Cutting Through the Noise: Using Deep Neural Network Metamodels for High Dimensional Nested Simulation

Tony Wirjanto Mingbin Feng Xintong Li

Abstract

Deep neural network models have gained great success in many applications, but their adoption in financial and actuarial applications has been received by regulators with some trepidation. The lack of transparency and interpretability of these models leads to skepticism about their resilience and reliability, which are important factors to ensure financial stability and insurance benefit fulfillment. In this study, we use stochastic simulation as a data generator to examine neural network models under controlled settings. Our study shows interesting findings in fundamental questions like "What do neural network models learn from noisy data?" and "How well do they learn from noisy data?", and the findings provides justifications for using deep neural networks in actuarial applications. Based on our findings, we propose an efficient nested simulation procedure that uses neural network models as metamodels to estimate risk measures of hedging errors for variable annuities. The proposed procedures use neural networks as metamodels to provide accurate loss predictions and concentrate simulation budget on tail scenarios while maintaining transparency in the estimation step, and we provide practical guidelines to extend our generic procedures for other financial and actuarial applications.

1 Introduction

Deep neural networks (Hastie et al., 2009; LeCun et al., 2015) have attracted attentions of researchers and practitioners due to their success in solving real-world supervised learning tasks such as AlphaGo (Silver et al., 2016) and ChatGPT (OpenAI, 2023). Since the first artificial neural network model (McCulloch and Pitts, 1943) and the first algorithm for training a perceptron (Rosenblatt, 1958), especially after the introduction of backpropagation (Rumelhart et al., 1985) and the growth of high-performance computing, the field of artificial neural network and deep learning in general has grown rapidly. Two specialized neural network architectures that are relevant to our study are recurrent neural networks (RNNs) (Williams and Zipser, 1989; Sutskever et al., 2014) and long-short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997; Chung et al., 2014), as we need to train metamodels that take sequential observations as input. Despite their success, deep neural network models are often criticized for their lack of transparency and interpretability, which hinders their adoption in financial and actuarial applications. Enormous research efforts are spent to test and improve the robustness of neural network models with carefully designed noise injection methods. Poole et al. (2014) show that injecting synthetic noise before and after hidden unit activations during training improves the performance of autoencoders. Neelakantan et al. (2015) improve learning for deep neural networks by injecting synthetic noise to the gradients during backpropagations. A branch of research has been devoted to understanding the resilience of neural network models to noise in training labels. For example, Luo et al. (2016) show that adding synthetic label noise to the convolutional neural network(CNN) can improve its ability to capture global features. Srivastava et al. (2014) quantify the error tolerance by injecting synthetic label noise with a custom Boltzmann machine hardware. Szegedy et al. (2013) find that neural networks are vulnerable to adversarial examples, and Goodfellow et al. (2014) design an efficient method to generate such noisy examples to exploit the vulnerability to adversarial perturbations. Carlini and Wagner (2017) design targeted attacks to training labels to test the robustness of neural networks. Instead of using synthetic noise, Jiang et al. (2020) inject real-world label noise and examine noise tolerance of neural networks with controlled noise levels. The aforementioned studies use real-world data, as is typically the case for many neural network studies, where noise is already present in the training labels before any noise injection. Users of real-world data have little control over the noise level of the original training labels and usually examine the effect of noisy data by injecting noise, but it is unclear whether the neural network model trained on noisy data actually learns the real, i.e., noiseless, feature-label relationship. Due to their lack of transparency and interpretability, the adoption of neural networks in financial and actuarial applications has been received by regulators with some skepticism.

The contributions of our study are two-fold:

1. We study what neural networks learn from noisy data by training them using simulated data based on well-designed simulation experiments. This is a novel way to study the effect of noisy data and error tolerance of neural network models as one can *reduce noise* in the data by increasing the number of replications in a simulation model. This new way of studying neural network models can provide more direct evidence on their transparency and interpretability.

2. We propose two generic nested simulation procedures that uses neural networks as metamodels to improve its efficiency while maintaining transparency. In essence, a pilot stage simulation is used to generate a large number of noisy data, which are then used to train a neural network metamodel. Depending on the application, the trained metamodel can serve two purposes: (1) to identify a set of tail scenarios, and (2) to estimate risk measures directly. The first procedure uses a neural network to identify a set of potential tail scenarios on which computations are performed in the second stage, while the second procedure uses neural network predictions to estimate risk measures directly. Our numerical results show that neural network metamodels can identify the tail scenarios accurately and so the proposed procedures can estimate tail risk measures with similar accuracy while, at the same time, using less simulation budget.

We are curious about fundamental questions like "What do deep neural networks learn from noisy data?" and "How well do neural networks learn from noisy data?". Data-driven answers to these questions prevail in the existing literature. In supervised learning, deep neural networks are believed to learn from the given data about the feature-label relationship to predict new labels for unseen features. Cross validation using to assess a subset, i.e., the validation set, of the original data, is a common way to access the quality of learning. Generalization error on the test labels is another popular assessment metric. But the test set is also a subset of the original data. In this study, we revisit these questions in a simulation context and propose an alternative approach to answer them. Instead of relying solely on real-data (splitting it into multiple subsets), we propose using stochastic simulation outputs as training labels for deep neural network models. By controlling the simulation design parameters, such as the number of independent replications, we can control the quality (and also the quantity) of the training labels fed into the neural networks. In such a controlled environment, we obtain more clear-cut answers to the above fundamental questions.

In nested simulation, a simulation model is used to generate a large number of outer scenarios, and each scenario is then used as an input to another simulation model. Borrowing terminologies from machine learning research, we can view a set of simulated outer scenarios and the estimated hedging errors for those scenarios as the *features* and (noisy) *labels*. One can train supervised learning models using these simulated features and labels. They are then used to replace the time-consuming inner simulations by the trained model. We refer to the trained supervised learning models as *metamodels* of the inner simulation, which is also known as the *surrogate models*. Metamodeling is a popular approach to reduce the computational burden of simulation-based applications by replacing the time-consuming simulation with a metamodel. The metamodel is trained using a set of simulated data, and it is used to predict the simulation output for new inputs. The study of metamodeling is an active research area in the simulation literature, and using deep neural networks as metamodels is a relatively new development. Lieu et al. (2022) use deep neural networks as metamodels of a simulation model for structural reliability analysis, and Salle and Yıldızoğlu (2014) show that neural network metamodels help achieve higher prediction accuracy that other metamodels in approximating agent-based simulation models. A popular metamodel in nested simulation procedures is stochastic kriging. Liu and Staum (2010) use stochastic kriging as a metamodel of Monte Carlo simulations to estimate the conditional value at risk (CVaR) of a portfolio of derivative securities, and Gan and Lin (2015) use stochastic kriging for an efficient valuation of large portfolios of variable annuity (VA) contracts. Other studies, such as Broadie et al. (2015), Hong et al. (2017), and Zhang et al. (2022) use regression, kernel smoothing, and the likelihood ratio method, respectively. Our study has three key distinctions over the existing ones:

1. our metamodel has high-dimensional inputs. In machine learning terminology, the features are high-dimensional vectors. To estimate the hedging error of a typical VA contract, the number of features is in the order of hundreds, which is at least one order of magnitude larger than the number of features in the aforementioned studies,

2. for estimating tail risk measures, our metamodel is only used for tail scenario identification but is *not* used in the estimation of the tail risk measures. This is a feature designed particularly to convince regulators that the losses used in estimating the risk measure are based on a transparent inner simulation model rather than on some black-box metamodels, and
3. using simulation models as data generators, we can decrease the noise level and get arbitrarily close to the true labels by increasing the number of replications in the simulation model. This design allows a systematic study of the effect of noisy training labels on the performance of neural network models in predicting the noiseless labels.

The rest of this article is organized as follows: Section 2 presents the problem settings for tail risk measures and dynamic hedging of VAs. Section 3 proposes an efficient two-stage nested simulation procedure that uses machine learning models as metamodels to help reduce simulation budget by only performing computations on identified tail scenarios. Section 4 proposes a one-stage nested simulation procedure that estimates risk measures directly with metamodel predictions. Section 5 demonstrates the efficiency of deep learning proxies and examines error tolerance of two LSTM proxy models with different numbers of trainable parameters. Practical suggestions are provided for the choice of suitable metamodels and simulation settings.

2 Problem Formulation

3 Two-Stage Nested Simulation with Deep Neural Network Metamodels

4 One-Stage Nested Simulation with Deep Neural Network Metamodels

5 Numerical Results

6 Conclusion

References

Mark Broadie, Yiping Du, and Ciamac C Moallemi. Risk estimation via regression. *Operations Research*, 63(5):1077–1097, 2015.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. Ieee, 2017.

Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. 2014. `https://arxiv.org/abs/1412.3555`. Accessed 7th Sep 2023.

Guojun Gan and Sheldon X Lin. Valuation of large variable annuity portfolios under nested simulation: A functional data approach. *Insurance: Mathematics and Economics*, 62:138–150, 2015.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. `https://arxiv.org/abs/1412.6572`. Accessed 1st Jan 2024.

Trevor Hastie, Robert Tibshirani, and Jerome H Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, volume 2. Springer, 2009.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

Jeff L Hong, Sandeep Juneja, and Guangwu Liu. Kernel smoothing for nested estimation with application to portfolio risk measurement. *Operations Research*, 65(3):657–673, 2017.

Lu Jiang, Di Huang, Mason Liu, and Weilong Yang. Beyond synthetic noise: Deep learning on controlled noisy labels. In *International conference on machine learning*, pages 4804–4815. PMLR, 2020.

Yann LeCun, Yoshua Bengio, and Geoffrey E Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

Qui X Lieu, Khoa T Nguyen, Khanh D Dang, Seunghye Lee, Joowon Kang, and Jaehong Lee. An adaptive surrogate model to structural reliability analysis using deep neural network. *Expert Systems with Applications*, 189:116104, 2022.

Ming Liu and Jeremy Staum. Stochastic kriging for efficient nested simulation of expected shortfall. *Journal of Risk*, 12(3):3, 2010.

Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 29, 2016.

Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5:115–133, 1943.

Arvind Neelakantan, Luke Vilnis, Quoc V Le, Ilya Sutskever, Lukasz Kaiser, Karol Kurach, and James Martens. Adding gradient noise improves learning for very deep networks. *arXiv preprint arXiv:1511.06807*, 2015. `https://arxiv.org/abs/1511.06807`. Accessed 1st Jan 2024.

OpenAI. Chatgpt, 2023. `https://chat.openai.com/chat`. Accessed 7th Sep 2023.

Ben Poole, Jascha Sohl-Dickstein, and Surya Ganguli. Analyzing noise in autoencoders and deep networks. *arXiv preprint arXiv:1406.1831*, 2014. `https://arxiv.org/abs/1406.1831`. Accessed 1st Jan 2024.

Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386, 1958.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California University San Diego La Jolla Institute for Cognitive Science, 1985.

Isabelle Salle and Murat Yıldızoğlu. Efficient eampling and meta-modeling for computational economic models. *Computational Economics*, 44:507–536, 2014.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL `https://proceedings.neurips.cc/paper_files/paper/2014/file/a14ac55a4f27472c5d894ec1c3c743d2-Paper.p`

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. `https://arxiv.org/abs/1312.6199`. Accessed 1st Jan 2024.

Ronald J Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1(2):270–280, 1989.

Kun Zhang, Mingbin Ben Feng, Guangwu Liu, and Shiyu Wang. Sample recycling for nested simulation with application in portfolio risk measurement. 2022. `https://arxiv.org/abs/2203.15929`. Accessed 2nd May 2023.