MONASH University

MONASH BUSINESS SCHOOL

Tidy Forecasting in R

Rob J Hyndman

useR2018: 13 July 2018

forecast package

Pre 2003	Private functions used for consulting
July/August 2003	**ets** and **thetaf** added
August 2006	v1.0 available on CRAN
May 2007	**auto.arima** added
Feb/March 2011	**tslm**, **stlf**, **naive**, **snaive** added
August 2011	v3.0. Box Cox transformations added
December 2011	**tbats** added
April 2012	Package moved to github
February 2016	v7.0. Added ggplot2 graphics
February 2017	v8.0. Added **checkresiduals**, **tsCV** and **%>%**
April 2018	v8.3. Added **mstl**
June 2018	$\approx$ 100,000 package downloads per month

fable package

A replacement for the forecast package.

Why change?

- Integrating with tidyverse packages
- Designed for forecasting many related time series
- Consistency of interface using formulas
- Distribution forecasting rather than point + interval
- Flexible transformations
- Sub-daily data and multiple seasonal data handled more easily
- Simpler interface for forecast reconciliation

Formula model specification

All modelling functions use a formula similar to `lm()` with automated modelling if RHS not specified.

```
t(y) ~ {model specification}
```

LHS: Response

- Defines the response variable from the data
- Specification of transformations (which are automatically back-transformed)

RHS: Specials

- Model specific special functions
- Exogenous regressors (if supported by model)

Example: Australian eating-out expenditure


```
library(tsibble)
(cafe <- as_tsibble(fpp2::auscafe))
```

```
## # A tsibble: 426 x 2 [1MONTH]
##        index value
##        <mth> <dbl>
##  1 1982 Apr 0.342
##  2 1982 May 0.342
##  3 1982 Jun 0.329
##  4 1982 Jul 0.338
##  5 1982 Aug 0.332
##  6 1982 Sep 0.342
##  7 1982 Oct 0.358
##  8 1982 Nov 0.375
##  9 1982 Dec 0.433
## 10 1983 Jan 0.369
## # ... with 416 more rows
```

Example: Australian eating-out expenditure


```
library(fable)
cafe %>% ARIMA(log(value) ~ pdq(2,1,1) + PDQ(2,1,2))
```

```
## # A mable: 1 model [1MONTH]
##   data                  model
##   <list>                <model>
## 1 <tsibble [426 x 2]> ARIMA(2,1,1)(2,1,2)[12]
```

Example: Australian eating-out expenditure


```
cafe %>% ARIMA(log(value) ~ pdq(2,1,1) + PDQ(2,1,2)) %>%
  summary()
```

```
## Series: log(value)
## ARIMA(2,1,1)(2,1,2)[12]
##
## Coefficients:
##          ar1     ar2    ma1   sar1    sar2    sma1   sma2
##       -0.925  -0.318  0.588  0.724  -0.213  -1.437  0.557
## s.e.   0.182   0.060  0.189  0.174   0.074   0.170  0.149
##
## sigma^2 estimated as 0.000554:  log likelihood=959.49
## AIC=-1903   AICc=-1902.6   BIC=-1870.8
##
## Training set error measures:
##                       ME     RMSE      MAE       MPE   MAPE     MASE
## Training set -0.0010597 0.037131 0.026734 -0.050621 1.7758 0.25586
##                    ACF1
## Training set -0.018411
```

Example: Australian eating-out expenditure


```
cafe %>% ARIMA(log(value) ~ pdq(2,1,1) + PDQ(2,1,2)) %>%
  forecast()
```

```
## # A fable: 1 forecast [1MONTH]
##   data                 model                     forecast
##   <list>               <model>                   <fc>
## 1 <tsibble [426 x 2]> ARIMA(2,1,1)(2,1,2)[12] ~t(N) [h=24]
```

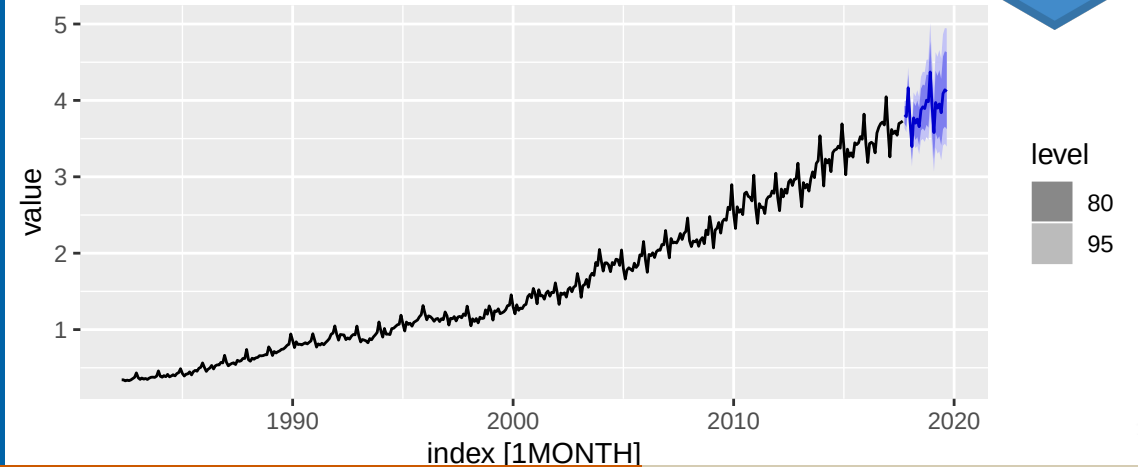
Example: Australian eating-out expenditure


```
cafe %>% ARIMA(log(value) ~ pdq(2,1,1) + PDQ(2,1,2)) %>%
  forecast() %>% summary(level=90)
```

```
## # A tsibble: 24 x 3 [1MONTH]
##       index  mean                 90%
##       <mth> <dbl>               <hilo>
##  1 2017 Oct  3.81 [3.6643, 3.9594]90
##  2 2017 Nov  3.79 [3.6150, 3.9669]90
##  3 2017 Dec  4.17 [3.9483, 4.3895]90
##  4 2018 Jan  3.73 [3.5033, 3.9560]90
##  5 2018 Feb  3.40 [3.1759, 3.6242]90
##  6 2018 Mar  3.77 [3.5075, 4.0488]90
##  7 2018 Apr  3.70 [3.4248, 3.9931]90
##  8 2018 May  3.76 [3.4573, 4.0692]90
##  9 2018 Jun  3.66 [3.3495, 3.9782]90
## 10 2018 Jul  3.88 [3.5359, 4.2353]90
## # ... with 14 more rows
```

Example: Australian eating-out expenditure

```
cafe %>% ARIMA(log(value) ~ pdq(2,1,1) + PDQ(2,1,2)) %>%
  forecast() %>% autoplot()
```



Example: Half-hourly electricity demand


```
elecdemand
```

```
## # A tsibble: 17,520 x 4 [30MINUTE]
##    index               Demand Temperature WorkDay
##    <dttm>               <dbl>       <dbl>   <dbl>
##  1 2014-01-01 00:00:00   3.91        18.2       0
##  2 2014-01-01 00:30:00   3.67        17.9       0
##  3 2014-01-01 01:00:00   3.50        17.6       0
##  4 2014-01-01 01:30:00   3.34        16.8       0
##  5 2014-01-01 02:00:00   3.20        16.3       0
##  6 2014-01-01 02:30:00   3.10        16.6       0
##  7 2014-01-01 03:00:00   3.04        16.6       0
##  8 2014-01-01 03:30:00   3.01        16.7       0
##  9 2014-01-01 04:00:00   3.02        16.2       0
## 10 2014-01-01 04:30:00   3.03        16.6       0
## # ... with 17,510 more rows
```

Example: Half-hourly electricity demand


```
fit2 <- ARIMA(elecdemand,
  Demand ~ Temperature + I(Temperature^2) + WorkDay)
summary(fit2)
```

```
## Series: Demand
## Regression with ARIMA(1,1,0)(2,0,2)[2] errors
##
## Coefficients:
##          ar1     sar1   sar2     sma1     sma2  Temperature  I(Temperature^2)
##       0.853  -0.181  0.523  -0.066  -0.792       -0.009                    0
## s.e.  0.005   0.015  0.012   0.012   0.011        0.002                    0
##       WorkDay
##         0.016
## s.e.    0.006
##
## sigma^2 estimated as 0.00846:  log likelihood=16949
## AIC=-33881   AICc=-33881   BIC=-33811
##
```

Example: Half-hourly electricity demand


```
fit2 <- ARIMA(elecdemand,
  Demand ~ Temperature + I(Temperature^2) + WorkDay)
summary(fit2)
```

```
## Series: Demand
## Regression with ARIMA(1,1,0)(2,0,2)[2] errors
##
## Coefficients:
##         ar1    sar1   sar2    sma1    sma2  Temperature  I(Temperature^2)
##       0.853  -0.181  0.523  -0.066  -0.792       -0.009                 0
## s.e.  0.005   0.015  0.012   0.012   0.011        0.002                 0
##       WorkDay
##         0.016
## s.e.    0.006
##
## sigma^2 estimated as 0.00846:  log likelihood=16949
## AIC=-33881   AICc=-33881   BIC=-33811
```

```
forecast(fit2, newdata=elecdemandfuture) %>% autoplot()
```

Example: Australian prison population


```
prison
```

```
## # A tsibble: 1,536 x 5 [1QUARTER]
## # Key:       state, gender, legal [32]
##    state gender legal    count     qtr
##    <fct> <fct>  <fct>    <dbl>   <qtr>
##  1 ACT   Female Remanded     2 2005 Q1
##  2 ACT   Female Remanded     4 2005 Q2
##  3 ACT   Female Remanded     1 2005 Q3
##  4 ACT   Female Remanded     4 2005 Q4
##  5 ACT   Female Remanded     4 2006 Q1
##  6 ACT   Female Remanded     6 2006 Q2
##  7 ACT   Female Remanded     9 2006 Q3
##  8 ACT   Female Remanded     6 2006 Q4
##  9 ACT   Female Remanded     4 2007 Q1
## 10 ACT   Female Remanded     4 2007 Q2
## # ... with 1,526 more rows
```

Example: Australian prison population


```
prison %>% ETS(count)
```

```
## # A mable: 32 models [1QUARTER]
## # Key:     state, gender, legal [32]
##    state gender legal     data                model
##    <fct> <fct>  <fct>     <list>              <model>
##  1 ACT   Female Remanded  <tsibble [48 x 2]> ETS(M,A,N)
##  2 ACT   Female Sentenced <tsibble [48 x 2]> ETS(A,A,N)
##  3 ACT   Male   Remanded  <tsibble [48 x 2]> ETS(M,N,N)
##  4 ACT   Male   Sentenced <tsibble [48 x 2]> ETS(A,N,N)
##  5 NSW   Female Remanded  <tsibble [48 x 2]> ETS(M,N,M)
##  6 NSW   Female Sentenced <tsibble [48 x 2]> ETS(M,N,M)
##  7 NSW   Male   Remanded  <tsibble [48 x 2]> ETS(M,A,A)
##  8 NSW   Male   Sentenced <tsibble [48 x 2]> ETS(M,A,A)
##  9 NT    Female Remanded  <tsibble [48 x 2]> ETS(M,N,N)
## 10 NT    Female Sentenced <tsibble [48 x 2]> ETS(M,A,A)
## # ... with 22 more rows
```

Example: Australian prison population


```
prison %>% ETS(count) %>% forecast()
```

```
## # A fable: 32 forecasts [1QUARTER]
## # Key:     state, gender, legal [32]
##    state gender legal     data                 model       forecast
##    <fct> <fct>  <fct>     <list>               <model>     <fc>
##  1 ACT   Female Remanded  <tsibble [48 x 2]> ETS(M,A,N) ~N [h=8]
##  2 ACT   Female Sentenced <tsibble [48 x 2]> ETS(A,A,N) ~N [h=8]
##  3 ACT   Male   Remanded  <tsibble [48 x 2]> ETS(M,N,N) ~N [h=8]
##  4 ACT   Male   Sentenced <tsibble [48 x 2]> ETS(A,N,N) ~N [h=8]
##  5 NSW   Female Remanded  <tsibble [48 x 2]> ETS(M,N,M) ~N [h=8]
##  6 NSW   Female Sentenced <tsibble [48 x 2]> ETS(M,N,M) ~N [h=8]
##  7 NSW   Male   Remanded  <tsibble [48 x 2]> ETS(M,A,A) ~N [h=8]
##  8 NSW   Male   Sentenced <tsibble [48 x 2]> ETS(M,A,A) ~N [h=8]
##  9 NT    Female Remanded  <tsibble [48 x 2]> ETS(M,N,N) ~N [h=8]
## 10 NT    Female Sentenced <tsibble [48 x 2]> ETS(M,A,A) ~N [h=8]
## # ... with 22 more rows
```

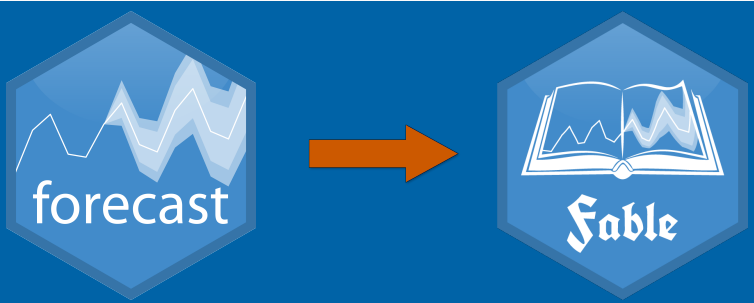
Example: Australian prison population


```
prison %>% ETS(count) %>% forecast()
```

```
## # A fable: 32 forecasts [1QUARTER]
## # Key:     state, gender, legal [32]
##    state gender legal     data                 model       forecast
##    <fct> <fct>  <fct>     <list>               <model>     <fc>
##  1 ACT   Female Remanded  <tsibble [48 x 2]> ETS(M,A,N) ~N [h=8]
##  2 ACT   Female Sentenced <tsibble [48 x 2]> ETS(A,A,N) ~N [h=8]
##  3 ACT   Male   Remanded  <tsibble [48 x 2]> ETS(M,N,N) ~N [h=8]
##  4 ACT   Male   Sentenced <tsibble [48 x 2]> ETS(A,N,N) ~N [h=8]
##  5 NSW   Female Remanded  <tsibble [48 x 2]> ETS(M,N,M) ~N [h=8]
##  6 NSW   Female Sentenced <tsibble [48 x 2]> ETS(M,N,M) ~N [h=8]
##  7 NSW   Male   Remanded  <tsibble [48 x 2]> ETS(M,A,A) ~N [h=8]
##  8 NSW   Male   Sentenced <tsibble [48 x 2]> ETS(M,A,A) ~N [h=8]
##  9 NT    Female Remanded  <tsibble [48 x 2]> ETS(M,N,N) ~N [h=8]
## 10 NT    Female Sentenced <tsibble [48 x 2]> ETS(M,A,A) ~N [h=8]
## # ... with 22 more rows
```

Aggregation and reconciliation not yet implemented.

Moving from forecast to fable



- All **forecast** `model()` functions will have equivalent **fable** `MODEL()` functions.
- All **fable** models produce `mable` class objects.
- `forecast()` works on `mable` objects to produce `fable` objects.
- **fable** will also replace the **hts** package

Extending fable

fable simplifies the model development process

Tools to easily create new fable models

- Easily create specials for model formulae
- Focus on model estimation and forecasts

Automatically supported fable functionality

- Transformations and back-transformations (with bias adjustments)
- Plotting tools
- Accuracy measures and evaluation
- Model combinations (hierarchies & ensembles)

More information

```
devtools::install_github("tidyverts/tsibble")
devtools::install_github("tidyverts/fable")
```

Di Cook

Earo Wang

Mitchell O'Hara-Wild

Follow our progress

- tidyverts.org
- robjhyndman.com/hyndsight