LETTER

Local Density Estimation Procedure for Autoregressive Modeling of Point Process Data

Nat PAVASANT[†a)], **Takashi MORITA**[††], ***Nonmembers***, **Masayuki NUMAO**[††], ***and*** **Ken-ichi FUKUI**[††], ***Members***

SUMMARY We proposed a procedure to pre-process data used in a vector autoregressive (VAR) modeling of a temporal point process by using kernel density estimation. Vector autoregressive modeling of point-process data, for example, is being used for causality inference. The VAR model discretizes the timeline into small windows, and creates a time series by the presence of events in each window, and then models the presence of an event at the next time step by its history. The problem is that to get a longer history with high temporal resolution required a large number of windows, and thus, model parameters. We proposed the local density estimation procedure, which, instead of using the binary presence as the input to the model, performed kernel density estimation of the event history, and discretized the estimation to be used as the input. This allowed us to reduce the number of model parameters, especially in sparse data. Our experiment on a sparse Poisson process showed that this procedure vastly increases model prediction performance.

key words: *point process, vector autoregressive, kernel density*

1. Introduction

A temporal point process is a series of discrete events in the continuous time. For example, the timestamp of stock market transactions [1], earthquakes [2], or neural activities [3]. Temporal point process can be modeled by its intensity function, which controls the rate of event occurrence at a specific time. They can be modeled by their expected distribution like Poisson or Hawkes process [4], and recent works have even been modeling this intensity function using neural network [1] or meta learning [5].

The vector autoregressive (VAR) model works by modeling each variable by its own history and is normally used in time series data. It can also be used to model a point process data by dividing the entire temporal history of the temporal point process into many windows with small lengths, which can then be converted into a binary time series whose value is based on the presence of data in each time window. The graphical view of this model is shown in Fig. 1. This approach, while lacking in accuracy compared to advanced models using neural network [1], has an advantage in that the model is simple and is easily extensible to other applications that are based on vector autoregressive model such as Granger causality [3], [6], which is a statistical process of inferring causality from the data based on the performance of a predictive model [7].

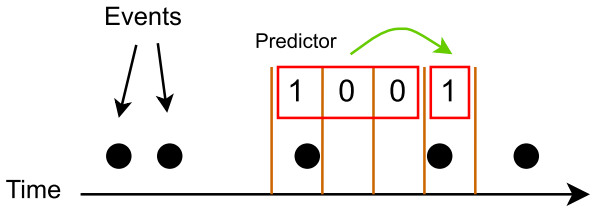


Fig. 1 A counting-based VAR model.

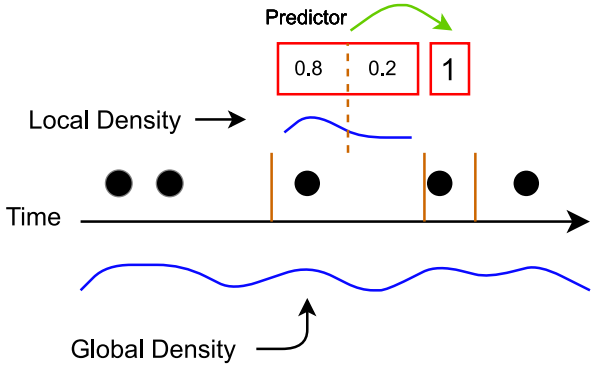


Fig. 2 A model using local density estimation, as opposed to global density.

However, one of the main problems of using vector autoregressive to model the point process is the problem of the history length. To allow the VAR model to cover longer history length, either we must increase the number of history windows, increasing the number of model parameters and complexity, or we must increase the size of each window, losing the time resolution in the process.

To solve this problem, we proposed a new procedure called *local density estimation*, which is a pre-processing step to modeling the VAR model. Specifically, instead of modeling the history of temporal point process data just by the presence of data, we instead perform a kernel density estimation over a fixed size of the temporal history and then apply auto-regression on the estimated density. This model is shown in Fig. 2, which also shows why we called this a local density estimation, as opposed to global density. The procedure allowed the VAR model to better capture the precise location of each data in the point process, especially on sparse data, as well as allow easy scaling to longer temporal history length by having a few parameters covering a long

Manuscript received November 30, 2023.
Manuscript revised March 29, 2024.
Manuscript publicized July 11, 2024.

[†]Graduate School of Information and Technology, Osaka University, Suita-shi, 565–0871 Japan.
[††]SANKEN (The Institute of Scientific and Industrial Research), Osaka University, Ibaraki-shi, 567–0047 Japan.
a) E-mail: p-nat@ai.sanken.osaka-u.ac.jp

DOI: 10.1587/transinf.2023EDL8084

time span, while keeping the number of inputs to the model at a manageable level.

Note that, this work only uses a linear VAR model. A simple extension to the non-linear model including non-linear VAR (NVAR) [8] is possible, as the proposed method can be used as a pro-processing step to transform discrete-time point process model to a continuous-time point process model. This is similar to how a transfer-entropy, a model-free method of estimating causality, can also be extended from discrete-time entropy to continuous-time entropy when applying on point process [9].

Using tophat kernel, which is similar to nearest neighbor density, in addition to a linear and a Gaussian kernel density model, we performed experiments with synthetic data generated with the Poisson model, which showed that our kernel-density pre-processing step improved the accuracy of prediction while still maintaining the same number of inputs.

2. Methodology

2.1 VAR Modeling of Point Process

Vector autoregressive (VAR) is a model where a variable at the current time step is predicted by the past value of itself. For a general VAR model, consider a time-series $\mathbf{A} = \{a_0, a_1, \ldots, a_n\}, a_i \in \mathbb{R}$, we can model a value of a_i by:

$$a_i = \beta_0 + \sum_{j=1}^{k} \beta_j a_{i-j} + \varepsilon_i, \tag{1}$$

where k is the number of lagged variables, β is the model parameter, and ε_i is the error term.

A cumulative incidence function (CIF) is a core process of modeling a point process. The function indicates the rate of event occurrence at the specific time t parameterized by the history of event occurrence:

$$\lambda(t|H(t)) = \lim_{\Delta \to 0} \frac{Pr[(N(t+\Delta) - N(t)) = 1]}{\Delta}, \tag{2}$$

where $N(t)$ is a counting measure of the event within the time of $(0, t]$, and $H(t)$ is an occurrence history of all event occurrences up to time t. The probability of the event occurring in a small time window $[t, t+\Delta)$ can be written as $\lambda(t|H(t))\Delta$.

To use VAR to model a point process CIF, we divided the timeline into small slices of window. We then take a number of events that occur in each slice of the window to be the value at each time step of the VAR model. More formally, consider a point process $\mathbf{X} = \{x_1, x_2, \cdots, x_n\}$ where x_i is a timestamp of each event in the point process. Let $T_0 = x_1$ and $T_1 = x_n$ be the minimum and maximum timestamp of the event, we divided the whole timeline into $K = (T_1 - T_0)/W$ slices of the window where W is the window size. Let R_i denote the number of occurrence of events in the time window $[T_0 + iW, T_0 + (i+1)W)$, and $R(t)$ denote the R_i that is correspondent to the time t.

To model the incidence function, a generalized linear model (GLM) framework was used to model the CIF. In GLM, the logarithm of the CIF was modeled using a linear combination of the occurrence history:

$$\log \lambda(t|\theta, H(t)) = \theta_0 + \sum_{m=1}^{k} \theta_m R(t - mW), \tag{3}$$

where θ_0 is a background activity, and θ_m is the effect of $R(t)$. A point process likelihood function [10] was used to fit the GLM model.

2.2 Local Density Estimation

In order for the standard VAR model to capture longer history, we need to either 1) increase the number of history slices, or 2) increase the windows size W. Both are not ideal: increasing the number of history slices results in an increased number of model parameters, which affect the runtime performance of the process; while increasing the size of the windows W results in reduced temporal accuracy. This can be problematic, especially in sparse data where a longer history length may be required.

To fix the aforementioned problems, we introduced 1-dimensional kernel density pre-processing to the VAR model. Instead of using the lagged variable directly, we sampled from a kernel density estimation trained on the event occurrence history of each prediction. Note that we only use event occurrence history relevant to each prediction for estimation to save on computational cost and avoid information leakage from the predictor target. This allowed us to increase the history length of the model while keeping the number of model parameters low and still keeping some accuracy. We called this procedure *local density estimation.*

Formally, given kernel K, bandwidth b, history length h, and number of parameter p, to model CIF at the time window $[t, t+\Delta)$, we first created a list of events during the time $[t-h, t)$, $\hat{\mathbf{X}} = \{x_i; t - h \leq x_i < t\}$. Then, we can discretize the estimated density $\mathbf{D} = \{d_1, \ldots, d_p\}$ from the event list $\hat{\mathbf{X}}$ using kernel density estimation. The density at d_i can be calculated using the following formula:

$$d_i = \frac{1}{\hat{n}} \sum_{j=1}^{\hat{n}} K((t - \frac{ih}{p-1}) - \hat{x}_j, b), \tag{4}$$

where $\hat{n} = \|\hat{\mathbf{X}}\|$. The discretized $\mathbf{D}$ is used instead of $R(t - mW)$ in Eq. (3) for modeling a point process:

$$\log \lambda(t|\theta, H(t)) = \theta_0 + \sum_{i=1}^{p} \theta_i d_i, \tag{5}$$

with p being the number of parameters, and can also be less than k used in the regular VAR model.

In this work, we used three types of kernels K: a tophat (TOP), a linear (LIN) and a Gaussian (GAU) kernel:

$$K_{TOP}(x, b) \propto 1 \text{ if } |x| < b, \tag{6}$$

$$K_{LIN}(x,b) \propto 1 - |x|/n \text{ if } |x| < b, \tag{7}$$

$$K_{GAU}(x,b) \propto \exp\left(-\frac{x^2}{2b^2}\right). \tag{8}$$

3. Experiments

We tested our proposed kernel-density pre-processing using tophat, linear, and Gaussian kernel against a regular vector autoregressive model using synthetic sparse Poisson process data. We then measured the mean-squared error of the prediction result, the log-likelihood of the GLM model in Eq. (3), and F1 score of each model for comparison.

Our synthetic data is regular Poisson process data have the interval between each event occurrence followed an exponential distribution:

$$\mathbf{L} = \{l_i \sim \text{Exp}(\lambda)\}, \tag{9}$$

$$\mathbf{X} = \{x_i = \sum_{j=0}^{i} l_j\}, \tag{10}$$

where λ is the exponential distribution mean. We added sparsity to this point process by randomly replacing s number of l_i with $g_i \sim \text{Uniform}[10, 1000]$. This created a random large gap within the timeline of the point process. We called this parameter s a sparsity count. In this work, we used $\lambda = 1$ for the Poisson process, which has an average interval of 1 and 90% of the intervals are less than 3. Sparsity count s of 0.01%, 0.02%. 0.05%, 0.1%, 0.2%, 0.5%, 1%, 2%, 5%, and 110% of all data were used. The histograms of the interval between events are shown in Fig. 3.

We generated a 100,000-points point process for the experiments, with 80% of the points being used for training and another 20% of the points for evaluation. We tested the regular VAR model (denoted as CNT), our proposed model with tophat kernel, linear (triangle) kernel, and Gaussian kernel (denoted as LIN and GAU). The details were described in Table 1. The history length was the overall length of the history being used in each prediction, and the number of parameters described the number of inputs to the model. This is also shown in Fig. 4. All VAR models have a window size of 1. All models have a target window size of 1.

We performed each experiment 10 times and took the average of the results. The MSE, the log-likelihood of the predictor, and the F1-score, calculated by thresholding the predictor output at 0.5, are shown in Fig. 5. In almost every case, the GAU5 model performed the best, followed closely by TOP5 and LIN5. The VAR model CNT5 and CNT20 performed worse in almost every case. Figure 5 (c) also shows that TOP5, LIN5, and GAU5 were also less affected by the sparsity. GAU5 outperformed CNT20 with significantly fewer model parameters; GAU5 had only 5 parameters, whereas CNT20 utilized 20. Note that as sparsity increased, the data can get extremely imbalanced so the MSEs were lower with higher sparsity.

We have also generated a smaller point process with 1,000 points and 10,000 points, and performed the same experiment with CNT20 (regular VAR with 20 parameters) and GAU5 (Gaussian kernel with 5 parameters). The F1-score is shown in Fig. 6. The result shows that the proposed method improved the result even with low number of points, especially at higher sparsity ratio.

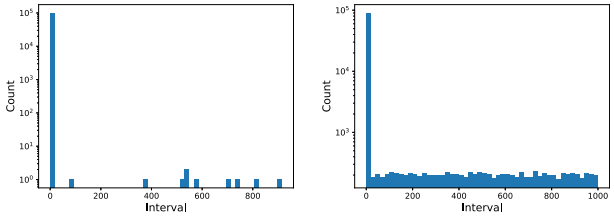
3.1 Complexity Analysis

Preparing a temporal point process data, specifically for sparse data, for VAR modeling has the complexity of $O(LN + N \log N)$ where L is the number of windows, W is the window size, and N is the number of data. This came from the following steps:

1. For each data point N:
   a. Find the points that are in the history length ($O(\log N)$ using binary search)
   b. Construct a history model from at most N points ($O(N)$)
2. However, in Step 1b, note that all points can be part of

Table 1 Models used in the experiment.

Name	Kernel	History	#Params	Bandwidth
CNT5	None	5	5	-
CNT20	None	20	20	-
TOP5	Tophat	5	5	2
TOP20	Tophat	20	5	5
LIN5	Linear	5	5	2
LIN20	Linear	20	5	5
GAU5	Gaussian	5	5	2
GAU20	Gaussian	20	5	5



(a) 0.01% Sparsity (b) 10% Sparsity

Fig. 3 Histogram of interval between events at different sparsity from the 100,000 points dataset

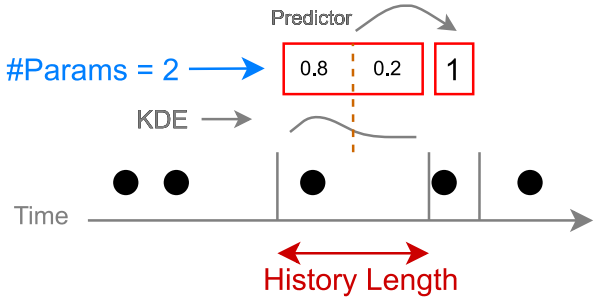
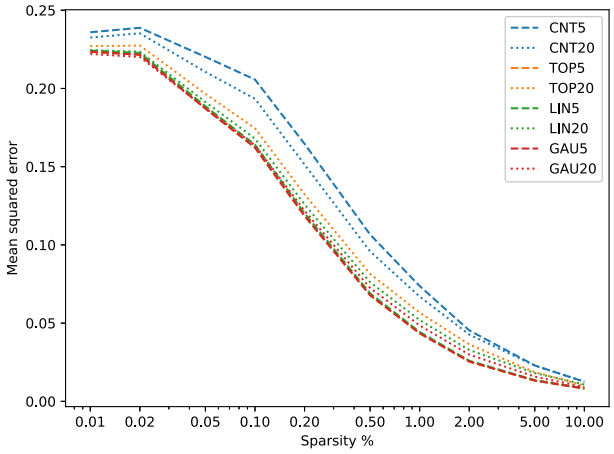
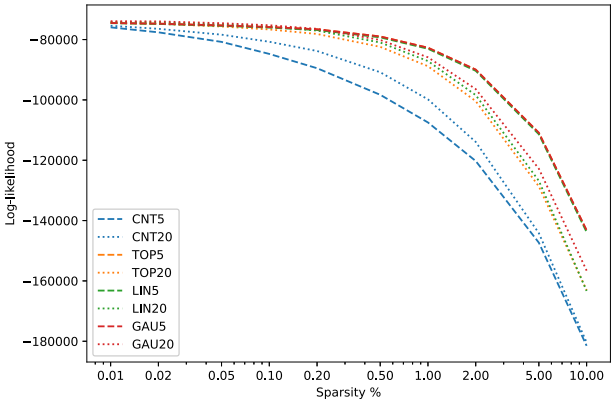


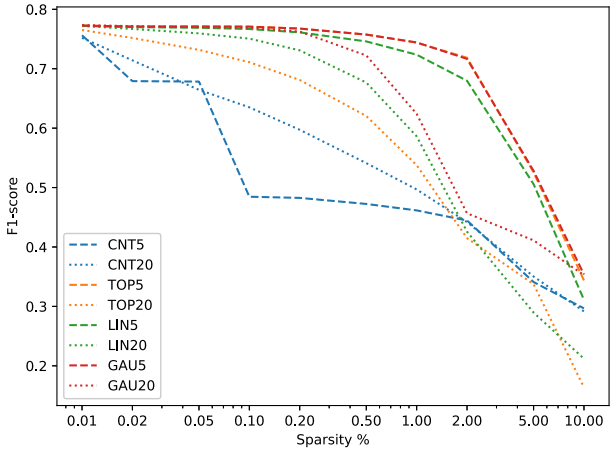
Fig. 4 Difference between the history length and the number of parameters.



(a) Mean squared error. Lower is better.



(b) Log-likelihood. Higher is better.



(c) F1-score. Higher is better.

Fig. 5 Mean squared error (MSE), log-likelihood, and F1-score of each model. CNT is a regular VAR model, while TOP, LIN, and GAU are proposed method with the tophat, linear, and Gaussian kernel, respectively.

at most L history models. Hence, step 1b amortized to $O(LN)$

Step 1, minus the amortized part, has the complexity of $O(N \log N)$. The amortized part is $O(LN)$, yielding the final complexity of $O(LN + N \log N)$.

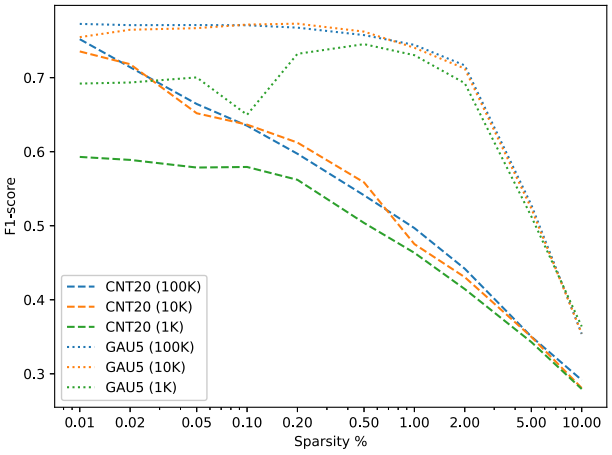


Fig. 6 Comparison of F1-score at each data size using CNT20 and GAU5. Higher is better.

For the proposed local density estimation, the complexity is $O(\frac{h}{w}N+Np+N \log N)$ where h is the history length, p is the number of history samples (number of parameters), and w is the time step used for the prediction target. Similarly, this came from:

1. For each data point N:
   a. Find the points that are in the history length ($O(\log N)$ using binary search)
   b. Construct and sample p samples of density from at most N points ($O(N + p)$)
2. However, in Step 1b, again, all points can be part of at most $\frac{h}{w}$ history models. Hence, step 1b amortized to $O(\frac{h}{w}N + Np)$

Note that $\frac{h}{w}$ is essentially L in the complexity of the regular VAR model. Hence, the proposed algorithm can only be asymptotically slower than the regular VAR model if and only if Np is larger than both $\frac{h}{w}N$ and $N \log N$, which seems unlikely, as part of the reason to use this procedure is to reduce the number of model parameter p to be less than $\frac{h}{w}$.

4. Conclusion

We have proposed a new procedure to pre-process data with kernel density estimation for modeling a point process using a vector autoregressive (VAR) model. Our experiments showed that on sparse data, our procedure increased the prediction performance significantly over the regular VAR model, especially with the Gaussian kernel, while still keeping the number of model parameters low. This procedure can be applied anywhere a VAR model is being used to model a point process, especially for Granger causality inference. The proposed method is also robust against the number of data points. While this work used the tophat, linear, and Gaussian kernels, which are the most common kernels, depend on the data, other kernels such as cosine can also be

used. We have also shown that the complexity of the proposed procedure is equivalent to the regular VAR model.

References

[1] T. Omi, N. Ueda, and K. Aihara, "Fully neural network based model for general temporal point processes," Advances in Neural Information Processing Systems, 2019.

[2] Y. Ogata, "Space-time point-process models for earthquake occurrences," Annals of the Institute of Statistical Mathematics, vol.50, no.2, pp.379–402, 1998.

[3] S. Kim, D. Putrino, S. Ghosh, and E.N. Brown, "A granger causality measure for point process models of ensemble neural spiking activity," PLOS Computational Biology, vol.7, no.3, p.e1001110, 2011.

[4] P. Bremaud, Point process calculus in time and space: An introduction with applications, pp.461–518, Springer International Publishing, 2020.

[5] W. Bae, M.O. Ahmed, F. Tung, and G.L. Oliveira, "Meta temporal point processes," Proc. Eleventh International Conference on Learning Representations, 2023.

[6] N. Pavasant, T. Morita, M. Numao, and K. Fukui, "Granger causality-based cluster sequence mining for spatio-temporal causal relation mining," International Journal of Data Science and Analytics, vol.17, no.3, pp.275–288, 2023.

[7] C.W.J. Granger, "Investigating causal relations by econometric models and cross-spectral methods," Econometrica, vol.37, no.3, pp.424–438, 1969.

[8] L. Kilian and H. Lütkepohl, Nonlinear Structural VAR Models, pp.609–658, Themes in Modern Econometrics, Cambridge University Press, 2017.

[9] G. Mijatovic, Y. Antonacci, T. Loncar-Turukalo, L. Minati, and L. Faes, "An information-theoretic framework to measure the dynamic interaction between neural spike trains," IEEE Trans. Biomed. Eng., vol.68, no.12, pp.3471–3481, 2021.

[10] W. Truccolo, U.T. Eden, M.R. Fellows, J.P. Donoghue, and E.N. Brown, "A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects," Journal of Neurophysiology, vol.93, no.2, pp.1074–1089, 2005.