Analytical Approaches for Predicting Variance in Construction Productivity using Regression Methods

by

Md Monjurul Hasan

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Construction Engineering and Management

Department of Civil and Environmental Engineering

University of Alberta

Abstract

Modeling productivity entails establishing the relationship between various factors that impact construction productivity, connecting input factors with output productivity. While prefabrication facilities may limit the influence of external project factors on productivity, complexity, and variability in product design, productivity-influencing factors from the internal project environment regulate the production flow and still causes productivity to vary broadly. Such factors are numerous, and it is impossible to account for every relevant detail in the model. In addition, the developed model needs to be logically driven, transparent, easy to use, and practical to secure the trust of the practitioners. Data collection efforts for maintaining the data-driven model must be practically optimized to reduce the overall overhead cost. Therefore, a systematic, transparent, logic-driven, and quantitative approach for modeling productivity would sufficiently transform a particular set of significant input parameters into the output of corresponding productivity. Furthermore, considering those input parameters that describe the fabrication of certain products, productivity variation in connection with each input parameter of the model is attributed to customization of the fabricated product of the same type, which needs to be accounted for to estimate the likely range of the predicted productivity. In other words, besides the point value prediction of productivity, the model should also provide the variance estimate of the prediction to gauge the associated variations.

Regression methods like multiple linear regression (MLR) and the model tree (MT) have been mainstream methods for quantitatively modeling labor productivity. Both of these techniques are instrumental in generating productivity prediction models that are transparent and

understandable; hence, model predictions can be trusted by decision-makers. It is important to note that even though model tree algorithms are considered as the nonlinear regression model, which combines decision trees and MLR analysis to establish complex-nonlinear relationships between variables. Such regression models are generally established based on analyzing the error terms between the predicted output and the target output without addressing the variance of the predicted output and the impact of individual input parameters on the variance. A model with high accuracy (the mean of the prediction close to the target value) but low precision (too high the variance of the prediction) would be deemed inadequate in the context of cost-estimating applications. An analytical method to account for the impact of the variability associated with each input parameter on the variation of the final output has yet to be formalized.

This research critically reviews established methods for variance analysis on commonly applied regression methods, namely MLR and model tree in cost estimating and productivity prediction for fabricated construction, as well as the impact of productivity variance on project cost budgeting. This research first proposes a novel method that integrates the error propagation theory with MLR modeling in an attempt to quantify the variance of the MLR predicted output. A metric based on the resulting variance analysis (i.e., the ratio of standard deviation over mean) for gauging the precision of the MLR model has also been proposed. Next, the variance analysis technique is used to enhance the model tree algorithm to extend its capacity to make predictions along with estimating the variance of the predicted output. A productivity modeling framework, therefore, has been formalized using the enhanced model tree algorithm to connect the unique design features of fabricated products (e.g., structural steel, prestressed concrete elements) as input with productivity as output. The productivity model's performance has been cross-checked

against the models prepared using MLR and artificial neural network (ANN) models. The enhanced model tree outperforms MLR in prediction accuracy and is preferred to ANN because (1) the variance is analytically predicted alongside the point-value output, and (2) the productivity model is explainable in terms of the reasoning logic for productivity prediction. In addition, two additional questions in connection with the variance in productivity prediction are addressed in this research, namely: (1) how the variability encoded in the fabrication productivity at work packages propagates from the work package level to the entire project level through the project network schedule and (2) how variations in project-level labor-hour are accumulated over the course of the project duration. The proposed methodology has been verified using Monte Carlo simulation and validated by conducting case studies based on fabrication projects in the real world. By bridging the gap in knowledge on variance analysis of nonlinear machine learning algorithms, this research advances computing in explainable artificial intelligence (XAI) with broad application potential to tackle a wide range of civil engineering problems beyond productivity modeling and analysis. A specific application has also been shown to generate a concrete strength prediction model as a generic Civil Engineering application case.

.

Preface

This thesis is the original work of Md Monjurul Hasan. It is structured in a paper-based format, consisting of a total of six chapters. The thesis includes an Introduction chapter at the beginning and a Conclusion chapter at the end. Currently, one chapter of the thesis is under revision, while two chapters have been successfully published in peer-reviewed journals. Additionally, one chapter has been published in conference proceedings.

Chapter 2 of this thesis has been published as

> *Monjurul Hasan and Ming Lu (2022) "Variance Analysis on Regression Models for Estimating Labor Costs of Prefabricated Components." Journal of Computing in Civil Engineering, Vol. 36, No. 5, 04022019 (1 -13), doi.org/10.1061/(ASCE)CP.1943-5487.0001037*

This chapter has been reprinted with the permission of the American Society of Civil Engineers (ASCE) Journal of Computing in Civil Engineering. Monjurul Hasan was responsible for conceptualization, methodology, data curation, formal analysis, validation, writing the original draft, and editing. Dr. Ming Lu was the supervisory authority and was involved with the conceptualization, methodology, and manuscript review and editing.

Chapter 3 of this thesis has been submitted for publication as

> *Monjurul Hasan and Ming Lu (2023) "Enhanced Model Tree Modeling Technique for Quantifying Output Variances Due to Random Data Sampling: Productivity Prediction Applications." Automation in construction, Elsevier (under review).*

Monjurul Hasan was responsible for conceptualization, methodology, data curation, formal analysis, validation, writing original the draft, and editing. Dr. Ming Lu was the supervisory authority and was involved with the conceptualization, methodology, and manuscript review and editing.

Chapter 4 of this thesis has been published as

> *Monjurul Hasan and Ming Lu (2023) "Estimating Output Variance of a Regressing Tree Model: Case Study of Concrete Strength Prediction." In Proceedings of ASCE Computing in Civil Engineering Conference, June 25-27, Cornville, Oregon.*

This chapter has been reprinted with the permission of the ASCE. Monjurul Hasan was responsible for conceptualization, methodology, data curation, formal analysis, validation, writing the original draft, and editing. Dr. Ming Lu was the supervisory authority and was involved with the conceptualization, methodology, and manuscript review and editing.

Chapter 5 of this thesis has been published as

> *Monjurul Hasan and Ming Lu (2021) "Error Propagation Model for Analyzing Project Labor Cost Budget Risks Due to Variations Inherent in Labor Productivity in Industrial Construction." Journal of Construction Engineering and Management, Vol. 147, No. 4, 04021007 (1 -11), doi.org/10.1061/(ASCE)CO.1943-7862.00020.*

This chapter has been reprinted with the permission of the ASCE Journal of Construction Engineering and Management. Monjurul Hasan was responsible for conceptualization, methodology, formal analysis, validation, writing the original draft, and editing. Dr. Ming Lu was

the supervisory authority and was involved with the conceptualization, methodology, and manuscript review and editing.

Since this thesis follows a paper-based format, the literature review was detached from each chapter instead to provide a comprehensive chapter for the literature review. Additionally, there may be some repetition of concepts and sentences in between the chapters, as each chapter is written as a standalone paper with its own research problem statement, research methods, problem validation, and case studies.

Acknowledgement

All praise and gratitude belong to ALLAH, the Most Beneficent, the Most Merciful, for granting me the strength and perseverance to successfully complete this thesis.

I am deeply thankful to Dr. Ming Lu, a Professor in the Department of Civil and Environmental Engineering at the University of Alberta, Canada, for his invaluable guidance, suggestions, and encouragement throughout my research. This work would not have been possible without his wise advice and unwavering support. I owe him an immense debt of gratitude. Furthermore, I would like to extend my heartfelt thanks to my doctoral committee members, Dr. Farook Hamzeh and Dr. Ahmed Hammad, for their valuable support with critical feedback during my research. I would also like to express my sincere appreciation to my examination committee members, Dr. Jeff Boisvert and Dr. Hubo Cai, for their generosity in sharing valuable insights and providing constructive suggestions, which have been instrumental in refining this thesis.

I am grateful to the Canadian Precast Concrete Institute (CPCI) and MITACS Canada for their financial support during the initial two years of my study. I want to acknowledge the Natural Science and Engineering Research Council of Canada (NSERC) for offering me the doctoral student scholarship to support this study. I want to express my appreciation to Robert Burak and Dr. Val Sylaj from CPCI, Kevin Kooiker and Jeff Church from Eagle Builders, and Jason Rabasse from Lafarge Precast, Edmonton, for sharing their invaluable industry insights.

Most importantly, I would like to thank my parents for their unconditional love, constant encouragement, and unwavering support throughout all stages of my life. Without their blessings and guidance, achieving this goal would have been impossible.

Finally, I would like to dedicate my thesis to my grandparents who always dreamed that I would become a doctorate.

Table of Contents

List of Tables

List of Figures

Acronyms

AI ..... Artificial Intelligence

ANN ..... Artificial Neural Network

ANP ..... Analytic Network Process

AON ..... Activity on Node

BIM ..... Building Information Model

CPM ..... Critical Path Method

DEMATEL ..... Decision-Making Trial and Evaluation Laboratory

EMT ..... Enhanced Model Tree

HFS ..... Hybrid Feature Selection

HPC ..... High Performance Concrete

KPI ..... Key Performance Indicator

LH ..... Labor Hours

MAE ..... Mean Absolute Error

MAPE ..... Mean Absolute Percentage Error

MLR ..... Multiple Linear Regression

MSE ..... Mean Square Error

MT ..... Model Tree

PCA ..... Principal Component Analysis

PRESS ..... Predicted Error Sum of Square

RMSE ..... Root Mean Square Error

SDR ..... Standard Deviation Reduction

STD ..... Standard Deviation

TOPSIS ..... Technique for Order of Preference by Similarity to the Ideal Solution

XAI ..... Explainable Artificial Intelligence

Chapter 1

Introduction

1.1 General

Industrial construction employs various trades in large-scale prefabrication operations to produce modules and structural components at an offsite facility, which are shipped to the field for rapid installation. Like any other construction project, the success of offsite construction projects depends on adequate planning, including estimating, scheduling, and budgeting. Construction planning starts with estimating the work content for individual work packages, followed by the generation of a production schedule considering the resource and technological constraints of the fabrication shop. While preparing the production schedule, it is also vital for the shop manager to know the margin of contingency in the project cost budget at different time points of the project production schedule for the purpose of project control. The production process in the indoor construction system of fabrication remains labor intensive due to the "made-to-order" nature of fabrication engineering. This means given the fabricated products of the same type, and there are variations in design parameters to cater to different clients' requirements and various project needs. Therefore, these unique design features of the bespoke construction products demand adjustments to the workflows of predefined production processes form job to job from time to time. The variability inherent in the production time is still significant in the indoor construction environment of fabrication on individual activity levels and the entire project level. Labor

productivity is the widely accepted determinant of performance among practitioners, and it is used as a piece of information crucial to estimating, scheduling, and budgeting on any construction project. Developing an analytical methodology for characterizing the effect of variability in productivity upon labor cost budgeting and production schedule is warranted and critical to managing prefabrication projects in construction.

1.2 Construction Productivity

The fact that productivity in the construction industry has remained stagnant during the last few decades has garnered wide attention from society (Barbosa et al. 2017). Despite the advancement of technology and mechanization of construction methods, the construction industry remains labor intensive, and labor costs make up from a quarter to half of the total project cost (Liu et al. 2011). Hence, labor productivity has subsequently served as a key performance indicator (KPI) for the construction sector and has been assessed at several levels with varying degrees of precision (Song and AbouRizk 2008). Labor productivity is used to measure industry trends and establish benchmarks against other industries (Vereen et al. 2016). Company-level or project-level productivity measurement provides internal and external benchmarks for comparing company or project trends/performances (Ellis and Lee 2006). Activity-level labor productivity estimates have a significant impact on project budgeting, scheduling, and control (El-Gohary et al. 2017). Among different levels, measuring activity level productivity –also called the detailed productivity estimate– is the most critical one; beyond a performance indicator, activity level productivity essentially provides inputs to many construction management applications. Moreover, industry and company levels of labor productivity can be measured by averaging and aggregating the detailed level productivity estimates. Given the importance of forecasting activity level

productivity, this study focuses on measuring and predicting labor productivity for project planning consisting of detailed estimating, scheduling, and project control. The term productivity refers to labor hours per unit of work used interchangeably with construction productivity and labor productivity in this research.

Estimating determines the work scope and predicts the financial resources needed to deliver a project from initiation to completion. An accurate cost estimate and a realistic contingency estimate based on assessing the risky elements that could potentially increase project cost are vital to delivering a project within the committed budget (Creedy et al. 2010). In general, uncertainties inherent in internal and external project environments present distinctive challenges in setting the contingency margin of an estimate in competitive tendering (Enshassi et al. 2013). Labor productivity is the basic piece of information essential to the project cost estimate in labor hours (LH) and the generation of project schedules and cost budgets. As shown in Eq. 1.1, the labor hour estimate LH for a particular work package is derived from the productivity P and total work unit W (quantity takeoff by a certain unit of measure) that needs to be done to complete the work in a given work package. Note the labor hour LH and work unit W express a linear relationship defined by the productivity P as the slope of the equation. Any variation in P must translate to LH as the estimated uncertainty. As illustrated in Fig. 1.1, when the productivity of work unit W varies within the range $[P_{\mathrm{L}}, P_{\mathrm{U}}]$, the corresponding estimate for labor hours (LH) also varies within the bounds of LH_{U} and LH_{L}. Therefore, LH will not have just one point value; rather will have a mean and estimated variance of the mean in order to reflect the uncertainty associated with the prediction. In a practical setting, a project consists of many work packages, and each of them is complicated by the variations of the different product types. Analytical methods for explaining

how associated variances in productivity estimates are aggregated from work package levels and exert an impact on the risk in the overall labor hour budget are absent from the literature in the construction domain.

$$LH = P \times W \qquad (1.1)$$

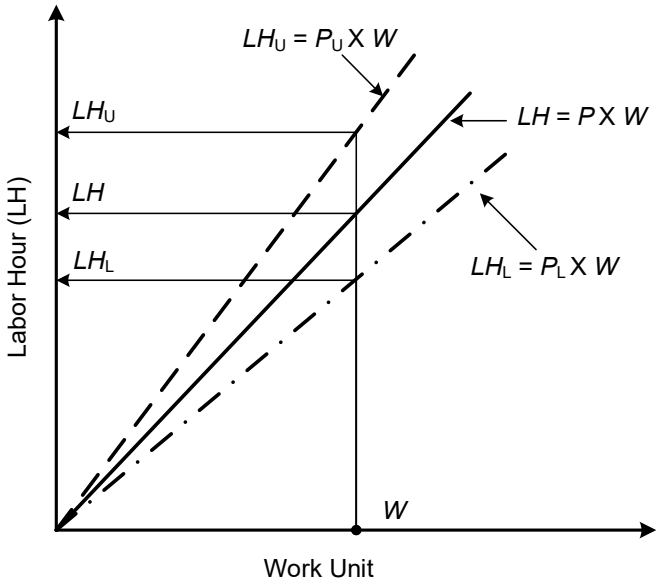


Figure 1.1: Labor hour (LH) estimate from work unit (W) and productivity (P) information.

Estimating productivity is considered a complex decision-making process. To obtain productivity data, the current practice mainly relies on sources such as the estimator's judgments, published data sources, and historical project data (Motwani et al. 1995; Song and AbouRizk 2008). The practice generally starts with the estimator deciding the activity level labor productivity (labor hour rate) for a given work package for a project based on a "base productivity." The "base productivity" is adjusted or modified to reflect the specific conditions expected to be encountered in the project. It is noteworthy that the base rate is often determined statistically from past historical data or industry benchmarks. Many companies and trade organizations publish

construction productivity data. For example, RSMeans Company (Gordian 2023) publishes quarterly construction cost and productivity data compiled from information sourced from consultants, contractors, and trade organizations. However, published productivity data only represent the industry average productivity rates without addressing the wide variation inherent in the performances of specific contractors on particular projects. Gauging the "level of difficulty" multiplier to adjust the base productivity rate up or down to reflect the complexity of work packages specific to a project demands the experience of estimators in a specific organizational and operational environment. Following such a procedure, a selection of relevant elements is identified that impacts labor productivity and taken into account (Lu et al. 2001) to prepare a point value estimate of productivity (Fig 1.2).

Figure 1.2 Productivity model input output relationship.

Thanks to a large quantity of influential factors in the project environment underlying the high variability of productivity, it becomes challenging to develop a mathematical model in an attempt to accommodate the impacts of various elements (known and unknown) by relying on a predetermined number of inputs correlated with target productivity output (Tsehayae and Fayek 2016). Also, using a limited selection of data sampled from a large productivity database at a

particular time to establish the productivity model potentially adds to the variance of the predicted productivity. For instance, given the identical input factors, at a different time, the dataset available for addressing the same productivity problem by MLR could change in size and content, resulting in differences in calibrated parameters and prediction results. As a result, the accuracy of the estimating model can be subject to question. Note that the precision of a productivity prediction model is defined with a certain confidence interval around the predicted point value. With the variance of the predicted productivity value known, a range of the required labor hours can be determined with definitive boundaries (Fig. 1.3). If the width of the range exceeds an acceptable threshold, i.e., the precision is too low; the model prediction should be rejected regardless of the high accuracy of the point value prediction.

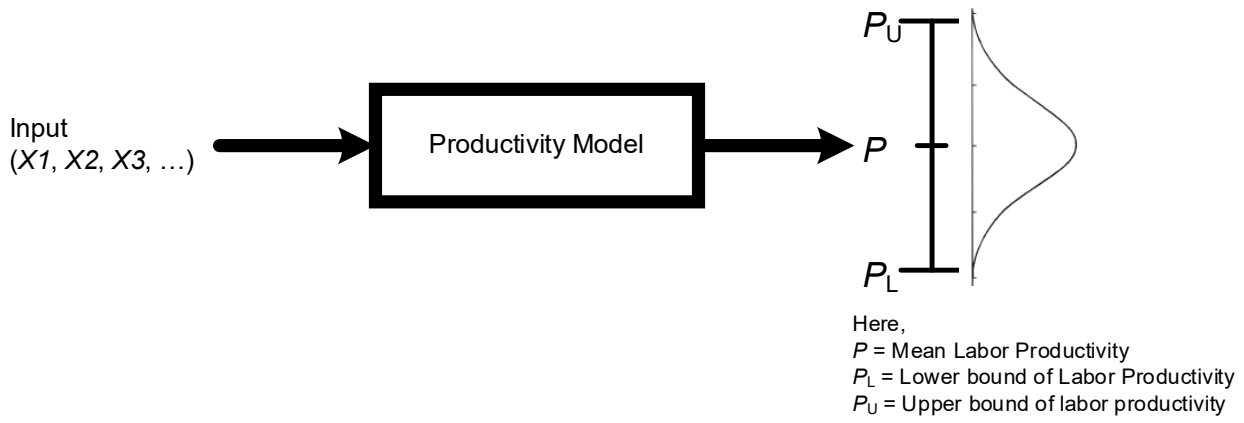


Figure 1.3 Productivity model input output relationship with estimated variance of the prediction.

When a productivity prediction model does not provide variance in connection with the predicted value, the decision on whether the precision is sufficient is left to be made by humans, and the

outcomes can be inconsistent and unreliable, reflecting the estimator's experience and temperament. As such, the insufficiency in the existing productivity prediction model could lead to confusion and mistrust, which would have a negative effect on the ensuing critical functions of project management such as estimating the labor cost to complete a scope of work for an activity or budgeting total cost for the whole project. It is apparent that appropriate analytical methods for modeling how input variables impact the productivity estimate with respect to variance analysis are necessary but absent in both knowledge and practice. In order to investigate the cause and impact of the input variables on productivity estimate, such variance analysis needs to establish the linkages between all the system model variables (both input and output) while sufficiently accounting for the specific application context as defined by the available data. In addition, existing models for estimating labor productivity place emphasis on the accuracy of point value prediction without accounting for the confidence of the predicted productivity value or implying the risk associated with the prediction. Therefore, this could have caused the lack of an estimator's confidence in predictive analytics, preventing the estimator from taking advantage of the predicted productivity value in preparing project estimates and cost budgets. This research contributes to the existing body of knowledge by generalizing a complete framework to account for the inevitable variations in connection with construction productivity and generate sufficient productivity prediction models based on regression method. The productivity in prefabricated structural components production provides the application context.

Prefabrication has been chosen since it offers a controlled factory-like environment compared with traditional construction methods. The indoor setting of prefabrication construction (structural steel, precast concrete) makes it less susceptible to external factors such as weather events,

allowing for better quality control, safety, and resource flow (Li et al. 2022; Zhou and Ren 2020). Despite the benefits afforded by the prefabrication and off-site construction facility, the production process in the indoor construction system of prefabrication remains labor intensive due to the "made-to-order" nature of prefabrication engineering. As a result, variance in productivity at individual work items (work packages) is still significant due largely to one-of-a-kind design features of the engineered products (Thomas et al. 2002; Hamzeh et al. 2019), thus making the problem of prefabrication in construction a class of its own (Assad et al. 2022). In preparing the future bid and making construction plans, there is a need to develop a methodology to check the sufficiency of the productivity prediction model by ensuring that the productivity model accounts for variabilities inherent in productivity due mainly to different design features and the internal environment (e.g., the multi-project concurrent processing). A systematic methodology to identify the most critical elements influencing the variation of productivity of a construction work package considering the variabilities in product design and other internal factors is also vital. Researchers have yet to pay much attention to this issue.

In summary, this research aims to improve the variance estimation technique of regression-based prediction models namely multiple linear regression (MLR) and model tree in construction productivity modeling. Unlike artificial neural network (ANN), MLR and model tree are logic driven machine learning model which ensures transparency in model development in explaining linear and nonlinear input-output relationships respectively. Such transparency in model structure is essential for gaining trust from practitioners. It is noteworthy that the model tree combines a decision tree with regression analysis by breaking down the big data set into small subsets, so that the nonlinear input-output relationship can be represented into a series of linear multi-variate

data models (Quinlan 1992). In the nutshell, Model Tree represents a tree structure made of a series of MLR models; each MLR represents a sub-model calibrated with a subset of the training data (Wang and Witten 1997; Solomatine and Xue 2004).

To materialize this vision, the research develops an analytical framework and uses it to create a labor productivity prediction model that relates product features as inputs to the target productivity as output, along with an estimate of the variance of the output in a quantitative fashion. The variance of the productivity estimate resulting from the system environment of a current application problem will be characterized in terms of a variance contribution ratio of each input factor; in the case of MLR or model tree, the variance contribution ratio is determined based on the standard deviation of the input slope parameter (namely, the productivity ratio on detailed activity). The research will also provide a method for analyzing the variance in productivity on steps at the fabrication project level; the proposed model will fix the variance of the total labor cost budget at a given control time in the scheduled project duration. An S stripe can be created as a visual aid for project planning by plotting the lower and upper bounds of the interval for cumulative labor hours budgeted at different points in the project timeline. This visually represents the risk to the labor cost budget due to variability in labor productivity, similar to how the S-Curve plots progress over time, as shown in Fig. 1.3. In Fig. 1.4(a), a traditional S-curve gives the cumulative LH estimate at a given time in the project duration, which does not portray the associated risk of the point estimate. In contrast, the proposed S stripe gives the upper and lower bounds of the cumulative LH estimate at a given time as a derived confidence level, as shown in Fig. 1.4(b).

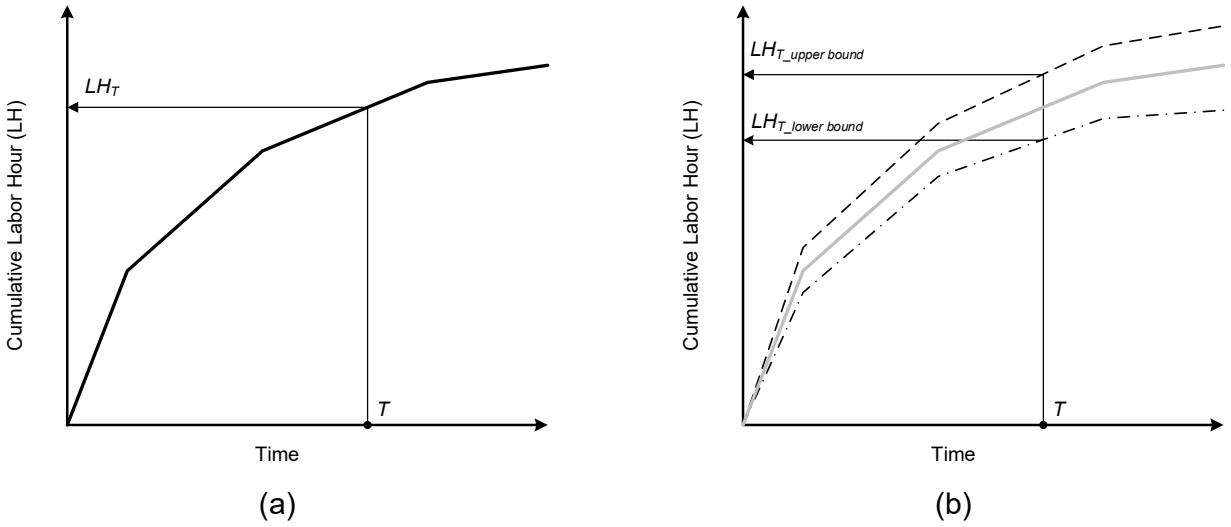


Figure 1.4 Cumulative LH required by a project at a given time: (a) S-curve; and (b) proposed S stripe.

In collaboration with industry partners in Alberta, the proposed research is illustrated and validated through an application case involving the estimation of labor costs for prefabrication construction projects. With technology, environment, labor, material, and management being held relatively constant in a controlled environment, prefabricated structural components and the industrialized construction setting of the construction process have given rise to a broad scope of data collection in a structured manner.

1.3 Background

This section presents recent advancements in construction productivity modeling, and regression applications in general in engineering, and application of error propagation theory in engineering problem solving and highlights the difference between my contribution from others.

1.3.1 Productivity Models

The literature has abundant evidence in terms of applying data analytics to improve productivity prediction (Minato and Ashley 1998). Various techniques have been utilized in productivity-related research to model the intricate empirical relationships between different factors (both internal and external to the project) and productivity rates (Hamza et al., 2019). Work-study techniques were adopted in several productivity models, in which only a few factors related to the work method were included (Thomas and Daily 1983). Such work-study models have limitations in relating the external project environment and management factors with the model output. The expectancy and action-response models are alternative techniques to explain construction productivity variations. The effort that an individual exerts on a job accounts for variations in job performance or productivity is captured in the expectancy model (Maloney and McFillen 1985). The action-response model visually displays the interaction of several factors that contribute to productivity loss (Halligan et al. 1994). Both models contributed to understanding productivity variances; nevertheless, they were limited in their ability to quantify the effects of many factors on construction productivity (Sommez and Rowings 1998). Statistical techniques in combination with linear regression models (Smith 1999; Mohsenijam and Lu 2019), and nonlinear regression models like artificial neural networks (ANNs) (Sonmez and Rowings 1998; Lu et al. 2001; El-Gohary et al. 2017), are two popular methods for productivity modeling. Linear regression and statistical analyses are generally limited by the number of influencing factors that can be included and their capability to measure the combined, nonlinear effect of the influencing factors (Yi and Chan 2014). On the other side, ANN models have shown good potential for quantitatively evaluating the effects of multiple factors on productivity, especially when a large quantity of

factors with complicated inter-factor nonlinear relationships are present (Tam et al. 2002; El-Gohary et al. 2017). ANN model acts like a black box without providing much explanation on the input-output relationship, which sometimes leads to mistrust among the practitioners and limits their application in the new project context. Another significant challenge for measuring and documenting the parameters influencing construction productivity arises from the subjective concepts involved in defining most parameters (Guo et al. 2017). An expert system is a technique used in estimating labor productivity for different construction activities for its adaptability in different project contexts described with limited data (Christian and Hachey 1995). Combining the fuzzy set theory with a rule-based system enhances the capability of an expert system by enabling the realistic constraints of subjective assessments of multiple contributing factors and has been used in productivity modeling for many construction projects (Fayek and Oduba 2005). However, expert systems generally have limited capabilities in identifying a mapping function to produce a generalized solution. Besides, rules obtained from domain experts to formulate the expert system can be affected by personal prejudices and attitudes because of the complex nature of productivity estimation (Yi and Chan 2014).

Besides the mainstream methodologies discussed, recent literature introduced the hybridization of data classification techniques combined with a regression model to improve the accuracy of labor productivity predictions (Elmousalami 2020; Mohsenijam et al. 2021). Neural-Network-Driven Fuzzy Reasoning (NNDFR), a hybrid intelligent structure, showed potential for modeling datasets with clear clusters (Mirahadi and Zayed 2016). Hybrid feature selection (HFS), which combines filter and wrapper methods with principal component analysis (PCA), has been used to identify relevant factors for developing labor productivity models (Ebrahimi et al. 2022). In addition,

combinations of methods such as the Decision-making trial and evaluation laboratory (DEMATEL) method, the Analytic Network Process (ANP) method, and the Technique for Order of Preference by Similarity to the Ideal Solution (TOPSIS) method was proposed for comparing the productivity of different construction methods in a single model (Shahpari et al. 2019).

Although there has been good progress in developing productivity estimation models, much of the focus has been on increasing prediction accuracy. While the importance of accuracy and precision is well-documented in the literature, there is less discussion of the importance of taking into account the risks associated with relying on a single prediction value. Portas and AbouRizk (1997) used an artificial neural network (ANN) model to predict productivity in different zones, providing a range estimate instead of a point estimate. Recognizing that estimators might not accept a single predicted value from a labor productivity model without information about the risks associated with its accuracy, Lu et al. (2000) presented the output of an ANN model as a distribution over a range of values. An expert system combined with fuzzy logic is being used to find productivity with a degree of association with linguistic output variables by Fayek and Oduba (2005). In the literature, ample range estimations are introduced to predict construction budget contingency where Monte Carlo simulation has been applied (Shaheen et al. 2007; Sonmez 2011). The model-predicted output in terms of the variance has yet to be investigated but is of vital importance and immediate relevance to estimating applications in practice. The precision of a model's prediction has been largely overlooked in validating productivity models in the literature. Specifically, model validation and input selection are largely based on evaluating error terms between the predicted output and the target output without addressing the variance of the predicted output and the impact on the variance due to individual input parameters. For any model, performance is

evaluated based on a fixed dataset. Whether this model is accepted or sufficient to serve the purpose has seldom been discussed in the literature. Formulating analytical solutions to tackle the identified fundamental limitation of labor productivity prediction models motivates me to conduct this research.

1.3.2 Regression Applications

MLR was used to predict the peak and average traffic volume (Lingras and Adamo 1996). Liu et al. (2010) applied MLR to determine the deterioration of the underground metallic pipe subject to different soil properties. Cao et al. (2018) tested the MLR model to predict the unit bid price and evaluated the prediction performance by mean absolute error (MAE), the mean absolute error percentage (MAPE), and mean square error (MSE). However, the precision of model prediction had been largely overlooked and model validation had not factored in the variance of the predicted output and its relationship with the input parameters. Choi et al. (2015) applied the data clustering technique combined with MLR model to determine the future maintenance cost of highways. To prove the robustness of the model, the predicted error sum of square (PRESS) statistics was used to determine the quality of prediction by comparing each observed response from a model calibrated with a new subset against that from the fitted model. Nonetheless, the variance in the model output resulting from using a different subset of the data in MLR was not investigated. In addition, MLR (linear regression) and artificial neural network (ANN) models were contrasted for risk assessment of the bridge maintenance performance projects (Elhag and Wang 2007). Both MLR and ANN were calibrated to model risk scores and risk categories. Pearson's correlation coefficient (r), and the root mean square error (RMSE), and MAPE were used to evaluate the accuracy performance of the proposed models. When the MLR model was

presented separately with the testing data set (a subset of the modeling data), significant changes in model accuracy (RSME, MAPE, and r values) were observed. Yet, the precision performance in terms of the model prediction variance was not assessed. Another relevant endeavor was to evaluate the performances of different data mining techniques in predicting the compressive strength of high-performance concrete by Chou et al. (2011). Nonlinear regressions (such as ANN and support vector machines) and MLR models were calibrated using manyfold cross-validation techniques, while model's prediction performance was evaluated based on r, RMSE, and MAPE (Chou et al., 2011).

In construction estimating applications, researchers used factor analysis techniques for MLR models to enhance the accuracy of the estimated project costs by selecting the more significant factors that contributed to the estimation (Trost and Oberlender 2003, Zayed and Halpin 2005; Lowe et al. 2006). Model accuracy was reported in most cases as the standard error or variance of the final output based on the estimated residuals. Portas and Abourizk (1997) maintained that a range of productivity estimates is always preferable to the estimator instead of having a point value estimate. Therefore, providing a confidence range around the predicted point value is highly desirable. Marchionni et al. (2016) applied one quantitative equation to estimate the interval of the predicted output of the MLR model developed to assess the water supply infrastructure cost. However, this interval prediction equation was constructed assuming that the residual mean is zero and has a constant variance. The parametric prediction intervals made under this assumption may not perform well when the dataset is relatively small and/or has outliers with rather large residuals (Olive 2007). Mohsenijam and Lu (2019) demonstrated a modified version of the above equation to quantify the prediction interval for an MLR model in modeling labor hours for

structural steel fabrication projects. It is noteworthy such established MLR variance estimation methods fall short in accounting for the impact of the uncertainty associated with each input parameter of the model on the uncertainty of the final output.

The model tree is generally categorized as a non-linear regression model (Chang and Kim 2011; Vanli et al. 2019) which applies binary rules to identify data classes suitable for creating MLR model for each class. The final model is expressed as a set of MLR models constrained by a set of rules delimiting each class. The Model Tree based approach was applied to various civil engineering problems such as predicting labor cost, workability of concrete (Mohsenijam et al. 2021), forecasting river flow (Taghi Sattari et al. 2013), sediment transport in pipes (Najafzadeh et al. 2017), predicting compressive strength of high-performance concrete mix (Deepa et al. 2010). In the construction application domain, decision tree was used in estimating productivity loss due to project change orders (Lee et al. 2004). Desai and Joshi (2010) applied decision trees with constant branch nodes to analyze and predict labor productivity. Despite its simplicity and explainability, model tree algorithm still lacks the variance estimate of the predicted output for a given set of inputs.

The literature review conducted for this research reveals a gap in the existing knowledge regarding measurements that quantify the contribution of each input variable of a regression-based model, such as MLR, to the overall uncertainty or variance of the predicted output. Currently, the only available approach to obtain such measurements is through experimental design and sensitivity analysis, which involves altering the input variables and observing the resulting changes in the output. This holds true not only for MLR models but also for model tree models, as the final expression of an MT model consists of a collection of MLR equations constrained by a set of rules.

Furthermore, it is common practice in civil construction estimation to develop the total estimate by aggregating the estimate for distinctly scoped work packages. Therefore, there is a practical need for calibrating a set of MLR equations –each denoting a model for one work package– in order to generate the final estimate. Variances in the MLR equations need to be aggregated in quantifying the uncertainty of the estimate, resulting in the variance of the final estimate. MLR based methods for the analysis of the variance in the final estimate by aggregating a collection of work package estimates are yet to be formalized.

1.3.3 Error Propagation Theory

The law of error propagation generalizes the relationship between random variable errors and the corresponding function (Koch 1999); and provides the fundamental formula for the evaluation of precision in adjustment theory (Amiri-Simkooei et al. 2016). Given a linear function, the variance or covariance of the error propagation model can be derived analytically. The error of any system output can be obtainable through functional substitution with truncated Taylor Series (considering up to first order derivative). A nonlinear function can be preprocessed with linearization by Taylor series expansion prior to applying error propagation, while it is theoretically possible to extend Taylor series expansion to any existing order in order to improve computing accuracy (Xue et al. 2015). Nonetheless, research and application of the error propagation theory in civil or construction engineering have remained scarce. Most of the related applications were found to fall in the field of automation control and sensor technology (Cho et al. 2004, Cantoni et al. 2007, Hasan and Lu 2017, Wai-Lok Lai et al. 2018). The error propagation theory has been a less applied alternative in construction research to substitute for "what-if"

scenario simulation experiments or Monte Carlo (MC) sampling techniques applied on a sufficient system model.

This research integrates the error propagation theory with MLR modeling to quantify the output variance in connection with labor cost estimating in prefabrication. Applying first-order derivatives to approximate the propagation of random errors in the MLR model makes the solution algorithm straightforward, practical, and sufficient and eliminates the need for additional professional computer software, sensitivity analysis, or MC simulation analysis. The MC simulation is used in the case study to cross-validate the results against the proposed model.

1.4 Problem Statement

A predominant approach in risk analysis pertaining to system output (such as labor productivity), and its susceptibility to fluctuations in input variables (factors that influence productivity), typically involves a two-step procedure. Firstly, Step 1 entails the establishment of a system model that delineates the interrelationship between input variables and the system's output. Subsequently, Step 2 involves sensitivity analysis, wherein an input variable is perturbed across its practical range of variation, giving rise to the observation of responses in the corresponding system output. In this way, the risks associated with the system output are inferred from the empirical data of observed system outputs, often through the execution of "what-if" scenario analyses or simulation analyses on the computer.

In Step 1, researchers often employ various modeling techniques such as multiple linear regression, fuzzy expert systems, or artificial neural networks (ANN) when the physical or logical processes of the system lack clarity, but historical data of the real-life system allow for direct input-to-

output mapping (Kisi et al., 2017). Alternatively, in cases where the underlying processes of the system can be distinctly and logically represented within a computer simulation platform, such as discrete event simulation or critical path network scheduling, operational simulation models can be established in Step 1 by mapping processes over time and space in the problem domain (Halpin and Riggs, 1992; Mulholland and Christian, 1999).

Analysis in Step 2 entails designing a significant number of plausible scenarios that represent variability and uncertainty in input variables, achieved through either design or random sampling. Subsequently, experiments are conducted by assessing each scenario using the established model on the computer. Examples of such analyses include what-if scenario analysis on an operational simulation model (Chan and Lu, 2008) or Monte Carlo simulation experiments wherein variability in input variables is described using statistical distributions (Lu et al., 2001). In general, successful implementation of this research approach necessitates expertise in modeling methods, computer tools, coding, and statistical analysis.

Regression serves as a valuable and practical analytical method for modeling complicated input-output relationships in real-world systems and facilitating the prediction of system behaviors through mathematical equations (Barrett and Gray, 1994). Various regression models have gained widespread acceptance as quantitative techniques applicable to a diverse range of problem domains (El-Abbasy et al., 2014). Notably, multiple linear regression (MLR) has emerged as the most commonly utilized method for estimation tasks, owing to its simplicity, flexibility, and ease of interpretation (Lowe et al., 2006). Besides, basic regression technique also serves as the foundation of the construction of the nonlinear regression models (e.g., Model Tree). The evaluation of an

MLR model's accuracy extensively relies on the analysis of relative and absolute errors and the statistical correlation between model outputs and target values (Olive, 2017).

The significance of each input factor is explicitly encoded in the first-order slope parameter, reflecting its directional and magnitudinal impact on the predicted output. Thus, input factors with slopes close to zero are regarded as less influential for the prediction and can be omitted without significant detriment to the overall model accuracy (Chan and Park, 2005; Mohsenijam et al., 2017). However, it is worth noting that an MLR model exhibiting high accuracy, i.e., the mean of the prediction closely aligns with the target value, but low precision, i.e., high variance of the prediction, may prove inadequate in the context of cost estimating applications. Despite its importance, the precision of the MLR predicted output, characterized by its variance, has been relatively underexplored in the literature and holds immediate relevance to practical estimating applications.

Conventional model validation techniques typically focus on evaluating error terms between the predicted output and the target output, overlooking the variance of the predicted output and its sensitivity to individual input parameters. Addressing this fundamental limitation of MLR models through the formulation of analytical solutions is the primary motivation driving this research endeavor.

In this research, productivity for fabricating a product has been defined by disaggregating the labor hours according to the major feature attributes of the product being produced (Eq. 1.2), as in Step 1. For example, the productivity of the formwork preparation for reinforced concrete columns can be derived from the three feature attributes: height of the formwork, area of the

plywood, and lumber weight (Peurifoy and Oberlender 2001). As shown in Eq. 1.2, P is established by combining respective productivity contributions based on three product feature attributes (P_1, P_2, and P_3). Productivity based on each feature attribute can be divided by the productivity of the standard product (or base product) of each feature attribute (Eq. 1.3). Say, for productivity in connection with feature 1 is P_a (given in Eq. 1.3), and dimension of the standard product is a_0; therefore, for a product dimension, productivity contribution would be $P_1 = \left(\frac{P_a}{a_0}\right) a$.

$$P = P_1 + P_2 + P_3 \qquad (1.2)$$

or,

$$P = \left(\frac{P_a}{a_0}\right) a + \left(\frac{P_b}{b_0}\right) b + \left(\frac{P_c}{c_0}\right) c \qquad (1.3)$$

Following the same formwork preparation example, a 3 m height ($a_0 = 3$) column is the standard height of a column and it contributes to 1.5 LH/m ($P_a = 1.5$) in the final activity level productivity, P due to the height as product attribute; so, for the 2.5 m (a) height of the column, the contribution (P_1) would be, $\frac{1.5}{3} \times 2.5 = 1.25$. Similarly, if 12 square meters (b) of plywood and 150 kg (c) lumber weight are two other attributes of the standard's column and their contributions to the activity level productivity are 2.5 LH/m ($P_b = 2.5$) and 1.8 LH/m ($P_c = 1.8$) respectively. So, for 2.5 m height column, with 10 sq. m. plywood and 110 kg lumber weight, the activity productivity would be as per Eq. 1.3, the final activity level productivity would be 4.4 LH/m ($P = 4.4$). According to the carpenter's handbook quoted in Peurifoy and Oberlender (2001), the final labor productivity for such formwork erection work package should vary from 3 to 5.0 LH/m.

$$P = \left(\frac{1.5}{3}\right) 2.5 + \left(\frac{2.5}{12}\right) 10 + \left(\frac{1.8}{150}\right) 110 = 4.4 \qquad (1.4)$$

As shown in Eq. 1.4, the productivity model equation resembles the basic form of MLR equation, where a, b, and c are the inputs and $\beta_a, \beta_b, and\ \beta_c$ are the slopes of the MLR equation. This productivity model definition is different from the existing models published in the literature.

$$P = \beta_a a + \beta_b b + \beta_a c = f(a, b, c) \qquad (1.5)$$

The rapid advancement of information technology has significantly alleviated the constraints on data availability for constructing quantitative decision support systems in construction estimating (Mohsenijam and Lu, 2019). Notably, the industry's adoption of building information modeling (BIM) systems has facilitated capturing historical data on design features for building components, including prefabricated products. Furthermore, job cost data, encompassing actual labor hours spent in fabrication, is consistently recorded in payroll and project cost control systems. As a result, substantial and high-quality datasets encompassing design features and labor cost information have become more accessible and cost-effective, with a continuous influx of data from recently completed projects (Lee et al., 2017). Leveraging such datasets, analytical methods based on MLR hold promise for analyzing model parameter variability in response to dynamic data changes. Furthermore, in the context of estimating applications, it is imperative to unveil the variability in predicted productivity estimates arising from the inherent variability of model parameters.

The primary intention of this research is to develop analytical methods based on the error propagation theory to address specific application requirements. Firstly, the focus is on establishing an MLR model with pertinent and independent input factors and verifying its point-value prediction accuracy. Subsequently, the research centers on addressing the variance of the

MLR's predicted output, which is synonymous with assessing the precision of the prediction. The quantification of the MLR output variance at specific input points within the problem domain's input space will be achieved through a derived analytical solution, which will be cross-checked using Monte Carlo (MC) simulation on a demonstration case. Furthermore, an MLR-based productivity model will be formulated to establish a variance estimate of the productivity prediction concerning the point value productivity. The research explores the feasibility of utilizing a metric based on the resulting variance, specifically the ratio of standard deviation to mean, as a means to assess the precision of the MLR model. A precision threshold, for instance, 10%, can be utilized to determine whether the MLR model meets the precision criteria. In cases where the precision falls below the threshold, the model will be deemed acceptable; otherwise, the model will be rejected, prompting a reexamination and refinement of input definitions and modeling data before considering model updates or alternative nonlinear regression models. Additionally, acknowledging the limitations of the MLR model in capturing non-linearity within productivity modeling for prefabricated products, this research introduces a nonlinear regression model named "model tree" and enhances its capabilities by integrating the variance analysis technique based on the variance analysis model formulated for MLR equations. Lastly, for project budget preparation, the research aims to develop a formalized methodology to address variance in the total labor hour budget by consolidating variance estimates of labor productivity at the work package level.

1.5 Research Questions

The research aims to answer the following research questions:

Research Question 1: Can the variance of an MLR mode's prediction be adequately explained using error propagation theory if the coefficients of the input parameters are described in statistical terms, such as mean and standard deviation (square root of the variance) instead of a constant?

Research Question 2: For activity level productivity estimates in construction projects, can the mean and variance of the productivity estimate be derived from the quantity takeoff of important feature attributes and their productivity contributions?

Research Question 3: In the context of labor hour estimation for construction projects, can the total variance of labor hour estimates at different control points throughout the project lifecycle be determined analytically, given the known variance of labor productivity estimates for different work packages, without resorting to a Monte Carlo simulation approach?

1.6 Objectives

The main objective of this research is to advance the knowledge and practices of labor budgeting for prefabrication projects in industrial construction, particularly addressing the inherent variations in labor productivity. The study aims to make significant academic contributions to the existing knowledge in the field by achieving the following objectives:

Objective 1: Develop a variance analysis technique for MLR-based prediction models by integrating error propagation theory by generalizing a methodology for:

- Quantifying input parameter variability through statistical measures like mean and variance.

- Calculating the variance of the predicted output from the MLR model by considering known variances of its parameters.
- Defining a metric to assess model sufficiency.

Objective 2: Quantify the influence of individual input variance on the ultimate prediction variance in a quantitative manner by:

- Identifying critical input parameters contributing the most to the final output variance.

Objective 3: Propose a labor productivity model linking product engineering design features as inputs to targeted productivity as output so as to make it appropriate for regression-based modeling by:

- Determining the contribution ratio of product features to the final productivity definition.
- Estimating the variance of the productivity prediction.
- Identifying crucial input product features based on their contribution to the total output variance.

Objective 4: Explore the capacity of the nonlinear regression method for the established model tree AI technique to quantify variance for the predicted output resulting from random sampling of training data by:

- Integrating proposed variance analysis technique for MLR equations with existing model tree algorithm.
- Introducing the Coefficient of variation as a metric to evaluate application soundness of the enhanced model tree.

Objective 5: Develop a framework for productivity modeling using the enhanced model tree method:

- Account for nonlinear relationships between product design features and labor productivity of a work package.
- Apply non-linear classifier algorithm from established model tree algorithm to decompose nonlinear productivity problem into data classes suitable for MLR applications.
- Use enhanced model tree method to predict point-value output and associated variance.
- Utilize 'coefficient of covariance' of enhanced model tree algorithm to test model applicability.

Objective 6: Introduce a methodology to address variance in total labor hour budget by consolidating variance estimates of labor productivity at the work package level:

- Determine lower and upper boundaries for cumulative labor hours budgeted at various control points throughout project lifecycle.
- Introduce a novel project control tool named the S stripe curve, visually representing estimated variance of total labor hours at specific time points of the project.
- Establish a confidence interval around the average value, providing insights into risk of labor cost budgeting due to inherent variations in labor productivity.

Overall, this research seeks to enhance the accuracy, transparency, and applicability of labor cost budgeting in industrial construction by incorporating innovative methodologies and techniques. By achieving these objectives, the study aims to contribute significantly to the field of engineering

applications, particularly in the context of explainable artificial intelligence and construction productivity modeling.

1.7 Research Methods

The research framework proposed in this study comprises three distinct modules, each tailored to address specific research objectives and corresponding research questions. As illustrated in Fig.1.5, Module 1 achieves Objective 1 and Objective 2 outlined in the thesis introduction, thereby providing answers to Research Question 1. Module 2 aims to solve Research Question 2 by fulfilling Objectives 3, 4, and 5. Finally, Module 3 accomplishes Objective 6 in order to answer Research Question 3. Details pertaining to each module are presented in discrete individual chapters, as shown in Fig. 1.6.

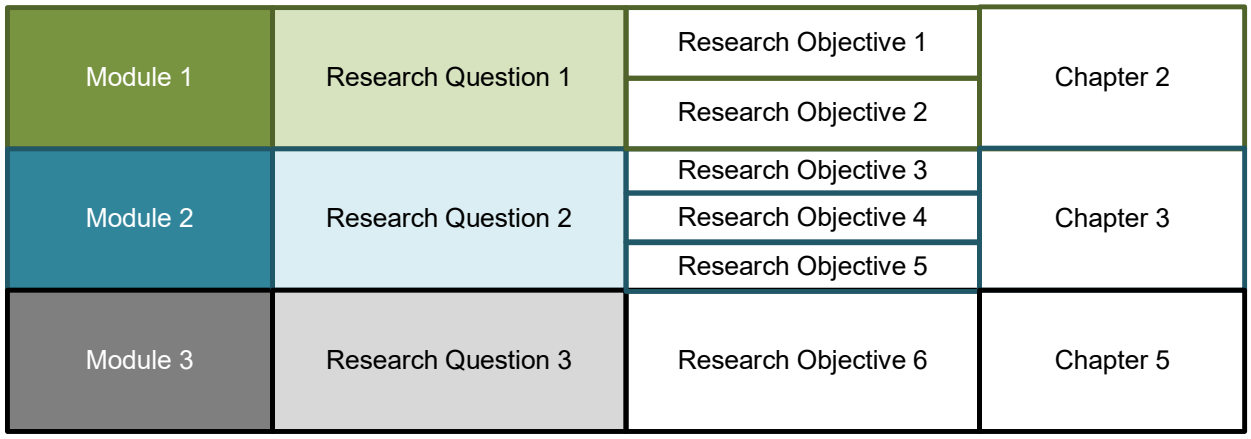


Figure 1.5: Relationship among modules, research questions and research objectives.

The schematic of the research framework is shown in Fig. 1.6.

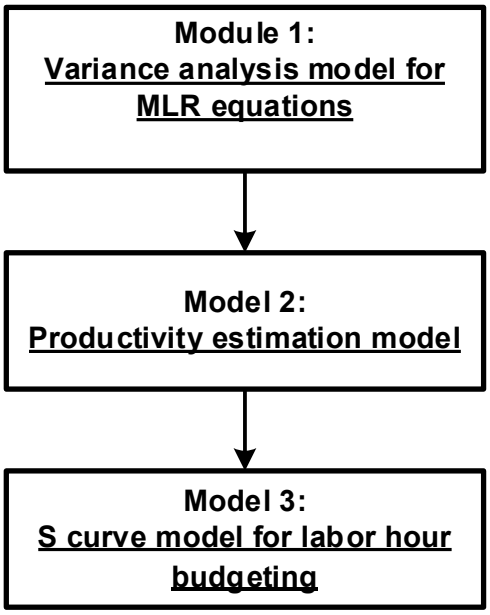


Figure 1.6: Research Framework.

As illustrated in the following figure (Fig. 1.6), Module 1 defines an analytical technique to establish the statistical depiction of the MLR model coefficients due to the variability of the input parameters in terms of mean and standard deviation. These variances of the input coefficients are responsible for the ultimate output variance. Final output variance can now be estimated by applying the error propagation theory in connection with the initial MLR equation.

Module 2 first defines the basic structure of the productivity model that relates the input design features of the fabricated products with the output productivity. The output productivity is expressed in the summation of the productivity contribution of the different product feature attributes compared with the base product's productivity contribution, as explained in the problem statement. Since the basic structure of the productivity model resembles the form of an MLR equation, the variance analysis techniques are then readily applied to find the output variance of the predictions. The presence of a higher degree of nonlinearity among input

parameters and target output may result in higher variance in predicted output, which makes the basic MLR model insufficient for practical use. Module 2 of the research also looks into integrating the data classification technique from the model tree algorithm to find data classes suitable for MLR applications. Then the model tree is enhanced in such a way that the MLR equation on each tree branch can predict the point-value output as well as the associated variance.

In Module 2, the defined work package productivity can be effectively utilized to derive the labor hour estimate with its associated variance for each specific work package. This procedure can be iterated for all work packages to obtain labor hour estimates along with their corresponding variances. To determine the overall budgeted labor hour variance for the entire project, Module 3 introduces an error propagation-based model that aggregates the labor hour variances of all work packages, thereby computing the total variance associated with the project's total labor hour requirement. By integrating current practices of estimating, scheduling, and budgeting in industrial construction, this research describes an error propagation model for calculating the varraince of the cumulative labor hours at particular time points of the project duration and establishing a confidence interval around the average value. Analogous to plotting the S-Curve, the lower bound and upper bound of the interval for cumulative labor hours budgeted at control points along the project duration can be articulated to form the S stripe, which visually portrays the risk of labor cost budget due to risks inherent in labor productivity. The application and verification of the proposed analytical methodology are illustrated with a steel fabrication project case. MC simulation is applied to the same project data to study the correlation between the two sets of results.

1.7.1 Module 1: Variance Analysis Model for MLR Equations

The generic form of the MLR equation is given in Eq. 1.6, where output Y is derived from independent input parameter X_i. Here, β_i is the slope of the MLR equation that corelate inputs with output and β_0 is the intercept.

$$Y = f(X) = \beta_0 + \sum_i \beta_i X_i \qquad (1.6)$$

Now, randomized sampling of the learning dataset such as n-fold strategies (taking a subset of the dataset in n times) in preparing the regression relationship between Y and X_i can be used to estimate the mean μ and variance $\sigma^2$of the slope parameter β:

$$\beta = (\mu, \sigma^2) \qquad (1.7)$$

Note Eq. 1.6 follows the generic format of the MLR equation and can be calibrated using the least-square optimization method (Lowe et al. 2006). To facilitate feature selection by identifying the most relevant features, applying a stepwise regression technique such as the one proposed by Mohsenijam et al. (2017) is also deemed appropriate and effective.

Now, the propagation of random error in the system follows the law of propagation of variance and covariance (POV), which can be expressed by Eq. 1.8.

$$C_y = J_{xy}\, C_x\, J_{yx}^T \qquad (1.8)$$

Here, C_y is the covariance matrix of random output y, and C_x is the covariance matrix of random input x.

Applying Eq. 1.8 in relation to Eq. 1.6, where coefficients of the MLR equation have been represented with statistical description, can derive the final variance estimation model for Eq. 1.6. The schematic of the method to establish the MLR with the capacity to estimate the associated variance of the prediction is shown in Fig. 1.6.

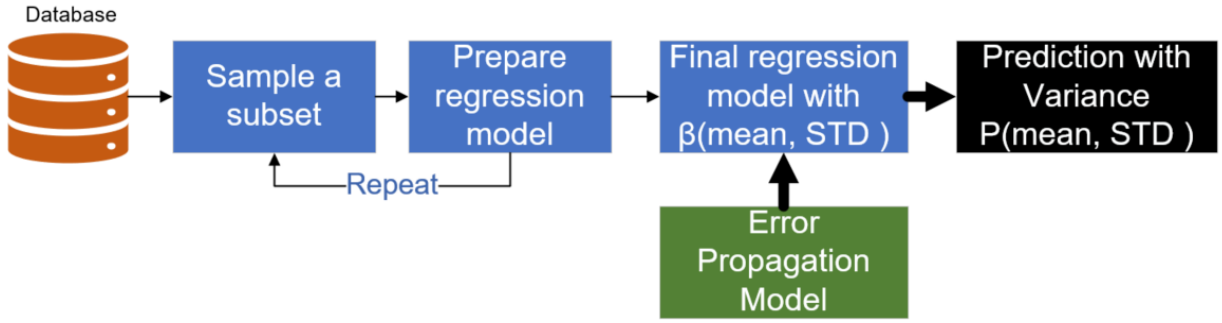


Figure 1.7: Conceptual model for variance analysis of an MLR equation.

1.7.2 Model 2: Productivity Estimating Model with Capacity to Derive Associated Variance of the Prediction

Labor productivity P_u of a particular project work package u, related with the product design feature l_n can be expressed with the following basic MLR equation:

$$P_u = f(l_n) = \beta_0 + \sum_n \beta_n l_n = \beta_0 + \sum_n \frac{P_n^0}{l_n^0} \times l_n \qquad (1.9)$$

Here, $\beta_i = \frac{P_n^0}{l_n^0}$; l_n^0 is the value base product's feature attribute n. P_n^0 is the productivity contribution of the feature attribute n of the base product. Details of the formulation of this equation are elaborated in Chapter 3. This equation resembles the basic form of an MLR equation; therefore, variance analysis technique developed in Module 1 can be applied directly. However,

when the input-output relationships are highly nonlinear, and the estimated variance of the linear model is too high, the basic model needs enhancement. Module 3 proposes a technique to enhance the model tree algorithm to represent nonlinear input and output relationships and make prediction of associated variance (Fig. 1.8).

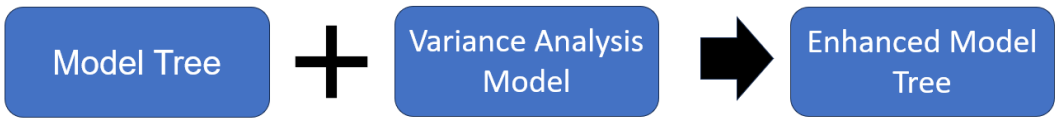


Figure 1.8: Conceptual model for preparing enhanced model tree model.

1.7.3 Model 3: S-Curve Model for Labor Hour Budgeting

In the context of industrial construction, direct labor costs are generally compiled from a comprehensive list of work items to be performed (Thomas and Sakarcan, 1994). It is noteworthy that in the current practice, the list of work items represents the estimator's interpretation of work performed by skilled trades.

The generic model of labor hour estimating for a particular work package of any project in the prefabrication application context can be expressed by Eq. 1.10 (taken from Eq. 1.1):

$$LH_T = \sum_u \left(P_u \times W_{u,}\right) \qquad (1.10)$$

Here,

$\sum LH$ = Total man hour required to complete a job;

P_u = Labor productivity (LH/Unit) for particular work package u;

W_u = Total work unit performed (quantity takeoff) for work package u.

In Module 3, a novel approach has been introduced, leveraging the error propagation theory to consolidate the labor hour variances from the work package level and calculate the overall labor hour variance for the entire project. The resulting estimated variance has been utilized to create the S stripes around the project curve (Labor Hours vs. project duration) with a given confidence level, representing the most optimistic and pessimistic scenarios of the labor hour requirements throughout the project's duration. Conceptual method to find the project S-Stipe is given in Fig. 1.9.

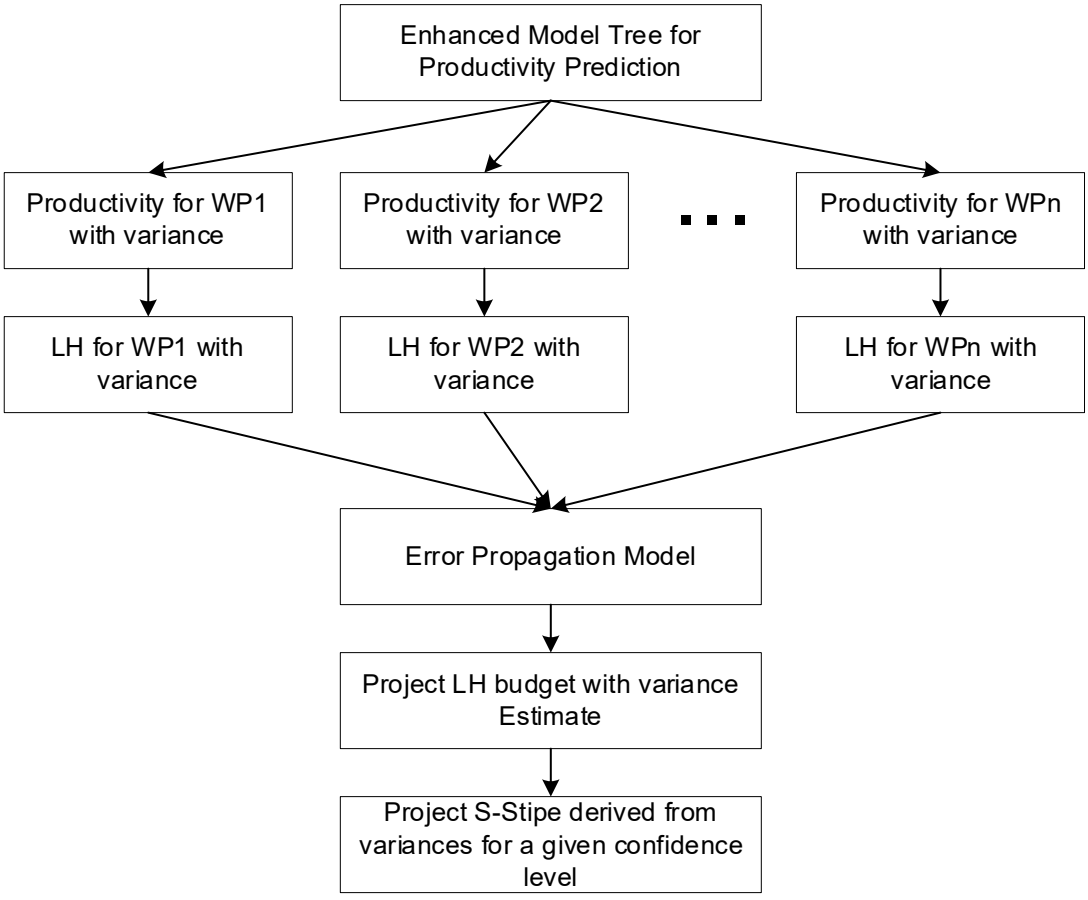


Figure 1.9: Conceptual model for preparing project S stripe curve.

1.8 Organization of the Thesis

This thesis is organized following a paper-based format that is consistent with the research framework shown in Fig. 1.6. It has a total of six chapters with Chapter 1 as the Introduction chapter and Chapter 6 as the conclusion chapter. Detailed contents of each chapter are listed as follows.

Chapter 1: This chapter introduces the productivity modeling problem for prefabrication industry, highlights the limitations of existing labor productivity prediction models in the construction industry, particularly in considering precision and variance analysis. The problem statement and research objectives are later delineated in this chapter, followed by proposed methods to address them.

Chapter 2: This chapter presents an analytical method to quantify the variance of MLR predicted outputs and elaborates its application potential in cost estimating applications. This chapter underscores the importance of precision within MLR models and the impact of individual input parameters on the variance of the prediction, thereby advancing regression modeling approaches concerning MLR variance analysis in general.

Chapter 3: The focus of this chapter is on enhancing the nonlinear regression method called model tree algorithm, a commonly applied machine learning algorithm and AI technique, to derive the variance in predicted productivity arising from sampling different datasets in productivity modeling application. The methodology decomposes the productivity problem into branches and takes advantage of the variance analysis technique introduced in the previous chapter to predict both point-value output and associated variance. The enhanced Model Tree outperforms MLR in

prediction accuracy; and is preferred to ANN because (1) the variance was analytically predicted alongside the point-value output and (2) the productivity model was explainable in terms of the reasoning logic for productivity prediction.

Chapter 4: This chapter presents a case study of developing a prediction model for the compressive strength of high-performance concrete using the enhanced model tree algorithm developed in Chapter 3. The model demonstrates high prediction accuracy and explainability with a reported variance estimate of the prediction. This application case study demonstrates the generalized application potential of the enhanced model tree algorithm in solving many civil engineering problems.

Chapter 5: The chapter explores labor budgeting in industrial construction, particularly in the context of prefabrication projects. An error propagation model is developed to calculate the standard deviation of cumulative labor hours at different project time points, enabling the establishment of confidence intervals around average values. The "S stripe" curve visually depicts the risk associated with labor cost budgeting due to labor productivity uncertainties.

Chapter 6: This chapter restates the research contributions and conclusions of this research. It also emphasizes the underlying assumptions and limitations of the presented research while suggesting potential avenues for future research. These include further advancing productivity modeling techniques and exploring the application potential of the developed algorithms in solving problems from other domains beyond the horizon of productivity modeling.

Overall, this research contributes original knowledge, innovative methods and valuable insights to regression modeling, variance analysis, and productivity modeling for prefabricated product

construction projects. It provides analytical solutions for handling precision and variance in prediction models and opens up promising directions for further research in civil engineering using machine learning techniques, with applications in explainable artificial intelligence (XAI).

Chapter 2

Variance Analysis on Regression Models for Estimating Labor Costs of Prefabricated Components

2.1 Introduction

Estimating determines the work scope and predicts financial resources needed to deliver a project from initiation to completion. An accurate cost estimate supplemented with a realistic contingency estimate, which is an assessment of risky elements that could potentially increase project cost, is vital to deliver a project within the committed budget (Creedy et al. 2010). In general, uncertainties inherent in both internal and external project environments present distinctive challenges in setting the contingency margin of an estimate in competitive tendering (Enshassi et al. 2013). Compared with the conventional stick-built practice, prefabrication mitigates uncertain factors that arise from the external project environment (e.g., the influence of the weather event, the availability of proper equipment, and competent trades) (Blismas et al. 2006). Nonetheless, the current labor-intensive work practice, along with custom-designed features of engineering products, still makes it challenging to eliminate all the uncertain factors leading to variations in productivity (Hasan and Lu 2021). In general, a crew of specialist trades performs interdependent tasks to fabricate a large number of made-to-order products based on bespoke specifications in an offsite facility. Estimating labor cost demands the prediction of crew work time. However, both

size and logic complexities inherent in prefabrication practice would impede establishing a project network model or a systematic process model to account for sufficient details (tasks, resource use, and logical relationships.) For instance, in the context of steel girder fabrication for bridge construction, the project network model would become too large and complex, making it practically unscalable for planning, communication, and scheduling analysis (Hasan and Lu 2019). Hence, instead of resorting to a process mapping and simulation approach, the practice of estimating for prefabricated components commonly relies on an analysis of historical project data: generalizing hidden patterns and implicit relationships between input factors (e.g., product features) and the output (e.g., working hours). In this context, regression models (including linear regression, nonlinear regression, and artificial neural network models) provide the appropriate quantitative methods to facilitate the process of "learning from data", resulting in cost-effective decision support to estimators (Mohsenijam and Lu 2019).

Regression offers an analytical yet practical method for modeling input-output relationships in complicated real-world systems and predicting system behaviors with mathematical equations (Barrett and Gray 1994). Various regression models have been widely accepted as the quantitative technique in modeling a wide range of application problems (El-Abbasy et al. 2014). In particular, multiple linear regression (MLR) has been the most commonly applied in support of estimating thanks to its simplicity, flexibility, and ease to communicate (Lowe et al. 2006). The accuracy of an MLR model is thoroughly evaluated based on the modeling data in terms of relative and absolute errors and statistical correlation between model outputs and target values (Olive 2017). The significance of each input factor is explicitly encoded in the first-order slope parameter, indicating its impact in direction and magnitude upon changing the predicted output. Hence,

input factors having close to zero slopes are considered less relevant to the prediction and can be removed without significantly compromising the overall model accuracy (Chan and Park 2005; Mohsenijam et al. 2017). It is noteworthy that an MLR model with high accuracy (the mean of the prediction is close to the target value) but low precision (too high the variance of the prediction) would be deemed insufficient in the context of cost estimating application. Nonetheless, precision of the MLR predicted output in terms of the variance has been rarely investigated but is of vital importance and immediate relevance to estimating applications in practice. The precision of model prediction had been largely overlooked in validating MLR models in the literature. Specifically, model validation is largely based on evaluating error terms between the predicted output and the target output without addressing the variance of the predicted output and the impact on the variance due to individual input parameters. Formulating analytical solutions to tackling the identified fundamental limitation of MLR models provides us research motivation.

With information technology advancing, the barrier of data availability in developing quantitative decision support solutions to construction estimating has been gradually removed (Mohsenijam and Lu 2019). Historical data on design features for building components (such as prefabricated products) are captured in the industry database underlying building information model (BIM) systems. Job cost data in terms of labor hours actually spent in fabrication are consistently recorded in payroll and project cost control systems. In practice, large datasets containing high-quality design features and labor cost data become inexpensively available and readily accessible, while such datasets continue to grow in size as more recently completed projects are appended (Lee et al. 2017). This extends an opportunity to develop analytical methods based on MLR aimed at analyzing the variability of model parameters in response to dynamic changes in the data

available for modeling. Additionally, the variability of the predicted cost estimate due to the variability inherent in model parameters needs to be revealed in estimating applications. The following research inquiries have arisen from practical needs in the construction industry and will be addressed in the present research.

1. Given a sufficient problem definition with relevant input features and consistent project data, a MLR model is established for predicting the labor cost based on input factors describing features of prefabricated jobs. What would be the impact of using randomly selected subsets of the whole dataset upon the fitted parameters of a MLR model? With a relatively large dataset, a subset can be randomly sampled to establish the MLR model, thereby making it possible to model the variability of the MLR model's parameters with statistical descriptors (i.e., mean and variance) instead of setting them as constants.
2. With the established MLR model, the point-value prediction based on constant values of model parameters at each particular input instance is first made. Given the mean and variance derived for parameters of the MLR model, normal distributions can be defined to denote parametric variability, thus opening up a new window of inquiry: how to analytically project the variance of the predicted output from MLR subject to the known variances on its model parameters? This variance of predicted output implies the instability of the prediction in the cost estimating application, which is commonly referred to as uncertainty, risk or contingency associated with the estimated cost.
3. Further, at each particular input instance, how to attribute the variance of the predicted output to each individual input factor of the MLR model? In other words, what is the contribution of an input factor given the variance in its associated MLR model parameter?

Analogous to applying the first-order slope in MLR to account for the sensitivity of each input factor to the predicted mean as a point-value, deriving exact solutions to address this question entails higher-order mathematics and presents a unique opportunity for computing research in civil engineering. Despite the fact that Monte Carlo simulation is frequently used in the literature for this purpose, an explicit, analytical solution has yet to be formalized.

The present research is intended to derive analytical methods based on the classic error propagation theory in order to cater to identified application needs. First, an MLR model is established with relevant and independent input factors with its accuracy verified for cost estimating. The research focus is then set on how to account for the variance of the MLR's predicted output, namely, the precision of its prediction. The variance of MLR output is quantified at a particular input point in the input space of the problem domain. The derived analytical solution is cross-checked by performing Monte Carlo (MC) simulation on the demonstration case. A metric based on the resulting variance (i.e., the ratio of standard deviation over mean) is found effective to gauge the precision of the MLR model. As long as the precision is below a certain threshold (say, 10%), one can conclude the MLR model passes the precision test. Otherwise, the model is rejected; input definition and modeling data need to be reexamined and refined prior to updating the MLR model or resorting to alternative nonlinear regression models.

For illustration of the proposed research, an application case in estimating labor cost for precast solid wall panels is presented. Wall panels are prefabricated in a precast plant with technology, environment, labor, material, and management being held relatively constant in a controlled environment. Hence, relevant input factors affecting labor hours in the fabrication of solid wall

panels are mainly design features that vary in different client orders, such as the length, width, thickness, reinforcement weight, and measurement of the openings. The base panel design is identified as the most commonly seen panel from the partner's historical data and the product features of this particular wall panel provide baselines to gauge the complexity in fabrication of possible variations of the wall panel. The average labor hours taken to precast the base panel serves as the common denominator in calculating a "*labor cost multiplier*" for each wall panel in the database. The whole data set contains 1000 records. Hence, the MLR model simply relates all the input factors– each denoting a ratio in regard to a specific product feature– with the output variable being the "labor cost multiplier". A process piping spool fabrication case was described for further demonstration of practical applicability, in which three sample spools were analyzed to verify the proposed method in estimating labor hours for various welding work items.

In the ensuing section, a critical review of literature and practice in regards to MLR applications, prefabrication estimating, and error propagation theory is described.

2.2 Review of Literature and Practice

2.2.1 Prefabrication Estimating

Industrial construction delivers projects essential to our utilities and basic industries, featuring large amounts of the highly complex process of piping, mechanical, electrical, and instrumentation work (Barrie and Paulson, 2001). Industrial construction "tends to be much more labor-intensive, though some of the largest hoisting and materials-handling equipment is also required" (Parker et al., 1984). Driven by the need for productivity improvement, industrial construction has pioneered prefabrication and offsite construction, gradually evolving into large-scale modular construction

practice (Borjeghaleh and Sardroud 2016). For instance, piping spools used to be prefabricated at a workshop on-site to reduce field handling and welding. This gradually evolved into an industrialized prefabrication plant and a module assembly yard for industrial module fabrication (Liu et al. 2016). The trend for industrialization of buildings ranging from exterior/interior wall panels to fully built home modules takes after the large-scale prefabrication practice in industrial construction. Regardless of being industrial construction or buildings, prefabrication at offsite facilities can be viewed as the basic engineering method. This paper looks into precast wall panels and prefabricated pipe spools as typical prefabrication problems in order to identify research needs for computing.

Defining work packages (WP) within the project work breakdown structure (WBS) is the first step in the estimation process. Next, with a sufficient WBS definition, work amount for each WP is scheduled and estimated, monitored, and controlled (PMI 2017). The estimator's job is to determine the quantity of work and choose the resource group to complete each individual work item (activity or task) in a particular WP. Then the total labor hours (LH) required to perform each given activity contained in a particular WP is estimated. In the application context of estimating labor hours for prefabricated products, this research follows the current practice to formalize a cost estimating model based on the product fabrication complexity factor. The basic idea is any variations against the *base product* will increase production complexity, thereby requiring additional work efforts. Therefore, a *complexity factor*, which can act as the multiplier on the base product labor cost, is directly correlated to labor hours (LH) required for fabricating a certain product type. These complexity factors are represented with a mean denoting the average

benchmark, along with a standard deviation accounting for the uncertainty in the labor hour prediction. An example case is illustrated (Fig. 1) to clarify the concept.

Case	A (Base product)	B	C	Product type
1	Yes			A
2	Yes	Yes		A-B
3	Yes		Yes	A-C
4	Yes	Yes	Yes	A-B-C

Figure 2.1: Products of varied complexity against base product in one hypothetical work package.

The above illustration shows different variation types for a particular work package in fabrication of a certain product. Here, A denotes the base product type. B and C are the complexity types that can be seen as variations of A. Case No. 1 denotes the base product. That means no additional complexity is factored, and productivity for Case 1 is set as the base productivity. In case No. 2, feature B is added on top of A. That means additional complexity is factored with the consideration of work item B. Now, if the base productivity is P_A, and the measured productivity for product type "$A - B$" is P_{A-B}, the complexity factor for the added work item B (F_{A-B}) would be $F_{A-B} = \frac{P_{A-B}}{P_A}$. In a similar manner, as shown in Case No. 3, if the measured productivity for product type "$A - C$" is P_{A-C}, the complexity factor for the added work item C (F_{A-C}) would be, $F_{A-C} = \frac{P_{A-C}}{P_A}$. Now, if there is one new product case, where both work items B and C are present in the work package, the productivity of the work package (Case No. 4) is defined with a new factor $F_{A-B-C} = \frac{P_{A-B-C}}{P_A}$.

A dataset containing one thousand precast wall panel records is tapped to analyze the precision of the MLR-based estimating model as it is continuously updated with newly appended project data.

2.2.2 Review of MLR Applications

MLR was used to predict the peak and average traffic volume (Lingras and Adamo 1996). Liu et al. (2010) applied MLR to determine the deterioration of the underground metallic pipe subject to different soil properties. Cao et al. (2018) tested the MLR model to predict the unit bid price and evaluated the prediction performance by mean absolute error (MAE), the mean absolute percentage error (MAPE), and mean square error (MSE). However, the precision of model prediction had been largely overlooked and model validation had not factored in the variance of the predicted output and its relationship with the input parameters. Choi et al. (2015) applied the data clustering technique combined with MLR model to determine the future maintenance cost of highways. To prove the robustness of the model, the predicted error sum of square (PRESS) statistics was used to determine the quality of prediction by comparing each observed response from a model calibrated with a new subset against that from the fitted model. Nonetheless, the variance in the model output resulting from using a different subset of the data in MLR was not investigated. In addition, MLR (linear regression) and artificial neural network (ANN) models were contrasted for risk assessment of the bridge maintenance performance projects (Elhag and Wang 2007). Both MLR and ANN were calibrated to model risk scores and risk categories. Pearson's correlation coefficient (r), and the root mean square error (RMSE), and MAPE were used to evaluate the accuracy performance of the proposed models. When the MLR model was presented separately with the testing data set (a subset of the modeling data), significant changes

in model accuracy (RSME, MAPE, and r values) were observed. Yet, the precision performance in terms of the model prediction variance was not assessed. Another relevant endeavor was to evaluate the performances of different data mining techniques in predicting the compressive strength of high-performance concrete by Chou et al. (2011). Nonlinear regressions (such as ANN and support vector machines) and MLR models were calibrated using manyfold cross-validation techniques, while model's prediction performance was evaluated based on r, RMSE, and MAPE (Chou et al., 2011).

Researchers have used factor analysis techniques for MLR models to enhance the accuracy of the estimated project costs by selecting the more significant factors that contribute to the estimation (Trost and Oberlender 2003, Zayed and Halpin 2005; Lowe et al. 2006). Model accuracy was reported in most cases as the standard error or variance of the final output based on the estimated residuals. Portas and Abourizk (1997) maintained that a range of productivity estimates is always preferable to the estimator instead of having a point value estimate. Therefore, providing a confidence range around the predicted point value is highly desirable. Marchionni et al. (2016) applied one quantitative equation to estimate the prediction interval of the predicted output of the MLR model developed to estimate the water supply infrastructure cost. However, this interval prediction equation is constructed with an assumption that the residual mean is zero and has a constant variance. The parametric prediction intervals made under this assumption may not perform well when the dataset is relatively small and/or has outliers with rather large residuals (Olive 2007). Mohsenijam and Lu (2019) demonstrated a modified version of the above equation to quantify the prediction interval for an MLR model in modeling labor hours for structural steel fabrication projects. It is noteworthy such established MLR variance estimation methods fall short

in accounting for the impact of the uncertainty associated with each input parameter of the model on the uncertainty of the final output. To our best knowledge, measurements quantifying the contribution of each input variable of an MLR model to the ultimate uncertainty (or variance) of the predicted output are missing in the literature. So far, the only approach available to derive such a measurement is by designing experiment scenarios and conducting sensitivity analysis (i.e., changing the input variable and observing the changes in output). Furthermore, it is common practice to develop the total estimate by aggregating the estimate for distinctly scoped work packages. Therefore, there is a practical need for calibrating a set of MLR equations –each denoting a model for one work package– in order to generate the final estimate. Variances in the MLR equations need to be aggregated in quantifying the uncertainty of the estimate, resulting in the variance of the final estimate. MLR based methods for final estimate variance analysis based on aggregating a collection of work package estimates are yet to be formalized.

2.2.3 Review of Error Propagation Theory

The law of error propagation generalizes the relationship between random variable errors and the corresponding function (Koch 1999); and provides the fundamental formula for evaluation of precision in adjustment theory (Amiri-Simkooei et al. 2016). Given a linear function, the variance or covariance of the error propagation model can be derived analytically. The error of any system output can be obtainable through functional substitution with truncated Taylor Series (considering up to first order derivative). A nonlinear function can be preprocessed with linearization by Taylor series expansion prior to applying error propagation, while it is theoretically possible to extend Taylor series expansion to any existing order in order to improve computing accuracy (Xue et al. 2015). Nonetheless, research and application of the error

propagation theory in civil or construction engineering have remained scarce. The majority of the related applications are found to fall in the field of automation control and sensor technology (Cho et al. 2004, Cantoni et al. 2007, Hasan and Lu 2017, Wai-Lok Lai et al. 2018). Hasan and Lu (2021) focused on applications of project scheduling and S-curve budgeting that is based on an activity-on-node (AON) project network model and critical path method (CPM) analysis. Error propagation theory was applied in determining variances in labor hour budgets at various stages of a project schedule, resulting in the analytical definition of S stripe instead of S-curve for the project labor cost budget.

In this research, we integrate the error propagation theory with MLR modeling in an attempt to quantify the variance of the output in connection with labor cost estimating in prefabrication, which is due to variations in complexity features of the prefabricated components. Applying first-order derivatives to approximate the propagation of random errors in the MLR model makes the solution algorithm straightforward, practical, and sufficient and eliminates the need for any additional simulation modeling effort for this purpose. The Monte Carlo (MC) simulation is used in the case study to cross-validate the results against the proposed model.

2.3 Method for Precision Analysis on MLR-based Estimating Model

The generic model of labor hour estimating in the prefabrication application context can be expressed by Eq. 2.1:

$$LH_{T,u} = \sum LH = \sum_j \left(P_{u,j} \times W_{u,j}\right) \qquad (2.1)$$

Here,

$\sum LH$ = Total man hour required to complete a job;

u = Product type;

P_u = Labor productivity (LH/Unit) for particular product type u;

$W_{u,j}$ = Total work unit performed (quantity takeoff) on product type u.

j = Work breakdown unit ID (e.g., formwork preparation, rebar preparation, laying out the insulation, pouring, and finishing)

Labor productivity $P_{u,j}$ for particular product type u, can be further elaborated with Eq. 2.2,

$$P_{u,j} = P_{(u,0,j)} \times F_{D,(u,i,j)} \qquad (2.2)$$

Here,

i = type of the product, variations, and design features that have an impact on labor hours (i.e., wall panel with variations in shape and technical design). Here, $i = 0$ denotes the default or base product with no variations.

j = Work breakdown unit ID (e.g., formwork preparation, rebar preparation, laying out the insulation, pouring, and finishing).

$F_{D,(u,i,j)}$= Design complexity factor for work breakdown unit j and variation type i for product type u.

Complexity factors can have inherent uncertainties which are represented by the mean value, μ, and standard deviation, σ

$$F_{D,(u,i,j)} = \frac{P_{D,(u,i,j)}}{P_{(u,0,j)}} = F_{D,(u,i,j)}\left(\mu_{F(u,i,j)}, \sigma_{F(u,i,j)}\right) \quad (2.3)$$

The complexity factor, $F_{D,(u,i,j)}$ is dependent on many independent feature attributes, k is added complexity to the base product u, and result in the variation of type i. Therefore, $F_{D,(u,i,j)}$ would be a function of all the complexities ($F_{(u,i,j,k)}$) resulting from the variations of different features of product type u, which is expressed as Eq. 2.4:

$$F_{D,(u,i,j)}\left(\mu_{F(u,i,j)}, \sigma_{F(u,i,j)}\right) = f\left(F_{(u,i,j,1)}, F_{D,(u,i,j,2)}, F_{D,(u,i,j,3)}, \dots\dots, F_{D,(u,i,j,k)}\right) \quad (2.4)$$

As previously mentioned in the introduction, the basic assumption of the model is that each feature variation (k) of the product u is measured against the relevant base product feature, the relationship can be expressed with Eq. 2.5:

$$F_{(u,i,j,k)} = f\left(\frac{l_{(u,i,j,k)}}{l_{(u,0,j,k)}}\right) \quad (2.5)$$

Therefore, Eq. 2.4 can be rewritten in the form of MLR equation:

$$F_{D,(u,i,j)}\left(\mu_{F(u,i,j)}, \sigma_{F(u,i,j)}\right) = \beta_{(u,i,j,0)} + \sum_{1}^{k} \beta_{(u,i,j,k)} \frac{l_{(u,i,j.k)}}{l_{(u,0,j,k)}}$$

or,

$$F_{D,(u,i,j)} = \beta_{(u,i,j,0)} + \sum_{1}^{k} F_{(u,i,j,k)} \quad (2.6)$$

Herein, $\beta_{(u,i,j,k)} = \left(\mu_{\beta(u,i,j,k)}, \sigma_{\beta(u,i,j,k)}\right)$, represents the coefficients in the MLR model to estimate the applicable labor cost multiplier against the base product; $\beta_{(u,i,j,0)} = \left(\mu_{\beta(u,i,j,0)}, \sigma_{\beta(u,i,j,0)}\right)$ represents the intercept of the MLR equation. Note Eq. 2.6 follows the generic format of the MLR

equation and can be calibrated using the least-square optimization method (Lowe et al. 2006). To facilitate feature selection by identifying the most relevant features, applying a stepwise regression technique such as the one proposed by Mohsenijam et al. (2017) is also deemed appropriate and effective.

Now, Eq. 2.1 can be rewritten as Eq. 2.7 for estimating labor cost in LH,

$$LH_{T,u} = \sum_j P_{0,(u,j)} \left(\beta_{(u,i,j,0)} + \sum_1^k F_{(u,i,j,k)} \right) W_{u.i,j} \; = \; f_T \qquad (2.7)$$

How to analyze the MLR variance based on the error propagation theory is interpreted as follows:

For any function, $y = f(x)$, measurement of the systematic error can be obtained by comparing the difference between y and its *Taylor Series* first expanded term, y_0, as expressed by the following equation (Eq. 8),

$$y - y_0 = \frac{\partial y}{\partial x}(x - x_0) \qquad (2.8)$$

Now, if y has m number of observations and each of them is dependent on n number of independent variables for x (assuming MLR is established), then the Eq. 2.8 becomes,

$$\begin{bmatrix} dy_1 \\ dy_2 \\ \vdots \\ dy_m \end{bmatrix} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \cdots & \frac{\partial y_1}{\partial x_n} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} & & \frac{\partial y_2}{\partial x_n} \\ \vdots & & \ddots & \vdots \\ \frac{\partial y_m}{\partial x_1} & \frac{\partial y_m}{\partial x_2} & \cdots & \frac{\partial y_m}{\partial x_n} \end{bmatrix} \begin{bmatrix} dx_1 \\ dx_2 \\ \vdots \\ dx_n \end{bmatrix}$$

or, $$dy = J_{xy} dx \qquad (2.9)$$

Eq. 2.9 is the general form of system error propagation where J_{xy} is called the Jacobian (Jacobian matrix) of the equation, and values of any particular measurement at an independent variable follow Gaussian distributions (e.g., Normal). Thus, the propagation of random error in the system follows the law of propagation of variance and covariance (POV), which can be expressed by Eq. 2.10.

$$C_y = J_{xy}\, C_x\, J_{yx}^T \qquad (2.10)$$

Here, C_y is the covariance matrix of random output y, and C_x is the covariance matrix of random input x.

Now, to examine the total error in final labor hours estimated due to the variance in the model coefficients, Eq. 2.10 is applied in connection with Eq. 2.7.

Therefore, Jacobian (J_T) of $LH_T\ (or\ f_T)$ is given as Eq. 2.11,

$$J_T = \begin{bmatrix} \frac{\partial f_T}{\partial \beta_{0,i}} & \frac{\partial f_T}{\partial \beta_{1,i}} & \frac{\partial f_T}{\partial \beta_{2,i}} & \cdots & \frac{\partial f_T}{\partial \beta_{k,i}} \end{bmatrix} \qquad (2.11)$$

Given all the coefficients are independent, the covariance matrix of the Eq. 2.7 is expanded in Eq. 2.12,

$$C_T = \begin{bmatrix} var(\beta_{0,i}) & Cov(\beta_{1,i},\beta_{0,i}) & Cov(\beta_{2,i},\beta_{0,i}) & & Cov(\beta_{k,i},\beta_{0,i}) \\ Cov(\beta_{0,i},\beta_{1,i}) & var(\beta_{1,i}) & Cov(\beta_{2,i},\beta_{1,i}) & \cdots & Cov(\beta_{k,i},\beta_{1,i}) \\ Cov(\beta_{0,i},\beta_{2,i}) & Cov(\beta_{1,i},\beta_{2,i}) & var(\beta_{2,i}) & & Cov(\beta_{k,i},\beta_{2,i}) \\ & \vdots & & \ddots & \\ Cov(\beta_{0,i},\beta_{k,i}) & Cov(\beta_{1,i},\beta_{k,i}) & Cov(\beta_{2,i},\beta_{k,i}) & & var(\beta_{k,i}) \end{bmatrix}$$

$$= \begin{bmatrix} var(\beta_{0,i}) & & & & \\ & var(\beta_{1,i}) & & & \\ & & var(\beta_{2,i}) & & \\ & & & \ddots & \\ & & & & var(\beta_{k,i}) \end{bmatrix} \quad (2.12)$$

Now the total variance ($\sigma_T{}^2$) can be extended as Eq. 2.13 from Eq. 2.10, resulting in Eq. 2.14 and Eq. 2.15.

$$\sigma_T{}^2 = \begin{bmatrix} \frac{\partial f_T}{\partial \beta_{0,i}} & \frac{\partial f_T}{\partial \beta_{1,i}} & \frac{\partial f_T}{\partial \beta_{2,i}} & \cdots & \frac{\partial f_T}{\partial \beta_{k,i}} \end{bmatrix} \begin{bmatrix} var(\beta_{0,i}) & & & & \\ & var(\beta_{1,i}) & & & \\ & & var(\beta_{2,i}) & & \\ & & & \ddots & \\ & & & & var(\beta_{k,i}) \end{bmatrix} \begin{bmatrix} \frac{\partial f_T}{\partial \beta_{0,i}} \\ \frac{\partial f_T}{\partial \beta_{1,i}} \\ \frac{\partial f_T}{\partial \beta_{2,i}} \\ \vdots \\ \frac{\partial f_T}{\partial \beta_{k,i}} \end{bmatrix} \quad (2.13)$$

or,

$$\sigma_T{}^2 = \left(\frac{\partial f_T}{\partial \beta_{0,i}}\right)^2 var(\beta_{0,1}) + \left(\frac{\partial f_T}{\partial F_{\beta_{1,i}}}\right)^2 var(\beta_{1,1}) + \left(\frac{\partial f_T}{\partial \beta_{2,i}}\right)^2 var(F_{3,1}) + \cdots + \left(\frac{\partial f_T}{\partial \beta_{k,i}}\right)^2 var(F_{\beta_{k,i}}) \quad (2.14)$$

or, $$\sigma_T{}^2 = G_{0,i} + G_{1,i} + G_{2,i} + \cdots + G_{k,i}; where, G_{k,i} = \left(\frac{\partial f_T}{\partial F_{k,i}}\right)^2 var(F_{k,i}) \quad (2.15)$$

Here, $G_{k,i}$ denotes the contribution to the total variance attributable to each individual factor's variance, $var(F_{k,i})$ for work breakdown unit j and product variation type i.

The influence factor (*IF*) of the variance of each complexity factor is defined as in Eq. 2.16. *IF* denotes the contribution in percentage of the variance of each input complexity to the total variance in the LH estimate.

$$IF_{k,i} = \frac{G_{k,i}}{{\sigma_T}^2} \qquad (2.16)$$

2.4 Recapitulating Application Steps

The *LH* variance estimating framework described above is developed by applying MLR combined with error propagation theory, which starts with the formulation of the labor hour estimating model and ends with verifying the variance of the model's prediction, as follows:

Step 1: Identify the work package u for which the estimating model needs to be formulated.

Step 2: Identify the number of the independent variables $\left(l_{(u,k)}\right)$ that contributes to the complexity of the product. Herein, the product complexity reflects LH requirement changes due to the shift in the contributing attribute value.

Step 3: Define one base product (can be the most common construction product constructed or built in the production facility) and its attributes along with base productivity ($P_{base,(u,j)}$) value, which can be compared against other variations of the products to define the product complexity factor ($F_{D,(u,i)}$) for a particular variation.

Step 4: Define the number of independent iterations N and the size of data (randomly selected) n used in each iteration in order to formulate the MLR model linking the input variables (complexity contributing attributes $l_{(u,k)}$) with the final output (product complexity $F_{D,(u,i)}$).

Step 5: Run the MLR regression analysis to formulate the product complexity estimation model as shown in Eq. 2.6 for a set of random data (n). Note, only relevant attributes should be considered in formulating Eq. 6. Running this process for N times will give the estimate of the

variances. $(\mu_{\beta(u,i,k)}, \sigma_{\beta(u,i,k)})$ of the regression coefficients $(\beta_{(u,i,k)})$. These variances of the regression coefficients contribute to the variance of final product complexity estimate$(\mu_{F(u,i)}, \sigma_{F(u,i)})$.

Step 6: Readily plug in the mean values of the regression coefficients into Eq. 2.7 to compute the mean LH.

Step 7: Apply the error propagation formula (Eq. 2.13) to estimate the variance $(\sigma_T{}^2)$ of the mean LH resulting from Step 6.

Step 8: Compare the obtained maximum practical variance $(\sigma_T{}^2)$ of the estimated LH_T with a predefined allowable variance threshold. If the estimated variance is less than the threshold, this model described in Eq. 2.7 is accepted.

Step 9. Compute the influence of the variance of the coefficients (which represents the variability of the input complexity in dimension) on the total variance accumulated on the estimated LHusing Eq. 2.16.

Suppose the variance exceeds the acceptable threshold, in that case, the MLR model can be improved by applying complementary data mining techniques (e.g., using various combinations of input variable sets [Chan and parker 2005; Mohsenijam et al. 2017), applying classification techniques to find different ranges of data clusters (Lu et al. 2019; Huang and Hsieh 2020), etc.]. Otherwise, a nonlinear estimating model can be adopted [e.g., scaling the data set using log

functions, applying nonlinear regression techniques (Kim at al. 2004; Bayram and Al-Jibouri 2016)]. The proposed method consisting of major steps is shown in a flowchart (Fig. 2.2).

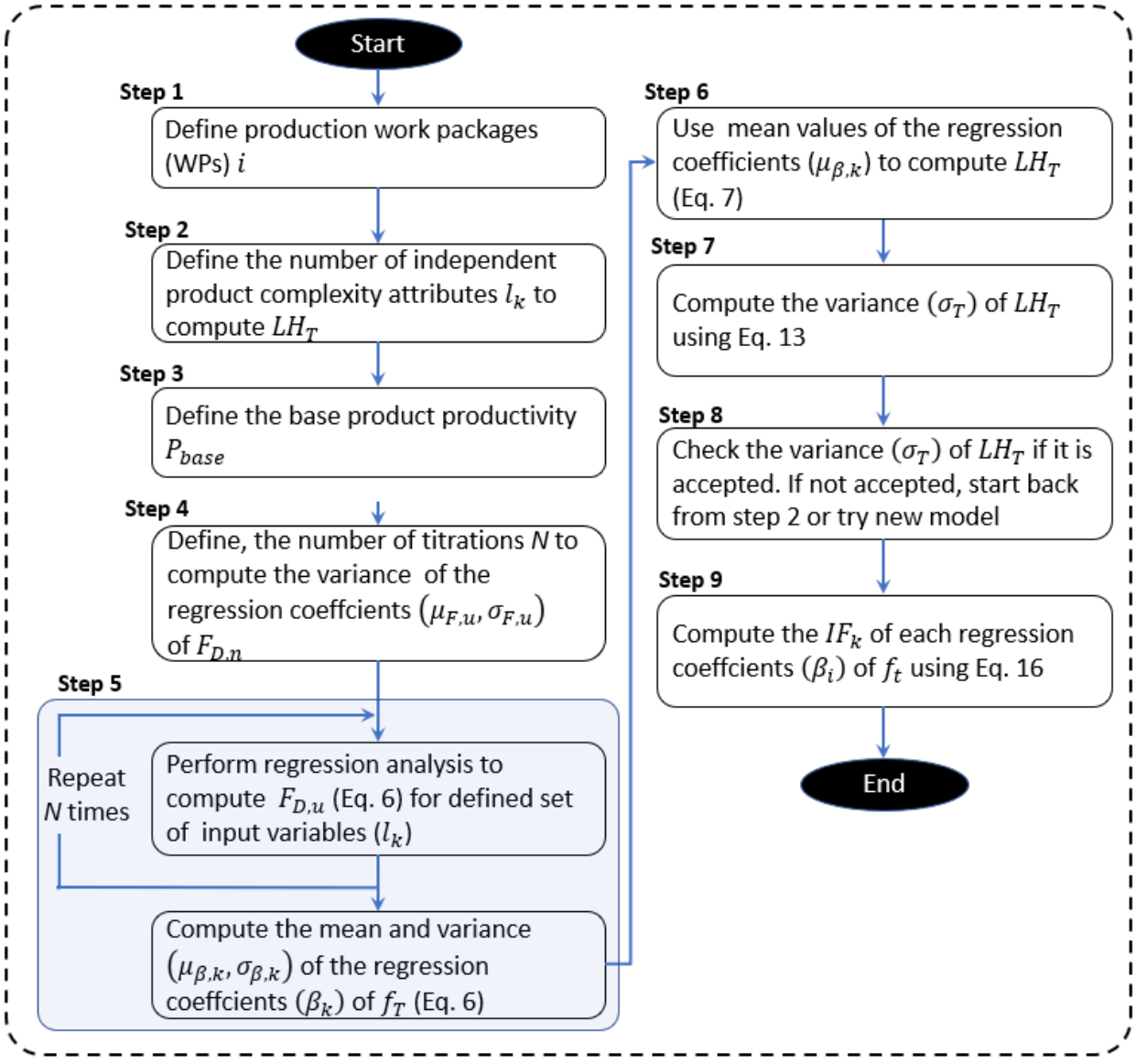


Figure 2.2: Major steps in the proposed method for determining IFI of the regression coefficients.

2.5 Application Case

In collaboration with one of Canada's major precast product producers, a data set of 1000 instances of precast wall panel products commonly used in building construction were prepared for demonstrating the application of the proposed research. Each instance has a record consisting of three attributes ($l_1, l_2, and\ l_3$) and labor hour estimate (LH_i) value. Table 1 summarizes data properties. Herein, $\boldsymbol{P_i}$ is the total labor hour required to produce per square meter floor of the wall panel. $\boldsymbol{l_1}$ can be thickness (in mm), $\boldsymbol{l_2}$ horizontal reinforcement (kg per sq m), $\boldsymbol{l_3}$ vertical reinforcement (kg per sq m). The schematic of a typical solid wall panel is given in Fig. 2.3 (plan view) and Fig. 2.4 (reinforcement details). The solid bunker type wall panel ("bunker" is industry jargon referring to the rectangular shape without any opening, solid wall panel) is the simplest type of wall panel that the partner company produces. The length, width, thickness, horizontal and vertical rebar details (weight and spacing of the rebar) are the attributes extracted from the company BIM data repository. The labor hour information for making each defined type of wall panel is extracted from the company's job costing system.

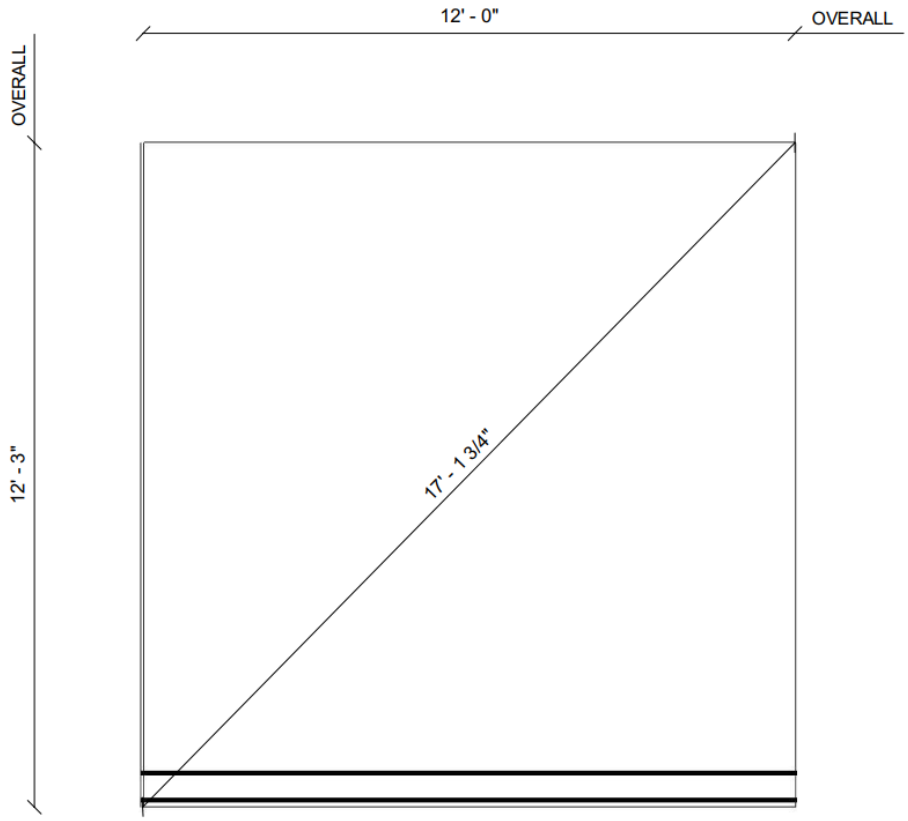
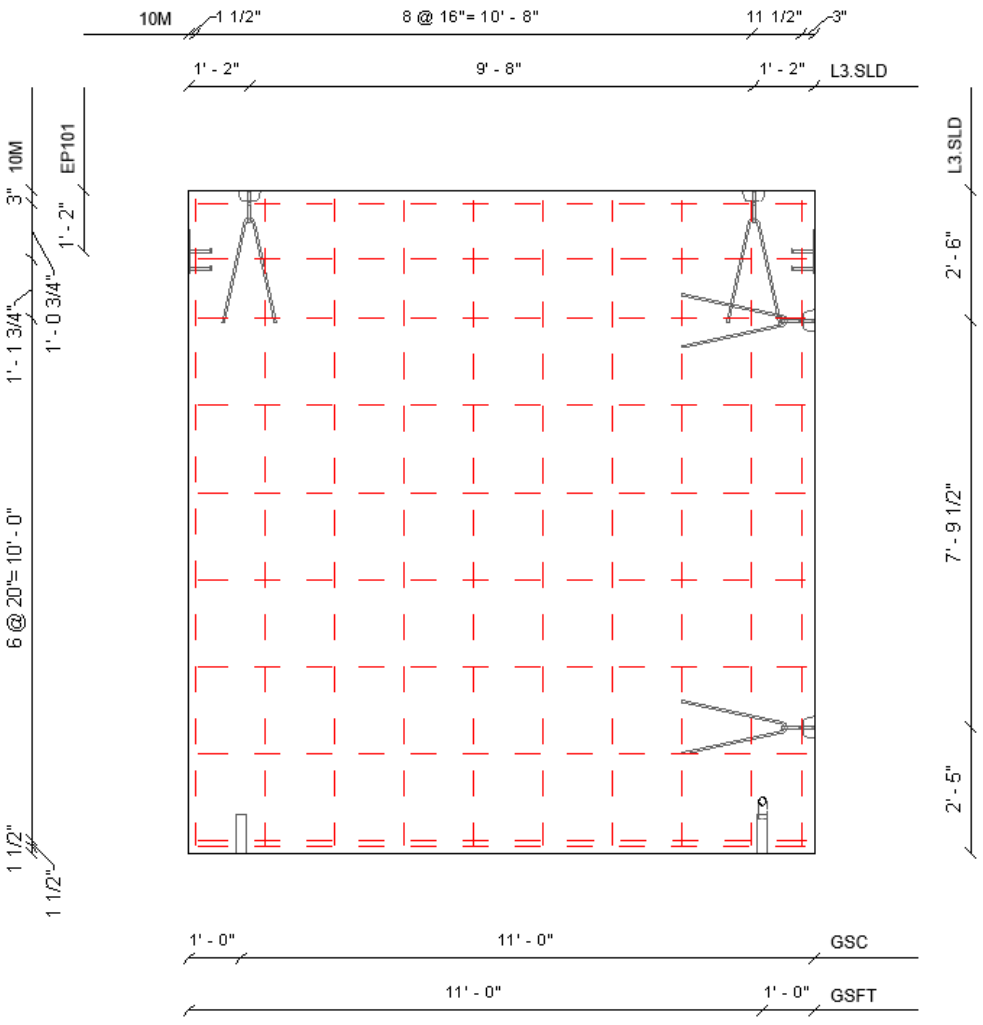


Figure 2.3: Plan view of a typical solid wall panel.



REBAR GRADE 400					
DIRECTION	TYPE	LENGTH (ft - in)	LENGTH (mm)	COUNT	TOTAL WEIGHT (kg)
Vertical	10M	11' - 11"	3632	22	62.66
Horizontal	10M	11' - 8"	3556	20	55.77

(b)

Figure 2.4: Reinforcement details of a typical solid wall panel.

Table 2.1 Data properties of the example case.

Properties	**Attributes**			**Productivity, (P_i)**
	l_1	l_2	l_3	
Minimum	3	6	11	3.49
Maximum	20	10	16	135.43
Average	11.59	8.03	13.57	69.02
Standard deviation	5.28	1.45	1.73	19.71

In this case, correlation analysis was performed for each pair of attributes, and results are summarized in Table 2.2, showing that all the correlation values are below 0.23, which indicates there is no significant correlation between the identified attributes. The two-tail t-test was also performed to check the significance of the difference between any two attributes. The results are summarized in the bottom part of Table 2.2, showing that all correlation coefficients r (absolute value) and regression coefficient (β) are nearly zero, and p-values from the two-tail t-test are below 0.05. This further corroborates that there is no correlation among the input variables, and the results are significant. Therefore, the input variables are verified to be independent of one another.

Table 2.2 Correlation analysis results for three attributes of the data set.

Attributes	$\boldsymbol{l_1}$	$\boldsymbol{l_2}$	$\boldsymbol{l_3}$
Correlation coefficient, r			
l_1	1		
l_2	0.02119	1	
l_3	0.03475	0.02757	1
Regression coefficient			
l_1	1		
l_2	0.00573	1	
l_3	-0.01125	0.03299	1
(Two tail $t-\ stat$ and $p-values$)			
l_1	-	-	-
l_2	(1.96, 2.24×10^{-80})	-	-
l_3	(1.96, 9.96×10^{-29})	(1.96, 0)	-

The proposed method is now applied to establish the labor hour estimating MLR equation and compute the variance. The step-by-step application procedure as per the Fig. 2.2 is explained with sample calculations elaborated below:

Step 1: for this example, we define only one work package (WP). Therefore, u = 1

Step 2: there are a total of three independent variables to determine the total labor hour (LH_T). Therefore, $k = 3$

Step 3: for this example case, we assume the average productivity of the entire dataset as the base productivity and average attribute values as the base dimensions. Therefore, the case data set, P_0= 69.02, and $l_{0,1}$= 11.59, $l_{0,2}$= 8.03, $l_{0,3}$= 13.57.

Note if all the input variables selected to construct the estimation model are clearly identifiable in any industry average productivity database (like RS-means). Data from such a database can also be employed to define the base (productivity) rate.

Step 4: we decide that a subset of 100 random records ($n = 100$) from the total data set will be used to perform the MLR analysis resulting in the regression equation for product complexity factor, $F_{D,(u,i,j)}$. Besides, there will be in total $N = 100$ repetitions of this process to compute the variance of the regression coefficients $(\beta_{(u,i,k)})$. Therefore, as per Eq. 2.6, the product complexity factor estimation equation ($F_{D,(u,i)}$) for this case example would become,

$$F_{D,(u,i)} = \beta_{(i,0)} + \left(\beta_{(i,1)}\frac{l_{(i,1)}}{l_{0\,(1)}} + \beta_{(i,2)}\frac{l_{(i,2)}}{l_{0\,(2)}} + \beta_{(i,3)}\frac{l_{(i,3)}}{l_{0(3)}}\right) \qquad (2.17)$$

According to Eq. 2.7 (for the case product), the new LH estimation model would be,

$$LH_T = P_0\left[\beta_{(i,0)} + \left(\beta_{(i,1)}\frac{l_{(i,1)}}{l_{0\,(1)}} + \beta_{(i,2)}\frac{l_{(i,2)}}{l_{0\,(2)}} + \beta_{(i,3)}\frac{l_{(i,3)}}{l_{0\,(3)}}\right)\right] W \qquad (2.18)$$

Here, for this example case, all the values (attribute) for the base product are, $P_0 = 69.02$, $l_{0\,(1)} = 11.59$, $l_{0\,(2)} = 8.03$, $l_{0\,(3)} = 13.57$.

Step 5: we select a subset of 100 records and perform the MLR equation for calculating product complexity factor $F_{D,(u,i,j)}$ and repeat the process 100 times to compute the mean and variance of the regression coefficients $\left(\beta_{(u,i,k)} = \left(\mu_{\beta(u,i,k)}, \sigma_{\beta(u,i,k)}\right)\right)$. The regression coefficients derived from the first ten iterations as per Eq. 2.17 are summarized in Table 2.3. From Table 2.3, it is evident that the derived coefficient values ($\beta_{(u,i,k)}$) for Eq. 2.17 vary significantly from iteration to iteration.

Furthermore, a hypothesis test was conducted to determine the significance of each coefficient in the calibrated linear regression equation. A linear regression t-test was performed to determine the validity of the hypothesis. The test hypothesis was confirmed by comparing the P-values with a desired significance level (α) derived from t statistics (Olive 2017). Here, the P-value is the probability of observing a sample statistic as extreme as the test statistic. The objective of the hypothesis test is to determine whether there is any significant linear relationship between an independent variable $l_{(i,k)}$ (in Eq. 2.17) and the dependent variable $F_{D,(u,i)}$. The coefficient value (the slope of the regression line $\beta_{(i,k)}$) equal to zero means there is a significant relationship between the independent and the dependent variable. Hence, the null hypothesis (H_0) for this case would be zero, and the alternate hypothesis (H_a) asserts that the slope would not be equal to zero. The ratio between the estimated mean ($\boldsymbol{\mu_{\beta(i,k)}}$) and the standard deviation ($\sigma_{\beta(k)}$) represents the t (stat) value, can be estimated using Eq. 2.19,

$$t = \frac{\boldsymbol{\mu_{\beta(i,k)}}}{\sigma_{\beta(k)}} \qquad (2.19)$$

Table 2.3 Values of the regression coefficients for first 10 iterations.

Iteration No. (N)	**Regression Coefficients ($\beta_{(u,i,k)}$)**			
	$\beta_{(i,0)}$	$\beta_{(i,1)}$	$\beta_{(i,2)}$	$\beta_{(i,3)}$
1	-0.148	0.285	0.343	0.535
2	-0.203	0.214	0.496	0.479
3	-0.064	0.098	0.419	0.577
4	0.375	0.228	-0.243	0.625
5	0.147	0.113	0.269	0.476
6	0.068	0.142	0.192	0.601
7	-0.166	0.202	0.319	0.668
8	-0.004	0.264	0.265	0.498
9	0.201	0.263	0.193	0.333
10	0.451	0.136	0.201	0.193

It is also important to note the degree of freedom of the relationship in finding the p- values from the standard t-distribution. The degree of freedom can be calculated using Eq. 2.20, where n is the number of observations, k' = number of independent variables. The estimated p values can be compared with the significance level α. If the p-value is less than α, we can reject the null hypothesis and accept the alternate hypothesis. Otherwise, we will accept the null hypothesis.

$$df = (n - k' - 1) \qquad (2.20)$$

For example, as per Eq. 2.20, the t-test statistic for intercept $\beta_{(i,0)}$ is: 0.047/0.081=0.581, and the corresponding P-value obtained from the standard two-tail distribution is 0.55607, which is greater than 0.05 (significance level). That means we cannot reject the null hypothesis. In other words, there is a significant chance that the value of $\beta_{(i,0)}$ is zero; therefore, we can consider the contribution of this coefficient is insignificant in the formulated MLR equation, and the intercept can be removed from the regression equation.

We checked the p values of the regression coefficient and kept the corresponding input variables that were found to be significant (p-value is less than 0.05 significance level). In this example case, all the input variables are kept, except for the intercept ($\boldsymbol{\beta_0}$), which is discarded as the p-value is greater than 0.05 (insignificant). The regression analysis results are summarized in the following table (Table 2.4).

Table 2.4 Regression analysis results summary (round 1).

Variables	Coefficients	$\mu_{\beta(k)}$	$\sigma_{\beta(k)}$	*t Stat*	*P-value*	*Remarks*
Intercept	$\beta_{(i,0)}$	0.047	0.081	0.581	0.5607	Insignificant
l_1	$\beta_{(i,1)}$	0.194	0.018	10.737	1.59E-25	Significant
l_2	$\beta_{(i,2)}$	0.241	0.045	5.280	1.58E-07	Significant
l_3	$\beta_{(i,3)}$	0.517	0.064	8.026	2.82E-15	Significant

After eliminating the insignificant variables in the MLR model, we selected a subset (100 records) of the entire data set, performed the regression analysis, and repeated the process for 100 independent iterations. The updated results are summarized in Table 2.5, and all variables are verified to be significant (P - values of the coefficients $\beta_{(i,k)}$ are less than 0.05). Therefore, the obtained MLR coefficient values ready to plug in Eq. 2.17 are: $\beta_{(i,0)}$= 0, $\beta_{(i,1)}$= (0.196, 0.017), $\beta_{(i,2)}$= (0.256, 0.037), and $\beta_{(i,3)}$= (0.546, 0.040).

Table 2.5 Regression analysis results summary (round 2).

Variables	Coefficients	$\mu_{\beta(k)}$	$\sigma_{\beta(k)}$	*t Stat*	*P-value*	*Remarks*
l_1	$\beta_{(i,1)}$	0.196	0.0176	11.175	2.1116E-27	Significant
l_2	$\beta_{(i,2)}$	0.256	0.0377	6.795	1.8549E-11	Significant
l_3	$\beta_{(i,3)}$	0.546	0.0400	13.665	4.2912E-39	Significant

2.5.1 Error Propagation Application for Variance Estimate

The variance of the predicted labor hour depends on the values of the input variables. In this case, the variance is estimated for three wall panel fabrication scenarios, namely: Scenario 1 for the maximum complex product (all l_k values are maximum), Scenario 2 for the minimum complex product (all l_k values are minimum), and Scenario 3 for the base product (l_k values are equal of $l_{base,k}$). In each scenario, a total of 100 work units (W) of the defined product type is assumed for deriving total labor hour.

Step 6 Plugging in the values of $l_{(i,1)}$, $l_{(i,2)}$, $l_{(i,3)}$, and W in Eq. 2.18 gives rise to the mean LH estimate for the three different scenarios of product complexities, respectively.

For example, for the minimum complex product (S1: variation type $i = min$), $l_{(i,1)} = 3$, $l_{(i,2)} = 6$, and $l_{(i,3)} = 11$. Therefore, the average LH for this product type (S1) would be (using Eq. 2.18):

$$LH_T = P_0 \left(\beta_{(i,1)} \frac{l_{(i,1)}}{l_{0\,(1)}} + \beta_{(i,2)} \frac{l_{(i,2)}}{l_{0\,(2)}} + \beta_{(i,3)} \frac{l_{(i,3)}}{l_{0\,(3)}} \right) W$$

Or,

$$LH_{T,S1} = 69.02 \left(0.196 \frac{3}{11.59} + 0.256 \frac{6}{8.03} + 0.546 \frac{11}{13.57} \right) 100 = 4725$$

Results for all three scenarios are shown in Table 2.6.

Table 2.6 Summary of the LH estimate for three different scenarios.

Scenario ID	Scenario Remark	l_1	l_2	l_3	Product Complexity, $F_{D,(i)}$	Total, *LH* (mean)	Standard deviation of estimated *LH*
S1	Minimum product complexity $i = min$	3	6	11	0.686	4732	298
S2	Base product complexity $i = base$	11.59	8.03	13.57	1.000	6899	398
S3	Maximum product complexity $i = max$	20	10	16	1.303	8995	505

Step 7: in this step, Eq. 2.13 (specified in Eq. 2.18 for the current case) is used to find the variance around the estimated LH for the three scenarios. The generic variance calculation model is expanded for this example case:

$$\sigma_T{}^2 = \left(\frac{\partial f_T}{\partial \beta_{0,i}}\right)^2 var(\beta_{0,1}) + \left(\frac{\partial f_T}{\partial F_{\beta_{1,i}}}\right)^2 var(\beta_{1,1}) + \left(\frac{\partial f_T}{\partial \beta_{2,i}}\right)^2 var(F_{3,1}) + \left(\frac{\partial f_T}{\partial \beta_{k,3}}\right)^2 var(F_{\beta_{k,3}}) \quad (2.21)$$

Taking the square root, we have the standard deviation (Standard Deviation) of the estimated total LH, for Scenario 1

$$\sigma_T =$$

$$\sqrt{\left(P_0 \frac{l_{(i,1)}}{l_{0\,(1)}} \times W\right)^2 var(\beta_{1,1}) + \left(P_0 \frac{l_{(i,2)}}{l_{0\,(2)}} \times W\right)^2 var(F_{3,1}) + \left(P_0 \frac{l_{(i,3)}}{l_{0\,(3)}} \times W\right)^2 var(F_{\beta_{k,3}})}$$

or,

$$\sigma_T = P_0 \times W_r \sqrt{\left(\frac{l_{(i,1)}}{l_{0\,(1)}}\right)^2 var(\beta_{0,1}) + \left(\frac{l_{(i,2)}}{l_{0\,(2)}}\right)^2 var(\beta_{1,1}) + \left(\frac{l_{(i,3)}}{l_{0\,(3)}}\right)^2 var(F_{\beta_{k,3}})}$$

$$\sigma_T = 69.02 \times 100 \sqrt{\left(\frac{3}{11.59}\right)^2 0.017^2 + \left(\frac{6}{8.03}\right)^2 0.037^2 + \left(\frac{11}{13.57}\right)^2 0.040^2} = 298.23$$

The last column of Table 2.6 shows the values of the standard deviation for the estimated LH in all the three scenarios.

Step 8: In this example case, we assume that the maximum allowed threshold for standard deviation of the estimated total labor hour LH_T is at 10% in order to accept the estimating model presented in Eq. 2.18. For all the three scenarios (Table 6), the estimated standard deviation of the estimated LH_T is found to be less than the 10% of the predicted mean [e.g., $(\sigma_{T,S3} = 505) < (10\%\ of\ LH_T\ for\ S3\ =\ 899)$]. Therefore, the LH_T estimation model (Eq. 2.18) is accepted.

Step 9: Influence of each individual attribute in total variance estimate can be measured using Eq. 2.16, which is elaborated for this case as Eq. 2.21.

$$IF_{k,i} = \frac{G_{k,i}}{{\sigma_T}^2}$$

or

$$IF_{k,i} = \frac{\left(\frac{\partial f_T}{\partial F_{k,i}}\right)^2 var(F_{k,i})}{{\sigma_T}^2}$$

Therefore, for Scenario 1, and l_1, the $IF_{1,S1}$ value would be,

$$IF_{1,S1} = \frac{\left(P_0 \frac{l_{(i,1)}}{l_{0\,(1)}} \times W_r\right)^2 var(\beta_{1,1})}{{\sigma_T}^2}$$

$$= \frac{\left(69.02 \times 100 \times \frac{3}{11.59}\right)^2 (0.018)^2}{298.23^2} = 0.011 = 1.1\%$$

Similarly,

$$IF_{2,S1} = \frac{\left(69.02 \times 100 \times \frac{6}{8.03}\right)^2 \times 0.038^2}{298.23^2} = 0.426 = 42.6\%$$

and,

$$IF_{3,S1} = \frac{\left(69.02 \times 100 \times \frac{11}{13.57} \times 0.040\right)^2}{298.23^2} = 0.426 = 42.6\%$$

Table 2.7 has the summary of $IF_{k,i}$ results for all three scenarios.

Table 2.7 Influence of each individual attributes on the total in calculating the total LH.

Scenario ID	**Scenario Remark**	$IF_{k,i}$		
		$\boldsymbol{l_1}$	$\boldsymbol{l_2}$	$\boldsymbol{l_3}$
S1	Minimum product complexity $i = min$	1.1%	42.6%	**56.3%**
S2	Maximum product complexity $i = max$	17.2%	41.3%	**41.5%**
S3	Base product complexity $i = base$	9.3%	42.7%	**48.0%**

From Table 2.7 it is observed on the three scenarios, input factors l_3 and l_2 exert more significant influence on the variance of LH predicted by the MLR, while the contribution of l_1 to the predicted LH variance is relatively less considerable.

2.5.2 Cross Checking against Monte Carlo Simulation

The data used in the illustration case (Table 2.1) were analyzed by applying Monte Carlo (MC) simulation directly on Eq. 2.18. Total LH and their variances for the three scenarios of product complexities were derived independently so as to enable cross verification. Input values for the MC simulation model are summarized in Table 2.8.

Table 2.8 Input values for the MC simulation analysis to estimate total labor hour.

Scenario	**Scenario Remark**	l_1	l_2	l_3	$\beta_{(i,1)}$ $(\mu_{\beta(1)}, \sigma_{\beta(1)})$	$\beta_{(i,1)}$ $(\mu_{\beta(1)}, \sigma_{\beta(1)})$	$\beta_{(i,1)}$ $(\mu_{\beta(1)}, \sigma_{\beta(1)})$
S1	Minimum product complexity $i = min$	3	6	11			
S2	Base product complexity $i = base$	11.59	8.03	13.57	(0.196, 0.0176)	(0.256, 0.0377)	(0.546, 0.04)
S3	Maximum product complexity $i = max$	20	10	16			

Based on 1000 simulation runs for each scenario, the derived average LH for the three scenarios is contrasted against that resulting from the proposed error propagation model in Fig. 2.5, with no significant difference found in the case of the mean values. On the other hand, the variance of

the LH estimate is observed to slightly increase with the increment of the complexity in the simulation case. Detailed results from these two methods are given in Table 2.9.

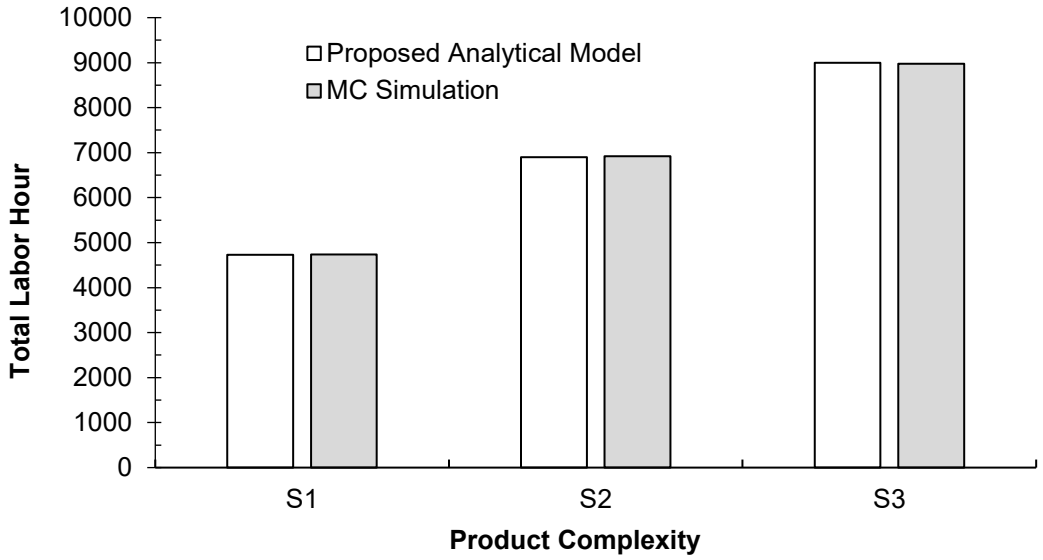


Figure 2.5: Mean LH resulting from 1000 MC simulation run and applying error propagation model.

Table 2.9 Result comparison for proposed analytical model vs. simulation model.

	S1 (Mean, Standard deviation)	**S2 (Mean, Standard deviation)**	**S3 (Mean, Standard deviation)**
LH from Proposed Analytical Model	(4798, 298)	(6899, 398)	(8995, 505)
LH from MC Simulation	(4737, 310)	(6917, 395)	(8972, 486)

2.5.3 Further Observation and Discussion

In quantifying the total labor hour estimate (LH_T) and its associated variance by applying the proposed analytical model and MC simulation, it is observed that the ratio of the standard deviation (σ_T) and estimated mean labor hour (LH_T) remains nearly constant for all three case scenarios (S1, S2, and S3). For the proposed analytical model and MC simulation model, the ratio of σ_T/LH_T is averaged 6% and 8.3%, respectively (Table 2.10).

Table 2.10 Analysis of the ratio of the standard deviation (σ_T) and estimated mean labor hour (LH_T) by the proposed analytical model and MC simulation.

Scenarios	**Proposed Analytical Model**	**MC Simulation**	**Ratio: Proposed Analytical Model/MC simulation**
	Standard deviation σ_T /Mean LH_T		
S1	6.3%	6.5%	0.97
S2	5.8%	5.7%	1.02
S3	5.6%	5.4%	1.04

It is notable that the ratio of the standard deviation derived from the proposed analytical model and the value resulting from simulation are 0.97, 1.02, and 1.04 in Scenario 1, 2, and 3, respectively. There may have some insignificant errors due mainly to the approximation in the error propagation model by keeping first-order derivative terms only. Besides, the preliminary assumption in developing the model is that all the input variables are independent (there is no

correlation between any two input variables). It is worth mentioning, as shown in Table 2.2, the regression coefficient (namely, r denoting the correlation between two variables) is not exactly zero but close enough to zero (the hypothesis was checked through statistical testing). That also accounts for the difference in the results between the proposed analytical model and MC simulation. Note one can adjust the acceptance threshold on the precision metric to a certain extent so as to account for the bias. Despite the bias, the resulting variance analysis is found effective in gauging the precision of the MLR model. As long as the precision (i.e., the ratio of standard deviation over mean) is below a certain threshold (say, 10%), one can conclude the MLR model passes the precision test. Otherwise, the model is rejected, and its input definition and data need to be reexamined and refined prior to updating the MLR model.

In this case study, all the three selected variables were found significant and sufficient to formulate the MLR model for estimating the LH (Eq. 2.18). The MLR model's parameters along with associated statistical descriptors are summarized in Table 2.5. Hence, this finding addresses the first research inquiry. It is worth mentioning that variable identification and selection is one indispensable step in problem definition and regression modeling. It is worth mentioning that variable identification and selection is one indispensable step in problem definition and regression modeling. To ensure a sufficient problem definition, it is advisable to start with the inclusion of as many variables as possible in the input space; then, significance and independence tests are performed to streamline the input factors. For instance, correlation analysis and standard t-test of the regression coefficients were applied in this study to reduce the dimensionally of the input space in regression modeling. Additionally, principal component analysis (Chan and Park, 2005) or stepwise regression (Mohsenijam et al. 2017) provides proper techniques to prune the

unnecessary input factors of the MLR model and complement the proposed method. Input factor selection and pruning analysis for MLR is taken out of the scope of this study. The second inquiry of this research is answered in Steps 6 and 7 in the case study. The point value estimate of the LH along with the variance estimate for the three application scenarios is given in Table 2.6. On the third research inquiry, the influence of the three MLR model input factors on the variance of the output are analytically derived using Eq. 2.16, with results summarized in Table 2.7. It is noteworthy that in the literature of MLR applications, such analytical findings as shown in Table 2.7 are only obtainable through using the MC random sampling in the context of checking sensitivity of model responses.

2.6 Pipe Spool Fabrication Case

For further demonstration of the practical applicability of the proposed methodology, a process piping spool fabrication case is described. The case is originally given in Lu et al. (2017) to demonstrate the productivity benchmarking and cost estimating practice in industrial construction. A processing plant construction project had all the piping spools prefabricated in a fabrication shop. Three sample spools were used to build the application case and verify the feasibility of the proposed research in estimating labor hours for fabricating piping spools (mainly fitting, welding and handling). Table 2.11 gives the takeoff summary for each spool. Note the nominal pipe size (NPS) specifies pipe inner diameter while the schedule number (Sch.) denotes pipe wall thickness.

Table 2.11 Required materials for fabricating the three sample spools.

Material Name	**Quantity**		
	Spool 1	**Spool 2**	**Spool 3**
152 mm, Sch.40 Pipe	12.1 (m)	5.4 (m)	7.3 (m)
50 mm, Sch.80 Pipe	1.1 (m)	-	
76 mm, Sch.40 Pipe	-	1.4(m)	-
50 mm, Sch.80 Elbow	1	-	-
152 mm, Sch.40 Elbow	-	2	2
50 mm, Sch.80 Flange	1	-	-
76 mm, Sch.40 Flange	-	1	-
50 mm× 152 mm, Sch.80/Sch.40 Olet	1	-	-
152 mm Hydro pipe	-	1	-
152 mm×76mm', Sch.40 Reducer	-	1	-
152 mm, Sch.40 Tee	-	1	-

The welding work package is the most significant in piping spool fabrication. The complexity of welding depends on welding type, wall thickness, and diameter of the pipe spool. All the

complexity factors for each spool case against the base weld type are given in Table 2.12. Note, the productivity of the base product (Butt Weld with 50 mm diameter and wall thickness being Schedule 80) is $P_0 = 0.43$ LH/ Unit. Each complexity factor is defined with a mean value and a standard deviation. Quantity takeoff for the welding work package for the three spool cases is given in Table 2.13.

Table 2.12 Weld complexity definition for the pipe spool case.

Variation ID, *i*	**Item for welding**	**Size (mm)**	**Wall Thickness**	**Complexity Factor (mean, standard deviation dev)**
1	Weld-neck flange	50	Sch 80	(1.05,0.09)
2	Weld-neck flange	76	Sch 80	(1.70,0.05)
3	Olet Weld	50	Sch 80	(4.20, 0.19)
4	Hydro Weld	152	Sch 40	(1.80, 0.15)
5	Butt Weld	152	Sch 40	(1.80, 0.23)
6	Butt Weld	76	Sch 40	(1.10, 0.11)
0	Butt Weld	50	Sch 80	(1, 0.03)

Table 2.13 Quantity takeoff for the welding work package on three spool cases.

Variation ID, *i*	**Quantity (counts)**		
	Spool 1	**Spool 2**	**Spool 3**
1	1	1	-
2	-		
3	1	-	-
4	-	1	-
5	1	6	3
6	-	1	-
0	2	-	-

It is worth mentioning that variances on input factors, in this case, were estimated by consulting with experienced estimators, as relevant data was not collected in the original estimating study. When a database containing historical data on spool configurations and labor hours in fabrication is available, it is advisable to characterize variances on input factors using the sampling technique given in Step 5 of the proposed methodology (shown in Fig. 2.2).

On pipe spool No. 1, as for the welding work package, the total LH required is determined as per Eq. 2.22 derived from Eq. 2.7.

$$LH_{T,u} = \sum_j P_{0,(u,j)} F_{(u,i,j)} W_{u.i,j} \qquad (2.22)$$

Here, j = Welding; u = Pipe spool, and i = variations in the welding. For welding work package of this pipe spool fabrication case, Eq. 2.23 is derived from Eq. 2.22.

$$LH_{T,Welding} = \{P_0 \times F_{(1)} \times W_{(1)}\} + \{P_0 \times F_{(3)} \times W_{(3)}\} + \{P_0 \times F_{(5)} \times W_{(5)}\} + \{P_0 \times F_{(0)} \times W_{(0)}\} \qquad (2.23)$$

Therefore, the mean LH estimate for Spool 1 would be derived as:

$$LH_{T,Welding} = \{0.43 \times (1.70) \times 1\} + \{0.43 \times (4.20) \times 1\} + \{0.43 \times (4.20) \times 1\} + \{0.43 \times (1) \times 2\} = 3.89$$

The variance of the total LH estimate as per Eq. 2.23 would be derived as:

$$\sigma_{T,Welding}{}^2 = (P_0 \times W_{(1)})^2 var(F_{(1)}) + (P_0 \times W_{(3)})^2 var(F_{(3)}) + (P_0 \times W_{(5)})^2 var(F_{(5)}) + (P_0 \times W_{(0)})^2 var(F_{(0)}) \qquad (2.24)$$

or,

$$\sigma_{T,Welding}{}^2 = 0.0015 + 0.0067 + 0.0098 + 0.00067 = 0.0186$$

Table 2. 14 summarizes LH estimate along with the associated variance estimate for all the three pipe spool cases, plus the upper and lower limits at 95% confidence level.

Variance contribution for each individual weld type on total LH can be calculated as per Eq. 2.24, with results given in Table 2.15. It is notable that Spool 3 requires only Variation 5 welding work; but for Spool 1, four variations (Variations 1,3,5,0) are applicable, with the most significant being

Variation 5 (Butt Weld Size 152 Sch 40 at 52.5%), followed by Variation 3 (Olet Weld at 35.8%), Variation 1 (Weld-neck flange at 8%), and Variation 0 (Base Butt Weld Size 50 Sch 80 at 3.6%). For Spool 2, four variations (Variations 2,4,5,6) are applicable, with the predominant variable being Variation 5 (Butt Weld Size 152 Sch 40 at 98.1%).

Table 2.14 Summary of all the LH with their variance estimate for all three pipe spool cases.

Pipe Spool ID.	**Total LH**	**Variance**	**Lower bound at 95% confidence level**	**Upper bound at 95% confidence level**
1	3.89	0.019	4.16	3.62
2	6.62	0.359	7.80	5.45
3	2.32	0.088	2.90	1.74

Table 2.15 Variance contribution for each individual weld type on total LH variance.

Variation ID, *i*	**Item**	**Size (mm)**	**Wall Thickness**	**Pipe Spool 1**	**Pipe Spool 2**	**Pipe Spool 3**
1	Weld-neck flange	50	Sch 80	8.0%	-	-
2	Weld-neck flange	76	Sch 80	-	0.1%	-
3	Olet Weld	50	Sch 80	35.8%	-	-

Variation ID, *i*	**Item**	**Size (mm)**	**Wall Thickness**	**Pipe Spool 1**	**Pipe Spool 2**	**Pipe Spool 3**
4	Hydro Weld	152	Sch 40	-	1.2%	-
5	Butt Weld	152	Sch 40	52.5%	98.1%	100.0%
6	Butt Weld	76	Sch 40	-	0.6%	-
0	Butt Weld	50	Sch 80	3.6%	-	-

2.7 Conclusions

Advances in information technology have gradually removed the barrier of data availability in rendering quantitative decision support to estimating. At present, relatively large datasets containing high-quality design features and labor cost data for industrial fabrication are readily accessible, while those datasets continue to grow in size as more recently completed fabrication projects are appended. This has provided us the opportunity to develop an analytical method aimed at (1) analyzing the variability of model parameters in response to dynamic changes in the data underpinning the model and (2) accounting for the variability of model predicted cost estimate due to the variability inherent in model parameters.

In the practical context of estimating labor hours for prefabricated products, this research formalizes a cost estimating model based on the product fabrication complexity factor. Any variations of a fabrication product against the base product will increase production complexity. Thus, a complexity factor, which acts as the multiplier on the base product labor cost, is directly

correlated to the production cost in terms of labor hour (LH) that is required for fabricating a certain product type. These complexity factors are represented with a mean (constant number) reflecting the average benchmark, along with a standard deviation denoting the uncertainty in labor hour prediction. The research is focused on accounting for the variance of the MLR's predicted output, namely, the precision of its prediction. The variance of MLR output is quantified at a particular input point in the input space.

The MLR equation derived from applying the analytical methods described in this paper represents a complexity factor-based estimating model. The newly proposed method sheds light on the variance of the MLR predicted estimate. It also provides insight on the effect of each input upon the variance of the MLR prediction. Hence, the proposed methodology eliminates the need to implement MC simulation techniques in practical applications. A process piping spool fabrication case was described for further demonstration of practical applicability, in which three sample spools were analyzed to verify the proposed research in estimating labor hours for various welding work items.

The proposed variance analysis is found especially effective to gauge the precision of the MLR model. For instance, as long as the precision (i.e., the ratio of standard deviation over mean) is below certain threshold (say, 10%), one can conclude the MLR model passes the precision test. Otherwise, the model is rejected, and its input definition and data need to be reexamined and refined prior to updating the MLR model. It is stressed the proposed variance analysis method relies on MLR as the base model; however, MLR may not be adequate in addressing highly complicated cost estimating problems where highly nonlinear relationship between input and output variables exists. Under such circumstances, the MLR model can be complemented with

more sophisticated data mining techniques (e.g., using various combinations of input variable sets, applying classification techniques to find different ranges of data clusters, etc.). Alternatively, nonlinear estimating models can be adopted (e.g., scaling the data set using log functions, applying nonlinear regression techniques like artificial neural networks, etc.). Driven by application needs, the research can be further enhanced by applying more sophisticated regression models and taking higher order Taylor expansion in applying the error propagation theory. Those extensions point to promising directions for further computing research in civil engineering.

Chapter 3

Enhanced Model Tree Modeling Technique for Quantifying Output Variances Due to Random Data Sampling: Productivity Prediction Applications

3.1 Introduction

Given a production process, productivity is generally defined as the ratio of the measurable output against the input of consumed resources. Quantifying productivity requires unambiguous specifications of the scope of work and appropriate measurements of input and output. Due to the labor-intensive nature of construction activities, labor productivity (labor hour required per unit of work) is the commonly applied productivity definition in the construction industry (Abdel-Hamid and Mohamed-Abdelhaleem 2022). Yet, productivity of a repetitive construction process could vary considerably thanks to variations in engineering design, work environment, and human factors (Portas and AbouRizk 1997; Durdyev et al. 2018). Hence, predicting productivity for a commonly practiced method in the construction field turns out to be a challenging decision, which requires prior work experiences and comprehensive evaluation of job elements specific to design features and the work environment (Song and AbouRizk 2008). To account for productivity variation in a particular application domain, productivity could be analytically modeled by multiple linear regressions (MLR), which summarized identified productivity elements in

connection with relevant input factors or job features (Edwards and Holt 2000). Nonetheless, productivity in construction still presents itself as a complicated problem that could render the use of linear models (such as MLR) to be inadequate in representation of the hidden patterns and nonlinear relationships hidden in the data (Tam et al. 2002). On the other hand, implementing nonlinear regression models or artificial intelligence (AI) methods such as artificial neural networks (ANN) would generally enhance model accuracy while compromising the model's explainability in terms of transparent reasoning logic (Naumets and Lu 2021).

The resource enumeration method is the mainstream approach for making cost budgets on resource-centric prefabrication projects (AbouRizk et al. 2001). This method involves a systematic selection of project work packages, quantity takeoffs, and productivity estimates. For industrial construction, productivity is measured by labor hours per unit of work. Multiplying the quantity takeoff with productivity yields labor hour (LH) estimates for individual work packages and hence the labor cost budget of the total project. Moreover, productivity is key to establishing activity time estimates and project schedule development. It is imperative to recognize that deviations from productivity benchmarks would translate into uncertainties and risks of the project system, which potentially exerts significant impact on project budget and schedule (Hasan and Lu 2021). Hence, adequacy and effectiveness of project planning and control is contingent upon the accuracy and precision of labor productivity.

How to characterize the precision of the estimated productivity remains vague and subjective. In civil construction cost estimating, it is common to impose a 5-10% contingency on the estimate as a rule of thumb (Ammar et al. 2023). However, high variability in productivity in the real world would defy this rule of thumb, resulting in a considerable underestimate of risks. This variability

can be determined by accounting for the primary factors contributing to productivity variation, assessing estimated ranges, and conducting a probabilistic analysis using Monte Carlo simulation (Barraza 2011). Lack of cost variation information has been identified as the major limitation of industrywide productivity and cost data services (Karlsen and Lereim 2005; Wang et al. 2012). Published industry benchmark productivity information (such as RS Means) ignores potentially the high variation in productivity.

3.1. Research Motivation

The hurdle of data availability in productivity study is being gradually overcome thanks to technological advances over the past few decades. The proliferation of building information model (BIM) and labor hours tracking automation in the construction industry -in particular, in the prefabrication of structural components and building subassemblies- has given rise to the accumulation of well structured, consistent data containing job features and labor hours in the industry. Historical data on design features for building components such as prefabricated products are recorded in the industry databases underlying BIM systems (Shen and Issa 2010). Job cost data, in terms of labor hours spent in prefabrication, are consistently tracked in payroll management systems and project control systems. In practice, large datasets containing high-quality design features and labor cost data become inexpensively available and readily accessible. Such datasets continue to grow as more recently completed projects are appended (Lee et al. 2017). This has spawned the industry need and provided the research impetus to model productivity by applying regressions and ANN and produce data driven predictive analytics on productivity performances in both short and long terms. In view of sustainable business development, productivity models are of strategic significance in terms of (1) making critical

project management decisions on estimating, scheduling, and cost budgeting; and (2) striving for better profitability, client satisfaction and business growth.

As the productivity database continuously expands, the data used for productivity modeling and analysis actually represents a random sample taken at a particular time. Given the same problem domain, different datasets can be sampled by different modelers in the same time period or by the same modeler at different time periods. As such, the productivity model calibrated by applying the same analytical method would result in different model parameters and a variance of the predicted productivity. For instance, given identical input factors, if the dataset available for MLR modeling varies in size and content, a different MLR equation was produced; given the same input settings, the predicted output varied (Hasan and Lu 2022). This has given rise to a new research problem in connection with productivity prediction by applying regressions or ANN models, namely: in addition to the accuracy (i.e., how accurate is the point-value output), the precision of a productivity prediction model is also crucial to validation and utilization of the model as decision support tool. Herein, the precision is herein defined as a specific confidence interval around the predicted point value due to the variance in sampling the dataset for model calibration. The quantification of precision of the prediction output resulting from any productivity requires the variance (standard deviation) to be known in statistics (Rodríguez et al. 2013). This present research is intended to fill this gap in knowledge by enhancing the Model Tree regarding its analytical capability of providing the variance for the predicted productivity, given a relatively large dataset available for a productivity problem. This research essentially builds on recent progresses in Model Tree applications in productivity study (Mohsenijam et al. 2021) and MLR

in variance analysis (Hasan and Lu 2022) in efforts to extend the capability of the nonlinear AI method of Model Tree in modeling construction productivity.

3.2. Research Overview

Non-parametric supervised learning methods like KNN (k-Nearest Neighbors), reliant on neighbor similarity for prediction, lack suitability for mathematically encapsulating input-output relationships (Guo et al. 2003). Classification trees categorically cluster data via binary recursive partitioning, generating tree-like prediction structures (Buntine 1992). Regression Trees, akin to classification, assign constant values to leaves, providing specific point predictions (Breiman et al. 2017). In contrast, the model tree is generally categorized as a nonlinear regression model (Chang and Kim 2011; Vanli et al. 2019), which applies binary rules to identify data classes suitable for creating an MLR model for each class. The final model is expressed as a set of MLR equations constrained by rules defining each class (Quinlan 1992). In addition, model tree is recognized as a logic-driven machine learning model which ensures transparency in model development in explaining linear and nonlinear input-output relationships, respectively. Having a clear and transparent model structure is essential to gain the trust of practitioners and promote the AI implementation in practice.

Model Tree was initially created as an extension of the classification tree technique by Morgan and Sonquist (1963), who utilized the automatic interaction detection (AID) method to construct regression trees. Regression tree produces a tree-structure-like predictive model, where the feature domain is divided into branches by nonlinear classifiers, and at the end of each branch, there are leaf nodes, each corresponding to a constant value. Model tree combines a decision tree with

regression analysis by breaking down the big data set into small subsets, so that the nonlinear input-output relationship can be represented into a series of linear multi-variate data models (Quinlan 1992). It is noteworthy that Model Tree stands out as the ideal choice of explainable AI method, outperforming other nonlinear regression and AI techniques in terms of preventing model overfitting, improving prediction accuracy, and enabling model explainability (Mohsenijam et al. 2021). Nonetheless, like other nonlinear regression models or AI methods, Model Tree only produces point-value predictions, lacking variance analysis in the predicted outputs.

In this research, a productivity problem is decomposed into branches (classes) by applying the non-linear classifier algorithm in Model Tree. Then the Model Tree is enhanced in such a way that the MLR equation on each tree branch can predict the point-value output as well as the associated variance due to the variances in the input variables (or productivity features). As illustrated in Figure 1, the enhanced model tree has n productivity features identified as inputs $(x_1, x_2, \ldots, x_n)$, which produces the point-value prediction Y along with the associated standard deviation (STD) of Y, σ_Y (square root of the variance Y).

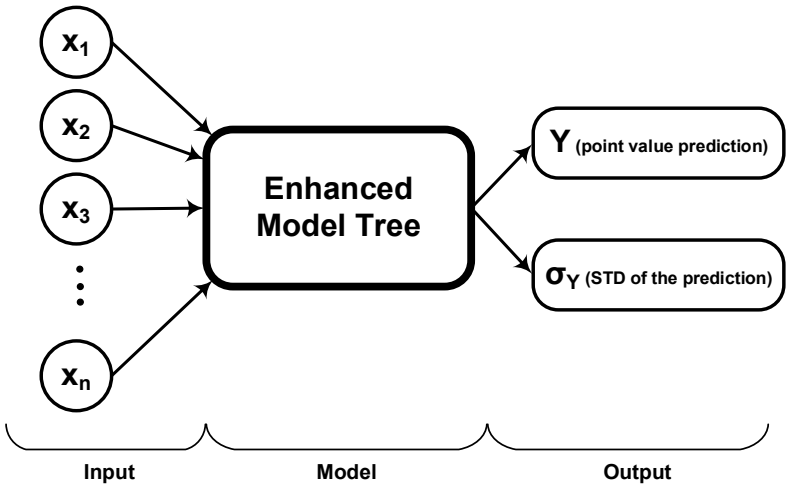


Figure 3.1: Input-output relationship of the proposed enhanced model tree.

As the environment of prefabrication operations in construction is better controlled and less susceptible to external factors (Lou et al. 2021), productivity of prefabrication of structural components and building assemblies of certain type is the chosen application domain. It is worth mentioning that prefabrication of the same-type structural components in building and infrastructure construction is still classified as "made-to-order" labor-intensive production processes; productivity could broadly fluctuate because of product variations in detailed elements or special features (Hamzeh et al. 2019).

The remainder of this paper is organized as follows. First, in the literature review section, established productivity models and their limitations are critically reviewed, along with model tree applications in investigating construction productivity and other civil engineering problems. Then the enhanced model tree framework for productivity modeling is proposed. Subsequently, a case study in the real-world context of structural steel fabrication productivity is presented to demonstrate step by step applications of the proposed methodology. To show the general applicability of the research, a second case study in piping spool fabrication productivity is described. Conclusions are drawn based on the observations from the case study. Research contributions and limitations are also summarized in the conclusion section.

3.2. Literature Review

3.2.1. Productivity Models

With a wide range of influential factors potentially accounting for the high variability of productivity in the construction project environment, developing productivity prediction models presents a long-standing challenge for construction research (Tsehayae and Fayek 2016). Applying data analytics to improve productivity prediction in construction was attempted in 1990s (e.g., Minato and Ashley 1998; Song and AbouRizk 2008; Bai et al. 2019). Generally, a regression model is established based on a selection of relevant elements identified to affect labor productivity to represent the intricate empirical relationships between different factors (both internal and external to the project) and productivity rates (Durdyev et al. 2018; Hamzeh et al., 2019). Statistical techniques coupled with linear regression models were commonly applied for productivity modeling (Smith 1999; Mohsenijam and Lu 2019). Note statistical analyses are largely restricted by the number of influencing factors that can be included while linear regression has limited capabilities to consider the combined, nonlinear effects of the influencing factors (Yi and Chan 2014). To tackle complex productivity modeling problems, nonlinear regression models as well as ANN had been widely applied, demonstrating good potential for quantitatively evaluating the effects of multiple factors in productivity prediction, especially when numerous factors could be involved in complicated nonlinear relationships (Tam et al. 2002; El-Gohary et al. 2017). ANN models generally demanded a large amount of structured data for training in modeling productivity. Besides, convoluted ANN algorithms rendered the majority of ANN models to act like "*black box*" without providing much explanation of reasoning logic. To a large degree, the *black box* had led

to user's mistrust in the productivity predicted from ANN models, hampering ANN applications in the construction industry (Naumets and Lu 2021). Another considerable hurdle in productivity modeling arose from the concepts or methods that could be subjective in nature and relevant in defining certain parameters (Guo et al. 2017). For instance, to adapt models to different project contexts described with limited data, expert system in combination with fuzzy set theory was used to estimate labor productivity in different construction activities (Christian and Hachey 1995; Fayek and Oduba 2005). In general, expert systems fell short in identifying a mapping function and generalizing rigorous solutions. In addition, rules obtained from domain experts could be subject to personal prejudices and biases (Yi and Chan 2014).

Besides mainstream quantitative methods, recent literature has introduced the hybridization of classification and regression techniques to improve the accuracy of labor productivity predictions (Elmousalami 2020; Mohsenijam et al. 2021). A hybrid intelligent structure called Neural-Network-Driven Fuzzy Reasoning showed its potential for modeling datasets with clear clusters (Mirahadi and Zayed 2016). Hybrid feature selection (HFS), which combined filter and wrapper methods with principal component analysis (PCA), was used to identify relevant factors for developing labor productivity models (Ebrahimi et al., 2022). In addition, the Decision-Making Trial and Evaluation Laboratory (DEMATEL) method, the Analytic Network Process (ANP) method, and the Technique for Order of Preference by Similarity to the Ideal Solution (TOPSIS) method were proposed for comparing the productivity of different construction methods in a single model (Shahpari et al. 2019). The Model Tree was a hybrid of decision tree and MLR and applied in estimating labor cost, providing promising explainable artificial intelligence (XAI) solutions (Naumets and Lu 2021).

Despite considerable progress in developing productivity prediction models, the focus of research had been placed on increasing prediction accuracy, largely overlooking the precision of the model due to the dynamic way by which data is selected from the available database. There had been little attention paid to the variance associated with a point-value prediction made by a model. On the other hand, model validations were mainly based on evaluating error terms between the predicted output and the target output, without addressing the variance of the predicted output. Indeed, validating whether the productivity model is sufficient to serve the application purpose has yet to consider the variance of the predicted values. This research tackles the identified fundamental limitation inherent in productivity prediction models by enhancing the model tree application.

3.2.2. Model Tree

The concept behind the model tree is illustrated along with linear regressions in Fig. 3.2. The left figure shows the regression line fitted to a disparate data set with low accuracy. However, accuracy of the linear regression analysis is improved by dividing the dataset into three individual sets constrained by branching rules established by the model tree.

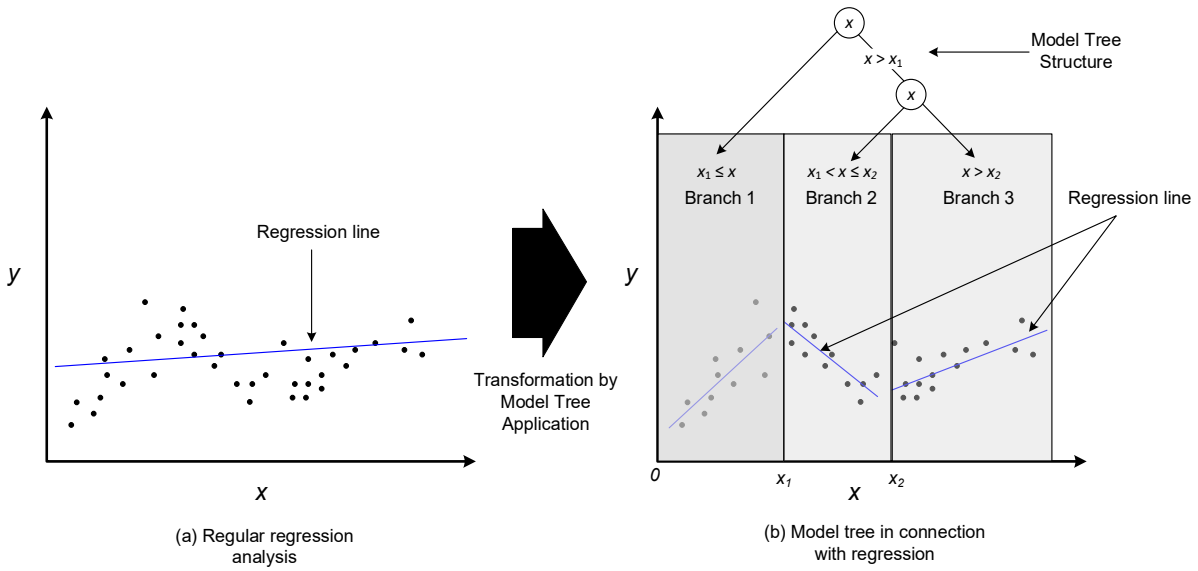


Figure 3.2: Illustration of improvement of regression analysis with model tree application.

Model Tree algorithms essentially address the classification problem by following the "divide and conquer" strategy (Solomatime and Xue 2004). The M5P algorithm for calibrating the model tree was originally introduced by Quinlan (1992), and later improved by Wang and Witten (1997), consisting of four steps: (a) branching of data to grow a complete tree; (b) development of a regression model at each node for pruning and prediction, (c) pruning the tree to avoid the over fitting problem; and (d) smoothing the tree to compensate for the sharp discontinuities caused by the splitting.

By the M5P model tree algorithm, elements (or input features) are selected to make branches based on their values, thus splitting the data into subsets that have similar target variable behaviors. The process involves finding the most informative feature and its corresponding split value that best separates the data into distinct groups. The selection of elements and their split values is determined using a recursive approach. Initially, the entire dataset is considered as one

node or branch. The algorithm searches through all input features to find the feature and its corresponding split value that result in the best split of the data into two subsets based on their target variable behavior. The "best split" is defined by a criterion such as the reduction in variance or the improvement in the model's predictive accuracy. The standard deviation of the class instances having reached a particular node is treated as a measure of the error at a particular node in the model tree. The splitting criterion is to maximize the expected reduction in this error by checking each possible value on each attribute. The attribute along with the associated splitting value that maximizes the expected error reduction at a node is set as the splitting criterion to partition the dataset (Wang and Witten 1996). The standard deviation reduction (SDR) is given by Eq (3.1).

$$SDR = STD(T) - \sum_i \frac{T_i}{|T|} STD(T_i) \qquad (3.1)$$

$$STD = \sqrt{\frac{\sum_j (y_j - \bar{y})}{n}} \qquad (3.2)$$

Where T is the is the set of data points that reach the node and T_i is the data point that result from splitting at the node and falls into one sub-space according to the chosen splitting parameter and STD is the standard deviation. That means, the resulting partition based on variable x_i at the value of a consists of two sets of observations: (1) observations where x_i is less than or equal to a, (2) observations where x_i is greater than a. $STD(T_i)$ is the standard deviation associated with the current subset. M5P determines the SDR value for input variables and then splits the dataset based on the specific variable at a particular value that would maximize the expected error reduction. Herein, STD is calculated for the output values by Eq. 3.2, based on the

observations subject to the current branching condition. Splitting would terminate if only a minimal number of instances remain in the branch or expected error reduction is insignificant. In the end, an MLR model is fitted for each end leaf node in the model tree, using the instances of data that has reached that specific node.

The Model Tree based approach was applied to various civil engineering problems such as predicting labor cost, workability of concrete (Mohsenijam et al. 2021), forecasting river flow (Taghi Sattari et al. 2013), sediment transport in pipes (Najafzadeh et al. 2017), predicting compressive strength of high-performance concrete mix (Deepa et al. 2010). In the construction application domain, decision tree was used in estimating productivity loss due to project change orders (Lee et al. 2004). Desai and Joshi (2010) applied decision trees with constant branch nodes to analyze and predict labor productivity. Despite its simplicity and explainability, the model tree algorithm still lacks the variance estimate of the predicted output for a given set of inputs. Variance estimate of productivity is crucial for reining in the risk of cost overruns during project budget formulation and in establishing a pragmatic project schedule. Currently, the only available approach to obtain such measurements is through experimental design and sensitivity analysis (e.g., using Monte Carlo simulation), which involves altering the input variables and observing the resulting changes in the output. The analytical method characterizing the variability of the input variables to connect those with the variance of the predicted output using the model tree algorithm is still missing in the literature.

3.3. Productivity Modeling Framework

3.3.1. Productivity model for prefabrication

For prefabricated components of certain type, productivity is specifically defined as the sum of productivity elements, each being linked with a particular feature attribute, where labor effort (labor hours) required to make the feature is determined against a standard base product. Consider the following example: the productivity of the formwork prepared for precast a wall panel means the labor hour (LH) required per unit area of wall panel (l_a), which can be derived by considering the five feature attributes: number of strands in the wall panel (l_{Str}), number of lifter arrangement (l_{lif}), height of the wall panel (l_h), opening size of the wall panel (l_{op}), and number of edges (l_{ed}). The features of the wall panel formwork are illustrated in Fig. 3.3.

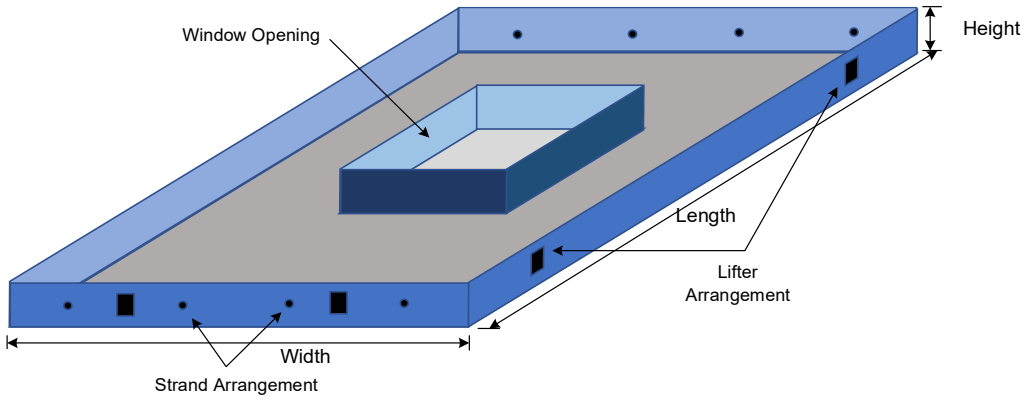


Figure 3.3: Typical precast wall panel formwork.

The productivity of formwork preparation, P is established by combining respective productivity components associated with each feature attribute, namely, number of strands (P_{Str}), number of lifter (P_{lif}), height (P_h), opening size (P_{op}), and edges (P_{ed}), as written as (Eq. 3.3),

$$P = P_{Str} + P_{lif} + P_h + P_{op} + P_{ed} \qquad (3.3)$$

For the base product, which has an area of $l_{a,0}$, number of strands $l_{Str,0}$, number of lifter arrangement $l_{lif,0}$, height of the wall panel $l_{h,0}$, opening size $l_{op,0}$, and number of edges $l_{ed,0}$ and the productivity component due to each feature is $P_{Str}, P_{lif}, P_h, P_{op},$ and P_{ed} respectively, then the base product's productivity is given as per Eq. 3.4,

$$P_0 = P_{Str,0} + P_{lif,0} + P_{h,0} + P_{op,0} + P_{ed,0} \qquad (3.4)$$

Now, given a new formwork with different dimensions, to decide each productivity component we can use linear interpolation for each feature against the standard product. Say, if the productivity component for number of strand for the standard wall panel is $P_{Str,0}$, and standard number of strand is $l_{Str,0}$; therefore, for a product dimension l_{Str}, the productivity component is $P_{Str} = \left(\frac{P_{Str,0}}{l_{Str,0}}\right) l_{Str}$. Following this approach, productivity of formwork preparation P is defined in Eq. 3.5:

$$P = \left(\frac{P_{Str,0}}{l_{Str,0}}\right) l_{Str} + \left(\frac{P_{lif,0}}{l_{lif,0}}\right) l_{lif} + \left(\frac{P_{h,0}}{l_{h,0}}\right) l_h + \left(\frac{P_{op,0}}{l_{op,0}}\right) l_{op} + \left(\frac{P_{ed,0}}{l_{ed,0}}\right) l_{ed} \qquad (3.5)$$

Following the same formwork preparation example, a 10 cm height is the height of the base product (wall panel) and its associated productivity component is 1.5 LH/sq. m. ($P_{h,0} = 1.5$), standard strand number is 4 and it contributes 1.9 LH/sq. m. ($P_{Str,0} = 1.9$), standard lifter number is 4 and it contributes 0.1 LH/ sq. m. ($P_{lif} = 0.1$), standard opening size is 0.25 sq. m. and it contributes 0.2 LH/sq. m. ($P_{op,0} = 0.1$), and standard number of edge is 4 and it contributes 0.05 LH/sq. m. ($P_{ed,0} = 0.05$). Now, for any new wall panel with dimensions, $l_{Str} = 6$, $l_{lif} = 4$, $l_h = 8\ cm$, $l_{op} = 0.44$ sq. m., and $l_{ed} = 4$, the productivity estimate P would be, 4.6 LH per sq. m. of the wall panel area, as elaborated in Eq 3.6.

$$P = \left(\frac{1.9}{4}\right)6 + \left(\frac{0.1}{4}\right)4 + \left(\frac{1.5}{10}\right)8 + \left(\frac{0.2}{0.25}\right)0.44 + \left(\frac{0.1}{4}\right)4 = 4.6\frac{LH}{sq.m.} \qquad (3.6)$$

The productivity given in the above precast example can now be generalized. If there are n component attributes for defining the productivity, the productivity P for the work package can be defined as sum of the productivity components P_n for n number of attributes. This productivity component P_n can be derived based on the standard (base) product. Given the number of feature attributes is n and productivity component is P_n^0, then for l_n dimension the productivity component can be determined as $P_n^0 \times \frac{l_n}{l_{n.0}}$. Therefore, the general form of the productivity model is given in Eq. 3.7.

$$P = \sum_n P_n = \sum_n \left(P_n^0 \times \frac{l_n}{l_n^0}\right) = P_1^0 \times \left(\frac{l_1}{l_1^0}\right) + P_n^0 \times \left(\frac{l_2}{l_2^0}\right) + \cdots + P_n^0 \times \left(\frac{l_n}{l_n^0}\right) \qquad (3.7)$$

Here, Eq. 3.5 resembles the basis form of an MLR equation, which is further given in a generic form in Eq. 3.8. In Eq. 3.8 the base product's productivity for attribute n is P_n^0 , which should be the coefficient β of MLR equation and product feature ratio of $\frac{l_n}{l_{n.0}}$ denotes be input variable x in the MLR equation, with output variable Y denoting productivity P.

$$Y = \beta_0 + \beta_1 \times x_1 + \beta_2 \times x_2 + \cdots + \beta_n \times x_n \qquad (3.8)$$

The productivity definition in the prefabrication context is essentially the MLR model. Next, the variance analysis on MLR is reviewed. Here, β_0 is the intercept of the MLR equation (Eq. 7), denoting the bias present in the productivity model (Eq. 3.7). Next, the variance analysis on MLR is reviewed.

3.3.2. Variance Analysis

Hasan and Lu (2022) proposed the variance analysis technique for MLR model predictions integrating the error propagation theory. The technique is to apply first-order derivatives to approximate the propagation of errors in the input factors of the MLR model. In this research, the Model Tree is enhanced by integrating the variance analysis on MLR equations at branches of the tree structure, resulting in quantifying the variance of the output predicted by the model tree for estimating prefabrication productivity.

The first step of MLR variance analysis is to determine the mean and variance of each coefficient of the MLR equation. To this end, a random subset of the data is sampled to derive the MLR productivity model; this process is repeated for N times to generate N number of MLR equations, based on which the average and standard deviation of the coefficients are determined. Here, in Eq. 3.6, β_n is the regression coefficient associated with input variable x_n. β_n is the intercept of the MLR equation representing the model bias. Now, if the MLR equation is established from one random data sample taken, constrained by the Model Tree's branching rules, β_n will be different each time given N number of sampling experiments. Therefore, there will be N number of MLR equations; form which, mean and standard deviation (μ, σ) of each β parameter can be calculated. Therefore, the final form of Eq. 3.8 is given in (Eq. 3.9):

$$Y(\mu_Y, \sigma_Y) = \beta_0(\mu_0, \sigma_0) + \beta_1(\mu_1, \sigma_1)x_1 + \cdots + \beta_n(\mu_n, \sigma_n)x_n \qquad (3.9)$$

Error propagation theorem explains the law of propagation of random error observing the law of propagation of variances and covariances (Taylor 2022), expressed by Eq. 3.10.

$$C_Y = J_{xY} C_x J_{Yx}^T \qquad (3.10)$$

Here, C_Y is the covariance matrix of random output Y, and C_x is the covariance matrix of random input x, and J_{xY} is the Jacobian (Jacobian matrix) of the Eq. 3.9. Jacobian (J_{xY}) of Y is given as Eq. 3.11,

$$J_{xY} = \left[\frac{\partial Y}{\partial \beta_0} \quad \frac{\partial Y}{\partial \beta_1} \quad \frac{\partial Y}{\partial \beta_2} \quad \dots \quad \frac{\partial Y}{\partial \beta_n}\right] \qquad (3.11)$$

Given all the coefficients are independent, the covariance matrix of the Eq. 3.10 is expanded in Eq. 3.12,

$$C_x = \begin{bmatrix} var(\beta_0) & & & & \\ & var(\beta_1) & & & \\ & & var(\beta_2) & & \\ & & & \ddots & \\ & & & & var(\beta_n) \end{bmatrix} \qquad (3.12)$$

Now, to examine the total error in output Y due to the error propagated from the variance in the model coefficients $\beta_n(\mu_n, \sigma_n)$, Eq. 3.10 is applied along with Eq. 3.9. The final output variance of the MLR model presented by Eq. 3.9 would be (Eq. 3.13),

$$C_Y = {\sigma_Y}^2 = \left[\frac{\partial Y}{\partial \beta_0} \quad \frac{\partial Y}{\partial \beta_1} \quad \frac{\partial Y}{\partial \beta_2} \quad \dots \quad \frac{\partial Y}{\partial \beta_n}\right] \begin{bmatrix} var(\beta_0) & & & & \\ & var(\beta_1) & & & \\ & & var(\beta_2) & & \\ & & & \ddots & \\ & & & & var(\beta_n) \end{bmatrix} \begin{bmatrix} \frac{\partial Y}{\partial \beta_0} \\ \frac{\partial Y}{\partial \beta_1} \\ \frac{\partial Y}{\partial \beta_2} \\ \vdots \\ \frac{\partial Y}{\partial \beta_n} \end{bmatrix}$$

$$= \left(\frac{\partial Y}{\partial \beta_0}\right)^2 var(\beta_0) + \left(\frac{\partial Y}{\partial \beta_1}\right)^2 var(\beta_1) + \cdots + \left(\frac{\partial Y}{\partial \beta_n}\right)^2 var(\beta_n) \qquad (3.13)$$

Here, each $\left\{\left(\frac{\partial Y}{\partial \beta_n}\right)^2 var(\beta_n)\right\}$ term denotes the variance contribution of attribute n to the prediction Y for a given set of input x_n. The Refer to Hasan and Lu (2021) for detailed decomposition processes of Eq. 3.9 and construction of Eq. 3.10.

The performance of each MLR equation in terms of variance in predicted output can then be checked based on a testing dataset. To evaluate the performance of the derived productivity model regarding the variance, we propose to use coefficient of covariance (α) as the index, which is the ratio of the standard deviation (STD) σ_P against the predicted output P_p given in Eq. 3.14. The acceptance of the model can be based on the preset threshold of the coefficient of covariance α_c. If α is less than α_c then the model should be accepted; otherwise rejected.

$$\alpha = \frac{\sigma_P}{P_p} \qquad (3.14)$$

3.3.3. Enhanced Model Tree Application Framework

The proposed productivity modeling framework integrates the variance analysis method for MLR models with the M5P algorithms, giving rise to an enhanced framework for applying the model tree.

First, the entire data set is separated into training and testing sets. The training set, which is used to calibrate the model, while the testing set is chosen at random and reserved for checking the model performance on unseen cases. The base product is defined by its known attribute values. Next, the M5P algorithms are applied to calibrate the model tree, whose performance is checked using the k-fold cross-validation technique by using the training set only. If the performance of the tree model is satisfactory, say square of correlation coefficient R^2 is greater than a limit

(>0.70), the modeling process will continue to the next step; otherwise, it is be terminated. The model tree algorithm partitions the training set into small subsets through applying non-linear classifiers, resulting in a series for MLR models at end leaf nodes.

Upon this point, the variance analysis is performed on each MLR model in the model tree, producing the variance estimate of the predicted productivity for any given set of inputs. Model's accuracy performance is evaluated by contrasting the model output against the target value and calculating correlation coefficient; model's precision performance is checked by deriving the coefficient of covariance α_c for both training and testing sets.

The following paraments related to model calibration are used in this study to execute the proposed framework, as follows:

(1) The number of folds k for k-fold cross validation of the model tree classification model can be set as $k = 5$,

(2) the minimum number of instances for each branch of the Model Tree is set as 30 instances to produce MLR models,

(3) the threshold to check the acceptance of the classification model considers square of correlation coefficient (R^2) greater than 0.70,

(4) the number of runs N to develop the MLR productivity model and reveal mean and variance of coefficients is set as $N = 100$, and

(5) finally, the threshold of model acceptance is set as: on model accuracy, the square of correlation coefficient (R^2) should be greater than 0.80; on model precision, the Coefficient of variation α should be less than 0.30.

Note,

- K-fold cross-validation divides observations into k groups (folds) randomly, using the first as validation while fitting on the remaining k − 1 folds. The choice of k depends on data availability, with no strict applicable rules (Kuhn and Johnson, 2018). However, each train/test group should be statistically representative of the dataset. For this study, $k = 5$ was selected based on data size. A comparable rationale is considered for determining the number of repetitions N.
- The number of instances in each tree branch is set to 30 to conduct significant statistics (Navidi 2011). If more data is available, this threshold can be set to any number higher than 30 and an iterative process can be followed to find a threshold to confirm the best performance for the model.
- On the other hand, a higher value of R^2, approaching 1, signifies a stronger correlation between actual and predicted values. Depending on the specific requirements of the model's application, the acceptance threshold of R^2 for the classification model is recommended as 0.80 for preliminary model validation.

The framework is illustrated in Figure 3.4, with application steps further elaborated as follows.

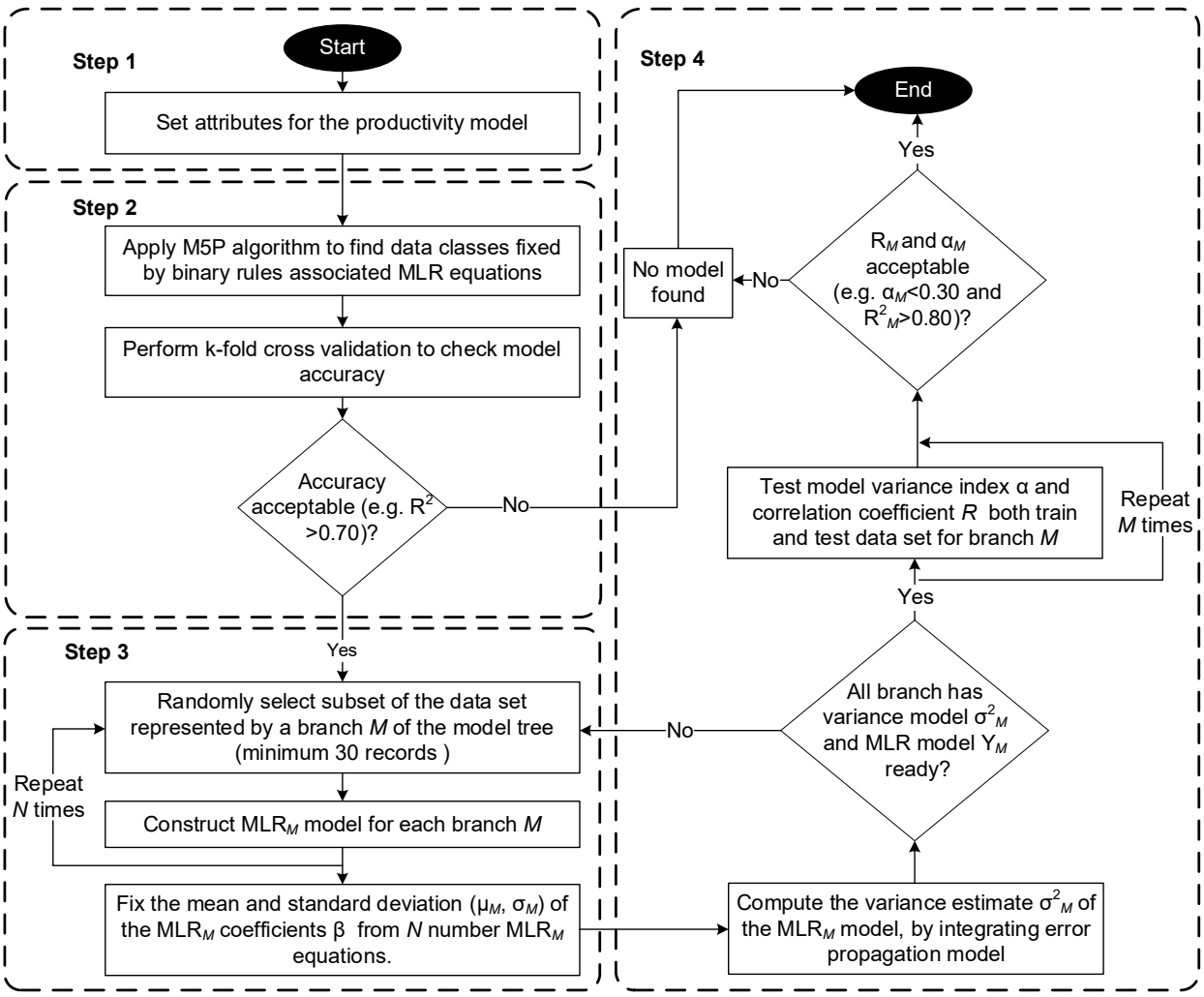


Figure 3.4: Construction productivity model computation framework.

Step 1: Select a set of attributes (n) that defines a prefabricated product along with the values of the base product attributes $l_{n.0}$.

Step 2: Apply the Model Tree algorithm to classify the datasets into small subsets suitable for constructing MLR models. To evaluate the performance of an MLR model, square of correlation coefficient (R^2), and mean absolute percentage error (MAPE) between the output and target

variables can be used. For this study, R^2 is used as the model accuracy performance index of the model tree classifier.

Step 3: If the model accuracy is satisfactory, then in the next step, select a subset of the total data represented by each branch of the model tree; data is randomly selected to prepare a productivity MLR model specific to that branch as per Eq. 3.8, where, coefficient β of MLR equation should be the productivity component P_n^0 for product feature $l_{n.0}$, and $\frac{l_n}{l_{n.0}}$ is represented by input variable x in the MLR equation. Repeating the process N times would result in the productivity model for branch M as per Eq. 3.9 where Y represents the predicted productivity P for input x_n .

Step 4: Estimate the variances as per Eq. 3.13 of the productivity MLR model in connection with each branch of the model tree. If the average Coefficient of variation α_{avg} for both test set ($\alpha_{avg,test}$), and training set ($\alpha_{avg,train}$) of productivity model P_M at branch M falls below the acceptable threshold (e.g. say less than 0.30), then the model can be accepted for application, otherwise should be rejected.

It is worth mentioning that a computer program was coded in Python 3.8 programming language (Python Software Foundation 2019) using Scikit-learn module (Pedregosa et al. 2011). The model tree implementation of the program resorted to the python-m5p package developed by Marie (2022). In the ensuing section, a case of modeling the productivity for prefabrication of structural steel is presented to demonstrate the application of the proposed methodology.

3.4. Case Study Structural Steel Fabrication

The available dataset consists of a total of 208 instances of structural steel fabrication performed by one fabricator over two years. Each record has 37 attributes describing the fabricated division plus one recorded productivity ratio (37 input attributes and 1 output variable). Among 37 input attributes four attributes (X1 to X4) are nominal factors denoting classes or types of the project, summarized in Table 3.1; the rest are all numerical parameters describing various aspects of the engineering design and project details, which are summarized in Table 3.2. Thirty-three instances were reserved from the original data set for testing the overall performance of the calibrated productivity model. The remaining 175 instances were used for model calibration. Note this data set is intended to model steel fabrication productivity at the division level, i.e., a division of structural steel fabrication is a well-defined scope of work with productivity rate recorded in terms of LH per kg.

Table 3.1 Data properties of the structural steel fabrication labor cost dataset for nominal attributes.

Attribute ID	Attribute	Number of distinct labels	Labels of the nominal attributes
X1	Scope of work	2	Supply Only, Supply & Erection
X2	Sector	5	Oil & Gas, Commercial & Institutional, Industrial & Mechanical, Transportation & Infrastructure, Others
X3	Location	6	SSE, SSW, SSV, SSS, SSP, SSB
X4	Module fabrication complexity	4	Medium, Light, Heavy, Very heavy

Table 3.2 Data properties of the structural steel fabrication labor cost dataset for numerical attributes.

Attribute ID	Attribute	Unit	Average	STD	Median	max	Min
X5	Hollow steel weight	kg	37555	106505	1772	1141020	0
X6	Hollow steel quantity	-	162	361	21	3475	0
X7	Hollow steel length	m	4581	13453	144	86505	0
X8	Wide flange weight	kg	112814	196409	22605	1012786	0
X9	Wide flange quantity	-	254	432	85	2494	0
X10	Wide flange length	m	7156	14413	1223	79886	0
X11	C-shape weight	kg	4830	9250	713	52088	0
X12	C-shape quantity	-	88	175	17	1397	0
X13	C-shape length	m	1218	3637	109	24715	0
X14	L-shape weight	kg	7955	12982	2743	89325	0

Attribute ID	Attribute	Unit	Average	STD	Median	max	Min
X15	L-shape quantity	-	394	687	60	4263	0
X16	L-shape length	m	2092	6012	286	57294	0
X17	Plate weight	kg	43115	107597	7521	887878	0
X18	Plate quantity	-	1078	1742	406	14586	0
X19	Plate length	m	947	1800	252	17325	0
X20	Round bar weight	kg	259	1867	0	25720	0
X21	Round bar quantity	-	189	917	0	9375	0
X22	Round bar length	m	55	505	0	7286	0
X23	Miscellaneous weight	kg	406	4412	0	63903	0
X24	Miscellaneous quantity	-	2	21	0	308	0

Attribute ID	Attribute	Unit	Average	STD	Median	max	Min
X25	Miscellaneous length	m	6	46	0	610	0
X26	S-shape weight	kg	224	986	0	11819	0
X27	S-shape quantity	-	141	835	0	10122	0
X28	S-shape length	m	20	80	0	649	0
X29	Wide T-shape weight	kg	1700	7331	0	58783	0
X30	Wide T-shape quantity	-	25	91	0	734	0
X31	Wide T-shape length	m	232	871	0	6510	0
X32	Pipe weight	kg	1199	7672	0	107490	0
X33	Pipe quantity	-	46	238	0	3108	0
X34	Pipe length	m	309	1659	0	17998	0

Attribute ID	Attribute	Unit	Average	STD	Median	max	Min
X35	Total weight of the module	kg	210055	285746	72801	1253866	1583
X36	Total quantity of module	-	2379	3132	1285	19855	12
X37	Total length of module	m	16616	33173	3903	197852	52
Y	*Total labor hours (output)*	LH	4006	5394	1597	31788	32

In this case, we set the number of folds $k = 5$ for k-fold cross validation of the model tree classification model, minimum number of instances = 30 for each branch of the Model Tree, square of correlation coefficient $R_M^2 = 0.70$, number of runs $N = 100$ to develop the MLR productivity model, and the threshold to the indices for accepting model $\alpha_c = 0.30$.

3.4.1. Step 1: Attribute Selection for Productivity Model

The majority of factors are numeric (real numbers, e.g., quantity takeoff, or percentage ratio). Nonetheless, in modeling labor productivity, some nominal factors are commonly encountered and need to be quantitatively represented to facilitate attribute selection analysis and enable regression model formulation. In the presented case study, the dummy encoding technique (Alkharusi 2012) is applied to represent a categorical variable to a set of binary variables (also referred to as "dummy variables"). Each category becomes a binary indicator, assuming 1 for presence and 0 for absence. Despite the efforts in ensuring independence in selecting inputs, this research thoughtfully addresses the possibilities of multiple elements having interactions with one another and forming a nonlinear model in making productivity predictions. A correlation-based greedy-stepwise search strategy was employed to select the most relevant features for modeling the problem (Caruana and Freitag 1994; Ranjan et al. 2021). Note the process of selecting suitable independent elements as model inputs is described in Appendix C based on the previous work published in Hasan and Lu (2022).

Selected attributes were 'total weight of structural pieces' (X35), 'plate quantity' (X18), wide flange weight (X8), 'plate length' (X19), 'wide T-shape (t beam) weight' (X29), 'L-shape (angle) length' (X16), and 'module fabrication complexity' (X4). Among the selected variables, only X4 is a nominal type of variable, and the rest are numeric. Herein, the weight of structural pieces in a fabrication division (X35) is directly related to the amount of work to be done. Plate quantity and length (X18) have an indirect impact on productivity as they provide insight into the number of beam-column joints and the complexity of shop plate processing. The weight and proportion of wide flange (X8) and T beams (X29) also affect fabrication productivity. Furthermore, the length

of angles (X16) can provide an idea of how the bracing arrangement impacts productivity. Lastly, the complexity of the structural system (X4) should be considered when selecting a fabrication style. In this case, all the variables selected align closely with common knowledge and practical knowhow in the domain of structural steel fabrication. .

With the attributes identified the productivity model for structural steel proposed is given in Eq. 3.15, (as per Eq. 3.7).

$$P = P^0_{X4-M} \times \left(\frac{l_{X4-M}}{l^0_{X4-M}}\right) + P^0_{X4-L} \times \left(\frac{l_{X4-L}}{l^0_{X4-L}}\right) + P^0_{X4-H} \times \left(\frac{l_{X4-H}}{l^0_{X4-H}}\right) + P^0_{X4-HV} \times \left(\frac{l_{X4-HV}}{l^0_{X4-HV}}\right) + P^0_{X16} \times \left(\frac{l_{X16}}{l^0_{X16}}\right) + P^0_{X29} \times \left(\frac{l_{X29}}{l^0_{X29}}\right) + P^0_{X19} \times \left(\frac{l_{X19}}{l^0_{X19}}\right) + P^0_{X8} \times \left(\frac{l_{X8}}{l^0_{X8}}\right) + P^0_{X18} \times \left(\frac{l_{X18}}{l^0_{X18}}\right) \quad (3.15)$$

3.4.2. Step 2: Model Tree Integration

With the selected attributes, the Model Tree algorithm of M5P was invoked to identify classes and subsets (branches), using a total of 175 samples, with the soothing and pruning functions enabled. The minimum number of instances option was set as 30. The model's performance was tested using 5-fold cross-validation techniques in terms of prediction accuracy. The resulting R^2 was 0.82, greater than the preset acceptance threshold of 0.70. The subsequent step was to generate the model tree with a total of 4 branches, shown in Fig. 3.5. Hence, each branch is linked with a productivity prediction MLR equation for structural steel fabrication. A sample classification performance at some control points of the Enhanced Model Tree algorithm illustrated with attribute 35 is provided in Appendix E.

The total productivity model consists of four MLR equations constrained by the boundary conditions set by each model tree branch. Note, by learning from the provided training data, the M5P algorithm had chosen the total weight of the steel division (unit of measure KG) as the main criterion in branching.

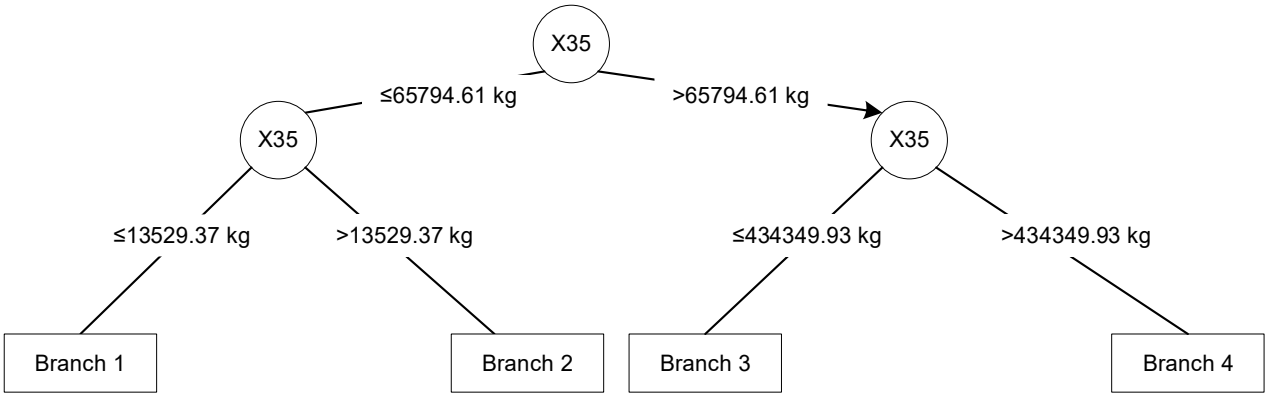


Figure 3.5: M5P model tree for labor productivity prediction data classification.

3.4.3. Step 3: Formulating Labor Productivity Prediction Model

In Eq. 3.15, productivity is defined as the labor hour required to complete per unit weight of steel fabrication (LH/kg), which is calculated by dividing Y over X35 for each of the instances. The base product attribute values (l_n^0) include l_{X16}^0, l_{X29}^0, l_{X19}^0, l_{X8}^0, and l_{X18}^0, which are 2092 m, 1700 kg, 947 m, 112814 kg, and 1078 respectively. In addition, for the nominal attribute (X4), the product attribute and base product attribute ratio$\left(\frac{l_{X4-M}}{l_{X4-M}^0}\right)$ equals either 1 or 0 (1 means the feature is present; 0 = feature is absent).

To construct the productivity model 80% of the records belonging to each branch (randomly selected) were taken each time to perform the MLR analysis; the process was repeated 100 times, resulting in the mean and standard deviation of each MLR coefficient. It is noted in the current case study, the intercept of the MLR model was set to zero (as p value was found insignificant). Once all the four branches of the model tree were calibrated, four MLR equations were derived for productivity prediction, each being associated with one branch of the model tree. The results are given in Table 3.3. Note the mean values of the model coefficients P_n^0 define the MLR equations to predict productivity (P) for a given set of input parameters (l_{X4-M}).

Table 3.3 Productivity model for 4 classifications represented by each of model tree branches.

Branch	Branch Logic (boundary conditions)	Productivity Model
1	Total weight of pieces (X35) ≤ 13529.37 kg	$P = (0.02610, 0.00280) \times \left(\frac{l_{X4-M}}{l^0_{X4-M}}\right) + (0.00517, 0.00549) \times \left(\frac{l_{X4-L}}{l^0_{X4-L}}\right) + (0.04727, 0.00385) \times \left(\frac{l_{X4-H}}{l^0_{X4-H}}\right) + (0.10248, 0.00729) \times \left(\frac{l_{X4-HV}}{l^0_{X4-HV}}\right) + (0.04608, 0.03092) \times \left(\frac{l_{X16}}{2092}\right) + (0.00311, 0.00673) \times \left(\frac{l_{X29}}{1700}\right) + (0.01005, 0.00556) \times \left(\frac{l_{X19}}{947}\right) + (-0.10416, 0.06668) \times \left(\frac{l_{X8}}{112814}\right) + (0.03409, 0.00704) \times \left(\frac{l_{X18}}{1078}\right)$

Branch	Branch Logic (boundary conditions)	Productivity Model
2	13529.37 kg < Total weight of pieces (X35) ≤ 65794.61 kg	$P = (0.03091, 0.00888) \times \left(\frac{l_{X4-M}}{l^0_{X4-M}}\right) + (0.01599, 0.00921) \times \left(\frac{l_{X4-L}}{l^0_{X4-L}}\right) + (0.09453, 0.01943) \times \left(\frac{l_{X4-H}}{l^0_{X4-H}}\right) + (0.20171, 0.01413) \times \left(\frac{l_{X4-HV}}{l^0_{X4-HV}}\right) + (-0.00200, 0.01033) \times \left(\frac{l_{X16}}{2092}\right) + (-0.01155, 0.00454) \times \left(\frac{l_{X29}}{1700}\right) + (-0.00505, 0.00428) \times \left(\frac{l_{X19}}{947}\right) + (-0.00960, 0.03072) \times \left(\frac{l_{X8}}{112814}\right) + (-0.00903, 0.01284) \times \left(\frac{l_{X18}}{1078}\right)$

Branch	Branch Logic (boundary conditions)	Productivity Model
3	65794.61 kg < Total weight of pieces (X35) ≤ 434349.93 kg	$P = (0.02358, 0.00215) \times \left(\frac{l_{X4-M}}{l^0_{X4-M}}\right) + (0.01050, 0.00327) \times \left(\frac{l_{X4-L}}{l^0_{X4-L}}\right) + (0.05596, 0.00454) \times \left(\frac{l_{X4-H}}{l^0_{X4-H}}\right) + (0.26468, 0.00634) \times \left(\frac{l_{X4-HV}}{l^0_{X4-HV}}\right) + (-0.00100, 0.00114) \times \left(\frac{l_{X16}}{2092}\right) + (-0.00010, 0.00064) \times \left(\frac{l_{X29}}{1700}\right) + (-0.00019, 0.00071) \times \left(\frac{l_{X19}}{947}\right) + (-0.00104, 0.00177) \times \left(\frac{l_{X8}}{112814}\right) + (0.00175, 0.00088) \times \left(\frac{l_{X18}}{1078}\right)$

Branch	Branch Logic (boundary conditions)	Productivity Model
4	Total weight of pieces (X35) > 434349.93 kg	$P = (0.01706, 0.00117) \times \left(\frac{l_{X4-M}}{l^0_{X4-M}}\right) + (0.00919, 0.00117) \times \left(\frac{l_{X4-L}}{l^0_{X4-L}}\right) + (0.05368, 0.00502) \times \left(\frac{l_{X4-H}}{l^0_{X4-H}}\right) + (0, 0) \times \left(\frac{l_{X4-HV}}{l^0_{X4-HV}}\right) + (0.00010, 0.00007) \times \left(\frac{l_{X16}}{2092}\right) + (0.00017, 0.00006) \times \left(\frac{l_{X29}}{1700}\right) + (0.00074, 0.00044) \times \left(\frac{l_{X19}}{947}\right) + (-0.00007, 0.00021) \times \left(\frac{l_{X8}}{112814}\right) + (0.00030, 0.00027) \times \left(\frac{l_{X18}}{1078}\right)$

Given a new fabrication job, for example, a division's weight 13500 kg with complexity level heavy, 'plate quantity' (X18) = 1138, Wide flange weight (X8) = 10528 kg, 'plate length' (X19) =1075 m, 'wide T-shape (t beam) weight' (X29) = 2124 kg, 'L-shape (angle) length' (X16) = 895 m, this

instance falls under the constraints of Branch 1 of the productivity model. MLR productivity model generated at Branch 1 is as Eq. 3.16:

$$\begin{aligned} P = \quad & (0.02610, 0.00280) \times \left(\frac{l_{X4-M}}{l^0_{X4-M}}\right) + (0.00517, 0.00549) \times \left(\frac{l_{X4-L}}{l^0_{X4-L}}\right) \\ & + (0.04727, 0.00385) \times \left(\frac{l_{X4-H}}{l^0_{X4-H}}\right) + (0.10248, 0.00729) \times \left(\frac{l_{X4-HV}}{l^0_{X4-HV}}\right) \\ & + (0.04608, 0.03092) \times \left(\frac{l_{X16}}{2092}\right) + (0.00311, 0.00673) \times \left(\frac{l_{X29}}{1700}\right) \\ & + \ (0.01005, 0.00556) \times \left(\frac{l_{X19}}{947}\right) + (-0.10416, 0.06668) \times \left(\frac{l_{X8}}{112814}\right) \\ & + (0.03409, 0.00704) \times \left(\frac{l_{X18}}{1078}\right) \qquad (3.16) \end{aligned}$$

Now plugging the mean of $P^0_n(\mu_{i,n})$ we get the productivity as 0.109 LH/kg (Eq. 3.17) in fabrication of this job.

$$\begin{aligned} P = \quad & 0.02610 \times 0 + 0.00517 \times 0 + 0.04727 \times 1 + \ 0.10248 \times 0 + 0.04608 \times \left(\frac{895}{2092}\right) \\ & + \ 0.00311 \times \left(\frac{2124}{1700}\right) + 0.01005 \times \left(\frac{1075}{947}\right) - 0.10416 \times \left(\frac{10528}{112814}\right) \\ & + 0.03409 \times \left(\frac{1138}{1078}\right) = \ 0.109 \frac{LH}{kg} \quad (3.17) \end{aligned}$$

3.4.4. Step 4: Variance Analysis of the Productivity Model

The variance analysis on each branch of the model tree is performed using Eq. 3.13. For the steel fabrication productivity model, variance of the output can be found from Eq. 3.18.

$$C_P = {\sigma_P}^2 = \left(\frac{\partial P}{\partial P^0_{X4-M}}\right)^2 var(P^0_{X4-M}) + \left(\frac{\partial P}{\partial P^0_{X4-L}}\right)^2 var(P^0_{X4-L}) + \left(\frac{\partial P}{\partial P^0_{X4-H}}\right)^2 var(P^0_{X4-H}) + \left(\frac{\partial P}{\partial P^0_{X4-VH}}\right)^2 var(P^0_{X4-VH}) + \left(\frac{\partial P}{\partial P^0_{X16}}\right)^2 var(P^0_{X16}) + \left(\frac{\partial P}{\partial P^0_{X29}}\right)^2 var(P^0_{X29}) + \left(\frac{\partial P}{\partial P^0_{X19}}\right)^2 var(P^0_{X19}) + \left(\frac{\partial P}{\partial P^0_{X8}}\right)^2 var(P^0_{X8}) + \left(\frac{\partial P}{\partial P^0_{X18}}\right)^2 var(P^0_{X18}) \quad (3.18)$$

For the example given in the previous section, the variance estimate on the predicted labor productivity 0.042 LH/kg would be 0.00039, therefore the standard deviation (STD) is 0.020 LH/kg, as elaborated in Eq. 3.19 and 3.20.

$$C_P = {\sigma_P}^2 = (0)^2(0.00280)^2 + (0)^2(0.00549)^2 + (1)^2(0.00385)^2 + (0)^2(0.00729)^2 + \left(\frac{895}{2092}\right)^2 (0.03092)^2 + \left(\frac{2124}{1700}\right)^2 (0.00673)^2 + \left(\frac{1075}{947}\right)^2 (0.00556)^2 + \left(\frac{10528}{112814}\right)^2 (0.06668)^2 + \left(\frac{1138}{1078}\right)^2 (0.00704)^2 = 0.00039 \quad (3.19)$$

or,

$$\sigma_P = \sqrt{0.00039} = 0.020 \frac{LH}{kg} \quad (3.20)$$

Now, as per Eq. 3.14, the Coefficient of variation for the predicted output is 0.182 (α_P) (Eq. 3.21).

$$\alpha_P = \frac{\sigma_P}{P_p} = \frac{0.020}{0.109} = 0.182 \quad (3.21)$$

3.4.5. Performance Evaluation: Calibrated Productivity Model

We evaluated the performance of the obtained productivity model with both the training and test datasets. The correlation between actual and predicted productivity is depicted in Fig. 3.6. Root

mean square error (RMSE), mean absolute percentage error (MAPE), and the average of Coefficient of variation (α_P) of the predicted outputs are summarized in Table 3.4. Note the performance indices on training and testing sets are closely aligned, with model performance on training set outperforming the testing set by small margins. The square of correlation coefficient (R^2) for both training and testing datasets are greater than the acceptable threshold of 0.80 and the average Coefficient of variation α_P for both training and testing datasets are below the acceptable threshold of 0.30. Therefore, the productivity model as produced was acceptable. It is noteworthy that the maximum Coefficient of variation possible was 1.08 for the training set and 0.84 for the testing set, respectively. This pointed to certain scenarios in which the modeler should pay special attention to the higher variability in the productivity.

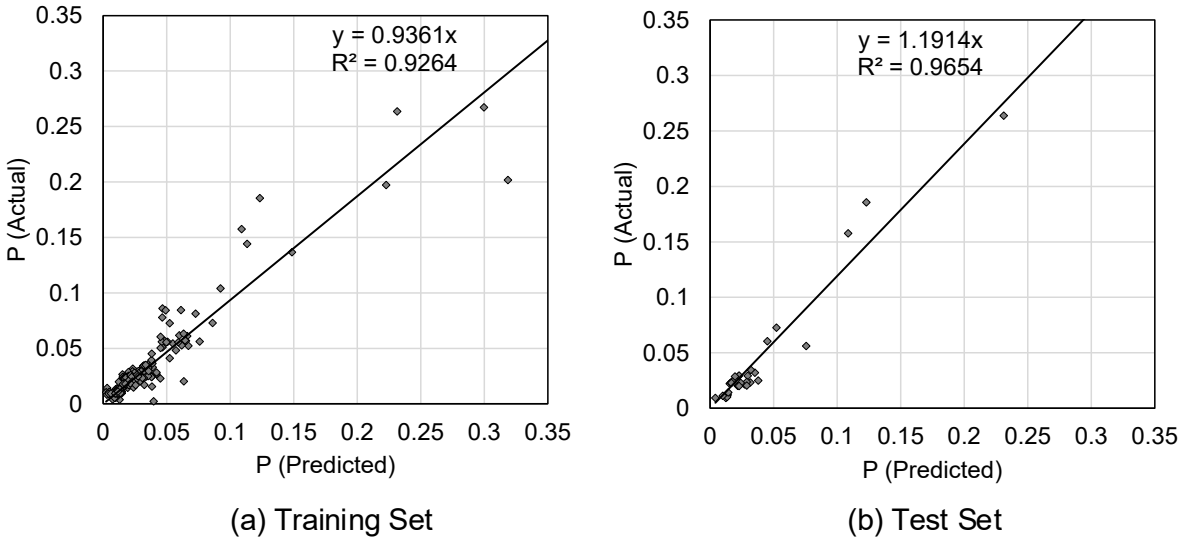


Figure 3.6: Correlation between actual and predicted labor productivity after model tree application for (a) training, (b) testing dataset.

Table 3.4 Performance indicator of the labor productivity model.

Performance indicator	Training Set	Testing Set
Square of correlation coefficient, R^2	0.93	0.97
Root mean square error (RMSE),	0.0139	0.0167
Mean absolute percentage error (MAPE)	27.8	27.3
Average – Coefficient of variation $(\alpha_{P,avg})$	0.29	0.25
Maximum – Coefficient of variation $(\alpha_{P,max})$	1.08	0.84

Apart from analyzing the performance of the overall model, the productivity models associated with each branch were separately evaluated, as shown in Fig. 3.7, revealing significantly better accuracy performance as of Branch 1, 3, and 4; the square of correlation coefficient (R^2) between actual and predicted productivity values surpassed 0.90.

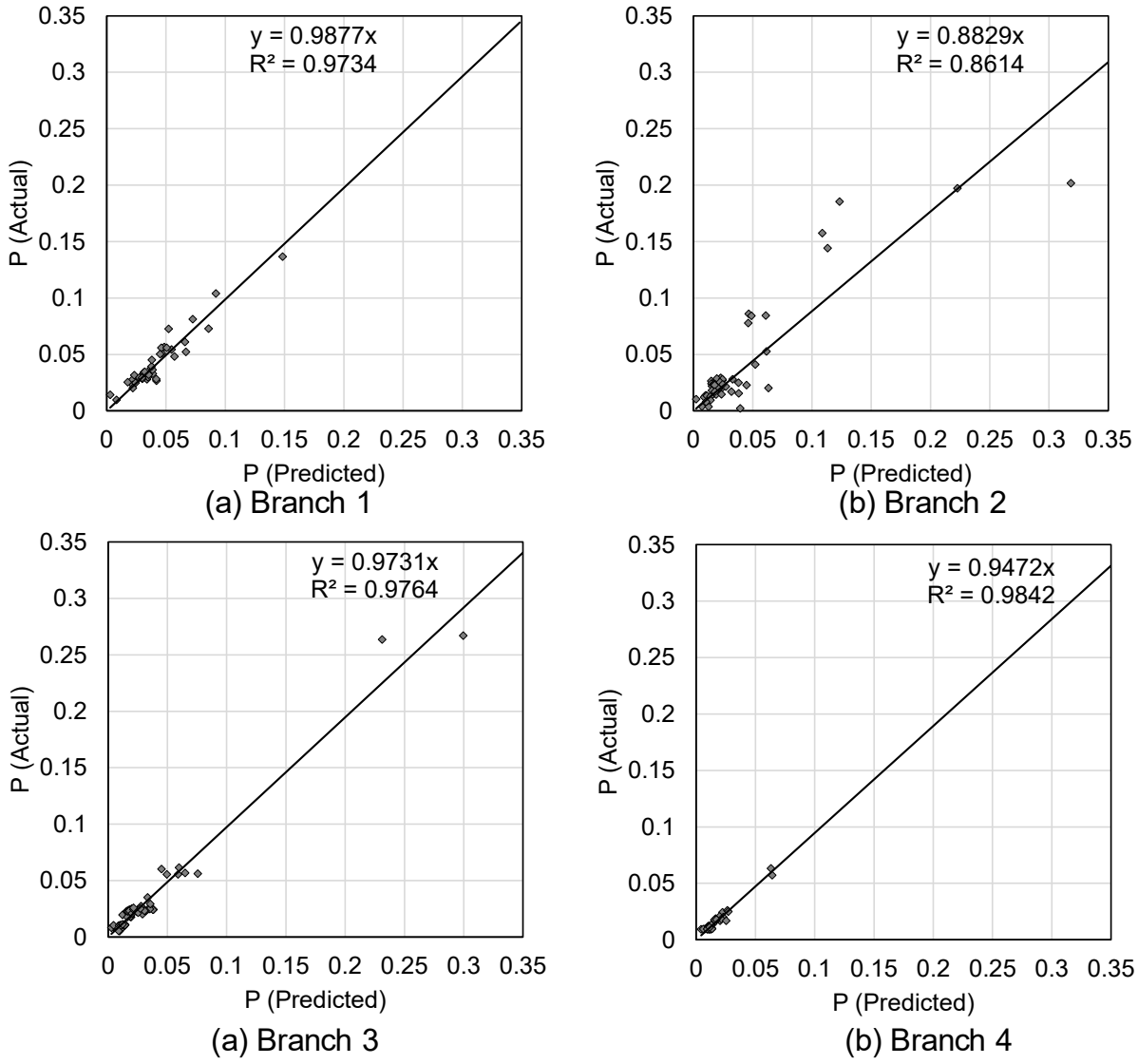


Figure 3.7: Prediction performance of all four productivity models.

However, Branch 2 had a slightly worse result but still fell below the acceptance threshold. As such, if the input attribute set falls under the constraints of Branches 1, 3, and 4, we can expect more reliable results in terms of lower output variance. While, if the input sets fall under the constraints of Branch 2, we anticipate less reliable predictions. Performance evaluation results for individual branch's productivity models are also summarized in Table 3.5.

Table 3.5 Performance indicators of all four productivity models.

Performance indicator	Branch 1	Branch 2	Branch 3	Branch 4
R^2	0.99	0.86	0.98	0.98
RMSE	0.008	0.023	0.008	0.003
MAPE	25.8	41.1	24.5	16.1
Average – Coefficient of variation $(\alpha_{P,avg})$	0.16	0.54	0.25	0.19
Maximum – Coefficient of variation $(\alpha_{P,max})$	0.60	1.08	0.79	0.66

In addition to the model performance in terms of prediction accuracy, the performance of prediction precision was also evaluated in terms of the absolute error of the labor productivity prediction (absolute value of (actual P – predicted P)) against the predicted standard deviation of the prediction for each instance (Eq. 3.22). Results are plotted in Figure 3.8, showing 94.3% of instances in the training set, the precision ratio falls below the cut value of 1.96; for the test dataset, the result is 90%. Note *1.96* is selected as it is the value to multiply the standard deviation in fixing the upper bound of the prediction at 95% confidence level in statistics.

$$Relative\ error, e\ =\ \frac{|Output - target|}{STD\ of\ the\ Output} = \frac{|Actual\ P - Predicted\ P|}{STD\ of\ the\ predicted\ P} \quad (3.22)$$

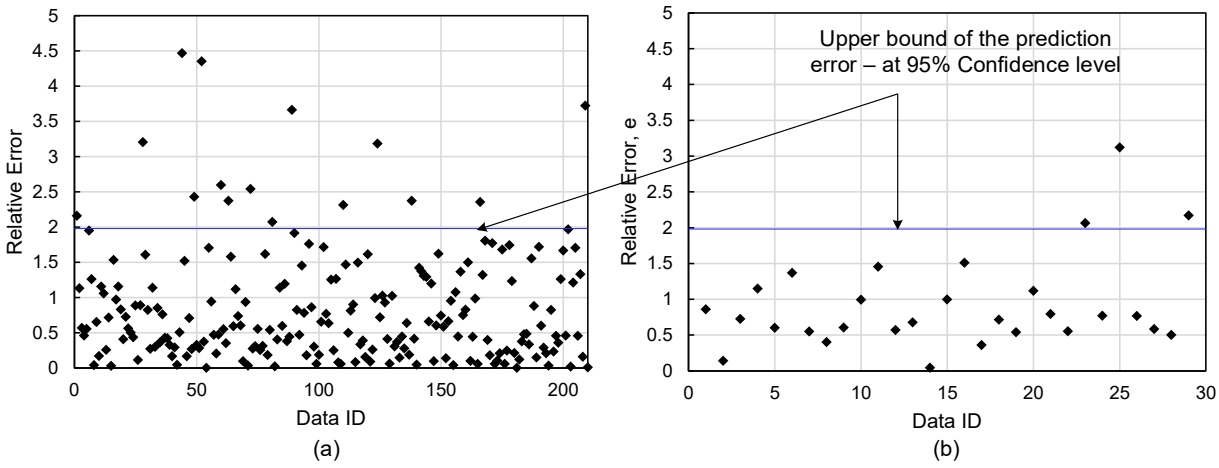


Figure 3.8: Performance of the variance prediction of the productivity model (a) training set, (b) test set.

3.4.6. Comparing Enhanced Model Tree with Established Models

For cross checking the performances of the enhanced Model Tree, the productivity problem had been independently modeled by MLR model and ANN model with the same input definitions and the same dataset. Both ANN and MLR models were created in the widely used, freely available WEKA platform for data mining applications (Frank et al. 2016). The obtained MLR model in WEKA is given in Eq. 23, which is one MLR equation in contrast with four MLR equations on branches of the model tree. The ANN model calibrated in Weka was configured with a learning rate of 0.3, momentum of 0.2 and one hidden layer with 5 nodes in it; the sigmoid function was used as activation functions. The experiment was conducted on a computer with an Intel Core i7-

4770 3.40GHz CPU and 16 GB of RAM. The training time required for the final productivity model derived using MLR, enhanced model tree and ANN algorithms was recorded as 0.004, 0.045, and 0.101 seconds respectively.

$$P = 0.0139 \times \left(\frac{l_{X4-M}}{l^0_{X4-M}}\right) + 0.0474 \times \left(\frac{l_{X4-H}}{l^0_{X4-H}}\right) + 0.173 \times \left(\frac{l_{X4-HV}}{l^0_{X4-HV}}\right) - 0.0002 \times \left(\frac{l_{X29}}{1700}\right) - 0.0011 \times \left(\frac{l_{X19}}{947}\right) - 0.0008 \times \left(\frac{l_{X8}}{112814}\right) + 0.0013 \times \left(\frac{l_{X18}}{1078}\right) + 0.113 \quad (3.23)$$

Results are summarized in Table 3.6 and following observations based on the current study are made:

- The enhanced Model Tree outperformed the standalone MLR in terms of model accuracy due to nonlinear modeling capacity of the Model Tree; the enhanced Model Tree resulted in higher R^2 on the training dataset (0.92) and smaller RMSE on the training dataset (0.0139).
- The enhanced Model Tree and ANN delivered comparable performances on model accuracy in terms of R^2 and RMSE. Nonetheless, the enhanced Model Tree was preferred to ANN because (1) the variance was analytically predicted alongside the point-value output and (2) the productivity model was explainable in terms of the reasoning logic for productivity prediction.

Table 3.6 Performance evaluation for various productivity models independently developed in the case study.

Performance Index	Enhanced Model Tree		MLR Model		ANN Model	
	Training Set	Testing Set	Training Set	Testing Set	Training Set	Testing Set
R^2	0.93	0.97	0.88	0.87	0.92	0.89
RMSE	0.0139	0.0167	0.0191	0.0073	0.0169	0.0077
Average – Coefficient of variation $(\alpha_{P,avg})$	0.29	0.25	-	-	-	-

The performance of a model tree consistently surpasses that of a standalone MLR approach. Model Tree demonstrates a level of accuracy comparable to ANNs in terms of RMSE, contingent upon the specific case, dataset characteristics, and the requisite experimental exploration. Determining whether the enhanced model tree fares better or worse than acounterpart model (e.g., ANN) necessitates a deliberate process of trial and error that aligns with the particular application requirements and data complexities.

The testing set consisted of 33 records randomly selected from the original data at the beginning and was kept separate throughout the analysis. In machine learning, the testing set is mainly reserved to prevent the model from overlearning (over-calibrated to the training set and predict the training set with high accuracy but perform poorly on the unseen test set.) In this case, RSME on the testing set for the enhanced model tree is still acceptable, small enough to indicate no overlearning based on the training set. Note that the RSME can be sensitive to distortion by outliers (Naumets and Lu 2021). The square of correlation coefficient R2 is the commonly used measure of the overall accuracy in machine learning (the closer to 1, the more accurate predictions overall against targets), which is also shown in Table 4. The R2 value for the training set of the enhanced model tree model is 0.97, and it is higher than that of MLR (0.87) and ANN (0.89). The model performance is evaluated considering multiple accuracy measures (R2 and RMSE) on the training set and the testing set, as well as the explainability of model logic.

3.5. Second Case: Pipe Spool Fabrication

A pipe spool refers to a prefabricated segment of a piping system consisting of diverse components like flanges, elbows, reducers, tees, supports, and pipes. These elements are preassembled into discrete units and eventually integrated into an industrial plant or production skid/module. This fabrication is commonly executed in a controlled shop environment for higher productivity, better quality control, and lower labor costs. A data set of seventy-one records of pipe spool fabrication projects were collected from over sixty industrial construction projects performed by an industry partner over a four year's horizon. Given the labor-intensive nature of spool fabrication, labor productivity presents itself as the major factor in making project cost budgets. Table 3.7 summarizes the properties of the pipe spool dataset used for the case study. Note this dataset was

prepared to demonstrate and validate novel algorithms for sensitivity analysis on a backpropagation ANN model (Lu et al 2001). In this case, only the enhanced Model Tree was applied to the dataset following the proposed methodology.

Table 3.7 Data properties of the structural steel fabrication labor cost dataset for numerical attributes.

Attribute ID	Attribute	Remarks	Min	Max
X1	In-line fitting (pcs) per foot of pipe in spool	Ratio indicating the average length of pipe section in spool	0	0.128434
X2	Non-in-line fitting (pcs) per foot of pipe in spool	Ratio indicating complexity of spool configuration	0.022643	0.811826
X3	Valve (pcs) per foot of pipe in spool	Ratio indicating complexity of spool configuration	0	0.094528
X4	Support (pcs) per foot of pipe in spool	Ratio indicating complexity of spool configuration	0	0.045499
X5	Flange (pcs) per foot of pipe in spool	Ratio indicating complexity of spool configuration	0.001326	0.730643

Attribute ID	Attribute	Remarks	Min	Max
X6	Multistation roll weld inches/total roll weld inches	Multistation roll weld requires extra handling between weld stations	0	0.42571
X7	Repair rate	Index of crew's proficiency	0	0.230769
X8	Radiography test requirement	Index of quality control stringency by specifications	0	100
X9	Non-CS units/total units	Non-CS component in fabrication requires extra care in storage, handling, and welding	0	1
X10	Shop workload	5-point rating based on shop workload in units and no. of concurrent jobs indicating how busy the shop was	1	5
X11	Drawing revision rate	5-point rating based on percent of revised spool drawings indicating drawing quality	1	3
X12	Priority rushed spools	A 5-point rating based on percent of rushed spool due to client priority indicating shop work schedules.	1	5

Attribute ID	Attribute	Remarks	Min	Max
X13	Rework spools	A 5-point rating based on percent of reworked spools due to drawing errors and quality defects	1	5
X14	Material shortage problems	A 5-point rating on efficiency of material supply	1	5
X15	Late drawing issues	A 5-point rating based on percent of late spool drawing issuance by client that impacts fabrication	1	5
X16	Nigh shift MHs/total MHs	Night shift affects labor productivity	0	0.340288
X17	Over time MHs/total MHs	Overtime affects labor productivity	0	0.263572
X18	Extra work MHs/total MHs	Extra work affects labor productivity	0	1.195918
X19	Apprenticeship MHs/total MHs	Welder qualification system affects labor productivity: Apprentice versus journeyman	0	0.517419
Y	Labor hour required per unit	Labor hour required per unit	0.132533	0.503078

Similar to the previous case, a correlation-based greedy-stepwise search strategy was employed to select the most pertinent input attributes in the data preparation stage. Out of the nineteen input attributes, eight were selected for pipe spool productivity modeling, specifically: non-in-line fitting (pcs) per foot of pipe in spool (X2), Support (pcs) per foot of pipe in spool (X4), flange (pcs) per foot of pipe in spool (X5), repair rate (X7), non-CS units/total units (X9), shop workload (X12), nigh shift MHs/total MHs (X16), and (over time MHs/total MHs)X17. The enhanced model tree algorithms were executed on the entire data set to calibrate the productivity model by modeling parameters as follows: number of folds for cross validation $k = 3$, minimum number of instances for each branch $I = 15$; the number of iterations on MLR productivity model $N = 30$; the variance index threshold to check the acceptance of the model α was set to be below 0.30. The number of folds to cross validate the model and the minimum number of instances for each branch were reduced considering the limited size of the dataset. The resulting productivity model is given in Eq. 3.24 (as per Eq. 3.7) and the variance of the estimated productivity is shown in Eq. 3.25. The enhanced model tree algorithm found four MLR equations classified by rules over the current scope of productivity study, summarized in Table 3.8.

$$P = P_{X2}^0 \times X2 + P_{X4}^0 \times X4 + P_{X5}^0 \times X5 + P_{X7}^0 \times X7 + P_{X9}^0 \times X9 + P_{X12}^0 \times X12 + P_{X16}^0 \times X16 + P_{X17}^0 \times X17 \quad (3.24)$$

$$C_P = {\sigma_P}^2 = \left(\frac{\partial P}{\partial P_{X2}^0}\right)^2 var(P_{X2}^0) + \left(\frac{\partial P}{\partial P_{X4}^0}\right)^2 var(P_{X5}^0) + \left(\frac{\partial P}{\partial P_{X5}^0}\right)^2 var(P_{X5}^0) + \left(\frac{\partial P}{\partial P_{X7}^0}\right)^2 var(P_{X7}^0) + \left(\frac{\partial P}{\partial P_{X9}^0}\right)^2 var(P_{X9}^0) + \left(\frac{\partial P}{\partial P_{X12}^0}\right)^2 var(P_{X12}^0) + \left(\frac{\partial P}{\partial P_{X16}^0}\right)^2 var(P_{X16}^0) + \left(\frac{\partial P}{\partial P_{X17}^0}\right)^2 var(P_{X17}^0) \quad (3.25)$$

Table 3.8 MLR equations with rules for pipe spool productivity model.

Branch	Branch Logic (boundary conditions)	Productivity Model
1	X9 ≤ 0.575, X4 ≤ 0.003	(0.7173, 0.5212)X2 - (26.3898, 23.8687)X4 - (0.3894, 0.8927)X5 – (0.1986, 0.3529)X7 – (0.1275, 0.1301)X9 + (0.0408, 0.0105)X12 + (0.3339,0.1916)X17
2	X9 ≤ 0.575, and 0.003 < X4 ≤ 0.015	(0.6004, 0.4319)X2 + (15.9546, 4.9743)X4 - (0.2415, 0.6736)X5 + (0.1683, 0.4105)X7 – (0.0837, 0.2067)X9 - (0.0045, 0.0152)X12 + (0.6777, 0.3511)X16 - (0.1744,0.2885)X17
3	X9 ≤ 0.575, and 0.015 < X4	(0.4847, 0.2012)X2 + (0.7383, 1.264)X4 - (0.0639, 0.2927)X5 - (0.4908, 0.2181)X7 + (0.2203, 0.2334)X9 + (0.0119, 0.0069)X12 + (0.0948, 0.0896)X16 + (1.6399,0.5893)X17
4	X9 > 0.575	(0.3177, 0.3919)X2 + (2.3519, 4.2931)X4 - (0.0767, 0.3955)X5 - (1.2147, 0.3940)X7 + (0.3955, 0.0623)X9 - (0.0315, 0.0153)X12 - (0.0927, 0.2104)X16 - (0.5982,0.3757)X17

It is noteworthy that the square of correlation coefficient (R2) between actual and predicted productivity values is 0.68 (lower than 0.80) while the average variance index of the derived model is 0.51 (greater than 0.30 threshold). In particular, the outcome implies precision of the productivity model is below the threshold of acceptance, hence, the predictions made by the model

would not be recommended for practical applications. The second spool fabrication case, with limited records available (only seventy-one in total), the entire dataset is only utilized for training the enhanced model tree and check the model logic (its explainability) against practical knowhow; as a "fail" case, it is demonstrated how to reject a model with the proposed method. This helps check and validate the generalizability of the proposed model. The research is not to generalize the performance comparison between the model tree, the ANN and MLR. Results from the first case provide adequate information in this regard.

3.6 Discussion of Research Contributions

The hurdle of data availability in productivity study has been gradually overcome due to technological advances over the past few decades such as the proliferation of building information model (BIM) and labor hours tracking automation in the construction industry. Large datasets containing design features and labor cost data become inexpensively available and readily accessible, which are potentially valuable for predicting productivity by applying regressions or artificial intelligence (AI). As the productivity database continuously expands, the data used for productivity modeling and analysis represents a random sample taken at a particular time for the underlying productivity problem definition. Given the same problem domain, different datasets can be sampled by different modelers at the same time or by the same modeler at different time periods. As such, the productivity model calibrated by the same analytical method would end up with different parameters, resulting in a variance of the predicted productivity. This has given rise to a new research problem in connection with productivity prediction by applying regressions or ANN models: in addition to the accuracy (i.e., how accurate is the point-value output), how to

determine the precision of a productivity prediction model (i.e., the variance of the output), which is also crucial to validation and utilization of the model as decision support tool.

This research improves the variance estimation technique of nonlinear regression-based prediction model -namely the model tree- for modeling construction productivity. Unlike artificial neural networks (ANN), the model tree is a logic driven machine learning model which ensures transparency in model development in regards to explaining nonlinear input-output relationships. The enhanced model tree is preferred over other machine learning techniques because (1) the variance in the productivity estimate is analytically predicted alongside the point-value output, and (2) the productivity model is explainable in terms of the reasoning logic for productivity prediction. Such transparency in model structure is essential for gaining trust from practitioners and promoting the implementation of the AI model. The model tree is utilized in this research by taking advantage of its function as a nonlinear classifier, which partitions the dataset into discrete subsets delineated by binary rules. The ensuing data classes within each subset lend themselves to straightforward MLR modeling. Next, the proposed approach enhances the model tree by coupling with variance analysis, which is connected with the application of a random sampling technique for statistically characterizing the MLR coefficients. This novel methodology aligns closely with the intended objective of devising explainable AI for estimating the variance associated with the prediction of productivity.

The "variance index" is introduced to check the model performance based on the presented method of variance calculation for predicted output. The variance index calculation requires the standard deviation (square root of variance), determined individually for each input setting. Note this is in contrast with calculating the variance of the output from the absolute errors by comparing actual

and predicted values over the entire model dataset. However, the traditional MLR and ANN models lack the capability to determine the variance associated with each prediction, thus disallowing quantification of the variance index.

It is noteworthy that the error propagation theory underpinning the enhanced model tree assumes that a coefficient in the MLR equation at each model tree branch follows a Gaussian distribution. Nonetheless, this assumption may not always hold true in the context of modeling labor productivity in construction, thus constraining the research application. On the other hand, when well-structured productivity data are not readily available, it would be an acceptable practice to approximate a normal distribution by estimating the mean and standard deviation in a way resembling range estimating or Program Evaluation and Review Technique (PERT) in construction management. That is to assume Gaussian distribution and derive mean and standard deviation from Optimistic (L), Most Likely (M), and Pessimistic (U) estimates (Peurifoy and Oberlender, 2001). Meanwhile, as data availability expands in precast and fabrication industry, an alternative emerges: fixing MLR element productivity parameters' mean and standard deviation by sampling the training set against the testing set. This sampling process is often linked to Gaussian distributions in statistics, which is demonstrated by correlated P-P plots on slopes of selected productivity elements in the steel fabrication case (shown in Appendix D).

An alternative method of calculating the variance of the predicted output can be using Ensemble learning technique. Ensemble learning involves combining the predictions of multiple machine learning models by averaging their outputs, often utilizing random subsets of data (Dietterich 2002). While this approach can provide predictions with variance estimates, it typically requires processing inputs using a large number of models to obtain an output along with a variance

estimate during the prediction stage. In contrast, the proposed enhanced model tree algorithm eliminates the need to process inputs separately to obtain the variance of the model, thereby substantially improving computing efficiency. substantially improving computing efficiency.

3.7 Conclusions

The present research enhances the established AI technique called Model Tree to quantity the variance for the predicted productivity due to random sampling of training data. In this research, a productivity problem is decomposed into branches (classes) by applying the non-linear classifier algorithm in Model Tree. Then the Model Tree is enhanced in such a way that the MLR equation on each tree branch can predict the point-value output as well as the associated variance. This research essentially builds on recent progresses in Model Tree applications in productivity study (Mohsenijam et al. 2021; Naumets and Lu 2021) and MLR in variance analysis (Hasan and Lu 2022) in efforts to extend the capability of the nonlinear AI method of Model Tree in modeling construction productivity. Prefabrication of the same-type structural components in building and infrastructure construction is still classified as "made-to-order" labor-intensive production processes; productivity could broadly fluctuate because of product variations in detailed elements or special features. A case of modeling the productivity for prefabrication of structural steel is presented to demonstrate the application of the proposed methodology. For cross checking, multiple linear regressions (MLR) and artificial neural networks (ANN) were independently applied to the same problem definition using the same data. The enhanced Model Tree outperformed MLR in prediction accuracy; and was preferrable over ANN considering (1) the prediction of the variance alongside the point-value output and (2) model explainability in terms of reasoning logic in prediction.

The enhanced model tree algorithms as described in this paper are completely data-driven and analytical, given the productivity data available is sufficient. As such, the mean and variance of each coefficient associated with each productivity component in the proposed methodology are derived from random sampling of the training data. When only limited data are available that do not allow for random sampling, those parameters can be determined by alternative means; for instance, by resorting to the knowhow of experienced practitioners, who have years of experiences in performing the work process and are able to give reliable estimates for the most likely value, the minimum, and the maximum. In such a case, the mean can be approximated as a weighted average of the three points, while the standard deviation can be estimated based on the range, e.g., taking one sixth of the range by assuming normality.

It is stressed that the research is not conducive to "reducing waste or delay by these variances" at all. The proposed method is intended to predict the variance associated with a productivity estimate given a certain set of job attributes. This variance is the potential risk accounting for labor cost overrun. A large variance prediction indicates a high risk for cost overrun to occur if the predicted mean value for a certain method is used in estimate and bidding. The research is not directly conducive to "reducing waste or delay by these variances". It only helps with selecting a less risky Method A (less variance) over Method B, note the predicted mean value on Method B can be more favorable than A considering the mean productivity alone.

Finally, not specific to productivity models in the domain of construction research, the focus of regressions and AI based prediction modeling research has been confined to increasing prediction accuracy, largely overlooking the precision of the model due to the dynamic way by which data is selected from the available database. Model validations have been mainly based on evaluating

error terms between the predicted output and the target output, without addressing the variance of the predicted output. Therefore, the enhanced Model Tree model framework holds potential to prove itself as explainable AI or non-linear regression technique across a wide range of application domains.

Chapter 4

Estimating Output Variance of a Regressing Tree Model: Case Study of Concrete Strength Prediction

4.1 Introduction

The application of machine learning algorithms has revolutionized the field of civil engineering. These algorithms enable engineers to analyze large datasets and make predictions that would otherwise be difficult or impossible using traditional methods (Huang et al. 2019). However, the increasing complexity of these algorithms has also highlighted the need for explainable artificial intelligence (XAI) techniques (Belle and Papantonis 2021). These techniques help ensure that the models are transparent and understandable; hence, model predictions can be trusted by decision makers. The Model Tree approach is one such technique that has become popular in recent years. This approach combines decision trees and regression analysis into an XAI modeling framework for generalizing complex-nonlinear relationships between variables (Frank et al. 1998). The Model Tree has been applied to address various civil engineering problems such as predicting the construction labor productivity and the workability of concrete (Mohsenijam et al. 2022), forecasting river flow (Taghi Sattari et al. 2013), sediment transport in pipes (Najafzadeh et al. 2017), predicting compressive strength of high-performance concrete mix (Deepa et al. 2010).

Typically, the performance of a numerical model is evaluated by calculating the relative and absolute errors and assessing the statistical correlation between the model's output and the target values (Alexander 2020). These measures aim to assess model bias, variance, and complexity (Yu et al., 2006). Root Mean Square Error (RMSE) or the coefficient of determination (R^2) are commonly applied measures for model accuracy. However, precision estimate, which is the variance estimate tied with individual model output, has yet to be investigated but is of vital importance and immediate relevance to many civil engineering applications. For instance, a concrete strength prediction model could produce a prediction based on the input data. However, it will not indicate the estimate of uncertainty surrounding the expected result, which can be crucial for design or quality assurance purposes.

In this research, the Model Tree algorithm has been enhanced with the intention to propose a framework to account for the variance estimate of the predicted output. The commonly applied model tree algorithm of M5P, has been chosen as the basis. M5P calibrates multiple linear regression (MLR) models at the leaf node of a decision tree by recursively splitting the data into subsets based on the values of the input variables. The variance analysis method for MLR model proposed by Hasan and Lu (2022) is integrated in M5P to estimate the variance of the predicted output. To demonstrate the application of the enhanced model tree approach, a common civil engineering problem: concrete strength prediction has been choosen. The dataset was taken from the University of California, Irvine, machine learning repository (UCI 2020). The remainder of this paper further explains the enhanced model tree approach as well as the application steps.

4.2 Variance Analysis on Model Tree

Combining a decision tree with regression analysis allows the breaking down of a large dataset into smaller subsets. This approach enables the representation of nonlinear input-output relationships through a series of linear multivariate data models using unique sets of binary rules. To implement the variance analysis technique proposed by Hasan and Lu (2022), it is necessary to determine the mean and standard deviation of the coefficients for each multiple linear regression (MLR) model in the model tree. Subsequently, the theory of error propagation can be applied to obtain a variance estimate for predictions given a specific set of inputs. The application framework for the enhanced model tree is illustrated in Fig. 4.1 and comprises three main steps.

Step 1. The enhanced model tree algorithm starts with applying the M5P algorithm to classify the dataset into small subsets suitable for constructing MLR models. M5P uses a divide-and-conquer strategy in breaking down the dataset of a complicated problem into smaller subsets until a stopping criterion (minimum number of instances) is met (Quinlan 1992). The n-fold cross-validation technique is used to check the performance of the decision tree model. The correlation coefficient, mean absolute error, and root mean squared error between the output and target variables are used to check the model's prediction accuracy. If the model accuracy is satisfactory, then move to the next step.

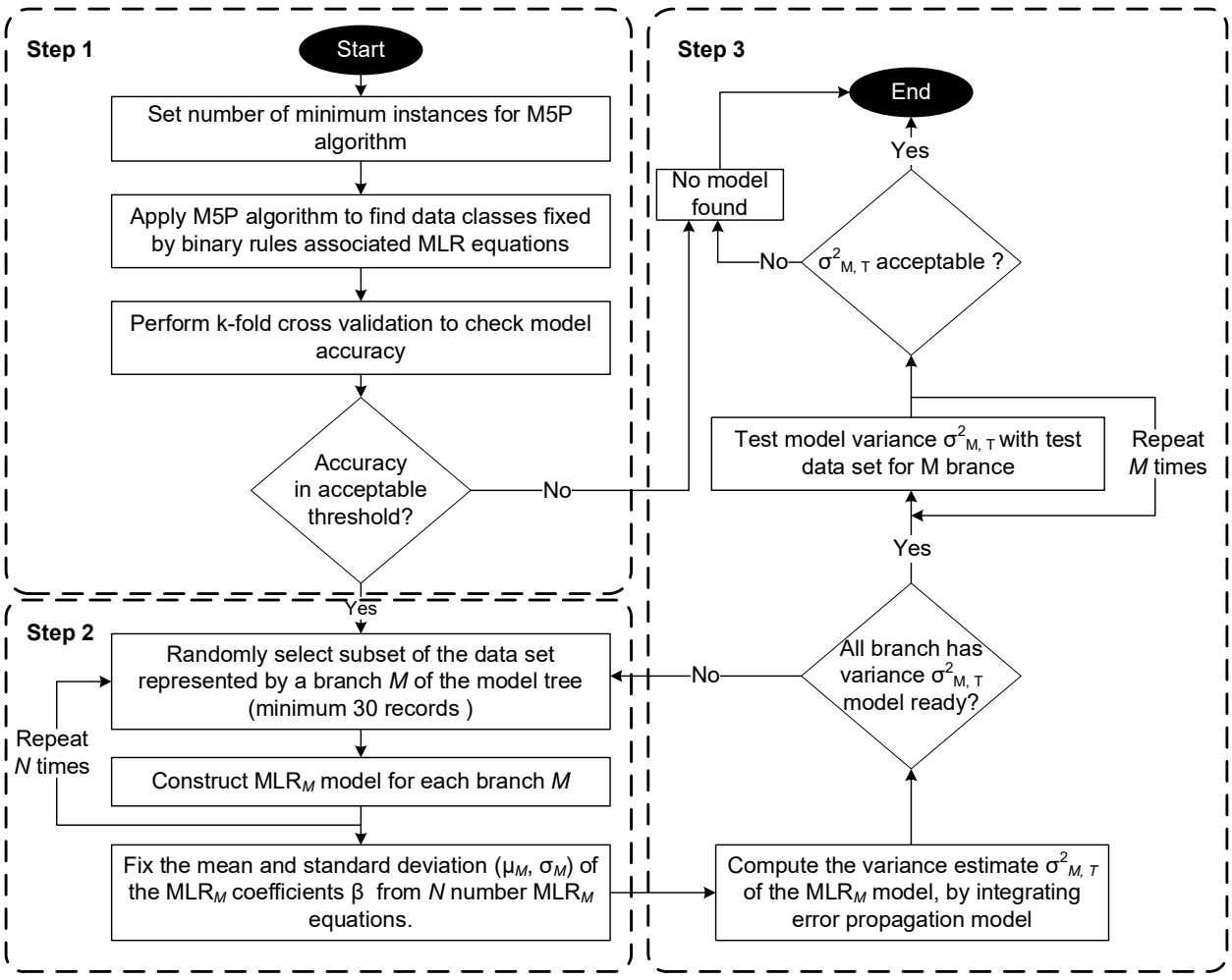


Figure 4.1: Variance analysis framework for m5p model.

Step 2. A subset of the total data at each branch of the model tree is randomly selected to prepare an MLR model specific to that branch. For M *th* branch of the decision tree, the MLR model that relates the input variable X_i with the output variable Y_M is given in Eq. 4.1.

$$Y_M = \beta_{0,M} + \beta_{1,M}X_{1,M} + \cdots + \beta_{i,M}X_{i,M} \qquad (4.1)$$

Here, $\beta_{i,M}$ is the regression coefficient associated with input variable $X_{i,M}$. $\beta_{0,M}$ is the intercept of the MLR equation representing the model bias. Now if the MLR equation had been calibrated to a random data sample from the available data at a specific branch, the $\beta_{i,M}$ would be different

each time. And for N number of random iterations, there will be N number of MLR equations form which mean and standard deviation (μ, σ) of each β parameter can be calculated. Therefore, the final form of Eq. 4.1 will be (Eq. 4.2):

$$Y_M = \beta_{0,M}(\mu_{0,M}, \sigma_{0,M}) + \beta_{1,M}(\mu_{1,M}, \sigma_{1,M})X_{1,M} + \cdots + \beta_{i,M}(\mu_{i,M}, \sigma_{i,M})X_{i,M} \qquad (4.2)$$

This process should be performed for all the model tree branches (total M times), to prepare total M number of MLR models constrained by the logic of each branch of the decision tree.

Step 3. Next step is to apply the error propagation theorem to estimate the variance of the prediction. Error propagation theorem explains the propagation of random errors from independent variables to the dependent variable in a numerical system, expressed by Eq. 4.3.

$$C_y = J_{xy} C_x J_{yx}^T \qquad (4.3)$$

Here, C_y is the covariance matrix of random output y, and C_x is the covariance matrix of random input x, and J_{xy} is the Jacobian (Jacobian matrix) of the Eq. 4.2.Now, to examine the total error in output Y_M due to the errors propagated from the model coefficients $\beta_{i,M}(\mu_{i,M}, \sigma_{i,M})$, Eq. 4.3 is applied in connection with Eq. 4.2. The final output variance of the MLR model is given in Eq. 4.4:

$$C_Y = {\sigma_{T,M}}^2 = \left(\frac{\partial Y_M}{\partial \beta_{0,M}}\right)^2 var(\beta_{0,M}) + \left(\frac{\partial Y_M}{\partial \beta_{1,M}}\right)^2 var(\beta_{1,M}) + \cdots + \left(\frac{\partial Y_M}{\partial \beta_{i,M}}\right)^2 var(\beta_{i,M}) \qquad (4.4)$$

The decomposition process of Eq. 4.3 and construction of Eq. 4.4 can be found in Hasan and Lu (2022). The variation level of each MLR equation then can be tested by using a metric, if the

results are satisfactory (less than the threshold), the model is accepted for making point value prediction as well as the associated variance estimate. Otherwise, the tree model is rejected.

4.3 Concrete Strength Prediction: Case Study

The dataset used in this study consists of a total of 1030 instances with eight attributes denoting high performance concrete (HPC) design properties and the corresponding compressive strength. For multifold cross validation in the development of the model tree, 90% (927) from the data set were used in model calibration, while the rest randomly selected 10% (103 instances) were reserved for verification purposes to test the performance of the overall framework. The basic data properties of the case study date set are given in Table 1.

Table 4.1 Data properties of the HPC dataset for strength prediction model.

ID	Attribute	Maximum	Minimum	Median	Mean	Standard Deviation
X1	Cement (kg/m3)	540	102	272.9	281.2	104.5
X2	Blast furnace slag (kg/m3)	359.4	0	22	73.9	86.2
X3	Fly ash (kg/m3)	200.1	0	0	54.2	64
X4	Water (kg/m3)	247	121.8	185	181.6	21.4

ID	Attribute	Maximum	Minimum	Median	Mean	Standard Deviation
X5	Super Plasticizer (kg/m3)	32.2	0	6.35	6.2	6.0
X6	Coarse Aggregate (kg/m3)	1145	801	968	972.9	77.7
X7	Fine Aggregate (kg/m3)	992.6	594	779.5	773.6	80.1
X8	Age (days)	365	1	28	45.7	63.1
Y	Compressive Strength (MPa)	82.6	2.33	34.4	35.8	16.7

Step 1. To build the concrete strength prediction model, an M5P model was first developed using a total of 927 (90% of the sample) samples by using the WEKA software (Frank et al. 2016), keeping the soothing and pruning function enabled. The least number of instances was set as 50 (above the minimum threshold of 30), which means at least fifty records are required to calibrate MLR and apply the variance analysis. The model's performance was tested using 10-fold cross-validation techniques. The resulting correlation coefficient was reported to be 0.90, the mean absolute error was 5.8, and the root mean square error was approximately 7.5. Given the practical

application context, these performance results are considered acceptable (Kasperkiewicz et al. 1995), and we proceeded to the next step of the framework application. It is noteworthy that the model tree generated a total of 8 branches ($M = 8$), shown in Figure 2. Hence, each branch will result in one linear model (LM) for concrete strength prediction.

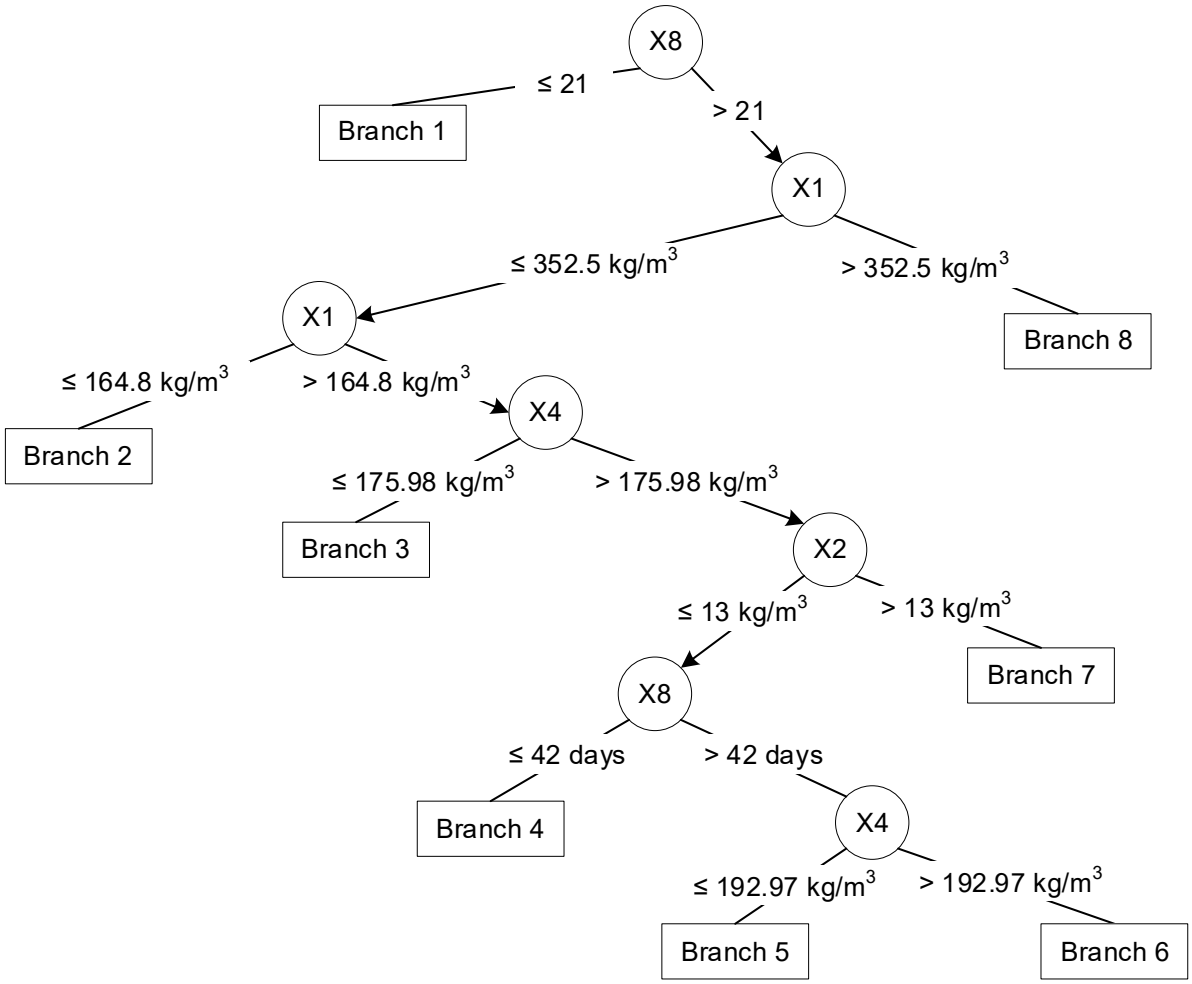


Figure 4.2: M5P model tree for concrete strength prediction model.

Step 2. In step 2 of the model development, the variance analysis model proposed by Hasan and Lu for each rule was applied to identify a linear model (LM) in the form of multiple linear regression (MLR) equations with associated variance estimate of the coefficients of the input

variables. The generic form of the MLR equations is given in Eq. 4.5 as per Eq. 4.2. Since there are eight input variables, the value of i would vary from 1 to 8.

$$Y_M = \beta_{0,M}(\mu_{0,M}, \sigma_{0,M}) + \beta_{1,M}(\mu_{1,M}, \sigma_{1,M})X_{1,M} + \cdots + \beta_{8,M}(\mu_{8,M}, \sigma_{8,M})X_{8,M} \quad (4.5)$$

Next, a random subset of data presented by the specific branch of the model tree was taken to develop an MLR model and this process was repeated 100 times (N = 100). The subset was selected in a way so that at least fifty records are available in the sample set to obtain statistically significant results. In each run, we obtained one MLR equation with a set of regression coefficient $\beta_{i,M}$. The mean and variance estimate $(\mu_{i,M}, \sigma_{i,M})$ of $\beta_{i,M}$ for the regression equation were then determined based on one hundred data samples. The procedure was followed to construct all the eight MLR models associated with the eight branches of the model tree, with mean and variance (μ_i,σ_i) estimates of the coefficients associated with input variables determined, given in Table 4.2.

The mean values of the MLR coefficients $\beta_{i,M}$ are used to predict concrete strength for a given mix design parameters ($X_{i,M}$). For example, if the mix properties of a concrete batch are: *X_1 = 190.34 kg/m^3, X_2= 0, X3=125.18 kg/m^3, X4=161.85 kg/m^3, X5=9.88 kg/m^3, X6=1088.1 kg/m^3, X7=802.59 kg/m^3, X_8=14 days,* then first we can determine that, this concrete mix falls under Branch 1. As per branch 1 MLR model, the concrete strength *Y_1* is (as per Eq. 4.6) *22.72 Mpa.*

Table 4.2 MLR equations for all 8 branches.

	Branch Logic	MLR Model
Branch 1	X8≤21days	$Y_1=(0.091,\ 0.004)X_1+(0.055,\ 0.005)X_2+(0.040,\ 0.008)X_3-(0.109,0.022)X_4+(0.376,\ 0.096)X_5+(0.001,\ 0.003)X_6-(0.001,\ 0.004)X_7+(1.328,\ 0.08)X_8$
Branch 2	X8>21days, and X1≤164.8 kg/m^3	$Y_2=(0.026,\ 0.042)X_1+(0.122,\ 0.008)X_2+(0.05,\ 0.01)X_3-(0.086,0.029)X_4+(0.062,\ 0.146)X_5+(0.003,\ 0.005)X_6+(0.015,\ 0.005)X_7+(0.069,\ 0.013)X_8$
Branch 3	X8>21days, and 164.8 kg/m^3<X1≤352.5 kg/m^3, and X4≤175.98 kg/m^3	$Y_2=(0.112,\ 0.022)X_1+(0.109,\ 0.014)X_2+(0.063,\ 0.025)X_3-(0.154,0.054)X_4+(0.03,\ 0.027)X_5+(0.017,\ 0.009)X_6+(0.009,\ 0.008)X_7+(0.168,\ 0.024)X_8$
Branch 4	X8>21days, and 164.8 kg/m^3<X1≤352.5 kg/m^3, and X4>175.98 kg/m^3, and X2≤13 kg/m^3, and X8≤ 42 days	$Y_2=(0.155,\ 0.022)X_1-(0.069,\ 0.038)X_2+(0.105,\ 0.022)X_3-(0.175,0.078)X_4+(0.236,\ 0.255)X_5+(0.002,\ 0.019)X_6+(0.023,\ 0.022)X_7-(0.245,\ 0.19)X_8$

	Branch Logic	MLR Model
Branch 5	X8>42days, and 164.8 kg/m^3<X1≤352.5 kg/m^3, and 175.98 kg/m^3<X4≤ 192.97 kg/m^3, and X2≤13 kg/m^3	*Y_2=(0.17, 0.027)X_1 +(0.071, 0.041)X_3-(0.215,0.139)X_4+(1.629, 0.628)X_5+(0.002, 0.013)X_6+(0.024, 0.017)X_7+(0.016, 0.001)X_8*
Branch 6	X8>42days, and 164.8 kg/m^3<X1≤352.5 kg/m^3, and X4> 192.97 kg/m^3, and X2≤13 kg/m^3	*Y_2=(0.17, 0.036)X_1+(0.69, 0.897)X_2+(0.061, 0.029)X_3-(0.574,0.207)X_4+(1.805, 1.724)X_5+(0.053, 0.023)X_6+(0.046, 0.025)X_7+(0.019, 0.006)X_8*
Branch 7	X8>21days, and 164.8 kg/m^3<X1≤352.5 kg/m^3, and X4>175.98 kg/m^3, and X2>13 kg/m^3	*Y_7=(0.09, 0.016)X_1+(0.04, 0.012)X_2-(0.021, 0.021X_3-(0.032,0.041)X_4+(0.792, 0.272)X_5+(0.01, 0.01)X_6+(0.005, 0.01)X_7+(0.051, 0.011)X_8*
Branch 8	X8>21days, and 352.5 kg/m^3<X1	*Y_8=(0.115, 0.012)X_1+(0.166, 0.013)X_2+(0.093, 0.016)X_3-(0.218,0.032)X_4-(0.414, 0.154)X_5+(0.017, 0.006)X_6+(0.025, 0.007)X_7+(0.039, 0.01)X_8*

$$Y_1 = (0.091)\,190.34 + (0.055)0 + (0.040)\,125.18 - (0.109)\,161.85 + (0.376)\,9.88 + (0.001)\,1088.1 - (0.001)\,802.59 + (1.328)\,14 = 22.72\,MPa \quad (4.6)$$

Step 3. Variance estimate model was formulated following Eq. 4.4 for all 8 MLR models as prepared in step 2 (given in Table 4.3). For example, the variance estimate model for Branch 1 would be (Eq. 4.7):

$$\sigma_{T,1}{}^2 = \left(X_{1,1}\right)^2 var\left(\beta_{1,1}\right) + \left(X_{2,1}\right)^2 var\left(\beta_{1,1}\right) + \cdots + \left(X_{8,1}\right)^2 var\left(F_{\beta_{i,M}}\right) \quad (4.7)$$

So, given the same concrete mix as previously mentioned in Step 2, the estimated variance of the prediction would be 38.16, as per (Eq. 4.8) (i.e., standard deviation of 6.17 MPa.) Note, the variance estimate result can be verified by Monte Carlo simulation approach and details of such verification methodology can be found in Hasan and Lu (2022).

$$\sigma_{T,1}{}^2 = (190.34 \times 0.004)^2 + (125.18 \times 0.008)^2 + (161.85 \times 0.022)^2 + (9.88 \times 0.096)^2 + (1088.1 \times 0.096)^2 + (802.59 \times 0.003)^2 + (14 \times 0.08)^2 = 38.16 \quad (4.8)$$

4.4 Model Performance

The performance of the enhanced model tree was tested with the reserved 10% of the data from the dataset -a total of 103 instances in the testing set. The concrete strength prediction was made for each case. The prediction result gave a point value prediction associated with a standard deviation of the prediction. The accuracy of the results was tested by comparing the point-value model output with the target output in the test set. The Pearson's coefficient of correlation (R^2) of the predicted and target concrete compressive strength is 0.88 and the mean absolute error of

the prediction is 5.91. The R square value at zero intercept is 0.98, which is deemed satisfactory. The output vs the target plot shows a near 45-degree tilt line as shown in Fig. 4.3(a).

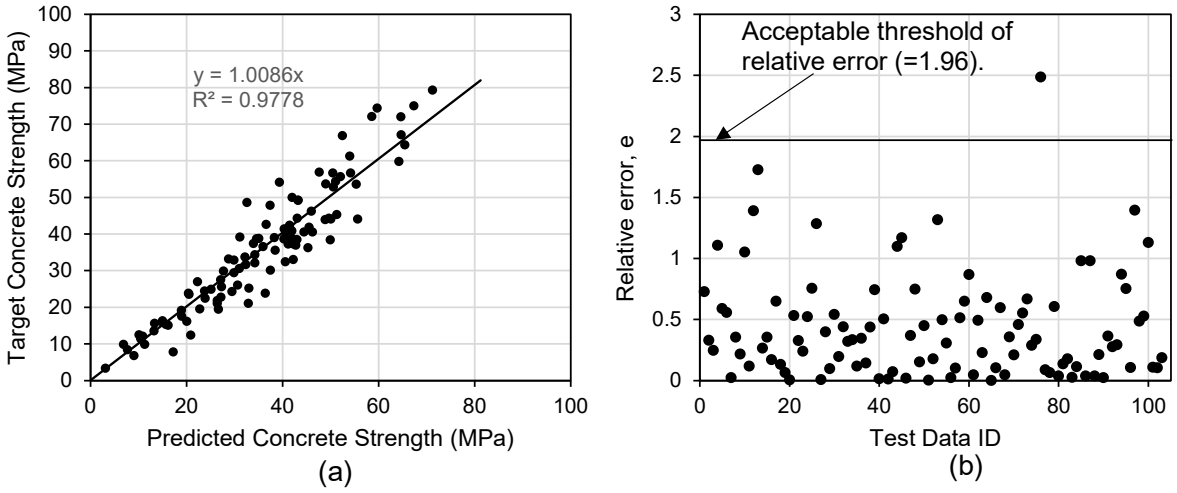


Figure 4.3: Model performance: (a) predicted concrete strength (b) predicted variance of the concrete strength.

To compare the prediction error with the predicted variance estimate (STD), a relative error term is defined to divide the actual absolute error over the standard deviation of the output as per Eq. 4.9. The error data for the 103 test records are plotted in Figure 4.3(b) showing that, all the prediction variances are below the upper threshold of the predicted error of 1.96. Note 1.96 is also the cut value applicable to the standard deviation in determining the 95% confidence interval of the sampled mean in statistics.

$$Relative\ error, e\ = \frac{|prediction\ error|}{STD\ of\ the\ Output} = \frac{|Output - target|}{STD\ of\ the\ Output} \quad (4.9)$$

4.4 Conclusion

The major limitation of the MLR model lies in its insufficiency to represent the nonlinear relationships between inputs and output common to engineering applications. Machine learning algorithms, including artificial neural networks, are much more powerful in coping with nonlinear regressions but lack transparency in explaining the model's decision-making logic. In contrast, decision trees are a type of machine learning algorithm that recursively partitions input data into subsets based on feature values, assigning a label or value to each subset based on the majority class or average value of the output variable. The resulting tree structure allows easy interpretation and visualization of data classification and model prediction. Combining decision trees with regression analysis, a model tree essentially breaks down nonlinear input-output relationships into a series of MLR models. The variance analysis technique proposed by Hasan and Lu (2022) can be integrated into the model tree to further enhance its capability of estimating variances of predicted point values for any given input setting, which has been demonstrated in a concrete strength prediction case in this paper. Although the application of the proposed framework is explained from the perspective of the concrete strength prediction model, it will potentially broaden the application scope in solving many civil engineering problems.

Chapter 5

Error Propagation Model for Analyzing Project Labor Cost Budget Risks in Industrial Construction

5.1 Introduction

Industrial construction encompasses a wide range of projects that are essential to our utilities and basic industries and generally features large amounts of the highly complex process of piping, mechanical, electrical, and instrumentation work (Barrie and Paulson, 2001). In contrast with infrastructure construction, industrial construction does not require fleets of construction equipment and plant (such as scrapers, loaders, cranes and trucks, etc.) to handle basic materials (such as earth, rock, concrete, and asphalt, etc.), but "tends to be much more labor-intensive, though some of the largest hoisting and materials-handling equipment is also required" (Parker et al., 1984). Hence, labor cost on major activities in industrial construction is conventionally determined independently of equipment use. To perform takeoff on a particular work package, the key is to determine labor productivity, which could vary broadly among different projects (Parker et al., 1984). For instance, for field pipe installation in industrial construction, the amount of work-in-place is usually counted in pipe footage; field productivity for pipe installation is measured in labor hours per foot of installed pipe. Then, the uncertainty in labor cost estimate (in labor hours required) is largely attributable to the variation in labor productivity.

Well established cost estimate and project schedule are consolidated into a reliable cost budget, which underpins effective cost control practice in project management and is crucial to successful project delivery (Ahuja et al. 1994). In industrial construction, labor cost accounts for the bulk of the total project cost, usually ranging from a quarter to a half of the total. Besides, among major project cost components (material, equipment, and labor), labor cost is considered as the project element with the highest risk (Hanna et al., 2005). Therefore, in making budgets for labor costs on industrial construction projects, the common practice is to apply labor hours (LH) based budgeting.

Cost planning starts with defining work packages under the project work breakdown structure (WBS), based on which work is scheduled and estimated, monitored, and controlled (PMI, 2017). Next, the estimator determines the quantity of work and chooses the resource group to complete each individual work item (activity) in a work package based on the project WBS. Then, the total LH required to perform that activity for a particular work package (WP) is calculated accordingly. By specifying sequencing constraints (logical and technological) between work packages, critical path method (CPM) based project planning tools (such as Primavera P6) can be applied to produce project schedule and labor cost budget.

Labor productivity (LH required per unit of work) provides the best indicator of the production efficiency and represents the key factor for estimating labor cost in construction (Dozzi and AbouRizk, 1993; Rojas and Aramvareekul, 2003). In fabricating and installing made-to-order components in industrial construction, every activity in a project has its unique parameters for describing work complexity and defining work content. The current practice of planning industrial construction relies on "productivity benchmark" data, which represents average productivity

performance determined statistically from corporate historical databases or national industry standards (Park et al., 2005). The "productivity benchmark" serves a good basis, which is generally overridden given available data and information on company-specific actual performances and on project-specific productivity-related factors (Park et al., 2005). The evidence of using data analytics for the purpose of better productivity prediction is well documented in the literature (Minato and Ashley, 1998). Various techniques have been utilized in productivity related research to model the complex relationships between different external influencing factors and the productivity rates, namely: statistical and regression models (Smith, 1999; Mohsenijam and Lu 2019), expert systems, artificial neural networks (Sonmez and Rowings, 1998; Lu et al., 2001), and operations simulation (Zayed and Halpin, 2005; Nasirzadeh and Nojedehi, 2013). It is commonly argued that in connection with the fabrication and installation of engineering products in industrial construction (such as pipe spool, structural steel components, precast concrete structural units, etc.), the takeoff on one particular work item needs to be multi-dimensional (dependent on multiple parameters each denoting a certain feature of the job or the project), instead of reducing to just one single parameter (e.g., total area, or weight, or volume).

In general, uncertainties inherent in both internal and external project environments presented distinctive challenges in establishing proper productivity benchmarks (Thomas, 2015). Methods for accounting for risks in project labor cost budgeting due to variations in labor productivity still lacks scientific rigor and can be a subjective decision process. This has potentially led to insufficient budgeting in the planning stage, thereby resulting in budget overrun in the execution stage.

5.1.2 Risk Analysis Methods: Critical Review

Major methods for analyzing risks inherent in a system output (such as labor cost budget) due to variations in input variables (such as productivity related factors) generally entail two steps, namely: Step 1: Establishing the system model to relate input variables with system output; Step 2: Given a valid system model, sensitivity analysis is applied by perturbing an input variable within its practical range of variation, resulting in the observation of system output response. Hence, risks in the system output are generalized from the observed system output data by performing "what-if" scenario analysis or simulation analysis.

In Step 1, researchers commonly resort to multiple linear regression, fuzzy expert system or artificial neural network (ANN) models through data-mining when physical or logical processes of the system are not clearly defined, but the real-life system has recorded historical data allowing direct input-to-output mapping (Kisi et al., 2017). Alternatively, given underlying processes of the system can be clearly, logically represented in a computer simulation platform (e.g., discrete event simulation or critical path network scheduling), operation simulation models can be established in Step 1 through mapping processes over time and space of the problem domain, (Halpin and Riggs, 1992; Mulholland and Christian, 1999).

Analyses in Step 2 demand the design of a large number of likely scenarios denoting variability and uncertainty on input variables (by design or by random sampling) and conducting experiments by assessing each scenario with the established model on the computer. Examples are what-if scenario analysis on an operations simulation model (Chan and Lu, 2008); or Monte Carlo simulation experiments by describing variability in input variables with statistical distributions

(Lu et al., 2001). In general, knowledge and knowhow in connection with the modeling method, the computer tool, coding, and statistical analysis are indispensable in implementing research.

The law of error propagation expresses the relationship between random variable errors and the corresponding function (Koch, 1999). It is the fundamental formula for the evaluation of precision in adjustment theory (Amiri-Simkooei et al. 2016). Nonetheless, error propagation has been a less applied alternative in construction research to substitute for "what-if" scenario simulation experiments or Monte Carlo sampling techniques in Step 2. With a well-established mathematical theory as its foundation, its effective application is dependent on a sufficient system model resulting from in Step 1 and the ensuing analytical formulation of the error propagation based on the system model. The user only needs to plug numbers in the derived formula in order to quantify the risks in the system output (labor cost budget) due to variations in input variables (labor productivity). Obviously, error propagation potentially provides more efficient and practical decision support in a deterministic time window, which does not require the use of professional computer software, sensitivity analysis, or Monte Carlo sampling.

Applying critical path method or discrete event simulation has been the mainstream methodology in construction system modeling. Monte Carlo simulation lends itself well to performing risk analysis on such models. One main advantage with Monte Carlo simulation lies in the ease with which to communicate its underlying random sampling process to domain subject experts who generally are not simulation experts. Hence, the Monte Carlo simulation has been the preferred technique for risk analysis in construction engineering and management. In short, uncertainty quantification and analysis based on error propagation theory has been less researched in the construction domain and is yet to be fully integrated with research and practice in construction

engineering despite its proven usefulness and maturity in other sectors (Veregin 1995; Lichti et al. 2005; Puatanachokchai & Mikhail, 2008; Wang et al. 2018). To the authors' best knowledge, there is not yet one published paper on applying error propagation theory for uncertainty quantification and risk analysis in construction planning.

5.1.2 Research Overview

The objectives of this research are to determine the effect of productivity uncertainties upon labor cost budgeting in a quantitatively reliable, statistically significant manner, while depicting the S-curve, which is a visual representation of accumulated labor cost (LH) of a project against project time, as the *S stripe* which portrays an interval about the mean value with a certain confidence level. In short, the present research develops an error propagation model to analyze the effect of labor productivity variability on labor cost budget. It proposes the concept of "S stripe" against the "S-curve" for representing the derived risks in the project budget. In particular, how to apply the proposed analytical methodology is illustrated with a steel fabrication project case. Monte Carlo simulation is applied to cross-validate the analytical method and to contrast simulation against the newly proposed method in the particular application context of steel fabrication labor cost budgeting. In drawing conclusions, the advantages of the newly proposed method are recapitulated from a practical application perspective, and the significance of the present research in terms of risk analysis for construction engineering and management in general, is also addressed. In the ensuing section, we first present a critical review of mainstream methods for risk analysis in construction.

5.2 Error Propagation Model for Risk Analysis in Labor Cost Budgeting

5.2.1 Multi-dimensional labor-hours estimating

In the context of industrial construction, direct labor costs are generally compiled from a comprehensive list of work items to be performed in (Thomas and Sakarcan, 1994). It is noteworthy that in the current practice, the list of work items represents the estimator's interpretation of work performed by skilled trades. Total labor-hours required (LH_T) for handling and connecting materials into the final product is calculated as per Eq. 5.1. Next, the required total labor-hour (LH_T) is used to determine the time duration of a specific work item based on the number of workers involved in the work item, as per Eq. 5.2.

$$LH_T = \sum LH = P_{LH} \times W_r \qquad (5.1)$$

Here,

$\sum LH$ = Total man hour required to complete any job.

P_{LH} = Labor productivity (LH/Unit);

W_r = Total Units of work performed (quantity takeoff).

Now, if we would like to find the total time required to complete the job, that would be,

$$T = \frac{LH_T}{n_L} \qquad (5.2)$$

Here, T = Time required to complete, Wr units of the job (quantity takeoff) to be done by the number of workers of particular skilled trades involved in the work item, namely n_L.

The takeoff on one particular work item is multi-dimensional (dependent on multiple parameters, each denoting a certain feature of the work item). The labor hours required is thus represented by a mathematical equation correlating multiple input features with labor hours. A general form of the equation is given in Eq. 5.3:

$$LH_T = \sum LH = P_{LH} \times W_r$$

$$= f(x_1, x_2, x_2, \dots\dots, x_n)$$

$$= f(x_i); i = 1, 2, 3, \dots\dots, n \qquad (5.3)$$

Here, x_n represents all the parameters in calculation of labor hours, each having associated productivity rate (P_{LH}) and work unit takeoff (W_r).

A data-driven model for establishing the labor cost budget (LH as shown in Eq. 5.1) essentially consolidates labor cost estimate and project schedule. In this research, the labor hours are calculated by a linear regression equation, as expressed in Eq. 5.4. The activity level uncertainties, which are identified to account for productivity variations, are represented in the coefficients (β) in terms of standard deviation (σ) of average value (μ).

$$LH_T = f(x_i)$$

$$= \beta_0 + \beta_1 x_1 + \beta_2 x_2, \dots\dots, +\beta_n x_n$$

$$(\beta_0, \beta_1, \beta_2 \dots\dots, \beta_n) = [(\mu_0, \sigma_0), (\mu_1, \sigma_1), (\mu_2, \sigma_2), \dots\dots, (\mu_n, \sigma_n)] \qquad (5.4)$$

In Eq. 5.4, x_i and β_i are the independent parameters and coefficients, where, μ_i and σ_i are the mean and standard deviation of productivity rate β_i. Herein, μ_i and σ_i represent the average productivity performance and the variability of the coefficient β_i, respectively; while, x_i denotes the associated quantity takeoff for a particular work item, which is constant.

Mohsenijam and Lu (2019) described a practical case of developing the labor hour regression equation for estimating total LH for each job (e.g., handling, cutting a division, etc.) in the application context of structural steel fabrication. The eleven input variables identified in Mohsenijam and Lu (2019) as the most relevant for determining required LH given a particular division of structural steel are shown in Table 1.

Table 5.1 List of variables to estimate LH for structural steel fabrication.

Variable (x_i) - Unit	Structural Feature Type	Unit
x_1	Material Weight	Weight (Kg)
x_2	Material Steel Section	Length (m)
x_3	Material Plate	Area (m^2)
x_4	Material Hollow Section	Length (m)
x_5	Material Cold Formed	Length (m)
x_6	Material Bars	Length (m)

Variable (x_i) - Unit	Structural Feature Type	Unit
x_7	Material Anchors	Quantity (number)
x_8	Connection: Weld	Length (m)
x_9	Connection: Bolted	Quantity (number)
x_{10}	Material Pipe	Length (m)
x_{11}	Connection: Stud	Quantity (number)

5.2.2 Preparing Project Budget

Total LH required to complete any activity j depends on productivity rate and quantity takeoff encoded in Eq. 5.5,

$$LH_{T,j} = f(x_i) = f_j = \beta_{0,j} + \beta_{1,j}x_1 + \beta_{2,j}x_2, \ldots\ldots, +\beta_{i,j}x_i; \quad i = 1, 2, 3, \ldots, n \tag{5.5}$$

Here, x_i and β_i are the quantity takeoff parameters and productivity coefficients, respectively. Note x_i is constant, while β_i is a random variable following normal distributions with mean and standard deviation known.

As per Eq. 2, total mean time to complete that activity should be,

$$T_j = \frac{LH_{T,j}}{n_{Lead,j}}$$

$$= \frac{1}{n_{Lead,j}} \times \left(\beta_{0,j} + \beta_{1,j}x_1 + \beta_{2,j}x_2, \ldots\ldots, + \beta_{i,j}x_j\right) \qquad (5.6)$$

Here, T_j, is the time duration for the activity j;

Now, with activity duration determined using Eq. 5.6, the total project duration is to add up activity time duration along the critical path. The generic equation to find the total project duration is as Eq. 5.7, where the number of critical activities (on the critical path) is M,

$$T_T = \sum_{j=0}^{M} T_j \qquad (5.7)$$

The cumulative LH required at a particular time point t of the project is equal to the total LH of all activities completed up to time t. If the number of activities completed or partially completed up to time t is N, total LH required for the project until t is determined as Eq. 5.8:

$$LH_{Total,\ t} = \sum_{j,k}^{N,t} \left(LH_{T,j} \times \frac{t_{j,k}}{T_j} \right); \qquad \begin{cases} \left(t_{j,k} = t - t_{st,j}\right) & when, \quad t_{j,i} < T_j \\ \left(t_{j,k} = T_j\right) & when, \quad t_{j,i} \geq T_j \end{cases}$$

$$= \ f_t(x_i) = \ f_t \qquad (5.8)$$

Here,

$t_{j,k}$ = Time elapsed for the activity j;

t = Time elapsed for the entire project,

$t_{st,j}$ = Start time of activity j as per project schedule.

5.2.3 S Curve Plotting

The guide to the Project Management Body of Knowledge (PMBOK) defines “an S-Curve is a sigmoid function, that is a mathematical process or function that results in an S-shaped curve also called a Sigmoid Curve" (PMI, 2017). In general, the slope of the curve is flatter at the beginning and near the end, but is steeper in the middle section. In the industrial construction practice, the highly complex, nonlinear problem of deriving S curve for project planning and control is simplified to tally the cumulative LH up to a data date in the project schedule using Eq. 8. The S curve can be generated by plotting $LH_{Total,\ t}$ values at different time points t along with the total project duration. In the present research, the standard deviation of the cumulative LH at a particular time point due to known standard deviations in labor productivity at each work package is analytically derived, which seamlessly integrates labor cost estimating, project planning, scheduling, and budgeting, as shown in Fig. 5.1.

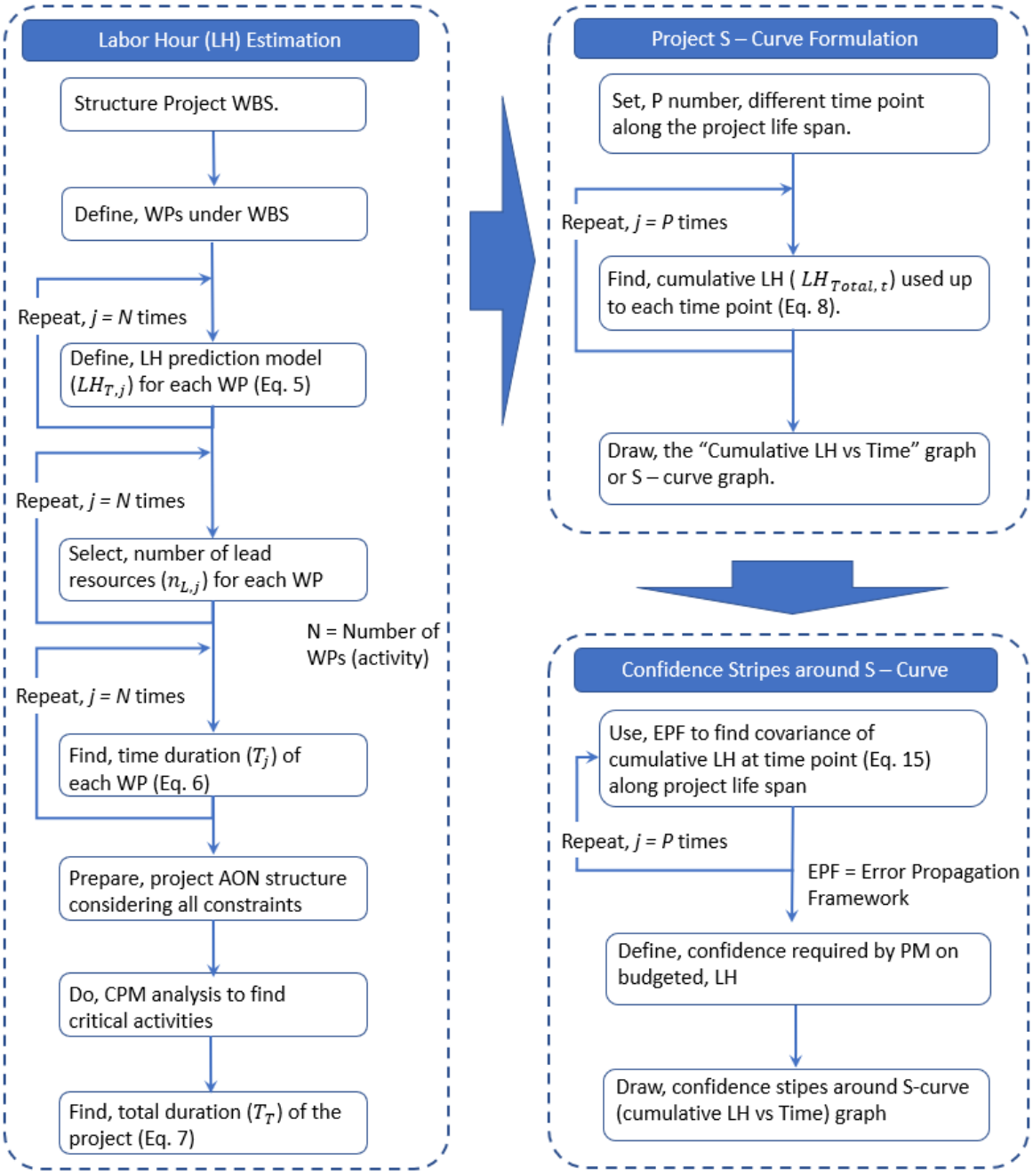


Figure 5.1: Proposed solution framework for construction project planning.

To establish a confidence interval around the average value, an error propagation model is critical to determine the standard deviation of the cumulative LH at a particular time point of the project. Analogous to plotting the S curve, the lower bound and upper bound of the interval at certain control points of the time line can be articulated to form the *S stripe*, resulting in visualization of the risk of labor cost budget due to variability inherent in labor productivity. In the following section, the analytical model development based on error propagation theory is illustrated with an application case.

5.3 Proposed Framework for Generating "S stripe"

5.3.1 Error Propagation Theory

The law of error propagation expresses the relationship between random variable errors and the corresponding function (Koch, 1999); providing the fundamental formula for evaluation of precision in adjustment theory (Amiri-Simkooei et al. 2016). Given a linear function, the variance or covariance of the error propagation model can be obtained analytically. A nonlinear function requires the linearization by Taylor series expansion prior to nonlinear error propagation; while it is theoretically possible to use Taylor series expansion to any existing order (Xue et al., 2015). The error of any system output can be obtained if the true nature of the error is known based on functional substitution with truncated Taylor Series (considering up to first order derivative).

In this research, we apply the error propagation theory to quantify the random error present in labor cost budgeting equations due to variations in labor productivity. Note variations in labor productivity are assumed to be random errors and denoted by the standard deviation of productivity rate for each work item in a work package. Applying first order derivatives to

approximate the propagation of random errors makes the solution algorithm simple and practical. The methodology presented herein is effective for characterizing random errors in the derived labor cost budget at each stage of the project and upon the final project completion. Besides, Monte Carlo (MC) simulation method is applied to the same case study for cross-validation and critical comparison against the new analytical method.

For any function, $y = f(x)$, measurement of the systematic error can be obtained by comparing the difference between y and its *Taylor Series* first expanded term, y_0, as expressed by the following equation (Eq. 5.9),

$$y - y_0 = \frac{\partial y}{\partial x}(x - x_0) \qquad (5.9)$$

Now, if y has m number of observations and each of them is dependent on n number of independent variables for x then the Eq. 5.9 becomes,

$$\begin{bmatrix} dy_1 \\ dy_2 \\ \vdots \\ dy_m \end{bmatrix} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \cdots & \frac{\partial y_1}{\partial x_n} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} & & \frac{\partial y_2}{\partial x_n} \\ \vdots & & \ddots & \vdots \\ \frac{\partial y_m}{\partial x_1} & \frac{\partial y_m}{\partial x_2} & \cdots & \frac{\partial y_m}{\partial x_n} \end{bmatrix} \begin{bmatrix} dx_1 \\ dx_2 \\ \vdots \\ dx_n \end{bmatrix}$$

or,

$$dy = J_{xy} dx \qquad (5.10)$$

Eq. 5.10 is the general form of system error propagation where J_{xy} is called the Jacobian (Jacobian matrix) of the equation, and values of any particular measurement at an independent variable follow Gaussian distributions (e.g., Normal) due to the presence of the randomness of error

(represented as standard deviation or variance). Thus, the propagation of random error in the system follows the law of propagation of variance and covariance (POV), which can be expressed by the following equation 11. Note, detailed derivation of Eq. 5.11 from Taylor Series expansion can be found in Appendix A.

$$C_y = J_{xy}\, C_x\, J_{yx}^T \qquad (5.11)$$

Here, C_y is the covariance matrix of random output y, and C_x is the covariance matrix of random input x.

Now, to examine the total error in labor hours propagated through different nodes of the project network up to time t, Eq. 5.11 is applied in connection with Eq. 5.8.

Therefore, Jacobian (J_{T_A}) of, $LH_{Total,\ t}$, ($or\ f_t$) should be,

$$J_t = \left[\frac{\partial f_t}{\partial \beta_{0,1}} \quad \frac{\partial f_t}{\partial \beta_{1,1}} \quad \frac{\partial f_t}{\partial \beta_{2,1}} \quad \cdots \quad \frac{\partial f_t}{\partial \beta_{i,j}}\right] \qquad (5.12)$$

And considering all the coefficients are independent (not correlated to each other), the covariance matrix of the Eq. 5.8 is expanded,

$$C_t = \begin{bmatrix} var(\beta_{0,1}) & & & & \\ & var(\beta_{1,1}) & & & \\ & & var(\beta_{2,1}) & & \\ & & & \ddots & \\ & & & & var(\beta_{i,j}) \end{bmatrix} \qquad (5.13)$$

Now the total variance ($\sigma_t{}^2$) adding up all relevant activities completed up to time point t can be computed as follows (Eq. 5.14) from Eq. 5.11,

$$\sigma_t{}^2 =$$

$$\begin{bmatrix} \frac{\partial f_t}{\partial \beta_{0,1}} & \frac{\partial f_t}{\partial \beta_{1,1}} & \frac{\partial f_t}{\partial \beta_{2,1}} & \cdots & \frac{\partial f_t}{\partial \beta_{i,j}} \end{bmatrix} \begin{bmatrix} var(\beta_{0,1}) & & & & \\ & var(\beta_{1,1}) & & & \\ & & var(\beta_{2,1}) & & \\ & & & \ddots & \\ & & & & var(\beta_{i,j}) \end{bmatrix} \begin{bmatrix} \frac{\partial f_t}{\partial \beta_{0,1}} \\ \frac{\partial f_t}{\partial \beta_{1,1}} \\ \frac{\partial f_t}{\partial \beta_{2,1}} \\ \vdots \\ \frac{\partial f_t}{\partial \beta_{i,j}} \end{bmatrix} \quad (5.14)$$

5.3.2 Generating "S stripe" based on CPM

The *"S stripe"* generation methodology is established on top of the classical CPM. The first step of the process is to identify all the paths possible to complete the project. The duration of each activity can then be calculated using the mean of productivity rate fixed in activity-level estimating. CPM is accordingly applied to identify the critical path and determine the mean of total project duration. The error propagation model is applied to characterize the variability of the required labor hours at a particular time point in terms of standard deviation. The mean of the cumulative labor hours required against time elapsed is plotted, resulting in the S-curve first. Then, by setting a certain level of confidence, the S stripe is plotted.

Steps to compute this propagated uncertainty through the project network are summed up as follows:

Step 1: Develop project WBS and project AON network diagram representing all logical constraints.

Step 2: Define activity duration equation (Eq. 5.6) for all activities in AON.

Step 3: Apply CPM to schedule activities and identify critical path based on mean activity duration calculated from Eq. 5.7.

Step 4: Apply error propagation formula (Eq. 5.14) to quantify the propagated uncertainty (in the form of standard deviation of cumulative LH at particular time points at project level).

Step 5: Generate S Curve by drawing the "Time duration vs mean cumulative LH" graph for the project, plus S stripe by articulating lower and higher bounds for a given confidence interval based on standard deviations of cumulative LH at control points as derived from Step 4.

5.4 Example Case

To demonstrate the application of the proposed methodology, a steel bridge girder fabrication project is considered. In a typical bridge girder fabrication project, steel plates of different dimensions and grades are transformed into "I" section girders in a fabrication shop. The fabrication operation mainly consists of the following main work packages (WPs): (A1) preparing plates, (A2) flanges preparation, (A3) web preparation, (A4) stiffener preparation, (A5) assembling girder (by fitting and welding flanges to web), (A6) stiffener fitting and welding, (A7) final finishing work on the girder (e.g., studding, sandblasting, finishing). The length and width of the girder being fabricated are 20m and 0.6m, respectively. Work package definition for this steel girder preparation project is shown in Table 5.2.

Table 5.2 Work breakdown structure of the example project.

ID	Work Packages	Predecessors	Direct Labor Cost (LH)	Productivity variables (μ, σ)	Number of Workers (Leading Trades) $\boldsymbol{n_{Lead}}$
A1	Preparing plates	-	$\beta_{1,A1}$	$\beta_{1,A1} = (32, 2)$	2
A2	Flanges preparation	A1	$\beta_{1,A2}x_g$	$\beta_{1,A2} = (4.8, 0.25)$	2
A3	Web preparation	A1	$\beta_{1,A3}x_g$	$\beta_{1,A3} = (6, 0.5)$	2
A4	Stiffener preparation	A1	$\beta_{1,A4}x_g + \beta_{2,A4}y_g$	$\beta_{1,A4} = (4.5, 0.2)$ $\beta_{2,A4} = (6, 0.25)$	2
A5	Assembling girder	A2, A3	$\beta_{1,A5}x_g$	$\beta_{1,A5} = (8\ , 0.15)$	4
A6	Stiffener fitting and welding	A4, A5	$\beta_{1,A6}x_g + \beta_{1,A6}y_g$	$\beta_{1,A6} = (9.6, 0.2)$ $\beta_{1,A6} = (16, 0.4)$	4

A7	Final finishing	A6	$\beta_{1,A7} x_g$	$\beta_{1,A7} = (8, 0.5)$	2
• *Here,* x_g *= Length of the girder, and* y_g *= Width of the girder.* • *For this example,* $x_g = 20\,m$ *and* $y_g = 0.6\,m$					

5.4.1 Step 1

The logical interdependencies among the work packages as given in the Table 2 "Predecessors" are represented in the AON diagram in Fig. 5.2.

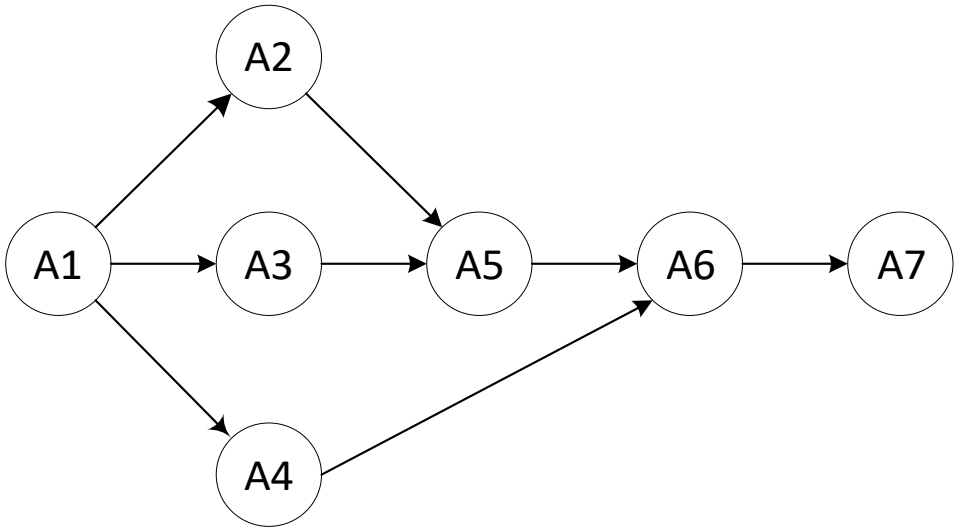


Figure 5.2: AON network diagram for the example girder preparation project.

5.4.2 Step 2

To prepare for CPM analysis, activity duration equation (Eq. 5.6) for all activities is given in Table 5.3.

Table 5.3 Activity duration determination for all work packages in case project.

Activity ID	Time equation, $\boldsymbol{T_j}$	Mean Duration (hr)
A1	$\frac{\beta_{1,A1}}{n_{A1}}$	16
A2	$\frac{\beta_{1,A2}x_g}{n_{A1}}$	48
A3	$\frac{\beta_{1,A3}x_g}{n_{A1}}$	60
A4	$\frac{\left(\beta_{1,A4}x_g + \beta_{2,A4}y_g\right)}{n_{A1}}$	46.8
A5	$\beta_{1,A5}x_g$	40
A6	$\frac{\left(\beta_{1,A6}x_g + \beta_{1,A6}y_g\right)}{n_{A1}}$	50.4
A7	$\frac{\beta_{1,A7}x_g}{n_{A1}}$	80

5.4.3 Step 3

CPM is applied to identify all the possible paths. Table 5.3 summarizes all the paths and related path lengths calculation. Duration of each path (path length) is calculated based on mean activity duration as given in Table 5.3 and summarized in Table 5.4.

Table 5.4 Duration of each path (path length) is calculated based on mean activity duration.

ID	Path	Total Duration	Critical Path?
P1	A1 – A2 – A5 – A6 – A7	16+48+40+50.4+80 = 234.4	No
P2	A1 – A3 – A5 – A6 – A7	16+60+40+50.4+80 = 246.4	Yes
P3	A1 – A4 – A6 – A7	16+46.8+50.4+80 = 193.2	No

5.4.4 Step 4

A bar chart project schedule resulting from CPM analysis is presented in Fig. 5.3.

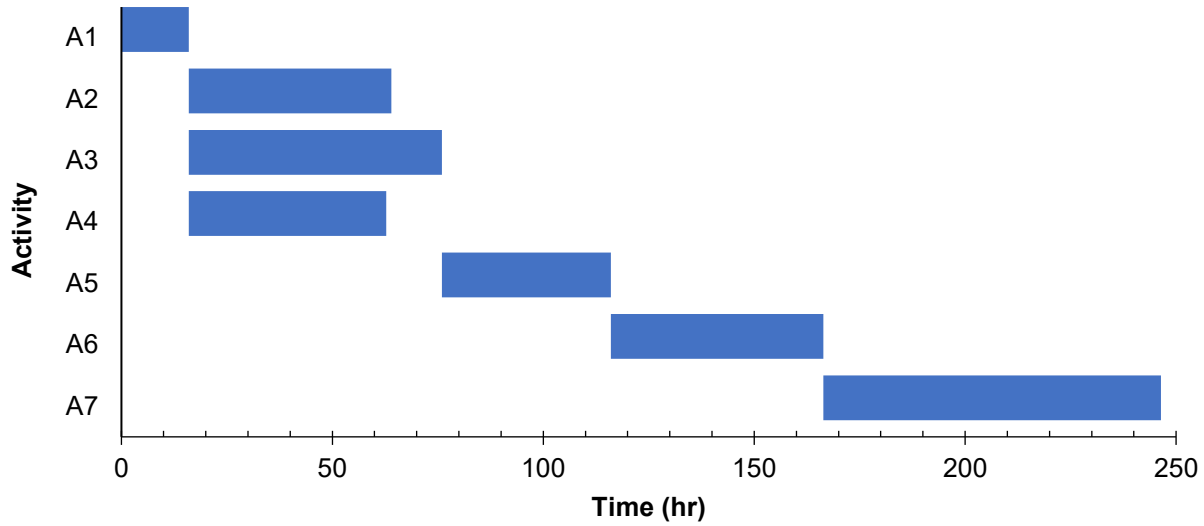


Figure 5.3: Project schedule for the example project case.

Eq. 5.14 is applied to quantify the propagated LH variance at different time points in project duration. For example, as we can see from Fig. 5.3, Activities A1, A2, A3, and A4 are 100% completed after 76 hr of the total project. Therefore, total LH budgeted for the project up to 76 hr is the sum of all LH required to complete A1, A2, A3, and A4, as determined by Eq. 5.8:

$$LH_{Total,\ t} = \sum_{j,k}^{N,t} \left(LH_{T,j} \times \frac{t_{j,k}}{T_j} \right)$$

$$= \sum_{j,k}^{4,76} \left(LH_{T,j} \times \frac{t_{j,k}}{T_j} \right)$$

$$= \left(LH_{T,A1} \times \frac{t_{A1,76}}{T_{A1}} \right) + \left(LH_{T,A1} \times \frac{t_{A1,76}}{T_{A1}} \right) + \left(LH_{T,A1} \times \frac{t_{A1,76}}{T_{A1}} \right) + \left(LH_{T,A1} \times \frac{t_{A1,76}}{T_{A1}} \right)$$

$$= (\beta_{1,A1} \times 1) + (\beta_{1,A2} x_g \times 1) + (\beta_{1,A3} x_g \times 1) + \left((\beta_{1,A4} x_g + \beta_{2,A4} y_g) \times 1 \right) = f_{76} \qquad (5.16)$$

$$= 32 + (4.8 \times 20) + (6 \times 20) + (4.5 \times 20 + 6 \times 0.6)$$

$$= 32 + 96 + 120 + 93.6 = 341.6 \text{ LH}$$

So, the mean of the total LH required to complete all jobs until 76 hr of the project is 341.6 LH.

Therefore, Jacobian (J_{76}) of, $LH_{Total,\ 76,}(\ or f_{76})$ should be (as per Eq. 5.12),

$$J_{76} = \left[\frac{\partial f_t}{\partial \beta_{1,A1}} \quad \frac{\partial f_t}{\partial \beta_{1,A2}} \quad \frac{\partial f_t}{\partial \beta_{1,A3}} \quad \frac{\partial f_t}{\partial \beta_{1,A4}} \quad \frac{\partial f_t}{\partial \beta_{2,A4}}\right] = [1 \quad 20 \quad 20 \quad 20 \quad 0.6]$$

And, the covariance matrix of f_{76} would be (as per Eq. 5.13),

$$C_{76} = \begin{bmatrix} var(\beta_{1,A1}) & & & & \\ & var(\beta_{1,A2}) & & & \\ & & var(\beta_{1,A3}) & & \\ & & & var(\beta_{1,A4}) & \\ & & & & var(\beta_{2,A4}) \end{bmatrix}$$

$$= \begin{bmatrix} 2^2 & & & & \\ & 0.25^2 & & & \\ & & 0.5^2 & & \\ & & & 0.2^2 & \\ & & & & 0.25^2 \end{bmatrix}$$

Now variance of the cumulative LH can be found using Eq. 5.14,

$$\sigma_{76}{}^2 = [1 \quad 20 \quad 20 \quad 20 \quad 0.6]\begin{bmatrix} 2^2 & & & & \\ & 0.25^2 & & & \\ & & 0.5^2 & & \\ & & & 0.2^2 & \\ & & & & 0.25^2 \end{bmatrix}\begin{bmatrix} 1 \\ 20 \\ 20 \\ 20 \\ 0.6 \end{bmatrix} = 145.02$$

So, the variance of total LH required at time point 76 hr of the project is 145.02. Variances for LH required at different times along the project duration are also determined in a similar fashion, which is summarized in Table 5.5.

Table 5.5 Variance, Standard Deviation and 95% Confidence Interval Derived for LH required at different times along the project duration.

Time point (hr)	Completed Activities	Mean - Cumulative LH	Variance - LH $(J_{T_{P2}} C_{T_{P2}} J_{T_{P2}}^T)$	STD - LH	95% Confidence Interval of Mean
16	A1	32	4	2.00	[28.1, 35.9]
76	A1, A2, A3, A4	341.6	145.02	12.04	[318.0, 365.2]
116	A1, A2, A3, A4, A5	485.6	154.02	12.41	[461.3, 509.9]

166.4	A1, A2, A3, A4, A5, A6	687.2	170.08	13.04	[661.6, 712.8]
246.4	A1, A2, A3, A4, A5, A6, A7	847.2	270.08	16.43	[815.0, 879.4]

5.4.5 Step 5

In Fig. 5.4, the S-curve ("Time duration vs Mean Cumulative LH" graph) is plotted; then, the lower and higher bounds of 95% confidence interval derived for LH required at different times are articulated to generate the S stripe as per Step 4.

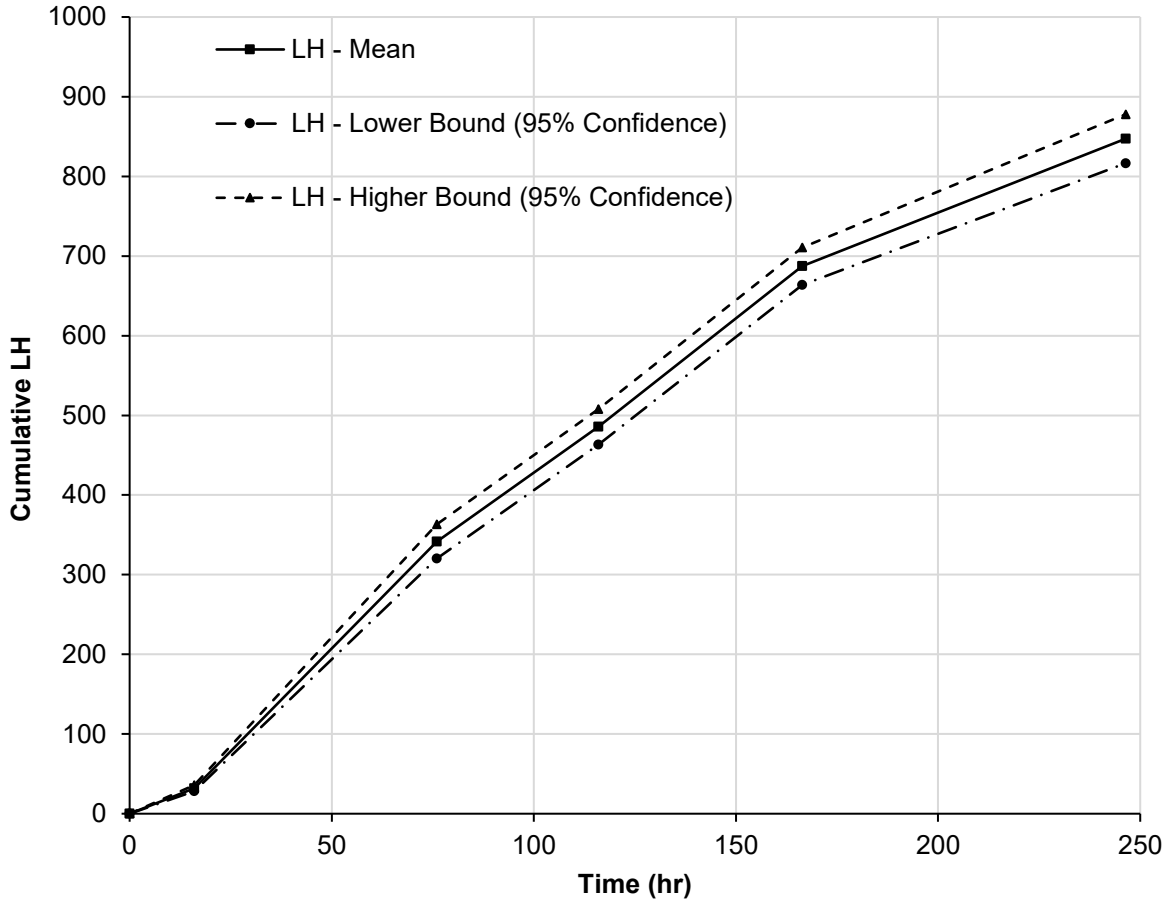


Figure 5.4: S-Curve and S stripe plotted for the cumulative LH over project duration.

5.5 Research Validation

To validate the application of the proposed methodology, the same project data, given in Table 5.1, was analyzed by applying Monte Carlo simulation, and the total project LH with time elapsed for the project was derived. The total LH required to complete the entire project along with 95 % confidence values (both upper and lower bound) are plotted for different number simulation runs in Fig. 5.5. Until 500 simulation runs, the cumulative LH required for the project fluctuates. It is

observed from Fig. 5.5 the mean and 95% confidence interval for the total LH budget on the case study project stabilize after the simulation run increases beyond 500.

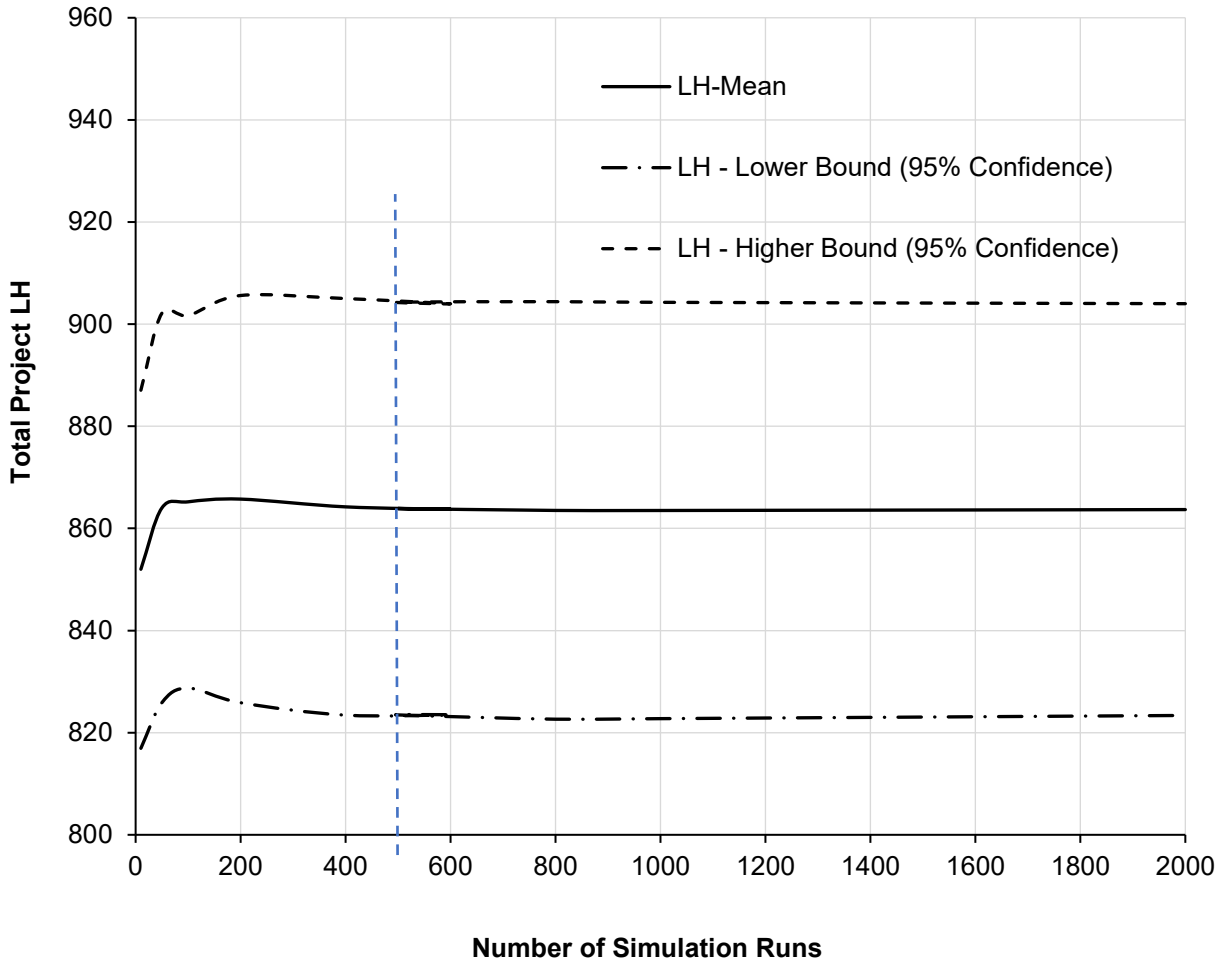


Figure 5.5: Values for total LH required for different number of simulation run.

Based on 1000 simulation runs, the average and variance for project LH budget are compared with the results from the error propagation model being proposed. When both sets of the results from two different methods are plotted side by side (Fig 5.6) no significant difference is observable visually up to 100 hours of the project duration. From 100 hours to 250 hours, the simulated labor

hours is marginally higher than obtained from the proposed analytical method. The correlation coefficient between the two sets of results is 0.9999 (Fig. 5.7). Results from two methods at selected control points of time are also contrasted in the Table 5.6.

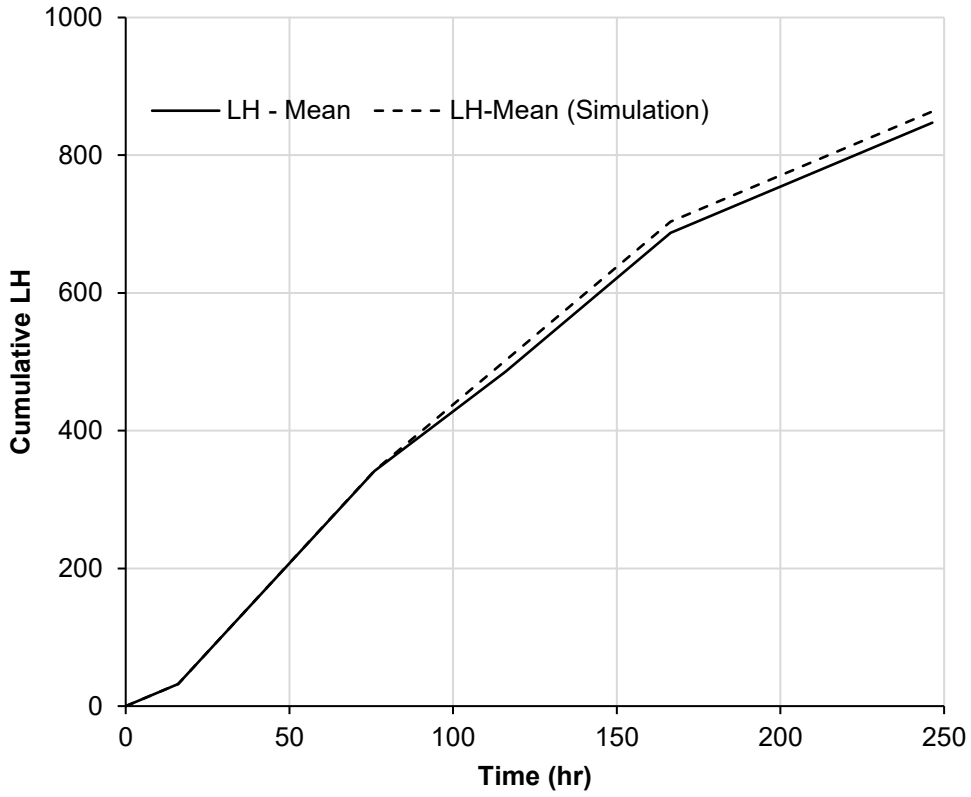


Figure 5.6: Average and variance for project LH budget resulting from 1000 simulation runs against results obtained by applying the error propagation model being proposed.

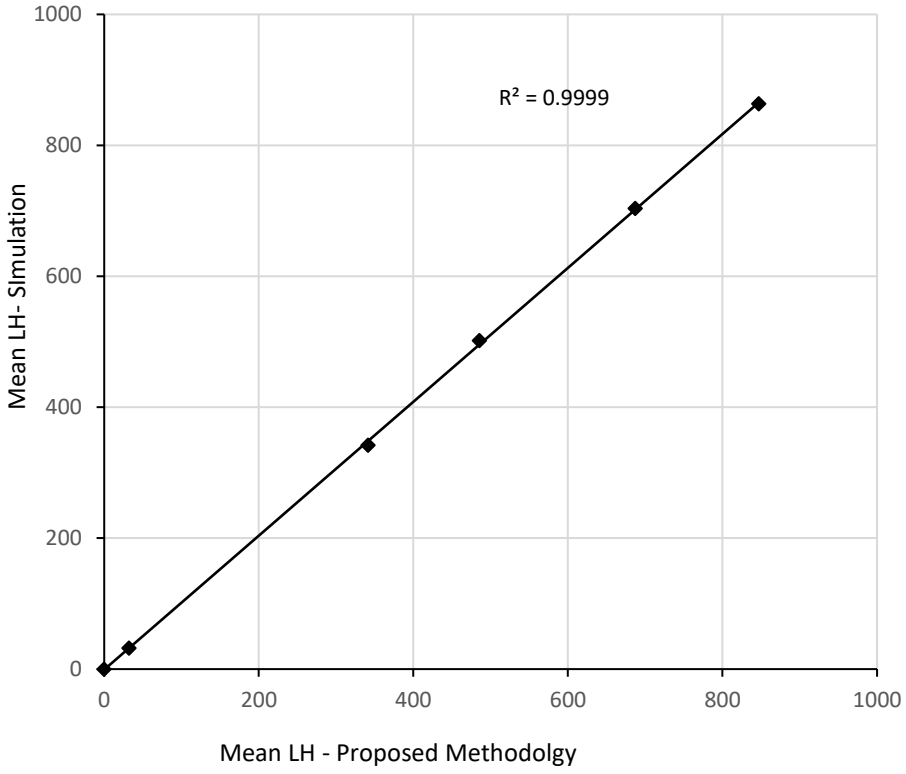


Figure 5.7: Mean of cumulating LH comparison: results obtained by applying proposed methodology vs simulated results.

Table 5.6 Result comparison between error propagation and simulation application.

Time point (hr)	Error Propagation Algorithm		Simulation Results (1000 runs)	
	Mean - Cumulative LH	STD - LH	Mean - Cumulative LH	STD - LH

16	32	2.00	32.2	2.5
76	341.6	12.04	342.1	15.3
116	485.6	12.41	501.9	16.2
166.4	687.2	13.04	703.6	16.7
246.4	847.2	16.43	863.5	20.8

5.6 Further observation

The case study has revealed two limitations of the simulation method: (1) setting the simulation model on a computing platform for an AON network diagram given project networks of practical size and complexity is not straightforward and requires programming skills; (2) simulation experiment design is time consuming; in particular, determining the sufficient number of simulation runs to obtain reliable sampling results entails trial and error specific to a case problem (note a minimum of 500 simulation runs is required in the above case study).

In contrast, the proposed analytical method will circumvent those barriers to obtain solutions and derive S stripe. The user only needs to plug the number in the derived formula in order to quantify the risks in the system output (such as labor cost budget) due to variations in input variables (such as labor productivity). For skilled trades working on highly specialized tasks (such as

welding, drilling), experienced tradespersons would have reliable know-how on the mean and variance of labor-hours required to complete a definitive scope of work. Obviously, the error propagation method potentially provides more efficient and practical decision support in a deterministic time window, not requiring the use of professional computer software or trial and error.

5.7 Summary and Conclusions

This research looks into current practices of planning, estimating, and budgeting prefabrication projects in industrial construction. Concurrently, it has identified the room for improvement in regards to the development of labor cost budget subject to variations inherent in labor productivity in the prefabrication processes involving skilled trades (such as welders, boilermakers, ironworkers, pipefitters). In particular, the paper proposes the analytical model based on error propagation theory for propagating uncertainties in labor productivity on the work package level to the derived labor cost budget on the project budget level. The resulting standard deviation of the cumulative labor hours at a particular time point of the project is further used to establish a confidence interval around the average value. Thus, analogous to plotting the S curve, the lower bound and upper bound of the interval at control points along the project duration can be articulated to form the S stripe, which visually portrays the risk in labor cost budgeting due to variations inherent in labor productivity. The proposed methodology has been verified by applying Monte Carlo simulation to the same project data in the case study.

For skilled trades working on highly specialized tasks (such as welding, drilling), experienced tradespersons would have reliable know-how on the mean and variance of labor-hours required to

complete a definitive scope of work. In practical application, the user only needs to plug the number in the derived formula in order to quantify the risks in the system output (such as labor cost budget) due to variations in input variables (such as labor productivity). Thus, the error propagation method potentially provides more efficient and practical decision support in a deterministic time window, not requiring time-consuming simulation experiments or using special computer software.

Chapter 6

Conclusion

6.1 Research Summary

This concluding chapter serves as a comprehensive summary of the findings and achievements of the research presented in this thesis. The key points discussed throughout the thesis are carefully highlighted, emphasizing the significant contributions made to both academic knowledge and practical applications. Additionally, the limitations that require further investigation are outlined, thereby opening up promising avenues for future research.

The literature review suggests that previous research on productivity modeling mainly focused on improving prediction accuracy without adequately considering the risks of relying on a single prediction value or the precision of a model's prediction. Validation and input selection were often based on comparing the model's predicted output with the target output (actual data) without considering the variance of the predicted output or the impact of individual input parameters on variance. This research addresses the identified limitations of labor productivity prediction models and provides analytical solutions for variance analysis of regression-based productivity prediction models for prefabrication construction. Particularly, this research combines error propagation theory with MLR modeling to determine the variance in estimating labor productivity and budgeting labor cost for prefabricated products. The attempt is made to create a metric based on

the resulting variance analysis (the ratio of standard deviation to mean) to objectively evaluate the precision of the MLR model. Establishing the MLR-type model is essential to examine how variability in productivity information at the work package level affects the entire project level through the project schedule and how uncertainty in project-level labor hours accumulates over the course of the project duration. Understanding the limitations of the MLR model in representing the nonlinear input-output relationship, this research further investigates a nonlinear regression model, namely the model tree algorithm. Coupling the generalized variance analysis model proposed for the MLR model, this research enhances the established model tree algorithm and enables it to make point value prediction and associated variance. The proposed enhanced model tree is adept in generalizing complex nonlinear input-output relationships by learning from data and providing interpretable rules for inferring variance estimates of the prediction.

Chapter 2 of this thesis critically reviews established methods for variance analysis on MLR models and proposes an analytical approach to quantify the variance of the predicted output. The proposed method integrates error propagation theory with MLR modeling, allowing for the assessment of precision in cost-estimating applications. The research presented in Chapter 2 emphasizes the significance of precision in MLR modeling and the impact of individual input parameters on the variance of the prediction. It advances regression modeling methods in general with respect to MLR variance analysis. By formalizing a cost-estimating model based on the product fabrication complexity factors using the proposed variance analysis technique for the MLR model, the study accounts for the variability of labor hour estimates for different product types due to their complex features. The complexity factors are represented with a mean and standard deviation, reflecting average benchmarks and variability in labor hour prediction. MLR variance

analysis technique is instrumental in quantifying the overall labor hour variability for different work packages arising from the product variability of product features.

The proposed methodology eliminates the need for Monte Carlo simulation techniques in practical applications, offering a more efficient approach to gauge the precision of the MLR model. The application of the MLR model with variance analysis technique is illustrated with an application case in estimating labor costs for precast solid wall panels. In addition to that, the research also demonstrates practical applicability through a process piping spool fabrication case, validating the proposed method in estimating labor hours for various welding work items.

Overall, this research advances the variance estimation technique of regression-based prediction model namely MLR in construction productivity modeling. It opens up promising directions for further computing research in civil engineering, encouraging the application of more sophisticated regression models and higher-order Taylor expansions in the future. However, the research acknowledges that MLR may not be sufficient for highly complicated cost estimating problems characterized with nonlinear relationships between input and output variables. In such cases, more sophisticated data mining techniques or nonlinear estimating models are advocated to complement the MLR model.

In the realm of practical productivity modeling for fabrications, where product design features are correlated with labor productivity, it is commonly observed that nonlinear input-output relationships are prevalent. This characteristic could hamper the widespread adoption of the Multiple Linear Regression (MLR) model in productivity modeling endeavors. This challenge has been tackled in Chapter 3. Research presented in Chapter 3 focuses on enhancing the model tree

algorithm, an a well-established commonly applied AI technique, to address the variance in predicted productivity arising from the random sampling of training data. With the increasing availability of large datasets containing design features and labor cost data in the construction industry, predictive modeling using regressions or AI has become more feasible. However, the use of different datasets sampled by various modelers or at different time periods may lead to diverse model parameters and, consequently, a variance in predicted productivity. This raises the need to determine the precision of productivity derived by of a prediction model in addition to its accuracy. The study decomposes the productivity problem into branches using the model tree's nonlinear classifier algorithm. Using the method outlined in Chapter 2, the model tree is enhanced to predict the point-value output and its associated variance in each branch. A case study on structural steel fabrication productivity with collected data from the industry demonstrates the application of the proposed methodology, with the enhanced model tree outperforming MLR in prediction accuracy and being preferred over ANN due to its variance prediction and model explainability. The methodology presented in the chapter is data-driven and analytical, which derives the mean and variance of each coefficient associated with productivity components primarily due to a random sampling of training data. In cases where limited data is available, alternative means, such as expert estimation based on the knowledge of experienced practitioners, can be employed. In short, the enhanced model tree shows high potential as an explainable AI or nonlinear regression technique with broader applicability in various domains beyond construction productivity.

Chapter 3 concludes by highlighting the fact that the enhanced model tree algorithm has a broader application potential in tackling many civil engineering problems. Chapter 4 of this thesis acts on this notion and presents a case study to develop a strength prediction model for high-performance

concrete. It is well-established that the relationships among input variables that impact the concrete strength are highly nonlinear. The enhanced model tree algorithm is found to be sufficient in terms of (1) predicting accuracy (the Pearson's coefficient of correlation of the predicted and target concrete compressive strength is 0.88), (2) explaining decision logic with the decision tree structure, and (3) producing variance estimate of the prediction. In summary, as data availability and complexity continue to grow in civil engineering applications, the proposed methodology offers a practical and powerful solution for accurate and transparent predictions, and further advances the field of engineering applications using machine learning techniques, and explainable artificial intelligence (XAI).

Chapter 5 of this thesis delves into labor cost budgeting in the context of industrial construction, wherein large-scale fabrication operations are employed to produce modules and structural components offsite, facilitating rapid installation in the field. An analytical methodology is developed to assess the impact of productivity variability at the work package level on labor cost budgeting for the total fabrication project. The proposed error propagation model calculates the variance of cumulative labor hours at specific time points during the project, establishing a confidence interval around the average value. Termed the "S stripe," this visual tool depicts the risk associated with labor cost budgeting due to labor productivity variabilities. Such a tool is instrumental in many applications, from project bidding to project control. The methodology is applied and rigorously verified in a case study of a steel fabrication project, demonstrating a close correlation with results obtained from MC simulation. The data-driven approach derived from the earlier chapters is applicable to determine the variability in project-level labor hour estimate due to productivity variability on each work package. However, when data evidence is limited, skilled

trades engaged in highly specialized tasks, such as welding and drilling, can be relied upon to confidently provide reliable information on the mean and variance of labor hours required to complete specific scopes of work. This provides versatility in the research application and strengthens its impact and potential for broader implementation in the construction industry.

6.2 Research Contributions

6.2.1 Academic Contributions

The present study examines the current practices of planning, estimating, and budgeting for prefabrication projects in the context of industrial construction. As a result, the room for improvement is identified in regard to the development of labor cost budget subject to variations inherent in labor productivity in the prefabrication processes involving skilled trades (such as welders, boilermakers, ironworkers, pipefitters). This research study makes following academic contributions to the existing knowledge as follows.

Academic contribution 1: This research proposes a variance analysis technique for multiple linear regression (MLR)-based prediction models derived analytically based on the error propagation theory in terms of:

- Developing a systematic approach to represent the variability of the model's input parameters through statistical measures like mean and variance.
- Formalizing a method to calculate the variance of the predicted output (e.g., productivity of a particular product) from an MLR model by considering the known variances of its parameters.

- Defining a metric to evaluate the sufficiency of the MLR model.

Academic contribution 2: In this research, an analytical method is proposed to quantify the influence of the variance of each input parameter upon the variance of the predicted output where input variables are related to predicted output using MLR,

- To identify the critical input parameters that contribute to the final output variance to a large degree.

Academic contribution 3: A labor productivity model that connects product engineering design features as inputs to the targeted productivity as the output has been proposed, allowing for:

- Determining the contribution ratio of product features to the final productivity definition.
- Estimating the variance of the productivity prediction.
- Identifying the crucial input product features based on their respective contributions to the total output variance.

Academic contribution 4: This research enhances the established AI technique called Model Tree to quantity the variance for the predicted output due to random sampling of training data through:

- Integrating the proposed variance analysis technique for MLR equations with the commonly applied model tree algorithm (M5P Tree algorithm).
- Proposing the use of Coefficient of variation as a metric to evaluate the application fitness of the model tree model.

Academic contribution 5: Realizing the need of accounting for the nonlinear relationships present between the product design features and labor productivity, this research proposes a framework for productivity modeling using the enhanced model tree method through,

- Applying the non-linear classifier algorithm from established model tree algorithm to decompose nonlinear productivity problem into branches (data classes) in order to make it suitable for MLR applications in corelating input features with output.
- Using the enhanced model tree method in such a way that the MLR equation on each tree branch can predict the point-value output as well as the associated variance.
- Taking full use of the Coefficient of variation of enhanced model tree algorithm to test the applicability of the prepared model.

Academic contribution 6: This research introduced a methodology to address the variance in the total labor hour budget by consolidating the variance estimates of labor productivity at the work package level through:

- Determining the lower and upper boundaries for cumulative labor hours budgeted at various control points throughout the project lifecycle, aiming for a specific confidence level.
- Introducing a novel project control tool named the S stripe, which visually represents the estimated variance of the total labor hours estimate at specific time points of the project. The S stripe establishes a confidence interval around the average value, akin to plotting the S curve. It articulates the lower and upper bounds of the interval at control points

along the project duration, effectively illustrating the risk in labor cost budgeting due to inherent variations in labor productivity.

6.2.2 Industry Contributions

This research proposed methods to prepare productivity models for fabrication projects to estimate productivity for fabrication products with associated variance. In practical application, the user only needs to plug the number in the derived formula in order to quantify the variance associated with the labor cost budget due to variations in input variables (product feature attributes). Thus, the error propagation-based regression methods provide more efficient and practical decision support in a deterministic time window, not requiring time-consuming simulation experiments or using special computer software.

The proposed enhanced model tree represents a significant addition to the data mining and AI toolbox in construction, regression, and AI-based prediction modeling domains. Traditionally, research efforts have been heavily geared towards augmenting prediction accuracy, often neglecting the equally crucial aspect of model precision. With the introduction of the enhanced model tree, industry practitioners can now benefit from a powerful tool that enhances prediction accuracy and addresses the variance in predicted output. Such an analytical capability enables a more comprehensive understanding of the model's precision and leads to more informed and effective decision-making processes, which are vital for gaining trust from practitioners in data-driven predictive models.

It is important to highlight that the enhanced model tree algorithms, as presented in this study, rely entirely on data-driven and analytical methods, given the availability of sufficient productivity

data. This approach allows for the derivation of the mean and variance of each coefficient associated with the productivity components by employing random sampling of the training data. However, when limited data are available, and random sampling is not feasible, domain experts with extensive experience in the work process can provide reliable inputs of the mean and variance on each productivity component, such as the most likely value, minimum, and maximum. In such situations, the mean is approximated as a weighted average of the three points, while the standard deviation is estimated based on the range. For instance, one common approach is assuming normality and calculating one-sixth of the range to obtain the standard deviation. By incorporating data-driven methods, expert know-how, and experiences, the proposed methodology affords a flexible and practical approach to handling various data availability scenarios in productivity modeling. This hybrid approach facilitates accurate and reliable productivity predictions, enabling better decision-making and resource allocation in various industrial applications.

The applicability of the enhanced Model Tree model extends far beyond productivity models in construction research. Its potential as an explainable AI or non-linear regression technique spans across various application domains. Consequently, professionals across different industries can harness the enhanced model tree to gain valuable insights into their specific processes and operations. By considering both prediction accuracy and model precision, industry professionals can elevate their analytical decision support to new heights.

Furthermore, with the ability to estimate the variance in labor cost budgeting due to inherent variations in labor productivity by consolidating variance estimates of labor productivity at the work package level, industry practitioners can better understand the risks associated with labor

cost budgeting in planning construction projects. Determining lower and upper boundaries for cumulative labor hours budgeted enables project managers to make more informed decisions during the project planning stages (preparing for the bid and constructing a realistic project schedule). The novel project control tool, known as the S stripe, visually represents the estimated variance of the total labor hours estimated at specific time points of the project and offers clarity and transparency for risk analysis and communication. By establishing a confidence interval around the average value, akin to plotting the S curve, the S stripe offers a clear and intuitive understanding of the inherent variations in labor productivity and their impact on labor cost budgeting. With this methodology in place, the industry gains a powerful means of assessing and managing risks associated with labor cost budgeting, especially in large-scale prefabrication operations and other labor-intensive construction projects. Industry professionals can rely on the S stripe to make more data-driven and well-informed decisions, ensuring that labor cost budgets align with project requirements and expectations to ensure informed resource allocations and better project control.

The method's versatility and applicability make it a valuable asset in the field of industrial fabrication and construction, providing an innovative approach to address the challenges posed by fluctuating and uncertain labor productivity. Overall, the methodology sets a new standard for labor cost estimating and budgeting in the industry, potentially advancing project control practices and contributing to improved project outcomes.

The analytical approach resulting from this research provides an alternative solution to the commonly applied Monte Carlo simulation technique within industrial contexts for performing uncertainty analysis on complicated numerical models. This alternative approach eliminates the

need for advanced computing resources in the application stage, making it particularly relevant to the real world setting of estimating productivity for construction components or activities. Moreover, it aligns seamlessly with the accumulated experience and expertise of estimators and trade workers by offering a direct and pragmatic method. Estimators can readily apply it by inputting values into analytical equations preprogrammed in a spreadsheet. As a result, this practical approach does not entail configuring complex computing platforms and applying simulation software in analyzing each estimating scenario. It is widely acknowledged that Monte Carlo simulation demands user's knowledge and training in computer programming and statistics, especially when simulation is applied to intricate systems and updating simulation setup is frequently justified. Applying Monte Carlo simulations in such complicated circumstances can be associated with sluggish performance and high costs. It is reasonable to anticipate that the proposed approach may substitute for Monte Carlo simulation and enhance both accessibility and acceptability of risk analysis in practical engineering and management applications, not confined to productivity estimating.

6.3. Research Limitations

The proposed variance analysis method relies on MLR as the base model; however, MLR may not address complicated cost-estimating problems where highly nonlinear relationships are present between input and output variables. Under such circumstances, the MLR model can be complemented with more sophisticated data mining techniques (e.g., using various combinations of input variable sets, applying classification techniques to find different ranges of data clusters, etc.). Alternatively, nonlinear analytical processing models can be adopted (e.g., scaling the data

set using log functions, applying nonlinear regression techniques like artificial neural networks, etc.).

The formulation of the variance estimation model in Chapter 2 is based on the assumption that the input variables used in the model are independent of one another. However, it is imperative to substantiate this assumption with robust statistical evidence. Should a significant correlation be present between the chosen input variables, it has the potential to compromise the effectiveness of the current model, thereby magnifying the fundamental limitation of the proposed research. In reality, the modeler needs to revisit the problem definition by consulting with domain experts, thus removing certain input variables or consolidating correlated input factors as one. In the long run, a reconsideration and redevelopment of the model formulation will become an appealing research opportunity to factor in the covariance between the input variables in the applied error propagation model.

Driven by application needs, this research has proposed an enhanced version of the model tree algorithm to overcome the limitations of the MLR models. Depending on the data quality, it can be found inadequate upon checking the Coefficient of variation formalized in this research. The shortfall can be due to not having sufficient records for each input feature or selected input features inadequately accounting for the productivity in the problem domain. On such occasions, models would require additional data collected on new features or must couple with more advanced data modeling or decision-making techniques (e.g., fuzzy inference system, agent-based modeling, etc.), applying more sophisticated regression models (regression with nonlinear functions) and taking higher-order Taylor expansion in applying the error propagation theory.

The proposed methodology is founded on the premise that the variability in input factors contributes to productivity variability and can be quantified through the variance estimate of the associated input parameter in the regression formulation. However, if the data distribution exhibits significant skewness to such an extent the error propagation theory is no longer applicable, the underlying Gaussian or normality assumption may become invalid, leading to an unacceptable model. The potential consequences of such an eventuality warrant further investigation.

6.4. Recommendations for Future Work

This section presents potential avenues for future research that arise from the findings and contributions of this study.

The success of applying the proposed research relies on identifying the appropriate input product features for productivity modeling. Selecting the right amount of relevant input features for preparing the productivity model is essential to achieving reliable outcomes. Many feature selection techniques can be employed in preparing the productivity model in the future to facilitate the identification of the input factors, which would complement this research.

The preparation of the variance analysis technique encompasses a recursive procedure that entails the sampling of a subset from the existing dataset. This process is instrumental in deriving the mean and variance of the coefficients in the Multiple Linear Regression (MLR) model, which are subsequently used in the construction of the variance estimation model. One established approach for achieving random sampling with replacement is the Bootstrap method, which offers a viable alternative to the arbitrary selection of the number of instances within the dataset during random sampling (Efron and Tibshirani 1994). Nonetheless, it is noteworthy that current research

endeavors have not yet delved into a critical comparison between these two sampling methods. This presents an avenue for future research endeavors to investigate the effect of various random sampling techniques on the proposed MLR variance analysis.

Even the enhanced model tree algorithm might reach its limits in certain scenarios. This would suggest that the model might necessitate incorporating additional data related to novel features or integration with more advanced data modeling or decision-making techniques (such as fuzzy inference systems, agent-based modeling, etc.). Furthermore, employing more sophisticated regression models, like regression with nonlinear functions, and incorporating higher-order Taylor expansions in applying error propagation theory could prove beneficial. Beyond the domain of construction engineering and management, these potential extensions offer promising avenues for future research in the field of civil engineering computing.

Accessing data directly from the BIM model, interpreting it with appropriate construction engineering and management insights, and feeding it into a productivity prediction model would be conducive to preparing cost estimate and budget in planning construction projects. When actual labor hour data is collected during construction, it can be fed back into the productivity model for further calibration and continual updating. Automating this approach would render the productivity modeling and planning framework self-sustaining, and further research in this direction can be pursued in the future.

With the increase in data availability and complexity within the civil engineering domain, the introduced methodology provides a novel solution for precise and transparent predictive analytics. Not limited to labor productivity estimating, this algorithm can be tested in many applications in

civil engineering (e.g., finding optimal solutions for shear-strength concrete masonry walls and establishing the factor of safety for model applications).

As a future research direction, it is essential to investigate the impact of skewed data distributions on the proposed methodology. This will help determine the validity of the assumption underlying the error propagation theory that input factors' variability contributes to productivity variability through variance estimates in regression. Exploring potential remedies or adjustments for dealing with skewed data distributions will enhance the applicability and robustness of the methodology in real-world applications.

In this research, the capability of the nonlinear regression technique, namely the model tree, has been enhanced by integrating the MLR based variance analysis technique. It will be worthwhile to investigate the use of the similar variance analysis framework to predict the output variance of an artificial neural network (ANN) model in future follow-up research.

References

Abdel-Hamid, M., & Mohamed-Abdelhaleem, H. (2022). Impact of poor labor productivity on construction project cost. International Journal of Construction Management, 22(12), pp. 2356–2363. https://doi.org/10.1080/15623599.2020.1788757

AbouRizk, S., Knowles, P., & Hermann, U. R. (2001). Estimating Labor Production Rates for Industrial Construction Activities. Journal of Construction Engineering and Management, 127(6), pp. 502–511. https://doi.org/10.1061/(asce)0733-9364(2001)127:6(502)

Ahuja, H. N., Dozzi, S. P., and AbouRizk S. M. (1994). Project Management: Techniques in Planning and Controlling Construction Projects. John Wiley and Sons, Inc. New York, 2nd Ed. ISBN: 978-0-471-59168-9

Alexander, M. (2020). Probability and Statistics for Data Sciences: Math + R + Data. J. of Qual. Tech., 52(4), 428–430.

Alkharusi, H. (2012). Categorical variables in regression analysis: A comparison of dummy and effect coding. International Journal of Education, 4(2), pp. 202 - 2010. https://doi.org/10.5296/ije.v4i2.1962

Amiri-Simkooei, A. R., Zangeneh-Nejad, F., and Asgari, J. (2016). On the covariance matrix of weighted total least-squares estimates. Journal of Surveying Engineering, 142(3), https://doi.org/10.1061/(ASCE)SU.1943-5428.0000153,04015014.

Ammar, T., Abdel-Monem, M., & El-Dash, K. (2023). Appropriate budget contingency determination for construction projects: State-of-the-art. Alexandria Engineering Journal, 78, pp. 88-103. https://doi.org/10.1016/j.aej.2023.07.035

Assaad, R. H., El-adaway, I. H., Hastak, M., & LaScola Needy, K. (2023). Key Factors Affecting Labor Productivity in Offsite Construction Projects. Journal of Construction Engineering and Management, 149(1), 04022158. https://doi.org/10.1061/jcemd4.coeng-12654

Bai, S., Li, M., Kong, R., Han, S., Li, H., & Qin, L. (2019). Data mining approach to construction productivity prediction for cutter suction dredgers. Automation in Construction. 105, 102833 https://doi.org/10.1016/j.autcon.2019.102833

Barbosa, F., Woetzel, H. J., Mischke, S. J., Ribeirinho, Z. M., Sridhar, L. M., Parsons, S. M., Bertram, P. N., & Brown, L. S. (2017). Reinventing Construction: A Route to Higher Productivity. Mckinsey Global Institute. McKinsey & Company. Retrieved from: https://www.mckinsey.com/capabilities/operations/our-insights/reinventing-construction-through-a-productivity-revolution (accessed on 02 February 2023)

Barraza, G. A. (2011). Probabilistic Estimation and Allocation of Project Time Contingency. Journal of Construction Engineering and Management, 137(4), pp. 259–265. https://doi.org/10.1061/(asce)co.1943-7862.0000280

Barrett, B. E., & Gray, J. B. (1994). A computational framework for variable selection in multivariate regression. Statistics and Computing, 4(3), 203–212. https://doi.org/10.1007/BF00142572

Barrie, D. S., and Paulson, B. C. (2001). Professional construction management: including CM, design-construct, and general contracting. McGraw-Hill series in construction engineering and project management. 3rd ed., McGraw-Hill, New York. ISBN 10: 0072551720, 9780072551723

Belle, V., & Papantonis, I. (2021). Principles and Practice of Explainable Machine Learning. Frontiers in Big Data. Frontiers Media S.A.

Blismas, N., Pasquire, C., & Gibb, A. (2006). Benefit evaluation for offsite production in construction. Construction Management and Economics, 24(2), 121–130. https://doi.org/10.1080/01446190500184444

Borjeghaleh, R. M., & Sardroud, J. M. (2016). Approaching Industrialization of Buildings and Integrated Construction Using Building Information Modeling. In Procedia Engineering (Vol. 164, pp. 534–541). Elsevier Ltd. https://doi.org/10.1016/j.proeng.2016.11.655

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (2017). Classification and regression trees. CRC Press. New York. https://doi.org/10.1201/9781315139470

Buntine, W. (1992). Learning classification trees. Statistics and Computing, 2(2), pp. 63–73. https://doi.org/10.1007/BF01889584

Cantoni, M., Weyer, E., Li, Y., Ooi, S. K., Mareels, I., & Ryan, M. (2007). Control of large-scale irrigation networks. Proceedings of the IEEE, 95(1), 75–91. https://doi.org/10.1109/JPROC.2006.887289

Cao, Y., Ashuri, B., & Baek, M. (2018). Prediction of Unit Price Bids of Resurfacing Highway Projects through Ensemble Machine Learning. Journal of Computing in Civil Engineering, 32(5), 04018043.

Caruana, R., & Freitag, D. (1994). Greedy attribute selection. In Proceedings of the Eleventh International Conference, Rutgers University, New Brunswick, NJ. (pp. 28-36). https://doi.org/10.1016/B978-1-55860-335-6.50012-X

Chan, S. L., & Park, M. (2005). Project cost estimation using principal component regression. Construction Management and Economics, 23(3), 295–304. https://doi.org/10.1080/01446190500039812

Chan, W. H., & Lu, M. (2008). Materials handling system simulation in precast viaduct construction: Modeling, analysis, and implementation. Journal of Construction Engineering and Management, 134(4), 300–310. https://doi.org/10.1061/(ASCE)0733-9364(2008)134:4(300)

Chang, Y.J., & Kim, H.-S. (2011). Tree-Structured Nonlinear Regression. Korean Journal of Applied Statistics, 24(5), pp. 759–768. https://doi.org/10.5351/kjas.2011.24.5.759

Cho, Y.-K., Haas, C. T., Sreenivasan, S. V., & Liapi, K. (2004). Position Error Modeling for Automated Construction Manipulators. Journal of Construction Engineering and Management, 130(1), 50–58. https://doi.org/10.1061/(asce)0733-9364(2004)130:1(50)

Choi, K., Kim, Y. H., Bae, J., & Lee, H. W. (2015). Determining Future Maintenance Costs of Low-Volume Highway Rehabilitation Projects for Incorporation into Life-Cycle Cost Analysis.

Journal of Computing in Civil Engineering, 30(4), 04015055. https://doi.org/10.1061/(asce)cp.1943-5487.0000533

Chou, J.-S., Chiu, C.-K., Farfoura, M., & Al-Taharwa, I. (2011). Optimizing the Prediction Accuracy of Concrete Compressive Strength Based on a Comparison of Data-Mining Techniques. Journal of Computing in Civil Engineering, 25(3), 242–253. https://doi.org/10.1061/(asce)cp.1943-5487.0000088

Christian, J., & Hachey, D. (1995). Effects of Delay Times on Production Rates in Construction. Journal of Construction Engineering and Management, 121(1), pp. 20–26. https://doi.org/10.1061/(asce)0733-9364(1995)121:1(20)

Creedy, G. D., Skitmore, M., & Wong, J. K. W. (2010). Evaluation of Risk Factors Leading to Cost Overrun in Delivery of Highway Construction Projects. Journal of Construction Engineering and Management, 136(5), 528–537. https://doi.org/10.1061/(asce)co.1943-7862.0000160

Deepa, C., SathiyaKumari, K., & Sudha, V. P. (2010). Prediction of the compressive strength of high performance concrete mix using tree based modeling. International Journal of Computer Applications, 6(5), pp. 18-24. Retrieved from: https://serkansubasi.net/Atiflar/ATIF%2027.pdf (Accessed on: 12 April 2023)

Desai, V. S., & Joshi, S. (2010). Application of decision tree technique to analyze construction project data. In Information Systems, Technology and Management: 4th International

Conference, ICISTM 2010, Bangkok, Thailand, (pp. 304-313). https://doi.org/10.1007/978-3-642-12035-0_30

Dietterich, T. G. (2002). Ensemble learning. The handbook of brain theory and neural networks, 2(1), 110-125. Retrieved from: https://courses.cs.washington.edu/courses/cse446/12wi/tgd-ensembles.pdf (Accessed on: 16 September 23).

Dixit, S., Mandal, S. N., Thanikal, J. V., & Saurabh, K. (2019). Evolution of studies in construction productivity: A systematic literature review (2006–2017). Ain Shams Engineering Journal, 10(3), 555–564. https://doi.org/10.1016/j.asej.2018.10.010

Doyle, A. and Hughes, W. (2000) The influence of project complexity on estimating accuracy. In: 16th Annual ARCOM Conference, 6-8 Sep 2000, Glasgow Caledonian University, 623-634. Available at http://centaur.reading.ac.uk/4295/

Dozzi, S.P. and AbouRizk, S. M. (1993). Productivity in Construction. Institute for Research in Construction, National Research Council, Ottawa, Canada. 54 pages. http://web.mit.edu/parmstr/Public/NRCan/nrcc37001.pdf (Accessed on 30 December 2022)

Durdyev, S., Ismail, S., & Kandymov, N. (2018). Structural Equation Model of the Factors Affecting Construction Labor Productivity. Journal of Construction Engineering and Management, 144(4). 04018007. https://doi.org/10.1061/(asce)co.1943-7862.0001452

Ebrahimi, S., Kazerooni, M., Sumati, V., & Fayek, A. R. (2022). Predictive model for construction labour productivity using hybrid feature selection and principal component analysis. Canadian Journal of Civil Engineering, 49(8), pp. 1366–1378. https://doi.org/10.1139/cjce-2021-0248

Edwards, D. J., & Holt, G. D. (2000). ESTIVATE: A model for calculating excavator productivity and output costs. Engineering, Construction and Architectural Management, 7(1), pp. 52–62. https://doi.org/10.1108/eb021132

Efatmaneshnik, M., & Ryan, M. J. (2019). On the Definitions of Sufficiency and Elegance in Systems Design. IEEE Systems Journal, 13(3), 2077–2088. https://doi.org/10.1109/JSYST.2018.2875152

Efron, B., & Tibshirani, R. J. (1994). An Introduction to the Bootstrap. An Introduction to the Bootstrap. Chapman and Hall/CRC. https://doi.org/10.1201/9780429246593

El-Abbasy, M. S., Senouci, A., Zayed, T., Mirahadi, F., & Parvizsedghy, L. (2014). Condition Prediction Models for Oil and Gas Pipelines Using Regression Analysis. Journal of Construction Engineering and Management, 140(6), 04014013. https://doi.org/10.1061/(asce)co.1943-7862.0000838

El-Gohary, K. M., Aziz, R. F., & Abdel-Khalek, H. A. (2017). Engineering Approach Using ANN to Improve and Predict Construction Labor Productivity under Different Influences. Journal of Construction Engineering and Management, 143(8), pp. 1–10. https://doi.org/10.1061/(asce)co.1943-7862.0001340

Elhag, T. M. S., & Wang, Y.-M. (2007). Risk Assessment for Bridge Maintenance Projects: Neural Networks versus Regression Techniques. Journal of Computing in Civil Engineering, 21(6), 402–409.

Ellis, R. D., & Lee, S. (2006). Measuring Project Level Productivity on Transportation Projects. Journal of Construction Engineering and Management, 132(3), 314–320. https://doi.org/10.1061/(asce)0733-9364(2006)132:3(314)

Elmousalami, H. H. (2020). Artificial intelligence and parametric construction cost estimate modeling: State-of-the-art review. Journal of Construction Engineering and Management, 146(1), 03119008. https://doi.org/10.1061/(ASCE)CO.1943-7862.000167

Enshassi, A., Mohamed, S., & Abdel-Hadi, M. (2013). Factors affecting the accuracy of pre-tender cost estimates in the Gaza Strip. Journal of Construction in Developing Countries, 18(1), 73–94. http://web.usm.my/jcdc/vol18_1_2013/art5_jcdc18-1.pdf (accessed on: 11 December 2022)

Fayek, A. R., & Oduba, A. (2005). Predicting Industrial Construction Labor Productivity Using Fuzzy Expert Systems. Journal of Construction Engineering and Management, 131(8), pp. 938–941. https://doi.org/10.1061/(asce)0733-9364(2005)131:8(938)

Frank, E., Hall, M. A., & Witten. I. H. (2016). The WEKA Workbench. Online Appendix for Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann, Fourth Edition, 2016.

Frank, E., Wang, Y., Inglis, S., Holmes, G., & Witten, I. H. (1998). Using model trees for classification. Machine Learning, 32(1), 63–76.

Guo, G., Wang, H., Bell, D., Bi, Y., & Greer, K. (2003). KNN model-based approach in classification. Lecture Notes in Computer Science (Including Subseries Lecture Notes in

Artificial Intelligence and Lecture Notes in Bioinformatics), 2888, pp. 986–996. https://doi.org/10.1007/978-3-540-39964-3_62

Guo, S. J., Chen, J. H., & Chiu, C. H. (2017). Fuzzy duration forecast model for wind turbine construction project subject to the impact of wind uncertainty. Automation in Construction, 81, pp. 401–410. https://doi.org/10.1016/j.autcon.2017.03.009

Halligan, D. W., Demsetz, L. A., Brown, J. D., & Pace, C. B. (1994). Action-Response Model and Loss of Productivity in Construction. Journal of Construction Engineering and Management, 120(1), 47–64. https://doi.org/10.1061/(asce)0733-9364(1994)120:1(47)

Halpin, D., and Riggs, L. (1992). Planning and analysis of construction operations. Wiley & Sons, Inc., New York. ISBN: 978-0-471-55510-0

Hamzeh, F. R., El Samad, G., & Emdanat, S. (2019). Advanced Metrics for Construction Planning. Journal of Construction Engineering and Management, 145(11), pp. 1–16. https://doi.org/10.1061/(asce)co.1943-7862.0001702

Hanna, A. S., Taylor, C. S., and Sullivan, K. T. (2005). Impact of extended overtime on construction labor productivity. Journal of Construction Engineering and Management, 131(6), 734–739. https://doi.org/10.1061/(ASCE)0733-9364(2005)131:6(734)

Hasan, M., & Lu, M. (2017). Error Quantification and Visualization in Using Sensors to Position Backhoe Excavator (pp. 150–157). Computing in Civil Engineering: Smart Safety, Sustainability, and Resilience, ASCE. Lin, KY, El-Gohery, N, and Tang, P. (Eds), 150-157. https://doi.org/10.1061/9780784480847.019

Hasan, M., & Lu, M. (2019). Planning and scheduling bridge girders fabrication through shop-floor operations simulation. In Proceedings of the 2019 European Conference on Computing in Construction (Vol. 1, pp. 75–84). University College Dublin. https://doi.org/10.35490/ec3.2019.162

Hasan, M., & Lu, M. (2021). Error Propagation Model for Analyzing Project Labor Cost Budget Risks in Industrial Construction. Journal of Construction Engineering and Management, 147(4), 04021007. https://doi.org/10.1061/(asce)co.1943-7862.0002010

Hasan, M., & Lu, M. (2022). Variance Analysis on Regression Models for Estimating Labor Costs of Prefabricated Components. Journal of Computing in Civil Engineering, 36(5), 04022019. https://doi.org/10.1061/(ASCE)CP.1943-5487.0001037

Huang, Y., Li, J., & Fu, J. (2019). Review on application of artificial intelligence in civil engineering. CMES – Comp. Model. in Eng. and Scien. Tech Science Press.

Jeong, J., Jeong, J., Lee, J., Kim, D., & Son, J. (2022). Learning-driven construction productivity prediction for prefabricated external insulation wall system. Automation in Construction, 141, 104441. https://doi.org/10.1016/j.autcon.2022.104441

Karimi, H., Taylor, T. R. B., & Goodrum, P. M. (2017). Analysis of the impact of craft labour availability on North American construction project productivity and schedule performance. Construction Management and Economics, 35(6), 368–380. https://doi.org/10.1080/01446193.2017.1294257

Karlsen, J. T., & Lereim, J. (2005). Management of Project Contingency and Allowance. Cost Engineering, 47(9), pp. 24-29. Retrieved from: https://www.proquest.com/openview/6879ffe94eb49ac74a8625bbd2b6a9bb/1?pq-origsite=gscholar&cbl=49080 (Accessed on 12 April 2023)

Kasperkiewicz, J., Racz, J., & Dubrawski, A. (1995). HPC Strength Prediction Using Artificial Neural Network. J. of Comp. in Civil Eng., 9(4), 279–284.

Kisi, K. P., Mani, N., Rojas, E. M., and Foster, E. T. (2017). Optimal productivity in labor-intensive construction operations: Pilot study. Journal of Construction Engineering and Management, 143(3). https://doi.org/10.1061/(ASCE)CO.1943-7862.0001257

Koch, K. R., Kuhlmann, H., and Schuh, W. D. (2010). Approximating covariance matrices estimated in multivariate models by estimated auto- and cross-covariances. Journal of Geodesy, 84(6), 383–397. https://doi.org/10.1007/s00190-010-0375-5

Kuhn, M., & Johnson, K. (2018). Applied Predictive Modeling. New York: Springer. (3rd ed.) ISBN-13 978-1461468486.

Lee, J., Park, Y. J., Choi, C. H., & Han, C. H. (2017). BIM-assisted labor productivity measurement method for structural formwork. Automation in Construction, 84, pp. 121–132. https://doi.org/10.1016/j.autcon.2017.08.009

Lee, M. J., Hanna, A. S., & Loh, W. Y. (2004). Decision tree approach to classify and quantify cumulative impact of change orders on productivity. Journal of computing in civil engineering, 18(2), pp. 132-144. https://doi.org/10.1061/(ASCE)0887-3801(2004)18:2(132)

Lichti, D. D., Gordon, S. J., & Tipdecho, T. (2005). Error models and propagation in directly georeferenced terrestrial laser scanner networks. Journal of Surveying Engineering, 131(4), 135–142. https://doi.org/10.1061/(ASCE)0733-9453(2005)131:4(135)

Lingras, P., & Adamo, M. (1996). Average and Peak Traffic Volumes: Neural Nets, Regression, Factor Approaches. Journal of Computing in Civil Engineering, 10(4), 300–306.

Liu, J., Siu, M. F. F., & Lu, M. (2016). Modular construction system simulation incorporating off-shore fabrication and multi-mode transportation. In Proceedings - Winter Simulation Conference 2016, pp. 3269–3280. https://doi.org/10.1109/WSC.2016.7822358

Liu, M., Ballard, G., & Ibbs, W. (2011). Workflow Variation and Labor Productivity: Case Study. Journal of Management in Engineering, 27(4), 236–242. https://doi.org/10.1061/(asce)me.1943-5479.0000056

Liu, Z., Sadiq, R., Rajani, B., & Najjaran, H. (2010). Exploring the Relationship between Soil Properties and Deterioration of Metallic Pipes Using Predictive Data Mining Methods. Journal of Computing in Civil Engineering, 24(3), 289–301. https://doi.org/10.1061/(asce)cp.1943-5487.0000032

Lowe, D. J., Emsley, M. W., & Harding, A. (2006). Predicting Construction Cost Using Multiple Regression Techniques. Journal of Construction Engineering and Management, 132(7), 750–758. https://doi.org/10.1061/(asce)0733-9364(2006)132:7(750)

Lu, M., Abourizk, S. M., and Hermann, U. H. (2001). Estimating labor productivity using probability inference neural network. Journal of Computing in Civil Engineering, 14(4), 241–248. https://doi.org/10.1061/(ASCE)0887-3801(2000)14:4(241)

Lu, M., Soleimanifar, M. & Mohsenijam, A. (2017). 'Practical Framework for Estimating and Planning Engineered Material Fabrication: Case of Piping Spools'. Proceedings of 2017 International Conference on Business, Big-Data, and Decision Sciences (ICBBD2017), Thailand, 2-4 August 2017. 41-48.

Luo, T., Xue, X., Tan, Y., Wang, Y., & Zhang, Y. (2021). Exploring a body of knowledge for promoting the sustainable transition to prefabricated construction. Engineering, Construction and Architectural Management. 28(9), pp. 2637 - 2666. https://doi.org/10.1108/ECAM-03-2020-0154

Maloney, W. F., & McFillen, J. M. (1985). Valence of and Satisfaction with Job Outcomes. Journal of Construction Engineering and Management, 111(1), 53–73. https://doi.org/10.1061/(asce)0733-9364(1985)111:1(53)

Marchionni, V., Cabral, M., Amado, C., & Covas, D. (2016). Estimating Water Supply Infrastructure Cost Using Regression Techniques. Journal of Water Resources Planning and Management, 142(4), 04016003. https://doi.org/10.1061/(asce)wr.1943-5452.0000627

Marié, S. (2022). python-m5p-M5 Prime regression trees in python, compliant with scikit-learn. In PyCon. DE & PyData Berlin, 2022. Retrieved form: https://hal.science/hal-03762155/ (accessed on 12 April 2023)

Minato, T., & Ashley, D. B. (1998). Data-Driven Analysis of "Corporate Risk" Using Historical Cost-Control Data. Journal of Construction Engineering and Management, 124(1), pp. 42–47. https://doi.org/10.1061/(asce)0733-9364(1998)124:1(42).

Mirahadi, F., & Zayed, T. (2016). Simulation-based construction productivity forecast using Neural-Network-Driven Fuzzy Reasoning. Automation in Construction, 65, pp. 102–115. https://doi.org/10.1016/j.autcon.2015.12.021

Mohsenijam, A., & M. Lu. (2019). Framework for developing labour-hour prediction models from project design features: Case study in structural steel fabrication. Canadian Journal of Civil Engineering. 46 (10): 871–880. https://doi.org/10.1139/cjce-2018-0349.

Mohsenijam, A., Lu, M., & Naumets, S. (2022). Integrating model tree and modified stepwise regression in concrete slump prediction and steel fabrication estimating. Can. J. of Civil Eng., 49(4), 478-486.

Mohsenijam, A., Siu, M.-F. F., & Lu, M. (2017). Modified Stepwise Regression Approach to Streamlining Predictive Analytics for Construction Engineering Applications. Journal of Computing in Civil Engineering, 31(3), 04016066. https://doi.org/10.1061/(asce)cp.1943-5487.0000636

Morgan, J. N., & Sonquist, J. A. (1963). Problems in the analysis of survey data, and a proposal. Journal of the American statistical association, 58(302), pp. 415-434. https://doi.org/10.2307/2283276

Motwani, J., Kumar, A., & Novakoski, M. (1995). Measuring construction productivity: a practical approach. Work Study, 44(8), 18–20. https://doi.org/10.1108/00438029510103310

Mulholland, B., and Christian, J. (1999). Risk assessment in construction schedules. Journal of Construction Engineering and Management, 125(1), 8–15. https://doi.org/10.1061/(ASCE)0733-9364(1999)125:1(8)

Najafzadeh, M., Laucelli, D. B., & Zahiri, A. (2017). Application of model tree and evolutionary polynomial regression for evaluation of sediment transport in pipes. KSCE Journal of Civil Engineering, 21, pp. 1956-1963. https://doi.org/10.1007/s12205-016-1784-7

Nasirzadeh, F., and Nojedehi, P. (2013). Dynamic modeling of labor productivity in construction projects. International Journal of Project Management, 31(6), 903–911. https://doi.org/10.1016/j.ijproman.2012.11.003

Naumets, S., & Lu, M. (2021). Investigation into explainable regression trees for construction engineering applications. Journal of Construction Engineering and Management, 147(8), 04021084. https://doi.org/10.1061/(asce)co.1943-7862.0002083

Navidi, W. C. (2006). Statistics for engineers and scientists. New York: McGraw-Hill. ISBN: 9780071222051

Olive, D. J. (2017). Linear regression. Linear Regression (pp. 1–494). Springer International Publishing. https://doi.org/10.1007/978-3-319-55252-1

Park, H. S. (2006). Conceptual framework of construction productivity estimation. KSCE Journal of Civil Engineering, 10(5), 311–317. https://doi.org/10.1007/bf02830084

Park, H. S., Thomas, S. R., and Tucker, R. L. (2005). Benchmarking of construction productivity. Journal of Construction Engineering and Management, 131(7), 772–778. https://doi.org/10.1061/(ASCE)0733-9364(2005)131:7(772)

Parker, AD., Barrie, D. S., and Snyder, R.M. (1984), Planning and Estimating Heavy Construction, McGraw-Hill, New York. ISBN 10: 007048489

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. the Journal of machine Learning research, 12, 2825-2830.

Peurifoy, R. L., Oberlender, G. D., (2001). Estimating Construction Costs (No. 5th ed.). New York, NY: McGraw-Hill. ISBN: 9780072435801.

PMI (Project Management Institute) (2017). A guide to the project management body of knowledge (PMBOK guide). 6th ed. Newtown Square, PA: PMI.

Portas, J., & AbouRizk, S. (1997). Neural network model for estimating construction productivity. Journal of construction engineering and management, 123(4), pp. 399-410. https://doi.org/10.1061/(ASCE)0733-9364(1997)123:4(399)

Project Management Institute (2017). A guide to the project management body of knowledge (PMBOK guide), 6th Ed., Newtown Square, Pennsylvania.

Puatanachokchai, C., and Mikhail, E. M. (2008). Adjustability and error propagation for true replacement sensor models. ISPRS Journal of Photogrammetry and Remote Sensing, 63(3), 352–364. https://doi.org/10.1016/j.isprsjprs.2007.10.001

Python Software Foundation (2019). Python 3.8.0. Link: https://www.python.org/downloads/release/python-380/ (accessed on 12 December 2022)

Quinlan, J.R. (1992). Learning with Continuous Classes. Proceedings of Australian Joint Conference on Artificial Intelligence, Hobart 16-18 November 1992, pp. 343-348. Retrieved from: https://www.rulequest.com/Personal/q.ai92.ps (accessed on 12 April 2023)

Radhakrishnan, R., & McAdams, D. A. (2005). A methodology for model selection in engineering design. Journal of Mechanical Design, Transactions of the ASME, 127(3), 378–387. https://doi.org/10.1115/1.1830048

Ranjan, A., Singh, V. P., Mishra, R. B., Thakur, A. K., & Singh, A. K. (2021). Sentence polarity detection using stepwise greedy correlation-based feature selection and random forests: an fMRI study. Journal of Neurolinguistics, 59, 100985 (pp. 1 -12). https://doi.org/10.1016/j.jneuroling.2021.100985

Rodríguez, J. D., Pérez, A., & Lozano, J. A. (2013). A general framework for the statistical analysis of the sources of variance for classification error estimators. Pattern Recognition, 46(3), 855–864. https://doi.org/10.1016/j.patcog.2012.09.007

Rojas, E. M., and Aramvareekul, P. (2003). Is construction labor productivity really declining? Journal of Construction Engineering and Management, 129(1), 41–46. https://doi.org/10.1061/(ASCE)0733-9364(2003)129:1(41)

Shaheen, A. A., Fayek, A. R., & AbouRizk, S. M. (2007). Fuzzy Numbers in Cost Range Estimating. Journal of Construction Engineering and Management, 133(4), 325–334. https://doi.org/10.1061/(asce)0733-9364(2007)133:4(325)

Shahpari, M., Saradj, F. M., Pishvaee, M. S., & Piri, S. (2020). Assessing the productivity of prefabricated and in-situ construction systems using hybrid multi-criteria decision making method. Journal of Building Engineering, 27(September 2019), 100979. https://doi.org/10.1016/j.jobe.2019.100979

Shen, Z., & Issa, R. R. A. (2010). Quantitative evaluation of the BIM-assisted construction detailed cost estimates. Electronic Journal of Information Technology in Construction, 15, pp. 234–257. Retrieved from: https://www.itcon.org/paper/2010/18 (accessed on 12 April 2023)

Smith, S. D. (1999). Earthmoving productivity estimation using linear regression techniques. Journal of Construction Engineering and Management, 125(3), pp. 133–141. https://doi.org/10.1061/(ASCE)0733-9364(1999)125:3(133)

Song, L., & AbouRizk, S. M. (2008). Measuring and Modeling Labor Productivity Using Historical Data. Journal of Construction Engineering and Management, 134(10), pp. 786–794. https://doi.org/10.1061/(asce)0733-9364(2008)134:10(786)

Sonmez, R., and Rowings, J. E. (1998). Construction labor productivity modeling with neural networks. Journal of Construction Engineering and Management, 124(6), 498–504. https://doi.org/10.1061/(ASCE)0733-9364(1998)124:6(498)

Taghi Sattari, M., Pal, M., Apaydin, H., & Ozturk, F. (2013). M5 model tree application in daily river flow forecasting in Sohu Stream, Turkey. Water Resources, 40, pp. 233-242.

Tam, C. M., Tong, T. K. l., & Tse, S. L. (2002, May 1). Artificial neural networks model for predicting excavator productivity. Engineering, Construction and Architectural Management. 9(5), pp. 446–452. https://doi.org/10.1108/eb021238

Taylor (2022). An Introduction to Error Analysis: The Study of Uncertainties in Physical Measurements. University Science Books. Melville, NY, USA. eISBN 978-1-940380-09-4.

Thomas, H. R. (2015). Benchmarking construction labor productivity. Practice Periodical on Structural Design and Construction, 20(4), 1–10. https://doi.org/10.1061/(ASCE)SC.1943-5576.0000141

Thomas, H. R., & Daily, J. (1983). Crew Performance Measurement Via Activity Sampling. Journal of Construction Engineering and Management, 109(3), 309–320. https://doi.org/10.1061/(asce)0733-9364(1983)109:3(309)

Thomas, H. R., and Sakarcan, A. S. (1994). Forecasting labor productivity using factor model. Journal of Construction Engineering and Management, 120(1), 228–239. https://doi.org/10.1061/(ASCE)0733-9364(1994)120:1(228)

Thomas, H. R., Horman, M. J., de Souza, U. E. L., & Zavřski, I. (2002). Reducing Variability to Improve Performance as a Lean Construction Principle. Journal of Construction Engineering and Management, 128(2), 144–154. https://doi.org/10.1061/(asce)0733-9364(2002)128:2(144)

Touran, A., & Liu, J. (2015). A Method for Estimating Contingency Based on Project Complexity. Procedia Engineering, 123, 574–580. https://doi.org/10.1016/j.proeng.2015.10.110

Trost, S. M., & Oberlender, G. D. (2003). Predicting Accuracy of Early Cost Estimates Using Factor Analysis and Multivariate Regression. Journal of Construction Engineering and Management, 129(2), 198–204. https://doi.org/10.1061/(asce)0733-9364(2003)129:2(198)

Tsehayae, A. A., & Fayek, A. R. (2016). System model for analysing construction labour productivity. Construction Innovation, 16(2), pp. 203–228. https://doi.org/10.1108/CI-07-2015-0040

UCI (University of California, Irvine). 2020. Concrete compressive strength data set. Accessed March 2, 2023. https://archive.ics.uci.edu/ml/datasets/Concrete+Compressive+Strength.

Vanli, N. D., Sayin, M. O., Mohaghegh N., M., Ozkan, H., & Kozat, S. S. (2019). Nonlinear regression via incremental decision trees. Pattern Recognition, 86, pp. 1–13. https://doi.org/10.1016/j.patcog.2018.08.014

Vereen, S. C., Rasdorf, W., & Hummer, J. E. (2016). Development and Comparative Analysis of Construction Industry Labor Productivity Metrics. Journal of Construction Engineering and Management, 142(7), 1–9. https://doi.org/10.1061/(asce)co.1943-7862.0001112

Veregin, H. (1995). Developing and testing of an error propagation model for GIS overlay operations. International Journal of Geographical Information Systems, 9(6), 595–619. https://doi.org/10.1080/02693799508902059

Wai-Lok Lai, W., Dérobert, X., & Annan, P. (2018). A review of Ground Penetrating Radar application in civil engineering: A 30-year journey from Locating and Testing to Imaging and Diagnosis. NDT and E International, 96, 58–78. https://doi.org/10.1016/j.ndteint.2017.04.002

Wang, J. X., Roy, C. J., & Xiao, H. (2018). Propagation of input uncertainty in presence of model-form uncertainty: A multifidelity approach for computational fluid dynamics applications. ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering, 4(1). https://doi.org/10.1115/1.4037452

Wang, N., Chang, Y. C., & El-Sheikh, A. A. (2012). Monte Carlo simulation approach to life cycle cost management. Structure and Infrastructure Engineering, 8(8), pp. 739–746. https://doi.org/10.1080/15732479.2010.481304.

Wang, Y., & Witten, I. H. (1996). Induction of model trees for predicting continuous classes. Proceedings of the 9th European Conference on Machine Learning Poster Papers. (pp. 1 -12) Retrieved from http://researchcommons.waikato.ac.nz/handle/10289/1183 (accessed on 12 April 2023)

Xue, J., Leung, Y., and Ma, J. H. (2015). High-order Taylor series expansion methods for error propagation in geographic information systems. Journal of Geographical Systems, 17(2), 187–206. https://doi.org/10.1007/s10109-014-0207-x

Yi, W., & Chan, A. P. C. (2014). Critical Review of Labor Productivity Research in Construction Journals. Journal of Management in Engineering, 30(2), pp. 214–225. https://doi.org/10.1061/(asce)me.1943-5479.0000194.

Yu, L., Lai, K. K., Wang, S., & Huang, W. (2006). A bias-variance-complexity trade-off framework for complex system modeling. In Lec.Notes in Comp. Sci.. 3980, 518–527. Springer Verlag.

Zayed, T. M., & Halpin, D. W. (2005). Productivity and Cost Regression Models for Pile Construction. Journal of Construction Engineering and Management, 131(7), 779–789. https://doi.org/10.1061/(asce)0733-9364(2005)131:7(779)

Appendix A

Uncertainty Quantification

A.1 Error Propagation

Measurement is a process subject to variation. Since all the measurements are subject to discrepancy, what is conceived as the actual value of a measurement is merely an estimate of the true value. The difference between a measured value of a quantity and its true value is considered as an error. If it is possible to obtain a reasonable estimation of true value, then it can be used to get an estimated value of error which is basically known as the residual. Among three different types of error, blunder should be avoided during the measuring process through verification. Systematic error of a derived quantities can be derived from the systematic error of the measured (known) quantities. This is done by functional substitution with truncated Taylor Series which behaves like removing the first term.

Taylor series is a representation of a function as an infinite sum of terms calculated from the values of its derivatives at a single point. Eq. A.1 expresses a function of *x, f(x)* which can be expanded value of $x = x_o$.

$$f(x) = \sum_{n=0}^{\infty} \frac{f^n(x_0)}{n!}(x - x_0)^n \qquad \text{(A. 1)}$$

Eq. A.1 can be further expanded as,

$$f(x) = f(x_0) + \frac{f^1(x_0)}{1!}(x - x_0) + \frac{f^2(x_0)}{2!}(x - x_0)^2 + \ \ldots\ldots\ldots \qquad (\mathrm{A.2})$$

As the value of n goes up, the higher-order terms become insignificant. Thus, keeping the first two terms of the series Eq. A.2 can be written as,

$$f(x) = f(x_0) + \frac{f^1(x_0)}{1!}(x - x_0) \qquad (A.3)$$

or,

$$y = y_0 + y'\Delta x$$

or,

$$y - y_0 = y'\Delta x$$

or,

$$\Delta y = y'\Delta x$$

or,

$$dy = \frac{\partial y}{\partial x}dx \qquad (A.4)$$

Now if y has m number of observations and each of them is dependent on n number of independent variables for x then the Eq. A.4 becomes,

$$\begin{bmatrix} dy_1 \\ dy_2 \\ \vdots \\ dy_m \end{bmatrix} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \cdots & \frac{\partial y_1}{\partial x_n} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} & & \frac{\partial y_2}{\partial x_n} \\ \vdots & & \ddots & \vdots \\ \frac{\partial y_m}{\partial x_1} & \frac{\partial y_m}{\partial x_2} & \cdots & \frac{\partial y_m}{\partial x_n} \end{bmatrix} \begin{bmatrix} dx_1 \\ dx_2 \\ \vdots \\ dx_n \end{bmatrix}$$

or,

$$dy = J_{xy} dx \qquad (A.5)$$

Eq. A.5 is the general form of systematic error propagation where J_{xy} is called the Jacobian (Jacobian matrix) of the equation. This one is the equation for quantifying the systematic error of a measurement.

A.1.1 Propagation of Random Error

It is not perfect using Eq. A.5 to quantify the random error of any measurement. Values of measurement follow Gaussian distribution due to the presence of the randomness of error. Standard deviation/Variance of any set of measured values is a reasonable estimate of randomness. Thus, propagation of random error follows the law of propagation of variance and covariance (POV) which can be expressed by the following equation,

$$\sum_{yy} = J_{xy} \sum_{xx} J_{yx}^T \qquad (A.6)$$

or,

$$C_y = J_{yx}\, C_x\, J_{yx}^T \qquad (A.7)$$

or,

$$\begin{bmatrix} \sigma_{y_1}^2 & \sigma_{y_1y_2} & \cdots & \sigma_{y_1y_m} \\ \sigma_{y_2y_1} & \sigma_{y_2}^2 & & \sigma_{y_1y_m} \\ \vdots & & \ddots & \vdots \\ \sigma_{y_my_1} & \sigma_{y_my_2} & \cdots & \sigma_{y_m}^2 \end{bmatrix} = J_{yx} \begin{bmatrix} \sigma_{x_1}^2 & \sigma_{x_1x_2} & \cdots & \sigma_{x_1x_n} \\ \sigma_{x_2x_1} & \sigma_{x_2}^2 & & \sigma_{x_1x_n} \\ \vdots & & \ddots & \vdots \\ \sigma_{x_nx_1} & \sigma_{x_nx_2} & \cdots & \sigma_{x_n}^2 \end{bmatrix} J_{yx}^T \qquad (A.8)$$

Here, $\sum_{yy}$ is the covariance matrix of random output y and $\sum_{xx}$ is the covariance matrix of random input x.

A.1.2 Explanation for $C_y = J_{yx}\, C_x\, J_{yx}^T$

For a specific measurement, any observation (input) x can be mapped onto output y. Output y basically can be expressed as a function of x in 2D space. Now if any error (Δx) exists in input value of x, this will propagate onto y through *f(x)*. This error can be quantified with the approximation of linearization of *f(x)* at the point *(x,y)* and the slope of the line would be $\frac{dy}{dx}$. Using the same basic principle of Eq. A.4 the measurement of the error would be,

$$\Delta y = \frac{dy}{dx}\Delta x \qquad (A.9)$$

If now a set of observation (input) of a specific measurement x, which is random in nature and follows normal distribution, can be mapped onto a set of random output (normally distributed also) y with a relationship function *f(x)*, its shape would be somewhat distorted and the resulting distribution would be asymmetric, certainly not Gaussian anymore.

When approximating *f(x)* by a first-order Taylor series expansion (Eq. A.3) about the point $x = \mu_x$, the following linear relationship can be obtained,

$$y = f(x) \approx f(\mu_x) + \frac{\partial f(\mu_x)}{\partial x}\,(x - \mu_x) \qquad (A.10)$$

If $y = f(x_1, x_2, x_3, \ldots \ldots, x_n)$ then the Eq. A.10 becomes,

$$y \approx f(\mu_1, \mu_2, \ldots \ldots, \mu_n) + \sum_{i=1}^{n} \left[\frac{\partial f}{\partial x}((\mu_1, \mu_2, \ldots \ldots, \mu_n)\right](x_i - \mu_i)$$

or,

$$y \approx a_o + \sum_{i=1}^{n} a_i\ (x_i - \mu_i) \qquad (A.11)$$

where, $a_o = f(\mu_1, \mu_2, \ldots \ldots, \mu_n)$ and $a_i = \frac{\partial f}{\partial x}((\mu_1, \mu_2, \ldots \ldots, \mu_n)$

Now

$$\mu_y = E[y] = E\left[a_o + \sum_{i=1}^{n} a_i\ (x_i - \mu_i)\right]$$

$$= E\,[a_o] + \sum_{i=1}^{n} E[a_i x_i] - E[a_i \mu_i]$$

$$= a_o + \sum_{i=1}^{n} a_i E[x_i] - a_i E[\mu_i]$$

$$= a_o + \sum_{i=1}^{n} a_i \mu_i - a_i \mu_i$$

$$= a_o + \sum_{i=1}^{n} a_i \mu_i - a_i \ = a_o$$

$$\mu_y = f(\mu_1, \mu_2, \dots\dots, \mu_n) \qquad (A.12)$$

And,

$$\sigma_y^2 = E\left[\left(y - \mu_y\right)^2\right]$$

$$\sigma_y^2 \approx E\left[\left(a_o + \sum_{i=1}^{n} a_i \ (x_i - \mu_i) - a_o\right)^2\right] = E\left[\left(\sum_{i=1}^{n} a_i \ (x_i - \mu_i)\right)^2\right]$$

$$= E\left[\sum_{i=1}^{n} {a_i}^2 \ (x_i - \mu_i)^2\right]$$

$$= \sum_{i=1}^{n} {a_i}^2 E[(x_i - \mu_i)^2]$$

$$= \sum_{i=1}^{n} {a_i}^2 \sigma_i^2$$

$$= \sum_{i=1}^{n} \left(\frac{\partial f}{\partial x_i}\right)^2 \sigma_i^2$$

So, the final equation becomes,

$$\sigma_y^2 = \sum_{i=1}^{n} \left(\frac{\partial f}{\partial x_i}\right)^2 \sigma_i^2 \qquad (A.13)$$

If y has m number of observation which is dependent on n number of variables of x, then general matrix form of Eq. A.13 would be,

$$\begin{bmatrix} \sigma_{y_1}^2 & \sigma_{y_1 y_2} & \cdots & \sigma_{y_1 y_m} \\ \sigma_{y_2 y_1} & \sigma_{y_2}^2 & & \sigma_{y_1 y_m} \\ \vdots & & \ddots & \vdots \\ \sigma_{y_m y_1} & \sigma_{y_m y_2} & \cdots & \sigma_{y_m}^2 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & & \frac{\partial f_2}{\partial x_n} \\ \vdots & & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix} \begin{bmatrix} \sigma_{x_1}^2 & \sigma_{x_1 x_2} & \cdots & \sigma_{x_1 x_n} \\ \sigma_{x_2 x_1} & \sigma_{x_2}^2 & & \sigma_{x_1 x_n} \\ \vdots & & \ddots & \vdots \\ \sigma_{x_n x_1} & \sigma_{x_n x_2} & \cdots & \sigma_{x_n}^2 \end{bmatrix} \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_2}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \\ \frac{\partial f_1}{\partial x_2} & \frac{\partial f_2}{\partial x_2} & & \frac{\partial f_m}{\partial x_n} \\ \vdots & & \ddots & \vdots \\ \frac{\partial f_1}{\partial x_n} & \frac{\partial f_2}{\partial x_n} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$

or,

$$C_y = J_{yx}\, C_x\, J_{yx}^T \qquad (A.14)$$

Which cross validates the Eq. A.6.

Appendix B

Source Codes for Program Implementation

B.1 General

All the programs are written in Python 3.8 environment using the Microsoft Visual Studio 2022 platform.

B.2 Generate Statistical Description of the Regression Model

B.2.1 Algorithm 1

```
#importing necessary libraries


import pandas as pd

from sklearn.linear_model import LinearRegression

from sklearn.utils import resample

import numpy as np


# Specify the absolute path of the Excel file

file_path = 'C:/Users/shuvo/Desktop/My Code Kitchen/Model
Tree/data.xlsx'
```

```
# Read the Excel file, skipping the first row as it contains column
labels

data = pd.read_excel(file_path)



#Extract the input and output variables from the DataFrame:


X = data.iloc[:, 1:]  # Input variables

y = data.iloc[:, 0]   # Output variable



#Initialize the parameters n (percentage of data to select) and m
(number of iterations):


n = 0.8  # 80% of the data will be used for each iteration

m = 10   # Number of iterations



#Create an empty DataFrame to store the model parameters:

parameters_df = pd.DataFrame(columns=['Intercept'] + list(X.columns))



#Run the regression model process m times:

for i in range(m):
```

```
    # Randomly select n% of the data
    X_sample, y_sample = resample(X, y, n_samples=int(n * len(X)),
random_state=i)

    # Fit the linear regression model
    model = LinearRegression()
    model.fit(X_sample, y_sample)

    # Store the model parameters in the DataFrame
    parameters_df.loc[i] = [model.intercept_] + list(model.coef_)



#Print the table of model parameters:
print(parameters_df)




#Calculate the average and standard deviation of the model parameters:
avg_parameters = parameters_df.mean()
std_parameters = parameters_df.std()

print("Average Parameters:")
print(avg_parameters)

print("Standard Deviation of Parameters:")
print(std_parameters)
```

```
#End of the
print ("code complete")
```

B.2.2 Explanation for Algorithm 2

Following the Python script that performs linear regression with bootstrapping on a dataset stored in an Excel file. The purpose of the script is to estimate the parameters of a linear regression model multiple times using different random subsets of the data to evaluate the stability and uncertainty of the model.

Here's a breakdown of the code:

1. Import necessary libraries: The script imports required libraries, including pandas for data manipulation, scikit-learn's LinearRegression for linear regression modeling, and numpy for numerical computations.
2. Specify the file path: The variable file_path contains the absolute path of the Excel file that contains the dataset.
3. Read the data: The script reads the data from the Excel file into a pandas DataFrame, skipping the first row as it contains column labels.
4. Extract input and output variables: The dataset is split into input variables X and the output variable y. X contains all the columns of the DataFrame except the first column (assuming the first column contains the output variable).

5. Set parameters for bootstrapping: The variable n represents the percentage of data to select for each iteration of bootstrapping (in this case, 80% of the data), and m represents the number of iterations to perform.
6. Create an empty DataFrame: The script creates an empty DataFrame called parameters_df to store the model parameters for each iteration.
7. Run the regression model process m times: The script runs a loop for m iterations. In each iteration, it randomly selects n% of the data for training using bootstrapping (random sampling with replacement). It then fits a linear regression model using scikit-learn's LinearRegression class and stores the model's intercept and coefficients in the parameters_df DataFrame.
8. Print the table of model parameters: After the loop finishes, the script prints the table of model parameters, showing the intercept and coefficients for each iteration.
9. Calculate the average and standard deviation of the model parameters: The script calculates the average and standard deviation of the model parameters from the parameters_df DataFrame.
10. Print the average and standard deviation of the model parameters: The script prints the average and standard deviation of the model parameters to evaluate the stability and uncertainty of the model.

This program employs bootstrapping to generate multiple linear regression models using different subsets of the data, and then it calculates the average and standard deviation of the model parameters to assess the robustness of the linear regression estimates.

B.3 Model Tree Algorithm to identify the Data Classes

B.3.1 Algorithm 2

```
# Import the necessary modules and libraries

import numpy as np

import pandas as pd

import seaborn as sns


from sklearn.model_selection import cross_val_score

from sklearn.tree import DecisionTreeRegressor, export_text

from m5py import M5Prime


# Load your dataset using pandas

file_path = 'C:/Users/shuvo/Desktop/My Code Kitchen/Model Tree/Data.csv'

# Read the Excel file, skipping the first row as it contains column
labels

data = pd.read_csv(file_path)  # Assuming the first row contains the
variable names


X = data.iloc[:, 1:]

# Extract variable names from the first row

feature_names = X.columns.tolist()


# Extract input features (X) and target variable (y)

X = data.iloc[:, 1:].values  # Exclude the first column as input
features; convert to NumPy array

y = data.iloc[:, 0].values  # First column as the target variable;
convert to NumPy array
```

```
print (type(X))

print (type(y))

input("hold")



# Define regression models and evaluate them on 10-fold CV

regr_0 = DecisionTreeRegressor()

regr_0_label = "Tree 0"

regr_0_scores = cross_val_score(regr_0, X, y, cv=10)


regr_1 = M5Prime(use_smoothing=False, use_pruning=False)

regr_1_label = "Tree 1"

regr_1_scores = cross_val_score(regr_1, X, y, cv=10)


regr_2 = M5Prime(use_smoothing=False)

regr_2_label = "Tree 2"

regr_2_scores = cross_val_score(regr_2, X, y, cv=10)


regr_3 = M5Prime(use_smoothing=True)

regr_3_label = "Tree 3"

regr_3_scores = cross_val_score(regr_3, X, y, cv=10)


scores = np.c_[regr_0_scores, regr_1_scores, regr_2_scores,
regr_3_scores]

avgs = scores.mean(axis=0)
```

```
stds = scores.std(axis=0)
labels = [regr_0_label, regr_1_label, regr_2_label, regr_3_label]

scores_df = pd.DataFrame(data=scores, columns=labels)
sns.violinplot(data=scores_df)

# Fit the final models and print the trees:
#
regr_0.fit(X, y)
print("\n----- %s" % regr_0_label)
print(export_text(regr_0, feature_names=feature_names))

regr_1.fit(X, y)
print("\n----- %s" % regr_1_label)
print(regr_1.as_pretty_text(feature_names=feature_names))

regr_2.fit(X, y)
print("\n----- %s" % regr_2_label)
print(regr_2.as_pretty_text(feature_names=feature_names))

regr_3.fit(X, y)
print("\n----- %s" % regr_3_label)
print(regr_3.as_pretty_text(feature_names=feature_names))
```

B.3.2 Explanations for Algorithm 2

Following code performs an evaluation of regression models on a dataset using cross-validation and compares the performance of different decision tree regression models, including M5Prime with different configurations.

Here's an explanation of the code:

1. Import necessary libraries: The script imports necessary libraries, including NumPy, pandas, seaborn for visualization, scikit-learn's DecisionTreeRegressor, and M5Prime from the m5py library.
2. Load the dataset: The script loads the dataset from the CSV file specified in the variable file_path using pandas.
3. Extract input features and target variable: The script extracts input features X and the target variable y from the dataset. The input features are taken from columns 1 and onward, while the target variable is taken from the first column.
4. Define and evaluate regression models using cross-validation: The script defines four regression models with different configurations: a standard DecisionTreeRegressor (regr_0) and three M5Prime models (regr_1, regr_2, and regr_3) with different settings for smoothing and pruning. It then evaluates the models' performance using 10-fold cross-validation and stores the cross-validation scores in regr_0_scores, regr_1_scores, regr_2_scores, and regr_3_scores.

5. Compute the mean and standard deviation of the cross-validation scores: The script calculates the average and standard deviation of the cross-validation scores for each model and stores them in avgs and stds.
6. Visualize the cross-validation scores: The script creates a DataFrame scores_df containing the cross-validation scores of all four models and visualizes the distribution of scores using a violin plot from seaborn.
7. Fit the final models and print the trees: The script fits the final models (regr_0, regr_1, regr_2, and regr_3) on the entire dataset and prints the decision trees for each model using the export_text and as_pretty_text functions from scikit-learn and M5Prime, respectively. The trees are printed with feature names to display the decision rules used by the models.

This program assesses the performance of various decision tree regression models and evaluate their efficacy on the provided dataset through cross-validation. The visualization and displayed trees offer valuable insights into the decision-making process of each model, aiding in comprehending their predictive abilities.

B.4 Program to Run CPM Analysis to Plot S-Curve

B.4.1 Algorithm 3

```
import networkx as nx
import matplotlib.pyplot as plt
import csv
```

```
def solve_cpm(tasks, dependencies):
    # Create a Directed Acyclic Graph (DAG)
    G = nx.DiGraph()

    # Add nodes to the graph
    for task in tasks:
        G.add_node(task['name'])

    # Add edges (dependencies) to the graph
    for dependency in dependencies:
        G.add_edge(dependency[0], dependency[1])

    # Calculate the earliest start and finish times
    earliest_start_time = {}
    earliest_finish_time = {}
    duration_map = {task['name']: task['duration'] for task in tasks}

    for task in nx.topological_sort(G):
        max_earliest_start_time = 0
        for predecessor in G.predecessors(task):
            max_earliest_start_time = max(max_earliest_start_time,
earliest_finish_time[predecessor])

        earliest_start_time[task] = max_earliest_start_time
        try:
            earliest_finish_time[task] = max_earliest_start_time +
duration_map[task]
```

```
        except Exception as e:
            print("exception ", e)

    # print the earliest start and finish times
    print("Earliest Start Time:", earliest_start_time)
    print("Earliest Finish Time:", earliest_finish_time)




    # Calculate the latest start and finish times
    latest_start_time = {}
    latest_finish_time = {}

    for task in list(nx.topological_sort(G))[::-1]:
        if not list(G.successors(task)):
            latest_finish_time[task] = earliest_finish_time[task]
            latest_start_time[task] = earliest_start_time[task]
        else:
            min_latest_finish_time = float('inf')
            for successor in G.successors(task):
                min_latest_finish_time = min(min_latest_finish_time,
latest_start_time[successor])

            latest_finish_time[task] = min_latest_finish_time
            latest_start_time[task] = min_latest_finish_time -
duration_map[task]
```

```
    # Calculate total float (slack)

    total_float = {}



    for task in tasks:

        total_float[task['name']] = latest_start_time[task['name']] -
earliest_start_time[task['name']]



    return G, earliest_start_time, earliest_finish_time,
latest_start_time, latest_finish_time, total_float




def display_schedule(G, earliest_start_time, earliest_finish_time,
latest_start_time, latest_finish_time, tasks):

    labels = {task['name']: task['name'] for task in tasks}



    # Create a horizontal bar chart

    plt.figure(figsize=(10, 6))



    # Determine the y-axis position for each task

    y_positions = range(len(G.nodes))



    # Draw the bars for each task

    for task, y in zip(G.nodes, y_positions):

        x_start = earliest_start_time[task]

        x_finish = earliest_finish_time[task]

        width = x_finish - x_start

        plt.barh(y, width, left=x_start, height=0.5, align='center',
alpha=0.8)
```

```
        plt.text(x_start + width / 2, y, labels[task], ha='center',
va='center', color='black')



        x_start = latest_start_time[task]

        x_finish = latest_finish_time[task]

        width = x_finish - x_start

        plt.barh(y, width, left=x_start, height=0.2, align='center',
alpha=0.3, color='gray')



    # Set the x-axis and y-axis labels

    plt.xlabel('Time')

    plt.ylabel('Tasks')



    # Set the y-axis ticks and labels

    plt.yticks(y_positions, [labels[node] for node in G.nodes])



    # Set the x-axis limit based on the maximum finish time

    max_finish_time = max(earliest_finish_time.values())

    plt.xlim(0, max_finish_time + 10)



    # Display the chart

    plt.title('Cortical Path Method Schedule')

    plt.grid(True)

    plt.show()



## Define the tasks and their durations
```

```
#tasks = [
#    {'name': 'A', 'duration': 4},
#    {'name': 'B', 'duration': 3},
#    {'name': 'C', 'duration': 2},
#    {'name': 'D', 'duration': 5},
#    {'name': 'E', 'duration': 6},
#    {'name': 'F', 'duration': 4},
#    {'name': 'G', 'duration': 2}
#]
# #dependencies = [('A', 'B'), ('A', ' C'), ('A', ' G'), ('A', ' E'),
('B', 'D'), ('C', 'D'), ('D', 'F'), ('E', 'F'), ('E', ' G'), ('F', 'G')]
### Define the dependencies between tasks
#dependencies = [
#    ('A', 'B'),
#    ('A', 'C'),
#    ('A', 'G'),
#    ('A', 'E'),
#    ('B', 'D'),
#    ('C', 'D'),
#    ('D', 'F'),
#    ('E', 'F'),
#    ('E', 'G'),
#    ('F', 'G')



#]
#print("task",tasks)
```

```
#print("dependenccies",dependencies)




# Specify the path to the CSV file

csv_file_path = 'C:/Users/shuvo/Desktop/My Code Kitchen/Model
Tree/tasks.csv'



#Read task and dependency information from CSV file

tasks = []

dependencies = []



with open(csv_file_path, 'r') as csvfile:

    reader = csv.DictReader(csvfile)

    for row in reader:

        tasks.append({'name': row['Task'], 'duration':
int(row['Duration'])})

        if row['Dependencies']:

            dependencies.extend([(row['Task'], dependency.strip()) for
dependency in row['Dependencies'].split(',')])

print("task",tasks)

print("dependenccies",dependencies)

# Solve the CPM problem

G, earliest_start_time, earliest_finish_time, latest_start_time,
latest_finish_time, total_float = solve_cpm(tasks, dependencies)



# Display the schedule

display_schedule(G, earliest_start_time, earliest_finish_time,
latest_start_time, latest_finish_time, tasks)
```

B.4.2 Explanations of Algorithm 3

This code is an implementation of the Critical Path Method (CPM) to schedule tasks in a project. The CPM is a project management technique that identifies the critical tasks and their dependencies to determine the minimum time required to complete the project.

Here's an explanation of the code:

1. Import necessary libraries: The script imports the necessary libraries, including networkx for handling the Directed Acyclic Graph (DAG) representing the tasks, matplotlib for visualization, and csv to read task and dependency information from a CSV file.
2. Define the solve_cpm function: This function takes the list of tasks and their dependencies as input and calculates various parameters of the CPM, including earliest start time, earliest finish time, latest start time, latest finish time, and total float (slack).
3. Define the display_schedule function: This function visualizes the schedule of tasks using a horizontal bar chart. It takes the CPM parameters and the task information as input and plots the tasks' start and finish times.
4. Read task and dependency information from a CSV file: The script reads task and dependency information from a CSV file specified in the variable csv_file_path.

5. Solve the CPM problem: The script calls the solve_cpm function with the tasks and dependencies read from the CSV file. The function calculates the CPM parameters and returns them.
6. Display the schedule: The script calls the display_schedule function with the CPM parameters and task information to visualize the tasks' schedule using a horizontal bar chart. The chart shows the earliest start and finish times (colored bars) and the latest start and finish times (gray bars) for each task.

This program allows project managers to use the CPM technique to schedule tasks in a project, identify critical tasks, and determine the project's minimum completion time. The visual representation of the schedule helps in understanding the project's timeline and identifying tasks with slack time, which can be delayed without affecting the project's overall duration.

B.5 Program to Perform Data Processing by Taking Input from CPM Analysis and Plot Gant Chart for S stripe plotting

B.5.1 Algorithm 4

```
import tkinter as tk
from tkinter import filedialog, messagebox
import pandas as pd
from tkinter import ttk
import matplotlib.pyplot as plt
from tkcalendar import Calendar
import matplotlib.dates as mpl_dates
```

```
# Function to import CSV file

def import_csv():

    # Open file dialog to select CSV file

    filepath = filedialog.askopenfilename(filetypes=[("CSV Files",
"*.csv")])


    if filepath:

        # Read CSV file into pandas DataFrame

        global df

        df = pd.read_csv(filepath, header=0)


        # Clear previous table content

        table.delete(1.0, tk.END)


        # Print the number of variables

        num_variables = df.shape[1]

        print(f"Number of Variables: {num_variables}")


        #Get the column names/variable names from the first row

        var_names = list(df.columns)

        print("Variable Names:")

        for var_name in var_names:

            print(var_name)


        # Update the combobox options for variable selection

        variable_activity.set("")
```

```
        variable_start_date.set("")

        variable_duration.set("")

        activity_combobox['values'] = var_names

        start_date_combobox['values'] = var_names

        duration_combobox['values'] = var_names


        # Display DataFrame as a table in the GUI

        table.insert(tk.END, df.to_string(index=False))


# Function to handle the calendar selection

def pick_date():

    def on_date_select():

        global selected_date

        selected_date = cal.selection_get().strftime("%Y-%m-%d")  # Get
the selected date from the calendar

        messagebox.showinfo("Selected Date", f"Selected Date:
{selected_date}")




        # Use the selected date for further processing

        selected_date = pd.to_datetime(selected_date)

        selected_date = pd.Series(selected_date)

        selected_date =
selected_date.repeat(len(df)).reset_index(drop=True)


    # Create a new window for the calendar

    top = tk.Toplevel(window)
```

```
    # Create the calendar
    cal = Calendar(top, selectmode="day", date_pattern="yyyy-mm-dd")
    cal.pack()

    # Create a button to confirm the date selection
    confirm_button = tk.Button(top, text="Select Date",
command=on_date_select)
    confirm_button.pack()

# Function to handle Gantt chart plotting
def plot_gantt_chart():
    activity = variable_activity.get()
    start_date = variable_start_date.get()
    duration = variable_duration.get()

    # Check if all variables are selected
    if not activity or not start_date or not duration:
        messagebox.showerror("Error", "Please select Activity, Start
Date, and Duration.")
        return

    # Get the start date and duration from the DataFrame
    start_dates = df[start_date]
    durations = df[duration]

    try:
        selected_dates = pd.to_datetime(selected_date)
```

```
        start_dates = pd.to_timedelta(start_dates, unit='d') +
selected_dates

    except ValueError:

        messagebox.showerror("Error", "Invalid Start Date format.")

        return



    # Calculate finish dates

    finish_dates = start_dates + pd.to_timedelta(durations, unit='d')

    #print(start_dates, finish_dates, duration)



    # Create a Gantt chart

    fig, ax = plt.subplots()

    bars = ax.barh(df[activity], durations, left=start_dates,
height=0.5)

    ax.set_xlabel("Duration")

    ax.set_ylabel("Activities")

    ax.set_title("Gantt Chart")



    # Format x-axis as dates

    ax.xaxis_date()



    # Set the date format of the x-axis labels

    date_format = mpl_dates.DateFormatter("%Y-%m-%d")

    ax.xaxis.set_major_formatter(date_format)

    plt.xticks(rotation=90)  # Rotate x-axis labels vertically



    # Add duration values at the end of each bar
```

```
    for bar, duration in zip(bars, durations):
        end_date = bar.get_x() + bar.get_width()  # Calculate the end
date position
        ax.text(end_date, bar.get_y() + bar.get_height() / 2,
str(duration),
                ha='left', va='center')

    # Show the plot
    plt.show()

# Create the GUI window
window = tk.Tk()
window.title("CSV File Editor")

# Create a frame for the buttons
button_frame = tk.Frame(window)
button_frame.pack(side=tk.TOP, padx=50, pady=50)

# Button to select CSV file
import_button = tk.Button(button_frame, text="Import CSV",
command=import_csv)
import_button.pack(side=tk.LEFT)

# Create the text widget and scrollbars
table = tk.Text(window, wrap=tk.NONE)
table.pack(fill=tk.BOTH, expand=True)

# Create the horizontal scrollbar
```

```
x_scrollbar = tk.Scrollbar(window, orient=tk.HORIZONTAL,
command=table.xview)

x_scrollbar.pack(fill=tk.X, side=tk.BOTTOM)



# Configure the text widget to use the scrollbar

table.configure(xscrollcommand=x_scrollbar.set)



# Configure the window to adjust the size of its contents

window.pack_propagate(False)



# Bind the resizing event to adjust table size

window.bind("<Configure>", lambda event:
table.configure(width=event.width-20, height=event.height-120))



# Frame for Gantt chart options

gantt_frame = tk.Frame(window)

gantt_frame.pack(pady=10)



# Label for Activity selection

activity_label = tk.Label(gantt_frame, text="Activity:")

activity_label.grid(row=0, column=0)



# Combobox for Activity selection

variable_activity = tk.StringVar()

activity_combobox = tk.ttk.Combobox(gantt_frame,
textvariable=variable_activity)

activity_combobox.grid(row=0, column=1)
```

```
# Label for Start Date selection

start_date_label = tk.Label(gantt_frame, text="Start Date:")

start_date_label.grid(row=1, column=0)


# Combobox for Start Date selection

variable_start_date = tk.StringVar()

start_date_combobox = tk.ttk.Combobox(gantt_frame,
textvariable=variable_start_date)

start_date_combobox.grid(row=1, column=1)


# Label for Duration selection

duration_label = tk.Label(gantt_frame, text="Duration:")

duration_label.grid(row=2, column=0)


# Combobox for Duration selection

variable_duration = tk.StringVar()

duration_combobox = tk.ttk.Combobox(gantt_frame,
textvariable=variable_duration)

duration_combobox.grid(row=2, column=1)


# Create a button to pick the project start date

pick_date_button = tk.Button(button_frame, text="Pick Start Date",
command=pick_date)

pick_date_button.pack(side=tk.LEFT)


# Button to plot Gantt chart

plot_button = tk.Button(window, text="Plot Gantt Chart",
command=plot_gantt_chart)
```

```
plot_button.pack(pady=10)


# Start the GUI event loop

window.mainloop()
```

Appendix C

Attribute Selection for The Case Study Dataset of Chapter 3

The result of the correlation analysis showing the significant correlation of each input variable with output (r) and the significant partial correlations (correlation coefficient greater than 0.5) in-between input variables (r') are presented in Table C1. Greedy search strategy involves an initial correlation study among the attributes to establish a rank based on Pearson's correlation coefficient (R) for each attribute connected to the output variable. Then attributes are selected to establish input output relationships following the correlation rank order. The performance of the model has then been assessed based on the correlation coefficient and F statistics of the model formulated in each step of the greedy search. Attributes with has partial correlation with selected attributes modeling are considered in model at the same time and their significance is tested observing the t statistics. Significant attributes are kept in the model, otherwise removed in the next iteration of the greedy search. The first three iterations of the greedy search process is shown below as an example of the search process.

Table C.1: shows all the correlation coefficients for input variable in relation with the output LH.

Variables	Correlation with the output, $\boldsymbol{r}$	Partial Correlation, $\boldsymbol{r'}$ (>0.5)
X35	0.738	X36(0.58), X37(0.65)
X36	0.597	-
X18	0.579	X36(0.84)
X9	0.561	X10(0.63), X18(0.59), X29(0.57), X30(0.55), X31(0.50), X35(0.63), X36(0.69)
X8	0.540	X9(0.73), X10(0.73), X35(0.86), X36(0.53), X37(0.62)
X17	0.480	-
X10	0.470	X11(0.53), X12(0.53), X13(0.55), X16(0.52), X35(0.70), X36(0.53), 37(0.90)
X37	0.447	-
X6	0.411	X7(0.65), X16(0.50), X35(0.55), X37(0.59)

Variables	Correlation with the output, r	Partial Correlation, r' (>0.5)
X5	0.409	X6 (0.87), X7(0.57), X35(0.61), X37(0.37)
X15	0.374	X36(0.62)
X14	0.364	X15(0.75), X16(0.57), X36(0.57)
X7	0.350	X10(0.73), X13(0.51), X16(0.68), X35(0.56), X37(0.91)
X19	0.319	-
X29	0.312	X30(0.91), X31(0.80)
X30	0.285	X31(0.78)
X3	0.276	
X11	0.267	X12(0.80), X13(0.83), X37(0.63)
X31	0.266	-
X12	0.264	X13(0.74), X37(0.61)

Variables	Correlation with the output, r	Partial Correlation, r' (>0.5)
X16	0.247	X37(0.74)
X13	0.242	X34(0.50), X37(0.67)
X2	0.235	-
X1	0.160	-
X4	0.138	-
X34	0.095	-
X21	0.080	-
X20	0.077	X23(0.91), X24(0.91)
X27	0.077	-
X28	0.049	-
X23	0.045	-
X24	0.040	-
X26	0.028	X28(0.60)

Variables	Correlation with the output, $\boldsymbol{r}$	Partial Correlation, $\boldsymbol{r'}$ (>0.5)
X25	0.028	-
X32	0.027	-
X33	0.020	
X22	0.018	-

Iteration 1: the attribute with the highest correlation (rank 1) is selected at the beginning to start with. The attributes which have partial correlation $\boldsymbol{r'}$ (greater than 0.50) with the selected attribute. In this example case, the rank 1 attribute is X35: the weight of the module. The partially correlated attributes with X15 are X36: the quantity of the module, and X37: the length of the module. We check record the regression coefficient of the input-output relationship, $R^1 = 0.771$ and F stat is found significant (9.68E-42<0.05). The regression analysis results for considering all the three attributes are given in Table C2. From Table C2 observing the t stat and P values of each attribute, we can conclude that all the attributes found to be significant therefore kept in the model at the end of the first iteration.

Table C2: Regression analysis result for iteration 1.

Attribute	t Stat	P-value	Significance
X35	10.49	4.75E-21	Significant
X36	5.01	1.12E-06	Significant
X37	-2.01	4.53E-02	Significant

Iteration 2: in iteration 2, the next ranked attribute is selected which is X36 (rank 2). This attribute is already accepted in the previous model. Hence, there is no need for further analysis.

Iteration 3: in iteration 3, ranked 3 attribute is X18 is added to the model and regression analysis is done. The recorded correlation coefficient of the model is $R^3 = 0.794$ and F stat is found significant (4.40E-45<0.05). The regression analysis results for considering all the three attributes are given in Table A3. From Table C3 observing the t stat and P values of each attribute, we found that presence of attribute X18 and X35 is significant however X36 and X37 found to be insignificant. Therefore, we remove the attributes X36 and X37 model and do the analysis again. The recorded correlation coefficient of the model is $R^{3\prime} = 0.793$ and F stat still found significant (5.31E-47<0.05). Since the deleting the attributes does not deteriorate the model's performance ($\Delta R^3 = |R^{3\prime} - R^3| = \ 0.001 < 0.02$), X36 and X37 both are kept removed from the model.

The results of the regression analysis are given in Table C4.

Table C3: Regression analysis result for iteration 3.

Attribute	t Stat	P-value	Significance
X18	4.618888	6.67E-06	Significant
X35	10.71123	1.07E-21	Significant
X36	-0.96218	0.337052	Insignificant
X37	-0.38591	0.699945	Insignificant

Table C4: Regression analysis result for iteration 3 after removing the attribute X36 and X37.

Attribute	t Stat	P-value	Significance
X18	6.97	3.79E-11	Significant
X35	13.05	4.59E-29	Significant

Iteration 4: in iteration 4, ranked 4 attribute is X9 is added to the model and regression analysis is done. The recorded correlation coefficient of the model is $R^4 = 0.793$ and F stat is found significant (8.04E-46<0.05). The regression analysis results do not show significant improvement ($\Delta R^4 = |R^4 - R^3| = 0.000 < 0.02$), so attribute X9 can be removed from the model.

Appendix D

Example of theoretical and observed probability plot (PP plot) on coefficients (productivity contribution) of the MLR model as presented by Branch 1 of the structural steel model.

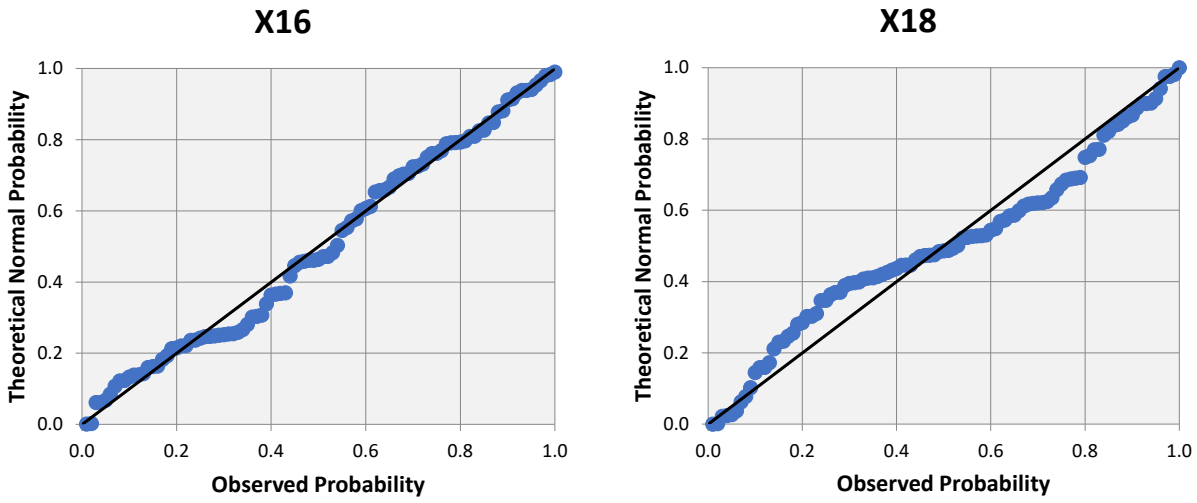


Figure D.1. PP plot of coefficients (productivity contribution) for variables X16 and X18 within the steel fabrication productivity model, as presented by Branch 1 of the enhanced model tree for variance prediction. (Note a perfect correlation in a PP plot indicates a normal distribution underlying the sampled data.)

Appendix E

Sample Classification Results from Enhanced Model Tree

Sample classification performance at some control points of the Enhanced Model Tree algorithm illustrated with attribute 35 is provided in the following table (added in Appendix 3). At the initial stage (base case), when no branching was considered, all 175 instances were used to prepare the MLR model resulting in an R^2 value of 0.77. The tabulated results show classification based on attribute 35 improves the prediction efficiency (value of R^2). The model tree classification (branching) algorithm terminates when no further improvement is observed, subject to the minimum number of instances being retained in each branch (in this case, set at 30). Consequently, no other attributes offer the opportunity for further branching and resulting in an R^2 value greater than 0.77; hence, they are not considered in the model tree classification.

Table E1: Model Tree classification performance at different control points.

Attribute	Branch Decomposition level	Branch Logic	R^2
X35	Base case (no branching)	-	0.77
	Level 1	X35>65794.61 kg	0.86
	Level 1	X35≤65794.61 kg	0.64

	Level 2	X35≤13529.37 kg	0.91
	Level 2	13529.37 kg ≤X35< 65794.61 kg	0.72
	Level 2	65794.61 kg ≤X35≤434349.93 kg	0.87
	Level 2	X35>434349.93 kg	0.93