Ensemble forecasts with fable

Rob J Hyndman

NYR Conference: 14 August 2020

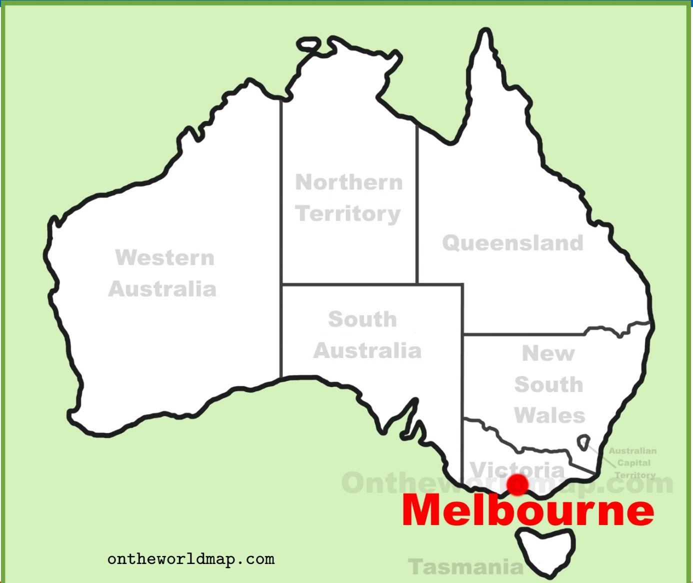
Outline

Outline

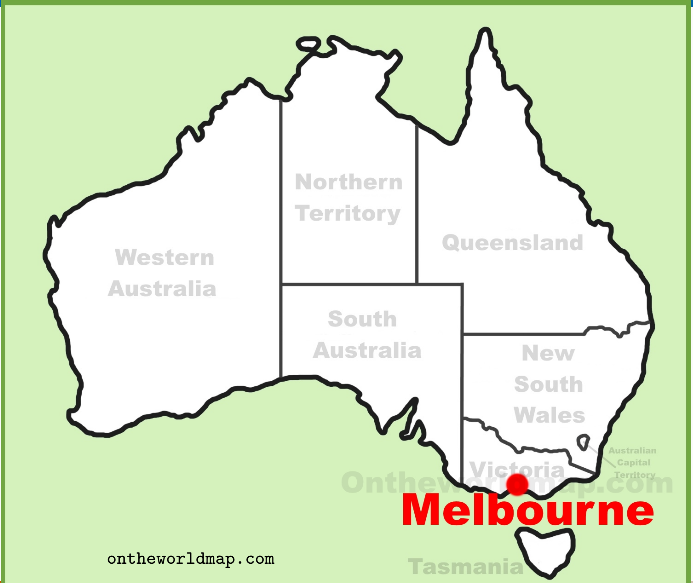
Where is Melbourne?

Where is Melbourne?

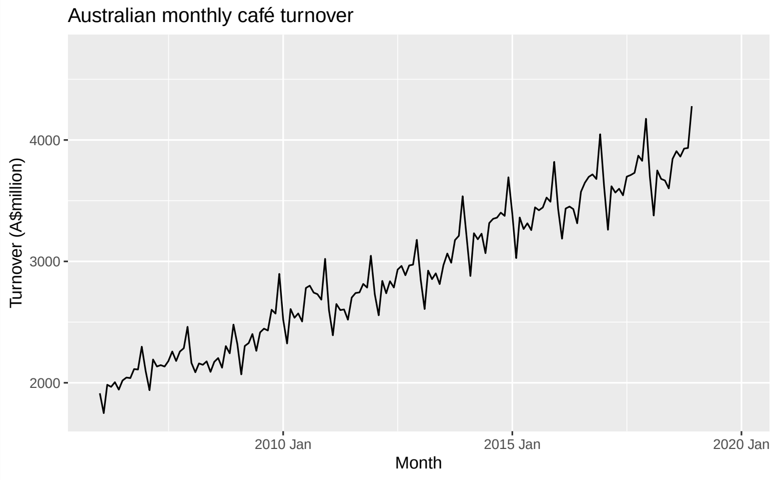
Where is Melbourne?



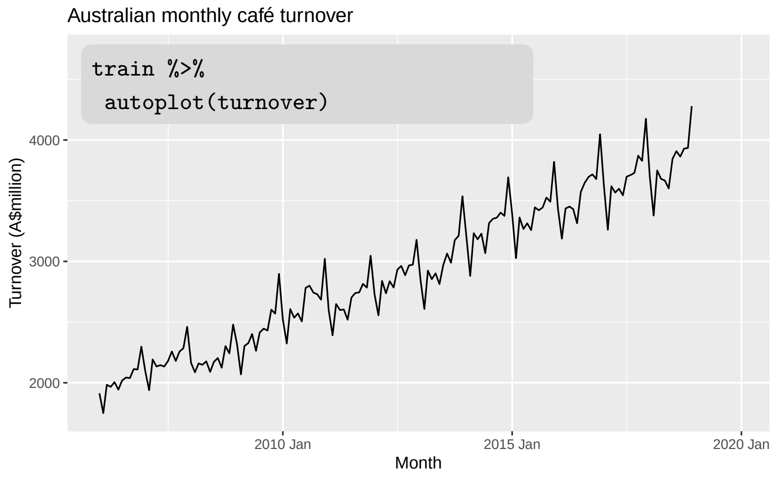
Where is Melbourne?



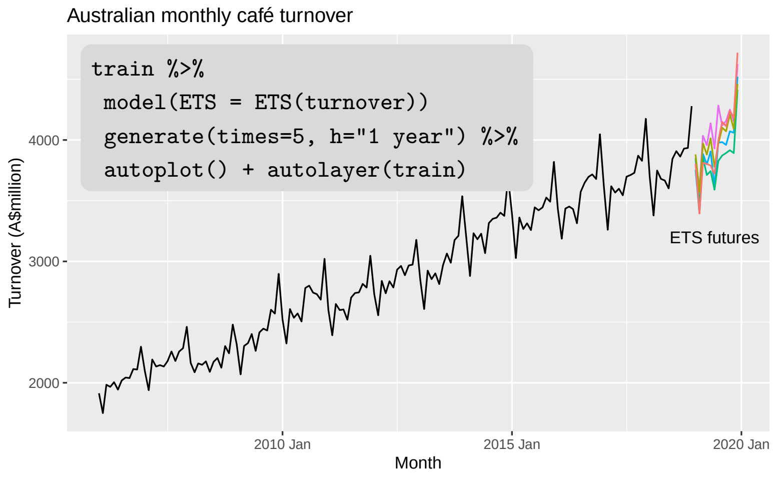
Australian monthly café turnover



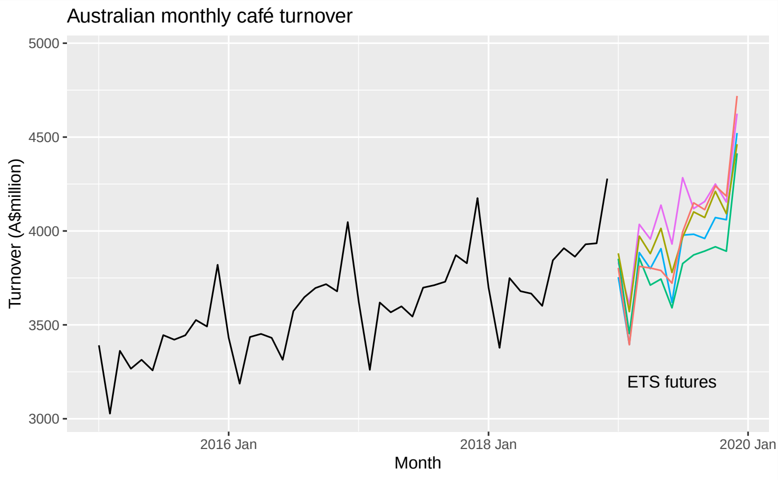
Australian monthly café turnover



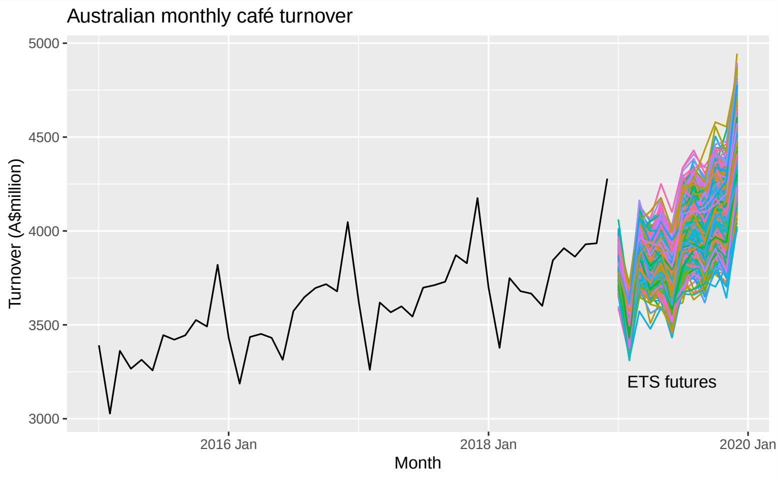
Australian monthly café turnover



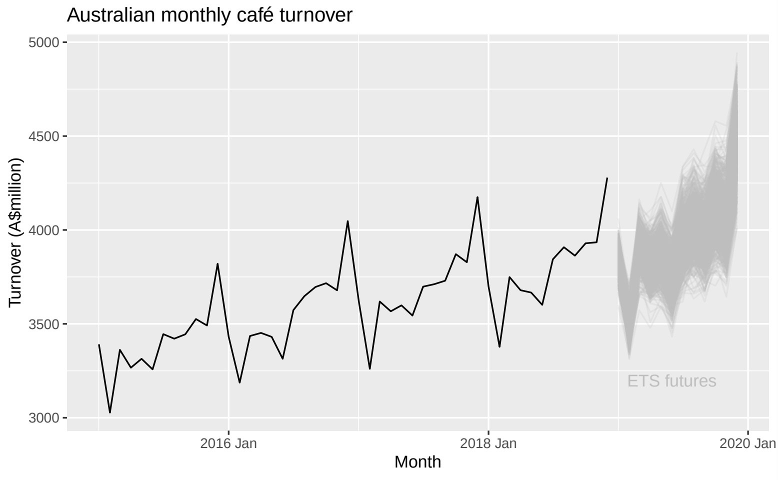
Australian monthly café turnover



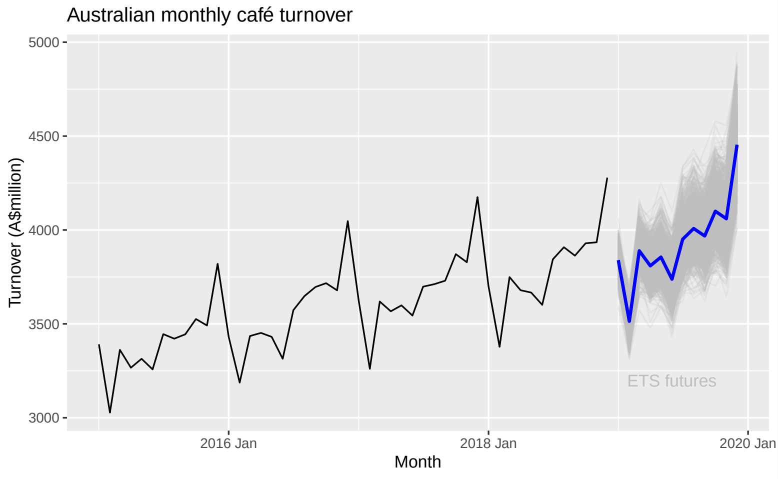
Australian monthly café turnover



Australian monthly café turnover

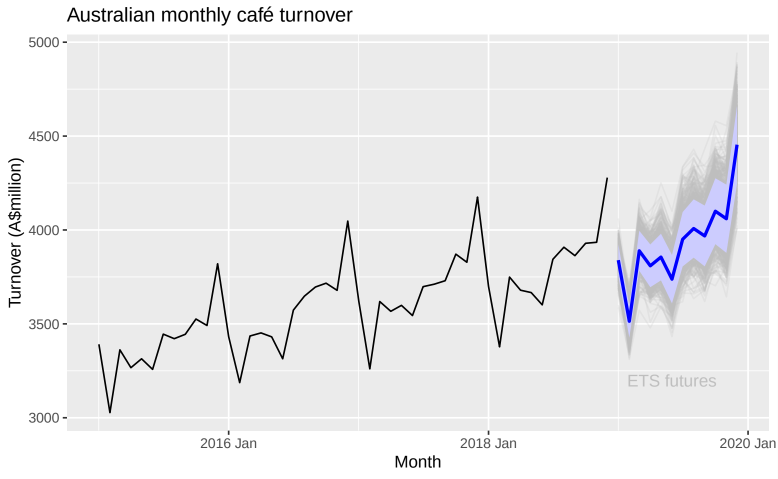


Forecasting using possible futures



Point forecasts: means of the sample paths.

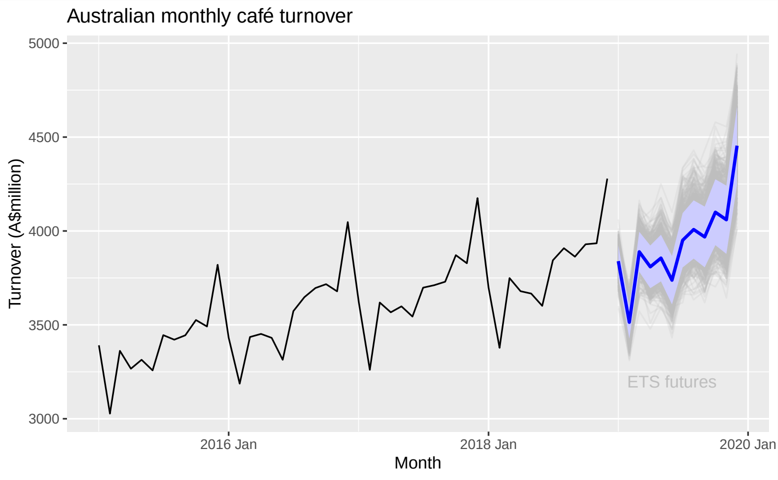
Forecasting using possible futures



Point forecasts: means of the sample paths.

Prediction intervals: middle 80% of the sample paths at each forecast horizon.

Forecasting using possible futures

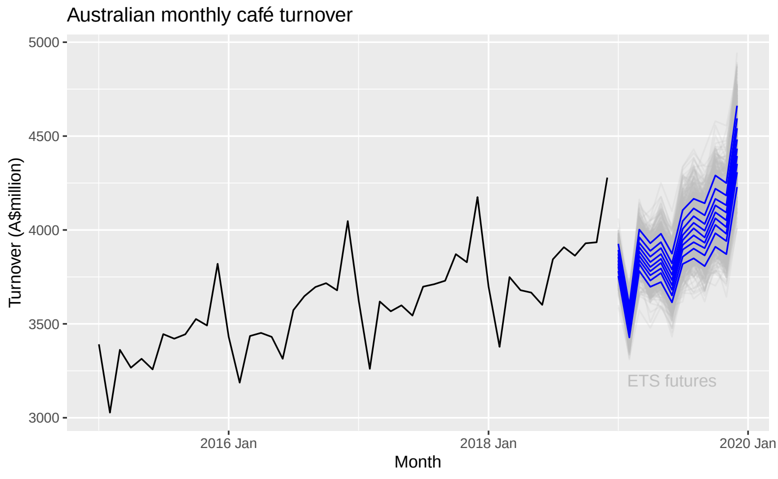


Point forecasts: means of the sample paths.

Prediction intervals: middle 80% of the sample paths at each forecast horizon.

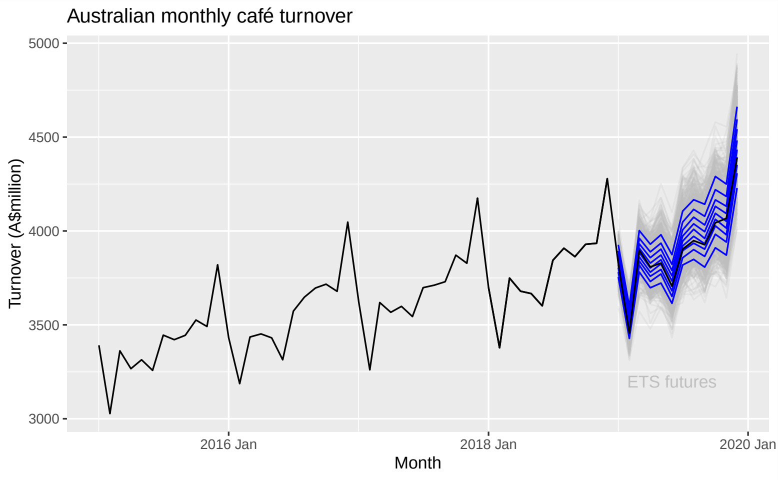
Quantile forecasts: Quantiles of the sample paths at each forecast horizon.

Quantile forecasts



Blue: Deciles for the ETS forecasts for the Australian monthly café turnover.

Quantile forecasts



Blue: Deciles for the ETS forecasts for the Australian monthly café turnover.
Black: Observed values.

Evaluating quantile forecasts

$f_{p,t}$ = quantile forecast with prob. p at time t.

y_t = observation at time t

Evaluating quantile forecasts

$f_{p,t}$ = quantile forecast with prob. p at time t.

y_t = observation at time t

Quantile score

$$Q_{p,t} = \begin{cases} 2(1-p)\left|y_t - f_{p,t}\right|, & \text{if } y_t < f_{p,t} \\ 2p\left|y_t - f_{p,t}\right|, & \text{if } y_t \geq f_{p,t} \end{cases}$$

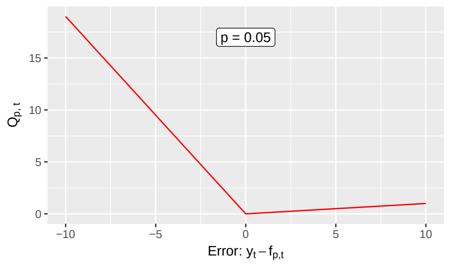
Evaluating quantile forecasts

$f_{p,t}$ = quantile forecast with prob. p at time t.

y_t = observation at time t

Quantile score

$$Q_{p,t} = \begin{cases} 2(1-p)\big|y_t - f_{p,t}\big|, & \text{if } y_t < f_{p,t} \\ 2p\big|y_t - f_{p,t}\big|, & \text{if } y_t \ge f_{p,t} \end{cases}$$



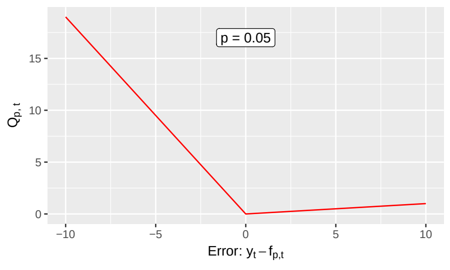
Evaluating quantile forecasts

$f_{p,t}$ = quantile forecast with prob. p at time t.

y_t = observation at time t

Quantile score

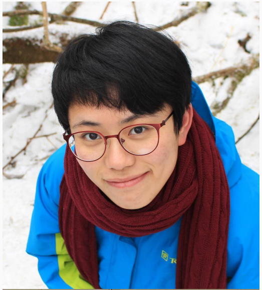
$$Q_{p,t} = \begin{cases} 2(1-p)|y_t - f_{p,t}|, & \text{if } y_t < f_{p,t} \\ 2p|y_t - f_{p,t}|, & \text{if } y_t \geq f_{p,t} \end{cases}$$



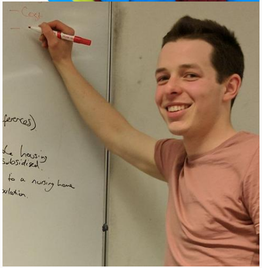
- Low $Q_{p,t}$ is good
- Multiplier of 2 often omitted, but useful for interpretation
- $Q_{p,t}$ like absolute error (weighted to account for likely exceedance)
- Average $Q_{p,t}$ over p = CRPS (Continuous Rank Probability Score)

Let’s do some coding

tidyverts.org



Earo Wang



Mitchell O’Hara-Wild

Evaluating quantile forecasts

```
auscafe
```

```
## # A tsibble: 168 x 2 [1M]
##        date turnover
##       <mth>    <dbl>
##  1 2006 Jan    1914.
##  2 2006 Feb    1750
##  3 2006 Mar    1984.
##  4 2006 Apr    1967.
##  5 2006 May    2005.
##  6 2006 Jun    1944.
##  7 2006 Jul    2019.
##  8 2006 Aug    2043.
##  9 2006 Sep    2039.
## 10 2006 Oct    2113.
## # ... with 158 more rows
```

Evaluating quantile forecasts

```r
auscafe %>%
  filter(year(date) <= 2018)
```

```
## # A tsibble: 156 x 2 [1M]
##          date turnover
##         <mth>    <dbl>
##  1 2006 Jan      1914.
##  2 2006 Feb      1750
##  3 2006 Mar      1984.
##  4 2006 Apr      1967.
##  5 2006 May      2005.
##  6 2006 Jun      1944.
##  7 2006 Jul      2019.
##  8 2006 Aug      2043.
##  9 2006 Sep      2039.
## 10 2006 Oct      2113.
## # ... with 146 more rows
```

Evaluating quantile forecasts

```
auscafe %>%
  filter(year(date) <= 2018) %>%
  model(
    ETS = ETS(turnover),
    ARIMA = ARIMA(turnover ~
                  pdq(d=1) + PDQ(D=1))
  )
```

```
## # A mable: 1 x 2
##             ETS                       ARIMA
##         <model>                     <model>
## 1 <ETS(M,A,M)> <ARIMA(0,1,1)(0,1,1)[12]>
```

Evaluating quantile forecasts

```r
auscafe %>%
  filter(year(date) <= 2018) %>%
  model(
    ETS = ETS(turnover),
    ARIMA = ARIMA(turnover ~
                  pdq(d=1) + PDQ(D=1))
  ) %>%
  forecast(h = "1 year")
```

```
## # A fable: 24 x 4 [1M]
## # Key:      .model [2]
##    .model     date          turnover .mean
##     <chr>    <mth>            <dist> <dbl>
##  1 ETS    2019 Jan  N(3839, 4272) 3839.
##  2 ETS    2019 Feb  N(3514, 4587) 3514.
##  3 ETS    2019 Mar  N(3889, 6892) 3889.
##  4 ETS    2019 Apr  N(3809, 7868) 3809.
##  5 ETS    2019 May  N(3856, 9385) 3856.
##  6 ETS    2019 Jun N(3738, 10098) 3738.
##  7 ETS    2019 Jul N(3951, 12748) 3951.
##  8 ETS    2019 Aug N(4008, 14670) 4008.
##  9 ETS    2019 Sep N(3968, 15941) 3968.
## 10 ETS    2019 Oct N(4100, 18726) 4100.
## # ... with 14 more rows
```

Evaluating quantile forecasts

```r
auscafe %>%
  filter(year(date) <= 2018) %>%
  model(
    ETS = ETS(turnover),
    ARIMA = ARIMA(turnover ~
                    pdq(d=1) + PDQ(D=1))
  ) %>%
  forecast(h = "1 year") %>%
  accuracy(data = auscafe,
    measures = list(crps=CRPS, rmse=RMSE)
  ) %>%
  arrange(crps)
```

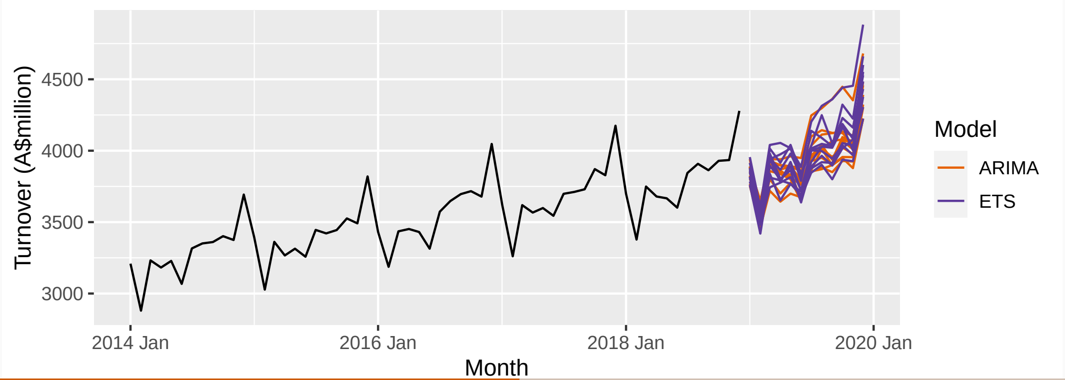
```
## # A tibble: 2 x 4
##   .model .type  crps  rmse
##   <chr>  <chr> <dbl> <dbl>
## 1 ETS    Test   31.3  41.1
## 2 ARIMA  Test   32.9  51.5
```

Outline

Ensemble forecasting

Ensemble forecasting: mix the forecast distributions from multiple models.

- "All models are wrong, but some are useful" (George Box, 1976)
- Allows diverse models to be included, while reducing impact of any specific model.
- Allows uncertainty of model selection to be incorporated.



Ensemble forecasting

```
auscafe %>%
  filter(year(date) <= 2018) %>%
  model(ETS = ETS(turnover),
        ARIMA = ARIMA(turnover ~
              pdq(d=1) + PDQ(D=1))
  ) %>%
  forecast(h = "1 year")
```

```
## # A fable: 24 x 4 [1M]
## # Key:     .model [2]
##    .model     date          turnover .mean
##    <chr>     <mth>            <dist> <dbl>
##  1 ETS    2019 Jan  N(3839, 4272) 3839.
##  2 ETS    2019 Feb  N(3514, 4587) 3514.
##  3 ETS    2019 Mar  N(3889, 6892) 3889.
##  4 ETS    2019 Apr  N(3809, 7868) 3809.
##  5 ETS    2019 May  N(3856, 9385) 3856.
##  6 ETS    2019 Jun N(3738, 10098) 3738.
##  7 ETS    2019 Jul N(3951, 12748) 3951.
##  8 ETS    2019 Aug N(4008, 14670) 4008.
##  9 ETS    2019 Sep N(3968, 15941) 3968.
## 10 ETS    2019 Oct N(4100, 18726) 4100.
## # ... with 14 more rows
```

Ensemble forecasting

```r
auscafe %>%
  filter(year(date) <= 2018) %>%
  model(ETS = ETS(turnover),
        ARIMA = ARIMA(turnover ~
                  pdq(d=1) + PDQ(D=1))
  ) %>%
  forecast(h = "1 year") %>%
  summarise(
    turnover = dist_mixture(
                turnover[1], turnover[2],
                weights=c(0.5,0.5))) %>%
  mutate(.model = "ENSEMBLE")
```

```
## # A fable: 12 x 3 [1M]
##        date     turnover .model
##       <mth>       <dist> <chr>
##  1 2019 Jan mixture(n=2) ENSEMBLE
##  2 2019 Feb mixture(n=2) ENSEMBLE
##  3 2019 Mar mixture(n=2) ENSEMBLE
##  4 2019 Apr mixture(n=2) ENSEMBLE
##  5 2019 May mixture(n=2) ENSEMBLE
##  6 2019 Jun mixture(n=2) ENSEMBLE
##  7 2019 Jul mixture(n=2) ENSEMBLE
##  8 2019 Aug mixture(n=2) ENSEMBLE
##  9 2019 Sep mixture(n=2) ENSEMBLE
## 10 2019 Oct mixture(n=2) ENSEMBLE
## 11 2019 Nov mixture(n=2) ENSEMBLE
## 12 2019 Dec mixture(n=2) ENSEMBLE
```

Ensemble forecasting

```r
auscafe %>%
  filter(year(date) <= 2018) %>%
  model(ETS = ETS(turnover),
        ARIMA = ARIMA(turnover ~
                  pdq(d=1) + PDQ(D=1))
  ) %>%
  forecast(h = "1 year") %>%
  summarise(
    turnover = dist_mixture(
                turnover[1], turnover[2],
                weights=c(0.5,0.5))) %>%
  mutate(.model = "ENSEMBLE") %>%
  as_fable(response = "turnover",
           key = .model,
           distribution = turnover)
```

```
## # A fable: 12 x 3 [1M]
## # Key:     .model [1]
##        date     turnover .model
##       <mth>       <dist> <chr>
##  1 2019 Jan mixture(n=2) ENSEMBLE
##  2 2019 Feb mixture(n=2) ENSEMBLE
##  3 2019 Mar mixture(n=2) ENSEMBLE
##  4 2019 Apr mixture(n=2) ENSEMBLE
##  5 2019 May mixture(n=2) ENSEMBLE
##  6 2019 Jun mixture(n=2) ENSEMBLE
##  7 2019 Jul mixture(n=2) ENSEMBLE
##  8 2019 Aug mixture(n=2) ENSEMBLE
##  9 2019 Sep mixture(n=2) ENSEMBLE
## 10 2019 Oct mixture(n=2) ENSEMBLE
## 11 2019 Nov mixture(n=2) ENSEMBLE
## 12 2019 Dec mixture(n=2) ENSEMBLE
```

Ensemble forecasting

```r
auscafe %>%
  filter(year(date) <= 2018) %>%
  model(ETS = ETS(turnover),
        ARIMA = ARIMA(turnover ~
                      pdq(d=1) + PDQ(D=1))
  ) %>%
  forecast(h = "1 year") %>%
  summarise(
    turnover = dist_mixture(
                turnover[1], turnover[2],
                weights=c(0.5,0.5))) %>%
  mutate(.model = "ENSEMBLE") %>%
  as_fable(response = "turnover",
           key = .model,
           distribution = turnover) %>%
  accuracy(data = auscafe,
    measures = list(crps=CRPS, rmse=RMSE))
```

```
## # A tibble: 1 x 4
##   .model   .type  crps  rmse
##   <chr>    <chr> <dbl> <dbl>
## 1 ENSEMBLE Test   31.7  45.1
```

Ensemble forecasting

```r
auscafe %>%
  filter(year(date) <= 2018) %>%
  model(ETS = ETS(turnover),
        ARIMA = ARIMA(turnover ~
                      pdq(d=1) + PDQ(D=1))
  ) %>%
  forecast(h = "1 year") %>%
  summarise(
    turnover = dist_mixture(
                turnover[1], turnover[2],
                weights=c(0.5,0.5))) %>%
  mutate(.model = "ENSEMBLE") %>%
  as_fable(response = "turnover",
           key = .model,
           distribution = turnover) %>%
  accuracy(data = auscafe,
    measures = list(crps=CRPS, rmse=RMSE))
```

```
## # A tibble: 1 x 4
##   .model   .type  crps  rmse
##   <chr>    <chr> <dbl> <dbl>
## 1 ENSEMBLE Test   31.7  45.1
```

Comparison:

```
## # A tibble: 2 x 4
##   .model .type  crps  rmse
##   <chr>  <chr> <dbl> <dbl>
## 1 ETS    Test   31.3  41.1
## 2 ARIMA  Test   32.9  51.5
```

Outline

Combination forecasting

Combination forecasting: take weighted average of forecasts from multiple models.

- Often a simple average is used.
- Reduces uncertainty associated with selecting a particular model.
- Combination forecasting usually improves point forecast accuracy.
- Mean forecast identical to that from corresponding weighted ensemble.
- Quantile forecasts need to account for correlations between forecast errors from component models.

Combination forecasting

```r
auscafe %>%
  filter(year(date) <= 2018) %>%
  model(
    ETS = ETS(turnover),
    ARIMA = ARIMA(turnover ~
                  pdq(d=1) + PDQ(D=1))
  )
```

```
## # A mable: 1 x 2
##              ETS                        ARIMA
##          <model>                      <model>
## 1 <ETS(M,A,M)> <ARIMA(0,1,1)(0,1,1)[12]>
```

Combination forecasting

```
auscafe %>%
  filter(year(date) <= 2018) %>%
  model(
    ETS = ETS(turnover),
    ARIMA = ARIMA(turnover ~
                  pdq(d=1) + PDQ(D=1))
  ) %>%
  mutate(COMB = (ETS + ARIMA)/2)
```

```
## # A mable: 1 x 3
##              ETS                        ARIMA
##          <model>                      <model>
## 1 <ETS(M,A,M)> <ARIMA(0,1,1)(0,1,1)[12]>
## # ... with 1 more variable:
## #   COMB <model>
```

Combination forecasting

```r
auscafe %>%
  filter(year(date) <= 2018) %>%
  model(
    ETS = ETS(turnover),
    ARIMA = ARIMA(turnover ~
                      pdq(d=1) + PDQ(D=1))
  ) %>%
  mutate(COMB = (ETS + ARIMA)/2) %>%
  forecast(h = "1 year")
```

```
## # A fable: 36 x 4 [1M]
## # Key:      .model [3]
##    .model     date        turnover .mean
##    <chr>     <mth>          <dist> <dbl>
##  1 ETS     2019 Jan  N(3839, 4272) 3839.
##  2 ETS     2019 Feb  N(3514, 4587) 3514.
##  3 ETS     2019 Mar  N(3889, 6892) 3889.
##  4 ETS     2019 Apr  N(3809, 7868) 3809.
##  5 ETS     2019 May  N(3856, 9385) 3856.
##  6 ETS     2019 Jun N(3738, 10098) 3738.
##  7 ETS     2019 Jul N(3951, 12748) 3951.
##  8 ETS     2019 Aug N(4008, 14670) 4008.
##  9 ETS     2019 Sep N(3968, 15941) 3968.
## 10 ETS     2019 Oct N(4100, 18726) 4100.
## # ... with 26 more rows
```

Combination forecasting

```r
auscafe %>%
  filter(year(date) <= 2018) %>%
  model(
    ETS = ETS(turnover),
    ARIMA = ARIMA(turnover ~
                    pdq(d=1) + PDQ(D=1))
  ) %>%
  mutate(COMB = (ETS + ARIMA)/2) %>%
  forecast(h = "1 year") %>%
  accuracy(
    data = auscafe,
    measures = list(crps=CRPS, rmse=RMSE)
  ) %>%
  arrange(crps)
```

```
## # A tibble: 3 x 4
##   .model .type  crps  rmse
##   <chr>  <chr> <dbl> <dbl>
## 1 COMB   Test   30.9  45.1
## 2 ETS    Test   31.3  41.1
## 3 ARIMA  Test   32.9  51.5
```

Combination forecasting

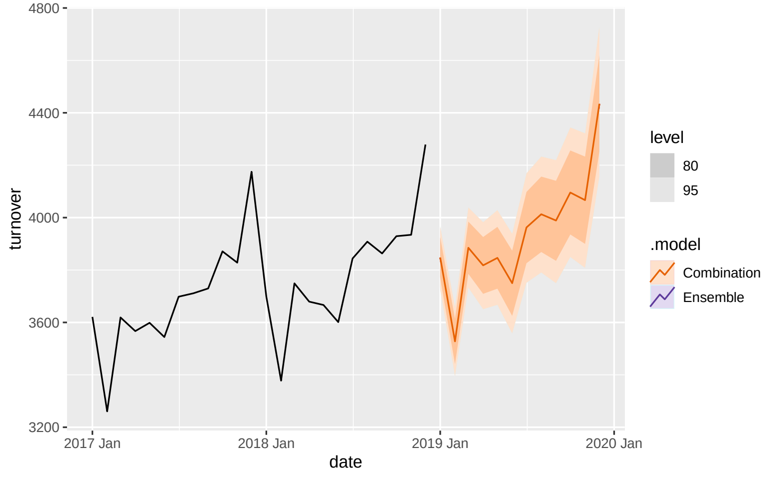
```
auscafe %>%
  filter(year(date) <= 2018) %>%
  model(
    ETS = ETS(turnover),
    ARIMA = ARIMA(turnover ~
                    pdq(d=1) + PDQ(D=1))
  ) %>%
  mutate(COMB = (ETS + ARIMA)/2) %>%
  forecast(h = "1 year") %>%
  accuracy(
    data = auscafe,
    measures = list(crps=CRPS, rmse=RMSE)
  ) %>%
  arrange(crps)
```

```
## # A tibble: 3 x 4
##   .model .type  crps  rmse
##   <chr>  <chr> <dbl> <dbl>
## 1 COMB   Test   30.9  45.1
## 2 ETS    Test   31.3  41.1
## 3 ARIMA  Test   32.9  51.5
```

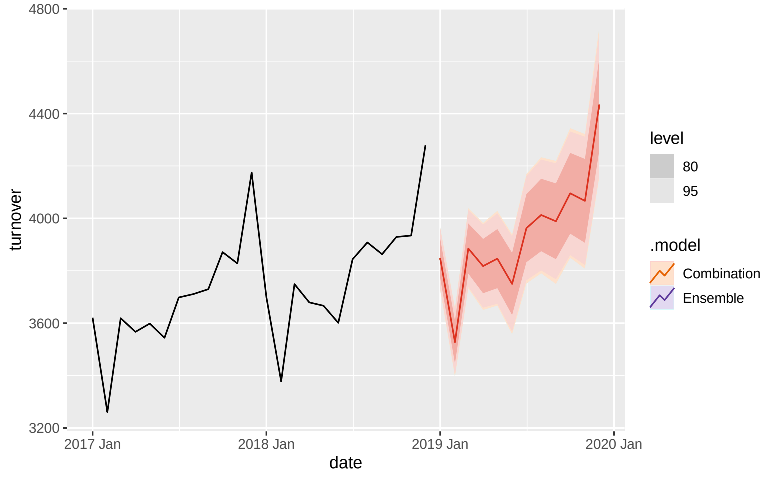
Comparison:

```
## # A tibble: 1 x 4
##   .model   .type  crps  rmse
##   <chr>    <chr> <dbl> <dbl>
## 1 ENSEMBLE Test   31.7  45.1
```

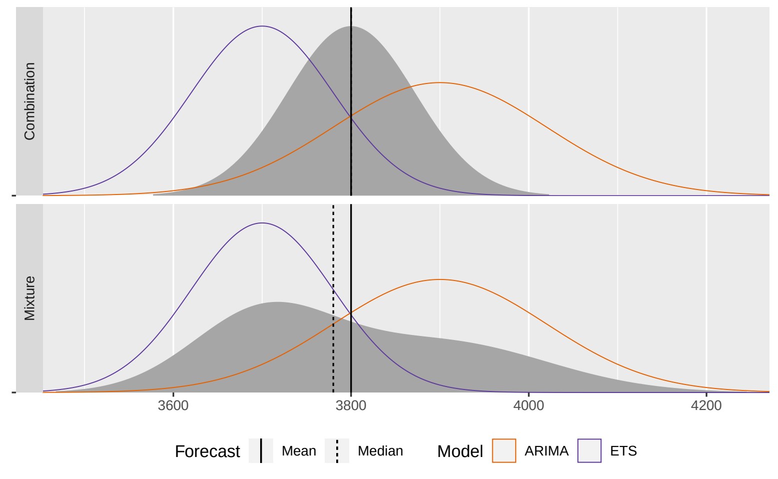
Combination vs ensemble forecasting



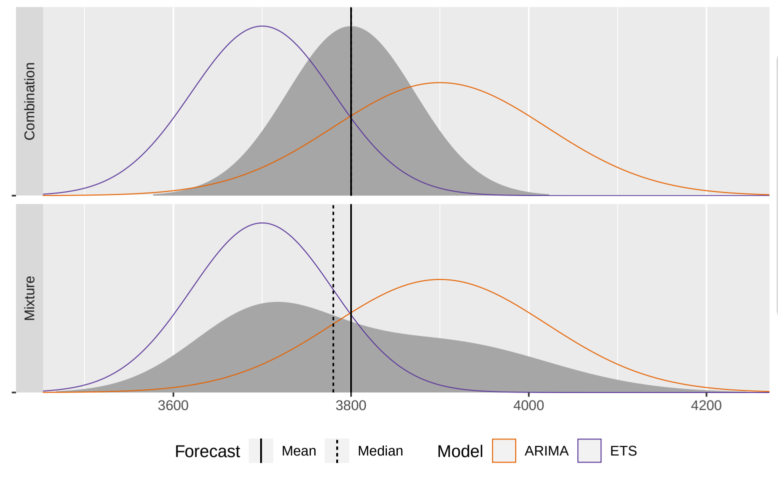
Combination vs ensemble forecasting



Combination vs ensemble forecasting



Combination vs ensemble forecasting



- Combinations involve averaging the distributions.
- Ensembles involve mixing the distributions.
- The means are the same, but other characteristics are different.

Outline

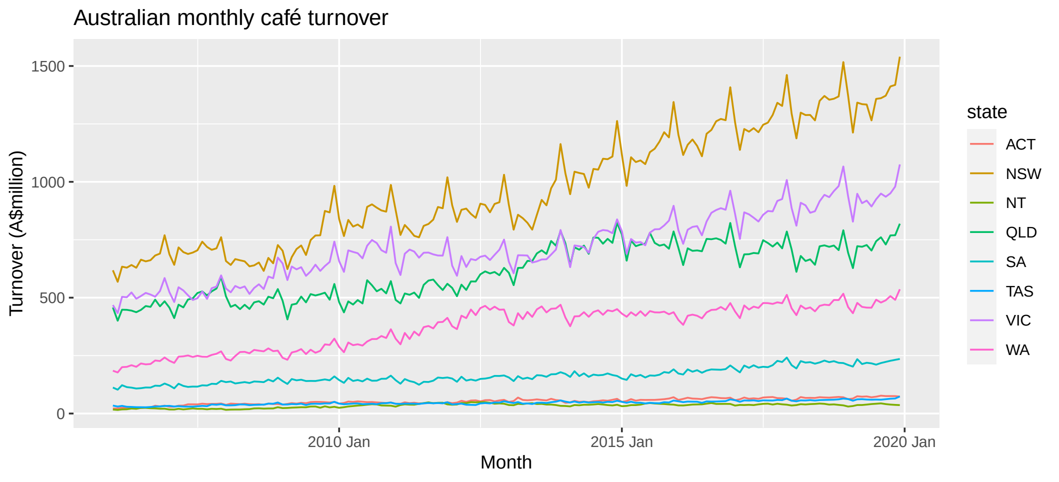
Forecasting many series

```r
cafe
```

```
## # A tsibble: 1,344 x 3 [1M]
## # Key:        state [8]
##          date state turnover
##         <mth> <chr>    <dbl>
##  1 2006 Jan ACT       21.7
##  2 2006 Feb ACT       21.9
##  3 2006 Mar ACT       24.9
##  4 2006 Apr ACT       24.8
##  5 2006 May ACT       27
##  6 2006 Jun ACT       24.5
##  7 2006 Jul ACT       24
##  8 2006 Aug ACT       26.1
##  9 2006 Sep ACT       26.2
## 10 2006 Oct ACT       33.7
## # ... with 1,334 more rows
```

Forecasting many series

```r
cafe %>% autoplot(turnover)
```



Forecasting many series

```r
cafe %>%
  filter(year(date) <= 2018)
```

```
## # A tsibble: 1,248 x 3 [1M]
## # Key:       state [8]
##        date state turnover
##       <mth> <chr>    <dbl>
##  1 2006 Jan ACT       21.7
##  2 2006 Feb ACT       21.9
##  3 2006 Mar ACT       24.9
##  4 2006 Apr ACT       24.8
##  5 2006 May ACT       27
##  6 2006 Jun ACT       24.5
##  7 2006 Jul ACT       24
##  8 2006 Aug ACT       26.1
##  9 2006 Sep ACT       26.2
## 10 2006 Oct ACT       33.7
## # ... with 1,238 more rows
```

Forecasting many series

```
cafe %>%
  filter(year(date) <= 2018) %>%
  model(
    ETS = ETS(turnover),
    ARIMA = ARIMA(turnover ~
                    pdq(d=1) + PDQ(D=1)),
    SNAIVE = SNAIVE(turnover)
  ) %>%
  mutate(
    COMB = (ETS+ARIMA)/2
  )
```

```
## # A mable: 8 x 5
## # Key:     state [8]
##   state              ETS
##   <chr>          <model>
## 1 ACT    <ETS(M,Ad,M)>
## 2 NSW    <ETS(M,Ad,M)>
## 3 NT      <ETS(A,N,A)>
## 4 QLD    <ETS(M,Ad,M)>
## 5 SA     <ETS(M,Ad,M)>
## 6 TAS     <ETS(A,N,A)>
## 7 VIC    <ETS(M,Ad,M)>
## 8 WA     <ETS(M,Ad,M)>
## # ... with 3 more variables:
## #   ARIMA <model>, SNAIVE <model>,
## #   COMB <model>
```

Forecasting many series

```r
cafe %>%
  filter(year(date) <= 2018) %>%
  model(
    ETS = ETS(turnover),
    ARIMA = ARIMA(turnover ~
                    pdq(d=1) + PDQ(D=1)),
    SNAIVE = SNAIVE(turnover)
  ) %>%
  mutate(
    COMB = (ETS+ARIMA)/2
  ) %>%
  forecast(h = "1 year")
```

```
## # A fable: 384 x 5 [1M]
## # Key:     state, .model [32]
##    state .model     date   turnover
##    <chr> <chr>     <mth>     <dist>
##  1 ACT   ETS    2019 Jan N(61, 9.6)
##  2 ACT   ETS    2019 Feb  N(64, 15)
##  3 ACT   ETS    2019 Mar  N(72, 26)
##  4 ACT   ETS    2019 Apr  N(67, 28)
##  5 ACT   ETS    2019 May  N(70, 36)
##  6 ACT   ETS    2019 Jun  N(67, 39)
##  7 ACT   ETS    2019 Jul  N(68, 46)
##  8 ACT   ETS    2019 Aug  N(70, 53)
##  9 ACT   ETS    2019 Sep  N(69, 57)
## 10 ACT   ETS    2019 Oct  N(70, 66)
## # ... with 374 more rows, and 1 more
## #   variable: .mean <dbl>
```

Forecasting many series

```r
cafe %>%
  filter(year(date) <= 2018) %>%
  model(
    ETS = ETS(turnover),
    ARIMA = ARIMA(turnover ~
                    pdq(d=1) + PDQ(D=1)),
    SNAIVE = SNAIVE(turnover)
  ) %>%
  mutate(
    COMB = (ETS+ARIMA)/2
  ) %>%
  forecast(h = "1 year") %>%
  accuracy(
    data = cafe,
    measures = list(crps=CRPS, rmse=RMSE)
  )
```

```
## # A tibble: 32 x 5
##    .model state .type  crps  rmse
##    <chr>  <chr> <chr> <dbl> <dbl>
##  1 ARIMA  ACT   Test   1.64  2.23
##  2 ARIMA  NSW   Test  18.4  28.4
##  3 ARIMA  NT    Test   2.19  3.89
##  4 ARIMA  QLD   Test  15.0  24.9
##  5 ARIMA  SA    Test   4.06  6.70
##  6 ARIMA  TAS   Test   1.52  2.70
##  7 ARIMA  VIC   Test  30.4  48.6
##  8 ARIMA  WA    Test   9.06 14.8
##  9 COMB   ACT   Test   2.02  3.31
## 10 COMB   NSW   Test  17.8  14.8
## # ... with 22 more rows
```

Forecasting many series

```r
cafe %>%
  filter(year(date) <= 2018) %>%
  model(
    ETS = ETS(turnover),
    ARIMA = ARIMA(turnover ~
                  pdq(d=1) + PDQ(D=1)),
    SNAIVE = SNAIVE(turnover)
  ) %>%
  mutate(
    COMB = (ETS+ARIMA)/2
  ) %>%
  forecast(h = "1 year") %>%
  accuracy(
    data = cafe,
    measures = list(crps=CRPS, rmse=RMSE)
  ) ->
  crps
```

```
## # A tibble: 32 x 5
##    .model state .type  crps  rmse
##    <chr>  <chr> <chr> <dbl> <dbl>
##  1 ARIMA  ACT   Test   1.64  2.23
##  2 ARIMA  NSW   Test  18.4  28.4
##  3 ARIMA  NT    Test   2.19  3.89
##  4 ARIMA  QLD   Test  15.0  24.9
##  5 ARIMA  SA    Test   4.06  6.70
##  6 ARIMA  TAS   Test   1.52  2.70
##  7 ARIMA  VIC   Test  30.4  48.6
##  8 ARIMA  WA    Test   9.06 14.8
##  9 COMB   ACT   Test   2.02  3.31
## 10 COMB   NSW   Test  17.8  14.8
## # ... with 22 more rows
```

Forecasting many series

```r
sn_crps <- crps %>%
  filter(.model=="SNAIVE") %>%
  select(state,crps) %>%
  rename(sn_crps = "crps")
sn_crps
```

```
## # A tibble: 8 x 2
##   state sn_crps
##   <chr>   <dbl>
## 1 ACT      3.06
## 2 NSW     27.5
## 3 NT       1.61
## 4 QLD     25.1
## 5 SA       5.16
## 6 TAS      2.71
## 7 VIC     16.8
## 8 WA      10.3
```

Forecasting many series

```r
sn_crps <- crps %>%
  filter(.model=="SNAIVE") %>%
  select(state,crps) %>%
  rename(sn_crps = "crps")
crps %>%
  filter(.model != "SNAIVE") %>%
  left_join(sn_crps, by="state") %>%
  mutate(
    skill = 100*(sn_crps - crps)/sn_crps
  ) %>%
  group_by(.model) %>%
  summarise(
    skill = mean(skill)
  )
```

```
## # A tibble: 3 x 2
##   .model skill
##   <chr>  <dbl>
## 1 ARIMA   9.91
## 2 COMB   14.6
## 3 ETS     6.23
```

More information

- Slides and code: **robjhyndman.com/nycr2020**
- Packages: **tidyverts.org**
- Forecasting textbook using fable package: **OTexts.com/fpp3**

Find me at ...

@robjhyndman

@robjhyndman

robjhyndman.com

rob.hyndman@monash.edu