Can machine learning algorithms effectively classify tank farms based on risk levels associated with their operational characteristics?

H. Leung, SN: 23218406, COMP0047

ABSTRACT

This study evaluates the potential of machine learning (ML) algorithms to classify tank farms based on operational risk levels. Leveraging a comprehensive dataset of Louisiana tank farms, the research entailed rigorous data preprocessing, exploratory analysis, and the application of various ML models. Despite models like Support Vector Classifier (SVC) and k-Nearest Neighbors (KNN) initially outperforming a baseline heuristic on validation data, a performance dip on an unseen test set highlighted the challenges of overfitting and model generalization. The findings affirm the viability of ML in enhancing industrial risk assessment but underscore the need for better data collection/augmentation, ongoing model refinement, and robust data strategies for real-world applicability.

I. INTRODUCTION

The management and operation of tank farms play a crucial role in the industrial sector, particularly within the domains of petroleum storage and distribution. Ensuring the safety and efficiency of these facilities not only bears economic significance but also holds implications for environmental protection, regulatory compliance, and general public safety. Amidst the myriad of operational characteristics, the ability to classify tank farms based on associated risk levels is paramount. This classification task has the potential to drive proactive measures, inform safety protocols, and optimize resource allocation.

However, the complexity and variability inherent in tank farm operations pose significant challenges in risk assessment. Traditional methods that rely on manual inspection and historical trend analysis may not suffice to capture the dynamic nature of risk factors. In response to this challenge, this study seeks to explore the efficacy of ML algorithms in discerning patterns within operational data that correlate with risk levels. Our research is pivoted around the question: Can machine learning algorithms effectively classify tank farms based on risk levels associated with their operational characteristics? By leveraging historical and operational data, this report investigates the potential of various ML algorithms to provide a systematic and data-driven approach to risk classification.

The report will follow a structured approach, beginning with the collection, preparation, and preliminary analysis of a rich dataset, followed by rigorous feature selection and ML model development. The models' performances are evaluated and validated through statistical measures, ensuring robustness and reliability in their predictive capabilities. The subsequent sections will detail the methodology adopted for data preprocessing, feature engineering, model selection, hypothesis testing, and validation, all of which culminate in a comprehensive performance evaluation. This report aims to contribute to the integration of ML in industrial risk management within the oil storage industry and to provide actionable insights for stakeholders in the tank farm industry.

II. METHODOLOGY & DATA

The comprehensive dataset utilized in this study encompasses a range of characteristics pertaining to tank farms located in Louisiana, USA. This dataset includes various features that are hypothesized to influence the risk level associated with tank storage levels, it encapsulates both static and dynamic attributes with a time frame of 4 years from 2014 to 2018, the original dataset contains 44685 rows and 14 columns.

While half the dataset contains identity data of the tanks such as Location, farm type, tank farm, tank id, etc. The main interesting part of the dataset contains information on the storage capabilities of the tanks themselves, with information such as the maximum volume of the tank, current tank volume, and fill percentage, which is the current tank volume divided by the maximum volume. The tank volume data was gathered by satellite images, made possible by the particular tank type that has a floating head that sits directly on top of the oil. As a result, the height of the tank head rises and falls with the volume of oil in the tank, the relative sizes of the exterior shadow cast by the tank itself and the interior shadow cast by the height of the tank head can be used to estimate the tank volume.

A. Data Augmentation

Central to our investigation was the derivation of a dependent target variable suitable for binary classification tasks. In the context of tank farm risk assessment, along with the data that was provided to us, opinions of industry experts[1] and insights extracted from the API Standard 2350 issued by the American Petroleum Institute, it was determined to categorize into two categories: risky or non-risky, predicated on their fill percentage. Tanks with a fill percentage exceeding 0.91 or falling below 0.09 were deemed risky. We acknowledge that this may lead to an exceptional performance of models in generalizing the correlation between fill percentage and risk level. Nonetheless, this approach allows us to scrutinize potential relationships between risk levels and other variables.

The reason these values were chosen was due to the risks associated with overfilling or underfilling the storage tanks. Filling tanks beyond their designed capacity can lead to mechanical stress and even structural failure. Additionally, overfilling reduces the 'ullage' space at the top of the tank which allows for the expansion of the oil, especially under higher temperatures during the summer, potentially leading to over-pressured situations[2]. As for underfilling, tanks that are consistently kept at low fill levels can be more susceptible to internal corrosion due to increased exposure of the tank walls to moisture and air. Also, in certain conditions, partially filled tanks can be more vulnerable to 'sloshing' effects during earthquakes or other movements, which can damage the tank structure and foundations. A reasonable concern as Louisiana has had 9 earthquakes in the last 365 days, while relatively small earthquakes at a magnitude of 3.0 and below, consequent earthquakes could gradually weaken the structure of the tanks. Furthermore, considering most of the tanks are situated near bodies of water as well as near civilisation, any spillage of these tanks could create big issues for the local population.

Larger tanks inherently carry more risk simply due to the larger volume of oil they contain. A breach or failure in a large tank can result in a more significant spill compared to smaller tanks. Larger volumes of oil also present more fuel for potential fires or explosions.

B. Data Preprocessing

Prior to the integration into our analysis pipeline, the dataset took crucial steps to ensure data quality and integrity. This process included unpacking values, normalization of feature scales, handling of missing values, and the encoding of categorical variables. Each step in the data preparation phase was undertaken with a dual focus on preserving the fidelity of the original information and optimizing the dataset for ML algorithms.

1. Handling Missing Values

Given the nature of historical and operational datasets, missing values are an expected occurrence. However, in our dataset, most of the data was intact and not null, except for the ”tank farm” column, where around 2000 values were undefined. Where the typical approach for handling these gaps would either be deletion or data imputation, for this specific case we were able to plot all the data points onto a map, and after cross-referencing with Google Earth we would allocate the specific tank in question to a tank farm they were a part of. However, some tanks that were by themselves and not geographically close to any tank farms were removed completely, leading to 2621 rows being dropped. Click here for an interactive map of the tanks.

2. Data Reduction

Upon further inspection of the dataset, there seemed to be duplicate rows with relation to ”tank id” and ”imaging time”, the same tank at the same time had two different entries of data, giving different ”fill pct” and ”tank volume” data. After further analysis and discussion, we decided to drop the duplicates and keep the first instance of the discrepancy. We also dropped a duplicate ”tank id” column as well as more unused columns that we believed would not provide any benefit to the training of our models. Specifically, the columns relating to the imaging metadata, ”scene id”, ”provider scene id”, ”scene source id”, two more columns ”farm type” and ”region” were also dropped as they contained the same one or two value throughout the whole dataset that was irrelevant to our study. Finally, the ”imaging time” column was also dropped as we believed having the year, month, and day was enough.

3. Categorical Variable Encoding

As the ”tank id” column values were in a string format that our ML algorithms could not process in their raw form, we converted these values into a numerical format using scikit-learn’s `LabelEncoder` [3].

4. Feature Engineering

We engineered new features to allow the ML models to comprehend the dataset more effectively and capture additional insights from the data. We unpacked two columns into additional columns. The ”Location” column was decomposed into ”Longitude” and ”Latitude”, while the ”image time round day” column was split into ”year”, ”month”, and ”day”, this was hypothesized to influence the risk level due to varying operational activity throughout the year.

5. Feature Scaling

An essential step in our preprocessing pipeline was feature scaling, which is crucial for models such as Support Vector Machines and k-Nearest Neighbors that are sensitive to the magnitude of variables as they rely on distance calculations. We employed `RobustScaler` from the scikit-learn library, a method particularly suited for datasets with outliers. Unlike standard scaling methods that are influenced by the mean and variance of the data, `RobustScaler` uses the median and interquartile range, thus ensuring that the scaling is resistant to outlier distortions. By transforming the features to a similar scale, `RobustScaler` enabled a more balanced and nuanced comparison across variables

6. Handling Data Imbalance

When investigating the dataset after implementing the new ”risk” column, we found that the class distribution of the risk labels was severely skewed, with an over-representation of the ”not risky” category. To prevent our models from inheriting a bias towards the majority class, we applied the Synthetic Minority Over-sampling Technique (SMOTE)[4] to the training data. SMOTE works by synthesizing new instances of the minority class, based on the feature space similarities of existing minority instances, thus creating a more balanced class distribution. By interpolating between neighboring samples, SMOTE introduces diversity to the dataset, allowing our models to learn more generalized patterns rather than simply replicating existing observations. The use of SMOTE was restricted to the training set to prevent the introduction of artificial bias into the model evaluation process.

It is important to note that while feature scaling and SMOTE oversampling are mentioned in the data preprocessing section within this report, they were applied after data exploration, right before model building, and thus do not contribute to the findings in the next section.

C. Data Exploration

Before delving into model-building, we conducted an exploratory data analysis (EDA) to uncover underlying patterns, spot anomalies, and form hypotheses about the tank farms' risk levels. This initial foray into the dataset was critical in shaping our understanding and guiding subsequent analytical strategies.

In this stage, we first examined the distribution of various features. Continuous variables such as tank volumes and fill percentages were visualized using histograms and box plots, revealing the central tendency and the type of distribution they follow. Categorical variables, including tank farms and months, were summarized through bar charts to display their frequency within the dataset.

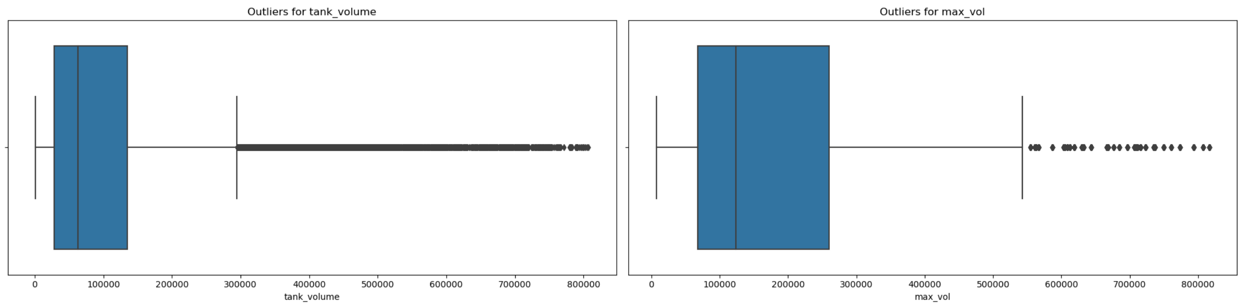


FIG. 1. Box plots for tank volume and max vol.

In Figure 1, the side-by-side box plots reveal a right-skewed distribution in both variables, with tank volume displaying a tighter concentration of values as indicated by a smaller IQR and a longer tail of sparse, extreme outliers. In contrast, max vol shows a broader IQR, suggesting greater variability within its middle 50% of values and fewer, less extreme outliers. These distributions suggest that while both variables have non-standard patterns, `tank_volume` has more pronounced deviations from the median, which could impact statistical models differently.

	Mean	Standard Deviation	Skewness	Kurtosis
fill pct	0.55	0.24	0.067	-1.13

TABLE I. Descriptive Statistics for Fill Percentage

Table I shows the calculated descriptive statistics for the fill percentage variable. The average fill percent is 0.55, indicating the central tendency of the tank fill percentage throughout the dataset, while a standard deviation of 0.24 suggests a moderate spread of fill percentages from the average. Skewness measures the asymmetry of the distribution of values around the mean, a skewness close to 0 suggests that the distribution of fill percent is fairly symmetrical and not significantly skewed to the left or right. Finally, a kurtosis value less than 0 suggests a distribution with lighter tails and a flatter peak compared to a normal distribution, implying fewer outliers in the data than expected in a normal distribution. These statistics suggest that the fill percent values are relatively normally distributed, with moderate variation and a distribution that is fairly symmetrical and somewhat flatter than a normal curve. This can be confirmed by looking at the relevant histogram in Figure 2, where it looks somewhat similar to a flatter normal distribution but not exactly due to the gaps in between bins.

The histogram for max volume and tank volume further confirms the analysis we made for their respective box plots in Figure 1, where the data is heavily skewed to the right, with a majority of data points in the first quadrant of the dataset, we can also theorise a log distribution judging from the shape of the distribution. Finally, the histogram for months presents an interesting distribution, with the most images taken during the winter season from November to January, and fewer records during the summer time. Looking at the shape it suggests a U-shaped bimodal distribution, indicating that there is a higher chance of a measurement being found at the tails than in the center of the distribution.

As mentioned earlier, although the fill percentage histogram is a bit flat, it still somewhat resembles a normal distribution. The fitter library [5] tests 111 distributions against the histogram and outputs the top 5 best-fitting probability density functions (PDF) that model

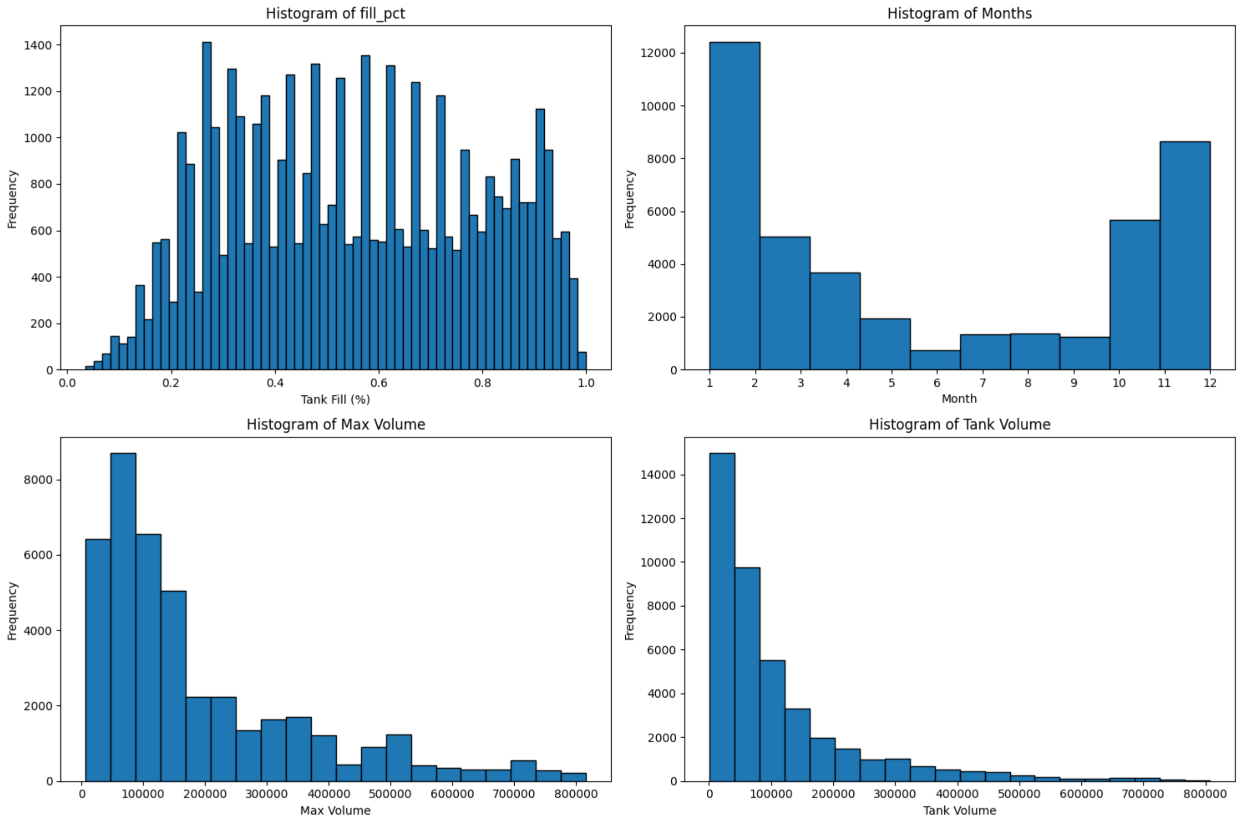


FIG. 2. Histograms of fill percentage, months, max volume, and tank volume.

the data. For the fill percent distribution, the best-fitting PDF was the generalized normal continuous random variable (gennorm). While for the tank volume histogram, it was the log normal continuous random variable (lognorm). Subsequently, we conducted a correlation analysis to discern the strength and direction of associations between variables. A heatmap of correlation coefficients offered a visual summary of these relationships, highlighting potential predictors of risk and informing our feature selection process. The Correlation Matrix heatmap provided in Figure 4 displays a strong positive relationship between max vol and tank volume, which suggests that as the maximum volume of the tanks increases, the actual volume tends to increase as well in a monotonic fashion. When looking at the row for ”risk”, it’s surprising to see that fill pct and risk only have a correlation score of 0.39, considering the risk rating was derived from fill pct. Spearman’s Coefficient was chosen over Pearson’s Coefficient as it does not assume a normal distribution of the data, an can be used with ordinal data, it’s also less sensitive to outliers and non-linear relationships which makes it applicable to our dataset.

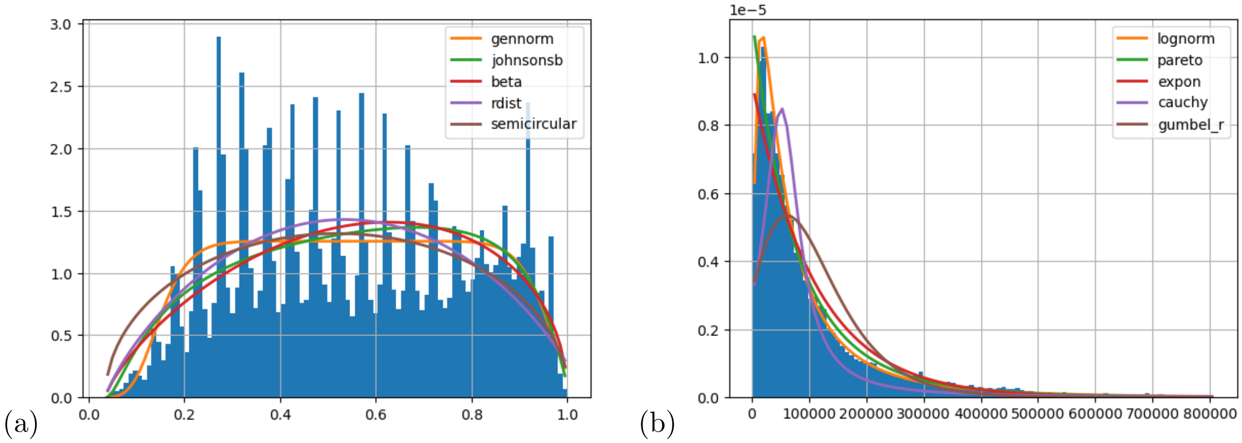
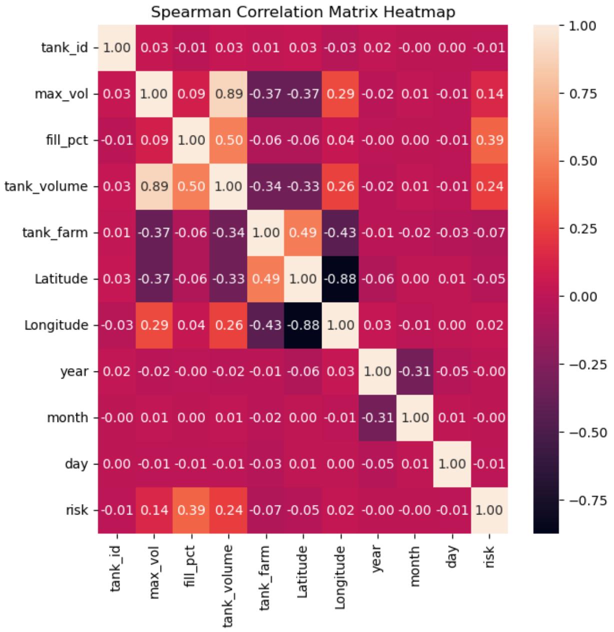


FIG. 3. Histograms of fill pct (left) and tank volume (right), along with distributions generated by the fitter library [5]



	tank_id	max_vol	fill_pct	tank_volume	tank_farm	Latitude	Longitude	year	month	day	risk
tank_id	1.00	0.03	-0.01	0.03	0.01	0.03	-0.03	0.02	-0.00	0.00	-0.01
max_vol	0.03	1.00	0.09	0.89	-0.37	-0.37	0.29	-0.02	0.01	-0.01	0.14
fill_pct	-0.01	0.09	1.00	0.50	-0.06	-0.06	0.04	-0.00	0.00	-0.01	0.39
tank_volume	0.03	0.89	0.50	1.00	-0.34	-0.33	0.26	-0.02	0.01	-0.01	0.24
tank_farm	0.01	-0.37	-0.06	-0.34	1.00	0.49	-0.43	-0.01	-0.02	-0.03	-0.07
Latitude	0.03	-0.37	-0.06	-0.33	0.49	1.00	-0.88	-0.06	0.00	0.01	-0.05
Longitude	-0.03	0.29	0.04	0.26	-0.43	-0.88	1.00	0.03	-0.01	0.00	0.02
year	0.02	-0.02	-0.00	-0.02	-0.01	-0.06	0.03	1.00	-0.31	-0.05	-0.00
month	-0.00	0.01	0.00	0.01	-0.02	0.00	-0.01	-0.31	1.00	0.01	-0.00
day	0.00	-0.01	-0.01	-0.01	-0.03	0.01	0.00	-0.05	0.01	1.00	-0.01
risk	-0.01	0.14	0.39	0.24	-0.07	-0.05	0.02	-0.00	-0.00	-0.01	1.00

FIG. 4. Spearman

D. Model Building

After a thorough preprocessing and exploratory data analysis phase, we progressed to the core of our study—the model building. This section outlines the methodologies employed to construct ML models capable of classifying tank farms based on their assessed risk levels.

1. Selection of Algorithms

Given the classification nature of our task, we selected a diverse array of ML algorithms to evaluate their performance and suitability for our specific context. The algorithms included:

- Logistic Regression
- Support Vector Machines (SVM)
- Random Forrest
- Decision Tree
- K-Nearest Neighbors
- Extreme Gradient Boosting (XGBoost)

2. Model Training

The dataset was first put through an 80/20 Train Test set split to ensure we have unseen data we can evaluate our final model. Each model was trained using the balanced dataset prepared in the preprocessing stage.

Cross-validation was utilized extensively to validate the models within the training phase. This not only provided insights into the models' stability and performance across different subsets of the data but also helped in safeguarding against overfitting. Confusion Matrices were created after cross-validation to understand how well our models were generalizing on the train data before we picked two models to move on to the final stage.

3. Baseline Model

To establish a benchmark for assessing the performance of our ML models, we implemented a baseline model using the DummyClassifier from [3] with the 'most-frequent' strategy. This simplistic approach involves predicting the most common class label in the training set for all observations, regardless of the input features. The rationale behind choosing this baseline model lies in its utility as a reference point: it sets a minimum performance threshold that any sophisticated model should surpass to be considered effective.

E. Hypothesis Testing

To assess the effectiveness and significance of our predictive models, we incorporated hypothesis testing into our methodology. This process is crucial for validating the assumptions made during model building and for confirming the statistical significance of the findings. We ended up picking SVC and KNN for the model comparison, as these models are not based on the estimation of coefficients that have known distributions, traditional statistical hypothesis tests (like t-tests or ANOVA) for coefficient significance are not applicable.

To compare the efficacy of our advanced models against the baseline model, we employed the Wilcoxon signed-rank test. This non-parametric test evaluates whether two related samples (the prediction accuracies of two models across multiple cross-validation folds) come from the same distribution. A significant result (p-value ¡ 0.05) indicates that one model consistently outperforms the other across the datasets.

III. RESULTS & DISCUSSION

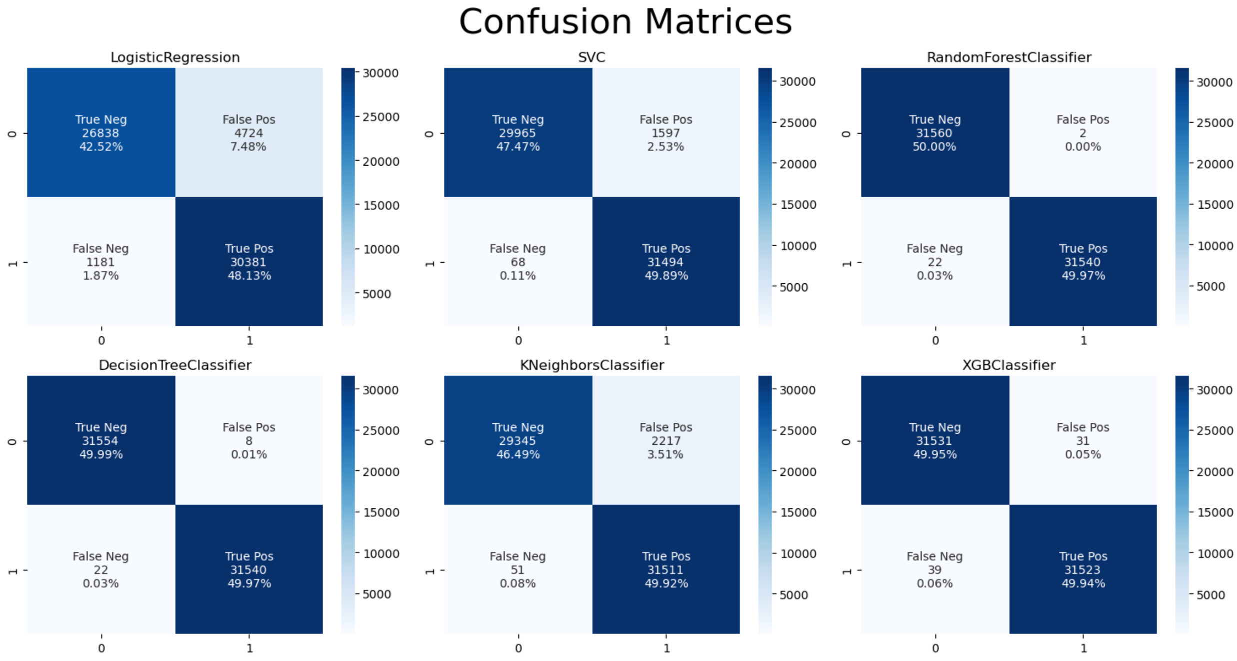


FIG. 5. Confusion Matrices for the 6 algorithms put into a 10-fold Cross-Validation

For the Confusion Matrices in Figure 5, ideally we want to minimize both False Positives (FP) and False Negatives (FN) while maximizing True Positives (TP) and True Negatives

Model	**Accuracy**	**Precision**	**Recall**	**F1-Score**
LR	0.906	0.865	0.963	0.911
SVC	0.974	0.952	0.998	0.974
RF	0.999	0.999	0.999	0.999
DT	0.999	0.999	0.999	0.999
KNN	0.964	0.934	0.998	0.965
XGB	0.999	0.999	0.999	0.999

TABLE II. Performance Metrics for the 6 models on Train set

(TN). In our case we care more about reducing the False Negatives than everything else as this refers to a Risky storage (1) being falsely classified as a Not-Risky (0) storage, the implication and consequence of this factor is much stronger than the FP. Both RandomForest (RF) and XGB have almost perfect performance with negligible false predictions, this might indicate overfitting, or that the models have effectively captured the underlying patterns in the data, or have accurately gauged that the risk rating was derived from fill percent. The SVC showed a strong balance between minimizing FP and FN which could be beneficial in scenarios where both types of errors are costly. Logistic Regression (LR) and K-Nearest Neighbors (KNN) showed a high number of false positives, which might suggest that it is less discriminative compared to the more complex models. We have very high results across the board in Figure II, multiple models scoring near perfect for all the metrics, suggesting overfitting. Moving on we'll be picking SVC and KNN as our final models as we believe the other models are already performing near perfect.

1. *SVC Permutation Importance*

In our study to illuminate the factors contributing to tank farm risk levels, we applied permutation importance, a model inspection technique, to evaluate the influence of each feature in the Support Vector Classifier (SVC). This method entails permuting each feature and observing the effect on model accuracy, providing a measure of feature relevance independent of the model structure.

The strongest importance scores were fill percent - 0.421, tank volume - 0.216, and max volume - 0.118. The rest of the negligible scores of several features indicate that not all collected data contribute meaningfully to risk prediction, highlighting opportunities for model simplification and focus on informative predictors.

2. Statistical Testing and Model Comparison

We first tested the SVC against the Baseline. The test yielded a p-value of 0.002, indicating a statistically significant difference in accuracy scores between the SVC and the baseline model. This suggests that the SVC is capturing complex patterns in the data that the baseline model, which relies on a simple heuristic, cannot. Similarly, the comparison between KNN and the Baseline resulted in a p-value 0.002, reaffirming a statistically significant improvement in predictive performance with the k-NN model.

The small p-values provide strong evidence against the null hypothesis of no difference in model performance, leading us to conclude that both the SVC and k-NN models have superior predictive capabilities compared to the baseline approach.

3. Unseen Test Data

As mentioned earlier, SVC and KNN were picked to be the final model, below are the performance metrics of the models on the unseen test set.

Model	**Accuracy**	**Precision**	**Recall**	**F1-Score**
SVC	0.957	0.594	0.992	0.743
KNN	0.927	0.450	0.828	0.583

TABLE III. Performance Metrics for SVC and KNN on Test Set

Upon final evaluation, a noticeable decline in performance metrics was observed when the models were applied to the unseen test set. Such a discrepancy between the validation and test results often highlights the challenges of overfitting during training or the potential discrepancies between the distributions of the training and test datasets. Another factor that could have contributed to this performance drop is a difference in data distribution between the training and test sets. It's essential to ensure that the train-test split is representative, especially in time-series data or datasets where shifts in context may occur over time. If the test data contains new patterns not present in the training data, model performance can degrade.

IV. CONCLUSION & OUTLOOK

This study embarked on a journey to ascertain the feasibility of using ML algorithms to classify tank farms based on risk levels derived from operational characteristics. Through diligent data collection, preprocessing, and exploratory analysis, we laid the groundwork for the development of several predictive models. Our findings in the model evaluation phase were promising, with algorithms like Random Forest and Support Vector Classifier demonstrating a substantial capacity for distinguishing between risk levels. However, there were major problems with the data used in this study, specifically the "risk" column that was engineered based on fill percentage, a more complex dependent variable is required for a task like this, incorporating historical reports and expert opinion, possibly by acquiring more information about the specific tanks themselves such as age, weight, material etc. All of these factors could contribute towards developing a more robust target variable, allowing for better models in the future.

The subsequent phase of model validation also revealed a reduction in performance metrics when the models were exposed to an unseen test set. This gap between validation and testing performance is a crucial reminder of the intricacies involved in machine learning tasks, particularly the challenges of overfitting and the importance of robust feature selection. It underscores the necessity for rigorous testing regimes that simulate real-world operational conditions as closely as possible.

Despite the decline in the unseen test set performance, the statistical significance of our models, as confirmed through hypothesis testing, signifies a clear advantage over simplistic, heuristic-based approaches. The advanced models' ability to discern complex patterns within the data provides a compelling case for their use in operational risk management in tank farms.

As for the future, there is a clear pathway with regard to data collection, augmentation and refining these models to better suit practical applications. Strategies such as incorporating regularization techniques, deploying ensemble methods, and conducting hyper-parameter optimization are likely to yield models with better generalizability.

[1] F. E. S. Team, As1940:2017 - storage and handling of flammable and combustible liquids standards (2018).

[2] C. Jamison, *A Study of Tank Overfill Incidents*, Ph.D. thesis (2019).

[3] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research **12**, 2825 (2011).

[4] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, Smote: Synthetic minority over-sampling technique, Journal of Artificial Intelligence Research **16**, 321–357 (2002).

[5] M. Boenn, *fitteR: Fit Hundreds of Theoretical Distributions to Empirical Data* (2022), r package version 0.2.0.