   MWD  PRES
##   <dbl> <chr> <chr> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1  2023 01    01    00    00      999   4.3   5    99    99    99      999 1011.
## 2  2023 01    01    00    10      999   4.5   5.4  99    99    99      999 1011.
## 3  2023 01    01    00    20      999   4.2   4.8  99    99    99      999 1010.
## 4  2023 01    01    00    30      999   4.2   4.8  99    99    99      999 1010
## 5  2023 01    01    00    40      999   3.9   4.3   0.41  9.09  3.43   112 1010.
## 6  2023 01    01    00    50      999   3.2   4.1   0.46 10     3.41    93 1010.
## # i 11 more variables: ATMP <dbl>, WTMP <dbl>, DEWP <dbl>, VIS <dbl>,
## #   TIDE <dbl>, Year <int>, Month <int>, Day <int>, Hour <int>, Minute <int>,
## #   Date <dttm>
```

#b

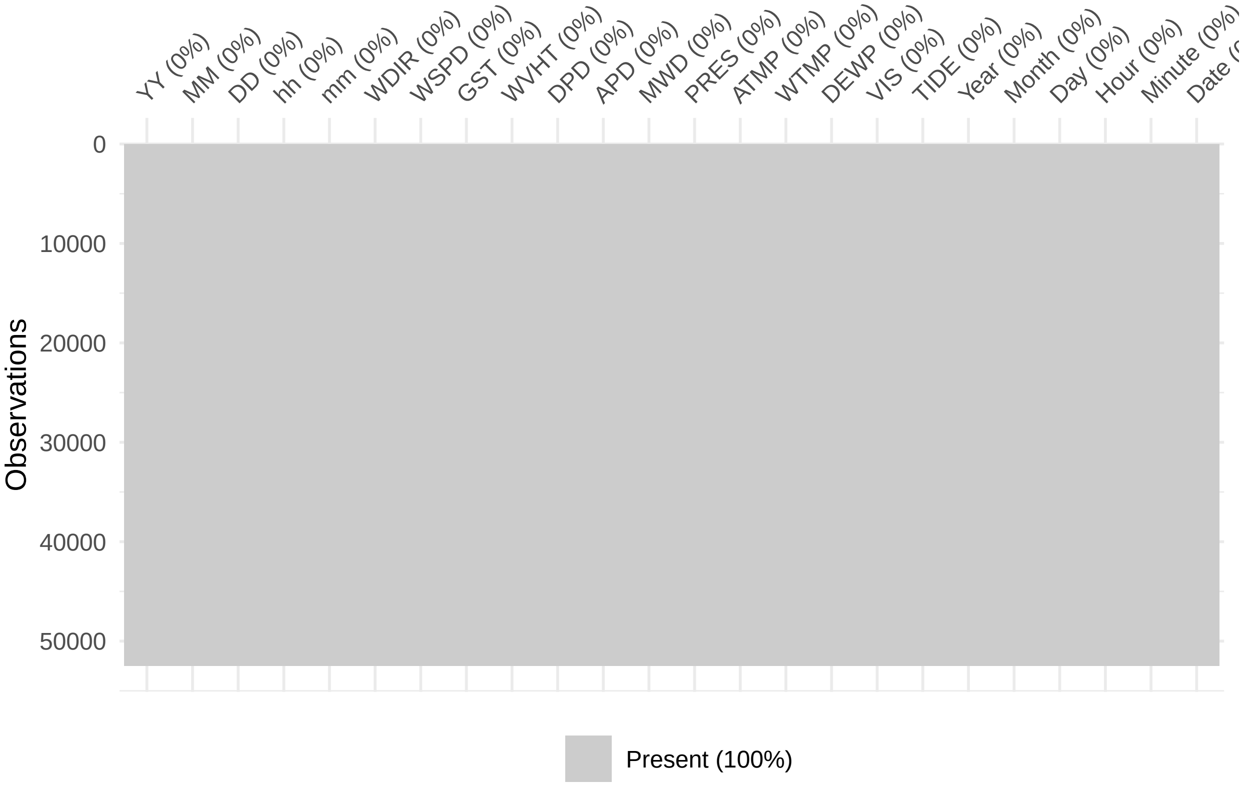
```r
library(dplyr)
buoy1 <- buoy %>%
  mutate(across(where(is.numeric), ~na_if(., 999)))
head(buoy1)
```

```
## # A tibble: 6 x 24
##      YY MM    DD    hh    mm     WDIR  WSPD   GST  WVHT   DPD   APD   MWD  PRES
##   <dbl> <chr> <chr> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1  2023 01    01    00    00       NA   4.3   5    99     99    99       NA 1011.
## 2  2023 01    01    00    10       NA   4.5   5.4 99     99    99       NA 1011.
## 3  2023 01    01    00    20       NA   4.2   4.8 99     99    99       NA 1010.
## 4  2023 01    01    00    30       NA   4.2   4.8 99     99    99       NA 1010
## 5  2023 01    01    00    40       NA   3.9   4.3  0.41  9.09  3.43   112 1010.
## 6  2023 01    01    00    50       NA   3.2   4.1  0.46 10     3.41    93 1010.
## # i 11 more variables: ATMP <dbl>, WTMP <dbl>, DEWP <dbl>, VIS <dbl>,
## #   TIDE <dbl>, Year <int>, Month <int>, Day <int>, Hour <int>, Minute <int>,
## #   Date <dttm>
```

```
na_count <- sapply(buoy, function(x) sum(is.na(x)))
print(na_count)
```

```
##      YY      MM      DD      hh      mm    WDIR    WSPD     GST    WVHT     DPD     APD
##       0       0       0       0       0       0       0       0       0       0       0
##     MWD    PRES    ATMP    WTMP    DEWP     VIS    TIDE    Year   Month     Day    Hour
##       0       0       0       0       0       0       0       0       0       0       0
## Minute    Date
##       0       0
```

```
library(ggplot2)
library(naniar)
vis_miss(buoy, warn_large_data = FALSE)
```

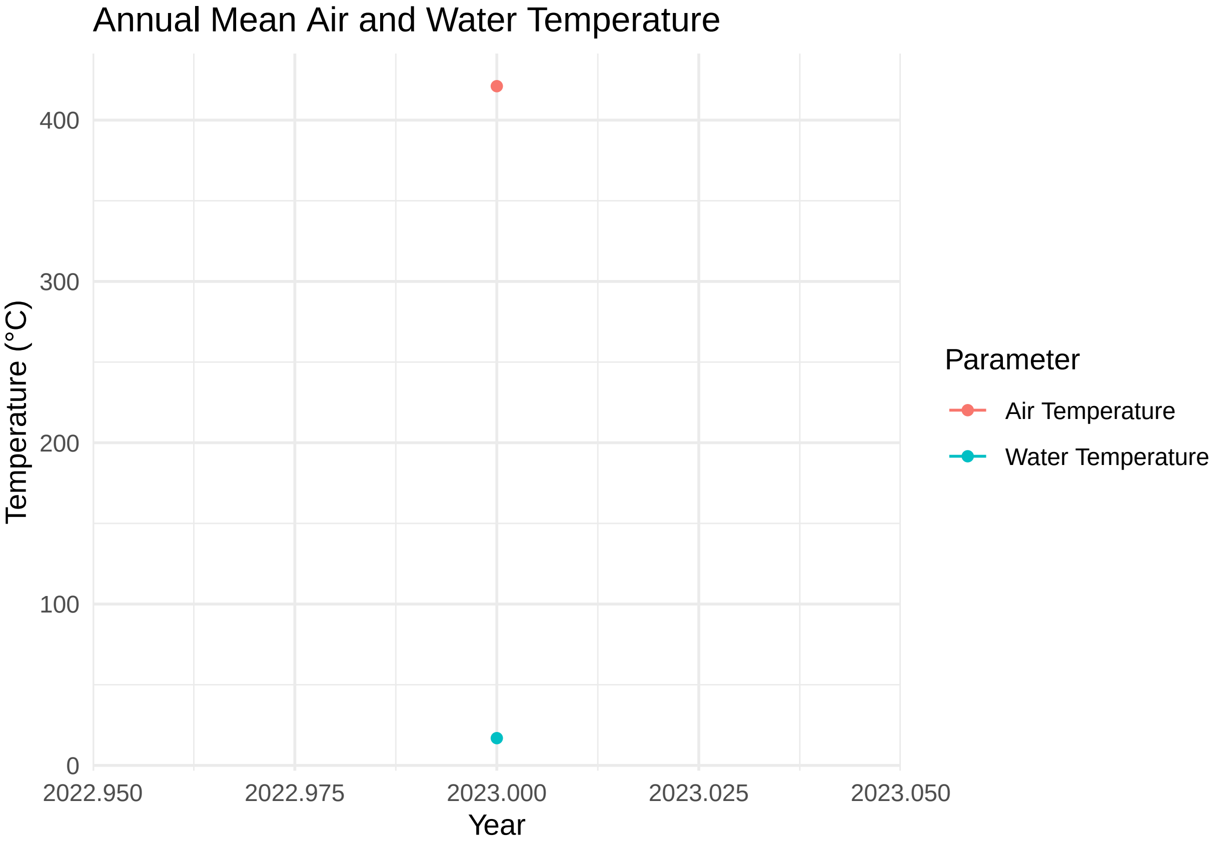


```
#As a bonus part, it indicates that the lost data pattern may be related to an external event, such as
```

#c

```
library(ggplot2)
library(dplyr)
annual_data <- buoy %>%
  group_by(Year = year(Date)) %>%
  summarize(
    Mean_ATMP = mean(ATMP, na.rm = TRUE),
    Mean_WTMP = mean(WTMP, na.rm = TRUE),
    Mean_WSPD = mean(WSPD, na.rm = TRUE)
  )
ggplot(annual_data, aes(x = Year)) +
  geom_point(aes(y = Mean_ATMP, color = "Air Temperature")) +
  geom_point(aes(y = Mean_WTMP, color = "Water Temperature")) +
  geom_line(aes(y = Mean_ATMP, color = "Air Temperature")) +
  geom_line(aes(y = Mean_WTMP, color = "Water Temperature")) +
  labs(title = "Annual Mean Air and Water Temperature",
       y = "Temperature (°C)",
       color = "Parameter") +
  theme_minimal()
```

```
## `geom_line()`: Each group consists of only one observation.
## i Do you need to adjust the group aesthetic?
## `geom_line()`: Each group consists of only one observation.
## i Do you need to adjust the group aesthetic?
```



#d

```
library(dplyr)
library(readr)
```

```r
library(ggplot2)
library(lubridate)

rainfall_data <- read_csv("Rainfall.csv")
```

```
## Rows: 31714 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): STATION, STATION_NAME, Measurement Flag
## dbl  (1): HPCP
## lgl  (1): Quality Flag
## dttm (1): DATE
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
rainfall_data <- rainfall_data %>%
  mutate(DATE = ymd(DATE))
```

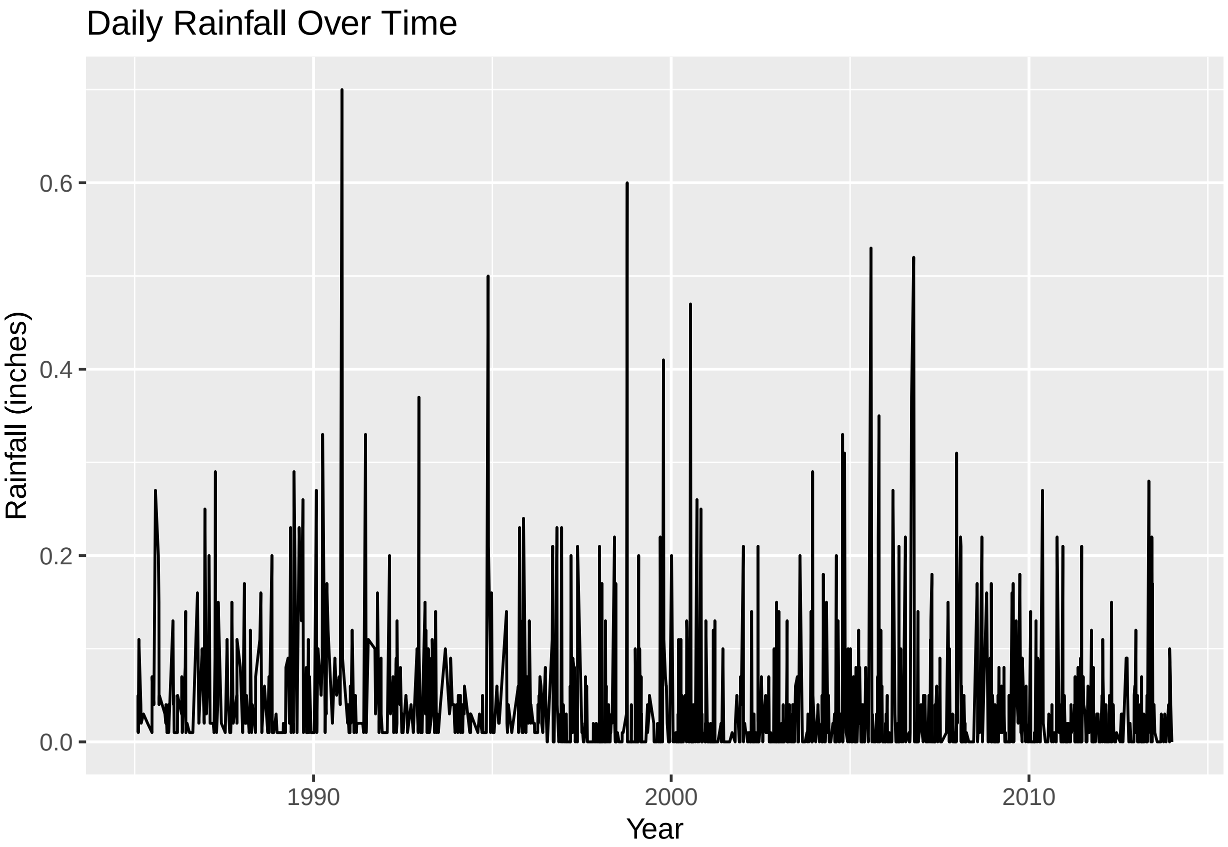
```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `DATE = ymd(DATE)`.
## Caused by warning:
## !  30430 failed to parse.
```

```r
rainfall_data <- rainfall_data %>%
  filter(year(DATE) >= 1985, year(DATE) <= 2013)
rainfall_stats <- rainfall_data %>%
  summarize(
    Total_Days = n(),
    Rain_Days = sum(HPCP > 0, na.rm = TRUE),
    No_Rain_Days = sum(HPCP == 0, na.rm = TRUE),
    Avg_Rainfall = mean(HPCP, na.rm = TRUE),
    Max_Rainfall = max(HPCP, na.rm = TRUE)
  )

print(rainfall_stats)
```

```
## # A tibble: 1 x 5
##   Total_Days Rain_Days No_Rain_Days Avg_Rainfall Max_Rainfall
##        <int>     <int>        <int>        <dbl>        <dbl>
## 1       1284       811          473       0.0381          0.7
```

```r
ggplot(rainfall_data, aes(x = DATE, y = HPCP)) +
  geom_line() +
  labs(title = "Daily Rainfall Over Time", x = "Year", y = "Rainfall (inches)")
```


Daily Rainfall Over Time
Rainfall (inches)
0.6
0.4
0.2
0.0
1990
2000
2010
Year