Chapter 4

Simulations

The main goal of this thesis is to estimate a regression function from data of the form (3.1), where the ε_i's are known to be time-correlated. In this chapter, we will perform a simulation study where we will use smoothing splines with model based penalties along with two of the smoothing parameter selectors described in the previous chapter. The motivation for this work comes from an environmental problem, namely ground-level ozone concentration. Although we will base our simulations on this problem, the techniques used here can very well be applied in other contexts.

4.1 Description

With this simulation study, we wish to show that combining model based penalized smoothing with a smoothing parameter selection technique that takes into account correlation will increase the accuracy of the estimation. Since the purpose of this thesis is to apply the proposed methods to environmental data analysis, we based our simulations on models motivated by our data set. We will, in the next chapter, analyse a data set of daily and monthly ozone levels. Such data are assumed to follow a periodic $\{1, sin, cos\}$ model while having correlated errors. For that reason, we use regression functions that are "similar" to the favoured parametric model ozone levels, namely $m(x) = \alpha_1 + \alpha_2 \cdot cos\,\omega x + \alpha_3 \cdot sin\,\omega x$. We chose $\omega = 8\pi$, with $x \in [0, 1]$, yielding four "cycles" in each data set. We implemented two methods of smoothing parameter selection: "Modified Cross-Validation" and "Blockwise Cross-Validation" that were described in sections (3.2.1) and (3.2.3) respectively. The optimal smoothing parameter for each of these two methods was obtained by using a *Quasi-Newton* algorithm to search for the location of MCV_{min} and BCV_{min}. In order to evaluate the performance of these methods, we will compare them with four commonly used techniques of analysis. Overall, we will use six methods in our simulation study:

LS: nonlinear least squares estimate based on the favoured parametric model assuming independent observations and $\omega = 8\pi$;

NLS: nonlinear least squares estimate based on the favoured parametric model assuming independent observations and where ω is estimated;

TS: time series method assuming $m(x) = \alpha_1 + \alpha_2 \cdot cos\,\omega x + \alpha_3 \cdot sin\,\omega x$ where $\omega = 8\pi$ is known. We also assume that the errors are correlated and follow a causal ARMA process. A more detailed description of that method will follow shortly.

IND: penalized least squares estimate based on the appropriate penalty $L = D^3 + \omega^2 D$ where $\omega = 8\pi$ is known and λ is chosen by ordinary CV assuming independent observations;

BCV: penalized least squares estimate based on the appropriate penalty $L = D^3 + \omega^2 D$ where $\omega = 8\pi$ is known and λ is chosen by blockwise cross-validation as described in section 3.2.3. The blocks were chosen by splitting the simulated data set into 4 differents groups, one group for each cycle;

MCV$_l$: penalized least squares estimate based on the appropriate penalty $L = D^3 + \omega^2 D$ where ω is known and λ is chosen by modified cross-validation as described in section 3.2.1. The values of l used with MCV$_l$ are $1, 2$ and 3 (remember that a value of $l = 0$ corresponds to ordinary CV, that is, IND).

Note that it is also possible to estimate the value of ω in the last four methods. For penalized least squares used in IND, BCV and MCV$_l$, Heckman and Ramsay (1996) describe four different approaches to estimating ω when it is assumed to be unknown. Due to limited time for computation, we only considered estimation of ω in NLS.

Now we will present in greater detail the pseudo-code for the time series method (TS) that was just mentioned, for estimating α_1, α_2 and α_3 in the parametric model.

1. Perform an ordinary least squares analysis on the data to get the estimate

$\hat{\boldsymbol{\alpha}}^{(0)} = \{\hat{\alpha}_1^{(0)}, \hat{\alpha}_2^{(0)}, \hat{\alpha}_3^{(0)}\}$.

2. Given $\hat{\boldsymbol{\alpha}}^{(k)}$, construct the residuals $\{\hat{Z}_i^{(k)}\} = \{Y_i - \hat{\alpha}_1^{(k)} - \hat{\alpha}_2^{(k)} \cdot cos\,\omega x_i - \hat{\alpha}_3^{(k)} \cdot sin\,\omega x_i\}$.

3. Check for serial correlation of $\{\hat{Z}_t^{(k)}\}$. If correlated, try fitting an autoregressive model of order p to $\{\hat{Z}_t^{(k)}\}$, where p is determined by looking at the partial auto-correlation function. Then use the fitted model to generate an estimated covariance matrix $\hat{\boldsymbol{\Sigma}}^{(k)}$. For example, if an AR(1) model is fitted, then

$$\hat{\boldsymbol{\Sigma}}_{ij}^{(k)} = \frac{|\hat{\phi}_1|^{|i-j|} \cdot \hat{\sigma}^2}{1 - \hat{\phi}_1^2}$$

where $\hat{\phi}_1$ is the estimate of the autoregressive coefficient and $\hat{\sigma}^2$ is the estimate of the variance. In the absence of serial correlation, STOP.

4. Given $\hat{\boldsymbol{\alpha}}^{(k)}$ and $\hat{\boldsymbol{\Sigma}}^{(k)}$, perform a weighted least squares analysis by $min_{\boldsymbol{\alpha}}(\mathbf{Y} - \mathbf{X}\boldsymbol{\alpha}^{(k)})'(\hat{\boldsymbol{\Sigma}}^{(k)})^{-1}(\mathbf{Y} - \mathbf{X}\boldsymbol{\alpha}^{(k)})$ to obtain $\hat{\boldsymbol{\alpha}}^{(k+1)}$. Here, $\mathbf{Y}$ and $\mathbf{X}$ are respectively the vector of $\{Y_i\}$ and the design matrix corresponding to the parametric model.

5. If $\|\hat{\boldsymbol{\alpha}}^{(k+1)} - \hat{\boldsymbol{\alpha}}^{(k)}\| < \delta$, where δ is the desired precision, STOP. Otherwise, return to step 2 using $\hat{\boldsymbol{\alpha}}^{(k+1)}$ instead of $\hat{\boldsymbol{\alpha}}^{(k)}$.

We used all six methods of estimation on regression functions that were "similar" to our favoured model. These regression functions were obtained by generating random functions. The procedure used for generating these random functions will be described in the next section.

We constructed 500 simulated data sets by generating the random functions $\mu_1, \ldots, \mu_{500}$. For each simulated data set, $j = 1, \ldots, 500$, we generated 100 data points

$$Y_{ij} = \mu_j(x_i) + \sigma\varepsilon_{ij} \quad i = 1, \ldots, 100 \tag{4.1}$$

where $x_i = i/100$, the ε_{ij}'s follow an AR(1) process with autoregressive coefficient $\phi = 0.5$ and σ is a positive real valued constant that was taken to be 15. (Note that we proceeded to the same simulations with $\phi = -0.5$.)

To quantify the results of each estimation method, we will use the following summary statistics:

PSE: Predictive Squared Error

$$PSE_j = \frac{1}{100}\sum_{i=1}^{100}(Y_{ij} - \hat{m}_j(x_i))^2 \;\; j = 1, \ldots, 500 \tag{4.2}$$

where the $\hat{m}_j(x_i)$'s are the values of the estimated regression functions at the time points $\{x_i\}$;

MSE: Mean Squared Error

$$MSE_j = \frac{1}{100}\sum_{i=1}^{100}(\hat{m}_j(x_i) - \mu_j(x_i))^2 \;\; j = 1, \ldots, 500. \tag{4.3}$$

4.2 Generation of Random Functions

In order to verify the efficiency of our methods, we wish to generate random functions that are close to the favoured parametric model. That choice leads us to the following penalty, $L = D^3 + \omega^2 D$, where the favoured model is spanned by the functions $1, cos\, \omega t, sin\, \omega t$. Therefore, we want to generate random processes $w(\cdot)$ and $\mu(\cdot)$ with $L\mu \equiv 0$ and

$$E(w(t)) \equiv \omega. \tag{4.4}$$

However, for the case where $w(\cdot)$ is not constant, there does not exist a closed form solution of the differential equation $L\mu = 0$. Thus, replace $L\mu \equiv 0$ with

$$E(L\mu) \equiv 0. \tag{4.5}$$

Let us now introduce some useful notation. Let $\epsilon(\cdot)$ be a white noise process with standard deviation σ_ϵ, $b(t) = \int_o^t \epsilon(s)\, ds$ be Brownian motion and $B(t) = \int_0^t b(s)\, ds$. Then, one can show that (4.4) and (4.5) are satisfied for $w(t) = \omega + b(t)$ and $\mu(t) = C\,[\alpha_0 + D^{-1} cos\,(\omega t + B(t))]$, where D^{-1} denotes the first anti-derivative operator. C is an arbitrary constant chosen to obtain data with reasonable magnitude. Here, α_0 is a normal random variable, independent of $B(\cdot)$, with mean zero and standard deviation σ_α and was chosen in order to avoid always having $\mu(0) = 0$.

In order to generate a curve μ, we need to generate an approximation of $B(\cdot)$. We do this by generating 100 independent normal random variables with mean zero and standard deviation σ_ϵ and we use their partial sums to define $b(\cdot)$. Next, we approximate $B(i/100)$, for $i = 1, \ldots, 100$, by numerical integration of $b(\cdot)$ using the trapezoidal rule with grid points $j/100$, $j = 1, \ldots, i$. Afterwards, we use the trapezoidal rule once more to approximate $D^{-1} cos\,(\omega t + B(t))$.

Figure 4.1 shows a sample of 3 of the 500 randomly generated curves. When generating those curves, we used $\omega = 8\pi$, $\sigma_\epsilon = 50$, $\sigma_\alpha = 0.01$, $\sigma = 15$

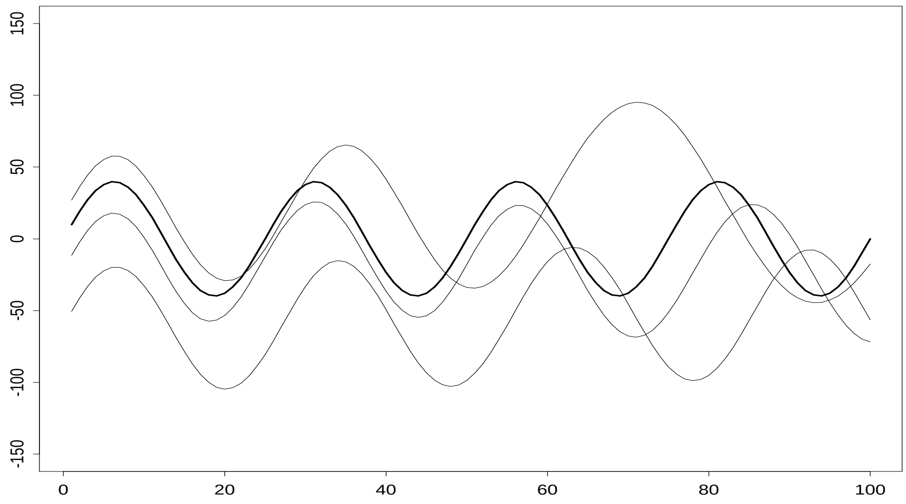

Figure 4.1: *Sample of Randomly Generated Curves. The solid line is the true curve.*

and $C = 1000$. We chose these values in order to obtain curves that are close to the favoured parametric model while still being different. The solid curve is the *true* curve and was generated using $\alpha_0 = 0$ and $B(t) \equiv 0$, that is, the true curve is $\mu(t) = 1000 \cdot D^{-1}cos(\omega t) = 1000 \cdot (sin(\omega t))/\omega$.

It is important to notice that since the simulated curves are not exactly parametric, this also means that some of these curves may not be periodic. Furthermore, in the event that a curve is roughly periodic, we are not assured that the period will be exactly 8π. Figure 4.1 demonstrates the fact that not all the randomly generated curves are periodic. In that figure, three curves, including the *true* curve, look periodic with four cycles while one curve is definitely not periodic and it has only three cycles, of unequal length.

4.3 Simulation Results

The results of the different methods with regards to the MSE and the PSE for positive and negative correlation are given in Tables 4.1 and 4.4, respectively.

Since the distributions of both the mean squared error and the predictive squared error are generally long-tailed, we also included summary statistics for the natural logarithm of the MSE and PSE. Figure 4.2 shows the distribution of the MSE and log(MSE) for the NLS and TS method when $\phi = 0.5$. This clearly demonstrates the skewness of the MSE score. Therefore, by taking the natural logarithm of the MSE, we diminish the influence of the skewness, thus showing the real behaviour of each estimating method. The histograms of the MSE and log(MSE) for the other methods are displayed in Appendix A. Although in our tables we provided the results of the MSE and PSE for each method, we will mainly focus on the MSE. Since the goal of our work is to to find a way of easily and efficiently estimating the regression function when the errors are correlated, we are more concerned about the performance of the methods with regards to the MSE. The PSE score, on the other hand, is not very meaningful in the context of this thesis since PSE helps us understand the behaviour of the methods in terms of predictive performance. The results of the PSE will however be somewhat meaningful when we will look at the number of parameters used by each technique. Indeed, an estimate that uses a large number of parameters will try to interpolate the data, and hence have a low PSE score. On the other hand, an estimate using a small number of parameters should have a large PSE score as it is not predicting

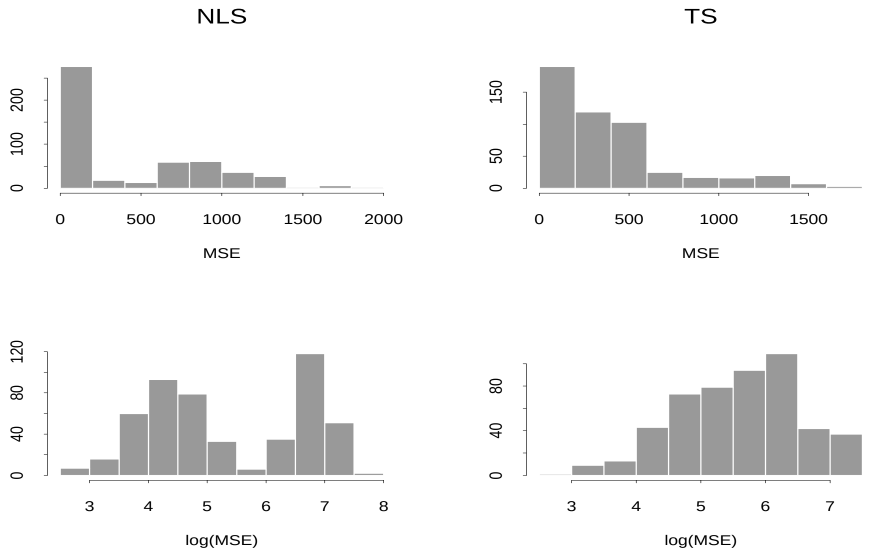

Figure 4.2: *MSE and log(MSE) for NLS (left) and TS (right) when* $\phi = 0.5$.

the observations well.

In the next two sections, we will analyse the results of the simulation study for positive and negative correlation. We will compare the log(MSE) results of the different methods via paired t-tests with a significance level of 5% to insure that the differences are significant. A paired t-test simply means that we apply the usual t-test to the differences of the log(MSE) scores for each pair of methods.

4.3.1 Positive Correlation

In terms of the log(MSE), all of the smoothing spline methods with model based penalty (IND, BCV and MCV_l), which are nonparametric methods of

Table 4.1: *Simulation Mean Results with* $\phi = 0.5$.

Method	**MSE (SE)**	**PSE (SE)**	**log(MSE) (SE)**	**log(PSE) (SE)**
LS	401.8 (16.1)	654.5 (16.8)	5.59 (0.04)	6.35 (0.02)
NLS	443.4 (20.6)	673.6 (21.3)	5.35 (0.06)	6.28 (0.03)
TS	398.3 (15.8)	651.5 (16.6)	5.59 (0.04)	6.34 (0.02)
IND	168.3 (3.2)	109.2 (2.5)	5.01 (0.02)	4.54 (0.03)
BCV	344.4 (16.4)	530.8 (18.9)	5.32 (0.05)	5.98 (0.04)
$\mathbf{MCV}_1$	111.8 (2.4)	172.6 (2.1)	4.60 (0.02)	5.11 (0.01)
$\mathbf{MCV}_2$	103.6 (2.5)	193.2 (2.9)	4.51 (0.02)	5.22 (0.01)
$\mathbf{MCV}_3$	97.8 (2.1)	440.2 (28.3)	4.47 (0.02)	5.61 (0.04)

estimation, outperformed the parametric methods (LS, NLS and TS). Part of the reason why LS, NLS and TS performed so poorly is due to the fact that we generated functions that were not exactly the favoured parametric model. Therefore, the parametric methods had a disadvantage right from the start since the parametric model that they were trying to fit was not exactly what the data were following.

Comparing the three parametric methods, we see that NLS does significantly better than both LS and TS. This can be explained by the fact that since it is estimating the period, NLS really has an advantage over the two other parametric methods. Figure 4.3 shows the histogram of the estimated values of ω obtained by NLS. In fact, since our simulated regression functions were not exactly the favoured parametric model, we generated regression functions that either were far from being periodic, or were roughly periodic, but perhaps with a "period" different from the one of the favoured parametric model, i.e., 8π. Therefore, in the case where the curves were roughly periodic

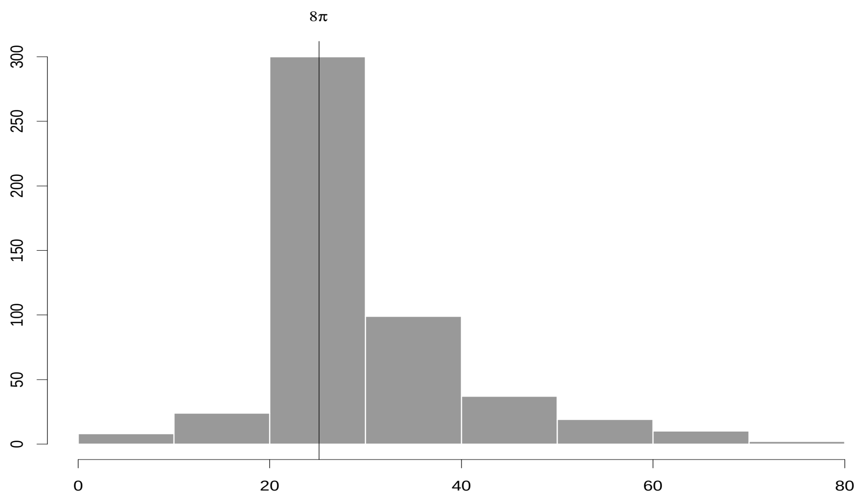

Figure 4.3: *Estimated values of ω for $\phi = 0.5$*

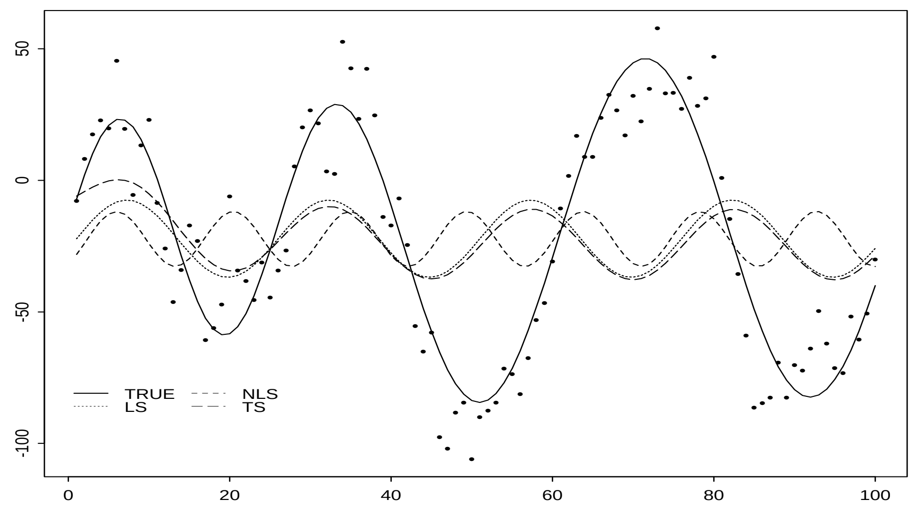

Figure 4.4: *Fits of Parametric Methods on Simulated Curve 6 for $\phi = 0.5$*

but with a period different from 8π, NLS was able to adapt more easily to the change in the period than TS and LS, hence performing better in terms of MSE. For curves that were far from periodic, the parametric methods generally performed poorly as they were restricted to a periodic parametric model. Figure 4.4 shows the fits of all parametric methods for one simulated curve. It is obvious that this curve is not periodic and it has only three cycles, of different durations. In this case, all parametric methods performed very poorly at estimating the underlying trend. LS and TS did not perform well since the specified ω of 8π forced them to give an estimate of the regression function with four cycles and NLS yielded $\hat{\omega} = 45.45$ which resulted in an estimate with approximately seven cycles. For the penalized smoothing spline methods, we did not estimate the period. Indeed, its value was fixed at 8π. However, this will not have an important effect since these methods are nonparametric, which means that they don't totally depend on the favoured parametric model.

By looking closely at Figure 4.2, we see that the distribution of log(MSE) of NLS is clearly bimodal. This bimodality is explained by the two features of the randomly generated curves mentioned above. Indeed, in the case where the curves are roughly periodic, NLS generally estimates the "period" quite accurately and yields a good estimate of the regression function. However, when the curves are far from periodic, NLS does not perform very well, hence yielding a higher MSE score which corresponds to the second mode in the MSE distribution.

Since TS is taking the correlation into account, one would think that this method would perform better than both LS and NLS which do not account for the correlation. As discussed previously, we know that TS does not perform better than NLS. However, TS does perform significantly better than LS. The mean results in Table 4.1 were rounded. Although TS and LS appear to have the same rounded mean log(MSE), the unrounded means are significantly different, by a paired t-test. The comparison between TS and LS is a good way of quantifying the effect of correlation on the estimation techniques. Indeed, the only difference between TS and LS is that TS takes correlation into account. Therefore, the fact that TS performs significantly better than LS indicates that accounting for correlation really helps improve the estimation.

When comparing the log(MSE) results, we can see that penalized smoothing with ordinary cross-validation (IND) did substantially worse than all the nonparametric methods that took correlation into account except for BCV. It is pretty obvious by looking at the log(MSE) values in Table 4.1 that IND is really affected by the correlation structure in the regression errors. Figures 4.5-7 show the fits of the different methods on the same simulated curve as in Figure 4.4. In Figure 4.5, we can see that IND yields an estimate of m that is too "wiggly", that is, the optimal smoothing parameter obtained by ordinary cross-validation is too small. In comparison, looking at the two other plots in Figures 4.6 and 4.7, we can see that the fits of all penalized methods are much smoother and, except for BCV (which we will discuss later), they follow the *true* regression curve quite faithfully. In a majority of the simulated data sets, the smoothing parameters chosen by BCV and MCV_l were bigger than

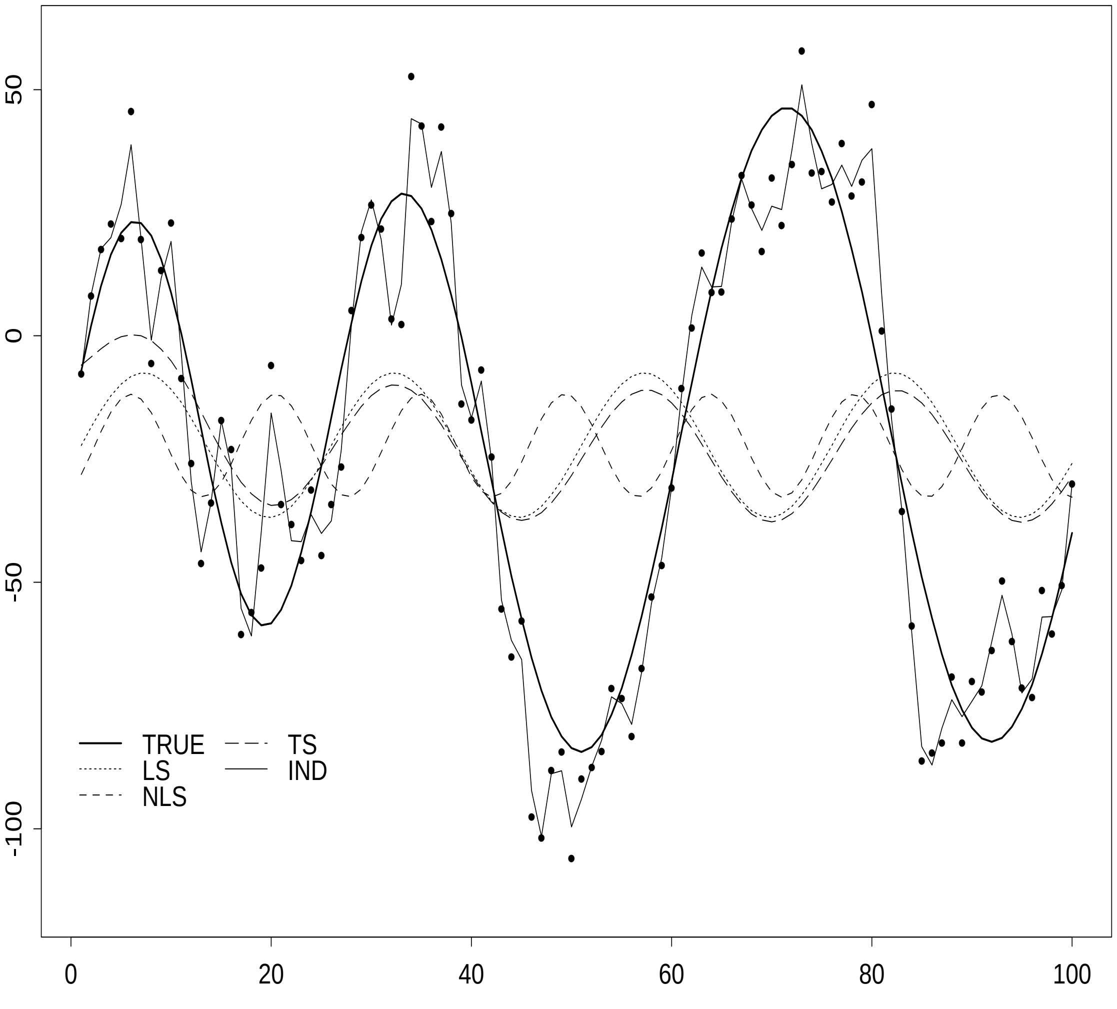

Figure 4.5: *Fits on Simulated Curve 6 for* $\phi = 0.5$

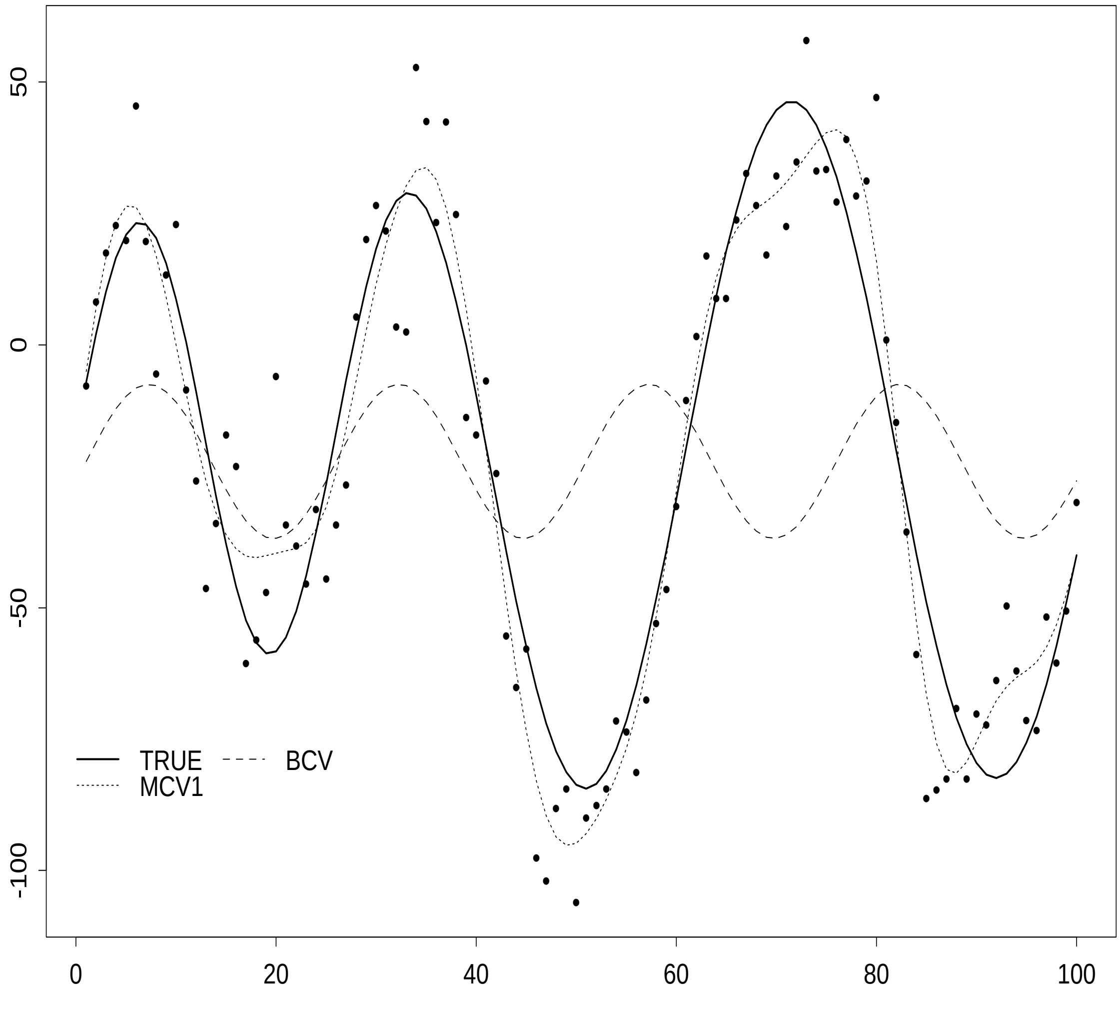

Figure 4.6: *Fits on Simulated Curve 6 for* $\phi = 0.5$

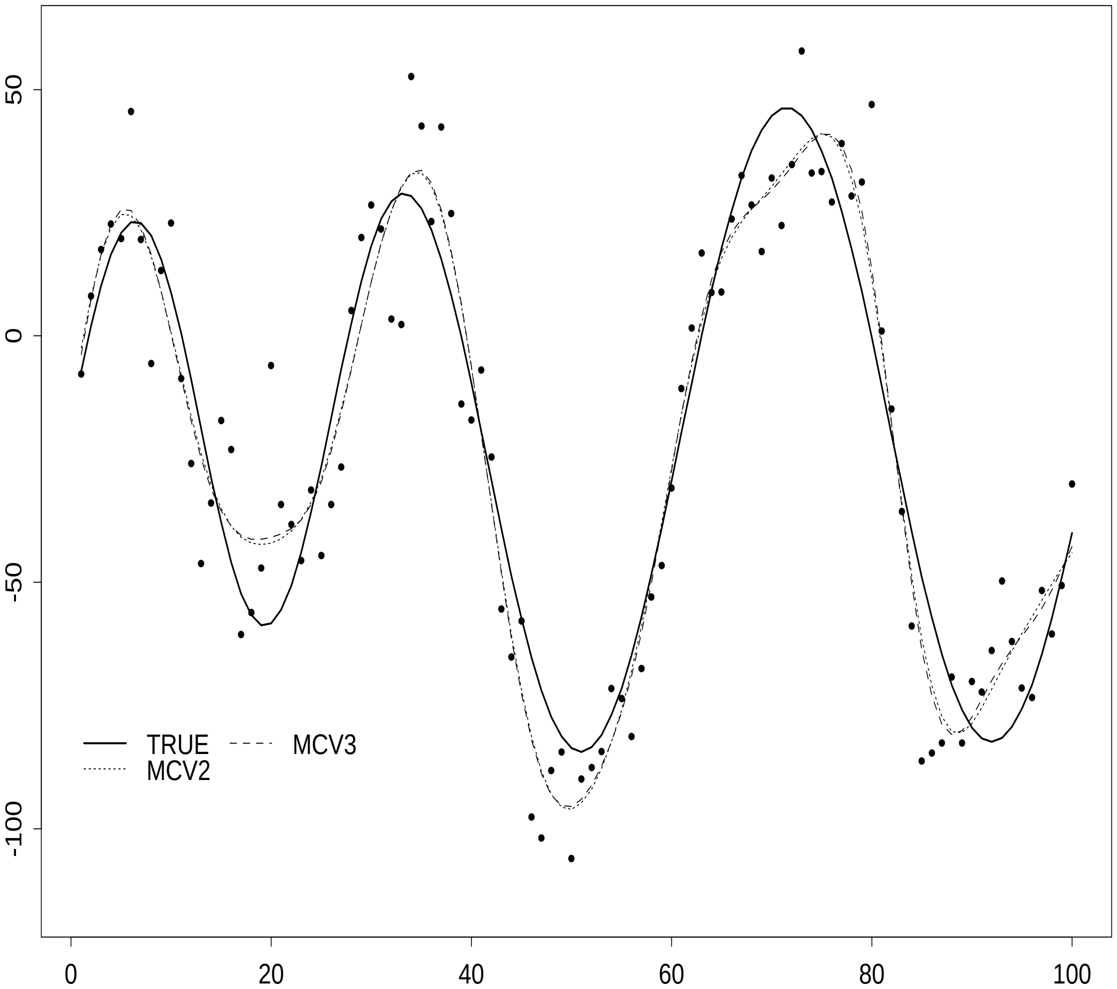

Figure 4.7: *Fits on Simulated Curve 6 for* $\phi = 0.5$

the one obtained by IND. This supports the discussion in section 3.1, that, for positive correlation, ordinary cross-validation has a tendancy to interpolate the data thus yielding a smoothing parameter that is too small.

The MCV_l methods performed significantly best in terms of log(MSE) score. In fact, for MCV_l, the log(MSE) is a decreasing function of l since the MCV_l methods perform better and better as the value of l increases, hence the better performance of MCV_3. This is due to the correlation structure as the bigger the value of l, the more MCV_l accounts for correlation thus alleviating its effect more efficiently. Indeed, we know that the residuals follow an AR(1) model where the $\mathrm{Cor}(\varepsilon_i, \varepsilon_{i+j})$, for $j = 1, 2$ and 3, is 0.50, 0.25 and 0.125 respectively. It is clear that observations adjacent in time are significantly correlated while observations three time points apart are weakly correlated. This is why MCV_2 and MCV_3 should perform best as they are eliminating more correlation than MCV_1. In practice, this is what happened. Indeed, MCV_2 and MCV_3 respectively performed best for 28% and 44% of the simulated data sets, for a total of 72% of the cases. However, when estimating $\mathrm{Cor}(\varepsilon_i, \varepsilon_{i+j})$ and the order of the autoregressive model of the residuals, we could not conclude on a data-based rule of selection for l as there was no clear pattern. For example, in simulated data set #22, MCV_1 performs best even though the estimated autoregressive order is six while in site #10, MCV_3 performs best even though the estimated autoregressive order is one and $\mathrm{Cor}(\varepsilon_i, \varepsilon_{i+3})$ is estimated as 0.05.

A more intuitive way to compare the performance of the different estimators is to look at the number of parameters used by each method. Please

Table 4.2: *Number of Parameters Used,* $\phi = 0.5$.

Method	**mean**	**median**	**S.E.**
IND	29.87	29.69	0.66
BCV	7.57	3.10	0.36
$\mathbf{MCV}_1$	14.79	13.91	0.25
$\mathbf{MCV}_2$	12.62	12.15	0.21
$\mathbf{MCV}_3$	11.50	11.33	0.18

refer back to section 2.2.3 for the definition of the effective number of parameters used by an estimate. Table 4.2 summarizes the number of parameters used by all penalized methods. The table does not include the parametric methods since they use a fixed number of parameters. Indeed, LS, NLS and TS always use three, four and three parameters respectively. IND uses on average significantly more parameters than any of the penalized methods that take correlation into account. In fact, IND uses more than twice the number of parameters than the other nonparametric methods. This explains why the estimate of m using IND is usually not as smooth as that of the other methods since IND tries to interpolate the data. Indeed, IND has the lowest mean log(PSE) which confirms that IND tracks the data too closely. One can also notice, by looking at Table 4.2, that the number of parameters used by MCV_l is a decreasing function of l. This shows that when the value of l increases, the smoothing parameter chosen by MCV_l increases, thus yielding an estimate of m that is closer to the favoured parametric family. Therefore, if the penalty is chosen correctly by the data analyst, the resulting estimate of m would be smoother and more precise.

The BCV method yields significantly the lowest effective number of parameters among all nonparametric methods. However, as discussed earlier, BCV does not perform well in terms of MSE. In fact, in approximately 60% of the simulated data sets, BCV uses less than 3.3 parameters which is roughly equivalent to using an infinite smoothing parameter. Remember that a smoothing spline with an infinite smoothing parameter lies in the kernel of the penalty, that is, it follows the favoured parametric model. Thus, in our case when BCV uses approximately three parameters, it becomes roughly the same method as LS. But why would BCV use so few parameters? Figure 4.8

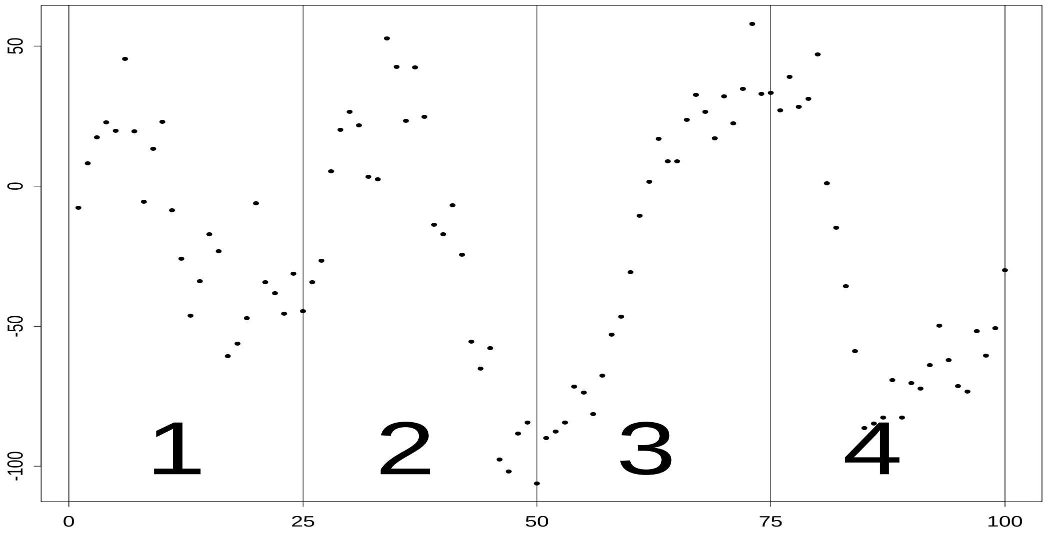

Figure 4.8: *Behaviour of BCV.*

helps us answer that question. The four vertical lines on the plot represent the boundaries of the four sections, or "cycles", used in the BCV method. BCV

Table 4.3: *Correlation between Number of Parameters Used,* $\phi = 0.5$.

METHODS	**BCV**	$\mathbf{MCV_1}$	$\mathbf{MCV_2}$	$\mathbf{MCV_3}$
IND	-0.054	0.288	0.119	0.040
BCV	-	-0.046	-0.077	-0.099
$\mathbf{MCV_1}$	-	-	0.565	0.265
$\mathbf{MCV_2}$	-	-	-	0.477

uses data in the last three sections to predict data in the first section. It uses data in the first, third and fourth section to predict data in the second section and so on. Obviously, when the data are far from periodic, that is, the four sections are different from each other (as is clearly the case in Figure 4.8), BCV will oversmooth the data in order to predict the deleted section, using a large smoothing parameter or equivalently few parameters. This behaviour of BCV indicates that it is not doing very well. As mentioned, BCV is oversmoothing the data when the four sections are quite different, thus leading to an estimate close to the parametric one. Note from Figure 4.6, BCV has fit a periodic model to the non-periodic data, but the amplitude is fairly small, and hence close to an average of the data. On the other hand, if the data in the four sections are similar, BCV will choose a small smoothing parameter and tend to interpolate the data.

One might wonder if there is a correlation between the optimal smoothing parameters obtained by the different methods used. To investigate this question, we look at the correlation between the number of parameters used, as displayed in Table 4.3. There is a noticeable correlation between the number of parameters used by MCV_l for its different values of l. In fact, for MCV_l, the correlation is stronger between two consecutive values of l. This is not surprising, since leaving out, say $2 \cdot 2 + 1 = 5$ points in the CV score should be similar to leaving out $2 \cdot 3 + 1 = 7$ points. Interestingly, the number of parame-

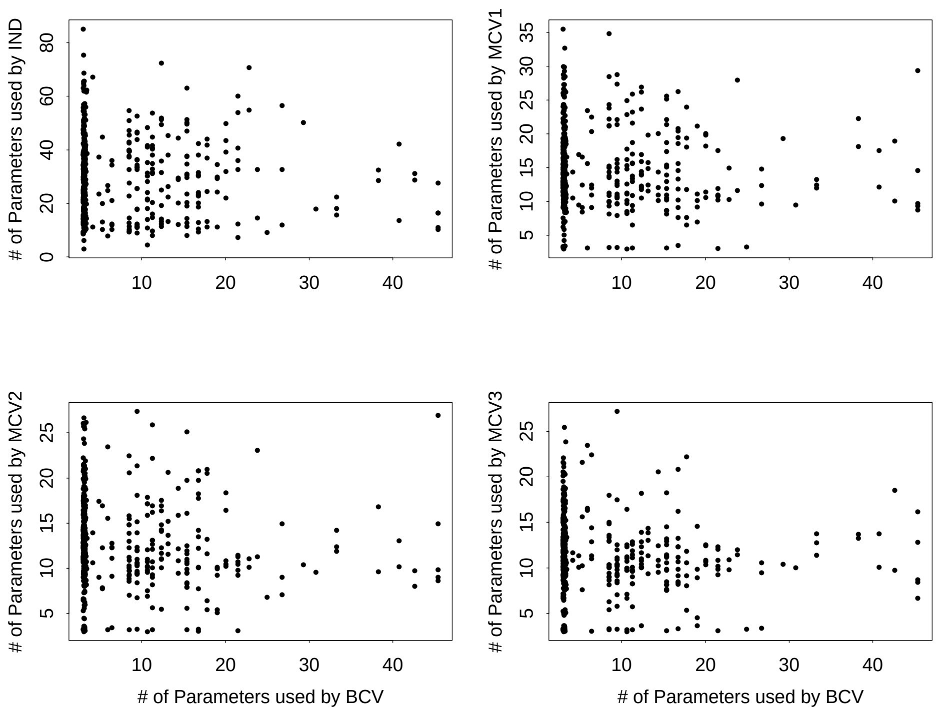

Figure 4.9: *Scaterplots of # of Parameters used by INd, MCV_l vs BCV.*

ters used by BCV is slightly negatively correlated with all the other penalized methods. To explain the negative correlation between BCV and the other penalized methods one has to remember that BCV uses a small number of parameters when the four groups in the data set are different. However, in that case, MCV_l and IND will generally use a bigger number of parameters since the data appear to be further away from the default parametric model. The correlation between BCV and the other methods is not very strong though, as can be seen by looking at Figure 4.9.

4.3.2 Negative Correlation

Table 4.4: *Simulation Mean Results with $\phi = -0.5$.*

Method	MSE (SE)	PSE (SE)	log(MSE) (SE)	log(PSE) (SE)
LS	380.8 (16.0)	672.5 (16.3)	5.48 (0.05)	6.39 (0.02)
NLS	394.1 (20.9)	682.0 (21.0)	4.15 (0.07)	6.31 (0.03)
TS	379.4 (15.9)	671.1 (16.1)	5.48 (0.05)	6.39 (0.02)
IND	18.0 (0.4)	295.4 (2.5)	2.80 (0.02)	5.67 (0.01)
BCV	315.4 (17.1)	597.1 (17.7)	4.80 (0.07)	6.21 (0.03)
MCV$_1$	17.3 (0.3)	280.2 (2.4)	2.77 (0.02)	5.62 (0.01)
MCV$_2$	15.0 (0.3)	289.2 (2.5)	2.60 (0.02)	5.65 (0.01)
MCV$_3$	14.8 (0.3)	287.1 (2.5)	2.59 (0.02)	5.64 (0.01)

For the case of negative correlation, once again LS, NLS and TS perform very poorly compared to the other methods. NLS performs significantly better than TS and LS, and TS and LS cannot be significantly differentiated. The reason for the superior performance of NLS compared to the two other parametric methods can be explained as in the positive correlation case. (See Figure 4.10 for the histogram of the $\hat{\omega}$'s in the case of negative correlation.)

In terms of log(MSE), all penalized methods, except for BCV, performed very well. Surprisingly, IND, with log(MSE) still significantly higher than MCV$_l$, is much closer to the methods that take correlation into account than in the positive correlation case. Indeed, on average, the fits of all penalized methods but BCV are very close to the *true* regression function and

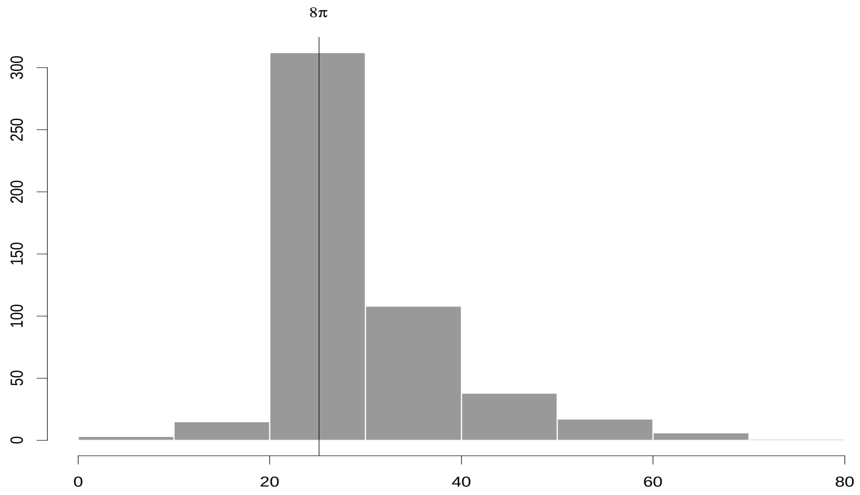

Figure 4.10: *Estimated values of ω for $\phi = -0.5$*

they are very similar to each other (see Figures 4.11-13, which use the same simulated curve as Figures 4.5-7). Once again, as shown in Figure 4.11, the NLS method did not succeed in estimating the period and yielded an estimate of the regression function that was far from the true one.

The MCV_l methods for $l = 2$ and 3 were not significantly different and overall they performed best. The similar performance of MCV_2 and MCV_3 is due to the fact that the correlation structure of the residuals is not as strong as in the previous section. Here again, we are led to the same conclusions as in the previous section when trying to find a selection rule for an optimal value of l.

Again, just like in the previous section, BCV is the penalized method that performs the worst in terms of log(MSE). The same conclusions as in the

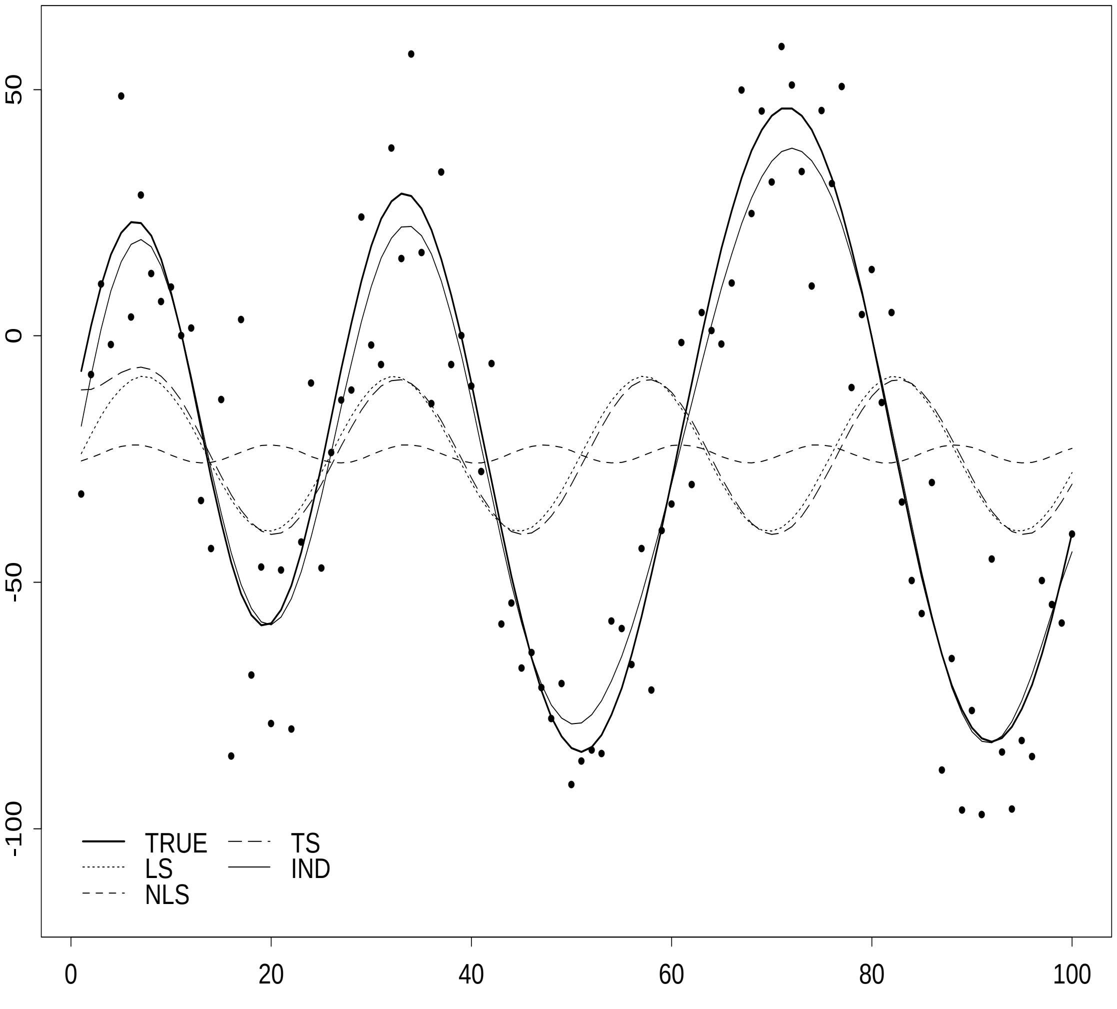

Figure 4.11: *Fits on Simulated Curve 6 for* $\phi = -0.5$

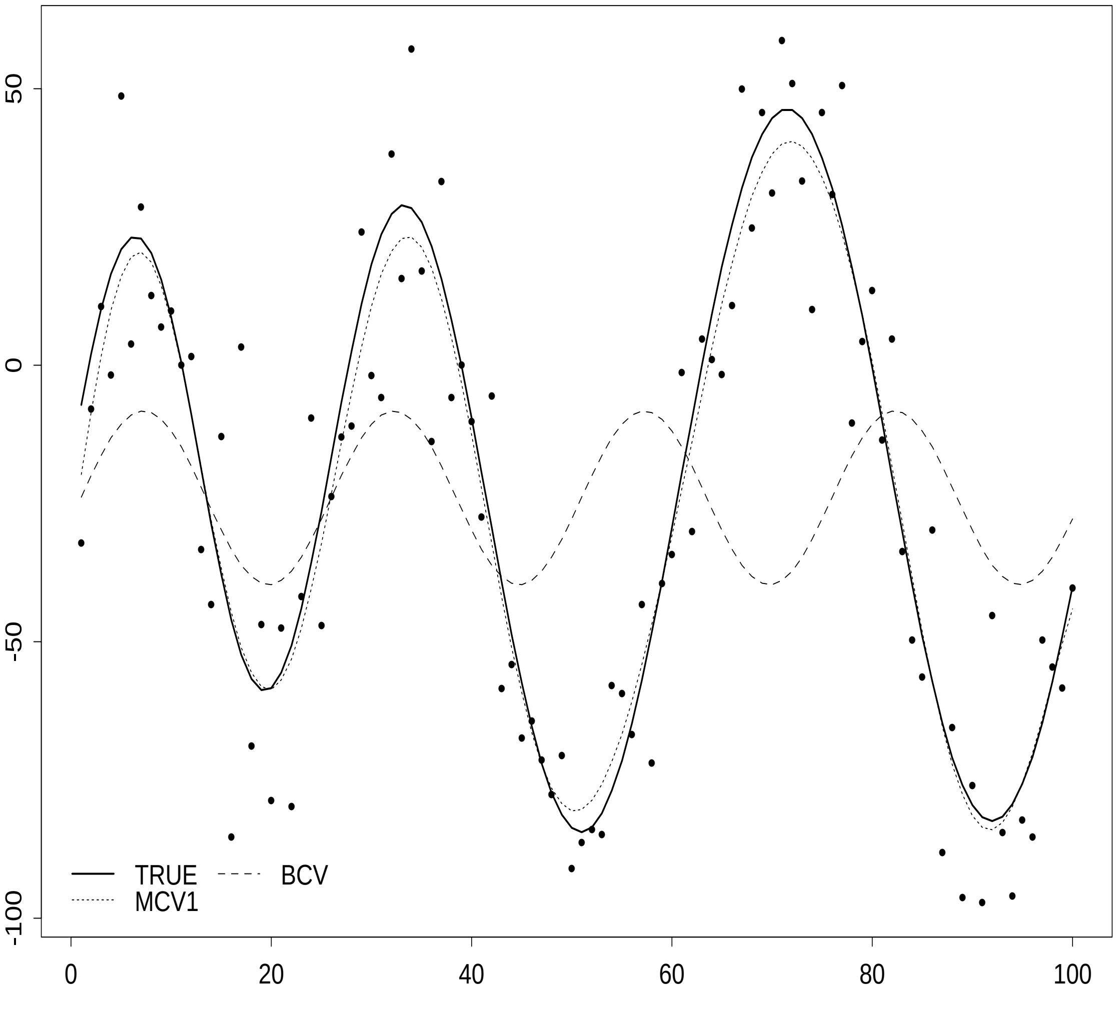

Figure 4.12: *Fits on Simulated Curve 6 for* $\phi = -0.5$

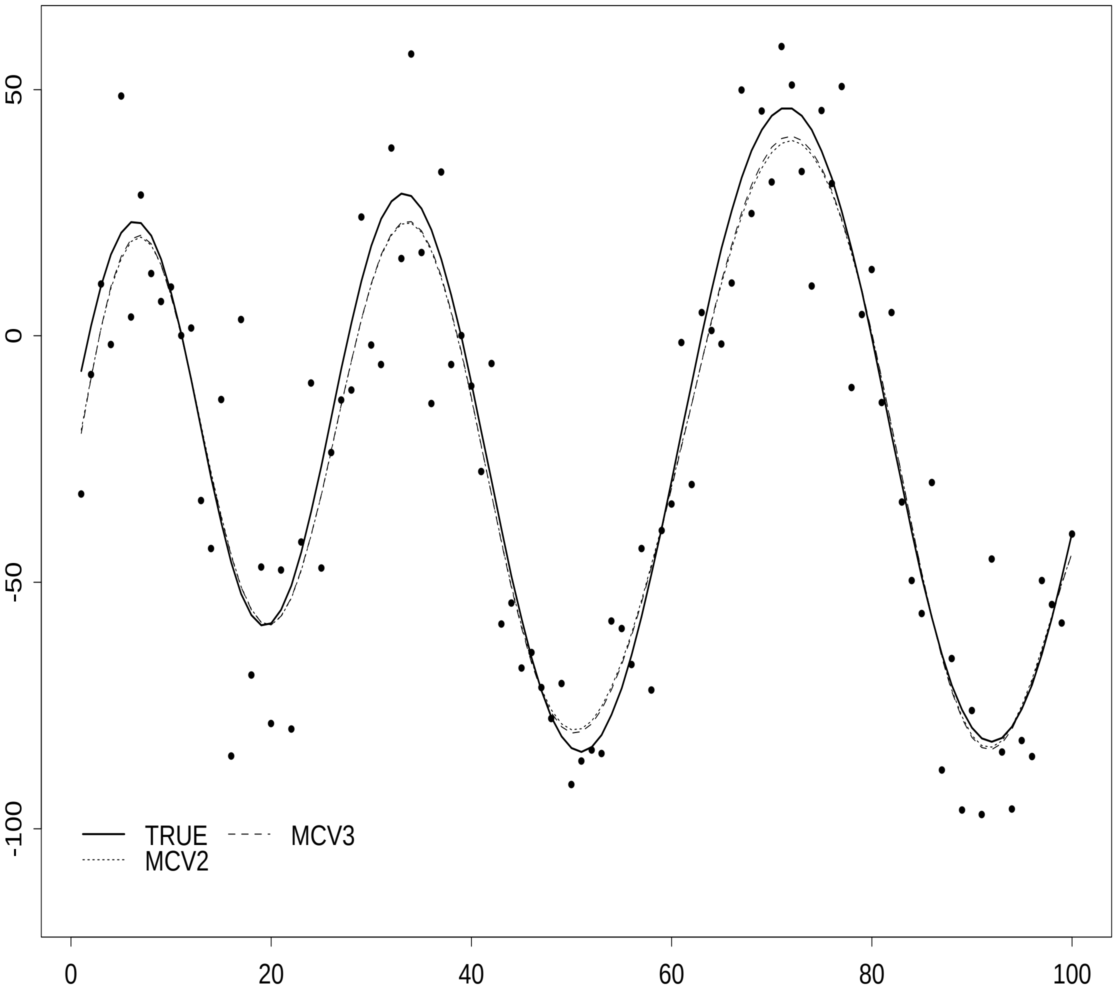

Figure 4.13: *Fits on Simulated Curve 6 for* $\phi = -0.5$

Table 4.5: *Number of Parameters Used, $\phi = -0.5$.*

Method	**mean**	**median**	**S.E.**
IND	10.35	10.79	0.11
BCV	7.13	3.04	0.37
MCV$_1$	16.42	16.12	0.19
MCV$_2$	11.77	12.27	0.12
MCV$_3$	12.60	12.91	0.12

positive correlation case arise here. Indeed, BCV still tends to oversmooth the data when the four groups are different, hence yielding a high MSE score.

As seen in Table 4.4, the effective number of parameters for MCV$_l$ is no longer a decreasing function of l. MCV$_1$ uses the most parameters among all smoothing methods. MCV$_2$ uses on average significantly fewer parameters than MCV$_1$, as one might expect, but also, due to the weaker correlation structure, uses significantly fewer parameters than MCV$_3$. BCV uses significantly less parameters than any other nonparametric method as it generally oversmooths the data.

The reason why most of the smoothing methods were roughly comparable is that negative correlation, while affecting the optimal smoothing parameter, does not have the same impact on the estimated regression function as positive correlation. IND is still influenced by the correlation structure. However, the effect of correlation does not have the same results on the estimate given by IND. Indeed, by looking once again at Table 4.5, one can see that IND yields on average a significantly smaller number of parameters used, that is, a bigger smoothing parameter, than MCV$_l$. Recall that IND is equivalent to ordinary cross-validation. Thus, the fact that IND leads to a bigger smoothing parameter when $\phi = -0.5$ is not surprising, as it agrees with what was mentioned in section 3.1. IND oversmooths the data and this shows up in its mean log(MSE) as it is higher than the one for MCV_l. However, we must not forget that we are now working in the context of smoothing splines with model based penalty. Remember section 2.2.2 where we introduced the notion of smoothing splines. We mentioned that the minimizer of (2.32) consists of two distinct parts: a penalty on the lack of fit and a penalty on the roughness of the estimating function. Therefore, by selecting a smoothing parameter bigger than the optimal one, IND in fact leads to an estimate that is closer to the least squares estimate of the kernel of P. Since the simulated functions are similar to the favoured parametric model of the chosen penalty P, the effect of correlation is causing IND to yield a smoothing parameter that results in an estimate of the regression function that is closer to the favoured parametric model.

Since the influence of correlation on IND is in some way alleviated by the appropriate choice of the penalty, it is not surprising to obtain high correlation between the number of parameters used by IND and MCV_l as these methods do very comparably. For the same reason as in the case where $\phi = 0.5$, the number of parameters used by BCV is still negatively correlated with the other nonparametric methods.

Table 4.6: *Correlation between Number of Parameters Used, $\phi = -0.5$.*

METHODS	**BCV**	**MCV$_1$**	**MCV$_2$**	**MCV$_3$**
IND	-0.275	0.417	0.916	0.831
BCV	-	-0.082	-0.232	-0.262
MCV$_1$	-	-	0.402	0.494
MCV$_2$	-	-	-	0.771

4.4 Conclusions

With this simulation study, we saw that both positive and negative correlation have an effect on the selection of the smoothing parameter. We saw that MCV$_l$ always performs best no matter if the data are positively or negatively correlated. Obviously, the choice of l is of importance as it can help improve on the accuracy of the estimation. Indeed, if the chosen value of l is too small, MCV$_l$ will not sufficiently alleviate the effect of the correlation. An extreme example that illustrates this is the results obtained via IND. Remember that IND is in fact equivalent to MCV$_l$ with $l = 0$. We saw, in the simulation study, that IND is affected by the correlation structure and has a tendancy to yield “wiggly” estimates in the case of positive correlation and to oversmooth estimates when the correlation is predominantly negative. On the other hand, the effect of an excessively large value of l is not clear since we only performed the simulation study with the values of $l = 1, 2, 3$. We saw earlier that the effect of the correlation at lag three was still present, therefore a value of $l = 3$ cannot be considered too large. The simulation study shows that the MSE decreases as the value of l increases and presumably an optimal value of l

would be one that takes a good amount of correlation into account, without leading to an estimate that oversmooths or interpolates the data. However, further reseach on the effect of a large l should be done before making any conclusion.

We consider two possible "Rules of thumb" that one might use in order to choose the value of l. The first one simply uses a value of $l = \hat{p}$ where $\hat{p}$ is the estimated order of the autoregressive model of the residuals. However, this rule does not seem appropriate in the simulation study. Indeed, in the simulation study with $\phi = 0.5$, 97% of the estimated autoregressive models are of order three or more, thus one should choose a value of at least $l = 3$. However, in the simulation study, $l = 1$ and $l = 2$ outperform $l = 3$ in terms of log(MSE) for 56% of the simulated data sets.

Alternatively, one might consider choosing l using estimates of $\mathrm{Cor}(\varepsilon_i, \varepsilon_{i+j})$. In that case, a value of $l = k$ is chosen, where k is the largest value of j for which $\widehat{\mathrm{Cor}}(\varepsilon_i, \varepsilon_{i+j}) < C$, with C a constant chosen by the data analyst. For the simulation study, this rule works well when using the true correlations but fails when we use the estimated correlations.

We have evidence that BCV will perform poorly if the data are not periodic with similar cycles. We also noticed that parametric techniques perform poorly compared to MCV_l since they are not flexible enough to track all the fine features of the data.

Another important point that came out of this simulation study is that the effect for both types of correlation is different. In fact, the results indicate that, as long as the penalty is correctly chosen, negative correlation will not have as large an impact on the performance of IND. It is still important to remember that, for both positive and negative correlation, MCV_l did significantly better than IND. Hence there is an obvious advantage to accounting for correlation. However, a key point is to notice that unlike positive correlation, which highly affects the estimation methods even if the penalty is chosen appropriately, the effect of negative correlation may also be alleviated by correctly choosing the favoured parametric model.