Demand Forecasting – Lab session

Mines Saint-Étienne, master GI, 2016 - 2017

The aim of the lab session is to perform and compare various forecasts for various time series. The methods described during the class are already implemented in Matlab. The datasets `data1.mat`, ..., `data4.mat` and the m-files can be found in the file `functions_matlab.zip`. After unzipping this folder, choose it as your working directory.

Question 1: Each dataset contains two vectors: `T` for the timepoints and `Y` for the time series. Load the various data sets using the command `load('data1.mat')` and plot the time series. For each one, can you recognize some of the features we have seen during the class?

Question 2: For each method (see below), try various parameters configuration in order to get an insight on their influence.

Question 3: Write a function that computes the Mean Square Error of the prediction.

Question 4: For each dataset, find a well suited method (of combination of method) with appropriate parameters values. Compute the MSE and plot the residuals to see if any pattern can be distinguished.

Description of the implemented forecasting methods

Let `T` denote the vector of time points where we have observations and `Y` be the vector of observed values.

Moving average `moving_average(T,Y,l)`

Arguments	`l` length of the window (oriented toward the past)
Output	a list containing the prediction locations and the forecast values

Example

```
l = 2;
[Tpred,Ypred] = moving_average(T,Y,l);
plot(Tpred,Ypred,'*');
```

Weighted moving average `weighted_moving_average(T,Y,l)`

Arguments `w` vector of weights

Output a list containing the prediction locations and the forecast values

Example

```
w = [0.1,0.5,1,2];
[Tpred,Ypred] = weighted_moving_average(T,Y,w);
plot(Tpred,Ypred,'o');
```

Exponential smoothing: `exponential_smoothing(T,Y,alpha)`

Arguments `alpha` is the smoothing parameter

Output a list containing the prediction locations and the forecast values

Example

```
alpha = 0.7;
[Tpred,Ypred] = exponential_smoothing(T,Y,alpha);
plot(Tpred,Ypred,'x');
```

Linear regression: `linear_regression(Tpred,T,Y)`

Arguments `Tpred` is the vector of prediction points

Output a list containing the prediction locations and the forecast values

Example

```
n = length(T);
Tpred = vertcat(T,2*T(n)-T(n-1));
[Tpred,Ypred] = linear_regression(Tpred,T,Y);
plot(Tpred,Ypred,'.');
```

As it is, the basis functions correspond to a constant and a linear function. Feel free to modify the m-file to add new ones!