Flexible futures

for fable functionality

Mitchell O'Hara-Wild (@mitchoharawild)

11 July 2019

Slides @ slides.mitchelloharawild.com/user2019

Forecasting with the tidyverts

Forecasting with the tidyverts

tidyverts.org

Forecasting with the tidyverts

tidyverts.org

(see me later for some stickers!)

tsibble

- A modern temporal data structure
- Provides tools for time-related analysis
- Integrates seamlessly with the tidyverse

More information:

- tidyverts/tsibble
- tsibble site
- rstudio::conf 2019 & useR!2018

Tidy temporal data structure

From `ts` to `tsibble`

```r
co2
```

```
#>         Jan    Feb    Mar    Apr    May    Jun    Jul    Aug    Sep    Oct    Nov    Dec
#> 1959 315.42 316.31 316.50 317.56 318.13 318.00 316.39 314.65 313.68 313.18 314.66 315.43
#> 1960 316.27 316.81 317.42 318.87 319.87 319.43 318.01 315.74 314.00 313.68 314.84 316.03
#> 1961 316.73 317.54 318.38 319.31 320.42 319.61 318.42 316.63 314.83 315.16 315.94 316.85
#> 1962 317.78 318.40 319.53 320.42 320.85 320.45 319.45 317.25 316.11 315.27 316.53 317.53
#> 1963 318.58 318.92 319.70 321.22 322.08 321.31 319.58 317.61 316.05 315.83 316.91 318.20
#> 1964 319.41 320.07 320.74 321.40 322.06 321.73 320.27 318.54 316.54 316.71 317.53 318.55
#> 1965 319.27 320.28 320.73 321.97 322.00 321.71 321.05 318.71 317.66 317.14 318.70 319.25
#> 1966 320.46 321.43 322.23 323.54 323.91 323.59 322.24 320.20 318.48 317.94 319.63 320.87
#> 1967 322.17 322.34 322.88 324.25 324.83 323.93 322.38 320.76 319.10 319.24 320.56 321.80
#> 1968 322.40 322.99 323.73 324.86 325.40 325.20 323.98 321.95 320.18 320.09 321.16 322.74
#> 1969 323.83 324.26 325.47 326.50 327.21 326.54 325.72 323.50 322.22 321.62 322.69 323.95
#> 1970 324.89 325.82 326.77 327.97 327.91 327.50 326.18 324.53 322.93 322.90 323.85 324.96
#> 1971 326.01 326.51 327.01 327.62 328.76 328.40 327.20 325.27 323.20 323.40 324.63 325.85
#> 1972 326.60 327.47 327.58 329.56 329.90 328.92 327.88 326.16 324.68 325.04 326.34 327.39
#> 1973 328.37 329.40 330.14 331.33 332.31 331.90 330.70 329.15 327.35 327.02 327.99 328.48
#> 1974 329.18 330.55 331.32 332.48 332.92 332.08 331.01 329.23 327.27 327.21 328.29 329.41
#> 1975 330.23 331.25 331.87 333.14 333.80 333.43 331.73 329.90 328.40 328.17 329.32 330.59
#> 1976 331.58 332.39 333.33 334.41 334.71 334.17 332.89 330.77 329.14 328.78 330.14 331.52
#> 1977 332.75 333.24 334.53 335.90 336.57 336.10 334.76 332.59 331.42 330.98 332.24 333.68
#> 1978 334.80 335.22 336.47 337.59 337.84 337.72 336.37 334.51 332.60 332.38 333.75 334.78
#> 1979 336.05 336.59 337.79 338.71 339.30 339.12 337.56 335.92 333.75 333.70 335.12 336.56
#> 1980 337.84 338.19 339.91 340.60 341.29 341.00 339.39 337.43 335.72 335.84 336.93 338.04
#> 1981 339.06 340.30 341.21 342.33 342.74 342.08 340.32 338.26 336.52 336.68 338.19 339.44
#> 1982 340.57 341.44 342.53 343.39 343.96 343.18 341.88 339.65 337.81 337.69 339.09 340.32
#> 1983 341.20 342.35 342.93 344.77 345.58 345.14 343.81 342.21 339.69 339.82 340.98 342.82
#> 1984 343.52 344.33 345.11 346.88 347.25 346.62 345.22 343.11 340.90 341.18 342.80 344.04
#> 1985 344.79 345.82 347.25 348.17 348.74 348.07 346.38 344.51 342.92 342.62 344.06 345.38
#> 1986 346.11 346.78 347.68 349.37 350.03 349.37 347.76 345.73 344.68 343.99 345.48 346.72
#> 1987 347.84 348.29 349.23 350.80 351.66 351.07 349.33 347.92 346.27 346.18 347.64 348.78
#> 1988 350.25 351.54 352.05 353.41 354.04 353.62 352.22 350.27 348.55 348.72 349.91 351.18
#> 1989 352.60 352.92 353.53 355.26 355.52 354.97 353.75 351.52 349.64 349.83 351.14 352.37
#> 1990 353.50 354.55 355.23 356.04 357.00 356.07 354.67 352.76 350.82 351.04 352.69 354.07
#> 1991 354.59 355.63 357.03 358.48 359.22 358.12 356.06 353.92 352.05 352.11 353.64 354.89
#> 1992 355.88 356.63 357.72 359.07 359.58 359.17 356.94 354.92 352.94 353.23 354.09 355.33
#> 1993 356.63 357.10 358.32 359.41 360.23 359.55 357.53 355.48 353.67 353.95 355.30 356.78
#> 1994 358.34 358.89 359.95 361.25 361.67 360.94 359.55 357.49 355.84 356.00 357.59 359.05
#> 1995 359.98 361.03 361.66 363.48 363.82 363.30 361.94 359.50 358.11 357.80 359.61 360.74
#> 1996 362.09 363.29 364.06 364.76 365.45 365.01 363.70 361.54 359.51 359.65 360.80 362.38
#> 1997 363.23 364.06 364.61 366.40 366.84 365.68 364.52 362.57 360.24 360.83 362.49 364.34
```

Tidy temporal data structure

From ts to tsibble

```
as_tsibble(co2)
```

```
#> # A tsibble: 468 x 2 [1M]
#>       index value
#>       <mth> <dbl>
#>  1 1959 Jan  315.
#>  2 1959 Feb  316.
#>  3 1959 Mar  316.
#>  4 1959 Apr  318.
#>  5 1959 May  318.
#>  6 1959 Jun  318
#>  7 1959 Jul  316.
#>  8 1959 Aug  315.
#>  9 1959 Sep  314.
#> 10 1959 Oct  313.
#> # … with 458 more rows
```

Tidy temporal data structure

Domestic tourism in Australia

```r
library(tsibble)
tourism
```

```
#> # A tsibble: 24,320 x 5 [1Q]
#> # Key:       Region, State, Purpose [304]
#>    Quarter Region   State           Purpose  Trips
#>      <qtr> <chr>    <chr>           <chr>    <dbl>
#>  1 1998 Q1 Adelaide South Australia Business  135.
#>  2 1998 Q2 Adelaide South Australia Business  110.
#>  3 1998 Q3 Adelaide South Australia Business  166.
#>  4 1998 Q4 Adelaide South Australia Business  127.
#>  5 1999 Q1 Adelaide South Australia Business  137.
#>  6 1999 Q2 Adelaide South Australia Business  200.
#>  7 1999 Q3 Adelaide South Australia Business  169.
#>  8 1999 Q4 Adelaide South Australia Business  134.
#>  9 2000 Q1 Adelaide South Australia Business  154.
#> 10 2000 Q2 Adelaide South Australia Business  169.
#> # … with 24,310 more rows
```

Exploring tourism in Australia

How does travel vary by purpose?

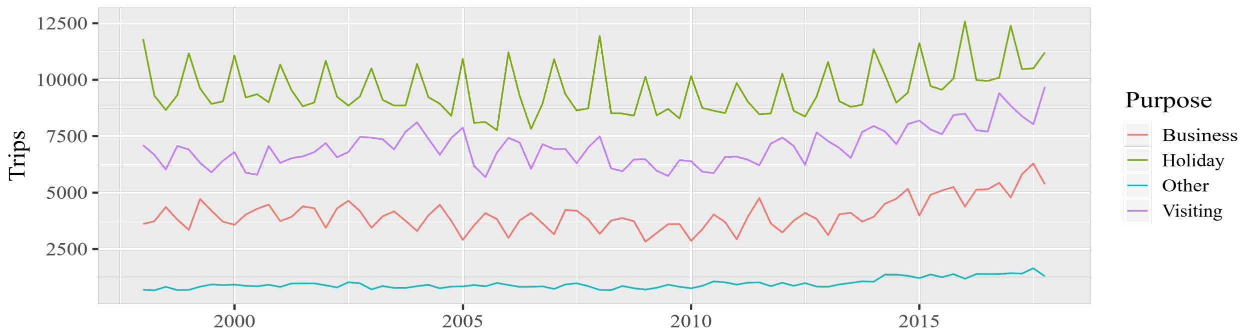
```
library(dplyr)
aus_travel <- tourism %>%
  group_by(Purpose) %>%
  summarise(Trips = sum(Trips))
```

```
#> # A tsibble: 320 x 3 [1Q]
#> # Key:       Purpose [4]
#>    Purpose  Quarter Trips
#>    <chr>      <qtr> <dbl>
#>  1 Business 1998 Q1 3599.
#>  2 Business 1998 Q2 3724.
#>  3 Business 1998 Q3 4356.
#>  4 Business 1998 Q4 3796.
#>  5 Business 1999 Q1 3335.
#>  6 Business 1999 Q2 4714.
#>  7 Business 1999 Q3 4190.
#>  8 Business 1999 Q4 3701.
#>  9 Business 2000 Q1 3562.
#> 10 Business 2000 Q2 4018.
#> # … with 310 more rows
```

Exploring tourism in Australia

How does travel vary by purpose?

```
library(ggplot2)
aus_travel %>%
  ggplot(aes(x = Quarter, y = Trips, colour = Purpose)) +
  geom_line()
```



feasts

- Graphics for time series
- Decompositions into structural components
- Feature extraction (summaries and statistical tests)

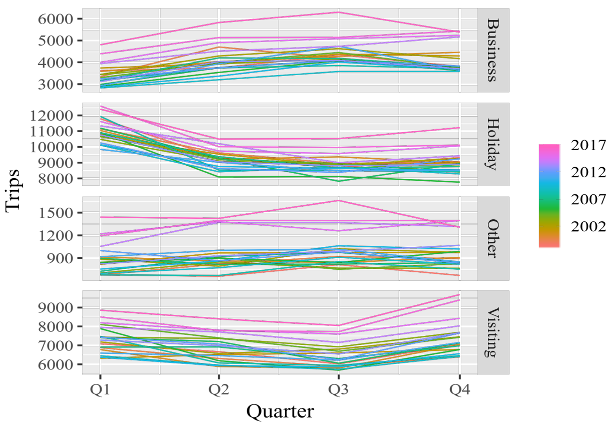
More information:

- tidyverts/feasts
- feasts site
- Rob Hyndman's talk at 11:48 tomorrow! (Ariane 1+2)

Exploring tourism in Australia

Time series graphics: season plots

```r
library(feasts)
aus_travel %>%
  gg_season(Trips)
```



Highlights seasonal structure by wrapping the x-axis over seasonal periods (years).

- Peak holiday travel in Q1 (summer)
- Peak business travel in Q2 & Q3
- Peak visiting travel in Q4 & Q1
- Other travel is largely non-seasonal

Exploring tourism in Australia

Time series decomposition: STL

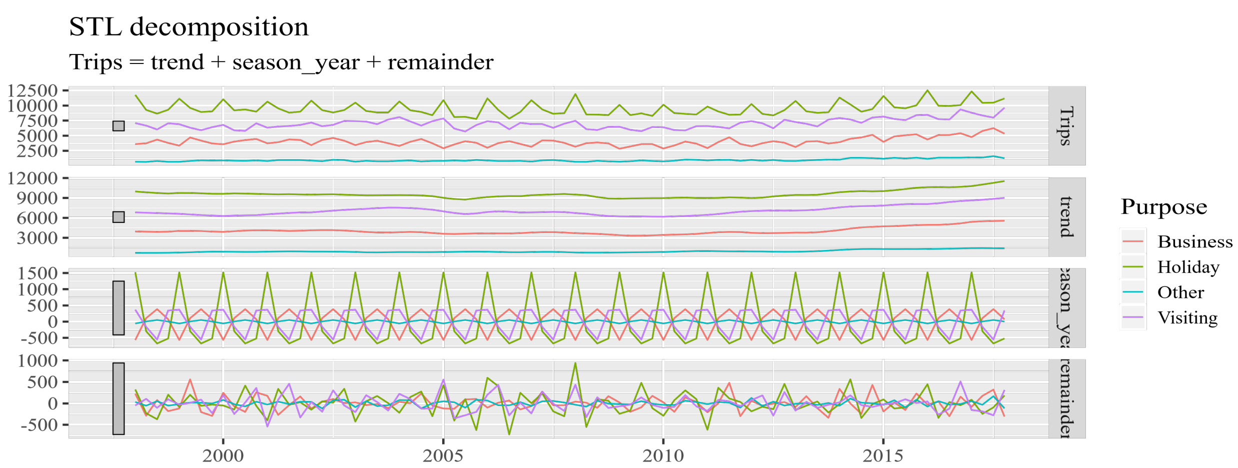
```r
aus_travel_stl <- aus_travel %>%
  STL(Trips ~ season(window = "periodic"))
```

```
#> # A dable:           320 x 7 [1Q]
#> # Key:               Purpose [4]
#> # STL Decomposition: Trips = trend + season_year + remainder
#>    Purpose  Quarter Trips trend season_year remainder season_adjust
#>    <chr>      <qtr> <dbl> <dbl>       <dbl>     <dbl>         <dbl>
#>  1 Business 1998 Q1 3599. 3949.      -577.      227.          4175.
#>  2 Business 1998 Q2 3724. 3910.       109.     -296.          3614.
#>  3 Business 1998 Q3 4356. 3886.       393.       77.0         3963.
#>  4 Business 1998 Q4 3796. 3910.        74.0    -188.          3722.
#>  5 Business 1999 Q1 3335. 4034.      -577.     -123.          3911.
#>  6 Business 1999 Q2 4714. 4034.       109.      571.          4605.
#>  7 Business 1999 Q3 4190. 4006.       393.     -210.          3797.
#>  8 Business 1999 Q4 3701. 3928.        74.0    -301.          3627.
#>  9 Business 2000 Q1 3562. 3889.      -577.      250.          4139.
#> 10 Business 2000 Q2 4018. 3981.       109.      -73.1         3908.
#> # … with 310 more rows
```

Exploring tourism in Australia

Time series decomposition: STL

```r
aus_travel_stl %>%
  autoplot()
```



Exploring tourism in Australia

Time series features: STL

```
aus_travel %>%
  features(Trips, feature_set(tags = "stl"))
```

```
#> # A tibble: 4 x 8
#>   Purpose  trend_strength seasonal_strength_year spikiness linearity curvature seasonal_pe
#>   <chr>             <dbl>                  <dbl>     <dbl>     <dbl>     <dbl>
#> 1 Business          0.899                  0.785   496322.     2327.     3611.
#> 2 Holiday           0.813                  0.910  2540902.     2019.     4257.
#> 3 Other             0.927                  0.357     3537.     1410.      908.
#> 4 Visiting          0.926                  0.796   601067.     3897.     3070.
```

- All series are trended
- Holiday travel is most seasonal, "Other" travel is least seasonal
- Seasonal peaks and troughs switch are Q1 and Q3 (varied by purpose)

Exploring tourism in Australia

Seeing the bigger picture

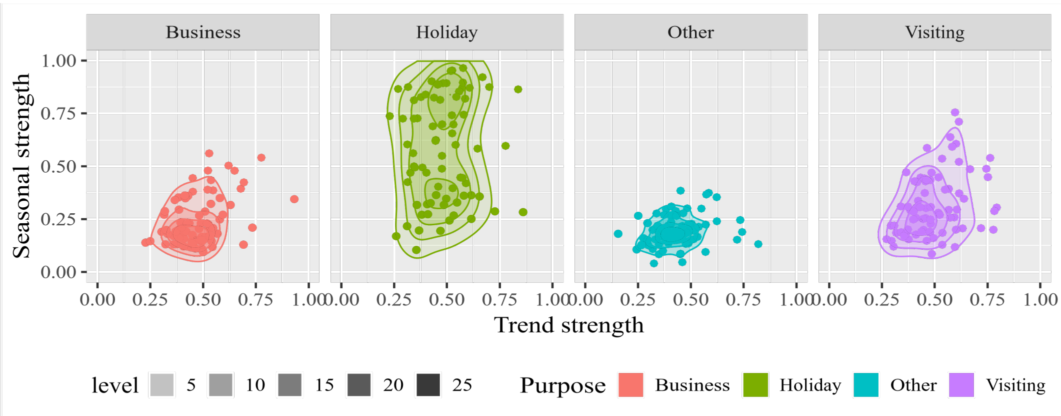
```
tourism
```

```
#> # A tsibble: 24,320 x 5 [1Q]
#> # Key:       Region, State, Purpose [304]
#>    Quarter Region   State           Purpose  Trips
#>      <qtr> <chr>    <chr>           <chr>    <dbl>
#>  1 1998 Q1 Adelaide South Australia Business  135.
#>  2 1998 Q2 Adelaide South Australia Business  110.
#>  3 1998 Q3 Adelaide South Australia Business  166.
#>  4 1998 Q4 Adelaide South Australia Business  127.
#>  5 1999 Q1 Adelaide South Australia Business  137.
#>  6 1999 Q2 Adelaide South Australia Business  200.
#>  7 1999 Q3 Adelaide South Australia Business  169.
#>  8 1999 Q4 Adelaide South Australia Business  134.
#>  9 2000 Q1 Adelaide South Australia Business  154.
#> 10 2000 Q2 Adelaide South Australia Business  169.
#> # … with 24,310 more rows
```

Exploring tourism in Australia

Seeing the bigger picture

```r
tourism_features <- tourism %>%
  features(Trips, feature_set(tags = "stl"))
```



fable

- Tidy evolution of the forecast package
- Models for time series forecasting
- Tools for analysing and manipulating models

More information:

- tidyverts/fable
- fable site
- Forecasting: Principles and Practice (3rd Ed.)

Forecasting with fable

Look at the data

```r
aus_travel
```

```
#> # A tsibble: 320 x 3 [1Q]
#> # Key:       Purpose [4]
#>    Purpose  Quarter Trips
#>    <chr>      <qtr> <dbl>
#>  1 Business 1998 Q1 3599.
#>  2 Business 1998 Q2 3724.
#>  3 Business 1998 Q3 4356.
#>  4 Business 1998 Q4 3796.
#>  5 Business 1999 Q1 3335.
#>  6 Business 1999 Q2 4714.
#>  7 Business 1999 Q3 4190.
#>  8 Business 1999 Q4 3701.
#>  9 Business 2000 Q1 3562.
#> 10 Business 2000 Q2 4018.
#> # … with 310 more rows
```

Forecasting with fable

Specify and estimate a model

```
aus_travel %>%
  model(ETS(Trips))
```

```
#> # A mable: 4 x 2
#> # Key:     Purpose [4]
#>   Purpose  `ETS(Trips)`
#>   <chr>    <model>
#> 1 Business <ETS(M,N,A)>
#> 2 Holiday  <ETS(M,N,M)>
#> 3 Other    <ETS(M,N,A)>
#> 4 Visiting <ETS(M,N,M)>
```

Forecasting with fable

Make some forecasts

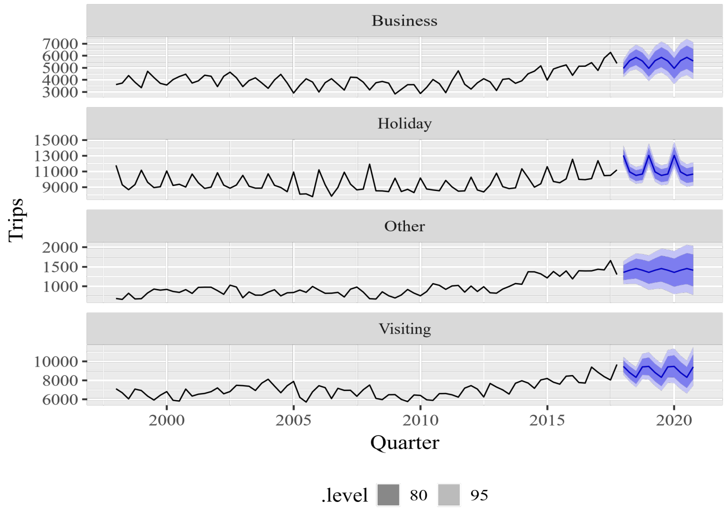
```r
aus_travel %>%
  model(ETS(Trips)) %>%
  forecast(h = "3 years")
```

```
#> # A fable: 48 x 5 [1Q]
#> # Key:     Purpose, .model [4]
#>    Purpose  .model     Quarter Trips .distribution
#>    <chr>    <chr>        <qtr> <dbl> <dist>
#>  1 Business ETS(Trips) 2018 Q1 4937. N(4937, 137658)
#>  2 Business ETS(Trips) 2018 Q2 5604. N(5604, 209086)
#>  3 Business ETS(Trips) 2018 Q3 5888. N(5888, 268396)
#>  4 Business ETS(Trips) 2018 Q4 5575. N(5575, 293305)
#>  5 Business ETS(Trips) 2019 Q1 4937. N(4937, 295999)
#>  6 Business ETS(Trips) 2019 Q2 5604. N(5604, 367634)
#>  7 Business ETS(Trips) 2019 Q3 5888. N(5888, 427147)
#>  8 Business ETS(Trips) 2019 Q4 5575. N(5575, 452257)
#>  9 Business ETS(Trips) 2020 Q1 4937. N(4937, 455161)
#> 10 Business ETS(Trips) 2020 Q2 5604. N(5604, 527003)
#> # … with 38 more rows
```

Forecasting with fable

Visualise the results!

```
aus_travel %>%
  model(ETS(Trips)) %>%
  forecast(h = "3 years")
  autoplot(aus_travel)
```



Forecasting with fable

Estimate multiple models

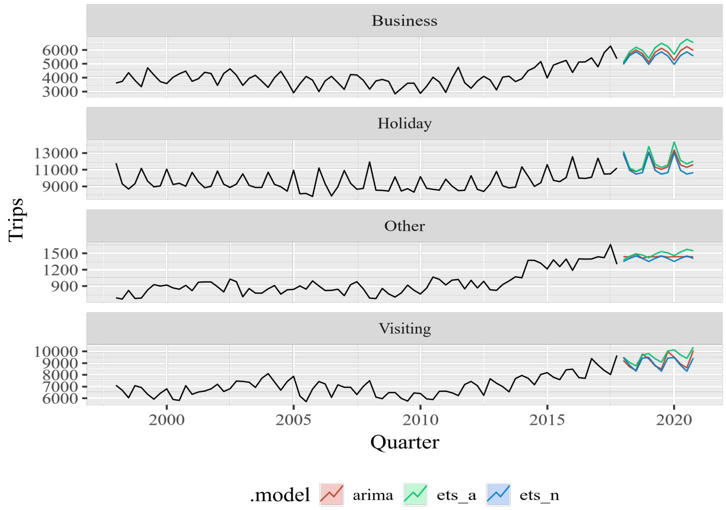
```
aus_travel %>%
  model(
    ets_n = ETS(Trips ~ t
    ets_a = ETS(Trips ~ t
    arima = ARIMA(Trips)
  )
```

```
#> # A mable: 4 x 4
#> # Key:     Purpose [4]
#>   Purpose  ets_n        ets_a        arima
#>   <chr>    <model>      <model>      <model>
#> 1 Business <ETS(M,N,A)> <ETS(M,A,A)> <ARIMA(0,1,1)(0,1,
#> 2 Holiday  <ETS(M,N,M)> <ETS(M,A,M)> <ARIMA(0,1,1)(0,1,
#> 3 Other    <ETS(M,N,A)> <ETS(M,A,A)> <ARIMA(0,1,1)(1,0,
#> 4 Visiting <ETS(M,N,M)> <ETS(A,A,A)> <ARIMA(1,0,1)(2,1,
```

Forecasting with fable

Forecasts from multiple models

```
aus_travel %>%
  model(
    ets_n = ETS(Trips ~ t
    ets_a = ETS(Trips ~ t
    arima = ARIMA(Trips)
  ) %>%
  forecast(h = "3 years")
  autoplot(aus_travel, le
```



Flexibility with...

Combination forecasting

(models are better when they work together)*

* Most of the time, performance improvements not guaranteed! :)

Ensemble forecasting

A simple average of forecasts

```
aus_travel %>%
  model(
    ets_n = ETS(Trips ~ t
    ets_a = ETS(Trips ~ t
    arima = ARIMA(Trips)
  )
```

```
#> # A mable: 4 x 4
#> # Key:     Purpose [4]
#>   Purpose  ets_n        ets_a        arima
#>   <chr>    <model>      <model>      <model>
#> 1 Business <ETS(M,N,A)> <ETS(M,A,A)> <ARIMA(0,1,1)(0,1,
#> 2 Holiday  <ETS(M,N,M)> <ETS(M,A,M)> <ARIMA(0,1,1)(0,1,
#> 3 Other    <ETS(M,N,A)> <ETS(M,A,A)> <ARIMA(0,1,1)(1,0,
#> 4 Visiting <ETS(M,N,M)> <ETS(A,A,A)> <ARIMA(1,0,1)(2,1,
```

Ensemble forecasting

A simple average of forecasts

```
aus_travel %>%
  model(
    ets_n = ETS(Trips ~ t
    ets_a = ETS(Trips ~ t
    arima = ARIMA(Trips)
  ) %>%
  mutate(
    combn = (ets_n + ets_
  )
```

```
#> # A mable: 4 x 5
#> # Key:     Purpose [4]
#>   Purpose  ets_n        ets_a        arima
#>   <chr>    <model>      <model>      <model>
#> 1 Business <ETS(M,N,A)> <ETS(M,A,A)> <ARIMA(0,1,1)(0,1,
#> 2 Holiday  <ETS(M,N,M)> <ETS(M,A,M)> <ARIMA(0,1,1)(0,1,
#> 3 Other    <ETS(M,N,A)> <ETS(M,A,A)> <ARIMA(0,1,1)(1,0,
#> 4 Visiting <ETS(M,N,M)> <ETS(A,A,A)> <ARIMA(1,0,1)(2,1,
```

Ensemble forecasting

A simple average of forecasts

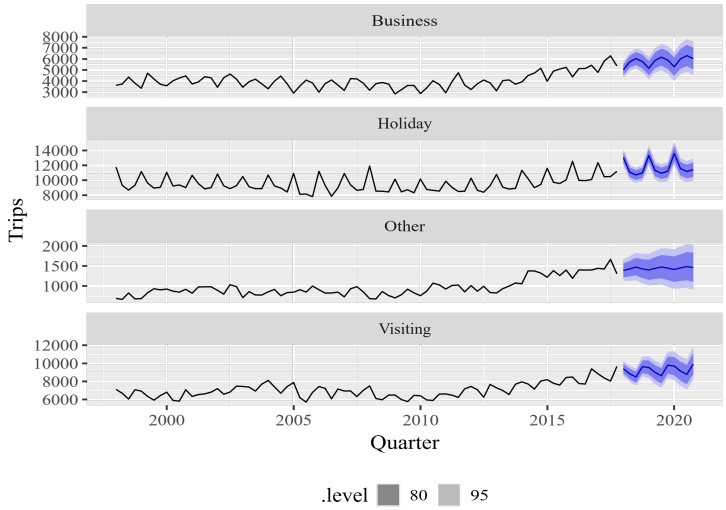
```
aus_travel %>%
  model(
    ets_n = ETS(Trips ~ t
    ets_a = ETS(Trips ~ t
    arima = ARIMA(Trips)
  ) %>%
  mutate(
    combn = (ets_n + ets_
  ) %>%
  select(combn) %>%
  forecast(h = "3 years")
```

```
#> # A fable: 48 x 5 [1Q]
#> # Key:     Purpose, .model [4]
#>    Purpose  .model Quarter Trips .distribution
#>    <chr>    <chr>    <qtr> <dbl> <dist>
#>  1 Business combn  2018 Q1 5004. N(5004, 122410)
#>  2 Business combn  2018 Q2 5731. N(5731, 174693)
#>  3 Business combn  2018 Q3 6036. N(6036, 218982)
#>  4 Business combn  2018 Q4 5759. N(5759, 244190)
#>  5 Business combn  2019 Q1 5144. N(5144, 260489)
#>  6 Business combn  2019 Q2 5871. N(5871, 321799)
#>  7 Business combn  2019 Q3 6176. N(6176, 375379)
#>  8 Business combn  2019 Q4 5898. N(5898, 409760)
#>  9 Business combn  2020 Q1 5284. N(5284, 435676)
#> 10 Business combn  2020 Q2 6010. N(6010, 507587)
#> # … with 38 more rows
```

Ensemble forecasting

A simple average of forecasts

```
aus_travel %>%
  model(
    ets_n = ETS(Trips ~ t
    ets_a = ETS(Trips ~ t
    arima = ARIMA(Trips)
  ) %>%
  mutate(
    combn = (ets_n + ets_
  ) %>%
  select(combn) %>%
  forecast(h = "3 years")
  autoplot(aus_travel)
```



Hybrid forecasting

Combining forecasts of decomposed components

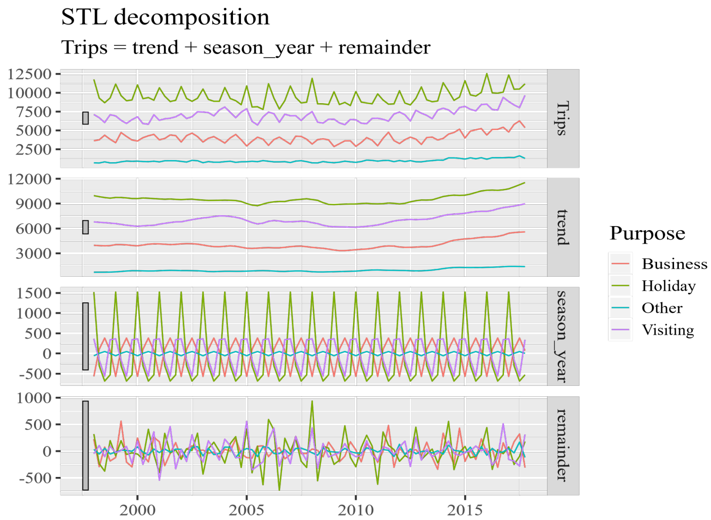
```
aus_travel_stl <- aus_tra
  STL(Trips ~ season(winc
```

```
#> # A dable:           320 x 7 [1Q]
#> # Key:               Purpose [4]
#> # STL Decomposition: Trips = trend + season_year + rema
#>    Purpose  Quarter Trips trend season_year remainder s
#>    <chr>      <qtr> <dbl> <dbl>       <dbl>     <dbl>
#>  1 Business 1998 Q1 3599. 3949.      -577.      227.
#>  2 Business 1998 Q2 3724. 3910.       109.     -296.
#>  3 Business 1998 Q3 4356. 3886.       393.       77.0
#>  4 Business 1998 Q4 3796. 3910.        74.0    -188.
#>  5 Business 1999 Q1 3335. 4034.      -577.     -123.
#>  6 Business 1999 Q2 4714. 4034.       109.      571.
#>  7 Business 1999 Q3 4190. 4006.       393.     -210.
#>  8 Business 1999 Q4 3701. 3928.        74.0    -301.
#>  9 Business 2000 Q1 3562. 3889.      -577.      250.
#> 10 Business 2000 Q2 4018. 3981.       109.      -73.1
#> # … with 310 more rows
```

Hybrid forecasting

Combining forecasts of decomposed components

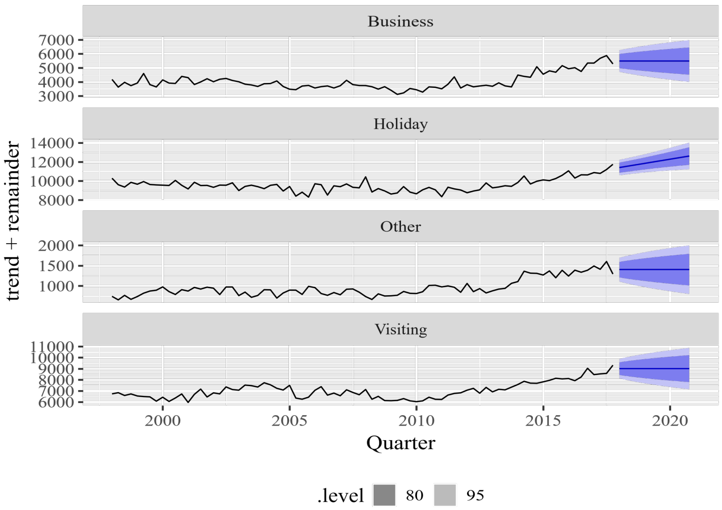
```
aus_travel_stl <- aus_tra
  STL(Trips ~ season(winc
aus_travel_stl %>%
  autoplot()
```



Hybrid forecasting

Combining forecasts of decomposed components

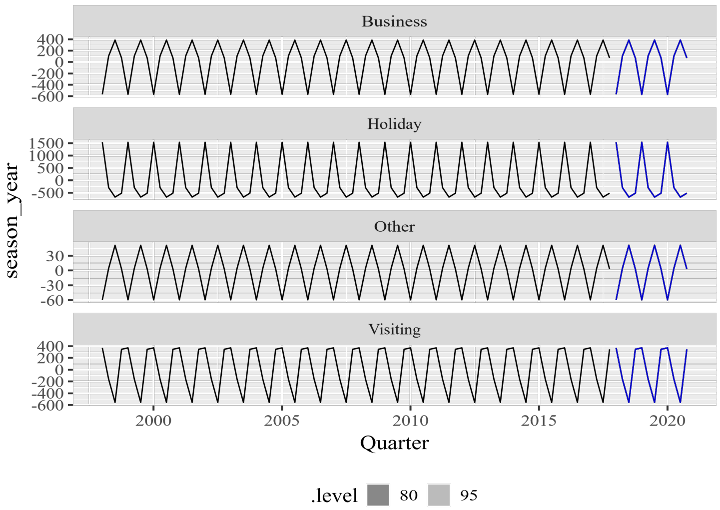
```
aus_travel_stl <- aus_tra
  STL(Trips ~ season(winc
aus_travel_stl %>%
  model(
    deseas = ETS(trend +
  ) %>%
  forecast(h = "3 years")
  autoplot(aus_travel_stl
```



Hybrid forecasting

Combining forecasts of decomposed components

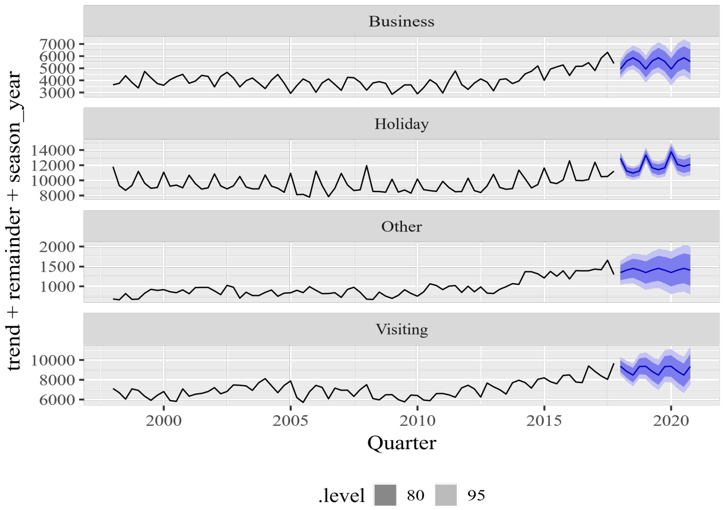
```
aus_travel_stl <- aus_tra
  STL(Trips ~ season(winc
aus_travel_stl %>%
  model(
    seas = SNAIVE(season_
  ) %>%
  forecast(h = "3 years")
  autoplot(aus_travel_stl
```



Hybrid forecasting

Combining forecasts of decomposed components

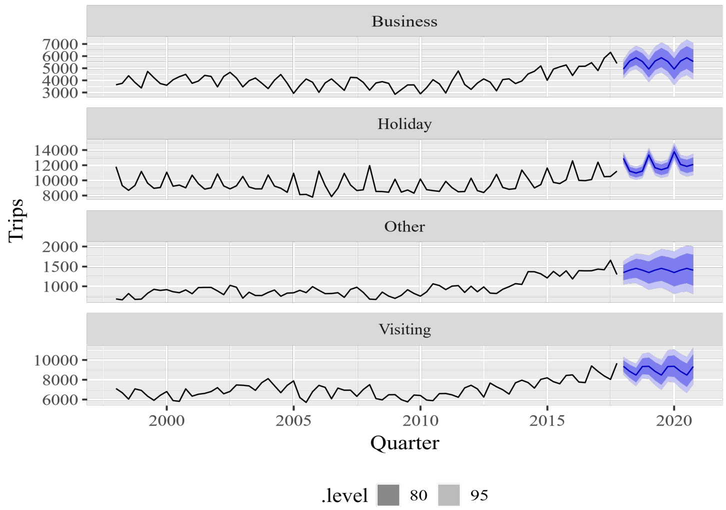
```
aus_travel_stl <- aus_tra
  STL(Trips ~ season(winc
aus_travel_stl %>%
  model(
    hybrid = ETS(trend +
      SNAIVE(season_year)
  ) %>%
  forecast(h = "3 years")
  autoplot(aus_travel_stl
```



Hybrid forecasting

Combining forecasts of decomposed components

```
aus_travel %>%
  model(
    hybrid = decompositic
      STL, Trips ~ seasor
      ETS(trend + remainc
      SNAIVE(season_year)
    )
  ) %>%
  forecast(h = "3 years")
  autoplot(aus_travel_stl
```



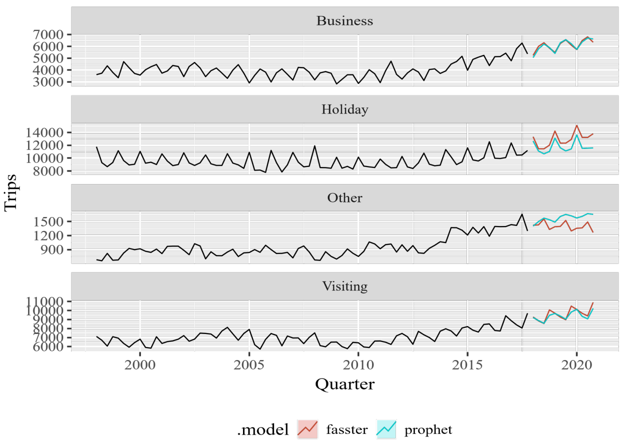
Flexibility with...

Extensibility

(get other people to implement models for you)

Adding new models to fable

```r
library(fasster)
library(fable.prophet)
aus_travel %>%
 model(
  fasster = fasster(Trips ~ sea
  prophet = prophet(Trips ~ sea
 ) %>%
 forecast(h = "3 years") %>%
 autoplot(aus_travel, level = N
```



- Direct comparison of new models
- Unified modelling interface
- Ensemble and hybrid support
- and more...

Summary

```r
library(fable)
library(tidyverse)
tsibble::tourism %>%
  group_by(Purpose) %>%
  summarise(Trips = sum(Trips)) %>%
  model(ets_n = ETS(Trips ~ trend("N")),
        ets_a = ETS(Trips ~ trend("A")),
        arima = ARIMA(Trips),
        fasster = fasster::fasster(Trips ~ season("year") + pol
        prophet = fable.prophet::prophet(Trips ~ season("year")
  mutate(combn = (ets_n + ets_a + arima + fasster)/4) %>%
  forecast(h = "3 years")
```

Summary

```
library(fable)
library(tidyverse)
tsibble::tourism %>%
  group_by(Purpose) %>%
  summarise(Trips = sum(Trips)) %>%
  model(ets_n = ETS(Trips ~ trend("N")),
        ets_a = ETS(Trips ~ trend("A")),
        arima = ARIMA(Trips),
        fasster = fasster::fasster(Trips ~ season("year") + pol
        prophet = fable.prophet::prophet(Trips ~ season("year")
  mutate(combn = (ets_n + ets_a + arima + fasster)/4) %>%
  forecast(h = "3 years")
```

Tidy temporal data suitable for the future of time series.

Summary

```
library(fable)
library(tidyverse)
tsibble::tourism %>%
  group_by(Purpose) %>%
  summarise(Trips = sum(Trips)) %>%
  model(ets_n = ETS(Trips ~ trend("N")),
        ets_a = ETS(Trips ~ trend("A")),
        arima = ARIMA(Trips),
        fasster = fasster::fasster(Trips ~ season("year") + pol
        prophet = fable.prophet::prophet(Trips ~ season("year")
  mutate(combn = (ets_n + ets_a + arima + fasster)/4) %>%
  forecast(h = "3 years")
```

Integrates seamlessly with the tidyverse.

Summary

```
library(fable)
library(tidyverse)
tsibble::tourism %>%
  group_by(Purpose) %>%
  summarise(Trips = sum(Trips)) %>%
  model(ets_n = ETS(Trips ~ trend("N")),
        ets_a = ETS(Trips ~ trend("A")),
        arima = ARIMA(Trips),
        fasster = fasster::fasster(Trips ~ season("year") + pol
        prophet = fable.prophet::prophet(Trips ~ season("year")
  mutate(combn = (ets_n + ets_a + arima + fasster)/4) %>%
  forecast(h = "3 years")
```

Flexible, and succinct formula model specification.

Summary

```r
library(fable)
library(tidyverse)
tsibble::tourism %>%
  group_by(Purpose) %>%
  summarise(Trips = sum(Trips)) %>%
  model(ets_n = ETS(Trips ~ trend("N")),
        ets_a = ETS(Trips ~ trend("A")),
        arima = ARIMA(Trips),
        fasster = fasster::fasster(Trips ~ season("year") + pol
        prophet = fable.prophet::prophet(Trips ~ season("year")
  mutate(combn = (ets_n + ets_a + arima + fasster)/4) %>%
  forecast(h = "3 years")
```

Extensible by design.

Summary

```r
library(fable)
library(tidyverse)
tsibble::tourism %>%
  group_by(Purpose) %>%
  summarise(Trips = sum(Trips)) %>%
  model(ets_n = ETS(Trips ~ trend("N")),
        ets_a = ETS(Trips ~ trend("A")),
        arima = ARIMA(Trips),
        fasster = fasster::fasster(Trips ~ season("year") + pol
        prophet = fable.prophet::prophet(Trips ~ season("year")
  mutate(combn = (ets_n + ets_a + arima + fasster)/4) %>%
  forecast(h = "3 years")
```

Natural interface for model combinations.

Summary

```
library(fable)
library(tidyverse)
tsibble::tourism %>%
  group_by(Purpose) %>%
  summarise(Trips = sum(Trips)) %>%
  model(ets_n = ETS(Trips ~ trend("N")),
        ets_a = ETS(Trips ~ trend("A")),
        arima = ARIMA(Trips),
        fasster = fasster::fasster(Trips ~ season("year") + pol
        prophet = fable.prophet::prophet(Trips ~ season("year")
  mutate(combn = (ets_n + ets_a + arima + fasster)/4) %>%
  forecast(h = "3 years")
```

Distributional forecasts in a data format.

Acknowledgements

Rob Hyndman

Earo Wang

Join our group.

Monash University is now hiring in business analytics.

See bit.ly/monash-ba for details.

Thanks!

Learn more: fable.tidyverts.org

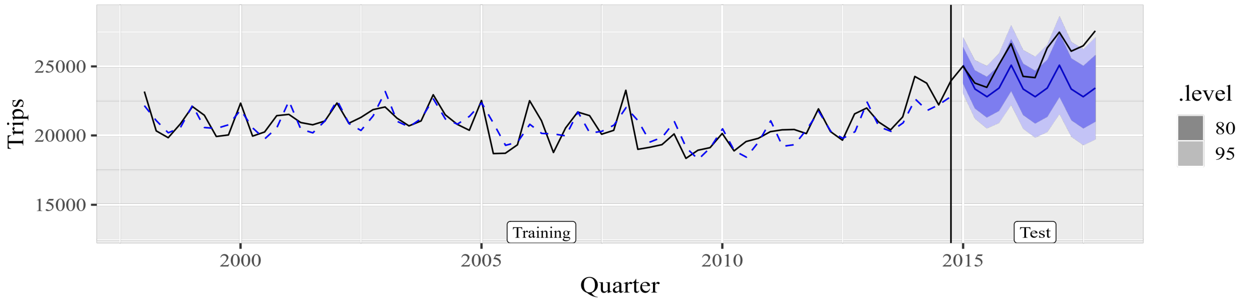
Keep updated: tidyverts.org

Review slides: slides.mitchelloharawild.com/user2019

Bonus

Accuracy evaluation

Forecast accuracy evaluation



MASE: Mean absolute scaled error

$$\text{MASE} = \frac{1}{h}\sum_{t=T+1}^{T+h}\left|\frac{e_t}{\text{scale}}\right|, \qquad \text{scale} = \frac{1}{T-m}$$

Comparing multiple models

Forecast accuracy

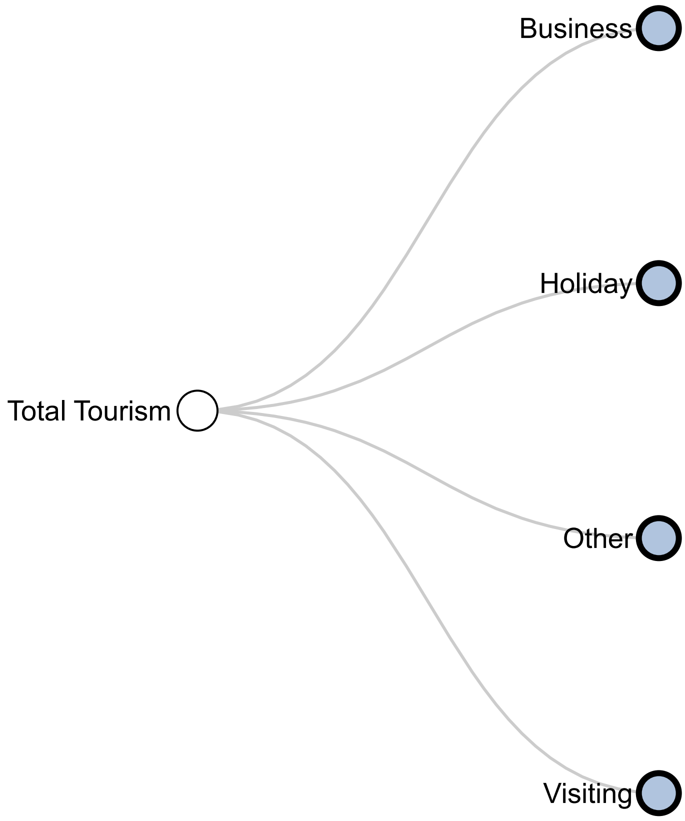
```r
aus_travel %>%
  filter(Quarter < yearqu
  model(
    ets_n = ETS(Trips ~ t
    ets_a = ETS(Trips ~ t
    arima = ARIMA(Trips)
  ) %>%
  forecast(h = "3 years")
  accuracy(aus_travel) %>
  arrange(MASE)
```

```
#> # A tibble: 12 x 10
#>    .model Purpose  .type     ME  RMSE   MAE   MPE  MAP
#>    <chr>  <chr>    <chr>  <dbl> <dbl> <dbl> <dbl> <dbl
#>  1 arima  Other    Test   -12.4  112.  76.3 -1.59  5.6
#>  2 ets_a  Other    Test   -43.9  101.  80.7 -3.66  6.0
#>  3 ets_n  Other    Test    71.4  129. 102.   4.62  7.1
#>  4 arima  Visiting Test   538.   686. 539.   6.18  6.2
#>  5 ets_a  Holiday  Test   581.   700. 581.   5.36  5.3
#>  6 ets_a  Visiting Test   612.   758. 612.   7.03  7.0
#>  7 ets_a  Business Test   423.   573. 448.   7.66  8.2
#>  8 ets_n  Visiting Test   652.   800. 652.   7.53  7.5
#>  9 ets_n  Business Test   511.   645. 513.   9.50  9.5
#> 10 ets_n  Holiday  Test   939.  1061. 939.   8.69  8.6
#> 11 arima  Business Test   820.   968. 883.  15.1  16.6
#> 12 arima  Holiday  Test  1261.  1380. 1261. 11.7  11.7
```

Bonus

Forecast reconciliation

Forecast reconciliation



Consider *all* series in the data...

```
tourism_aggregated <- tourism %>%
  aggregate_key((State / Region) * Purpose, Trips = sum(Trips))
```

```
#> # A tsibble: 34,000 x 5 [1Q]
#> # Key:       State, Region, Purpose [425]
#>    State   Region  Purpose Quarter  Trips
#>    <chr>   <chr>   <chr>     <qtr>  <dbl>
#>  1 <total> <total> <total> 1998 Q1 23182.
#>  2 <total> <total> <total> 1998 Q2 20323.
#>  3 <total> <total> <total> 1998 Q3 19827.
#>  4 <total> <total> <total> 1998 Q4 20830.
#>  5 <total> <total> <total> 1999 Q1 22087.
#>  6 <total> <total> <total> 1999 Q2 21458.
#>  7 <total> <total> <total> 1999 Q3 19914.
#>  8 <total> <total> <total> 1999 Q4 20028.
#>  9 <total> <total> <total> 2000 Q1 22339.
#> 10 <total> <total> <total> 2000 Q2 19941.
#> # … with 33,990 more rows
```

Modelling may take a while...

Fortunately, this is embarrassingly parallel!

```
library(future)
plan(multiprocess)
tourism_fit <- tourism_aggregated %>%
  filter(Quarter < yearquarter("2015 Q1")) %>%
  model(
    ets_n = ETS(Trips ~ trend("N")),
    ets_a = ETS(Trips ~ trend("A")),
    arima = ARIMA(Trips)
  ) %>%
  mutate(combn = (ets_n + ets_a + arima)/3)
```

```
#> # A mable: 425 x 7
#> # Key:     State, Region, Purpose [425]
#>    State  Region  Purpose  ets_n        ets_a        arima                             com
#>    <chr>  <chr>   <chr>    <model>      <model>      <model>                           <mo
#>  1 <total> <total> <total>  <ETS(M,N,M)> <ETS(M,A,M)> <ARIMA(1,0,1)(1,1,0)[4]>          <CO
#>  2 <total> <total> Business <ETS(M,N,M)> <ETS(M,A,A)> <ARIMA(0,0,1)(2,1,0)[4]>          <CO
#>  3 <total> <total> Holiday  <ETS(M,N,M)> <ETS(M,A,M)> <ARIMA(0,0,0)(0,1,2)[4]>          <CO
#>  4 <total> <total> Other    <ETS(M,N,A)> <ETS(M,A,A)> <ARIMA(0,1,1)(1,0,0)[4]>          <CO
#>  5 <total> <total> Visiting <ETS(M,N,M)> <ETS(A,A,A)> <ARIMA(1,0,1)(2,1,0)[4]>          <CO
#>  6 ACT     <total> <total>  <ETS(A,N,N)> <ETS(M,A,N)> <ARIMA(0,0,0) w/ mean>            <CO
#>  7 ACT     <total> Business <ETS(M,N,A)> <ETS(M,A,A)> <ARIMA(0,0,0)(1,0,0)[4] w/ mean> <CO
#>  8 ACT     <total> Holiday  <ETS(M,N,A)> <ETS(M,A,A)> <ARIMA(0,0,0)(1,0,0)[4] w/ mean> <CO
#>  9 ACT     <total> Other    <ETS(A,N,N)> <ETS(M,A,N)> <ARIMA(0,0,0) w/ mean>            <CO
#> 10 ACT     <total> Visiting <ETS(A,N,N)> <ETS(A,A,N)> <ARIMA(1,0,0) w/ mean>            <CO
#> # … with 415 more rows
```

Forecast reconciliation

MinT with covariance shrink

```
tourism_fc_reconciled <- tourism_fit %>%
  reconcile(coherent = min_trace(combn, method = "shrink")) %>%
  forecast(h = "3 years")
```

```
#> # A fable: 25,500 x 7 [1Q]
#> # Key:     State, Region, Purpose, .model [2,125]
#>    State   Region  Purpose .model Quarter  Trips .distribution
#>    <chr>   <chr>   <chr>   <chr>    <qtr>  <dbl> <dist>
#>  1 <total> <total> <total> ets_n  2015 Q1 25108. N(25108, 1078489)
#>  2 <total> <total> <total> ets_n  2015 Q2 23339. N(23339, 1171475)
#>  3 <total> <total> <total> ets_n  2015 Q3 22771. N(22771, 1343242)
#>  4 <total> <total> <total> ets_n  2015 Q4 23417. N(23417, 1661871)
#>  5 <total> <total> <total> ets_n  2016 Q1 25108. N(25108, 2188347)
#>  6 <total> <total> <total> ets_n  2016 Q2 23339. N(23339, 2130906)
#>  7 <total> <total> <total> ets_n  2016 Q3 22771. N(22771, 2256904)
#>  8 <total> <total> <total> ets_n  2016 Q4 23417. N(23417, 2628507)
#>  9 <total> <total> <total> ets_n  2017 Q1 25108. N(25108, 3300157)
#> 10 <total> <total> <total> ets_n  2017 Q2 23339. N(23339, 3092023)
#> # … with 25,490 more rows
```

(Interface for reconciliation is still experimental.)

Forecast reconciliation

Is it better? Forecast accuracy

```
tourism_fc_reconciled %>%
  accuracy(tourism_aggregated) %>%
  group_by(.model) %>%
  summarise_at(vars(ME:ACF1), median) %>%
  arrange(MASE)
```

```
#> # A tibble: 5 x 8
#>   .model      ME  RMSE   MAE   MPE  MAPE  MASE    ACF1
#>   <chr>    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>   <dbl>
#> 1 coherent  6.34  18.2  14.6 NA      22.6 0.934 -0.0834
#> 2 ets_n     8.73  18.9  15.1  7.23  25.0 0.982 -0.0852
#> 3 combn     8.77  19.4  15.5  7.39  24.1 0.995 -0.0848
#> 4 ets_a     7.65  20.2  15.8 NA      26.0 1.03  -0.0733
#> 5 arima     9.01  20.3  16.5  8.07  26.2 1.07  -0.0849
```

Bonus

Multivariate modelling

Multivariate modelling

```
lung_deaths <- cbind(mdeaths, fdeaths) %>%
  as_tsibble(pivot_longer = FALSE)
```

```
#> # A tsibble: 72 x 3 [1M]
#>       index mdeaths fdeaths
#>       <mth>   <dbl>   <dbl>
#>  1 1974 Jan    2134     901
#>  2 1974 Feb    1863     689
#>  3 1974 Mar    1877     827
#>  4 1974 Apr    1877     677
#>  5 1974 May    1492     522
#>  6 1974 Jun    1249     406
#>  7 1974 Jul    1280     441
#>  8 1974 Aug    1131     393
#>  9 1974 Sep    1209     387
#> 10 1974 Oct    1492     582
#> # … with 62 more rows
```

Multivariate modelling

```
lung_deaths %>%
  model(VAR(vars(mdeaths, fdeaths) ~ AR(3) + fourier("year", 4)
  forecast() %>%
  autoplot(lung_deaths)
```