Advanced Time Series Analysis: Computer Exercise 1

Autumn 21, 01622
Assessment 1
January 4, 2022

Magne Egede Rasmussen, s183963,
Nicolaj Hans Nielsen, s184335
& Anton Ruby Larsen, s174356

Danmarks
Tekniske Universitet

Contents

1 Part 1

In this exercise we are to simulate different non-linear parametric models. We will focus on a large family of non parametric models called threshold models which are described on page 18-23 in the notes. Threshold models split the real line up into l disjoint intervals, $R_1, R_2, ..., R_l$, which together cover the whole real line. This is illustrated in figure 1.

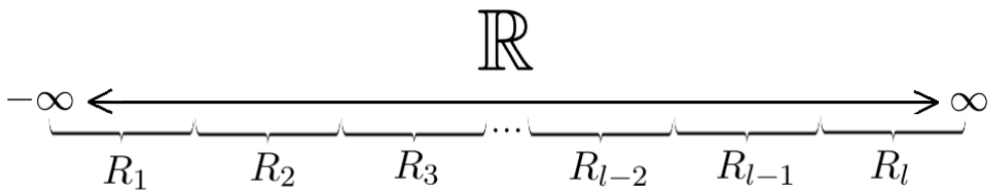


Figure 1 – Disjoint intervals which covers the whole real line for a threshold model

1.1 SETAR

The first model from this family we will describe is the SETAR model. SETAR stands for Self-Exciting Threshold AR and is given by:

$$X_t = a_0^{(J_l)} + \sum_{i=1}^{k_{J_t}} a_i^{(J_t)} X_{t-i} + \epsilon_t^{(J_t)} \tag{1.1}$$

where

$$J_t = \begin{cases} 1 & X_{t-d} \in R_1 \\ 2 & X_{t-d} \in R_2 \\ \vdots & \vdots \\ l & X_{t-d} \in R_l \end{cases} \tag{1.2}$$

Again here l codes for the number of regimes, the parameter d determines how many time-steps back we need to look to determine the current regime and lastly we have l k's which each encode the order of the relevant regimes AR model. We specify a specific SETAR model by writing $SETAR(l, d; k_1, \ldots, k_l)$

We first try to simulate 2,000 points of a SETAR(2,1;1,2) model with AR models given as

$$X_t = \begin{cases} -3 + X_{t-1} + \varepsilon_t^{(1)} & if \quad X_{t-1} \leq 0 \\ 3 + X_{t-1} - 0.5X_{t-2} + \varepsilon_t^{(2)} & if \quad X_{t-1} > 0 \end{cases} \tag{1.3}$$

We plot the simulated points

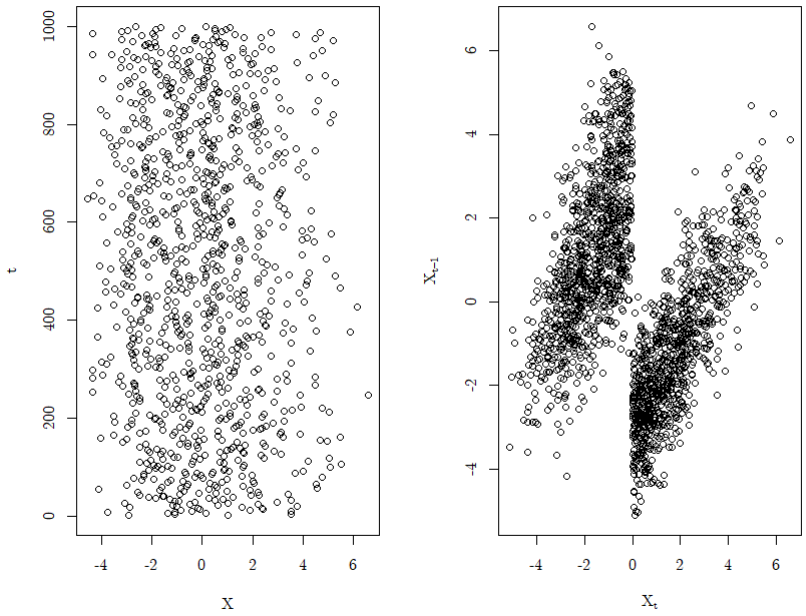


Figure 2 – We have simulated 2000 points of 1.3. The left plot is the first 1000 points and the right plot is X_t plotted against X_{t-1}

We see in figure 2 that no obvious patterns are apparent from the the left plot but if we plot against one time lag the regimes becomes very apparent and the abrupt shift at 0 is very clear. We now try to see how the model shifts regimes.

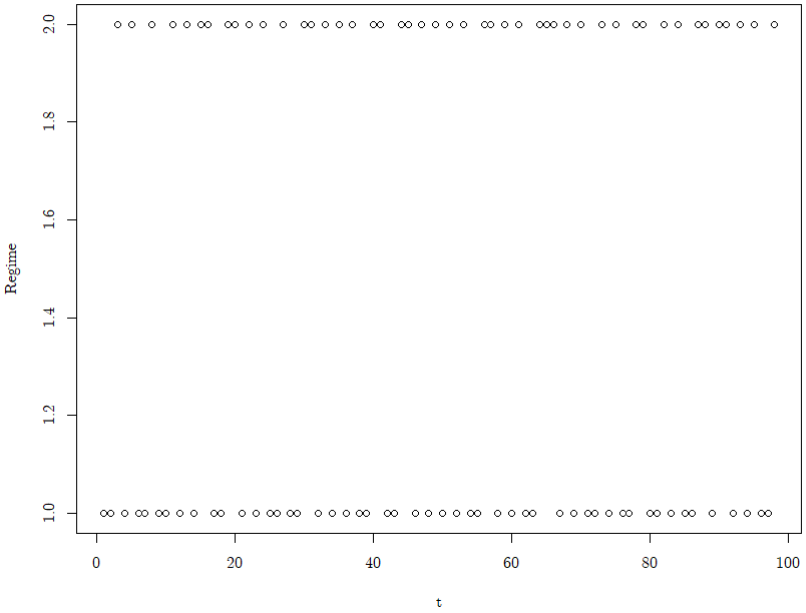


Figure 3 – We have plotted how 1.3 shifts regimes as time goes.

We see that the model shift regime every time point or every second. If we calculate how

large a fraction it spends in every regime we see that 49% of the time it is in regime 1 and 51% of the time it is in regime 2.

We now try to set $d = 2$ in 1.3 and obtain the following model.

$$X_t = \begin{cases} -3 + X_{t-1} + \varepsilon_t^{(1)} & if \quad X_{t-2} \leq 0 \\ 3 + X_{t-1} - 0.5X_{t-2} + \varepsilon_t^{(2)} & if \quad X_{t-2} > 0 \end{cases} \tag{1.4}$$

We again simulate 2000 points but because we know no information will be obtained from no time lag we will only plot X_t against X_{t-1} and X_t against X_{t-2}.

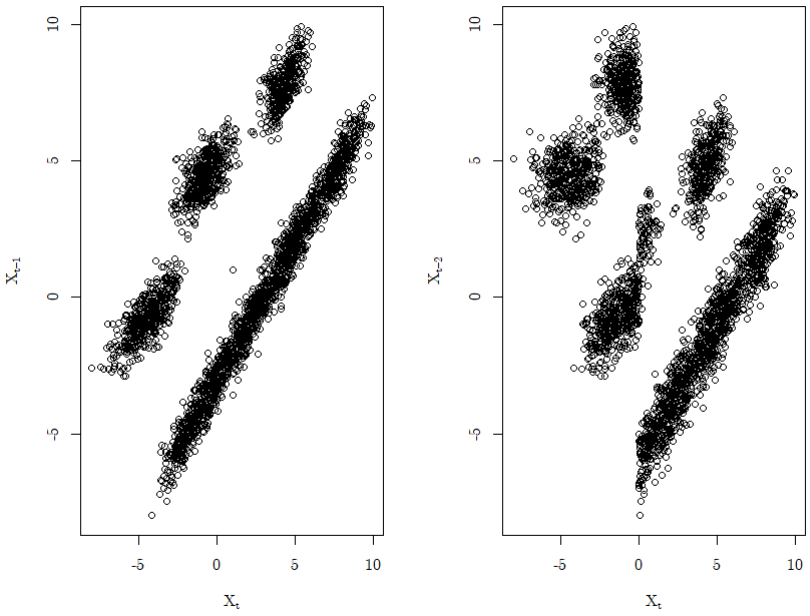


Figure 4 – We have simulated 2000 points of 1.4 and the left plot X_t against X_{t-1} and the right plot is X_t against X_{t-2}

We see in figure 4 that we see a very non linear behaviour from the left plot but the regime changes are not very apparent. However if we look at the right plot the change at 0 is again very apparent. We again plot the regime changes in figure 5

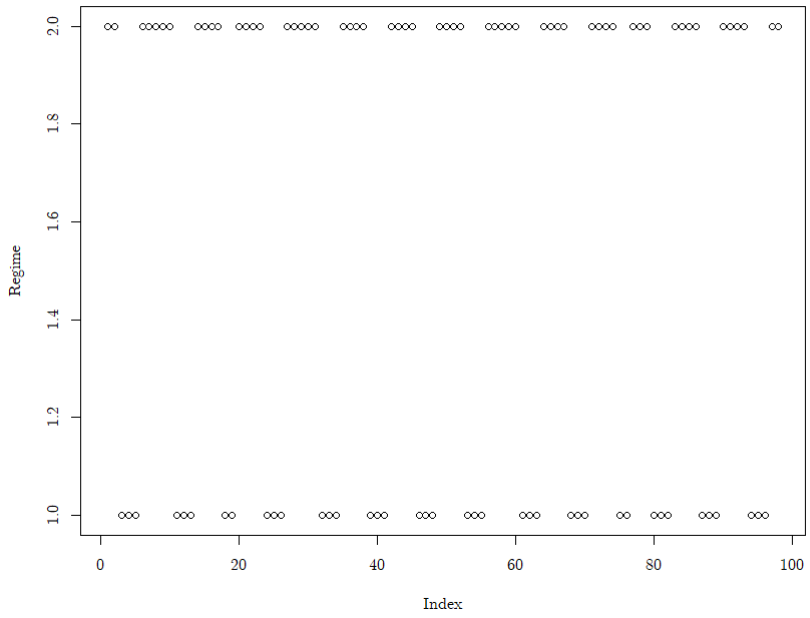


Figure 5 – Regime changes of 1.4

We see that quite a bit more in regime 2 from he looks of figure 5 and if we calculate the fraction we also see that we are in regime 2 59% of the time.

1.2 IGAR

The next model we will investigate is the Independently Governed AR model (IGAR). This is also a threshold model but instead of the current regime being determined by the value of some previous data point we now sample the current regime from a multinomial distribution every time. Hence the model is given by

$$X_t = a_0^{(J_l)} + \sum_{i=1}^{k_{J_t}} a_i^{(J_t)} X_{t-i} + \epsilon_t^{(J_t)} \tag{1.5}$$

where

$$J_t = \begin{cases} 1 & \text{with prob } p_1 \\ 2 & \text{with prob. } p_2 \\ \vdots & \vdots \\ l & \text{with prob. } 1 - \sum_{i=1}^{l-1} p_i \end{cases} \tag{1.6}$$

We will simulate a 2000 points of a IGAR(2;1,2) model given by

$$X_t = \begin{cases} -3 + X_{t-1} + \varepsilon_t^{(1)} & \text{With probability 0.75} \\ 3 + X_{t-1} - 0.5X_{t-2} + \varepsilon_t^{(2)} & \text{With probability 0.25} \end{cases} \tag{1.7}$$

We plot the simulated data.

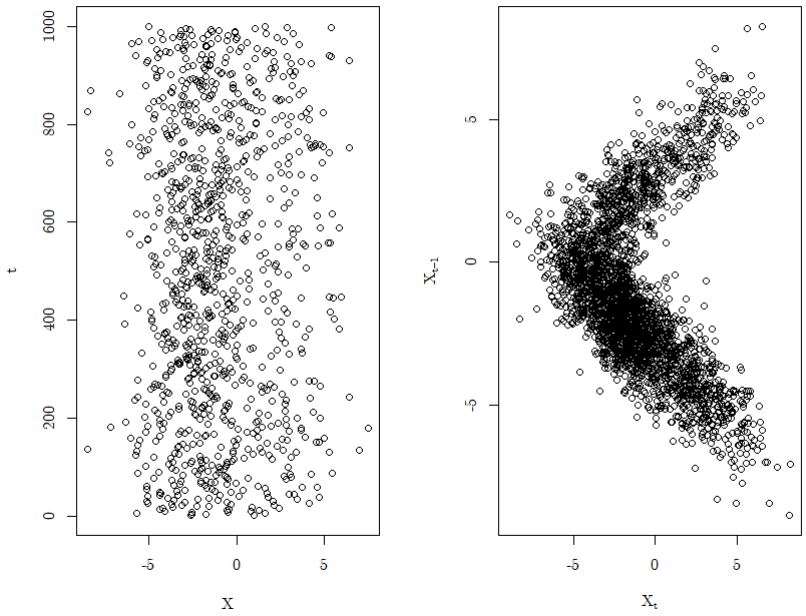


Figure 6 – We have simulated 2000 points of 1.7. The left plot is the first 1000 points and the right plot is X_t plotted against X_{t-1}

We see in figure 6 that the left plot is not that informative besides maybe having a higher density in the negative values. In the right plot we have plotted X_t against X_{t-1} and here we see a very clear pattern where two legs forms from $X_1 \approx -7$ and $X_{t-1} = 0$. If we calculate the fraction of time spend in each regime we get that 26% of the time is spend in regime 2 and 76% of the time is spend in regime 1 which is as expected.

1.3 MMAR

The next model we will simulate is yet another model from the threshold family called the Markov Mixture AR model (MMAR). It changes regime according the a Markov chain but the rest of the model follows the same structure as the SETAR and IGAR model. The model we will simulate here will be as the IGAR model we simulated but the regime changes will follow the Markov chain given by

$$\mathbf{P} = \begin{array}{c} \\ 1 \\ 2 \end{array} \begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \left\| \begin{array}{cc} 0.8 & 0.2 \\ 0.1 & 0.9 \end{array} \right\| \end{array} \tag{1.8}$$

We simulate 2000 data points as for the previous models.

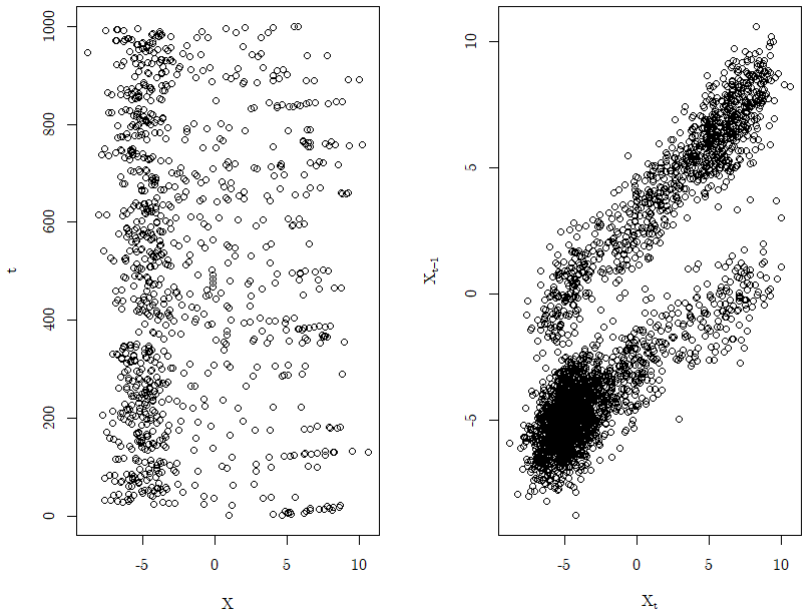


Figure 7 – We have simulated 2000 points of the MMAR model following the Markov chain given by P. The left plot is the first 1000 points and the right plot is X_t plotted against X_{t-1}

We see in figure 7 that the left plot is not that informative besides having a fairly higher density in the negative values. In the right plot we have plotted X_t against X_{t-1} and here we see the two states as the dark spot in the lower left corner and the top right corner. The two diagonal traces coming out of each state is the transitions from the other state.

The expected time in each state we can calculate by finding the stationary distribution of $\mathbf{P}$. We find it to be

$$\pi = \begin{bmatrix} \frac{1}{3} \\ \frac{2}{3} \end{bmatrix} \tag{1.9}$$

If we calculate the fraction of time spend in each regime we get that 67% of the time is spend in regime 2 and 33% of the time is spend in regime 1 which fits nicely with the theory.

2 Part 2

In part 2 of the computer exercise we again consider a Self-Exciting Threshold AR model. In correspondence with the task description we now look at a SETAR$(2,1;1,1)$ which in short notation can be written as SETAR$(2,1;1)$. Hence, both the regime selection and all predictions depend only on values from the last time step.

We have selected to look at the SETAR$(2,1;1)$ given by:

$$X_t = \begin{cases} 3 + 0.6X_{t-1} + \varepsilon_t^{(1)} & X_{t-1} \leq 0 \\ -3 + 0.4X_{t-1} + \varepsilon_t^{(2)} & X_{t-1} > 0 \end{cases} \tag{2.1}$$

At first, let us calculate the theoretical conditional mean $M(x)$ of (2.1). Based on the general described on page 38 in [1], the conditional mean for the SETAR$(2,1;1)$ is given by the piecewise expression:

$$M(x) = \mathbb{E}\{X_{t+1}|X_t = x\} = \begin{cases} -3 + 0.4x & \text{if } x > 0 \\ 3 + 0.6x & \text{if } x \leq 0 \end{cases} \tag{2.2}$$

Using indicator functions, we can rewrite the conditional mean as:

$$M(x) = \mathbf{1}_{\{x>0\}}(x)\,[-3 + 0.4x] + \mathbf{1}_{\{x\leq 0\}}(x)\,[3x + 0.6x] \tag{2.3}$$

2.1 Local Regression Models

For linear time series models it is well-known that the conditional mean is linear. Looking at our found $M(x)$ from (2.2) we see that for our selected SETAR$(2,1;1)$ linearity is definitely not the case. In fact, $M(x)$ is piecewise linear. Hence for a data based estimation of the conditional mean we must consider new methods.

One solution is to use kernel estimators which was proposed by Nadaraya in 1964 and is useful for non-parametric regression in general. By applying product kernels (often used in practice) the estimated conditional mean $\hat{M}(x_{i_1}, \ldots, x_{i_d})$ can be found as:

$$\hat{M}(x_{i_1}, \ldots, x_{i_d}) = \frac{\frac{1}{N-i_d}\sum_{s=i_d+1}^{N} X_s \prod_{j=1}^{d} k\{h^{-1}(x_{i_j} - X_{s-i_j})\}}{\frac{1}{N-i_d+i_1}\sum_{s=i_d+1}^{N+i_1} \prod_{j=1}^{d} k\{h^{-1}(x_{i_j} - X_{s-i_j})\}} \tag{2.4}$$

This type of kernel estimate is equivalent to a local weighted least squares estimate, which will be elucidated in the following subsection: 'Kernels & Distance Measures'. In equation (2.4) k is the selected kernel, h the bandwidth and N the amount of given data points. Finally the indexes $i_1, \ldots i_d$ should be interpreted as in Tjöstheim [2]:

$$M(x_{i_1}, \ldots, x_{i_d}) = \mathbb{E}[X_{t+1}|X_{t-i_1} = x_{i_1}, \ldots, X_{t-i_d} = x_{i_d}] \tag{2.5}$$

As we are interested in estimating the conditional mean for the specific model SETAR$(2,1;1)$, which only takes the the previous time step into account, we simplify (2.4):

$$\hat{M}(x) = \frac{\sum_{s=1}^{N} X_s\ k\{h^{-1}(x - X_{s-1})\}}{\sum_{s=1}^{N} k\{h^{-1}(x - X_{s-1})\}} \tag{2.6}$$

Equation (2.4) is in literature referred to as the Nadaraya-Watson estimator, while (2.6) is a special case with $d = 1$ and can be used to find $\hat{M}(x)$ for the SETAR$(2,1;1)$ from (2.1).

2.1.1 Kernels & Distance Measures

In section 2.1 it was touched upon, how the kernel estimator is closely connected to the local weighted least-squares estimate:

$$\hat{M}(x) = \frac{\sum_{s=1}^{N} X_s \; w_s}{\sum_{s=1}^{N} w_s} \tag{2.7}$$

It is seen that (2.6) and (2.7) are equivalent if we select the weights $w_s = k\{h^{-1}(x - X_{s-1})\}$. Hence an interpretation of a kernel function is to see it as some relative weight factor caused by known data points. As for weighting functions, it often make sense to make smooth transitions which indeed is the case for the two selected kernel functions introduced in section 2.1.2.

Equation (2.6) is as mentioned a specific case of (2.4) with $d = 1$. In the general case product kernels are applied. Here the selection of the distance measure becomes of interest. The most classic used distance measures are the L_1, L_2 and L_∞ norm. It is not obvious in every case which norm is optimal, but typically L_2 is used since it is invariant under rotation. The norm choice effects the form of a multivariate kernel function (a product kernel). When the amount of data point increases the differences in the kernel structure becomes less relevant. Hence, it is not always necessary to optimise your distance measure. In figure 8 selected p-norms are illustrated to give an intuition about the impact on the structure.

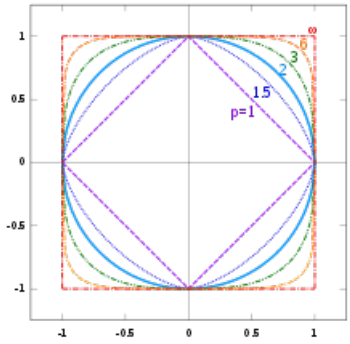


Figure 8 – Illustration of different p-norms. The choice of metric has an impact when $d > 1$ but becomes less important as the number of samples grow. The illustration also shows the invariance of rotation for the L_2 norm.

2.1.2 Kernel Functions of Interest

There are multiple candidates when it comes to selecting the kernel function k. We will in this exercise mainly consider two possibilities: A Gaussian kernel and the Epanechnikov kernel. The Gaussian kernel K_h^{Gau} and corresponding function k^{Gau} are given by

$$K_h^{\text{Gau}}(u) = h^{-1} k^{\text{Gau}}(uh^{-1}) = \frac{1}{h\sqrt{2\pi}} \exp\left(-\frac{u^2}{2h^2}\right) \tag{2.8}$$

where on the other hand the Epanechnikov kernel K_h^{Epa} and function k^{Epa} are given by:

$$K_h^{\text{Epa}}(u) = h^{-1} k^{\text{Epa}}(uh^{-1}) = \frac{3}{4h}\left(1 - \frac{u^2}{h^2}\right) \mathbf{1}_{\{|u| \leq h\}}(u) \tag{2.9}$$

Both kernel functions k^{Gau} and k^{Epa} are a part of all L^p spaces for p real and greater than zero. This is caused by the general kernel condition: $\int k_q(u)du = 1$ where q indicates the q'th order kernel function. While both functions are in L^p spaces, k^{Epa} is also in the space of compacted supported functions $C_c(\mathbb{R})$ which can be of modelling and numeric interest (for example with physical bounds or non-negativity). Further, Härdle showed in 1990 that if the optimal bandwidth h is selected, the Epanechnikov kernel minimises the mean squared error among all kernels.

2.1.3 Span Size & Flexible Span: Bias-variance Trade Off

For larger data sets the kernel choice is not that crucial, since the estimates are going to look fairly similar no matter the choice of Kernel function. However the choice of bandwidth h becomes very important. The bandwidth of the kernel is a free parameter highly correlated with the smoothness of your estimations. Setting the bandwidth is a classic bias-variance trade off:

- h large: The variance is small but the bias becomes large. For the extreme case $h \to \infty$, the estimate converges towards the mean of all known values.
- h small: The bias is small but the variance becomes large. Now for the extreme case $h \to 0$, the estimate spikes at the known data points and are 0 otherwise.

The are different methods determining an optimal bandwidth, many uses either the expected L^2 risk function MISE or Silverman's rule of thumb [3]. Here we will use Härdle's bandwidth and therefore try minimising a formula similar to:

$$\text{MSE2}(h) = \frac{1}{N}\sum_{i=1}^{N}[\hat{M}(X_i) - X_{i+1}]^2\pi(X_i) \tag{2.10}$$

Equation (2.10) is not suitable as the optimal solution is definitely found for $h \to 0$, since then $\hat{M}(X_i) - X_{i+1} = 0$ for all know data. Hence a modification is needed to take variance into account also, which is deployed by the use of cross validation with respect to the known values $\hat{M}(X_i)$.

Until now we have considered a constant bandwidth for the entire estimation. Let us now look into an adaptive solution for selecting k, namely using nearest neighbours. Modelling-wise it is often a good idea to select a variable bandwidth. A main feature regards the tails of a distribution. As the density of points is often low in such areas, the nearest neighbour spans a large interval and the local value of h becomes large. Hence the nearest neighbours method tends to keep low variance and higher bias in these tails, which implies a more smooth estimate there.

To apply a k-nearest neighbour method, the amount of neighbours k must indeed be selected. The method of finding an optimal amount of neighbours to use is similar to optimising a fixed bandwidth, although here potential k's are running in the natural numbers and are thereby not continuous. We want to secure that MSE2(k) is small. Again, the issue is that $h \to 0$ is an optimal solution based on our known data. Hence we apply cross validation but since k is discrete, this amounts to computing CV(k) for a set of chosen integer values and find the minimum.

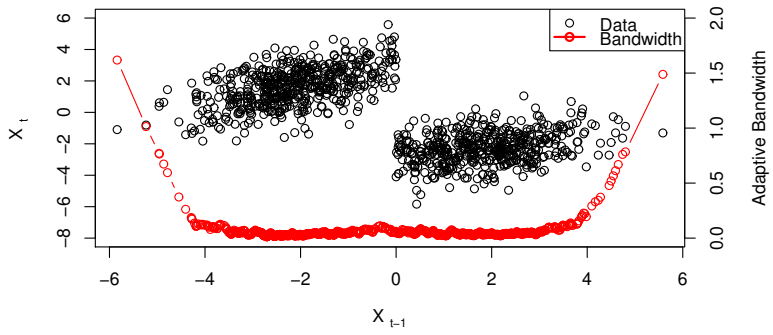


Figure 9 – Optimal adaptive bandwidth ($k = 11$) plotted together with our data distribution.

2.2 Results & Comparisons

Using the theory from section (2.1) we now start to estimate the conditional mean. For the estimations we use 1000 simulated values of the SETAR model given by equation (2.1).

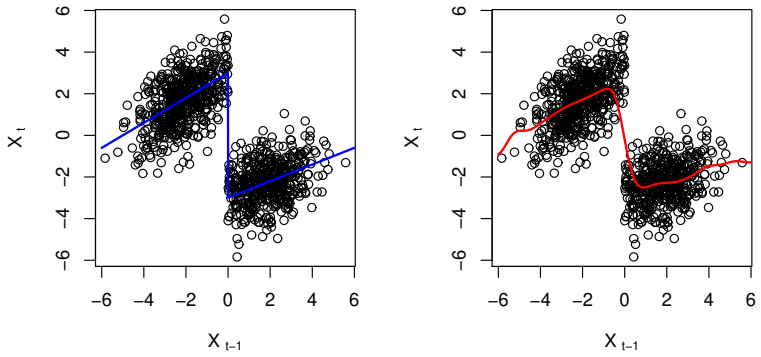


Figure 10 – On both plots we see the simulated values of X_t plotted against X_{t-1}. The blue line indicates the analytical conditional mean, while on the right plot an example of a Gaussian kernel estimate with fixed bandwidth ($h = 0.4$) is shown. The plots indicate that kernel estimates seem to align with the conditional mean as expected.

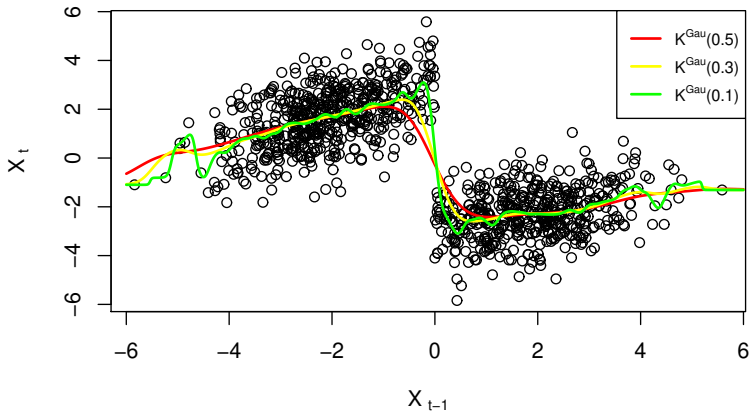


Figure 11 – In figure 10 we created a kernel estimate with $h = 0.4$. Here we have tried different fixed bandwidths. The results show that small bandwidths capture the transition between states but reduce smoothness as expected from section (2.1.3).

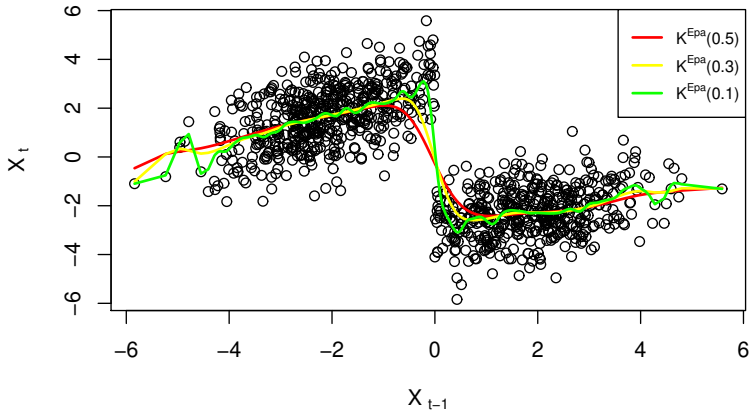


Figure 12 – Here we also try a range of fixed bandwidths but with respect to an Epanechnikov kernel. Already with 1000 points the two kernel types seem to output similar estimations.

Combining cross validation and the MSE2 formula from equation (2.10), we find optimal bandwidths by minimising our objective for respectively a Gaussian kernel and a Epanechnikov kernel. The results are illustrated below together with the analytic conditional mean.

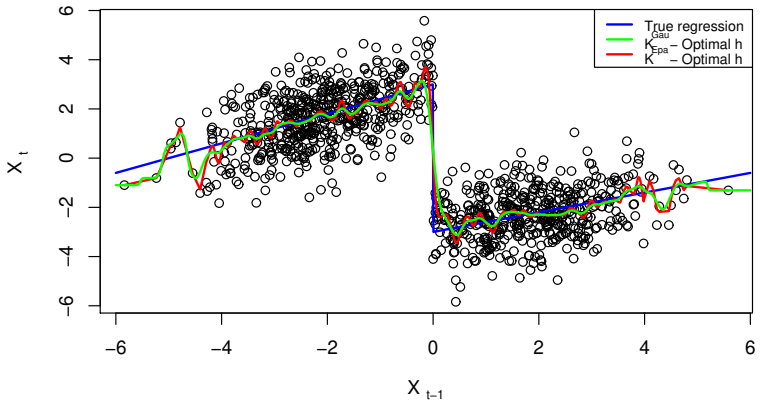


Figure 13 – For both kernel types we see that the optimal bandwidth favours modelling the state transition (the non-linear jump of equation (2.1)). Based on the difference between true regression and the two estimations, we are left with a high variance problem around the tails.

2.2.1 Increased Estimation Complexity

Although we are fairly pleased with the estimation results shown in figure (13), we would like to add another level of complexity to our models with the aim of fixing the variance problem at the tails. Here we introduce locally linear regression, which in many papers are thought to be superior to kernel regression. Contrary to kernel estimation, local linear estimation fits a straight line close to each point instead of constants. Thus, locally linear estimation removes a bias term from the kernel estimator, which create a smoother and often better behaviour near the boundary of the points.

The local linear regression has been computed by solving one weighted least square problem. For optimal bandwidths we see the difference between kernel and local linear regression below:

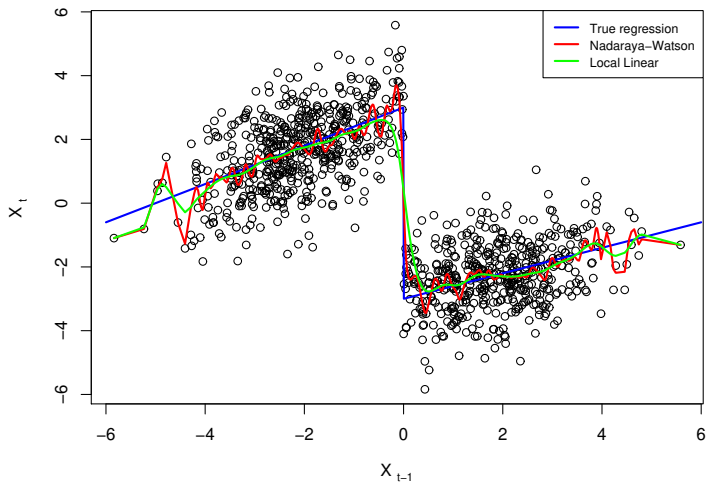


Figure 14 – Optimal local linear estimation vs. local kernel estimation for estimating the conditional mean using an Epanechnikov kernel. We see that the optimal local linear estimate is more smooth which for example deals with the high variance tail issue.

Finally, we implement an adaptive bandwidth based on the theory from section (2.1.3). We add the adaptiveness on top of local linear estimates as it seems to obtain better fits:

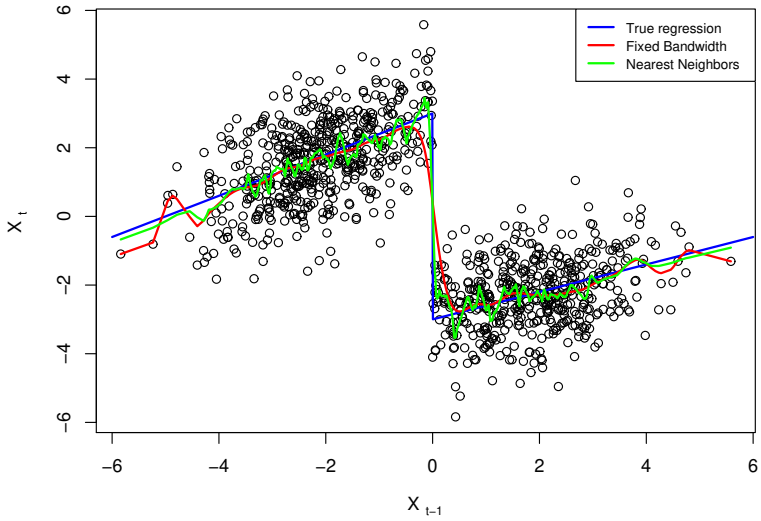


Figure 15 – Fixed vs adaptive bandwidth added to a local linear regression model based on the Epanechnikov kernel. Here both the fixed and adaptive line are based on the bandwidth minimising MSE2 using cross validation. We see that the nearest neighbours bandwidth in general seems to align better with the analytic conditional mean. On the other hand it also seems to decrease the local estimate smoothness.

Based on the illustrations through this section, we conclude that the modelling choice of kernel function is not that important when simulating a 1000 values from our SETAR$(2, 1; 1)$. Hence for simplicity and numerical efficiency we would recommend selecting an Epanechnikov kernel. For better estimations (especially at the tails) it is an idea to use local linear regression and test adaptive bandwidth using nearest neighbours.

3 Part 3

In the following, we will calculate the cumulative conditional mean and variance

$$\begin{aligned}\Lambda(s) &= \int_a^s \lambda(x)dx \\ \Gamma(s) &= \int_a^s \gamma(x)dx\end{aligned} \tag{3.1}$$

where we restrict our attention to some interval $[a, b]$ and the SETAR(2;1;1) model from task 2 equation (2.1) with the conditional mean described in equation (2.2). We will express the model using the indicator function:

$$M(x) = \mathbf{1}_{\{x>0\}}(x)\left[-3 + 0.4x\right] + \mathbf{1}_{\{x\leq 0\}}(x)\left[3 + 0.6x\right] \tag{3.2}$$

Using equation (3.2), we will estimate the theoretical mean for each of the regimes below. In the following section, we will utilise the following property of a an indicator function $\int_\infty^\infty \mathbf{1}_{\{0<x\leq\beta\}}(x)f(x)\mathrm{d}x = \int_0^\beta f(x)\mathrm{d}x$ for some generic function f.

3.1 Theoretical conditional mean

We want to calculate the theoretical conditional mean inside some interval $X \in [\alpha, x)$ for each of the regimes using equation (3.2).
Consider the first regime for which $x \leq 0$ then:

$$\begin{aligned}\Lambda(x) &= \int_\alpha^x \mathbf{1}_{\{s\leq 0\}}(s)\left[3 + 0.6x\right]\mathrm{d}s \\ &= \int_\alpha^x \left[3x + 0.6s\right]\mathrm{d}s \\ &= 3\left(x - a\right) + \frac{3}{10}\left(x^2 - a^2\right)\end{aligned} \tag{3.3}$$

Consider now the second regime for which $x > 0$ then

$$\begin{aligned}\Lambda(x) &= \int_\alpha^x M(s)\mathrm{d}s \\ &= \int_\alpha^0 M(s)\,\mathrm{d}s + \int_0^s M(s)\mathrm{d}s \\ &= \int_\alpha^0 3 + 0.6s\,\mathrm{d}s + \int_0^x -3 + 0.4s\,\mathrm{d}s \\ &= -3a - \frac{3}{10}a^2 - 3x + \frac{1}{5}x^2\end{aligned} \tag{3.4}$$

hence the theoretical cumulative mean is:

$$\Lambda(x) = \begin{cases} -3a - \frac{3}{10}a^2 - 3x + \frac{1}{5}x^2 & \text{if } x > 0 \\ 3\left(x - a\right) + \frac{3}{10}\left(x^2 - a^2\right) & \text{if } x \leq 0 \end{cases} \tag{3.5}$$

This we will use in the section below.

3.2 Non-parametric estimators

Using notation from equation (3.2) and (3.3) in [1] let $\hat{\lambda}(x)$ and $\hat{\gamma}(x)$ denote the non-parametric estimates of the conditional mean and variance respectively. Let $I_1, ..., I_{d_n}$ be partitions of $[a, b]$ of equal length h_n. Then we introduce the estimators

$$\begin{aligned} \hat{\Lambda}(s) &= \int_a^s \hat{\lambda}(x)\mathrm{d}s \\ \hat{\Gamma}(s) &= \int_a^s \hat{\gamma}(x)\mathrm{d}s \end{aligned} \tag{3.6}$$

where

$$\begin{aligned} \hat{\lambda}(x) &= \frac{(nh_n)^{-1}\sum_{t=1}^n X_t \mathbf{1}_{\{X_{t-1}\in I_x\}}}{(nh_n)^{-1}\sum_{t=1}^N \mathbf{1}_{\{X_{t-1}\in I_x\}}} \\ \hat{\gamma}(x) &= \frac{(nh_n)^{-1}\sum_{t=1}^n \left(X_t - \hat{\lambda}(x)\right)^2 \mathbf{1}_{\{X_{t-1}\in I_x\}}}{(nh_n)^{-1}\sum_{t=1}^n \mathbf{1}_{\{X_{t-1}\in I_x\}}} \end{aligned} \tag{3.7}$$

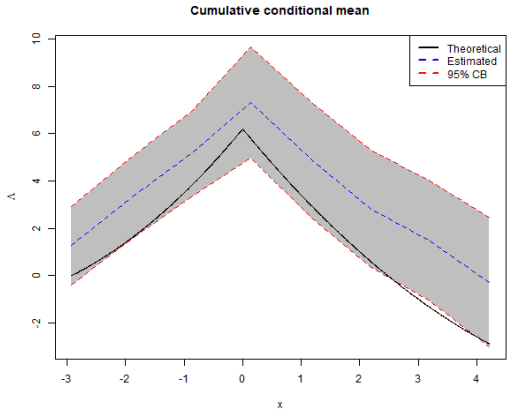
Using the definition from (3.11) [1] let $\hat{f}$ be the histogram estimator for f and introduce $\hat{H}(s) = \int_a^s \hat{\gamma}/\hat{f}\mathrm{d}s$. We can under weak conditions see [1] introduce an asymptotic $100\,(1-\alpha)\,\%$ confidence band:

$$\hat{\Lambda}(x) \pm c_\alpha n^{-1/2}\hat{H}(b)^{1/2}\left(1 + \frac{\hat{H}(x)}{\hat{H}(b)}\right) \quad x \in [a, b] \tag{3.8}$$

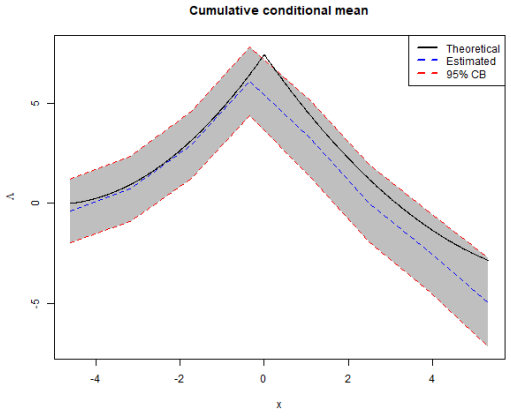
where c_α can be obtained as the upper α quantile of the $\sup_{t\in[0,1/2]} |B^0(t)|$ distribution where B^0 is a Brownian bridge. In the following, we will explore the properties of this technique.

3.3 Results

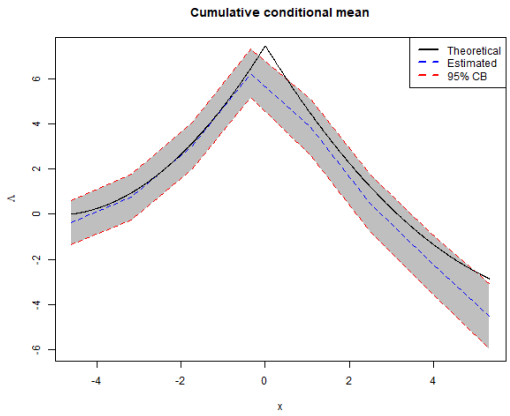
From equation (3.8), we see that for a fixed h_n, α, and b, it seems that the band will decrease as we increase n. To explore this claim and the asymptotic behaviour of the cumulative conditional mean technique, we will let n take values of 100, 500, 1000, and 100,000. The samples will be from the same SETAR model hence we can use the calculated theoretical result of equation (3.5). Along the estimate, we plot the confidence band and the theoretical value where we use 8 bins. α change slightly when we add more samples because the interval on which we have values expands slightly.



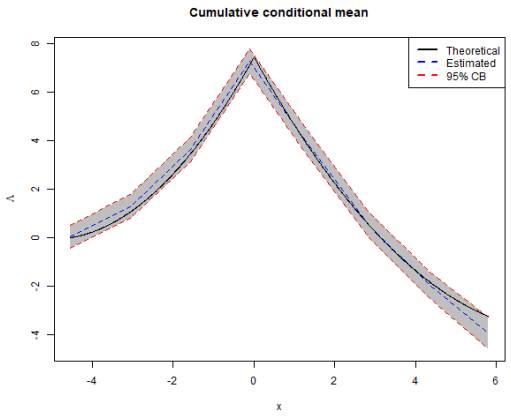
(a) Cumulative condition mean, 100 points



(b) Cumulative condition mean, 500 points



(c) Cumulative condition mean, 1000 points



(d) Cumulative condition mean, 100,000 points

Figure 16 – Cumulative conditional means with increasing amounts of samples

In figure 16, we see that the 95% confidence band is decreasing as more samples are used to estimate the conditional mean. Further, for each of the cases, we see that the theoretical mean seems to reside inside the conditional bands most of the time as we expect for a 95% condition band. This suggests that empirically $\hat{\Lambda}(s)$ convergers towards $\Lambda(s)$ when we increase n hence the technique seems to work as expected on simulated data.

4 Part 4

In the following, we consider how to model the heat load, Φ_t. As we heat up our houses when it is cold outside, we could imagine that Φ_t depends on some relationship between the internal temperature of a building, T_t^i, and the external temperature T_t^e. If we assume that the internal temperature is kept constant, then we could use the model

$$\Phi_t = U_a \left(T_t^i - T_t^e\right) + \epsilon_t \tag{4.1}$$

where U_a is the heat-loss coefficient of buildings. However, U_a is not constant but has some non-linearly dependency on the wind speed, W_t. Therefore, we will consider a conditional parametric model

$$\Phi_t = \left(T_t^i - T_t^e\right) U_a\left(W_t\right) + \epsilon_t \tag{4.2}$$

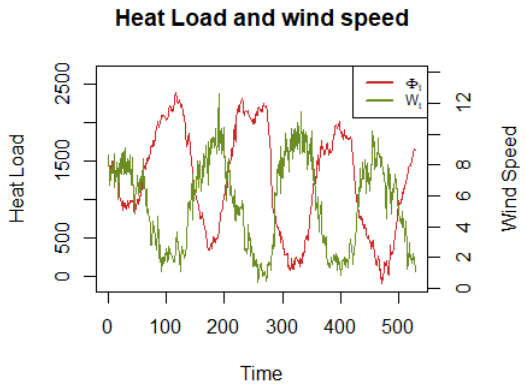
where $U_a\left(W_t\right)$ is a local polynomial estimate hence $U_a\left(W_t\right)$ will be varying coefficients conditioned on the wind speed e.g.

$$U_a\left(W_t\right) \approx \theta_0 + \theta_1 W_t + \theta_2 W_t^2 \tag{4.3}$$

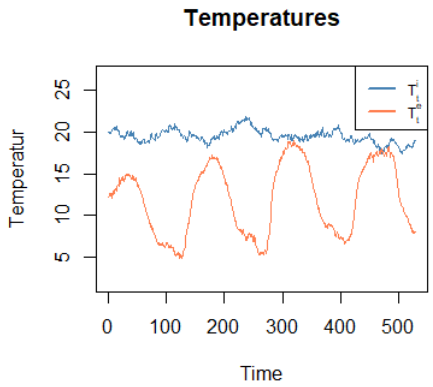
wherein this case the order of the polynomial is 2, $p = 2$.

4.1 Data

To grasp the dynamics of the system, we plot the available data



(a) Heat load temperature and wind speed



(b) Temperatures internal, T_t^i, and external,T_t^e

Figure 17 – Time series for part 4

Consider first the recurrent pattern of W_t, Φ_t and T_t^e. From figure 17a, it seems that W_t and Φ_t are negatively correlated. Compared with figure 17b, it seems that W_t and T_t^e are positively correlated as they oscillate with the same period and are synchronised.
From figure 17b, we see that the internal temperature, T_t^i, seems more constant over time which is also assumed for the model in equation (4.1).

4.2 Estimation of heat-loss coefficient

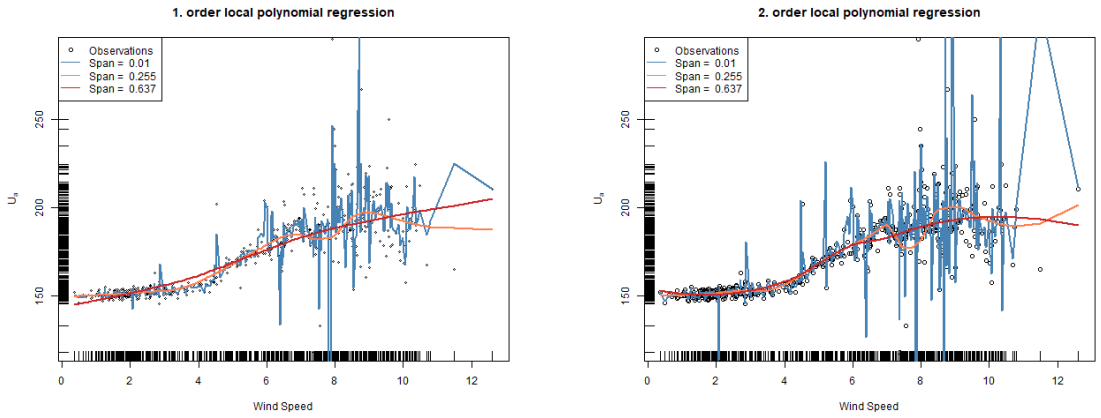
Based on equation (4.2), we can consider

$$\frac{\Phi_t}{T_t^i - T_t^e} = U_a\left(W_t\right) \tag{4.4}$$

to estimate the local polynomial model $U_a\left(W_t\right)$. Let $\Psi_t = \frac{\Phi_t}{T_t^i - T_t^e}$ for ease of notation, then we find the optimal local polynomials by solving for θ^* in

$$\theta^* = \arg\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} K_h\left(w_t - W_t\right) \left[\Psi_t - \sum_{j=0}^{p} \theta_j \left(W_t - w_t\right)^j\right]^2 \tag{4.5}$$

We consider only $p = 1$ and $p = 2$ hence local polynomials from degree 1 to degree 2. We will use the function *loess* which takes an argument *span* $\in\ ]0, 1]$ that is the amount of points to include for the estimation of the local polynomials where 1 would be all points. We let K be a Gaussian kernel in the following analysis. In figure 18, we investigate what the span parameter means for each of the polynomials.



(a) Local polynomials of degree 1

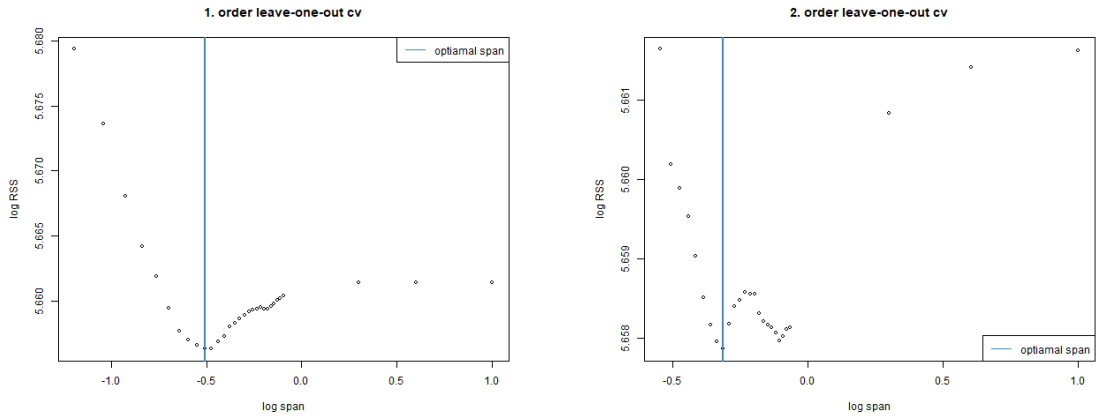
(b) Local polynomials of degree 2

Figure 18 – Local polynomials

In figure 18, we see that a large span leads to a more stable estimate less prone to points far away from the average local behaviour. Comparing figure 18a and figure 18b, we see that when we increase the order of the local polynomials, the fit is closer to the data but also more prone to outliers. Data is quite sparse for wind speeds above 10 and the second-order polynomials in figure 18b seem to fluctuate more in this area. This could be indicative of overfitting because a new point in this area would greatly impact the estimator. In short, the variance of our estimator increases when we increase the complexity of our model.
Interestingly, if we compare figure 18a and figure 18b with span = 0.637, then we see that the 1. order polynomial is more biased. If we focus on wind speeds around 4, then the 1. order polynomial is far above most of the data points due to the upward curvature of the points for $x > 4$. However, the 2. order polynomial is not as biased as the second-order term can capture the local curvature.
We see that when we move to higher-order polynomials, we can decrease the bias with a payoff in the variance of the estimator.

4.3 Optimal span

Using leave-one-out cross-validation, we will now find the optimal span for polynomials of orders 1 and 2. To find the optimal span, we will search extensively around values that minimizes the residual sum of squares.



(a) LOO for local polynomial of degree 1 **(b)** LOO for local polynomials of degree 2

Figure 19 – Log-log base 10 plot of optimal span using LOO

In figure 19a, we see that using LOO the loss function for the first order local polynomial is unimodal and the found optimal span is around 0.3096 with RSS on 453214.6. In figure 19b, we see that the optimal is larger and the loss function is bimodal but slightly lower for the modal with optimum span 0.485 and 454846.4.
It seems that the 1. order polynomial fit is slightly better when we compare the RSS values, however, they perform very similarly when compared:

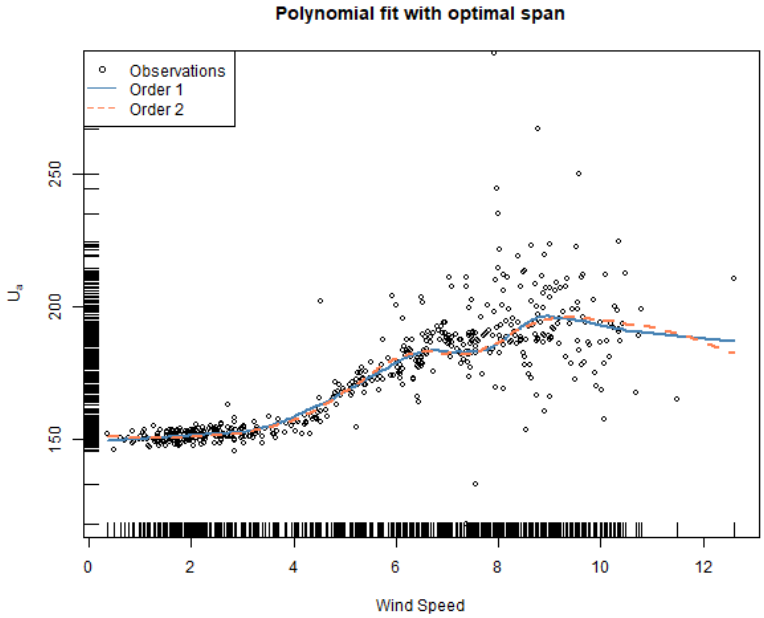


Figure 20 – The found models for the local polynomials with optimal span

In figure 20, the local polynomials deviate mostly when the wind speed is large and the data is sparse. However, for most of the time, the estimate is close to identical. Explicitly, we discovered that a polynomial with a low degree and a small span gives an estimate close to that of a polynomial with a higher degree and a larger span.

5 Part 5

In this exercise we are given an univariat dataset consisting of 800 datapoints which are plotted in figure 21.

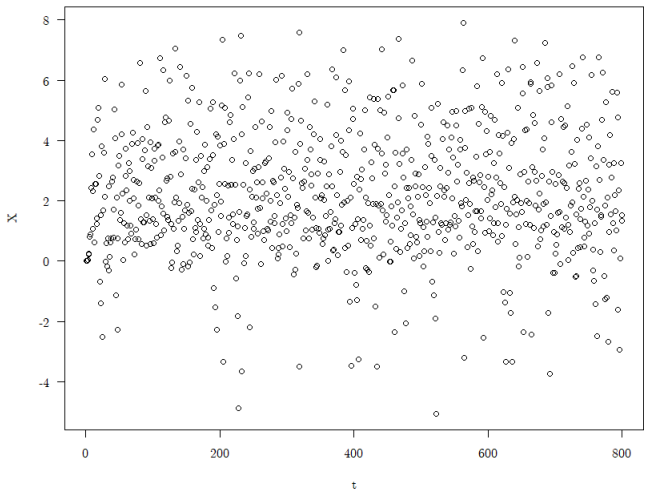


Figure 21

Just by looking at the data in figure 21 we see that the distribution is a bit skewed to the positive side.

5.1 Linear analysis

We try to fit a linear model to the dataset. We start by fitting an ARMA(0,0) model and analyze the ACF and PACF of the residuals.

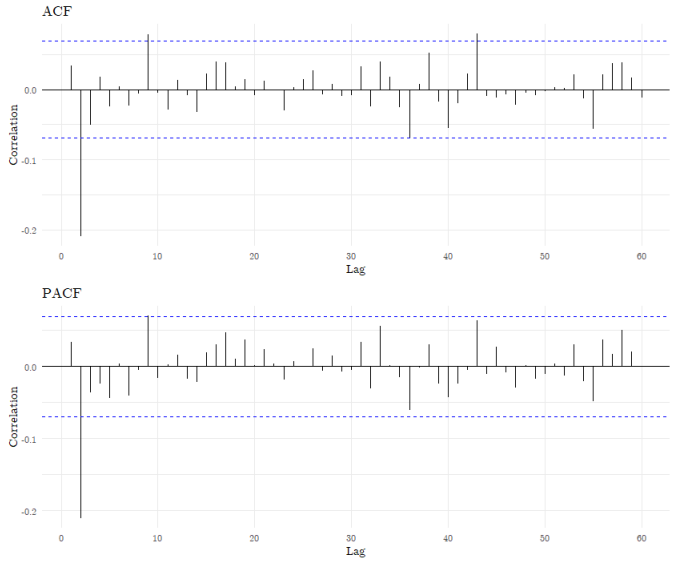


Figure 22

In figure 22 we see the ACF and PACF of the ARMA(0,0) where we see a significant

negative spike in lag 2 of both the ACF and PACF. We hence try to fit an ARMA(2,2) model.

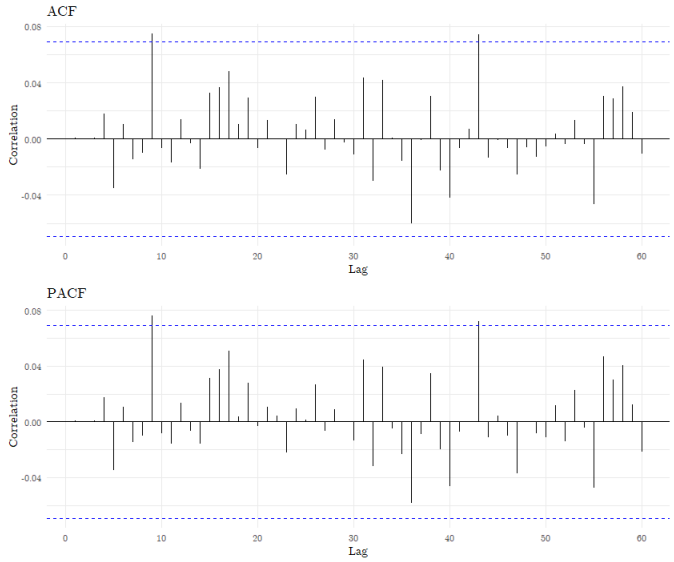


Figure 23

We see that the ACF and PACF of the ARMA(2,2) in figure 23 has no significant spikes. We hence proceed with further residual analysis.

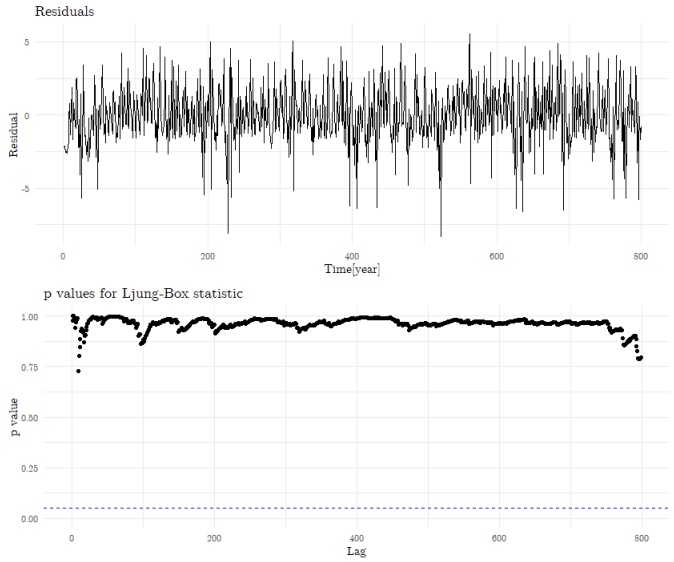


Figure 24

In figure 24 the residuals of the ARMA(2,2) are plotted together with a Ljung-Box test. The residuals are a bit skewed to the negative side but no points of the Ljung-Box test are significant. We therefore proceed with the testing.

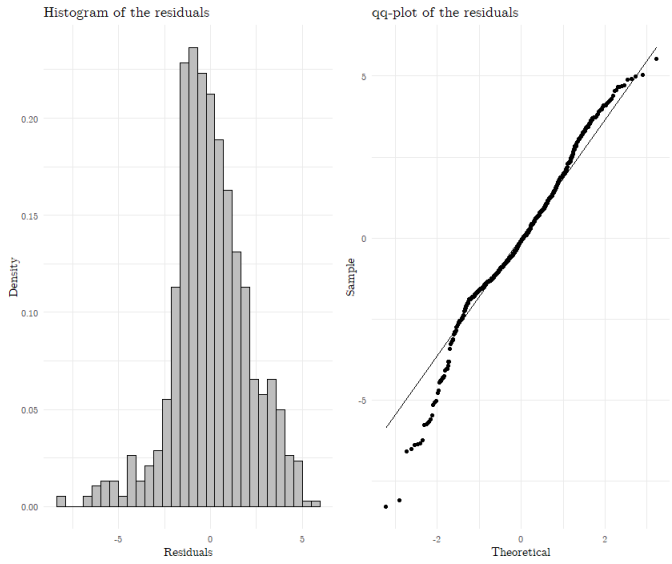


Figure 25 – Histogram and qq-plot over the residuals

In figure 25 a histogram of the residuals is plotted together with a qq-plot. We see higher density for positive values in both the histogram and the qq-plot. We hence try to test the residuals for any non-linear dependencies.

5.2 Non-linear analysis

To do this we use the Lag Dependence Function(LDF). The LDF is a non-linear extension of the sample autocorrelation function and is given by

$$LDF(k) = \text{sign}\left(\hat{f}_k(b) - \hat{f}_k(a)\right)\sqrt{\left(\tilde{R}^2_{0(k)}\right)_+} \tag{5.1}$$

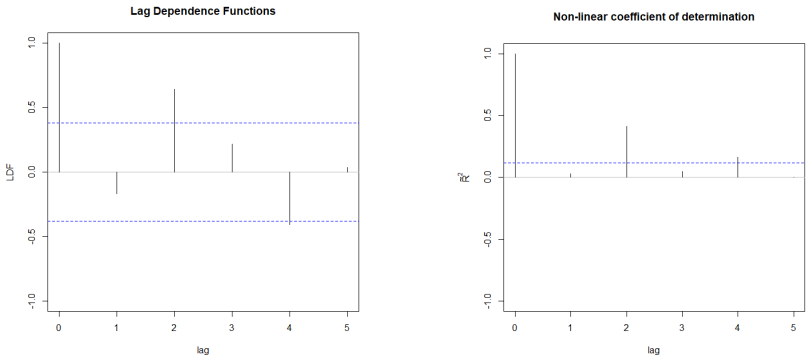
Where a and b respectively are the minimal and maximal observation in the data set and the subscript '+' indicates truncation of negative values. Besides these we have two unknown quantities in 5.1. First we have the $\hat{f}_k$ function which is some non-parametric function which is given by

$$f_k(x) = E\left[X_t \mid X_{t-k} = x\right] \tag{5.2}$$

Next we have the $\tilde{R}^2_{0(k)}$ which is a non-linear extension of the linear coefficient of determination, R^2. It is found by the following expression.

$$\tilde{R}^2_{0(k)} = \frac{\sum_{t=k}^{T}(X_t - \bar{X})^2 - \sum_{t=k}^{T}(\hat{f}_k(X_{t-k}) - X_t)^2}{\sum_{t=k}^{T}(X_t - \bar{X})^2} \tag{5.3}$$

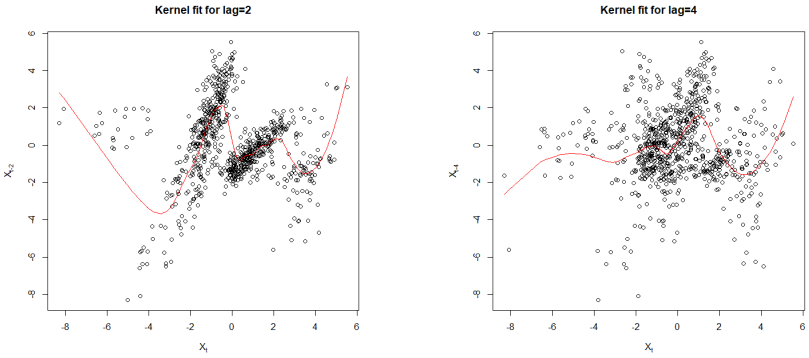
We here see that 5.3 describes if the time series X_t is described by a non-linear dependency of lag k given by $\hat{f}_k$. We calculate the non-linear coefficient of determination together with the Lag Dependence Function for the first 5 lags by use of a second order local polynomial regression.



(a) Lag Dependence Function **(b)** Non-linear coefficient of determination

Figure 26

Lag 0 is 1 by definition but besides that we see that the second and fourth lag are significant so we plot the non-parametric fit to the data for these lags.



(a) k = 2 **(b)** k = 4

Figure 27

We see there is a clear structure for $k = 2$ but the structure is not as pronounced for $k = 4$. We hence start by investigating $k = 2$. We plot the residuals and the data with the time series lagged 2 time steps to see how the residuals are distributed compared to the data

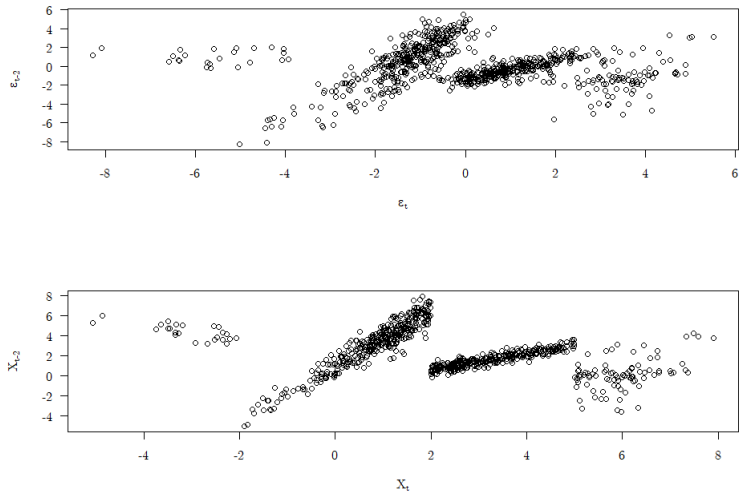


Figure 28

We see a very clear pattern which looks very much like a SETAR(4,2;2) model. We hence try to fit 4 ordinary least squares in each regime and obtain the following model.

$$X_t = \begin{cases} 2.28 - 0.68X_{t-2} + \varepsilon_t^{(1)}, & \varepsilon_t^{(1)} \sim \mathcal{N}(0, 0.54^2) & \text{if } X_{t-2} < -2 \\ 0.90 + 2.73X_{t-2} + \varepsilon_t^{(2)}, & \varepsilon_t^{(2)} \sim \mathcal{N}(0, 0.79^2) & \text{if } -2 \leq X_{t-2} < 2 \\ -1.00 + 0.80X_{t-2} + \varepsilon_t^{(3)}, & \varepsilon_t^{(3)} \sim \mathcal{N}(0, 0.32^2) & \text{if } 2 \leq X_{t-2} < 5 \\ -6.00 + 1.02X_{t-2} + \varepsilon_t^{(4)}, & \varepsilon_t^{(4)} \sim \mathcal{N}(0, 1.41^2) & \text{if } 5 \leq X_{t-2} \end{cases} \tag{5.4}$$

We simulate 800 points of the new model and plot them together with the original data.

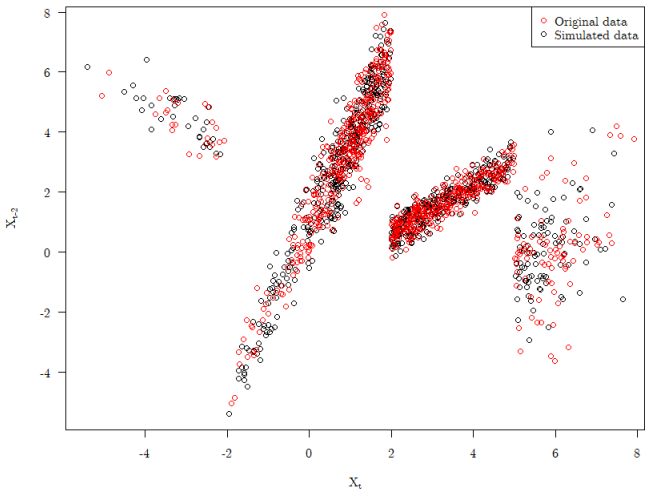


Figure 29 – Simulated versus original data

We see the simulated data fits the original data very well so we try to analyze the residuals.

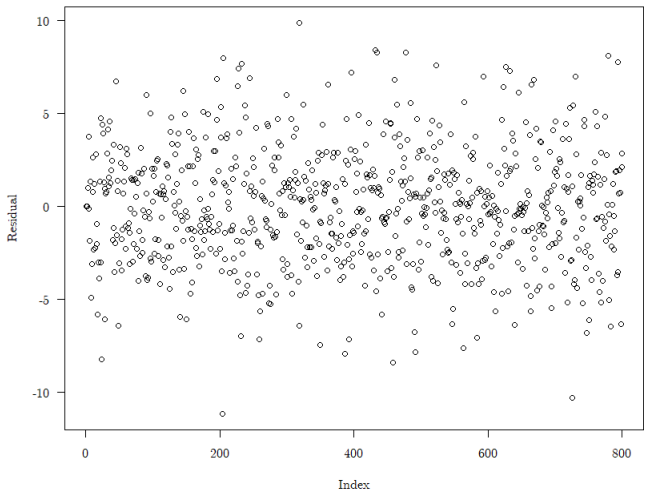


Figure 30 – Plot over the residuals

We see in figure 29 that the data looks very normal so we proceed with a histogram and qq-plot.

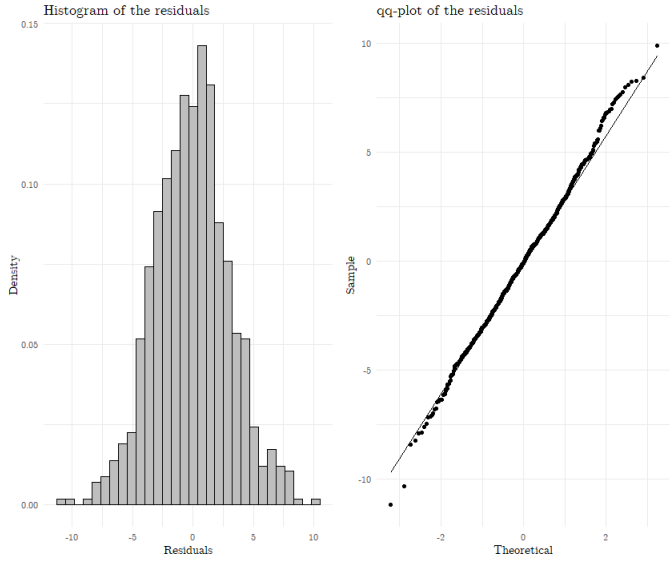
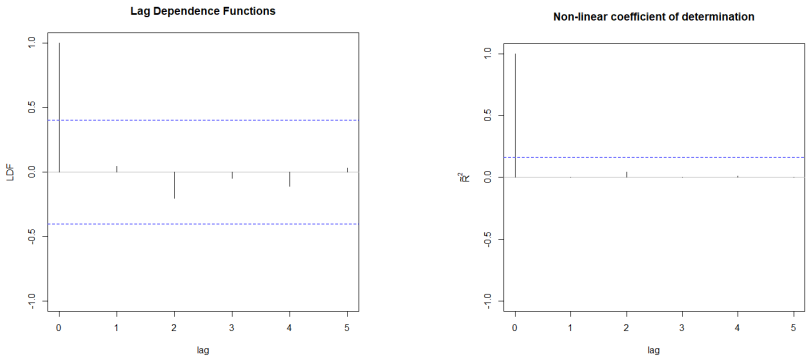


Figure 31 – Histogram and qq-plot over the residuals

Again the residuals are almost perfectly normal distributed but just for good measure we also try to investigate if any non-linear dependencies are left.



(a) Lag Dependence Function

(b) Non-linear coefficient of determination

Figure 32

We see that neither in the non linear analysis we find anything. Hence we conclude that model 5.4 fits the data well enough to accept it as a final model.

6 References

[1] H. Madsen and J. Holst, "Modelling non-linear and non-stationary time serie," *IMM*, 2016.

[2] D. Tjöstheim, "Non-linear time series: A selective review," *Scandinavian Journal of Statistics*, 1994.

[3] A. Schindler, "Bandwidth selection in nonparametric kernel estimation," *Faculty of Economic Sciences of the Georg-August-Universitat Göttingen.*