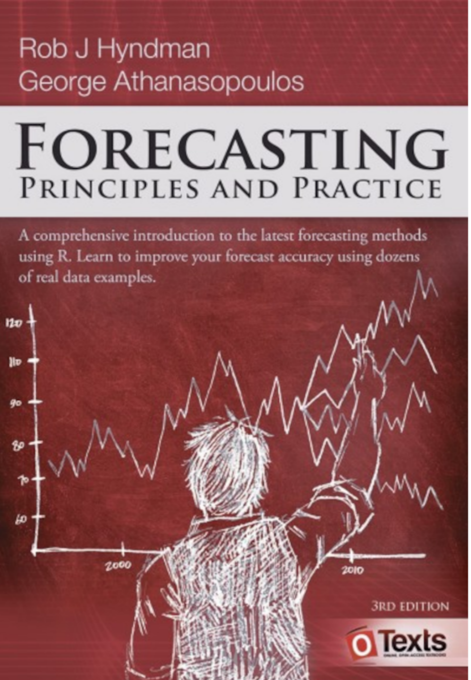


9. ARIMA models

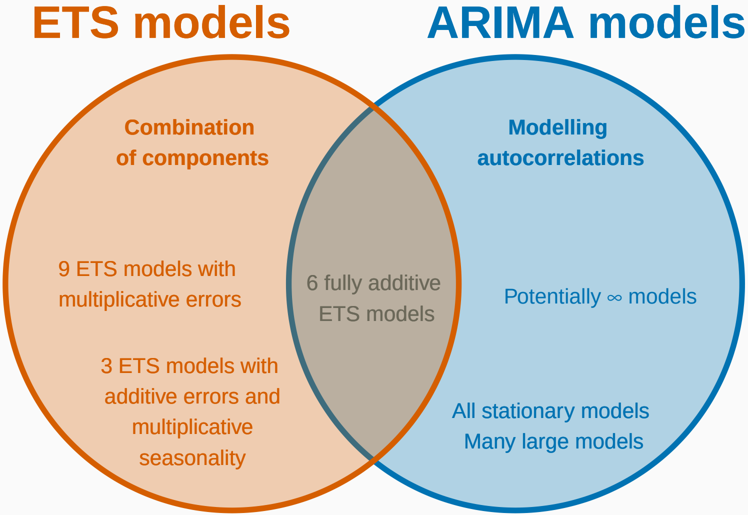
9.10 ARIMA vs ETS

OTexts.org/fpp3/

ARIMA vs ETS

- Myth that ARIMA models are more general than exponential smoothing.
- Linear exponential smoothing models all special cases of ARIMA models.
- Non-linear exponential smoothing models have no equivalent ARIMA counterparts.
- Many ARIMA models have no exponential smoothing counterparts.
- ETS models all non-stationary. Models with seasonality or non-damped trend (or both) have two unit roots; all other models have one unit root.

ARIMA vs ETS


ETS models
ARIMA models
Combination of components
Modelling autocorrelations
9 ETS models with multiplicative errors
6 fully additive ETS models
Potentially ∞ models
3 ETS models with additive errors and multiplicative seasonality
All stationary models
Many large models

Equivalences

ETS model	ARIMA model	Parameters
ETS(A,N,N)	ARIMA(0,1,1)	$\theta_1 = \alpha - 1$
ETS(A,A,N)	ARIMA(0,2,2)	$\theta_1 = \alpha + \beta - 2$
		$\theta_2 = 1 - \alpha$
ETS(A,A_d,N)	ARIMA(1,1,2)	$\phi_1 = \phi$
		$\theta_1 = \alpha + \phi\beta - 1 - \phi$
		$\theta_2 = (1 - \alpha)\phi$
ETS(A,N,A)	ARIMA(0,0,m)(0,1,0)$_m$	
ETS(A,A,A)	ARIMA(0,1,$m + 1$)(0,1,0)$_m$	
ETS(A,A_d,A)	ARIMA(1,0,$m + 1$)(0,1,0)$_m$	

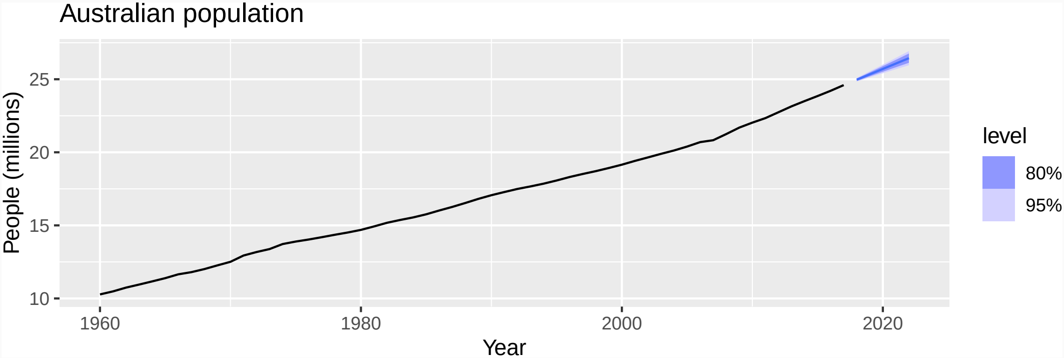
Example: Australian population

```r
aus_economy <- global_economy |>
  filter(Code == "AUS") |>
  mutate(Population = Population / 1e6)
aus_economy |>
  slice(-n()) |>
  stretch_tsibble(.init = 10) |>
  model(ets = ETS(Population), arima = ARIMA(Population)) |>
  forecast(h = 1) |>
  accuracy(aus_economy) |>
  select(.model, ME:RMSSE)
```

```
## # A tibble: 2 x 8
##   .model     ME   RMSE    MAE   MPE  MAPE  MASE RMSSE
##   <chr>   <dbl>  <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 arima  0.0420 0.194  0.0789 0.277 0.509 0.317 0.746
## 2 ets    0.0202 0.0774 0.0543 0.112 0.327 0.218 0.298
```

Example: Australian population

```r
aus_economy |>
  model(ETS(Population)) |>
  forecast(h = "5 years") |>
  autoplot(aus_economy) +
  labs(title = "Australian population", y = "People (millions)")
```



Example: Cement production

```r
cement <- aus_production |>
  select(Cement) |>
  filter_index("1988 Q1" ~ .)
train <- cement |> filter_index(. ~ "2007 Q4")
fit <- train |>
  model(
    arima = ARIMA(Cement),
    ets = ETS(Cement)
  )
```

Example: Cement production

```r
fit |>
  select(arima) |>
  report()
```

```
## Series: Cement
## Model: ARIMA(1,0,1)(2,1,1)[4] w/ drift
##
## Coefficients:
##          ar1     ma1   sar1    sar2    sma1  constant
##       0.8886  -0.237  0.081  -0.234  -0.898      5.39
## s.e.  0.0842   0.133  0.157   0.139   0.178      1.48
##
## sigma^2 estimated as 11456:  log likelihood=-464
## AIC=941   AICc=943   BIC=957
```

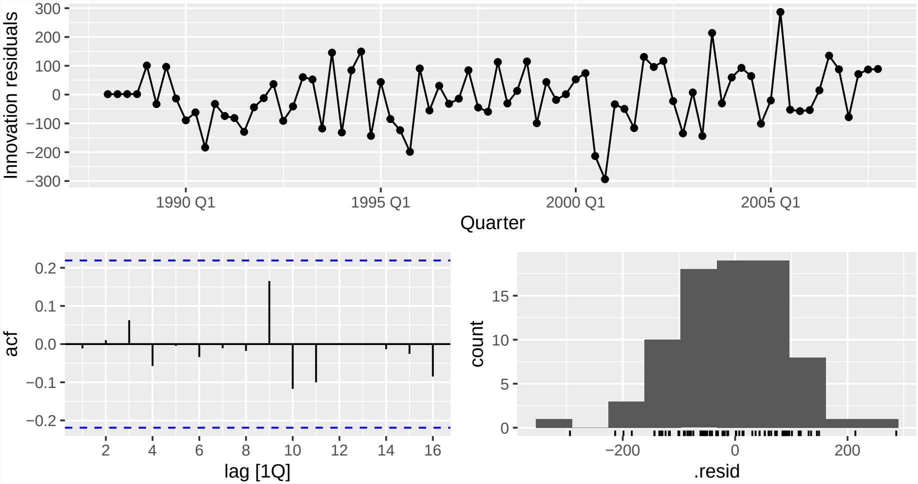
Example: Cement production

```r
fit |>
  select(ets) |>
  report()
```

```
## Series: Cement
## Model: ETS(M,N,M)
##   Smoothing parameters:
##     alpha = 0.753
##     gamma = 1e-04
##
##   Initial states:
##  l[0] s[0] s[-1] s[-2] s[-3]
##  1695 1.03  1.05  1.01 0.912
##
##   sigma^2:  0.0034
##
##  AIC AICc  BIC
## 1104 1106 1121
```

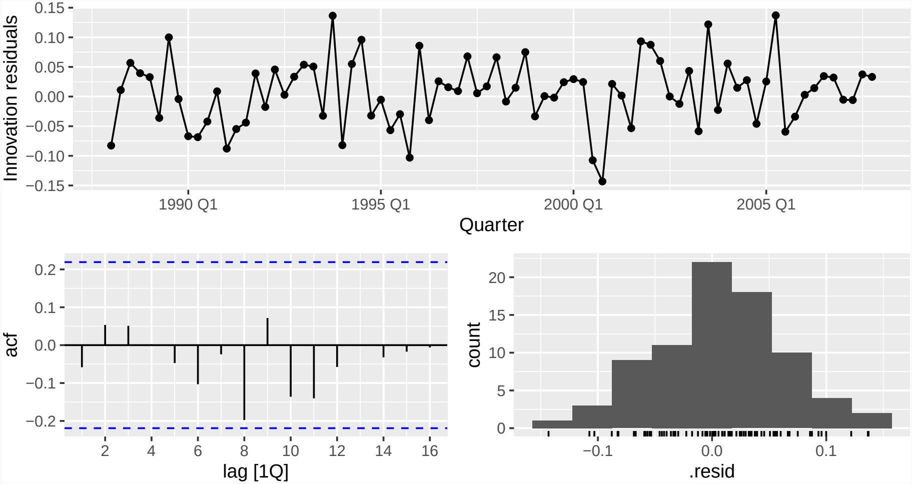
Example: Cement production

```r
gg_tsresiduals(fit |> select(arima), lag_max = 16)
```



Example: Cement production

```r
gg_tsresiduals(fit |> select(ets), lag_max = 16)
```



Example: Cement production

```r
fit |>
  select(arima) |>
  augment() |>
  features(.innov, ljung_box, lag = 16, dof = 6)
```

```
## # A tibble: 1 x 3
##   .model lb_stat lb_pvalue
##   <chr>    <dbl>     <dbl>
## 1 arima     6.37     0.783
```

Example: Cement production

```r
fit |>
  select(ets) |>
  augment() |>
  features(.innov, ljung_box, lag = 16)
```

```
## # A tibble: 1 x 3
##   .model lb_stat lb_pvalue
##   <chr>    <dbl>     <dbl>
## 1 ets       10.0     0.865
```

Example: Cement production

```r
fit |>
  forecast(h = "2 years 6 months") |>
  accuracy(cement) |>
  select(-ME, -MPE, -ACF1)
```

```
## # A tibble: 2 x 7
##   .model .type  RMSE   MAE  MAPE  MASE RMSSE
##   <chr>  <chr> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 arima  Test   216.  186.  8.68  1.27  1.26
## 2 ets    Test   222.  191.  8.85  1.30  1.29
```

Example: Cement production

```r
fit |>
  select(arima) |>
  forecast(h = "3 years") |>
  autoplot(cement) +
  labs(title = "Cement production in Australia", y = "Tonnes ('000)")
```