Delay Prediction for Subcontracted Production Orders

Department of Industrial Engineering
Hacettepe University

***Abstract*—Subcontracted manufacturing is essential in defence supply chains, yet late deliveries from external workshops frequently disrupt downstream assembly and integration plans. In practice, delays are rarely explained by workshop reliability alone; difficult-to-produce materials, prototype builds, and testing capability constraints can dominate lead-time variability even when a supplier is historically dependable. This study develops a regression-based delay prediction framework under confidentiality constraints by constructing a synthetic but operationally realistic dataset that captures workshop capability, product complexity, material criticality, and planning pressure indicators such as workload, downtime risk, urgency, and lead-time tightness. The target variable is a non-negative continuous delay duration defined as late days beyond the promised lead time. Four models Ridge Regression, Decision Tree Regression, Support Vector Regression (RBF), and Gradient Boosting Regression are trained using an end-to-end preprocessing pipeline and evaluated with nested cross-validation. Two outer validation schemes are reported: standard KFold and GroupKFold by workshop, to quantify performance both within a known workshop population and under cross-workshop generalisation. The data exhibit a pronounced lateness imbalance, with most orders delayed (approximately 80%) and a smaller on-time subset (approximately 20%), which motivates reporting multiple error metrics beyond R^2. Results indicate that Gradient Boosting is strongest under random splits, while Ridge is more stable when workshops are held out. Permutation importance and clustering-based risk profiling further highlight material criticality, difficulty, and workshop capability as key drivers of delay risk. Overall, the work demonstrates a transparent and reproducible pipeline for delay-risk aware planning when real operational data cannot be released.**

***Index Terms*—production delay, subcontracting, defence electronics, regression, nested cross-validation, group k-fold, permutation importance, risk profiling**

I. Introduction

Late deliveries from subcontracted workshops remain a persistent source of disruption in high-complexity manufacturing environments such as defence electronics. External suppliers often provide critical items including electronic boards, wiring harnesses, control modules, and specialised RF assemblies that directly feed final assembly and integration stages. When these items arrive after their promised dates, downstream schedules must be reworked, buffer inventories are consumed, expediting and coordination costs increase, and programme level delivery commitments can be jeopardised.

The operational reasons for subcontracting delays are multi-causal and rarely separable. While supplier capability and historical on-time delivery (OTD) performance matter, delays frequently arise even when a subcontractor is generally reliable. In many cases, the dominant sources of lateness are product and material driven: prototype builds and late engineering changes increase rework likelihood, difficult-to-procure or hard-to-manufacture materials introduce supply uncertainty, and testing requirements can create bottlenecks when specialised equipment or qualified personnel are limited. These effects are amplified by short-term planning pressure, such as high workload, downtime risk, urgent orders, and tight promised lead times relative to process effort. As a result, simple safety margins or rule-based planning approaches often fail to capture the combined and nonlinear nature of these drivers.

Machine learning (ML) offers a practical way to learn order level delay patterns from historical records and to provide planners with forward looking risk estimates. However, real defence electronics data are typically subject to strict information security policies and cannot be shared in academic settings. To address this constraint, the present study constructs a synthetic dataset designed to be operationally realistic rather than merely random. The dataset is then used to build and evaluate a regression-based delay prediction pipeline with careful validation choices that explicitly test whether performance depends on workshop reuse between training and test samples.

II. Description of the Problem and Problem Domain

A. Delay Definition and Prediction Objective

A subcontracted order is considered late when its realised lead time exceeds the promised lead time agreed between the prime manufacturer and the workshop. For order i,

$$L_i = C_i - P_i, \tag{1}$$

where C_i denotes the actual lead time (in calendar days) and P_i denotes the promised lead time. Since the focus of this study is on the *magnitude* of lateness rather than a late/on-time label, the learning task is formulated as a regression problem with a non-negative target:

$$D_i = \max(L_i, 0). \tag{2}$$

Here, D_i represents the number of late days beyond the promised lead time, and it is truncated at zero to exclude early deliveries from the penalty signal. The predictive objective is therefore to estimate D_i for a new order from its observable characteristics (product complexity, material criticality, workshop capability and short-term planning pressure). These continuous delay estimates are intended to support prioritisation and proactive intervention, for example by highlighting orders expected to experience severe delay and enabling earlier expediting, capacity rebalancing, or escalation with the subcontractor.

B. Operational Drivers and Class Imbalance Context

Several classes of factors jointly influence subcontracted lead times. Product complexity variables capture revision level, prototype status, testing requirements, and expected processing steps, each of which increases process uncertainty and rework probability. Material-related indicators represent supply fragility and manufacturability difficulty, which can dominate lateness even when the workshop performs well historically. Workshop capability and reliability features encode skill level, OTD rate, average delay, and delay variability, reflecting persistent performance differences across subcontractors. Finally, planning pressure variables such as workload level, downtime risk, urgency, and schedule tightness represent short-term conditions that can turn an otherwise feasible plan into a delayed outcome.

In this domain, delay events are not balanced. The dataset used in this study exhibits a strong lateness skew, with most orders delayed (approximately 80%) and a smaller subset on-time (approximately 20%). Although the prediction task is formulated as regression, this imbalance is operationally meaningful because it implies that "near-zero delay" cases are relatively rare, and evaluation should therefore consider robust error metrics (e.g., MAE, RMSLE, median absolute error) in addition to R^2.

III. Background and Related Work

A substantial body of literature has investigated the use of machine learning methods for predicting lead times, delivery delays, and supplier performance in manufacturing and supply chain environments. Early studies demonstrated that data-driven regression models can outperform traditional rule-based or statistical approaches when operational data exhibit nonlinear relationships and heterogeneous drivers. In particular, Oliveira et al. [1] applied multiple machine learning techniques to forecast purchasing lead times in an industrial supply chain and showed that support vector machines and ensemble-based learners achieve robust accuracy on tabular production data with mixed feature types.

More recent contributions have focused explicitly on late delivery prediction at the supplier or subcontractor level. Steinberg et al. [2] developed regression models to estimate delivery delays for low-volume, high-variety manufacturing systems, comparing tree-based ensembles and linear baselines on real procurement data. Their findings highlight that flexible models can capture complex delay mechanisms, but may also become sensitive to supplier-specific patterns, raising concerns about generalisation when new suppliers are introduced.

Beyond narrow forecasting tasks, a parallel research stream has examined machine learning as a tool for broader supply chain performance evaluation and risk analysis. Lima-Junior and Carpinetti [3] integrated neural networks with SCOR-based performance metrics, demonstrating that machine learning models can support managerial decision making by translating operational indicators into predictive performance signals. Review articles in the supply chain analytics literature further emphasise the growing role of artificial intelligence in managing uncertainty, supplier risk, and congestion effects, while also stressing the importance of interpretability and validation rigor when such models are deployed in practice [4].

From a methodological perspective, several studies caution against overly optimistic evaluation practices based on random data splits. When observations are grouped by entities such as suppliers, workshops, or production lines, random cross-validation can leak group-specific information into both training and test sets. As a result, grouped cross-validation schemes, in which entire suppliers are held out during testing, are increasingly recommended to assess true transferability across organisational units. This distinction between within-group prediction and cross-group generalisation is particularly relevant for subcontracted manufacturing, where decision makers may need to forecast delivery risk for both familiar and newly onboarded suppliers.

Finally, clustering-based segmentation has been proposed as a complementary decision-support layer in supply chain analytics. Rather than replacing regression-based prediction, clustering techniques are often used to translate continuous risk estimates into interpretable profiles that support prioritisation and planning. Such hybrid approaches align with the broader trend in operations analytics toward combining predictive accuracy with managerial interpretability, especially in environments characterised by complex interactions and gradual risk gradients rather than sharp class boundaries.

IV. Design/Description of the Solution

A. Synthetic Data Generation

Since real subcontractor data from defence electronics programmes cannot be disclosed due to information security constraints, a fully synthetic yet operationally realistic dataset was constructed. The objective was not to replicate any specific historical record, but to emulate the main behavioural patterns that govern delivery performance in subcontracted production, including complexity, congestion, material risk and workshop capability.

At the workshop level, each subcontractor was assigned a fixed performance profile characterised by technical skill, historical on–time delivery (OTD) rate, average delay and delay variability. These attributes were sampled from narrow random ranges to reflect relatively stable long–term differences between workshops. At the order level, product and planning characteristics were generated through weighted random sampling. Product category, engineering revision index, prototype status, testing requirement and material criticality were used as proxies for technical complexity, while workload level, downtime risk, urgency and order quantity captured short–term operational pressure and planning tightness.

Delivery delay was generated through a latent continuous score that combines three structural elements. First, linear terms represent the direct effects of individual drivers, such as higher revision levels, prototype builds, extensive testing, critical materials and low workshop skill. Second, several interaction terms were introduced to model conditional effects that arise only when factors coincide. For example, prototype orders processed under very high workload, or test–intensive items with highly critical materials, were assigned additional delay contributions. Similarly, urgent orders handled by low–skill workshops were penalised more strongly than either factor alone. These interactions ensure that the simulated delay process reflects the multiplicative risk amplification commonly observed in real subcontracting environments.

Third, nonlinear congestion effects were embedded through a convex workload component that increases rapidly once utilisation exceeds a high threshold. This captures the empirical observation that heavily loaded workshops experience disproportionately large delays due to queueing, rework and coordination losses. Finally, stochastic noise drawn from a normal distribution was added to the latent delay score to represent unobserved influences and natural variability, ensuring that outcomes are not deterministically fixed by the explanatory variables.

The resulting continuous delay was truncated to a plausible range and converted into a non–negative lateness measure D_i. The realised lead time was computed as the sum of the promised lead time and this positive delay, so that lateness emerges endogenously from a combination of nonlinear structure, interaction effects and random variation, rather than being imposed by an ad–hoc rule.

B. Column Description

A brief explanation of each column included in the synthetic subcontracting dataset is provided below.

- **Order_ID**: Unique identifier assigned to each production order.
- **Workshop_ID**: Identifier of the subcontractor workshop responsible for executing the order.
- **Product_Category**: Type of product manufactured (for example electronic board, control module or RF unit).
- **Revision_Index**: Engineering revision level that reflects design maturity.
- **Prototype_Flag**: Indicator specifying whether the order corresponds to a prototype item.
- **Test_Required**: Boolean variable stating whether functional or environmental testing is required.
- **Expected_Steps**: Estimated number of processing steps needed to complete the order.
- **Material_Criticality** and **Material_Criticality_Num**: Qualitative and numerical representations of supply chain criticality.
- **Workshop_Skill**: Technical capability rating of the subcontractor.
- **Workshop_OTD_Rate**: Historical on–time delivery performance of the workshop.
- **Workshop_Avg_Delay**: Average number of days by which past orders were delayed.
- **Workshop_Delay_Var**: Variability of historical delay behaviour.
- **Workload_Level**: Current utilisation level of the subcontractor's resources.
- **Downtime_Risk**: Estimated probability of downtime occurring during production.
- **Quantity**: Total quantity requested for the order.
- **Urgency_Level**: Priority classification of the order (normal, urgent or critical).
- **Promised_LT**: Lead time committed by the subcontractor.
- **Actual_LT**: Realised production duration recorded after completion.
- **LateFlag**: Binary indicator showing whether the order experienced a delay.
- **Delay_Days_Positive**: Continuous non–negative delay duration used as the regression target.

C. Feature Engineering

Several composite features were constructed to encode schedule pressure, technical complexity, workshop reliability, congestion and material supply risk in a compact and operationally meaningful form.

Schedule aggressiveness is captured through lead–time tightness, defined as Promised_LT divided by Expected_Steps, together with lead–time slack computed as Promised_LT minus Expected_Steps. A binary tight–schedule flag indicates structurally infeasible commitments when the slack is negative. These variables quantify how restrictive the promised delivery date is relative to the required processing effort.

Product and process complexity are summarised by a weighted index that combines engineering revision level, prototype status, testing requirement and expected process steps. Higher values represent orders that are more exposed to rework, engineering changes and verification delays.

Workshop capability is represented using two complementary indicators. The first aggregates technical skill, historical on–time delivery rate and average delay into a single reliability score. The second normalises the on–time delivery rate by historical delay magnitude, penalising workshops that appear punctual only because they tolerate long delays.

Capacity pressure and instability are captured by a congestion index that combines workload level with the workshop's historical delay and delay variability. This reflects the nonlinear increase in lateness risk when highly utilised workshops are also slow and volatile.

Material supply risk is modelled through the numeric material criticality score together with a discretised criticality group (low, medium, high), allowing the models to exploit both continuous and threshold effects of procurement difficulty.

Together, these engineered features translate heterogeneous operational inputs into structured risk channels that improve both predictive accuracy and interpretability of the delay models.

D. Model Portfolio and Rationale

Four regression models were selected to provide complementary inductive biases. Ridge Regression provides a stable linear baseline under multicollinearity and is expected to generalise well when additive effects dominate. A Decision Tree Regressor provides an interpretable nonlinear baseline that captures thresholds and interactions but can overfit without ensembling. SVR with an RBF kernel captures smooth nonlinear relationships in scaled feature space and serves as a strong classical nonlinear benchmark. Gradient Boosting Regression acts as a high-performing ensemble of shallow trees capable of representing complex nonlinear patterns and interactions in tabular industrial data. Reporting multiple models reduces the risk of over-claiming based on a single algorithm and clarifies which conclusions are robust across learners.

E. Preprocessing Pipeline and Training Procedure

All models were trained within a single end-to-end Pipeline to avoid leakage and to ensure identical preprocessing across validation folds. Numeric variables were median-imputed and standardised, categorical variables were imputed using the most frequent category and one-hot encoded with unseen-category safety. Model hyperparameters were tuned with RandomizedSearchCV using inner cross-validation, optimizing negative RMSE. The search spaces were chosen to control model complexity: Ridge tunes α, the Decision Tree tunes depth and minimum split/leaf sizes, SVR tunes (C, γ, ϵ), and Gradient Boosting tunes the number of estimators, learning rate, depth, and subsampling. Performance was evaluated using a multi-metric suite including RMSE, MAE, R^2, adjusted R^2, RMSLE, MAPE-adjusted, SMAPE, and median absolute error to reflect both average behaviour and robustness under skew.

F. Validation Design: Random Splits vs. Workshop Holdout

Two outer validation schemes were intentionally used. Outer KFold with shuffling approximates the scenario in which future orders follow a similar workshop mixture as the historical set, allowing the same workshop to appear across train and test folds. Outer GroupKFold by Workshop_ID enforces strict workshop separation, acting as a stress test for cross-workshop generalisation and directly addressing memorisation concerns. Both schemes used three outer folds, while two inner folds were used inside the hyperparameter search. This nested design ensures that hyperparameter choices are not influenced by the outer test fold and that reported performance reflects held-out behaviour.

V. Results and Discussion

After constructing the engineered feature set and defining the nested cross–validation framework, all four regression models (Ridge, Decision Tree, Support Vector Regression with RBF kernel, and Gradient Boosting) were trained and evaluated under two complementary outer validation schemes. For each outer fold, hyperparameters were selected by inner cross validation using randomised search, and performance was then measured on held–out data that were never seen during tuning. This design ensures that the reported metrics reflect genuine out of sample generalisation rather than optimistic in–sample fit.

TABLE I: Outer KFold performance (mean ± std across 3 outer folds).

Model	RMSE	MAE	R^2	Adj. R^2	RMSLE
Gradient Boosting	0.892 ± 0.031	0.682 ± 0.017	0.887 ± 0.015	0.873 ± 0.017	0.248 ± 0.006
SVR (RBF)	0.939 ± 0.015	0.707 ± 0.007	0.875 ± 0.007	0.860 ± 0.008	0.264 ± 0.010
Ridge Regression	1.000 ± 0.029	0.786 ± 0.014	0.858 ± 0.006	0.841 ± 0.007	0.267 ± 0.001
Decision Tree	1.506 ± 0.246	1.157 ± 0.222	0.676 ± 0.083	0.638 ± 0.093	0.397 ± 0.064

TABLE II: Outer GroupKFold performance with workshop separation (mean ± std across 3 outer folds).

Model	RMSE	MAE	R^2	Adj. R^2	RMSLE
Ridge Regression	1.264 ± 0.130	1.009 ± 0.120	0.764 ± 0.052	0.737 ± 0.055	0.372 ± 0.064
Gradient Boosting	1.504 ± 0.852	1.207 ± 0.748	0.647 ± 0.321	0.610 ± 0.348	0.404 ± 0.238
SVR (RBF)	1.523 ± 0.315	1.244 ± 0.294	0.663 ± 0.084	0.626 ± 0.083	0.453 ± 0.151
Decision Tree	2.177 ± 0.527	1.690 ± 0.433	0.310 ± 0.208	0.231 ± 0.230	0.519 ± 0.134

Two outer splitting strategies were deliberately applied. Standard KFold represents the case in which future orders are drawn from the same pool of subcontractors observed during training, while GroupKFold enforces strict separation by workshop and therefore simulates prediction for new or previously unseen subcontractors. Comparing these two regimes allows us to distinguish between general patterns of production risk and workshop–specific effects that do not necessarily transfer across suppliers.

A. Cross–Validated Performance Summary

Tables I and II report the mean and standard deviation of the main error metrics across the three outer folds for both validation schemes. Root mean squared error (RMSE) is used as the primary ranking criterion, since it penalises large under or over estimations of delay more strongly than absolute metrics and is well aligned with operational risk.

Under KFold, Gradient Boosting achieves the best mean RMSE and the highest R^2, indicating that the nonlinear interactions embedded in the synthetic delay generation mechanism are captured effectively when training and test sets share workshop identities. Support Vector Regression follows closely, while Ridge performs slightly worse but remains stable. The single decision tree exhibits substantially higher error and variance, reflecting its limited ability to generalise without ensembling.

Under GroupKFold, the ranking changes in a meaningful way. Ridge regression becomes the best performing and most stable model, while Gradient Boosting and SVR suffer both higher error and much larger variability across folds. This contrast is operationally important: it shows that a significant part of the predictive signal exploited by the nonlinear models is correlated with workshop identity or workshop specific behaviour. When a workshop is entirely held out, those patterns are no longer available, and simpler linear structure generalises more reliably.

Taken together, these results do not indicate naive memorisation of individual orders, but rather a realistic form of group dependence. In practical terms, the KFold results

TABLE III: Representative hyperparameters selected by RandomisedSearchCV under each outer validation scheme.

Model	Outer KFold	Outer GroupKFold
Ridge Regression	$\alpha \approx 31–75$	$\alpha \approx 120–194$
Decision Tree	depth $\approx$ 3–None, leaf $\approx$ 1–5	depth $\approx$ 5–10, leaf $\approx$ 3–7
SVR (RBF)	$C \approx 5–25$, $\epsilon \approx 0.1–0.4$	$C \approx 5–30$, $\epsilon \approx 0.2–0.5$
Gradient Boosting	200–700 trees, lr $\approx$ 0.03–0.10	300–700 trees, lr $\approx$ 0.02–0.12

describe expected performance when predicting for known subcontractors, whereas the GroupKFold results provide a conservative estimate of performance when new or weakly observed workshops enter the system. This distinction is essential for interpreting model quality in a subcontracting environment.

B. Hyperparameter optimization via Randomised Search

All regression models were tuned using RandomisedSearchCV within a nested cross-validation framework. For each outer training fold, hyperparameters were optimised exclusively on the corresponding inner folds, and the selected configuration was then evaluated on the held-out outer fold. This design prevents information leakage and ensures that reported performance reflects genuine generalisation rather than optimization bias.

Randomised search was preferred over exhaustive grid search for two reasons. First, several hyperparameters lie on continuous domains, where random sampling is known to be more efficient than grid-based exploration. Second, the computational budget was intentionally limited to reflect realistic industrial constraints, making randomised optimization a practical and reproducible choice. Across all models, the optimization objective was minimisation of root mean squared error, evaluated via inner-fold cross-validation.

Table III reports representative optimal hyperparameter values selected during the outer cross-validation process. For each model and validation scheme, the table shows typical values observed across folds, directly extracted from the optimization logs. While the exact configuration varies slightly from fold to fold, the selected parameters are consistent in magnitude and reflect stable optimization behaviour rather than random fluctuations.

The optimization results provide additional insight into model behaviour. Under GroupKFold, Ridge consistently favours stronger regularisation, indicating that simpler linear structures generalise more reliably when entire workshops are held out. In contrast, Gradient Boosting selects deeper ensembles and lower learning rates, reflecting its attempt to capture nonlinear interactions when sufficient within-workshop information is available. Overall, the randomised optimization procedure complements the cross-validation results by confirming that performance differences across models are structural rather than artefacts of poor tuning.

C. Diagnostic Interpretation for Gradient Boosting

To complement the summary metrics, Figures 1–4 visualise out-of-fold predictions for Gradient Boosting under both outer schemes. These plots are helpful because high R^2 alone can hide systematic errors, especially when the target distribution is skewed and contains many repeated integer-like values.

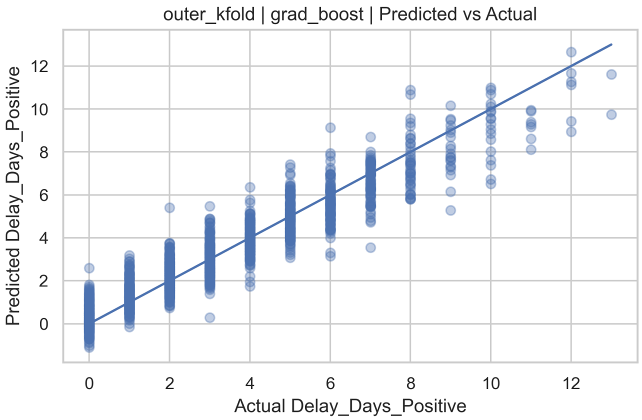

Fig. 1: Out-of-fold predicted versus actual delays for Gradient Boosting under outer KFold.

In the predicted-versus-actual plot under outer KFold (Fig. 1), the point cloud follows the diagonal trend closely for small and medium delays, which is consistent with the high R^2 reported in Table I. The densest region occurs at low delay values, indicating that most orders cluster around short delays, while fewer points appear at the high-delay tail. Two behaviours are visible in that tail. First, dispersion increases as actual delay grows, meaning that long delays are harder to predict precisely than short delays. Second, there are indications of regression-to-the-mean: some large delays are underpredicted and some moderate delays are overpredicted. This is typical in operational delay forecasting because extreme outcomes are often driven by factors that are either unobserved or only weakly represented by the available features.

The corresponding residuals-versus-predicted plot under outer KFold (Fig. 2) provides a clearer view of error structure. The residual band widens as predicted delay increases, which indicates heteroskedasticity: the variance of the error grows with the expected delay level. In applied terms, the model is more reliable for routine, low-risk orders and less reliable for high-risk cases, which is operationally plausible. The visible diagonal stripe pattern is also expected here because the target is effectively discretised in days; when many observations share the same integer-valued actual delay, residuals form parallel lines in a residual-vs-predicted coordinate system. This should not be interpreted as an algorithmic failure; it is mainly a geometric artefact of plotting residuals when the outcome has repeated levels.

Under outer GroupKFold, the predicted-versus-actual plot (Fig. 3) becomes visibly more dispersed. The diagonal trend is still present, which shows that the model continues to learn meaningful global relationships, but the scatter around the diagonal is larger, indicating reduced calibration when the

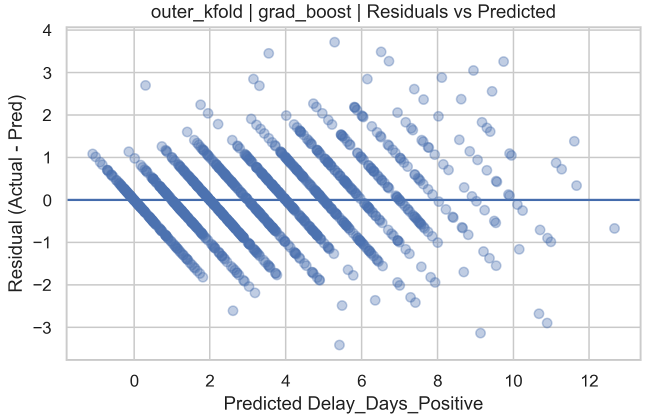

Fig. 2: Residuals versus predicted delay for Gradient Boosting under outer KFold.

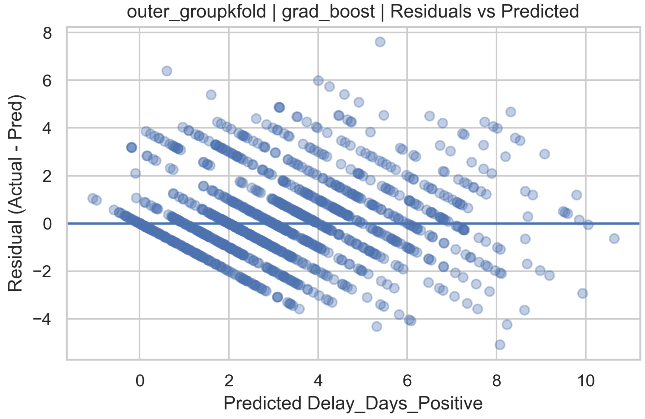

Fig. 4: Residuals versus predicted delay for Gradient Boosting under outer GroupKFold.

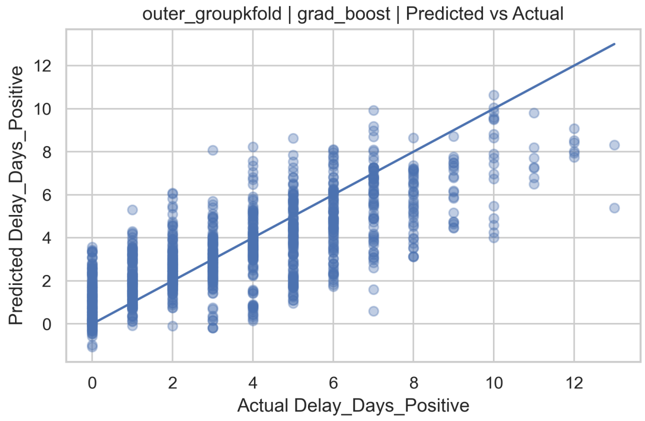

Fig. 3: Out-of-fold predicted versus actual delays for Gradient Boosting under outer GroupKFold.

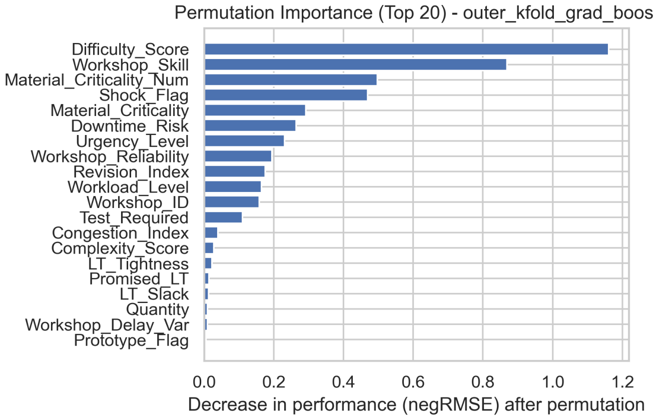

Fig. 5: Permutation importance (top 20) for Gradient Boosting under outer KFold.

workshop has not been observed in training. The corresponding residual plot (Fig. 4) shows stronger spread and more extreme outliers, confirming that Gradient Boosting is less stable when generalising across workshops.

Overall, the combined evidence from Tables I–II and Figures 1–4 supports two conclusions: Gradient Boosting is the most accurate model when predicting delays for familiar workshops, but its performance degrades under strict workshop separation, in which case simpler linear models such as Ridge offer more robust generalisation.

D. Permutation Importance and Managerial Interpretation

Permutation importance was computed by permuting raw input variables prior to one-hot encoding and measuring the resulting deterioration in predictive accuracy, expressed as the decrease in negative RMSE. This procedure preserves a one-to-one mapping between importance scores and operationally meaningful features, avoiding the interpretability problems caused by dummy-variable expansion.

Under outer KFold with Gradient Boosting (Fig. 5), the most influential drivers are dominated by technical difficulty and workshop capability. Difficulty Score, Workshop Skill and Material Criticality Num appear at the top of the ranking, followed by shock and downtime indicators. This pattern is consistent with the synthetic delay mechanism, which combines nonlinear effects of complexity, capability and disruption. Because the same workshops appear in both training and testing, the model can also exploit stable workshop-specific baselines, which explains the non-negligible contribution of Workshop ID and related reliability variables.

Under outer GroupKFold with Ridge (Fig. 6), the ranking shifts toward transferable supply-side and capability signals. Material Criticality Num becomes the dominant feature, followed by Workshop Skill and Workshop Reliability. This is a realistic outcome: material risk reflects upstream supply constraints that affect any workshop in a similar way, whereas workshop identifiers and highly correlated workshop-specific patterns cannot be exploited when an entire workshop is held out. The reduced importance of difficulty-related aggregates

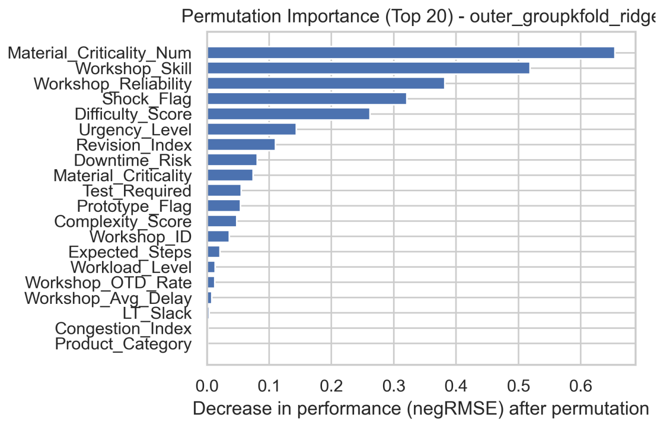

Fig. 6: Permutation importance (top 20) for Ridge under outer GroupKFold.

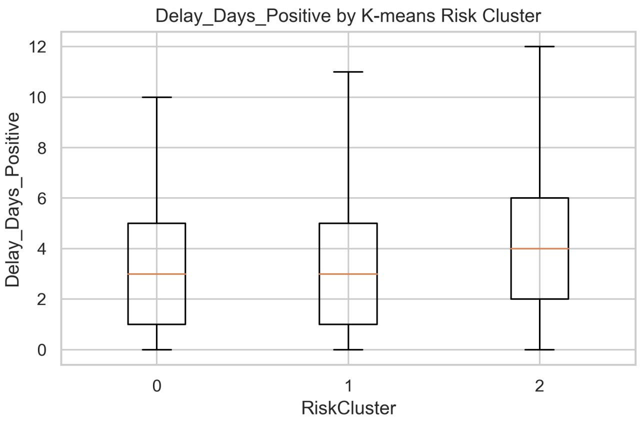

Fig. 7: Distribution of positive delay days across K-means risk clusters.

in this setting indicates that part of their explanatory power is tied to workshop allocation rather than purely to product characteristics.

From a managerial perspective, these two views are complementary rather than contradictory. When forecasting delays for known subcontractors, technical difficulty and local workshop capability are the primary risk drivers. When predicting for new or infrequently observed workshops, material criticality and general capability indicators become more reliable warning signals. The consistency of this shift across validation schemes suggests that the models capture structurally meaningful delay mechanisms rather than relying on spurious memorisation.

E. Risk Profiling via K-means Clustering

In addition to supervised delay prediction, an unsupervised K-means model was used to group orders into a small number of operational risk profiles based on complexity, congestion, schedule tightness, urgency, and material-related features. The purpose of this step is not to compete with regression accuracy, but to provide a transparent, model-independent view of structural delay risk that can support prioritisation and managerial screening.

Silhouette analysis indicates that three clusters provide the most meaningful partition of the risk space, although the moderate silhouette values suggest that delay risk is continuous rather than forming sharply separated classes. This is consistent with real subcontracting environments, where multiple drivers such as materials, testing, workload and workshop capability interact gradually rather than through hard thresholds.

Figure 7 shows the distribution of realised delay days across the three risk clusters. A clear ordering is visible. Cluster 2 corresponds to the high-risk regime: it has the highest median delay and the widest upper tail, indicating both systematically larger delays and a higher probability of extreme outcomes. Clusters 0 and 1 represent lower-risk regimes with similar medians, but Cluster 0 exhibits slightly heavier tails, meaning that it occasionally escalates into severe delay despite having a moderate typical level.

From an operational perspective, this segmentation complements the regression results in an important way. While the regression models estimate the expected delay magnitude for each order, clustering identifies persistent structural risk regimes. Orders falling into the high-risk cluster can be flagged for early intervention such as material verification, expediting or tighter follow-up, even when point predictions remain uncertain.

The consistency between clustering and the residual diagnostics of the regression models further supports this interpretation. Both analyses show that high-delay cases are more volatile and harder to predict accurately, which justifies treating them as a separate risk class rather than relying solely on point forecasts. Together, supervised prediction and unsupervised risk profiling provide a coherent and practically useful view of subcontracted delay risk.

VI. Conclusion

This paper proposed a transparent and operationally grounded framework for predicting delivery delays in subcontracted defence electronics production. Because real industrial data cannot be disclosed for security and confidentiality reasons, a fully synthetic dataset was constructed to reproduce the main structural drivers of lateness observed in practice, including material criticality, engineering complexity, testing and prototype requirements, workload congestion, and heterogeneous workshop capability. Unlike many purely illustrative datasets, the generation process embedded nonlinear congestion effects and interaction terms, ensuring that delays emerge endogenously from realistic operational mechanisms rather than from ad–hoc rules.

Four regression models were trained within a unified preprocessing and feature engineering pipeline and evaluated using a nested cross validation design. Within this structure, model hyperparameters were optimised using RandomisedSearchCV inside each outer training fold. This design choice ensures that hyperparameter selection is fully separated from outer test data and prevents optimistic bias. Moreover, the resulting parameter configurations provide additional insight into model behaviour, revealing how different algorithms adjust their complexity when faced with varying generalisation requirements across validation schemes.

Two complementary outer validation schemes were deliberately adopted. Random KFold reflects the common industrial scenario in which future orders originate from workshops that are already present in historical data, whereas GroupKFold enforces a more demanding setting in which entire workshops are held out, mimicking the challenge of forecasting performance for new or drifting suppliers. This dual evaluation is essential because it distinguishes apparent accuracy from genuine transferability.

The results show that Gradient Boosting achieves the strongest predictive performance when training and test data share workshop identities, confirming that the nonlinear interactions embedded in the synthetic delay mechanism are effectively captured by flexible learners. Hyperparameter optimization under this scheme tends to favour richer ensemble configurations with moderate learning rates, allowing the model to exploit workshop-specific patterns. However, when evaluated under strict workshop separation, Ridge regression emerges as the most stable model, exhibiting lower error and substantially less fold to fold variability than more complex alternatives. In this setting, the optimised Ridge models consistently select stronger regularisation, reflecting the need to suppress workshop-dependent effects and to rely on more transferable global relationships.

Diagnostic plots and permutation importance analyses further support this interpretation. Residual patterns demonstrate that prediction uncertainty increases with delay magnitude, which is operationally realistic for high risk orders, and that Gradient Boosting loses calibration when faced with unseen workshops. At the same time, feature importance rankings shift in a coherent way across validation schemes: under random splits, difficulty and workshop capability dominate, whereas under workshop holdout, transferable drivers such as material criticality and general reliability become more prominent. This consistency between error structure, hyperparameter behaviour, and variable importance indicates that the models learn meaningful operational relationships rather than relying on spurious memorisation.

From a managerial perspective, the framework therefore supports two complementary use cases. When the objective is to prioritise and schedule orders within a stable network of known subcontractors, Gradient Boosting offers superior accuracy and risk discrimination. When robustness across suppliers is required, for example in supplier onboarding or portfolio rebalancing, simpler linear models with stronger regularisation provide safer and more reliable forecasts. By explicitly combining conservative validation with data-driven hyperparameter optimization, the study avoids overclaiming a single universal performance figure and instead aligns model choice with deployment context.

Overall, the combination of realistic synthetic data, nested hyperparameter optimization, conservative validation against group leakage, multi metric evaluation for skewed delay distributions, and interpretable risk decomposition yields a coherent and extensible methodology. Once secure access to real subcontractor data becomes possible, the same framework can be applied directly, allowing defence electronics planners to move from reactive delay handling toward proactive, data–driven risk management. [1]

REFERENCES

[1] M. B. de Oliveira, P. K. S. Agrawal, and A. K. P. de Carvalho, "Lead time forecasting with machine learning techniques," in *Proc. Int. Conf. on Applied Industrial Artificial Intelligence*, 2021, pp. 1–12.

[2] F. Steinberg, P. Burggräf, J. Wagner, B. Heinbach, and A. Brintrup, "A novel machine learning model for predicting late supplier deliveries of low-volume high-variety products with application in a German machinery industry," *Supply Chain Analytics*, vol. 1, 2023, Art. no. 100003.

[3] F. R. Lima-Junior and L. C. Carpinetti, "Predicting supply chain performance based on SCOR metrics and neural networks," *International Journal of Production Economics*, vol. 212, pp. 19–38, 2019.

[4] G. Baryannis, S. Validi, S. Dani, and G. Antoniou, "Supply chain risk management and artificial intelligence: state of the art and future research directions," *International Journal of Production Research*, vol. 57, no. 7, pp. 2179–2202, 2019, doi:10.1080/00207543.2018.1530476.

[1] The synthetic dataset generation script and the full modelling pipeline are available at: https://github.com/zekayapayken/delay-prediction-subcontracting