Machine Learning-based Classification of Online Industrial Datasets

Rastislav Fáber
Faculty of Chemical and Food Technology
Slovak University of Technology in Bratislava
Bratislava, Slovakia
rastislav.faber@stuba.sk

Karol L'ubušký
Slovnaft, a.s.
Bratislava, Slovakia
karol.lubusky@slovnaft.sk

Radoslav Paulen
Faculty of Chemical and Food Technology
Slovak University of Technology in Bratislava
Bratislava, Slovakia
radoslav.paulen@stuba.sk

***Abstract*—We aim to incorporate data analytics into industrial process control by utilizing machine learning (ML) algorithms to classify the real-time data of online analyzers. Real-time visualization of results onto a front-end system (i.e., refinery control room) provides an extensive view of the production process, increasing efficiency of production. Selected ML classifiers are assessed according to the performance metrics based on individual scores. These parameters, along with the complexity of implementation, provide an adequate pointer for selecting a suitable classifier model to serve as a decision-making tool. In our use case, accurate categorization of measurements provides a cheap validation guideline that would otherwise be not possible. Computed metrics indicate a difficulty to classify the cases when the slight deviations (drifts) occur from real values. Based on the true positivity rate, linear SVM separation is desirable for data drift prediction (64 %), while k-Means is more successful in detecting outliers (65 %) and normal operation (99 %).**

***Index Terms*—Machine Learning, Data Classification, Alkylation Process, Analytics, Industry 4.0**

I. Introduction

Systematic technological advancements constantly incite the development in digitalization, automation, and optimization of industrial processes. Industrial complexes, such as refineries, are at the forefront of technology to meet the ever-growing needs of operational demands (equipped with advanced sensors and regulated in real time), coining the term — the fourth industrial revolution; or Industry 4.0. However, certain parts of plant operation progress quite slowly as the industry is capital-intensive and the integrity of production is crucial [1], [2]. A refinery is a network of closely related unit processes that communicate with each other (material flows, data lines, etc.) to transform raw materials into more valuable products [3]. Incorporating computer science and artificial intelligence (i.e., machine learning algorithms) into the industrial sphere results in efficient utilization of available information and to its correct interpretation wherever needed. This paper reflects this modern-day trend by proposing a machine-assisted indication based on real-time industrial measurement classification.

The aim of this paper is to leverage stored (unused) data and provide more meaningful information about the process to the control room. Constant monitoring allows for timely actions and ensures steady production, as multiple sensors and process analyzers provide useful real-time measurements to indicate the current values of process variables (operating point). Implementation of advanced technologies ensures more sustainable operation with fewer required external interventions, and achieves higher yields and earnings. Data analytics provides an opportunity to uncover hidden patterns (i.e., parameter correlations); *(i)* providing more information to human operators in the plant control room for monitoring, *(ii)* enabling autonomous plant management via data-driven decision-making [4]. In this paper, we utilize well established and well understood classification algorithms to train a decision-making tool that classifies real-time data points.

TABLE I
Commonly used ML algorithms.

Algorithm	Principle	Ref
k-Means	*partitioning (centroid–based)*	[6]
DBSCAN	*density–based*	[7]
Decision Tree	*logic–based*	[8]
k-Nearest Neighbours (KNN)	*distance–based*	[9]
Support Vector Machine (SVM)	*statistical learning*	[10]
Neural Network	*deep learning*	[6]

Machine learning (ML) is a method of choice for data analytics as it does not require any complex specification of the production process to extract relevant data [5]. Since the efficacy of an ML algorithm depends not only on the type of a problem, but also on the quality of available data, it is often reasonable to train multiple models and verify whether they achieve satisfactory results. Also, we opt for an algorithm seamlessly implementable onto an industrial hardware that would be transparent for the plant operators.

Table I lists selected ML algorithms. k-Means is a well known clustering algorithm (unsupervised learning) which partitions data points into clusters based on the distance from the nearest cluster center [6]. DBSCAN is also a widely used clustering algorithm that classifies data based on density [7]. Decision Tree resembles a branching tree comprised of conditions and rules (nodes), based on which a dataset is classified with appropriate labels [8]. KNN uses k neighbours of each data point to assign the label based on the largest group of equally classified neighbors [9]. The SVM model can provide a simple linear classification (hyperplane decision

boundary), which presents a preferable solution for a real-time measurement verification — easily implementable in industry [10]. Neural Networks refer to layers of neurons, where one neuron comprises a weight, a bias, and an activation function to process the input (input layer), and outputs the computed output (output layer) [6].

We implement three selected ML algorithms for a specific data-based problem in an industrial refinery — the industrial partner, Slovnaft, a.s. — for real-time classification of measurements. Collected data represents measured weight concentration (% w/w) via three online analyzers, which exhibit (real-time) unidentifiable fluctuations. Analyzers are installed on isobutane material streams of an akylation unit within the refinery. Collected data points are targeted for Advanced Process Control (APC) as disturbance variables (DVs). Currently, the automatic control approach (reliable on laboratory values) is not usable in the intended way, and it is necessary to propose a method of real-time classification for all measurements. This concept would not only verify data points for APC, but also bring important information to the alkylation control room and serve as an early indication method.

II. Process Description

Alkylation, or alkylate production, is based on the combination of reactive C_3–C_4 olefins with i-C_4 isobutane to form more desirable high-octane (> 87) branched C_7–C_9 isoparaffins (alkylate) with superior blending properties — a key component for clean gasoline [3], [11], [12]. According to the simplified diagram (Fig. 1), a mixture of reactants (olefins and isobutane) is cooled down and fed into a reactor (three parallel units). For efficient production, it is crucial to maintain an optimal ratio of the reactants — online analyzers mounted on one i-C_4 storage stream and two i-C_4 recycle streams.

To eliminate the formation of unwanted by-products, contactors are refrigerated by a cooling cycle to control the temperature of the reaction mixture at a constant value of $5 – 10\,°C$ — as alkylation is an exothermic reaction. An expansion valve reduces the pressure of the refrigerant (cooling by expansion) which results in overall reduction in temperature. This type of a cooling cycle enables to use the product stream from contactors to serve as a refrigerant. The right-hand side of the diagram describes the compression of vapours taken from the separator for further utilization of liquid isobutane, as well as additional processing of low purity isobutane in the fractionator. Material stream of i-C_4 is recycled and final products are collected (including the alkylate). Undoubtedly, the amount of mixed i-C_4 is directly related to the amount of i-C_4 that is recycled which increases the demand of downstream (separation) units and reduces the profitability of the process.

The efficacy of industrial production is paramount as the quantity and composition of products depends on it. Among other parameters (temperature in the contactors, feed composition, etc.), alkylation reactions depend on the supply of isobutane in excess — i-C_4 is less soluble in the acid phase than olefins. Since the online analyzers show erratic behavior and do not provide reliable measurements (% w/w of i-C4 recycles), operators have to double-check them regularly — meaning the APC deployment is delayed/ineffective.

To achieve a noticeable improvement in production, it is necessary to eliminate this issue, as mixing ratio directly affects the qualitative characteristics (octane number) of the alkylate product. Currently, the implemented control strategy relies on a Robust Multivariable Predictive Control (Profit Controller or RMPCT). The optimal ratio of isobutane to olefins in the feed mixture is the setpoint for control. The recycle flow of i-C_4 is the manipulated variable (MV) to keep the % w/w value within the optimal limit [3].

III. Problem Definition

As current real-time measurements cannot be used for automatic control (anomaly in % w/w could result in undesirably shifting the mixing ratio), the control strategy is amended to a simple single-input and single-output (SISO) system. As a temporary solution, process operators require laboratory sampling to be more frequent than usual (% w/w value is frozen when an anomaly is detected) to manually calibrate the analyzers for more accurate ratio control. The concentration of olefins presents the same problem and is evaluated in the laboratory. All the aforementioned issues result in unnecessary action that leads to complications and most importantly leads to unnecessary expenses for frequent laboratory tests.

We propose a machine-assisted method of decision-making based on data classification (Fig. 2) that would allow us to classify real-time process measurements and inform the operator whether they are suitable for process control, or faulty and should be omitted — repeated faulty measurements require calibration of the online analyzers.

As shown in Figure 3, the comparison of laboratory samples (referred to as ground truth — red cross) with the online analyzer measurements (blue), we differentiate between two types of anomalies in the industrial dataset: *outliers* and *drifts*. We consider a data point as an *outlier* when a sudden steep jump from the current value of concentration is indicated. A drift is observable when the measured % w/w gradually deviates from the actual value (approximated with blue dashed line) over time. Based on these observations, calculating the difference in values of consecutive measurements should serve as a valid method for distinguishing between all cases. If a long-term deviation from the *normal* value is detected, the analyzer is manually calibrated to the most recent laboratory sample or a new sample is requested at the lab.

To develop a reliable decision-making tool that would serve as an early indicator for plant operators, we choose from a number of already existing classification algorithms to train a model that achieves the best performance in real-time measurement classification and at the same time is not difficult to implement on industrial hardware.

With three defined possible labels, we annotate the pre-processed dataset to make sure we compare the predicted indicator labels ($\hat{Y}$) with correctly labeled data points (Y). We opt for the *DBSCAN* algorithm [7]. DBSCAN is designed to separate distant points (noise) from more densely arranged

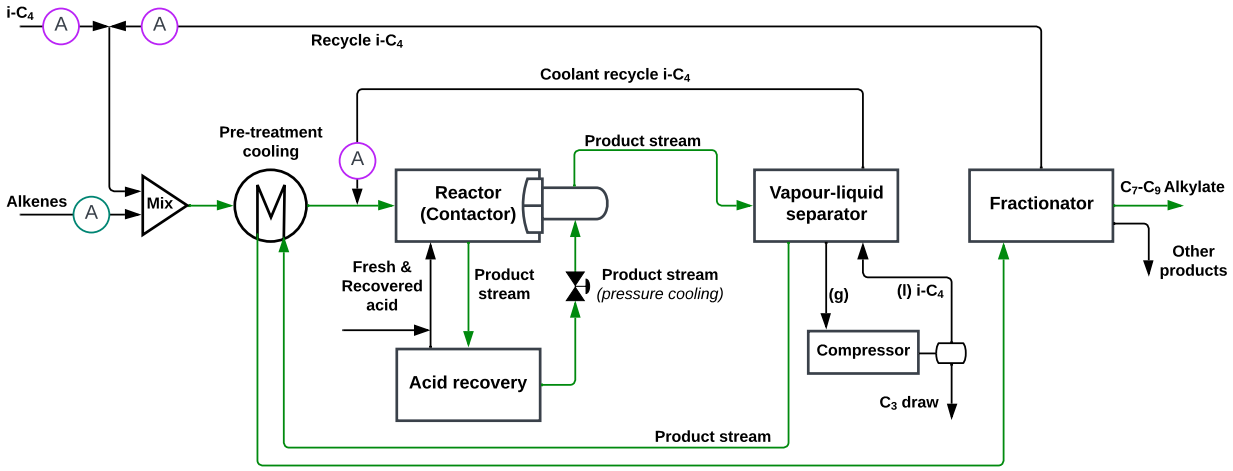

Fig. 1. A simplified diagram of an alkylation unit.

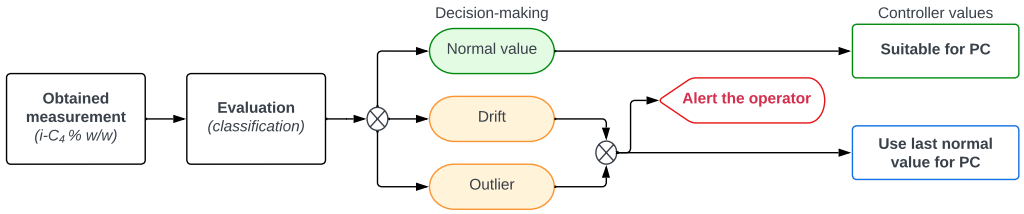

Fig. 2. Machine-assisted decision-making flowchart for alkylation unit(s) process control (PC).

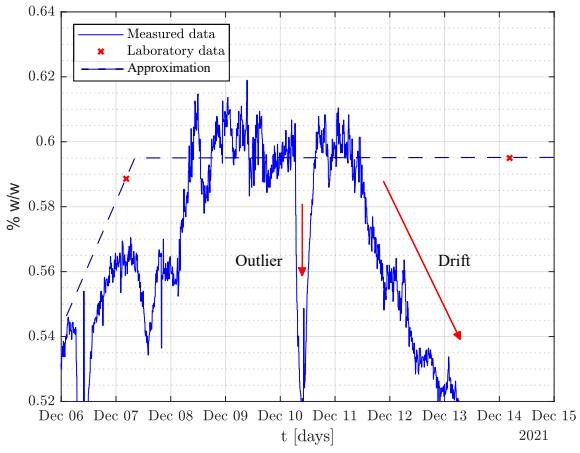

Fig. 3. Measured isobutane % w/w compared to laboratory samples (anonymized values).

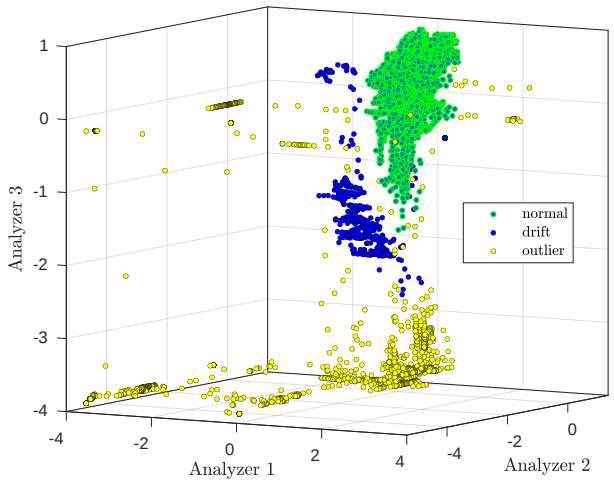

Fig. 4. Ground truth labels based on the DBSCAN algorithm.

data, based on two input parameters; *eps* regards the neighbourhood radius around a data point, and *minPts* defines the minimum number of points inside this radius. Dense regions are then identified based on the defined search radius (*eps*) within the dataset. With proper tuning, this method allows to reliably annotate all data points of the process based on the distance of individual clusters from the origin into the three pre-defined classes (visualized in Fig. 4).

IV. Data Pre-processing

The industrial dataset represents minutely weight concentration measurements from the past year of alkylate production. The total size exceeds 5×10^5 data points. Since the data tags are not annotated in any way (we have no information about possible shutdowns or other changes in operating conditions), we first transform raw data into workable format, while maintaining the overall integrity [13], [14]:

Data cleaning. Initially, it is necessary to filter out any

discrepancies from the raw dataset. Our primary focus is on evident missing or corrupted data points that must be filled/estimated or removed. Although removing a complete data point due to one missing value is not optimal, the percentage of missing data is not substantial to affect the features/trends of the dataset in any way.

Data transformation. We standardize the data as:

$$x_{i,stand} = \frac{1}{\sigma_{x,i}}(x_i - \bar{x}_i), \tag{1}$$

where σ_x represents the standard deviation vector of the dataset x with the mean value $\bar{x}$. This transformation is also used for plotting in this work to seal the data confidentiality.

Data reduction. An important part of data pre-processing is also the effort to minimize the time (computational complexity) when training the classification model — "indicator". Since we work with a higher number of data points, we use a fifteen-minute moving average — effectively reducing the data 15:1 (some information from the dataset could be suppressed).

Finally, we divide the standardised industrial data (three i-C_4 material streams) into a training set consisting of 80 % of the total data points and a testing set consisting of the remaining 20 % data points (based on random indexing). By doing so, we prevent possible over-training of the machine learning classifier by testing its efficacy on a different dataset.

V. Data Clustering

When it comes to clustering, there is no one-size-fits-all algorithm. For example, if the data points are not linearly separable, DBSCAN may not be able to accurately cluster the data — this could lead to complications when implementing methods such as linear SVM. In such cases, the use of alternative clustering algorithms should be considered; based on a careful analysis of the dataset and the properties of the data points. We employ the k-Means algorithm from MATLAB "*kmeans*" routine to partition the data into clusters. Each data point is assigned into a cluster based on the distance to the (current) nearest centroid (center of a cluster). We predetermine the optimal number of clusters by incorporating the elbow method (plotting the sum of the squared distance between points in a cluster and the cluster centroid) [15]. This approach indicates a suitable division of data points into four clusters (according to the breakpoint of the curve in Fig. 5), however, recent studies suggest to use alternative methods to choose the number of clusters [16].

Additionally, we compute a Silhouette score (s) for each cluster (i), to verify the distribution of data points as:

$$s_i = \frac{(b_i - a_i)}{\max(a_i, b_i)}, \tag{2}$$

where a represents the average distance between each point within a cluster; and b stands for the average inter-cluster distance, i.e., the average distance between all clusters.

The silhouette plot in Fig. 6 shows that the data is split into varying number of clusters (y-axis); each with its own s_i score. The width of each cluster in the four plots (y-axis) represents the total amount of data points within these clusters;

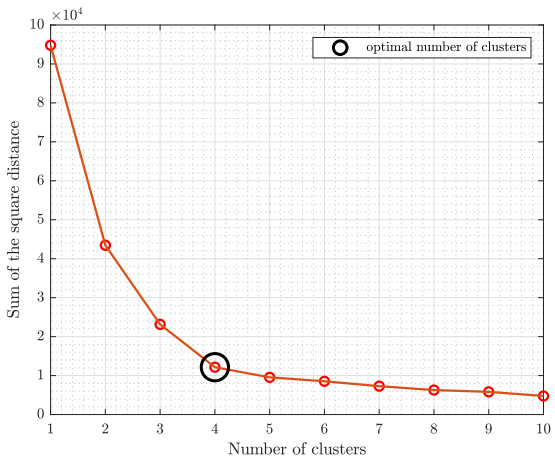

Fig. 5. Within-cluster squared distances function of the number of clusters.

TABLE II
AVERAGE SILHOUETTE SCORES CORRESPONDING TO THE CHOSEN NUMBER OF CLUSTERS.

No. of clusters	two	three	four	five
Average silhouette score	0.8350	0.7958	0.8198	0.3772

the corresponding silhouette coefficient refers to the distance to other clusters. When choosing the optimal number of clusters from the Silhouette plot, it is not sufficient to select the largest number from comparing the average scores from Table II; a verification of the following conditions is required:

1. all the clusters should have a Silhouette score greater than the average score of the dataset (red dashed line).
2. large fluctuations in cluster size should not be present.

For our application, the second rule does not apply, since the data points during normal operation significantly exceed the disturbances and drifts of the alkylation unit combined. From within these plots, we look for the largest, and at the same time

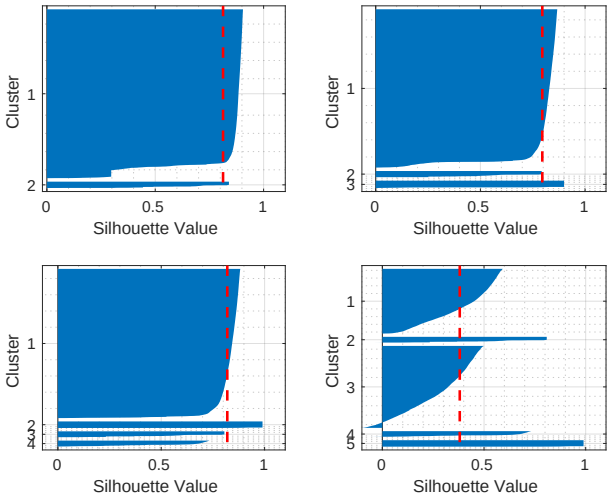

Fig. 6. A silhouette plot from the clustered training data.

comparable silhouette coefficients (preferably 0.8 or greater). The bottom left plot (four clusters) has the highest score, yet the first rule is not met; increased number of clusters results either in the distribution of outliers into multiple clusters, or results with data very close to the decision boundary. The top right plot (three clusters) appears acceptably close to the average silhouette score and shows smaller fluctuation in comparison; indicating proper cluster separation [15].

Upon evaluation, we conclude that the optimal number of clusters is three. Compared to results obtained by DBSCAN, the centroid for the *normal* cluster is correctly located around the mean value and the centroid for *outlier*s is amid the extreme values. The *drift* cluster centroid appears, however, suboptimal as it involves some outliers from the third analyzer.

VI. Data Classification

We train four different kinds of classification models using three well-understood machine-learning methods. Labeled data points (representing different clusters) from a clustering algorithm are used to train a classification model. To determine the quality of a trained classifier, an unknown dataset (testing data) is classified. To find a sutiable classification model, we calculate its performance metrics — recall, precision, specificity, accuracy, and F1-score [17]; as well as a confusion matrix. This provides a benchmark (criterion for selection) for a trained classifier and enables to evaluate its effectiveness.

1) k-Means & SVM: This model combines k-Means clustering and an SVM classifier. Linear SVM classifier is used as the resulting separation hyperplanes provide a transparent way of assessing classifier performance by industrial operators. The SVM model calculates parameters of a linear decision boundary (a hyperplane in a 3D space), that differentiates between two clusters. Based on the calculated parameters of the hyperplane, we classify a measurement into the defined clusters and label them by predefined categories (Fig. 7). To compute the SVM classifier, we solve the following optimization problem:

$$\min_{w,b,\xi\geq 0} \ \frac{1}{N}\sum_{i=1}^{N}\xi_i + \lambda w^{\mathrm{T}} w \tag{3a}$$

$$\text{s.t. } y_i\left(w^{\mathrm{T}}x_i - b\right) \geq 1-\xi_i, \ \forall i \in \{1,2,\ldots,N\}, \tag{3b}$$

where N defines the number of training data points, λ is a tunable parameter to obtain soft/hard-margin classifier with an offset b, y_i defines the label for each data point (commonly $y_i \in \{-1,1\}$), and w defines the normal vector to the hyperplane. The dataset is not linearly separable, therefore we introduce slack variables ξ_i into the objective function and relax the linear separability constraint [18]. We train two SVM separators to distinguish between the three classes.

2) k-Means & KNN: The initial step is to supply the positions of centroids (k-Means algorithm) to train the model. Complexity-wise, this model requires to store the training dataset for prediction, which is not desirable for industrial application. On the other hand, it should provide clear results for the uneven distribution of data points into clusters.

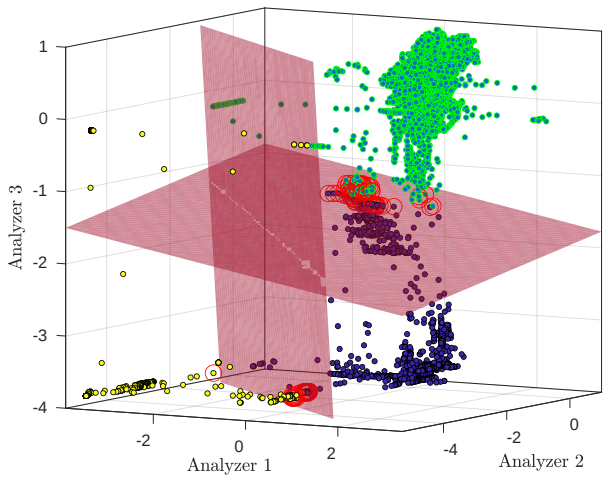

Fig. 7. Dataset divided into clusters (k-Means), then separated by hyperplanes (SVM).

3) k-Means: We use the k-Means algorithm for classification by itself. We base the prediction on the Euclidean distance between each data point (testing dataset) and the centroid point of each of the k clusters (MATLAB "*pdist2*" routine).

4) k-Means & SVM + diff: We re-use the k-Means & SVM approach, yet we include the backward differences between consecutive time points among the dataset. This approach aims to improve the classification performance of the *drift* data as they exhibit an outstanding slope patterns (see Fig. 3).

VII. Achieved Results

For classification evaluation, there are four cases of prediction success. E.g., for evaluation of the *normal* label — true positivity (TP) corresponds to the values correctly predicted as *normal*; false positivity (FP) corresponds to *normal* data point being mispredicted as either *drift* or *outlier*; false negativity (FN) corresponds to either *drift* or *outlier* being mispredicted as *normal*; true negativity (TN) corresponds to the remaining predictions (*drift* and *outlier* being correctly predicted). Analogical assignments can be made for *drift* and *outlier*.

There is a variety of possible performance metrics to choose from, and there is no general rule to prioritize one over others; the same model can excel in one metric but underperform in a different one [17], [19], [20]. **Recall (TP rate)** represents correctly predicted positives w.r.t. all positives; TP/(TP+FN). **Specificity (TN rate)** represents correctly predicted negatives w.r.t. all negatives; TN/(TN+FP). **Precision** reveals the rate of truly positive predictions from those predicted as positive; TP/(TP+FP). **Accuracy** represents the correctly classified measurements; (TP+TN)/(TP+FP+FN+TN). **F1 score** provides a more explicit performance representation as it is calculated as a harmonic average of Recall and Precision.

Numerical results listed in Table III represent the performance metrics (rows) for each classified label (n = *normal*, d = *drift*, o = *outlier*), and each model (columns). In our case study, specificity (n) and Recall (d and o) are used for evaluating the efficacy of trained classifiers. Overall, it is less

TABLE III
PERFORMANCE METRICS OF TRAINED CLASSIFIERS.

	SVM	k-NN	k-Means	SVM + diff	
Accuracy	**0.9629**	0.9628	0.9532	0.9525	*n*
	0.9538	0.9534	0.9315	0.9431	*d*
	0.9656	0.9653	**0.9693**	0.9656	*o*
Recall	0.9827	0.9828	**0.9997**	0.9709	*n*
	0.6296	0.6260	0.3527	**0.6426**	*d*
	0.6465	0.6364	**0.6566**	0.5253	*o*
Precision	**0.9755**	0.9752	0.9498	0.9752	*n*
	0.9048	0.9043	**1.0000**	0.7813	*d*
	0.2105	0.2072	**0.2355**	0.1857	*o*
Specificity	**0.8157**	0.8136	0.6059	**0.8157**	*n*
	0.9922	0.9922	**1.0000**	0.9787	*d*
	0.9696	0.9694	**0.9732**	0.9711	*o*
F1-score	**0.9791**	0.9790	0.9741	0.9730	*n*
	0.7425	0.7399	0.5214	0.7052	*d*
	0.3176	0.3127	**0.3467**	0.2744	*o*

cumbersome for the plant operators, when the *normal* label is classified as an *outlier* (the setpoint is set to the last correct value), opposed to the classification of an *outlier* as *normal* (the process diverges and the operator is not notified).

Specificity of k-Means (0.6059) for *normal* cases states that the classifier failed to predict *drifts* and *outliers*, and misclassified 60 % of total TN measurements as *normal* values. The SVM + diff model achieved the best *drift* detection, while the k-Means classifier is more effective in predicting *outliers*, based on recall (*d* and *o*). While comparing the models among themselves, KNN did not dominate in any metric but gave consistent results towards the higher end of overall performance, and never underperformed. SVM model with differences was expected to outperform the regular SVM classifier, however, we can see the opposite in every metric other than *drift* detection. This result could be theoretically improved by optimizing the clustering method to better reflect the ground truth. It is, therefore, inconclusive whether any of the proposed classifiers is suitable for this particular dataset just yet. APC variables need to be classified correctly, one possibility could involve a voting strategy between the least complex models. Other possibility involves treating the analyzers values separately, which would increase the automated validation complexity moderately.

Lasly, we test whether the initial clustering of training data by DBSCAN instead of k-Means can increase the indicators performance. Expectedly, a significant improvement can be achieved in detecting both anomalies — Specificity (TN rate); 96 % (SVM + diff model). The Recall metric (TP rate) remained around 60 %. These results point to inappropriateness of the linear SVM classifier, despite general industrial preference for linear models. A resolution might lie in involving further process variables among the classification features.

VIII. CONCLUSIONS

In this paper, the aim was to utilize historical industrial data, and unveil any hidden information to the alkylation unit operators. We studied multiple ML algorithms and selected four, based on the method of clustering, to be comprehensive. This paper should create a firm starting point in data analytics and provide room for further optimization of used methods, and to follow up on the achieved results with searching for more effective methods. Results indicate the difficulty to correctly predict drifts from the normal operation. Based on the TP rate, SVM model with differences predicts 64 % of all drifts correctly, while the k-Means classifier predicts outliers (65 %) and normal operation (99 %) most reliably. The prediction of both anomalies combined is 82 % performed by both SVM models.

ACKNOWLEDGMENTS

This research is funded by the Slovak Research and Development Agency under the projects APVV-21-0019, by the Scientific Grant Agency of the Slovak Republic under the grant VEGA 1/0691/21, and by the European Commission under the grant no. 101079342 (Fostering Opportunities Towards Slovak Excellence in Advanced Control for Smart Industries).

REFERENCES

[1] Thumeera R. Wanasinghe, Ray Gosine, Lesley James G.K.I. Mann, Oscar De Silva, and Peter J. Warrian. The internet of things in the oil and gas industry: A systematic review. *IEEE*, 2020.
[2] Hutama A. Bramantyo, Bagus Satrio Utomo, and Efrilia M. Khusna. Data processing for iot in oil and gas refineries. *J. Commun. Netw.*, 2022.
[3] James G Speight. *The refinery of the future*. Gulf Professional Publishing, Elsevier, 2020.
[4] Tyler Wall, CFE Media and Technology. How to get started with industrial data analytics, 2022.
[5] Muzammil khan, Salman Raza Naqvi, Zahid Ullah, Syed Ali Ammar Taqvi, Muhammad Nouman Aslam Khan, Wasif Farooq, Muhammad Taqi Mehran, DagmarJuchelková, and Libor Štěpanec. Applications of machine learning in thermochemical conversion of biomass-a review. *Fuel*, 332:126055, 2023.
[6] Batta Mahesh. Machine learning algorithms -a review. *IJSR*, 2019.
[7] Martin Ester, Hans-Peter Kriegel, Jiirg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. 1996.
[8] Michael D Twa, Srinivasan Parthasarathy, Cynthia Roberts, Ashraf M Mahmoud, Thomas W Raasch, and Mark A Bullimore. Automated decision tree classification of corneal shape. *Optometry and Vision Science*, 82(12):1038—1046, 2005.
[9] Aized Amin Soofi and Arshad Awan. Classification techniques in machine learning: Applications and issues. *J. Basic Appl.*, 13:459–465, 2017.
[10] Muzammil Khan Muhammad Taqi Mehran, Zeeshan Ul Haq, Zahid Ullah, Salman Raza Naqvi, Mehreen Ihsan, and Haider Abbass. Applications of artificial intelligence in covid-19 pandemic: A comprehensive review. *Expert Systems with Applications*, 185:115695, 2021.
[11] Sven IvarHommeltoft. Isobutane alkylation: Recent developments and future perspectives. *Applied Catalysis A: General*, 221(1):421–428, 2001.
[12] Fuels PALL Corporation and Chemicals. Refineries: Application focus - h2so4 alkylation unit, 2018.
[13] Brian Malley and Daniele Ramazzotti and Joy Tzung-yu Wu . *Data Pre-processing*, pages 115–141. 2016.
[14] Max Kuhn and Kjell Johnson. *Data Pre-processing*, pages 27–59. 2013.
[15] Anmol Tomar. Stop using elbow method in k-means clustering, instead, use this!, 2022.
[16] Erich Schubert. Stop using the elbow criterion for k-means and how to choose the number of clusters instead, 2022.
[17] Salma Ghoneim. Accuracy, recall, precision, f-score & specificity, which to optimize on?, 2019.
[18] Corinna Cortes and Vladimir Vapnik. Support-vector networks, 1995.
[19] Naeem Seliya, Taghi M. Khoshgoftaar, and Jason Van Hulse. A study on the relationships of classifier performance metrics. *21st IEEE ICTAI*, 2009.
[20] Bradley J. Erickson and Felipe Kitamura. Performance metrics for machine learning models. *PubMed Central*, 2021.