Method Description

General Information

Type of Entry (*Academic, Practitioner, Researcher, Student*)	**Student**
First Name	**Nikoletta-Zampeta**
Last Name	**Legaki**
Country	**Greece**
Type of Affiliation (*University, Company-Organization, Individual*)	**University**
Affiliation	**National Technical University of Athens, Forecasting & Strategy Unit**

Team Members (*if applicable*):

1st Member	
First Name	Nikoletta-Zampeta
Last Name	Legaki
Country	Greece
Affiliation	National Technical University of Athens
2nd Member	
First Name	Aikaterini
Last Name	Koutsouri
Country	Greece
Affiliation	National Technical University of Athens

Information about the method utilized

Name of Method	**Theta - BoxCox**
Type of Method (*Statistical, Machine Learning, Combination, Other*)	**Statistical**
Short Description (up to 200 words)	**Deseasonalize data, apply Box-Cox Transformation and forecast with Theta**

Extended Description:

Our methodology is composed of three discrete steps in order to produce the final forecasts. Results were produced with the programming language R using the statistical packages *"forecast" (v 8.3)* and *"M4comp2018"*. Source code is available in GitHub, according to the instructions. More precisely, our methodology goes as follows:

1. Import the dataset: (M4dataset from GitHub)
2. Decompose the time series (if needed) to estimate the seasonal component: An autocorrelation test is applied for all the series of the Competition (non-seasonal data, e.g. yearly ones are excluded), using the *"acf()"* function of R. In case of a positive result (90% confidence), we strip the series off their seasonal component,

using the classical multiplicative decomposition, and obtain the seasonally adjusted data. Function *"decompose()"* was used for the calculation of the seasonal component.

3. Apply the Box-Cox transformation: On its bases, the Theta method (that will be used for extrapolating the series) generates linear forecasts. In order this simple forecasting model to produce reasonable results, even in cases of nonlinear trend, we adjust the historical data (without the seasonal component) using the Box-Cox transformation. The transformation depends on the parameter λ [0,1] and is defined as follows:

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

   For the transformation and the calculations, the functions: "BoxCox.lambda()" and *"BoxCox()"* are used.

4. Generate forecasts based on the **theta** method: The forecasts of each seasonally adjusted and normalized series are produced using the Theta method, exploiting the R- function "*thetaf ()"*. The respective forecast horizon is used per case.

5. Reverse Box-Cox transformation: Having generated our forecast, the data are rescaled using the reverse Box-Cox transformation, which is defined as:

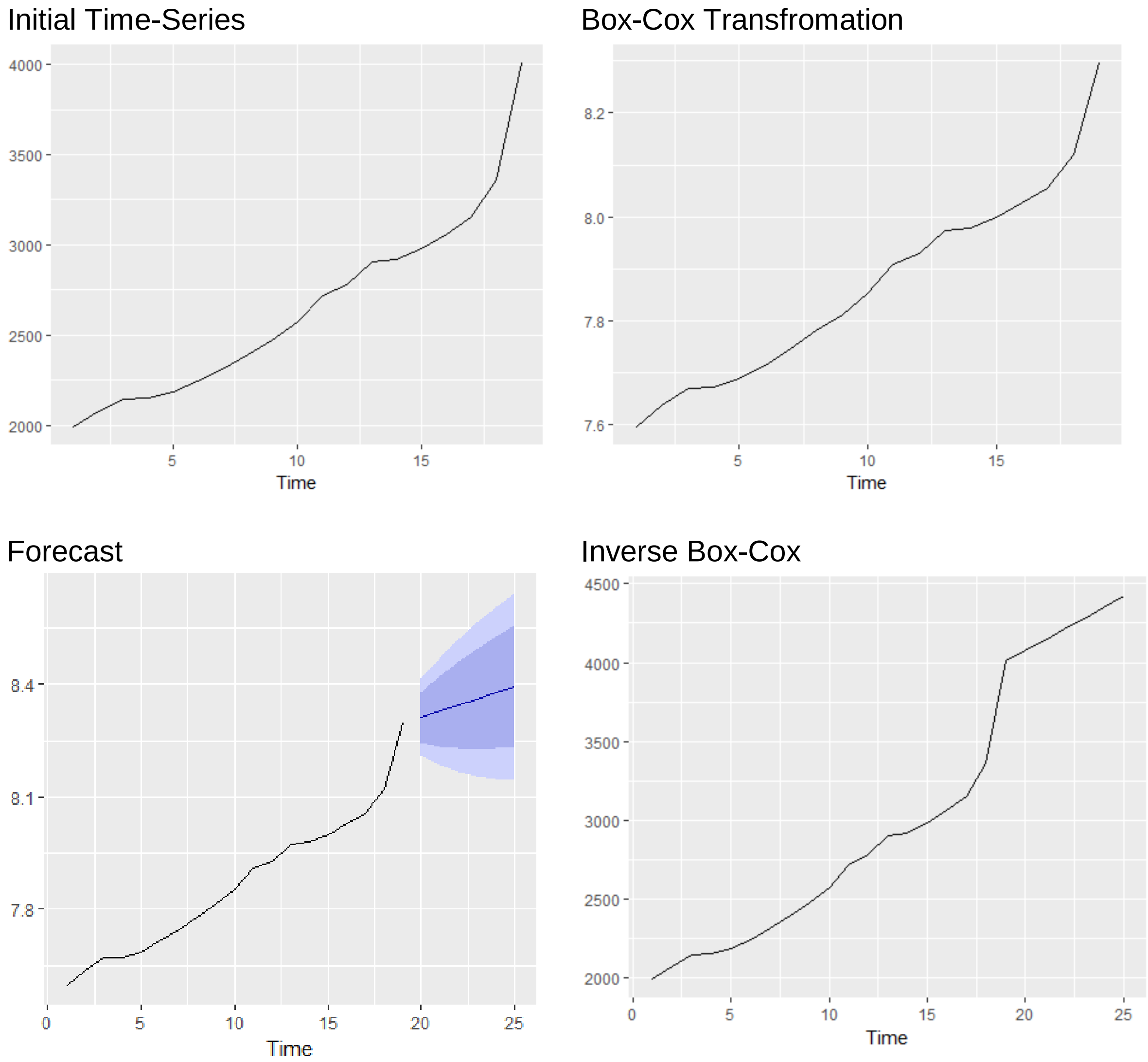
$$y_t = \begin{cases} \exp(w_t) & \lambda = 0; \\ (\lambda w_t + 1)^{1/\lambda} & \text{otherwise.} \end{cases}$$

   For the reverse transformation and the calculations, the functions: *"InvBoxCox()"* is used with the respective parameters that have been calculated in step 3.

6. Re-seasonalization: The scaled forecasts of step 5 are seasonally adjusted using the respective seasonal indices calculated in step 2, if applicable.

Final results are obtained in a single dataset, as proposed. A graphical example of the methodology is presented below for yearly time series:

Example in Yearly Time-Series


Initial Time-Series
4000
3500
3000
2500
2000
5
10
15
Time
Box-Cox Transfromation
8.2
8.0
7.8
7.6
5
10
15
Time
Forecast
8.4
8.1
7.8
0
5
10
15
20
25
Time
Inverse Box-Cox
4500
4000
3500
3000
2500
2000
0
5
10
15
20
25
Time