

Forecasting: principles and practice

Rob J Hyndman

3.2 Forecasting with multiple
seasonality

Outline

1 Time series with complex seasonality

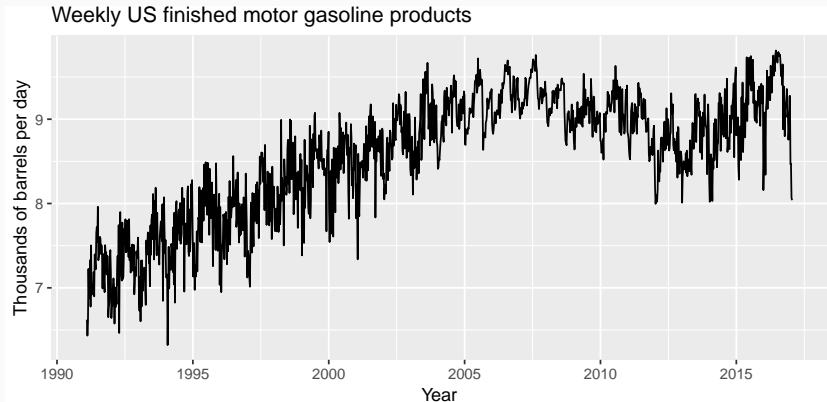
2 Lab session 17

3 Neural network models

4 Lab session 18

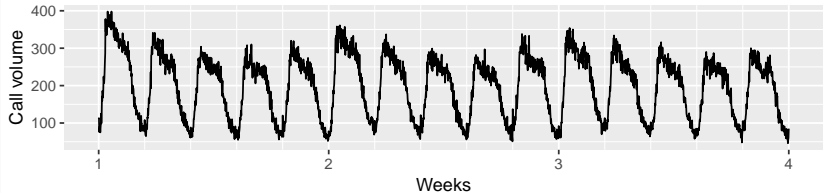
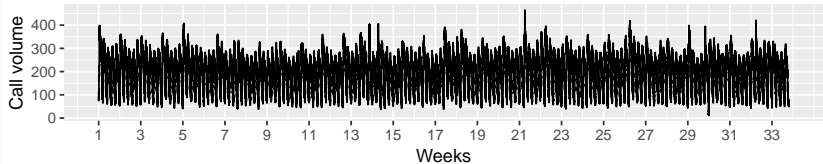
5 Lab session 19

Examples

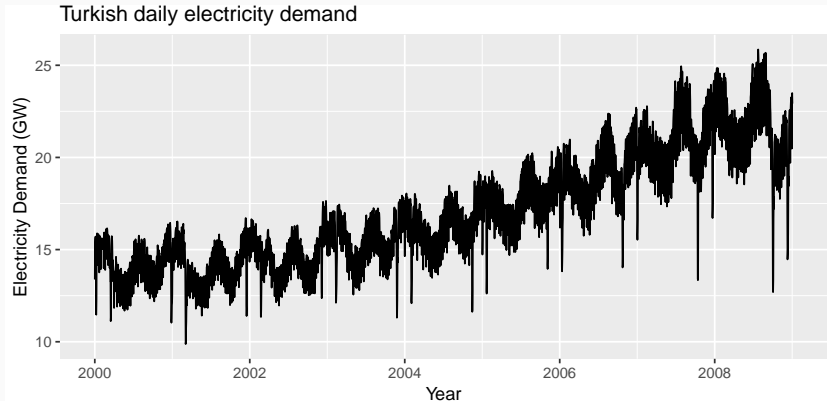


Examples

5 minute call volume at North American bank



Examples



TBATS

Trigonometric terms for seasonality

Box-Cox transformations for heterogeneity

ARMA errors for short-term dynamics

Trend (possibly damped)

Seasonal (including multiple and
non-integer periods)

TBATS model

y_t = observation at time t

$$y_t^{(\omega)} = \begin{cases} (y_t^\omega - 1)/\omega & \text{if } \omega \neq 0; \\ \log y_t & \text{if } \omega = 0. \end{cases}$$

$$y_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^M s_{t-m_i}^{(i)} + d_t$$

$$\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t$$

$$b_t = (1 - \phi)b + \phi b_{t-1} + \beta d_t$$

$$d_t = \sum_{i=1}^p \phi_i d_{t-i} + \sum_{j=1}^q \theta_j \left(\frac{s_{j,t}^{(i)}}{s_{j,t-1}^{(i)}} \cos \lambda_j^{(i)} + \frac{s_{j,t-1}^{*(i)}}{s_{j,t-1}^{(i)}} \sin \lambda_j^{(i)} + \gamma_1^{(i)} d_t \right) + \sum_{j=1}^q \theta_j \left(-\frac{s_{j,t}^{(i)}}{s_{j,t-1}^{(i)}} \sin \lambda_j^{(i)} + \frac{s_{j,t-1}^{*(i)}}{s_{j,t-1}^{(i)}} \cos \lambda_j^{(i)} + \gamma_2^{(i)} d_t \right)$$

$$s_t^{(i)} = \sum_{i=1}^{k_i} s_{i,t}^{(i)}$$

TBATS model

y_t = observation at time t

Box-Cox transformation

$$y_t^{(\omega)} = \begin{cases} (y_t^\omega - 1)/\omega & \text{if } \omega \neq 0; \\ \log y_t & \text{if } \omega = 0. \end{cases}$$

$$y_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^M s_{t-m_i}^{(i)} + d_t$$

$$\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t$$

$$b_t = (1 - \phi)b + \phi b_{t-1} + \beta d_t$$

$$d_t = \sum_{i=1}^p \phi_i d_{t-i} + \sum_{j=1}^q \theta_j \varepsilon_{t-j} + \varepsilon_t$$

$$s_{j,t}^{(i)} = s_{j,t-1}^{(i)} \cos \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \sin \lambda_j^{(i)} + \gamma_1^{(i)} d_t$$

$$s_{j,t}^{*(i)} = -s_{j,t-1}^{(i)} \sin \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \cos \lambda_j^{(i)} + \gamma_2^{(i)} d_t$$

$$s_t^{(i)} = \sum_{i=1}^{k_i} s_{i,t}^{(i)}$$

TBATS model

y_t = observation at time t

Box-Cox transformation

$$y_t^{(\omega)} = \begin{cases} (y_t^\omega - 1)/\omega & \text{if } \omega \neq 0; \\ \log y_t & \text{if } \omega = 0. \end{cases}$$

M seasonal periods

$$y_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^M s_{t-m_i}^{(i)} + d_t$$

$$\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t$$

$$b_t = (1 - \phi)b + \phi b_{t-1} + \beta d_t$$

$$d_t = \sum_{i=1}^p \phi_i d_{t-i} + \sum_{j=1}^q \theta_j \left(\sum_{i=1}^q s_{j,t-i}^{(i)} \cos \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \sin \lambda_j^{(i)} + \gamma_1^{(i)} d_t \right) \\ + \sum_{j=1}^q \theta_j \left(\sum_{i=1}^q s_{j,t-i}^{(i)} \sin \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \cos \lambda_j^{(i)} + \gamma_2^{(i)} d_t \right)$$

$$s_t^{(i)} = \sum_{i=1}^{k_i} s_{i,t}^{(i)}$$

TBATS model

y_t = observation at time t

Box-Cox transformation

$$y_t^{(\omega)} = \begin{cases} (y_t^\omega - 1)/\omega & \text{if } \omega \neq 0; \\ \log y_t & \text{if } \omega = 0. \end{cases}$$

M seasonal periods

$$y_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^M s_{t-m_i}^{(i)} + d_t$$

global and local trend

$$\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t$$

$$b_t = (1 - \phi)b + \phi b_{t-1} + \beta d_t$$

$$d_t = \sum_{i=1}^p \phi_i d_{t-i} + \sum_{j=1}^q \theta_j^{(i)} \left[\frac{s_{j,t}^{(i)}}{s_{j,t-1}^{(i)}} \cos \lambda_j^{(i)} + \frac{s_{j,t-1}^{*(i)}}{s_{j,t-1}^{(i)}} \sin \lambda_j^{(i)} + \gamma_1^{(i)} d_t \right]$$

$$s_{j,t}^{(i)} = -s_{j,t-1}^{(i)} \sin \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \cos \lambda_j^{(i)} + \gamma_2^{(i)} d_t$$

$$s_t^{(i)} = \sum_{i=1}^{k_i} s_{i,t}^{(i)}$$

TBATS model

y_t = observation at time t

$$y_t^{(\omega)} = \begin{cases} (y_t^\omega - 1)/\omega & \text{if } \omega \neq 0; \\ \log y_t & \text{if } \omega = 0. \end{cases}$$

Box-Cox transformation

M seasonal periods

$$y_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^M s_{t-m_i}^{(i)} + d_t$$

global and local trend

$$\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t$$

$$b_t = (1 - \phi)b + \phi b_{t-1} + \beta d_t$$

ARMA error

$$d_t = \sum_{i=1}^p \phi_i d_{t-i} + \sum_{j=1}^q \theta_j \varepsilon_{t-j} + \varepsilon_t$$

$$s_{j,t}^{(i)} = s_{j,t-1}^{(i)} \cos \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \sin \lambda_j^{(i)} + \gamma_1^{(i)} d_t$$

$$s_{j,t}^{(i)} = -s_{j,t-1}^{(i)} \sin \lambda_j^{(i)} + s_{j,t-1}^{*(i)} \cos \lambda_j^{(i)} + \gamma_2^{(i)} d_t$$

$$s_t^{(i)} = \sum_{k_i} s_{i,t}^{(i)}$$

TBATS model

y_t = observation at time t

$$y_t^{(\omega)} = \begin{cases} (y_t^\omega - 1)/\omega & \text{if } \omega \neq 0; \\ \log y_t & \text{if } \omega = 0. \end{cases}$$

Box-Cox transformation

M seasonal periods

$$y_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^M s_{t-m_i}^{(i)} + d_t$$

global and local trend

$$\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t$$

$$b_t = (1 - \phi)b + \phi b_{t-1} + \beta d_t$$

ARMA error

$$d_t = \sum_{i=1}^p \phi_i d_{t-i} + \sum_{j=1}^q \theta_j \varepsilon_{t-j} + \varepsilon_t$$

Fourier-like seasonal terms

$$s_t^{(i)} = \sum_{j=1}^{k_i} s_{i,t}^{(j)}$$

TBATS model

y_t = observation at time t

$$y_t^{(\omega)} = \begin{cases} \text{TBATS} & \text{if } \omega = 0; \\ \text{Trigonometric} & \text{otherwise.} \end{cases}$$

$$y_t^{(\omega)} = \ell_t + b_t + d_t + \varepsilon_t$$

$$\ell_t = \ell$$

$$b_t = (b_j)_{j=1}^M$$

$$d_t = \sum_{i=1}^p \phi_i d_{t-i} + \sum_{j=1}^q \theta_j \varepsilon_{t-j} + \varepsilon_t$$

$$s_t^{(i)} = \sum_{i=1}^{k_i} s_{i,t}^{(i)}$$

Box-Cox transformation

M seasonal periods

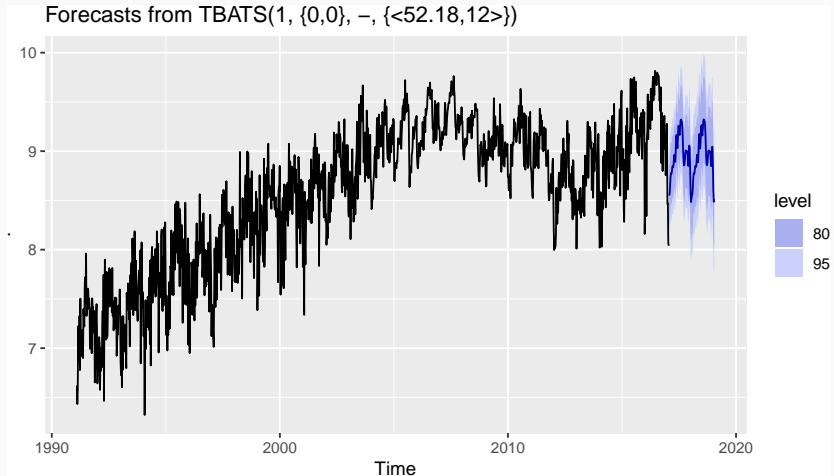
global and local trend

ARMA error

Fourier-like seasonal terms

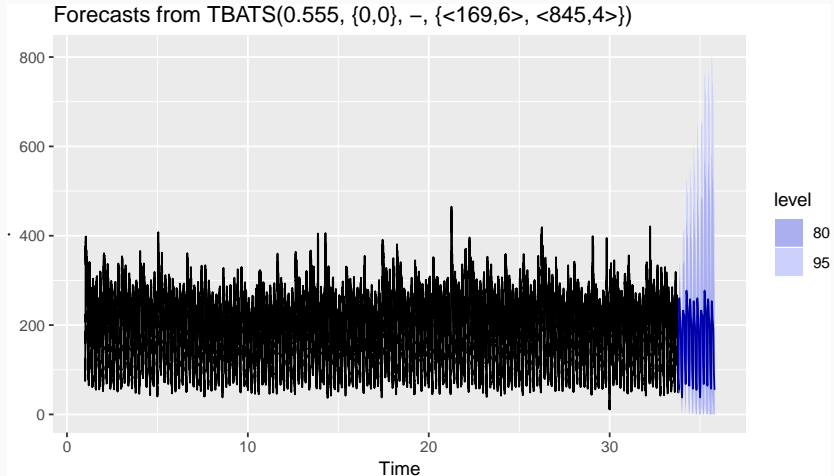
Complex seasonality

gasoline %>% tbats %>% forecast %>% autoplot



Complex seasonality

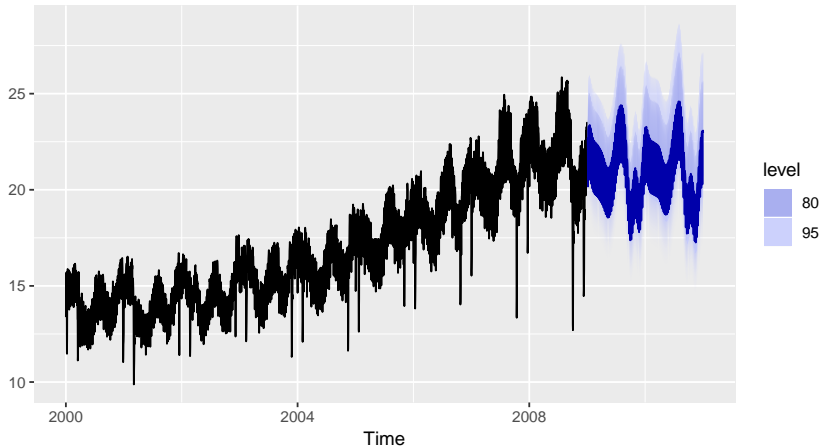
```
calls %>% tbats %>% forecast %>% autoplot
```



Complex seasonality

```
telec %>% tbats %>% forecast %>% autoplot
```

Forecasts from TBATS(0.005, {4,2}, -, {<7,3>, <354.37,7>, <365.25,3>})



TBATS

Trigonometric terms for seasonality

Box-Cox transformations for heterogeneity

ARMA errors for short-term dynamics

Trend (possibly damped)

Seasonal (including multiple and non-integer periods)

- Handles non-integer seasonality, multiple seasonal periods.
- Entirely automated
- Prediction intervals often too wide
- Very slow on long series

Outline

1 Time series with complex seasonality

2 Lab session 17

3 Neural network models

4 Lab session 18

5 Lab session 19

Lab Session 17

Outline

1 Time series with complex seasonality

2 Lab session 17

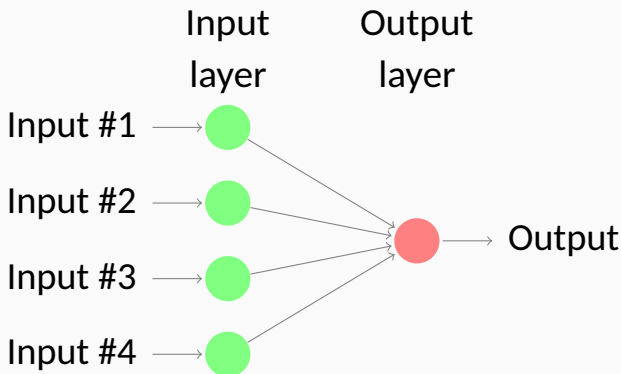
3 Neural network models

4 Lab session 18

5 Lab session 19

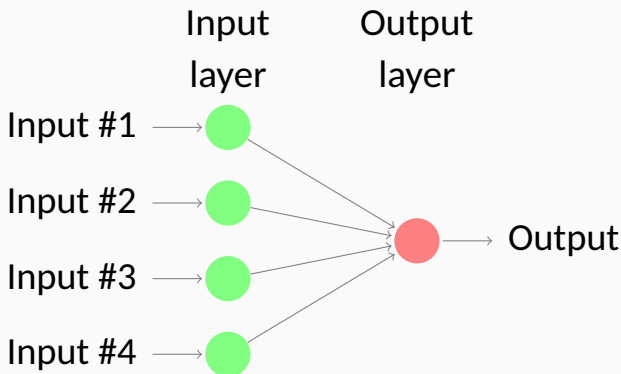
Neural network models

Simplest version: linear regression



Neural network models

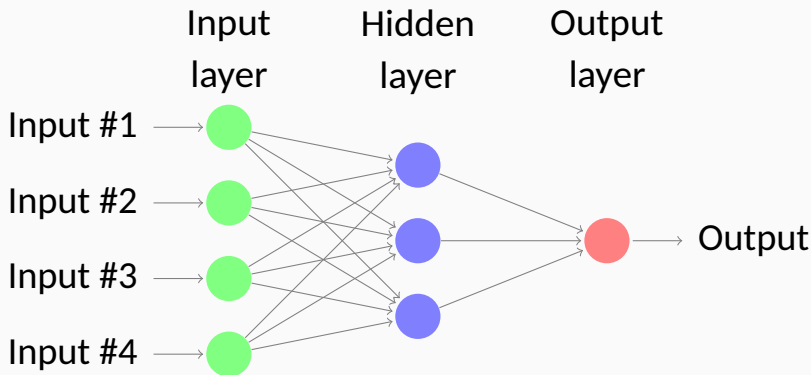
Simplest version: linear regression



- Coefficients attached to predictors are called “weights”.
- Forecasts are obtained by a linear combination of inputs.
- Weights selected using a “learning algorithm” that

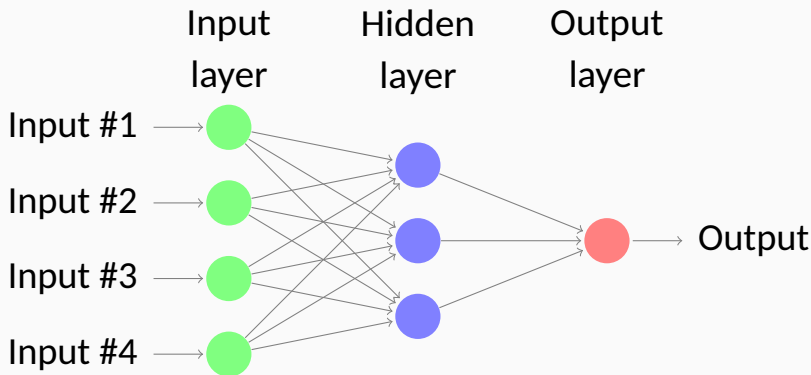
Neural network models

Nonlinear model with one hidden layer



Neural network models

Nonlinear model with one hidden layer



- A **multilayer feed-forward network** where each layer of nodes receives inputs from the previous layers.
- Inputs to each node combined using linear

Neural network models

Inputs to hidden neuron j linearly combined:

$$z_j = b_j + \sum_{i=1}^4 w_{i,j} x_i.$$

Modified using nonlinear function such as a sigmoid:

$$s(z) = \frac{1}{1 + e^{-z}},$$

This tends to reduce the effect of extreme input values, thus making the network somewhat robust to outliers.

Neural network models

- Weights take random values to begin with, which are then updated using the observed data.
- There is an element of randomness in the predictions. So the network is usually trained several times using different random starting points, and the results are averaged.
- Number of hidden layers, and the number of nodes in each hidden layer, must be specified in advance.

NNAR models

- Lagged values of the time series can be used as inputs to a neural network.
- $NNAR(p, k)$: p lagged inputs and k nodes in the single hidden layer.
- $NNAR(p, 0)$ model is equivalent to an $ARIMA(p, 0, 0)$ model but without stationarity restrictions.
- Seasonal $NNAR(p, P, k)$: inputs $(y_{t-1}, y_{t-2}, \dots, y_{t-p}, y_{t-m}, y_{t-2m}, y_{t-Pm})$ and k neurons in the hidden layer.
- $NNAR(p, P, 0)_m$ model is equivalent to an $ARIMA(p, 0, 0)(P, 0, 0)_m$ model but without

NNAR models in R

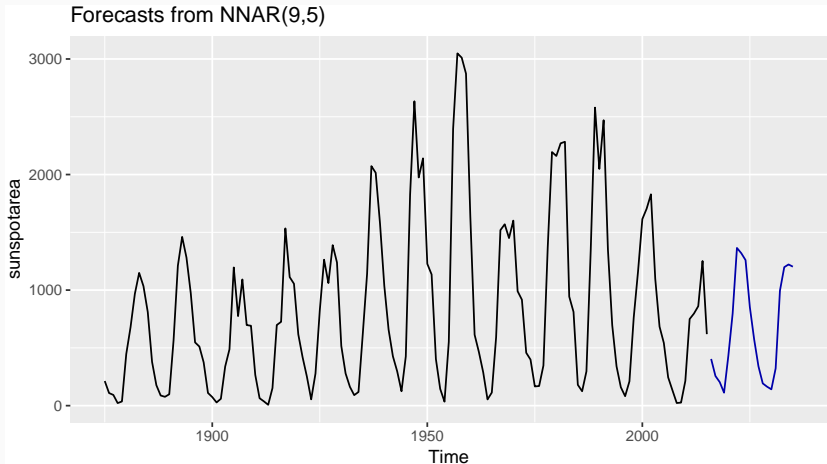
- The `nnetar()` function fits an $\text{NNAR}(p, P, k)_m$ model.
- If p and P are not specified, they are automatically selected.
- For non-seasonal time series, default p = optimal number of lags (according to the AIC) for a linear $\text{AR}(p)$ model.
- For seasonal time series, defaults are $P = 1$ and p is chosen from the optimal linear model fitted to the seasonally adjusted data.
- Default $k = (p + P + 1)/2$ (rounded to the nearest integer)

Sunspots

- Surface of the sun contains magnetic regions that appear as dark spots.
- These affect the propagation of radio waves and so telecommunication companies like to predict sunspot activity in order to plan for any future difficulties.
- Sunspots follow a cycle of length between 9 and 14 years.

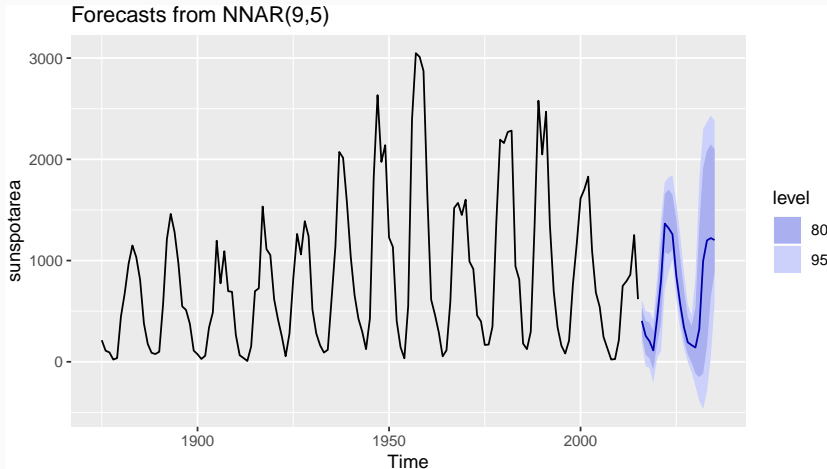
NNAR(9,5) model for sunspots

```
fit <- nnetar(sunspotarea)
fit %>% forecast(h=20) %>% autoplot
```



Prediction intervals by simulation

```
fit %>% forecast(h=20, PI=TRUE) %>% autoplot
```



Outline

1 Time series with complex seasonality

2 Lab session 17

3 Neural network models

4 Lab session 18

5 Lab session 19

Lab Session 18

Outline

1 Time series with complex seasonality

2 Lab session 17

3 Neural network models

4 Lab session 18

5 Lab session 19

Lab Session 19