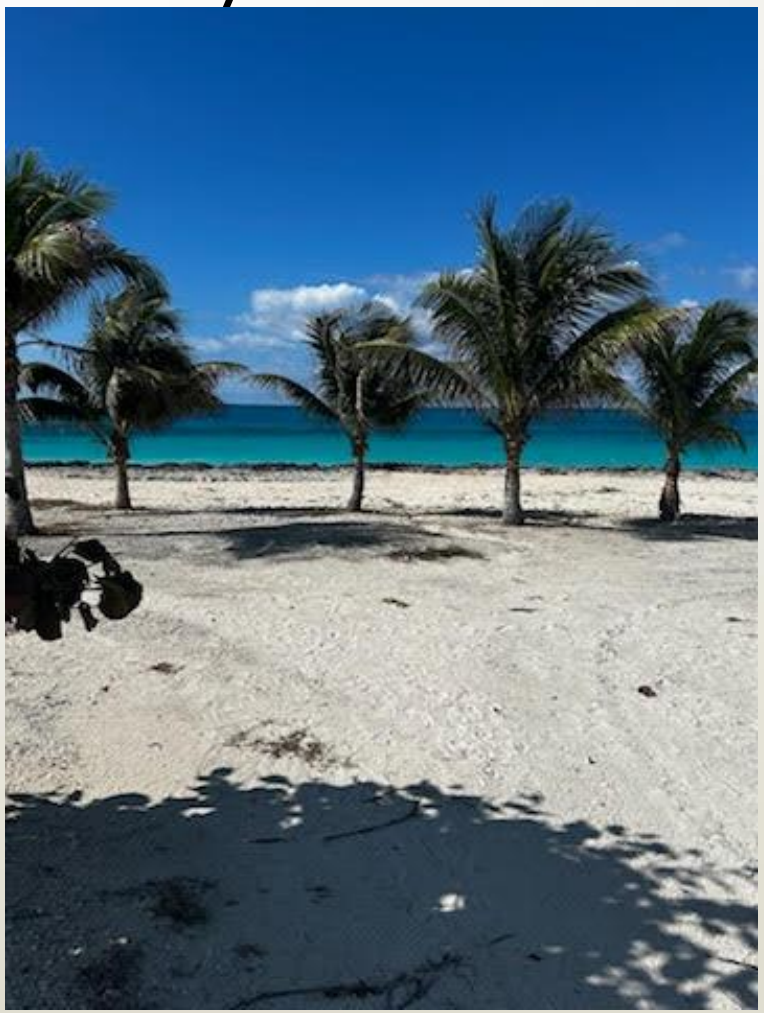
"One of the lessons that I grew up with was to always stay true to yourself and never let what somebody else says distract you from your goals." - Michelle Obama



# Tibble and Tsibble

Dr. Susan Simmons

Institute for Advanced Analytics

# Time Series Data

- A time series is an ordered sequence of observations.
  - Ordering is typically through ***equally spaced*** time intervals.
  - Possibly through space as well.
- Used in a variety of fields:
  - Agriculture: Crop Production
  - Economics: Stock Prices
  - Engineering: Electric Signals
  - Meteorology: Wind Speeds
  - Social Sciences: Crime Rates

# Time Series Data

- We will begin our time series discussions with univariate time series (only one time series…one variable, we will call it Y).
- Multivariate time series will be in Fall 2.

# Time Series Data

- We will begin our time series discussions with univariate time series (only one time series…one variable, we will call it Y).
- Multivariate time series will be in Fall 2.

| Date | Y |
|---|---|
| January 2000 | 23 |
| February 2000 | 18 |
| March 2000 | 20 |
| April 2000 | 25 |
| May 2000 | 21 |

# Time Series Data

- We will begin our time series discussions with univariate time series (only one time series…one variable, we will call it Y).
- Multivariate time series will be in Fall 2.

| Date | Y |
|---|---|
| January 2000 | 23 |
| February 2000 | 18 |
| March 2000 | 20 |
| April 2000 | 25 |
| May 2000 | 21 |

$Y_1$

# Time Series Data

- We will begin our time series discussions with univariate time series (only one time series…one variable, we will call it Y).
- Multivariate time series will be in Fall 2.

| Date | Y |
|---|---|
| January 2000 | 23 |
| February 2000 | 18 |
| March 2000 | 20 |
| April 2000 | 25 |
| May 2000 | 21 |

$Y_2$

# Time Series Data

- We will begin our time series discussions with univariate time series (only one time series…one variable, we will call it Y).
- Multivariate time series will be in Fall 2.

| Date | Y |
|---|---|
| January 2000 | 23 |
| February 2000 | 18 |
| March 2000 | 20 |
| April 2000 | 25 |
| May 2000 | 21 |

$Y_3$

# Time Series Data

- We will begin our time series discussions with univariate time series (only one time series…one variable, we will call it Y).
- Multivariate time series will be in Fall 2.

| Date | Y |
|---|---|
| January 2000 | 23 |
| February 2000 | 18 |
| March 2000 | 20 |
| April 2000 | 25 |
| May 2000 | 21 |

$Y_3$

$Y_t$

***CAREFUL: Since we are assuming equally spaced, you will need to take care of missing values !!***

# Tibble and Tsibble

# Careful on calling functions

- Names of functions are similar across different packages!
- Need to use package name with function (otherwise can get error)
  - stats::filter()
  - dplyr::filter()

# Tibble

- Tibble (part of dplyr) is designed for large data frames (for REALLY large data frames, look into data.table)….similar to data.frame
- Viewing a data set
  - View(name of data) - this will open an interactive view
  - glimpse(name of data)
  - name of data
- To pipe in tibble, you can either use |>  or %>%

# Example

- Read in ebay data set from Time series github:

library(dplyr)

library(readr)

```
ebay <-
read_csv('https://raw.githubusercontent.com/sjsimmo2/TimeSeries/master/ebay9899.csv')
```

Note: if you have a data frame already in R, you can convert it to a tibble by simply as_tibble(name of data)

# 1999 – Ebay stock

- Say we want to look at those stocks in the year 1999
- We can easily filter by date:

```
ebay_1999 <- ebay |>
    filter(DATE > "1998-12-31" & DATE < "2000-01-01")
View(ebay_1999)
```

Intitled1* | ebay_1999 | temp | weather | temp2 | ebay | R In

Filter

| | OpeningPrice | DailyHigh | DailyLow | ClosingPrice | Volume | DATE |
|---|---|---|---|---|---|---|
| 1 | NA | NA | NA | NA | NA | 1999-01-01 |
| 2 | 78.7917 | 83.6667 | 77.3333 | 80.0000 | 783700 | 1999-01-04 |
| 3 | 76.0208 | 77.5000 | 73.2500 | 74.6667 | 1319900 | 1999-01-05 |
| 4 | 78.0000 | 95.0000 | 77.6667 | 94.3333 | 1910100 | 1999-01-06 |
| 5 | 90.7083 | 105.0000 | 90.0417 | 99.3333 | 2077200 | 1999-01-07 |
| 6 | 105.0000 | 107.0000 | 95.6875 | 97.4583 | 1568700 | 1999-01-08 |
| 7 | 101.3540 | 101.6670 | 92.6667 | 97.1667 | 771300 | 1999-01-11 |
| 8 | 93.3333 | 97.5000 | 80.0000 | 80.1667 | 1109400 | 1999-01-12 |

ntitled1* | ebay_1999 | temp | weather | temp2 | ebay | R In

Filter

| | OpeningPrice | DailyHigh | DailyLow | ClosingPrice | Volume | DATE |
|---|---|---|---|---|---|---|
| 240 | 160.2500 | 167.5000 | 159.8750 | 167.0625 | 2591400 | 1999-12-02 |
| 241 | 169.8750 | 181.3750 | 169.5000 | 178.7500 | 5176400 | 1999-12-03 |
| 242 | 177.7500 | 181.8750 | 173.0000 | 176.5000 | 1976700 | 1999-12-06 |
| 243 | 176.6875 | 178.2500 | 162.3750 | 163.7500 | 5327500 | 1999-12-07 |
| 244 | 164.0000 | 170.1250 | 160.1875 | 162.4375 | 3882700 | 1999-12-08 |
| 245 | 166.2500 | 170.0000 | 157.8750 | 160.6250 | 3713300 | 1999-12-09 |
| 246 | 166.5000 | 167.2500 | 160.1250 | 164.4375 | 3100100 | 1999-12-10 |
| 247 | 164.8125 | 170.0625 | 160.5000 | 161.5000 | 2362100 | 1999-12-13 |

Showing 229 to 247 of 247 entries, 6 total columns

# Tsibble

# Tsibble objects

- The package we will be using in Time Series 1 is the fable package (Hyndman)
- To utilize most functionality in this package, you need to turn your data set into a tsibble (think "time series…ts")
- When converting to a tsibble, you have to tell it the information regarding time…is this monthly data, hourly data, yearly data, etc….can do others
- The "time" information is provide by specifying "index"
- If you have multiple time series in one file, you can identify the different ones with "key"

# Tsibble continued

- When you view your tsibble object, it will show you what it thinks the time is ( [1M] = Monthly data, [1Q] = Quarterly data, [1D] = Daily data, etc).
- To demonstrate, we will take a look at two data sets (within the datasets library). They are as follows:
  - airquality - daily data set - Daily air quality measurements in New York for May to September 1973.
  - austres – quarterly data set - Numbers (in thousands) of Australian residents measured quarterly from March 1971 to March 1994

# Airquality

```
> library(datasets)
> library(fpp3)
── Attaching packages ──────────────────────────────────── fpp3 0.5 ──
✓ tibble     3.2.1     ✓ tsibble    1.1.3
✓ dplyr      1.1.2     ✓ tsibbledata 0.4.1
✓ tidyr      1.3.0     ✓ feasts     0.3.1
✓ lubridate  1.9.2     ✓ fable      0.3.4
✓ ggplot2    3.4.2     ✓ fabletools  0.3.3
── Conflicts ────────────────────────────────────── fpp3_conflicts ──
✖ lubridate::date()    masks base::date()
✖ dplyr::filter()      masks stats::filter()
✖ tsibble::intersect() masks base::intersect()
✖ tsibble::interval()  masks lubridate::interval()
✖ dplyr::lag()         masks stats::lag()
✖ tsibble::setdiff()   masks base::setdiff()
✖ tsibble::union()     masks base::union()
```

# Airquality

```
airquality
```

| | Ozone | Solar.R | Wind | Temp | Month | Day |
|---|---|---|---|---|---|---|
| 1 | 41 | 190 | 7.4 | 67 | 5 | 1 |
| 2 | 36 | 118 | 8.0 | 72 | 5 | 2 |
| 3 | 12 | 149 | 12.6 | 74 | 5 | 3 |
| 4 | 18 | 313 | 11.5 | 62 | 5 | 4 |
| 5 | NA | NA | 14.3 | 56 | 5 | 5 |
| 6 | 28 | NA | 14.9 | 66 | 5 | 6 |
| 7 | 23 | 299 | 8.6 | 65 | 5 | 7 |

```
airquality2 <- airquality %>% mutate(date= lubridate::make_date(1973,Month,Day))
```

# Airquality as tsibble

```
air_ts <- as_tsibble(airquality2,index=date)
air_ts
# A tsibble: 153 x 7 [1D]
   Ozone Solar.R  Wind  Temp Month   Day date
   <int>   <int> <dbl> <int> <int> <int> <date>
 1    41     190   7.4    67     5     1 1973-05-01
 2    36     118   8      72     5     2 1973-05-02
 3    12     149  12.6    74     5     3 1973-05-03
```

# Austres

```
> austres
         Qtr1     Qtr2     Qtr3     Qtr4
1971          13067.3 13130.5 13198.4
1972 13254.2 13303.7 13353.9 13409.3
1973 13459.2 13504.5 13552.6 13614.3
1974 13669.5 13722.6 13772.1 13832.0
```

```
is.ts(austres)
[1] TRUE
```

# Austres

```
> austres
         Qtr1     Qtr2     Qtr3     Qtr4
1971          13067.3 13130.5 13198.4
1972 13254.2 13303.7 13353.9 13409.3
1973 13459.2 13504.5 13552.6 13614.3
1974 13669.5 13722.6 13772.1 13832.0
```

```
is.ts(austres)
[1] TRUE
```

```
austres_ts <- austres %>% as_tsibble()
> head(austres_ts)
# A tsibble: 6 x 2 [1Q]
     index  value
     <qtr>  <dbl>
1 1971 Q2 13067.
2 1971 Q3 13130.
3 1971 Q4 13198.
4 1972 Q1 13254.
5 1972 Q2 13304.
6 1972 Q3 13354.
```

# Tibble to Tsibble

- If a data set is a tibble with a date variable, it is easy to convert to tsibble
- Using the prison example from Hyndman's book

```
prison <- readr::read_csv("https://OTexts.com/fpp3/extrafiles/prison_population.csv")
```

# Prison data set

Prison

```
# A tibble: 3,072 × 6
   Date       State Gender Legal     Indigenous Count
   <date>     <chr> <chr>  <chr>     <chr>      <dbl>
 1 2005-03-01 ACT   Female Remanded  ATSI           0
 2 2005-03-01 ACT   Female Remanded  Non-ATSI       2
 3 2005-03-01 ACT   Female Sentenc… ATSI           0
 4 2005-03-01 ACT   Female Sentenc… Non-ATSI       5
```

# Manipulate tibble and covert to tsibble

```
prison <- prison |>
  mutate(Quarter = yearquarter(Date)) |>
  select(-Date) |>
  as_tsibble(key = c(State, Gender, Legal, Indigenous),
             index = Quarter)

prison
# A tsibble: 3,072 x 6 [1Q]
# Key:       State, Gender, Legal, Indigenous [64]
   State Gender Legal    Indigenous Count Quarter
   <chr> <chr>  <chr>    <chr>      <dbl>   <qtr>
 1 ACT   Female Remanded ATSI           0 2005 Q1
 2 ACT   Female Remanded ATSI           1 2005 Q2
 3 ACT   Female Remanded ATSI           0 2005 Q3
 4 ACT   Female Remanded ATSI           0 2005 Q4
 5 ACT   Female Remanded ATSI           1 2006 Q1
```

64 different time series!! 8 states, 2 genders, 2 legal statuses and 2 indigenous statuses

# Missing values

- Since we need to assume equally spaced data points, we should know if our data has gaps in it (implicit missing values)
  - has_gaps() checks for implicit missingness
  - count_gaps() summarizes time ranges
  - scan_gaps() gives a detailed report
  - fill_gaps() turns them into explicit missingness

# PM 2.5

- Data set pm2_5Wake2023.csv was downloaded from the EPA
- Sensor at Triple Oak station recording particulate matter 2.5 (PM2.5) which is a mixture of microscopic particles or droplets in that are 2.5 micrometers or less in diameter (pollutants)

```
library(readr)
pm25 <- read_csv('Q:\\My Drive\\Fall 2017 - Time Series\\DataR\\pm2_5WAke2023.csv')
pm25.1<- pm25 %>% mutate(date2 = mdy(Date)) %>% as_tsibble(index=date2)
```

```
> pm25.1
# A tsibble: 350 x 23 [1D]
Date Source `Site ID` POC Daily Mean PM2.5 Con…¹ Units `Daily AQI Value` `Local Site Name` `Daily Obs Count`
<chr> <chr> <dbl> <dbl> <dbl> <chr> <dbl> <chr> <dbl>
1 1/1/… AQS 371830021 3 5.5 ug/m… 31 Triple Oak 1
2 1/2/… AQS 371830021 3 7.1 ug/m… 39 Triple Oak 1
3 1/3/… AQS 371830021 3 10.3 ug/m… 53 Triple Oak 1
4 1/4/… AQS 371830021 3 4.8 ug/m… 27 Triple Oak 1
5 1/5/… AQS 371830021 3 4.2 ug/m… 23 Triple Oak 1
6 1/6/… AQS 371830021 3 3.5 ug/m… 19 Triple Oak 1
7 1/7/… AQS 371830021 3 4.7 ug/m… 26 Triple Oak 1
```

```
> pm25.1
# A tsibble: 350 x 23 [1D]
Date Source `Site ID` POC Daily Mean PM2.5 Con…¹ Units `Daily AQI Value` `Local Site Name` `Daily Obs Count`
<chr> <chr> <dbl> <dbl> <dbl> <chr> <dbl> <chr> <dbl>
1 1/1/… AQS 371830021 3 5.5 ug/m… 31 Triple Oak 1
2 1/2/… AQS 371830021 3 7.1 ug/m… 39 Triple Oak 1
3 1/3/… AQS 371830021 3 10.3 ug/m… 53 Triple Oak 1
4 1/4/… AQS 371830021 3 4.8 ug/m… 27 Triple Oak 1
5 1/5/… AQS 371830021 3 4.2 ug/m… 23 Triple Oak 1
6 1/6/… AQS 371830021 3 3.5 ug/m… 19 Triple Oak 1
7 1/7/… AQS 371830021 3 4.7 ug/m… 26 Triple Oak 1
```

```
> has_gaps(pm25.1)
# A tibble: 1 x 1
.gaps
<lgl>
1 TRUE
```

```
> pm25.1
# A tsibble: 350 x 23 [1D]
Date Source `Site ID` POC Daily Mean PM2.5 Con…¹ Units `Daily AQI Value` `Local Site Name` `Daily Obs Count`
<chr> <chr> <dbl> <dbl> <dbl> <chr> <dbl> <chr> <dbl>
1 1/1/… AQS 371830021 3 5.5 ug/m… 31 Triple Oak 1
2 1/2/… AQS 371830021 3 7.1 ug/m… 39 Triple Oak 1
3 1/3/… AQS 371830021 3 10.3 ug/m… 53 Triple Oak 1
4 1/4/… AQS 371830021 3 4.8 ug/m… 27 Triple Oak 1
5 1/5/… AQS 371830021 3 4.2 ug/m… 23 Triple Oak 1
6 1/6/… AQS 371830021 3 3.5 ug/m… 19 Triple Oak 1
7 1/7/… AQS 371830021 3 4.7 ug/m… 26 Triple Oak 1
```

```
> has_gaps(pm25.1)
# A tibble: 1 x 1
.gaps
<lgl>
1 TRUE
```

```
> scan_gaps(pm25.1)
# A tsibble: 15 x 1 [1D]
date2
<date>
1 2023-04-18
2 2023-04-19
3 2023-04-20
4 2023-04-21
5 2023-04-22
6 2023-04-23
7 2023-04-24
8 2023-04-25
9 2023-04-26
10 2023-04-27
11 2023-08-08
12 2023-10-23
13 2023-10-24
14 2023-10-25
15 2023-10-26
```

```
pm25.1
# A tsibble: 350 x 23 [1D]
Date Source `Site ID` POC Daily Mean PM2.5 Con…¹ Units `Daily AQI Value` `Local Site Name` `Daily Obs Count`
<chr> <chr> <dbl> <dbl> <dbl> <chr> <dbl> <chr> <dbl>
1 1/1/… AQS 371830021 3 5.5 ug/m… 31 Triple Oak 1
2 1/2/… AQS 371830021 3 7.1 ug/m… 39 Triple Oak 1
3 1/3/… AQS 371830021 3 10.3 ug/m… 53 Triple Oak 1
4 1/4/… AQS 371830021 3 4.8 ug/m… 27 Triple Oak 1
5 1/5/… AQS 371830021 3 4.2 ug/m… 23 Triple Oak 1
6 1/6/… AQS 371830021 3 3.5 ug/m… 19 Triple Oak 1
7 1/7/… AQS 371830021 3 4.7 ug/m… 26 Triple Oak 1
```

```
has_gaps(pm25.1)
# A tibble: 1 x 1
.gaps
<lgl>
1 TRUE
```

```
count_gaps(pm25.1)
# A tibble: 3 x 3
.from .to .n <date> <date> <int>
1 2023-04-18 2023-04-27 10
2 2023-08-08 2023-08-08 1
3 2023-10-23 2023-10-26 4
```

```
scan_gaps(pm25.1)
# A tsibble: 15 x 1 [1D]
date2
<date>
1 2023-04-18
2 2023-04-19
3 2023-04-20
4 2023-04-21
5 2023-04-22
6 2023-04-23
7 2023-04-24
8 2023-04-25
9 2023-04-26
10 2023-04-27
11 2023-08-08
12 2023-10-23
13 2023-10-24
14 2023-10-25
15 2023-10-26
```

```
> pm25.1
# A tsibble: 350 x 23 [1D]
Date Source `Site ID` POC Daily Mean PM2.5 Con…¹ Units `Daily AQI Value` `Local Site Name` `Daily Obs Count`
<chr> <chr> <dbl> <dbl> <dbl> <chr> <dbl> <chr> <dbl>
1 1/1/… AQS 371830021 3 5.5 ug/m… 31 Triple Oak 1
2 1/2/… AQS 371830021 3 7.1 ug/m… 39 Triple Oak 1
3 1/3/… AQS 371830021 3 10.3 ug/m… 53 Triple Oak 1
4 1/4/… AQS 371830021 3 4.8 ug/m… 27 Triple Oak 1
5 1/5/… AQS 371830021 3 4.2 ug/m… 23 Triple Oak 1
6 1/6/… AQS 371830021 3 3.5 ug/m… 19 Triple Oak 1
7 1/7/… AQS 371830021 3 4.7 ug/m… 26 Triple Oak 1
```

```
> has_gaps(pm25.1)
# A tibble: 1 x 1
.gaps
<lgl>
1 TRUE
```

```
> count_gaps(pm25.1)
# A tibble: 3 x 3
.from .to .n <date> <date> <int>
1 2023-04-18 2023-04-27 10
2 2023-08-08 2023-08-08 1
3 2023-10-23 2023-10-26 4
```

```
> scan_gaps(pm25.1)
# A tsibble: 15 x 1 [1D]
date2
<date>
1 2023-04-18
2 2023-04-19
3 2023-04-20
4 2023-04-21
5 2023-04-22
6 2023-04-23
7 2023-04-24
8 2023-04-25
9 2023-04-26
10 2023-04-27
11 2023-08-08
12 2023-10-23
13 2023-10-24
14 2023-10-25
15 2023-10-26
```

```
> pm25.2<-pm25.1 %>% fill_gaps()
```

Good references:
R for Data Science book (online 2nd edition: Chapter 1 Data visualization)

Hyndman's Forecasting: Principles and Practice (online Version 3: Chapter 2.1 tsibble objects)

# Happy tsibbling!


Getty images: Lisa Gagney