

Frozen LLM-Generated Preprocessing for Medical-Claims Adherence Classification: An Internal Model-Sensitivity Study

Herman Yuliansyah¹, Ihya' Nashirudin Abrar², and Muhammad Kunta Biddinika²

¹Department of Informatics, Faculty of Industrial Technology, Universitas Ahmad Dahlan, Yogyakarta, Indonesia

²Master Program of Informatics, Faculty of Industrial Technology, Universitas Ahmad Dahlan, Yogyakarta, Indonesia

Corresponding author: Herman Yuliansyah (e-mail: herman.yuliansyah@uad.ac.id), **Author(s) Email:** Ihya' Nashirudin Abrar (e-mail: Ihyakpati1144@gmail.com), Muhammad Kunta Biddinika (e-mail: mkuntab@uad.ac.id)

Abstract Large language models (LLMs) can generate data-preprocessing code from data profiles, but their value relative to predefined expert workflows in clinical tabular classification remains unclear. This study evaluated whether frozen LLM-generated preprocessing matches a Manual Expert workflow for medication-adherence classification and identified the components responsible for performance differences. We analyzed 24,071 de-duplicated medical-claims records from 2022, comparing Manual Expert preprocessing against frozen code generated by DeepSeek, Claude Sonnet 4.6, GPT-5.4, and iterative CAAFE. Stateful transformations were fitted exclusively within training folds. Performance was assessed using repeated 5 × 10-fold cross-validation, a locked 20% holdout with 2,000 bootstrap samples, and Nadeau–Bengio-corrected equivalence testing. With LightGBM, all pipelines were equivalent in accuracy, Matthews correlation coefficient (MCC), and area under the receiver operating characteristic curve (AUC). Cross-validation accuracy ranged from 0.8204 to 0.8229, and every corrected 90% confidence interval remained within the prespecified ±0.01 equivalence margin; the widest interval extended from −0.0041 to 0.0009. Component ablation attributed the entire improvement over raw data to ratio features (+0.0018). Log1p transformations produced no improvement (+0.0000), whereas scaling slightly reduced accuracy (−0.0007), consistent with tree-based models' insensitivity to monotonic transformations. Excluding the outcome-proximal variable UNITSTOTAL reduced holdout accuracy from 0.820 to 0.758 and AUC from 0.903 to 0.839, although pipeline equivalence persisted (accuracy span: 0.0018). However, equivalence depended on the estimator: naive Logistic Regression exhibited an accuracy span of 0.202, decreasing to 0.161 after uniform scaling and nested regularization tuning, with differences concentrated in CAAFE. Thus, frozen LLM-generated preprocessing can match expert-designed preprocessing for internal, tree-based medication-adherence classification. Gains primarily reflect feature construction rather than distributional adjustment. These findings also caution against assuming universal preprocessing superiority. Preprocessing strategies should therefore be evaluated jointly with downstream estimators, while claims-based prediction results require careful assessment of outcome-proximal information before being interpreted as genuine predictive performance.

Keywords Large language models; Automated data preprocessing; Medication adherence; Clinical tabular data; Statistical equivalence

1. Introduction

Non-communicable diseases (NCDs), including type 2 diabetes and cardiovascular conditions, account for a dominant share of global mortality and place a disproportionate burden on health systems in low- and middle-income countries [1]. Because these conditions require long-term pharmacological management, outcomes depend heavily on medication adherence, defined as the degree to which patients follow prescribed regimens [2], [3]. In practice, adherence is difficult to sustain, particularly in resource-constrained settings [4]. Persistent non-adherence accelerates disease progression and drives avoidable complications,

hospitalizations, and treatment costs [4]. The growing availability of medical-claims and insurance records has therefore made machine learning (ML) a practical tool for identifying adherence-related patterns [5], [6], [7], [8]. ML-based models have been studied for non-adherence in type 2 diabetes [9], [10], treatment-default classification [11], cardiovascular-risk and event prediction [12], [13], and clinical decision support [14]. In the present cross-sectional setting, however, classification performance should be interpreted as internal separability rather than early or prospective risk prediction. Prior studies on medication-adherence prediction have primarily evaluated conventional machine-learning pipelines with manually specified preprocessing. A systematic review of

25 studies identified Random Forest, Logistic Regression, and Support Vector Machine as the most frequently used algorithms [15]. On the Zimbabwean medical-claims dataset used here, Kanyongo et al. [16] established a benchmark in which Random Forest achieved 88.2% accuracy and an area under the receiver operating characteristic curve (AUC) of 0.935, and released the curated dataset [17]. Subsequent studies applied ensemble and deep-learning models to medication-adherence prediction [18] and boosting with engineered clinical features to refill adherence in type 2 diabetes [19]. Our earlier study on the same Zimbabwean data compared XGBoost, LightGBM, and CatBoost under one fixed manual feature-engineering workflow [20]. Collectively, these studies demonstrate the predictive value of machine learning for adherence assessment, but they specified preprocessing manually and did not compare it with code generated by large language models under an otherwise identical evaluation protocol.

Parallel work has explored LLM-assisted tabular preparation outside medication adherence. CAAFE (Context-Aware Automated Feature Engineering) generates and validates candidate features on general tabular benchmarks [21], whereas Jellyfish instruction-tunes local LLMs for imputation, error detection, and schema matching and reports performance approaching proprietary LLMs on data-preprocessing tasks [22]. Given that preprocessing quality affects downstream performance and remains labor-intensive [23], [24], two research gaps remain. First, no leakage-controlled, head-to-head benchmark compares frozen LLM-generated preprocessing against a predefined expert workflow for medication-adherence prediction. Second, prior studies have rarely isolated, through component ablation, which transformations account for any observed gain. These gaps leave unresolved whether LLM-generated preprocessing offers measurable value beyond an expert-defined workflow when the data, downstream model, folds, and evaluation protocol are held constant.

The novelty of this study is therefore in the comparison design rather than in performance. We evaluate CAAFE here as a measured participant under a common protocol rather than adopting it as the framework, and it contributed a single conjunction feature; Jellyfish addresses data cleaning rather than the statistical transformation and feature construction studied here. The frozen setting is the central design choice: each pipeline was generated once from a textual training-data profile and executed as written, with no tuning against any result, which removes the iterative repair path that can quietly absorb downstream information. The cost of that choice is acknowledged, in that the results represent a lower bound on what these models could achieve with feedback. Finally, prior adherence studies do not report fold-wise restriction of transformation fitting, and without that control an equivalence claim cannot be distinguished from shared leakage.

It is useful to distinguish three related but non-equivalent notions that LLMs touch in tabular data preparation. Automated feature engineering searches a space of candidate features and validates them against a downstream metric, as in CAAFE. LLM code generation

instead asks a model to emit a complete, executable preprocessing function from a textual data profile in a single pass. Statistical preprocessing refers to the individual transformation and scaling decisions skewness correction, ratio construction, standardization that such code contains. This study does not evaluate LLM reasoning quality or generated-code quality in isolation; it evaluates the utility of the resulting frozen pipeline as a whole, under a fixed downstream model, identical folds, and one evaluation protocol, so that the only variable is the source of the preprocessing. To address these gaps, we propose a leakage-controlled benchmarking framework that compares one predefined Manual Expert pipeline with frozen preprocessing code produced by DeepSeek, Claude Sonnet 4.6, GPT-5.4, and iterative CAAFE. All pipelines are evaluated on identical stratified folds and the same locked holdout, while each stateful transformation is fitted only on the corresponding training partition to prevent leakage [14], [25], [26]. Component ablation and model-class sensitivity analyses are used to separate the value of individual transformations from the influence of the downstream estimator. Accordingly, this study aims to determine whether frozen LLM-generated preprocessing can match a predefined Manual Expert workflow for medication-adherence classification under identical internal-validation conditions. The scope is deliberately limited to internal record-level validation; no claim of patient-level, temporal, external, or clinical deployment validity is made.

The contributions are fourfold: (1) a controlled comparison of one expert-defined pipeline, three one-shot LLM outputs, and iterative CAAFE; (2) leakage-controlled fold-wise evaluation with a locked holdout; (3) ablation of preprocessing components; and (4) analysis of model-class sensitivity, calibration, subgroup behavior, prediction overlap, and proxy-feature dependence. The remainder of this paper is organized as follows. Section II presents the dataset, five preprocessing pipelines, LLM generation process, model training, statistical framework, and reproducibility controls. Section III reports comparative performance, complexity, ablation, repeated validation, model-class sensitivity, SHAP, calibration, and error analysis. Section IV interprets the findings, relates them to previous research, discusses limitations, and outlines future work. Section V concludes the study.

II. Materials and Method

Fig. 1 illustrates the overall research flow; Fig. 2 details the five-stage LLM preprocessing. All components use `random_state = 42`. Algorithm 1 outlines the full computational procedure, from generating each frozen pipeline through fold-wise fitting, model training, and the final corrected equivalence decision.

A. Dataset

The dataset originates from Cimas Medical Aid Society, Zimbabwe [16], [17], [27]. The source data paper describes anonymized record-level and claims data collected from 1 January to 31 December 2022 and states that recurring transactions were aggregated by BENEFICIARY before retaining one annual row per beneficiary. Adherence was defined using the proportion

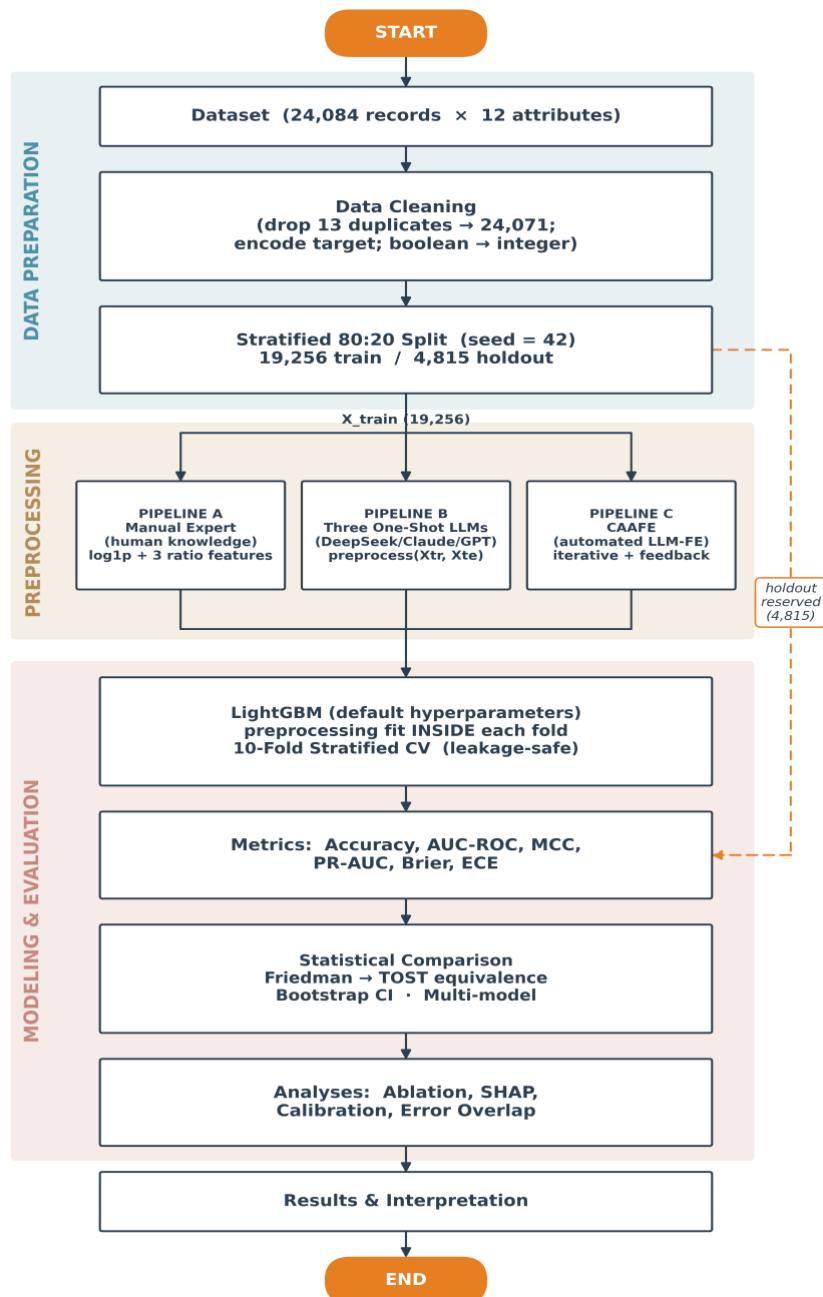


Fig. 1. Overall research workflow for comparing the five preprocessing pipelines under leakage-controlled model evaluation.

of days covered (PDC): at least 9 of 12 expected monthly medication refills (75%); fewer than 9 were classified as non-adherent. The supplied analytic workbook contains 24,084 rows and 11 predictors; removing 13 exact duplicate rows leaves 24,071 observations (40.2% adherent), split into 19,256 training and 4,815 locked holdout observations (seed=42). Table 1 describes the eleven predictors and the binary adherence outcome. Source documentation reports beneficiary-level aggregation and de-duplication, but BENEFICIARY and transaction dates are absent from the released workbook, so that aggregation cannot be independently verified; the present study therefore treats each row as an annual record and validates only at record level, and annual predictors and the outcome belong to the same measurement year. We therefore analyze internal cross-

sectional classification rather than prospective prediction. Adherent records number 9,669 overall (40.2%), 7,735 in the training pool, and 1,934 in the locked holdout, so class balance is preserved across partitions. Table 2 reports the descriptive statistics of all eleven predictors, continuous and binary.

B. Preprocessing: Manual Expert, Three One-Shot LLMs, and CAAFE

Table 3 summarizes the five core preprocessing pipelines: one predefined Manual Expert workflow, three frozen one-shot LLM functions, and iterative CAAFE. Manual Expert adds log1p transforms and three domain-motivated ratio features without scaling. In every cross-validation fold, stateful transformations are fitted only on the fold-training partition and then applied to the fold-

Table 1. Description of predictor variables and the binary medication-adherence outcome.

Variable	Type	Description
ADHERENCE	Binary (target)	1 = ADHERENT (PDC $\geq$ 75%); 0 = NON-ADHERENT
AGE	Continuous	16 to 103 years
ANNUALCONTRIBUTION	Continuous	Annual premium (ZWL)
ANNUALCLAIMAMOUNT	Continuous	Annual claims (ZWL)
UNITSTOTAL	Continuous	Units dispensed/year
GENDER_M	Binary	Male (1) / Female (0)
SCHEMETYPE_MEDIUM	Binary	Medium insurance
SCHEMETYPE_PREMIUM	Binary	Premium insurance
DIAGNOSIS_HYPERTENSION	Binary	Hypertension vs Diabetes
COVERTYPE_STANDARD	Binary	Standard cover
COMORBIDITY_NO_COMORBIDITY	Binary	No comorbidity
COMPLICATIONDEVELOPMENT_NO_COMPLICATION	Binary	No complication

Table 2. Descriptive statistics of the eleven predictor variables in the complete analysis dataset.

Predictor	Type	Median / n	IQR / %	Range	Skew	Outliers	Miss.
AGE	Cont.	56.0	46.0–66.0	16.0–103.0	0.16	13	0
ANNUALCONTRIBUTION	Cont.	6,889,900.0	2,718,696.0 – 6,889,900.0	211,764.0–11,770,788.0	-0.67	0	0
ANNUALCLAIMAMOUNT	Cont.	40,482.0	12,337.3–102,120.1	1.0–9,401,312,187.4	150.35	2,447	0
UNITSTOTAL	Cont.	345.0	120.0–720.0	1.0–21,290,640.0	102.79	1,556	0
GENDER_M	Bin.	13,063	54.3%	0–1			0
SCHEMETYPE_MEDIUM	Bin.	5,146	21.4%	0–1			0
SCHEMETYPE_PREMIUM	Bin.	17,882	74.3%	0–1			0
DIAGNOSIS_HYPERTENSION	Bin.	19,266	80.0%	0–1			0
COVERTYPE_STANDARD	Bin.	19,261	80.0%	0–1			0
COMORBIDITY_NO_COMORBIDITY	Bin.	17,096	71.0%	0–1			0
COMPLICATIONDEVELOPMENT_NO_COMPLICATION	Bin.	21,760	90.4%	0–1			0

Note: Analysis set: 24,071 records after removing 13 exact duplicates. Cont. = continuous; Bin. = binary; IQR = interquartile range; Outliers = observations beyond $1.5 \times$ IQR; Miss. = missing values. Continuous predictors are summarized by median, IQR, full range, skewness, and outlier count; binary predictors are summarized by n and percentage coded 1.

validation partition. The comparison therefore remains focused on manual-expert versus LLM-generated preprocessing. Table 4 lists the complete set of transformations and engineered features produced by each pipeline, together with the fitted quantities that are learned on the fold-training partition. Each one-shot LLM received the same training-data profile and system prompt through OpenRouter at temperature 0.0. The requested interface was preprocess(X_{train} ,

X_{test} =None), requiring fitted state to be learned from X_{train} and reused for X_{test} . The returned code was frozen before evaluation; no live API call occurs in the canonical experiment. Exact hosted-model regeneration is not assumed because provider routing and model snapshots can change. CAAFE was run separately as an iterative comparator, and its accepted feature code was likewise frozen.

Table 3. Characteristics and design differences of the five evaluated preprocessing pipelines.

Pipeline	Source and model	Design origin	Generations	Validation feedback	Key operations	Features
Manual Expert	Human domain knowledge	Predefined expert workflow (inherited)	n/a	No	log1p(CLAIM, CONTRIB) + 3 ratios	17
DeepSeek	OpenRouter: deepseek-chat-v3.1	Single one-shot LLM generation	1	No	2 ratios + Yeo-Johnson + StandardScaler	13
Claude Sonnet 4.6	OpenRouter: claude-sonnet-4.6	Single one-shot LLM generation	1	No	log/sqrt + 2 ratios + interactions + RobustScaler	22
GPT-5.4	OpenRouter: gpt-5.4	Single one-shot LLM generation	1	No	ratios + many interactions + StandardScaler	34
CAAFE	Iterative, DeepSeek backend	Iterative LLM feature engineering	5 iterations	Yes (CV feedback)	iterative feature generation + CV verification	12

Table 4. Transformations, engineered features, and fold-wise fitted components of the five preprocessing pipelines.

Pipeline (final features)	Transformations and engineered features (fitted on the training fold only)
Manual Expert (17)	log1p(CLAIM), log1p(UNITS), log1p(CONTRIB); CLAIM_PER_UNIT = log1p(CLAIM/(UNITS+1)); CLAIM_RATIO = log1p(CLAIM/(CONTRIB+1)); UNITS_PER_AGE = UNITS/(AGE+1). No scaling.
DeepSeek (13)	CONTRIBUTION_CLAIM_RATIO = CONTRIB/(CLAIM+1); UNITS_PER_CLAIM = UNITS/(CLAIM+1); Yeo-Johnson PowerTransformer on the four continuous predictors and two ratios; StandardScaler.
Claude Sonnet 4.6 (22)	CLAIM_LOG = log1p(max(CLAIM - min_train, 0)); UNITS_LOG likewise; CONTRIB_SQRT = sqrt(CONTRIB - min + ε); CLAIM_TO_CONTRIBUTION = CLAIM_LOG/(CONTRIB_SQRT+ε); CLAIM_PER_UNIT = CLAIM_LOG/(UNITS_LOG+ε); AGE×COMORBIDITY; AGE×NO_COMPLICATION; PREMIUM×AGE; HYPERTENSION×COMORBIDITY; UNITS_LOG×COMORBIDITY; AGE². RobustScaler on 14 continuous and engineered columns.
GPT-5.4 (34)	CONTRIB_reflect_log1p; CLAIM_log1p; UNITS_log1p; CLAIM_TO_CONTRIBUTION_RATIO and its log1p; CLAIM_PER_UNIT and its log1p; CONTRIBUTION_PER_UNIT and its log1p; AGE×{comorbidity, hypertension, complication}; {premium, medium, standard, comorbidity, hypertension, complication}×claimratio_log; AGE². StandardScaler on all non-binary columns.
CAAFE (12)	The 11 raw predictors plus one verified feature accepted after five cross-validation-checked iterations: SIMPLE_CASE = 1 if (no comorbidity and no complication) else 0. No scaling.

C. How Large Language Models Perform Data Preprocessing

Unlike traditional programmatic preprocessing, where a human writes explicit transformation rules, LLM-augmented preprocessing can generate and validate transformation code from dataset context [21]. Understanding this mechanism is essential for interpreting why LLMs succeed or fail at specific preprocessing decisions. Fig. 2 illustrates the complete flow.

Stage 1 Data Profiling: Before the LLM can reason about preprocessing, the raw dataset must be translated into a structured textual summary that the LLM can process [28]. We compute the following descriptive

statistics for every feature: data type (continuous, binary, categorical), number of unique values, count of missing values, skewness coefficient, interquartile range (IQR), outlier count (observations beyond $1.5 \times \text{IQR}$), and Pearson correlation with the target variable. The profiling step also reports global dataset properties: total rows, total features, target distribution (class balance ratio), and the number of duplicate rows removed. This profile typically spans 30–50 lines of text. For example, the profile for ANNUALCLAIMAMOUNT reads: 'dtype=float64, missing=0, unique=18030, skew=+111.25, outliers=1959, target_corr=+0.1876.'

Stage 2 LLM Statistical Diagnosis: The structured profile is supplied with a fixed system prompt [29], [30].

The prompt requests statistical diagnosis and a two-input preprocessing function that fits any stateful transformer on training data only. The profile contains aggregate

that fails the required two-input contract is rejected rather than repaired or evaluated with independently fitted transforms.

The five-stage design creates an intentional asymmetry between the comparators. The Manual Expert workflow was specified before the present comparison from published prior work [20] and uses a compact set of log and ratio features without scaling. Each one-shot LLM instead receives a textual training-data profile and produces one frozen pipeline without performance feedback; CAAFE provides an iterative comparator with validation feedback. The experiment therefore compares one predefined human-authored workflow with one archived generation from each one-shot model. It does not estimate the full variability of either human or LLM preprocessing.

The transformations evaluated across the five pipelines are defined in Eqs. (1)–(10) [31], [32]. In these equations, x denotes the raw value of a continuous predictor and z its transformed value; a and b are two continuous predictors combined into a ratio; b in Eq. (6) is a binary category indicator; c_1 and c_2 are binary indicators of the absence of comorbidity and of complications; and $\varepsilon = 1$ is a stabilizing constant that prevents division by zero, chosen on the counting scale of the denominator. All fitted quantities carry the subscript *train*, denoting that they are estimated on the training partition of the current fold only and applied unchanged to the validation partition: the shift s , the minimum and maximum, the mean μ and standard deviation σ , the median and the first and third quartiles, and the Yeo–Johnson parameter λ estimated by maximum likelihood [33]. This fold-wise restriction is the leakage control on which the present comparison rests. MinMaxScaler and QuantileTransformer are imported by the frozen code [34] but are not applied by any pipeline, so no equation is given for them.

$$z = \ln(1 + \max(x - s_{\text{train}}, 0)) \quad (1)$$

$$z = \sqrt{\max(x - \min_{\text{train}}(x) + \varepsilon, 0)} \quad (2)$$

$$z = \ln(1 + \max(x_{\max, \text{train}} - x, 0)) \quad (3)$$

$$r = \frac{a}{b + \varepsilon} \quad (4)$$

$$r_{\log} = \ln\left(1 + \frac{a}{b + \varepsilon}\right) \quad (5)$$

$$x_{\text{int}} = b \cdot r_{\log} \quad (6)$$

$$x_{\text{SIMPLE}} = \mathbf{1}[c_1 = 1 \wedge c_2 = 1] \quad (7)$$

$$z = \frac{x - \mu_{\text{train}}}{\sigma_{\text{train}}} \quad (8)$$

$$z = \frac{x - \text{med}_{\text{train}}(x)}{Q_{3, \text{train}} - Q_{1, \text{train}}} \quad (9)$$

$$\psi_{\lambda}(x) = \begin{cases} \frac{(x+1)^{\lambda-1}}{\lambda}, & \lambda \neq 0, x \geq 0 \\ \ln(x+1), & \lambda = 0, x \geq 0 \\ -\frac{(1-x)^{2-\lambda}-1}{2-\lambda}, & \lambda \neq 2, x < 0 \\ -\ln(1-x), & \lambda = 2, x < 0 \end{cases} \quad (10)$$

D. Model Training and Evaluation

Gradient-boosted decision trees are strong baselines for tabular clinical data and can match or surpass deep networks [35]; this choice is also supported by our work on the same dataset [20]. We therefore use LightGBM

How an LLM Generates a Preprocessing Pipeline

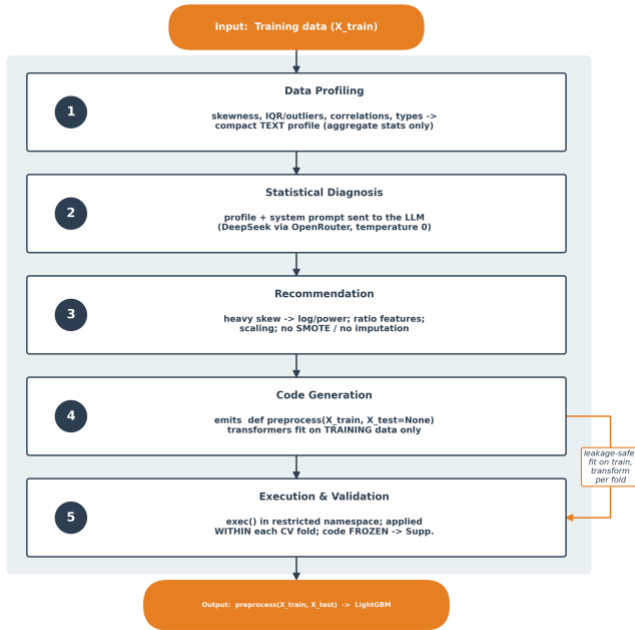


Fig. 2. Five-stage workflow for LLM-generated preprocessing, from data profiling to leakage-controlled validation.

feature statistics and target association computed from the training partition, not the locked holdout.

Stage 3 Recommendation Generation: Based on the statistical diagnosis, the LLM generates preprocessing recommendations with natural-language reasoning [21], [30]. For instance, upon detecting skewness of +111 in ANNUALCLAIMAMOUNT, a typical response is: 'Apply log1p transformation to ANNUALCLAIMAMOUNT to reduce extreme positive skew and normalize the distribution.' The LLM may also recommend feature engineering: 'Create CLAIM_PER_UNIT = ANNUALCLAIMAMOUNT / UNITSTOTAL as a domain-relevant ratio.' However, recommendation quality varies across models and prompts. This stage also exposes a key limitation: transformations may be suggested from statistical patterns without evidence that they improve a tree-based model on this dataset.

Stage 4 Executable Code Generation: Each model returns a Python function with the required preprocess(X_train, X_test=None) interface. The archived code is reviewed as experimental material, frozen, and then applied without performance-driven editing. Because executing generated code is inherently risky, the public workflow explicitly treats these archived files as trusted experiment artifacts rather than accepting arbitrary model output.

Stage 5 Leakage-Controlled Execution and Validation: For each CV split, preprocessing is fit on the fold-training records and applied to fold-validation records before model fitting. The final holdout is transformed only with state learned from the complete training partition. Code

Table 5. Classifier configurations and hyperparameters used in the evaluation.

Estimator configuration	Hyperparameters in force	Convergence
LightGBM (all analyses)	boosting_type=gbdt, n_estimators=100, learning_rate=0.1, num_leaves=31, max_depth=-1, min_child_samples=20, subsample=1.0, colsample_bytree=1.0, reg_alpha=0.0, reg_lambda=0.0, random_state=42 (library defaults, no tuning)	n/a
LR as delivered	solver=lbfgs, penalty=L2, C=1.0 (default), max_iter=5000, no scaling	converged; max 642 iterations
LR uniformly scaled	StandardScaler + solver=lbfgs, penalty=L2, C=1.0, max_iter=5000	converged; max 88 iterations
LR scaled + nested- tuned	StandardScaler + LogisticRegressionCV, solver=lbfgs, penalty=L2, C in {0.01, 0.1, 1, 10, 100} by 5-fold inner CV, max_iter=5000	converged; max 75 iterations

[36] as the primary classifier and gradient boosting [37] as the reference model class. Because gradient-boosted trees are generally insensitive to feature scaling and many monotonic transformations, although implementation-specific binning and numerical effects can produce small differences, we additionally evaluate Logistic Regression as a scale-sensitive linear model. Both classifiers were evaluated on identical 10-fold stratified splits and a locked holdout with default hyperparameters. Hyperparameter optimization is a separate performance lever [38], [39], [40], [41] and was not allowed to confound the preprocessing comparison. The primary robustness analysis repeated stratified 10-fold CV five times (50 paired test folds, random_state=2026), with preprocessing refitted inside every fold. Accuracy, MCC, and AUC were recorded per fold. Final LightGBM models were evaluated on the unchanged holdout using accuracy, weighted precision, weighted recall, weighted F1, AUC-ROC [42], precision-recall area under the curve (PR-AUC), Matthews correlation coefficient (MCC), Brier score, and expected calibration error (ECE) [43]; feature contributions were summarized with SHapley Additive exPlanations (SHAP) values [44]. Stratified bootstrap resampling (2,000 samples) quantified holdout uncertainty. Descriptive subgroup metrics were calculated for gender, age, diagnosis, comorbidity, complication, cover type, and scheme type. Decision curves across thresholds 0.10–0.90 were exploratory because the data support contemporaneous classification rather than prospective decisions. Table 5 lists every hyperparameter actually in force for both estimators and reports the Logistic Regression convergence status.

E. Statistical Framework

The original fixed-fold Friedman test [45] was retained only as a descriptive continuity analysis to facilitate comparison with previous studies. The primary statistical comparisons were conducted using repeated 5 × 10-fold cross-validation results with the Nadeau–Bengio corrected standard error [46], which accounts for dependence introduced by overlapping training sets. Corrected 90% confidence intervals were used for the two one-sided tests (TOST) of the paired accuracy differences [47], [48]. Operational equivalence was declared only when the entire corrected 90% confidence interval fell within the pre-specified ±0.01 accuracy margin. We chose the ±0.01 margin as an operational comparability

threshold representing a one-percentage-point accuracy difference considered too small to justify selecting a more complex preprocessing pipeline. It is approximately 1.4 times the observed mean fold-to-fold standard deviation (0.0072) and substantially larger than the standard error of the 10-fold mean (0.0023). The margin is operational rather than clinically validated. Given the large analytical sample (N = 24,071), the study provides sufficiently precise estimates of model performance [49]. Since all pairwise comparisons were pre-specified against the Manual Expert reference pipeline and interpreted primarily through confidence intervals and standardized effect sizes rather than isolated null-hypothesis significance tests, no additional Holm multiple-comparison adjustment [50] was applied.

To complement the repeated cross-validation analysis, 2,000 stratified bootstrap resamples of the independent holdout test set [51] were performed. These bootstrap analyses produced 95% confidence intervals for the performance of each pipeline and paired 90% confidence intervals for the accuracy differences used in the TOST equivalence analysis. Subgroup performance differences were reported descriptively and should not be interpreted as causal estimates of algorithmic fairness. The classification metrics are defined in Eqs. (11)–(13) [52], where TP, TN, FP, and FN denote the counts of true positives, true negatives, false positives, and false negatives; TPR and FPR are the true-positive and false-positive rates; and $Prec_n$ and Rec_n are precision and recall at the n -th decision threshold. The Matthews correlation coefficient, reported as a prevalence-robust complement to accuracy, is defined in Eq. (14).

$$\begin{aligned}
 Acc &= \frac{TP+TN}{TP+TN+FP+FN} \\
 Prec &= \frac{TP}{TP+FP} \\
 Rec &= \frac{TP}{TP+FN} \\
 Spec &= \frac{TN}{TN+FP}
 \end{aligned} \quad (11)$$

$$F_1 = \frac{2 \cdot Prec \cdot Rec}{Prec + Rec} \quad (12)$$

$$AUC = \int_0^1 TPR(FPR^{-1}(u))du \quad (13)$$

$$AP = \sum_n (Rec_n - Rec_{n-1}) Prec_n \quad (14)$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad (14)$$

Accuracy was designated as the primary equivalence endpoint because the pre-specified ±0.01 operational

margin was defined on the accuracy scale. The Matthews correlation coefficient (MCC), defined in Eq. (14), was retained as the primary balanced performance metric because it provides a balanced assessment of classification performance under class imbalance [53], [54]. Equivalence analyses for MCC and AUC were treated as secondary sensitivity analyses using the same absolute ± 0.01 exploratory margin; these secondary margins are operational rather than clinically validated. The complete procedure, including the generation stage that produces the frozen pipelines, is set out step by step in Algorithm 1.

The estimators underlying the equivalence analysis are defined in Eqs. (15)–(20) [46]. Here $a_{r,i}$ is the metric obtained on fold i of repeat r , with R repeats and k folds, so that $N = Rk$; $\bar{d}$ and σ are the mean and standard deviation of the paired per-fold differences; $n_{\text{test}}/n_{\text{train}}$ is the ratio of validation- to training-partition sizes within a fold, equal to $1/9$ for 10-fold partitioning; $\nu = N - 1$ is the number of degrees of freedom; $\delta = 0.01$ is the operational equivalence margin; and θ^* denotes the ordered bootstrap replicates of a statistic. Eq. (16) is required because the folds share training data, which makes the naive variance of the paired difference too small [46].

$$\bar{a} = \frac{1}{Rk} \sum_{r=1}^R \sum_{i=1}^k a_{r,i} \quad (15)$$

$$\hat{V}ar_{\text{corr}} = \left(\frac{1}{Rk} + \frac{n_{\text{test}}}{n_{\text{train}}} \right) \hat{\sigma}^2 \quad (16)$$

$$CI_{90\%} = \bar{d} \pm t_{0.95, \nu} \sqrt{\hat{V}ar_{\text{corr}}} \quad (17)$$

$$t_L = \frac{\bar{d} + \delta}{\sqrt{\hat{V}ar_{\text{corr}}}}, t_U = \frac{\bar{d} - \delta}{\sqrt{\hat{V}ar_{\text{corr}}}} \quad (18)$$

$$p_{\text{TOST}} = \max(P(T > t_L), P(T < t_U)) \quad (19)$$

$$CI_{95\%} = [\theta_{(0.025)}^*, \theta_{(0.975)}^*] \quad (20)$$

The Friedman test statistic and the corresponding Nemenyi critical difference are given in Eqs. (21) and (22), respectively.

$$\chi_F^2 = \frac{12N}{k(k+1)} \sum_{j=1}^k R_j^2 - 3N(k+1) \quad (21)$$

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}} \quad (22)$$

In Eqs. (21) and (22), N denotes the number of cross-validation folds, k is the number of evaluated pipelines, R_j is the mean rank of pipeline j , and q_α is the critical value of the Studentized range distribution. Two pipelines were considered significantly different only when the absolute difference between their mean ranks exceeded the critical difference. The paired effect size, Brier score, and Expected Calibration Error (ECE) are defined in Eqs. (23)–(25), respectively.

$$d_z = \frac{\bar{d}}{s_d} \quad (23)$$

$$\text{Brier} = \frac{1}{N} \sum_{i=1}^N (p_i - y_i)^2 \quad (24)$$

$$\text{ECE} = \sum_b \frac{|B_b|}{N} |\text{acc}(B_b) - \text{conf}(B_b)| \quad (25)$$

In Eqs. (23)–(25), $\bar{d}$ and s_d denote the mean and standard deviation of the paired per-fold accuracy differences, so that Cohen's paired effect size is $d_z = \frac{\bar{d}}{s_d}$ [55]; p_i is the predicted probability for sample i and y_i its true label in

the Brier score; and B_b is the b -th equal-width calibration bin, weighted by $\frac{|B_b|}{N}$ in the ECE. The Nadeau–Bengio correction inflates the naive variance of the paired difference by $(\frac{1}{N} + \frac{\rho}{(1-\rho)})$, where $\rho = \frac{1}{k}$ is the test-fold fraction. The TOST procedure tests $H_0: |\text{mean difference}| \geq 0.01$ against $H_1: |\text{mean difference}| < 0.01$; operational equivalence is declared only when the corrected 90% confidence interval lies entirely within the ± 0.01 margin. Predicted probabilities and the 0.5 decision threshold are identical across pipelines.

F. Reproducibility

The canonical offline workflow uses Python 3.11, fixed random_state=42, archived LLM code, one shared data loader, and one evaluator for all pipelines and classifiers. Exact dependencies are recorded in requirements.txt. Two result files back the reported numbers. The file outputs/final_canonical_results.json holds the single stratified 10-fold run (random_state = 42) used for Tables 6, 8, 9, 11 and 12 and for Figs. 3–5. The file outputs/revision_core_results.json holds the repeated 5×10 -fold run (seeds 2026+r) used for Tables 7, 10, and 13 and for the corrected equivalence tests. Each table caption states which protocol it reports. Raw prompts, responses, model identifiers, and generated code are retained in Supplementary Material. Hosted-model outputs remain version-sensitive despite temperature 0.0. Each one-shot pipeline was generated once, on 28 June 2026, through OpenRouter at temperature 0.0, using the model identifiers deepseek/deepseek-chat-v3.1, anthropic/claude-sonnet-4.6, and openai/gpt-5.4. Each model received an identical system prompt and a textual training-data profile (per-feature dtype, unique count, missing count, skewness, IQR, $1.5 \times \text{IQR}$ outlier count, and Pearson target correlation; plus global row and feature counts, class balance, and duplicates removed) and was asked for a single function preprocess(X_{train} , X_{test} =None) that fits any stateful transformer on X_{train} only. The single temperature-0.0 response was accepted without performance-driven editing; code had to satisfy the two-input contract and was otherwise frozen. No generated pipeline was excluded, as all three passed the contract. CAAFE was run separately as an iterative comparator with cross-validation feedback, and its accepted feature code was likewise frozen. The verbatim prompts, raw responses, model identifiers, frozen code, random seeds, and evaluators are available at <https://github.com/ihyaabrar/llm-vs-manual-preprocessing> so that the evaluation is fully auditable; hosted-model outputs remain version-sensitive despite temperature 0.0, so exact regeneration is not assumed. Only temperature was set explicitly; all unspecified sampling parameters, including top_p, max_tokens, and any seed, used the OpenRouter and provider defaults, and provider routing was not pinned. These values were not recorded at generation time and are therefore not reported as fixed settings.

To test whether supplying target correlations in the data profile encouraged supervised feature selection, each one-shot pipeline was regenerated from a profile

that omitted the target-correlation line. Accuracy changed only marginally and non-systematically: DeepSeek 0.8194 to 0.8190 (-0.0005); Claude Sonnet 4.6 0.8186 to 0.8201 (+0.0015); and GPT-5.4 0.8238 to 0.8221 (-0.0017). All changes lie within ± 0.002 and inside the operational margin, with no consistent direction, indicating that the supplied target-association information did not

materially drive pipeline quality. This check was run as a self-contained regeneration experiment with its own resampling seed, so only the paired within-model difference between the two profile variants is interpretable; the absolute values are not directly comparable with the frozen pipelines reported in Table 6.

Algorithm 1. Frozen LLM-Generated Preprocessing, Leakage-Controlled Repeated Cross-Validation, and Statistical Equivalence Evaluation.

Input: Dataset D, LLM set L, expert workflow p_{exp} , classifier set M, repeats R, folds k, margin $\delta = 0.01$	
Output: Frozen pipeline set P, corrected equivalence decisions, and holdout estimates	
1	Split D into locked holdout H (20%) and training pool T (80%), seed 42 → <i>holdout locked first</i>
2	profile $\leftarrow$ DataProfile(T) $\triangleright$ dtype, unique, missing, skewness, IQR, outliers, target correlation → <i>data profile</i>
3	for each LLM I in L do
4	code _I $\leftarrow$ Generate(I, profile, prompt), temperature 0.0 → <i>one-shot generation</i>
5	Freeze code _I $\triangleright$ never tuned on results → <i>freezing</i>
6	end for
7	code _{caafe} $\leftarrow$ CAAFE(T); Freeze code _{caafe} → <i>iterative comparator</i>
8	P $\leftarrow$ { p_{exp} } $\cup$ {code _I : I $\in$ L} $\cup$ {code _{caafe} } → <i>five pipelines</i>
9	for each pipeline p in P do
10	S(p) $\leftarrow$ $\emptyset$
11	for r = 1 to R do → <i>repeated cross-validation</i>
12	Partition T into k stratified folds, seed 2026 + r → <i>fold separation</i>
13	for each fold f do
14	T _{tr} $\leftarrow$ T \ f; T _{val} $\leftarrow$ f
15	$\theta \leftarrow$ FitPreprocessing(p, T _{tr}) → <i>fitted on training fold only</i>
16	X _{tr} $\leftarrow$ Transform(θ , T _{tr})
17	X _{val} $\leftarrow$ Transform(θ , T _{val}) → <i>leakage control</i>
18	for each classifier m in M do
19	h $\leftarrow$ Fit(m, X _{tr}) → <i>training</i>
20	Append accuracy, MCC and AUC of h on X _{val} to S(p) → <i>metrics, Eqs. (11)–(14)</i>
21	end for
22	end for
23	end for
24	$\theta_p \leftarrow$ FitPreprocessing(p, T)
25	Evaluate p on Transform(θ_p , H) → <i>locked holdout</i>
26	Bootstrap H 2,000 times for 95% CI → <i>uncertainty</i>
27	end for
28	for each p in P \ {p _{ref} } do
29	d $\leftarrow$ per-fold differences S(p _{ref}) – S(p), N = R · k
30	SE _c $\leftarrow$ NadeauBengio(d), $\rho = 1/k$ → <i>variance correction</i>
31	CI90 $\leftarrow$ mean(d) $\pm$ t(0.95, N–1) · SE _c → <i>corrected interval</i>
32	if CI90 $\subset$ (– δ , + δ) then → <i>equivalence test, Eqs. (15)–(20)</i>
33	Declare p equivalent to p _{ref}

```

34     end if
35 end for
36 return metrics, equivalence decisions, holdout estimates
    
```

III. Results

A. Core Finding: Comparable Performance Within the Operational Margin

Table 6 reports tightly clustered internal performance across the five frozen pipelines. LightGBM CV accuracy ranges from 0.8191 to 0.8229, and absolute CV-holdout gaps are at most 0.0038 (Fig. 3(b)). The descriptive Friedman statistic is chi-squared = 8.35 (df = 4, p = 0.080). Corrected TOST intervals from the repeated-CV analysis fall within the operational ± 0.01 margin. Because cross-validation folds are dependent, the corrected repeated-CV intervals not the uncorrected fixed-fold test provide the primary basis for interpretation. The findings support comparable internal record-level performance under this design; they do not establish clinical interchangeability. Table 7 reports the full corrected equivalence statistics for accuracy, MCC, and AUC. Table 8 reports paired Cohen's

and the three one-shot LLM pipelines, all pairwise effects are small or negligible ($|d_z| \leq 0.47$). CAAFE shows larger standardized effects against several pipelines ($|d_z|$ up to 0.87), largely because its paired fold differences have low variance; the absolute accuracy differences nevertheless remain inside the ± 0.01 operational margin, including, in this descriptive single 10-fold run, the Manual-versus-CAAFE comparison (mean difference +0.0028, TOST p < 0.001). The corresponding repeated 5 × 10-fold estimate in Table 7 is smaller (+0.0010) because it averages 50 folds across different seeds; both lie well within the ± 0.01 margin, and the repeated-CV value is used for inference. Effect sizes characterize the standardized paired differences and are not used alone to claim practical superiority. Fig. 3(a) visualizes the CV accuracy distributions across all five pipelines, confirming the tight clustering of performance.

Table 6. Cross-validation and locked-holdout performance of the five preprocessing pipelines using LightGBM.

Pipeline	Provider	CV Acc	Holdout Acc	AUC	Feats	CV - Holdout Gap	MCC / PR-AUC
Manual Expert	Human-defined baseline	0.8218	0.8199	0.9036	17	+0.0019	0.643 / 0.827
DeepSeek	Frozen OpenRouter response	0.8229	0.8193	0.9060	13	+0.0036	0.641 / 0.833
Claude Sonnet 4.6	Frozen OpenRouter response	0.8220	0.8183	0.9022	22	+0.0038	0.638 / 0.825
GPT-5.4	Frozen OpenRouter response	0.8209	0.8218	0.9062	34	-0.0009	0.646 / 0.833
CAAFE	Frozen iterative DeepSeek code	0.8191	0.8195	0.9027	12	-0.0005	0.642 / 0.826

d_z effect sizes for the five pipelines. Among Manual Expert

B. Pipeline Complexity and Performance

Table 7. Nadeau–Bengio-corrected paired equivalence analysis relative to the Manual Expert pipeline.

Metric	Manual minus	Mean difference	Corrected SE	Corrected 90% CI	TOST p	Within ± 0.01
Accuracy	DeepSeek	-0.0016	0.0015	[-0.0041, +0.0009]	<0.001	Yes
Accuracy	Claude 4.6	-0.0001	0.0016	[-0.0027, +0.0025]	<0.001	Yes
Accuracy	GPT-5.4	-0.0002	0.0015	[-0.0026, +0.0023]	<0.001	Yes
Accuracy	CAAFE	+0.0010	0.0017	[-0.0019, +0.0038]	<0.001	Yes
MCC	DeepSeek	-0.0031	0.0030	[-0.0083, +0.0020]	0.015	Yes
MCC	Claude 4.6	+0.0008	0.0033	[-0.0047, +0.0064]	0.004	Yes
MCC	GPT-5.4	+0.0004	0.0031	[-0.0048, +0.0055]	0.002	Yes
MCC	CAAFE	+0.0018	0.0035	[-0.0041, +0.0077]	0.012	Yes
AUC	DeepSeek	-0.0004	0.0006	[-0.0014, +0.0006]	<0.001	Yes
AUC	Claude 4.6	+0.0010	0.0008	[-0.0004, +0.0023]	<0.001	Yes
AUC	GPT-5.4	+0.0002	0.0008	[-0.0011, +0.0014]	<0.001	Yes
AUC	CAAFE	+0.0009	0.0007	[-0.0003, +0.0021]	<0.001	Yes

Despite comparable LightGBM accuracy, the pipelines differ markedly in complexity. Manual Expert uses 17 features (log1p plus three domain ratios, without scaling). DeepSeek is the most parsimonious one-shot pipeline (13 features), and CAAFE is leaner still (12 features), adding one verified interaction after five iterations. Claude Sonnet 4.6 is moderate (22 features), whereas GPT-5.4 is the most complex (34 features, including ratios, interactions, and standardization). The evaluated LLM pipelines therefore do not uniformly increase feature count. For LightGBM, additional scaling and interaction terms produced little change in accuracy, while the predefined manual workflow retained a compact set of transformations.

C. Ablation Study

The ablation study (Table 9) clarifies why the pipelines cluster under LightGBM. Ratio features alone match the Full Manual Expert accuracy (0.8218), whereas log1p transforms show no improvement over the raw baseline (0.8201). This indicates that ratio engineering, rather than

Repeated 5 × 10-fold validation produced closely clustered performance across all five pipelines (Table 10), with mean accuracy ranging from 0.8204 to 0.8229. Every Nadeau–Bengio-corrected 90% interval for Manual-minus-candidate accuracy lay inside the pre-specified ±0.01 operational margin. Stratified holdout bootstrap intervals also overlapped, supporting internal comparability without asserting clinical equivalence. Subgroup diagnostics identified lower accuracy for records with complications (maximum within-pipeline complication-status gap 0.122) and age-related variation. Because the task is contemporaneous, cross-sectional classification without a defined clinical intervention or an agreed benefit-harm ratio, the decision-curve analysis is not reported in the main results [56]; it is retained in the Supplementary Material as clearly labeled exploratory content, together with the full subgroup tables, and no claim of clinical utility is drawn from it. Across pipelines, the widest Nadeau–Bengio-corrected 90% interval for Manual-minus-candidate accuracy ranged from -0.0041 to 0.0009, and Manual Expert holdout accuracy had a

Table 9. Ablation analysis of preprocessing-component contributions using 10-fold cross-validation with LightGBM.

Ablation Variant	Features	Accuracy	SD	Delta from Raw
Raw (No Preprocessing)	11	0.8201	0.0071	+0.0000
log1p Only	15	0.8201	0.0071	+0.0000
Ratio Features Only	14	0.8218	0.0075	+0.0018
RobustScaler Only	11	0.8193	0.0077	-0.0007
log1p + Ratio	18	0.8218	0.0075	+0.0018
log1p + Scale	15	0.8193	0.0077	-0.0007
Full Manual Expert	17	0.8218	0.0075	+0.0018

Table 8. Pairwise Cohen’s d_z effect-size matrix for 10-fold cross-validation accuracy.

	Manual Expert	DeepSeek	Claude 4.6	GPT-5.4	CAAFE
Manual Expert	–	-0.41 (small)	-0.09 (negl.)	+0.26 (small)	+0.71 (medium)
DeepSeek	+0.41 (small)	–	+0.37 (small)	+0.47 (small)	+0.84 (large)
Claude 4.6	+0.09 (negl.)	-0.37 (small)	–	+0.29 (small)	+0.87 (large)
GPT-5.4	-0.26 (small)	-0.47 (small)	-0.29 (small)	–	+0.40 (small)
CAAFE	-0.71 (medium)	-0.84 (large)	-0.87 (large)	-0.40 (small)	–

the total number of transformations, accounts for the small observed gain. The evaluated LLM pipelines produced ratio or interaction features that preserved the dominant signal, while additional scaling and power transformations were largely inert for LightGBM. These findings are specific to this dataset and model class. The small feature-count difference between log1p + Ratio (18) and Full Manual Expert (17) arises because the ablation harness applies log1p to all four continuous predictors, including AGE, whereas the frozen Manual Expert artifact logs only CLAIM, UNITS, and CONTRIB. Thus, the ablation-only log1p(AGE) feature is not retained in the production artifact.

D. Repeated Validation, Bootstrap Uncertainty, and Subgroup Diagnostics

95% bootstrap interval of 0.8098 to 0.8301. Maximum within-pipeline subgroup accuracy gaps ranged from 0.090 (CAAFE) to 0.122 (DeepSeek). The ordering runs opposite to overall accuracy: the most accurate pipeline showed the widest subgroup gap and the least accurate the narrowest. With the five pipelines spanning only 0.0032 in accuracy, this pattern is descriptive rather than evidence of a systematic accuracy–disparity trade-off. However, it does show that equivalence in aggregate accuracy does not imply equivalence across subgroups.

E. Model-Class Sensitivity

To test whether the equivalence depends on the downstream estimator, all five pipelines were re-evaluated with Logistic Regression in three configurations: as delivered, with a uniform

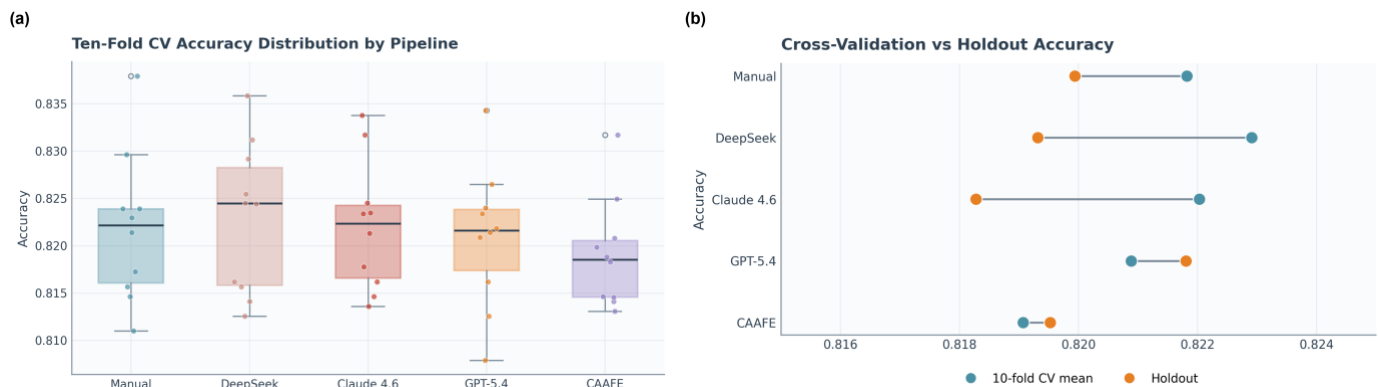


Fig. 3. Internal classification performance across the five preprocessing pipelines: (a) distribution of 10-fold cross-validation accuracy and (b) comparison of cross-validation and locked-holdout accuracy.

StandardScaler, and with uniform scaling plus nested-cross-validated L2 regularization (lbfgs solver, max_iter = 5000, C selected by 5-fold inner cross-validation over {0.01, 0.1, 1, 10, 100}), as reported in Table 11. All configurations in Table 11 use the same single stratified 10-fold protocol (shuffle, random_state = 42) as the LightGBM results in Table 6, so the estimators are compared under an identical resampling design. Under LightGBM, the accuracy span across pipelines was only 0.0038. As delivered, the Logistic-Regression span reached 0.202, because pipelines that omit scaling (Manual Expert 0.772; CAAFE 0.586) were penalized. After uniform scaling and tuning, the span narrowed to 0.161 and was concentrated almost entirely in CAAFE

174 for non-adherent records). Because adherence is defined from monthly refill counts and dispensed units rise mechanically with refills, UNITSTOTAL may act as an outcome-proximal feature. This concentration of discriminative signal also helps explain the similar LightGBM results, because preprocessing the remaining features has limited room to alter predictions. Removing UNITSTOTAL lowers holdout accuracy from 0.820 to 0.758 (AUC 0.903 to 0.839), and removing both UNITSTOTAL and ANNUALCLAIMAMOUNT lowers accuracy to 0.635, near the 0.598 majority-class baseline. The reported accuracy therefore reflects internal separability and should not be interpreted as prospective predictability. Across shared features, pairwise

Table 10. Repeated 5 × 10-fold cross-validation performance, holdout bootstrap uncertainty, and maximum subgroup accuracy gaps.

Pipeline	Repeated CV Acc	Repeated CV MCC	Repeated CV AUC	Holdout Acc (95% CI)	Corrected 90% CI vs Manual	Max subgroup Acc gap
Manual Expert	0.8213	0.642	0.903	0.8098 to 0.8301	Reference	0.117
DeepSeek	0.8229	0.645	0.903	0.8085 to 0.8297	-0.0041 to 0.0009	0.122
Claude Sonnet 4.6	0.8214	0.641	0.902	0.8073 to 0.8285	-0.0027 to 0.0025	0.108
GPT-5.4	0.8215	0.641	0.902	0.8108 to 0.8322	-0.0026 to 0.0023	0.107
CAAFE	0.8204	0.640	0.902	0.8083 to 0.8299	-0.0019 to 0.0038	0.090

(0.628), whose iteratively generated features remain poorly suited to a linear model; the manual and three one-shot-LLM pipelines fell within about 0.005 of one another (0.784–0.789). The model-class sensitivity is therefore real, but the divergence seen under a naively configured linear model reflects the absence of scaling in some pipelines and CAAFE specifically, rather than feature engineering in general. Preprocessing should be chosen jointly with the downstream estimator, favoring the simplest auditable pipeline that preserves performance.

F. SHAP, Calibration, and Error Analysis

SHAP analysis identifies UNITSTOTAL as the leading feature in every frozen pipeline. This dominance warrants scrutiny: although UNITSTOTAL has almost no linear correlation with the label (point-biserial $r = -0.002$), it alone attains 0.873 holdout AUC through a nonlinear threshold (median 692 dispensed units for adherent records versus

importance-rank correlations range from $p = 0.70$ to 0.99; agreement is high, but the explanations are not identical across pipelines. Calibration remains strong on the locked holdout (Fig. 4(b), Table 12; ECE < 0.02). Error overlap

Table 11. Model-class sensitivity of 10-fold cross-validation accuracy under LightGBM and Logistic Regression configurations.

Pipeline	LGBM	LR raw	LR scal.	LR tuned
Manual Expert	0.8218	0.7716	0.7892	0.7886
DeepSeek	0.8229	0.7841	0.7841	0.7843
Claude 4.6	0.8220	0.7864	0.7881	0.7879
GPT-5.4	0.8209	0.7885	0.7886	0.7884
CAAFE	0.8191	0.5863	0.6278	0.6281

(Fig. 4(a)) describes cases consistently missed by the evaluated pipelines without asserting that those records are intrinsically unclassifiable. Table 13 shows that the pipelines remain equivalent when UNITSTOTAL and all UNITSTOTAL-derived features are removed. Fig. 5 displays the Manual Expert importance profile, while the canonical JSON contains all five profiles. This dependence should be read as a methodological concern rather than a secondary sensitivity result. UNITSTOTAL counts dispensed medication units, while the PDC label is computed from medication refills, so the feature and label are derived from the same underlying events. A model that relies on UNITSTOTAL cannot be used to identify at-risk patients early, because by the time the value is

alone. Given a Brier spread of only 0.0021 across pipelines, these orderings should not be over-interpreted; the substantive finding is that all five pipelines are adequately calibrated internally.

IV. Discussion

A. Interpretation of Comparable Performance

The expanded validation supports the central finding. Across 50 paired repeated-CV test folds, mean accuracy differed by less than 0.004 across pipelines, and corrected 90% intervals remained inside the operational ± 0.01 margin. Holdout bootstrap confidence intervals overlapped substantially. The evidence therefore supports comparable internal performance among these

Table 12. Calibration performance of the five preprocessing pipelines on the locked holdout set.

Pipeline	Brier Score (lower is better)	ECE (lower is better)
Manual Expert	0.1210	0.0094
DeepSeek	0.1196	0.0153
Claude Sonnet 4.6	0.1214	0.0099
GPT-5.4	0.1195	0.0136
CAAFE	0.1216	0.0119

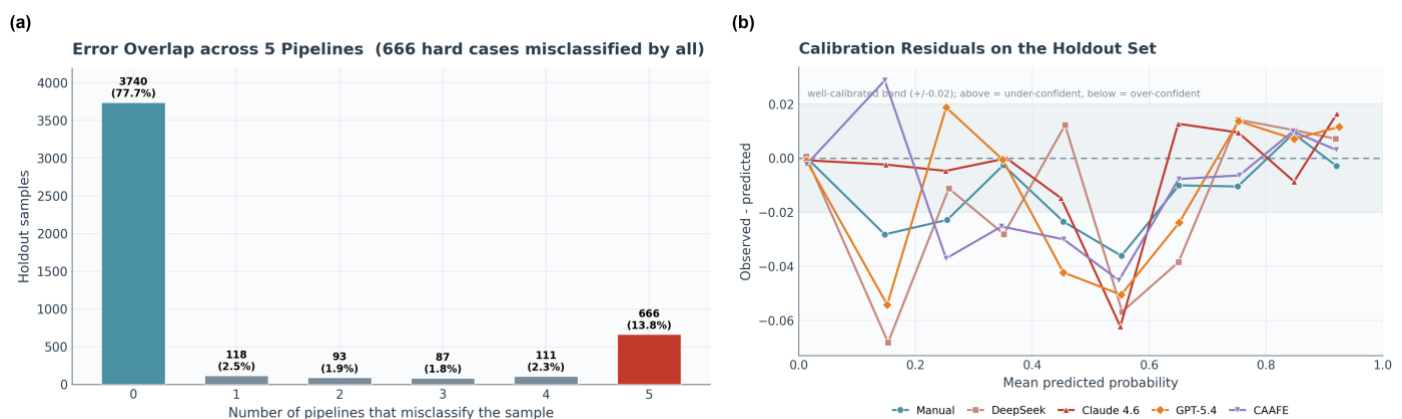


Fig. 4. Locked-holdout diagnostic analysis: (a) overlap of misclassification errors across the five preprocessing pipelines and (b) calibration residuals expressed as observed minus predicted probabilities.

observed, the adherence behavior has already occurred; it is confined to retrospective labeling rather than prospective intervention targeting. The magnitude is substantial: removing UNITSTOTAL lowers holdout accuracy from 0.820 to 0.758 and AUC from 0.903 to 0.839, and removing ANNUALCLAIMAMOUNT as well drives accuracy to 0.635, close to the 0.598 majority-class baseline. Reported high accuracy on this dataset should therefore be examined for outcome-proximal information before it is read as predictive skill.

Brier scores were tightly clustered (0.1195–0.1216), and ECE remained below 0.02 for every pipeline (0.0094–0.0153). The two calibration measures did not rank the pipelines in the same order: GPT-5.4 and DeepSeek attained the lowest Brier scores but the highest ECE values, whereas the Manual Expert and Claude Sonnet 4.6 pipelines showed the opposite pattern. This is expected because the Brier score combines calibration with discrimination, whereas ECE isolates calibration

five frozen workflows under LightGBM. It does not show that LLMs replace human experts: Manual Expert is one predefined workflow, and each one-shot LLM is represented by one archived generation. Comparable performance under this deliberately asymmetric design supports LLM-generated code as a potentially useful and auditable preprocessing aid, subject to human review, rather than as a substitute for domain expertise. The narrow LightGBM accuracy spread should be interpreted as robustness of the dominant predictive signal to several alternative preprocessing implementations, not as evidence that the pipelines are functionally identical. The pipelines differ in feature counts, transformations, and explanatory profiles, and their ordering changes under Logistic Regression. Therefore, preprocessing should be selected jointly with the downstream estimator, with preference for the simplest auditable pipeline that preserves performance rather than the pipeline that creates the most features.

B. Leakage-Controlled Evaluation

while researchers inspect, test, and approve them before

Table 13. Sensitivity analysis after removing UNITSTOTAL and its derived features.

Pipeline	CV Acc	CV MCC	CV AUC	Corrected 90% CI (Acc)	TOST p	Holdout Acc	Holdout 95% CI
Manual Expert	0.7558	0.5092	0.8387	reference	reference	0.7616	[0.7491, 0.7734]
DeepSeek	0.7556	0.5089	0.8386	[-0.0023, +0.0029]	<0.001 ✓	0.7576	[0.7452, 0.7697]
Claude 4.6	0.7556	0.5115	0.8378	[-0.0032, +0.0037]	<0.001 ✓	0.7599	[0.7479, 0.7718]
GPT-5.4	0.7551	0.5064	0.8379	[-0.0022, +0.0037]	<0.001 ✓	0.7533	[0.7416, 0.7651]
CAAFE	0.7569	0.5117	0.8391	[-0.0037, +0.0015]	<0.001 ✓	0.7585	[0.7462, 0.7703]

A critical methodological insight emerged during this study. Proper train-test separation in preprocessing is essential: scalers and transformers must be fit on training data only and applied to test data via `transform()`. When this principle is violated as can easily happen with auto-generated code that calls `fit_transform()` independently on each split, pipelines that include scaling (all three LLMs) exhibit apparent overfitting that is entirely an artifact of data leakage, not a genuine model failure. This underscores that the methodology of evaluation is as important as the preprocessing itself [25]. This is consistent with evidence that data partitioning materially affects measured performance stability [26]. Freezing generated code before evaluation is central to the validity and auditability of this comparison. Regenerating prompts or revising code after observing validation results would turn preprocessing into an implicit, unreported hyperparameter search. Archiving prompts, responses, model identifiers, code, random seeds, and evaluators allows reviewers to distinguish an LLM suggestion from a

evaluation.

C. Model Sensitivity and Clinical Interpretation

The ablation result that ratio features account for the observed LightGBM gain while `log1p` contributes none clarifies why the pipelines cluster. The evaluated LLMs proposed transformations justified by the supplied statistical profile, including transformations that the tree model did not require. This observation should not be generalized as a universal LLM tendency from one archived generation per model. Importantly, scaling that was largely inert for LightGBM became beneficial under Logistic Regression, demonstrating that preprocessing complexity is model-relative and must be evaluated jointly with the downstream estimator.

The ablation pattern admits a mechanistic explanation. LightGBM partitions data with threshold tests of the form $x \leq t$, and for any strictly increasing transformation g , the test $g(x) \leq g(t)$ induces exactly the same partition. Both `log1p` and `RobustScaler` are strictly increasing, so neither can create a split that the untransformed feature does not

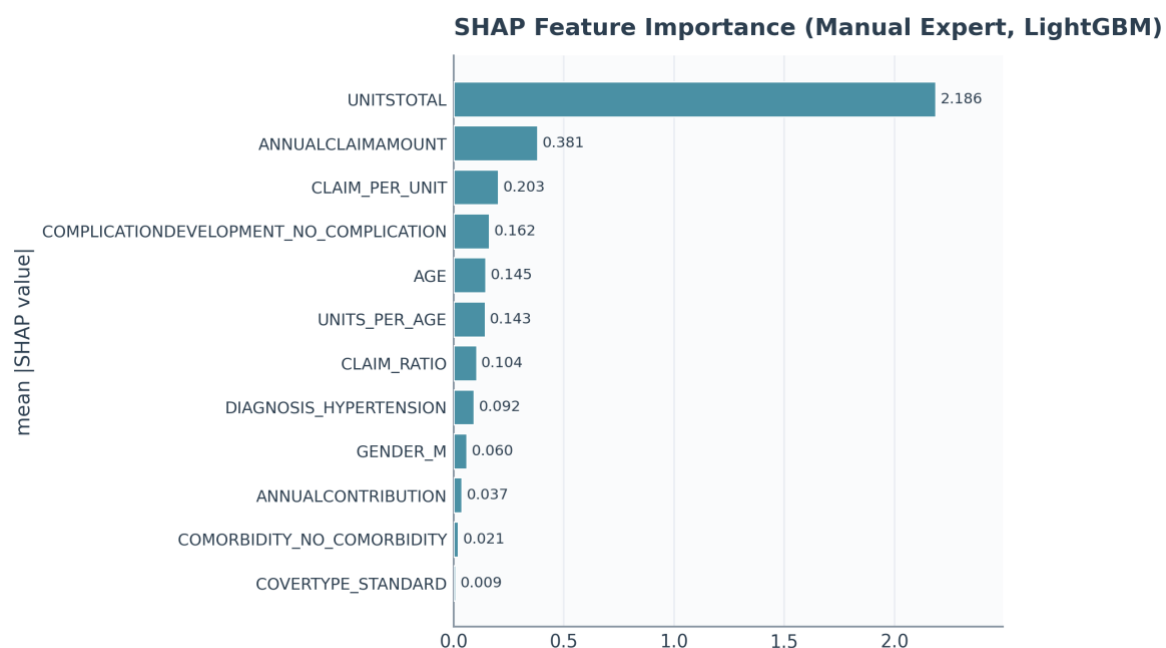


Fig. 5. SHAP-based global feature importance for the Manual Expert preprocessing pipeline with the LightGBM classifier.

transformation that has passed fold-safe empirical validation. In practice, the appropriate role is human-in-the-loop: the LLM proposes executable transformations,

already admit; the observed delta of +0.0000 for `log1p` is therefore the predicted result rather than a weak one. The small negative shift for scaling (-0.0007) arises because `log1p` was appended as additional columns (11 to 15

Table 14. Subgroup classification performance with bootstrap 95% confidence intervals.

Subgroup	n	Accuracy	Bootstrap 95% CI	Sensitivity	Specificity	MCC
age ≤45	1,143	0.850	[0.828, 0.871]	0.715	0.888	0.583
age 46–60	1,825	0.827	[0.810, 0.843]	0.870	0.800	0.654
age >60	1,847	0.795	[0.776, 0.813]	0.903	0.671	0.595
Male	2,557	0.826	[0.811, 0.840]	0.878	0.793	0.655
Female	2,258	0.814	[0.798, 0.829]	0.855	0.784	0.630
Hypertension	3,886	0.827	[0.815, 0.838]	0.870	0.799	0.655
Diabetes	929	0.791	[0.766, 0.816]	0.855	0.739	0.592
Comorbidity	1,380	0.792	[0.770, 0.814]	0.890	0.696	0.597
No comorbidity	3,435	0.831	[0.818, 0.843]	0.854	0.818	0.654
Complication	454	0.714	[0.672, 0.756]	0.784	0.656	0.440
No complication	4,361	0.831	[0.820, 0.842]	0.876	0.801	0.664
Standard	3,839	0.826	[0.814, 0.838]	0.872	0.796	0.656
Non-standard	976	0.795	[0.772, 0.819]	0.846	0.762	0.595
Premium	3,547	0.815	[0.803, 0.828]	0.869	0.775	0.637
Non-premium	1,268	0.833	[0.812, 0.853]	0.858	0.821	0.649

features) whereas RobustScaler replaced values in place, and LightGBM discretizes each feature into a histogram whose bin edges depend on the spacing of observed values rather than on rank alone; a monotone transformation preserves order but alters spacing, so a few values may cross bin boundaries. That shift is a discretization artifact roughly one-tenth the size of the between-fold standard deviation. Ratio features differ in kind: a/b is not a monotone function of any single feature, and the boundary $a/b \leq c$ is equivalent to $a \leq cb$, an oblique boundary that an axis-parallel tree can only approximate with a staircase of splits, each consuming depth and leaving fewer observations per leaf. Supplying the ratio directly grants that boundary in a single split. For tree-based models, the value of LLM-generated preprocessing therefore lies in feature construction rather than skewness correction or scaling.

Subgroup analysis adds an important qualification. Performance was relatively similar by gender, but varied more by age, diagnosis, comorbidity, and especially complication status. For Manual Expert, holdout accuracy was 0.714 among records with complications versus 0.831 without complications; similar gaps occurred across LLM pipelines. These differences may reflect clinical heterogeneity, sample size, or data-generation effects and should not be interpreted as causal bias. They show that aggregate equivalence can coexist with weaker performance in clinically relevant subgroups. Table 14 reports accuracy, bootstrap intervals, sensitivity, specificity, and MCC for every subgroup examined. Calibration and error-overlap analyses provide further context. ECE below 0.02 on the locked holdout indicates good internal probability calibration, but does not ensure calibration under temporal or geographic shift. The 666 records misclassified by all five pipelines suggest that a subset of errors reflects common limitations in the available variables rather than a particular preprocessing

strategy. Addressing these cases may require richer longitudinal or clinical information instead of additional transformations of the same cross-sectional attributes.

D. Comparison with Previous Research

Compared with published work on the same dataset, the Manual Expert accuracy in the present study (0.822 with default LightGBM hyperparameters) is close to the approximately 0.816 reported in our earlier boosting comparison [20] and below the 0.882 reported by Kanyongo et al. using Random Forest with a curated eight-feature subset [16]. These values are not directly interchangeable because the studies differ in feature subsets, algorithms, preprocessing implementations, and validation designs. The present study addresses a distinct question: it holds the evaluation procedure constant across preprocessing sources to isolate the contribution and model sensitivity of preprocessing rather than to identify the highest-performing classifier.

Relative to CAAFE [21] and Jellyfish [22], which demonstrate LLM-assisted feature engineering or data preparation on broader tabular tasks, this study adds a clinical-domain, leakage-controlled comparison against a predefined expert workflow. The adherence setting is especially restrictive because UNITSTOTAL is closely related to the refill-derived outcome; high internal accuracy can therefore coexist with weak prospective interpretability. This context explains why the present contribution is methodological a fair comparison of preprocessing sources rather than a claim that an LLM pipeline achieves a new state of the art in medication-adherence prediction. Table 15 sets this study against related work on dataset, model, label construction, validation design, and leakage control.

E. Research Limitations

These findings should be interpreted in light of several limitations. First, source documentation reports

Table 15. Quantitative comparison of the present study with related medication-adherence and LLM-assisted preprocessing studies.

Study	Dataset, size, and label	Classifier	Pre-processing	Validation and leakage	ACC.	AUC	Key limitation
Kanyongo et al. [16]	Cimas claims, Zimbabwe (same dataset); PDC $\geq$ 75%, record-level	Random Forest	Curated 8-feature subset, manual	Cross-validation	88.2%	0.935	No explicit fold-wise leakage control
Abrar et al. [20]	Cimas claims, Zimbabwe; 24,084 records, 11 predictors (40.15% adherent); PDC-based, record-level	LightGBM	One fixed manual pipeline; class weighting	Stratified 80:20 split + 10-fold CV; Wilcoxon	0.8163 (CV 0.8147 $\pm$ 0.0069)	0.9044	One manual pipeline; no LLM comparison
Gu et al. [18]	Connected sharps-bin records; 342,174 records over >5 years; Next adherence state, event-history based	LSTM	Event-history feature set	Deployed on a true future test set (temporal)	77.35%	0.8390	Not claims data; self-injecting population
Chen et al. [19]	TMUCRD, Taiwan; 4,134 patients, 2004–2020; MPR $\geq$ 80%, patient-level	XGBoost	Seven structured predictor domains	Separate internal and external test sets	Not reported	0.782 internal / 0.771 external	Restricted to first-time insulin users with T2DM
CAAFE [21]	OpenML tabular benchmarks; Varies by benchmark task	LLM feature engineering	LLM-generated and validated features, iterative	Cross-validation on OpenML	Not reported	0.798 $\rightarrow$ 0.822	Non-clinical; no expert comparator
Jellyfish [22]	Data-preparation tasks (imputation, error detection, schema matching); Task-dependent	Instruction-tuned local LLM	LLM-driven data preparation	Task-specific benchmarks	Not reported	Not reported	Not adherence; not a leakage-controlled comparison
This study	Cimas claims, Zimbabwe; 24,071 records (40.2% adherent); 19,256 train / 4,815 locked holdout; PDC $\geq$ 75%, record-level, cross-sectional	LightGBM and Logistic Regression	Five pipelines: one expert + three frozen LLM + CAAFE	Repeated 5 $\times$ 10-fold; fold-wise fitting; locked holdout; TOST $\pm$ 0.01	0.8204–0.8229	0.903	Cross-sectional internal; UNITSTOTAL outcome-proximal

beneficiary-level aggregation and de-duplication, but the public analytic workbook omits BENEFICIARY, preventing independent patient-group verification. Second, annual predictors and adherence are derived from the same 2022 measurement year; this is cross-sectional classification and cannot establish forward temporal prediction. Third, the study uses one Zimbabwean dataset, one predefined manual workflow, and one archived generation per one-shot LLM, so it does not fully estimate human-expert or LLM-generation variability. A limited sensitivity check found five independent DeepSeek regenerations (temperature 0.7) at 0.8186 ± 0.0009 accuracy with 14–17 features and one Claude Sonnet 4.6 regeneration at 0.822, but this check is insufficient for broad multi-model generalization. Fourth, UNITSTOTAL is partly a mechanical proxy for the refill-based label, so measured accuracy overstates genuine prospective predictability. Fifth, the ± 0.01 margin

is operational rather than clinically validated. Sixth, subgroup estimates are internal, and some groups are much smaller than others; no external cohort or prospective evaluation is available. Seventh, exploratory decision curves cannot establish clinical utility without a future-time prediction design. All findings require confirmation on independent, prospectively collected cohorts before clinical use.

F. Future Work

Future work should evaluate patient-level and temporally separated cohorts from multiple health systems, repeat each LLM generation across prompts and temperatures, and test expert workflows designed by multiple domain specialists. Analyses should also remove or prospectively measure outcome-proximal variables such as UNITSTOTAL, pre-specify clinically meaningful equivalence margins, and assess calibration drift, subgroup stability, decision-curve utility, computation cost,

and code-review burden. These extensions are necessary before the workflow can be considered for prospective clinical or operational use.

V. Conclusion

This study set out to determine whether frozen LLM-generated preprocessing can match a predefined Manual Expert workflow for medication-adherence classification when the data, downstream model, folds, and evaluation protocol are held constant. Within the pre-specified operational ± 0.01 margin, frozen preprocessing code from DeepSeek, Claude Sonnet 4.6, GPT-5.4, and CAAFE achieved LightGBM performance statistically equivalent to the Manual Expert workflow, with repeated 5×10 -fold accuracy ranging from 0.8204 to 0.8229, and the equivalence held for accuracy, MCC, and AUC alike. Component ablation localized the source of the small advantage over raw data to ratio features (+0.0018); log1p transformations contributed nothing (+0.0000) and scaling marginally reduced accuracy (-0.0007), which is what monotone-invariant tree splits predict. The equivalence also survived removal of the outcome-proximal feature UNITSTOTAL: holdout accuracy fell from 0.820 to 0.758 and holdout AUC from 0.903 to 0.839, while cross-validated accuracy fell to about 0.756, yet the five pipelines stayed indistinguishable (span 0.0018). Equivalence was nevertheless model-class dependent. Under Logistic Regression, the accuracy span reached 0.202 as delivered and narrowed to 0.161 after uniform scaling and nested-tuned regularization, with the residual spread concentrated in CAAFE. Preprocessing should therefore be selected jointly with the downstream estimator rather than in isolation, and for tree-based models its value lies in feature construction rather than distributional correction. Because UNITSTOTAL is mechanically related to the refill-derived label, the absolute accuracy reported here reflects internal separability rather than prospective predictability. Future work should therefore validate these findings on external and temporally separated cohorts from multiple health systems, repeat LLM generation across prompts, temperatures, and models, remove or prospectively measure outcome-proximal variables such as UNITSTOTAL, and replace the operational ± 0.01 margin with a clinically validated equivalence threshold before any prospective clinical or operational use.

Data Availability

The de-identified dataset is available from Mendeley Data, Version 2, under a CC0 1.0 public-domain dedication (DOI: 10.17632/zkp7sbbx64.2) [27].

<https://data.mendeley.com/datasets/zkp7sbbx64/2>

Code Availability

The Python code, the frozen LLM-generated preprocessing functions, the verbatim prompts and raw model responses, the evaluators, and all result files supporting this study are publicly available at <https://github.com/ihyaabrar/llm-vs-manual-preprocessing> and archived on Zenodo

(<https://doi.org/10.5281/zenodo.23228032>).

The repository includes the data loader, the statistical-analysis notebooks, and instructions to reproduce every table and figure reported here.

Declaration of Generative AI Use

The large language models studied here DeepSeek, Claude Sonnet 4.6, GPT-5.4, and the model underlying CAAFE were the object of investigation rather than tools used to prepare this manuscript. Their role, the prompts, the generation date, the model identifiers, and the freezing procedure are described in full in the Materials and Methods section, and the generated code is archived in the public repository. Separately, and in line with the journal policy on generative AI, the authors disclose that an AI assistant (Claude, Anthropic) was used during manuscript preparation for language editing, restructuring the Abstract and the Conclusion, and drafting explanatory passages written in response to reviewer comments. It was not used to design the study, to run or analyze the experiments, or to generate any result, figure, or the graphical abstract. The authors reviewed, verified, and edited all text produced with that assistance and are responsible and accountable for the entire contents of this work.

References

- [1] M. Naghavi *et al.*, "Global burden of 288 causes of death and life expectancy decomposition in 204 countries and territories and 811 subnational locations, 1990–2021: a systematic analysis for the Global Burden of Disease Study 2021," *The Lancet*, vol. 403, no. 10440, pp. 2100–2132, May 2024, doi: 10.1016/S0140-6736(24)00367-2.
- [2] E. Malaya, A. Piątkowska, M. Panek, P. Kuna, M. Kupczyk, and G. Kardas, "Medication adherence in allergic diseases and asthma: a literature review," *Front. Pharmacol.*, vol. 15, Art. no. 1488665, Dec. 2024, doi: 10.3389/fphar.2024.1488665.
- [3] A. Chantzaras and J. Yfantopoulos, "Determinants of medication adherence in patients with diabetes, hypertension, and hyperlipidemia," *Hormones*, vol. 24, no. 2, pp. 443–459, Jun. 2025, doi: 10.1007/s42000-025-00631-9.
- [4] A. J. Nelson, N. J. Pagidipati, and H. B. Bosworth, "Improving medication adherence in cardiovascular disease," *Nat. Rev. Cardiol.*, vol. 21, no. 6, pp. 417–429, Jun. 2024, doi: 10.1038/s41569-023-00972-1.
- [5] R. Khera *et al.*, "Use of Machine Learning Models to Predict Death After Acute Myocardial Infarction," *JAMA Cardiol.*, vol. 6, no. 6, p. 633, Jun. 2021, doi: 10.1001/jamacardio.2021.0122.
- [6] M. K. Biddinika, A. Masitha, H. Herman, and V. A. N. Fatimah, "Machine Learning Techniques for Heart Disease Prediction Using a Multi-Algorithm Approach," *JUITA: Jurnal Informatika*, vol. 12, no. 2, p. 149, Nov. 2024, doi: 10.30595/juita.v12i2.24153.

- [7] V. Koesmahargyo *et al.*, "Accuracy of machine learning-based prediction of medication adherence in clinical research," *Psychiatry Res.*, vol. 294, p. 113558, Dec. 2020, doi: 10.1016/j.psychres.2020.113558.
- [8] K.-S. Kim, B. Kim, and K. Han, "Big Data Research for Diabetes-Related Diseases Using the Korean National Health Information Database," *Diabetes Metab. J.*, vol. 49, no. 1, pp. 13–21, Jan. 2025, doi: 10.4093/dmj.2024.0780.
- [9] M. Li *et al.*, "Development and assessment of novel machine learning models to predict medication non-adherence risks in type 2 diabetics," *Front. Public Health*, vol. 10, Art. no. 1000622, Nov. 2022, doi: 10.3389/fpubh.2022.1000622.
- [10] E. A. Kassaw, A. K. Sendekie, B. M. Enyew, and B. B. Abate, "Machine learning applications to classify and monitor medication adherence in patients with type 2 diabetes in Ethiopia," *Front. Endocrinol. (Lausanne)*, vol. 16, Art. no. 1486350, Mar. 2025, doi: 10.3389/fendo.2025.1486350.
- [11] M. Owusu-Adjei, J. Ben Hayfron-Acquah, F. Twum, and G. Abdul-Salaam, "Machine Learning Modeling of Disease Treatment Default: A Comparative Analysis of Classification Models," *Adv. Public Health*, vol. 2023, pp. 1–11, Jan. 2023, doi: 10.1155/2023/4168770.
- [12] M. D. Teja and G. M. Rayalu, "Optimizing heart disease diagnosis with advanced machine learning models: a comparison of predictive performance," *BMC Cardiovasc. Disord.*, vol. 25, no. 1, p. 212, Mar. 2025, doi: 10.1186/s12872-025-04627-6.
- [13] K. Nabrdalik *et al.*, "Machine Learning Predicts Cardiovascular Events in Patients With Diabetes: The Silesia Diabetes-Heart Project," *Curr. Probl. Cardiol.*, vol. 48, no. 7, p. 101694, Jul. 2023, doi: 10.1016/j.cpcardiol.2023.101694.
- [14] A. Pfob, S.-C. Lu, and C. Sidey-Gibbons, "Machine learning in medicine: a practical introduction to techniques for data pre-processing, hyperparameter tuning, and model comparison," *BMC Med. Res. Methodol.*, vol. 22, no. 1, p. 282, Nov. 2022, doi: 10.1186/s12874-022-01758-8.
- [15] W. Kanyongo and A. E. Ezugwu, "Machine learning approaches to medication adherence amongst NCD patients: A systematic literature review," *Inform. Med. Unlocked*, vol. 38, p. 101210, 2023, doi: 10.1016/j.imu.2023.101210.
- [16] W. Kanyongo, A. E. Ezugwu, T. Moyo, and J. V. Fonou Dombau, "Machine learning-based classification of medication adherence among patients with noncommunicable diseases," *Inform. Med. Unlocked*, vol. 52, p. 101611, 2025, doi: 10.1016/j.imu.2024.101611.
- [17] W. Kanyongo, A. E.-S. Ezugwu, T. Moyo, and J. V. F. Dombau, "Data Wrangling and Generation for Machine Learning Models in Medication Adherence Analytics: A practical Standpoint using Patient-Level and Medical Claims Data," *Data Intell.*, vol. 7, no. 2, pp. 485–526, Jun. 2025, doi: 10.3724/2096-7004.di.2024.0037.
- [18] Y. Gu *et al.*, "Predicting medication adherence using ensemble learning and deep learning models with large scale healthcare data," *Sci. Rep.*, vol. 11, no. 1, p. 18961, Sep. 2021, doi: 10.1038/s41598-021-98387-w.
- [19] Y.-L. Chen, P.-A. Nguyen, C.-H. Chien, M.-H. Hsu, D.-M. Liou, and H.-C. Yang, "Machine learning-based prediction of medication refill adherence among first-time insulin users with type 2 diabetes," *Diabetes Res. Clin. Pract.*, vol. 207, p. 111033, Jan. 2024, doi: 10.1016/j.diabetes.2023.111033.
- [20] I. N. Abrar, M. K. Biddinika, and H. Yuliansyah, "Comparative Evaluation of Boosting Ensemble Models for Medication Adherence Prediction in Patients with Non-Communicable Diseases," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 15, no. 3, pp. 462–472, Jul. 2026, doi: 10.32736/sisfokom.v15i3.2712.
- [21] N. Hollmann, S. Müller, and F. Hutter, "Large Language Models for Automated Data Science: Introducing CAAFE for Context-Aware Automated Feature Engineering," in *Advances in Neural Information Processing Systems 36*, San Diego, California, USA: Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2023, pp. 44753–44775. doi: 10.52202/075280-1938.
- [22] H. Zhang, Y. Dong, C. Xiao, and M. Oyamada, "Jellyfish: Instruction-Tuning Local Large Language Models for Data Preprocessing," in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2024, pp. 8754–8782. doi: 10.18653/v1/2024.emnlp-main.497.
- [23] 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, p. 5461, Aug. 2024, doi: 10.3390/s24175461.
- [24] 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, no. 1, p. 140, Oct. 2021, doi: 10.1186/s40537-021-00516-9.
- [25] K. Mahmud Sujon, R. Binti Hassan, Z. Tusnia Towshi, M. A. Othman, M. Abdus Samad, and K. Choi, "When to Use Standardization and Normalization: Empirical Evidence From Machine Learning Models and XAI," *IEEE Access*, vol. 12, pp. 135300–135314, 2024, doi: 10.1109/ACCESS.2024.3462434.
- [26] V. Singh, M. Pencina, A. J. Einstein, J. X. Liang, D. S. Berman, and P. Slomka, "Impact of train/test sample regimen on performance estimate stability of machine learning in cardiovascular imaging," *Sci. Rep.*, vol. 11, no. 1, p. 14490, Jul. 2021, doi: 10.1038/s41598-021-93651-5.

- [27] W. Kanyongo, T. Moyo, A. E.-S. Ezugwu, and J. V. F. Dombeu, "Dataset for analysing medication adherence among diabetes and hypertension patients: Patient-level and medication refill data, *Mendeley Data*, Version 2, 2024, doi: 10.17632/zkp7sbbx64.2.
- [28] N. Sambasivan, S. Kapania, H. Highfill, D. Akrong, P. Paritosh, and L. M. Aroyo, "‘Everyone wants to do the model work, not the data work’: Data Cascades in High-Stakes AI," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, New York, NY, USA: ACM, May 2021, pp. 1–15. doi: 10.1145/3411764.3445518.
- [29] J. White *et al.*, "A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT," in *Proceedings of the 30th Conference on Pattern Languages of Programs*, in PLoP '23. USA: The Hillside Group, 2023, Art. no. 5, pp. 1–31, doi: 10.64346/PLoP2023p05.
- [30] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing," *ACM Comput. Surv.*, vol. 55, no. 9, pp. 1–35, Sep. 2023, doi: 10.1145/3560815.
- [31] G. E. P. Box and D. R. Cox, "An Analysis of Transformations," *J. R. Stat. Soc. Series B Stat. Methodol.*, vol. 26, no. 2, pp. 211–243, Jul. 1964, doi: 10.1111/j.2517-6161.1964.tb00553.x.
- [32] M. Kuhn and K. Johnson, *Feature Engineering and Selection*. Boca Raton, FL: Chapman and Hall/CRC, 2019. doi: 10.1201/9781315108230.
- [33] I.-K. Yeo, "A new family of power transformations to improve normality or symmetry," *Biometrika*, vol. 87, no. 4, pp. 954–959, Dec. 2000, doi: 10.1093/biomet/87.4.954.
- [34] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [35] R. Shwartz-Ziv and A. Armon, "Tabular data: Deep learning is not all you need," *Information Fusion*, vol. 81, pp. 84–90, May 2022, doi: 10.1016/j.inffus.2021.11.011.
- [36] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, "A comparative analysis of gradient boosting algorithms," *Artif. Intell. Rev.*, vol. 54, no. 3, pp. 1937–1967, Mar. 2021, doi: 10.1007/s10462-020-09896-5.
- [37] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA: ACM, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.
- [38] E. Elgeldawi, A. Sayed, A. R. Galal, and A. M. Zaki, "Hyperparameter Tuning for Machine Learning Algorithms Used for Arabic Sentiment Analysis," *Informatics*, vol. 8, no. 4, p. 79, Nov. 2021, doi: 10.3390/informatics8040079.
- [39] M. Wojciuk, Z. Swiderska-Chadaj, K. Siwek, and A. Gertych, "Improving classification accuracy of fine-tuned CNN models: Impact of hyperparameter optimization," *Heliyon*, vol. 10, no. 5, p. e26586, Mar. 2024, doi: 10.1016/j.heliyon.2024.e26586.
- [40] S. Shekhar, A. Bansode, and A. Salim, "A Comparative study of Hyper-Parameter Optimization Tools," in *2021 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, IEEE, Dec. 2021, pp. 1–6. doi: 10.1109/CSDE53843.2021.9718485.
- [41] L. González-Castro, M. Chávez, P. Duflo, V. Bleret, G. Del Fiol, and M. López-Nores, "Impact of Hyperparameter Optimization to Enhance Machine Learning Performance: A Case Study on Breast Cancer Recurrence Prediction," *Applied Sciences*, vol. 14, no. 13, p. 5909, Jul. 2024, doi: 10.3390/app14135909.
- [42] T. Yang and Y. Ying, "AUC Maximization in the Era of Big Data and AI: A Survey," *ACM Comput. Surv.*, vol. 55, no. 8, pp. 1–37, Aug. 2023, doi: 10.1145/3554729.
- [43] M. Cearns, T. Hahn, S. Clark, and B. T. Baune, "Machine learning probability calibration for high-risk clinical decision-making," *Australian & New Zealand Journal of Psychiatry*, vol. 54, no. 2, pp. 123–126, Feb. 2020, doi: 10.1177/0004867419885448.
- [44] S. M. Lundberg *et al.*, "From local explanations to global understanding with explainable AI for trees," *Nat. Mach. Intell.*, vol. 2, no. 1, pp. 56–67, Jan. 2020, doi: 10.1038/s42256-019-0138-9.
- [45] S. García, A. Fernández, J. Luengo, and F. Herrera, "Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power," *Inf. Sci. (N. Y.)*, vol. 180, no. 10, pp. 2044–2064, May 2010, doi: 10.1016/j.ins.2009.12.010.
- [46] C. Nadeau and Y. Bengio, "Inference for the Generalization Error," *Mach. Learn.*, vol. 52, no. 3, pp. 239–281, Sep. 2003, doi: 10.1023/A:1024068626366.
- [47] D. J. Schuirmann, "A comparison of the Two One-Sided Tests Procedure and the Power Approach for assessing the equivalence of average bioavailability," *J. Pharmacokinet. Biopharm.*, vol. 15, no. 6, pp. 657–680, Dec. 1987, doi: 10.1007/BF01068419.
- [48] D. Lakens, "Equivalence Tests," *Soc. Psychol. Personal. Sci.*, vol. 8, no. 4, pp. 355–362, May 2017, doi: 10.1177/1948550617697177.
- [49] F. Faul, E. Erdfelder, A. Buchner, and A.-G. Lang, "Statistical power analyses using G*Power 3.1: Tests for correlation and regression analyses," *Behav. Res. Methods*, vol. 41, no. 4, pp. 1149–1160, Nov. 2009, doi: 10.3758/BRM.41.4.1149.
- [50] M. Aickin and H. Gensler, "Adjusting for multiple testing when reporting research results: the Bonferroni vs Holm methods.," *Am. J. Public*

Health, vol. 86, no. 5, pp. 726–728, May 1996, doi: 10.2105/AJPH.86.5.726.

- [51] B. Efron, "Better Bootstrap Confidence Intervals," *J. Am. Stat. Assoc.*, vol. 82, no. 397, pp. 171–185, Mar. 1987, doi: 10.1080/01621459.1987.10478410.
- [52] T. Saito and M. Rehmsmeier, "The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets," *PLoS One*, vol. 10, no. 3, p. e0118432, Mar. 2015, doi: 10.1371/journal.pone.0118432.
- [53] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, no. 1, p. 6, Dec. 2020, doi: 10.1186/s12864-019-6413-7.
- [54] K. Davagdorj, J. S. Lee, V. H. Pham, and K. H. Ryu, "A Comparative Analysis of Machine Learning Methods for Class Imbalance in a Smoking Cessation Intervention," *Applied Sciences*, vol. 10, no. 9, p. 3307, May 2020, doi: 10.3390/app10093307.
- [55] D. Lakens, "Calculating and reporting effect sizes to facilitate cumulative science: a practical primer for t-tests and ANOVAs," *Front. Psychol.*, vol. 4, Art. no. 863, 2013, doi: 10.3389/fpsyg.2013.00863.
- [56] A. J. Vickers and E. B. Elkin, "Decision Curve Analysis: A Novel Method for Evaluating Prediction Models," *Medical Decision Making*, vol. 26, no. 6, pp. 565–574, Nov. 2006, doi: 10.1177/0272989X06295361.

AUTHOR BIOGRAPHY



Herman Yuliansyah received the bachelor's (S.T.) degree from Universitas Muhammadiyah Yogyakarta, Indonesia, in 2007, the M.Eng. degree in information technology from Universitas Gadjah Mada, Yogyakarta, Indonesia, in 2011, and the Ph.D. degree in computer science from Universiti Kebangsaan Malaysia in 2023. He is currently an associate professor in the Department of Informatics, Faculty of Industrial Technology, Universitas Ahmad Dahlan, Yogyakarta, Indonesia. His research interests include data mining, social network analysis, machine learning, and health informatics. He also supervises graduate research in the Master's Program of Informatics at the same faculty. His published work spans association-rule mining for web-based data analysis, link prediction in social networks using graph attention networks, and secure management of medical records. He is the corresponding author of this study



Ihya' Nashirudin Abrar received the bachelor's (S.Kom.) degree in informatics from Universitas Muhammadiyah Pontianak, Indonesia, in 2022. He is currently pursuing a master's degree in informatics at the Faculty of Industrial Technology, Universitas Ahmad Dahlan, Yogyakarta, Indonesia. His research interests include machine learning and medical data analytics. His first-authored work includes a study on liver-disease classification that applied the elbow method to select the number of clusters, published in 2023, and a comparative evaluation of boosting ensemble models for medication-adherence prediction on the same medical-claims dataset used in the present study, published in 2026. This study forms part of his master's research.



Muhammad Kunta Biddinika received the bachelor's (S.T.) degree in nuclear engineering from Universitas Gadjah Mada, Indonesia, in 2004, the M.Eng. degree in nuclear engineering from the Tokyo Institute of Technology, Japan, in 2009, and the D.Eng. degree in environmental science and engineering from the Tokyo Institute of Technology, Japan, in 2016. He is currently a lecturer in the Master's Program in Informatics, Faculty of Industrial Technology, Universitas Ahmad Dahlan, Yogyakarta, Indonesia. His research interests include renewable energy, scientometrics and bibliometric analysis, engineering education, and machine-learning applications. His recent work includes a multi-algorithm comparison of machine-learning techniques for heart-disease prediction, published in 2024. He also contributes to graduate supervision in the Master Program in Informatics.

Frozen LLM-Generated Preprocessing for Medical-Claims Adherence Classification: An Internal Model-Sensitivity Study

One predefined expert workflow and four LLM-generated pipelines, evaluated under identical leakage-controlled folds.

