MONASH University

MONASH BUSINESS SCHOOL

ETF3231/5231: Business forecasting

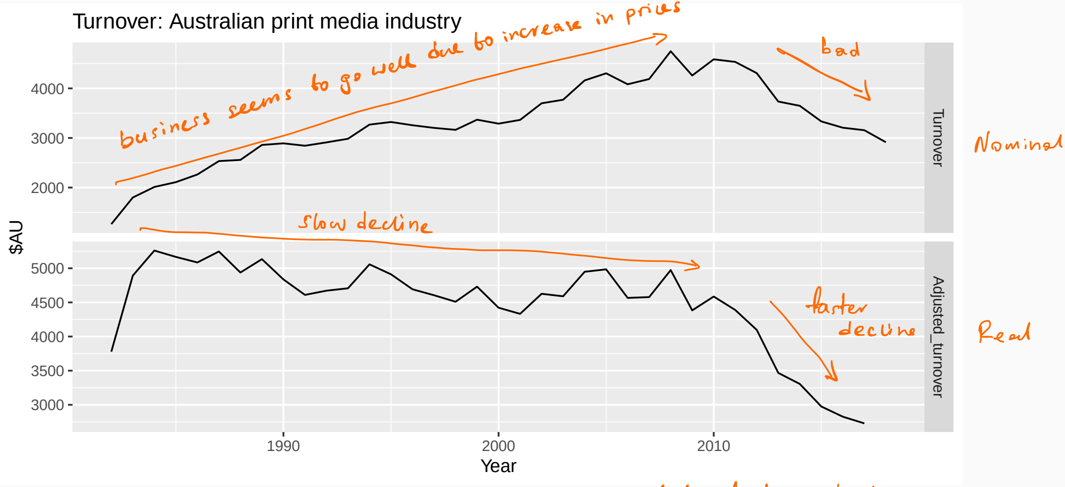
Week 3: Time series decomposition

`https://bf.numbat.space/`

Outline

Outline

Inflation adjustments



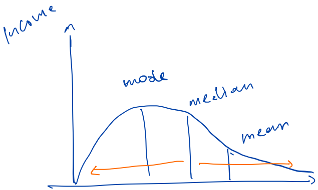
The real story: print media is going down relative to the rest of the economy

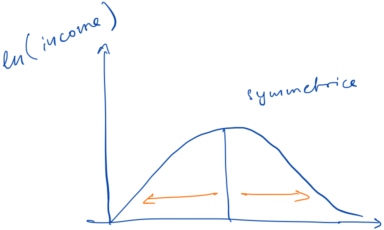
* Must return forecasts to the original scale (NO NEED FOR ASSIGNMENT)

Mathematical transformations

If the data show different variation at different levels of the series, then a transformation can be useful.

Food v Income





Challenge: back-transforming (returns the median - we need an adjustment to get the mean).

Mathematical transformations

If the data show different variation at different levels of the series, then a transformation can be useful.

Denote original observations as $y_1, \dots, y_T$ and transformed observations as $w_1, \dots, w_T$.

Mathematical transformations for stabilizing variation

Square root	$w_t = \sqrt{y_t}$	$\downarrow$
Cube root	$w_t = \sqrt[3]{y_t}$	Increasing
Logarithm	$w_t = \log(y_t)$	strength

Logarithms, in particular, are useful because they are more interpretable: changes in a log value are relative (percent) changes on the original scale.

y_1
y_2
y_3
$\vdots$
y_{t-1}
y_t
y_{t+1}
$\vdots$
y_{T-1}
y_T

$$\% \Delta y_t = \left(\frac{y_t - y_{t-1}}{y_{t-1}} \right) \times 100$$

$$\simeq \ln \left(y_t / y_{t-1} \right) \times 100$$

$$= \left(\ln(y_t) - \ln(y_{t-1}) \right) \times 100$$

- Taking the differences of the logs is useful in this way.
  i.e. shows percentage changes

Two effects : 1. stabilises variance
2. percentage changes

Box-Cox transformations

Each of these transformations is close to a member of the family of

Died 2013 "All models are wrong" later extended to "Some are useful".

Box-Cox transformations:

Designed so that transformation is continuous in λ

$$w_t = \begin{cases} \log(y_t), & \lambda = 0; \\ (\underline{\text{sign}}(y_t)|y_t|^\lambda - 1)/\lambda, & \lambda \neq 0. \end{cases}$$

modification of the original to allow for -ve values of y provided that $\lambda > 0$ by Bickel & Doksum

- Actually the Bickel-Doksum transformation (allowing for $y_t < 0$)
- $\lambda = 1$: (No substantive transformation)
- $\lambda = \frac{1}{2}$: (Square root plus linear transformation)
- $\lambda = 0$: (Natural logarithm)
- $\lambda = -1$: (Inverse plus 1)

Transformations

- Often no transformation needed.
- Simple transformations are easier to explain and work well enough.
- Transformations can have very large effect on PI. upper limit can be extremely large
- If some data are zero or negative, then use $\lambda > 0$.
- `log1p()` can also be useful for data with zeros. the trick is to add 1 to each observation, recall $\log(1) = 0$
- Choosing logs is a simple way to force forecasts to be positive Extremely useful if projecting down
- Transformations must be reversed to obtain forecasts on the original scale. (Handled automatically by `fable`.) more on this in the next chapter

Transformations

We often use logs.

- Often no transformation needed.
- Simple transformations are easier to explain and work well enough.
- Transformations can have very large effect on PI.
- If some data are zero or negative, then use $\lambda > 0$.
- `log1p()` can also be useful for data with zeros.
- Choosing logs is a simple way to force forecasts to be positive
- Transformations must be reversed to obtain forecasts on the original scale. (Handled automatically by `fable`.)

Outline

Time series patterns

Recall

Trend pattern exists when there is a long-term increase or decrease in the data.

Cyclic pattern exists when data exhibit rises and falls that are *not of fixed period* (duration usually of at least 2 years).

Seasonal pattern exists when a series is influenced by seasonal factors (e.g., the quarter of the year, the month, or day of the week).

Time series decomposition

$$y_t = f(S_t, T_t, R_t)$$

where
$y_t =$ data at period t
$T_t =$ trend-cycle component at period t
$S_t =$ seasonal component at period t
$R_t =$ remainder component at period t

Additive decomposition: $y_t = S_t + T_t + R_t.$

Multiplicative decomposition: $y_t = S_t \times T_t \times R_t.$

Time series decomposition

- Additive model appropriate if magnitude of seasonal fluctuations does not vary with level.
- If seasonal are proportional to level of series, then multiplicative model appropriate.
- Multiplicative decomposition more prevalent with economic series ABS uses this almost exclusively
- Alternative: use a Box-Cox transformation, and then use additive decomposition. * This is what we will do.
- Logs turn multiplicative relationship into an additive relationship:

$$y_t = S_t \times T_t \times R_t \quad \Rightarrow \quad \log y_t = \log S_t + \log T_t + \log R_t.$$

Question: when do we observe an additive V multiplicative?

US Retail Employment

```
us_retail_employment <- us_employment %>%
  filter(year(Month) >= 1990, Title == "Retail Trade") %>%
  select(-Series_ID)
us_retail_employment
```

```
## # A tsibble: 357 x 3 [1M]
##        Month Title        Employed
##        <mth> <chr>           <dbl>
##  1 1990 Jan Retail Trade   13256.
##  2 1990 Feb Retail Trade   12966.
##  3 1990 Mar Retail Trade   12938.
##  4 1990 Apr Retail Trade   13012.
##  5 1990 May Retail Trade   13108.
##  6 1990 Jun Retail Trade   13183.
##  7 1990 Jul Retail Trade   13170.
##  8 1990 Aug Retail Trade   13160.
##  9 1990 Sep Retail Trade   13113.
## 10 1990 Oct Retail Trade   13185.
```

Switch to R and discuss the series.

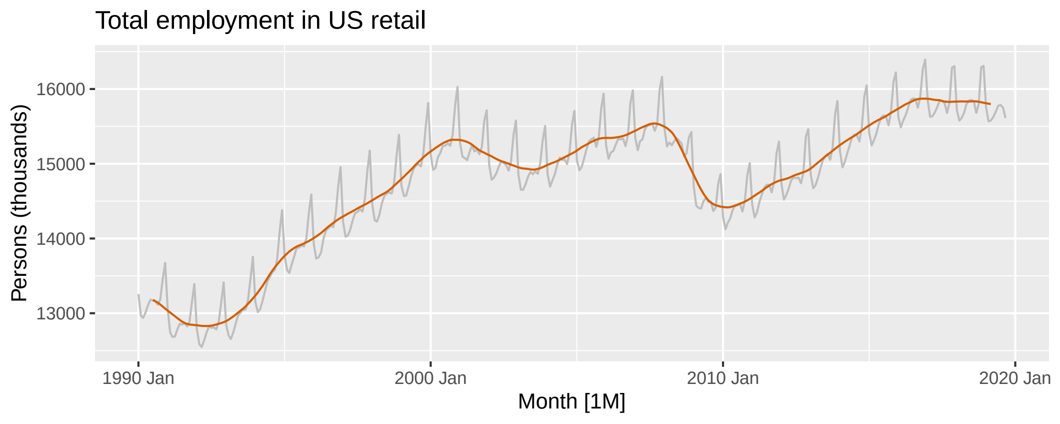
Outline

- Quarterly 2 x 4 MA

$$\hat{T}_t = \frac{1}{8} y_{t-2} + \frac{1}{4} y_{t-1} + \frac{1}{4} y_t + \frac{1}{4} y_{t+1} + \frac{1}{8} y_{t+2}$$

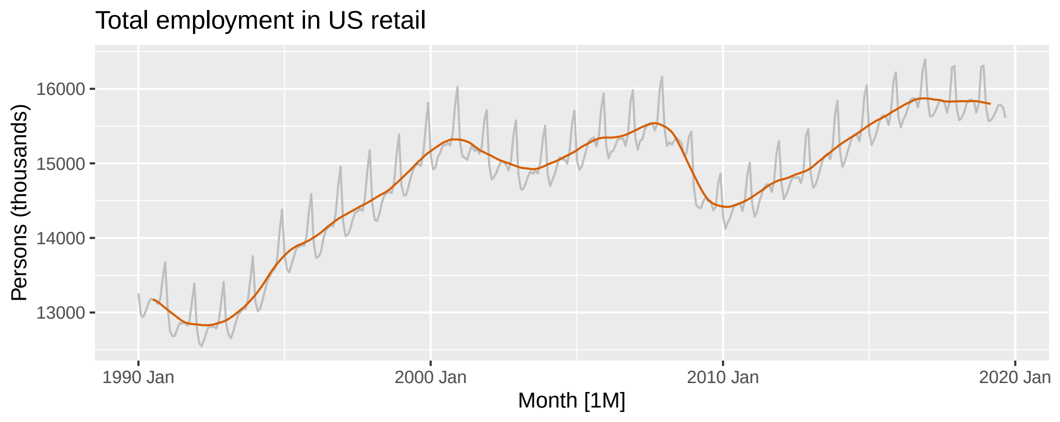
- Monthly 2 x 12 MA

Moving average trend-cycle



What are the problems here?

Moving average trend-cycle



What are the problems here?

- no end points
- MAs sensitive to outliers

Outline

1. Estimate $\hat{T}_t$ using MAs
2. De-trended
   Additive : $y_t - \hat{T}_t = \hat{S}_t + \hat{R}_t$
   Multi : $y_t / \hat{T}_t = \hat{S}_t \times R_t$
3. Estimate S_t by taking averages of successive seasons, e.g., same quarter, and adjust
   $$S^{(1)} + S^{(2)} + \dots + S^{(m)} = 0$$
   $$S^{(1)} + S^{(2)} + \dots + S^{(m)} = 1$$
4. Additive: $\hat{R}_t = y_t - \hat{T}_t - \hat{S}_t$
   Multi : $\hat{R}_t = y_t / (\hat{T}_t \hat{S}_t)$

Additive classical decomposition

y_t

$=$

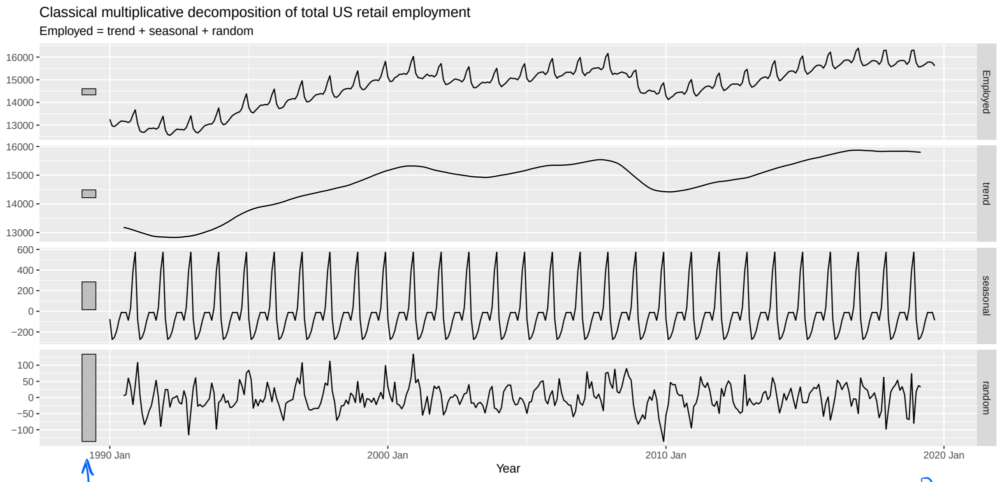
T_t

$+$

S_t

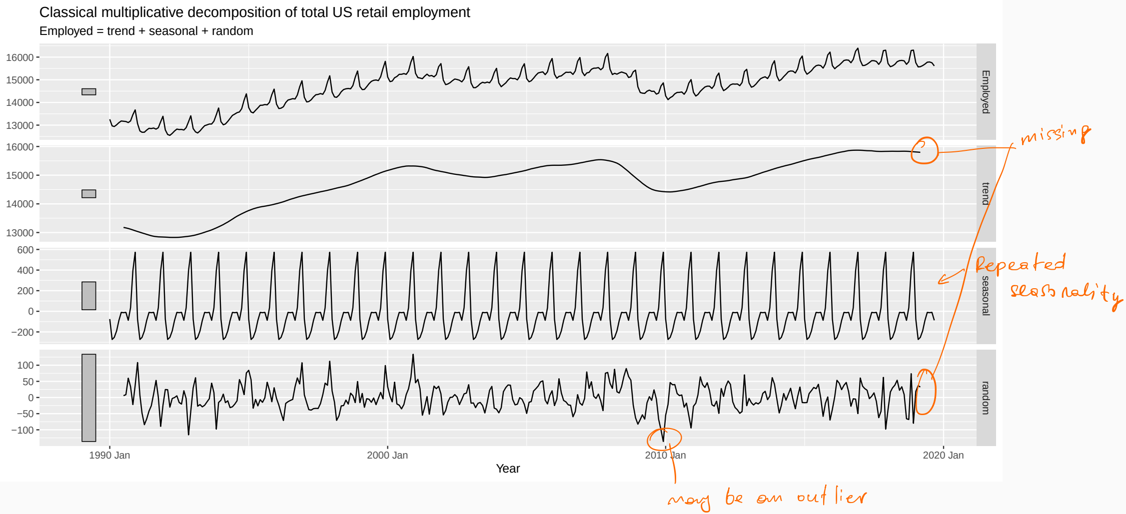
$+$

R_t



Problems ?

Additive classical decomposition



Multiplicative classical decomposition

y_t

=

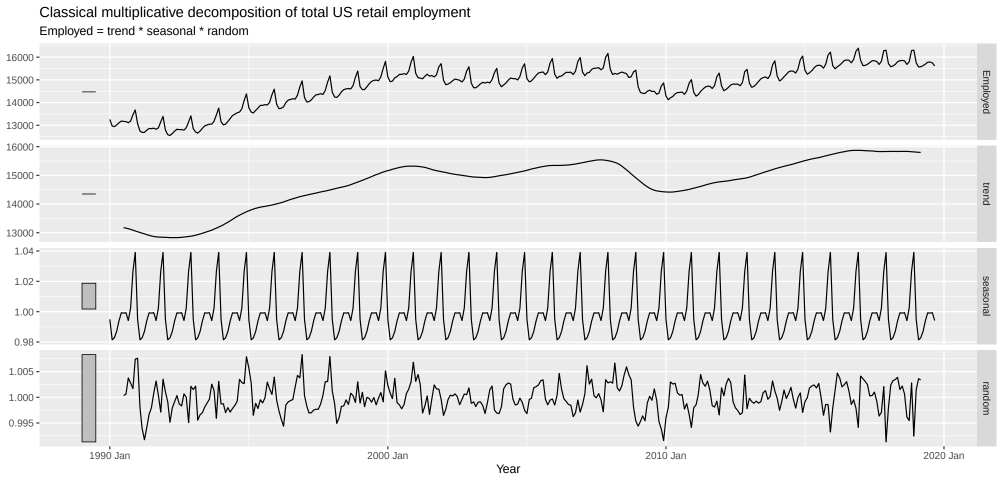
T_t

×

S_t

×

R_t



← how much the trend is inflated by as a ratio.

Comments on classical decomposition

- Estimate of trend is unavailable for first few and last few observations.
- Seasonal component repeats from year to year. May not be realistic.
- Not robust to outliers.
- Newer methods designed to overcome these problems.

Outline

History of time series decomposition

- Classical method originated in 1920s.
- Census II method introduced in 1957. Basis for X-11 method and variants (including X-12-ARIMA, X-13-ARIMA)

  around for a while

  last 10 years

  last 5 years

  → forecast forward & backwards
- STL method introduced in 1983 → Not used by any statistical agency. Developed at Bell Labs (NJ). Not as developed as others.
- TRAMO/SEATS introduced in 1990s.

National Statistics Offices

- ABS uses X-12-ARIMA
- US Census Bureau uses X-13ARIMA-SEATS
- Statistics Canada uses X-12-ARIMA
- ONS (UK) uses X-12-ARIMA
- EuroStat use X-13ARIMA-SEATS

Outline

STL decomposition

Local rergession

- STL: "Seasonal and Trend decomposition using Loess"
- Very versatile and robust.

have used it for weekly data in forecasting comps

- Unlike X-12-ARIMA, STL will handle any type of seasonality.

A bit of subjectivity

- Seasonal component allowed to change over time, and rate of change controlled by user.
- Smoothness of trend-cycle also controlled by user.
- Robust to outliers

Disadvantages limitations

- No trading day or calendar adjustments.
- Only additive.
- Take logs to get multiplicative decomposition.
- Use Box-Cox transformations to get other decompositions.

STL decomposition

```
us_retail_employment %>%
  model(STL(Employed)) %>%
  components()
```

how many consecutive obs. to be used to estimate trend

- `trend(window = ?)` controls wiggliness of trend component.

how many consecutive years to estimate seas. comp.

- `season(window = ?)` controls variation on seasonal component.
- `season(window = 'periodic')` is equivalent to an infinite window.

Default setting (often these work very well – no need to change them)

- Season `window = 13`
- Trend `window = nextodd(ceiling((1.5*period)/(1-(1.5/s.window)))`
- Robust `robust=FALSE`