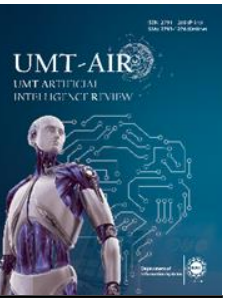
UMT Artificial Intelligence Review (UMT-AIR)
Volume 6 Issue 1, Fall 2026
ISSN$_{(P)}$: 2791-1276, ISSN$_{(E)}$: 2791-1268
Homepage: https://journals.umt.edu.pk/index.php/UMT-AIR



Title:	**PRISM: A Framework for Preprocessing Impact Analysis in Machine Learning via Signature-based Modeling**
Author (s):	Abu Bakar Shabbir[1] and Khadija Amber[2]
Affiliation (s):	[1]School of Systems and Technology, University of Management and Technology, Lahore, Pakistan [2]Department of Computer Science, Bahauddin Zakariya University, Bosan Road, Multan, Pakistan
DOI:	https://doi.org/10.32350.umt-air.61.01
History:	Received: January 22, 2026, Revised: February 14, 2026, Accepted: March 20, 2026, Published: June 08, 2026
Citation:	A. B. Shabbir, K. Amber, "PRISM: A Framework for Preprocessing Impact Analysis in Machine Learning via Signature-based Modeling," *UMT Artif. Intell. Rev.*, vol. 6, no. 1, pp. 01-23, Jun. 2026, doi: https://doi.org/10.32350.umt-air.61.01.
Copyright:	© The Authors
Licensing:	This article is open access and is distributed under the terms of Creative Commons Attribution 4.0 International License
Conflict of Interest:	Author(s) declared no conflict of interest

A publication of
Department of Information System, Dr. Hasan Murad School of Management
University of Management and Technology, Lahore, Pakistan

PRISM: A Framework for Preprocessing Impact Analysis in Machine Learning via Signature-based Modeling

Abu Bakar Shabbir[1*] and Khadija Amber[2]

[1]School of Systems and Technology, University of Management and Technology, Lahore, Pakistan
[2]Department of Computer Science, Bahauddin Zakariya University, Bosan Road, Multan, Pakistan

ABSTRACT The quality and transformation of input data are crucial for Machine Learning (ML) performance, along with the model architecture. Data Preprocessing (DP) is an important factor in the learning outcome. However, it has been hardly addressed as a formally modeled transformation process. The current study argued for a new analytical framework known as PRISM (Preprocessing Impact Signature Modeling). Preprocessing can be systematically described as behavioral transformation operators that impact the process of model learning over datasets and algorithms. PRISM gives a structured interpretation of behavioral differences not only based on the differences in performance after preprocessing but also on two constructs: Preprocessing Impact Vector (PIV) and Preprocessing Signature Tensor (PST). The PIV provides performance differences in several performance measures, and the PST further extends these differences between sets of models and datasets for comparative and multi-dimensional analysis. It is evaluated on six real-world datasets, including classification and regression problems, with different ML models, such as linear models, tree-based models, ensemble models, and distance-based models. Experimental results show that there is no simple, uniform, and performance-driven preprocessing effect instead, there are structured, model-dependent, and dataset sensitive patterns. Tree-based models are insensitive to preprocessing transformations, while distance-based models are quite sensitive. Moreover, preprocessing is more consistently successful in stabilization of the model than in absolute accuracy gains, especially in the case of noise and data heterogeneity. PRISM offers an interpretable, scale-invariant, and scalable analytical framework for understanding the behavior of preprocessing in ML systems. The proposed approach moves towards a more principled and data-driven approach for preprocessing analysis and designing ML pipelines.

INDEX TERMS behavioral modeling, data preprocessing, machine learning, preprocessing impact vector, preprocessing signature tensor

I. INTRODUCTION

In recent years, Machine Learning (ML) has emerged as a pivotal paradigm in the field of data-driven systems, allowing them to make intelligent decisions in various sectors. These sectors include healthcare diagnostics, financial forecasting, industrial monitoring, environmental modeling, and predictive analytics. Although the models have become increasingly sophisticated in architecture, optimization techniques, and computational capacity, recent research consistently points to the fact that model performance is not necessarily correlated with model sophistication. Rather, it is now largely determined by the quality, structure, and changes of the input data. This transformation has changed the Data Preprocessing (DP) into a crucial part of the ML process, not just a preliminary step.

*Corresponding Author: ABUBAKARSHABBIR64@GMAIL.COM

Although it is a crucial step, in the literature the preprocessing is still considered as a procedural or engineering step, and is not explicitly modelled as a transformation process. For instance, Barouki et al. [1] found that the real-world implications of complex systems are due to interconnections and multi-factorial interactions. This systemic approach makes sense from the "intuitive" point of view that the impact of the preprocessing steps should be analyzed in the context of the features of the dataset and/or the behavior of the adopted model. Similarly, Hammoudeh et al. [2] demonstrated that the individual training samples may have measurable impact on model predictions. This further highlights the structured and non-trivial impact of data transformations on learning behavior. These approaches are usually instance level influences, though, and do not represent the changes in behavior across models.

The field of preprocessing techniques and pipelines has a considerable amount of literature. In [3], Koukaras et al. provided a structured taxonomy of different types of preprocessing operations, such as cleaning, transformation, feature construction, feature selection, and dimensionality reduction. This highlights their contribution to enhance robustness and interpretability. As reported by Fernandes et al. [4], data preparation can be as much as 80% of the ML process. Both of these studies are essentially descriptive, however, and there is no formal way to quantify the change in model behavior, in a consistent and comparable manner, due to the preprocessing.

The challenge of dealing with the missing data is an example that further highlights the complexity of preprocessing effects. Emmanuel et al [5] and Hameed et al [6] demonstrated that the different imputation methods (such as KNN, regression and EM) produce different results based on the missingness mechanism (MCAR, MAR, MNAR) and the nature of the data. The results provide evidence that preprocessing is highly context-dependent. However, the evaluation occurs based on performance rather than structured behavioral modelling. Likewise, Călin et al. [7] and Biju et al. [8] reported better predictive results in agricultural and industrial data, respectively. This was done after applying a domain-specific preprocessing, with respect to the same data without preprocessing but they did not generalize their results into a single analytical representation.

From automation point of view, there are some frameworks, such as Preptimize [9] and DIANA [10] that consider adaptive preprocessing selection mechanisms based on optimization and knowledge-based systems. In the end, these methods focus more on pipeline optimization and predictive performance and less on how preprocessing affects the behavior of the model in various contexts. This means that they do not offer a theoretical framework for comparative analysis between datasets and learning algorithms.

One common shortcoming in all of the literature reviewed is the lack of systematic consideration of preprocessing as a part of the pipeline. Additionally, in all of the reviewed literature, preprocessing is either considered as an isolated step in the pipeline or as a fixed pipeline step. There is no unified scale-invariant framework that can be applied consistently to document the impact of preprocessing transformation on a model's behavior for diverse datasets and families of algorithms.

In order to solve this, this study came up with the PRISM (Preprocessing Impact

Signature Modeling) framework. PRISM introduces the concept of a redefinition of preprocessing as a behavior transformation operator instead of a fixed data preparation process. It provides two structured representations, the Preprocessing Impact Vector (PIV) and the Preprocessing Signature Tensor (PST). The aim is to quantify and analyze the changes in model performance in a standard and comparable way as a result of preprocessing. The formulation allows to systematically assess the interactions between data structures, model inductive biases, and different preprocessing strategies.

This work is essentially moving the role of preprocessing from a supporting pipeline element, to a first-class analytical element within ML systems. The formalization of the preprocessing as a measurable behavioral change opens the ground for analyzing, interpreting, and comparing the preprocessing effects in a (unified and) structured way across a variety of datasets and learning models.

The rest of this study is structured as follows. In section 4, the PRISM framework is described, along with formal definitions of the PIV and PST. The datasets employed in the study are explained in section 5, both in classification and regression. Repeatedly, consistency and reproducibility of the experiments are ensured by describing the preprocessing pipeline in section 6 and the experimental design. Section 7 presents the experimental setup, including model setups, metrics, and dual-pipeline approach. In section 8, the experimental results and analysis are presented, focusing on the changes in behavior induced by the preprocessing in the datasets and models. Lastly, in section 9, future research directions on data-centric ML and preprocessing-aware modeling frameworks are discussed.

The results of the research in the field of ML and data-centric systems in recent years consistently indicate that the performance of the models is not only determined by the complexity of the algorithm. However, it is also significantly influenced by the representation of the data, the data quality interventions, and the preprocessing of the data. Although there are various applications where preprocessing plays an important role in the ML pipeline, it remains an under-formalized step in the literature, as observed. Examples include healthcare, industrial systems, federated learning, environmental analysis, and big data analytics. Yet, most of the previous research takes a procedural or engineering approach to preprocessing as opposed to a well-defined, quantifiable transformation process between models and datasets. This restriction provides motivation for the development of a single analytical framework, such as PRISM, where preprocessing is a behavioral transformation operator.

Barouki et al. [1] have a foundational perspective that recognizes the complexity and interdependence of variables at the system level, explored in the context of interconnected public health, environmental and human activity impacts on change. Their findings highlight thatcomplex outcomes are the result of multi-factor interactions rather than single factors. This is also in line with PRISM's assumption that preprocessing effects are driven by interactions between the data structure and the model inductive bias. Likewise, Hammoudeh et al. [2] formally studied the influence of training data on the model prediction, which showed that single data points may have a profound impact on the model prediction. They believe that retraining and gradient-based influence

frameworks suggest that data transformations have structured effects on model behavior but on an instance level rather than on data transformation level modeling.

There is a considerable amount of literature directly dedicated to preprocessing as a crucial step in the pipeline of ML. Koukaras et al. [3] presented a thorough literature review and classified the preprocessing techniques into main categories. These included data cleaning, data transformation, data feature construction, data feature selection, and data dimensionality reduction. They highlighted the robustness and interpretability of preprocessing, however, it is highly dependent on the task and data. The method they used was, however, procedural and did not quantify the impact of the preprocessing as a normalized behavior change; PRISM does. Likewise, Fernandes et al. [4] pointed out that data preparation can account for as much as 80% of the entire data science process, which includes profiling, integration, transformation, and repair. Their work provided a detailed structural decomposition but did not have a common mathematical model of preprocessing effects for all models.

The importance of preprocessing is further highlighted by the research on handling the missing data. The imputation approaches, such as mean, regression, KNN, expectation-maximization, and fuzzy clustering methods are analyzed in detail by Emmanuel et al. [5] and Hameed et al. [6]. Both of the studies determined that none of the methods of imputation is universally best as its effectiveness depends on the dataset and on the nature of missingness (MCAR, MAR, MNAR). This directly supports the main premise of PRISM, that there is no global transfer of preprocessing effects. The studies, however, assessed the benefits of preprocessing with respect to the accuracy of the predictions that can be made without also modelling cross-model behavioral variation.

Empirical domain-specific studies also confirm the sensitivity to preprocessing. Imputation and outlier handling have been demonstrated to have a significant positive effect on the prediction of agricultural yield, with an increase in R^2 from 0.723 to 0.938 through imputation and outlier handling, as demonstrated by Călin et al. [7]. Likewise, Biju et al. [8] investigated the impact of sensor noise in industrial environments and showed that sensor noise is modeled by different distributions (Gaussian, Flicker, Brown and Uniform), and has a different degree of degradation. Both studies showed that preprocessing had an impact on the results of the model but they did not provide a single analytical form, such as PRISM, that encompasses the impact of processing.

Automation-based framework extends the research in the preprocessing. Usmani et al. [9] suggested an automated framework called Preptimize. This framework combines data profiling, data imputation, data transformation, and model selection of forecasting models using genetic algorithm. Preprocessing and forecasting strategies are dynamically selected based on the characteristics of the datasets, including stationarity, skewness, and missingness pattern. This not only saves time and manual work but also emphasizes optimization rather than the interpretability of effects of the preprocessing. By analogy, Sancricca et al. [10] proposed DIANA, an AI approach focused on the data that suggests data preprocessing activities based on the knowledge base of past empirical results. While DIANA brings adaptive preprocessing decision-making, it does not structure preprocessing effects as similar,

comparable signatures across models.

Berahmand et al. [11] offered a comprehensive taxonomy of autoencoders, such as variational, convolutional, recurrent, and graph-based autoencoders. The models implicitly compress the data to lower dimensional representations, which makes it more efficient and robust to learn. Their emphasis was on representation quality, however, compared to none of their emphasis was on comparative behavioral impact across model families, which PRISM makes explicit as PIV and PST formulations.

In large-scale and distributed systems, Lopez-Miguel [12] mentioned feature selection and discretization as the important preprocessing operations in the context of big data environment. The study classified the techniques into three categories: filter, wrapper, and embedded techniques, and provided a discussion of scalability issues. No attempt, however, was made to formalize the effect of such transformations on downstream model behavior in various learning paradigms. In the same manner, Kamencay et al., [13] elaborated on the applications of sensors as well as DL in various fields including IoT, NLP, and computer vision and showed how preprocessing can improve the performance of complex real-world systems. However, the analysis was not theoretically unified but was still application-oriented.

The importance of data distribution for the performance of models is also highlighted in federated and transfer learning research. Shuang Dai et al. [14] reviewed transfer learning and federated learning in the context of online learning, explaining the problems including non-IID data, system heterogeneity, and privacy constraints. They showed that the learning stability and the behavior of convergence are highly sensitive to the variations in data distribution, which, conceptually, is similar to PRISM's perspective on the concept of preprocessing as a context dependent transformation. However, they were not concerned with preprocessing impact modeling, as they were interested in distributed learning systems.

The significance of preprocessing is verified in industrial and applied ML studies. Park et al. [15] demonstrated that the use of a combination of scaling, dimensionality reduction, and oversampling methods (SMOTE, ADASYN) increases semiconductor defect prediction performance. ML-based imputation algorithm approaches, especially the Random Forest (RF) algorithm, are shown to be more useful than the traditional statistical methods to predict KPIs in the telecom domain by Dahj et al. [16]. Both studies demonstrated that the choice of preprocessing critically affects model performance. However, it does not use generalizable behavioral modeling.

The reviewed literature showed a general agreement that preprocessing is considered as a crucial step in the success of ML, evaluated in disparate and task-specific and metric-specific ways. Current methods either include pipeline optimization [9, 10], specific preprocessing methods [5, 6], feature engineering techniques [11] or influence analysis at instance level (Hammoudeh & Lowd). However, there is no common and scalable, multi-model representation of the preprocessing effects in any of these studies.

This space is the direct motivation for the PRISM framework proposed in this research. PRISM builds upon the existing literature on procedural evaluation by formalizing preprocessing as a

transformation and accounting for its effect in terms of PIVs) and PSTs to model the preprocessing stage as a structured behavioral process. PRISM allows systematic, comparable, and interpretable analysis of the effects of preprocessing across datasets, models, and metrics in a single analytical framework, unlike previous work. Figure 1 shows the issues with traditional evaluation of preprocessing, as the models would behave differently from one performance metric to another, often in an unclear way. It highlights the importance of a structured framework, such as PRISM for systematic documentation of impacts of the preprocessing stage.

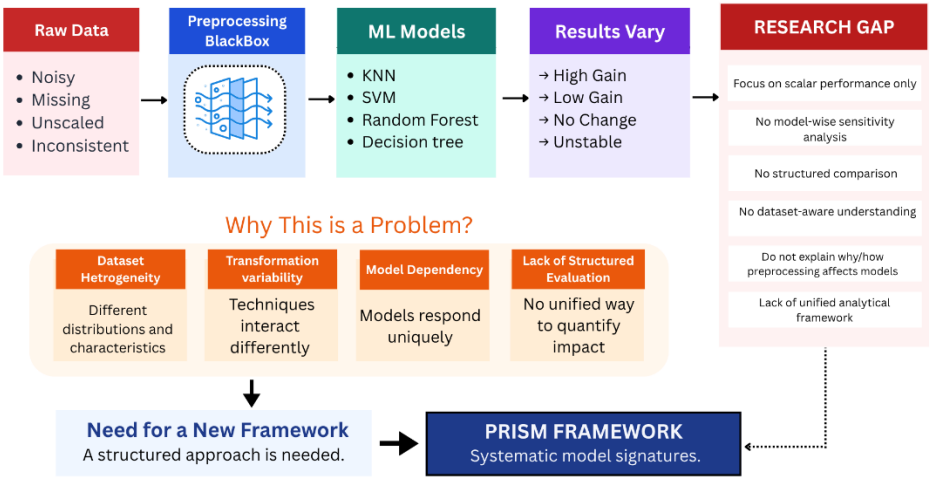


FIGURE 1. Limitations of traditional preprocessing evaluation in ML pipelines, highlighting inconsistent model behavior and the need for a structured framework (PRISM)

A. PROPOSED FRAMEWORK: PRISM

The proposed PRISM framework presents a systematic and analytical paradigm of reinterpreting the role of preprocessing in ML systems. In traditional pipelines, the preprocessing is generally assumed to be a preliminary stage, the effectiveness of which is measured by absolute changes in the performance metrics (accuracy, F1-score or error reduction). Nevertheless, these scalar assessments cannot be used to represent the intrinsic behavioral changes that are brought about by preprocessing in various models and datasets. Conversely, PRISM re-frames preprocessing as a relative and normalized transformer of model performance, the effects of which can only be observed via relative and normalized model performance changes. This view allows a more expressive and interpretable way of thinking about how preprocessing interacts with both dataset characteristics and inductive biases of models.

The hypothesis, that preprocessing fails to have a consistent or globally transferable impact, lies at the heart of PRISM. Rather, the same preprocessing operations may have different responses given different interactions between dataset structure and the learning mechanism of the model. This drives the desire to have a representation

that captures changes in relative performance in a scale-invariant fashion, thus allowing meaningful comparisons across models, datasets, and performance measures. Figure 2 illustrates a multi-layered PRISM architecture that combines datasets, preprocessing transformations, ML models, and evaluation metrics with a feedback-driven analytical engine.

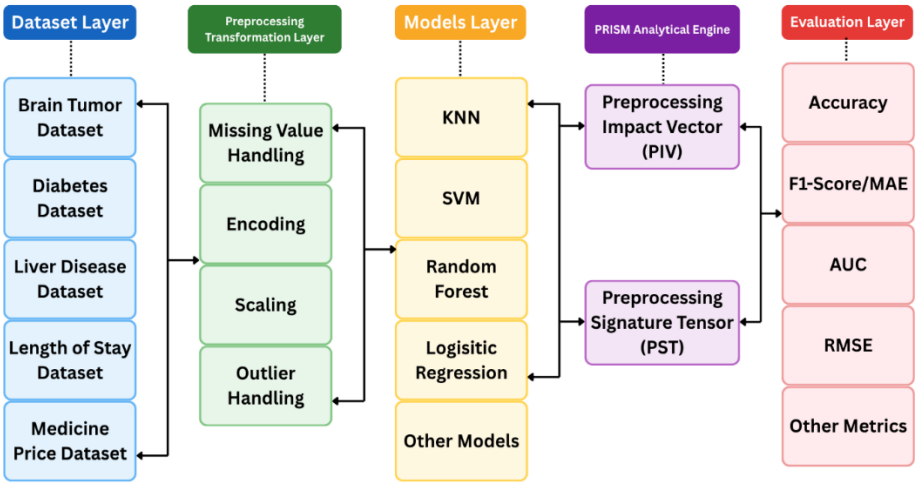


FIGURE 2. PRISM framework showing datasets, preprocessing, models, analytical engine (PIV, PST), and evaluation layers with a feedback loop for iterative optimization.

1) PREPROCESSING IMPACT VECTOR (PIV)

To formally quantify the effect of preprocessing, PRISM defines the PIV for a given model–dataset pair (m, d). The PIV encodes the normalized change in performance induced by a preprocessing transformation. Let $f_{\text{pre}}(d)$denote the preprocessed version of dataset d. For a set of task-consistent evaluation metrics $\{\phi_1, \phi_2, \dots, \phi_p\}$, each component of the PIV is defined as a normalized performance shift:

$$\Delta\phi_i^{(m,d)} = \frac{\phi_i(m, f_{\text{pre}}(d)) - \phi_i(m, d)}{|\,\phi_i(m, d)\,| + \epsilon}$$

where, $\epsilon > 0$ ensures numerical stability. The resulting vector is expressed as:

$$\mathbf{PIV}_{m,d} = [\Delta\phi_1^{(m,d)}, \Delta\phi_2^{(m,d)}, \dots, \Delta\phi_p^{(m,d)}]$$

This normalization ensures scale invariance, allowing fair comparison across metrics with different ranges and magnitudes. To maintain semantic consistency, PRISM defines PIV within task-specific metric spaces. For classification tasks, metrics, such as accuracy, F1-score, and AUC are used. Whereas, for regression tasks, error-based metrics, such as RMSE and MAE are considered. By doing so, PRISM avoids mixing heterogeneous evaluation criteria within a single representation.

Through this formulation, preprocessing is no longer interpreted as a simple improvement mechanism, rather as a differential operator that induces measurable behavioral changes in model performance.

2) PREPROCESSING SIGNATURE TENSOR (PST)

While the PIV captures preprocessing impact for an individual model–dataset pair, PRISM extends this representation to a global analytical structure through the PST. This tensor aggregates PIVs across a set of models $\mathcal{M}$ and datasets $\mathcal{D}$, providing a unified view of preprocessing behavior.

Formally, the PST is defined as:

$\mathcal{T} \in \mathbb{R}^{|\mathcal{M}|\times|\mathcal{D}|\times p}$

where, each entry is given by:

$\mathcal{T}_{i,j,:} = \mathbf{PIV}_{m_i,d_j}$

This tensor-based formulation allows preprocessing to be interpreted as a multi-dimensional transformation field over the joint model–dataset space. Unlike traditional scalar benchmarking, the PST enables structured analysis of how preprocessing influences different models across diverse datasets and evaluation metrics. It facilitates the identification of global patterns, including model-specific sensitivity trends, dataset-dependent effectiveness, and metric-level variability under preprocessing transformations.

3) PREPROCESSING AS A BEHAVIORAL TRANSFORMATION OPERATOR

A key conceptual contribution of PRISM is the formalization of preprocessing as a behavioral transformation operator rather than a standalone data preparation step. Let $\mathcal{M}(m,d)$ denote the performance of model m on dataset d. The preprocessing-induced behavioral change can be expressed as:

$\mathbf{PIV}_{m,d} = \mathcal{M}(m, f_{\text{pre}}(d)) - \mathcal{M}(m,d)$

More generally, the interaction can be modeled as a joint functional:

$\mathbf{PIV}_{m,d} = \mathcal{G}(m,d,f_{\text{pre}})$

which explicitly captures the dependency of preprocessing effects on the model, dataset, and transformation itself. This formulation emphasizes that preprocessing has no intrinsic utility in isolation. Its impact emerges only through its interaction with a specific learning algorithm and data distribution. Consequently, preprocessing should be understood as a context-sensitive operator, governed by both data geometry and model inductive bias.

4) STATISTICAL ROBUSTNESS OF PREPROCESSING SIGNATURES

To ensure that the observed preprocessing effects are robust and not artifacts of random variation, PRISM incorporates a statistical formulation of the PIV. Instead of relying on single-run differences, the preprocessing impact is estimated as an expectation over multiple experimental runs:

$\mathbf{PIV}_{m,d} = \mathbb{E}[\Delta \boldsymbol{\phi}_{m,d}] \pm \boldsymbol{\sigma}_{m,d}$

where $\sigma_{m,d}$ represents the variability across repeated trials, such as cross-validation folds or bootstrap samples. This probabilistic formulation ensures that preprocessing signatures are stable, reliable, and statistically meaningful. In practice, statistical significance of performance shifts can further be validated using paired hypothesis testing procedures, strengthening the reliability of the analysis.

5) SIGNATURE INTERPRETATION AND MODEL SENSITIVITY ANALYSIS

PRISM enables systematic analysis of preprocessing sensitivity by comparing PIV representations across models and datasets. The difference in preprocessing response between two models m_i and m_j on a dataset d can be quantified using a distance measure in the PIV space:

$$\text{Dist}(m_i, m_j \mid d) = \| \mathbf{PIV}_{m_i,d} - \mathbf{PIV}_{m_j,d} \|_2$$

This formulation provides a principled basis for identifying clusters of models with similar preprocessing behavior and for quantifying robustness patterns. Empirically, models that rely on feature geometry—such as distance-based methods—tend to exhibit large-magnitude PIVs, indicating high sensitivity to preprocessing operations, such as scaling and normalization. In contrast, tree-based models typically exhibit near-zero PIV magnitudes, reflecting invariance to monotonic transformations of input features. These observations support the central premise of PRISM that preprocessing sensitivity is an emergent property of model–data interaction, rather than a fixed characteristic of datasets alone.

6) CONTRIBUTION OF PRISM

To conclude, PRISM framework presents a single and understandable model to study preprocessing effects in ML systems. PRISM provides a means of transforming the traditional performance-based evaluation paradigm into a behavior-based analysis paradigm based on tensors and a global structure. Such a paradigm shift can enable researchers to go beyond the isolated metric improvements and instead see the fundamental way through which preprocessing can change the learning dynamics.

PRISM has made three main contributions to the extent that a scale-invariant representation of the preprocessing impact in a form understandable by a user is presented: (i) a scale-invariant representation of the preprocessing impact in a form comprehensible by a user, (ii) a framework based on tensors that allow comparative analysis across models and datasets, and (iii) a behavioral interpretation that positions preprocessing as a first-class analytical object. Together, these contributions offer a basis for future research into data-centric ML, robustness analysis, and automated pipeline optimization.

II. DATASETS DESCRIPTION

A. CLASSIFICATION DATASETS

1) BRAIN TUMOR DATASET

The brain tumor dataset is a biomedical dataset that is structured for a binary classification task of predicting the presence or absence of a tumor. It consists of both numerical and categorical diagnostic features. This dataset is based on a medical imaging application but is delivered in a tabular form, which is easily consumable by standard ML pipelines. The distribution of features in this dataset is relatively clean and the variations are moderate, which means that a good baseline performance can be achieved even without preprocessing. The main concern for distance metric-based models, such as KNN and SVM is that they are still susceptible to the differences of feature scaling. Per PRISM, this is a low-to-moderate complexity dataset where the effects of preprocessing/intervention can be well understood, are stable, and mostly involve feature scaling transformations.

2) LIVER DISEASE DATASET

The liver disease dataset comprises biochemical and physiological features related to liver disease. It is a binary classification dataset. Additionally, it is more challenging because apart from missing values, feature imbalance and overlapping class distributions make it even more difficult versus brain tumor data. Some values are missing, especially in A/G, that would cause irregularity that needs

imputation, greatly influencing the model behaviour. This dataset is very appropriate for the analysis of preprocessing due to the skewness of the data and data quality issues. The effects of preprocessing vary from model to model, indicating that preprocessing is not generally beneficial.

3) DIABETES DATASET

The diabetes dataset is a standard healthcare classification benchmark which includes features, such as glucose level, BMI, age, and insulin indicators. It has moderate class imbalance and implicit missing values in raw training, causing bias. Preprocessing is critical for good performance. The dataset is also non-linear, with relationships between features and target that are non-linear. This means that linear and non-linear models can be tested. For PRISM, it is a medium complexity dataset for which the preprocessing leads towards measurable performance variations across models, that are structured.

B. REGRESSION DATASETS

1) HEALTHCARE INVESTMENT DATASET

In this study, a regression dataset was constructed to model the outcomes of healthcare investments in terms of low-dimensional numerical features that had linear and mild non-linear relationships. It is very sensitive to preprocessing, such as scaling and normalization. Intrinsic feature relationships may be sometimes altered by the preprocessing in a linear model. It is important in PRISM as it demonstrates both positive and negative preprocessing effects depending on model type.

2) LENGTH OF STAY DATASET

This medical dataset is large scale and is used to predict the length of time a patient spends in the hospital based on the numerical, categorical, and temporal features. It is noisy, heterogeneous, and contains outliers, making preprocessing more dependent. The transformations (scaling, normalization) and outlier treatments have a significant impact on the stability and accuracy of the models. In PRISM, it is used to analyze robustness of the regression models.

3) MEDICINE PRICE DATASET

This dataset contains features that are highly scaled and a target distribution that is skewed, such that predicting the price of the drugs is difficult. This is very sensitive to preprocessing due to direct influence of scaling on RMSE and MAE. It is appropriate for scale-invariant behaviour in PRISM and demonstrates the significant impact of preprocessing on model performance interpretations.

C. DATASET SUMMARY AND EXPERIMENTAL JUSTIFICATION

The datasets chosen are a variety of classifications and regressions that span the healthcare and economic sectors, and exhibit a range in complexity, noise, and feature distribution. This diversity allows to assess the preprocessing in diverse conditions, not in a uniform one.

These datasets support meaningful PRISM-based preprocessing signatures over low, medium, and high complexity datasets and provide the opportunity to observe positive and negative preprocessing effects. Overall, this dataset collection presents a solid base for systematic analysis of the influence of preprocessing in ML.

D. DATA PREPROCESSING PIPELINE

The preprocessing pipeline is developed as a structured, controlled, and reproducible experimental setting to guarantee methodological uniformity between

different datasets. This allows for systematic investigation of preprocessing-induced behavioral changes within the PRISM paradigm. This study broke away from the traditional way of thinking of workflows as preparatory or preliminary steps. Moreover, it regarded preprocessing as a first order transformation operator by examining its impact explicitly as a comparison of preprocessed and raw data representations. To avoid inconsistencies in the sampling or experimental design, the performance of the pipeline is assessed on six real-world datasets, both classification and regression, and the differences observed are solely due to preprocessing (Figure 3).

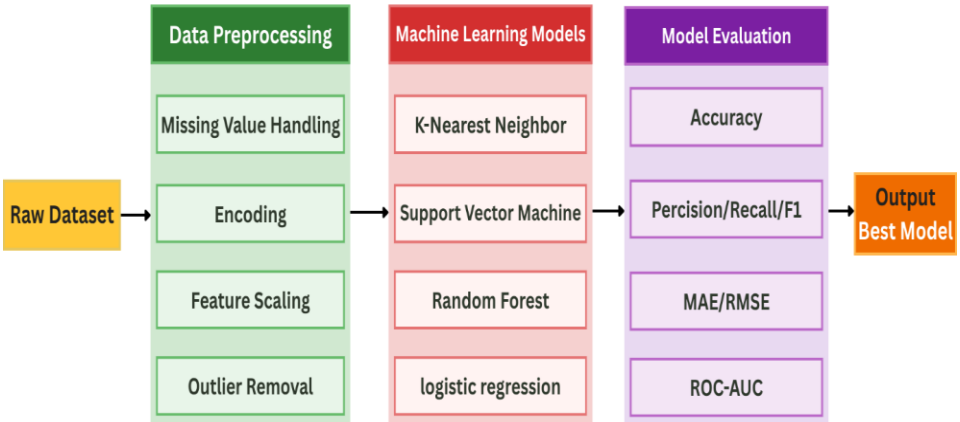


FIGURE 3. End-to-end ML pipeline highlighting preprocessing as a transformation layer influencing downstream model performance.

E. DATA CLEANING AND FEATURE REPRESENTATION

Data preprocessing includes data cleaning which makes the datasets structurally valid, statistically consistent, and analytically reliable prior to model training. Missing value handling is dataset specific: In the Liver Disease dataset, the values missing from the Albumin and Globulin Ratio feature are filled in using a backward fill strategy to ensure that the distribution of the feature is as close to local continuity as possible with minimum distortion. All other datasets are complete, with no confounding effects from use of different imputation methods. Structural checks are used to verify that there are no duplicate records, that the feature data types are correct, and that the consistency of training and testing partitions is not violated.

1) HYBRID APPROACH TO ENCODING CATEGORICAL VARIABLES

Label encoding is performed on ordinal variables, where the idea is to retain the inherent ranking hierarchy between them. One hot encoding is performed on nominal variables to avoid any unwarranted ordinal ordering assumptions on the variables and to prevent collinearity in the feature representation. This way, the models are compatible with each other, the semantics are well-preserved, and no artificial correlations are created that could affect learning.

F. FEATURE SCALING AND OUTLIER HANDLING

Feature scaling was performed using Z-score normalization, defined as $z = \frac{x-u}{\sigma}$, where μ and σ denote the mean and standard deviation of the feature, respectively. This ensures that distributions are centered, variance is one, optimization is stable, and distance-based models, such as KNN and SVM perform better. PRISM scales directly the size and orientation of PIVs.

Outlier values are removed from the data using a method of outlier detection named the Interquartile Range (IQR) method. This results in the removal of skewness, the regression becoming more stable, and the confidence in the classification boundaries being more trustworthy. The same outlier handling technique is applied universally to all datasets. This ensures that the preprocessing signatures are the result of the systematic transformations rather than the noise coming from individual datasets.

G. EXPERIMENTAL CONTROL AND DUAL-PIPELINE DESIGN

Across all datasets, the ratio of train-test split was fixed at 80/20. On the other hand, stratification was implemented for classification tasks and a controlled random seed was used to ensure reproducibility. Both raw and processed datasets were divided in the same manner so that any differences in performance could be attributed to the preprocessing transformations.

A dual-pipeline model was employed where the two datasets included: raw data (unmodified) and processed data (fully transformed). This not only made it possible to directly measure the performance differences caused by preprocessing but also helped in generating PRISM representations (PIV and PST) while keeping preprocessing as the main experimental variable. Preprocessing, in this expression, becomes a step that is observable as a behavior change in the preprocessing pipeline.

H. PRISM INTEGRATION

The preprocessing pipeline meets the analytical goals of PRISM, allowing for consistency, comparability, traceability, and reproducibility of all experiments. Preprocessing is formalized as a transformation operator depending on the context. The effect of the transformation is systematically quantified with the PIVs and aggregated with the PSTs. Table I presents an overview of the preprocessing operations applied to each dataset, which were also performed in a uniform manner, thus allowing for unbiased and structured analysis of preprocessing behavior.

TABLE I

DATASET-WISE VARIATION IN PREPROCESSING IMPACT SIGNATURES, REFLECTING STRUCTURAL COMPLEXITY AND PREPROCESSING DEPENDENCY UNDER PRISM FRAMEWORK

Dataset	Structural Complexity	Noise Level	Missingness	Preprocessing Impact Strength	Signature Stability	PRISM Interpretation
Brain Tumor	Low–Medium	Low	Minimal	High	High	Clean structure amplifies scaling sensitivity
Liver Disease	High	High	Present	Medium	Low	Noise + imbalance induce

Dataset	Structural Complexity	Noise Level	Missingness	Preprocessing Impact Strength	Signature Stability	PRISM Interpretation
Diabetes	Medium	Medium	Implicit missingness	Medium	Medium	unstable signatures Balanced preprocessing responsiveness
Length of Stay	High	Very High	Present	Very High	Medium	Strong preprocessing dependency due to heterogeneity
Medicine Price	Medium	Medium	None	Medium	Medium	Scale-driven but structurally stable dataset
Healthcare Investment	Low–Medium	Low	None	Low–Medium	High	Weak preprocessing dependency

I. EXPERIMENTAL SETUP

1) OVERVIEW OF EXPERIMENTAL DESIGN

The experimental design for this study is part of a controlled, multi-dimensional evaluation system that is systematically used to study the extent to which behavior of ML models is affected by preprocessing. This study evaluated each of these two datasets in two conditions: unprocessed (raw) and preprocessed. This design enables a direct and controlled measurement of the behavioral changes due to preprocessing and minimizing external variability.

The experimental setting is deliberately created to aid generalizability over a wide range of areas. It is used on six real-world datasets with classification and regression problems, where the areas of application are mixed with applications in areas, such as healthcare and economic forecasting. The processing of each dataset is done separately, to maintain the inherent distributional properties of data, however, still allowing cross-dataset comparative results using the PRISM analytical framework. This ensures that observations are unique to the dataset and have a universal interpretation.

All experiments follow a standard and repeatable workflow, including standard data loading, fixed train-test splits, standardized model training, and evaluation parameters. The only variable difference between the execution of the experiments is the use of preprocessing transformations. The separation of this type becomes important so that one can be confident that the differences in observed results are not due to the effects of the preprocessing but due to the experimental noise or configuration bias. As shown in Figure 4, both pipelines are trained on the same sets of data, using the same models, the same splits so, any variation in performance is due to the preprocessing transformations.

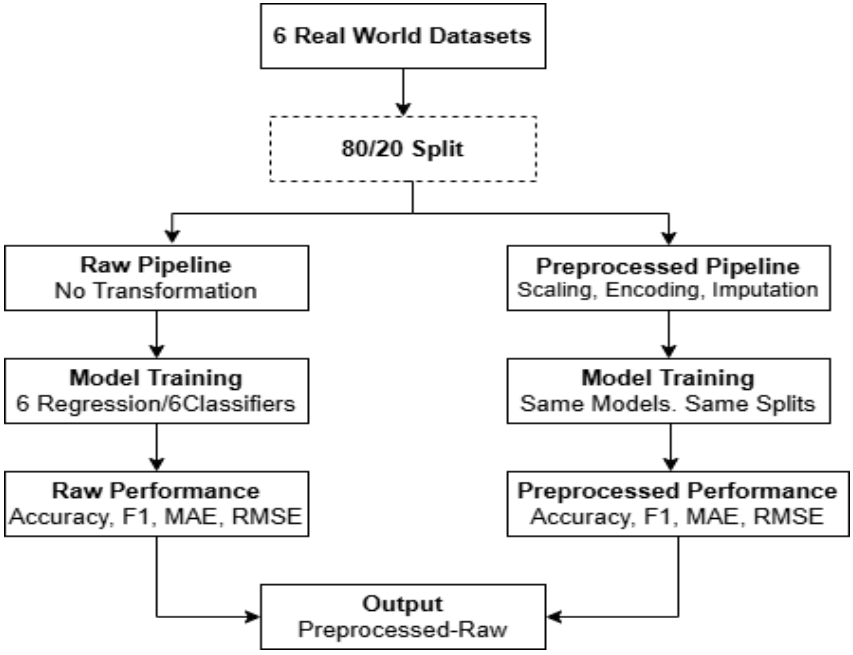


FIGURE 4. Dual pipeline experimental design showing parallel evaluation of raw and preprocessed data across identical model configurations, culminating in PIV computation.

2) MACHINE LEARNING (ML) MODELS CONFIGURATION

This is done by a variety of supervised learning models that try to capture a range of inductive biases and learning behaviors. The desired outcome is to ensure that the impact of preprocessing is not investigated separately for each algorithmic family rather tested with fundamentally different learning paradigms.

For classification problems, the six representative models are employed. These include Logistic Regression (LR) as a linear (probabilistic) model, Decision Tree (DT) as a hierarchical (rule-based) learner, Random Forest (RF), which is an ensemble bagging model, K-Nearest Neighbors (KNN), an instance-based (distance-based) model, Support Vector Machine (SVM), a margin-based kernel method, and Naïve Bayes (NB), a model based on independence-based probability. The difference in the sensitivity to scaling, noise, and feature distribution of the models is well-documented and therefore apt for analyzing the preprocessing behavior.

In regression activities, Multiple Linear Regression, Ridge Regression, Lasso Regression, Decision Tree (DT) Regressor, Random Forest Regressor, and Gradient Boosting Regressor were included in the study. This decision ensures that both parametric and non-parametric methods are included and a comprehensive study of the effects of preprocessing in continuous prediction models is carried out. Most importantly, the same default or poorly-tuned hyperparameters are used for training all the models, to avoid the problem of optimization bias and to keep preprocessing the primary experimental variable being studied. Table II shows that the profiles of

preprocessing sensitivity of different model families are distinct, with the underlying concept being the preprocessing sensitivity for behavioral analysis of the experimental results based on PRISM.

TABLE II
METRIC-WISE CONTRIBUTION TO PIVS ACROSS DIFFERENT MODEL FAMILIES

Metric Type	Sensitivity Contribution	Models Most Affected	PRISM Role
Accuracy / R^2	High	KNN, SVM, Linear Models	Global performance shift indicator
F1-Score / MAE	Medium	LR, NB	Class imbalance sensitivity signal
ROC-AUC	Very High (classification)	SVM, KNN	Decision boundary transformation indicator
RMSE	High (regression)	Linear + Gradient Models	Error sensitivity amplifier
MAE	Medium	All regression models	Stability indicator

3) DATA SPLITTING AND EVALUATION PROTOCOL

The same 80/20 train-test split is used for uniformity throughout all datasets. Random seeds are set for reproduction and statistical consistency. In classification problems, the classes are distributed in both training and test sets, and in regression problems, standard random sampling is used.

The raw pipeline and preprocessed pipeline are subject to the same conditions: data partitions, model configuration, and model initializations. This tight control removes the effect of sampling or stochastic process and any observed performance difference is caused by the effect of the preprocessing transformations.

4) PREPROCESSING PIPELINE CONFIGURATION

All datasets are preprocessed in the same way to guarantee methodological uniformity. The missing data is addressed only as needed, and statistically sound methods are used to avoid bias in imputation procedures. Label encoding is used for ordinal features, one-hot encoding is used for nominal features.

StandardScaler is important to normalize the numerical features, with mean=0 and variance=1, which is required for models based on distances and gradients. To improve the stability of the model, statistical techniques for outlier detection and outlier removal are used. The datasets are maintained in parallel and can be compared directly in the framework of PRISM.

5) EVALUATION METRICS AND PERFORMANCE MEASUREMENT

The evaluation strategy incorporates both predictive performance and the changes in behavior resulting from the preprocessing. Accuracy, Precision, Recall, F1-Score, and ROC-AUC are used as indicators for classification. To measure variance explanation and prediction error, R^2 Score along with RMSE and MAE are used in regression.

The metrics are calculated independently for raw and preprocessed pipelines. These differences are represented as PIVs that give a structured and normalized representation of the effects of preprocessing.

6) EXPERIMENTAL CONTROL AND FAIRNESS CONSIDERATIONS

Tight control procedures are in place to achieve experimental validity. The preprocessing steps, data splits, evaluation metrics, and random states do not change from one pipeline to another. No further preprocessing and hyper parameter tuning are added beyond what is already provided.

To avoid any data leakage, every transformation on the data is only done on the training data before evaluating on the test data. This prevents any changes in preprocessing from affecting the behavior of the model.

7) PRISM INTEGRATION IN EXPERIMENTAL WORKFLOW

The experimental pipeline is consistent with how each dataset-model pair was able to produce results in both raw and preprocessed conditions, using the PRISM framework. Such differences are represented as PIVs, which reflect the amplitude and direction of the preprocessing difference.

PIVs are summarized over datasets and models to create the PST and facilitate higher-level analysis of preprocessing behavior patterns.

The design is overall, controlled, and reproducible for analyzing the effect of preprocessing. This allows for the separation of the variable of interest, preprocessing, from other observed differences in order to determine the behavioral differences. This has the advantage that it allows both traditional evaluation and the PRISM framework, and that the preprocessing is not only a preparatory step but can also be viewed as a structured behavioral operator.

III. RESULTS AND ANALYSIS

A. CLASSIFICATION RESULTS

Brain tumor, liver disease, and diabetes datasets were used in classification experiments with six ML models in both raw and preprocessed datasets. The findings showed that the preprocessing has a significant impact on the performance of the model and different models have different performances on the same data.

For the brain tumor dataset, the near-optimal results were achieved after preprocessing. RF outperformed with the highest accuracy (0.9950), while LR (0.9899) and SVM (0.9866) followed. NB even improved to 0.9765. However, the performance of the raw data for sensitive models was very low: KNN went down from 0.9849 (processed) to 0.8194 (raw), and SVM from 0.9866 to 0.7888 (raw). Again, in raw form ROC-AUC for KNN decreased to 0.5000, indicating a loss of discriminative power because the features were not scaled.

Liver Disease: Moderate but steady improvements. RF gave the highest accuracy (0.7059) followed by LR (0.6863) and KNN (0.6275). Raw data revealed higher instability for NB, particularly in the ROC AUC, and irregular behavior for this. This dataset is more sensitive to noise and missing values, with the preprocessing primarily the result of stability, not accuracy.

The diabetes dataset showed a strong improvement in consistency as a result of preprocessing. SVM reached 0.7578 accuracy, LR 0.7266, and KNN 0.7344. In

the case of medical features, the KNN was improved from 0.3285 (raw) to 0.7495 (processed), which proves the importance of scaling for heterogeneous medical features.

B. REGRESSION RESULTS

Regression experiments were performed on healthcare investment, length of stay, and medicine price datasets using R^2, RMSE, and MAE. In healthcare investment, preprocessing increased the performance of the ensemble, while it decreased the performance of the linear assumptions. RF (0.9209) and DT (0.9286) exhibited good non-linear alignment with preprocessing, proving to be strong models. LR exhibited instability (R^2 = 0.7522 raw vs lower processed stability), while RF (0.9209) and DT (0.9286) performed well under preprocessing.

Preprocessing was found to improve the performance considerably for LS. This resulted in Gradient Boosting having an R^2 of 0.9743 and very low error (RMSE = 0.3371, MAE = 0.2695) compared to raw configurations. The same was observed with Ridge Regression, where the R2 values were more stable following the preprocessing.

The effects were mixed in medicine price. Ridge and Lasso models were not improved significantly with ~0.8408 R^2, while ensemble models produced moderate increase in R^2. Interestingly, the raw performance of the RF was competitive, suggesting that there is strong inherent structure in the dataset.

C. KEY OBSERVATIONS

In all datasets, the preprocessing mostly contributed to improve the stability of the results and did not always lead towards higher performance. Distance-based models (KNN, SVM) are the most sensitive models and achieve drastic improvements (e.g., SVM Diabet ROC-AUC: 0.3285 → 0.7495; KNN Brain Tumor: 0.8194 → 0.9849). The relatively stable systems are the tree-based models (DT, RF) with a slight improvement. The results of regression showed that the effectiveness of the preprocessing process is closely influenced by the structure of the datasets and the distribution of features.

These interactions between model and dataset are summarized in Table III by the PST across pipelines.

TABLE III
AGGREGATED PROJECTION OF THE PST SHOWING MODEL–DATASET INTERACTION PATTERNS

Dataset\Model Type	Distance-based (KNN/SVM)	Linear Models	Tree-based Models	Ensemble Models
Brain Tumor	High Signature	Medium	Low	Low
Liver Disease	Medium–High (unstable)	Medium	Low	Low–Medium
Diabetes	High	Medium	Low	Low
Length of Stay	Very High	Medium–High	Low	Medium
Medicine Price	Medium	Medium	Low	Medium
Healthcare Investment	Low–Medium	Low–Medium	Low	Low

D. SIGNATURE-BASED OBSERVATION (PRISM PERSPECTIVE)

Based on a PRISM-based interpretation, the effect of preprocessing on each dataset-model combination produces a distinct pattern of preprocessing effects. This confirms that the effects of preprocessing are structured patterns of effects but not uniform transformations.

Brain tumor dataset provides an example where positive impact signature of KNN and SVM of high magnitude is generated, pointing to the heavy reliance of the scaling transformations. Conversely, DT and RF generate low-variance signatures, which are an indication of preprocessing invariance.

Signatures in liver dataset are smaller but more variable. This indicates that missing data and imbalance of features bring instability, which is partially addressed but not fully addressed by preprocessing.

The diabetes dataset generates balanced yet always positive signatures in most models. This indicates that preprocessing improves separability in moderately structured data distributions.

In regression tasks, length of stay data has high-amplitude positive signatures, particularly with ensemble models, and medicine price data has mixed-signature behaviour, where some models perform better and others remain steady or slightly degraded.

Overall, these patterns empirically validate the PRISM hypothesis: preprocessing impact is not uniform but exists as a structured, model–dataset dependent behavioral signature.

IV. DISCUSSION

A. REFRAMING PREPROCESSING AS BEHAVIORAL TRANSFORMATION

The experimental results clearly support the main objective of this research. The main objective was that preprocessing is not a passive and auxiliary process rather a behavioral transformation operator, which affects the interaction between models and data. For classification and regression, the preprocessing always resulted in changes in the model behavior, which were not uniform rather were structured. Conceptually this is similar to the systems-thinking approach proposed by Barouki et al. [1] who considered that complex scenarios result from interactions between multiple factors. Similarly, PRISM shows that preprocessing effects emerge from the interplay of the structure of the data, the inductive bias of the model, and the type of transformation in the data. Data from the PIV and PST additionally indicates that these "improvements" are not improvements in general, instead changes in model behavior in specific directions for specific metrics. This is a distinct change from the standard way of evaluating the effect of preprocessing, which usually involves number of errors or accuracy changes.

B. MODEL SENSITIVITY AND STRUCTURAL DEPENDENCE

One important conclusion one can draw from the results is that there is a distinct layering of the sensitivity to the preprocessing. Distance-based models, such as KNN and SVM showed high amplitude PIVs, particularly for the datasets having heterogeneous feature scales including diabetes and brain tumor. Conversely, the tree models (RF, DT) were relatively stable to monotonic transformations. Hammoudeh et al. [2]

pointed out that data influence is not the same for all models but depends on the learning mechanism and the representation sensitivity. PRISM takes this concept one step further, however, to focus not on the individual data points but on the greater effects of global transformation on the model under the preprocessing. Regression experiments demonstrated that in the high noise environment (such as Length of Stay), it is most beneficial to use ensemble methods when preprocessing. Whereas, in some cases, the linear models were found to be unstable when transforming aggressively. This indicates that there is no universal benefit to preprocessing in the sense that it helps to improve performance when it does not actually. This means that not all preprocessing is beneficial in terms of learning but it shifts the learning trade-off between variance and bias.

C. DATASET COMPLEXITY AND PREPROCESSING BEHAVIOR

The study also clearly showed that the magnitude of the impact of the preprocessing depends on the dataset structure. Highly predictable and stable preprocessing signatures were observed for low complexity datasets (Brain Tumor). Whereas, unstable or inconsistent preprocessing signatures were observed for high complexity datasets (Liver Disease, Length of Stay). This is in line with Koukaras et al. [3], who stated that the effectiveness of preprocessing methods varies from dataset to dataset. However, PRISM goes beyond this observation by quantifying dataset dependence using structured signature representations instead of descriptive taxonomy. Specifically, datasets containing missingness and noise (liver disease) yielded poor signature stability. This suggests that preprocessing is not enough to fully stabilize the learning behavior in very noisy environments. One important limitation is that the uncertainty of the data may not be completely removed through preprocessing.

D. LIMITATIONS OF CURRENT PREPROCESSING PARADIGMS

The study, however, showed that the current preprocessing paradigms have some major drawbacks. The first is that most of the traditional methods regard the preprocessing as a static operation in the pipeline instead of a dynamic transformation process. As Fernandes et al. [4] pointed out, data preparation is a key component of the ML workflow but it is not well-formalized regarding its analytical contribution. This need was addressed here with the PRISM, which is formulated currently, however, it still has empirical estimates, and not full theoretical guarantees. Secondly, influence-based approaches, such as Hammoudeh et al. [2], are instance-level approaches but PRISM is transformation-level approach. This abstraction comes with its own drawback, however, which is a lack of granularity in interpretation on a feature or sample-by-sample basis. This is something that has to be addressed in future research by combining instance-level and transformation-level modeling. Thirdly, there are preprocessing optimization frameworks based on the automaton, such as Preptimize [9] or DIANA [10], which optimize the selection of preprocessing steps but do not explain how a preprocessing step affects model behavior. There is a trade-off between explanation and automation in PRISM as it does not provide automated decision-making yet provides interpretation.

E. CHALLENGES IN PREPROCESSING IMPACT MODELING

The current study comes with a number of

methodological and conceptual difficulties. One of the significant challenges is the non-linearity of the effects of preprocessing. The impact of preprocessing varies across model families and datasets, making it challenging to establish universally applicable preprocessing rules. The second problem is the instability of metric dependence. While PIV normalization minimizes scale bias, evaluation metrics, such as accuracy, F1, RMSE, and AUC yield varying patterns of sensitivity. This makes it difficult to sum up the preprocessing signatures into a single understandable value. The third challenge is the statistical variance in the preprocessing step of the impact estimation. However, these effects of preprocessing may vary with repeated runs and variance modelling due to the stochastic nature of learning dynamics, particularly in ensemble and gradient-based models. This adds noise to the stability of the signature, as in noisy datasets, such as liver disease. Lastly, the computational burden can be substantial when building PSTs over several models and datasets, especially for large scale applications. This is an obstacle to scalability, unless efficient approximation strategies are added.

F. FUTURE WORK DIRECTIONS

This research showed some areas of potential research.

As for further research, probabilistic PRISM modeling should be investigated, in which the PIVs are considered as probability distributions, not as vectors. This would better reflect uncertainty in the preprocessing effects and be representative of real-world training uncertainty.

Secondly, integration with federated and distributed learning systems is a natural extension. However, as pointed out by Shuang Dai et al. [14], the non-IID data substantially impacts the model convergences. PRISM could be extended to support distributed nodes and calculate the impact of preprocessing on each one of them, which would allow for cross-institutional preprocessing analysis.

Thirdly, hybrid systems combining PRISM and the automatic selection systems for preprocessing (such as Preptimize and DIANA) might result in interpretable and adaptive systems. This would enable systems to not only choose a preprocessing pipeline but also explain the effect this pipeline would have on the system.

Finally, future research should focus on a feature-level decomposition of PIVs, thus providing fine-grained attribution of effects of preprocessing to individual features. This would help to close the gap between transformation-level and instance-level interpretability.

Lastly, if PRISM could be extended to DL and unstructured data domains (images, text and graphs), its usefulness would be greatly enhanced. The current results are for tabular data sets but with high-dimensional representations, the signature behaviors may be completely different when preprocessing is applied to the data.

Overall, this study demonstrated that preprocessing must be reinterpreted as a structured, measurable, and model-dependent transformation process. While PRISM successfully formalizes this perspective, it also exposes deeper complexities in modeling data transformations across heterogeneous systems. The findings suggest that future ML research should move beyond performance-centric evaluation and adopt behavior-centric frameworks that explicitly model how data transformations reshape learning dynamics.

By situating preprocessing as a first-class analytical object, this work contributed toward a more principled, interpretable, and data-centric foundation for modern ML systems.

G. CONCLUSION

This study proposed PRISM, a comprehensive structure for viewing data preprocessing as a well-organized, quantifiable, and ML model-dependent transformation process. Other than the usual way of judging preprocessing from a single perspective of the improvement in performance, PRISM brings in a behavioral viewpoint that measures the change of model behavior due to preprocessing through the PIV and accumulates those changes through the PST. This representation gives a methodical, scale-free, and comparative study of the effect of preprocessing across different datasets, models, and evaluation metrics.

Evaluation on six real-world classification and regression datasets confirms that preprocessing does not result in the same level of improvements. However, it leads towards behavioral changes that have a certain pattern and are dependent on dataset complexity and model inductive bias. Models relying on distance measures, for instance KNN and SVM, feature changes due to preprocessing. In particular, unlike many other model families, tree-based models are generally less affected by feature scaling. Besides, as per the study, preprocessing improved the reliability of models more frequently than increasing their performance, in particular when dealing with noisy and varied datasets.

PRISM changes the role of preprocessing from an afterthought pipeline step to a major analytical element of ML. Through the formalization of preprocessing effects in the form of behavioral signatures, the study laid a basis for the development of more interpretable and data-centric ML systems. Future research would be directed at extending PRISM to probabilistic modeling, DL, and automated preprocessing selection setups.

Author Contribution

Abu Bakar Shabbir: conceptualization, methodology, investigation, formal analysis, data curation, visualization, writing – original draft, writing – review & editing. **Khadija Amber:** conceptualization, methodology, validation, investigation, formal analysis, visualization, writing – original draft, writing – review & editing.

Conflict of Interest

The authors of the manuscript have no financial or non-financial conflict of interest in the subject matter or materials discussed in this manuscript.

Data Availability Statement

Data supporting the findings of this study will be made available by the corresponding author upon request.

Funding Details

No funding has been received for this research.

Generative AI Disclosure Statement

The authors did not use any type of generative artificial intelligence software for this research.

REFERENCES

[1] R. Barouki *et al.*, "The COVID-19 pandemic and global environmental change: Emerging research needs," *Environ. Int.*, vol. 146, art. no. 106272, Jan. 2021, https://doi.org/10.1016/j.envint.2020.106272.

[2] Z. Hammoudeh and D. Lowd, "Training data influence analysis and estimation: A survey," *Mach. Learn.*, vol. 113, pp. 2351–2403, March 2024, https://doi.org/10.1007/s10994-023-06495-7.

[3] P. Koukaras and C. Tjortjis, "Data preprocessing and feature engineering

for data mining: Techniques, tools, and best practices," *AI*, vol. 6, no. 10, art. no. 257, Oct. 2025, https://doi.org/10.3390/ai6100257.

[4] A. A. A. Fernandes, M. Koehler, N. Konstantinou, P. Pankin, N. W. Paton, and R. Sakellariou, "Data preparation: A technological perspective and review," *SN Comput. Sci.*, vol. 4, art. no. 425, Jun. 2023, https://doi.org/10.1007/s42979-023-01828-8.

[5] T. Emmanuel, T. Maupong, D. Mpoeleng, T. Semong, B. Mphago, and O. Tabona, "A survey on missing data in machine learning," *J. Big Data*, vol. 8, art. no. 140, Oct. 2021, https://doi.org/10.1186/s40537-021-00516-9.

[6] W. M. Hameed and N. A. Ali, "Missing value imputation techniques: A survey," *UHD J. Sci. Technol.*, vol. 7, no. 1, pp. 72–81, Mar. 2023, https://doi.org/10.21928/uhdjst.v7n1y2023.pp72-81.

[7] A. D. Călin, A. M. Coroiu, and H. B. Mureşan, "Analysis of preprocessing techniques for missing data in the prediction of sunflower yield in response to the effects of climate change," *Appl. Sci.*, vol. 13, no. 13, art. no. 7415, Jun. 2023, https://doi.org/10.3390/app13137415.

[8] V. G. Biju, A.-M. Schmitt, and B. Engelmann, "Assessing the influence of sensor-induced noise on machine-learning-based changeover detection in CNC machines," *Sensors*, vol. 24, no. 2, art. no. 330, Jan. 2024, https://doi.org/10.3390/s24020330.

[9] M. Usmani, Z. A. Memon, A. Zulfiqar, and R. Qureshi, "Preptimize: Automation of time series data preprocessing and forecasting," *Algorithms*, vol. 17, no. 8, art. no. 332, Aug. 2024, https://doi.org/10.3390/a17080332.

[10] C. Sancricca, G. Siracusa, and C. Cappiello, "Enhancing data preparation: Insights from a time series case study," *J. Intell. Inf. Syst.*, vol. 62, pp. 1503–1530, Dec. 2024, https://doi.org/10.1007/s10844-024-00867-8.

[11] K. Berahmand, F. Daneshfar, E. S. Salehi, Y. Li, and Y. Xu, "Autoencoders and their applications in machine learning: A survey," *Artif. Intell. Rev.*, vol. 57, no. 2, art. no. 28, 2024, https://doi.org/10.1007/s10462-023-10662-6.

[12] I. D. Lopez-Miguel, "Survey on preprocessing techniques for big data projects," in *Proc. 4th XoveTIC Conf.*, A Coruña, Spain, 2021, vol. 7, no. 1, art. no. 14, https://doi.org/10.3390/engproc2021007014.

[13] P. Kamencay, P. Hockicko, and R. Hudec, "Sensors data processing using machine learning," *Sensors*, vol. 24, no. 5, art. no. 1694, Mar. 2024, https://doi.org/10.3390/s24051694.

[14] S. Dai and F. Meng, "Addressing modern and practical challenges in machine learning: A survey of online federated and transfer learning," *Appl. Intell.*, vol. 53, pp. 11045–11072, 2023, https://doi.org/10.1007/s10489-022-04065-3.

[15] H.-J. Park, Y.-S. Koo, H.-Y. Yang, Y.-S. Han, and C.-S. Nam, "Study on data preprocessing for machine learning based on semiconductor manufacturing processes," *Sensors*, vol. 24, no. 17, art. no. 5461, Aug. 2024, https://doi.org/10.3390/s24175461.

[16] J. N. M. Dahj and K. A. Ogudo, "Machine learning-based imputation approach with dynamic feature extraction for wireless RAN performance data preprocessing," *Symmetry*, vol. 15, no. 6, art. no. 1161, May 2023, https://doi.org/10.3390/sym15061161.