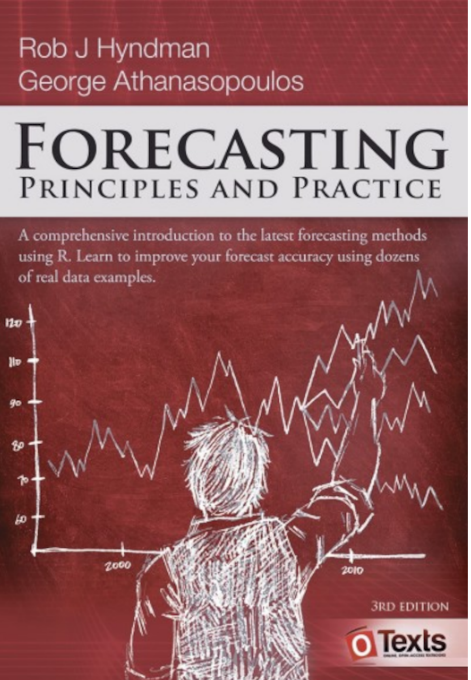


7. Time series regression models

7.4 Some useful predictors

OTexts.org/fpp3/

Trend

Linear trend

$$x_t = t$$

- $t = 1, 2, \ldots, T$
- Strong assumption that trend will continue.

Dummy variables

If a categorical variable takes only two values (e.g., ‘Yes’ or ‘No’), then an equivalent numerical variable can be constructed taking value 1 if yes and 0 if no. This is called a **dummy variable**.

Variable	dummy
Yes	1
Yes	1
No	0
Yes	1
No	0
No	0
Yes	1
Yes	1
No	0
No	0

Dummy variables

If there are more than two categories, then the variable can be coded using several dummy variables (one fewer than the total number of categories).

Day	d1	d2	d3	d4
Monday	1	0	0	0
Tuesday	0	1	0	0
Wednesday	0	0	1	0
Thursday	0	0	0	1
Friday	0	0	0	0
Monday	1	0	0	0
Tuesday	0	1	0	0
Wednesday	0	0	1	0
Thursday	0	0	0	1
Friday	0	0	0	0

Beware of the dummy variable trap!

- Using one dummy for each category gives too many dummy variables!
- The regression will then be singular and inestimable.
- Either omit the constant, or omit the dummy for one category.
- The coefficients of the dummies are relative to the omitted category.

Uses of dummy variables

Seasonal dummies

- For quarterly data: use 3 dummies
- For monthly data: use 11 dummies
- For daily data: use 6 dummies
- What to do with weekly data?

Uses of dummy variables

Seasonal dummies

- For quarterly data: use 3 dummies
- For monthly data: use 11 dummies
- For daily data: use 6 dummies
- What to do with weekly data?

Outliers

- If there is an outlier, you can use a dummy variable to remove its effect.

Uses of dummy variables

Seasonal dummies

- For quarterly data: use 3 dummies
- For monthly data: use 11 dummies
- For daily data: use 6 dummies
- What to do with weekly data?

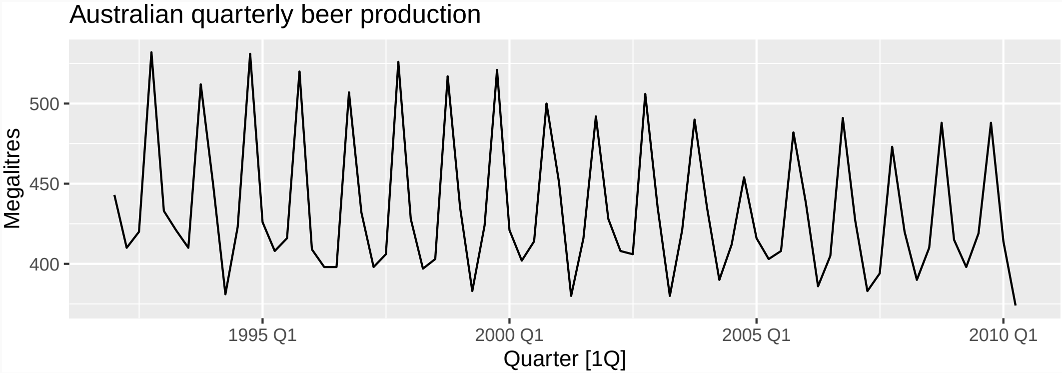
Outliers

- If there is an outlier, you can use a dummy variable to remove its effect.

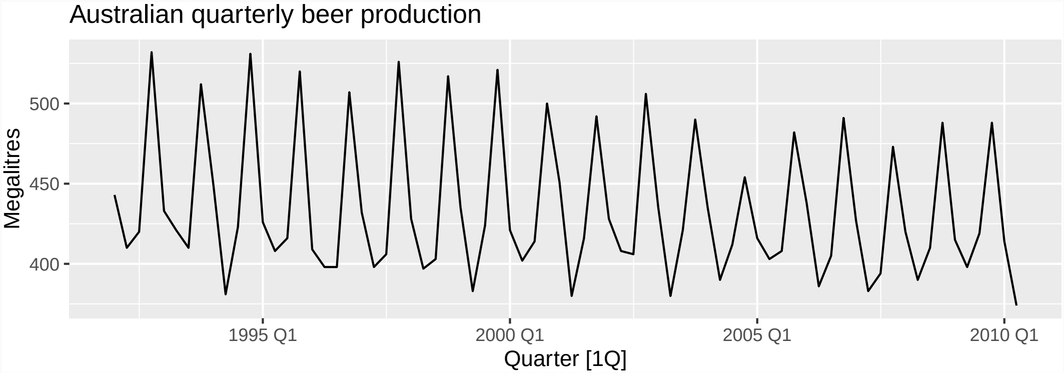
Public holidays

- For daily data: if it is a public holiday, dummy=1, otherwise dummy=0.

Beer production revisited



Beer production revisited



Regression model

$$y_t = \beta_0 + \beta_1 t + \beta_2 d_{2,t} + \beta_3 d_{3,t} + \beta_4 d_{4,t} + \varepsilon_t$$

- $d_{i,t} = 1$ if t is quarter i and 0 otherwise.

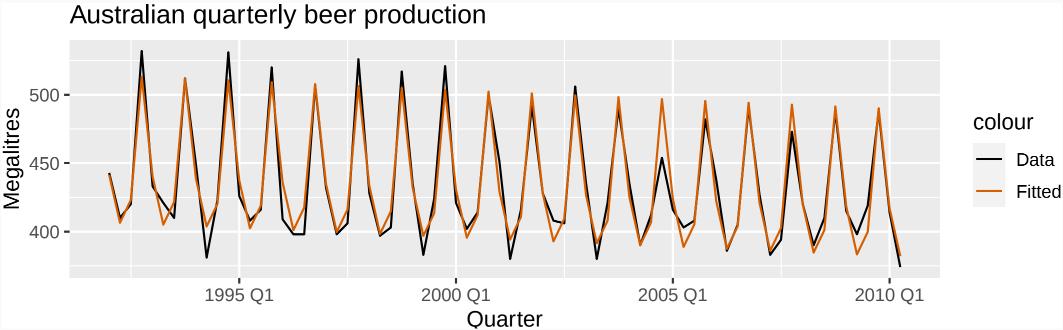
Beer production revisited

```r
fit_beer <- recent_production |> model(TSLM(Beer ~ trend() + season()))
report(fit_beer)
```

```
## Series: Beer
## Model: TSLM
##
## Residuals:
##     Min      1Q Median      3Q     Max
##   -42.9    -7.6   -0.5     8.0    21.8
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   441.8004     3.7335  118.33  < 2e-16 ***
## trend()        -0.3403     0.0666   -5.11  2.7e-06 ***
## season()year2 -34.6597     3.9683   -8.73  9.1e-13 ***
## season()year3 -17.8216     4.0225   -4.43  3.4e-05 ***
## season()year4  72.7964     4.0230   18.09  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.2 on 69 degrees of freedom
## Multiple R-squared: 0.924,   Adjusted R-squared: 0.92
## F-statistic:  211 on 4 and 69 DF, p-value: <2e-16
```

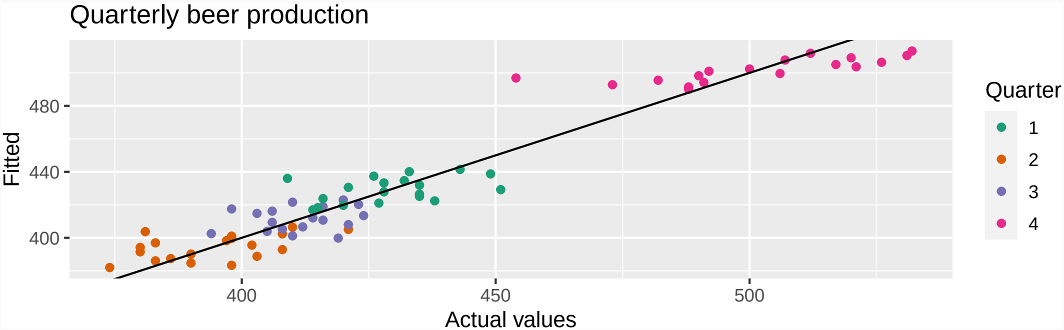
Beer production revisited

```
augment(fit_beer) |>
  ggplot(aes(x = Quarter)) +
  geom_line(aes(y = Beer, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  labs(y = "Megalitres", title = "Australian quarterly beer production") +
  scale_colour_manual(values = c(Data = "black", Fitted = "#D55E00"))
```



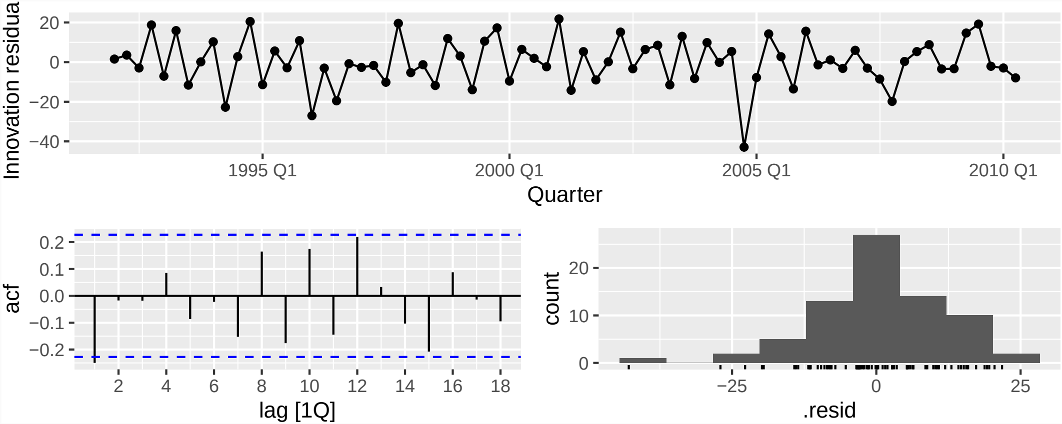
Beer production revisited

```r
augment(fit_beer) |>
  ggplot(aes(x = Beer, y = .fitted, colour = factor(quarter(Quarter)))) +
  geom_point() +
  labs(y = "Fitted", x = "Actual values", title = "Quarterly beer production") +
  scale_colour_brewer(palette = "Dark2", name = "Quarter") +
  geom_abline(intercept = 0, slope = 1)
```



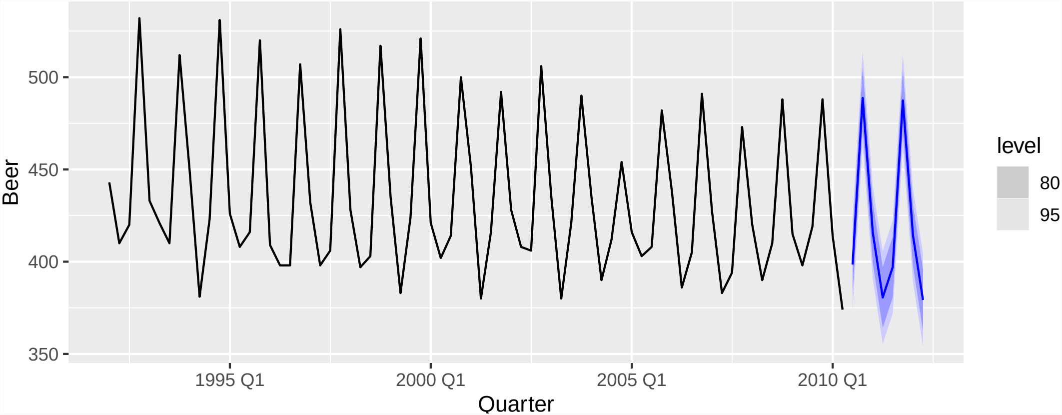
Beer production revisited

```
fit_beer |> gg_tsresiduals()
```



Beer production revisited

```
fit_beer |>
  forecast() |>
  autoplot(recent_production)
```



Intervention variables

Spikes

- Equivalent to a dummy variable for handling an outlier.

Intervention variables

Spikes

- Equivalent to a dummy variable for handling an outlier.

Steps

- Variable takes value 0 before the intervention and 1 afterwards.

Intervention variables

Spikes

- Equivalent to a dummy variable for handling an outlier.

Steps

- Variable takes value 0 before the intervention and 1 afterwards.

Change of slope

- Variables take values 0 before the intervention and values $\{1, 2, 3, \ldots\}$ afterwards.

Holidays

For monthly data

- Christmas: always in December so part of monthly seasonal effect
- Easter: use a dummy variable $v_t = 1$ if any part of Easter is in that month, $v_t = 0$ otherwise.
- Ramadan and Chinese new year similar.

Distributed lags

Lagged values of a predictor.

Example: x is advertising which has a delayed effect

$$
\begin{aligned}
x_1 &= \text{advertising for previous month;} \\
x_2 &= \text{advertising for two months previously;} \\
&\vdots \\
x_m &= \text{advertising for } m \text{ months previously.}
\end{aligned}
$$

Fourier series

Periodic seasonality can be handled using pairs of Fourier terms:

$$s_k(t) = \sin\left(\frac{2\pi kt}{m}\right) \qquad c_k(t) = \cos\left(\frac{2\pi kt}{m}\right)$$

$$y_t = a + bt + \sum_{k=1}^{K} [\alpha_k s_k(t) + \beta_k c_k(t)] + \varepsilon_t$$

- Every periodic function can be approximated by sums of sin and cos terms for large enough K.
- Choose K by minimizing AICc.
- Called "harmonic regression"

```
TSLM(y ~ trend() + fourier(K))
```

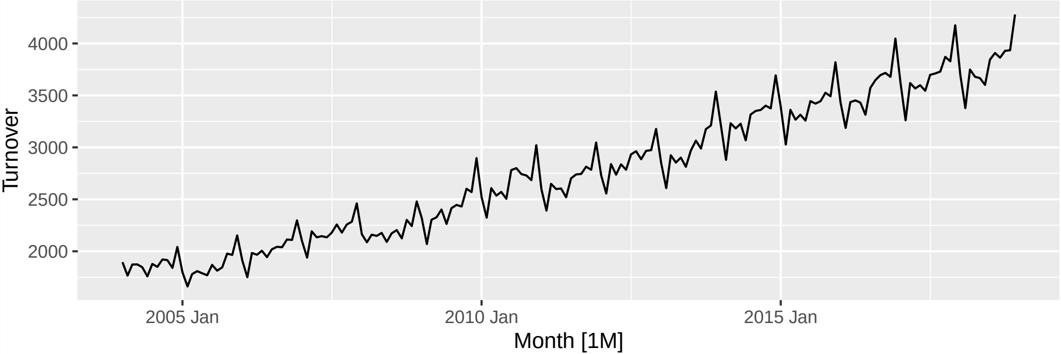
Harmonic regression: beer production

```r
fourier_beer <- recent_production |> model(TSLM(Beer ~ trend() + fourier(K = 2)))
report(fourier_beer)
```

```
## Series: Beer
## Model: TSLM
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -42.9   -7.6   -0.5    8.0   21.8
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         446.8792     2.8732  155.53  < 2e-16 ***
## trend()              -0.3403     0.0666   -5.11  2.7e-06 ***
## fourier(K = 2)C1_4    8.9108     2.0112    4.43  3.4e-05 ***
## fourier(K = 2)S1_4 -53.7281     2.0112  -26.71  < 2e-16 ***
## fourier(K = 2)C2_4 -13.9896     1.4226   -9.83  9.3e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.2 on 69 degrees of freedom
## Multiple R-squared: 0.924,   Adjusted R-squared: 0.92
## F-statistic:  211 on 4 and 69 DF, p-value: <2e-16
```

Harmonic regression: eating-out expenditure

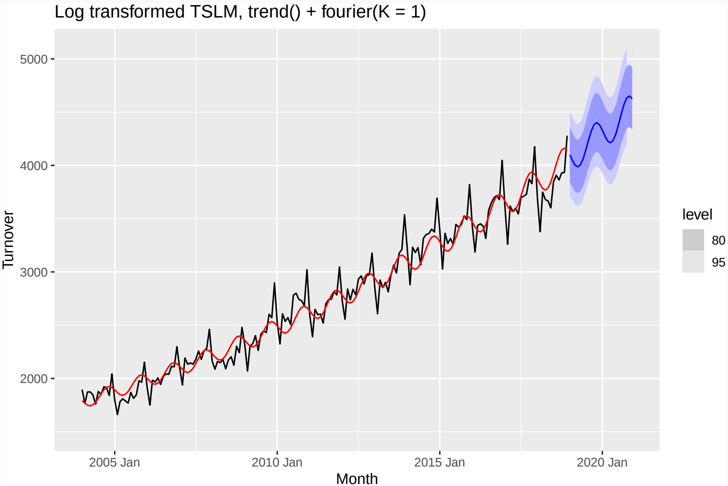
```
aus_cafe <- aus_retail |>
  filter(Industry == "Cafes, restaurants and takeaway food services",
         year(Month) %in% 2004:2018) |>
  summarise(Turnover = sum(Turnover))
aus_cafe |> autoplot(Turnover)
```



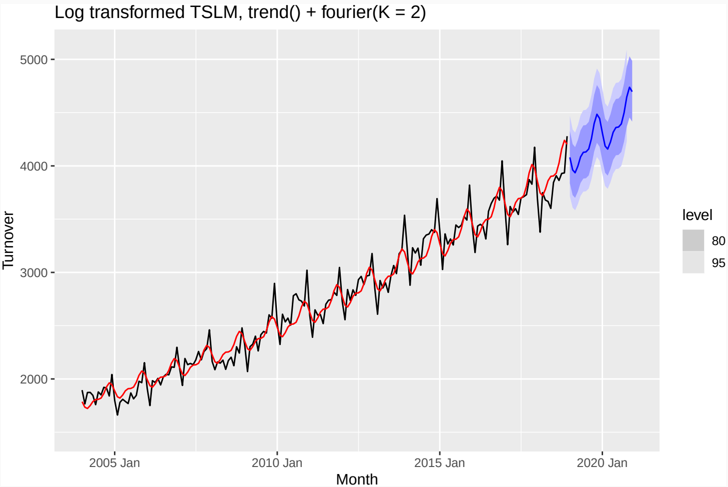
Harmonic regression: eating-out expenditure

```r
fit <- aus_cafe |>
  model(
    K1 = TSLM(log(Turnover) ~ trend() + fourier(K = 1)),
    K2 = TSLM(log(Turnover) ~ trend() + fourier(K = 2)),
    K3 = TSLM(log(Turnover) ~ trend() + fourier(K = 3)),
    K4 = TSLM(log(Turnover) ~ trend() + fourier(K = 4)),
    K5 = TSLM(log(Turnover) ~ trend() + fourier(K = 5)),
    K6 = TSLM(log(Turnover) ~ trend() + fourier(K = 6))
  )
```

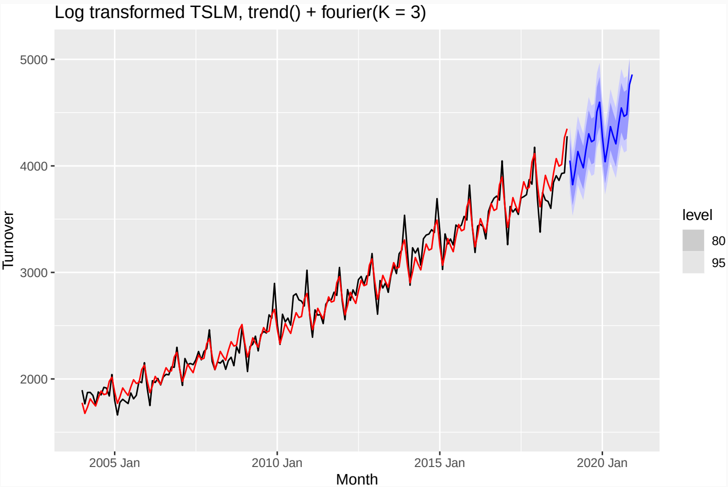
Harmonic regression: eating-out expenditure



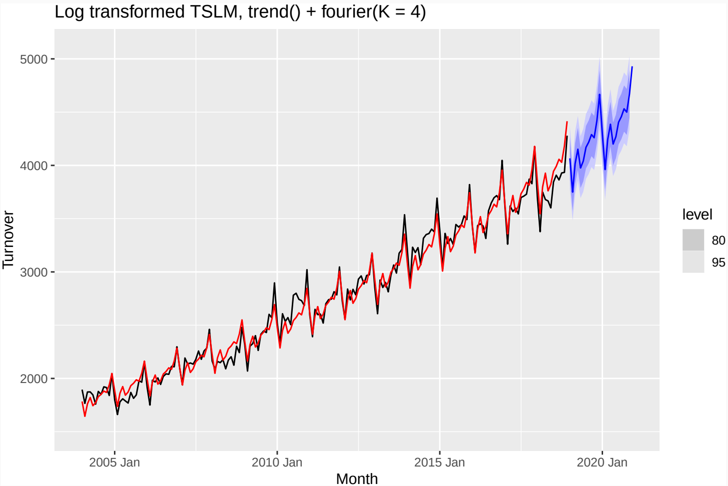
Harmonic regression: eating-out expenditure



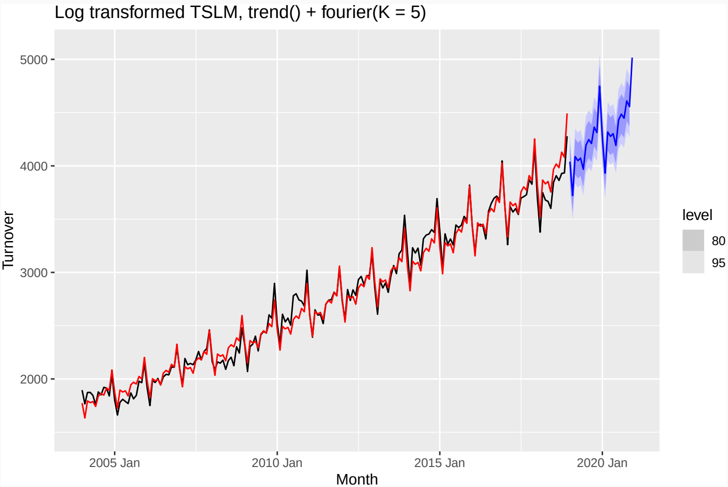
Harmonic regression: eating-out expenditure



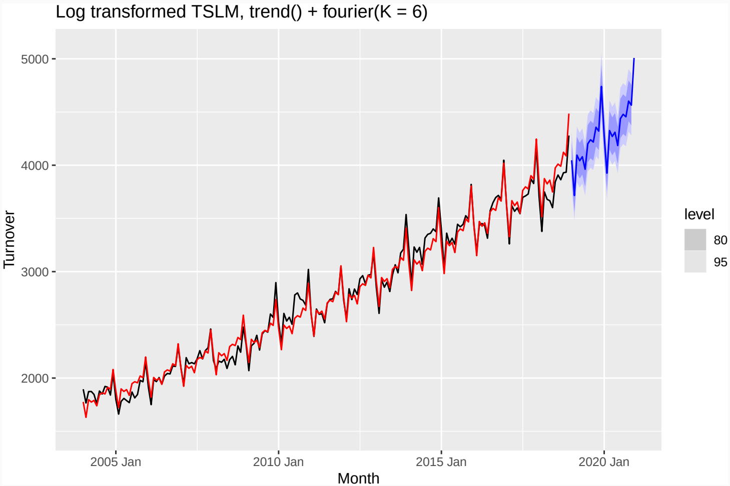
Harmonic regression: eating-out expenditure



Harmonic regression: eating-out expenditure



Harmonic regression: eating-out expenditure



Fourier series

Periodic seasonality can be handled using pairs of Fourier terms:

$$s_k(t) = \sin\left(\frac{2\pi kt}{m}\right) \qquad c_k(t) = \cos\left(\frac{2\pi kt}{m}\right)$$

$$y_t = a + bt + \sum_{k=1}^{K} [\alpha_k s_k(t) + \beta_k c_k(t)] + \varepsilon_t$$

- Every periodic function can be approximated by sums of sin and cos terms for large enough K.
- $K \leq m/2$
- m can be non-integer
- Particularly useful for large m.