AARHUS UNIVERSITY

Data Science 11

Chris Mathys

Master's Degree Programme in Cognitive Science

Spring 2023

Outline

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Multiple regression and forecasting

$$y_t = \beta_0 + \beta_1 x_{1,t} + \beta_2 x_{2,t} + \cdots + \beta_k x_{k,t} + \varepsilon_t.$$

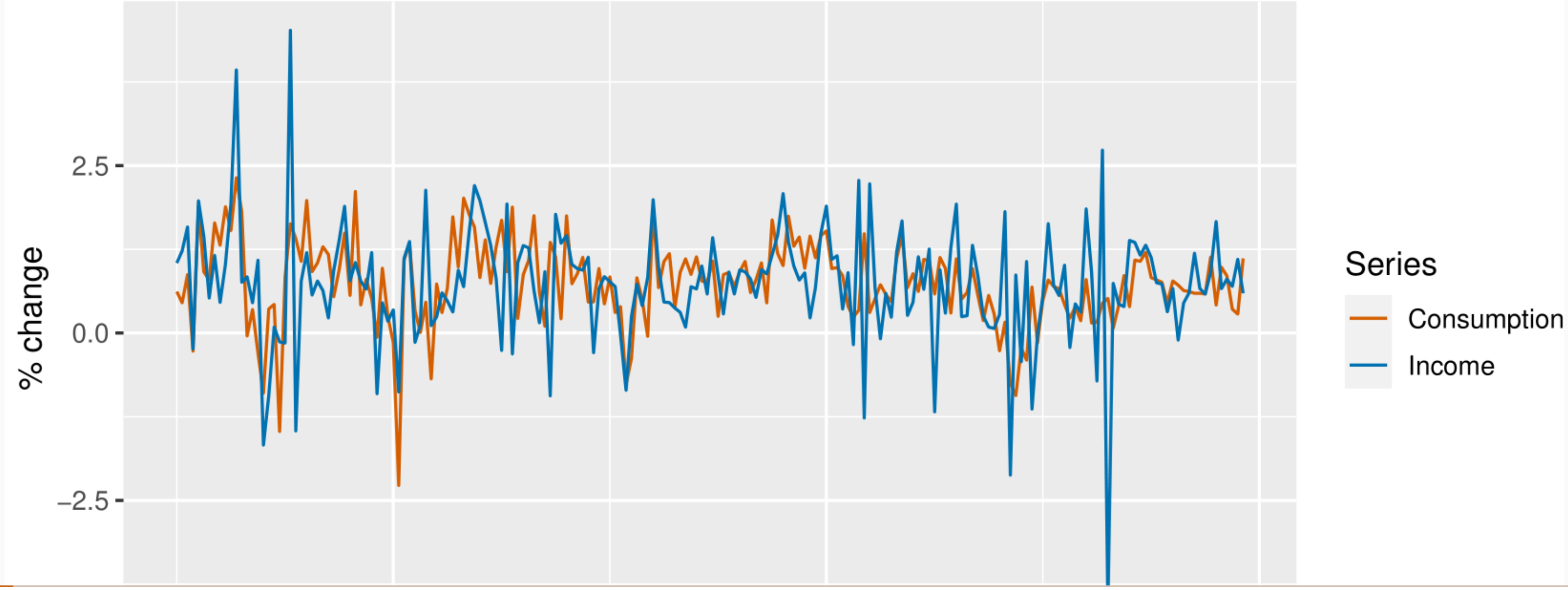
- y_t is the variable we want to predict: the "response" variable
- Each $x_{j,t}$ is numerical and is called a "predictor". They are usually assumed to be known for all past and future times.
- The coefficients $\beta_1, \ldots, \beta_k$ measure the effect of each predictor after taking account of the effect of all other predictors in the model.

That is, the coefficients measure the **marginal effects**.

- ε_t is a white noise error term

Example: US consumption expenditure

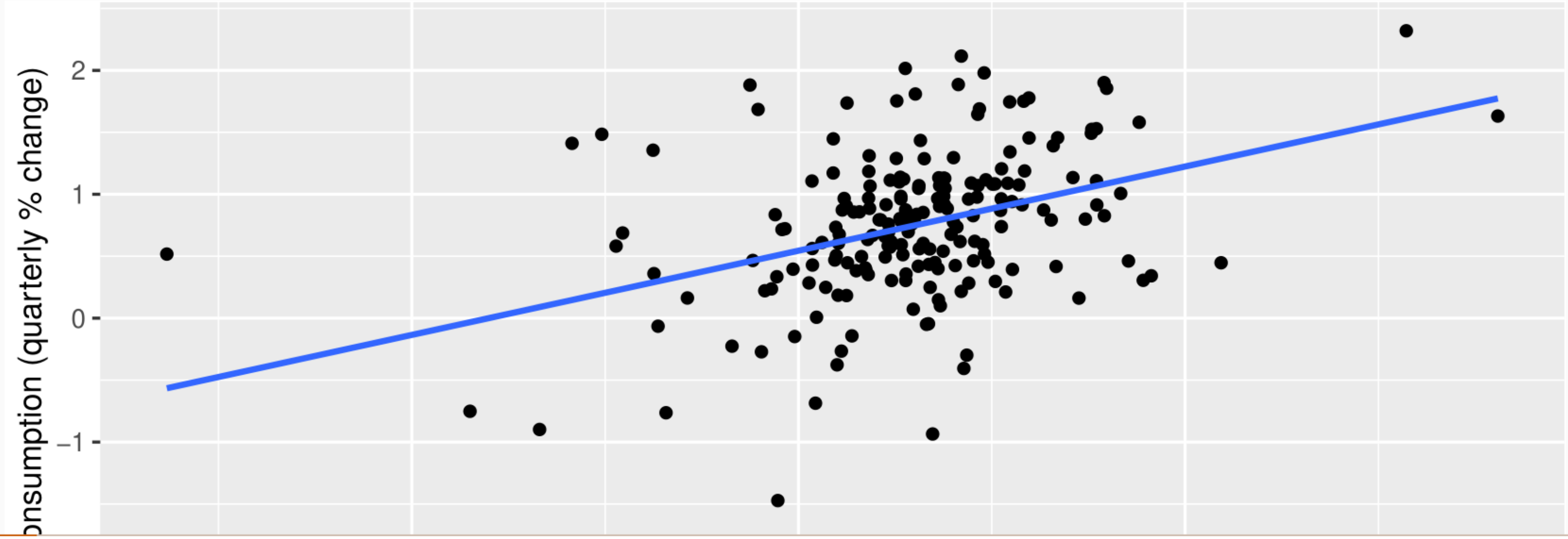
```
us_change %>%
  pivot_longer(c(Consumption, Income), names_to="Series") %>%
  autoplot(value) +
  labs(y="% change")
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: US consumption expenditure

```
us_change %>%
  ggplot(aes(x = Income, y = Consumption)) +
    labs(y = "Consumption (quarterly % change)",
         x = "Income (quarterly % change)") +
    geom_point() + geom_smooth(method = "lm", se = FALSE)
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: US consumption expenditure

```
fit_cons <- us_change %>%
  model(lm = TSLM(Consumption ~ Income))
report(fit_cons)
```

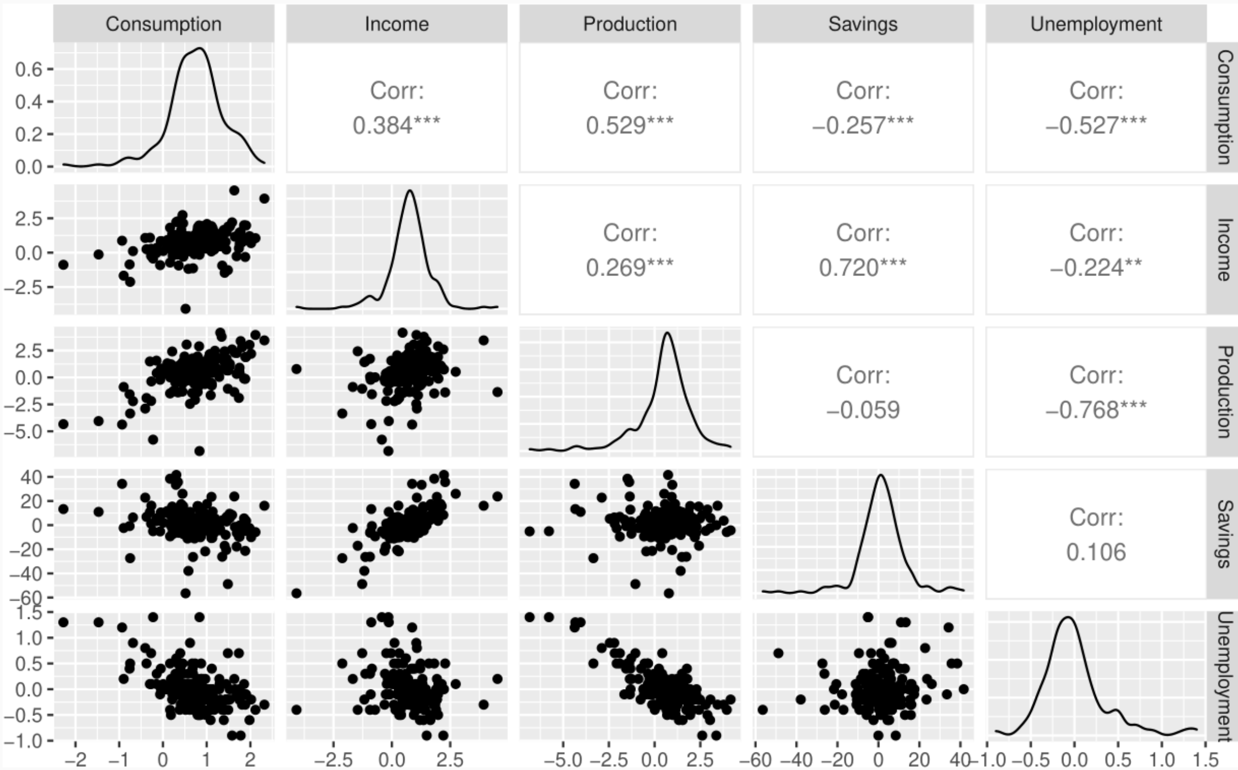
```
## Series: Consumption
## Model: TSLM
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.582 -0.278  0.019  0.323  1.422
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.5445     0.0540   10.08  < 2e-16 ***
## Income        0.2718     0.0467    5.82  2.4e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.591 on 196 degrees of freedom
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: US consumption expenditure

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: US consumption expenditure



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

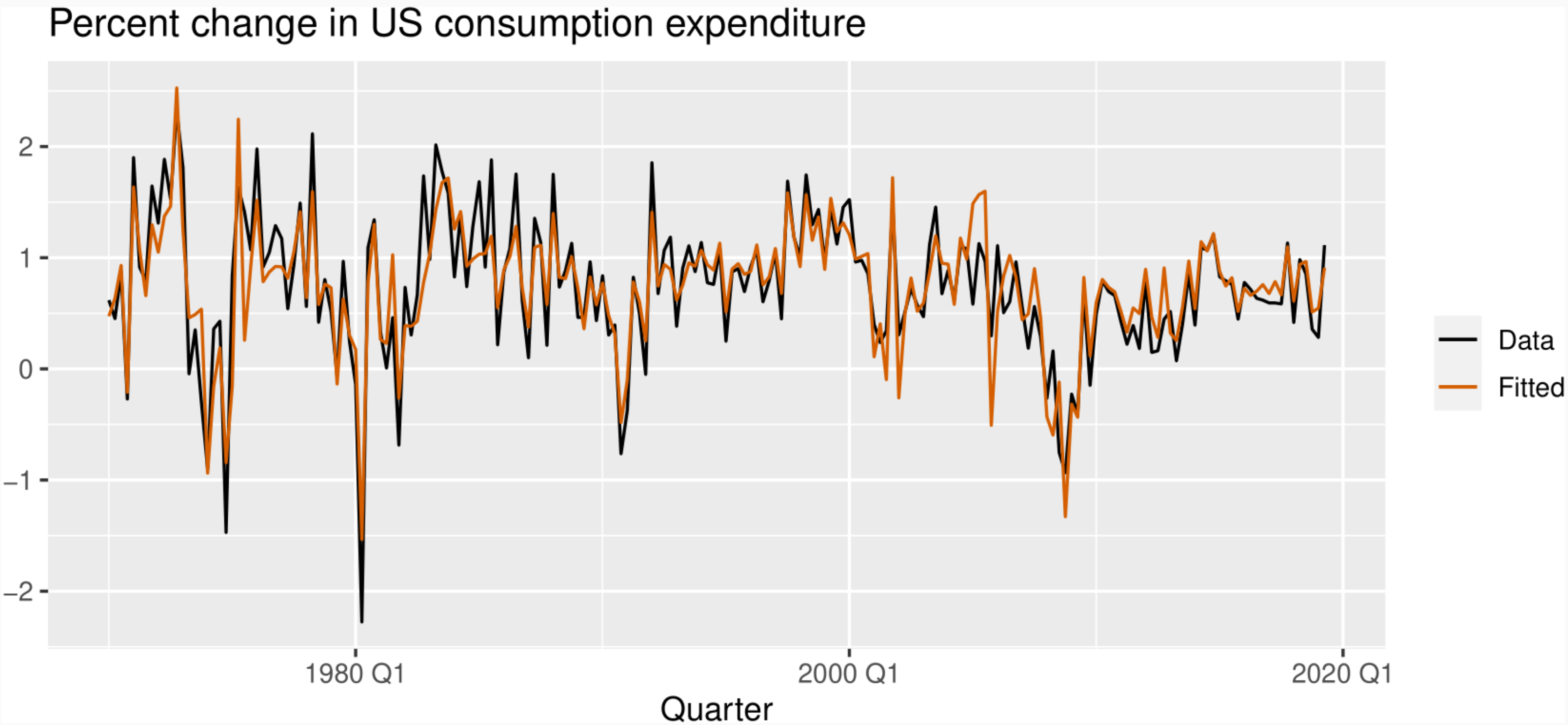
Example: US consumption expenditure

```
fit_consMR <- us_change %>%
  model(lm = TSLM(Consumption ~ Income + Production + Unemployment + Savings))
report(fit_consMR)
```

```
## Series: Consumption
## Model: TSLM
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.906 -0.158 -0.036  0.136  1.155
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.25311    0.03447    7.34  5.7e-12 ***
## Income        0.74058    0.04012   18.46  < 2e-16 ***
## Production    0.04717    0.02314    2.04    0.043 *
## Unemployment -0.17469    0.09551   -1.83    0.069 .
## Savings      -0.05289    0.00292  -18.09  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.31 on 193 degrees of freedom
## Multiple R-squared: 0.768    Adjusted R-squared: 0.763
```

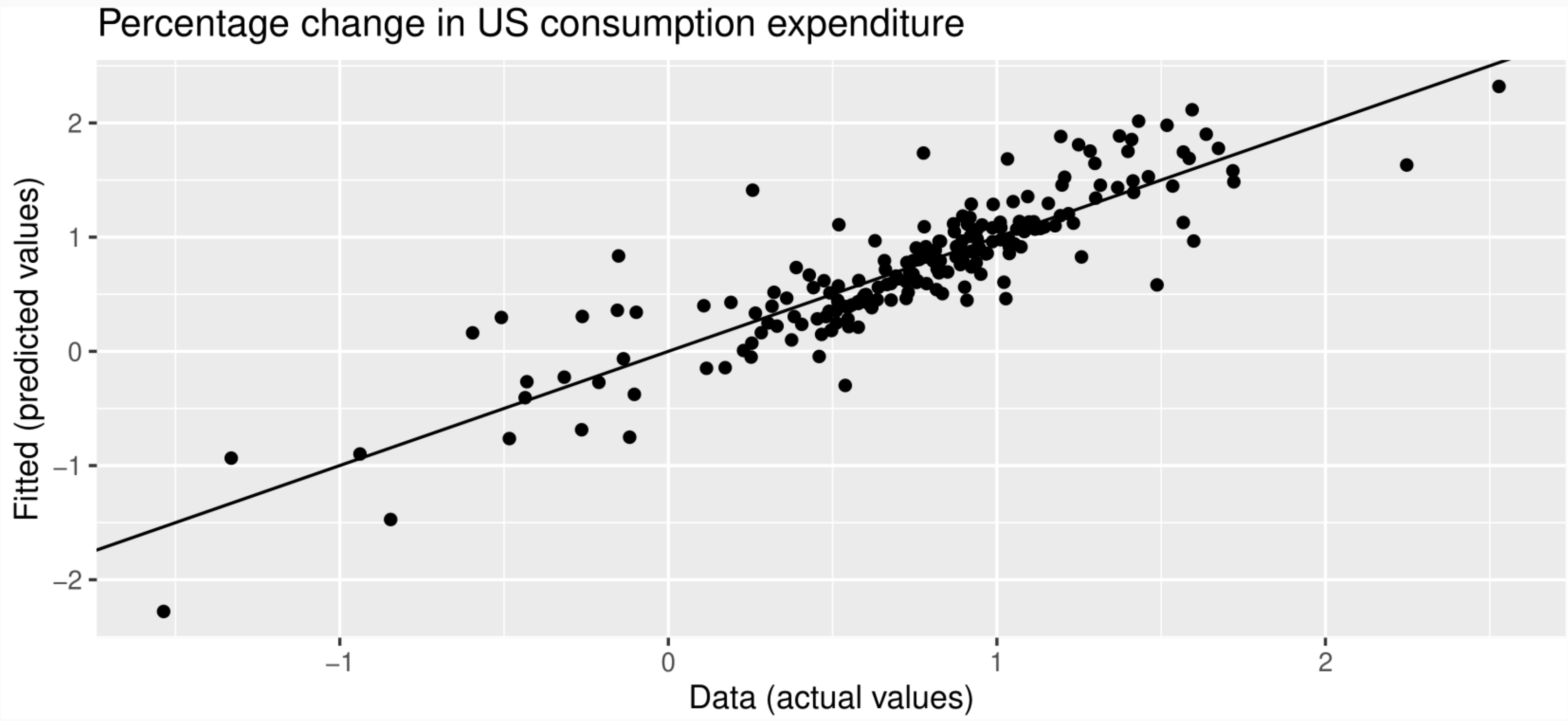

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: US consumption expenditure



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

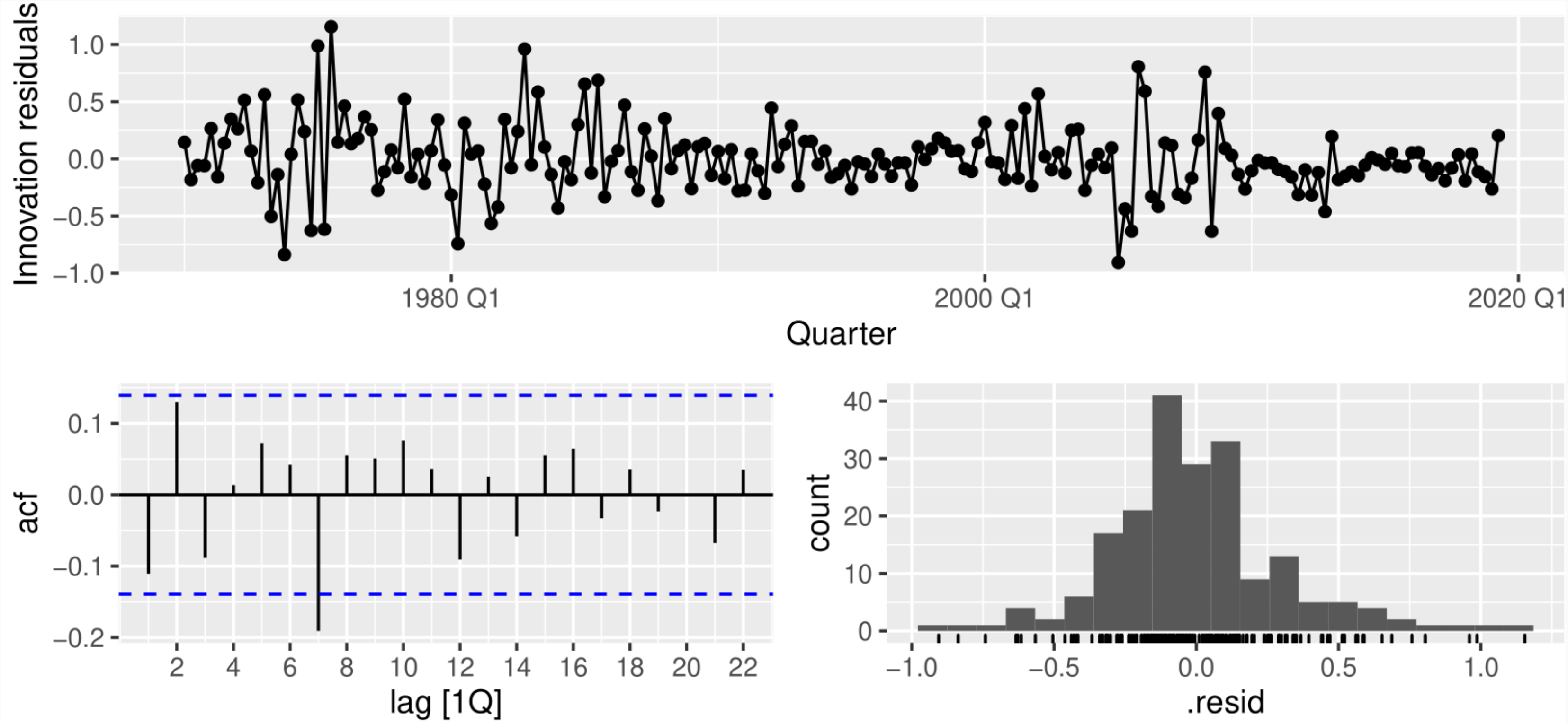
Example: US consumption expenditure



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: US consumption expenditure

```
fit_consMR %>% gg_tsresiduals()
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Outline

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Trend

Linear trend

$$x_t = t$$

- $t = 1, 2, \ldots, T$
- Strong assumption that trend will continue.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Dummy variables

If a categorical variable takes only two values (e.g., 'Yes' or 'No'), then an equivalent numerical variable can be constructed taking value 1 if yes and 0 if no. This is called a **dummy variable**.

	A	B
1	Yes	1
2	Yes	1
3	No	0
4	Yes	1
5	No	0
6	No	0
7	Yes	1
8	Yes	1
9	No	0
10	No	0
11	No	0
12	No	0
13	Yes	1
14	No	0

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Dummy variables

If there are more than two categories, then the variable can be coded using several dummy variables (one fewer than the total number of categories).

	A	B	C	D	E
1	Monday	1	0	0	0
2	Tuesday	0	1	0	0
3	Wednesday	0	0	1	0
4	Thursday	0	0	0	1
5	Friday	0	0	0	0
6	Monday	1	0	0	0
7	Tuesday	0	1	0	0
8	Wednesday	0	0	1	0
9	Thursday	0	0	0	1
10	Friday	0	0	0	0
11	Monday	1	0	0	0
12	Tuesday	0	1	0	0
13	Wednesday	0	0	1	0
14	Thursday	0	0	0	1
15	Friday	0	0	0	0

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Beware of the dummy variable trap!

- Using one dummy for each category gives too many dummy variables!
- The regression will then be singular and inestimable.
- Either omit the constant, or omit the dummy for one category.
- The coefficients of the dummies are relative to the omitted category.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Uses of dummy variables

Seasonal dummies

- For quarterly data: use 3 dummies
- For monthly data: use 11 dummies
- For daily data: use 6 dummies
- What to do with weekly data?

Outliers

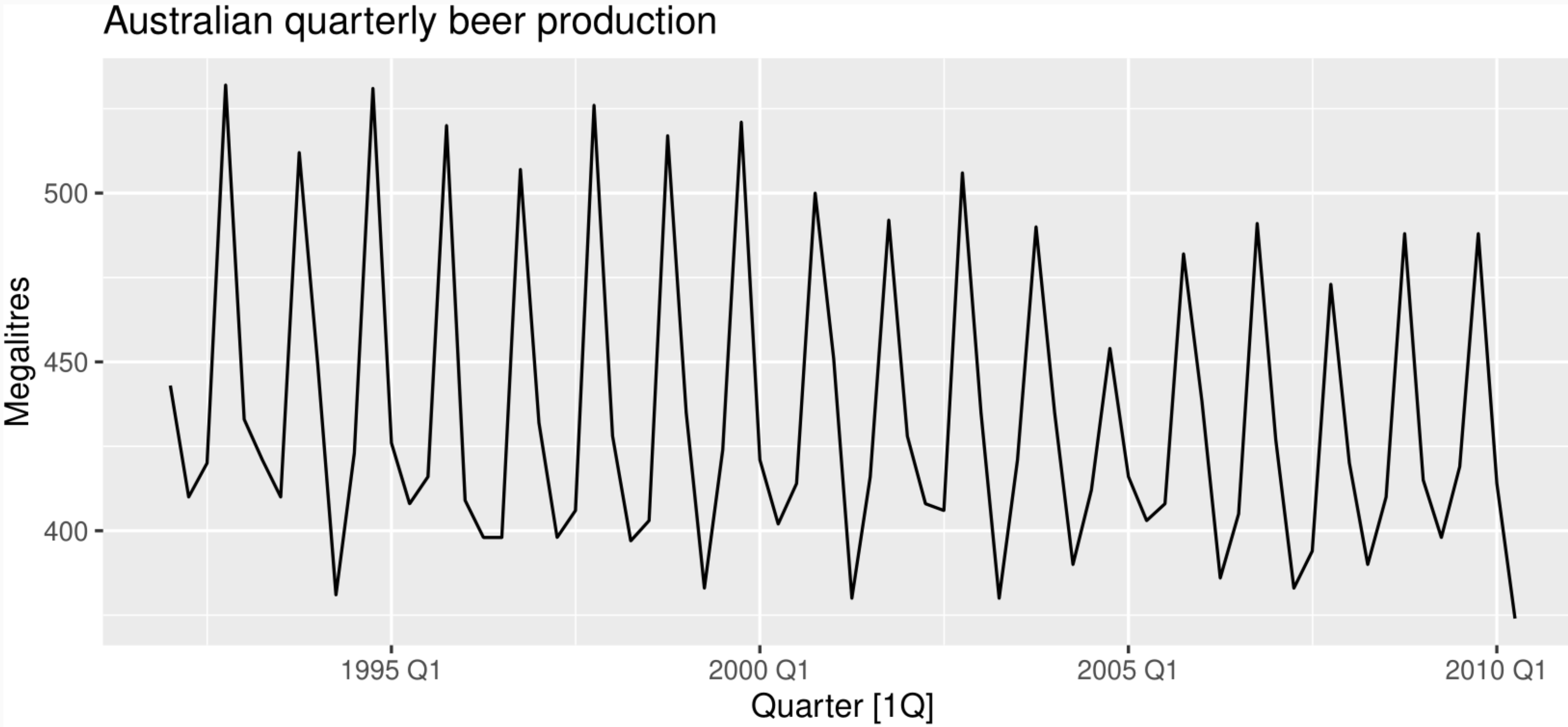
- If there is an outlier, you can use a dummy variable to remove its effect.

Public holidays

- For daily data: if it is a public holiday, dummy=1, otherwise dummy=0.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Beer production revisited



Regression model

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Beer production revisited

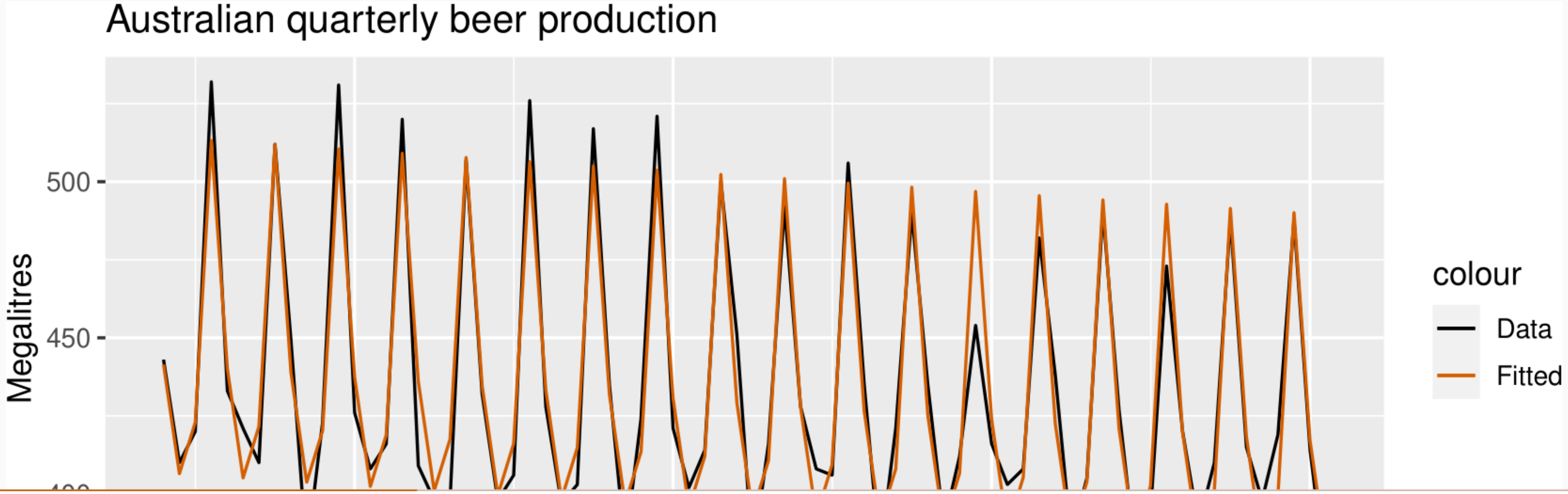
```r
fit_beer <- recent_production %>% model(TSLM(Beer ~ trend() + season()))
report(fit_beer)
```

```
## Series: Beer
## Model: TSLM
##
## Residuals:
##     Min      1Q  Median      3Q     Max
##   -42.9    -7.6    -0.5     8.0    21.8
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    441.8004     3.7335  118.33  < 2e-16 ***
## trend()         -0.3403     0.0666   -5.11  2.7e-06 ***
## season()year2 -34.6597     3.9683   -8.73  9.1e-13 ***
## season()year3 -17.8216     4.0225   -4.43  3.4e-05 ***
## season()year4  72.7964     4.0230   18.09  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.2 on 69 degrees of freedom
## Multiple R-squared: 0.924,    Adjusted R-squared: 0.92
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

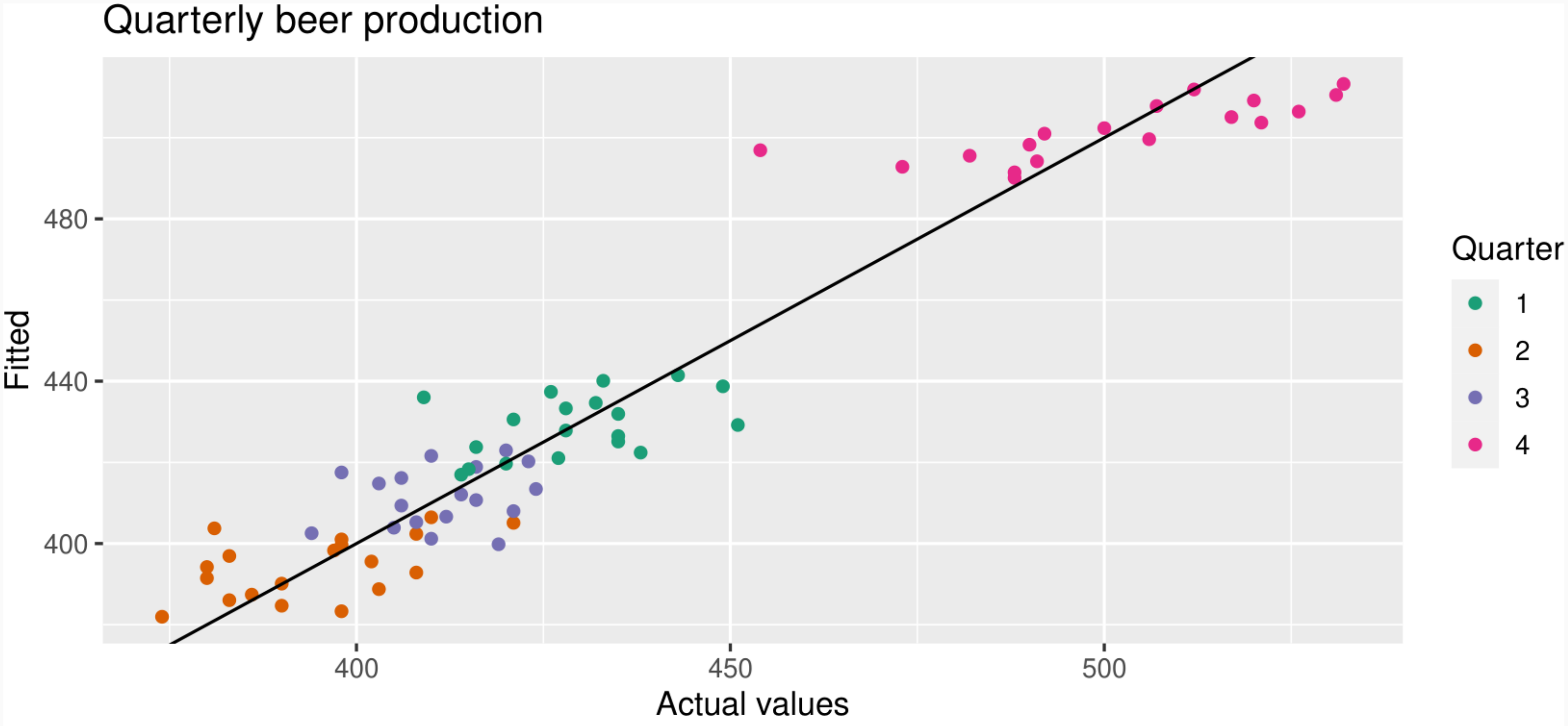
Beer production revisited

```
augment(fit_beer) %>%
  ggplot(aes(x = Quarter)) +
  geom_line(aes(y = Beer, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  labs(y="Megalitres",title ="Australian quarterly beer production") +
  scale_colour_manual(values = c(Data = "black", Fitted = "#D55E00"))
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

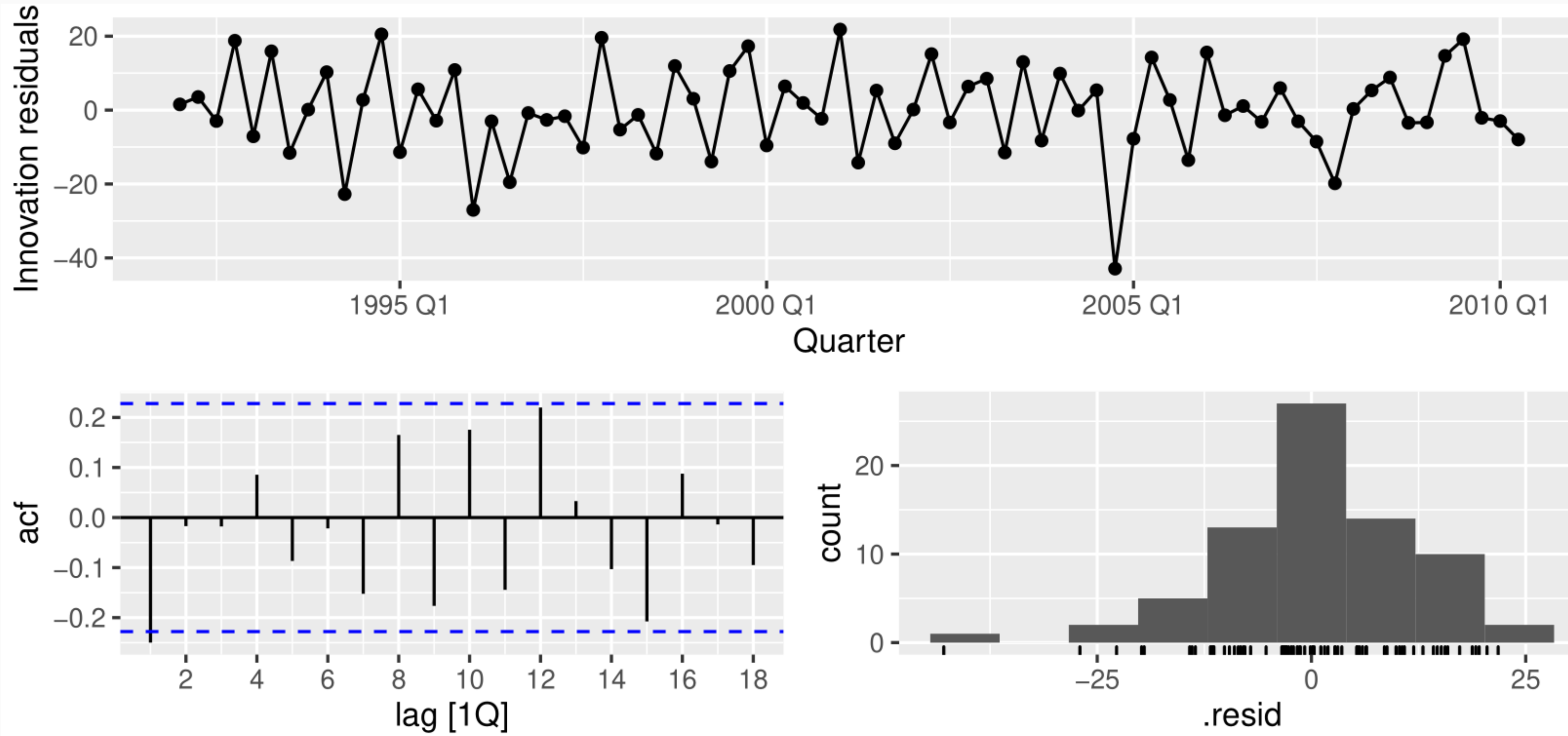
Beer production revisited



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Beer production revisited

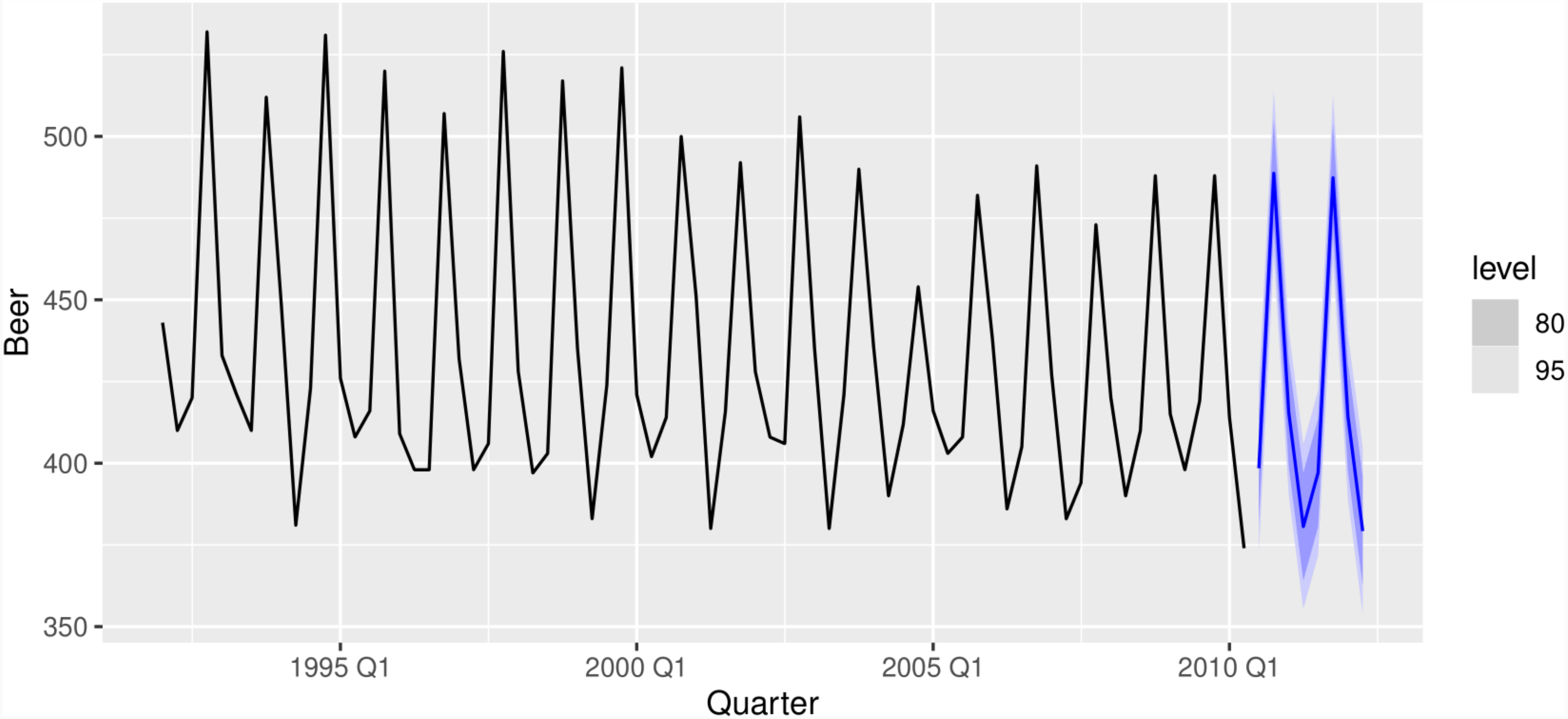
```
fit_beer %>% gg_tsresiduals()
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Beer production revisited

```
fit_beer %>% forecast %>% autoplot(recent_production)
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Fourier series

Periodic seasonality can be handled using pairs of Fourier terms:

$$s_k(t) = \sin\left(\frac{2\pi kt}{m}\right) \qquad c_k(t) = \cos\left(\frac{2\pi kt}{m}\right)$$

$$y_t = a + bt + \sum_{k=1}^{K} [\alpha_k s_k(t) + \beta_k c_k(t)] + \varepsilon_t$$

- Every periodic function can be approximated by sums of sin and cos terms for large enough K.
- Choose K by minimizing AICc.
- Called "harmonic regression"

```
TSLM(y ~ trend() + fourier(K))
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Harmonic regression: beer production

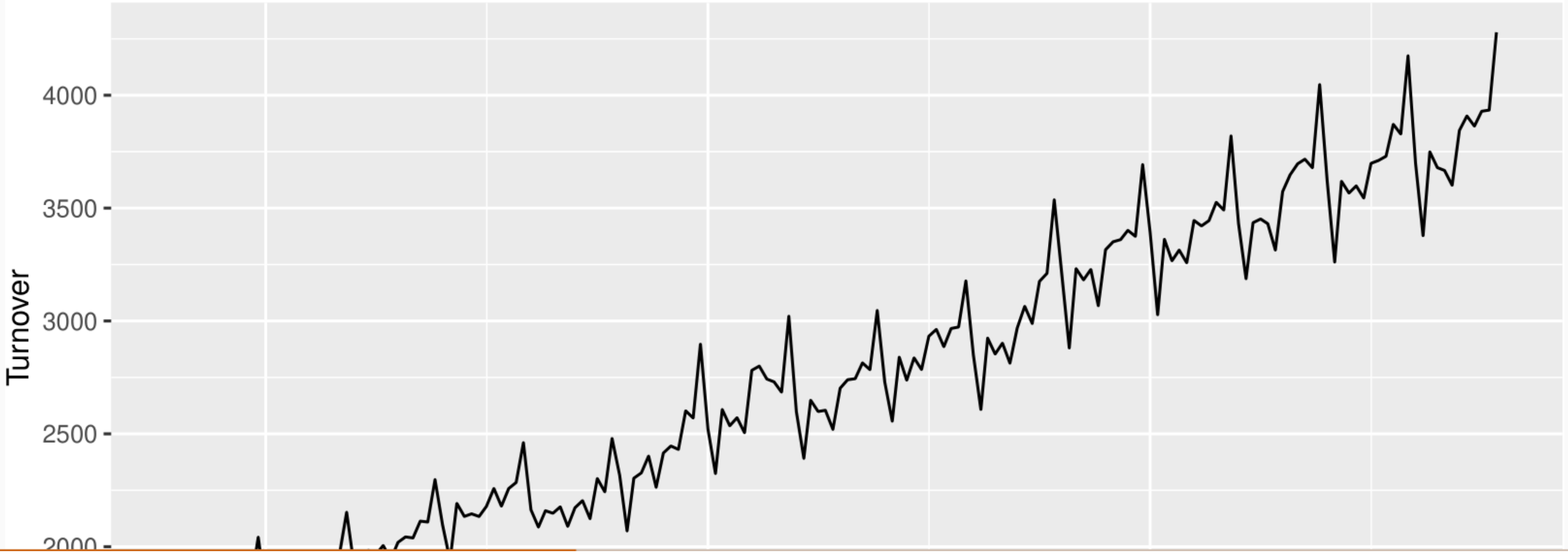
```r
fourier_beer <- recent_production %>% model(TSLM(Beer ~ trend() + fourier(K=2)))
report(fourier_beer)
```

```
## Series: Beer
## Model: TSLM
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -42.9   -7.6   -0.5    8.0   21.8
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          446.8792     2.8732  155.53  < 2e-16 ***
## trend()               -0.3403     0.0666   -5.11  2.7e-06 ***
## fourier(K = 2)C1_4    8.9108     2.0112    4.43  3.4e-05 ***
## fourier(K = 2)S1_4 -53.7281     2.0112  -26.71  < 2e-16 ***
## fourier(K = 2)C2_4 -13.9896     1.4226   -9.83  9.3e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.2 on 69 degrees of freedom
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Harmonic regression: eating-out expenditure

```
aus_cafe <- aus_retail %>% filter(
    Industry == "Cafes, restaurants and takeaway food services",
    year(Month) %in% 2004:2018
  ) %>% summarise(Turnover = sum(Turnover))
aus_cafe %>% autoplot(Turnover)
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

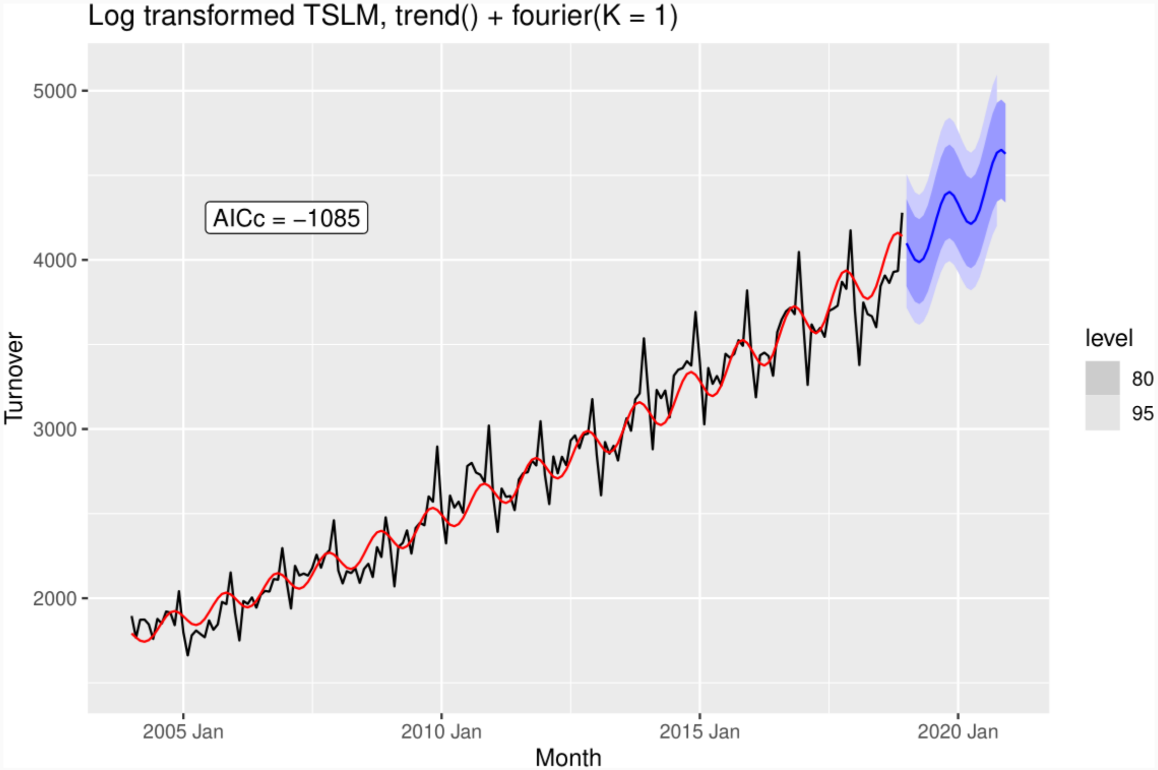
Harmonic regression: eating-out expenditure

```
fit <- aus_cafe %>%
  model(K1 = TSLM(log(Turnover) ~ trend() + fourier(K = 1)),
        K2 = TSLM(log(Turnover) ~ trend() + fourier(K = 2)),
        K3 = TSLM(log(Turnover) ~ trend() + fourier(K = 3)),
        K4 = TSLM(log(Turnover) ~ trend() + fourier(K = 4)),
        K5 = TSLM(log(Turnover) ~ trend() + fourier(K = 5)),
        K6 = TSLM(log(Turnover) ~ trend() + fourier(K = 6)))
glance(fit) %>% select(.model, r_squared, adj_r_squared, AICc)
```

```
## # A tibble: 6 x 4
##   .model r_squared adj_r_squared   AICc
##   <chr>      <dbl>         <dbl>  <dbl>
## 1 K1         0.962         0.962 -1085.
## 2 K2         0.966         0.965 -1099.
## 3 K3         0.976         0.975 -1160.
## 4 K4         0.980         0.979 -1183.
## 5 K5         0.985         0.984 -1234.
## 6 K6         0.985         0.984 -1232.
```

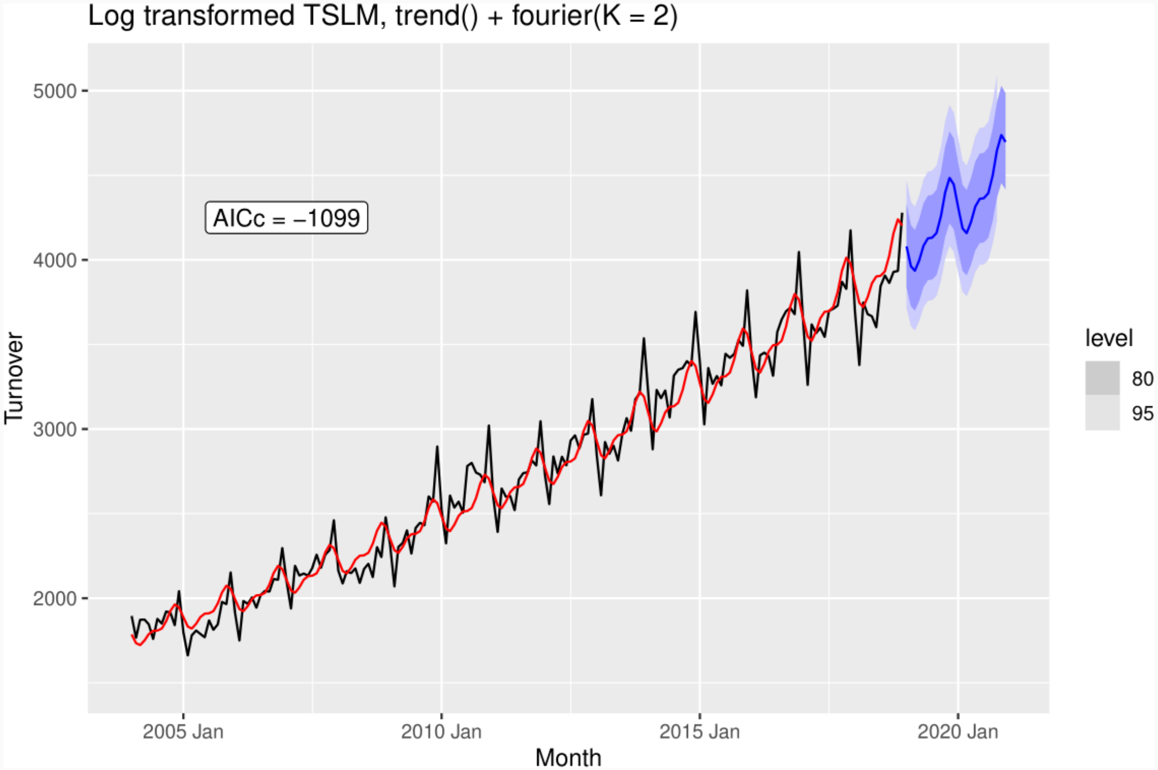

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Harmonic regression: eating-out expenditure



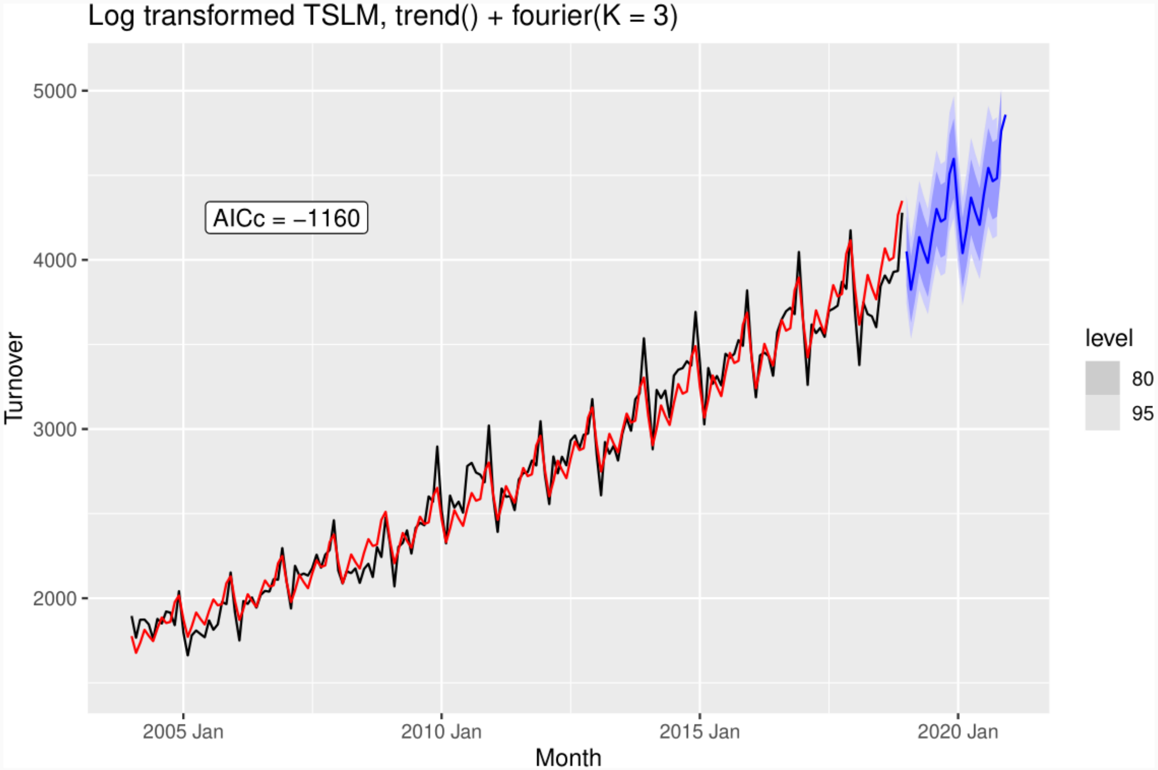
Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Harmonic regression: eating-out expenditure



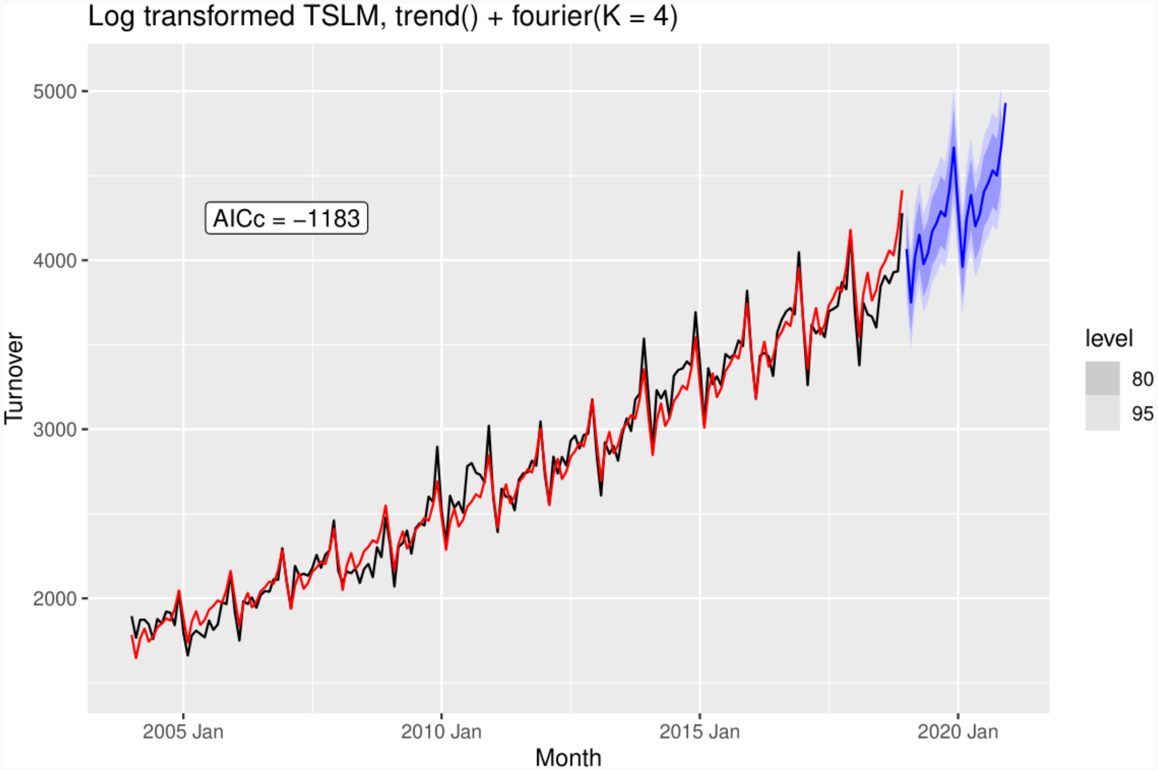
Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Harmonic regression: eating-out expenditure



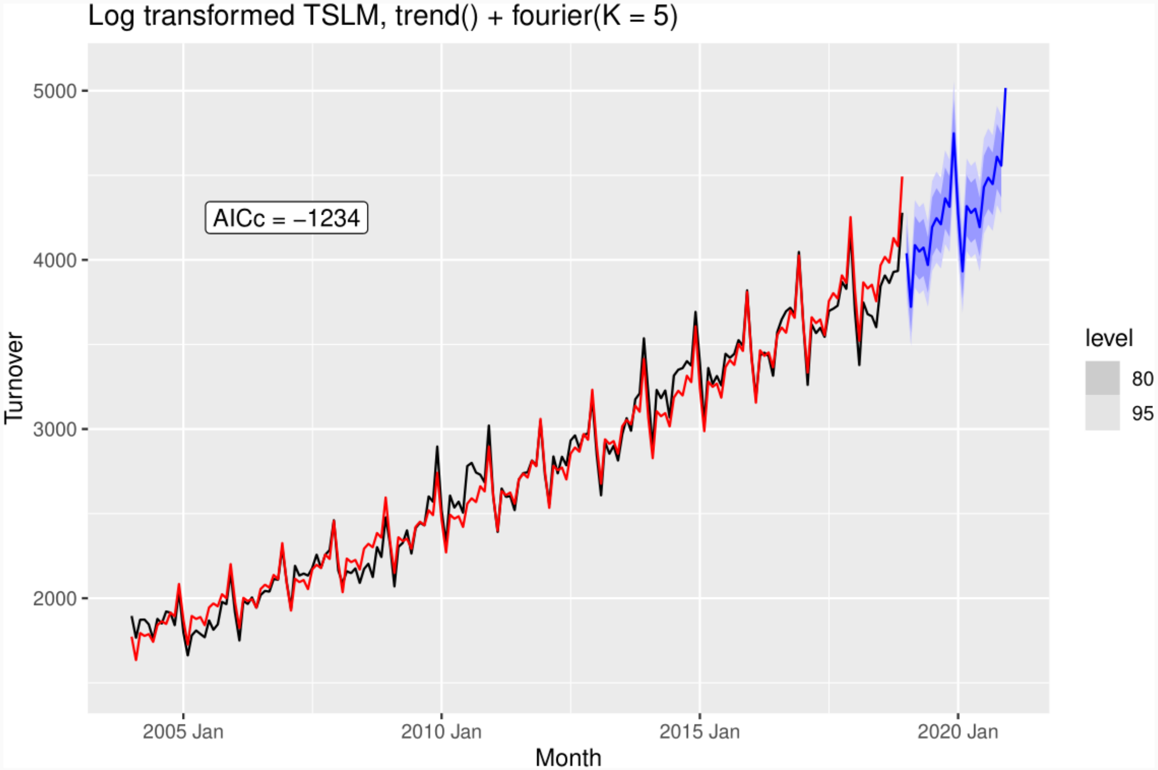
Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Harmonic regression: eating-out expenditure



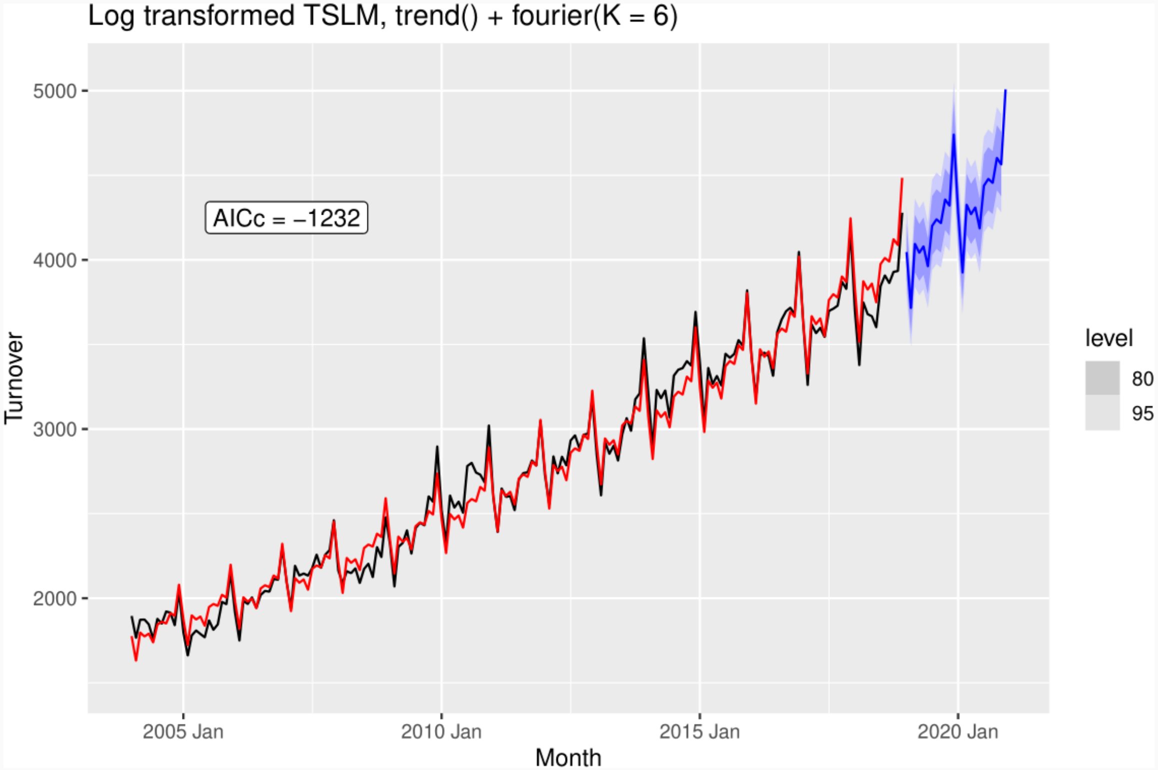
Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Harmonic regression: eating-out expenditure



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Harmonic regression: eating-out expenditure



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Intervention variables

Spikes

- Equivalent to a dummy variable for handling an outlier.

Steps

- Variable takes value 0 before the intervention and 1 afterwards.

Change of slope

- Variables take values 0 before the intervention and values $\{1, 2, 3, \dots\}$ afterwards.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Holidays

For monthly data

- Christmas: always in December so part of monthly seasonal effect
- Easter: use a dummy variable $v_t = 1$ if any part of Easter is in that month, $v_t = 0$ otherwise.
- Ramadan and Chinese new year similar.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Trading days

With monthly data, if the observations vary depending on how many different types of days in the month, then trading day predictors can be useful.

$$
\begin{aligned}
z_1 &= \text{\# Mondays in month;} \\
z_2 &= \text{\# Tuesdays in month;} \\
&\vdots \\
z_7 &= \text{\# Sundays in month.}
\end{aligned}
$$

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Distributed lags

Lagged values of a predictor.

Example: x is advertising which has a delayed effect

$$\begin{aligned} x_1 &= \text{advertising for previous month;} \\ x_2 &= \text{advertising for two months previously;} \\ &\vdots \\ x_m &= \text{advertising for } m \text{ months previously.} \end{aligned}$$

Nonlinear trend

Piecewise linear trend with bend at τ

$$x_{1,t} = t$$

$$x_{2,t} = \begin{cases} 0 & t < \tau \\ (t - \tau) & t \geq \tau \end{cases}$$

Quadratic or higher order trend

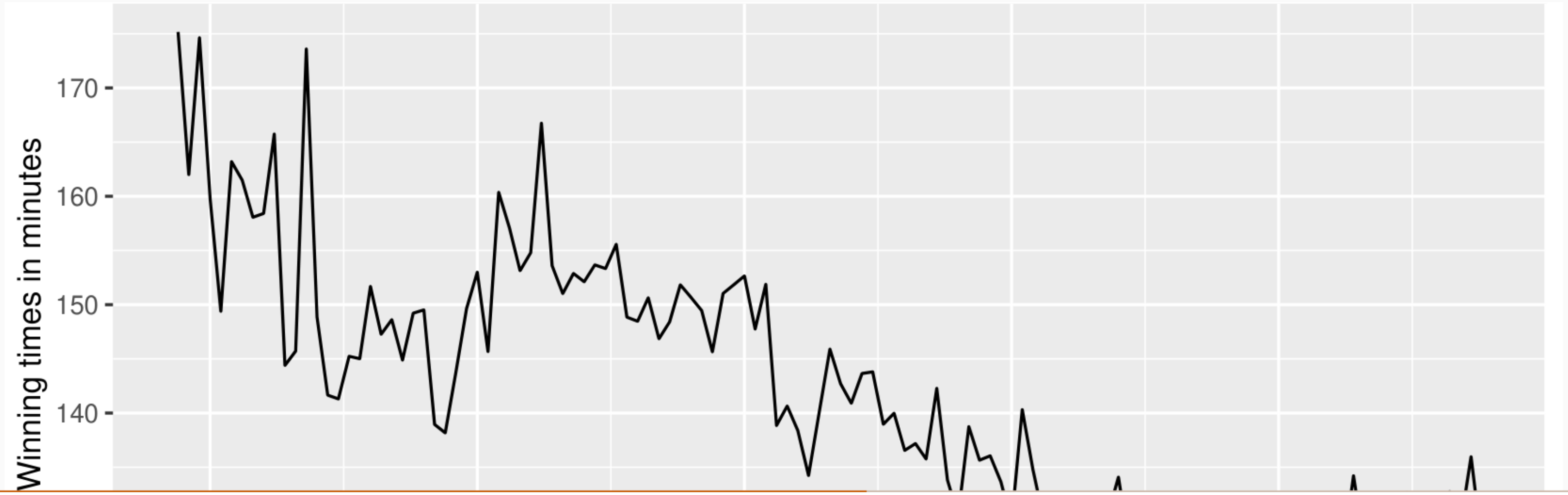
$$x_{1,t} = t, \quad x_{2,t} = t^2, \quad \ldots$$

NOT RECOMMENDED!

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Boston marathon winning times

```
marathon <- boston_marathon %>%
  filter(Event == "Men's open division") %>%
  select(-Event) %>%
  mutate(Minutes = as.numeric(Time)/60)
marathon %>% autoplot(Minutes) +
  labs(y="Winning times in minutes")
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Boston marathon winning times

```
fit_trends <- marathon %>%
  model(
    # Linear trend
    linear = TSLM(Minutes ~ trend()),
    # Exponential trend
    exponential = TSLM(log(Minutes) ~ trend()),
    # Piecewise linear trend
    piecewise = TSLM(Minutes ~ trend(knots = c(1940, 1980)))
  )
```

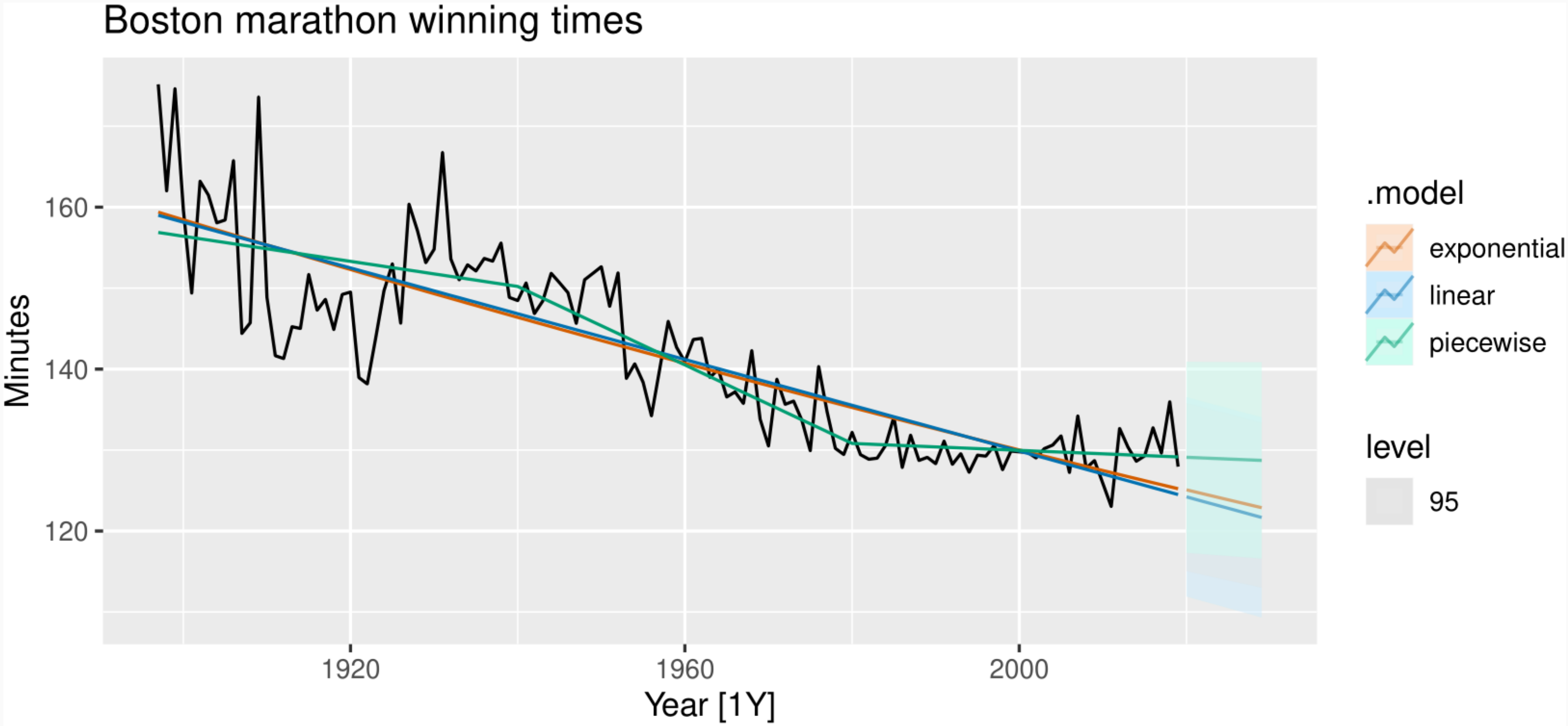
```
fit_trends
```

```
## # A mable: 1 x 3
##     linear exponential piecewise
##    <model>      <model>   <model>
## 1   <TSLM>       <TSLM>    <TSLM>
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Boston marathon winning times

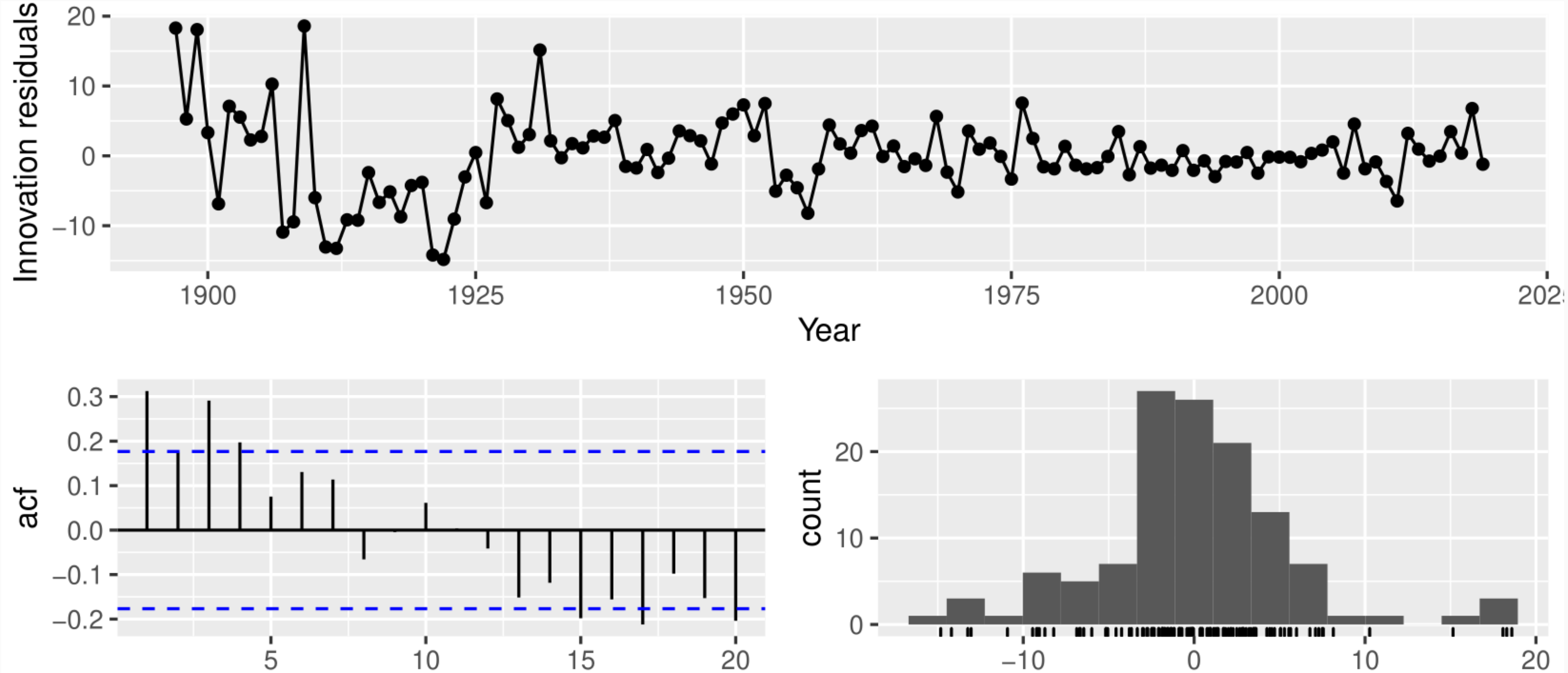
```
fit_trends %>% forecast(h=10) %>% autoplot(marathon)
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Boston marathon winning times

```
fit_trends %>% select(piecewise) %>%
  gg_tsresiduals()
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Outline

Multiple regression and forecasting

For forecasting purposes, we require the following assumptions:

- ε_t are uncorrelated and zero mean
- ε_t are uncorrelated with each $x_{j,t}$.

It is **useful** to also have $\varepsilon_t \sim \text{N}(0, \sigma^2)$ when producing prediction intervals or doing statistical tests.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Residual plots

Useful for spotting outliers and whether the linear model was appropriate.

- Scatterplot of residuals ε_t against each predictor $x_{j,t}$.
- Scatterplot residuals against the fitted values $\hat{y}_t$
- Expect to see scatterplots resembling a horizontal band with no values too far from the band and no patterns such as curvature or increasing spread.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Residual patterns

- If a plot of the residuals vs any predictor in the model shows a pattern, then the relationship is nonlinear.
- If a plot of the residuals vs any predictor **not** in the model shows a pattern, then the predictor should be added to the model.
- If a plot of the residuals vs fitted values shows a pattern, then there is heteroscedasticity in the errors. (Could try a transformation.)

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Outline

Comparing regression models

Computer output for regression will always give the R^2 value. This is a useful summary of the model.

- It is equal to the square of the correlation between y and $\hat{y}$.
- It is often called the "coefficient of determination' '.
- It can also be calculated as follows:
$$R^2 = \frac{\sum(\hat{y}_t - \bar{y})^2}{\sum(y_t - \bar{y})^2}$$
- It is the proportion of variance accounted for (explained) by the predictors.

Comparing regression models

However . . .

- R^2 does not allow for "degrees of freedom''.
- Adding *any* variable tends to increase the value of R^2, even if that variable is irrelevant.

To overcome this problem, we can use *adjusted* R^2:

$$\bar{R}^2 = 1 - (1 - R^2)\frac{T-1}{T-k-1}$$

where k = no. predictors and T = no. observations.

Maximizing $\bar{R}^2$ is equivalent to minimizing $\hat{\sigma}^2$.

$$\hat{\sigma}^2 = \frac{1}{T-k-1}\sum_{t=1}^{T}\varepsilon_t^2$$

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Akaike's Information Criterion

$$\text{AIC} = -2\log(L) + 2(k+2)$$

where L is the likelihood and k is the number of predictors in the model.

- AIC penalizes terms more heavily than $\bar{R}^2$.
- Minimizing the AIC is asymptotically equivalent to minimizing MSE via **leave-one-out cross-validation** (for any linear regression).

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Corrected AIC

For small values of T, the AIC tends to select too many predictors, and so a bias-corrected version of the AIC has been developed.

$$\text{AIC}_{\text{C}} = \text{AIC} + \frac{2(k+2)(k+3)}{T-k-3}$$

As with the AIC, the AIC_{C} should be minimized.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Bayesian Information Criterion

$$\text{BIC} = -2\log(L) + (k+2)\log(T)$$

where L is the likelihood and k is the number of predictors in the model.

- BIC penalizes terms more heavily than AIC
- Also called SBIC and SC.
- Minimizing BIC is asymptotically equivalent to leave-v-out cross-validation when $v = T[1 - 1/(log(T) - 1)]$.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Leave-one-out cross-validation

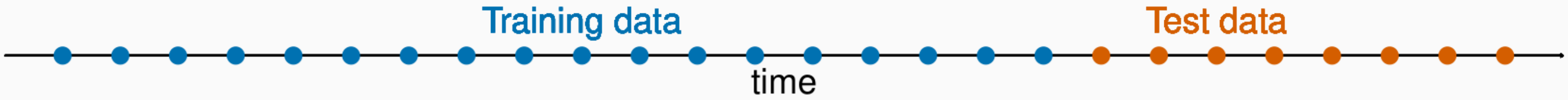
For regression, leave-one-out cross-validation is faster and more efficient than time-series cross-validation.

- Select one observation for test set, and use *remaining* observations in training set. Compute error on test observation.
- Repeat using each possible observation as the test set.
- Compute accuracy measure over all errors.

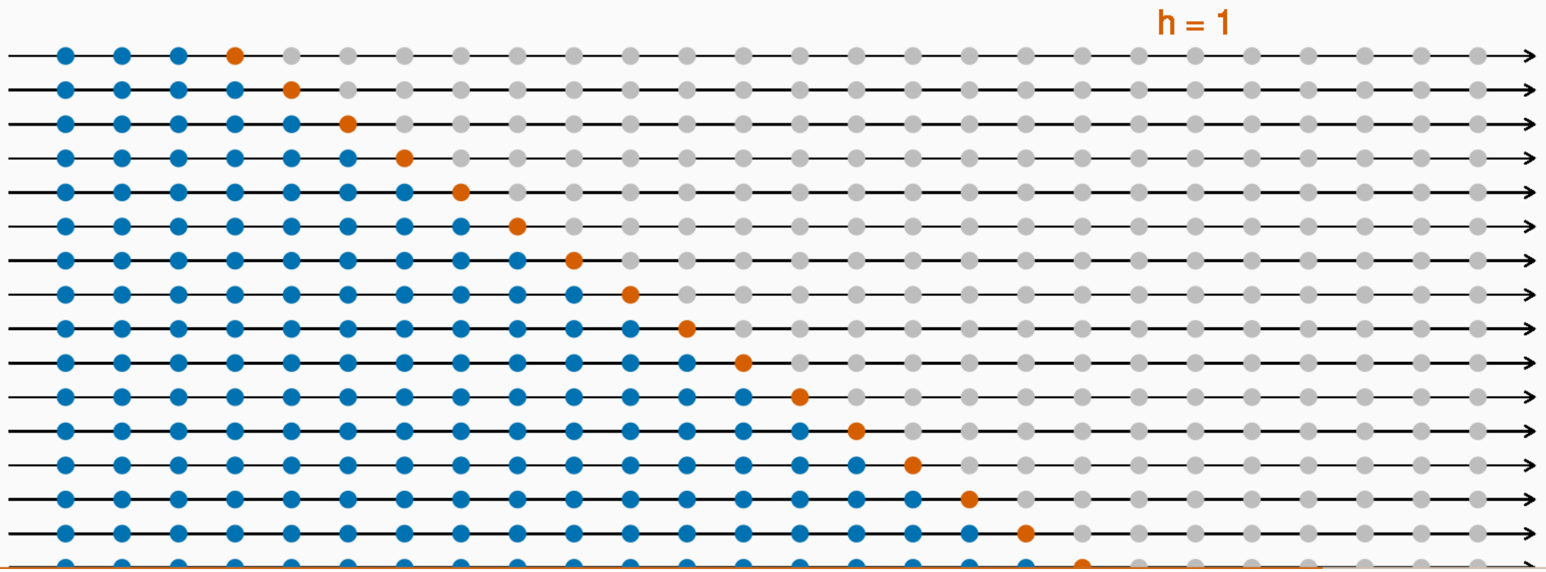
Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Cross-validation

Traditional evaluation



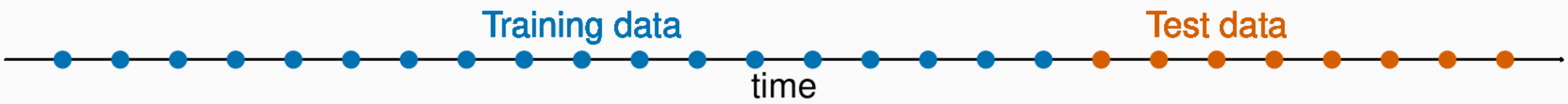
Time series cross-validation



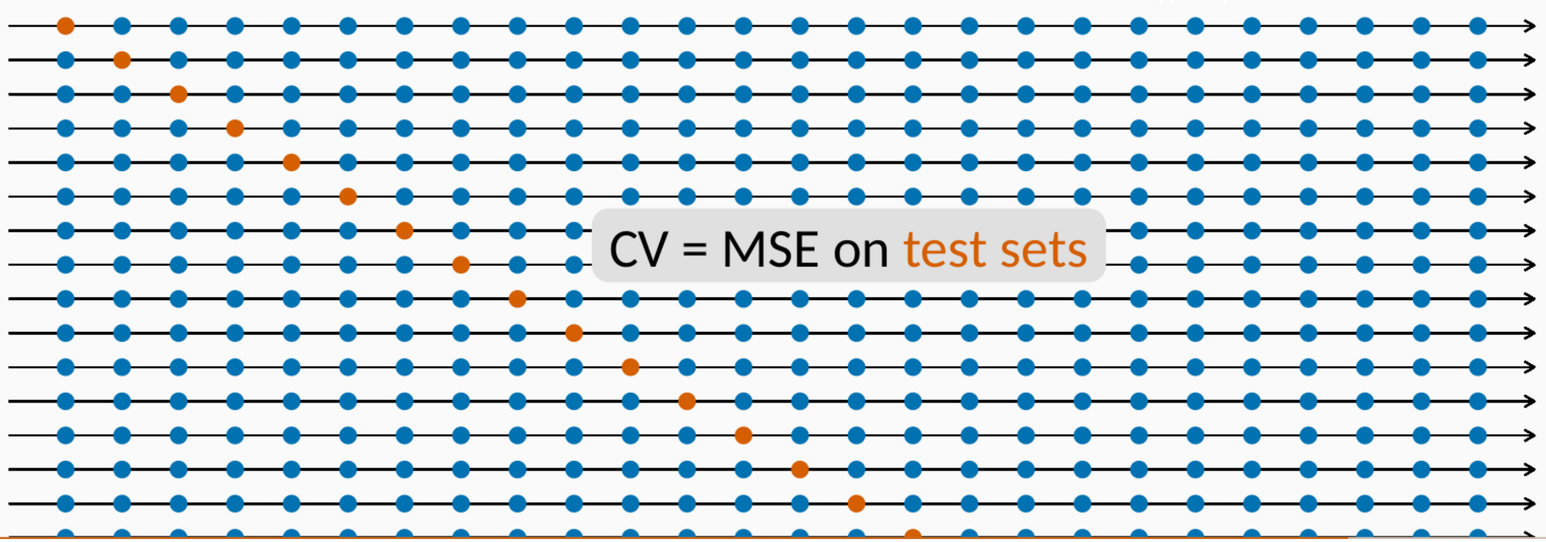
Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Cross-validation

Traditional evaluation



Leave-one-out cross-validation



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Comparing regression models

```
glance(fit_trends) %>%
  select(.model, r_squared, adj_r_squared, AICc, CV)
```

```
## # A tibble: 3 x 5
##   .model      r_squared adj_r_squared  AICc       CV
##   <chr>           <dbl>         <dbl> <dbl>    <dbl>
## 1 linear          0.728         0.726  452. 39.1
## 2 exponential     0.744         0.742 -779.  0.00176
## 3 piecewise       0.767         0.761  438. 34.8
```

- Be careful making comparisons when transformations are used.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Choosing regression variables

Best subsets regression

- Fit all possible regression models using one or more of the predictors.
- Choose the best model based on one of the measures of predictive ability (CV, AIC, AICc).

Warning!

- If there are a large number of predictors, this is not possible.
- For example, 44 predictors leads to 18 trillion possible models!

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Choosing regression variables

Backwards stepwise regression

- Start with a model containing all variables.
- Try subtracting one variable at a time. Keep the model if it has lower CV or AICc.
- Iterate until no further improvement.

Notes

- Stepwise regression is not guaranteed to lead to the best possible model.
- Inference on coefficients of final model will be wrong.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Outline

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Ex-ante versus ex-post forecasts

- *Ex ante forecasts* are made using only information available in advance.
  - require forecasts of predictors
- *Ex post forecasts* are made using later information on the predictors.
  - useful for studying behaviour of forecasting models.
- trend, seasonal and calendar variables are all known in advance, so these don't need to be forecast.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Scenario based forecasting

- Assumes possible scenarios for the predictor variables
- Prediction intervals for scenario based forecasts do not include the uncertainty associated with the future values of the predictor variables.

Building a predictive regression model

- If getting forecasts of predictors is difficult, you can use lagged predictors instead.

$$y_t = \beta_0 + \beta_1 x_{1,t-h} + \cdots + \beta_k x_{k,t-h} + \varepsilon_t$$

- A different model for each forecast horizon h.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

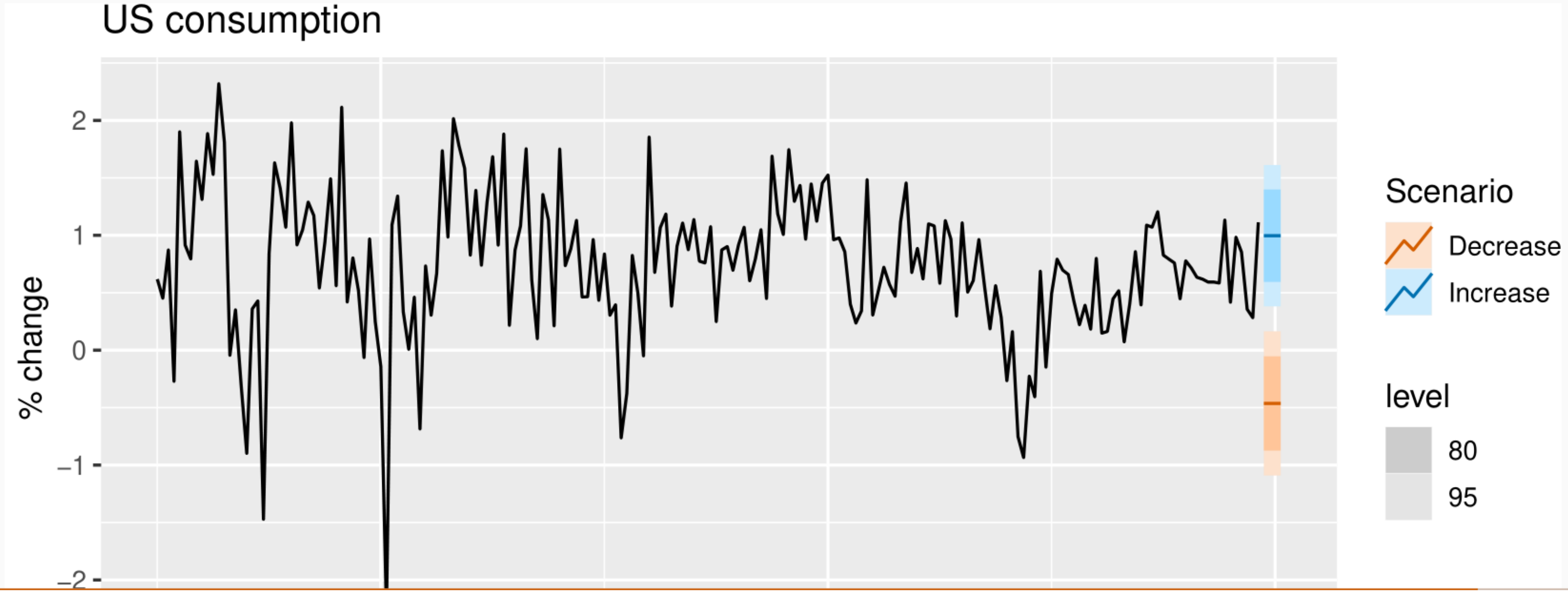
US Consumption

```
fit_consBest <- us_change %>%
  model(
    TSLM(Consumption ~ Income + Savings + Unemployment)
  )

future_scenarios <- scenarios(
  Increase = new_data(us_change, 4) %>%
    mutate(Income=1, Savings=0.5, Unemployment=0),
  Decrease = new_data(us_change, 4) %>%
    mutate(Income=-1, Savings=-0.5, Unemployment=0),
  names_to = "Scenario")

fc <- forecast(fit_consBest, new_data = future_scenarios)
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

US Consumption

```
us_change %>% autoplot(Consumption) +
  labs(y="% change in US consumption") +
  autolayer(fc) +
  labs(title = "US consumption", y = "% change")
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Outline

Matrix formulation

$$y_t = \beta_0 + \beta_1 x_{1,t} + \beta_2 x_{2,t} + \cdots + \beta_k x_{k,t} + \varepsilon_t.$$

Let $\boldsymbol{y} = (y_1, \dots, y_T)'$, $\boldsymbol{\varepsilon} = (\varepsilon_1, \dots, \varepsilon_T)'$, $\boldsymbol{\beta} = (\beta_0, \beta_1, \dots, \beta_k)'$ and

$$\boldsymbol{X} = \begin{bmatrix} 1 & x_{1,1} & x_{2,1} & \dots & x_{k,1} \\ 1 & x_{1,2} & x_{2,2} & \dots & x_{k,2} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{1,T} & x_{2,T} & \dots & x_{k,T} \end{bmatrix}.$$

Then

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}.$$

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Matrix formulation

Least squares estimation

Minimize: $(\boldsymbol{y} - \boldsymbol{X}\beta)'(\boldsymbol{y} - \boldsymbol{X}\beta)$

Differentiate wrt β gives

$$\hat{\beta} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y}$$

(The "normal equation".)

$$\hat{\sigma}^2 = \frac{1}{T-k-1}(\boldsymbol{y} - \boldsymbol{X}\hat{\beta})'(\boldsymbol{y} - \boldsymbol{X}\hat{\beta})$$

Note: If you fall for the dummy variable trap, $(\boldsymbol{X}'\boldsymbol{X})$ is a singular matrix.

Likelihood

If the errors are iid and normally distributed, then

$$\boldsymbol{y} \sim \mathrm{N}(\boldsymbol{X}\beta, \sigma^2 \boldsymbol{I}).$$

So the likelihood is

$$L = \frac{1}{\sigma^T (2\pi)^{T/2}} \exp\left(-\frac{1}{2\sigma^2}(\boldsymbol{y} - \boldsymbol{X}\beta)'(\boldsymbol{y} - \boldsymbol{X}\beta)\right)$$

which is maximized when $(\boldsymbol{y} - \boldsymbol{X}\beta)'(\boldsymbol{y} - \boldsymbol{X}\beta)$ is minimized.

So MLE = OLS.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Multiple regression forecasts

Optimal forecasts

$$\hat{y}^* = \text{E}(y^* | \boldsymbol{y}, \boldsymbol{X}, \boldsymbol{x}^*) = \boldsymbol{x}^* \hat{\boldsymbol{\beta}} = \boldsymbol{x}^* (\boldsymbol{X}'\boldsymbol{X})^{-1} \boldsymbol{X}'\boldsymbol{y}$$

where $\boldsymbol{x}^*$ is a row vector containing the values of the predictors for the forecasts (in the same format as $\boldsymbol{X}$).

Forecast variance

$$\text{Var}(y^* | \boldsymbol{X}, \boldsymbol{x}^*) = \sigma^2 \left[1 + \boldsymbol{x}^* (\boldsymbol{X}'\boldsymbol{X})^{-1} (\boldsymbol{x}^*)'\right]$$

- This ignores any errors in $\boldsymbol{x}^*$.
- 95% prediction intervals assuming normal errors:

$$\hat{y}^* \pm 1.96 \sqrt{\text{Var}(y^* | \boldsymbol{X}, \boldsymbol{x}^*)}.$$

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Outline

Correlation is not causation

- When x is useful for predicting y, it is not necessarily causing y.
- e.g., predict number of drownings y using number of ice-creams sold x.
- Correlations are useful for forecasting, even when there is no causality.
- Better models usually involve causal relationships (e.g., temperature x and people z to predict drownings y).

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Multicollinearity

In regression analysis, multicollinearity occurs when:

- Two predictors are highly correlated (i.e., the correlation between them is close to ± 1).
- A linear combination of some of the predictors is highly correlated with another predictor.
- A linear combination of one subset of predictors is highly correlated with a linear combination of another subset of predictors.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Multicollinearity

If multicollinearity exists. . .

- the numerical estimates of coefficients may be wrong (worse in Excel than in a statistics package)
- don't rely on the *p*-values to determine significance.
- there is no problem with model *predictions* provided the predictors used for forecasting are within the range used for fitting.
- omitting variables can help.
- combining variables can help.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Outline

Historical perspective

- Developed in the 1950s and 1960s as methods (algorithms) to produce point forecasts.
- Combine a “level”, “trend” (slope) and “seasonal” component to describe a time series.
- The rate of change of the components are controlled by “smoothing parameters”: α, β and γ respectively.
- Need to choose best values for the smoothing parameters (and initial states).
- Equivalent ETS state space models developed in the 1990s and 2000s.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Big idea: control the rate of change

α controls the flexibility of the **level**

- If α = 0, the level never updates (mean)
- If α = 1, the level updates completely (naive)

β controls the flexibility of the **trend**

- If β = 0, the trend is linear
- If β = 1, the trend changes suddenly every observation

γ controls the flexibility of the **seasonality**

- If γ = 0, the seasonality is fixed (seasonal means)
- If γ = 1, the seasonality updates completely (seasonal naive)

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

A model for levels, trends, and seasonalities

We want a model that captures the level (ℓ_t), trend (b_t) and seasonality (s_t).

How do we combine these elements?

Additively?

$$y_t = \ell_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t$$

Multiplicatively?

$$y_t = \ell_{t-1} b_{t-1} s_{t-m}(1 + \varepsilon_t)$$

Perhaps a mix of both?

$$y_t = (\ell_{t-1} + b_{t-1}) s_{t-m} + \varepsilon_t$$

How do the level, trend and seasonal components evolve over time?

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

ETS models

General notation E T S : **E**xponen**T**ial **S**moothing

Error **T**rend **S**eason

Error: Additive (`"A"`) or multiplicative (`"M"`)

Trend: None (`"N"`), additive (`"A"`), multiplicative (`"M"`), or damped (`"Ad"` or `"Md"`).

Seasonality: None (`"N"`), additive (`"A"`) or multiplicative (`"M"`)

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Outline

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Simple methods

Time series $y_1, y_2, \ldots, y_T$.

Random walk forecasts

$$\hat{y}_{T+h|T} = y_T$$

Average forecasts

$$\hat{y}_{T+h|T} = \frac{1}{T}\sum_{t=1}^{T} y_t$$

- Want something in between these methods.
- Most recent data should have more weight.

Simple Exponential Smoothing

Forecast equation

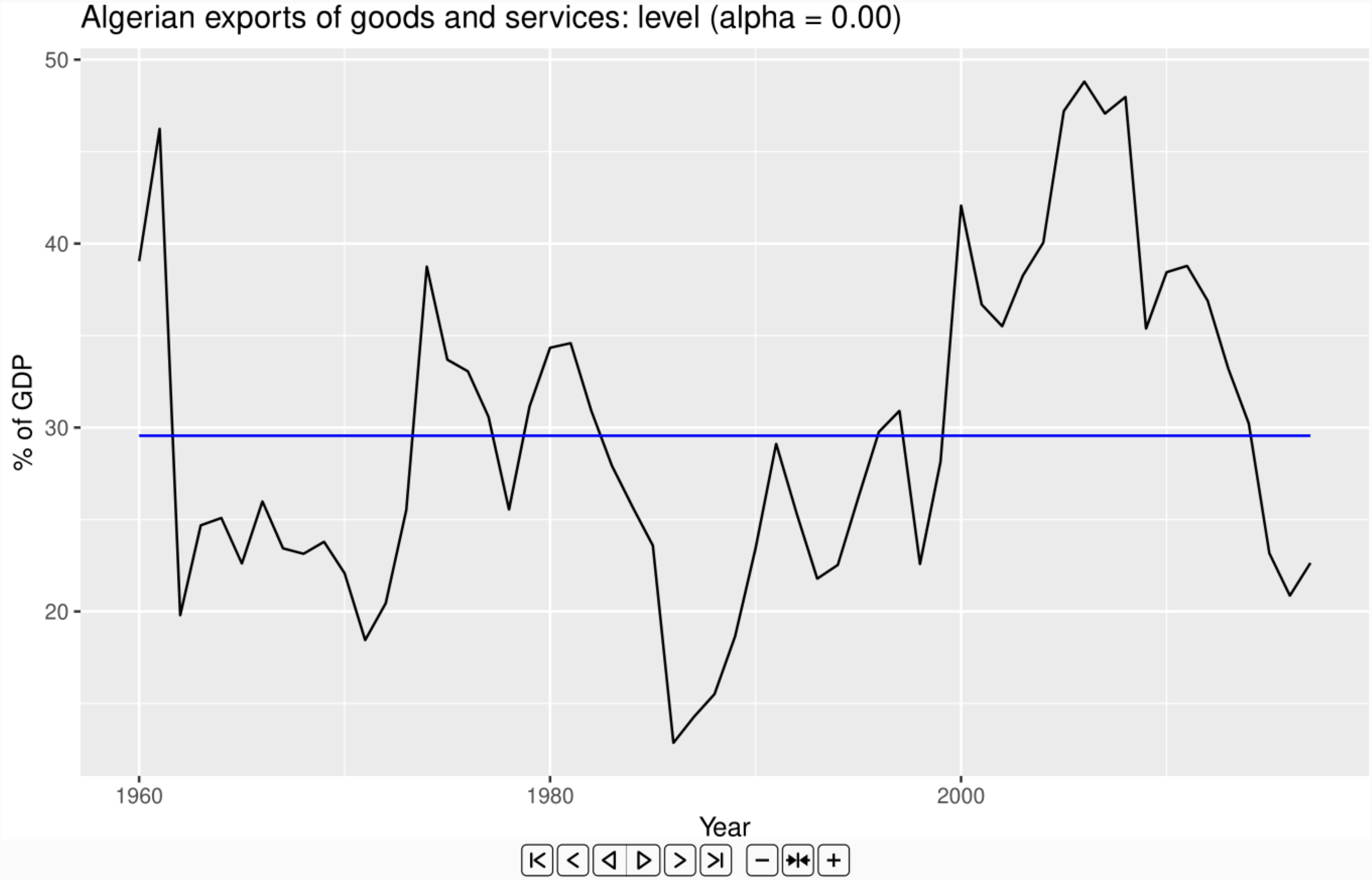
$$\hat{y}_{T+1|T} = \alpha y_T + \alpha(1-\alpha)y_{T-1} + \alpha(1-\alpha)^2 y_{T-2} + \cdots,$$

where $0 \leq \alpha \leq 1$.

	Weights assigned to observations for:			
Observation	$\alpha = 0.2$	$\alpha = 0.4$	$\alpha = 0.6$	$\alpha = 0.8$
y_T	0.2	0.4	0.6	0.8
y_{T-1}	0.16	0.24	0.24	0.16
y_{T-2}	0.128	0.144	0.096	0.032
y_{T-3}	0.1024	0.0864	0.0384	0.0064
y_{T-4}	$(0.2)(0.8)^4$	$(0.4)(0.6)^4$	$(0.6)(0.4)^4$	$(0.8)(0.2)^4$
y_{T-5}	$(0.2)(0.8)^5$	$(0.4)(0.6)^5$	$(0.6)(0.4)^5$	$(0.8)(0.2)^5$

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Simple Exponential Smoothing



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Simple Exponential Smoothing

Component form

Forecast equation	$\hat{y}_{t+h\|t} = \ell_t$
Smoothing equation	$\ell_t = \alpha y_t + (1-\alpha)\ell_{t-1}$

- ℓ_t is the level (or the smoothed value) of the series at time t.
- $\hat{y}_{t+1|t} = \alpha y_t + (1-\alpha)\hat{y}_{t|t-1}$

Iterate to get exponentially weighted moving average form.

Weighted average form

$$\hat{y}_{T+1|T} = \sum_{j=0}^{T-1} \alpha(1-\alpha)^j y_{T-j} + (1-\alpha)^T \ell_0$$

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Optimising smoothing parameters

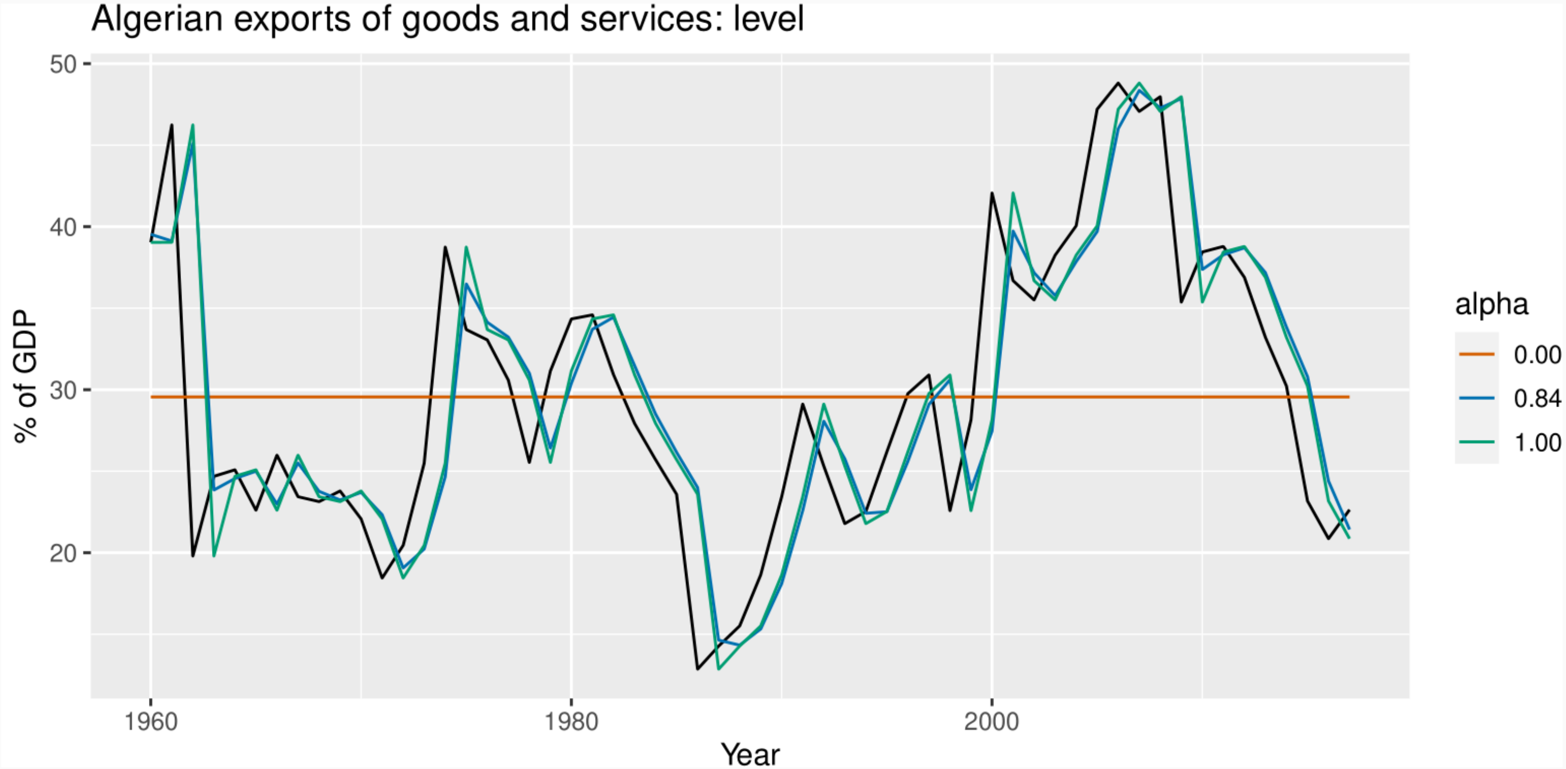
- Need to choose best values for α and ℓ_0.
- Similarly to regression, choose optimal parameters by minimising SSE:

$$\text{SSE} = \sum_{t=1}^{T}(y_t - \hat{y}_{t|t-1})^2.$$

- Unlike regression there is no closed form solution — use numerical optimization.
- For Algerian Exports example:
  - $\hat{\alpha} = 0.8400$
  - $\hat{\ell}_0 = 39.54$

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Simple Exponential Smoothing



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Models and methods

Methods

- Algorithms that return point forecasts.

Models

- Generate same point forecasts but can also generate forecast distributions.
- A stochastic (or random) data generating process that can generate an entire forecast distribution.
- Allow for “proper” model selection.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

ETS(A,N,N): SES with additive errors

Component form

Forecast equation $\hat{y}_{t+h|t} = \ell_t$

Smoothing equation $\ell_t = \alpha y_t + (1 - \alpha)\ell_{t-1}$

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

Error correction form

$$y_t = \ell_{t-1} + e_t$$

$$\ell_t = \ell_{t-1} + \alpha(y_t - \ell_{t-1})$$

$$= \ell_{t-1} + \alpha e_t$$

Specify probability distribution for e_t, we assume $e_t = \varepsilon_t \sim \text{NID}(0, \sigma^2)$.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

ETS(A,N,N): SES with additive errors

Measurement equation	$y_t = \ell_{t-1} + \varepsilon_t$
State equation	$\ell_t = \ell_{t-1} + \alpha\varepsilon_t$

where $\varepsilon_t \sim \text{NID}(0, \sigma^2)$.

- "innovations" or "single source of error" because equations have the same error process, ε_t.
- Measurement equation: relationship between observations and states.
- State equation(s): evolution of the state(s) through time.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

ETS(M,N,N): SES with multiplicative errors.

- Specify relative errors $\varepsilon_t = \frac{y_t - \hat{y}_{t|t-1}}{\hat{y}_{t|t-1}} \sim \text{NID}(0, \sigma^2)$
- Substituting $\hat{y}_{t|t-1} = \ell_{t-1}$ gives:
  - $y_t = \ell_{t-1} + \ell_{t-1}\varepsilon_t$
  - $e_t = y_t - \hat{y}_{t|t-1} = \ell_{t-1}\varepsilon_t$

Measurement equation	$y_t = \ell_{t-1}(1 + \varepsilon_t)$
State equation	$\ell_t = \ell_{t-1}(1 + \alpha\varepsilon_t)$

- Models with additive and multiplicative errors with the same parameters generate the same point forecasts but different prediction intervals.

ETS(A,N,N): Specifying the model

```
ETS(y ~ error("A") + trend("N") + season("N"))
```

By default, an optimal value for α and ℓ_0 is used.

α can be chosen manually in `trend()`.

```
trend("N", alpha = 0.5)
trend("N", alpha_range = c(0.2, 0.8))
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Algerian Exports

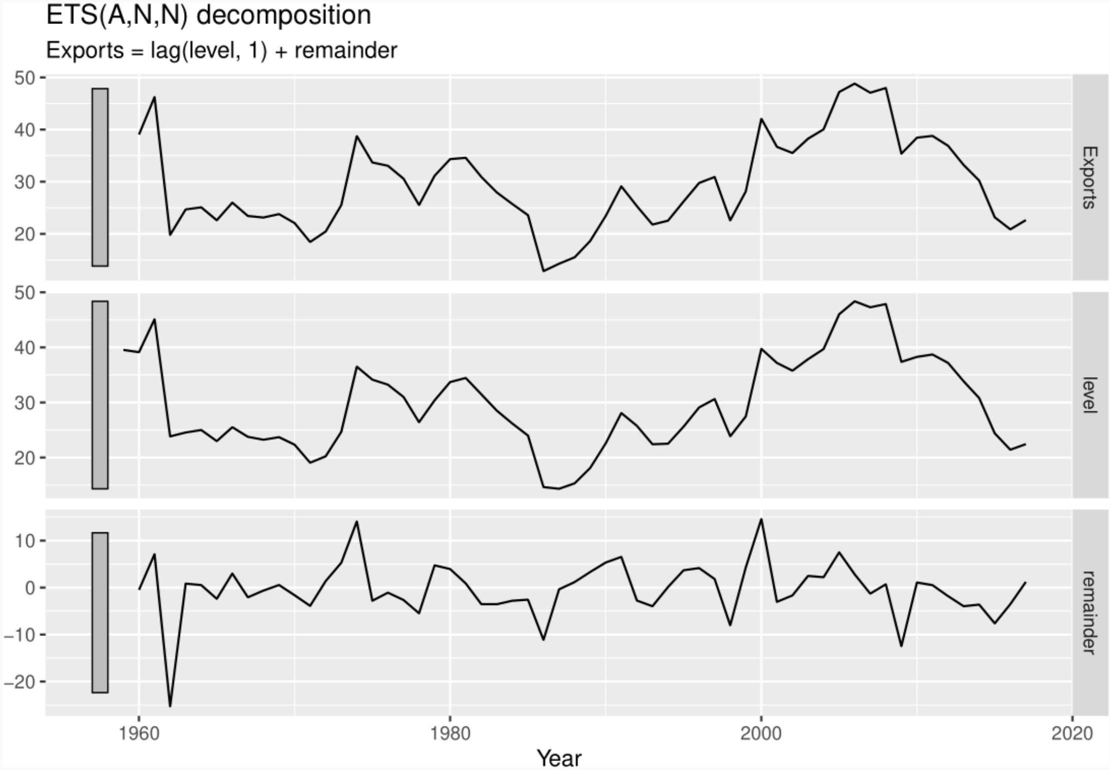
```r
algeria_economy <- global_economy %>%
  filter(Country == "Algeria")
fit <- algeria_economy %>%
  model(ANN = ETS(Exports ~ error("A") + trend("N") + season("N")))
report(fit)
```

```
## Series: Exports
## Model: ETS(A,N,N)
##   Smoothing parameters:
##     alpha = 0.84
##
##   Initial states:
##  l[0]
##  39.5
##
##   sigma^2:  35.6
##
##  AIC AICc  BIC
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Algerian Exports

```
components(fit) %>% autoplot()
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Algerian Exports

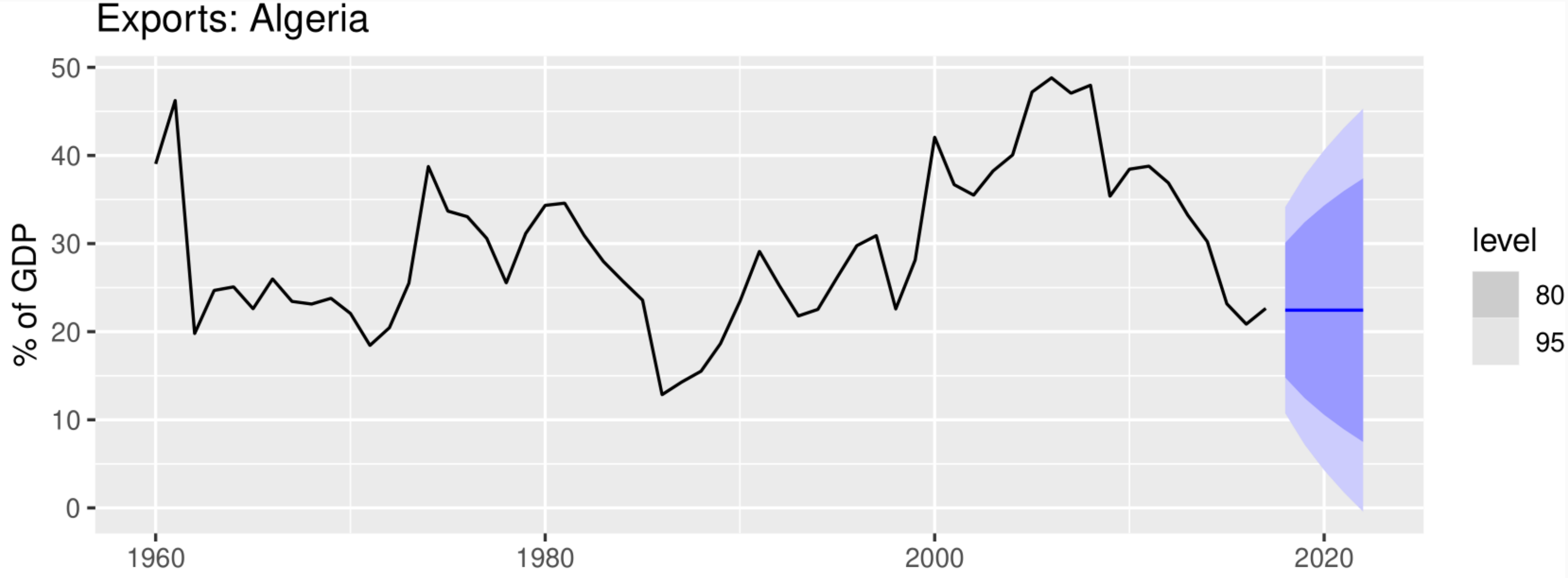
```r
components(fit) %>%
  left_join(fitted(fit), by = c("Country", ".model", "Year"))
```

```
## # A dable: 59 x 7 [1Y]
## # Key:     Country, .model [1]
## # :        Exports = lag(level, 1) + remainder
##    Country .model  Year Exports level remainder .fitted
##    <fct>   <chr>  <dbl>   <dbl> <dbl>     <dbl>   <dbl>
##  1 Algeria ANN     1959    NA    39.5    NA        NA
##  2 Algeria ANN     1960    39.0  39.1    -0.496    39.5
##  3 Algeria ANN     1961    46.2  45.1     7.12     39.1
##  4 Algeria ANN     1962    19.8  23.8   -25.3      45.1
##  5 Algeria ANN     1963    24.7  24.6     0.841    23.8
##  6 Algeria ANN     1964    25.1  25.0     0.534    24.6
##  7 Algeria ANN     1965    22.6  23.0    -2.39     25.0
##  8 Algeria ANN     1966    26.0  25.5     3.00     23.0
##  9 Algeria ANN     1967    23.4  23.8    -2.07     25.5
## 10 Algeria ANN     1968    23.1  23.2    -0.630    23.8
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Algerian Exports

```
fit %>%
  forecast(h = 5) %>%
  autoplot(algeria_economy) +
  labs(y = "% of GDP", title = "Exports: Algeria")
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Outline

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Holt's linear trend

Component form

$$\begin{aligned} \text{Forecast} \quad & \hat{y}_{t+h|t} = \ell_t + hb_t \\ \text{Level} \quad & \ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\ \text{Trend} \quad & b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}, \end{aligned}$$

- Two smoothing parameters α and β^* ($0 \leq \alpha, \beta^* \leq 1$).
- ℓ_t level: weighted average between y_t and one-step ahead forecast for time t, ($\ell_{t-1} + b_{t-1} = \hat{y}_{t|t-1}$)
- b_t slope: weighted average of $(\ell_t - \ell_{t-1})$ and b_{t-1}, current and previous estimate of slope.
- Choose $\alpha, \beta^*, \ell_0, b_0$ to minimise SSE.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

ETS(A,A,N)

Holt's linear method with additive errors.

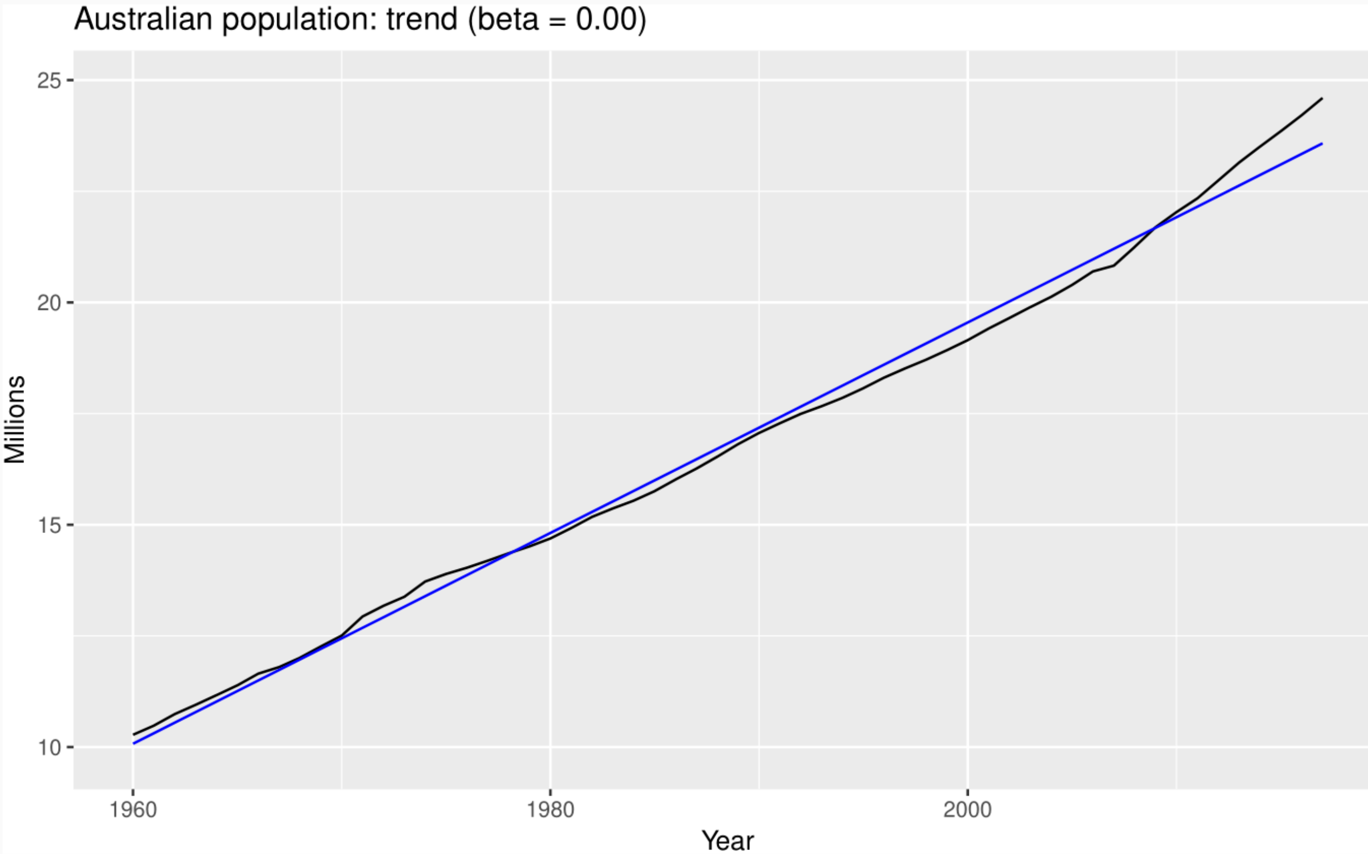
- Assume $\varepsilon_t = y_t - \ell_{t-1} - b_{t-1} \sim \text{NID}(0, \sigma^2)$.
- Substituting into the error correction equations for Holt's linear method

$$
\begin{aligned}
y_t &= \ell_{t-1} + b_{t-1} + \varepsilon_t \\
\ell_t &= \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t \\
b_t &= b_{t-1} + \alpha\beta^*\varepsilon_t
\end{aligned}
$$

- For simplicity, set $\beta = \alpha\beta^*$.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Exponential smoothing: trend/slope



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

ETS(M,A,N)

Holt's linear method with multiplicative errors.

- Assume $\varepsilon_t = \frac{y_t - (\ell_{t-1} + b_{t-1})}{(\ell_{t-1} + b_{t-1})}$
- Following a similar approach as above, the innovations state space model underlying Holt's linear method with multiplicative errors is specified as

$$
\begin{aligned}
y_t &= (\ell_{t-1} + b_{t-1})(1 + \varepsilon_t) \\
\ell_t &= (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t) \\
b_t &= b_{t-1} + \beta(\ell_{t-1} + b_{t-1})\varepsilon_t
\end{aligned}
$$

where again $\beta = \alpha\beta^*$ and $\varepsilon_t \sim \text{NID}(0, \sigma^2)$.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

ETS(A,A,N): Specifying the model

```
ETS(y ~ error("A") + trend("A") + season("N"))
```

By default, optimal values for β and b_0 are used.

β can be chosen manually in `trend()`.

```
trend("A", beta = 0.004)
trend("A", beta_range = c(0, 0.1))
```

Example: Australian population

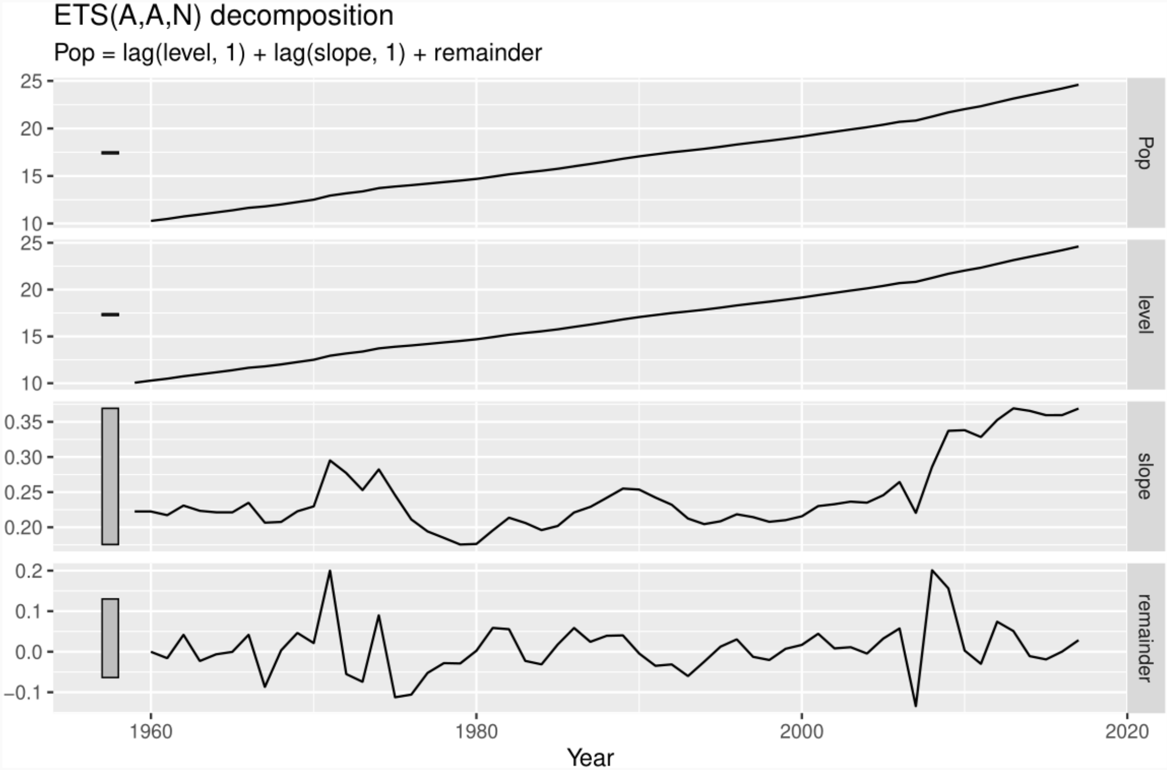
```
aus_economy <- global_economy %>% filter(Code == "AUS") %>%
  mutate(Pop = Population / 1e6)
fit <- aus_economy %>%
  model(AAN = ETS(Pop ~ error("A") + trend("A") + season("N")))
report(fit)
```

```
## Series: Pop
## Model: ETS(A,A,N)
##   Smoothing parameters:
##     alpha = 1
##     beta  = 0.327
##
##   Initial states:
##  l[0]  b[0]
##  10.1 0.222
##
##   sigma^2:  0.0041
##
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Australian population

```
components(fit) %>% autoplot()
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

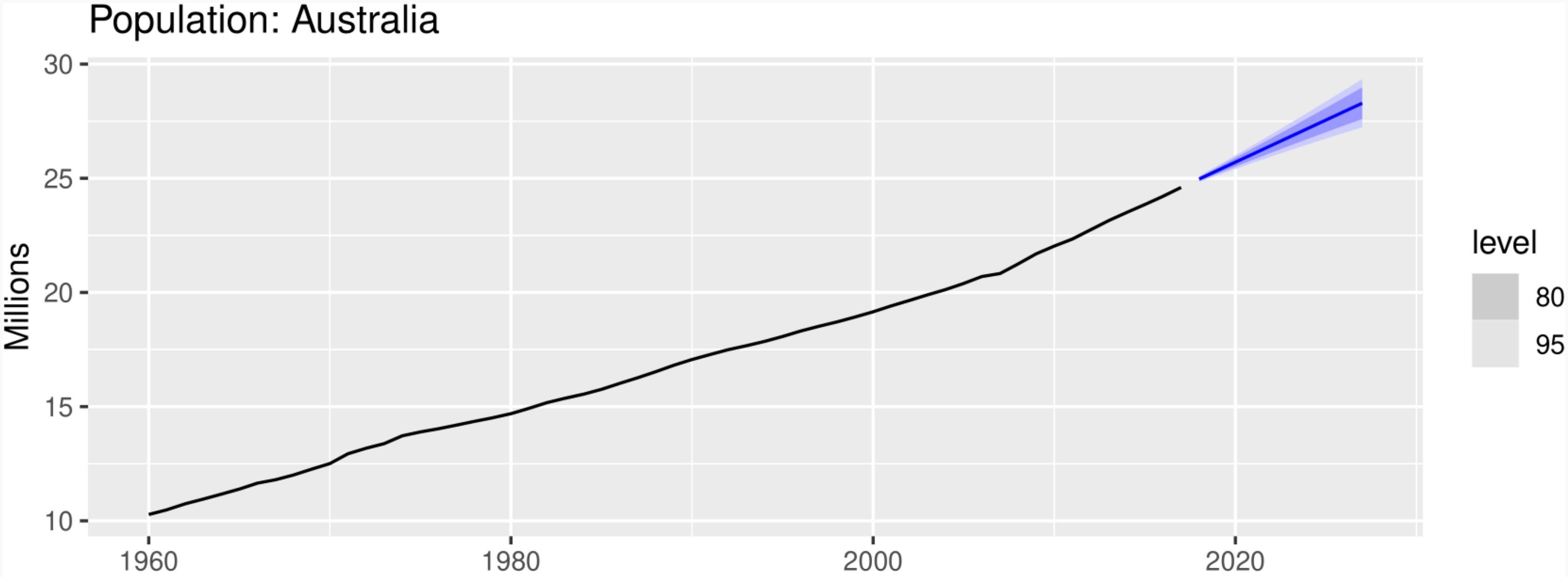
Example: Australian population

```
components(fit) %>%
  left_join(fitted(fit), by = c("Country", ".model", "Year"))
```

```
## # A dable: 59 x 8 [1Y]
## # Key:     Country, .model [1]
## # :        Pop = lag(level, 1) + lag(slope, 1) + remainder
##    Country   .model  Year   Pop level slope remainder .fitted
##    <fct>     <chr>  <dbl> <dbl> <dbl> <dbl>     <dbl>   <dbl>
##  1 Australia AAN     1959  NA    10.1 0.222 NA           NA
##  2 Australia AAN     1960  10.3  10.3 0.222 -0.000145    10.3
##  3 Australia AAN     1961  10.5  10.5 0.217 -0.0159      10.5
##  4 Australia AAN     1962  10.7  10.7 0.231  0.0418      10.7
##  5 Australia AAN     1963  11.0  11.0 0.223 -0.0229      11.0
##  6 Australia AAN     1964  11.2  11.2 0.221 -0.00641     11.2
##  7 Australia AAN     1965  11.4  11.4 0.221 -0.000314    11.4
##  8 Australia AAN     1966  11.7  11.7 0.235  0.0418      11.6
##  9 Australia AAN     1967  11.8  11.8 0.206 -0.0869      11.9
## 10 Australia AAN     1968  12.0  12.0 0.208  0.00350     12.0
```

Example: Australian population

```r
fit %>%
  forecast(h = 10) %>%
  autoplot(aus_economy) +
  labs(y = "Millions", title = "Population: Australia")
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Damped trend method

Component form

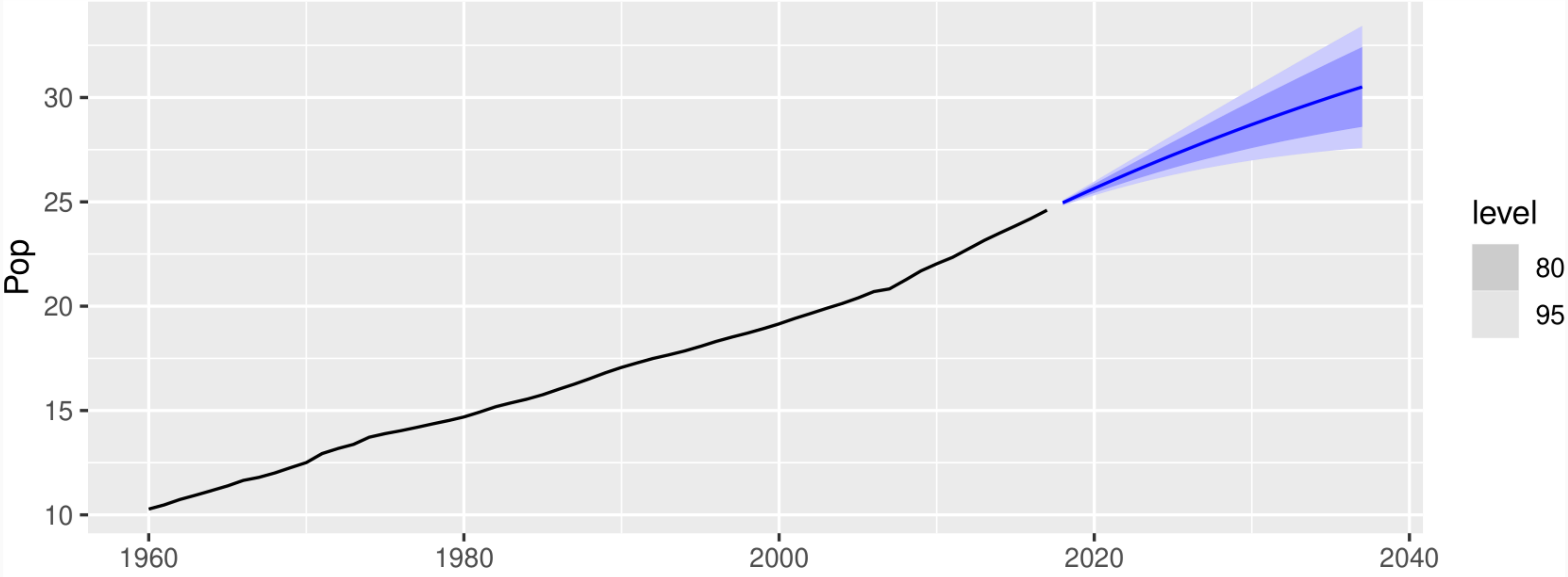
$$\hat{y}_{t+h|t} = \ell_t + (\phi + \phi^2 + \cdots + \phi^h)b_t$$
$$\ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + \phi b_{t-1})$$
$$b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1}.$$

- Damping parameter $0 < \phi < 1$.
- If $\phi = 1$, identical to Holt's linear trend.
- As $h \to \infty$, $\hat{y}_{T+h|T} \to \ell_T + \phi b_T/(1-\phi)$.
- Short-run forecasts trended, long-run forecasts constant.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Australian population

```
aus_economy %>%
  model(holt = ETS(Pop ~ error("A") + trend("Ad") + season("N"))) %>%
  forecast(h = 20) %>%
  autoplot(aus_economy)
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Australian population

```r
fit <- aus_economy %>%
  filter(Year <= 2010) %>%
  model(
    ses = ETS(Pop ~ error("A") + trend("N") + season("N")),
    holt = ETS(Pop ~ error("A") + trend("A") + season("N")),
    damped = ETS(Pop ~ error("A") + trend("Ad") + season("N"))
  )
```

```r
tidy(fit)
accuracy(fit)
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Australian population

term	SES	Linear trend	Damped trend
α	1.00	1.00	1.00
β^*		0.30	0.40
ϕ			0.98
NA		0.22	0.25
NA	10.28	10.05	10.04
Training RMSE	0.24	0.06	0.07
Test RMSE	1.63	0.15	0.21
Test MASE	6.18	0.55	0.75
Test MAPE	6.09	0.55	0.74
Test MAE	1.45	0.13	0.18

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Outline

Holt-Winters additive method

Holt and Winters extended Holt's method to capture seasonality.

Component form

$$
\begin{aligned}
\hat{y}_{t+h|t} &= \ell_t + hb_t + s_{t+h-m(k+1)} \\
\ell_t &= \alpha(y_t - s_{t-m}) + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\
b_t &= \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1} \\
s_t &= \gamma(y_t - \ell_{t-1} - b_{t-1}) + (1-\gamma)s_{t-m}
\end{aligned}
$$

- k = integer part of $(h-1)/m$. Ensures estimates from the final year are used for forecasting.
- Parameters: $0 \le \alpha \le 1$, $0 \le \beta^* \le 1$, $0 \le \gamma \le 1-\alpha$ and m = period of seasonality (e.g. $m = 4$ for quarterly data).

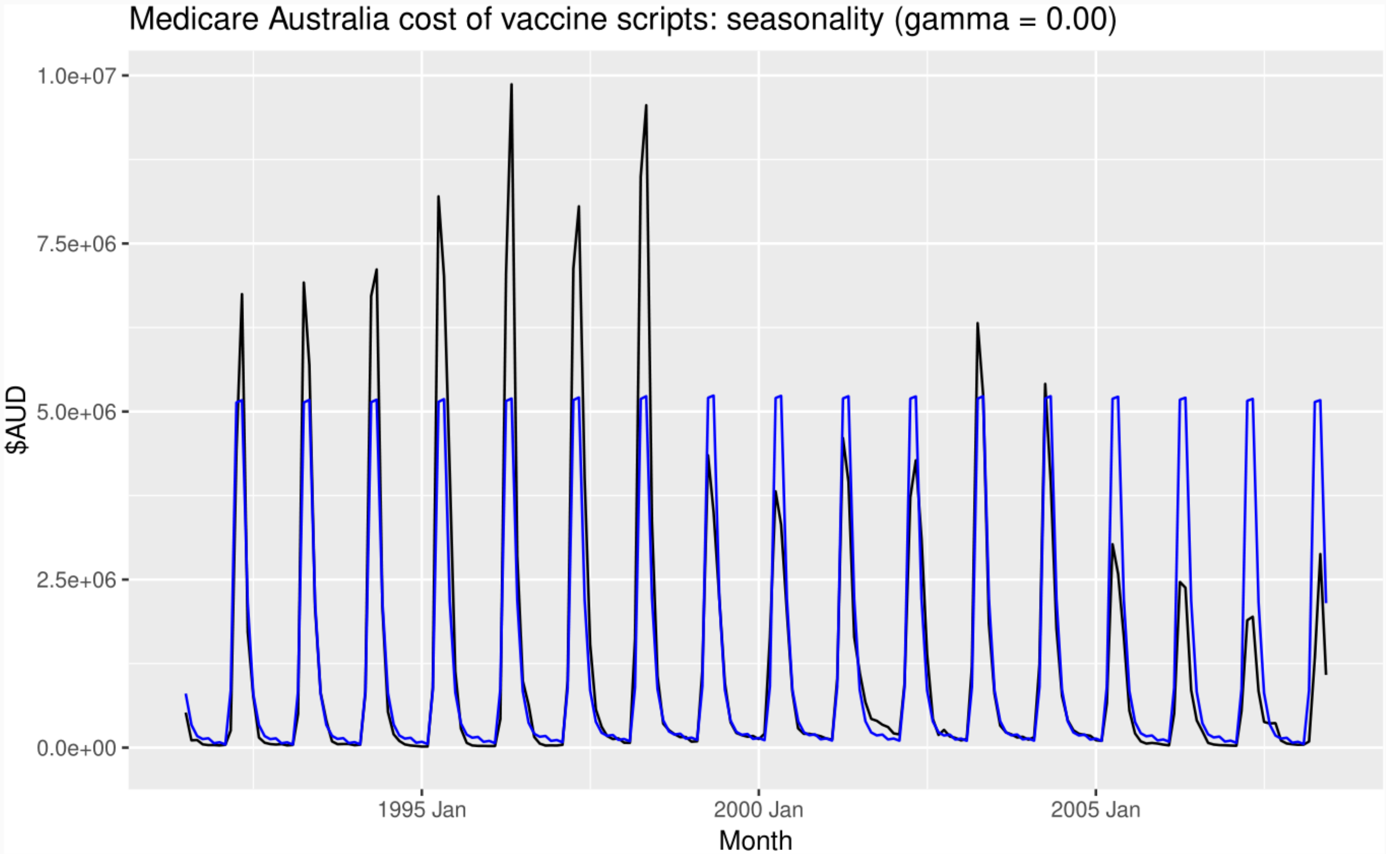
Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Holt-Winters additive method

- Seasonal component is usually expressed as $s_t = \gamma^*(y_t - \ell_t) + (1 - \gamma^*)s_{t-m}.$
- Substitute in for ℓ_t: $s_t = \gamma^*(1 - \alpha)(y_t - \ell_{t-1} - b_{t-1}) + [1 - \gamma^*(1 - \alpha)]s_{t-m}$
- We set $\gamma = \gamma^*(1 - \alpha)$.
- The usual parameter restriction is $0 \leq \gamma^* \leq 1$, which translates to $0 \leq \gamma \leq (1 - \alpha)$.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Exponential smoothing: seasonality



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

ETS(A,A,A)

Holt-Winters additive method with additive errors.

Forecast equation	$\hat{y}_{t+h\|t} = \ell_t + hb_t + s_{t+h-m(k+1)}$
Observation equation	$y_t = \ell_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t$
State equations	$\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t$
	$b_t = b_{t-1} + \beta\varepsilon_t$
	$s_t = s_{t-m} + \gamma\varepsilon_t$

- Forecast errors: $\varepsilon_t = y_t - \hat{y}_{t|t-1}$
- k is integer part of $(h-1)/m$.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Holt-Winters multiplicative method

Seasonal variations change in proportion to the level of the series.

Component form

$$
\begin{aligned}
\hat{y}_{t+h|t} &= (\ell_t + hb_t)s_{t+h-m(k+1)} \\
\ell_t &= \alpha \frac{y_t}{s_{t-m}} + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\
b_t &= \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1} \\
s_t &= \gamma \frac{y_t}{(\ell_{t-1} + b_{t-1})} + (1-\gamma)s_{t-m}
\end{aligned}
$$

- k is integer part of $(h-1)/m$.
- Additive method: s_t in absolute terms — within each year $\sum_i s_i \approx 0$.
- Multiplicative method: s_t in relative terms — within each year $\sum_i s_i \approx m$.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

ETS(M,A,M)

Holt-Winters multiplicative method with multiplicative errors.

$$
\begin{aligned}
\text{Forecast equation} \quad & \hat{y}_{t+h|t} = (\ell_t + hb_t)s_{t+h-m(k+1)} \\
\text{Observation equation} \quad & y_t = (\ell_{t-1} + b_{t-1})s_{t-m}(1 + \varepsilon_t) \\
\text{State equations} \quad & \ell_t = (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t) \\
& b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1})\varepsilon_t \\
& s_t = s_{t-m}(1 + \gamma\varepsilon_t)
\end{aligned}
$$

- Forecast errors: $\varepsilon_t = (y_t - \hat{y}_{t|t-1})/\hat{y}_{t|t-1}$
- k is integer part of $(h-1)/m$.

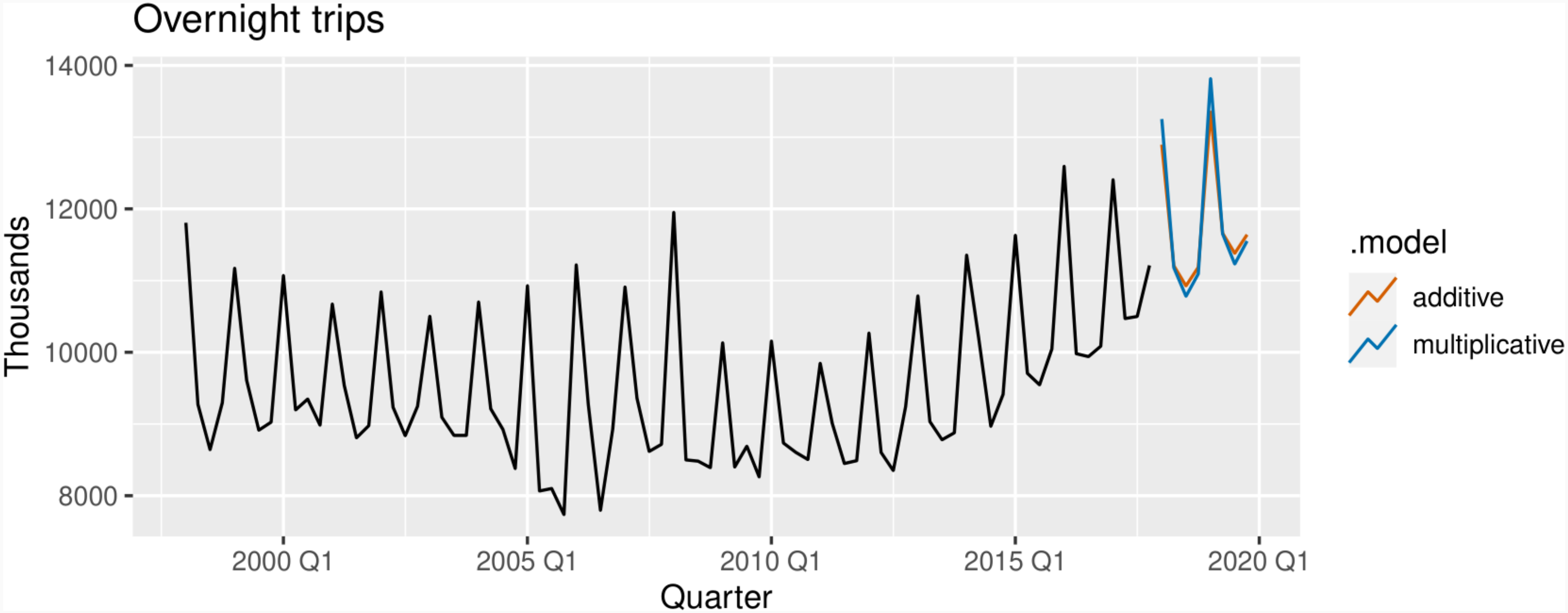
Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Australian holiday tourism

```
aus_holidays <- tourism %>%
  filter(Purpose == "Holiday") %>%
  summarise(Trips = sum(Trips))
fit <- aus_holidays %>%
  model(
    additive = ETS(Trips ~ error("A") + trend("A") + season("A")),
    multiplicative = ETS(Trips ~ error("M") + trend("A") + season("M"))
  )
fc <- fit %>% forecast()
```

Example: Australian holiday tourism

```
fc %>%
  autoplot(aus_holidays, level = NULL) +
  labs(y = "Thousands", title = "Overnight trips")
```

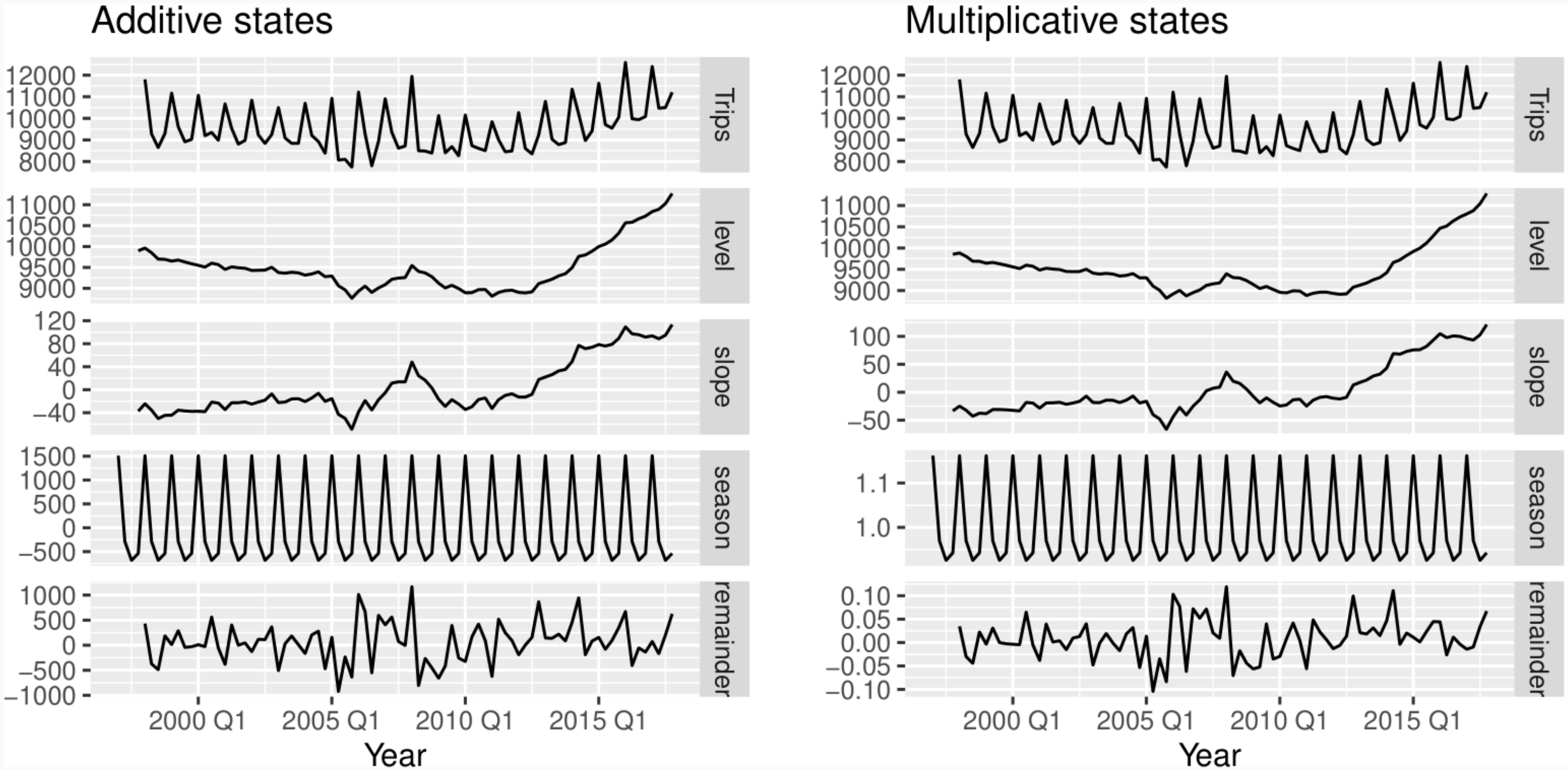


Estimated components

```
components(fit)
```

```
## # A dable: 168 x 7 [1Q]
## # Key:      .model [2]
## # :         Trips = lag(level, 1) + lag(slope, 1) + lag(season, 4) +
## #   remainder
##    .model    Quarter  Trips level slope season remainder
##    <chr>       <qtr>  <dbl> <dbl> <dbl>  <dbl>     <dbl>
##  1 additive 1997 Q1     NA    NA   NA    1512.       NA
##  2 additive 1997 Q2     NA    NA   NA    -290.       NA
##  3 additive 1997 Q3     NA    NA   NA    -684.       NA
##  4 additive 1997 Q4     NA  9899. -37.4  -538.       NA
##  5 additive 1998 Q1 11806. 9964. -24.5  1512.      433.
##  6 additive 1998 Q2  9276. 9851. -35.6  -290.     -374.
##  7 additive 1998 Q3  8642. 9700. -50.2  -684.     -489.
##  8 additive 1998 Q4  9300. 9694. -44.6  -538.      188.
```

Estimated components



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Holt-Winters damped method

Often the single most accurate forecasting method for seasonal data:

$$\begin{aligned}
\hat{y}_{t+h|t} &= [\ell_t + (\phi + \phi^2 + \cdots + \phi^h)b_t]s_{t+h-m(k+1)} \\
\ell_t &= \alpha(y_t/s_{t-m}) + (1-\alpha)(\ell_{t-1} + \phi b_{t-1}) \\
b_t &= \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1} \\
s_t &= \gamma \frac{y_t}{(\ell_{t-1} + \phi b_{t-1})} + (1-\gamma)s_{t-m}
\end{aligned}$$

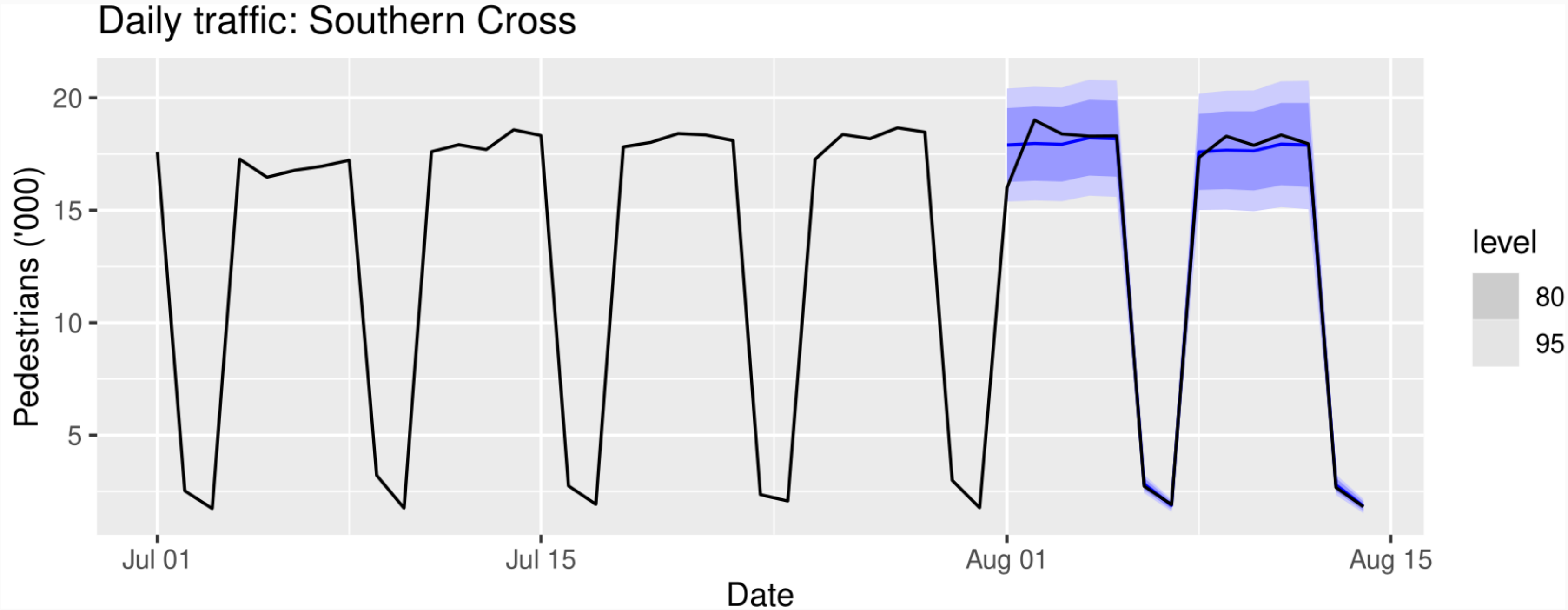
Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Holt-Winters with daily data

```
sth_cross_ped <- pedestrian %>%
  filter(
    Date >= "2016-07-01",
    Sensor == "Southern Cross Station"
  ) %>%
  index_by(Date) %>%
  summarise(Count = sum(Count) / 1000)
sth_cross_ped %>%
  filter(Date <= "2016-07-31") %>%
  model(
    hw = ETS(Count ~ error("M") + trend("Ad") + season("M"))
  ) %>%
  forecast(h = "2 weeks") %>%
  autoplot(sth_cross_ped %>% filter(Date <= "2016-08-14")) +
  labs(
    title = "Daily traffic: Southern Cross",
    y = "Pedestrians ('000)"
  )
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Holt-Winters with daily data



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Outline

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Exponential smoothing methods

	Trend Component	Seasonal Component N (None)	A (Additive)	M (Multiplicative)
N	(None)	(N,N)	(N,A)	(N,M)
A	(Additive)	(A,N)	(A,A)	(A,M)
A_d	(Additive damped)	(A_d,N)	(A_d,A)	(A_d,M)

(N,N): Simple exponential smoothing
(A,N): Holt's linear method
(A_d,N): Additive damped trend method
(A,A): Additive Holt-Winters' method
(A,M): Multiplicative Holt-Winters' method
(A_d,M): Damped multiplicative Holt-Winters' method

There are also multiplicative trend methods (not recommended).

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

ETS models

Additive Error

Trend Component		**Seasonal Component** N (None)	A (Additive)	M (Multiplicative)
N	(None)	A,N,N	A,N,A	A,N,M
A	(Additive)	A,A,N	A,A,A	A,A,M
A_d	(Additive damped)	A,A_d,N	A,A_d,A	A,A_d,M

Multiplicative Error

Trend Component		**Seasonal Component** N (None)	A (Additive)	M (Multiplicative)
N	(None)	M,N,N	M,N,A	M,N,M
A	(Additive)	M,A,N	M,A,A	M,A,M
A_d	(Additive damped)	M,A_d,N	M,A_d,A	M,A_d,M

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Additive error models

Trend	Seasonal N	Seasonal A	Seasonal M
N	$y_t = \ell_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha\varepsilon_t$	$y_t = \ell_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha\varepsilon_t$ $s_t = s_{t-m} + \gamma\varepsilon_t$	$y_t = \ell_{t-1}s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha\varepsilon_t/s_{t-m}$ $s_t = s_{t-m} + \gamma\varepsilon_t/\ell_{t-1}$
A	$y_t = \ell_{t-1} + b_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t$ $b_t = b_{t-1} + \beta\varepsilon_t$	$y_t = \ell_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t$ $b_t = b_{t-1} + \beta\varepsilon_t$ $s_t = s_{t-m} + \gamma\varepsilon_t$	$y_t = (\ell_{t-1} + b_{t-1})s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t/s_{t-m}$ $b_t = b_{t-1} + \beta\varepsilon_t/s_{t-m}$ $s_t = s_{t-m} + \gamma\varepsilon_t/(\ell_{t-1} + b_{t-1})$
A_d	$y_t = \ell_{t-1} + \phi b_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha\varepsilon_t$ $b_t = \phi b_{t-1} + \beta\varepsilon_t$	$y_t = \ell_{t-1} + \phi b_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha\varepsilon_t$ $b_t = \phi b_{t-1} + \beta\varepsilon_t$ $s_t = s_{t-m} + \gamma\varepsilon_t$	$y_t = (\ell_{t-1} + \phi b_{t-1})s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha\varepsilon_t/s_{t-m}$ $b_t = \phi b_{t-1} + \beta\varepsilon_t/s_{t-m}$ $s_t = s_{t-m} + \gamma\varepsilon_t/(\ell_{t-1} + \phi b_{t-1})$

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Multiplicative error models

Trend	Seasonal N	Seasonal A	Seasonal M
N	$y_t = \ell_{t-1}(1+\varepsilon_t)$ $\ell_t = \ell_{t-1}(1+\alpha\varepsilon_t)$	$y_t = (\ell_{t-1}+s_{t-m})(1+\varepsilon_t)$ $\ell_t = \ell_{t-1}+\alpha(\ell_{t-1}+s_{t-m})\varepsilon_t$ $s_t = s_{t-m}+\gamma(\ell_{t-1}+s_{t-m})\varepsilon_t$	$y_t = \ell_{t-1}s_{t-m}(1+\varepsilon_t)$ $\ell_t = \ell_{t-1}(1+\alpha\varepsilon_t)$ $s_t = s_{t-m}(1+\gamma\varepsilon_t)$
A	$y_t = (\ell_{t-1}+b_{t-1})(1+\varepsilon_t)$ $\ell_t = (\ell_{t-1}+b_{t-1})(1+\alpha\varepsilon_t)$ $b_t = b_{t-1}+\beta(\ell_{t-1}+b_{t-1})\varepsilon_t$	$y_t = (\ell_{t-1}+b_{t-1}+s_{t-m})(1+\varepsilon_t)$ $\ell_t = \ell_{t-1}+b_{t-1}+\alpha(\ell_{t-1}+b_{t-1}+s_{t-m})\varepsilon_t$ $b_t = b_{t-1}+\beta(\ell_{t-1}+b_{t-1}+s_{t-m})\varepsilon_t$ $s_t = s_{t-m}+\gamma(\ell_{t-1}+b_{t-1}+s_{t-m})\varepsilon_t$	$y_t = (\ell_{t-1}+b_{t-1})s_{t-m}(1+\varepsilon_t)$ $\ell_t = (\ell_{t-1}+b_{t-1})(1+\alpha\varepsilon_t)$ $b_t = b_{t-1}+\beta(\ell_{t-1}+b_{t-1})\varepsilon_t$ $s_t = s_{t-m}(1+\gamma\varepsilon_t)$
A_d	$y_t = (\ell_{t-1}+\phi b_{t-1})(1+\varepsilon_t)$ $\ell_t = (\ell_{t-1}+\phi b_{t-1})(1+\alpha\varepsilon_t)$ $b_t = \phi b_{t-1}+\beta(\ell_{t-1}+\phi b_{t-1})\varepsilon_t$	$y_t = (\ell_{t-1}+\phi b_{t-1}+s_{t-m})(1+\varepsilon_t)$ $\ell_t = \ell_{t-1}+\phi b_{t-1}+\alpha(\ell_{t-1}+\phi b_{t-1}+s_{t-m})\varepsilon_t$ $b_t = \phi b_{t-1}+\beta(\ell_{t-1}+\phi b_{t-1}+s_{t-m})\varepsilon_t$ $s_t = s_{t-m}+\gamma(\ell_{t-1}+\phi b_{t-1}+s_{t-m})\varepsilon_t$	$y_t = (\ell_{t-1}+\phi b_{t-1})s_{t-m}(1+\varepsilon_t)$ $\ell_t = (\ell_{t-1}+\phi b_{t-1})(1+\alpha\varepsilon_t)$ $b_t = \phi b_{t-1}+\beta(\ell_{t-1}+\phi b_{t-1})\varepsilon_t$ $s_t = s_{t-m}(1+\gamma\varepsilon_t)$

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Estimating ETS models

- Smoothing parameters α, β, γ and ϕ, and the initial states ℓ_0, b_0, $s_0, s_{-1}, \dots, s_{-m+1}$ are estimated by maximising the "likelihood" = the probability of the data arising from the specified model.
- For models with additive errors equivalent to minimising SSE.
- For models with multiplicative errors, **not** equivalent to minimising SSE.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Innovations state space models

Let $\boldsymbol{x}_t = (\ell_t, b_t, s_t, s_{t-1}, \dots, s_{t-m+1})$ and $\varepsilon_t \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2)$.

$$
\begin{aligned}
y_t &= \underbrace{h(\boldsymbol{x}_{t-1})}_{\mu_t} + \underbrace{k(\boldsymbol{x}_{t-1})\varepsilon_t}_{e_t} \\
\boldsymbol{x}_t &= f(\boldsymbol{x}_{t-1}) + g(\boldsymbol{x}_{t-1})\varepsilon_t
\end{aligned}
$$

Additive errors

$k(x) = 1. \qquad y_t = \mu_t + \varepsilon_t.$

Multiplicative errors

$k(\boldsymbol{x}_{t-1}) = \mu_t. \qquad y_t = \mu_t(1 + \varepsilon_t).$

$\varepsilon_t = (y_t - \mu_t)/\mu_t$ is relative error.

Innovations state space models

Estimation

$$
\begin{aligned}
L^*(\boldsymbol{\theta}, \mathbf{x}_0) &= T \log\left(\sum_{t=1}^{T} \varepsilon_t^2 \right) + 2 \sum_{t=1}^{T} \log |k(\mathbf{x}_{t-1})| \\
&= -2 \log(\text{Likelihood}) + \text{constant}
\end{aligned}
$$

- Estimate parameters $\boldsymbol{\theta} = (\alpha, \beta, \gamma, \phi)$ and initial states $\mathbf{x}_0 = (\ell_0, b_0, s_0, s_{-1}, \dots, s_{-m+1})$ by minimizing L^*.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Parameter restrictions

Usual region

- Traditional restrictions in the methods $0 < \alpha, \beta^*, \gamma^*, \phi < 1$ (equations interpreted as weighted averages).
- In models we set $\beta = \alpha\beta^*$ and $\gamma = (1 - \alpha)\gamma^*$.
- Therefore $0 < \alpha < 1, \quad 0 < \beta < \alpha \quad$ and $0 < \gamma < 1 - \alpha$.
- $0.8 < \phi < 0.98$ — to prevent numerical difficulties.

Admissible region

- To prevent observations in the distant past having a continuing effect on current forecasts.
- Usually (but not always) less restrictive than *traditional* region.
- For example for ETS(A,N,N):
  traditional $0 < \alpha < 1$ while *admissible* $0 < \alpha < 2$.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Model selection

Akaike's Information Criterion

$$\text{AIC} = -2\log(\text{L}) + 2k$$

where L is the likelihood and k is the number of parameters initial states estimated in the model.

Corrected AIC

$$\text{AIC}_\text{c} = \text{AIC} + \frac{2k(k+1)}{T-k-1}$$

which is the AIC corrected (for small sample bias).

Bayesian Information Criterion

$$\text{BIC} = \text{AIC} + k[\log(T) - 2].$$

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

AIC and cross-validation

Minimizing the AIC assuming Gaussian residuals is asymptotically equivalent to minimizing one-step time series cross validation MSE.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Automatic forecasting

From Hyndman et al. (IJF, 2002):

- Apply each model that is appropriate to the data. Optimize parameters and initial values using MLE (or some other criterion).
- Select best method using AICc:
- Produce forecasts using best method.
- Obtain forecast intervals using underlying state space model.

Method performed very well in M3 competition.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: National populations

```
fit <- global_economy %>%
  mutate(Pop = Population / 1e6) %>%
  model(ets = ETS(Pop))
fit
```

```
## # A mable: 263 x 2
## # Key:     Country [263]
##    Country                      ets
##    <fct>                    <model>
##  1 Afghanistan         <ETS(A,A,N)>
##  2 Albania             <ETS(M,A,N)>
##  3 Algeria             <ETS(M,A,N)>
##  4 American Samoa      <ETS(M,A,N)>
##  5 Andorra             <ETS(M,A,N)>
##  6 Angola              <ETS(M,A,N)>
##  7 Antigua and Barbuda <ETS(M,A,N)>
##  8 Arab World          <ETS(M,A,N)>
##  9 Argentina           <ETS(A,A,N)>
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: National populations

```r
fit %>%
  forecast(h = 5)
```

```
## # A fable: 1,315 x 5 [1Y]
## # Key:     Country, .model [263]
##    Country     .model  Year             Pop .mean
##    <fct>       <chr>  <dbl>          <dist> <dbl>
##  1 Afghanistan ets     2018    N(36, 0.012) 36.4
##  2 Afghanistan ets     2019    N(37, 0.059) 37.3
##  3 Afghanistan ets     2020     N(38, 0.16) 38.2
##  4 Afghanistan ets     2021     N(39, 0.35) 39.0
##  5 Afghanistan ets     2022     N(40, 0.64) 39.9
##  6 Albania     ets     2018 N(2.9, 0.00012)  2.87
##  7 Albania     ets     2019   N(2.9, 6e-04)  2.87
##  8 Albania     ets     2020  N(2.9, 0.0017)  2.87
##  9 Albania     ets     2021  N(2.9, 0.0036)  2.86
## 10 Albania     ets     2022  N(2.9, 0.0066)  2.86
## # ... with 1,305 more rows
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Australian holiday tourism

```
holidays <- tourism %>%
  filter(Purpose == "Holiday")
fit <- holidays %>% model(ets = ETS(Trips))
fit
```

```
## # A mable: 76 x 4
## # Key:     Region, State, Purpose [76]
##    Region                     State              Purpose          ets
##    <chr>                      <chr>              <chr>        <model>
##  1 Adelaide                   South Australia    Holiday <ETS(A,N,A)>
##  2 Adelaide Hills             South Australia    Holiday <ETS(A,A,N)>
##  3 Alice Springs              Northern Territory Holiday <ETS(M,N,A)>
##  4 Australia's Coral Coast    Western Australia  Holiday <ETS(M,N,A)>
##  5 Australia's Golden Outback Western Australia  Holiday <ETS(M,N,M)>
##  6 Australia's North West     Western Australia  Holiday <ETS(A,N,A)>
##  7 Australia's South West     Western Australia  Holiday <ETS(M,N,M)>
##  8 Ballarat                   Victoria           Holiday <ETS(M,N,A)>
##  9 Barkly                     Northern Territory Holiday <ETS(A,N,A)>
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Australian holiday tourism

```r
fit %>%
  filter(Region == "Snowy Mountains") %>%
  report()
```

```
## Series: Trips
## Model: ETS(M,N,A)
##   Smoothing parameters:
##     alpha = 0.157
##     gamma = 1e-04
##
##   Initial states:
##  l[0] s[0] s[-1] s[-2] s[-3]
##   142  -61   131 -42.2 -27.7
##
##   sigma^2:  0.0388
##
##  AIC AICc  BIC
##  852  854  869
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Australian holiday tourism

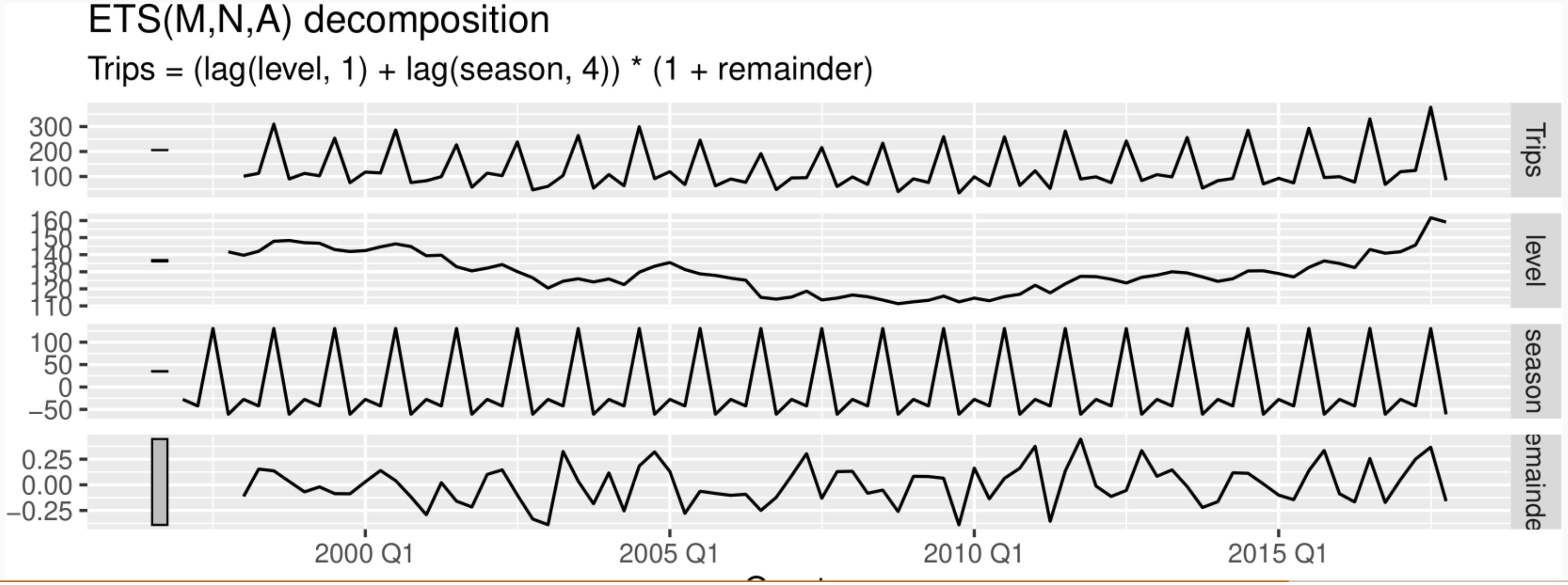
```r
fit %>%
  filter(Region == "Snowy Mountains") %>%
  components(fit)
```

```
## # A dable: 84 x 9 [1Q]
## # Key:     Region, State, Purpose, .model [1]
## # :        Trips = (lag(level, 1) + lag(season, 4)) * (1 + remainder)
##    Region          State Purpose .model Quarter Trips level season remainder
##    <chr>           <chr> <chr>   <chr>    <qtr> <dbl> <dbl>  <dbl>     <dbl>
##  1 Snowy Mountai~ New ~ Holiday ets    1997 Q1  NA      NA   -27.7   NA
##  2 Snowy Mountai~ New ~ Holiday ets    1997 Q2  NA      NA   -42.2   NA
##  3 Snowy Mountai~ New ~ Holiday ets    1997 Q3  NA      NA   131.    NA
##  4 Snowy Mountai~ New ~ Holiday ets    1997 Q4  NA     142.  -61.0   NA
##  5 Snowy Mountai~ New ~ Holiday ets    1998 Q1 101.    140.  -27.7   -0.113
##  6 Snowy Mountai~ New ~ Holiday ets    1998 Q2 112.    142.  -42.2    0.154
##  7 Snowy Mountai~ New ~ Holiday ets    1998 Q3 310.    148.  131.     0.137
##  8 Snowy Mountai~ New ~ Holiday ets    1998 Q4  89.8   148.  -61.0    0.0335
##  9 Snowy Mountai~ New ~ Holiday ets    1999 Q1 112.    147.  -27.7   -0.0687
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Australian holiday tourism

```
fit %>%
  filter(Region == "Snowy Mountains") %>%
  components(fit) %>%
  autoplot()
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

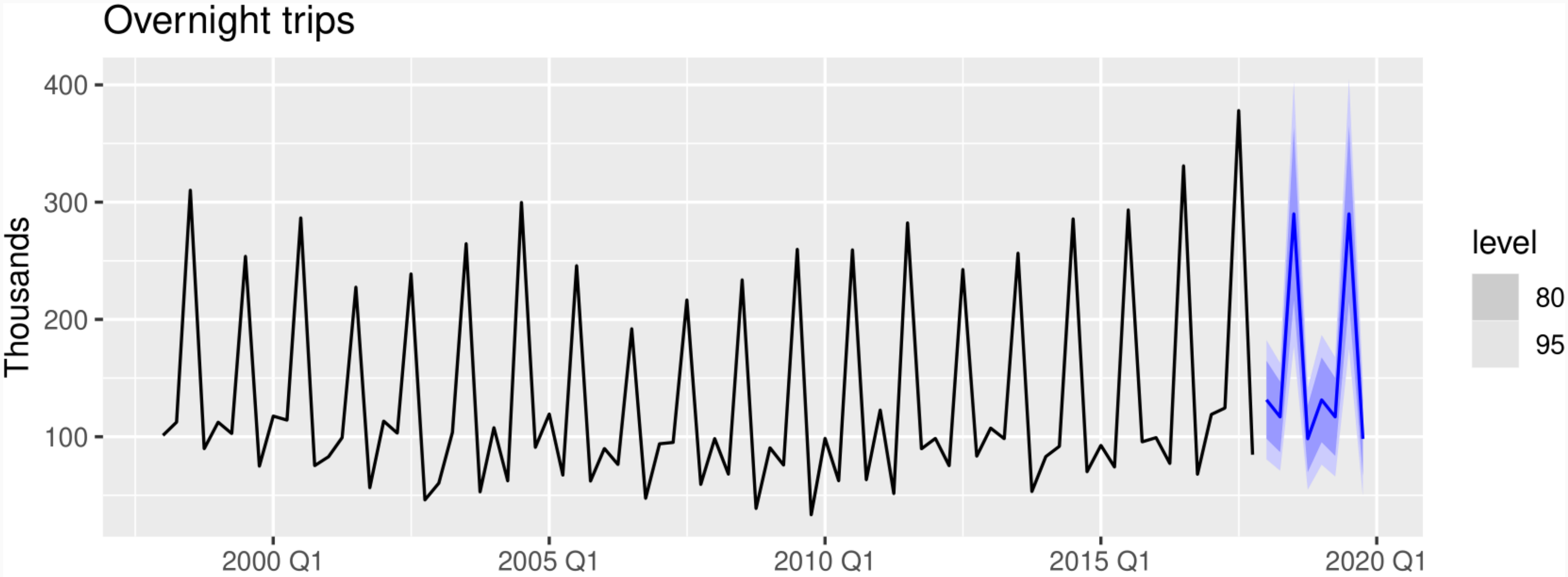
Example: Australian holiday tourism

```
fit %>% forecast()
```

```
## # A fable: 608 x 7 [1Q]
## # Key:     Region, State, Purpose, .model [76]
##    Region          State           Purpose .model Quarter        Trips .mean
##    <chr>           <chr>           <chr>   <chr>    <qtr>       <dist> <dbl>
##  1 Adelaide        South Australia Holiday ets    2018 Q1 N(210, 457) 210.
##  2 Adelaide        South Australia Holiday ets    2018 Q2 N(173, 473) 173.
##  3 Adelaide        South Australia Holiday ets    2018 Q3 N(169, 489) 169.
##  4 Adelaide        South Australia Holiday ets    2018 Q4 N(186, 505) 186.
##  5 Adelaide        South Australia Holiday ets    2019 Q1 N(210, 521) 210.
##  6 Adelaide        South Australia Holiday ets    2019 Q2 N(173, 537) 173.
##  7 Adelaide        South Australia Holiday ets    2019 Q3 N(169, 553) 169.
##  8 Adelaide        South Australia Holiday ets    2019 Q4 N(186, 569) 186.
##  9 Adelaide Hills South Australia Holiday ets    2018 Q1   N(19, 36)  19.4
## 10 Adelaide Hills South Australia Holiday ets    2018 Q2   N(20, 36)  19.6
## # ... with 598 more rows
```

Example: Australian holiday tourism

```
fit %>% forecast() %>%
  filter(Region == "Snowy Mountains") %>%
  autoplot(holidays) +
  labs(y = "Thousands", title = "Overnight trips")
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Residuals

Response residuals

$$\hat{e}_t = y_t - \hat{y}_{t|t-1}$$

Innovation residuals

Additive error model:

$$\hat{\varepsilon}_t = y_t - \hat{y}_{t|t-1}$$

Multiplicative error model:

$$\hat{\varepsilon}_t = \frac{y_t - \hat{y}_{t|t-1}}{\hat{y}_{t|t-1}}$$

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Australian holiday tourism

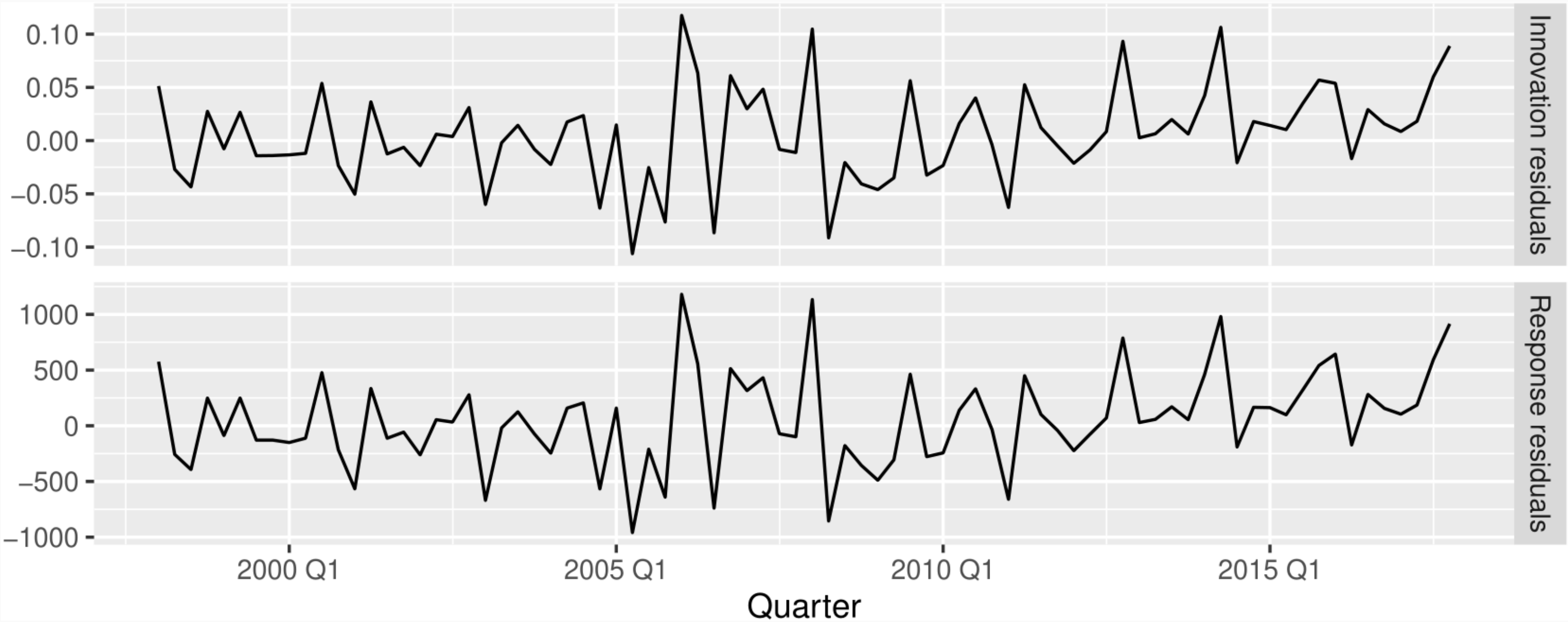
```r
aus_holidays <- tourism %>%
  filter(Purpose == "Holiday") %>%
  summarise(Trips = sum(Trips))
fit <- aus_holidays %>%
  model(ets = ETS(Trips)) %>%
  report()
```

```
## Series: Trips
## Model: ETS(M,N,M)
##   Smoothing parameters:
##     alpha = 0.358
##     gamma = 0.000969
##
##   Initial states:
##  l[0]  s[0] s[-1] s[-2] s[-3]
##  9667 0.943 0.927 0.968  1.16
##
##   sigma^2:  0.0022
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Australian holiday tourism

```
residuals(fit)
residuals(fit, type = "response")
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Australian holiday tourism

```
fit %>%
  augment()
```

Innovation residuals (`.innov`) are given by $\hat{\varepsilon}_t$ while regular residuals (`.resid`) are $y_t - \hat{y}_{t-1}$. They are different when the model has multiplicative errors.

```
## # A tsibble: 80 x 6 [1Q]
## # Key:       .model [1]
##    .model Quarter  Trips .fitted .resid   .innov
##    <chr>    <qtr>  <dbl>   <dbl>  <dbl>    <dbl>
##  1 ets    1998 Q1 11806.  11230.  576.   0.0513
##  2 ets    1998 Q2  9276.   9532. -257.  -0.0269
##  3 ets    1998 Q3  8642.   9036. -393.  -0.0435
##  4 ets    1998 Q4  9300.   9050.  249.   0.0275
##  5 ets    1999 Q1 11172.  11260.  -88.0 -0.00781
##  6 ets    1999 Q2  9608.   9358.  249.   0.0266
##  7 ets    1999 Q3  8914.   9042. -129.  -0.0142
##  8 ets    1999 Q4  9026.   9154. -129.  -0.0140
##  9 ets    2000 Q1 11071.  11221. -150.  -0.0134
## 10 ets    2000 Q2  9196.   9308. -111.  -0.0120
## # ... with 70 more rows
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Some unstable models

- Some of the combinations of (Error, Trend, Seasonal) can lead to numerical difficulties; see equations with division by a state.
- These are: ETS(A,N,M), ETS(A,A,M), ETS(A,A_d,M).
- Models with multiplicative errors are useful for strictly positive data, but are not numerically stable with data containing zeros or negative values. In that case only the six fully additive models will be applied.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Exponential smoothing models

Additive Error

Trend Component		Seasonal Component: N (None)	A (Additive)	M (Multiplicative)
N	(None)	A,N,N	A,N,A	~~A,N,M~~
A	(Additive)	A,A,N	A,A,A	~~A,A,M~~
A_d	(Additive damped)	A,A_d,N	A,A_d,A	~~A,A_d,M~~

Multiplicative Error

Trend Component		Seasonal Component: N (None)	A (Additive)	M (Multiplicative)
N	(None)	M,N,N	M,N,A	M,N,M
A	(Additive)	M,A,N	M,A,A	M,A,M
A_d	(Additive damped)	M,A_d,N	M,A_d,A	M,A_d,M

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Outline

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Forecasting with ETS models

Traditional point forecasts: iterate the equations for $t = T+1, T+2, \dots, T+h$ and set all $\varepsilon_t = 0$ for $t > T$.

- Not the same as $\mathrm{E}(y_{t+h}|\mathbf{x}_t)$ unless seasonality is additive.
- `fable` uses $\mathrm{E}(y_{t+h}|\mathbf{x}_t)$.
- Point forecasts for ETS(A,$*$,$*$) are identical to ETS(M,$*$,$*$) if the parameters are the same.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: ETS(A,A,N)

$$\begin{aligned}
y_{T+1} &= \ell_T + b_T + \varepsilon_{T+1} \\
\hat{y}_{T+1|T} &= \ell_T + b_T \\
y_{T+2} &= \ell_{T+1} + b_{T+1} + \varepsilon_{T+2} \\
&= (\ell_T + b_T + \alpha\varepsilon_{T+1}) + (b_T + \beta\varepsilon_{T+1}) + \varepsilon_{T+2} \\
\hat{y}_{T+2|T} &= \ell_T + 2b_T
\end{aligned}$$

etc.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: ETS(M,A,N)

$$
\begin{aligned}
y_{T+1} &= (\ell_T + b_T)(1 + \varepsilon_{T+1}) \\
\hat{y}_{T+1|T} &= \ell_T + b_T. \\
y_{T+2} &= (\ell_{T+1} + b_{T+1})(1 + \varepsilon_{T+2}) \\
&= \left\{(\ell_T + b_T)(1 + \alpha\varepsilon_{T+1}) + [b_T + \beta(\ell_T + b_T)\varepsilon_{T+1}]\right\}(1 + \varepsilon_{T+2}) \\
\hat{y}_{T+2|T} &= \ell_T + 2b_T
\end{aligned}
$$

etc.

Forecasting with ETS models

Prediction intervals: can only be generated using the models.

- The prediction intervals will differ between models with additive and multiplicative errors.
- Exact formulae for some models.
- More general to simulate future sample paths, conditional on the last estimate of the states, and to obtain prediction intervals from the percentiles of these simulated future paths.

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Prediction intervals

PI for most ETS models: $\hat{y}_{T+h|T} \pm c\sigma_h$, where c depends on coverage probability and σ_h is forecast standard deviation.

(A,N,N) $\quad \sigma_h = \sigma^2\Big[1+\alpha^2(h-1)\Big]$

(A,A,N) $\quad \sigma_h = \sigma^2\Big[1+(h-1)\{\alpha^2+\alpha\beta h+\tfrac{1}{6}\beta^2 h(2h-1)\}\Big]$

(A,A$_d$,N) $\quad \sigma_h = \sigma^2\Big[1+\alpha^2(h-1)+\frac{\beta\phi h}{(1-\phi)^2}\{2\alpha(1-\phi)+\beta\phi\}-\frac{\beta\phi(1-\phi^h)}{(1-\phi)^2(1-\phi^2)}\{2\alpha(1-\phi^2)+\beta\phi(1+2\phi-\phi^h)\}\Big]$

(A,N,A) $\quad \sigma_h = \sigma^2\Big[1+\alpha^2(h-1)+\gamma k(2\alpha+\gamma)\Big]$

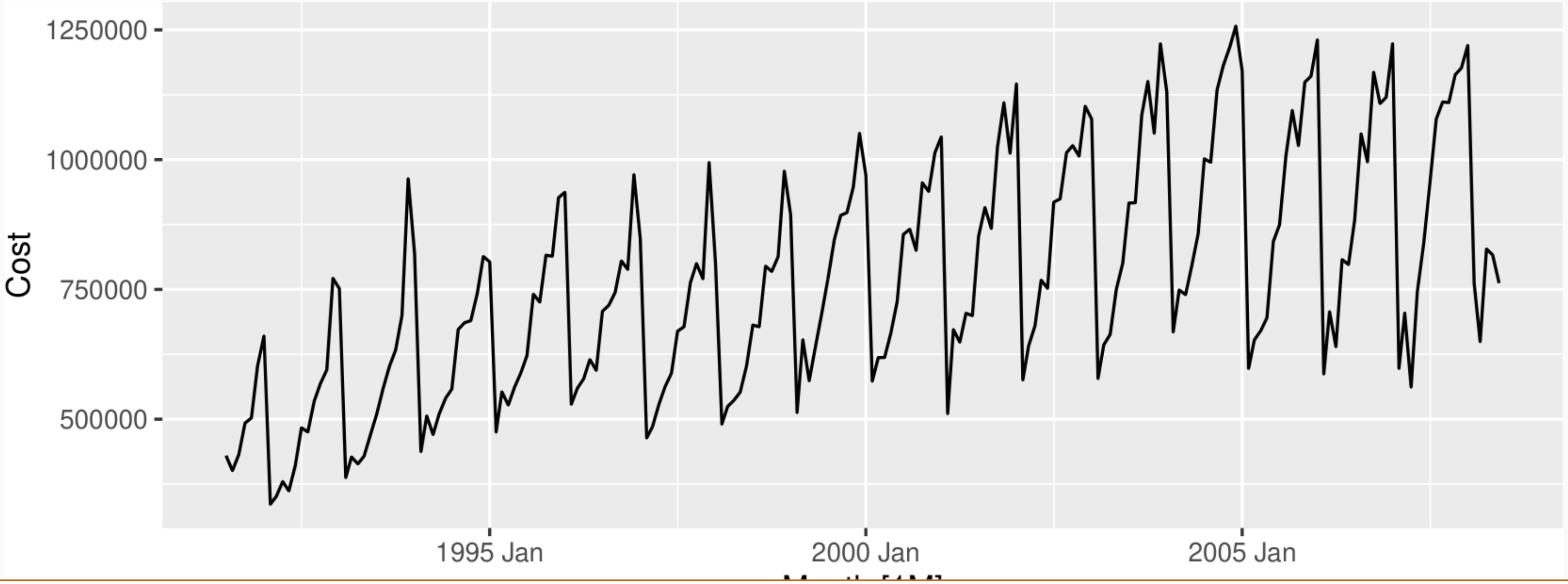
(A,A,A) $\quad \sigma_h = \sigma^2\Big[1+(h-1)\{\alpha^2+\alpha\beta h+\tfrac{1}{6}\beta^2 h(2h-1)\}+\gamma k\{2\alpha+\gamma+\beta m(k+1)\}\Big]$

(A,A$_d$,A) $\quad \sigma_h = \sigma^2\Big[1+\alpha^2(h-1)+\frac{\beta\phi h}{(1-\phi)^2}\{2\alpha(1-\phi)+\beta\phi\}-\frac{\beta\phi(1-\phi^h)}{(1-\phi)^2(1-\phi^2)}\{2\alpha(1-\phi^2)+\beta\phi(1+2\phi-\phi^h)\}$
$\quad + \gamma k(2\alpha+\gamma)+\frac{2\beta\gamma\phi}{(1-\phi)(1-\phi^m)}\{k(1-\phi^m)-\phi^m(1-\phi^{mk})\}\Big]$

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Corticosteroid drug sales

```r
h02 <- PBS %>%
  filter(ATC2 == "H02") %>%
  summarise(Cost = sum(Cost))
h02 %>% autoplot(Cost)
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Corticosteroid drug sales

```r
h02 %>%
  model(ETS(Cost)) %>%
  report()
```

```
## Series: Cost
## Model: ETS(M,Ad,M)
##   Smoothing parameters:
##     alpha = 0.307
##     beta  = 0.000101
##     gamma = 0.000101
##     phi   = 0.978
##
##   Initial states:
##    l[0] b[0]  s[0] s[-1] s[-2] s[-3] s[-4] s[-5] s[-6] s[-7] s[-8] s[-9]
##  417269 8206 0.872 0.826 0.756 0.773 0.687  1.28  1.32  1.18  1.16   1.1
##  s[-10] s[-11]
##    1.05  0.981
##
##   sigma^2:  0.0046
##
##  AIC AICc  BIC
## 5515 5519 5575
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Corticosteroid drug sales

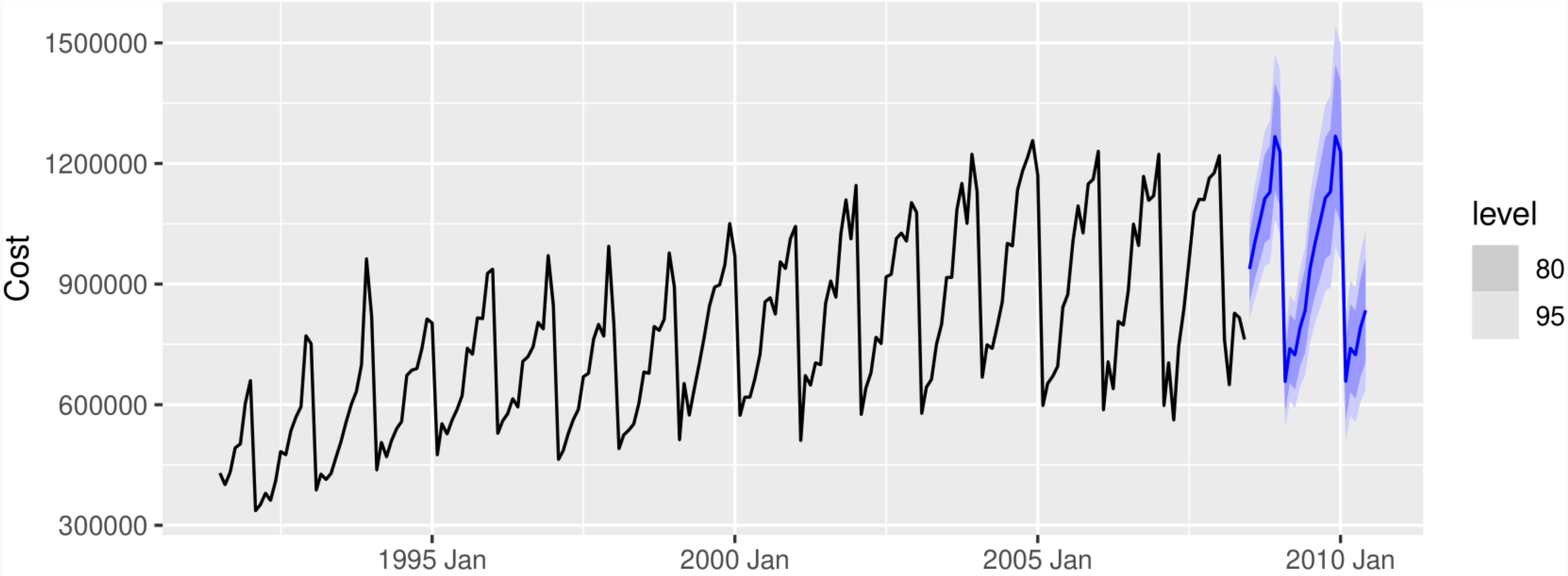
```r
h02 %>%
  model(ETS(Cost ~ error("A") + trend("A") + season("A"))) %>%
  report()
```

```
## Series: Cost
## Model: ETS(A,A,A)
##   Smoothing parameters:
##     alpha = 0.17
##     beta  = 0.00631
##     gamma = 0.455
##
##   Initial states:
##     l[0] b[0]   s[0]   s[-1]   s[-2]   s[-3]   s[-4]  s[-5]  s[-6]  s[-7]
##   409706 9097 -99075 -136602 -191496 -174531 -241437 210644 244644 145368
##   s[-8] s[-9] s[-10] s[-11]
##   130570 84458  39132 -11674
##
##   sigma^2:  3.5e+09
##
##   AIC AICc  BIC
## 5585 5589 5642
```


Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Corticosteroid drug sales

```
h02 %>%
  model(ETS(Cost)) %>%
  forecast() %>%
  autoplot(h02)
```



Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides

Example: Corticosteroid drug sales

```
h02 %>%
  model(
    auto = ETS(Cost),
    AAA = ETS(Cost ~ error("A") + trend("A") + season("A"))
  ) %>%
  accuracy()
```

Model	MAE	RMSE	MAPE	MASE	RMSSE
auto	38649	51102	4.99	0.638	0.689
AAA	43378	56784	6.05	0.716	0.766

Source: Rob J Hyndman, https://github.com/robjhyndman/ETC3550Slides