Lab 6: Data Mining.
Time series

---

1. Time series

Time series analysis aims to model and explain a time-dependent series of data points. The analysis is usually based on statistical techniques. Time series appear in various domains and can be of various types. Typical examples are: daily temperature recorded during one month/year, daily exchange rate, prices of stocks etc.

There are various processing tasks on timeseries:

- *Visualization:*
  - Analyze the evolution in time of the measure (R: for ts objects one can use ts.plot or plot.ts; the two functions are different only in the way they deal with multiple series)
- *Pre-processing*
  - Fill in missing values (simplest approach: linear interpolation)
- *Analysis:*
  - Trend
    - it can be identified by removing the noise using *smoothing*:
      - Moving Average (in R one can use the function filter to define the smoothing filter)
      - Exponential Smoothing
    - it can be removed by *differencing*: the detrended timeseries can be obtained from the initial one by computing the difference between the neighboring values (this procedure can be recursively repeated for high order differencing); in practice the first order differences allows to remove linear trends, while second order differences are used to remove quadratic trend). R: function diff
  - Seasonal component
    - it can be identified by computing the average of the values corresponding to one period. For instance for data which are recorded monthly and there is an yearly seasonal effect (similar pattern of values for each year) one can compute the average corresponding to each month (average of values recorded in each January, average of values recorded in each February etc.)
    - it can be removed by computing differences between values corresponding to a time delay equal to the period. For instance for monthly data having a seasonal component at the level of one year, the time delay (lag) used in the removal of the seasonal component is L=12.
  - Noise, residuals
    - It can be obtained from the initial timeseries by removing the trend and the seasonal component
- Stationarity: a timeseries is (weakly) stationary if its local statistical characteristics do not change in time (nu există tendință și nici caracter sezonier). A nonstationary series can be transformed in a stationary one by corresponding differencing.

Rmk: the 3 components of a timeseries (trend, seasonal component and noise) can be obtained in R using decompose.

- *Forecasting*: estimate future values based on the current and previous values in the timeseries (based on a model describing the dependence between the current value and previous ones). Main steps in forecasting:
  - remove the trend and the seasonal components
  - chose a model and estimate its parameters based on the data in the timeseries
    - ARMA(p,q) – for stationary timeseries
    - ARIMA(p,d,q) – for nonstationary timeseries

Rmk:

- The quality of a model depends on:
  - Its accuracy (e.g. sum of squared errors)
  - Its complexity (number of parameters which should be inferred from data)
- If two models are similar with respect to their accuracy then the simplest model is preferred (a quantitative measure can be the Akaike Information Criterion – AIC- which is smaller for better models).

Rmk: The forecasting can be interpreted as a regression problem, characterized by:

- *Input:* The input data are previous values in a time series
- *Output:* The result is the next value(s) in the series
- *Model:* a regression model describing the relationship between the current value in the timeseries and several previous values (the number of previous values which are considered to have an influence on the current value is the so-called *time-lag*)

Let suppose that the we have a timeseries $X_1, X_2, \ldots, X_n$ and let us consider that T is the time lag. Thus it is considered that a current value x_i depends on $X_{i-1}, X_{i-2}, \ldots, X_{i-T}$. Therefore the sequence of values in the timeseries can be transformed in another dataset having T attributes playing the role of predictors and an attribute playing the role of predicted value:

T predictor attributes	*attribute to be predicted*
$X_1\ X_2\ \ldots X_i \ldots X_T$	X_{T+1}
$X_2\ X_3\ \ldots X_{i+1} \ldots X_{T+1}$	X_{T+2}
…	
$X_{n-T}\ X_{n-T+1}\ \ldots X_{n-i} \ldots X_{n-1}$	X_n

Based on the dataset containing the lagged attributes one can construct a regression model (in the same way as in the case of non-temporal data). One of the main issues is the choice of the time lag

Exercise 1. /R Run the commands in Ex1.r.

Exercise 2./R Run the commands in Ex2.r.

Exercise 1. /Weka

a) Open the file airlines.arff (containing the monthly passenger numbers for an airline for the years 1949 – 1960)
b) Construct a new dataset by using a time lag of 12. Hint: you can use the Weka editor, Excel or any text editor

c) Apply a linear regression model to the new dataset and predict the next value in the timeseries

Exercise 2./Weka (optional – only for Weka versions >=3.7.3)

a) Install the Time Series Forecasting package by using Weka GUI Chooser ->Tools->Package manager and by selecting for installation timeSeriesForecasting
b) Open the file airlines.arff
c) Forecast the next 6 values by using one of the following models for regression: (i) linear regression; (ii) multilayer perceptron; (iii) random forests. Hint: the selection of the model is done by using the Advanced Configuration->Based Learner panel

Remarks: details on the package TimeSeriesForecasting can be found at http://wiki.pentaho.com/display/DATAMINING/Time+Series+Analysis+and+Forecasting+with+Weka.

Homework.

1. Write R functions to transform a timeseries by:
    a. moving average smoothing – the window size will be a parameter of the function
    b. exponential smoothing – the smoothing coefficient will be a parameter of the function
    c. differencing – the differencing order will be a parameter of the function

Rmk. The values at the timeseries extremities will be processed by adjusting correspondingly the moving window size .