Supplementary Notes for "Adversarial Orthogonal Regression: Two non-Linear Regressions for Causal Inference"

Mohammad Reza Heydari[1], Saber Salehkaleybar[1] and Kun Zhang [2]
[1] *Department of Electrical Engineering, Sharif University of Tschnology, Tehran, Iran*
[2] *Department of Philosophy, Carnegie Mellon University, Pittsburgh, United States*
The Journal of Neural Networks
(Dated: October, 2020)

All the codes, figures, and data are available online on this github link.

TOY EXAMPLES: COMPARISON BETWEEN REGRESSION METHODS

In this part, we compare AdOR/AdOSE with eight regression methods as described in part **5.2. Toy Examples** of the paper. Again, the model has a form of $Y = f(X) + N = f(X) + kN^{'}$; in which, 300 samples drawn from $X \sim U(-1, 1)$ and $N^{'}$ has different distributions. Parameters of each distribution are chosen randomly for each trial. The constant k is chosen to have a similar signal-to-noise ratio $SNR = \mathbb{E}\left[f(X)^2\right]/\mathbb{E}\left[N^2\right] = 0.5$ for each trial. Here we compare our methods with competitors on different choices of non-linear function $f(X)$: $f(X) = X^2$ (the same as Figure *3* of paper), $f(X) = 2\sin(\pi X)$, $f(X) = e^{2X}$, and $f(X) = sigmoid(5X)$. We did not use any normalization method for AdOR and AdOSE training. R network in both AdOR and AdOSE is constructed by three hidden layers; layer 1 with 6 units and *tanh* activation, layer 2 with 15 units and *sigmoid* activation, and layer 3 with 10 units and $leaky - ReLU$ activation. MI and KL networks have the structure of three layers with 30 units in each layer and $leaky - ReLU$ activation function.

$$f(X) = X^2$$

The predicted functions of each method are shown in Figure 1 for the case of $N \sim F(8.5, 12.2)$. ISEs and MSEs for different distributions are shown in Figure 2. Table I shows the properties and distributions of noise N for each trial.

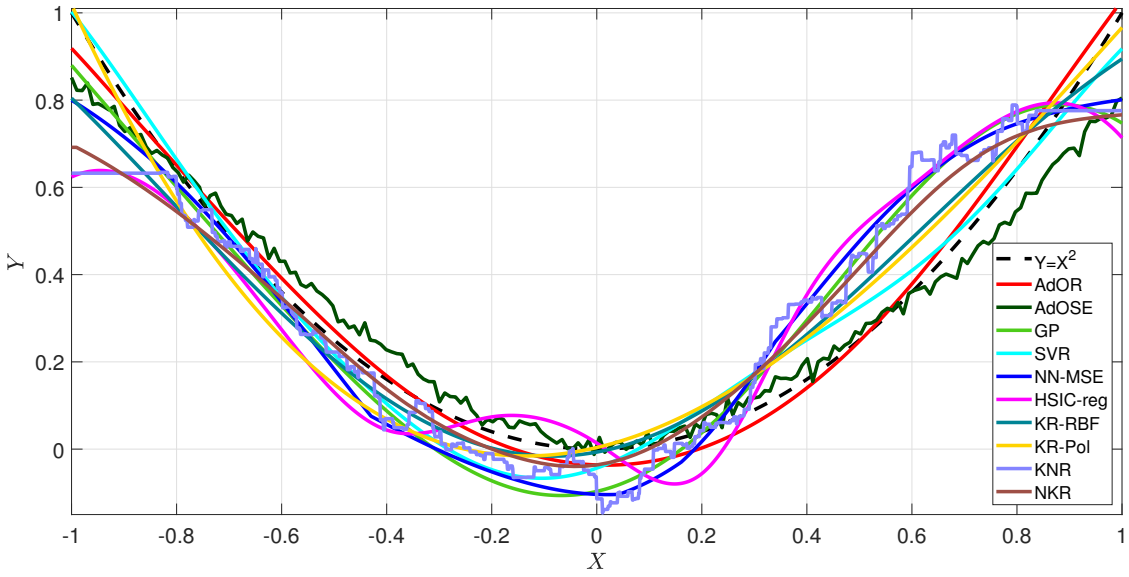


FIG. 1: Example of different methods' outputs. $Y = X^2 + N$ and $N \sim F(8.5, 12.2)$

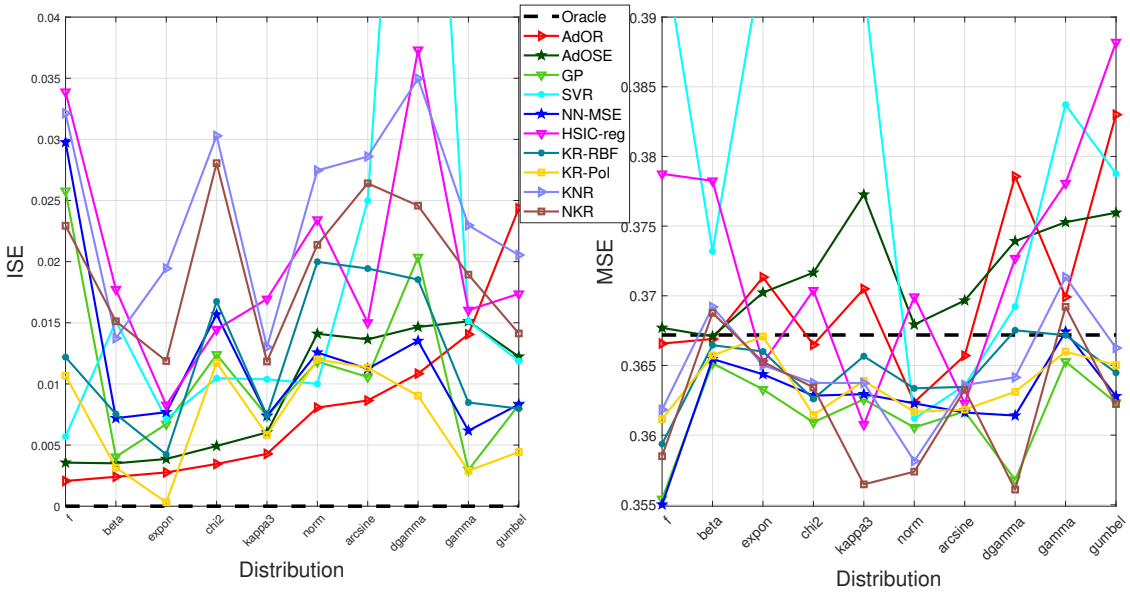
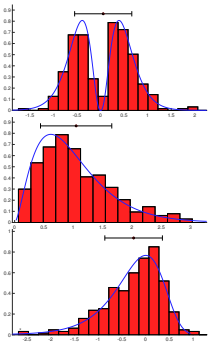


FIG. 2: ISE (left) and MSE (right) of methods for the model $Y = X^2 + N$ for different distributions of noise N.

TABLE I: Properties of the noise for $f(X) = X^2$. Figures show the PDF (PMF) of distributions and histograms of samples $N = kN^{'}$ with mean and std as horizontal lines. ($\mathbf{B}(.,.)$: Beta function, $\Gamma(.)$: Gamma function)

NAME	PDF(PMF) of $N^{'}$	PARAMETERS	HISTOGRAM of $N = kN^{'}$
f	$\frac{1}{\mathbf{B}\left(\frac{d_1}{2},\frac{d_2}{2}\right)}\left(\frac{d_1}{d_2}\right)^{\frac{d_1}{2}} x^{\frac{d_1}{2}-1}\left(1+\frac{d_1}{d_2}x\right)^{-\frac{d_1+d_2}{2}}$	$d_1 = 8.46, d_2 = 12.24$	
beta	$x^{\alpha-1}(1-x)^{\beta-1}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}$	$\alpha = 1.28, \beta = 1.92$	
expon	e^{-x}	–	
chi2	$\frac{1}{2^{k/2}\Gamma(k/2)}x^{k/2-1}e^{-x/2}$	$k = 1.66$	
kappa3	$a\,(a+x^a)^{-(a+1)/a}$	$a = 3.02$	
norm	$\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}x^2}$	–	
arcsine	$\frac{1}{\pi\sqrt{x(1-x)}}$	–	

dgamma	$\frac{\lvert x\rvert^{(\gamma-1)}e^{-\lvert x\rvert}}{2\Gamma(\gamma)}$	$\gamma = 3.65$	
gamma	$\frac{1}{\Gamma(k)}x^{k-1}e^{-x}$	$k = 2.65$	
gumbel	$e^{-\left(x+e^{-x}\right)}$	–	

$$f(X) = 2\sin(\pi X)$$

The predicted functions of each method are shown in Figure 3 for the case of $N \sim chi2(3.4)$. ISEs and MSEs for different distributions are shown in Figure 4. Table II shows the properties and distributions of noise N for each trial.

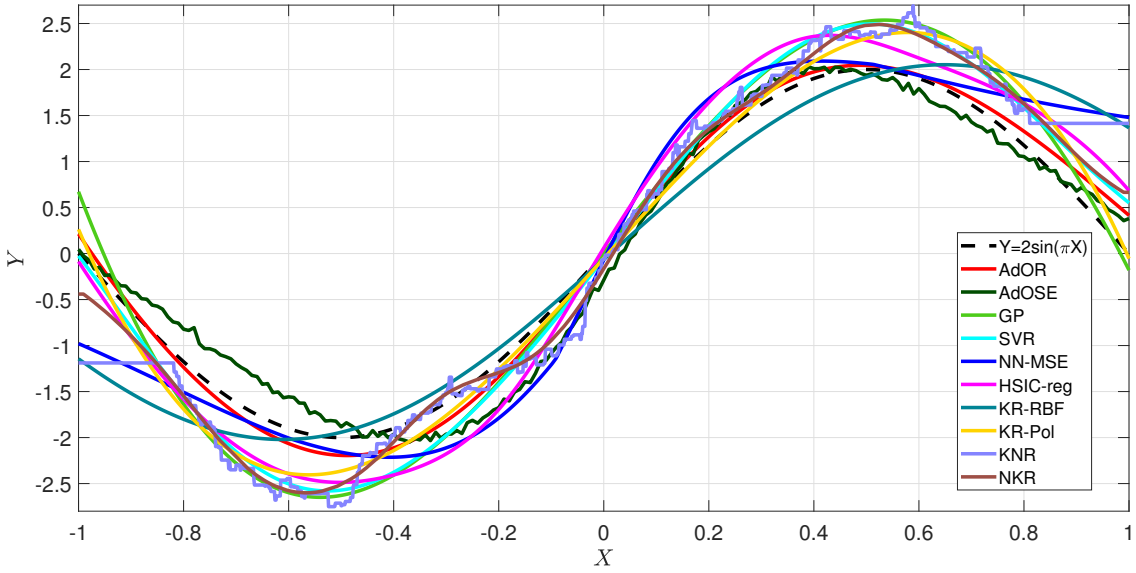


FIG. 3: Example of different methods' outputs. $Y = 2\sin(\pi X) + N$ and $N \sim chi2(3.4)$

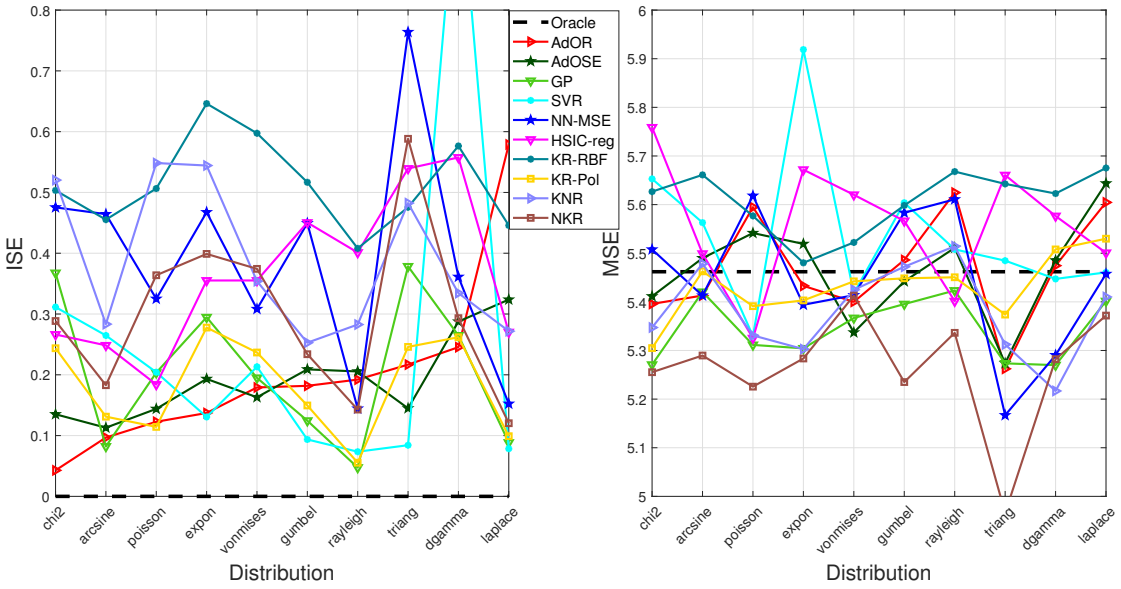
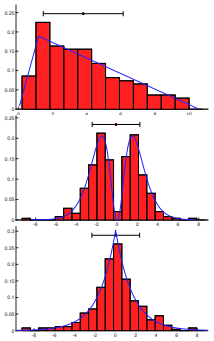


FIG. 4: ISE (left) and MSE (right) of methods for the model $Y = 2\sin(\pi X) + N$ for different distributions of noise N.

TABLE II: Properties of the noise for $f(X) = 2\sin(\pi X)$. Figures show the PDF (PMF) of distributions and histograms of samples $N = kN^{'}$ with mean and std as horizontal lines. ($\Gamma(.)$: Gamma function, $\mathbf{I}_0(.)$: modified Bessel function of order zero)

NAME	PDF(PMF) of $N^{'}$	PARAMETERS	HISTOGRAM of $N = kN^{'}$
chi2	$\frac{1}{2^{k/2}\Gamma(k/2)} x^{k/2-1} e^{-x/2}$	$k = 3.36$	
arcsine	$\frac{1}{\pi\sqrt{x(1-x)}}$	–	
poisson	$\frac{\lambda^k e^{-\lambda}}{k!}, \quad k = 1, 2, \ldots$	$\lambda = 8.78$	
expon	e^{-x}	–	
vonmises	$\frac{\kappa \cos(x)}{2\pi \mathbf{I}_0(\kappa)}$	$\kappa = 1.05$	
gumbel	$e^{-\left(x+e^{-x}\right)}$	–	
rayleigh	$xe^{\frac{-x^2}{2}}$	–	

triang	$\frac{c-\lvert x\rvert}{c^2}, \quad \lvert x\rvert \leq c$	$c = 0.11$	
dgamma	$\frac{\lvert x\rvert^{(\gamma-1)} e^{-\lvert x\rvert}}{2\Gamma(\gamma)}$	$\gamma = 3.52$	
laplace	$\frac{1}{2} e^{-\lvert x\rvert}$	–	

$$f(X) = e^{2X}$$

The predicted functions of each method are shown in Figure 5 for the case of $N \sim expon$. ISEs and MSEs for different distributions are shown in Figure 6. Table III shows the properties and distributions of noise N for each trial.

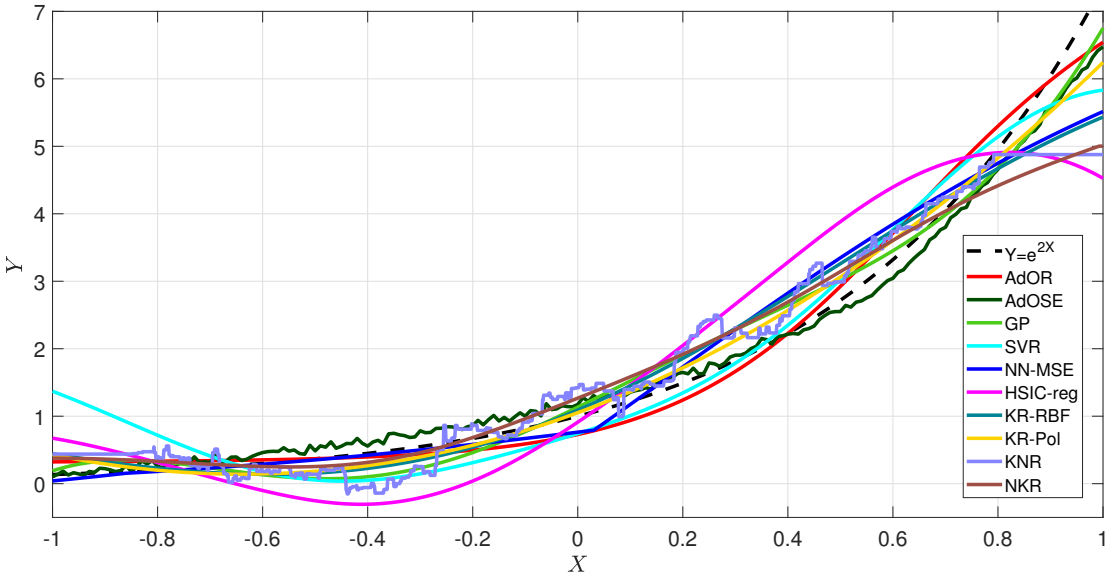


FIG. 5: Example of different methods' outputs. $Y = e^{2X} + N$ and $N \sim expon$

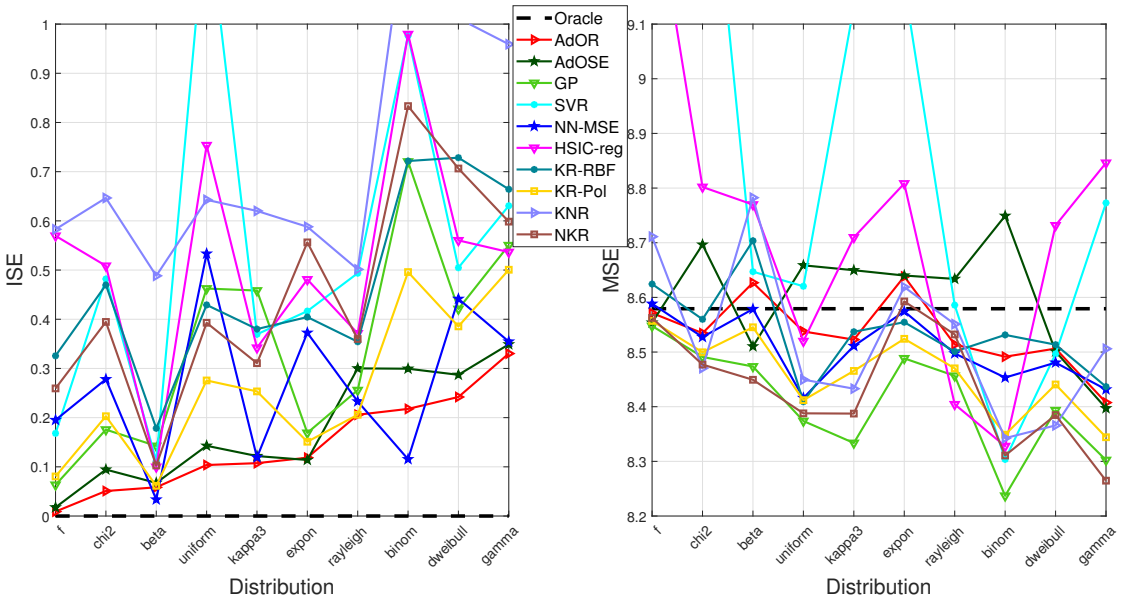
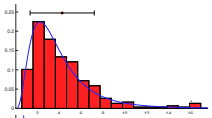
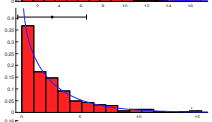
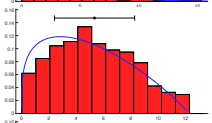
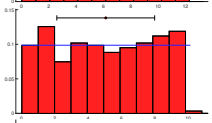
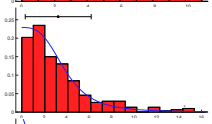
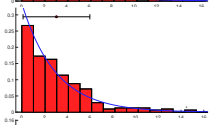
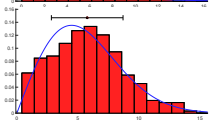
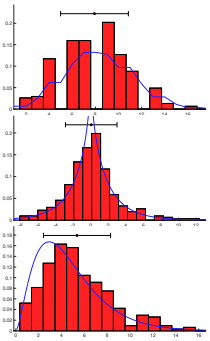


FIG. 6: ISE (left) and MSE (right) of methods for the model $Y = e^{2X} + N$ for different distributions of noise N.

TABLE III: Properties of the noise for $f(X) = e^{2X}$. Figures show the PDF (PMF) of distributions and histograms of samples $N = kN^{'}$ with mean and std as horizontal lines. ($\mathbf{B}(.,.)$: Beta function, $\Gamma(.)$: Gamma function)

NAME	PDF(PMF) of $N^{'}$	PARAMETERS	HISTOGRAM of $N = kN^{'}$
f	$\frac{1}{\mathbf{B}\left(\frac{d_1}{2},\frac{d_2}{2}\right)}\left(\frac{d_1}{d_2}\right)^{\frac{d_1}{2}} x^{\frac{d_1}{2}-1}\left(1+\frac{d_1}{d_2}x\right)^{-\frac{d_1+d_2}{2}}$	$d_1 = 8.46, d_2 = 12.74$	
chi2	$\frac{1}{2^{k/2}\Gamma(k/2)} x^{k/2-1} e^{-x/2}$	$k = 1.66$	
beta	$x^{\alpha-1}(1-x)^{\beta-1}\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}$	$\alpha = 1.28, \beta = 1.92$	
uniform	$\frac{1}{b-a}, \quad a \le x \le b$	$a = 0.02, b = 9.84$	
kappa3	$a\,(a + x^a)^{-(a+1)/a}$	$a = 3.02$	
expon	e^{-x}	–	
rayleigh	$xe^{\frac{-x^2}{2}}$	–	

binom	$\binom{N}{k} p^k (1-p)^{N-k}, \quad k = 1, 2, \ldots, N$	$N = 17, p = 0.33$	
dweibull	$\frac{c}{2} \lvert x \rvert^{c-1} e^{-\lvert x \rvert^c}$	$c = 0.92$	
gamma	$\frac{1}{\Gamma(k)} x^{k-1} e^{-x}$	$k = 2.65$	

$$f(X) = sigmoid(5X)$$

The predicted functions of each method are shown in Figure 7 for the case of $N \sim uniform(0., 2.7)$. ISEs and MSEs for different distributions are shown in Figure 8. Table IV shows the properties and distributions of noise N for each trial.

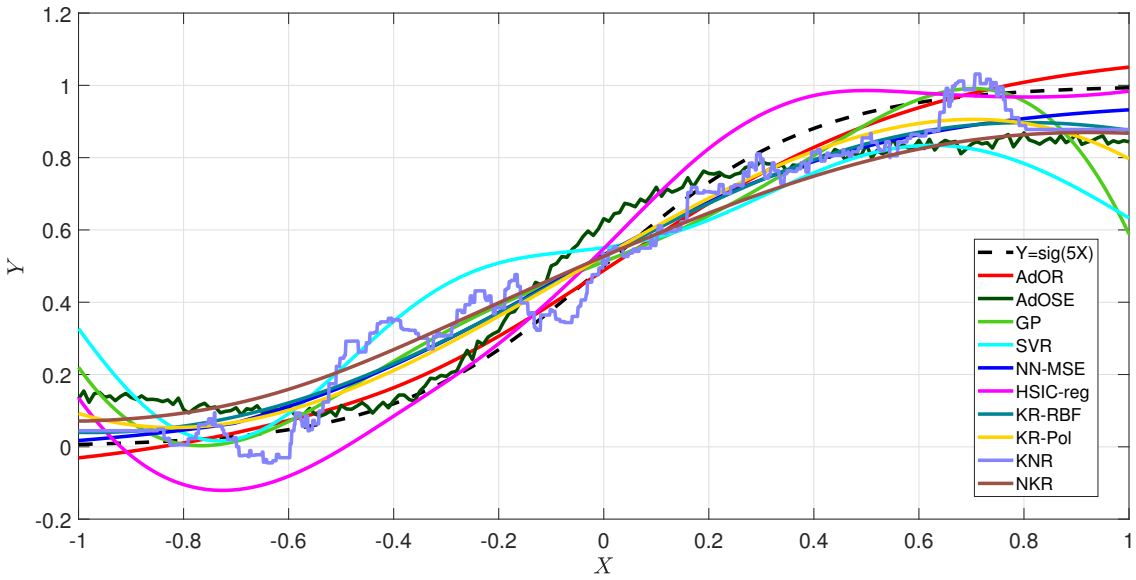


FIG. 7: Example of different methods' outputs. $Y = sigmoid(5X) + N$ and $N \sim uniform(0., 2.7)$

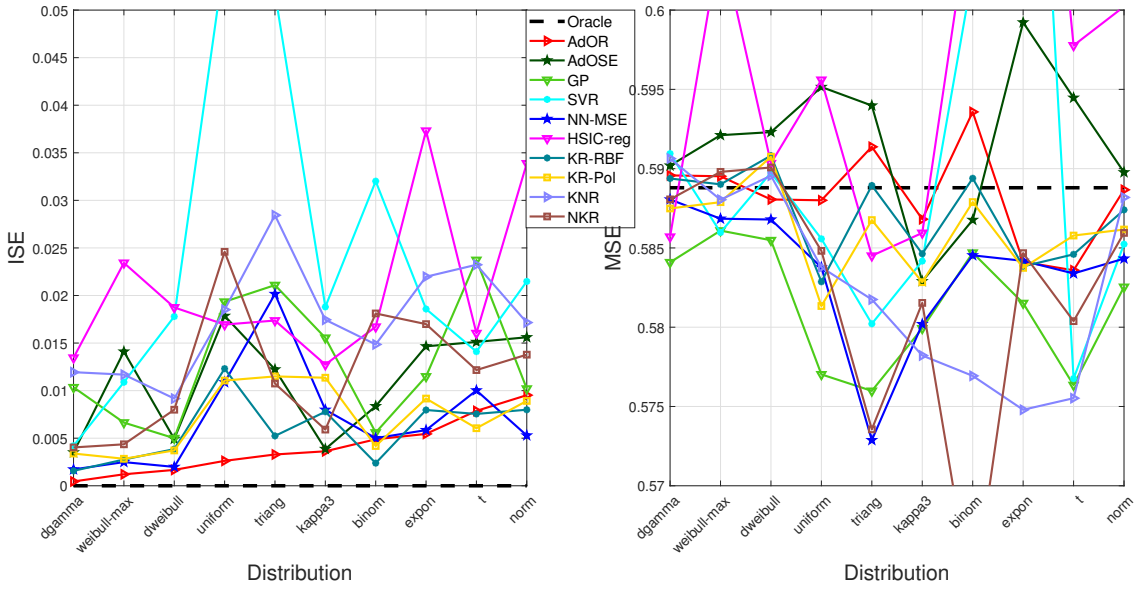
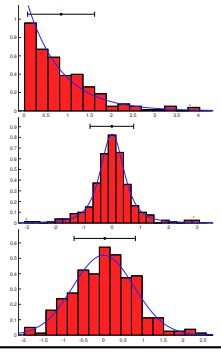


FIG. 8: ISE (left) and MSE (right) of methods for the model $Y = sigmoid(5X) + N$ for different distributions of noise N.

TABLE IV: Properties of the noise for $f(X) = sigmoid(5X)$. Figures show the PDF (PMF) of distributions and histograms of samples $N = kN^{'}$ with mean and std as horizontal lines. ($\Gamma(.)$: Gamma function)

NAME	PDF(PMF) of $N^{'}$	PARAMETERS	HISTOGRAM of $N = kN^{'}$
dgamma	$\frac{\lvert x\rvert^{(\gamma-1)}e^{-\lvert x\rvert}}{2\Gamma(\gamma)}$	$\gamma = 1.21$	
weibull-max	$c(-x)^{c-1}e^{-(-x)^c}, \quad x \leq 0$	$c = 4.56$	
dweibull	$\frac{c}{2}\lvert x\rvert^{c-1}e^{-\lvert x\rvert^c}$	$c = 1.21$	
uniform	$\frac{1}{b-a}, \quad a \leq x \leq b$	$a = 0.01, b = 2.68$	
triang	$\frac{c-\lvert x\rvert}{c^2}, \quad \lvert x\rvert \leq c$	$c = 0.27$	
kappa3	$a\,(a + x^a)^{-(a+1)/a}$	$a = 5.56$	
binom	$\binom{N}{k}p^k\,(1-p)^{N-k}, \quad k = 1, 2, \ldots, N$	$N = 20, p = 0.72$	

expon	e^{-x}	–	
t	$\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sqrt{\nu\pi}\Gamma\left(\frac{\nu}{2}\right)}\left(1+\frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}}$	$\nu = 2.81$	
norm	$\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}x^2}$	–	

CAUSAL DIRECTION DISCOVERY

In this section, the results of applying AdOR and AdOSE on causal direction discovery task are discussed, similar to the part **5.3. Causal Direction Discovery in Benchmark Datasets** of the paper. In order to finding the true direction, as discussed in the paper, mutual information of the residual and regressors for both directions $X \to Y$ and $Y \to X$ are computed. Then, the direction with lower mutual information is determined as the true direction. Here we are going to declare the procedure of training, and show more results about them.

First, both columns X and Y are normalized with sample mean and std: $X_{norm} = \frac{X - mean(X)}{std(X)}$, $Y_{norm} = \frac{Y - mean(Y)}{std(Y)}$. Only for AdOSE, we removed samples that are less than 5% or more than 95% quantiles. We found that this pre-processing step helps greatly AdOSE not to diverge in training phase. Similar to toy examples, R network in both AdOR and AdOSE is constructed by three hidden layers; layer 1 with 6 units and *tanh* activation, layer 2 with 15 units and *sigmoid* activation, and layer 3 with 10 units and $leaky - ReLU$ activation. MI and KL networks have the structure of three layers with 30 units in each layer and $leaky - ReLU$ activation function. Furthermore, we used 20 models obtained at last iterations in order to have a better function approximation, which is an ensemble strategy.

Note: In all pairs, X is cause and Y is effect.

Tuebingen Dataset

We attached the predicted function of AdOR/AdOSE for each pair in both directions $X \to Y$ and $Y \to X$ in supplementary folder `AdOR_Results_on_Tuebingen_Dataset` and `AdOSE_Results_on_Tuebingen_Dataset` with estimated mutual information. Please refer to this directory. (github link)

SIM Datasets

[1] proposes four datasets with following description:

> "We considered four different scenarios. *SIM* is the default scenario without confounders. *SIM-c* includes a one-dimensional confounder, whose influences on X and Y are typically equally strong as the influence of X on Y . The setting *SIM-ln* has low noise levels, and we would expect IGCI to work well in this scenario. Finally, *SIM-G* has approximate Gaussian distributions for the cause X and approximately additive Gaussian noise (on top of a nonlinear relationship between cause and effect)." [1]

Rates of true direction discovery and ROC curves are reported in the Figure *5* and *6* of the paper, respectively. In this part, we report scatter plots of estimated mutual information in both directions for each pair in Figure 9. Also, we show examples of predicted functions in both directions for some pairs in Figure 10 up to Figure 13.

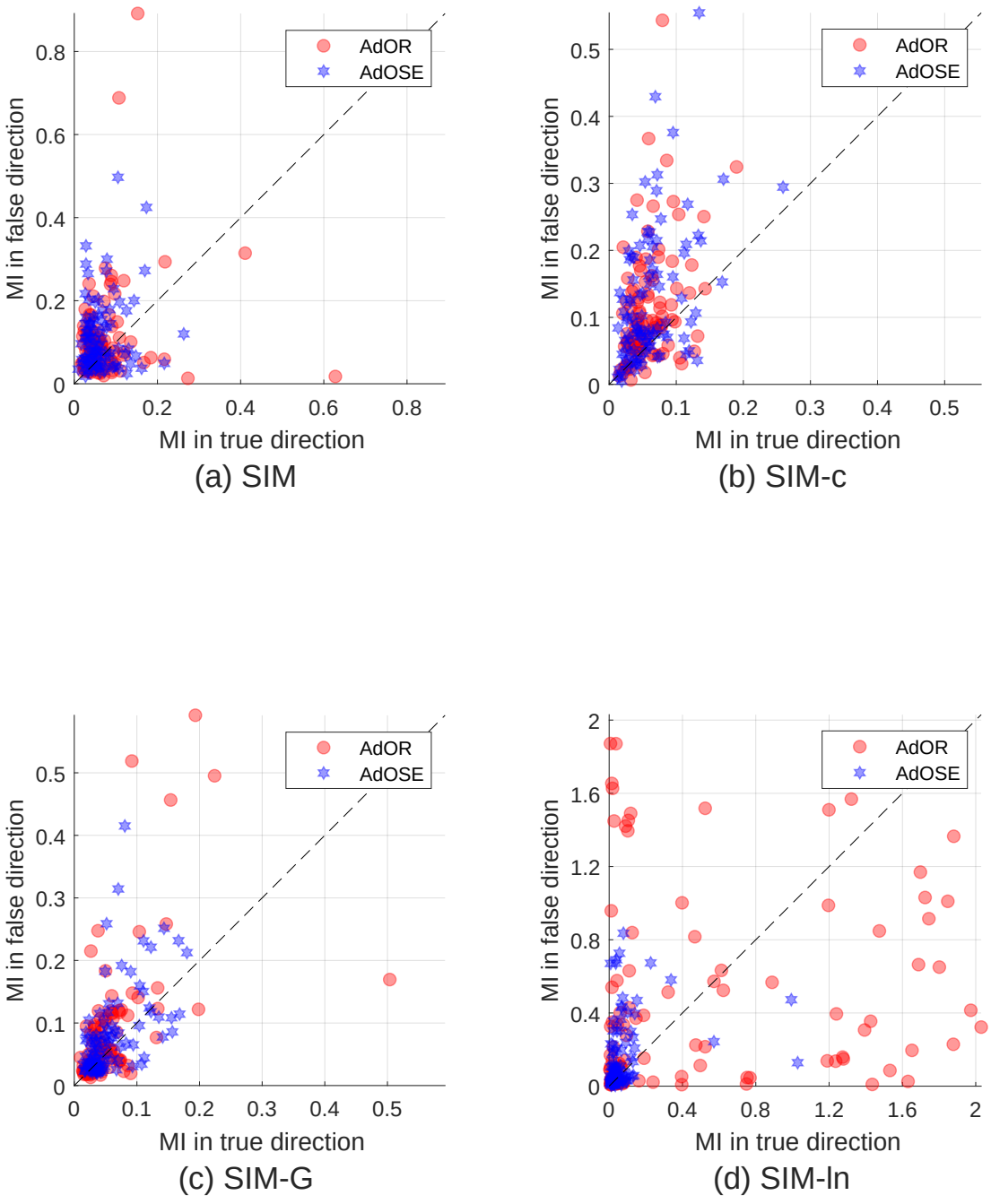


FIG. 9: Scatter plot of estimated mutual information for *SIM* datasets. Each point belongs to the each pair. Points upper than the dashed line are true discovered directions. (True direction: $X \to Y$, False direction: $Y \to X$)

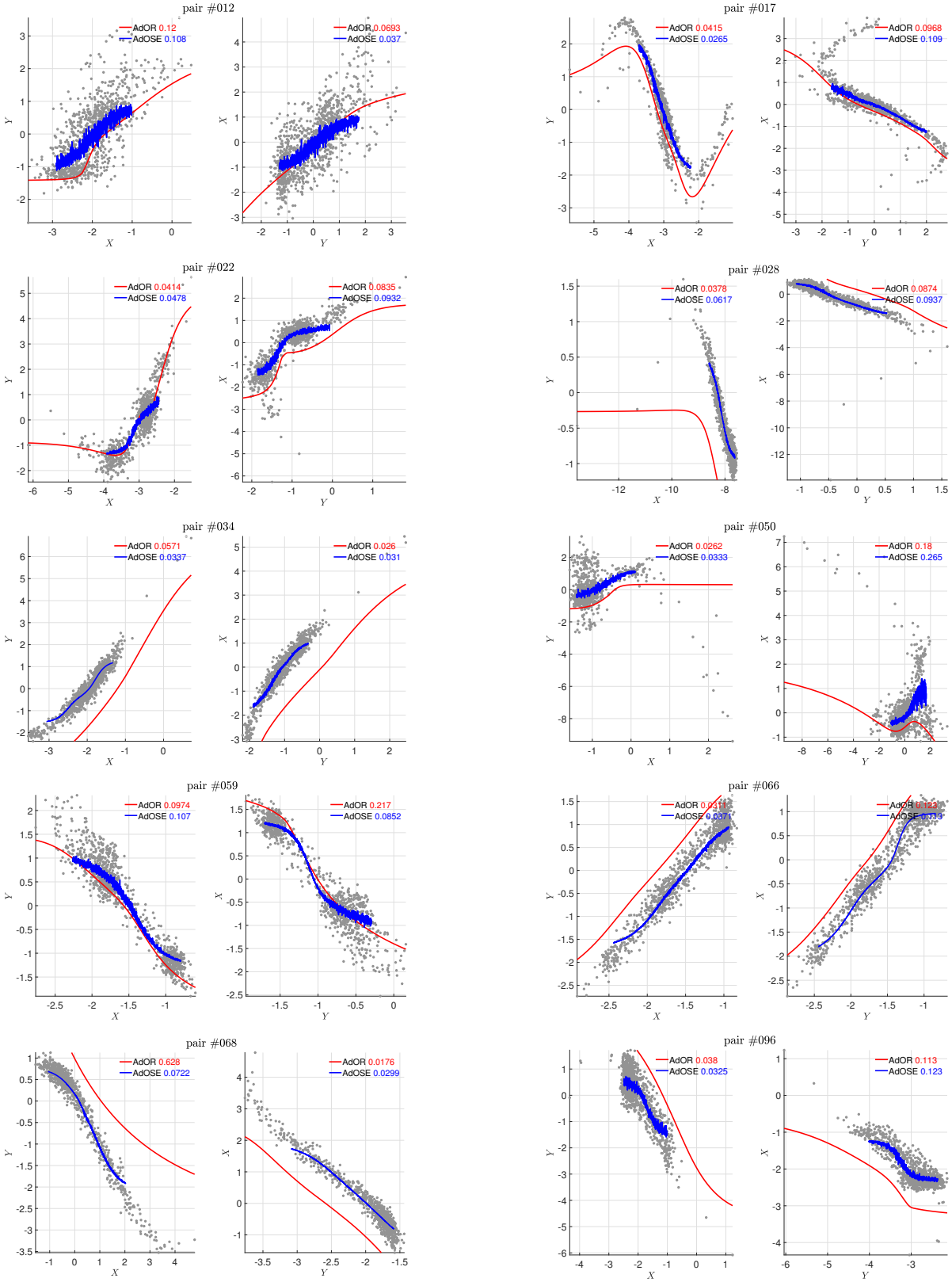


FIG. 10: Predicted function of AdOR and AdOSE for ten example pairs of *SIM* dataset. Regressions are done in both directions $X \to Y$ and $Y \to X$. Estimated mutual information is written at the legend of each figure.

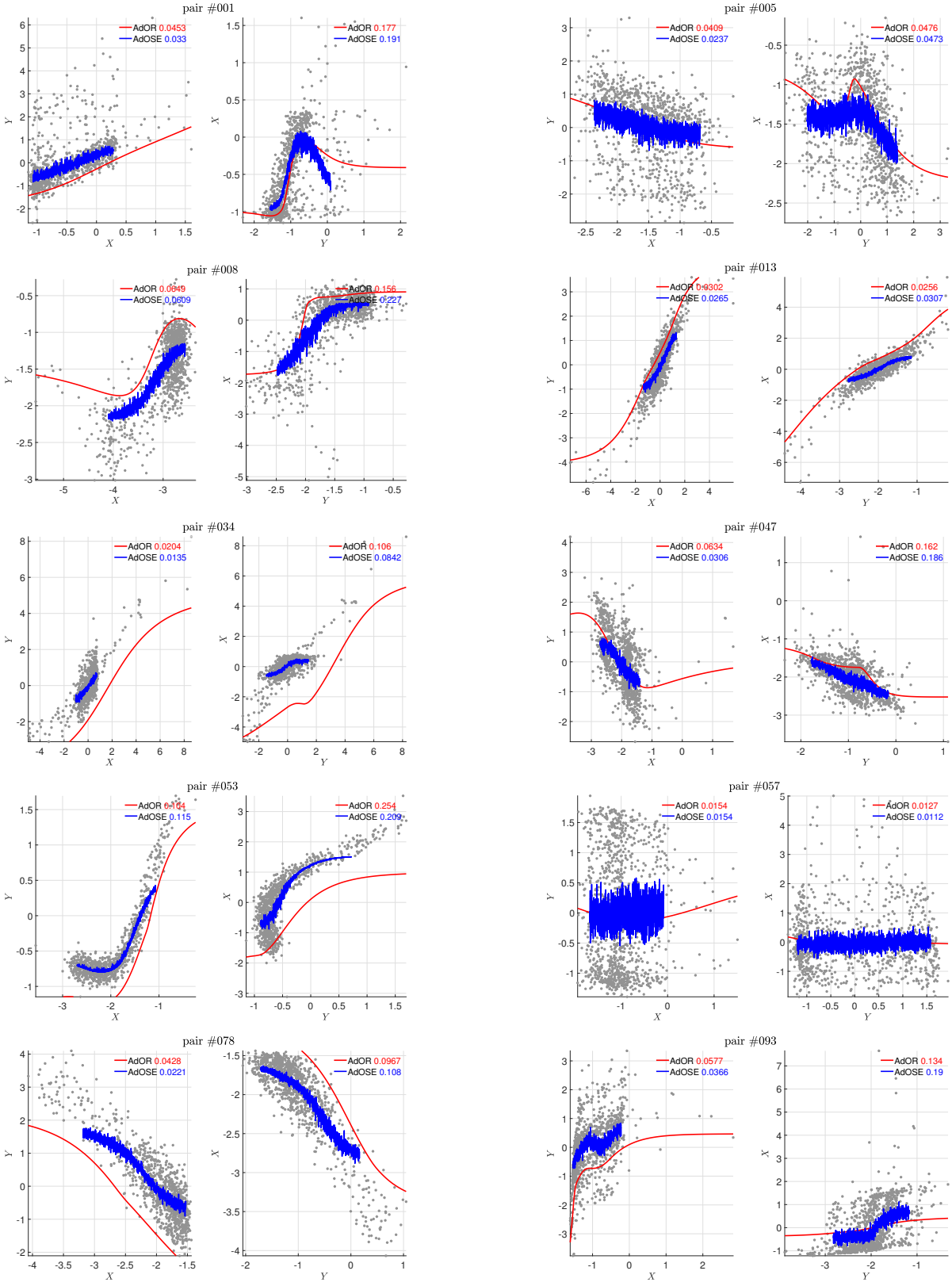


FIG. 11: Predicted function of AdOR and AdOSE for ten example pairs of *SIM-c* dataset. Regressions are done in both directions $X \to Y$ and $Y \to X$. Estimated mutual information is written at the legend of each figure.

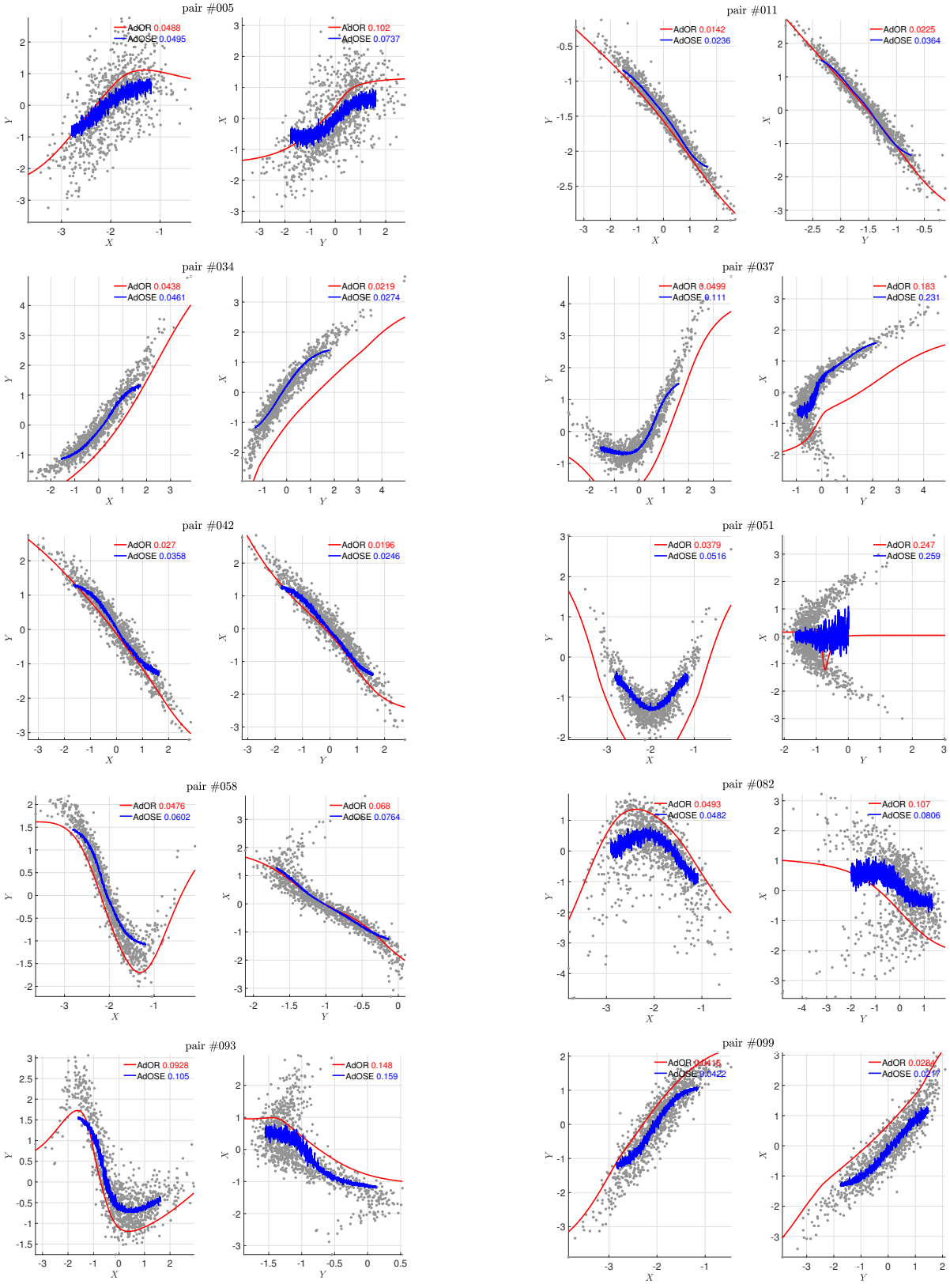


FIG. 12: Predicted function of AdOR and AdOSE for ten example pairs of *SIM-G* dataset. Regressions are done in both directions $X \to Y$ and $Y \to X$. Estimated mutual information is written at the legend of each figure.

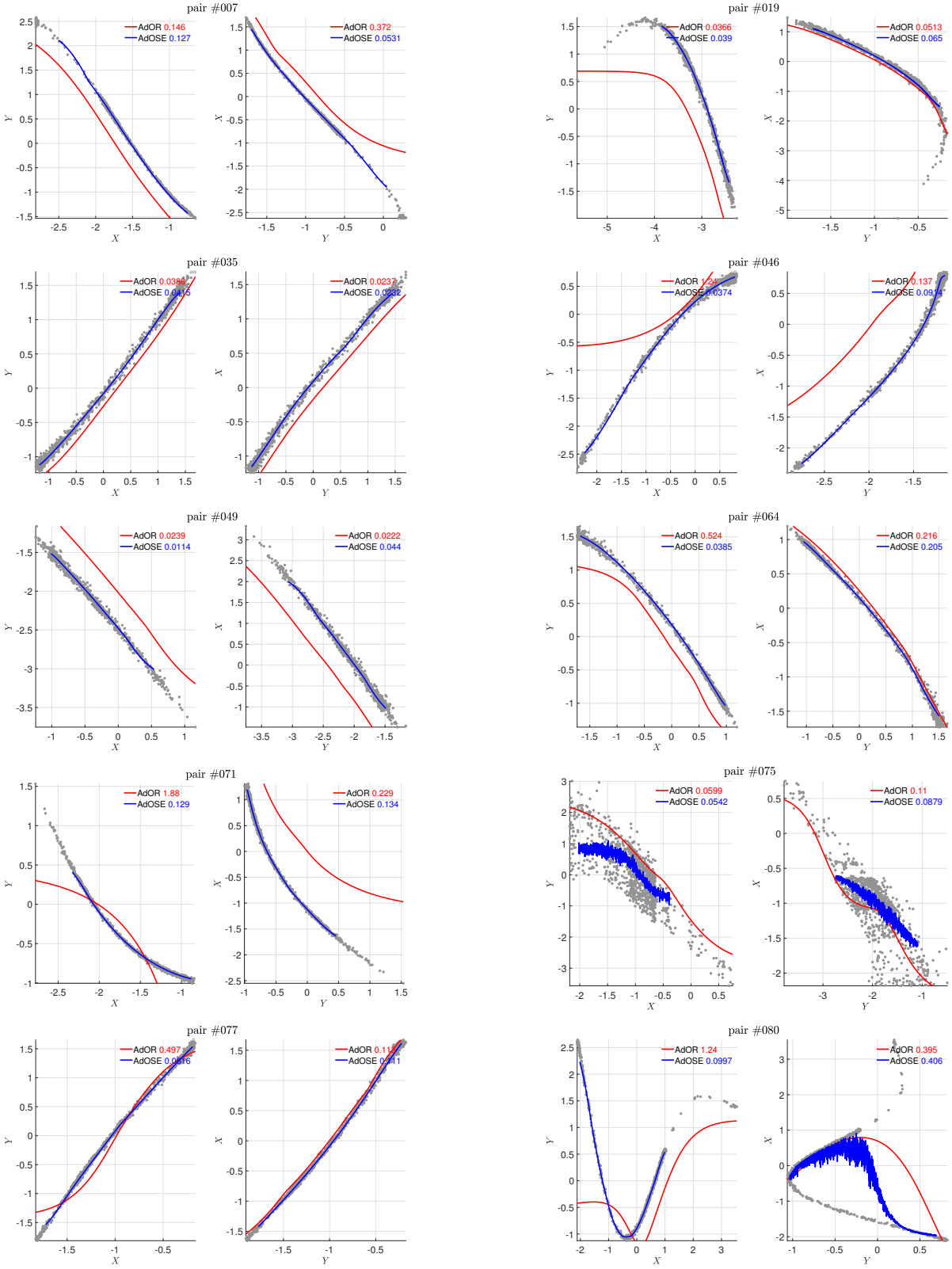


FIG. 13: Predicted function of AdOR and AdOSE for ten example pairs of *SIM-ln* dataset. Regressions are done in both directions $X \to Y$ and $Y \to X$. Estimated mutual information is written at the legend of each figure.

CODE AVAILABILITY

All codes are available in `Code` directory. We implemented AdOR and AdOSE using Python 3 and Tensorflow 1 package [2]. AdOR and AdOSE packages are placed in `ador.py` and `adose.py`. We also attached examples of how to use them with detailed explanation. (github link)

[1] Mooij JM, Peters J, Janzing D, Zscheischler J, Scholkopf B. Distinguishing cause from effect using observational data: methods and benchmarks. The Journal of Machine Learning Research. 2016 Jan 1;17(1):1103-204.

[2] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M. and Ghemawat, S., 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467.