UPPSALA UNIVERSITY
DEPT OF INFORMATION TECHNOLOGY DIVISION OF SYSTEMS AND CONTROL
last rev. September 22, 2014

Empirical Modelling of Some Processes 2014 - Project Description

Abstract

The purpose of this project is to illustrate how empirical modelling/system identification can be applied model systems. The default project has a focus on Energy systems; estimation of a wind speed spectrum, modelling of a small scale solar heating system. In the project you should also study a "new method". Several possible extra tasks are also outlined including prediction of the heat demand in a suburb of Uppsala. See also the course program for grading of the project.

Contents

1 Introduction

This project aims to illustrate how system identification can be used to obtain information about the dynamics of a system. Two energy systems are used for illustrations. The data used in this project includes wind speed data from wind turbines in Gotland and heat storage temperature of a solar heating system. In the final task a new method for modelling is studied.

Since the data sets come from real systems, the modelling procedure may not be as ”convenient” as the case with fabricated data. No model structure can imitate the real system to perfection, which makes this project resemble a real-life modelling task.

2 Project overview

2.1 General description

Sun and wind are two examples of relatively safe and clean renewable energy sources. Unlike fossil fuels, there is no risk of these energy sources ending in a foreseeable future. However, neither wind nor solar insolation can be stored and used at a later time; the energy that is not utilized immediately is lost. This causes fluctuations in electricity/heat production depending on the weather situation.

For a wind turbine there is also the additional problem of stochastic rapid changes in wind speed, also known as turbulence. When designing wind turbines or when choosing suitable spots to place them, it is important to examine the characteristics of the wind, both in terms of average wind speed and turbulence.

In a similar way it can be useful to be able to predict how much energy can be expected from a solar heating system. Knowledge of expected system behaviour is used when deciding whether to install such a system in your house, how it should be dimensioned, and to what extent an additional heat source will be needed. One way of achieving this information could be to use a computer model of the system.

In energy production (extra task) it is also important to make good forecasts of electricity consumption and heat consumption, so that sufficient effect can be provided to the customers. Besides daily and weekly variations, the demand could also be expected to be affected by weather conditions.

In the basic ”default” project you will study the following modelling problems:

- *Identification of wind speed dynamics.* Spectral estimation using parametric and non-parametric methods applied to wind speed data. The data have been provided by Dr. H. Bergström, Department of Earth Sciences, Uppsala University. Details are given in Section 5
- *Modelling of a solar heating system.* Performance comparison of different model structures (both black-box and grey-box) applied to data from a small scale solar heating system. The number of parameters in the grey-box model will be reduced and compared with the non-reduced model. Details are given in Section 6. The data have been provided by Prof. L. Ljung, Linköping University.
- *Study of a new method.* In this task you can choose between a number of methods to study. This sub project should preferably be documented in a separate report.

The procedure for finding appropriate models (model validation) is fundamental in the project and should be extensively documented.

The project should be solved in groups consisting of *4 students/group*. We can accept groups with less number of students but the maximum group size is four.

Data files for the project tasks can be downloaded from
the course homepage

The results are to be presented both in an oral presentation and a written report, see Section 3.

2.2 Supervision

When you need project help, write an email to the responsible supervisor describing your problem.

Project	Supervisor	Email
Wind speed	Johannes Nygren	Johannes.Nygren@it.uu.se
Solar heating system	John Nordstrand	john_nordstrand@hotmail.com
A new method	Johan Wågberg	johan.wagberg@it.uu.se
Extra tasks in Ch 10 and 11	Daniel Håkansson	Daniel.Hakansson.9265@student.uu.se
Other Extra tasks	Contact Bengt for info	bc@it.uu.se

3 Project examination

In order to pass the project the following is required:

- A final oral presentation of the project results. See timeEdit for schedule.
- A passed written report. The dead-line for submitting the project report is 2014-10-22 (23.59). Please **send the report in word or *pdf-format* to studentportalen**.
- The report should be corrected according to the supervisors' comments within three weeks after you received the comments.

The project report should be written carefully, in order to be understood by a person without prior knowledge of the project. Make sure to motivate your experiments by use of the theory of the course. The theory that you use should also be briefly summarised in the report since the reader cannot be expected to have memorized the literature used in the course. Summarize important findings in tables (for example model accuracy for different model orders) and illustrative plots. Make sure to describe what variables are plotted and in what units. Define all used variables.

Relevant **Matlab code** should also be provided in electronic form preferably in an Appendix to the report. Avoid sending Matlab code in a separate email.

Some general guidelines on how to write reports can be found on the web, e.g. at http://www.luth.se/depts/lib/rapport/index.html.
See also the instructions for writing a project report given out in the Control course. See http://www.it.uu.se/edu/course/homepage/empmod/vt10/SkrivaTekniskRapport.pdf

A typical layout of the project report may be:

1. Introduction
2. Basic description of the applications (wind turbines, solar heating systems,...) to give a context to the project assignments.
3. Task 1 - Identification of wind speed dynamics. Experimental design, modelling and validation, summary of used theory, evaluation and results.
4. Task 2 - Modelling of a solar heating system. Experimental design, modelling and validation, summary of used theory, evaluation and results.
5. Task 3 - A new method. This task may be written as a separate report. Outline: Abstract, Introduction (motivate the method, also give references to literature), "The method" (outline of the method), Illustration (use data to illustrate and compare the method), Conclusions. The report may have a length of around 5-10 pages
6. Extra task (optional).
7. Concluding discussion.
8. References.

4 Validation of models

See Ch 14 in the text book and the summary in the text for the individual task (note in particular the help on how to get a data structure to vector form and how FIT is defined)

5 Task 1 – Wind speed modelling and prediction

Data for the following task can be found in *vind.mat*.

5.1 Data description

The file *vind.mat* contains one matrix, *tid_hela*, and one vector y which contains information about wind speed as measured from a mast located some 150 meters apart near four wind turbines.

Variable	Content/Structure
tid_hela	Five-column matrix containing information on the **time** (year, month, day, hour and minute) for each data sample (on the corresponding row in the other matrices).
y	Vector containing the **wind speed**.

The wind speed data is given in m/s. The sampling time is 1 sec, which means that the first 60 rows of *tid_hela* are identical since the smallest time unit used in this matrix is minutes.

5.2 Task outline

Estimate the spectrum of the wind speed using nonparametric as well as parametric methods.

- Select a subset of 2000 samples that looks stationary (that is a subset with approximately constant mean and variance).
- **Remove** the mean from the data set. Note, that in order to be able to compare the parametric and non parametric methods the data set used in these estimation methods has to be pretreated in the same way (otherwise the estimates will differ for low frequencies). By removing the mean we have a data set adapted for standard models.
- Use a non parametric method, and compare the spectrum obtained with different choices of windows. Try to determine a window size that gives a spectral density estimate that corresponds to your feeling for how a wind spectral density should look like.
  Hint 1: A good feeling can be obtained from Example 10.3 ("Vindmodeller") in Ljung and Glad (2004). What does that example suggests: A low or high order of M (window size in spa)?
- Estimate the spectrum using an AR model. How does the order of the AR model affect the parametric estimate? Can you find an AR model that gives a similar spectral density estimate to what was selected with the non parametric method?
- Using the found AR models and the MATLAB function 'compare', make predictions of the wind speed 1, and 3 seconds ahead. How well can you predict the wind speed for the different prediction horizons? Present numerical results, for instance 'fit' that can be obtained from 'compare', in a table for a number of AR models. Finally, show that for a long prediction horizon, say 10 seconds, the prediction becomes basically useless.

5.3 Some hints

- You may want to divide the data in two sets.
- When importing data without an input signal (as in the winddata case) , the input signal in ident (in the box "Workspace variable") should be specified as an empty matrix "[]".
- For data without an input signal, the choice "noise spectrum" gives the spectrum in ident.

- For time series models (for example AR models), it is not possible to simulate the model since the model does not have any input signal. The model can only be used for prediction (one or several steps ahead).

6 Task 2 – A solar heating system

This task is performed with data from a small scale solar heating system (Figure 1). The idea is that the sun heats the air in the solar collector on the roof, a fan pumps the air into the heat storage from which the warm air can be used to heat the building. The data needed for this task is found in *solardat.mat*.

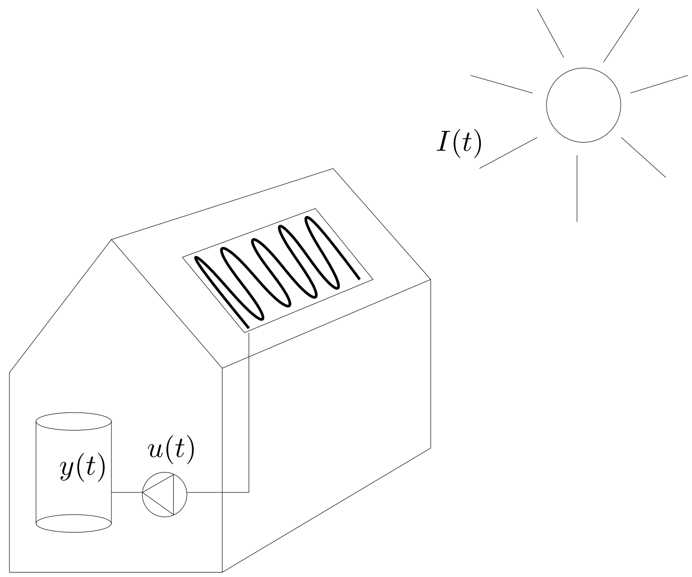

Figure 1: A schematic picture of the solar heating system, with the measured variables irradiance (solar insolation), $I(t)$, pump speed, $u(t)$, and heat storage temperature, $y(t)$.

6.1 Data description

The file *solardat.mat* contains information about *solar insolation*, *pump speed*, and *heat storage* for a small scale solar heating system. The matrices containing this information are named *sol* (solar insolation), *flakt* (fan speed), and *temp* (storage temperature). The sampling period is 10 minutes, and the experiment lasted for approximately 48 hours (in total there are 296 samples of each parameter).

6.2 Task outline

The first thing to do is to split the data set in two parts, and use the first 20 hours (120 samples) for calibration and the rest for validation.

6.2.1 Linear modelling

Find the least squares estimates of the coefficients a_i, b_i and c_i, $i = 1, 2$ for the model structure (a linear ARX-model)

$$y(t) + a_1 y(t-1) + a_2 y(t-2) = b_1 u(t-1) + b_2 u(t-2) + c_1 I(t-1) + c_2 I(t-2) + e(t)$$

using the calibration data. Also try a few other model orders and see if you can improve the model performance. Validate the **simulated** model on the second subset. In order to be able to compare with the modelling approach below, we recommend that you calculate the MSE (see below) for the models. If you use the Ident GUI you may need to export the model and the data to the Matlab command window in order to be able to calculate the MSE (drag the model to the 'To Workspace' box).

6.2.2 Grey-box modelling

Perform a mass balance calculation of the energy in the system and use this to find an alternative parameterization (see Ljung & Glad 2004, pp.369-374 and the Errata on the course home page).

Main task:

- Calibrate the grey-box model, and compare the performance to that of the linear model.
- Reduce the model order of the non-linear model by excluding one (or more) parameter at the time. You should make *a new estimation* of the model parameters for every reduced model order. Find a model order that give the best (in an MSE sense) simulation of the system.

To measure the model accuracy, use the MSE[1] of the prediction error

$$MSE = \frac{1}{N}\sum_{t=1}^{N}(y(t) - \hat{y}(t))^2, \tag{1}$$

where y is the measured output from the validation data and $\hat{y}$ is the **simulated** model output. It is recommended to write a m-function with input y and $\hat{y}$. Note that the summation can be done by vectors (like $X' * X$). Remember the scaling with $1/N$. Check your code carefully.

6.3 Comments and hints

When performing the *first part* of the task you should treat the data like you would at any black-box modelling, i.e. remove means etc. Note that if you want to use the Matlab/Ident functions for estimation, the above model needs to be treated as an ARX with *two* inputs. In Matlab this is achieved by the use of hard brackets $[u_1, u_2]$, so that a N by 2 matrix is imported to Ident as input signals (also the output signal needs to be imported). If you want to use the **ARX** function, the following example shows how to proceed: `ARX([y u_1 u_2],[3 2 1 1 1])`. This gives an estimate of an ARX model with $na = 3$, $nb_1 = 2$, $nb_2 = 1$ and a time delay of 1 for both u_1 and u_2.

For the second part of the task you will need to write your own code since the nonlinear model structures you will need are not available in the toolbox functions. When you write this code there are some things that are good to keep in mind. Before the identification the outdoor temperature should be removed, but since that is not given you can remove an estimated average outdoor temperature ($21°C$). As an extra (optional) experiment you may also remove some

[1]Note that MSE is not equal to the variance of the prediction error unless the mean value of the error is zero.

Table 1: Known and unknown parameters during the calibration and validation phases.

	Known quantities	Unknown quantity
Calibration	y, u, I	θ
Validation	u, I, θ	$\hat{y}$

sinusoidal function representing the daily temperature variations. However, the constant $21°C$ version should be used for the model reduction.

Note that in grey box modelling the parameters correspond to actual physical properties and therefore the means cannot automatically be removed, e.g. the pump is a binary signal where $u(t) = 0$ corresponds to the pump being turned off and $u(t) = 1$ to the pump being turned on. By removing the mean we would obtain a binary signal of (approximately) $u(t) = -0.5$ and $u(t) = 0.5$ respectively. The physical interpretation of this might be that air is pumped out of the heat storage at occasion, which is clearly not the case. Since the reparametrization includes division by $u(t)$ numerical problems will occur when $u(t) = 0$. This can easily be avoided by adding a small number (e.g. 0.01 or 0.05) to $u(t)$ before the reparametrization.

In the calibration you will need to calculate the θ that minimizes (in a least squares sense) the prediction error

$$\varepsilon(t) = y(t) - \varphi^T(t)\theta. \tag{2}$$

Here $\varphi(t)$ will be given by the model derivation. An easy way to obtain θ is to do as in Section 4.1 of Computer lab 1. However, you will need to form your own ϕ (φ) matrix. When doing that you may want to create sub-vectors containing the data corresponding to $y(t)$, $y(t-1)$, $u(t-2)$ etc. (depending on the model structure). Say *e.g.* that the model contains $y(t)$ to $y(t-3)$, you can then form

$$\begin{aligned} y(t) &= y(4:N) \\ y(t-1) &= y(3:(N-1)) \\ &\vdots \\ y(t-3) &= y(1:(N-3)) \end{aligned}$$

Note that the index of the first element in the vector is 1, and that it is the largest time delay used that determines how long the sub-vectors will be (cf. above time delay 3 gives vectors that are $N-3$ elements long). You may also need products of the type $y(t-1)u(t-2)$ in which case the element-wise multiplication and derivative ('.*' and './' respectively) will come in handy.

When validating the model you are required in this task to **simulate** the model output. Note that in doing so you must only use the measured output for initialization and for comparison when you have the full $\hat{y}$ vector. In a simulation $\hat{y}$ does not depend on the measured output y. The simulation of $\hat{y}(t)$ will contain old values of $\hat{y}$, like $\hat{y}(t-1)$ etc. you will need to write the simulation as a loop in Matlab. (If you do not write a loop, or if your loop contains the measured output, y, you will not obtain the simulated output).

One way of thinking about the calibration and validation is that the former is to estimate the model parameters (θ is unknown), whereas the validation is a means of testing the model (as if y was unknown). Table 1 shows what is known and unknown during the different phases of the identification.

It is important to remember that what can be considered a “good” model very much depends on the application. Sometimes the data is generated by a system that can be well described by the model structure, and sometimes you have to make simplifications. The amount of noise in the data also affects the size of the prediction error. In this task you can expect to get an MSE of approximately 10, at best, for the validation data set.

7 Task 3 – Study of a new method

Study one of the following methods (using a literature study and illustrations of the method in Matlab). Some start-up literature will be provided in SP.

- The instrumental variable (IV) method. IV is a method which gives an unbiased estimate (as the number of data points goes to infinity) even if the noise is coloured. This is in contrast to ARX modelling!. The parameters in IV method can be estimated analytically (in contrast to, for example, OE). In the project you should describe the method and compare ARX and IV both for the case when the data is white and the case when data is coloured. ARX and IV should be implemented in Matlab by yourselves. That is do not use command like ARX and IV since they contains tricks which make a fair comparison harder. Also not thet the backslash operator can NOT be used for the IV-method.

- Subspace identification. This is a relatively new method for estimating MIMO systems in state space form. It gives an interesting link to Reglerteknik II (where control of MIMO systems is a major topic). In the project you should describe the method and illustrate it using data from a simulated MIMO system (you may consider a system from Reglerteknik II). You can use n4sid for the estimation part.

- Ridge regression. Ridge regression is a "twist" on linear regression which can handle singular or close to singular problems. Generate data from a FIR model with poor excitation and compare ordinary regression with ridge regression. In particular you should study how well the models could predict the validation data.

- Neural Network modelling. A very popular black model is Artificial Neural Networks (ANN). Generate data from a nonlinear system and investigate how well ANN can describe the dynamics. This task is of particular interest to those who select the extra task described in Section 8.6.

8 Optional Tasks

You can choose to solve an additional problem. A successful completion of an optional task *may* increase the grade of the project. If you decide to perform an extra task, *please contact the course responsible teacher for advice and material.*

To many of the extra tasks, additional literature is available.

8.1 Predicting the sea level

The sea level can be regarded as the output from a dynamic system where the temperature is the input signal. The sea level is increased when the (mean) temperature is increased due to melting of ice and a decreased water density. In this sub project you will use data consisting of sea level (output, unit mm) and the mean temperature of the earth (input, unit C). The goal of the project is to estimate and analyse dynamic models. One example is to predict how much the sea level increases in steady state for a one degree increase in the mean temperature.

8.2 Use PCA (principal component analysis and/or PCR (principal component regression) on some data sets

See the guest lecture by Linda Åmand for details. An additional tutorial text is available on request. We recommend you to try to find some data by your own (it may not be from a technical system).

8.3 Estimation of a grey box model to wind data

Estimate the Von Karman spectra (see text book on p 236. This is a parametric model of the wind spectra using only two parameters. The idea is to first estimate the spectra of the wind with a non parametric method (etfe or spa) and the estimate the parameters of the wind model by a (nonlinear) least squares criterion. This can be done with fminsearch.

8.4 Recursive estimation of the wind dynamics

Estimate the wind dynamics using a recursive method. Implement by yourselves the recursive least squares method with forgetting factor for estimating AR parameters. Use the whole wind data set. Study how much the estimated parameters vary over time. Also calculate the spectra for a number of different times.

8.5 Output-Error modelling of the solar heating system

Identification of models (including the grey-box model) using an OE model. Since the criterion for evaluating the model is based on simulation (on the validation data) it seems reasonable that an OE model would be a better choice of model. Use fminsearch for the minimization.

8.6 Modelling of the solar heating system using Artificial Neural Networks

Use narx and noe models (available in the toolbox ident) for the data from solar heating system. The challenge: Get even better results than for the grey box model!

8.7 Prediction of electricity consumption in a city

This task is described in Section 10.

8.8 Prediction of heat consumption in a district heating system

This project has been created by Ingrid Budde (former STS-student) partly based on here Master thesis work in 2014. We also thank Vattenfall for proving the data for this task. This task is described in Section 11.

9 Remarks

Note that there are several possibilities to exchange for example Task 2 with another project. In case you are interested in any of these projects additional information can be provided. Three examples are

- Estimating the oxygen dynamics in an activated sludge process (Wastewater treatment)
  Students which has taken Reglerteknik II and the course Wastewater treatment have the possibility to chose a project combining knowledge from these courses. The project is to use a Kalman filter to estimate the oxygen transfer rate in a batch experiment. This problem is of high relevance in order to find how, for example, different detergents affect the wastewater.
- Stability monitoring in a Nuclear power plant
  Students which has a background and/or an interest in nuclear power engineering may replace the sub project "Modelling of a solar heating system" with a sub project devoted to modelling the stability margin in a nuclear power plant.
- Improve the sound from a loudspeaker by prefiltering (New project for 2014)
  Estimate the transfer function of a real loudspeaker and design a prefilter tries to improve the sound! (This project is new for 2014)

You are most welcome to suggest other optional tasks (discuss your proposal with the supervisors). This could, for example, include data from another application.

10 Optional Task – Prediction of electricity consumption

The goal is to predict (one step ahead) the electricity consumption as well as possible. (Hence do not evaluate your model by simulation instead use prediction)

10.0.1 Data description

The data for this problem can be found in *eldata.mat*. The data consists of electricity consumption, and outdoor temperature in Helsinki during 1986. The data is sampled every hour throughout the year (in total 8760 samples) and is stored in a MATLAB matrix (8760×11) called *helsinki*, where the eleven columns correspond to:

Column No.	Description
1	Year
2	Month
3	Day
4	Day of the week (1=Monday, 2=Tuesday, etc.)
5	Hour
6	Electricity consumption
7	Heat1
8	Heat2
9	Heat3
10	Outdoor temperature (°C)
11	Wind speed (m/s)

In other words the input signal is outdoor temperature ($helsinki(:,10)$), and the output is electricity consumption ($helsinki(:,6)$). Columns 1–5 can potentially be used to get a feeling for the daily and weekly consumption patterns. (Columns 7–9, and 11 can be completely ignored).

10.0.2 Task outline

First of all: It is recommended to only consider data from the winter period (it is not possible to calibrate an accurate model which cover all seasons).

Divide the chosen data into subsets for calibration and validation. Be careful to choose the subsets so that you get sufficiently exciting signals. Find a dynamic discrete time model that give a good prediction of the electricity consumption in Helsinki, using outdoor temperature (and old electricity consumption data). Compare different model structures (typically ARX and ARMAX), see section 4) and model orders. Validate the models carefully, so that you can motivate your choices in the report. Present numerical results (in tabular form) as well as figures for both the calibration and the validation.

To model daily variations a 24 hour differential operator, Δ_{24}, (applied to the output) can be useful. Similarly, there are variations that are repeated every week, so a Δ_{168} is also of interest (1 week = 7 days*24 hours = 168 hours). To include both weekly and daily variations in the model use $\Delta_{24}\Delta_{168}$.

$$\begin{aligned}
\Delta_{24} &= (1 - q^{-24}) \\
\Delta_{168} &= (1 - q^{-168}) \\
\Delta_{24}\Delta_{168} &= (1 - q^{-24})(1 - q^{-168}) = 1 - q^{-24} - q^{-168} + q^{-192}
\end{aligned}$$

The procedure is then:

1. Differentiate the output signal with $\Delta_{24}\Delta_{168}$
2. Estimate a model from the differentiated data. This will give a model predicting $z(t)$ where $z(t) = \Delta_{24}\Delta_{168}y(t)$
3. The original data is the predicted by $y(\hat{t}) = z(\hat{t}) + y(t-24) + y(t-168) - y(t-192)$

Also try to differentiate both the output and input signals with $\Delta_{24}\Delta_{168}$ and compare with the approach above (where only the output is differentiated). Also check how much the input signal contribute to the prediction capacity (that is also try models with no input signal)

11 Optional Task – Prediction of the heat consumption in a district heating (fjärrvärme) system

The goal with this task is to predict the heat consumption, or the heat load, one step ahead for a district heat system (hence do not evaluate your model by simulation, instead use prediction). The benefit of predicting the heat load in a district heating system is to optimize the production in the heat plant and to be sure that enough heat is produced during peak loads in the winter season.

11.0.3 Data description

The data for this problem can be found in *heatdata.mat*. The data consists of the district heat load for the urban area of Sävja in Uppsala and the outdoor temperature for Uppsala during 2012. The data is sampled every hour throughout the year (in total 8784 samples as 2012 was a leap year) and is stored in 3 MATLAB arrays 8784x1 named:

savja: Heat consumption (MW)
temp: Outdoor temperature (?C)
time: time and date (YYYY-MM-DD HH:MM)

11.0.4 Task outline

First of all: It is recommended to only consider data from the winter period (it is very hard to calibrate an accurate model which covers all seasons). One suggestion is to try January and February which gives about 1500 samples (it is not possible to calibrate an accurate model which covers all seasons).

Divide the chosen data into subsets for calibration and validation. Be careful to choose the subsets so that you get sufficiently exciting signals. Find a dynamic discrete time model that give a good prediction of the heat consumption in Sävja, using outdoor temperature (and old heat consumption data). Compare different model structures (typically AR, ARX and ARMAX) and model different orders. Validate the models carefully, so that you can motivate your choices in the report. Present numerical results (in tabular form) as well as figures for both the calibration and the validation.

Since the consumption also varies with the behaviour of the customers over the time of day and week it can be useful to differentiate the signals. To model daily variations a 24 hour differential operator, Δ_{24}, (applied to the output) can be useful. Similarly, there are variations that are repeated every week, so Δ_{168} is also of interest to use. (1 week = 7 days*24 hours = 168 hours). To include both weekly and daily variations in the model use $\Delta_{24}\Delta_{168}$.

$$\begin{aligned}
\Delta_{24} &= (1 - q^{-24}) \\
\Delta_{168} &= (1 - q^{-168}) \\
\Delta_{24}\Delta_{168} &= (1 - q^{-24})(1 - q^{-168}) = 1 - q^{-24} - q^{-168} + q^{-192}
\end{aligned}$$

The procedure (items 2-3 should be used for each model/model-order) is then:

1. Differentiate the output signal with $\Delta_{24}\Delta_{168}$. Use for example filter in Matlab.
2. Estimate a model from the differentiated data. This will give a model predicting $z(t)$ where $z(t) = \Delta_{24}\Delta_{168}y(t)$
3. The original data is the predicted by $\hat{y(t)} = \hat{z(t)} + y(t-24) + y(t-168) - y(t-192)$. Use different validation techniques to judge the model.

Also try to differentiate both the output and input signals with $\Delta_{24}\Delta_{168}$ and compare with the approach above (where only the output is differentiated). Also check how much the input signal contribute to the prediction capacity (that is also try models, like AR, with no input signal)

12 References

Ljung, L. and Glad, T. 2004. *Modellbygge och simulering, 2:a upplagan.* Studentlitteratur, Lund.