ACS OMEGA

Open Access

http://pubs.acs.org/journal/acsodf

Article

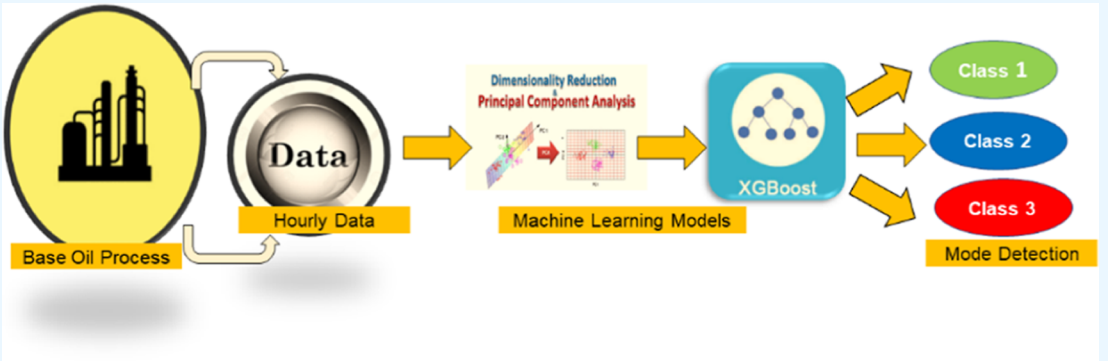
Investigative Analysis of Automatic Mode Detection for a Lubricant Base Oil Production Plant Using PCA and Machine-Learning Models

Muhamad Amir Mohd fadzil, Adi Aizat Razali, Haslinda Zabiri,* and Amar Haiqal Che Hussin

Cite This: *ACS Omega* 2024, 9, 3525−3540

Read Online

ACCESS | Metrics & More | Article Recommendations



ABSTRACT: Lubricants are important fluids and are commonly used to suppress friction between two metallic surfaces and as a medium for heat transportation. In an industrial plant considered in this study, the base oil mode changes can only be detected based on the kinematic viscosity values obtained using lab analysis. Since the lab analysis data are only available every 8 h, detecting the change in the production modes for 4, 6, and 10 cSt and the transitions among them are significantly delayed, causing unnecessary off-spec products that have to be directed to the slopping tank. In this paper, the innovativeness of the work comes from the idea of trying to unravel the underlying pattern of the plant data that correlate to the changes in the base oil modes and using that to classify hourly the kinematic viscosity values. Hence, a novel industrial application is presented to predict the class of base oil mode change on an hourly basis that can significantly reduce the losses in terms of off spec products and sloping tank wastes. The modes are segregated into three classes based on the values of kinematic viscosity. The classes are C-1 (4 cSt), C-2 (6 cSt), and C-3 (10 cSt). Anything in between the stipulated thresholds is called transition [T-12 (C-1 to C-2), T-21(C-2 to C-1), T-23 (C-2 to C-3), T-31 (C-3 to C-1), and T-32 (C-3 to C-2)]. To unravel the pattern, principal component analysis (PCA) is utilized on 42,000 operating plant data. After a thorough analysis, the third principal component provides the highest correlation to the eight classes of the base oil mode changes [C-1 (4 cSt), C-2 (6 cSt), and C-3 (10 cSt) and the transitions T-12 (C-1 to C-2), T-21(C-2 to C-1), T-23 (C-2 to C-3), T-31 (C-3 to C-1), and T-32 (C-3 to C-2)]. This third principal component is then utilized together with plant process variable values as inputs to four machine learning models, namely, XGBOOST, Random Forest, and CatBoost algorithms to predict the mode of the base oil hourly. The overall comparison analysis shows that utilizing the XGBoost algorithm for the prediction of the eight classes of the base oil modes at a faster hourly rate results in the most consistent classification accuracy of 92.96% for the test set and 89.22% in the deployment set. This capability to predict the mode change in the hourly basis can significantly reduce the losses in terms of off spec products in the production line.

1. INTRODUCTION

Lubricants are important fluids and are commonly used to suppress the friction between two metallic surfaces and as a medium for heat transportation. These base oils are obtained from various sources such as petroleum raffinates,[1] biobased oils,[2] olefins,[3] and plastic wastes,[4] with some additives to enhance their properties. A few examples of processes used to produce the base oils are solvent extraction,[5] severe hydrocracking,[6] olefin polymerization,[7] and esterification.[8]

In an industrial plant involving severe hydrocracking for producing three modes of base oil 4, 6, and 10 cSt, the information on the current mode being processed is only made available via lab sample analysis. However, the lab sampling has its own limitation, as it is done only every 8 h due to the

Received: September 22, 2023
Revised: December 16, 2023
Accepted: December 22, 2023
Published: January 11, 2024

https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540

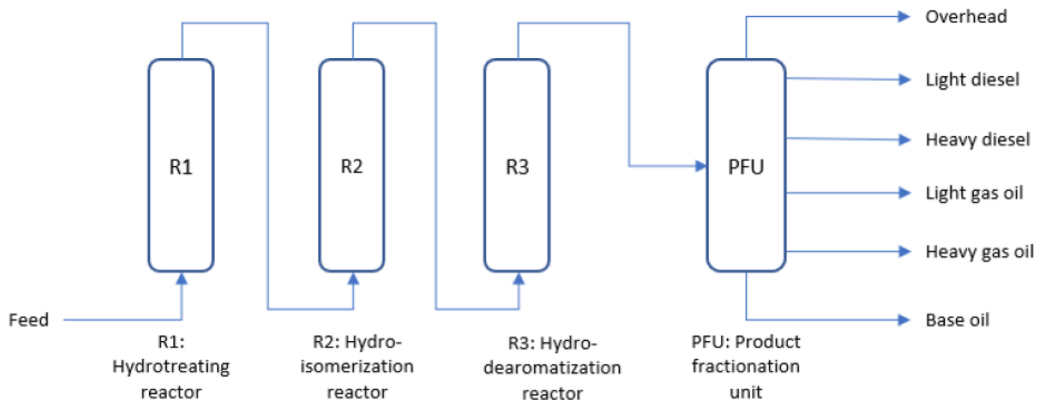


Figure 1. Simplified process flow diagram of the base oil processing plant from ref 10. Copyright [2021] [M. A. M. Fadzil et al.].

significant time consumption for the analyses. The most significant change in the product quality is during the transition between the 10 cSt product and 4 cSt product. During this transition period, the product is diverted into a sloping tank (off-specs). In the current practice, the decision to end the transition period is made based on the laboratory analysis results which is normally only available at the 10th hour from the start of the transition. Due to this, significant delay is encountered in the decision making causing useful products to be diverted unnecessarily into the slopping tank.

Hence, in this paper, the innovativeness of the work comes from the idea of trying to unravel the underlying pattern of the plant data that correlates to the changes in the base oil modes and using that to predict hourly the kinematic viscosity values. This hourly prediction is very crucial as the existing plant operation has to rely solely on the lab analysis, which is available only every 8 h, causing unnecessary losses due to off-spec products that has to be directed to the slopping tank. Hence, a novel industrial application is presented to predict the base oil mode change on an hourly basis that can significantly reduce the losses in terms of off spec products and sloping tank wastes.

The modes are segregated into three classes based on the values of kinematic viscosity. The classes are C-1 (4 cSt), C-2 (6 cSt), and C-3 (10 cSt). Anything in between the stipulated thresholds is called transition [T-12 (C-1 to C-2), T-21(C-2 to C-1), T-23 (C-2 to C-3), T-31 (C-3 to C-1), and T-32 (C-3 to C-2)]. To unravel the pattern, the authors utilized principal component analysis (PCA) on 42,000 operating plant data.[9] After a thorough analysis, the third principal component provides the highest correlation to the eight classes of the base oil mode changes [C-1 (4 cSt), C-2 (6 cSt), C-3 (10 cSt), and the transitions T-12 (C-1 to C-2), T-21(C-2 to C-1), T-23 (C-2 to C-3), T-31 (C-3 to C-1), and T-32 (C-3 to C-2)]. This third principal component is then utilized together with process variables values as inputs to four machine learning models, namely, XGBOOST, Random Forest, and CatBoost algorithms to predict the mode of the base oil hourly.

2. LITERATURE REVIEW

2.1. Base Oil Process Description. The data for this investigative study are retrieved from a production plant that produces base oil from a severe hydrocracking process. The hydrocracking process involves three series of reactions which are hydrotreating, hydroisomerization, and hydrodearomatization. The feedstock used originates from the waxy raffinate obtained from crude oil atmospheric distillation. This waxy raffinate is then distilled under vacuum condition to obtain three different grades with the final product kinematic viscosities of 4 cSt (C-1), 6 cSt (C-2), and 10 cSt (C-3) ranges. In the hydrotreating process, the feedstock is fed into the hydrotreating reactor to reduce the sulfur and nitrogen contents to an acceptable level. The hydrotreating reactor also cracks the hydroisomerization reaction, where the base oil molecules are isomerized to lower its pour point. Next, the product undergoes the hydrodearomatization process to increase the oxidative stability of the base oil product oxidative stability. Finally, the product is distilled under vacuum conditions to remove the

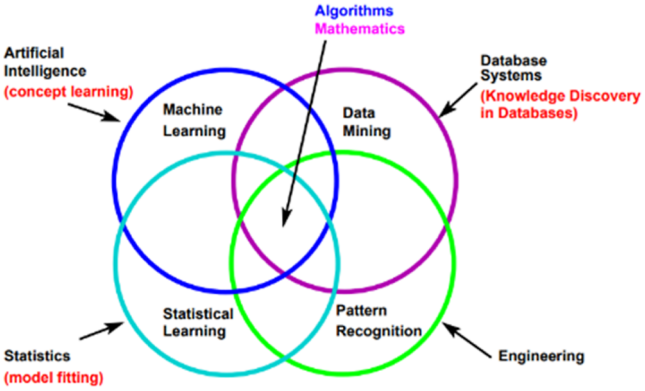


Figure 2. Venn-diagram describing the multi domain view of machine learning adapted from lecture slides by ref 12. Copyright [2021] [Liviu Ciortuz Department of Computer Science, University of Iasi, Romania].

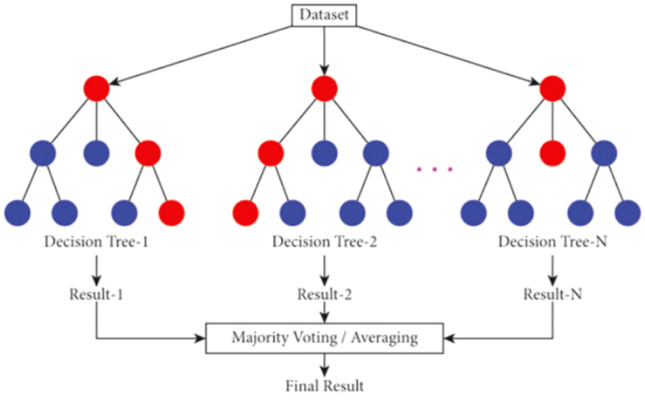


Figure 3. Illustration of random forest tree adapted with permission from ref 25. Copyright [2021] [M.Y. Khan et al.].

https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540

lighter materials. Figure 1 shows a simplified flow diagram of the industrial base oil processing plant taken from ref 10.

2.2. Machine Learning Algorithms. Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy. Machine learning is a teamwork between multiple disciplines with each of them strive to solve the formulated general learning problem in their individual own way. The general learning problem is described by ref 11 as "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance a tasks in T, as measured by P, improves with experience E". Figure 2 illustrates the multidomain view of machine learning connections with other relevant disciplines of study.

There are two broad categories of machine learning problems: supervised and unsupervised learning. Supervised learning, also known as supervised machine learning, is a subcategory of machine learning and artificial intelligence. It is defined by its use of labeled data sets to train algorithms to classify data or predict outcomes accurately. Some examples of supervised machine learning are support vector regression (SVR),[13] decision tree regression (DTR),[14] and extreme gradient boosting (XGBoost).[15] Gradient boosting is a supervised learning algorithm that attempts to accurately predict a target variable by combining the estimates of a set of simpler, weaker models. It is a popular supervised-learning algorithm used for the regression and classification on large data sets. It uses sequentially built shallow decision trees to provide accurate results and a highly scalable training method that avoids overfitting.

Unsupervised learning is the training of a machine using information that is neither classified nor labeled and allowing the algorithm to act on that information without guidance. Unlike supervised learning, no training is given to the machine. Therefore, the machine is restricted to finding the hidden structure in unlabeled data by itself. Some examples of unsupervised learning algorithms include K-means clustering,[16] PCA,[17] and hierarchical clustering.[18]

2.2.1. XGBOOST Algorithm. The supervised learning algorithm XGBoost models the connection between input features and output labels using decision trees.[19] The algorithm creates a group of decision trees, each of which is trained to predict the residual error of the one before it. The total of all trees' predictions makes up the final forecast.[20]

For predictive monitoring, XGBoost has been used in a number of areas including fault diagnosis, anomaly detection, and quality prediction. XGBoost has been used in fault diagnosis to find and identify faults in mechanical systems, such as bearings and turbines.[21] XGBoost has been used in anomaly detection to identify an unusual behavior in a range of systems, including computer networks, production lines, and transportation systems,[22] as well as for quality prediction method a variety of manufacturing processes, including the production of semiconductors and steel.[23]

2.2.2. Random Forest Algorithm. Random forest (RF) is an ensemble learning method employed in classification, regression, and other tasks that operates by constructing a multitude of decision trees.[24] A randomly selected subset of the training data and features are used to construct each decision tree, as shown in Figure 3. The final prediction is created by combining the outputs of the trees either through voting or averaging. Due to its ability for dealing with high-dimensional data, nonlinear relationships, and interactions between features, RF is one of the popular machine learning algorithms.[24]

Numerous industries have made an extensive use of RF for applications involving predictive maintenance of bearings in rotating machinery,[26] and failure detection of wind turbine gearboxes.[27] It is a popular option for predictive maintenance applications due to its capacity to handle high-dimensional data, nonlinear relationships, and interactions between features.

2.2.3. CatBoost Algorithm. CatBoost is a gradient boosting algorithm that models the association between input features and output labels using decision trees.[28] The algorithm handles categorical data by one-hot encoding them into numerical values and then employing an ordered boosting variation of gradient boosting. In order to increase the model's precision and speed, the algorithm also makes use of techniques like gradient-based sampling and feature importance calculation.

For predictive monitoring, CatBoost has been used in a number of areas including fault diagnosis, anomaly detection, and quality prediction. CatBoost has been used in fault diagnosis to find and identify faults in a variety of systems, including engines, pumps, and bearings,[29] in anomaly detection to identify an unusual behavior in a variety of systems, including computer networks, manufacturing processes, and financial systems[30] and to predict product quality in a number of manufacturing processes, including the production of food and semiconductors.[41]

3. METHODOLOGY

The general methodology for executing this study is illustrated in Figure 4. The first step is collecting plant data from the plant information system (PI).

3.1. Data Set Descriptions. The original data set consists of approximately 42,000 data points of hourly process variable values (excluding any lab samples) collected from an industrial base oil processing facility. Historical data from January 1, 2016 to June 29, 2020 were used in the training of the models. The preprocessed data are then split into 70% training set and 30% validation set. An additional out-of-sample data set from June 29, 2020 to January 13, 2021, that has not been used during the machine learning models training/validation runs, is utilized as the test set to get a better estimation of the generalization performance of the models.

The data set consists of 40 input variables and 1 output variable (the classes of the product grades). Table 1 lists the input variables considered in this study. Note that variable t_3 is the third principal component derived from PCA, which is described in detail in Section 3.2. Tank level data are also added to check whether the level change from the feed tank to the first reactor (hydrotreating reactor) in Figure 1 would have any impact in facilitating the mode change detection. These feed tanks come in pairs for each product grade, i.e., Tank1a and Tank1b refer to the feedstock for C-1, Tank2a and b for C-2, and Tank3a and b for C-3.

The collected data consist of erratic values that need to be removed. These issues may occur due to faulty sensors, connections error, tag no longer existing, and uncalibrated instruments. Prior to model development, data cleaning is performed to remove all of the error values. The input which has erratic values of more than 50% of the total data is removed, while inputs with faulty readings below that are replaced by the median of the input value, resulting in the final data of 10,000 points. This step is important to ensure PCA can provide accurate results in discovering the underlying patterns that the

https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540

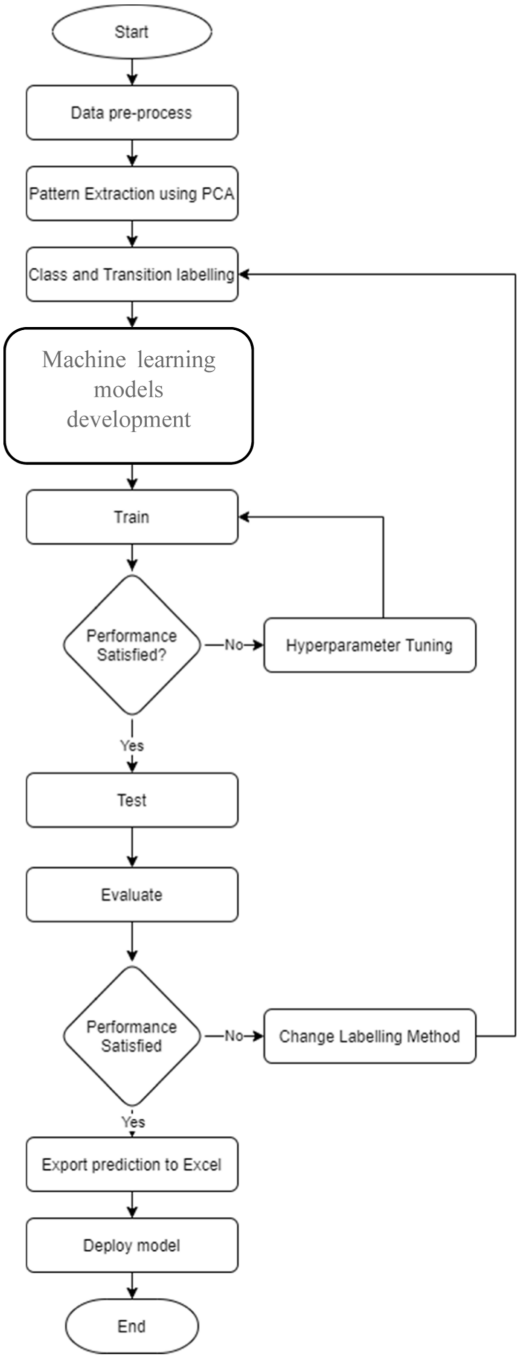


Figure 4. Overall methodology.

data holds. The final data set is then cleaned from outliers by using the z score method.[31]

The kinematic viscosity dictates the class of the hydrocarbon. The range of the hydrocarbon classes is defined as follows:

C-1: 3.50 ≤ kinematic viscosity ≤ 4.80

C-2: 5.40 ≤ kinematic viscosity ≤ 7.40

C-3: 8.91 ≤ kinematic viscosity ≤ 13.00

C-1, C-2, and C-3 are considered the main classes. Four transitions between each class are also being considered and labeled as T12 for transition from Class 1 and Class 2, T21 for transition from Class 2 and Class 1, T23 for transition from Class 2 and Class 3, and T31 for transition from Class 2 and Class 3.

Table 1. List of Input Variables Used in This Study

	Input Variables		
1	HDT_ART	21	LBO_DrawTemp
2	HDT_PInlet	22	MPSteamFlow rate
3	HDT_POutlet	23	Diesel_Drawrate
4	ISO_ART	24	PD_TopPA
5	ISO_Pinlet	25	PD_PA
6	ISO_POutlet	26	PD_FlowtoStorageTank
7	HDA_ART	27	PD_BedTemp
8	HDA_Pinlet	28	MD_Drawrate
9	HDT_Poutlet	29	MD_Bed_Temp
10	Feed_Flow rate	30	HD_Drawrate
11	UCOfeed_Flow rate	31	HD_BedTemp
12	PFU_Vacuum	32	LBO_Drawrate
13	PFU_MPSteamflow rate	33	LBODrawTemp
14	PFU_OvhdTemp	34	Tank1a.Level
15	MD_Flow rate	35	Tank1b.Level
16	C1901_TopTemp	36	Tank2a.Level
17	C1905_TopTemp	37	Tank2b.Level
18	F1903_Outlet_Temp	38	Tank3a.Level
19	C1905_OverflashTemp	39	Tank3b.Level
20	C1905_OverflashFlow rate	40	t_3
	Output Variable		
1	product class		

3.2. Pattern Extraction Using Principal Component Analysis. PCA is one of the statistical methods that is used to find and recognize underlying patterns in these high dimensional data. PCA is a dimensionality reduction technique used to obtain a smaller data set consisting only of important features from the model inputs with a set of related variables via linear features learning. In this study, the PCA model is used to generate principal components (features). The decomposition of the model input data matrix $\mathbf{x}$ via PCA is mathematically expressed in eq 1:

$$\mathbf{x} = \mathbf{T}\mathbf{P}^{\mathrm{T}} + \mathbf{E} \tag{1}$$

where $\mathbf{T}$ is the score matrix with a dimension of $m \times k$, $\mathbf{P}$ is the loading matrix with a dimension of $n \times k$, $\mathbf{E}$ is the residual matrix with a dimension of $m \times n$, and k is the total number of retained principal components.

PCA is the oldest technique[32] of multivariate data analysis which is being widely used. PCA is a dimensionality reduction technique used to reduce the dimension of the input data set into a new data set which preserves as much statistical information (variability) as possible for further analysis purposes.[33] During the dimensionality reduction process, the algorithm strives toward finding a new set of variables that are uncorrelated with each other with maximum variance. These variables, which are also termed the principal components, are the linear representations of features in the original data set.

In this paper, SIMCA Free Trial software[34] is used to analyze the data set via PCA to take advantage of the excellent features available on SIMCA. The scores calculated are then extracted and manually plotted. The PCA is applied on the data set consisting of hourly process variable values as listed in Table 1. This multivariate data analysis using PCA is initially utilized to extract the key principal components that capture the key variations in the data with respect to mode changes in the base oil and the transitions between modes. In order to develop these principal components, eigenvalues and eigenvector problems are to be solved. Three PCA analyses are done, namely, the first

https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540

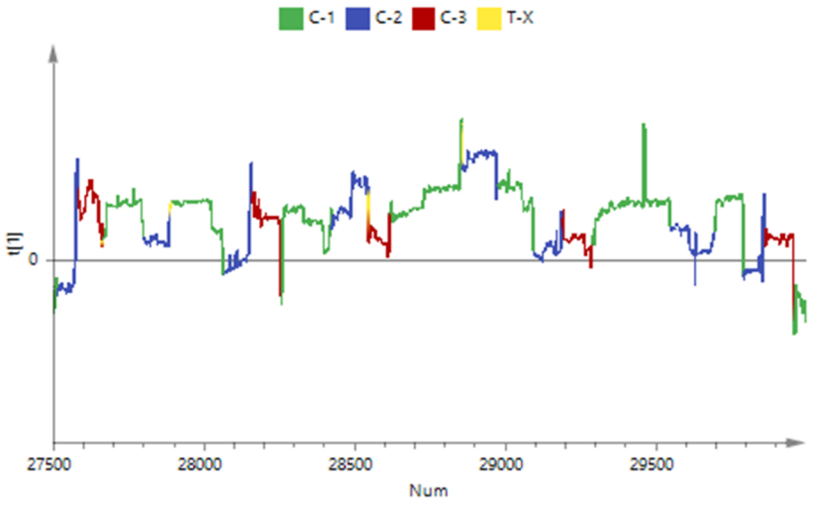


Figure 5. *t_1* score plot of the data.

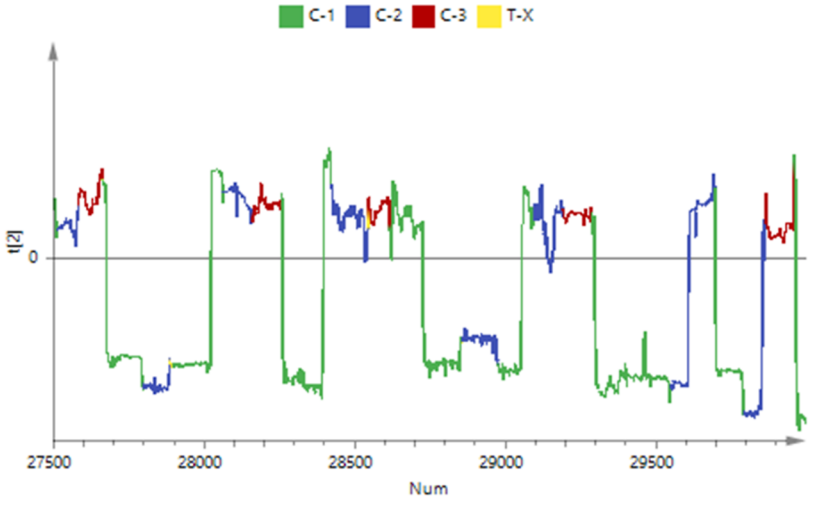


Figure 6. *t_2* score plot of the data.

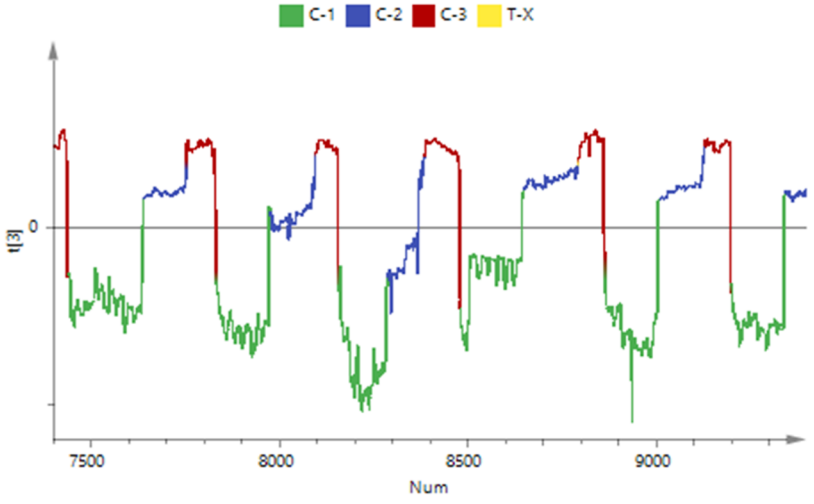


Figure 7. *t_3* score plot of the data.

principal component (*t_1*), the second principal component (*t_2*), and the third principal component (*t_3*).

Figure 5 shows the line plot for *t_1* scores colored according to the base oil classes. The *y*-axis represents the *t_1* scores, and the *y*-axis represents the hourly real data points value. At this level, the class segregation of C-1, C-2, and C-3 is not obvious. Upon validation with plant operators, major variations captured by this first principal component are dominated by the plant

https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540

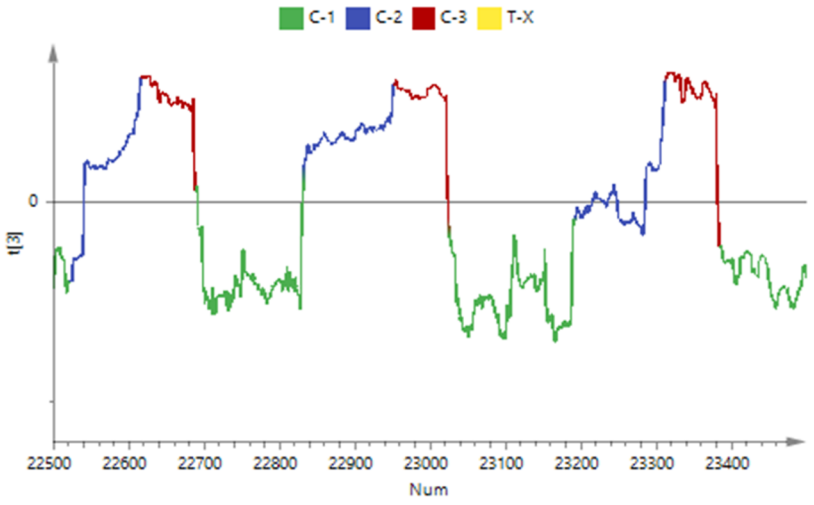


Figure 8. Close up of the C-1 to C-2 transition on the *t_3* plot.

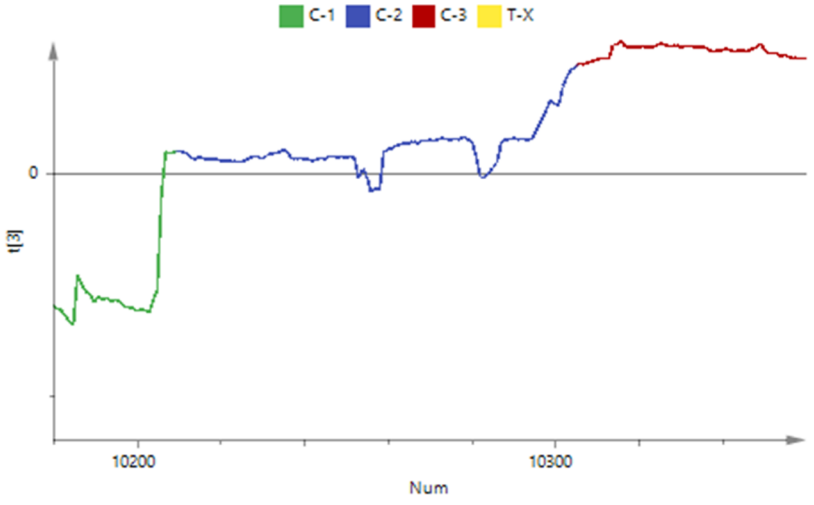


Figure 9. Close up of C-2 to C-3 transition on the *t_3* plot.

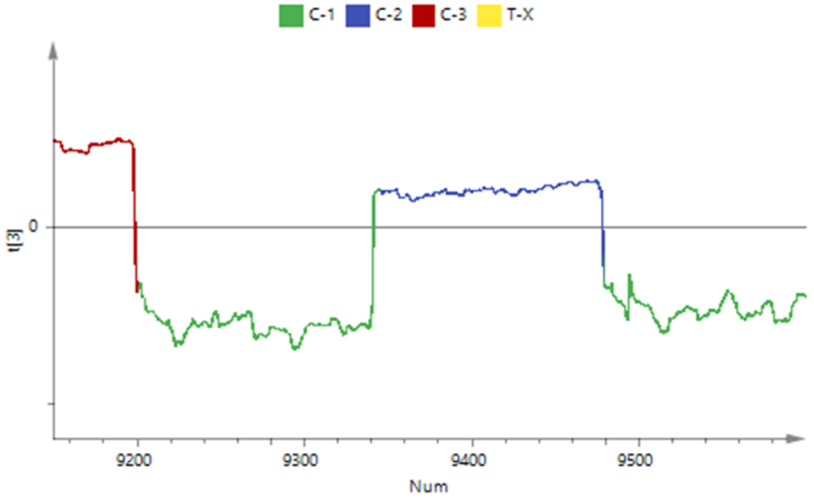


Figure 10. Close up of C-3 to C-1 transition on the *t_3* plot.

changes due to plants shutdown. Hence, the analysis proceeds to *t_2* analysis, as shown by Figure 6. At this level, the class segregation has become relatively clearer with the emergence of two regions, positive and negative, on the *t_2* score axis. However, there is still mixed-up during transitions, which results in additional classes to emerge. This makes labeling for machine learning training (C-1, C-2, and C-3) not yet possible.

https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540

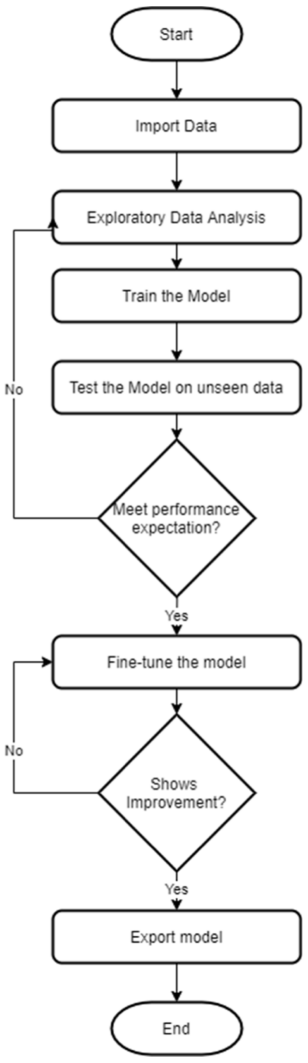


Figure 11. Flowchart of developing the machine learning classification model.

The transitions between modes can be observed satisfactorily using the third principal component *t*_3, as shown in Figure 7. The colors are defined based on the range of kinematic viscosity of the hydrocarbon taken at each data point. It is observed that class transitions from C-1 to C-2 and vice versa are usually followed by a sudden fluctuation in the value of *t*_3. The same observation applies to C-2 to C-3 and vice versa, where there is a subtle fluctuation of *t*_3 in between. Hence it can be concluded that *t*_3 score plots show the best representation (with significant observable changes) for the mode transitions. Further close-up analyses shown in Figures 8−10 preserved the same trends. For example, the transition between data points 9340 and 9350 in Figure 8 shows how the *t*_3 score values experienced a sudden and very steep score increase before the class changes from C-1 to C-2. Similarly, the same pattern is observed in Figures 9 and 10.

3.3. Class and Transitions Labeling. The modes obtained from the third PCA based on only hourly data process variables are carefully labeled as C-1, C-2, and C-3 to identify the main modes of 4, 6, and 10 cSt. The third principal component *t*_3 is chosen because it can capture most of the transitions/mode variations as shown in the previous section. Using the information obtained from this third principal component plot, the transition modes/classes are then identified accordingly as T-12 (Class 1 to Class 2), T-21(Class 2 to Class 1), T-23(Class 2 to Class 3), T-31(Class 3 to Class 1), and T-32(Class 3 to Class 2) in the 42,000 data sets.

The class and transition labeling is crucial for any supervised machine learning algorithm as it functions as the reference for the algorithm to learn and compute the prediction later on. Once the data labeling is completed, the data are then split into a 70:30 ratio, where 70% of the data is used for training/validation and 30% for external testing or deployment to test the generalization capability of the developed model. 80% of that 70% is used for training and the remaining 20% is used for validation. All of the remaining 30% from the first ratio split is used as real test data (or referred to as deployment data). Then, machine learning models are developed from the data. The model with the most satisfactory performance is deployed to predict the classes of the testing data.

Table 2. Data Pairing for Model Development

data set	main data	tank level data
benchmark	10,000 data set	not used
	41 variables (40 inputs and 1 output)	
Run 1	10,000 data set	default tank level
	41 variables (40 inputs and 1 output)	Pseudo Tank1a
		Pseudo Tank1b
		Pseudo Tank2a
		Pseudo Tank2b
		Pseudo Tank3a
		Pseudo Tank3b
Run 2	4 highest correlated features (above 53%)	default tank level
		Pseudo Tank1a
		Pseudo Tank1b
		Pseudo Tank2a
		Pseudo Tank2b
		Pseudo Tank3a
		Pseudo Tank3b
Run 3	5 correlative features	default tank level
		Tank1a
		Tank1b
		Tank2a
		Tank2b
		Tank3a
		Tank3b

3.4. Developing the Machine Learning Model for Mode Classification. The steps of developing the machine learning model are illustrated in Figure 11.

Table 2 provides a detailed description of the characteristics of the data set used to build the machine learning models. Each pairing is decided by leveraging process knowledge, trial and error, and tuning of the algorithm to reach the optimum accuracy. There are four data sets considered: benchmark, Run 1, Run 2, and Run 3. For Run 2 and Run 3, a heat map is utilized (see Figure 12) to determine the correlative values among the variables in order to select the most significant ones. In Run 2, the variables with correlation values higher than 53% are chosen; meanwhile, for Run 3, feed flow rate is added as the additional data. The respective correlative values for the selected variables are shown in Table 3.

Machine learning models are then developed using Python for each data set listed in Table 2 and the performance is analyzed. Hyper-parameters are optimized via trial and error methods, and the best performance for each XGBoost, Random Forest, and

https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540

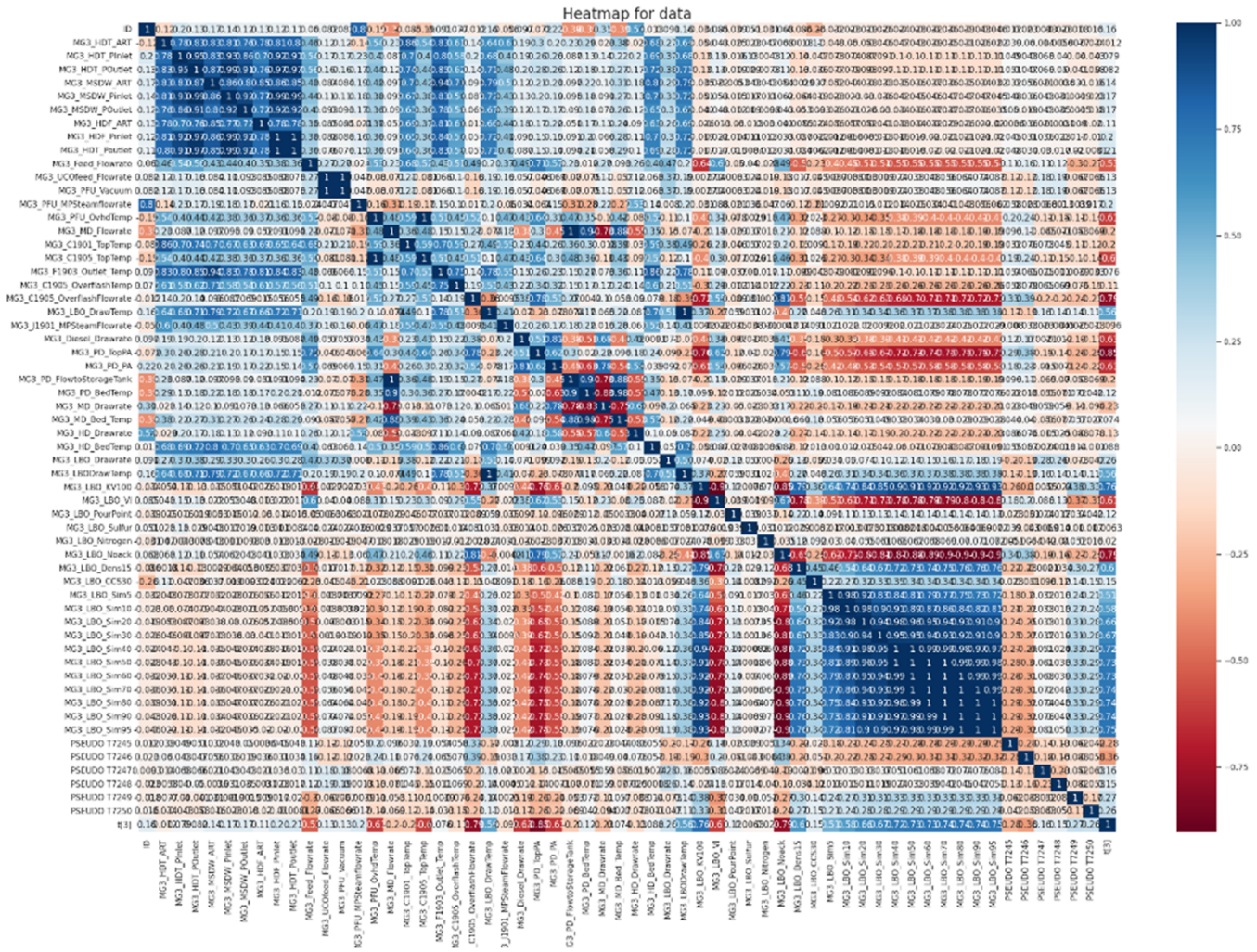


Figure 12. Pearson correlation heat map on the data set.

Table 3. Correlative Values for Variables Selected for Run 2 and Run 3

variable name	correlative value
Run 2 Selected Variables	
C1905_OverflashFlow rate	0.699507
PD_TopPA	0.716433
PD_PA	0.532214
*t*_3	0.704034
Run 3 Selected Variables	
C1905_OverflashFlow rate	0.699507
PD_TopPA	0.716433
PD_PA	0.532214
*t*_3	0.704034
Feed_Flow rate	0.524323

CatBoost models are recorded. The settings of the model parameters are summarized in Table 4.

The performance of each model is reflected by the confusion matrices and the average accuracy (ACC). ACC is calculated as the number of all correct predictions divided by the total number of the data set.[35] The best accuracy is 1.0, whereas the worst is 0. It can also be calculated by the 1 − error (ERR). Specifically, ACC is calculated as shown by eq 2:

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \quad (2)$$

where TP is the number of true positives, TN is the number of true negatives, FP is the number of false positives, and FN is the number of false negatives.

4. RESULTS AND DISCUSSION

4.1. Model Performance Comparison Analysis. For classification, a confusion matrix is used to evaluate the performance of the machine learning models. As mentioned previously, each model is tested using four different data sets, and the corresponding accuracy is recorded. To ease the analysis, the corresponding representation used in the confusion matrix for each of the product class is summarized in Table 5.

4.1.1. Benchmark Data Performance Analysis. The confusion matrices for the benchmark data are presented in both real values and percentages, as shown in Figure 13. It can be observed that with benchmark data, all the models work well in classifying the base oil into its three main classes which are C-1 (i.e., Class 0), C-2 (i.e., Class 1), and C-3 (i.e., Class 2). However, all three models XGBoost, Random Forest, and CatBoost have low accuracies in classifying the transition classes especially in class T-21 (i.e., Class 4), i.e., the transition from C-1 to C-2. This could be due to the fact that in the 10,000 data set

https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540

Table 4. Model Parameter Settings

Parameters	Value
	XGBoost
Alpha	0
Booster	gbtree (uses tree models)
Gamma	0 (making the algorithm nonconservative)
Lambda	0
learning_rate	0.3
max_depth	6
normalize_type	tree (i.e., trees have the same weight of each of dropped trees)
num_parallel_tree	1
n_jobs	1
Objective	"multi:softprob"
n_estimators	100
	Random Forest
max_features	Sqrt max_features = sqrt(n_features)
n_estimators	100
max_leaf nodes	none unlimited number of leaf nodes
min_sample_leaf	1
min_sample split	2
n_jobs	1
random_state	none the state is not randomized
oob_score	false the default accuracy_score is used.
	CatBoost
Iterations	1000
Depth	6
learning_rate	0.1
loss_function	"MultiClass"
Verbose	200

Table 5. Corresponding Representation of Confusion Matrix Numbering and the Classification

confusion matrix numbering	class
0	C-1 (main)
1	C-2 (main)
2	C-3 (main)
3	T-12 (transition)
4	T-21 (transition)
5	T-23 (transition)
6	T-31 (transition)

used in this study, the transition T-21 has the least data in both the training and testing sets.

It can be observed also from Figure 13 that the XGBoost model surpasses the performance of both Random Forest and CatBoost algorithms as higher percentages (as well as real values) of correct detection are shown in the diagonal matrix of the XGBoost model. As typical in any machine learning application, the efficiency of a model type highly dependent on the nature of the data being analyzed.[36] In this study, it is apparent that XGBoost is able to classify the base-oil mode change data more accurately in comparison with the other two models.

4.1.2. Run 1 Data Performance Analysis. From Figure 14, it can be observed that with run 1 data all the models work almost similarly well in classifying the base oil into its three main classes. Similarly, the CatBoost also has difficulty in classifying the T-21 (i.e., Class 4). Only XGBoost and Random Forest models are able to correctly classify this transition to some extent with the accuracy of 25%.

4.1.3. Run 2 Data Performance Analysis. As described in Section 3.2, Run 2 uses only variables with correlation values higher than 53% as determined from the Pearson correlation map. As shown in Figure 15, all three models are able to accurately classify the main classes C-1, C-2, and C-3. Reducing the number of variables, however, affects the classification accuracy for the transition class T-21 (Class 4). The lack of data for this particular transition mode proves to cause difficulty for most models to properly identify the class.

4.1.4. Run 3 Data Performance Analysis. Using Run 3 data set, a similar behavior observed in the previous sections is observed as shown in Figure 16.

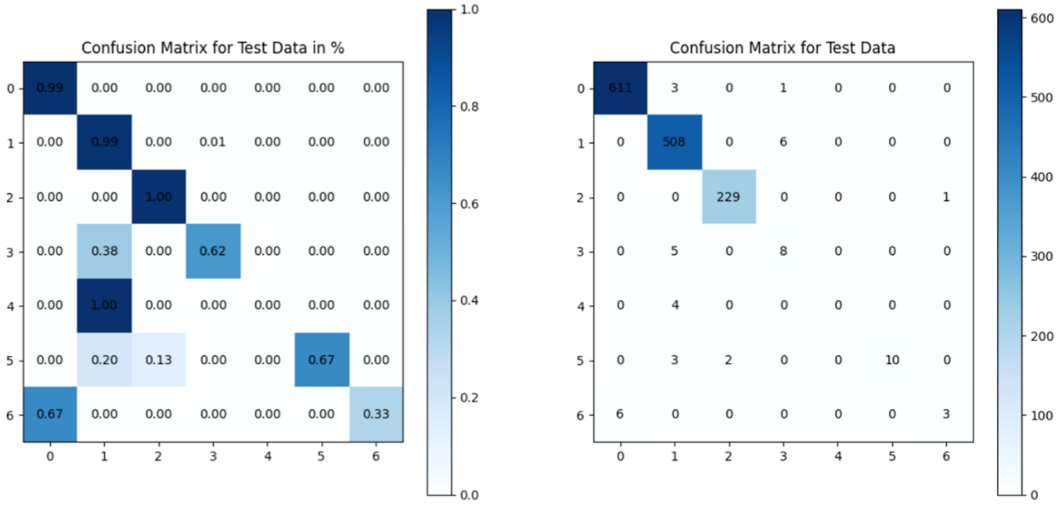
4.2. Overall Analysis and Discussion. From the analysis of the model performance comparison in Section 4.1 and as summarized in Table 6, it is shown that XGBoost models are able to retain satisfactory performances of over 90% accuracy when tested with different sets of data. As one of the gradient-based ensemble classifiers, XGBoost increases its confidence in classification gradually and thus make it possess greater classification power in comparison to standard neural network model in general.[36] CatBoost is another variant of gradient boosting techniques that has been equipped with "prediction shift" shield by the introduction of permutation-driven alternatives and innovative algorithm for processing categorical features. These additional features have given CatBoost the ability to perform unbiased boosting.[37] On the other hand, Random Forest is a classic example of ensemble classifier which on its own is a "black box" model which made it not be able to be visualized and hence made it unexplainable. To overcome this, it works together with a white box decision tree model as a weak learner. In general, all three of these machine learning models can be applied for classification problems and can perform reasonably well. It will be significantly subjective according to the nature of the data and other case-by-case criteria to determine which model is better.[38,39]

The final XGBoost model is then tested for its generalization capability using the test data set (or deployment data set). It is observed from the confusion matrices in Figure 17a that the performance of the model in predicting the modes are consistent for all the different data sets especially for the main classes C-1, C-2, and C-3. However, it is apparent from the confusion matrices that when the transitions are taken into consideration, the addition of tank level data is helpful, as shown in Figure 17b. It is observed that with tank level data addition, the model manages to have a correct prediction for T-21 class (Class 4).

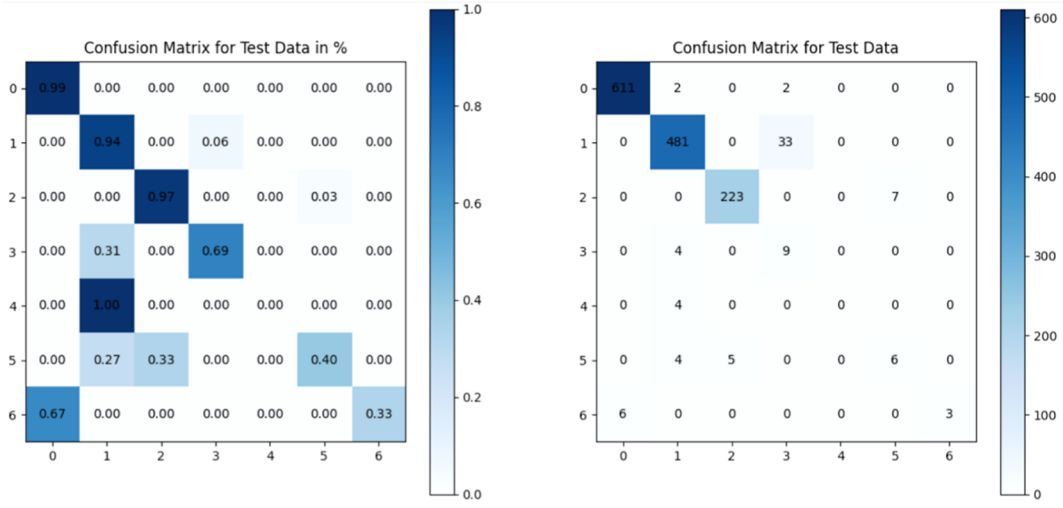
5. CONCLUSIONS

This paper considers a real practical issue faced by a real industrial plant involving severe hydrocracking for producing three modes of base oil 4, 6, and 10 cSt. In this paper, a proactive method with the aim to minimize the production loss due to lab sampling delay has been studied and developed using plant historical data to classify hourly the correct mode for the base oil. As the current practice in this plant relies on the laboratory analysis results which is normally only available at the eighth hours from the start of the transition, large amounts of useful products have to be diverted unnecessarily into the slopping tank. In this study, an automated mode detection tool using XGBOOST has been developed and shown to be able to satisfactorily provide early detection of the mode change as well as the transition modes by utilizing the base oil correlative pattern unraveled by the third principal component calculated by PCA together with the plant hourly operating data. This is a

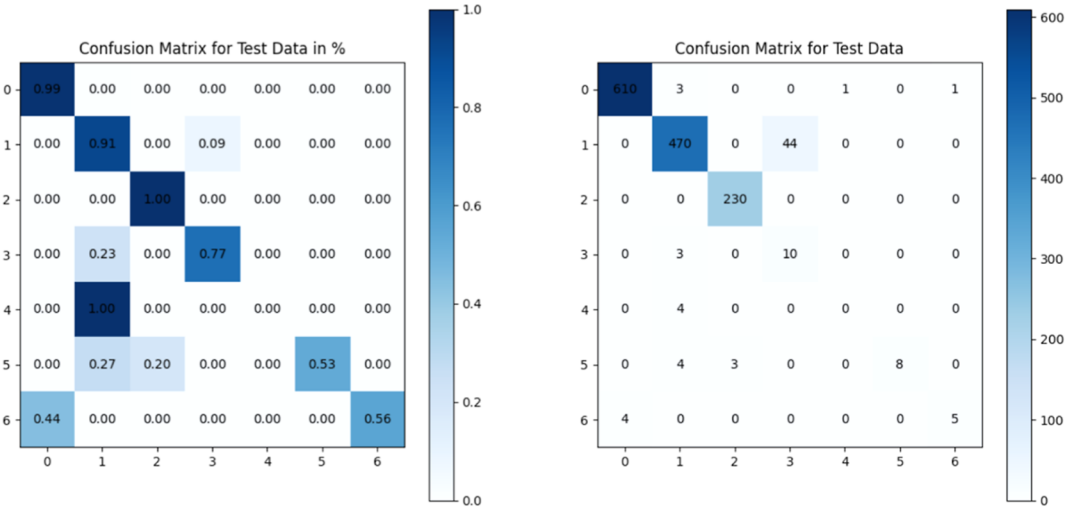
https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540



(a) XGBOOST model



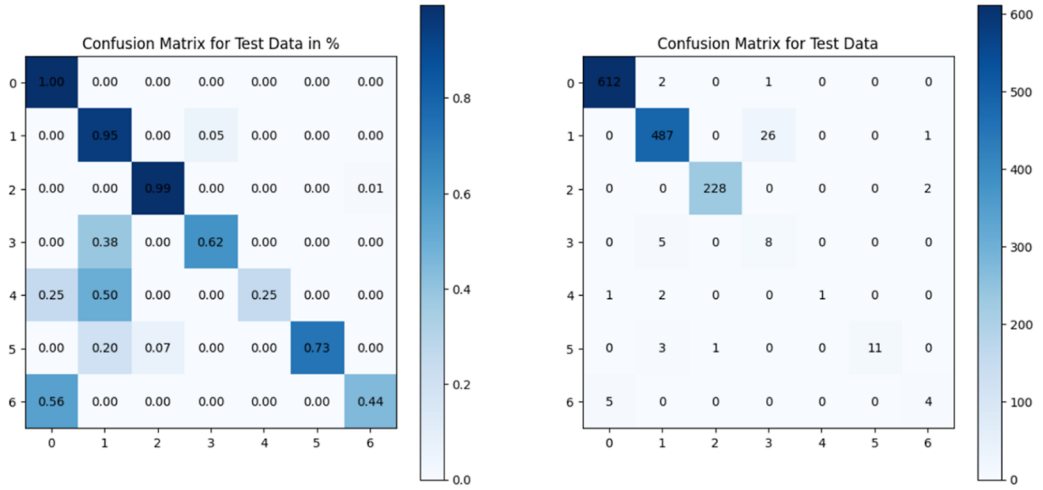
(b) Random Forest model



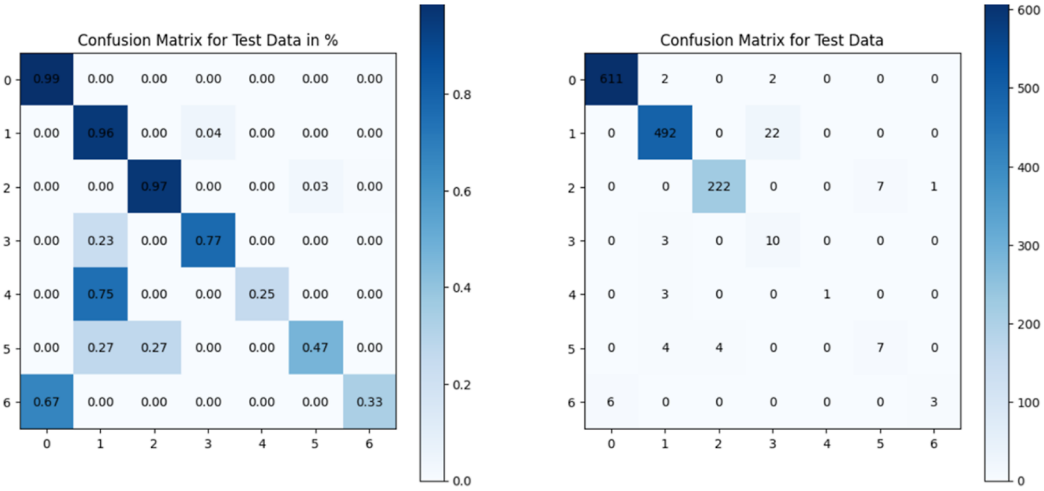
(c) CatBoost model

Figure 13. Classification performance comparison for benchmark data.

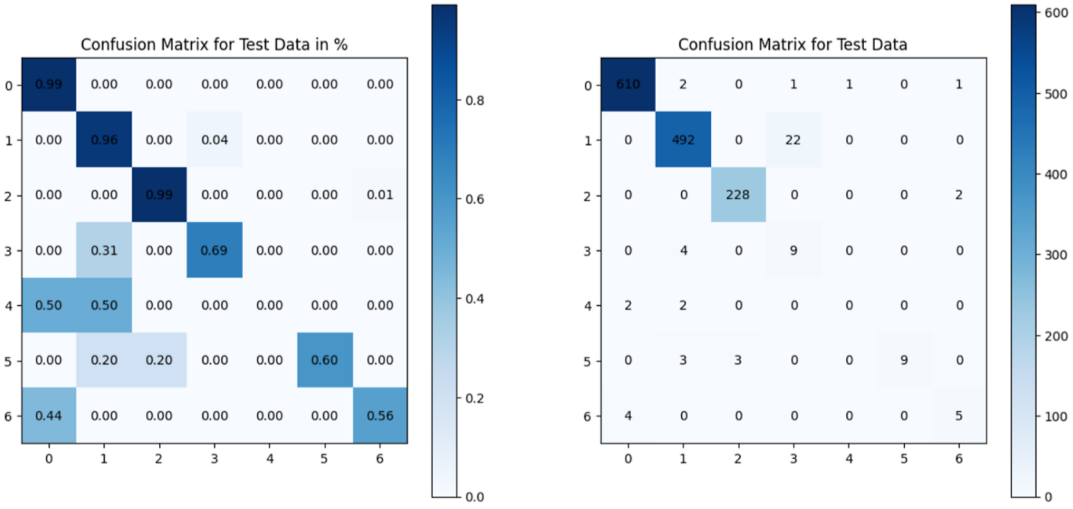
https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540



(a) XGBoost model



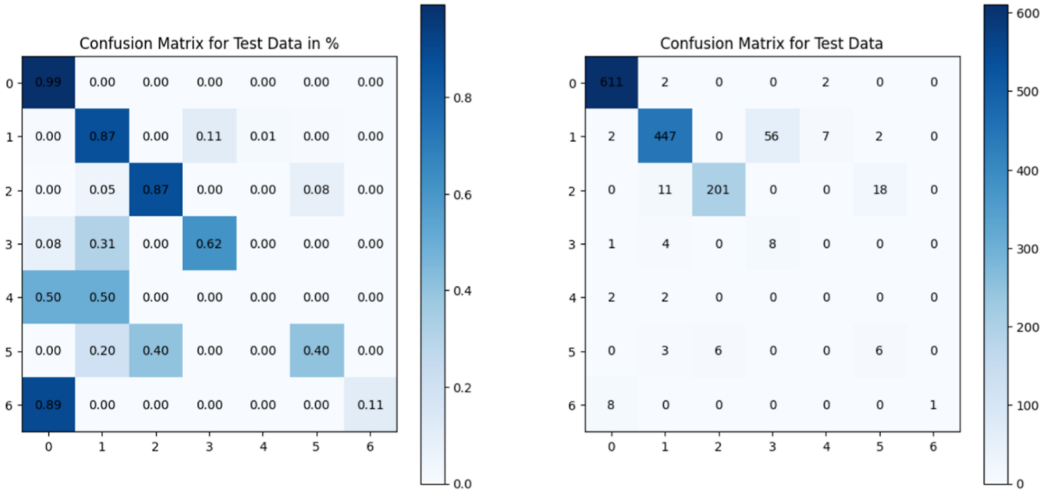
(b) Random Forest model



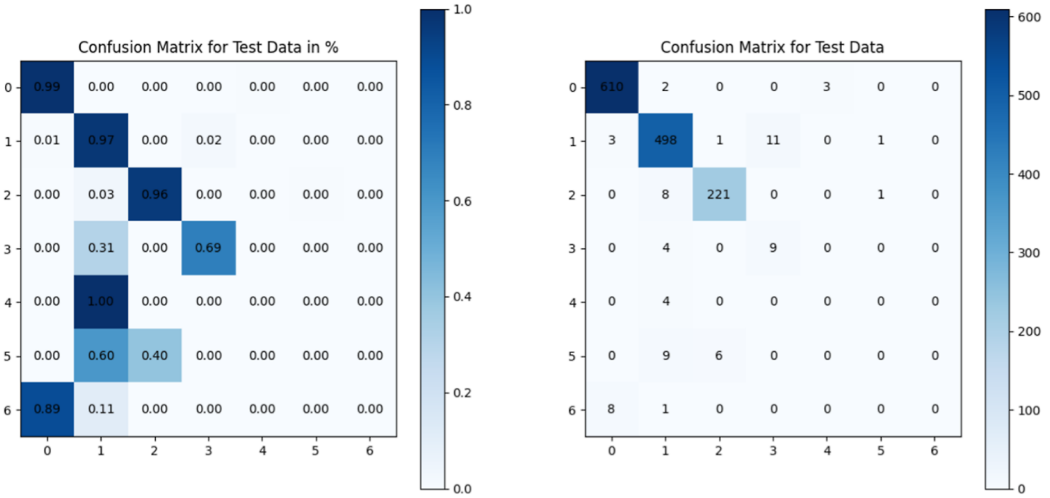
(c) CatBoost model

Figure 14. Classification performance for Run 1 data.

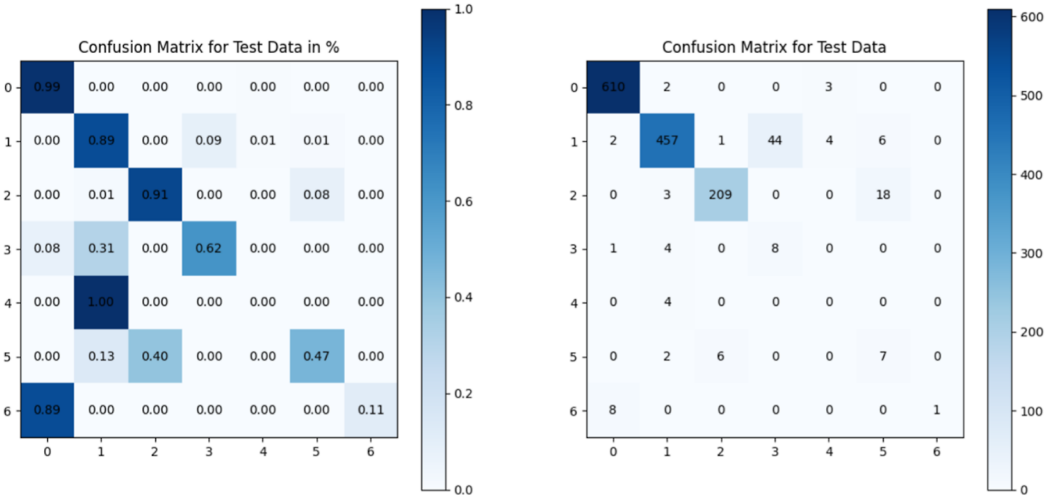
https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540



(a) XGBoost model



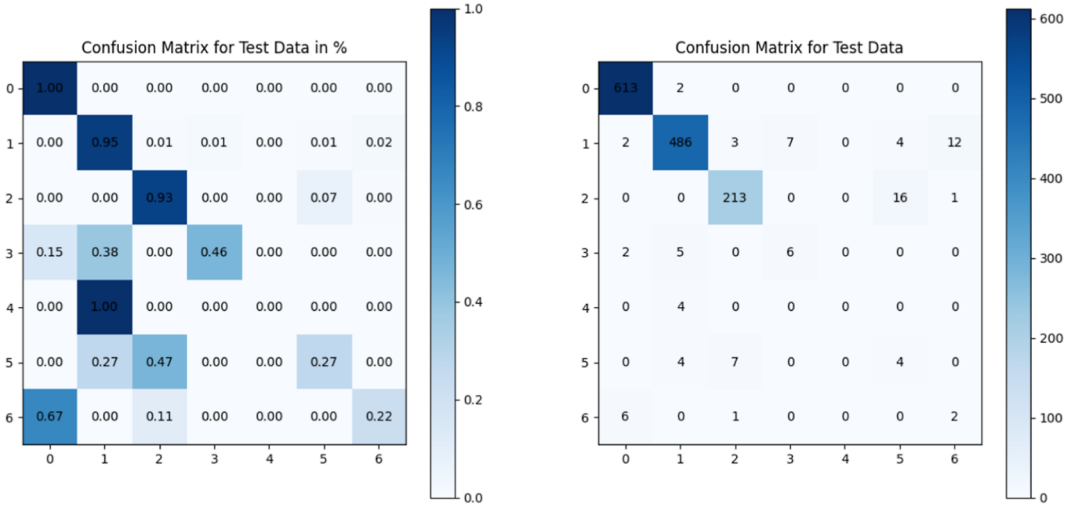
(b) Random Forest model



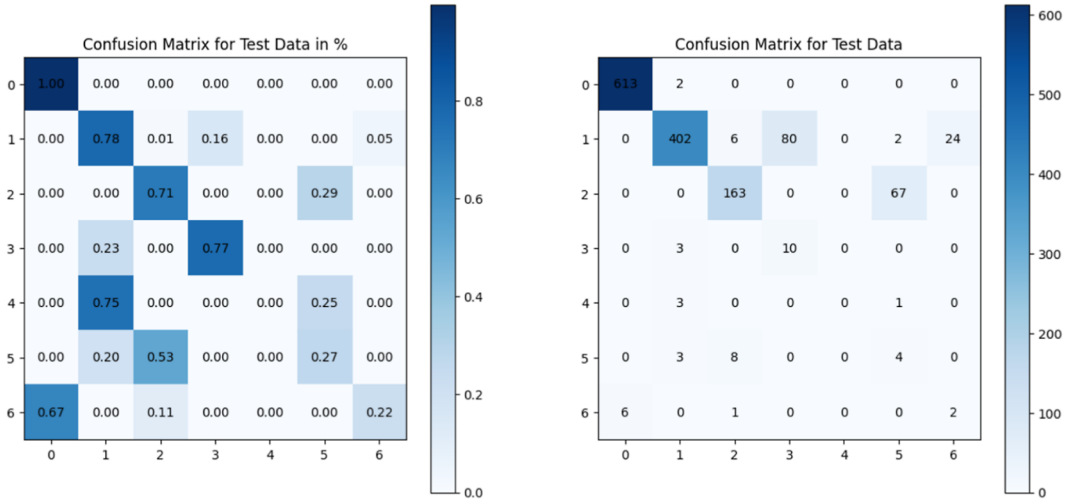
(c) CatBoost model

Figure 15. Classification performance for Run 2 data.

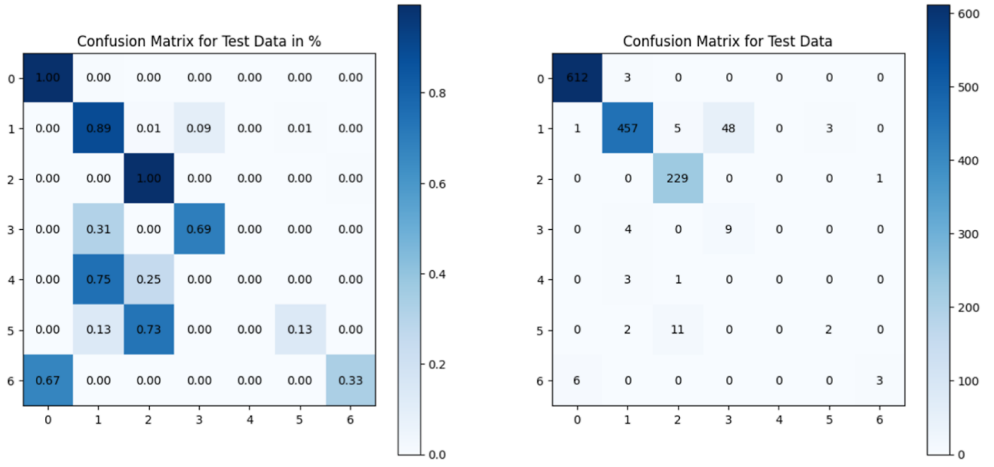
https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540



(a) XGBoost model



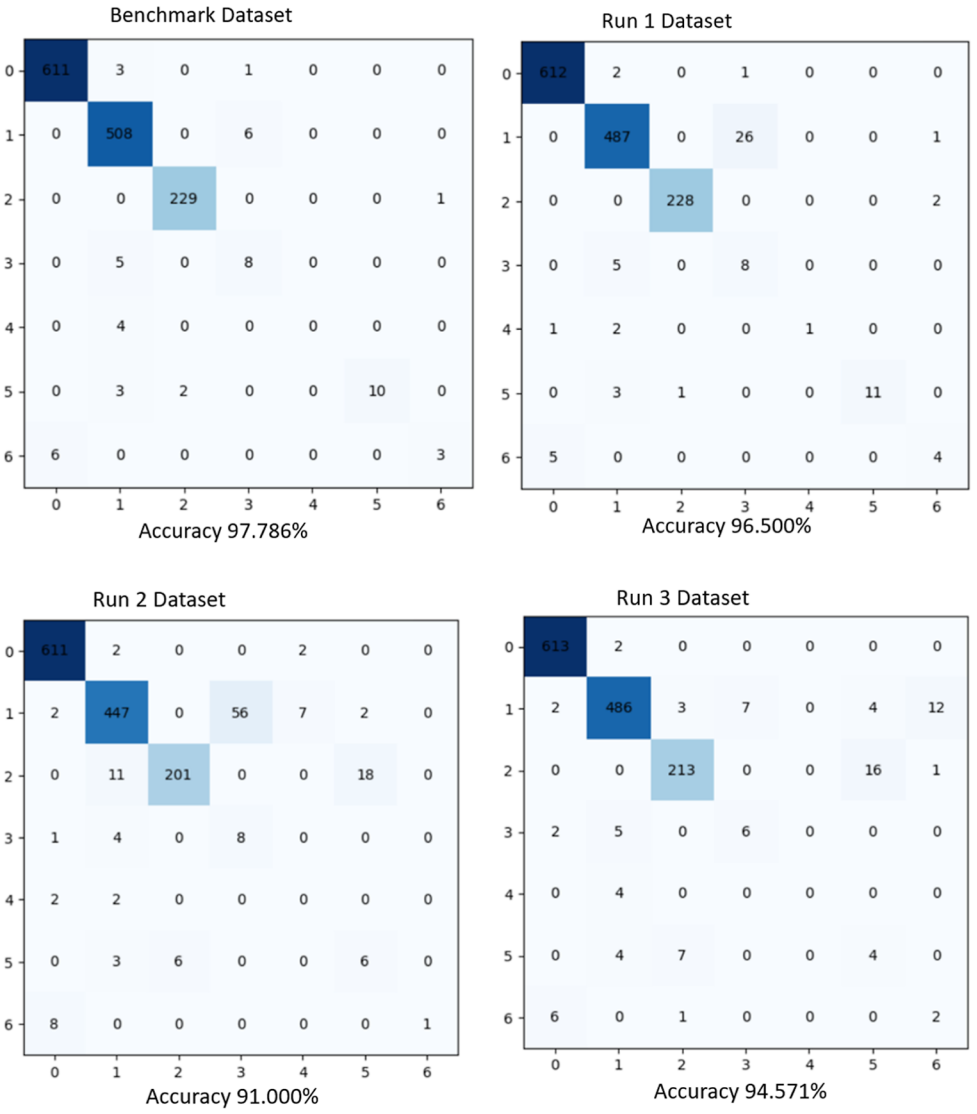
(b) Random Forest model



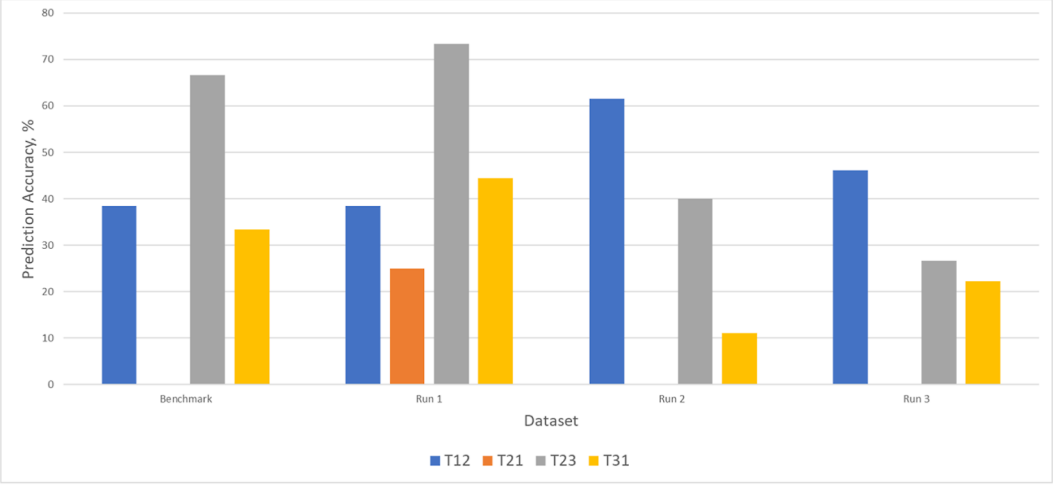
(c) CatBoost model

Figure 16. Classification performance for Run 3 data.

https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540



(a) Confusion matrix for XGBoost performance for all dataset



(b) Average percentage performance

Figure 17. XGBoost performance in predicting the transitions with different deployment data sets.

https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540

Table 6. Comparison of All Models Performance Analysis for All Runs

data set	XGBoost accuracy (%)	Random Forest accuracy (%)	CatBoost accuracy (%)
benchmark	97.786	96.357	95.214
Run 1	96.500	96.071	96.643
Run 2	91.000	95.214	92.286
Run 3	94.571	86.357	93.714

promising alternative to minimize the reliance on the lab sampling and minimize the production losses.

AUTHOR INFORMATION

Corresponding Author

Haslinda Zabiri − *Department of Chemical Engineering, Universiti Teknologi PETRONAS, Bandar Seri Iskandar 32610 Perak, Malaysia;* orcid.org/0000-0003-1821-1028; Email: haslindazabiri@utp.edu.my

Authors

Muhamad Amir Mohd fadzil − *Group Research & Technology, PETRONAS, Kawasan Institusi Bangi, Kajang 43000 Selangor, Malaysia*

Adi Aizat Razali − *Group Research & Technology, PETRONAS, Kawasan Institusi Bangi, Kajang 43000 Selangor, Malaysia*

Amar Haiqal Che Hussin − *Department of Chemical Engineering, Universiti Teknologi PETRONAS, Bandar Seri Iskandar 32610 Perak, Malaysia*

Complete contact information is available at: https://pubs.acs.org/10.1021/acsomega.3c07331

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

The authors would like to thank Universiti Teknologi PETRONAS and MRA grant number 015MD0-051 for the technical support and funding provided.

REFERENCES

(1) Mehrkesh, A. H.; Hajimirzaee, S.; Hatamipour, M. S. A Generalized Correlation for Characterization of Lubricating Base-oils from Their Viscosities. *Chin. J. Chem. Eng.* **2010**, *18* (4), 642−647.

(2) Prasannakumar, P.; Edla, S.; Thampi, A. D.; Arif, M.; Santhakumari, R. A comparative study on the lubricant properties of chemically modified Calophyllum inophyllum oils for bio-lubricant applications. *J. Cleaner Prod.* **2022**, *339*, 130733.

(3) Baek, J. W.; Ko, J. H.; Park, J. H.; Park, J. Y.; Lee, H. J.; Seo, Y. H.; Lee, J.; Lee, B. Y. α-Olefin Trimerization for Lubricant Base Oils with Modified Chevron−Phillips Ethylene Trimerization Catalysts. *Organometallics* **2022**, *41* (17), 2455−2465.

(4) Lu, T.; Jan, K.; Chen, W.-T. Hydrothermal liquefaction of pretreated polyethylene-based ocean-bound plastic waste in supercritical water. *J. Energy Inst.* **2022**, *105*, 282−292.

(5) Velasco-Calderón, J. C.; Garcia-Figueroa, A. A.; Lopez Cervantes, J. L.; Gracia-Fadrique, J. Regeneration of used lubricating oil by solvent extraction and phase diagram analysis. *Curr. Res. Green Sustainable Chem.* **2020**, *3*, 100010.

(6) 2. Lubricant technology a survey. In *Tribology Series*; Sethuramiah, A., Ed.; Elsevier, 2003; Vol. *42*, pp 35−65.

(7) Sattler, A.; Ho, S.; Paccagnini, M.; Padilla, C. Nickel catalyzed oligomerization of internal olefins: Towards lubricant basestocks and high cetane diesel fuels. *Polyhedron* **2020**, *182*, 114469.

(8) Lau, P.-C.; Kwong, T.-L.; Yung, K.-F. Manganese glycerolate catalyzed simultaneous esterification and transesterification: The kinetic and mechanistic study, and application in biodiesel and bio-lubricants synthesis. *Renewable Energy* **2022**, *189*, 549−558.

(9) Kherif, F.; Latypova, A. Principal component analysis. In *Machine Learning*; Elsevier, 2020; pp 209−225.

(10) Fadzil, M. A. M.; Zabiri, H.; Razali, A. A.; Basar, J.; Syamzari Rafeen, M. Base Oil Process Modelling Using Machine Learning. *Energies* **2021**, *14* (20), 6527.

(11) Mitchell, T. M.; Mitchell, T. M. *Machine Learning (No. 9)*; McGraw-Hill New York, 1997.

(12) L., Ciortuz. Machine Learning Lecture Slides for Postgraduate Level. Available: https://profs.info.uaic.ro/~ciortuz/SLIDES/ml0.pdf (accessed December 1, 2021).

(13) Lin, G.; Lin, A.; Gu, D. Using support vector regression and K-nearest neighbors for short-term traffic flow prediction based on maximal information coefficient. *Inf. Sci.* **2022**, *608*, 517−531.

(14) Arabameri, A.; Chandra Pal, S.; Rezaie, F.; Chakrabortty, R.; Saha, A.; Blaschke, T.; Di Napoli, M.; Ghorbanzadeh, O.; Thi Ngo, P. T. Decision tree based ensemble machine learning approaches for landslide susceptibility mapping. *Geocarto Int.* **2022**, *37* (16), 4594−4627.

(15) Pan, S.; Zheng, Z.; Guo, Z.; Luo, H. An optimized XGBoost method for predicting reservoir porosity using petrophysical logs. *J. Pet. Sci. Eng.* **2022**, *208*, 109520.

(16) Abdullah, D.; Susilo, S.; Ahmar, A. S.; Rusli, R.; Hidayat, R. The application of K-means clustering for province clustering in Indonesia of the risk of the COVID-19 pandemic based on COVID-19 data. *Qual. Quantity* **2022**, *56* (3), 1283−1291.

(17) Zhang, Y.; Wang, Y. Forecasting crude oil futures market returns: A principal component analysis combination approach. *Int. J. Forecast.* **2023**, *39*, 659.

(18) Xu, N.; Xu, C.; Finkelman, R. B.; Engle, M. A.; Li, Q.; Peng, M.; He, L.; Huang, B.; Yang, Y. Coal elemental (compositional) data analysis with hierarchical clustering algorithms. *Int. J. Coal Geol.* **2022**, *249*, 103892.

(19) Song, J.; Jin, L.; Xie, Y.; Wei, C. Optimized XGBoost based sparrow search algorithm for short-term load forecasting. *2021 IEEE International Conference on Computer Science, Artificial Intelligence and Electronic Engineering (CSAIEE)*; Institute of Electrical and Electronics Engineers, 2021; pp 213−217.

(20) Zou, M.; Jiang, W.-G.; Qin, Q.-H.; Liu, Y.-C.; Li, M.-L. Optimized XGBoost model with small dataset for predicting relative density of Ti-6Al-4V parts manufactured by selective laser melting. *Materials* **2022**, *15* (15), 5298.

(21) Wu, Z.; Wang, X.; Jiang, B. Fault diagnosis for wind turbines based on ReliefF and eXtreme gradient boosting. *Appl. Sci.* **2020**, *10* (9), 3258.

(22) Gawali, S.; Agale, P.; Ghorpade, S.; Gawade, R.; Nimat, P. Intrusion detection using hidden Markov model and XGBoost algorithm. *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.* **2020**, *6*, 466−470.

(23) Takalo-Mattila, J.; Heiskanen, M.; Kyllonen, V.; Maatta, L.; Bogdanoff, A. Explainable Steel Quality Prediction System Based on Gradient Boosting Decision Trees. *IEEE Access* **2022**, *10*, 68099−68110.

(24) Dhali, S.; Pati, M.; Ghosh, S.; Banerjee, C. An efficient predictive analysis model of customer purchase behavior using random forest and XGBoost algorithm. *2020 IEEE 1st International Conference for Convergence in Engineering (ICCE)*; Institute of Electrical and Electronics Engineers, 2020; pp 416−421.

(25) Khan, M. Y.; Qayoom, A.; Nizami, M. S.; Siddiqui, M. S.; Wasi, S.; Raazi, S. M. K.-u.-R. Automated prediction of Good Dictionary EXamples (GDEX): a comprehensive experiment with distant supervision, machine learning, and word embedding-based deep learning techniques. *Complexity* **2021**, *2021*, 1−18.

(26) Alfarizi, M. G.; Tajiani, B.; Vatn, J.; Yin, S. Optimized random forest model for remaining useful life prediction of experimental bearings. *IEEE Trans. Ind. Inf.* **2023**, *19*, 7771−7779.

(27) Gan, H.; Jiao, B. Fault diagnosis of wind Turbine's gearbox based on improved GA random forest classifier. *DEStech Trans. Eng. Technol. Res.* **2018**, 206−210.
(28) Zhong, C.; Geng, F.; Zhang, X.; Zhang, Z.; Wu, Z.; Jiang, Y. Shear Wave Velocity Prediction of Carbonate Reservoirs Based on CatBoost. *2021 4th International Conference on Artificial Intelligence and Big Data (ICAIBD)*; Institute of Electrical and Electronics Engineers, 2021; pp 622−626.
(29) Yuan, Z.; Zhou, T.; Liu, J.; Zhang, C.; Liu, Y. Fault diagnosis approach for rotating machinery based on feature importance ranking and selection. *Shock Vib.* **2021**, *2021*, 1−17.
(30) Pan, Y.; Zhang, L. Data-driven estimation of building energy consumption with multi-source heterogeneous data. *Appl. Energy* **2020**, *268*, 114965.
(31) Colan, S. D. The Why and How of Z Scores. *J. Am. Soc. Echocardiogr.* **2013**, *26* (1), 38−40.
(32) Hotelling, H. Analysis of a complex of statistical variables into principal components. *J. Educ. Psychol.* **1933**, *24* (6), 417−441.
(33) Zabiri, H.; Ramasamy, M. NLPCA as a diagnostic tool for control valve stiction. *J. Process Control* **2009**, *19* (8), 1368−1376.
(34) SIMCA Free Trial. Available: https://www.sartorius.com/en/products/process-analytical-technology/data-analytics-software/mvda-software/simca/simca-free-trial-download (Accessed date: October 2021).
(35) Coenen, F. *On the use of confusion matrixes*; University of Liverpool, 2012.
(36) Wu, J.; Li, Y.; Ma, Y. Comparison of XGBoost and the neural network model on the class-balanced datasets. *2021 IEEE 3rd International Conference on Frontiers Technology of Information and Computer (ICFTIC)*; Institute of Electrical and Electronics Engineers, 2021; pp 457−461.
(37) Prokhorenkova, L.; Gusev, G.; Vorobev, A.; Dorogush, A. V.; Gulin, A. CatBoost: unbiased boosting with categorical features. *Advances in Neural Information Processing Systems*, 2018; Vol. *31*.
(38) Hancock, J.; Khoshgoftaar, T. M. Performance of CatBoost and XGBoost in Medicare Fraud Detection. *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2020; pp 572−579.
(39) Yin, L.; Li, B.; Li, P.; Zhang, R. Research on stock trend prediction method based on optimized random forest. *CAAI Trans. Intell. Technol.* **2023**, *8* (1), 274−284.

https://doi.org/10.1021/acsomega.3c07331
ACS Omega 2024, 9, 3525−3540