Cecchinato, Nedda and Wolff, Rodney C. (2008) Edgeworth expansion for the sample autocorrelation function.

Edgeworth expansion for the sample autocorrelation function

Nedda Cecchinato, Rodney Wolff

School of Mathematical Sciences
Queensland University of Technology

1 Introduction

Fractionally integrated process, introduced independently by Granger and Joyeux (1980) and Hosking (1981), exhibit second-order dependence structures of rich variety, and stir much interest by way of their mathematical properties and their applications in modelling real phenomena. Their mathematical complexity offers significant challenges in deriving estimates of parameters relating to the long memory behaviour, both in parametric and non-parametric models, with the latter having slower convergence properties. Some seminal papers include those by Yajima (1985), Fox and Taqqu (1986), and Dahlhaus (1988, 1989) on parametric estimation, and by Hurst (1951), Geweke and Porter-Hudak (1983), Robinson (1995) and Hurvich et al. (1998).

Where analytical methods are unavailable or intractable, bootstrap methods have enabled significant advances in inference. Starting with Künsch (1989) block bootstrap, model-based time domain inference has benefited from such key developments as the local bootstrap (Paparoditis and Politis, 1999), and the sieve bootstrap (Kreiss, 1992). The non-parametric approach has been treated by Franke and Härdle (1992) and Dahlhaus and Janas (1996). Frequency-domain approaches are embraced most successfully by the so-called phase scrambling methods (such as in Theiler et al., 1992). The limitation with most current theory is that it is limited to short-range dependent processes, or limited kinds of long memory processes.

We have in mind to consider the ACF bootstrap (as it is called), based on a result of Ramsey (1974), which generates a surrogate series X_t^* from a given series through the series' sample autocorrelation function $\hat{\rho}_k$, using the Durbin-Levinson algorithm. In particular, we wish to assess formally the validity of ACF bootstrap by showing that the surrogate autocovariance $\hat{\gamma}_k^*$ function and autocorrelation $\hat{\rho}_k^*$ function, i.e., the second order functions of $\{X_t^*\}$, converge in some appropriate mode to the sample autocovariance $\hat{\gamma}_k$

function and autocorrelation $\hat{\rho}_k$ function of the given series, respectively, and then also to the theoretical values γ_k and ρ_k of the data generating process.

To be precise, we distinguish between *weak long memory* (weakly dependent process) with $0 \leq d \leq 0.25$ and *strong long memory* (strongly dependent process) with $0.25 < d < 0.5$, where d is the order of fractional integration. It is know that the value of d determines the asymptotic behaviours of sample autocovariance and autocorrelation functions. The sample autocovariance and autocorrelation functions of weakly dependent processes are asymptotically Normally distributed, whereas their limiting distribution for strongly dependent processes, the Rosenblatt distribution, is complicated and difficult to handle.

We wish to study the normalised quantities $C_k = n^{1-2d}(\hat{\gamma}_k - \gamma_k)$ and $R_k = n^{1-2d}(\hat{\rho}_k - \rho_k)/(1-\rho_k)$ for strong long memory processes with the help of Edgeworth expansions (Hall, 1992). The aim of the study is to understand how the sample autocovariance and autocorrelation functions behave asymptotically for increasing values of d, and to identify the convergence rate using computational means, particularly in the region $0.25 < d < 0.5$. Further, we investigate the convergence of the bootstrap autocorrelation function for long memory processes. We compared the sample and the bootstrap autocorrelation functions in terms of standard deviation and bias.

The paper is organised as follows. In Section 2, we briefly introduce Edgeworth and Cornish-Fisher expansions and we study, by means of extensive simulation, the properties of sample autocovariance and autocorrelation functions of fractionally integrated processes. In Section 3 we show that the ACF bootstrap can replicate the second order dependence structure of any long memory processes ($0 < d < 0.5$) with Gaussian and non-Gaussian innovations. We conclude the paper and propose some future development in Section 4.

2 Edgeworth and Cornish-Fisher expansions

Asymptotic Normality is a common and desirable property of estimators. However this is not always the case and Edgeworth expansions can be a useful tools to investigate or correct asymptotic distributions. Not all estimators satisfy assumptions for a central limit theorem or sometimes convergence is so slow that the Normal approximation turns out to be very poor. Thus, we investigate if it is possible to correct with Edgeworth expansions the asymptotic distribution of the sample autocovariance and autocorrelation functions of fractionally integrated processes with $0.25 < d < 0.5$. We aim to find how many terms of the expansion really influence the convergence, and how far from Normality sample autocovariance and autocorrelation functions of strong long memory processes lie.

The general formula for an Edgeworth expansion to approximate distributions of estimates $\hat{\theta}$ of unknown quantities θ_0 is given by

$$P\left\{n^{1/2}\left(\frac{\hat{\theta}-\theta_0}{\sigma}\right) \leq x\right\}$$
$$= \Phi(x) + n^{-1/2}p_1(x)\phi(x) + \ldots + n^{-j/2}p_j(x)\phi(x) + \ldots,$$

where n is the sample size, σ the standard deviation of $\hat{\theta}$, $\Phi(\cdot)$ and $\phi(\cdot)$ are the standard Normal distribution and density function, respectively, $p_j(\cdot)$ is a polynomial depending on cumulants up to order $3j-1$ and is an odd or even function according to whether j is even or odd, respectively (Hall, 1992).

In our case we want to study the normalised quantities $C_k = n^{1-2d}(\hat{\gamma}_k - \gamma_k)$ and $R_k = n^{1-2d}(\hat{\rho}_k - \rho_k)/(1-\rho_k)$ because we know that they are asymptotically distributed as a Rosenblatt distribution (see Albin, 1998a,b). We aim to see how far from Normality these two quantities depart and if the first two terms of Edgeworth expansions correct the non-Normality adequately for the following reasons. Higher order correction terms become very unstable because of all the cumulants we need to estimate and, in any case, Hall (1992) (pg. 94) already warned that the results based on high order correction are unattainable. In the literature there are not many papers on this topic, besides the work of Hosking (1996). The Rosenblatt distribution is quite complicated and if the sample distributions are not very far from Normality then it could be easier to prove that the ACF bootstrap is a consistent method for long memory Gaussian processes.

From the paper of Hosking (1996), we know that the normalisation constant is n^{1-2d} and we approximate the asymptotic distributions by Edgeworth expansion as

$$P(C_k^* \leq z) = \Phi(z) + \textstyle\sum_{j=1}^{\infty} \left(n^{1-2d}\right)^{-j} P_j(z)\phi(z),$$
$$P(R_k^* \leq z) = \Phi(z) + \textstyle\sum_{j=1}^{\infty} \left(n^{1-2d}\right)^{-j} p_j(z)\phi(z),$$

where $C_k^* = C_k/\sqrt{\text{var}(C_k)}$, $R_k^* = R_k/\sqrt{\text{var}(R_k)}$ and P_j and p_j are the usual orthogonal polynomials of Edgeworth expansions. In the simulation study we do not divide C_k and R_k by their standard deviations: only when asymptotic Normality is assessed the variance is given by the well known Bartlett's formulas (see Bartlett, 1946). However, the scale in not important because we compare the Monte Carlo distribution with the Normal distribution and the Normal distribution corrected with Edgeworth expansion in terms of quantiles, thus we do not need to divide by the standard error. For our purpose it is more useful to have an approximation of the quantiles of the distribution to have a graphical insight through Q-Q plots. They can be calculated through Cornish-Fisher expansions, the inverse formula of

Edgeworth expansions, given by

$$C^*_{k,\alpha} = z_\alpha + \sum_{j=1}^{\infty} \left(n^{2d-1}\right)^j Q_j(z_\alpha), \tag{1}$$
$$R^*_{k,\alpha} = z_\alpha + \sum_{j=1}^{\infty} \left(n^{2d-1}\right)^j q_j(z_\alpha), \tag{2}$$

where $C^*_{k,\alpha}$, $R^*_{k,\alpha}$ and z_α are the α-level quantiles of C^*_k, R^*_k and of the Normal distribution respectively, and Q_j and q_j are of degree at most $j+1$, odd for even j and even for odd$j$, and depend on cumulants only up to order $j+1$ (for more detail, see Hall, 1992).

We compare the distributions of C_k and R_k with three different levels of approximations based on formulas (1) and (2). The first is the Normal approximation where we shall not consider any correction. In the second and third cases, respectively, we consider up to the first and up to the second terms:

$$\begin{aligned} C^*_{k,\alpha} &\doteq_1 C_{k,\alpha} = z_\alpha + \left(n^{2d-1}\right) Q_1(z_\alpha), \\ R^*_{k,\alpha} &\doteq_1 R_{k,\alpha} = z_\alpha + \left(n^{2d-1}\right) q_1(z_\alpha), \end{aligned} \tag{3}$$
$$\begin{aligned} C^*_{k,\alpha} &\doteq_2 C_{k,\alpha} = z_\alpha + \left(n^{2d-1}\right) Q_1(z_\alpha) + \left(n^{2d-1}\right)^2 Q_2(z_\alpha), \\ R^*_{k,\alpha} &\doteq_2 R_{k,\alpha} = z_\alpha + \left(n^{2d-1}\right) q_1(z_\alpha) + \left(n^{2d-1}\right)^2 q_2(z_\alpha). \end{aligned} \tag{4}$$

The coefficients Q_1 and Q_2 are given by

$$\begin{aligned} Q_1(x) &= -P_1(x) \\ Q_2(x) &= P_1(x)P_1'(x) - \frac{1}{2}xP_1(x)^2 - P_2(x), \end{aligned}$$

where P_1 and P_2 are the Edgeworth expansion's coefficients with

$$\begin{aligned} P_1(x) &= -\frac{1}{6}k_3(x^2-1) \\ P_2(x) &= -x\left\{\frac{1}{24}k_4(x^2-3) + \frac{1}{72}k_3^2(x^4-10x^2+15)\right\} \end{aligned}$$

and k_3 and k_4 are the third and fourth order cumulants

$$k_3(\hat{\gamma}_k) = E\{(\hat{\gamma}_k - E[\hat{\gamma}_k])^3\} \tag{5}$$
$$k_4(\hat{\gamma}_k) = E\{(\hat{\gamma}_k - E[\hat{\gamma}_k])^4\} - 3\text{Var}\{\hat{\gamma}_k\}^2. \tag{6}$$

The same relationship is valid between q_1, q_2 and p_1, p_2 with the third and fourth cumulants of $\hat{\rho}_k$.

2.1 Numerical Experiments

We present the results of a wide numerical experiment. The aim of the experiment is to compare the distribution of the normalised sample autocovariance function, C_k ($k = 0, 1, 2, 5, 10$), and sample autocorrelation function, R_k ($k = 1, 2, 5, 10$), of a strong long memory process with three different theoretical distributions: the Normal distribution and two corrected distributions, calculated using equations (3) and (4).

The Monte Carlo experiment consists of generating $S = 20000$ series of length $n = 300, 1000$ and 2000 for different values of the memory parameter $d = 0.26, 0.27, \ldots, 0.49$. We use simulated data for two purposes. Firstly, we calculate the normalised sample autocovariance and autocorrelation functions, C_k and R_k, for each series, obtaining estimates of their distributions. Secondly, we thus obtain reliable estimates of the third and fourth cumulants of C_k and R_k, based on formulas (5) and (6). The theoretical third and fourth cumulants are too burdensome to evaluate numerically, being the solutions of the multiple integral of equation (see Albin, 1998a). As we already explained in Section 2, C_k and R_k were not normalised in the simulations by their standard errors, because we are interested in comparing the quantiles of the distributions, so the scale is not important.

We preliminarily explore the data through Q-Q plots of the standard Normal distribution, of ${}_1C_k$ and of ${}_2C_k$ versus C_k. In the case of the autocovariance function, see Figures (1)-(3), the data are clearly non-Normal, and non-Normality is more pronounced for larger values of the memory parameter. On the other hand, it seems that for longer series the distribution of the sample autocovariance function gets closer to the Normal distribution. When we analyse the effects of the Cornish-Fisher corrections, we find that the first correction has a visible impact: the Normal distribution with one correction term is closer to the sample distribution. The second correction has almost no impact, suggesting that only the first correction is significant: we do not draw the second order correction in the graphs because it would superimpose the first order correction. For longer series, when the distribution of the sample autocovariance function is closer to normality, the contribution of Cornish-Fisher is smaller. Similar Q-Q plots, given in Figures (4)-(6), of the autocorrelation function seem noticeably different. The sample autocorrelation function has a distribution quite close to Normality, and a slight deviation can be observed for larger values of the parameter, i.e., $d = 0.45, 0.49$. The contributions of both the first and the second correction terms seemed to be irrelevant from a graphical point of view. It must be underlined that, for $d = 0.49$, the convergence of the autocorrelation function is so slow that the correction is not well estimated and the first order correction is behaving in a unusual way. Another indication of the slow convergence rate of the sample autocovariance and autocorrelation functions can be noticed in the ordinates of the figures: it should be centred around zero, however when the value of the memory parameter d increases, the bias becomes huge for both the sample autocovariance and the autocorrelation functions. This issue was already investigated by Newbold and Agiakloglou (1993).

An interesting insight of the sample distribution is given by Figure 7: for increasing values of d (abscissa) and different sample sizes, there is the estimate of the sample kurtosis (part a) and of the sample skewness (part b) for the autocovariance and autocorrelation functions. The two quantities are positively correlated with the memory parameter d in the case of the autocovariance function meaning that the sample distribution becomes more skewed and heavy tailed, whereas they remain approximately constant and with values closed to the Normal distribution in the case of the sample autocorrelation function.

As a second step we investigate if the improvements pointed out with the graphical analysis are statistically significant. We studied three standard linear regression models, where the dependent variable Y is the quantile of the sample autocovariance (autocorrelation) function while the covariate changes:

$$\begin{aligned} &\text{Model 1:} \quad Y = \alpha_0 + \alpha_1 X_1 + \varepsilon_1, \\ &\text{Model 2:} \quad Y = \beta_0 + \beta_1 X_2 + \varepsilon_2, \\ &\text{Model 3:} \quad Y = \tau_0 + \tau_1 X_3 + \varepsilon_3, \end{aligned} \tag{7}$$

where X_1 is the quantile of the Normal distribution, whereas X_2 and X_3 are the quantiles of the Normal distribution corrected with one and two terms, respectively, of the Cornish-Fisher expansion. As expected the R squared values are all close to unity with $R^2(Y,X_1) < R^2(Y,X_2) < R^2(Y,X_3)$. We cannot use the F test to compare the three models because they are not nested. An alternative is offered by the Cox test. This test statistic is Normally distributed if the errors ε_i, $i = 1,2,3$, are Gaussian. Even though this is not the case, we performed a small Monte Carlo experiment to check the Normality of the test when the hypotheses are violeted (the results are available upon request from the corresponding author). We refer to the Cox test with $\hat{S}_{k,1}$ and $\hat{s}_{k,1}$ when we compare the first two models, $\hat{S}_{k,2}$ and $\hat{s}_{k,2}$ for the second and the third, where k indicates the lag, capital letter for sample autocovariance function and lower case letter for sample autocorrelation.

In Tables 1-3 we report the observed results of the test for the sample autocovaraince function: for each value of the long memory parameter, in the first column there is the value of the test $\hat{S}_{k,1}$ ($\hat{s}_{k,1}$) when we compare the Normal approximation with the first order approximation (first and second model of Equations (7)), while the second column reports the values of the test $\hat{S}_{k,2}$ ($\hat{s}_{k,2}$) when we compare the first and the second order approximations (second and third model of Equations (7)). These values have to be compared with the 5% quantile of the Normal distribution, i.e., ± 1.96. For

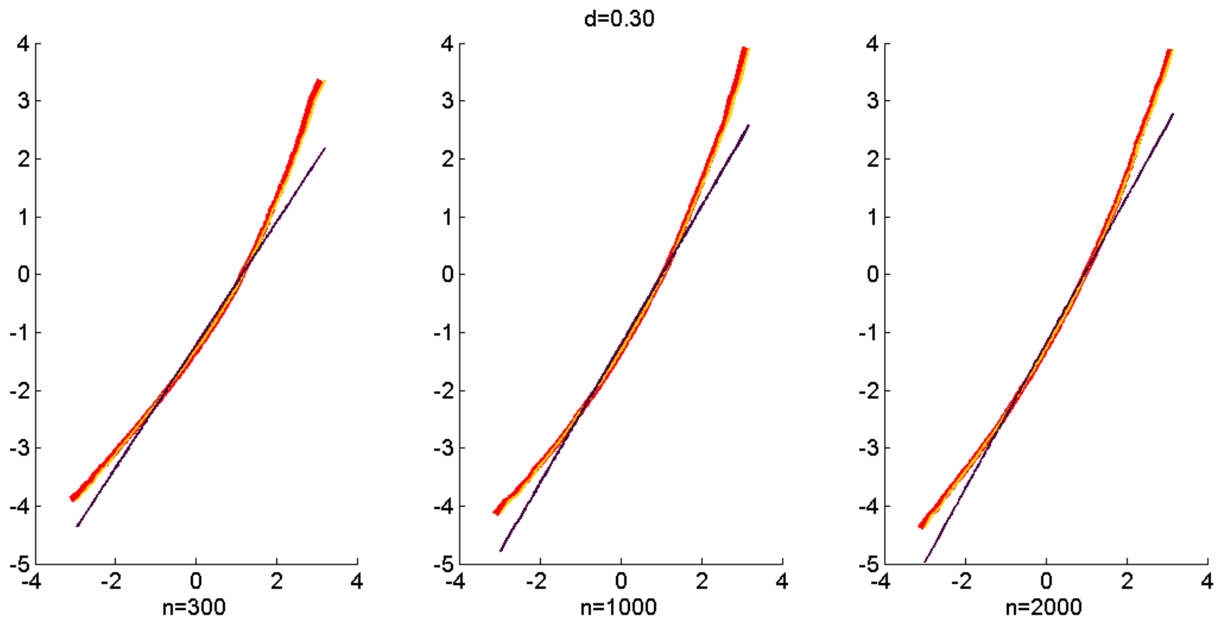


Figure 1: Q-Q plots of the sample autocovariance function for the memory parameter $d = 0.30$ and for different sample sizes, $n = 300, 1000, 2000$; the red line is the Normal approximation, the yellow line is the first order approximation of the Cornish-Fisher expansion (see equation (1)) and the black line is the benchmark.

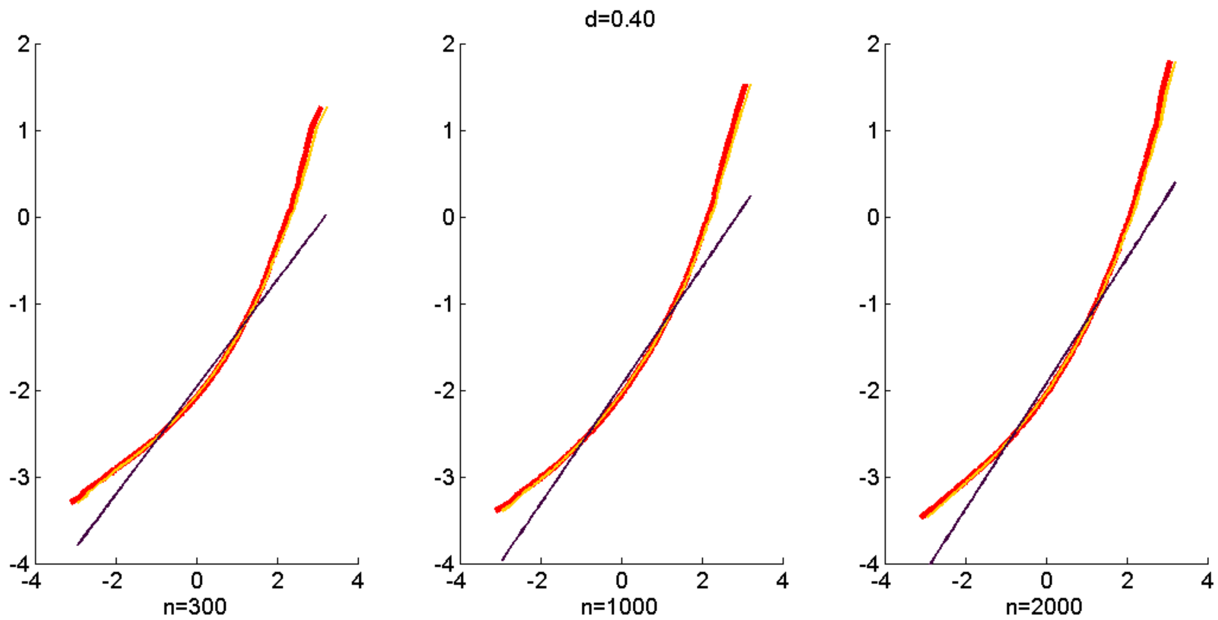


Figure 2: Q-Q plots of the sample autocovariance function for the memory parameter $d = 0.40$ and for different sample sizes, $n = 300, 1000, 2000$; the red line is the Normal approximation, the yellow line is the first order approximation of the Cornish-Fisher expansion (see equation (1)) and the black line is the benchmark.

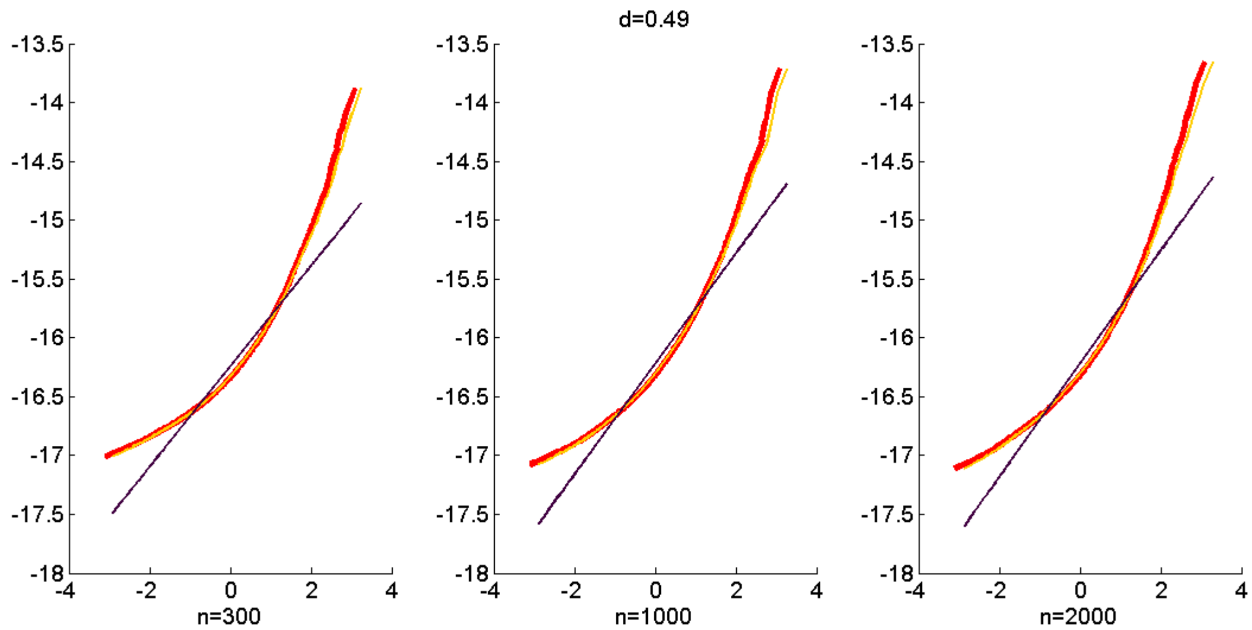


Figure 3: Q-Q plots of the sample autocovariance function for the memory parameter $d = 0.49$ and for different sample sizes, $n = 300, 1000, 2000$; the red line is the Normal approximation, the yellow line is the first order approximation of the Cornish-Fisher expansion (see equation (1)) and the black line is the benchmark.

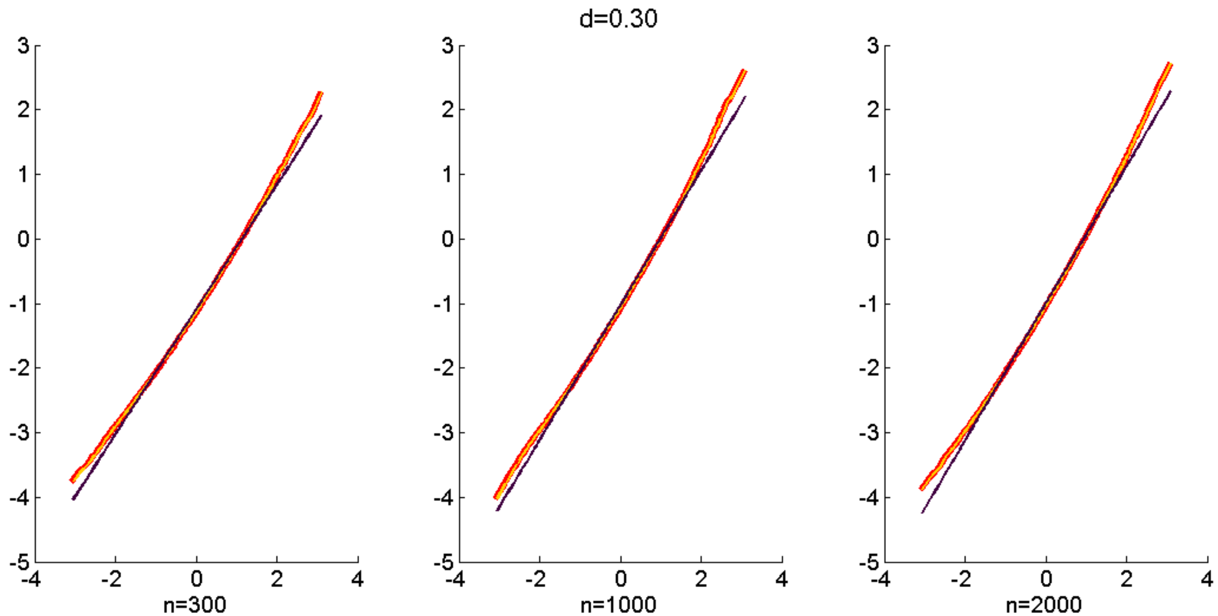


Figure 4: Q-Q plots of the sample autocorrelation function for the memory parameter $d = 0.30$ and for different sample sizes, $n = 300, 1000, 2000$; the red line is the Normal approximation, the yellow line is the first order approximation of the Cornish-Fisher expansion (see equation (2)) and the black line is the benchmark.

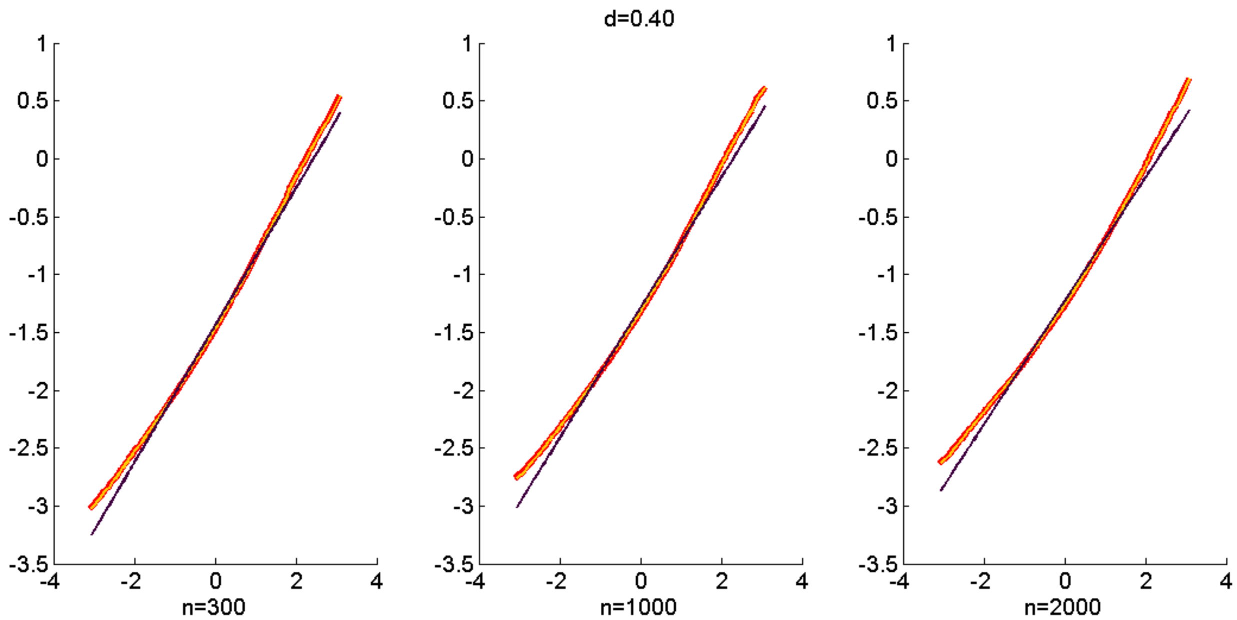


Figure 5: Q-Q plots of the sample autocorrelation function for the memory parameter $d = 0.40$ and for different sample sizes, $n = 300, 1000, 2000$; the red line is the Normal approximation, the yellow line is the first order approximation of the Cornish-Fisher expansion (see equation (2)) and the black line is the benchmark.

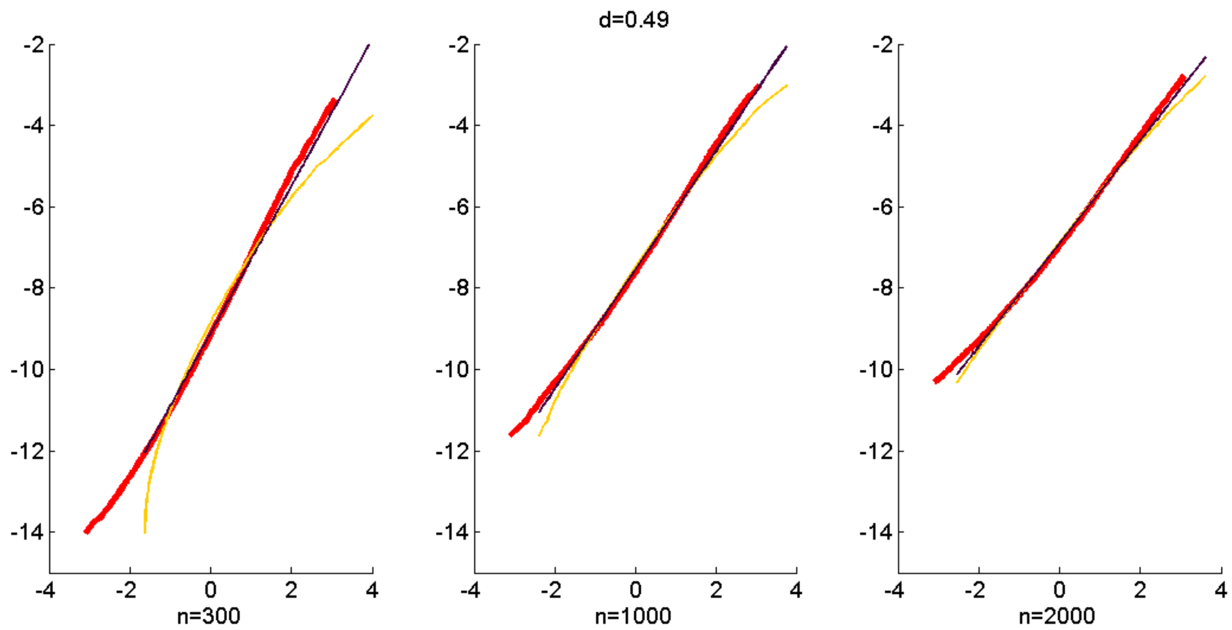
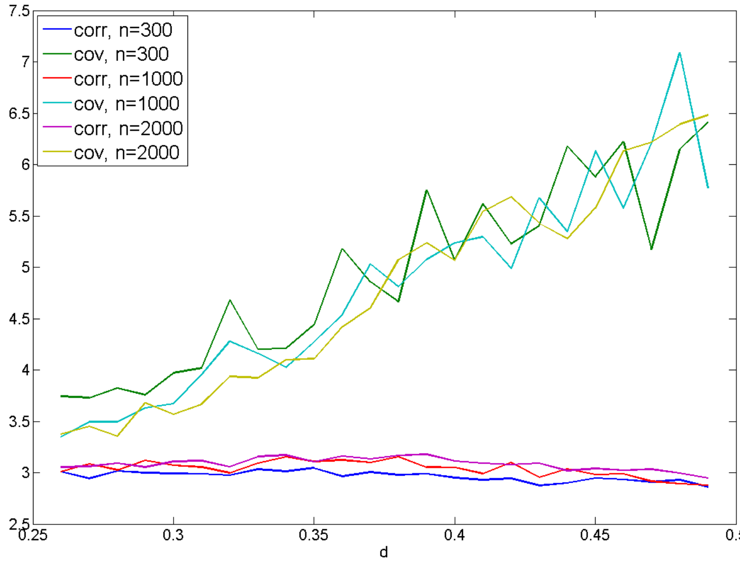
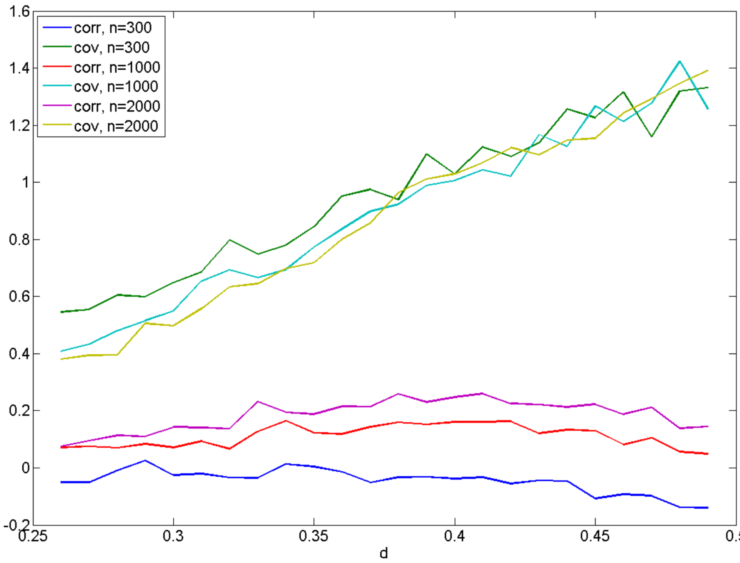


Figure 6: Q-Q plots of the sample autocorrelation function for the memory parameter $d = 0.49$ and for different sample sizes, $n = 300, 1000, 2000$; the red line is the Normal approximation, the yellow line is the first order approximation of the Cornish-Fisher expansion (see equation (2) and the black line is the benchmark.



(a) kurtosis



(b) skewness

Figure 7: Sample kurtosis $k_4(x) = E\{(x - E[x])^4\} - 3\text{Var}\{x\}^2$ and sample skewness $k_3(x) = E\{(x - E[x])^3\}$ for increasing values of $d = 0.26, 0.27, \ldots, 0.49$ of the sample autocorrelation and autocovariance function for different series length $n = 300, 1000, 2000$.

the autocovariance function the first order correction term is significant, as we already noticed in the Q-Q plots. The second order term is not significant for values of d smaller than 0.42. However, the correction of the second term becomes less significant, when increasing the sample size, suggesting that, asymptotically, the second term is negligible for any value of the memory parameter d.

In Tables 4-6 we report the observed results of the test for the sample autocorrelation function. The results for the sample autocorrelation function are controversial. The second order correction is not as significant as for the autocovariance function. From the Q-Q plot it seemed that the sample autocorrelation function is quite close to Normality; however the observed values of the Cox test indicate that the first order correction is very significant.

We conclude this section by noting that, from the Q-Q plots, the sample autocorrelation function seems closer to Normality than the sample autocovariance function; however the first order correction is significant in both cases. This is not a surprise since the sample distributions are very asymmetric and the first correction depends on moments up to the third order and corrects asymmetry, whereas the second order correction depends on the moments up to the fourth order and correct tails. As regards the convergence rate of the sample autocovariance function, it really depends on the value of d: the larger it is the slower the convergence. Increasing the sample size diminishes the importance of the correction.

3 Comparing sample and bootstrap autocorrelation functions

Now that we can compare the sample autocovariance and autocorrelation functions with their bootstrap estimates, we wish to show that the ACF bootstrap is consistent for long memory processes because it can replicate their second order structure. This statement is supported by the observed average values and standard deviations of $\hat{\gamma}_k$, $\hat{\gamma}_k^*$, $\hat{\rho}_k$ and $\hat{\rho}_k^*$ and by comparing their distributions.

We run a wide simulation experiment and compare the sample autocorrelation function with the bootstrap autocorrelation function. For sake of completeness we consider the range of values of $d = 0.1, 0.15, 0.2, \ldots, 0.45$ and increasing series lengths $n = 100, 200, \ldots, 2000$. We repeat the same experiment with innovations distributed as Chi-squared with one degree of freedom, and Student t with four and six degrees of freedom, to support our belief that ACF bootstrap can replicate the second order structure of a process X_t even when the innovations are non-Gaussian.

In Figures 8 and 9 we show the empirical densities of the sample and bootstrap variance and autocovariance function. From the densities of the variance, it is interesting to notice that the distribution is quite skewed and for large values of d it is also quite biased. Increasing the sample size from 1000 to 2000 the density moves to the right significantly. Also for the autocorrelation at lag one, there is bias for $d = 0.4, 0.45$. In both cases the bootstrap densities seem to follow the Monte Carlo pattern, but there are fewer extreme values and the distributions are a bit more concentrated.

The results for all sample sizes considered are given in Tables 10 and 11. We report in the first column the difference between the Monte Carlo sample autocorrelation function and the bootstrap estimate, $\hat{\rho}_k - \hat{\rho}^*_k$. The bootstrap variance is biased downward. The bias is positively correlated with the strength of the long memory and negatively correlated with the sample size. However, for sample sizes larger than 1000 the bias is always less than 2%. In terms of standard deviation, using the bootstrap technique there is on average an improvement and its order is positively correlated with the value of the memory parameter.

4 Conclusions

For a strong long memory Gaussian process, i.e., $0.25 < d < 0.5$, we have found that the sample autocovariance function is not Normal, although we have a significant improvement correcting the Normal distribution with the first term of the Cornish-Fisher expansion. Deviation from Normality gets smaller when increasing the sample size, whereas it is more evident for stronger long memory processes; also the second order correction term has a more important contribution.

The sample autocorrelation function is closer to Normality and small deviation from Normality can be detected for very large values of the memory parameter, i.e., $d > 0.4$. In this case the Cornish-Fisher corrections seem not to give any contribution from a graphical point of view (Figures (4)-(6), even though a significant contribution is detected by the Cox test.

By means of simulation, we showed that the ACF bootstrap can replicate the second order structure of a long memory Gaussian process. The method gave satisfactory results also with non-Gaussian processes, indicating that at least the second order structure is preserved, and asymmetry or extreme values in the innovation distribution do not affect the performance dramatically.

This work is open to future developments:

- it is a good starting point to try to prove theoretically the consistency of ACF bootstrap, and
- given the good results on non-Gaussian time series, it is probably possible to prove that the ACF bootstrap replicates second order structure of linear processes no matter what innovations drive the data generating process.

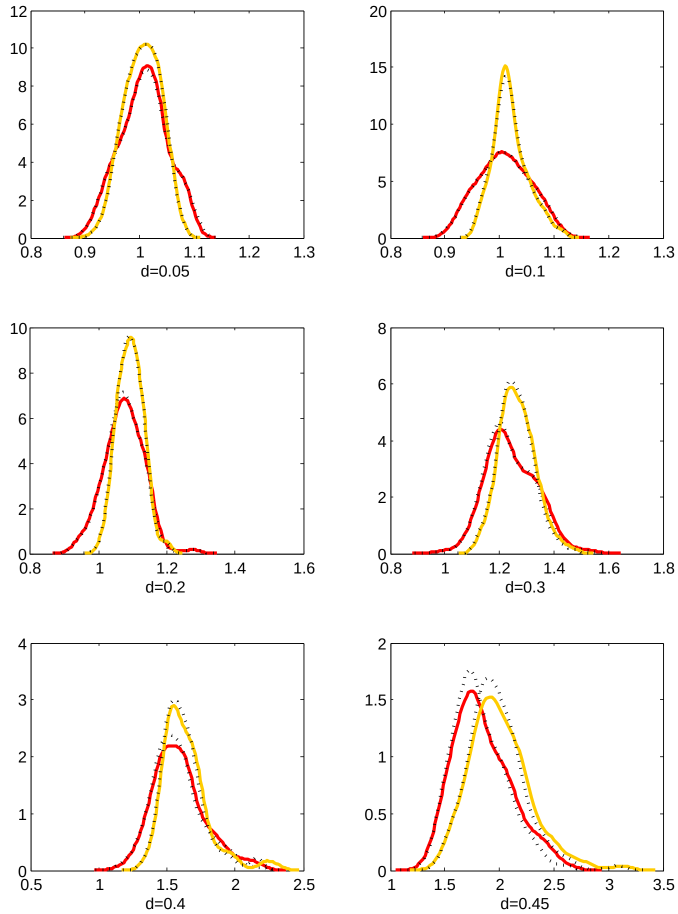


Figure 8: Plots of the density of the sample and the bootstrap variance for different values of $d = 0.05, 0.1, 0.2, 0.3, 0.4, 0.45$ and different series length. The red line is for the sample autocorrelation function with $n = 1000$, the yellow line is for $n = 2000$. The dotted lines are the bootstrap autocorrelation densities.

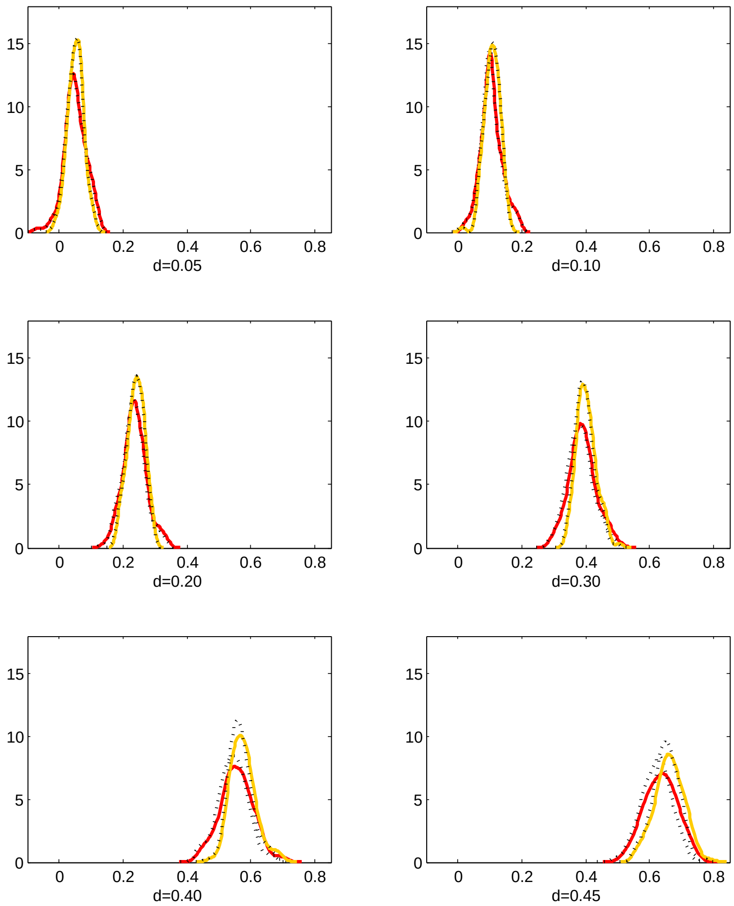


Figure 9: Plots of the density of the sample autocorrelation function and the bootstrap autocorrelation function at lag $k = 1$ for different values of $d = 0.05, 0.1, 0.2, 0.3, 0.4, 0.45$ and different series length. The red line is for the sample autocorrelation function with $n = 1000$, the yellow line is for $n = 2000$. The dotted lines are the bootstrap autocorrelation densities.

lag	$d = 0.26$		$d = 0.27$		$d = 0.28$		$d = 0.29$	
0	-35.04	*0.40*	-32.02	*0.27*	-30.56	*0.20*	-30.13	*0.13*
1	-29.96	*0.20*	-29.64	*0.14*	-29.56	*0.07*	-29.31	*0.03*
2	-29.44	*0.17*	-29.24	*0.10*	-29.33	*-0.01*	-29.34	*0.04*
5	-28.34	*0.03*	-28.60	*-0.08*	-28.71	*-0.17*	-28.93	*-0.05*
10	-27.24	*-0.19*	-28.06	*-0.21*	-27.81	*-0.38*	-28.07	*-0.27*
lag	$d = 0.30$		$d = 0.31$		$d = 0.32$		$d = 0.33$	
0	-29.82	*-0.03*	-29.22	*-0.14*	-29.28	*-0.38*	-29.17	*-0.29*
1	-29.47	*-0.07*	-29.17	*-0.12*	-28.78	*-0.47*	-29.33	*-0.24*
2	-29.27	*-0.12*	-29.01	*-0.21*	-28.72	*-0.54*	-29.26	*-0.31*
5	-28.79	*-0.24*	-28.95	*-0.27*	-28.69	*-0.70*	-29.06	*-0.43*
10	-28.26	*-0.35*	-28.06	*-0.54*	-28.28	*-0.79*	-28.64	*-0.56*
lag	$d = 0.34$		$d = 0.35$		$d = 0.36$		$d = 0.37$	
0	-29.30	*-0.21*	-29.08	*-0.40*	-28.58	*-0.89*	-29.01	*-0.63*
1	-29.32	*-0.25*	-29.16	*-0.39*	-28.57	*-0.93*	-29.14	*-0.63*
2	-29.29	*-0.32*	-29.21	*-0.41*	-28.64	*-0.94*	-29.14	*-0.68*
5	-29.18	*-0.37*	-28.97	*-0.57*	-28.68	*-0.99*	-28.95	*-0.86*
10	-29.03	*-0.50*	-28.64	*-0.72*	-28.50	*-1.15*	-28.81	*-1.01*
lag	$d = 0.38$		$d = 0.39$		$d = 0.40$		$d = 0.41$	
0	-29.20	*-0.64*	-28.61	*-1.36*	-28.96	*-1.09*	-28.91	*-1.58*
1	-29.33	*-0.61*	-28.60	*-1.44*	-29.09	*-1.04*	-29.13	*-1.38*
2	-29.33	*-0.67*	-28.78	*-1.40*	-29.22	*-0.98*	-29.25	*-1.33*
5	-29.28	*-0.79*	-28.47	*-1.64*	-29.15	*-1.14*	-29.20	*-1.48*
10	-29.04	*-1.03*	-28.39	*-1.70*	-28.95	*-1.35*	-29.05	*-1.78*
lag	$d = 0.42$		$d = 0.43$		$d = 0.44$		$d = 0.45$	
0	-29.03	*-1.30*	-28.98	*-1.54*	-28.88	*-2.19*	-29.09	*-2.11*
1	-29.19	*-1.25*	-29.13	*-1.48*	-29.02	*-2.12*	-29.27	*-1.91*
2	-29.29	*-1.23*	-29.19	*-1.46*	-29.06	*-2.08*	-29.30	*-1.94*
5	-29.32	*-1.32*	-29.19	*-1.56*	-29.12	*-2.11*	-29.42	*-1.90*
10	-29.17	*-1.56*	-29.31	*-1.57*	-29.11	*-2.14*	-29.44	*-1.98*
lag	$d = 0.46$		$d = 0.47$		$d = 0.48$		$d = 0.49$	
0	-28.91	*-2.62*	-29.50	*-1.57*	-29.17	*-2.74*	-29.22	*-3.11*
1	-29.07	*-2.45*	-29.58	*-1.52*	-29.25	*-2.62*	-29.33	*-2.93*
2	-29.14	*-2.36*	-29.69	*-1.45*	-29.34	*-2.52*	-29.34	*-2.93*
5	-29.25	*-2.33*	-29.73	*-1.53*	-29.43	*-2.57*	-29.46	*-2.82*
10	-29.12	*-2.65*	-29.69	*-1.74*	-29.43	*-2.72*	-29.57	*-2.81*

Table 1: Cox test to compare the Monte Carlo distribution of the sample autocovariance function of a long memory process for different values of the memory parameter d at different lags, $k = 0, 1, 2, 5, 10$, for $n = 300$.

lag	$d = 0.26$		$d = 0.27$		$d = 0.28$		$d = 0.29$	
0	-29.96	*0.18*	-29.45	*0.09*	-29.44	*0.10*	-29.14	*0.05*
1	-29.50	*0.12*	-29.32	*0.07*	-29.33	*0.08*	-29.26	*0.04*
2	-29.38	*0.12*	-29.37	*0.07*	-29.20	*0.06*	-29.33	*0.02*
5	-29.30	*0.10*	-28.58	*0.02*	-29.36	*0.05*	-28.86	*0.00*
10	-29.06	*0.09*	-28.26	*-0.05*	-28.99	*0.04*	-28.73	*-0.03*
lag	$d = 0.30$		$d = 0.31$		$d = 0.32$		$d = 0.33$	
0	-29.10	*0.02*	-28.62	*-0.06*	-28.74	*-0.19*	-29.12	*-0.21*
1	-29.22	*0.01*	-28.96	*-0.04*	-29.01	*-0.18*	-29.04	*-0.24*
2	-29.40	*0.01*	-28.66	*-0.09*	-29.12	*-0.16*	-29.01	*-0.24*
5	-29.40	*-0.02*	-28.87	*-0.07*	-28.92	*-0.18*	-29.28	*-0.21*
10	-29.23	*-0.05*	-28.77	*-0.10*	-29.23	*-0.19*	-29.27	*-0.21*
lag	$d = 0.34$		$d = 0.35$		$d = 0.36$		$d = 0.37$	
0	-29.29	*-0.15*	-28.92	*-0.28*	-29.04	*-0.41*	-29.12	*-0.65*
1	-29.40	*-0.16*	-29.19	*-0.26*	-29.14	*-0.37*	-29.10	*-0.67*
2	-29.53	*-0.12*	-29.30	*-0.24*	-29.23	*-0.36*	-29.22	*-0.66*
5	-29.46	*-0.15*	-29.17	*-0.30*	-29.53	*-0.31*	-29.44	*-0.57*
10	-29.43	*-0.21*	-29.24	*-0.31*	-29.12	*-0.48*	-29.35	*-0.60*
lag	$d = 0.38$		$d = 0.39$		$d = 0.40$		$d = 0.41$	
0	-29.29	*-0.52*	-28.86	*-0.80*	-28.51	*-1.12*	-29.09	*-1.05*
1	-29.36	*-0.52*	-28.89	*-0.82*	-28.73	*-1.05*	-29.21	*-1.01*
2	-29.37	*-0.53*	-28.99	*-0.80*	-28.81	*-1.02*	-29.29	*-0.96*
5	-29.36	*-0.59*	-29.18	*-0.73*	-28.98	*-0.98*	-29.27	*-1.04*
10	-29.47	*-0.58*	-29.31	*-0.69*	-29.01	*-0.99*	-29.33	*-1.04*
lag	$d = 0.42$		$d = 0.43$		$d = 0.44$		$d = 0.45$	
0	-29.23	*-0.98*	-28.97	*-1.52*	-29.34	*-1.28*	-29.14	*-1.94*
1	-29.20	*-1.01*	-29.06	*-1.46*	-29.38	*-1.29*	-29.22	*-1.89*
2	-29.25	*-1.01*	-29.09	*-1.46*	-29.41	*-1.27*	-29.19	*-1.96*
5	-29.36	*-1.02*	-29.25	*-1.40*	-29.52	*-1.20*	-29.32	*-1.85*
10	-29.48	*-0.98*	-29.39	*-1.34*	-29.66	*-1.13*	-29.39	*-1.85*
lag	$d = 0.46$		$d = 0.47$		$d = 0.48$		$d = 0.49$	
0	-29.16	*-1.85*	-29.07	*-2.57*	-28.86	*-3.52*	-29.45	*-2.30*
1	-29.19	*-1.83*	-29.12	*-2.53*	-28.87	*-3.50*	-29.47	*-2.30*
2	-29.21	*-1.84*	-29.10	*-2.56*	-28.92	*-3.43*	-29.48	*-2.28*
5	-29.26	*-1.83*	-29.09	*-2.62*	-29.03	*-3.28*	-29.51	*-2.27*
10	-29.30	*-1.85*	-29.22	*-2.52*	-29.08	*-3.19*	-29.57	*-2.27*

Table 2: Cox test to compare the Monte Carlo distribution of the sample autocovariance function of a long memory process for different values of the memory parameter d at different lags, $k = 0, 1, 2, 5, 10$, for $n = 1000$.

lag	$d = 0.26$		$d = 0.27$		$d = 0.28$		$d = 0.29$	
0	-29.13	*0.12*	-28.88	*0.07*	-28.65	*0.07*	-28.67	*0.04*
1	-29.07	*0.10*	-28.93	*0.06*	-29.29	*0.06*	-28.81	*0.02*
2	-29.20	*0.10*	-28.61	*0.07*	-29.24	*0.05*	-28.87	*0.02*
5	-29.06	*0.09*	-28.70	*0.04*	-28.91	*0.06*	-28.96	*0.01*
10	-28.46	*0.07*	-29.46	*0.04*	-29.59	*0.04*	-28.95	*-0.01*
lag	$d = 0.30$		$d = 0.31$		$d = 0.32$		$d = 0.33$	
0	-28.76	*0.03*	-28.96	*0.00*	-29.09	*-0.07*	-28.75	*-0.12*
1	-29.01	*0.02*	-29.20	*0.00*	-29.15	*-0.07*	-29.09	*-0.10*
2	-29.19	*0.02*	-29.07	*-0.02*	-29.13	*-0.07*	-29.11	*-0.10*
5	-29.02	*0.01*	-29.05	*-0.03*	-29.23	*-0.08*	-28.93	*-0.13*
10	-29.44	*0.00*	-28.99	*-0.05*	-29.47	*-0.07*	-29.14	*-0.12*
lag	$d = 0.34$		$d = 0.35$		$d = 0.36$		$d = 0.37$	
0	-28.90	*-0.20*	-29.05	*-0.20*	-28.88	*-0.34*	-29.18	*-0.38*
1	-29.29	*-0.15*	-29.15	*-0.20*	-29.10	*-0.31*	-29.13	*-0.40*
2	-29.20	*-0.17*	-29.15	*-0.20*	-29.20	*-0.31*	-29.25	*-0.40*
5	-29.23	*-0.19*	-29.08	*-0.24*	-29.13	*-0.35*	-29.31	*-0.38*
10	-29.15	*-0.21*	-29.16	*-0.24*	-29.30	*-0.31*	-29.35	*-0.41*
lag	$d = 0.38$		$d = 0.39$		$d = 0.40$		$d = 0.41$	
0	-28.77	*-0.65*	-28.71	*-0.78*	-29.00	*-0.75*	-29.00	*-1.06*
1	-28.76	*-0.66*	-28.74	*-0.78*	-29.04	*-0.73*	-29.01	*-1.07*
2	-28.96	*-0.61*	-28.83	*-0.75*	-29.08	*-0.74*	-29.01	*-1.06*
5	-29.08	*-0.57*	-28.83	*-0.77*	-29.18	*-0.72*	-29.04	*-1.09*
10	-29.14	*-0.58*	-28.81	*-0.82*	-29.24	*-0.72*	-29.08	*-1.08*
lag	$d = 0.42$		$d = 0.43$		$d = 0.44$		$d = 0.45$	
0	-28.95	*-1.26*	-29.22	*-1.23*	-29.18	*-1.22*	-29.14	*-1.61*
1	-29.01	*-1.23*	-29.25	*-1.20*	-29.22	*-1.21*	-29.19	*-1.59*
2	-29.05	*-1.21*	-29.32	*-1.16*	-29.24	*-1.21*	-29.18	*-1.60*
5	-29.12	*-1.18*	-29.38	*-1.12*	-29.27	*-1.21*	-29.15	*-1.66*
10	-29.15	*-1.18*	-29.50	*-1.07*	-29.38	*-1.15*	-29.24	*-1.60*
lag	$d = 0.46$		$d = 0.47$		$d = 0.48$		$d = 0.49$	
0	-29.31	*-2.00*	-29.05	*-2.42*	-29.08	*-2.82*	-28.98	*-3.25*
1	-29.36	*-1.94*	-29.08	*-2.40*	-29.12	*-2.77*	-29.02	*-3.19*
2	-29.39	*-1.93*	-29.13	*-2.35*	-29.12	*-2.78*	-29.04	*-3.18*
5	-29.47	*-1.81*	-29.13	*-2.36*	-29.13	*-2.77*	-29.05	*-3.17*
10	-29.55	*-1.76*	-29.24	*-2.27*	-29.25	*-2.64*	-29.09	*-3.16*

Table 3: Cox test to compare the Monte Carlo distribution of the sample autocovariance function of a long memory process for different values of the memory parameter d at different lags, $k = 0, 1, 2, 5, 10$, for $n = 2000$.

lag	$d = 0.26$		$d = 0.27$		$d = 0.28$		$d = 0.29$	
1	-21.09	*0.00*	-24.99	*-0.57*	8.99	*-0.66*	-17.71	*0.01*
2	-23.06	*0.00*	-23.70	*0.00*	-28.28	*-0.13*	-24.23	*-0.14*
5	-28.76	*0.01*	-27.65	*-0.02*	-29.28	*0.00*	-29.00	*0.00*
10	-28.12	*-0.04*	-28.32	*-0.08*	-29.14	*-0.03*	-29.32	*-0.03*
lag	$d = 0.30$		$d = 0.31$		$d = 0.32$		$d = 0.33$	
1	-16.44	*0.06*	-13.31	*-0.06*	-18.64	*-0.20*	-12.49	*-0.68*
2	-25.35	*0.02*	-26.97	*-0.05*	-22.93	*-0.03*	-26.27	*-0.14*
5	-29.96	*-0.01*	-29.70	*-0.01*	-29.49	*-0.01*	-29.49	*0.00*
10	-29.80	*-0.01*	-28.98	*-0.05*	-29.71	*0.00*	-29.84	*-0.02*
lag	$d = 0.34$		$d = 0.35$		$d = 0.36$		$d = 0.37$	
1	-4.07	*-0.20*	17.92	*-2.70*	-18.16	*-0.97*	-24.68	*0.01*
2	-25.08	*-0.12*	-27.46	*-0.04*	-19.18	*-0.40*	-25.61	*-1.34*
5	-29.48	*-0.01*	-29.89	*0.00*	-29.99	*0.00*	-29.00	*0.00*
10	-29.89	*-0.01*	-29.62	*-0.11*	-29.64	*-0.02*	-30.24	*0.00*
lag	$d = 0.38$		$d = 0.39$		$d = 0.40$		$d = 0.41$	
1	-20.85	*-0.07*	-14.84	*0.16*	-23.18	*-0.93*	-22.79	*-3.36*
2	-26.02	*-0.10*	-25.04	*-0.53*	-22.33	*-0.99*	-25.50	*-2.30*
5	-29.84	*-0.06*	-29.98	*-0.23*	-29.63	*-0.22*	-29.62	*-0.52*
10	-30.20	*-0.01*	-30.26	*-0.09*	-30.27	*-0.03*	-30.30	*-0.05*
lag	$d = 0.42$		$d = 0.43$		$d = 0.44$		$d = 0.45$	
1	-26.14	*-1.22*	-23.66	*-6.61*	-25.28	*-5.75*	-27.62	*-1.16*
2	-21.96	*-4.64*	-20.27	*-9.41*	-25.41	*-12.63*	-22.62	*-8.78*
5	-29.25	*-1.02*	-27.15	*-2.16*	-26.81	*-1.72*	-25.25	*-5.09*
10	-29.93	*-0.15*	-29.29	*-0.47*	-29.45	*-0.45*	-28.68	*-0.90*
lag	$d = 0.46$		$d = 0.47$		$d = 0.48$		$d = 0.49$	
1	-26.65	*-2.29*	-30.40	*-9.32*	-39.23	*-22.16*	129.91	*-27.57*
2	-17.17	*-9.83*	-15.77	*-19.81*	-19.60	*-55.67*	103.68	*-12.60*
5	-23.65	*-6.64*	-20.49	*-15.24*	-12.41	*-214.04*	71.80	*-8.23*
10	-28.15	*-1.51*	-27.62	*-4.45*	-49.55	*-98.48*	87.08	*-21.60*

Table 4: Cox test to compare the Monte Carlo distribution of the sample autocorrelation function of a long memory process for different values of the memory parameter d at different lags, $k = 1, 2, 5, 10$, for $n = 300$.

lag	$d = 0.26$		$d = 0.27$		$d = 0.28$		$d = 0.29$	
1	-25.85	*0.02*	-21.33	*-0.29*	-24.42	*-0.02*	-20.74	*-0.55*
2	-26.41	*-0.02*	-24.59	*-0.15*	-28.40	*0.01*	-26.22	*-0.05*
5	-29.08	*0.01*	-28.42	*-0.04*	-29.50	*0.01*	-29.14	*-0.03*
10	-29.38	*0.01*	-28.15	*-0.05*	-29.40	*-0.02*	-28.82	*-0.07*
lag	$d = 0.30$		$d = 0.31$		$d = 0.32$		$d = 0.33$	
1	-19.86	*-0.36*	-24.51	*-0.11*	-26.24	*-0.01*	-26.44	*-0.12*
2	-24.18	*-0.26*	-27.46	*-0.02*	-27.84	*-0.07*	-28.62	*-0.02*
5	-28.21	*-0.05*	-29.00	*-0.01*	-29.03	*-0.03*	-28.88	*-0.07*
10	-28.99	*-0.04*	-29.27	*-0.04*	-29.85	*-0.01*	-29.26	*-0.05*
lag	$d = 0.34$		$d = 0.35$		$d = 0.36$		$d = 0.37$	
1	-25.70	*-0.27*	-24.23	*-0.27*	-22.50	*-0.36*	-25.35	*-0.23*
2	-28.27	*-0.13*	-26.05	*-0.15*	-23.82	*-0.26*	-27.14	*-0.12*
5	-29.36	*-0.07*	-28.84	*-0.05*	-28.91	*-0.02*	-28.78	*-0.07*
10	-29.15	*-0.12*	-30.13	*-0.01*	-29.83	*-0.01*	-29.77	*-0.03*
lag	$d = 0.38$		$d = 0.39$		$d = 0.40$		$d = 0.41$	
1	-24.65	*-0.45*	-27.11	*-0.06*	-27.90	*-0.06*	-28.83	*-0.03*
2	-27.51	*-0.14*	-29.19	*-0.03*	-29.07	*-0.01*	-29.70	*-0.02*
5	-29.15	*-0.06*	-30.11	*0.00*	-29.62	*-0.01*	-30.18	*0.00*
10	-29.58	*-0.06*	-29.79	*-0.04*	-30.25	*0.00*	-30.41	*0.01*
lag	$d = 0.42$		$d = 0.43$		$d = 0.44$		$d = 0.45$	
1	-26.78	*-0.26*	-29.56	*-0.17*	-27.11	*-0.10*	-28.86	*-0.04*
2	-28.43	*-0.10*	-29.25	*-0.06*	-29.34	*0.00*	-29.87	*-0.18*
5	-29.63	*-0.04*	-30.10	*-0.03*	-30.21	*0.00*	-29.83	*-0.10*
10	-30.04	*-0.03*	-30.34	*-0.02*	-30.20	*0.00*	-30.24	*-0.14*
lag	$d = 0.46$		$d = 0.47$		$d = 0.48$		$d = 0.49$	
1	-26.04	*-0.01*	-27.18	*-1.71*	-22.52	*-12.70*	28.63	*-10.73*
2	-27.97	*-0.51*	-27.27	*-2.58*	-27.51	*-13.88*	35.08	*-9.36*
5	-29.40	*-0.64*	-28.27	*-1.59*	-32.02	*-6.48*	33.53	*-8.76*
10	-29.67	*-0.42*	-28.94	*-1.01*	-32.25	*-6.77*	27.72	*-7.08*

Table 5: Cox test to compare the Monte Carlo distribution of the sample autocorrelation function of a long memory process for different values of the memory parameter d at different lags, $k = 1, 2, 5, 10$, for $n = 1000$.

lag	$d = 0.26$		$d = 0.27$		$d = 0.28$		$d = 0.29$	
1	-23.13	*-0.06*	-27.27	*-0.02*	-26.47	*-0.07*	-26.53	*-0.02*
2	-27.52	*-0.01*	-25.52	*-0.09*	-26.29	*-0.09*	-26.98	*-0.08*
5	-29.01	*0.00*	-26.22	*-0.08*	-27.71	*-0.03*	-28.25	*-0.04*
10	-29.12	*0.01*	-29.00	*0.00*	-26.90	*-0.12*	-28.74	*-0.02*
lag	$d = 0.30$		$d = 0.31$		$d = 0.32$		$d = 0.33$	
1	-25.87	*-0.10*	-25.08	*-0.12*	-27.12	*-0.03*	-28.50	*-0.07*
2	-28.32	*-0.01*	-27.21	*-0.07*	-28.43	*-0.01*	-29.22	*-0.02*
5	-29.83	*0.00*	-28.14	*-0.04*	-29.08	*-0.01*	-28.88	*-0.07*
10	-29.73	*-0.01*	-28.93	*-0.03*	-29.68	*0.00*	-28.69	*-0.11*
lag	$d = 0.34$		$d = 0.35$		$d = 0.36$		$d = 0.37$	
1	-26.53	*-0.15*	-26.94	*-0.07*	-27.02	*-0.14*	-27.93	*-0.06*
2	-27.91	*-0.12*	-28.70	*-0.03*	-27.78	*-0.10*	-29.01	*-0.03*
5	-29.43	*-0.05*	-28.90	*-0.04*	-28.67	*-0.06*	-28.62	*-0.09*
10	-29.69	*-0.04*	-29.40	*-0.03*	-29.11	*-0.05*	-29.45	*-0.07*
lag	$d = 0.38$		$d = 0.39$		$d = 0.40$		$d = 0.41$	
1	-28.15	*-0.11*	-26.37	*-0.20*	-28.44	*-0.05*	-29.43	*-0.02*
2	-28.50	*-0.10*	-28.08	*-0.09*	-28.42	*-0.10*	-29.50	*-0.02*
5	-29.11	*-0.10*	-28.80	*-0.05*	-29.33	*-0.04*	-29.58	*-0.01*
10	-29.29	*-0.08*	-29.22	*-0.04*	-29.73	*-0.03*	-29.97	*0.00*
lag	$d = 0.42$		$d = 0.43$		$d = 0.44$		$d = 0.45$	
1	-27.94	*-0.04*	-28.27	*-0.05*	-29.59	*0.00*	-30.03	*0.01*
2	-28.77	*-0.04*	-29.05	*-0.02*	-30.06	*0.00*	-30.05	*0.02*
5	-28.85	*-0.07*	-29.86	*0.01*	-30.03	*0.00*	-30.36	*0.02*
10	-29.72	*-0.01*	-29.75	*-0.05*	-30.41	*0.01*	-30.43	*0.03*
lag	$d = 0.46$		$d = 0.47$		$d = 0.48$		$d = 0.49$	
1	-29.58	*0.00*	-29.06	*-0.01*	-29.44	*-0.03*	-20.20	*-5.23*
2	-29.99	*0.00*	-29.19	*0.00*	-30.10	*-0.30*	-25.85	*-7.65*
5	-29.89	*0.00*	-29.50	*-0.05*	-30.10	*-0.29*	-28.31	*-9.86*
10	-30.29	*-0.03*	-29.76	*-0.04*	-30.22	*-0.49*	-35.97	*-15.91*

Table 6: Cox test to compare the Monte Carlo distribution of the sample autocorrelation function of a long memory process for different values of the memory parameter d at different lags, $k = 1, 2, 5, 10$, for $n = 1000$.

	$d = 0.05$				$d = 0.10$				$d = 0.15$			
n	$\hat{\gamma}_0$	$\hat{\gamma}_0^*$	$\hat{se}(\hat{\gamma}_0)$	$\hat{se}(\hat{\gamma}_0^*)$	$\hat{\gamma}_0$	$\hat{\gamma}_0^*$	$\hat{se}(\hat{\gamma}_0)$	$\hat{se}(\hat{\gamma}_0^*)$	$\hat{\gamma}_0$	$\hat{\gamma}_0^*$	$\hat{se}(\hat{\gamma}_0)$	$\hat{se}(\hat{\gamma}_0^*)$
100	**1.0204**	**1.0225**	0.1151	0.1150	**1.0046**	**1.0078**	0.1522	0.1523	**1.0303**	**1.0324**	0.1414	0.1409
200	**0.9989**	**1.0006**	0.0768	0.0771	**1.0250**	**1.0273**	0.1006	0.1009	**1.0207**	**1.0224**	0.0960	0.0963
300	**1.0116**	**1.0126**	0.0670	0.0674	**1.0104**	**1.0115**	0.0892	0.0890	**1.0196**	**1.0203**	0.0882	0.0883
400	**1.0079**	**1.0090**	0.0669	0.0681	**1.0064**	**1.0074**	0.0739	0.0740	**1.0411**	**1.0410**	0.0695	0.0692
500	**0.9936**	**0.9942**	0.0501	0.0504	**1.0205**	**1.0209**	0.0681	0.0671	**1.0500**	**1.0498**	0.0804	0.0798
600	**0.9944**	**0.9957**	0.0508	0.0517	**1.0063**	**1.0073**	0.0642	0.0637	**1.0287**	**1.0278**	0.0672	0.0673
700	**1.0095**	**1.0102**	0.0471	0.0474	**1.0143**	**1.0153**	0.0596	0.0594	**1.0307**	**1.0307**	0.0559	0.0561
800	**1.0024**	**1.0030**	0.0505	0.0508	**1.0115**	**1.0113**	0.0546	0.0542	**1.0459**	**1.0453**	0.0537	0.0538
900	**1.0078**	**1.0084**	0.0425	0.0432	**1.0141**	**1.0142**	0.0491	0.0491	**1.0366**	**1.0361**	0.0579	0.0576
1000	**1.0028**	**1.0032**	0.0366	0.0364	**1.0092**	**1.0096**	0.0460	0.0463	**1.0447**	**1.0442**	0.0514	0.0516
1100	**1.0032**	**1.0035**	0.0390	0.0390	**1.0294**	**1.0295**	0.0460	0.0457	**1.0469**	**1.0465**	0.0495	0.0489
1200	**1.0048**	**1.0053**	0.0366	0.0366	**1.0248**	**1.0248**	0.0397	0.0396	**1.0484**	**1.0478**	0.0436	0.0437
1300	**1.0053**	**1.0053**	0.0326	0.0327	**1.0106**	**1.0112**	0.0418	0.0423	**1.0456**	**1.0448**	0.0428	0.0424
1400	**1.0099**	**1.0101**	0.0286	0.0288	**1.0135**	**1.0134**	0.0392	0.0391	**1.0408**	**1.0404**	0.0474	0.0472
1500	**1.0012**	**1.0014**	0.0337	0.0335	**1.0150**	**1.0144**	0.0386	0.0385	**1.0438**	**1.0438**	0.0429	0.0425
1600	**0.9990**	**0.9992**	0.0381	0.0385	**1.0223**	**1.0221**	0.0392	0.0392	**1.0383**	**1.0382**	0.0356	0.0358
1700	**1.0041**	**1.0043**	0.0333	0.0335	**1.0197**	**1.0193**	0.0359	0.0361	**1.0388**	**1.0377**	0.0370	0.0372
1800	**1.0041**	**1.0047**	0.0300	0.0300	**1.0141**	**1.0141**	0.0352	0.0355	**1.0459**	**1.0457**	0.0398	0.0391
1900	**1.0053**	**1.0053**	0.0314	0.0312	**1.0204**	**1.0200**	0.0367	0.0366	**1.0463**	**1.0457**	0.0348	0.0347
2000	**1.0017**	**1.0020**	0.0348	0.0345	**1.0197**	**1.0198**	0.0319	0.0318	**1.0414**	**1.0413**	0.0374	0.0375

Table 7: Comparison between sample and bootstrap variances, of fractionally Gaussian integrated noise for $d = 0.05, 0.10, 0.15$ and increasing series length.

	$d = 0.20$				$d = 0.25$				$d = 0.30$			
n	$\hat{\gamma}_0$	$\hat{\gamma}_0^*$	$\hat{se}(\hat{\gamma}_0)$	$\hat{se}(\hat{\gamma}_0^*)$	$\hat{\gamma}_0$	$\hat{\gamma}_0^*$	$\hat{se}(\hat{\gamma}_0)$	$\hat{se}(\hat{\gamma}_0^*)$	$\hat{\gamma}_0$	$\hat{\gamma}_0^*$	$\hat{se}(\hat{\gamma}_0)$	$\hat{se}(\hat{\gamma}_0^*)$
100	**1.0413**	**1.0424**	0.1624	0.1619	**1.0766**	**1.0757**	0.1589	0.1541	**1.1182**	**1.1134**	0.2066	0.1986
200	**1.0439**	**1.0431**	0.1041	0.1025	**1.1052**	**1.1011**	0.1389	0.1362	**1.1801**	**1.1710**	0.1714	0.1626
300	**1.0648**	**1.0620**	0.1128	0.1112	**1.1258**	**1.1225**	0.1224	0.1204	**1.2029**	**1.1950**	0.1370	0.1301
400	**1.0732**	**1.0707**	0.0840	0.0821	**1.1193**	**1.1172**	0.0962	0.0958	**1.2046**	**1.1975**	0.1088	0.1054
500	**1.0718**	**1.0700**	0.0804	0.0801	**1.1341**	**1.1288**	0.0972	0.0934	**1.1987**	**1.1903**	0.1280	0.1214
600	**1.0829**	**1.0801**	0.0806	0.0788	**1.1327**	**1.1286**	0.0789	0.0762	**1.2123**	**1.2069**	0.1050	0.1025
700	**1.0710**	**1.0691**	0.0609	0.0605	**1.1502**	**1.1471**	0.0727	0.0720	**1.2261**	**1.2192**	0.0904	0.0863
800	**1.0821**	**1.0807**	0.0656	0.0648	**1.1439**	**1.1403**	0.0834	0.0835	**1.2326**	**1.2265**	0.1018	0.0997
900	**1.0770**	**1.0757**	0.0540	0.0535	**1.1491**	**1.1451**	0.0740	0.0726	**1.2518**	**1.2446**	0.1097	0.1031
1000	**1.0745**	**1.0733**	0.0565	0.0559	**1.1477**	**1.1451**	0.0659	0.0642	**1.2446**	**1.2381**	0.0902	0.0868
1100	**1.0785**	**1.0768**	0.0579	0.0569	**1.1468**	**1.1433**	0.0664	0.0653	**1.2308**	**1.2264**	0.0760	0.0752
1200	**1.0916**	**1.0900**	0.0482	0.0483	**1.1448**	**1.1411**	0.0597	0.0573	**1.2510**	**1.2456**	0.0749	0.0702
1300	**1.0912**	**1.0898**	0.0523	0.0521	**1.1502**	**1.1474**	0.0589	0.0581	**1.2397**	**1.2344**	0.0893	0.0861
1400	**1.0851**	**1.0837**	0.0487	0.0481	**1.1462**	**1.1443**	0.0531	0.0524	**1.2419**	**1.2379**	0.0757	0.0746
1500	**1.0837**	**1.0828**	0.0490	0.0488	**1.1524**	**1.1497**	0.0610	0.0602	**1.2490**	**1.2450**	0.0711	0.0677
1600	**1.0903**	**1.0886**	0.0421	0.0415	**1.1530**	**1.1502**	0.0521	0.0512	**1.2388**	**1.2348**	0.0587	0.0560
1700	**1.0868**	**1.0852**	0.0418	0.0419	**1.1551**	**1.1526**	0.0491	0.0478	**1.2591**	**1.2534**	0.0704	0.0678
1800	**1.0902**	**1.0888**	0.0375	0.0379	**1.1627**	**1.1604**	0.0502	0.0498	**1.2661**	**1.2614**	0.0769	0.0747
1900	**1.0977**	**1.0968**	0.0381	0.0381	**1.1579**	**1.1551**	0.0485	0.0483	**1.2609**	**1.2565**	0.0734	0.0709
2000	**1.0927**	**1.0916**	0.0372	0.0368	**1.1534**	**1.1505**	0.0507	0.0492	**1.2655**	**1.2602**	0.0644	0.0622

Table 8: Comparison between sample and bootstrap variances, of fractionally Gaussian integrated noise for $d = 0.20, 0.25, 0.30$ and increasing series length.

	$d = 0.35$				$d = 0.40$				$d = 0.45$			
n	$\hat{\gamma}_0$	$\hat{\gamma}_0^*$	$\hat{se}(\hat{\gamma}_0)$	$\hat{se}(\hat{\gamma}_0^*)$	$\hat{\gamma}_0$	$\hat{\gamma}_0^*$	$\hat{se}(\hat{\gamma}_0)$	$\hat{se}(\hat{\gamma}_0^*)$	$\hat{\gamma}_0$	$\hat{\gamma}_0^*$	$\hat{se}(\hat{\gamma}_0)$	$\hat{se}(\hat{\gamma}_0^*)$
100	**1.2065**	**1.1928**	0.2436	0.2305	**1.2456**	**1.2310**	0.2834	0.2703	**1.5009**	**1.4642**	0.3574	0.3278
200	**1.2522**	**1.2375**	0.1922	0.1822	**1.4147**	**1.3910**	0.2629	0.2434	**1.5934**	**1.5519**	0.3061	0.2776
300	**1.2956**	**1.2840**	0.1942	0.1802	**1.4611**	**1.4406**	0.2374	0.2252	**1.6809**	**1.6422**	0.3515	0.3227
400	**1.3163**	**1.3048**	0.1673	0.1600	**1.5402**	**1.5203**	0.2633	0.2483	**1.7376**	**1.7030**	0.3579	0.3292
500	**1.3552**	**1.3447**	0.1740	0.1644	**1.5215**	**1.4976**	0.2351	0.2125	**1.7439**	**1.7113**	0.3415	0.3085
600	**1.3764**	**1.3623**	0.1417	0.1345	**1.5596**	**1.5347**	0.2460	0.2272	**1.8080**	**1.7724**	0.3205	0.2911
700	**1.3614**	**1.3492**	0.1337	0.1248	**1.5571**	**1.5382**	0.2112	0.1970	**1.7961**	**1.7608**	0.3277	0.2987
800	**1.3614**	**1.3506**	0.1464	0.1367	**1.5573**	**1.5397**	0.1812	0.1717	**1.8552**	**1.8234**	0.2544	0.2365
900	**1.3636**	**1.3544**	0.1267	0.1212	**1.5625**	**1.5463**	0.1603	0.1497	**1.8849**	**1.8438**	0.3291	0.2864
1000	**1.3890**	**1.3801**	0.1299	0.1256	**1.5909**	**1.5718**	0.1907	0.1775	**1.8489**	**1.8167**	0.2562	0.2337
1100	**1.3898**	**1.3791**	0.1325	0.1246	**1.5837**	**1.5659**	0.1670	0.1569	**1.9098**	**1.8777**	0.2946	0.2705
1200	**1.4002**	**1.3887**	0.1244	0.1199	**1.5951**	**1.5791**	0.1623	0.1531	**1.8647**	**1.8370**	0.2924	0.2674
1300	**1.3991**	**1.3882**	0.1144	0.1054	**1.6048**	**1.5888**	0.1546	0.1452	**1.9579**	**1.9279**	0.3020	0.2786
1400	**1.4056**	**1.3953**	0.1065	0.0997	**1.6116**	**1.5902**	0.1978	0.1858	**1.9419**	**1.9102**	0.2485	0.2302
1500	**1.3863**	**1.3752**	0.1156	0.1085	**1.5969**	**1.5793**	0.1624	0.1522	**1.9801**	**1.9459**	0.3115	0.2776
1600	**1.3956**	**1.3886**	0.0981	0.0941	**1.6223**	**1.6027**	0.1928	0.1748	**1.9852**	**1.9508**	0.2783	0.2536
1700	**1.4183**	**1.4088**	0.1156	0.1097	**1.6368**	**1.6189**	0.1642	0.1518	**1.9746**	**1.9388**	0.2917	0.2543
1800	**1.4141**	**1.4037**	0.1161	0.1082	**1.6470**	**1.6299**	0.1720	0.1613	**2.0033**	**1.9699**	0.3096	0.2885
1900	**1.4028**	**1.3949**	0.1046	0.1007	**1.6257**	**1.6114**	0.1849	0.1723	**1.9956**	**1.9638**	0.2538	0.2296
2000	**1.4035**	**1.3953**	0.0972	0.0925	**1.6384**	**1.6233**	0.1689	0.1554	**2.0084**	**1.9783**	0.2784	0.2568

Table 9: Comparison between sample and bootstrap variances, of fractionally Gaussian integrated noise for $d = 0.35, 0.40, 0.45$ and increasing series length.

	$d = 0.05$				$d = 0.10$				$d = 0.15$			
n	$\hat{\rho}_1$	$\hat{\rho}_1^*$	$\hat{se}(\hat{\rho}_1)$	$\hat{se}(\hat{\rho}_1^*)$	$\hat{\rho}_1$	$\hat{\rho}_1^*$	$\hat{se}(\hat{\rho}_1)$	$\hat{se}(\hat{\rho}_1^*)$	$\hat{\rho}_1$	$\hat{\rho}_1^*$	$\hat{se}(\hat{\rho}_1)$	$\hat{se}(\hat{\rho}_1^*)$
100	**0.0225**	**0.0192**	0.0657	0.0630	**0.0355**	**0.0266**	0.1113	0.1029	**0.0635**	**0.0517**	0.1052	0.0981
200	**0.0160**	**0.0140**	0.0586	0.0572	**0.0571**	**0.0525**	0.0743	0.0716	**0.0728**	**0.0664**	0.0812	0.0781
300	**0.0211**	**0.0192**	0.0551	0.0539	**0.0572**	**0.0537**	0.0581	0.0573	**0.0937**	**0.0885**	0.0676	0.0653
400	**0.0151**	**0.0140**	0.0447	0.0439	**0.0564**	**0.0542**	0.0544	0.0535	**0.0899**	**0.0855**	0.0576	0.0559
500	**0.0273**	**0.0261**	0.0413	0.0411	**0.0622**	**0.0594**	0.0462	0.0451	**0.1000**	**0.0967**	0.0471	0.0463
600	**0.0263**	**0.0253**	0.0366	0.0366	**0.0610**	**0.0588**	0.0440	0.0439	**0.1060**	**0.1027**	0.0462	0.0454
700	**0.0227**	**0.0218**	0.0332	0.0330	**0.0524**	**0.0509**	0.0431	0.0428	**0.0978**	**0.0953**	0.0386	0.0380
800	**0.0251**	**0.0246**	0.0353	0.0350	**0.0556**	**0.0541**	0.0351	0.0349	**0.1000**	**0.0969**	0.0397	0.0395
900	**0.0232**	**0.0226**	0.0330	0.0329	**0.0564**	**0.0552**	0.0347	0.0345	**0.1117**	**0.1093**	0.0377	0.0372
1000	**0.0207**	**0.0201**	0.0284	0.0281	**0.0589**	**0.0573**	0.0327	0.0327	**0.1054**	**0.1032**	0.0364	0.0358
1100	**0.0225**	**0.0222**	0.0312	0.0309	**0.0619**	**0.0604**	0.0353	0.0351	**0.1055**	**0.1031**	0.0352	0.0349
1200	**0.0224**	**0.0220**	0.0286	0.0283	**0.0616**	**0.0603**	0.0307	0.0307	**0.1099**	**0.1077**	0.0332	0.0326
1300	**0.0249**	**0.0245**	0.0273	0.0273	**0.0590**	**0.0582**	0.0273	0.0273	**0.1024**	**0.1005**	0.0277	0.0271
1400	**0.0240**	**0.0234**	0.0246	0.0247	**0.0623**	**0.0613**	0.0276	0.0273	**0.0970**	**0.0955**	0.0349	0.0344
1500	**0.0271**	**0.0267**	0.0263	0.0264	**0.0613**	**0.0602**	0.0278	0.0277	**0.1015**	**0.0999**	0.0308	0.0306
1600	**0.0298**	**0.0295**	0.0213	0.0213	**0.0612**	**0.0600**	0.0284	0.0284	**0.1020**	**0.1007**	0.0279	0.0277
1700	**0.0241**	**0.0236**	0.0241	0.0239	**0.0649**	**0.0641**	0.0249	0.0251	**0.1034**	**0.1017**	0.0272	0.0269
1800	**0.0270**	**0.0266**	0.0224	0.0225	**0.0639**	**0.0633**	0.0238	0.0236	**0.1076**	**0.1062**	0.0276	0.0272
1900	**0.0242**	**0.0241**	0.0251	0.0251	**0.0611**	**0.0602**	0.0242	0.0239	**0.1080**	**0.1069**	0.0271	0.0266
2000	**0.0270**	**0.0266**	0.0240	0.0241	**0.0599**	**0.0594**	0.0239	0.0238	**0.1067**	**0.1052**	0.0254	0.0255

Table 10: Comparison between sample and bootstrap autocorrelation functions at lag one, of fractionally Gaussian integrated noise for $d = 0.05, 0.10, 0.15$ and increasing series length.

	$d=0.20$				$d=0.25$				$d=0.30$			
n	$\hat{\rho}_1$	$\hat{\rho}_1^*$	$\hat{se}(\hat{\rho}_1)$	$\hat{se}(\hat{\rho}_1^*)$	$\hat{\rho}_1$	$\hat{\rho}_1^*$	$\hat{se}(\hat{\rho}_1)$	$\hat{se}(\hat{\rho}_1^*)$	$\hat{\rho}_1$	$\hat{\rho}_1^*$	$\hat{se}(\hat{\rho}_1)$	$\hat{se}(\hat{\rho}_1^*)$
100	**0.0874**	**0.0729**	0.1044	0.0965	**0.1508**	**0.1288**	0.1177	0.1056	**0.3245**	**0.2972**	0.1084	0.1002
200	**0.1132**	**0.1035**	0.0743	0.0700	**0.1665**	**0.1524**	0.0832	0.0757	**0.3517**	**0.3315**	0.0796	0.0735
300	**0.1449**	**0.1356**	0.0699	0.0654	**0.1851**	**0.1739**	0.0763	0.0712	**0.3632**	**0.3479**	0.0712	0.0671
400	**0.1471**	**0.1401**	0.0515	0.0489	**0.1869**	**0.1789**	0.0672	0.0649	**0.3710**	**0.3586**	0.0620	0.0589
500	**0.1440**	**0.1380**	0.0563	0.0544	**0.2092**	**0.1995**	0.0566	0.0529	**0.3687**	**0.3567**	0.0604	0.0572
600	**0.1444**	**0.1384**	0.0460	0.0440	**0.2020**	**0.1944**	0.0495	0.0463	**0.3793**	**0.3697**	0.0491	0.0466
700	**0.1439**	**0.1395**	0.0440	0.0427	**0.2039**	**0.1976**	0.0434	0.0422	**0.3823**	**0.3728**	0.0446	0.0421
800	**0.1554**	**0.1511**	0.0457	0.0443	**0.2149**	**0.2085**	0.0495	0.0484	**0.3897**	**0.3814**	0.0487	0.0467
900	**0.1542**	**0.1505**	0.0406	0.0396	**0.2141**	**0.2076**	0.0439	0.0421	**0.3896**	**0.3813**	0.0483	0.0449
1000	**0.1568**	**0.1534**	0.0393	0.0388	**0.2087**	**0.2034**	0.0434	0.0414	**0.3903**	**0.3827**	0.0424	0.0405
1100	**0.1521**	**0.1485**	0.0402	0.0394	**0.2178**	**0.2123**	0.0413	0.0394	**0.3860**	**0.3799**	0.0375	0.0370
1200	**0.1606**	**0.1569**	0.0340	0.0334	**0.2137**	**0.2092**	0.0379	0.0365	**0.3902**	**0.3841**	0.0402	0.0384
1300	**0.1593**	**0.1564**	0.0335	0.0331	**0.2153**	**0.2103**	0.0446	0.0433	**0.3909**	**0.3846**	0.0430	0.0412
1400	**0.1499**	**0.1467**	0.0296	0.0289	**0.2172**	**0.2136**	0.0322	0.0313	**0.3960**	**0.3905**	0.0366	0.0354
1500	**0.1550**	**0.1523**	0.0293	0.0288	**0.2214**	**0.2173**	0.0374	0.0359	**0.3948**	**0.3894**	0.0358	0.0342
1600	**0.1612**	**0.1587**	0.0308	0.0302	**0.2209**	**0.2167**	0.0353	0.0341	**0.3921**	**0.3875**	0.0302	0.0289
1700	**0.1501**	**0.1475**	0.0298	0.0291	**0.2185**	**0.2148**	0.0357	0.0348	**0.3950**	**0.3894**	0.0340	0.0328
1800	**0.1627**	**0.1602**	0.0290	0.0282	**0.2207**	**0.2171**	0.0316	0.0307	**0.4050**	**0.3999**	0.0402	0.0388
1900	**0.1598**	**0.1575**	0.0281	0.0277	**0.2240**	**0.2201**	0.0313	0.0306	**0.4040**	**0.3990**	0.0349	0.0336
2000	**0.1568**	**0.1548**	0.0253	0.0249	**0.2242**	**0.2206**	0.0319	0.0307	**0.4021**	**0.3971**	0.0306	0.0294

Table 11: Comparison between sample and bootstrap autocorrelation functions at lag one, of fractionally Gaussian integrated noise for $d = 0.20, 0.25, 0.30$ and increasing series length.

	d = 0.35				d = 0.40				d = 0.45			
n	$\hat{\rho}_1$	$\hat{\rho}_1^*$	$\hat{se}(\hat{\rho}_1)$	$\hat{se}(\hat{\rho}_1^*)$	$\hat{\rho}_1$	$\hat{\rho}_1^*$	$\hat{se}(\hat{\rho}_1)$	$\hat{se}(\hat{\rho}_1^*)$	$\hat{\rho}_1$	$\hat{\rho}_1^*$	$\hat{se}(\hat{\rho}_1)$	$\hat{se}(\hat{\rho}_1^*)$
100	**0.3727**	**0.3383**	0.1183	0.1078	**0.4228**	**0.3857**	0.1099	0.0985	**0.5306**	**0.4822**	0.1072	0.0964
200	**0.4128**	**0.3898**	0.0886	0.0814	**0.4880**	**0.4586**	0.0924	0.0861	**0.5653**	**0.5295**	0.0853	0.0786
300	**0.4344**	**0.4167**	0.0824	0.0757	**0.5080**	**0.4856**	0.0759	0.0705	**0.5912**	**0.5637**	0.0750	0.0656
400	**0.4417**	**0.4265**	0.0706	0.0669	**0.5343**	**0.5150**	0.0774	0.0717	**0.5975**	**0.5748**	0.0783	0.0724
500	**0.4566**	**0.4433**	0.0621	0.0581	**0.5374**	**0.5186**	0.0623	0.0571	**0.6075**	**0.5870**	0.0722	0.0649
600	**0.4619**	**0.4479**	0.0578	0.0539	**0.5476**	**0.5299**	0.0626	0.0566	**0.6193**	**0.6000**	0.0615	0.0553
700	**0.4654**	**0.4529**	0.0493	0.0457	**0.5497**	**0.5349**	0.0612	0.0565	**0.6228**	**0.6051**	0.0638	0.0575
800	**0.4702**	**0.4591**	0.0515	0.0480	**0.5480**	**0.5342**	0.0551	0.0520	**0.6339**	**0.6168**	0.0490	0.0453
900	**0.4627**	**0.4530**	0.0455	0.0428	**0.5532**	**0.5407**	0.0462	0.0427	**0.6359**	**0.6191**	0.0566	0.0499
1000	**0.4757**	**0.4667**	0.0448	0.0423	**0.5560**	**0.5433**	0.0511	0.0467	**0.6353**	**0.6202**	0.0498	0.0449
1100	**0.4792**	**0.4697**	0.0448	0.0413	**0.5578**	**0.5461**	0.0497	0.0466	**0.6441**	**0.6292**	0.0522	0.0472
1200	**0.4825**	**0.4732**	0.0458	0.0433	**0.5625**	**0.5515**	0.0445	0.0418	**0.6367**	**0.6236**	0.0530	0.0488
1300	**0.4792**	**0.4700**	0.0426	0.0391	**0.5629**	**0.5523**	0.0444	0.0413	**0.6535**	**0.6401**	0.0499	0.0455
1400	**0.4838**	**0.4754**	0.0418	0.0388	**0.5665**	**0.5544**	0.0498	0.0464	**0.6517**	**0.6381**	0.0444	0.0404
1500	**0.4837**	**0.4753**	0.0406	0.0379	**0.5657**	**0.5555**	0.0409	0.0375	**0.6559**	**0.6425**	0.0493	0.0431
1600	**0.4820**	**0.4753**	0.0408	0.0390	**0.5705**	**0.5599**	0.0453	0.0408	**0.6595**	**0.6463**	0.0437	0.0390
1700	**0.4903**	**0.4827**	0.0414	0.0390	**0.5746**	**0.5647**	0.0402	0.0364	**0.6575**	**0.6448**	0.0444	0.0391
1800	**0.4884**	**0.4810**	0.0430	0.0395	**0.5773**	**0.5676**	0.0458	0.0423	**0.6611**	**0.6484**	0.0475	0.0428
1900	**0.4838**	**0.4774**	0.0368	0.0344	**0.5687**	**0.5600**	0.0458	0.0419	**0.6615**	**0.6494**	0.0426	0.0384
2000	**0.4873**	**0.4810**	0.0354	0.0335	**0.5748**	**0.5662**	0.0395	0.0359	**0.6615**	**0.6497**	0.0448	0.0401

Table 12: Comparison between sample and bootstrap autocorrelation functions at lag one, of fractionally Gaussian integrated noise for $d = 0.35, 0.40, 0.45$ and increasing series length.

References

Albin, J. (1998a). A note on rosenblatt distributions. *Statistics & Probability Letters 40*(1), 83–91.

Albin, J. (1998b). On extremal theory for self-similarity processes. *The Annals of Probability 26*(2), 743–793.

Bartlett, M. S. (1946). On the theoretical specification and sampling properties of autocorrelated time-series. *Supplement of the Journal of the Royal Statistical Society 8*(1), 27–41.

Dahlhaus, R. (1988). Small sample effects in time series analysis: a new asymptotic theory and a new estimate. *The Annals of Statistics 16*(2), 808–841.

Dahlhaus, R. (1989). Efficient parameter estimation for self-similar processes. *Annals of Statistics 17*, 1749–1766.

Dahlhaus, R. and D. Janas (1996). A frequency domain bootstrap for ratio statistics in time series analysis. *The Annals of Statistics 24*(5), 1934–1963.

Fox, R. and M. Taqqu (1986). Large-sample properties of parameter estimates for strongly dependent stationary gaussian time series. *The Annals of Statistics 14*(2), 517–532.

Franke, J. and W. Härdle (1992). On bootstrapping kernel spectral estimates. *The Annals of Statistics 20*(1), 121–145.

Geweke, J. and S. Porter-Hudak (1983). The estimation and application of long memory time series model. *Journal of Time Series Analysis 4*(4), 221–238.

Granger, C. and R. Joyeux (1980). An introduction to long memory time series models and fractional differencing. *Journal of Time Series Analysis 1*(1), 15–29.

Hall, P. (1992). Effect of bias estimation on coverage accuracy of bootstrap confidence intervals for a probability density. *The Annals of Statistics 20*(2), 675–694.

Hosking, J. (1981). Fractional differencing. *Biometrika 68*(1), 165–176.

Hosking, J. (1996). Asymptotic distributions of the sample mean, autocovariances, and autocorrelations of long-memory time series. *Journal of Econometrics 73*, 261–284.

Hurst, H. (1951). Long-term storage capacity of reservoirs. *Transactions of the American Society of Civil Engineers 116*, 770–799.

Hurvich, C., R. Deo, and J. Brodsky (1998). The mena squared error of Geweke and Porter-Hudak's estimator of the memory parameter of a long-memory time series. *Journal of Time Series Analysis 19*(1), 19–46.

Kreiss, J. (1992). *Bootstrap procedures for AR(∞)-process.* Springer: Heidelberg.

Künsch, H. (1989). The jackknife and the bootstrap for general stationary observations. *The Annals of Statistics 17*(3), 1217–1241.

Newbold, P. and C. Agiakloglou (1993). Bias in the sample autocorrelations of fractional noise. *Biometrika 80*(3), 698–702.

Paparoditis, E. and D. Politis (1999). The local bootstrap for periodogram statistics. *Journal of Time Series Analysis 20*(2), 193–222.

Ramsey, F. (1974). Characterization of the partial autocorrelation function. *Annals of Statistics 2*, 1296–1301.

Robinson, P. (1995). Gaussian semiparametric estimation of long range dependence. *The Annals of Statistics 23*(5), 1630–1661.

Theiler, J., S. Eubank, A. Longtin, B. Galdrikian, and J. Farmer (1992). Testing for nonlinearity in time series: the method of surrogate data. *Physica D 58*, 77–94.

Yajima, Y. (1985). On estimation of long-memory time series models. *Australian Journal of Statistics 27*, 303–320.