02427 Advanced Time Series Analysis

Computer exercise 1

This exercise starts with some modelling with non-linear models and afterwards methods for nonparametric estimation are used to identify functional dependencies.

Part 1

Simulate different non-linear models. Three good models to start out with are: SETAR(2;1;1), IGAR(2;1), and MMAR(2;1). Try different parameters. For reporting, write the models and discuss the most essential features of each model together with some informative plots.

Hints

R and Matlab: Check out the script `3dPlotting.R`. It starts out showing a simple way to implement a simulation of a process with a for-loop.

Part 2

Compute the theoretical conditional mean, $M(x) = E\{X_{t+1}|X_t = x\}$, for a SETAR(2,1,1) of your own choice.

Simulate 1000 values from the chosen SETAR model. Use these simulated data and a local regression model to estimate the $\widehat{M}(x) = E\{X_{t+1}|X_t = x\}$. Try different bandwidths and comment on your findings.

Hints

R: You can get inspired by the script in `3dPlotting.R` and explore local regression with `loess()` or `lm()`. See `?loess` for which type of kernel etc. is used. Using `lm()` and doing the weights with your own kernel function enables more models to be fitted, which is especially useful for later computer exercises.

Matlab: The function in `regsmooth1D.m` does one-dimensional local polynomial regression.

Part 3

Use the cumulative conditional means technique in connection with the chosen SETAR model in part 2. Compare with the theoretical cumulative conditional mean and explore the assymptotic behaviour.

Hints

R and Matlab: The scripts in `cumulativeMeans.R` and `cumulativeMeans.m` are implementations of the cumulative conditional means technique.

Part 4

During the heating season, the heat-loss coefficient of buildings, U_a, is often estimated under the assumption that the internal temperature is kept constant, so that the heat load is described by

$$\Phi_t = U_a(T_t^i - T_t^e) + \epsilon_t,$$

where Φ_t is heat load, U_a is the heat loss coefficient and T^i and T^e are the internal and external temperatures. The problem with this approach is that the heat loss coefficient depends nonlinearly on the wind speed. Using the data from `DataPart4.csv`, but with U_a as a non-parametric function of the wind speed, i.e

$$\Phi_t = U_a(W_t)(T_t^i - T_t^e) + \epsilon_t.$$

Plot the estimated function $U_a(W_t)$.

Hints

It might be useful to rewrite the deterministic part of the model as

$$U_a(W_t) = \frac{\Phi_t}{T_t^i - T_t^e}.$$

R: In `3dPlotting.R` local polynomial regression is carried out, both using `loess()` and `lm()`. Conditional parametric models can be fitted with either: using `loess()` the parameter `parametric` needs to be altered and using `lm()` the way the weigths are calculated needs to be altered.

Matlab: The function in `regsmooth2D.m` fits either a local polynomial regression model or a conditional parametric model. Look into it to learn what is done differently for the two modelling techniques.

Part 5

Open the data from `DataPart5.csv` and model it using an ARMA model, increasing the model order assisted by the ACF and PACF, until the residuals look white. When you are reasonably satisfied compute the LDF of the residuals. Do you find any significant non-linearities? Propose a better model structure.

Hints

Check the n-step residuals versus residuals plot, where n is the significant LDF lag, i.e. e_t versus e_{t-n}.

R: The script `ldf.R` is a way to estimate lagged dependent functions.

Matlab: The script in `ldf.m` does an estimation of Lagged Dependent Functions. The script in `ldfone.m` can be used to see the estimated dependence function for some lag.