Manually or Autonomously Operated Drones: Impact on Sensor Data towards Machine Learning

1st Rialda Spahic
Department of Engineering Cybernetics
Norwegian University of Science and Technology
Trondheim, Norway
rialda.spahic@ntnu.no

2nd Mary Ann Lundteigen
Department of Engineering Cybernetics
Norwegian University of Science and Technology
Trondheim, Norway
mary.a.lundteigen@ntnu.no

***Abstract*—The growing need for autonomous systems in offshore industries has contributed to the increased use of machine learning methods. These systems promise to improve safety in operations. However, the methods as enablers of autonomy are susceptible to various failures while interpreting data and making decisions. Several studies have highlighted the lack of research on the reliability and resilience of autonomous systems powered by these standard methods. Recent research provides sets of data interpretation methods. Despite the popularity of machine learning, there is a significant drop in knowledge when these methods result in failures. These failures further support autonomous systems in making wrong decisions. For autonomous systems, resilience and safety management should be an integrated functionality for recovery from risky situations and reporting of incidents. This research proposes an overview of machine learning methods for interpreting sensor data captured by drones operated manually and autonomously. We apply Isolation Forest for anomaly detection analysis and evaluate the Decision tree, Random forest, kNN, Logistic Regression, SVM, and, Naive Bayes for classification analysis. The methods are chosen based on their adequacy and comparative research prevalence. Comparison between the two drone operation modes contributes to understanding the reliability level for autonomously collected data. This research's results provide an evaluation of machine learning methods' performance across sensor data.**

***Index Terms*—autonomous systems, sensor data analysis, machine learning, classification, anomaly detection**

I. Introduction

Autonomous systems (AS) have shown potential in enhancing safety in the industry by replacing human activities during dangerous operations in remote environments. These systems can perform tasks that require little to no human intervention by actively interpreting the real-time collected data. Timely decisions based on previously learned knowledge come from historical insights and domain expert inputs. There is a considerable number of machine learning (ML) methods that enable autonomy. However, these methods need to be reliable and trusted within safety-critical circumstances. The potential of ML depends on the data that the AS collects. In particular, sensor data can be overwhelming for the methods to answer with desired results. This data is most often varying from sound, video, image, pressure, temperature, and gas sensor measurements. The environment can potentially overwhelm the collected data with extensive noise that impacts the final decisions and results that AS provide.

During operations and asset surveillance, researchers are often interested in occurrences in the environment that are distinct from expected operating times, such as the presence of material degradation, misplaced objects, or biological growth. It is possible that during ML analysis, considerable amounts of data would lack distinct samples that are interesting to research. Therefore, methods can discard the data outside of the ordinary as noise. It is important to curate the data to avoid disregarding vital information hidden in the noise. Over the last decade, the industry interest in employing AS to perform tasks and reduce human efforts has continuously increased.

This paper compares ML performance on the sensor data collected by a manual and an autonomously operated underwater drone. Data collected by the same drone under different operating modes can widen our comprehension of autonomy dependability. We analyze the data through anomaly detection and classification methods. Anomaly detection identifies abnormalities in the data, contributing to fault prevention and predictive maintenance [1]. Classification methods, known as classifiers, are learning tasks that predict the data category of given data points. By applying these methods to the manually and autonomously collected sensor data, we build machine learning models that provide us with an insight into the reliability of the methods that enable autonomy.

II. Related Work

There is a considerable amount of research on finding the best methods to evaluating sensor data. Related work of this paper presents semi-organized comparative research. We single out the studies within the context of autonomous systems and highlight the motivation of using ML methods. In the following paragraphs, we discuss the sensor data analysis challenges through applied anomaly detection and predictive capabilities of classification methods in AS.

A. Anomaly Detection in Sensor Data

As earlier mentioned, environments in which drones operate can be noisy and disruptive. Anomaly or outlier detection is an important research area that contributes to fault prevention and predictive maintenance [1]. Erhan et al. [1] review anomaly detection methods employed in sensor systems. Authors highlight the data volumes, network efficiencies, information fu-

sion, and biases as some of the anomaly detection challenges. Due to the ample employment of sensors in smart devices such as the Internet of Things, Erhan et al. [1] argue that the sensor systems have become dominant generators of data. The authors identify different anomaly detection methods. Their research contributes to the study of sensor systems constraints and their impact on machine learning and anomaly detection. Authors also classify anomalies based on their source. These are typically sensor recordings that are distinct from expected behavior. Erhan et al. [1] point out that real-world data is necessary to validate the effectiveness of anomaly detection methods. However, anomalies occur unexpectedly and can be scarce in real-world data. Therefore, it can be challenging to generate them artificially [1].

Anwar et al. [2] propose a novel ML framework using feature extraction and SVM with varying kernels. The motivation behind their research is to eliminate disruptive sounds such as birds, airplanes, or thunderstorms as anomalies. This elimination would provide the detection of nearby amateur drones more accurate. Authors approach the problem by gathering real-time acoustic data and classifying the noise with Mel frequency cepstral coefficients, Linear predictive cepstral coefficients, and SVM. SVM has proved to be an efficient method for classifying noisy environments using small batches of data. In this research, ML promised a cost-effective and accurate tool with minimized chances of misclassification between classes [2].

B. Classification Methods in Sensor Data

Increased interest in autonomous systems has led to an increase in the use of machine learning methods. Choi et al. [3] explore the application of traditional ML methods employed in Unmanned Aerial Vehicles (UAVs) for autonomous operations. The authors explain that the collected data can show the method's performance more realistically when the testing environment is heterogeneous, consisting of various operational circumstances. They also advise testing the models in smaller batches of non-ideal settings to track AUVs' performance under disturbances.

Moustafa et al. [4] propose an autonomous Intrusion Detection Scheme (IDS) for real-time complex attack scenarios from drone networks. They use the predictive capabilities of ML for autonomous detection of malicious events in drone networks. Their research compares the following methods to classify cyber-attacks in drone networks: Decision tree, k-Nearest Neighbor (kNN), Naive Bayes, Support Vector Machine (SVM), and Deep learning Multi-layer Perceptron. The authors have synthetically created three different attack scenarios for testing vulnerabilities and recognizing attacks on time. Moustafa et al. [4] depicted a concept of targeted awareness towards different settings and involved detecting false alarm samples. In this research, the Decision tree has proven to be the best classifier, followed by multi-layer perceptron and kNN. The least performing classifier was Naive Bayes. The authors mention the opportunity to extend their research for more complex networks, simulating more sophisticated attacks. However, there is a usual lack of justification of the method failure in the context of their model.

The sensor data collected by the AS can be challenging to evaluate. Consequently, the employment of machine learning has a remarkable impact on the performance and efficiency of the AS [5]. The authors highlight that among the benefits of machine learning, there are three main issues: security, certification, and cost. To establish strong evidence behind these methods, De Dominicis et al. [5] suggest having a quantitative assessment about the system performance after introducing ML. The advice is to carry out a benchmarking analysis comparing 'novel methods' with the traditional solutions. The comparison should encompass prediction capabilities, robustness, integrity, and reliability.

III. Data and Methods

A. Research Purpose and Expectations

This research analyzes the differences between the data collected by a manual and autonomous drone operation by applying anomaly detection and classification methods. The expected result is that the differences between manual and autonomous operations are minimal due to the same sensors and pre-programmed mission plans. Therefore, the autonomously operated drone should provide the same level of reliability as the manually operated drone. The second expectation is that the non-linear classifiers that perform well on high dimensional and correlated data, such as Random Forest, will prevail over the linear methods (SVM, Logistic Regression, and Naive Bayes) due to the dataset's dimensionality, non-variability, and imbalance.

B. Research Data

The data used for this research, collected by Castellini et al. [6], is multivariate data containing sensor measurements of six data acquisition campaigns performed by underwater drones for water monitoring. The authors explored lakes and rivers of different locations in Spain and Italy for data collection. There are 11 monitored features in the dataset that results in 20,187 total samples. We have separated these features into general information of the area, water measurements, and drone measurements (see Table I). The available information of the site contains area location and time during drone exploration. Water measurements are specific sensor data that monitor water temperature, dissolved oxygen in the water, and electrical conductivity. Finally, drone-specific measurements monitor the drone's internal state, such as battery voltage, signals to propellers, and direction of the drone's bow.

Additionally, the data of each campaign is labeled by Castellini et al. [6] based on the drone operation status, drone curving, location in the water, and the status of the water flow. These four labels consist of set values. In the dataset, each value is represented by a number:

- Drive values: autonomous (2) , manual (1), unlabeled (0);

- Flow values: upstream (3), downstream (2), no water flow (1), unlabeled (0);
- Curves values: turning (2), no turning (1) ;
- Water values: out of water (2), in the water (1), unlabeled (0);

For this research, we merged the six data acquisition campaigns into one complete dataset. We then divided the complete dataset based on the 'Drive' label into manual and autonomous datasets. Division by drive allows exploring the data collected by the drone when it is manually operated and compare it to the data collected during autonomous operation. We select the Flow label as the *ground truth* (GT). GT is a measurement that classification methods predict. The GT Flow contains values for water monitoring that yield essential contextual information for the domain experts [7].

The sensor data represented in this paper is non-varied data with limited sensor inputs collected by simple drone missions. Additionally, the ground truth consists of a more significant number of samples within upstream and unknown water flow values than downstream and no water flow values. The imbalanced representation of the ground truth values in this data can restrict the performance of ML models.

TABLE I
FEATURE DESCRIPTION OF THE DATASET BY CASTELLINI ET AL. [6]

Feature	Category	Description
Latitude	General	Latitude of the area
Longitude	General	Longitude of the area
Altitude	General	Height above sea level
Date and time	General	5,6 h of runtime
EC	Water	Water electrical conductivity
Temperature	Water	Water temperature
DO	Water	Water dissolved oxygen
m0 current	Drone	Signal to propeller 0
m1 current	Drone	Signal to propeller 1
Heading	Drone	Compass direction in which the drone's bow is pointed
Voltage	Drone	Drone's battery voltage

C. Research Methods

Different methods allow the evaluation of model performance to justify the best methods across this dataset. We select the ML methods following their adequacy and comparative research prevalence. We identify the abnormalities in the data through the anomaly detection method, Isolation Forest. Following the anomaly detection, we analyze the data features by applying feature selection and ranking to understand the relationships among the dataset's features and their relationship with the ground truth. We apply data validation with the hold-out method to divide the data into training and testing datasets to prepare for classification. Finally, we compare different classification methods to analyze the predictive performance of the model. In this section, we justify the selected methods for analyzing the Castellini et al. [6] data.

1) Anomaly Detection with Isolation Forest: This model-based approach to anomaly detection isolates anomalies with low computational requirements. Isolation Forest works well in high-dimensional problems, and deals well with many irrelevant attributes [8]. Liu et al. [8] highlight the problem of anomalies being few, making them prone to isolation. Isolation Forest partitions instances repeatedly and recursively until they are isolated, producing shorter paths for anomalies [8]. The method does not use distance or density measures to detect outliers that eliminate computational costs making it a good fit for large and non-linear datasets.

2) Feature Selection and Ranking: Choosing a reduced feature set makes the model easier to interpret, removes inessential information, reduces the dataset's size, and lowers the possibility of overfitting [9]. Overfitting the model is an error that occurs when the training on the data results with high accuracy. However, the testing results with poor accuracy and is typically caused by high variance in the data.

a) Lasso Regularization: The Least Absolute Shrinkage and Selection Operator (Lasso) is a powerful regularization and feature selection method. This method applies the regularization or shrinking process by penalization of the coefficients of regression features [9]. The features regularized to zero are pruned from the model. The model, therefore, has the potential of reduced variance without a considerable increase of bias.

b) Filter Method with Pearson Correlation: Filter methods choose features through statistical tests and correlation by ranking them on their usefulness to the model [9].

3) Data Validation with Hold-Out Method: This validation method divides the data into two non-overlapping sets, training and testing. The hold-out is the testing set and can contain any percentage of the original dataset. The time for learning in the hold-out method is lesser than in comparable cross-validation methods [10]. The hold-out can eliminate the problem of overfitting, avoid uneven distribution, and introduce a clear division of data with stratification [10].

4) Classification Methods:

a) Decision Tree: The Decision tree classifier performs well on nonparametric, complex datasets. This method classifies samples into branch-like elements and constructs an inverted tree to make decisions [11]. However, the Decision tree can result in overfitting when working on small datasets or datasets with strongly correlated features.

b) Random Forest: A popular classifier, Random forest, constructs multiple decision trees with randomly selected subsets of features and training data. It is less sensitive to overfitting because of the considerable number of decision trees produced randomly. Random forest performs well on datasets with high dimensionality and highly correlated data, making this method a promising approach in heterogeneous research [12].

c) k-Nearest Neighbor (kNN): Classifier kNN forms around finding similarities in data. Therefore data quality is crucial to this method. KNN calculates the nearest points in data and nominates the sign of majority. Choosing the k number is often considered arbitrary; however, a larger value

of k number can reduce the effects of anomalous points [13]. Due to its sensitiveness to data quality, the method performs the best with smaller data batches with eliminated anomalies.

d) Logistic Regression: This classifier produces quick outputs that can be interpreted as probability and therefore used for ranking. Logistic Regression is not sensitive to overfitting; however, it underperforms on non-linear data.

e) Support Vector Machine (SVM): The SVM classifier has the advantage of performing well within high-dimensional space [14]. It fits a hyperplane that separates classes in data and positions every new data point within this hyperplane. However, this method is computationally expensive, slow on extensive data, and challenging to interpret.

f) Naive Bayes: Naive Bayes represents decision-making under uncertainty, or probabilistic approach to deduction [13]. This simple method is computationally fast, easy to interpret, and performs well with high-dimensional data. However, Naive Bayes will underperform if the data features are highly correlated or calculate the probability of zero if an unknown class in test data appears [15].

IV. Implementation, Results and Discussion

For this research, we implemented the models using the sklearn module[1] for Python with default hyperparameters. For every experiment, we analyzed the complete dataset containing 20,187 samples, manual and autonomous datasets. After removing the non-labeled or unknown Drive value, the manual set results in 7,586, and the autonomous set in 7,530 samples.

A. Anomaly Detection Results

TABLE II
Anomalies Analyzed with Isolation Forest

Analyzed data	Number of anomalies	% of anomalies
Complete dataset	2019	10.0014
Manual operation	759	10.0052
Autonomous operation	753	10.0000

Isolation Forest for anomaly detection outputs compelling results for the three datasets. According to the results (see Table II), the distribution of anomalies is comparable. A similar number of samples, approx.10%, of each dataset are identified as anomalies. Uniform distribution of detected anomalies promises a comparable data reliability level by manually or autonomously operated drones. Almost all of the identified anomalies have a GT value of 0. In the autonomous operation dataset, all anomaly samples belong to GT value 0. The anomaly detection method removed 100% of the samples with GT value 0, leaving only GT value 3 in the data. Similarly, in the manual operation dataset, only 0.0052% of the anomalies are not samples with GT value 0. This small number of non-zero anomalies are scattered around the manual dataset without showing a significant pattern. Autonomous and manual operation datasets display uniform distribution of anomalies, making them nearly equivalent in performance. However, GT value 0 represents unlabeled or unknown water flow which can be essential contextual information for the domain experts [7], particularly when analyzing sensor data performance. Hence, we retain the samples with GT value 0 in the dataset.

B. Feature Selection and Correlation Ranking Results

Pearson correlation results in uneven distribution of highly correlated features regarding the GT. For the complete dataset, there are seven highly correlated features: electrical current (ec), drive, water, altitude, longitude, latitude, and water temperature. For the manual dataset, only four features highly correlate to the GT: altitude, longitude, latitude, and water temperature. Lastly, for autonomous data, highly correlated features are voltage, altitude, longitude, latitude, m0 current and, m1 current. A high correlation between features can impact the classification, such as biased predictions due to a strong relationship of two or more features. The impact of these results is visible in the classification analysis.
Contrastingly, the Lasso Regularization method resulted in a more uniform set of selected features regarding their importance in the dataset: ec, dissolved oxygen, temperature, altitude, and heading. Furthermore, selected features are penalized to a significantly low coefficient, nearly pruned from the dataset. This method typically penalizes correlated features, potentially removing important information and creating unstable models [16]. With this information, we retain the entire set of features for the classification analysis.

C. Data Validation

We use the hold-out method for the data validation, where 60% of the data is split for training and 40% for testing the model. GT values' distribution is undeviating for train and test sets (Table III and Table IV). The autonomous operation dataset does not contain the GT values 2 and 1, and there is a heavy imbalance of the existing values, 3 and 0. The complete and manual operation datasets also contain differences between the GT values. However, a more uniform distribution of the GT values can contribute to the model's lessened sensitivity for the data's bias.

TABLE III
Distribution of the ground truth values on train set

Ground truth value	Complete	Manual	Autonomous
3	6880	2357	4316
2	514	532	0
1	383	381	0
0	4335	1281	202

[1]Scikit-learn Machine Learning in Python: https://scikit-learn.org/stable/

TABLE IV
DISTRIBUTION OF THE GROUND TRUTH VALUES ON TEST SET

Ground truth value	Complete	Manual	Autonomous
3	4499	1569	2862
2	341	323	0
1	259	261	0
0	2976	882	150

Fig. 1. Imbalanced data: Choosing Performance Metrics, adapted from [17]

D. Supervised Classification Results

Choosing a metric is likely the most critical phase in the project. Figure 1 illustrates the steps necessary for choosing an appropriate performance metric for imbalanced datasets. The metric is used to evaluate and compare all models. Choosing the incorrect metric can result in selecting the incorrect algorithm. The measure must reflect the most critical facts about a model or its forecasts for the project or its stakeholders [17]. Furthermore, essential indicators of models' performance are the trade-offs in the data: bias and variance. Bias in the data indicates the inaccuracy of the model's prediction compared to the data's actual values. The biases can occur during the training phase, where the model is 'simplified' to make the GT easier to predict. Alternatively, high variance indicates that the method learned the noise instead of the output. The high variance can cause overfitting. High variance and low bias relate to the high model complexity. The optimal model performance is the crossing point of bias error with variance error. Results of feature selection, feature correlation, and imbalance of the GT values can set expectations for the prediction capabilities. The test data results (see Figure 2) of the three datasets show high accuracy for all classifiers. Accuracy, the selected performance metric, describes the measure of correctly classified records. The results are presented in a box plot, Figure 2, showing the spread of the accuracy scores across data validation for each algorithm.

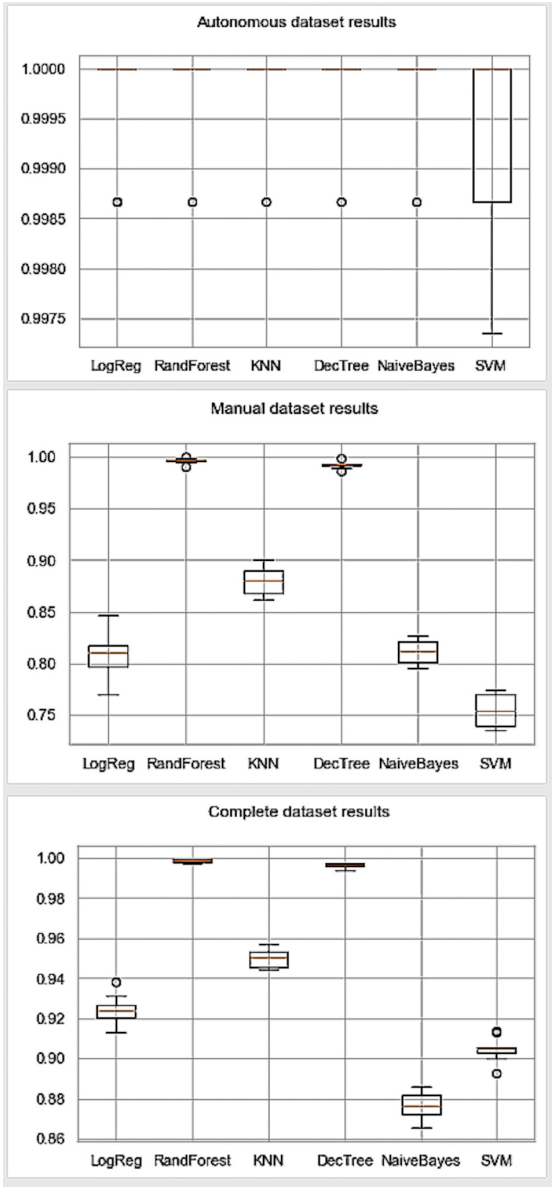

Fig. 2. Algorithm Comparison: Autonomous, Manual and Complete Datasets

a) Manually-Collected Sensor Data Classification: There is a considerable difference between linear and non-linear methods for the manually-collected dataset. As expected, non-linear methods, Decision Tree and Random Forest, performed with higher accuracy than all three linear methods. The prevalence of GT value '0' contributes to good prediction results. However, through observation of the confusion matrices resulted by linear methods, it is evident that the less represented GT values' prediction is erroneous. Generally, the performance of non-linear methods on this model is adequate, making Random Forest the most reliable classifier for this dataset.

b) Autonomously-Collected Sensor Data Classification: The autonomously-collected data illustrates significantly different results when compared to the manual data. The GT in the autonomous operation set contains only two values, 0 and 3. The 100% accuracy on the testing set can happen if the test set overlaps with the training set. However, in this case, the test and training sets are separate and not overlapping. In earlier anomaly detection results, the Isolation Forest has eliminated the samples with GT value 0, which can indicate a clear difference between these two values in the dataset. The poor distribution of the GT values results in an uncomplicated model that predicts with 100% accuracy. Arguably, the autonomous drone operated when the operation is unobserved (value 0) or even exclusively within the selected upstream environment (value 3). This model requires data complex enough to avoid bias and with a significantly less data imbalance.

c) Complete Sensor Data Classification: A complete dataset exhibits both previous analysis' results as a combination of manually and autonomously collected data. The high correlation of features in the data causes the model's high performance with Decision trees and Naive Bayes classifiers. Other classifiers that are less sensitive to high correlation, such as Logistic Regression or kNN, are sensitive to data quality, such as data imbalance in this dataset.

The results of the autonomous operation model do not meet reliability expectations. Therefore, repeated data collection methods in more complicated scenarios can improve the balance and complexity of the dataset [7]. Alternatively, manually-collected data proved to be inherently different from the autonomously-collected data. As a novel contribution, we suggest that future data is collected from planned manual and autonomous drone missions in more complex environments, recording the same sensor measures. A human operator of the manual drone should follow the same path as the preprogrammed autonomous drone. Following these requirements, we can obtain consistent data from both operations and avoid significant data differences.

V. Conclusion and Future Work

This paper analyzed the difference between manual operation and an autonomous operation of an underwater drone. We explicitly identified the similarities and the differences between the two operation modes through anomaly detection and classification methods during the analysis. Our research recognized the vital role of sensor data variations of different operation modes in the context of prevalent machine learning methods' performance and identified the gaps in which these methods underperformed. Unfortunately, unbalanced data is pervasive in research and industry, resulting in skewed classification results and reduced reliability for machine learning methods.

Future research will elaborate the connection between autonomous operations and the machine learning methods' performance with more varied and balanced testing data. Analysis of the sensor data through machine learning methods' performance can indicate reliability under more complex autonomous drone operations environments.

Acknowledgment

This research is a part of BRU21 – NTNU Research and Innovation Program on Digital and Automation Solutions for the Oil and Gas Industry (www.ntnu.edu/bru21) and supported by Equinor.

References

[1] L. Erhan, M. Ndubuaku, M. Di Mauro, W. Song, M. Chen, G. Fortino, O. Bagdasar, and A. Liotta, "Smart anomaly detection in sensor systems: A multi-perspective review," *Inf. Fusion*, vol. 67, no. September 2020, pp. 64–79, 2021. [Online]. Available: https://doi.org/10.1016/j.inffus.2020.10.001

[2] M. Z. Anwar, Z. Kaleem, and A. Jamalipour, "Machine Learning Inspired Sound-Based Amateur Drone Detection for Public Safety Applications," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2526–2534, 2019.

[3] S. Y. Choi and D. Cha, "Unmanned aerial vehicles using machine learning for autonomous flight; state-of-the-art," *Adv. Robot.*, vol. 33, no. 6, pp. 265–277, 2019. [Online]. Available: https://doi.org/10.1080/01691864.2019.1586760

[4] N. Moustafa and A. Jolfaei, "Autonomous detection of malicious events using machine learning models in drone networks," *DroneCom 2020 - Proc. 2nd ACM MobiCom Work. Drone Assist. Wirel. Commun. 5G Beyond*, pp. 61–66, 2020.

[5] D. De Dominicis and D. Accardo, "Software and sensor issues for autonomous systems based on machine learning solutions," in *2020 IEEE Int. Work. Metrol. AeroSpace, Metroaerosp. 2020 - Proc.*, 2020, pp. 545–549.

[6] A. Castellini, D. Bloisi, J. Blum, F. Masillo, and A. Farinelli, "Multivariate sensor signals collected by aquatic drones involved in water monitoring: A complete dataset," *Data Br.*, vol. 30, p. 105436, 2020. [Online]. Available: https://doi.org/10.1016/j.dib.2020.105436

[7] A. Castellini, G. Beltrame, M. Bicego, J. Blum, M. Denitto, and A. Farinelli, "Unsupervised activity recognition for autonomous water drones," *Proc. ACM Symp. Appl. Comput.*, pp. 840–842, 2018.

[8] F. T. Liu, K. M. Ting, and Z. H. Zhou, "Isolation forest," in *Proc. - IEEE Int. Conf. Data Mining, ICDM*. IEEE, 2008, pp. 413–422.

[9] V. Fonti, "Feature Selection using LASSO," *VU Amsterdam*, pp. 1–26, 2017.

[10] S. Yadav and S. Shukla, "Analysis of k-Fold Cross-Validation over Hold-Out Validation on Colossal Datasets for Quality Classification," in *Proc. - 6th Int. Adv. Comput. Conf. IACC 2016*, no. Cv. IEEE, 2016, pp. 78–83.

[11] Y. Y. Song and Y. Lu, "Decision tree methods: applications for classification and prediction," *Shanghai Arch. Psychiatry*, vol. 27, no. 2, pp. 130–135, 2015.

[12] M. Belgiu and L. Drăgu, "Random forest in remote sensing: A review of applications and future directions," *ISPRS J. Photogramm. Remote Sens.*, vol. 114, pp. 24–31, 2016.

[13] M. J. Islam, Q. M. J. Wu, M. Ahmadi, and M. A. Sid-Ahmed, "Investigating the Performance of Naive- Bayes Classifiers and K- Nearest Neighbor Classifiers." International Conference on Convergence Information Technology (ICCIT 2007), 2008, pp. 1541–1546.

[14] A. Bouzalmat, J. Kharroubi, and A. Zarghili, "Comparative study of PCA, ICA, LDA using SVM classifier," *J. Emerg. Technol. Web Intell.*, vol. 6, no. 1, pp. 64–68, 2014.

[15] J. Boyer, "Evaluate and select a machine learning algorithm - IBM Garage Practices." [Online]. Available: https://www.ibm.com/garage/method/practices/reason/evaluate-and-select-machine-learning-algorithm/

[16] L. Tolo̧si and T. Lengauer, "Classification with correlated features: Unreliability of feature ranking and solutions," *Bioinformatics*, vol. 27, no. 14, pp. 1986–1994, 2011.

[17] J. Brownlee, "Framework for Imbalanced Classification Projects," mar 2020. [Online]. Available: https://machinelearningmastery.com/framework-for-imbalanced-classification-projects/