workshop.nectric.com.au/tidyfc2024

Time Series Analysis & Forecasting Using R

12. Accuracy evaluation

Outline

Outline

Tidy time series data

Use `as_tsibble()` to convert a dataset into a tsibble.

Identify which column(s) are:

- The `index` variable
- Identifying `key` variable(s)
- Measured variable(s)

Visualising time series

- Time plot: `data |> autoplot(y)`
- Season plot: `data |> gg_season(y)`
- Seasonal subseries plot: `data |> gg_subseries(y)`
- Lag plot: `data |> gg_lag(y)`
- ACF plot: `data |> ACF(y) |> autoplot()`

Transformations and decompositions

Simplify patterns with transformations:

- Population and inflation adjustments
- Mathematical transformations (`log()`, `sqrt()`, `box_cox()`)

Separate trend and seasonal patterns with decomposition:

- `STL()` decomposition (additive, choose windows)
- Extract decomposition with `components()`
- Produce seasonally adjusted data for decision making

Forecasting basics

Estimate a model on data with `model()`

Benchmark forecasting methods:

- Simple average: `MEAN(y)`
- Naive method: `NAIVE(y)`
- Seasonal naive method: `SNAIVE(y)`
- RW w/ drift: `RW(y ~ drift())`

Outline

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: $e_t = y_t - \hat{y}_{t|t-1}$.

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

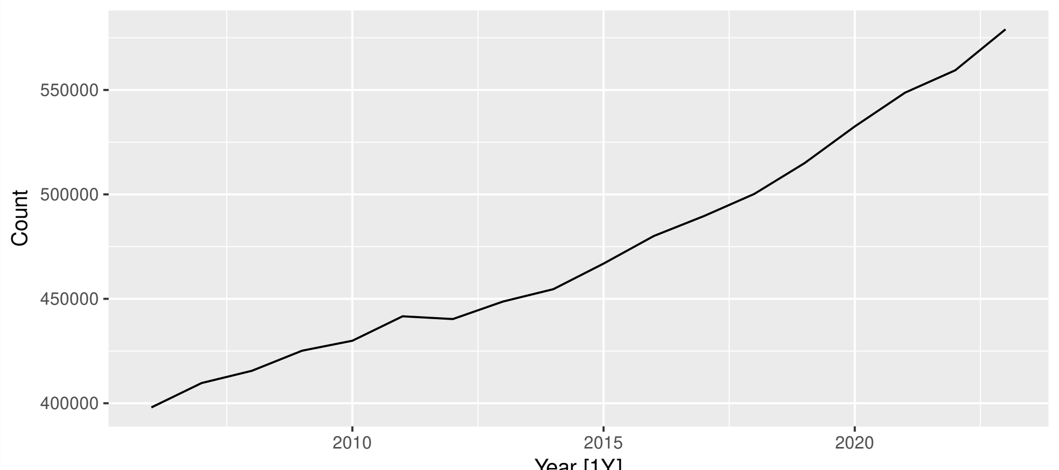
1. $\{e_t\}$ uncorrelated. If they aren't, then information left in residuals that should be used in computing forecasts.
2. $\{e_t\}$ have mean zero. If they don't, then forecasts are biased.

Useful properties (for prediction intervals)

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

Total in-school staff

```
total_staff <- staff |>
  summarise(Count = sum(`In-School Staff Count`))
total_staff |> autoplot(Count)
```



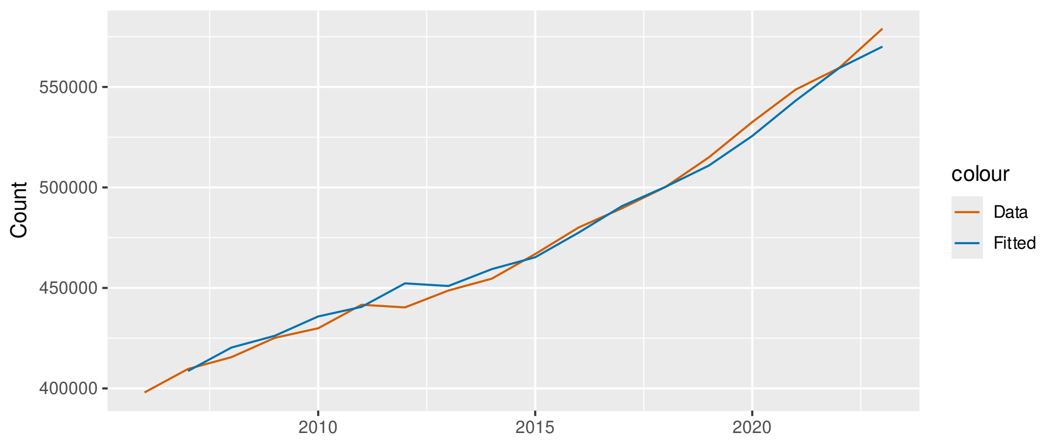
Total in-school staff

```
fit <- total_staff |> model(RW(Count ~ drift()))
augment(fit)
```

```
# A tsibble: 18 x 6 [1Y]
# Key:       .model [1]
   .model                 Year  Count .fitted   .resid   .innov
   <chr>                 <dbl>  <dbl>   <dbl>    <dbl>    <dbl>
 1 RW(Count ~ drift())  2006 398003      NA       NA       NA
 2 RW(Count ~ drift())  2007 409678 408652.    1026.    1026.
 3 RW(Count ~ drift())  2008 415541 420327.   -4786.   -4786.
 4 RW(Count ~ drift())  2009 425166 426190.   -1024.   -1024.
 5 RW(Count ~ drift())  2010 429933 435815.   -5882.   -5882.
 6 RW(Count ~ drift())  2011 441631 440582.    1049.    1049.
 7 RW(Count ~ drift())  2012 440313 452280.  -11967.  -11967.
 8 RW(Count ~ drift())  2013 448711 450962.   -2251.   -2251.
 9 RW(Count ~ drift())  2014 454615 459360.   -4745.   -4745.
10 RW(Count ~ drift())  2015 466867 465264.    1603.    1603.
11 RW(Count ~ drift())  2016 480077 477516.    2561.    2561.
12 RW(Count ~ drift())  2017 489645 490726.   -1081.   -1081.
```

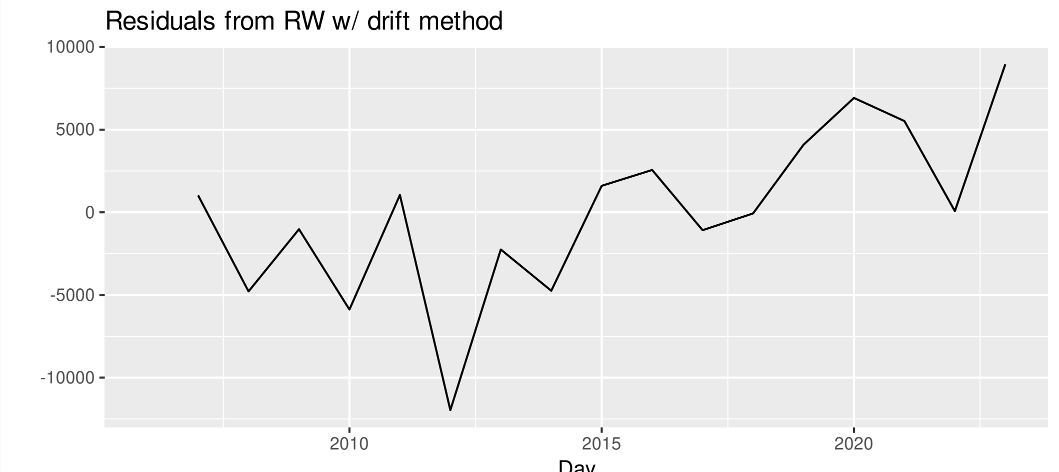
Total in-school staff

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



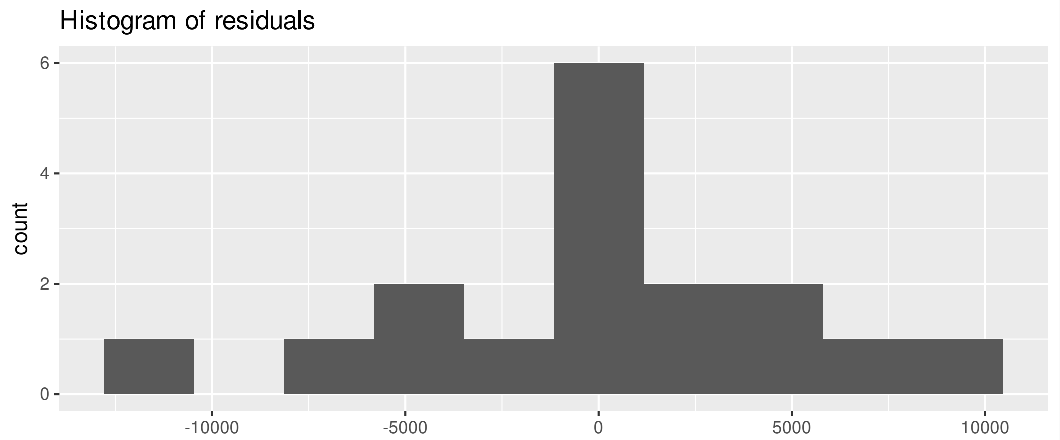
Total in-school staff

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



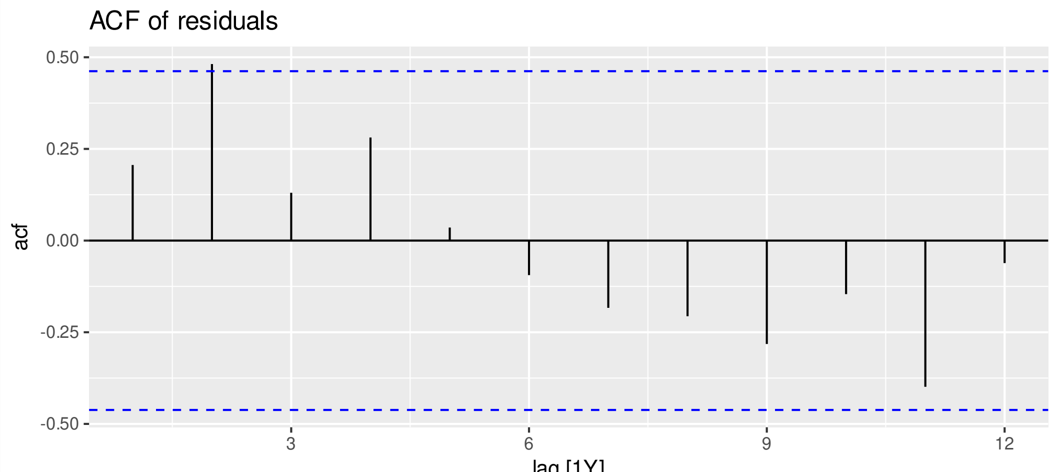
Total in-school staff

```
augment(fit) |>
  ggplot(aes(x = .resid)) +
  geom_histogram(bins = 10) +
  labs(title = "Histogram of residuals")
```



Total in-school staff

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

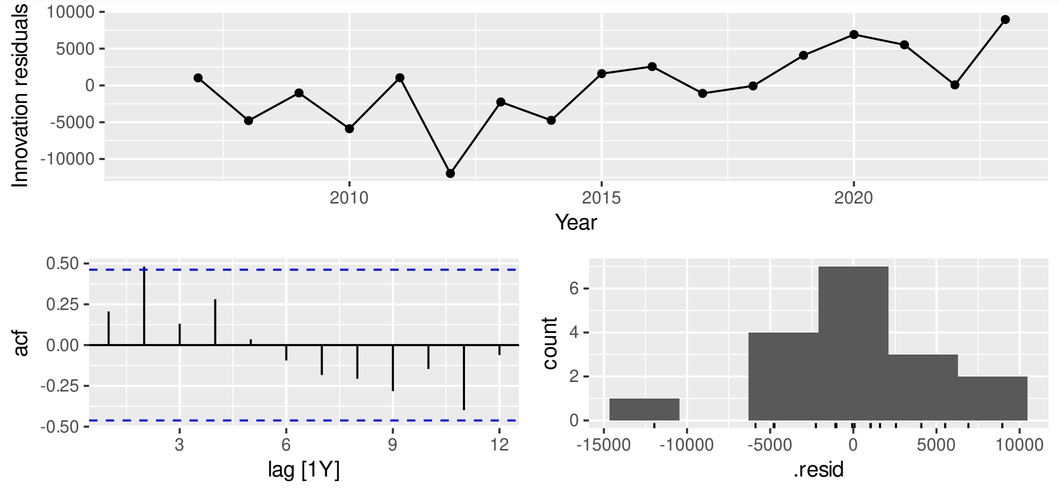


ACF of residuals

- 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

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



Ljung-Box test

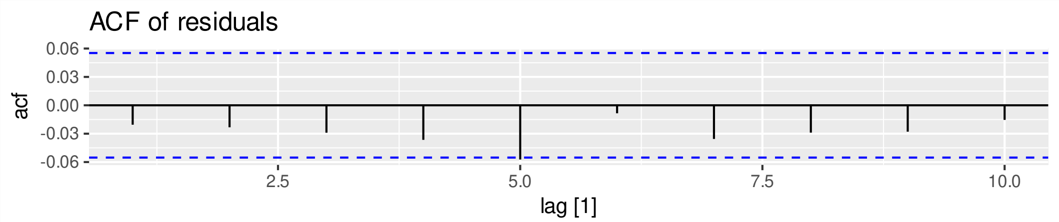
Test whether *whole set* of r_k values is significantly different from zero set.

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

- 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 and T large, $Q \sim \chi^2$ with ℓ degrees of freedom.

Ljung-Box test

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




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

```
# A tibble: 1 x 3
  .model                lb_stat lb_pvalue
  <chr>                   <dbl>     <dbl>
1 RW(Count ~ drift())      15.3     0.121
```

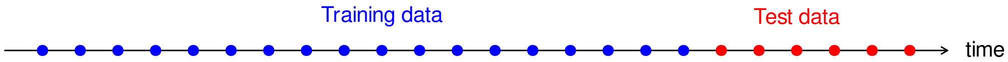
Outline

Lab Session 10

- Compute RW w/ drift forecasts for total student enrolments in Australia (`students`).
- Test if the residuals are white noise. What do you conclude?

Outline

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

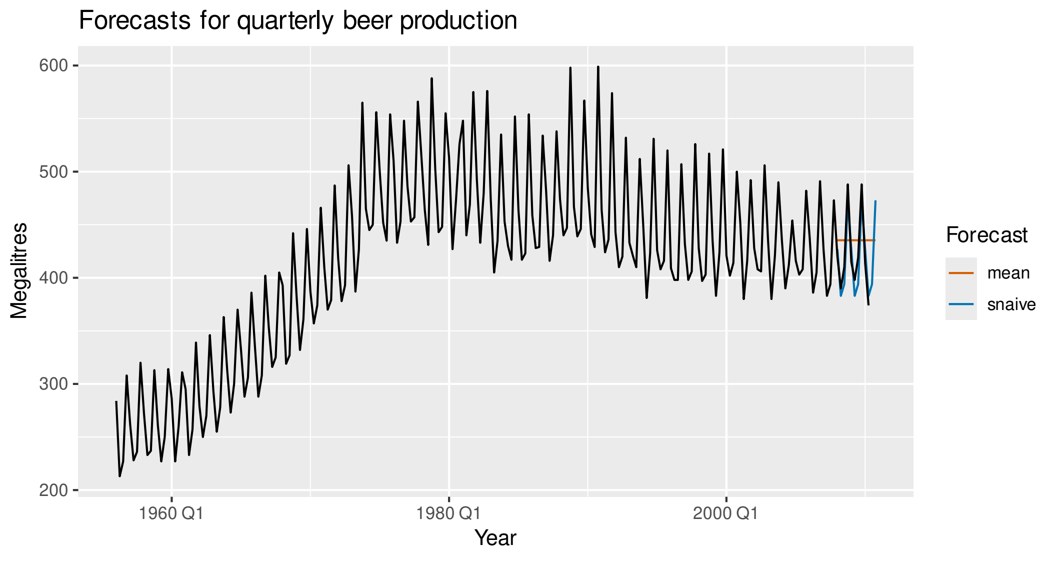
Forecast "error": the difference between an observed value and its forecast.

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

Measures of forecast accuracy

```r
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

$$
\begin{aligned}
y_{T+h} &= (T+h)\text{th observation, } h = 1, \dots, H \\
\hat{y}_{T+h|T} &= \text{its forecast based on data up to time } T. \\
e_{T+h} &= y_{T+h} - \hat{y}_{T+h|T}
\end{aligned}
$$

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

- Mean error is an indicator of bias.
- On training accuracy, it is expected to be 0.

Measures of forecast accuracy

$$
\begin{aligned}
y_{T+h} &= (T+h)\text{th observation, } h = 1, \dots, H \\
\hat{y}_{T+h|T} &= \text{its forecast based on data up to time } T. \\
e_{T+h} &= y_{T+h} - \hat{y}_{T+h|T}
\end{aligned}
$$

$$
\begin{aligned}
\text{MAE} &= \text{mean}(|e_{T+h}|) \\
\text{MSE} &= \text{mean}(e_{T+h}^2) \qquad \text{RMSE} = \sqrt{\text{mean}(e_{T+h}^2)} \\
\text{MAPE} &= 100\text{mean}(|e_{T+h}|/|y_{T+h}|)
\end{aligned}
$$

Measures of forecast accuracy

$$
\begin{aligned}
y_{T+h} &= (T+h)\text{th observation, } h = 1,\dots,H \\
\hat{y}_{T+h|T} &= \text{its forecast based on data up to time } T. \\
e_{T+h} &= y_{T+h} - \hat{y}_{T+h|T}
\end{aligned}
$$

$$
\begin{aligned}
\text{MAE} &= \text{mean}(|e_{T+h}|) \\
\text{MSE} &= \text{mean}(e_{T+h}^2) \qquad \text{RMSE} = \sqrt{\text{mean}(e_{T+h}^2)} \\
\text{MAPE} &= 100\text{mean}(|e_{T+h}|/|y_{T+h}|)
\end{aligned}
$$

- MAE, MSE, RMSE are all scale dependent.
- MAPE is scale independent but is only sensible if $y_t \gg 0$ for all t, and y has a natural zero.

Measures of forecast accuracy

Mean Absolute Scaled Error

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

Measures of forecast accuracy

Mean Absolute Scaled Error

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

- For non-seasonal series, scale uses naïve forecasts:

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

- For seasonal series, scale uses seasonal naïve forecasts:

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

where m is the seasonal frequency

Measures of forecast accuracy

Mean Absolute Scaled Error

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

- For non-seasonal series, scale uses naïve forecasts:

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

- For seasonal series, scale uses seasonal naïve forecasts:

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

where m is the seasonal frequency

Proposed by Hyndman and Koehler (IJF, 2006).

Measures of forecast accuracy

Root Mean Squared Scaled Error

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

- For non-seasonal series, scale uses naïve forecasts:
$$Q = \frac{1}{T-1}\sum_{t=2}^{T}(y_t - y_{t-1})^2$$
- For seasonal series, scale uses seasonal naïve forecasts:
$$Q = \frac{1}{T-m}\sum_{t=m+1}^{T}(y_t - y_{t-m})^2$$
where m is the seasonal frequencyq

Proposed by Hyndman and Koehler (IJF, 2006).

Measures of forecast accuracy

```r
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

Lab Session 11

- Create a training set for employed Australian students (`student_labour`) 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?

> **Finished early?**
>
> Repeat the exercise using the Australian takeaway food turnover data (`aus_retail`) with a test set of four years.