Normality Testing for Vectors on Perceptron Layers

Youmna Karaki, Halina Kaubasa, and Nick Ivanov

***Abstract*—Designing optimal topology of network graph is one of the most prevalent issues in neural network applications. Number of hidden layers, number of nodes in layers, activation functions, and other parameters of neural networks must suit the given data set and the prevailing problem. Massive learning datasets prompt a researcher to exploit probability methods in an attempt to find optimal structure of a neural network. Classic Bayesian estimation of network hyperparameters assumes distribution of specific random parameters to be Gaussian. Multivariate Normality Analysis methods are widespread in contemporary applied mathematics. In this article, the normality of probability distribution of vectors on perceptron layers was examined by the Multivariate Normality Test. Ten datasets from University of California, Irvine were selected for the computing experiment. The result of our hypothesis on Gaussian distribution is negative, ensuring that none of the set of vectors passed the criteria of normality.**

***Index Terms*—Bayesian Optimization, Gaussian Distribution, Hyperparameters, Neural Networks.**

I. Introduction

Many industries have been disrupted by the influx of neural networks. The last decade has yielded an incredible amount of attention at neural networks in many areas such as face recognition, big data clusterization, and signal processing.

In real deep learning projects, tuning hyperparameters is the primary key to build a network that provides accurate predictions for a specific problem. Common hyperparameters comprise the number of network layers, nodes in each layer, the activation function, and how many times (epochs) training should be repeated. Hyperparameters determine how the neural network is structured, how it is trained, and how its different elements function. The optimization problems for neural network size reduction and hyperparameters are well known. Actually, one of the first books on this topic was published by Kevin Swingler in 1996 [1]. Optimizing hyperparameters is an art: there are several ways ranging from manual trial and error to sophisticated algorithmic methods.

Recognized algorithms for hyperparameters estimation are the Grid search, Random search, Bayesian optimization, Gradient approach, and Evolutionary optimization.

Grid search assumes a researcher can construct multidimensional grid for feasible values of parameters. The idea behind the method lies in comparison of objective function values on grid points [2].

Random search [3] for estimation of neural network hyperparameters is an extension of the grid search. A statistical distribution is implemented for each hyperparameter under tuning, and their values are randomly sampled using the distributions.

Most papers on Bayesian optimization assume that the researcher is able to observe the objective function. Bayesian approach exploits past evaluation results to construct a probabilistic mapping hyperparameters to a probability of objective function values [4], [5]. The advantage of Bayesian method lies in looking for better hyperparameters based on previous trials.

As for Gradient approach [6], gradients are computed based on performance of cross-validation with respect to all hyperparameters. This occurs via chaining derivatives backwards through the training procedure. One can find optimization solution by any method of the first order.

Evolutionary algorithms are methods of the global optimization of black-box functions with noise. Evolutionary hyperparameter search follows the biological concept. Initial set that named initial population contains random generated hyperparameters [7], [8]. Algorithm checks fitness of each element of population and replaces the worst element with new one generated through evaluation procedure; that is crossover and mutation operations. The algorithm stops when the evaluation does not improve the population.

Bayesian optimization is considered the most contemporary and systematic method for neural network hyperparameters optimization. It cannot guarantee optimal solution; however, it provides near-optimal reasonable values of hyperparameters.

II. Problem, Materials and Software

The problem under consideration lies in the domain of neural network hyperparameters optimization. Bayesian optimization of neural network hyperparameters estimates random vector error calculation [9], which is supposedly normally distributed. Actually, input set of neural network in some papers presumed to be distributed in accordance with normal law [10]. Authors of this article have checked normality of vector sets on neural network layers using hypothesis-testing methods. Neural networks were designed as multilayer perceptrons with 3-5 hidden layers. Each learning data was passed through layers; and vector values on each layer were considered as material for numerical testing.

The initial data were collected from an open library of the machine learning datasets [11] of University of California,

Published on September 11, 2020.

Y. Karaki, Belarussian State University of Informatics and Radioelectronics, 220013, Minsk, Republic of Belarus.
(e-mail: youmna_karaki@yahoo.com)
H. Kaubasa, Belarussian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus.
(e-mail: g.kovbasa@gmail.com)
N. Ivanov, Belarussian State University of Informatics and Radioelectronics, 220013, Minsk, Republic of Belarus.
(e-mail: ivanovnn@gmail.com)

Irvine. In total, 11 datasets were checked. They are as follows: Brazilian high school (3 problems of classification), Hepatitis, Lymphography, Liver disorder (2 classifications), Dermatology, Glass identification, Adult (census data), and Wine quality. As usual, nominative variables were digitized in a conventional manner; e.g. a woman's gender was indicated as 0, whereas a man's gender as 1. Other nominative characteristics were enumerated by integers 0, 1, …, n.

Open software Anaconda [12] with its instruments Keras and TensorFlow in Python were used for our neural network design and learning procedures. Keras provides an access into the structure of network layers, so a researcher can easily get matrices for further multivariate testing.

Numerous software vendors offer systems and modules for statistical analysis. R language is one of the most prevalent languages in data science. It provides an efficient interface for testing process. R script for the multivariate data examination was developed with Multi Variate Normality (MVN) library from the CRAN project [13]. Calculation was done for neural networks as well as MVN on usual desktop. It exploited only CPU unit. Computing time for each dataset under examination was within 25 minutes in total for both, neural network and MVN test.

III. Multivariate Normality Tests

Mathematicians have discovered normal distribution about two centuries ago. Fisher and Kolmogorov-Smirnov tests for one-dimension data are such reliable measurements. There are special applications for testing multidimensional data. The measure of non-normality for both univariate and multivariate data depends upon asymmetry, tail weight, outliers, and modality. Both univariate and multivariate skewness and kurtosis measure the same characteristics. However, the comparison is done on the joint distribution of numerous variables against a multivariate normal distribution. This is an alternative to the comparison of one variable distribution against a univariate normal distribution. Skewness and kurtosis are the most efficient values as H. Scheffe has remarked in his book [14]. He noted that kurtosis and skewness are the key indicators of the degree to which nonnormality impacts the usual inferences made in variance analysis.

Moreover, skewness and kurtosis are an instinctive way to comprehend normality. If skewness differs from zero, then distribution deviates from symmetry; whereas if kurtosis differs from zero, then distribution diverges from normality in tail mass and shoulder.

There are various formulations for skewness and kurtosis in literature. In 1998, Joanes and Gill [15] originated three common formulations for univariate skewness and kurtosis.

If a sample variance is normally distributed, then kurtosis is equal to zero. It implies that the standard error of variance will be underestimated when kurtosis is positive; and overestimated otherwise. Kurtosis has an impact on variance estimates when the sample sizes are large; whereas in small samples, mean estimates are only affected. Yuan et al. [16] indicated that the characteristics of mean estimates are not influenced by either skewness or kurtosis asymptotically; however, standard error of sample variance is actually a function of kurtosis.

Tests on multivariate normality were verified by K. V. Mardia, K. V. Baringhaus [17], and N. Henze [18]. Actually, there exists several measures for multivariate skewness and kurtosis; however, Mardia's ones are definitely the most common.

IV. Experiment on Datasets Normality

In our experiment, a test was performed on the distribution of neurons in the neural network in order to verify the hypothesis of a multidimensional normal distribution. There are tables demonstrating the skewness and kurtosis of our neural network, generated chi-square multivariance, t- and normal distribution. The hypothesis of univariate normality on single component of some initial datasets was checked as well.

Large hidden layers usually permit the neural network to suit the training data very well. However, since regularization is typically used, it is essential to go for large hidden layers. Using the same size for all of the hidden layers most likely works better than choosing a decreasing or increasing size. Using a first hidden layer that is larger than the input layer, tends to work better too. With unsupervised pre-training, the layers ought to be much bigger than when implementing purely supervised optimization.

Univariate normality for two initial datasets had been verified by skewness and kurtosis estimation approach as well. The significance level for all tests was equal to 0.05.

We created and trained 11 neural networks, instances of a multilayer perceptron; as a training sample, we selected multivariate statistical data on various topics, such as human diseases, characteristics of school students, etc.

Before computing results of the perspective values of the datasets and in order to simplify and make it possible to compare the results of the test, every dataset size was restricted to 8 divisions and from 150 to 600 vectors.

MVN package has methods to calculate the mean and other significant parameters of the data. To concretely demonstrate the impact of skewness and kurtosis, tests were implemented.

As a result of the tests carried out on different variants of the trained neural networks, the following skewness and kurtosis values were obtained.

TABLE I: Multivariate Normality Test

Data	Size	Skewness	Kurtosis	MVN
Braz_School	395*8	1.278e3	2.618e1	No
Braz_Stud32	395*8	9.684e2	1.533e1	No
Braz_Stud33	395*8	2.116e3	3.297e1	No
Hepatitis	154*8	2.684e2	6.092e-1	No
Lymphography	148*8	1.332e3	3.569e1	No
Liver1	165*5	5.577e1	2.880e1	No
Liver2	165*5	5.428e2	2.815e1	No
Dermatol	366*6	2.232e3	1.288e2	No
Glass	214*7	3.932e3	1.041e2	No
Adult	562*8	2.386e3	3.146e2	No
Wine	500*7	4.336e3	7.677e2	No

TABLE II: UNIVARIATE PARAMETERS FOR BRAZ_SCHOOL

№	Mean	Std. Dev	Median	Skew	Kurtosis	Normal
1	1.740e-1	2.448e-1	1.147e-1	8.269e-1	2.262e-1	No
2	1.715e-1	1.965e-1	1.769e-1	-8.148e-2	-3.690e-1	Yes
3	1.279e-1	2.106e-1	1.224e-1	1.036e-1	-7.000e-2	Yes
4	1.915e-1	1.645e-1	1.834e-1	2.538e-1	9.413e-1	No
5	2.696e-2	9.171e-1	2.199e-2	4.406e-1	1.188e0	No
6	8.447e-2	1.286e-2	8.463e-2	6.300e-1	2.268e0	No
7	5.345e-2	5.377e-2	4.840e-2	1.2907e0	4.663e0	No
8	4.051e-2	1.314e-1	1.988e-2	5.389e-1	-1.282e-1	No

TABLE III. UNIVARIATE PARAMETERS FOR HEPATITIS

№	Mean	Std. Dev	Median	Skew	Kurtosis	Normal
1	4.624e-4	1.640e-3	4.854e-4	-4.853e-1	1.815e-1	No
2	4.561e-1	3.418e-1	4.280e-1	-1.006e-1	-7.676e-1	No
3	2.647e-1	4.162e-1	1.819e-1	-5.003e-2	-8.358e-1	No
4	1.774e-1	3.553e-1	1.544e-1	8.843e-2	-6.118e-1	Yes
5	2.041e-1	3.-1793	1.441e-1	2.802e-1	-6.705e-1	No
6	1.322e-1	2.562e-1	6.537e-2	9.728e-1	5.972e-1	No
7	6. 620e-2	9.590e-1	6.530e-2	-1.048e-1	-3.164e-1	Yes
8	4.631e-4	3.331e-2	-5.762e-3	1.187e0	1.184e0	No

V. CONCLUSION

During testing, it was evident that the distribution of neurons in a neural network is not multivariate normal. Through conducting a comparative analysis of the skewness and kurtosis values for various types of multivariate distributions, it was clear that for the chi-square case, the skewness and kurtosis values for the sample size 200-400 are out of the critical values of the multivariate normal distribution.

Some components of initial datasets have occurred inside the confident interval for univariate normality.

The values extracted from our experiment prove the independence of the results of testing the distribution of neurons of a neural network from the input data for training, the number of network layers, and the number of nodes in each layer.

ACKNOWLEDGMENT

First and Foremost, praises and thanks to God, the Almighty, for His showers of blessings throughout our work to complete this paper successfully.

Y. Karaki and H. Kaubasa would like to express their deep and sincere gratitude to their supervisor, Dr. Nick Ivanov for his diligent and thorough efforts in contributing to this paper.

We are also extremely grateful to our caring and loving families. Their continuous support and encouragement to complete this paper is much appreciated and duly noted.

REFERENCES

[1] K. Swingler, "Applying neural networks: a practical guide," London, Academic Press, 1996, p. 303.

[2] J. Bergstra, and Y. Bengio, "Random search for hyperparameter optimization," The Journal of Machine Learning Research, vol. 13, no. 1, pp. 281-305, 2012.

[3] J. Bergstra, D. Yamins,, and D. Cox, "Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures", The Journal of Machine Learning Research, v.28, n.1, 2013, pp. 118–123.

[4] J. Snoek, H. Larochelle, R.P. Adams, Practical Bayesian optimization of machine learning algorithms, Proc. of Advances in Neural Information Processing Systems 25, NIPS 3-6 Dec 2012, Lake Tahoe, Nevada, USA, pp. 2960–2968.

[5] J. Lampinen, and A. Vehtari, "Bayesian approach for neural networks - review and case studies," Neural Networks, vol. 14, no. 3, 2001, pp. 7-24.

[6] D. Maclaurin, D. Duvenaud,, and R.P. Adams, "Gradient-based hyperparameter optimization through reversible learning", Proc. of 32nd Int. Conf. on Machine Learning, ICML, 6–11 July 2015, Lille, France, , pp. 2113–2122.

[7] D. Orive, G. Sorrosal, C.E. Borges, C. Martin, and A. Alonso-Vicario, "Evolutionary algorithms for hyperparameter tuning on neural networks models", Proc. of the 26th European Modeling and Simulation Symposium, EMSS, 10-12 Sept. 2014, Bordeaux, France, pp. 402 - 410.

[8] E. Bochinski, T. Senst, and T. Sikora, "Hyperparameter optimization for convolutional neural network committees based on evolutionary algorithms", Proc. of 2017 IEEE Image Processing Int. Conf., ICIP-2017, 17-20 Sept. 2017, Beijing, China, pp. 3924-3928.

[9] F. Dernoncourt, and J.Y. Lee, "Optimizing neural network hyperparameters with Gaussian processes for dialog act classification", Proc. of IEEE Spoken Language Technology Workshop, SLT 2016, 13-16 Dec. 2016, San Diego, USA, pp. 406-413.

[10] P. Muguran, "Hyperparameters Optimization in Deep Convolutional Neural Network /Bayesian Approach with Gaussian Process Priors", arXiv:1712.07233v1, 19 Dec. 2017, p.10.

[11] University California Irvine Machine Learning Data Sets. http://archive.ics.uci.edu/ml/datasets.html, Retrieved 4 Feb, 2019.

[12] https://anaconda.org, Retrieved 7 Oct., 2019.

[13] https://cran.r-project.org, Retrieved 7 Oct., 2019.

[14] H. Sheffe, "The analysis of variance", New York, John Wiley & Sons, 1999, p. 477.

[15] D.N. Joanes, and C.A. Gill, "Comparing Measures of Sample Skewness and Kurtosis", The Statistician, vol. 47, Part 1, 1998, pp. 183-189.

[16] K.H. Yuan, P.M. Bentler, and W. Zhang, "The effect of skewness and kurtosis on mean and covariance structure analysis: the univariate case and its multivariate implication", Sociological Methods & Research, vol. 34, no. 2, 2005, pp. 240–258.

[17] K.V. Mardia, "Measure of multivariate skewness and kurtosis with applications", Biometrika, 57, 1970, pp. 519-530.

[18] L. Baringhaus, and N. Henze, "Limit distributions for measures of multivariate skewness and kurtosis based on projections", Journal of Multivariate Analysis, vol. 38, 1991, pp. 51–69.

Youmna Karaki (Lebanon, 1983) is pursuing her PhD in computer science at Belarusian State University of Informatics and Radioelectronics, Minsk. Her area of specialization is artificial neural networks.

She has published 2 articles for now. She has more than 17 years of teaching experience at different Lebanese Universities. She is currently a Lecturer at Arts, Sciences, and Technology University in Lebanon.

Halina Kaubasa (Belarus, 1999) has graduated from Belarusian State University of Informatics and Radioelectronics in 2020 where she got Bachelor degree as a systems engineer. She is now a Master student in the field of computer engineering at Belarusian State University of Informatics and Radioelectronics, Minsk. Her fields of interest include artificial intelligence and robotics. H. Kaubasa has published 3 articles for now. She is currently working as a software developer in the field of data protection and network administration.

Nick Ivanov (Belarus, 1949) has graduated from Belarusian State University in 1972; his specialty is applied mathematics. His fields of interest are network security and artificial neural networks.

He has published 1 monograph and more than 70 papers. He works now as an Associate Professor at Belarusian State University of Informatics and Radioelectronics.