

Multivariate monthly water demand prediction using ensemble and gradient boosting machine learning techniques

Paul Banda, Muhammed. A Bhuiyan. Kevin Zhang, and Andy Song

Abstract — Water management planning requires reliable and accurate water demand forecasting. Water demand prediction is affected by variables, such as climate, socio-economic, and demographic data. This paper investigates urban monthly average water demand prediction, using classical, ensemble, and gradient boosting-based machine learning models, using the available monthly water demand, climatic, economic, and demographic data. Three train-test data split schemes on water demand timeseries were considered to determine the effect of data size on water demand prediction. Sensitivity analysis was employed to reduce input feature dimensionality while maintaining model accuracy. A univariate timeseries (water demand only) produced R2 scores up to 0.91, which increased to 0.94 with the addition of calendar and climatic features. Increasing the training data size from 70% to 90% improved the RMSE and MAE scores by ensemble and gradient boosting methods, with the random forest and the AdaBoost models showing improvements of up to 69%. The sensitivity analysis revealed a successful input reduction scheme from a potential 17 input attributes to seven inputs. Gradient boosting models showed robust and faster execution time, especially with the increase in training data, which is attractive for medium-term urban water demand forecasting.

Keywords — water management, water demand, machine learning, ensemble, gradient boosting

I. INTRODUCTION

Water demand prediction is an essential paradigm for water resource planning and utilisation in the water sector. It can strategically inform water sources and treatment plants' development and expansion for the present and future generations through water demand prediction efforts. Water utility authorities make strategic decisions from water demand forecasting models. Many countries in the world are expected to face water shortages. Global population growth, climatic change, and

economic advancement continue to pressure the already scarce water resource. These factors point to the need for careful planning and management of the available water resources [1].

Knowledge-driven methods were used to develop short-term water demand prediction models using temperature and other weather forecasting information [2, 3]. Knowledge-driven methods consider explaining and accounting for the factors that affect water demand, such as population, economy, temperature, and rainfall, among others. Data-driven techniques are broadly categorised into traditional and artificial intelligence-based (A.I.) procedures account for most water demand modelling studies.

A. AI-based water demand prediction studies

Machine learning techniques provide a useful approach to water demand prediction by learning the structures in the input and output data patterns from historical data.

B. Classic and hybrid ANN

A wavelet analysis on water demand timeseries combined with an ANN (wavelet-ANN model) outperformed the MLR, multiple nonlinear regression (MNL), ARIMA, and ANN for daily water demand forecasts using weather variables [4]. Rainfall amount was a useful variable for weekly municipal water demand forecasting using the ANN model [5]. A seasonal wavelet transform-based model, combined with ANN, was effective in monthly water consumption prediction, compared to a discrete wavelet transformed ANN [6]. Prediction models were developed for monthly water demand forecasts for a city in Turkey, using neural network variations on eight yearlong datasets, with correlations between actual and predicted values reaching as far as 0.78 [7]. The neural network variations and K-nearest neighbour techniques were used for daily, weekly,

Manuscript received 3 September, 2021; revised 13 December, 2021; accepted 5 January, 2022; published 19 January, 2022.

This is an open access article distributed under the terms of the Creative Commons Attribution 4.0 licence (CC BY <http://creativecommons.org/licenses/by/4.0/>). Unrestricted use (including commercial), distribution and reproduction is permitted provided that credit is given to the original author(s) of the work, including a URI or hyperlink, this public license and a copyright

notice. This article has been subject to single-blind peer review by a minimum of two reviewers. Digital Object Identifier <https://doi.org/10.55066/proc-icec.2021.14>

Paul Banda is a research student with the Royal Melbourne Institute of Technology, Australia (e-mail: s3639345@student.rmit.edu.au)

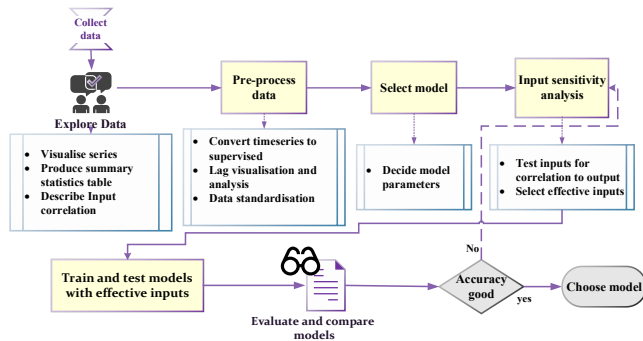


Fig. 1. A schematic illustration of the monthly average water demand forecasting.

and monthly water demand predictions, using nine yearlong datasets in Iran [8]. A constant rate model [9] was successfully used to model water demand in the United Arab Emirates. A dynamic ANN using weather variables outperformed the ARIMA, and traditional ANN models for hourly, daily, weekly, and monthly urban water demand forecasting [10].

C. Fuzzy and hybrid systems

A 9-year water consumption profile was used for water consumption prediction for Istanbul city using a fuzzy logic system that performed better than the ARIMA model [11]. Fuzzy cognitive maps outperformed the ARIMA, linear regression, Holt-Winters, and ANN models in daily water demand forecasting [12]. A parallel adaptive weighting strategy [13] was effective in short-term water demand forecasting. A successful fuzzy inference system tool was designed for monthly agricultural and rural water reuse [14]. A hybrid fuzzy neural network model was proposed in [15] to forecast daily water consumption in Tehran. An ANFIS model was proposed to predict the hourly water demand in northern China [16]. Fuzzy and neuro-fuzzy models were developed for daily water demand forecasting in Tehran using climatic and calendar variables [17].

D. Deep learning techniques

Genetic programming techniques in water demand forecasting were reviewed in [18]. Extreme learning machines were used for short-term urban water demand forecasting [19], while a deep belief network proved useful for daily forecasting of water demand [20]. Most recently, hourly and daily urban water demand forecasting was conducted using the popular Long-short-term memory (LSTM) deep learning model [21]. The review paper [22] summarises the available studies that have employed traditional and AI-based techniques.

E. Contribution of the study

Ensemble and gradient boosting tree-based machine learning methods have demonstrated great success in different application domains [23]. However, these ensemble and gradient boosting methods have been scarcely applied for short-term water demand modelling [16], [17], [18], and rarely applied for medium-term

(monthly) water demand forecasting. Despite the popularity of deep learning models, machine learning models are still competitive, particularly in small training data situations, shorter training time, and minimal parameter tuning requirements. This study discusses ensemble learning tree-based algorithms for monthly water demand prediction. Water demand data is challenging to collate because of financial and time constraints; thus, determining the most critical inputs for creating accurate water demand models is an invaluable process. In this study, in addition to developing robust water demand prediction models, a well-defined inputs selection process that aids model training efficiency while maintaining prediction accuracy is explained.

II. PREDICTIVE ALGORITHMS FOR WATER DEMAND FORECASTING

This section gives a brief overview of the machine learning algorithms used for water demand prediction. Prediction of water demand uses regression analysis to determine the input-output relationship that is, to ascertain the relationship between the independent and the dependent variables. Five models comprising the classical and ensemble predictive machine learning algorithms, namely, multiple linear regression (MLR), decision tree, random forest (R.F.), AdaBoost, and LightGBM, were used to map the relationship between inputs and outputs. The need to explore more algorithms and water demand prediction accuracy improves, making up the primary consideration for selecting these models. Furthermore, the encouraging prediction performance results in similar water demand studies [32-35] inspired further investigation of these and other models. A flow chart on choosing the monthly average water demand modelling process is shown in Fig. 1. It begins with data collection, understanding the data, and visualising the timeseries and drawing up data summary statistics. The next step is to pre-process the data to get it ready for modelling, which comprises steps like standardising the data and converting the timeseries to supervised learning. The model selection step includes deciding on the model parameters, the input sensitivity analysis, and the model's training and testing step. Finally, the model is evaluated, and if the results are not satisfactory, the process reverts to sensitivity analysis again (dashed line).

A. Random forest model

Random forest (R.F.) is an ensemble-based learning algorithm used for regression and classification problems [24]. R.F. is an ensemble of models that uses the decision tree approach to data collection. R.F. overcomes the decision tree's instability problem by using a set of trees rather than a single tree for prediction. Initially, an individual tree is trained by taking note of a random subset of observations. A random subset of the variable is then considered to split the decision, thus creating a diverse set of trees essential for improving the ensemble model's

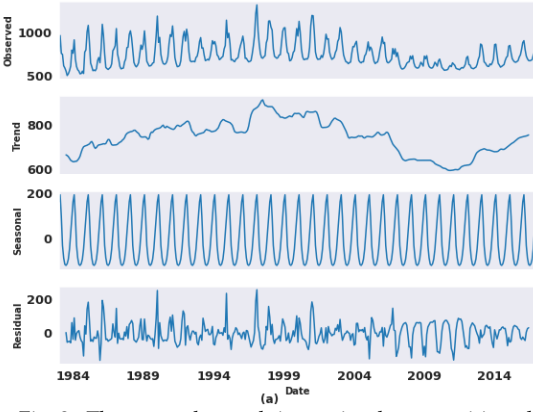


Fig. 2. The water demand timeseries decomposition elements.

overall prediction performance. The use of high diversity trees complements the instability issue of the decision tree by combining the multiple trees' predictions. In perspective, an ensemble of H decision trees,

$$T_1(X), T_2(X), \dots, T_z(X),$$

where $X = x_1, x_2, \dots, x_n$ is a n -dimension input vector.

The final ensemble has H outputs:

$$Y'_1 = T_1(X), Y'_2 = T_2(X), Y'_3 = T_3(X), \dots, Y'_H = T_H(X) \quad (2)$$

where Y'_H is the prediction value by decision tree number H . The average score of all the single randomly generated decision trees is defined as Y' .

R.F. models compensate for the decision tree weaknesses of instability by using results from a set of trees.

B. Adaptive Boosting (AdaBoost) ensemble model

Boosting in ensemble learning models attempts to create more robust models from several weaker regressors or classifiers. To correct the original first model's errors, boosting creates more models until the maximum number of models is created or required training and data accuracy is attained. AdaBoost is primarily used to boost any machine learning model's performance, primarily weak learners [25]. Decision trees are the popular candidate models considered for boosting and are adopted in this study.

Using the training set $T = (x_i, y_i)$ the AdaBoost regressor model relates input values in $x_i \in \mathbb{R}^D$.

To the i th response value $y_i \in \mathbb{R}$, $i = 1, \dots, N$ on the prediction function [26]

$$f'(x) = \inf \left\{ y \in Y : \sum_{t: h_t \leq y} \log \frac{1}{\beta_t} \geq \frac{1}{2} \sum_t \log \frac{1}{\beta_t} \right\} \quad (3)$$

where $\inf$ represents the infimum of the resulting set, h_t is the t^h base learner, and β_t is the coefficient of the t^h base learner.

C. Light Gradient Boosting (LightGBM) ensemble model

The LightGBM algorithm is another gradient boosting decision tree (GBDT) technique that determines the split value by using the gradient-based one side sampling (GOSS) and exclusive feature elimination [27]. The

LightGBM algorithm grows its trees leaf-wise (starting with the best) in which the leaf with maximum delta loss is chosen to grow. Leaf-wise tree growth is different from the depth-wise tree growth implemented by conventional ensemble techniques in that it does not calculate the information gain of all the samples, which allows it to have faster convergence and improved accuracy. However, leaf-wise tree growth often results in overfitting when data is small. There is a need to guard overfitting by adjusting parameters, for example, the minimum data in a leaf. The optimisation method of the LightGBM algorithm aims to retain and preserve some of the smaller gradient samples (one-side sampling) while keeping all larger gradient samples, reduces the search space, and strengthens the information gain from the smaller gradient samples. A more detailed explanation of the operation of the LightGBM algorithm is given in [27].

D. Extreme Gradient Boosting (XGBoost) ensemble model

XGBoost is an alternative framework for implementing gradient-boost trees that use histogram-based statistics to represent discretised feature values [28]. XGBoost is popular in algorithm competitions, such as Kaggle, because it is a fast, scalable, and efficient tree boosting stochastic gradient machine learning algorithm. In this study, the XGBoost algorithm is adopted for timeseries forecasting. The XGBoost model is an ensemble of new trees that aim to correct the errors of other trees within the model. While the gradient boosting model uses only the first-order derivative information during model training, the XGBoost implements the more accurate Taylor expansion, leveraging the cost function's first-order and second-order derivatives information. The XGBoost model handles well defined missing values processing strategies making it robust. Also, the use of the L1 and L2 regularisation terms ensures stronger generalisation and prevents overfitting. The finalised loss function can be represented as below [28]. While the GBDT uses only the first-order derivative information when training the model, the XGB performs the Taylor expansion of the cost function and uses the first-order and second-order derivatives information. A simplified XGBoost objective function at step t , which is a sum of quadratic functions of a variable, can be written as [28],

$$L_t = \sum_{i=1}^n [g_i f_t(x_i) + \frac{1}{2} (h_i f_t^2(x_i))] + \Omega(f_t) \quad (4)$$

where g_i and h_i are first and second-order gradient statistics on the loss function, f_t is the function variable, x_i represent the input observation, Ω is the regularisation term.

III. DATA COLLECTION

Melbourne is a coastal city located in southeastern Australia with a population of almost 5 million people covering an area of 9.992 km² of land. Melbourne's water is mainly sourced from protected catchments and the

states' forests. According to the Koppen climate classification, Melbourne city has a temperate oceanic climate. The city warms up in summer with mean temperatures between 14 - 25.3°C, and the winter averages from 6.5 - 14.2°C.

Melbourne's water demand (W.D.) dataset does not contain missing values; all the calendar timestamp values are present. A total of twelve potential climatic and economic input variables were investigated to test their effect on water demand prediction. These input variables are temperature (T°C), rainfall (Rn) in mm, solar radiation (Rs) in M.J./m², evaporation (Ev) in mm, water price in \$, gross domestic product (GDP), water restriction levels 1-4 (e.g. RL1, RL2, RL3, and RL4), population size (Pop) and previous month water demand. In addition to these input variables, calendar inputs (Cals), comprising the quarter of the year, the month of the year, the day of the week, the day of the year, week of the year (i.e. a total of 5 inputs), were also included to bring the number of the total input to 17 variables. The output is chosen as the current water demand values.

A. Dataset visualisation

This section visualises the water demand timeseries' temporal structure and the observations' distribution using plots to learn and derive useful insights for use during the modelling process, such as model choice. Fig. 2 shows the decomposed water demand timeseries and its elements like the trend, seasonality, and residuals. The line plots show an undulant water demand profile each year, with high demand observed towards the end and beginning of the year. In the trend plotting, an increasing trend can be seen up to 1997, and then a decreasing trend takes precedent until 2011. After 2011, a rising water demand trend again begins to manifest. A robust seasonal fluctuation is evident in the timeseries, while the residuals are highly irregular in the pattern - all these factors present challenges during the modelling process. The spike in 1997 may be attributed to the start of drought in the city, leading to the high-water demand.

The available dataset ranges from January 1983 to December 2016, equivalent to 34 years of monthly average observations (408 monthly observations).

Melbourne's mean water demand was 744.3 ML, with a maximum water demand of 1325 ML obtained during the peak of the summer season. The lowest water demand value of 500.8 ML was recorded during the winter season. The heaviest average monthly rainfall event recorded 186.8 mm, while the peak air temperature was 30.50 °C.

B. Selection of effective lags for water demand modelling

The formal definition of a lag can be expressed as for data values, $Y_1, Y_2, \dots, Y_N$, as the k -period (or k th) lag of the value Y_i that occurred k time points before the time i :

$$\text{Lag}_k Y_i = Y_i - k \quad (4)$$

A lag plot shown in Fig. 3 is a timeseries visual analysis technique used to identify effective lags. This lag plot shows a water demand timeseries scatter plot and its lagged values. The lag plot can check correlation possibilities between water demand this month and the previous lagged months and check for outliers. The correlation between water demand and its lagged values can be positive or negative, while an unidentifiable pattern represents a no relationship. Lags 1, 12, and 24 of Fig. 3 show moderate correlation, while lag 6 shows a weak correlation to the water demand timeseries (y_t).

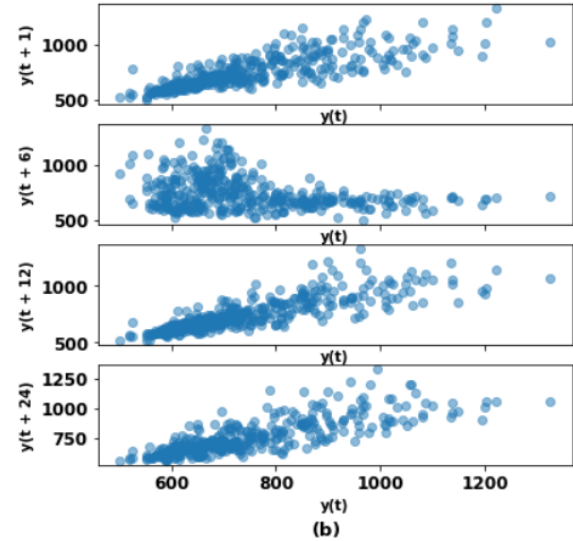


Fig. 3. A lag scatter plots of the monthly water demand timeseries.

Following the presented visual analysis technique, lags 1, 12, 24, 36, and 48 were selected for use as inputs and were combined with the climate and calendar inputs.

C. Variable correlation and feature selection

The Pearson correlation coefficient [29] describes the linear relationship between two Normally distributed datasets. The coefficients vary between -1 and +1, and a zero value implies no correlation, while correlations of -1 and +1 point to an exact linear relationship. Positive correlations suggest an increase in x with an increase in y and vice versa.

Both positive and negative correlations exist among the available input datasets, with rainfall, water price, GDP, and population being negatively correlated to water demand. Evaporation and temperature values have the highest correlations to water demand. The results from the correlation table assist in input selection and dimensionality reduction.

The redundant features in the dataset can negatively influence model accuracy, training time and result in over-fitting. There is a need to select features that significantly affect the prediction output of choice. There is a strong correlation among climatic variables, like, solar radiation and evaporation, temperature and evaporation, and

temperature and solar radiation. The high correlation necessitates using just one input (from the three inputs). Using all the three inputs introduce redundancy, as these inputs will have an almost similar effect on water demand. Following these input datasets, strong correlations exist between population and GDP, and price and population necessitate selecting one input from these two inputs. Overall, from 5 highly correlated potential variables, only 2 (e.g. population and temperature) were chosen as representative variables due to the relative ease of obtaining those data.

D. Re-framing series to supervised learning

The water demand timeseries dataset can be restructured into a supervised learning problem using the Pandas shift function [30], which takes in the values 1, 12, 24, 36, and 48 (previous time step values) to predict the next month's water demand value (next time step value). The Pandas shift function helps create columns pushed forward or backward for both lag observations and forecast observation.

E. Water demand timeseries training and testing set split

Timeseries forecasting involves making predictions into the future, with the training and testing set used for model creation and model evaluation. The test set should comprise the most recent data to simulate the natural real world set-up. In a timeseries, random sampling is not ideal as it creates a look-ahead scenario, where future water demand observations may be contained in the training set after random sampling. The other shortfall of this splitting strategy (i.e. random splitting) negates the temporal component embedded into the timeseries. Therefore, timeseries data should be split following the temporal order. A multiple train/test split strategy of 70-30, 80-20, and 90-10 following the water demand timeseries' temporal order are adopted in this study. This split strategy provides a more robust estimate of the expected model performance across the various data size.

F. Data transformation

The considerable differences in numerical ranges between the input and output values require that the dataset be standardised. Standardisation scales each feature such that the distribution is centred around 0, with a standard deviation of 1. Standardisation allows comparability among input datasets, and it also enhances the training efficiency of models since the numerical condition of optimisation is improved. As such, the mean and standard deviation for each feature is calculated, then the feature is scaled based on (5);

$$z = \frac{x_i - \mu}{\sigma} \quad (5)$$

where z is the standardised value, x_i is the observed water demand value, μ is the mean, and σ is the standard deviation of the water demand dataset.

All machine learning algorithms were implemented in the Python programming language using the Scikit-Learn library [31]. All model development and experimental tasks were conducted on a Windows operated computer (Intel Core i5 2.40GHz 8GB RAM).

Assessment of the models' skill is done using the mean absolute error (MAE), root-mean-square error (RMSE), and R-squared (R^2) [10] evaluation metrics.

IV. RESULTS AND DISCUSSION

The need to discover more efficient and accurate water demand forecasting models motivated the selection and testing of s developed machine learning models for the average monthly water demand prediction task. Data were available from January 1983 to December 2016, where three splits of training and testing data were created: 70-30, 80-20, and 90-10, respectively. Seventeen potential inputs were tested on their effect on water demand, and the obtained results were discussed in this section. Following a series of extensive experimentation to identify the most significant lag, the study noted that lag 12 produced the best performance according to the set evaluation metrics. The results discussed in this section are for lag 12 only to allow for brevity and adequate discussion on the findings.

A. Input sensitivity analysis

An understanding of the essential input datasets for accurate water demand prediction is crucial for efficient model performance and reducing data collection costs. The selected and tested input combinations comprised of

- (i) previous water demand (W.D.) only,
- (ii) W.D. and calendar inputs (Cals),
- (iii) W.D., Cals and Temperature (T),
- (iv) W.D., Cals, T, solar radiation, (S.R.) and evaporation (Ev),
- (v) Cals, T, population (Pop), restriction levels (R.L.), rainfall, and
- (vi) all available inputs (all).

Input combination (i) simulates the limited data scenario where only the water demand variable is available. Input combination (ii) takes advantage of feature engineering, wherein calendar inputs are generated in addition to water demand values only. The effect of temperature is tested in the input combination (iii), which simulates the most common and readily available input combination. The impact of using climatic variables only is tested in (iv). The effect of dimensionality reduction is tested in input combination (v), where only those inputs with no strong correlations with other inputs are considered. The temperature has high correlations with evaporation and solar radiation, and so does population and water price, in which case only temperature and population are chosen as representative inputs. Lastly, input combination (vi) simulates all available datasets used regardless of any relations. Fig. 4

shows the relationships between input size and the prediction RMSE values, with the number of inputs on the x-axis and RMSE on the y-axis.

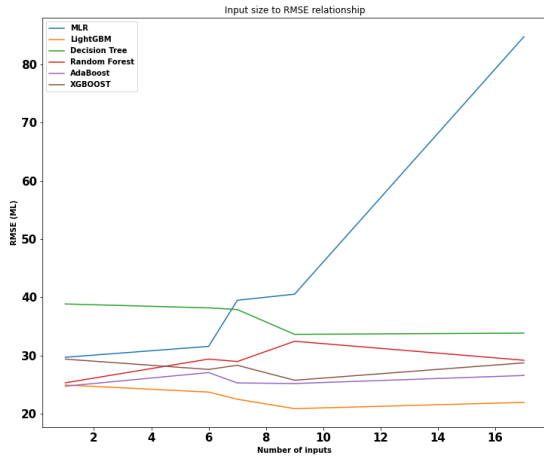


Fig. 4. Model performance against the number of features for the 90:10 train test split scheme.

Apart from the MLR model, all other models show minimal variation in RMSE scores (model performance) with the increasing number of inputs, as seen in Fig. 4. The MLR model scored the highest performance with one input (previous W.D. at lag 12) for all tested training dataset sizes. Any further addition of input results in a deterioration of model performance. The MLR model captures the linear relationship between inputs and outputs, with input combination (i).

Further addition of other variables resulted in degradation in model performance. It is advisable to use the input combination (i) only when considering the linear regression model. This finding clues to nonlinear relationships between inputs and water demand. The rest of the models record their lowest RMSE values between seven inputs (i.e. combination (iii) of W.D., Cals, and T) and nine inputs (i.e. combination (iv) of W.D., Cals, T, SR, and Ev) regardless of training data size.

The best performing model (LightGBM) has input combination (iii) of W.D., Cals, T, Rs, Ev with the lowest RMSE score (20.86 ML) and the highest R^2 value (0.94). This result is consistent with the 70-30 and 80-20 testing splits. However, for the LightGBM, an input combination without Rs and Ev inputs suffer a loss in performance of 7% RMSE and 2% MAE evaluation metrics. Compared to the opportunity cost of data collection and model efficiency, a model using W.D., Cals, and T inputs are still favourable. Table II shows the models' performance results in the testing phase following the sensitivity analysis exercise, and the results confirm the benefits of eliminating redundant features. The LightGBM model and most models using all inputs record higher RMSE scores, relative to fewer inputs, and this finding is consistent across all data sizes and tested models. It can be concluded that the dimensionality reduction efforts through the sensitivity analysis exercise are beneficial. There was no

significant loss in accuracy due to using fewer input variables.

B. Effect of data size

A large training set risks model overfitting, while smaller training data size results in high variance model parameters. As indicated earlier, the evaluation metrics of three training/testing splits contrived from the Melbourne City monthly water demand dataset are shown in Table II for the tested models and input combinations. A general improvement in all models' performance occurs with the increasing training data size and the number of input combinations. RMSE and MAE values for the random forest and AdaBoost were improved by up to 69% as the training data size increased from 70% to 90%. The 90-10 training/test split strategy had the best results for all tested models and input combinations, followed by the 80-10 and lastly the 70-30 split strategy. The observation that all models' accuracy tends to increase with increased training data size may be attributed to the existence of more recent water demand observations in the training data. Thus, the latter values of water demand (closer to 2016) are more relevant to predicting water demand than earlier observations.

The superior LightGBM model hyperparameters that provided the best fit to the data comprised a learning rate of 0.04, with five leaves forming a tree with a maximum depth of three. At the same time, a bagging frequency of five and a feature fraction of 0.7 at each iteration ensured quick convergence. The LightGBM model also demonstrated a faster training time among the tested ensemble methods. Random Forest and AdaBoost models offered competitive accuracy; however, they experienced longer training time than the LightGBM model, making the LightGBM model the choice model in this prediction task.

C. Model performance against actual observation

Fig. 5 shows the predictions of all the models against the actual water demand values for the duration of the testing phase period using the 90-10 train-test split scheme. All the models have learned the pattern of rising and falling water demand and trends. However, the MLR and decision tree models deviate from the actual observations, with the latter models almost always underestimating all lower water demand values. On the other hand, the R.F. model overestimated the observed dataset's higher water demands. However, the gradient boosting-based models, the LightGBM, XGBoost, and AdaBoost models, showed robustness as they closely followed the actual observations' shape more accurately throughout the dataset's length

TABLE I
WATER DEMAND PREDICTION RESULTS IN THE TESTING PHASE USING DIFFERENT DATA TRAIN-TEST SPLITS

70-30 split	MLR			LightGBM			Decision Tree			Random Forest			AdaBoost			XGBoost		
Inputs	RMSE	MAE	R2	RMSE	MAE	R2	RMSE	MAE	R2	RMSE	MAE	R2	RMSE	MAE	R2	RMSE	MAE	R2
WD, only	39.21	31.4	0.76	39.8	28.6	0.76	64.11	49.58	0.37	44.92	34.02	0.69	40.95	30.44	0.74	42.64	32.23	0.72
WD, Cals	55.5	40.18	0.53	51.78	37.6	0.59	77.53	57.73	0.08	76.16	54.53	0.11	61.04	44.2	0.43	56	40.65	0.51
WD, T, Cals	52.21	40	0.58	50.74	36.9	0.6	79.98	61.79	0.02	79.64	57.69	0.03	58.92	44.98	0.47	48.57	37.25	0.64
WD, Cals, T, SR, Ev	67.45	50.52	0.3	61.01	42.8	0.43	76.5	59.05	0.1	87.88	63.82	-0.18	71.31	51.28	0.22	52.17	40.4	0.58
Cal, T, Pop, RL, Rn	2*e16	1*e16	-6.8	56.55	30.7	0.57	83.51	68.42	-	84.95	64.04	-0.11	66.72	52.76	0.32	50.3	39.68	0.61
All inputs	209.61	179.68	-5.7	61.16	42.3	0.51	83.51	68.43	-	89.96	68.43	-0.24	79.53	61.45	0.02	58.61	43.66	0.47
80-20 split																		
WD, only	31.83	22.85	0.87	32.89	24.3	0.87	37.58	29.54	0.83	33.56	22.69	0.86	33.23	23.52	0.86	34.66	24.39	0.85
WD, Cals	37.49	29.01	0.83	30.13	23.3	0.89	38.98	29.42	0.81	32.57	23.1	0.87	32.81	23.12	0.87	29.03	21.58	0.9
WD, T, Cal	40.12	30.67	0.8	28.44	21.2	0.9	39.02	30.63	0.81	30.79	22.59	0.88	30.6	22.04	0.88	28.43	21.06	0.9
WD, Cals, T, SR, Ev	43.39	33.83	0.77	28.25	21.6	0.9	40.81	29.65	0.79	33.06	23.26	0.87	30.38	22.8	0.89	30.38	21.73	0.89
Cal, T, Pop, RL, Rn	8 *e14	5*e14	-	29.73	23.3	0.89	40.32	30.4	0.8	31.34	23.41	0.88	31.37	25.07	0.88	33.08	26.44	0.87
All inputs	104.37	79.67	-	30.49	23.2	0.89	40.9	29.73	0.79	31.79	23.22	0.88	30.13	23.66	0.89	32.99	26.19	0.87
90-10 split																		
WD, only	29.69	23.12	0.87	24.92	18.6	0.91	38.85	32.52	0.79	25.3	19.02	0.91	24.72	19.92	0.91	29.37	22.05	0.88
WD, Cals	31.56	25.82	0.86	23.71	18.3	0.92	38.18	28.24	0.79	29.38	20.97	0.88	27.06	21.59	0.9	27.6	21.64	0.89
WD, Cals, T	39.48	31.43	0.78	22.48	16.8	0.93	37.89	31.78	0.8	28.96	21.13	0.88	25.28	20.5	0.91	28.31	21.45	0.89
WD, Cals, T, SR, Ev	40.54	32.05	0.77	20.86	16.4	0.94	33.63	26.14	0.84	32.44	20.72	0.85	25.18	20.09	0.91	25.74	20.91	0.91
Cal, T, Pop, RL, Rn	4*e16	4*e16	-2.6	21.57	16.3	0.93	39.22	29.95	0.78	27.49	19.72	0.89	26.62	21.41	0.9	28.27	21.24	0.89
All inputs	84.79	67.71	-	21.94	16.9	0.93	33.84	26.3	0.84	29.18	21.23	0.88	26.56	21.42	0.9	28.73	21.35	0.88

T

The state-of-the-art R.F. model did show some competitive performance. However, it did overestimate the peak values of water demand, as shown in Fig. 5. The R.F. model could not supersede the gradient boosting models' performance at all the considered train-test splits. The R.F. model attempts to reduce the complexity of the models (bagging). In contrast, gradient boosting techniques increase model complexity which was favourable for this dataset. As a result, the gradient boosting models by adding more complexity superseded the R.F. model. However, because the R.F. model is more straightforward to tune than its gradient boosting counterparts, it remains a reasonable choice for this prediction task.

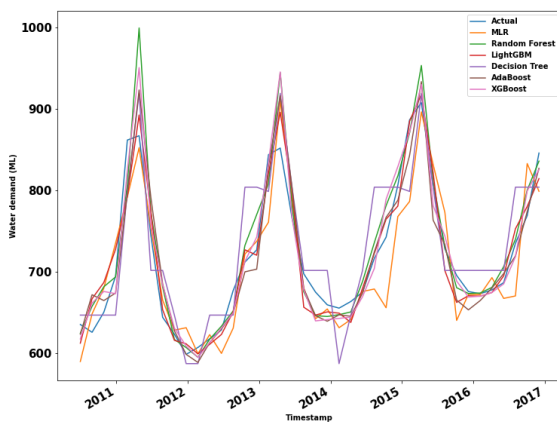


Fig. 5. A comparative evaluation of model performance in the testing phase using the 90-10 train test split.

V. CONCLUSION

Accurate monthly water demand prediction models are crucial for strategic planning and resource allocation for city water supply. Most water demand studies have been conducted for essentially short-term water demand predictions, with limited studies examining monthly (medium-term) water demand predictions using the tree-based gradient boosting methods. The monthly water demand studies considered mostly univariate water demand timeseries using the smaller datasets because of water demand data collection difficulties. This study investigated the multivariate monthly water demand prediction using three different training data sizes and in addition, conducted a sensitivity analysis to determine the most critical factors affecting urban monthly water demand in Melbourne. Seventeen potential inputs comprising climatic, economic, demographic, calendar, and previous water demand observations were investigated on their effect on water demand prediction.

The sensitivity analysis was conducted for input dimensionality reduction and model training efficiency improvement, leading to data collection cost reduction. The sensitivity analysis exercise determined that input combinations comprising the previous year's water demand observations, calendar inputs, and climatic inputs, were essential for improving the monthly water demand prediction. The gradient boosting group of machine learning models, mainly the LightGBM model offered efficient, robust and satisfactory accuracy, with the R2 score reaching 0.94. The study observed that, for the city of Melbourne, more current observations of water demand and climatic inputs (closer to the year 2016) are critical for improving monthly water demand prediction. In situations where only Melbourne's monthly water demand observations are available, reliable monthly water predictions with R² scores of up to 0.91 are still possible,

particularly with the gradient boosting LightGBM model and XGBoost models.

The development of monthly water demand prediction models is greatly hindered by data availability and collation, which has limited research advancement in medium to long-term water demand prediction studies. Future works will investigate transfer learning, which allows the transfer of information from water demand data-rich cities to cities with limited data to enhance their (limited data cities) model prediction accuracy, will be investigated.

REFERENCES

- [1] Matrosov, E.S., S. Padula, and J.J. Harou, Selecting portfolios of water supply and demand management strategies under uncertainty—contrasting economic optimisation and ‘robust decision making’ approaches. *Water resources management*, 2013. 27(4): p. 1123-1148.
- [2] Tian, D., C.J. Martinez, and T. Asefa, Improving short-term urban water demand forecasts with reforecast analog ensembles. *Journal of Water Resources Planning and Management*, 2016. 142(6): p. 04016008.
- [3] Adamowski, J., K. Adamowski, and A. Prokoph, A spectral analysis based methodology to detect climatological influences on daily urban water demand. *Mathematical Geosciences*, 2013. 45(1): p. 49-68.
- [4] Adamowski, J., et al., Comparison of multiple linear and nonlinear regression, autoregressive integrated moving average, artificial neural network, and wavelet artificial neural network methods for urban water demand forecasting in Montreal, Canada. *Water Resources Research*, 2012. 48(1).
- [5] Bougadis, J., K. Adamowski, and R. Diduch, Short-term municipal water demand forecasting. *Hydrological Processes: An International Journal*, 2005. 19(1): p. 137-148.
- [6] Altunkaynak, A. and T.A. Nigussie, Monthly water consumption prediction using season algorithm and wavelet transform-based models. *Journal of Water Resources Planning and Management*, 2017. 143(6): p. 04017011.
- [7] Firat, M., M.E. Turan, and M.A. Yurdusev, Comparative analysis of neural network techniques for predicting water consumption time series. *Journal of hydrology*, 2010. 384(1-2): p. 46-51.
- [8] Ghiassi, M., F. Fa'al, and A. Abrishamchi, Large metropolitan water demand forecasting using DAN2, FTDNN, and KNN models: A case study of the city of Tehran, Iran. *Urban Water Journal*, 2017. 14(6): p. 655-659.
- [9] Mohamed, M.M. and A.A. Al-Mualla, Water demand forecasting in Umm Al-Quwain using the constant rate model. *Desalination*, 2010. 259(1-3): p. 161-168.
- [10] Ghiassi, M., D.K. Zimbra, and H. Saidane, Urban water demand forecasting with a dynamic artificial neural network model. *Journal of Water Resources Planning and Management*, 2008. 134(2): p. 138-146.
- [11] Altunkaynak, A., M. Özger, and M. Çakmakci, Water consumption prediction of Istanbul city by using fuzzy logic approach. *Water Resources Management*, 2005. 19(5): p. 641-654.
- [12] Salmeron, J.L., W. Froelich, and E.I. Papageorgiou, Application of fuzzy cognitive maps to the forecasting of daily water demand. in *Proceedings of*. 2015.
- [13] Sardinha-Lourenço, A., et al., Increased performance in the short-term water demand forecasting through the use of a parallel adaptive weighting strategy. *Journal of Hydrology*, 2018. 558: p. 392-404.
- [14] Almeida, G., et al., Estimating the potential water reuse based on fuzzy reasoning. *Journal of environmental management*, 2013. 128: p. 883-892.
- [15] Azadeh, A., N. Neshat, and H. Hamidipour, Hybrid fuzzy regression-artificial neural network for improvement of short-term water consumption estimation and forecasting in uncertain and complex environments: case of a large metropolitan city. *Journal of Water Resources Planning and Management*, 2012. 138(1): p. 71-75.
- [16] Liu, H., T. Deng, and H. Zhang, Research on forecasting method of urban water demand based on fuzzy theory. in *2009 Sixth International Conference on Fuzzy Systems and Knowledge Discovery*. 2009. IEEE.
- [17] Tabesh, M. and M. Dini, Fuzzy and neuro-fuzzy models for short-term water demand forecasting in Tehran. 2009.
- [18] Mohammad-Azari, S., O. Bozorg-Haddad, and H.A. Loáiciga, State-of-art of genetic programming applications in water-resources systems analysis. *Environmental Monitoring and Assessment*, 2020. 192(2): p. 73.
- [19] Mouatadid, S. and J. Adamowski, Using extreme learning machines for short-term urban water demand forecasting. *Urban water journal*, 2017. 14(6): p. 630-638.
- [20] Xu, Y., et al., A novel dual-scale deep belief network method for daily urban water demand forecasting. *Energies*, 2018. 11(5): p. 1068.
- [21] Mu, L., et al., Hourly and daily urban water demand predictions using a long short-term memory based model. *Journal of Water Resources Planning and Management*, 2020. 146(9): p. 05020017.
- [22] Muhammad, A.U., X. Li, and J. Feng, Artificial Intelligence Approaches for Urban Water Demand Forecasting: A Review. in *International Conference on Machine Learning and Intelligent Communications*. 2019. Springer.
- [23] Amasyali, K. and N.M. El-Gohary, A review of data-driven building energy consumption prediction studies. *Renewable and Sustainable Energy Reviews*, 2018. 81: p. 1192-1205.
- [24] Breiman, L., Random forests. *Machine learning*, 2001. 45(1): p. 5-32.
- [25] Schapire, R.E., Explaining adaboost, in *Empirical inference*. 2013, Springer. p. 37-52.
- [26] Drucker, H. Improving regressors using boosting techniques. in *ICML*. 1997.
- [27] Ke, G., et al. Lightgbm: A highly efficient gradient boosting decision tree. in *Advances in neural information processing systems*. 2017.
- [28] Chen, T. and C. Guestrin. Xgboost: A scalable tree boosting system. in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 2016.
- [29] Freedman, D., R. Pisani, and R. Purves, *Statistics Edition 2007*, 4th. 2007.
- [30] McKinney, W. Data structures for statistical computing in python. in *Proceedings of the 9th Python in Science Conference*. 2010. Austin, TX.
- [31] Pedregosa, F., et al., Scikit-learn: Machine learning in Python. *Journal of machine learning research*, 2011. 12(Oct): p. 2825-2830.