Extreme learning machines for variance-based global sensitivity analysis

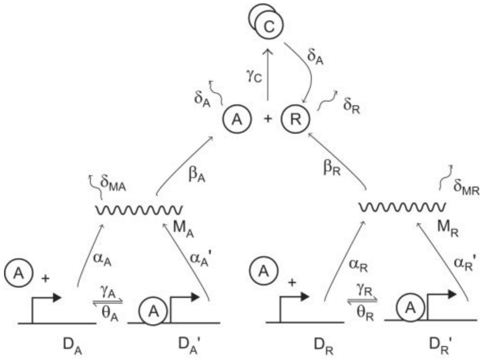
John Darges

Department of Mathematics
North Carolina State University
Joint work with Alen Alexanderian and Pierre Gremaud

August 10, 2022

Motivating example - genetic oscillator

Biochemical model describing circadian rhythm regulation:



Which rate constants need to be measured most accurately so we can determine the concentration of R?

Image credit[1]

[1] J.G. Vilar, H.Y. Kueh, N. Barkai, S. Leibler. Mechanisms of noise-resistance in genetic oscillators. 2002.

Introduction: Variance-based global sensitivity analysis

- Consider a model $y = f(\boldsymbol{x})$ where $y \in \mathbb{R}$, and $\boldsymbol{x} \sim \pi(\boldsymbol{x})$ has independently distributed entries
- Sobol' indices are invaluable tools for GSA which measure the contribution of each input to variance in model output:

$$S_k := \frac{\mathrm{var}[f_k(x_k)]}{\mathrm{var}[f(\boldsymbol{x})]}, \quad S_k^T := 1 - \frac{\mathrm{var}[\mathbb{E}(f(\boldsymbol{x})|x_j,\ j \neq k)]}{\mathrm{var}[f(\boldsymbol{x})]}$$

- $f_k(x_k) := \int f(\boldsymbol{x}) d\boldsymbol{x}_{-k} - \mathbb{E}(f\boldsymbol{x})$, where $d\boldsymbol{x}_{-k}$ denotes integrating over all inputs **except** x_k
- First order Sobol' index S_k measures influence of x_k **outside of** interactions
- Total Sobol' index S_k^T measures influence of x_k **including** interactions with other inputs

Introduction: Computing Sobol' indices

- Monte Carlo (MC) methods generally used to estimate Sobol' indices
- However this is intractable when f is costly to evaluate
- Instead can construct surrogate model $\hat{f} \approx f$ which is cheap to evaluate
- Some surrogate models (e.g. polynomial chaos[2], Gaussian processes[3]) admit analytic formulas for Sobol' indices

[2] B. Sudret. Global sensitivity analysis using polynomial chaos expansions. 2008.

[3] A. Marrel, B. Iooss, B. Laurent, O. Roustant. Calculations of Sobol indices for the gaussian process metamodel. 2009.

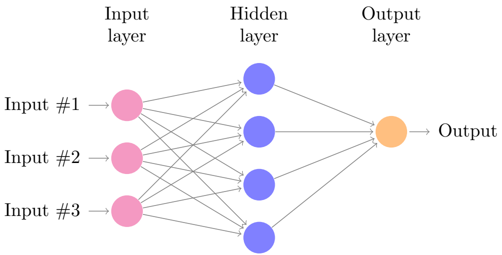
Problem statement

- Let $y = f(\boldsymbol{x})$, where $\boldsymbol{x} \in [0,1]^d$ has independent uniformly distributed entries
- f is computationally expensive to evaluate and/or input dimension d is large
- Can we develop a neural network-based surrogate method which admits analytic formulas for Sobol' indices?

Background: Single layer neural networks

A single layer neural network has the form $\hat{f}(\boldsymbol{x}) = \boldsymbol{\beta}^\top(\phi(\mathbf{W}\boldsymbol{x} + \boldsymbol{b}))$

- $\mathbf{W}$ -hidden layer weight matrix
- $\boldsymbol{b}$ - hidden layer biases
- $\boldsymbol{\beta}$ - output weights
- ϕ - activation function



We train the neural network by solving the nonlinear least squares problem for training points $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_m, y_m)$, where $y_i = \hat{f}(\boldsymbol{x}_i)$

$$\arg\min_{\mathbf{W}, \boldsymbol{b}, \boldsymbol{\beta}} \sum_{i=1}^{m} \left(\hat{f}(\boldsymbol{x}_i; \mathbf{W}, \boldsymbol{b}, \boldsymbol{\beta}) - y_i\right)^2$$

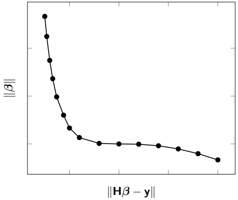
Background: Extreme learning machines

- $\mathbf{W}, \boldsymbol{b}$ independently sampled randomly (e.g. from standard normal distribution)[4]
- Solve the L_2 regularized linear least squares problem to find output weights

$$\arg\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{H}\boldsymbol{\beta} - \mathbf{y}\|_2^2 + \frac{\alpha}{2}\|\boldsymbol{\beta}\|_2^2$$

- $\boldsymbol{y} = [\ y_1 \quad \cdots \quad y_m\ ]^\top$ and $H_{ij} = \phi(\boldsymbol{w}_j^\top \boldsymbol{x}_i + b_j)$
- Computationally quick and easy to use but requires more hidden layer neurons

We determine the regularization parameter α by the L-curve method[5]



[4] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew. Extreme learning machine: Theory and applications. 2006.
[5] P. C. Hansen. Getting Serious: Choosing the Regularization Parameter 2010.

Variance-based GSA with ELMs

- Analytically integrating ELM surrogate should be easy if we want Sobol' index formulas
- Common ML activation functions (e.g. sigmoid) do not make integration easy
- However, activation function can be any smooth non-polynomial function[6]
- Set $\phi(t) = e^t \longrightarrow$ we derive analytic formulas in terms of $\boldsymbol{b}, \mathbf{W}$, and $\boldsymbol{\beta}$
- After training ELM, **obtain Sobol' indices for free**[7]

$$S(\hat{f}) = S(\boldsymbol{b}, \mathbf{W}, \boldsymbol{\beta}), \quad S^T(\hat{f}) = S^T(\boldsymbol{b}, \mathbf{W}, \boldsymbol{\beta})$$

[6] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew. Extreme learning machine: Theory and applications. 2006.

[7] J. Darges, A. Alexanderian, P.A. Gremaud. Extreme learning machines for variance-based global sensitivity analysis. 2022.

Genetic oscillator

- Stiff ODE system (expensive to solve)
- 16 reaction rate parameters are uncertain
- Each parameter uniformly distributed in interval $\pm 5\%$ of respective nominal value
- Study average concentration in time of species R as QoI:

$$f(\boldsymbol{x}) = \frac{1}{T}\int_0^T R(t;\boldsymbol{x})\,dt$$

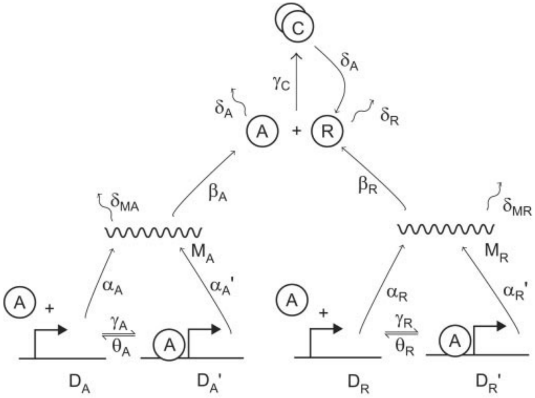
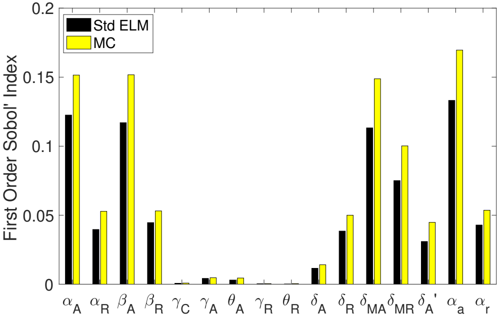


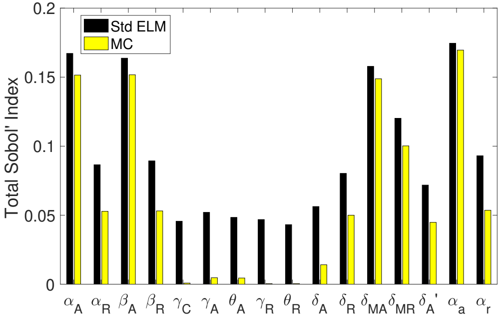
Image credit[8]

[8] J.G. Vilar, H.Y. Kueh, N. Barkai, S. Leibler. Mechanisms of noise-resistance in genetic oscillators. 2002.

GSA for genetic oscillator using ELM surrogate

Experimental setup: 3000 training size, 1000 hidden layers, $\alpha = 10^{-4}$





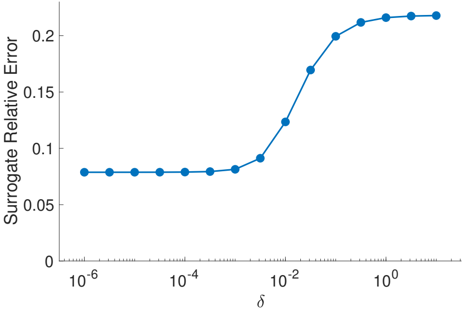
ELM surrogate overestimates higher-order indices compared to MC[9]

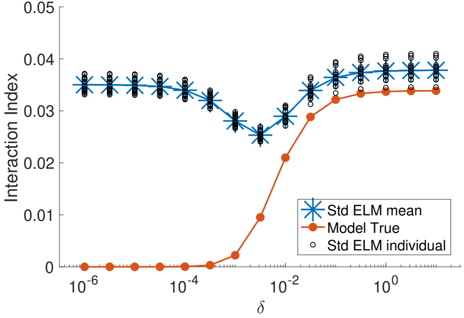
[9] M. Merritt, A. Alexanderian, P.A. Gremaud. Multiscale global sensitivity analysis for stochastic chemical systems. 2021.

ELM and variable interactions

Consider $f_\delta(\boldsymbol{x}) = \sum_{k=1}^{15} x_k + \delta \prod_{j=1}^{d}(1+x_j)$, $\boldsymbol{x} \in [0,1]^{15}$ where δ controls variable interactions

Note: Interaction indices $S_i^{\text{int}} = S_i^T - S_i$ are the same for all inputs



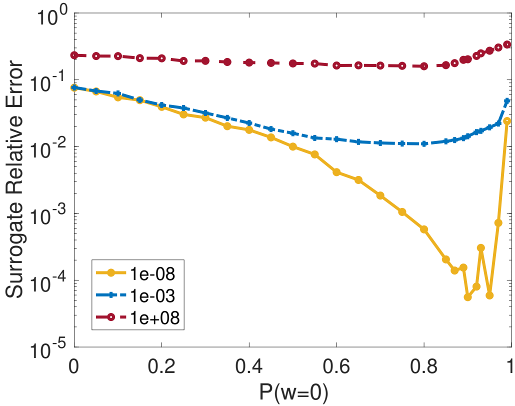


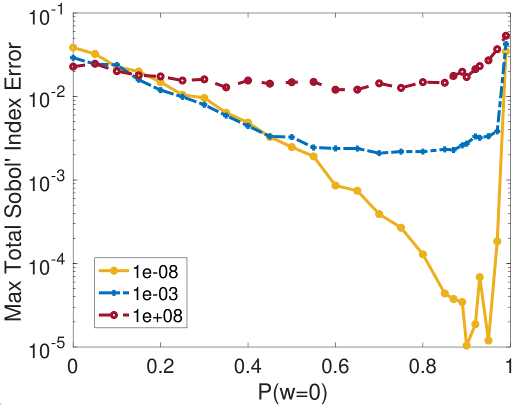
ELM surrogate overestimates higher-order indices when interactions are negligible

Sparsification

- **Issue:** ELM may overestimate higher order Sobol' indices
- Higher order Sobol' indices correspond to influence of interactions
- **Idea:** We can reduce influence of interaction terms by making inner weight matrix sparse
- Sparse weight matrix $\mathbf{W}_s = \mathbf{B} \circ \mathbf{W}$, where $\mathbf{B}_{ij} = \begin{cases} 0 & \text{with probability } p, \\ 1 & \text{with probability } 1 - p \end{cases}$
- How do we know which p to use?

Sparse weight ELM

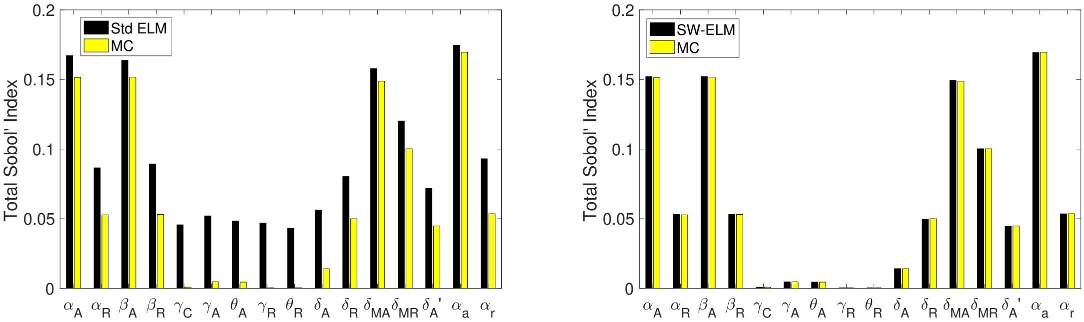




Sparse Weight ELM: Choose p by testing which value gives the best surrogate error on a validation set

GSA for genetic oscillator using SW-ELM

Standard ELM surrogate (left) compared to SW-ELM (right) with $p = 0.9$



Note: SW-ELM also performs well with FAR fewer training points

Summary and future work

- We use ELM as a quick and easy tool for variance-based GSA
- With exponential activation function, we derive analytic expressions of Sobol' indices for uniformly and normally distributed inputs
- After training surrogate, we obtain Sobol' indices for no additional cost
- We developed sparse weight ELM to improve GSA performance without sacrificing speed and simplicity of ELM
- Manuscript submitted[10] to *Reliability Engineering & System Safety*

[10] J. Darges, A. Alexanderian, P.A. Gremaud. Extreme learning machines for variance-based global sensitivity analysis. 2022.

References I