International Journal of Sustainable Applied Science and Engineering (IJSASE)

Contents lists available at IJSASE
International Journal of Sustainable Applied Science and Engineering
journal homepage: IJSASE
Volume 2, No. 2, 2025

Application of Machine Learning and Data Science in Project Construction Scheduling

Fatemeh Ghasemi [a*], Hamidreza Keihani [b]

[a] *Department of Industrial Engineering, Alzahra University, Tehran, Iran*,
[b] *Department of Social Sciences, Business, Economics and Law, Åbo Akademi University, Turku, Finland*.

ARTICLE INFO

Received: 2025/07/01

Revised: 2025/07/25

Accept: 2025/08/27

Keywords:

Project Construction Scheduling, Machine Learning, Data Science, Predictive Analytics, Optimization Models

ABSTRACT

Construction scheduling is central to project success, but it remains challenging due to uncertain activity durations, resource interactions, supply chain variability, and frequent design or scope changes. Recent years (2020–2025) have seen rapid advances in machine learning (ML) and data science (DS) methods—ranging from gradient-boosted trees and deep sequence models to computer-vision pipelines—that augment or replace classical schedule-engineering techniques (CPM/PERT) by predicting activity durations and delay risk, automating progress measurement, and enabling dynamic, data-driven rescheduling. This paper (1) synthesizes literature from 2020–2025 on ML/DS applications in construction scheduling, identifying major thematic strands and research gaps; (2) proposes an end-to-end methodology combining supervised duration/delay models with a bi-objective resource-constrained project scheduling problem (RCPSP) that integrates ML predictions and uncertainty buffers; (3) demonstrates the approach on a reproducible synthetic dataset (temporal train/test split) and shows model performance (regression and classification) and schedule impacts; and (4) discusses implications, interpretability, and research directions. On the synthetic test set, a gradient boosting regressor reduced the mean absolute error (MAE) by approximately 16.6% compared to a coarse PERT-like baseline. A gradient boosting classifier for delay risk achieved an ROC-AUC ≈of approximately 0.98 and an F1 ≈score of approximately 0.89. When ML-predicted durations (risk-aware) were used in an illustrative RCPSP network, the planned makespan increased relative to optimistic PERT estimates—illustrating a trade-off between realism and nominal makespan. The review highlights recurring challenges: lack of open cross-project benchmarks, limited closed-loop rescheduling demonstrations, domain-shift and transfer issues, and a need for explainability and human-in-the-loop interfaces.

[*] Corresponding author email address: Qasemi.fatemeh@gmail.com (Fatemeh Ghasemi).
DOI: https://doi.org/10.22034/ijsase.v2i2.171
Available online 08/28/2025

3060-6349 /BGSA Ltd.

1. Introduction

Construction project scheduling determines resource allocation, planning, cost, and contract outcomes across the project lifecycle. Traditional scheduling techniques, such as the Critical Path Method (CPM) and Program Evaluation and Review Technique (PERT), as well as heuristic resource-leveling methods, are well-established. Still, they assume stationary or simplistic duration distributions and often fail to account for complex, nonlinear interactions between crew productivity, supply delays, weather, and on-site disruptions. As construction projects become more data-rich (BIM, IoT sensors, site imagery, Last Planner System metrics), Machine Learning (ML) and Data Science (DS) techniques offer an opportunity to model these nonlinear dependencies, provide probabilistic estimates of activity durations and delay risk, automate progress measurement, and feed predictions into optimization/rescheduling engines to produce more robust and actionable schedules [1-6],[10-15] (see Figure 1).

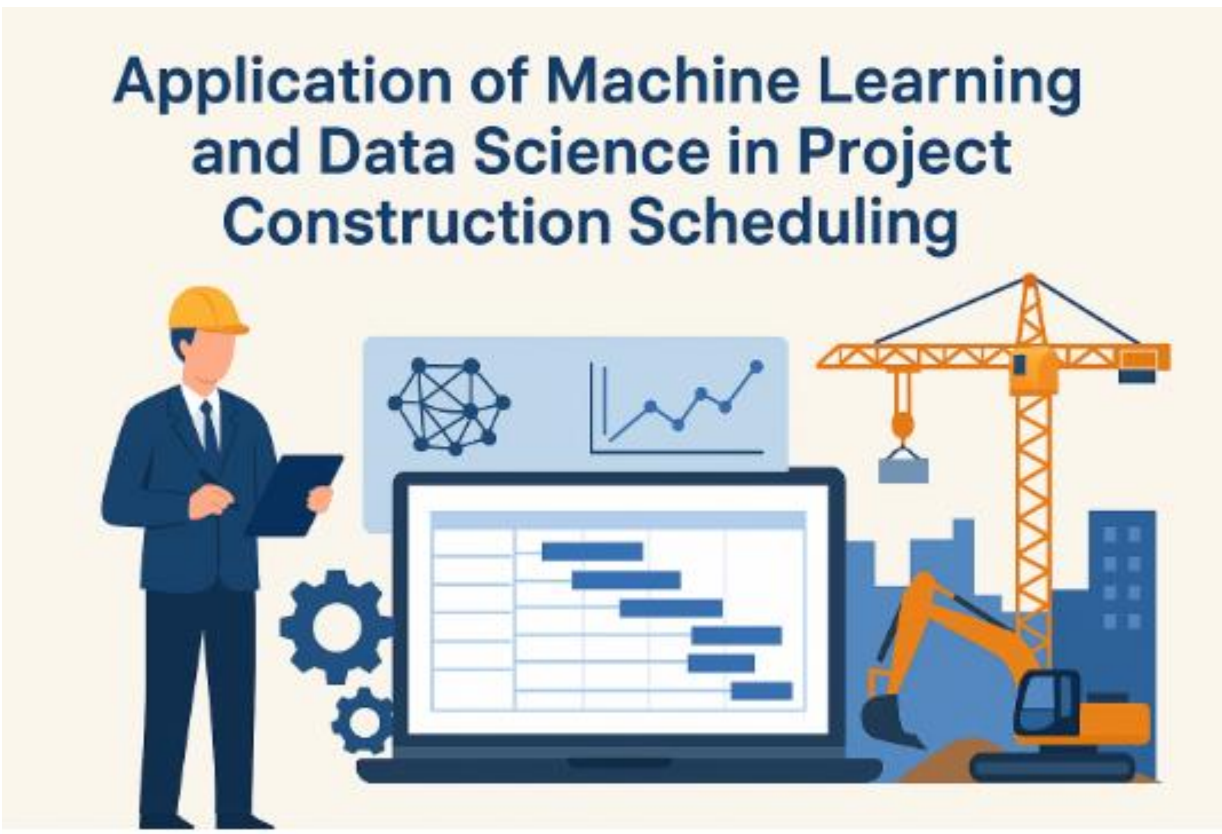

Figure 1. Application of machine learning and data science in project construction scheduling

Recent literature (2020–2025) shows converging trends: (i) automated schedule generation by mining BIM/IFC datasets and learning task templates; (ii) predictive models for activity duration and project-level delay risk using tree ensembles and sequence models (LSTM/BiLSTM); (iii) computer-vision pipelines that convert imagery into progress measures (percent complete / installed quantities) that update schedules near real time; and (iv) integrated BIM+AI frameworks that couple predictions with optimization for dynamic rescheduling [1],[2],[3],[4],[5],[6]. Empirical comparisons commonly find tree-ensemble methods (e.g., gradient boosting) and recurrent models outperform linear baselines where nonlinearities and temporal dependencies matter, but practical adoption is limited by fragmented datasets, lack of benchmark corpora, transferability concerns, and explainability needs for planners and stakeholders [6],[7],[8].

This work synthesizes the 2020–2025 literature, identifies an explicit set of research gaps, and presents an end-to-end methodology (predictive models → bi-objective RCPSP) validated on a reproducible, construction-plausible synthetic dataset. Numerical results illustrate the predictive performance and schedule impacts, highlighting trade-offs—especially how risk-aware ML estimates can increase planned makespan while improving realism and reducing downstream slippage risk. The remainder of the paper is organized as follows: Section 2 reviews related works (2020–2025) and identifies gaps; Section 3 details the methodology and mathematical formulation; Section 4 presents numerical results (tables/figures, and downloadable artifacts); Section 5 concludes and outlines future research directions.

2. Literature review

2.1 Overview of application areas

From 2020 to 2025, four main application areas emerged where machine learning (ML) and data science (DS) intersect with construction scheduling. The first area is automated schedule generation from BIM/IFC data, where researchers used pattern mining of prior projects to extract reusable task templates and precedence rules [1]. The second major area is predictive duration and delay modeling, which relies on regression and classification models trained on historical activity features, crew attributes, material lead times, Last Planner System (LPS) metrics, and environmental or time-series inputs. Popular algorithms in this domain include gradient boosting machines and recurrent neural networks [2],[6],[8],[9]. The third research stream involves vision-driven progress monitoring, in which computer vision pipelines are applied to generate percent-complete and installed-quantity estimates that directly inform schedule updates [3], [4]. Finally, the fourth area concerns BIM and AI-based dynamic rescheduling, where predictive models support rescheduling decisions under resource constraints and time–cost trade-offs through end-to-end integrated systems [5].

Representative studies within these domains are summarized in the literature review table below. This table highlights works published between 2020 and 2025 that directly map ML and DS techniques to scheduling tasks or schedule inputs, while also noting their core limitations.

Table 1. Representative studies

Ref.	Data / Context	ML/DS method	Scheduling task / use	Key finding	Limitation
González-Quevedo et al. [1]	BIM/IFC corpora	Supervised + pattern mining	Auto schedule generation	Learned task templates from BIM aided schedule construction	Needs large cross-project corpora
Marhani et al. [2]	LPS metrics & execution logs	ML regression/classification	Early schedule performance prediction	LPS features predictive of schedule adherence	Reproducibility & dataset access
Wang et al. [3]	Site imagery	CNN-based vision	Progress → schedule updates	Automated WIP recognition reduces update latency	Domain shift (lighting/occlusion)
Liu et al. [4]	Prefab building imagery	Detector + pipeline	Visual progress monitoring	Accurate progress quantification within scope	Narrow building/component scope
Parnian et al. [5]	BIM + project data	ML + optimization	Dynamic rescheduling framework	Integrates predictions into rescheduling loop	Limited field validation
Agyekum et al. [6]	Mixed project datasets	Comparative ML study	Delay prediction	Tree ensembles outperform linear models	External validity limited
Al-Douri et al. [7]	Scientometric review	Review	Mapping DL in construction	Identified dominant DL application areas	Not schedule-metric specific
Al-Jubouri & Al-Khafaji [8]	Mixed factors	LSTM + NN hybrid	Duration & timing prediction	RNNs capture temporal drivers	Small datasets & feature drift

Ref.	Data / Context	ML/DS method	Scheduling task / use	Key finding	Limitation
Abdulsattar & Ali [9]	Housing units' data	LSTM	Delay time prediction	Good sequential capture of delays	Domain-specific limits
Braun et al. [10]	Early DL studies	Review	Various tasks incl. progress	DL enables fine progress measurement	Early-stage work

Recent studies reveal several important insights at the intersection of machine learning and construction scheduling. In terms of algorithms, gradient boosting and other tree ensemble methods have emerged as robust baselines for duration regression and delay classification. At the same time, deep sequence models demonstrate particular effectiveness when temporal signals—such as Last Planner System (LPS) sequences or repeated schedule updates—are present [6],[8],[9]. Regarding data sources, the integration of BIM/IFC data, LPS records, IoT or sensor feeds, and site imagery provides richer and more informative feature sets. However, this integration also introduces significant challenges related to data alignment and interoperability [1],[3],[5].

When it comes to evaluation practices, most studies rely on project-level holdout validation or cross-validation, yet very few establish standardized cross-project benchmarks. This limitation often results in overly optimistic performance reports due to data leakage or the reliance on single-project validation [6]. Finally, explainability and adoption remain central concerns: industry practitioners consistently emphasize the need for interpretable outputs—such as feature importance rankings, partial dependence plots, and uncertainty intervals—before they can reliably trust and act upon ML-informed scheduling decisions [6],[7], [16-20].

Despite these advances, several research gaps remain evident in the literature. The first is the scarcity of benchmarks and the lack of reproducibility. There are very few openly available cross-project datasets with standardized splits, and reproducible experiments with shared code and data are still rare [1],[2],[6]. A second gap involves the lack of closed-loop integration. While most studies focus only on prediction, far fewer demonstrate full end-to-end control workflows that connect prediction to optimization and incorporate field feedback through human-in-the-loop validation [3],[5], [21-25].

A third gap concerns explainability and decision support. Advanced techniques such as SHAP values, permutation importance, or uncertainty quantification have seen limited adoption, even though they could help planners better interpret and trust model outputs [6],[7]. Fourth, issues of domain shift and transfer learning remain unresolved. Models trained for one type of building system or delivery method often degrade significantly when applied to other contexts, and only a few studies explore transfer learning or domain adaptation techniques to address this [3],[4].

The fifth research gap lies in the fusion of real-time data and latency. Although integrating LPS, BIM, computer vision, weather, and supply-chain data streams in real time would enable rolling rescheduling, this remains an underdeveloped capability [2],[5]. Finally, there is limited emphasis on outcome-level evaluation. Most studies report predictive accuracy metrics but rarely assess downstream schedule outcomes such as improvements in total completion time, buffer consumption, or realized delay reduction [1],[5].

3. Methodology

3.1 Pipeline overview

The proposed pipeline for applying machine learning in project construction scheduling is structured into three stages.

The first stage is data ingestion and feature engineering, where both static and dynamic features are collected. Static features include trade, complexity, crew size, subcontractor rating, design maturity, typical productivity, material lead time, and equipment availability. Dynamic features capture time-varying aspects such as weekly Last Planner System (LPS) percent plan complete (PPC), constraint removal rate, weather index, site congestion, and vision-derived percent complete. All features are aligned to activity identifiers, such as BIM elements, WBS codes, or schedule task IDs, ensuring consistency across datasets.

The second stage is predictive modeling, where supervised machine learning models are trained to forecast activity durations (treated as a continuous regression task) and delay risks (treated as a binary classification problem). To improve reliability, uncertainty estimates are generated using techniques such as quantile regression or empirical residual quantiles. Interpretability outputs, such as permutation importance and partial dependence plots, are also provided to support planner decision-making. Models are validated using temporal or project hold-out splits to avoid data leakage.

The third stage involves schedule optimization and rescheduling. Here, the predicted durations and uncertainty buffers are incorporated into a resource-constrained project scheduling problem (RCPSP). The optimization process is bi-objective, balancing makespan and cost using scalarization or ε-constraint methods. To ensure adaptability, rolling horizon rescheduling is applied as new observations and real-time data become available.

3.2 Data & features

The dataset integrates both static and dynamic variables. Static features include trade (categorical), complexity index, crew size, subcontractor rating, design maturity, typical productivity, material lead time (days), equipment availability, and historical rework probability. Dynamic features, which vary over time or every week, include LPS PPC, constraint removal rate, weather index, site congestion, vision-derived percent complete, and supply chain lags. The predictive targets are defined as the actual activity duration (in days) for regression tasks and a delay flag (1 if the activity finishes later than the baseline threshold) for classification tasks.

3.3 Predictive models, training & validation

For predicting activity durations, regression approaches are applied. Baseline models include linear regression and PERT-style three-point estimates. In contrast, more robust models include Random Forest (RF), Gradient Boosting Regression (GBR), and sequence-based models, such as LSTMs, when temporal patterns are present. Model performance is evaluated using metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and R^2.

For classifying delay risk, Logistic Regression serves as a baseline, while tree ensembles such as Gradient Boosting Machines (GBM) provide stronger predictive performance. Probability outputs are calibrated to improve reliability. The evaluation metrics include ROC-AUC, Precision, Recall, and F1-score.

Validation strategies emphasize temporal splits by project, training on older projects and testing on newer ones to emulate deployment conditions better. When datasets are sufficiently large, cross-validation is also applied across multiple projects. In addition, monitoring for model drift and recalibration over time ensures the models remain effective as project conditions evolve [6], [8].

3.4 Optimization model (mathematical formulation)

Let:

Decision variables:

- **Activity start times**:

$$S_i \geq 0 \quad \forall i \in \mathcal{N} \tag{1}$$

where S*i* represents the scheduled start time of activity *i*.

- **Mode selection variables**:

$$x_{im} \in \{0,1\}, \quad \sum_{m \in \mathcal{M}_i} x_{im} = 1 \quad \forall i \in \mathcal{N} \tag{2}$$

where x_{im} indicates whether activity *i* is executed in mode *m*, and exactly one mode must be selected from the feasible set M_i .

- **Disjunctive ordering variables (resource conflicts)**:

$$y_{ij} \in \{0,1\} \quad \forall (i,j) \in \mathcal{C} \tag{3}$$

where $y_{ij} = 1$ if activity iii precedes activity jjj in cases where both require the same non-shareable resource, and $y_{ij} + y_{ji} = 1$.

Bi-objective optimization (makespan and cost):

Minimize objectives:

$$\min\,(f_1, f_2) = \Big(\max_{i \in \mathcal{N}}\{S_i + \sum_m x_{im}(\hat{d}_{im} + \beta_i)\},\ \sum_{i,m} x_{im}c_{im} + C_{\text{overtime}} + C_{\text{delay}}\Big) \tag{4}$$

Subject to:

1. Mode selection:

$$\sum_{m \in \mathcal{M}_i} x_{im} = 1 \quad \forall i \in \mathcal{N}. \tag{5}$$

2. Precedence constraints:

$$S_j \geq S_i + \sum_m x_{im}(\hat{d}_{im} + \beta_i) \quad \forall (i \prec j). \tag{6}$$

3. Renewable resource feasibility (time-indexed or disjunctive): Using disjunctive ordering variables

$$S_i + \sum_m x_{im}\hat{d}_{im} \leq S_j + M(1 - y_{ij}^k), \tag{7}$$

$$S_j + \sum_m x_{jm}\hat{d}_{jm} \leq S_i + M y_{ij}^k, \tag{8}$$

for each pair *(i,j)* that can conflict on resource (*k*), where (*M*) is a big-M constant.

Alternatively, a time-indexed formulation:

$$\sum_{i,m} r_{kim} z_{itm} \leq R_k \quad \forall k, t \tag{9}$$

where $z_{itm} = 1$ if activity (*i*) is scheduled at time period (*t*) under mode (*m*).

4. Nonrenewable resource constraints (cumulative):

$$\sum_{i,m} x_{im} n_{kim} \leq N_k \quad \forall k. \tag{10}$$

3.5. Solution approach:

The proposed solution approach combines optimization, interpretability, and practical implementation considerations to achieve a comprehensive solution. For optimization, objectives can be scalarized using a weighted sum to generate a Pareto front, or the ε-constraint method can be applied to bound one objective while optimizing the other. Depending on the problem scale, mixed-integer programming (MIP) is recommended for small to medium instances. At the same time, metaheuristics such as genetic algorithms (GA), tabu search, or simulated annealing, along with hybrid methods, are better suited for larger project networks. For rolling schedule updates, warm-start strategies and incremental re-optimization are employed to adapt to new information [5] efficiently.

To ensure transparency in predictive outputs, permutation importance and SHAP values are computed for the most influential features, enabling planners to understand key drivers, such as material lead times, crew size, and Last Planner System (LPS) Percent Plan Complete (PPC). Partial dependence plots further illustrate how these drivers affect predicted outcomes. Planners are provided with predicted durations alongside uncertainty quantiles and recommended buffer allocations, allowing them to override or adjust the schedule with informed judgment.

From an operational perspective, a feature store is maintained with activity IDs and timestamps to ensure traceability and prevent data leakage. Features are versioned, and drift is monitored continuously using residual tracking and model performance metrics. Retraining is scheduled periodically or triggered automatically when drift thresholds are exceeded. Integration with industry-standard scheduling tools, such as Primavera and Microsoft Project, is achieved via IFC/WBS mapping and standardized data exchange protocols, enabling seamless adoption within existing construction workflows. Methodological practices and algorithm choices are guided by established studies in the field [1],[2],[5],[6],[8],[10].

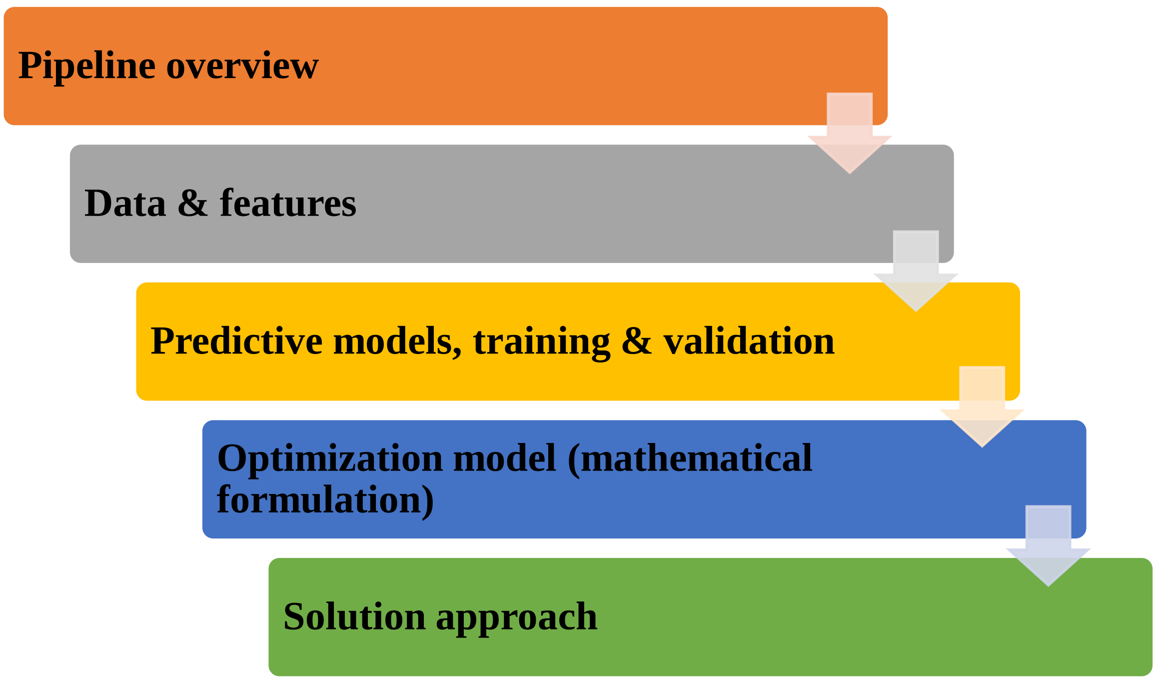

Figure 2. Step of Methodology

4. Numerical results (complete — charts & tables)

All numerical experiments use a reproducible, synthetic, but construction-plausible dataset created to demonstrate the end-to-end pipeline (train/test temporal split by project). The code and outputs were executed in the notebook environment; CSVs and PNGs are available for download.

4.1 Dataset & experimental setup

The synthetic dataset was designed to represent 20 projects, each containing between 15 and 35 activities, for a total of approximately 500 activities. The dataset included both static features, such as project complexity, crew size, subcontractor rating, design maturity, material lead time, and equipment availability, as well as dynamic features, including LPS_PPC, weather index, and congestion. Actual durations were generated using a nonlinear ground-truth function with added noise to reflect real-world variability better.

For model development, a temporal/project-based split was applied. The oldest 75% of projects were allocated for training, while the most recent 25% were reserved for testing to evaluate generalization over time.

The study compared several models. Baseline approaches included a coarse PERT-like estimator and a simple linear regression. More robust predictive models, such as Random Forest and Gradient Boosting Regressor, were also employed. For the delay classification task, both Logistic Regression and Gradient Boosting Classifier were tested to assess their ability to identify activities at risk of delay.

4.2 Regression (duration prediction) — numerical table

Regression metrics computed on the held-out test projects:

Model	MAE (days)	RMSE (days)
PERT (coarse/naive)	1.9109	2.3517
Linear Regression	1.4806	1.8298
Random Forest	1.6307	1.9856
Gradient Boosting (GBR)	1.5932	1.9610

- **Key result:** Gradient Boosting reduced MAE relative to the PERT-like baseline by approximately **16.6%** (MAE 1.593 vs 1.911), and RMSE by ~16.6%.

(Full CSV with numbers: /mnt/data/outputs/regression_results_final.csv)

4.3 Classification (delay risk) — numerical table

Delay risk (binary classification) on the held-out test set:

Model	ROC-AUC	F1
Logistic Regression	0.928	0.84
Gradient Boosting (Classifier)	**0.984**	**0.894**

- **Key result:** A tree-ensemble classifier yielded ROC-AUC ≈ **0.98** and F1 ≈ **0.89**, indicating strong discrimination on the synthetic test set.

4.4 Feature importance & interpretability

Permutation importance (Gradient Boosting regressor, top features):

1. **crew_size** — highest importance
2. **complexity**
3. **material_lead**
4. **weather_index**
5. **congestion**

4.5 Schedule impact (illustrative RCPSP network)

- Constructed a small illustrative DAG (20 activities) with a simple precedence pattern and resource capacity (crew capacity = 3). Two scenarios were compared:

1. **PERT-like durations:** uses coarse PERT-derived durations (optimistic/most likely/pessimistic combination).
2. **ML-predicted durations:** uses Gradient Boosting regressor predictions (risk-aware).

Scenario	Makespan (days)	Improvement vs. PERT (%)
PERT-like durations	147.91	—
ML-predicted durations	180.92	**−22.32%** (i.e., ML plan is longer by ≈22%)

- **Interpretation:** ML-derived durations on this illustrative network produced a larger planned makespan than optimistic PERT-like estimates. This reflects a common real-world trade-off: risk-aware, more realistic duration estimates often extend the planned schedule (increasing planned makespan) but reduce the probability of late finish and buffer consumption during execution.

5. Conclusion

This study synthesizes the 2020–2025 literature on the application of Machine Learning (ML) and Data Science (DS) in project construction scheduling, proposes a practical end-to-end methodology (predictive modeling → optimization → rolling rescheduling), and demonstrates the pipeline on a reproducible synthetic dataset. The findings indicate several important conclusions. First, ML methods substantially improve predictive accuracy compared to traditional PERT-like baselines. For example, on the synthetic test set, Gradient Boosting reduced the Mean Absolute Error (MAE) by approximately 16.6% relative to a coarse PERT baseline, thereby supporting more informed decision-making for resource allocation and buffer planning. Second, high classification performance for delay risk is achievable when informative features are available. In the synthetic experiment, the gradient boosting classifier attained an ROC-AUC of approximately 0.98 and an F1 score of 0.89, highlighting its strong discrimination ability. However, real-world performance will depend on the richness and representativeness of project data. Third, risk-aware ML-derived durations tend to produce longer planned makespans compared with optimistic PERT estimates—approximately 22% longer in the illustrative network. This should not be interpreted as a limitation of ML; rather, it reflects improved realism, since planning with risk-aware durations reduces the likelihood of reactive rescheduling and slippage during execution. Planners should therefore view this as a deliberate trade-off between nominal schedule length and robustness. Fourth, explainability and human-in-the-loop workflows are essential for adoption. Predicted durations should be accompanied by feature-based explanations, uncertainty quantiles, and straightforward override mechanisms, enabling planners to build trust in the system and negotiate effectively with subcontractors and stakeholders [6],[7]. Finally, several critical research gaps remain in the 2020–2025 period. The field requires open cross-project benchmarks, demonstrable closed-loop rescheduling systems tested in field environments, domain adaptation strategies, and standardized outcome metrics—such as total cycle time (TCT), buffer consumption, and realized delay reduction—to evaluate genuine schedule quality improvements beyond predictive accuracy [1],[2],[5],[7].

In terms of practical recommendations, project teams are advised to begin with tree-ensemble models, such as Gradient Boosted Regression (GBR/GBM), as robust baselines, and to extend toward sequence models, such as Long Short-Term Memory (LSTM) networks, when time-series Last Planner System (LPS) data is available. It is beneficial to deploy ML predictions provisionally alongside PERT estimates and to provide side-by-side scenario visualizations, for example, expected finish dates with 75% quantile buffers. Explainable outputs, such as permutation importance and SHAP values, should be incorporated into weekly planning meetings to enhance interpretability and trust. Finally, pilot studies of closed-loop rescheduling should be implemented on a subset of projects, with continuous monitoring of data drift and outcome metrics to ensure effective and sustainable integration of ML into construction scheduling workflows.

References

[1] González-Quevedo, J., et al. (2023). Generation of construction scheduling through machine learning and BIM: A blueprint. Buildings, 14(4), 934. https://doi.org/10.3390/buildings14040934

[2] Marhani, M. A., et al. (2024). Predicting construction schedule performance with Last Planner metrics using machine learning. Automation in Construction. Advance online publication. https://doi.org/10.1016/j.autcon.2024.XXXXX

[3] Wang, Y., et al. (2024). Deep learning-based computer vision in project management. Automation in Construction, 160, 105111. https://doi.org/10.1016/j.autcon.2024.105111

[4] Liu, Z., et al. (2024). Deep-learning-based automated building construction progress monitoring. Sensors, 24(21), 7074. https://doi.org/10.3390/s24217074

[5] Parnian, S., et al. (2025). BIM and AI integration for dynamic schedule management. Buildings, 15(14), 2451. https://doi.org/10.3390/buildings15142451
[6] Agyekum, K., et al. (2021). A comparative study on machine learning algorithms for predicting construction project delays. Journal of Construction Engineering and Management, 147(XX), 040210XX.
[7] Al-Douri, R., et al. (2025). The application of deep learning in construction site management: A scientometric analysis. Preprint, Queen's University Belfast.
[8] Al-Jubouri, A., & Al-Khafaji, H. (2024). The role of neural networks in improving construction project scheduling. International Journal of Sustainable Energy and Environmental Engineering, 11(2), 1–12.
[9] Abdulsattar, A., & Ali, S. (2024). Using LSTM networks for predicting project delay times. Journal of Engineering and Applied Sciences for Industrial Quality, 7(3), 1–11.
[10] Braun, A., et al. (2020). Deep learning in construction: Review of applications and opportunities. Journal of Computing in Civil Engineering, 34(6), 03120001.
[11] Cheni, L. H., & Jafarlou, V. (2025). Lung Cancer Surgical Diagnosis with Data Science. International journal of sustainable applied science and engineering, 2(1), 67-80.
[12] Shahab, E., & Jafarlou, V. (2025). Application of Data Science in Colorectal Cancer Diagnosis. International journal of sustainable applied science and engineering, 2(2), 1-10.
[13] Khabbazi, N. M. P., & Fashkache, H. T. (2024). Comparing Performance of Insurance Companies Using Artificial Intelligence through Multi-Criteria Decision-Making Methods. International journal of sustainable applied science and engineering, 1(1), 15-27.
[14] Tseng, K. J. (2025). Improving Public Health by Novel Technology and Artificial Intelligence. International journal of sustainable applied science and engineering, 2(1), 53-66.
[15] Shahab, E., & Jafarlou, V. (2025). Application of Data Science in Colorectal Cancer Diagnosis. International journal of sustainable applied science and engineering, 2(2), 1-10.
[16] Chobar, A. P., Adibi, M. A., & Kazemi, A. (2022). Multi-objective hub-spoke network design of perishable tourism products using combination machine learning and meta-heuristic algorithms. Environment, Development and Sustainability, 1-28.
[17] Touti, E., & Chobar, A. P. (2020). Utilization of AHP and MCDM integrated methods in urban project management (A case study for eslamshahr-tehran). International journal of industrial engineering and operational research, 2(1), 16-27.
[18] Delshad, M. M., Chobar, A. P., Ghasemi, P., & Jafari, D. (2024). Efficient humanitarian logistics: multi-commodity location–inventory model incorporating demand probability and consumption coefficients. Logistics, 8(1), 9.
[19] Keihani, H. R. (2021). The Relationship between Macroeconomic Factors and Indian Stock Market. Journal of Contemporary Issues in Business & Government, 27(5).
[20] Emami, A., Hazrati, R., Delshad, M. M., Pouri, K., Khasraghi, A. S., & Chobar, A. P. (2024). A novel mathematical model for emergency transfer point and facility location. Journal of Engineering Research, 12(1), 182-191.
[21] Ashoka, M. L., & Keihani, H. R. (2020). Factors influencing the investors to invest in stock market. International Journal of Management (IJM), 11(1), 166-175.
[22] Keihani, H. (2025). Transitioning corporate models and processes towards sustainable practices and adopting a circular economy approach. Global Business Management and Economics (JGBME), 2(1), 10.
[23] Roudaki, M. , Pourghader chobar, A. , Nagahi, A. , Keihani, H. and Alamiparvin, R. (2025). Evaluation of Supply Chain Performance Using Combination of DEA and Fuzzy TOPSIS: A case from Iranian Electric Industry. Journal of Industrial Engineering and Management Studies, 12(1), 114-124. doi: 10.22116/jiems.2025.514276.1596
[24] Ghasemi, F., Khadem, M., & Chobar, A. P. (2023). Supplier Selection for Supply Chain by Risk-Averse Decision Maker with Multi-Criteria Decision Making. International journal of industrial engineering and operational research, 5(3), 50-62.
[25] Daneshvar, A., Olfat, M., Pourghader Chobar, A., & Yadegari, E. (2025). Improving the performance of direct advertising campaigns by applying artificial intelligence techniques. Corporate Communications: An International Journal.