

# Exploratory time series analysis using R

## 1. Wrangling time series data



# Outline

- 1 Introduction
- 2 Time series data and tsibbles
- 3 Example: Australian prison population
- 4 Example: Australian pharmaceutical sales
- 5 Lab Session 1

# Outline

- 1 Introduction
- 2 Time series data and tsibbles
- 3 Example: Australian prison population
- 4 Example: Australian pharmaceutical sales
- 5 Lab Session 1

# Instructor



## Rob J Hyndman

🏠 [robjhyndman.com](https://robjhyndman.com)

@robjhyndman@aus.social

🐙 @robjhyndman

✉️ [rob.hyndman@monash.edu](mailto:rob.hyndman@monash.edu)

# Instructor



## Rob J Hyndman

🏠 [robjhyndman.com](http://robjhyndman.com)

@robjhyndman@aus.social

🐙 @robjhyndman

✉️ [rob.hyndman@monash.edu](mailto:rob.hyndman@monash.edu)

## Helpers

- Nuwani Palihawadana
- Swen Kuh

## Key reference

**Hyndman, R. J. & Athanasopoulos, G. (2021) *Forecasting: principles and practice*, 3rd ed.**

## Key reference

**Hyndman, R. J. & Athanasopoulos, G. (2021) *Forecasting: principles and practice*, 3rd ed.**

**[OTexts.org/fpp3/](https://otexts.org/fpp3/)**

## Key reference

**Hyndman, R. J. & Athanasopoulos, G. (2021) *Forecasting: principles and practice*, 3rd ed.**

**[OTexts.org/fpp3/](https://otexts.org/fpp3/)**

- Free and online
- Data sets in associated R package
- R code for examples



# Key reference

**Hyndman, R. J. & Athanasopoulos, G. (2021) *Forecasting: principles and practice*, 3rd ed.**

**[OTexts.org/fpp3/](https://otexts.org/fpp3/)**

- Free and online
- Data sets in associated R package
- R code for examples

## Install required packages

```
install.packages(c("tidyverse", "fpp3", "GGally"))
```

# Approximate outline

Session	Topic	Chapter
1	Wrangling time series data	2
2	Visualizing trend and seasonal patterns	2, 3
3	Compute time series features for large collections of time series	4

**[bit.ly/etsa2022](https://bit.ly/etsa2022)**

# Outline

- 1 Introduction
- 2 Time series data and tsibbles
- 3 Example: Australian prison population
- 4 Example: Australian pharmaceutical sales
- 5 Lab Session 1



# Time series data

- Four-yearly Olympic winning times
- Annual Google profits
- Quarterly Australian beer production
- Monthly rainfall
- Weekly retail sales
- Daily IBM stock prices
- Hourly electricity demand
- 5-minute freeway traffic counts
- Time-stamped stock transaction data

# Class packages

```
library(fpp3)
```

Loads:

- Data sets
- Some tidyverse packages
- tsibble, tsibbledata, feasts and fable

# tsibble objects

```
global_economy
```

```
## # A tsibble: 15,150 x 6 [1Y]
```

```
## # Key:      Country [263]
```

##	Year	Country	GDP	Imports	Exports	Population
##	<dbl>	<fct>	<dbl>	<dbl>	<dbl>	<dbl>
##	1	1960 Afghanistan	5377777811.	7.02	4.13	8996351
##	2	1961 Afghanistan	5488888896.	8.10	4.45	9166764
##	3	1962 Afghanistan	5466666678.	9.35	4.88	9345868
##	4	1963 Afghanistan	7511111191.	16.9	9.17	9533954
##	5	1964 Afghanistan	8000000044.	18.1	8.89	9731361
##	6	1965 Afghanistan	10066666638.	21.4	11.3	9938414
##	7	1966 Afghanistan	13999999967.	18.6	8.57	10152331
##	8	1967 Afghanistan	16733333418.	14.2	6.77	10372630
##	9	1968 Afghanistan	13733333367.	15.2	8.90	10604346
##	10	1969 Afghanistan	14088888922.	15.0	10.1	10854428

# tsibble objects

```
global_economy
```

```
## # A tsibble: 15,150 x 6 [1Y]
```

```
## # Key:      Country [263]
```

##	Year	Country	GDP	Imports	Exports	Population
##	Index	<fct>	<dbl>	<dbl>	<dbl>	<dbl>
##	1	1960 Afghanistan	5377777811.	7.02	4.13	8996351
##	2	1961 Afghanistan	5488888896.	8.10	4.45	9166764
##	3	1962 Afghanistan	5466666678.	9.35	4.88	9345868
##	4	1963 Afghanistan	7511111191.	16.9	9.17	9533954
##	5	1964 Afghanistan	8000000044.	18.1	8.89	9731361
##	6	1965 Afghanistan	10066666638.	21.4	11.3	9938414
##	7	1966 Afghanistan	13999999967.	18.6	8.57	10152331
##	8	1967 Afghanistan	16733333418.	14.2	6.77	10372630
##	9	1968 Afghanistan	13733333367.	15.2	8.90	10604346
##	10	1969 Afghanistan	14088888922.	15.0	10.1	10854428



# tsibble objects

```
global_economy
```

```
## # A tsibble: 15,150 x 6 [1Y]
```

```
## # Key:      Country [263]
```

##	Year	Country	GDP	Imports	Exports	Population
##	Index	Key	<dbl>	<dbl>	<dbl>	<dbl>
##	1	1960 Afghanistan	5377777811.	7.02	4.13	8996351
##	2	1961 Afghanistan	5488888896.	8.10	4.45	9166764
##	3	1962 Afghanistan	5466666678.	9.35	4.88	9345868
##	4	1963 Afghanistan	7511111191.	16.9	9.17	9533954
##	5	1964 Afghanistan	8000000044.	18.1	8.89	9731361
##	6	1965 Afghanistan	10066666638.	21.4	11.3	9938414
##	7	1966 Afghanistan	13999999967.	18.6	8.57	10152331
##	8	1967 Afghanistan	16733333418.	14.2	6.77	10372630
##	9	1968 Afghanistan	13733333367.	15.2	8.90	10604346
##	10	1969 Afghanistan	14088888922.	15.0	10.1	10854428

# tsibble objects

```
global_economy
```

```
## # A tsibble: 15,150 x 6 [1Y]
```

```
## # Key:      Country [263]
```

```
##      Year Country      GDP Imports Exports Population
```

```
##      Index  Key      Measured variables
```

```
## 1  1960 Afghanistan 5377777811.    7.02    4.13    8996351
```

```
## 2  1961 Afghanistan 5488888896.    8.10    4.45    9166764
```

```
## 3  1962 Afghanistan 5466666678.    9.35    4.88    9345868
```

```
## 4  1963 Afghanistan 7511111191.   16.9    9.17    9533954
```

```
## 5  1964 Afghanistan 8000000044.   18.1    8.89    9731361
```

```
## 6  1965 Afghanistan 10066666638.  21.4   11.3    9938414
```

```
## 7  1966 Afghanistan 13999999967.  18.6    8.57   10152331
```

```
## 8  1967 Afghanistan 16733333418.  14.2    6.77   10372630
```

```
## 9  1968 Afghanistan 13733333367.  15.2    8.90   10604346
```

```
## 10 1969 Afghanistan 14088888922.  15.0   10.1   10854428
```

# tsibble objects

```
tourism
```

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:           Region, State, Purpose [304]
##   Quarter Region   State Purpose   Trips
##   <qtr> <chr>      <chr> <chr>    <dbl>
## 1 1998 Q1 Adelaide SA      Business 135.
## 2 1998 Q2 Adelaide SA      Business 110.
## 3 1998 Q3 Adelaide SA      Business 166.
## 4 1998 Q4 Adelaide SA      Business 127.
## 5 1999 Q1 Adelaide SA      Business 137.
## 6 1999 Q2 Adelaide SA      Business 200.
## 7 1999 Q3 Adelaide SA      Business 169.
## 8 1999 Q4 Adelaide SA      Business 134.
## 9 2000 Q1 Adelaide SA      Business 154.
## 10 2000 Q2 Adelaide SA      Business 169.
```

# tsibble objects

tourism

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:           Region, State, Purpose [304]
##   Quarter Region  State Purpose  Trips
##   <qtr> <chr>    <chr> <chr>    <dbl>
## 1 1998 Q1 Adelaide SA      Business 135.
## 2 1998 Q2 Adelaide SA      Business 110.
## 3 1998 Q3 Adelaide SA      Business 166.
## 4 1998 Q4 Adelaide SA      Business 127.
## 5 1999 Q1 Adelaide SA      Business 137.
## 6 1999 Q2 Adelaide SA      Business 200.
## 7 1999 Q3 Adelaide SA      Business 169.
## 8 1999 Q4 Adelaide SA      Business 134.
## 9 2000 Q1 Adelaide SA      Business 154.
## 10 2000 Q2 Adelaide SA      Business 169.
```

Domestic visitor  
nights in thousands  
by state/region and  
purpose.

# tsibble objects

tourism

```
## # A tsibble: 24,320 x 5 [1Q]
## # Key:           Region, State, Purpose [304]
##   Quarter Region  State Purpose  Trips
##   Index  <chr>    <chr> <chr>    <dbl>
## 1 1998 Q1 Adelaide SA      Business 135.
## 2 1998 Q2 Adelaide SA      Business 110.
## 3 1998 Q3 Adelaide SA      Business 166.
## 4 1998 Q4 Adelaide SA      Business 127.
## 5 1999 Q1 Adelaide SA      Business 137.
## 6 1999 Q2 Adelaide SA      Business 200.
## 7 1999 Q3 Adelaide SA      Business 169.
## 8 1999 Q4 Adelaide SA      Business 134.
## 9 2000 Q1 Adelaide SA      Business 154.
## 10 2000 Q2 Adelaide SA      Business 169.
```

Domestic visitor  
nights in thousands  
by state/region and  
purpose.

# tsibble objects

tourism

```
## # A tsibble: 24,320 x 5 [1Q]
```

```
## # Key:           Region, State, Purpose [304]
```

```
##   Quarter Region  State Purpose  Trips
```

```
##   Index  Keys      <dbl>
```

```
## 1 1998 Q1 Adelaide SA      Business 135.
```

```
## 2 1998 Q2 Adelaide SA      Business 110.
```

```
## 3 1998 Q3 Adelaide SA      Business 166.
```

```
## 4 1998 Q4 Adelaide SA      Business 127.
```

```
## 5 1999 Q1 Adelaide SA      Business 137.
```

```
## 6 1999 Q2 Adelaide SA      Business 200.
```

```
## 7 1999 Q3 Adelaide SA      Business 169.
```

```
## 8 1999 Q4 Adelaide SA      Business 134.
```

```
## 9 2000 Q1 Adelaide SA      Business 154.
```

```
## 10 2000 Q2 Adelaide SA      Business 169.
```

Domestic visitor  
nights in thousands  
by state/region and  
purpose.

# tsibble objects

tourism

```
## # A tsibble: 24,320 x 5 [1Q]
```

```
## # Key:           Region, State, Purpose [304]
```

```
##   Quarter Region  State Purpose  Trips
```

```
##   Index  Keys      Measure
```

```
## 1 1998 Q1 Adelaide SA      Business 135.
```

```
## 2 1998 Q2 Adelaide SA      Business 110.
```

```
## 3 1998 Q3 Adelaide SA      Business 166.
```

```
## 4 1998 Q4 Adelaide SA      Business 127.
```

```
## 5 1999 Q1 Adelaide SA      Business 137.
```

```
## 6 1999 Q2 Adelaide SA      Business 200.
```

```
## 7 1999 Q3 Adelaide SA      Business 169.
```

```
## 8 1999 Q4 Adelaide SA      Business 134.
```

```
## 9 2000 Q1 Adelaide SA      Business 154.
```

```
## 10 2000 Q2 Adelaide SA      Business 169.
```

Domestic visitor  
nights in thousands  
by state/region and  
purpose.

# tsibble objects

- A `tsibble` allows storage and manipulation of multiple time series in R.
- It contains:
  - ▶ An index: time information about the observation
  - ▶ Measured variable(s): numbers of interest
  - ▶ Key variable(s): optional unique identifiers for each series
- It works with tidyverse functions.



# The tsibble index

For observations more frequent than once per year, we need to use a time class function on the index.

```
z
```

```
## # A tibble: 5 x 2
##   Month      Observation
##   <chr>         <dbl>
## 1 2019 Jan           50
## 2 2019 Feb           23
## 3 2019 Mar           34
## 4 2019 Apr           30
## 5 2019 May           25
```

# The tsibble index

For observations more frequent than once per year, we need to use a time class function on the index.

```
z ▷  
  mutate(Month = yearmonth(Month)) ▷  
  as_tsibble(index = Month)
```

```
## # A tsibble: 5 x 2 [1M]
```

```
##       Month Observation
```

```
##      <mth>         <dbl>
```

```
## 1 2019 Jan          50
```

```
## 2 2019 Feb          23
```

```
## 3 2019 Mar          34
```

```
## 4 2019 Apr          30
```

```
## 5 2019 May          25
```

# The tsibble index

Common time index variables can be created with these functions:

Frequency	Function
Quarterly	<code>yearquarter( )</code>
Monthly	<code>yearmonth( )</code>
Weekly	<code>yearweek( )</code>
Daily	<code>as_date( ), ymd( )</code>
Sub-daily	<code>as_datetime( )</code>

# Outline

- 1 Introduction
- 2 Time series data and tsibbles
- 3 Example: Australian prison population
- 4 Example: Australian pharmaceutical sales
- 5 Lab Session 1

# Australian prison population



# Read a csv file and convert to a tibble

```
prison <- readr::read_csv("data/prison_population.csv")
```

```
## # A tibble: 3,072 x 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 Other         2
## 3 2005-03-01 ACT    Female Sentenced ATSI           0
## 4 2005-03-01 ACT    Female Sentenced Other         0
## 5 2005-03-01 ACT    Male   Remanded ATSI           7
## 6 2005-03-01 ACT    Male   Remanded Other        58
## 7 2005-03-01 ACT    Male   Sentenced ATSI           0
## 8 2005-03-01 ACT    Male   Sentenced Other         0
## 9 2005-03-01 NSW    Female Remanded ATSI          51
## 10 2005-03-01 NSW    Female Remanded Other       131
## # ... with 3,062 more rows
```

# Read a csv file and convert to a tibble

```
prison <- readr::read_csv("data/prison_population.csv") ▷  
  mutate(Quarter = yearquarter(date))
```

```
## # A tibble: 3,072 x 7
```

```
##   date      state gender legal      indigenous count Quarter  
##   <date>    <chr> <chr>  <chr>    <chr>         <dbl>   <qtr>  
## 1 2005-03-01 ACT    Female Remanded ATSI           0 2005 Q1  
## 2 2005-03-01 ACT    Female Remanded Other         2 2005 Q1  
## 3 2005-03-01 ACT    Female Sentenced ATSI           0 2005 Q1  
## 4 2005-03-01 ACT    Female Sentenced Other         0 2005 Q1  
## 5 2005-03-01 ACT    Male   Remanded ATSI           7 2005 Q1  
## 6 2005-03-01 ACT    Male   Remanded Other        58 2005 Q1  
## 7 2005-03-01 ACT    Male   Sentenced ATSI           0 2005 Q1  
## 8 2005-03-01 ACT    Male   Sentenced Other         0 2005 Q1  
## 9 2005-03-01 NSW    Female Remanded ATSI          51 2005 Q1  
## 10 2005-03-01 NSW    Female Remanded Other       131 2005 Q1
```

# Read a csv file and convert to a tibble

```
prison <- readr::read_csv("data/prison_population.csv") ▷  
  mutate(Quarter = yearquarter(date)) ▷  
  select(-date)
```

## # A tibble: 3,072 x 6

##	state	gender	legal	indigenous	count	Quarter
##	<chr>	<chr>	<chr>	<chr>	<dbl>	<qtr>
##	1 ACT	Female	Remanded	ATSI	0	2005 Q1
##	2 ACT	Female	Remanded	Other	2	2005 Q1
##	3 ACT	Female	Sentenced	ATSI	0	2005 Q1
##	4 ACT	Female	Sentenced	Other	0	2005 Q1
##	5 ACT	Male	Remanded	ATSI	7	2005 Q1
##	6 ACT	Male	Remanded	Other	58	2005 Q1
##	7 ACT	Male	Sentenced	ATSI	0	2005 Q1
##	8 ACT	Male	Sentenced	Other	0	2005 Q1
##	9 NSW	Female	Remanded	ATSI	51	2005 Q1
##	10 NSW	Female	Remanded	Other	124	2005 Q1



# Read a csv file and convert to a tsibble

```
prison <- readr::read_csv("data/prison_population.csv") ▷  
  mutate(Quarter = yearquarter(date)) ▷  
  select(-date) ▷  
  as_tsibble(  
    index = Quarter,  
    key = c(state, gender, legal, indigenous)  
  )
```

```
## # 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
```

# Outline

- 1 Introduction
- 2 Time series data and tsibbles
- 3 Example: Australian prison population
- 4 Example: Australian pharmaceutical sales
- 5 Lab Session 1

# Australian Pharmaceutical Benefits Scheme



# Australian Pharmaceutical Benefits Scheme

The **Pharmaceutical Benefits Scheme** (PBS) is the Australian government drugs subsidy scheme.

# Australian Pharmaceutical Benefits Scheme

The **Pharmaceutical Benefits Scheme** (PBS) is the Australian government drugs subsidy scheme.

- Many drugs bought from pharmacies are subsidised to allow more equitable access to modern drugs.
- The cost to government is determined by the number and types of drugs purchased. Currently nearly 1% of GDP.
- The total cost is budgeted based on forecasts of drug usage.
- Costs are disaggregated by drug type (ATC1 x15 / ATC2 84), concession category (x2) and patient type (x2), giving  $84 \times 2 \times 2 = 336$  time series.

# Working with tsibble objects

## PBS

```
## # A tsibble: 67,596 x 9 [1M]
## # Key:      Concession, Type, ATC1, ATC2 [336]
##   Month Concession  Type      ATC1  ATC1_desc      ATC2  ATC2_~1 Scripts  Cost
##   <nth> <chr>      <chr>    <chr> <chr>          <chr> <chr>    <dbl> <dbl>
## 1 1991 Jul Concessional Co-payments A      Alimentary tr~ A01  STOMAT~    18228 67877
## 2 1991 Aug Concessional Co-payments A      Alimentary tr~ A01  STOMAT~    15327 57011
## 3 1991 Sep Concessional Co-payments A      Alimentary tr~ A01  STOMAT~    14775 55020
## 4 1991 Oct Concessional Co-payments A      Alimentary tr~ A01  STOMAT~    15380 57222
## 5 1991 Nov Concessional Co-payments A      Alimentary tr~ A01  STOMAT~    14371 52120
## 6 1991 Dec Concessional Co-payments A      Alimentary tr~ A01  STOMAT~    15028 54299
## 7 1992 Jan Concessional Co-payments A      Alimentary tr~ A01  STOMAT~    11040 39753
## 8 1992 Feb Concessional Co-payments A      Alimentary tr~ A01  STOMAT~    15165 54405
## 9 1992 Mar Concessional Co-payments A      Alimentary tr~ A01  STOMAT~    16898 61108
## 10 1992 Apr Concessional Co-payments A      Alimentary tr~ A01  STOMAT~    18141 65356
## # ... with 67,586 more rows, and abbreviated variable name 1: ATC2_desc
```

# Working with tsibble objects

We can use the `filter()` function to select rows.

```
PBS ▷  
  filter(ATC2 = "A10")
```

```
## # A tsibble: 816 x 9 [1M]  
## # Key:      Concession, Type, ATC1, ATC2 [4]  
##   Month Concession  Type      ATC1 ATC1_desc  ATC2 ATC2_~1 Scripts  Cost  
##   <mt> <chr>         <chr>    <chr> <chr>      <chr> <chr>    <dbl> <dbl>  
## 1 1991 Jul Concessional Co-payments A      Alimentary t~ A10  ANTIDI~  89733 2.09e6  
## 2 1991 Aug Concessional Co-payments A      Alimentary t~ A10  ANTIDI~  77101 1.80e6  
## 3 1991 Sep Concessional Co-payments A      Alimentary t~ A10  ANTIDI~  76255 1.78e6  
## 4 1991 Oct Concessional Co-payments A      Alimentary t~ A10  ANTIDI~  78681 1.85e6  
## 5 1991 Nov Concessional Co-payments A      Alimentary t~ A10  ANTIDI~  70554 1.69e6  
## 6 1991 Dec Concessional Co-payments A      Alimentary t~ A10  ANTIDI~  75814 1.84e6  
## 7 1992 Jan Concessional Co-payments A      Alimentary t~ A10  ANTIDI~  64186 1.56e6  
## 8 1992 Feb Concessional Co-payments A      Alimentary t~ A10  ANTIDI~  75899 1.73e6  
## 9 1992 Mar Concessional Co-payments A      Alimentary t~ A10  ANTIDI~  89445 2.05e6
```

# Working with tsibble objects

We can use the `select( )` function to select columns.

```
PBS ▷  
  filter(ATC2 = "A10") ▷  
  select(Month, Concession, Type, Cost)
```

```
## # A tsibble: 816 x 4 [1M]  
## # Key:      Concession, Type [4]  
##      Month Concession  Type          Cost  
##      <mt> <chr>        <chr>        <dbl>  
## 1 1991 Jul Concessional Co-payments 2092878  
## 2 1991 Aug Concessional Co-payments 1795733  
## 3 1991 Sep Concessional Co-payments 1777231  
## 4 1991 Oct Concessional Co-payments 1848507  
## 5 1991 Nov Concessional Co-payments 1686458  
## 6 1991 Dec Concessional Co-payments 1843079  
## 7 1992 Jan Concessional Co-payments 1564702  
## 8 1992 Feb Concessional Co-payments 1732508
```



# Working with tsibble objects

We can use the `summarise()` function to summarise over keys.

```
PBS ▷  
  filter(ATC2 = "A10") ▷  
  select(Month, Concession, Type, Cost) ▷  
  summarise(total_cost = sum(Cost))
```

```
## # A tsibble: 204 x 2 [1M]
```

```
##       Month total_cost
```

```
##       <mth>      <dbl>
```

```
## 1 1991 Jul      3526591
```

```
## 2 1991 Aug      3180891
```

```
## 3 1991 Sep      3252221
```

```
## 4 1991 Oct      3611003
```

```
## 5 1991 Nov      3565869
```

```
## 6 1991 Dec      4306371
```

```
## 7 1992 Jan      5088335
```

```
## 8 1992 Feb      2814520
```

# Working with tsibble objects

We can use the `mutate()` function to create new variables.

```
PBS ▷  
  filter(ATC2 = "A10") ▷  
  select(Month, Concession, Type, Cost) ▷  
  summarise(total_cost = sum(Cost)) ▷  
  mutate(total_cost = total_cost / 1e6)
```

```
## # A tsibble: 204 x 2 [1M]  
##       Month total_cost  
##       <mth>      <dbl>  
## 1 1991 Jul         3.53  
## 2 1991 Aug         3.18  
## 3 1991 Sep         3.25  
## 4 1991 Oct         3.61  
## 5 1991 Nov         3.57  
## 6 1991 Dec         4.31  
## 7 1992 Jan         5.09
```

# Working with tsibble objects

We can use the `mutate()` function to create new variables.

```
PBS ▷  
  filter(ATC2 = "A10") ▷  
  select(Month, Concession, Type, Cost) ▷  
  summarise(total_cost = sum(Cost)) ▷  
  mutate(total_cost = total_cost / 1e6) → a10
```

```
## # A tsibble: 204 x 2 [1M]  
##       Month total_cost  
##       <mth>      <dbl>  
## 1 1991 Jul         3.53  
## 2 1991 Aug         3.18  
## 3 1991 Sep         3.25  
## 4 1991 Oct         3.61  
## 5 1991 Nov         3.57  
## 6 1991 Dec         4.31  
## 7 1992 Jan         5.09
```

# Outline

- 1 Introduction
- 2 Time series data and tsibbles
- 3 Example: Australian prison population
- 4 Example: Australian pharmaceutical sales
- 5 Lab Session 1

# Lab Session 1

- 1 Download `tourism.xlsx` from `http://robjhyndman.com/data/tourism.xlsx`, and read it into R using `read_excel()` from the `readxl` package.
- 2 Create a `tsibble` which is identical to the `tourism` `tsibble` from the `tsibble` package.
- 3 Find what combination of `Region` and `Purpose` had the maximum number of overnight trips on average.
- 4 Create a new `tsibble` which combines the `Purposes` and `Regions`, and just has total trips by `State`.