On estimation of nonparametric regression models with autoregressive and moving average errors

Qi Zheng *

Department of Bioinformatics & Biostatistics, University of Louisville

and

Yunwei Cui

Department of Mathematics, Towson University

and

Rongning Wu

Zicklin School of Business at Baruch College, The City University of New York

The `stats::arima()` function of R is capable of fitting the regression coefficients and ARMA parameter simultaneously. We performed a study to compare the performance of our R code and `stats::arima()` for estimating the smooth functions. Especially, we compared $\rho(\cdot)$ and $\rho_{19}(\cdot)$ for three smooth functions as used in Zheng, Cui, and Wu (2023)

$$\begin{cases} f_1(X_t) = 1 - 6X_t + 36X_t^2 - 53X_t^3 + 22X_t^5; \\ f_2(X_t) = \sin(2\pi X_t) + 2X_t^2; \\ f_3(X_t) = \arctan(5X_t - 5/2) - X_t^2/3. \end{cases}$$

In this simulation, the B-spline basis matrix is generated by the `splines::bs()` function, for which the knots are automatically generated by its default method as a sequence of equally spaced quantile points. For sample path with length $n = 1000$, 9 inner knots are used.

After examining various model specifications, we discovered that the two functions produce nearly identical results in terms of smooth function accuracy. However, our R code appears to generate slightly better results than stats::arima() does.

*Corresponding author. University of Louisville, Louisville, KY 40202, USA (Phone: 502-852-8780; Fax: 502-852-3294; E-mail: qi.zheng@louisville.edu)

Table 1: Comparing $\hat{g}(\cdot)$ and $g_0(\cdot)$, when ϵ_t's follow an ARMA(1, 1), or an AR(2), or an MA(2) process, X_t's are serially correlated and satisfy the conditions of Theorem 2, and innovations ζ_t's have a t distribution with degrees of freedom ν. All sample paths have $n = 1000$. In the table, `one step` denotes the R routine developed by Zheng, Cui, and Wu (2023), whereras `arima()` denotes the R function provided by `stats` package.

	ARMA(1,1), $(\phi, \theta) = (0.2, -0.5)$					
	$f_1(X_t)$		$f_2(X_t)$		$f_3(X_t)$	
	$\rho(g_0, \hat{g})$	$\rho_{19}(g_0, \hat{g})$	$\rho(g_0, \hat{g})$	$\rho_{19}(g_0, \hat{g})$	$\rho(g_0, \hat{g})$	$\rho_{19}(g_0, \hat{g})$
`one step`	0.1986	0.0568	0.1680	0.0444	0.1556	0.0406
`arima()`	0.1987	0.0568	0.1684	0.0444	0.1559	0.0407
	AR(2), $(\phi_1, \phi_2) = (0.5, 0.1)$					
	$f_1(X_t)$		$f_2(X_t)$		$f_3(X_t)$	
	$\rho(g_0, \hat{g})$	$\rho_{19}(g_0, \hat{g})$	$\rho(g_0, \hat{g})$	$\rho_{19}(g_0, \hat{g})$	$\rho(g_0, \hat{g})$	$\rho_{19}(g_0, \hat{g})$
`one step`	0.1939	0.0609	0.1783	0.0682	0.1828	0.0581
`arima()`	0.1957	0.0612	0.1797	0.0686	0.1832	0.0583
	MA(2), $(\theta_1, \theta_2) = (0.4, 0.2)$					
	$f_1(X_t)$		$f_2(X_t)$		$f_3(X_t)$	
	$\rho(g_0, \hat{g})$	$\rho_{19}(g_0, \hat{g})$	$\rho(g_0, \hat{g})$	$\rho_{19}(g_0, \hat{g})$	$\rho(g_0, \hat{g})$	$\rho_{19}(g_0, \hat{g})$
`one step`	0.2068	0.0542	0.1937	0.0646	0.1950	0.0542
`arima()`	0.2068	0.0542	0.1937	0.0646	0.1950	0.0542