BITS Pilani
Pilani Campus

Research Presentation

Ayushi Gaur
2019PHXP0023P

Outline

- Multivariate Time Series
- Components of Time Series
- How to model a time series?
- Comparison of modeling techniques

Multi-variate Time Series

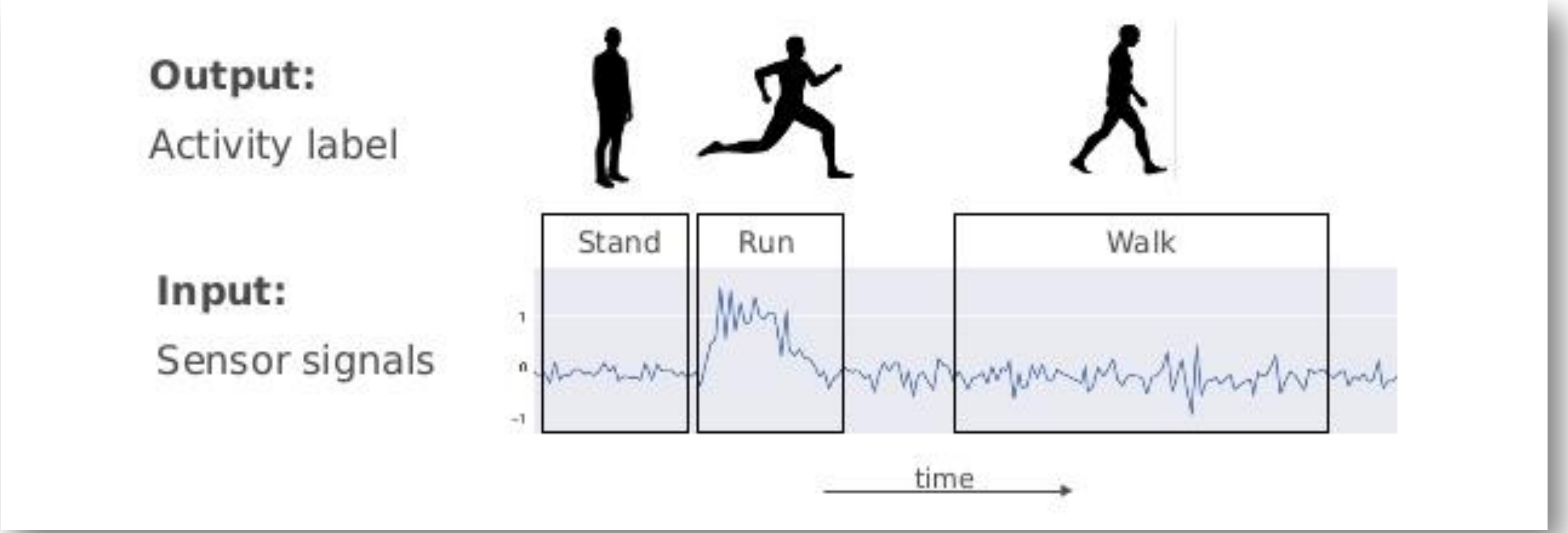
Multivariate Time Series (MVTS)

A *time series data stream* is an infinite sequence of elements

$$S = \{x_1, \ldots, x_i, \ldots\}$$

where x_i is a d-dimensional data vector arriving at time stamp i.

Human activity analysis



Ordóñez, F. J., & Roggen, D. (2016). Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. *Sensors*, *16*(1), 115.

BITS Pilani
Pilani Campus

Components of time series

Components of time series

Time series consist of four components:

(1) Seasonal variations: that repeat over a specific period such as a day, week, month, season, etc.

Components of time series

Time series consist of four components:

(1) Seasonal variations: that repeat over a specific period such as a day, week, month, season, etc.,

(2) Trend: Trend is defined as long term increase or decrease in the data. It can be linear or non-linear

Components of time series

Time series consist of four components:

(1) Seasonal variations: that repeat over a specific period such as a day, week, month, season, etc.,

(2) Trend: Trend is defined as long term increase or decrease in the data. It can be linear or non-linear

(3) Cyclical variations: A cyclic pattern exists when data exhibit rises and falls that are not of fixed period.

Components of time series

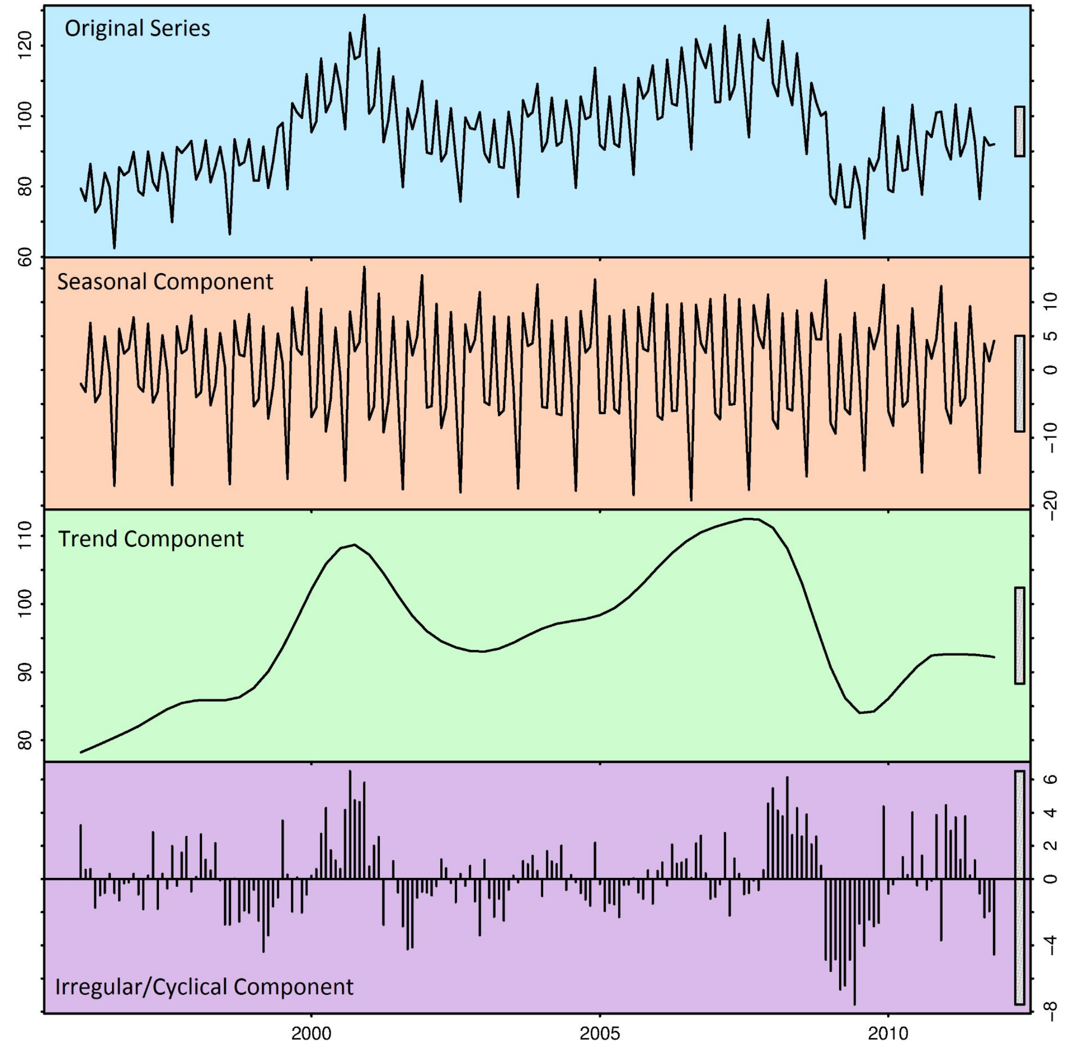
Time series consist of four components:

(1) Seasonal variations: that repeat over a specific period such as a day, week, month, season, etc.,

(2) Trend: Trend is defined as long term increase or decrease in the data. It can be linear or non-linear

(3) Cyclical variations: A cyclic pattern exists when data exhibit rises and falls that are not of fixed period.

(4) Random variations: that do not fall under any of the above three classifications.


Original Series
Seasonal Component
Trend Component
Irregular/Cyclical Component
60
80
100
120
10
5
0
−10
−20
80
90
100
110
6
4
2
0
−4
−8
2000
2005
2010

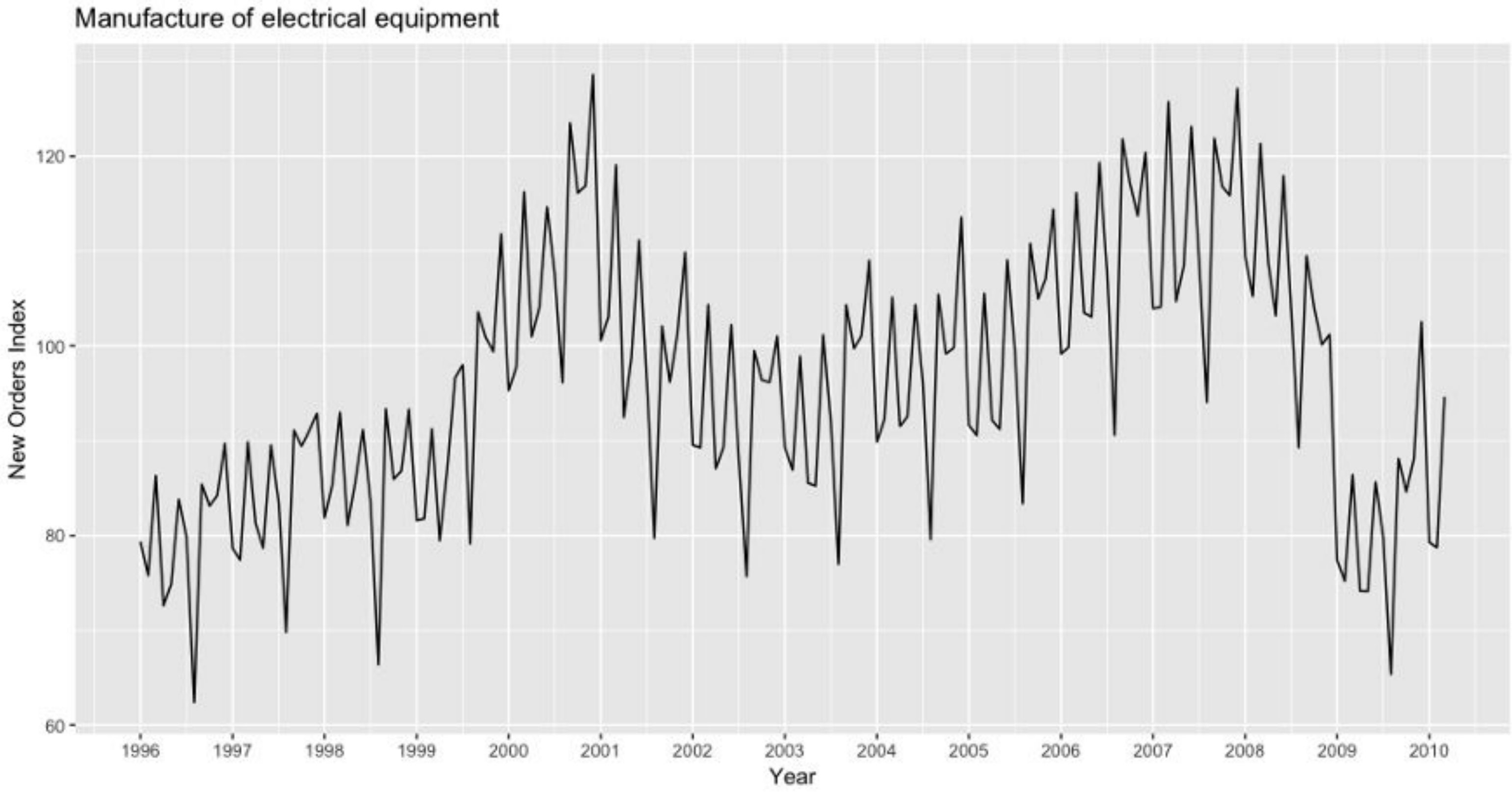
How to model a time series?

Techniques to model time series

1. **Naïve, SNaïve**
2. **Seasonal decomposition (+ any model)**
3. **Exponential smoothing**
4. **ARIMA, SARIMA**
5. **GARCH**
6. **Dynamic linear models**
7. **TBATS**
8. **Prophet**
9. **NNETAR**
10. **LSTM**

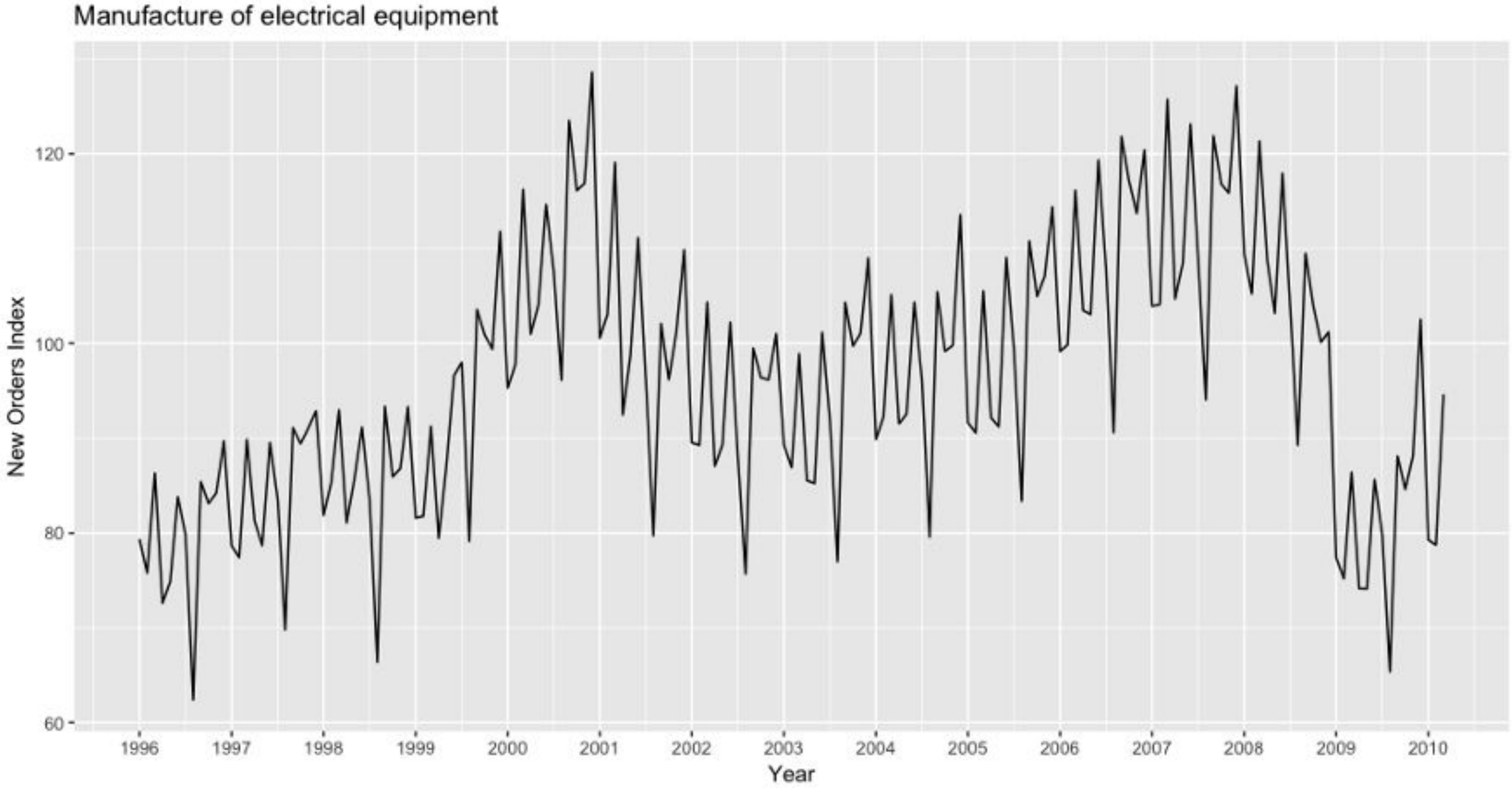
Dataset

The dataset corresponds to monthly manufacture of electrical equipment (computer, electronic and optical products) in the Euro area (17 countries) in the period January 1996-March 2012. We keep the **last 2 years for testing** purposes.



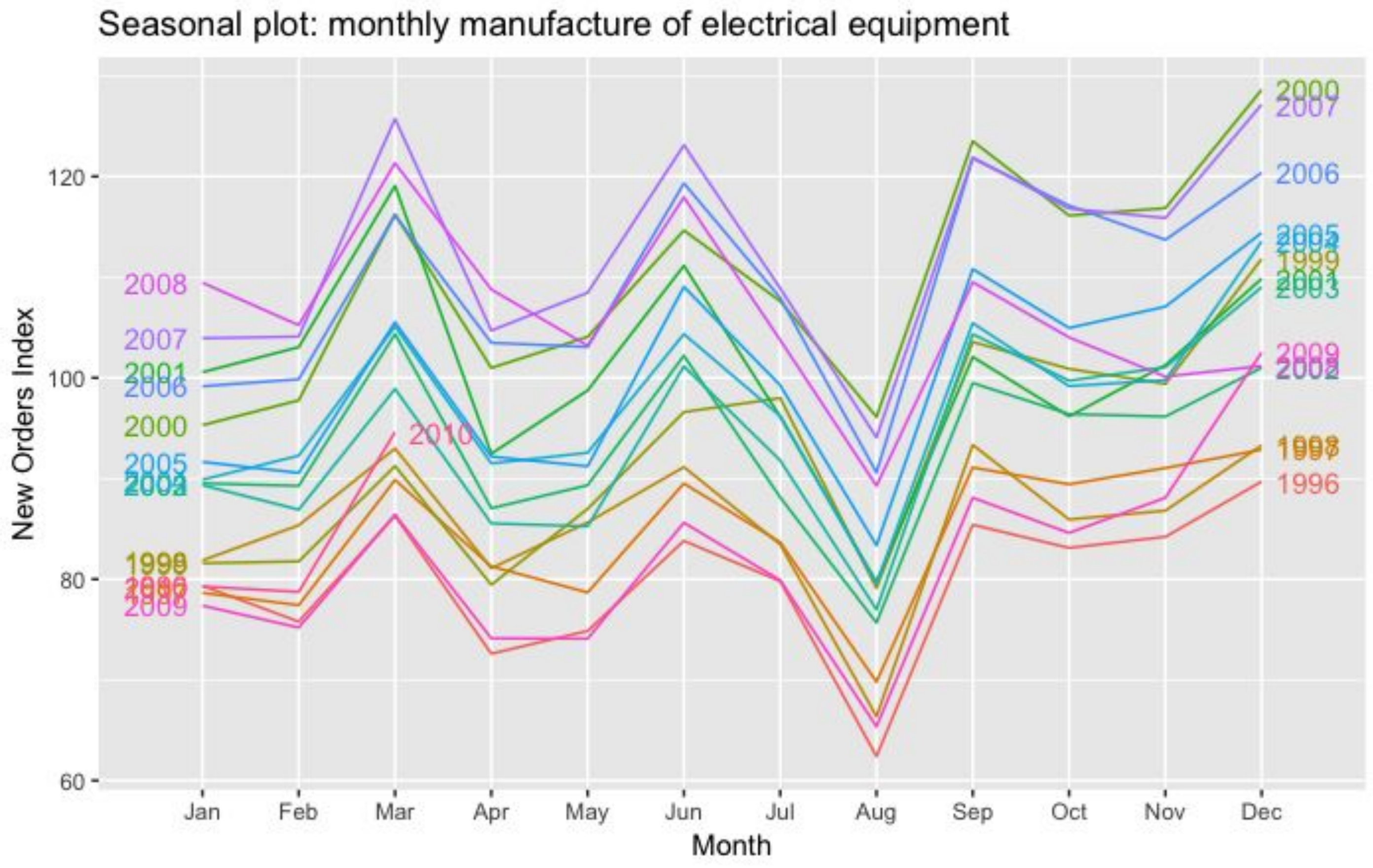
Dataset

The time series has a peak at the end of 2000 and another one during 2007. The huge decrease that we observe at the end of 2008 is probably due to the global financial crisis which occurred during that year.

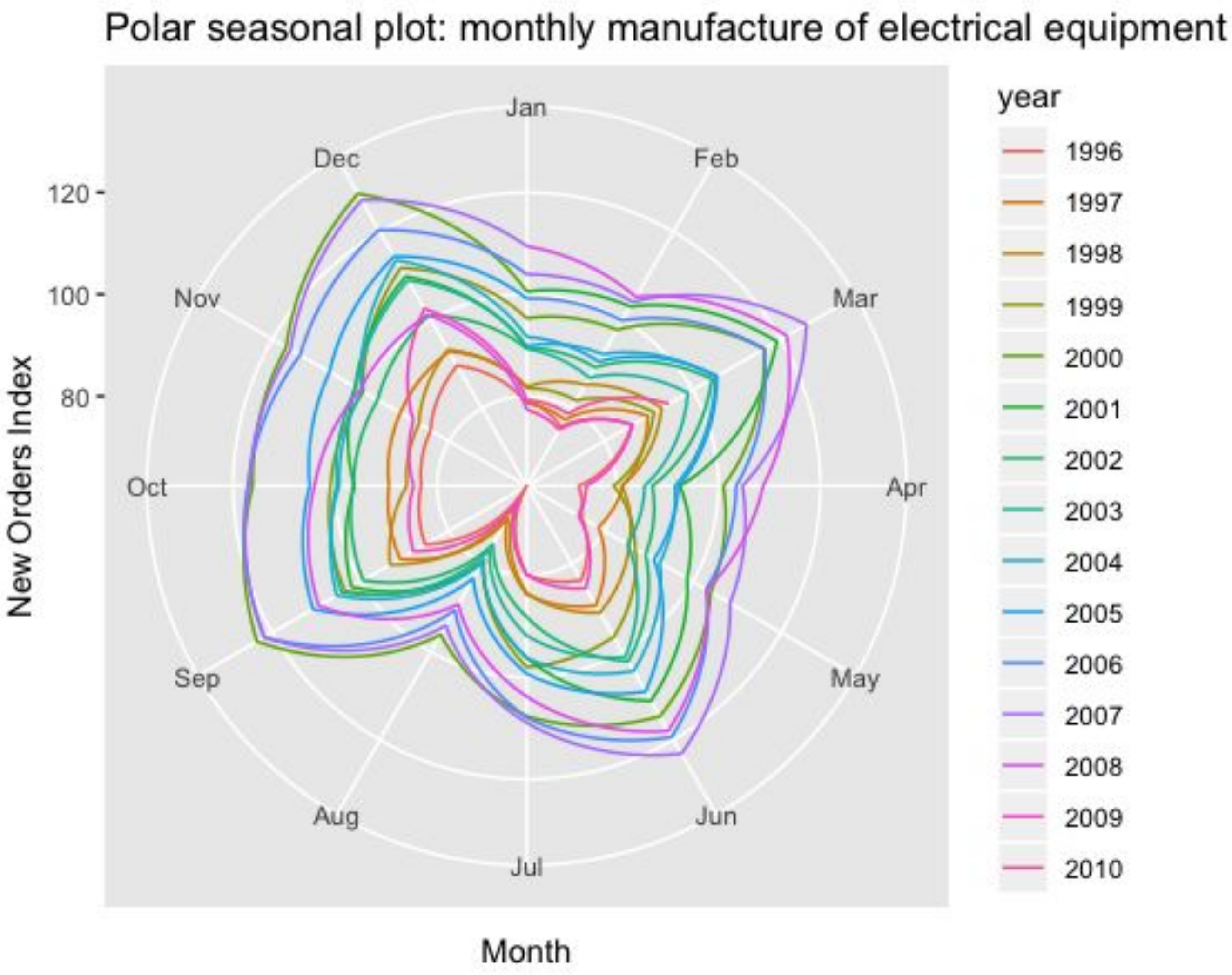


Seasonal pattern in the dataset

There seems to be a yearly seasonal pattern. To better visualise this, we show data for each year separately in both original and polar coordinates.



Seasonal pattern in the dataset



Naïve, SNaïve

- In the Naïve model, the forecasts for every horizon correspond to the last observed value.

$$\hat{Y}(t+h|t) = Y(t)$$

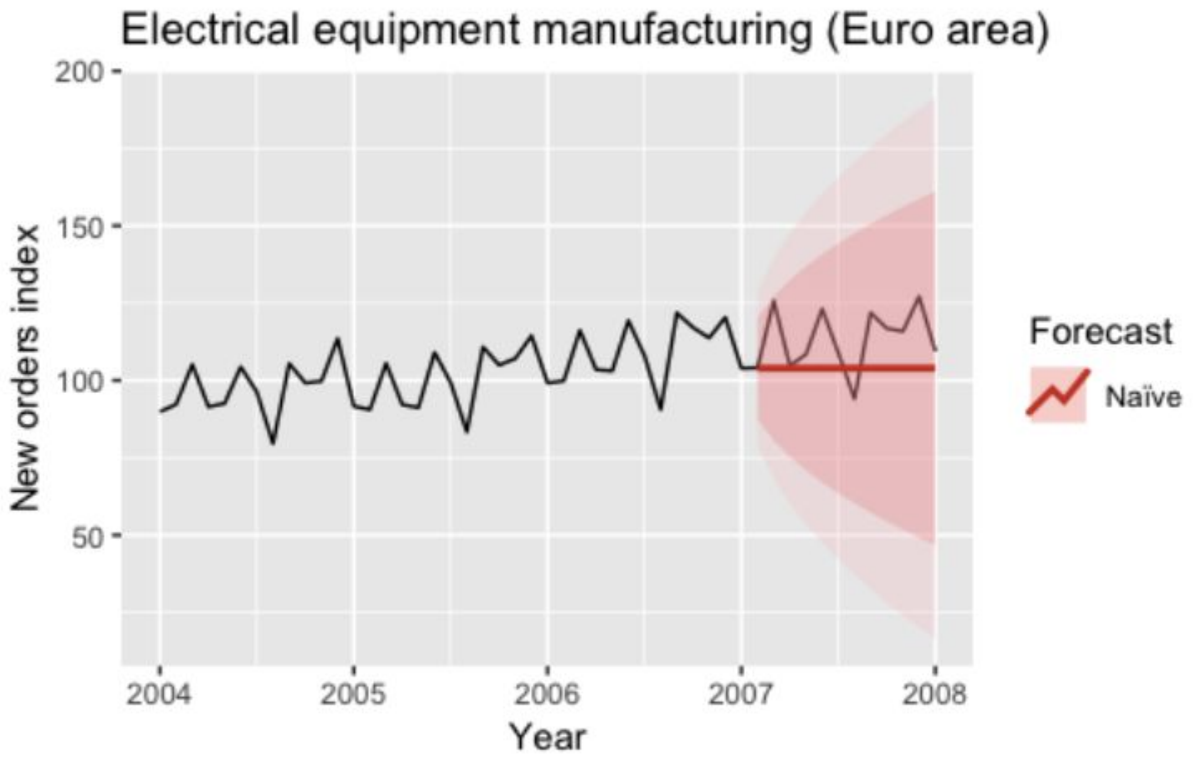
- assumes that the stochastic model generating the time series is a random walk.

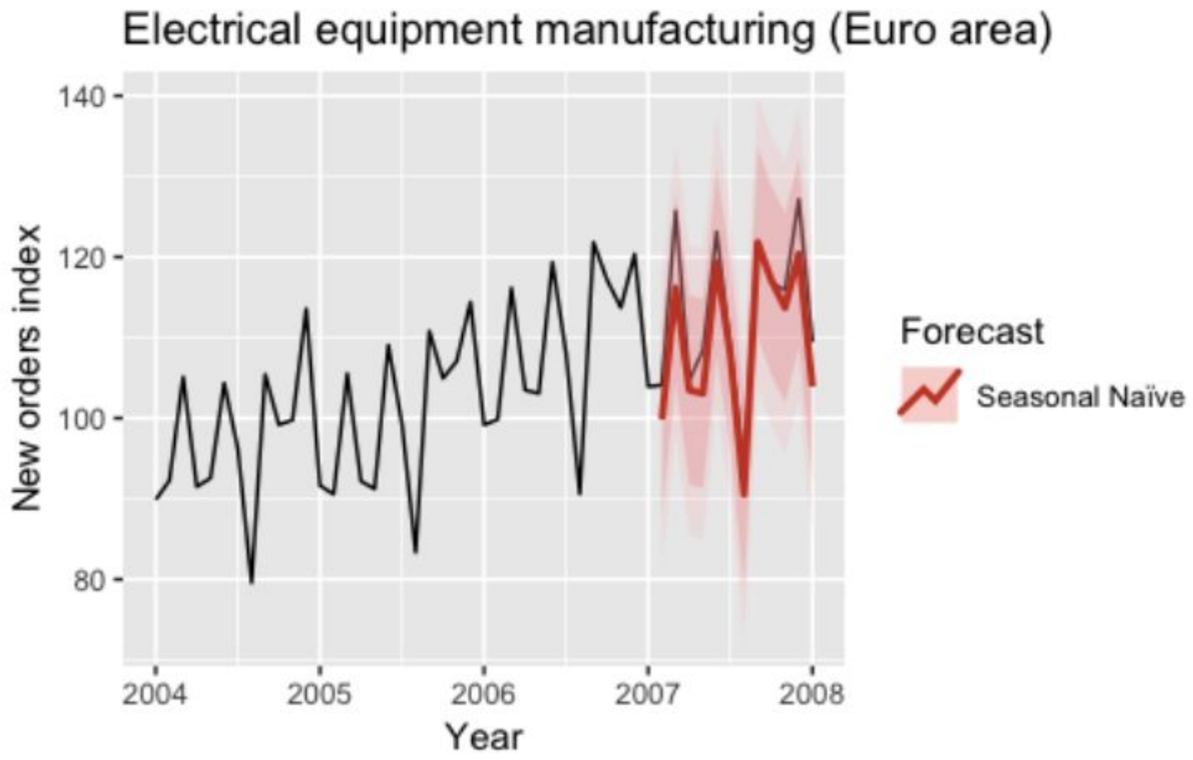
- extension given by the SNaïve (Seasonal Naïve) model. Assuming that the time series has a seasonal component and that the period of the seasonality is T, the forecasts given by the SNaïve model are given by:

$$\hat{Y}(t+h|t) = Y(t+h-T)$$

- Therefore the forecasts for the following T time steps are equal to the previous T time steps. In our application, the SNaïve forecast for the next year is equal to the last year's observations.

Naïve, SNaïve





Seasonal decomposition (+ any model)

- If data shows some seasonality (e.g. daily, weekly, quarterly, yearly) it may be useful to decompose the original time series into the sum of three components:

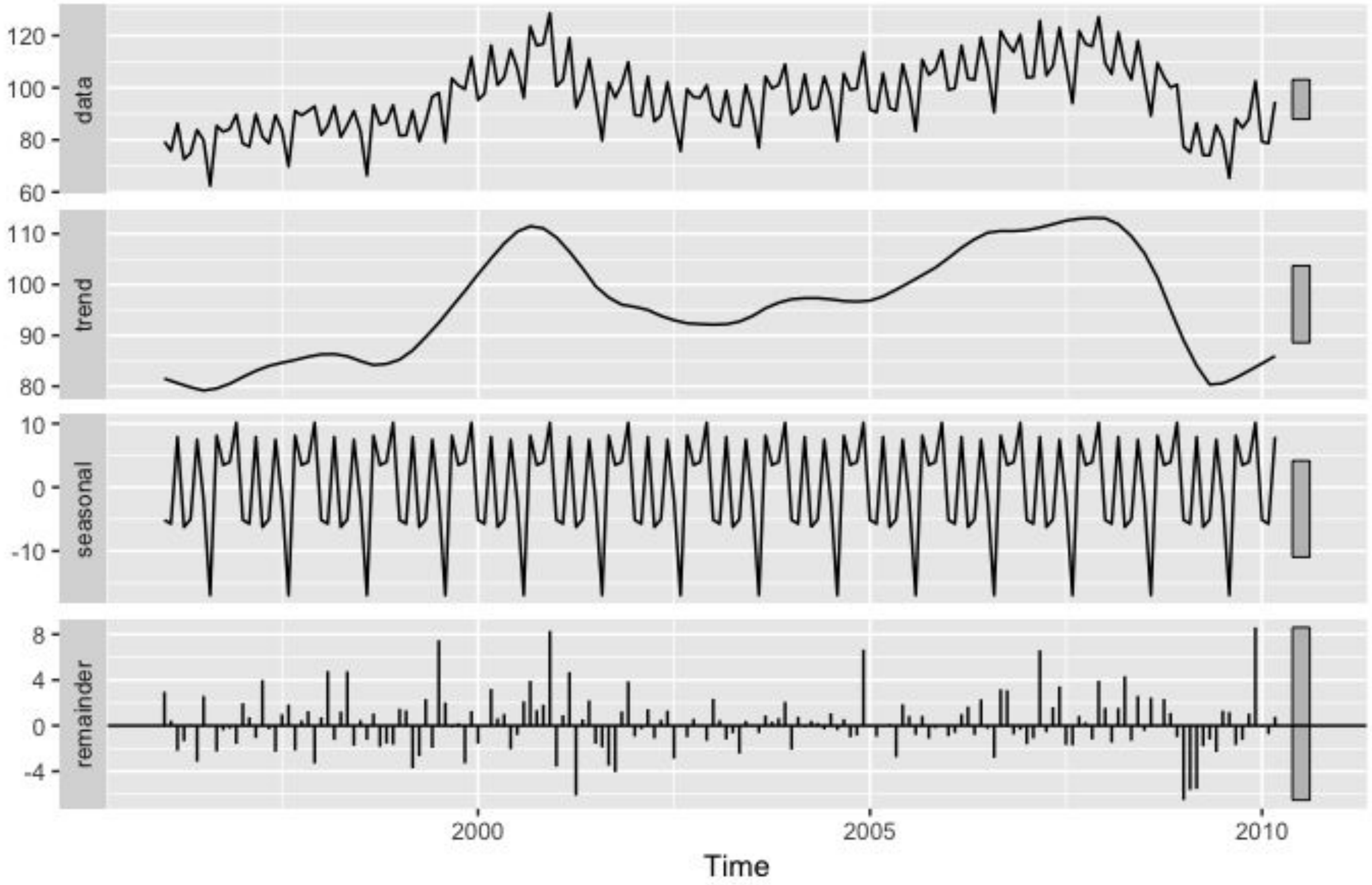
  $$Y(t) = S(t) + T(t) + R(t)$$

  where *S(t)* is the seasonal component, *T(t)* is the trend-cycle component, and *R(t)* is the remainder component

- classical decomposition technique:
  - Estimating trend T(t) through a rolling mean
  - Computing S(t) as the average detrended series **Y(t)-T(t)** for each season (e.g. for each month)
  - Computing the remainder series as **R(t)=Y(t)-T(t)-S(t)**

- extended in several ways. Its extensions allow:
  - having a non-constant seasonality
  - computing initial and last values of the decomposition
  - avoiding over-smoothing

Seasonal decomposition (+ any model)

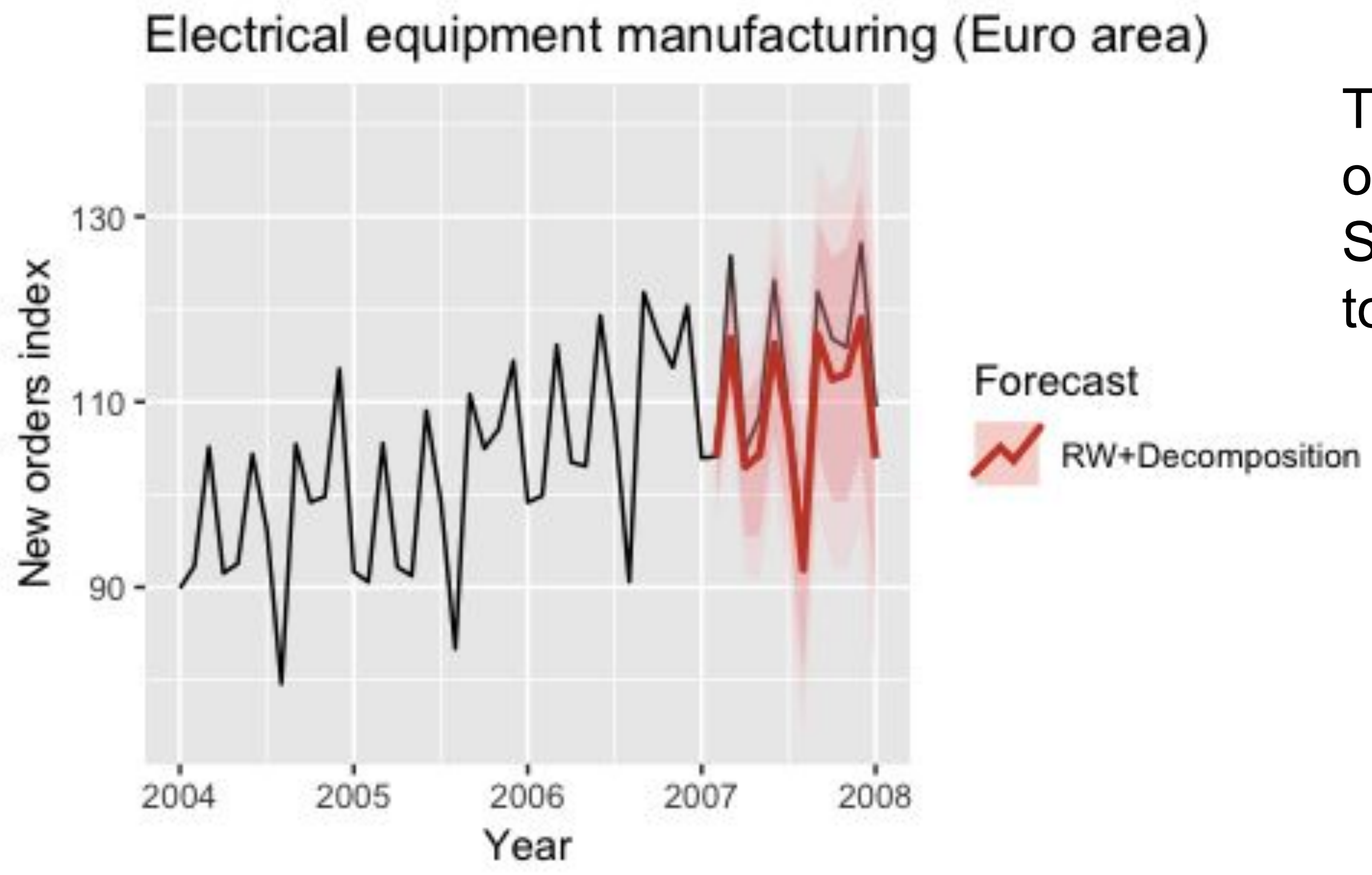


Seasonal decomposition (+ any model)

- One way to use the decomposition for forecasting purposes is the following:
  - **Decompose** the training time series with some decomposition algorithm (e.g. STL): Y(t)= S(t)+T(t)+R(t).
  - **Compute** the seasonally adjusted time series Y(t)-S(t). Used to forecast the evolution of the seasonally adjusted time series in a model.
  - **Add** to the forecasts the seasonality of the last time period in the time series (in our case, the fitted S(t) for last year).

In the following picture we show the seasonally adjusted industrial production index time series.

Seasonal decomposition (+ any model)



The following plot shows the predictions obtained for the year 2007 by using the STL decomposition and the naïve model to fit the seasonally adjusted time series.

Exponential smoothing

- Exponential smoothing is one of the most successful classical forecasting methods. In its basic form it is called simple exponential smoothing and its forecasts are given by:

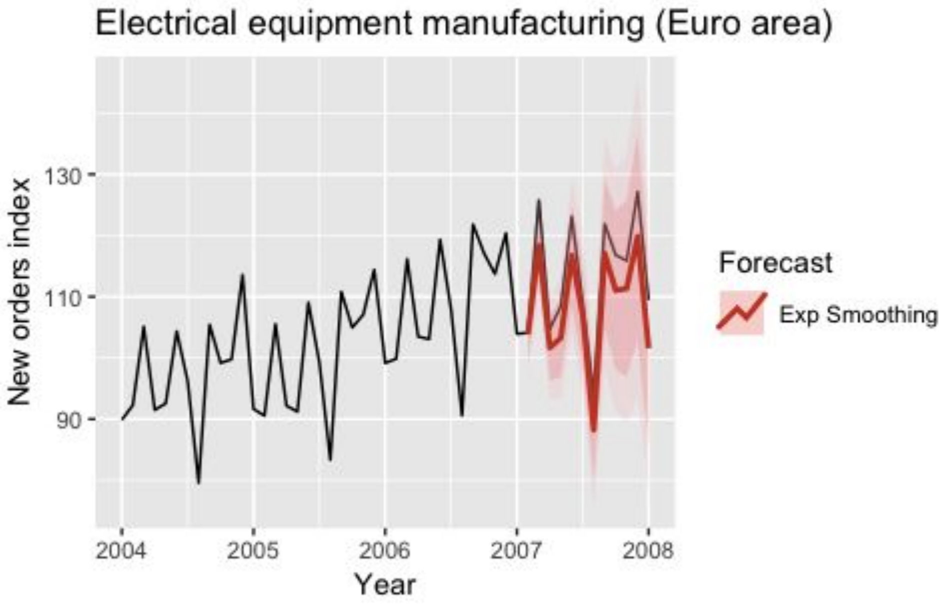
$$\hat{Y}(t+h|t) = \alpha y(t) + \alpha(1-\alpha)y(t-1) + \alpha(1-\alpha)^2 y(t-2) + \dots$$

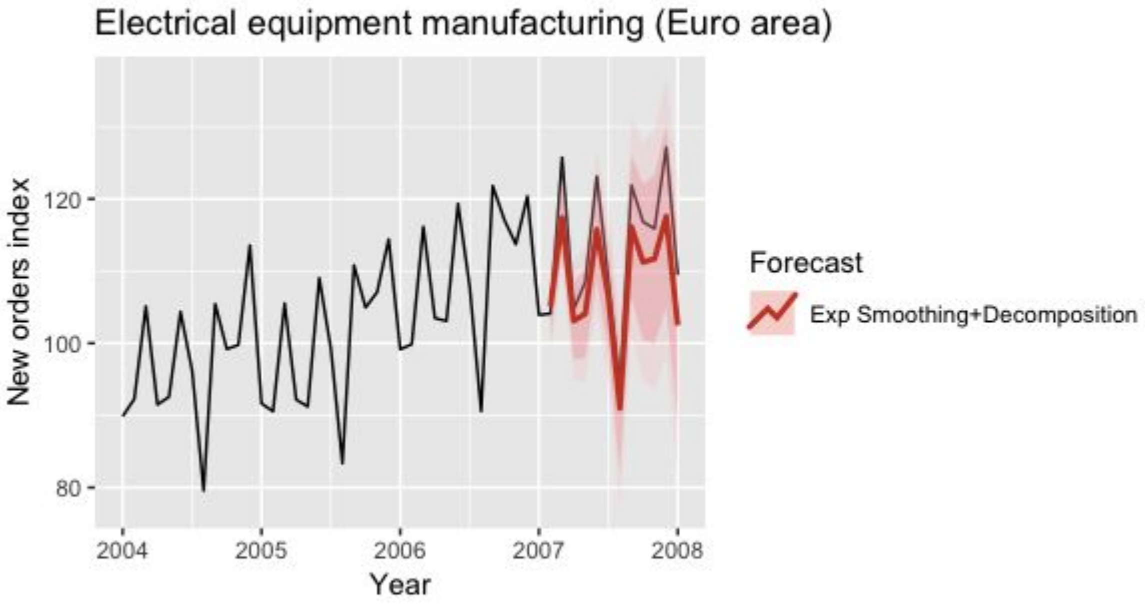
$$\text{with } 0<\alpha<1$$

- We can see that forecasts are equal to a weighted average of past observations and the corresponding weights decrease exponentially as we go back in time.

Exponential smoothing

The following plots show the predictions obtained for the year 2007 by using exponential smoothing models (automatically selected) to fit both the original and the seasonally adjusted time series.



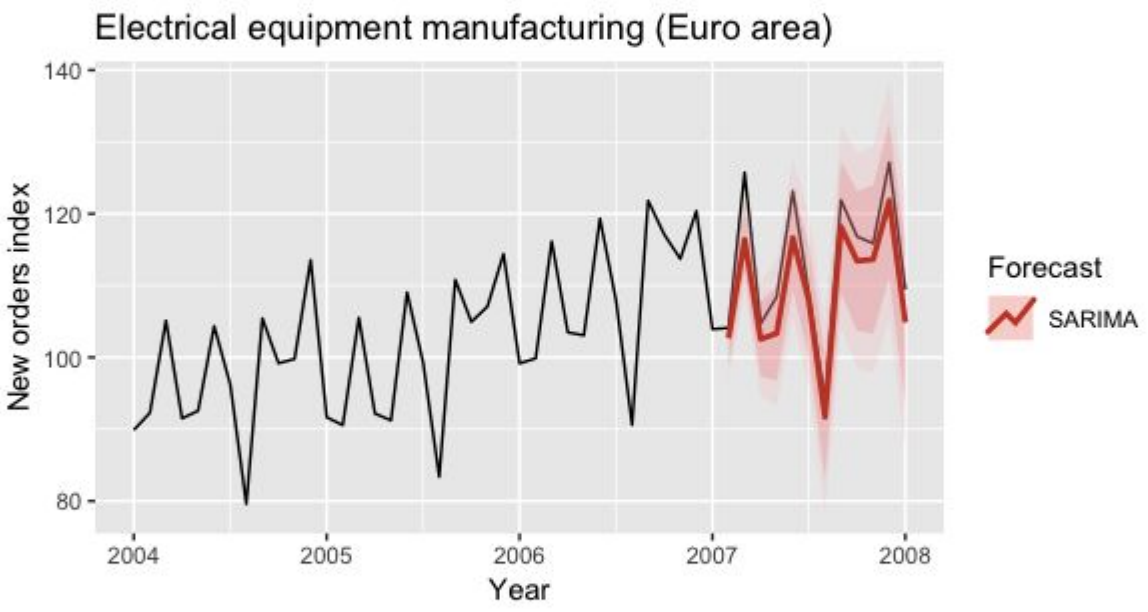


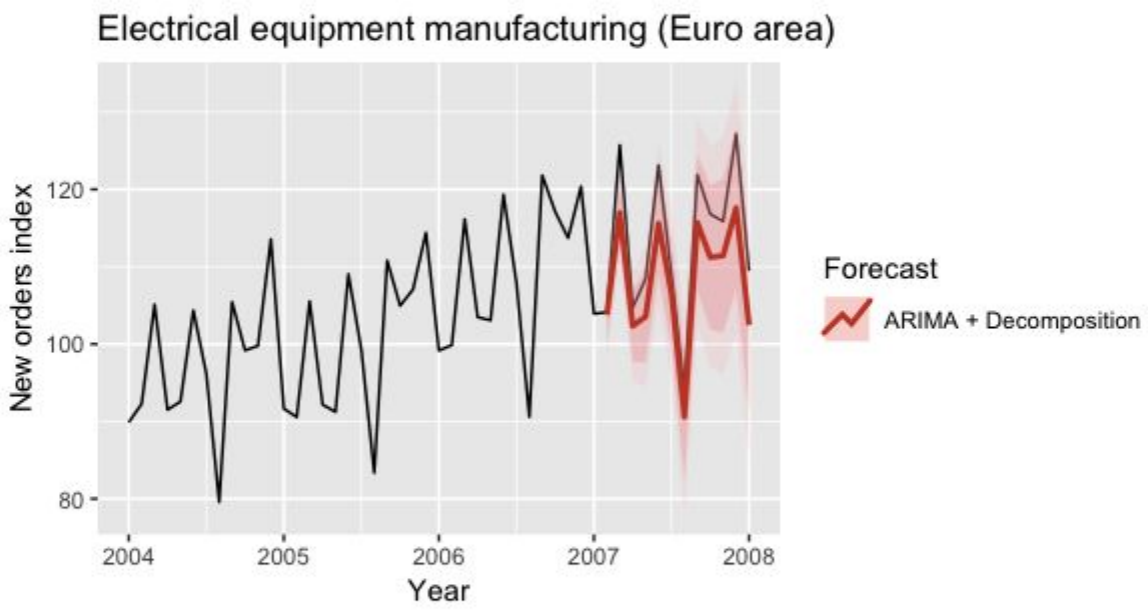
ARIMA, SARIMA

- most widely used approaches for time series forecasting
- **A**uto**R**egressive **I**ntegrated **M**oving **A**verage
- In an AutoRegressive model the forecasts correspond to a linear combination of past values of the variable. In a Moving Average model the forecasts correspond to a linear combination of past forecast errors.
- Basically, the ARIMA models combine these two approaches. Since they require the time series to be stationary, differencing (Integrating) the time series may be a necessary step, i.e. considering the time series of the differences instead of the original one.
- The **SARIMA** model (**S**easonal **ARIMA**) extends the ARIMA by adding a linear combination of seasonal past values and/or forecast errors.

ARIMA, SARIMA

The following plots show the predictions obtained for the year 2007 by using a SARIMA model and an ARIMA model on the seasonally adjusted time series.



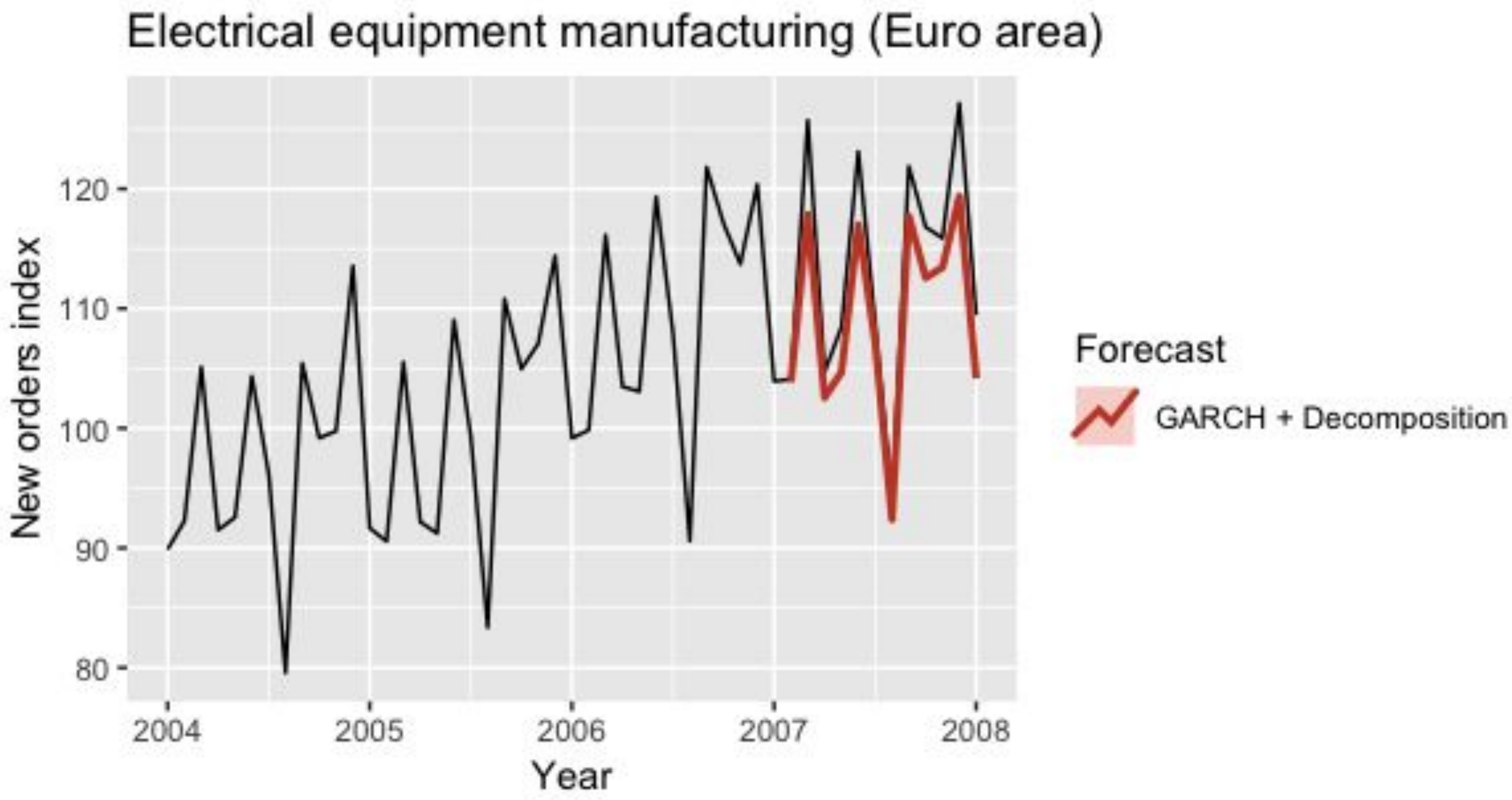


GARCH

- The previous models assumed that the error terms in the stochastic processes generating the time series were **homoskedastic**, i.e. with constant variance.

- Instead, the **GARCH** model assumes that the variance of the error terms follows an **A**uto**R**egressive **M**oving **A**verage (**ARMA**) process, therefore allowing it to change in time. It is particularly useful for modelling financial time series whose volatility changes across time. The name is an acronym for **G**eneralised **A**uto**R**egressive **C**onditional **H**eteroskedasticity.

GARCH



Electrical equipment manufacturing (Euro area)
New orders index
120
110
100
90
80
2004
2005
2006
2007
2008
Year
Forecast
GARCH + Decomposition

Dynamic linear models

- Dynamic linear models represent another class of models for time series forecasting. The idea is that at each time t these models correspond to a linear model, but the regression coefficients change in time.
- An example of dynamic linear model is given below:

$$\mathbf{y(t) = \alpha(t) + t\beta(t) + w(t)}$$

$$\mathbf{\alpha(t) = \alpha(t-1) + m(t)}$$

$$\mathbf{\beta(t) = \beta(t-1) + r(t)}$$

$$\mathbf{w(t) \sim N(0,W)\ ,\ m(t) \sim N(0,M)\ ,\ r(t) \sim N(0,R)}$$

- In the previous model the coefficients α(t) and β(t) follow a random walk process.
- Dynamic linear models can be naturally modelled in a Bayesian framework, however maximum likelihood estimation techniques are still available.

Dynamic linear models

The following plot shows the predictions obtained for the year 2007 by using a dynamic linear model to fit the seasonally adjusted time series

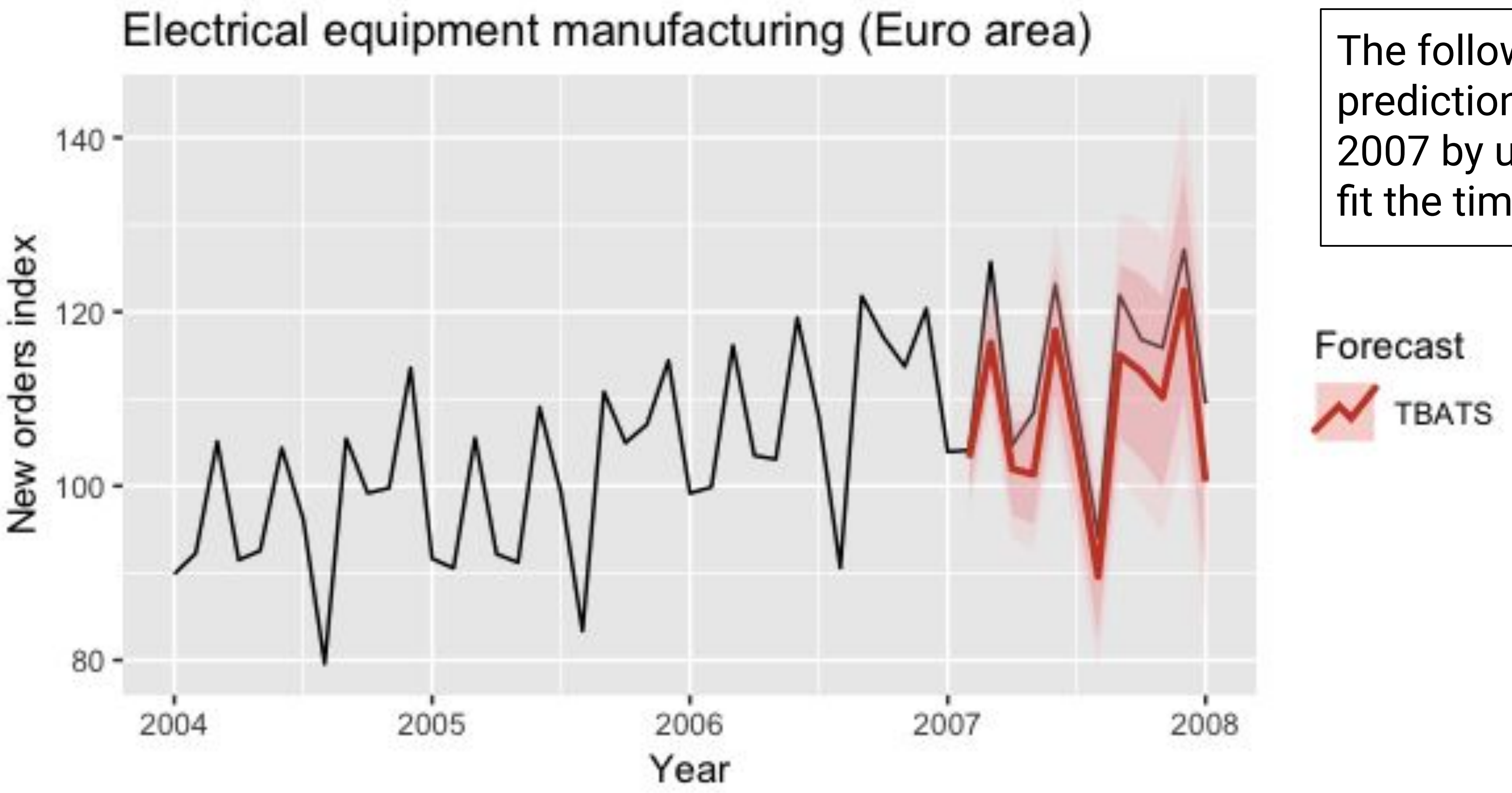
TBATs

- The **TBATS** model is a forecasting model based on exponential smoothing. The name is an acronym for **T**rigonometric, **B**ox-Cox transform, **A**RMA errors, **T**rend and **S**easonal components.

- The main feature is its capability to deal with multiple seasonalities by modelling each seasonality with a trigonometric representation based on Fourier series.

- A classic example of complex seasonality is given by daily observations of sales volumes which often have both weekly and yearly seasonality.

TBATs



The following plot shows the predictions obtained for the year 2007 by using a TBATS model to fit the time series.

Prophet

- Allows to deal with multiple seasonalities. It is an open source software released by Facebook's Core Data Science team.
- The prophet model assumes that the the time series can be decomposed as follows:

$$\mathbf{y(t) = g(t) + s(t) + h(t) + \varepsilon(t)}$$

  The three terms g(t), s(t) and h(t) correspond respectively to trend, seasonality and holiday. The last term is the error term.
- The model fitting is framed as a curve-fitting exercise, therefore it does not explicitly take into account the temporal dependence structure in the data. This also allows to have **irregularly spaced observations**.

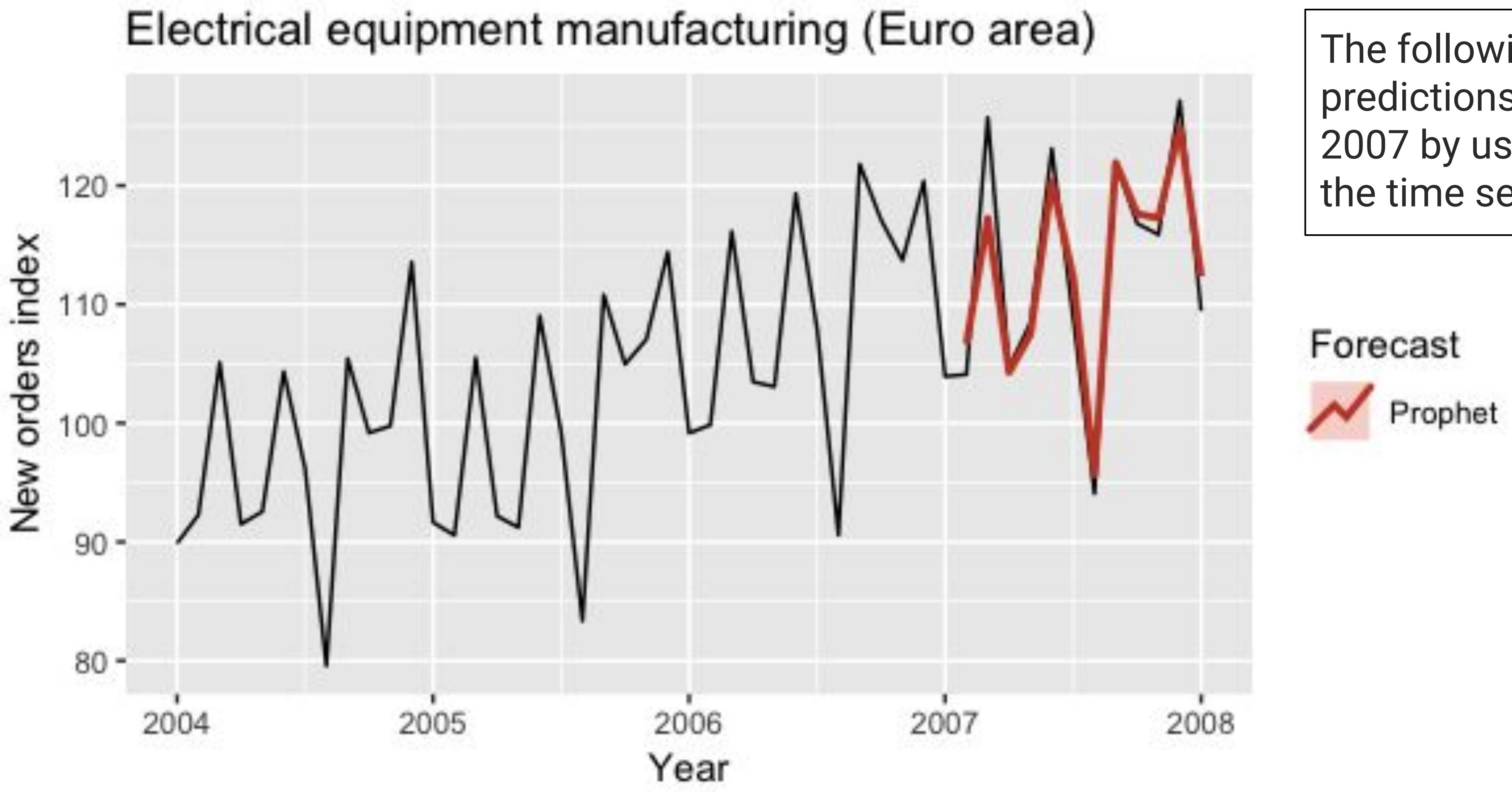
Prophet

- There are two options for trend time series:
  - a saturating growth model,
  - and a piecewise linear model.

- The multi-period seasonality model relies on Fourier series. The effect of known as **custom holydays** and can be easily incorporated into the model.

- The prophet model is inserted in a Bayesian framework and it allows to make full posterior inference to include model parameter uncertainty in the forecast uncertainty.

Prophet



The following plot shows the predictions obtained for the year 2007 by using a Prophet model to fit the time series.

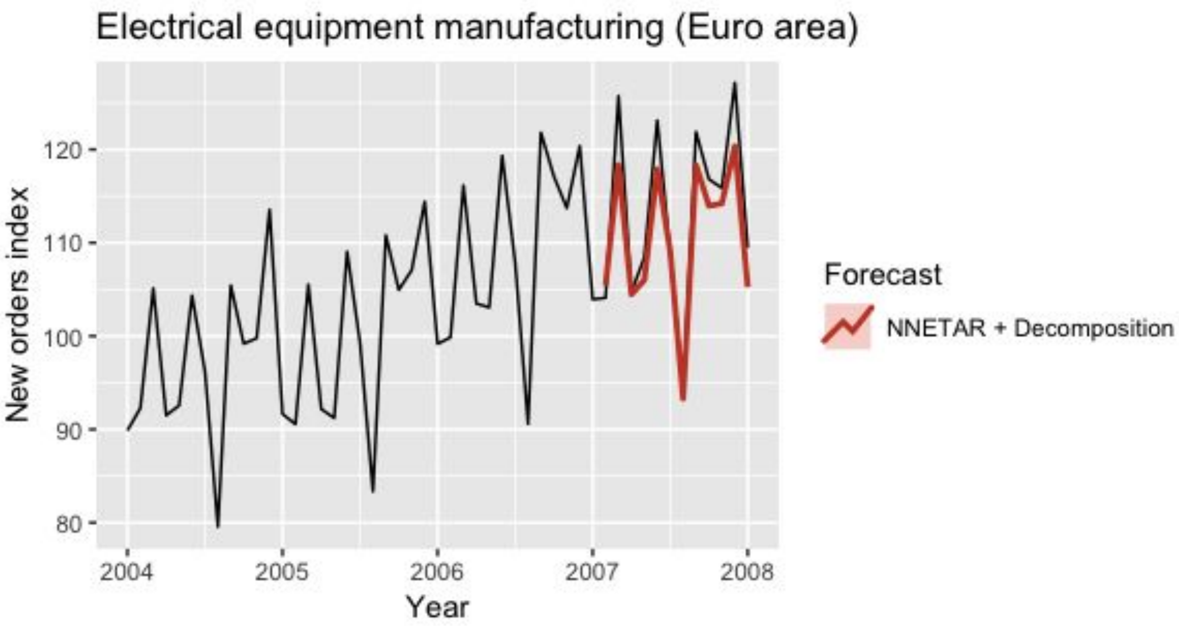
NNETAR

- The **NNETAR** model is a fully connected neural network. The acronym stands for **N**eural **NET**work **A**uto**R**egression.

- The NNETAR model takes in input the last elements of the sequence up to time *t* and outputs the forecasted value at time *t+1*. To perform multi-steps forecasts the network is applied iteratively.

- In presence of seasonality, the input may include also the seasonally lagged time series.

NNETAR



The following plots show the predictions obtained for the year 2007 obtained by using a NNETAR model with seasonally lagged input and a NNETAR model on the seasonally adjusted time series.

LSTM

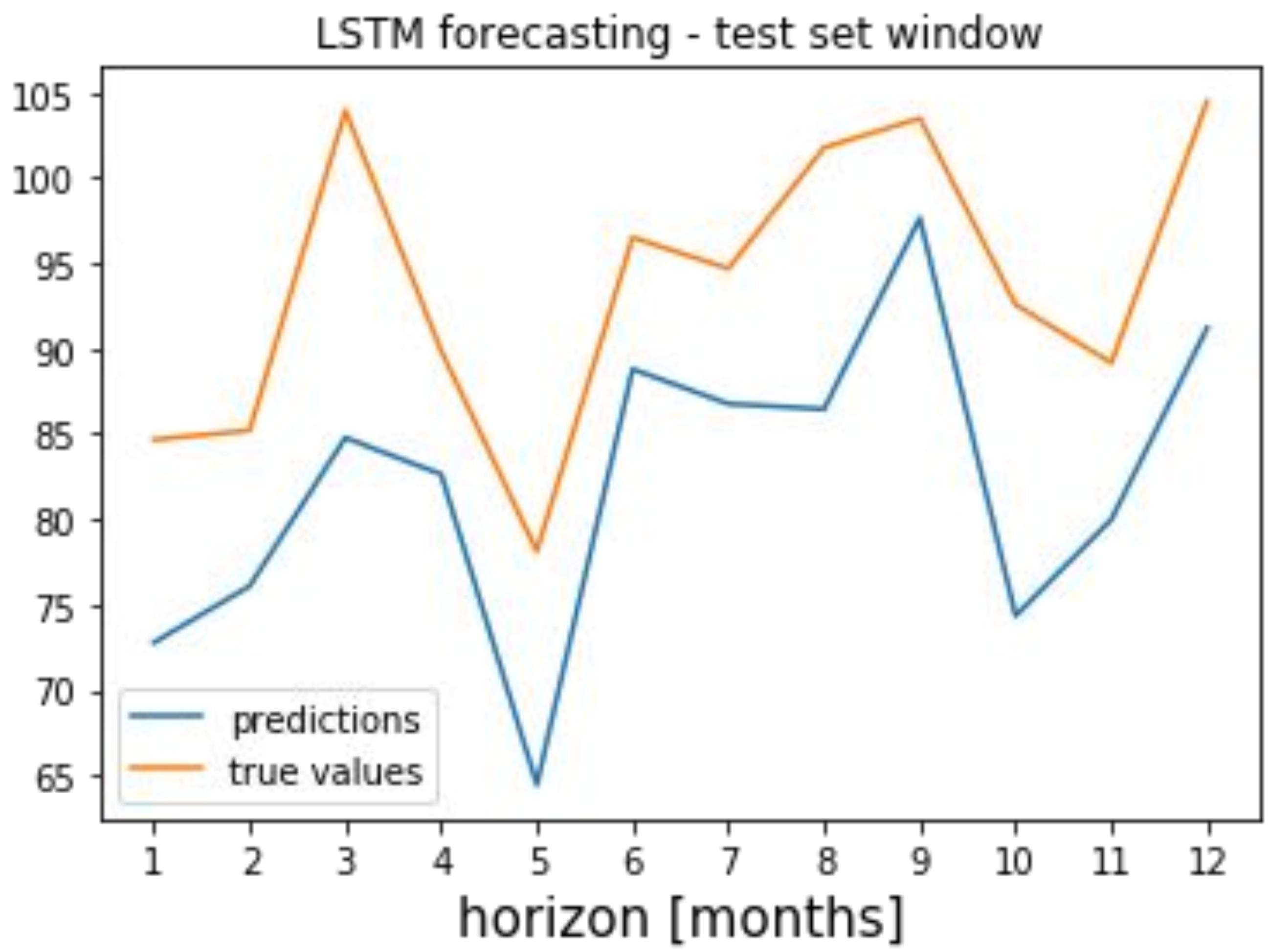
- LSTM models can be used to forecast time series (as well as other Recurrent Neural Networks).

- LSTM is an acronym that stands for **L**ong-**S**hort **T**erm **M**emories.

- The state of an LSTM network is represented through a **state space vector**. This technique allows to keep tracks of dependencies of new observations with past ones (even very far ones).

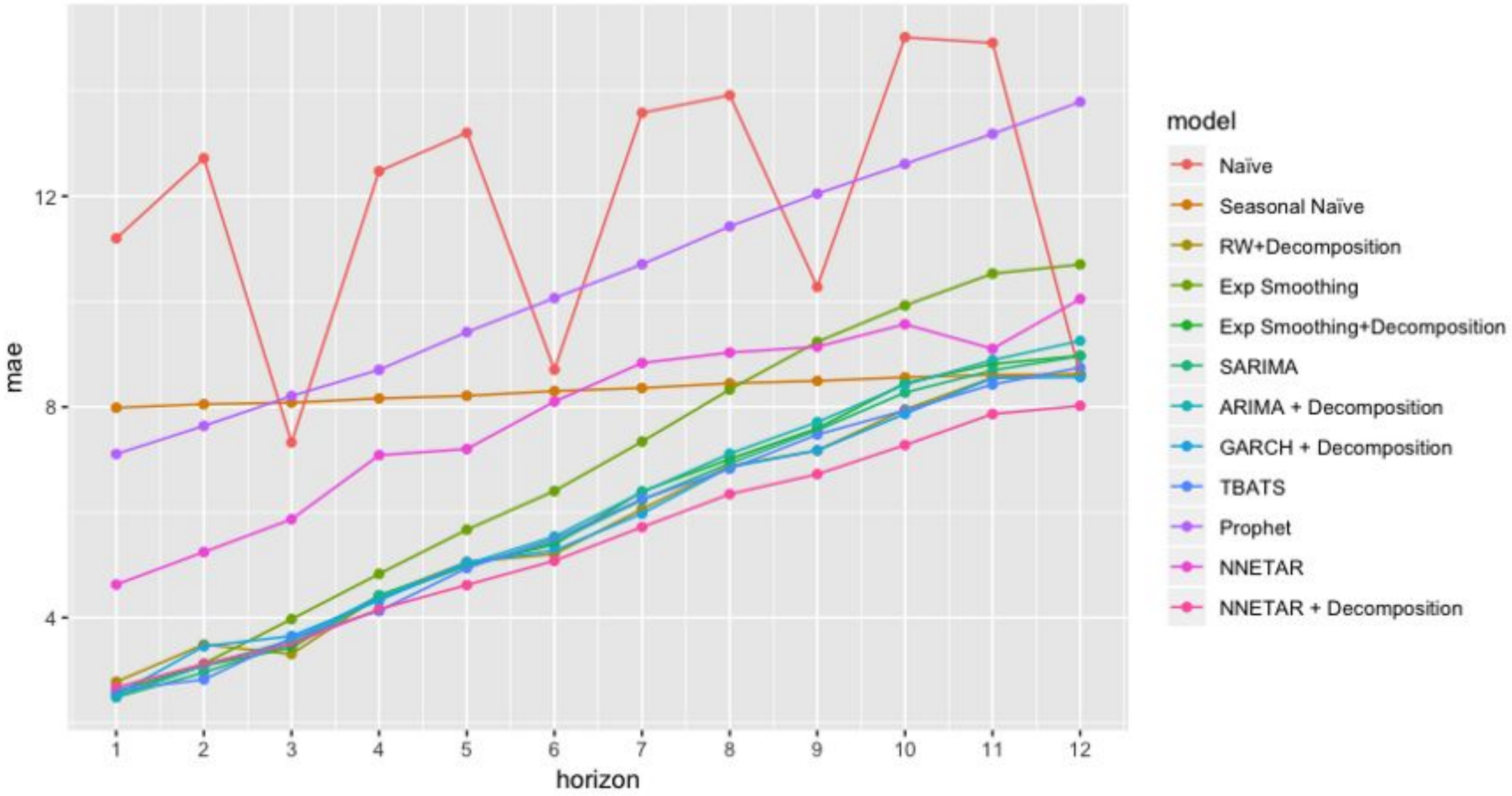
- LSTMs are complex models and they are rarely used for predicting a single time-series, because they require a large amount of data to be estimated. However, they are commonly used when predictions are needed for a large number of time-series

LSTM



The following plot shows the predictions for the first year in the test set obtained by fitting a LSTM model on the seasonally adjusted time series.

Comparison



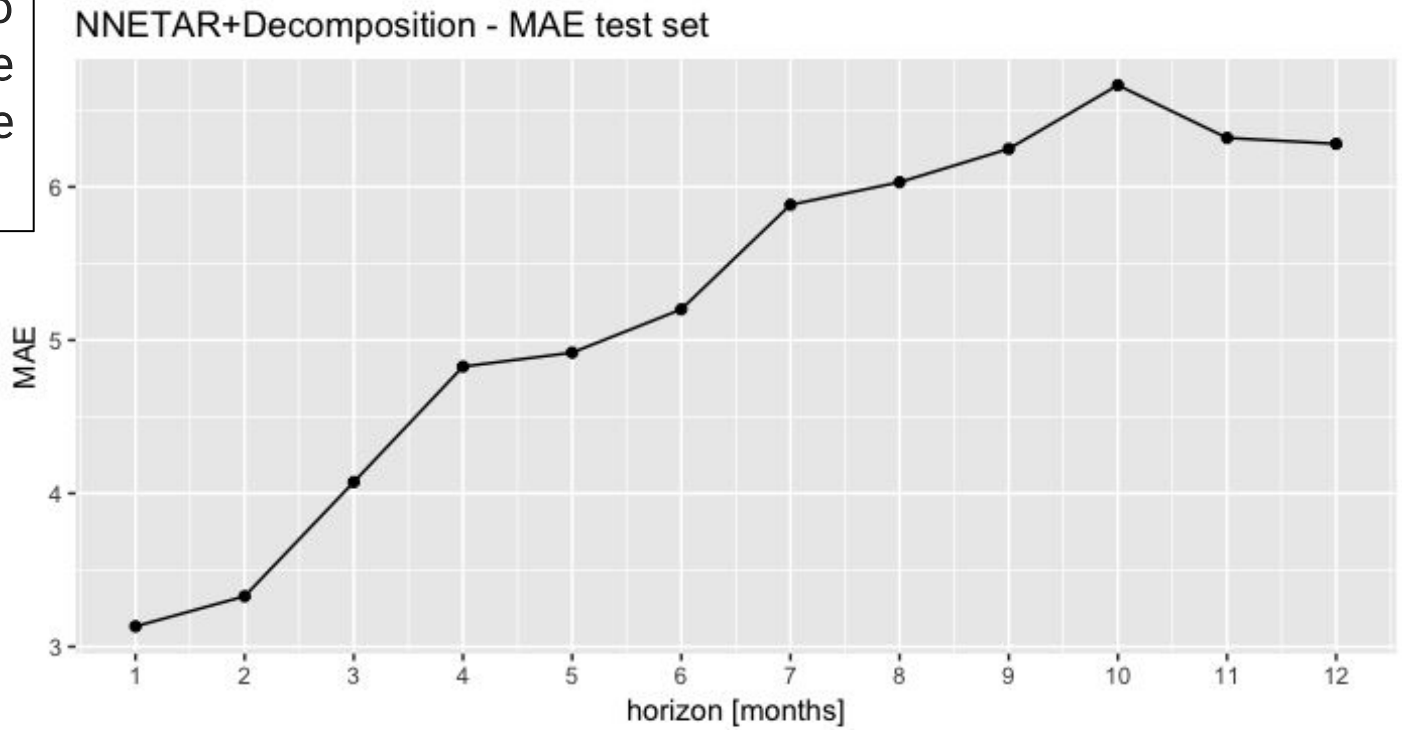
We can see that, for time horizons **greater than 4**, the NNETAR model on the seasonally adjusted data performed better than the others.

Overall Mean Absolute Error computed by averaging over different time horizons

```
"Naïve : 11.83"
"Seasonal Naïve : 8.32"
"RW+Decomposition : 5.79"
"Exp Smoothing : 6.88"
"Exp Smoothing+Decomposition : 5.93"
"SARIMA : 5.86"
"ARIMA + Decomposition : 5.99"
"GARCH + Decomposition : 5.77"
"TBATS : 5.77"
"Prophet : 10.41"
"NNETAR : 7.82"
"NNETAR + Decomposition : 5.43"
```

The **NNETAR model on the seasonally adjusted** data was the **best model** for this application since it corresponded to the lowest cross-validated MAE.

To get an **unbiased estimation** of the best model performance, we computed the MAE on the test set, obtaining an estimate equal to **5,24**. In the following picture we can see the MAE estimated on the test set for each time horizon.



How to further improve performance

Other techniques to increase models performance could be:

- Using different models for different time horizons
- Combining multiple forecasts (e.g. considering the average prediction)
- Bootstrap Aggregating

How to further improve performance

Other techniques to increase models performance could be:

- Using different models for different time horizons
- Combining multiple forecasts (e.g. considering the average prediction)
- Bootstrap Aggregating

Bootstrap Aggregation

1. **Decompose** the original time series
2. **Generate** a set of similar time series by randomly shuffling chunks of the Remainder component
3. **Fit** a model on each time series
4. **Average** forecasts of every model

innovate achieve lead

BITS Pilani
Pilani Campus

Thank You