# Supplementary Materials

## 1 Tuning and architectures of models

The model we use includes *support vector machine* (SVM), *multilayer perceptron* (MLP), and *long-short term memory* (LSTM) neural network.

For the SVM regression model, we use 5-fold cross-validation [2], training the model on 4 folds and leaving the other 1 fold as the validation set to search for hyperparameters. We use the grid searching method and pick the best combination of hyperparameters based on the L2-norm of residuals $\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|_2$ of the validation set. The hyperparameter space for search is shown as follows.

Denote the general kernel as Equation 1,

$$f(\boldsymbol{x}) = \beta_0 + \sum_{i=1}^{N} \alpha_i K(x_i, \boldsymbol{x}) \tag{1}$$

where $\alpha_i$ is the contribution of $i$-th training sample to the decision boundary, $N$ is the batch size, $K(\cdot)$ is the kernel function, each $x_i$ when $\alpha_i \neq 0$ is a support vector, and $\beta_0$ is the intercept. The SVM regression is to minimize the objective function as Equation 2,

$$\frac{1}{2}\sum_{j=1}^{d} \beta_j^2 + C\sum_{i=1}^{N} \max(0, |y_i - f(x_i)| - \epsilon) \tag{2}$$

where $d$ is the dimension of kernel, $\beta_j, j = 1, ..., d$ is the coefficients of the decision boundary, $C$ is the strength of penalty, $f$ is the kernel in Equation 1.

We consider the hyperparameter space $C \in \{0.01, 0.1, 1, 10\}$ and $K \in \{K_1, K_2\}$ where $K_1(x_i, \boldsymbol{x}) = (\gamma x_i^T \boldsymbol{x} + 1)^d$ and $K_2(x_i, \boldsymbol{x}) = \exp(-\gamma\|x_i - \boldsymbol{x}\|^2)$. Since the number of independent variables is $mN$, we assign $\gamma = \frac{1}{mN}$.

For the MLP model, the structure is shown in Figure 1.

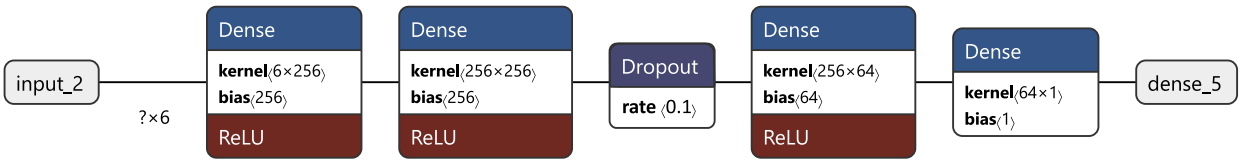


Figure 1: Structure of MLP

We split the training set again, using the first 80% observations to fit this neural network and the last 20% observation as the validation set. We apply the early stopping mechanism, which means the model will stop training not only when the number of iterations reaches 5000, but also when the MSE of the validation set does not decrease for 60 iterations. When the training process halts, we save the best model in training history.

We use Adam optimizer [1] with a learning rate of 0.01 to train this neural network. The loss function is mean squared error (MSE). After the training process converges, we apply this model to the test set and obtain the residuals, which are used to calculate the causality statistic.

Similarly, we apply this training process to the LSTM model, whose structure is shown in Figure 2.

To further improve our model by diminishing the effect of noise, we apply L2 regularization on both neural networks. Denote the kernel weight of a particular neural layer is $\boldsymbol{w}$, and the loss function of training this neural network is $\mathcal{L}$. When we apply L2 regularization, the loss function becomes $\mathcal{L} + \lambda\|\boldsymbol{w}\|_2$. In this case, we use this method as follows.

- For the MLP model, we use $\lambda = 0.1$ on the first and second dense layer, thus building the MLP-L2 model.
- For the LSTM model, we use $\lambda = 0.01$ on the LSTM layer and the first dense layer, thus building the LSTM-L2 model.

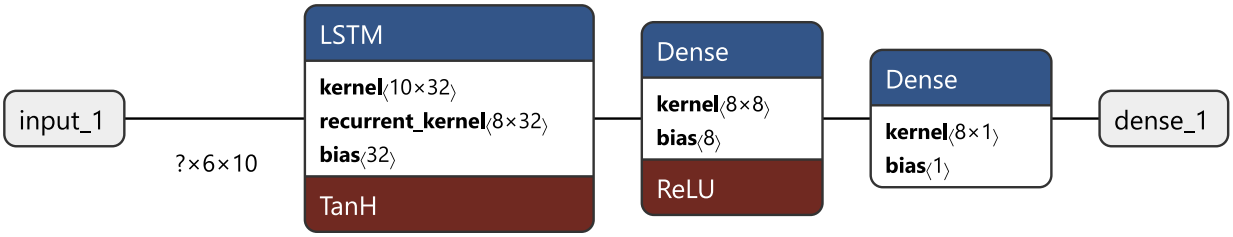


Figure 2: Structure of LSTM

## 2 Lorenz96 system

We use the Lorenz96 system to validate methods. The algorithm is an ordinary differential equation as Equation 3,

$$\begin{gathered} x_{1:N,0} \sim N(0, \sigma_0^2) \\ \frac{\mathrm{d}x_{i,t}}{\mathrm{d}t} = (x_{i+1,t} - x_{i-2,t})x_{i-1,t} - x_{i,t} + f \\ i = 1, ..., N \quad t = 0, \delta, 2\delta..., L\delta \end{gathered} \tag{3}$$

where $N = 10, L = 3000, \delta = 0.1, \sigma_0 = 0.01, f = 10$. With the generated sequences, we remove the first 1000 time points where the system has yet to reach a chaotic state. Hence, we obtain a dataset with 10 series, each with 2000 time points.

## References

[1] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

[2] Stone, M.: Cross-validatory choice and assessment of statistical predictions. Journal of the Royal Statistical Society: Series B (Methodological) **36**(2), 111–133 (1974)