

Application of Artificial Neural Networks (ANNs) for Estimation of Direct Radiation Data

SR-19-12; Mar 2020

Young-Hoon Jin P.E., Yazmin Avila, and Abel Porras P.E.

City of Austin
Watershed Protection Department
Environmental Resource Management Division

Abstract

Long-term simulation of the Gridded Surface Subsurface Hydrologic Analysis (GSSHA) model requires hydrometeorological data such as barometric pressure, relative humidity, total sky cover, wind speed, dry bulb temperature, direct radiation, and global radiation on an hourly basis for a simulation period. Data for each of these inputs except direct radiation are accessible through a website maintained by the National Oceanic and Atmospheric Administration (NOAA). The direct radiation data is available at the National Solar Radiation Data Base (NSRDB) based on computer simulation models one year after other data are collected. Thus, if ERM staff wish to model the current year of stream flow using GSSHA, it is necessary to establish a model for estimation of hourly direct radiation data for the current year. The present study applies Artificial Neural Networks (ANNs) to build a model for the estimation of direct radiation using relative humidity, dry bulb temperature, total sky cover, barometric pressure, wind speed, the corresponding month, hour data, and global radiation as input variables. The results from the application of ANNs reveals that the model with eleven hidden nodes for two hidden layers, namely ANNs (8-11-11-1), estimates the direct radiation with high accuracy during the data periods for testing as well as training of the model. In conclusion, it is feasible to use the application of ANNs to estimate the direct radiation which allows for more accurate long-term simulations using the GSSHA model for the current year.

Introduction

One of the key aims of the Environmental Resource Management (ERM) Division has been to understand and relate biological responses to stream hydrology. Scoggins (2000) demonstrated that hydrologic parameters proposed by Poff and Ward (1989) and Richter, et al. (1996) could be related to stream biology. In 2010, the ERM Division expanded upon this idea by forming a linear regression model relating various flow metrics to biological responses of benthic macroinvertebrates in the streams (Glick et al., 2010). Hydrologic metrics are calculated by examining a hydrograph for a period of time and reducing the

hydrograph to a single number, better known as a scalar response. For example, the average flow during a time period is a common hydrologic metric that is computed. The work in 2010 used the daily average flow to construct the hydrographs which were used to calculate the metrics. However, it was thought that the use of smaller time steps in the hydrograph could lead to a more accurate computation of hydrologic metrics. In 2011, ERM further refined the linear regression model between stream hydrology and the response of benthic macroinvertebrates by using hydrologic metrics computed based on stream flow data collected on 15-minute intervals (Richter, 2011).

There is a large amount of data lost when translating a hydrograph down to a single numerical number such as the average flow or peak flow during a time period. Using an advanced statistical technique, called Functional Data Analysis (FDA), users can develop associations between functionals¹ and either other functionals or scalars (Stewart-Koster et al., 2014). Thus, one can build a prediction model for benthic macroinvertebrate responses computed as numerical scores (i.e. scalars) using the entire streamflow hydrograph (i.e. functional). Richter (personal comm.) developed such a model using yearlong hydrographs from existing gage stations. Since benthic macro-invertebrates provide ecological function to the streams and are a key component in ERM's long-term environmental monitoring program, expanding upon a benthic macro-invertebrate prediction model to both gaged and ungauged sites may be useful in developing salient water quality objectives throughout the stream (Herrington, 2018). This requires the ability of watershed models to estimate these yearlong hydrographs anywhere along the stream.

Streamflow at the ungauged sites can be simulated by a watershed model using land use and topographic information, as well as historical observations of precipitation and hydrometeorological information. The ERM Division is currently simulating long-term streamflow using the Gridded Surface Subsurface Hydrologic Analysis (GSSHA) model which requires various hydrometeorological data to estimate evaporation (Downer and Ogden, 2006).

The hydrometeorological data includes seven parameters such as barometric pressure, relative humidity, total sky cover, wind speed, dry bulb temperature, direct radiation, and global radiation on an hourly basis. The former five parameters and global radiation can be downloaded from National Centers for Environmental Information of National Oceanic and Atmospheric Administration (NOAA), where historic and recent data are available. Direct radiation data can be obtained from the National Solar Radiation Database (NSRDB) but is only available historically as the direct radiation data is produced from a model after the other data is processed. NSRDB uses the Physical Solar Model (PSM), which is a physics-based modeling approach, to provide solar radiation data for the entire United States [<https://nsrdb.nrel.gov/about/u-s-data.html>]. At the time of this study, direct radiation data was available until 2017.

Therefore, it is necessary to estimate current direct radiation data to predict recent benthic macroinvertebrate scores at gaged/ungauged sites. Direct radiation, the authors have found, is a complex phenomenon that is dependent on a host of other non-linear factors. The present study applies Artificial Neural Networks (ANNs), which is a theoretical modeling approach, to build a model for estimation of direct radiation with the purpose of utilizing the estimated data for the long-term simulation of the GSSHA model.

This report aims to:

1. document the process by which an ANN can be constructed from training and testing data to make estimations for an unknown and non-linear process;
2. apply this process to predict direct radiation data for 2018; and

¹ We can refer to functionals as the space of all functions.

- inform the reader to the many possibilities of machine learning to large sets of complex environmental data.

METHODS

Direct Radiation is a useful quantity because it provides an estimate of energy available for environmental processes used in the simulation of stream flow, such as evaporation of soil water. It is defined as the radiation that has not experienced scattering from the atmosphere (Sørensen, 2017). Thus, as direct radiation enters the Earth’s atmosphere, it is impacted by several variables including time of day, day of year, location on Earth, cloud cover, relative humidity, barometric pressure, and other minor constituents in the air that may refract light waves. Relative humidity, dry bulb temperature, total sky cover, barometric pressure, and wind speed data were measured at the weather station in Austin Camp Mabry, TX, and were downloaded from National Centers for Environmental Information of NOAA for the data period of January 2016 to April 2019.

The NSRDB provides modeled hourly and half-hourly values of the most common measurements for direct radiation — global horizontal irradiance (GHI), direct normal irradiance (DNI) and diffuse horizontal radiance (GHI) for the entire United States in 4 km gridded segments. The database is updated yearly by the National Renewable Energy Laboratory (NREL) and data is publicly available via the NSRDB Viewer Application [<https://nsrdb.nrel.gov>]. The most recent updated model is the Physical Solar Model (PSM) version 3 and the data is available from 1998 to 2017. Direct Radiation was downloaded for the data period from 2016 to 2017.

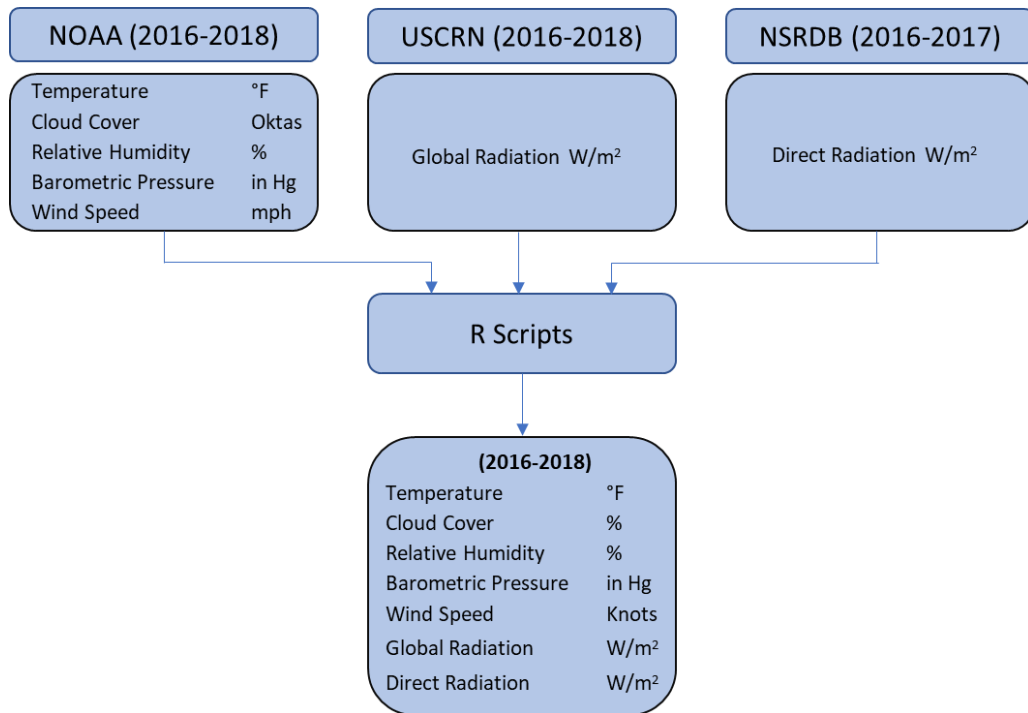


Figure 1. Flow chart for data preparation with data sources/period, scripts, and output units

The U.S. Climate Reference Network (USCRN) is a systematic and sustained network of climate monitoring stations with sites across the U.S. Data is updated daily by the National Centers for Environmental Information of NOAA and is available on sub-hourly, hourly, daily, and monthly values for each station. Hourly data include air temperature, precipitation, global solar radiation, relative humidity, surface infrared temperature, soil moisture, and soil temperature. Hourly global radiation was obtained from the station located 33 miles northwest of Austin (ID No. 23907) for the period from 2016 to 2019. The data is available in ASCII text format.

Figure 1 shows the overall process of data collection and transformation with data sources, downloaded data period, scripts for data conversion (Appendix A), and output units. The output format with the units is listed in Appendix B. The whole dataset is split into three subsets for training, testing, and estimation. The data measured in 2016 and 2017 are used for training and testing, respectively. The variables used to estimate direct radiation were month, hour of the day, relative humidity, dry bulb temperature, total sky cover, barometric pressure, wind speed, and global solar radiation. Direct radiation data measured in 2016 and 2017 are compared with the outputs from the model described below. Data for the six parameters in 2018 are used for estimation of direct radiation which is not available in the year.

Theory of Artificial Neural Networks

Artificial Neural Networks (ANNs) have been applied to estimate/predict or classify variables in numerous research fields. Many researchers have been documenting the applicability and feasibility of ANNs to build estimation, prediction, and classification models (Maier and Dandy, 2000; Hsu et al., 2002; Jin et al., 2011; Qazi et al., 2015). In general, ANNs have input nodes, one or multiple hidden layers, and one output layer. Each layer also has nodes which sum the weighted signals including a bias from previous layers and transfer the summed values to activation (or transfer) functions. Outputs from the activation functions are forwarded to the next layer. Input nodes, hidden and output layers are fully connected by weights.

ANNs with response variables from which to train the algorithm are known as supervised learning methods. During the training or learning process, the weights of ANNs are updated to reduce errors between the target values (e. g. observed data) and outputs from ANNs using the error Back-propagation Algorithm (BPA), representatively. The BPA performs backward calculations to update weights of respective networks between output and hidden layers, between hidden layers, and between hidden layers and input nodes. This training process also involves setting multiple model parameters such as numbers of hidden layers and nodes in each layer, learning rate, momentum constant, number of training epochs, and stopping criteria.

The learning rate is a step size taken in weight space to achieve a target value, and the momentum is used to speed up the training process (Maier and Dandy, 2000). The learning rate remains fixed during training process and the momentum is applied to weights at the previous step to update weights at the current step. Stopping criteria are used to decide when to stop the training process, for example, based on a threshold for partial derivatives of error function or maximum steps for the training process (Maier and Dandy, 2000; Fritsch et al., 2019).

In general, activation functions can be selected among logistic, hyperbolic tangent, and linear functions for nodes in each layer. Logistic and hyperbolic tangent functions are non-linear and determine the output range from a node between 0 and 1 and between -1 and 1, respectively. Empirical studies indicate that the hyperbolic tangent function should be used when data are noisy and have mildly non-linear

relationships (Kalman and Kwasny, 1992; Maier and Dandy, 1998), while non-sigmoidal functions perform better when data are noiseless and contain highly non-linear relationships (Moody and Yarvin, 1992). Linear functions in the output layer can be used to improve extrapolation ability of a Multi-Input and Multi-Output (MIMO) model using ANNs (Jin et al., 2004).

ANNs with multiple hidden layers are called Deep Learning, a sub-field within machine learning. The deep learning is defined as a powerful multi-layered architecture for pattern recognition, signal detection, and classification or prediction (Hodnett and Wiley, 2018). For the present study, the Package ‘neuralnet’ (Fritsch et al., 2019) coded in R programming language is used for the modeling process of ANNs. The package employs the Resilient back-PROPagation (RPROP) algorithm, which is known as a faster weight update mechanism than the traditional BPA. There is no need to specify the learning rate and momentum constant for training when using RPROP because the algorithm only uses the signs of gradients to update weights (Riedmiller and Braun, 1993).

APPLICATION TO DIRECT RADIATION ESTIMATION

Model Configuration

A mathematical form of the model for the present study is represented as below.

$$DR(t) = ANNs[Month(t), Hour(t), Tmp(t), Hmd(t), Cld(t), Prs(t), Wnd(t), GR(t)]$$

where, DR(t) represents direct radiation at time t with unit of [Wh/m²], and Month(t) for month, Hour(t) for hour, Tmp(t) for dry bulb temperature [°F], Hmd(t) for relative humidity [%], Cld(t) for total sky cover [%], Prs(t) for barometric pressure [in Hg], Wnd(t) for wind speed [kts], and GR(t) for global radiation [Wh/m²] at time t. ANNs represent the model to estimate direct radiation using the input data.

The ANNs for this study were structured consisting of eight input nodes, two hidden layers with eight to twenty-four nodes for each model, and one output layer with one node. The ANNs were trained with the model parameters listed in Table 1. Two hidden layers for a model have the same number of nodes. The threshold in the table is a numeric value meaning the threshold for the partial derivatives of the error function as a stopping criterion (Fritsch et al., 2019).

Table 1. Model parameters used for training of ANNs

Parameter	Value	Remark
Threshold	0.01	Stopping criterion
Maximum steps	1e+06	Stopping criterion
Error function	SSE	Sum of squared errors
Activation function	Logistic	Applied to all layers (two hidden and output layers)

Data Pre/Post Processing

Input and target data for ANNs are rescaled into the range of [0.1, 0.9] using maximum and minimum values of each variable for the training period data. The calculation process of maximum and minimum values for rescaling each variable should not include other datasets for testing or estimation. The maximum and minimum values are used for data post-processing.

Model Identification

Figure 2 illustrates the structure of an ANN with two hidden layers and one output layer, which is denoted as ANNs (8-11-11-1). The notation means that the model has 8 input nodes, 11 nodes in each hidden layer, and one node in the output layer. ANNs with increasing numbers of nodes in the hidden layer were run to identify the best model structure based on their performance.

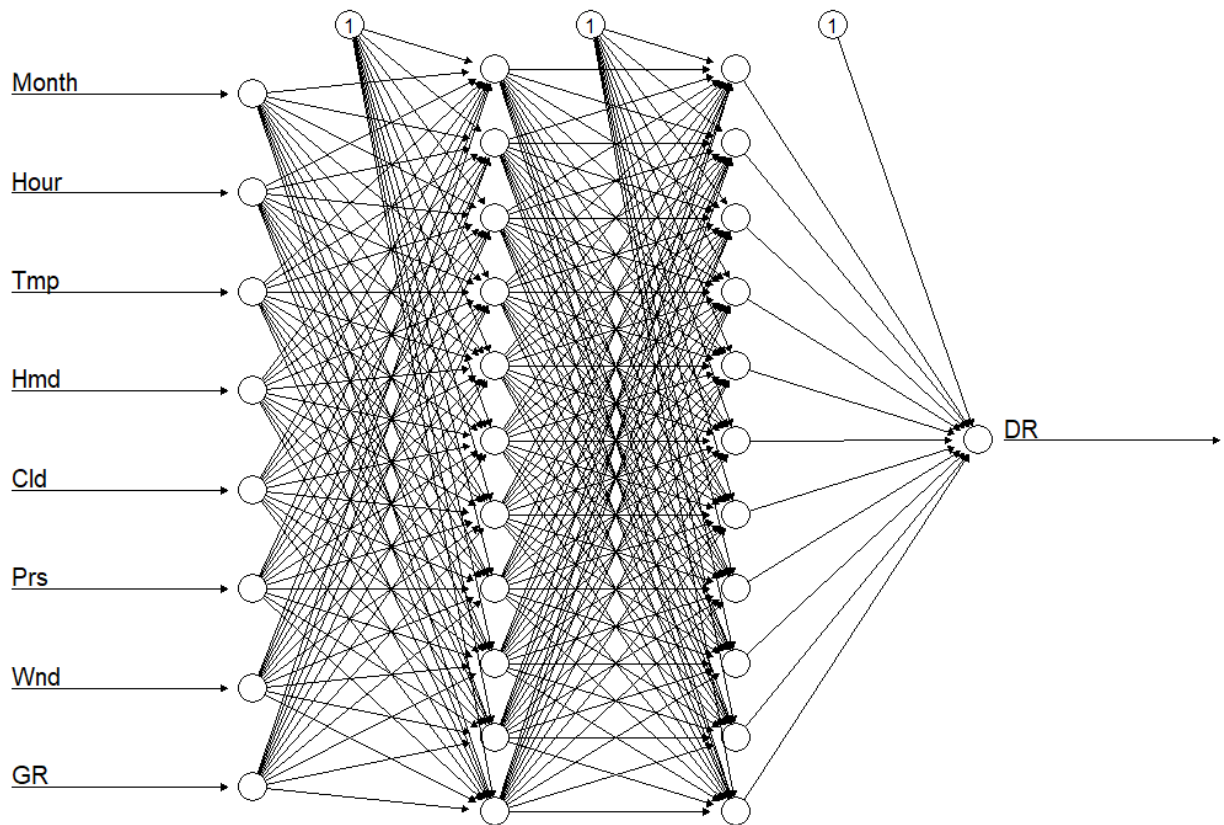


Figure 2. Model structure of ANNs (8-11-11-1)

Results

R^2 and RMSE were calculated for ANNs models with eight input nodes, eight to twenty-four nodes in the two hidden layers, and one output node for training and testing. The ANNs (8-11-11-1) was identified as the best-balanced model for both training and testing phases. These model runs, when plotted against R^2 and Root Mean Squared Error (RMSE), provide a visual indication of the best-balanced performance for both training and testing to avoid overfitting, as shown in Figure 3 and Figure 4. Higher R^2 values and lower RMSE values indicate better performance. Although the ANNs (8-24-24-1) shows the best performance for training, it has much worse performance for testing than ANNs (8-11-11-1). It is interpreted that the model overfitted to the training data and poorly estimated the direct radiation when using a new dataset.

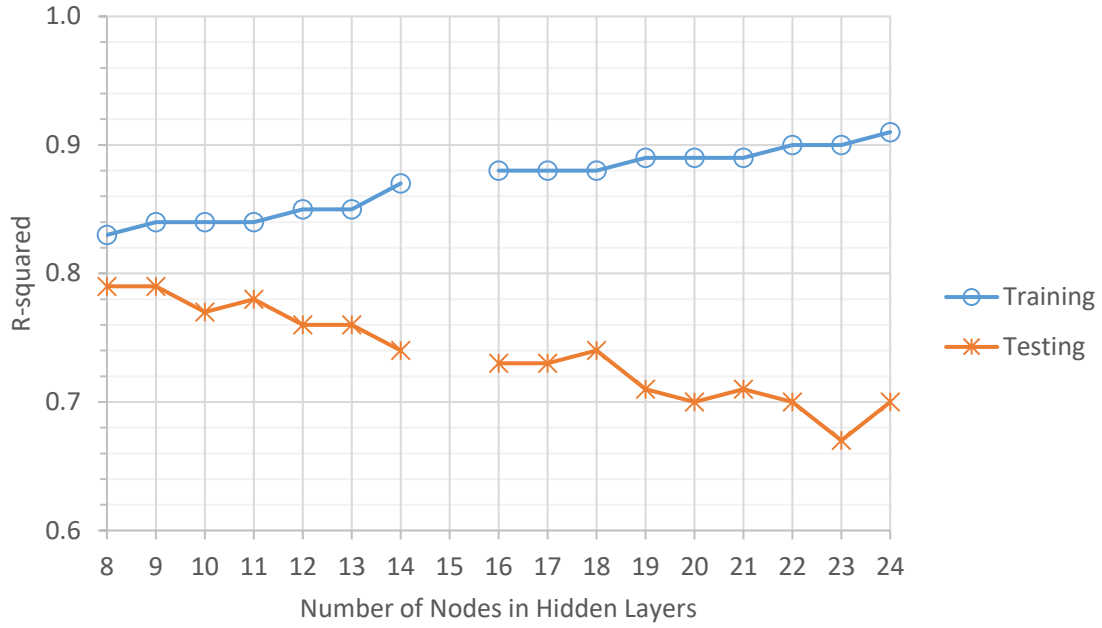


Figure 3. Model performance based on R-squared

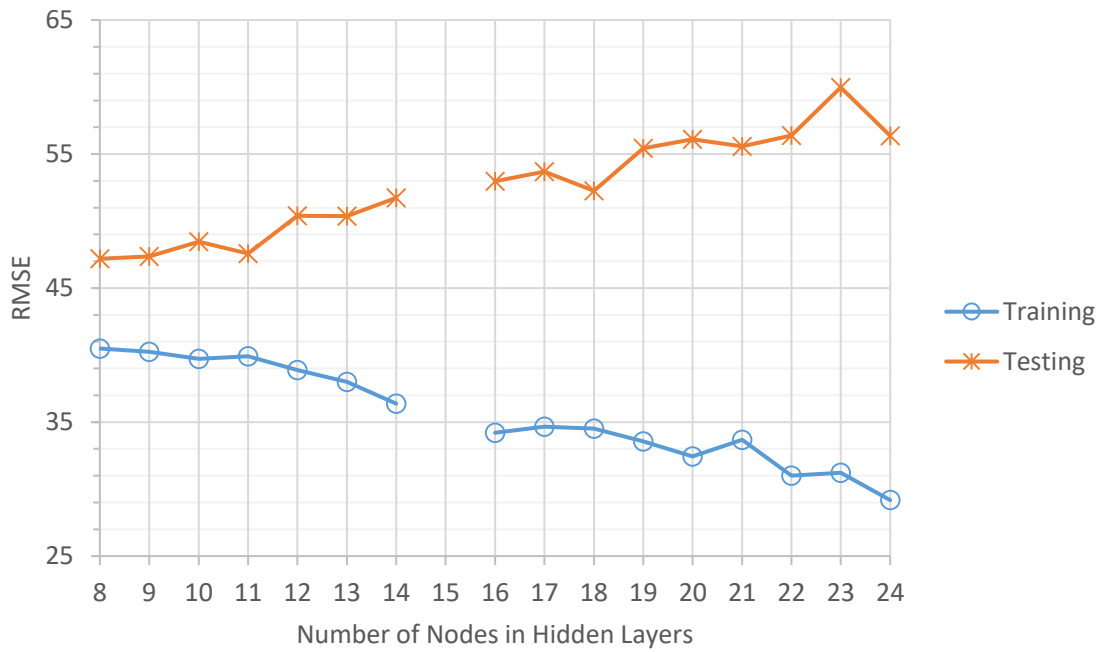


Figure 4. Model performance based on RMSE

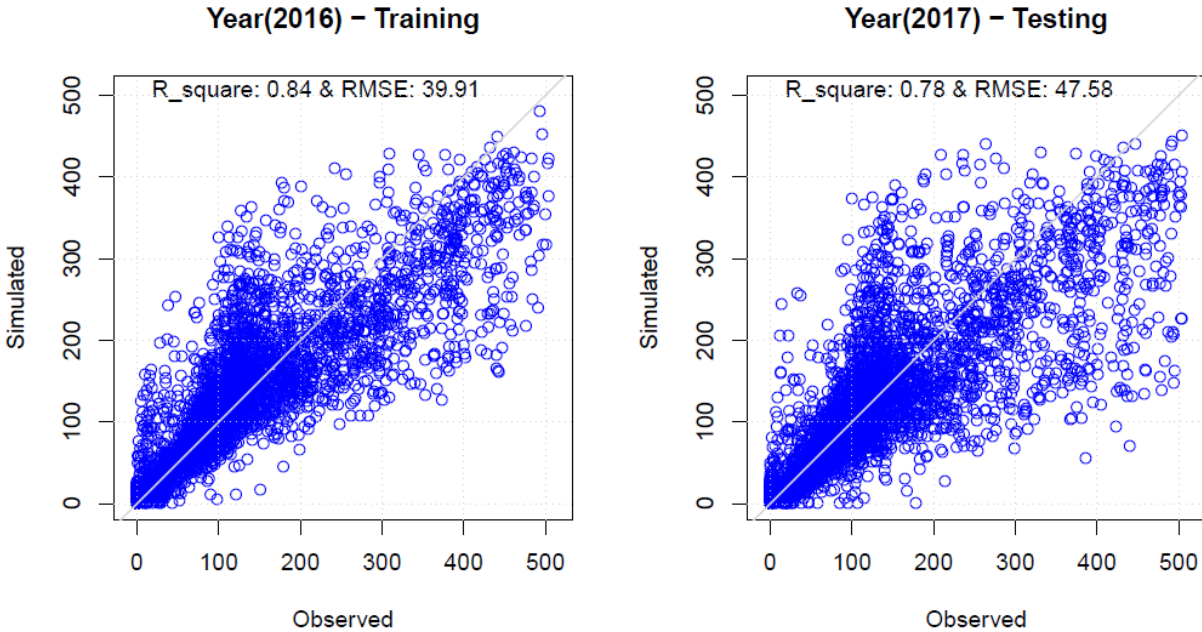


Figure 5. Scatter plots of ANNs (8-11-11-1) for training and testing

The models with eight to ten nodes in each hidden layer show significant underestimation for high direct radiation during the training phase. ANNs (8-11-11-1) estimates the high direct radiation better than the other models. The scatter plots between observed and estimated data are shown in Figure 5 for training and testing of the ANNs (8-11-11-1). ANNs (8-15-15-1) has no performance results because the model did not reach the threshold listed in Table 1 within the maximum steps for training. The scatter plots show the strong linear relationship between observed and estimated direct radiation with overestimation of the data ranged from 100 to 200 [Wh/m²] and slight underestimation for high direct radiation during training and testing phases.

In addition, the detailed graphical comparison between observed and estimated direct radiation is shown in Figures 6 and 7 using monthly histograms for training and testing phases. It should be noted that zero values of estimated direct radiation were excluded for better graphical visualization. In general, the overlapped histograms in the figures show similar distributions of observed and estimated data during winter but different distributions during the months including high direct radiation.

Weights between input nodes and the first hidden layer are listed in Table 2 in a [9 x 11] matrix form representing (number of input nodes + a bias = 9) and (number of hidden nodes in the first hidden layer = 11). Table 3 has a [12 x 11] matrix showing the weights between the first and second hidden layers; (number of hidden nodes in the first hidden layer + a bias = 12) and (number of hidden nodes in the second hidden layer = 11). Table 4 shows the weights between the second hidden layer and output layer in a [12 (number of hidden nodes in the second hidden layer + a bias) x 1 (number of output node)] matrix format. The identified ANNs was applied to estimate direct radiation data in 2018 which are not available in the year.

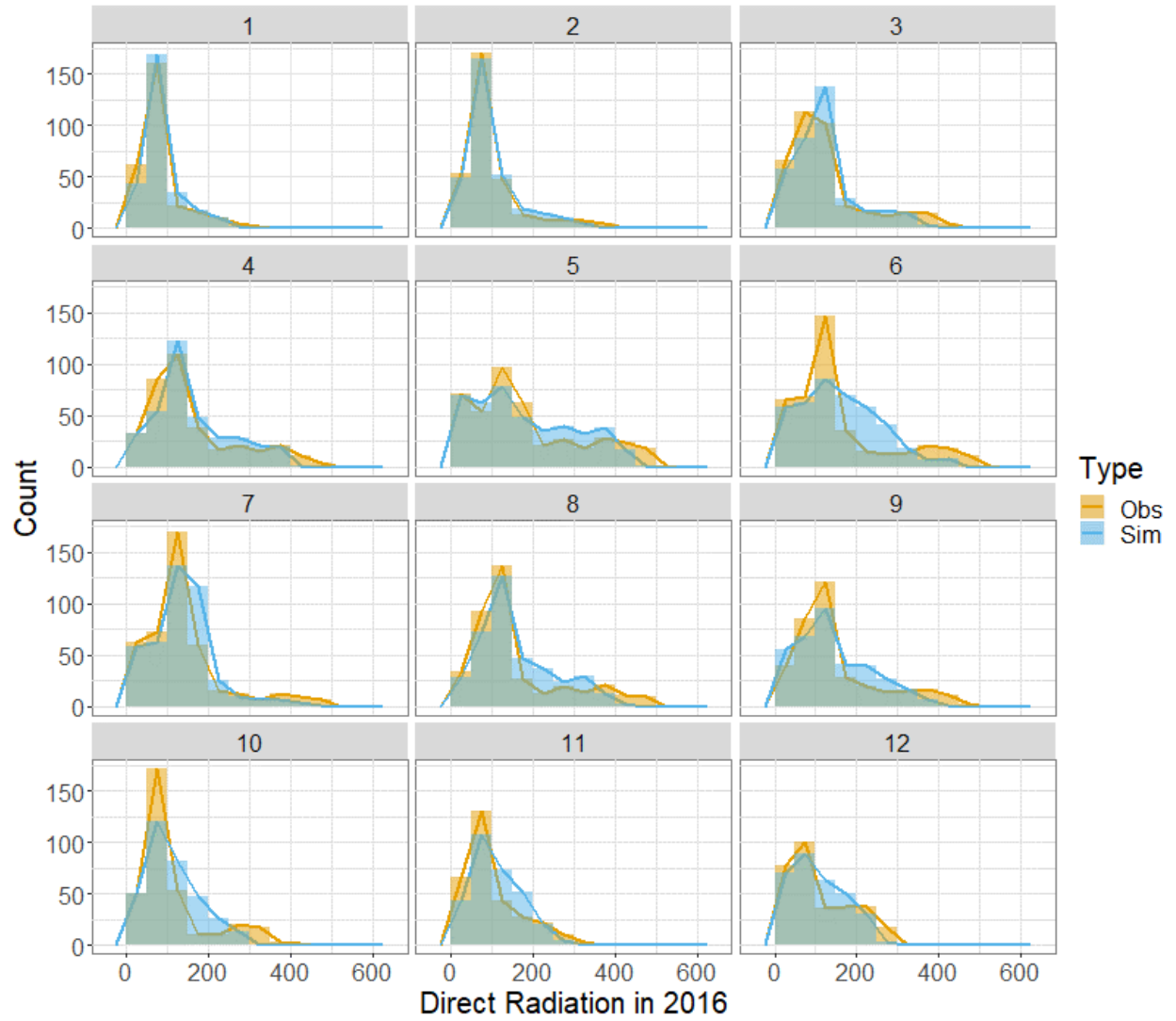


Figure 6. Histograms of observed and estimated direct radiation for training on a monthly basis

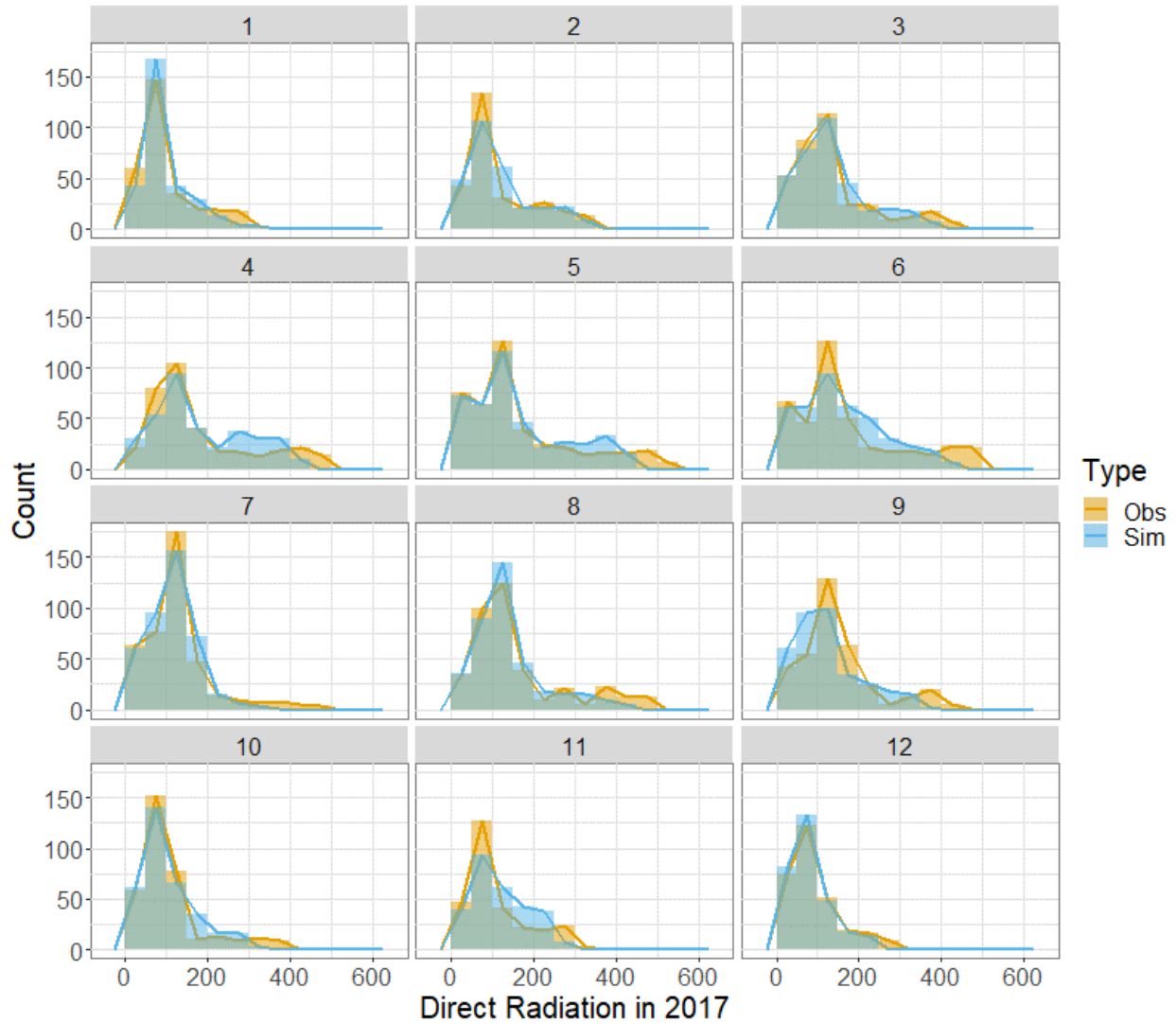


Figure 7. Histograms of observed and estimated direct radiation for testing on a monthly basis

Table 2. Weights between input nodes and the first hidden layer

W_{ji}	1st Hidden Layer →	Node1	Node2	Node3	Node4	Node5	Node6	Node7	Node8	Node9	Node10	Node11
		[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]
bias	[1,]	0.26953	1.92547	1.15530	-0.11683	-3.93657	3.13339	-1.23579	2.75554	17.11107	-52.09809	4.40571
Month	[2,]	-0.40412	2.24476	-3.95490	1.31128	-0.54023	-10.59450	0.01273	-2.29174	-16.20166	-139.82333	-0.76732
Hour	[3,]	4.44247	-0.99689	-0.49715	-2.67311	14.49029	0.18042	1.40504	-6.59462	-4.39436	96.71415	-5.20938
Tmp	[4,]	-0.70439	4.17799	0.58598	0.50596	-0.31805	-1.29744	-1.23071	3.69985	4.80134	-131.62736	-0.11794
Hmd	[5,]	-2.40226	4.18897	0.13675	2.29924	-0.76050	1.18118	-1.65618	-6.21246	1.01793	8.82604	-0.62947
Cld	[6,]	-1.15486	8.16143	-0.12807	1.25201	0.04877	-0.04446	-0.43658	-1.46038	-4.10042	-67.94317	-0.35142
Prs	[7,]	-0.41264	-5.01021	0.05279	0.11695	-0.60396	-0.88188	-0.90750	2.43522	-4.74803	197.92023	-0.06569
Wnd	[8,]	0.05566	0.61946	0.17447	-0.04953	-0.30380	-0.37525	-0.39261	0.04900	4.18411	97.07667	-0.05416
GR	[9,]	1.27943	-13.77166	0.98844	-18.73339	-1.54678	-0.06686	18.91579	2.61540	-60.16151	179.42532	-0.38634

Table 3. Weights between the first and the second hidden layers

W_{kj}	2nd Hidden Layer →	Node1	Node2	Node3	Node4	Node5	Node6	Node7	Node8	Node9	Node10	Node11
		[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]
bias	[1,]	0.04340	-1.53089	-1.74553	0.52753	-4.16646	-0.51228	3.27577	0.69547	5.12980	-0.68213	4.35228
Node1	[2,]	-6.87381	-5.59290	6.30890	-7.13632	-1.80805	-3.06660	-1.34308	-0.46671	8.81389	-2.38840	11.13643
Node2	[3,]	-0.41722	2.79233	-2.76909	-2.82235	-3.13354	0.36698	1.34479	-0.02419	-3.38450	-1.70596	-5.48031
Node3	[4,]	2.71196	5.27822	-1.31756	-3.05012	25.95493	-1.48770	-0.52923	-4.64924	0.59656	4.21131	-0.28511
Node4	[5,]	-3.57505	-5.17507	5.66458	-2.48064	7.07074	-4.70517	4.64898	-2.78355	15.78965	-22.89777	-1.77135
Node5	[6,]	5.22583	13.77269	1.79963	-0.02570	2.13326	1.76991	-4.67171	1.29552	-6.83545	2.83745	-0.14124
Node6	[7,]	-1.96413	-6.53633	-0.18478	3.29500	-13.34335	1.22196	60.12272	3.35530	3.02541	-0.37474	-0.91023
Node7	[8,]	3.24129	-6.89412	1.50058	0.57552	3.58710	1.97089	0.39779	0.41283	-1.46372	-1.86475	0.26800
Node8	[9,]	-8.92169	-1.25947	-0.27922	-2.92103	-5.60015	-4.11831	4.48342	2.50861	-4.31572	0.31646	-6.52362
Node9	[10,]	-0.73548	-0.12414	1.26422	-2.94405	-4.51363	-2.64742	4.72779	-0.10836	-1.17816	-1.45527	2.70781
Node10	[11,]	-0.17160	0.95037	-1.12042	-2.24296	-1.71296	-0.47068	0.86277	1.11698	-0.58864	0.46639	-1.95345
Node11	[12,]	-3.72782	-2.01684	-3.27077	19.13386	4.53249	-1.03995	-6.13349	-1.34773	-3.21864	0.97161	-5.65270

Table 4. Weights between the second hidden layer and output layer

W_{lk}	2nd Hidden Layer →	bias	Node1	Node2	Node3	Node4	Node5	Node6	Node7	Node8	Node9	Node10	Node11
		[1,]	[2,]	[3,]	[4,]	[5,]	[6,]	[7,]	[8,]	[9,]	[10,]	[11,]	[12,]
Node1	[,1]	0.04196	1.88897	1.57073	-1.85561	1.79212	-1.25434	-4.63158	-0.99232	-2.01125	-1.51798	2.75082	1.78893

Conclusions

ANNs were applied to estimate recent direct radiation, which is one of the hydrometeorological data for long-term simulation of streamflow using GSSHA model. ANN (8-11-11-1) was identified as the best-balanced model for the estimation through training and testing phases. The model showed high correlations between observed and estimated data for the training and testing with slight underestimation of high direct radiation.

The following ideas can be studied and applied to build a better estimation model using ANNs in the future.

1. Input and target data can be rescaled into a narrower range than the one used in the study to avoid generating values less than zero.
2. Combination of different activation functions for each layer can be explored to overcome the underestimation of high direct radiation and to improve extrapolation ability.
3. In the present study, the two hidden layers were given the same number of hidden nodes for a brief exploration of ANNs' application to estimate the hydrometeorological data during the model identification process. More detailed investigation for model identification can be performed with different numbers of nodes in each hidden layer.
4. It is recommended to explore other free and open-source software libraries such as TensorFlow and Apache MXNet for machine learning or deep learning to build estimation/prediction models

for the more detailed research described above. The open-source software libraries have greater variety of deep learning models than the package used for the present study. In particular, the Long Short-Term Memory (LSTM) networks, one of the deep learning models, have been applied to model rainfall-runoff relationship (Kratzert et al., 2018).

Furthermore, ANNs can be used for a plethora of prediction or classification problems posed by ERM.

Below are a list of other ERM applications:

1. There are areas outside the City of Austin Extraterritorial Jurisdiction (ETJ) that may impact water quality within the City of Austin proper. For instance, much of Onion Creek lies outside the COA ETJ. Knowing existing land use within the Onion Creek watershed can inform policy and decisions making by the COA. However, this land use is currently unknown or ambiguous. An ANN can be trained to classify areal land cover images outside COA ETJ to their respective land use.
2. ANNs can also be trained to classify future land use from areal land cover images outside COA ETJ.
3. ERM currently uses a diatomist to classify diatoms under a microscope. This can be a time-consuming process that can strain the diatomist's eyesight. ANNs can be trained to perform taxonomy using images from sampled diatoms.
4. ANNs can also be trained to predict BUG scores from a shorter simulated hydrograph.
5. ANNs can be trained to even predict the simulated hydrographs from the GSSHA model on a watershed given precipitation events. In other words, the precipitation events and hydrographs are used as input and target data to train/test the ANNs. Once the ANNs are built for a watershed, the simulation to generate hydrographs for new precipitation events would take much shorter time than the GSSHA model takes.

References

- Downer, C. W. and Ogden, F. L. (2006) Gridded surface subsurface hydrologic analysis (GSSHA) user's manual, Engineer Research and Development Center (ERDC), US Army Corps of Engineers.
- Fritsch, S., Guenther, F., and Wright, M. (2019) Package 'neuralnet', <https://github.com/bips-hb/neuralnet>.
- Glick, R. H., Gosselink, L., Bai, B. and Herrington, C. (2010) Impacts of Stream Hydrologic Characteristics on Ambient Water Quality and Aquatic Health in the Austin, Texas Area, City of Austin technical report SR-10-18.
- Herrington, C. S. (2017) Objective Zero: Improving the water quality mission objectives of the Watershed Protection Department, City of Austin technical report SR-18-02.
- Hodnett, M. and Wiley, J. (2018) R Deep Learning Essentials, 2nd Edition, Packt, 7-20.
- Hsu, K.-L., Gupta H. V., Goa, X., Sorooshian, S., and Imam B. (2002) Self-organizing linear output map (SOLO): An artificial neural network suitable for hydrologic modeling and analysis, Water Resources Research, 38.12, 1-17.
- Jin, Y.-H., Kawamura, A., Jinno, K., and Berndtsson, R. (2004) Dynamical behavior of multivariate time series for SOI, precipitation/Temperature in Fukuoka and its prediction by artificial neural networks, Memoirs of the Faculty of Engineering, Kyushu University, 64, 45-62.

- Jin, Y.-H., Kawamura, A., Park, S.-C., Nakagawa, N., Amaguchi, H., and Olsson, J. (2011) Spatiotemporal classification of environmental monitoring data in the Yeongsan River basin, Korea, using self-organizing maps, *Journal of Environmental Monitoring*, 13, 2886-2894.
- Kalman, B. L. and Kwasny, S. C. (1992) Why Tanh? Choosing a sigmoidal function, *Proceedings of the International Joint Conference on Neural Networks*, Baltimore, MD IEEE, New York.
- Kratzert, F., Klotz, D., Brenner, C., Schulz, K., and Herrnegger, M. (2018) Rainfall-runoff modelling using Long Short-Term Memory (LSTM) networks, *Hydrology and Earth System Sciences*, 22, 6005-6022.
- Maier, H. R. and Dandy, G. C. (1998) The effect of internal parameters and geometry on the performance of back-propagation neural networks: An empirical study, *Environmental Modelling & Software*, 13(2), 193-209.
- Maier, H. R. and Dandy, G. C. (2000) Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications, *Environmental Modelling & Software*, 15, 101-124.
- Moody, J. and Yarvin, N. (1992) Networks with learned unit response functions, *Advances in Neural Information Processing Systems 4*. Morgan Kaufmann, San Mateo, CA.
- Poff, L. N. and Ward, J. V. (1989) Implications of streamflow variability and predictability for lotic community structure: A regional analysis of streamflow patterns, *Canadian Journal of Fisheries and Aquatic Sciences*, 46, 1805-1817.
- Qazi, A., Fayaz, H., Wadi, A., Raj, R. G., and Rahim, N. A. (2015) The artificial neural network for solar radiation prediction and designing solar systems: a systematic literature review, *Journal of Cleaner Production*, 104, 1-12.
- Richter, A. (2011) Linking biologic metrics to hydrologic characteristics in Austin, Texas streams, City of Austin technical report SR-11-15
- Richter, B. D., Baumgartner, J. V., Powell, J., and Braun, D. P. (1996) A method for assessing hydrologic alteration within ecosystems, *Conservation Biology* 10(4), 1163-1174.
- Riedmiller, M. and Braun, H. (1993) A direct adaptive method for faster backpropagation learning: The RPROP algorithm. *Proceedings of the IEEE International Conference on Neural Networks (ICNN)*, 586-591. San Francisco.
- Scoggins, M. (2000) Effects of hydrology on bioassessment in Austin, Texas, City of Austin technical report SR-00-02.
- Sengupta, M., Xie, Y., Lopez, A., Habte, A., Maclaurin, G., and Shelby, J. (2018) The National Solar Radiation Data Base (NSRDB), *Renewable and Sustainable Energy Reviews*, 89, 51-60.
- Stewart-Koster, B., Olden, J. D., and Gido, K. B. (2014) Quantifying flow-ecology relationships with functional linear models, *Hydrological Sciences Journal*, 59(3-4), 1-16.
- Sørensen, B. (2017) Origin of renewable energy flows, *Renewable Energy (Fifth Edition)*, 39-218.

Appendix – A: R-script for data arrangement

```
# Load and attach add-on packages
library(tictoc)
library(plyr)
library(tidyverse)
library(lubridate)

tic() # Start timer
rm(list = ls()) # Clear the working environment

# Input the folder location in which Meteorological (MT) Data are located
pathName1 <- "C:/ . . . . . / . . . . . /NOAA-LCD"
# Input the folder location in which Global Radiation (GR) data are located
pathName2 <- "C:/ . . . . . / . . . . . /Solar_33NW"
# Input the folder location in which Direct Radiation (DR) data are located
pathName3 <- "C:/ . . . . . / . . . . . /NSRDB_CampMabry"
# Input the folder location and filename for output
pathName4 <- "C:/ . . . . . / . . . . . /DR_Prediction"
fileName <- "Data_for_Prediction.dat"

# Arrange MT
MT <- read.csv(file.path(pathName1, "LCD-20100101-20190430-CampMabry.csv"),
              header = F, skip = 1, stringsAsFactors = FALSE)
colnames(MT) <- c("Date", "Tmp", "Hmd", "Cld", "Prs", "Wnd")

# Remove duplicates based on dates
MT[, 1] <- substr(MT[, 1], 1, 13)
MT <- MT[!duplicated(MT[, 1]), ]
# Remove NAs
index <- matrix(NA, dim(MT)[1], 1)
for (i in 1:dim(MT)[1]) {index[i] <- any(is.na(MT[i, ]))}
MT <- MT[!index, ]
# Change the format of time information
MT[, 1] <- ymd_h(MT[, 1])
# Convert Cld to percentage (%)
MT$Cld <- strsplit(MT$Cld, ":")
for (i in 1:dim(MT)[1]) {
  nStr <- length(MT$Cld[[i]])
  x <- MT$Cld[[i]][nStr]
  nCld <- nchar(x)
  ifelse(nCld > 1,
        x <- str_sub(x, 1, 2),
        x <- x)
  ifelse(nCld > 1,
        MT$Cld[i] <- as.integer(strsplit(x, " ")[[1]][1]),
        MT$Cld[i] <- 0)
  ifelse(MT$Cld[i] > 8,
        MT$Cld[i] <- 100,
        MT$Cld[i] <- as.integer(MT$Cld[i]) * 12.5)
}
MT$Cld <- as.numeric(MT$Cld)
MT$Cld <- unlist(MT$Cld)
# Convert MPG to Knot
MT$Wnd <- MT$Wnd * 0.868976

# Generate a list of names of files in the folder for GR
files <- list.files(path = pathName2, pattern = "*.txt", full.names = T)
# Read global radiation data downloaded from NW33(?)
GR <- sapply(files, read.table, header = FALSE,
            stringsAsFactors = FALSE, simplify = FALSE)
# Arrange global radiation data
GR <- bind_rows(GR)[, c(1, 4, 5, 14)]
colnames(GR) <- c("Date", "YMD", "Hr", "GR")
GR$Date <- ymd(GR$YMD)
hour(GR$Date) <- GR$Hr / 100
```

```

GR <- GR[, c(1, 4)]

# Generate a list of names of files in the folder for DR
files <- list.files(path = pathName3, pattern = "*.csv", full.names = T)
# Read direct radiation data downloaded from NSRDB
DR <- sapply(files, read.table, header = FALSE, skip = 3,
             sep = ",", stringsAsFactors = FALSE, simplify = FALSE)
DR <- bind_rows(DR)
DR <- DR[, c(1:4, 6)]
DR[, 1] <- ymd_h(apply(DR[, 1:4], 1, paste, collapse = "-"))
DR <- DR[!duplicated(DR[, 1]), c(1, 5)]
colnames(DR) <- c("Date", "DR")

# Join MT, GR, and DR and remove NA
HMET <- join(MT, GR, by = "Date")
# Remove -99999 or NA from HMET$GR
HMET <- HMET[!(HMET$GR == -99999 | (is.na(HMET$GR))), ]
HMET <- join(HMET, DR, by = "Date")
HMET <- bind_cols("Year" = year(HMET$Date),
                 "Month" = month(HMET$Date),
                 "Day" = day(HMET$Date),
                 "Hour" = hour(HMET$Date),
                 HMET[, 2:dim(HMET)[2]])

HMET$Month <- sprintf("%2i", HMET$Month)
HMET$Day <- sprintf("%2i", HMET$Day)
HMET$Hour <- sprintf("%2i", HMET$Hour)
HMET$Tmp <- sprintf("%3i", HMET$Tmp)
HMET$Tmp <- noquote(format(HMET$Tmp, justify = "right"))
HMET$Hmd <- sprintf("%3i", HMET$Hmd)
HMET$Hmd <- noquote(format(HMET$Hmd, justify = "right"))
HMET$Cld <- sprintf("%3.1f", HMET$Cld)
HMET$Cld <- noquote(format(HMET$Cld, justify = "right"))
HMET$Prs <- sprintf("%2.2f", HMET$Prs)
HMET$Prs <- noquote(format(HMET$Prs, justify = "right"))
HMET$Wnd <- sprintf("%2.2f", HMET$Wnd)
HMET$Wnd <- noquote(format(HMET$Wnd, justify = "right"))
HMET$GR <- sprintf("%4i", HMET$GR)
HMET$DR <- sprintf("%3i", HMET$DR)

# Save the output
write.table(HMET, file.path(pathName4, fileName),
           quote = FALSE, row.names = FALSE, col.names = TRUE)

toc() # Compute and show elapsed time

```

Appendix – B: Output

Year	Month	Day	Hour	Tmp	Hmd	Cld	Prs	Wnd	GR	DR
2010	1	1	0	43	71	100.0	29.71	6.08	0	0
2010	1	1	1	39	70	50.0	29.75	6.08	0	0
2010	1	2	22	43	74	87.5	29.51	0.00	0	0
2010	1	3	0	42	79	87.5	29.49	0.00	0	0
2010	1	3	1	43	80	100.0	29.49	0.00	0	0
.
.
.
2017	12	31	21	29	78	100.0	29.87	6.95	0	0
2017	12	31	22	28	81	100.0	29.88	8.69	0	0
2017	12	31	23	28	81	100.0	29.90	8.69	0	0
2018	1	1	0	26	75	100.0	29.91	6.95	0	NA
2018	1	1	1	26	69	100.0	29.92	7.82	0	NA
2018	1	1	2	26	63	100.0	29.94	6.95	0	NA
.
.
.
2019	4	30	19	79	69	100.0	29.15	7.82	20	NA
2019	4	30	20	78	76	100.0	29.16	6.95	0	NA
2019	4	30	21	77	79	100.0	29.16	8.69	0	NA
2019	4	30	22	77	79	100.0	29.17	6.95	0	NA
2019	4	30	23	77	79	100.0	29.17	6.08	0	NA