https://workshop.f4sg.org/africast/

Africast-Time Series Analysis & Forecasting Using R

5. Basic modeling and forecasting

Outline

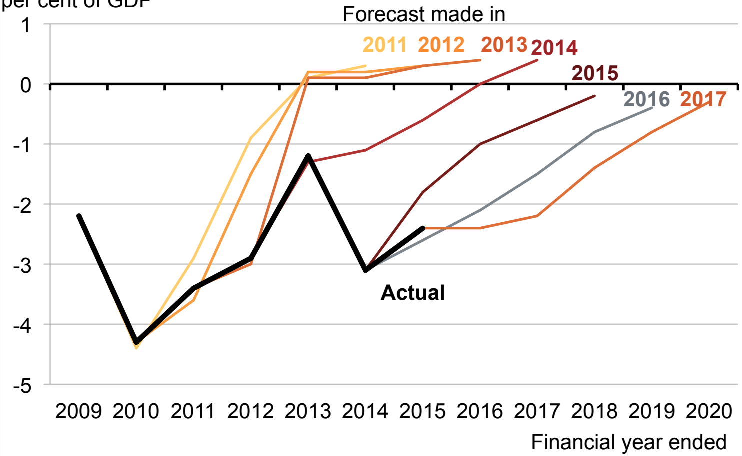
Outline

Forecasting is difficult

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

GRATTAN Institute

Actual and forecast Commonwealth underlying cash balance
per cent of GDP



Grattan analysis of Commonwealth Budget Papers

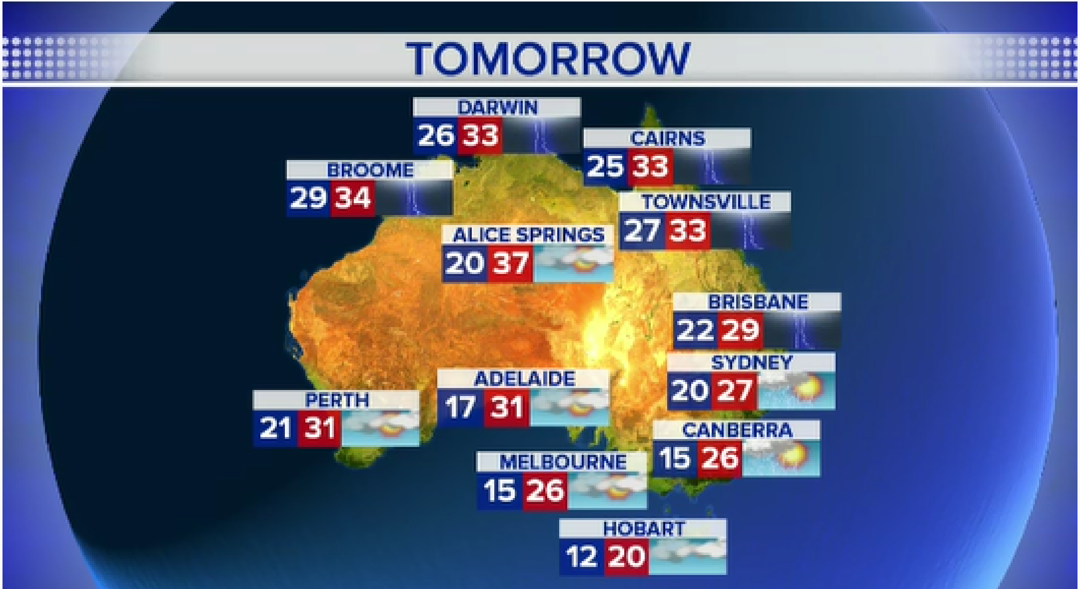
What can we forecast?

What can we forecast?

What can we forecast?

What can we forecast?

What can we forecast?



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:

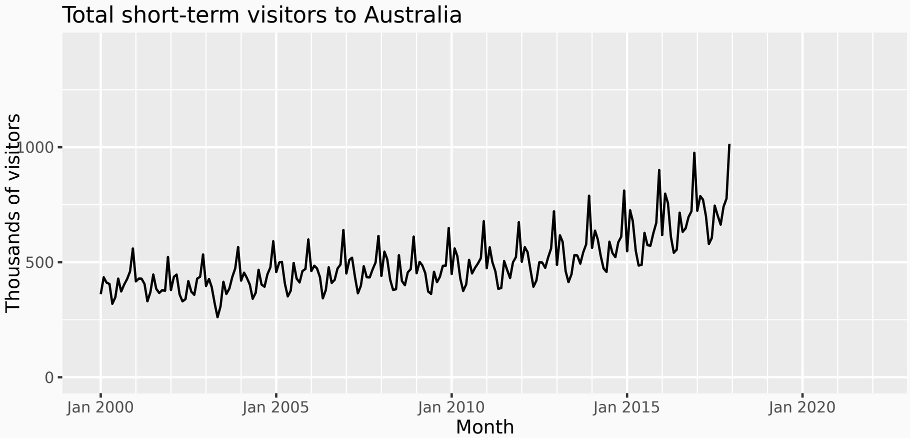
- we have a good understanding of the factors that contribute to it
- there is lots of data available;
- the forecasts cannot affect the thing we are trying to forecast.
- there is relatively low natural/unexplainable random variation.
- the future is somewhat similar to the past

Random futures

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

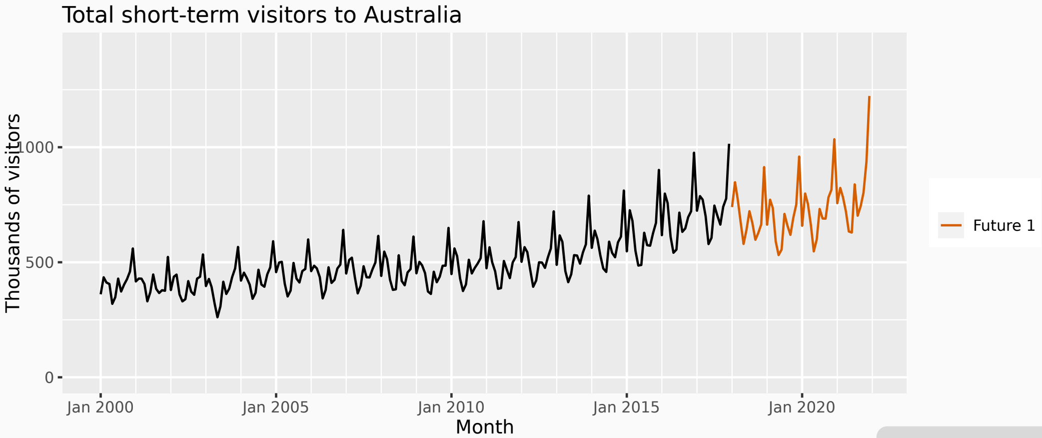
Random futures

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



Random futures

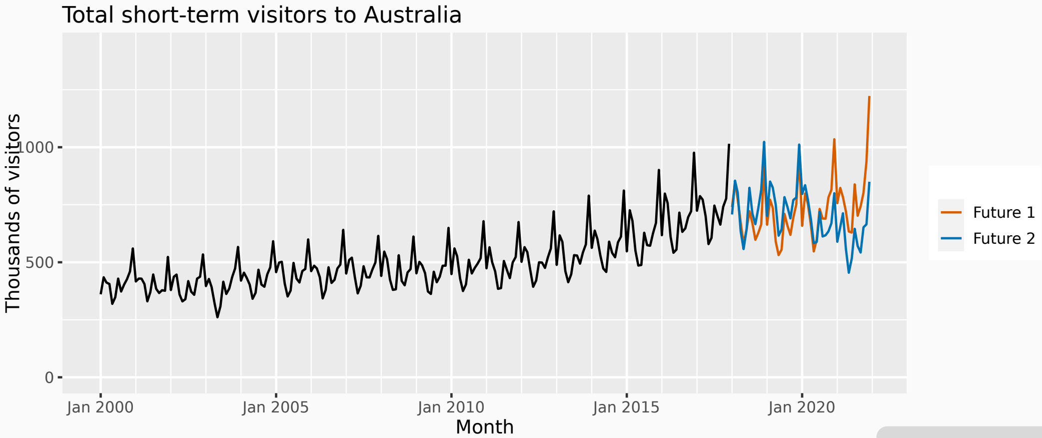
A forecast is an estimate of the probabilities of possible futures.



Simulated futures from an ETS

Random futures

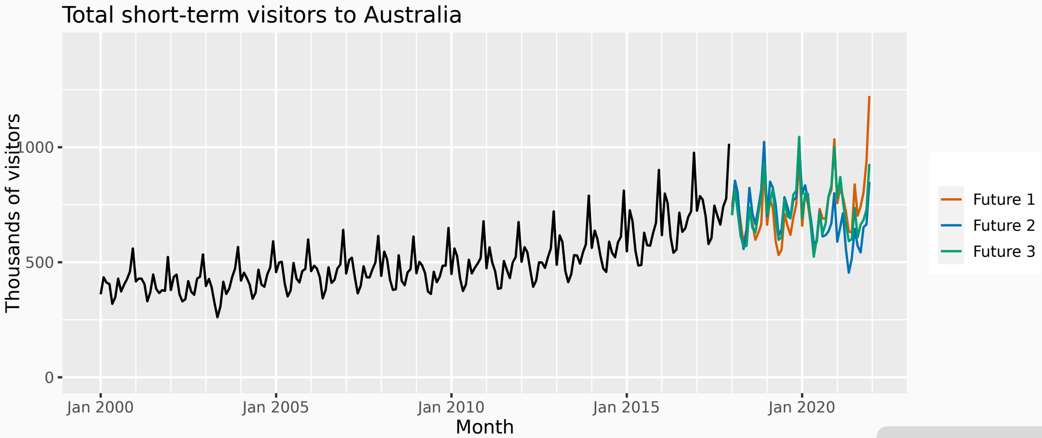
A forecast is an estimate of the probabilities of possible futures.



Simulated futures from an ETS

Random futures

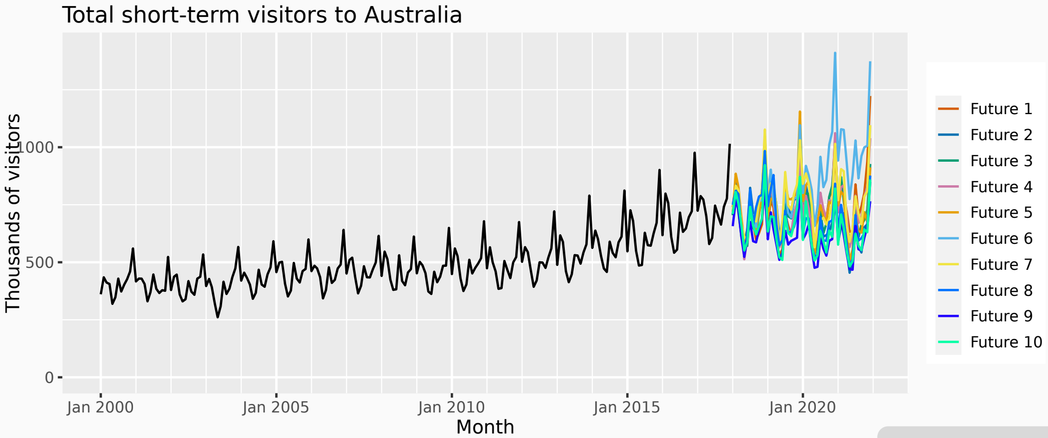
A forecast is an estimate of the probabilities of possible futures.



Simulated futures from an ETS

Random futures

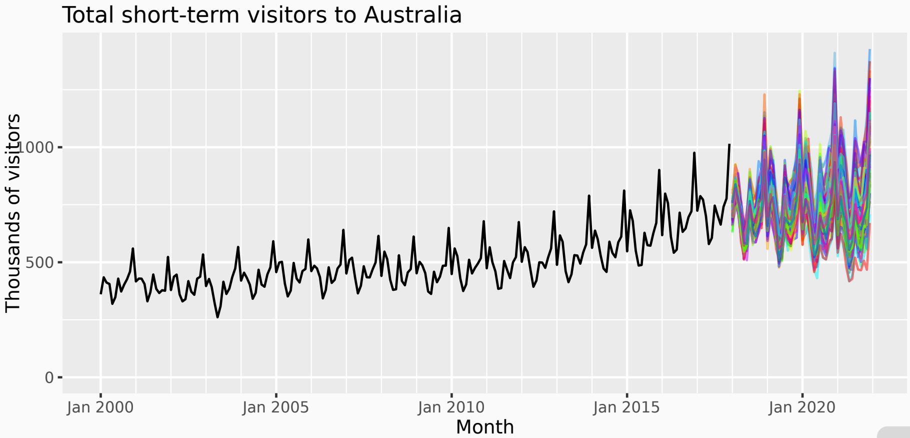
A forecast is an estimate of the probabilities of possible futures.



Simulated futures from an ETS

Random futures

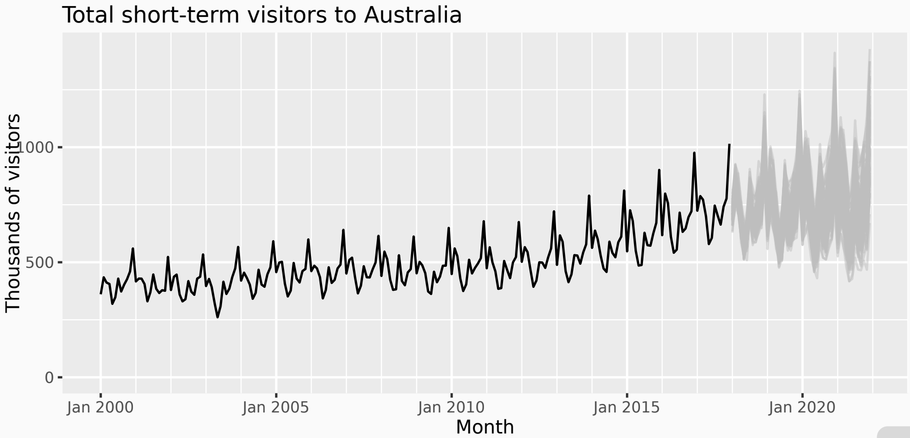
A forecast is an estimate of the probabilities of possible futures.



Simulated futures from an ETS

Random futures

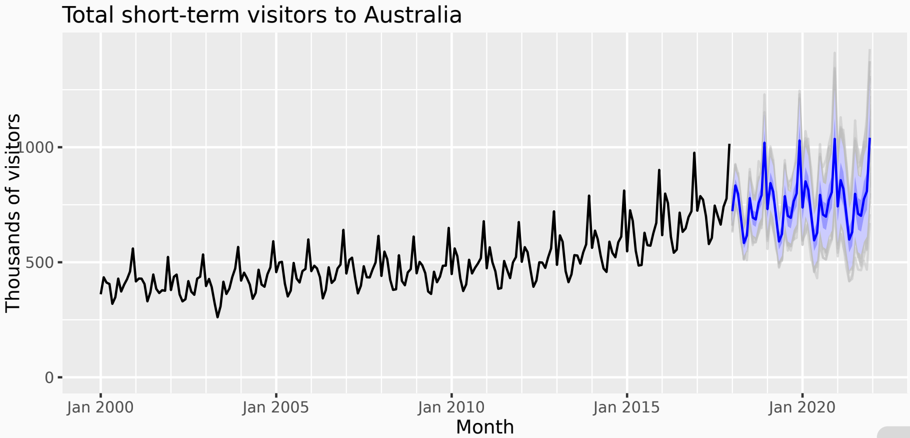
A forecast is an estimate of the probabilities of possible futures.



Simulated futures from an ETS

Random futures

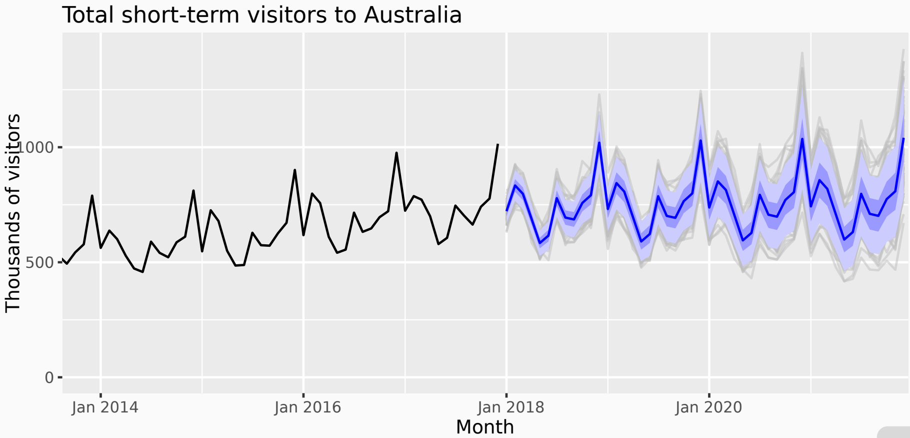
A forecast is an estimate of the probabilities of possible futures.



Simulated futures from an ETS

Random futures

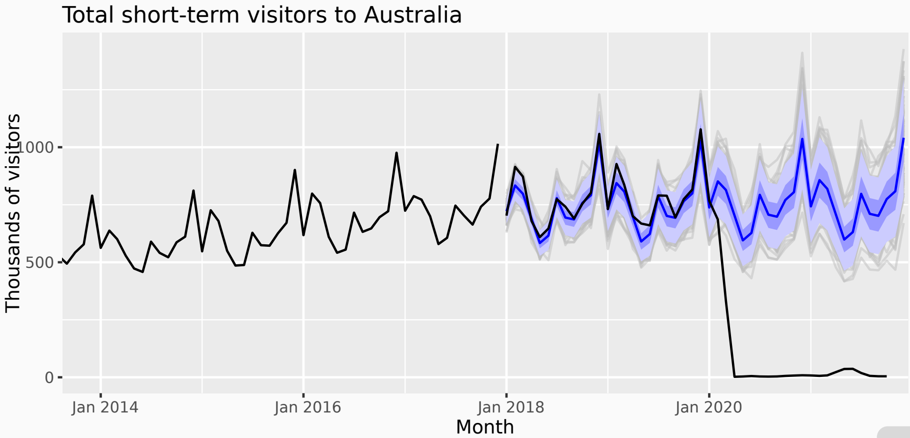
A forecast is an estimate of the probabilities of possible futures.



Simulated futures from an ETS

Random futures

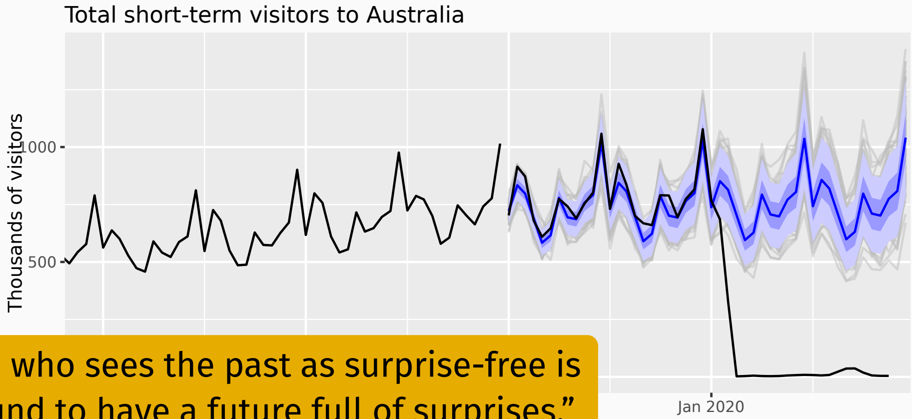
A forecast is an estimate of the probabilities of possible futures.



Simulated futures from an ETS

Random futures

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



"He who sees the past as surprise-free is bound to have a future full of surprises."
(Amos Tversky)

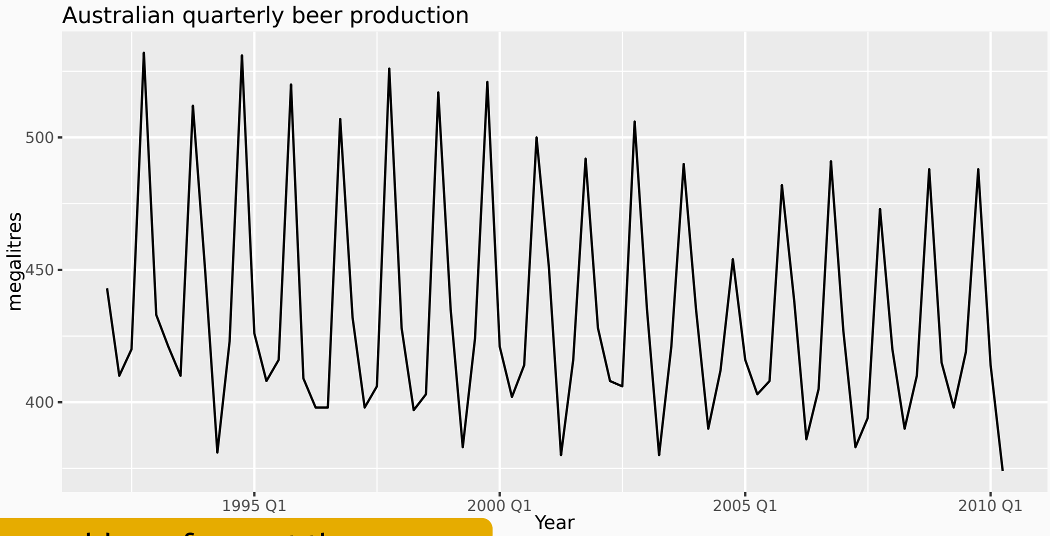
Simulated futures from an ETS

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: $\hat{y}_{T+h|T} = \text{E}[y_{T+h} \mid y_1, \dots, y_T]$.
- 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.

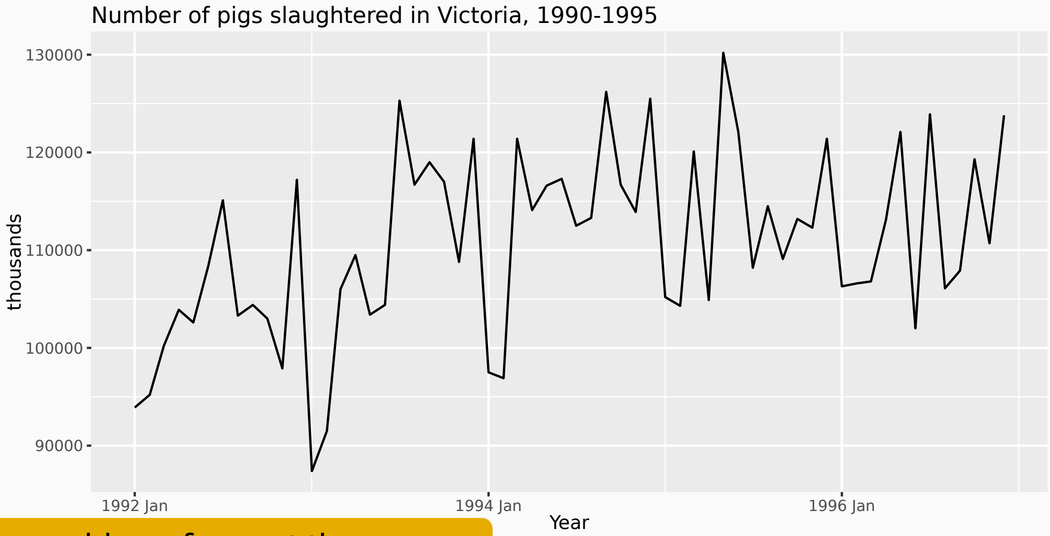
Outline

Some simple forecasting methods



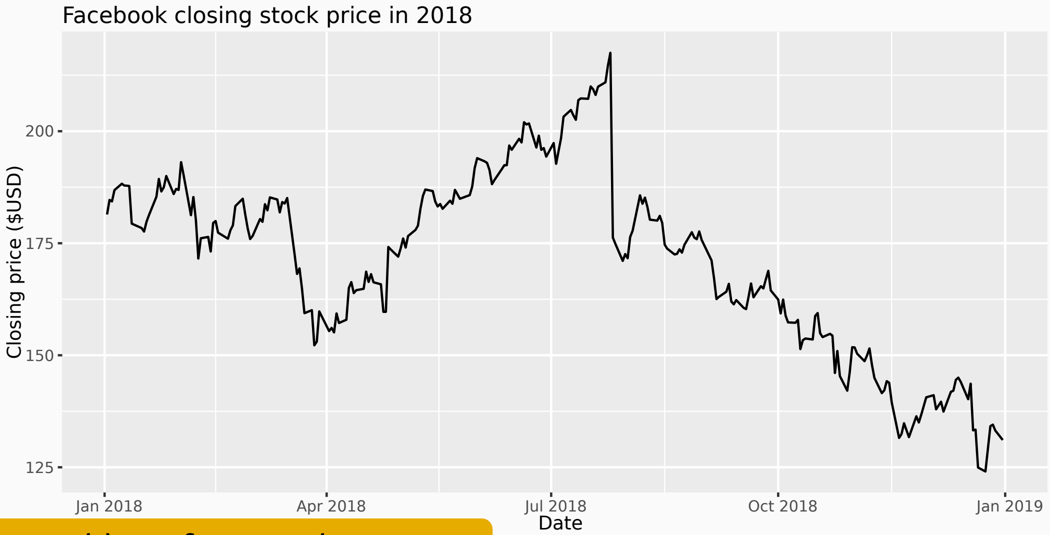
How would you forecast these series?

Some simple forecasting methods



How would you forecast these series?

Some simple forecasting methods

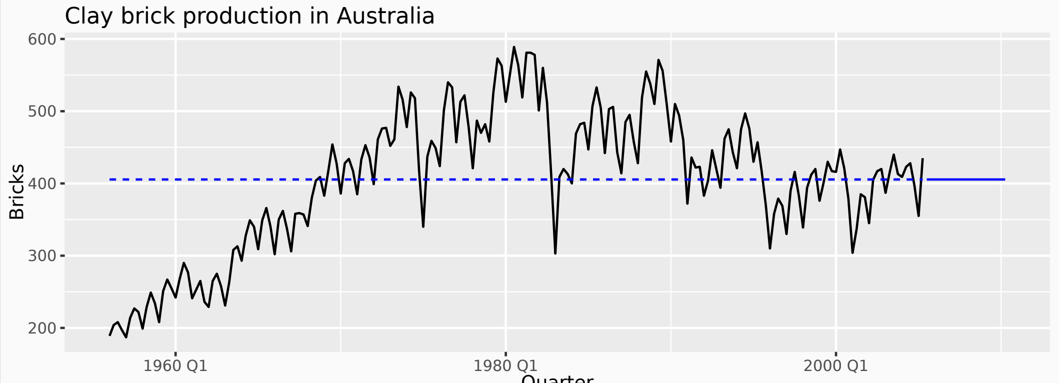


How would you forecast these series?

Some simple forecasting methods

`MEAN(y)`: Average method

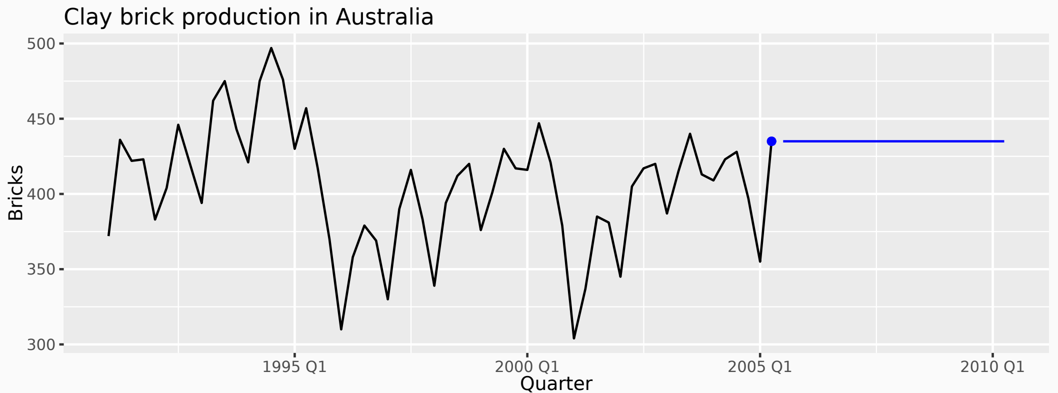
- Forecast of all future values is equal to mean of historical data $\{y_1, \dots, y_T\}$.
- Forecasts: $\hat{y}_{T+h|T} = \bar{y} = (y_1 + \dots + y_T)/T$



Some simple forecasting methods

`NAIVE(y)`: Naïve method

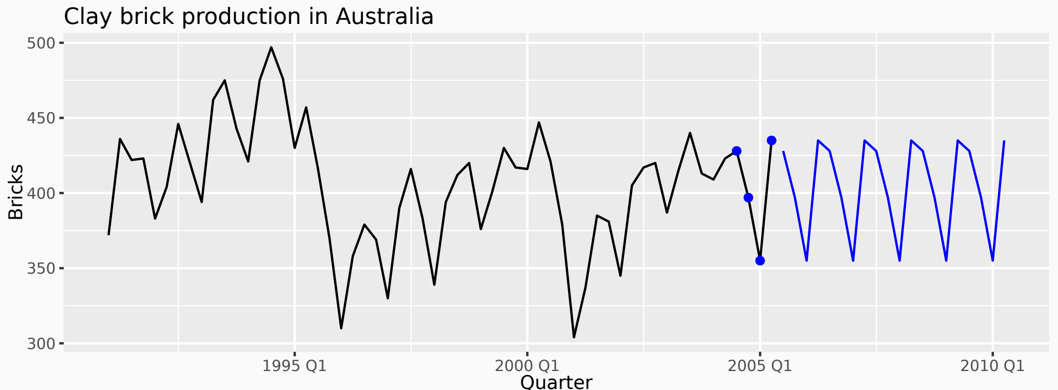
- Forecasts equal to last observed value.
- Forecasts: $\hat{y}_{T+h|T} = y_T$.
- Consequence of efficient market hypothesis.



Some simple forecasting methods

`SNAIVE(y ~ 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

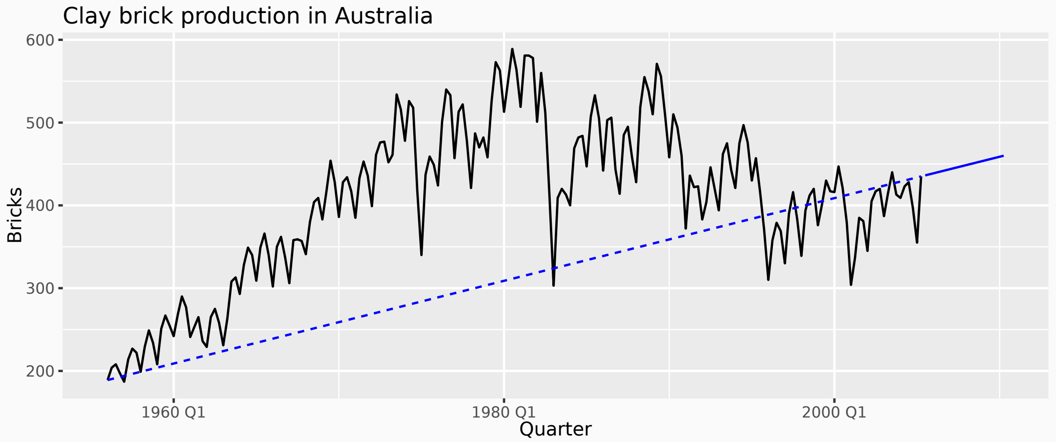
- Forecasts equal to last value plus average change.
- Forecasts:

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

- 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

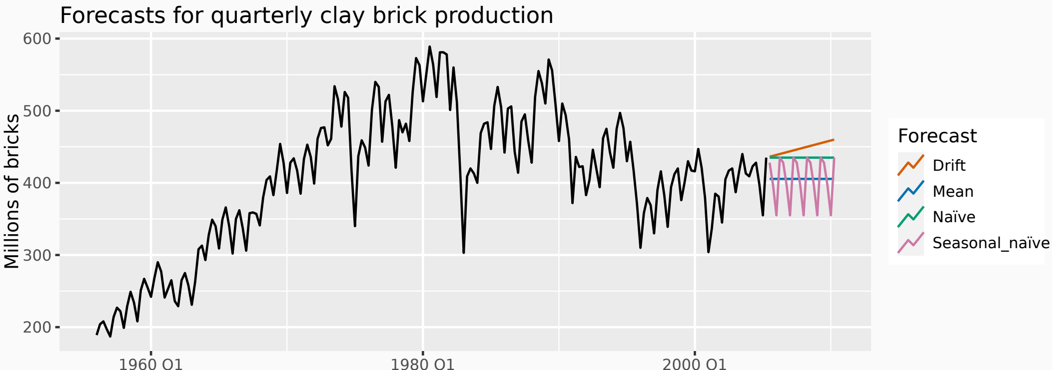
```r
brick_fc <- brick_fit |>
  forecast(h = "5 years")
```

```
# 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

```r
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))
```

```
# A tsibble: 80 x 6 [1Q]
# Key:       .model [4]
   .model         Quarter        Bricks .mean              `50%`            `75%`
   <chr>            <qtr>        <dist> <dbl>             <hilo>           <hilo>
 1 Seasonal_naïve 2005 Q3 N(428, 2336)   428 [395, 461]50 [372, 484]75
 2 Seasonal_naïve 2005 Q4 N(397, 2336)   397 [364, 430]50 [341, 453]75
 3 Seasonal_naïve 2006 Q1 N(355, 2336)   355 [322, 388]50 [299, 411]75
 4 Seasonal_naïve 2006 Q2 N(435, 2336)   435 [402, 468]50 [379, 491]75
 5 Seasonal_naïve 2006 Q3 N(428, 4672)   428 [382, 474]50 [349, 507]75
 6 Seasonal_naïve 2006 Q4 N(397, 4672)   397 [351, 443]50 [318, 476]75
 7 Seasonal_naïve 2007 Q1 N(355, 4672)   355 [309, 401]50 [276, 434]75
 8 Seasonal_naïve 2007 Q2 N(435, 4672)   435 [389, 481]50 [356, 514]75
```

Prediction intervals

```r
brick_fc |>
  hilo(level = c(50, 75)) |>
  mutate(lower = `50%`$lower, upper = `50%`$upper)
```

```
# A tsibble: 80 x 8 [1Q]
# Key:       .model [4]
   .model         Quarter         Bricks .mean          `50%`          `75%` lower upper
   <chr>            <qtr>         <dist> <dbl>         <hilo>         <hilo> <dbl> <dbl>
 1 Seasonal_naïve 2005 Q3 N(428, 2336)   428 [395, 461]50 [372, 484]75  395.  461.
 2 Seasonal_naïve 2005 Q4 N(397, 2336)   397 [364, 430]50 [341, 453]75  364.  430.
 3 Seasonal_naïve 2006 Q1 N(355, 2336)   355 [322, 388]50 [299, 411]75  322.  388.
 4 Seasonal_naïve 2006 Q2 N(435, 2336)   435 [402, 468]50 [379, 491]75  402.  468.
 5 Seasonal_naïve 2006 Q3 N(428, 4672)   428 [382, 474]50 [349, 507]75  382.  474.
 6 Seasonal_naïve 2006 Q4 N(397, 4672)   397 [351, 443]50 [318, 476]75  351.  443.
 7 Seasonal_naïve 2007 Q1 N(355, 4672)   355 [309, 401]50 [276, 434]75  309.  401.
 8 Seasonal_naïve 2007 Q2 N(435, 4672)   435 [389, 481]50 [356, 514]75  389.  481.
```

Fitted values

- $\hat{y}_{t|t-1}$ is the forecast of y_t based on observations $y_1, \dots, y_t$.
- 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

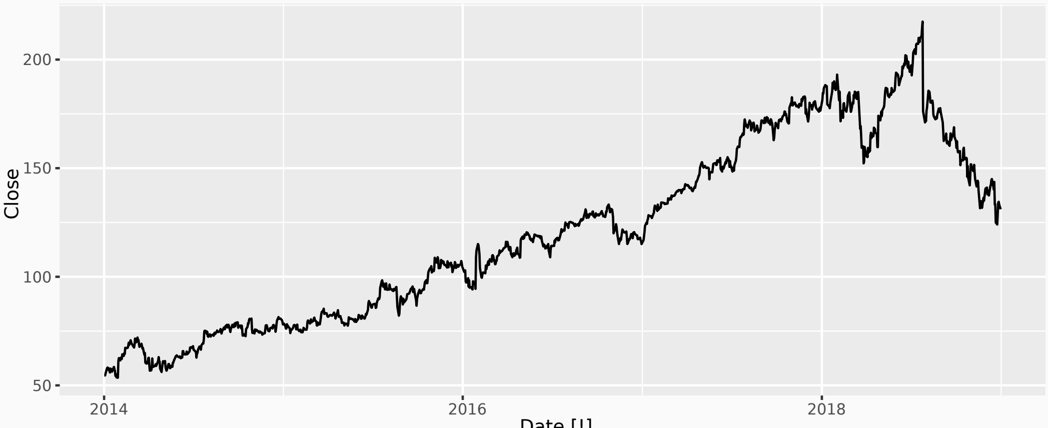
Residuals in forecasting: difference between observed value and its fitted value: $e_t = y_t - \hat{y}_{t|t-1}$.

Forecasting residuals

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

Facebook closing stock price

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



Facebook closing stock price

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

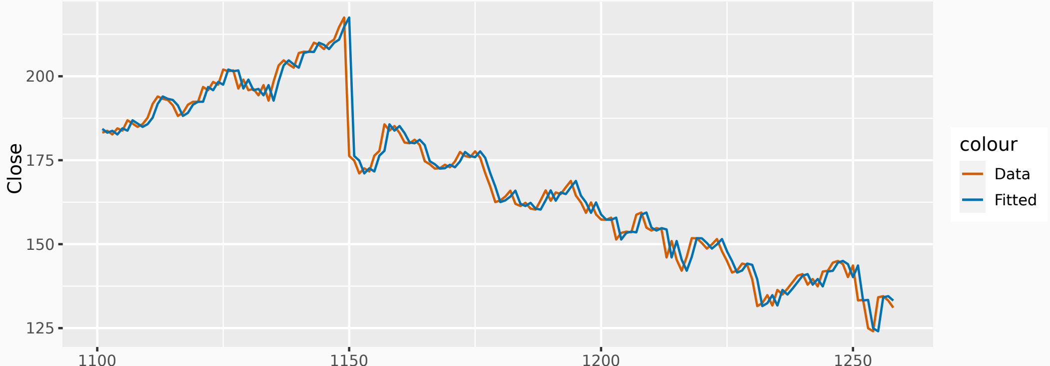
```
# A tsibble: 1,258 x 7 [1]
# Key:       Symbol, .model [1]
   Symbol .model       trading_day Close .fitted .resid .innov
   <chr>  <chr>              <int> <dbl>   <dbl>  <dbl>  <dbl>
 1 FB     NAIVE(Close)           1  54.7    NA   NA     NA
 2 FB     NAIVE(Close)           2  54.6    54.7 -0.150 -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
```

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

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