**Conference Paper**

# Artificial Intelligence-Driven Sensitivity Analysis: Present-Day Approaches in Software-Defined Networking

Ezechi, C., Akinsolu, M. O., Sakpere, W., Sangodoyin, A. O., and Akinsolu, F. T.

This is a paper presented at 5th International Conference on Emerging Trends in Electrical, Electronic and Communications Engineering (ELECOM), Balaclava, Mauritius, 20-22 November 2024.

The published version is available at: https://ieeexplore.ieee.org/document/10892185



**Recommended citation:**

Ezechi, C., Akinsolu, M. O., Sakpere, W., Sangodoyin, A. O., and Akinsolu, F. T. (2024), 'Artificial Intelligence-Driven Sensitivity Analysis: Present-Day Approaches in Software-Defined Networking,' 2024 5th International Conference on Emerging Trends in Electrical, Electronic and Communications Engineering (ELECOM), Balaclava, Mauritius, November 20-22, 2024, pp. 1-5, doi: 10.1109/ELECOM63163.2024.10892185.

# Artificial Intelligence-Driven Sensitivity Analysis: Present-Day Approaches in Software-Defined Networking


Chekwube Ezechi
*Lead City University*
Ibadan, Nigeria
chekwube.ezechi@lcu.edu.ng

Mobayode O. Akinsolu
*Wrexham University*
Wrexham, UK
mobayode.akinsolu@wrexham.ac.uk

Wilson Sakpere
*Lead City University*
Ibadan, Nigeria
sakpere.wilson@lcu.edu.ng

Abimbola O. Sangodoyin
*University of Lincoln*
Lincoln, UK
asangodoyin@lincoln.ac.uk

Folahanmi T. Akinsolu
*Lead City University*
Ibadan, Nigeria
akinsolu.folahanmi@lcu.edu.ng



***Abstract*— This paper presents an overview of how artificial intelligence (AI) techniques are being explored for sensitivity analysis in the context of software-defined networking (SDN). Sensitivity analysis (SA) is pivotal in determining the influence of variable inputs on system outputs, a process essential for the enhancement of SDN's performance and dependability. The incorporation of AI techniques, especially machine learning algorithms, has led to substantial progress in executing both local and global sensitivity analyses within SDN frameworks. Such progress is instrumental in improving the network's adaptability, operational efficiency, and security measures. This paper highlights some of the present-day methodologies and practical applications in this area, focusing on the role of AI in refining sensitivity analysis in SDN. The objective is to provide a brief overview of the latest research developments for scholars engaged in this rapidly growing field.**



***Keywords—artificial intelligence (AI), machine learning (ML), sensitivity analysis (SA), and software-defined networking (SDN).***


## I. INTRODUCTION

Artificial intelligence (AI) and software-defined networking (SDN) are transformative technologies reshaping the landscape of computing and network management. AI premises on processing extensive data to extract meaningful insights, proving beneficial in sectors like healthcare, finance, autonomous vehicles, and others [1]. As 5G networks become increasingly prevalent and with the present-day advent of 6G networks, the relevance of AI-driven sensitivity analysis (SA) within SDN has never been more critical. 5G networks, characterized by their ultra-low latency, enhanced connectivity, and higher data rates, rely heavily on sophisticated network management and optimization strategies [2].

Looking ahead, 6G networks are expected to further enhance connectivity with even greater speeds, lower latencies, and advanced capabilities such as holographic communications, ubiquitous connectivity, and integrated AI functionalities [3]. The complexity of 6G networks will necessitate even more advanced AI-driven techniques for network management. The integration of AI within SDN for 6G networks will enable more autonomous and intelligent network operations, pushing the boundaries of current telecommunications technology [4]. SDN primarily enhances network management by separating the control plane from the data plane, improving network flexibility and control [5]. The essence of SDN lies in centralizing network intelligence into controllers, abstracting the network infrastructure from applications, and relegating network devices to mere data packet forwarding [6]. This concept stems from virtualization, which segregates software management from hardware.

Typically, SDN's architecture comprises three layers: the data plane that forms the foundational physical layer, the control plane that offers a virtualized environment, and the management plane that hosts network applications [6]. These layers embody SDN's core principles: centralized intelligence, programmability, and highlevel abstraction [6]. The abstraction layer is particularly vital for deploying AI solutions, machine learning (ML) techniques in particular, as it provides extensive network data and facilitates dynamic modifications [7]. SDN represents a paradigm shift, empowering network administrators to orchestrate network services by abstracting lower-level functions [5]. This abstraction fosters a dynamic and efficient approach to network management, surpassing the capabilities of conventional network architectures.

As networks grow in complexity and size, optimizing performance and security becomes increasingly challenging [8]. SA emerges as a critical tool in this scenario, providing sophisticated methods to bolster SDN's effectiveness [9]. SA does so by pinpointing key parameters and assessing their influence on network performance. A crucial aspect of model development and evaluation involves a thorough delineation and comprehension of how variations in model parameters affect predictions. SA is a vital step in this process, examining the significance of input parameters on the model's output and quantifying the impact of each input on the overall results [10].

Traditionally, SA involves analyzing the relationships between various sources of uncertainty in the inputs to models and the uncertainties in the outputs of models (numerical or otherwise) [11]. While closely related to uncertainty analysis (UA), SA is distinct in that UA focuses on characterizing uncertainties in the model's predictiveness without pinpointing which assumptions contribute most to the uncertainties [9]. Ideally, UA should precede SA, as uncertainties need to be estimated before they can be apportioned [12]. However, this sequence is not always necessary, and model optimization applications may not require the quantification of uncertainty [12].

The literature offers various categorizations for SA techniques. For instance, [13] and [14] classify SA techniques based on their scope as local or global SA and based on their framework as deterministic or statistical SA. In [15], [16], and [17], statistical frameworks for SA are generally assumed to stem from the design of experiments, with classifications according to the parameter space of interest to include both local and global SA methods. Consequently, this work also categorizes SA methods into local and global SA methods. Given the increasing convergence of AI and SDN, numerous practical approaches have recently been proposed to implement AI-driven SA in this context. These AI-driven techniques have demonstrated significant advantages in the characterization, robustness, security, and performance optimization of SDN environments [18]. In this paper, some of the contemporary approaches reported in recent literature are discussed, aiming to highlight their practical applications and real-world effectiveness.

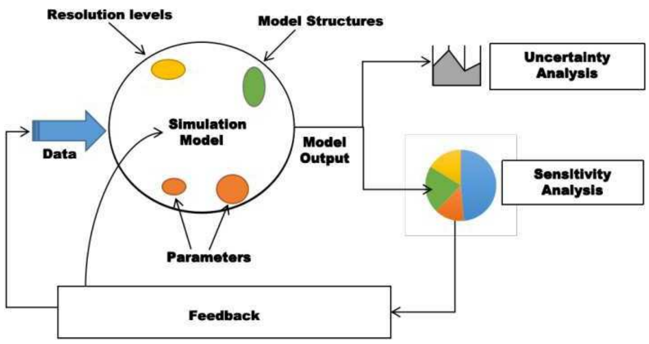


Figure 1. Illustration of a typical sensitivity analysis implementation.

## II. OVERVIEW OF SA

The origins of SA date back to scientific and engineering studies in the early 20th century, where it was employed to assess the reliability and performance of systems under varying conditions [19]. With advancements in computational methods, SA has become an integral component of modern analytical practices. In many applications, the main purpose of SA is to ascertain, under a given set of assumptions, how various values of independent variables affect a specified dependent variable [10]. This approach helps identify the most influential variables in determining the behavior of complex systems, making SA an essential technique for understanding and improving models, and thus invaluable in decision-making processes [20].

SA methodologies can be broadly classified into two categories: local and global [20]. Local SA (LSA) examines the effects of small deviations around a specific input benchmark, whereas global SA (GSA) evaluates the impacts across the entire spectrum of input possibilities, thereby providing a more comprehensive perspective on the dynamics between inputs and outputs [20]. SA involves quantifying output uncertainty attributable to different sources of input uncertainty [20]. For instance, Figure 1 [12], provides a basic illustration of the interaction between input variables and output responses, laying the groundwork for a standard SA procedure.

In Figure 1, the various sources of uncertainty are processed through the model, resulting in an empirical output distribution (represented by the grey curve). The variability of the model's output, as indicated by its variance, can then be analyzed according to its sources, facilitating a thorough SA. LSA and GSA are further discussed as follows:

### *A. LSA*

By concentrating on the sensitivity bothering on a set of parametric measures, LSA evaluates the local influence of input factor changes on a model's response [21], [22]. When examining the sensitivity of an input factor in LSA, the other parametric measures are typically fixed. In many cases, LSA is implemented by estimating how small perturbations around nominal input parametric values affect the model output [23]. This disturbance is often implemented on a single parameter sequentially, roughly estimating the first-order partial derivative of the model's output concerning the perturbed parameter, as demonstrated in [24].

As reported in [25], this derivative can be estimated using effective adjoint methods that handle various parameters. Regardless of the technique used, sensitivity coefficients derived from deterministic LSA have the benefit of being intuitive to comprehend in addition to being numerically efficient [26]. Their equivalency to the derivative of the output concerning each parameter around a predefined point (i.e., nominal parameter values) is primarily responsible for their intuitiveness. As a result, these parametric coefficients, regardless of the range of parameter variations as exemplified in [13], may be easily compared across many modelled systems. To evaluate the sensitivity for a certain fixed set of input parametric measures or values, LSA may additionally entail the use of the partial derivatives of the model's response to the input parameters [27], [28].

Considering the model, $F$, for the system described as follows:

$$y = F(x, \gamma) \tag{1}$$

The dependent variable **y** is influenced by the independent variation of x and the parameters $\gamma = [\gamma_1, ..., \gamma_r]$ of F, as indicated by the LSA. The primary idea of LSA is based on calculating the impact of model parameter $\gamma$ or pattern characteristics $xi$, i = 1,..., N on the output value $y_j$, j= 1,..., N, where N and J stand for the number of outputs and features, respectively, following a training procedure [24],[25]. The actual coefficients $S_{ji}$ characterise this influence.

$$S_{j,i}^{(p)} = \frac{a y_j(x_1^{(p)}, x_2^{(p)}, ..., x_N^{(p)})}{a y_j} \tag{2}$$

Based on the $p$th training pattern, $p = 1, ..., P$, Equation (2) gives the sensitivity value of the $j$th neural network output signal on the ith attribute of the input vector **x**. If the inputs to the model act additively or linearly, or if there is little uncertainty in the inputs, the LSA technique may be useful [26].

### *B. GSA*

GSA studies how various sources of input uncertainty can be attributed to uncertainty in a model's output [11]. By thoroughly characterizing the output response surface that is Figure 1. Illustration of a typical sensitivity analysis implementation. produced, GSA aims to investigate the complete input parameter space and determine the relative importance of each input. This characterization often concentrates on locating the system's crucial points, such as saddle points, minima, and maxima, within a global deterministic framework [13], [25]. GSA aids in comprehending the extremes of the system's behavior under many circumstances by identifying these crucial areas. Some statistical methods for GSA ([16], [20]), examine the model output's variance [32], [33], correlation [34], or elementary effects [34] to ascertain level of spread and relationships.

GSA methods also provide a statistical characterization of the output response surface, offering insights into how input uncertainties propagate through the model [15], [16]. For example, variance-based methods quantify how much of the output variance can be attributed to each input parameter [32], [33], while correlation-based methods assess the strength and direction of linear relationships between inputs and outputs [34]. Elementary effects methods evaluate the sensitivity of the output by systematically varying one input parameter at a time across its entire range and observing the effects [35]. GSA studies the effects of changing parametric measures uniformly and concurrently throughout their entire spectrum [15], [16]. This holistic approach ensures that GSA provides a comprehensive understanding of how input uncertainties influence the model's behavior, making it a powerful tool for identifying key drivers of variability and for improving the robustness and reliability of complex models.

Because different GSA methods have different characterizations, they can produce false results that are not comparable to results from LSA approaches [26]. This is because GSA methods do not have a uniform definition for sensitivity coefficients. So, the use of different GSA techniques may occasionally result in contradictory and inconsistent parameter significance rankings [16]. Moreover, the results of GSA may be significantly influenced by the input parameters' range of variation and the assumed probability distribution of those values [14], [25]. Despite these challenges, GSA effectively handles both linear and nonlinear responses, and can reveal the relationships between multiple input parameters [36]. A commonly used method for implementing GSA is the variance-based SA (Sobol' method), which breaks down the model output's variance into fractions that can be linked to individual inputs or groups of inputs [20]. The total variance $V(Y)$ of the model output $Y$ in the Sobol' method is often expressed as:

$$V(Y) = \sum_{i=1}^{k} V_i + \sum_{i<j} V_{ij} + \sum_{i<j<l} V_{ijl} + \ldots + V_{12\ldots k} \tag{3}$$

where $V_i$ is the variance ascribed to the *i-th* input alone, $V_{ij}$ is the variance that can be ascribed to the interaction between the *i-th* and *j-th* inputs, $V_{ijl}$ is the variance that can be ascribed to the interaction among the *i-th*, *j-th*, and *l-th* inputs, and $V_{12\ldots k}$ is the variance ascribed to the interaction among all $k$ inputs.

## III. AI-DRIVEN SA IN SDN

Within the framework of SDN, SA is an indispensable tool for network administrators, helping to elucidate the effects that various network parameters and configurations have on network performance [24]. These insights are crucial for refining network operations, resolving technical issues, and making strategic decisions regarding network enhancements and modifications [24]. The impact of AI, ML techniques in particular, has been significant across various domains, including SDN. In SDN, LSA and GSA are essential for understanding how different parameters influence network performance [37]. AI-driven sensitivity analysis, leveraging ML techniques, offers a more robust and comprehensive understanding of these influences. This section summarily discusses some of the recent approaches for ML-assisted LSA and GSA in SDN.

### *A. ML-assisted LSA in SDN*

One common approach for ML-assisted LSA involves using gradient-based methods [38]. These methods compute the gradient of a performance metric with respect to a particular parameter. Techniques such as gradient boosting machines (GBMs) and neural networks are frequently employed to approximate these gradients efficiently [38], [39]. GBMs can model the relationship between network parameters and performance metrics in SDN environments, allowing network administrators to identify which parameters have the most significant local impact on the network performance [38]. Similarly, neural networks, particularly those utilizing backpropagation, can be employed to estimate the gradients of performance metrics resulting from input parameters [39]. This enables precise LSA and assists in fine-tuning individual network settings in SDN environments. When training neural networks, several kinds of batch training algorithms are employed, each having unique properties and performance in terms of speed, accuracy, and memory requirements [24]. Levenberg-Marquardt is considered one of the most efficient training algorithms for training artificial neural networks (ANNs) due to its speed and precision, but it requires substantial computational memory, which can be a limitation for certain applications [40], [41].

The work carried out in [24], adopted the Levenberg-Marquardt algorithm to train an ANN to assist in the evaluation of the sensitivity of SDN performance metrics, specifically throughput, jitter, and response time, under various forms of distributed denial of service (DDoS) attacks. The evaluation involved analyzing the distinctions between predicted target values and actual target values of the ANN model when additive white Gaussian noise was added to the SDN performance metrics, severally, showing that the SDN performance metrics are all sensitive to DDoS attacks. The work in [24] and other similar works demonstrate that Levenberg-Marquardt algorithm's balanced trade-off between speed and memory usage will be a benefit to 5G and future 6G's higher data rates and ultra reliable low latency [42]. Increased data rates, reduced latency, and enhanced capacity of 5G and future 6G networks require efficient algorithms for real-time data processing such as the implementation of SA involving network parameters in SDN environments [43].

Another technique used in LSA is perturbation analysis, which involves slightly altering a single parameter and observing the resulting change in performance metrics [44]. This approach, when combined with ML models, can provide detailed insights into local sensitivity [44]. Support vector machines (SVMs), for example, can be used to model the boundary conditions of network performance [44]. By perturbing input parameters and observing the changes, one can determine the local sensitivity as reported in [44]. Additionally, feature importance scores in tree-based ML models like Random Forests and GBMs can also indicate how changes in specific parameters affect the model's predictions [45]. By examining these feature importance scores, network administrators can identify which parameters are most influential locally, helping them prioritize adjustments for optimal performance in SDN environments [45].

### *B. ML-assisted GSA in SDN*

GSA typically works by examining the effect of varying all parameters simultaneously across their entire range [10]. In SDN, this robust approach is essential for understanding the overall behavior of the network and for identifying the most critical parameters that influence the performance of the network. By exploring the full parametric space, GSA provides a holistic view of how different inputs affect the network, enabling more informed decision-making and optimization strategies in SDN environments. Variance-based methods such as the Sobol' method are a prominent category within GSA. These methods mainly work by decomposing the variance of the output performance metrics into contributions from each input parameter [46], and they can be significantly enhanced by ML models, which provide accurate approximations of the variance components. For instance, Sobol' indices that are used to quantify the contribution of each parameter to the output variance can be can efficiently computed by ML models such as neural networks and GBMs, offering valuable insights [46].

Monte Carlo simulations also play a crucial role in GSA [47], and their efficiency can be improved with ML techniques. By training models on a subset of data, predictions can be made across a larger parameter space, facilitating a more extensive GSA. Gaussian process, in particular, can model the distribution of network performance metrics across different parametric settings in this context [47]. Running Monte Carlo simulations on such models provides a comprehensive view of GSA [47]. Additionally, such ML models can serve as surrogate models to approximate complex network behaviors for implementing GSA in SDN [48]. Kriging or Gaussian Process Regression is a well-known effective surrogate modeling technique used in this way to provide accurate approximations of the network's performance landscape [48].
This approach allows for extensive exploration of the parametric space with reduced computational effort, making it highly suitable for GSA in SDN environments as carried out in [48].

A recent fairly approach in GSA is presented in [49], where exploratory data analysis (EDA) is employed to describe and picture disparities in SDN performance metrics across emulated scenarios, alongside linear regression to infer the sensitivity of these SDN performance metrics to these scenarios. Experimental results reveal that the SDN performance metrics fluctuate with changes in the SDN scenarios relative to various DDoS attacks, indicating their sensitivity to the attack scenarios with some interactions between them. The work in [49] strongly suggests that ML-assisted GSA can help in identifying critical parameters that influence reliability and performance in SDN environments. By understanding the global sensitivity of network performance to various inputs, network administrators can prioritize resources and configurations that maximize efficiency and reliability [50]. For present-day 5G networks and future 6G networks, ML-assisted GSA will be crucial in managing the increased complexity and ensuring that the networks can meet the stringent requirements of new applications [51].

## IV. CONCLUSION

The integration of AI, specifically ML techniques, into SA for SDN represents a significant advancement in network management. ML techniques enhance both local and global SA, offering more robust understanding of network behavior and enabling better optimization strategies. By leveraging AI, network administrators can better understand the intricate dynamics of network parameters, leading to improved fine-tuning of specific components and more robust fault detection and resolution mechanisms in SDN environments. As SDN continues to evolve, AI-driven SA will become increasingly vital in ensuring network reliability, efficiency, and security. The ability to precisely analyze the impact of various parameters allows for proactive management and swift adaptation to changing network conditions. This not only enhances performance but also fortifies the network against potential vulnerabilities.

The fusion of AI with SDN's flexible architecture promises to unlock new levels of operational excellence. Some of the recent works in this domain have been highlighted in this paper. Future research should focus on developing more sophisticated AI models and exploring their applications across diverse SDN environments. This will help fully realize the potential of AI driven SA. By advancing these technologies, more intelligent and adaptive network management solutions that will drive the next generation of SDN innovations can be anticipated. The ongoing evolution of AI and SDN holds tremendous promise for creating networks that are not only more efficient and reliable but also capable of meeting the complex demands of modern and future digital ecosystems such as 5G and 6G networks.

## REFERENCES


[1] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Pearson, 2020.

[2] C. Benzaid and T. Taleb, "AI-Driven Zero Touch Network and Service Management in 5G and Beyond: Challenges and Research Directions," *IEEE Network*, vol. 34, no. 2, pp. 186-194, Mar./Apr. 2020.

[3] Wang *et al.*, "On the Road to 6G: Visions, Requirements, Key Technologies and Testbeds," *IEEE Communications Surveys & Tutorials*, vol. PP, no.1, pp. 1-1, 2023.

[4] A. Alhammadi *et al.*, "Artificial Intelligence in 6G Wireless Networks: Opportunities, Applications, and Challenges," *International Journal of Intelligent Systems*, 2024.

[5] D. Kreutz, F. M. V. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14- 76, 2015.

[6] T. A. Nguyen *et al.*, "Availability Modeling and Analysis for Software Defined Networks," in *Proceedings of the IEEE Pacific Rim Dependable Computing Conference (PRDC)*, 2015, pp. 27.

[7] A. Guo and C. Yuan, "Network Intelligent Control and Traffic Optimization Based on SDN and Artificial Intelligence," Electronic, vol. 10, no .6, p. 700, 2021

[8] K. N. Qureshi *et al.*, "A distributed software defined networking model to improve the scalability and quality of services for flexible green energy internet for smart grid systems," *Computers & Electrical Engineering*, vol. 84, pp. 106634, 2020.

[9] T. G. Trucano et al., "Calibration, validation, and sensitivity analysis: What's what," *Reliability Engineering & System Safety*, vol. 91, pp.1331-1357, 2006.

[10] B. Iooss and P. Lemaître, "A Review on Global Sensitivity Analysis Methods," in *Handbook of Uncertainty Quantification*, R. Ghanem, D. Higdon, and H. Owhadi, Eds. Springer, 2014, pp. 101-122.

[11] A. Saltelli, "Sensitivity analysis for importance assessment," *Risk Analysis*, vol. 22, pp. 579-590, 2002.

[12] A. Saltelli et al., "Why So Many Published Sensitivity Analyses Are False. A Systematic Review of Sensitivity Analysis Practices," *Environmental Modelling and Software*, vol. 114, pp. 29-39, 2019.

[13] M. Ionescu-Bujor and D. G. Cacuci, "A Comparative Review of Sensitivity and Uncertainty Analysis of Large-Scale Systems - I: Deterministic Methods," *Nuclear Science and Engineering*, vol. 147, pp. 189-203, 2004.

[14] D. G. Cacuci and M. Ionescu-Bujor, "A Comparative Review of Sensitivity and Uncertainty Analysis of Large-Scale Systems - II: Statistical Methods," *Nuclear Science and Engineering*, vol. 147, pp. 204- 217, 2004.

[15] B. Iooss and P. Lemaitre, "A Review on Global Sensitivity Analysis Methods," in *Uncertainty Management in Simulation-Optimization of Complex Systems*, A. Dellino and C. Meloni, Eds. Springer, 2015, pp. 101-122.

[16] A. Saltelli *et al.*, *Global Sensitivity Analysis. The Primer*. West Sussex, UK: John Wiley & Sons, 2008.

[17] T. Santner, B. Williams, and W. Notz, *The Design and Analysis of Computer Experiments*. New York, NY, USA: Springer, 2003.

[18] U. Umoga *et al.*, "Exploring the potential of AI-driven optimization in enhancing network performance and efficiency," *Magna Scientia Advanced Research and Reviews*, vol. 10, pp. 368-378, 2024.

[19] S. Razavi et al., "The Future of Sensitivity Analysis: An Essential Discipline for Systems Modeling and Policy Support," *Environmental Modelling & Software*, vol. 137, p. 104954, 2020.

[20] A. Saltelli et al., Sensitivity analysis in practice: A guide to assessing scientific models. Wiley, 2004.

[21] X. Zhou and H. Lin, "Local sensitivity analysis," in Springer International Publishing, Cham, pp. 1130–1131.

[22] J. M. Zurada, A. Malinowski, and I. Cloete, "Sensitivity analysis for minimization of input data dimension for feedforward neural network," in *Proceedings of IEEE International Symposium on Circuits and Systems-ISCAS94*, 1994, vol. 6, pp. 447–450.

[23] C. Rojas et al., "A critical look at efficient parameter estimation methodologies of electrochemical models for Lithium-Ion cells," *Journal of Energy Storage*, vol. 80, p. 110384, 2024.

[24] A. O. Sangodoyin *et al.*, "A deductive approach for the sensitivity analysis of software-defined network parameters," *Simulation Modelling Practice and Theory*, vol. 103, no. 102099, pp. 1-15, 2020.

[25] D. G. Cacuci and M. Ionescu-Bujor, "Sensitivity and uncertainty analysis, data assimilation, and predictive best-estimate model calibration," in *Handbook of Nuclear Engineering*, D. G. Cacuci, Ed. Springer, 2010, pp. 1913-2051.

[26] S. Razavi and H. V. Gupta, "What do we mean by sensitivity analysis? The need for comprehensive characterization of 'global' sensitivity in Earth and Environmental systems models," *Water Resources Research*, vol. 51, pp. 3070-3092, 2015.

[27] A. Hunter, L. Kennedy, J. Henry, and I. Ferguson, "Application of neural networks and sensitivity analysis to improved prediction of trauma survival," *Computer Methods and Programs in Biomedicine*, vol. 62, no. 1, pp. 11–19, 2000.

[28] J. Kirch *et al.*, "The effect of model rescaling and normalization on sensitivity analysis on an example of a MAPK pathway model," *EPJ Nonlinear Biomedical Physics*, vol. 4, 2016.

[29] J. M. Zurada, A. Malinowski, and S. Usui, "Perturbation method for deleting redundant inputs of perceptron networks," *Neurocomputing*, vol. 14, no. 2, pp. 177–193, 1997.

[30] P. A. Kowalski and M. Kusy, "Sensitivity analysis for probabilistic neural network structure reduction," I*EEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 5, pp. 1919–1932, 2017.

[31] A. Saltelli and P. Annoni, "How to avoid a perfunctory sensitivity analysis," *Environmental Modelling & Software*, vol. 25, no. 12, pp. 1508–1517, 2010.

[32] I. M. Sobol, "Global sensitivity analysis for nonlinear mathematical models and their Monte Carlo estimates," *Mathematics and Computers in Simulation*, vol. 55, no. 1-3, pp. 271-280, 2001.

[33] R. Cukier, H. Levine, and K. Shuler, "Nonlinear sensitivity analysis of multiparameter model systems," Journal of Computational Physics, vol. 26, pp. 1-42, 1978.

[34] J. C. Helton, "Uncertainty and sensitivity analysis techniques for use in performance assessment for radioactive waste disposal," *Reliability Engineering & System Safety*, vol. 42, pp. 327-367, 1993.

[35] M. D. Morris, "Factorial sampling plans for preliminary computational experiments," *Technometrics*, vol. 33, no. 2, pp. 161-174, 1991.

[36] K. G. Link et al., "A local and global sensitivity analysis of a mathematical model of coagulation and platelet deposition under flow," *PLoS One*, vol. 13, no. 7, pp. e0200917, 2018.

[37] C. Qin *et al.*, "Comparative Study of Global Sensitivity Analysis and Local Sensitivity Analysis in Power System Parameter Identification," *Energies*, vol. 16, p. 5915, 2023.

[38] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of Statistics*, pp. 1189-1232, 2001.

[39] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[40] M. T. Hagan and M. B. Menhaj, "Training feedforward networks with the marquardt algorithm," *IEEE Transactions on Neural Networks*, vol. 5, no. 6, pp. 989-993, 1994.

[41] H. Yu and B.M. Wilamowski, Levenberg-marquardt Training, *Intelligent Systems*. CRC Press, 2018. 12-1.

[42] Y. Liu et al., "Machine Learning for 6G Enhanced Ultra-Reliable and Low-Latency Services," *IEEE Wireless Communications*, vol. 30, no. 2, pp. 48-54, 2023.

[43] I. Salah et al., "Comparative Study of Efficiency Enhancement Technologies in 5G Networks - A Survey," *Procedia Computer Science*, vol. 182, pp. 150-158, 2021.

[44] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995.

[45] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5- 32, 2001.

[46] I. M. Sobol', "Sensitivity analysis for nonlinear mathematical models," *Mathematical Modeling and Computational Experiment*, vol. 1, no. 4, pp. 407-414, 1993.

[47] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2006.

[48] T. J. Santner, B. J. Williams, and W. I. Notz, *The Design and Analysis of Computer Experiments*, Springer, 2018.

[49] M. O. Akinsolu *et al.*, "Behavioral Study of Software-Defined Network Parameters Using Exploratory Data Analysis and Regression-Based Sensitivity Analysis," Mathematics, vol. 10, pp. 2536, 2022.

[50] N. Khan, "5G Network: Techniques to Increase Quality of Service and Quality of Experience," *International Journal of Computer Networks and Applications*, vol. 9, pp. 476-496, 2022.

[51] X. You *et al.*, "Towards 6G Wireless Communication Networks: Vision, Enabling Technologies, and New Paradigm Shifts," *Science China Information Sciences*, vol. 64, pp. 1-74, 2021.