

Machine Learning-Based Prediction of Construction Project Costs and Overruns Using a Comparative Study of Benchmark Datasets

Mustafa R. Al-Saadi

Assistant Lecturer, Department of Computer Science, College of Science, Mustansiriyah University, Baghdad, Iraq
mustafa.r.alsaadi@uomustansiriyah.edu.iq

Abstract

A common failure in construction project delivery around the world is cost overruns and schedule delays. This research benchmarks five machine learning models (Random Forest (RF), Extreme Gradient Boosting (XGBoost), Light Gradient Boosting Machine (LightGBM), Support Vector Machine (SVM), and a Multilayer Perceptron Artificial Neural Network (ANN)) on two public benchmark datasets, under a single unified protocol: (1) the UCI Residential Building Data Set consisting of 107 predictors and 372 projects for construction cost and sales price prediction, and (2) 2,590 project-phase records from the New York City (NYC) Open Data portal, for time and cost overrun prediction and delay-risk classification. All models were tuned by randomized search with 5-fold cross-validation and evaluated on held-out test sets, and the best models were interpreted using SHapley Additive exPlanations (SHAP). XGBoost achieved the best cost-prediction performance on the UCI benchmark ($R^2 = 0.978$, $MAPE = 8.74\%$), exceeding recently published results, with SHAP attributing predictions primarily to the preliminary cost estimate, construction duration, and lagged macro-economic indices. On the NYC data, point regression of overrun magnitude from administrative features alone proved weak (best $R^2 = 0.32$), whereas binary delay-risk classification was strong (LightGBM $F1 = 0.734$, $AUC = 0.829$) on a nearly balanced sample, matching or exceeding prior questionnaire-based delay classifiers. The findings indicate that gradient-boosted ensembles are the most robust default for heterogeneous construction data, and that machine learning can already serve as an effective delay-triage tool using routinely archived administrative records — a practice directly transferable to public project governance in developing contexts such as Iraq.

Keywords: Machine Learning, Construction Management, Cost Prediction, Schedule Delay, Cost Overrun, XGBoost, SHAP, Engineering Management.

1. Introduction

Construction projects are notoriously prone to exceeding their planned budgets and schedules. Audits of large public portfolios routinely find that a majority of projects finish late and over budget; an analysis of more than 5,100 New York City capital projects, for example, found that over half experienced cost overruns and nearly three-quarters of delayed projects were attributable to recurring

managerial and budgetary causes [1]. The impacts are heightened in developing economies where fiscal space is constrained, public infrastructure demands are pressing and public funds are scarce. In Iraq in particular, the delivery of public works often suffers due to bureaucratic constraints, funding delays and supply chain difficulties [2] and effective early-warning mechanisms for overrun delivery are rarely used in the practice of project management ministerial and provincial level.

In recent years, machine learning (ML) has demonstrated its potential to enhance the construction industry by providing a viable alternative to regression and expert-judgment-based estimation. The field of machine learning (ML) has been gaining momentum in recent years as a viable method for construction industry estimation, as well as an alternative to regression and expert-judgment-based estimation. Only some of these models have been used for cost estimation, overrun prediction and delay-risk classification with promising results in different countries and project types [3-9] but three limitations remain in the literature. First, most studies use one model or a very small number of models chosen for the specific task, making it difficult to distinguish the merits of the different models from the tuning effort that went into them; a systematic comparison of baseline and ensemble learners using a common evaluation protocol is a rarity [3]. Second, model interpretability is a recognized limitation: many high accuracy models are essentially black boxes and so are not readily accepted by project-management practitioners [8, 10] and the recent trend in applying explainable-AI (XAI) in the construction domain has been maturing and lacks in uniformity [10-13]. Third, the studies often are based on custom-made questionnaire data, limiting the possibility to reproduce the results, while there are underutilized benchmark datasets available for public use and can provide cumulative and comparable evidence.

This study fills these gaps by using a common two-dataset benchmarking framework. The five most popular ML models, RF, XGBoost, LightGBM, SVM and ANN, are trained, tuned, and evaluated in the same manner on (i) the well-scrimmaged, public UCI Residential Building Data Set, which is the most widely reused construction cost prediction benchmark, and (ii) a dataset of real project-phase records, programmatically pulled from the NYC Open Data portal, on which actual and planned schedules and budgets are compared to calculate time overruns, cost overruns, and delay labels. SHAP analysis is applied to the best model of each task to expose the drivers of its predictions. The contributions are threefold: (1) a systematic, equally tuned comparison of five model families across four prediction tasks on two public datasets; (2) empirical evidence that the tractability of overrun prediction depends decisively on problem formulation — weak point regression versus strong delay-risk classification from administrative features; and (3) explainability-based managerial insights, including evidence that ML cost models operate as data-driven correction layers over preliminary engineering estimates. The rest of the paper is organized as follows: In Section 2, we review related work; in Section 3, we outline the datasets and methodology; in Section 4, we present and discuss the results; and finally, in Section 5, we conclude.

2. Related Work

2.1 Machine Learning for Construction Cost and Overrun Prediction:

Gradient-boosted ensembles have been proven to perform in the top tier worldwide on tabular construction data. Alshboul et al. [4] tested XGBoost, deep neural networks, and RF for predicting green building costs, achieving the highest R^2 value of 0.96 for XGBoost. Hamdan et al. [3] cross-validated fifteen regression algorithms on 836 Jordanian public projects, reporting CatBoost as best (test $R^2 = 0.883$) followed by stacking, XGBoost (0.844), and RF (0.802), with variation orders carrying approximately 41% of feature importance. Coffie and Cudjoe [11] applied XGBoost with SHAP to Ghanaian projects (RMSE = 0.202), finding the initial contract amount and initial duration to be the strongest overrun predictors. For delay prediction, Gondia et al. [5] achieved 74.5% and 78.4% accuracy with decision-tree and naïve Bayesian classifiers on Egyptian building projects; Asadi et al. [6] reported 79.4% and 73.5% for construction-logistics delays in Qatar; and Alsulamy [7] found LightGBM superior to CatBoost and XGBoost across all time-overrun severity bands in Saudi projects. Joint prediction of both time and cost overruns is comparatively rare: Al mnaseer et al. [9] used a Tabu-Search-tuned ANN on 191 Jordanian projects (cost $R^2 = 0.938$; time $R^2 = 0.939$), and El-Kholy [16] predicted both delay and cost-overrun percentages of highway projects with ANN paradigms.

2.2 Explainable AI in Construction Analytics:

A distinct XAI wave has emerged since 2024. Beyond Coffie and Cudjoe [11], recent studies combine ensembles with SHAP for prefabricated-construction delay risk (stacking ensemble, $R^2 = 0.931$) [12], delay assessment integrated with multi-criteria decision analysis (XGBoost, $R^2 = 0.96$; SHAP and partial-dependence analysis identifying unrealistic scheduling, contractor financial constraints, and delayed payments as dominant drivers) [13], and BIM-integrated joint cost-schedule-carbon risk prediction (XGBoost cost $R^2 = 0.939$, schedule $R^2 = 0.928$) [17]. However, only a few construction studies use XAI and uncertainty and interpretability are usually dealt with separately as opposed to being part of the same construction model.[18 ,10]

2.3 Studies in the Iraqi Context:

There are applications of ML in Iraq to project performance, but they have similar limitations. Yaseen et al. [2] presented a novel model (Random Forest–Genetic Algorithm) to classify construction project delays in Diyala based on the questionnaire result achievements as delay factors (accuracy 91.67%). Ali et al. [19] combined the XGBoost input selection with the RF, ANN, and SVM models to estimate the cost of a survey of 90 building projects from Iraq, the highest R^2 (0.87) was obtained for the XGBoost + SVM model, where the most important predictor was inflation. Zehawi [20] proposed ANN models for estimation of the cost of construction of highways in Iraq, using 291 projects with a value of $R = 0.989$. Kamil [21] used 35 projects of urban pavements in Najaf to evaluate the deep neural networks for estimation of delay risk. In this corpus, studies tend

to be based on questionnaires or small samples of experts, focus on one specific target (cost and/or delay, but not both), study only a few models and do not use SHAP to explain. The unified multi-model, multi-task, explainable framework proposed for the public benchmark databases in the present work is intended to be readily applicable to Iraqi project databases which are to be systematically archived in the future.

2.4 Summary of Prior Studies and Research Gap:

The most closely related studies are summarized in Table 1. There are three gaps readily apparent: first, only a handful of works benchmark multiple families of models using a single well-tuned protocol; second, SHAP-based explainability has only recently been found in studies conducted internationally and not yet in any Iraqi study; and finally, joint prediction of both time and cost outcomes within a single explainable framework is quite uncommon. The current study is situated at the juncture of these three gaps.

Table (1): Comparison of representative prior studies on ML-based cost/delay prediction

Study	Country / context	Data & size	Best model	Reported performance	XAI	Target
Yaseen et al. [2], 2020	Iraq (Diyala)	Questionnaire	RF-GA	Acc 91.7%	—	Time
Ali et al. [19], 2022	Iraq	90 building projects	XGBoost-RF	R^2 0.87	—	Cost
Zehawi [20], 2024	Iraq (highways)	291 projects	ANN	R 0.989	—	Cost
Kamil [21], 2025	Iraq (Najaf)	35 pavement projects	DNN	Validated ANN	—	Time
Gondia et al. [5], 2020	Egypt	Survey (buildings)	Naïve Bayes	Acc 78.4%	—	Time
Al mnaseer et al. [9], 2023	Jordan	191 projects	ANN + Tabu	R^2 0.938 / 0.939	—	Both
Alshboul et al. [4], 2022	Green buildings	Project data	XGBoost	R^2 0.96	—	Cost
Coffie & Cudjoe [11], 2024	Ghana	Projects 2016–18	XGBoost	RMSE 0.202	SHAP	Cost
Hamdan et al. [3], 2025	Jordan	836 public projects	CatBoost	R^2 0.883	SHAP	Cost
Alsulamy [7], 2025	Saudi Arabia	TO severity bands	LightGBM	Best in all bands	—	Time
Shyam & Tiwari [13], 2026	India	45 delay variables	XGBoost	R^2 0.96	SHAP +PDP	Time/cost/quality
Zawaideh [17], 2026	High-rise (BIM)	BIM-derived data	XGBoost	R^2 0.939 / 0.928	SHAP	Both
This study	UCI (Iran) + NYC (USA)	372 + 2,590 records	XGBoost / LightGBM	R^2 0.978; AUC 0.829	SHAP	Both + delay class

3. Methodology

3.1 Proposed Framework:

Figure 1 depicts the proposed framework. Two public data sources feed a common preparation stage (type coercion, missing- and low-coverage-column removal, outlier trimming, derivation of overrun targets from planned-versus-actual records, and one-hot encoding of categorical fields). The prepared data are split 80/20 into training and test partitions with a fixed seed, after which five model families are trained under an identical pipeline combining feature standardization, randomized hyperparameter search, and 5-fold cross-validation across four prediction tasks. Held-out evaluation with a shared metric set enables a like-for-like model comparison, and SHAP analysis of the best model per task converts predictive accuracy into interpretable drivers, which are finally synthesized into managerial insights. The framework is dataset-agnostic by design: any project registry containing planned and actual dates and budgets can be substituted at the input stage, which is what enables the transfer to Iraqi project data discussed in Section 5.

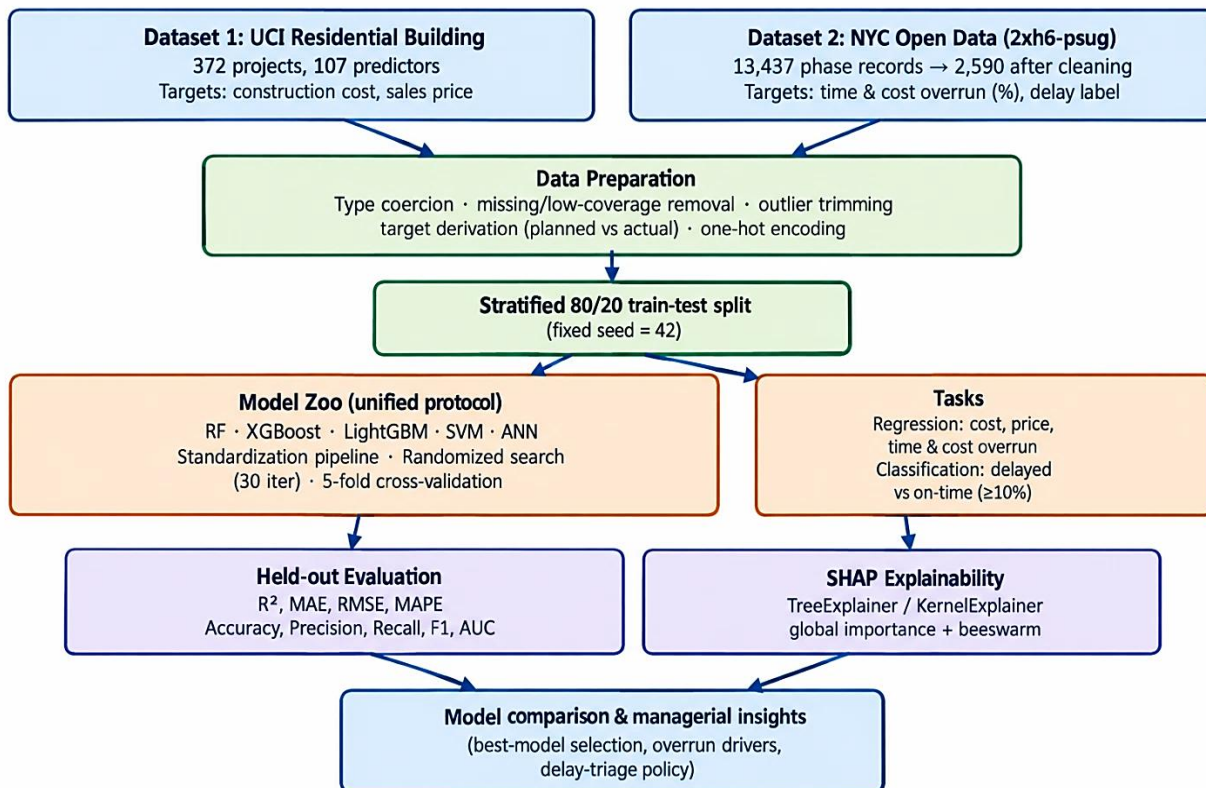


Figure (1): Proposed framework: data preparation, unified multi-model training, evaluation, and SHAP-based explainability

3.2 Datasets:

Two public datasets were used. Dataset 1 is the UCI Residential Building Data Set [14], comprising 372 residential building projects constructed in Tehran between 1993 and 2008. Predictors include eight project physical and financial variables (V-1 to V-8, e.g., floor area, lot area, preliminary estimated costs, construction duration, and unit price at project start) and nineteen macro-economic variables and indices, each at five time lags (95 variables), together with the completion year; the two targets are the actual construction cost (V-10) and the actual sales price (V-9). After type coercion and removal of low-coverage columns, 107 numeric predictors were retained for all 372 projects.

Dataset 2 was derived from the NYC Open Data portal (Capital Project Schedules and Budgets, School Construction Authority, dataset identifier 2xh6-psuq) [22], downloaded programmatically (13,437 raw project-phase records, 14 fields). Records with missing planned/actual dates or budgets, planned durations of 30 days or fewer, budgets below a nominal floor, or overruns outside the -90% to $+500\%$ range (data-entry noise) were removed, leaving 2,590 usable phase records. For each record, the percentage time overrun and percentage cost overrun were computed from planned-versus-actual schedule dates and budgeted-versus-actual costs. Only features at phase outset were retained: project type, phase name, status, geographic district (one hot encoded with rare categories collapsed), planned duration, budget, and start year/month. A binary delay label was created as those samples with time overrun greater than 10%, which resulted in a near-balanced classification problem (45.6% delayed).

3.3 Models and Training Protocol:

The following classification models were chosen for comparison and benchmarking: Random Forest, XGBoost, LightGBM, SVM (RBF kernel; SVR/SVC) and a Multilayer Perceptron ANN with early stopping. The features were standardized independently of each model that was put into a pipeline, which meant that no information leakage occurred during training phase. Hyperparameters were tuned by randomized search (30 iterations; 15 for classifiers) over standard grids (number of estimators, tree depth or leaves, learning rate, subsampling for boosters; C , γ , ϵ for SVM; hidden-layer sizes, L2 penalty, and learning rate for the ANN) with 5-fold cross-validation on an 80% training partition, optimizing mean absolute error for regression and F1-score for classification. Final performance was reported on the held-out 20% test partition (stratified for classification). Class imbalance safeguards (balanced class weights) were enabled where applicable. All experiments used a fixed random seed (42) for reproducibility.

3.4 Evaluation Metrics and Explainability:

Regression tasks were evaluated using the coefficient of determination (R^2), mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE); classification tasks using accuracy, precision, recall, F1-score, and the area under the ROC curve (AUC). The best model of each task was interpreted with SHAP [23]: Tree Explainer for tree

ensembles and Kernel Explainer with a background sample for kernel/neural models, producing global importance (mean |SHAP|) and beeswarm summaries. The full pipeline was implemented in Python 3 using scikit-learn, XGBoost, LightGBM, and SHAP, and the complete code is available from the authors for reproducibility.

4. Results and Discussion

4.1 Cost Prediction on the UCI Benchmark:

Table 2 summarizes model performance in predicting actual construction cost. XGBoost achieved the best performance (Test $R^2 = 0.978$, MAE = 16.40, MAPE = 8.74%), followed by LightGBM ($R^2 = 0.971$) and the ANN ($R^2 = 0.966$), while RF and SVM trailed at $R^2 = 0.930$ and 0.935. Cross-validated and held-out scores are closely aligned for all models (e.g., XGBoost CV $R^2 = 0.954 \pm 0.033$ versus Test $R^2 = 0.978$), indicating that hyperparameter search did not overfit the training partition. The achieved accuracy is higher than $R^2 = 0.96$ for XGBoost-based green building cost prediction [4] and 0.883 for CatBoost on 836 Jordanian projects [3] and supports both studies which showed that boosted tree ensembles outperform bagging ensembles and classical learners when applied to tabular construction data.

Table (2): Model comparison for construction cost prediction (UCI dataset). CV = 5-fold cross-validation on the training set; Test = held-out 20% test set

Model	CV R2 (mean±std)	CV MAE	CV RMSE	Test R2	Test MAE	Test RMSE	Test MAPE (%)
XGBoost	0.954 ± 0.033	17.55	32.61	0.978	16.40	25.03	8.74
LightGBM	0.926 ± 0.023	24.00	43.14	0.971	19.32	28.72	10.47
ANN (MLP)	0.930 ± 0.037	27.03	40.81	0.966	21.86	30.68	10.50
SVM	0.834 ± 0.083	30.73	63.85	0.935	26.25	42.83	13.72
Random Forest	0.934 ± 0.035	22.44	39.59	0.930	25.51	44.42	12.93

The same protocol applied to the secondary target (actual sales price) reinforced this pattern while exposing an instructive divergence (Table 3). Tree-based ensembles remained strong (XGBoost Test $R^2 = 0.978$, MAPE = 6.88%; RF 0.956; LightGBM 0.947), whereas SVM and the ANN degraded sharply ($R^2 = 0.393$ and 0.664). Remarkably, this instability is also present in cross-validation (CV $R^2 = 0.795 \pm 0.159$, which is an order of magnitude larger than XGBoost's CV $R^2 = 0.984 \pm 0.016$). The “problem variables” are the macro-economic variables that are used as lags in the equation, and their scale heterogeneity and multicollinearity is more pronounced and has more of an impact on the sales price, compared to the other variables. This scale heterogeneity and multicollinearity is addressed natively by gradient-boosted trees but is problematic for kernel and gradient-descent based learners even after standardization. In engineering management practice, where data sets frequently include physical and financial and economic variables on widely varying scales, boosted tree ensembles are thus the more conservative choice.

Table (3): Model comparison for sales price prediction (UCI dataset)

Model	CV R2 (mean±std)	CV MAE	CV RMSE	Test R2	Test MAE	Test RMSE	Test MAPE (%)
XGBoost	0.970 ± 0.016	87.49	193.86	0.978	106.13	198.92	6.88
Random Forest	0.952 ± 0.014	125.95	244.46	0.956	166.62	284.56	11.17
LightGBM	0.908 ± 0.040	165.35	340.72	0.947	173.95	312.45	15.24
ANN (MLP)	0.795 ± 0.159	319.98	473.25	0.664	508.93	785.53	37.27
SVM	0.353 ± 0.063	512.65	920.85	0.393	642.20	1055.89	49.67

The predicted-vs-actual scatter for XGBoost on the construction cost is shown in figure 2: predictions are close to the identity line over all of the range of construction cost, and only slightly dampened for a minority of very large projects, which is the typical behaviour of a tree ensemble that extrapolates close to the edges of the training distribution.

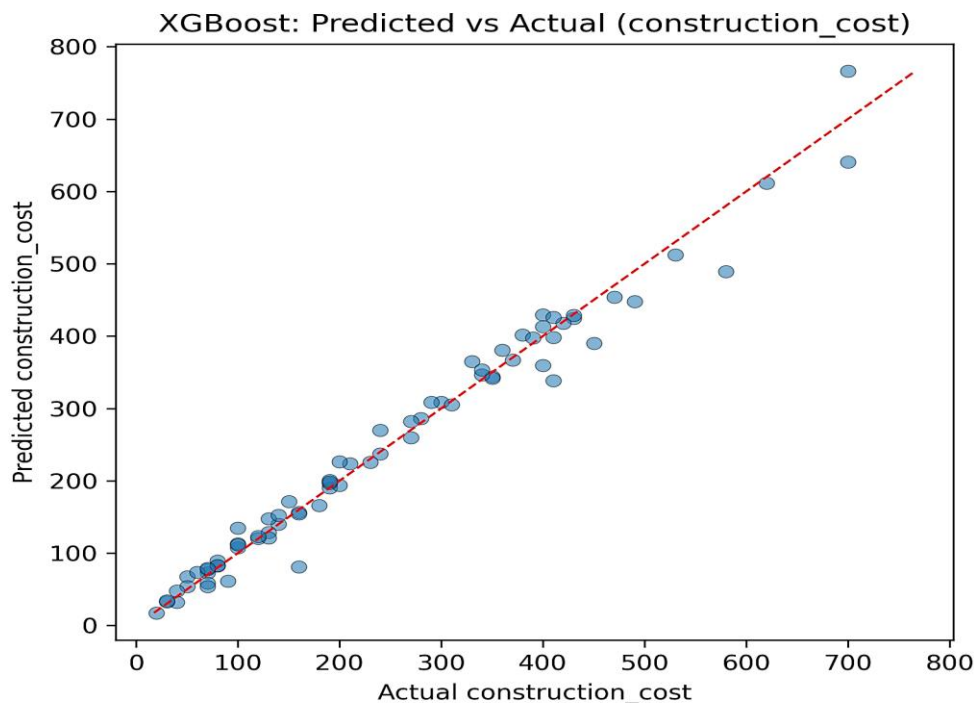


Figure (2): XGBoost predicted versus actual construction cost on the UCI held-out test set; the red dashed line is the identity line

Residuals as shown in Figure 2 are small and symmetric around the identity line for all projects in the low- and mid-cost range and there is no systematic bias; only two projects with the largest costs are slightly underestimated, as noted in the description of the boundary-extrapolation behavior of the tree ensembles above.

A similar scatter for the secondary target can be seen in Figure 3. Even with a wider range of prices

in sales (up to $\approx 6,300$ units), the predictions of XGBoost stay close to the identity line, visually confirming the low MAPE value of 6.88% reported in Table 3 and showing the robustness of the boosted ensemble to the more economically oriented target.

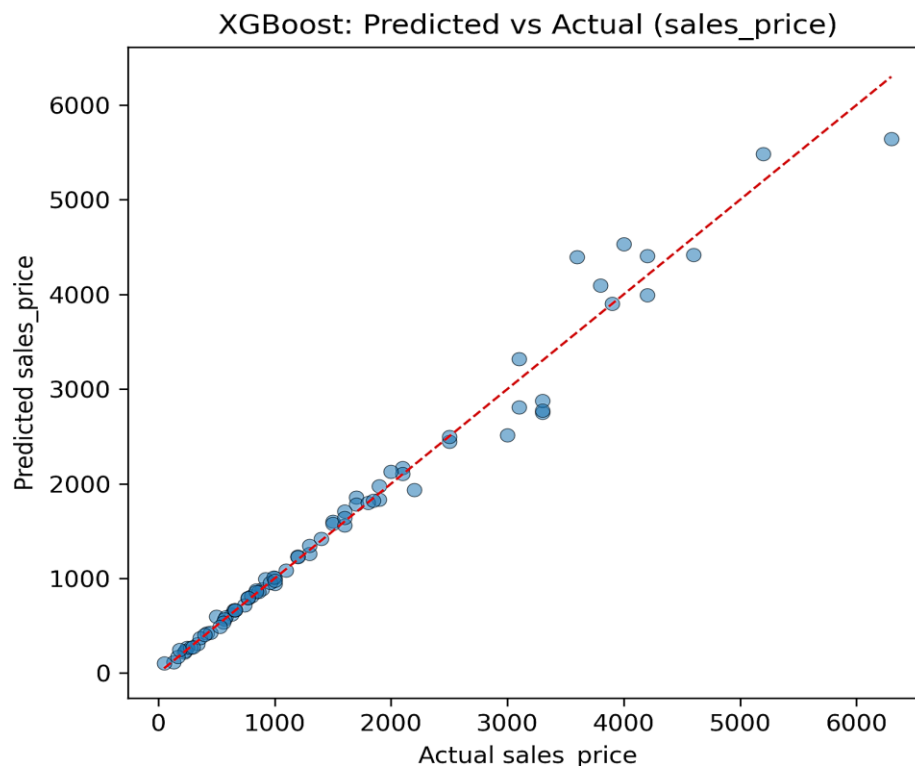


Figure (3): XGBoost predicted versus actual sales price on the UCI held-out test set

Figure 3 also helps explain why the weaker learners are dominated by the upper tail of the error metrics: a large number of observations fall in a small range below 2,000 units, where all models perform well, whereas the small number of observations in the high price region are exactly the ones where kernel-based and gradient-descent models are less accurate.

A cost model was built using XGBoost and the SHAP analysis of this model (Figures 4 and 5) indicate a very concentrated attribution structure. The initial estimated construction cost at project start (V-5) is the most important predictor by an order of magnitude (mean $|\text{SHAP}| \approx 116$ while the next feature, the duration of construction (V-7), is approximately 14), followed by the equivalent preliminary estimated cost in the base year (V-6) and the project completion year; lagged macro-economic index variables contribute smaller but systematic corrections. These effects can be seen in the beeswarm plot above and are both monotone and economically coherent: namely, higher preliminary estimates and longer durations increase the actual cost predicted. The managerial

interpretation is straight-forward: given a model of the relation between actual costs and the engineer's initial estimation as a function of time and the macro-economic evolution over the length of the construction, the model represents a data-driven correction layer on top of the conventional early-stage estimation. This is consistent with the SHAP results obtained by Coffie and Cudjoe [11] who found that the initial contract amount and initial duration were also the most important features, and the benefit of using black-box learners in combination with explainability tools over accuracy alone.

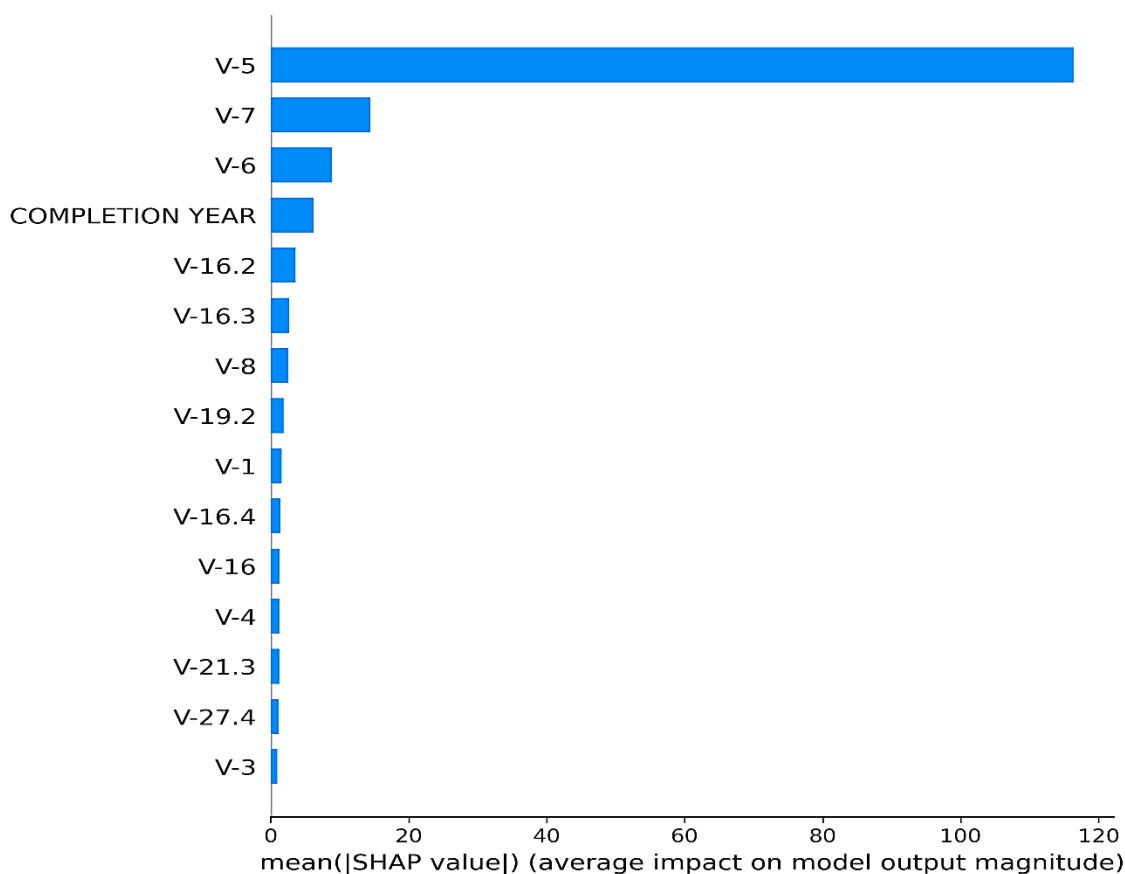


Figure (4): Global SHAP feature importance (mean |SHAP| value) for the XGBoost construction-cost model

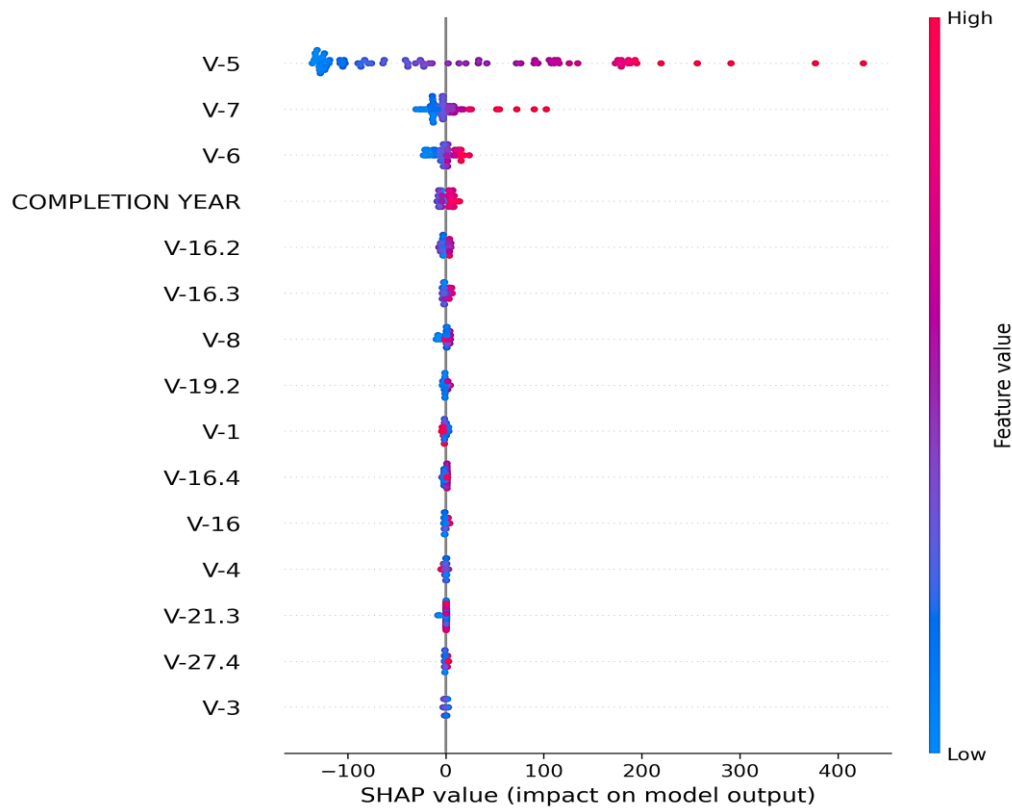


Figure (5): SHAP beeswarm summary for the XGBoost construction-cost model; each point is one test project, colored by feature value

Figures 4 and 5 together give a layer of transparency to raw accuracy measures that is unavailable: Figure 4 indicates how strongly the model relies on the preliminary estimate (V-5), and Figure 5 confirms that the direction of each of the larger effects is an economically sensible one; this is a necessary precondition for practitioner trust before such a model is deployed for estimating departments.

The dataset used is the NYC project, which is available online. The data set is the NYC project, available online.

The NYC experiments focus on the more difficult, more management-relevant problem of predicting overruns, or how "off" execution is from plans, from only administrative features that are available at phase outset. Absolute cost prediction (Table 4) was not as well established as regression of the exact overrun magnitude (Table 3): The best models had $R^2 = 0.162$ (RF) and $R^2 = 0.318$ (XGBoost) for the time and cost overruns, respectively, and all five models were within a small range. The model ranking is consistent with the UCI experiment, not the lack of the learning algorithms, but due to the limited feature set: overrun magnitudes are known to be influenced by

features not included in public administrative records, including change orders, payment delays, financial capacity of the contractor and scheduling realism.

Table (4): Overrun regression on the NYC data. CV = 5-fold cross validation; Test = held out 20% test set. MAPE will be excluded as actual overruns close to zero make it undefined

Model	Time CV R2 (mean±std)	Time Test R2	Time Test MAE	Time Test RMSE	Cost CV R2 (mean±std)	Cost Test R2	Cost Test MAE	Cost Test RMSE
Random Forest	0.206 ± 0.026	0.162	45.35	69.67	0.310 ± 0.038	0.303	39.52	60.22
XGBoost	0.164 ± 0.032	0.130	46.62	70.99	0.303 ± 0.031	0.318	39.28	59.56
LightGBM	0.158 ± 0.034	0.116	47.52	71.55	0.281 ± 0.036	0.311	38.97	59.89
SVM	0.047 ± 0.057	0.127	43.27	71.09	0.153 ± 0.009	0.188	41.19	65.02
ANN (MLP)	0.142 ± 0.014	0.093	52.69	72.46	0.246 ± 0.021	0.275	40.61	61.40

The regression behavior of the best models on each overrun target is shown in Figures 6 and 7, respectively. Both scatter have the typical signature of under-informative features, with the predictions compacting toward the sample mean in the dense central part of the scatter (between about -50% and +100%) and a marked flattening on the extremes (outside the range of about -50% to +100%).

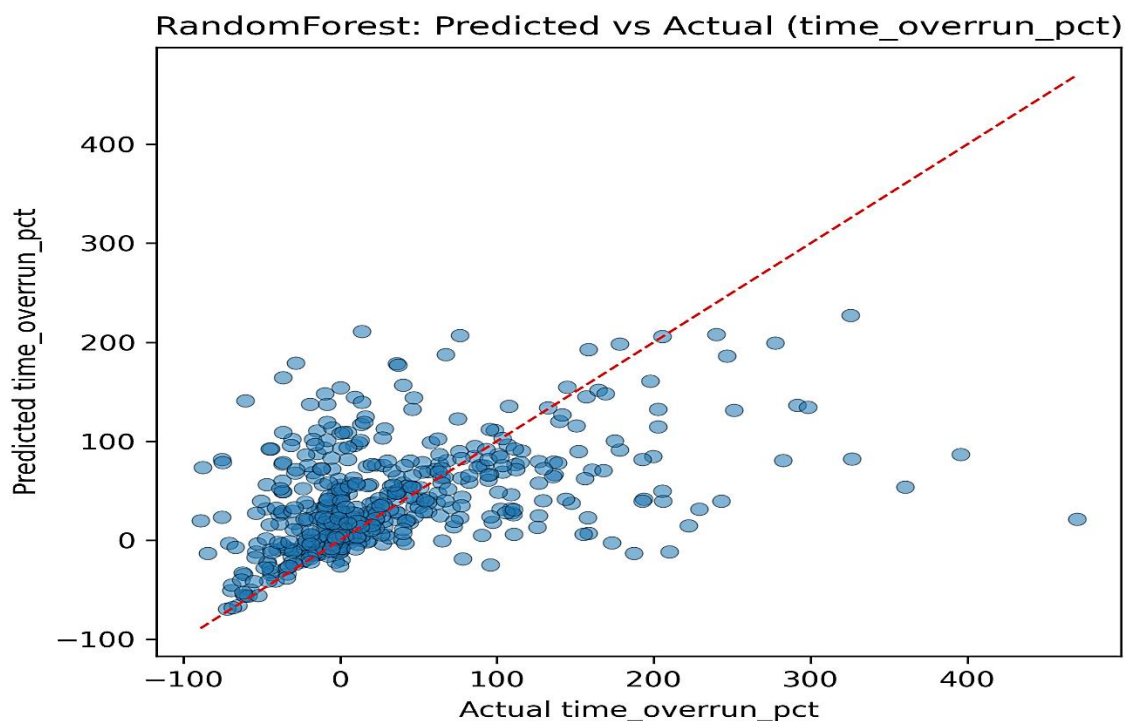


Figure (6): Random Forest predicted versus actual time overrun (%) on the NYC held-out test set

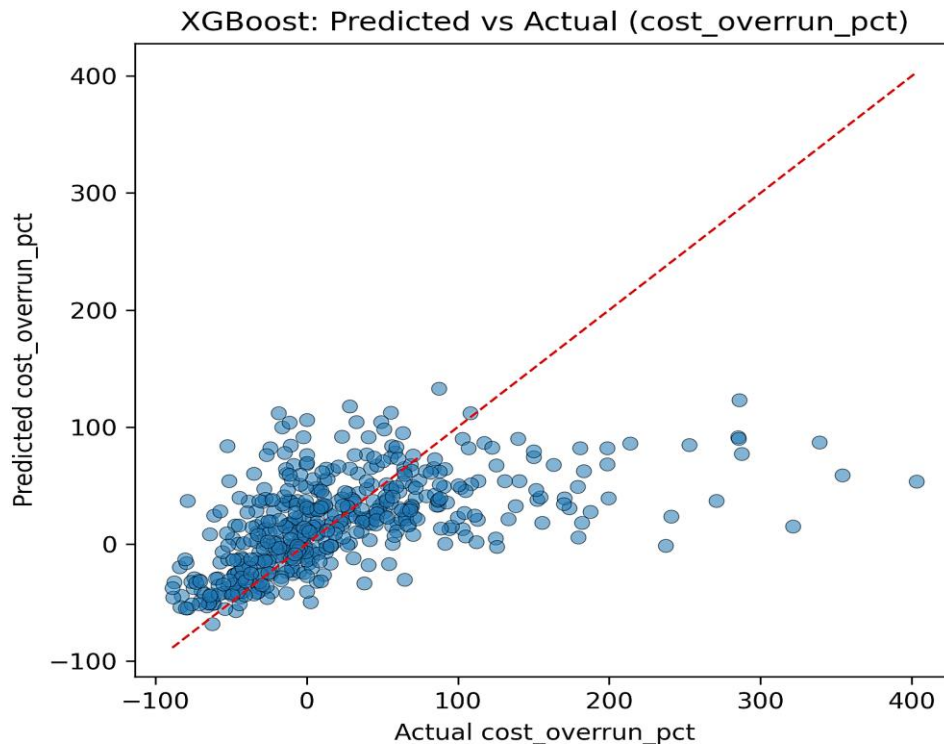


Figure (7): XGBoost predicted versus actual cost overrun (%) on the NYC held-out test set

The horizontal band of points at the far right of the scatter plot, representing points outside of the +100% actual overrun, are visible in both plots, indicating that the administrative feature set is simply not discriminatory enough to distinguish between moderate and extreme overruns — a region where change-order and payment-history variables are likely to have discriminative power. This inspires this reformulation to be studied next.

In stark contrast, rephrasing the problem as a delay-risk classification – will this phase be delayed by more than 10%?— yielded strong and practically useful discrimination (Table 5). LightGBM achieved the best F1-score (0.734, accuracy 74.9%) with AUC = 0.829, closely followed by XGBoost (F1 = 0.722, identical accuracy but a different operating point: higher precision, 0.728, at lower recall, 0.716) and RF (F1 = 0.721, AUC = 0.829). The choice between LightGBM and XGBoost is thus effectively a precision-recall policy choice rather than an accuracy difference. These results compare favorably with prior delay classifiers built on richer, questionnaire-derived features: 74.5–78.4% accuracy in Egypt [5] and 73.5–79.4% in Qatar [6]. Achieving comparable discrimination from routinely collected administrative records alone — on a nearly balanced sample of 2,590 real project phases — indicates that useful early-warning delay screening does not require bespoke data collection.

Table (5): Delay classification on the NYC dataset (delayed = time overrun > 10%; class balance 45.6%; held-out test set)

Model	Accuracy	Precision	Recall	F1-score	AUC
LightGBM	0.749	0.710	0.758	0.734	0.829
XGBoost	0.749	0.728	0.716	0.722	0.813
Random Forest	0.737	0.698	0.746	0.721	0.829
SVM	0.716	0.667	0.754	0.708	0.774
ANN (MLP)	0.695	0.699	0.581	0.634	0.771

The confusion matrix of the best classifier is shown in Figure 8. Of 518 test phases, LightGBM correctly identified 209 of 282 on-time phases and 179 of 236 delayed phases, corresponding to an accuracy of 74.9%, precision of 71.0%, and recall of 75.8% for the delayed class. Errors are moderately balanced between false alarms (73) and misses (57), with the classifier slightly favoring recall — a desirable property for an early-warning screening tool, where the cost of missing a genuinely at-risk phase exceeds the cost of an unnecessary managerial review.

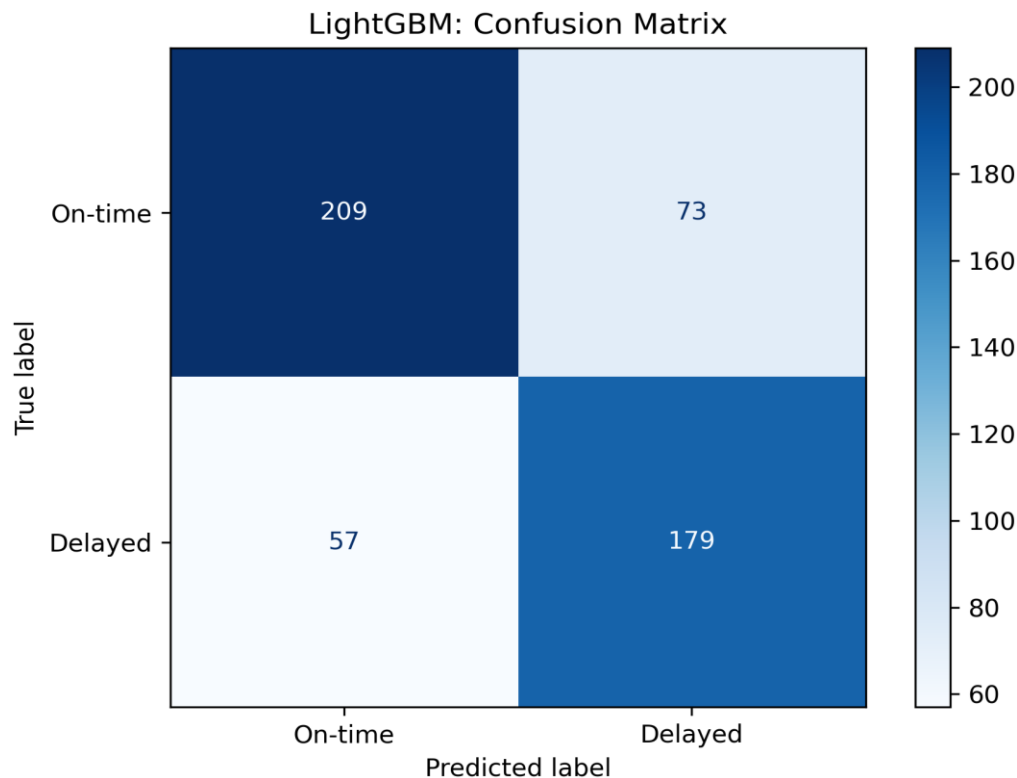


Figure (8): Confusion matrix of the LightGBM delay classifier on the NYC held-out test set (delayed = time overrun > 10%)

To expose what drives the NYC overrun models, Figures 9 and 10 present the SHAP beeswarm summaries for the best time-overrun and cost-overrun regressors, respectively. The most influential feature for time overrun is the planned duration of the phase, followed by the start year, budget and start month, with the phase-type indicators contributing only slightly; longer planned durations tend to cause the time overrun to be larger, while older start years tend to cause the time overrun to be smaller, partially because of an actual improvement trend and partially because of right-censoring of recently started phases. In the case of cost overrun (Figure 20), the budget itself has a large influence with a very clear monotone pattern: small-budget phases give large positive SHAP values and large-budget phases give negative SHAP values, in keeping with the well-documented base-size effect of a decreasing percentage overrun as the budget increases. The scope phase indicator has a positive contribution for cost overrun and a negative for time overrun; that is, the phases in this portfolio that are at the beginning tend to overspend and the phases at the end tend to be under budget.

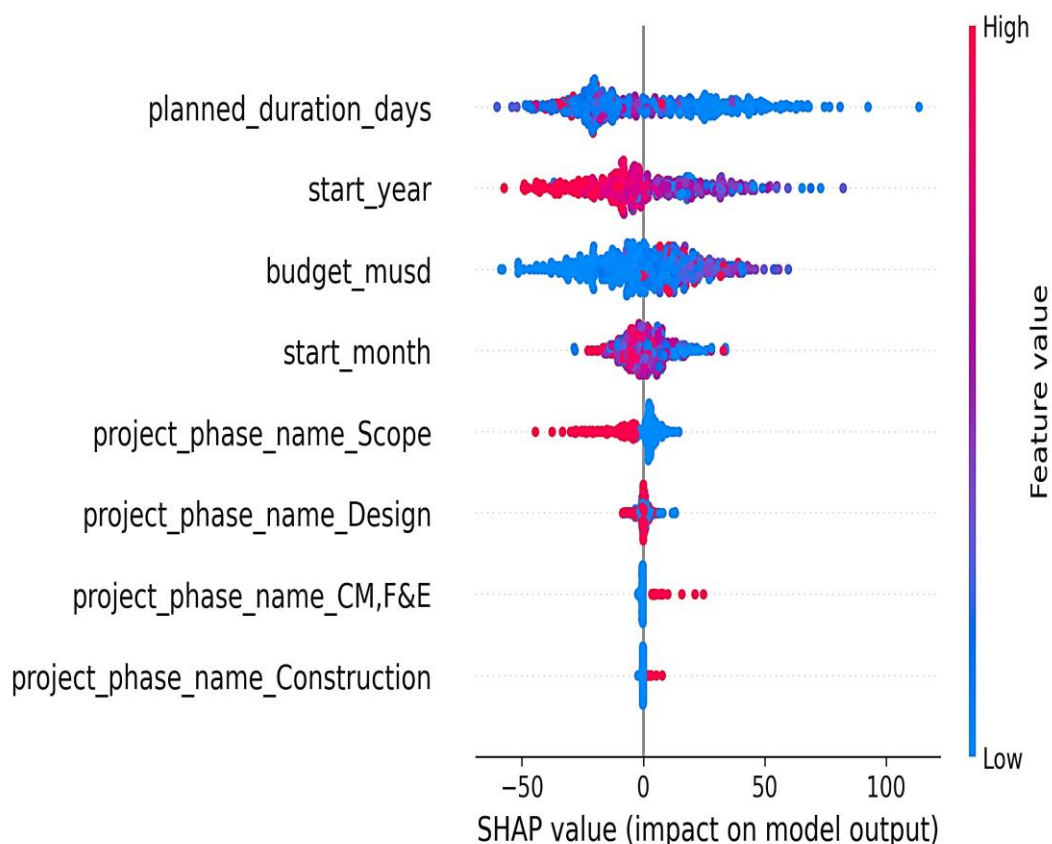


Figure (9): SHAP beeswarm summary for the Random Forest time-overrun model (NYC dataset)

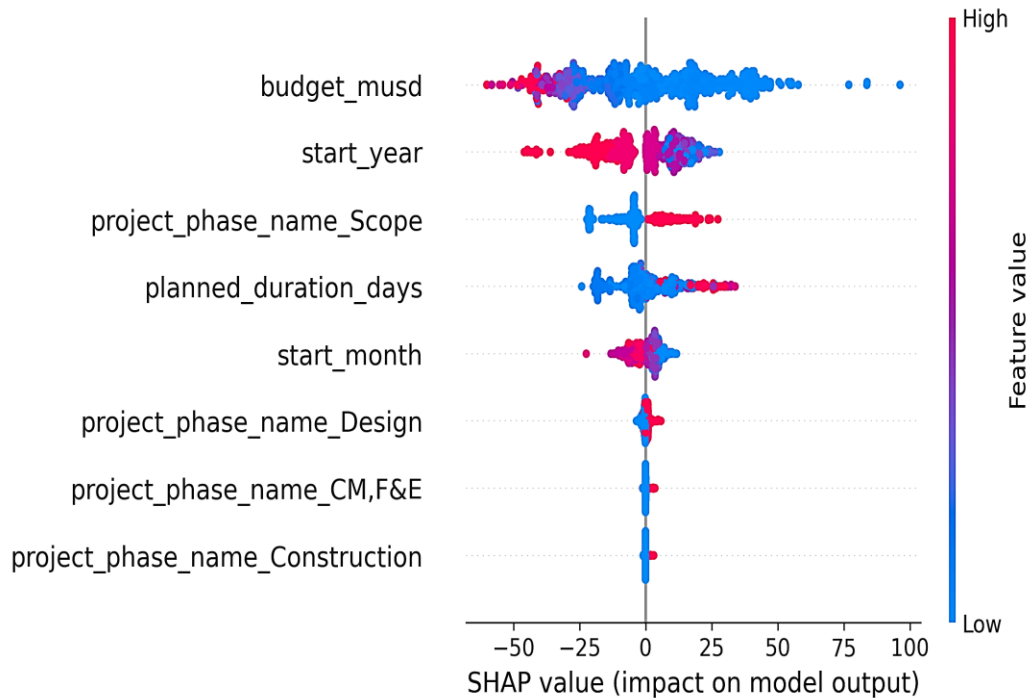


Figure (10): SHAP beeswarm summary for the XGBoost cost-overrun model (NYC dataset)

Together, Figures 9 and 10 substantiate the managerial reading of Section 4.3: the drivers available in administrative records (budget scale, timing, phase type) shape overrun risk in coherent, explainable ways, but the largest overruns are governed by information those records do not contain.

4.3 Discussion:

Two of these findings come from the comparison of the two sets of data. First, gradient-boosted trees ensembles consistently placed at the top performance level on various datasets, targets and problem formulations, reinforcing the observation made in recent literature [3, 4, 7] and developing it to a unified protocol where the three baselines (bagging, kernel and neural) are tuned equally. Second, the difference in robustness between tree ensembles and SVM/ANN grows as the features become more heterogeneous, as in practice mixed physical-financial-economic feature sets are common. Thirdly, and most significantly, the tractability of overrun prediction is strictly problem dependent: point prediction of overrun size is poor ($R^2 \leq 0.32$) if only administrative features are included while the binary version of delay-risk screening is good ($AUC \approx 0.83$). For project-management offices, machine learning can already be used effectively to prioritize and flag high-risk project phases at the beginning, but to quantify the amount of overrun that one can expect requires enriching the data pipelines with additional variables, such as change order data and payment and contractor-capacity data, as suggested by the feature-importance results of [3] and [13].

These results also point to an implication for developing contexts, like Iraq, as follows: the NYC experiment shows that even a small number of administrative fields (planned/actual dates, budgets, project type) can be systematically archived across public projects to enable delay-risk screening, which can then serve as a self-breeding ground for this process. Comprehensively implementing such archiving in Iraqi ministries and provincial engineering directorates would help in providing locally calibrated versions of the present framework that would overcome the small-sample and questionnaire-dependence limitations of previous studies conducted in Iraq [2, 19–21].

4.4 Limitations:

These findings are limited by four factors. First, this generalization of UCI to other contexts should be tested because the UCI data set was collected from one city and time (Tehran, 1993-2008), and the NYC data set was collected from one agency type (school construction). Second, the overrun targets derived from NYC are subject to any noise in the agency reported planned dates and budgets, which outlier trimming reduces but doesn't remove. Third, the extent of the administrative feature set is limited by the presence of known overrun drivers (change orders, payment delays, contractor characteristics), which are not included. Fourth, the 10%-delay-threshold that provides a well-balanced set of classes is a policy choice, so is the sensitivity of classification performance to the threshold.

5. Conclusions and Future Work

This research compared five machine learning models using the same equally optimized protocol for four prediction tasks with two public construction datasets, along with explainability using SHAP. When evaluated on the UCI benchmark, XGBoost achieved state-of-the-art construction cost prediction ($R^2 = 0.978$, MAPE = 8.74%) and SHAP showed that it is a data-driven correction mechanism for initial engineering estimates, primarily based on the preliminary cost estimate, construction duration, and macro-economic trend. The discrimination performance of delay-risk classification, using only administrative features, was strong ($F1 = 0.734$, $AUC = 0.829$) on real NYC project-phase records, and the discrimination performance of point regression of overrun magnitude was weak, indicating that problem formulation, not algorithm choice, is the limiting factor under limited features. Future work will continue to be done in three directions: (1) building and applying the framework to a dedicated Iraq-based construction-project dataset, with the data collection instrument, developed along this study, containing joint time-and-cost targets; (2) enriching the features with additional change order, payment and contractor variables to reduce the gap in the regression model; and (3) extending the model zoo to include stacking/voting ensembles and to quantify uncertainty, as well as the explainability layer with the SHAP weighted multi-criteria decision analysis for risk prioritization.

Data availability:

Both datasets are publicly available: the UCI Residential Building Data Set

(<https://doi.org/10.24432/C5S896>) and the NYC Open Data Capital Project Schedules and Budgets dataset (identifier 2xh6-psuq). The complete analysis code, including automatic dataset download, the unified training protocol, and all figure-generation scripts, is openly available at <https://github.com/mustafa-r-alsaadi/ML-Based-Prediction-.git> to support full reproducibility of the reported results.

References

1. Office of the New York State Comptroller, "Better Reporting and Monitoring of NYC Capital Projects Needed," Albany, NY, USA, Apr. 2024.
2. Z. M. Yaseen, Z. H. Ali, S. Q. Salih, and N. Al-Ansari, "Prediction of risk delay in construction projects using a hybrid artificial intelligence model," *Sustainability*, vol. 12, no. 4, p. 1514, 2020, doi: 10.3390/su12041514.
3. R. Hamdan et al., "Predicting cost overrun in construction projects using machine learning algorithms: the case of Jordan," *Engineering, Construction and Architectural Management*, vol. 33, no. 6, 2025, doi: 10.1108/ecam-09-2024-1209.
4. O. Alshboul, A. Shehadeh, G. Almasabha, and A. S. Almuflih, "Extreme gradient boosting-based machine learning approach for green building cost prediction," *Sustainability*, vol. 14, no. 11, p. 6651, 2022, doi: 10.3390/su14116651.
5. A. Gondia, A. Siam, W. El-Dakhkhni, and A. H. Nassar, "Machine learning algorithms for construction projects delay risk prediction," *Journal of Construction Engineering and Management*, vol. 146, no. 1, p. 04019085, 2020, doi: 10.1061/(ASCE)CO.1943-7862.0001736.
6. A. Asadi, M. Alsubaey, and C. Makatsoris, "A machine learning approach for predicting delays in construction logistics," *International Journal of Advanced Logistics*, vol. 4, no. 2, pp. 115–130, 2015.
7. S. Alsulamy, "Predicting construction delay risks in Saudi Arabian projects: A comparative analysis of CatBoost, XGBoost, and LGBM," *Expert Systems with Applications*, vol. 268, p. 126268, 2025, doi: 10.1016/j.eswa.2024.126268.
8. F. Mostofi et al., "Cost overrun risk assessment and prediction in construction projects: A Bayesian network classifier approach," *Buildings*, vol. 12, no. 10, p. 1660, 2022, doi: 10.3390/buildings12101660.
9. R. Al mnaseer, S. Al-Smadi, and H. Al-Bdour, "Machine learning-aided time and cost overrun prediction in construction projects: application of artificial neural network," *Asian Journal of Civil Engineering*, vol. 24, no. 7, pp. 2583–2593, 2023, doi: 10.1007/s42107-023-00665-7.
10. C. Emmanuel et al., "Explainable Artificial Intelligence (XAI): Precepts, methods, and opportunities for research in construction," *arXiv: 2211.06579*, 2022.

11. G. Coffie and S. Cudjoe, "Using extreme gradient boosting (XGBoost) machine learning to predict construction cost overruns," *International Journal of Construction Management*, vol. 24, no. 16, pp. 1742–1750, 2024, doi: 10.1080/15623599.2023.2289754.
12. Y. Tian, Y. Chen, and Y. Ning, "Identifying and predicting delay risks in prefabricated construction: an explainable ensemble learning approach," *Advanced Engineering Informatics*, 2026, doi: 10.1016/j.aei.2025.104219.
13. R. Shyam and S. Tiwari, "Predictive assessment of construction delays on schedule, cost, and quality performance using machine learning and multi-criteria decision analysis," *Journal of The Institution of Engineers (India): Series A*, 2026, doi: 10.1007/s40030-026-00989-y.
14. M. Rafiei, "Residential Building [Dataset]," *UCI Machine Learning Repository*, 2015, doi: 10.24432/C5S896.
15. M. H. Rafiei and H. Adeli, "Novel machine-learning model for estimating construction costs considering economic variables and indexes," *Journal of Construction Engineering and Management*, vol. 144, no. 12, 2018, doi: 10.1061/(ASCE)CO.1943-7862.0001570.
16. A. El-Kholy, "Exploring the best ANN model based on four paradigms to predict delay and cost overrun percentages of highway projects," *International Journal of Construction Management*, vol. 21, no. 7, pp. 694–712, 2021, doi: 10.1080/15623599.2019.1580001.
17. F. Zawaideh, "BIM-integrated explainable machine learning for cost, schedule, and carbon risk prediction in high-rise building construction," *Asian Journal of Civil Engineering*, 2026, doi: 10.1007/s42107-026-01761-0.
18. "Transparent and reliable construction cost prediction using advanced machine learning and explainable AI," 2025 (Science Direct, in press).
19. Z. H. Ali et al., "Developing an integrative data intelligence model for construction cost estimation," *Complexity*, vol. 2022, p. 4285328, 2022, doi: 10.1155/2022/4285328.
20. R. N. Zehawi, "Prediction and estimation of highway construction cost using machine learning," *Engineering, Technology & Applied Science Research*, vol. 14, no. 5, pp. 17222–17231, 2024, doi: 10.48084/etasr.8285.
21. B. A. Kamil, "An empirical evaluation of the performance of deep neural networks on delay risk prediction in urban flexible pavement projects in Iraq," *Engineering, Technology & Applied Science Research*, vol. 15, no. 5, pp. 28343–28349, 2025, doi: 10.48084/etasr.13781.
22. NYC Open Data, "Capital Project Schedules and Budgets," dataset 2xh6-psuq, City of New York.
23. S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems 30 (NeurIPS)*, 2017, pp. 4765–4774.