Extremes manuscript No.
(will be inserted by the editor)

Online supplement for: "A Kolmogorov–Arnold Neural Model for Cascading Extremes"

Miguel de Carvalho[1,2] · **Clemente Ferrer**[3] · **Ronny Vallejos**[3]

Received: date / Accepted: date

1 Supplementary Monte Carlo Evidence

In all Monte Carlo simulations reported below, we generated 500 samples.

1.1 Three-Layer KANE: Extending Simulations from the Paper

Here, we extend the Monte Carlo simulations from the paper in a variety of ways. Firstly, we consider additional sample sizes. Fig. 1.1 depicts the trajectories of the fitted models. In line with the evidence presented in the paper, the fits remain satisfactory, and as expected performance improves with larger sample sizes.

Secondly, we present in Fig. 1.2 the fits that result from Scenarios B1, B2 and C, but where we consider dependence between the covariates through a Clayton copula

$$C(a,b) = \max(a^{-\theta} + b^{-\theta} - 1, 0), \quad (a,b) \in [0,1]^2,$$

with $\theta \in \{1,5,10\}$, and where we now set $n = 2\,500$. As expected, the quality of the fits improves in regions with more observations.

Thirdly, we consider an additional simulation scenario—Scenario D—to assess the performance of the Frank–Hall variant of our approach; Scenario D modifies Scenario C by using probabilities derived from an ordered probit model. Given $\tau_1 < \tau_2$, the follow-up events are simulated as

$$\mathbf{I}_{u,\mathbf{x}} \mid Y_{\mathbf{x}} > u \sim \text{Multi}_3\left(m_{\text{D1}}(\mathbf{x};u), m_{\text{D2}}(\mathbf{x};u), m_{\text{D3}}(\mathbf{x};u)\right),$$

where $m_{\text{D1}}(\mathbf{x};u) = \Phi(\tau_1 - \mathbf{x}^{\text{T}}\boldsymbol{\beta} + 1/u)$, $m_{\text{D2}}(\mathbf{x};u) = \Phi(\tau_2 - \mathbf{x}^{\text{T}}\boldsymbol{\beta} + 1/u) - \Phi(\tau_1 - \mathbf{x}^{\text{T}}\boldsymbol{\beta} + 1/u)$, and $m_{\text{D3}}(\mathbf{x};u) = 1 - \Phi(\tau_2 - \mathbf{x}^{\text{T}}\boldsymbol{\beta} + 1/u)$. The true ordered jth-category POC surfaces in this case are

$$\begin{cases} \alpha_{\mathbf{I}}^{(1)}(\mathbf{x}) = \lim\limits_{u\to\infty} m_{\text{D1}}(\mathbf{x};u) = \Phi(\tau_1 - \mathbf{x}^{\text{T}}\boldsymbol{\beta}), \\ \alpha_{\mathbf{I}}^{(2)}(\mathbf{x}) = \lim\limits_{u\to\infty} m_{\text{D2}}(\mathbf{x};u) = \Phi(\tau_2 - \mathbf{x}^{\text{T}}\boldsymbol{\beta}) - \Phi(\tau_1 - \mathbf{x}^{\text{T}}\boldsymbol{\beta}), \\ \alpha_{\mathbf{I}}^{(3)}(\mathbf{x}) = \lim\limits_{u\to\infty} m_{\text{D3}}(\mathbf{x};u) = \Phi(\tau_2 - \mathbf{x}^{\text{T}}\boldsymbol{\beta}). \end{cases}$$

We set $\tau_1 = -0.5$, $\tau_2 = 0.5$, and $\boldsymbol{\beta} = (1,-1)^{\text{T}}$. In Fig. 1.3, we present the fits of both the Frank–Hall variant of our model, which accounts for the ordinal nature of the data, and the softmax approach, which disregards this structure. Interestingly, the fit quality appears similar across the two approaches.

✉ Miguel de Carvalho
Miguel.deCarvalho@ed.ac.uk

[1] School of Mathematics, University of Edinburgh, The King's Buildings, Edinburgh, EH9 3FD, UK
[2] Department of Mathematics, University of Aveiro, Portugal
[3] Department of Mathematics, Universidad Técnica Federico Santa Maria, Valparaíso, Chile

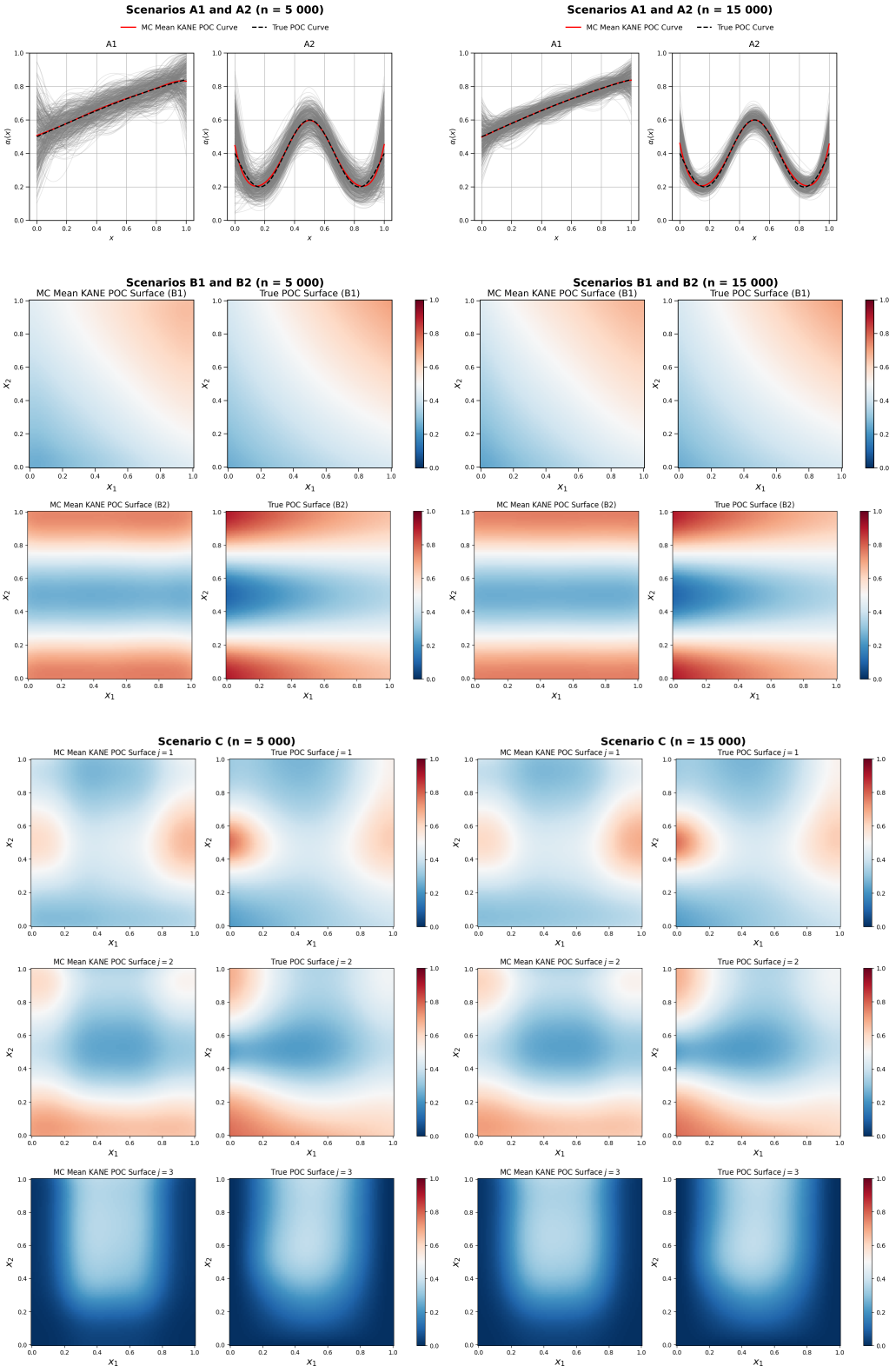

Fig. 1.1 Monte Carlo means for Scenarios A, B, and C for three-layer KANE.

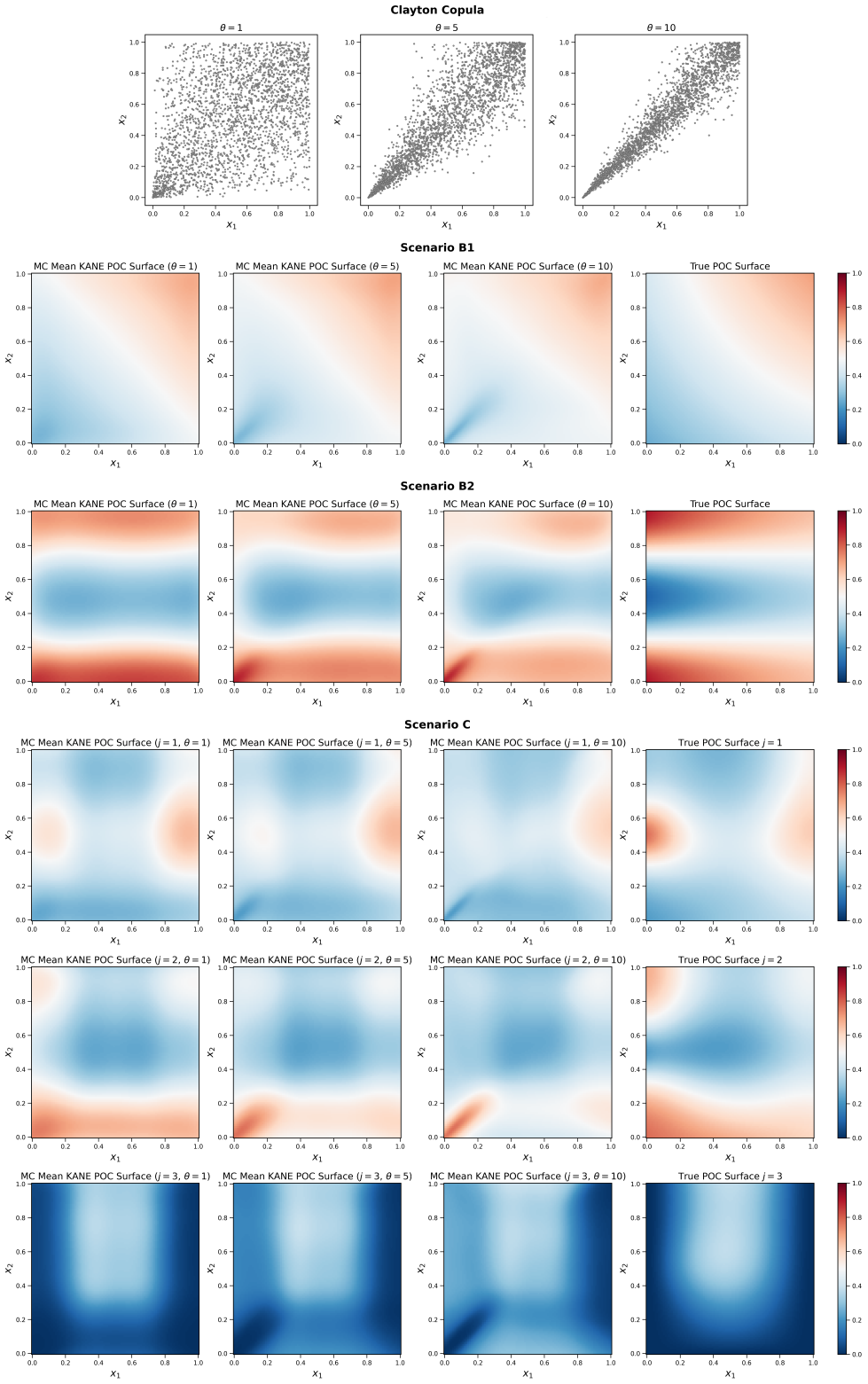

Fig. 1.2 Monte Carlo means for Scenarios A, B, and C for three-layer KANE with dependent covariates.

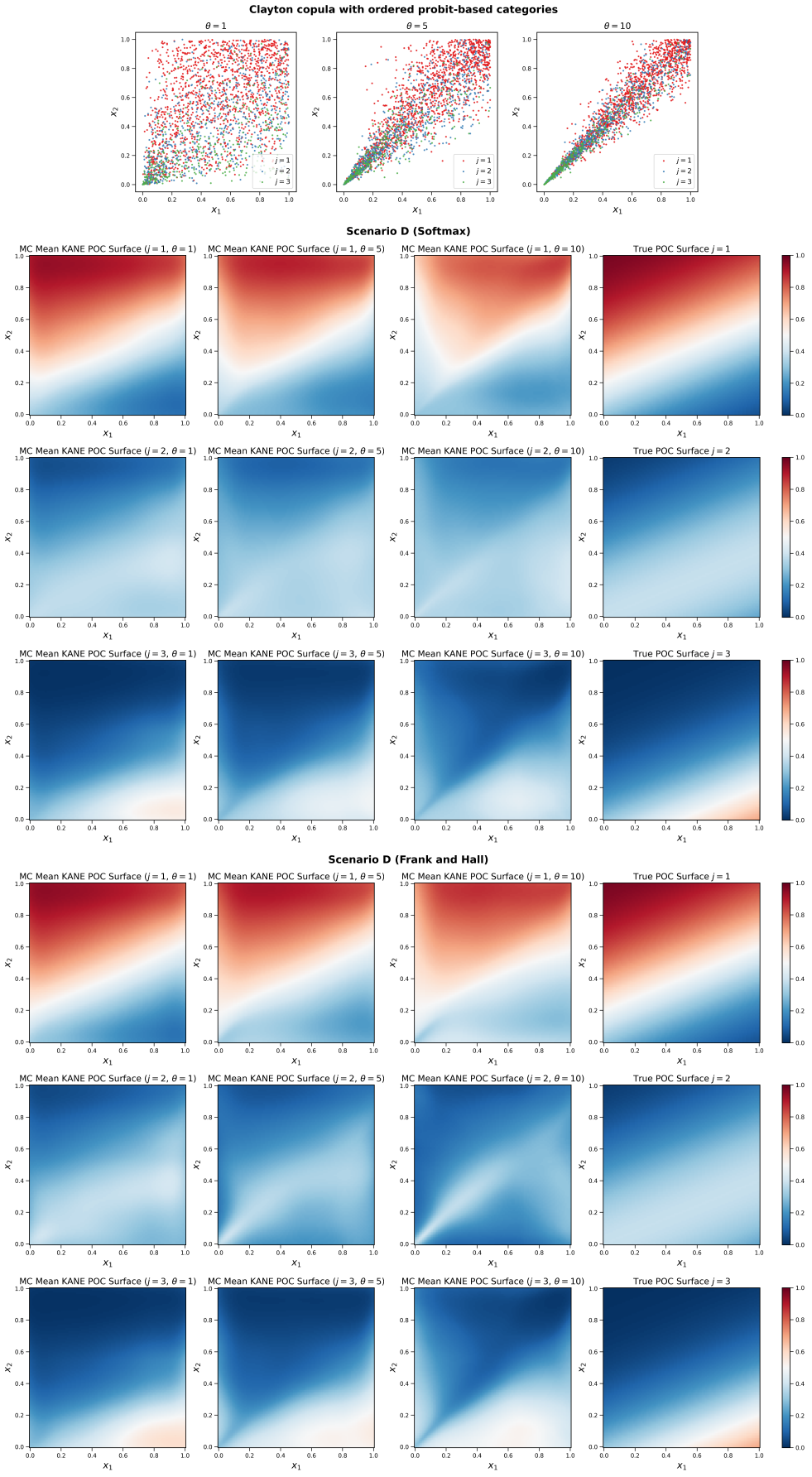

Fig. 1.3 Monte Carlo means for Scenario D for three-layer KANE.

1.2 Coverage Properties

Here we assess the coverage properties of the bootstrap for the POC curve. Specifically, we construct nonparametric bootstrap confidence bands for the POC curve, in Scenarios A1 and A2 with $n = 15\,000$, via a resampling cases bootstrap. We generate 500 datasets and, for each, perform 100 bootstrap resamples. The KANE POC curve is estimated for each resample, and 95% confidence intervals are built using the percentile method. Coverage is then computed as the proportion of cases in which the true POC curve lies within these intervals. Figure 1.4 suggests an overall satisfactory performance, with coverage close to the nominal level.

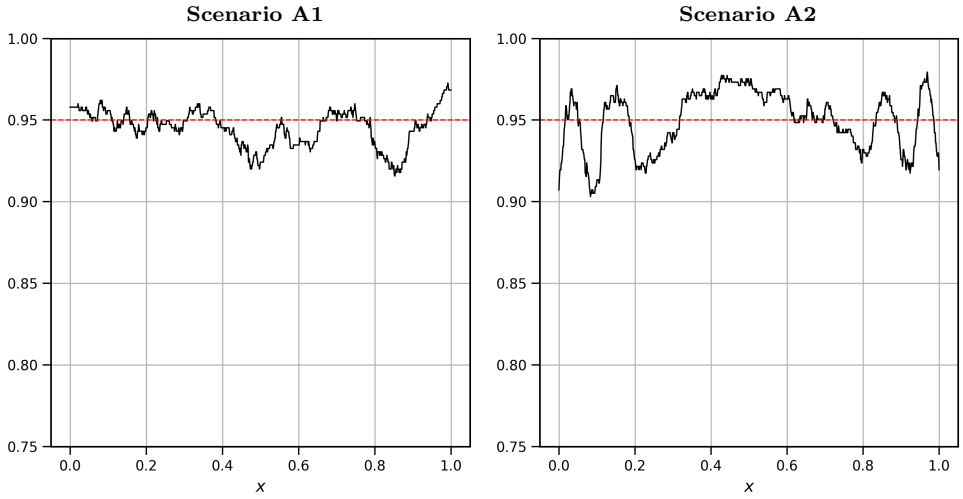

Fig. 1.4 Pointwise coverage of confidence bands for POC curve at the 95% confidence level.

1.3 Deep KANE

This section reports the performance of the deep KANE framework at fitting the same scenarios as Section 4 of the paper, by using a four-layer structure. As can be seen in Fig. 1.5 the fits are overall satisfactory, though not as accurate as in the canonical three-layer structure evaluated over Section 4 of the paper.

1.4 Fitting POC Surfaces via Generalized Additive Models and Multi-Layer Perceptrons

In this section we compare the performance of estimators for our POC surface based on a Generalized Additive Model (GAM) and a Multi-Layer Perceptron (MLP). Trivially, when the outer functions in the Kolmogorov–Arnold representation are set to identity, our KANE model reduces to a GAM. For comparison, we replicate the scenarios from Section 4 of the paper using a logistic model with a GAM-based linear predictor. Following Eilers and Marx (2021, Appendix A), the implementation employs 10 cubic P-splines with a second-order difference penalty, with the smoothing parameter chosen via the Unbiased Risk Estimator (UBRE)—the default prediction error criterion in the `mgcv` package for non-Gaussian families.

Using the same experimental framework, we compare the KANE model with an alternative neural network architecture by implementing a simple MLP consisting of one hidden layer (32 units, ReLU activation) and a sigmoid output layer. We train the model using LBFGS optimizer with binary cross-entropy loss for 100 epochs.

Table 1 presents the MISE for both approaches. As shown in the table, the neural models KANE and MLP consistently outperform GAM across all scenarios, with KANE and MLP alternating in performance depending on the scenario and sample size configuration.

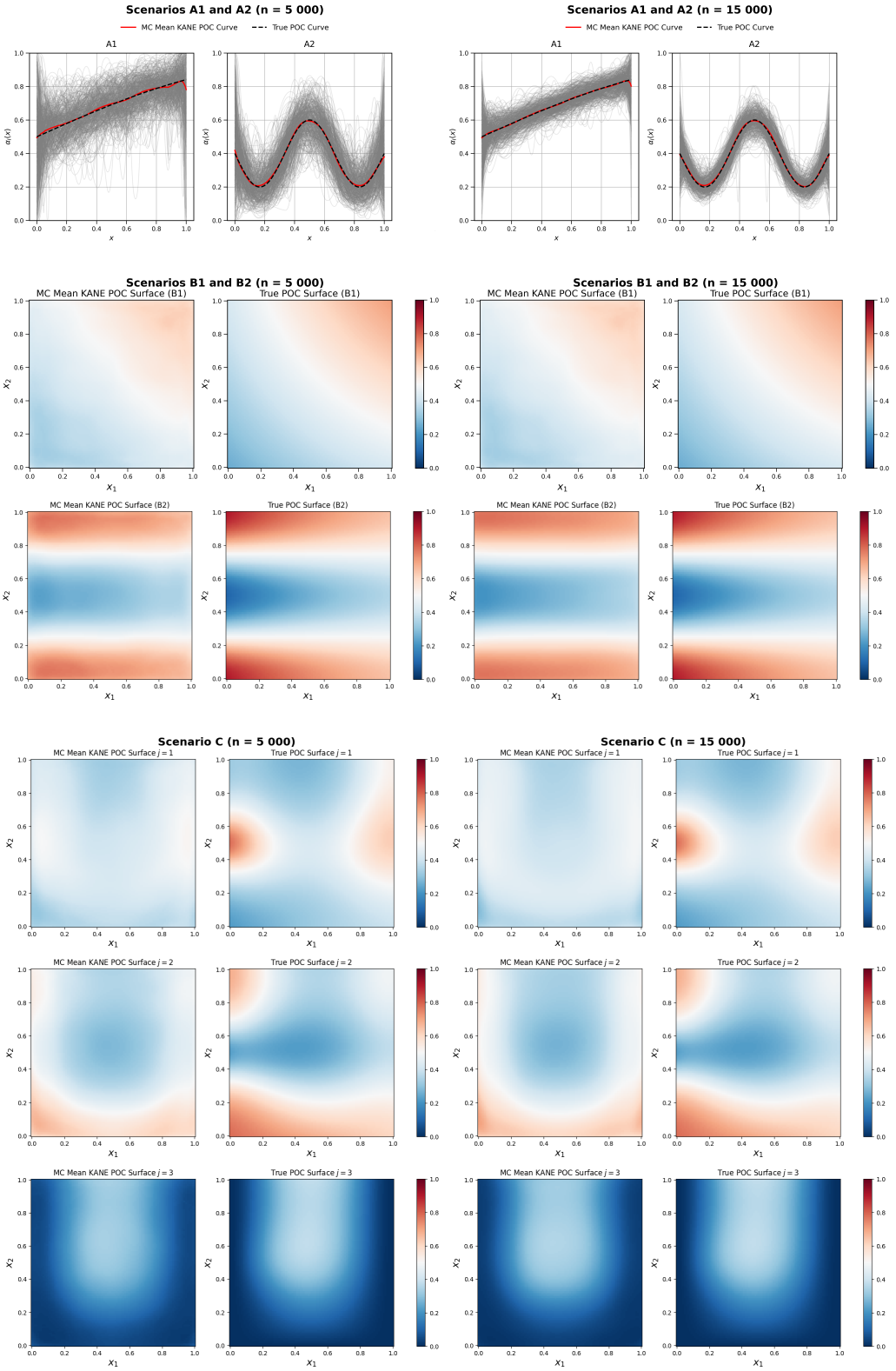

Fig. 1.5 Monte Carlo means for Scenarios A, B, and C for four-layer KANE.

Table 1 MISE values across scenarios and sample sizes for GAM and MLP approaches. Blue indicates cases where KANE outperforms MLP, while red indicates cases where MLP outperforms KANE.

Scenario	Sample Size					
	5 000		10 000		15 000	
	GAM	MLP	GAM	MLP	GAM	MLP
A1	9.993×10^{-6}	1.842×10^{-5}	6.467×10^{-6}	1.091×10^{-5}	2.804×10^{-6}	1.342×10^{-5}
A2	4.114×10^{-4}	1.033×10^{-2}	2.358×10^{-4}	7.229×10^{-3}	1.774×10^{-4}	5.929×10^{-3}
B1	1.090×10^{-4}	2.463×10^{-5}	9.743×10^{-5}	2.298×10^{-5}	8.751×10^{-5}	2.911×10^{-5}
B2	3.052×10^{-3}	1.635×10^{-3}	2.970×10^{-3}	1.156×10^{-3}	2.923×10^{-3}	8.685×10^{-4}
C (j=1)	2.031×10^{-3}	1.396×10^{-3}	3.369×10^{-3}	8.938×10^{-4}	3.532×10^{-3}	7.107×10^{-4}
C (j=2)	3.495×10^{-3}	1.568×10^{-3}	4.659×10^{-3}	9.122×10^{-4}	2.658×10^{-3}	7.370×10^{-4}
C (j=3)	1.907×10^{-3}	9.013×10^{-4}	1.130×10^{-3}	3.468×10^{-4}	1.129×10^{-3}	1.867×10^{-4}

1.5 Increased Covariate Dimension

We now examine the KANE's performance as the covariate dimension increases. For these numerical experiments the follow-up events are simulated from a Bernoulli process with

$$I_{u,\mathbf{x}} \mid Y_{\mathbf{x}} > u \sim \text{Bern}\left\{m_d(\mathbf{x}; u)\right\},$$

for $\mathbf{x} \in [0,1]^p$. Here

$$m_d(\mathbf{x}; u) = g\Big(0.5 + \sum_{i=1}^{p} \Big\{0.25e^{-x_i} \cos\left(2\pi x_i\right) + \frac{1}{u^2}\Big\}\Big),$$

where g is the sigmoid function. The features are independently drawn from the standard uniform distribution, whereas the trigger event is again simulated from a unit Fréchet distribution. The corresponding true POC surface is

$$\alpha_I(\mathbf{x}) = \lim_{u\to\infty} m_d(\mathbf{x}; u) = g\Big(0.5 + \sum_{i=1}^{p} \Big\{0.25e^{-x_i} \cos\left(2\pi x_i\right)\Big\}\Big),$$

and is estimated with a three-layer KANE under the same optimization settings described in the paper. We consider $p \in \{2,3,4,5\}$ on a standard laptop (Apple M3), sample size $n = 5\,000$, and again 500 simulations. For each setting, the MISE and the average runtime per iteration are computed. Table 2 present the results. As can be seen in the table, the model can be fitted on a standard laptop for low-dimensional settings, but when the number of features increases ($d > 5$) the computational cost rises substantially.

Table 2 MISE and average runtime per iteration across dimensions.

Metric	Dimension (p)			
	2	3	4	5
MISE	0.0038	0.0182	0.0451	0.0598
Average Runtime (s)	0.68	3.22	5.76	128.23

1.6 On the Inner and Outer Functions

This experiment illustrates that under our KANE model while the POC surface is identifiable, whereas the inner and outer functions may not be. To show this, we generate data from a canonical three-layer KANE specification; we fix $d = 1$ and set

$$I_{u,x} \mid Y_x > u \sim \text{Bernoulli}\{m(x; u)\},$$

for $x \in [0,1]$. Here

$$m(x; u) = g\Big(\Phi_1^{(2)}\big(\Phi_{1,1}^{(1)}(x)\big) + \Phi_2^{(2)}\big(\Phi_{2,1}^{(1)}(x)\big) + \Phi_3^{(2)}\big(\Phi_{3,1}^{(1)}(x)\big) + 1/u^2\Big),$$

and g is the sigmoid function. The corresponding true POC curve is $\alpha_I(x) = \lim_{u\to\infty} m(x;u)$, and it consists a well-specified KANE model in the sense that it follows from (1.6) that this specification obeys Equation (2.7) in the paper with $d = 1$. As ground truth, the components are taken to be

$$\Phi_{i,1}^{(1)}(x) = x^i \sin(5x), \qquad \Phi_i^{(2)}(x) = 0.1\, x^i, \qquad i = 1, 2, 3.$$

The fitted model uses the same simulation setup and architecture as in Scenarios A1–A2 with $n = 15\,000$; both inner and outer functions are once again modeled using by B–splines with $p = 3$ and $m = 2$. After fitting the model, spline coefficients and knots are extracted to reconstruct the learned components $\hat{\Phi}_{i,1}^{(1)}$ and $\hat{\Phi}_i^{(2)}$.

A Monte Carlo simulation was conducted to assess performance across different simulated datasets. Figure 1.6 shows that, although the Monte Carlo mean POC Curve tracks $\alpha_I(x)$ closely, the inner and outer functions cannot be accurately learned from the data. The lack of identifiability of the inner and outer functions is not surprising; in fact, as noted earlier, when all outer functions reduce to the identity, our KANE model becomes a GAM which is known to require identifiability constraints; these are typically imposed through sum-to-zero conditions on the smooth functions (Wood 2017, Section 6.1).

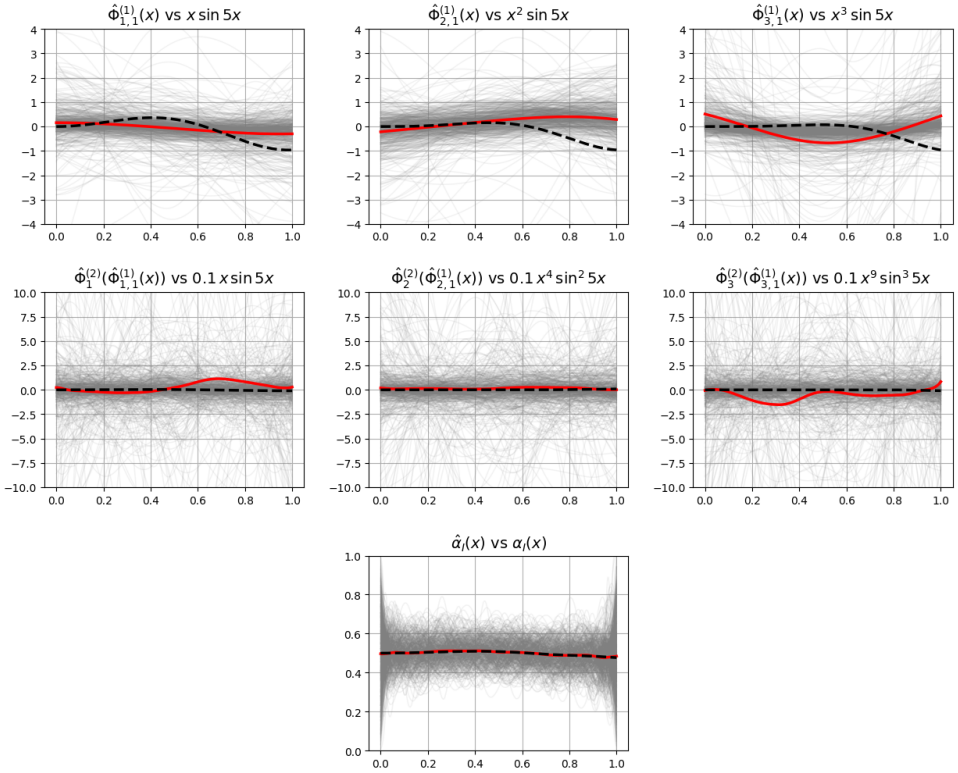

Fig. 1.6 Identifiability assessment under a three-layer KANE model. Red curves show the Monte Carlo mean; dotted curves show the ground truth.

2 Supplementary Empirical Analyses

Figure 1.7 reports 10 trajectories of the QQ-boxplots of the randomized residuals, hence complementing those reported in Section 5 of the paper. These QQ-boxplots provide further evidence in favor of the fits presented in Section 5.

3 Further Technical Details

This section verifies that the simulation examples in the main paper satisfy the assumptions of Theorem 2. We work out the details explicitly for Scenario A2, while the other cases can be

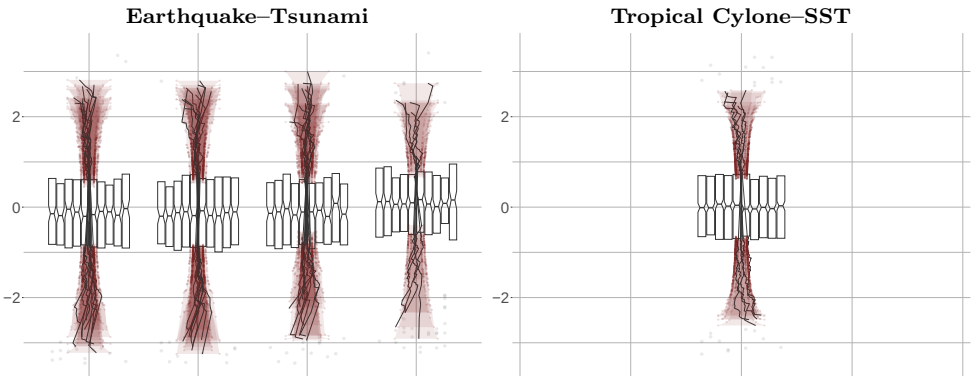

Fig. 1.7 10 trajectories of QQ-boxplots of Dunn–Smyth residuals for empirical illustrations from Section 5 in the paper.

treated analogously and are therefore omitted. First, it trivially follows that for Scenario A2

$$\mathbf{x} \mapsto P(I_{\mathbf{x}} = 1 \mid Y_{\mathbf{x}} > u) = 0.2 \sin\{3\pi(x-1)^2\} + 0.4 + 1/u^2,$$

is continuous for all $u \in \mathbb{R}$, and hence it obeys (a). Second, in terms of (b), in order to ensure that for all $\varepsilon > 0$ there exists u_0 such that for all $u \geq u_0$,

$$|P(I_{\mathbf{x},u} = 1 \mid Y_{\mathbf{x}} > u) - \alpha(\mathbf{x})| < \varepsilon,$$

it suffices to set $u_0 = 2/\sqrt{\varepsilon}$; this follows immediately given that in Scenario A2,

$$|P(I_{\mathbf{x},u} = 1 \mid Y_{\mathbf{x}} > u) - \alpha(\mathbf{x})| = 1/u^2.$$

Hence, (b) holds.

References

Eilers PH, Marx BD (2021) Practical Smoothing: The Joys of P-splines. Cambridge University Press, Cambridge

Wood SN (2017) Generalized Additive Models: An Introduction with R. Chapman & Hall/CRC, Boca Raton, FL