



**Thiago da Silva Ribeiro**

**Machine Learning for Failure Detection in Bakery  
Industrial Effluents Treatment by Electrocoagulation**

**Tese de Doutorado**

Thesis presented to the Programa de Pós-Graduação em Engenharia de Materiais e de Processos Químicos e Metalúrgicos do Departamento de Engenharia Química e de Materiais of PUC-Rio in partial fulfillment of the requirements for the degree of Doutor em Engenharia de Materiais e de Processos Químicos e Metalúrgicos.

Advisor: Prof. Maurício Leonardo Torem  
Co-advisor: Prof. Brunno Ferreira dos Santos

Rio de Janeiro  
October 2022



**Thiago da Silva Ribeiro**

## **Machine Learning for Failure Detection in Bakery Industrial Effluents Treatment by Electrocoagulation**

Thesis presented to the Programa de Pós-Graduação em Engenharia de Materiais e de Processos Químicos e Metalúrgicos do Departamento de Engenharia Química e de Materiais of PUC-Rio in partial fulfillment of the requirements for the degree of Doutor em Engenharia de Materiais e de Processos Químicos e Metalúrgicos. Approved by the Examination Committee:

**Prof. Maurício Leonardo Torem**

Advisor

Departamento de Engenharia Química e de Materiais – PUC-Rio

**Prof. Brunno Ferreira dos Santos**

Co-advisor

Departamento de Engenharia Química e de Materiais – PUC-Rio

**Prof. Roberto Bentes de Carvalho**

Departamento de Engenharia Química e de Materiais – PUC-Rio

**Prof. Helon Vicente Hultmann Ayala**

Departamento de Engenharia Mecânica – PUC-Rio

**Prof. Maurício Bezerra de Souza Júnior**

UFRJ

**Prof. Flávio Vasconcelos da Silva**

UNICAMP

Rio de Janeiro, October 13<sup>th</sup>, 2022

All rights reserved.

### **Thiago da Silva Ribeiro**

The author received a bachelor's degree in Environmental Engineering from UVA (2015) and a master's degree in Materials and Chemical and Metallurgical Processes Engineering from PUC-Rio (2017). During his master's and doctoral studies, the author was awarded a Grade 10 scholarship from FAPERJ. He is currently employed as a data scientist at Radix.

### **Bibliographic data**

Ribeiro, Thiago da Silva.

Machine Learning for Failure Detection in Bakery Industrial Effluents Treatment by Electrocoagulation / Thiago da Silva Ribeiro; advisor: Maurício Leonardo Torem; co-advisor: Brunno Ferreira dos Santos. – 2022.

163 f. : il. color. ; 30 cm

Tese (doutorado) – Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Química e de Materiais, 2022.

Inclui bibliografia

1. Engenharia Química e de Materiais – Teses. 2. Aprendizado de Máquina. 3. Seleção de Atributos. 4. Espaço de Cores HSV. 5. Detecção de Falha. 6. Estação de Tratamento de Efluentes. 7. Eletrocoagulação. I. Torem, Maurício Leonardo. II. dos Santos, Brunno Ferreira. III. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Química e de Materiais. IV. Título.

CDD: 620.11

I dedicate this thesis work to my  
beloved family.

## Acknowledgments

My heartfelt thanks to God for the gift of life. This doctorate thesis was only possible because of the numerous supports I received along the way.

I would like to express my recognition to Professor Maurício for his assistance during the preceding seven years of my master's and doctoral studies. My gratitude goes to Professor Brunno for his constant encouragement since his arrival at PUC-Rio; his assistance has been key to the conclusion of this thesis. I would also like to thank the rest of the thesis committee for their support and insightful comments, which have greatly contributed.

I would like to acknowledge VentilAQUA, notably engineer António, for allowing the development of this investigation in order to collect real-world process data. António, appreciate helping me feel closer despite the fact that an ocean and a worldwide pandemic distanced us.

I was given the chance to be Isabela's father. Source of diversified learning and ongoing faith renewal in the future. I apologise for my absences, particularly this past year, when I was unable to participate in your development. Gratitude to my wife Déborah, partner for 20 years of a well-lived life, for doing everything I couldn't. Recognize that you made all of this possible.

I will never be able to thank my parents, Adão and Neuza, enough for always being there for me and supporting me. What a blessing it was for me to grow up surrounded by so much love and humanistic principles.

Friendships established throughout my master's and doctoral studies made this adventure much more enjoyable. I hope I have had as good an influence on you as you have had on me.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. I would also like to thank CNPq and FAPERJ for funding this research.

## Abstract

Ribeiro, Thiago da Silva; Torem, Maurício Leonardo (Advisor); dos Santos, Brunno Ferreira (Co-advisor). **Machine Learning for Failure Detection in Bakery Industrial Effluents Treatment by Electrocoagulation**. Rio de Janeiro, 2022. 163p. Tese de Doutorado - Departamento de Engenharia Química e de Materiais, Pontifícia Universidade Católica do Rio de Janeiro.

Electrocoagulation is an emerging wastewater treatment method that combines the benefits of coagulation, flotation, and electrochemistry. As a result of the inherent complexity of processes associated with wastewater treatment plants, it is difficult to respond swiftly and correctly to the dynamic circumstances that are necessary to ensure effluent quality. Therefore, this thesis aims to identify the operational condition of a wastewater treatment plant that has adopted electrocoagulation for treating bakery wastewater. Three operational conditions based on effluent clarification and reaction sludge were the target variables. The thesis is divided into two essays. The first endeavor used seven feature selection methods to select the most important features in a given dataset. The performance of neural network classification models trained on the original feature set was compared to the performance of those that were trained on a subset of features that had been curated using feature selection techniques. The model that utilised feature selection was found to have the best performance (F1-score = 0.92) and an improvement of more than 30% in preventing false positives. The second contribution brought a model that could detect anomalous process behavior using only wastewater surface color images from two small-size camera modules. The performance of various methods, including MLP, LSTM, SVM, and XGBoost was assessed. The LSTM model outperformed the others in terms of macro average Precision (84.620%), Recall (84.531%), and F1-score (84.499%), but the XGBoost model comes closely in second with Precision (83.922%), Recall (82.272%), and F1-score (83.005%).

## Keywords

Machine Learning; Feature Selection; HSV Color Space; Fault Detection; Wastewater Treatment Plant; Electrocoagulation.

## Resumo

Ribeiro, Thiago da Silva; Torem, Maurício Leonardo; dos Santos, Brunno Ferreira. **Aprendizado de Máquina para Detecção de Falhas no Tratamento de Efluentes Industriais da Indústria de Panificação por Eletrocoagulação**. Rio de Janeiro, 2022. 163p. Tese de Doutorado - Departamento de Engenharia Química e de Materiais, Pontifícia Universidade Católica do Rio de Janeiro.

A eletrocoagulação é um método emergente de tratamento de efluentes que combina os benefícios da coagulação, flotação e eletroquímica. Devido à complexidade inerente às operações de uma estação de tratamento de efluentes, é um desafio reagir com rapidez e precisão às condições dinâmicas necessárias para manter a qualidade do efluente. Portanto, esta tese tem como objetivo identificar a condição operacional de uma estação de tratamento de efluentes que adotou a eletrocoagulação para o tratamento de efluentes de panificação. Três condições operacionais baseadas em clarificação do efluente e lodo da reação foram as variáveis-alvo. A tese está dividida em dois ensaios. O primeiro usou sete métodos de seleção de atributos para selecionar as variáveis mais importantes em um determinado conjunto de dados. O desempenho dos modelos de classificação de redes neurais treinados no conjunto de atributos original foi comparado ao desempenho daqueles que foram treinados em um subconjunto curado usando técnicas de seleção de atributos. O modelo que utilizou a seleção de atributos apresentou o melhor desempenho (F1-score = 0,92) e uma melhoria de mais de 30% na prevenção de falsos positivos. A segunda contribuição trouxe um modelo que poderia detectar o comportamento anômalo do processo usando apenas imagens coloridas da superfície do efluente obtidas através de dois módulos de câmera de tamanho pequeno. O desempenho de vários métodos, incluindo MLP, LSTM, SVM e XGBoost foi avaliado. O modelo LSTM superou os outros em termos de Precisão (84,620%), Recall (84,531%) e F1-score (84,499%), mas o modelo XGBoost vem em segundo lugar com Precisão (83,922%), Recall (82,272 %) e F1-score (83,005%).

## Palavras-chave

Aprendizado de Máquina; Seleção de Atributos; Espaço de Cores HSV; Detecção de Falha; Estação de Tratamento de Efluentes; Eletrocoagulação.

## Table of contents

1	Introduction	16
1.1	Objective and contributions	18
1.2	Thesis outline	18
2	Machine learning in WWTP	20
2.1	Fault detection in wastewater treatment plant	24
2.2	Computer vision and colorimetric analysis for wastewater monitoring	28
2.3	Feature selection	33
2.4	Machine learning algorithms	37
2.4.1	Multi-Layer Perceptron (MLP)	38
2.4.2	Long Short-Term Memory (LSTM)	40
2.4.3	Support Vector Machine (SVM)	43
2.4.4	Extreme Gradient Boosting (XGBoost)	45
3	Electrocoagulation	48
3.1	Fundamentals of EC	51
3.2	EC reactor design	58
3.3	Main factors influencing the EC process	61
3.4	EC treatment of food processing wastewater	63
3.5	EC modelling	65
3.5.1	Statistical modelling	66
3.5.2	Modelling based on knowledge	68
3.5.3	Machine learning modelling	69
4	Fault detection in bakery industrial effluents treatment by electrocoagulation using neural networks and feature selection	72
4.1	Article Manuscript: Fault detection in bakery industrial effluents treatment by electrocoagulation using neural networks and feature selection	72
4.1.1	Introduction	72



4.1.2	Methods and materials	77
4.1.3	Results	90
4.1.4	Conclusions	101
5	Computer vision-based monitoring of a bakery industrial effluents treatment by electrocoagulation via a machine learning approach	103
5.1	Article Manuscript: Computer vision-based monitoring of a bakery industrial effluents treatment by electrocoagulation via a machine learning approach	103
5.1.1	Introduction	103
5.1.2	Methods	107
5.1.3	Model	112
5.1.4	Results and discussion	117
5.1.5	Conclusions	130
6	Conclusions	131
	Bibliography	133
	A Model construction algorithm	149

## List of figures

Figure 1 - Trends in artificial intelligence-based wastewater treatment publications between 1995 and 2019 (adapted from Zhao et al., 2020).	22
Figure 2 - Loop for process monitoring (adapted from Md Nor; Che Hassan; Hussain, 2020).	24
Figure 3 - Framework of classification for fault detection methods (adapted from Mattera et al., 2018).	25
Figure 4 - Visible light electromagnetic spectrum (adapted from Frauenfelder, 2021).	29
Figure 5 - RGB colour space (adapted from Broek, 2005).	30
Figure 6 - HSV colour space (adapted from Chen; Wang; Lal, 2020).	31
Figure 7 - Cross-industry standard process (CRISP) for machine learning (adapted from Plotnikova; Dumas; Milani, 2020).	33
Figure 8 - Feature selection methods, such as filter, wrapper, and embedded method (adapted from Xie et al., 2020).	34
Figure 9 - Venn diagram illustrating the relationship between each subset of artificial intelligence (adapted from Vrana; Singh, 2020).	38
Figure 10 - Structure of a Perceptron (adapted from Rosenblatt, 1958).	39
Figure 11 - LSTM unit (adapted from Sharma et al., 2021).	41
Figure 12 - Linear SVM model (adapted from Huang et al., 2018).	43
Figure 13 - XGBoost's architecture layout (adapted from Kiangala; Wang, 2021).	45

Figure 14 - Interconnected and intricate nature of the EC process (adapted from Moneer; El Nemr, 2012).	50
Figure 15 - EC's domains of knowledge (adapted from Bhagawati et al., 2022).	51
Figure 16 - E-pH diagram of iron at 25 °C (generated using the HSC chemistry software).	54
Figure 17 - E-pH diagram of aluminium at 25 °C (generated using the HSC chemistry software).	54
Figure 19 - Electrical double layer diagram (adapted from Tahreen; Jami; Ali, 2020).	57
Figure 19 - Orientation of wastewater flow (view from the side): (a) horizontal flow; (b) vertical flow (adapted from Lu; Zhang; Li, 2021).	59
Figure 20 - Electrodes arrangements in EC cells (adapted from Garcia-Segura et al., 2017).	60
Figure 21 - Network visualization of EC modelling research (generated using the VOSviewer software).	66
Figure 22 - The relative positions of the sensors in the WWTP.	79
Figure 23 - The ranking of features using the different feature selection methods.	91
Figure 24 - Searching for optimal hyperparameter configuration for both groups (with and without feature selection).	94
Figure 25 - Box plot described the macro-averaged F1-score of the models trained on datasets with and without feature selection.	95
Figure 26 - Box plot displaying the ROC-AUC values of models trained on datasets with and without feature selection.	96
Figure 27 - The normalised confusion matrix of the best model that utilised feature selection.	98
Figure 28 - The normalised confusion matrix for the best model that did not utilise feature selection.	98

Figure 29 - Training and validation loss during the learning process of the model that utilised feature selection.	100
Figure 30 - Training and validation loss during the learning process of the model that did not utilise feature selection.	101
Figure 31 - ESP32-CAM modules locations in the WWTP.	109
Figure 32 - Histograms of H values in the EC unit's inlet and after EC process.	110
Figure 33 - Histograms of S values in the EC unit's inlet and after EC process.	111
Figure 34 - Histograms of V values in the EC unit's inlet and after EC process.	111
Figure 35 - Overall computational flowchart.	116
Figure 36 - Searching for optimal hyperparameter configuration on an NVIDIA Tesla P100-PCIE-16GB for each algorithm.	117
Figure 37 - Training time on an NVIDIA Tesla P100-PCIE-16GB for each algorithm.	118
Figure 38 - Boxplot of 5-fold cross-validation evaluation metrics using four distinct machine learning algorithms (Class 0: Not clarified, showing turbidity).	119
Figure 39 - Boxplot of 5-fold cross-validation evaluation metrics using four distinct machine learning algorithms (Class 1: Clarified, showing low turbidity).	120
Figure 40 - Boxplot of 5-fold cross-validation evaluation metrics using four distinct machine learning algorithms (Class 2: Clarified, although the system had an excessive electrode and energy consumption).	121
Figure 41. Hyperparameters search results for the MLP algorithm.	122
Figure 42 - Hyperparameters search results for the LSTM algorithm.	124
Figure 43 - Hyperparameters search results for the SVM algorithm.	125

Figure 44 - Hyperparameters search results for the XGBoost algorithm. 126

Figure 45 - Confusion matrices comparing the best performing model for each algorithm for the testing dataset. 128

## List of tables

Table 1 - Overview of the number of variables and observations in the dataset.	80
Table 2 - Characteristics of the dataset.	80
Table 3 - Information related to the feature-selection methods used in this paper.	87
Table 4 - Hyperparameters tested.	88
Table 5 - Aggregated ranking of the features in the dataset.	93
Table 6 - Topology of the neural network.	96
Table 7 - Performance metrics.	97
Table 8 - Hyperparameters tested (MLP).	113
Table 9 - Hyperparameters tested (LSTM).	113
Table 10 - Hyperparameters tested (SVM).	114
Table 11 - Hyperparameters tested (XGBoost).	114
Table 12 - Best MLP model hyperparameter configuration.	123
Table 13 - Best LSTM model hyperparameter configuration.	124
Table 14 - Best SVM model hyperparameter configuration.	125
Table 15 - Best XGBoost model hyperparameter configuration.	127

## List of Abbreviations

EC – Electrocoagulation  
WWTP – Wastewater Treatment Plant  
ORP – Oxidation-Reduction Potential  
ANN – Artificial Neural Network  
SVM – Support Vector Machine  
XGBoost – Extreme Gradient Boosting  
LSTM – Long Short-Term Memory  
MLP – Multi-Layer Perceptron  
ANOVA – Analysis of Variance  
MI – Mutual Information  
RFE – Recursive Feature Elimination  
SHAP – Shapley Additive Explanations  
PI – Permutation Importance  
RF – Random Forest  
DAF – Dissolved Air Flotation  
PLC – Programmable Logic Controller  
BO – Bayesian Optimisation  
ROC – Receiver Operating Characteristic Curve  
ROC-AUC – Area under the Receiver Operating Characteristic Curve  
RGB – Red, Green, Blue  
HSV – Hue, Saturation, Value

# 1

## Introduction

Not only is our existence dependent on water, but so is our economic well-being. Water plays a part in all of our production processes. There are no replacements, and while it is renewable, its supply is limited. Today, everyone is worried about the possibility of a water shortage in the face of rising, mostly population-driven, water needs, as well as the repercussions this might have on our energy and food production (Diaz-Elsayed et al., 2019).

Multiple factors contribute to water pollution, including industrial wastes, mining operations, sewage, chemical fertilisers, energy use, among others (Mousazadeh et al., 2021). Constant efforts must be undertaken to safeguard water supplies in this situation.

In general, the challenges faced during wastewater treatment are rather complicated, since effluent comprises numerous kinds of contaminants based on its source. Consequently, there are several sorts of effluents to treat, each with its own properties necessitating unique treatment techniques (Crittenden et al., 2012).

The bakery industry is one of the world's most important food industries, and its manufacturing size and methods vary greatly (Jerome; Singh; Dwivedi, 2019). While baking effluent normally does not include toxic compounds, it is rich in organic matter (mostly flour and sugar) and oil/grease. Little levels of detergents, yeast, salt, and other food additives are also present (Mohan; Vivekanandhan; Priyadharshini, 2017). The significant daily water consumption in the bakery industry (10–300 thousand gallons, mostly utilised for cleaning operations) is also an environmental issue, especially considering that at least half of this water is disposed as wastewater (Chen et al., 2004).

By creating coagulants on-site with sacrificial anodes, the electrocoagulation (EC) technique is useful for removing a variety of contaminants from water and wastewater. In the EC process, entrapment in precipitation (sweep coagulation), adsorption, and charge neutralisation are the primary pollutant removal processes.



Among the existing treatment techniques, EC is a potential alternative wastewater treatment approach because to its lower operating costs, simple design, quick sedimentation, little or no chemical addition, and low sludge production (Das; Sharma; Purkait, 2022). Today, the foremost objective of chemical and environmental engineers is to design a wastewater treatment plant (WWTP) that can permit decentralised wastewater treatment (Alabi; Telukdarie; Van Rensburg, 2019). EC is an effective decentralised technique in this regard.

Due to its capacity to remove a vast array of organic and inorganic pollutants, EC has been extensively used. It has been used widely for the treatment of a variety of industrial wastewaters, including textile wastewaters (Suhan et al., 2020), pharmaceutical wastewaters (Ensano et al., 2019), mining wastewaters (Ribeiro et al., 2019), dairy wastewaters (Akansha et al., 2020), and petroleum wastewaters (Hansen et al., 2019), among others.

Industry 4.0 was designed from the start to solve a number of the issues affecting the globe today, including resource use and energy efficiency (Ghobakhloo, 2020). Industry 4.0 does not require stand-alone technology for any of these processes, but rather a mix of technologies, procedures, and digital solutions. Internet of things also plays a significant role in real-time process control. Machine learning can often build the required connections between various monitoring sensors and derive conclusions from real-time data. Such machine learning-assisted analytic procedures might be carried out constantly, with periodic findings preventing process failure (Bufler et al., 2017).

In recent years, WWTPs have been exposed to an unprecedented quantity of data as a result of falling sensor costs, the rising prevalence of wireless connectivity, and the proliferation of mobile devices able to continuously collect data and do complex computations (Kijak, 2021).

Increasing automation of WWTP enables access to massive data and the development of data-driven solutions (Newhart et al., 2019). The majority of fault detection systems are data-driven because they can discover abnormal circumstances more quickly, are simpler to deploy, and need less previous expertise (Md Nor; Che Hassan; Hussain, 2020).

Recent developments in data-driven process and performance monitoring may provide the wastewater treatment sector with a chance to decrease costs and

enhance operations. Due to the nonlinearity and growing complexity of the current wastewater treatment process, data-driven fault detection techniques are in demand (Dairi et al., 2019).

## 1.1

### Objective and contributions

This study aims to identify the operational condition of a WWTP that has adopted EC for treating bakery wastewater. Three operational conditions based on effluent clarification and reaction sludge were the target variables. Initially, eleven features, including conductivity, pH, flow rate, voltage, current, polarity, and oxidation-reduction potential (ORP), were monitored. Using an Artificial Neural Network (ANN), the effectiveness of several feature selection methods (filter, wrapper, and embedded) relative to the original dataset (with all features) was evaluated. Furthermore, images of the surface of the wastewater were captured using a dedicated image acquisition system. The mean values of the color channels (HSV) across all pixels were selected as features for numerous algorithms, including ANNs, Support Vector Machines (SVMs), and Extreme Gradient Boosting (XGBoost). Several metrics, such as the F1-score, precision, and recall, were used to compare the models.

Thus, the contributions presented in this thesis are:

1. Address a knowledge gap on data-driven fault detection in EC-based WWTPs. There is presently limited research that employ realistic EC operational conditions;
2. Apply a better comprehension and relative importance of the features for the EC process;
3. The nonexistence, to the author's best knowledge, of a data-driven model that could detect anomalous process behavior using only wastewater surface color images.

## 1.2

### Thesis outline

This thesis is organised into chapters covering the following topics:

- **Chapter 2:** Review of the current literature on machine learning algorithms for the operation of WWTPs, with a focus on fault detection. Theoretical understanding of the algorithms and feature selection methods used in this thesis;
- **Chapter 3:** Review of the most relevant literature on the modelling of wastewater treatment by EC. Understanding of EC's fundamental aspects;
- **Chapter 4:** A method for selecting features by evaluating seven distinct feature selection strategies for fault detection in WWTP operating conditions using ANNs;
- **Chapter 5:** A computer vision-based machine learning model for fault detection in WWTP operating conditions using wastewater surface color images from two small-size camera modules;
- **Chapter 6:** Conclusions about the results obtained in the two case studies and suggestions for future work.

According to the requirements of Pontifical Catholic University of Rio de Janeiro, chapters 4 and 5 are author-approved versions of unpublished manuscripts. The candidate was the principal author, having developed the investigations, analyzed the data, and written the manuscripts. All the other authors provided their experience in a variety of aspects, critically reviewed for significant intellectual substance, and gave final permission to the version to be published. VentilAQUA, a Portuguese company, enabled these investigations by granting access to their database and allowing for research.

## 2

## Machine learning in WWTP

Initially, industry 4.0 was conceived of as the fourth revolution to emerge in the industrial sector, but this conception has developed over the last several years. Industry 4.0 now entails the digital transformation of both industrial and consumer markets, from the introduction of smart production to the digitization of all value delivery channels (Xu; Xu; Li, 2018).

Industry 4.0 technologies provide crucial prospects for future innovation and corporate expansion. Industry 4.0 is being implemented using artificial intelligence, internet of things, big data, and other emerging technologies (Olsen; Tomlin, 2020).

The virtualization concept of industry 4.0 and the transfer of sensor data obtained from the real world into simulation-based models of smart components throughout the value network provide great potential for operation prediction and optimization (Oztemel; Gursev, 2020).

Production maintenance would include processing vast amounts of data through continuous real-time monitoring and providing alarms based on predictive approaches such as ANNs. Particularly, industry 4.0 offers new opportunities for enhanced asset management, including real-time remote tracking, intelligent monitoring, and alarm-driven preventative maintenance (Zonta et al., 2020).

When evaluating the resource component of a new industrial development, it is prudent to prioritise fundamental resources such as water. The impossibility to substitute water with other sources and the complication surrounding its efficient distribution make water a crucial resource within the context of industry 4.0 and sustainable development (Ghobakhloo, 2020).

Recent years have seen the water sector embrace the industry 4.0 movement, often known as water 4.0 or smart water. The water 4.0 plan is the idea of the Germany Water Partnership and intends to convert present conventional water systems into water infrastructure and management systems of the twenty-first century. This will enable decentralisation of water management systems. The water

sector will employ the daily data set to make intelligent decisions that improve the availability and quality of water (Alabi; Telukdarie; Van Rensburg, 2019).

According to Bufler et al. (2017), water 4.0 emphasizes digitalization and automation on a framework for resource-efficient, adaptable, and competitive water management. As a consequence, water 4.0 contains the same key characteristics and terminology as industry 4.0 and integrates them into a systematic, water management environment. Additionally, water 4.0 facilitates the development of sustainable water management systems (Kijak, 2021).

Artificial intelligence and machine learning are technologies/methods used in the water sector to provide leverage for quality water treatment processes by identifying water-related problems in advance and notifying the water field engineer in real-time. Internet of things is a potential technology for water industry since it enables remote real-time monitoring of water-related concerns (Yasin et al., 2021). It is anticipated that big data and analytics technology would turn the water business into a fully data-driven sector (Alabi; Telukdarie; Van Rensburg, 2019).

Solano, Krause and Wollgens (2022) proposed and assessed a sensing and actuating system enabled by the internet of things for localising illicit industrial discharges of polluted wastewater in sewer networks. Designing an internet of things system and its real-time algorithm for anomaly identification and localization in wastewater networks presented unique challenges due to the characteristics of the sewer environment. The internet of things system, including its anomaly detection and localization algorithm, was built in a low-power microcontroller and tested in running wastewater containing several types of hazardous industrial waste.

Cicceri et al. (2021) developed a method based on a smart system to maintain the efficacy of the WWTP and assure clean water quality. It is able to monitor the water's purity and the inlet and outlet flows in real time, implementing the appropriate regulations depending on the observed values. The raw data created by an internet of things platform as part of a real-world case study executed in Italy is gathered and housed on a server that can analyse and handle real-time plant information.

Emerging artificial intelligence and machine learning, in conjunction with smart technologies, are filling a need in water applications that was previously neglected by conventional approaches and ways of thinking (Lowe; Qin; Mao,

2022). Artificial intelligence, machine learning, and smart technologies are anticipated to model and solve complex challenges in water applications by virtue of their generalizability, robustness, and relative simplicity of design in order to reduce costs and improve operations. Water applications that have made significant use of machine learning include water and wastewater treatment, monitoring of natural systems, and precision/water-based agriculture (Zhao et al., 2020).

With the advancement of artificial intelligence, Zhao et al. (2020) reported that the number of publications using artificial intelligence to wastewater treatment research was 19 times higher in 2019 than in 1995, and papers had 36 more citations on average (Figure 1). There are around 150 references to studies exploring the uses of ANN models for simulating and forecasting the operation of WWTPs.

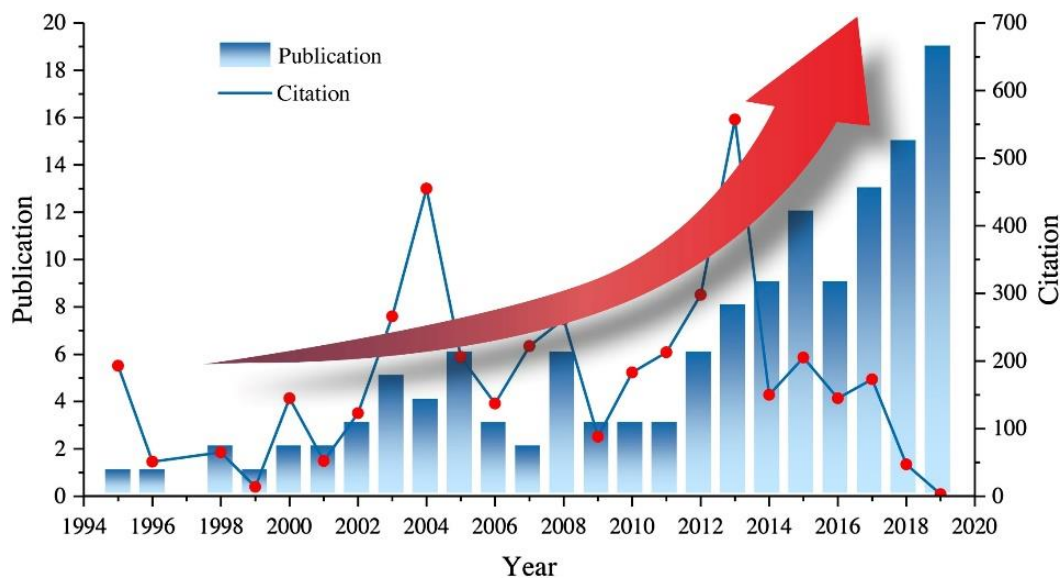


Figure 1 - Trends in artificial intelligence-based wastewater treatment publications between 1995 and 2019 (adapted from Zhao et al., 2020).

Based on the results of artificial intelligence models and the autoregressive integrated moving average model, Nourani, Asghari and Sharghi (2021) determined that artificial intelligence models are more suitable for predicting WWTP parameters than the autoregressive integrated moving average linear model. Although the linear model offered acceptable results in forecasting effluent and influent biological oxygen demand parameters, it has low performance in comparison to artificial intelligence models owing to the linear model's limited capacity to handle nonlinear and nonstationary time series.

Due to the complexity and non-linearity of wastewater treatment facilities, Hejabi et al. (2021) showed that mathematical models are insufficient to explain their behavior. Artificial intelligence models are thus an alternative to linear approaches. Using the Monte Carlo uncertainty approach, the ANN and SVM models' accuracy was evaluated. Depending on the complexity of the investigation, the findings of the uncertainty analysis demonstrated that the artificial intelligence approaches may be dependable under varied scenarios.

Forecasting of WWTP important features helps grasp and forecast plant behavior in order to enhance process design and controls, increase system dependability, minimize operating costs, and promote performance optimization. Cheng et al. (2020) used a municipal wastewater treatment facility dataset. In addition, the exponential smoothing filter is used to preprocess the noisy raw data before applying deep learning. Models are evaluated using mean absolute percentage error and root mean square error. In terms of efficiency, the gated recurrent unit converges much faster than LSTM. The trained model may then be included into optimization scenarios to provide data-driven recommendations for optimum daily WWTP operation, pollutant removal, and cost reduction.

Newhart et al. (2021) demonstrated the capability of machine learning to estimate the near-real-time peracetic acid disinfection performance of a full-scale wastewater treatment facility, taking water quality and operational changes into account. The authors have demonstrated that traditional and batch literature models for peracetic acid concentration do not accurately describe the full-scale disinfection performance of peracetic acid for secondary wastewater effluent at the treatment facility, and that a machine learning modeling approach could supplement expensive analyzers in process control. For instance, if a recurrent neural network is employed to forecast pre- and post-disinfection concentrations, deviations from the predictions might offer signals of process disruptions or inefficiencies to wastewater treatment facility operations. An ANN might reliably forecast real-time for direct control.

Fault detection is a prominent use of machine learning algorithms in WWTPs. The section that follows will highlight some of the most recent contributions to this field.

## 2.1

### Fault detection in wastewater treatment plant

A fault is the unwanted divergence of at least one distinctive feature of a system from its normal, acceptable, or standard state. As seen in Figure 2, after the successful diagnosis of a fault, process recovery is performed to close the monitoring loop. Modern process systems are huge in size and complexity, resulting in an increase in requirements for their safety and dependability. There is a rising interest in creating solutions to handle faults that occur in industrial process, therefore ensuring safe and efficient outputs (Li et al., 2020). Fault detection is the essential technique for solving this issue.

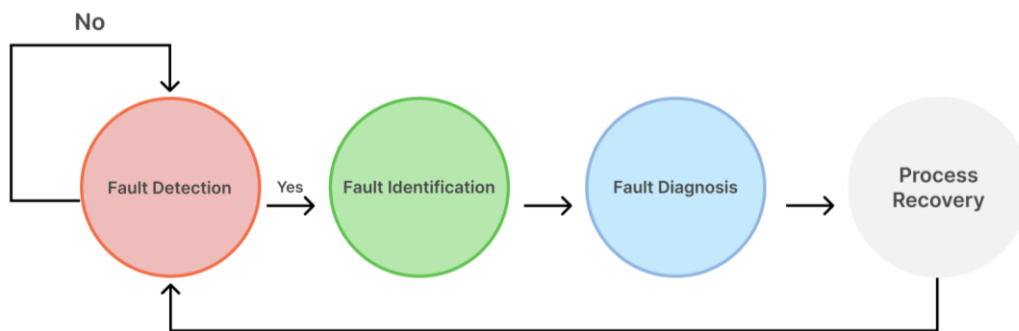


Figure 2 - Loop for process monitoring (adapted from Md Nor; Che Hassan; Hussain, 2020).

Fault detection may be accomplished by first-principles, data-driven, or knowledge-based methods (Figure 3). First-principle methods need the development of a mathematical model based on theoretical understanding. This method is often ineffective due to the intricacy of the generated mathematical model. The knowledge-based method, on the other hand, requires previous comprehension or knowledge of the links between faults and model parameters or states. It is also difficult to apply this method to large-scale systems due to the time and expertise necessary to create these complicated fault models (Venkatasubramanian et al., 2003).



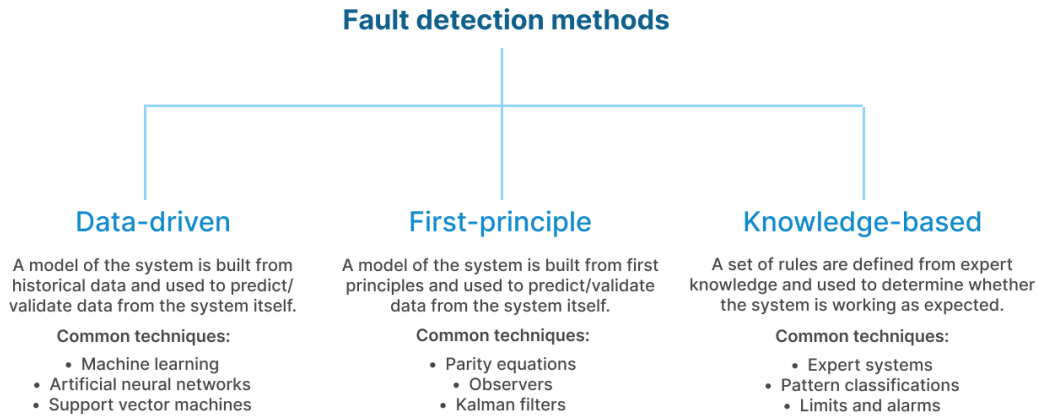


Figure 3 - Framework of classification for fault detection methods (adapted from Mattera et al., 2018).

The technique of using parity equations to produce residuals is a solution based on first principles. This methodology is based on the variances between predicted process states or outputs and actual values. It is possible to obtain the parity equations from the mathematical model equation. There are several methods for designing parity equations. These include discrete state-space models, time continuous state-space models, and transfer function models, among others (Fagarasan; Iliescu, 2008). The basic notion is to use real measurements to verify the parity (consistency) of the system's mathematical equations (analytical redundancy relations). Regardless, the process parameters must be understood in advance. This model-based fault detection approach executes a model ( $H_M$ ) concurrently with the process ( $H_P$ ), therefore detecting an output error (Equation 1).

$$r_y(s) = [H_P(s) - H_M(s)]u(s) \quad (1)$$

where  $H_P$  represents the transfer function,  $H_M$  represents the process model,  $r_y$  represents the output error, and  $u$  represents the input variables.

Due to the nonlinearity and growing complexity of the contemporary process industry, data-driven techniques are in demand. The primary downside of a data-driven strategy is that it requires vast quantities of raw historical data. This has been mostly irrelevant for some years, since the widespread use of distributed control systems and soft computing technologies has simplified the data collecting process. In addition, database and data mining technologies provide dependable

technological support for the development of data-driven modeling techniques in industrial processes (Md Nor; Che Hassan; Hussain, 2020).

Data-driven analytical approaches rely substantially on the obtained data type. It is essential to comprehend the unique structure and qualities of the data to decide the data's organization and use. A wastewater treatment facility collects data from a range of sources, including laboratory analysis, online sensor readings, operations and maintenance management, customer and technology manufacturer data (Newhart et al., 2019).

Historically, WWTPs have lacked data-driven process management, with daily operational choices regarded more of an art than a science. Despite the specific difficulties presented by WWTP data, data-driven system automation and real-time control are essential to the functioning of a contemporary WWTP. To decrease the effect of a fault on effluent water quality, plant operators must be ready to react swiftly to a system malfunction in order to avoid equipment damage or system failure (Mamandipoor et al., 2020).

Luca et al. (2021) proposed a principal components analysis fault detection approach for the dissolved oxygen sensor utilized in the case study wastewater treatment facility. Using a calibrated WWTP model, datasets corresponding to normal and malfunctioning sensor behavior were generated. Bias, fixed-value, and complete-failure faults were quickly identified. Supporting the safe and effective management of complex, nonlinear, time-varying, and regulated wastewater treatment processes, the provided model is of great practical value.

Kini and Madakyaru (2022) presented a data-driven fault detection technique based on the Kantorovich distance for monitoring sensor defects in WWTPs. In this research, the Kantorovich distance measure is integrated with a modeling framework based on dynamic principal component analysis. This metric calculates the difference between two datasets and utilizes it as a measure of failure. The Kantorovich distance measure is calculated by comparing the residuals of typically functioning data against anomalous data. The simulation findings demonstrate that the Kantorovich distance metric is better to other based fault indicators.

Dairi et al. (2019) developed data-driven unsupervised anomaly detection methodologies based on deep learning techniques and clustering algorithms to monitor and identify influent conditions for sustainable and resilient WWTP

operations. This technique introduced and integrated the capacity to distinguish temporal relationships in multivariate time series from recurrent neural network and the sensitivity to detect faults from one-class SVMs. The features consist of seven years of daily measurements of twenty-one variables from a municipal wastewater treatment facility, including temperature, pH, conductivity, and chemical oxygen demand. Compared to stand-alone clustering approaches, the model exhibited the highest efficiency and precision.

Mihály, Simon-Várhelyi and Cristea (2022) investigated the design and training of ANN models for predicting energy and effluent quality indices with the intention of using them to accelerate the optimization of WWTPs to determine the optimal setpoint values for the nitrates and dissolved oxygen control loops. The measurements of WWTP influent and process variables were used to collect plant data for a period of 22 days, with a sample interval of 30 minutes. The screening phase revealed that networks trained directly with effluent quality as a single output variable had a greater potential for making accurate predictions, that recurrent neural networks with two hidden layers were the least promising due to their excessively long training duration, and that radial basis neural networks were the most promising type.

Han et al. (2018) developed a soft sensor approach to monitor effluent total phosphorus and ammonia nitrogen concentrations. In this monitoring system, a fuzzy neural network is employed to create the soft sensor model, and principal component analysis is utilised to choose the input features. This system includes on-line sensors for dissolved oxygen, pH, temperature, ORP, and total suspended solids, among others. The model was evaluated in a full-scale wastewater treatment facility, and the findings showed that the monitoring values closely matched the detection values with a narrow margin of error and excellent precision. Specifically, the reaction time of the suggested system is faster than existing techniques, enhancing its capacity to address difficulties associated with real-time monitoring and control.

The vast number of implementations of data-driven models is attributable to their simple formulation that does not need complete understanding of the system model, their capacity to decrease the time and cost of model deployment, their straightforward implementation, and their flexibility. In addition, these methods can

accommodate high-dimensional and coupled process variables, particularly in complex and large-scale systems (Md Nor; Che Hassan; Hussain, 2020). Nevertheless, it is important to note that there is currently a knowledge gap in the development of data-driven models to detect faults in EC-based process.

## 2.2

### **Computer vision and colorimetric analysis for wastewater monitoring**

The eyes of Industry 4.0 are machine vision systems with optical imaging capabilities and innovative real-time monitoring sensors (Javaid et al., 2022). Image processing and computer vision represent the most current advancements in computer science. It employs information science, technology, and mathematics, among other fields of expertise. Computer vision has shown to be very adaptable, precise, reproducible, and cost-effective (Alonso et al., 2019). Such benefits have prompted manufacturers to use machine-vision techniques.

In the life sciences, colour is discussed in terms of the human visual process and perception in the optical observer's consciousness. The attribute known as colour is determined by the combined intensities of the wavelengths contained in a beam of visible light reflected off the surface of an object (Hastings; Rubin, 2012). Visible light is a kind of electromagnetic radiation with wavelengths between 380 and 780 nanometers that can be perceived by the human eye (Figure 4). To be seen, an object must either produce light or reflect or transmit incident light from an external source. The perception of colour may be attributed to a physiological reaction to a physical input.

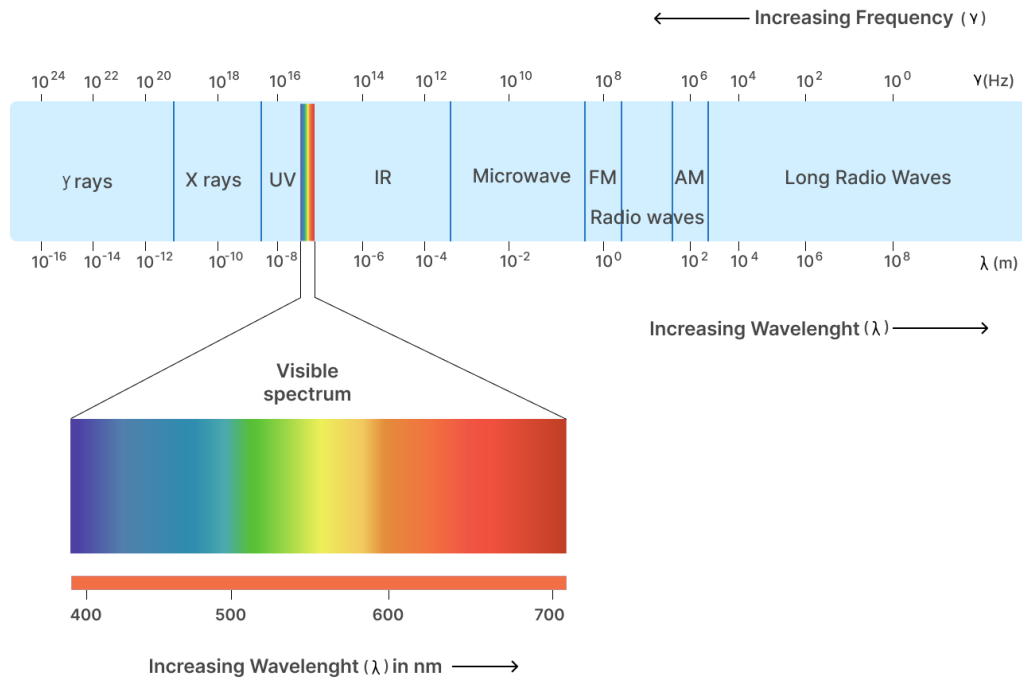


Figure 4 - Visible light electromagnetic spectrum (adapted from Frauenfelder, 2021).

In addition to processing reflected light from an object's surface, colorimeters and spectrophotometers convert light flux data into colour parameters that may be used to assess and compute colour differences (Fernandes et al., 2020). Lighting is an essential component in computer vision systems. Vision systems, like human eyes, are impacted by the intensity and quality of light. Illumination devices provide light that exposes the examined target objects (Capitán-Vallvey et al., 2015).

Numerous ways for measuring and mathematically expressing colour have been developed, allowing for more precise colour communication (Fan et al., 2021). A "colour space" is a particular arrangement of colours. Since "colour space" refers to a specific combination of colour model and mapping function, it is often used to refer to a colour model informally.

A unit cube is widely used to represent the colour space of computer-based display systems. In RGB space (Figure 5), each colour (Red, Green, and Blue) corresponds to one of the three orthogonal coordinate axes. Along each axis of the colour cube, the colours vary from having no contribution to a completely saturated hue. The colour cube is solid, and each point (colour) inside the cube is determined by a triple of R, G, and B values. The diagonal line of the cube from black (0, 0, 0) to white (1, 1, 1) symbolises all the greys, while the red, green, and blue axes, respectively, represent all the colours (Broek, 2005).

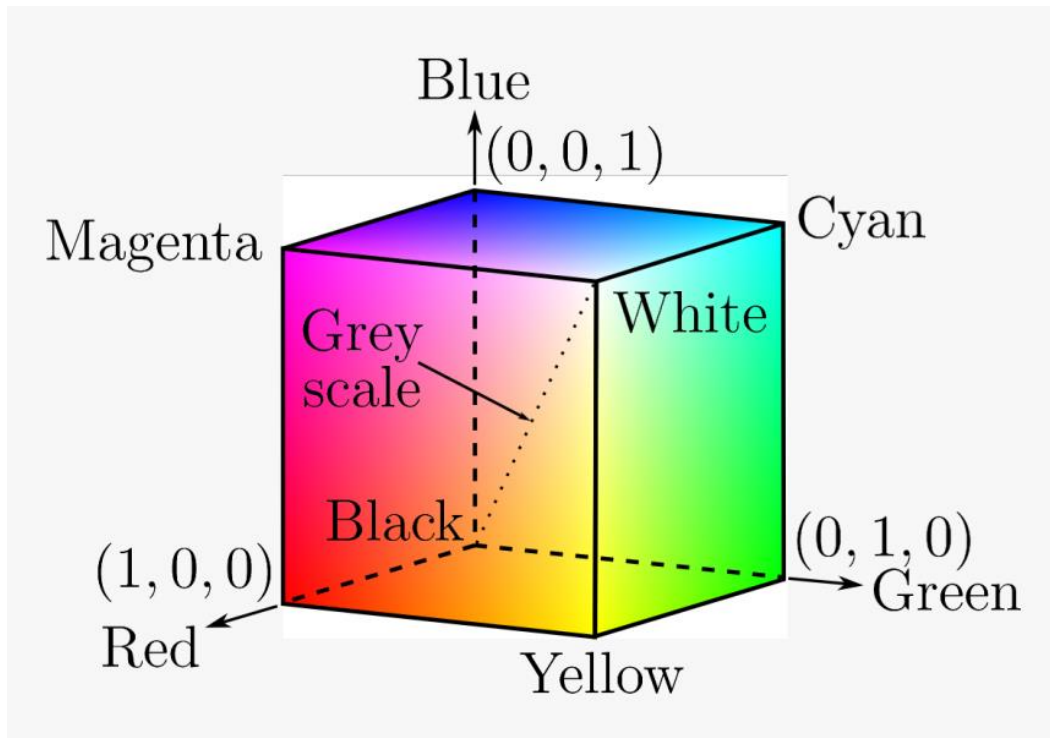


Figure 5 - RGB colour space (adapted from Broek, 2005).

The HSV colour space (Figure 6) is a nonlinear modification of RGB based on a mathematical paradigm. Hue (H), saturation (S), and value (V) are the three primary components of HSV. Hue is defined as the length of the radiation spectrum and ranges from  $0^\circ$  to  $360^\circ$  ( $0^\circ$  is red and  $55^\circ$  is a shade of yellow, for example); saturation is defined as colour purity, with intensity ranging from 0% to 100% (0% means no colour and 100% intense colour); and value is the brightness of the colour, also ranging from 0% to 100% (Liu; Chen; Fang, 2018).

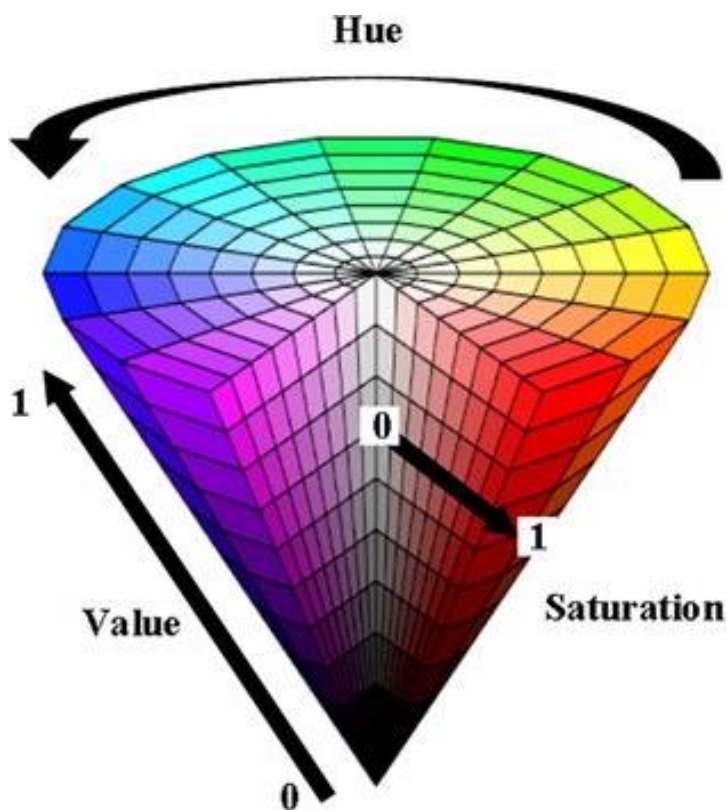


Figure 6 - HSV colour space (adapted from Chen; Wang; Lai, 2020).

Colourimetry is the measurement of colour changes in solutions or other kinds of analytical equipment employing optical or image detectors, using the colour coordinate or a corresponding magnitude to construct the qualitative and quantitative analytical parameter (Fernandes et al., 2020). Color is therefore included into the vast array of qualities utilised for chemical and physical sensing. An efficient colour representation must operate on the whole colour space to capture channel correlations, encode the colour variety of the database without adding redundancy, and eliminate colour artefacts (Duchesne; Liu; Macgregor, 2012).

Consequently, the emergence of imaging technologies, along with the study of colour features, has made it feasible to objectively measure colour for the purpose of wastewater chemical information.

Xing et al. (2022) suggested a cellphone-based colorimetric multi-channel sensor with good sensitivity and stability for measuring several environmental pollutants concurrently. In order to increase the sensitivity of the sensor, a delicate optical path system was constructed utilising a diffraction grating to divide six white beams passing through numerous coloured samples, allowing the camera on the mobile phone to record the diffracted light for image analysis. As a successful

proof-of-concept, the sensor was used to concurrently detect turbidity, orthophosphate, ammonia nitrogen, and three heavy metals with excellent sensitivity and consistency. Due to the sensor's low cost, simple operation, excellent mobility, and multi-index measurement, it is extremely applicable and significant in the fields of environmental monitoring, diagnostics, and early warning detection.

Das, Chetry and Nath (2021) suggested a low-cost, small, field-portable colorimetric analyzer based on a smartphone for the measurement of phosphate content in aqueous medium. The camera phone takes photographs of the reagent-treated test samples; then, the RGB colour model of the images is transformed to the HSV colour model. An Android application converts the RGB colour model to the HSV colour model. The V-channel value in HSV colour space is correlated with phosphate content. By measuring the absorbance at a particular wavelength, the acquired findings are compared to the laboratory-grade reference spectrophotometer. The developed smartphone sensing device can test phosphate content with high accuracy and precision (2% relative standard deviation). As it offers a rapid and user-friendly analytical platform for in-field applications, the sensing technology may be employed as an alternative to the current detection sensors.

Damirchi et al. (2019) developed a straightforward and innovative kinetic spectrophotometric approach for the sensitive and highly selective measurement of Brilliant Green. Based on the interaction of Brilliant Green with Triton X-100 in micellar medium at room temperature, the approach was developed. This approach compares the performance of the spectrophotometer with a cheap and readily accessible detector (digital compact camera). The suggested approach was very straightforward and quick, since it omitted the time-consuming stages of sample preparation and setting up experimental conditions. The given approach may also be utilised as a field test when a digital camera is used as a detector. The suggested approach is more sensitive than conventional colorimetric techniques and has superior selectivity for distinguishing Brilliant Green from other triphenylmethane dyes.

This brief review showed that the authors produced good findings for specific cases. Besides the advancements made over the previous decade, real-world



applications, such as WWTP monitoring in real time, still face several challenges. Overall, machine vision with suitable image processing algorithms is a very promising tool for wastewater monitoring.

## 2.3 Feature selection

Figure 7 depicts the predictive analysis life cycle. The goal of the cross-industry standard process for machine learning is to provide practitioners with standards for implementing machine learning on big databases. The first step is to acquire the necessary data for modelling. The data preparation process involves data cleansing and integration. The influential features are then extracted using feature selection methods. The model is developed and deployed using one or more machine learning techniques. Afterwards, the resulting models are assessed using several performance metrics.

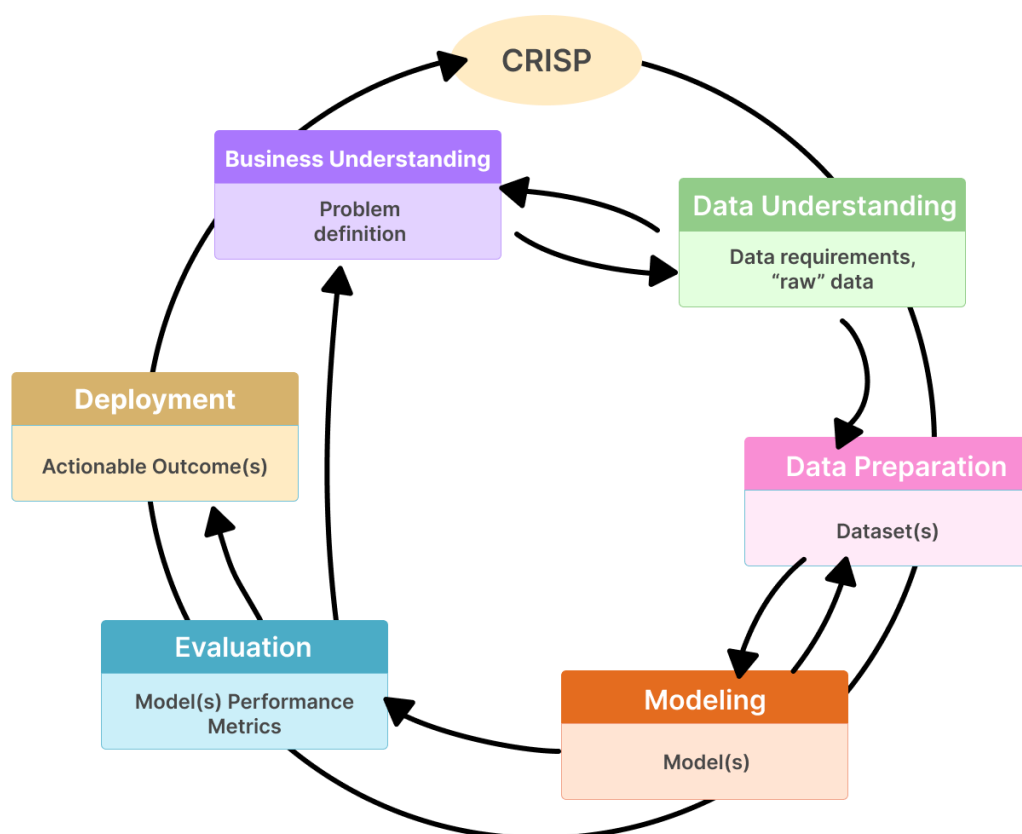


Figure 7 - Cross-industry standard process (CRISP) for machine learning (adapted from Plotnikova; Dumas; Milani, 2020).

Massive volumes of high-dimensional data are widespread in numerous sectors in the age of big data; when machine learning algorithms are applied to such

data, the curse of dimensionality arises. It refers to the phenomena in which data gets sparser in high-dimensional space, which has a negative impact on algorithms built for low-dimensional space. Moreover, the presence of high-dimensional features would greatly increase the processing and memory storage needs. Consequently, feature selection has become one of the most frequent and crucial data preparation methods, as well as a central part of machine learning (Li; Liu, 2017).

Feature selection in supervised learning is often considered as a search task in a space of feature subsets. To conduct this search, a beginning point, a technique to explore the space of subsets, an evaluation function, and a terminating condition must be defined (Asir; Appavu; Jebamalar, 2016).

Wrapper, filter, and embedded methods are the three main strategies for feature selection (Figure 8).

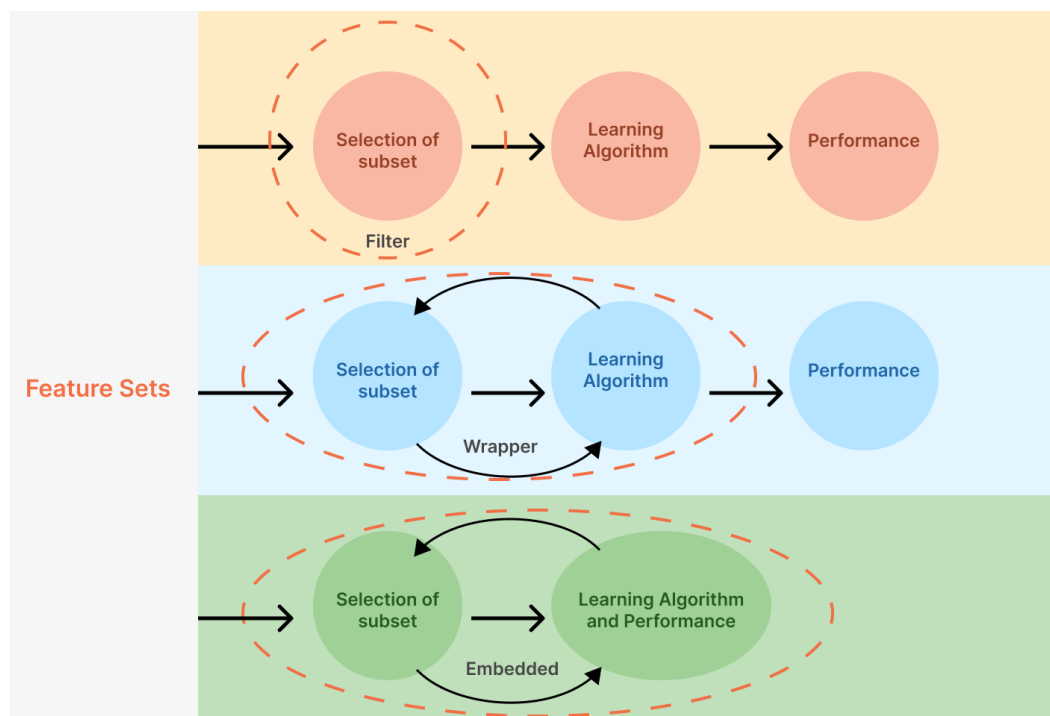


Figure 8 - Feature selection methods, such as filter, wrapper, and embedded method (adapted from Xie et al., 2020).

The filter method evaluates subsets of features using a metric that is independent of any learning technique. This method is effective and quick to calculate. However, filtering algorithms may overlook qualities that are not valuable on their own but may be quite beneficial when paired with other features (Hopf; Reifenrath, 2021).

Mutual information is a crucial notion in information theory, serving as an example of filter technique. Mutual information may be used for nonlinear transformation and high-order statistics extraction. Entropy, divergence, and mutual information are fundamental notions outlined under the theory of information. Mutual information is a measure of the amount of information one random variable has about another random variable. This term is significant in the context of feature selection because it provides a method for quantifying the importance of a subset of features in relation to the output vector  $C$  (Vergara; Estévez, 2014). Mutual information is defined formally as follows:

$$I(x; y) = \sum_{i=1}^n \sum_{j=1}^n p(x(i), y(j)) \cdot \log \left( \frac{p(x(i), y(j))}{p(x(i)) \cdot p(y(j))} \right) \quad (2)$$

where  $p(x(i), y(j))$  is the joint probability mass function and  $p(x(i))$  and  $p(y(j))$  are the marginal probabilities. When  $x$  and  $y$  are statistically independent, mutual information is zero. Therefore, mutual information is directly linked to the entropy of the variables.

Wrappers evaluate feature subsets based on the performance quality of a modelling method, which is assumed to be a black box evaluator. Similar to filters, the evaluation is repeated for each subset, and subset creation is based on the search technique. Wrappers are much slower than filters in discovering sufficiently appropriate subsets due to their dependence on the resource requirements of the modelling process. In addition, the feature subsets are skewed in favour of the modelling method on which they were tested (Jovic; Brkic; Bogunovic, 2015).

Recursive feature elimination is a typical wrapper feature selection method that aims to enhance generalisation performance by eliminating the least significant features whose deletion would have the smallest impact on training errors. In addition, there is a tight relationship between recursive feature elimination and support vector machines. Due to the fact that the weighted vector is positioned where the decision boundary has the greatest margin, if the weighted vector has a high value for a specific feature, it implies that this defining feature can clearly distinguish between classes (Jeon; Oh, 2020). The following equation (3) is used to determine the variable weight for the assessment of variable importance.

$$W_s = \frac{1}{q} \sum_{i=1}^q W_i \quad (3)$$

where  $W_i$  is the weighted vector positioned at the point where the decision border has the greatest margin, and  $q$  denotes the entire number of hyper-planes.

Embedded method is an embedded feature selection mechanism that embeds feature selection inside the learning process and leverages its attributes to drive feature assessment. On a computational level, the embedded technique is more efficient and more manageable than the wrapper method, albeit having comparable performance. This is due to the fact that the embedded technique eliminates the repetitive execution of the classifier and the analysis of each feature subset (Jovic; Brkic; Bogunovic, 2015).

Bagherzadeh et al. (2021) studied the impact of seven distinct feature selection strategies on improving the accuracy of total nitrogen prediction in the WWTP influent flow. Selecting an appropriate feature selection for collecting the most accurate input data improved the accuracy of prediction (up to 20 percent). The authors proved the excellent accuracy of mutual information feature selection model prediction. In addition, the findings revealed that the wrapper feature selection was superior to conventional filtering techniques for determining the degree of features' significance.

Anter et al. (2020) suggested a new dynamic model for fault detection in WWTPs utilising an enhanced binary whale optimization method, chaos theory, and a fuzzy algorithm. The model was created to solve the challenge of feature selection in high-dimensional nonlinear data. On a nonlinear, complex dataset with uncertainty, missing values, and a high number of features, the suggested model was verified. With a limited number of variables (10), the suggested model obtained its best level of accuracy of 0.8667, and the optimization process took 0.998s.

Zounemat-Kermani et al. (2022) assessed the ability of kriging-based and machine learning models to forecast the effluent arsenic content of a wastewater treatment facility. The authors asserted that the use of the feature selection technique not only simplified and improved the model's topology, but also increased the performance of the created models (around 7.8 percent performance enhancement of the root mean square error).

Khan et al. (2018) developed a generalised classification model for monitoring activated sludge wastewater treatment. Using sequential feature selection and least absolute shrinkage and selection operator, the authors conducted feature selection in the context of classification. The approach suggested by the authors provided strong identification of state across numerous wastewater treatment facilities and their respective states.

On the basis of time-series models and statistical process control, Sánchez-Fernández et al. (2018) offer an universal technique for fault detection. To comprehend the dynamic relations, the auto- and cross-correlations in an industrial process are characterised by a process of dynamic feature selection. With positive findings, the performance of the approach was verified at a WWTP.

## **2.4**

### **Machine learning algorithms**

Machine learning is a subfield of artificial intelligence approach that enables systems to acquire knowledge automatically without explicit programming (Figure 9). It focuses mostly on the creation of computer programs that can retrieve data and train themselves. This process begins with the examination of data and the search for patterns in order to make better choices (Alom et al., 2019).

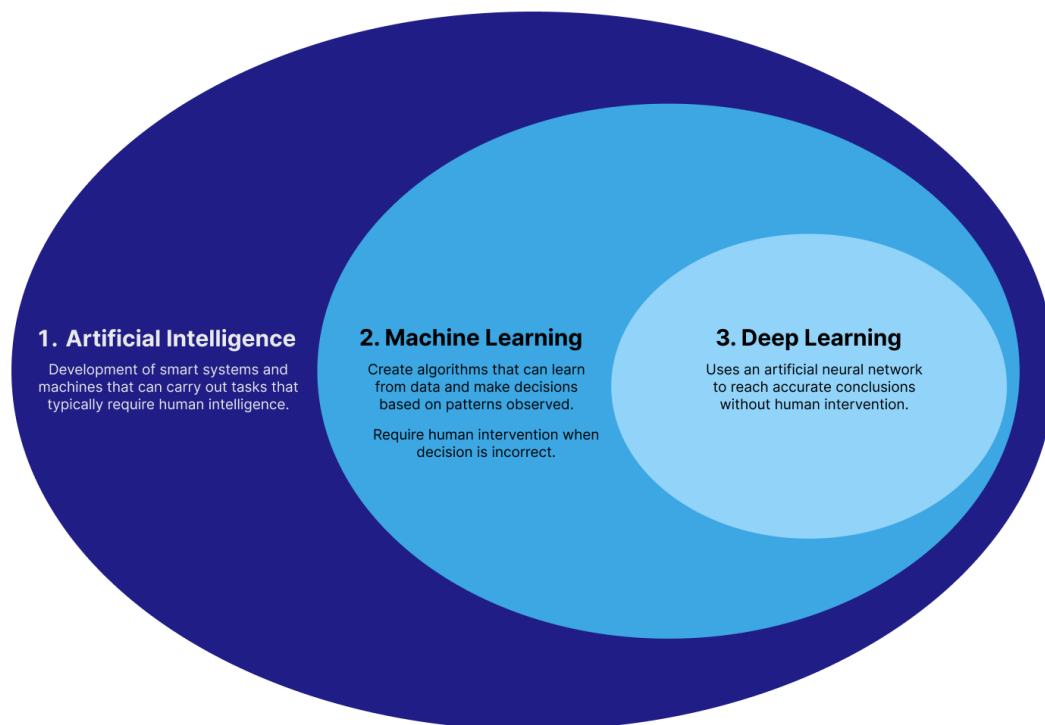


Figure 9 - Venn diagram illustrating the relationship between each subset of artificial intelligence (adapted from Vrana; Singh, 2020).

With the use of labels, supervised machine learning algorithms use the acquired information from past and current data to predict future occurrences. This strategy begins with the training of a dataset, after which machine learning develops an inferred function to predict output values. With a sufficient training procedure, the system is able to provide outcomes based on input data. The method for machine learning compares the generated results with the real and predicted outcomes to find flaws and adjust the model accordingly (Sarker, 2021).

A classic version of the supervised machine learning task is the classification problem, in which the model must estimate the behaviour of a function that maps a vector into one of many classes by seeing several input-output samples of the function (Nasteski, 2017). In the next section, the recent studies on supervised machine learning are reviewed.

#### 2.4.1 Multi-Layer Perceptron (MLP)

In 1958, psychologist Rosenblatt did pioneering research on Perceptrons. The Perceptron was an electrical device designed in line with biological principles and

capable of learning (Rosenblatt, 1958). Each Perceptron contains inputs that are weighted, an activation function, and a single output (Figure 10).

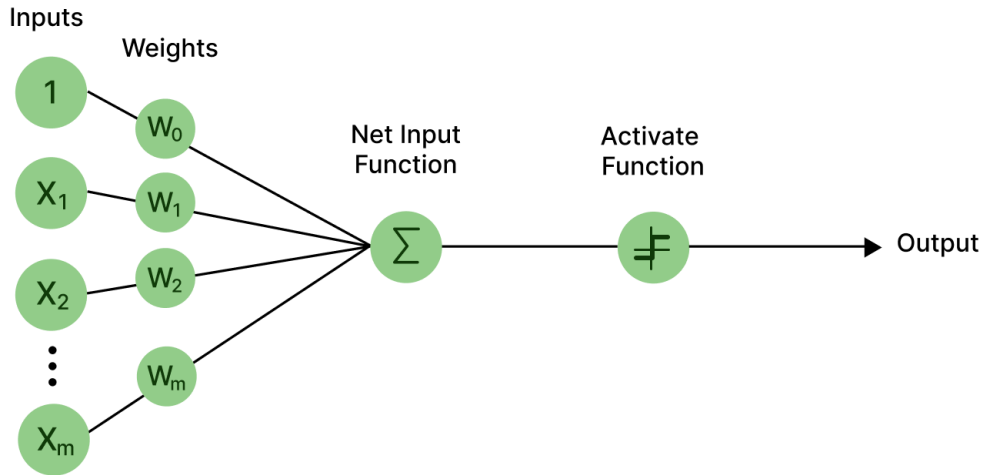


Figure 10 - Structure of a Perceptron (adapted from Rosenblatt, 1958).

A typical MLP is a fully connected network with an input layer that receives input data, an output layer that predicts the model's output, and one or more hidden layers between these two that learns data patterns (Lecun; Bengio; Hinton, 2015). Weights and bias are applied to the inputs (Equation 4) and fed into an activation function (Equation 5). MLP is a feedforward neural network in which each neuron has a differentiable activation function; this network may conduct static mapping between an input space and an output space, therefore a properly trained MLP is able to approximate continuous functions.

$$u(\mathbf{x}) = \sum_{i=1}^n w_i x_i + b_i \quad (4)$$

$$y = f(u(\mathbf{x})) \quad (5)$$

A backpropagation algorithm is commonly used to train an MLP (Leung; Haykin, 1991). The backpropagation algorithm is a generalization of the delta rule that uses gradient descent to minimize the loss function by computing the error's derivative with respect to any weight in the network and then adjusting the weight (Equation 6). The topology in MLP is crucial because a deficiency of connections can prevent the network from solving the problem, while an excess of connections can cause overfitting.

$$x_{k+1} = x_k - \eta_k g_k \quad (6)$$

where  $x_k$  represents the matrix of current weight value and bias value;  $g_k$  represents the gradient of the current function; and  $\eta_k$  indicates the learning rate.

Arismendy et al. (2020) suggested the construction of a system that predicts the behaviour of a process based on its data to aid in the management of a wastewater treatment facility, which receives discharges from manufacturing industries. Using features such as flow rate, suspended particles, and pH, the work utilised predictive analysis on chemical oxygen demand to evaluate the efficiency of the process. To build this system, a MLP with two hidden layers and each containing 22 neurons was developed. The findings indicate that the model had a mean absolute percentage error of 10.8 percent, which is consistent with its excellent performance based on historical data.

Bekkari and Zeddouri (2019) introduced a MLP with backpropagation to forecast the ten-month effluent chemical oxygen demand performance of a wastewater treatment facility. Input factors included influent variables such as pH, temperature, suspended solids, and biochemical oxygen demand. The authors discovered that multilayer perceptron is a dependable model for WWTPs in order to give potent tools for forecasting performance and a foundation for regulating the operation of the process.

Bagheri et al. (2015) used a MLP to mimic a sequencing batch reactor. ANNs were trained to forecast the concentration of phosphorus in the reactor. The features that were considered were sludge residence time, reaction time, filling time, mixed volatile suspended solids, and influent concentration. The findings indicated that the experimental and simulated concentrations were almost identical. The authors found that MLP models were more accurate than other investigated models when trained with less input data values.

#### **2.4.2 Long Short-Term Memory (LSTM)**

A recurrent neural network is one of the most effective methods for real-time contextual information. In contrast to MLP, recurrent neural network incorporates feedback connections between nodes and layers, allowing it to process input sequences of any length. The LSTM model is a robust recurrent neural system that



was created to solve the problem of exploding/vanishing gradients. The LSTM unit shown in Figure 11 consists of a cell, an input gate, an output gate, and a forget gate (Alom et al., 2019).

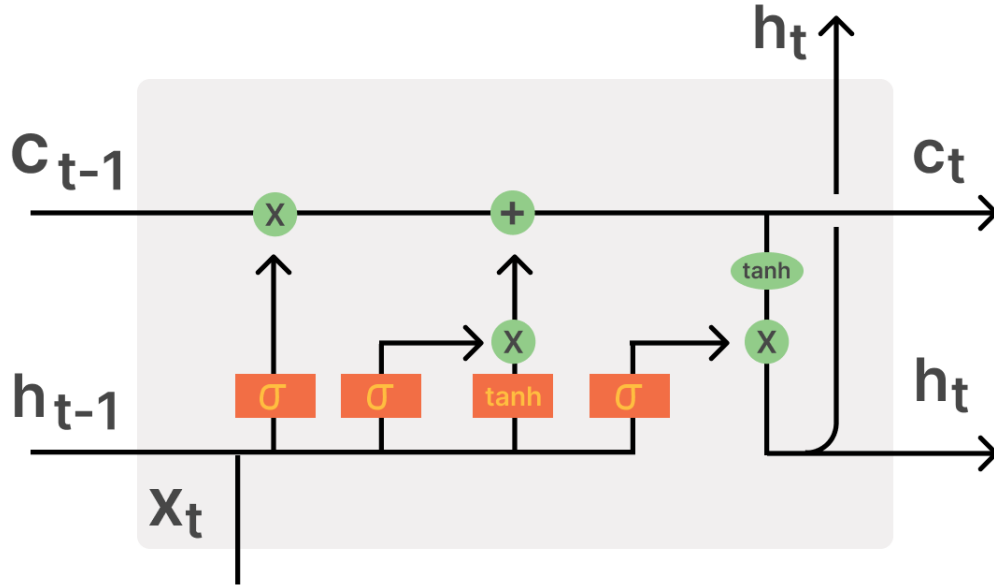


Figure 11 - LSTM unit (adapted from Sharma et al., 2021).

The cell state refers to long-term memory, previous information is kept within the cell due to their recursive nature. The forget gate is used to change the cell state, by multiplying 0 to a previous state, the forget gate outputs values indicating which information to forget. If the forget gate's output is 1, the information is stored in the cell. Which information should enter the cell state is determined by the input gate. Finally, the output gate specifies which data should be transmitted to the next hidden state (Yu et al., 2019).

The LSTM algorithm's feedforward process can be described as follows (Equation 7–9):

$$i^t = g(w_i x^t + p_i y^{t-1} + q_i \cdot c^{t-1} + b_i) \quad (7)$$

$$l^t = \sigma(w_l x^t + p_l y^{t-1} + q_l \cdot c^{t-1} + b_l) \quad (8)$$

$$f^t = \sigma(w_f x^t + p_f y^{t-1} + q_f \cdot c^{t-1} + b_f) \quad (9)$$

where  $x^t$  represents the input data for time  $t$ ;  $i^t$ ,  $l^t$ , and  $f^t$  represent the activations of the input gate, output gate, and forget gate, respectively;  $w$ ,  $p$ ,  $q$ , and  $b$  represent

the input weightings, output weightings, memory cells weightings, and bias of the corresponding activations, respectively.

Following that, the output gate value, memory cell state, and output value can be determined as follows (Equation 10–12):

$$c^t = i^t \cdot l^t + c^{t-1} \cdot f^t \quad (10)$$

$$o^t = \sigma(w_o x^t + p_o y^{t-1} + q_o \cdot c^t + b_o) \quad (11)$$

$$y^t = h(c^t) \cdot o^t \quad (12)$$

where  $c^t$  is the memory cell's state and  $o^t$  is the output gate's value;  $g$ ,  $\sigma$ , and  $h$  are the input, forget, and output gates' activation functions, respectively.

Yaqub et al. (2020) suggested a LSTM model to predict the ammonium, total nitrogen, and total phosphorus removal efficiencies of an anaerobic membrane bioreactor. As inputs, the parameters of the influent wastewater, including total organic content, chemical oxygen demand, and ORP, were assessed. When the unseen dataset was evaluated for prediction, the proposed LSTM model yielded good results, correctly tracing the nonlinear behaviour of the system in wastewater treatment. The authors stated that the model is promising for estimating the system's real-time nutrient removal efficiency and may assist in the development of process control techniques.

Cheng et al. (2020) used deep learning-based models as soft-sensors for predicting WWTP's key factors, such as influent flow, influent biochemical oxygen demand, effluent chloride, and power consumption. As a case study, actual data from a municipal wastewater treatment facility in Saudi Arabia were utilised to evaluate the suggested forecasting algorithms. The suggested models produce good predicting results without requiring any data distribution assumptions. In terms of precision, the LSTM soft-sensor achieves the best result.

Hwangbo et al. (2021) demonstrated the use of deep learning to quantify large-scale, long-term wastewater treatment facility data for process modelling of nitrous oxide concentrations. Under the prediction horizon, the last day of the whole time series data was specified, and the performance of the forecasting model was assessed based on an increase in the quantity of accessible historical data preceding the prediction horizon. As inputs, around 750,000 measures were employed,

including influent flow rate, air flow rate, temperature, ammonium, nitrate, and dissolved oxygen. This study selected the LSTM to conduct modelling for nitrous oxide emission predictions and found that the model outperforms the other methods evaluated.

### 2.4.3 Support Vector Machine (SVM)

The notion of the SVM was inherited from the ANN, or one might argue that the SVM is the mathematical extension of the ANN. SVM conducts classification by translating the initial training data into a multidimensional space and generating a hyperplane with greater dimensions. SVM is an effective mathematical learning strategy based on hyperplanes. The algorithm searches for those vector points, referred to as support vectors, which define the decision boundary and provide a significant marginal separation across classes (Figure 12). In the decision plane, SVM distinguishes classes with the greatest marginal distance (Wang, 2005).

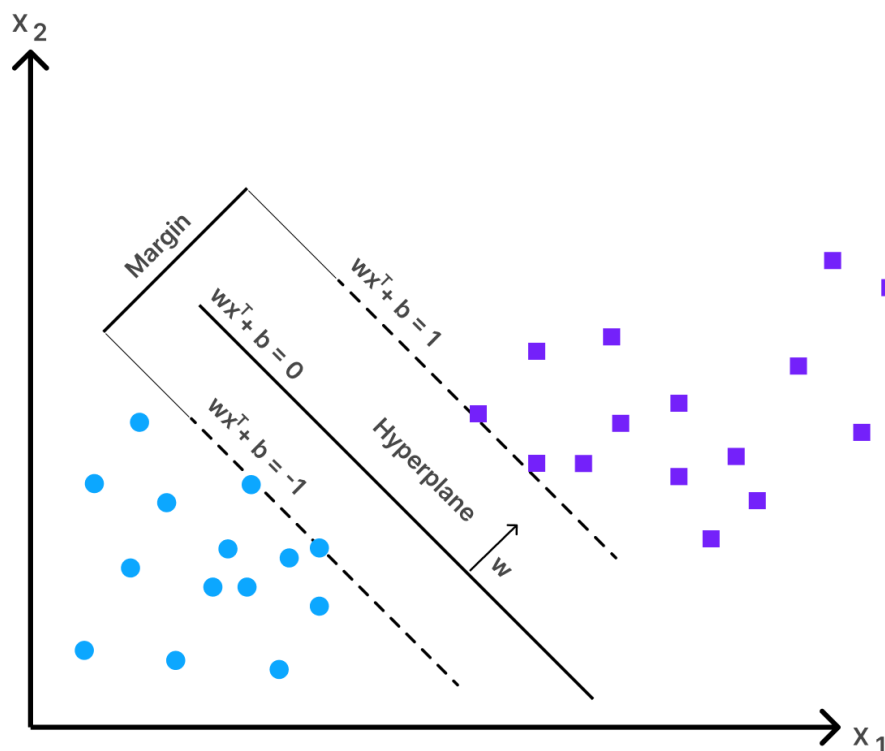


Figure 12 - Linear SVM model (adapted from Huang et al., 2018).

The separation (margin) between the decision borders is maximized in a highly dimensional space called the feature space by determining decision functions directly from the training data using SVM. This classification approach reduces the

training data's classification inaccuracies and improves generalization ability (Chauhan; Dahiya; Sharma, 2019).

The following is the overall equation for the additional hyperplane (Equation 13):

$$y_i(w \cdot x_i - b) \geq 1, \forall 1 \leq i \leq n \quad (13)$$

where  $w$  stands for the normal vector,  $b$  for the bias,  $\cdot$  for the dot product, and  $x_i$  stands for the dimensional vector that has to be categorized into  $y_i$ .

With a linear method in an adequate feature space, the employment of kernel functions gives a strong and logical approach to detecting nonlinear relations. Where kernel is a function with the following definition (Equation 14):

$$K(x_i, x_j) = [\varphi(x_i), \varphi(x_j)] \quad (14)$$

where  $\varphi$  is the kernel function that converts input space to feature space.

In an SVM model, two types of parameters must be optimized: the penalty factor  $C$  ( $C > 0$ ) and the kernel function parameters, which might be linear, polynomial, or radial basis functions.  $C$  is a fixed and adjustable parameter that determines the severity of the punishment in the event of incorrect samples (Yang; Li; Yang, 2015).

SVM is gaining popularity because it has a solid mathematical basis and seems to perform well in several varied real-world applications (Chauhan; Dahiya; Sharma, 2019).

Guo et al. (2015) built different machine learning models for estimating the effluent total nitrogen content at a Korean wastewater treatment facility. Utilizing daily data on water quality and meteorological data, the performance of models was assessed using the coefficient of determination and other metrics. In the training phase, the SVM model exhibited a greater accuracy. This research revealed that these models might be a viable way for predicting water quality as an early warning system for WWTP water quality management.

To forecast the daily flow rates for the wastewater treatment facility, Najafzadeh and Zeinolabedini (2019) developed a number of soft computing

techniques, including SVM. Flow rates datasets have been utilised over a five-year period to construct artificial intelligence algorithms. Statistical error indicators were used to analyse the performance of suggested models throughout the training and testing phases. In comparison to other machine learning approaches, SVM demonstrated the highest level of efficiency during the training stages, and testing stages demonstrated that it accurately predicted the flow rates parameter with an acceptable degree of accuracy.

Nourani et al. (2018) used three distinct nonlinear models based on artificial intelligence to forecast the performance of a wastewater treatment facility in terms of effluent biological oxygen demand, chemical oxygen demand, and total nitrogen. The findings indicate that SVM is more robust than the other approaches. The results also revealed that the uncertainty involved in the process may be addressed for the deployment of these models in the real world.

#### 2.4.4 Extreme Gradient Boosting (XGBoost)

Gradient boosting is a powerful tool in the field of supervised learning, allowing for cutting-edge classification performance. At its foundation, XGBoost is a decision tree boosting algorithm. Boosting is an ensemble learning strategy that involves developing many models in a sequential order, with each new model aiming to correct for flaws in the prior model. Each additional model added to the ensemble is a decision tree in tree boosting (Figure 13). A gradient descent technique is used to minimize loss in this type of boosting procedure (Ferreira; Figueiredo, 2012).

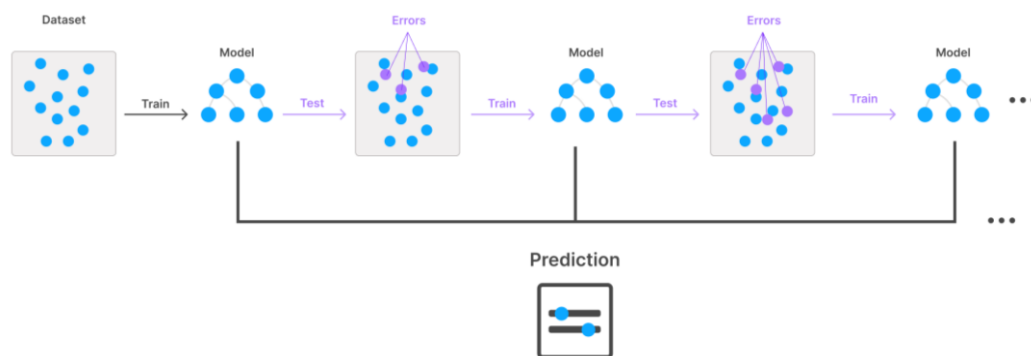


Figure 13 - XGBoost's architecture layout (adapted from Kiangala; Wang, 2021).

The algorithm adds up all of the  $K$  tree results to get the final predicted value,  $\hat{y}_l$ , which is represented as (Equation 15–16):

$$\hat{y}_l = \sum_{k=1}^K f_k(x_l), \quad f_k \in F \quad (15)$$

$$F = \{f(x) = w_{q(x)}\}, \quad \left(q: R^m \rightarrow T, w \in R^T\right) \quad (16)$$

where  $F$  stands for the set of decision trees,  $m$  stands for the number of features,  $f(x)$  stands for one of the trees, and  $w_{q(x)}$  stands for the leaf node weight. The number of leaf nodes is represented by  $T$ , and the structure of each tree is represented by  $q$ , which maps the sample to the corresponding leaf node.

XGBoost's predicted value is the sum of the values of each tree's leaf nodes. The model's aim is to learn these  $k$  trees so that the following objective function can be minimized (Equation 17–18):

$$L^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_l) + \sum_{k=1}^K \Omega(f_k) \quad (17)$$

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \|\omega\|^2 \quad (18)$$

where  $l$  is the loss related to the difference between the estimated and true values. The penalty of the decision tree is adjusted using  $\Omega$  regularization, which can prevent overfitting.  $\gamma$  is a hyperparameter that determines the model's complexity, and  $T$  is the number of leaf nodes. The penalty coefficient for the leaf weight  $\omega$ , which is normally constant, is  $\lambda$  (Tanha et al., 2020).

XGBoost scalability is the result of numerous significant system and algorithmic enhancements. Among these advances are an unique tree learning technique for dealing with sparse data and a weighted quantile sketch process for dealing with instance weights in approximation tree learning. Parallel and distributed computing accelerates learning, hence accelerating model exploration (Chen; Guestrin, 2016).

XGBoost may be considered to explicitly evaluate the bias-variance tradeoff during fitting. In order to prevent needlessly raising variance, neighbourhoods are maintained as big as feasible and are only made narrower when sophisticated structure becomes evident. Using smaller neighbourhoods in these locations may

significantly decrease bias while adding just a tiny amount of variation (Ferreira; Figueiredo, 2012).

Lu and Ma (2020) introduced decision tree-based machine learning models to improve the short-term accuracy of water quality prediction. Six water quality indicators, including water temperature, dissolved oxygen, pH, specific conductance, turbidity, and dissolved organic matter, were predicted using the models. As a foundation for performance assessment, six error measures were proposed, and the outcomes of the models were compared. The absolute mean percentage error of XGBoost was 4.60 percent on average, providing the most accurate predictions.

Buras and Solano Donado (2021) offered a solution to the challenge of detecting a wastewater pollutant and locating its source point in the wastewater network, given a time-series of wastewater measurements gathered by sensors in the sewage network. The algorithm XGBoost was used to estimate the distance between the source and the sensor. Using simulated electrical conductivity and pH measurements of wastewater in European city sewers, the models were trained. The findings were quite precise, with an average precision and recall of 96%.

Jeong et al. (2021) assessed the XGBoost model's ability to forecast the micropollutant removal efficiencies of reverse osmosis and nanofiltration membranes. The authors demonstrated that XGBoost has a sufficient comprehension of size exclusion, but insufficient understanding of electrostatic interactions and adsorption. The results recommended that future research should concentrate more on preventing data leaks and assessing the mechanistic understanding of machine learning models.

### 3 Electrocoagulation

Before beginning the literature review of the most recent publications that have contributed to the modelling of the EC process, it is important to examine certain fundamental principles inherent to this method.

EC has a long history; the first sewage treatment facility was constructed in London in 1889 (Vik et al., 1984). Despite some encouraging findings, the viability of this method was restricted because to the comparatively high cost of investment and electricity compared to chemical coagulation. In recent years, however, there has been a resurgence of scientific, commercial, and environmental interest in this technique, mostly owing to the desire for alternate water treatment solutions. Small-scale EC has established a place in the water treatment sector, proving to be a dependable and successful technique, but its full potential cannot be realised without a deeper technical knowledge (Holt et al., 2002). Typically, empirical investigations are conducted to determine the primary EC operating parameters. The technique has been designed to decrease energy usage and maximise rates of pollution removal (Mollah et al., 2001).

EC reactors are electrochemical cells consisting of an array of electrodes in contact with polluted water, with *in situ* coagulant generation serving as their defining characteristic. To release the coagulant, a voltage difference must be applied between the electrodes. The needed electrode potentials may be derived from the half-cell electrochemical reactions that occur at each electrode, which vary based on the operating pH and the species present in the system (Holt et al., 2002).

EC is capable of removing a diverse array of contaminants, including suspended particles, trace elements, dyes, organic compounds, ions, oils, and greases. The pollutant's physicochemical properties affect its interactions in the EC reactor and, therefore, its removal path. Ions, for instance, are likely to be electroprecipitated, while suspended solids are likely to be adsorbed to flocs (Mollah et al., 2004).



EC is characterised by three key mechanisms: (1) formation of coagulants by electrolytic oxidation of the anode, (2) destabilisation of pollutants, particle suspension and emulsion breakdown, and (3) aggregation of the destabilised phases to form a floc (Vasudevan et al., 2010). In addition, the following physicochemical reactions (Mollah et al., 2004) might occur during EC:

1. Cathodic reduction of contaminants in the solution;
2. Electrophoretic migration of ions in solution;
3. Electroflotation of flocs by O<sub>2</sub> and H<sub>2</sub> bubbles generated at the electrodes;
4. Other processes (eg chemical dissolution).

EC is thus a complex process involving several physicochemical phenomena. The majority of research on EC has used an empirical methodology (Chen; Chen; Yue, 2002). In other words, they identified the pollutant removal capabilities of this technology (within a certain set of operating conditions and reactor specifications), but were unable to isolate (and quantify) the primary processes causing pollutant removal. Figure 14 depicts the interconnected and intricate nature of the EC process (Holt et al., 2002). Obviously, the coagulant and its hydrolysis products may have several interactions with the pollutant, other ionic species, and gas bubbles.

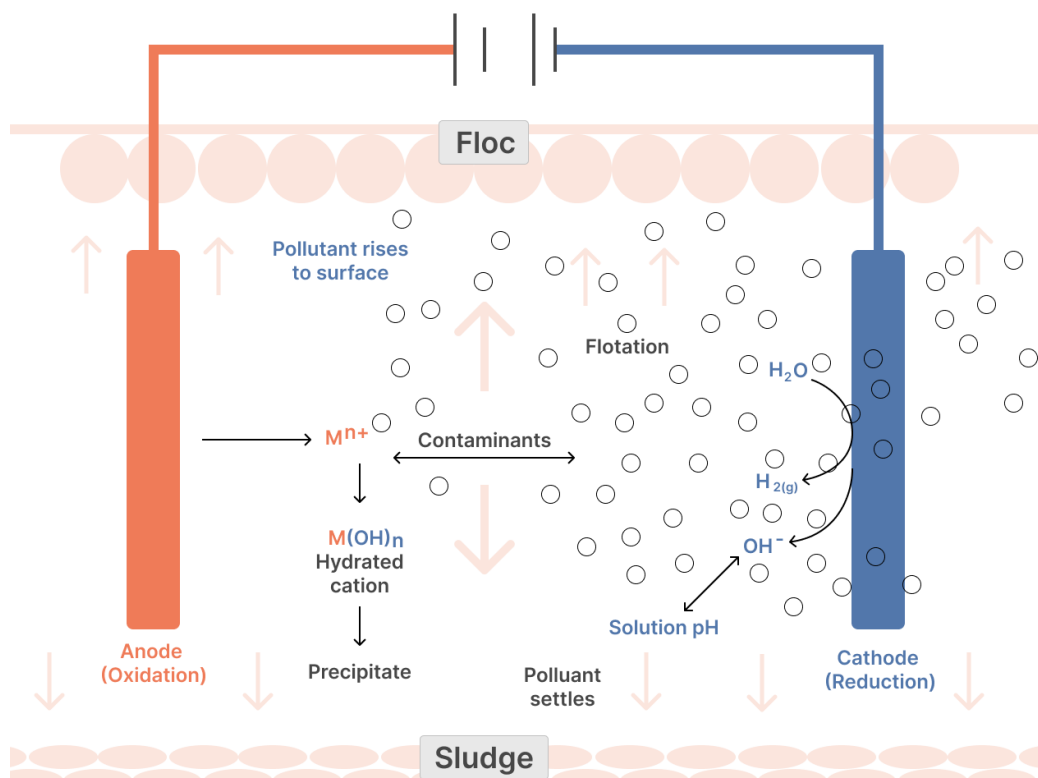


Figure 14 - Interconnected and intricate nature of the EC process (adapted from Moneer; El Nemr, 2012).

It is feasible to distinguish three fundamental fields of study that interact in the EC process: electrochemistry, coagulation, and flotation. These domains of knowledge may be represented by a Venn diagram, where the intersection of the three disciplines forms EC (Figure 15).

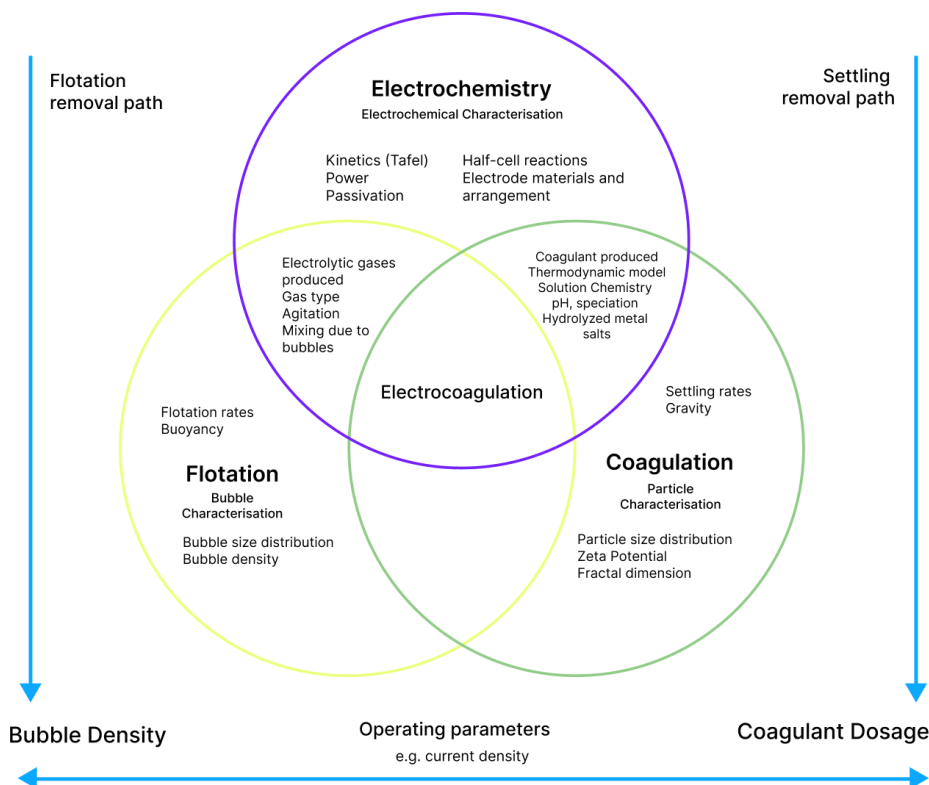


Figure 15 - EC's domains of knowledge (adapted from Bhagawati et al., 2022).

The electrochemical processes that occur in the EC cell define, among other things, the coagulant cation and the properties of bubbles. Electrochemistry is hence one of the cornerstones of EC (Moreno et al., 2009).

### 3.1 Fundamentals of EC

Electrochemistry is the study of electron transport at the solution-electrode interface. In an electrochemical experiment, the electrode behaves as either an electron source or collector. As a consequence of the separation of charges between the electrode and the electrolyte solution, a potential difference arises across the interface. This results in ions in the immediate region of the electrode surface reorienting themselves in an attempt to reach the most energetically stable configuration and preserve electroneutrality. Consequently, ions with opposing charges are attracted to the electrode surface, while ions with the same charge are repelled (Bockris; Reddy, 2000).

Faradaic processes are electron transfer reactions between an electrode and electroactive substances in solution. Faraday's law governs such chemical reactions (Xiao et al., 1996). Faraday's law says that the quantity ( $m$ ) of a substance consumed or generated at one of the electrodes of an electrochemical cell is proportional to the electrical charge ( $q$ ) passing through the electrode (Equation 19).

$$m = \frac{q}{nF} = \frac{It}{nF} \quad (19)$$

where  $n$  and  $F$  are the number of moles of electrons in the electrode reaction and Faraday's constant ( $96485 \text{ C mol}^{-1}$ ) respectively,  $m$  is the quantity of a substance consumed or generated in mol, and  $q$  is the electrical charge in coulombs. Current intensity,  $I$ , in amperes, multiplied by time,  $t$ , in seconds, is equal to electric charge.

There may be other processes, such as adsorption and desorption. These are known as non-faradaic processes (Rubinstein, 1995).

Although Faraday's work established the relationship between electrical charge and the amount of species generated or consumed during an electrochemical reaction, until the advent of thermodynamics, the relationship between the equilibrium potential of an electrochemical reaction and the activities of the electroactive species (participants) remained obscure (Perez, 2004). Nernst proposed an equation to compute the equilibrium potential of an electrochemical reaction based on thermodynamic laws (Xiao et al., 1996).

Electrochemical reactions are heterogeneous chemical processes involving electron transport. According to Equation 20, the Gibbs free energy of an electrochemical reaction determines the maximum electrical work ( $W_{ele}$ ) at constant temperature and pressure (Atkins; De Paula, 2013).

$$W_{ele} = -\Delta G = nFE_{eq} \quad (20)$$

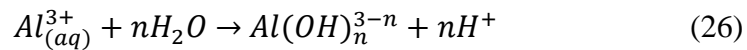
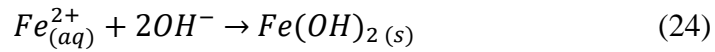
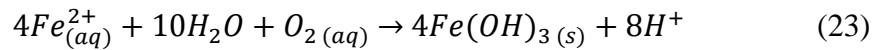
where  $n$  is the number of moles of participating electrons,  $F$  is the Faraday constant,  $E_{eq}$  is the equilibrium potential of the electrochemical reaction in volts,  $\Delta G$  is the Gibbs free energy of the electrochemical reaction in joule  $\text{mol}^{-1}$ , and  $W_{ele}$  is the maximum electrical work in joule  $\text{mol}^{-1}$ .

Consequently, the Nernst equation (Equation 21) is used to determine the equilibrium potential ( $E_{eq}$ ) for any half-cell reaction (Xiao et al., 1996).

$$E_{eq} = -\frac{\Delta G^0}{nF} - \frac{RT}{nF} \sum v_i \ln c_i \quad (21)$$

where  $n$  is the number of moles of electrons involved in the reaction,  $F$  is the Faraday constant,  $R$  is the universal gas constant,  $T$  is the absolute temperature,  $\Delta G^0$  is the change in the standard Gibbs free energy in joule mol<sup>-1</sup>,  $v$  is the stoichiometric coefficient of species  $i$ ,  $c$  is the concentration of species  $i$  and  $E_{eq}$  is the equilibrium potential in volts.

Equations 22–26 express the anodic reactions for both aluminium and iron electrodes.



As indicated by the iron E-pH diagram (Figure 16), iron may produce divalent or trivalent cations depending on the solution's pH and potential. In contrast, aluminium dissolves solely as trivalent cations (Figure 17). Using the iron and aluminium E-pH diagrams, it was possible to predict the stable compounds of each element under varied circumstances.

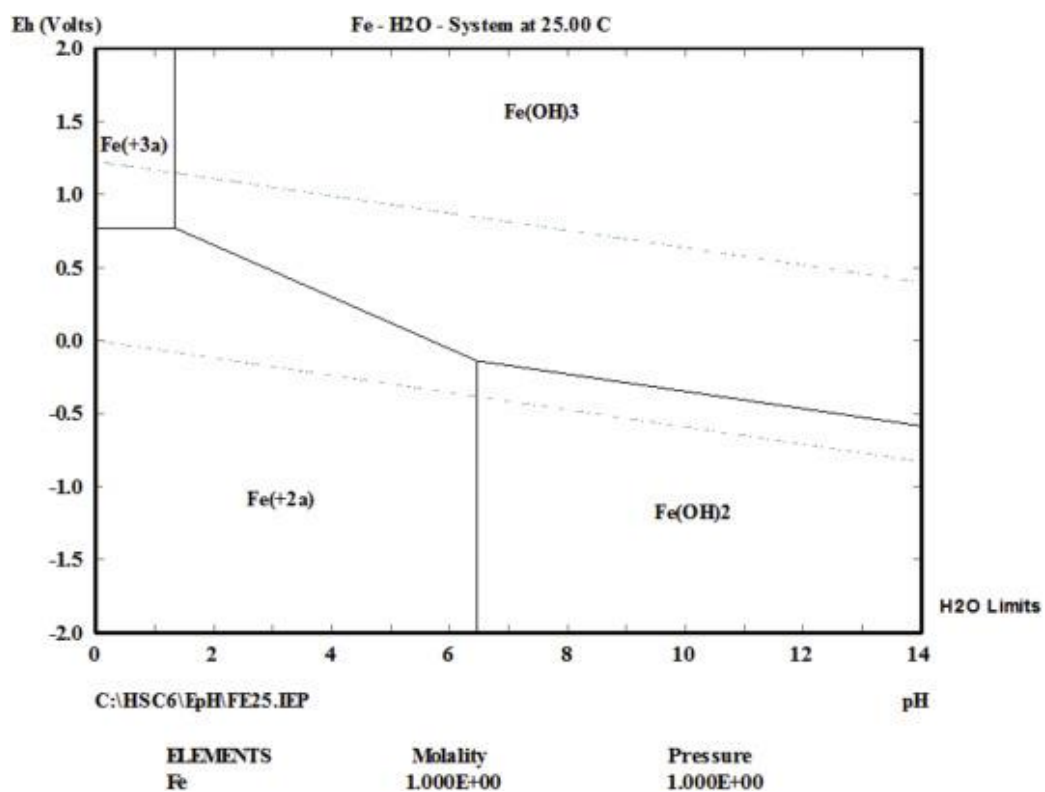


Figure 16 - E-pH diagram of iron at 25 °C (generated using the HSC chemistry software).

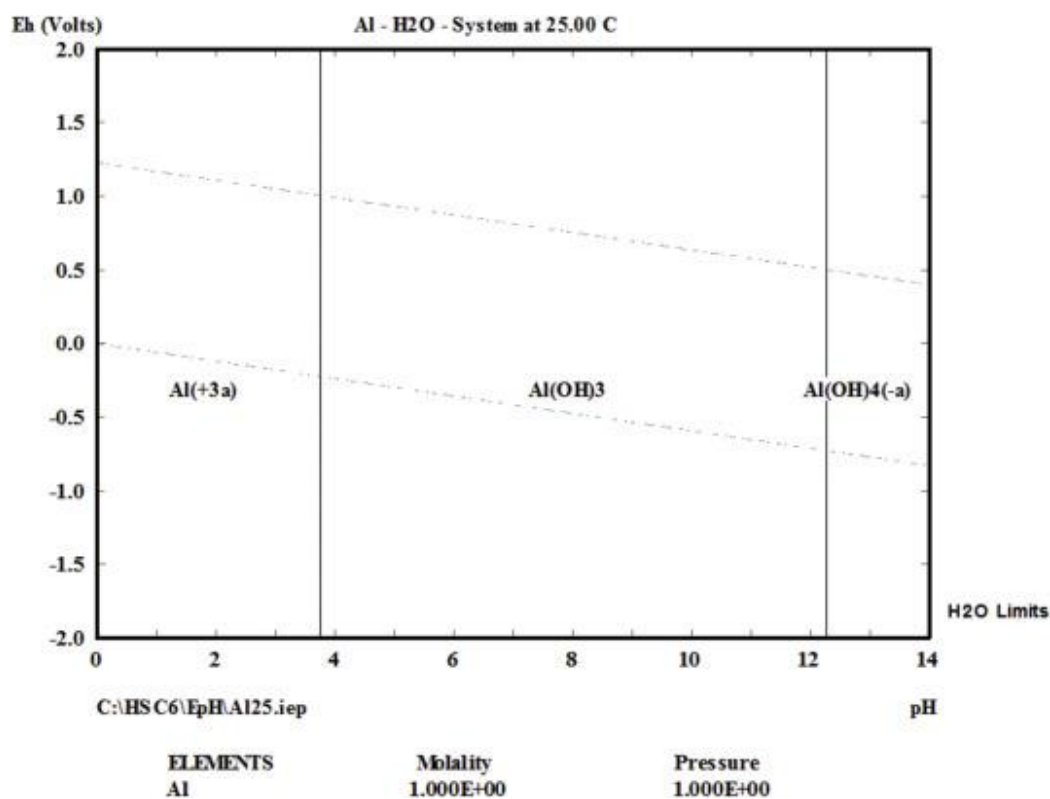
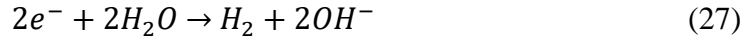


Figure 17 - E-pH diagram of aluminium at 25 °C (generated using the HSC chemistry software).

As indicated before, various side reactions occur in the EC cell, including the formation of hydrogen bubbles and  $\text{OH}^-$  ions at the cathode, which result in a rise in the pH of the solution, as shown in Equation 27.



Note that thermodynamics describes the circumstances under which diverse species are stable, but does not specify the pace at which equilibrium is reached. Therefore, to forecast the performance of any reactor, it is necessary to quantify the reaction kinetics (Levenspiel, 1999). The rate of coagulant addition in an EC reactor is governed by the kinetics of the electrode.

Electrode reactions are heterogeneous and take place at the electrode's interface with the solution. Potential and concentration gradients exist between the electrode surface and the solution's bulk. Before the transfer of electrons may occur at the surface, the chemical species to be reduced or oxidised must diffuse from the bulk of the solution to the surface of the electrode (by a process of mass transfer). Therefore, electrochemical processes are governed by electron or mass transfer (Rubinstein, 1995).

At equilibrium, the electrochemical reaction proceeds in both the oxidation and reduction directions at the same rate. An equilibrium potential,  $E_{\text{eq}}$ , is established. If this potential is changed in any way, the electrode is said to have been polarised. The overpotential is the polarisation level. When the net current across an electrode is not zero, there is a deviation from the equilibrium potential of the electrodes. This fluctuation denotes an overpotential whose value is dependent on the electrolyte's resistance, activation energy, and concentration (Newman; Thomas-Alyea, 2012). Overpotential is sometimes seen as the sum of the components in Equation 28.

$$\eta = \eta_{ohm} + \eta_{acti} + \eta_{conc} \quad (28)$$

$\eta_{ohm}$  represents the ohmic polarisation in volts,  $\eta_{acti}$  represents the activation overpotential in volts, and  $\eta_{conc}$  represents the concentration overpotential in volts.

Coagulation, which describes the interaction between the coagulant and any pollutant, is a basic process of all EC reactors. The function of the coagulant is to destabilize the colloidal suspension, therefore lowering attractive forces and permitting particle aggregation. Depending on the physicochemical properties of the solution, pollutant, and coagulant, many coagulation processes have been proposed (e.g., neutralization of surface charges, compression of the electrical double layer, bridging, and sweeping) (Lindström, 1989). The primary coagulation mechanism for each EC reactor depends on the operating settings of the reactor, the type and concentration of the pollutant, and the concentration of the coagulant.

Pollutants are generally colloidal particles, which, because of their electrostatic stability, are not readily removed by sedimentation or flotation. Due to their small size and huge surface area, these particles possess certain features. In EC, a pollutant's physicochemical properties influence its stability. Similar surface charges on pollutants repel each other. These attractive forces produce a stable colloidal system (Everett, 2007). The particle's surface net charge influences the distribution of ions in its vicinity, raising the concentration of counter-ions close to the surface. Consequently, an electrical double layer forms at the particle-solution interface as shown in Figure 18 (Hunter, 1993).



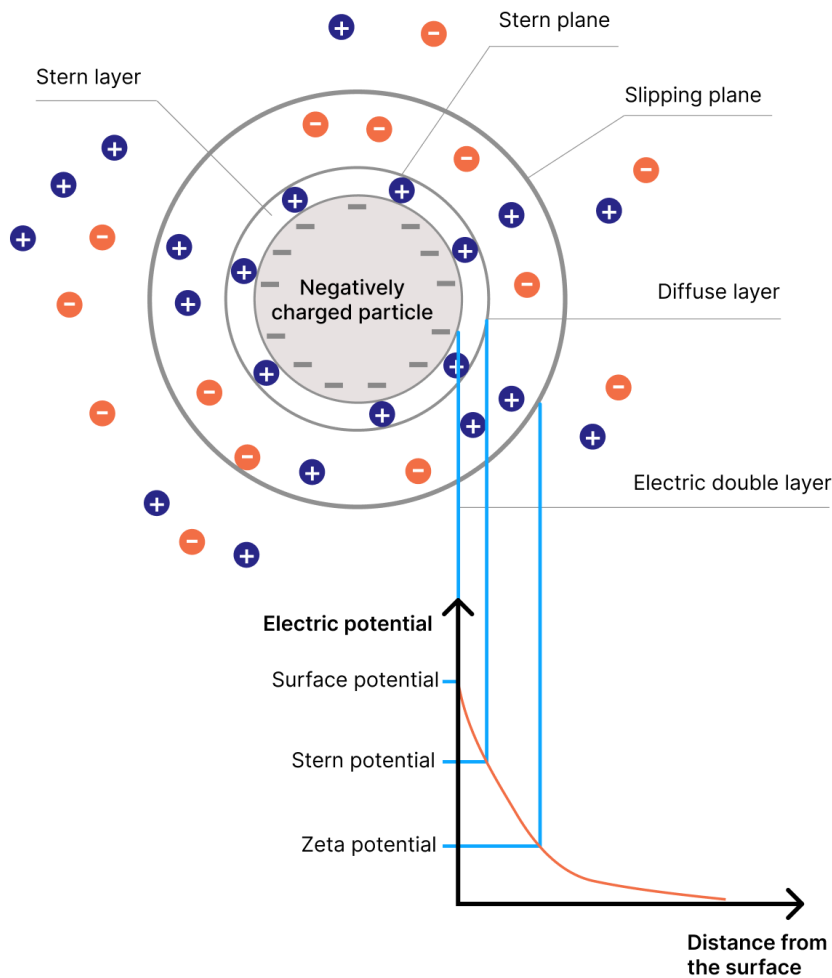


Figure 18 - Electrical double layer diagram (adapted from Tahreen; Jami; Ali, 2020).

When the interaction is entirely electrostatic, the rise in ionic strength of the solution causes a compression of the electrical double layer and, subsequently, a reduction in the repulsion energy between the particles. This mechanism of destabilization happens in the presence of electrolytes that are indifferent, since the counter-ion does not adsorb on the particle's surface, but rather raises the ionic strength of the system (Bratby, 2016).

Under optimal circumstances of coagulant concentration and pH, metallic coagulants may react with accessible hydroxides in the solution to precipitate metallic hydroxides. These metal hydroxides produced by the coagulation process are amorphous precipitates, since crystallization occurs extremely slowly (Duan; Gregory, 2003). In this instance, metal hydroxides are extremely insoluble and they precipitate in a polymerized state, generating molecules with a high molecular

weight. When insoluble compounds precipitate, they entrap particles that settle together (Bratby, 2016). This process is known as sweep flocculation.

Thus, electrochemistry, coagulation, and flotation are the three foundations of EC. Each component is a well-understood technology in its own field of expertise. Nonetheless, it is evident that there is a need to comprehend how these technologies interact in an EC reactor, as will be detailed in the section on EC reactor design that follows.

### **3.2 EC reactor design**

The design of the reactor is the focal point of an electrochemical water treatment process; the performance of the reactor has a direct impact on the operation and cost of the process, since it influences many of the other units, such as settlers and filters (Al-Raad; Hanafiah, 2021). Therefore, it is necessary to analyze this aspect if the objective is to develop systems that operate in continuous flow, paying close attention to the development of an appropriate geometrical design, the selection of the electrode material, and the selection of auxiliary equipment such as pumps and settling tanks, among others. It is also important to limit the potential drop between the electrodes in order to increase conversion efficiency and decrease energy consumption throughout the operation. During electrolytic procedures, it should also be considered that effluent and different solid species alter the hydrodynamic conditions of the reactor (Lu; Zhang; Li, 2021).

Depending on the position of the electrode plates, the flow in the reactors may be horizontal or vertical (Figure 19), and it can be split into many channels or a single channel. Vertical electrodes are more often employed in EC reactors than horizontal electrodes because the electrode plates may be readily changed (López-Guzmán; Flores-Hidalgo; Reynoso-Cuevas, 2021). In particular, upflow mode is often chosen in order to create more metal hydroxides and give a longer flow residence time. In addition, according to the literature (Lu; Zhang; Li, 2021), perforated electrode plates have been created and employed to induce the treated solution to flow along a circuitous route and boost the effectiveness of water mixing.

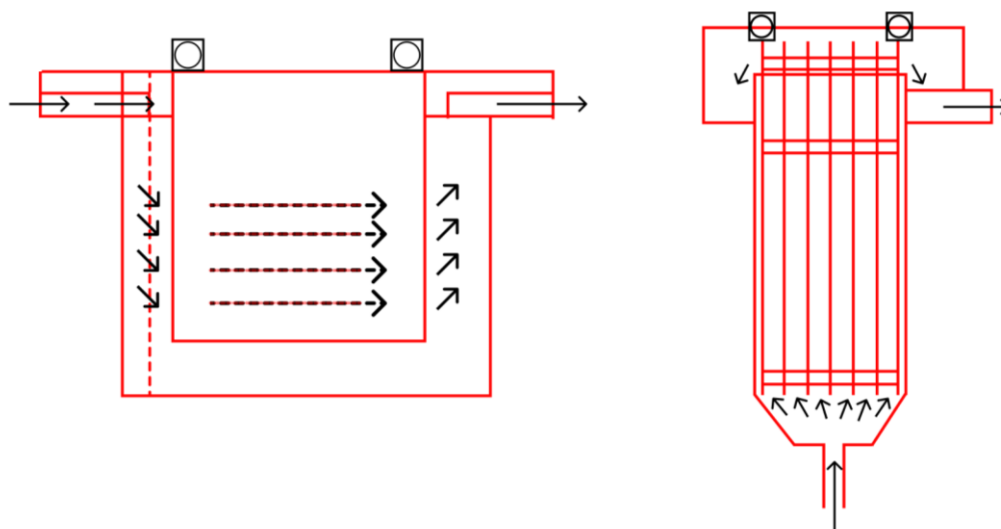


Figure 19 - Orientation of wastewater flow (view from the side): (a) horizontal flow; (b) vertical flow (adapted from Lu; Zhang; Li, 2021).

The electrode material has a significant impact on the EC's effectiveness, and cost. Al and Fe electrodes have been extensively used for EC with successful removal of pollutants due to their aptitude for efficient coagulation, cost-effectiveness, availability, dependability, and non-hazardous attributes (Garcia-Segura et al., 2017).

In an EC treatment, the distance between electrodes is crucial, since it controls the electrostatic field between the anode and cathode. The electrostatic field is greatest when the distance between electrodes is shortest. Therefore, the metal hydroxides that assist coagulation by aiding in the formation of flocs deteriorate as a result of the intense collisions caused by the strong electrostatic attraction (Bazrafshan et al., 2015). Consequently, EC efficiency is poor at the closest possible distance between electrodes. In contrast, a wider inter-electrode gap delays the development of metal hydroxide-flocs as a result of diminished electrostatic forces. Electrode spacing beyond the optimal drastically affects EC efficiency, necessitating an increase in power consumption to compensate for the slower transit of released ions between the anode and cathode (Shahedi et al., 2020). Consequently, it is essential to conduct EC with the optimal interelectrode spacing.

The way of connecting the electrodes in the EC cell influences not only the removal efficiency, but also the energy consumption and cost. Monopolar electrodes in parallel connections, monopolar electrodes in serial connections, and bipolar electrodes in serial connections are the most common configurations

(Emamjomeh; Sivakumar, 2009). These layouts of electrodes connections are seen in Figure 20.

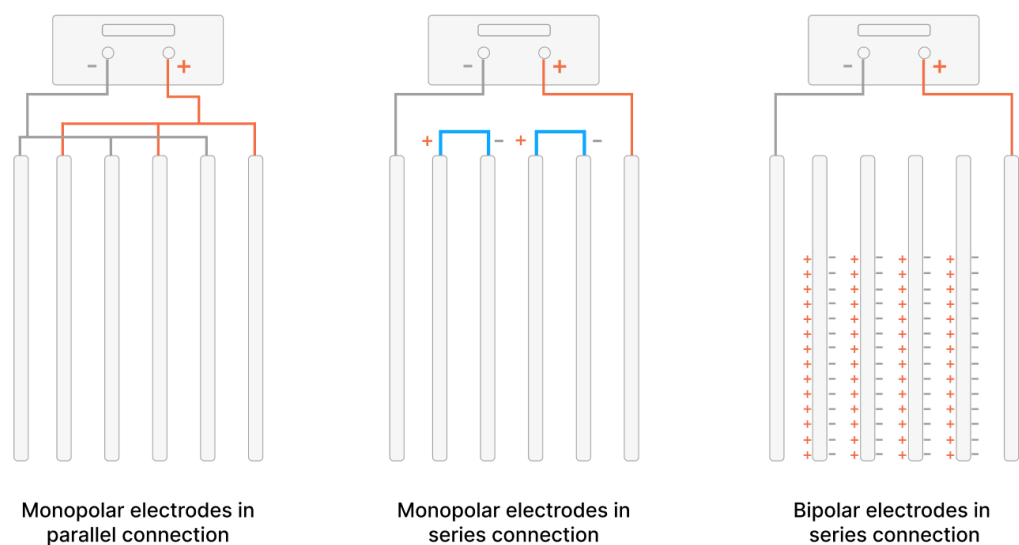


Figure 20 - Electrodes arrangements in EC cells (adapted from Garcia-Segura et al., 2017).

In an electrochemical cell with monopolar electrodes, each electrode functions as an anode or cathode depending on its electrical polarity. In monopolar electrodes with parallel connections, each sacrificial anode is directly linked to the other anode in the cell, while the cathodes are similarly connected. In the design of monopolar electrodes with serial connections, each anode-cathode pair is internally connected, but is not linked to the outside electrodes. In the case of bipolar electrodes, each electrode, with the exception of the exterior ones, which are monopolar, has a distinct polarity on either side depending on the charge of the electrode in front of it (Moussa et al., 2017).

It is important to note that when a serial configuration is utilised, bigger potential differences are necessary, yet the same current is spread across all electrodes. In contrast, in parallel mode, the electric current is shared across the linked electrodes in the electrochemical reactor based on their resistance. However, when parallel configurations are used, significant energy consumption benefits are realised (Gheraout; Alghamdi; Gheraout, 2019).

Recent developments in EC reveal that the removal rate is dependent on operational factors such as the current density, pH, conductivity of the water to be treated, and treatment time. The next section will outline how these key operating factors impact EC.

### 3.3 Main factors influencing the EC process

pH is one of the essential EC factors for the removal of pollutants. Depending on the type of the pollutants, the influence of initial pH on EC processes differs. The sacrificial electrodes may undergo chemical dissolution in alkaline and acidic liquids. Particularly thermodynamically unstable, aluminium reduces water in hydroxide and alkaline solutions. The pH also influences the speciation of the coagulant in solution, the surface charge of precipitates, and the complicated interactions with pollutants, such as co-precipitation and adsorption processes (Shahedi et al., 2020). Thus, the influence of pH on EC processes is dependent upon the type of the anode, the composition of the wastewater being treated, and the pollutants of interest.

The investigation of the link between pH and removal efficiency is complicated because pH might fluctuate during EC. The influence of pH is further confounded by the pH change in the EC cell, since the pH at the electrode surfaces may vary substantially from the pH of the bulk solution (Zaied et al., 2020).

Current density is often used to indicate the current intensity, which is one of the most important design factors for EC processes. In the EC technique, the current density is the most important element in determining the reaction rate, as it impacts the coagulant dose and bubble production rate, floc size and growth rate, bubble size, reaction kinetics, and energy consumption (Mousazadeh et al., 2021). Both the cathodic and anodic reaction rates are governed by the current density. The explanation of the link between current density and dissolved metal based on Faraday's law is crucial (Lu; Zhang; Li, 2021).

The applied current density and voltage are characteristics that are interrelated. In an electrochemical cell, a rise in voltage will result in an increase under current, and vice versa; however, in situations such as passivation of the electrodes, significant overpotentials will reduce the current. The applied current density, which determines the rate of coagulant dosage and, therefore, contaminant elimination, may influence cell voltage through multiple overpotentials (Mollah et al., 2004).

At the same applied voltage, the increase in ionic strength of the solution induces an increase in the current density. Therefore, it is vital to evaluate the influence of the solution's conductivity on the removal of contaminants. The solution's conductivity is determined by the type and concentration of the electrolyte. There are a variety of accessible electrolytes, including NaCl, BaCl<sub>2</sub>, KCl, Na<sub>2</sub>SO<sub>4</sub> and KI (Sahu; Mazumdar; Chaudhari, 2014).

Sodium chloride (NaCl) is often added to boost the solution's conductivity. The conductivity of the EC reactor impacts the efficiency of the current, the applied voltage, and therefore its energy consumption. In contrast, an excessive quantity of NaCl causes an increase in aluminium electrode consumption owing to corrosion. The addition of NaCl must be restricted and optimised for this reason. The best electrolytes seem to be monovalent ion salts (Chen, 2004).

Electrode passivation is one of the primary operational issues in EC. It has been extensively observed and acknowledged that the passivation of electrodes, notably aluminium electrodes, is harmful to the performance of the process. On the electrode surface, the creation of an inhibitory layer, often an oxide, hinders metal dissolution and electron transfer, hence restricting the addition of coagulant to the solution. This layer thickens with time, diminishing the efficacy of EC (Shahedi et al., 2020).

Certainly, the use of novel materials, alternative kinds of arrangements, and more complex operating tactics (such as the periodic switching of electrode polarity) have contributed to a decrease in electrode passivation. In addition, the presence of anions decreases passivation. In particular, the addition of Cl<sup>-</sup> to the solution hinders the process of electrode passivation. In many instances, a mechanical cleaning of the surface of the electrodes is also required (Kabdaşlı et al., 2012).

It is essential to model EC in order to systematically assess the EC process. To date, several models, including as phenomenological, adsorption, variable-order kinetic, and flocculation models, have been developed to represent the EC process (Gheraout, 2019). EC modelling is very useful for enhancing the design and decreasing both equipment and running expenses. It can rapidly give us with solid and precise answers to EC challenges, allowing us to estimate EC cell performance throughout a broad range of operating circumstances (Hakizimana et al., 2017).

### 3.4

#### EC treatment of food processing wastewater

Food processing is among the sectors that consume the most water resources and have a vital role in achieving sustainable development goals. Water-intensive sectors, such as food processing, have become a concern to limited freshwater resources, and significant efforts are being made to develop and implement new water management strategies for these businesses (Asgharnejad et al., 2021).

The effluents from food processing are high in oil, suspended solids, and biochemical oxygen demand (Oh; Logan, 2005). Dairy, bakery, meat and poultry, oil extraction, and fish processing are the key food processing industries that generate effluents (Compton et al., 2018).

It is estimated that the food processing sector in the United States generates 1.4 billion litres of wastewater annually (Oh; Logan, 2005). Process functions (e.g., water is being used in the process as a raw material) and non-process functions (e.g., water is being consumed as a utility for applications such as washing, cooling, and heating) account for the majority of water consumption in food processing (Walsh; Cusack; O'Sullivan, 2016).

Depending on the kind of feedstock and industrial process, the flow rate, and pollutant concentration in wastewaters from food processing facilities vary greatly (Compton et al., 2018). This type of effluent is frequently released untreated into water streams and soil since it is quickly biodegradable. However, the high biochemical and chemical oxygen demands of these effluents, as well as the presence of large quantities of nutrients, organic carbon, organic nitrogen, inorganics, suspended and dissolved solids, can lead to deoxygenation of rivers, contamination of groundwater, and a change in abiotic environmental factors (Klemes; Smith; Kim, 2008). With more strict environmental standards, the treatment of effluents from food processing businesses before disposal has become required (Feitshans, 2013).

Due to the high biodegradability of these effluents, the presence of compounds that might block the biodegradation processes and the seasonality of many food processing industries can limit the use of traditional aerobic and anaerobic biological processes (Hou; Ji; Zang, 2018). Given that each type of

wastewater from food processing has unique considerations, the optimal solution is the development of effective treatment systems that can be fine-tuned to handle the respective effluent.

Among the developing technologies researched for wastewater treatment, EC has a number of properties that might be useful for the management of food industry effluents: small treatment units, compared to biological systems, of simple construction, assembly, operation, and maintenance, allowing treatment to be conducted at the point of effluent production; versatility, with easy adjustment of operational variables, according to the quality and quantity of effluent, and process interruption, for seasonal effluent production (Brillas; Cabot; Casado, 2003).

Sardari et al. (2018) investigated the use of EC pretreatment prior to ultrafiltration for fouling minimization and steady water recovery in poultry processing effluent. The authors focused on wastewater generated by chillers during chilling operations. The chiller operation is one of the final unit procedures prior to cooking. Blood, oil, grease, and fat particles are the primary pollutants in chiller water, along with the total suspended solids. In one week of experiments, the recovered water volume increased by more than 30 percent.

Sharma et al. (2018) studied the efficacy of EC processes in removing total organic carbon, oil and grease, and suspended solids from canola oil processing effluent sources. For each set of batch EC runs, untreated wastewater was injected into a glass reactor. For all observed current densities, the aluminium electrodes removed more than 80% of dissolved organic carbon, as determined by the authors. At low current densities, iron electrode was incapable of removing more than 16% of dissolved organic carbon. Using aluminium electrodes, the EC method effectively eliminates almost 100 percent of suspended solids.

Gomes et al. (2018) investigated the effectiveness of horizontally and vertically arranged iron electrodes in the treatment of wastewater from a chicken processing facility during EC. The efficacy of the treatment has been measured using chemical oxygen demand, and the removal pathways have been identified using floc characterisation. By adjusting pH, EC operating time, and current density, the treatment efficacy was evaluated. It was determined that the effluent quality after EC treatment was sufficient to fulfil regulatory discharge criteria.



In the United States and Europe, the baking industry is among the major consumers of water. The average daily water consumption in the bakery business ranges from 10,000 to 300,000 gallons per day, with approximately half of this water being discharged (Haque et al., 2016). As such, it is necessary for bakery manufacturers and enterprises to treat this wastewater and regulate this type of pollution in accordance with wastewater standards.

The use of EC to the treatment of bakery effluent has been the subject of only one study to date, despite the fact that the area is of major importance. Santana et al. (2018) assessed the use of EC as an alternative/complementary method to treat bakery effluent efficiently, with an emphasis on determining the ideal operating conditions and the kinetics of EC at the optimally determined set of process variables. A 23-factorial design of EC experiments with iron and aluminium electrodes was used to explore for the best condition. At the optimal pH (7.0) and voltage (12 V), kinetic studies of EC with aluminium electrodes were conducted by periodically measuring the pH, turbidity, apparent colour, concentration of oil/grease, chemical oxygen demand, concentration of chloride anion, and electric conductivity of wastewater. Almost all of the measured pollutant parameters, and especially the oil/grease content (traditionally poorly removed in fat traps), were decreased to insignificant levels.

### **3.5 EC modelling**

Simulation of EC wastewater treatment has been the subject of a substantial amount of study in recent years. There are three primary types of modelling for EC: statistical modelling, modelling based on knowledge, and machine learning modelling.

Publications retrieved by the database Scopus using the keywords electrocoagulation and modeling and released between 2004 and 2022 are presented in this section.

In bibliometric research, visualization of similarities that intends to facilitate the construction and viewing of bibliometric maps is gaining prominence. This strategy permits quick literature collection and the establishment of interrelationships between selected articles. Figure 21 stands for the network

visualization that appeared in the research papers. The cloud map displays the frequency of the keyword in the articles as well as their relationship. Each color reflects a collection of terms that have been merged into clusters. As may be seen, statistical methods play a significant role in EC modelling. Methods that employ machine learning algorithms are still scarce in the literature; thus, a knowledge gap exists. This thesis intends to contribute to diminish this gap.

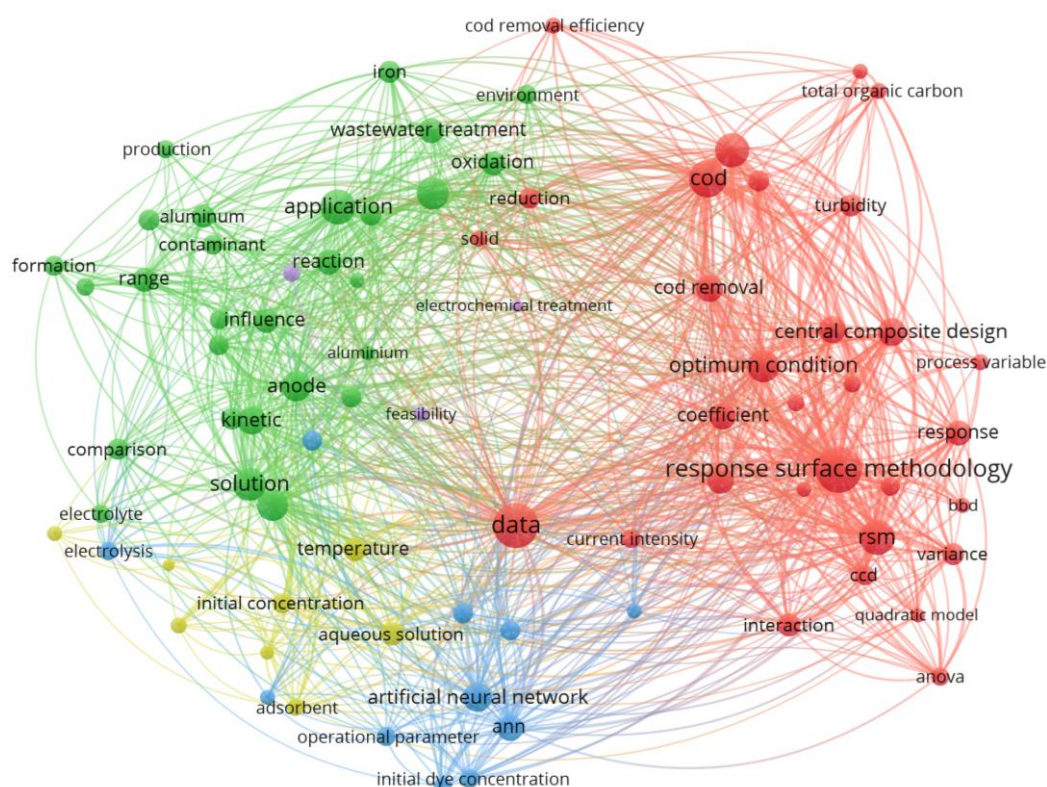


Figure 21 - Network visualization of EC modelling research (generated using the VOSviewer software).

### 3.5.1 Statistical modelling

Designing a multivariable experiment is greatly aided by statistical tools. A one-factor-at-a-time investigation of any process is fundamentally inadequate for determining the relationships between processes. The response surface methodology is a mathematical and statistical approach to experiment design. The goal is to maximise a response that is affected by several independent factors (Nair; Makwana; Ahammed, 2013). Numerous studies have employed response surface approach to optimise the parameters of the EC process, as mentioned below.

Gautam et al. (2022) established a systematic experimental design to examine the treatability of leachate using EC, with the decrease in chemical oxygen demand serving as the response variable. For design of experiments and process optimization, response surface approach was used, and a three-dimensional surface response was developed to comprehend the link between process parameters and response variables. Under optimal circumstances, a decrease in chemical oxygen demand of around 90 percent may be obtained. Thus, the authors concluded that EC may be utilised for the effective treatment of many types of wastewater, including leachate, and that the process can be adjusted utilising the response surface technique.

Mariah and Pak (2020) described batch EC using iron electrodes as a simple and cost-effective approach for treating green dye-containing water solutions. Using response surface methodology, experiments were designed and the procedure was optimised. Three variables (pH, current, and treatment duration) were selected to analyse and improve the process, as well as to investigate the influence of variables on removal efficiency and electrical energy consumption. The removal effectiveness was 96.11 percent under optimal condition, as determined by multiple response optimization.

Nariyan et al. (2018) utilised EC to remove uranium from mine water in Finland. The removal efficiency, isotherms, and kinetic data were derived after studying the influence of the electrode type, current density, and reaction time. The EC process was optimised using response surface methodology in order to determine which parameter is statistically significant and if there is an interaction between the parameters. Fisher's F test was used to estimate the overall significance of the model based on a statistical analysis of the data and an assessment of the variance. Validation of the model's fit was performed using the related probability values and the coefficient of determination. Under optimal circumstances, 97.7 percent of uranium was removed.

Khan et al. (2020) suggested the EC procedure for the total removal of arsenic from an aqueous medium with little energy usage. Rapid arsenic removal of around 95% was achieved under optimal circumstances. To determine the influence of process variables such as applied current, initial concentration, initial pH, and runtime on response variables, their experimental values were adjusted in line with

the design. The contributions of process parameters were investigated with insights into EC mechanisms and formation of distinct metal complexes responsible for adsorption-based arsenic removal. Checking the resulting surface models revealed a relatively excellent match with high values of coefficient of determination for removal efficiency and energy consumption of 0.93 and 0.98, respectively.

### 3.5.2

#### Modelling based on knowledge

Numerous publications have investigated the kinetics of EC in order to study and simulate the process as well as construct an EC system based on the classical kinetic law (Hakizimana et al., 2017). In addition, standard adsorption kinetics models have been frequently employed to better comprehend the EC processes and for modelling purposes (Gheraout, 2019). Since the quantity of coagulant produced can be calculated for a given period using Faraday's equation, the adsorption phenomena may be used to simulate pollution abatement.

Wu et al. (2021) suggested an innovative EC reactor for removing azo dyes from aqueous solutions using EC. Under ideal circumstances, the authors determined that the methyl orange decolorization rate was 92.35 percent. The adsorption of methyl orange on iron hydroxide may be represented by second-order dynamics models, indicating that the chemisorption process governs the adsorption. Consequently, methyl orange was more suited for the Langmuir adsorption isotherm line, and the findings were favourable and consistent with the experimental data.

Ilhan et al. (2019) evaluated the removal of chromium, copper, zinc, and nickel, which is one of the most crucial components in metal plating wastewater treatment. The current density, initial pH, and time, which are significant EC process factors, were examined. Using kinetic model investigation, the removal process was analysed. When kinetic modelling was performed, it was determined that pseudo second order kinetics was more applicable. This demonstrated that the metal hydroxides produced by EC remove heavy metals by adsorption.

Das and Nandi (2019) conducted experimental research to improve different EC operating settings for the effective removal of iron (Fe(II) ions) from drinking water. Analysis of the kinetics of the EC process suggests that Fe (II) ion

elimination follows a first-order kinetic model with respect to various operating parameters. In addition, kinetic analysis has been performed to determine the impact of process factors such as current density and initial concentration of Fe(II) ions on its removal. During EC, the findings revealed that the rate constant increased as the current density rose, owing to the increasing availability of aluminium ion flocs.

Bener et al. (2019) examined the EC removal of total organic carbon, colour, turbidity, and dissolved particles from pretreated textile wastewater. The trials were adjusted to Turkey's water standards, and the findings were compared to irrigation reuse requirements. A kinetic analysis was conducted to establish the reaction order and rate constants. Models of first and second order kinetics were studied. With increasing current density, the reaction rate increases for both models; hence, the second-order kinetic model with coefficient of determination values between 0.962 and 0.986 was deemed more appropriate.

### **3.5.3 Machine learning modelling**

As a result of the intricate interactions between input parameters and outputs, electrochemical methods for wastewater treatment are extremely complex nonlinear systems (Kabdaşlı et al., 2012).

Describing the phenomenological process of EC requires a variety of partial differential equations, which are sometimes difficult to solve and include several model parameters (Pirdashti et al., 2013). In phenomenological modelling, all relevant species inside an electrochemical cell are accounted for, necessitating additional information on reaction pathways (Hakizimana et al., 2017). In addition, statistical modelling is only beneficial for quadratic approximations, therefore to apply statistical approaches successfully, the search window must be adequately limited, necessitating either further tests or a thorough understanding of the EC system beforehand (Nair; Makwana; Ahammed, 2013).

The widespread use of machine learning methods in process modelling and optimization is attributable to their very accurate prediction capabilities. Because they can approximate any function with a high-dimensional system and are versatile

in modelling nonlinear behaviour patterns, these approaches are expected to produce more acceptable outcomes than statistical methods (Zhao et al., 2020).

Hasani et al. (2018) introduced an innovative use of ANNs for modelling and optimization of a new alternating pulse current EC-flotation method for the removal of humic acid from aqueous environments. With a coefficient of determination of 0.971, computational findings demonstrated that ANN modelling accurately replicated experimental data and predicted optimal performance.

Using computational methods, Morales-Rivera et al. (2020) developed and optimised an EC treatment for the removal of chemical oxygen demand from cold meat effluent. The ANN was assessed with 4, 6, 8, 10, and 12 hidden neurons to find a satisfactory model. For each architecture, 200 simulations were done and the mean square error was calculated. With a correlation value of 0.99, the created model effectively describes the process.

Gholami Shirkoohi et al. (2022) employed machine learning models, such as adaptive neuro-fuzzy inference systems, ANNs, and SVM, to forecast the removal efficiency of phosphate from wastewaters using the EC process. By incorporating metaheuristic methods such as genetic algorithms and particle swarm optimization into these models, the ideal hyperparameters were determined. The mean square error, the coefficient of determination, and the mean absolute percentage error were used as comparison metrics to assess the performance of the models. Results indicated that ANNs performed better than other models.

Zhang et al. (2020) developed a model for predicting the concentration of heavy metal ions during the EC process. The authors introduced an integrated prediction model by coupling a kinetics model with ANN model to correct for the kinetics model's inaccuracies caused by industrial process uncertainties. All experimental simulation findings for model validation based on industrial process data revealed that the integrated model is very accurate and can forecast the concentration and trend of heavy metal ions. When industrial circumstances change dramatically, the benefits of the integrated approach may be highlighted more clearly.

Zhu et al. (2021) suggested a technique for predicting the EC reactor's removal rate based on a deep learning LSTM. In order to facilitate the model ability to analyse and learn the overall trend of the data and to more accurately predict the

reactor removal rate, the change gradient value of the historical removal rate at two adjacent times of the reactor is extracted as the feature value to reflect the removal rate's change trend. In addition to serving as one of the model's input variables, the feature value is utilised to forecast the EC reactor's removal rate. Validated using real industrial process data, the results demonstrate that the method's projected value is closer to the actual value and has a coefficient of determination of 0.928.

## 4

### **Fault detection in bakery industrial effluents treatment by electrocoagulation using neural networks and feature selection**

The first part of this work used seven feature selection methods to select the most important features in a given dataset. The performance of neural network classification models trained on the original feature set was compared to the performance of those that were trained on a subset of features that had been curated using feature selection techniques. The model that utilised feature selection was found to have the best performance (F1-score = 0.92) and an improvement of more than 30% in preventing false positives.

This section contains the manuscript version of the article that presents the results of the development of the model.

#### 4.1

##### **Article Manuscript: Fault detection in bakery industrial effluents treatment by electrocoagulation using neural networks and feature selection**

Thiago da Silva Ribeiro<sup>a</sup>, António José dos Santos Rodrigues<sup>b</sup>, Brunno Ferreira dos Santos<sup>a</sup>, and Maurício Leonardo Torem<sup>a</sup>

<sup>a</sup>Department of Chemical and Materials Engineering, Pontifical Catholic University of Rio de Janeiro, Rua Marquês de São Vicente, 225, Gávea, Rio de Janeiro, RJ 22453-900, Brazil

<sup>b</sup>VentilAQUA, S.A., Estrada da Ponte, Lote A. Antanhol, 3040-575, Coimbra, Portugal

#### 4.1.1

##### **Introduction**

Globally, water resources are becoming increasingly scarce as a result of increases in population, climate change, industrial development, and changing water consumption patterns. Since industrial wastewater contains a variety of pollutants, it must be treated to reduce pollution and meet water emission standards.



Several factors, such as wastewater sources and treatment techniques, influence the quality of industrial wastewater discharge (Diaz-Elsayed et al., 2019).

The bakery industry is one of the world's most important food industries, and its manufacturing size and methods vary greatly (Jerome; Singh; Dwivedi, 2019). While baking effluent normally does not include toxic compounds, it is rich in organic matter (mostly flour and sugar) and oil/grease. Little levels of detergents, yeast, salt, and other food additives are also present (Mohan; Vivekanandhan; Priyadharshini, 2017). The significant daily water consumption in the bakery industry (10–300 thousand gallons, mostly utilised for cleaning operations) is also an environmental issue, especially considering that at least half of this water is disposed as wastewater (Chen et al., 2004).

EC combines the benefits of coagulation, flotation, and electrochemistry, and is effective in the treatment of water and wastewater (Shahedi et al., 2020). Coagulation and EC are based on similar principles; both methods aim to remove particles from wastewater by weakening or neutralising the repulsive forces that keep them suspended in water. Coagulant species are generated *in situ* during the EC process by utilising an electric current applied between electrodes to electro-dissolve a sacrificial anode, commonly made of iron or aluminium. In this approach, water is converted to hydrogen gas and hydroxyl anions at the cathode.

The metal cations generated by this process pass through a variety of equilibrium reactions in water, including acid-base, complexation, precipitation, and redox reactions. Since anode oxidation can lead to either ferrous or ferric cations, the electrochemical reactions occurring at iron electrodes during EC are more complex than those that occur when using aluminium electrodes (Moussa et al., 2017).

Several factors have an impact on EC. In particular, the current controls the coagulant dosage rate, bubble generation rate, as well as floc size and growth, all of which affect the efficiency of the EC process. The rate of anodic dissolution and hydrogen production increases as the current density increases, resulting in the greater removal efficiency of effluent pollutants (Sahu; Mazumdar; Chaudhari, 2014). In addition, the cell voltage is a function of the equilibrium potential, along with the anode and cathode overpotentials (Chen; Chen; Yue, 2002). pH is another critical parameter in the EC process, which determines the redox processes involved

and the solubility of the hydroxides generated. Furthermore, the pH significantly impacts the pitting corrosion of the electrodes.

Magnisali, Yan and Vayenas (2022) presented several case studies involving commercialised EC systems, as well as real process data gathered from technology providers around the world that covered a variety of industries. They found that EC could be a good alternative to established treatments. The review also forecasted how the maturity of the technology and its ease of adaptation would allow EC to continue to grow in the market.

Monitoring wastewater quality is becoming increasingly important around the world since it allows for a better understanding of both treated and untreated effluent, as well as better control of treatment plants (Newhart et al., 2019). The early detection and diagnosis of faults in plant operations could lead to increased safety and productivity. Model-, data-, and knowledge-based fault detection and diagnosis technologies have been applied within the industry, but researchers are paying an increasing amount of attention to data-driven methodologies since they provide high diagnostic accuracy and do not require building a phenomenological model.

The foundation of data-driven approaches is to take advantage of the many datasets that are available in the system being examined. Machine learning algorithms have been utilised to automate and make decisions in a wide range of complex operations due to their stability and reliability. They have been employed in real-time monitoring and data analysis (Sundui et al., 2021), which are critical for assuring safe and consistent WWTP operations.

Data-driven anomaly detection methodologies based on deep learning methods and clustering algorithms were proposed and validated by Dairi et al. (2019); their findings indicated that an unsupervised one-class SVM scheme was capable of monitoring WWTPs without the need to verify any assumptions regarding the distribution of the data. Mamandipoor et al. (2020) proposed a method to detect collective defects in WWTP sensor data, taking into account the data's multivariate, non-linear, and temporal behaviour. This new method overperformed traditional monitoring methods, achieving a fault detection rate of over 92%. In addition, Kazemi et al. (2021) validated a data-driven framework for fault detection

in anaerobic digestion. The authors concluded that combining a SVM with univariate control charts increased the detection of small faults.

Combining machine learning algorithms with feature selection techniques has become a necessity in many applications as there are very few data-driven approaches that are designed to deal with large numbers of features. An excess of features can result in several issues, such as i) irrelevant features resulting in the overfitting of the training data, thereby lowering the model's accuracy (i.e., poor generalisation); ii) models with high complexity limiting their interpretability, consequently hampering any decision-making; iii) models with several features that become impractical and difficult to implement (Tang; Alelyani; Liu, 2014).

To overcome this problem, machine learning should be preceded by a feature selection step, which refers to the process of selecting a subset of original attributes to minimise the feature space based on a set of criteria. This is accomplished by reducing the number of features, allowing the model to focus on the most important data while enhancing their quality, resulting in a better understanding of the processes that are described by the selected features (Venkatesh; Anuradha, 2019).

Alternative approaches, such as feature extraction or dimensionality reduction techniques (such as principal component analysis), may also produce strong results that increase the accuracy of a machine-learning system's predictive model. However, in most circumstances, they are less interpretable than feature-selection solutions (Kudelina et al., 2021).

Filters, wrappers, and embedding methods are among the statistical methods that can be used in feature selection techniques.

Filtering is a preprocessing step that takes place before classification and is thus independent of the prediction method used; i.e. no learning algorithms are applied. Many mathematical expressions, such as correlation-based methods, gain ratios, or information gain, have been used to evaluate the feature relevance. Filter techniques have the advantage of being able to scale to very large datasets, their computational ease and speed, and their independence from the classification algorithm (Hopf; Reifenrath, 2021). However, a typical shortcoming of filter methods is how most proposed methods are univariate and neglect the interactions with the classifier; specifically, the search in the feature subset space is separated from the search in the hypothesis space.

Wrapper techniques evaluate the subset of features by using the predictor as a black box and the predictor's performance as an objective function. Since assessing  $2^N$  subsets is a non-deterministic polynomial time hard problem, suboptimal subsets are discovered using search algorithms, which then heuristically select a subset for further analysis. The primary disadvantage of these methods is that they are more prone to overfitting than filter techniques, while also being relatively computationally costly (Deng et al., 2019).

Embedded methods are a mix of filters and wrappers that employ different learning algorithms. Unlike filters and wrappers, these methods use machine learning algorithms that include feature selection as part of the model-building process (i.e. feature selection is embedded with the learning algorithm). Embedded methods are similar to wrapper methods in that they try to find the most useful subset of features (Li; Li; Liu, 2017). Embedded methods have the advantage of including the interactions of the classification model while being significantly less computationally costly than wrapper approaches.

Cui, El-Arroudi and Weng (2019) proposed an approach for feature extraction in high-impedance fault detection. The authors demonstrated that the suggested strategy considerably improved the classification performance of their ANN in their effective feature set. Lee and Wen (2020), meanwhile, employed a feature selection approach to detect induction motor failures. The features chosen by selection techniques were used to categorise and classify the four types of failure using an ANN. They found that feature selection could lower the number of features and the costs of operating the system while still achieving a high degree of accuracy.

ANNs are powerful predictive tools that are premised on the universal approximation theorem and are utilised for both regression and classification problems. The collective work of the unit building blocks, i.e. the neurons, is key to ANNs. These neurons are modelled after the behaviour of biological neurons (Zou; Han; So, 2008). An ANN is considered to be a 'black box' model since it is difficult to interpret fundamentally compared to other models such as linear regression.

An ANN aims to predict an outcome based on the information contained within a training set. Current implementations use performance metrics to determine the applicability and validity of the model for specific use cases.

However, focusing solely on these measurements can lead to biases that favour specific input features. Therefore, identifying the input features that contribute most to the target output prediction in the ANN learning process is critical to its performance (Abiodun et al., 2018).

In this study, an ANN was employed to evaluate the effectiveness of the rank aggregation of several feature-selection methods (filter, wrapper, and embedded) compared to the original dataset (with all features), in the context of identifying the operational condition of a wastewater treatment plant. In addition, this research aimed to apply a better comprehension of the examined data and the relative importance of the features for the EC process.

#### **4.1.2 Methods and materials**

This research applied seven feature selection methods: the univariate statistical model of ANOVA F-value, MI, a Relief-based algorithm, RFE, SHAP, PI, and an embedded approach that employed a RF model. A new feature set was generated by averaging the scores from the outcomes of these seven feature selection methods — this process is known as rank aggregation, and it aims to create a new consensus ranking by combining the results of multiple rankings. This results in a new order of significance for the features selected. The top four features in the ranking list were selected based on the authors' expert knowledge of the EC domain.

Feature selection and machine learning algorithms were implemented using the following libraries in Python (v3.10.2): scikit-learn (v1.0.2), scikit-rebate (v0.62), ELI5 (v0.11.0), SHAP (v0.40.0), and TensorFlow (v2.8.0). If no additional parameters were supplied, the methods' default values were utilised when using the feature selection algorithms. This paper presents the seven feature selection methods, the ANN employed, as well as the differences in the performance of models trained using the original feature set to those trained using the rank-aggregated features.

##### **4.1.2.1 Dataset description**

In this study, data were collected from a decentralised EC WWTP that was supplied and commissioned by VentilAQUA (Coimbra, Portugal). The location of the unit was a bakery in Slovenia.

VentilAQUA's VABEC® technology is an EC continuous-flow system, multi-electrode cell that is composed of electrodes made of materials that are suitable for oxidation and coagulation, with a modular configuration and an internal geometry designed for optimum efficiency. Following the chemical reaction phase, a flotation procedure is employed for solid-liquid separation. The DAF unit is a pre-assembled, compact system built using VentilAQUA's VAMEF® technology.

This system contains a dedicated electrical box with a power rectifier to supply an electrical current to the electrodes, make amperage adjustments to meet the treatment objectives, as well as carry out an automatic, scheduled power-potential shift as an anti-passivation procedure. An electrical control panel equipped with a PLC manages the operation of the entire unit.

The features that were monitored to determine the process quality and efficiency were as follows (Figure 22): conductivity (before and after the EC process); pH (before and after the EC process, and inside the DAF system); flow (in both the EC and DAF system); voltage; current; polarity; and ORP (inside DAF system).

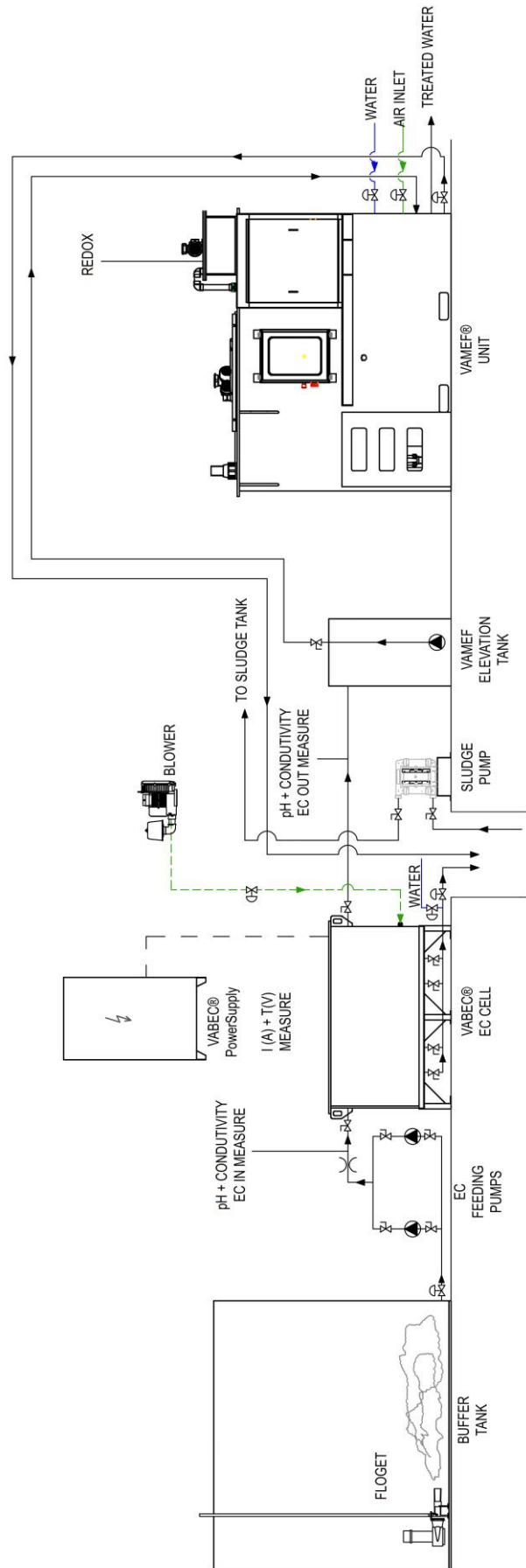


Figure 22 - The relative positions of the sensors in the WWTP.

The target variables were three operational modes based on the effluent clarification and the reaction sludge. We treated the problem as a classification task, with three classes based on expert knowledge as follows:

- Class 0: Not clarified, showing turbidity;
- Class 1: Clarified, showing low turbidity;
- Class 2: Clarified, although the system had an excessive electrode and energy consumption.

Tables 1 and 2 provide a general overview of the variables and characteristics of the dataset, respectively.

Table 1 - Overview of the number of variables and observations in the dataset.

<b>Number of input variables</b>	<b>11</b>
<b>Number of output variables</b>	<b>1</b>
<b>Number of observations</b>	<b>1207</b>

Table 2 - Characteristics of the dataset.

	Mean	Minimum	Maximum	Interquartile range (IQR)	Standard deviation	Variance
<b>EC Flow (m<sup>3</sup>/h)</b>	2.952	1.509	3.454	0.003	0.184	0.034
<b>Conductivity IN (mS/cm)</b>	1.140	0.823	1.727	0.143	0.107	0.011
<b>pH IN EC</b>	4.354	4.162	4.770	0.191	0.171	0.029
<b>Current (A)</b>	394.154	172.658	747.848	187.414	107.666	11,591.947
<b>Tension (V)</b>	4.502	0.728	6.372	1.358	0.818	0.669
<b>Polarity</b>	Categorical (binary variable): Class 0: 53%; Class 1: 47%					
<b>Conductivity OUT (mS/cm)</b>	1.183	0.885	1.777	0.128	0.110	0.012
<b>pH OUT EC</b>	4.885	4.598	5.259	0.298	0.203	0.041
<b>VAMEF Flow (m<sup>3</sup>/h)</b>	3.002	1.909	4.868	0.034	0.106	0.011
<b>ORP VAMEF (mV)</b>	-507.395	-635.276	-9.185	123.680	72.242	5218.972
<b>pH VAMEF</b>	8.095	6.457	10.470	0.903	0.765	0.585
<b>Operational modes (target variable)</b>	Categorical (three classes): Class 0: 34%; Class 1: 40%; Class 2: 26%					



#### 4.1.2.2 Feature-selection methods

The chosen feature-selection methods can capture various aspects of the data, and thus the combination of these techniques allows for the exploration of their complementary potential in classifying wastewater treatment plant operating conditions.

##### 4.1.2.2.1 Filters

###### ○ ANOVA

ANOVA is a statistical method used to assess whether the means of two or more groups change significantly. This technique assumes a linear relationship between the factors and the dependent variable, in addition to the normal distribution of the variables. ANOVA employs F-tests to statistically analyse the equality of means, which relates to the intergroup to intragroup variability ratio. Since groups are described as discrete variables, this method may be utilised to solve classification problems. ANOVA carries out an analysis of variance for each feature that explains the class variable. The score is determined by the statistic's value. The greater the statistic, the wider the variance between the classes' mean values for the associated feature. The score for every feature,  $X_k$ , is defined as follows:

$$J_{ANOVA}(X_k) = \frac{\sum_{i=1}^l n_i (\bar{x}_i^{(k)} - \bar{x}^{(k)})^2 / (l-1)}{\sum_{i=1}^l \sum_{j=1}^{n_i} (x_{ij}^{(k)} - \bar{x}_i^{(k)})^2 / (n-l)} \quad (29)$$

where  $l$  represents the number of classes of  $Y$  and  $x_{ij}^{(k)}$  represents the observed values of feature  $X_k$  for class  $i$  instances.  $\bar{x}_i^{(k)}$  is the mean value of  $X_k$  for class  $i$  whereas  $\bar{x}^{(k)}$  is the mean value of  $X_k$  for the whole data set.

This test's null hypothesis is that the feature has the same value regardless of the value of  $y$  being predicted. ANOVA F-value may be used for feature selection, where features unrelated to the target variable can be ignored (Dhal; Azad, 2021).

- MI

Entropy, divergence, and MI are fundamental notions outlined under the theory of information. In probability and information theory, MI is a measure of redundancy between two variables. Specifically, this method measures the amount of information gained about one variable when the other is observed, and it can capture any kind of relationship between two variables. MI is closely related to the Shannon entropy, which is a measure of random variable uncertainty.

Let  $X$  and  $Y$  each have their own empirical probability mass function  $p$ . Then, the definition of  $Y$ 's entropy is:

$$H(Y) = -\sum_y p(y) \log_2(p(y)) \quad (30)$$

And the conditional entropy of  $Y$  given  $X$  is determined as follows:

$$H(Y|X) = \sum_x p(x) H(Y|X = x) = \sum_x p(x) (-\sum_y p(y|x) \log_2(p(y|x))) \quad (31)$$

If  $X$  and  $Y$  share a significant amount of MI, this indicates that they are closely related. In contrast, if  $X$  and  $Y$  are fully independent, then the MI between them is equal to 0 (Vergara; Estévez, 2014). The MI of two variables is thus defined as:

$$I(Y; X) = H(Y) - H(Y|X) \quad (32)$$

It may be understood as the reduction in uncertainty about  $Y$  due to knowledge of  $X$ . In light of the symmetry property, it may alternatively be interpreted as the amount of shared information between  $X$  and  $Y$ . The expression quantifies the feature's importance based on the information it contains about  $Y$ . The objective is to identify features that are highly important and minimum redundant.

- Relief-based algorithms

The Relief algorithm and its variations can find feature dependencies through the use of individually evaluated filter techniques (Venkatesh; Anuradha, 2019). Instead of searching through feature combinations, these algorithms use the concept of nearest neighbours to obtain feature statistics that indirectly account for such interactions.

Relief is an algorithm that iterates over  $m$  randomly picked training instances ( $R_i$ ) without replacement, where  $m$  is a user-specified value. Each cycle,  $R_i$  represents the *target* instance, and the feature score vector  $W$  is updated depending on observed feature value differences between the target and its neighbours. Therefore, the distance between the *target* instance and all other instances is computed each cycle. Relief detects two closest neighbour instances of the *target*: one with the same class, referred to as the *nearest hit* ( $H$ ), and one with the opposite class, referred to as the *nearest miss* ( $M$ ). In the last phase of the cycle, the weight of a feature  $A$  is updated in  $W[A]$  (Equation 33).

$$w_j^{(t+1)} = w_j^{(t)} - (x_j - H_j)^2 + (x_j - M_j)^2 \quad (33)$$

Features whose values vary between  $R_i$  and  $M$  support the premise that they are informative of outcome, hence increasing the quality estimate  $W[A]$ . In contrast, features with disparities between  $R_i$  and  $H$  give evidence to the opposite, hence decreasing the quality assessment  $W[A]$  (Urbanowicz et al., 2018).

#### 4.1.2.2.2 Wrappers

- RFE

RFE is a feature selection method based on wrappers. This technique is a *greedy* algorithm and model-based reverse search strategy. It begins its search with the whole feature set, and its performance is measured by the accuracy of its predictions. At each stage of the iterative process, RFE fits a classifier with all current features, computes a ranking standard for each feature, and eliminates the feature with the lowest ranking standard. RFE calculates the final feature

significance index by combining the coefficient standard of each feature with the loss function of the predictor. First, the appropriate feature ranking computation standard must be defined. The usual standard for feature coefficients in classification problems is defined as follows:

$$w_j = \frac{\mu_j(+)-\mu_j(-)}{\varepsilon_j(+)+\varepsilon_j(-)} \quad (34)$$

where  $\mu_j$  and  $\varepsilon_j$  are the mean and standard deviation of the  $j$ th eigenvalue for all samples belonging to class (+) or class (−), respectively.  $w_j$  is positively correlated with the magnitude of the relationship between the  $j$ th feature and the target value.

In a predictive model, RFE assigns weights to all features. Based on the weights given to each feature, the least important feature is removed from the feature set at each step. The procedure is repeated until the desired number of features has been obtained (Khaire; Dhanalakshmi, 2019). On the basis of the classification model and correlation coefficient, the following criteria for ranking the features may be established:

$$R_j = \frac{1}{2} \frac{\partial^2 J}{\partial w_j^2} w_j^2 \quad (35)$$

where  $J$  is the loss function of the classification model on the training data,  $w_j$  is the correlation coefficient of the  $j$ th feature, and  $R_j$  is an approximation of the sensitivity of the  $j$ th feature to the classifier, i.e., the degree to which the  $j$ th feature influences the target value.

#### ○ SHAP

SHAP is an additive explanation approach that attempts to explain the contribution of each feature in the dataset to the model's predicted output. More specifically, SHAP approximates the Shapley values, a concept in game theory that estimates the contribution of each subset of features to a model's prediction given a dataset with  $m$  features.

The Shapley value is defined as the marginal contribution of variable value to prediction over all subsets of features that are possible. In other words, it is one way to share the aggregate gains among the features, assuming they all *cooperate*. The amount that each feature receives at the conclusion of a prediction is defined as follows:

$$\Phi_i(x) = \sum_{S \subset F \setminus \{i\}} \frac{|S|!|F|-|S|-1!}{|F|!} f_{S \cup \{i\}}(x_{S \cup \{i\}}) - f_S(x_S) \quad (36)$$

where  $x$  indicates the observation input.  $\Phi_i(x)$  is the Shapley value for the feature  $i$  for the input  $x$  for the model  $f$ .  $F$  is the set of all features.  $f_S$  is the trained model on the subset of features  $S$ .  $f_{S \cup \{i\}}$  is the trained model on the subset of features  $S \cup \{i\}$ .  $x_S$  is the restricted input of  $x$  given the subset of features  $S$ .  $x_{S \cup \{i\}}$  is the restricted input of  $x$  given the subset of features  $S \cup \{i\}$ .

While obtaining the exact solution for Shapley values is impossible due to the exponential nature of the problem, SHAP approximates the solution using alternative feature-dependence assumptions for ensemble tree models or using specific weighted linear regression for any other model (Lundberg; Lee, 2017).

- PI

PI is a model inspection approach applicable to any fitted estimator for tabular data. This method is defined as the drop in model score caused by randomly shuffling a single feature value. This approach disrupts the relationship between the feature and the target; hence, the decrease in model score indicates how dependent the model is on the feature. PI is model-independent and may be computed several times with various permutations of the feature (Chen; Ishwaran, 2012).

#### 4.1.2.2.3 Embedding

- RF

This study used an embedding approach that employed an RF model, a decision tree-based algorithm. An RF model is a collection of decision trees, where each node of the tree is constructed using random feature selection. A decision tree aims to recursively partition an existing dataset into halves based on the feature that best increases the purity of a node. To evaluate the final relevance of a feature, RF calculates the mean of impurity decrease in each feature across all of its trees. The features chosen at the beginning of the tree construction are ranked first in feature ranking and can be used for model testing and validation (Speiser et al., 2019).

Out-of-bag randomization and the Gini index are two strategies used in random forest models to assess the importance of features. Out-of-bag randomization reveals the importance of a feature by comparing the out-of-bag errors before and after permuting the feature's values. The Gini splitting index is an alternate way for evaluating the relevance of features in ensemble methods based on decision trees. It may provide more sparse feature importance scores than out-of-bag randomization and, as a result, is more interpretable. The Gini splitting index of the  $i$ th feature in an individual decision tree with  $T$  internal nodes is defined as the total of this feature's contributions to the purity of partitioned data in corresponding nodes:

$$s(X_i) = \sum_{t=1}^T g(t) I(v(t) = i) \quad (37)$$

where  $I(v(t) = i)$  denotes whether the  $i$ th feature is chosen in the  $t$ th node to divide the corresponding region and  $g(t)$  is the class purity gain as assessed by the Gini index. The Gini splitting index of the  $i$ th feature in a random forest is the mean of all Gini splitting indices for all random trees.

Table 3 summarises the libraries and functions that were used to implement the feature selection methods. Default values were utilised in all algorithms unless otherwise stated.

Table 3 - Information related to the feature-selection methods used in this paper.

Method	Library	Function	Reference
<b>ANOVA</b>	scikit-learn	f_classif()	Pedregosa et al. (2011)
<b>MI</b>	scikit-learn	mutual_info_classif()	Pedregosa et al. (2011)
<b>Relief</b>	scikit-rebate	Relief()	Urbanowicz et al. (2018)
<b>-based</b>			
<b>RFE</b>	scikit-learn	RFE(estimator=LogisticRegression())	Pedregosa et al. (2011)
<b>SHAP</b>	scikit-learn; shap	RandomForestClassifier(); TreeExplainer()	Lundberg et al. (2020)
<b>PI</b>	scikit-learn; eli5	LogisticRegression(); PermutationImportance()	Korobov and Lopuhin (2021)
<b>RF</b>	scikit-learn	RandomForestClassifier()	Pedregosa et al. (2011)

A new set of the best four features was created by averaging the results of the seven feature selection methods. The Borda rank aggregation method was used to provide a vector position for each feature in the ranking list. The main advantage of the Borda approach is that it is computationally efficient, and can be implemented in linear time (Lin, 2010).

#### 4.1.2.3 Model

A feedforward ANN employs layers of non-linear ‘hidden’ units between its inputs and its outputs. It trains its feature detectors by adjusting the weights on the incoming connections of these hidden units, thereby allowing it to predict the proper output when given an input vector. A multilayer ANN can distort the input space to linearly separate data classes (Lecun; Bengio; Hinton, 2015).

This paper adopted a Bayesian optimisation (BO) approach to automate the selection of optimal hyperparameter values for the models, and the F1-score with a macro average was chosen as the optimisation target. The objective of BO is to determine the combination of hyperparameters that results in the lowest validation error. BO can be described using Equation 38, where  $X$  is the space of possible hyperparameters and the objective function  $f$  aims to minimise validation error. This technique employs a Bayes rule-based surrogate probability model, in which the values of the next iteration are determined based on the results of prior iterations (Snoek; Larochelle; Adams, 2012).

$$x^* = \operatorname{argmin}_{x \in X} f(x) \quad (38)$$

The implementation of BO was based on the open-source *wandb* client provided by Weights & Biases, but only after manually picking a reasonable set of hyperparameter values (Table 4). The GPU computation time for each group (with and without feature selection) was limited to 24 hours on an NVIDIA Tesla P100-PCIE-16GB. Before training the ANN models, the values of the numeric input variables were standardised by subtracting the mean and scaling it to the unit variance.

Table 4 - Hyperparameters tested.

Hyperparameter	Tested settings
Epoch	min:1, max:500 (discrete uniform integer distribution)
Activation function	linear, relu, sigmoid, tanh, selu, elu
No. of hidden layers	min:1, max:5 (discrete uniform integer distribution)
No. of neurons per hidden layer	min:1, max:250 (discrete uniform integer distribution)
Loss function	categorical_crossentropy, kl_divergence, poisson
Learning rate	min:0.00001, max:0.01 (continuous uniform distribution)
Batch size	32, 64, 128, 256, 512
Dropout	0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9
Optimiser function	adam, SGD, adamax, nadam
Kernel initialiser	he_uniform, glorot_uniform, lecun_uniform

#### 4.1.2.4

##### Model performance metrics

An estimation of the prediction error is required to assess the performance of fitted models. K-fold cross-validation is one of the most widely used methods for classifier model selection and error estimates (Fushiki, 2011). During K-fold cross-validation, one part of the dataset is designated as the testing dataset, while the remaining  $k - 1$  parts are designated as the training dataset. This paper used five-fold cross-validation to assess the ANN models. The datasets were separated, and the average of the performance metrics was calculated.



The F1-score with a macro average and the area under the receiver operating characteristic curve (ROC-AUC) were the metrics by which the performance of the classifiers described in this paper was evaluated.

The F1-score evaluates the performance of a classification model by taking the harmonic mean of the precision and the recall of the classifier. The F1-score formula can be read as a weighted average of precision and recall that ranges from 0 to 1, where 0 is the worst score, while 1 is the best score. Precision and recall have the same relative contribution to the F1-score, so the harmonic mean can be used to discover the ideal trade-off between the two metrics. This metric is unaffected by class size since classes of varying sizes are equally weighted in the numerator, which means that the impact of the largest classes is equal to that of the smallest (Grandini; Bagli; Visani, 2020).

The F1-score with a macro average (Equation 39) combines the precision (PPV; Equation 40) and the recall (TPR; Equation 41) of the model to assess its ability to accurately classify data.

$$F1 = \frac{2}{|C|} \sum_{i=1}^{|C|} \frac{(TPR_i * PPV_i)}{(TPR_i + PPV_i)} \quad (39)$$

$$PPV = \frac{TP}{TP + FP} \quad (40)$$

$$TPR = \frac{TP}{TP + FN} \quad (41)$$

In the above Equations, TP refers to the number of true positive predictions compared to the test data, FP refers to the number of false positive predictions, FN refers to the number of false negative predictions, and C is the number of classes.

The ROC curve is a two-dimensional representation of the probabilistic model's classification performance at different thresholds, while the ROC-AUC is a scalar metric that measures the overall performance.

After evaluating the performance metrics of the trained models, the best models trained on datasets with and without feature selection were compared. To evaluate both models in more detail, the confusion matrices of the models were compared, allowing for the visualisation of a variety of classification performance metrics, such as accuracy, precision, and recall (Raschka, 2014).

### **4.1.3 Results**

#### **4.1.3.1 Feature-selection experiments**

Figure 23 compares the findings of the feature selection approaches utilised in this study. The diagram reveals that certain features were always ranked first (i.e. were deemed to have the greatest importance), regardless of the technique used.

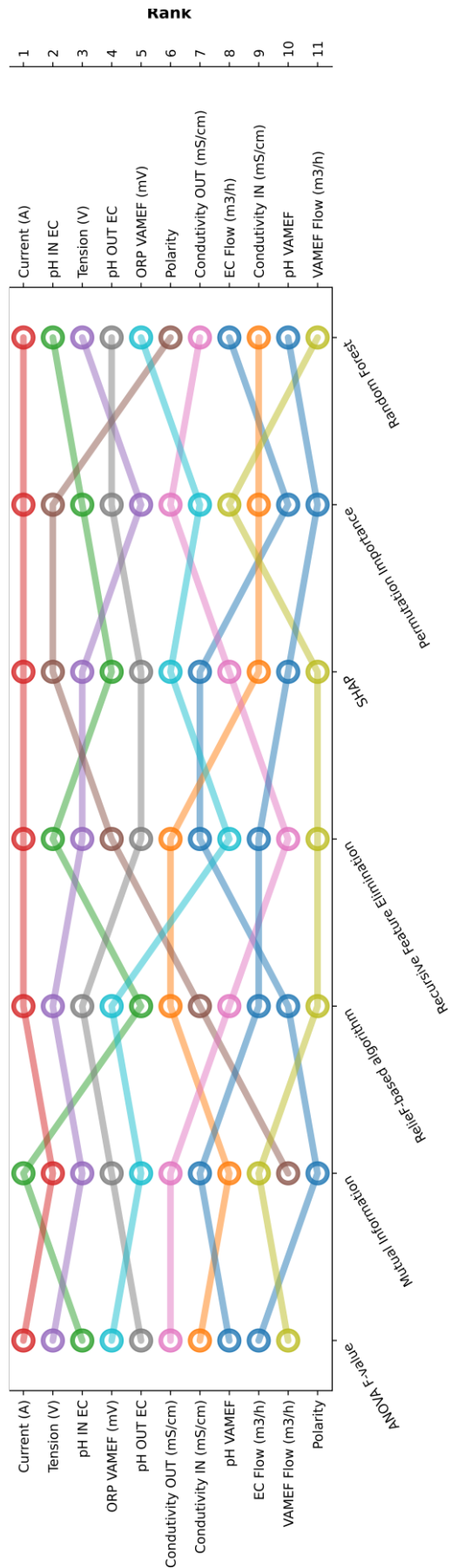


Figure 23 - The ranking of features using the different feature selection methods.

Regardless of the method used, the current feature is almost always ranked first (the only exception was the MI technique, in which it was ranked second). The importance of this feature is consistent with the key role that current density plays in the EC process. According to Hakizimana et al. (2017), the current is a critical parameter in the EC process as it predicts the structure and the evolution rate of the flocs generated. The authors highlighted the impact of this variable when modelling techniques for the simulation and scaling of EC operations.

The applied voltage frequently ranks among the top three features except for the PI technique. In an EC process, the operating voltage is crucial because it impacts the energy consumption, the mass transfer at electrodes, and the mixing of the solution. Safonyk et al. (2019) proposed a low-energy mode of operation for EC wastewater treatment control and developed an EC process model to determine the optimal applied voltage that minimised power consumption. Similarly, this paper found that the applied voltage feature was of critical importance.

Two additional features stand out when evaluating their overall importance: the pH of the effluent before and after the EC process. Indeed, the pH of the effluent is an important feature in an EC process, as it influences the type and amount of metal hydroxide complexes that are generated. Graça et al. (2019) developed a mathematical model that predicted the pH of the EC process. The model was designed to serve as a foundation for the development of more complex models that could be used to estimate the performance of an EC cell. This paper similarly emphasises the importance of this feature in the EC process.

The previously highlighted features comprised the top four features as ranked by the Borda rank aggregation method (Table 5). This subset of features was then used as the input for a multilayer perceptron ANN, and the performance of this model was compared with a similar model trained on the complete feature set.

Table 5 - Aggregated ranking of the features in the dataset.

# RANK	FEATURE
1	Current (A)
2	pH IN EC
3	Tension (V)
4	pH OUT EC
5	ORP VAMEF (mV)
6	Polarity
7	Conductivity OUT (mS/cm)
8	Conductivity IN (mS/cm)
9	EC Flow (m <sup>3</sup> /h)
10	pH VAMEF
11	VAMEF Flow (m <sup>3</sup> /h)

#### 4.1.3.2

##### Comparative performance evaluation

Two metrics were chosen to assess the performance of the feature selection strategy: the F1-score with a macro average and the area under the ROC curve (ROC-AUC). These performance indicators are frequently employed to assess the success of classification models in data mining applications (Grandini; Bagli; Visani, 2020).

Figure 24 presents the progress of hyperparameter optimization process that was conducted for each group (with and without feature selection). The diagram reveals that the optimization algorithm improves with iterations, therefore optimal hyperparameter values were found in the search space for both groups. However, it is important to note that the group trained on the feature selected dataset found the optimal hyperparameter configuration faster while also consistently outperforming the group trained on the dataset without feature selection.

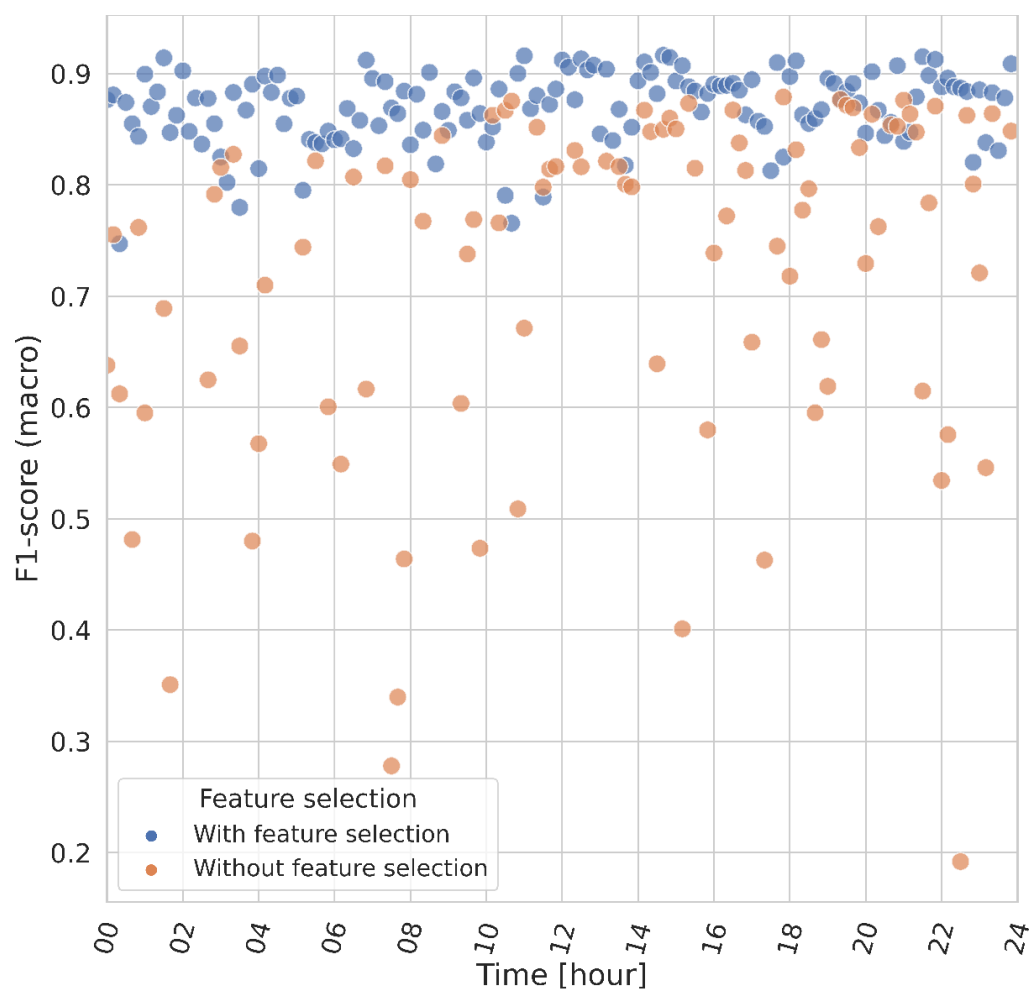


Figure 24 - Searching for optimal hyperparameter configuration for both groups (with and without feature selection).

The box plot in Figure 25 shows the distribution of the F1-scores for both groups. The median line of the box representing the group with feature selection lies outside of the box representing the group without feature selection, indicating that there is likely to be a difference between the two groups. In addition, it is important to highlight the difference in the interquartile ranges of the two groups. This value is noticeably higher for the set with all features includes, indicating that it has a wider distribution.

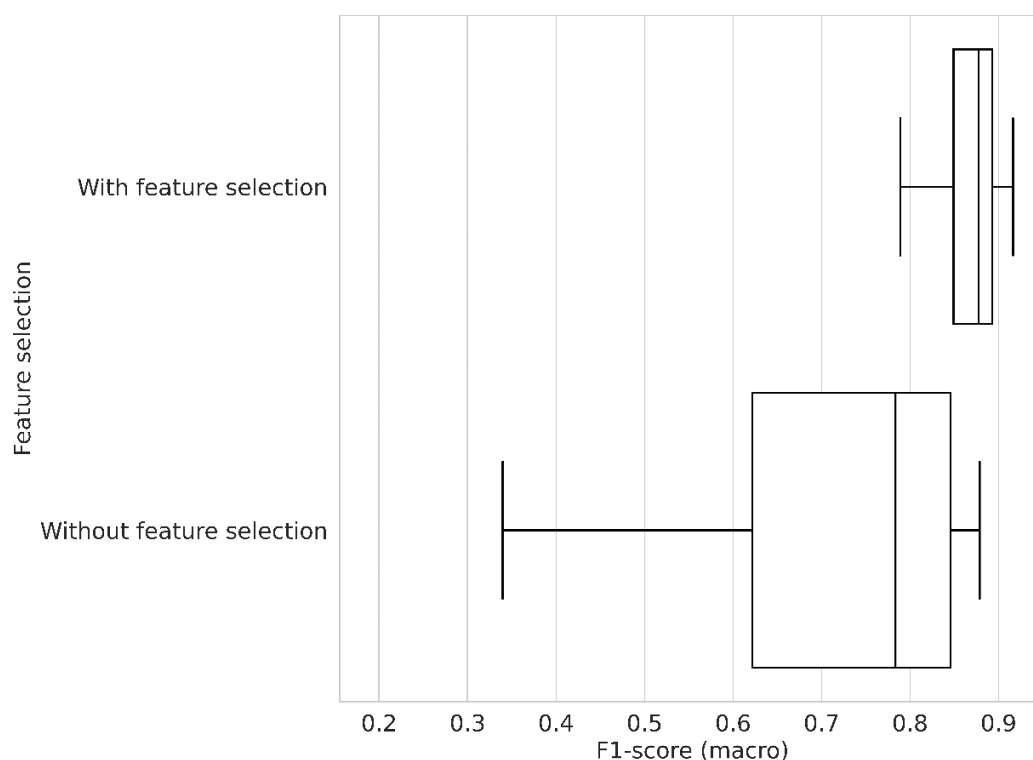


Figure 25 - Box plot described the macro-averaged F1-score of the models trained on datasets with and without feature selection.

The box plot in Figure 26 describes the ROC-AUC value for each of the three classes in this data for both groups (with and without feature selection). Interestingly, class 0 and class 2 exhibited a similar distribution of ROC-AUC scores. This may indicate that the models that used feature selection did not benefit significantly from the reduction in dimensionality in terms of predicting these classes. In contrast, the distribution of ROC-AUC scores in class 1 value shows that the models with feature selection had a different distribution than those with all features.

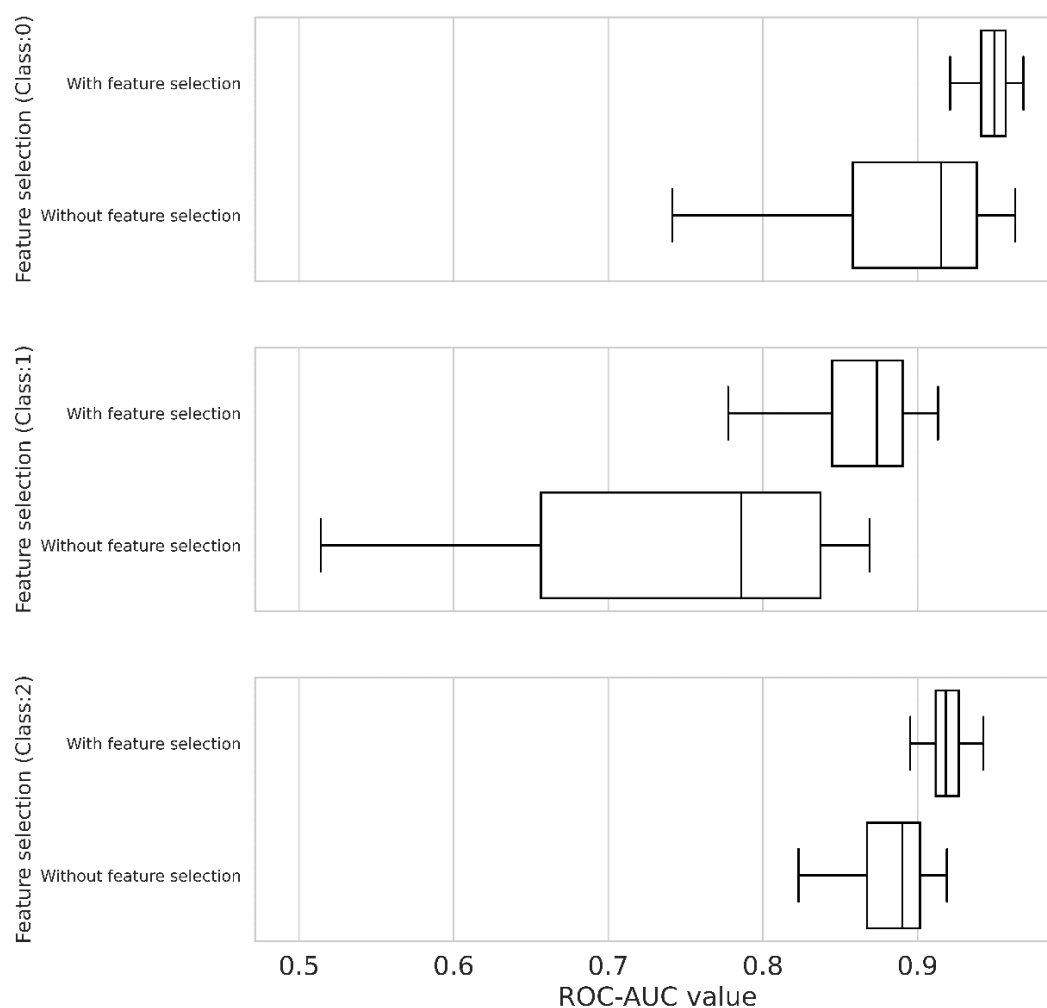


Figure 26 - Box plot displaying the ROC-AUC values of models trained on datasets with and without feature selection.

After evaluating the performance metrics of the trained models, the best model was selected according to its F1-score. Table 6 compares the topology of the best neural network models in each group (with and without feature selection), while Table 7 compares their F1-scores and other complementary performance metrics.

Table 6 - Topology of the neural network.

Layer		With feature selection	Without feature selection
<b>Number of layers of neurons</b>	-	3	5
	1	181	16
	2	231	44
	3	169	185
	4	-	111
	5	-	56



Table 7 - Performance metrics.

	With feature selection	Without feature selection
<b>F1-score</b>	0.916366	0.866118
<b>ROC-AUC (class: 0)</b>	0.938072	0.933283
<b>ROC-AUC (class: 1)</b>	0.891847	0.820592
<b>ROC-AUC (class: 2)</b>	0.916883	0.884144

The confusion matrices of the best models trained on the datasets with and without feature selection are presented in Figures 27 and 28, respectively. The model that utilised feature selection outperformed the model that did not utilise feature selection in terms of specificity and precision across all classes. When adopting feature selection methods, classes 0 and 1 were the most improved in terms of specificity, while classes 1 and 2 were the most improved in terms of precision. In both models, class 0 had the highest accuracy, while class 1 had the lowest. In terms of class 1, the model that utilised feature selection had an accuracy of 0.91, while the model without feature selection had an accuracy of 0.86.

Considering the context in which this model will be used, it is important to minimise the occurrence of false-positives corresponding to class 1, which is the best-case scenario for operational conditions. This type of error would cause the system to fail to send out warnings, meaning no maintenance would be performed. In this context, the model that utilised feature selection had a false positive rate of 0.08 for class 1, while the model that did not utilise feature selection had a false positive rate of 0.12. Hence, adopting feature selection methods results in an improvement of more than 30%.

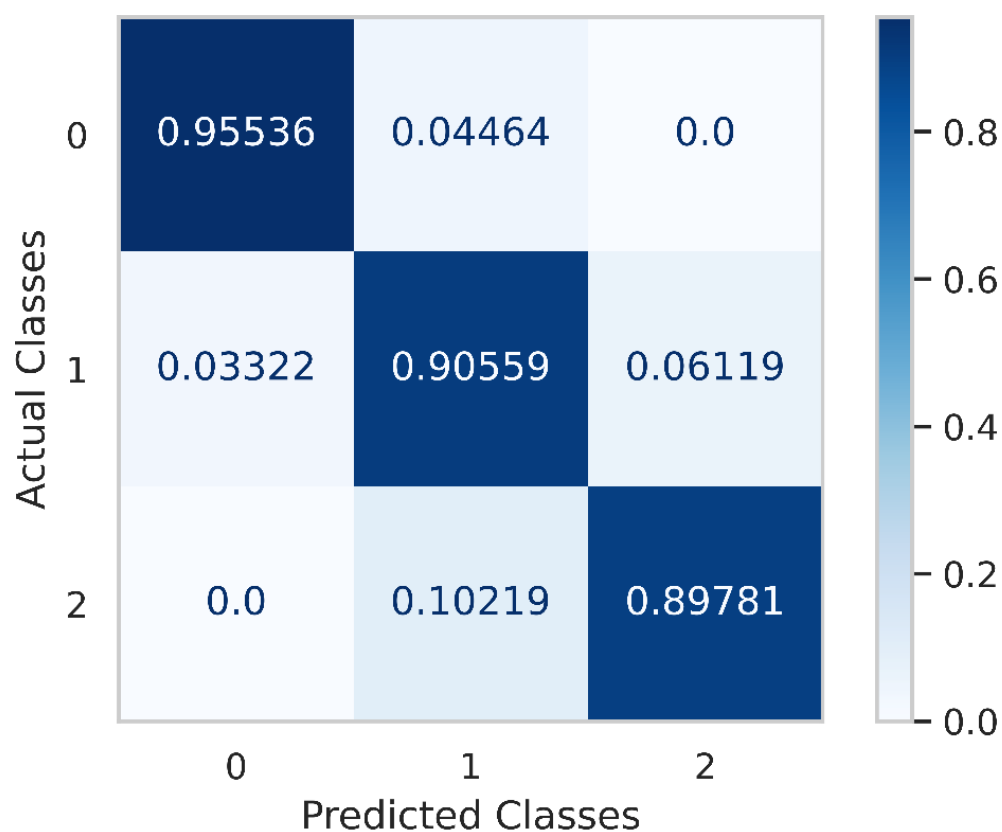


Figure 27 - The normalised confusion matrix of the best model that utilised feature selection.

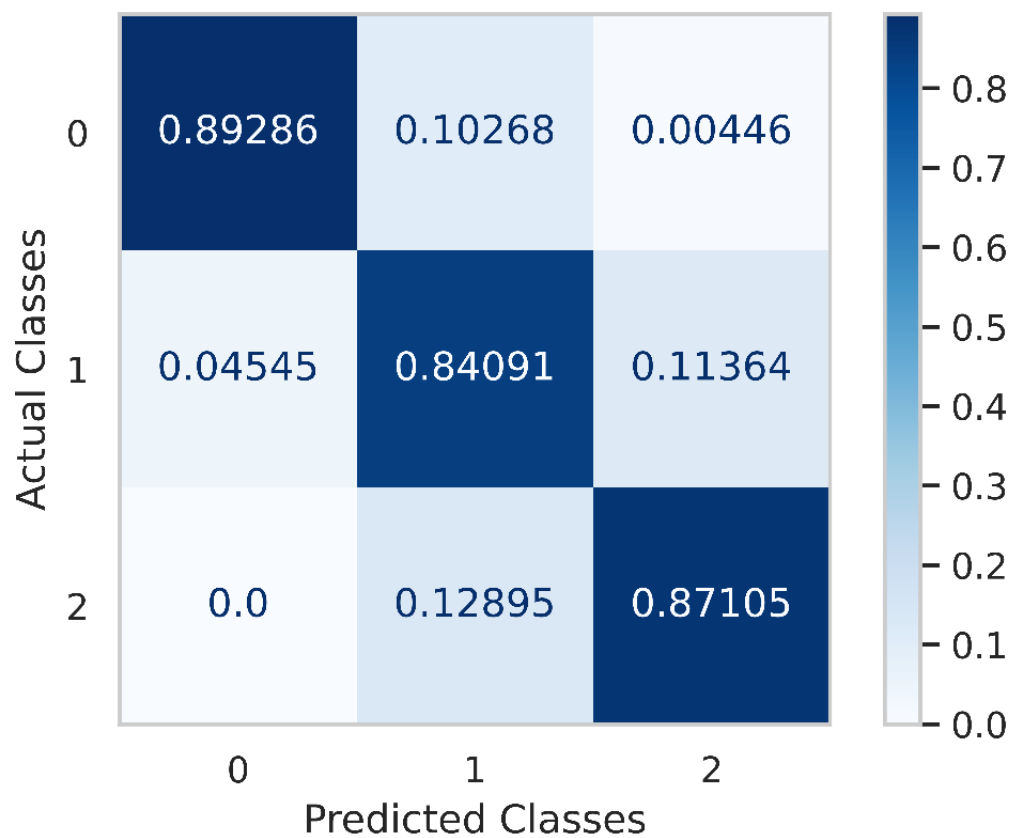


Figure 28 - The normalised confusion matrix for the best model that did not utilise feature selection.

Other authors have also confirmed the higher performance attained in this study with the usage of feature selection techniques. Bagherzadeh et al. (2021) created a model for predicting total nitrogen in the WWTP. The researchers discovered that selecting acceptable features may improve prediction accuracy by around 20%. In their investigation, ANOVA, MI, and RF were tested, with the MI approach yielding the most accurate prediction findings. In addition, the investigators discovered that ANN accuracy decreases dramatically when redundant features are included.

Kimura et al. (2019) developed a novel strategy for selecting features that makes advantage of the random forest's importance metric. This approach used the conventional variable importance measure; an alternative is the permutation-based variable importance measure. The approach was then used to identify important input variables for a model of a real sewage treatment facility. The authors used a leave-one-out cross-validation approach to assess the generalisation errors of random forests with all 42 input variables and those with the 17 variables deemed significant. The generalisation errors of random forests with 42 variables for regression problems were 2,780, 1,162, 1,467, and 3,188, while those with 17 variables were 2,706, 1,115, 1,423 and 3,167. These findings indicate that the prediction performance of the model may be enhanced by eliminating factors determined to be unimportant.

Zounemat-Kermani et al. (2022) assessed the capability of machine learning models (including ANN) to forecast the effluent arsenic concentration of a wastewater treatment facility. In the first case, each of the seven independent variables was considered while building data-driven models. For the second scenario, the forward selection (wrapper approach) k-fold cross-validation technique was used to identify effective explanatory influent parameters. Using the feature selection approach in the second scenario not only made the model architecture simpler and more effective, but it also improved the performance of the built models (e.g., a 7.8% improvement in root mean square error).

Machine learning models with a large number of parameters, such as deep neural networks, are extremely powerful. However, overfitting is a severe issue in such networks. A good diagnostic tool for assessing model behaviour is to compare

the learning curves obtained in both the training and validation datasets. An overfitted model is defined as a classifier with a continuous drop in training loss and minimum validation loss. For both models used in this study, the training and validation losses decreased simultaneously toward closely spaced horizontal asymptotes, indicating that overfitting was not an issue (Figures 29 and 30). It is worth noting the presence of temporary spikes in validation loss for the model with feature selection, which could indicate that the present local minima did not generalise as well as the local minima from the end of the previous epoch (Salesky et al., 2020). However, when the model stabilised, the validation loss only exhibited minor fluctuations and spikes that were similar to the training loss.

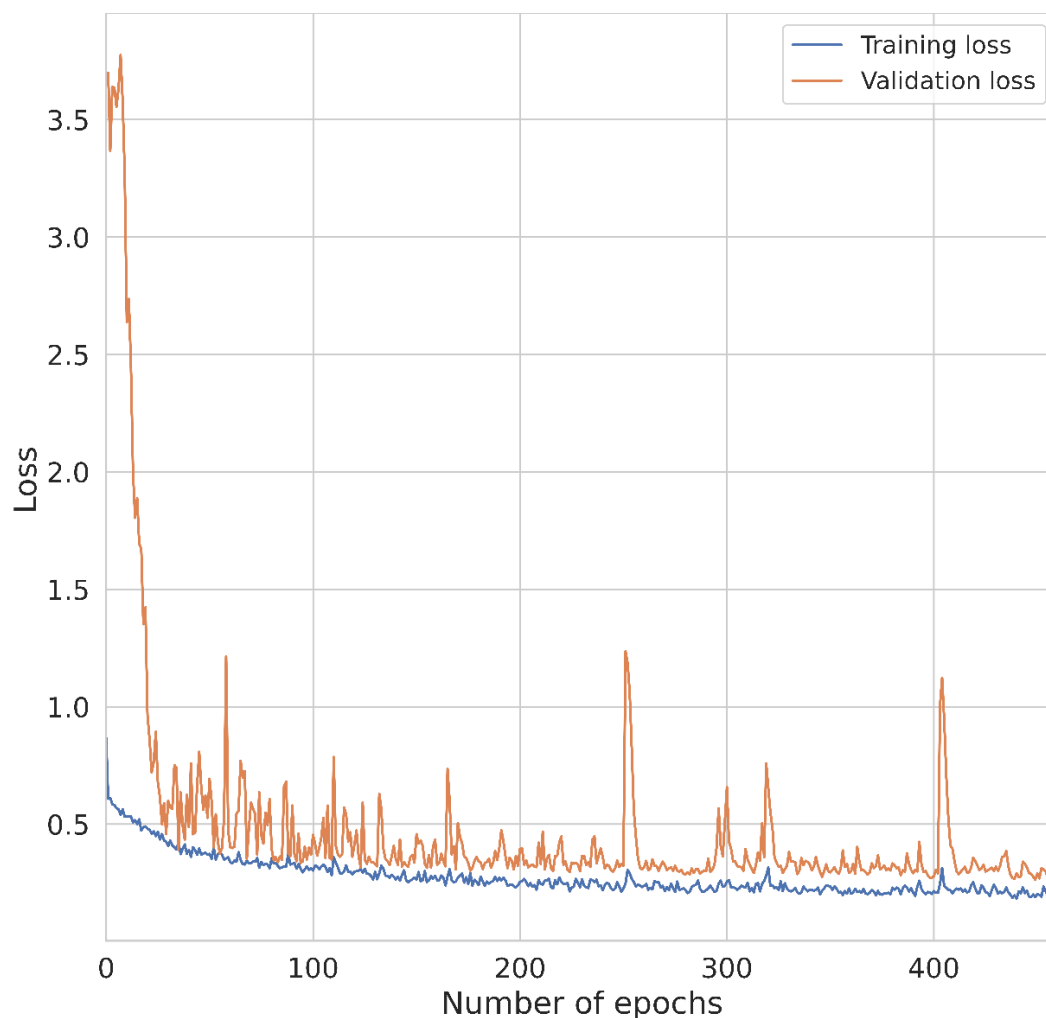


Figure 29 - Training and validation loss during the learning process of the model that utilised feature selection.

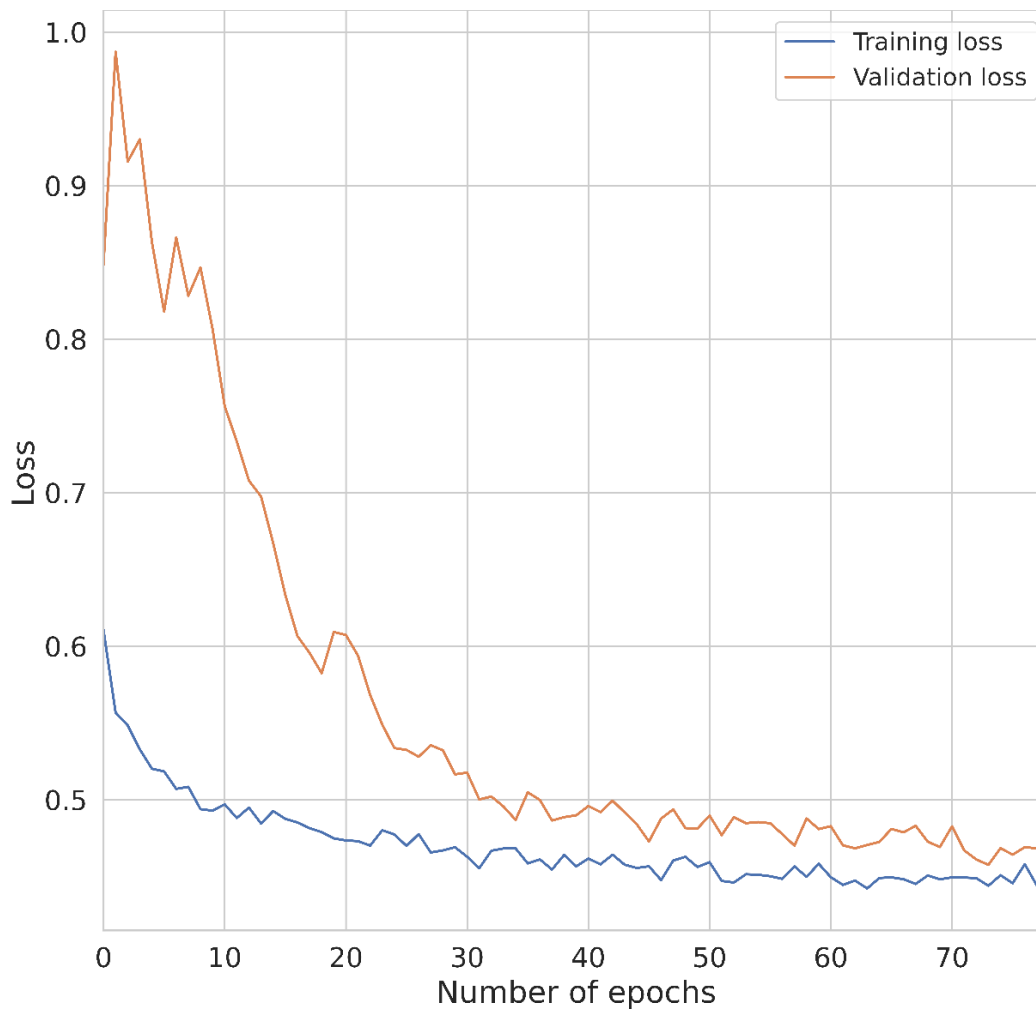


Figure 30 - Training and validation loss during the learning process of the model that did not utilise feature selection.

#### 4.1.4 Conclusions

This paper described a method for the selection of features by considering seven different feature selection techniques. According to the performance metrics, the F1-score of the best model that utilised feature selection was 0.92, which was superior to the F1-score of the model that did not use feature selection, at 0.87. Another improvement that should be highlighted was the false positive rate associated with class 1; this type of error was successfully minimised by the model that utilised feature selection. It is also worth noting that the model that utilised feature selection was highly explainable since it captured the key features of the EC process. This research finds that the dimensionality of the data can be reduced while still maintaining or even boosting the prediction performance of the classifier. For future work, it is possible to implement the model into production in order to assess

real-time monitoring and a more diverse training set including a wider range of industrial effluents.

## 5

### **Computer vision-based monitoring of a bakery industrial effluents treatment by electrocoagulation via a machine learning approach**

The second part of this work proposed a model-based soft sensor that could detect anomalous process behavior using wastewater surface color images from two small-size camera modules. It could provide early warning to plant operators when such conditions were detected. This paper assesses the performance of various methods, including MLP, LSTM, SVM, and XGBoost. The LSTM model outperforms the others in terms of macro average Precision (84.620%), Recall (84.531%), and F1-score (84.499%), but the XGBoost model comes closely in second with Precision (83.922%), Recall (82.272%), and F1-score (83.005%).

This section contains the manuscript version of the article that presents the results of the development of the model.

#### **5.1**

##### **Article Manuscript: Computer vision-based monitoring of a bakery industrial effluents treatment by electrocoagulation via a machine learning approach**

Thiago da Silva Ribeiro<sup>a</sup>, António José dos Santos Rodrigues<sup>b</sup>, Brunno Ferreira dos Santos<sup>a</sup>, and Maurício Leonardo Torem<sup>a</sup>

<sup>a</sup>Department of Chemical and Materials Engineering, Pontifical Catholic University of Rio de Janeiro, Rua Marquês de São Vicente, 225, Gávea, Rio de Janeiro, RJ 22453-900, Brazil

<sup>b</sup>VentilAQUA, S.A., Estrada da Ponte, Lote A. Antanhol, 3040-575, Coimbra, Portugal

#### **5.1.1**

##### **Introduction**

One of the primary global societal concerns is the ongoing reduction in water resource availability. Climate change, urbanization, and population increase are all driving up water demand, which must be successfully handled with current

technologies. As a result of the presence of a variety of newly identified pollutants, the composition of industrial wastewater has become more complicated in recent decades (Crini; Lichtfouse, 2019). Therefore, several novel technologies and techniques are currently being implemented in conventional WWTPs. EC, for instance, is an emerging wastewater treatment method that combines the benefits of coagulation, flotation, and electrochemistry (Das; Sharma; Purkait, 2022).

The bakery industry is one of the world's most important food industries, and its manufacturing size and methods vary greatly (Jerome; Singh; Dwivedi, 2019). While baking effluent normally does not include toxic compounds, it is rich in organic matter (mostly flour and sugar) and oil/grease. Little levels of detergents, yeast, salt, and other food additives are also present (Mohan; Vivekanandhan; Priyadharshini, 2017). The significant daily water consumption in the bakery industry (10–300 thousand gallons, mostly utilised for cleaning operations) is also an environmental issue, especially considering that at least half of this water is disposed as wastewater (Chen et al., 2004).

Due to some distinctive attributes, such as lower operating costs, rapid sedimentation and minimal sludge generation, EC technology has gained considerable momentum in the past years (Vepsäläinen; Sillanpää, 2020). The EC process relies on the formation of coagulant species in situ from a sacrificial anode, typically using either aluminum or iron electrodes. The resulting monomeric and polymeric species have large surface areas and can trap pollutants through electrostatic attraction or surface complexation (Shahedi et al., 2020). The relative importance of EC process variables is difficult to assess, especially because there are dynamic interactions, due to the complexity and variety of influencing factors (Catañeda et al., 2019).

As a consequence of increased operational and management costs, modern WWTPs are being challenged to maintain and improve effluent quality while ensuring efficient operation and cost optimization. The availability of real-time measurements of important process indicators is a fundamental need for achieving these objectives. These indicators are required to effectively monitor plant operation, process performance and economic efficiency, all of which have immediate implications for environmental compliance, safety, management planning and profitability (Zhang; Tooker; Mueller, 2020). The capacity to access



indicators in real time is critical for implementing advanced process control and optimization methods in WWTPs.

A fault is defined as a divergence from a predefined range of normal values for a parameter (Park; Fan; Hsu, 2020). Data-driven and model-based fault detection methods are two examples of fault detection techniques. A model-based approach is difficult and time consuming to develop due to the intricacy of obtaining first principle models of complex processes, such as EC. Data-driven techniques, on the other hand, are created entirely from historical and online data, with no need for a phenomenological model (Md Nor; Che Hassan; Hussain, 2020).

Data-driven methods have attracted great interest in fault detection, especially with the rise of big data. Industry 4.0 has accelerated the development of associated technologies such as the Internet of Things, wireless sensor networks and cloud computing (Bousdekis et al., 2021). Simultaneously, data gathering and storage become more accessible, hastening the emergence of the industrial big data era (Bousdekis et al., 2021). As a result, effective measurement is required, therefore, process sensors have progressed from simple mechanical indicators to Industry 4.0 smart sensors. The smart sensor is self-calibrating and self-optimizing. It is simple to integrate into the process environment and operates independently (Kalsoom et al., 2020).

The availability of parameter values acquired by on-line analysis is always associated with high investment and maintenance costs, and results obtained through off-line analysis have time-delayed responses, making real-time monitoring problematic (Newhart et al., 2019). The vast amount of process data that is frequently measured in WWTPs makes data-driven modeling an appealing soft-sensor design option (Dairi et al., 2019). In comparison to traditional methods, data-driven methods have demonstrated higher performance in real-time operation and lower false alarm rate (Ching; So; Morck, 2021).

Soft sensing uses existing data and information to estimate or forecast physical quantities or quality in industrial operations. Soft sensors are software-based or embedded, unlike physical sensors. Soft sensors are digital representations of hardware sensing devices in virtual space, although in many situations there is no real equivalent (Jiang et al., 2021).

Sensor measurements drift with time, even with routine maintenance and calibration, and the drift can differ amongst sensors. Soft sensors should be built on sensors that are robust and easy to operate and that can be validated effectively, in order to be appealing for industrial applications (Jiang et al., 2021).

The computer vision-based system is relatively low-cost, flexible and capable of high-resolution measurements (Capitán-Vallvey et al., 2015). As a result of the high-quality spatial information provided by these systems, each pixel of the image effectively works as a sensor. Model deployment can be obtained with computer vision using cost-efficient parallel hardware, extract more important and relevant features and improve image interpretation with cutting-edge algorithms, and develop a model with good generalization and high accuracy using big data (Javaid et al., 2022).

Images can be used to extract a variety of data (color, texture and morphology). Color components have been employed by many studies to extract color features related to physical or chemical parameters (Damirchi; Heidari, 2018; Santiago; Sevilla, 2022). An abstract mathematical model called as color space is utilized in color image processing to characterize colors in terms of intensity values. A variety of color spaces are available, including RGB and HSV, for various types of applications (Phuangsaibai; Jakmunee; Kittiwachana, 2021).

In RGB color space, color information (chrominance) and intensity information (luminance) are combined. Therefore, the RGB color space is often not suitable for color-based image segmentation and analysis. In computer graphics, scientific computing, and other domains, the HSV color space is extensively employed (Chernov; Alander; Bochko, 2015). When utilized in optical sensing, the most interesting characteristic of HSV color space is the ability of color representation in a single parameter, H, avoiding redundant coordinate information (Lv et al., 2021).

H, S, and V are the three attributes used to describe HSV colors. The color red, green, blue, and yellow with a spectrum range of 0–360 degrees is referred to as H. V relates to the brightness of color and provides the achromatic idea of the color, whilst S refers to the purity of the color and takes the value of 0–100 percent.

Color recognition has become widely employed in different industrial detection and automatic control fields as modern industrial production moves

toward a high-speed, automated direction (Fernandes et al., 2020). Color sensor detects color by comparing the object color to the reference color, and then outputs the detection results if they are consistent within a defined error range.

A model was proposed in this study that could detect anomalous process behavior using wastewater surface color images. This paper assesses the performance of various methods, including ANNs, SVMs and ensemble machine learning algorithms.

A typical MLP has an input layer that accepts data, an output layer that predicts model output, and one or more hidden layers that learn data patterns. An activation function receives input weights and bias. A MLP is a feedforward ANN with differentiable neuron activation functions that can map an input space to an output space statically (Lecun; Bengio; Hinton, 2015).

Recurrent neural networks provide real-time contextual information. Recurrent neural networks integrate feedback connections between nodes and layers, enabling them to handle any length input sequence. LSTM is a recurrent neural system built to tackle exploding/vanishing gradients. Cell, input gate, output gate, and forget gate make up an LSTM unit (Alom et al., 2019).

Using SVMs, the margin between decision boundaries is maximised in a high-dimensional space called the feature space. This classification method enhances generalisation by reducing training data classification errors (Chauhan; Dahiya; Sharma, 2019).

Gradient boosting improves classification performance in supervised learning. XGBoost uses decision tree boosting. Boosting is an ensemble learning approach that develops multiple models sequentially, each seeking to remedy defects in the previous model. Tree boosting adds a decision tree to each extra model. In this boosting process, gradient descent minimises loss (Ferreira; Figueiredo, 2012).

### **5.1.2 Methods**

The current study's methodology can be divided into three parts: image acquisition and dataset creation methods, feature extraction, and modelling procedures. The images of the wastewater surface were obtained in a WWTP setup

designed for image acquisition. True color images, also known as RGB images, were used to extract HSV color histogram-based features. For the classification of images into distinct operational conditions, four machine learning algorithms were employed.

#### 5.1.2.1

##### Image acquisition and dataset creation methods

In this study, data were collected from a decentralised EC WWTP that was supplied and commissioned by VentilAQUA (Coimbra, Portugal). The location of the unit was a bakery in Slovenia.

VentilAQUA's VABEC® technology is an EC continuous-flow system, multi-electrode cell that is composed of electrodes made of materials that are suitable for oxidation and coagulation, with a modular configuration and an internal geometry designed for optimum efficiency. Following the chemical reaction phase, a flotation procedure is employed for solid-liquid separation. The DAF unit is a pre-assembled, compact system built using VentilAQUA's VAMEF® technology.

The images of the wastewater surface were taken in a dedicated image acquisition setup. Throughout the experiment, the illumination is maintained to ensure that all images are acquired in the same setting.

The camera used in this study was the ESP32-CAM, a small-size camera module with deep sleep current and a minimum of 6mA that can run independently as a system, measuring only 27 x 40.5 x 4.5mm. The ESP32-CAM is constituted of an OV2640 camera module that can capture images with resolutions up to 1600 × 1200 pixels. A low-power 32-bit MCU with a clock frequency up to 240MHz, 520KB internal SRAM memory, and 4MB external PSRAM memory are included.

Two ESP32-CAM modules were employed in this study, as shown in Figure 31, one in the EC unit's inlet and the other in the lifting tank just after the EC process. The ESP32-CAM captured JPEG images of the wastewater surface and used microSD card module for automated data storage.

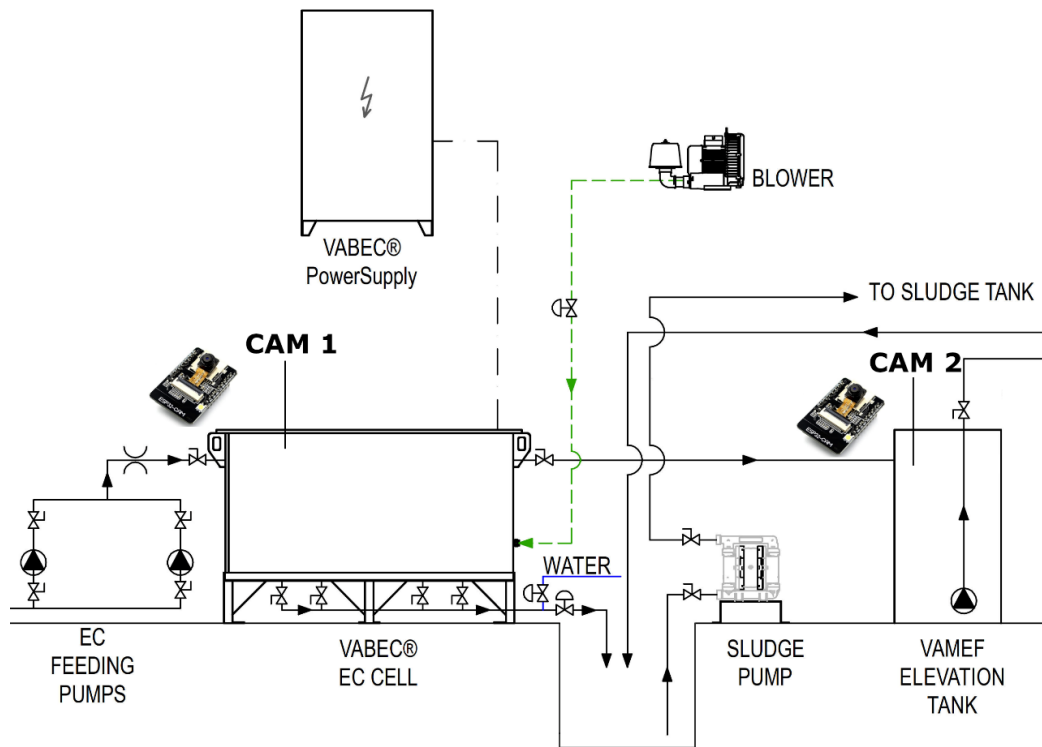


Figure 31 - ESP32-CAM modules locations in the WWTP.

The target variables were three operational modes based on the effluent clarification and the reaction sludge. We treated the problem as a classification task, with three classes based on expert knowledge as follows:

- Class 0: Not clarified, showing turbidity;
- Class 1: Clarified, showing low turbidity;
- Class 2: Clarified, although the system had an excessive electrode and energy consumption.

Original dataset consists of 1207 records with three outcomes which are 34% class 0 samples, 40% class 1 samples, and 26% class 2 samples.

The entire dataset is divided into 70 percent training and validation dataset and 30 percent test dataset. The training and validation dataset is used for the stratified 5-fold hyperparameter optimization and algorithm comparison step. Before training the models, the values of the numeric input variables were standardised by subtracting the mean and scaling it to the unit variance.

### 5.1.2.2 Feature extraction

The H, S, and V color channels were used as color features in this paper. However, using all of the information at each pixel was neither practicable or necessary, since the mean values can be used to represent each sample's color distribution. Therefore, the mean values of the H, S, and V channels over all pixels in an image were chosen as the features of each sample. Figures 32, 33 and 34 present the comparative histograms between the H, S and V color channels both at the EC unit's inlet and after the EC process, respectively.

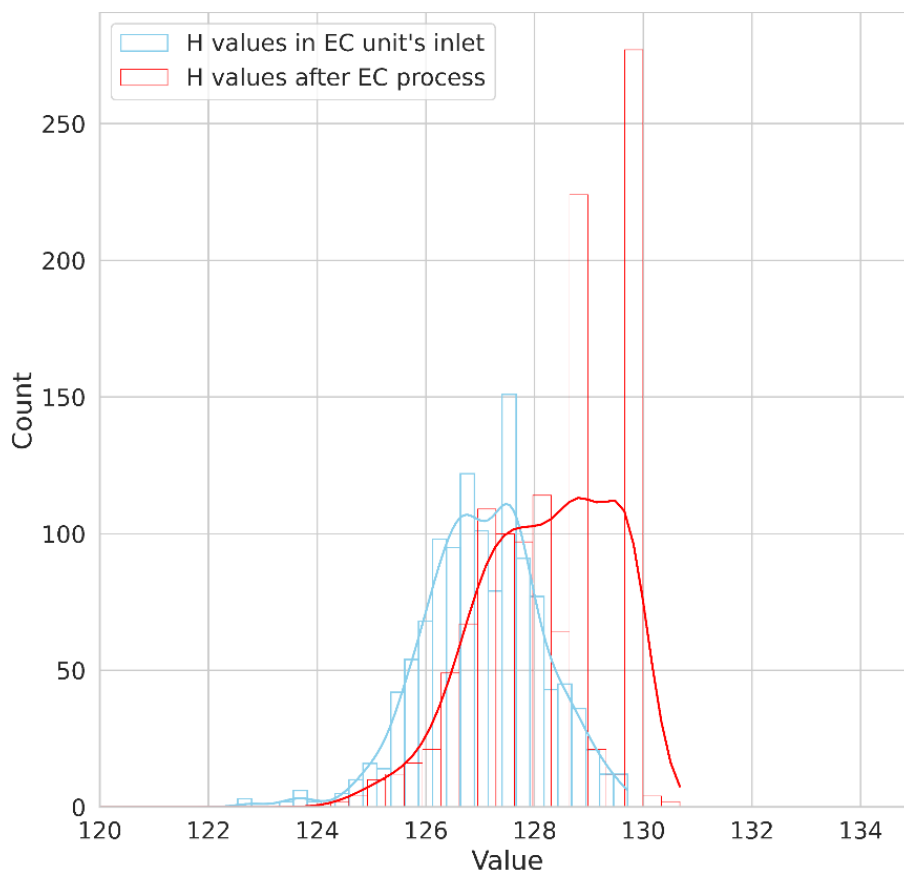


Figure 32 - Histograms of H values in the EC unit's inlet and after EC process.

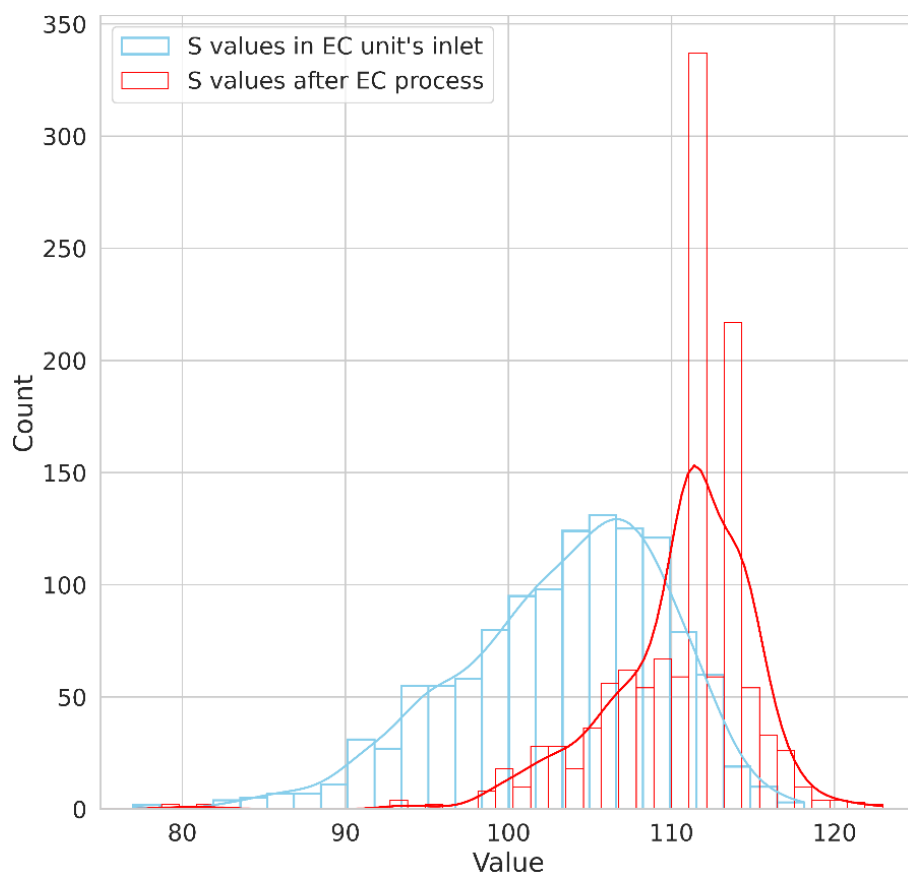


Figure 33 - Histograms of S values in the EC unit's inlet and after EC process.

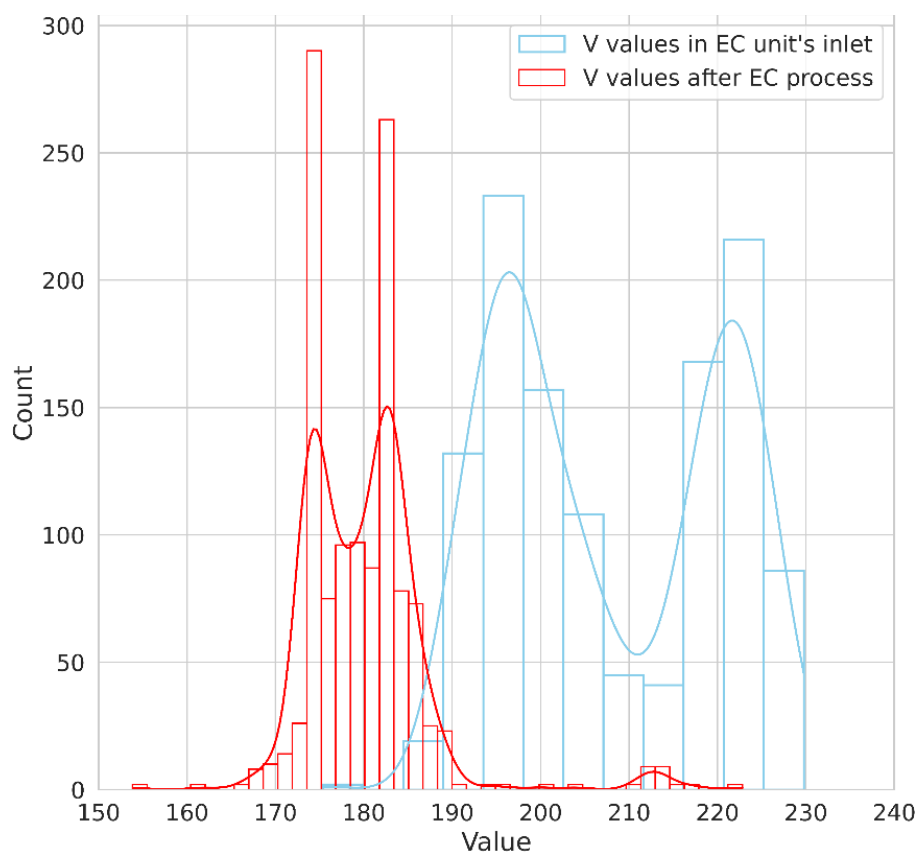


Figure 34 - Histograms of V values in the EC unit's inlet and after EC process.

The images were acquired in the RGB color space. The transformation from RGB color space to HSV color space is nonlinear, and was performed using the `CV_BGR2HSV` method present in the OpenCV library (Open Source Computer Vision Library). This method takes an RGB image to convert as an  $m$ -by- $n$ -by-3 numeric array as its input parameter. The third dimension of RGB specifies the corresponding red, green, and blue intensities of each pixel. The output parameter is an HSV image returned as an  $m$ -by- $n$ -by-3 numeric array with values between 0 and 1. The third dimension of HSV specifies, for each pixel, the hue, saturation, and value, correspondingly.

The mean approach was chosen to extract the features for each sample by representing each image with the average values of H, S, and V over all pixels. The 6 values were used as input variables in the models since two ESP32-CAM modules were employed in this study.

### 5.1.3 Model

This paper adopted a BO approach to automate the selection of optimal hyperparameter values for the models, and the F1-score with a macro average was chosen as the optimisation target. BO can be described using Equation 42, where  $X$  is the space of possible hyperparameters and the objective function  $f$  aims to minimise validation error. This technique employs a Bayes rule-based surrogate probability model, in which the values of the next iteration are determined based on the results of prior iterations (Snoek; Larochelle; Adams, 2012).

$$x^* = \operatorname{argmin}_{x \in X} f(x) \quad (42)$$

The implementation of BO was based on the open-source *wandb* client provided by Weights & Biases, but only after manually picking a reasonable set of hyperparameter values (Tables 8–11). Four distinct classification algorithms were chosen for comparison: MLP, LSTM, SVM, and XGBoost. The models were trained on an NVIDIA Tesla P100-PCIE-16GB.



Table 8 - Hyperparameters tested (MLP).

Hyperparameter	Tested settings
Epoch	min:1, max:500 (discrete uniform distribution on integers)
Activation function	linear, relu, sigmoid, tanh, selu, elu
# hidden layers	min:1, max:5 (discrete uniform distribution on integers)
# neurons per hidden layers	min:1, max:250 (discrete uniform distribution on integers)
Loss function	categorical_crossentropy, kl_divergence, poisson
Learning rate	min:0.00001, max:0.01 (continuous uniform distribution)
Batch size	32, 64, 128, 256, 512
Dropout	0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9
Optimizer function	adam, SGD, adamax, nadam
Kernel initializer	he_uniform, glorot_uniform, lecun_uniform

TensorFlow is a Google-developed open-source software framework for numerical computation based on the dataflow programming paradigm that can be utilized in machine learning applications. TensorFlow was employed to build MLP models in this paper.

Table 9 - Hyperparameters tested (LSTM).

Hyperparameter	Tested settings
Epoch	min:1, max:500 (discrete uniform distribution on integers)
# hidden layers	min:1, max:5 (discrete uniform distribution on integers)
# neurons per hidden layers	min:1, max:250 (discrete uniform distribution on integers)
Loss function	categorical_crossentropy, kl_divergence, poisson
Learning rate	min:0.00001, max:0.01 (continuous uniform distribution)
Batch size	32, 64, 128, 256, 512
Dropout	0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9
Optimizer function	adam, SGD, adamax, nadam
Kernel initializer	he_uniform, glorot_uniform, lecun_uniform
Time steps	min:1, max:50 (discrete uniform distribution on integers)

CuDNNLSTM is an LSTM variation that can only be used on GPU devices with the Tensorflow-gpu backend. When compared to LSTM, CuDNNLSTMs have proven to be substantially faster. CuDNNLSTM implements the tanh activation function by default. Therefore, CuDNNLSTM was used to build the LSTM layers in this paper.

Table 10 - Hyperparameters tested (SVM).

Hyperparameter	Tested settings
$C$	min:0.1, max:100 (discrete uniform distribution)
$\gamma$ (gamma)	min:0.0001, max:10 (discrete uniform distribution)
$d$ (degree)	min:1, max:6 (discrete uniform distribution on integers)
Decision function shape	one vs the rest, one vs one
Kernel function	polynomial, radial basis function, sigmoid

For the Python programming language, scikit-learn is an open source machine learning package. The library was created to work with the NumPy and SciPy libraries, which are used to create a set of mathematical and scientific packages. Therefore, the scikit-learn library was used to build an SVM classifier in this paper.

Table 11 - Hyperparameters tested (XGBoost).

Hyperparameter	Tested settings
Booster	gbtree, gblinear
Maximum depth of a tree	min:1, max:12 (discrete uniform distribution on integers)
Boosting learning rate	min:0.0001, max:1 (discrete uniform distribution)
Subsample ratio of the training instances	1, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1
Number of boosting iterations	min:10, max:5000 (discrete uniform distribution on integers)
Loss function	softmax, softmax

This paper employs the XGBoost library to build the XGBoost classifier, which is an improved distributed gradient boosting algorithm with great efficiency, flexibility, and portability.

The algorithms were compared by analyzing the F1-score, precision, and recall of all trained models. The following step is to evaluate the performance of the optimised models.

#### 5.1.3.1.1

##### Model performance metrics

The F1-score with a macro average was the primary metric by which the performance of the classification models described in this study was evaluated. The F1-score evaluates the performance of a classification model by taking the harmonic mean of the precision and the recall of the classifier. The F1-score formula can be read as a weighted average of precision and recall that ranges from 0 to 1, where 0 is the worst score, while 1 is the best score. Precision and recall have the same relative contribution to the F1-score, so the harmonic mean can be used to discover the ideal trade-off between the two metrics. This metric is unaffected by class size since classes of varying sizes are equally weighted in the numerator, which means that the impact of the largest classes is equal to that of the smallest (Grandini; Bagli; Visani, 2020).

The F1-score with a macro average (Equation 43) combines the precision (PPV; Equation 44) and the recall (TPR; Equation 45) of the model to assess its ability to accurately classify data.

$$F1 = \frac{2}{|C|} \sum_{i=1}^{|C|} \frac{(TPR_i * PPV_i)}{(TPR_i + PPV_i)} \quad (43)$$

$$PPV = \frac{TP}{TP + FP} \quad (44)$$

$$TPR = \frac{TP}{TP + FN} \quad (45)$$

In the above Equations, TP refers to the number of true positive predictions compared to the test data, FP refers to the number of false positive predictions, FN refers to the number of false negative predictions, and C is the number of classes.

After selecting the best model for each algorithm, the test results are computed using the 30% test dataset, which has been untouched since the beginning of the process. Figure 35 illustrates the methodology's overall computational architecture.

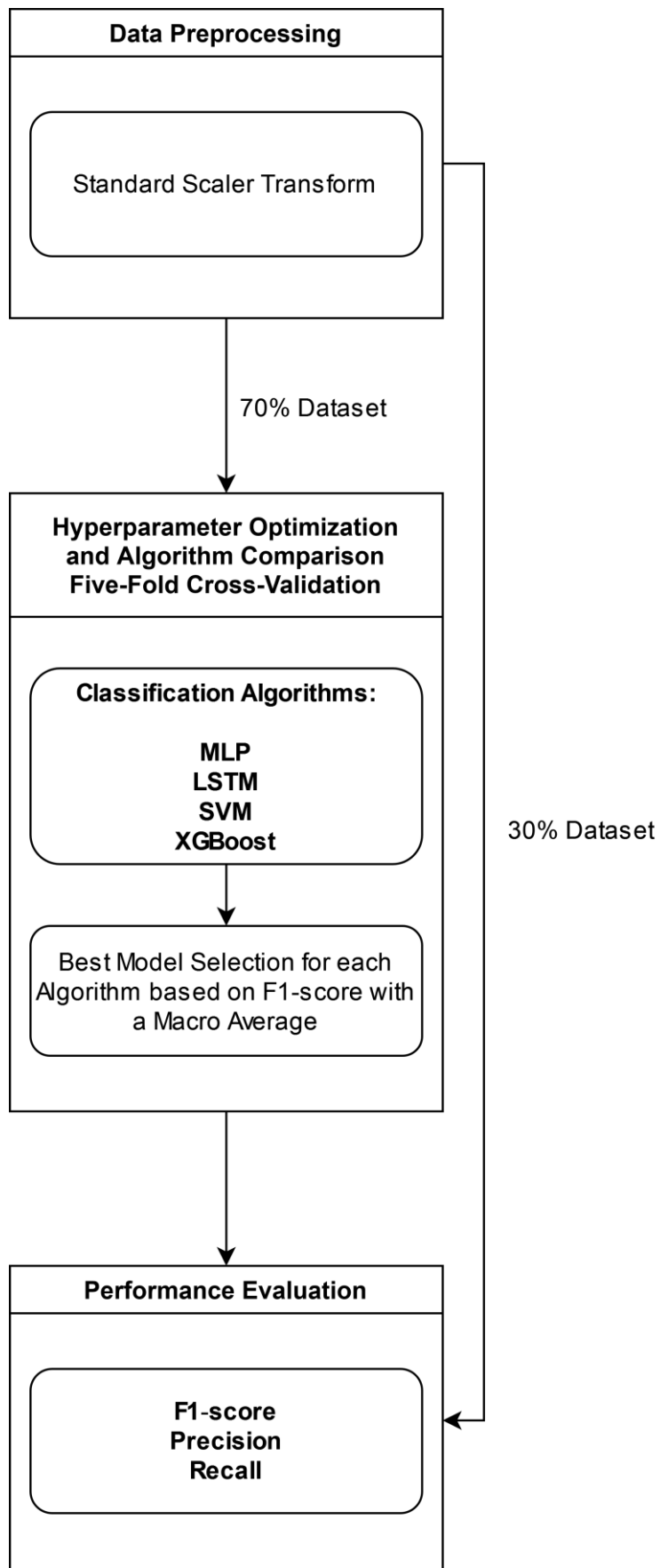


Figure 35 - Overall computational flowchart.

### 5.1.4 Results and discussion

#### 5.1.4.1 Comparison of machine learning algorithms

Figure 36 presents the progress of hyperparameter optimization. The scatter plot reveals that the optimization algorithm finds optimal hyperparameter values in the search space for each algorithm. The SVM and XGBoost algorithms require substantially less computation time to optimise their hyperparameters. This discovery is supported by the fact that these algorithms require less time to train than deep learning techniques such as deep MLP and LSTM (Figure 37).

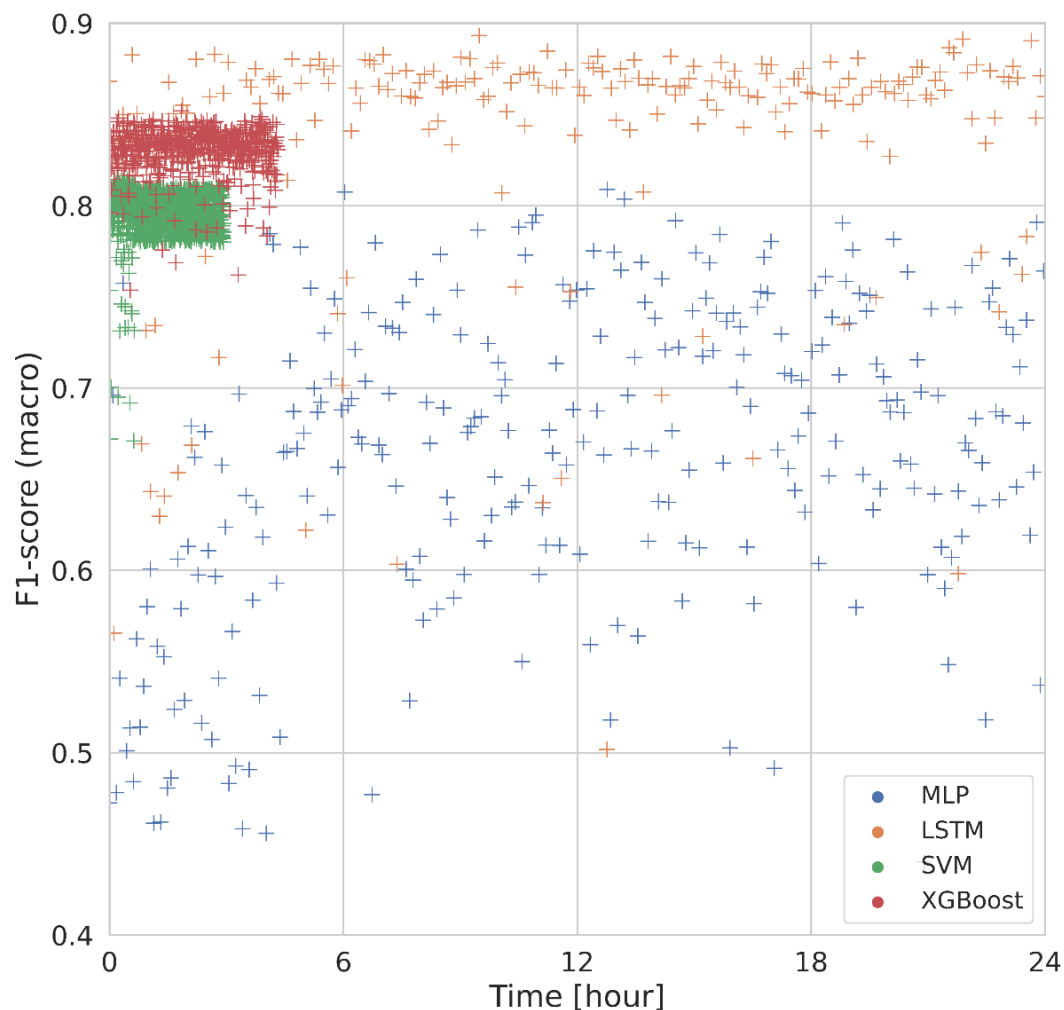


Figure 36 - Searching for optimal hyperparameter configuration on an NVIDIA Tesla P100-PCI-E-16GB for each algorithm.

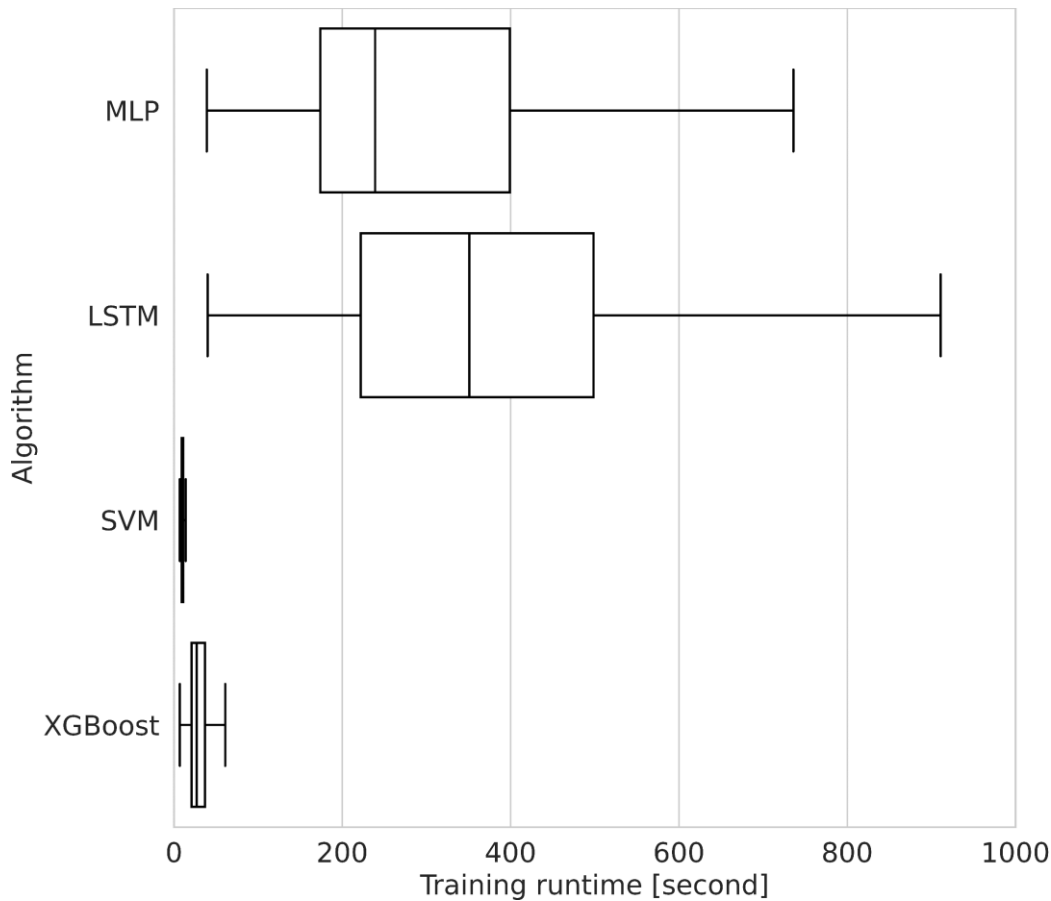


Figure 37 - Training time on an NVIDIA Tesla P100-PCIE-16GB for each algorithm.

Figures 38–40 provide an initial assessment of how well each machine learning algorithm performs based on the stratified 5-fold cross-validation metrics.

Figure 38 shows that all algorithms produced similar results for the metrics evaluated for class 0 (Not clarified, showing turbidity). However, it is worth noting that the LSTM algorithm has the highest Recall value, while the SVM algorithm has the lowest. The Recall measure for this class is meaningful since it indicates the occurrence of false negatives, or occasions in which the system may be diagnosed as faultless. It is also notable that the MLP algorithm has the lowest precision for class 0, indicating a larger likelihood of false positives.

Figure 39 illustrates the metrics evaluated for class 1 (Clarified, showing low turbidity). The precision metric for this class is extremely important for this paper since it reflects the occurrence of false positives, or scenarios in which the system would be diagnosed as faultless. This classification would make it impossible to send alerts to the maintenance team, even if the system was not functioning normally. This type of error would result in significant financial expenses as well as threats to human health and the environment. Therefore, the LSTM algorithm

achieved the highest value for this measure, followed by the algorithms XGBoost, MLP, and SVM.

Figure 40 presents the values of the metrics evaluated for class 2 (Clarified, although the system had an excessive electrode and energy consumption). This is a challenging class, as it is not directly tied to a system failure, but to an operating condition outside the optimal region, resulting in inefficiency in operating expenses. Once again, the LSTM algorithm outperformed in both the Recall and Precision metrics. The results obtained by the XGBoost algorithm are also notable.

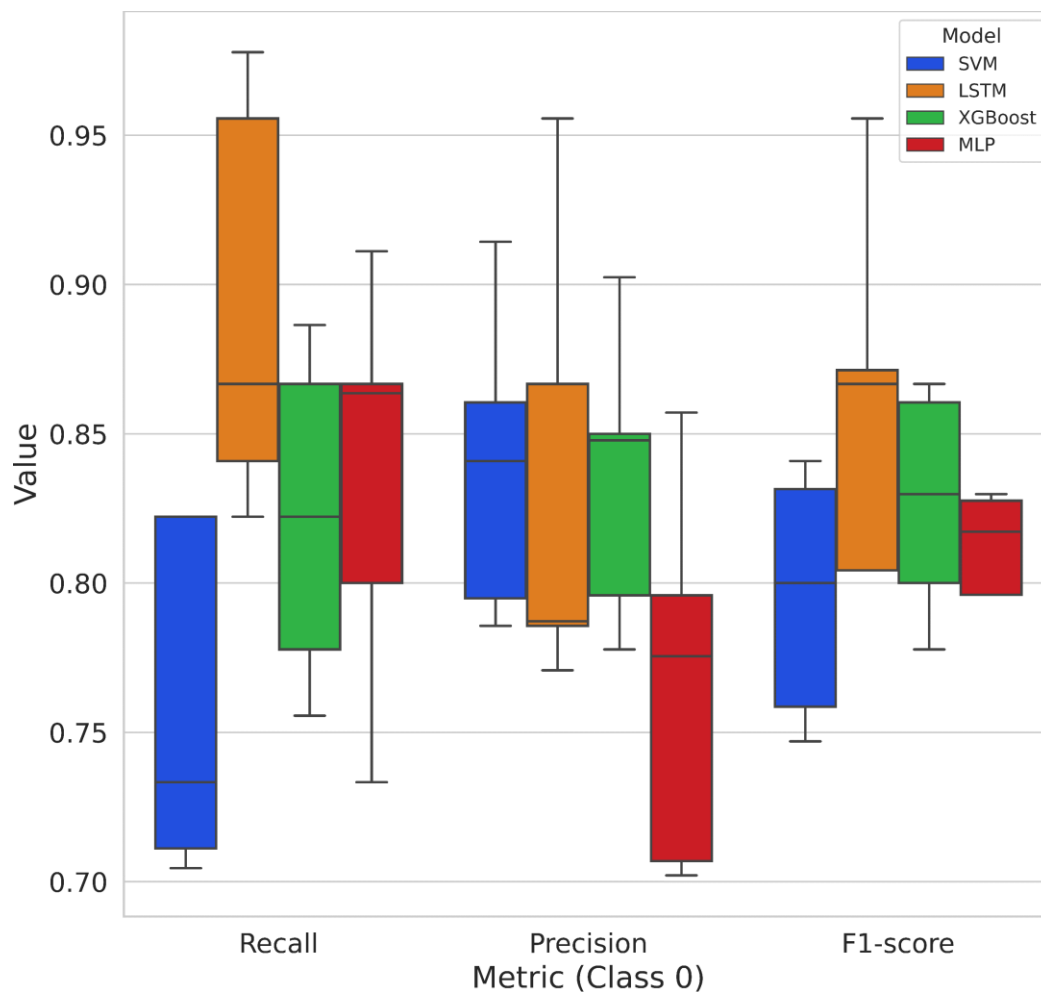


Figure 38 - Boxplot of 5-fold cross-validation evaluation metrics using four distinct machine learning algorithms (Class 0: Not clarified, showing turbidity).

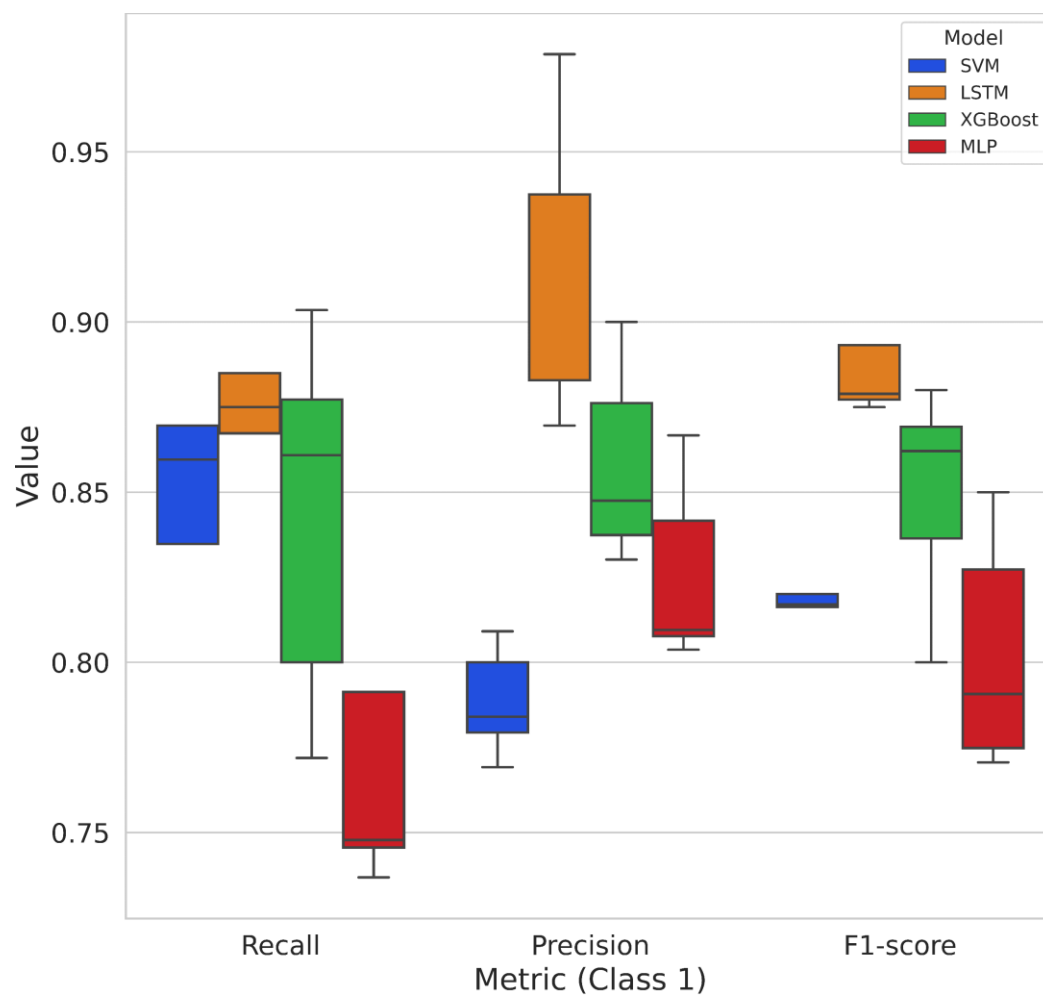


Figure 39 - Boxplot of 5-fold cross-validation evaluation metrics using four distinct machine learning algorithms (Class 1: Clarified, showing low turbidity).



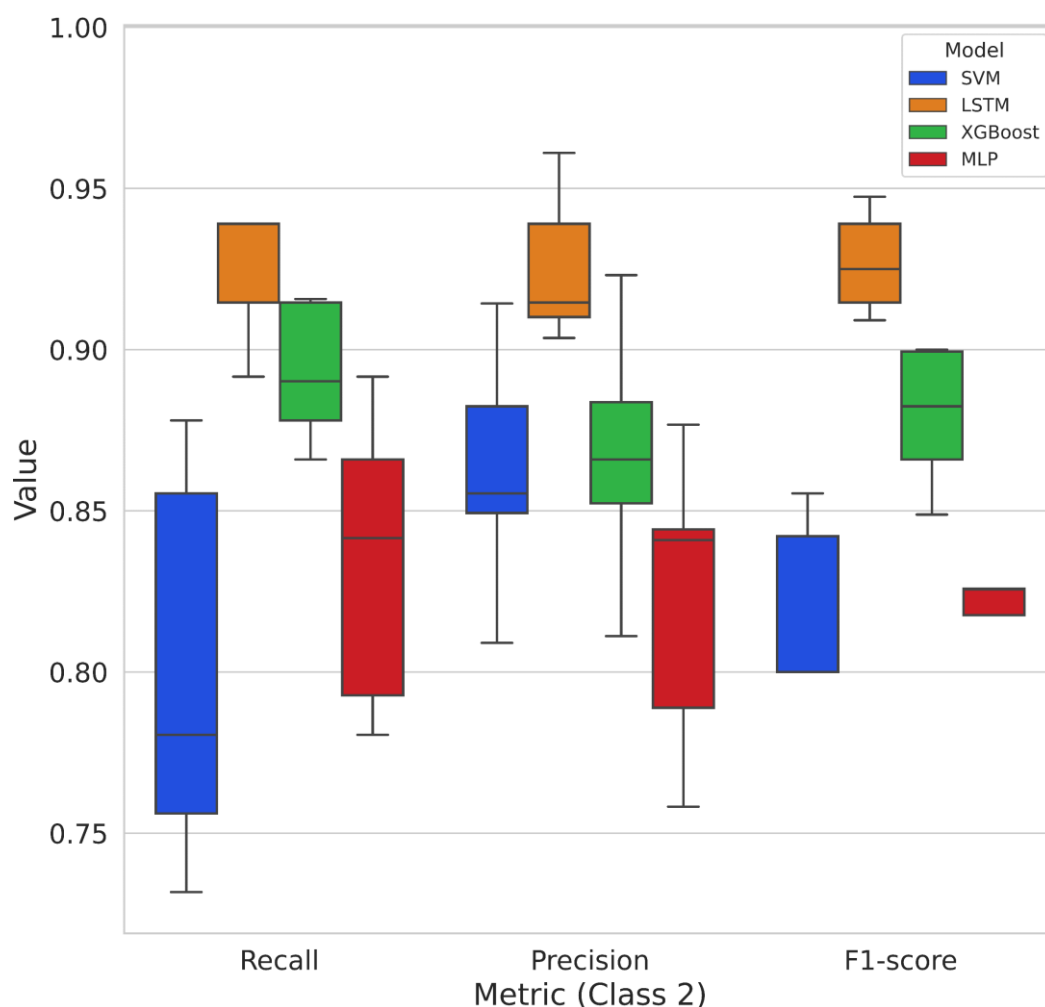


Figure 40 - Boxplot of 5-fold cross-validation evaluation metrics using four distinct machine learning algorithms (Class 2: Clarified, although the system had an excessive electrode and energy consumption).

#### 5.1.4.2 Hyperparameter analysis

This study evaluated a range of different hyperparameters in order to identify values that provide the highest performance for each of the four algorithms. The models were examined for correlations between results and hyperparameters. The optimal configuration was determined by assessing the most influential hyperparameters across all algorithms.

##### ○ MLP

The choice of hidden layers and hidden neurons has a considerable impact on the performance of deep neural networks. The optimal number of hidden layers and hidden neurons is determined by factors such as problem complexity, number of

input and output neurons, and number of training samples. Figure 41 shows that performance improves with 3 to 4 hidden layers and 400 to 600 hidden neurons.

Figure 41 also reveals that the Nadam optimizer was used in the majority of the best models. When compared to other optimizers, the Nadam optimizer performs better. Adam (Adaptive Moment Estimation) and NAG (Nesterov accelerated gradient) are combined in Nadam (Nesterov-accelerated Adaptive Moment Estimation). This enables for more precise gradient-direction steps by updating the parameters with momentum before calculating gradient.

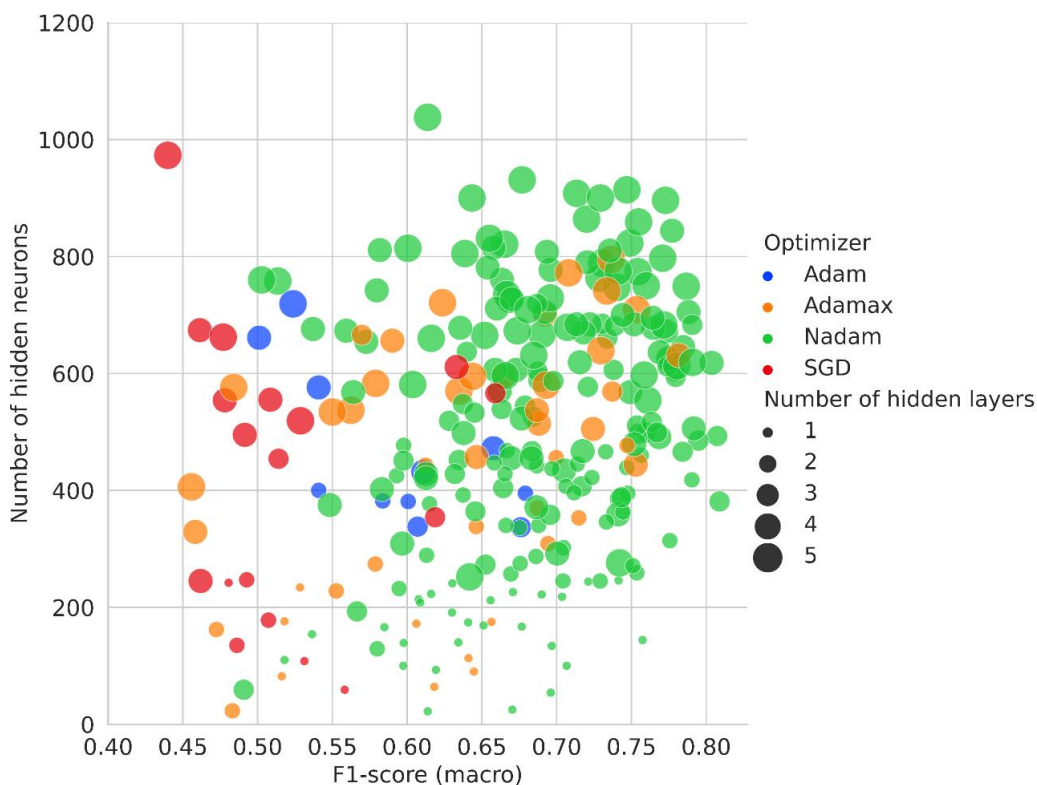


Figure 41. Hyperparameters search results for the MLP algorithm.

Therefore, the best model has the following hyperparameter configuration (Table 12):

Table 12 - Best MLP model hyperparameter configuration.

Hyperparameter	Tested settings
Epoch	467
Activation function	Hidden layer 1: tanh; Hidden layer 2: elu; Hidden layer 3: tanh
# hidden layers	3
# neurons per hidden layers	Hidden layer 1: 56; Hidden layer 2: 201; Hidden layer 3: 124
Loss function	kl_divergence
Learning rate	0.008974
Batch size	256
Dropout	Hidden layer 1: 0.2; Hidden layer 2: 0.1; Hidden layer 3: 0.1
Optimizer function	nadam
Kernel initializer	Hidden layer 1: lecun_uniform; Hidden layer 2: lecun_uniform; Hidden layer 3: lecun_uniform

- LSTM

Figure 42 shows that performance improves with 3 hidden layers and 300 to 500 hidden neurons. Because LSTM has a large number of parameters to learn when compared to other algorithms, increasing the number of hidden neurons and hidden layers has a greater impact on training time and, eventually, generalisation capability.

As loss functions, the best models use Kullback–Leibler (KL) divergence (Figure 42). The KL divergence is a widely used measure of distance across distributions that has both computational and theoretical advantages and leads to a convex optimization problem.

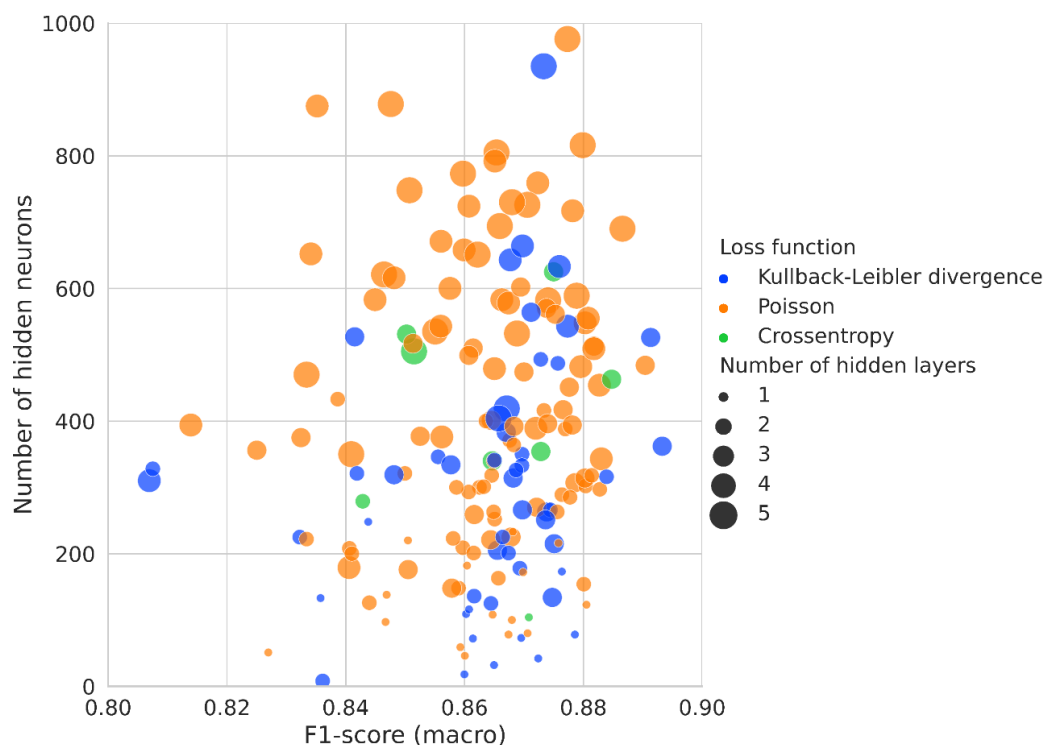


Figure 42 - Hyperparameters search results for the LSTM algorithm.

Therefore, the best model has the following hyperparameter configuration (Table 13):

Table 13 - Best LSTM model hyperparameter configuration.

Hyperparameter	Tested settings
Epoch	400
# hidden layers	3
# neurons per hidden layers	Hidden layer 1: 18; Hidden layer 2: 165; Hidden layer 3: 179
Loss function	kl_divergence
Learning rate	0.001078
Batch size	32
Dropout	Hidden layer 1: 0.1; Hidden layer 2: 0.3; Hidden layer 3: 0.2
Optimizer function	Nadam
Kernel initializer	Hidden layer 1: gloriot_uniform; Hidden layer 2: he_uniform; Hidden layer 3: lecun_uniform
Time steps	10

#### ○ SVM

A higher gamma value is more suited to detailed modelling. A gamma value that is too high, on the other hand, decreases the sample's range of influence,

resulting in a more irregular decision boundary. A higher  $C$  is also preferable to a more complex decision boundary. A  $C$  value that is too high, in opposition, may lead the model to lose its capability to generalise and may even overfit the data. Therefore, the best models have a  $C$  value between 70 and 90 and a gamma value about 6 (Figure 43).

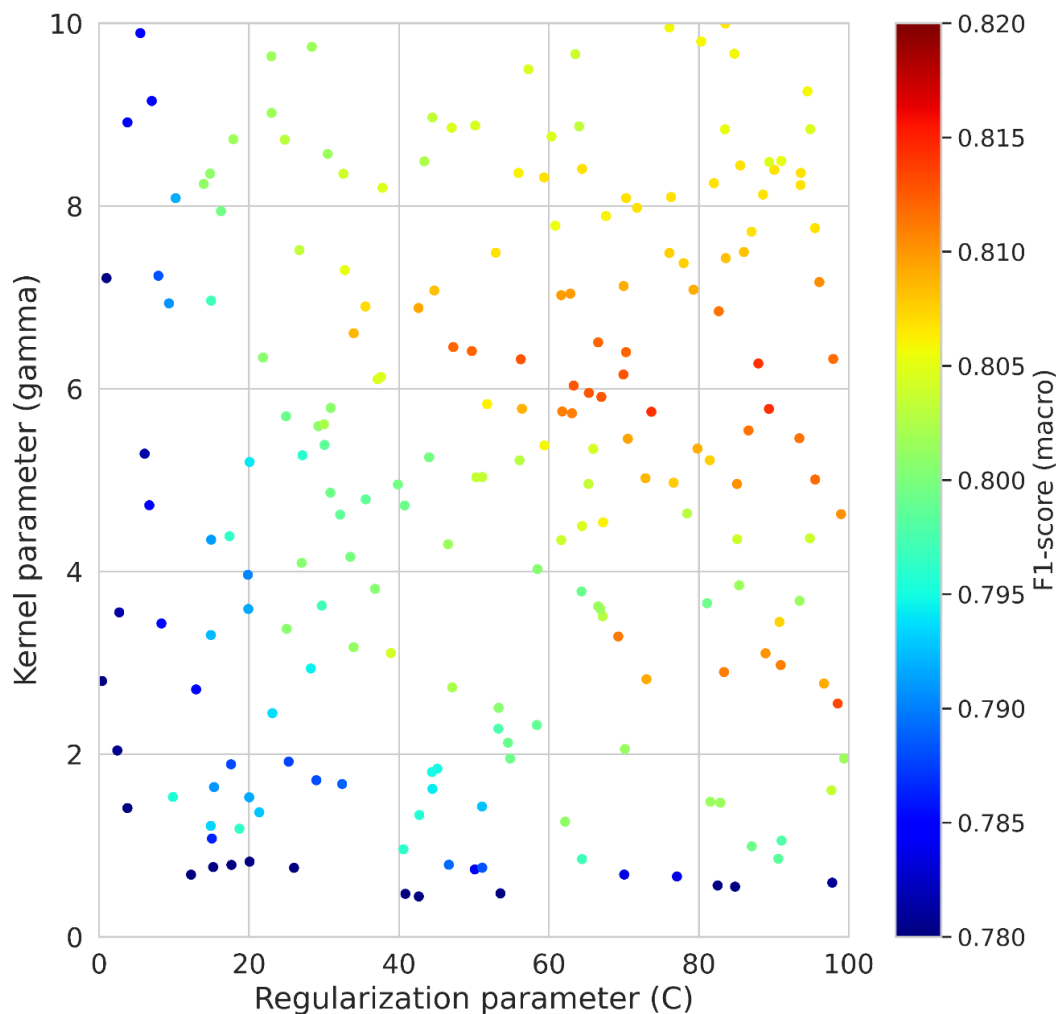


Figure 43 - Hyperparameters search results for the SVM algorithm.

Therefore, the best model has the following hyperparameter configuration (Table 14):

Table 14 - Best SVM model hyperparameter configuration.

Hyperparameter	Tested settings
$C$	73.6749
$\gamma$ (gamma)	5.7478
Decision function shape	one vs one
Kernel function	radial basis function

○ XGBoost

Two key hyperparameters for the XGBoost algorithm are the learning rate and the number of boosting rounds. The learning rate parameter influences the step weight, indicating how quickly the model learns. In XGBoost, the number of boosting rounds equals the number of decision trees. If the learning rate is set to a low value rather than a larger value, the number of boosting rounds required to achieve model capacity is proportionally increased. The best models have 300 to 1000 boosting rounds and a learning rate of up to 0.3 (Figure 44).

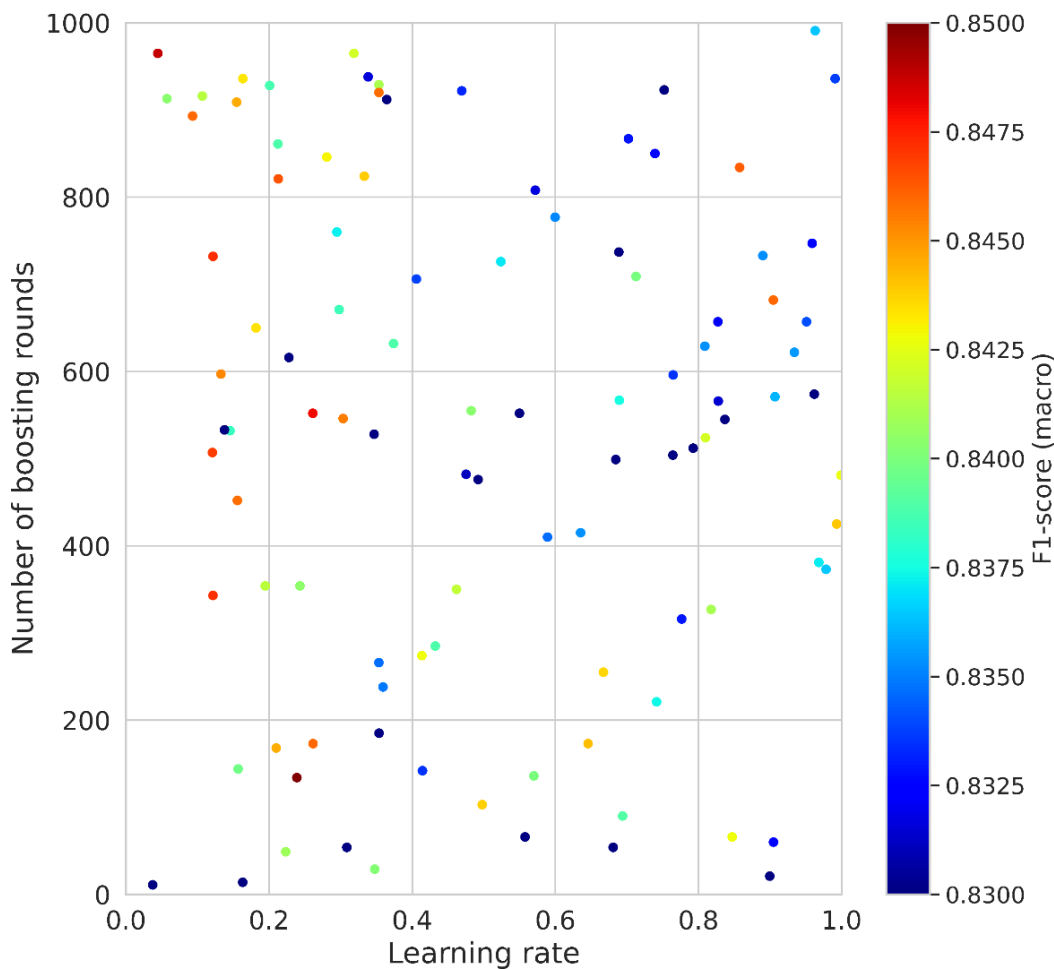


Figure 44 - Hyperparameters search results for the XGBoost algorithm.

Therefore, the best model has the following hyperparameter configuration (Table 15):

Table 15 - Best XGBoost model hyperparameter configuration.

Hyperparameter	Tested settings
Booster	gbtree
Maximum depth of a tree	11
Boosting learning rate	0.238985
Subsample ratio of the training instances	0.6
Number of boosting iterations	134
Loss function	softmax

#### 5.1.4.3

##### Best performing model evaluation

For each machine learning algorithm, the best performing model, according to F1-score with a macro average, was selected. For each best model, the results involving the 30% testing dataset were compared using confusion matrices (Figure 45).

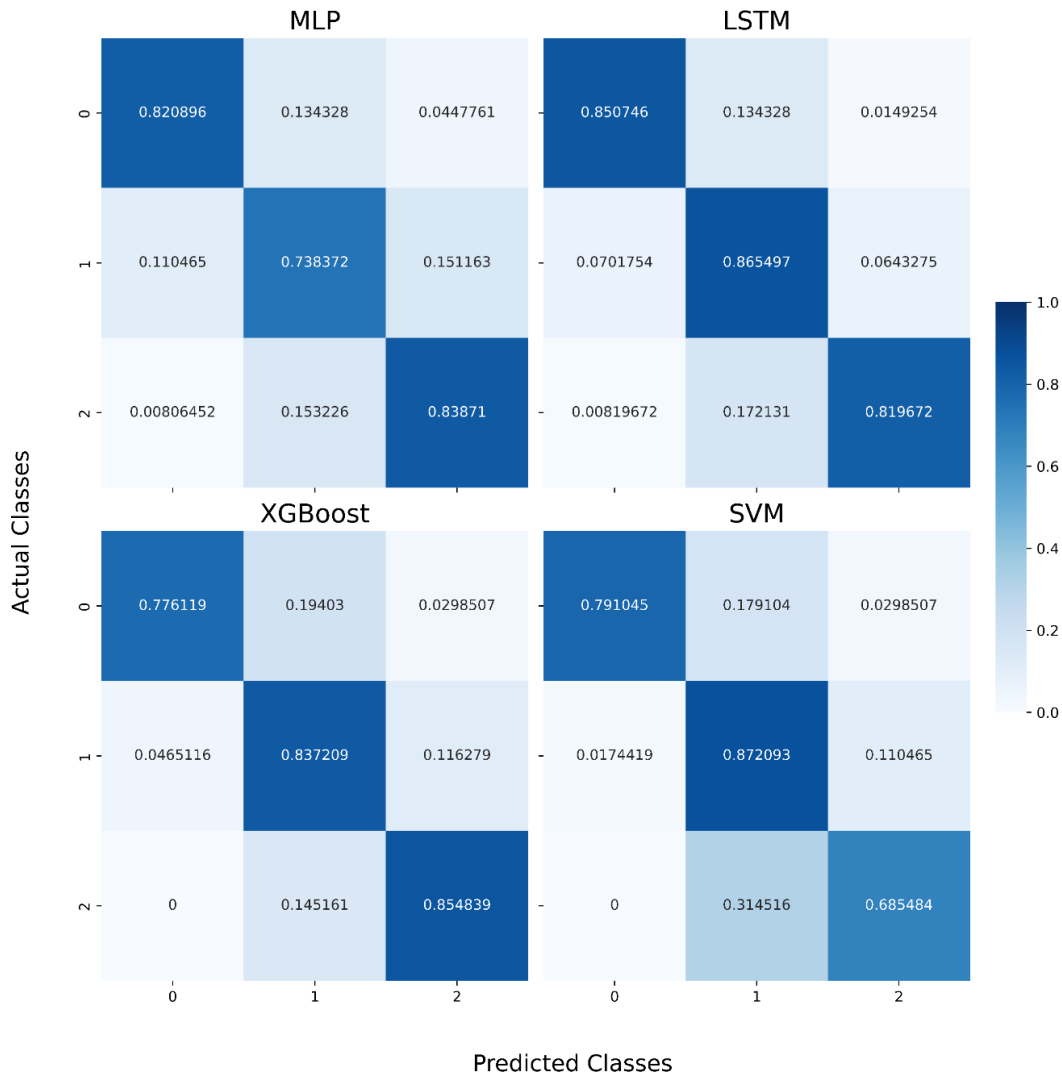


Figure 45 - Confusion matrices comparing the best performing model for each algorithm for the testing dataset.

It's noteworthy to mention that, in terms of overall statistics, referring to the testing dataset, two of the compared algorithms stand out: LSTM and XGBoost. The F1-score (macro) obtained was 0.84499 for the LSTM model and 0.83005 for the XGBoost model, respectively. While SVM and MLP models obtained 0.79662 and 0.78691, respectively.

As previously discussed in this paper, precision with respect to class 1 is a critical metric for the current application of this research. In terms of this metric, the models that used the LSTM, XGBoost, and MLP methods obtained 0.83146, 0.82286, and 0.81935, respectively. The model that used SVM produced a poor result, 0.73892. In fact, the SVM-based model incorrectly classified a significant number of events as belonging to class 1 when the true classification would be class 2.



The recall referring to classes 0 and 2 has a major impact as well. While the model with the best recall in class 0 was the one that used LSTM (0.85075), the best recall in class 2 was obtained by the XGBoost method (0.85484).

This is the first time, to the author's best knowledge, that this approach has been addressed in EC wastewater treatment. Nonetheless, a number of authors are developing machine learning-based approaches for the colorimetric water quality monitoring. These researchers' conclusions demonstrate that the findings of this study are promising and lead to future technological advancements.

Sajed et al. (2020) offered a novel technique for detecting  $\text{Pb}^{2+}$  ions in aqueous solutions utilizing processed RGB images and a machine learning algorithm on a mobile device. As a consequence of the agglomeration of gold nanoparticles caught by a smartphone camera, the color shifts from red wine to violet. A regression model using machine learning techniques was developed and trained to explain the concentration of  $\text{Pb}^{2+}$  as a function of RGB values. The use of nonlinear regression resulted in an accurate estimation of the  $\text{Pb}^{2+}$  concentration. Root mean square error and average absolute error had statistical values of 0.1244 and 0.0943, respectively, between machine learning predictions and experimental outcomes.

Mutlu et al. (2017) presented a smartphone-based machine learning method for identifying pH values automatically. The suggested support vector machine classifier is given the mean R, G, and B values retrieved from images of pH strips collected in three distinct sets of experiments: "with apparatus", "without apparatus", and dual-illumination tests. The classification accuracy of the support vector machine classifier is 100%, with perfect sensitivity and specificity ( $\text{AUC}=1$ ) for both the "with equipment" and "without apparatus" studies. The suggested technique was able to identify non-integer pH values to the closest integer. Additional testing on dual-illuminated pH strips and non-integer pH levels demonstrate that colorimetric detection using machine learning can adapt to increasingly diverse illumination circumstances and is an excellent candidate for completely automating the detection of pH values without human involvement.

Silva et al. (2022) built a smartphone-based portable apparatus based on the colorimetric method for determining the copper and iron concentrations in water samples. In addition to generating calibration curves for copper and iron, the studies

were able to identify the quantities of each metal in samples with unknown values. The findings were comparable to those of commercially available spectrophotometry testing equipment, therefore the authors asserted that this device had potential applications for digital water monitoring and the determination of a larger variety of water quality indicators.

Consequently, colorimetric systems with camera modules have proven significant promise as a portable, sensitive, and cost-effective method for analyzing diverse analytes and conditions, according to the aforementioned literature.

### **5.1.5 Conclusions**

A strategy for designing a data-driven model to detect anomalous process behaviour in a decentralised EC WWTP was proposed in this paper. MLP, LSTM, SVM, and XGBoost algorithms were used to find a suitable model for classifying images into distinct operational conditions. The LSTM model outperforms the others in terms of macro average Precision (84.620%), Recall (84.531%), and F1-score (84.499%), but the XGBoost model comes closely in second with Precision (83.922%), Recall (82.272%), and F1-score (83.005%). It is important to emphasize that this was a pioneering study, combining images from full-scale WWTP with machine learning models. As a result, this research has demonstrated the potential and efficacy of the presented model-based approach in detecting WWTP faults, allowing integration with decision support systems to maintain high performance. For future work, it is conceivable to incorporate the model into production in order to evaluate real-time monitoring and a more diversified training set that includes a larger variety of industrial effluents.

## 6 Conclusions

The present research contributed to the progress of data-driven fault detection methods in EC WWTP. The approaches given here offer WWTP operators with a decision support mechanism that enables for the discovery of anomalies.

Chapter 4 detailed a strategy for selecting features by examining seven distinct feature selection approaches. An ANN classifier was trained and validated using the best features. The F1-score of the best model that used feature selection was 0.92, which was higher than the F1-score of the model that did not utilise feature selection, which was 0.87. It is also worth mentioning that the feature selection model was highly explainable since it encapsulated the fundamental elements of the EC process. This study discovered that the dimensionality of the data may be lowered while preserving or even improving the classifier's prediction performance.

Chapter 5 describes an approach for developing a data-driven model to identify anomalous process behaviour in EC WWTP using images. To the best of this author's knowledge, this is the first time this approach has been addressed in EC wastewater treatment. Two small-size camera modules, as well as image processing and machine learning algorithms, were employed. This approach is less complex and less costly than sensors such as pH, oxidation-reduction potential, and electric conductivity.

The LSTM model outperforms the others in terms of macro average Precision (0.84620), Recall (0.84531), and F1-score (0.84499), but the XGBoost model comes closely in second with Precision (0.83922), Recall (0.82272), and F1-score (0.83005).

The results of the performance metrics obtained only with the wastewater surface images are strongly encouraging. When compared with the metrics obtained using physical sensors, there is no significant drop in performance. This finding is

definitely one of the most significant contributions of this thesis, as well as a direction for future research.

Some suggestions are presented for future research:

- Conduct tests across multiple WWTPs: Employ data from different industrial effluents to evaluate the trained model's performance. Evaluate a potential decline in performance by comparing the model's efficacy after being retrained with the new effluent data to its performance before the retraining;
- Design a convolutional neural network: Collect additional wastewater surface images and develop a convolutional neural network to expand the research reported in this thesis. Additionally, investigate the performance of knowledge transfer in the dataset utilizing state-of-the-art models;
- Develop a general framework for operational control system fault diagnosis: After a fault has been identified, a fault diagnosis can be performed in order to obtain direct estimation for operational optimal control.

## Bibliography

ABIODUN, O. I. et al. State-of-the-art in artificial neural network applications: A survey. **Heliyon**, v. 4, n. 11, p. e00938, Nov. 2018.

AKANSHA, J. et al. Treatment of dairy industry wastewater by combined aerated electrocoagulation and phytoremediation process. **Chemosphere**, v. 253, p. 126652, 2020.

ALABI, M. O.; TELUKDARIE, A.; VAN RENSBURG, N. J. Water 4.0: An Integrated Business Model from an Industry 4.0 Approach. In: IEEE INTERNATIONAL CONFERENCE ON INDUSTRIAL ENGINEERING AND ENGINEERING MANAGEMENT (IEEM). **Anais...** Macao, Macao: IEEE, Dec. 2019. Available: <<https://ieeexplore.ieee.org/document/8978859/>>. Access at: 24 jul. 2022.

ALOM, M. Z. et al. A State-of-the-Art Survey on Deep Learning Theory and Architectures. **Electronics**, v. 8, n. 3, p. 292, 5 Mar. 2019.

ALONSO, V. et al. Industry 4.0 implications in machine vision metrology: an overview. **Procedia Manufacturing**, v. 41, p. 359-366, 2019.

AL-RAAD, A. A.; HANAFIAH, M. M. Removal of inorganic pollutants using electrocoagulation technology: A review of emerging applications and mechanisms. **Journal of Environmental Management**, v. 300, p. 113696, 15 Dec. 2021.

ANTER, A. M.; GUPTA, D.; CASTILLO, O. A novel parameter estimation in dynamic model via fuzzy swarm intelligence and chaos theory for faults in wastewater treatment plant. **Soft Computing**, v. 24, n. 1, p. 111-129, Jan. 2020.

ARISMENDY, L. et al. Intelligent System for the Predictive Analysis of an Industrial Wastewater Treatment Process. **Sustainability**, v. 12, n. 16, p. 6348, 7 ago. 2020.

ASGHARNEJAD, H. et al. Comprehensive review of water management and wastewater treatment in food processing industries in the framework of water-food-environment nexus. **Comprehensive Reviews in Food Science and Food Safety**, v. 20, n. 5, p. 4779-4815, 2021.

ASIR, D.; APPAVU, S.; JEBAMALAR, E. Literature Review on Feature Selection Methods for High-Dimensional Data. **International Journal of Computer Applications**, v. 136, n. 1, p. 9-17, 17 Feb. 2016.

ATKINS, P.; DE PAULA, J. **Elements of physical chemistry**. Oxford: Oxford University Press, USA, 2013.

BAGHERI, M. et al. Modeling of a sequencing batch reactor treating municipal wastewater using multi-layer perceptron and radial basis function artificial neural networks. **Process Safety and Environmental Protection**, v. 93, p. 111-123, Jan. 2015.

BAGHERZADEH, F. et al. Comparative study on total nitrogen prediction in wastewater treatment plant and effect of various feature selection methods on machine learning algorithms performance. **Journal of Water Process Engineering**, v. 41, p. 102033, Jun. 2021.

BAZRAFSHAN, E. et al. Heavy metals removal from aqueous environments by electrocoagulation process—a systematic review. **Journal of environmental health science and engineering**, v. 13, n. 1, p. 1-16, 2015.

BEKKARI, N.; ZEDDOURI, A. Using artificial neural network for predicting and controlling the effluent chemical oxygen demand in wastewater treatment plant. **Management of Environmental Quality: An International Journal**, v. 30, n. 3, p. 593-608, 8 Apr. 2019.

BENER, S. et al. Electrocoagulation process for the treatment of real textile wastewater: Effect of operative conditions on the organic carbon removal and kinetic study. **Process Safety and Environmental Protection**, v. 129, p. 47-54, Sep. 2019.

BOCKRIS, J. O.; REDDY, A. K. Environmentally Oriented Electrochemistry. **Modern Electrochemistry 2B: Electroics in Chemistry, Engineering, Biology, and Environmental Science**, p. 1989-2053, 2000.

BOUSDEKIS, A. et al. A Review of Data-Driven Decision-Making Methods for Industry 4.0 Maintenance Applications. **Electronics**, v. 10, n. 7, p. 828, 31 Mar. 2021.

BRATBY, J. **Coagulation and Flocculation in Water and Wastewater Treatment**. London: IWA Publishing, 2016.

BRILLAS, E.; CABOT, P.-L.; CASADO, J. Electrochemical methods for degradation of organic pollutants in aqueous media. In: **CHEMICAL Degradation Methods for Wastes and Pollutants**. Boca Raton, Florida: CRC Press, 2003. p. 221-284.

BUFLER, R. et al. **Water 4.0—An Important Element for the Germany Water Industry**. Germany Water Partnership. [s.l.: s.n.], [200?].

BURAS, M. P.; SOLANO DONADO, F. Identifying and Estimating the Location of Sources of Industrial Pollution in the Sewage Network. **Sensors**, v. 21, n. 10, p. 3426, 14 May 2021.

CAPITÁN-VALLVEY, L. F. et al. Recent developments in computer vision-based analytical chemistry: A tutorial review. **Analytica Chimica Acta**, v. 899, p. 23-56, Oct. 2015.

CATAÑEDA, L. F. et al. Mathematical modeling and simulation of the reaction environment in electrochemical reactors. **Current Opinion in Electrochemistry**, v. 16, p. 75-82, Aug. 2019.

CHAUHAN, V. K.; DAHIYA, K.; SHARMA, A. Problem formulations and solvers in linear SVM: a review. **Artificial Intelligence Review**, v. 52, n. 2, p. 803-855, Aug. 2019.

CHEN, G. Electrochemical technologies in wastewater treatment. **Separation and purification Technology**, v. 38, n. 1, p. 11-41, 2004.

CHEN, J. P. et al. Bakery waste treatment. In: **HANDBOOK of Industrial and Hazardous Wastes Treatment**. Boca Raton, Florida: CRC Press, 2004. p. 1201-1220.

CHEN, T.; GUESTRIN, C. XGBoost: A Scalable Tree Boosting System. In: KDD '16: THE 22ND ACM SIGKDD INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING. **Anais...** San Francisco California USA: ACM, 13 ago. 2016. Available: <<https://dl.acm.org/doi/10.1145/2939672.2939785>>. Access at: 24 July 2022

CHEN, X.; CHEN, G.; YUE, P. L. Investigation on the electrolysis voltage of electrocoagulation. **Chemical Engineering Science**, v. 57, n. 13, p. 2449-2455, 2002.

CHEN, X.; ISHWARAN, H. Random forests for genomic data analysis. **Genomics**, v. 99, n. 6, p. 323-329, Jun. 2012.

CHENG, T. et al. Forecasting of Wastewater Treatment Plant Key Features Using Deep Learning-Based Models: A Case Study. **IEEE Access**, v. 8, p. 184475-184485, 2020.

CHERNOV, V.; ALANDER, J.; BOCHKO, V. Integer-based accurate conversion between RGB and HSV color spaces. **Computers & Electrical Engineering**, v. 46, p. 328-337, Aug. 2015.

CHING, P. M. L.; SO, R. H. Y.; MORCK, T. Advances in soft sensors for wastewater treatment plants: A systematic review. **Journal of Water Process Engineering**, v. 44, p. 102367, Dec. 2021.

CICCERI, G. et al. A Novel Architecture for the Smart Management of Wastewater Treatment Plants. In: IEEE INTERNATIONAL CONFERENCE ON SMART COMPUTING (SMARTCOMP). **Anais...** Irvine, CA, USA: IEEE, Aug. 2021. Available: <<https://ieeexplore.ieee.org/document/9556228/>>. Access at: 24 jul. 2022

COMPTON, M. et al. Food processing industry energy and water consumption in the Pacific northwest. **Innovative food science & emerging technologies**, v. 47, p. 371-383, 2018.

CRINI, G.; LICHTFOUSE, E. Advantages and disadvantages of techniques used for wastewater treatment. **Environmental Chemistry Letters**, v. 17, n. 1, p. 145-155, March 2019.

CRITTENDEN, J. C. et al. **MWH's Water Treatment: Principles and Design**. New Jersey: Wiley, 2012.

CUI, Q.; EL-ARROUDI, K.; WENG, Y. A Feature Selection Method for High Impedance Fault Detection. **IEEE Transactions on Power Delivery**, v. 34, n. 3, p. 1203-1215, June 2019.

DA SILVA RIBEIRO, T. et al. Removal of boron from mining wastewaters by electrocoagulation method: Modelling experimental data using artificial neural networks. **Minerals Engineering**, v. 131, p. 8-13, 2019.

DAIRI, A. et al. Deep learning approach for sustainable WWTP operation: A case study on data-driven influent conditions monitoring. **Sustainable Cities and Society**, v. 50, p. 101670, Oct. 2019.

DAMIRCHI, S. et al. A comparison between digital camera and spectrophotometer for sensitive and selective kinetic determination of brilliant green in wastewaters. **Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy**, v. 206, p. 232-239, Jan. 2019.

DAMIRCHI, S.; HEIDARI, T. Evaluation of digital camera as a portable colorimetric sensor for low-cost determination of inorganic arsenic (III) in industrial wastewaters by chemical hydride generation assisted-Fe(III) – 1, 10-phenanthroline as a green color agent. **Journal of the Iranian Chemical Society**, v. 15, n. 11, p. 2549-2557, Nov. 2018.

DAS, D.; NANDI, B. K. Removal of Fe (II) ions from drinking water using Electrocoagulation (EC) process: Parametric optimization and kinetic study. **Journal of Environmental Chemical Engineering**, v. 7, n. 3, p. 103116, June 2019.

DAS, P. P.; SHARMA, M.; PURKAIT, M. K. Recent progress on electrocoagulation process for wastewater treatment: A review. **Separation and Purification Technology**, v. 292, p. 121058, July 2022.

DE SANTANA, M. M. et al. Electrochemical treatment of wastewater from a bakery industry: Experimental and modeling study. **Process Safety and Environmental Protection**, v. 116, p. 685-692, 2018.

DENG, X. et al. Feature selection for text classification: A review. **Multimedia Tools and Applications**, v. 78, n. 3, p. 3797-3816, Feb. 2019.



DHAL, P.; AZAD, C. A comprehensive survey on feature selection in the various fields of machine learning. **Applied Intelligence**, 23 July 2021.

DIAZ-ELSAIED, N. et al. Wastewater-based resource recovery technologies across scale: A review. **Resources, Conservation and Recycling**, v. 145, p. 94-112, June 2019.

DUAN, J.; GREGORY, J. Coagulation by hydrolysing metal salts. **Advances in colloid and interface science**, v. 100, p. 475-502, 2003.

DUCHESNE, C.; LIU, J. J.; MACGREGOR, J. F. Multivariate image analysis in the process industries: A review. **Chemometrics and Intelligent Laboratory Systems**, v. 117, p. 116-128, Aug. 2012.

EMAMJOMEH, M. M.; SIVAKUMAR, M. Review of pollutants removed by electrocoagulation and electrocoagulation/flotation processes. **Journal of Environmental Management**, v. 90, n. 5, p. 1663-1679, 2009.

ENSANO, B. M. B. et al. Applicability of the electrocoagulation process in treating real municipal wastewater containing pharmaceutical active compounds. **Journal of Hazardous Materials**, v. 361, p. 367-373, 2019.

EVERETT, D. H. **Basic principles of colloid science**. London: Royal Society of Chemistry, 2007.

FAGARASAN, I.; ILIESCU, S. ST. Parity equations for fault detection and isolation. In: IEEE INTERNATIONAL CONFERENCE ON AUTOMATION, QUALITY AND TESTING, ROBOTICS. **Anais...** 2008. p.

FAN, Y. et al. Digital image colorimetry on smartphone for chemical analysis: A review. **Measurement**, v. 171, p. 108829, 2021.

FEITSHANS, T. A. Environmental Regulations and the Food Industry. **Guide to US Food Laws and Regulations**, p. 121-141, 2013.

FERNANDES, G. M. et al. Novel approaches for colorimetric measurements in analytical chemistry – A review. **Analytica Chimica Acta**, v. 1135, p. 187-203, Oct. 2020.

FERREIRA, A. J.; FIGUEIREDO, M. A. T. Boosting Algorithms: A Review of Methods, Theory, and Applications. In: ZHANG, C.; MA, Y. (Eds.). **Ensemble Machine Learning**. Boston, MA: Springer US, 2012. p. 35-85.

FUSHIKI, T. Estimation of prediction error by using K-fold cross-validation. **Statistics and Computing**, v. 21, n. 2, p. 137-146, April 2011.

GARCIA-SEGURA, S. et al. Electrocoagulation and advanced electrocoagulation processes: A general review about the fundamentals, emerging applications and its association with other technologies. **Journal of Electroanalytical Chemistry**, v. 801, p. 267-299, 2017.

GAUTAM, P. et al. Synergistic optimization of electrocoagulation process parameters using response surface methodology for treatment of hazardous waste landfill leachate. **Chemosphere**, v. 290, p. 133255, Mar. 2022.

GHERNAOUT, D. Electrocoagulation Process: A Mechanistic Review at the Dawn of its Modeling. **Journal of Environmental Science and Allied Research**, 2019.

GHERNAOUT, D.; ALGHAMDI, A.; GHERNAOUT, B. Electrocoagulation Process: A Mechanistic Review at the Dawn of its Modeling. **Journal of Environmental Science and Allied Research**, v. 2, n. 1, p. 22-38, 15 Feb. 2019.

GHOBAKHLOO, M. Industry 4.0, digitization, and opportunities for sustainability. **Journal of Cleaner Production**, v. 252, p. 119869, April 2020.

GHOLAMI SHIRKOOHI, M. et al. A comparison of artificial intelligence models for predicting phosphate removal efficiency from wastewater using the electrocoagulation process. **Digital Chemical Engineering**, v. 4, p. 100043, Sept. 2022.

GOMES, A. J. et al. Electrochemical remediation of chicken processing plant wastewater. **Journal of Environmental Chemical Engineering**, v. 6, n. 5, p. 6028-6036, 2018.

GRANDINI, M.; BAGLI, E.; VISANI, G. Metrics for Multi-Class Classification: an Overview. **arXiv:2008.05756 [cs, stat]**, 13 Aug. 2020.

GUO, H. et al. Prediction of effluent concentration in a wastewater treatment plant using machine learning models. **Journal of Environmental Sciences**, v. 32, p. 90-101, June 2015.

HAKIZIMANA, J. N. et al. Electrocoagulation process in water treatment: A review of electrocoagulation modeling approaches. **Desalination**, v. 404, p. 1-21, Feb. 2017.

HAN, H. et al. Data-driven intelligent monitoring system for key variables in wastewater treatment process. **Chinese Journal of Chemical Engineering**, v. 26, n. 10, p. 2093-2101, Oct. 2018.

HANSEN, H. K. et al. Selenium removal from petroleum refinery wastewater using an electrocoagulation technique. **Journal of Hazardous Materials**, v. 364, p. 78-81, 2019.

HAQUE, M. A. et al. Valorization of bakery waste for biocolorant and enzyme production by *Monascus purpureus*. **Journal of Biotechnology**, v. 231, p. 55-64, 2016.

HASANI, G. et al. A novel ANN approach for modeling of alternating pulse current electrocoagulation-flotation (APC-ECF) process: Humic acid removal from aqueous media. **Process Safety and Environmental Protection**, v. 117, p. 111-124, July 2018.

HASTINGS, G. D.; RUBIN, A. Colour spaces-a review of historic and modern colour models. **African Vision and Eye Health**, v. 71, n. 3, p. 133-143, 2012.

HEJABI, N. et al. Evaluation of the effluent quality parameters of wastewater treatment plant based on uncertainty analysis and post-processing approaches (case study). **Water Science and Technology**, v. 83, n. 7, p. 1633-1648, 1 April 2021.

HOLT, P. K. et al. A quantitative comparison between chemical dosing and electrocoagulation. **Colloids and Surfaces A: Physicochemical and Engineering Aspects**, v. 211, n. 2-3, p. 233-248, 2002.

HOPF, K.; REIFENRATH, S. Filter Methods for Feature Selection in Supervised Machine Learning Applications. Review and Benchmark. **arXiv:2111.12140 [cs, stat]**, 23 nov. 2021.

HOU, L.; JI, D.; ZANG, L. Inhibition of anaerobic biological treatment: A review. In: IOP CONFERENCE SERIES: EARTH AND ENVIRONMENTAL SCIENCE. **Anais...** [s.l.]: IOP Publishing, 2018. p.

HUNTER, R. J. **Introduction to modern colloid science**. Oxford: Oxford University Press, 1993.

HWANGBO, S. et al. Integrated Model for Understanding N<sub>2</sub>O Emissions from Wastewater Treatment Plants: A Deep Learning Approach. **Environmental Science & Technology**, v. 55, n. 3, p. 2143-2151, 2 fev. 2021.

ILHAN, F. et al. Electrocoagulation process for the treatment of metal-plating wastewater: Kinetic modeling and energy consumption. **Frontiers of Environmental Science & Engineering**, v. 13, n. 5, p. 73, out. 2019.

JAVAID, M. et al. Exploring impact and features of machine vision for progressive industry 4.0 culture. **Sensors International**, v. 3, p. 100132, 2022.

JEON, H.; OH, S. Hybrid-Recursive Feature Elimination for Efficient Feature Selection. **Applied Sciences**, v. 10, n. 9, p. 3211, 4 May 2020.

JEONG, N.; CHUNG, T.; TONG, T. Predicting Micropollutant Removal by Reverse Osmosis and Nanofiltration Membranes: Is Machine Learning Viable? **Environmental Science & Technology**, v. 55, n. 16, p. 11348-11359, 17 ago. 2021.

JEROME, R. E.; SINGH, S. K.; DWIVEDI, M. Process analytical technology for bakery industry: A review. **Journal of Food Process Engineering**, v. 42, n. 5, p. e13143, 2019.

JIANG, Y. et al. A Review on Soft Sensors for Monitoring, Control, and Optimization of Industrial Processes. **IEEE Sensors Journal**, v. 21, n. 11, p. 12868-12881, 1 jun. 2021.

JOVIC, A.; BRKIC, K.; BOGUNOVIC, N. A review of feature selection methods with applications. In: 38TH INTERNATIONAL CONVENTION ON INFORMATION AND COMMUNICATION TECHNOLOGY, ELECTRONICS AND MICROELECTRONICS (MIPRO). **Anais...** Opatija, Croatia: IEEE, May 2015. Available: <<http://ieeexplore.ieee.org/document/7160458/>>. Access at: 24 jul. 2022

KABDAŞLI, I. et al. Electrocoagulation applications for industrial wastewaters: a critical review. **Environmental Technology Reviews**, v. 1, n. 1, p. 2-45, 2012.

KALSOOM, T. et al. Advances in Sensor Technologies in the Era of Smart Factory and Industry 4.0. **Sensors**, v. 20, n. 23, p. 6783, 27 nov. 2020.

KAZEMI, P. et al. Data-driven techniques for fault detection in anaerobic digestion process. **Process Safety and Environmental Protection**, v. 146, p. 905-915, fev. 2021.

KHAIRE, U. M.; DHANALAKSHMI, R. Stability of feature selection algorithm: A review. **Journal of King Saud University - Computer and Information Sciences**, p. S1319157819304379, jun. 2019.

KHAN, M. B. et al. Generalized classification modeling of activated sludge process based on microscopic image analysis. **Environmental Technology**, v. 39, n. 1, p. 24-34, 2 jan. 2018.

KHAN, S. U. et al. Energy Efficient Rapid Removal of Arsenic in an Electrocoagulation Reactor with Hybrid Fe/Al Electrodes: Process Optimization Using CCD and Kinetic Modeling. **Water**, v. 12, n. 10, p. 2876, 16 out. 2020.

KIJAK, R. Defining Water 4.0. In: KIJAK, R. (Ed.). **Water Asset Management in Times of Climate Change and Digital Transformation**. Cham: Springer International Publishing, 2021. p.

KINI, K. R.; MADAKYARU, M. Kantorovich Distance based Fault Detection Scheme: An Application to Wastewater Treatment Plant. **IFAC-PapersOnLine**, v. 55, n. 1, p. 345-350, 2022.

KLEMES, J.; SMITH, R.; KIM, J.-K. **Handbook of water and energy management in food processing**. Amsterdam: Elsevier, 2008.

KOROBOV, M.; LOPUHIN, K. **Eli5**. Available: <<https://eli5.readthedocs.io/en/latest/overview.html>>. Access at: ???

KUDELINA, K. et al. Trends and Challenges in Intelligent Condition Monitoring of Electrical Machines Using Machine Learning. **Applied Sciences**, v. 11, n. 6, p. 2761, 19 mar. 2021.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **Nature**, v. 521, n. 7553, p. 436-444, 28 May 2015.

LEE, C.-Y.; WEN, M.-S. Establish Induction Motor Fault Diagnosis System Based on Feature Selection Approaches with MRA. **Processes**, v. 8, n. 9, p. 1055, 29 ago. 2020.

LEUNG, H.; HAYKIN, S. The complex backpropagation algorithm. **IEEE Transactions on signal processing**, v. 39, n. 9, p. 2101-2104, 1991.

LEVENSPIEL, O. Theories of chemical reaction rates. **Ind. Eng. Chem. Res.**, v. 38, p. 4140-4143, 1999.

LI, J.; LIU, H. Challenges of Feature Selection for Big Data Analytics. **IEEE Intelligent Systems**, v. 32, n. 2, p. 9-15, mar. 2017.

LI, W. et al. Process fault diagnosis with model- and knowledge-based approaches: Advances and opportunities. **Control Engineering Practice**, v. 105, p. 104637, dez. 2020.

LI, Y.; LI, T.; LIU, H. Recent advances in feature selection and its applications. **Knowledge and Information Systems**, v. 53, n. 3, p. 551-577, dez. 2017.

LIN, S. Rank aggregation methods: Rank aggregation methods. **Wiley Interdisciplinary Reviews: Computational Statistics**, v. 2, n. 5, p. 555-570, set. 2010.

LINDSTRÖM, T. Some fundamental chemical aspects on paper forming. **Fundamentals of Papermaking**, v. 1, p. 309-412, 1989.

LIU, Y.; CHEN, Y.; FANG, X. A Review of Turbidity Detection Based on Computer Vision. **IEEE Access**, v. 6, p. 60586-60604, 2018.

LÓPEZ-GUZMÁN, M.; FLORES-HIDALGO, M. A.; REYNOSO-CUEVAS, L. Electrocoagulation Process: An Approach to Continuous Processes, Reactors Design, Pharmaceuticals Removal, and Hybrid Systems-A Review. **Processes**, v. 9, n. 10, 2021.

LOWE, M.; QIN, R.; MAO, X. A Review on Machine Learning, Artificial Intelligence, and Smart Technology in Water Treatment and Monitoring. **Water**, v. 14, n. 9, p. 1384, 24 abr. 2022.

LU, H.; MA, X. Hybrid decision tree-based machine learning models for short-term water quality prediction. **Chemosphere**, v. 249, p. 126169, jun. 2020.

LU, J.; ZHANG, P.; LI, J. Electrocoagulation technology for water purification: An update review on reactor design and some newly concerned pollutants removal. **Journal of Environmental Management**, v. 296, p. 113259, out. 2021.

LUCA, A.-V. et al. Data Driven Detection of Different Dissolved Oxygen Sensor Faults for Improving Operation of the WWTP Control System. **Processes**, v. 9, n. 9, p. 1633, 10 set. 2021.

LUNDBERG, S. M. et al. From local explanations to global understanding with explainable AI for trees. **Nature Machine Intelligence**, v. 2, n. 1, p. 56-67, jan. 2020.

LUNDBERG, S. M.; LEE, S.-I. **A Unified Approach to Interpreting Model Predictions**, v. 30, 2017.

LV, C. et al. Stereo Matching Algorithm Based on HSV Color Space and Improved Census Transform. **Mathematical Problems in Engineering**, v. 2021, p. 1-17, 20 jul. 2021.

MAGNISALI, E.; YAN, Q.; VAYENAS, D. V. Electrocoagulation as a revived wastewater treatment method-practical approaches: a review. **Journal of Chemical Technology & Biotechnology**, v. 97, n. 1, p. 9-25, jan. 2022.

MAMANDIPOOR, B. et al. Monitoring and detecting faults in wastewater treatment plants using deep learning. **Environmental Monitoring and Assessment**, v. 192, n. 2, p. 148, fev. 2020.

MARIAH, G. K.; PAK, K. S. Removal of brilliant green dye from aqueous solution by electrocoagulation using response surface methodology. **Materials Today: Proceedings**, v. 20, p. 488-492, 2020.

MD NOR, N.; CHE HASSAN, C. R.; HUSSAIN, M. A. A review of data-driven fault detection and diagnosis methods: applications in chemical process systems. **Reviews in Chemical Engineering**, v. 36, n. 4, p. 513-553, 26 May 2020.

MIHÁLY, N.-B.; SIMON-VÁRHELYI, M.; CRISTEA, V. M. Data-driven modelling based on artificial neural networks for predicting energy and effluent quality indices and wastewater treatment plant optimization. **Optimization and Engineering**, 3 May 2022.

MOHAN, S.; VIVEKANANDHAN, V.; PRIYADHARSHINI, S. Performance evaluation of modified UASB Reactor for treating bakery effluent. **International Journal of Applied Environmental Sciences**, v. 12, n. 11, p. 1883-1894, 2017.

MOLLAH, M. et al. Fundamentals, present and future perspectives of electrocoagulation. **Journal of Hazardous Materials**, v. 114, n. 1-3, p. 199-210, 18 out. 2004.

MOLLAH, M. Y. A. et al. Electrocoagulation (EC - science and applications. **Journal of Hazardous Materials**, v. 84, n. 1, p. 29-41, 2001.

MORALES-RIVERA, J. et al. Modeling and Optimization of COD Removal from Cold Meat Industry Wastewater by Electrocoagulation Using Computational Techniques. **Processes**, v. 8, n. 9, p. 1139, 11 set. 2020.

MORENO C, H. A. et al. Electrochemical reactions for electrocoagulation using iron electrodes. **Industrial & Engineering Chemistry Research**, v. 48, n. 4, p. 2275-2282, 2009.

MOUSAZADEH, M. et al. A systematic diagnosis of state of the art in the use of electrocoagulation as a sustainable technology for pollutant treatment: An updated review. **Sustainable Energy Technologies and Assessments**, v. 47, p. 101353, out. 2021.

MOUSSA, D. T. et al. A comprehensive review of electrocoagulation for water treatment: Potentials and challenges. **Journal of Environmental Management**, v. 186, p. 24-41, 2017.

MUTLU, A. Y. et al. Smartphone-based colorimetric detection via machine learning. **Analyst**, v. 142, n. 13, p. 2434-2441, 2017.

NAIR, A. T.; MAKWANA, A. R.; AHAMMED, M. M. The use of response surface methodology for modelling and analysis of water and wastewater treatment processes: a review. **Water Science and Technology**, v. 69, n. 3, p. 464-478, nov. 2013.

NAJAFZADEH, M.; ZEINOLABEDINI, M. Prognostication of waste water treatment plant performance using efficient soft computing models: An environmental evaluation. **Measurement**, v. 138, p. 690-701, May 2019.

NARIYAN, E.; SILLANPÄÄ, M.; WOLKERSDORFER, C. Uranium removal from Pyhäsalmi/Finland mine water by batch electrocoagulation and optimization with the response surface methodology. **Separation and Purification Technology**, v. 193, p. 386-397, mar. 2018.

NASTESKI, V. An overview of the supervised machine learning methods. **Horizons.B**, v. 4, p. 51-62, 15 dez. 2017.

NEWHART, K. B. et al. Data-driven performance analyses of wastewater treatment plants: A review. **Water Research**, v. 157, p. 498-513, jun. 2019.

NEWHART, K. B. et al. Prediction of Peracetic Acid Disinfection Performance for Secondary Municipal Wastewater Treatment Using Artificial Neural Networks. **ACS ES&T Water**, v. 1, n. 2, p. 328-338, 12 fev. 2021.

NEWMAN, J.; THOMAS-ALYEA, K. E. **Electrochemical systems**. New Jersey: John Wiley & Sons, 2012.

NOR, N. M.; HASSAN, C. R. C.; HUSSAIN, M. A. A review of data-driven fault detection and diagnosis methods: applications in chemical process systems. **Reviews in Chemical Engineering**, v. 36, n. 4, p. 513-553, 2020.

NOURANI, V.; ASGHARI, P.; SHARGHI, E. Artificial intelligence based ensemble modeling of wastewater treatment plant using jittered data. **Journal of Cleaner Production**, v. 291, p. 125772, abr. 2021.

NOURANI, V.; ELKIRAN, G.; ABBA, S. I. Wastewater treatment plant performance analysis using artificial intelligence – an ensemble approach. **Water Science and Technology**, v. 78, n. 10, p. 2064-2076, 21 dez. 2018.

OH, S.; LOGAN, B. E. Hydrogen and electricity production from a food processing wastewater using fermentation and microbial fuel cell technologies. **Water Research**, v. 39, n. 19, p. 4673-4682, 2005.

OLSEN, T. L.; TOMLIN, B. Industry 4.0: Opportunities and Challenges for Operations Management. **Manufacturing & Service Operations Management**, v. 22, n. 1, p. 113-122, jan. 2020.

OZTEMEL, E.; GURSEV, S. Literature review of Industry 4.0 and related technologies. **Journal of Intelligent Manufacturing**, v. 31, n. 1, p. 127-182, jan. 2020.

PARK, Y.-J.; FAN, S.-K. S.; HSU, C.-Y. A Review on Fault Detection and Process Diagnostics in Industrial Processes. **Processes**, v. 8, n. 9, p. 1123, 9 set. 2020.

PEDREGOSA, F. et al. Scikit-learn: Machine Learning in Python. **Machine Learning in Python**, v. 12, p. 2825-2830, 2011.

PEREZ, N. **Electrochemistry and corrosion science**. Oklahoma: Springer, 2004.

PHUANGSAIJAI, N.; JAKMUNEE, J.; KITTIWACHANA, S. Investigation into the predictive performance of colorimetric sensor strips using RGB, CMYK, HSV, and CIELAB coupled with various data preprocessing methods: a case study on an analysis of water quality parameters. **Journal of Analytical Science and Technology**, v. 12, n. 1, p. 19, dez. 2021.

PIRDASHTI, M. et al. Artificial neural networks: applications in chemical engineering. **Reviews in Chemical Engineering**, v. 29, n. 4, p. 205-239, 2013.

ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. **Psychological Review**, v. 65, n. 6, p. 386, 1958.

RUBINSTEIN, I. Fundamentals of physical electrochemistry. Physical electrochemistry: principles, methods, and applications. New York: Marcell Dekker, Inc, 1995.



SAHU, O.; MAZUMDAR, B.; CHAUDHARI, P. K. Treatment of wastewater by electrocoagulation: a review. **Environmental Science and Pollution Research**, v. 21, n. 4, p. 2397-2413, fev. 2014.

SAJED, S. et al. High-Performance Estimation of Lead Ion Concentration Using Smartphone-Based Colorimetric Analysis and a Machine Learning Approach. **ACS Omega**, v. 5, n. 42, p. 27675-27684, 27 out. 2020.

SALESKY, E. et al. Optimizing segmentation granularity for neural machine translation. **Machine Translation**, v. 34, n. 1, p. 41-59, abr. 2020.

SÁNCHEZ-FERNÁNDEZ, A. et al. Fault detection based on time series modeling and multivariate statistical process control. **Chemometrics and Intelligent Laboratory Systems**, v. 182, p. 57-69, nov. 2018.

SANTIAGO, J. B.; SEVILLA, F. B. Smartphone-based digital colorimetric measurement of dimethyl sulfide in wastewater. **Microchemical Journal**, v. 172, p. 106952, jan. 2022.

SARDARI, K. et al. Electrocoagulation followed by ultrafiltration for treating poultry processing wastewater. **Journal of Environmental Chemical Engineering**, v. 6, n. 4, p. 4937-4944, 2018.

SARKER, I. H. Machine Learning: Algorithms, Real-World Applications and Research Directions. **SN Computer Science**, v. 2, n. 3, p. 160, May 2021.

SHAHEDI, A. et al. A review on industrial wastewater treatment via electrocoagulation processes. **Current Opinion in Electrochemistry**, v. 22, p. 154-169, 2020.

SHARMA, S. et al. Organic pollutant removal from edible oil process wastewater using electrocoagulation. In: IOP CONFERENCE SERIES: EARTH AND ENVIRONMENTAL SCIENCE. **Anais...**Bristol: IOP Publishing, 2018. p.

SILVA, G. M. E et al. Smartphone-based spectrometry system as a prescreening assessment of copper and iron for real time control of water pollution. **Journal of Environmental Management**, v. 323, p. 116214, dez. 2022.

SNOEK, J.; LAROCHELLE, H.; ADAMS, R. P. Practical Bayesian Optimization of Machine Learning Algorithms. **Advances in Neural Information Processing Systems**, v. 25, p. 9, 2012.

SOLANO, F.; KRAUSE, S.; WOLLGENS, C. An Internet-of-Things Enabled Smart System for Wastewater Monitoring. **IEEE Access**, v. 10, p. 4666-4685, 2022.

SPEISER, J. L. et al. A comparison of random forest variable selection methods for classification prediction modeling. **Expert Systems with Applications**, v. 134, p. 93-101, nov. 2019.

- SUHAN, M. B. K. et al. Comparative degradation study of remazol black B dye using electro-coagulation and electro-Fenton process: Kinetics and cost analysis. **Environmental Nanotechnology, Monitoring & Management**, v. 14, p. 100335, 2020.
- SUNDUI, B. et al. Applications of machine learning algorithms for biological wastewater treatment: Updates and perspectives. **Clean Technologies and Environmental Policy**, v. 23, n. 1, p. 127-143, jan. 2021.
- TANHA, J. et al. Boosting methods for multi-class imbalanced data classification: an experimental review. **Journal of Big Data**, v. 7, n. 1, p. 70, dez. 2020.
- URBANOWICZ, R. J. et al. Relief-based feature selection: Introduction and review. **Journal of Biomedical Informatics**, v. 85, p. 189-203, set. 2018.
- VAN DEN BROEK, E. L. Human-centered content-based image retrieval. **Neuroscience Research Communications - NEUROSCI RES COMMUN**, 1 jan. 2005.
- VASUDEVAN, S. et al. Removal of NO<sub>3</sub>-from drinking water by electrocoagulation—an alternate approach. **Clean–Soil, Air, Water**, v. 38, n. 3, p. 225-229, 2010.
- VENKATASUBRAMANIAN, V. et al. A review of process fault detection and diagnosis Part I: Quantitative model-based methods. **Computers and Chemical Engineering**, p. 19, 2003.
- VENKATESH, B.; ANURADHA, J. A Review of Feature Selection and Its Methods. **Cybernetics and Information Technologies**, v. 19, n. 1, p. 3-26, 1 mar. 2019.
- VEPSÄLÄINEN, M.; SILLANPÄÄ, M. Electrocoagulation in the treatment of industrial waters and wastewaters. In: **ADVANCED Water Treatment**. Amsterdam: Elsevier, 2020. p. 1-78.
- VERGARA, J. R.; ESTÉVEZ, P. A. A review of feature selection methods based on mutual information. **Neural Computing and Applications**, v. 24, n. 1, p. 175-186, jan. 2014.
- VIK, E. A. et al. Electrocoagulation of potable water. **Water Research**, v. 18, n. 11, p. 1355-1360, 1984.
- WALSH, B. P.; CUSACK, D. O.; O’SULLIVAN, D. An industrial water management value system framework development. **Sustainable Production and Consumption**, v. 5, p. 82-93, 2016.
- WANG, L. **Support vector machines: theory and applications**. New York: Springer Science & Business Media, 2005. v. 177

WU, Z. et al. Continuous flowing electrocoagulation reactor for efficient removal of azo dyes: Kinetic and isotherm studies of adsorption. **Environmental Technology & Innovation**, v. 22, p. 101448, May 2021.

XIAO, H. et al. Studies on the corrosion and the behavior of inert anodes in aluminum electrolysis. **Metallurgical and Materials Transactions B**, v. 27, n. 2, p. 185-193, 1996.

XING, Y. et al. A cellphone-based colorimetric multi-channel sensor for water environmental monitoring. **Frontiers of Environmental Science & Engineering**, v. 16, n. 12, p. 155, dez. 2022.

XU, L. D.; XU, E. L.; LI, L. Industry 4.0: state of the art and future trends. **International Journal of Production Research**, v. 56, n. 8, p. 2941-2962, 18 abr. 2018.

YANG, Y.; LI, J.; YANG, Y. **The research of the fast SVM classifier method.** In: 12TH INTERNATIONAL COMPUTER CONFERENCE ON WAVELET ACTIVE MEDIA TECHNOLOGY AND INFORMATION PROCESSING (ICCWAMTIP). **Anais...** Chengdu, China: IEEE, dez. 2015. Available: <<http://ieeexplore.ieee.org/document/7493959/>>. Access at: 2 jul. 2022.

YAQUB, M. et al. Modeling of a full-scale sewage treatment plant to predict the nutrient removal efficiency using a long short-term memory (LSTM) neural network. **Journal of Water Process Engineering**, v. 37, p. 101388, out. 2020.

YASIN, H. M. et al. IoT and ICT based Smart Water Management, Monitoring and Controlling System: A Review. **Asian Journal of Research in Computer Science**, p. 42-56, 5 May 2021.

YU, Y. et al. A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. **Neural Computation**, v. 31, n. 7, p. 1235-1270, jul. 2019.

ZAIED, B. K. et al. A comprehensive review on contaminants removal from pharmaceutical wastewater by electrocoagulation process. **Science of The Total Environment**, v. 726, p. 138095, jul. 2020.

ZHANG, F. et al. An integrated prediction model of heavy metal ion concentration for iron electrocoagulation process. **Chemical Engineering Journal**, v. 391, p. 123628, jul. 2020.

ZHANG, W.; TOOKER, N. B.; MUELLER, A. V. Enabling wastewater treatment process automation: leveraging innovations in real-time sensing, data analysis, and online controls. **Environmental Science: Water Research & Technology**, v. 6, n. 11, p. 2973-2992, 2020.

ZHAO, L. et al. Application of artificial intelligence to wastewater treatment: A bibliometric analysis and systematic review of technology, economy,

management, and wastewater reuse. **Process Safety and Environmental Protection**, v. 133, p. 169-182, jan. 2020.

ZHU, H. et al. A prediction method of electrocoagulation reactor removal rate based on Long Term and Short Term Memory–Autoregressive Integrated Moving Average Model. **Process Safety and Environmental Protection**, v. 152, p. 462-470, ago. 2021.

ZONTA, T. et al. Predictive maintenance in the Industry 4.0: A systematic literature review. **Computers & Industrial Engineering**, v. 150, p. 106889, dez. 2020.

ZOU, J.; HAN, Y.; SO, S.-S. **Overview of Artificial Neural Networks**, p. 14-22, 2008.

ZOUNEMAT-KERMANI, M. et al. Prediction of effluent arsenic concentration of wastewater treatment plants using machine learning and kriging-based models. **Environmental Science and Pollution Research**, v. 29, n. 14, p. 20556-20570, mar. 2022.

## A Model construction algorithm

The author would like to include here, for future reference, the Python code used to train the models.

```
import numpy as np
import pandas as pd
import keras
from tensorflow.keras.utils import to_categorical
from sklearn.utils import class_weight
from sklearn.model_selection import train_test_split
from sklearn.model_selection import KFold, StratifiedKFold
from sklearn.preprocessing import StandardScaler
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Dropout
from keras.layers import BatchNormalization
from keras.layers import Activation

dataset = pd.read_excel("Datalog_VA+PUC_R3_HSV_Thiago.xlsx", "
DATA_OK_Relevantes", index_col=None, na_values=["NA"], engine=
"openpyxl")
dataset = dataset.iloc[:,0:19]
dataset = dataset.drop(['Energia (Kwh/m3)'], axis=1)
X = dataset.iloc[:,[0,1,2,11,12,13]].values
y = dataset.iloc[:,17].values

y_categorical = to_categorical(y,3)

class_weights = class_weight.compute_class_weight('balanced',n
p.unique(dataset['ESTADO']),dataset['ESTADO'])
class_weights = dict(enumerate(class_weights))

X_train, X_test, y_train, y_test = train_test_split(X, y_categ
orical, test_size=0.3, stratify=y_categorical, random_state=12
3, shuffle=True)
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)

# Configure the sweep - specify the parameters to search throu
gh, the search strategy, the optimization metric et all.
sweep_config = {
```

```

# 'method': 'random',
'method': 'bayes',
'metric': {
    'name': 'cp_total_per_fold_mean',
    'goal': 'maximize'
},
'parameters': {
    'epoch': {
        'distribution': 'int_uniform',
        'max': 500,
        'min': 1
    },
    'activation_1': {
        'distribution': 'categorical',
        'values': ['linear', 'relu', 'sigmoid', 'tanh', 'se
lu', 'elu']
    },
    'activation_2': {
        'distribution': 'categorical',
        'values': ['linear', 'relu', 'sigmoid', 'tanh', 'se
lu', 'elu']
    },
    'activation_3': {
        'distribution': 'categorical',
        'values': ['linear', 'relu', 'sigmoid', 'tanh', 'se
lu', 'elu']
    },
    'activation_4': {
        'distribution': 'categorical',
        'values': ['linear', 'relu', 'sigmoid', 'tanh', 'se
lu', 'elu']
    },
    'activation_5': {
        'distribution': 'categorical',
        'values': ['linear', 'relu', 'sigmoid', 'tanh', 'se
lu', 'elu']
    },
    'n_layers': {
        'distribution': 'int_uniform',
        'max': 5,
        'min': 1
    },
    'layer_1': {
        'distribution': 'int_uniform',
        'max': 250,
        'min': 1
    },
    'layer_2': {

```

```

        'distribution': 'int_uniform',
        'max': 250,
        'min': 1
    },
    'layer_3': {
        'distribution': 'int_uniform',
        'max': 250,
        'min': 1
    },
    'layer_4': {
        'distribution': 'int_uniform',
        'max': 250,
        'min': 1
    },
    'layer_5': {
        'distribution': 'int_uniform',
        'max': 250,
        'min': 1
    },
    'loss': {
        'distribution': 'categorical',
        'values': ['categorical_crossentropy', 'kl_diverge
nce', 'poisson']
    },
    'learning_rate': {
        'distribution': 'uniform',
        'max': 1e-2,
        'min': 1e-5
    },
    'batch_size': {
        'values': [32, 64, 128, 256, 512]
    },
    'Dropout_1': {
        'values': [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7,
0.8, 0.9]
    },
    'Dropout_2': {
        'values': [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7,
0.8, 0.9]
    },
    'Dropout_3': {
        'values': [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7,
0.8, 0.9]
    },
    'Dropout_4': {
        'values': [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7,
0.8, 0.9]
    },

```

```

        'Dropout_5': {
            'values': [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7,
0.8, 0.9]
        },
        'optimizer': {
            'distribution': 'categorical',
            'values': ['adam', 'SGD', 'adamax', 'nadam']
        },
        'kernel_initializer_1': {
            'distribution': 'categorical',
            'values': ['he_uniform', 'glorot_uniform', 'lecun_
uniform']
        },
        'kernel_initializer_2': {
            'distribution': 'categorical',
            'values': ['he_uniform', 'glorot_uniform', 'lecun_
uniform']
        },
        'kernel_initializer_3': {
            'distribution': 'categorical',
            'values': ['he_uniform', 'glorot_uniform', 'lecun_
uniform']
        },
        'kernel_initializer_4': {
            'distribution': 'categorical',
            'values': ['he_uniform', 'glorot_uniform', 'lecun_
uniform']
        },
        'kernel_initializer_5': {
            'distribution': 'categorical',
            'values': ['he_uniform', 'glorot_uniform', 'lecun_
uniform']
        },
        'kernel_initializer_6': {
            'distribution': 'categorical',
            'values': ['he_uniform', 'glorot_uniform', 'lecun_
uniform']
        },
    }
}

# Initialize a new sweep
# Arguments:
#     - sweep_config: the sweep config dictionary defined above
#     - entity: Set the username for the sweep

```



```

# - project: Set the project name for the sweep
sweep_id = wandb.sweep(sweep_config, entity="thiagoribeiro1",
project="V4.0_sweeps-VentilAQUA_Datalog_HSV_MLP_3")

def train():
    config_defaults = {
        "n_layers": 1,
        "layer_1": 1,
        "activation_1": "linear",
        "layer_2": 1,
        "activation_2": "linear",
        "layer_3": 1,
        "activation_3": "linear",
        "layer_4": 1,
        "activation_4": "linear",
        "layer_5": 1,
        "activation_5": "linear",
        "optimizer": "adam",
        "loss": "categorical_crossentropy",
        "epoch": 100,
        "batch_size": 32,
        "learning_rate": 1e-3,
        "Dropout_1":0.3,
        "Dropout_2":0.3,
        "Dropout_3":0.3,
        "Dropout_4":0.3,
        "Dropout_5":0.3,
        "kernel_initializer_1": "he_uniform",
        "kernel_initializer_2": "he_uniform",
        "kernel_initializer_3": "he_uniform",
        "kernel_initializer_4": "he_uniform",
        "kernel_initializer_5": "he_uniform",
        "kernel_initializer_6": "he_uniform",

    }

    # Initialize a new wandb run
    wandb.init(config=config_defaults)

    # Config is a variable that holds and saves hyperparameters
    and inputs
    config = wandb.config

    import tensorflow.compat.v1.keras.backend as K
    import tensorflow as tf
    tf.compat.v1.enable_eager_execution()

    model = Sequential()

```

```

if config.n_layers==1:
    model.add(Dense(units=config.layer_1, kernel_initializer=c
onfig.kernel_initializer_1))
    model.add(BatchNormalization())
    model.add(Activation(config.activation_1))
    model.add(Dropout(config.Dropout_1))
    wandb.log({"n_neurons_hidden": np.sum(config.layer_1)})
if config.n_layers==2:
    model.add(Dense(units=config.layer_1, kernel_initializer=c
onfig.kernel_initializer_1))
    model.add(BatchNormalization())
    model.add(Activation(config.activation_1))
    model.add(Dropout(config.Dropout_1))
    model.add(Dense(units=config.layer_2, kernel_initializer=c
onfig.kernel_initializer_2))
    model.add(BatchNormalization())
    model.add(Activation(config.activation_2))
    model.add(Dropout(config.Dropout_2))
    wandb.log({"n_neurons_hidden": np.sum(config.layer_1+confi
g.layer_2)})
if config.n_layers==3:
    model.add(Dense(units=config.layer_1, kernel_initializer=c
onfig.kernel_initializer_1))
    model.add(BatchNormalization())
    model.add(Activation(config.activation_1))
    model.add(Dropout(config.Dropout_1))
    model.add(Dense(units=config.layer_2, kernel_initializer=c
onfig.kernel_initializer_2))
    model.add(BatchNormalization())
    model.add(Activation(config.activation_2))
    model.add(Dropout(config.Dropout_2))
    model.add(Dense(units=config.layer_3, kernel_initializer=c
onfig.kernel_initializer_3))
    model.add(BatchNormalization())
    model.add(Activation(config.activation_3))
    model.add(Dropout(config.Dropout_3))
    wandb.log({"n_neurons_hidden": np.sum(config.layer_1+confi
g.layer_2+config.layer_3)})
if config.n_layers==4:
    model.add(Dense(units=config.layer_1, kernel_initializer=c
onfig.kernel_initializer_1))
    model.add(BatchNormalization())
    model.add(Activation(config.activation_1))
    model.add(Dropout(config.Dropout_1))
    model.add(Dense(units=config.layer_2, kernel_initializer=c
onfig.kernel_initializer_2))
    model.add(BatchNormalization())

```

```

        model.add(Activation(config.activation_2))
        model.add(Dropout(config.Dropout_2))
        model.add(Dense(units=config.layer_3, kernel_initializer=c
onfig.kernel_initializer_3))
        model.add(BatchNormalization())
        model.add(Activation(config.activation_3))
        model.add(Dropout(config.Dropout_3))
        model.add(Dense(units=config.layer_4, kernel_initializer=c
onfig.kernel_initializer_4))
        model.add(BatchNormalization())
        model.add(Activation(config.activation_4))
        model.add(Dropout(config.Dropout_4))
        wandb.log({"n_neurons_hidden": np.sum(config.layer_1+confi
g.layer_2+config.layer_3+config.layer_4)})
        if config.n_layers==5:
            model.add(Dense(units=config.layer_1, kernel_initializer=c
onfig.kernel_initializer_1))
            model.add(BatchNormalization())
            model.add(Activation(config.activation_1))
            model.add(Dropout(config.Dropout_1))
            model.add(Dense(units=config.layer_2, kernel_initializer=c
onfig.kernel_initializer_2))
            model.add(BatchNormalization())
            model.add(Activation(config.activation_2))
            model.add(Dropout(config.Dropout_2))
            model.add(Dense(units=config.layer_3, kernel_initializer=c
onfig.kernel_initializer_3))
            model.add(BatchNormalization())
            model.add(Activation(config.activation_3))
            model.add(Dropout(config.Dropout_3))
            model.add(Dense(units=config.layer_4, kernel_initializer=c
onfig.kernel_initializer_4))
            model.add(BatchNormalization())
            model.add(Activation(config.activation_4))
            model.add(Dropout(config.Dropout_4))
            model.add(Dense(units=config.layer_5, kernel_initializer=c
onfig.kernel_initializer_5))
            model.add(BatchNormalization())
            model.add(Activation(config.activation_5))
            model.add(Dropout(config.Dropout_5))
            wandb.log({"n_neurons_hidden": np.sum(config.layer_1+confi
g.layer_2+config.layer_3+config.layer_4+config.layer_5)})

        model.add(Dense(units=3, kernel_initializer=config.kernel_in
itializer_6))
        model.add(BatchNormalization())
        model.add(Activation("softmax"))

```

```

if config.optimizer=='adam':
    opt=keras.optimizers.Adam(learning_rate=config.learning_rate)

if config.optimizer=='SGD':
    opt=keras.optimizers.SGD(learning_rate=config.learning_rate)

if config.optimizer=='adamax':
    opt=keras.optimizers.Adamax(learning_rate=config.learning_rate)

if config.optimizer=='nadam':
    opt=keras.optimizers.Nadam(learning_rate=config.learning_rate)

model.compile(loss=config.loss, optimizer=opt, metrics=['accuracy', 'AUC'])

model.fit(X_train, y_train, batch_size=config.batch_size, epochs=config.epoch, validation_data=(X_test,y_test), callbacks=[WandbCallback()], class_weight=class_weights)

y_test_arg = np.argmax(y_test,axis=1)
Y_pred = np.argmax(model.predict(X_test),axis=1)
cm = ConfusionMatrix(actual_vector=y_test_arg, predict_vector=Y_pred)

cm_holdout=cm.matrix
cm_holdout_copy=cm.matrix
cm_holdout=ConfusionMatrix(matrix=cm_holdout)
cm_holdout_copy=ConfusionMatrix(matrix=cm_holdout_copy)
cp = Compare({"cm1":cm_holdout,"cm2":cm_holdout_copy})
cp_overall = cp.scores.get('cm1').get('overall')
cp_class = cp.scores.get('cm1').get('class')
cp_total = cp_overall + cp_class

wandb.log({"ConfusionMatrix": cm.to_array()})
wandb.log({"Class_stat": cm.class_stat})
wandb.log({"Overall_stat": cm.overall_stat})
wandb.log({"ACC_Macro": cm.ACC_Macro})
wandb.log({"Overall_ACC": cm.Overall_ACC})
wandb.log({"F1_Macro": cm.F1_Macro})
wandb.log({"SOA6 (Matthews's benchmark)": cm.SOA6})
wandb.log({"Zero-one loss": cm.ZeroOneLoss})
wandb.log({"Hamming loss": cm.HammingLoss})
wandb.log({"AUC_class_0": cm.AUC[0]})
wandb.log({"AUC_class_1": cm.AUC[1]})

```

```

wandb.log({"AUC_class_2": cm.AUC[2]})
wandb.log({"Cross_Entropy": cm.CrossEntropy})
wandb.log({"Kullback-Leibler divergence": cm.KL})
wandb.log({"Cp_total": cp_total})
wandb.log({"Cp_overall": cp_overall})
wandb.log({"Cp_class": cp_class})

# Define the K-fold Cross Validator
skf = StratifiedKFold(n_splits=5, shuffle=True, random_state
=123)

# K-fold Cross Validation model evaluation
ACC_Macro_per_fold = []
Hamming_loss_per_fold = []
F1_Macro_per_fold = []
Cross_Entropy_per_fold = []
AUC_class_0_per_fold = []
AUC_class_1_per_fold = []
AUC_class_2_per_fold = []
Kullback_Leibler_divergence_per_fold = []
cp_total_per_fold = []
cp_overall_per_fold = []
cp_class_per_fold = []

fold_no = 1
for train, test in skf.split(X, y):

    sc = StandardScaler()
    X[train] = sc.fit_transform(X[train])
    X[test] = sc.transform(X[test])

    y_train_categorical = to_categorical(y[train],3)
    y_test_categorical = to_categorical(y[test],3)

    model_K = Sequential()

    if config.n_layers==1:
        model_K.add(Dense(units=config.layer_1, kernel_initializ
er=config.kernel_initializer_1))
        model_K.add(BatchNormalization())
        model_K.add(Activation(config.activation_1))
        model_K.add(Dropout(config.Dropout_1))
        wandb.log({"n_neurons_hidden": np.sum(config.layer_1)})
    if config.n_layers==2:
        model_K.add(Dense(units=config.layer_1, kernel_initializ
er=config.kernel_initializer_1))
        model_K.add(BatchNormalization())
        model_K.add(Activation(config.activation_1))

```

```

        model_K.add(Dropout(config.Dropout_1))
        model_K.add(Dense(units=config.layer_2, kernel_initializer=
config.kernel_initializer_2))
        model_K.add(BatchNormalization())
        model_K.add(Activation(config.activation_2))
        model_K.add(Dropout(config.Dropout_2))
        wandb.log({"n_neurons_hidden": np.sum(config.layer_1+con
fig.layer_2)})
        if config.n_layers==3:
            model_K.add(Dense(units=config.layer_1, kernel_initializ
er=config.kernel_initializer_1))
            model_K.add(BatchNormalization())
            model_K.add(Activation(config.activation_1))
            model_K.add(Dropout(config.Dropout_1))
            model_K.add(Dense(units=config.layer_2, kernel_initializ
er=config.kernel_initializer_2))
            model_K.add(BatchNormalization())
            model_K.add(Activation(config.activation_2))
            model_K.add(Dropout(config.Dropout_2))
            model_K.add(Dense(units=config.layer_3, kernel_initializ
er=config.kernel_initializer_3))
            model_K.add(BatchNormalization())
            model_K.add(Activation(config.activation_3))
            model_K.add(Dropout(config.Dropout_3))
            wandb.log({"n_neurons_hidden": np.sum(config.layer_1+con
fig.layer_2+config.layer_3)})
        if config.n_layers==4:
            model_K.add(Dense(units=config.layer_1, kernel_initializ
er=config.kernel_initializer_1))
            model_K.add(BatchNormalization())
            model_K.add(Activation(config.activation_1))
            model_K.add(Dropout(config.Dropout_1))
            model_K.add(Dense(units=config.layer_2, kernel_initializ
er=config.kernel_initializer_2))
            model_K.add(BatchNormalization())
            model_K.add(Activation(config.activation_2))
            model_K.add(Dropout(config.Dropout_2))
            model_K.add(Dense(units=config.layer_3, kernel_initializ
er=config.kernel_initializer_3))
            model_K.add(BatchNormalization())
            model_K.add(Activation(config.activation_3))
            model_K.add(Dropout(config.Dropout_3))
            model_K.add(Dense(units=config.layer_4, kernel_initializ
er=config.kernel_initializer_4))
            model_K.add(BatchNormalization())
            model_K.add(Activation(config.activation_4))
            model_K.add(Dropout(config.Dropout_4))

```

```

        wandb.log({"n_neurons_hidden": np.sum(config.layer_1+config.layer_2+config.layer_3+config.layer_4)})
        if config.n_layers==5:
            model_K.add(Dense(units=config.layer_1, kernel_initializer=config.kernel_initializer_1))
            model_K.add(BatchNormalization())
            model_K.add(Activation(config.activation_1))
            model_K.add(Dropout(config.Dropout_1))
            model_K.add(Dense(units=config.layer_2, kernel_initializer=config.kernel_initializer_2))
            model_K.add(BatchNormalization())
            model_K.add(Activation(config.activation_2))
            model_K.add(Dropout(config.Dropout_2))
            model_K.add(Dense(units=config.layer_3, kernel_initializer=config.kernel_initializer_3))
            model_K.add(BatchNormalization())
            model_K.add(Activation(config.activation_3))
            model_K.add(Dropout(config.Dropout_3))
            model_K.add(Dense(units=config.layer_4, kernel_initializer=config.kernel_initializer_4))
            model_K.add(BatchNormalization())
            model_K.add(Activation(config.activation_4))
            model_K.add(Dropout(config.Dropout_4))
            model_K.add(Dense(units=config.layer_5, kernel_initializer=config.kernel_initializer_5))
            model_K.add(BatchNormalization())
            model_K.add(Activation(config.activation_5))
            model_K.add(Dropout(config.Dropout_5))
            wandb.log({"n_neurons_hidden": np.sum(config.layer_1+config.layer_2+config.layer_3+config.layer_4+config.layer_5)})

        model_K.add(Dense(units=3, kernel_initializer=config.kernel_initializer_6))
        model_K.add(BatchNormalization())
        model_K.add(Activation("softmax"))

        if config.optimizer=='adam':
            opt=keras.optimizers.Adam(learning_rate=config.learning_rate)

        if config.optimizer=='SGD':
            opt=keras.optimizers.SGD(learning_rate=config.learning_rate)

        if config.optimizer=='adamax':
            opt=keras.optimizers.Adamax(learning_rate=config.learning_rate)

```

```

if config.optimizer=='nadam':
    opt=keras.optimizers.Nadam(learning_rate=config.learning
_rate)

    model_K.compile(loss=config.loss, optimizer=opt, metrics=[
'accuracy', 'AUC'])

    model_K.fit(X[train], y_train_categorical, batch_size=config.batch_size, epochs=config.epoch, validation_data=(X[test],y_test_categorical), class_weight=class_weights)

    y_test_arg = np.argmax(y_test_categorical,axis=1)
    Y_pred = np.argmax(model_K.predict(X[test]),axis=1)
    cm = ConfusionMatrix(actual_vector=y_test_arg, predict_vector=Y_pred)

    cm_kfold=cm.matrix
    cm_kfold_copy=cm.matrix
    cm_kfold=ConfusionMatrix(matrix=cm_kfold)
    cm_kfold_copy=ConfusionMatrix(matrix=cm_kfold_copy)
    cp = Compare({"cm1":cm_kfold,"cm2":cm_kfold_copy})
    cp_overall = cp.scores.get('cm1').get('overall')
    cp_class = cp.scores.get('cm1').get('class')
    cp_total = cp_overall + cp_class
    cp_total_per_fold.append(cp_total)
    cp_overall_per_fold.append(cp_overall)
    cp_class_per_fold.append(cp_class)

    ACC_Macro_per_fold.append(cm.ACC_Macro)
    Hamming_loss_per_fold.append(cm.HammingLoss)
    F1_Macro_per_fold.append(cm.F1_Macro)
    Cross_Entropy_per_fold.append(cm.CrossEntropy)
    AUC_class_0_per_fold.append(cm.AUC[0])
    AUC_class_1_per_fold.append(cm.AUC[1])
    AUC_class_2_per_fold.append(cm.AUC[2])
    Kullback_Leibler_divergence_per_fold.append(cm.KL)

if fold_no==1:
    cm_1 = cm.to_array()
    wandb.log({"ConfusionMatrix_fold_no_1": cm.to_array()})
    wandb.log({"Class_stat_fold_no_1": cm.class_stat})
    wandb.log({"Overall_stat_fold_no_1": cm.overall_stat})

if fold_no==2:
    cm_2 = cm.to_array()
    wandb.log({"ConfusionMatrix_fold_no_2": cm.to_array()})
    wandb.log({"Class_stat_fold_no_2": cm.class_stat})
    wandb.log({"Overall_stat_fold_no_2": cm.overall_stat})

```



```

if fold_no==3:
    cm_3 = cm.to_array()
    wandb.log({"ConfusionMatrix_fold_no_3": cm.to_array()})
    wandb.log({"Class_stat_fold_no_3": cm.class_stat})
    wandb.log({"Overall_stat_fold_no_3": cm.overall_stat})

if fold_no==4:
    cm_4 = cm.to_array()
    wandb.log({"ConfusionMatrix_fold_no_4": cm.to_array()})
    wandb.log({"Class_stat_fold_no_4": cm.class_stat})
    wandb.log({"Overall_stat_fold_no_4": cm.overall_stat})

if fold_no==5:
    cm_5 = cm.to_array()
    wandb.log({"ConfusionMatrix_fold_no_5": cm.to_array()})
    wandb.log({"Class_stat_fold_no_5": cm.class_stat})
    wandb.log({"Overall_stat_fold_no_5": cm.overall_stat})

fold_no = fold_no + 1

wandb.log({"ACC_Macro_per_fold_mean": np.mean(ACC_Macro_per_fold)})
wandb.log({"ACC_Macro_per_fold_std": np.std(ACC_Macro_per_fold)})
wandb.log({"Hamming_loss_per_fold_mean": np.mean(Hamming_loss_per_fold)})
wandb.log({"Hamming_loss_per_fold_std": np.std(Hamming_loss_per_fold)})
wandb.log({"F1_Macro_per_fold_mean": np.mean(F1_Macro_per_fold)})
wandb.log({"F1_Macro_per_fold_std": np.std(F1_Macro_per_fold)})
wandb.log({"Cross_Entropy_per_fold_mean": np.mean(Cross_Entropy_per_fold)})
wandb.log({"Cross_Entropy_per_fold_std": np.std(Cross_Entropy_per_fold)})
wandb.log({"AUC_class_0_per_fold_mean": np.mean(AUC_class_0_per_fold)})
wandb.log({"AUC_class_0_per_fold_std": np.std(AUC_class_0_per_fold)})
wandb.log({"AUC_class_1_per_fold_mean": np.mean(AUC_class_1_per_fold)})
wandb.log({"AUC_class_1_per_fold_std": np.std(AUC_class_1_per_fold)})
wandb.log({"AUC_class_2_per_fold_mean": np.mean(AUC_class_2_per_fold)})

```

```

wandb.log({"AUC_class_2_per_fold_std": np.std(AUC_class_2_per_fold)})
wandb.log({"Kullback-Leibler_divergence_per_fold_mean": np.mean(Kullback-Leibler_divergence_per_fold)})
wandb.log({"Kullback-Leibler_divergence_per_fold_std": np.std(Kullback-Leibler_divergence_per_fold)})
wandb.log({"cp_total_per_fold_mean": np.mean(cp_total_per_fold)})
wandb.log({"cp_total_per_fold_std": np.std(cp_total_per_fold)})
wandb.log({"cp_overall_per_fold_mean": np.mean(cp_overall_per_fold)})
wandb.log({"cp_overall_per_fold_std": np.std(cp_overall_per_fold)})
wandb.log({"cp_class_per_fold_mean": np.mean(cp_class_per_fold)})
wandb.log({"cp_class_per_fold_std": np.std(cp_class_per_fold)})

ConfusionMatrix_per_fold_sum = np.add(cm_1, cm_2)
ConfusionMatrix_per_fold_sum = np.add(ConfusionMatrix_per_fold_sum, cm_3)
ConfusionMatrix_per_fold_sum = np.add(ConfusionMatrix_per_fold_sum, cm_4)
ConfusionMatrix_per_fold_sum = np.add(ConfusionMatrix_per_fold_sum, cm_5)
wandb.log({"ConfusionMatrix_per_fold_sum": ConfusionMatrix_per_fold_sum})

from time import time
start = time()
model.predict_classes(X_test)
record_time = time()-start
tf.compat.v1.disable_eager_execution()
trainable_count = int(np.sum([K.count_params(p) for p in set(model.trainable_weights)]))
non_trainable_count = int(np.sum([K.count_params(p) for p in set(model.non_trainable_weights)]))
total_params = trainable_count + non_trainable_count
wandb.log({"Total params": total_params})
wandb.log({"Trainable params": trainable_count})
wandb.log({"Time": record_time})

# Initialize a new sweep
# Arguments:

```

```
# - sweep_id: the sweep_id to run - this was returned above by wandb.sweep()
# - function: function that defines your model architecture and trains it
wandb.agent(sweep_id, train)
```