Applied Economic Forecasting

5. Time Series Decomposition

Section 1

Time series components

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). It tends to repeat itself at a fixed period.
- **Irregular** pattern consists of unpredictable or random fluctuations.

Time series patterns

We can decompose a time series into several components:

1. Trend component represents the underlying growth (or decline) in a time series. The trend may be produced, for example, by consistent population change, inflation, technological change, and productivity increases. The trend component is denoted by T.
2. Cyclical component a series of wavelike fluctuations or cycles of more than one year's duration. Changing economic conditions generally produce cycles.

- In practice, cycles are often difficult to identify and are frequently regarded as part of the trend. In this case, the underlying general growth (or decline) component is called the trend-cycle.

3. Seasonal component typically found in quarterly, monthly, or weekly data. Seasonal variation refers to a more or less stable pattern of change that appears annually and repeats itself year after year. Seasonal patterns occur because of the influence of the weather or because of calendar-related events such as school vacations and national holidays.

4. Remainder (Irregular) component consists of unpredictable or random fluctuations. These fluctuations are the result of a myriad of events that individually may not be particularly important but whose combined effect could be large.

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(Irregular) component at period

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

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

Mixed components model: $Y_t = T_t \times S_t + I_t$

Time series decomposition

Additive vs Multiplicative model

Additive

- Appropriate if magnitude of seasonal fluctuations does not vary with level.

Multiplicative

- Appropriate if seasonal are proportional to level of series.
- Multiplicative decomposition more prevalent with economic series

Alternative

We could use a Box-Cox transformation, and then use additive decomposition.

- Logs turn multiplicative relationship into an additive relationship:

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

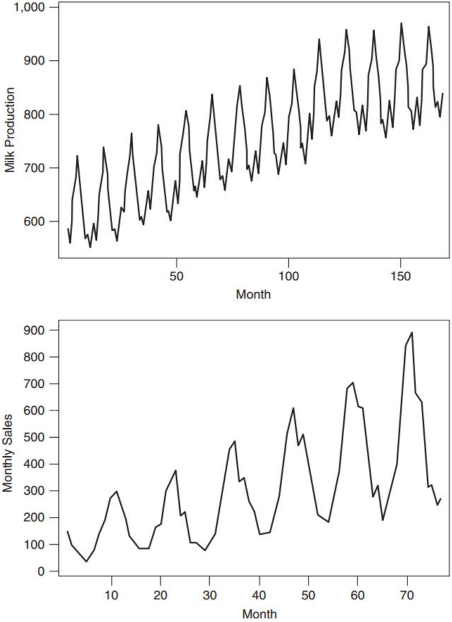


FIGURE 5-1 Time Series with Constant Variability (Top) and a Time Series with Variability Increasing with Level (Bottom)

Time Trend

- Trends are long-term movements in a time series.
- Trend can be linear or nonlinear
- If the trend is roughly linear, like a straight line, then

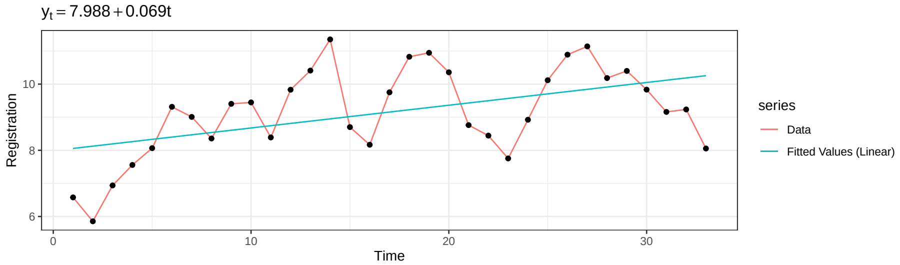
$$\hat{T}_t = b_0 + b_1 t \tag{1}$$

where $\hat{T}$ is the predicted value for the trend at time t

t represents time taking $t = 1, 2, 3, ..., T$

b_1 is the average decrease (increase) in T over each period in time

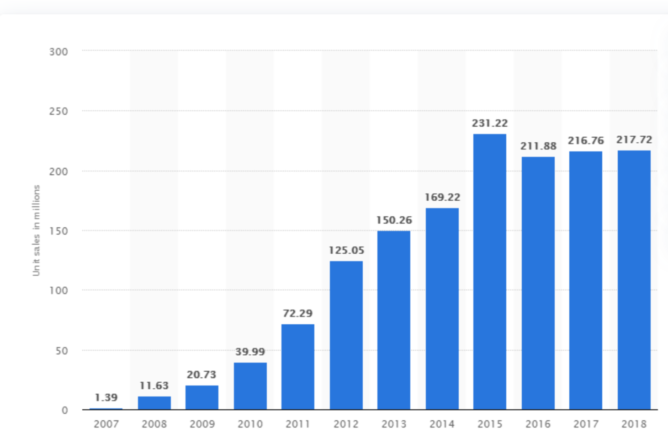
```
register <- ts(readxl::read_xls("register.xls"))
fitlinear <- tslm(register~trend)
fit.line <- fitted(fitlinear)
coef.fit1 <- round(coef(fitlinear),3)
autoplot(register, series = "Data") + autolayer(fit.line,
  series = "Fitted Values (Linear)") + geom_point() +
  labs(y="Registration", title = bquote(y[t] == .(coef.fit1[1]) +
        .(coef.fit1[2]) * t))
```



Additional Trend Curves

What if the trend is not linear?

The graph below presents an example of a non-linear trend in terms of Apple iPhone sales.

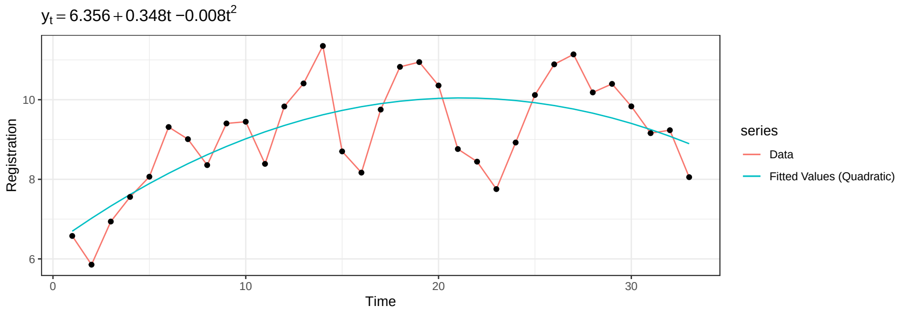


Quadratic Trend

- Consider nonlinear trend functions, such as polynomial, exponential, etc.
- For instance, a quadratic trend (polynomial of order 2):

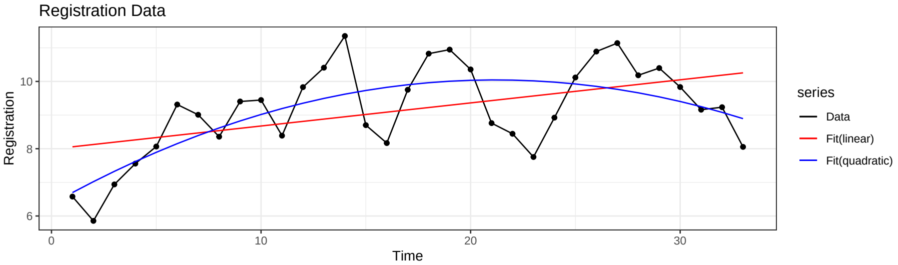
$$\widehat{\mathrm{T}}_t = b_0 + b_1 t + b_t t^2$$

```
fitquad <- tslm(register ~ trend + I(trend^2))
fit.quad <- fitted(fitquad)
coef.fit2 <- round(coef(fitquad),3)
autoplot(register, series = "Data") + autolayer(fit.quad,
    series = "Fitted Values (Quadratic)") + geom_point() +
   labs(y="Registration",
        title = bquote(y[t] == .(coef.fit2[1]) +
            .(coef.fit2[2]) * t ~ .(coef.fit2[3]) *t^2))
```



Trend Curve

Which Trend model will be best for forecasting?



	MAE	RMSE	MAPE
Linear	0.990	1.157	11.274
Quadratic	0.774	0.945	8.617

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)
- STL method introduced in 1983
- TRAMO/SEATS introduced in 1990s.

National Statistics Offices

- X-12-ARIMA:
  - Created by the U.S. Census Bureau
  - ABS
  - Statistics Canada
  - ONS (UK)
  - U.S. Bureau of Labor Statistics
- X-13-ARIMA-SEATS:
  - US Census Bureau
  - EuroStat

Classical Decomposition

- There are two forms of classical decomposition:
  - an additive decomposition and
  - a multiplicative decomposition.
- In the classical decomposition, we assume that the sesonal component is constant from year to year.
- For multiplicative seasonality, the m values that form the seasonal component are sometimes called the "seasonal indices."

See notebook and In-class codes here!

Drawbacks of the classical decomposition

1. The estimate of the trend-cycle is unavailable for the first few and last few observations. Consequently, there is also no estimate of the remainder component for the same time periods.
2. The trend-cycle estimate tends to over-smooth rapid rises and falls in the data.
3. Assume that the seasonal component repeats from year to year. For many series, this is a reasonable assumption, but for some longer series it is not.
4. Occasionally, the values of the time series in a small number of periods may be particularly unusual (think outliers). The classical method is not robust to these kinds of unusual values.

Section 2

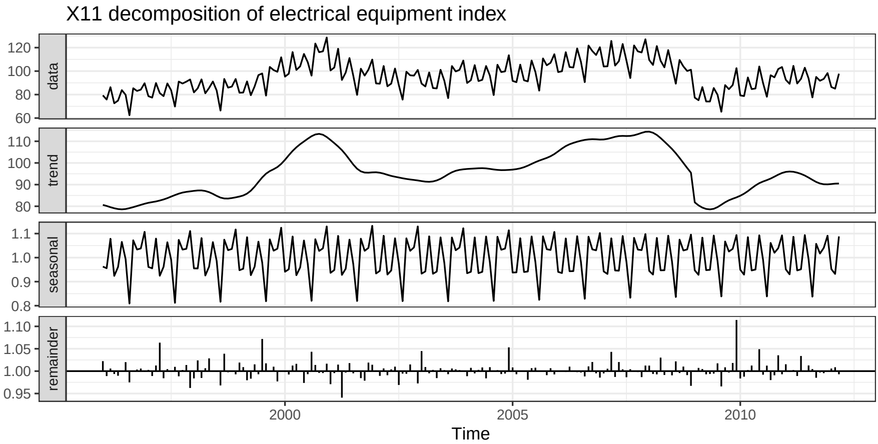
X-11 decomposition

X-11 decomposition

- This method is based on classical decomposition, but includes many extra steps and features to overcome the drawbacks of classical decomposition (See Slide 18).
  - Trend-cycle estimates are available for all observations including the end points.
  - Seasonal component is allowed to vary slowly over time.
- X11 also includes methods for handling trading day variation, holiday effects and the effects of known predictors.
- The process is entirely automatic and tends to be highly robust to outliers and level shifts in the time series.

X-11 decomposition

```
fit <- seasonal::seas(elecequip, x11="")
autoplot(fit) +
  ggtitle("X11 decomposition of electrical equipment index")
```



X-11 decomposition

The `seas` function (from the `seasonal` package) has a few useful functions that allows us to extract specific components of the decomposition. These can help with visualizing aspects of the data.

Helper functions

- `seasonal()` extracts the seasonal component
- `trendcycle()` extracts the trend-cycle component
- `remainder()` extracts the remainder component.
- `seasadj()` returns the seasonally adjusted series.

Section 3

Quick Digression

Seasonal adjustment

You would recall from our classical decomposition earlier that we have two easy ways of getting the seasonally adjusted data.

1. Additive decomposition: seasonally adjusted data given by

$$y_t - S_t = T_t + R_t$$

2. Multiplicative decomposition: seasonally adjusted data given by

$$y_t / S_t = T_t \times R_t$$

Focusing on the Additive Model

- We use estimates of S based on past values to seasonally adjust a current value.
- Seasonally adjusted series reflect **remainders** as well as **trend**.

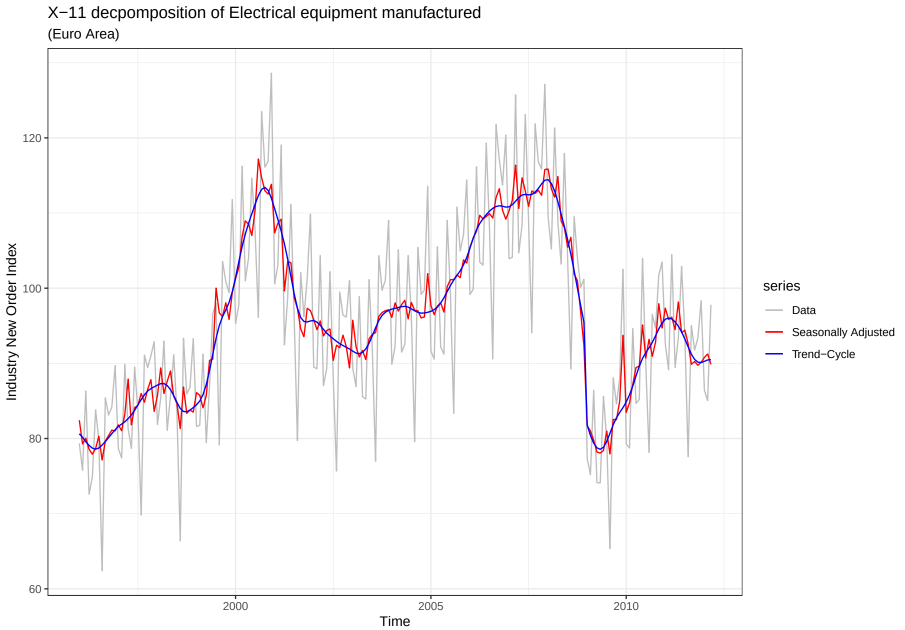
$$Y_t - S_t = T_t + R_t$$

- It is better to use the trend-cycle component to look for turning points.

X-11 decomposition

For example, we can plot the `elecequip` data earlier against its trendcycle and seasonally adjusted components.

```
autoplot(elecequip, series = "Data") +
  autolayer(seasadj(fit), series = "Seasonally Adjusted") +
  autolayer(trendcycle(fit), series="Trend-Cycle") +
  labs(title = "X-11 decpomposition of Electrical
       equipment manufactured",
  subtitle = "(Euro Area)",
  y = "Industry New Order Index") +
  scale_color_manual(values = c("grey","red","blue"))
```


X−11 decpomposition of Electrical equipment manufactured
(Euro Area)
Industry New Order Index
120
100
80
60
2000
2005
2010
Time
series
Data
Seasonally Adjusted
Trend−Cycle

(Dis)advantages of X-11

Advantages

- Relatively robust to outliers
- Completely automated choices for trend and seasonal changes
- Very widely tested on economic data over a long period of time.

Disadvantages

- No prediction/confidence intervals
- Ad hoc method with no underlying model
- Only developed for quarterly and monthly data

Extensions: X-12-ARIMA and X-13-ARIMA

- The X-11, X-12-ARIMA and X-13-ARIMA methods are based on Census decomposition.
- These allow adjustments for trading days and other explanatory variables.
- Known outliers can be omitted.
- Level shifts and ramp effects can be modelled.
- Missing values estimated and replaced.
- Holiday factors (e.g., Easter, Labour Day) can be estimated.

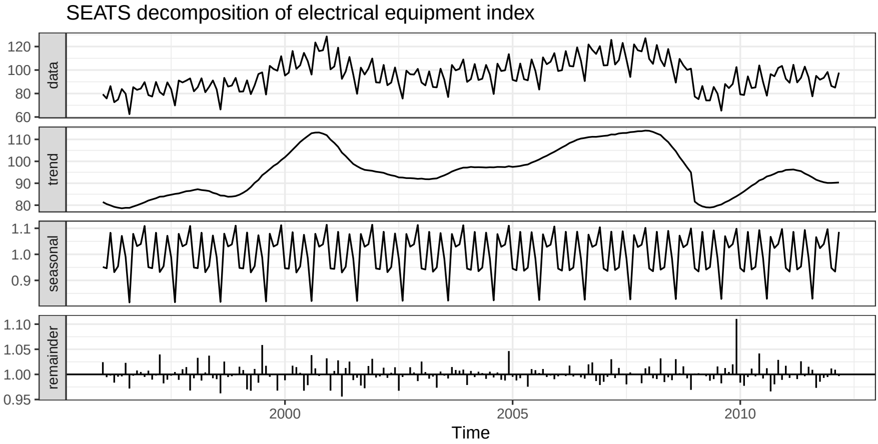
Section 4

Seasonal Extraction in ARIMA Time Series (SEATS) decomposition

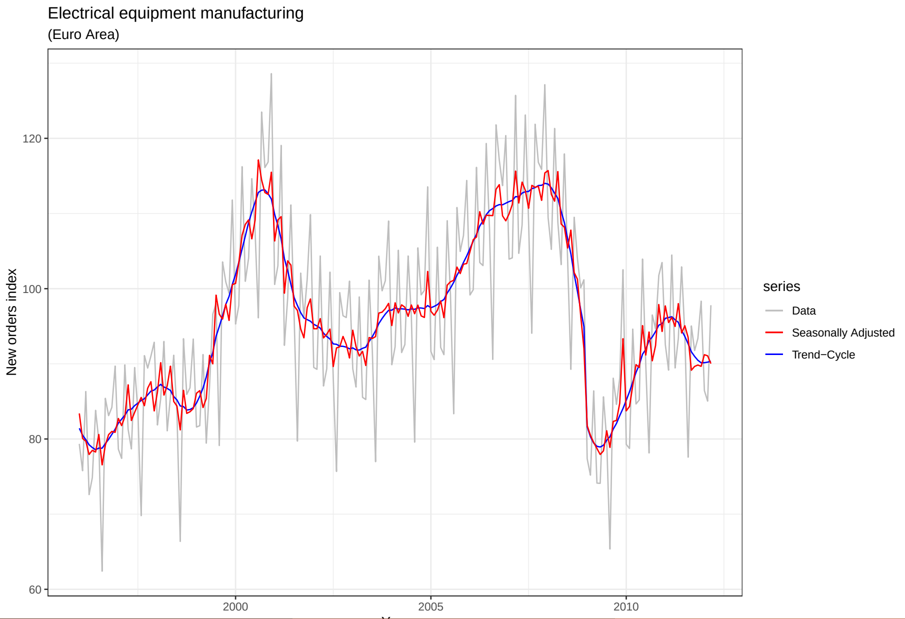
SEATS decomposition

- Developed at the Bank of Spain, and is now widely used by government agencies around the world.
- The procedure works **only with quarterly and monthly data**.
  - So seasonality of other kinds, such as daily data, or hourly data, or weekly data, require an alternative approach.

SEATS decomposition



SEATS decomposition



(Dis)advantages of SEATS

Advantages

- Model-based
- Smooth trend estimate
- Allows estimates at end points
- Allows changing seasonality
- Developed for economic data

Disadvantages

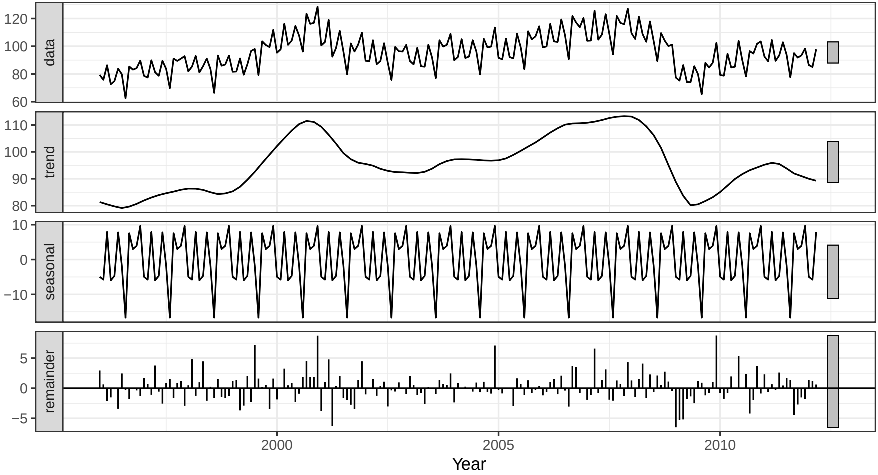
- Only developed for quarterly and monthly data

Section 5

Seasonal and Trend decomposition using Loess (STL) Method

STL decomposition

```
fit <- elecequip %>% stl(t.window=13, s.window="periodic",
           robust=TRUE)
autoplot(fit) + xlab("Year")
```



STL decomposition

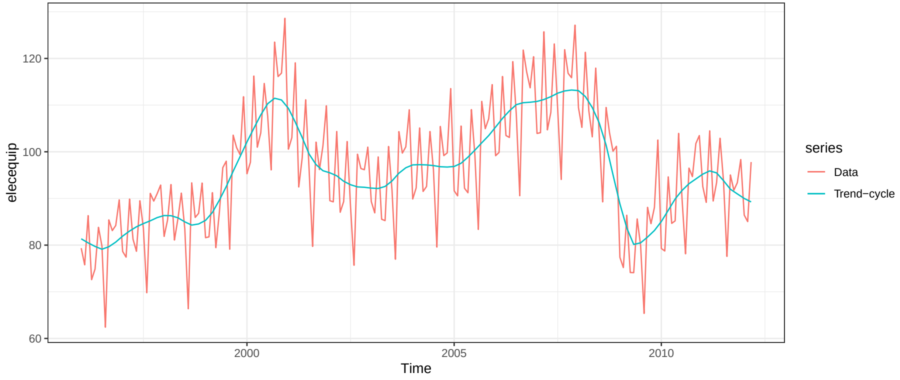
```
stl(elecequip,s.window=5)

stl(elecequip, t.window=15,
  s.window="periodic", robust=TRUE)
```

Quick Notes

- `t.window` and `s.window` control how rapidly the trend-cycle and seasonal components can change.
- Both should be odd numbers:
  - `t.window` is the number of consecutive observations to be used when estimating the trend-cycle.
  - `s.window` is the number of consecutive years to be used in estimating each value in the seasonal component.
- Smaller values allow for more rapid changes.

Euro electrical equipment

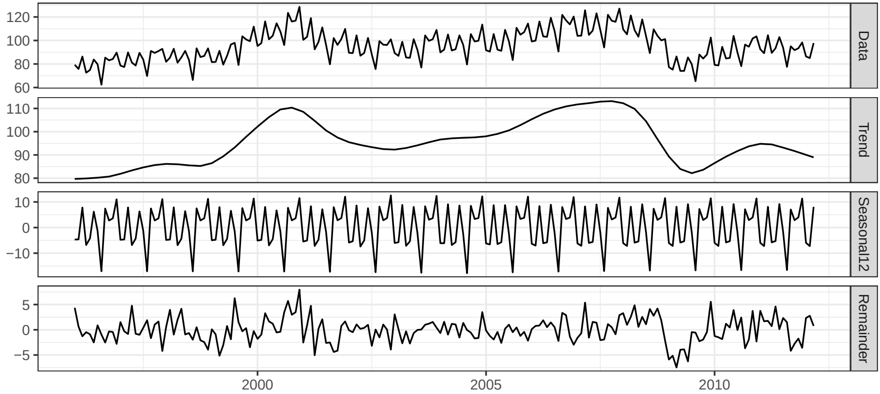
```r
autoplot(elecequip, series="Data") +
  autolayer(trendcycle(fit), series="Trend-cycle")
```



STL decomposition

We can use the `mstl` function to automate the STL calculations. There are several drawbacks to using this method so I would argue against relying on this.

```
elecequip %>% mstl() %>% autoplot()
```



Your turn

Repeat the decomposition using

```
elecequip %>%
  stl(s.window=7, t.window=11) %>%
  autoplot()
```

What happens as you change `s.window` and `t.window`?

(Dis)advantages of SEATS

Advantages

- Loess is a method for estimating nonlinear relationships.
  - Very versatile and robust.
- Unlike SEATS and X11, STL will handle any type of seasonality, not only monthly and quarterly data.
- Seasonal component allowed to change over time, and rate of change controlled by user.
- Smoothness of trend-cycle can be controlled by user.
- Robust to outliers

Disadvantages

- Not able to handle trading day or calendar adjustments.
- Only allows for additive models.
  - We can obtain an additive decomposition by first taking logs, then back transforming the components.
  - We can use the Box-Cox transformations to get other decompositions.

Section 6

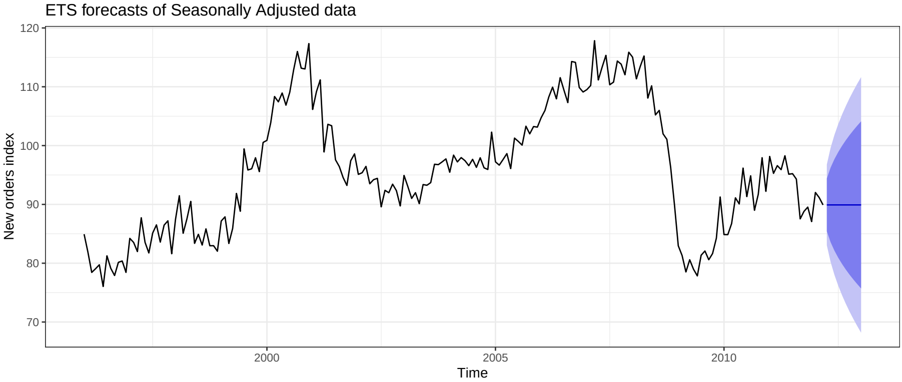
Forecasting with decomposition

Forecasting with decomposition

- Forecast seasonal component by repeating the last year
- Forecast seasonally adjusted data using non-seasonal time series method.
- Combine forecasts of seasonal component with forecasts of seasonally adjusted data to get forecasts of original data.
- Sometimes a decomposition is useful just for understanding the data before building a separate forecasting model.

Electrical equipment

```r
fit <- stl(elecequip, t.window=13, s.window="periodic")
fit %>% seasadj() %>% naive() %>%
  autoplot(series="Seasonal Adjusted") +
  labs(y = "New orders index",
       title = "ETS forecasts of Seasonally Adjusted data")
```



Electrical equipment

Exploring the results produced by the STL function:

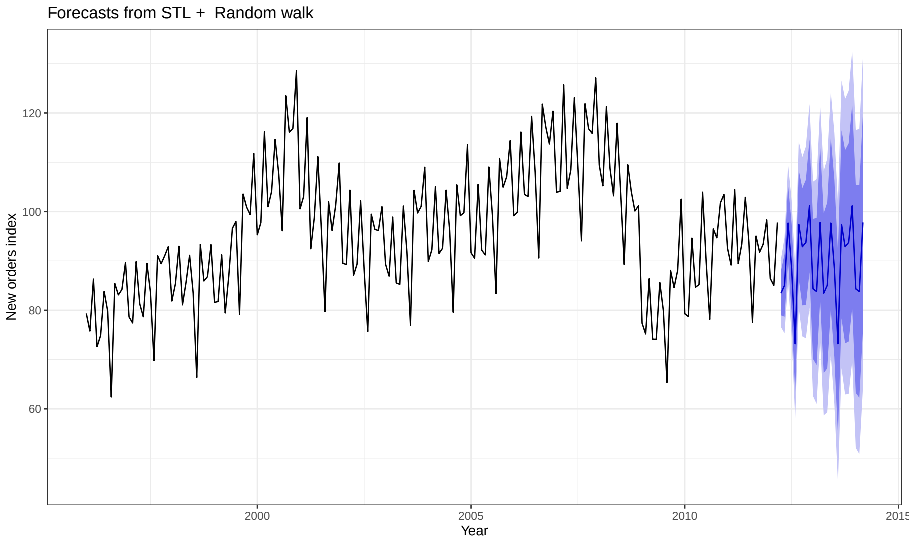
```r
head(fit$time.series)
```

```
##            seasonal    trend  remainder
## Jan 1996 -5.585463 81.97999  2.9554746
## Feb 1996 -6.114891 81.41464  0.4802485
## Mar 1996  7.885094 80.84930 -2.4143901
## Apr 1996 -6.456929 80.34599 -1.2890627
## May 1996 -4.874037 79.84269 -0.1086499
## Jun 1996  7.766608 79.45116 -3.4077635
```

We see that it produces the `seasonal`, `trend`, and `remainder` components.

- We will need to “reseasonalize” the data by adding in the seasonal naïve forecasts of the seasonal component.
    - This can be achieved using the `forecast()` function applied to the `stl` object.
    - You need to specify the method being used on the seasonally adjusted data, and the function will do the reseasonalizing for you.

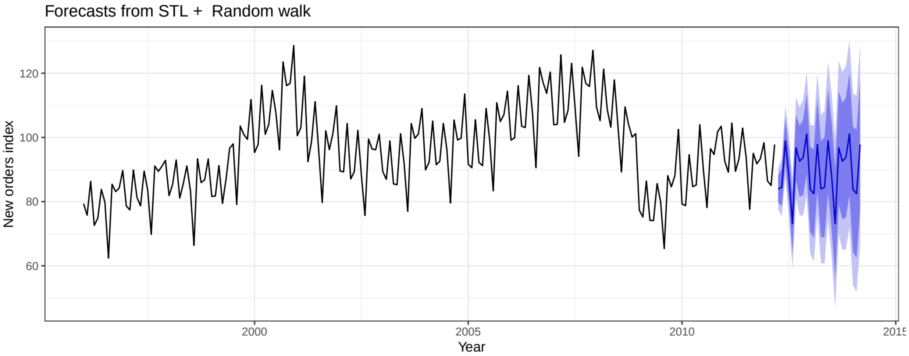
```r
fit %>% forecast(method='naive') %>%
  autoplot() + labs(y = "New orders index", x = "Year")
```



Forecasting and decomposition

A short-cut approach is to use the `stlf()` function. The following code will decompose the time series using STL, forecast the seasonally adjusted series, and return the reseasonalised forecasts.

```
elecequip %>% stlf(method='naive') %>%
  autoplot() + labs(y = "New orders index", x = "Year")
```



Decomposition and prediction intervals

- It is common to take the prediction intervals from the seasonally adjusted forecasts and modify them with the seasonal component.
- This ignores the uncertainty in the seasonal component estimate.
- It also ignores the uncertainty in the future seasonal pattern.