ETC3550
Applied forecasting for business and economics

Ch12. Some practical forecasting issues

OTexts.org/fpp3/

Outline

Outline

3

Models for different frequencies

Models for annual data

- ETS, ARIMA, Dynamic regression

Models for different frequencies

Models for annual data
- ETS, ARIMA, Dynamic regression

Models for quarterly data
- ETS, ARIMA/SARIMA, Dynamic regression, Dynamic harmonic regression, STL+ETS, STL+ARIMA

Models for different frequencies

Models for annual data
- ETS, ARIMA, Dynamic regression

Models for quarterly data
- ETS, ARIMA/SARIMA, Dynamic regression, Dynamic harmonic regression, STL+ETS, STL+ARIMA

Models for monthly data
- ETS, ARIMA/SARIMA, Dynamic regression, Dynamic harmonic regression, STL+ETS, STL+ARIMA

Models for different frequencies

Models for weekly data

- ARIMA/SARIMA, Dynamic regression, Dynamic harmonic regression, STL+ETS, STL+ARIMA, TBATS

Models for different frequencies

Models for weekly data

- ARIMA/SARIMA, Dynamic regression, Dynamic harmonic regression, STL+ETS, STL+ARIMA, TBATS

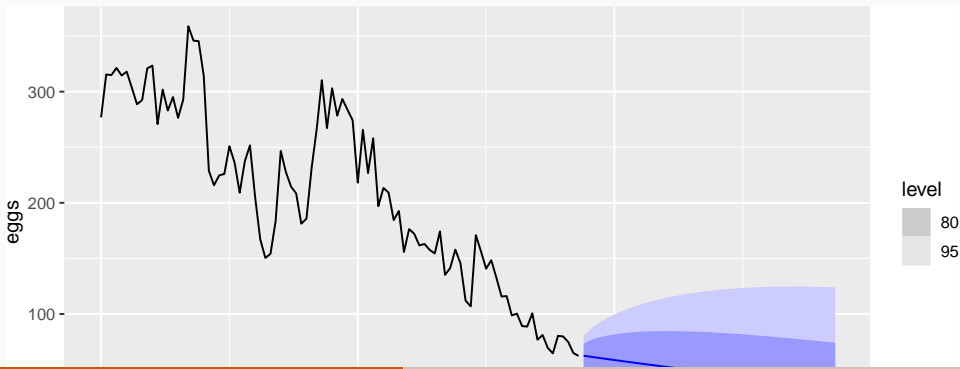## Models for daily, hourly and other sub-daily data

- ARIMA/SARIMA, Dynamic regression, Dynamic harmonic regression, STL+ETS, STL+ARIMA, TBATS

Outline

Positive forecasts

```
recent_prices <- prices %>% filter(!is.na(eggs))
recent_prices %>%
  model(ETS(log(eggs) ~ error("A") + trend("A") + season("N"))) %>%
  forecast(h=50) %>%
  autoplot(recent_prices)
```

Outline

8

Forecast combinations

Clemen (1989)

"The results have been virtually unanimous: combining multiple forecasts leads to increased forecast accuracy. . . . In many cases one can make dramatic performance improvements by simply averaging the forecasts."

Forecast combinations

```r
aus_cafe <- aus_retail %>%
  filter(Industry == "Cafes, restaurants and catering services") %>%
  summarise(Turnover = sum(Turnover))
fc <- aus_cafe %>%
  filter(Month <= yearmonth("2013 Sep")) %>%
  model(
    ETS = ETS(Turnover),
    ARIMA = ARIMA(Turnover)
  ) %>%
  mutate(
    Combination = (ETS + ARIMA)/2
  ) %>%
  forecast(h = "5 years")
```

Forecast combinations

```
fc %>% autoplot(aus_cafe, level = NULL) +
  labs(x = "Year", y = "$ billion",
       title = "Australian monthly expenditure on eating out")
```


Australian monthly expenditure on eating out

11
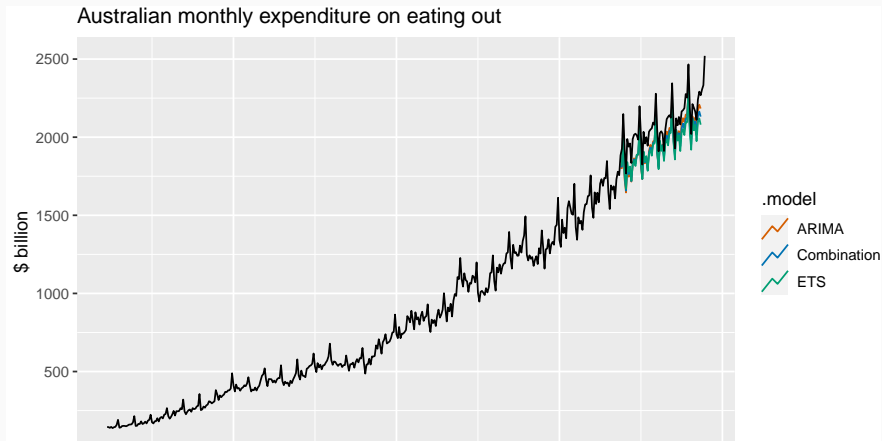
Forecast combinations

```
fc %>% accuracy(aus_cafe)
```

```
## # A tibble: 3 x 10
##   .model       .type    ME  RMSE   MAE   MPE  MAPE  MASE RMSSE  ACF1
##   <chr>        <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 ARIMA        Test   112.  122.  112.  5.44  5.44  1.80  1.50 0.510
## 2 Combination  Test   120.  125.  120.  5.81  5.81  1.93  1.55 0.382
## 3 ETS          Test   128.  133.  128.  6.18  6.18  2.06  1.64 0.324
```

Outline

Missing values

Functions which can handle missing values

- ARIMA()
- TSLM()
- NNETAR()
- VAR()
- FASSTER()

Models which cannot handle missing values

- ETS()
- STL()
- TBATS()

Missing values

Functions which can handle missing values

- ARIMA()
- TSLM()
- NNETAR()
- VAR()
- FASSTER()

Models which cannot handle missing values
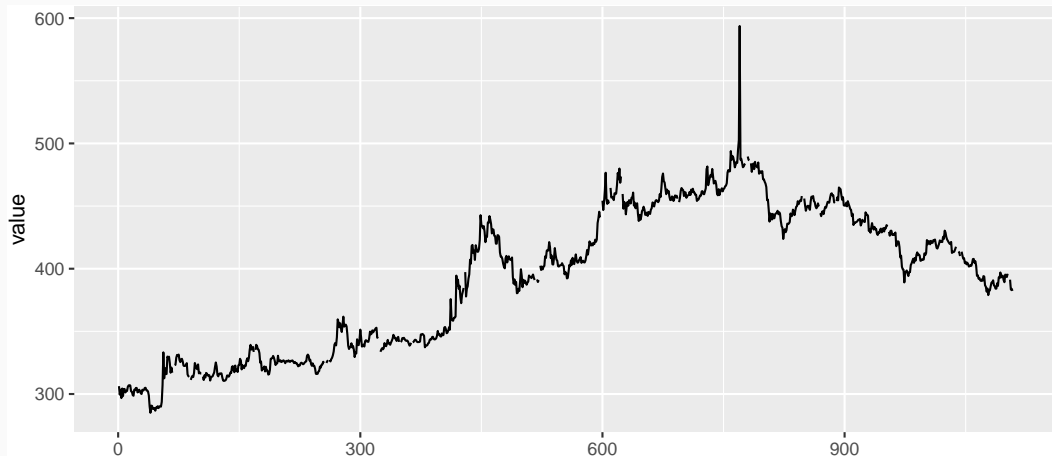
- ETS()
- STL()
- TBATS()

What to do?

1. Model section of data after last missing value.

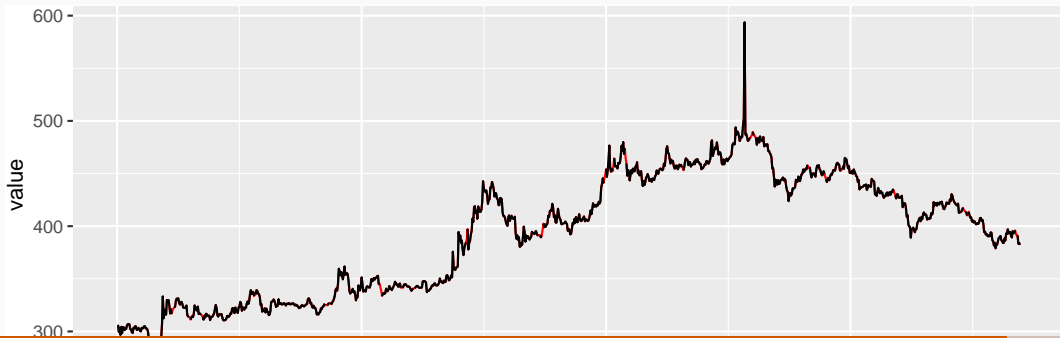# Missing values

```
gold <- as_tsibble(forecast::gold)
gold %>% autoplot(value)
```

Missing values

```
gold_complete <- gold %>%
  model(ARIMA(value)) %>%
  interpolate(gold)
gold_complete %>%
  autoplot(value, colour = "red") +
  autolayer(gold, value)
```

Outline

Outliers

```
fit <- gold %>%
  model(ARIMA(value))
augment(fit) %>%
  mutate(stdres = .resid/sd(.resid, na.rm=TRUE)) %>%
  filter(abs(stdres) > 10)


## # A tsibble: 2 x 7 [1]
## # Key:        .model [1]
##    .model        index value .fitted .resid .innov stdres
##    <chr>         <dbl> <dbl>   <dbl>  <dbl>  <dbl>  <dbl>
## 1 ARIMA(value)    770  594.    499.   94.7   94.7   16.4
## 2 ARIMA(value)    771  487.    562.  -74.8  -74.8  -12.9
```