Online Anomaly Detection for Drinking Water Quality Using a Multi-objective Machine Learning Approach

Victor Henrique Alves Ribeiro
Industrial and Systems Engineering Graduate Program
Pontifical Catholic University of Paraná (PUCPR)
Curitiba, Paraná, Brazil
victor.henrique@pucpr.edu.br

Gilberto Reynoso-Meza
Industrial and Systems Engineering Graduate Program
Pontifical Catholic University of Paraná (PUCPR)
Curitiba, Paraná, Brazil
g.reynosomeza@pucpr.br

ABSTRACT

This document proposes the use of multi-objective machine learning in order to solve the problem of online anomaly detection for drinking water quality. Such problem consists of an imbalanced data set where events, the minority class, must be correctly detected based on a time series denoting water quality data and operative data. In order to develop two different robust systems, signal processing and feature engineering are used to prepare the data, while evolutionary multi-objective optimization is used for feature selection and ensemble generation. The proposed systems are tested with hold-out validation during optimization, and are expected to generalize well the predictions for future testing data.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning**; *Ensemble methods*; *Continuous space search*;

KEYWORDS

Machine learning, evolutionary computation, time series, anomaly detection.

ACM Reference format:
Victor Henrique Alves Ribeiro and Gilberto Reynoso-Meza. 2018. Online Anomaly Detection for Drinking Water Quality Using a Multi-objective Machine Learning Approach. In *Proceedings of Genetic and Evolutionary Computation Conference Companion, Kyoto, Japan, July 15–19, 2018 (GECCO '18 Companion),* 2 pages.
https://doi.org/10.1145/3205651.3208202

1 INTRODUCTION

This document proposes two solutions for the GECCO 2018 Challenge "Internet of Things: Online Anomaly Detection for Drinking Water Quality" [7]. The problem is composed of a time series with six water quality indicators and three operational data attributes, where events must be accurately detected. In order to solve such problem, the use of feature engineering, machine learning and evolutionary computing techniques are used. Specifically, the proposed solutions makes use of two different approaches for multi-objective machine learning [4]: multi-objective feature selection, for support vector machines (SVM) [1, 2]; and multi-objective ensemble generation [3], for decision trees (DT) [6].

The remainder of this document is presented as follows: section 2 presents the proposed methodologies; and section 3 presents the conclusions.

2 METHODOLOGY

2.1 Feature Engineering

The first step focuses on adjusting the data for classification. Since the problem deals with time series, signal processing and statistics techniques are used in order to create the following new features:

- *Imputing*: imputes average value on missing points;
- *Detrending*: removes time related distortions;
- *Simple moving average*: smooths the data;
- *Moving standard deviation*: creates dispersion indicators;
- *Moving minimum and maximum*;
- *Squared features*: maps the data to a new nonlinear space;
- *Short-time Fourier transform*: detects the signal's frequencies.

2.2 Multi-objective Machine Learning

Two different approaches for multi-objective machine learning are proposed. The first one is responsible for creating an optimized classifier using feature selection. It is composed of the following multi-objective problem (MOP):

$$\min_{\boldsymbol{x_g}} \boldsymbol{J_g(x_g)} = [-J_1(\boldsymbol{x_g}), -J_2(\boldsymbol{x_g}), -J_3(\boldsymbol{x_g}), -J_4(\boldsymbol{x_g}), J_5(\boldsymbol{x_g})] \quad (1)$$

subject to:

$$x_{gi} \in \{0, 1\}, i = [1, \ldots, n] \quad (2)$$

where the objectives are: global accuracy ($J_1(\boldsymbol{x_g})$ [dimensionless]); true positive rate ($J_2(\boldsymbol{x_g})$ [dimensionless]); true negative rate ($J_3(\boldsymbol{x_g})$ [dimensionless]); F1 score ($J_4(\boldsymbol{x_g})$ [dimensionless]), or the harmonic mean of sensitivity and precision; and classifier's complexity ($J_5(\boldsymbol{x_g})$), defined as the number of used features. The decision variables are: (x_{gi}) selection of each of the n features.

The set of non-dominated solutions is created by using the second version of the spherical pruning multi-objective differential evolution (spMODE-II) [8], where 200 generations are performed on a population of 50 individuals, using a crossover ratio of 0.5 and a scaling factor of 0.5. The final classifier is achieved by selecting the model with best physical programming ranking [5] according to Table 1, where F1 score and complexity are analyzed in order to achieve a model that generalizes well.

GECCO '18 Companion, July 15–19, 2018, Kyoto, Japan

ACM ISBN 978-1-4503-5764-7/18/07.
https://doi.org/10.1145/3205651.3208202

Table 1: Preference matrix for model selection. Five scaled preference ranges have been defined: highly desirable (HD), desirable (D), tolerable (T) undesirable (U) and highly undesirable (HU).

Preference Matrix											
	←	HD	→←	D	→←	T	→←	U	→←	HU	→
Objective	S_i^0		S_i^1		S_i^2		S_i^3		S_i^4		S_i^5
F1 score	0.00		0.10		0.20		0.40		0.60		1.00
Complexity	0.00		0.10		0.20		0.40		0.60		1.00

The second approach is responsible for generating an optimized ensemble of classifiers, and is composed of two MOPs. The first one is defined as the previous, which results on a set of non-dominated classifiers. After this, the second MOP is responsible for selecting such classifiers to form an optimized ensemble, and is stated as follows:

$$\min_{\boldsymbol{x_s}} \boldsymbol{J_s}(\boldsymbol{x_s}) = [-J_1(\boldsymbol{x_s}), -J_2(\boldsymbol{x_s}), -J_3(\boldsymbol{x_s}), -J_4(\boldsymbol{x_s}), J_6(\boldsymbol{x_s})] \quad (3)$$

subject to:

$$x_{si} \in \{0, 1\}, i = [1, \ldots, m] \quad (4)$$

where the new objective is the ensemble's complexity ($J_6(\boldsymbol{x_s})$), defined as the sum of the features from the ensemble members. The decision variables are: (x_{si}) selection of each of the m classifiers returned from the first MOP.

The second MOP is also optimized with spMODE-II, using the same optimization parameters as before. This results on a set of non-dominated ensembles, where the most preferable solution is selected using physical programming, also according to Table 1.

2.3 Online Event Detector

The final event detectors, performed by the *detect* function, are created according to the flowchart in Figure 1. First, the trained and optimized model is loaded; then, the dataset row is inserted in a buffer, composed of 1000 rows. In the feature transformation step, the tasks described in section 2.1 are performed with the buffer data. The resulting features are then fed to the classifier, which returns the event prediction.

The *destruct* function, on the other hand, performs the cleaning of the created buffer and the loaded model. Finally, the *getOutline* function returns the submission title and authors names. For the proposed event detectors, both developed in R, the following packages are necessary: rpart (4.1-13), e1071 (1.6-8), and pracma (2.1.4).

3 CONCLUSION

The present document proposes two methodologies for creating an online drinking-water event monitoring system. To do so, two different predictors are created using multi-objective machine learning during the training step. Both systems are optimized based on the F1 score, and are expected to achieve high results on the competition. If desirable results are achieved, the developed systems can be used as base for an online drinking-water event detector on a real-world system.

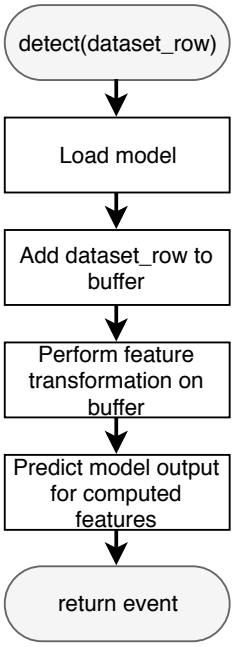

Figure 1: The proposed *detect* function for online drinking-water quality monitoring.

ACKNOWLEDGMENTS

This work is supported by the Brazilian Federal Agency for Support and Evaluation of Graduate Education (CAPES) through the grant PROSUC/159063/2017-0. This work is also under the research initiative *Multi-objective optimisation design (MOOD) procedures for engineering systems: Industrial applications, unmanned aerial systems and mechatronic devices*, supported by the National Council of Scientific and Technological Development of Brazil (CNPq) through the grant PQ-2/304066/2016-8.

REFERENCES

[1] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. 1992. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*. ACM, 144–152.
[2] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning* 20, 3 (1995), 273–297.
[3] Shenkai Gu, Ran Cheng, and Yaochu Jin. 2015. Multi-objective ensemble generation. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 5, 5 (2015), 234–245.
[4] Yaochu Jin and Bernhard Sendhoff. 2008. Pareto-based multiobjective machine learning: An overview and case studies. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 38, 3 (2008), 397–415.
[5] Achille Messac. 1996. Physical programming: effective optimization for computational design. *AIAA journal* 34, 1 (1996), 149–158.
[6] Brijain R Patel and Kushik K Rana. 2014. A Survey on Decision Tree Algorithm For Classification. *Ijedr* 2, 1 (2014), 1–5.
[7] Frederik Rehbach, Steffen Moritz, Sowmya Chandrasekaran, Margarita Rebolledo, Martina Friese, and Thomas Bartz-Beielstein. 2018. GECCO 2018 Industrial Challenge: Monitoring of drinking-water quality. (2018). http://www.spotseven.de/wp-content/uploads/2018/03/rulesGeccoIc2018.pdf
[8] Gilberto Reynoso-Meza, Javier Sanchis, Xavier Blasco, and Sergio García-Nieto. 2014. Multiobjective evolutionary algorithms for multivariable PI controller tuning. *Applied Soft Computing* 24 (2014), 341 – 362.