

Time Series Analysis & Forecasting Using R

bit.ly/fable2023

6. Introduction to forecasting



Outline

- 1 Statistical forecasting
- 2 Benchmark methods
- 3 Lab Session 11
- 4 Residual diagnostics
- 5 Lab Session 12
- 6 Forecast accuracy measures
- 7 Lab Session 13

Outline

1 Statistical forecasting

2 Benchmark methods

3 Lab Session 11

4 Residual diagnostics

5 Lab Session 12

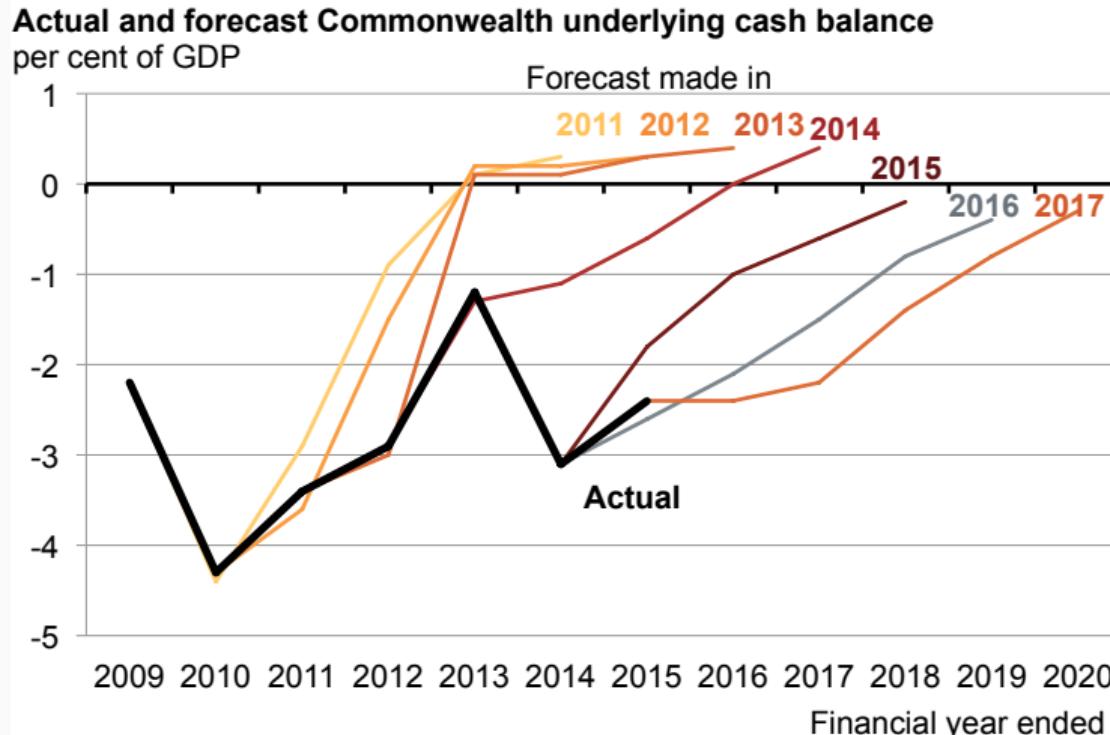
6 Forecast accuracy measures

7 Lab Session 13

Forecasting is difficult

Commonwealth plans to drift back to surplus
show the triumph of experience over hope

GRATTAN
Institute



What can we forecast?



What can we forecast?



What can we forecast?

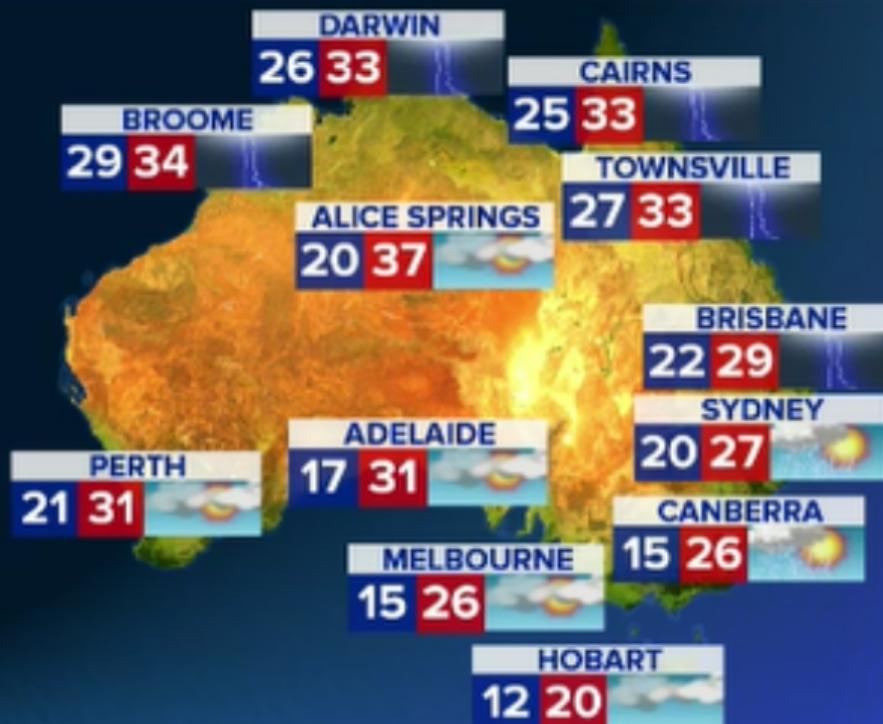


What can we forecast?



What can we forecast?

TOMORROW



What can we forecast?



What can we forecast?



Which is easiest to forecast?

- 1 daily electricity demand in 3 days time
- 2 timing of next Halley's comet appearance
- 3 time of sunrise this day next year
- 4 Google stock price tomorrow
- 5 Google stock price in 6 months time
- 6 maximum temperature tomorrow
- 7 exchange rate of \$US/AUS next week
- 8 total sales of drugs in Australian pharmacies next month

Which is easiest to forecast?

- 1 daily electricity demand in 3 days time
 - 2 timing of next Halley's comet appearance
 - 3 time of sunrise this day next year
 - 4 Google stock price tomorrow
 - 5 Google stock price in 6 months time
 - 6 maximum temperature tomorrow
 - 7 exchange rate of \$US/AUS next week
 - 8 total sales of drugs in Australian pharmacies next month
-
- how do we measure “easiest”?
 - what makes something easy/difficult to forecast?

Factors affecting forecastability

Something is easier to forecast if:

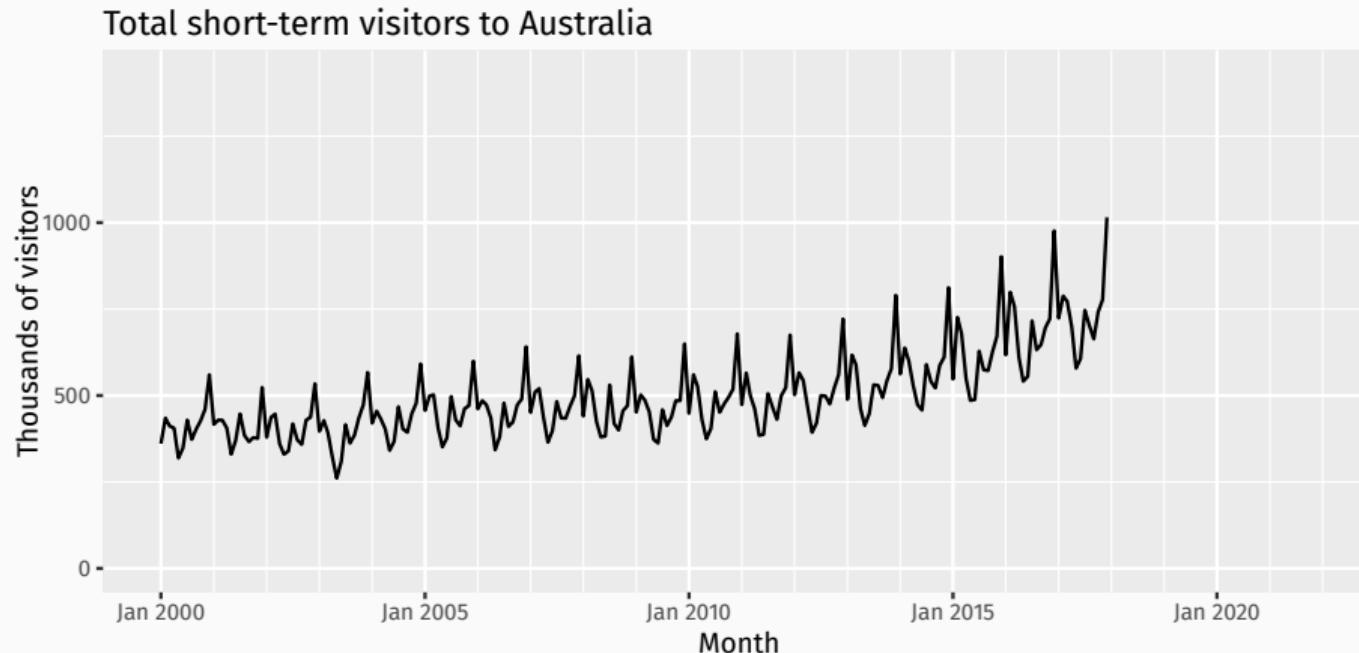
- we have a good understanding of the factors that contribute to it
- there is lots of data available;
- the forecasts cannot affect the thing we are trying to forecast.
- there is relatively low natural/unexplainable random variation.
- the future is somewhat similar to the past

Random futures

A forecast is an estimate of the probabilities of possible futures.

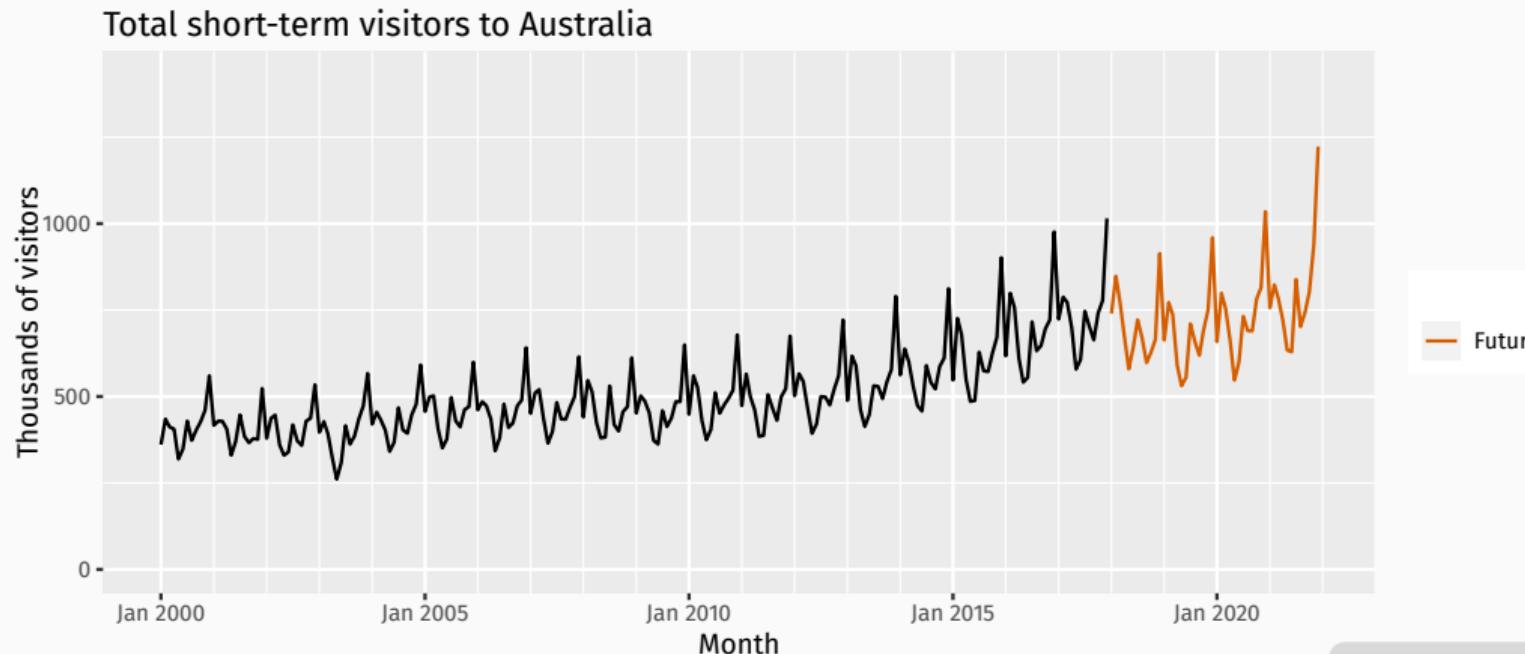
Random futures

A forecast is an estimate of the probabilities of possible futures.



Random futures

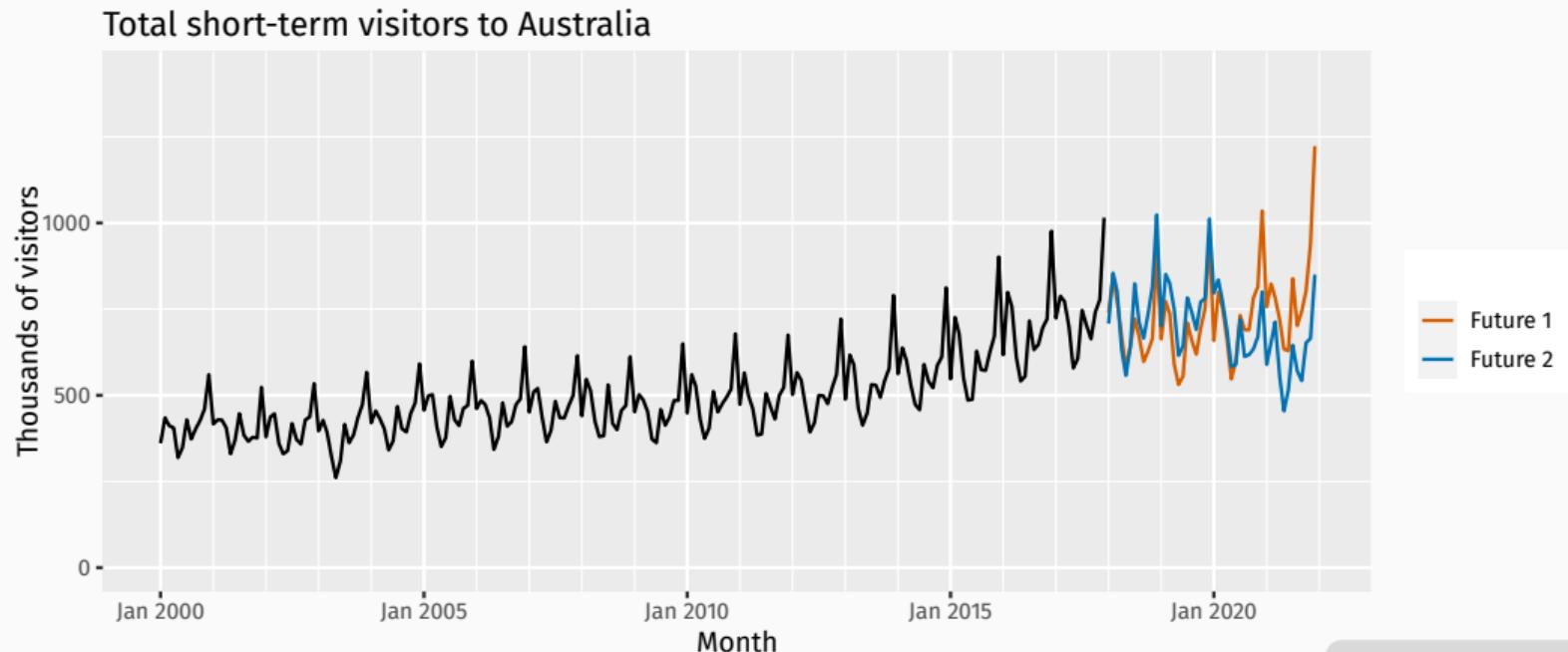
A forecast is an estimate of the probabilities of possible futures.



Simulated futures
from an ETS model

Random futures

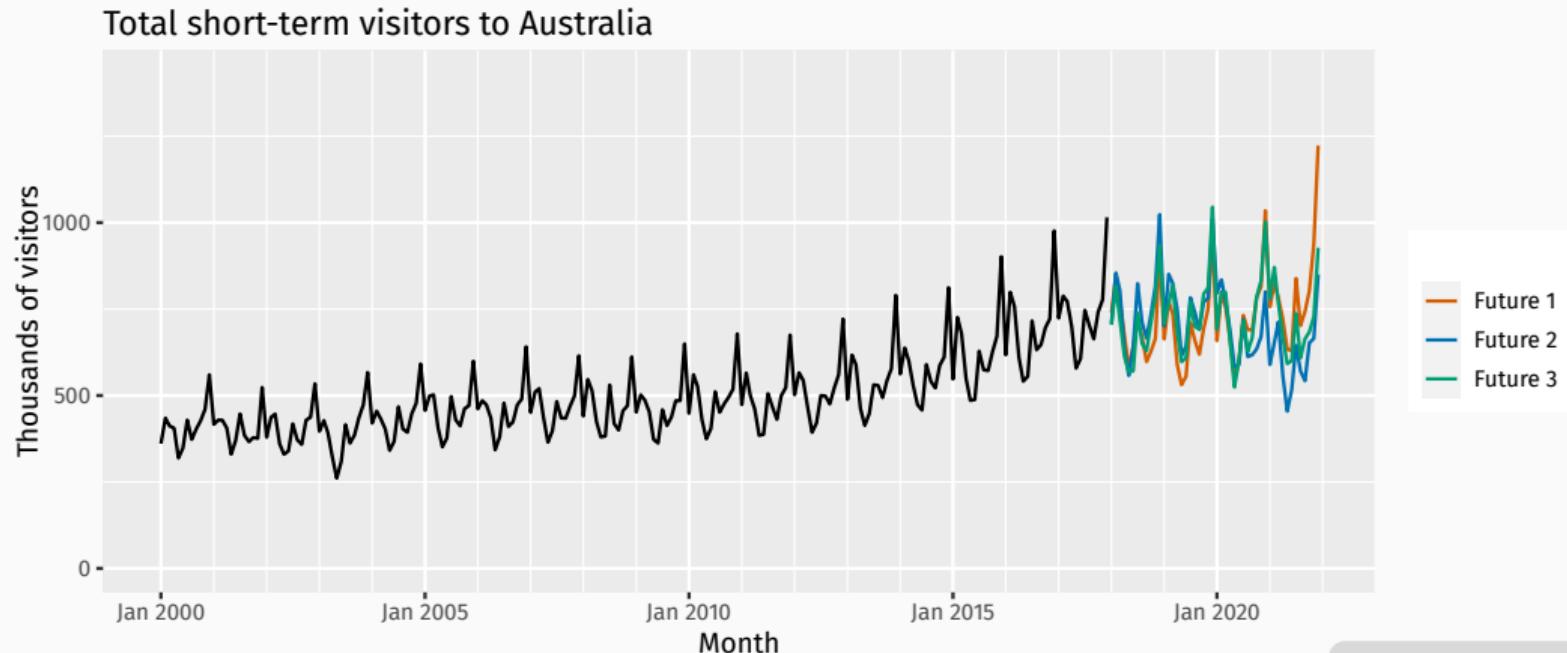
A forecast is an estimate of the probabilities of possible futures.



Simulated futures
from an ETS model

Random futures

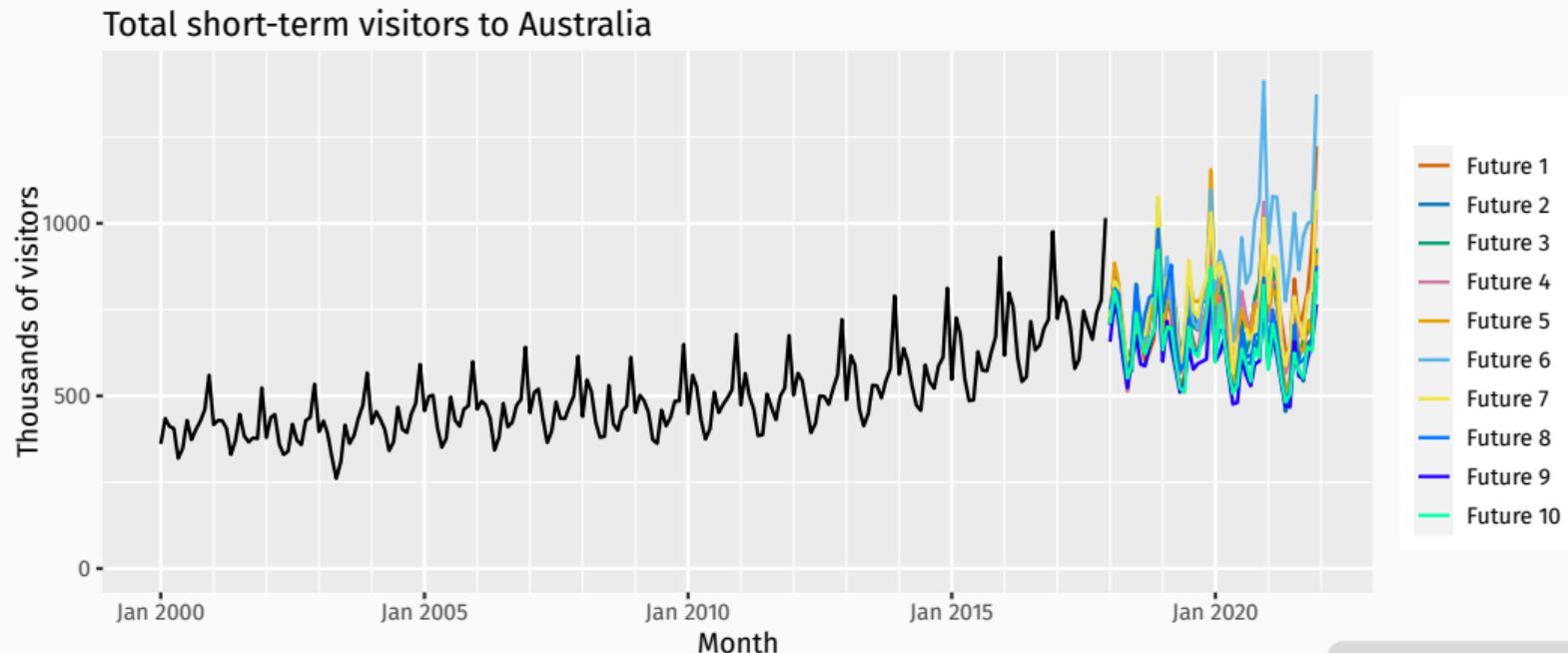
A forecast is an estimate of the probabilities of possible futures.



Simulated futures
from an ETS model

Random futures

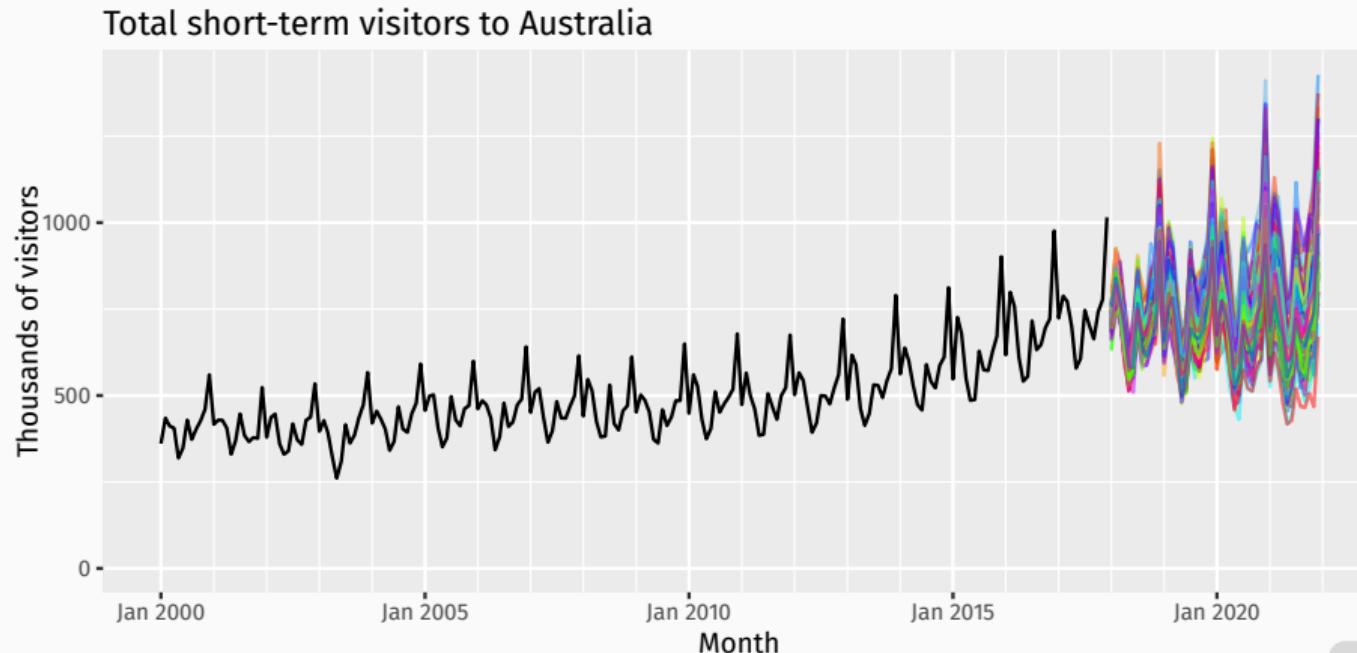
A forecast is an estimate of the probabilities of possible futures.



Simulated futures
from an ETS model

Random futures

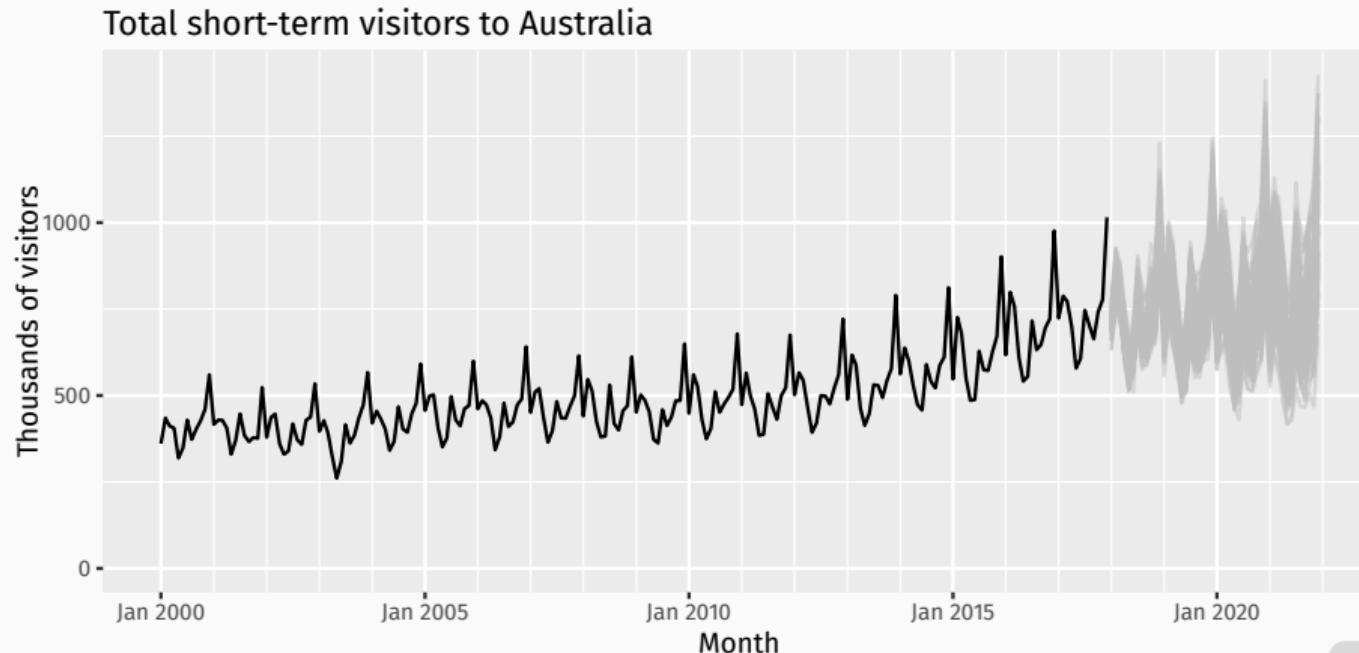
A forecast is an estimate of the probabilities of possible futures.



Simulated futures
from an ETS model

Random futures

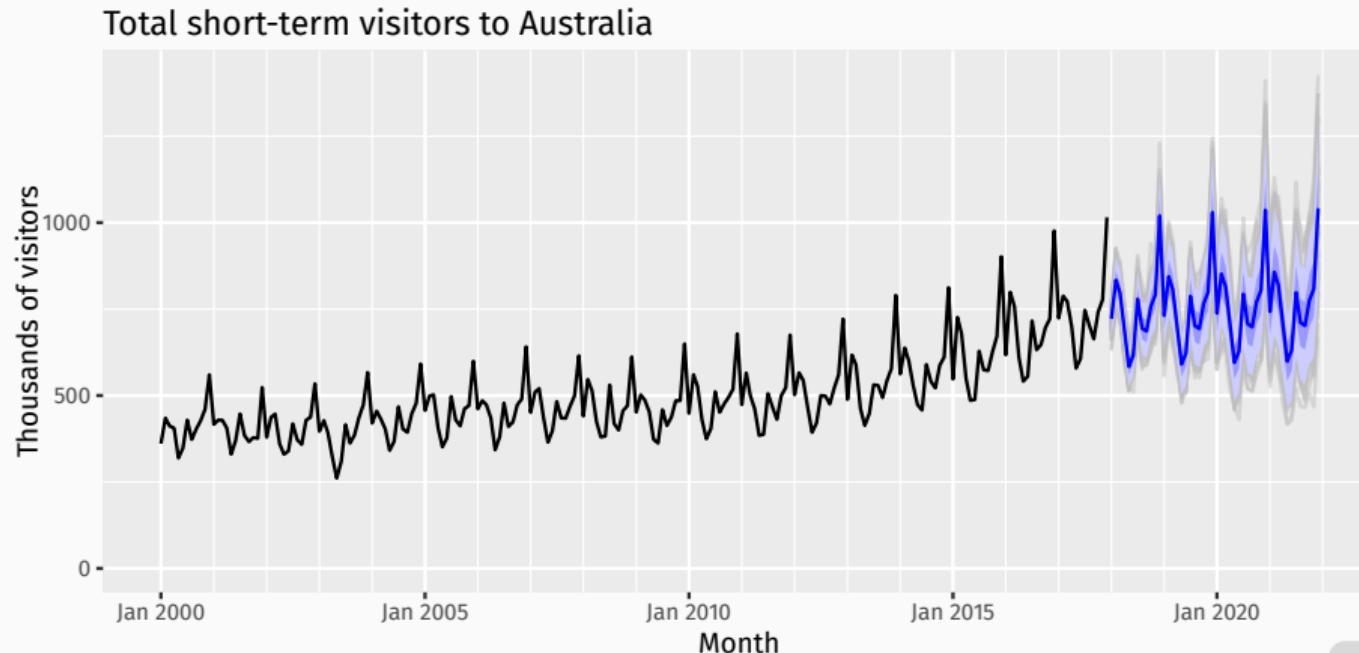
A forecast is an estimate of the probabilities of possible futures.



Simulated futures
from an ETS model

Random futures

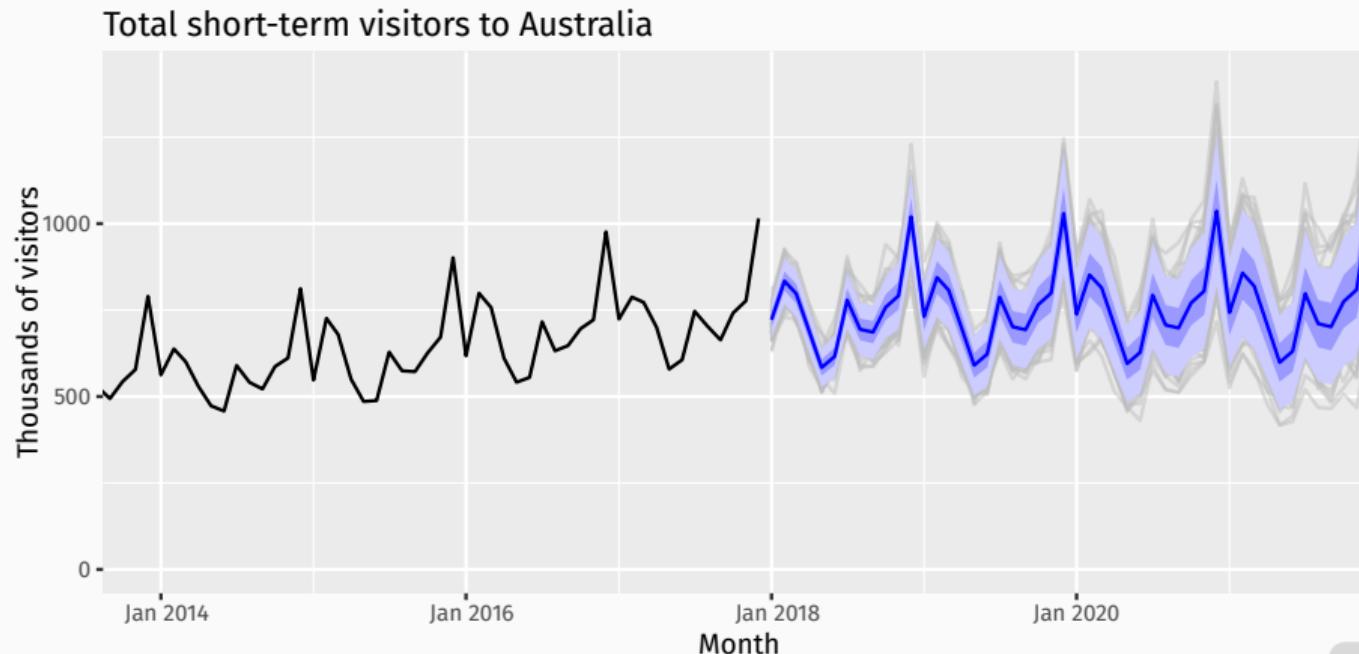
A forecast is an estimate of the probabilities of possible futures.



Simulated futures
from an ETS model

Random futures

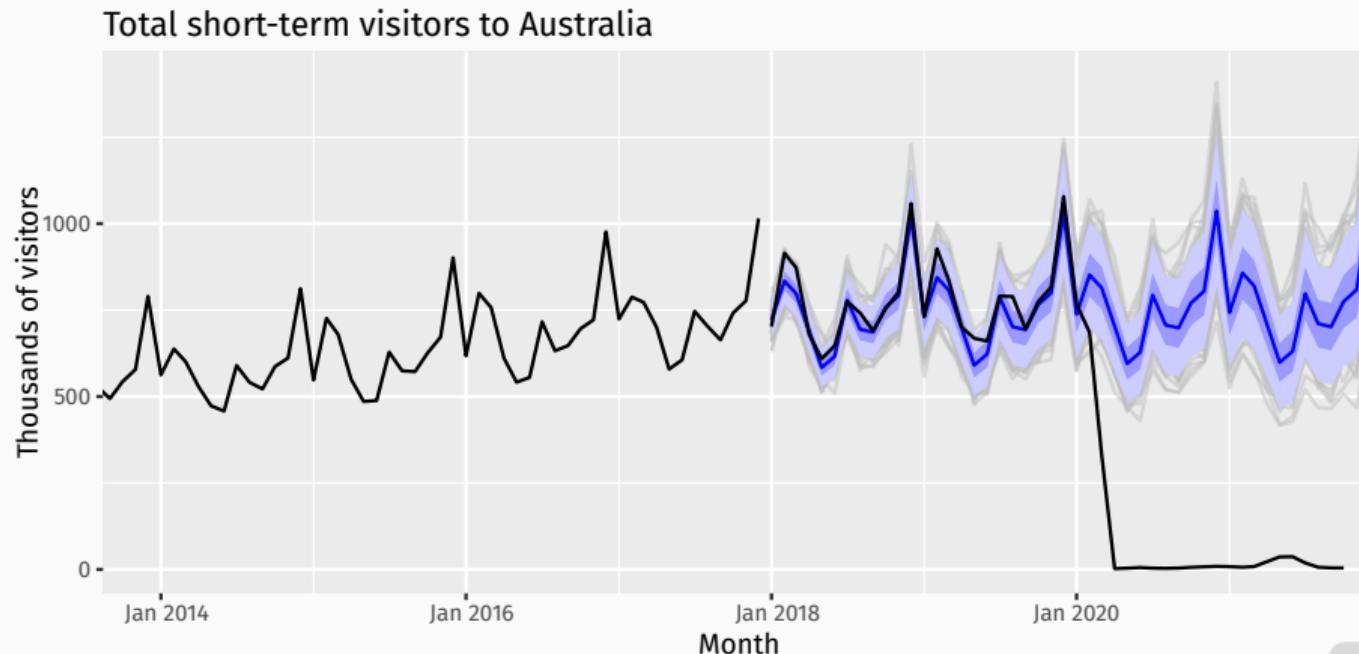
A forecast is an estimate of the probabilities of possible futures.



Simulated futures
from an ETS model

Random futures

A forecast is an estimate of the probabilities of possible futures.

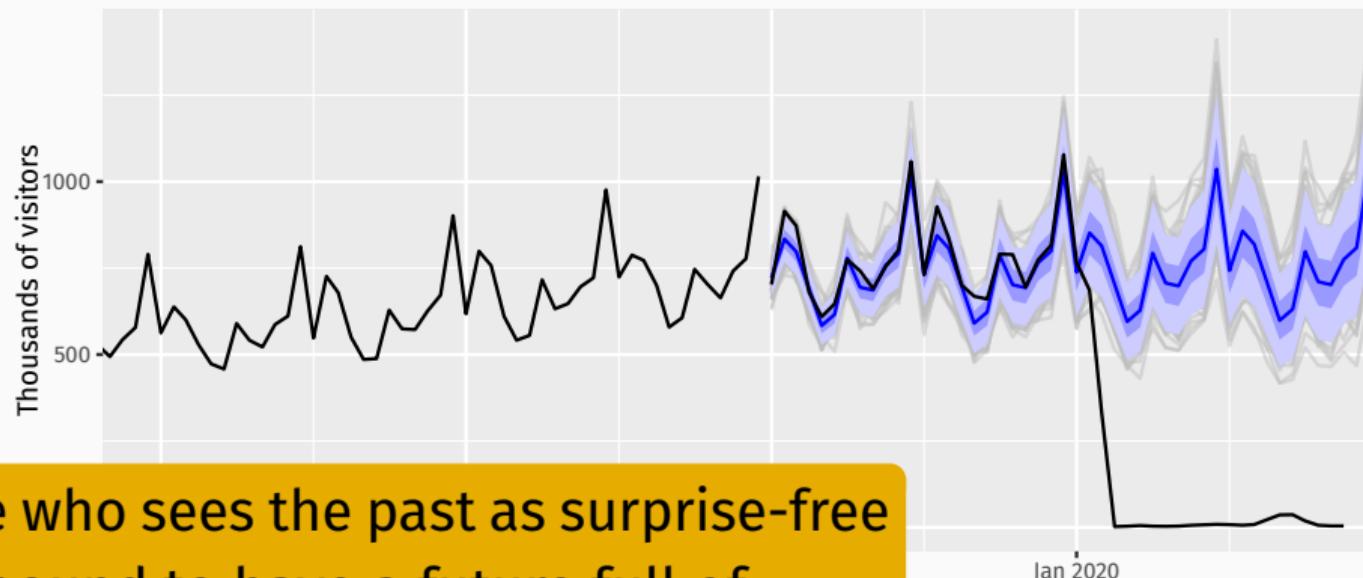


Simulated futures
from an ETS model

Random futures

A forecast is an estimate of the probabilities of possible futures.

Total short-term visitors to Australia



“He who sees the past as surprise-free
is bound to have a future full of
surprises.”

Jan 2020

Simulated futures
from an ETS model

Statistical forecasting

- Thing to be forecast: y_{T+h} .
- What we know: y_1, \dots, y_T .
- Forecast distribution: $y_{T+h|t} = y_{T+h} \mid \{y_1, y_2, \dots, y_T\}$.
- Point forecast: $\hat{y}_{T+h|T} = E[y_{T+h} \mid y_1, \dots, y_T]$.
- Forecast variance: $\text{Var}[y_t \mid y_1, \dots, y_T]$
- Prediction interval is a range of values of y_{T+h} with high probability.

Outline

1 Statistical forecasting

2 Benchmark methods

3 Lab Session 11

4 Residual diagnostics

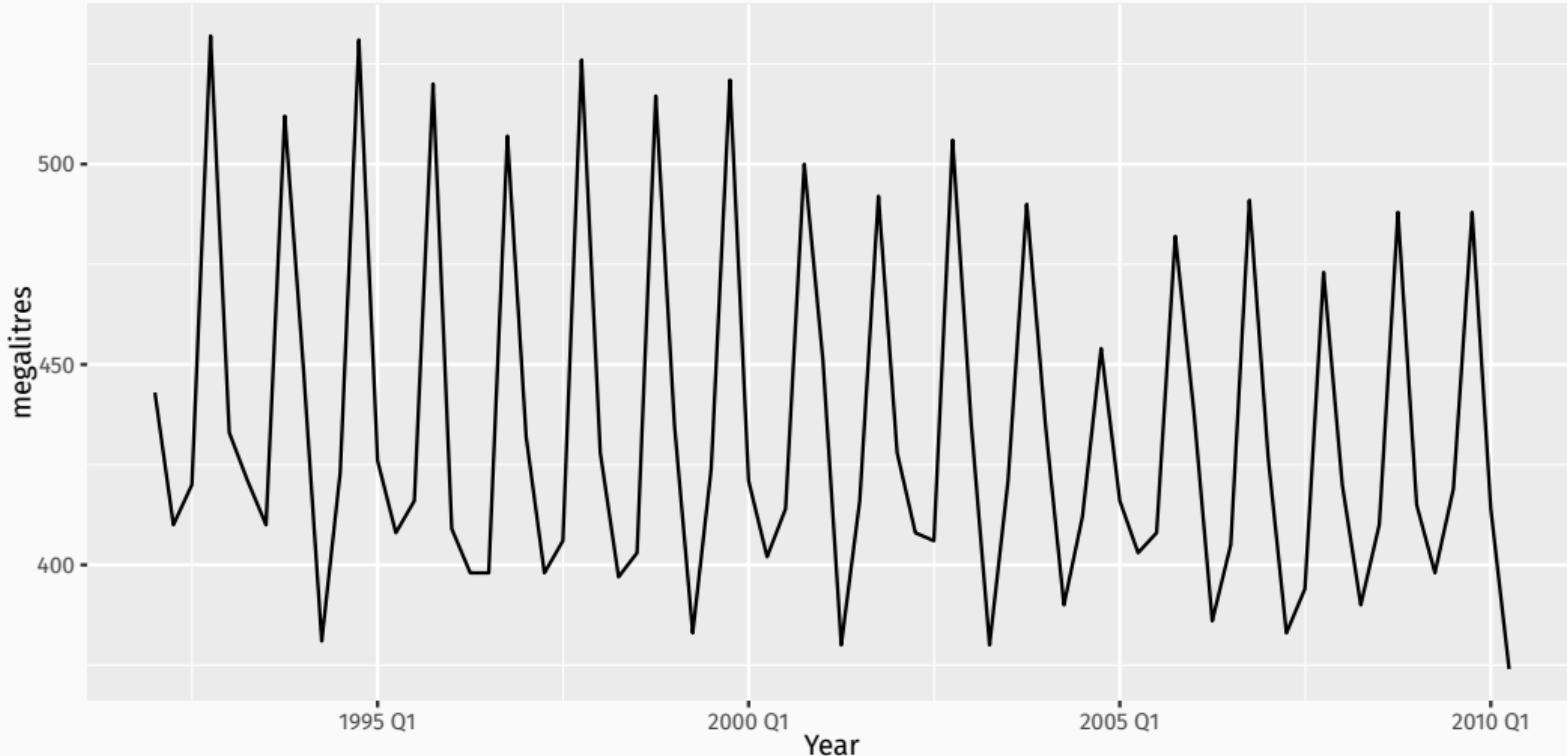
5 Lab Session 12

6 Forecast accuracy measures

7 Lab Session 13

Some simple forecasting methods

Australian quarterly beer production

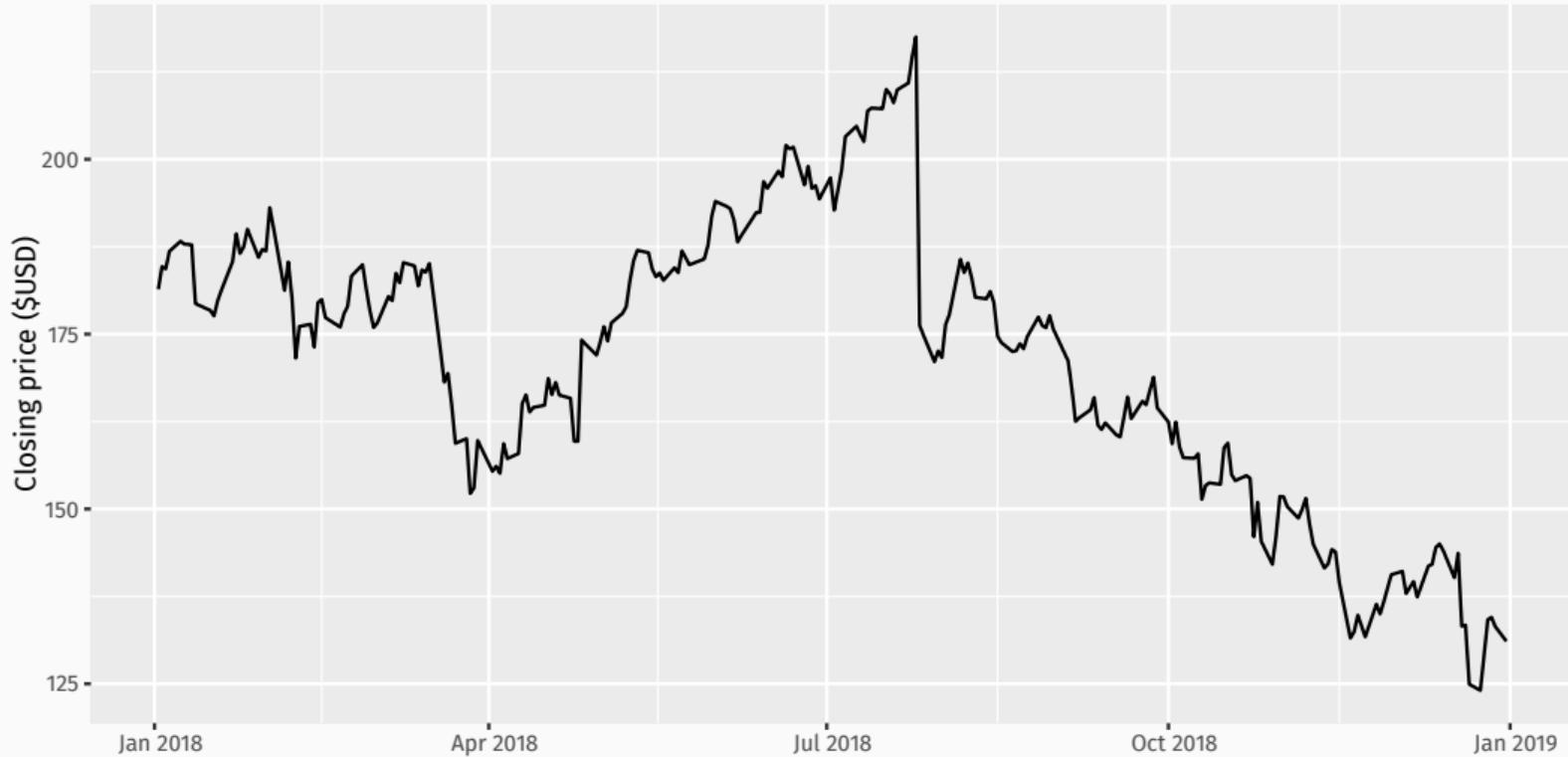


Some simple forecasting methods



Some simple forecasting methods

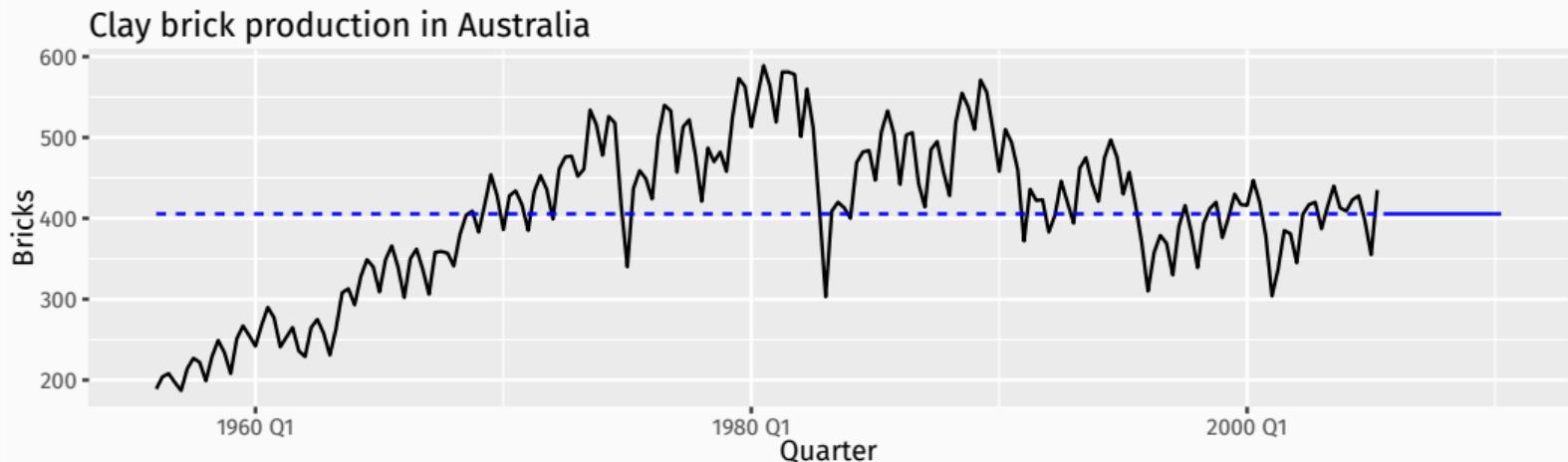
Facebook closing stock price in 2018



Some simple forecasting methods

MEAN(y): Average method

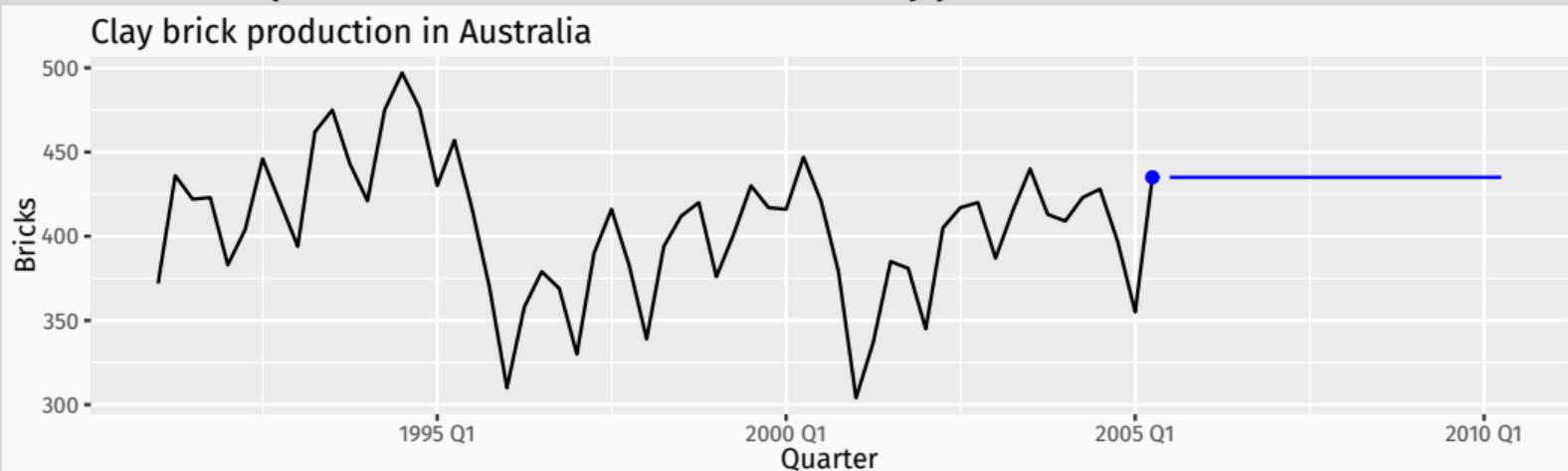
- Forecast of all future values is equal to mean of historical data $\{y_1, \dots, y_T\}$.
- Forecasts: $\hat{y}_{T+h|T} = \bar{y} = (y_1 + \dots + y_T)/T$



Some simple forecasting methods

NAIVE(y): Naïve method

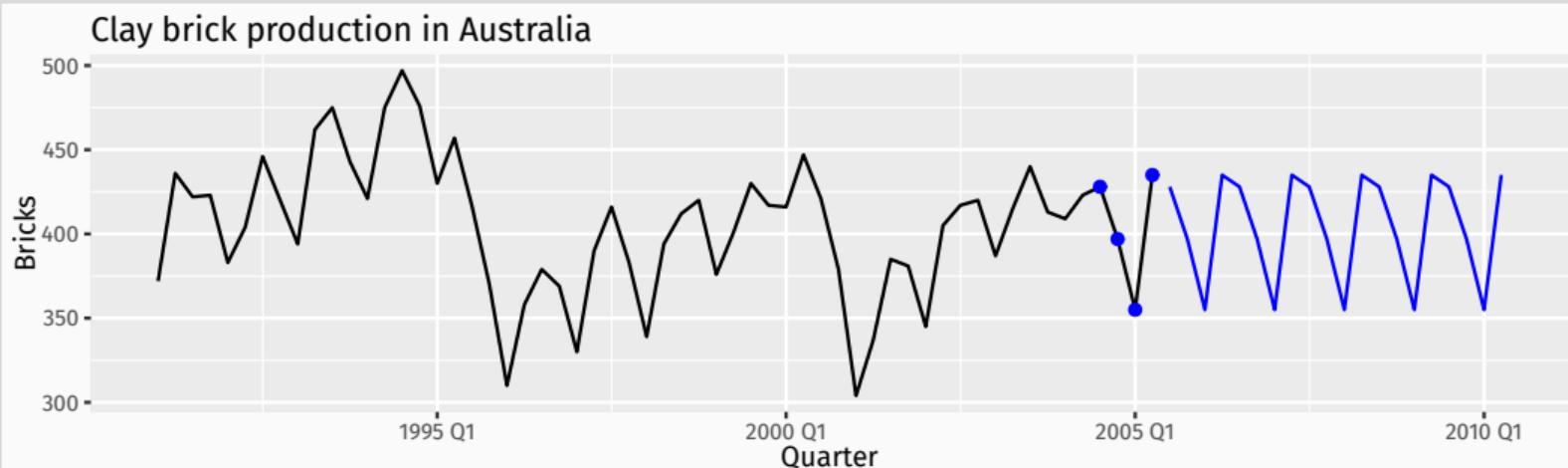
- Forecasts equal to last observed value.
- Forecasts: $\hat{y}_{T+h|T} = y_T$.
- Consequence of efficient market hypothesis.



Some simple forecasting methods

SNAIVE($y \sim \text{lag}(m)$): Seasonal naïve method

- Forecasts equal to last value from same season.
- Forecasts: $\hat{y}_{T+h|T} = y_{T+h-m(k+1)}$, where m = seasonal period and k is the integer part of $(h - 1)/m$.



Some simple forecasting methods

RW(y ~ drift()): Drift method

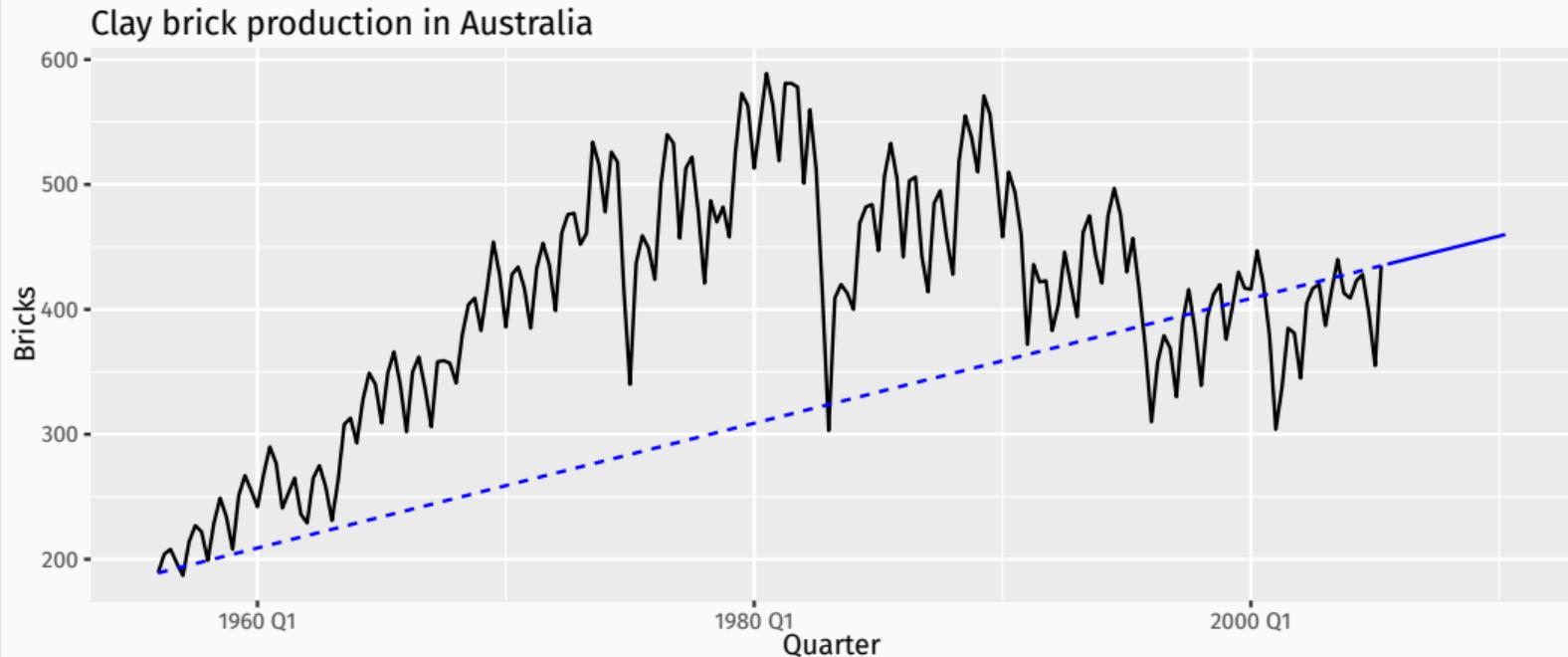
- Forecasts equal to last value plus average change.
- Forecasts:

$$\begin{aligned}\hat{y}_{T+h|T} &= y_T + \frac{h}{T-1} \sum_{t=2}^T (y_t - y_{t-1}) \\ &= y_T + \frac{h}{T-1} (y_T - y_1).\end{aligned}$$

- Equivalent to extrapolating a line drawn between first and last observations.

Some simple forecasting methods

Drift method



Model fitting

The `model()` function trains models to data.

```
brick_fit <- aus_production |>  
  filter(!is.na(Bricks)) |>  
  model(  
    `Seasonal_naïve` = SNAIVE(Bricks),  
    `Naïve` = NAIVE(Bricks),  
    Drift = RW(Bricks ~ drift()),  
    Mean = MEAN(Bricks)  
)
```

```
# A mable: 1 x 4  
  Seasonal_naïve   Naïve          Drift      Mean  
  <model> <model>       <model> <model>  
1     <SNAIVE> <NAIVE> <RW w/ drift> <MEAN>
```

A `mable` is a model table, each cell corresponds to a fitted model.

Producing forecasts

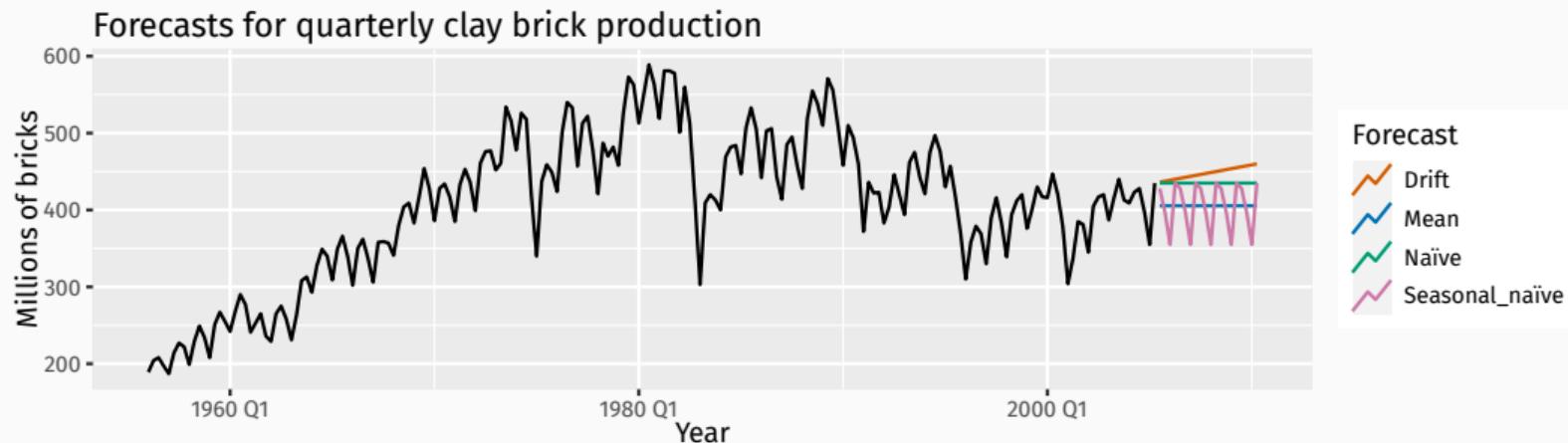
```
brick_fc <- brick_fit |>  
  forecast(h = "5 years")
```

```
# A fable: 80 x 4 [1Q]  
# Key:     .model [4]  
  
.model      Quarter      Bricks .mean  
<chr>       <qtr>       <dist> <dbl>  
1 Seasonal_naïve 2005 Q3 N(428, 2336)    428  
2 Seasonal_naïve 2005 Q4 N(397, 2336)    397  
3 Seasonal_naïve 2006 Q1 N(355, 2336)    355  
4 Seasonal_naïve 2006 Q2 N(435, 2336)    435  
# i 76 more rows
```

A fable is a forecast table with point forecasts and distributions.

Visualising forecasts

```
brick_fc |>  
  autoplot(aus_production, level = NULL) +  
  labs(title = "Forecasts for quarterly clay brick production",  
       x = "Year", y = "Millions of bricks") +  
  guides(colour = guide_legend(title = "Forecast"))
```



Prediction intervals

```
brick_fc |> hilo(level = c(50, 75))
```

| # | .model | Quarter | Bricks | .mean | `50%` | `75%` |
|----|----------------|---------|--------------|-------|------------|------------|
| | <chr> | <qtr> | <dist> | <dbl> | <hilo> | <hilo> |
| 1 | Seasonal_naïve | 2005 Q3 | N(428, 2336) | 428 | [395, 461] | [372, 484] |
| 2 | Seasonal_naïve | 2005 Q4 | N(397, 2336) | 397 | [364, 430] | [341, 453] |
| 3 | Seasonal_naïve | 2006 Q1 | N(355, 2336) | 355 | [322, 388] | [299, 411] |
| 4 | Seasonal_naïve | 2006 Q2 | N(435, 2336) | 435 | [402, 468] | [379, 491] |
| 5 | Seasonal_naïve | 2006 Q3 | N(428, 4672) | 428 | [382, 474] | [349, 507] |
| 6 | Seasonal_naïve | 2006 Q4 | N(397, 4672) | 397 | [351, 443] | [318, 476] |
| 7 | Seasonal_naïve | 2007 Q1 | N(355, 4672) | 355 | [309, 401] | [276, 434] |
| 8 | Seasonal_naïve | 2007 Q2 | N(435, 4672) | 435 | [389, 481] | [356, 514] |
| 9 | Seasonal_naïve | 2007 Q3 | N(428, 7008) | 428 | [372, 484] | [332, 524] |
| 10 | Seasonal_naïve | 2007 Q4 | N(397, 7008) | 397 | [341, 453] | [301, 493] |

Prediction intervals

```
brick_fc |>  
  hilo(level = c(50, 75)) |>  
  unpack_hilo(c("50%", "75%"))
```

| # | A tsibble: 80 x 8 [1Q] | # Key: | .model | [4] | .model | Quarter | Bricks | .mean | `50%_lower` | `50%_upper` | `75%_lower` | `75%_upper` |
|----|---------------------------------|--------|--------|-----|--------|---------|--------------|-------|-------------|-------------|-------------|-------------|
| | | | | | <chr> | <qtr> | <dist> | <dbl> | <dbl> | <dbl> | <dbl> | <dbl> |
| 1 | Seasonal ~ 2005 Q3 N(428, 2336) | | | | 1 | 2005 Q3 | N(428, 2336) | 428 | 395. | 461. | 372. | |
| 2 | Seasonal ~ 2005 Q4 N(397, 2336) | | | | 2 | 2005 Q4 | N(397, 2336) | 397 | 364. | 430. | 341. | |
| 3 | Seasonal ~ 2006 Q1 N(355, 2336) | | | | 3 | 2006 Q1 | N(355, 2336) | 355 | 322. | 388. | 299. | |
| 4 | Seasonal ~ 2006 Q2 N(435, 2336) | | | | 4 | 2006 Q2 | N(435, 2336) | 435 | 402. | 468. | 379. | |
| 5 | Seasonal ~ 2006 Q3 N(428, 4672) | | | | 5 | 2006 Q3 | N(428, 4672) | 428 | 382. | 474. | 349. | |
| 6 | Seasonal ~ 2006 Q4 N(397, 4672) | | | | 6 | 2006 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 7 | Seasonal ~ 2007 Q1 N(355, 4672) | | | | 7 | 2007 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 8 | Seasonal ~ 2007 Q2 N(435, 4672) | | | | 8 | 2007 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 9 | Seasonal ~ 2007 Q3 N(428, 4672) | | | | 9 | 2007 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 10 | Seasonal ~ 2007 Q4 N(397, 4672) | | | | 10 | 2007 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 11 | Seasonal ~ 2008 Q1 N(355, 4672) | | | | 11 | 2008 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 12 | Seasonal ~ 2008 Q2 N(435, 4672) | | | | 12 | 2008 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 13 | Seasonal ~ 2008 Q3 N(428, 4672) | | | | 13 | 2008 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 14 | Seasonal ~ 2008 Q4 N(397, 4672) | | | | 14 | 2008 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 15 | Seasonal ~ 2009 Q1 N(355, 4672) | | | | 15 | 2009 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 16 | Seasonal ~ 2009 Q2 N(435, 4672) | | | | 16 | 2009 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 17 | Seasonal ~ 2009 Q3 N(428, 4672) | | | | 17 | 2009 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 18 | Seasonal ~ 2009 Q4 N(397, 4672) | | | | 18 | 2009 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 19 | Seasonal ~ 2010 Q1 N(355, 4672) | | | | 19 | 2010 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 20 | Seasonal ~ 2010 Q2 N(435, 4672) | | | | 20 | 2010 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 21 | Seasonal ~ 2010 Q3 N(428, 4672) | | | | 21 | 2010 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 22 | Seasonal ~ 2010 Q4 N(397, 4672) | | | | 22 | 2010 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 23 | Seasonal ~ 2011 Q1 N(355, 4672) | | | | 23 | 2011 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 24 | Seasonal ~ 2011 Q2 N(435, 4672) | | | | 24 | 2011 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 25 | Seasonal ~ 2011 Q3 N(428, 4672) | | | | 25 | 2011 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 26 | Seasonal ~ 2011 Q4 N(397, 4672) | | | | 26 | 2011 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 27 | Seasonal ~ 2012 Q1 N(355, 4672) | | | | 27 | 2012 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 28 | Seasonal ~ 2012 Q2 N(435, 4672) | | | | 28 | 2012 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 29 | Seasonal ~ 2012 Q3 N(428, 4672) | | | | 29 | 2012 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 30 | Seasonal ~ 2012 Q4 N(397, 4672) | | | | 30 | 2012 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 31 | Seasonal ~ 2013 Q1 N(355, 4672) | | | | 31 | 2013 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 32 | Seasonal ~ 2013 Q2 N(435, 4672) | | | | 32 | 2013 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 33 | Seasonal ~ 2013 Q3 N(428, 4672) | | | | 33 | 2013 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 34 | Seasonal ~ 2013 Q4 N(397, 4672) | | | | 34 | 2013 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 35 | Seasonal ~ 2014 Q1 N(355, 4672) | | | | 35 | 2014 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 36 | Seasonal ~ 2014 Q2 N(435, 4672) | | | | 36 | 2014 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 37 | Seasonal ~ 2014 Q3 N(428, 4672) | | | | 37 | 2014 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 38 | Seasonal ~ 2014 Q4 N(397, 4672) | | | | 38 | 2014 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 39 | Seasonal ~ 2015 Q1 N(355, 4672) | | | | 39 | 2015 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 40 | Seasonal ~ 2015 Q2 N(435, 4672) | | | | 40 | 2015 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 41 | Seasonal ~ 2015 Q3 N(428, 4672) | | | | 41 | 2015 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 42 | Seasonal ~ 2015 Q4 N(397, 4672) | | | | 42 | 2015 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 43 | Seasonal ~ 2016 Q1 N(355, 4672) | | | | 43 | 2016 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 44 | Seasonal ~ 2016 Q2 N(435, 4672) | | | | 44 | 2016 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 45 | Seasonal ~ 2016 Q3 N(428, 4672) | | | | 45 | 2016 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 46 | Seasonal ~ 2016 Q4 N(397, 4672) | | | | 46 | 2016 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 47 | Seasonal ~ 2017 Q1 N(355, 4672) | | | | 47 | 2017 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 48 | Seasonal ~ 2017 Q2 N(435, 4672) | | | | 48 | 2017 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 49 | Seasonal ~ 2017 Q3 N(428, 4672) | | | | 49 | 2017 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 50 | Seasonal ~ 2017 Q4 N(397, 4672) | | | | 50 | 2017 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 51 | Seasonal ~ 2018 Q1 N(355, 4672) | | | | 51 | 2018 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 52 | Seasonal ~ 2018 Q2 N(435, 4672) | | | | 52 | 2018 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 53 | Seasonal ~ 2018 Q3 N(428, 4672) | | | | 53 | 2018 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 54 | Seasonal ~ 2018 Q4 N(397, 4672) | | | | 54 | 2018 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 55 | Seasonal ~ 2019 Q1 N(355, 4672) | | | | 55 | 2019 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 56 | Seasonal ~ 2019 Q2 N(435, 4672) | | | | 56 | 2019 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 57 | Seasonal ~ 2019 Q3 N(428, 4672) | | | | 57 | 2019 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 58 | Seasonal ~ 2019 Q4 N(397, 4672) | | | | 58 | 2019 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 59 | Seasonal ~ 2020 Q1 N(355, 4672) | | | | 59 | 2020 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 60 | Seasonal ~ 2020 Q2 N(435, 4672) | | | | 60 | 2020 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 61 | Seasonal ~ 2020 Q3 N(428, 4672) | | | | 61 | 2020 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 62 | Seasonal ~ 2020 Q4 N(397, 4672) | | | | 62 | 2020 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 63 | Seasonal ~ 2021 Q1 N(355, 4672) | | | | 63 | 2021 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 64 | Seasonal ~ 2021 Q2 N(435, 4672) | | | | 64 | 2021 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 65 | Seasonal ~ 2021 Q3 N(428, 4672) | | | | 65 | 2021 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 66 | Seasonal ~ 2021 Q4 N(397, 4672) | | | | 66 | 2021 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 67 | Seasonal ~ 2022 Q1 N(355, 4672) | | | | 67 | 2022 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 68 | Seasonal ~ 2022 Q2 N(435, 4672) | | | | 68 | 2022 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 69 | Seasonal ~ 2022 Q3 N(428, 4672) | | | | 69 | 2022 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 70 | Seasonal ~ 2022 Q4 N(397, 4672) | | | | 70 | 2022 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 71 | Seasonal ~ 2023 Q1 N(355, 4672) | | | | 71 | 2023 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 72 | Seasonal ~ 2023 Q2 N(435, 4672) | | | | 72 | 2023 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 73 | Seasonal ~ 2023 Q3 N(428, 4672) | | | | 73 | 2023 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 74 | Seasonal ~ 2023 Q4 N(397, 4672) | | | | 74 | 2023 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 75 | Seasonal ~ 2024 Q1 N(355, 4672) | | | | 75 | 2024 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 76 | Seasonal ~ 2024 Q2 N(435, 4672) | | | | 76 | 2024 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 77 | Seasonal ~ 2024 Q3 N(428, 4672) | | | | 77 | 2024 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 78 | Seasonal ~ 2024 Q4 N(397, 4672) | | | | 78 | 2024 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 79 | Seasonal ~ 2025 Q1 N(355, 4672) | | | | 79 | 2025 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 80 | Seasonal ~ 2025 Q2 N(435, 4672) | | | | 80 | 2025 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 81 | Seasonal ~ 2025 Q3 N(428, 4672) | | | | 81 | 2025 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 82 | Seasonal ~ 2025 Q4 N(397, 4672) | | | | 82 | 2025 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 83 | Seasonal ~ 2026 Q1 N(355, 4672) | | | | 83 | 2026 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 84 | Seasonal ~ 2026 Q2 N(435, 4672) | | | | 84 | 2026 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 85 | Seasonal ~ 2026 Q3 N(428, 4672) | | | | 85 | 2026 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 86 | Seasonal ~ 2026 Q4 N(397, 4672) | | | | 86 | 2026 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 87 | Seasonal ~ 2027 Q1 N(355, 4672) | | | | 87 | 2027 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 88 | Seasonal ~ 2027 Q2 N(435, 4672) | | | | 88 | 2027 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 89 | Seasonal ~ 2027 Q3 N(428, 4672) | | | | 89 | 2027 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 90 | Seasonal ~ 2027 Q4 N(397, 4672) | | | | 90 | 2027 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 91 | Seasonal ~ 2028 Q1 N(355, 4672) | | | | 91 | 2028 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 92 | Seasonal ~ 2028 Q2 N(435, 4672) | | | | 92 | 2028 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 93 | Seasonal ~ 2028 Q3 N(428, 4672) | | | | 93 | 2028 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 94 | Seasonal ~ 2028 Q4 N(397, 4672) | | | | 94 | 2028 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 95 | Seasonal ~ 2029 Q1 N(355, 4672) | | | | 95 | 2029 Q1 | N(355, 4672) | 355 | 309. | 401. | 276. | |
| 96 | Seasonal ~ 2029 Q2 N(435, 4672) | | | | 96 | 2029 Q2 | N(435, 4672) | 435 | 389. | 481. | 356. | |
| 97 | Seasonal ~ 2029 Q3 N(428, 4672) | | | | 97 | 2029 Q3 | N(428, 4672) | 428 | 376. | 464. | 333. | |
| 98 | Seasonal ~ 2029 Q4 N(397, 4672) | | | | 98 | 2029 Q4 | N(397, 4672) | 397 | 351. | 443. | 318. | |
| 99 | Seasonal ~ 2030 Q1 N(355, 4672) | | | | 99 | 2030 Q1 | N(355, 4672) | 35 | | | | |

Outline

1 Statistical forecasting

2 Benchmark methods

3 Lab Session 11

4 Residual diagnostics

5 Lab Session 12

6 Forecast accuracy measures

7 Lab Session 13

Lab Session 11

- Produce forecasts using an appropriate benchmark method for household wealth (`hh_budget`). Plot the results using `autoplot()`.
- Produce forecasts using an appropriate benchmark method for Australian takeaway food turnover (`aus_retail`). Plot the results using `autoplot()`.

Outline

1 Statistical forecasting

2 Benchmark methods

3 Lab Session 11

4 Residual diagnostics

5 Lab Session 12

6 Forecast accuracy measures

7 Lab Session 13

Fitted values

- $\hat{y}_{t|t-1}$ is the forecast of y_t based on observations y_1, \dots, y_{t-1} .
- We call these “fitted values”.
- Sometimes drop the subscript: $\hat{y}_t \equiv \hat{y}_{t|t-1}$.
- Often not true forecasts since parameters are estimated on all data.

For example:

- $\hat{y}_t = \bar{y}$ for average method.
- $\hat{y}_t = y_{t-1} + (y_T - y_1)/(T - 1)$ for drift method.

Forecasting residuals

Residuals in forecasting: difference between observed value and its fitted value: $e_t = y_t - \hat{y}_{t|t-1}$.

Forecasting residuals

Residuals in forecasting: difference between observed value and its fitted value: $e_t = y_t - \hat{y}_{t|t-1}$.

Assumptions

- 1 $\{e_t\}$ uncorrelated. If they aren't, then information left in residuals that should be used in computing forecasts.
- 2 $\{e_t\}$ have mean zero. If they don't, then forecasts are biased.

Forecasting residuals

Residuals in forecasting: difference between observed value and its fitted value: $e_t = y_t - \hat{y}_{t|t-1}$.

Assumptions

- 1 $\{e_t\}$ uncorrelated. If they aren't, then information left in residuals that should be used in computing forecasts.
- 2 $\{e_t\}$ have mean zero. If they don't, then forecasts are biased.

Useful properties (for prediction intervals)

- 3 $\{e_t\}$ have constant variance.
- 4 $\{e_t\}$ are normally distributed.

Facebook closing stock price

```
fb_stock <- gafa_stock |>  
  filter(Symbol == "FB")  
fb_stock |> autoplot(Close)
```



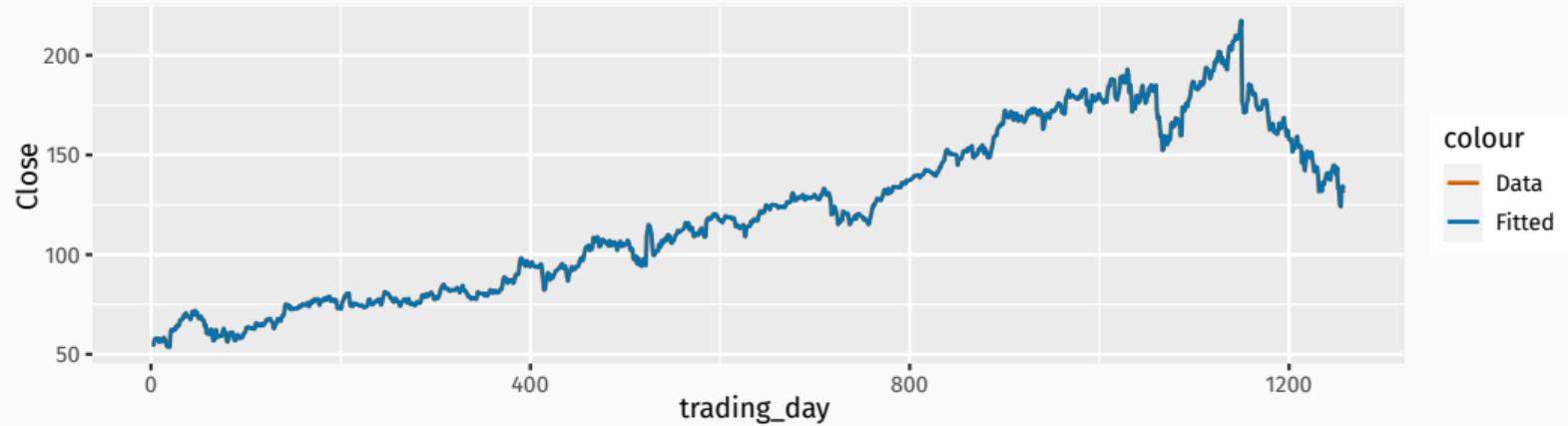
Facebook closing stock price

```
fb_stock <- fb_stock |>  
  mutate(trading_day = row_number()) |>  
  update_tsibble(index = trading_day, regular = TRUE)  
fit <- fb_stock |> model(NAIVE(Close))  
augment(fit)
```

```
# A tsibble: 1,258 x 7 [1]  
# Key:     Symbol, .model [1]  
  Symbol .model      trading_day Close .fitted .resid .innov  
  <chr>  <chr>        <int> <dbl>    <dbl>   <dbl>   <dbl>  
1 FB     NAIVE(Close)      1  54.7     NA     NA     NA  
2 FB     NAIVE(Close)      2  54.6    54.7 -0.150 -0.150  
3 FB     NAIVE(Close)      3  57.2    54.6   2.64   2.64  
4 FB     NAIVE(Close)      4  57.9    57.2   0.720  0.720  
5 FB     NAIVE(Close)      5  58.2    57.9   0.310  0.310  
6 FB     NAIVE(Close)      6  57.2    58.2  -1.01  -1.01  
7 FB     NAIVE(Close)      7  57.9    57.2   0.720  0.720  
8 FB     NAIVE(Close)      8  55.9    57.9  -2.03  -2.03  
9 FB     NAIVE(Close)      9  57.7    55.9   1.83   1.83
```

Facebook closing stock price

```
augment(fit) |>  
  ggplot(aes(x = trading_day)) +  
  geom_line(aes(y = Close, colour = "Data")) +  
  geom_line(aes(y = .fitted, colour = "Fitted"))
```



Facebook closing stock price

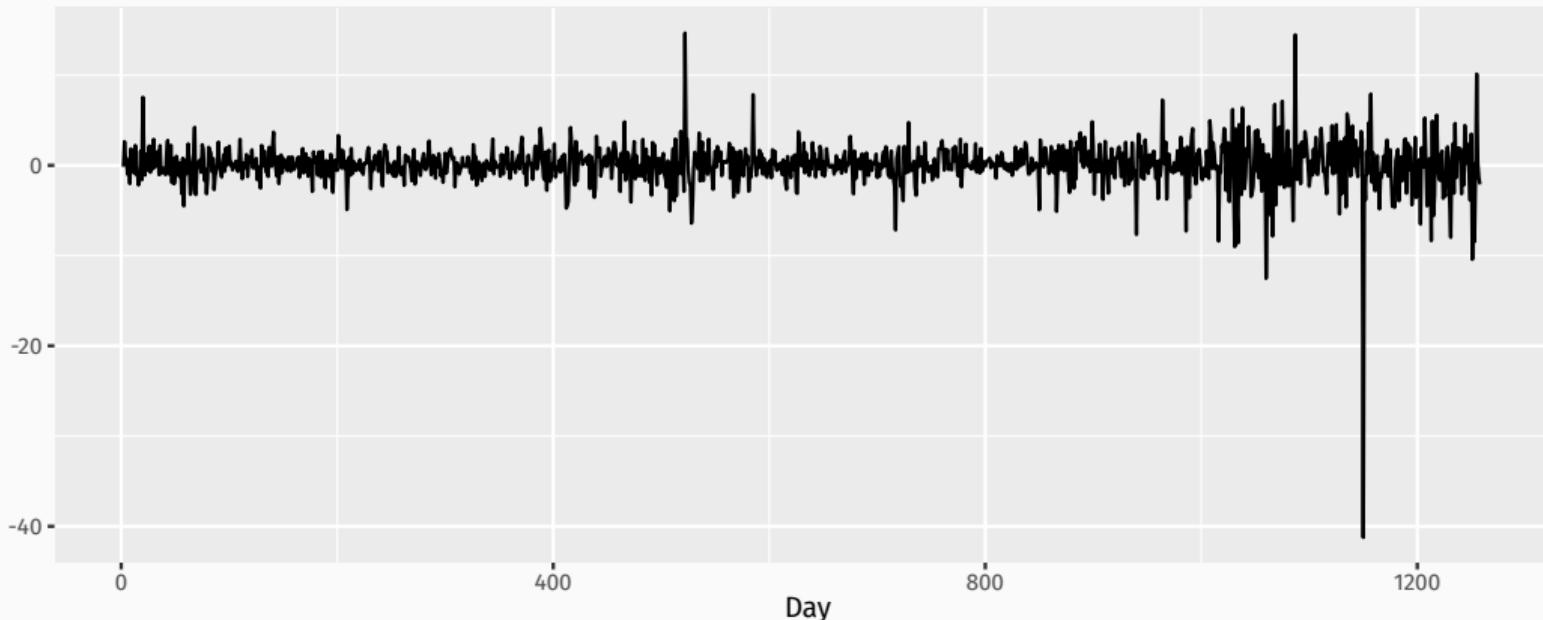
```
augment(fit) |>  
  filter(trading_day > 1100) |>  
  ggplot(aes(x = trading_day)) +  
  geom_line(aes(y = Close, colour = "Data")) +  
  geom_line(aes(y = .fitted, colour = "Fitted"))
```



Facebook closing stock price

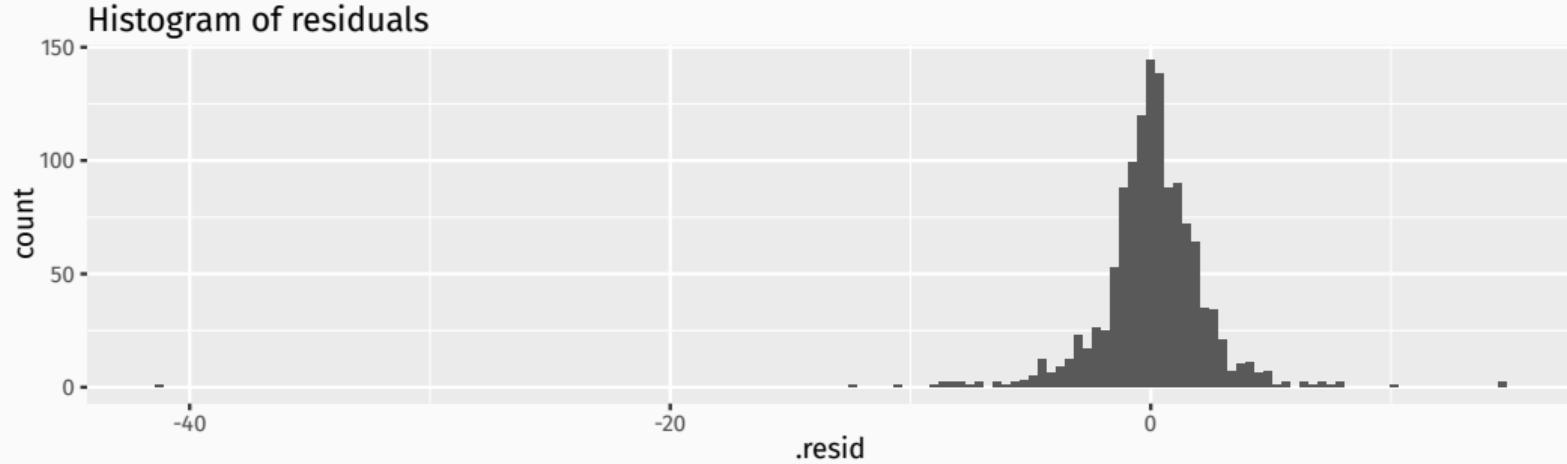
```
augment(fit) |>  
  autoplot(.resid) +  
  labs(x = "Day", y = "", title = "Residuals from naïve method")
```

Residuals from naïve method



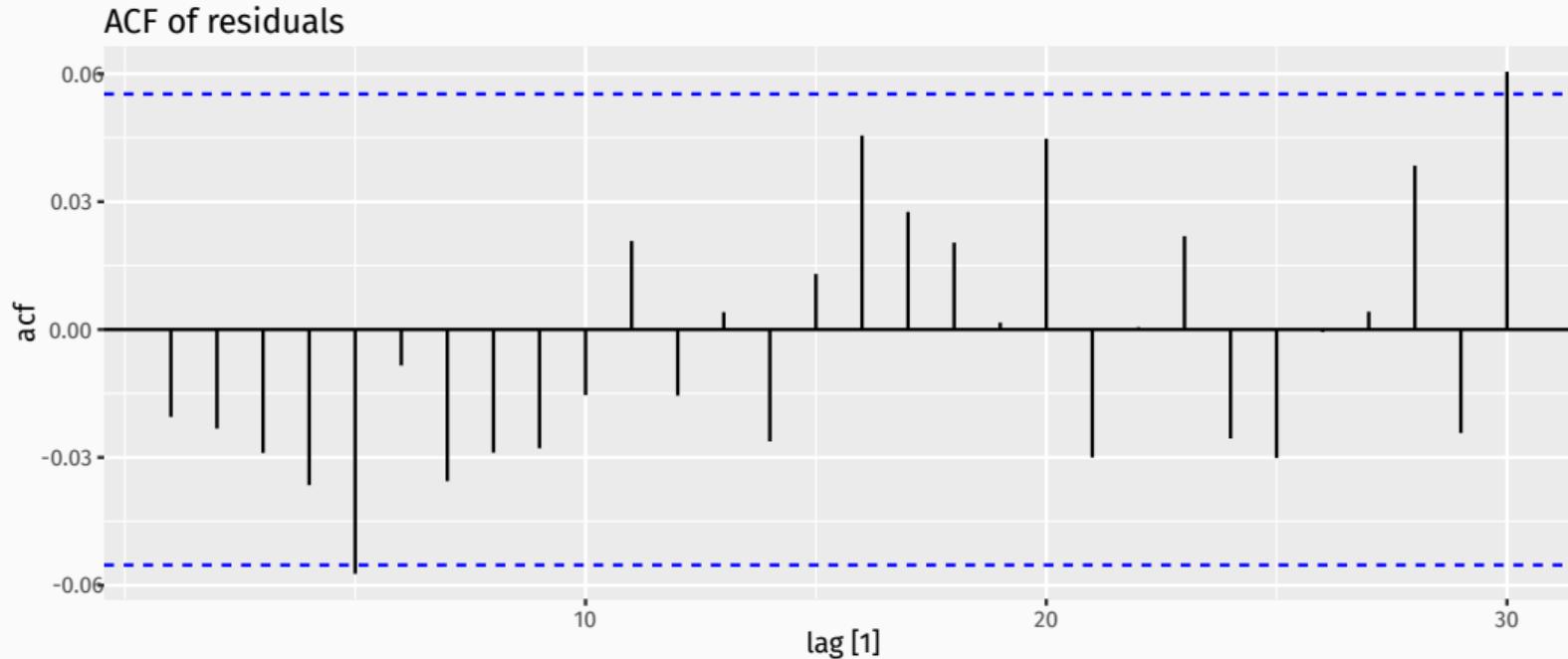
Facebook closing stock price

```
augment(fit) |>  
  ggplot(aes(x = .resid)) +  
  geom_histogram(bins = 150) +  
  labs(title = "Histogram of residuals")
```



Facebook closing stock price

```
augment(fit) |>  
ACF(.resid) |>  
autoplot() + labs(title = "ACF of residuals")
```

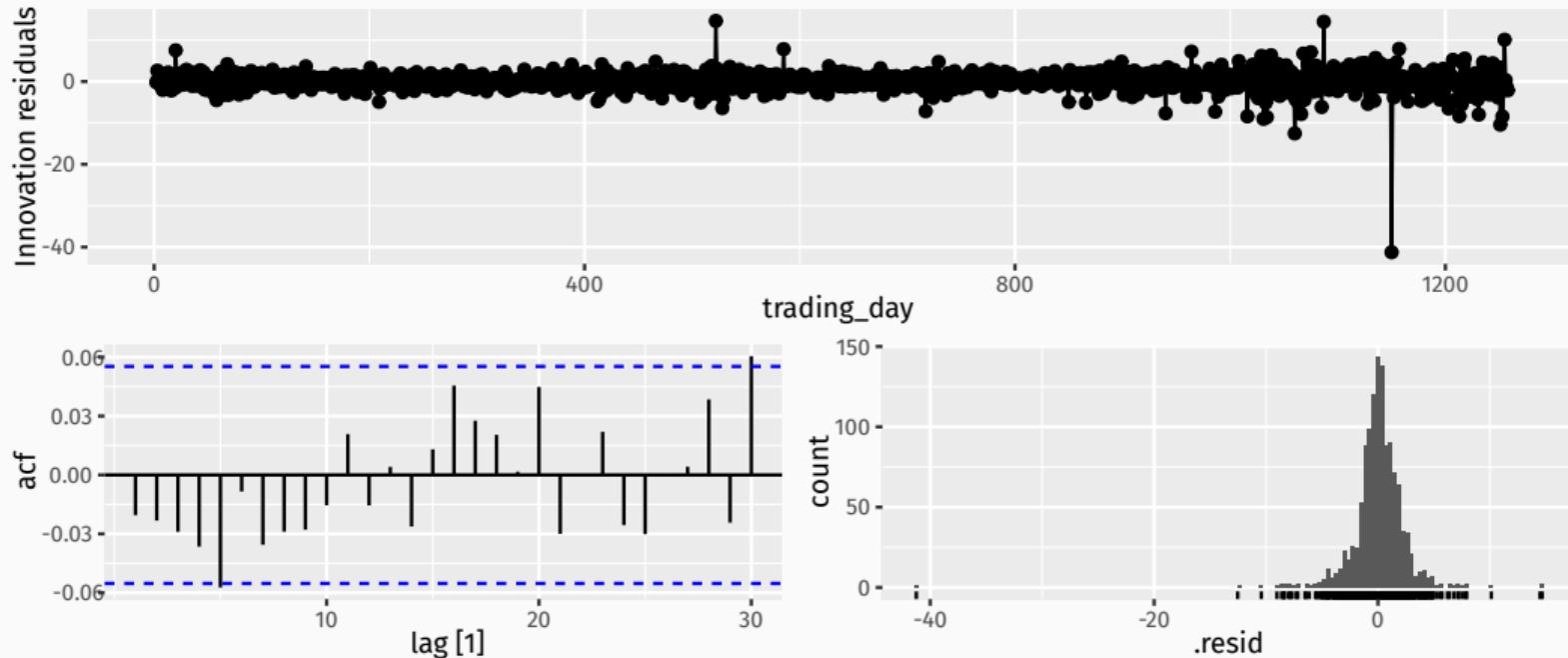


ACF of residuals

- We assume that the residuals are white noise (uncorrelated, mean zero, constant variance). If they aren't, then there is information left in the residuals that should be used in computing forecasts.
- So a standard residual diagnostic is to check the ACF of the residuals of a forecasting method.
- We *expect* these to look like white noise.

Combined diagnostic graph

```
fit |> gg_tsresiduals()
```



Ljung-Box test

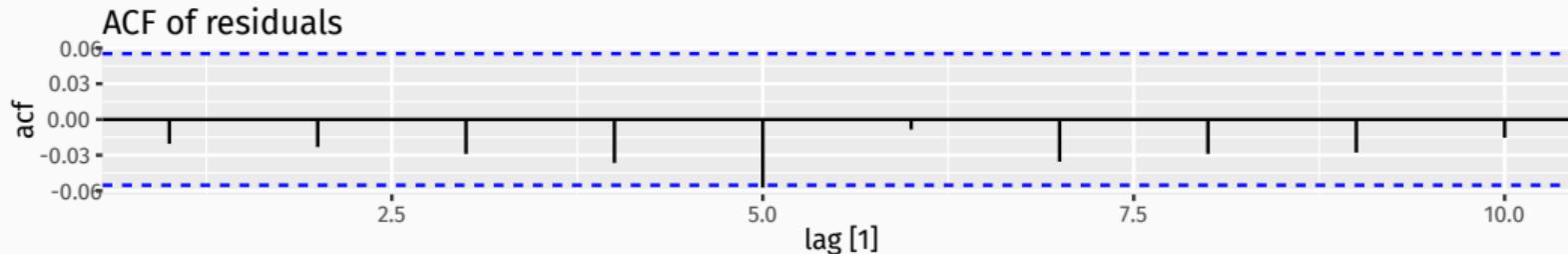
Test whether *whole set* of r_k values is significantly different from zero set.

$$Q = T(T + 2) \sum_{k=1}^h (T - k)^{-1} r_k^2 \quad \text{where } h = \max \text{ lag and } T = \# \text{ observations.}$$

- If each r_k close to zero, Q will be **small**.
- If some r_k values large (+ or -), Q will be **large**.
- My preferences: $h = 10$ for non-seasonal data, $h = 2m$ for seasonal data.
- If data are WN, $Q \sim \chi^2$ with $(h - K)$ degrees of freedom where K = no. parameters in model.
- When applied to raw data, set $K = 0$.

Ljung-Box test

$$Q = T(T + 2) \sum_{k=1}^h (T - k)^{-1} r_k^2 \quad \text{where } h = \max \text{ lag and } T = \# \text{ observations.}$$



```
# lag=h and dof=K
augment(fit) |> features(.resid, ljung_box, dof = 0, lag = 10)
```

```
# A tibble: 1 x 4
  Symbol .model      lb_stat lb_pvalue
  <chr>  <chr>       <dbl>     <dbl>
1 FB     NAIVE(Close) 12.1      0.276
```

Outline

1 Statistical forecasting

2 Benchmark methods

3 Lab Session 11

4 Residual diagnostics

5 Lab Session 12

6 Forecast accuracy measures

7 Lab Session 13

Lab Session 12

- Compute seasonal naïve forecasts for quarterly Australian beer production.
- Test if the residuals are white noise. What do you conclude?

Outline

- 1 Statistical forecasting
- 2 Benchmark methods
- 3 Lab Session 11
- 4 Residual diagnostics
- 5 Lab Session 12
- 6 Forecast accuracy measures
- 7 Lab Session 13

Training and test sets



- A model which fits the training data well will not necessarily forecast well.
- Forecast accuracy is based only on the test set.

Forecast errors

Forecast “error”: the difference between an observed value and its forecast.

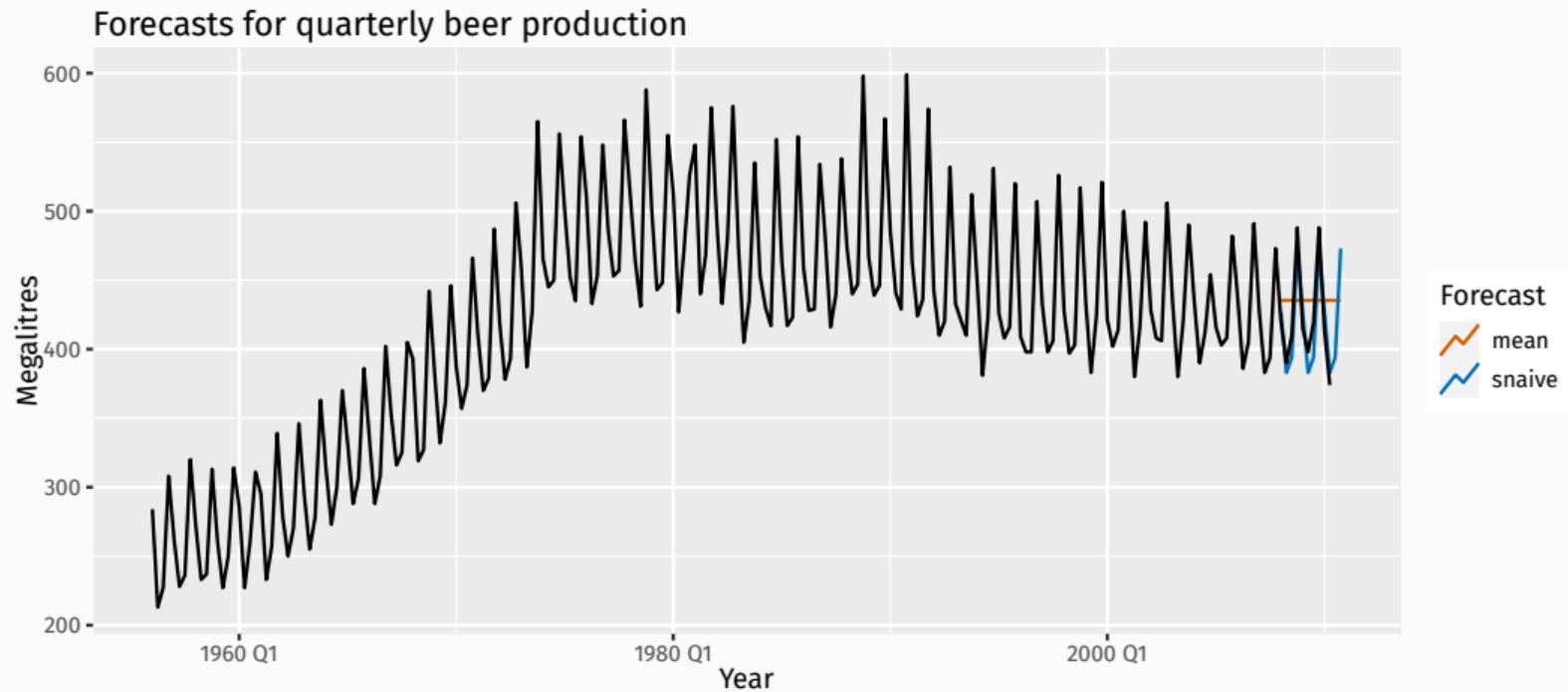
$$e_{T+h} = y_{T+h} - \hat{y}_{T+h|T},$$

where the training data is given by $\{y_1, \dots, y_T\}$

Measures of forecast accuracy

```
beer_fit <- aus_production |>
  filter(between(year(Quarter), 1992, 2007)) |>
  model(
    snaive = SNAIVE(Beer),
    mean = MEAN(Beer)
  )
beer_fit |>
  forecast(h = "3 years") |>
  autoplot(aus_production, level = NULL) +
  labs(title ="Forecasts for quarterly beer production",
       x ="Year", y ="Megalitres") +
  guides(colour = guide_legend(title = "Forecast"))
```

Measures of forecast accuracy



Measures of forecast accuracy

y_{T+h} = $(T + h)$ th observation, $h = 1, \dots, H$

$\hat{y}_{T+h|T}$ = its forecast based on data up to time T .

$e_{T+h} = y_{T+h} - \hat{y}_{T+h|T}$

MAE = $\text{mean}(|e_{T+h}|)$

MSE = $\text{mean}(e_{T+h}^2)$

RMSE = $\sqrt{\text{mean}(e_{T+h}^2)}$

MAPE = $100\text{mean}(|e_{T+h}|/|y_{T+h}|)$

Measures of forecast accuracy

y_{T+h} = $(T + h)$ th observation, $h = 1, \dots, H$

$\hat{y}_{T+h|T}$ = its forecast based on data up to time T .

$e_{T+h} = y_{T+h} - \hat{y}_{T+h|T}$

MAE = $\text{mean}(|e_{T+h}|)$

MSE = $\text{mean}(e_{T+h}^2)$

RMSE = $\sqrt{\text{mean}(e_{T+h}^2)}$

MAPE = $100\text{mean}(|e_{T+h}|/|y_{T+h}|)$

- MAE, MSE, RMSE are all scale dependent.
- MAPE is scale independent but is only sensible if $y_t \gg 0$

Measures of forecast accuracy

Mean Absolute Scaled Error

$$\text{MASE} = \text{mean}(|e_{T+h}| / Q)$$

where Q is a stable measure of the scale of the time series $\{y_t\}$.

Proposed by Hyndman and Koehler (IJF, 2006).

For non-seasonal time series,

$$Q = (T - 1)^{-1} \sum_{t=2}^T |y_t - y_{t-1}|$$

Measures of forecast accuracy

Mean Absolute Scaled Error

$$\text{MASE} = \text{mean}(|e_{T+h}|/Q)$$

where Q is a stable measure of the scale of the time series $\{y_t\}$.

Proposed by Hyndman and Koehler (IJF, 2006).

For seasonal time series,

$$Q = (T - m)^{-1} \sum_{t=m+1}^T |y_t - y_{t-m}|$$

Measures of forecast accuracy

```
beer_fc <- forecast(beer_fit, h = "3 years")
accuracy(beer_fc, aus_production)
```

```
# A tibble: 2 x 10
  .model .type     ME   RMSE    MAE    MPE   MAPE   MASE RMSSE     ACF1
  <chr>  <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
1 mean    Test   -13.8  38.4  34.8 -3.97  8.28  2.20  1.96 -0.0691
2 snaive  Test     5.2  14.3  13.4  1.15  3.17  0.847 0.729  0.132
```

Outline

1 Statistical forecasting

2 Benchmark methods

3 Lab Session 11

4 Residual diagnostics

5 Lab Session 12

6 Forecast accuracy measures

7 Lab Session 13

Lab Session 13

- Create a training set for household wealth (`hh_budget`) by withholding the last four years as a test set.
- Fit all the appropriate benchmark methods to the training set and forecast the periods covered by the test set.
- Compute the accuracy of your forecasts. Which method does best?
- Repeat the exercise using the Australian takeaway food turnover data (`aus_retail`) with a test set of four years.