Proceedings of CECNet 2021
A.J. Tallón-Ballesteros (Ed.)

doi:10.3233/FAIA210413

Investigating Automated Hyper-Parameter Optimization for a Generalized Path Loss Model

Usman Sammani SANI [a,c] , Daphne Teck Ching LAI [a,b,1] and Owais Ahmed MALIK [a,b]
[a] *Digital Science, Faculty of Science, Universiti Brunei Darussalam*
[b] *Institute of Applied Data Analytics, Universiti Brunei Darussalam*
[c] *Department of Electrical Engineering, Bayero University, Kano, Nigeria*

Abstract. This work aims at developing a generalized and optimized path loss model that considers rural, suburban, urban, and urban high rise environments over different frequencies, for use in the Heterogenous Ultra Dense Networks in 5G. Five different machine learning algorithms were tested on four combined datasets, with a sum of 12369 samples in which their hyper-parameters were automatically optimized using Bayesian optimization, HyperBand and Asynchronous Successive Halving (ASHA). For the Bayesian optimization, three surrogate models (the Gaussian Process, Tree Structured Parzen Estimator and Random Forest) were considered. To the best of our knowledge, few works have been found on automatic hyper-parameter optimization for path loss prediction and none of the works used the aforementioned optimization techniques. Differentiation among the various environments was achieved by the assignment of the clutter height values based on International Telecommunication Union Recommendation (ITU-R) P.452-16. We also included the elevation of the transmitting antenna position as a feature so as to capture its effect on path loss. The best machine learning model observed is K Nearest Neighbor (KNN), achieving mean Coefficient of Determination (R^2), average Mean Absolute Error (MAE) and mean Root Mean Squared Error (RMSE) values of 0.7713, 4.8860dB, and 6.8944dB, respectively, obtained from 100 different samplings of train set and test set. Results show that machine learning can also be used to develop path loss models that are valid for a certain range of distances, frequencies, antenna heights, and environment types. HyperBand produced hyper-parameter configurations with the highest accuracy in most of the algorithms.

Keywords. Path loss, environment, hyper-parameter optimization, feature importance

1. Introduction

Path loss is the decrease in the strength of radio frequency signal strength as it travels from the transmitter to the receiver. For effective design, expansion and monitoring in mobile networks, knowledge of the propagation characteristics of radio signals is required in order to determine the base station transmitting power and antenna height for a given cell radius [1],[2]. Propagation models are used in predicting path loss, which are mathematical expressions used in determining path loss based on frequency, antenna heights, distance etc. Path loss models are classified as either Deterministic models,

[1] Corresponding Author, School of Digital Science, Universiti Brunei Darussalam, Gadong, Brunei Darussalam; E-mail: daphne.lai@ubd.edu.bn

Empirical models, or Stochastic models. Empirical models are the most widely used but their accuracy may diminish if used in an environment different from the one they were developed [3]. Machine learning models are also used to predict path loss and the accuracy of such models outperforms that of empirical models. Such models have parameters that are set before training and their performances depend on them. These parameters are called hyper-parameters and a user might try to use the default setting of a software package or try to optimize their values. There are manual [4] and automated [5] ways to optimize hyper-parameters. The present study focuses on automated optimization of the hyper parameters of a generalized path loss model for multiple frequencies and environments.

Based on analysis of existing machine learning models, it was observed that the following optimization approaches were applied: no hyper-parameter optimization [6], manual tuning [7] and automated hyper-parameter tuning [8]. In terms of the environment type, existing models were developed for urban [9], suburban [10], or rural [11] environments individually, while others in combination [12]. The problem is that if we choose to work at specific frequencies and environment, then for a wide change in frequency another model is required for predicting path loss. As such several models must be developed for the unique environments and frequencies. This can limit the incorporation of machine models in Radio Frequency Planning tools.

Automated hyper-parameter optimization is carried out to increase accuracy, reduce time to tune hyper-parameters manually and to have consistent repeatability in observed results. The traditional way of hyper-parameter tuning is either through Grid search or Random search. Grid search has dimensionality curse for large search space and Random search has limited accuracy [13]. This work is aimed at developing an automatically optimized generalized model that can predict path loss for different environments and frequencies. The contributions of this work are:

a. Development of generalized path loss models that consider various frequencies and environments, with a clutter height feature used in differentiating the environments.
b. Use of state-of-the-art hyper-parameter optimization algorithms to automatically optimize the hyper-parameters of machine learning models used in the prediction of pathloss.

2. Methodology

This section describes the dataset used and its preparation, optimization algorithms used and the implementation process.

2.1. Data preparation

Dataset used in this work is a combination of four datasets from different environments, comprising of rural, suburban, urban, and urban high rise terrains, making a total of 18,720 data points. The datasets are made up of path loss values and eleven features namely, latitude, longitude, elevation, distance, frequency, transmitting antenna height (ht), receiving antenna height (hr), difference in latitude between transmitting and receiving antenna (distance_x), difference in longitude between transmitting and receiving antenna (distance_y), elevation at transmitting antenna height position (tAntennaElev) and clutter height. Three of the datasets used are public datasets available at [14], [15], and [16], and were originally used in [17], [18] and [19], respectively. The fourth dataset is a set of measurements carried out by the authors. The properties of the datasets are given in Table 1. Out of the 9480 data points for the urban area, 3,129 samples were extracted representing a stratified sample of 33% of the total so as to balance the samples per environment. A total 12,369 samples were used. Figure 1 shows the distribution of each feature by environment.

Table 1. Dataset properties

Dataset	Frequency (MHz)	Data points	Description
1 [17]	1835.2, 1840.8, 1864 and 1836	9308	Urban area
2 [18]	868	5624	3349 samples for urban high rise and 2275 for rural area
3 [19]	1800	3616	Sub urban area
4	2140	172	Urban area

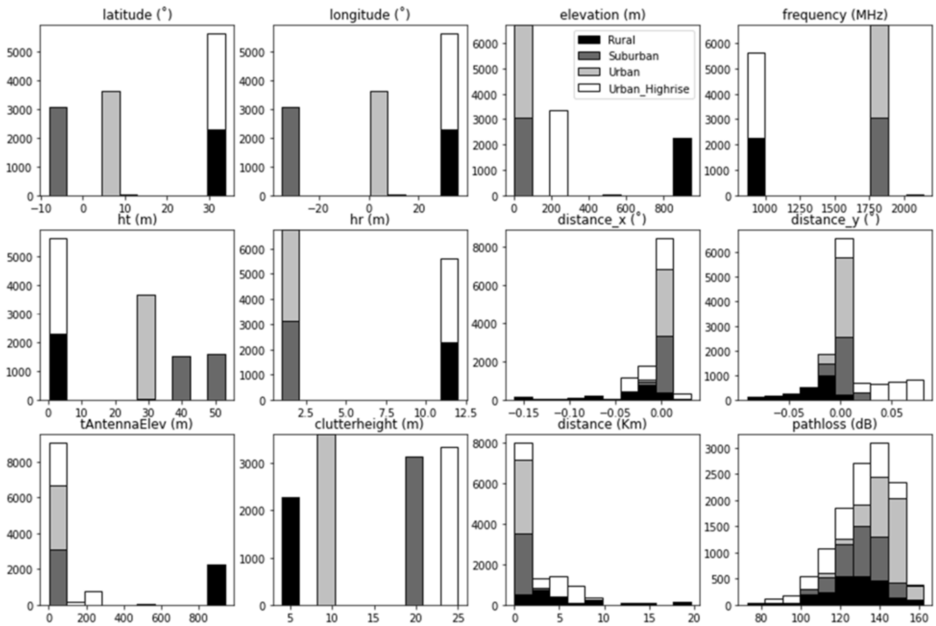

Figure 1. Distribution plots of features with respect to terrains

2.2. Optimization Algorithms

Optimization algorithms used are Bayesian optimization, Hyperband and Asynchronous Successive Halving (ASHA). Bayesian optimization is a sequential model-based optimization technique that takes prior information, use the present sample and then produce posterior information based on a criterion set by a utility function, U known as the acquisition function. It is used in solving functions in which their computation of extrema is expensive, evaluation of their derivative is hard, or they are non-convex. Hyberband and ASHA are early stopping approaches that allocate more resources to promising hyper-parameter configurations than unpromising ones in order to avoid waste. Resources refer to time or number of iterations [13].

2.3. Implementation

The performances of five machine learning algorithms are investigated, namely, KNN, RF, a single hidden layer MLP with Adam optimizer as weights updater, Gradient Boosting (GB) and Extreme Gradient Boosting (XGB) under their best hyper-parameter settings. Six different methods were used for the determination of the hyper-parameters which included Random search, Bayesian optimization using three different types of surrogates (GP, RF and TPE), HyperBand and ASHA. Eighty percent of the data points were used as train set in optimizing the hyper-parameter and twenty percent were used to test the performance. The features in both the train and test sets were scaled to have a mean of zero and standard deviation of one. The hyper-parameter values were evaluated using a 10-fold cross validation with the best score and time taken by each of the algorithms recorded. Each of the optimization algorithms was set for 50 trials/evaluations from which the best trial was selected. Five python packages were used to implement the automated hyper-parameter optimized machine learning process: XGBoost, Scikit-learn for the remaining machine learning models and random search, Scikit-Optimize for Bayesian search using GP and RF surrogate models, Hyperopt for Bayesian optimization using TPE surrogate and Optuna for HyperBand and ASHA.

The best hyper-parameter configuration from each search method was later used in the respective algorithm for performance evaluation. The performances of 100 different samplings of the train set and test set were averaged in each case and the mean MAE, mean RMSE, and mean R^2 were recorded. To examine the improvement in performance provided by the optimization processes, each of the algorithms was also run using the default hyper-parameter settings in the Scikit-learn package or XGBoost.

3. Results and Discussion

The following subsections present the results from various aspects of the study and the discussions about the results obtained. For result analysis, Bayesian optimization with Gaussian process is abbreviated BGP, BRF as an abbreviation for Bayesian optimization with RF surrogate model and BTPE for Bayesian optimization with TPE surrogate model.

3.1. Hyper-parameter Optimization

The performance metrics from the evaluation of the hyper-parameters resulting from the search algorithms, and the time taken by each of the search algorithms to complete 50 evaluations are presented in Table 2. Experiments were conducted with an Intel® Core™ i7-8700 CPU @ 3.20GHz × 12, with 15.6 GB of Random-Access Memory (RAM) and Linux Ubuntu release 20.04 as the operating system. Variations were observed across the five machine learning algorithms based on the hyper-parameter optimization algorithm that spent the highest and least amount of time to complete 50 iterations of the search as well as the algorithm that produced the highest score/least error as presented in Table 3. From Table 2, reduction in RMSE of 0.3693dB, 1.6318dB, 0.5681dB, and 0.922dB in KNN, MLP, RF, and XGB, respectively was achieved by the best optimization algorithm with respect to the default setting.

Table 2. Results of Hyper-parameter optimization

Algorithm		Time (mins)	Mean MAE (dB)	Mean RMSE (dB)	Mean R^2
KNN	Default	N/A	5.0357	7.2637	0.7460
	Random	4.1552	5.2967	7.5349	0.7270
	BGP	5.7399	5.0954	7.3435	0.7406
	BRF	4.7528	5.1499	7.3850	0.7374
	BTPE	**2.8291**	5.1739	7.3995	0.7364
	Hyperband	6.7423	**4.8860**	**6.8944**	**0.7713**
	ASHA	4.0985	4.8995	6.9203	0.7694
MLP	Default	N/A	7.0158	9.0367	0.6150
	Random	**4.1380**	6.9376	8.9774	0.6206
	BGP	7.1437	6.7411	8.7105	0.6435
	BRF	31.9863	**5.5831**	**7.4049**	**0.7415**
	BTPE	21.9405	5.9917	7.8796	0.7080
	Hyperband	12.3198	6.4114	8.3240	0.6740
	ASHA	9.8036	6.1387	8.0014	0.6987
RF	Default	N/A	6.8148	9.0916	0.6049
	Random	6.6318	7.0879	9.2457	0.5913
	BGP	9.1431	6.7877	8.9476	0.6205
	BRF	14.8185	7.0726	9.2129	0.5938
	BTPE	7.0694	6.9779	9.0628	0.6074
	Hyperband	5.4783	**6.4035**	**8.5235**	**0.6573**
	ASHA	**3.4771**	6.6433	8.7038	0.6416
GB	Default	N/A	**7.0895**	**9.2150**	**0.5966**
	Random	37.2973	7.2513	9.5577	0.5633
	BGP	20.2965	7.2310	9.5234	0.5691
	BRF	32.9644	7.2808	9.6443	0.5582
	BTPE	43.2859	7.1539	9.4587	0.5727
	Hyperband	26.7169	7.2611	9.4831	0.5626
	ASHA	**2.6171**	8.4016	11.0389	0.3834
XGB	Default	N/A	7.8449	10.4265	0.4842
	Random	3.3760	7.9297	10.5898	0.4688
	BGP	7.3779	7.9464	10.6292	0.4654
	BRF	7.7370	7.7580	10.3710	0.4911
	BTPE	4.3214	7.8279	10.4070	0.4862
	Hyperband	**2.5661**	**7.2991**	**9.5045**	**0.5725**
	ASHA	3.0020	9.5639	13.1130	0.1695

Table 3. Summary of performance

Algorithm	Highest Accuracy	Highest Speed	Lowest Speed
KNN	Hyperband	BTPE	Hyperband
MLP	BRF	Random	BRF
RF	Hyperband	ASHA	BRF
GB	Default	ASHA	BTPE
XGB	Hyperband	Hyperband	BRF

KNN is the best algorithm for this problem as it yielded the least values of the error metrics and a higher R^2 value. This is due to its stability, especially for large number of 14 neighbors as presented in Table 4. Other algorithms consisting of MLP and Tree based algorithms are less stable, and therefore overfit. HyperBand returned hyper-parameter values with the highest accuracy in most of the machine learning algorithms. Tables 4 to 8 present the default hyper-parameter values in the packages and values obtained from the various optimization methods.

Table 4. Hyper-parameter values of KNN

Parameter	Default	Random	BGP	BRF	BTPE	Hyperband	ASHA
Number of neighbors	5	4	4	4	4	14	13
Algorithm	auto	Ball tree	Ball tree	Kd tree	Kd tree	Ball tree	Ball tree
Leaf size	30	21	24	34	28	19	40
Minkowski Distance metric power	2	8	2	4	5	6	6

Table 5. Hyper-parameter values of RF

Parameter	Default	Random	BGP	BRF	BTPE	Hyperband	ASHA
Number of trees	100	214	162	292	192	260	173
Maximum depth	None	39	74	90	49	81	65
Maximum features per split	Auto	6	9	6	4	10	9
Minimum samples split	2	6	2	18	18	58	73
Minimum samples leaf	1	6	7	12	2	18	20

Table 6. Hyper-parameter values of MLP

Parameter	Default	Random	BGP	BRF	BTPE	Hyperband	ASHA
Number of neurons	100	108	21	115	96	32	102
Activation	relu	relu	tanh	logistic	tanh	logistic	logistic
Alpha	0.0001	0.0010	0.0009	0.0031	0.3391	0.1000	0.0080
Epsilon	1e-8	0.0010	0.2744	0.0059	0.8714	0.8518	0.5340
Beta_1	0.9	0.1000	0.4420	0.5526	0.8559	0.6329	07819
Beta_2	0.99	0.9900	0.0225	0.7798	0.8223	0.4865	0.5990

Table 7. Hyper-parameter values of GB

Parameter	Default	Random	BGP	BRF	BTPE	Hyperband	ASHA
Number of trees	100	176	63	188	153	40	215
Maximum depth	3	55	87	75	69	45	24
Maximum features per split	None	5	6	8	7	4	8
Minimum samples split	2	47	45	64	60	40	56
Minimum samples leaf	1	9	9	4	20	26	95
Loss	LS	LS	LS	LS	LS	Huber	LAD

Table 8. Hyper-parameter values of XGB

Parameter	Default	Random	BGP	BRF	BTPE	Hyperband	ASHA
Number of trees	100	68	253	246	281	184	119
Learning rate	0.3	1	0.2742	0.2755	0.1893	0.2449	0.0476
Booster	gbtree	gbtree	gbtree	dart	gbtree	gbtree	gbtree

3.2. Feature Importance

Feature importance from the best KNN model was also considered using the permutation importance method as depicted in Figure 2. It was observed that all features are relevant in the path loss prediction as none was scored zero in feature importance. A novel feature in this model is the “tAntennaElev” that stands for elevation at the location of transmitting antenna. The reason for using it is that since a multiple transmitter model is considered, the altitude of the transmitting antenna relative to the receiving antenna becomes variable also because different transmitting antenna positions have different elevation heights above sea level. The model was observed to give the feature a moderate priority greater than that of some of the features used in earlier empirical models such as frequency and the heights of the antennas. The height of the receiving antenna had the least priority.

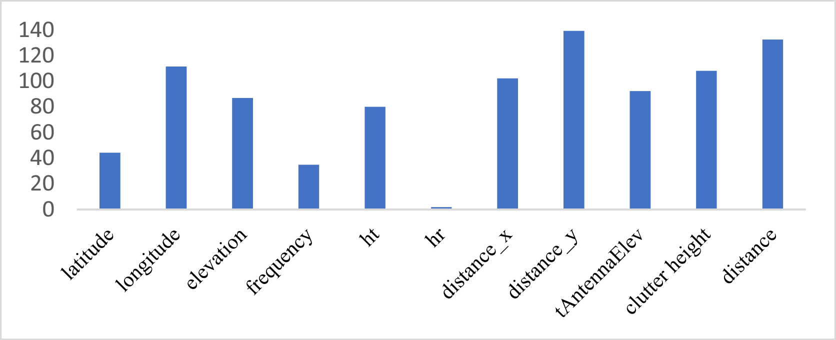

Figure 2. Feature importance of the best KNN model

3.3. Performance by Environment

The best optimized machine learning algorithm obtained is KNN. The R^2 value, MAE and RMSE from each of the environments were computed from the model's predictions on test data as presented in Table 9, with three scenarios. In the first scenario, 100 different samplings of Train-Test split were used in testing the model with unoptimized/default hyper-parameter configuration and results were averaged. In the second scenario, a single Train-test split was used while 100 different samplings were used in the third scenario as in the first scenario. It was observed that results in the first scenario had the least accuracy while the second scenario had the highest accuracy. Meanwhile, the third scenario had an accuracy in between the two other scenarios. The second scenario had the best accuracy because a single Train-Test split was used. This Train-Test split was also used during the optimization of hyper-parameters, while in the third scenario, the hyper-parameters obtained using the Train-Test split in the second scenario were used in checking performance using 100 different Train-Test splits. Thus, results in the second scenario are due to overfitting to the single Training data. Therefore, result from the third scenario should be used in estimating the performance of the obtained hyper-parameters as it reveals how the hyper-parameter setting responds to variations in training and testing data. The prediction plots per environment are shown in Figure 3. Figure 3(a) shows the measured path loss and that predicted by the model in the rural environment. It will be observed that large distances were covered, especially in the rural environment. This is because the dataset used contains measurements carried out by [18] from a Long Range Wide Area Network (LoRaWAN) and the distance covered in rural environment was larger due to little obstacles or structures. For the urban highrise in Figure 3(d), even though the measurements are from a LoRaWAN, the distance covered is smaller than in the rural environment due to the presence of high density of buildings. Figure 3(b) and Figure 3(c) represents the suburban and urban environments, respectively. Low distances were covered in these environments. It was observed that the values predicted by the model in suburban environment have the least fitting to the actual measured data as reflected in its performance metrics in Table 2. In addition to the frequencies in Table 1 in which the model is valid, the range of transmitting antenna height for which it is valid is from 0.2m to 53m and that of receiving antenna is from 1m to 12m as shown in Figure 1.

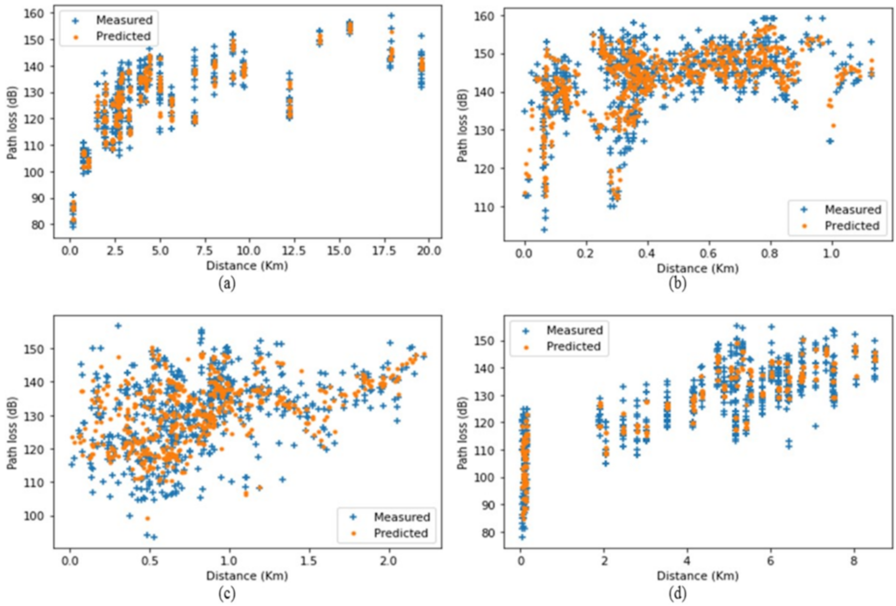

Figure 3. Predicted and measured path loss for (a) rural (b) suburban (c) urban (d) urban highrise environments

Table 9. Optimized and un-optimized performance metrics for the best KNN model

Environment	Un-optimized (100 runs)			Optimized (1 run)			Optimized (100 runs)		
	Mean R^2	Mean MAE	Mean RMSE	R^2	MAE	RMSE	Mean R^2	Mean MAE	Mean RMSE
Rural	0.8952	3.3787	4.9395	**0.9491**	**2.5720**	**3.4109**	0.8875	3.4985	5.1140
Suburban	0.1918	5.8191	8.3359	**0.4704**	**4.5385**	**6.4173**	0.2704	5.5744	7.9179
Urban	0.5518	5.2174	7.3314	**0.7492**	**4.3772**	**5.8047**	0.5846	5.1744	7.0588
Urban High rise	0.7419	5.2329	7.6560	**0.8529**	**4.1689**	**5.9110**	0.7818	5.0031	7.1369

4. Conclusion

We developed a novel path loss model valid for various environments, antenna heights, and frequencies, using the clutter height recommended in ITU-R P.452-16 as feature that differentiates among the environments. The model was obtained by comparing the performances of five machine learning algorithms in which state of the art optimization techniques were used in optimizing their hyper-parameters. To the best or our knowledge, no existing work has applied these techniques in the hyper-parameter optimization of machine learning algorithm for path loss models. We demonstrated that HyperBand optimization produced much improved results. The performance of each of the hyper-parameter optimization algorithms was observed to be dependent on the machine learning algorithm whose hyper-parameters are being optimized, as expected. Hyper-parameters obtained using HyperBand gave the best results in most of the algorithms. The best machine learning algorithm observed is K Nearest Neighbor resulting in an R^2 value of 0.7713. We demonstrated how machine learning models that consider similar properties adopted by the empirical approach, such as range of distances, antenna heights and frequencies can be developed, but with much improved results. We also demonstrated that evaluation of hyper-parameters repeatedly with various versions of Train-Test splits from the same dataset reveals the adaptive response of the hyper-parameters to variations in training data. Otherwise, training using only one trainset could result in an overfit. Our method improved accuracy, reduced the time for the optimization of hyper-parameters and the chance of having an overfitted model.

References

[1] Mahasukhon P, Sharif H, Hempel M, Zhou T, Wang W, Ma T. Propagation path loss estimation using nonlinear multi-regression approach. In: Proceedings IEEE International Conference on Communications. Cape Town, South Africa; 2010.

[2] Popoola SI, Oseni OF. Empirical Path Loss Models for GSM Network Deployment in. Int Ref J Eng Sci. 2014;3(6):85–94.

[3] Ogbulezie J, Onuu M. Site specific measurements and propagation models for GSM in three cities in Northern Nigeria. Am J Sci Ind Res. 2013;4(2):238–45.

[4] Ebhota VC, Isabona J, Srivastava VM. Signal power loss prediction based on artificial neural networks in microcell environment. In: Proceedings 2017 IEEE 3rd International Conference on Electro-Technology for National Development, NIGERCON 2017. Owerri, Nigeria; 2018. p. 250–7.

[5] Probst P, Boulesteix AL, Bischl B. Tunability: Importance of hyperparameters of machine learning algorithms. J Mach Learn Res. 2019;20(March).

[6] Benmus TA, Abboud R, Shatter MK. Neural network approach to model the propagation path loss for great Tripoli area at 900, 1800, and 2100 MHz bands. In: Proceedings 16th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering, STA 2015. Monastir, Tunisia; 2015. p. 793–8.

[7] Popoola SI, Faruk N, Oloyede AA, Atayero AA, Surajudeen-Bakinde NT, Olawoyin LA. Characterization of Path Loss in the VHF Band using Neural Network Modeling Technique. In: Proceedings 2019 19th International Conference on Computational Science and Its Applications, ICCSA 2019. Saint Petersburg, Russia: IEEE; 2019. p. 166–71.

[8] Lin KP, Hung KC, Lin JC, Wang CK, Pai PF. Applying least squares support vector regression with genetic algorithms for radio-wave path-loss prediction in Suburban environment. Lect Notes Electr Eng. 2010;67 LNEE:861–8.

[9] Mom JM, Mgbe CO, Igwue GA. Application of Artificial Neural Network For Path Loss Prediction In Urban Macrocellular Environment. Am J Eng Res. 2014;03(02):270–5.

[10] Popescu I, Nafornita I, Kanatas A, Moraitis N. Neural Model for Path Loss Prediction in Suburban Environment. 2001;(July).

[11] Eichie JO, Oyedum OD, Ajewole MO, Aibinu AM. Comparative analysis of basic models and artificial neural network based model for path loss prediction. Prog Electromagn Res M. 2017;61:133–46.

[12] Ayadi M, Ben Zineb A, Tabbane S. A UHF Path Loss Model Using Learning Machine for Heterogeneous Networks. IEEE Trans Antennas Propag. 2017;65(7):3675–83.

[13] Yu T, Zhu H. Hyper-Parameter Optimization: A Review of Algorithms and Applications. 2020;1–56. Available from: http://arxiv.org/abs/2003.05689

[14] GitHub - timotrob/SVR_PATHLOSS: The Database relative to paper A Proposal for Path Loss Prediction in Urban Environments using Support Vector Regression [Internet]. [cited 2021 Jan 2]. Available from: https://github.com/timotrob/SVR_PATHLOSS

[15] LoRaWAN Measurement Campaigns in Lebanon | Zenodo [Internet]. [cited 2021 Jan 2]. Available from: https://zenodo.org/record/1560654#.X-9B-VUzbIV

[16] GitHub - lamvng/Path-loss-prediction: Applying Multilayer Perceptron to predict Path Loss value. [Internet]. [cited 2021 Jan 2]. Available from: https://github.com/lamvng/Path-loss-prediction

[17] Timoteo RDA, Cunha DC, Cavalcanti GDC. A proposal for path loss prediction in urban environments using support vector regression. In: Proceedings of The Tenth Advanced International Conference on Telecommunications. 2014. p. 119–24.

[18] El Chall R, Lahoud S, El Helou M. LoRaWAN network: Radio propagation models and performance evaluation in various environments in Lebanon. IEEE Internet Things J. 2019;6(2):2366–78.

[19] Popoola SI, Atayero AA, Arausi OD, Matthews VO. Path loss dataset for modeling radio wave propagation in smart campus environment. Data Br [Internet]. 2018;17:1062–73. Available from: http://dx.doi.org/10.1016/j.dib.2018.02.026