Report on Implementation of Machine Learning on Bank Churn Data Exploration

Table of **Contents**

[**INTRODUCTION:** 3](#_Toc194693147)

[**LITERATURE REVIEW** 4](#_Toc194693148)

[**COMPARISON BETWEEN LIME, PFI AND PDP** 6](#_Toc194693149)

[**DATASET:** 6](#_Toc194693150)

[**EXPERIMENTAL DESIGN:** 7](#_Toc194693151)

[**Data Preprocessing –** 7](#_Toc194693152)

[**Data Transformation:** 8](#_Toc194693153)

[**ML Model Development:** 9](#_Toc194693154)

[**ML Results and Evaluation Metrics:** 10](#_Toc194693155)

[**Experimental results and discussion:** 11](#_Toc194693156)

[**CONCLUSION** 14](#_Toc194693157)

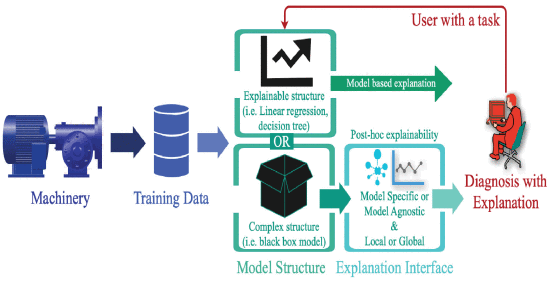
[**LIMITATIONS -** 15](#_Toc194693158)

[**REFERENCES:** 15](#_Toc194693159)

# **INTRODUCTION:**

Explanation of machine learning is the ability of the model to make understandable, comprehensible explanations of the predictions and conclusions. Predictive modelling is required because it will allow the users, primarily scientists, to possess the ability to comprehend the mechanism of the model and why the model will give a specific result. XML makes the model results scientifically accountable because they are manageable, comprehensible, and explanatory. [4]

The XAI seeks to provide a collection of novel or enhanced ML techniques that create understandable models that, combined with effective explanation methods, facilitate end-users to comprehend, believe, and successfully guarantee the management of the next generation of AI technologies [7].

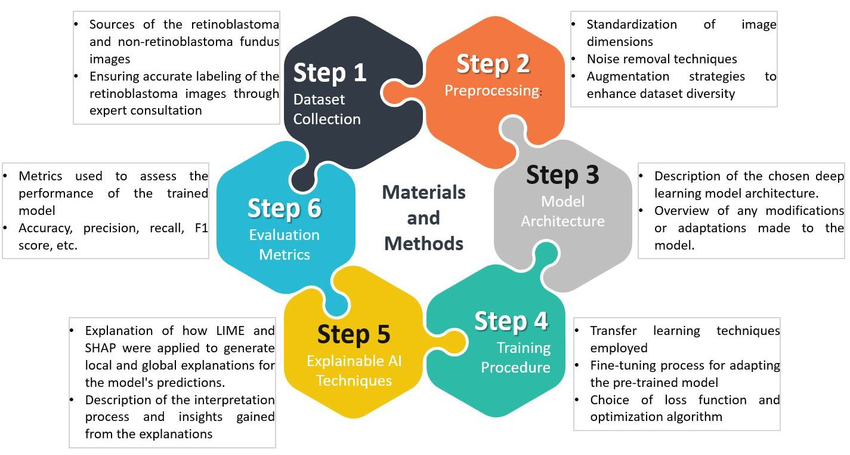


**Figure 1: Concept of Explainable AI [7]**

Explainable Machine Learning (XML) enhances predictive modelling to interpret decisions and make them transparent. This is critical in scientific applications, making the model outputs comprehendible and aligned with the domain. Incorporating XML builds trust and enables sound scientific conclusions.[4]

Machine learning explainability identifies factors influencing model decisions, enhancing transparency through domain knowledge and visualisation tools like heatmaps. It includes intrinsic methods for inherently interpretable models and post hoc techniques for analysing complex models. Explainability is crucial for high-stakes applications, ensuring trust, accountability, and bias detection.[4]

This review examines recent advancements in explainable machine learning (XML), comparing methods, their impact on interpretability, and their pros and cons.



**Figure 2. Illustration of the Overall Framework. [1]**

# **LITERATURE REVIEW**

**LOCAL INTERPRETABLE MODEL-AGNOSTIC EXPLANATIONS (LIME)**

The LIME technique is model-agnostic and is used to produce local explanations through perturbation of the input data and approximation of the model predictions using an interpretable surrogate model. The LIME was also used to present instance-specific explanations for predictive modelling. By creating the locally weighted approximations, the researchers illustrated that the LIME helped find the biases and inconsistencies of machine learning models, especially for the cases when explanations in real-time were needed.[1]

Local Interpretable Model-Agnostic Explanations (LIME) produce local explanations by perturbing input data and fitting a simpler model to estimate predictions [7]. LIME was applied to deep learning models to interpret them better through heatmaps and feature importance overlays. It provided greater transparency in deep neural networks and facilitated easier debugging. Nevertheless, LIME is still worth applying to understand complicated models.[2]

LIME enhances the algorithm's performance with explanations of the local interpretability of a prediction of a complex model. It does this by creating a less complex, interpretable substitute model to interpret the behaviour of a black-box model near a given prediction [5]. LIME can work with black-box models and can highlight feature importances instance-wise. [7]. This makes the model understandable for human users across most fields like healthcare, finance, and law, for which interpretability is highly important. [5]

On the other hand, the LIME is fast and agile in generating explanations for individual instances. In the experiment, the task of deep learning was particularly beneficial, which provided a straightforward, intuitive understanding of the model's behaviour. [1]

LIME suffers from instability in that explanations can be in disaccord with input data changes to a minimal degree. It is essential to tune LIME carefully so that untrustworthy interpretations should not be given [7]. Although effective, the papers also noted the following critical weakness: explanations by LIME may be inconsistent across several runs, decreasing the reliability for high-stakes deployment. In addition, the perturbation-based mechanism of LIME could sometimes generate misleading explanations, especially for highly non-linear models. [1],[2]

**PARTIAL DEPENDENCE PLOTS(PDPs)**

Partial Dependence Plots (PDPs) give global explanations in predictive modelling by showing the average marginal influence of input variables on predictions [6]. They aid in the detection of nonlinear correlations as well as the visualisation of feature effects across a model's prediction domain. PDPs provide clear insights into feature contributions by smoothing the influence of other features. However, they might be misleading in datasets with highly correlated characteristics [3].

The PDPs succeed in justifying ecological assumptions and showing consistent patterns across predictions. However, the authors also recognised that the correlated predictors of environmental data can refute the interpretation of PDP. The PDPs constitute a starting point for improving model transparency, whereby they are traditionally combined with other graphical tools, such as ICE and ALE plots, to cope with their inherent limitations.[3], [6]

Partial Dependence Plots (PDP) provide a global view of feature impact by showing the average impact of a variable on predictions with others being controlled. [7] Partial Dependence Plots (PDPs) were recognised as valuable model interpretability tools for predictive modelling. [3], [6]

A key advantage is the ability to visualise feature effects globally by showing how predictions change with varying feature values while keeping others fixed. This helps researchers confirm domain-specific assumptions and nonlinear relationships [3], [6]. For example, PDPs were utilised to test the effects of land-use factors like forest cover and development on stream health, justifying the predictions of ecology based on model interpretation [3]. PDPs are easy to visualise and interpret and good at detecting monotonic relations but need the assumption of independent features and could be misleading when variables are not independent [7].

PDPs will not provide instance-specific explanations and, therefore, will not be highly effective at understanding predictions at the instance level [7]. One of the substantial limitations is that PDPs assume feature independence, and applying the latter for correlated variables leads to misleading conclusions—a common scenario for accurate data. In the presence of feature correlation, the latter may be unreliable and recommends their combination with ICE plots or ALE plots for showing individual variation and local effects [6]. Likewise, ecological variables tend to show association, which can reduce the validity of the inferences derived from PDP. [3]

Likewise, ecological variables tend to show association, which can reduce the validity of the inferences derived from PDP.

**PERMUTATION FEATURE IMPORTANCE**

PFI is a model-agnostic approach to approximating a feature's importance towards a model's predictive performance by measuring the difference in model performance when the feature values are randomly permuted. These features are tested for the model's performance and accuracy by varying the feature values and viewing the changes in the model performance. [9]

PFI in Random Forest calculates feature importance by determining the loss of accuracy when feature values are randomly specified [8]. The technique optimises resource usage in predictive software failure analysis, lowering calculation costs by up to 99.25% with high accuracy. PFI improves the model's performance by choosing key features, bridging prediction strategy gaps, and being cost-effective and accurate in fault prediction [9].

While PFI increases efficiency, it has some drawbacks, including the possibility of accidentally eliminating critical features, resulting in decreased model fidelity. Furthermore, its implementation can be computationally expensive for big feature sets, resulting in longer processing times. Despite the goal of improving interpretability, PFI may make it more difficult to completely grasp the model's decision-making process due to frequent feature reductions [8].

The value of the permutation feature in Random Forest is a valuable tool for investigating how individual features influence the model's overall prediction performance [9]. Due to the improvement provided by these advantages, we utilised PFI with our Random Forest model so that the model would be a more precise predictor with computation efficiency [8].

## **COMPARISON BETWEEN LIME, PFI AND PDP**

Popular explainable AI methods for interpreting machine learning models include Partial Dependence Plots (PDP), LIME, and Permutation Feature Importance (PFI). While assuming feature independence, which may not always hold, PFI provides a global perspective by quantifying the decrease in model accuracy when a feature is randomly shuffled. This highlights the overall relevance of features. Although it can be unstable and sensitive to data perturbations, LIME, on the other hand, offers local, instance-specific explanations by fitting a basic surrogate model around a prediction. This makes it helpful for comprehending particular decisions. PDP helps identify nonlinear trends by providing a global picture of how changes in a single feature impact the model's predictions on average. Still, it loses its accuracy when features are coupled. LIME delves into the logic behind particular predictions, whereas PFI and PDP provide insight into model behaviour across the dataset, making them corresponding techniques for improving model interpretability.

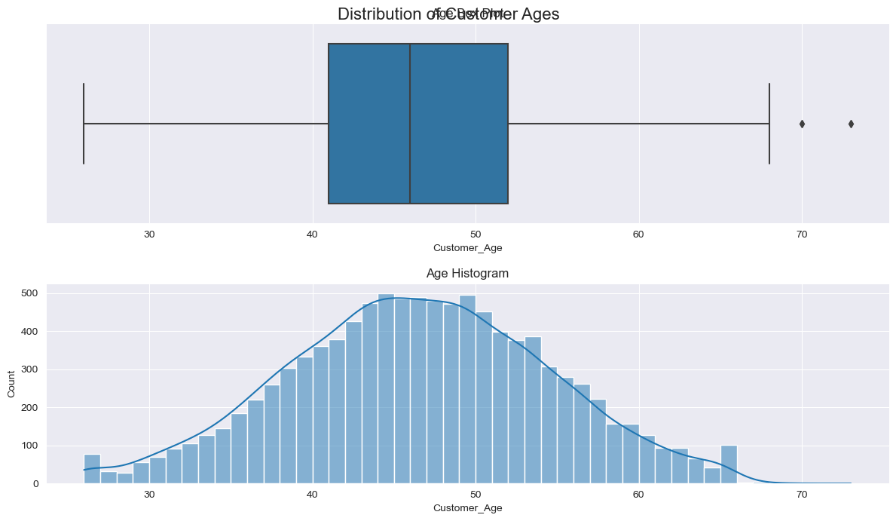
# **DATASET:**

Dataset Name: **Bank Churn Data Exploration And Churn Prediction**

In this dataset, there are **10,127 bankchurners'** demographic information, credit card usage trends, and customer status (whether they have churned or are still active), which are all recorded in the BankChurners dataset. **Customer age, gender, income, education degree, credit limit, number of transactions, and spending patterns are essential characteristics.** Attrition\_Flag, the primary target variable, determines if a client is an "Existing Customer" or an "Attrited Customer." The dataset may have been utilised for churn modelling because it contains two columns about predictions made by a Naive Bayes classifier. To determine the elements that lead to client attrition, this dataset helps examine consumer behaviour and develop prediction models.

# **EXPERIMENTAL DESIGN:**

## **Data Preprocessing –**

* Data Selection: Irrelevant columns were removed.
* **Exploratory Data Analysis (EDA):** 
  + The distribution of customer age was analysed using box plots and histograms. 
  + Gender distribution was calculated for overall Platinum cardholders and Blue cardholders.

A white background with black text

AI-generated content may be incorrect.

* + The distribution of dependent count was visualised using box plots and histograms.

A close-up of a graph

AI-generated content may be incorrect.

* + Proportions of education level, marital status, income category, and card category were calculated and summarised.

A screenshot of a computer screen

AI-generated content may be incorrect.

* + Distribution of months on book, total relationship count, credit limit, and total transaction amount were analysed using box plots and histograms.

A diagram showing a number of credit limit

AI-generated content may be incorrect.

## **Data Transformation:**

* + The target variable "Attrition\_Flag" was encoded to numerical values (0 and 1).
  + The "Gender" variable was encoded to numerical values (0 and 1).
  + Categorical variables ("Education\_Level", "Income\_Category", "Marital\_Status", "Card\_Category") were one-hot encoded.
  + Original categorical columns and "CLIENTNUM" were dropped.
* Correlation Analysis:
* Pearson correlation was computed, and a heatmap was generated to visualise feature correlations.

A screenshot of a graph

AI-generated content may be incorrect.

* Oversampling:
* The SMOTE technique addressed the class imbalance in the target variable.

# **ML Model Development:**

**Random Forest Classifier:**

Arandom forest classifier is a good choice when data has a non-linear relationship and is robust to outliers.

* **Non-linear Relationships:** Random Forest excels at capturing complex, non-linear relationships. The plots of Total\_Trans\_Amt and Credit\_Limit show non-linearities in the data.
* **Robustness to Outliers:** Some customers have high transaction amounts in our data. Random Forest is generally robust to outliers, meaning these extreme values are less likely to disproportionately influence the model's performance.

**AdaBoost Classifier:**

We've expressly set the learning\_rate to 0.7, a crucial hyperparameter that controls how much each boosting round influences the final prediction.

**Support Vector Machine (SVM):**

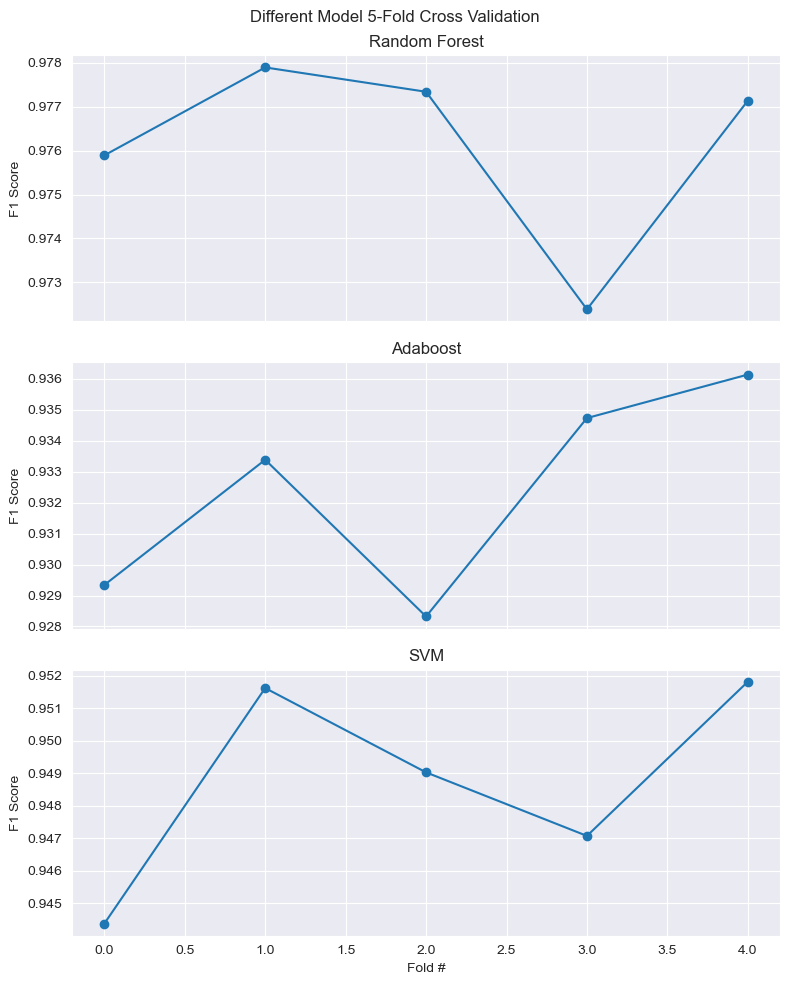
SVMs are excellent at finding optimal hyperplanes to separate classes, and they are especially effective in this dataset type as we have high dimensions. We use a radial basis function (RBF) kernel, allowing SVM to model complex, non-linear boundaries.

We use make\_pipeline from scikit-learn. Because it neatly chains together data preprocessing and modelling steps.

**StandardScaler():** Before training data for our models, especially SVM, it's crucial to scale it. StandardScaler ensures all features have the same scale, preventing features with larger scales from dominating the model.

**Cross-Validation:**

We use cross\_val\_score with five folds (cv=5). This means each model is trained and validated 5 times on different subsets of the training data.



# **ML Results and Evaluation Metrics:**

**F1 score:** We chose F1 as our evaluation metric because it balances precision and recall. This is particularly important in churn prediction, where we want to minimise false positives (predicting churn when the customer stays) and false negatives (missing customers who will churn).

By averaging the F1 scores across the five folds, we get a more robust estimate of each model's performance than a single train-test split.

On our oversample dataset, we got a 98% F1 score for the Random Forest model, 93% on the Adaboost model, and SVM got 95%, which is a relatively good model. In the original data, the models say the F1 score for random forest is 0.83, significantly lower than the oversampled data. For AdaBoost, it decreased to 0.73, and for SVM, it again reduced to 0.68.

**Precision-Recall Curve**:

The **Precision-Recall curves** indicate exceptional model performance, with near-perfect classification for both classes. Class 0 (Customers got Churned) achieves a perfect AUC of 1.000, while class 1(Customers who are not churned) scores 0.994. The micro-average AUC of 0.999 highlights the model's overall robustness.

A screenshot of a computer

AI-generated content may be incorrect.

Fig 3 – Precision-Recall Curve

# **Experimental results and discussion:**

We used LIME on the Random Forest classifier from the 5th test data instance. The model predicted a probability of 0.22 for class 1. (likely churn). The intercept (base prediction) is 0.424, and the local prediction is 0.223.LIME identified the customer's single status, contact count (2-3), and not having income below $40k or a High School/Graduate education as factors decreasing churn likelihood. No substantial factors increasing churn were found. This suggests the model associates this profile (5th instance) with lower churn risk.

A white background with text

AI-generated content may be incorrect.

Fig 4 – the LIME result, for instance, 5

From the 15th test data instance, the model predicted a probability of 0.39 for class 1 (likely churn) for this instance. The intercept (base prediction) is 0.2969, and the local prediction is 0.3410. Being single and having a graduate education increased to be a churn prediction. Conversely, having a contact count of 2, not having income below $40K, a transaction change of 0.47, and not having a high school education decreased the predicted churn probability. The model weighs single status and graduate education as stronger indicators of potential churn for this customer.

A screen shot of a computer

AI-generated content may be incorrect.

Fig 5 – the LIME result, for instance, 15

While we implement **permutation importance** to understand which features are most influential for predicting churn customers, we foundTotal\_Trans\_Ct (Total Transaction Count) is the most influential, followed by Total\_Trans\_Amt (Total Transaction Amount). The number of customer relationships (Total\_Relationship\_Count) also has a moderate impact. Gender has significantly less predictive power. This suggests that the model heavily relies on customer activity to determine churn risk, with customer transactional behaviour being the strongest indicator. In contrast, other customer attributes are less important in this model**.**

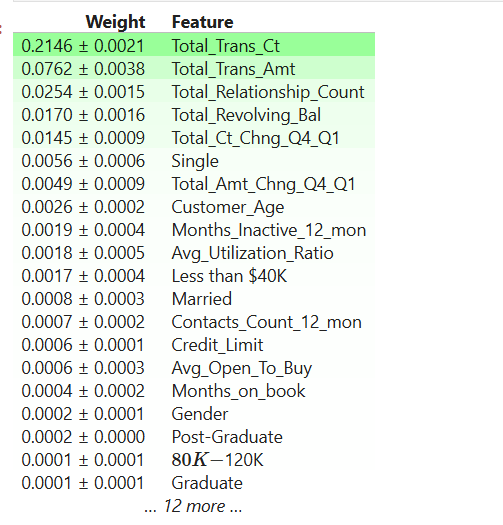


Fig 6 – Permutation Importance of features.

But while we implement **PDP** on the influential features, it shows that Total\_Revolving\_Bal and Dependent\_count significantly impact predictions. A low revolving balance strongly influences the model, while having one dependent lowers prediction, but two or more increase it. Total\_Trans\_Ct, Total\_Trans\_Amt, and Customer\_Age have little to no effect, as their plots remain primarily flat.

A graph of a number of graphs

AI-generated content may be incorrect.

Fig 7 – Partial Dependency plot

# **CONCLUSION**

This literature review explores key methodologies for machine learning interpretability, focusing on SHAP, LIME, and Partial Dependence Plots (PDP). Grounded in game-theoretic principles, SHAP provides local and global explanations, making it particularly valuable in high-stakes domains like finance and healthcare, where fairness and reliability are critical. LIME, a model-agnostic approach, offers rapid local explanations but can be sensitive to variations across runs, impacting consistency. PDP effectively visualises the impact of individual features on predictions but struggles with correlated features, limiting its reliability in complex models. Future research should improve SHAP’s computational efficiency, enhance LIME’s robustness, and refine PDP to handle feature dependencies better. Additionally, integrating these techniques into deep learning models and hybrid frameworks can further enhance explainability, fostering greater transparency and trust in AI-driven decision-making.

LIME results said that being single and having a graduate education increased churn risk, while factors like higher contact count and income above $40K reduced it. The explanations are instance-specific, highlighting how different factors influence customer churn**.** OnPermutation importance, Total\_Trans\_Ct (Total Transaction Count) was found to be the most important feature, followed by Total\_Trans\_Amt (Total Transaction Amount) and Total\_Relationship\_Count. Gender had less influential power, suggesting that customer transactions more influence churn than demographic attributes. And PDPs showed Total\_Revolving\_Bal and Dependent\_count significantly impact churn predictions, while Total\_Trans\_Ct, Total\_Trans\_Amt, and Customer\_Age had minimal influence.

# **LIMITATIONS -**

LIME's explanations vary across instances, meaning it lacks global interpretability and may be unstable with different sampling strategies. Permutation importance only measures feature importance at the model level. It does not explain how individual features impact predictions and assume feature independence, which might not hold in real-world data, leading to misleading conclusions.

# **REFERENCES:**

[1]

Bader Aldughayfiq, F. Ashfaq, N. Z. Jhanjhi, and M. Humayun, “Explainable AI for Retinoblastoma Diagnosis: Interpreting Deep Learning Models with LIME and SHAP,” *Diagnostics*, vol. 13, no. 11, pp. 1932–1932, Jun. 2023, doi: https://doi.org/10.3390/diagnostics13111932.

[2]

K. Ashraf, Sadia Nawar, Md. Hamid Hosen, Mohammad Tanvirul Islam, and Mohammed Nazim Uddin, “Beyond the Black Box: Employing LIME and SHAP for Transparent Health Predictions with Machine Learning Models,” Mar. 2024, doi: https://doi.org/10.1109/icaccess61735.2024.10499522.

[3]

K. O. Maloney *et al.*, “Explainable machine learning improves interpretability in the predictive modelling of biological stream conditions in the Chesapeake Bay Watershed, USA,” *Journal of Environmental Management*, vol. 322, pp. 116068–116068, Nov. 2022, doi: https://doi.org/10.1016/j.jenvman.2022.116068.

[4]

R. Roscher, B. Bohn, M. F. Duarte, and J. Garcke, “Explainable Machine Learning for Scientific Insights and Discoveries,” *IEEE Access*, vol. 8, pp. 42200–42216, 2020, doi: https://doi.org/10.1109/access.2020.2976199.

[5]

Zhang, Y., Chen, L., & Tian, Y. (2025). A Method for Evaluating the Interpretability of Machine Learning Models in Predicting Bond Default Risk Based on LIME and SHAP. arXiv preprint arXiv:2502.19615.,

[6]

Hall, P. (2019, August). On the art and science of explainable machine learning: Techniques, recommendations, and responsibilities. In Proceedings of the KDD (Vol. 19).,

[7]

Shreyas Gawde, S. Patil, S. Kumar, P. Kamat, K. Kotecha, and Sultan Alfarhood, “Explainable Predictive Maintenance of Rotating Machines using LIME, SHAP, PDP, ICE,” *IEEE Access*, pp. 1–1, Jan. 2024, doi: <https://doi.org/10.1109/access.2024.3367110>.

[8]

Masahiro Hayashitani, Eiji Yumoto, and K. Nishihara, “Highly Interpretable Agitation Detection Model by Reducing Features with Permutation Importance,” pp. 1–3, Nov. 2024, doi: https://doi.org/10.1109/healthcom60970.2024.10880767.

[9]

A. Khan, A. Ali, J. Khan, F. Ullah, and M. Faheem, “Using Permutation-Based Feature Importance for Improved Machine Learning Model Performance at Reduced Costs,” *IEEE Access*, vol. PP, no. 99, p. 1, Jan. 2025, doi: https://doi.org/10.1109/ACCESS.2025.3544625.