Applying Tree Ensemble to Detect Anomalies in Real-World Water Composition Dataset

Minh Nguyen(✉) and Doina Logofătu

Department of Computer Science and Engineering,
Frankfurt University of Applied Sciences, 60318 Frankfurt, Germany
mhuy@stud.fra-uas.de

Abstract. Drinking water is one of fundamental human needs. During delivery in distribution network, drinking water is susceptible to contaminants. Early recognition of changes in water quality is essential in the provision of clean and safe drinking water. For this purpose, Contamination warning system (CWS) composed of sensors, central database and event detection system (EDS) has been developed. Conventionally, EDS employs time series analysis and domain knowledge for automated detection. This paper proposes a general data driven approach to construct an automated online event detention system for drinking water. Various tree ensemble models are investigated in application to real-world water quality data. In particular, gradient boosting methods are shown to overcome challenges in time series data imbalanced class and collinearity and yield satisfied predictive performance.

Keywords: Tree ensemble · Gradient boosting · Random forest
Anomaly detection · Time series · Water quality · Contamination
Class imbalance

1 Introduction

Water is vital for all known forms of life and for the growth and development of human civilization. The provision of clean and safe drinking water is a necessity and a challenge for countries around the world. Drinking water supply and its distribution network are highly sensible to any kinds of contaminations. Water supply companies need to frequently monitor water and environmental data. In the modern days, these data are collected by highly sensible sensors and analyzed to detect any kinds of anomalies. Early recognition of changes in water quality enables water supply companies to counteract in time. For this purpose, contamination warning system (CWS) has been developed. The system composes of multiple sensor station, a Supervisory Control and Data Acquisition (SCADA) central database and an event detection system (EDS) [1].

In the recent years, machine learning methods have shown remarkable performance in different time series prediction and analysis problem [2] and have driven

© Springer Nature Switzerland AG 2018
H. Yin et al. (Eds.): IDEAL 2018, LNCS 11314, pp. 429–438, 2018.
https://doi.org/10.1007/978-3-030-03493-1_45

the operation of many automated system. This paper proposes an automated event detection system that detects the contamination event from variation in real-world water composition data collected from surrogate sensors and operational data. The system is driven by various tree ensemble methods that are tolerant to collinearity and class imbalance. The empirical performance of these ensemble models is evaluated and presented.

2 Related Work

The use of surrogate sensor to collect water quality parameter for contamination detection has been reported in [3,4]. Parameters like Turbidity, pH and Chlorine were found to be effective indicators of water contamination. Motivated by these findings, a range of anomalous contamination event detection techniques has been developed over the year. The work in [3] uses the limit of detection of 3 standard deviation interval. Time increment, linear filters (generally known as autoregressive (AR) moving average) and multivariate nearest neighbor (MV-NN) which are studied in [5], detect anomalies by taking the past and current measurements into account. Real-time event adaptive detection, identification and warning (READiw) methodology proposed in [6] recognize anomalies from the adaptively back-tracked background sensors data in a moving time window. Probabilistic approach [7] applies AR model to estimate future water quality parameters whose residuals are assigned with probabilities. The anomalous probabilities of residual are searched using Dempster-Schafer fusion. Similar approach proposed in [8] utilizes Artificial Neural Network (ANN) for parameters estimation and Bayesian sequential analysis for probabilities update. Supervised machine learning techniques like Logistic Regression, Linear Discriminant Analysis, ANN, Support Vector Machine (SVM) are applied in [9] to drinking water quality event detection. An exhaustive list of different machine learning and big data techniques suggested in various systems of water contamination detection is compared in the survey [10]. Some of the studied methods are Fast fuzzy C-mean clustering, GIS with Ant Colony algorithm, Radial Basis Network Function, ANN, Least square SVM. On a general note, tree boosting technique has been shown to give state-of-the-art results on many standard classification benchmarks [11] and has been implemented in real-world production pipeline [12]. Inspired by these studies, we think tree boosting is applicable to the present classification problem of drinking water contamination.

3 Problem Description

The dataset is provided by the Thüringer Fernwasserversorgung (TFW) and is made publicly available via GECCO industrial challenge 2018 [13]. The objective is to develop an online system that monitors water quality and operational status to detect remarkable changes in water quality, or contamination events. The system shall have reasonable response time to online input water data, output whether a remarkable change occur at a single point of time.

For the monitoring of the water quality, measurements are collected at significant points throughout the whole water distribution system, in particular at the outflow of the waterworks and the in- and outflow of the water towers. A part of the water is bypassed through a sensor system located at different stations near the outflow of a waterworks, where the most important water quality indicators and operational data are measured. For detailed description of the sensors data, refer to Table 1.

The target variable indicates the contamination caused by regular checks and replacement of sensors. Fig. 1 shows a snapshot of time series with such events marked in red. As a part of EDS Evaluation Process, simulated theoretical contamination events are also added into the data [1,13], as shown in Fig. 2. Both kind of contamination events are marked as variable EVENT.

Table 1. Variables in dataset and their descriptions

Variable	Description
Time	Time of measurement, given in following format: yyyy-mm-dd HH:MM:SS
Tp	The temperature of the water, given in $^{\circ}C$
Cl	Amount of chlorine dioxide in the water, given in mg/L (MS1)
pH	pH value of the water
Redox	Redox potential, given in mV
Leit	Electric conductivity of the water, given in μS/cm
Trueb	Turbidity of the water, given in NTU
Cl_2	Amount of chlorine dioxide in the water, given in mg/L (MS2)
Fm	Flow rate at water line 1, given in m^3/h
Fm_2	Flow rate at water line 2, given in m^3/h
EVENT	Marker if this entry should be considered as a remarkable change resp. event, given in boolean

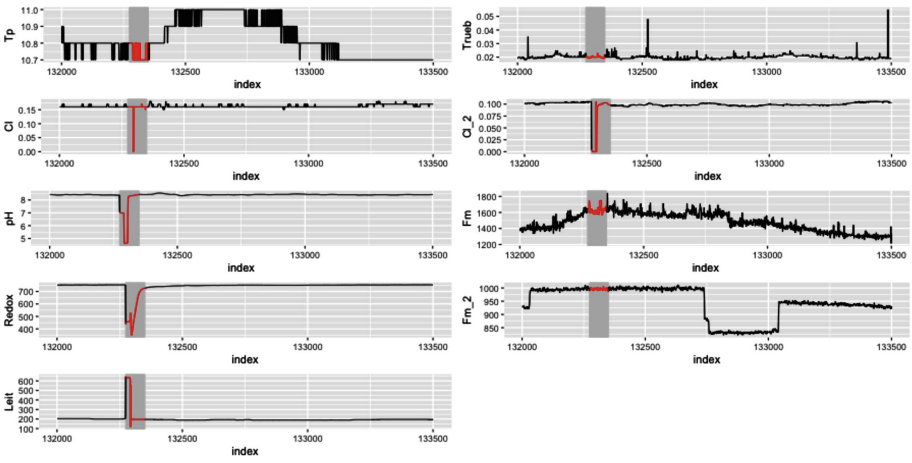

Fig. 1. Time series of sensors data of one day with original event marked in red [13].

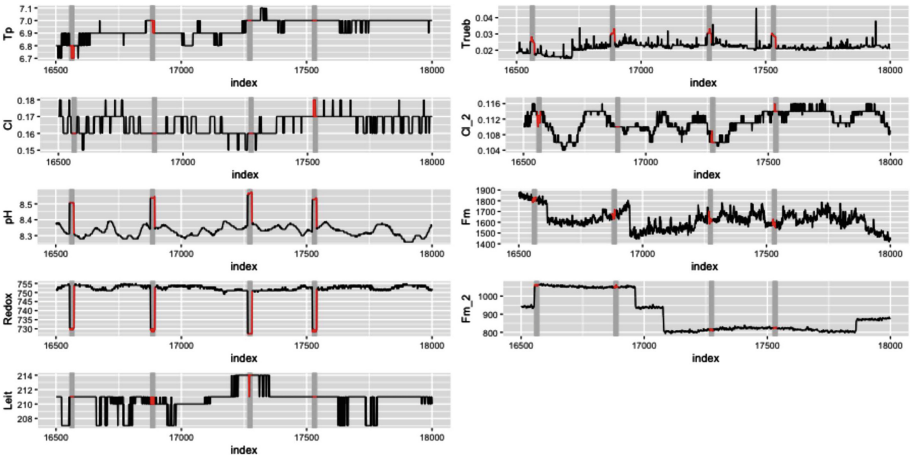

Fig. 2. Time series of sensors data of one day with simulated event marked in red [13].

4 Data Analysis

The problem is a supervised binary classification problem with 10 predictors and 1 target variable. The training data ranges from 03/08/2016 till 08/11/2016, equivalent to 3 months or 97 days. The data is recorded every 1 min, amounts up to 139,566 observations. Temporal trend might have great potential in data analysis and prediction, but the time span of 3 months is not sufficient to extract temporal trend without biases.

The number of missing values is the same for all predictors and is negligible (1045 samples or 0.75%). One can apply one of the imputation techniques studied in [14] to avoid performance degradation. All observations with missing values are labeled False and target variable contains no missing value. Considering these points, missing values are handled by forward fill, i.e. to fill missing values with its preceding values. The ratio of target variable is highly imbalanced with 98.76% False and 1.23% True (1726 samples). High class imbalance can lead to biased evaluation score like accuracy.

The target variable has medium correlated with `Redox` measurement and small correlation with `pH` and `Cl_2`, as shown in Fig. 3. On the other hand, predictors `Tp, pH, Redox, Fm` have medium to strong pairwise correlations. Collinearity caused by correlated predictors is problematic to many predictive models and deserves extra attention in modeling.

5 Proposed Approach

Our proposed predictive system contains 2 phases: Offline Training phase and Online Prediction phase. In the training phase, several predictive models are fitted and evaluated with cross validation. The model with the best cross-validated

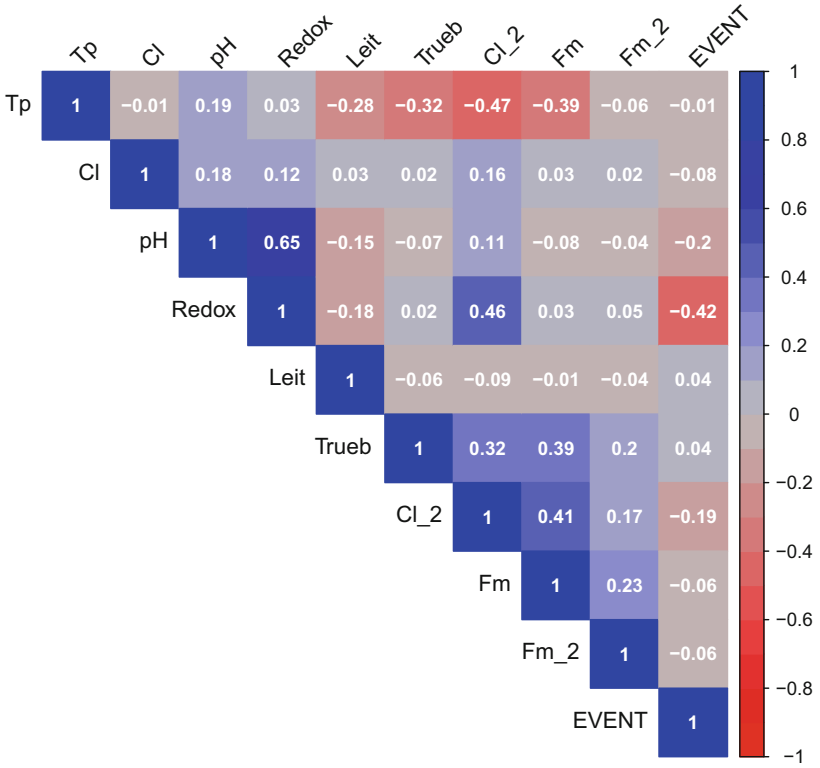

Fig. 3. Pairwise Correlation plot.

score is used in the prediction phase to generate prediction from unseen online data.

The training phase contains the pipeline of Data Preparation, Parameter Tuning and Model Evaluation that trains different tree-based models. Data preparation involves data type conversion, handling missing values and splitting data into train and test set with ratio 8:2. Cross validation is realized with stratified sampling. Parameter Tuning utilizes exhaustive search over the parameter grid, and each set of selected parameters is evaluated with 5-fold cross validation. The metric used to evaluate the learning models is F1-score. F1 is defined as the harmonic mean of recall and precision, thus is consistent in dataset with imbalanced class.

$$F_1 = \frac{2}{\frac{1}{\text{recall}} + \frac{1}{\text{precision}}} \tag{1}$$

We investigate various tree ensemble models as the core learning algorithm. The models are evaluated using cross-validation and their results are presented in Sect. 6. Due to the short time span of the training data, time series analysis

is not effective and can cause temporal bias toward the training data. Thus, to avoid adding unnecessary complexity to the online detector, we will not consider time series analysis techniques for modeling such as Moving average, Fourier Transform, ARIMA, etc.

5.1 Tree Ensemble Model

Tree-based model is essentially ensemble classifier with decision tree as base classifier. The main idea of ensemble learning is to combine many diverse classifiers to obtain a new classifier that outperforms each one of them. Decision tree is preferred as base classifier in ensemble because it is weak learner, where small changes in input data results in large changes in prediction output, thus brings diversity to ensemble and consequently reduce the variance.

There are several techniques to construct ensemble, for instance Bagging [15] and Boosting [16]. Various preprocessing technique that handle imbalanced class are embedded in state of the art ensembles [17]. In addition, ensembles can model complex relationship between predictors and are tolerant to collinearity [18]. Our experiments investigate the performance of the following tree ensembles: Random Forest [19] (RF), Regularized Random Forest [20] (RRF), Extreme Gradient Boosting [21] (xgbTree), Extreme Gradient Boosting with Dropout [22] (xgbDART). Random Forest introduces random sampling of predictors to bagged tree to reduce variance. Extreme Gradient Boosting is a scalable implementation of Gradient Boosting [23].

5.2 Gradient Boosting

Decision tree is a simple classifier, contains a set of rules that decide how to split the feature space into disjoint regions $R_j, \quad j = 1, \ 2, \ \ldots N$. Given constant y_i assigned to each region, the prediction rule is $x \in R_j \Rightarrow f(x) = \gamma_j$. According to [24], tree can be formally defined as $T(x;\ \Theta) = \ \sum_{j=1}^{J} \gamma_j I(x \in R_j)$, with tree parameter $\Theta = \{R_j; \gamma_j\}_i^J$. The boosted tree model is the sum of trees $f_M(x) = \sum_{m=1}^{M} T(x;\ \Theta_m)$.

Algorithm 1 describes how a generic version of Gradient Boosting works [24], where the specific algorithm is realized by defining the loss function. For K-class classification problems, the multinomial deviance loss function is defined

$$L(y_i,\ p(x_i)) = -\sum_{k=1}^{K} I(y_i = G_k) \log p_k(x_i) \tag{2}$$

6 Experimental Results

Tree ensemble models `RF`, `RRF`, `xgbDART`, `xgbTree` are tuned and cross-validated in the training phase described above. Table 2 shows the results of cross-validation in term of average and standard deviation. Extreme Gradient Boosting

Algorithm 1. Pseudocode for Generic Gradient Tree Boosting algorithm

Input: Training set $\{(x_i, y_i)\}_{i=1}^{N}$, Loss function $L\left(y_i, f\left(x_i\right)\right)$, Number of iterations M
Output: Fitted model $\hat{f}\left(x\right)$

1. Initialize model

$$f_0(x) = arg \sum_{i=1}^{N} L\left(y_i, \gamma\right)$$

2. For $m = 1$ to M:
   (a) For $i = 1, 2, \ldots, N$ compute pseudo residuals

$$r_{\mathrm{im}} = -\left[\frac{\partial L\left(y_i, f\left(x_i\right)\right)}{\partial f\left(x_i\right)}\right]_{f=f_{m-1}}$$

   (b) Fit a regression tree to the target r_{im} giving terminal regions R_{jm}, $j = 1, 2, \ldots, N$
   (c) For $j = 1, 2, \ldots, J_m$, where J_m is the size of each tree, compute

$$\gamma_{\mathrm{jm}} = \arg \sum_{x_i \in R_{\mathrm{jm}}}^{N} L\left(y_i, f_{m-1}\left(x_i\right) + \gamma\right)$$

   (d) Update the model

$$f_{m(x)} = f_{m-1}(x) + \sum_{j=1}^{J_m} \gamma_{\mathrm{jm}} I\left(x \in R_{\mathrm{jm}}\right)$$

3. Output $\hat{f}\left(x\right) = f_M(x)$

and its variant yield the most optimal cross-validated score and model stability. Model `xgbDART` yields the best F1-score of 0.9238 and standard deviation of 0.06. Model `xgbTree` yields a slightly smaller F1-score of 0.9164 and standard deviation of 0.05.

The recall metric, indicating the true positive rate, i.e. the rate of detecting anomaly correctly, is more prioritized than the precision, because high recall means less anomaly is missed. In Table 3, `xgbDART` achieves highest recall of 0.8852 and standard deviation of 0.07. RF performs slightly better than `xgbTree`, 0.8694 compared to 0.8650. On the other hand, RF has greater variance than `xgbTree`, approximatedly 0.01 in standard deviation.

Regarding the training time, `xgbTree` is remarkably faster than `xgbDART` (approximately 14 times faster). `xgbTree` implements scalable distributed architecture and is also the fastest among all investigated model. Regularization in Extreme Gradient Boosting and Random Forest slow down the training time. The training time in the offline phase is independent with the execution of the online prediction system, and can be neglected from the model selection criteria.

Feature importance is a metric used in tree-based model to indicate the contribution of each feature to the improvement of the prediction. Random Forest

Table 2. Cross validation results of tree-based models

Model	Train time (s)	F1	F1 SD	Accuracy	Accuracy SD
xgbDART	**2616.05**	**0.9238**	**0.0656**	0.9284	0.0609
xgbTree	**186.14**	**0.9164**	**0.0536**	0.9240	0.0440
RF	659.18	0.8660	0.0444	0.8665	0.0480
RRF	14201.53	0.8245	0.0877	0.8393	0.0749

Table 3. Cross validation results of tree-based models (cont.)

Model	Train time (s)	AUC	AUC SD	Recall	Recall SD
xgbDART	2616.05	0.9279	0.0609	**0.8852**	**0.0721**
xgbTree	186.14	0.9233	0.0442	0.8650	0.0895
RF	659.18	0.8662	0.0473	**0.8694**	**0.1022**
RRF	14201.53	0.8384	0.0744	0.7758	0.1302

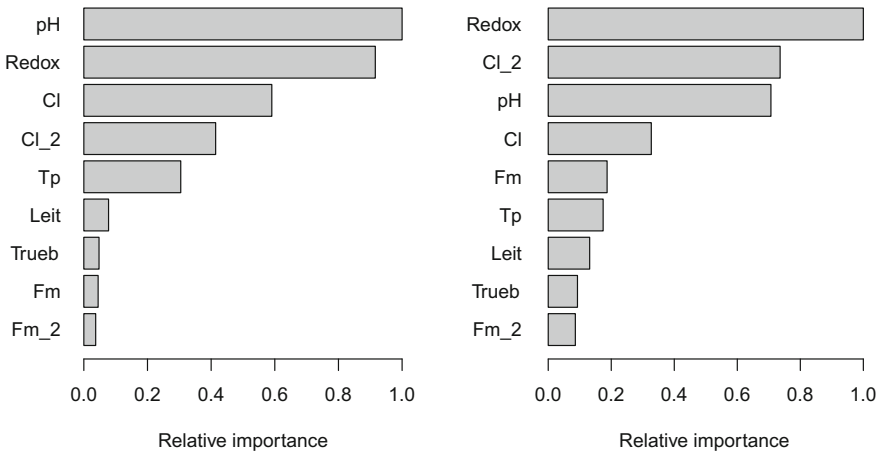

Fig. 4. Relative feature importance of Extreme Gradient Boosting (`xgbTree`) on the left and Random Forest (`RF`) on the right

computes feature importance from the decrease in Gini index at each split while Extreme Gradient Boosting feature importance measures the improvement in accuracy when spliting. Feature importance of the fitted xgbTree and RF model are illustrated in Fig. 4, relatively to the most important feature. Top three highest importances are similar for both models, namely pH of water, Redox potential and chlorine dioxide (`pH, Redox` and `Cl` for `xgbTree` and `Redox, Cl_2` and `pH` for `RF`).

7 Conclusion

The present approach shows that the minimal machine learning workflow consisting of data analysis, learning model selection and proper cross-validation pipeline could yield satisfactory prediction results. Tree ensembles are able to perform well in scenario with imbalanced class and collinearity in the dataset. Extreme Gradient Boosting excels in performance as well as training time. However, the fact that the dataset is limited in size and time range must be taken into consideration when interpreting the results. In case the training data is not exemplary, it is likely that the fitted model is biased toward the training data, thus make it challenging to investigate the generalization capability of the predictive system.

During our work, we found that techniques such as dimensional reduction, unsupervised learning and time series analysis exhibit potential on future improvement of the present tree ensemble approach. In addition, it worth further investigation on data augmentation and the method of adding simulated theoretical values into the present dataset. On top of that, the gist of ensemble methods can be applied to other learning models that was studied in related literature, which is referred to as meta ensemble or model stacking. Meta ensemble leverages the diversity of different base learner to attain better final results. Nevertheless, the more effective solution lies in data collection and data quality assurance. Given a larger dataset, the same workflow with time series analysis integrated could extract valuable temporal feature, decorrelate sensor values and consequently achieve significantly improved performance.

References

1. McKenna, S.A., Hart, D.B., Murray, R., Haxton, T.: Testing and evaluation of water quality event detection algorithms. In: Clark, R.M., Hakim, S., Ostfeld, A. (eds.) Handbook of Waterand Wastewater Systems Protection, pp. 369–396. Springer, New York (2011). https://doi.org/10.1007/978-1-4614-0189-6_19
2. Hamilton, J.D.: Time Series Analysis. Princeton University Press, Princeton (1994)
3. Byer, D., Carlson, K.H.: Real-time detection of intentional chemical contamination in the distribution system. J.- Am. Water Work. Assoc. **97**(7), 130–133 (2005)
4. Hall, J., Szabo, J.: WaterSentinel Online Water Quality Monitoring as an Indicator of Drinking Water Contamination. Environmental Protection Agency, Washington, DC, USA (2005)
5. Klise, K.A., McKenna, S.A.: Multivariate applications for detecting anomalous water quality. In: Water Distribution Systems Analysis Symposium 2006, Cincinnati, Ohio, United States, pp. 1–11. American Society of Civil Engineers, March 2008
6. Jeffrey Yang, Y., Haught, R.C., Goodrich, J.A.: Real-time contaminant detection and classification in a drinking water pipe using conventional water quality sensors: techniques and experimental results. J. Environ. Manag. **90**(8), 2494–2506 (2009)
7. Hou, D., He, H., Huang, P., Zhang, G., Loaiciga, H.: Detection of water-quality contamination events based on multi-sensor fusion using an extented Dempster-Shafer method. Meas. Sci. Technol. **24**(5), 055801 (2013)

8. Perelman, L., Arad, J., Housh, M., Ostfeld, A.: Event detection in water distribution systems from multivariate water quality time series. Environ. Sci. Technol. **46**(15), 8212–8219 (2012)
9. Muharemi, F., Logofătu, D., Andersson, C., Leon, F.: Approaches to building a detection model for water quality: a case study. In: Sieminski, A., Kozierkiewicz, A., Nunez, M., Ha, Q.T. (eds.) Modern Approaches for Intelligent Information and Database Systems. SCI, vol. 769, pp. 173–183. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-76081-0_15
10. Kang, G., Gao, J.Z., Xie, G.: Data-driven water quality analysis and prediction: a survey. In: 2017 IEEE Third International Conference on Big Data Computing Service and Applications (BigDataService), pp. 224–232, April 2017
11. Li, P.: Robust logitboost and adaptive base class (ABC) logitboost. In: Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence, UAI 2010, Arlington, Virginia, United States, pp. 302–311. AUAI Press (2010)
12. He, X., et al.: Practical lessons from predicting clicks on ads at Facebook. In: Proceedings of the Eighth International Workshop on Data Mining for Online Advertising, ADKDD 2014, New York, NY, USA, pp. 5:1–5:9. ACM (2014)
13. Rehbach, F., Moritz, S., Chandrasekaran, S., Rebolledo, M., Friese, M., Bartz-Beielstein, T.: GECCO 2018 Industrial Challenge, Monitoring of drinking-water quality (2018)
14. Muharemi, F., Logofătu, D., Leon, F.: Review on general techniques and packages for data imputation in R on a real world dataset. In: Nguyen, N.T., Pimenidis, E., Khan, Z., Trawiński, B. (eds.) ICCCI 2018. LNCS (LNAI), vol. 11056, pp. 386–395. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-98446-9_36
15. Breiman, L.: Bagging predictors. Mach. Learn. **24**(2), 123–140 (1996)
16. Schapire, R.E., Freund, Y., Bartlett, P., Lee, W.S.: Boosting the margin: a new explanation for the effectiveness of voting methods. Ann. Stat. **26**(5), 1651–1686 (1998)
17. Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., Herrera, F.: A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. IEEE Trans. Syst., Man, Cybern. Part C (Appl. Rev.) **42**(4), 463–484 (2012)
18. Dormann, C.F., et al.: Collinearity: a review of methods to deal with it and a simulation study evaluating their performance. Ecography **36**(1), 27–46 (2013)
19. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
20. Deng, H., Runger, G.: Gene selection with guided regularized random forest. Pattern Recognit. **46**(12), 3483–3489 (2013)
21. Chen, T., Guestrin, C.: XGBoost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2016, New York, NY, USA, pp. 785–794. ACM (2016)
22. Rashmi, K., Gilad-Bachrach, R.: Dart: dropouts meet multiple additive regression trees. In: International Conference on Artificial Intelligence and Statistics, pp. 489–497 (2015)
23. Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Ann. Stat. **29**(5), 1189–1232 (2001)
24. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. SSS. Springer, New York (2009). https://doi.org/10.1007/978-0-387-84858-7