Finite-Sample Performance of NeighBlock and NeighCoeff Estimators

T. Tony Cai
Department of Statistics
The Wharton School
University of Pennsylvania
Philadelphia, PA 19104

Bernard W. Silverman
Department of Mathematics
University of Bristol
Bristol BS8 1TW
U.K.

Abstract

We discuss in this report additional numerical results on NeighBlock and NeighCoeff estimators introduced in Cai and Silverman [4]. Comparisons, both numerical and visual, are made with conventional wavelet estimators, VisuShrink, SureShrink and TI denoising. Numerical results show that both NeighBlock and NeighCoeff perform excellently. The estimators also produce qualitatively appealing reconstructions.

Keywords: Adaptivity; Wavelet; Block Thresholding; Nonparametric function Estimation; Mean Squared Error.
AMS 1991 Subject Classification: Primary 62G07, Secondary 62G20.

1 Numerical Results

A simulation study was conducted to compare the numerical performance of the NeighBlock and NeighCoeff estimators with Donoho and Johnstone's VisuShrink and SureShrink as well as Coifman and Donoho's Translation-Invariant (TI) denoising method. SureShrink selects the threshold at each resolution level by minimizing Stein's unbiased estimate of risk. In the simulation, we use the hybrid method proposed in Donoho and Johnstone [6]. The TI denoising method was introduced by Coifman and Donoho [3], and is equivalent to averaging over estimators based on all the shifts of the original data. This method has various advantages over the universal thresholding methods. For further details see the original papers.

We implement the NeighBlock and NeighCoeff estimators in the software package S+Wavelets. The SPLUS codes implementing the estimators can be found at

http://www.stat.purdue.edu/people/tcai/neighblock.html

We compare the numerical performance of the methods using eight test functions representing different level of spatial variability. The test functions are plotted in Figure 2. Sample sizes ranging from $n = 512$ to $n = 8192$ and root-signal-to-noise ratios (RSNR) from 3 to 7 were considered. The RSNR is the ratio of the standard deviation of the function values to the standard deviation of the noise. Several different wavelets were used.

We report in detail the results using Daubechies' compactly supported wavelet *Symmlet* 8. Tables 1 - 5 report the average squared errors over 60 replications with sample sizes ranging from $n = 512$ to $n = 8192$ and RSNR from 3 to 7. Graphical presentations are given in Figures 3 - 7. Different combinations of wavelets and signal-to-noise ratios yield basically the same results.

The NeighBlock and NeighCoeff methods both uniformly outperform VisuShrink in all examples. For five of the eight test functions, Doppler, Bumps, Blocks, Spikes and Blip, our methods have better precision with sample size n than VisuShrink with sample size $2n$ for all sample sizes where the comparison is possible. The NeighCoeff method is slightly better than NeighBlock in almost all cases, and outperforms the other methods as well. The NeighCoeff method is also better than TI denoising in most cases, especially when the underlying function is of significant spatial variability. In terms of the mean square error criterion, the only conceivable competitor among the standard methods is SureShrink. Apart from being somewhat superior to SureShrink in mean square error, our methods yield noticeably better results visually. Our estimates do not contain the spurious fine-scale effects that are often contained in the SureShrink estimator.

It would be interesting to include comparisons with Hall, Kerkyacharian and Picard's block thresholding estimator. We do not include the HKP estimator here for two reasons. It is not easy to implement, and furthermore simulation results by Hall, Penev, Kerkyacharian, and Picard [7] show that even the translation-averaged version of their estimator has little advantage over VisuShrink when the signal to noise ratio is high. Our simulation shows that NeighBlock uniformly outperforms VisuShrink in all examples, and indeed the relative performance of VisuShrink is even worse for values of RSNR higher than the one presented in detail. Therefore we expect our estimator to perform favorably over HKP's estimator in terms of mean squared error, at least in the case of high signal-to-noise-ratio.

The curious behavior of some of the methods with the Waves signal calls for some explanation. Throughout, the primary resolution level $j_0 = [\log_2 \log n] + 1$ was used for all methods. Thus, $j_0 = 3$ for $n \leq 2048$, and $j_0 = 4$ for $n = 4096$ and 8192. This change in the value of j_0 affects whether or not the high frequency effect in the Waves signal is felt in the lowest level of wavelet coefficients. For $j_0 = 3$, the standard methods all smooth out the high frequency effect to some extent, because of applying a soft threshold with fixed threshold. An attractive feature of the NeighCoeff and NeighBlock methods is that they are not sensitive to the choice of primary resolution level in this way, because the threshold adapts to the presence of signal in all the coefficients.

2 Qualitative Comparisons

Both NeighBlock and NeighCoeff are appealing quantitatively as well as qualitatively. The reconstructions jump where the target function jump; the reconstruction is smooth where the target function is smooth. They do not contain the spurious fine-scale structure contained in some wavelet estimators, but adapt well to subtle changes in the underlying functions.

Figure 1 shows a typical segment of the result of the four methods applied to the inductance plethysmography data analyzed, for example, by Abramovich, Sapatinas and Silverman [1]. It can be seen that VisuShrink smooths out the broad features of the curve, while the SureShrink estimator allows through high frequency effects that are almost certainly spurious.

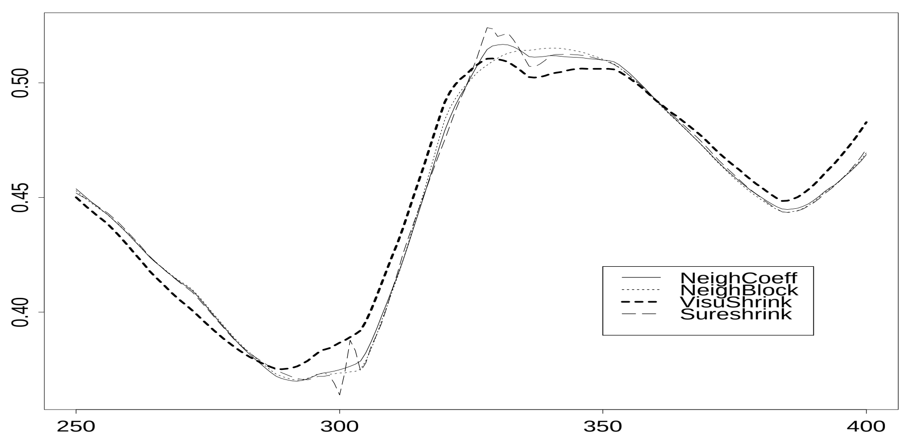

Figure 1: Curve estimates for a segment of the inductance plethysmography data. ——— NeighCoeff; ······ NeighBlock; – – – VisuShrink; — — SureShrink. The VisuShrink estimate smooths out the broad features, while the SureShrink estimate contains high frequency effects near times 300 and 335, both of which are presumably spurious.

In Figures 8 - 15, we compare the visual quality of the reconstructions of the five methods using the eight test functions. We use sample size $n = 1024$ and $RSNR = 3$ in all cases. It is clear from the figures, NeighBlock and NeighCoeff yield visually appealing reconstructions.

3 Test functions and numerical comparisons

The formulae of the test functions. (The test functions are normalized so that all of the functions have the same $s.d.(f) = 100$.) Doppler, HeaviSine, Bumps and Blocks are from Donoho and Johnstone (1994). Blip and Wave are from Marron, Adak, Johnstone, Neumann and Patil (1995).

1. *Doppler.*

$$f(x) = 34.5856 \cdot \sqrt{x(1-x)} \sin(2.1\pi/(x+.05))$$

2. *HeaviSine.*

$$f(x) = 3.3662 \cdot [4 \sin 4\pi x - \operatorname{sgn}(x - .3) - \operatorname{sgn}(.72 - x)]$$

3. *Bumps.*

$$f(x) = 15.0769 \cdot \sum h_j K((x - x_j)/w_j) \qquad K(x) = (1 + |x|)^{-4}.$$

$$\begin{array}{llllllllllll} (x_j) & = (.1, & .13, & .15, & .23, & .25, & .40, & .44, & .65, & .76, & .78, & .81) \\ (h_j) & = (4, & 5, & 3, & 4, & 5, & 4.2, & 2.1, & 4.3, & 3.1, & 5.1, & 4.2) \\ (w_j) & = (.005, & .005, & .006, & .01, & .01, & .03, & .01, & .01, & .005, & .008, & .005) \end{array}$$

4. *Blocks.*

$$f(x) = 4.7606 \cdot \sum h_j K(x - x_j) \qquad K(x) = (1 + \operatorname{sgn}(x))/2.$$

$$\begin{array}{llllllllllll} (x_j) & = (.1, & .13, & .15, & .23, & .25, & .40, & .44, & .65, & .76, & .78, & .81) \\ (h_j) & = (4, & -5, & 3, & -4, & 5, & -4.2, & 2.1, & 4.3, & -3.1, & 5.1, & -4.2) \end{array}$$

5. *Spikes.*

$$\begin{aligned} f(x) \quad &= \quad 15.6676 \cdot \Big[e^{-500(x-0.23)^2} + 2\, e^{-2000(x-0.33)^2} \\ &\qquad + 4\, e^{-8000(x-0.47)^2} + 3\, e^{-16000(x-0.69)^2} + e^{-32000(x-0.83)^2} \Big] \end{aligned}$$

6. *Blip.*

$$f(x) = 50.9859 \cdot [(0.32 + 0.6x + 0.3 e^{-100(x-0.3)^2}) I_{(0,.8]}(x) + (-0.28 + 0.6x + 0.3 e^{-100(x-1.3)^2}) I_{[.8,1]}(x)]$$

7. *Corner.*

$$f(x) = 62.3865 \cdot [10x^3(1-4x^2) I_{(0,.5]}(x) + 3(0.125 - x^3)x^4 I_{(.5,.8]}(x) + 59.4432(x-1)^3 I_{(.8,1]}(x)]$$

8. *Wave.*

$$f(x) = 63.2301 \cdot [.5 + .2\cos(4\pi x) + .1\cos(24\pi x)]$$

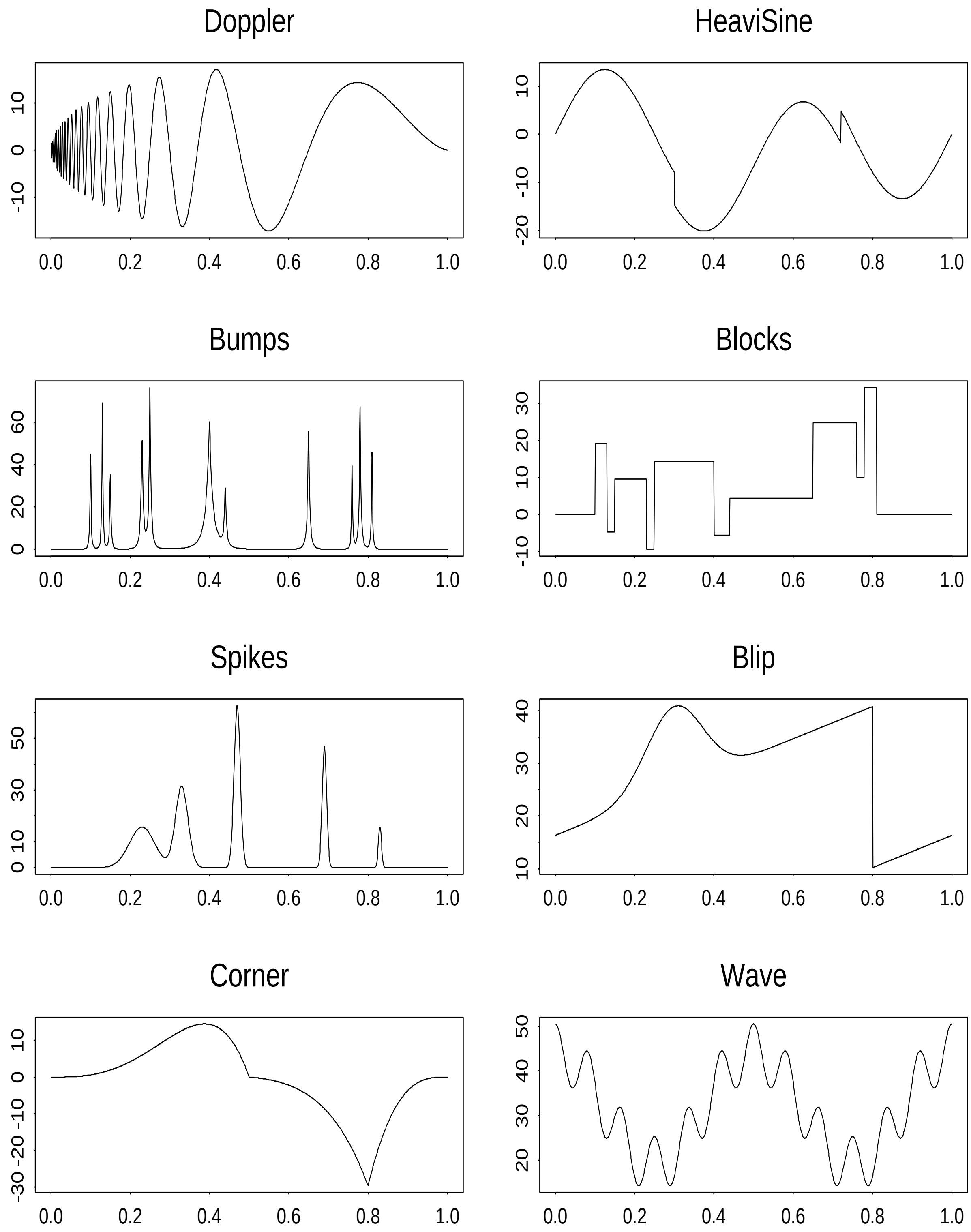

Figure 2: Test functions. Doppler, HeaviSine, Bumps and Blocks are from Donoho and Johnstone [5]. Blip and Wave are from Marron, Adak, Johnstone, Neumann and Patil [8]. The test functions are normalized so that every function has standard deviation 10.

Table 1: Mean Squared Error From 60 Replications (RSNR=3)

n	NeighCoeff	NeighBlock	SureShrink	TI denoising	VisuShrink
Doppler					
512	2.22	2.36	2.91	5.13	6.76
1024	1.34	1.35	1.98	3.36	4.49
2048	0.83	0.82	1.23	2.24	2.96
4096	0.51	0.50	0.68	1.25	1.61
8192	0.30	0.26	0.43	0.77	1.05
HeaviSine					
512	0.82	0.82	0.81	0.81	0.83
1024	0.59	0.63	0.56	0.62	0.63
2048	0.46	0.47	0.41	0.48	0.51
4096	0.28	0.36	0.30	0.29	0.36
8192	0.16	0.23	0.18	0.20	0.26
Bumps					
512	6.73	8.38	7.17	15.90	20.98
1024	3.66	4.24	4.04	10.08	13.63
2048	2.11	2.28	2.50	6.34	8.99
4096	1.08	1.75	1.54	3.42	5.09
8192	0.57	0.90	0.73	2.05	3.14
Blocks					
512	5.49	6.30	5.68	10.45	11.84
1024	3.78	4.09	3.65	7.37	8.29
2048	2.28	2.42	2.16	4.99	5.55
4096	1.39	1.96	1.42	2.92	3.38
8192	0.83	1.23	0.95	1.94	2.32
Spikes					
512	1.92	2.19	2.00	4.88	6.13
1024	1.18	1.31	1.35	3.11	4.00
2048	0.67	0.70	0.76	1.80	2.48
4096	0.38	0.49	0.42	0.71	1.19
8192	0.22	0.25	0.25	0.41	0.78
Blip					
512	1.06	1.33	1.50	1.80	1.94
1024	0.70	0.83	0.98	1.20	1.36
2048	0.39	0.43	0.55	0.77	0.93
4096	0.24	0.39	0.37	0.43	0.52
8192	0.13	0.19	0.21	0.28	0.34
Corner					
512	0.67	0.74	0.76	0.61	1.06
1024	0.36	0.41	0.40	0.40	0.69
2048	0.19	0.21	0.22	0.26	0.43
4096	0.11	0.15	0.13	0.12	0.16
8192	0.06	0.07	0.06	0.07	0.10
Wave					
512	2.65	2.84	3.15	5.75	7.14
1024	1.36	1.43	2.90	3.67	5.08
2048	0.55	0.54	3.18	2.22	3.27

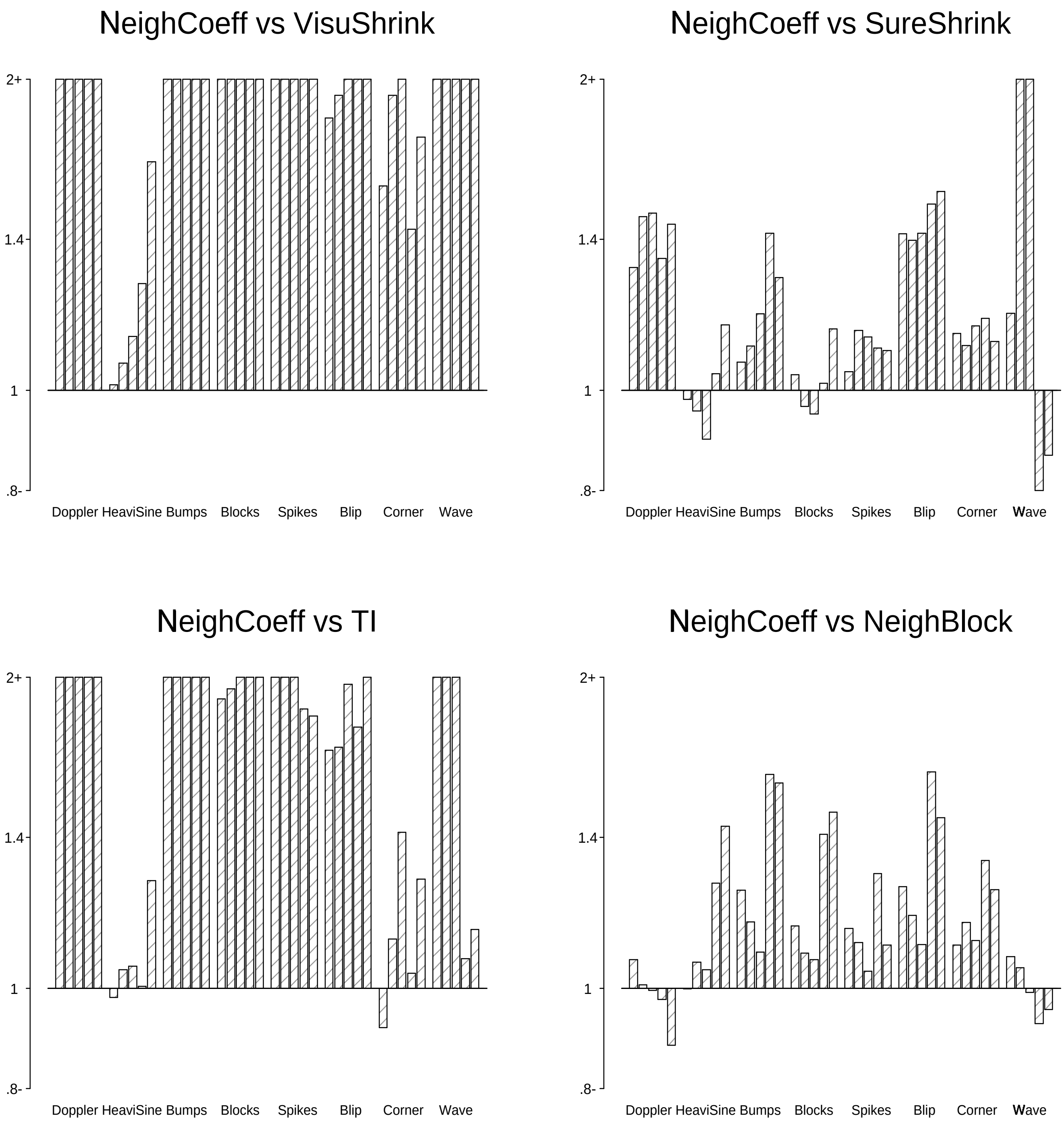

Figure 3: RSNR=3. The vertical bars represent the ratios of the MSEs of various estimators to the corresponding MSE of the NeighCoeff estimator. The higher the bar the better the relative performance of the NeighCoeff estimator. The bars are plotted on a log scale and are truncated at the value 2 of the original ratio. For each signal the bars are ordered from left to right by the sample sizes (n=512 to 8192).

Table 2: Mean Squared Error From 60 Replications (RSNR=4)

n	NeighCoeff	NeighBlock	SureShrink	TI denoising	VisuShrink
Doppler					
512	1.41	1.52	2.09	3.53	4.54
1024	0.91	0.93	1.36	2.32	3.09
2048	0.54	0.52	0.72	1.43	1.91
4096	0.31	0.28	0.46	0.81	1.08
8192	0.19	0.15	0.26	0.49	0.71
HeaviSine					
512	0.64	0.64	0.62	0.64	0.66
1024	0.45	0.48	0.43	0.48	0.52
2048	0.29	0.31	0.29	0.34	0.41
4096	0.17	0.25	0.20	0.21	0.27
8192	0.10	0.15	0.11	0.14	0.19
Bumps					
512	3.75	4.63	4.29	10.77	14.90
1024	2.04	2.39	2.38	6.33	9.18
2048	1.22	1.31	1.71	3.91	5.95
4096	0.62	1.02	0.80	2.12	3.24
8192	0.33	0.52	0.47	1.26	2.00
Blocks					
512	3.37	3.93	3.76	7.31	8.28
1024	2.21	2.50	2.14	4.96	5.67
2048	1.39	1.51	1.51	3.36	3.84
4096	0.80	1.30	0.96	1.94	2.35
8192	0.50	0.78	0.64	1.28	1.59
Spikes					
512	1.24	1.45	1.41	3.05	4.04
1024	0.68	0.76	0.83	1.81	2.51
2048	0.40	0.41	0.46	1.09	1.59
4096	0.24	0.27	0.27	0.45	0.81
8192	0.13	0.14	0.12	0.25	0.53
Blip					
512	0.64	0.80	1.01	1.11	1.30
1024	0.41	0.47	0.60	0.75	0.95
2048	0.20	0.22	0.34	0.47	0.61
4096	0.13	0.21	0.22	0.28	0.35
8192	0.08	0.11	0.13	0.17	0.23
Corner					
512	0.40	0.46	0.44	0.42	0.73
1024	0.21	0.22	0.24	0.27	0.44
2048	0.10	0.11	0.13	0.17	0.28
4096	0.06	0.08	0.07	0.08	0.11
8192	0.03	0.04	0.04	0.05	0.06
Wave					
512	1.49	1.72	2.82	3.77	5.15
1024	0.66	0.69	3.12	2.32	3.37
2048	0.28	0.28	3.14	1.33	2.02

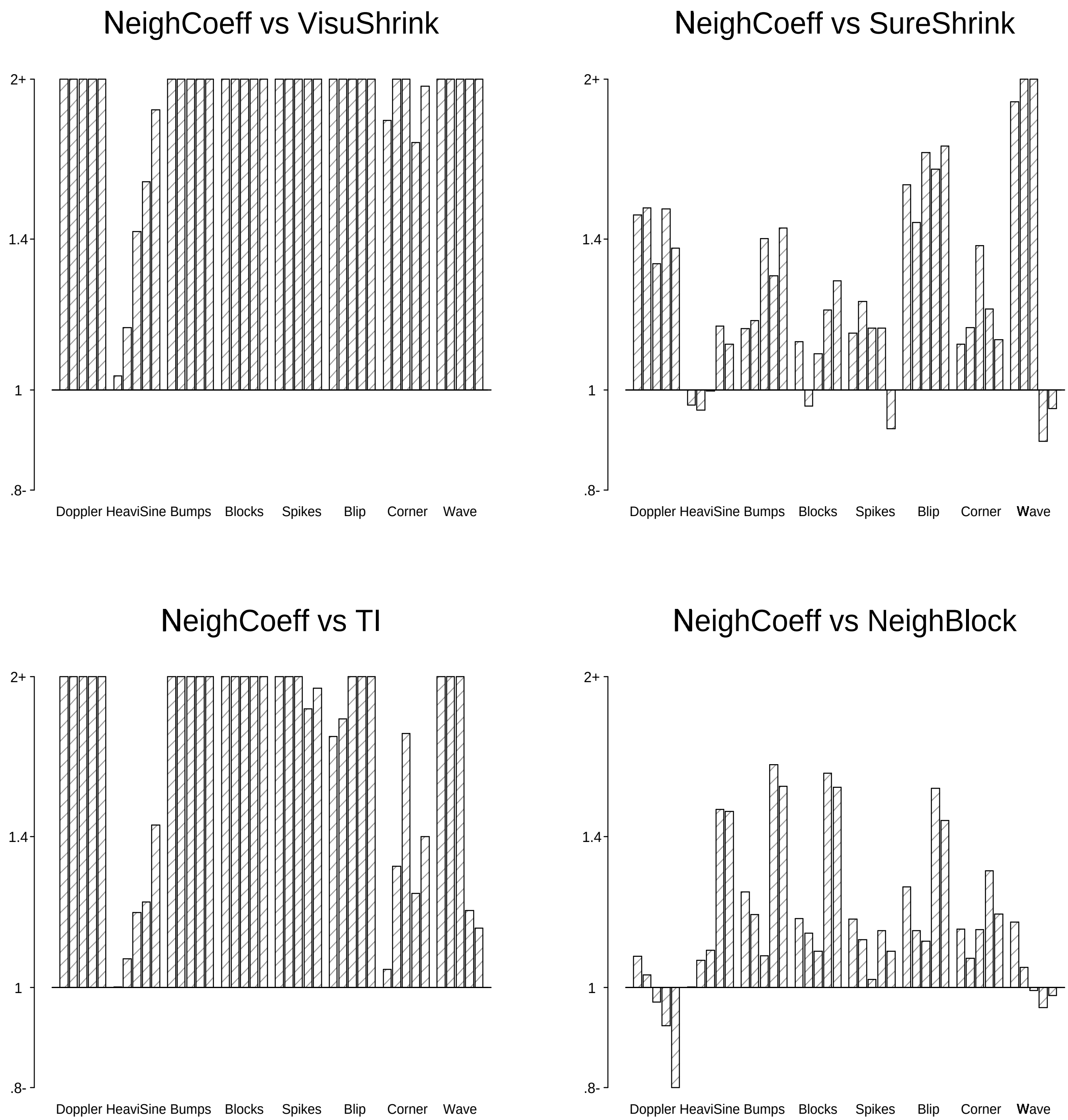

Figure 4: RSNR=4. The vertical bars represent the ratios of the MSEs of various estimators to the corresponding MSE of the NeighCoeff estimator. The higher the bar the better the relative performance of the NeighCoeff estimator. The bars are plotted on a log scale and are truncated at the value 2 of the original ratio. For each signal the bars are ordered from left to right by the sample sizes (n=512 to 8192).

Table 3: Mean Squared Error From 60 Replications (RSNR=5)

n	NeighCoeff	NeighBlock	SureShrink	TI denoising	VisuShrink
Doppler					
512	1.05	1.12	1.62	2.65	3.44
1024	0.65	0.66	0.93	1.66	2.24
2048	0.37	0.36	0.54	1.01	1.40
4096	0.21	0.18	0.34	0.55	0.80
8192	0.13	0.09	0.18	0.34	0.52
HeaviSine					
512	0.51	0.56	0.50	0.54	0.57
1024	0.33	0.37	0.33	0.39	0.44
2048	0.20	0.22	0.22	0.27	0.34
4096	0.11	0.17	0.13	0.16	0.20
8192	0.06	0.10	0.08	0.11	0.14
Bumps					
512	2.26	2.96	2.27	7.72	11.04
1024	1.33	1.53	1.71	4.48	6.65
2048	0.77	0.84	1.09	2.71	4.16
4096	0.41	0.69	0.57	1.49	2.33
8192	0.22	0.34	0.34	0.85	1.40
Blocks					
512	2.23	2.74	2.60	5.38	6.41
1024	1.37	1.62	1.59	3.58	4.16
2048	0.88	0.98	1.03	2.40	2.82
4096	0.52	0.89	0.71	1.39	1.70
8192	0.32	0.53	0.44	0.89	1.16
Spikes					
512	0.89	1.02	1.06	2.06	2.79
1024	0.49	0.52	0.55	1.29	1.82
2048	0.28	0.27	0.33	0.72	1.15
4096	0.16	0.18	0.16	0.30	0.60
8192	0.08	0.09	0.08	0.17	0.38
Blip					
512	0.40	0.51	0.66	0.75	0.96
1024	0.23	0.28	0.42	0.51	0.70
2048	0.14	0.14	0.24	0.32	0.44
4096	0.08	0.13	0.15	0.18	0.26
8192	0.05	0.07	0.09	0.12	0.16
Corner					
512	0.24	0.29	0.29	0.30	0.50
1024	0.12	0.14	0.16	0.19	0.32
2048	0.07	0.08	0.09	0.12	0.20
4096	0.04	0.05	0.05	0.06	0.08
8192	0.02	0.03	0.03	0.03	0.05
Wave					
512	0.82	0.94	2.94	2.57	3.72
1024	0.37	0.37	3.09	1.50	2.38
2048	0.17	0.18	3.39	0.88	1.31

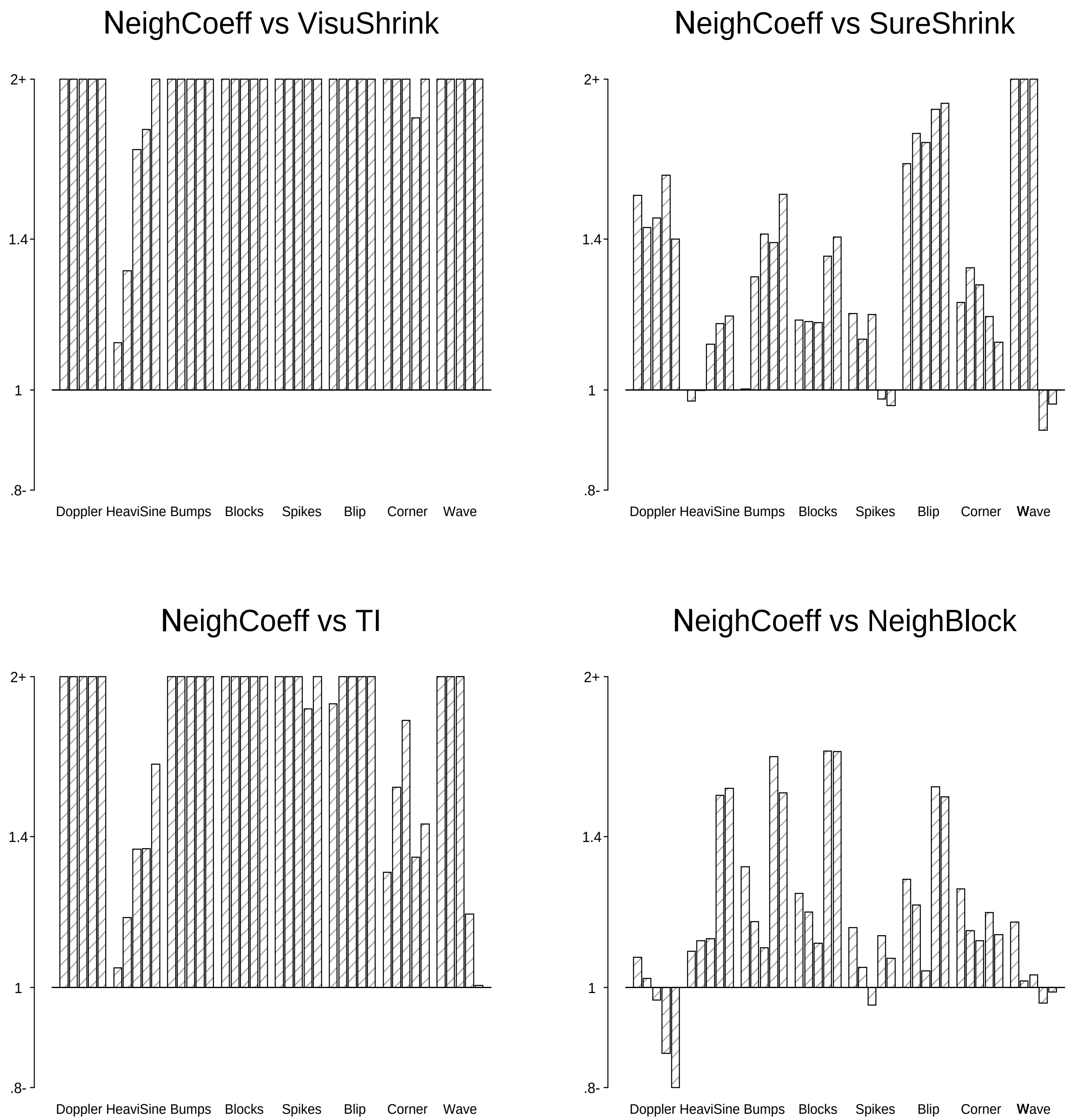

Figure 5: RSNR=5. The vertical bars represent the ratios of the MSEs of various estimators to the corresponding MSE of the NeighCoeff estimator. The higher the bar the better the relative performance of the NeighCoeff estimator. The bars are plotted on a log scale and are truncated at the value 2 of the original ratio. For each signal the bars are ordered from left to right by the sample sizes (n=512 to 8192).

Table 4: Mean Squared Error From 60 Replications (RSNR=6)

n	NeighCoeff	NeighBlock	SureShrink	TI denoising	VisuShrink
Doppler					
512	0.83	0.91	1.24	2.09	2.66
1024	0.49	0.51	0.68	1.27	1.71
2048	0.28	0.27	0.44	0.77	1.11
4096	0.16	0.13	0.22	0.42	0.61
8192	0.09	0.07	0.13	0.25	0.39
HeaviSine					
512	0.41	0.47	0.42	0.44	0.50
1024	0.24	0.28	0.27	0.31	0.38
2048	0.15	0.16	0.17	0.22	0.29
4096	0.08	0.13	0.09	0.12	0.16
8192	0.05	0.07	0.06	0.08	0.11
Bumps					
512	1.51	1.93	1.57	5.68	8.41
1024	0.91	1.06	1.31	3.27	5.09
2048	0.54	0.58	0.74	1.96	3.12
4096	0.30	0.49	0.44	1.08	1.74
8192	0.16	0.24	0.26	0.62	1.03
Blocks					
512	1.51	1.91	1.73	4.10	4.91
1024	0.93	1.12	1.17	2.75	3.19
2048	0.60	0.68	0.80	1.80	2.17
4096	0.35	0.62	0.48	1.03	1.31
8192	0.21	0.37	0.34	0.67	0.89
Spikes					
512	0.63	0.75	0.75	1.53	2.10
1024	0.36	0.38	0.43	0.93	1.39
2048	0.21	0.20	0.24	0.52	0.87
4096	0.11	0.12	0.10	0.21	0.46
8192	0.06	0.06	0.06	0.12	0.29
Blip					
512	0.28	0.33	0.45	0.56	0.75
1024	0.15	0.18	0.30	0.37	0.53
2048	0.10	0.10	0.17	0.23	0.34
4096	0.06	0.08	0.11	0.13	0.19
8192	0.03	0.05	0.06	0.08	0.12
Corner					
512	0.19	0.20	0.22	0.24	0.39
1024	0.09	0.10	0.12	0.15	0.25
2048	0.05	0.05	0.06	0.09	0.15
4096	0.03	0.03	0.03	0.04	0.06
8192	0.02	0.02	0.02	0.03	0.03
Wave					
512	0.51	0.58	3.10	1.83	2.83
1024	0.24	0.25	3.20	1.13	1.68
2048	0.12	0.13	3.52	0.62	0.93

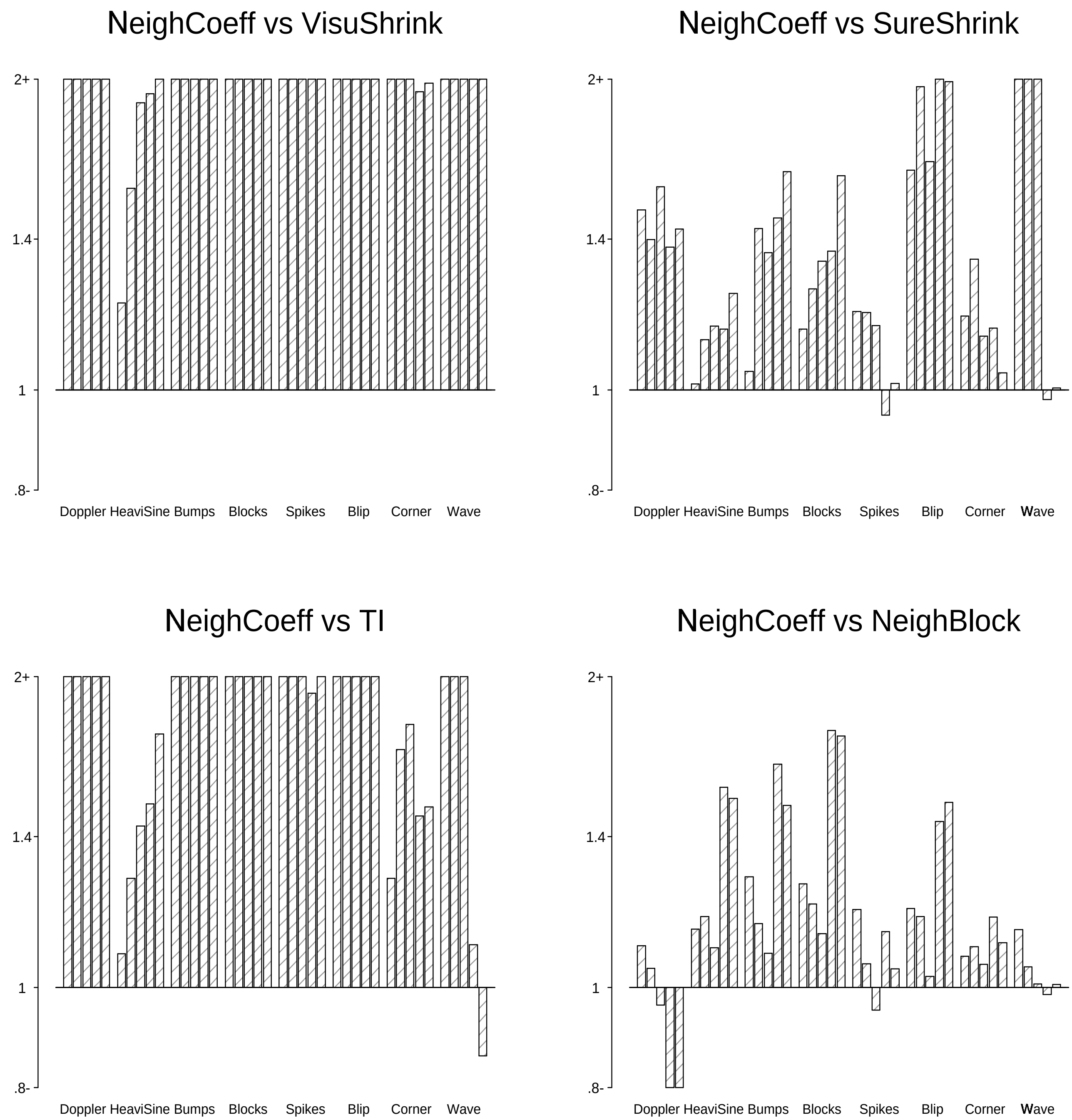

Figure 6: RSNR=6. The vertical bars represent the ratios of the MSEs of various estimators to the corresponding MSE of the NeighCoeff estimator. The higher the bar the better the relative performance of the NeighCoeff estimator. The bars are plotted on a log scale and are truncated at the value 2 of the original ratio. For each signal the bars are ordered from left to right by the sample sizes (n=512 to 8192).

Table 5: Mean Squared Error From 60 Replications (RSNR=7)

n	NeighCoeff	NeighBlock	SureShrink	TI denoising	VisuShrink
Doppler					
512	0.63	0.72	0.90	1.61	2.10
1024	0.39	0.41	0.55	1.00	1.39
2048	0.21	0.21	0.34	0.59	0.87
4096	0.12	0.09	0.18	0.32	0.49
8192	0.07	0.05	0.10	0.19	0.30
HeaviSine					
512	0.33	0.39	0.35	0.37	0.44
1024	0.20	0.22	0.22	0.27	0.32
2048	0.11	0.13	0.13	0.18	0.24
4096	0.06	0.10	0.08	0.10	0.13
8192	0.04	0.06	0.05	0.07	0.09
Bumps					
512	1.12	1.41	1.21	4.45	6.64
1024	0.66	0.78	1.01	2.54	3.98
2048	0.40	0.44	0.57	1.52	2.45
4096	0.22	0.36	0.34	0.81	1.33
8192	0.12	0.18	0.21	0.48	0.81
Blocks					
512	1.02	1.33	1.25	3.14	3.94
1024	0.66	0.79	0.81	2.10	2.54
2048	0.42	0.48	0.64	1.38	1.76
4096	0.25	0.46	0.37	0.81	1.06
8192	0.15	0.27	0.27	0.51	0.71
Spikes					
512	0.49	0.56	0.57	1.16	1.70
1024	0.28	0.29	0.34	0.68	1.09
2048	0.16	0.15	0.15	0.39	0.69
4096	0.08	0.09	0.08	0.16	0.37
8192	0.04	0.04	0.04	0.09	0.22
Blip					
512	0.20	0.24	0.36	0.43	0.60
1024	0.11	0.13	0.23	0.28	0.42
2048	0.07	0.08	0.13	0.17	0.28
4096	0.04	0.06	0.09	0.10	0.15
8192	0.02	0.03	0.05	0.06	0.10
Corner					
512	0.13	0.15	0.17	0.20	0.31
1024	0.07	0.07	0.09	0.12	0.19
2048	0.04	0.04	0.05	0.08	0.12
4096	0.02	0.03	0.03	0.03	0.05
8192	0.01	0.02	0.01	0.02	0.03
Wave					
512	0.36	0.40	3.24	1.48	2.21
1024	0.19	0.19	3.36	0.85	1.27
2048	0.10	0.10	3.56	0.47	0.69

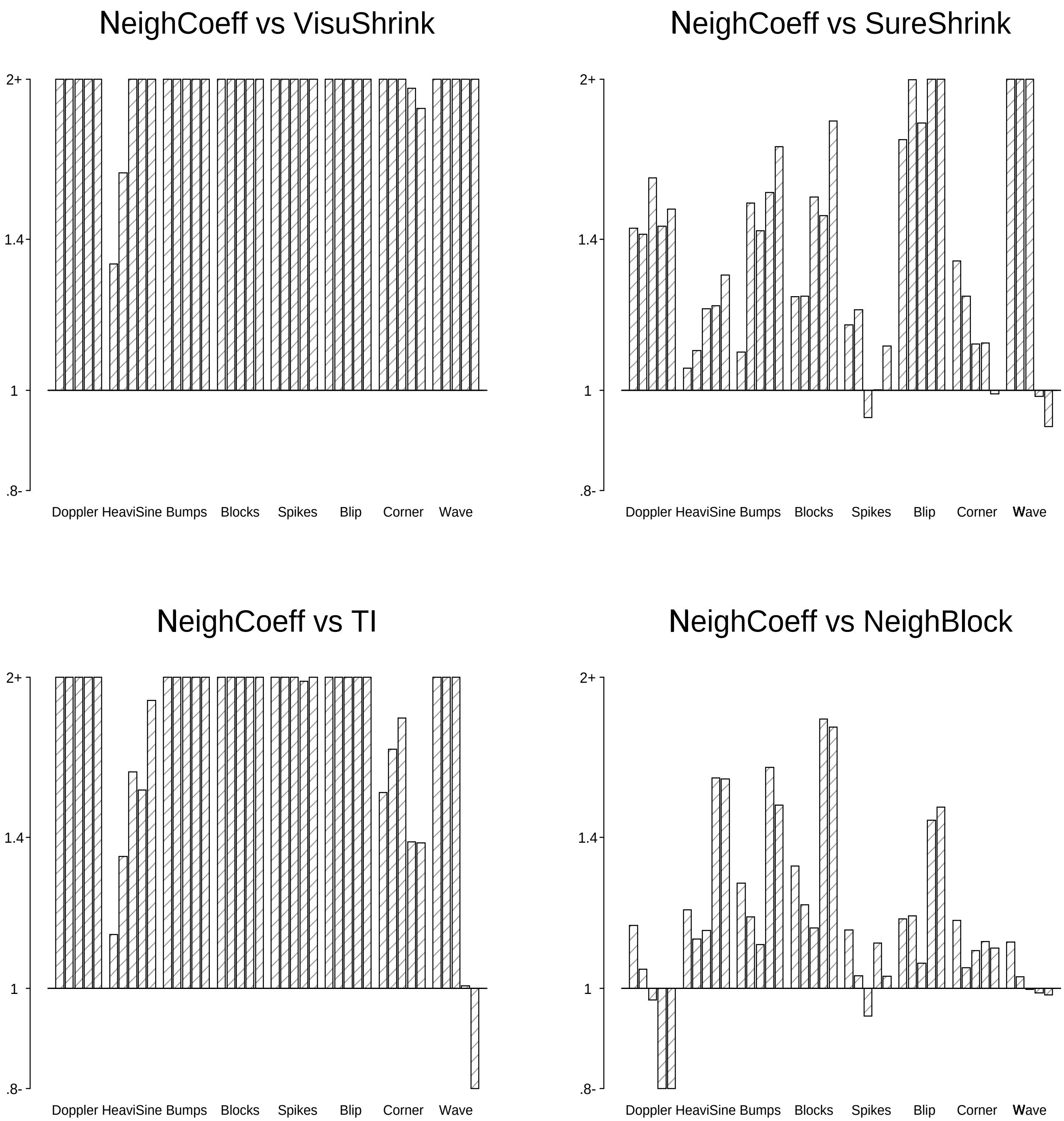

Figure 7: RSNR=7. The vertical bars represent the ratios of the MSEs of various estimators to the corresponding MSE of the NeighCoeff estimator. The higher the bar the better the relative performance of the NeighCoeff estimator. The bars are plotted on a log scale and are truncated at the value 2 of the original ratio. For each signal the bars are ordered from left to right by the sample sizes (n=512 to 8192).

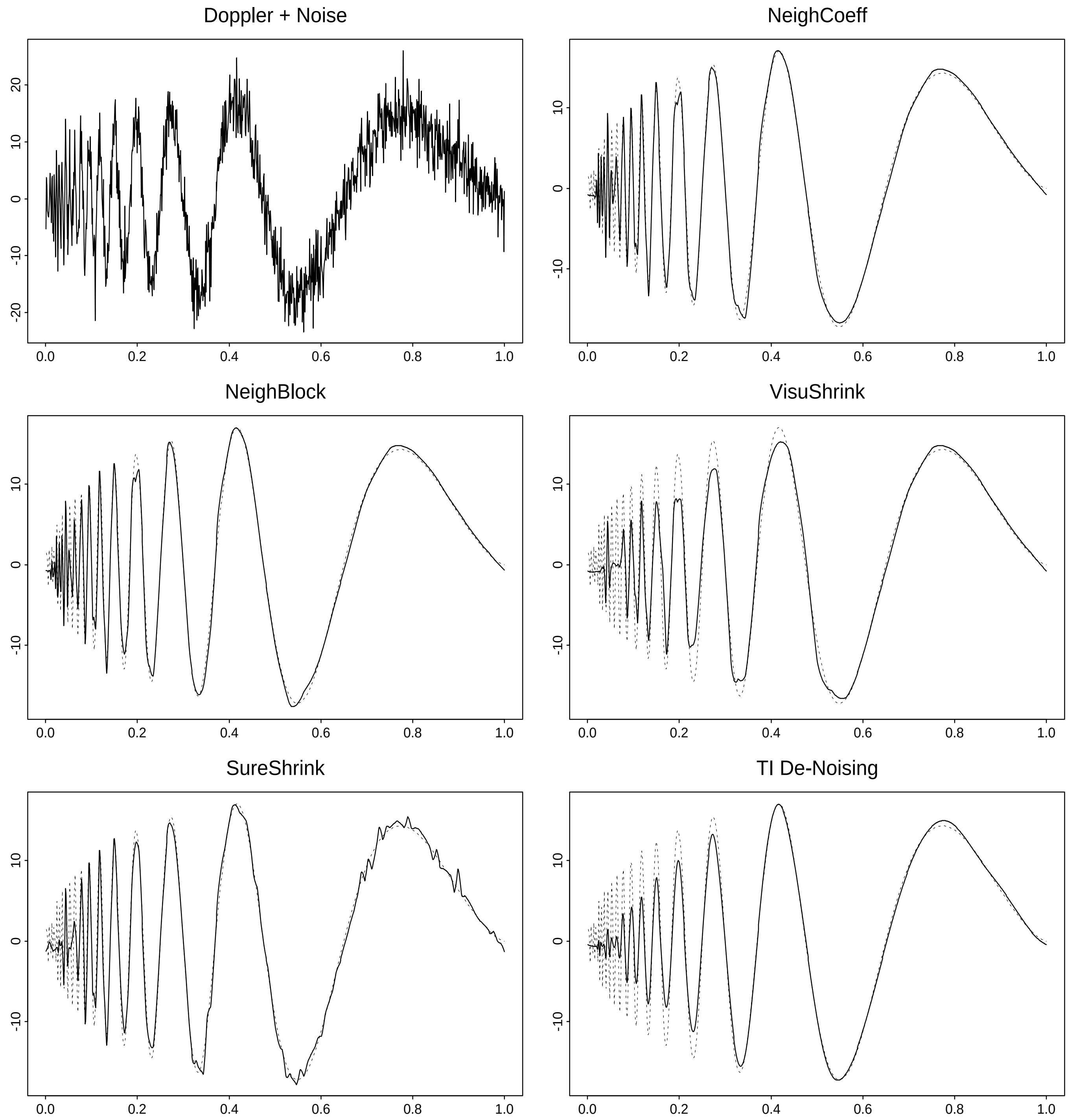

Figure 8: $RSNR = 3,\ n = 1024$, Doppler signal. Visual comparison of the reconstructions of NeighCoeff , NeighBlock, VisuShrink, SureShrink, and TI denoising methods.

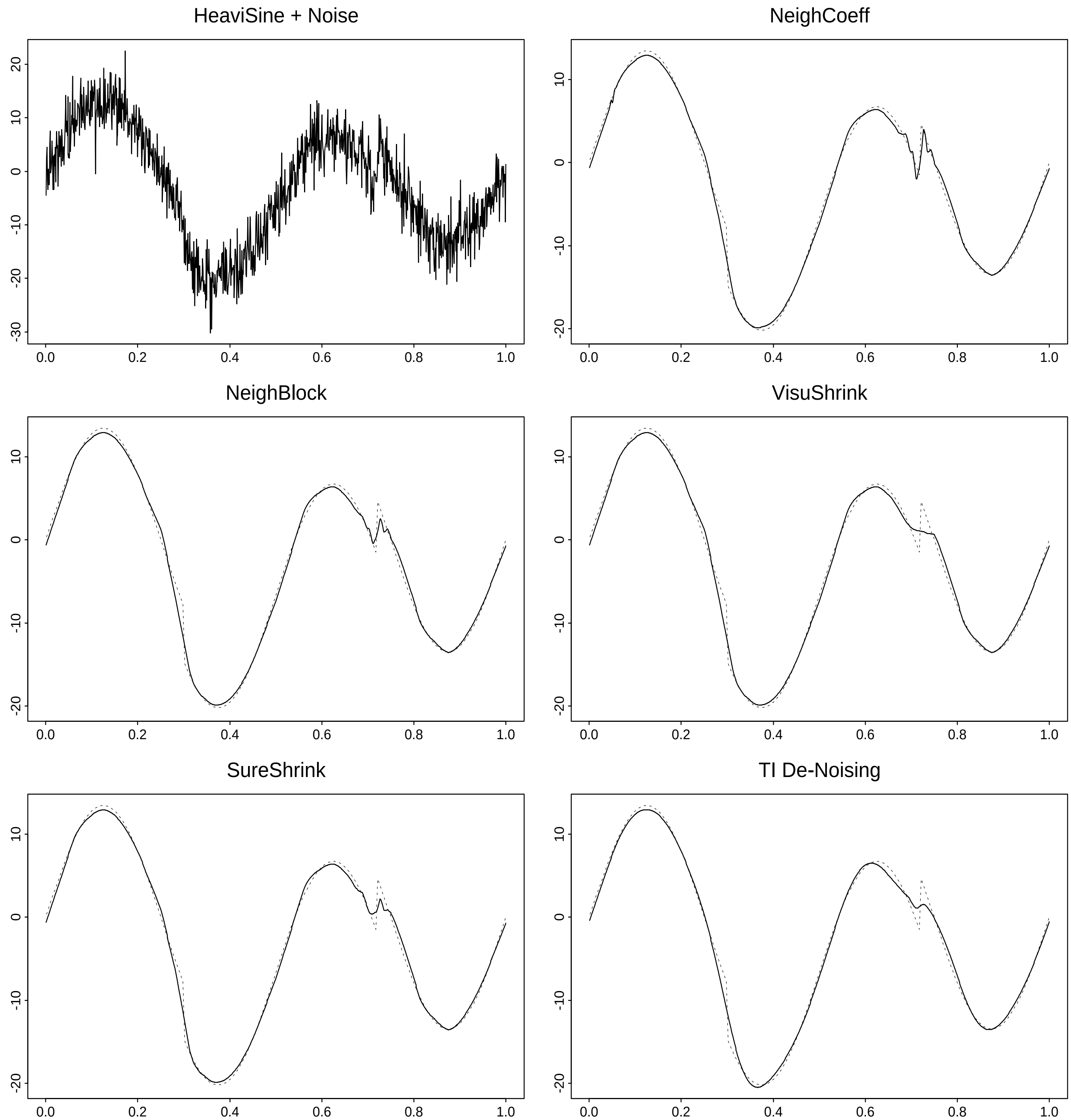

Figure 9: $RSNR = 3,\ n = 1024$, HeaviSine signal.

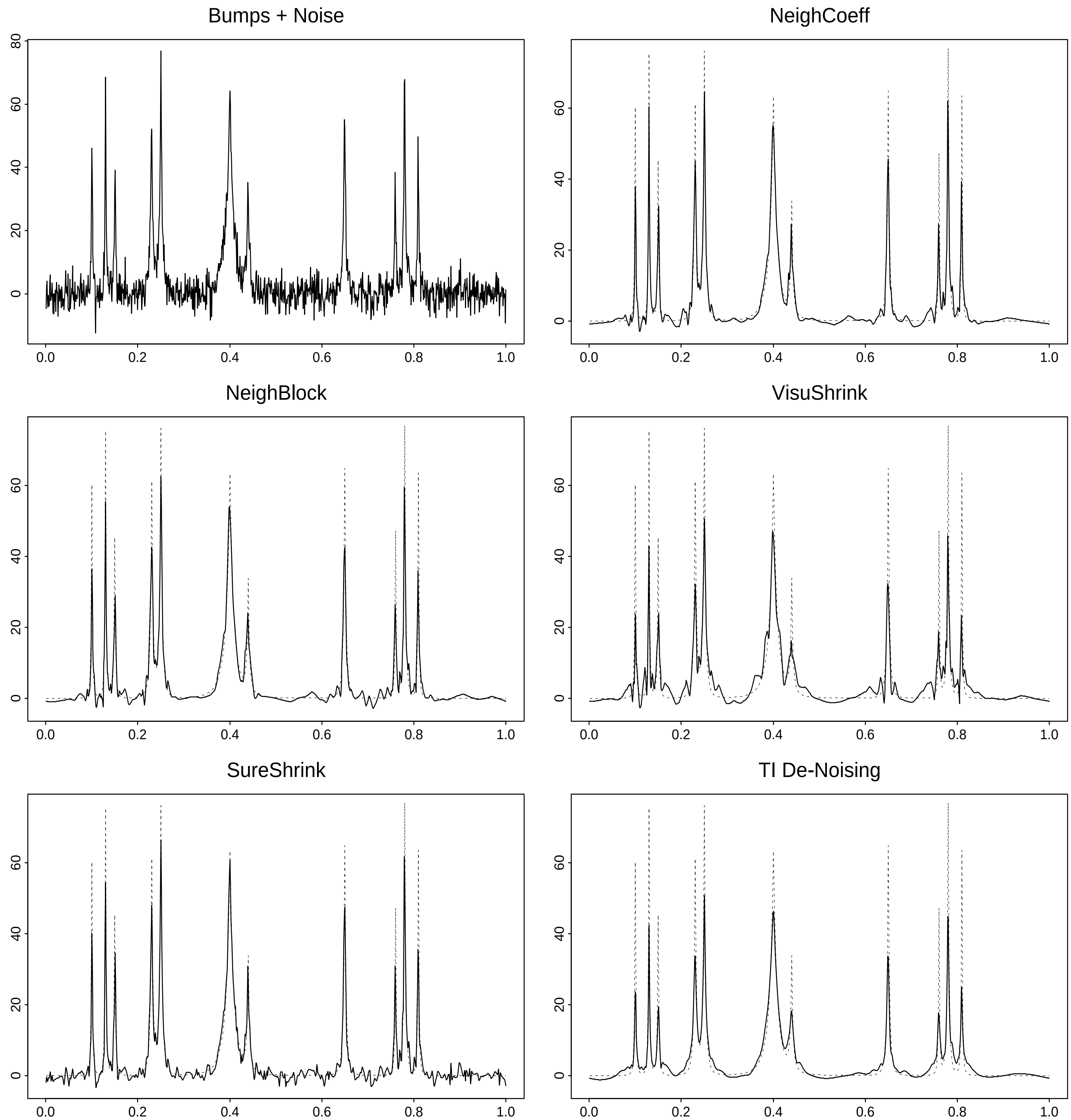

Figure 10: $RSNR = 3,\ n = 1024$, Bumps signal.

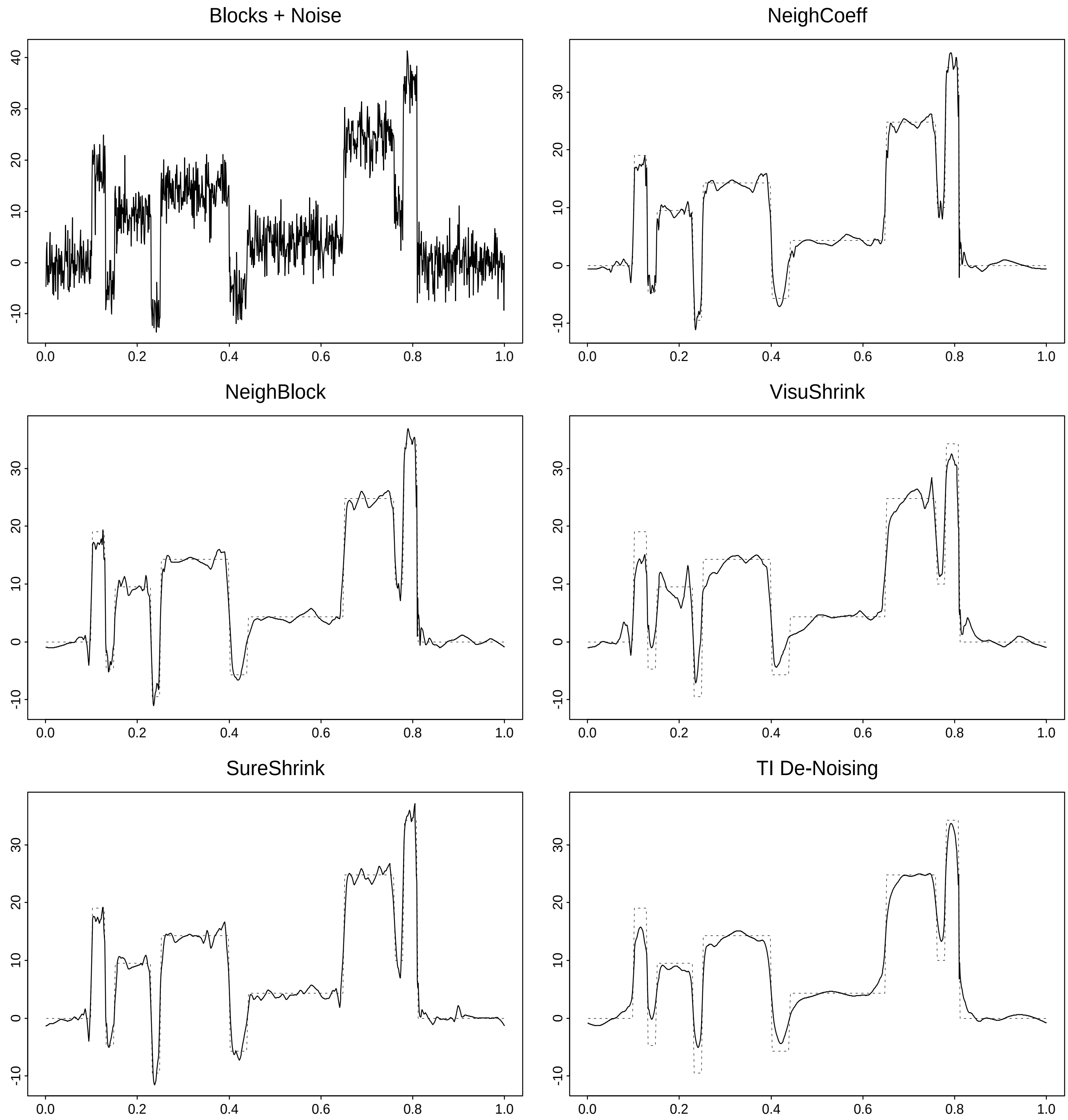

Figure 11: $RSNR = 3,\ n = 1024$, Blocks signal.

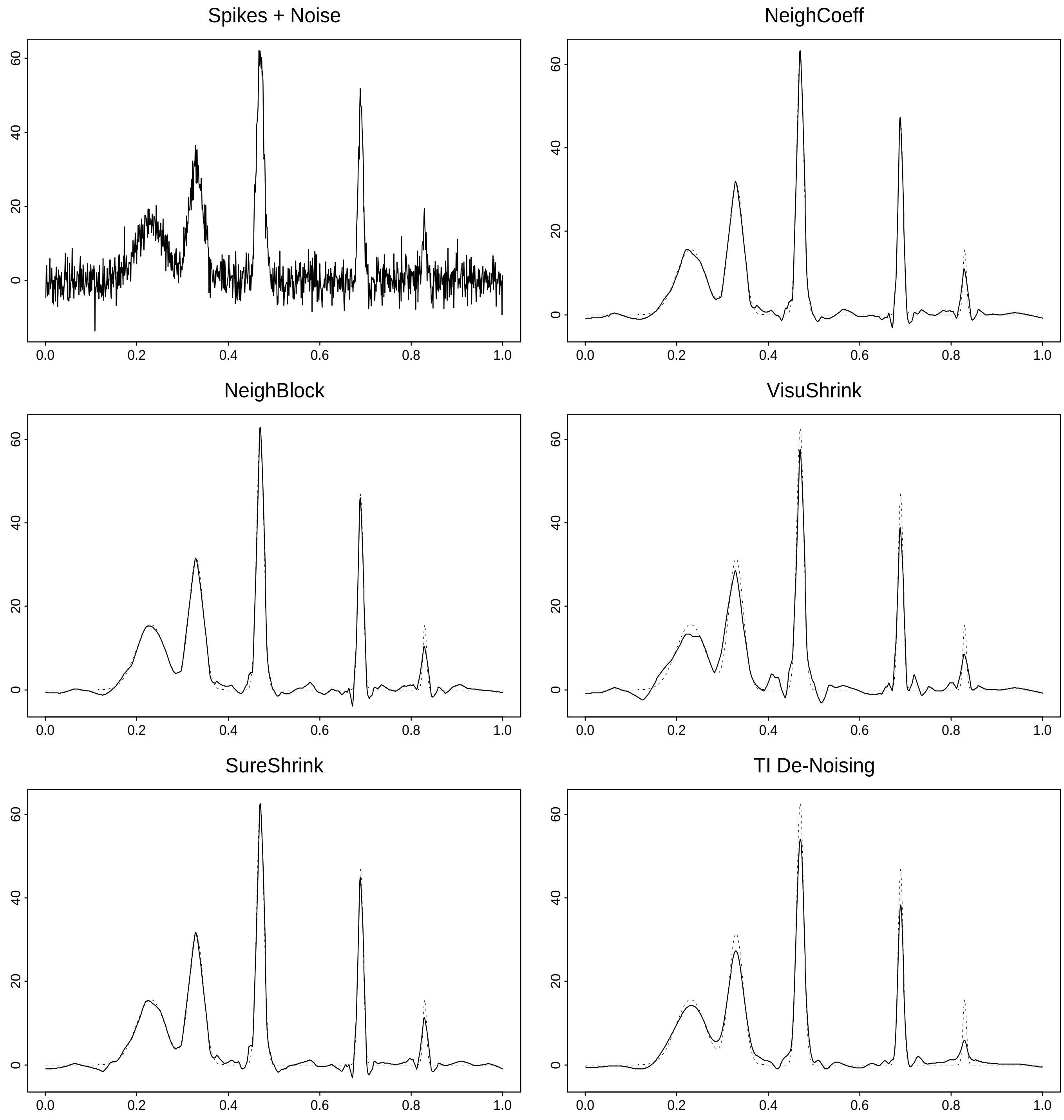

Figure 12: $RSNR = 3,\ n = 1024$, Spikes signal.

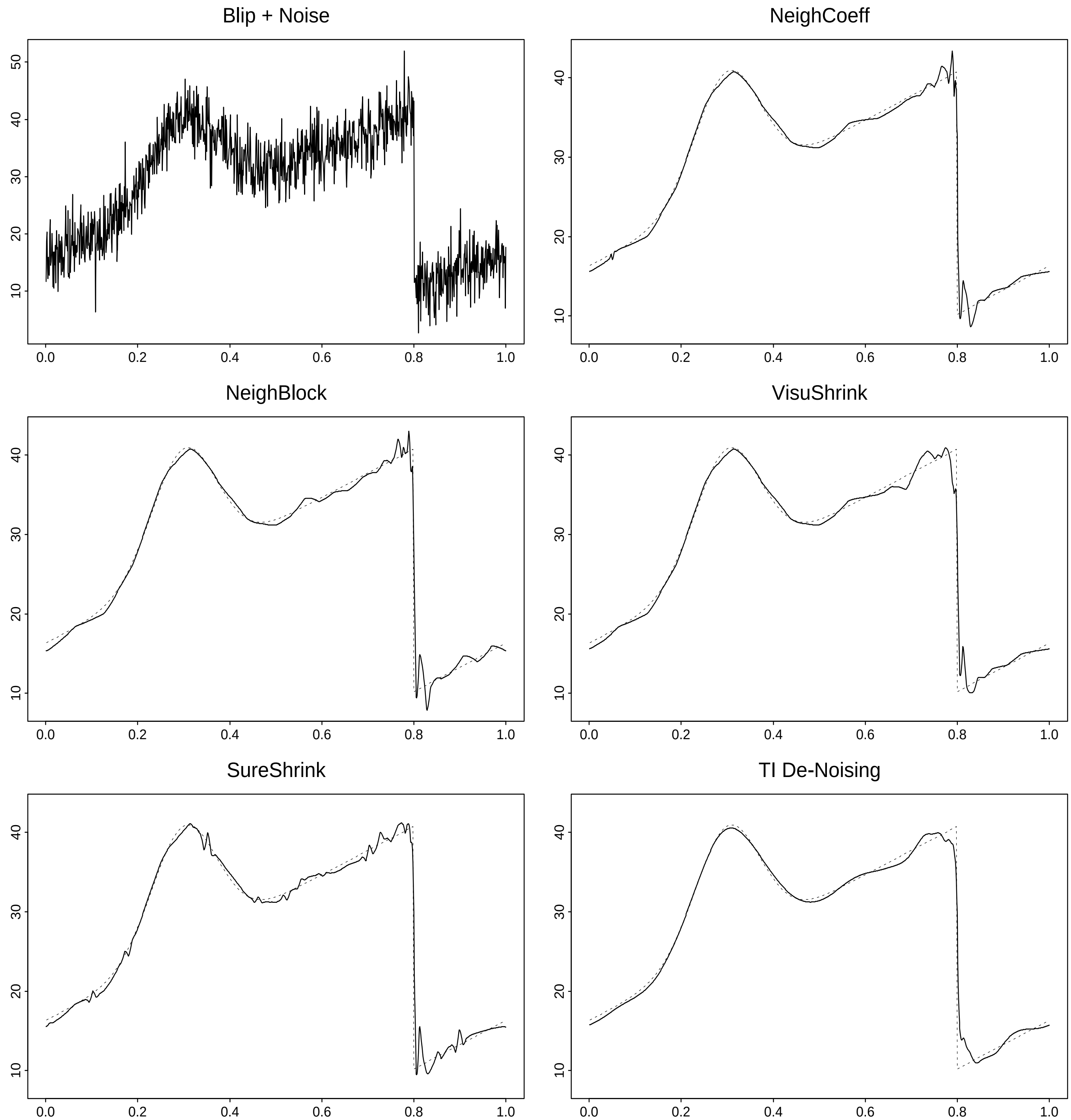

Figure 13: $RSNR = 3,\ n = 1024$, Blip signal.

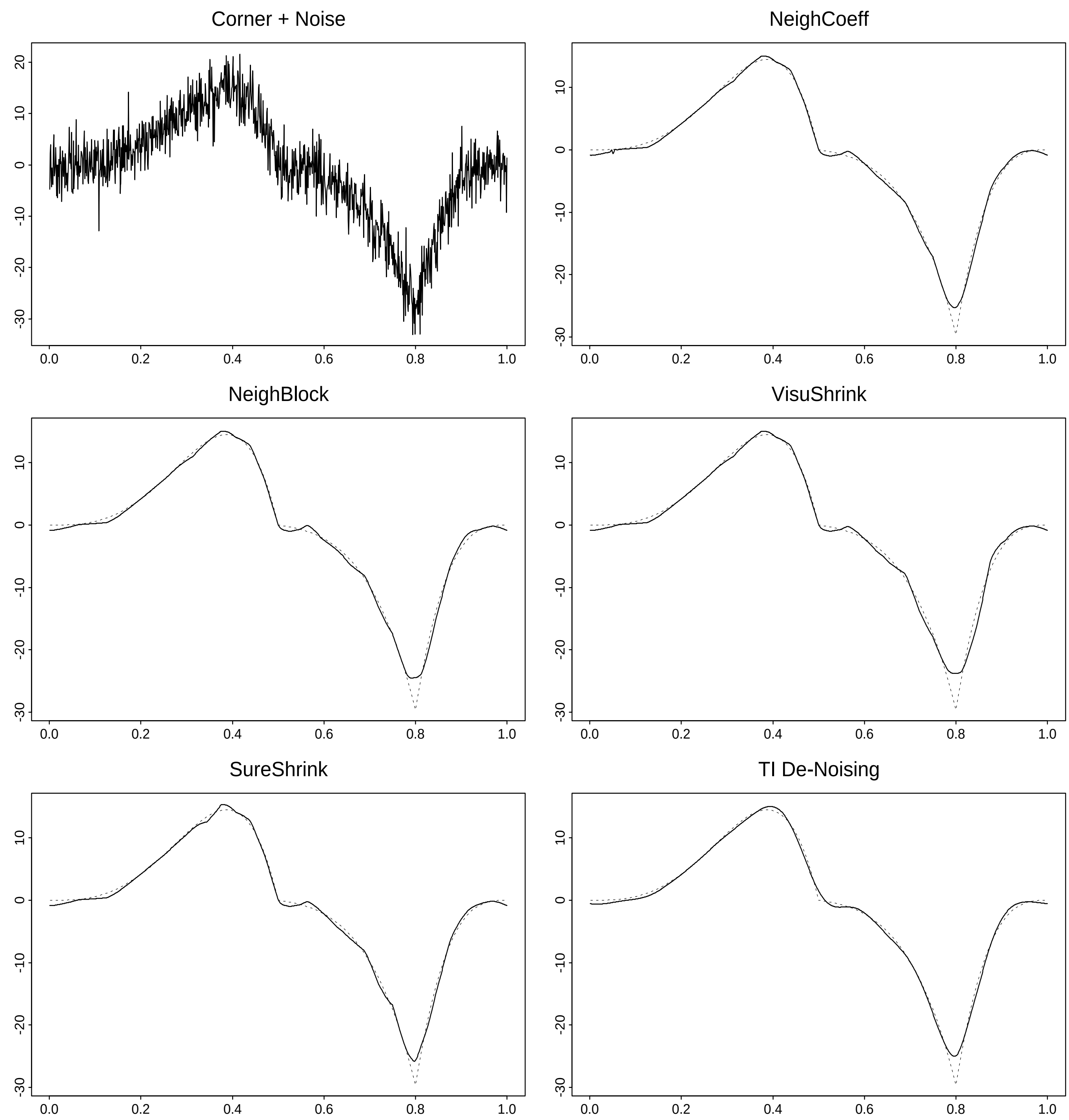

Figure 14: $RSNR = 3,\ n = 1024$, Corner signal.

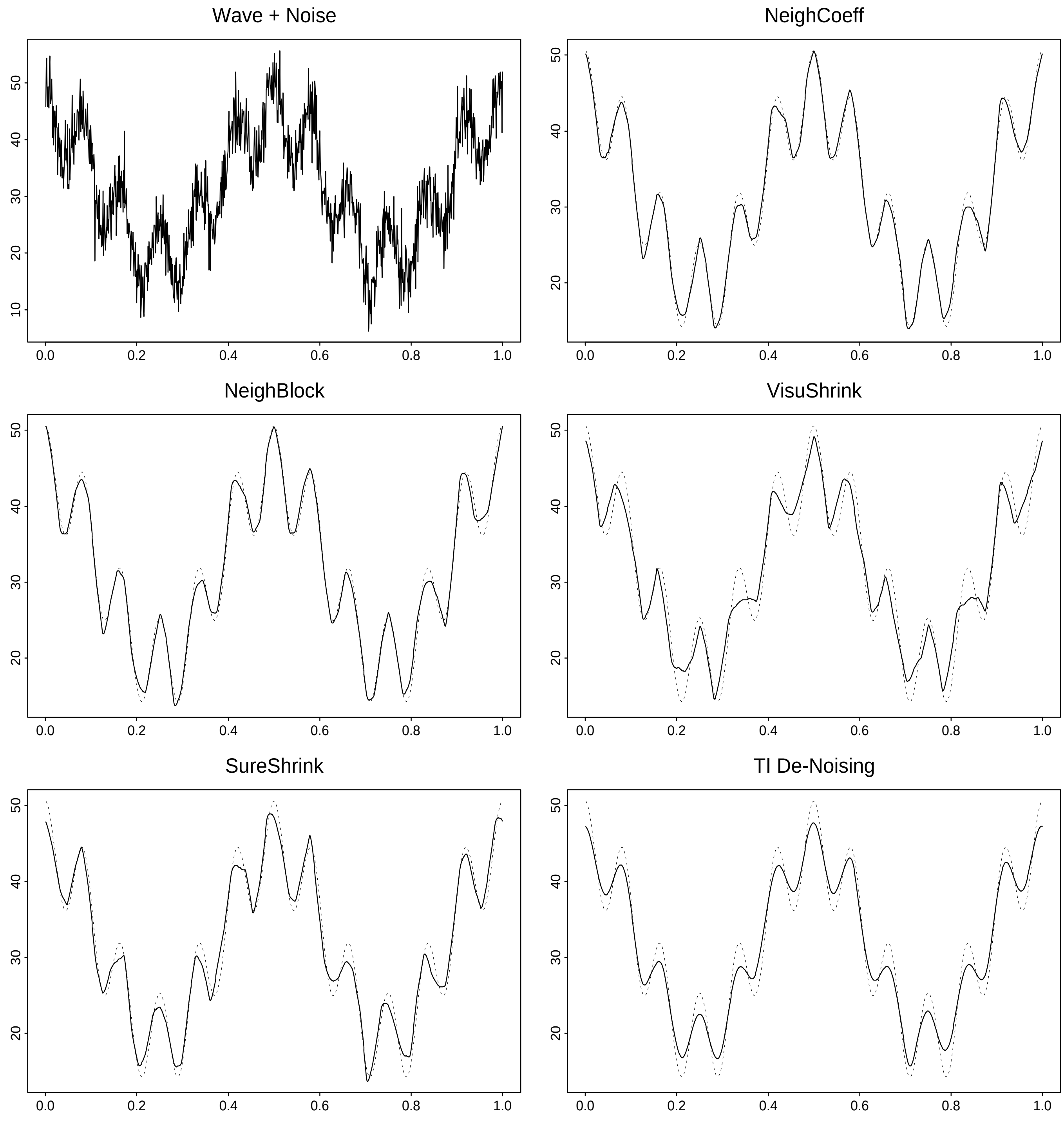

Figure 15: $RSNR = 3,\ n = 1024,$ Wave signal.

References

[1] Abramovich, F., Sapatinas, T. & Silverman, B.W. (1998). Wavelet thresholding via a Bayesian approach. *J. Roy. Stat. Soc. Ser. B*, **60**, (in press).

[2] Cai, T. (1999). Adaptive wavelet estimation: A block thresholding and oracle inequality approach. *Ann. Statist.* **27**, 898-924.

[3] Coifman, R.R. & Donoho, D.L. (1995). Translation invariant denoising. *Wavelets and Statistics*, Ed. Anestis Antoniadis, Springer-Verlag, Lecture Notes, 1995, 125-150.

[4] Cai, T. & Silverman, B.W. (1998). Incorporating Information on Neighboring Coefficients into Wavelet Estimation. Technical Report, Department of Statistics, Purdue University.

[5] Donoho, D.L. & Johnstone, I.M. (1995). Ideal spatial adaptation via wavelet shrinkage. *Biometrika*, **81**, 425-455.

[6] Donoho, D.L. & Johnstone, I.M. (1994). Adapting to unknown smoothness via wavelet shrinkage. Technical Report, Stanford University.

[7] Hall, P., Penev, S., Kerkyacharian, G. & Picard, D. (1996). Numerical performance of block thresholded wavelet estimators. Manuscript.

[8] Marron, J.S., Adak, S., Johnstone, I.M., Neumann, M.H. and Patil, P. (1995). Exact risk analysis of wavelet regression. Technical Report, Stanford University.