Applying machine learning techniques to improving truck productivity prediction accuracy at mine sites

by

Chengkai Fan

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

Doctor of Philosophy

in

Mining Engineering

Department of Civil and Environmental Engineering
University of Alberta

Abstract

In oil sands mining, off-the-road trucks play a leading role in transporting bulk materials (ores and waste). The productivity of truck haulage (also referred to as truck productivity), defined as the truck payload per unit time in each truck haulage cycle, is of great interest to the mining industry since truck productivity is directly associated with a mine's overall productivity. Accurate truck productivity prediction is significant for making budget decisions and developing mine planning. However, the current approach used for predicting truck productivity has four major concerns, leading to inaccurate predictions of truck productivity at mine sites. First, the approach (i.e., curve-fitting) is built based on average values. Second, only one input variable (i.e., haul distance) is involved. Third, simple regression method (i.e., least squares) is used to construct prediction models. Fourth, temporal resolutions are not considered in building prediction models.

In response to these concerns, this Ph.D. thesis aims to apply machine learning techniques to improving truck productivity prediction accuracy at mine sites. In particular, this thesis focuses on developing a unified toolkit for truck productivity prediction in oil sands mining, which consists of various machine learning models built based on massive truck haulage data at varying temporal resolutions (e.g., per cycle, hour, day, week, and month). These machine learning algorithms were employed to train complex relationships between truck productivity and multiple input variables, analyze the contributions of input variables to the model output, investigate the effect of temporal resolutions on establishing prediction models, and design a unified graphical user interface (GUI).

The results showed that Gaussian mixture modeling (GMM) efficiently clustered massive truck haulage data into three subgroups (i.e., low, medium, and high truck productivity) and significantly improved the model accuracy. For example, a multiple linear regression (MLR) model reached a coefficient of determination (R^2) of 75% based on GMM analysis, which was much higher than the MLR model (23%) before clustering. After that, nonlinear algorithms were used to build more complex truck productivity prediction models. The results presented that the tree-based ensemble models performed better than single models in predicting truck productivity. Also, the Bayesian regularized neural network (BRNN) model outperformed the back propagation neural network (BPNN) and extreme learning machine (ELM) models. For these machine learning models without considering temporal resolutions, haul distance contributed the most in constructing linear and nonlinear relationships. When involving temporal resolutions, the nonlinear relationship between inputs and truck productivity progressively diminished with decreasing temporal resolutions (i.e., from hourly to monthly). Regardless of the temporal resolutions, the three most influential input variables were haul distance, empty speed, and ambient temperature. In addition, mining engineers can make more accurate predictions of truck productivity at the weekly resolution compared with other resolutions. The feature importance of the four weather-related input variables increased as decreasing temporal resolutions. Extreme weather, such as extreme wind speed, precipitation, and relative humidity, had a certain effect on truck-shovel allocation at mine sites. Finally, a unified GUI was designed and developed for the first time to predict truck productivity at varying temporal resolutions.

Overall, this thesis developed a unified toolkit to improve truck productivity prediction in oil sands mining. The findings will help mine management better understand and forecast truck productivity for hauling efficiency improvement, strategic decision-making, and cost reductions in oil sands mining and other mine sites using truck haulage.

Preface

This thesis is an original work by Chengkai Fan, which applied machine learning techniques to develop a toolkit for improving truck productivity prediction at mine sites. This thesis is based on five journal papers that have been published or submitted for consideration towards publication.

Chapter 2 of this thesis has been published as **C. Fan**, N. Zhang, B. Jiang, W.V. Liu, Preprocessing large datasets using Gaussian mixture modeling to improve prediction accuracy of truck productivity at mine sites, *Archives of Mining Sciences*. © Committee of Mining, Polish Academy of Sciences. 67 (2022) 661-680.

Chapter 3 of this thesis has been published as **C. Fan**, N. Zhang, B. Jiang, W.V. Liu, Prediction of truck productivity at mine sites using tree-based ensemble models combined with Gaussian mixture modeling, *International Journal of Mining, Reclamation and Environment*. © Taylor & Francis. 37 (2023) 66-86.

Chapter 4 of this thesis has been published as **C. Fan**, N. Zhang, B. Jiang, W.V. Liu, Weighted ensembles of artificial neural networks based on Gaussian mixture modeling for truck productivity prediction at open-pit mines, *Mining, Metallurgy & Exploration*. © Springer. 40 (2023) 583-598.

Chapter 5 of thesis has been summitted for peer review as **C. Fan**, N. Zhang, B. Jiang, W.V. Liu, Improved extreme machine learning for rapid estimation of mining truck cycle time based on feature selection and unsupervised clustering techniques, *Expert Systems with Applications*. © Elsevier. (2023). (Under review)

Chapter 6 of this thesis has been submitted for peer review as **C. Fan**, C.B. Arachchilage, N. Zhang, B. Jiang, W.V. Liu, Interpretable data-driven models for assessing truck productivity in open-pit mining under real-site weather conditions with varying temporal resolutions, *Journal of Mining, Reclamation and Environment*. © Taylor & Francis. (With editor)

In this thesis, my work includes conceptualization, methodology, coding, modeling, data analysis, writing, review, and editing. Dr. Wei Victor Liu was my academic supervisor, who contributed to conceptualization, supervision, resources, review, and editing. Dr. Bei Jiang was my academic co-supervisor, who was involved in conceptualization, supervision, review, and editing. Na Zhang assisted in coding in Chapters 2 and 3 and was involved in review and editing in Chapters 4 and 5. Chathuranga Balasooriya Arachchilage assisted in coding and review in Chapter 6.

Dedication

This thesis is dedicated to my mom Qunsong Lu, my dad Hengsheng Fan,

and the rest of my family.

I love you all dearly.

Acknowledgement

First and foremost, I would like to extend my most sincere thanks to my advisor Dr. Wei Victor Liu and co-advisor Dr. Bei Jiang for their invaluable and generous help during my Ph.D. research. In addition to academic guidance, my two advisors have provided me with great mental support. During the pandemic of COVID-19, I overcame frustration and went through demoralizing times with the selfless care from my advisors. I am incredibly proud to have these two advisors, whose profound knowledge, rich experience, and full enthusiasm will continue to enlighten me to move forward in my academic career.

Next, I would like to express my gratitude to my committee members, Dr. Karim El-Basyouny, Dr. Leila Hashemian, Dr. Yong Li, and Dr. Bo Zhang, for their insightful comments and advice on my research and their endorsement of my Ph.D. candidacy exam.

My appreciation goes to Dr. Frank Fan from the industrial collaboration project and Na Zhang and Dr. Rui Qin from the University of Alberta for their technical help in this Ph.D. research. Dr. Frank Fan assisted us in obtaining massive real-site truck haulage data and answered technique questions about the practical engineering sites. In addition, Dr. Frank Fan often organized workshops for our research team and site engineers, which offered a communication platform to report my research progress and inspire me by the site experience. Na Zhang helped me tremendously in my Ph.D. research, who guided me in R language programming and machine learning applications. I also thank Dr. Qin Rui for his thoughtful guidance in Python language programming.

My colleagues Dr. Linping Wu, Dr. Guangping Huang, Dr. Huawei Xu, Dr. Omer Ugurlu, Dr. Shaosen Ma, Dr. Jun Wang, Dr. Lifeng Xu, Yunting Guo, Jian Zhao, Chathuranga Balasooriya Arachchilage, Zhiqiang Feng, Kourosh Gholami, and Hanzhi Yang are deeply appreciated. I sincerely appreciate my former advisor Dr. Qi Li at the Chinese Academy of Sciences, who encouraged me to leave my comfort zone and pursue my Ph.D. at an overseas university. Moreover, I would like to express my gratitude to my friends in Canada and outside of Canada.

I would like to thank the Government of Alberta and my department for recognizing my academic achievement and providing me with scholarships. These scholarships are one of the critical pillars that encourage and guarantee my full dedication to research.

Finally, I deeply appreciate my parents and grandparents for their understanding, encouragement, and love. I have not visited my family for four years because of repeated epidemics and intense research tasks. Through video calls, I feel their ongoing care and support all the time. I would like to express my big “thank you” to my roommate, Jiahao Jian, for his strong support in my daily life. I also thank my fluffy kids, my cat Frisbee and dog Caicai, for bringing me spiritual comfort and joy.

Table of contents

List of tables

List of figures

Chapter 1. Introduction

1.1. Research background

Oil sands mining is a vital pillar of Canada's national economy (Stringham, 2012). By 2035, it will provide more than 905,000 jobs and contribute $2.1 trillion to federal revenues (Honarvar et al., 2011b). In oil sands mining, off-the-road truck haulage is the dominant bulk material handling method for transporting ores and wastes (Ma et al., 2021). The productivity of truck haulage (or referred to as "truck productivity"), is directly related to a mine's overall productivity (Alarie & Gamache, 2002), which significantly affects mine planning (e.g., truck-shovel scheduling, fleet sizing, and budget decision), production, income, and expenditure (Chanda & Gardiner, 2010; Upadhyay et al., 2020).

To predict truck productivity, mining engineers usually adopt a curve-fitting approach based on historical truck haulage data and then continue extrapolation from the fitted curve (Cervantes et al., 2019). As shown in Figure 1.1, the red dots are first obtained by taking the average truck productivity in increments of a specific haul distance interval (e.g., 0.2 km). After that, a curve fitting is conducted based on these red dots to establish a simple prediction model (i.e., the fitted curve) using the least square approach. Finally, this prediction model is employed to predict or extrapolate truck productivity in the future when longer haul distance (e.g., > 18 km) occurs as mining faces advance. This curve-fitting approach is simple and easy to implement, which has been extensively promoted in Alberta's oil sands mines for strategic planning purposes (Obaia, 2020). However, there are four concerns with using this curve to predict truck productivity.

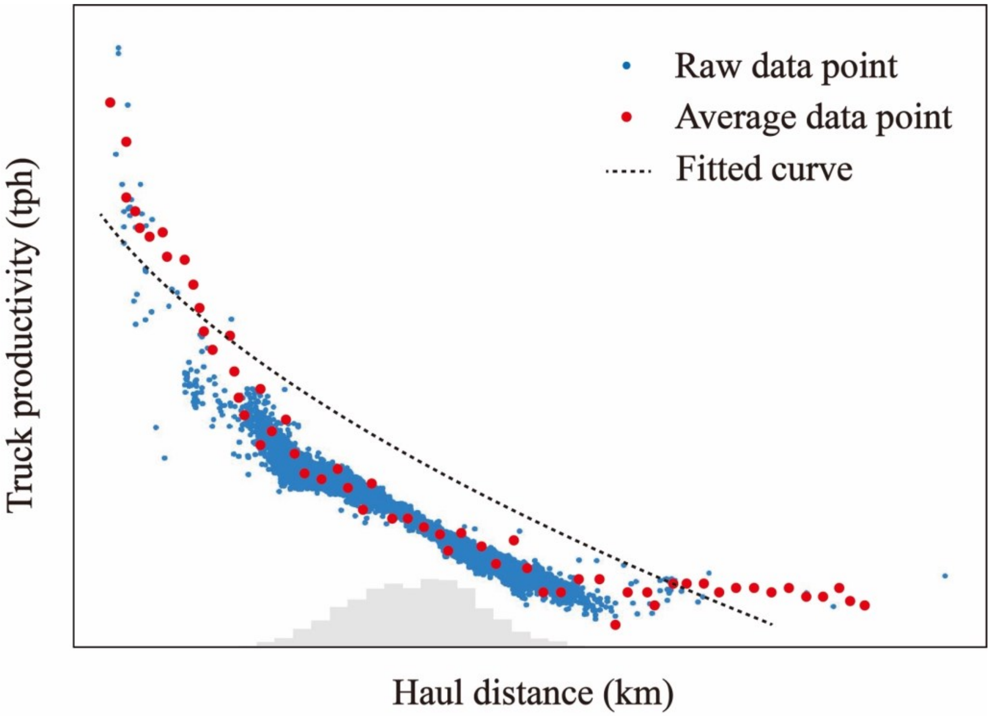


Figure 1.1 A fitted curve (dashed line) of truck productivity created by the local mining company using a curve-fitting approach based on average truck haulage data (red dots).

First, average data may have a smooth effect and lose the variability of individual truck cycles, which potentially cause misleading results (Wackerly et al., 2014). At mine sites, individual truck cycles may vary significantly due to changes in truck haulage processes, such as truck type, haul route, running speed, and real-site weather (Chanda & Gardiner, 2010; Schexnayder et al., 1999). Second, only one input variable (i.e., haul distance) is considered in modeling, which may lead to inaccurate truck productivity prediction models. This is because truck productivity can also be affected by variables associated with operating mine sites (Chanda & Gardiner, 2010). For example, truck speed determines the length of cycle time to affect truck productivity (Schexnayder et al., 1999). Likewise, Ma et al. (2023) reported that a rise in ambient temperature (e.g., from 20 °C to 40 °C) induced an increase in truck tire temperature (e.g., from 54 °C to 69 °C), which caused tire fatigue damage, thus influencing truck cycle time and truck productivity. Third, the curve-fitting approach is limited to building a simple regression between one input (haul distance) and one output (truck productivity). However, if multiple input variables are incorporated, there are potentially more complex relationships between truck productivity and its input variables (Chanda & Gardiner, 2010). These complex relationships require more robust regression methods to build rather than a simple regression (i.e., fitted curve). Finally, the temporal scales (or resolutions) of real-site weather conditions are not considered in the current method. For instance, according to Environmental Canada (MEP, 2023), the maximum precipitation over a week (e.g., 85.30 mm) can have a more substantial impact on road conditions and driving habits than an hour (e.g., 14.10 mm) (Xing et al., 2019). To encapsulate, it may be inaccurate to predict and extrapolate truck productivity only from a simple regression based on the average values of truck productivity and haul distance. Due to the individual features, multiple influential input variables, potential complex relationships, and temporal effects, this curve-fitting approach is no longer appropriate for establishing truck productivity prediction models at mine sites. Therefore, additional solutions are urgently required to provide an accurate forecast of truck productivity for mining companies.

To achieve accurate predictions, data-driven machine learning techniques have attracted increasing attention in recent years (Ahmed et al., 2020; Hyder et al., 2019; Pao, 2008; Perai et al., 2010). Machine learning refers to data-driven analytical algorithms that possesses three major advantages. First, it can efficiently deal with massive amounts of data (Kocheturov et al., 2019). In this work, historical data for the past six years (2016-2021) are available from two management systems, including the Caterpillar's Vital Information Management System (Siami-Irdemoosa & Dindarloo, 2015) and Environmental Canada (MEP, 2023). For example, there are approximately 300,000 data points in 2019 alone. It is challenging to process these data without using machine learning (Al-Jarrah et al., 2015; Pu et al., 2019). Second, machine learning can build complex relationships (i.e., prediction models) between multiple input and output variables (Fei et al., 2020; Tsanas & Xifara, 2012). For instance, there are potentially multiple influential variables affecting truck productivity, such as haul distance, running speed, truck types, destinations, ambient temperature, wind speed, and precipitation (Cervantes et al., 2019; Schexnayder et al., 1999; Sun et al., 2018). Finally, a toolkit can be developed based on machine learning models to facilitate direct use by mining engineers for forecasting truck productivity more easily and quickly. This can alleviate the need for complex modeling analysis and intensive computation (Djandja et al., 2023).

Machine learning provides mining companies with a new solution to forecast truck productivity for hauling efficiency improvement, strategic decision making, and cost reductions at mine sites. Despites its significance, no studies have been conducted to build truck productivity prediction models using machine learning as an alternative to the curve-fitting approach. Therefore, it is of great interest to employ machine learning to handle massive real-site data, establishing complex relationships, and develop a unified toolkit for better understanding and predict truck productivity at mine sites.

1.2. Literature review

1.2.1. Definition of Truck productivity

In open-pit mining, truck productivity (unit: tonnes per hour, tph) is a measure of the amount of ores (unit: tonnes) that can be moved by a mining truck in a given period of time (unit: minutes, mins) (Ercelebi & Bascetin, 2009), which can be formulated as follows:

$$Truck\ productivity\ (tph) = 60\ \times \frac{Payload\ (tonnes)}{Truck\ cycle\ time\ (mins)} \qquad (1\text{-}1)$$

where payload refers to the capacity of a mining truck loaded with ores. Truck cycle time indicates the time it takes for a mining truck to complete a haulage cycle. A haulage cycle (shown in Figure

1.2) usually consists of four stages: loading, hauling, dumping, and returning. In times of high truck volume and low shovel utilization, trucks queue and incur waiting times, including waiting time at shovel, waiting time at dump, and spotting time (spotting time refers to the time that a shovel with ores already has been waiting for a truck to arrive). As a result, truck cycle time (TCT) contains loading time (LT), hauling time (HT), waiting time at dump (WTD), unloading (dumping) time (UT), returning time (RT), spotting time (ST), and waiting time at shovel (WTS), which can be expressed as

$$TCT = LT + HT + WTD + UT + RT + ST + WTS \tag{1-2}$$

For a given truck payload (usually fully loaded), the main reason behind the factors affecting truck productivity is the length of truck cycle time. In other words, the various components of cycle time (e.g., hauling time) directly determine truck productivity (Chanda & Gardiner, 2010; Smith et al., 2000).

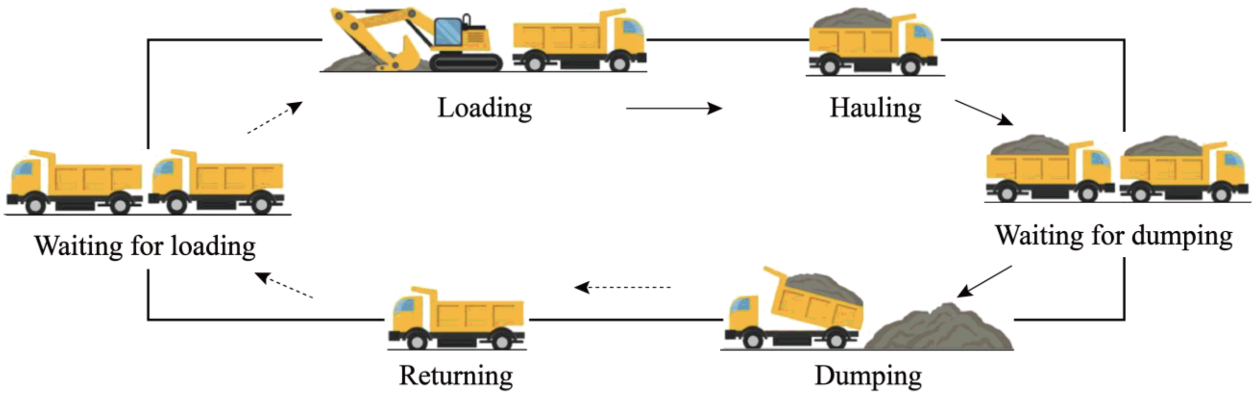


Figure 1.2 A schematic diagram for a truck cycle in open-pit mines (Solid and dashed arrows indicate loaded and unloaded trucks, respectively).

1.2.2. Traditional methods to predict truck productivity

To predict truck productivity, various simulation models and algorithms have been proposed by researchers based on sequential tasks performed by trucks at mine sites (Baek & Choi, 2019). These methods include, but are not limited to, discrete-event simulation models (Moradi Afrapoli et al., 2019), queuing theory (Sembakutti et al., 2017), goal programming (Upadhyay et al., 2020), and stochastic programming (Rimélé et al., 2020). These methods estimate truck productivity by optimizing truck dispatch (Sun et al., 2018). For example, Soofastaei et al. (2016) investigated the effect of payload variance on truck bunching (dispatching problem) and the resulting loss of truck productivity using a discrete-event simulation model. Similarly, Moradi Afrapoli et al. (2019) built a multi-objective model for real-time truck dispatch to maximize truck and shovel productivity. They compared it with a benchmark model and a discrete-event simulation model. Nevertheless, there are problems with these methods because of unexpected events during truck haulage, such as extreme weather and shovel availability reduction. To ensure accurate simulations, these models and algorithms need to be continually updated, resulting in increased time and labor costs (Baek & Choi, 2019).

1.2.3. Basic concept of machine learning

To address the limitations in simulation methods, machine learning (ML) based on historical data has been initiated as a new research direction (Fan et al., 2022, 2023b; Khambra & Shukla, 2023; Zabin et al., 2022). Machine learning is a data science category at the intersection of computer science, math, and statistics that has made tremendous progress in engineering applications over the past two decades (Jordan & Mitchell, 2015). Table 1.1 lists the major developments of machine learning throughout its history (Pu et al., 2019). Machine learning refers to a series of analytical data algorithms that automatically build complex relationships between input and output variables

(Fei et al., 2020), as illustrated in Figure 1.3. First, machine learning models can be trained by capturing the implicit or explicit relationships between existing inputs and outputs in training data using various machine learning algorithms. After that, with the established prediction models, unknown outputs can be predicted based on new inputs. In other words, the model functions between inputs and outputs are described based on training data and machine learning algorithms. To build regression models, machine learning algorithms are usually split into two subcategories: supervised learning and unsupervised learning algorithms (Alrfou et al., 2022; Yin et al., 2022). These two subcategories will be briefly introduced and reviewed in this section.

Table 1.1 Major developments of machine learning throughout its history.

Research progress	Main purpose
McCulloch and Pitts (1943)	Proposed a hierarchical model of a neural network.
Rosenblatt (1958)	Put forward the concept of ‘‘Perceptron”; designed the first computer neural network.
Hubel and Wiesel (1962)	Put forward the famous ‘‘Hubel-Wiesel biological visual model” from research on the cerebral cortex of cats.
Rumelhart et al. (1986)	Published backpropagation algorithm (BP).
LeCun et al. (1989)	Proposed a prevailing convolutional neural network (CNN) and derived an efficient training method for CNN based on BP algorithm.
Cortes and Vapnik (1995)	Developments of machine learning models like logistic regression, support vector machine, boosting algorithms.
Hinton and Salakhutdinov (2006)	Proposed a deep learning model that utilized a multi-layer neural network to approximate functions.

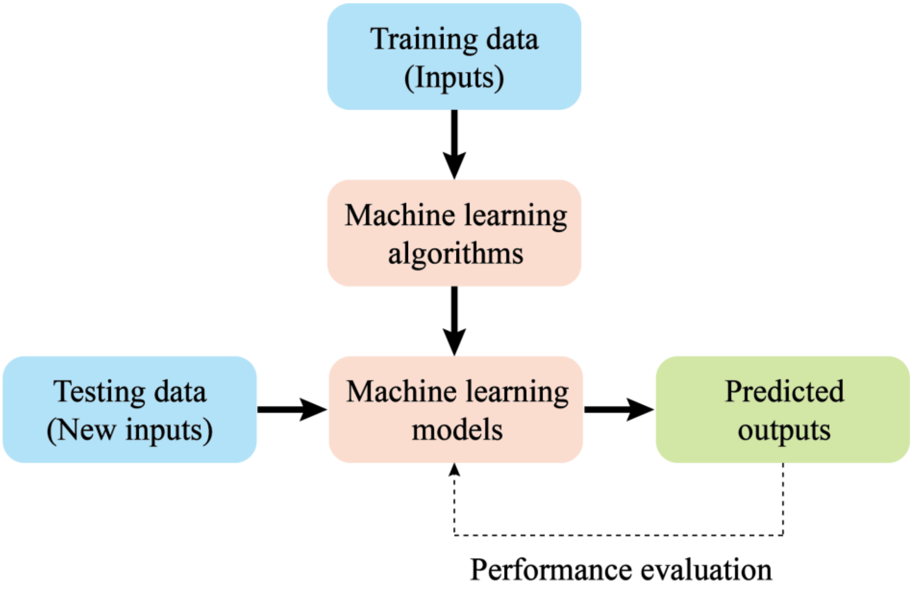


Figure 1.3 A basic concept of machine learning.

1.2.3.1. Supervised learning algorithms and relevant applications

Supervised learning refers to machine learning algorithms that rely on labeled input and output variables to construct prediction models (David & James, 1987). Supervised learning algorithms usually include, but are not limited to, (1) multiple linear regression (MLR) (Tan et al., 2014); (2) support vector regression (SVR) (Zhong et al., 2019); (3) tree-based algorithms such as decision tree (DT) (Pu et al., 2018), random forest (RF) (Rodriguez-Galiano et al., 2015), adaptive boosting (AdaBoost) (Feng et al., 2020), gradient boosting regression (GBR) (Kaplan et al., 2021), and extreme gradient boosting (XGBoost), and (4) artificial neural networks (ANNs) such as back propagation neural network (BPNN) (Zou et al., 2009), extreme learning machine (ELM) (Pan et al., 2020), and Bayesian regularized neural network (BRNN) (Çetinkaya & Baykan, 2020). These algorithms have been extensively applied in many engineering applications. Their concepts and relevant applications are briefly reviewed below.

(1) MLR is a statistical approach for building prediction models in regression problems because

of its simple calculation and explicit equation (Li et al., 2015). It has been used in many aspects of mining engineering for prediction, such as coal production (Li et al., 2015), blast-induced ground vibration (Saadat et al., 2014), and rock fragmentation (Enayatollahi et al., 2014). For example, Enayatollahi et al. (2014) built an MLR model for predicting rock fragmentation in open-pit mines. The result showed that the coefficient of determination (R^2) of this model was 85%. Similarly, Ghiasi et al. (2016) predicted the number of boulders produced in blasting operations of an open-pit mine using MLR. The results showed that the R^2 and root mean square error values were 89% and 0.19. Nevertheless, MLR is limited to building linear relationships between input and output variables. Nonlinear relationships need robust machine learning algorithms (e.g., SVR, DT, RF, AdaBoost, GBR, XGBoost, and ANNs) to establish.

(2) SVR was firstly proposed by Vapnik and Lerner (1963) and developed to be one of tools with strong potential in data-driven areas (Cortes & Vapnik, 1995; Rodriguez-Galiano et al., 2015). SVR is widely used in mining engineering, such as cost estimation (Nourali & Osanloo, 2019), blasting operations (Hasanipanah et al., 2017), drilling risk evaluation (Liang et al., 2019), and mining subsidence (Li et al., 2014). These previous studies have demonstrated the great potential of SVR in capturing complex input-output relationships and building accurate prediction models. For instance, Li et al. (2021) involved 19 input variables to establish a prediction model of blasting fragmentation size using SVR. The results showed that the R^2 and mean square error values for the testing dataset were 83.53% and 0.0035. Akin to Li et al. (2021), Huang and Xue (2022) proposed an SVR model to predict flyrock distance based on six input variables and 240 blasting events. The model showed a high prediction accuracy, of which the R^2 attained 92.94%.

(3) DT, RF, AdaBoost, GBR, and XGBoost are tree-based algorithms for dealing with regression

and classification problems (Jun & Cheng, 2017; Liu et al., 2023; Nasir Amin et al., 2023). DT is a single-tree model, whereas RF, AdaBoost, GBR, and XGBoost are ensemble learning algorithms integrating numerous DT (Erdal, 2013). RF uses a bagging method to overcome the shortcomings of high variance and overfitting issues in DT (Ohadi et al., 2020). Unlike RF, AdaBoost, GBR, and XGBoost utilize a boosting rather than a bagging method (Aydin & Iban, 2023). These tree-based algorithms have been applied in the mining industry for predictions such as rockburst (Pu et al., 2018), mine subsidence (Lee & Park, 2013), ore sorting (Lessard et al., 2014), and rock strength assessment (Liang et al., 2016). Moreover, ensemble learning algorithms that integrate numerous DTs can usually enhance model predictability (Dou et al., 2019). For example, Rodriguez-Galiano et al. (2015) constructed an RF model (integrating 50 DTs) to forecast mineral prospectivity at mine sites. The study showed that the prediction accuracy of the RF model was about 39% higher than that of the single DT model. Likewise, Liang et al. (2020) compared the accuracy of a GBR model (integrating 1200 DTs) and a DT model in predicting hard rock pillar stability. The results showed that the accuracy of the GBR model was 83.1%, whereas the accuracy of the DT model was 59.2%.

(4) BPNN, ELM, and BRNN are three well-known ANN algorithms. The difference between the three algorithms lies in the setting of the weights among the neurons (Goodarzi et al., 2010; Liu et al., 2019). BPNN assumes that the weights are fixed values; BRNN assumes that the weights are random variables and follow Gaussian distributions, while ELM treats the weights as some random values. Therefore, ELM does not require intense computation compared with BPNN and BRNN (Fikret Kurnaz & Kaya, 2018; Zhang et al., 2016). These ANNs have been extensively applied to many aspects of mining engineering because of their strong ability to map nonlinear relationships between input and output variables, thus providing robust

predictions (Nguyen et al., 2020; Thai et al., 2021; Trivedi et al., 2014; Xue et al., 2020). For example, Trivedi et al. (2014) built a BPNN model to predict the distance covered by blast-induced flyrock in limestone mines. The results showed that the R^2 of the BPNN model was 98.3%, whereas it was 81.5% in the case of a statistical multiple regression model. Likewise, Xue et al. (2020) established five ML models, including a BPNN model and an ELM model, for predicting rockburst intensity in deeply buried areas. The study showed that the proposed ELM model had a higher average accuracy of 97.57% than the BPNN model (62.13%). Close to the research by Trivedi et al. (2014) and Xue et al. (2020), Nguyen et al. (2020) proposed a BRNN model to forecast air-blast overpressure induced by blasting at open-pit coal mines. The study showed that the BRNN model performed well in predicting overpressure, with an R^2 of 93.6%.

According to the literature review, it is promising to apply supervised machine learning algorithms to build prediction models. Nevertheless, no studies have been found using these algorithms to forecast truck productivity in open-pit mining. Thus, it will be worth developing truck productivity prediction models using machine learning for better mine planning and decision-making.

1.2.3.2. Unsupervised learning algorithms and relevant applications

Unlike supervised learning, unsupervised learning is usually adopted to analyze unlabeled data for disclosing hidden data groups or patterns without human intervention (Usama et al., 2019; Xu & Saleh, 2021). Clustering is a typical unsupervised learning algorithm that assigns each data point into a specific class and extract potential hidden patterns (Alam & Paul, 2020). In clustering, K-means and Gaussian mixture modeling (GMM) are two extensively applied unsupervised methods because they are easy to implement and efficient for dealing with massive data (Capó et al., 2017). They identify several classes from a data population and assign data points with more similarities to the same subgroup (Fan et al., 2022). The key distinction between the two methods lies in the principles of assigning data points to classes. K-means assumes that each data point falls into the specific class where the centroid is closest to it. The centroid is updated iteratively until the squared distance sum between the centroid and each data point is minimized (Liu et al., 2020). Unlike K-means, GMM is known as a probability-based clustering approach that assigns data points to a specific class when they have the maximum class posterior probability (Grün & Leisch, 2007). These two methods have been shown to effectively deal with massive data and enhance the model predictability (Liu et al., 2020). For example, Liu et al. (2020) applied K-means to classify the single-crystal superalloy creeping data and developed prediction models of creep rupture life. K-means recognized eight homogeneous classes that were intimately linked to the creep mechanisms, which improved the model accuracy of creep rupture life with an increase in R^2 from 71.02% to 91.76%. Similarly, Ni et al. (2020) adopted GMM to identified two classes (low and high) from massive hydrological data and built an XGBoost model for forecasting monthly streamflow. The results presented that the model's performance increased by about 11% based on GMM analysis. Lu et al. (2019) also classified building heating data using GMM to identify sub-patterns (including six classes). Then, they trained models for predicting the hourly heating load, whose performance was enhanced by approximately 20% because of GMM analysis.

However, based on the current literature, no studies have reported the application of unsupervised clustering methods to preprocess massive data obtained from oil sands mines; it is still unknown if these methods can be used to improve the model predictability of truck productivity at mine sites.

1.2.4. Influential parameters affecting truck productivity

According to the literature review, influential parameters affecting truck productivity often divided into two categories: truck haulage-related and weather-related variables (Baek & Choi, 2020; Choi et al., 2021; Jung & Choi, 2021). These variables include, but are not limited to, distance (e.g., haul distance), running speed (e.g., empty speed), haul routes, truck and shovel numbers, ambient temperature, precipitation, wind speed, and relative humidity. These variables are observed by site engineers and associated with truck cycle time, thus influencing truck productivity (Chanda & Gardiner, 2010). For instance, Cervantes et al. (2019) reported that mining companies often plotted a fitted line between haul distance and truck productivity because the increase in haul distance directly affects the increase in truck cycle time, thereby reducing truck productivity. According to Schexnayder et al. (1999), empty speed determined the travel time from dumping sites to loading sites, affecting truck productivity. Also, Ma et al. (2021) reported that high ambient temperature could enhance tire temperatures and cause rubber failure of the off-the-road tire at mine sites, thus affecting truck productivity and ore production. Moreover, Asamer and Reinthaler (2010) analyzed the data from U.S. highway administrations. They demonstrated that heavy precipitation led to a 35% reduction in running speed, thus increasing travel time. Similarly, relative humidity and wind speed may interfere with road conditions (e.g., wetness or dryness) and driver's vision (Choi & Nieto, 2011; Silion & Foşalău, 2014), influencing driving habits and travel time.

These variables can be obtained from two sources: Vital Information Management System (VIMS) (Siami-Irdemoosa & Dindarloo, 2015) and Environmental Canada (MEP, 2023). These two data management systems are rich in information related to truck haulage and weather. Nevertheless, these data have never been used as training datasets for truck productivity prediction models.

1.3. Research objectives

The overall objective of this thesis aims to apply machine learning techniques to improving truck productivity prediction accuracy at mine sites. In particular, this thesis focuses on developing a unified toolkit for truck productivity prediction in oil sands mining based on massive historical data with varying temporal resolutions. To achieve this overall objective, five sub-objectives are proposed as follows:

(1) To understand and preprocess massive truck haulage data (weather data included) at mine sites.

(2) To build linear and nonlinear prediction models of truck productivity using machine learning.

(3) To compare the effect of clustering techniques on improving models' prediction accuracy.

(4) To investigate the effect of temporal resolutions on truck productivity predictive modeling.

(5) To develop a unified toolkit for predicting truck productivity at varying temporal resolutions.

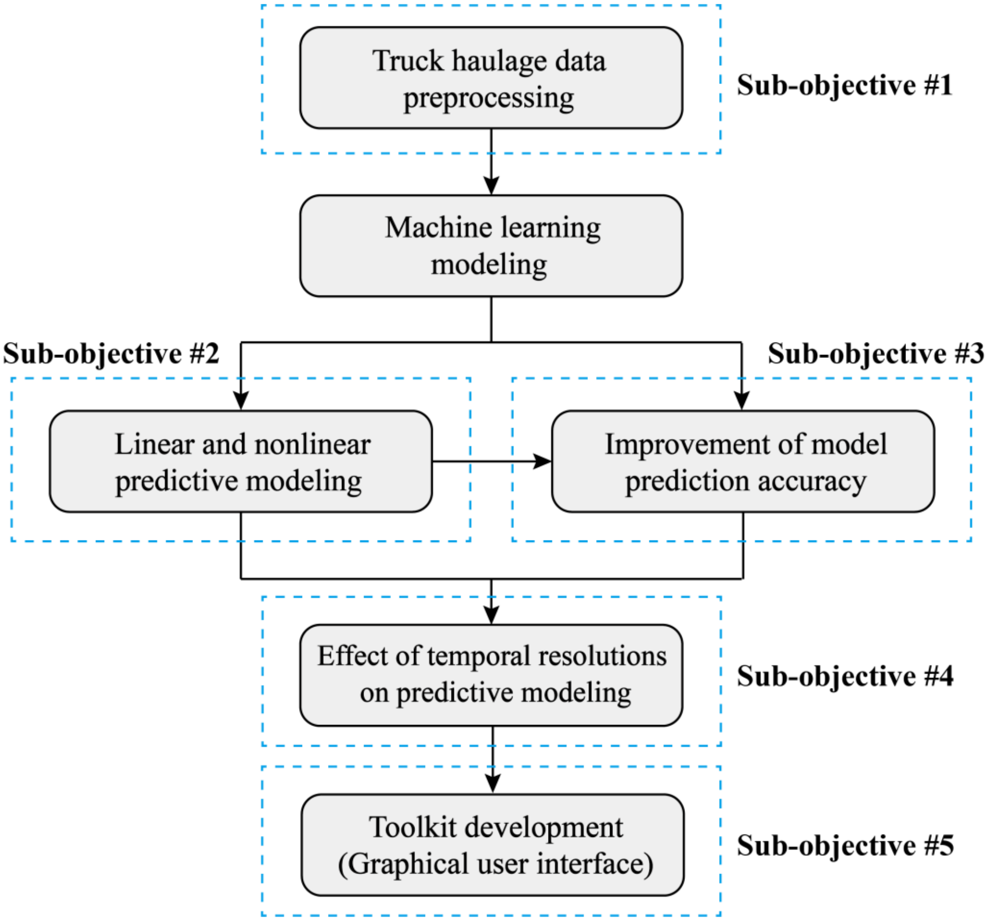

Truck haulage data preprocessing
Sub-objective #1
Machine learning modeling
Sub-objective #2
Sub-objective #3
Linear and nonlinear predictive modeling
Improvement of model prediction accuracy
Effect of temporal resolutions on predictive modeling
Sub-objective #4
Toolkit development (Graphical user interface)
Sub-objective #5

Figure 1.4 An overall flowchart illustrating the connection between the five sub-objectives.

The connection between the five sub-objectives is illustrated in Figure 1.4. As shown in Figure 1.4, truck haulage data may originate from various sources, such as sensor networks (Gui et al., 2011), remote sensing (Gu et al., 2010), and wireless communication (Sabniveesu et al., 2015). Regardless of the source, truck haulage data at mine sites are unique and massive. From the view of statistics, massive data are usually preprocessed by clustering techniques (Dindarloo & Siami-Irdemoosa, 2017; Shahin et al., 2004). Therefore, the sub-objective #1 is conducted to preprocess truck haulage data using clustering methods for potentially improving model predictability. Based on the clustering results, the sub-objective #2 performs linear and nonlinear (i.e., MLR, DT, RF, GBR, XGBoost, and ANNs) between multiple input variables and truck productivity because these algorithms are commonly used and can provide accurate prediction models for end users (Chanda & Gardiner, 2010; Kueh, 2021). Since the effect of different clustering techniques may vary in improving prediction accuracy, a comparative study of these clustering techniques is necessary. Therefore, the sub-objective #3 compares two widely applied clustering techniques, K-means and GMM, to explore their effects on enhancing model accuracy. In sub-objectives #2 and #3, the truck productivity prediction models are built based on massive data from numerous individual truck cycles, but temporal resolutions have been taken into account in these models. For example, weekly precipitation may have a more substantial influence on truck productivity than hourly precipitation (Xing et al., 2019). Thus, the sub-objective #4 investigates the effect of temporal resolutions on establishing truck productivity prediction models in oil sands mining. To facilitate access to the solutions of this Ph.D. research by site engineers and researchers, the sub-objective #5 developed and proposed a unified toolkit (i.e., graphical user interface, GUI) based on the best prediction models established in the sub-objective #4. This GUI can predict truck productivity at varying temporal resolutions, which will be instrumental in making decisions more easily and quickly for mining engineers and researchers.

1.4. Thesis statement and thesis outline

Thesis statement: Machine learning can deal with massive amounts of data at mine sites, establish complex relationships between multiple input and output variables, and is the basis of developing a unified toolkit for facilitating direct use by mining engineers. Machine learning can be substituted for the current curve-fitting approach used by mining companies to solve the problems of simple regression, single variable, and temporal effects.

As shown in Figure 1.5, this thesis includes seven chapters, presenting in a paper-based format. The summary of each chapter is listed as follows.

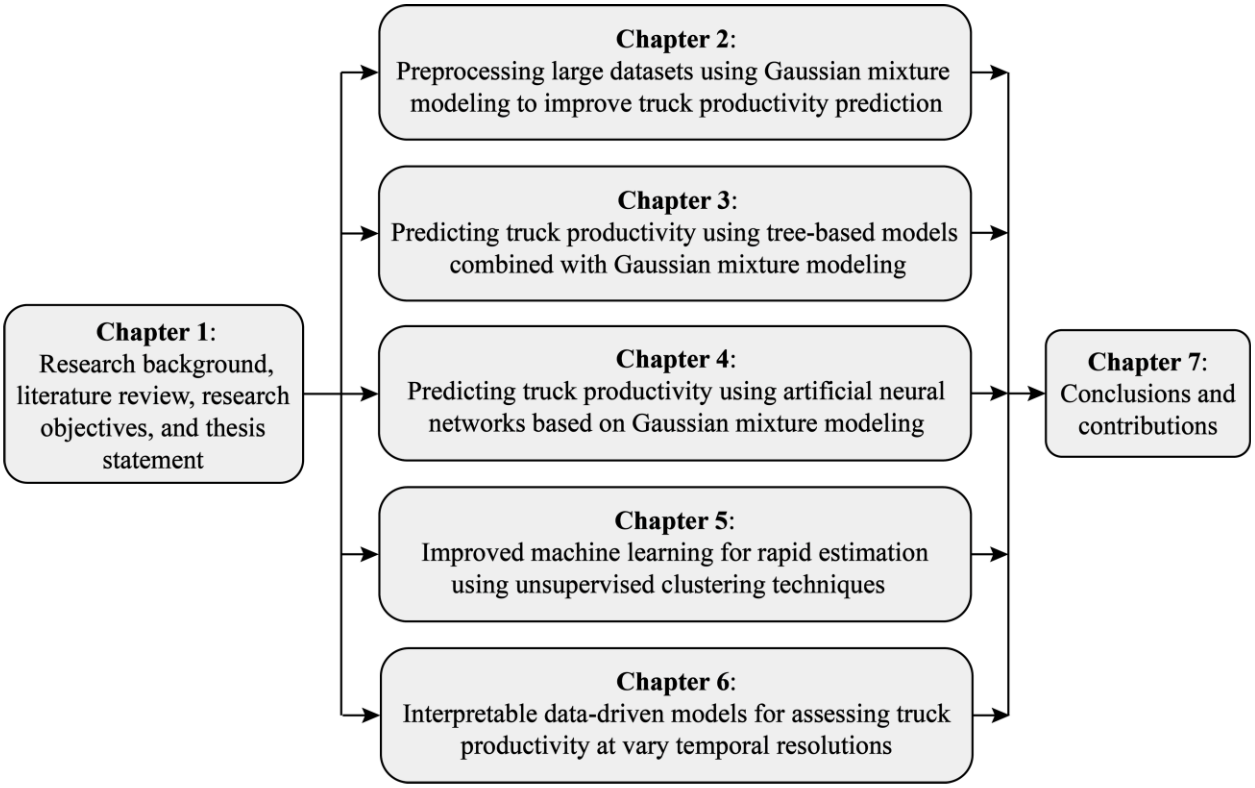

Chapter 1: Research background, literature review, research objectives, and thesis statement
Chapter 2: Preprocessing large datasets using Gaussian mixture modeling to improve truck productivity prediction
Chapter 3: Predicting truck productivity using tree-based models combined with Gaussian mixture modeling
Chapter 4: Predicting truck productivity using artificial neural networks based on Gaussian mixture modeling
Chapter 5: Improved machine learning for rapid estimation using unsupervised clustering techniques
Chapter 6: Interpretable data-driven models for assessing truck productivity at vary temporal resolutions
Chapter 7: Conclusions and contributions

Figure 1.5 An overall flowchart showing the outline of this thesis.

Chapter 1 introduced the research background on the prediction of truck productivity at mine sites, highlights the current research problems, and describes the research objectives. To predict truck productivity, mining engineers often use a curve-fitting approach based on historical truck haulage data and then continue extrapolation from the fitted curve. However, there are four concerns with using this curve to forecast truck productivity: (1) insufficient information on the averaged data; (2) single input variable; (3) simple regression, and (4) lack of consideration on temporal effects. To address these concerns, four improvements are proposed in this research: using individual truck cycles, involving multiple influential input variables, building potential complex relationships, and considering temporal effects. Therefore, additional solutions are urgently required to provide an accurate forecast of truck productivity. Machine learning can handle massive amounts of data and establish complex relationships between multiple input and output variables. This research aims to improve truck productivity prediction in oil sands mining using machine learning.

Chapter 2 aimed to handle massive data of truck haulage using Gaussian mixture modeling (GMM) for developing a novel and accurate prediction model of truck productivity. In this chapter, a large dataset of truck haulage collected at operating mine sites was clustered by GMM into three latent classes before the prediction model was built. The labels of these latent classes generated a latent variable. Two multiple linear regression (MLR) models were then constructed, including the ordinary-MLR (O-MLR) and the GMM-MLR models. The GMM-MLR model incorporated the observed input variables and a latent variable in the form of interaction terms. The O-MLR model was the baseline model and did not involve the latent variable. The GMM-MLR model performed considerably better than the O-MLR model in predicting truck productivity. The interaction terms quantitatively measured the differences in how the observed input variables affected truck productivity in three classes (high, medium, and low truck productivity). The haul distance was the most crucial input variable in the GMM-MLR model. This study provides new insights into handling massive data at mine sites and a more accurate prediction model for truck productivity.

Chapter 3 developed prediction models using tree-based ensemble learning algorithms based on the truck haulage dataset to forecast truck productivity. In Chapter 2, GMM was used to preprocess the massive truck haulage data and constructed a linear prediction model. In this chapter, two nonlinear tree-based ensemble learning algorithms, including random forest (RF) and gradient boosting regression (GBR), were proposed in combination with GMM to train prediction models. GMM was adopted as a clustering technique to extract a latent variable from the training dataset.

MLR and decision tree (DT) as single learning algorithms were used to construct prediction models to be compared with the tree-based ensemble models. The results showed that the tree-based ensemble models performed better than single models in predicting truck productivity with and without GMM clustering. Moreover, GMM significantly increased the predictability of truck productivity prediction models by considering the latent variable. From the relative importance analysis, haul distance was the most influential factor among the observed input variables. Finally, the GMM-RF and GMM-GBR models with high accuracy were the proposed models for predicting truck productivity at mine sites.

Chapter 4 established prediction models between truck productivity and its input variables based on the real-site dataset using artificial neural networks (ANNs). In addition to the nonlinear tree-based algorithms in Chapter 3, ANNs are also well-known nonlinear algorithms used to construct regression models. For the first time, this chapter used a back propagation neural network (BPNN), an extreme learning machine (ELM), and a Bayesian regularized neural network (BRNN) coupled with GMM to deal with the complex truck haulage data and build three weighted ensemble (WE) models to predict truck productivity. DT, RF, GBM, and extreme gradient boosting (XGBoost) were used to build models to be compared with the weighted ensemble models. The results showed that the WE-BRNN had a higher accuracy than the WE-BPNN and WE-ELM models. The proposed weighted ensemble models performed better than the benchmark models in predicting truck productivity, indicating that a weighted ensemble approach based on the GMM analysis significantly improved the model accuracy. Based on the relative importance analysis, haul distance was the most crucial input variable for predicting truck productivity. This study provides a new approach to predicting truck productivity, which will help mining companies make sound budget decisions and improve mine planning.

Chapter 5 compared the effect of two commonly used clustering techniques on enhancing models' prediction accuracy. GMM is not the only unsupervised clustering method; other methods (e.g., K-means) may also improve the model predictability by preprocessing massive data. In this chapter, three extreme machine learning algorithms, including ELM, extremely randomized trees (ERT), and XGBoost, were employed to train prediction models because they are known for providing fast computations. To further decrease computational costs and improve model accuracy, this chapter conducted a comparative study of two unsupervised clustering techniques: K-means and GMM. The results showed that XGBoost outperformed ELM and ERT in predicting mining truck cycle time (equivalent to predicting truck productivity). GMM improved the model accuracy significantly, but K-means could not increase the model predictability.

Chapter 6 investigated the effect of temporal resolutions on building truck productivity prediction models. From Chapter 2 to Chapter 5, the predictive model is built on a large amount of individual trucking data while ignoring the effect of temporal resolution. This chapter constructed prediction models of truck productivity incorporating real-site weather conditions with varying temporal resolutions (i.e., hourly, daily, weekly, and monthly) for the first time. After that, the prediction models were combined with Shapley Additive exPlanations (SHAP) to offer quantitative and qualitative analysis for each input variable's effect on the model outputs. The results presented that the nonlinear relationship between input variables and truck productivity progressively diminished with decreasing temporal resolutions (i.e., from hourly to monthly). Mining engineers can make more accurate forecasts of truck productivity at the weekly resolution compared with other resolutions. Regardless of the temporal resolutions, the three most influential input parameters were haul distance, empty speed, and ambient temperature. Extreme weather, such as strong wind speed, heavy precipitation, and extreme relative humidity, had a certain effect on truck-shovel allocation at mine sites. Meanwhile, a unified graphical user interface was developed to predict hourly, daily, weekly, and monthly truck productivity in open-pit mining.

Chapter 7 enumerated the main research conclusions, contributions, limitations of this thesis. Recommendations for future research are also discussed.

Chapter 2. Preprocessing large datasets using Gaussian mixture modeling to improve prediction accuracy of truck productivity at mine sites

This chapter has been published as **C. Fan**, N. Zhang, B. Jiang, W.V. Liu, Preprocessing large datasets using Gaussian mixture modeling to improve prediction accuracy of truck productivity at mine sites, *Archives of Mining Sciences*. © Committee of Mining, Polish Academy of Sciences. 67 (2022) 661-680.

Nomenclatures

BIC	Bayesian information criteria
C	The number of estimated parameters
EM	Expectation-maximization
f_k	Probability density function
GMM	Gaussian mixture modeling
GMM-MLR	Gaussian mixture modeling-based multiple linear regression
k	The *k*th latent classes
K	The number of latent classes
L	Likelihood of a set of data points
LMG	Lindeman, Merenda, and Gold
m	The *m*th input variable
M	The number of input variables
$M!$	Factorial of *M*
MAE	Mean absolute error
MAPE	Mean absolute percentage error
MEMS	Michelin Earthmover Management System

MLR	Multiple linear regression
n	The nth data point
N	The number of data points
O-MLR	Ordinary-multiple linear regression
p	Permutation of input variables
P	Mixture model
R^2	Coefficient of determination
RMSE	Root mean square error
tph	Tonnes per hour
VIMS	Vital Information Management System
x_m	The mth input variable
y	Output variable
$\bar{y}$	Mean value of y
$\hat{y}$	Predicted value of y
β_0	Intercept of the linear function
β_m	Regression coefficient
γ_{nk}	Posterior probability

ϵ	Random error of linear function
θ	Parameter vector of the density function
λ_k	A set of data points that maximize γ_{nk}
μ_k	Mean vector of the density function
π_k	Weight of the *k*th latent class
Σ_k	Covariance matrix
$\varnothing$	Parameter set of the mixture model

2.1. Introduction

Oil sands mining plays a vital role in Canada's economy (Sleep et al., 2018). In 2017 alone, it contributed CAD$13 billion to the national revenues and created more than 228,000 direct and indirect jobs (CAPP, 2018). In oil sands mining, truck haulage is a dominant means of transporting ores and wastes (Katta et al., 2019). The productivity of truck haulage (or referred to as truck productivity), defined as truck payload per unit time in each truck cycle, is directly related to a mine's overall productivity (Alarie & Gamache, 2002). Therefore, it is of great significance to predict truck productivity, which affects a mine's production, planning, income, and expenditure (Alarie & Gamache, 2002; Bartos, 2007).

To predict truck productivity, researchers attempt to establish data-driven prediction models based on historical datasets (Chanda & Gardiner, 2010). The datasets may originate from various sources, such as sensor networks (Gui et al., 2011), remote sensing (Gu et al., 2010), wireless communication (Sabniveesu et al., 2015), a Vital Information Management System (VIMS) (Siami-Irdemoosa & Dindarloo, 2015), and Michelin Earthmover Management System (MEMS) (Zhang et al., 2018). Regardless of the source, datasets at mine sites are usually large. For example, Baek and Choi (2020) obtained two large datasets from limestone quarries, including 16,217 and 16,005 data points, respectively. The datasets were used to build prediction models for morning and afternoon ore production over two months. Likewise, a dataset collected from oil sands mines in this study was even larger, with more than 300,000 data points covering truck haulage information for an entire year.

Large datasets are usually preprocessed by clustering techniques (Dindarloo & Siami-Irdemoosa, 2017; Shahin et al., 2004). Clustering is a data mining technique that assigns each data point into a specific class (Alam & Paul, 2020). In each class, the assigned data points share more similarities than those in the other classes (Yang et al., 2017). Commonly used clustering techniques include K-means (Alam & Paul, 2020), hierarchical clustering (Tu et al., 2021), density-based spatial clustering (Wang & Hamilton, 2005), and Gaussian mixture modelling (GMM) (Santos et al., 2017). Of these, GMM is the superior technique for preprocessing large datasets, showing potential for handling massive amounts of data from mine sites. GMM is a probability distribution-based clustering technique that identifies latent classes from a large dataset (Diaz-Rozo et al., 2020). In GMM, each class is assumed to follow a Gaussian distribution. Together, these classes form a mixture of Gaussian distributions, which are also known as multi-peak Gaussian distributions (Bishop, 2006). According to the central limit theorem (Rice, 1995), large datasets observed in engineering often present multi-peak Gaussian distributions. This applies to truck haulage data obtained from oil sands mines (Cervantes et al., 2019). For instance, in Figure 2.1, the haul distance, ranging from 0 to 10 km, is plotted in a column chart. Each range of haul distance falls under a density ranging from 0 to 0.4. The density refers to the fraction of a range divided by the total size of data. As shown in Figure 2.1, the column can either be described by superimposed density curves of a single Gaussian distribution (Figure 2.1(a)) or a multi-peak Gaussian distribution (Figure 2.1(b)). The multi-peak Gaussian distribution presents two peaks of haul distance, which includes additional information. Relying on these peaks, GMM has the ability to identify latent classes (Ge et al., 2018), thereby increasing model predictability. For example, Lu et al. (2019) used GMM to identify four classes from multi-peak heating load data and then built prediction models separately based on the datasets included in each class. The research showed that the accuracy of prediction models was enhanced by at least 20% based on the identification results. Similar to the research by Lu et al. (2019), Ni et al. (2020) obtained large streamflow datasets with multi-peak Gaussian distributions and used GMM to divide them into several classes.

Each class was then fitted with a single model, and the final prediction was a weighted sum of these models. The results showed that the accuracy of the proposed model for streamflow was improved by about 11% compared with the prediction models built based on the original large datasets. In addition, GMM has the ability to generate latent variables; the latent variable is defined as the labels of classes, which can be involved in modeling to improve prediction accuracy (Lubke & Luningham, 2017; Parsons, 2020). From the above studies, GMM has advantages for in-depth data mining with multi-peak Gaussian distributions (Ye et al., 2019). Thus, GMM may be a more suitable option to improve prediction models because large datasets of truck haulage are usually under multi-peak Gaussian distributions. However, according to the current literature, no research has reported the application of GMM to preprocess large datasets obtained from mine sites; it is still unknown if GMM can be used to improve the model predictability of truck productivity at mine sites.

To this end, this study was designed to handle large datasets of truck haulage using GMM for developing a novel and accurate prediction model of truck productivity. The large dataset had 303,712 groups of data, which was collected from active oil sands mines in Northern Alberta, Canada. GMM was first used to cluster the large dataset. After that, a latent variable was extracted to build the prediction model in conjunction with other input variables (Berlin et al., 2013). Because the multiple linear regression (MLR) method is a computationally efficient tool and can provide explicit formulae for engineers (Ciulla & D'Amico, 2019), it was adopted to build the prediction models. The main contribution of this study was the first application of GMM to preprocess massive amounts of data to improve model predictability of truck productivity.

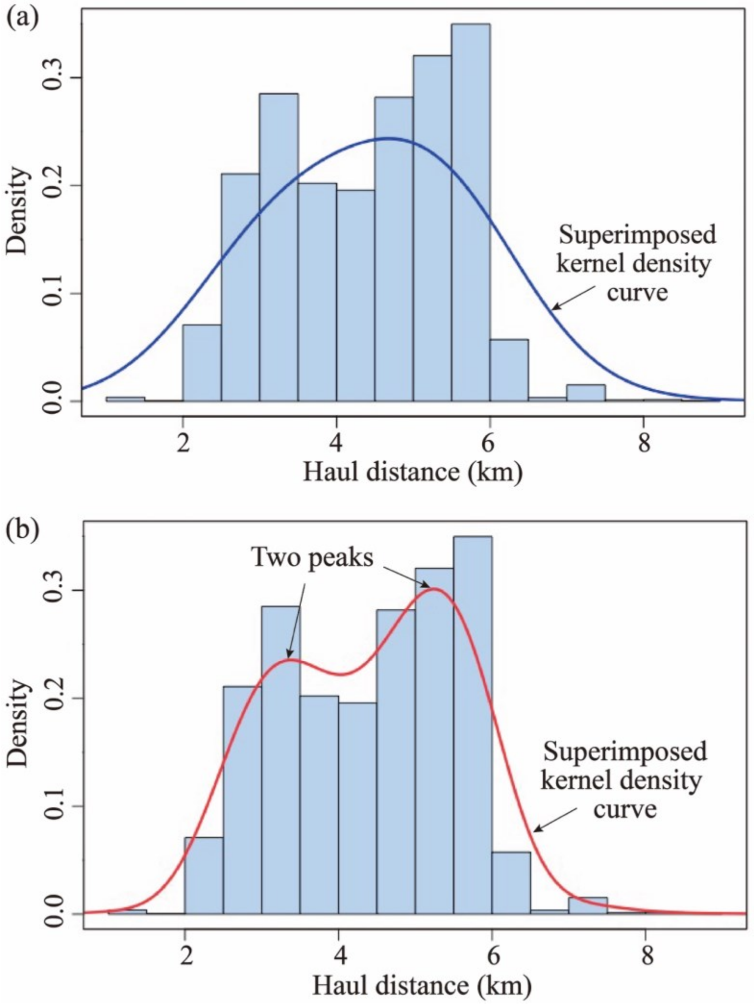


Figure 2.1 Data distributions from oil sands mines (using the haul distance as an example). (a) Haul distance is described by a single Gaussian distribution; (b) Haul distance is described by a multi-peak Gaussian distribution.

2.2. Methodology

Figure 2.2 illustrates the flowchart of the overall methodology. A large dataset from the mine data management system was split into a training dataset and a testing dataset for model training and evaluation. Before the modeling, the training dataset was clustered by GMM into three latent classes, and a latent variable was generated by the labels of these classes. Two MLR models were then built on the training dataset, including the ordinary-MLR (O-MLR) model and the GMM-

MLR model. The GMM-MLR model was the proposed model for predicting truck productivity, incorporating the latent variable. The O-MLR model was the baseline model without involving the latent variable. The testing dataset was used to evaluate the performance of two MLR models. The performance of each model was quantified by four commonly used parameters in statistics (Wu et al., 2020): the adjusted R^2, the root mean square error (RMSE), the mean absolute error (MAE), and the mean absolute percentage error (MAPE). Finally, the Lindeman, Merenda, and Gold (LMG) method was selected to determine the relative importance of input variables to the GMM-MLR model since LMG is a simple and efficient method when an MLR model contains few input variables (Tian et al., 2016). The abovementioned training process was implemented in RStudio software using the R (version 4.1.3) language environment.

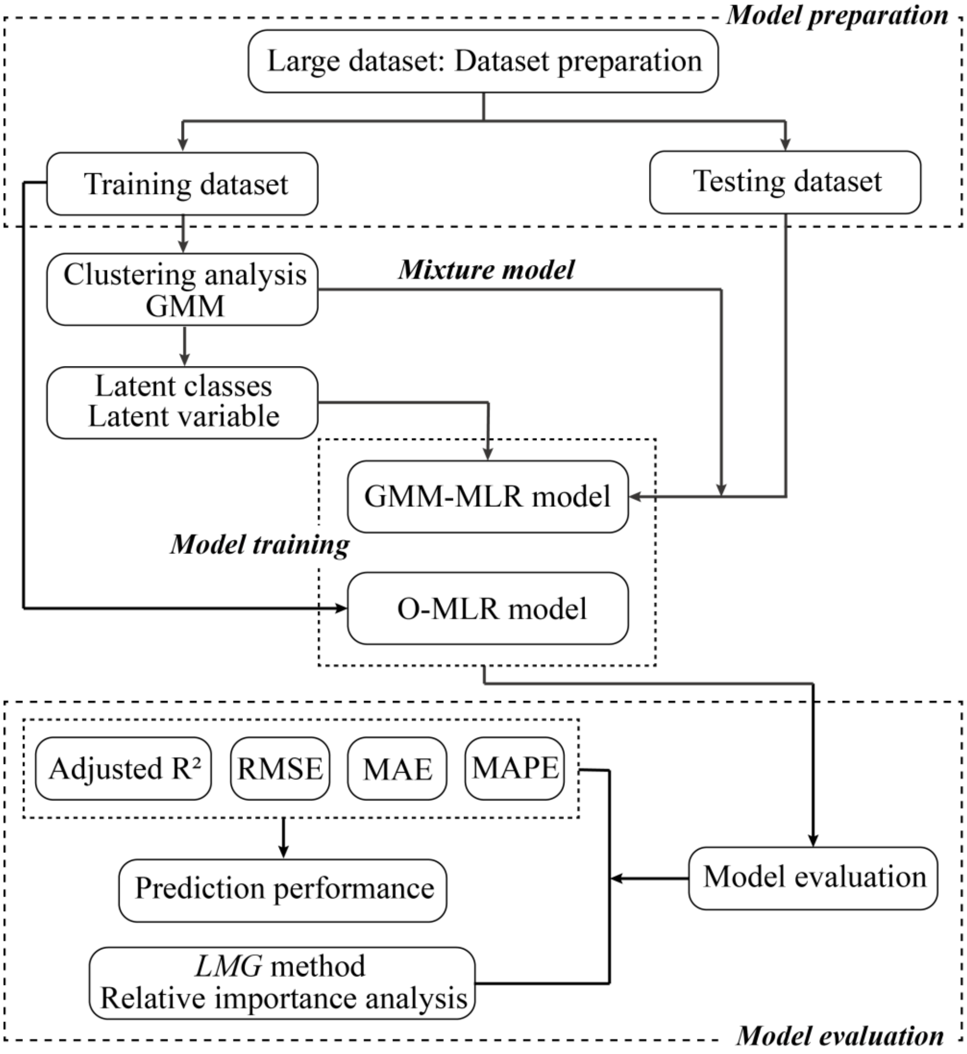


Figure 2.2 Flowchart showing the execution process of methodology.

2.2.1. Multiple linear regression (MLR)

MLR is a common statistical technique for building prediction models (Ciulla & D'Amico, 2019). It has been widely applied in the fields of agriculture (Dhulipala & Patil, 2020), environment (Tan et al., 2014), and energy (Maaouane et al., 2021) because of its simple structure and efficient calculation (Ciulla & D'Amico, 2019). In addition, mining companies often utilize MLR to build prediction models because it can provide explicit expressions for engineers to use easily (Cervantes et al., 2019; Chanda & Gardiner, 2010). MLR obtains the best-fitting line by minimizing the square sum of vertical deviations from data points to a fitted line (Maaouane et al., 2021). This line describes the linear relationship between an output variable and a set of input variables. Suppose that $x = \{x_1, x_2, \dots, x_M\}$ is the input vector, where *M* is the number of input variables, and y is the output variable. The linear relationship can be expressed as follows:

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_m x_m + \cdots + \beta_M x_M + \epsilon \tag{2-1}$$

where *β₀* is the constant term that denotes the intercept, *βₘ* is the regression coefficients linked to the *m*th input variable, and ϵ is the random error term. Equation (2-1) represents a prediction model based on the MLR method.

2.2.2. Gaussian mixture modeling (GMM)

GMM is an unsupervised clustering technique that identifies several latent classes from a data population (Bishop, 2006). A set of data points in each class adheres to a specific Gaussian distribution. Statistically, GMM generates a mixture model, which is defined as the weighted combination of *k* Gaussians, representing the probability density function of the data population. The description of the mixture model is written as follows (Leisch, 2004):

$$P(y|x, \emptyset) = \sum_{k=1}^{K} \pi_k f_k(y|x, \theta_k) \tag{2-2}$$

where $f(y|x, \theta_k)$ denotes the probability density function of the *k*th class; θ_k is the parameter vector, which is defined as (μ_k, Σ_k); μ_k and Σ_k are the mean vector and the covariance matrix, respectively; the parameter π_k is the weight of the *k*th class, also known as the mixture coefficient, which is non-negative together with $\sum_{k=1}^{K} \pi_k = 1$; and $\emptyset$ indicates the parameter set of the mixture model, which is written as $\{\pi_k, \theta_k\}$.

To determine the mixture model, GMM first estimates the parameter set $\{\pi_k, \theta_k\}$ from all data points. This estimation can be conducted using the expectation-maximization (EM) algorithm to maximize log-likelihood (log L) (Fu et al., 2021):

$$log\, L = \sum_{n=1}^{N} \log(P(y|x, \phi)) = \sum_{n=1}^{N} \log(\sum_{k=1}^{K} \pi_k f(y|x, \theta_k)) \tag{2-3}$$

where N is the number of data points. The EM algorithm determines the parameter set $\{\pi_k, \theta_k\}$ through an iterative process, mainly including the E-step and the M-step. In the E-step, data points are assigned to a class with the maximum posterior probability (Leisch, 2004). Based on the Bayes' theorem (Li et al., 2019), the posterior probability that a data point (x_n, y_n) belongs to each class is given by

$$\gamma_{nk} = \frac{\pi_k f_k(y_n|x_n, \theta_k)}{\sum_{k=1}^{K} \pi_k f(y_n|x_n, \theta_k)} \tag{2-4}$$

The data point is assigned to the kth class when

$$\lambda_k = \underset{k \epsilon \{1,2,\ldots,K\}}{\operatorname{argmax}} \gamma_{nk} \tag{2-5}$$

where λ_k represents a set of data points that has the maximum posterior probability, γ_{nk}. Later, in the M-step, with the γ_{ik}, the parameter set $\{\pi_k, \theta_k\}$ can be further estimated by the likelihood setting in Equation (2-3). These two steps are repeated until the maximum log-likelihood is reached. As a result, the parameter set can be acquired from the EM process.

After the parameters set is estimated, GMM starts to determine the optimal number of latent classes. In this study, the Bayesian information criterion (BIC) was selected as a metric to optimize the number because it has been commonly used in engineering and proved to be superior to other methods in a rigorous study (Russell & Raftery, 2009). The BIC formula is shown below:

$$BIC = -2logL + ClogN \tag{2-6}$$

where C means the number of estimated parameters. The criterion for the optimal number is to minimize the BIC value to achieve a more proper mixture model of the data population (McLachlan et al., 2019).

2.2.3. Dataset preparation and preprocessing

The large dataset contained 303,712 groups of data covering a full year of truck haulage cycles. Before the prediction models were built, the dataset was randomly and proportionally split into training (75%) and testing datasets (25%). Both the training and testing datasets had five input variables observed from the mine sites. These five input variables were chosen because they have been noted by practicing engineers at mine sites and are all associated with truck cycle time. They were related to haulage operations, haul routes, and meteorological factors, which were also selected with reference to the research by Chanda and Gardiner (2010). The observed input variables included haul distance (x_1, km), empty speed (x_2, km/h), destination (x_3), ambient temperature (x_4, °C), and precipitation (x_5, mm/h). The first three variables were monitored and identified by the installed sensors on trucks. The remaining two variables were obtained from the local meteorological observatory (MEP, 2023). Table 2.1 shows these five input variables, of which the haul distance, empty speed, and ambient temperature were continuous variables. The destination and precipitation were categorical variables, which means that they had several distinct categories. For example, there were three destinations at the mine sites, denoted as D_1, D_2, and D_3. Figure 2.3 shows the statistical information about these observed input variables (x_m) and the output variable (y). In Figure 2.3, the superimposed density curves represent the distribution of these variables. The continuous variables, including the haul distance, empty speed, and ambient temperature, were represented by the skewed Gaussian and multi-peak Gaussian distributions.

Remarkably, the multi-peak Gaussian distributions shown by the haul distance and ambient temperature indicated that the original dataset had a mixture of Gaussians, which provided the rationale for selecting GMM to preprocess the dataset (Ma et al., 2014).

By preprocessing the training dataset using GMM, several latent classes were identified from all data points, and a latent variable was generated by the labels of classes. This latent variable was also a categorical variable with several distinct categories; it was in conjunction with other observed input variables to establish the GMM-MLR model. As for the testing dataset, the data points were grouped into several classes based on the mixture model obtained in GMM. The results of the GMM analysis and the number of latent classes will be explained and discussed in detail in Section 2.3.1.

Table 2.1 The input variables (x_m), characteristics, and their descriptions.

Input variable	Unit	Type	Description
Haul distance (x_1)	km	Continuous	Listing haul distance for each cycle from a loading area to a dumping area
Empty speed (x_2)	km/h	Continuous	Listing running speed of empty truck for each cycle
Destination (x_3)	-	Categorical	Listing three destinations of truck haulage: D_1, D_2, and D_3
Ambient temperature (x_4)	℃	Continuous	Listing ambient temperature per hour at the local mining area
Precipitation (x_5)	mm/h	Categorical	Listing precipitation per hour at mine sites with three categories: no precipitation (P_1), 0-1 mm/h (P_2), and larger than 1 mm/h (P_3)

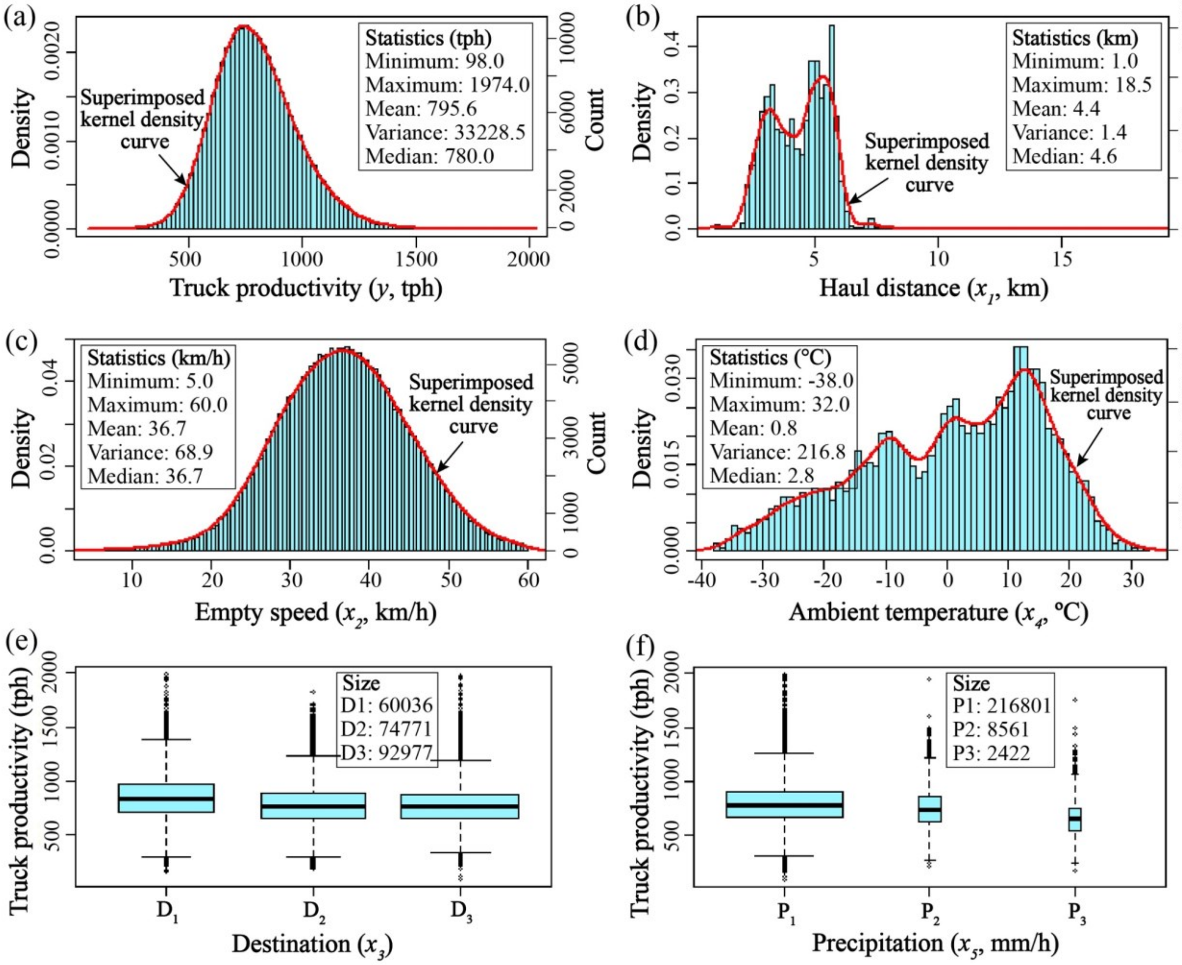


Figure 2.3 The output variable and observed input variables in the training dataset. (a) The output variable (y): truck productivity (unit: tph, tonnes per hour); (b)-(d) show the histograms of the continuous variables: haul distance (x_1), empty speed (x_2), and ambient temperature (x_4); (e)-(f) show the boxplots of two categorical variables with three categories: destination (x_3) and precipitation (x_5). ("###": the input information is not disclosed as it is the proprietary property of mining companies.)

2.2.4. Performance criteria for prediction models

To investigate the effect of GMM on prediction performance, two MLR models were built for comparison. One was the GMM-MLR model that considered a latent variable generated from the

GMM analysis. The other was the O-MLR model, serving as the baseline model without involving the latent variable. To assess the performance of these two models, four performance parameters were adopted in this study (Wu et al., 2020). These parameters were RMSE, MAE, MAPE, and the adjusted R^2. They are calculated as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{n=1}^{N} y_n - \hat{y}_n)^2} \tag{2-7}$$

$$MAE = \frac{1}{N}\sum_{n=1}^{N} |y_n - \hat{y}_n| \tag{2-8}$$

$$MAPE = \frac{1}{N}\sum_{n=1}^{N} |\frac{y_n - \hat{y}_n}{y_n}| \tag{2-9}$$

where y_n is the actual values, indicating the measured truck productivity in the testing dataset; $\hat{y}_n$ is the predicted truck productivity. RMSE shows the standard deviation of the residuals between actual and predicted values; MAE is used to characterize the absolute error between actual and predicted values, while MAPE denotes the relative error (Wu et al., 2020). The adjusted R^2 is calculated based on R^2. Both are shown, respectively, as Equation (2-10)-(2-11):

$$R^2 = 1 - \frac{\sum_n^N (y_n - \hat{y}_n)^2}{\sum_n^N (y_n - \bar{y}_n)^2} \tag{2-10}$$

$$R_{adj}^2 = 1 - \frac{(1-R^2)(N-1)}{N-M-1} \tag{2-11}$$

where $\bar{y}_n$ is the mean of actual values and M is the number of input variables. Both R^2 and the adjusted R^2 represent the degree to which data points fit a curve, ranging from 0 to 1. The adjusted R^2 is generally smaller than R^2 because input variables unrelated to the output variable are screened when calculating the adjusted R^2; therefore, the adjusted R^2 indicates the goodness of fit more accurately than R^2 (Mittlböck, 2002). The prediction model with a higher adjusted R^2 and a lower RMSE, MAE, and MAPE has better prediction accuracy.

2.2.5. The Lindeman, Merenda, and Gold (LMG) method

To evaluate the contributions of input variables to the proposed GMM-MLR model, a quantitative method was introduced to calculate the relative importance of each input variable. This method is called the LMG method (Groemping, 2006). It is straightforward and efficient when an MLR model contains few input variables (Tian et al., 2016). The LMG method can consider all sequences of an input variable entering an MLR model. The relative importance of this input variable is calculated by averaging the R^2 of all possible orderings, which can be determined according to Equation (2-12):

$$LMG = \frac{1}{M!}\sum_{p\ permutation} seq\{R^2(x_m|p)\} \qquad (2\text{-}12)$$

where *M!* is the factorial of *M*; *p* represents the permutation of input variables before entering x_m, and $seq\{R^2(x_m|p)\}$ refers to the R^2 of the prediction model after entering x_m in the permutation *p*. The relative importance of x_m is the average value of R^2 under all permutations.

2.3. Results and discussion

2.3.1. GMM analysis

In this study, GMM was applied to cluster the training dataset under the principle of minimizing BIC. As a result, the training dataset was clustered into three latent classes, as shown in Figure 2.4. In Figure 2.4(a), taking truck productivity as an example, the number of data points was different in each class. The boxplot showed that Class 1 (C_1) had the lowest number of data points (6,684), while Classes 2 and 3 (C_2 and C_3) had 119,145 and 101,955 data points, respectively. Q_1 and Q_3 were the 25th and 75th percentiles in each class, depicting the distribution interval of data points

(Patil et al., 2018). Figure 2.4(b) shows the frequency histogram of truck productivity in each latent class. According to the definition of GMM (Bishop, 2006), the data points in each latent class are

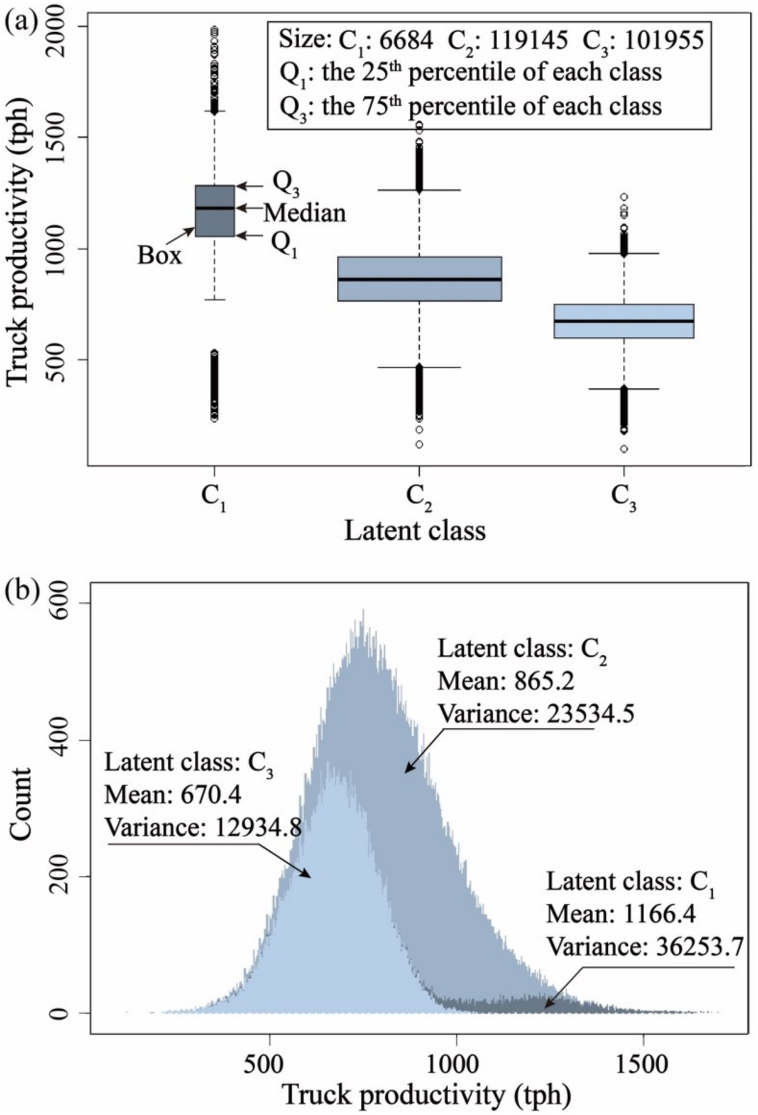


Figure 2.4 Extraction of three latent classes from the training dataset. (a) Boxplots of three classes; (b) Histogram: truck productivity corresponds to three latent classes, which are described by Gaussian distributions.

described by a Gaussian distribution. The mean values of each Gaussian were around 1,166 tph, 865 tph, and 670 tph. As shown in Figure 2.4, the training dataset was well partitioned into three latent classes. Amid these classes, the value of truck productivity varied significantly, in the order of $C_1 > C_2 > C_3$. This can be known as the high, medium, and low truck productivity at mine sites. This is similar to Ni et al. (2020); in their research, the streamflow data were also clustered into three latent classes using GMM. A prediction model was then developed for monthly low flow forecasting based on the GMM analysis; the R^2 of this model was increased from 0.59 to 0.66 compared to the baseline model without the GMM analysis. This suggests that implementing GMM may improve the model accuracy of truck productivity in this study.

2.3.2. Establishment, interpretation, and comparison of prediction models

2.3.2.1. O-MLR model

The O-MLR model was a baseline model established on the training dataset without the implementation of GMM. The explicit equation of this model can be written as

$$y = \beta_0 + \sum_{m=1}^{5} \beta_m x_m \tag{2-13}$$

where y was the output variable. β_0 was the intercept of the equation, and β_m was the regression coefficients linked to the mth observed input variable (x_m). The regression parameters of Equation (2-13) can be seen in Table 2.2 presents the observed input variables (x_m), regression coefficients (β_m), significance test results (p-values), and intercept (β_0). The observed input variables included the haul distance (x_1), empty speed (x_2), destination (x_3), ambient temperature (x_4), and precipitation (x_5). The regression coefficients describe the mathematical relationship between each input variable and the output variable (Wei, 1990). For example, the haul distance's regression coefficient (β_1) was a negative value (-62.70), indicating that the truck productivity was reduced by 62.70 tph when the haul distance increased by 1 km. The same result was found by Schexnayder et al. (1999); their data proved that the truck productivity dropped by 374 tph when the haul distance rose by 1.3 km. Hence, the truck productivity had a negative relationship with the haul distance. The *p*-values for regression coefficients represent whether these relationships are statistically significant (Ge, 2008). In statistics, if a *p*-value is smaller than a significance level (usually 0.05), the relationship between the input and output variables is significant (Iqbal & Sun, 2014). As shown in Table 2.2, the relationships between three continuous variables (haul distance, empty speed, and ambient temperature) and truck productivity were statistically significant because their *p*-values were less than 0.05. Also, two categorical variables (destination and precipitation) were significantly related to truck productivity, except for the second category (D_2) of destination, as its *p*-value (0.546) was larger than 0.05. In short, almost all the input variables had a significant relationship with truck productivity, suggesting the trained O-MLR model can be used to predict truck productivity.

Table 2.2 The regression parameters and significance test results for the O-MLR model.

Input variable		Regression coefficient		*p*-value	Significance test
x_1	Haul distance	β_1	-62.70	$< 2\times10^{-16}$	Reject
x_2	Empty speed	β_2	4.91	$< 2\times10^{-16}$	Reject
x_4	Ambient temperature	β_4	-1.44	$< 2\times10^{-16}$	Reject
x_3	Destination (D_2)	β_3	-0.57	0.546	Accept
	Destination (D_3)		-11.71	$< 2\times10^{-16}$	Reject
x_5	Precipitation (P_2)	β_5	-34.31	$< 2\times10^{-16}$	Reject
	Precipitation (P_3)		-75.51	$< 2\times10^{-16}$	Reject
Intercept		β_0	900.20	$< 2\times10^{-16}$	Reject

Note: If the *p*-value is less than 0.05, the null hypothesis that *x* and *y* are not significantly related will be rejected; otherwise, it will be accepted. For example, the *p*-value (0.546) for the second category (D_2) of x_3 is larger than 0.05; as a result, the null hypothesis is accepted.

2.3.2.2. GMM-MLR model (incorporation of a latent variable and its interaction terms)

After the implementation of GMM, the training dataset was employed to build the GMM-MLR model. The explicit expression of this model can be given by

$$y = \beta_0 + \sum_{m=1}^{5} \beta_m x_m + \beta_6 x_6 + \sum_{m=1}^{5} \beta_{m+6} (x_m \times x_6) \tag{2-14}$$

where β_6 was the regression coefficients of the latent variable (x_6), and β_{m+6} was the regression coefficients of interaction terms ($x_m \times x_6$) between the five observed input variables (x_m) and the latent variable (x_6). Compared with Equation (2-13), two more terms were incorporated in Equation (2-14), including an independent term and a set of interaction terms. The independent term was constituted by a latent variable (x_6) and its regression coefficient (β_6). The latent variable was a categorical variable with three categories (C_1, C_2, and C_3), and the GMM analysis showed that it was related to the five observed input variables. Hence, a set of interaction terms was considered in the GMM-MLR model between the five observed input variables and the latent variable. The interaction term refers to the product of two or more input variables in a regression equation (Jaccard et al., 2003). For instance, in Equation (2-14), the haul distance (x_1) had an interaction term ($x_1 \times x_6$) with the latent variable (x_6).

Table 2.3 lists the detailed regression parameters of Equation (2-14), including the input variables, interaction terms, regression coefficients, *p*-values, and intercept. As shown in Table 2.3, the GMM-MLR model incorporated the five observed input variables, a latent variable and five sets of interaction terms. The regression coefficients in Table 2.3 will be explained in detail in Section 2.3.2.3. As for the *p*-values, almost all the input variables and interaction terms had a significant relationship with the truck productivity since their *p*-values were smaller than 0.05. Thus, the established GMM-MLR model can also be applied for predicting truck productivity.

Table 2.3 The regression parameters and significance test results for the GMM-MLR model.

	Input variable and interaction term		Regression coefficient	p-value	Significance test
x_1	Haul distance	β_1	-105.92	$< 2\times10^{-16}$	Reject
x_2	Empty speed	β_2	0.52	4.29×10^{-5}	Reject
x_4	Ambient temperature	β_4	-4.23	$< 2\times10^{-16}$	Reject
x_3	Destination (D_2)	β_3	-42.20	$< 2\times10^{-16}$	Reject
	Destination (D_3)		-40.58	$< 2\times10^{-16}$	Reject
x_5	Precipitation (P_2)	β_5	-44.28	6.26×10^{-15}	Reject
	Precipitation (P_3)		-71.90	6.58×10^{-10}	Reject
x_6	Latent variable (C_2)	β_6	-643.08	$< 2\times10^{-16}$	Reject
	Latent variable (C_3)		-973.95	$< 2\times10^{-16}$	Reject
$x_1 \times x_6$	Haul distance × latent variable (C_2)	β_7	8.48	2.28×10^{-16}	Reject
	Haul distance × latent variable (C_3)		75.91	$< 2\times10^{-16}$	Reject
$x_2 \times x_6$	Empty speed × latent variable (C_2)	β_8	9.68	$< 2\times10^{-16}$	Reject
	Empty speed × latent variable (C_3)		4.99	$< 2\times10^{-16}$	Reject
$x_4 \times x_6$	Ambient temperature × latent variable (C_2)	β_{10}	2.22	$< 2\times10^{-16}$	Reject
	Ambient temperature × latent variable (C_3)		3.23	$< 2\times10^{-16}$	Reject
$x_3 \times x_6$	Destination (D_2) × latent variable (C_2)	β_9	51.92	$< 2\times10^{-16}$	Reject
	Destination (D_2) × latent variable (C_3)		32.95	$< 2\times10^{-16}$	Reject
	Destination (D_3) × latent variable (C_2)		41.95	$< 2\times10^{-16}$	Reject
	Destination (D_3) × latent variable (C_3)		20.83	6.78×10^{-13}	Reject
$x_5 \times x_6$	Precipitation (P_2) × latent variable (C_2)	β_{11}	11.65	0.046	Reject
	Precipitation (P_2) × latent variable (C_3)		14.41	0.015	Reject
	Precipitation (P_3) × latent variable (C_2)		-4.74	0.690	Accept
	Precipitation (P_3) × latent variable (C_3)		25.21	0.037	Reject
	Intercept	β_0	1,616.21	$< 2\times10^{-16}$	Reject

Note: If the p-value is less than 0.05, the null hypothesis that x and y are independent will be rejected; otherwise, it will be accepted. For example, the p-value (4.29×10^{-5}) for x_1 is less than 0.05; as a result, the null hypothesis is rejected.

2.3.2.3. Interpretation of interaction terms

The interaction term implies that the effect of an input variable on an outcome depends not only on that particular input variable but on other input variables (Moy et al., 2015). For instance, in the GMM-MLR model, the effect of haul distance on truck productivity depended on both the haul distance and the latent variable. Furthermore, the GMM analysis demonstrated that the latent variable could represent three classes of truck productivity: C_1 (high values), C_2 (medium values), and C_3 (low values). This means that the interaction terms can further characterize the effects of the five observed input variables on each class of truck productivity. Also, these effects can be quantitatively measured through the regression coefficients of the established GMM-MLR model.

In Table 2.3, there are 11 sets of regression coefficients. Among them, the regression coefficients (β_1 to β_5) for each observed input variable (x_m) indicated the effect of the input variable on the truck productivity belonging to C_1. The regression coefficients (β_7 to β_{11}) of each interaction term ($x_m \times x_6$) suggested the effect of the input variable on the truck productivity belonging to C_2 and C_3. As shown in Figure 2.5, the haul distance and precipitation were used as examples to interpret the regression coefficients. In Figure 2.5(a), there were three negative values: -105.92 tph, -97.44 tph, and -30.01 tph. Of these values, -105.92 was the β_1, indicating that the high truck productivity (C_1) was reduced by 105.92 tph when the haul distance increased by 1 km. The values of -97.44 tph and -30.01 tph were calculated from the sum of the β_1 (-105.92) and β_7 (8.48 and 75.91), meaning that the medium (C_2) and low (C_3) truck productivity decreased by 97.44 tph and 30.01 tph when the haul distance rose by 1 km. Likewise, the effects of the precipitation (P_2 and P_3) on three classes of truck productivity are illustrated in Figure 2.5(b)-(c). In Figure 2.5(b), the high, medium, and low truck productivity were reduced by 44.28 tph, 32.63 tph, and 29.87 tph when the precipitation (P_2) increased by 1 mm/h. In Figure 2.5(c), the effect of the precipitation (P_3) on the medium truck productivity (C_2) was ignored as the *p*-value of this term was larger than 0.05. The high and low truck productivity dropped by 71.90 tph and 46.69 tph, respectively, when the precipitation (P_3) rose by 1 mm/h. Thus, the interaction terms revealed that the effect of each observed input variable on truck productivity was significantly different between the three classes. The finding was similar to that in studies by Kyburz et al. (2011) and Lunt (2015), who were interested in the effect of treated time on a radiographic damage score for subjects in an early or late treated group. To evaluate the difference between the groups, Kyburz et al. (2011) and Lunt (2015) constituted an interaction term in a regression model. The results proved that the interaction term could also measure the different effects between groups.

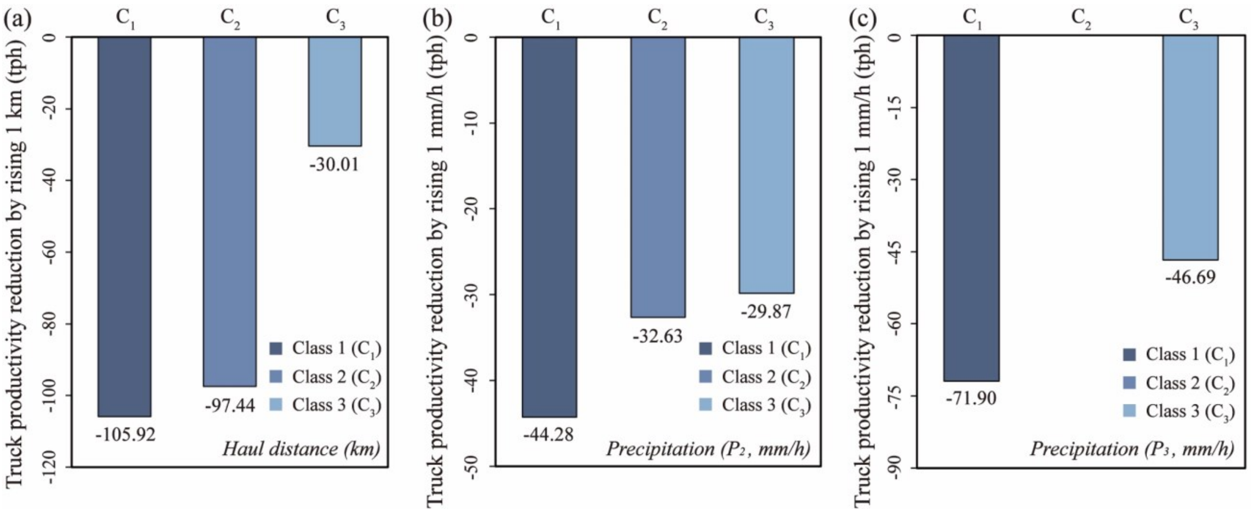


Figure 2.5 The effects of the observed input variables on each class of truck productivity (C_1, C_2, and C_3 represented the high, medium, and low truck productivity, respectively). (a) The effects of the haul distance. (b) The effects of the precipitation (P_2). (c) The effects of the precipitation (P_3).

2.3.2.4. Comparison between O-MLR and GMM-MLR models

Figure 2.6 shows the scatterplots of the actual (on the vertical axis) and predicted (on the horizontal axis) truck productivity. The $y = x$ is a 45-degree diagonal line. The closer the scatters along the $y = x$ line, the better the prediction (Liu et al., 2020). As shown in Figure 2.6, the scatters generated by the GMM-MLR model were closer along the line, which means that the GMM-MLR model performed better than the O-MLR model. To quantitatively evaluate the performance of the established models, four parameters were calculated for each model from the testing dataset. The results are listed in Table 2.4, which shows that the GMM-MLR model was more accurate than the O-MLR model. The GMM-MLR model had a lower RMSE, MAE, and MAPE, and a higher adjusted R^2, with values of 91.87, 72.58, 0.10, and 0.75. Accordingly, these four performance parameters of the O-MLR model were 160.27, 124.31, 0.17, and 0.23. In terms of the adjusted R^2 alone, the accuracy of the GMM-MLR model (the adjusted $R^2 = 0.75$) was three times higher than the O-MLR model (the adjusted $R^2 = 0.23$). In other words, the GMM-MLR model performed well in predicting truck productivity. After using GMM to preprocess the large dataset, the model predictability was considerably enhanced by incorporating the latent variable and its interaction terms. This provides new insights and inspiration for engineers to handle massive amounts of engineering data in their future work. Similar findings were also noted in the research by Ho Park et al. (2021), who incorporated seven input variables and constituted 11 sets of interaction terms in a linear regression model for post-event flood waste estimation. The results showed that the adjusted R^2 of the prediction model was increased from 0.36 to 0.59 when the model was added with these input variables and interaction terms.

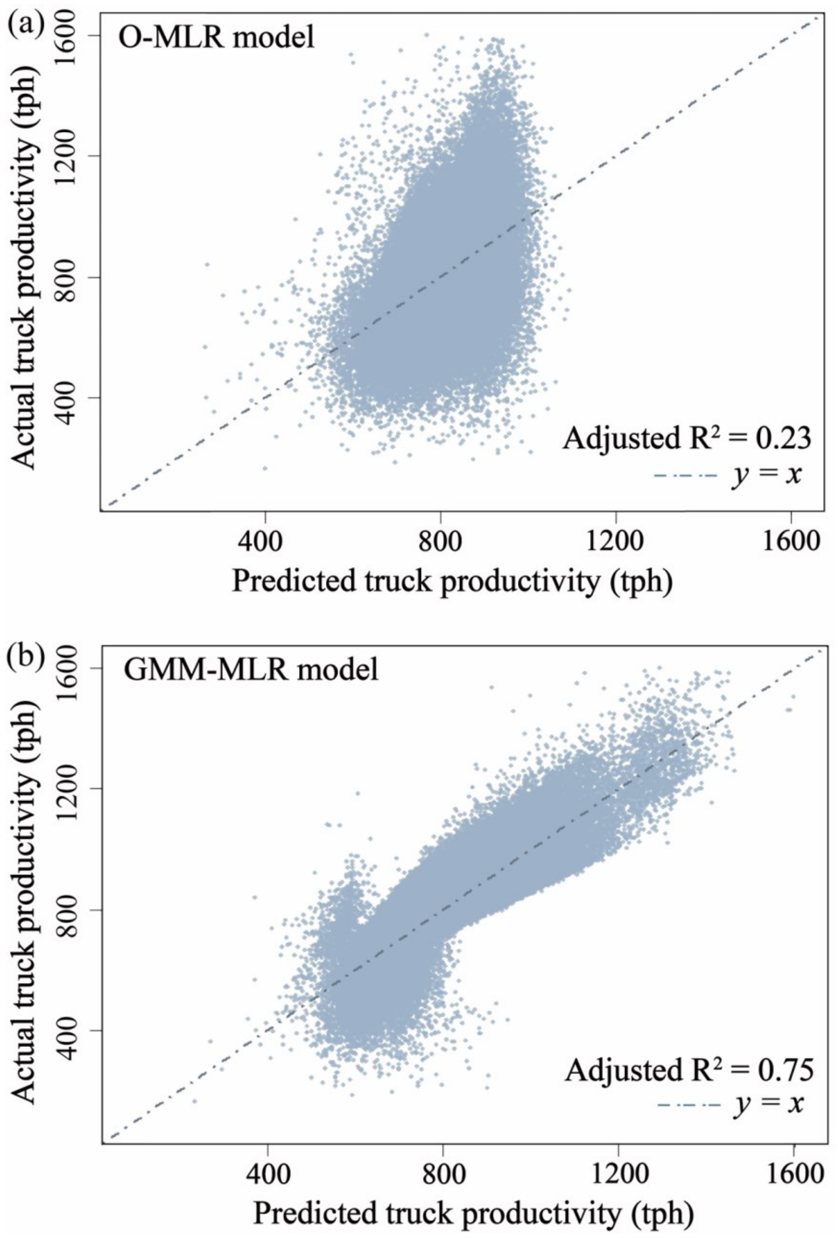


Figure 2.6 Scatterplots of the actual truck productivity in the testing dataset and the predicted truck productivity generated by the O-MLR and GMM-MLR models. (a) The O-MLR model; (b) The GMM-MLR model.

Table 2.4 Performance evaluation by four parameters for the trained models.

Prediction model	RMSE	MAE	MAPE	Adjusted R^2
GMM-MLR	91.82	72.58	0.10	0.75
O-MLR	160.27	124.31	0.17	0.23

2.3.3. Relative importance analysis of observed input variables

In this study, the LMG method was adopted to determine the relative importance of each observed input variable. Figure 2.7 shows the relative importance of these observed input variables in the GMM-MLR model. The vertical axis represented the five observed input variables; the horizontal axis was the relative importance proportion (in percentage) of each one. The relative importance for the input variables was ranked as haul distance (54.65%) > empty speed (23.14%) > ambient temperature (13.82%) > destination (6.22%) > precipitation (2.18%). Among these variables, the haul distance had the highest relative importance, indicating its effect on truck productivity was greater than that of other input variables. Cervantes et al. (2019) reported that mining companies often plotted a fitted line between haul distance and truck productivity because the increase in haul distance directly affects the increase in cycle time, thereby reducing truck productivity. Similar to the study by Cervantes et al. (2019), the results from the relative importance analysis also proved that the haul distance was a critical input variable in predicting truck productivity. After the haul distance, the analysis showed that the empty speed had the second-highest relative importance, with a value of 23.14%. According to Schexnayder et al. (1999), the empty speed determined the travel time from dumping sites to loading sites, affecting truck productivity. The relative importance of the destination was 6.22%, indicating its effect on the truck productivity was not significant. This is reasonable since the destination cannot directly affect the payload weight and cycle time length (Navarro Torres et al., 2019). The sum of the relative importance of the ambient temperature and precipitation was 16.01%, showing that the meteorological factors had a certain contribution to the GMM-MLR model. Similar to the research by Sun et al. (2018), the prediction accuracy was enhanced by 5.13% when considering the effect of meteorological factors. To summarize, the observed input variables contributed differently to the GMM-MLR model, with haul distance being the most crucial input variable. The relative importance analysis can help mine engineers to gain a comprehensive understanding of the real-world influences affecting truck productivity, thus providing appropriate suggestions and methods to improve truck productivity.

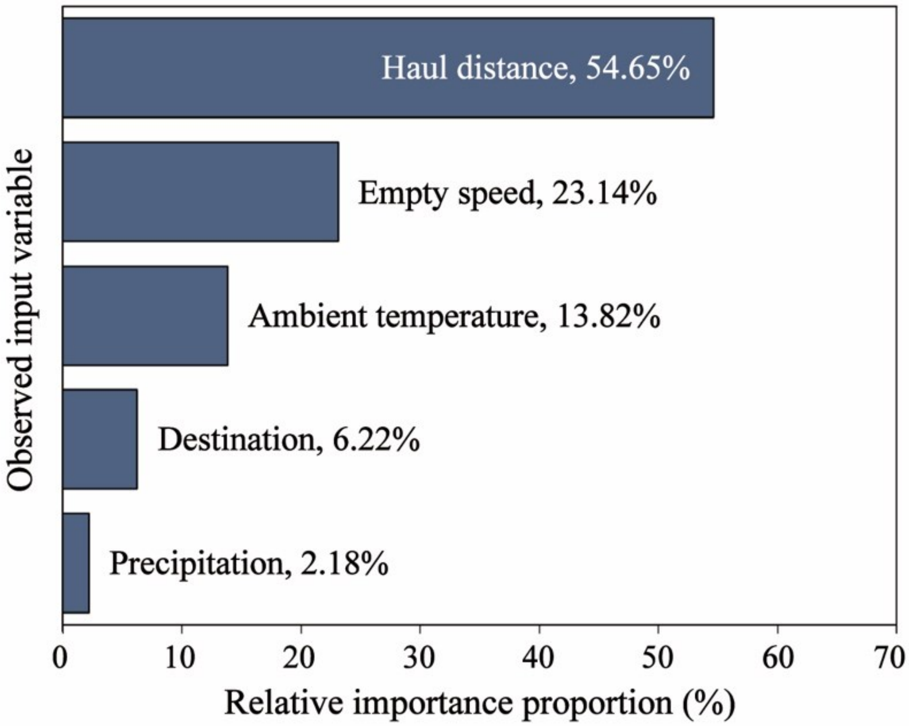


Figure 2.7 The relative importance of the observed input variables in the GMM-MLR model.

2.3.4. Advantage, limitation, and future improvement of proposed model

In this study, the GMM-MLR model was the proposed model for predicting truck productivity. Unlike previous studies (Baek & Choi, 2020; Chanda & Gardiner, 2010; Sun et al., 2018), this proposed model not only considered input variables observed at mine sites but involved unobserved variables (i.e., latent variables) obtained from the GMM analysis. Due to the involvement of latent variables, the model accuracy of truck productivity was considerably enhanced (e.g., the R^2 was increased from 0.23 to 0.75). Despite its better performance, the proposed GMM-MLR model had limitations in this study. Much research will be required to further the prediction model. For instance, although GMM has advantages in dealing with large datasets with multi-peak Gaussian distributions, it is not the only clustering technique (Shirkhorshidi et al., 2014). Previous studies have shown that clustering techniques such as K-means and fuzzy C-means improved model accuracy (Liu et al., 2020; Wu et al., 2009). A comparative study between clustering techniques may be helpful to improve prediction models. In addition, more input variables, such as tire temperature, wind speed, and elevation, can be considered in the future to build prediction models. According to Ma et al. (2021), high tire temperature may cause rubber failure, affecting truck speed and cycle time. Likewise, wind speed and elevation over the haul route may have an impact on truck speed and driver's vision (Chanda & Gardiner, 2010; Sun et al., 2018). However, these parameters are not included in the currently proposed model. Furthermore, the modeling approach used in this study was the MLR method, while more robust algorithms, such as support vector machine (Drosou & Koukouvinos, 2017), random forest (Cakir et al., 2021), and artificial neural network (Tadeusiewicz, 2015), can also provide accurate prediction models. In the future, these algorithms will be used to increase model predictability.

2.4. Conclusions

This study aimed to handle large datasets of truck haulage at mine sites using Gaussian mixture modeling (GMM) for developing a novel and accurate prediction model of truck productivity based on multiple linear regression (MLR). The main conclusions are listed below:

(1) GMM significantly improved the predictability of the truck productivity prediction model by preprocessing large truck haulage datasets. For example, the adjusted R^2 of the ordinary-MLR (O-MLR) model was only 0.23, whereas the GMM-MLR improved the predictability more than three times, with an adjusted R^2 of 0.75. This information can provide new insights and inspiration for engineers to deal with massive amounts of engineering data in their future work.

(2) Interaction terms quantitatively measured the significant differences in the effect of an observed input variable on truck productivity between classes. For instance, when the haul distance increased by 1 km, the high (Class 1), medium (Class 2), and low (Class 3) truck productivity dropped by 105.92 tph, 97.44 tph, and 30.01 tph, respectively. Hence, the effect of the haul distance on high truck productivity was more significant than that on medium and low truck productivity, showing the significant differences between the classes revealed by the interaction terms.

(3) Among the observed input variables, the haul distance was the most crucial input variable of the GMM-MLR model. The relative importance of the haul distance was 54.65%, which was higher than that of the empty speed (23.14%), destination (6.22%), ambient temperature (13.82%), and precipitation (2.18%). The relative importance analysis helps mine engineers to gain a comprehensive understanding of the real-world influences affecting truck productivity, thus providing appropriate suggestions and methods to improve truck productivity.

(4) The GMM-MLR model with higher accuracy is expressed as an explicit and straightforward equation, which can help mine engineers predict truck productivity at mine sites.

Chapter 3. Prediction of truck productivity at mine sites using tree-based ensemble models combined with Gaussian mixture modeling

This chapter has been published as **C. Fan**, N. Zhang, B. Jiang, W.V. Liu, Prediction of truck productivity at mine sites using tree-based ensemble models combined with Gaussian mixture modeling, *International Journal of Mining, Reclamation and Environment*. © Taylor & Francis. 37 (2023) 66-86.

Nomenclatures

BIC	Bayesian information criteria
C	Mixture model complexity
CART	Classification and regression tree
DT	Decision tree
EM	Expectation-maximization
f_k	Probability density function
FS	Forward stagewise
GBR	Gradient boosting regression
GMM	Gaussian mixture modeling
GMM-DT	Gaussian mixture modeling-based decision tree
GMM-GBR	Gaussian mixture modeling-based gradient boosting regression
GMM-MLR	Gaussian mixture modeling-based multiple linear regression
GMM-RF	Gaussian mixture modeling-based random forest
k	The *k*th latent classes
K	The number of latent classes
L	Likelihood of a set of data points

m	The mth input variable
MAE	Mean absolute error
$MAPE$	Mean absolute percentage error
ML	Machine learning
MLR	Multiple linear regression
n	The nth data point
N	The number of data points
OOB	Out-of-bag
P	Mixture model
PDF	Probability density function
Q_1, Q_3	The 25th and 75th percentiles
R^2	Coefficient of determination
$RMSE$	Root mean square error
t	t-fold cross-validation
tph	Tonnes per hour
x_m	The mth input variable
y	Output variable

$\bar{y}$	Mean value of y
$\hat{y}$	Predicted value of y
β_0	Intercept of the linear function
β_m	Regression coefficient
γ_{nk}	Posterior probability
θ	Parameter vector of the density function
λ_n	A set of data points that maximize γ_{nk}
μ_k	Mean vector of the density function
π_k	Weight of the *k*th latent class
Σ_k	Covariance matrix
$\emptyset$	Parameter set of the mixture model

3.1. Introduction

Alberta's oil sands mining is essential to Canada's economy (Giesy et al., 2010). It has been estimated that oil sands mining would generate approximately CAD$1.7 trillion in federal and provincial taxes over the next two decades (CAPP, 2018). In oil sands mining, truck haulage is the dominant bulk material handling means for transporting ores and wastes at operating mine sites (Ma et al., 2021). The productivity of truck haulage (also referred to as truck productivity) directly relates to a mine's overall productivity (Alarie & Gamache, 2002). Accurate truck productivity prediction at operating mine sites is of great significance for making budget decisions and developing sound mine planning (Chanda & Gardiner, 2010).

To obtain accurate predictions, machine learning (ML) algorithms have received major attention and are extensively applied to various mining applications (Lei et al., 2018; Liang et al., 2020; Nourali & Osanloo, 2020; Rodriguez-Galiano et al., 2015). ML usually refers to a series of analytical data algorithms that automatically build explicit or implicit relationships between output and input variables (Fei et al., 2020). ML algorithms mainly include, but are not limited to, multiple linear regression (MLR) (Ahmed et al., 2020), decision tree (DT) (Pu et al., 2018), random forest (RF) (Rodriguez-Galiano et al., 2015), and gradient boosting regression (GBR) (Kaplan et al., 2021). Of these, MLR and DT are single learning algorithms that train a single model throughout the modelling process, while RF and GBR are known as ensemble learning algorithms (Guo et al., 2021). Ensemble learning is an ML technique that integrates several base models to gain an ensemble model with better performance than a single base model (Erdal, 2013). RF and GBR are two widely used tree-based ensemble models that integrate numerous decision trees (DTs) to improve model predictability (Dou et al., 2019). For example, Rodriguez-Galiano et al. (2015) trained an RF model (the integration of 50 trees) to forecast mineral prospectivity at mine sites.

The study showed that the prediction accuracy of the RF model was about 39% higher than that of the single DT model. Likewise, Liang et al. (2020) compared the accuracy of a GBR model (the integration of 1200 trees) and a DT model in predicting hard rock pillar stability. The results showed that the accuracy of the GBR model was 83.1%, whereas the accuracy of the DT model was 59.2%. In addition, the literature review shows that the tree-based ensemble models usually outperformed the MLR model (Ahmed et al., 2020; Lei et al., 2018). For instance, Lei et al. (2018) built two prediction models using the RF and MLR algorithms to forecast the spontaneous combustion of coal during underground coal exploitation. The relative prediction error of the RF model (2.4%) was lower than that of the MLR model (9.5%). Similarly, Ahmed et al. (2020) proposed a GBR model for predicting the calorific value of a lignite deposit in Thar, Pakistan. The study showed that the predictability of the GBR model was enhanced by 12.5% compared with the MLR model. Close to the research by Lei et al. (2018) and Ahmed et al. (2020), Sun et al. (2021) reported that the accuracy of the tree-based ensemble models (e.g., RF and GBR models) was two times higher than that of the MLR model when predicting the uniaxial compressive strength of coal-grout materials. Therefore, it is promising to apply tree-based ensemble learning algorithms to building prediction models. However, according to the current literature, no studies have reported the use of tree-based ensemble models to predict truck productivity.

To this end, the objective of this study was to develop prediction models based on the truck haulage dataset using tree-based ensemble learning algorithms to forecast truck productivity. The truck haulage dataset contained 298,608 data points. Before modelling, Gaussian mixture modelling (GMM) as a clustering approach was first used to preprocess the large dataset since GMM has been proved an efficient method for handling massive amounts of data and enhancing model prediction accuracy (Diaz-Rozo et al., 2020). After that, RF and GBR were adopted to construct prediction models of truck productivity. Also, MLR and DT as single learning algorithms were used to build models to be compared with the tree-based ensemble models. This study offered two main contributions: the first application of tree-based ensemble models to predict truck productivity and the use of GMM to further increase model predictability.

3.2. Study framework, methods, and datasets

3.2.1. Overview of study framework

Figure 3.1 shows the overview of the study framework. The truck haulage dataset collected from operating oil sand mines was divided into training (70%) and testing (30%) datasets. Before the prediction models were built, GMM clustered the training dataset into three latent classes. The labels of these latent classes constructed a latent variable that was considered to be an additional input variable (Berlin et al., 2013). Then, four ML algorithms, including RF, GBR, MLR, and DT, were used to build prediction models. Of these, the GMM-RF, GMM-GBR, GMM-MLR, and GMM-DT models were established based on the training dataset preprocessed by GMM, incorporating the input variables observed at mine sites and a latent variable. Also, the RF, GBR, MLR, and DT models were trained based on the observed input variables in the original training dataset. The built-in hyperparameters of these models were optimised by a grid search approach based on five-fold cross-validation. After that, the testing dataset was used to evaluate the performance of these eight prediction models. In this study, four metrics were chosen to quantify the prediction performance (Wu et al., 2020): the mean absolute percentage error (MAPE), the root mean square error (RMSE), the mean absolute error (MAE), and the coefficient of determination (R^2). Lastly, the RF algorithm was adopted to analyze the relative importance of the observed input variables in predicting truck productivity.

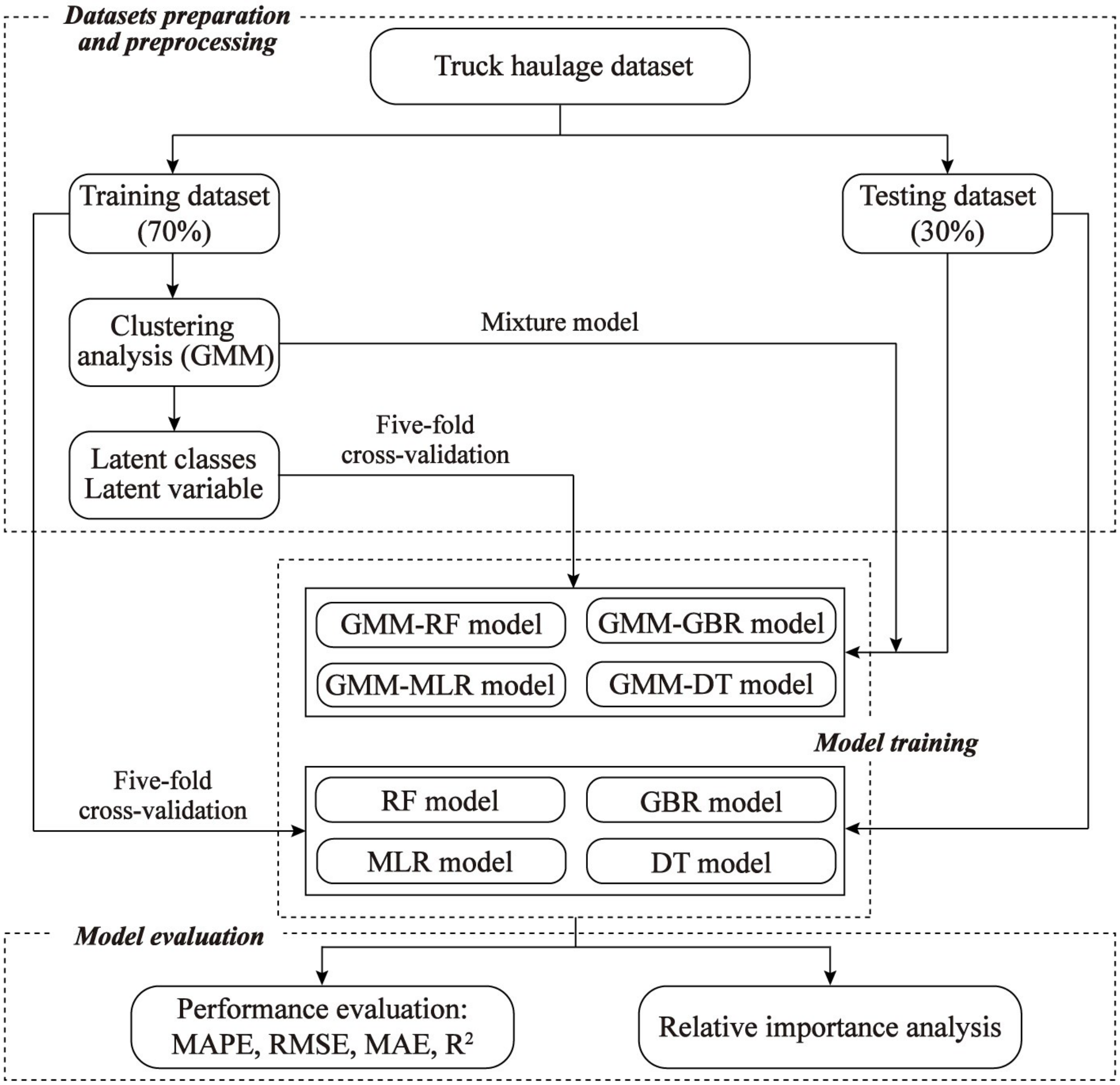


Figure 3.1 Schematic diagram of the study framework for predicting truck productivity.

3.2.2. Machine learning algorithms for building prediction models

3.2.2.1. Decision tree (DT)

The DT algorithm, often referred to as classification and regression trees (CART), was proposed by Breiman et al. (1984) and constitutes the basis of the RF and GBM models. DT is a decision-making method that uses a hierarchical tree-shaped architecture, which comprises a root node, internal nodes, leaf nodes, and branches for each node (Breiman et al., 1984). As shown in Figure

3.2, truck productivity as a continuous output variable is offered as an example to illustrate the process of using a DT model. The input variables used in this DT model were haul distance, waiting at shovel, ambient temperature, and empty speed. The numbers in each node indicated the proportion of a dataset and the mean value of the output variable (Krzywinski & Altman, 2017). For example, the root node contained the original dataset with all data points (100%) having the mean value of truck productivity (806 tph). As the tree grew, the root node split into new internal nodes representing two divided subsets. This split was based on the value of haul distance for minimizing the mean square error (MSE) in each subset (Krzywinski & Altman, 2017). As a result, subset (a) with a haul distance >= 3.8 km was in the left branch, taking 54% of the original dataset and having an average truck productivity of 742 tph; subset (b) with a haul distance < 3.8 km was in the right branch, taking 46% of the original dataset and having an average truck productivity of 880 tph. Likewise, all internal nodes obeyed this growth rule to generate new tree branches and nodes until a set of leaf nodes (also known as terminal nodes) with homogeneous datasets was created (Liang et al., 2016). The leaf nodes represent the predictions through the path from the root to the terminal. In this study, the prediction of truck productivity was a regression problem, and the prediction outcome was the mean value offered in each leaf node.

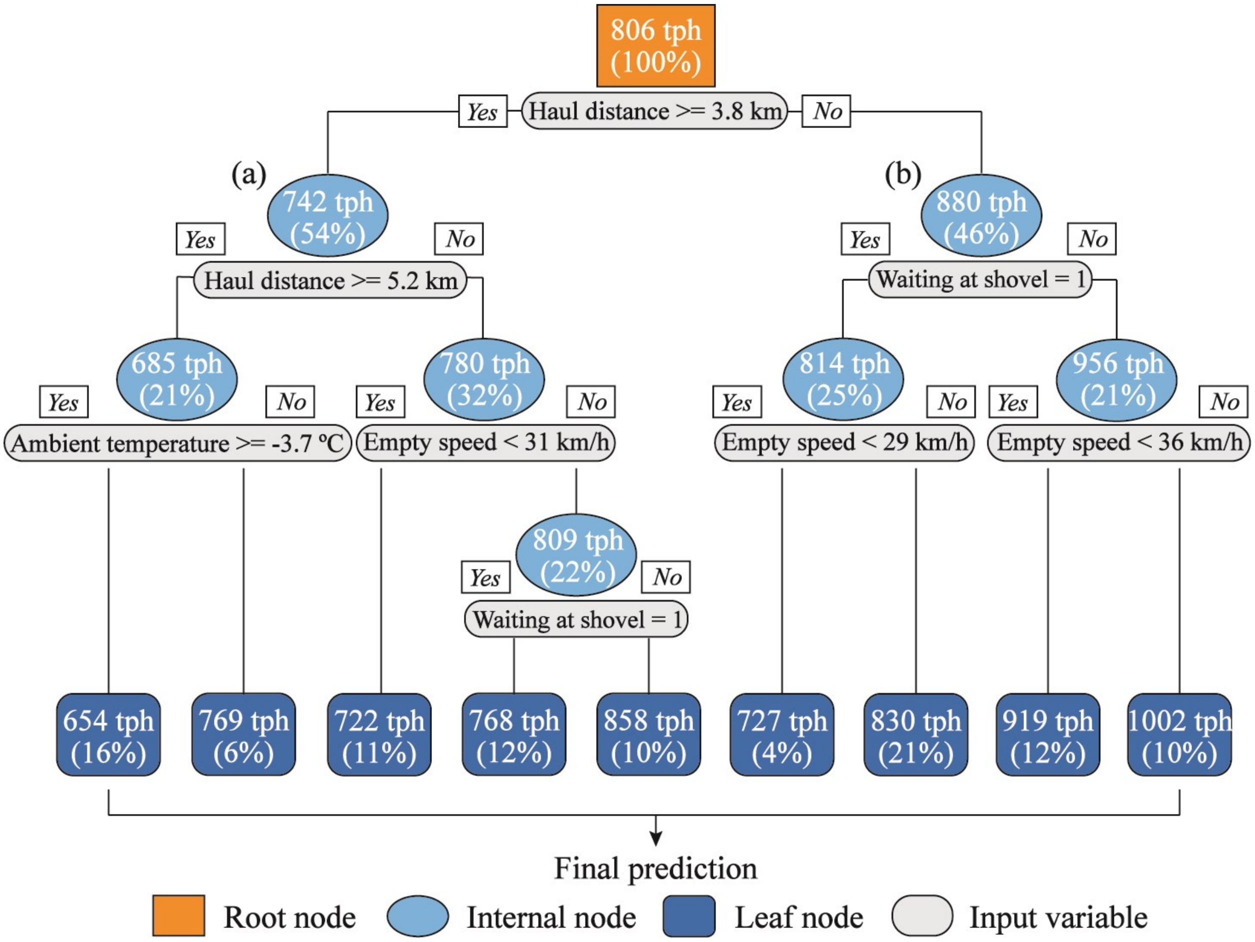


Figure 3.2 A representation of the DT model for predicting truck productivity. Two values of each node represent the proportion of the dataset belonging to this node and the mean of the output variable (e.g., truck productivity, unit: tph (tonne per hour)) in this dataset.

3.2.2.2. Random forest (RF)

The RF algorithm is an ensemble method integrating the performance of numerous DT algorithms (CART) to classify and predict outcomes (Jun & Cheng, 2017). Compared with DT, RF utilizes a bagging method to overcome the shortcomings of high variance and overfitting in the DT algorithm (Ohadi et al., 2020). Bagging is short for bootstrapping and aggregation (Breiman, 2001). As shown in Figure 3.3, bootstrapping is a sampling technique to obtain a subset of each tree by randomly resampling the original dataset with replacement. After that, a portion of input variables is randomly selected from the overall input variables as pairwise comparisons for best splitting at the root node and internal nodes in each tree. Each tree makes decisions separately, and the final prediction of the RF algorithm is reached by averaging the decisions of all trees, which is referred to as aggregation (Ohadi et al., 2020). Although this reduces the strength of each tree, it decreases the prediction variance of the RF algorithm by considering the ensemble results, thereby improving prediction accuracy (Rodriguez-Galiano et al., 2015). In addition, RF provides an additional subset as an unseen dataset to assess whether each tree is overfitted (Breiman, 2001). In bootstrapping, because of the random sampling with replacement, some data points may be used multiple times in subsets, whereas others may never be used. These data points that are not sampled for training trees are contained in an out-of-bag (OOB) subset to compute the prediction error for each tree (Peters et al., 2007). RF increases the number of trees until the error converges, thus avoiding overfitting (Rodriguez-Galiano et al., 2015). In brief, RF offers a robust ensemble algorithm through bagging technology, which is superior to the performance of the single DT algorithm.

Furthermore, the RF algorithm analyses the relative importance of input variables. The principle is that RF excludes one input variable from the overall input variables and measures the reduction in model accuracy based on the OOB error estimate, thereby determining the relative importance of this input variable. In this study, RF was adopted to determine the relative importance of input variables observed at mine sites in addition to building prediction models.

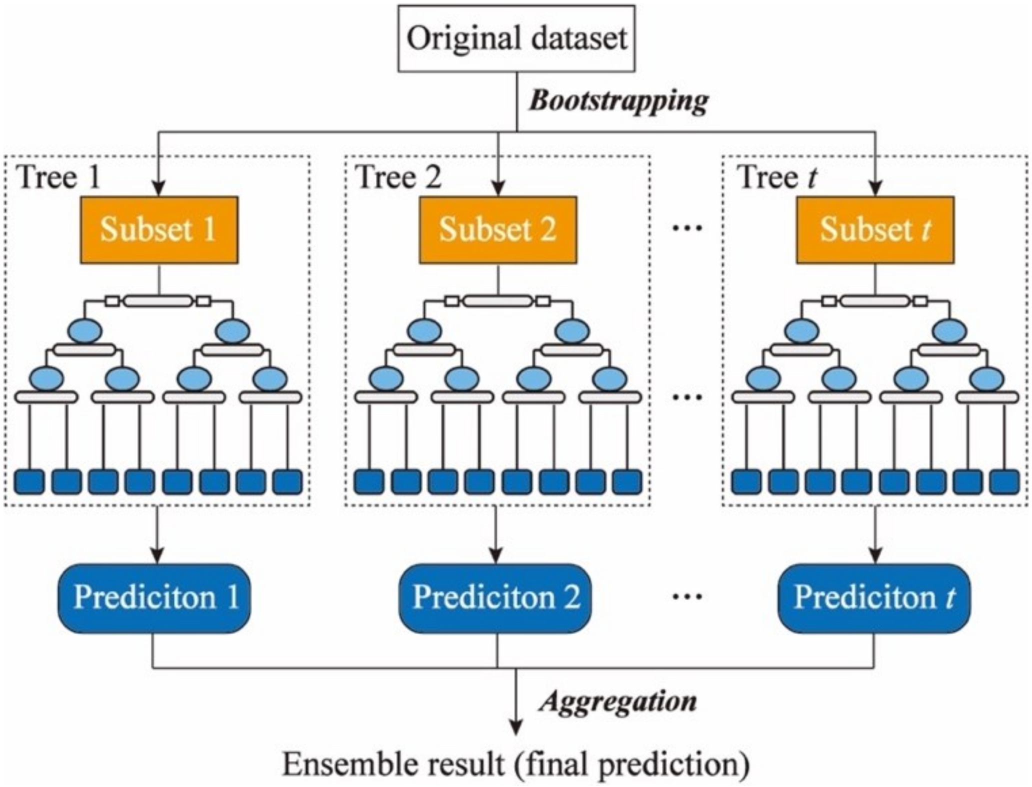


Figure 3.3 An illustration of the concept of the RF algorithm.

3.2.2.3. Gradient boosting regression (GBR)

The GBR algorithm is another ensemble approach for classification and regression problems (Friedman, 2001). Similar to RF, GBR usually combines a series of DT algorithms (CART) to enhance the performance of a single DT algorithm (Breiman, 2001). However, unlike RF, GBR adopts a boosting method rather than a bagging method to construct each tree (Friedman, 2001). Boosting is a sequential process in which each tree learns, improves, and corrects prediction errors made by preceding trees (Simsekler et al., 2021). This is different from what happens during the training stage of bagging, as each tree is built into the RF algorithm in an independent and parallel way. In boosting, every newly trained tree places emphasis on data points that have been incorrectly predicted by previous trees. To achieve an optimal combination of trees in GBR, the residual errors of these data points are specified with a loss function that is minimized through a forward stagewise (FS) strategy. In GBR, the loss function refers to the extent to which the predicted values deviate from the actual values. The FS strategy is an iterative process of minimizing the expected value of the loss function by adding new trees in sequence at each iteration without adjusting the parameters of the existing trees, also known as the functional gradient descent. The iteration of adding a new tree is terminated when the minimum average value of the loss function is acquired. Then, the successively established trees are combined into a strong ensemble learner for predicting the final result.

3.2.2.4. Multiple linear regression (MLR)

MLR is a commonly used approach for building prediction models in regression problems because of its easy calculation and explicit interpretation (Li et al., 2015). It has been utilized in predictions for many aspects at mines, such as coal production (Li et al., 2015), blast-induced ground vibration (Saadat et al., 2014), and rock fragmentation (Enayatollahi et al., 2014). Unlike DT, RF, and GBR, which establish nonlinear relationships, MLR assumes a linear relationship between a set of input variables and an output variable. This linear relationship is described as a best-fitted line, which can be acquired by minimizing the sum of squares of the vertical deviation from each data point to the line (Xie et al., 2021). In this study, the MLR model as a baseline model was compared with the models built by tree-based ensemble learning algorithms.

3.2.3. Datasets preparation and preprocessing

3.2.3.1. Datasets preparation

The truck haulage dataset was collected from oil sands mines in Northern Alberta, Canada. It contained 298,608 truck cycles generated by transporting ores and covered an entire year of truck productivity. This dataset differs slightly in size from the dataset in Chapter 2 mainly because of the deeper cleaning of the raw data as the understanding of the real-site data increased. The truck haulage dataset was proportionally and randomly divided into a training dataset (70%) and a testing dataset (30%). The training dataset was prepared for constructing prediction models based on ML algorithms. The testing dataset, as an unseen dataset, was used to test the model's prediction performance. Tables 3.1 and 3.2 show the statistical information of these two large datasets. The primary statistics included the minimum (min.), maximum (max.), median, mean, 25th percentile (Q_1), and 75th percentile (Q_3) values in each dataset. Also, both the training and testing datasets had one output variable (y) and seven input variables (x_m).

The output and input variables were directly measured at mine sites. Among them, the output variable was truck productivity (y, tph), defined as the truck payload per unit time in each truck cycle (Ercelebi & Bascetin, 2009). The observed input variables comprised the haul distance (x_1, km), empty speed (x_2, km/h), ambient temperature (x_3, °C), destination (x_4), spotting (x_5), waiting at shovel (x_6), and waiting at dump (x_7). These seven input variables were all associated with truck cycle time (Chanda & Gardiner, 2010; Fan et al., 2022), which were selected mainly based on the experience of practising engineers at mine sites and the availability of data. With the exception of the ambient temperature provided by the local weather station (MEP, 2023), the remaining input variables were provided by the mine sites. A detailed description of these seven input variables is shown in Table 3.3, where the first three inputs (x_1, x_2, and x_3) were continuous variables, and the last four inputs (x_4, x_5, x_6, and x_7) were categorical variables. Figure 3.4 shows the distribution characteristics of these input and output variables. In Figure 3.4(a)-(d), the horizontal axis represents the output and input variables, which were plotted in column charts showing specific ranges. The vertical axis shows the probability density of each continuous variable. The density refers to the portion of each range divided by the total size of data points. Notably, the density curves for the continuous variables (e.g., haul distance and ambient temperature) presented multi-peak Gaussian distributions, which indicated that the original dataset had a mixture of Gaussian distributions (Li et al., 2018). This provided the rationale for adopting GMM to preprocess the dataset in this study and will be explained in detail in Section 3.2.3.2. Figure 3.4(e)-(h) shows the boxplots of four categorical input variables and the number of data points in each label. In summary, all these observed input variables were involved in prediction models. The contribution of each observed input variable to the prediction model will be analyzed in Section 3.3.4.

Despite this, there are still limitations of the dataset in this study. Other potential input variables that have not been included can also affect truck cycle time, such as loaded speed (Cervantes et al., 2019), elevation (Chanda & Gardiner, 2010), and tire temperature (Ma et al., 2021). Depending on availability, these additional input variables may be added to future studies for building prediction models of truck productivity.

Table 3.1 Statistics, output, and input variables of the training dataset (including 209,026 data points).

Statistic	y (tph)	x_1 (km)	x_2 (km/h)	x_3 (°C)	x_4 (label)	x_5 (label)	x_6 (label)	x_7 (label)
Min.	###	1.00	5.30	-38.00	1	0	0	0
Q_1	###	3.37	31.30	-10.00	1	0	0	1
Median	###	4.61	36.80	2.90	2	1	1	1
Mean	###	4.42	36.88	0.88	2.15	0.65	0.50	0.96
Q_3	###	5.39	42.40	12.70	3	1	1	1
Max.	###	18.50	60.00	32.80	3	1	1	1

(“###”: the input information is not disclosed as it is the proprietary property of mining companies.)

Table 3.2 Statistics, output, and input variables of the testing dataset (including 89,582 data points).

Statistic	y (tph)	x_1 (km)	x_2 (km/h)	x_3 (°C)	x_4 (label)	x_5 (label)	x_6 (label)	x_7 (label)
Min.	###	1.00	5.40	-38.00	1	0	0	0
Q_1	###	3.38	31.30	-10.00	1	0	0	1
Median	###	4.61	36.80	2.80	2	1	1	1
Mean	###	4.42	36.93	0.85	2.15	0.65	0.50	0.96
Q_3	###	5.39	42.40	12.70	3	1	1	1
Max.	###	17.73	60.00	32.80	3	1	1	1

(“###”: the input information is not disclosed as it is the proprietary property of mining companies.)

Table 3.3 A detailed description of seven input variables (x_m).

Input variable	Type	Description
Haul distance (x_1, km)	Continuous	The distance for each loaded truck from a loading site to a dumping site
Empty speed (x_2, km/h)	Continuous	The speed of each empty truck returning from a dumping site to a loading site
Ambient temperature (x_3, °C)	Continuous	The ambient temperature per hour at mine sites
Destination (x_4)	Categorical	The labels (1, 2, and 3): three destinations of truck haulage at mine sites
Spotting (x_5)	Categorical	Two labels (0 and 1): zero (0) and non-zero (1) spotting time for each truck (spotting time refers to the time that a shovel with ores already has been waiting for a truck to arrive (Dzakpata et al., 2016))
Waiting at shovel (x_6)	Categorical	Two labels (0 and 1): zero (0) and non-zero (1) wait time at a shovel for each truck
Waiting at dump (x_7)	Categorical	Two labels (0 and 1): zero (0) and non-zero (1) wait time at a dumping site for each truck

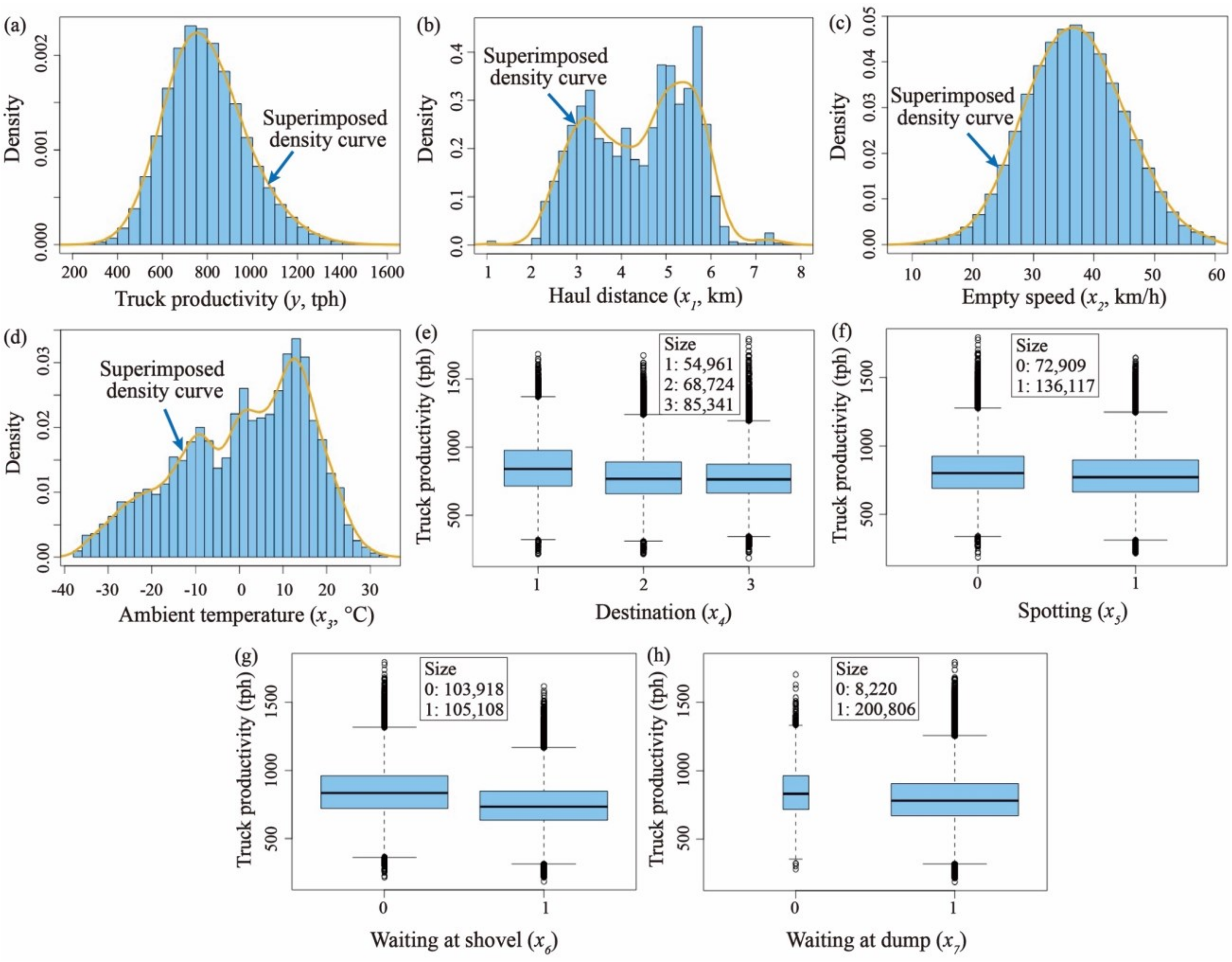


Figure 3.4 The distribution features of output and input variables in the training dataset. (a) The histogram of a continuous output: truck productivity (y, tph); (b)-(d) The histograms of three continuous inputs: haul distance (x_1), empty speed (x_2), and ambient temperature (x_3); (e)-(f) The boxplots of four categorical inputs: destination (x_4), spotting (x_5), waiting at shovel (x_6), and waiting at dump (x_7).

3.2.3.2. Gaussian mixture modeling (GMM) for datasets preprocessing

GMM is a probability distribution-based clustering approach, which has been proven to be an efficient method for preprocessing large datasets of streamflow (Ni et al., 2020), seismic activities (Kuyuk et al., 2012), and wind power (Ye et al., 2019). In GMM, data points in a large dataset are assigned into k latent classes, each of which is assumed to follow a specific Gaussian distribution (Bishop, 2006). A weighted combination of k Gaussian distributions forms a mixture of Gaussians, also known as multi-peak Gaussian distributions, which can be described as a mixture model (Leisch, 2004):

$$P(y|x, \emptyset) = \sum_{k=1}^{K} \pi_k f_k(y|x, \theta_k) \tag{3-1}$$

where $P(y|x, \emptyset)$ is the mixture model indicating the probability density function (PDF) of the data population; $f_k(y|x, \theta_k)$ represents the PDF of the kth latent class; $\emptyset$ denotes the parameter set $\{\pi_k, \theta_k\}$. Of these, π_k is the non-negative weight of the kth class together with $\sum_{k=1}^{K} \pi_k = 1$, θ_k is the parameter vector (μ_k, Σ_k), and μ_k and Σ_k are the mean vector and covariance matrix, respectively.

To obtain the mixture model, GMM first estimates the parameter set $\{\pi_k, \theta_k\}$ and then determines the optimal number of the latent classes. The parameter estimation is generally carried out using an expectation-maximization (EM) algorithm (Leisch, 2004), which is divided into two steps. In *E*-step, the posterior probability is calculated for a data point (x_i, y_i) assigned to each class (Leisch, 2004):

$$\gamma_{nk} = \frac{\pi_k f_k(y_n|x_n, \theta_k)}{\sum_{k=1}^{K} \pi_k f_k(y_n|x_n, \theta_k)} \tag{3-2}$$

This data point belongs to the kth class when

$$\lambda_k = \underset{k \epsilon \{1,2,\ldots,K\}}{\operatorname{argmax}} \gamma_{nk} \tag{3-3}$$

where λ_k is the set of data points that has the maximum posterior probability, γ_{nk}. With the γ_{nk}, $\{\pi_k, \theta_k\}$ can be further estimated in the *M*-step by maximizing the log-likelihood (*log L*) in Equation (3-5) (Leisch, 2004):

$$\pi_k = \frac{1}{N} f_k = \frac{1}{N} \sum_{n=1}^{N} \gamma_{nk} \tag{3-4}$$

$$log\,L = \sum_{n=1}^{N} \log(P(y|x, \phi)) = \sum_{n=1}^{N} \log(\sum_{k=1}^{K} \pi_k f_k(y|x, \theta_k)) \tag{3-5}$$

where *N* is the number of data points. The *E*- and *M*-steps are iteratively computed until the maximum *log L* is reached. Later, GMM starts to determine the optimal number (k) of latent classes by minimizing the Bayesian information criterion (BIC) value (Lu et al., 2019):

$$BIC = -2logL + ClogN \tag{3-6}$$

where *C* is the number of estimated parameters. The BIC method was adopted as a metric since it has been shown to outperform other methods in a rigorous study (Russell & Raftery, 2009). Finally, the mixture model is obtained from the data population, representing the multi-peak Gaussian distribution.

According to the central limit theorem (Rice, 1995), the observations of many variables in engineering often present multi-peak Gaussian distributions. This applies to variables (e.g., the haul distance in Figure 3.4) in truck haulage datasets observed from oil sands mines (Cervantes et al., 2019). Relying on these peaks, GMM can recognize latent classes and generate latent variables, thereby improving model accuracy (Sonta et al., 2018). Thus, GMM was adopted in this study to preprocess large datasets before prediction models were built.

3.2.4. Performance evaluation for prediction models

To investigate the predictability of tree-based ensemble models, four ML models (RF, GBR, MLR, and DT models) were established for comparison. Of these, the RF and GBR models were tree-based ensemble models, while the MLR and DT models were baseline models. Moreover, to assess the effect of GMM on model performance, four additional ML models (GMM-RF, GMM-GBR, GMM-MLR, and GMM-DT models) were constructed on the training dataset preprocessed by GMM for comparison. This study used four performance metrics to evaluate these eight prediction models, including MAPE, RMSE, MAE, and R^2 (Wu et al., 2020). They are written as follows:

$$MAPE = \frac{100\%}{N}\sum_{n=1}^{N}|\frac{y_n-\hat{y}_n}{y_n}| \tag{3-7}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(y_n-\hat{y}_n)^2} \tag{3-8}$$

$$MAE = \frac{1}{N}\sum_{n=1}^{N}|y_n-\hat{y}_n| \tag{3-9}$$

$$R^2 = 1-\frac{\sum_{n}^{N}(y_n-\hat{y}_n)^2}{\sum_{n}^{N}(y_n-\bar{y}_n)^2} \tag{3-10}$$

where y_n is the measured values; $\hat{y}_n$ is the predicted values; and $\bar{y}_n$ is the mean value of the measured values. MAPE shows the percentage of error relative to the measured values, and RMSE means the standard deviation of the residuals between the measured and predicted values. MAE is the absolute error between the measured and predicted values, and R^2 indicates the degree to which data points fit a curve, ranging from 0 to 1 (Wu et al., 2020). Overall, the prediction model with a higher R^2 and a lower MAPE, RMSE, and MAE has better performance.

3.2.5. Hyperparameters tuning

Before applying the proposed ML algorithms for predictions, built-in hyperparameters are required to be pre-tuned to improve the performance of prediction models (Xue et al., 2021). In this study, the main goal of the hyperparameters tuning is to control the complexity of prediction models, making the model less overfitting (Ohadi et al., 2020). Table 3.4 presents the hyperparameters that need to be tuned and their search space. For the RF algorithm, the hyperparameters were *mtry* (the number of input variables available for splitting at each node) and *min.node.size* (the minimum number of observations in a leaf node). For the GBR algorithm, the hyperparameters were *ntrees* (the total number of trees in GBR), *interaction.depth* (the number of splits in each tree), *shrinkage* (learning rate), and *n.minobsinnode* (the minimum number of observations in a leaf node).

To obtain the optimal hyperparameters, a grid search method was adopted in this study since it is easy to implement and has a sound optimization effect (Erdogan Erten et al., 2021). First, the grid search method defines a search space of hyperparameters as a grid. After that, a validation dataset is split from the training dataset and subjected to *t*-fold cross-validation to establish and evaluate prediction models based on every position in the grid. The prediction models are constructed using *t* - 1 folds and tested using the remaining one-fold, which repeats t times with different folds used as the testing fold. Finally, the *t*-fold cross-validation performance (the RMSE value used as a metric (Sun et al., 2021)) is the average performance calculated in each fold (Qi & Tang, 2018). In this study, *t* was set to be five, which were recommended by Liang et al. (2020) and Wu et al. (2020).

Table 3.4 Hyperparameters and their search space for the proposed algorithms.

Algorithm	Hyperparameter	Type	Range, step
RF	*mtry*	Integer	[1-*m*[*]], 1
	min.node.size	Integer	[4-30], 2
GBR	*n.trees*	Integer	[400-2800], 200
	interaction.depth	Integer	[2-10], 1
	shrinkage	Float	[0.05-0.25], 0.05
	n.minobsinnode	Integer	[4-14], 1

[*] *m*: the total number of input variables for predicting truck productivity.

3.3. Results and discussion

3.3.1. GMM preprocessing

Figure 3.5 shows the three latent classes identified from the training dataset using GMM preprocessing. Taking truck productivity as an example, the vertical axis shows the value of truck productivity in each class, and the horizontal axis indicates the labels of three latent classes (i.e., 1, 2, and 3). In addition, the statistical information of the latent classes is listed in Figure 3.5. Class 2 had 93,543 data points, which was more than Class 1 (63,117) and Class 3 (52,366). Q_1 and Q_3 represent the distribution interval of data points in each class. In Class 3, the value of truck productivity was between 875 tph (Q_1) to 1,095 tph (Q_3), with the median and mean values of 975 tph and 981 tph, respectively. For Classes 2 and 1, the ranges, median, and mean values of truck productivity successively decreased. This means the value of truck productivity varied considerably in these three latent classes, in the order of Class 1 < Class 2 < Class 3, representing the low, medium, and high truck productivity at operating oil sands mines. This study was similar to the research by Lu et al. (2019), in which they identified six latent classes from heating load data using GMM and built a prediction model based on the GMM preprocessing. That research showed that the model accuracy was improved by about 20%. Hence, preprocessing of large datasets using GMM in this study had the potential to increase model predictability.

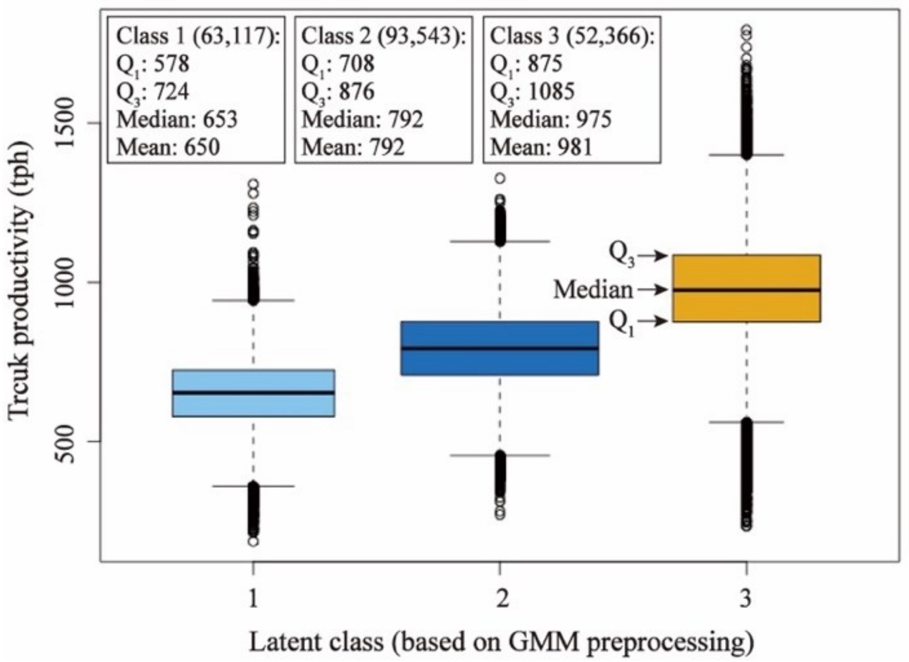


Figure 3.5 Identification of three latent classes (represented by labels of 1, 2, and 3) from the training dataset using GMM.

3.3.2. Determination of hyperparameters

In this study, the hyperparameters of the RF and GBR algorithms were tuned by grid search based on the five-fold cross-validation. To assess the effectiveness of using grid search, the RMSE was calculated for each combination of hyperparameters in the prescribed search space (Sun et al., 2021). As shown in Figure 3.6, the RF algorithm (including the RF and GMM-RF models) was taken as an example to illustrate the process of obtaining the optimal hyperparameters. In Figure 3.6(a), for the RF model, the search range of *mtry* was set as [1, 7]. When *mtry* approached two, the RMSE value significantly dropped (from 154.6 to 137.3). It further decreased to 133.2 when *mtry* reached three and could not be reduced with the number of input variables. Thus, the optimal *mtry* for the RF model was three. Similarly, the other hyperparameter *min.node.size* (search range was set as [4, 30]) was determined to be 22 when the minimum RMSE value was attained. The GMM-RF model was built based on the training dataset after implementing GMM and had the same built-in hyperparameter as the baseline model (i.e., the RF model). In Figure 3.6(c)-(d), the optimal *mtry* and *min.node.size* was selected to be 4 and 16 based on the minimum RMSE values. In this study, the number of trees was not tuned in the RF algorithm because the increase in the number insignificantly improved the model performance and increased the computational cost. As a result, the number of trees was set to the default value (500) for the RF and GMM-RF models. Furthermore, this study tuned four hyperparameters for the GBR algorithm. The determination of these is shown in Table 3.5 for the GBR and GMM-GBR models, respectively. The results were similar to the studies by Naghibi et al. (2017) and Yu et al. (2020) in which the hyperparameters tuning reduced the risk of overfitting and improved model accuracy to some extent. The results showed that the accuracy of tuned RF and GBR models was 85.6% and 99.9%, which was higher than that of untuned RF (84.6%) and GBR (98.9%) models.

Table 3.5 Determination of hyperparameters for the GBR model and the GMM-GBR model.

Algorithm	*ntrees*	*interaction.depth*	*shrinkage*	*n.minobsinnode*
GBR	2600	7	0.15	6
GMM-GBR	600	8	0.15	14

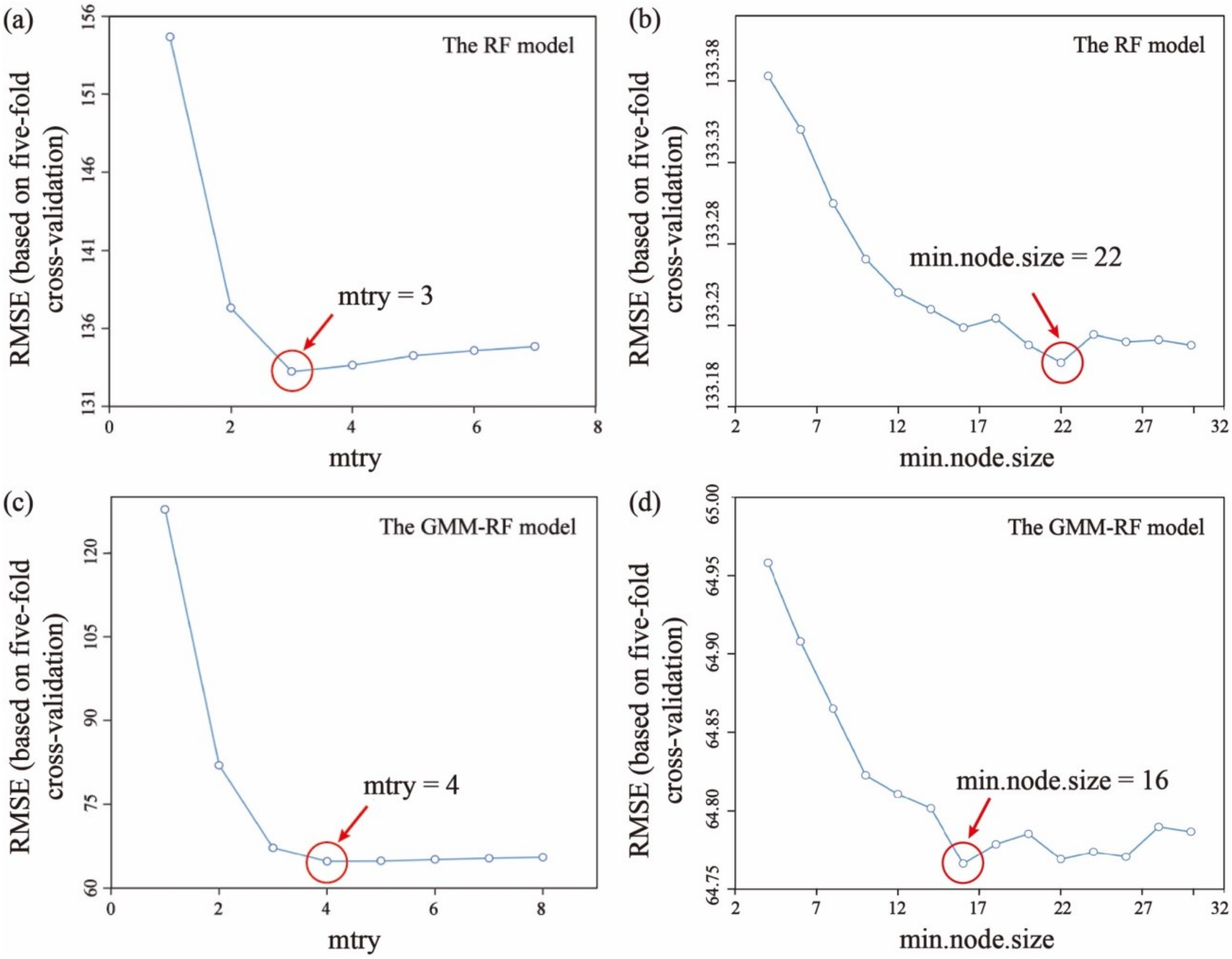


Figure 3.6 Determination of hyperparameters for the RF model and the GMM-RF model.

3.3.3. Performance comparison and evaluation of prediction models

3.3.3.1. Comparing RF, GBR, MLR, and DT models

Figure 3.7 shows the scatterplots of the predicted truck productivity (on the vertical axis) obtained from the RF, GBR, MLR, and DT models and the measured truck productivity (on the horizontal axis) in the testing and training datasets. The smaller deviation between the predicted and measured values, the closer the scatter points along the *y* = *x* line (Liu et al., 2020). As shown in Figure 3.7, the scatter points generated from the four prediction models were not closely distributed along both sides of the line, indicating that the prediction performance of these models was not high.

Table 3.6 lists quantitative metrics based on the testing and training datasets to evaluate the prediction performance. From Table 3.6, in terms of the testing dataset, the MAPE, RMSE, MAE, and R^2 were 14.1%, 134.29, 104.66 and 44.05% for the RF model, and 13.96%, 133.42, 103.78, and 44.76% for the GBR model. Accordingly, these metrics were 15.10%, 143.56, 112.02, and 36.06% for the MLR model, and 15.68%, 148.08, 116.01, and 31.96% for the DT model. Therefore, the RF and GBR models had lower MAPE, RMSE, and MAE, and higher R^2 than those of the MLR and DT models, indicating that the tree-based ensemble models performed better than the single models in predicting truck productivity. A similar finding was reported in the study by Zhang et al. (2021), who established two tree-based ensemble models (the RF and GBR models) to estimate diaphragm wall deflections in anisotropic clays. The results showed that the R^2 of the RF (98.2%) and GBR (98.9%) models was higher than the single DT model (91.4%). To conclude, although the RF and GBR models were not as accurate, they outperformed the MLR and DT models in predicting truck productivity.

Table 3.6 Performance of the RF, GBR, MLR, and DT models on testing and training datasets.

Prediction model	Testing dataset				Training dataset			
	MAPE (%)	RMSE	MAE	R^2 (%)	MAPE (%)	RMSE	MAE	R^2 (%)
RF	14.10	134.29	104.66	44.05	13.50	128.83	100.43	48.31
GBR	13.96	133.42	103.78	44.76	13.58	129.62	101.06	47.67
MLR	15.10	143.56	112.02	36.06	15.01	143.34	111.67	36.00
DT	15.68	148.08	116.01	31.96	15.59	147.76	115.64	32.00

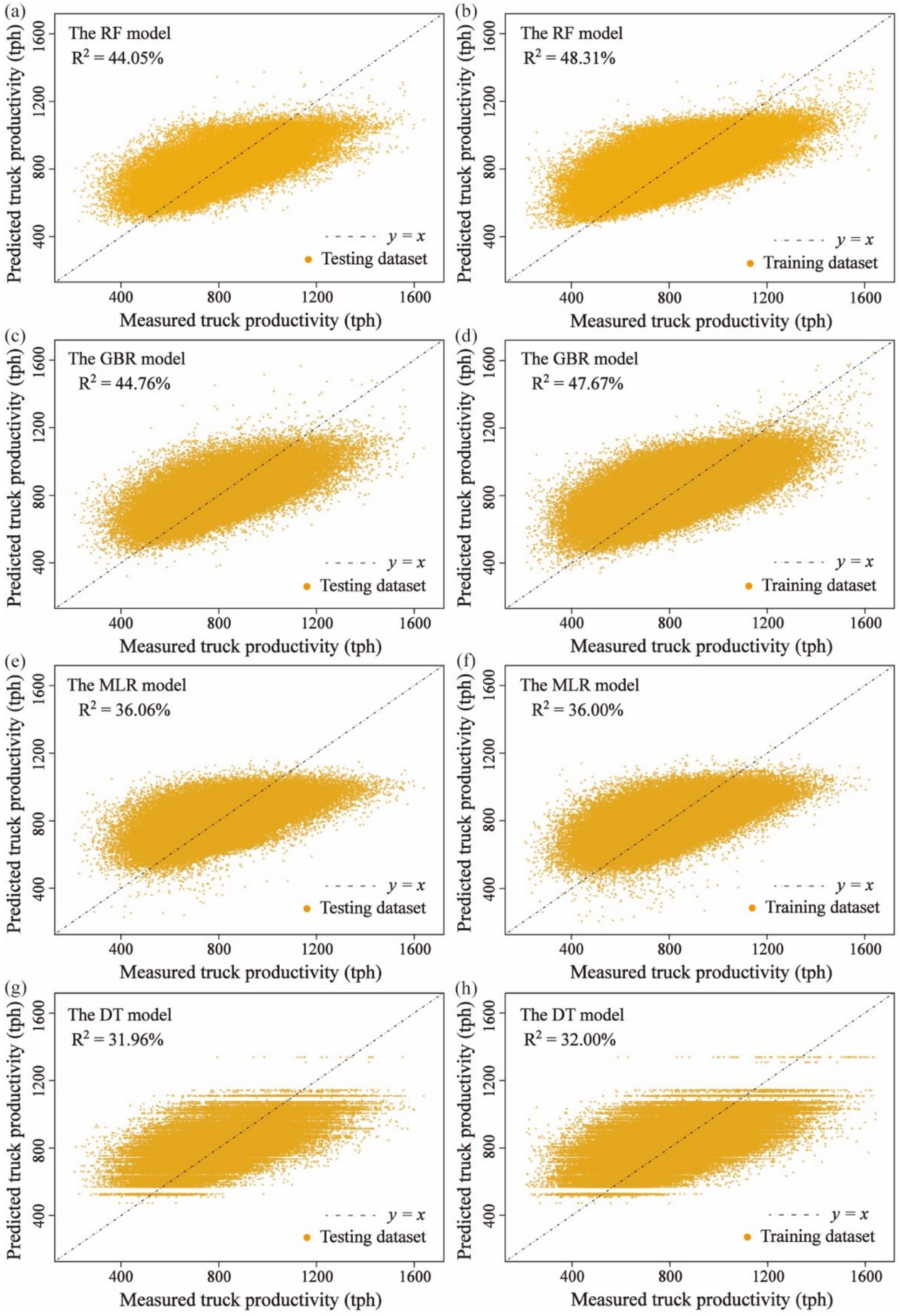


Figure 3.7 Scatterplots of the measured truck productivity and predicted truck productivity. The

RF model evaluated by (a) testing dataset and (b) training dataset; the GBR model evaluated by (c) testing dataset and (d) training dataset; the MLR model evaluated by (e) testing dataset and (f) training dataset; and the DT model evaluated by (g) testing dataset and (h) training dataset.

3.3.3.2. Comparing GMM-RF, GMM-GBR, GMM-MLR, and GMM-DT models

Figure 3.8 shows the scatterplots of the predicted truck productivity obtained from the GMM-RF, GMM-GBR, GMM-MLR, and GMM-DT models and the measured truck productivity in the testing and training datasets. As shown in Figure 3.8, the scatter points generated from these four prediction models were densely distributed along the $y = x$ line, indicating that these prediction models performed well in predicting truck productivity. This was related to the GMM preprocessing, and its effect on model performance will be evaluated in detail in Section 3.3.3.3. Table 3.7 lists quantitative metrics based on the testing and training datasets to evaluate the prediction performance. From Table 3.7, in terms of the testing dataset, the GMM-RF model had the lowest MAPE, RMSE, and MAE, and the highest R^2, with values of 6.77%, 64.33, 49.78, and 87.16%. Its performance was slightly higher than the GMM-GBR (6.81%, 64.77, 50.10, and 86.98%), and superior to the GMM-MLR (7.61%, 77.06, 55.17, and 81.57%) and GMM-DT models (8.61%, 82.28, 63.35, and 78.99%). Thus, the tree-based ensemble models still outperformed the single models in predicting truck productivity. Akin to the study by Chen et al. (2021): after implementing GMM, the accuracy of the ensemble model was 12% higher than the single model in predicting dam deformation.

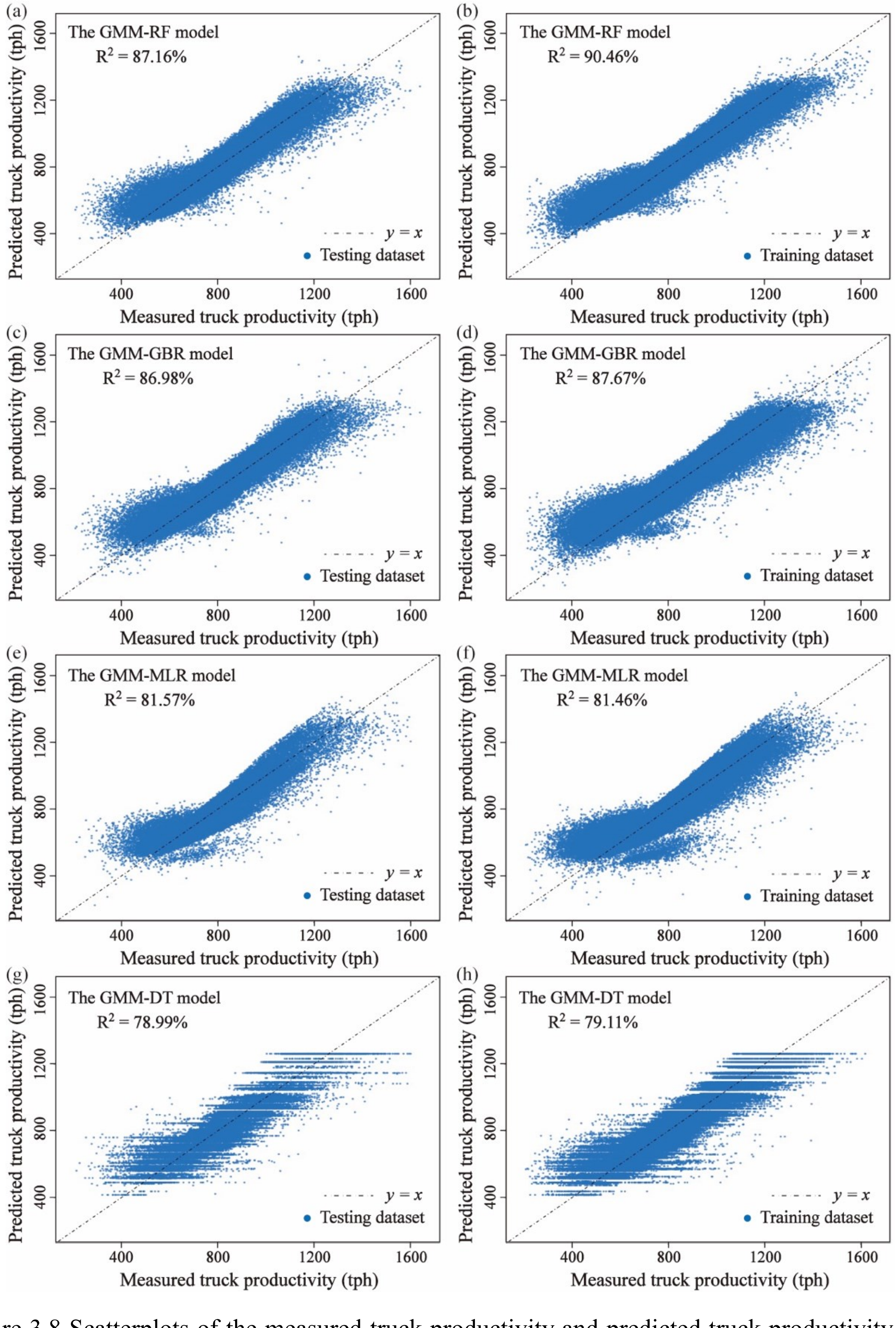


Figure 3.8 Scatterplots of the measured truck productivity and predicted truck productivity. The

GMM-RF model evaluated by (a) testing dataset and (b) training dataset; the GMM-GBR model evaluated by (c) testing dataset and (d) training dataset; the GMM-MLR model evaluated by (e) testing dataset and (f) training dataset; and the GMM-DT model evaluated by (g) testing dataset and (h) training dataset.

Table 3.7 Performance of the GMM-RF, GMM-GBR, GMM-MLR, and GMM-DT models on testing and training datasets.

Prediction model	Testing dataset				Training dataset			
	MAPE (%)	RMSE	MAE	R^2 (%)	MAPE (%)	RMSE	MAE	R^2 (%)
GMM-RF	6.77	64.33	49.78	87.16	5.89	55.36	43.19	90.46
GMM-GBR	6.81	64.77	50.10	86.98	6.63	62.92	48.90	87.67
GMM-MLR	7.61	77.06	55.17	81.57	7.60	77.16	55.21	81.46
GMM-DT	8.61	82.28	63.35	78.99	8.55	81.90	62.98	79.11

3.3.3.3. Effect of implementing GMM on model performance

Figure 3.9 shows the performance comparisons of the trained models with and without implementing GMM preprocessing. For the ensemble models, the GBR and GMM-GBR models were used as examples. In Figure 3.9(b), the GMM-GBR model had lower MAPE, RMSE, and MAE, and a higher R^2, with the values of 6.81%, 64.77, 50.10, and 86.98%. These four metrics of the GBR model were 13.96%, 133.42, 103.78, and 44.76%. In terms of the R^2, the accuracy of the GMM-GBR model (R^2 = 86.98%) was about two times higher than the GBR model (R^2 = 44.76%). Therefore, the GMM-GBR model performed better than the GBR model in predicting truck productivity. Similar research by Ni et al. (2020) proposed a GBR model coupled with GMM for monthly low streamflow forecasting. The research showed that the R^2 of the proposed model was improved by 12% compared to the GBR model without GMM preprocessing. Also, for the single models, the MLR and GMM-MLR models were used as examples. In Figure 3.9(c), in terms of the R^2, the accuracy of the GMM-MLR model ($R^2 = 81.57\%$) was over two times higher than the MLR model ($R^2 = 36.06\%$). In other words, the GMM-MLR model performed well in predicting truck productivity. This information can provide new solutions for mining engineers to handle engineering data with multi-peak Gaussian distributions at other mine sites and to build more accurate prediction models. To conclude, GMM considerably improved the performance of the ensemble and single models by involving a latent variable. The latent variable was a class-related categorical variable constructed by the labels of the latent classes. This agreed with studies by Lunt (2015) and Kyburz et al. (2011) in which a latent variable was constituted by the labels of early and late treated classes for patients and was used in conjunction with other input variables to analyze radiographic damage scores accurately.

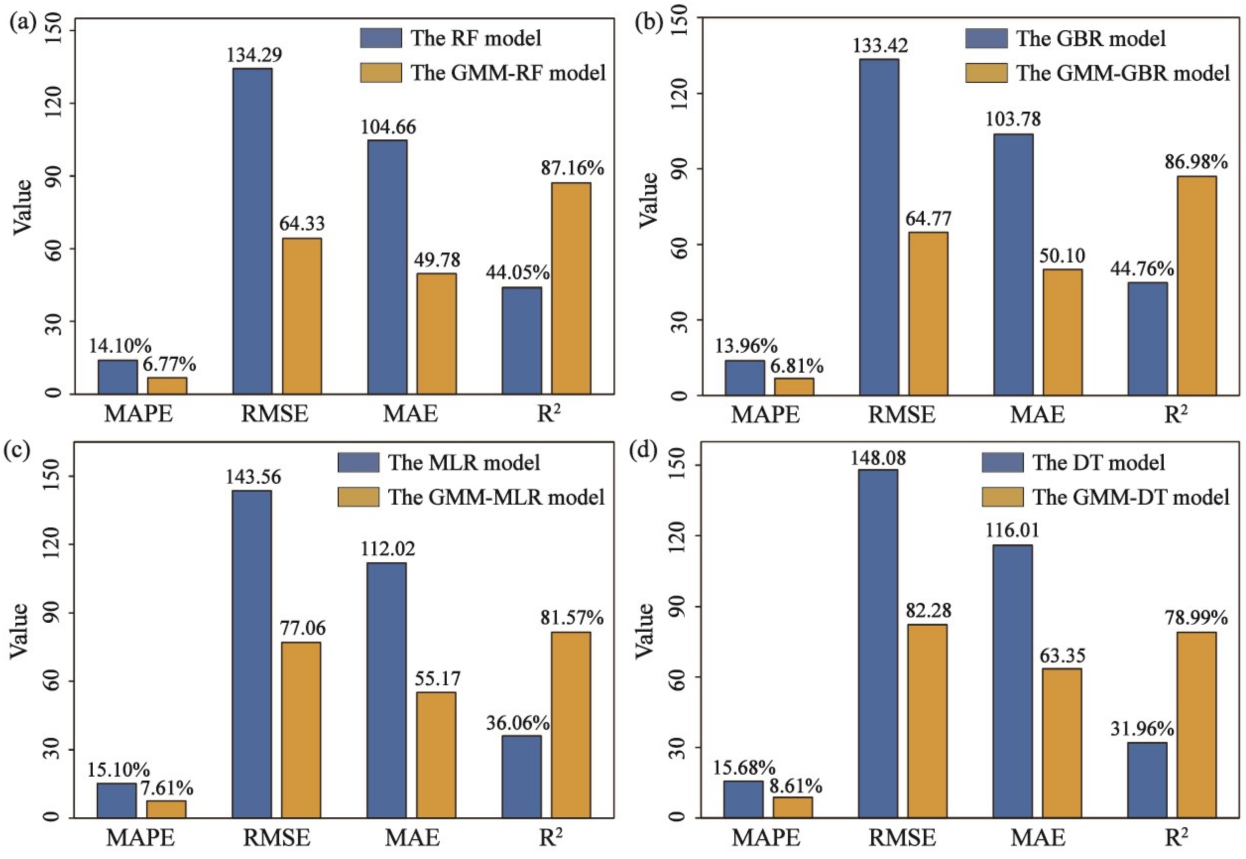


Figure 3.9 Performance comparisons of the trained models with and without implementing GMM preprocessing based on the testing dataset. (a) comparison between the RF and GMM-RF models; (b) comparison between the GBR and GMM-GBR models; (c) comparison between the MLR and GMM-MLR models; (d) comparison between the DT and GMM-DT models.

3.3.4. Relative importance of observed input variables

In Section 3.3.3.2, the performance of the GMM-RF model was seen to be slightly higher than that of the GMM-GBR model, which was significantly better than the GMM-MLR and GMM-DT models. This means that among all models in this study, the GMM-RF model was the most accurate in predicting truck productivity. The accuracy of prediction models is closely related to input variables (Wu et al., 2020). The input variables observed at mine sites directly represented the actual truck haulage process. Thus, this study focused on analyzing the contributions of the observed input variables to the GMM-RF model.

Figure 3.10 shows the relative importance (in percentage) of the observed input variables in the GMM-RF model. The relative importance of these seven input variables was ranked as haul distance (43.51%) > empty speed (21.77%) > waiting at shovel (18.18%) > ambient temperature (12.71%) > destination (2.29%) > spotting (1.03%) > waiting at dump (0.52%). Among these input variables, haul distance had the highest relative importance, indicating that it was the most crucial input variable for predicting truck productivity. It was reported by Cervantes et al. (2019) that mining companies generally observed that haul distance has the most impact on truck productivity and often built a fitted line between truck productivity and haul distance. This is because an increase in haul distance directly causes an increase in cycle time, thereby decreasing truck productivity. Next, empty speed, with a relative importance of 21.77%, played the second most important role in predicting truck productivity since empty speed determines a part of cycle time (the travel time from a dumping site to a loading site), thus affecting truck productivity (Schexnayder et al., 1999). After empty speed, waiting at shovel as a categorical input variable had the third-highest relative importance (18.18%). According to Ercelebi and Bascetin (2009), when truck fleet size increased from three to five, the wait time at shovel increased from 2.48 min to 3.11 min, and shovel utilization was reduced from 0.78 to 0.72. This led to an increase in cycle time and a decrease in truck productivity. In their research, wait time at dump varied from 0.54 min to 0.59 min, indicating that the wait time at dump was shorter than the wait time at shovel. Similar to Soofastaei et al. (2016), the spot time is usually around 0.5 min, with only a small impact on cycle time. Those on-site observations showed that the wait time at shovel took longer than the wait time at dump and spot time, which had a more pronounced effect on cycle time or truck productivity. This applies in the current study: waiting at shovel (18.18%) on truck productivity was more significant to truck productivity than spotting (1.03%) and waiting at dump (0.52%). Then, the relative importance of ambient temperature was 12.71%, which contributed a lot to the model accuracy. Sun et al. (2018) also reported that the model accuracy was enhanced by 5.13% when it included the meteorological factor. Lastly, destination made a small contribution (2.29%) to the GMM-RF model since it cannot directly affect the cycle time or truck payload (Navarro Torres et al., 2019). Overall, the input variables contributed differently to the GMM-RF model, with haul distance being the most influential input variable. Also, mining engineers can quantitatively assess and select valid input variables based on their relative importance in order to accurately predict truck productivity.

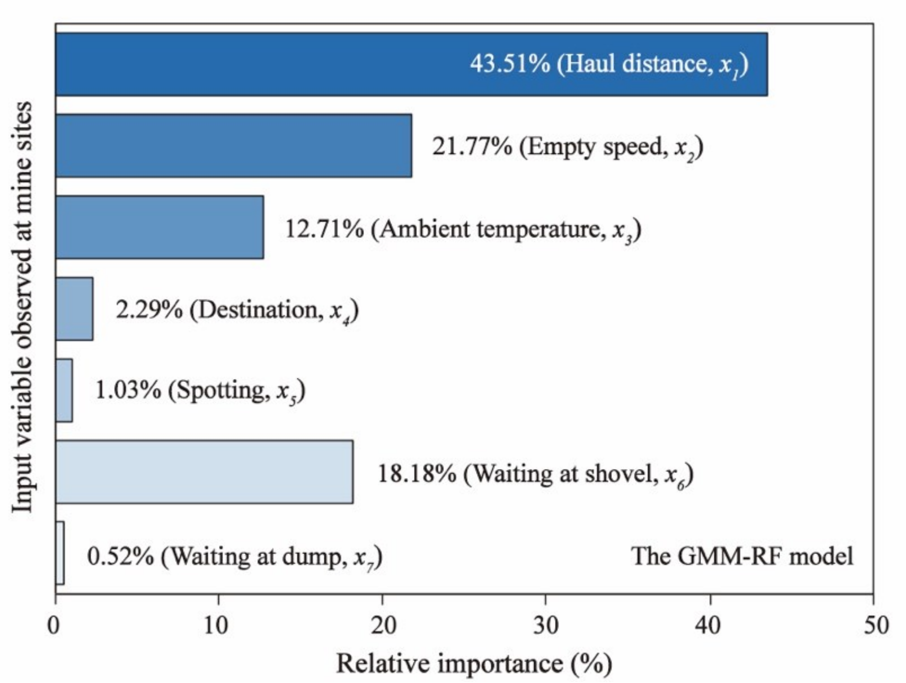


Figure 3.10 Relative importance analysis of input variables observed at mine sites.

3.4. Conclusions

For the first time, this study used random forest (RF) and gradient boosting regression (GBR) models to predict truck productivity at mine sites and adopted Gaussian mixture modelling (GMM) to further improve the model predictability. The main conclusions are summarised below:

(1) The tree-based ensemble models performed better than single models in predicting truck productivity (without and with GMM preprocessing). For example, without GMM preprocessing, the R^2 of the RF model was 44.05%, which was higher than that of the decision tree model (the DT model), with a value of 31.96%. With GMM preprocessing, the R^2 of the GMM-RF model (87.16%) remained higher than the GMM-DT model (78.99%).

(2) GMM significantly increased the predictability of truck productivity prediction models (both tree-based ensemble models and single models) by considering a latent variable. For instance, the R^2 of the GMM-GBR model (86.98%) was about two times higher than the GBR model (44.76%). Also, the R^2 of the GMM-MLR model (81.57%) was over two times higher than the MLR model (36.06%). This information can provide new solutions for mining engineers to handle engineering data with multi-peak Gaussian distributions at other mine sites and to build more accurate prediction models.

(3) Based on-site observation, haul distance was the most influential variable among the observed input variables in predicting truck productivity. The relative importance of haul distance was 43.51%, which was higher than empty speed (21.77%), waiting at shovel (18.18%), ambient temperature (12.71%), destination (2.29%), spotting (1.03%), and waiting at dump (0.52%). This information helps mining engineers select valid input variables to accurately predict truck productivity.

(4) The final proposed prediction models were the highly accurate GMM-RF and GMM-GBR models. In this study, the GMM-RF and GMM-GBR models had higher R^2, with values of 87.16% and 86.98%, indicating that these two models had the potential to accurately predict truck productivity at mine sites.

Chapter 4. Weighted ensembles of artificial neural networks based on Gaussian mixture modeling for truck productivity prediction at open-pit mines

This chapter has been published as **C. Fan**, N. Zhang, B. Jiang, W.V. Liu, Weighted ensembles of artificial neural networks based on Gaussian mixture modeling for truck productivity prediction at open-pit mines, *Mining, Metallurgy & Environment*. © Springer. 40 (2023) 583-598.

Nomenclatures

ANN	Artificial neural network
aj	Output of the hidden layer
b_j	Bias term between input and hidden neurons
b_l	Bias term between hidden and output neurons
BIC	Bayesian information criteria
BPNN	Back propagation neural network
BRNN	Bayesian regularized neural network
C	Mixture model complexity
D	Data point (x, y)
DT	Decision tree
ELM	Extreme learning machine
EM	Expectation-maximization
F	Loss function
f_k	Probability density function
$f(w\|D, \mu, \sigma)$	Posterior probability based on Bayesian theorem
$f(w\|\mu)$	Prior probability for the weights

$f(D\|w,\sigma)$	Likelihood function
$f(D\|\mu,\sigma)$	Normalization factor
GBR	Gradient boosting regression
GMM	Gaussian mixture modeling
k	The *k*th latent classes
K	The number of latent classes
L	Likelihood of a set of data points
m	The *m*th input variable
MAE	Mean absolute error
ML	Machine learning
n	The *n*th data point
N	The number of data points
P	Mixture model
PC	Personal computer
Q_1, Q_3	The 25th and 75th percentiles
R^2	Coefficient of determination
RMSE	Root mean square error

tph	Tonnes per hour
WE-BPNN	Weighted ensemble-back propagation neural network
WE-BRNN	Weighted ensemble-Bayesian regularized neural network
WE-ELM	Weighted ensemble-extreme learning machine
w_k	The posterior probability for class *k*
w_{mj}	Weights between input neurons and hidden neurons
w_{jl}	Weights between hidden neurons and output neurons
XGBoost	Extreme gradient boosting
x_m	The *m*th input variable
y	Output variable
$\bar{y}$	Mean value of *y*
$\hat{y}$	Predicted value of *y*
β_0	Intercept of the linear function
β_m	Regression coefficient
γ_{nk}	Posterior probability
θ	Parameter vector of the density function
λ_n	A set of data points that maximize γ_{nk}

μ_k	Mean vector of the density function
π_k	Weight of the *k*th latent class
Σ_k	Covariance matrix
$\emptyset$	Parameter set of the mixture model
Ω	Regularization penalty term

4.1. Introduction

In open-pit mining operations, truck haulage is the dominant bulk material handling means for transporting ores (Ma et al., 2021). The productivity of truck haulage (or referred to as "truck productivity"), defined as the truck payload per unit time in each truck haulage cycle, is of great interest to mining companies because it directly relates to mine production, operations, and planning (e.g., truck-shovel scheduling, fleet sizing, budget decisions, and employment) (Chanda & Gardiner, 2010; Upadhyay et al., 2020).

As for predicting truck productivity, many simulation models and algorithms have been proposed by researchers based on sequential tasks performed by trucks (Baek & Choi, 2019). These simulation models and algorithms include, but are not limited to discrete-event simulation models (Moradi Afrapoli et al., 2019), queuing theory (Sembakutti et al., 2017), goal programming (Upadhyay et al., 2020), and stochastic programming (Rimélé et al., 2020). Nevertheless, there are problems with these methods because of unexpected events during truck haulage, such as extreme weather and shovel availability reduction. To ensure accurate simulations, these models and algorithms need to be continually updated, resulting in increased time and labor costs (Baek & Choi, 2019).

In response to these problems, machine learning (ML) based on massive real-site data rather than simulation methods has been initiated as a new research direction (Fan et al., 2022, 2023b). ML is a collective name for a series of data-driven algorithms that automatically extract knowledge from massive amounts of raw data and model complex relationships between inputs and outputs (Pu et al., 2019). Among various ML methods, artificial neural networks (ANNs) are well-known algorithms that are inspired by interconnected neurons in biological neural networks (Rana et al., 2020). Currently, commonly used ANNs usually include back propagation neural network (BPNN)

(Wu et al., 2020), extreme learning machine (ELM) (Sattar et al., 2019), and Bayesian regularized neural network (BRNN) (Demirbay et al., 2020). These ANNs have been extensively applied to many aspects of mining engineering because of their strong ability to map nonlinear relationships between input and output variables, thus providing robust predictions (Nguyen et al., 2020; Thai et al., 2021; Trivedi et al., 2014; Xue et al., 2020). For example, Trivedi et al. (2014) built a BPNN model to predict the distance covered by blast-induced flyrock in limestone mines. The results showed that the coefficient of determination (R^2) of the BPNN model was 98.3%, whereas it was 81.5% in the case of a statistical multiple regression model. Likewise, Xue et al. (2020) established five ML models, including a BPNN model and an ELM model, for predicting rockburst intensity in deeply buried areas. The study showed that the proposed ELM model had the highest average accuracy of 97.57%, which outperformed the BPNN model (62.13%) and other comparative models, such as the random forest (RF) model (63.40%), the gradient boosting regression (GBR) model (65.13%), and the decision tree (DT) model (58.79%). Close to the research by Trivedi et al. (2014) and Xue et al. (2020), Nguyen et al. (2020) proposed a BRNN model to forecast air-blast overpressure induced by blasting at open-pit coal mines. The research showed that the BRNN model performed well in predicting overpressure, with an R^2 of 93.6%. Therefore, the application of ANNs to construct accurate prediction models has great potential. Despite this potential, the research is scarce in the previous literature on applying ANNs to build prediction models between truck productivity and its influencing parameters (i.e., input variables). These input variables include, but are not limited to, haul distance, truck speed, and weather conditions (e.g., ambient temperature) (Fan et al., 2022, 2023b), which are all associated with truck cycle time and thus affect truck productivity (Sun et al., 2018).

To this end, this study aims to establish prediction models between truck productivity and its input variables based on the real-site dataset using ANNs. The dataset contained more than 290,000 data points, which were collected from open-pit mines in Northern Alberta, Canada. Unlike previous studies that directly built ANN models (Nguyen et al., 2020; Trivedi et al., 2014; Xue et al., 2020), this study first adopted Gaussian mixture modeling (GMM) as a clustering technique to divide the dataset into three latent classes to reduce computational complexity. GMM has proven an efficient clustering method for massive data and can improve model prediction accuracy (Ji et al., 2014). After that, three ANN algorithms, including BPNN, ELM, and BRNN, were used to build regression models in each class. Finally, the weighted ensembles of the ANN models in each class offered the final prediction of truck productivity. Moreover, as comparative ML methods, DT, RF, GBR, and extreme gradient boosting (XGBoost) were also applied to build prediction models.

The innovation of this paper lies in three aspects. First, an in-depth analysis of the unique and massive data from the open pit mines was performed. Second, this study was the first one using ANNs to construct complex nonlinear relationships between truck productivity and its influencing parameters. Third, for the first time, the prediction accuracy of ANN models was enhanced by combining a clustering technique. The contribution of this study is to construct accurate prediction models for truck productivity using ANNs combined with GMM based on real-site massive data from mine sites.

4.2. Methodology and data

4.2.1. Development of proposed model

Figure 4.1 shows the executive process of building the proposed prediction model in this study. It involves data partitioning, modeling methods, and model evaluation, which are described in detail in the following five steps. *Step 1*: The real-site data was randomly and proportionally split into a training dataset (70%) and a testing dataset (30%). The training dataset was prepared for building prediction models based on various ML algorithms. The testing dataset was used to validate the model's prediction performance. *Step 2*: The training dataset was divided into *K* latent classes by an unsupervised clustering model (i.e., GMM). Afterward, the data points in each latent class were used to train three regression models (i.e., BPNN, ELM, BRNN) of truck productivity. The latent classes were the links between the clustering and regression models. The detailed clustering results will be explained in Section 4.3.1. Moreover, a mixture model was obtained from the clustering analysis. *Step 3*: According to the mixture model, the testing dataset was divided into corresponding *K* classes. Also, *K* posterior probabilities (w_1, w_2, …, w_k, …, w_K) were calculated for each data point in the testing dataset. This means that each data point had a posterior probability corresponding to each class (Grün & Leisch, 2007), which will be explained in Section 4.3.1. *Step 4*: The *K* classes generated from the testing dataset in *Step 3* were used to evaluate the performance of the prediction models built in *Step 2*. For instance, Class 1 of the testing dataset was applied to assess the performance of three models (i.e., BPNN, ELM, and BRNN) built in Class 1 of the training dataset. RMSE (root mean square error), MAE (mean absolute error), and R^2 (coefficient of determination) were selected as the performance metrics based on the research by Wu et al. (2020). *Step 5*: The entire testing dataset was fed into all the ANN models in *Step 4*. Each model predicted truck productivity in a parallel manner. The final prediction was a weighted ensemble of the ANN models from all classes, with the weights being the calculated probabilities in *Step 3* (Ni et al., 2020). The evaluation of the final prediction will be discussed in detail in Section 4.3.4. The overall training process was carried out in RStudio software using the R language (version 4.1.3) environment on a personal computer (PC). This PC has a 64-bit operating system with an Intel Core i7-12700K (3.60 GHz) processor and 16.0 GB of random-access memory.

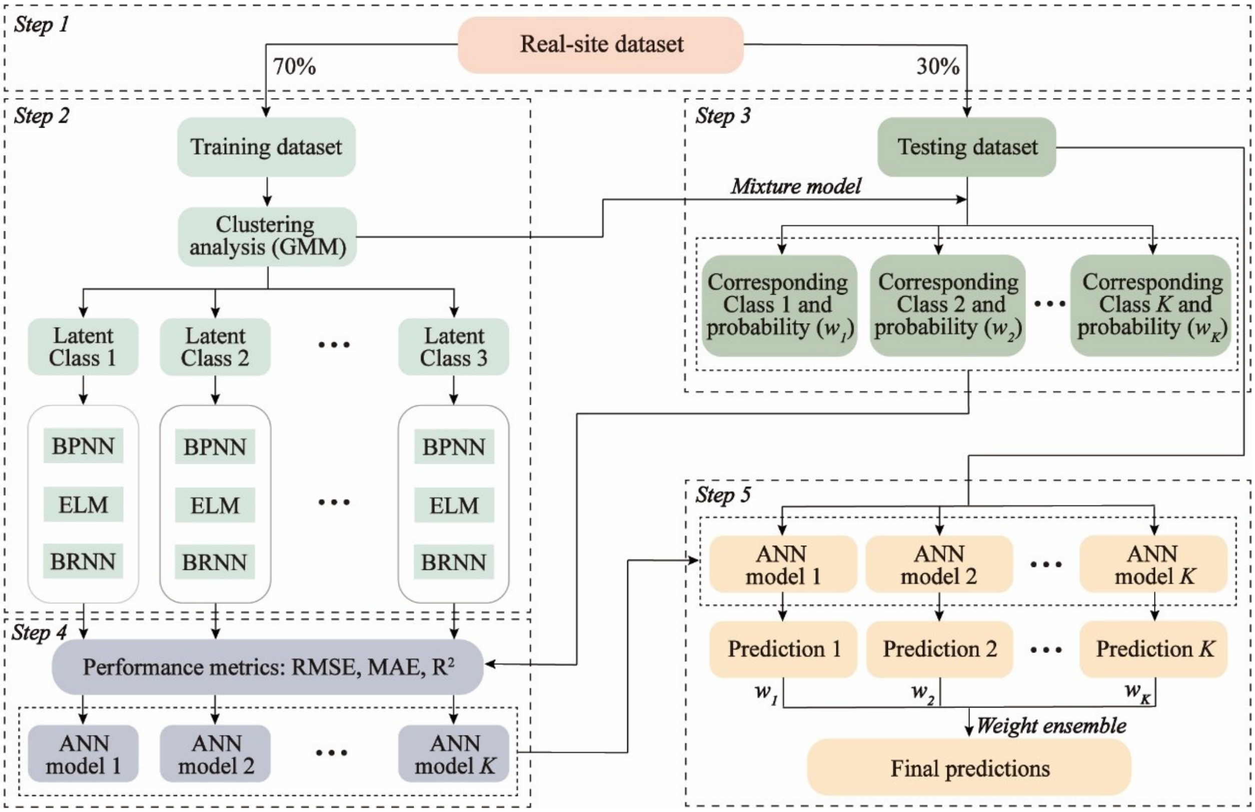


Figure 4.1 The flowchart of implementing a weighted ensemble of artificial neural network (ANN) models (Note: GMM: Gaussian mixture modeling; BPNN: back propagation neural network; ELM: extreme learning machine; BRNN: Bayesian regularized neural network; RMSE: root mean squared error; MAE: mean absolute error, and R^2: coefficient of determination).

4.2.2. Data collection and preparation

The data used in this study came from open-pit mines in Alberta, Canada, which have been compiled into a tabular dataset and stored in the data management system. Before modeling, the tabular dataset was cleaned of blank rows caused by recording errors. Part of the processed dataset is listed in Table 4.1, where each row of data was generated from each truck cycle. As shown in Table 4.1, the output variable is truck productivity (y) in tonne per hour (tph). The input variables consist of haul distance (x_1, km), empty speed (x_2, km/h), ambient temperature (x_3, °C), and waiting at shovel (x_4). These variables were selected because they have been observed by site engineers and were associated with truck cycle time according to our previous research (Fan et al., 2022). Haul distance represents the distance between a loading site and a dumping site. Empty speed refers to the running speed of an unloaded truck returning to the loading site. Empty speed, rather than haul (loaded truck) speed, was considered to increase the independence between inputs. Ambient temperature is a numeric variable that acts as an environmental influencing factor. Waiting at shovel is a binary variable, which contains two labels (0 and 1) indicating that each truck has zero or non-zero waiting time at a shovel. Based on our previous research (Fan et al., 2023b), waiting at shovel as an input impacted truck productivity significantly. The dataset, including these input and output variables, was then randomly and proportionally split into a training dataset (70%) and a testing dataset (30%).

Taking the training dataset as an example, the statistical distributions of the included input and target variables are shown in Figure 4.2. In Figure 4.2(a)-(d), the horizontal axis indicates the numeric variables plotted in histograms with specific ranges; the vertical axis shows the probability density, which refers to the fraction of each range divided by the total amount of data. Among these input variables, haul distance and ambient temperature present multi-peak Gaussian distributions, implying that the dataset collected at mine sites has a mixture of Gaussians (Li et al., 2018). This suggested the rationale for selecting GMM to cluster the dataset in this study, which will be explained in Section 4.2.4. Figure 4.2(a)-(d) also lists the minimum (min), maximum (max), mean, and variance (var) values of these numeric variables. Figure 4.2(e) shows the boxplot of the binary variable (waiting at shovel) and indicates the data size for each label. Before the prediction models were built, the numeric variables in the training and testing datasets were rescaled to be in the range of zero to one using min-max data scaling. This is a commonly used method to normalize the statistical distribution of numerical variables to ensure all variables are relevant equally (Arachchilage et al., 2023).

Nevertheless, there are still some limitations to the input variables considered in this study. Other input variables have not been considered that can also affect truck cycle time and thus affect truck productivity, such as operator habits (Sun et al., 2018), tire properties and rolling resistance (Ma et al., 2022), and truck benching (Soofastaei et al., 2016). These variables may be added to build the prediction models of truck productivity in future studies based on their availability.

Table 4.1 Part of the processed dataset after cleaning (the dataset contained 298,608 data points covering an entire year of truck haulage cycles).

Input variable (x_m)				Target variable (y)
Haul distance (x_1, km)	Empty speed (x_2, km/h)	Ambient temperature (x_3, °C)	Waiting at shovel (x_4, label)	Truck productivity (y, tph)
3.87	37.8	-38	1	###
3.28	37.1	-27	1	###
3.64	28.4	-20	1	###
5.43	28.3	-13	1	###
6.02	35.3	-3.6	0	###
4.25	44.6	0.7	1	###
4.52	31.9	11.8	0	###
2.55	22.3	17.8	0	###
4.2	41.8	28.5	1	###
2.19	37.6	32.8	0	###

("###": the input information is not disclosed as it is the proprietary property of mining companies.)

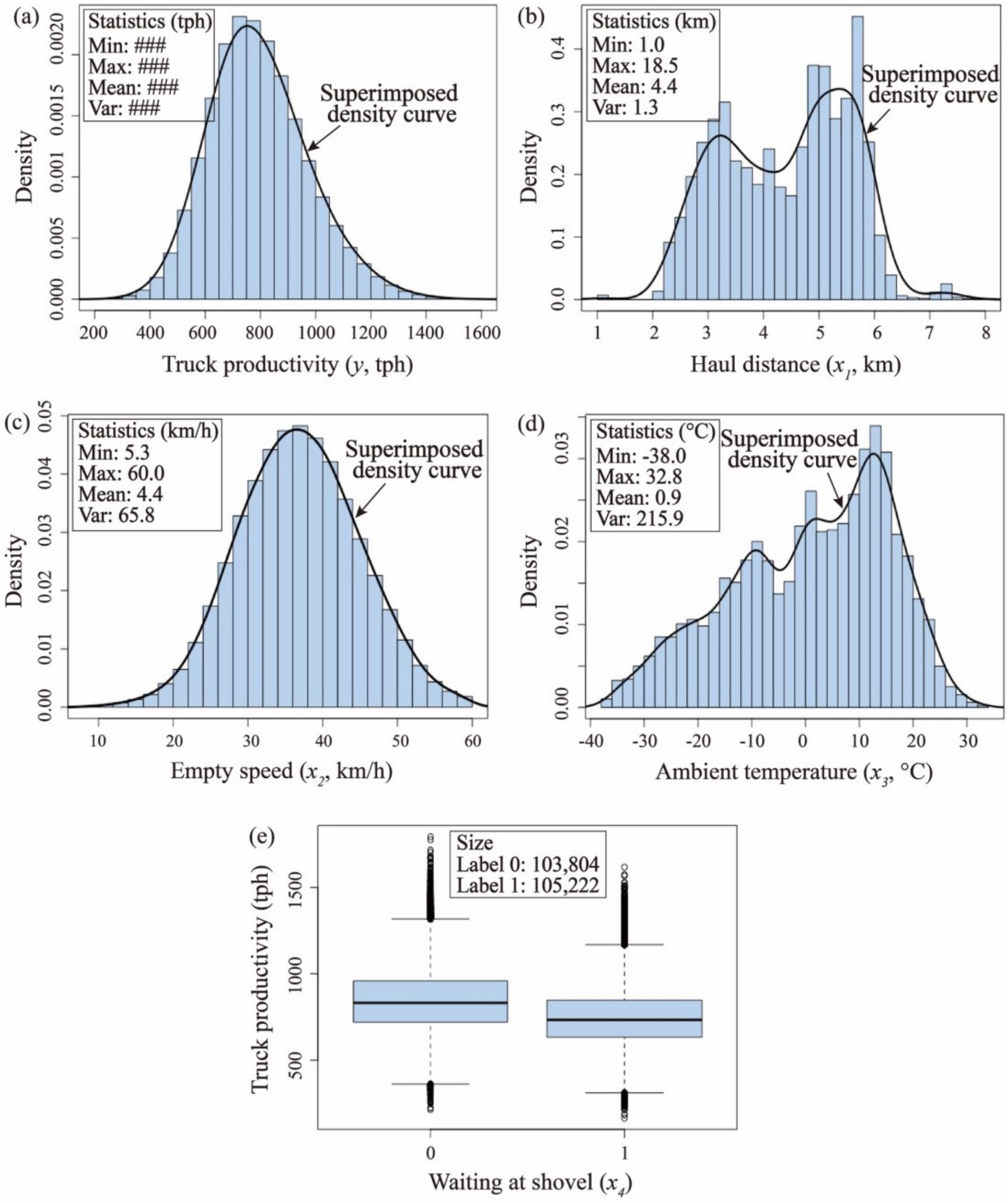


Figure 4.2 Statistical distributions of inputs and the output in the training dataset. (a) The frequency histogram and density curve of the numeric target variable (*y*): truck productivity in tonnes per hour (tph); (b)-(d) The frequency histograms and density curves of three numeric inputs: haul distance (*x_1*) in kilometers (km), empty speed (*x_2*) in kilometers per hour (km/h), and ambient temperature (*x_3*) in degrees Celsius (°C); (e) The boxplot of the binary input: waiting at shovel (*x_4*). (“###”: the input information is not disclosed as it is the proprietary property of mining companies.)

4.2.3. Machine learning methods

4.2.3.1. Back propagation neural network (BPNN)

BPNN is a computational method that deals with regression and classification problems to model complex nonlinear relationships (Çolak, 2022). BPNN emulates the basic structure of biological neural networks, which comprises three layers: input, hidden, and output (Cui & Jing, 2019). As shown in Figure 4.3, the prediction of truck productivity is offered as an example to illustrate the process of building a BPNN model. The input layer consists of input variables that influence truck productivity, including haul distance (x_1), empty speed (x_2), ambient temperature (x_3), and waiting at shovel (x_4). These input variables, referred to as "neurons" in the input layer, are connected to the hidden layer with j neurons. The connection between two neurons is assigned with a weight (w_{mj}), and a linear combination of the weights is transformed using an activation function $f(\cdot)$ to generate the output a_j of the hidden layer (Glória et al., 2016):

$$a_j = f(\sum_{m=1}^{4} w_{mj} x_m + b_j), j = 1,2,\ldots,J \tag{4-1}$$

where m is the mth input variables; j is the jth hidden neurons, and b_j is the bias term. In BPNN, the activation function is defined as a function that maps the input to the desired output (Mouloodi et al., 2022). After that, the output a_j is considered to be new inputs linked to the output layer with new estimated weights (w_{jl}). Likewise, the sum of weights is transformed using an activation function $g(\cdot)$ to generate the final output (y) of the output layer (Glória et al., 2016):

$$y = g(\sum_{j=1}^{m} w_{jl} a_j + b_l), l = 1 \tag{4-2}$$

where l equals 1 since there is only one target variable in this study, and b_l is the bias term. In order to reduce the prediction error of truck productivity, the final output is backpropagated to update the weights and biases during the training process (Wu et al., 2020). In brief, the hidden layer plays a dominant role between the input and output layers, which adds nonlinearity to the system via activation functions to model complex relationships (Kramer, 1991). In this study, the BPNN model was built using the "*nnet*" and "*caret*" packages installed in the R language environment.

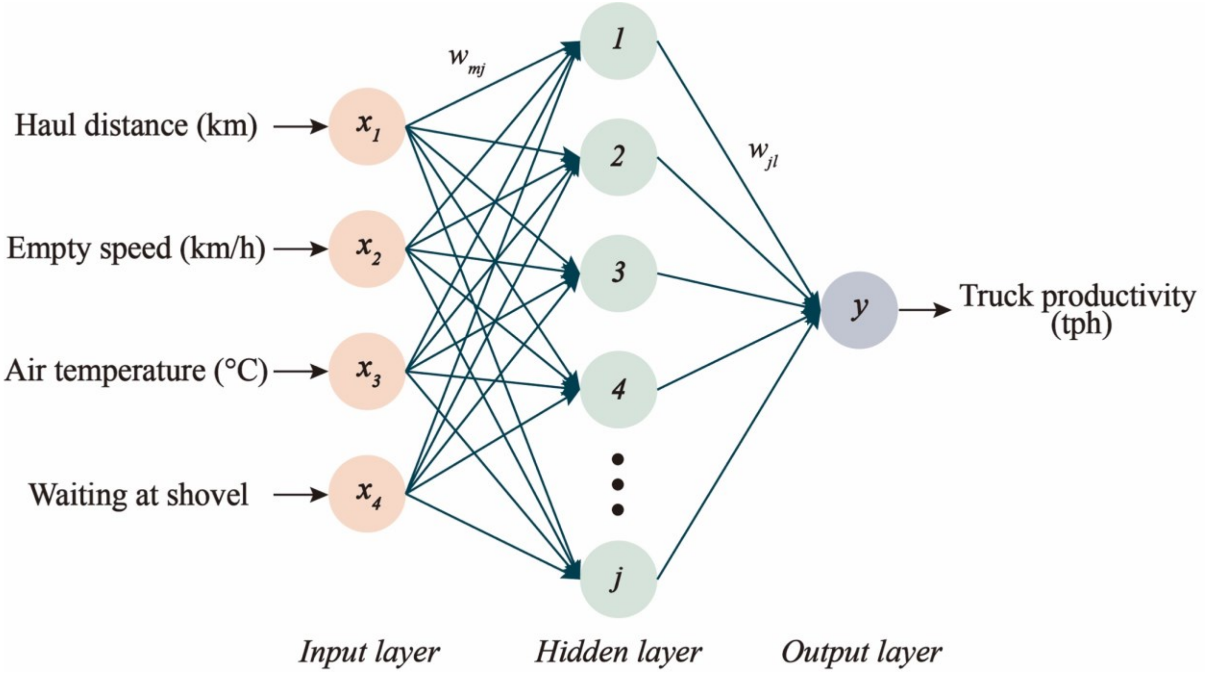


Figure 4.3 The basic structure of a multilayer BPNN.

4.2.3.2. Bayesian regularized neural network (BRNN)

BRNN is also an ANN algorithm with a multilayer structure for classifying and predicting outcomes, which was first proposed by MacKay (1992). The main difference between the standard BPNN and BRNN is the setting of weights (Goodarzi et al., 2010). The former assumes that the weights are fixed values between neurons, which may cause overfitting issues during the training process (e.g., a loss of generalization ability due to a fitting of noise) (Ticknor, 2013). The latter assumes the weights are random variables (Goodarzi et al., 2010). Specifically, BRNN considers a prior probability distribution (usually forming a Gaussian distribution) for these weights (w) and infers their posterior probability distribution, which can be assessed based on the Bayesian theorem (Glória et al., 2016):

$$f(w|D,\mu,\sigma) = \frac{f(W|\mu)f(D|w,\sigma)}{f(D|\mu,\sigma)} \tag{4-3}$$

where D represents observed data points (x_n, y_n), μ and σ are the parameter vectors of the Gaussian distribution function, $f(w|\mu)$ indicates the prior probability distribution for the weights, $f(D|w,\sigma)$ is the likelihood function, and $f(D|\mu,\sigma)$ is the normalization factor, which guarantees that the total probability equals one. The optimal weights can be determined by maximizing the posterior probability $f(w|D,\mu,\sigma)$ under the Bayesian framework (Saini, 2008). With the optimal weights, BRNN minimizes the regularized objective function to reduce prediction error and improve generalization ability (Shi et al., 2019). In this study, the BRNN model was built using the "*brnn*" and "*caret*" packages installed in the R language environment.

4.2.3.3. Extreme learning machine (ELM)

ELM is another ANN algorithm for the single hidden layer feedforward network that was proposed by Huang et al. (2006). ELM has the same structure as BPNN and BRNN, including an input layer, a hidden layer, and an output layer. However, the setting of weights and biases in ELM differs from BPNN and BRNN (Wang et al., 2021). For BPNN, the parameters (i.e., weights and biases) are updated during the training process through back propagation until the fixed parameters are determined to form a neural network. For BRNN, the parameters are considered random variables following Gaussian distributions that are assessed based on the Bayesian theorem. Unlike these parameter estimation methods, ELM assigns random weights to the connections between the input and hidden layers and random biases in the hidden layers. Meanwhile, these weights and biases remain unchanged during the training process. The weights between the hidden and output layers are the only parameters that need to be learned during the training process. Therefore, compared to BPNN and BRNN, ELM converges faster since iterative learning is not required to construct the network (Wang et al., 2021). This results in ELM having high learning speeds when dealing with large amounts of data (Liu et al., 2021). In this study, the ELM model was built using the "*elmNN*" and "*caret*" packages installed in the R language environment.

4.2.3.4. Other machine learning methods

In this study, four widely used ML methods were also applied as comparative approaches to build prediction models of truck productivity, including DT, RF, GBR, and XGBoost. Their basic principles are summarized below.

- DT was proposed by Breiman et al. (1984) and consists of a root node, internal nodes, terminal nodes (or leaf nodes), and branches between nodes. DT uses a set of hierarchical decisions on input features to make predictions. First, the root node (the whole dataset) is split into two internal nodes (subsets) based on an input feature to minimize the mean square error (MSE) in each subset (Krzywinski & Altman, 2017). Following this split rule, all internal nodes then generate new tree branches and nodes until a set of terminal nodes with homogeneous subsets is created. Finally, the terminal nodes indicate the predictions through the path from the root to the terminal.

- RF is an ensemble learning method that integrates numerous DTs to gain an ensemble model with better performance (Xue et al., 2020). RF combines a series of DTs through a bagging technique to improve the prediction performance (Breiman, 2001). "Bagging" is a portmanteau of bootstrapping and aggregating. "Bootstrapping" is a sampling technique that obtains a subset for training each DT by randomly sampling the whole dataset with replacement. Then, a portion of input features is randomly sampled from overall input features for best splitting at the root and internal nodes in each DT. The final prediction of RF is to average the decisions of all DTs, which is called “aggregating”. Also, in bagging, RF retains an additional subset (also known as an “out-of-bag” subset) from the training dataset to evaluate whether each DT is overfitted. In short, RF offers a robust prediction through the bagging technique, which is superior to a single DT (Milad et al., 2022).

- GBR is also an ensemble learning method that combines a series of DTs for dealing with regression and classification problems (Friedman, 2001). Unlike RF, GBR uses a boosting technique instead of the bagging technique to generate DTs (Friedman, 2001). Boosting is a successive process in which each DT learns, improves, and corrects the prediction errors made by the previous DT. This differs from bagging since each DT is built into RF in a parallel and independent manner (Ribeiro & dos Santos Coelho, 2020). In boosting, each well-trained DT focuses on data that have been inaccurately predicted by the previous DT. The residual errors of these data are specified with a loss function F. GBR adopts an iterative process to minimize the expectation of the loss function F by adding new DTs in sequence, which is also known as functional gradient descent. Finally, these successively trained DTs are combined into an ensemble learner to predict the outcome.

- XGBoost is another tree-based ensemble learning algorithm based on the boosting technique, which was proposed by Chen and Guestrin (2016). It is an extension of GBM, aiming to avoid overfitting problems as well as improve computational ability (Ribeiro & dos Santos Coelho, 2020). Unlike GBM, in XGBoost, a regularization penalty term Ω with weights is added to the objective function in addition to the loss function F. This can help to control the model complexity, thus preventing overfitting issues (Su et al., 2022). Moreover, XGBoost implements many features, such as parallel and distributed computing (Chen & Guestrin, 2016), to provide a fast algorithm. In brief, XGBoost effectively integrates numerous weak learners (DTs) into one strong learner (ensemble model) and improves the generalization ability (Mohammed & Ismail, 2022).

In this study, the DT, RF, GBR, and XGBoost models were built using packages "*rpart*", "*ranger*", "*gbm*", "*xgboost*" combined with the package "*caret*" installed in the R language environment. In addition to building prediction models of truck productivity, these four tree-based models were used to determine the relative importance of each input variable. Further details about the principles of determining the relative importance can be found in Vitale et al. (2014), Delen et al. (2013), and Onyekwena et al. (2022).

4.2.4. Gaussian mixture modeling (GMM)

GMM is a probability distribution-based clustering technique that recognizes latent classes from a data population (Bishop, 2006). In GMM, each class is assumed to follow a specific Gaussian distribution. The weighted sum of these Gaussians forms a mixture of Gaussians to represent the overall probability distribution of the data population. This probability distribution can be written as a mixture model (Leisch, 2004):

$$P(y|x,\emptyset) = \sum_{k=1}^{K} \pi_k f_k(y|x,\mu_k,\Sigma_k) \tag{4-4}$$

where K is the number of classes. $\emptyset$ indicates the parameter set $\{\pi_k,\mu_k,\Sigma_k\}$ of the mixture model. Of these, π_k is the weight of the kth class, which is non-negative together with $\sum_{k=1}^{K} \pi_k = 1$. μ_k and Σ_k are the mean vector and covariance matrix, respectively. $P(y|x,\emptyset)$ is the mixture model. $f_k(y|x,\mu_k,\Sigma_k)$ is the probability distribution of the kth class. To determine the mixture model, GMM first applies a two-step algorithm, expectation and maximization (EM), to estimate the parameter set $\{\pi_k, \mu_k, \Sigma_k\}$. In E-step, the posterior probability of each data point (x_n, y_n) assigned to each class is calculated (Leisch, 2004):

$$\gamma_{nk} = \frac{\pi_k f_k(y_n|x_n,\mu_k,\Sigma_k)}{\sum_{k=1}^{K} \pi_k f_k(y_n|x_n,\mu_k,\Sigma_k)} \quad (4\text{-}5)$$

The data point belongs to a specific class *k* if it has the maximum posterior probability in this *k*th class (Grün & Leisch, 2007). With the γ_{nk}, the parameter set $\{\pi_k, \mu_k, \Sigma_k\}$ can be estimated in the M-step by maximizing the log-likelihood function (*log L*) (Leisch, 2004):

$$log\, L = \sum_{n=1}^{N} \log(P(y|x,\phi)) = \sum_{n=1}^{N} \log(\sum_{k=1}^{K} \pi_k f_k(y|x,\mu_k,\Sigma_k)) \quad (4\text{-}6)$$

$$\pi_k = \frac{1}{N} f_k = \frac{1}{N} \sum_{n=1}^{N} \gamma_{nk} \quad (4\text{-}7)$$

where *N* is the number of data points. In GMM, the EM algorithm is an iterative process that does not terminate until the maximum *log L* is reached. After that, the optimal number of latent classes is obtained by minimizing the Bayesian information criterion (BIC) value (Mehrjou et al., 2016):

$$BIC = -2logL + ClogN \quad (4\text{-}8)$$

where *C* is the number of estimated parameters, indicating the complexity of the mixture model. Finally, the mixture model is determined in GMM, showing the overall probability distribution of the mixture of Gaussians.

GMM was adopted in this study mainly because the real-site dataset collected at mine sites presents the multi-peak Gaussian distributions, also known as a mixture of Gaussians (Li et al., 2018). For example, as shown in Figure 4.2 in Section 4.2.2., both haul distance and ambient temperature present a mixture of Gaussians. In addition, GMM has been proven helpful in reducing computational costs and improving model predictability when dealing with large amounts of data, such as streamflow (Ni et al., 2020), heat load (Lu et al., 2019), and seismic signals (Kuyuk et al., 2012). Therefore, GMM was used to identify latent classes in this study.

4.2.5. Performance metrics

Three performance metrics were used in this study to quantitatively assess the accuracy of prediction models: R^2, MAE, and RMSE, which are listed as follows (Huo et al., 2021):

$$R^2 = 1 - \frac{\sum_n^N (y_n - \hat{y}_n)^2}{\sum_n^N (y_n - \bar{y}_n)^2} \tag{4-9}$$

$$MAE = \frac{1}{N} \sum_{n=1}^{N} |y_n - \hat{y}_n| \tag{4-10}$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{n=1}^{N} (y_n - \hat{y}_n)^2} \tag{4-11}$$

where y_n is the observed truck productivity; $\hat{y}_n$ is the predicted truck productivity, and $\bar{y}_n$ is the average value of observed truck productivity. R^2 is scaled between 0 and 1, which measures the goodness of fit of a regression model and reflects the degree to which data points fit a curve. MAE indicates the absolute error mean between the observed and predicted values, and RMSE represents the standard deviation of the residuals between the observed and predicted values (Huo et al., 2021). In brief, a prediction model with a higher R^2 and a lower RMSE and MAE has better prediction accuracy.

4.2.6. Hyperparameters tuning

For ML algorithms, built-in hyperparameters are required to be tuned during the modeling process to reduce the overfitting risks, thus improving the model performance (Wu et al., 2020). In this study, for BPNN, the hyperparameters were *size* (the number of neurons in the hidden layer) and *decay* (the regularization parameter to avoid overfitting). For ELM, the hyperparameters were *nhid* (the number of neurons in the hidden layer) and *actfun* (activation function). Finally, for BRNN, the hyperparameter was *neuron* (the number of neurons in the hidden layer).

Table 4.2 lists these hyperparameters and their search ranges. In this study, a method of five-fold cross-validation combined with grid search was used to tune the hyperparameters since this method is easy to perform and has good optimization results (Wu et al., 2020). The grid search first defined a grid of hyperparameters in each algorithm according to the search ranges. After that, each class of the training dataset was randomly partitioned into five folds. Four folds were used to train prediction models with the hyperparameters traversing each position in the grid. The remaining one fold was used to test the performance of the trained models by calculating the RMSE value (Sun et al., 2021). This process was repeated five times with different folds as the test fold. The optimal hyperparameters were obtained from the trained model with the lowest RMSE value.

Table 4.2 Hyperparameters of the ML algorithms and their search ranges in this study.

Algorithm	Hyperparameter	Range and step	Reference
BPNN	*size*	[10-50], 5	Ripley and Venables (2022)
	decay	[0.01-0.1], 0.01	
ELM	*nhid*	[1-20], 2	Mouselimis et al. (2022)
	actfun	[sin, purelin, transig, radbas]	
BRNN	*neuron*	[10-20], 1	Rodriguez and Gianola (2021)

4.3. Results and discussion

4.3.1. Clustering analysis using Gaussian mixture modeling

Figure 4.4 shows the latent classes identified from the training and testing datasets and the relationships between truck productivity and latent classes. As shown in Figure 4.4(a), the training dataset with 209,026 data points was clustered into three latent classes: Class 1 (57,848), Class 2 (93,604), and Class 3 (57,574). Q_1 and Q_3 are the 25th and 75th percentiles in each class, representing the distribution interval of data points. In Class 1, the value of truck productivity ranged mainly between 570 tph (Q_1) and 715 tph (Q_3), with median and mean values of 644 tph and 640.7 tph. In Classes 2 and 3, the Q_1, Q_3, median, and mean values of truck productivity were increased successively. This implies that truck productivity varied significantly among these three classes, in the order of Class 1 < Class 2 < Class 3, which can be known as low, medium, and high truck productivity at mine sites. Likewise, in Figure 4.4(b), based on the mixture model, the testing dataset with 89,582 data points was correspondingly partitioned into three latent classes: Class 1 (24,668), Class 2 (40,286), and Class 3 (24,628). The Q_1, Q_3, median, and mean values of truck productivity were listed in ascending order within these three classes, indicating low, medium, and high values of truck productivity. In addition, according to the GMM analysis, each data point in the testing dataset was calculated with a probability corresponding to each class (Grün & Leisch, 2007). As a result, each data point had three probabilities; some of them are listed in Table 4.3 as examples. This study was similar to the research by Ni et al. (2020), in which they identified two classes (low and high flow) from large streamflow datasets using GMM. Each class was then fitted with an XGBoost model, and the final prediction was the weighted ensemble of these XGBoost models in all classes. In this study, the weights were the probabilities calculated in the GMM analysis. The study showed that the prediction accuracy was improved by approximately 11% after using the weighted ensemble. Therefore, it was promising to apply GMM to identify latent classes in this study.

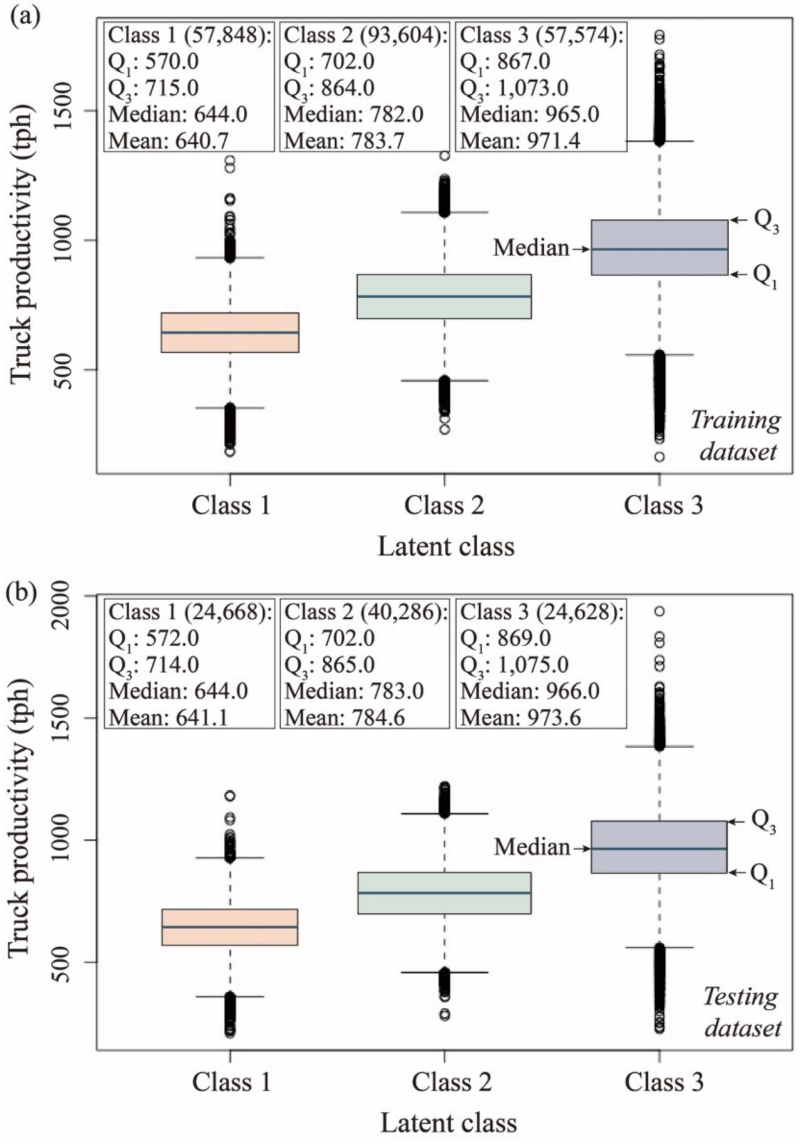


Figure 4.4 Identifying latent classes from the training and testing datasets. (a) Three latent classes were identified by GMM clustering from the training dataset; (b) Three latent classes were correspondingly identified from the testing dataset based on the mixture model obtained in GMM (Note: Q_1 and Q_3: the 25th and 75th percentiles of each class).

Table 4.3 Probabilities calculated for part of the data points in the testing dataset corresponding to three classes (the total number of data points in the testing dataset was 89,582).

Number	Class 1	Class 2	Class 3
1	0.3488	0.3808	0.2704
2	0.3097	0.4997	0.1906
3	0.3048	0.3761	0.3191
4	0.0233	0.3580	0.6187
5	0.1715	0.5882	0.2403
6	0.0781	0.2759	0.6460
7	0.3958	0.4061	0.1981
8	0.9639	0.0316	0.0045
9	0.0823	0.7625	0.1552
10	0.0721	0.6106	0.3173

4.3.2. Determination of hyperparameters

In this study, the hyperparameters built into the ML models were tuned using a grid search method based on five-fold cross-validation. Three ANN models (BPNN, ELM, and BRNN) were built based on three latent classes of the training dataset. As a result, a total of nine prediction models were required to be tuned for optimal hyperparameters. Table 4.4 shows the determined hyperparameters for these prediction models in each class. With the optimal hyperparameters, these models were considered the optimal models in the hyperparametric search range for predicting truck productivity. This is similar to the research by Moayedi et al. (2019), in which the hyperparameters were tuned to avoid overfitting issues and improve the model accuracy, thus providing more reliable landslide susceptibility mapping using ANN. Based on the values of R^2

(98.99% and 97.33%) and RMSE (0.039 and 0.111), the prediction performance of the tuned ANN model was higher than that of the untuned ANN model.

Table 4.4 Optimal hyperparameters for the ML models built in each class of the training dataset.

Algorithm	Hyperparameter	Class 1	Class 2	Class 3
BPNN	*size*	15	20	40
	decay	0.05	0.5	0.01
ELM	*nhid*	8	9	9
	actfun	purelin	purelin	purelin
BRNN	*neuron*	17	18	18

4.3.3. Performance comparison of ANN models built in each class

Figure 4.5 shows the scatterplots of the observed (horizontal axis) and predicted (vertical axis) truck productivity obtained from the BPNN, ELM, and BRNN models based on Classes 1, 2, and 3 of the testing dataset. Figure 4.6 shows the scatterplots of the observed and predicted truck productivity obtained from the BPNN, ELM, and BRNN models based on Classes 1, 2, and 3 of the training dataset. The more minor the deviation between the observed and predicted values, the closer are the scatter points along the diagonal ($y = x$) line (Piñeiro et al., 2008). Taking the results in Figure 4.5 as examples, in Classes 1 and 3, the scatter points generated by these three models were not evenly distributed on both diagonal sides, indicating that the performance of these three models was not high in predicting truck productivity in Classes 1 and 3. In other words, the four observed input variables currently available were insufficient for the accurate prediction of truck productivity in Class 1 (low values) and Class 3 (high values). This suggests that there are many other unobserved influencing factors at mine sites that may affect truck productivity, such as equipment overhaul, road maintenance, route changes, and personnel shifts (Alarie & Gamache, 2002). However, in Class 2, the scatter points were well distributed along the diagonal line, indicating relatively small deviations between the observed and predicted truck productivity. Also, RMSE, MAE, and R^2 quantified the performance of these models in Class 2. These performance metrics were 47.19, 39.76, and 84.39% for the BPNN model, 46.25, 39.21, and 85.01% for the ELM model, and 45.94, 38.98, and 85.20% for the BRNN model. Therefore, these three models had higher accuracy in predicting the truck productivity of Class 2 (medium values). Although the BPNN, ELM, and BRNN models performed differently in these three classes, the BRNN model outperformed the other two models. This agreed with the study by Potočnik et al. (2019), who built three models (BPNN, ELM, and BRNN) for the short-term prediction of building temperatures. The results showed that the RMSE of the BRNN model (0.065) was less than that of the BPNN (0.069) and ELM (0.073) models.

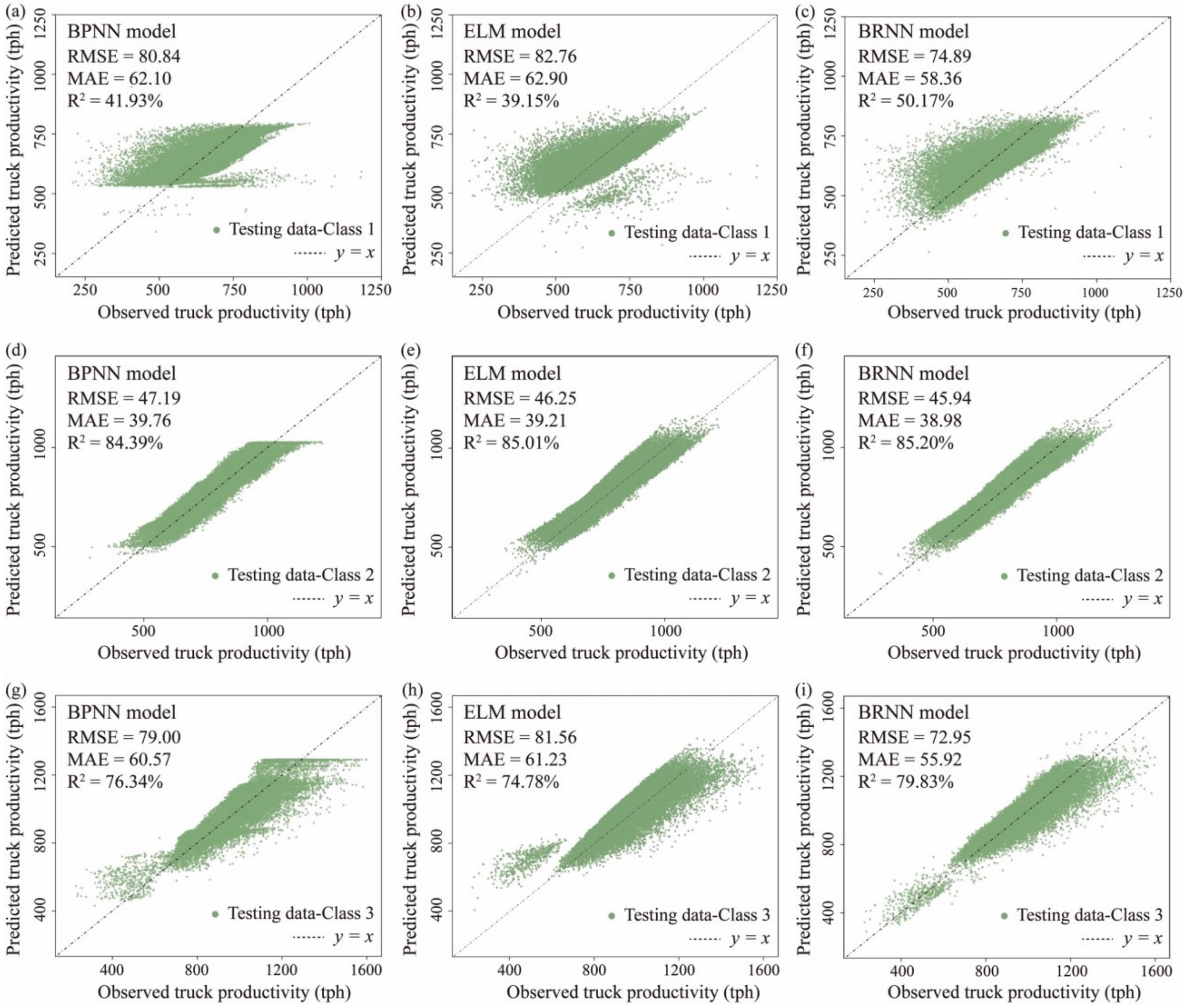


Figure 4.5 Scatterplots of the observed truck productivity and predicted truck productivity. (a) BPNN, (b) ELM, and (c) BRNN models evaluated by Class 1 of the testing dataset; (d) BPNN, (e) ELM, and (f) BRNN models evaluated by Class 2 of the testing dataset; (g) BPNN, (h) ELM, and (i) BRNN models evaluated by Class 3 of the testing dataset.

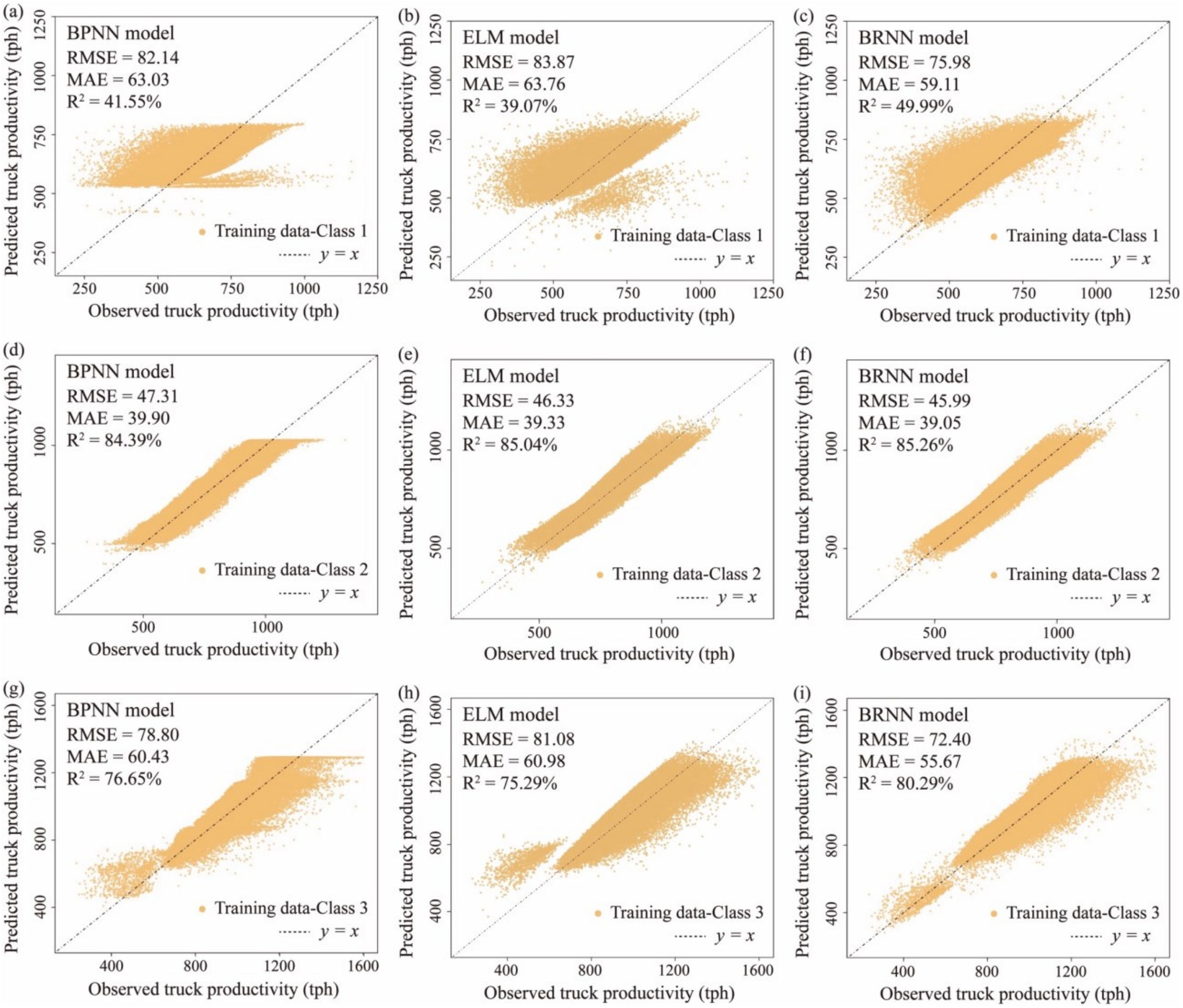


Figure 4.6 Scatterplots of the observed truck productivity and predicted truck productivity. (a) BPNN, (b) ELM, and (c) BRNN models evaluated by Class 1 of the training dataset; (d) BPNN, (e) ELM, and (f) BRNN models evaluated by Class 2 of the training dataset; (g) BPNN, (h) ELM, and (i) BRNN models evaluated by Class 3 of the training dataset.

4.3.4. Performance evaluation of weighted ensembles of ANN models

In Section 4.3.3., three ANN models (i.e., BPNN, ELM, and BRNN) were established in each class of the training dataset. The testing dataset with 89,582 data points was then fed into these three ANN models to predict truck productivity. After that, the final predictions were provided by the weighted ensembles of these ANN models (which were also referred to as the WE-BPNN, WE-ELM, and WE-BRNN models in this study) from the three classes.

To evaluate the performance of the WE-BPNN, WE-ELM, and WE-BRNN models, four commonly used ML models were built based on the training dataset as benchmark models. These were the DT, RF, GBR, and XGBoost models. Table 4.5 shows the performance comparison of these seven prediction models along with the running time. From Table 4.5, the WE-ELM (109.2 s), DT (77.4 s), and XGBoost (307.2 s) models required shorter running times. This is caused by random weights assignment in ELM (Wang et al., 2021), a single tree construction in DT (Krzywinski & Altman, 2017), and parallel or distributed algorithms application in XGBoost (Chen & Guestrin, 2016). Compared with these models, the running time was much longer for the WE-BRNN (14,004.0 s), WE-BPNN (9072.0 s), RF (2726.4 s), and GBR (1998.6 s) models because of complex regularization operations (Shi et al., 2019), multiple hyperparameters tuning, and more tree constructions (Ribeiro & dos Santos Coelho, 2020) occurring during the modeling process. Regarding prediction accuracy, in terms of the testing dataset, the WE-BRNN model had the lowest RMSE and MAE and the highest R^2 in the three weighted ensemble models, with values of 66.23, 46.61, and 86.34%. Accordingly, the three performance metrics of the WE-BPNN and WE-ELM models were 69.42, 48.21, and 84.99%, and 69.43, 47.51, and 84.98%. Therefore, the WE-BPNN and WE-ELM models were close in performance; however, the WE-BRNN model still performed better than these two models in predicting truck productivity. Furthermore, these three weighted ensemble models were compared with the other four ML models. Table 4.5 shows that although the XGBoost model was the best model of the benchmark models, the R^2 of the WE-BRNN (86.34%), WE-BPNN (84.99%), and WE-ELM (84.98%) models was more than two times higher than that of the XGBoost model (42.23%). Hence, based on the GMM analysis, the performance of the proposed WE-BRNN, WE-BPNN, and WE-ELM models was considerably better than these benchmark models. This is because GMM is an unsupervised clustering technique that enables data with more similarities to be clustered into the same group (or referred to as latent class in this study) (Li et al., 2018). Since the prediction models were built on the data with more similarities in each class, these models more accurately described the relationships between the inputs and output in the corresponding class (Liu et al., 2020). Therefore, the weighted ensemble (WE) of these models performed better than the model built based on the original training data. In other words, a weighted ensemble approach based on the GMM analysis significantly improved the model's accuracy. A similar finding was reported by Akram et al. (2018), who built a weighted ensemble prediction model for indoor localization based on the GMM analysis. The result showed that the model accuracy was enhanced from 79% to 89% compared to the baseline model. To conclude, the proposed weighted ensemble models with better performance can provide mining companies with a new approach to predicting truck productivity.

Table 4.5 Comparison of weighted ensembles of ANN models with other machine learning models.

Model	Dataset	RMSE	MAE	R^2 (%)	Running time (s)
WE-BRNN model	Training	66.53	46.79	86.24	14,004.0
	Testing	66.23	46.61	86.34	
WE-BPNN model	Training	69.63	48.40	84.92	9,072.0
	Testing	69.42	48.21	84.99	
WE-ELM model	Training	69.69	47.73	84.88	109.2
	Testing	69.43	47.51	84.98	
DT model	Training	138.19	107.76	40.62	77.4
	Testing	139.03	108.39	39.79	
RF model	Training	135.70	105.81	42.73	2726.4
	Testing	137.37	107.04	41.22	
GBR model	Training	136.43	106.31	42.12	1998.6
	Testing	136.94	106.68	41.59	
XGBoost model	Training	134.68	104.96	43.60	307.2
	Testing	136.18	106.08	42.23	

4.3.5. Relative importance analysis

In this study, the relative importance of the four input variables was provided by four tree-based models: DT, RF, GBR, and XGBoost. The results obtained from each model are shown in a radar chart in Figure 4.7(a). In Figure 4.7(a), the vertices of the irregular polygons indicate the four input variables. The distance of the vertices from the center along the axis is the relative importance of each input variable. According to Li et al. (2022), each of the tree-based models provided a set of different relative importance due to the different principles of these models. Although the input variables with less relative importance were slightly different in these four models, the critical input variable was consistent. As shown in Figure 4.7(a), haul distance had the highest relative importance in the four models, indicating that it was the most pivotal input for predicting truck productivity. This is in line with site observations (Cervantes et al., 2019); haul distance is often cited by mining companies as the main factor affecting truck productivity since it can directly affect the cycle time. Figure 4.7(b) shows the average importance score (in percentage) of the four input variables in the four models. The relative importance ranking of input variables was haul distance (46.07%) > empty speed (19.25%) > ambient temperature (18.23%) > waiting at shovel (16.45%). Besides haul distance, the other three input variables also played important roles in predicting truck productivity. Empty speed had the second-highest relative importance, with a value of 19.25%. This determines the traveling time from dumping sites to loading sites, thus influencing truck productivity (Schexnayder et al., 1999). After empty speed, the relative importance of ambient temperature (18.23%) was slightly less than that of empty speed (19.25%), indicating ambient temperature also played an essential role in predicting truck productivity. Ambient temperature can influence operator habits (Sun et al., 2018), tire performance (Ma et al., 2022), and even road conditions (Svenson & Fjeld, 2017) at mine sites, thus affecting cycle time. According to Sun et al. (2018), the prediction accuracy of truck cycle time was enhanced by about 5% when ambient temperature and other meteorological factors were considered. Lastly, waiting at shovel had a relative importance of 16.45%, which also affected truck productivity to some extent. In results similar to the study by Ercelebi and Bascetin (2009), the wait time at shovel was extended from 2.48 min to 3.11 min when the truck fleet size increased from three trucks to five. This resulted in an increase in cycle time and a decrease in truck productivity. In short, according to the relative importance analysis in the four models, all four input variables played important roles in forecasting truck productivity. Among them, haul distance was the most critical input for predicting truck productivity. This can also be seen by the impact of parameter size on the model performance (as shown in Table 4.6). Taking the DT model as example, the model performance was continuously improved with each additional input variable. When haul distance was selected to be added to the model, the accuracy of the DT model was significantly enhanced (from 20.15% to 39.41%). This was consistent with our previous study (Fan et al., 2022). Through this relative importance analysis, mining engineers can gain an in-depth understanding of the major real-world influences on truck productivity. Based on these results, they can construct more accurate prediction models by considering multiple influencing factors related to truck productivity, thus providing accurate parameter estimates for mine planning and budgeting decisions.

Table 4.6 Parameter selection and the corresponding model performance.

Model	Parameter selection	RMSE	MAE	R^2 (%)
DT-1	waiting at shovel	172.11	136.15	7.73
DT-2	waiting at shovel + ambient temperature	162.70	128.10	17.55
DT-3	waiting at shovel + ambient temperature + empty speed	160.11	125.80	20.15
DT-4	waiting at shovel + ambient temperature + empty speed + haul distance	139.47	108.81	39.41

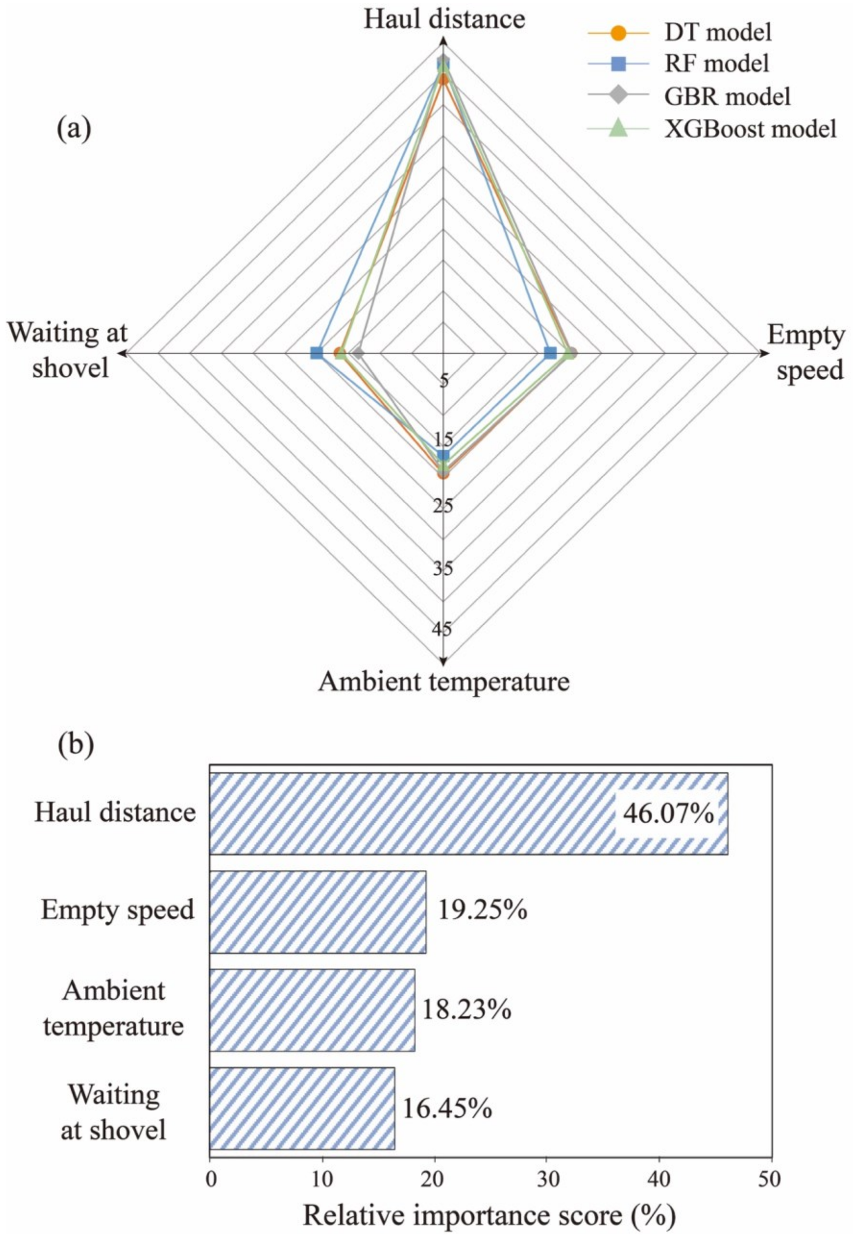


Figure 4.7 The relative importance of input variables. (a) The relative importance of input variables obtained from four tree-based models; (b) The average relative importance score of input variables.

4.4. Conclusions

The truck haulage data from open-pit mine sites are usually massive and multidimensional with multi-peak Gaussian distributions. Artificial neural networks (ANNs) are well-known machine learning algorithms to handle massive and multidimensional data for building models. Moreover, Gaussian mixture modeling (GMM) is a suitable option for processing the data under multi-peak Gaussian distributions and enhancing model prediction accuracy. Hence, for the first time, this study combined the knowledge of statistics and mining engineering to deal with the complex truck haulage data and improve the prediction of truck productivity at mine sites. A back propagation neural network (BPNN), an extreme learning machine (ELM), and a Bayesian regularized neural network (BRNN) coupled with GMM were adopted to build three weighted ensembles models, WE-BPNN, WE-ELM, and WE-BRNN, for predicting truck productivity. The main conclusions are summarized as follows:

(1) The BRNN model outperformed the BPNN and ELM models in predicting low, medium, and high values of truck productivity. For example, the RMSE, MAE, and R^2 were 45.94, 38.98, and 85.20% for the BRNN model, while these metrics were 47.19, 39.76, and 84.39% for the BPNN model, and 46.25, 39.21, and 85.01% for the ELM model.

(2) The WE-BRNN had a higher accuracy than the WE-BPNN and WE-ELM models. For instance, in terms of the testing dataset, the WE-BRNN model had the lowest RMSE and MAE and the highest R^2 in the three weighted ensemble models, with values of 66.23, 46.61, and 86.34%. Accordingly, the three performance metrics of the WE-BPNN and WE-ELM models were 69.42, 48.21, and 84.99%, and 69.43, 47.51, and 84.98%, respectively.

(3) The proposed weighted ensemble models performed better than the benchmark models in

predicting truck productivity, indicating that a weighted ensemble approach based on the GMM analysis significantly improved the model's accuracy. For example, although the extreme gradient boosting (XGBoost) model was the best model of the benchmark models, the R^2 of the WE-BRNN (86.34%), WE-BPNN (84.99%), and WE-ELM (84.98%) models was more than two times higher than that of the XGBoost model (42.23%). This analysis provides mining companies with a new approach to predicting truck productivity.

(4) Based on the relative importance analysis, haul distance was the most crucial input variable for predicting truck productivity. The relative importance ranking of input variables was haul distance (46.07%) > empty speed (19.25%) > ambient temperature (18.23%) > waiting at shovel (16.45%). This finding helps mining engineers gain an in-depth understanding of the major real-world influences on truck productivity.

Chapter 5. Improved extreme machine learning for rapid estimation of mining truck cycle time based on feature selection and unsupervised clustering techniques

This chapter has been summitted for peer review as **C. Fan**, N. Zhang, B. Jiang, W.V. Liu, Improved extreme machine learning for rapid estimation of mining truck cycle time based on feature selection and unsupervised clustering techniques, *Expert Systems with Applications*. © Elsevier. (2023). (Under review)

Nomenclatures

ANN	Artificial neural network
ANOVA	Analysis of variance
BIC	Bayesian information criteria
BPNN	Back propagation neural network
BRNN	Bayesian regularized neural network
C	The number of estimated parameters
DT	Decision tree
ELM	Extreme learning machine
EM	Expectation-maximization
ERT	Extremely randomized tree
f_k	Probability density function
GBR	Gradient boosting regression
GMM	Gaussian mixture modeling
k	The *k*th latent classes
K	The number of latent classes
KM	K-means

L	Likelihood of a set of data points
m	The *m*th input variable
M	The number of input variables
MAE	Mean absolute error
n	The *n*th data point
N	The number of data points
NN	Neural networks
p	Number of data points in each class clustered by K-means
P	Mixture model
R^2	Coefficient of determination
RF	Random forest
RFE	Recursive feature elimination
RMSE	Root mean square error
tph	Tonnes per hour
XGBoost	Extreme gradient boosting
x_m	The *m*th input variable
y	Output variable

$\bar{y}$	Mean value of *y*
$\hat{y}$	Predicted value of *y*
γ_{nk}	Posterior probability
θ	Parameter vector of the density function
λ_k	A set of data points that maximize γ_{nk}
μ_k	Mean vector of the density function
π_k	Weight of the *k*th latent class
Σ_k	Covariance matrix
$\emptyset$	Parameter set of the mixture model

5.1. Introduction

Oil sands mining contributes immensely to the national economic output of Canada (Arciszewski et al., 2022). The GDP of oil sands mining has been projected at CAD$ 2,106 billion for Canada over 25 years (2010 - 2035) (Honarvar et al., 2011a). In oil sands mining, off-the-road trucks play a leading role in transporting bulk materials (ores and waste) at open-pit mine sites (Ma et al., 2023). The time required for a truck to complete a haulage cycle is referred to as truck cycle time, consisting of time for loading, hauling, dumping, returning, and waiting (Song et al., 2017). Mining truck cycle time is of great interest to the resource industry since it is a critical indicator in assessing the maximum productivity achievable between load and dump sites, which directly affects a mine's overall productivity, production targets, planning, and budgets (Cervantes et al., 2019; Chanda & Gardiner, 2010).

To estimate truck cycle time, machine learning has attracted considerable attention because of its ability to build data-driven prediction models (Fan et al., 2023b, 2023d). Machine learning is a general term for a set of mathematical algorithms that can automatically acquire information from historical data and establish input-output relationships (i.e., prediction models) (Arachchilage et al., 2023). These algorithms have been successfully employed for various mining applications, such as rockburst prediction (Zhou et al., 2016), cement materials (Sahari Moghaddam et al., 2020), mining safety (Zhou et al., 2019), and waste management (Daware et al., 2022; Fan et al., 2019). Among these algorithms, extreme learning machine (ELM), extremely randomized trees (Extra-trees or ERT), and extreme gradient boosting (XGBoost) are well-known extreme machine learning methods for handling massive amounts of data, which are superior in prediction performance and time efficiency (Fan et al., 2023d; Wang et al., 2022; Zhang et al., 2023). For example, Dhini et al. (2022) proposed an ELM model for fault diagnosis of steam turbines in thermal power plants. The research presented that ELM achieved sound prediction accuracy, with a coefficient of determination (R^2) of 96.58%. Meanwhile, the computational time of this model was 54.81 s, which performed significantly faster than the widely used backpropagation neural network (BPNN) model (161.12 s). Likewise, both Chencho et al. (2022) and Saeed et al. (2021) built ERT and random forest (RF) models for two-element damage quantification of concrete structures and fault detection in wireless sensor networks, respectively. According to Chencho et al. (2022), the accuracy of the ERT model (98.5%) was better than that of the RF model (97.5%); the computational time of ERT (22.70 s) was considerably shorter than the latter (101.59 s). Saeed et al. (2021) also observed that ERT (81.20%) was superior to RF (79.60%) and reduced the running time by more than half, from 190 s to 90 s. Moreover, Demir and Sahin (2023) used XGBoost and two traditional boosting methods, including gradient boosting regression (GBR) and adaptive boosting (AdaBoost), to forecast soil liquefaction in earthquake engineering. Compared with GBR and AdaBoost, XGBoost demonstrated the highest prediction accuracy (96.75%) and the smallest computational cost (5.15 s). Therefore, these extreme machine learning algorithms have substantial potential to build accurate prediction models and provide fast computation for mining truck cycle time.

Nevertheless, no research has been found that uses extreme machine learning algorithms to estimate mining truck cycle time based on the current literature review. These extreme algorithms may bring vast benefits to predicting truck cycle time, which entails large amounts of real-site datasets and complex relationships between numerous inputs and outputs (Fan et al., 2022, 2023b). For instance, the dataset in this study contained over 8,600 data points encompassing a full year's truck haulage information. Moreover, many input variables are available at mine sites according to the site observation (Fan et al., 2023d; Song et al., 2017), such as haul distance, time-related factors, truck running speeds, haul routes, and meteorological conditions (e.g., ambient temperature and precipitation). These usable data and variables may cause a dramatic growth in computation costs and high nonlinearity of prediction models (Fan et al., 2022). Extreme algorithms can overcome these prominent problems, yet this potential remains unknown.

To fill this research gap, the purpose of this study was to provide accurate and rapid estimations of mining truck cycle time by developing extreme machine learning models based on numerous inputs and massive truck haulage data. Unlike previous studies that directly constructed prediction models (Arachchilage et al., 2023; Chanda & Gardiner, 2010), this study conducted two prior comparative studies before proposing the prediction models: (1) comparing feature selection methods and (2) comparing unsupervised clustering techniques. First, three feature selection approaches were utilized to determine the optimal subset of input variables, including decision tree (DT), analysis of variance (ANOVA), and recursive feature elimination (RFE). These methods can remove irrelevant information and noise, thus decreasing computational costs and improving prediction performance (Liu et al., 2022). Next, two unsupervised clustering techniques, including K-means and Gaussian mixture modeling (GMM), were applied to preprocess the datasets with the selected inputs, as these two techniques have been shown to effectively deal with massive data and enhance the model predictability (Fan et al., 2023b). Finally, the ELM, ERT, and XGBoost models of truck cycle time were built based on feature selection and clustering analysis.

The novelty of this study stems from four aspects. First, an insightful analysis was carried out on unique and massive truck haulage data in open-pit mining. Second, three feature selection methods were compared and applied to analyze the real-world factors affecting mining truck cycle time. Third, ELM, ERT, and XGBoost were used to provide rapid truck cycle time estimations for the first time. Forth, the ELM, ERT, and XGBoost models were combined with K-means and GMM to investigate the influence of clustering techniques on models' accuracy. This paper contributes to the development of extreme hybrid models based on feature selection approaches and clustering techniques to estimate mining truck cycle time accurately and rapidly.

5.2. Methodology and Data Preparation

5.2.1. Research framework

Figure 5.1 demonstrates the overall research framework. In Figure 5.1, the data from truck haulage in oil sands mine sites were tabulated and separated into two datasets: training and testing. To remove irrelevant information in these datasets, three feature selection approaches (DT, ANOVA, and RFE) were first applied to select the essential inputs. Next, the best-selected subset was classified by K-means and GMM to identify subgroups from massive data, respectively. Meanwhile, the labels of these subgroups constituted additional variables and were treated as new input variables (i.e., new categorical variables) incorporated with the selected subset to become new training datasets (Fan et al., 2022). The testing dataset was correspondingly classified based on K-means and GMM analysis. After that, ELM, ERT, and XGBoost were adopted to develop prediction models using the new training datasets and the initially selected subset, respectively. Finally, the testing dataset was utilized to assess the prediction performance. Four recommended performance metrics were recommended (Wu et al., 2020): the root mean square error (RMSE), the mean absolute error (MAE), R^2, and running time. Overall, the training procedure was conducted on a personal computer (PC) using R programming (version 4.1.3) in RStudio software. This PC has a 64-bit operating system with 16.0 GB of RAM and an Intel Core i7-12700K (3.60 GHz) processor.

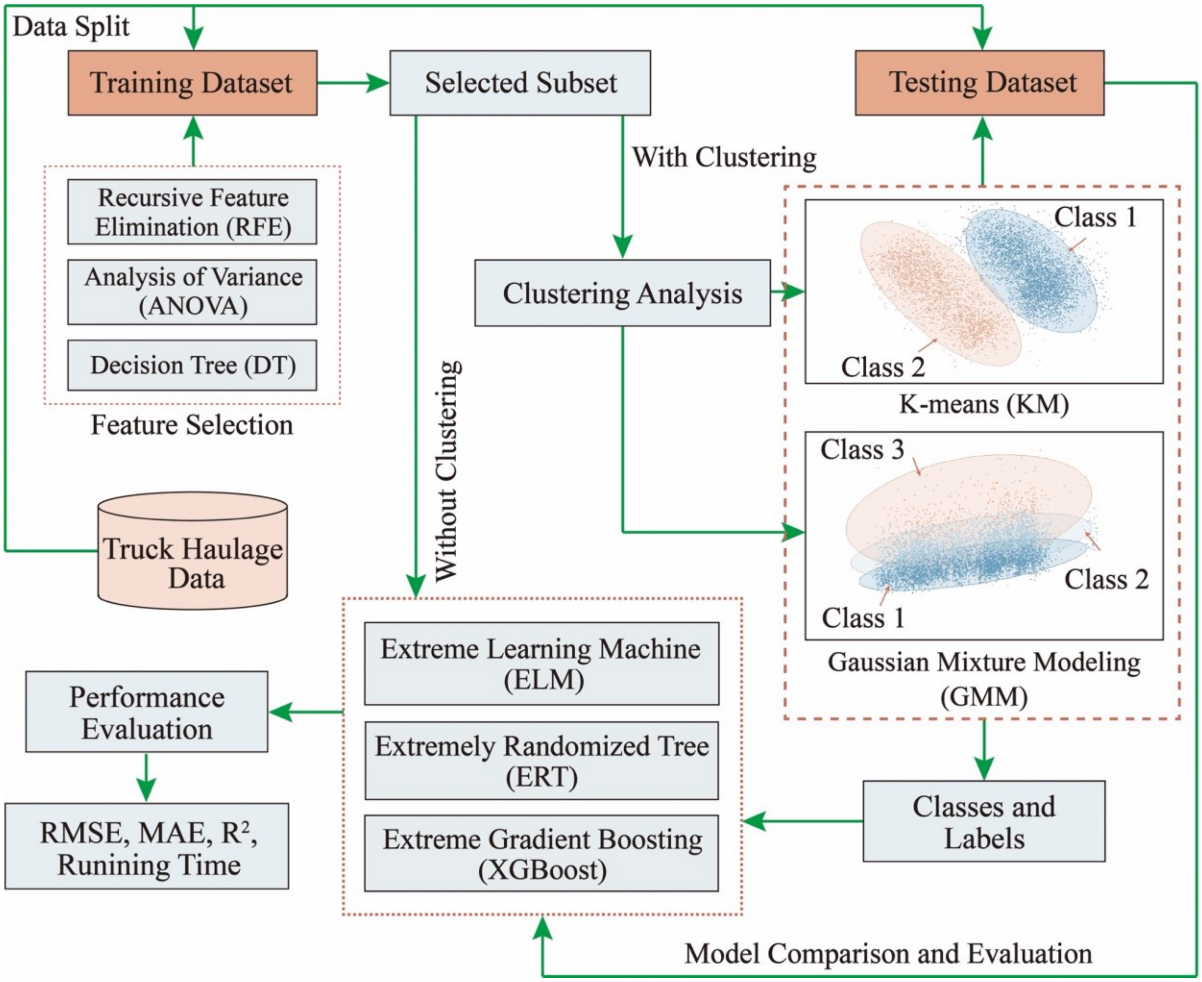

Data Split
Training Dataset
Selected Subset
Testing Dataset
With Clustering
Recursive Feature Elimination (RFE)
Analysis of Variance (ANOVA)
Decision Tree (DT)
Feature Selection
Clustering Analysis
Class 1
Class 2
K-means (KM)
Class 3
Class 2
Class 1
Gaussian Mixture Modeling (GMM)
Without Clustering
Truck Haulage Data
Extreme Learning Machine (ELM)
Extremely Randomized Tree (ERT)
Extreme Gradient Boosting (XGBoost)
Performance Evaluation
RMSE, MAE, R², Runining Time
Classes and Labels
Model Comparison and Evaluation

Figure 5.1 Proposed study framework for extreme machine learning models developed in this study.

5.2.2. Data description

The massive dataset was obtained from operating mine sites in Alberta, Canada. It had 8,683 groups of data points, each representing a truck haulage cycle recorded for each hour of the year. The dataset was partitioned into training (80%) and testing (20%) datasets. This split proportion was determined and recommended based on common practice in previous studies (Arachchilage et al., 2023). Both these two datasets included an output variable (cycle time, y) and ten input variables (x_m). These inputs were haul distance (x_1, km), empty speed (x_2, km/h), ambient temperature (*x_3*, °C), wind speed (*x_4*, km/h), waiting at shovel (*x_5*), waiting at dump (*x_6*), spotting (*x_7*), month (*x_8*), destination (*x_9*), and precipitation (*x_{10}*). They were selected because mining engineers observed these variables at mine sites and used them in previous studies to build prediction models (Chanda & Gardiner, 2010; Fan et al., 2023b). Using the training dataset as an instance, a detailed description of all inputs and the output is presented in Table 5.1. In Table 5.1, the variables are divided into continuous variables (the first five) and categorical variables (the last six). Amid them, the inputs related to truck haulage were provided by the mine data management system; the weather-related inputs were collected from the local meteorological observatory (MEP, 2023). Table 5.1 lists the statistical information of these variables, including minimum, mean, and maximum. Besides, the linear correlation (*r*) between each input and output was calculated by the extensively used Pearson correlation coefficient method (Baek & Choi, 2020). As shown in Table 1, the correlation between some inputs and truck cycle time is low or even zero, such as wind speed (0), waiting at dump (0.05), and spotting (0.08). The contribution of these variables in constructing models may be insignificant, but instead, they may increase the computational time and reduce the model predictability (Liu et al., 2022). This provided the rationale for conducting feature selection analysis (remove or retain these variables) in this study. Furthermore, the statistical distribution features of all inputs and the output are presented in Figure 5.2. Figure 5.2(a)-(e) shows the distribution feature of each continuous variable using a histogram with specific columns. The horizontal direction denotes the variables; the vertical direction represents the density (proportion), indicating the fraction of each column divided by the total number of data points. Figure 5.2(f)-(k) presents the boxplots of six categorical variables.

Table 5.1 Descriptions and statistical information for the output (y) and inputs (x_m) in this study.

Variable	Description	Minimum	Mean	Maximum	r
Cycle time (y, min)	The time for each truck to finish a haulage cycle	14.67	27.82	78.30	1
Haul distance (x_1, km)	The distance between loading and dumping sites	1.02	4.47	11.12	0.37
Empty speed (x_2, km/h)	The truck speed from a dumping site to a loading site	6.40	37.00	59.90	-0.19
Ambient temperature (x_3, ℃)	The hourly ambient temperature at mine sites	-38	0.68	32.80	0.13
Wind speed (x_4, km/h)	The hourly wind speed (at 10 m) at mine sites	0	5.71	34	0
Waiting at shovel (x_5)	Without and with wait time at a shovel for each truck cycle: two labels (0 and 1)	0	0.49	1	0.27
Waiting at dump (x_6)	Without and with wait time at a dumping site for each truck cycle: two labels (0 and 1)	0	0.96	1	0.05
Spotting (x_7)	Without and with spotting time for each truck cycle: two labels (0 and 1)	0	0.63	1	0.08
Month (x_8)	12 months of the year: twelve labels (1-12)	1	6.55	12	0.15
Destination (x_9)	Three dumping sites of truck haulage at mine sites: the labels (1, 2, and 3)	1	2.19	3	0.14
Precipitation (x_{10})	Without and with hourly precipitation: two labels (0 and 1)	0	0.05	1	0.10

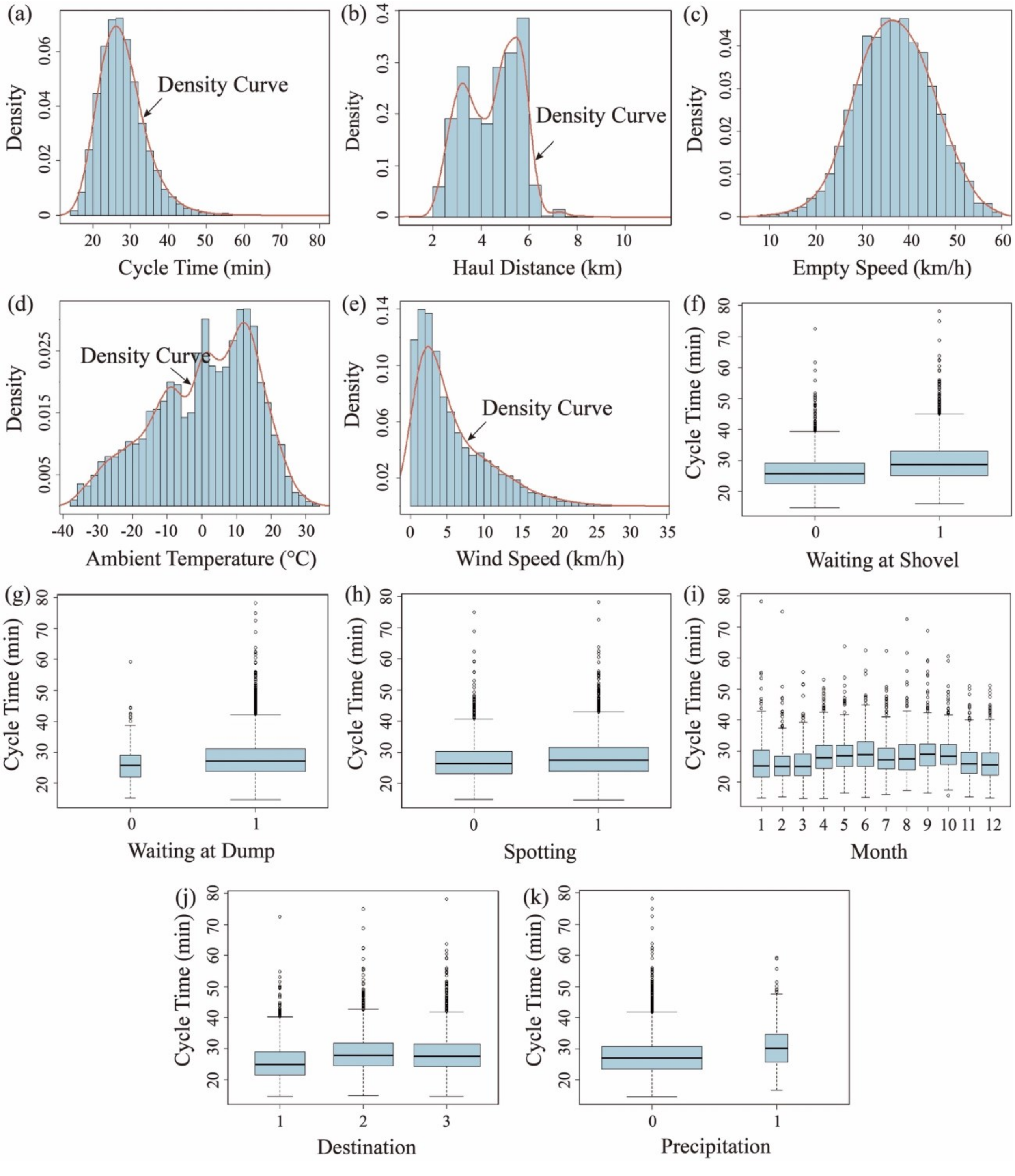


Figure 5.2 Distributions with density curves and boxplots for the (a) output and (b)-(k) input variables.

5.2.3. Feature selection approaches

Feature selection approaches are broadly classified into wrapped, embedded, and filter methods (Boratto et al., 2023; Olu-Ajayi et al., 2023). This study carried out a comparative study of three feature selection approaches to determine the optimal subset of input variables. The basic ideas are briefly introduced below. The wrapped method, such as RFE, usually adopts specific machine learning algorithms (e.g., linear model or RF) to assess subsets of input variables and select the best subset that produces the highest performance. RFE works by recursively removing the least essential input variables from a specific model (e.g., RF) according to the change in model accuracy or some other criterion (e.g., regression coefficients) until the optimal number of inputs is achieved (Bahl et al., 2019). The embedded method directly incorporates feature selection into the model training of a machine learning algorithm, such as DT. DT selects input variables for the best division at the root and internal nodes; simultaneously, it ranks the relative variable importance using some criterion (e.g., Gini index) (Zhou et al., 2021). Unlike the wrapped and embedded methods, the filter method, such as ANOVA, uses statistical tests (e.g., significance tests) to estimate the relevance of input variables, which is independent of any particular machine learning method. ANOVA computes the *p*-value when sequentially adding each input into a prediction model (Sheikhan et al., 2013). Meanwhile, the inputs with the lowest *p*-values (e.g., < 0.01) are retained, indicating a strong correlation with the output.

5.2.4. Clustering analysis algorithms

K-means and GMM are two extensively applied unsupervised clustering techniques since they are easy to implement and efficient for dealing with massive data (Capó et al., 2017). They both identify several classes from a data population and assign data points with more similarities to the same group (Fan et al., 2022). The key distinction between the two techniques lies in the principles of assigning data points to classes. K-means assumes that each data point falls into the specific class where the centroid is closest to it. This centroid is updated iteratively until the squared distance sum between the centroid and each data point is minimized (Liu et al., 2020). Unlike K-means, GMM is known as a probability-based clustering approach that assigns data points to a specific class when they have the maximum class posterior probability (Grün & Leisch, 2007). Detailed information on K-means and GMM is described below.

5.2.4.1. K-means

K-means is a distance-based unsupervised clustering approach, which segments a data population into K subsets based on the difference in distance between data points. Assuming a group of data points {X_1, X_2, …, X_n}, where X_i indicates the *i*th data point. K-means finds the centroids {c_1, c_2, …, c_K} for K classes that minimize the squared sum of the distance between a data point X_i and its closest centroid c_k. This distance $Dis(X_i, c_k)$ can be calculated as follows:

$$Dis(X_i, c_k) = \sqrt{(X_i)^2 - (c_k)^2} \tag{5-1}$$

Initially, K-means randomly selected K centroids for the total data points. Next, the distance between the initial centroid and each data point is computed using Equation (5-1). Under the principle of the nearest centroid, the data are assigned to the same latent class that is labeled as C_k, which can be expressed as (Liu et al., 2020)

$$C_k = argmin_{k\in\{1,2,\ldots,K\}} Dis(X_i, c_k) \tag{5-2}$$

After that, K-means recalculate the locations of K centroids once all data points are assigned to the class C_k according to the following formula:

$$c'_k = (\textstyle\sum_{X\in C_k} X)/p \tag{5-3}$$

where p denotes the total number of data points in a specific class after K-means clustering.

K-means conducts this process iteratively (i.e., calculating distance and finding new centroids) until these centroids no longer change. Moreover, although the number of K (K > 1) is set by users according to corresponding clustering requirements, the optimal number can be determined by the gap statistic method, which is suggested by Sinaga and Yang (2020).

5.2.4.2. Gaussian mixture modeling

GMM has shown its performance in many aspects of handling massive data, such as streamflow prediction (Ni et al., 2020), wind power forecast (Ge et al., 2018), and heat load pattern recognition (Lu et al., 2019). GMM models the distribution of a data population by assuming that it comprises a mixture of Gaussian distributions. In other words, GMM assumes that the data population is generated by finite subsets (i.e., latent classes), each with its own Gaussian distribution (Fan et al., 2022). The distribution function of the mixture of Gaussians can be formulated as (Leisch, 2004)

$$P(y|x,\emptyset) = \sum_{k=1}^{K} \pi_k f_k(y|x,\mu_k,\Sigma_k) \tag{5-4}$$

where $P(y|x,\emptyset)$ indicates the distribution function of data points. $\emptyset$ denotes the parameter set $\{\pi_k,\mu_k,\Sigma_k\}$ of the distribution function. K is the number of latent classes. π_k is the mixture coefficient (non-negative) of the kth latent class with $\sum_{k=1}^{K}\pi_k = 1$. $f_k(y|x,\mu_k,\Sigma_k)$ is the distribution function of the kth class. μ_k and Σ_k are means and variances.

GMM aims to estimate the parameter set that constitutes the mixture of Gaussians. To achieve this aim, GMM uses a two-step strategy, which is known as Expectation-Maximization (EM). In the E-step, the probabilities (γ_{nk}) assigned to latent classes are calculated for each data point (Leisch, 2004):

$$\gamma_{nk} = \frac{\pi_k f_k(y_n|x_n,\mu_k,\Sigma_k)}{\sum_{k=1}^{K} \pi_k f_k(y_n|x_n,\mu_k,\Sigma_k)} \tag{5-5}$$

In the M-step, the parameter set is estimated with the γ_{nk} through maximizing the log-likelihood (*log L*) (Leisch, 2004):

$$log\, L = \sum_{n=1}^{N} \log(P(y|x,\phi)) = \sum_{n=1}^{N} \log(\sum_{k=1}^{K} \pi_k f_k(y|x,\mu_k,\Sigma_k)) \tag{5-6}$$

where $\pi_k = \frac{1}{N} f_k = \frac{1}{N} \sum_{n=1}^{N} \gamma_{nk}$. *N* refers to the number of data points. In GMM, the EM algorithm iteratively estimates the parameters until the maximized *log L* is attained. Finally, the optimal number of latent classes is obtained by the minimization of the Bayesian information criterion (BIC) (Mehrjou et al., 2016):

$$BIC = -2logL + ClogN \tag{5-7}$$

where *C* indicates the number of parameters.

5.2.5. Machine learning algorithms and performance metrics

This study used three extreme machine learning algorithms (ELM, ERT, and XGBoost) to develop prediction models of mining truck cycle time. These extreme algorithms are characterized by computational efficiency, whose principles are briefly introduced below:

ELM is a neural network (NN) algorithm consisting of three layers: an input layer, an output layer, and a hidden layer (Fan et al., 2023d), as depicted in Figure 5.3. ELM can handle massive amounts of data, solve classification and regression tasks, and construct complex nonlinear relationships between inputs and outputs (Huang et al., 2006). Unlike traditional NNs, the built-in parameters (e.g., weights between input and hidden nodes) in ELM are not required to be tuned because these parameters are randomly initialized and remain constant throughout the training process (Dhini et al., 2022). This makes ELM extremely efficient and fast compared to other ANNs.

ERT is a tree-based ensemble algorithm that integrates numerous DTs (see Figure 5.4) for regression and classification tasks (Saeed et al., 2021). ERT is similar to RF, which builds a series of trees in a parallel way based on a bagging technique (i.e., bootstrapping and aggregation techniques) to improve model performance (Fan et al., 2023b). "Bootstrapping" indicates a sampling method that randomly samples subsets from the initial dataset with a replacement for training DTs. Meanwhile, a sample of input variables is arbitrarily selected at each DT node for best splitting. "Aggregation" is the averaging of the decisions across all DTs.

Unlike RF, ERT makes splits at nodes based on entirely random input variables and thresholds. This results in more diverse and faster tree growth in ERT, thus enhancing model accuracy and generalization capability (Zhang et al., 2023). XGBoost is another tree-based ensemble method for solving supervised learning problems, such as regression and classification (Chen & Guestrin, 2016). Unlike ERT, XGBoost builds a large number of DTs in a sequential manner based on a boosting technique, each of which attempts to learn and correct the errors of the preceding tree. In addition, XGBoost performs fast computation mainly because it adopts some built-in techniques, such as approximate greedy algorithm, parallel processing, and tree pruning (Chen & Guestrin, 2016). These techniques help to prevent overfitting problems, improve model performance, and make the XGBoost algorithm more efficient (Fan et al., 2023d).

These extreme models were constructed in RStudio with installed packages "*elmNN*", "*ranger*", "*xgboost*", and "*caret*". To evaluate the model performance, three quantitative indicators were used: MAE, RMSE, and R^2, which are given below (Huo et al., 2021; Zhu & Xie, 2023):

$$MAE = \frac{1}{N}\sum_{n=1}^{N} |y_n - \hat{y}_n | \quad (5\text{-}8)$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(y_n - \hat{y}_n)^2} \quad (5\text{-}9)$$

$$R^2 = 1 - \frac{\sum_n^N (y_n - \hat{y}_n)^2}{\sum_n^N (y_n - \bar{y}_n)^2} \quad (5\text{-}10)$$

where y_n is the measured cycle time; $\hat{y}_n$ denotes the predicted cycle time, and $\bar{y}_n$ represents the mean of measured cycle time. MAE is the mean of absolute errors between the measured cycle time and the predicted cycle time. RMSE is the standard deviation of the errors between the measured and the predicted cycle time. R^2 is a measure of the goodness of fit, ranging from zero to one.

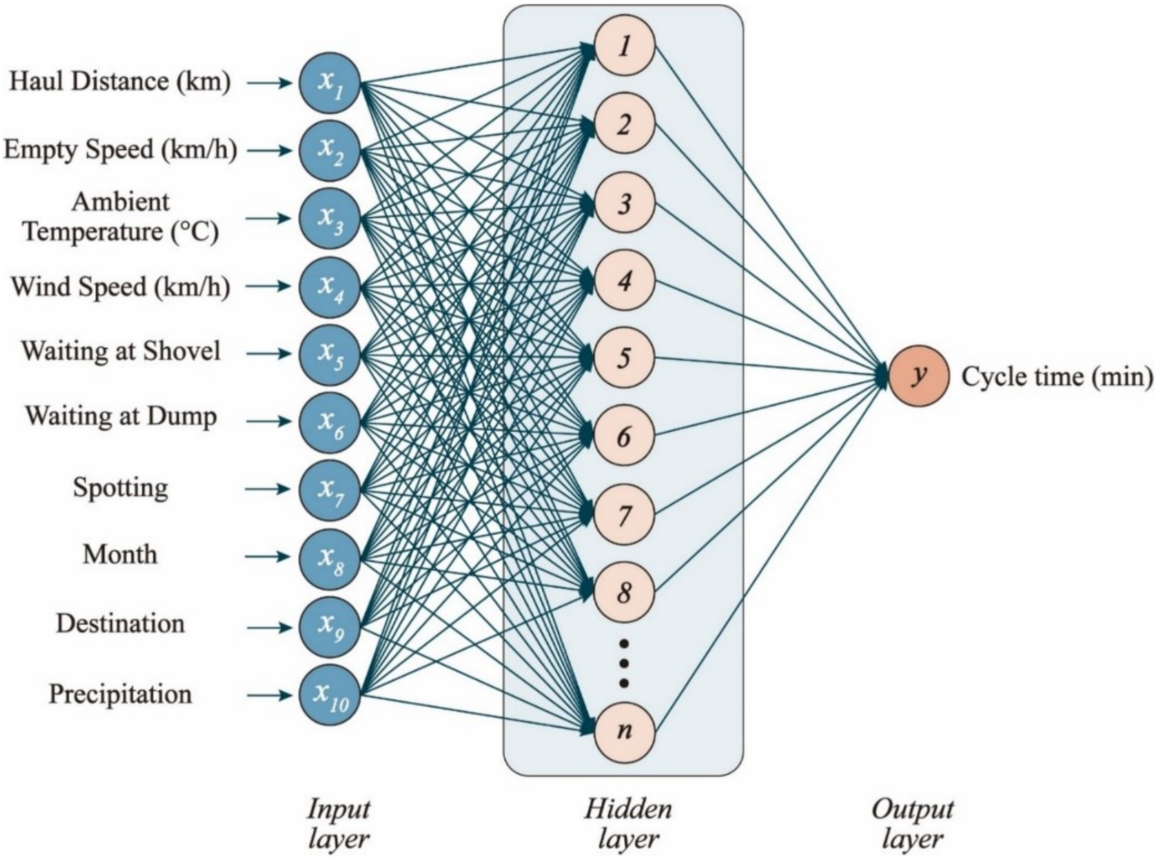


Figure 5.3 Typical architecture of the ELM model.

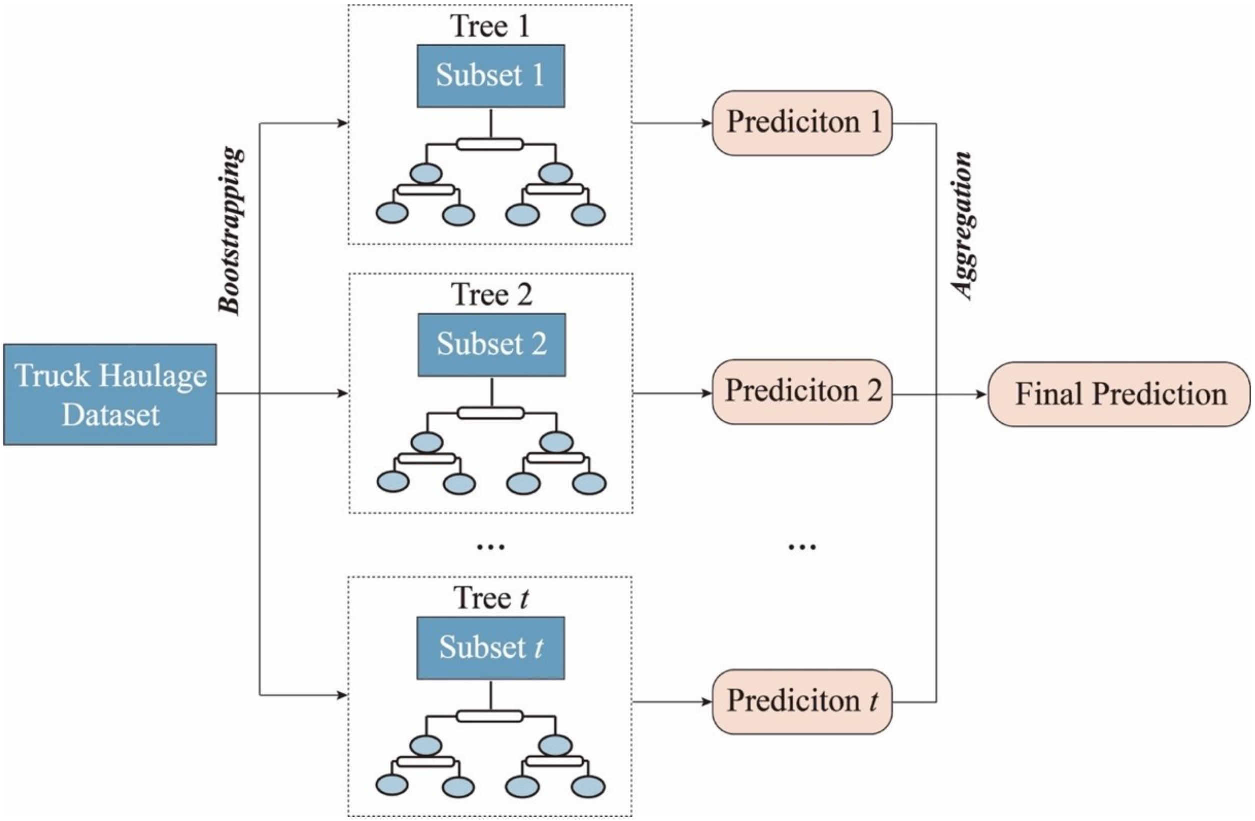


Figure 5.4 Typical architecture of the ensemble tree-based models.

5.3. Results and Discussion

5.3.1. Feature selection and feature importance

Three feature selection approaches, including DT, ANOVA, and RFE, were availed in this study to remove redundant information (i.e., irrelevant input variables) from the datasets. Then, the input variables chosen by each algorithm were applied to build and compare the ML model (e.g., XGBoost) to determine the best feature selection method (Liu et al., 2022). Using the RFE algorithm as an example, Figure 5.5 illustrates the results of feature selection under the principle of minimizing RMSE values. In Figure 5.5, the number of input variables is increased from one to ten. When the number reaches five, RMSE drops significantly from 5.90 to 5.19. It descends further to 5.00 when the number attains seven and cannot decrease as the number of input variables grows. Therefore, RFE retained seven input variables. These inputs (in Table 5.2) were haul distance, empty speed, waiting at shovel, month, destination, ambient temperature, and precipitation. From Table 5.2, ANOVA also selected seven input variables, but they were a new combination of haul distance, empty speed, waiting at shovel, waiting at dump, spotting, precipitation, and month. Compared to RFE and ANOVA, DT only kept six input variables, including waiting at shovel, haul distance, empty speed, destination, ambient temperature, and spotting. Afterward, these three subsets of inputs and the original inputs were involved in constructing four XGBoost models (i.e., RFE-XGBoost, ANOVA-XGBoost, DT-XGBoost, and XGBoost), whose performance was evaluated by the testing dataset. In Table 5.2, the RFE-XGBoost model has the smallest RMSE (5.12), MAE (3.78), and the highest R^2 (33.54%), which performs better than the DT-XGBoost (5.13, 3.79, and 33.27%), ANOVA-XGBoost (5.15, 3.79, and 32.90%), and XGBoost (5.13, 3.79, and 33.37%) models. Moreover, the running time of the RFE-XGBoost model (305.4 s) was lower than that of the XGBoost model (320.4 s), although it was longer than the ANOVA-XGBoost (300.6 s) and DT-XGBoost models (273.6 s). Akin to the study by Liu et al. (2022), they adopted three feature selection methods (including the RFE algorithm) to eliminate irrelevant inputs and trained the XGBoost models for estimating the buildings' energy consumption. In terms of the model's performance and running time, the RFE-XGBoost had the highest R^2 (72.8%) and shortest running time (262 s). In short, RFE was the best feature selection method that improved the model performance and removed redundant variables, thus reducing the computational cost.

Furthermore, RFE determined the variable importance of the selected seven inputs by iteratively removing the least important features and evaluating the built-in model's performance (Bahl et al., 2019). As shown in Figure 5.6, the importance scores (in percentage) of these inputs are ranked as follows: haul distance (32.60%) > empty speed (21.15%) > waiting at shovel (20.00%) > month (9.04%) > destination (7.36%) > ambient temperature (6.62%) > precipitation (3.23%). Among these inputs, haul distance, empty speed, and waiting at shovel were the most critical variables affecting truck cycle time. At mine sites, mining engineers usually built a fitted line (i.e., prediction model) between truck cycle time and haul distance because they observed that haul distance had a dominant impact on truck cycle time (Cervantes et al., 2019). After haul distance, empty speed had the second-highest importance score (21.25%) since it can determine the travel time (a portion of cycle time) between dumping and loading sites (Schexnayder et al., 1999). Waiting at shovel was a binary variable (zero and non-zero) in this study, with an importance of 20.00%. In accordance with Ercelebi and Bascetin (2009), truck cycle time consisted of loading, hauling, dumping, returning time, and waiting time at dumps or shovels. Hence, the increase in waiting time contributes to a growth in cycle time. Moreover, other inputs influenced the truck cycle time to some extent. The variable importance of month was 9.04%, which may be attributed to the variations of time-related factors such as ambient temperature and precipitation in different months. For example, according to Ma et al. (2023), truck tire temperature escalated from 54 °C to 69 °C when the ambient temperature at mine sites increased from 20 °C to 40 °C. This led to a reduction in tire performance (e.g., fatigue), thus increasing truck cycle time and decreasing truck productivity (Ma et al., 2022). Due to the potential effect of multicollinearity (Paliwal & Kumar, 2011), the importance of ambient temperature (6.62%) and precipitation (3.23%) was relatively lower than that of month (9.04%). Lastly, destination had an importance of 7.36%, which may be because destination is associated with truck haul length, thus affecting truck cycle time (Both & Dimitrakopoulos, 2020). In addition, the importance can be assessed by investigating the effect of input size on the prediction performance, as shown in Table 5.3. For instance, XGBoost showed the largest improvement in R^2 (from 17.55% to 33.54%) when the model involved haul distance, indicating that it was the most influential input. This agreed with our previous studies (Fan et al., 2022, 2023d). In brief, the RFE selected seven input variables and provided importance scores for each input. These scores help mining engineers gain insight into the main factors influencing truck cycle time at mine sites, leading to more accurate predictions of truck cycle time and the estimation of overall mine productivity (Fan et al., 2023b).

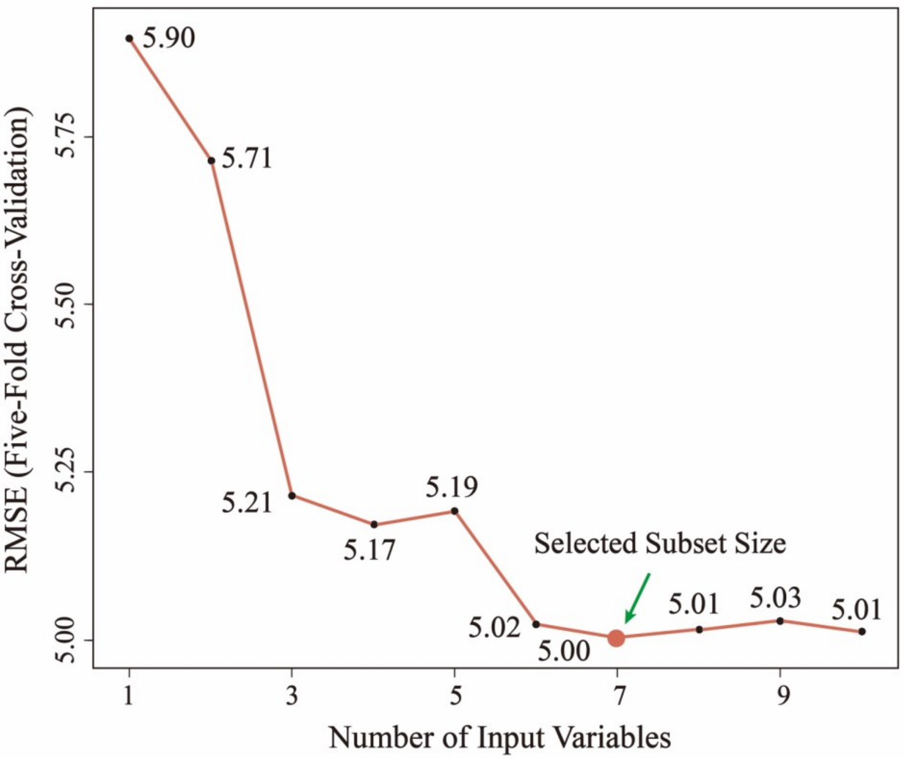


Figure 5.5 Feature selection analysis using recursive feature elimination based on the training dataset.

Table 5.2 Comparison of feature selection methods based on the XGBoost models.

Method	Model	Subset Size	Selected Input	Performance Metrics			Time (s)
				RMSE	MAE	R^2 (%)	
RFE	RFE-XGBoost	7	Haul distance, empty speed, waiting at shovel, month, destination, ambient temperature, precipitation	5.12	3.78	33.63	305.4
ANOVA	ANOVA-XGBoost	7	Haul distance, empty speed, waiting at shovel, waiting at dump, spotting, precipitation, month	5.15	3.79	32.90	300.6
DT	DT-XGBoost	6	Waiting at shovel, haul distance, empty speed, destination, ambient temperature, spotting	5.13	3.79	33.27	273.6
Null	XGBoost	10	Haul distance, empty speed, ambient temperature, wind speed, waiting at shovel, waiting at dump, spotting, month, destination, precipitation	5.13	3.79	33.37	320.4

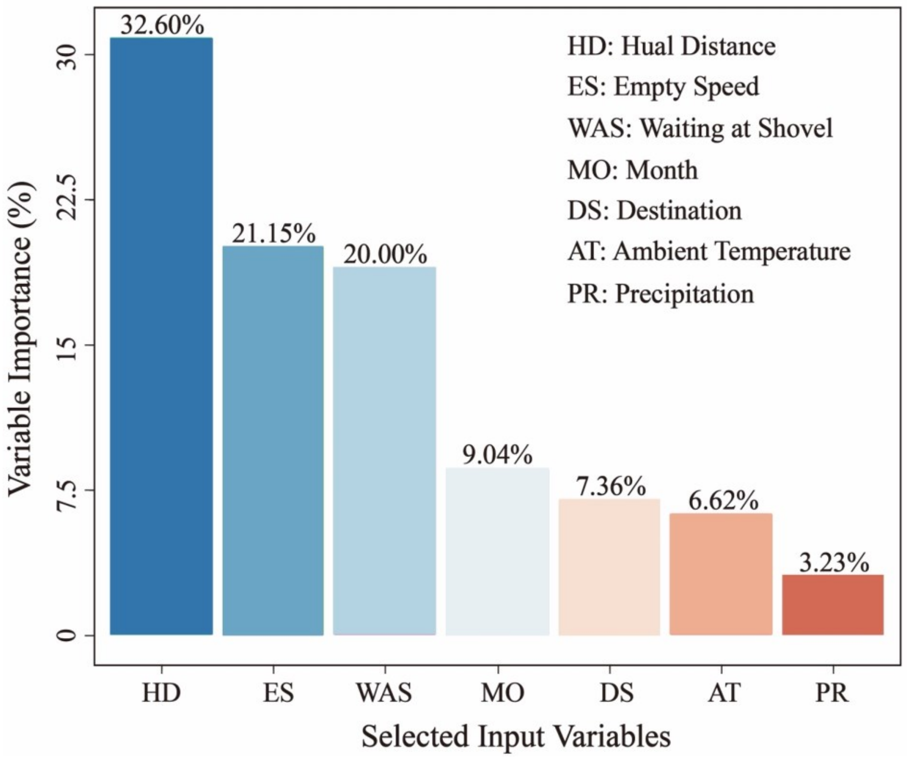


Figure 5.6 Variable importance analysis based on recursive feature elimination.

Table 5.3 Different input sizes of input variables and the resulting model performance.

XGBoost	Input Size	Performance Metrics		
		RMSE	MAE	R^2 (%)
Model 1	Precipitation	6.28	4.76	0.63
Model 2	Precipitation + ambient temperature	6.17	4.69	3.37
Model 3	Precipitation + ambient temperature + destination	6.11	4.61	5.21
Model 4	Precipitation + ambient temperature + destination + month	6.10	4.59	5.68
Model 5	Precipitation + ambient temperature + destination + month + waiting at shovel	5.88	4.43	12.41
Model 6	Precipitation + ambient temperature + destination + month + waiting at shovel + empty speed	5.70	4.28	17.55
Model 7	Precipitation + ambient temperature + destination + month + waiting at shovel + empty speed + haul distance	5.12	3.78	33.54

5.3.2. Clustering analysis based on truck haulage data

To improve truck cycle time predictions, this study used K-means and GMM to classify the massive truck haulage data (including the seven selected inputs by RFE). The clustering analysis of these two techniques will be presented as follows.

5.3.2.1. K-means clustering analysis

Figure 5.7 shows the clustering results using K-means based on the truck haulage data (training dataset). According to the gap statistic method (Tibshirani et al., 2001), two classes were ultimately recommended as the optimal number of classes. In Figure 5.7, these two classes are visualized in a two-dimensional scatter plot with two primary principal components (PC1 and PC2, with weights of 24.4% and 17.9%). This is because principal component analysis extracts predominant features from all input variables, which can be utilized to intuitively display the data distributions by a two-dimensional (2D) graphical demonstration (Fan et al., 2023c). As presented in Fig. 7, it can be noted that Class 1 and Class 2 are almost entirely divided into two datasets with different centroids. In Class 1, the number of data points was 4,356, whereas the number in Class 2 was 2,591. Moreover, using eight boxplots, Fig. 8 illustrates the statistical characteristics of inputs and output in these two classes. The first four represent the relationships among the classes and continuous variables; the last four show the relationships among the classes and categorical variables. As can be seen from these boxplots, for the continuous variables, only ambient temperature (Fig. 8(d)) is well split into two classes, with mean values of 10.03 ℃ and -15.04 ℃, respectively. All other continuous variables have overlapping distributions in two classes. Likewise, for the categorical variables, a clear relationship can be uniquely observed between months and classes (Fig. 8(f)). The data points in Class 1 are almost distributed from March to October, while the data points in Class 2 are mainly present in January, February, November, and December. This corresponds to the values of ambient temperature in two classes, representing the warm and cold temperatures in Northern Alberta, Canada. Therefore, it can be concluded that these two classes recognized by K-means from the massive truck haulage data were closely related to the ambient temperature. Similar results were discovered by Liu et al. (2020), who applied K-means to classify the single-crystal superalloy creeping data and developed prediction models of creep rupture life. The research showed that K-means identified eight homogeneous classes that were intimately associated with the creep mechanisms, which further improved the model accuracy of creep rupture life with an increase in R^2 from 71.02% to 91.76%.

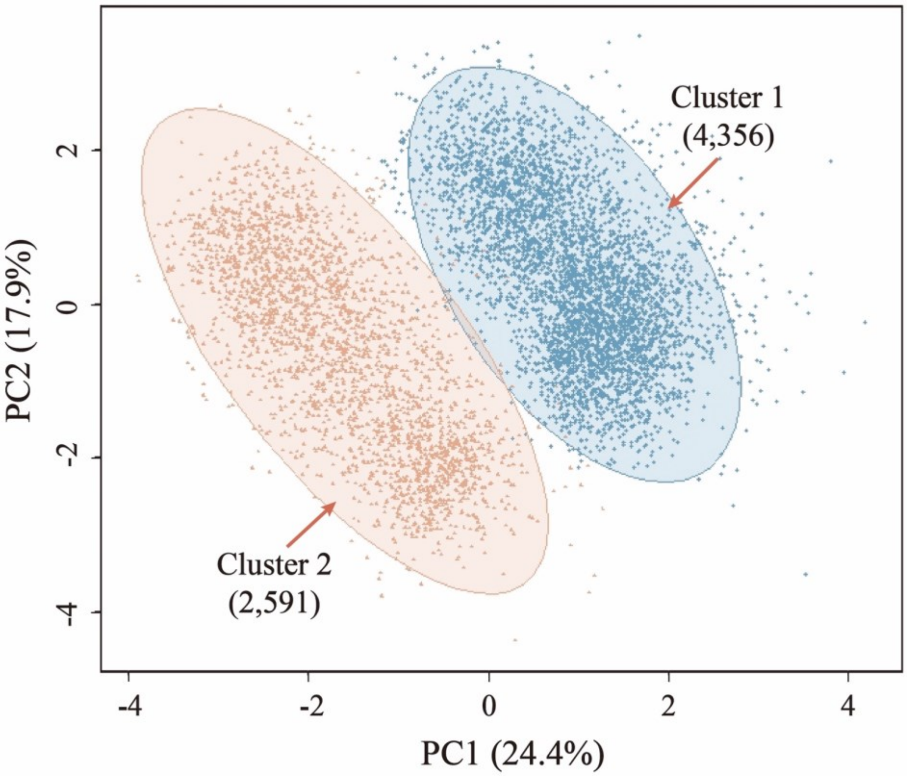


Figure 5.7 Identifying two classes from the K-means clustering (Distribution of data points in each class represented by a 2D scatter diagram with two major principal components).

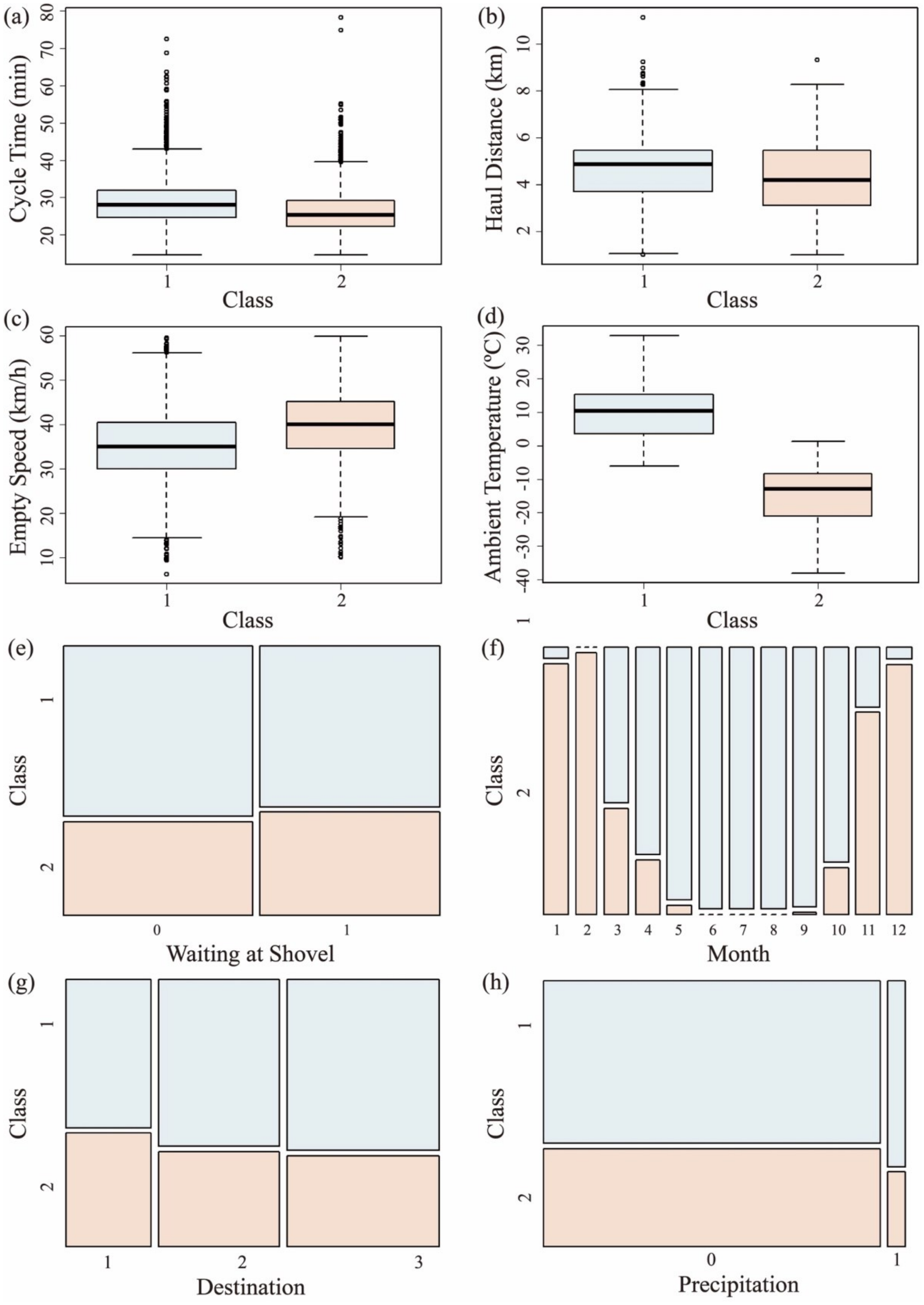


Figure 5.8 Boxplots of input variables to corresponding classes based on the K-means clustering.

5.3.2.2. GMM clustering analysis

In addition to K-means, this study adopted the GMM algorithm to classify massive truck haulage data. Following the principle of minimizing the BIC value (McLachlan et al., 2019), the optimal amount of classes was ascertained. As a result, three classes were extracted from the training dataset by GMM, as shown in Figure 5.9. These three classes were visualized in a two-dimensional scattering diagram of cycle time and haul distance. The data points in each class were covered by an ellipse that represented a Gaussian distribution (Shimizu & Kaneko, 2020). The number of data points varied in these three classes. Classes 1 (3,471) and 2 (3,099) had more than 3,000 data points, while Class 3 contained 377 data points. Despite this, truck cycle times grow accordingly in each class as haul distance increases. This is in agreement with the site observations (Chanda & Gardiner, 2010). Furthermore, Figure 5.10 shows the data distribution in each class using histograms and boxplots. After conducting GMM analysis, a large amount of truck haulage data came from three different populations, each of which followed a standard Gaussian distribution (e.g., as evidenced by the density curves) according to the definition of GMM (Fan et al., 2022). Figure 5.10(c) presents the relationship between cycle time and classes. Among these classes, truck cycle time significantly varied, showing the order: Class 3 > Class 2 > Class 1 (e.g., the mean values were 41.26 min, 30.44 min, and 24.02 min). This indicates short, medium, and long truck cycle times at mine sites. Hence, these classes extracted by GMM were well linked to truck cycle time (i.e., the output variable). This is comparable with the studies of Ni et al. (2020) and Lu et al. (2019). In the study of Ni et al. (2020), they adopted GMM to extract two latent classes (low and high streamflows) from massive hydrological data and built an XGBoost model for forecasting monthly streamflow. The results presented that the model’s performance increased by about 11% based on GMM analysis. Similarly, Lu et al. (2019) classified building heating data using GMM to identify sub-patterns (including six classes). After that, they trained models for predicting the hourly heating load, whose performance was enhanced by approximately 20% because of GMM analysis.

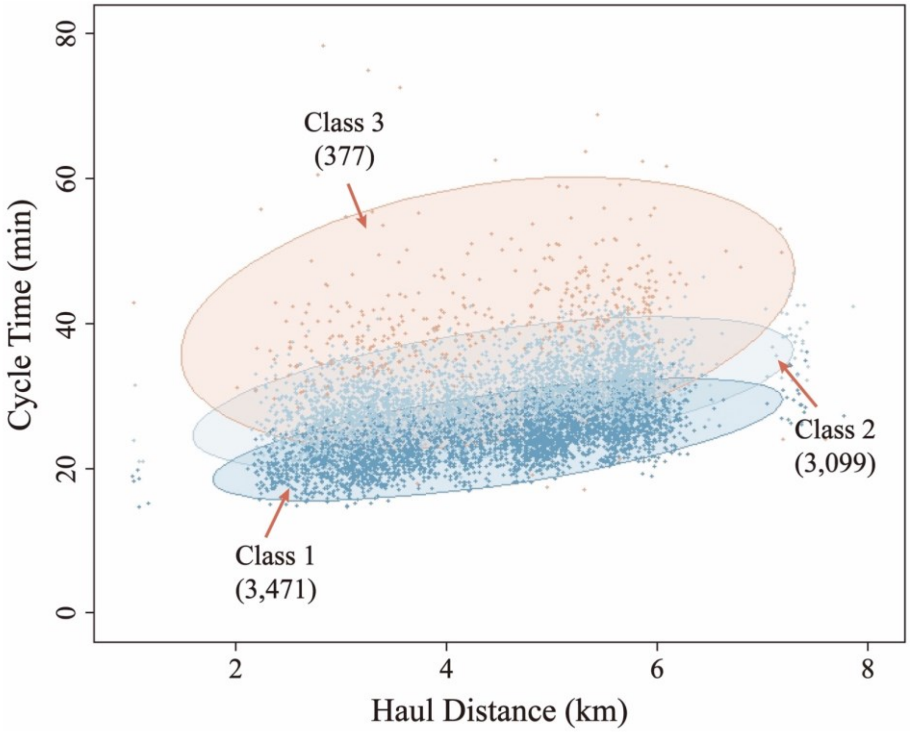


Figure 5.9 Scattering distribution of data points in three classes (represented by a two-dimensional scattering plot of cycle time and haul distance).

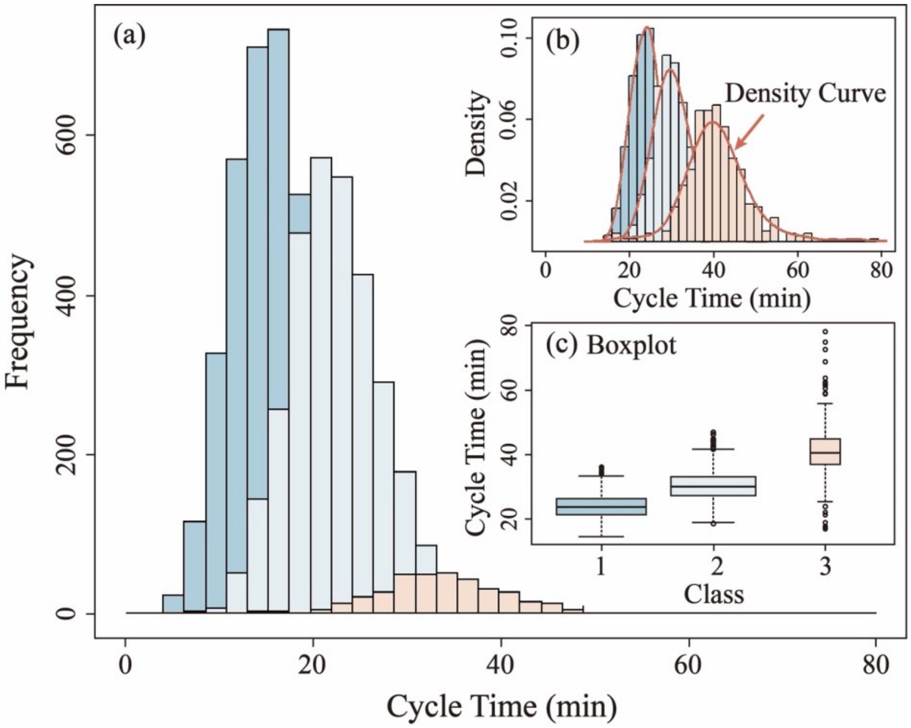


Figure 5.10 Identifying three classes from the GMM analysis. (a) The data in each class follow standard Gaussian distributions. (b) Density curves for each class. (c) A boxplot for three classes.

5.3.3. Evaluation of clustering-based extreme machine learning models

In Section 5.3.1, seven input variables were selected by RFE. Based on these inputs and the output, in Section 5.3.2, two unsupervised clustering techniques, including K-means (abbreviated as KM only in models) and GMM, were adopted to identify classes from the massive truck haulage data. The labels of these classes constituted additional variables and were considered as new inputs (i.e., new categorical variables) incorporated with the selected inputs to become new training datasets (Fan et al., 2022, 2023b). After that, the ELM, ERT, and XGBoost algorithms were applied to establish prediction models of truck cycle time. With the new training datasets, six hybrid prediction models were constructed, which were referred to as the KM-RFE-ELM, KM-RFE-ERT, KM-RFE-XGBoost, GMM-RFE-ELM, GMM-RFE-ERT, and GMM-RFE-XGBoost models.

Moreover, three baseline models (RFE-ELM, RFE-ERT, and RFE-XGBoost) were built on the original dataset (i.e., the seven inputs and the output) and compared to the clustering analysis-based models.

Figure 5.11 exhibits the scatter points of measured cycle time (horizontal) from the testing datasets and the predicted cycle time (vertical) from these nine prediction models. The more evenly distributed these scatter points are along the 45-degree diagonal, the better the forecasting is (Fan et al., 2023b). In Figure 5.11(a)-(f), the scatter points from these six prediction models (baseline and K means-based models) are dispersedly distributed along the diagonal, especially for the longer truck cycle time. This is because many complex factors contribute to the long cycle time at mine sites, such as weather changes, equipment overhaul, work shifts, and road maintenance (Alarie & Gamache, 2002; Fan et al., 2023d); however, these factors were not included yet as inputs due to the data availability, resulting in poor predictions of long cycle time. Compared to Figure 5.11(a)-(f), Figure 5.11(g)-(i) shows the scatter points that are uniformly distributed along the diagonal, indicating that the GMM-based models worked well in predicting cycle time. It can also be seen in Figure 5.12 through three performance metrics: RMSE, MAE, and R^2. For instance, GMM-RFE-XGBoost had the highest R^2 (80.37%) and the least RMSE (2.78) and MAE (1.98). Its performance was considerably higher than the KM-RFE-XGBoost (5.14, 3.80, and 33.06%) and RFE-XGBoost (5.12, 3.78, and 33.63%) models. Therefore, two findings can be drawn from this study: (1) GMM significantly improved the models' accuracy. This is mainly because the truck haulage data usually have a mixture of Gaussian distributions, referred to as multi-peaked Gaussian distributions (as shown in Figure 5.2(b) and (d)) (Bishop, 2006). Depending on these specific peaks, GMM can recognize classes from massive data and extract corresponding latent variables (Ge et al., 2018). In Section 5.3.2.2, the additional variable from GMM analysis was strongly connected with truck cycle time. As a consequence, the model performance can be enhanced when combining these latent variables as new inputs in building prediction models. This is consistent with our previous studies (Fan et al., 2022, 2023b); for example, the R^2 of the truck productivity prediction model was increased from 44% to 87% based on GMM analysis. (2) K-means was unable to increase the models' predictability. In Section 5.3.2.1, a categorical variable with two labels was extracted through K-means analysis, which showed a close relationship with ambient temperature. Nevertheless, ambient temperature has been considered an input variable in predicting truck cycle time, whose variable importance was 6.62% (see Section 5.3.1). The inclusion of the categorical variable obtained from K-means analysis may increase the model's multicollinearity and computational complexity, thus reducing the model's accuracy (Paliwal & Kumar, 2011). In summary, GMM performed better than K-means in enhancing the model predictability. This study was similar to the results from Virupakshappa and Oruklu (2019); they built prediction models for detecting and positioning flaw echoes in ultrasonic data based on three unsupervised clustering techniques, including K-means and GMM. The results showed that the detection accuracy achieved 93% with GMM, which was higher than that of K-means (89%).

Furthermore, an additional finding can be obtained by comparing these extreme machine learning models: the XGBoost algorithm outperformed the ELM and ERT algorithms in predicting truck cycle time. For example, for the baseline models, the R^2 of the RFE-XGBoost model was 33.63%, while the RFE-ELM and RFE-ERT models were 29.95% and 27.64%. This applies to the clustering-based models. For example, the R^2 of GMM-RFE-XGBoost was 80.37%, which was higher than GMM-RFE-ELM (79.10%) and GMM-RFE-ERT (75.81%). This may be because XGBoost constructs a loss function and adopts a gradient descent method to find the optimal weights for each feature, thus increasing the model predictability (Chen & Guestrin, 2016). In

ELM, the weights are randomly chosen and optimized by a linear algorithm to reduce the residuals between predicted and actual values (Huang et al., 2006). As for ERT, it simply averages the predictions over all randomized trees (Geurts et al., 2006). This is close to Cao et al. (2022), who built the XGBoost and ELM models to predict the deformation and damage of super-high arch dams. The research proved that the XGBoost model's RMSE (0.90) was lower than the ELM model (1.10). Janizadeh et al. (2022) also demonstrated that the XGBoost model (R^2 = 92%) performed better than the ERT model (R^2 = 91%) when mapping the flood hazard susceptibility.

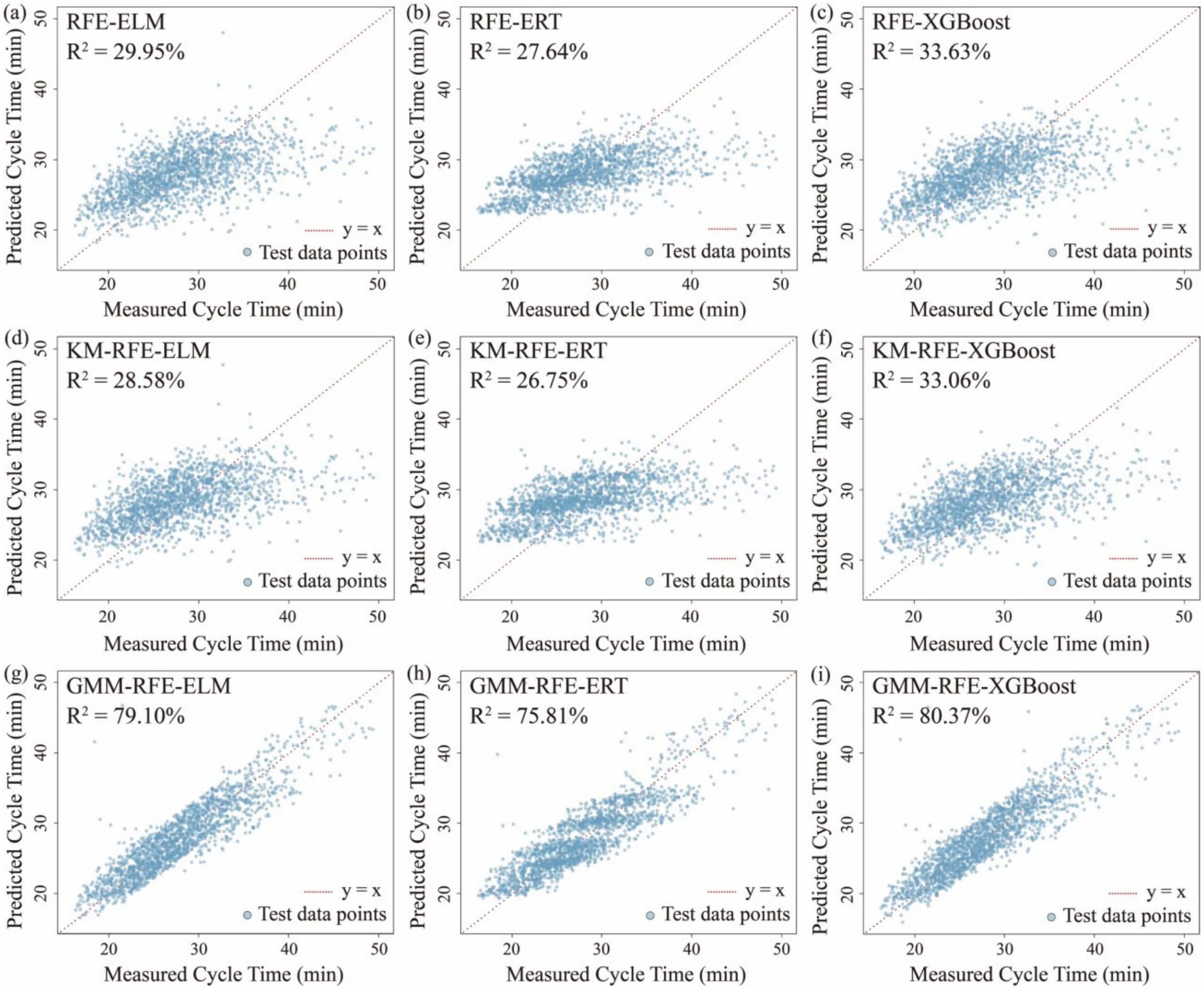


Figure 5.11 Scatterplots of the predicted cycle time and measured cycle time from nine prediction models.

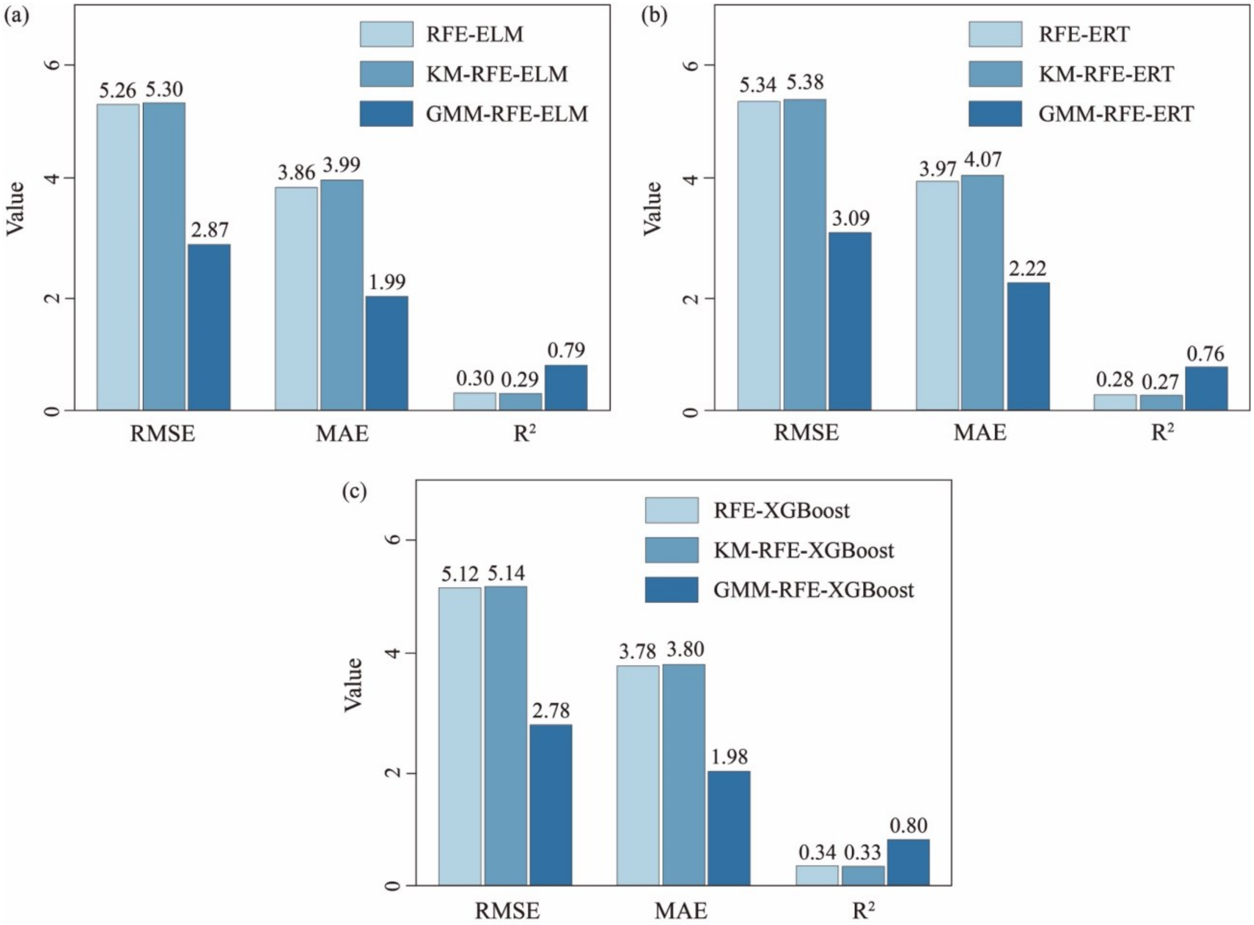


Figure 5.12 Comparisons of extreme machine learning models with and without clustering analysis. (a) Trained models using extreme learning machine (ELM). (b) Trained models using extremely randomized tree (ERT). (c) Trained models using extreme gradient boosting (XGBoost).

5.3.4. Rapid estimation of extreme machine learning models

In the last section (5.3.3), the GMM-based models (i.e., GMM-RFE-XGBoost, GMM-RFE-ELM, and GMM-RFE-ERT) performed best in estimating truck cycle time at mine sites. In addition to that, these extreme machine learning models had superiority with respect to computational efficiency. Figure 5.13 shows the changes in running time and performance of these three models when tunning one built-in hyperparameter. Using Figure 5.13(a) as an example, the vertical axis includes the RMSE values and running time; the horizontal axis is the amount of hidden neurons (nhid). The green dashed line indicates the location of the lowest RMSE and the corresponding optimal hyperparameter. As shown in Figure 5.13(a), the RMSE value drops as the number of hidden neurons increases. When the RMSE reached the minimum value (2.87), the optimal number of hidden neurons was 22; meanwhile, the running time was only 0.36 s. Likewise, for the GMM-RFE-ERT and GMM-RFE-XGBoost models, the optimal hyperparameters are displayed in Figure 5.13(b)-(c). The running times to build these two models were less than one second (0.61 s and 0.89 s) when tuning one built-in hyperparameter. Moreover, the ELM, ERT, and XGBoost algorithms usually contain multiple hyperparameters that need to be tuned. Consequently, the total running time for each GMM-based model is listed in Table 5.4. For these three prediction models, the running times were 3.2 s, 183.0 s, and 314.4 s, respectively, in the order of GMM-RFE-ELM < GMM-RFE-ERT < GMM-RFE-XGBoost. Compared to these extreme models, the other commonly used models often need more computational cost. For example, the ANN models, such as the Bayesian regularized neural network (BRNN) and BPNN, took a long time (14,004.0 s and 9,072.0 s) in our previous study (Fan et al., 2023d). Therefore, the extreme machine learning models were more computationally efficient for the rapid estimation of truck cycle time. The main reason for the fast assessment is the random initialization of weights between the layers in ELM (Huang et al., 2006), random selections of features for each tree and splitting thresholds in ERT (Zhang et al., 2023), and parallel processing and approximate greedy algorithm for the best split in XGBoost (Qiu et al., 2022). The same results were found in the research by Sekhar Roy et al. (2018), Chencho et al. (2022), and Liu et al. (2018). They all demonstrated that EML, ERT, and XGBoost provided rapid estimations. For instance, Chencho et al. (2022) constructed an ERT model for single-element damage quantification of civil engineering structures. The ERT model showed powerful performance in computation (the training time was 13.42 s) and prediction (e.g.,

R^2 = 99.9%). To sum up, this study proposed three computationally efficient models that will facilitate mining engineers to rapidly estimate truck cycle time and make better decisions.

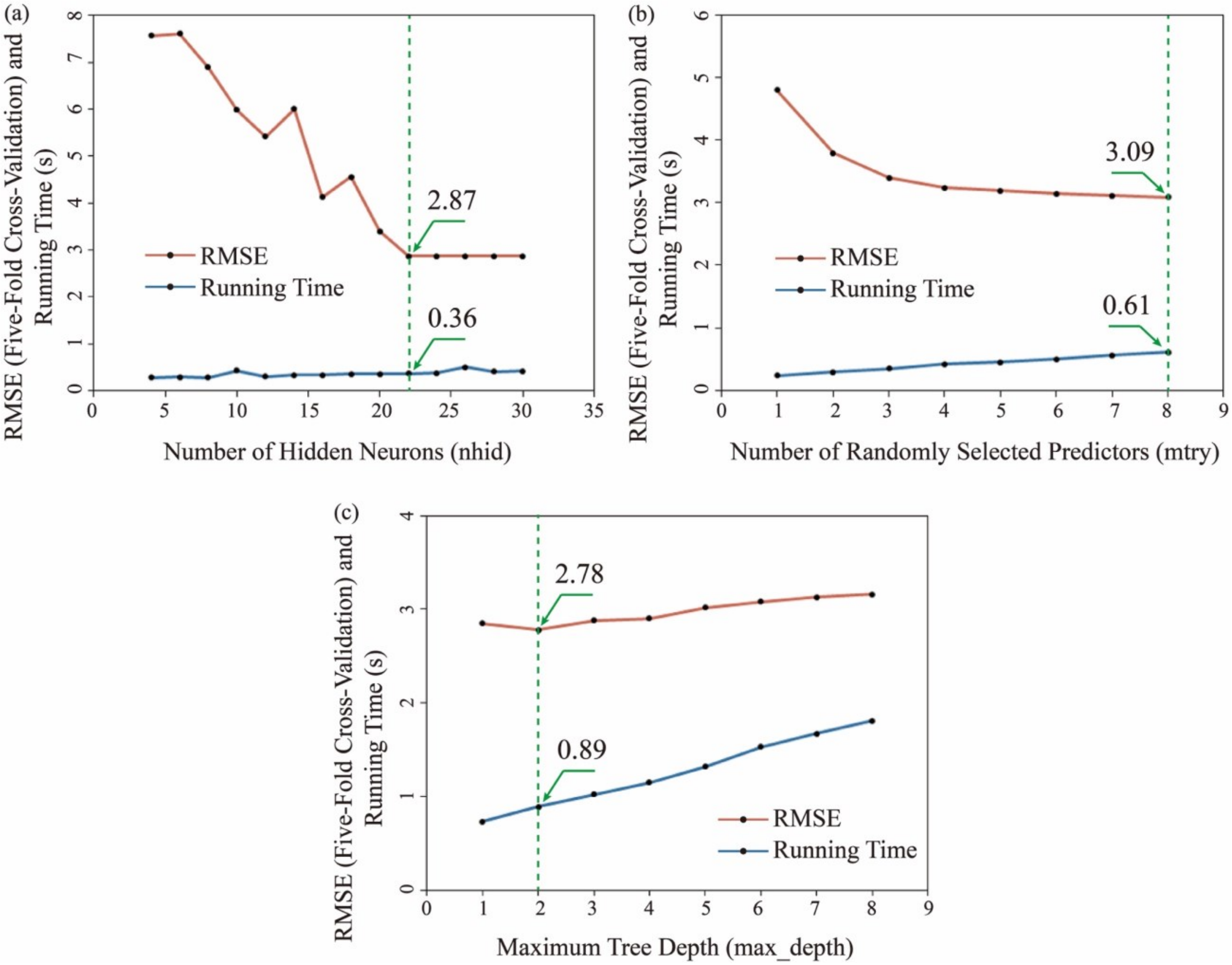


Figure 5.13 Relationships of model performance (RMSE), hyperparameter tuning, and running time of (a) GMM-RFE-ELM, (b) GMM-RFE-ERT, and (c) GMM-RFE-XGBoost models.

Table 5.4 Running times of machine learning models between this and previous studies.

Algorithm	Model	Input Variable	Running Time (s)	Reference
Bayesian regularized neural network (BRNN)	BRNN	Haul distance, empty speed, ambient temperature, waiting at shovel	14,004.0	Fan et al. (2023d)
Backpropagation neural network (BPNN)	BPNN		9,072.0	
Random forest (RF)	RF		2,726.4	
Gradient boosting regression (GBR)	GBR		1,998.6	
Extreme learning machine (ELM)	GMM-RFE-ELM	Haul distance, empty speed, waiting at shovel, month, destination, ambient temperature, precipitation	3.2	This study
Extremely randomized tree (ERT)	GMM-RFE-ERT		183.0	
Extreme gradient boosting (XGBoost)	GMM-RFE-XGBoost		314.4	

5.4. Conclusions

Extreme machine learning is known for handling massive data and providing fast computations, such as XGBoost, ELM, and ERT. For the first time, this study applied these extreme machine learning algorithms to create prediction models of mining truck cycle time based on massive and multidimensional data from mine sites. Moreover, this study investigated and compared the effects of three feature selection methods (DT, ANOVA, and RFE) and two unsupervised clustering techniques (K-means and GMM) on model performance. The main findings are concluded below:

(1) RFE (the wrapped method) was the best feature selection method that improved the model performance and removed redundant variables. For example, the RFE-XGBoost model selected seven input variables from the original dataset, which achieved the lowest RMSE

(5.12), MAE (3.78), and the highest R^2 (33.63%) compared to the ANOVA-XGBoost (5.15, 3.79, and 32.90%), DT-XGBoost (5.13, 3.79, 33.27%), and XGBoost (5.13, 3.79, and 33.37%) models.

(2) Among the selected inputs, haul distance, empty speed, and waiting at shovel were the most critical variables affecting truck cycle time. The variable importance ranking presents here: haul distance (32.60%) > empty speed (21.15%) > waiting at shovel (20.00%) > month (9.04%) > destination (7.36%) > ambient temperature (6.62%) > precipitation (3.23%). These scores help engineers gain insight into the main factors influencing truck cycle time at mine sites, leading to more accurate predictions of truck cycle time and the estimation of overall mine productivity.

(3) GMM performed better than K-means and significantly improved the models' accuracy. For instance, the GMM-RFE-XGBoost model had the highest R^2, with values of 80.37%. Its performance was considerably better than the RFE-XGBoost (33.63%) and KM-RFE-XGBoost (33.06%) models.

(4) The XGBoost algorithm outperformed the ELM and ERT algorithms in predicting truck cycle time. For example, the R^2 of the GMM-RFE-XGBoost model was 80.37%, which was higher than that of the GMM-RFE-ELM (79.10%) and GMM-RFE-ERT (75.81%).

(5) The extreme hybrid models provided rapid estimations of mining truck cycle time. For the GMM-based models, the runtimes were 3.2 s, 183.0 s, and 314.4 s for GMM-RFE-ELM, GMM-RFE-ERT, and GMM-RFE-XGBoost, respectively. Among them, the GMM-RFE-ELM had the shortest running time without decreasing significant accuracy. This will facilitate engineers to estimate truck cycle time and make decisions rapidly.

Chapter 6. Interpretable data-driven models for assessing truck productivity in open-pit mining under rea-site weather conditions with varying temporal resolutions

This chapter has been submitted for peer review as **C. Fan**, C.B. Arachchilage, N. Zhang, B. Jiang, W.V. Liu, Interpretable data-driven models for assessing truck productivity in open-pit mining under real-site weather conditions with varying temporal resolutions, *International Journal of Mining, Reclamation and Environment*. © Taylor & Francis. (With editor)

Nomenclatures

AdaBoost	Adaptive boosting
DT	Decision tree
GBR	Gradient boosting regression
GUI	Graphical user interface
MAE	Mean absolute error
MANOBS	Manual of Surface Weather Observation Standards
MLR	Multiple linear regression
n	The *n*th prediction
N	The number of total predictions
OSM	Oil sands magazine
PDP	Partial dependence plot
R^2	Coefficient of determination
RAM	Random access memory
RF	Random forest
RMSE	Root mean square error
SHAP	SHapley Additive exPlanation

SVR	Support vector regression
tph	Tonnes per hour
XGBoost	Extreme gradient boosting
y	Output variable
$\bar{y}$	Mean value of y
$\hat{y}$	Predicted value of y

6.1. Introduction

In open-pit mining, off-the-road truck haulage plays a crucial role in facilitating the movement of bulk materials (e.g., ore, waste, and overburden) from mining faces to designated locations within mine sites (Ma et al., 2023). The productivity of truck haulage (or truck productivity) is a measure of the amount of ores (unit: tonnes) that can be moved by a mining truck in a given period of time (unit: hours), which is a critical determinant of the overall productivity and cost-effectiveness of mining operations (Chanda & Gardiner, 2010; Fan et al., 2022).

To assess truck productivity, researchers have initiated data-driven modeling to build the relationship between truck productivity and vital influencing parameters (Fan et al., 2023b, 2023d; Sun et al., 2018). These parameters usually refer to the variables associated with truck haulage conditions, such as haul distance, truck types, truck speed, payload, and number of allocated trucks (Cervantes et al., 2019; Choi et al., 2022; Fan et al., 2023a). In addition to the truck haulage conditions, variables related to real-site weather conditions are seen as essential parameters, such as ambient temperature, precipitation, relative humidity, and wind speed (Medinac et al., 2020; Sagberg et al., 2015). These weather-related variables affect truck productivity by influencing truck cycle time at mine sites (Fan et al., 2023b). For example, Ma et al. (2023) reported that a rise in ambient temperature (e.g., from 20 ℃ to 40 ℃) induced an increase in truck tire temperature (e.g., from 54 ℃ to 69 ℃), which caused tire fatigue damage and affected truck cycle time. Likewise, Asamer and Reinthaler (2010) statistically analyzed the data from U.S. highway administrations. They demonstrated that adverse weather conditions (e.g., heavy precipitation) led to a 35% reduction in running speed, thus increasing travel time. Similarly, relative humidity and wind speed may interfere with road conditions (e.g., wetness or dryness) and driver’s vision (Choi & Nieto, 2011; Silion & Foşalău, 2014), influencing driving habits and travel time. With global warming (Chong et al., 2023; Fan et al., 2019) and more frequent extreme weather (Bag et al., 2022), the impacts of weather-related variables have become increasingly pronounced. For instance, according to Environmental Canada (MEP, 2023), the highest ambient temperature recorded in the Fort McMurray region, where open-pit mines are widely distributed in Northern Alberta, jumped from 33.2 °C in 2016 to 40.5 °C in 2021.

Currently, these weather-related variables have been involved in building prediction models of truck productivity and have contributed to the model output (Fan et al., 2022, 2023b, 2023d; Sun et al., 2018). For example, Sun et al. (2018) built regression models of truck cycle time (equivalent to truck productivity) incorporating ambient temperature, relative humidity, and precipitation. The research presented that the proposed model achieved a maximum R^2 (coefficient of determination) of 89%, and the model accuracy was improved by 5.13% with these weather-related variables. In our previous studies (Fan et al., 2022, 2023b, 2023d), ambient temperature and precipitation were also included to establish prediction models of truck productivity. The results reported that the prediction models' R^2 reached 75%-87%, and these two weather variables contributed more than 15% to the model output. However, the temporal scales (or resolutions) of real-site weather conditions were not taken into account in these studies. For instance, the maximum precipitation over a week (e.g., 85.30 mm in this study) can have a more substantial impact on road conditions and driving habits than an hour (e.g., 14.10 mm in this study) (Xing et al., 2019). There is a notable research gap in considering the real-site weather conditions with varying temporal resolutions in predicting mine truck productivity.

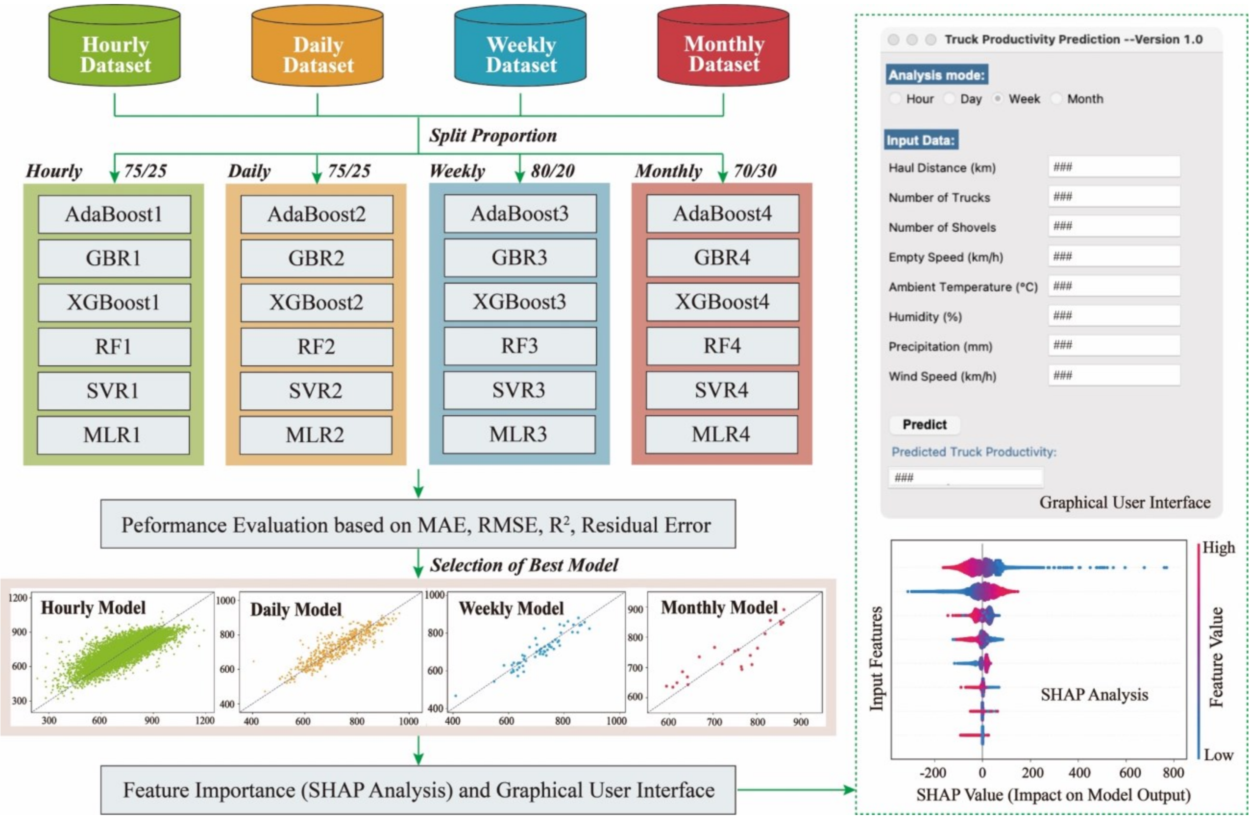
To this end, the purpose of this research was to construct truck productivity prediction models incorporating real-site weather conditions with varying temporal resolutions. Four datasets (including the hourly, daily, weekly, and monthly datasets) were prepared after processing massive raw data from operating open-pit mine sites. With these four datasets, six extensively applied and efficient machine learning algorithms were then adopted to train prediction models (Arachchilage et al., 2023), including support vector regression (SVR), random forest (RF), adaptive boosting (AdaBoost), gradient boosting regression (GBR), extreme gradient boosting (XGBoost), and multiple linear regression (MLR). After that, this study employed SHapley Additive exPlanations (SHAP) to interpret the best prediction models based on the four datasets since SHAP can provide the qualitative and quantitative analysis of each input variable's effect on model outputs (Lu et al., 2021). Meanwhile, a unified graphical user interface (GUI) was designed and developed for end users.

The novelty of this study consists of the following four aspects. First, this study innovatively explored the effect of temporal resolutions on establishing truck productivity prediction models. Second, this study is the first to combine SHAP and machine learning models to decipher the influence of input variables on truck productivity at varying temporal resolutions. Third, for the first time, this research analyzed the influence of extreme weather on truck productivity and truck-shovel allocation at mine sites. Fourth, a simple and easy-to-use GUI was first developed to estimate hourly, daily, weekly, and monthly truck productivity. This study contributes solutions for predicting truck productivity at varying temporal resolutions and a unified GUI for mining engineers and researchers to make decisions more easily and quickly.

6.2. Methodology

6.2.1. Overall methodological framework

Figure 6.1 illustrates the general framework of the methodology. Four datasets with varying temporal resolutions, including hourly, daily, weekly, and monthly, were used to construct truck productivity prediction models in open-pit mining. Before the models were built, due to its fast computation (Fan et al., 2022), MLR was used to evaluate the effect of the three split proportions (70/30, 75/25, and 80/20) on model performance and to select the best-split ratio for training/testing datasets at each temporal resolution. Next, six machine learning algorithms were utilized to build models based on the four training datasets, including AdaBoost, GBR, XGBoost, RF, SVR, and MLR. The models' performance was evaluated by the four indicators recommended by Arachchilage et al. (2023): root mean square error (RMSE), mean absolute error (MAE), R^2, and residual error. After that, the four best prediction models were selected to forecast truck productivity at hourly, daily, weekly, and monthly resolutions. Based on these four models, SHAP, a widely used approach for model interpretation (Djandja et al., 2023), was adopted to analyze how input variables affect the model output and rank these input variables' importance. Moreover, a simple and easy-to-use GUI was provided for mining engineers to assess truck productivity at varying temporal resolutions. The models and GUI were developed on a personal computer (a 16.0 GB of RAM and an Intel Core i7-12700K processor) by applying Scikit-learn (version 1.2.2) and Tkinter packages based on a Python language programming environment (version 3.10.9).



Analysis mode:
○ Hour ○ Day ● Week ○ Month

Input Data:

Haul Distance (km)	###
Number of Trucks	###
Number of Shovels	###
Empty Speed (km/h)	###
Ambient Temperature (°C)	###
Humidity (%)	###
Precipitation (mm)	###
Wind Speed (km/h)	###

Predicted Truck Productivity: ###

Figure 6.1 A schematic representation of the study framework for evaluating mining truck productivity. ("###": the input information is not disclosed as it is the proprietary property of mining companies.)

6.2.2. Machine learning algorithms

This study adopted six machine learning algorithms (AdaBoost, GBR, XGBoost, RF, SVR, and MLR) to train prediction models because these algorithms have been extensively used in various mining applications, such as ore production prediction (Choi et al., 2022), concrete material design (Bangaru et al., 2019), and mining equipment maintenance (Zhang et al., 2022). The principles of these algorithms are briefly introduced as follows:

- RF is an ensemble learning algorithm combining a large amount of single decision trees (DT) to gain higher performance (Fan et al., 2023b). RF mainly adopts a bagging technique to generate numerous DTs, as shown in Figure 6.2(a). Bagging randomly selects a series of subsets and a fraction of input variables from the original data to train DTs. These DTs are trained in a parallel manner and provide predictions for the target variables simultaneously. The average prediction of all DTs is the final prediction of the RF model.

- AdaBoost, GBR, and XGBoost are also tree-based ensemble learning methods integrating numerous DTs, as shown in Figure 6.2(b). Unlike RF, these three algorithms utilize a boosting rather than a bagging technique (Fan et al., 2023b). In boosting, each DT is generated sequentially and focuses on the data points incorrectly predicted by the previous DT. AdaBoost assigns higher weights to these data points and prioritizes them in the following DT. The DTs in GBR have the same training process as AdaBoost, but GBR uses a gradient descent algorithm to find the best direction for each iteration. XGBoost is an optimization of GBR that combines additional features, such as regularization techniques and parallel computing, to improve model performance, enhance computational efficiency, and avoid overfitting (Fan et al., 2023d).

- SVR is a well-known single-learning algorithm for regression problems (Arachchilage et al., 2023). As shown in Figure 6.3, SVR creates the optimal hyperplane by maximizing the margin to split the space of input variables and make sure the shortest vertical distance between data points and the hyperplane. The data points closely distributed along the margins are known as support vectors. SVR achieves this by using kernel functions to map the input variables to a higher dimensional space. The radial basis function is the most widely used kernel to capture nonlinear relationships (Onyekwena et al., 2022).

- MLR is also a single learning algorithm that presumes a linear relationship between an individual output and multiple input variables (Fan et al., 2022). A best-fitted line can describe this linear relationship, which is defined to minimize the sum of square errors of each data point from the line.

For the nonlinear algorithms (AdaBoost, GBR, XGBoost, RF, and SVR), built-in hyperparameters need to be adjusted to achieve higher prediction accuracy and control model complexity (Fan et al., 2023b; Xue et al., 2021). A sequential model-based optimization (SMBO) algorithm was adopted to search the optimal hyperparameters in the modeling because SMBO has shown effective optimization results and is computationally inexpensive to the previous study (Bo et al., 2022). The underlying principle of SMBO is to search the pre-defined hyperparameters space sequentially by constructing a surrogate model (e.g., Gaussian process) of the objective function. This search process is usually combined with the five-fold cross-validation (in Figure 6.4), which splits the training dataset into five folds and iteratively uses four of them as new training datasets to train models with a set of hyperparameters. This procedure is iterated five times until each remaining fold is taken as the testing dataset. RMSE is used as an indicator to evaluate the model performance for each set of hyperparameters (Fan et al., 2023b).

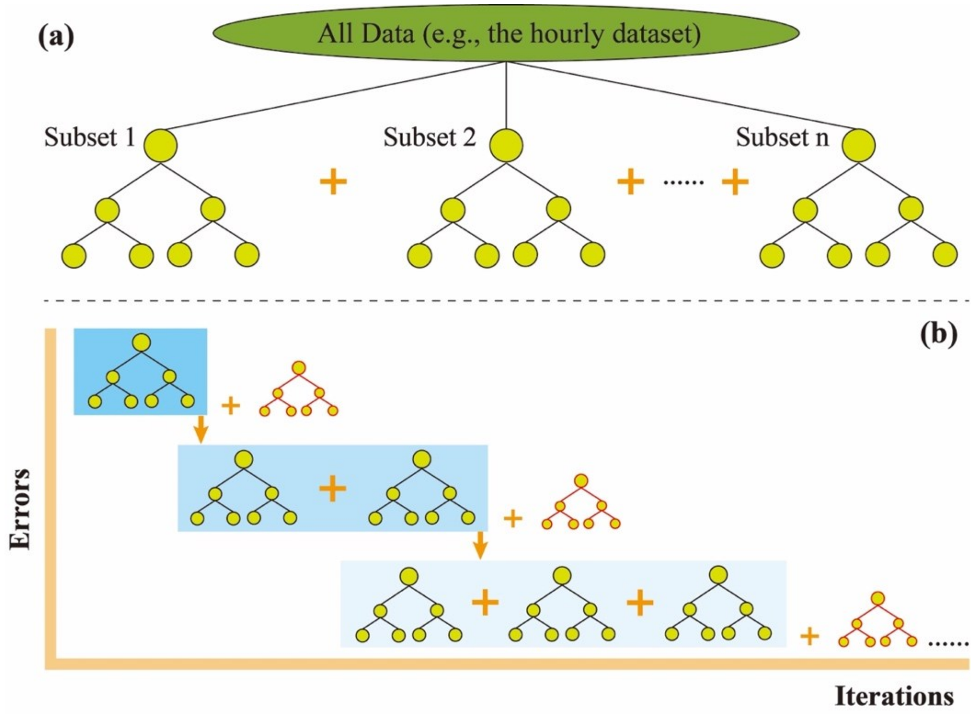


Figure 6.2 The basic ideas of two tree-based algorithms: (a) bagging tree and (b) boosted trees.

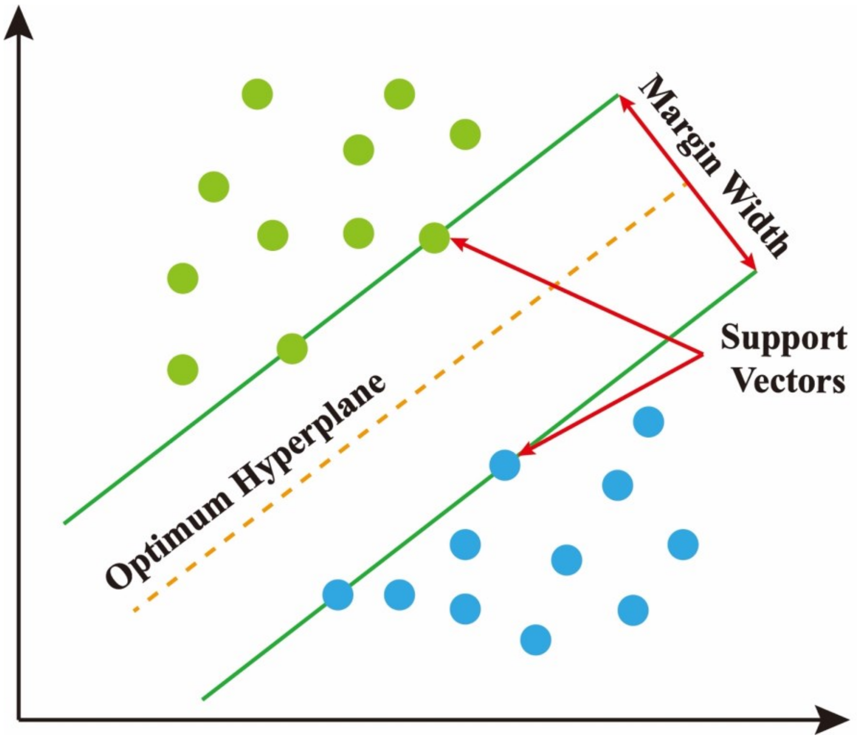


Figure 6.3 The basic idea of the SVR algorithm.

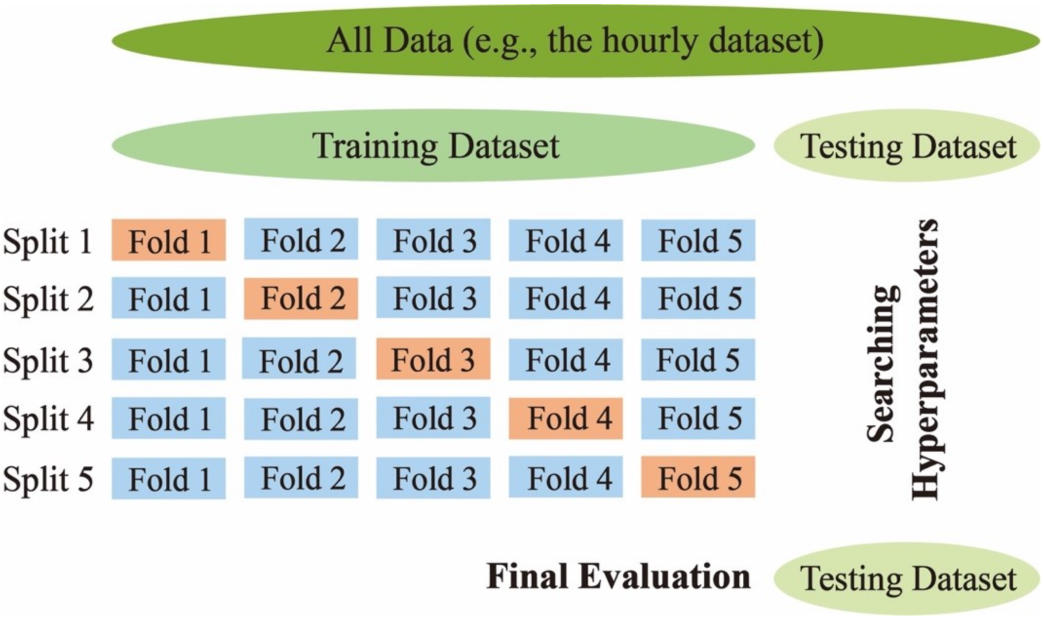


Figure 6.4 The visualization for the five-fold cross-validation.

6.2.3. Performance indicators for evaluating prediction models

To evaluate and compare the accuracy of prediction models, the four extensively applied indicators were adopted because they are easy to implement and calculate and give an intuitive indication of the errors between the observed values (y_n) and predicted values ($\hat{y}$) (Wu et al., 2020). These indicators are MAE, RMSE, R^2, and residual error, which can be computed based on the following equations:

$$MAE = \frac{1}{N}\sum_{n=1}^{N}|y_n - \hat{y}_n| \tag{6-1}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(y_n - \hat{y}_n)^2} \tag{6-2}$$

$$R^2 = 1 - \frac{\sum_n^N (y_n - \hat{y}_n)^2}{\sum_n^N (y_n - \bar{y}_n)^2} \tag{6-3}$$

$$Residual\ error = y_n - \hat{y}_n \tag{6-4}$$

where n indicates the nth prediction, N is the total number of predictions, and $\bar{y}_n$ is the mean of observed values. A regression model with a larger R^2 and smaller MAE, RMSE, and residual error represents a better prediction performance (Arachchilage et al., 2023).

6.2.4. Feature importance analysis method

To make the prediction models more transparent, the SHAP method was employed in this study to interpret these models by a qualitative analysis of the correlation between input variables and the output and a quantitative analysis of feature importance (Djandja et al., 2023; Yiu et al., 2022). SHAP is a game theory-based approach developed by Lundberg and Lee (2017). Each data point of an input variable is treated as a player in a game, in which the profit is the model output (i.e., the prediction). Based on this prediction, a unique SHAP value is offered to the input variable for that data point, which shows the deviation from the average forecast of all data points (Lu et al., 2021). The SHAP method can quantify feature importance by averaging the absolute SHAP values of each input variable. This method combines the individual contribution of all data points for an input variable to obtain a holistic understanding of this input's importance to the model output (Eker et al., 2021).

6.3. Data Description and Characterization

Six-year (2016-2021) truck haulage data were collected from Alberta's operating oil sands mines. There were 1,625,590 individual truck cycles after erroneous and blank records were removed. It is challenging to handle such a massive amount of data; therefore, this study averaged six years of data at hourly, daily, weekly, and monthly resolutions to obtain four new datasets. This reduces computational costs and enables the investigation of input-output relationships at varying temporal resolutions, providing a unique insight into mining planning and decision-making. The variables and their statistical information of these four new datasets are presented in Tables 6.1-6.4. Each dataset had eight input variables. Using Table 6.1 as an example, the input variables were haul distance (x_1), empty speed (x_2), number of trucks (x_3), number of shovels (x_4), ambient temperature (x_5), humidity (x_6), precipitation (x_7), and wind speed (x_8). These input variables were selected based on the site experience, which were correlated with truck cycle time (Chanda & Gardiner, 2010; Fan et al., 2023b). The distribution characteristics of these nine variables were presented by four statistics, including maximum, minimum, mean, and standard deviation. Notably, the amount of data points in these four datasets varied significantly. The hourly dataset contained the most data points (47,777), much more than the daily (2,028), weekly (302), and monthly (71) datasets. From hourly to monthly, the ranges for all variables except precipitation became narrower due to data aggregation (Bodesheim et al., 2018). This can also be observed from the boxplots of input variables in Figure 6.5. For example, wind speed fell between 0.30 km/h to 145.90 km/h in the hourly data, while its range narrowed to 6.00 km/h to 16.30 km/h in the monthly data. Before the models were trained, all variables were scaled between zero to one to ensure that they were equal in relevance (Arachchilage et al., 2023).

Table 6.1 Input variables in the hourly data (47,777 data points in total) with statistical information.

Variable	Unit	Maximum	Minimum	Mean	Standard Deviation
Haul Distance (x_1)	km	15.62	1.00	4.63	0.78
Empty Speed (x_2)	km/h	60.00	8.45	34.30	5.30
Number of Trucks (x_3)	-	###	###	###	###
Number of Shovels (x_4)	-	###	###	###	###
Ambient Temperature (x_5)	°C	40.20	-39.70	1.26	14.40
Humidity (x_6)	%	96.00	12.00	67.77	17.48
Precipitation (x_7)	mm	14.10	0.00	0.04	0.31
Wind Speed (x_8)	km/h	145.90	0.30	10.62	6.44

(“###”: the input information is not disclosed as it is the proprietary property of mining companies.)

Table 6.2 Input variables in the daily data (2,028 data points in total) with statistical information.

Variable	Unit	Maximum	Minimum	Mean	Standard Deviation
Haul Distance (x_1)	km	7.25	1.96	4.62	0.71
Empty Speed (x_2)	km/h	46.84	15.22	34.29	4.63
Number of Trucks (x_3)	-	###	###	###	###
Number of Shovels (x_4)	-	###	###	###	###
Ambient Temperature (x_5)	°C	32.20	-34.90	1.23	14.01
Humidity (x_6)	%	94.10	22.90	67.73	13.30
Precipitation (x_7)	mm	47.90	0.00	1.04	3.34
Wind Speed (x_8)	km/h	29.20	2.00	10.65	4.34

(“###”: the input information is not disclosed as it is the proprietary property of mining companies.)

Table 6.3 Input variables in the weekly data (302 data points in total) with statistical information.

Variable	Unit	Maximum	Minimum	Mean	Standard Deviation
Haul Distance (x_1)	km	6.33	2.38	4.62	0.65
Empty Speed (x_2)	km/h	43.30	18.33	34.22	4.39
Number of Trucks (x_3)	-	###	###	###	###
Number of Shovels (x_4)	-	###	###	###	###
Ambient Temperature (x_5)	°C	25.90	-30.60	1.26	13.61
Humidity (x_6)	%	90.00	35.80	67.77	10.94
Precipitation (x_7)	mm	85.30	0.0	6.95	10.28
Wind Speed (x_8)	km/h	20.20	3.90	10.64	2.76

("###": the input information is not disclosed as it is the proprietary property of mining companies.)

Table 6.4 Input and output variables in the monthly data (71 data points in total) with statistical information.

Variable	Unit	Maximum	Minimum	Mean	Standard Deviation
Haul Distance (x_1)	km	5.83	3.44	4.60	0.57
Empty Speed (x_2)	km/h	41.59	23.39	34.07	3.97
Number of Trucks (x_3)	-	###	###	###	###
Number of Shovels (x_4)	-	###	###	###	###
Ambient Temperature (x_5)	℃	19.50	-21.50	1.79	12.75
Humidity (x_6)	%	82.10	43.60	67.25	9.09
Precipitation (x_7)	mm	141.70	0.50	33.74	31.89
Wind Speed (x_8)	km/h	16.30	6.00	10.68	2.02

("###": the input information is not disclosed as it is the proprietary property of mining companies.)

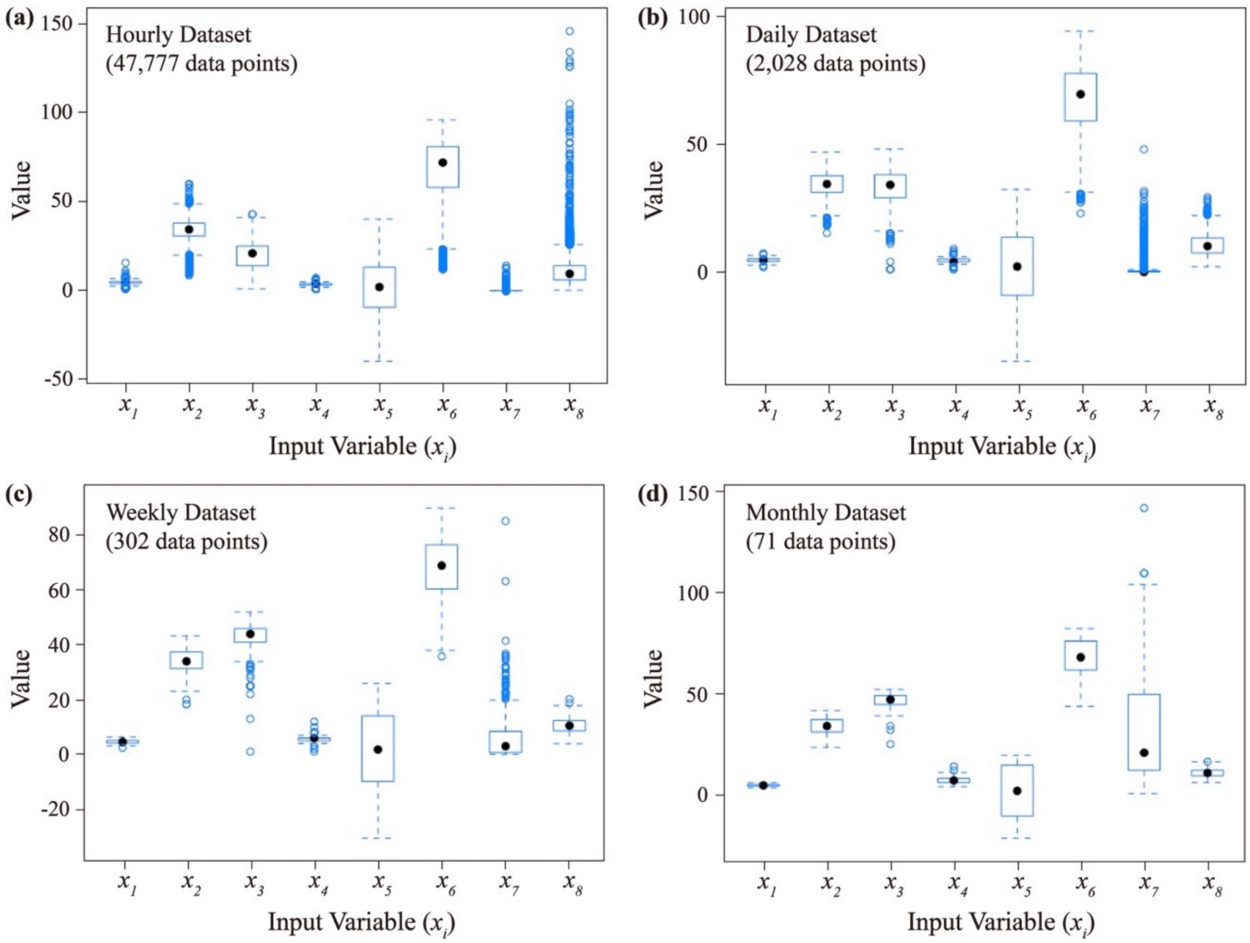


Figure 6.5 Boxplots of input variables in the (a) hourly, (b) daily, (c) weekly, and (d) monthly datasets.

6.4. Results and Discussion

6.4.1. Selection of best-split ratio for training and testing datasets

The best-split ratios for four temporal-resolutions data (i.e., hourly, daily, weekly, and monthly) were investigated before building prediction models of truck productivity. Three widely applied ratios were chosen for splitting the data into training and testing datasets, including 70/30, 75/25, and 80/20, which were suggested by Bui et al. (2020) and Goel et al. (2022). Table 6.5 lists the prediction performance of the MLR models in each data resolution based on these three split ratios. MLR was employed for modeling due to its advantage of fast computation and less overfitting

(Fan et al., 2022). From Table 6.5, using the hourly data as an example, the model had the lowest MAE (60.52) and RMSE (79.12) when the split proportion was 75/25 (i.e., 75% and 25% for the training dataset and testing dataset). The R^2 of the model was 0.56 at both the 75/25 and 70/30 ratios, but still slightly higher than the R^2 at the 80/20 ratio (0.55). This indicates that the split ratio had a certain impact on model performance (Ewees et al., 2020), and 75/25 was the best ratio for partitioning the hourly data. Similarly, according to the model performance evaluation in Table 6.5, the best-split ratios for the daily, weekly, and monthly data were 75/25, 80/20, and 70/30, respectively. This study is analogous to the previous studies by Fan et al. (2023c) and Nguyen et al. (2021). Both studies investigated the effect of split ratio on model performance. The former constructed MLR models based on four different split ratios to predict ore production in oil sands mining. The results reported that the model showed the highest accuracy (e.g., $R^2 = 91.87\%$) when the ratio was 80/20. The latter designed nine proportions (the proportion of the training dataset was added from 10% to 90%) and demonstrated that 70/30 was the best-split ratio by building and evaluating prediction models for soil shear strength in civil constructions. Overall, 75/25, 75/25, 80/20, and 70/30 were the best-split ratios for hourly, daily, weekly, and monthly data for the purpose of building more complex truck productivity prediction models, which will be discussed in Section 6.4.2.

Table 6.5 Selection of best-split ratio for datasets with varying temporal resolutions based on multiple linear regression.

Dataset	Split Ratio	Performance Evaluation on Testing Dataset		
		MAE	RMSE	R^2
Hourly	70/30	60.76	79.37	0.56
	75/25	**60.52**	**79.12**	**0.56**
	80/20	60.75	79.63	0.55
Daily	70/30	39.58	50.85	0.73
	75/25	**39.58**	**50.51**	**0.73**
	80/20	39.67	50.57	0.72
Weekly	70/30	30.89	41.26	0.77
	75/25	30.00	36.15	0.82
	80/20	**28.61**	**34.29**	**0.85**
Monthly	**70/30**	**32.54**	**40.91**	**0.79**
	75/25	36.83	44.40	0.75
	80/20	37.78	46.27	0.73

6.4.2. Selection of best model at varying temporal resolutions

Section 6.4.1 selected the four best training/testing ratios for splitting hourly, daily, weekly, and monthly data. With these split ratios, five more complex models were constructed for each temporal resolution to attain potentially higher prediction accuracy, including AdaBoost, GBR, XGBoost, RF, and SVR. MLR, as a simple linear model, was also employed to be compared with these five nonlinear models. Table 6.6 lists the performance evaluation and comparison of six ML models based on training and testing datasets at varying temporal resolutions.

As presented in Table 6.6, for the testing dataset, the GBR model showed the highest R^2 (0.65) and the smallest MAE (53.49) and RMSE (70.15) among the six models built on the hourly data. Therefore, GBR was the best model for predicting hourly truck productivity. In particular, GBR (RMSE = 70.15) was superior to MLR (RMSE = 79.12), indicating a complex nonlinear relationship between the hourly truck productivity and its input variables (Fan et al., 2023b). This applies to the models built on the daily data: all the nonlinear models outperformed MLR in forecasting daily truck productivity. For example, the RMSE of AdaBoost, GBR, XGBoost, RF, and SVR was 50.10, 49.15, 47.30, 48.57, and 45.50, while that of MLR was 50.51. Among them, SVR was the best model for forecasting daily truck productivity due to its highest R^2 (0.78) and lowest MAE (34.29) and RMSE (45.50). Moreover, for the models established on the weekly data, SVR (RMSE = 33.83) outperformed MLR (RMSE = 34.29) slightly, but all the tree-based models (e.g., GBR with RMSE = 46.92) underperformed MLR. This suggested that the relationship between the weekly truck productivity and its input variables was more nearly linear (Fan et al., 2023d). This was more pronounced in the models trained on the monthly data. Among the six models, the five nonlinear models showed overfitting problems; these models performed much better on the training dataset than on the testing dataset (Tien Bui et al., 2019). For instance, the MAE, RMSE, and R^2 of SVR were 17.01, 23.25, and 0.91 on the monthly training dataset; conversely, these performance indicators were 35.00, 41.59, and 0.78 on the monthly testing dataset. Compared with these nonlinear models, MLR performed relatively well, with less overfitting in estimating monthly truck productivity. As a result, MLR was selected as the best model for the monthly data. From the hourly to the monthly resolutions, it can be observed that the nonlinear relationship between truck productivity and its input variables progressively diminished. This can be attributed to two reasons: (1) The aggregation (e.g., the averaged data at varying temporal resolutions in this study) of hourly data into monthly data reduces variability and noise, thus smoothing out short-term fluctuations in hourly data (Bodesheim et al., 2018). For example, from the hourly to monthly data, the standard deviations of almost all variables decreased (e.g., the humidity dropped from 17.48% to 9.09%), as shown in Tables 6.1-6.4. (2) The weather-related input variables exhibit seasonal and periodic patterns, which may contribute to more linear behavior at the monthly level (Nourani et al., 2019). For instance, ambient temperature and precipitation in Northern Alberta have a significant increase from May to August each year (Ma et al., 2023; MEP, 2023). Similar results were reported by Yuval and Hsieh (2002), who found that the RMSE values for the nonlinear and linear models were 0.93 and 1.06 at the daily resolution, 0.44 and 0.48 at the weekly resolution, and 0.24 and 0.25 at the monthly resolution when simulating precipitation on the British Columbia coast. From the daily to monthly resolutions, the performance of the two models became closer, indicating that the input-output relationship degenerated from a nonlinear to linear regression.

Furthermore, Table 6.7 lists the optimal hyperparameters for these best models (hourly-GBR, daily-SVR, and weekly-SVR) except MLR (no hyperparameters). Also, Figure 6.6 displays the scatter plots and residual errors of the predicted results (vertical) and actual values (horizontal) from these best models at varying temporal resolutions. As shown in Figure 6.6, the weekly-SVR model had a higher R^2 (0.85) than the hourly-GBR (0.65), daily-SVR (0.78), and weekly-SVR (0.79) models. This implies that mining engineers can make more accurate estimates of truck productivity at the weekly resolution compared with other resolutions, leading to more rational decision-making and planning in the week-to-week operations at mine sites. This is comparable to Ma et al. (2019) and Singh and Yassine (2018). Ma et al. (2019) built eight prediction models at three temporal resolutions to improve the prediction accuracy of air quality. The results observed that the proposed model built at the weekly (RMSE = 7.23) level was more accurate than those at the hourly (RMSE = 8.04) and daily (RMSE = 9.69) levels due to the reduction in the data variance. Closely, Singh and Yassine (2018) explored the influence of temporal resolutions on household energy consumption forecasting. They found that the proposed model achieved the highest accuracy at the hourly (81.89%) data instead of the daily (75.88%), weekly (79.23%), and monthly (74.74%) data. To encapsulate, GBR, SVR, SVR, and MLR were chosen as the best prediction models for hourly, daily, weekly, and monthly resolutions. These models were also the basis for the SHAP analysis (impact of input variables on model output, one- and two-way partial dependence plots, and feature importance), which will be discussed in Sections 6.4.3 and 6.4.4.

Table 6.6 Performance evaluation and comparison of ML models based on the training and testing datasets.

Data	Model	Performance Evaluation						Time (s)
		MAE		RMSE		R^2		
		Train	Test	Train	Test	Train	Test	
Hourly	AdaBoost	54.22	56.94	67.45	73.61	0.68	0.62	865.25
	GBR	**50.53**	**53.49**	**65.32**	**70.15**	**0.70**	**0.65**	**1,434.07**
	XGBoost	50.70	54.19	65.67	70.93	0.70	0.64	326.52
	RF	51.89	55.40	67.84	72.86	0.68	0.62	2,565.19
	SVR	54.43	54.61	72.16	71.83	0.64	0.63	8,580.23
	MLR	60.96	60.52	79.63	79.12	0.56	0.56	0.02
Daily	AdaBoost	40.44	39.68	49.91	50.10	0.75	0.73	100.33
	GBR	37.11	38.69	47.16	49.15	0.78	0.74	76.45
	XGBoost	34.45	37.18	44.19	47.30	0.80	0.76	47.49
	RF	36.04	37.38	46.15	48.57	0.79	0.75	162.02
	SVR	**32.19**	**34.29**	**43.74**	**45.50**	**0.81**	**0.78**	**235.76**
	MLR	41.22	39.58	53.08	50.51	0.72	0.73	0.01
Weekly	AdaBoost	31.24	32.58	36.37	42.50	0.85	0.77	55.67
	GBR	33.93	36.65	43.35	46.92	0.81	0.76	88.40
	XGBoost	29.11	32.19	40.85	44.85	0.81	0.74	35.97
	RF	31.97	31.45	40.83	44.27	0.81	0.75	43.44
	SVR	**25.13**	**26.59**	**35.39**	**33.83**	**0.86**	**0.85**	**124.24**
	MLR	34.32	28.61	43.65	34.29	0.78	0.85	<0.01
Monthly	AdaBoost	10.89	44.50	13.72	54.61	0.97	0.62	48.71
	GBR	2.98	36.11	4.67	47.98	0.99	0.71	28.72
	XGBoost	6.53	40.61	9.12	49.58	0.99	0.69	34.65
	RF	16.41	44.79	19.90	53.98	0.94	0.63	34.36
	SVR	17.04	35.00	23.25	41.59	0.91	0.78	44.76
	MLR	**21.85**	**32.54**	**27.04**	**40.91**	**0.88**	**0.79**	**<0.01**

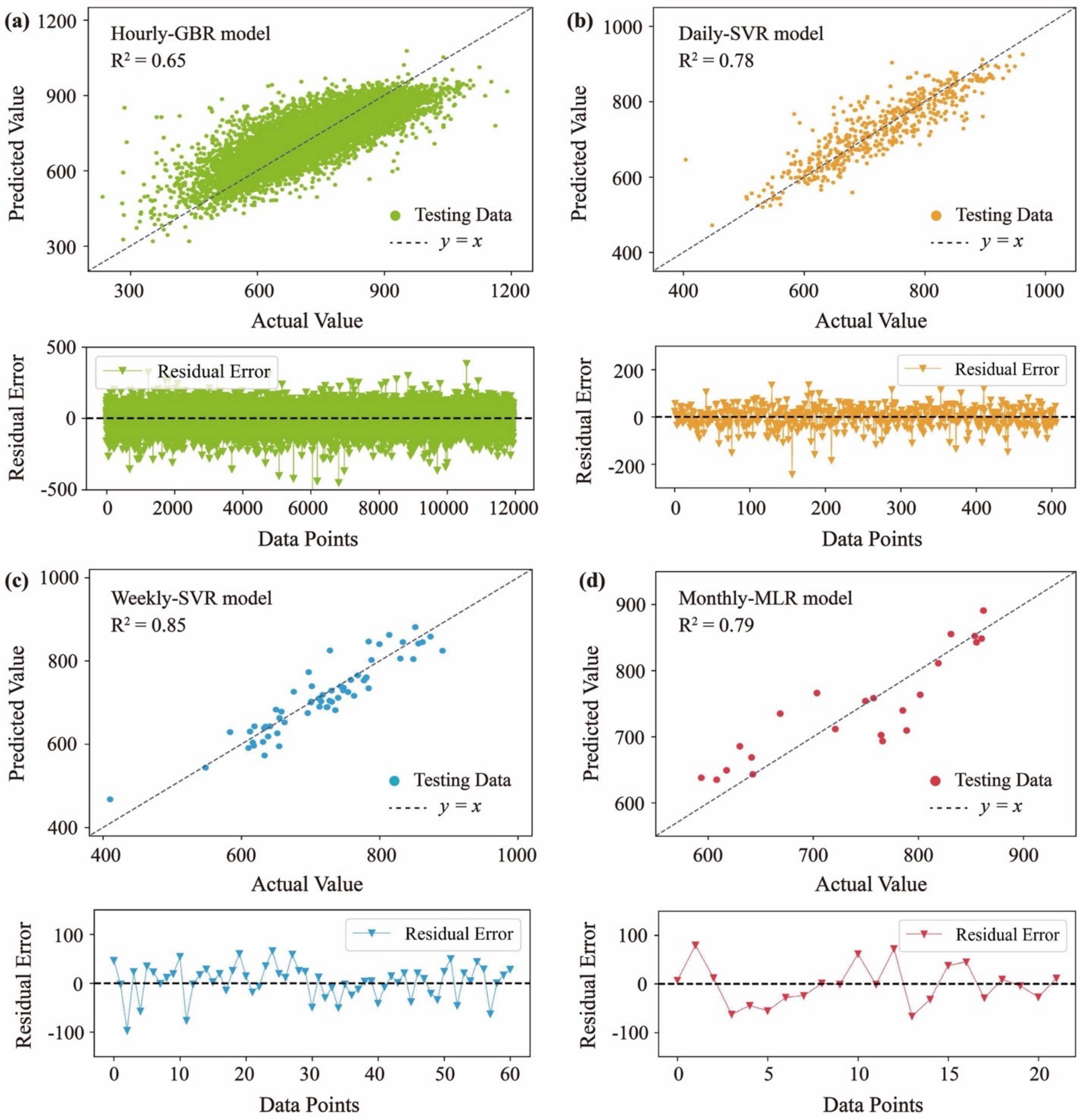


Figure 6.6 Scatter plots and residual plots of the best models at varying dataset resolutions, including (a) Hourly-GBR model, (b) Daily-SVR model, (c) Weekly-SVR model, and (d) Monthly-MLR model.

Table 6.7 Optimal hyperparameters of the best models at varying temporal resolutions.

Dataset	Model	Hyperparameter	Optimal Value
Hourly	GBR	*max_depth*	7
		learning_rate	0.13
		min_samples_split	10
		min_samples_leaf	2
		n_estimators	261
Daily	SVR	*C*	267.41
		gamma	2.62
		epsilon	10
Weekly	SVR	*C*	1000
		gamma	0.35
		epsilon	0.001

6.4.3. Feature importance analysis at varying temporal resolutions

In this study, the SHAP method was utilized to gain an in-depth understanding of how input variables affect the model (i.e., the hourly-GBR, daily-SVR, weekly-SVR, and monthly-MLR models) output and to pinpoint the most influential input variables at varying temporal resolutions (Djandja et al., 2023). The results of the SHAP analysis are presented in Figures 6.7-6.9. Figure 6.7 shows the qualitative analysis of the input variables' influence on the hourly, daily, weekly, and monthly truck productivity in the four models. To further understand the exact relationships between each input and the prediction, the SHAP partial dependence plots (PDPs) are displayed in Figure 6.8. Finally, Figure 6.9 gives the quantitative analysis of input variables' importance ranking at varying temporal resolutions.

As shown in Figure 6.7, the *y*-axis indicates the input variables; the *x*-axis denotes the SHAP values. Each dot in the summary plot designates a SHAP value (each data point's prediction minus the average prediction of all data points) for an input variable and a data point in the training dataset. The dot's color represents the input variable's value for that data point, from low (blue) to high (red). Remarkably, the vertical line (zero SHAP value) on the *x*-axis indicates the average prediction of truck productivity. On this basis, a negative SHAP value suggests that an input variable's effect leads to a lower prediction than the average, while a positive SHAP value means that its effect contributes to a higher prediction than the average. Using the hourly-GBR model in Figure 6.7(a) as an example, the longer the haul distance (red dots), the greater the negative SHAP value; reversely (blue dots), the larger the positive SHAP value. This indicates that haul distance had a negative relationship with truck productivity. Cervantes et al. (2019) also showed that truck productivity decreased with increasing haul distance based on Canada's oil sands company database. Likewise, in Figure 6.7(a), ambient temperature, number of trucks, humidity, wind speed, and precipitation negatively correlate with truck productivity. A larger number of trucks at mine sites causes an increase in the waiting time (e.g., waiting time at shovels or dumps), resulting in a longer cycle time and lower truck productivity (Anani & Awuah-Offei, 2013; Fan et al., 2023b). Weather-related input variables may affect drivers' vision (Sun et al., 2018), driving habits (Sagberg et al., 2015), and road conditions (Medinac et al., 2020), thus increasing cycle time and reducing truck productivity. Unlike these inputs, empty speed and number of shovels positively influence truck productivity. Empty speed determines the time it takes for a truck to return from dump sites to load sites, which can affect cycle time (Fan et al., 2022). Also, increasing the number of shovels improves shovel utilization and reduces the waiting time at shovels, thus enhancing truck productivity (Ercelebi & Bascetin, 2009). These first indications (the negative and positive relationships) between inputs and the output also apply to the daily-SVR, weekly-SVR, and monthly-MLR models in Figure 6.7(b)-(d).

In addition to the first indications, SHAP offered the one-way PDPs to observe the functional relationship (linear, monotonic, or complex nonlinear) between one input variable and the prediction. As shown in Figure 6.8(a)-(d), the truck haulage-related input variables in the hourly-GBR model are used as examples due to the largest amount of data encompassing rich information in the hourly dataset (47,777 data points). The PDPs for the weather-related input variables will be explained in Section 6.4.4 separately. In Figure 6.8(a), the predicted truck productivity (average prediction of truck productivity plus SHAP value) drops at a decreasing rate as haul distance rises from 0 to about 4.7 km (turning point), but the predictions are still higher than the average prediction (indicating by red dashed line). When haul distance continues to increase, the predicted truck productivity is always less than the average prediction. In contrast to the monotonic decrease in the prediction and haul distance, Figure 6.8(b) shows a monotonic increase in the prediction and empty speed. Notably, when empty speed is above about 34 km/h (turning point), the predicted truck productivity is greater than the average prediction. Unlike these two inputs, the number of trucks shows a complex nonlinear relationship with the predicted truck productivity in Figure 6.8(c). The predicted truck productivity presents two peaks (truck numbers around 8 and 20). This may be attributed to the change in truck-shovel allocation and the increase in shovel numbers (Bakhtavar & Mahmoudi, 2020). However, the predicted truck productivity decreases and falls below the average prediction when more than 20 trucks (turning point) are scheduled per hour. Figure 6.8(d) also presents a nonlinear relationship between the number of shovels and the predicted truck productivity. When the number of shovels exceeds four (turning point) per hour, the prediction is over the average prediction and gradually tends to level off. This study is akin to

Li (2022); Mangalathu et al. (2022); Ransom et al. (2022), who adopted one-way PDPs to analyze the individual effects of input variables on prediction; however, it is still challenging to quantify the importance ranking of input variables through PDPs.

The SHAP method can quantify feature importance by averaging the absolute SHAP values of each input variable. As shown in Figure 6.9(a)-(d), the *x*-axis is the average of the absolute SHAP values representing the feature importance; the *y*-axis is the input variables in descending order of importance. The feature importance in Figure 6.9(d) based on the monthly-MLR model was not adopted in this study because the model had an overfitting problem that prevented it from providing reliable results. From Figure 6.9(a)-(c), regardless of the temporal resolutions, the three most critical input variables were haul distance, empty speed, and ambient temperature. For instance, for the daily-SVR model, these three inputs' importance (mean absolute SHAP value) were 44.23, 41.99, 21.72, which was higher than that of number of trucks (10.65), humidity (4.66), number of shovels (3.98), wind speed (2.92), precipitation (1.40). This echoes our previous studies (Fan et al., 2022, 2023b, 2023d). This is mainly because these three inputs are closely related to truck cycle time. For instance, it was reported by Ma et al. (2023) that high ambient temperature (e.g., from 20 ℃ to 40 ℃) induced the increase in truck tire temperature (e.g., from 54 ℃ to 69 ℃), leading to tire fatigue and affecting truck cycle time. Moreover, it can be observed that the importance of the four weather-related input variables (highlighted by the red lines in Figure 6.9) increased as decreasing temporal resolutions: hourly (27.47) < daily (30.70) < weekly (34.89). This may be associated with data aggregation. First, this study averaged the data at hourly, daily, and weekly resolutions to smooth out noise and short-term fluctuations, resulting in more pronounced and influential trends and patterns in weather variables over longer time periods (Bodesheim et al., 2018; Nourani et al., 2019). For example, for the hourly and weekly models, the importance of ambient temperature, humidity, and wind speed increased from 21.73 to 23.92, 3.76 to 4.52, and 1.02 to 5.09, respectively. Second, precipitation has a cumulative effect over a long time period (Wen et al., 2019). Total precipitation during a week (e.g., maximum 85.30 mm in Table 6.3) can have a more substantial effect on road conditions and driving habits than an hour (e.g., maximum 14.10 mm in Table 6.1) (Xing et al., 2019). As a result, the importance of precipitation enhanced from 0.96 (hourly) to 1.36 (weekly). Similarly, Webb et al. (2003) built regression models between ambient temperature and water temperature of the River Exe at varying temporal resolutions (e.g., hourly, daily, and weekly). The results showed that the influence (explainable variance in percentage) of ambient temperature rose from hourly (67.4%) to daily (84.2%) and weekly (92.2%) incrementally.

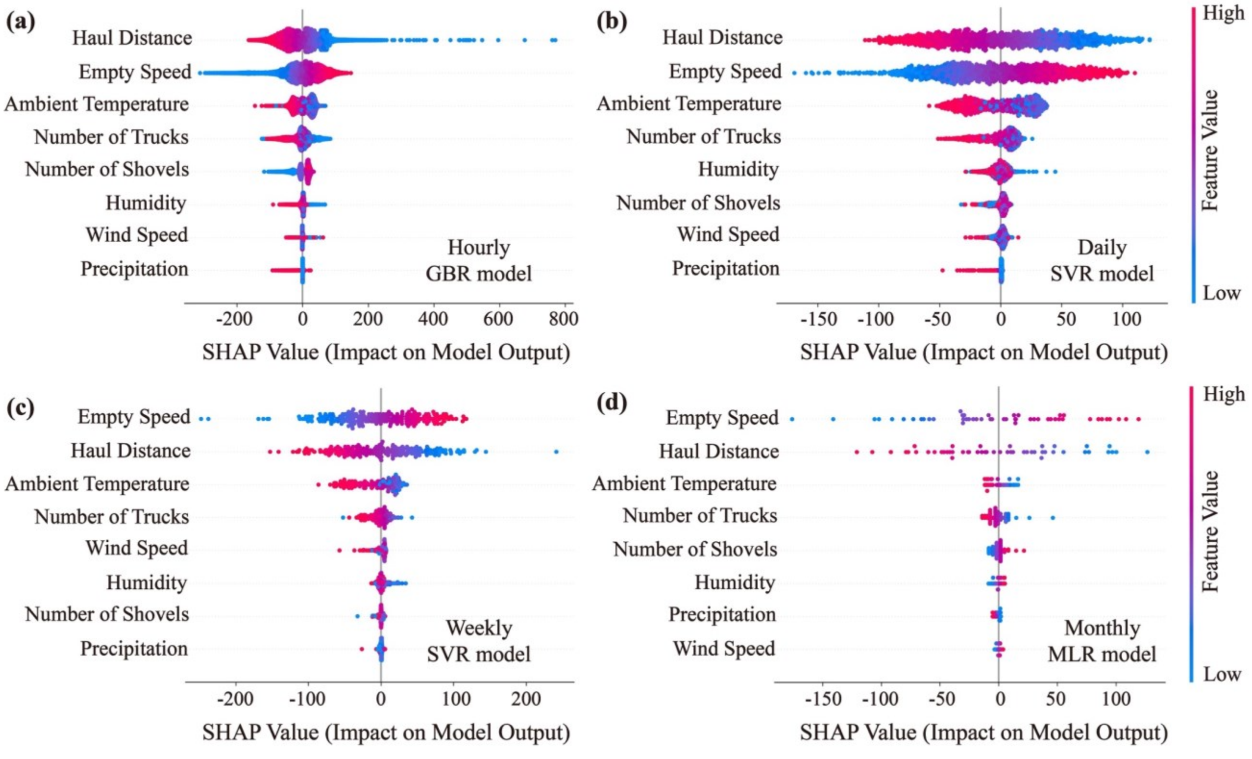


Figure 6.7 SHAP summary plots of ten input variables and their instances' impacts on the model output.

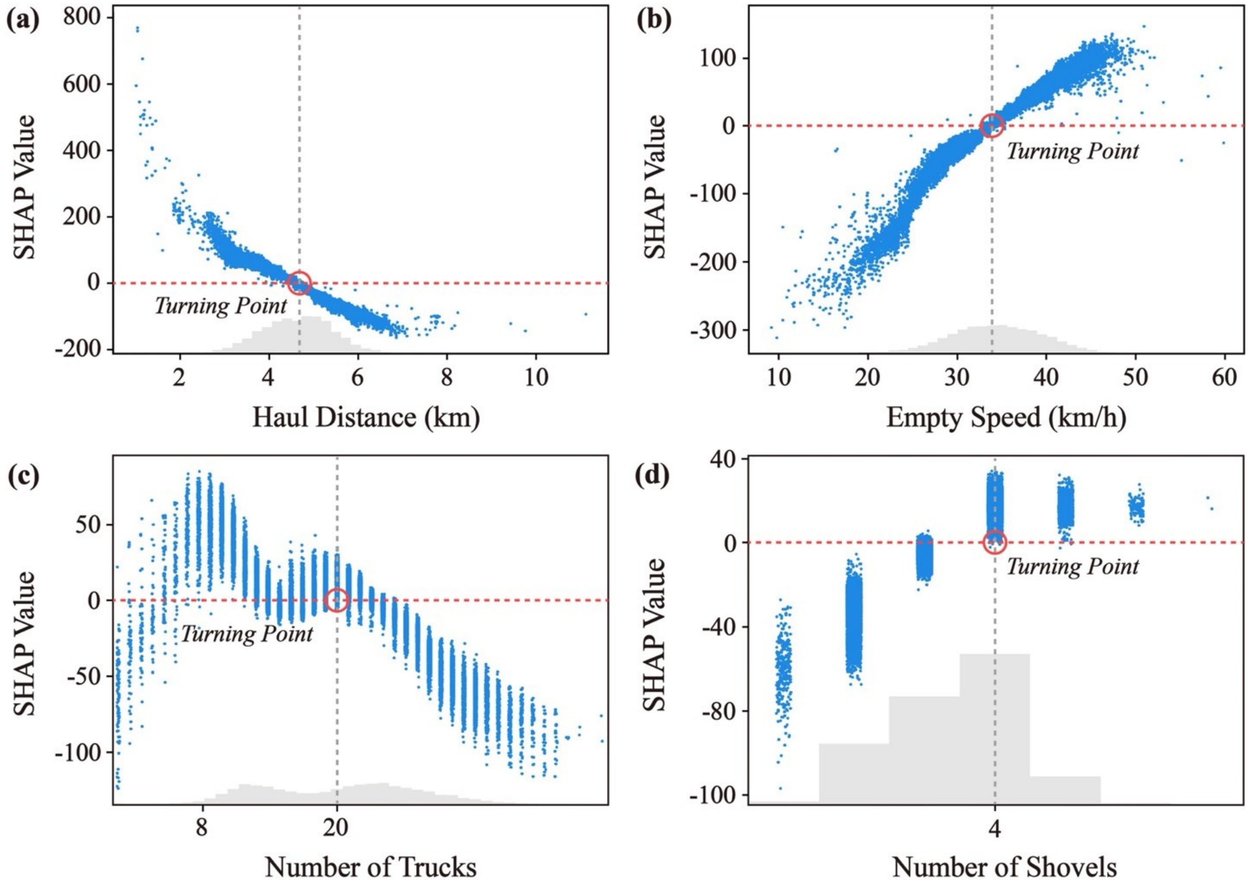


Figure 6.8 Relationships between the truck haulage-related input variables and SHAP values represented by one-way partial dependence plots (PDP) from the SHAP analysis based on the hourly-GBR model.

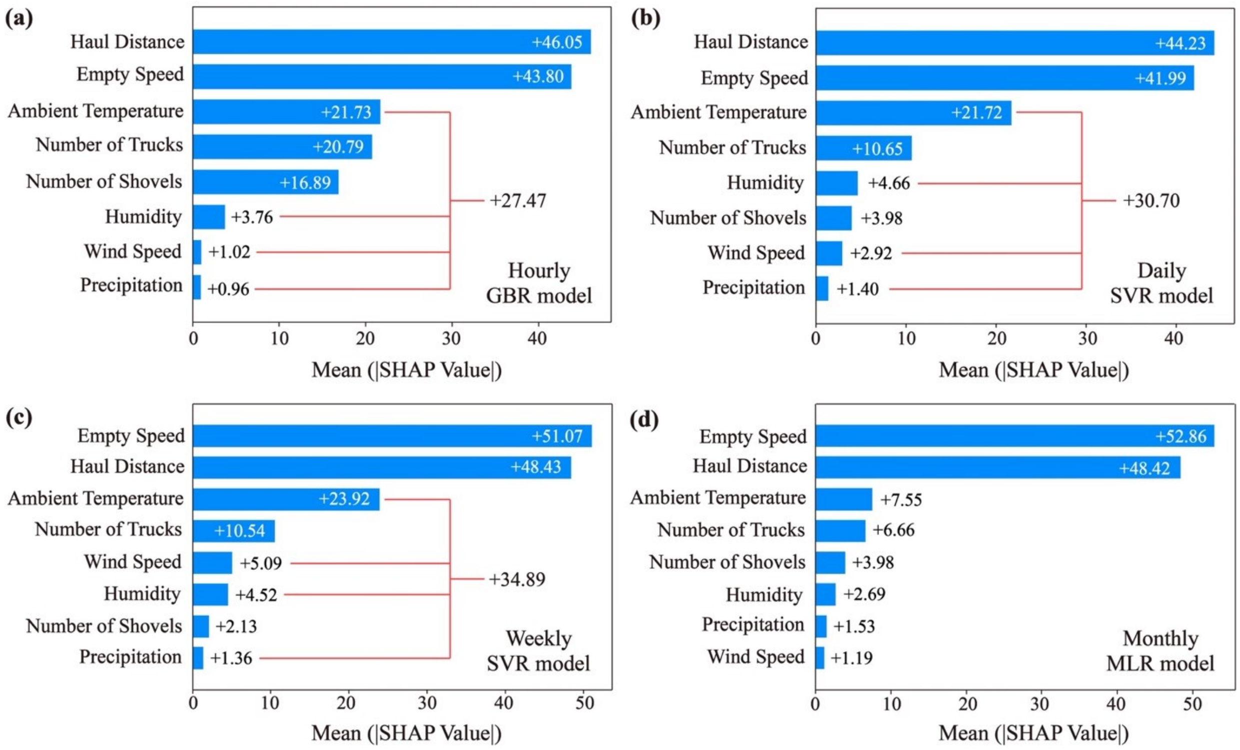


Figure 6.9 Feature importance analysis for four prediction models at varying temporal resolutions: (a) hourly-GBR; (b) daily-SVR; (c) weekly-SVR, and (d) monthly-MLR. The feature importance of the monthly-MLR model was not adopted in this study because this model had an overfitting problem that prevented it from providing reliable results.

6.4.4. Effect of extreme weather on truck productivity and truck-shovel allocation

In the last section, the one-way PDPs (Figure 6.8) were utilized to observe the individual relationships between the prediction and the truck haulage-related input variables. Unlike the one-way PDPs, the SHAP method can also offer two-way PDPs, which show the interactive relationships between two input variables and the prediction (Djandja et al., 2023). To provide insight into the truck-shovel allocation in extreme weather, this section used two-way PDPs to describe the interactions between the prediction and the weather-related input variables and the number of trucks (or shovels), as shown in Figures 6.10 and 6.11.

In Figures 6.10 and 6.11, the *x*-axis is the weather variables; the left and right *y*-axes are the SHAP values and the number of trucks (or shovels), respectively. The zero SHAP value (red dashed line) on the *y*-axis indicates the average prediction of truck productivity. The number of trucks (or shovels) is denoted by the color bar, ranging from minor (blue) to major (red). The yellowish overlay area in each plot represents the hourly weather extremes defined with reference to the Manual of Surface Weather Observation Standards (MANOBS, 2021) and Lou et al. (2017); Masud et al. (2021); Wheeler and Wilkinson (2004). For example, it is considered heavy rain when more than 8 mm of precipitation falls per hour. As shown in Figure 6.10(a), the predicted truck productivity (average prediction plus SHAP value) rises (below -12 ℃) and then drops (above -12 ℃) with ambient temperature. In other words, the predicted truck productivity reaches the maximum of around -12 ℃, which is in line with OSM (2021), indicating the optimal mining temperature is about -10 ℃ at oil sands mines in Alberta. At the extreme temperature, the predicted truck productivity declines dramatically from -30 ℃ to -40 ℃ and 30 ℃ to 40 ℃, but the correlation between the number of trucks and the prediction is not significant. In Figure 6.10(b), when the relative humidity (in percentage) is lower than 25%, the red dots are more distributed above the average prediction than the blue dots, suggesting that allocating more trucks can improve truck productivity in the arid working environment. For high relative humidity (>80%), the predicted truck productivity drops from the average prediction. This may be because road conditions (dryness or wetness) are related to relative humidity (Silion & Foşalău, 2014). After relative humidity, Figure 6.10(c) shows a complex nonlinear relationship between the predicted truck productivity and wind speed. For the extreme wind speed (>90 km/h), the blue dots are located further above the average prediction relative to the red dots, indicating that lowering the number of trucks in stormy weather may enhance truck productivity. This is attributed to the fact that storms can affect driving visibility (Choi & Nieto, 2011). Finally, Figure 6.10(d) displays that the predicted truck productivity almost decreases to a greater or lesser extent (below the average prediction), regardless of the precipitation. Under extreme precipitation (>8 mm/h), the red dots are below but closer to the average prediction, which implies that increasing the number of trucks during heavy rainfall can mitigate the decrease in truck productivity. These observations also apply to the interactions between the prediction and the weather variables and the number of shovels in Figure 6.11. Similar research was found by Ramhormozi et al. (2022) and Roh et al. (2016), who investigated the effect of extreme weather conditions on truck speed prediction and truck traffic volume. For example, Roh et al. (2016) demonstrated that when the cold ambient temperature (-20 °C or lower) interacted with snowfall, truck traffic fell sharply as snowfall increased (15 cm or higher). In brief, there were complex nonlinear relationships between the prediction and the weather-related input variables. Extreme weather, such as extreme wind speed, relative humidity, and precipitation, had a certain effect on truck-shovel allocation. This study is the first to use data-driven models and methods to investigate truck-shovel allocations in extreme weather, which provides new insights into mine planning for mining engineers.

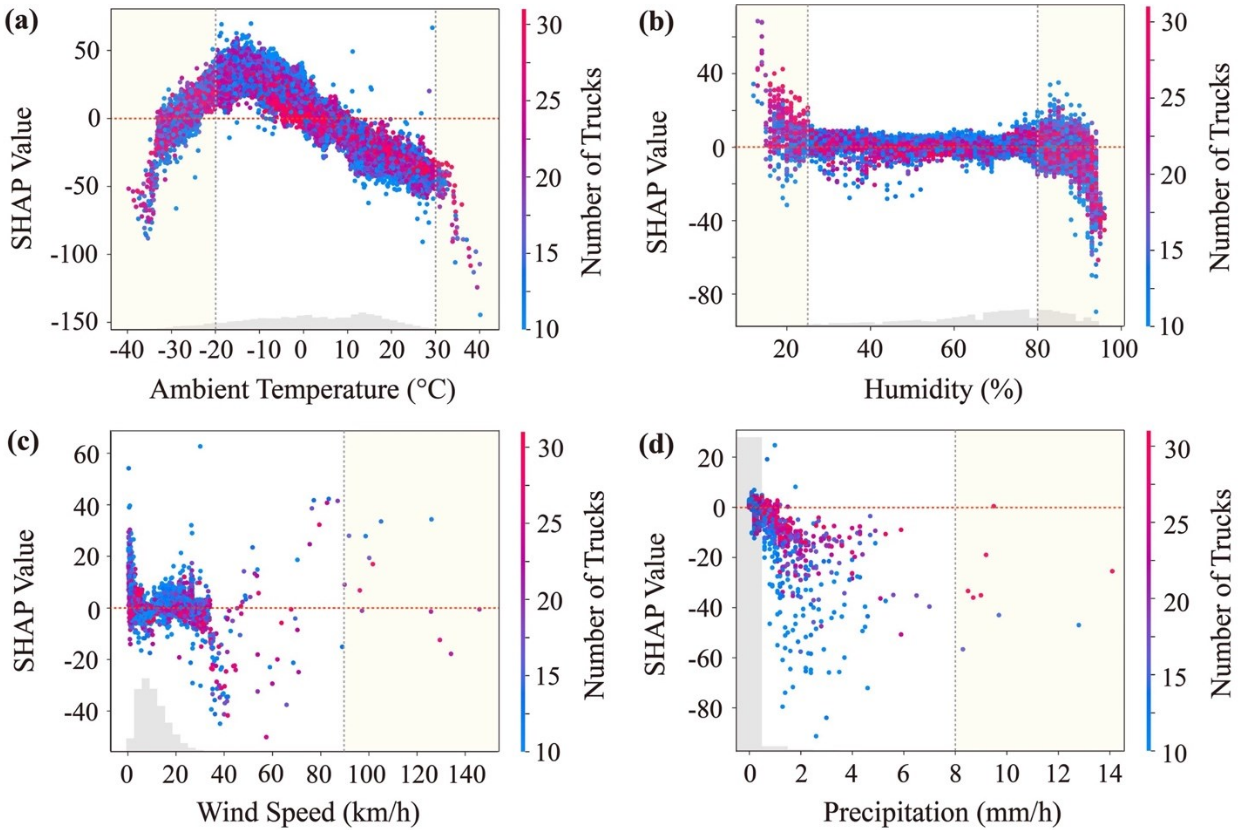


Figure 6.10 The interaction between the prediction (hourly truck productivity) and the weather-related input variables and truck allocation represented by two-way partial dependence plots.

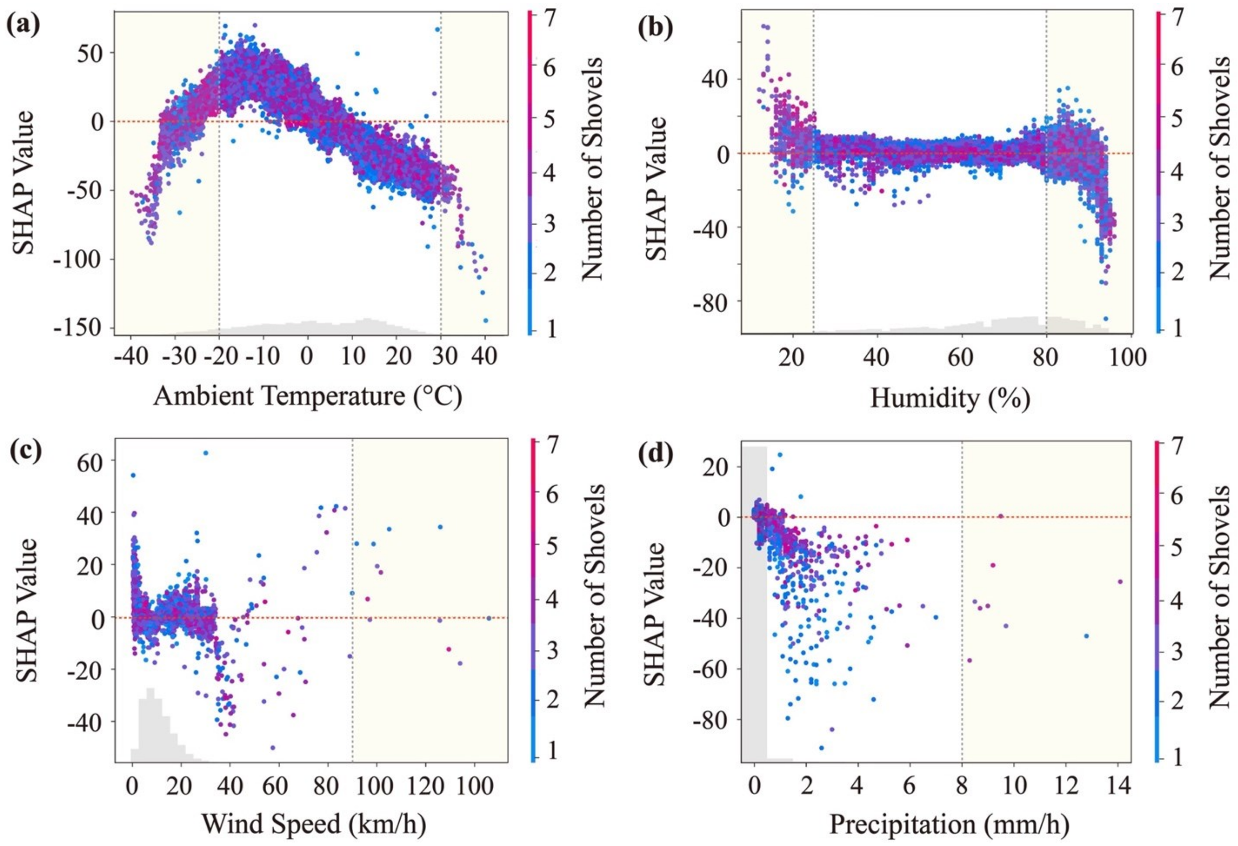
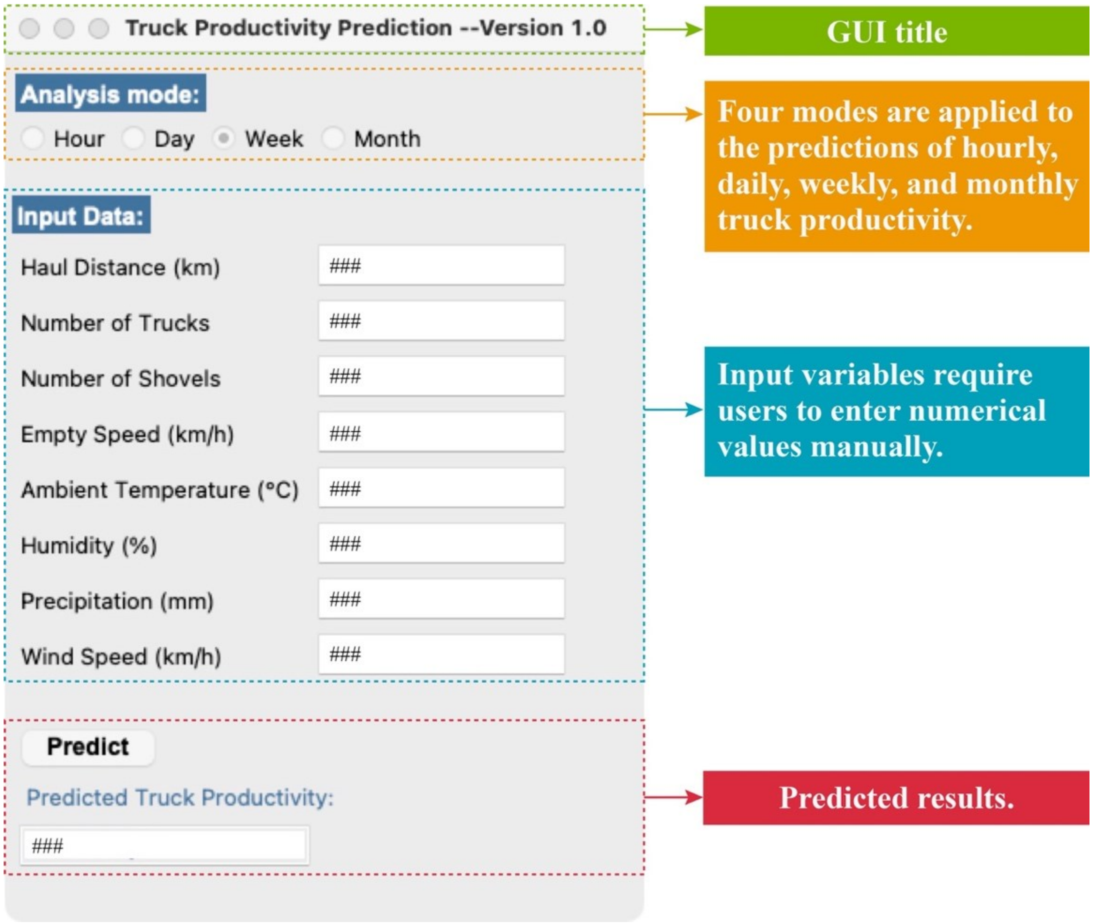


Figure 6.11 The interaction between the prediction (hourly truck productivity) and the weather-related input variables and shovel allocation represented by two-way partial dependence plots.

6.5. Graphical User Interface for Truck Productivity Prediction

To facilitate access to the solutions of this study by site engineers and researchers, a unified GUI was developed based on the best prediction models retained from Section 6.4.2, which is illustrated in Figure 6.12. This GUI consists of four essential parts: GUI title, analysis modes, input data, and predict button. The GUI title shows the functional purpose (truck productivity prediction) and the current version (version 1.0). The analysis mode contains four radio buttons: Hour, Day, Week, and Month, indicating the well-established models (hourly-GBR, daily-SVR, weekly-SVR, and monthly-MLR models) for predicting average truck productivity per hour, day, week, and month, respectively. The input data lists eight blank input variables that require the users to enter the numerical values manually. These eight inputs include four truck haulage-related inputs (haul distance, number of trucks, number of shovels, and empty speed) and four weather-related inputs (ambient temperature, humidity, precipitation, and wind speed). Finally, the predict button gives the predicted results rapidly based on the input data. The prediction fails when there are blank entries or incorrect (e.g., 9,999 km for haul distance) and invalid (e.g., 34.55 for the number of trucks) entries. The entire GUI framework was generated using a Tkinter package (Moore, 2021) and implemented in a Python-based programming environment (version 3.10.9). To conclude, this is the first study to provide a GUI for predicting mining truck productivity at varying temporal resolutions. This GUI can alleviate the need for complex modeling analysis and intensive computation, which will be instrumental in making decisions more easily and quickly for mining engineers and researchers.



Truck Productivity Prediction --Version 1.0

Analysis mode:

○ Hour ○ Day ● Week ○ Month

Input Data:

Haul Distance (km): ###

Number of Trucks: ###

Number of Shovels: ###

Empty Speed (km/h): ###

Ambient Temperature (°C): ###

Humidity (%): ###

Precipitation (mm): ###

Wind Speed (km/h): ###

Predict

Predicted Truck Productivity: ###

GUI title

Four modes are applied to the predictions of hourly, daily, weekly, and monthly truck productivity.

Input variables require users to enter numerical values manually.

Predicted results.

Figure 6.12 The GUI for assessing hourly, daily, weekly, and monthly truck productivity. ("###": the input information is not disclosed as it is the proprietary property of mining companies.)

6.6. Limitations and Future Prospective

This study explored the influence of temporal resolutions on modeling, analyzed the contribution of input variables to the model output, and developed a simple and easy-to-use GUI for mining engineers. Nevertheless, this study has the following limitations that require further future work to improve it. First, both nonlinear and linear models suffered from overfitting problems when selecting the best model for the monthly data. For example, the R^2 of SVR was 0.91 on the training data but 0.71 on the testing data. To reduce overfitting, more data points need to be included in the training dataset (Arachchilage et al., 2023). Second, the GUI is required to be further upgraded and optimized in future studies. As more data come, the models in the GUI will be updated to improve prediction accuracy and avoid overfitting. Moreover, the GUI will include additional features, such as reading tabulated data without manual input, plotting, and automatically analyzing the relationships between variables and outputs. Third, more input variables are taken into account in the modeling when data avail, such as tire temperature (Ma et al., 2023), solar radiation (Modenese et al., 2018), snowfall (Fan et al., 2023c), and road elevation (Medinac et al., 2020). These variables may influence truck cycle time and truck productivity. For example, Ma et al. (2023) reported that an increase in tire temperature leads to rubber failure, decreasing truck tire performance and affecting truck cycle time.

6.7. Conclusions

Data-driven modeling (i.e., machine learning) has been initiated as a new direction for assessing mine truck productivity. This study used six machine learning methods to establish prediction models between eight input variables and truck productivity and explored the temporal effects (i.e., hourly, daily, weekly, and monthly) on the selection of the best models. Furthermore, SHAP (Shapley Additive exPlanations) was utilized as a model interpretation method to analyze how the input variables affect the model output and to identify the most influential inputs. The principal findings are summarized below:

(1) The nonlinear relationship between input variables and truck productivity progressively diminished with decreasing temporal resolutions (i.e., from hourly to monthly). For example, regarding RMSE on the testing datasets, the nonlinear GBR (70.15) performed better than the linear MLR (79.12) at the hourly resolution. At the daily resolution, GBR (49.15) performed close to but still better than MLR (50.51). However, GBR (46.92) underperformed MLR (34.29) at the weekly resolutions and showed more significant overfitting than MLR at the

monthly resolution.

(2) Mining engineers can make more accurate predictions of truck productivity at the weekly resolution compared with other resolutions. This study selected the four best models: hourly-GBR, daily-SVR, weekly-SVR, and monthly-MLR models. For these models, the weekly-SVR model had a higher R^2 (0.85) than the hourly-GBR (0.65), daily-SVR (0.78), and weekly-SVR (0.79) models.

(3) Regardless of the temporal resolutions, the three most influential input variables were haul distance, empty speed, and ambient temperature. For instance, for the daily-SVR model, the importance (i.e., mean absolute SHAP value) of these three variables were 44.23, 41.99, 21.72, which was higher than that of the number of trucks (10.65), humidity (4.66), number of shovels (3.98), wind speed (2.92), and precipitation (1.40).

(4) The feature importance of the four weather-related input variables increased as decreasing temporal resolutions. For example, at the hourly and weekly resolutions, ambient temperature, humidity, wind speed, and precipitation's importance rose from 21.73 to 23.92, 3.76 to 4.52, 1.02 to 5.09, and 0.96 to 1.36, respectively. In addition, the importance sum of these input variables escalated on hourly (27.24), daily (30.70), and weekly resolutions (34.89).

(5) Extreme weather, such as extreme wind speed, precipitation, and relative humidity, had a certain effect on truck-shovel allocation at mine sites. For instance, under extreme precipitation (>8 mm/h), increasing the number of trucks during heavy rainfall can mitigate the decrease in truck productivity. This study is the first to investigate truck-shovel allocations in extreme weather, which provides new insights into mine planning for mining engineers.

(6) A unified GUI was designed and developed for the first time to predict hourly, daily, weekly,

and monthly truck productivity at mine sites. This GUI can alleviate the need for complex modeling analysis and intensive computation, which will be instrumental in making decisions more easily and quickly for mining engineers and researchers.

Chapter 7. Conclusion and future work

7.1. Conclusions

Overall, this thesis aims to apply machine learning techniques to improving truck productivity prediction accuracy at mine sites. In particular, this thesis focuses on developing a unified toolkit for truck productivity prediction in oil sands mining, which consists of various machine learning models built based on massive truck haulage data at varying temporal resolutions (e.g., per cycle, hour, day, week, and month). The findings will help mine management better understand and predict truck productivity for hauling efficiency improvement, strategic decision making, and cost reductions in oil sands mining. The main concluding remarks of this thesis are enumerated as follows:

(1) GMM significantly enhanced the model predictability of truck productivity by preprocessing massive truck haulage data and performed better than K-means. For example, the R^2 of the GMM-GBR model (86.98%) was about two times higher than the GBR model (44.76%). Moreover, in terms of XGBoost, the R^2 of the model was much greater based on GMM analysis (80.37%) compared with K-means analysis (33.06%). This information can provide new insights and inspiration for engineers to deal with massive amounts of engineering data in their future work.

(2) The tree-based ensemble models performed better than the single DT models in predicting truck productivity without and with GMM analysis. For instance, without GMM analysis, the R^2 of the RF model was 44.05%, which was higher than that of the decision tree model (the DT model), with a value of 31.96%. With GMM analysis, the R^2 of the GMM-RF model (87.16%) remained higher than the GMM-DT model (78.99%).

(3) The BRNN model outperformed the BPNN and ELM models in predicting low, medium, and high values of truck productivity. For example, the RMSE, MAE, and R^2 were 45.94, 38.98, and 85.20% for the BRNN model, while these metrics were 47.19, 39.76, and 84.39% for the BPNN model, and 46.25, 39.21, and 85.01% for the ELM model.

(4) Haul distance contributed the most in constructing linear and nonlinear prediction models of truck productivity when temporal resolutions were not considered. For example, for the MLR model, the relative importance of the haul distance was 54.65%, which was higher than that of empty speed (23.14%), destination (6.22%), ambient temperature (13.82%), and precipitation (2.18%). As for the RF model, the relative importance of haul distance was 43.51%, which was higher than empty speed (21.77%), waiting at shovel (18.18%), ambient temperature (12.71%), destination (2.29%), spotting (1.03%), and waiting at dump (0.52%). This finding helps mining engineers gain an in-depth understanding of the major real-world influences on truck productivity.

(5) When considering temporal resolutions (e.g., daily and weekly), the three most influential input variables were haul distance, empty speed, and ambient temperature regardless of the resolutions. For instance, for the daily-SVR model, 44.23, 41.99, 21.72, which was higher than that of the number of trucks (10.65), humidity (4.66), number of shovels (3.98), wind speed (2.92), and precipitation (1.40). Similarly, for the weekly-SVR model, the importance of these three variables were 48.43, 51.07, 23.92, which was greater than that of the number of trucks (10.54), wind speed (5.09), humidity (4.52), number of shovels (2.13), and precipitation (1.36).

(6) Mining engineers can make more accurate predictions of truck productivity at the weekly resolution compared with other resolutions. This study selected the four best prediction models

for varying temporal resolutions: hourly-GBR, daily-SVR, weekly-SVR, and monthly-MLR models. For these models, the weekly-SVR model had a higher R^2 (0.85) than the hourly-GBR (0.65), daily-SVR (0.78), and weekly-SVR (0.79) models. This helps more rational decision-making and planning in the week-to-week operations at mine sites.

(7) Extreme weather, such as extreme wind speed, precipitation, and relative humidity, had a certain effect on truck-shovel allocation at mine sites. For example, under extreme precipitation (>8 mm/h), increasing the number of trucks during heavy rainfall can mitigate the decrease in truck productivity. This research is the first to investigate truck-shovel allocations in extreme weather, which provides new insights into mine planning for mining engineers.

(8) A unified GUI was designed and developed for the first time to predict truck productivity at varying temporal resolutions. This GUI is easy to use and consists of four basic modes: hour, day, week, and month, which correspond to the well-established models (i.e., hourly-GBR, daily-SVR, weekly-SVR, and monthly-MLR models) for predicting average truck productivity per hour, day, week, and month, respectively.

7.2. Key contributions

The findings from this thesis are significant for mining engineers and researchers in the resource industry. The key contributions of this Ph.D. research are summarized below:

(1) For the first time, unique and massive truck haulage data from the VIMS were used as training data for machine learning to build truck productivity prediction models at mine sites. This will benefit mining engineers and researchers better understand the dynamic and complex process of truck haulage under real-site operating conditions, thus estimating truck productivity more accurately.

(2) This study first proposed the use of GMM unsupervised clustering to preprocess massive truck haulage data for improving the model predictability. Truck haulage data usually present multi-peak Gaussian distributions, which is the rationale of selecting GMM to handle truck haulage data. This also implies that this study provides a potential solution to deal with massive data with multi-peak Gaussian distributions from similar engineering problems.

(3) For the first time, machine learning techniques were employed to construct accurate regression models of truck productivity. These machine learning models can replace the traditional curve-fitting approach in oil sands mining companies to obtain more accurate predictions. Meanwhile, the experience associated with the application of machine learning can also be extended to other open-pit mines using truck haulage.

(4) This study innovatively investigated the effect of temporal resolutions on establishing truck productivity prediction models in open-pit mining. Real-site weather conditions are often seen as variables affecting truck productivity, but temporal resolutions of weather conditions have not been considered in previous studies. Understanding the effect of temporal resolutions will benefit mining engineers in making more sound forecasts as well as short- and long-term mine planning.

(5) This study was the first to analyze the influence of extreme weather on truck-shovel allocation in open-pit mining. Truck-shovel scheduling is a core issue at mine sites as it is associated with a mine's production, profit, and expenditure. Numerous previous studies have been conducted to optimize truck-shovel allocation, but truck-shovel allocation under extreme weather has not been investigated. With global warming and more frequent extreme weather, this investigation is critical because the impact of weather conditions is becoming more pronounced. This study

used machine learning techniques to analyze this notable engineering problem, which provides mining engineers with new insights into mine planning.

(6) This study is the first one that crafted a unified GUI to estimate truck productivity at varying temporal resolutions. This GUI is a successful example of machine learning committing to real impacts in mining engineering. It significantly alleviates the need for complex modeling analysis and intensive computation, which will be instrumental in making decisions more easily and quickly for mining engineers and researchers.

7.3. Limitations and future work

This thesis has developed a toolkit based on machine learning and massive truck haulage data for improving truck productivity prediction, but there are still some limitations as listed below.

(1) In this study, restrained by data availability and proprietorship, only the data acquired from the VIMS and Environmental Canada were utilized to train truck productivity prediction models. Therefore, additional input variables that have not been involved may affect truck productivity, such as truck tire properties (Ma et al., 2023), loaded speed (Fan et al., 2023b), and pavement elevation (Chanda & Gardiner, 2010), and driver habits (Sun et al., 2018). For example, Ma et al. (2021) reported that high tire temperatures could cause rubber failure of the off-the-road tire at mine sites, thus affecting truck productivity. These potential influencing inputs may be added to future work to construct truck productivity prediction models at mine sites.

(2) This study mainly utilized a grid search and a sequential model-based optimization algorithm to tune the hyperparameters built in machine learning algorithms. However, there are various optimization algorithms that have been proven helpful in tuning hyperparameters, such as genetic optimization (Chung & Shin, 2020), whale optimization (Ge et al., 2022), and particle

swarm optimization (Bardhan et al., 2022). Therefore, these optimization algorithms will be utilized in future work to enhance the generalizability of the prediction model.

(3) The GUI is required to be further upgraded and optimized in future studies. As more data come, the models in the GUI will be re-trained to improve prediction accuracy and avoid overfitting. Moreover, this GUI will include additional functions, such as reading tabulated data without manual input, plotting, and automatically analyzing the partial relationships between inputs and the output.

(4) The current machine learning models were established based on large amounts of training data, leading to a high computational complexity. When new data comes, these machine learning algorithms need to be rerun, which takes a lot of time and memory. Online machine learning may be the solution to address this challenge, as its only necessary to process a small batch of data and keep prediction models updated when new data arrive (Carvajal Soto et al., 2019). This ensures the accuracy of prediction models and reduces the need for PC memory.

(5) Commonly used machine learning algorithms were employed in this study to deal with the regression problem. Some more advanced algorithms have yet to be used in open-pit mining or other related fields, such as deep neural networks (Yuan et al., 2021), reinforcement learning (Marugán, 2023), transfer learning (Ma et al., 2019), and image processing algorithms (Jing et al., 2022). These methods have great potential to cope with different types, distributions, and variability of data and increase the generalization ability of prediction models. In the future, there are many more research efforts to develop intelligent mining using these algorithms.

Bibliography

Ahmed, W., Muhammad, K. & Siddiqui, F.I. (2020). Predicting calorific value of Thar lignite deposit: A comparison between back-propagation neural networks (BPNN), gradient boosting trees (GBT), and multiple linear regression (MLR). Applied Artificial Intelligence, 34(14), 1124-1136.

Akram, B.A., Akbar, A.H. & Shafiq, O. (2018). HybLoc: Hybrid indoor Wi-Fi localization using soft clustering-based random decision forest ensembles. IEEE Access, 6, 38251-38272.

Al-Jarrah, O.Y., Yoo, P.D., Muhaidat, S., Karagiannidis, G.K. & Taha, K. (2015). Efficient machine learning for big data: A review. Big Data Research, 2(3), 87-93.

Alam, M.S. & Paul, S. (2020). A comparative analysis of clustering algorithms to identify the homogeneous rainfall gauge stations of Bangladesh. Journal of Applied Statistics, 47(8), 1460-1481.

Alarie, S. & Gamache, M. (2002). Overview of solution strategies used in truck dispatching systems for open pit mines. International Journal of Surface Mining, Reclamation and Environment, 16(1), 59-76.

Alrfou, K., Kordijazi, A., Rohatgi, P. & Zhao, T. (2022). Synergy of unsupervised and supervised machine learning methods for the segmentation of the graphite particles in the microstructure of ductile iron. Materials Today Communications, 30, 103174.

Anani, A. & Awuah-Offei, K. (2013). Incorporating cycle time dependency in truck-shovel modeling. In: Society for Mining, Metallurgy & Exploration (SME) Annual Meeting. Denver, Colorado.

Arachchilage, C.B., Fan, C., Zhao, J., Huang, G. & Liu, W.V. (2023). A machine learning model to predict unconfined compressive strength of alkali-activated slag-based cemented paste backfill. Journal of Rock Mechanics and Geotechnical Engineering (In press).

Arciszewski, T.J., Hazewinkel, R.R.O. & Dubé, M.G. (2022). A critical review of the ecological status of lakes and rivers from Canada's oil sands region. Integrated Environmental Assessment and Management, 18(2), 361-387.

Asamer, J. & Reinthaler, M. (2010). Estimation of road capacity and free flow speed for urban roads under adverse weather conditions. In: 13th International IEEE Conference on Intelligent Transportation Systems (pp. 812-818).

Aydin, H.E. & Iban, M.C. (2023). Predicting and analyzing flood susceptibility using boosting-based ensemble machine learning algorithms with SHapley Additive exPlanations. Natural Hazards, 116(3), 2957-2991.

Baek, J. & Choi, Y. (2019). Deep neural network for ore production and crusher utilization prediction of truck haulage system in underground mine. Applied Sciences 9(19), 4180.

Baek, J. & Choi, Y. (2020). Deep neural network for predicting ore production by truck-haulage systems in open-pit mines. Applied Sciences, 10(5), 1657.

Bag, S., Rahman, M.S., Srivastava, G., Chan, H.L. & Bryde, D.J. (2022). The role of big data and predictive analytics in developing a resilient supply chain network in the South African mining industry against extreme weather events. International Journal of Production Economics, 251, 108541.

Bahl, A., Hellack, B., Balas, M., Dinischiotu, A., Wiemann, M., Brinkmann, J., Luch, A., Renard, B.Y. & Haase, A. (2019). Recursive feature elimination in random forest classification supports nanomaterial grouping. NanoImpact, 15, 100179.

Bakhtavar, E. & Mahmoudi, H. (2020). Development of a scenario-based robust model for the optimal truck-shovel allocation in open-pit mining. Computers & Operations Research, 115, 104539.

Bangaru, S.S., Wang, C., Hassan, M., Jeon, H.W. & Ayiluri, T. (2019). Estimation of the degree of hydration of concrete through automated machine learning based microstructure analysis - A study on effect of image magnification. Advanced Engineering Informatics, 42, 100975.

Bardhan, A., Kardani, N., Alzo'ubi, A.K., Roy, B., Samui, P. & Gandomi, A.H. (2022). Novel integration of extreme learning machine and improved Harris hawks optimization with particle swarm optimization-based mutation for predicting soil consolidation parameter. Journal of Rock Mechanics and Geotechnical Engineering, 14(5), 1588-1608.

Bartos, P.J. (2007). Is mining a high-tech industry?: Investigations into innovation and productivity advance. Resources Policy, 32(4), 149-158.

Berlin, K.S., Williams, N.A. & Parra, G.R. (2013). An introduction to latent variable mixture modeling (part 1): Overview and cross-sectional latent class and latent profile analyses. Journal of Pediatric Psychology, 39(2), 174-187.

Bishop, C.M. (2006). Pattern Recognition and Machine Learning. Verlag New York: Springer.

Bo, Y., Liu, Q., Huang, X. & Pan, Y. (2022). Real-time hard-rock tunnel prediction model for rock mass classification using CatBoost integrated with Sequential Model-Based Optimization. Tunnelling and Underground Space Technology, 124, 104448.

Bodesheim, P., Jung, M., Gans, F., Mahecha, M.D. & Reichstein, M. (2018). Upscaled diurnal cycles of land-atmosphere fluxes: a new global half-hourly data product. Earth System Science Data, 10(3), 1327-1365.

Boratto, T.H.A., Cury, A.A. & Goliatt, L. (2023). Machine learning-based classification of bronze alloy cymbals from microphone captured data enhanced with feature selection approaches. Expert Systems with Applications, 215, 119378.

Both, C. & Dimitrakopoulos, R. (2020). Joint stochastic short-term production scheduling and fleet management optimization for mining complexes. Optimization and Engineering, 21(4), 1717-1743.

Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5-32.

Breiman, L., Friedman, J.H., Olshen, R.A. & Stone, C.J. (1984). Classification and regression trees (1st Edition ed.). Belmont, CA: Wadsworth.

Bui, D.T., Khosravi, K., Tiefenbacher, J., Nguyen, H. & Kazakis, N. (2020). Improving prediction of water quality indices using novel hybrid machine-learning algorithms. Science of The Total Environment, 721, 137612.

Cakir, M., Guvenc, M.A. & Mistikoglu, S. (2021). The experimental application of popular machine learning algorithms on predictive maintenance and the design of IIoT based condition monitoring system. Computers & Industrial Engineering, 151, 106948.

Cao, E., Bao, T., Li, H., Xie, X., Yuan, R., Hu, S. & Wang, W. (2022). A hybrid feature selection-multidimensional LSTM framework for deformation prediction of super high arch dams. KSCE Journal of Civil Engineering, 26(11), 4603-4616.

Capó, M., Pérez, A. & Lozano, J.A. (2017). An efficient approximation to the K-means clustering for massive data. Knowledge-Based Systems, 117, 56-69.

CAPP. (2018). A strong energy sector is key to ensure Canada's prosperity for the future. In: Canadian Association of Petroleum Producers (CAPP).

Carvajal Soto, J.A., Tavakolizadeh, F. & Gyulai, D. (2019). An online machine learning framework for early detection of product failures in an Industry 4.0 context. International Journal of Computer Integrated Manufacturing, 32(4-5), 452-465.

Cervantes, E.G., Upadhyay, S.P. & Askari-Nasab, H. (2019). Improvements to production planning in oil sands mining through analysis and simulation of truck cycle times. In: Mining Optimization Laboratory (MOL) (pp. 142-156): University of Alberta.

Çetinkaya, A. & Baykan, Ö.K. (2020). Prediction of middle school students' programming talent using artificial neural networks. Engineering Science and Technology, an International Journal, 23(6), 1301-1307.

Chanda, E.K. & Gardiner, S. (2010). A comparative study of truck cycle time prediction methods in open-pit mining. Engineering, Construction and Architectural Management, 17(5), 446-460.

Chen & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In: Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785-794). New York, NY, USA: ACM Digital Library.

Chen, Wang, X., Cai, Z., Liu, C., Zhu, Y. & Lin, W. (2021). DP-GMM clustering-based ensemble learning prediction methodology for dam deformation considering spatiotemporal differentiation. Knowledge-Based Systems, 222, 106964.

Chencho, Li, J., Hao, H., Wang, R. & Li, L. (2022). Structural damage quantification using ensemble-based extremely randomised trees and impulse response functions. Structural Control and Health Monitoring, 29(10), e3033.

Choi, Y., Nguyen, H., Bui, X.N. & Nguyen-Thoi, T. (2022). Optimization of haulage-truck system performance for ore production in open-pit mines using big data and machine learning-based methods. Resources Policy, 75, 102522.

Choi, Y., Nguyen, H., Bui, X.N., Nguyen-Thoi, T. & Park, S. (2021). Estimating ore production in open-pit mines using various machine learning algorithms based on a truck-haulage system and support of internet of things. Natural Resources Research, 30(2), 1141-1173.

Choi, Y. & Nieto, A. (2011). Optimal haulage routing of off-road dump trucks in construction and mining sites using Google Earth and a modified least-cost path algorithm. Automation in Construction, 20(7), 982-997.

Chong, D., Wang, N., Su, S. & Li, L. (2023). Global warming impact assessment of asphalt pavement by integrating temporal aspects: A dynamic life cycle assessment perspective. Transportation Research Part D: Transport and Environment, 117, 103663.

Chung, H. & Shin, K.-s. (2020). Genetic algorithm-optimized multi-channel convolutional neural network for stock market prediction. Neural Computing and Applications, 32(12), 7897-7914.

Ciulla, G. & D'Amico, A. (2019). Building energy performance forecasting: A multiple linear regression approach. Applied Energy, 253, 113500.

Çolak, A.B. (2022). Prediction of viscous dissipation effects on magnetohydrodynamic heat transfer flow of copper-poly vinyl alcohol Jeffrey nanofluid through a stretchable surface using artificial neural network with Bayesian Regularization. Chemical Thermodynamics and Thermal Analysis, 6, 100056.

Cortes, C. & Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273-297.

Cui, K. & Jing, X. (2019). Research on prediction model of geotechnical parameters based on BP neural network. Neural Computing and Applications, 31(12), 8205-8215.

David, E.R. & James, L.M. (1987). Learning Internal Representations by Error Propagation. In: Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations (pp. 318-362): MITP.

Daware, S., Chandel, S. & Rai, B. (2022). A machine learning framework for urban mining: A case study on recovery of copper from printed circuit boards. Minerals Engineering, 180, 107479.

Delen, D., Kuzey, C. & Uyar, A. (2013). Measuring firm performance using financial ratios: A decision tree approach. Expert Systems with Applications, 40(10), 3970-3983.

Demir, S. & Sahin, E.K. (2023). An investigation of feature selection methods for soil liquefaction prediction based on tree-based ensemble algorithms using AdaBoost, gradient boosting, and XGBoost. Neural Computing and Applications, 35(4), 3173-3190.

Demirbay, B., Kara, D.B. & Uğur, Ş. (2020). A Bayesian regularized feed-forward neural network model for conductivity prediction of PS/MWCNT nanocomposite film coatings. Applied Soft Computing, 96, 106632.

Dhini, A., Surjandari, I., Kusumoputro, B. & Kusiak, A. (2022). Extreme learning machine - radial basis function (ELM-RBF) networks for diagnosing faults in a steam turbine. Journal of Industrial and Production Engineering, 39(7), 572-580.

Dhulipala, S. & Patil, G.R. (2020). Freight production of agricultural commodities in India using multiple linear regression and generalized additive modelling. Transport Policy, 97, 245-258.

Diaz-Rozo, J., Bielza, C. & Larrañaga, P. (2020). Machine-tool condition monitoring with Gaussian mixture models-based dynamic probabilistic clustering. Engineering Applications of Artificial Intelligence, 89, 103434.

Dindarloo, S.R. & Siami-Irdemoosa, E. (2017). Data mining in mining engineering: results of classification and clustering of shovels failures data. International Journal of Mining, Reclamation and Environment, 31(2), 105-118.

Djandja, O.S., Kang, S., Huang, Z., Li, J., Feng, J., Tan, Z., Salami, A.A. & Lougou, B.G. (2023). Machine learning prediction of fuel properties of hydrochar from co-hydrothermal carbonization of sewage sludge and lignocellulosic biomass. Energy, 271, 126968.

Dou, J., Yunus, A.P., Tien Bui, D., Merghadi, A., Sahana, M., Zhu, Z., Chen, C.-W., Khosravi, K., Yang, Y. & Pham, B.T. (2019). Assessment of advanced random forest and decision tree algorithms for modeling rainfall-induced landslide susceptibility in the Izu-Oshima Volcanic Island, Japan. Science of The Total Environment, 662, 332-346.

Drosou, K. & Koukouvinos, C. (2017). Proximal support vector machine techniques on medical prediction outcome. Journal of Applied Statistics, 44(3), 533-553.

Dzakpata, I., Knights, P.F., Kizil, M., Nehring, M. & Aminossadati, S.M. (2016). Truck and shovel versus in-pit conveyor systems: A comparison of the valuable operating time. In: 2016 Coal Operators' Conference (pp. 463-476). Wollongong, Australia: The University of Wollongong.

Eker, S., Garcia, D., Valin, H. & van Ruijven, B. (2021). Using social media audience data to analyse the drivers of low-carbon diets. Environmental Research Letters, 16(7), 074001.

Enayatollahi, I., Aghajani Bazzazi, A. & Asadi, A. (2014). Comparison between neural networks and multiple regression analysis to predict rock fragmentation in open-pit mines. Rock Mechanics and Rock Engineering, 47(2), 799-807.

Ercelebi, S.G. & Bascetin, A. (2009). Optimization of shovel-truck system for surface mining. Journal of the Southern African Institute of Mining and Metallurgy, 109, 433-439.

Erdal, H.I. (2013). Two-level and hybrid ensembles of decision trees for high performance concrete compressive strength prediction. Engineering Applications of Artificial Intelligence, 26(7), 1689-1697.

Erdogan Erten, G., Bozkurt Keser, S. & Yavuz, M. (2021). Grid search optimised artificial neural network for open stope stability prediction. International Journal of Mining, Reclamation and Environment, 35(8), 600-617.

Ewees, A.A., Elaziz, M.A., Alameer, Z., Ye, H. & Jianhua, Z. (2020). Improving multilayer perceptron neural network using chaotic grasshopper optimization algorithm to forecast iron ore price volatility. Resources Policy, 65, 101555.

Fan, C., Li, Q., Ma, J. & Yang, D. (2019). Fiber Bragg grating-based experimental and numerical investigations of CO_2 migration front in saturated sandstone under subcritical and supercritical conditions. Greenhouse Gases: Science and Technology, 9(1), 106-124.

Fan, C., Zhang, N., Jiang, B. & Liu, W.V. (2022). Preprocessing large datasets using Gaussian mixture modelling to improve prediction accuracy of truck productivity at mine sites. Archives of Mining Sciences, 67(4), 661-680.

Fan, C., Zhang, N., Jiang, B. & Liu, W.V. (2023a). Improved extreme machine learning for rapid estimation of mining truck cycle time based on feature selection and unsupervised clustering techniques. Expert Systems with Applications (Under review).

Fan, C., Zhang, N., Jiang, B. & Liu, W.V. (2023b). Prediction of truck productivity at mine sites using tree-based ensemble models combined with Gaussian mixture modelling. International Journal of Mining, Reclamation and Environment, 37(1), 66-86.

Fan, C., Zhang, N., Jiang, B. & Liu, W.V. (2023c). Using deep neural networks coupled with principal component analysis for ore production forecasting at open-pit mines. Journal of Rock Mechanics and Geotechnical Engineering (In press).

Fan, C., Zhang, N., Jiang, B. & Liu, W.V. (2023d). Weighted ensembles of artificial neural networks based on Gaussian mixture modeling for truck productivity prediction at open-pit mines. Mining, Metallurgy & Exploration, 40(2), 583-598.

Fei, W., Narsilio, G.A., van der Linden, J.H. & Disfani, M.M. (2020). Network analysis of heat transfer in sphere packings. Powder Technology, 362, 790-804.

Feng, D.C., Liu, Z.T., Wang, X.D., Chen, Y., Chang, J.Q., Wei, D.F. & Jiang, Z.M. (2020). Machine learning-based compressive strength prediction for concrete: An adaptive boosting approach. Construction and Building Materials, 230, 117000.

Fikret Kurnaz, T. & Kaya, Y. (2018). The comparison of the performance of ELM, BRNN, and SVM methods for the prediction of compression index of clays. Arabian Journal of Geosciences, 11(24), 770.

Friedman, J.H. (2001). Greedy function approximation: A gradient boosting machine. The Annals of Statistics, 29(5), 1189-1232.

Fu, Y., Liu, X., Sarkar, S. & Wu, T. (2021). Gaussian mixture model with feature selection: An embedded approach. Computers & Industrial Engineering, 152, 107000.

Ge, Z. (2008). Effectiveness of the t-test in multiple linear regression modeling of environmental systems. Environmental Engineering Science, 26(2), 377-384.

Ge, S., Gao, W., Cui, S., Chen, X. & Wang, S. (2022). Safety prediction of shield tunnel construction using deep belief network and whale optimization algorithm. Automation in Construction, 142, 104488.

Ge, F., Ju, Y., Qi, Z. & Lin, Y. (2018). Parameter estimation of a Gaussian mixture model for wind power forecast error by Riemann L-BFGS optimization. IEEE Access, 6, 38892-38899.

Geurts, P., Ernst, D. & Wehenkel, L. (2006). Extremely randomized trees. Machine Learning, 63(1), 3-42.

Ghiasi, M., Askarnejad, N., Dindarloo, S.R. & Shamsoddini, H. (2016). Prediction of blast boulders in open pit mines via multiple regression and artificial neural networks. International Journal of Mining Science and Technology, 26(2), 183-186.

Giesy, J.P., Anderson, J.C. & Wiseman, S.B. (2010). Alberta oil sands development. Proceedings of the National Academy of Sciences, 107(3), 951.

Glória, L.S., Cruz, C.D., Vieira, R.A.M., de Resende, M.D.V., Lopes, P.S., de Siqueira, O.H.G.B.D. & Fonseca e Silva, F. (2016). Accessing marker effects and heritability estimates from genome prediction by Bayesian regularized neural networks. Livestock Science, 191, 91-96.

Goel, S., Guleria, K. & Panda, S.N. (2022). Anomaly based intrusion detection model using supervised machine learning techniques. In: 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO) (pp. 1-5).

Goodarzi, M., Chen, T. & Freitas, M.P. (2010). QSPR predictions of heat of fusion of organic compounds using Bayesian regularized artificial neural networks. Chemometrics and Intelligent Laboratory Systems, 104(2), 260-264.

Groemping, U. (2006). Relative importance for linear regression in R: The package relaimpo. Journal of Statistical Software, 1(1).

Grün, B. & Leisch, F. (2007). Fitting finite mixtures of generalized linear regressions in R. Computational Statistics & Data Analysis, 51(11), 5247-5252.

Gu, Q., Lu, C., Guo, J. & Jing, S. (2010). Dynamic management system of ore blending in an open pit mine based on GIS/GPS/GPRS. Mining Science and Technology (China), 20(1), 132-137.

Gui, Y., Tao, Z., Wang, C. & Xie, X. (2011). Study on remote monitoring system for landslide hazard based on wireless sensor network and its application. Journal of Coal Science and Engineering (China), 17(4), 464-468.

Guo, C., Liu, M. & Lu, M. (2021). A dynamic ensemble learning algorithm based on k-means for ICU mortality prediction. Applied Soft Computing, 103, 107166.

Hasanipanah, M., Shahnazar, A., Bakhshandeh Amnieh, H. & Jahed Armaghani, D. (2017). Prediction of air-overpressure caused by mine blasting using a new hybrid PSO-SVR model. Engineering with Computers, 33(1), 23-31.

Hinton, G.E. & Salakhutdinov, R.R. (2006). Reducing the Dimensionality of Data with Neural Networks. Science, 313(5786), 504.

Ho Park, M., Ju, M., Jeong, S. & Young Kim, J. (2021). Incorporating interaction terms in multivariate linear regression for post-event flood waste estimation. Waste Management, 124, 377-384.

Honarvar, A., Rozhon, J., Millington, D., Walden, T., Murillo, C.A. & Walden, Z. (2011a). Economic Impacts of New Oil Sands Projects in Alberta (2010-2035). In. Calgary, Alberta: Canadian Energy Research Institute.

Honarvar, A., Rozhon, J., Millington, D., Walden, T., Murillo, C.A. & Walden, Z. (2011b). Economics impacts of new oil sands projects in Alberta (2010-2035). In: (Study No. 124 ed.).

Huang & Xue, J. (2022). Optimization of SVR functions for flyrock evaluation in mine blasting operations. Environmental Earth Sciences, 81(17), 434.

Huang, Zhu, Q. & Siew, C.K. (2006). Extreme learning machine: Theory and applications. Neurocomputing, 70(1), 489-501.

Hubel, D.H. & Wiesel, T.N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. J Physiol, 160(1), 106-154.

Huo, W., Li, W., Zhang, Z., Sun, C., Zhou, F. & Gong, G. (2021). Performance prediction of proton-exchange membrane fuel cell based on convolutional neural network and random forest feature selection. Energy Conversion and Management, 243, 114367.

Hyder, Z., Siau, K. & Nah, F. (2019). Artificial intelligence, machine learning, and autonomous technologies in mining industry. Journal of Database Management (JDM), 30(2), 67-79.

Iqbal, K. & Sun, D. (2014). Development of thermo-regulating polypropylene fibre containing microencapsulated phase change materials. Renewable Energy, 71, 473-479.

Jaccard, J., Turrisi, R. & Jaccard, J. (2003). Interaction Effects in Multiple Regression. Thousand Qaks, CA: Sage.

Janizadeh, S., Vafakhah, M., Kapelan, Z. & Mobarghaee Dinan, N. (2022). Hybrid XGboost model with various Bayesian hyperparameter optimization algorithms for flood hazard susceptibility modeling. Geocarto International, 37(25), 8273-8292.

Ji, Z., Xia, Y., Sun, Q., Chen, Q. & Feng, D. (2014). Adaptive scale fuzzy local Gaussian mixture model for brain MR image segmentation. Neurocomputing, 134, 60-69.

Jing, Y., Zhang, L., Hao, W. & Huang, L. (2022). Numerical study of a CNN-based model for regional wave prediction. Ocean Engineering, 255, 111400.

Jordan, M.I. & Mitchell, T.M. (2015). Machine learning: Trends, perspectives, and prospects. Science, 349(6245), 255-260.

Jun, M.A. & Cheng, J.C.P. (2017). Selection of target LEED credits based on project information and climatic factors using data mining techniques. Advanced Engineering Informatics, 32, 224-236.

Jung, D. & Choi, Y. (2021). Systematic review of machine learning applications in mining: Exploration, exploitation, and reclamation. Minerals, 11(2), 148.

Kaplan, U.E., Dagasan, Y. & Topal, E. (2021). Mineral grade estimation using gradient boosting regression trees. International Journal of Mining, Reclamation and Environment, 35(10), 728-742.

Katta, A.K., Davis, M., Subramanyam, V., Dar, A.F., Mondal, M.A.H., Ahiduzzaman, M. & Kumar, A. (2019). Assessment of energy demand-based greenhouse gas mitigation options for Canada's oil sands. Journal of Cleaner Production, 241, 118306.

Khambra, G. & Shukla, P. (2023). Novel machine learning applications on fly ash based concrete: An overview. Materials Today: Proceedings, 80, 3411-3417.

Kocheturov, A., Pardalos, P.M. & Karakitsiou, A. (2019). Massive datasets and machine learning for computational biomedicine: trends and challenges. Annals of Operations Research, 276(1), 5-34.

Kramer, M.A. (1991). Nonlinear principal component analysis using autoassociative neural networks. AIChE Journal, 37(2), 233-243.

Krzywinski, M. & Altman, N. (2017). Classification and regression trees. Nature Methods, 14(8), 757-758.

Kueh, A.B.H. (2021). Artificial neural network and regressed beam-column connection explicit mathematical moment-rotation expressions. Journal of Building Engineering, 43, 103195.

Kuyuk, H.S., Yildirim, E., Dogan, E. & Horasan, G. (2012). Application of k-means and Gaussian mixture model for classification of seismic activities in Istanbul. Nonlin. Processes Geophys., 19(4), 411-419.

Kyburz, D., Gabay, C., Michel, B.A., Finckh, A. & for the physicians of the, S.-R. (2011). The long-term impact of early treatment of rheumatoid arthritis on radiographic progression: a population-based cohort study. Rheumatology, 50(6), 1106-1110.

LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W. & Jackel, L.D. (1989). Backpropagation Applied to Handwritten Zip Code Recognition. Neural Computation, 1(4), 541-551.

Lee, S. & Park, I. (2013). Application of decision tree model for the ground subsidence hazard mapping near abandoned underground coal mines. Journal of Environmental Management, 127, 166-176.

Lei, C., Deng, J., Cao, K., Ma, L., Xiao, Y. & Ren, L. (2018). A random forest approach for predicting coal spontaneous combustion. Fuel, 223, 63-73.

Leisch, F. (2004). FlexMix: A general framework for finite mixture models and latent class regression in R. Journal of Statistical Software, 1(8).

Lessard, J., de Bakker, J. & McHugh, L. (2014). Development of ore sorting and its impact on mineral processing economics. Minerals Engineering, 65, 88-97.

Li, Z. (2022). Extracting spatial effects from machine learning model using local interpretation method: An example of SHAP and XGBoost. Computers, Environment and Urban Systems, 96, 101845.

Li, Lei, Y. & Pan, D. (2015). Economic and environmental evaluation of coal production in China and policy implications. Natural Hazards, 77(2), 1125-1141.

Li, L., Liu, Z., Armaghani, D.J., Xiao, P. & Zhou, J. (2022). Novel ensemble intelligence methodologies for rockburst assessment in complex and variable environments. Scientific Reports, 12(1), 1844.

Li, K., Ma, Z., Robinson, D. & Ma, J. (2018). Identification of typical building daily electricity usage profiles using Gaussian mixture model-based clustering and hierarchical clustering. Applied Energy, 231, 331-342.

Li, Y., Schofield, E. & Gönen, M. (2019). A tutorial on Dirichlet process mixture modeling. Journal of Mathematical Psychology, 91, 128-144.

Li, Wu, K. & Zhou, D.W. (2014). Extraction algorithm of mining subsidence information on water area based on support vector machine. Environmental Earth Sciences, 72(10), 3991-4000.

Li, L., Yang, F., Ren, M., Zhang, X., Zhou, J. & Khandelwal, M. (2021). Prediction of blasting mean fragment size using support vector regression combined with five optimization algorithms. Journal of Rock Mechanics and Geotechnical Engineering, 13(6), 1380-1397.

Liang, W., Luo, S., Zhao, G. & Wu, H. (2020). Predicting hard rock pillar stability using GBDT, XGBoost, and LightGBM algorithms. Mathematics, 8(5).

Liang, M., Mohamad, E.T., Faradonbeh, R.S., Jahed Armaghani, D. & Ghoraba, S. (2016). Rock strength assessment based on regression tree technique. Engineering with Computers, 32(2), 343-354.

Liang, H., Zou, J., Li, Z., Khan, M.J. & Lu, Y. (2019). Dynamic evaluation of drilling leakage risk based on fuzzy theory and PSO-SVR algorithm. Future Generation Computer Systems, 95, 454-466.

Liu, H., Chen, C., Lv, X., Wu, X. & Liu, M. (2019). Deterministic wind energy forecasting: A review of intelligent predictors and auxiliary methods. Energy Conversion and Management, 195, 328-345.

Liu, J., Jiang, L., Chen, Y., Liu, Z., Yuan, H. & Wen, Y. (2023). Study on prediction model of liquid hold up based on random forest algorithm. Chemical Engineering Science, 268, 118383.

Liu, Y., Luo, H., Zhao, B., xiaoyong, Z. & Han, Z. (2018). Short-term power load forecasting based on clustering and XGBoost method. In: 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS) (pp. 536-539).

Liu, Z., Luo, S., Tseng, M., Liu, H., Li, L. & Hashan Md Mashud, A. (2021). Short-term photovoltaic power prediction on modal reconstruction: A novel hybrid model approach. Sustainable Energy Technologies and Assessments, 45, 101048.

Liu, X., Tang, H., Ding, Y. & Yan, D. (2022). Investigating the performance of machine learning models combined with different feature selection methods to estimate the energy consumption of buildings. Energy and Buildings, 273, 112408.

Liu, Y., Wu, J., Wang, Z., Lu, X., Avdeev, M., Shi, S., Wang, C. & Yu, T. (2020). Predicting creep rupture life of Ni-based single crystal superalloys using divide-and-conquer approach based machine learning. Acta Materialia, 195, 454-467.

Lou, C., Liu, H., Li, Y., Peng, Y., Wang, J. & Dai, L. (2017). Relationships of relative humidity with PM2.5 and PM10 in the Yangtze River Delta, China. Environmental Monitoring and Assessment, 189(11), 582.

Lu, S., Chen, R., Wei, W., Belovsky, M. & Lu, X. (2021). Understanding heart failure patients EHR clinical features via SHAP interpretation of tree-based machine learning model predictions. AMIA Annu Symp Proc, 2021, 813-822.

Lu, Y., Tian, Z., Peng, P., Niu, J., Li, W. & Zhang, H. (2019). GMM clustering for heating load patterns in-depth identification and prediction model accuracy improvement of district heating system. Energy and Buildings, 190, 49-60.

Lubke, G.H. & Luningham, J. (2017). Fitting latent variable mixture models. Behaviour Research and Therapy, 98, 91-102.

Lundberg, S.M. & Lee, S.I. (2017). A unified approach to interpreting model predictions. In: Proceedings of the 31st International Conference on Neural Information Processing Systems (pp. 4768–4777). Long Beach, California, USA: Curran Associates Inc.

Lunt, M. (2015). Introduction to statistical modelling 2: categorical variables and interactions in linear regression. Rheumatology, 54(7), 1141-1144.

Ma, J., Cheng, J.C.P., Lin, C., Tan, Y. & Zhang, J. (2019). Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques. Atmospheric Environment, 214, 116885.

Ma, S., Fan, C. & Liu, W.V. (2023). Effects of site operating conditions on real site TKPH (tonne-kilometer-per-hour) of ultra-large off-the-road tires. Part D: Journal of Automobile Engineering, (In press).

Ma, S., Huang, G., Obaia, K., Moon, S.W. & Liu, W.V. (2021). Hysteresis loss of ultra-large off-the-road tire rubber compounds based on operating conditions at mine sites. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, 236(2-3), 439-450.

Ma, Z., Li, H., Sun, Q., Wang, C., Yan, A. & Starfelt, F. (2014). Statistical analysis of energy consumption patterns on the heat demand of buildings in district heating systems. Energy and Buildings, 85, 464-472.

Ma, S., Wu, L. & Liu, W.V. (2022). Numerical investigation of temperatures in ultra-large off-the-road tires under operating conditions at mine sites. Journal of Thermal Science and Engineering Applications, 15(2).

Maaouane, M., Zouggar, S., Krajačić, G. & Zahboune, H. (2021). Modelling industry energy demand using multiple linear regression analysis based on consumed quantity of goods. Energy, 225, 120270.

MacKay, D.J. (1992). Bayesian interpolation. Neural computation, 4(3), 415-447.

Mangalathu, S., Karthikeyan, K., Feng, D.C. & Jeon, J.S. (2022). Machine-learning interpretability techniques for seismic performance assessment of infrastructure systems. Engineering Structures, 250, 112883.

MANOBS. (2021). MANOBS-manual of surface weather observation standards. In. Gatineau, Quebec: Government of Canada.

Marugán, A.P. (2023). Applications of reinforcement learning for maintenance of engineering systems: A review. Advances in Engineering Software, 183, 103487.

Masud, B., Cui, Q., Ammar, M.E., Bonsal, B.R., Islam, Z. & Faramarzi, M. (2021). Means and extremes: Evaluation of a CMIP6 multi-model ensemble in reproducing historical climate characteristics across Alberta, Canada. Water, 13(5), 737.

McCulloch, W.S. & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, 5(4), 115-133.

McLachlan, G.J., Lee, S.X. & Rathnayake, S.I. (2019). Finite mixture models. Annual Review of Statistics and Its Application, 6(1), 355-378.

Medinac, F., Bamford, T., Hart, M., Kowalczyk, M. & Esmaeili, K. (2020). Haul road monitoring in open pit mines using unmanned aerial vehicles: A case study at Bald Mountain Mine Site. Mining, Metallurgy & Exploration, 37(6), 1877-1883.

Mehrjou, A., Hosseini, R. & Nadjar Araabi, B. (2016). Improved Bayesian information criterion for mixture model selection. Pattern Recognition Letters, 69, 22-27.

MEP. (2023). Current and historical Alberta weather station data viewer. In. Edmonton, Canada: Ministry of Environment and Parks (MEP), Government of Alberta.

Milad, A., Hussein, S.H., Khekan, A.R., Rashid, M., Al-Msari, H. & Tran, T.H. (2022). Development of ensemble machine learning approaches for designing fiber-reinforced polymer composite strain prediction model. Engineering with Computers, 38(4), 3625-3637.

Mittlböck, M. (2002). Calculating adjusted R2 measures for Poisson regression models. Computer Methods and Programs in Biomedicine, 68(3), 205-214.

Moayedi, H., Mehrabi, M., Mosallanezhad, M., Rashid, A.S.A. & Pradhan, B. (2019). Modification of landslide susceptibility mapping using optimized PSO-ANN technique. Engineering with Computers, 35(3), 967-984.

Modenese, A., Korpinen, L. & Gobba, F. (2018). Solar radiation exposure and outdoor work: An underestimated occupational risk. International Journal of Environmental Research and Public Health, 15(10), 2063.

Mohammed, H.R.M. & Ismail, S. (2022). Proposition of new computer artificial intelligence models for shear strength prediction of reinforced concrete beams. Engineering with Computers, 38(4), 3739-3757.

Moore, D.A. (2021). Python GUI programming with Tkinter: Design and build functional and user-friendly GUI applications. Birmingham, UK: Packt Publishing Ltd.

Moradi Afrapoli, A., Tabesh, M. & Askari-Nasab, H. (2019). A multiple objective transportation problem approach to dynamic truck dispatching in surface mines. European Journal of Operational Research, 276(1), 331-342.

Mouloodi, S., Rahmanpanah, H., Gohari, S., Burvill, C. & Davies, H.M.S. (2022). Feedforward backpropagation artificial neural networks for predicting mechanical responses in complex nonlinear structures: A study on a long bone. Journal of the Mechanical Behavior of Biomedical Materials, 128, 105079.

Mouselimis, L., Gosso, A. & de Jonge, E. (2022). Package ‘elmNNcpp’. In: (pp. 9): CRAN.

Moy, R.L., Chen, L.S. & Kao, L.J. (2015). Multiple Linear Regression. In: R.L. Moy, L.S. Chen & L.J. Kao, Study Guide for Statistics for Business and Financial Economics: A Supplement to the Textbook by Cheng-Few Lee, John C. Lee and Alice C. Lee (pp. 223-240). Cham: Springer International Publishing.

Naghibi, S.A., Ahmadi, K. & Daneshi, A. (2017). Application of support vector machine, random forest, and genetic algorithm optimized random forest models in groundwater potential mapping. Water Resources Management, 31(9), 2761-2775.

Nasir Amin, M., Iftikhar, B., Khan, K., Faisal Javed, M., Mohammad AbuArab, A. & Faisal Rehman, M. (2023). Prediction model for rice husk ash concrete using AI approach: Boosting and bagging algorithms. Structures, 50, 745-757.

Navarro Torres, V.F., Ayres, J., Carmo, P.L.A. & Silveira, C.G.L. (2019). Haul productivity optimization: An assessment of the optimal road grade. In: E. Widzyk-Capehart, A. Hekmat & R. Singhal, Proceedings of the 27th International Symposium on Mine Planning and Equipment Selection - MPES 2018 (pp. 345-353). Cham: Springer International Publishing.

Nguyen, Bui, X.-N., Bui, H.-B. & Mai, N.-L. (2020). A comparative study of artificial neural networks in predicting blast-induced air-blast overpressure at Deo Nai open-pit coal mine, Vietnam. Neural Computing and Applications, 32(8), 3939-3955.

Nguyen, Ly, H.-B., Ho, L.S., Al-Ansari, N., Le, H.V., Tran, V.Q., Prakash, I. & Pham, B.T. (2021). Influence of data splitting on performance of machine learning models in prediction of shear strength of soil. Mathematical Problems in Engineering, 2021, 4832864.

Ni, L., Wang, D., Wu, J., Wang, Y., Tao, Y., Zhang, J. & Liu, J. (2020). Streamflow forecasting using extreme gradient boosting model coupled with Gaussian mixture model. Journal of Hydrology, 586, 124901.

Nourali, H. & Osanloo, M. (2019). Mining capital cost estimation using Support Vector Regression (SVR). Resources Policy, 62, 527-540.

Nourali, H. & Osanloo, M. (2020). A regression-tree-based model for mining capital cost estimation. International Journal of Mining, Reclamation and Environment, 34(2), 88-100.

Nourani, V., Molajou, A., Tajbakhsh, A.D. & Najafi, H. (2019). A wavelet based data mining technique for suspended sediment load modeling. Water Resources Management, 33(5), 1769-1784.

Obaia, K. (2020). Current truck productivity curve at Mildred Lake operations (Email communication).

Ohadi, B., Sun, X., Esmaieli, K. & Consens, M.P. (2020). Predicting blast-induced outcomes using random forest models of multi-year blasting data from an open pit mine. Bulletin of Engineering Geology and the Environment, 79(1), 329-343.

Olu-Ajayi, R., Alaka, H., Sulaimon, I., Balogun, H., Wusu, G., Yusuf, W. & Adegoke, M. (2023). Building energy performance prediction: A reliability analysis and evaluation of feature selection methods. Expert Systems with Applications, 225, 120109.

Onyekwena, C.C., Xue, Q., Li, Q., Wan, Y., Feng, S., Umeobi, H.I., Liu, H. & Chen, B. (2022). Support vector machine regression to predict gas diffusion coefficient of biochar-amended soil. Applied Soft Computing, 127, 109345.

OSM. (2021). Evolution of Mining Equipment in the Oil Sands. In: Oil Sands Magazine (OSM).

Paliwal, M. & Kumar, U.A. (2011). Assessing the contribution of variables in feed forward neural network. Applied Soft Computing, 11(4), 3690-3696.

Pan, Z., Meng, Z., Chen, Z., Gao, W. & Shi, Y. (2020). A two-stage method based on extreme learning machine for predicting the remaining useful life of rolling-element bearings. Mechanical Systems and Signal Processing, 144, 106899.

Pao, H.T. (2008). A comparison of neural network and multiple regression analysis in modeling capital structure. Expert Systems with Applications, 35(3), 720-727.

Parsons, O.E. (2020). A Gaussian mixture model approach to classifying response types. In: N. Bouguila & W. Fan, Mixture Models and Applications (pp. 3-22). Cham: Springer International Publishing.

Patil, K., Nagwani, N.K. & Tripathi, S. (2018). A parametric study of partitioning and density based clustering techniques for boxplot generation. In: 2018 3rd International Conference for Convergence in Technology (I2CT) (pp. 1-5).

Perai, A.H., Nassiri Moghaddam, H., Asadpour, S., Bahrampour, J. & Mansoori, G. (2010). A comparison of artificial neural networks with other statistical approaches for the prediction of true metabolizable energy of meat and bone meal. Poultry Science, 89(7), 1562-1568.

Peters, J., Baets, B.D., Verhoest, N.E.C., Samson, R., Degroeve, S., Becker, P.D. & Huybrechts, W. (2007). Random forests as a tool for ecohydrological distribution modelling. Ecological Modelling, 207(2), 304-318.

Piñeiro, G., Perelman, S., Guerschman, J.P. & Paruelo, J.M. (2008). How to evaluate models: Observed vs. predicted or predicted vs. observed? Ecological Modelling, 216(3), 316-322.

Potočnik, P., Vidrih, B., Kitanovski, A. & Govekar, E. (2019). Neural network, ARX, and extreme learning machine models for the short-term prediction of temperature in buildings. Building Simulation, 12(6), 1077-1093.

Pu, Y., Apel, D.B. & Lingga, B. (2018). Rockburst prediction in kimberlite using decision tree with incomplete data. Journal of Sustainable Mining, 17(3), 158-165.

Pu, Y., Apel, D.B., Liu, V. & Mitri, H. (2019). Machine learning methods for rockburst prediction-state-of-the-art review. International Journal of Mining Science and Technology, 29(4), 565-570.

Qi, C. & Tang, X. (2018). Slope stability prediction using integrated metaheuristic and machine learning approaches: A comparative study. Computers & Industrial Engineering, 118, 112-122.

Qiu, Y., Zhou, J., Khandelwal, M., Yang, H., Yang, P. & Li, C. (2022). Performance evaluation of hybrid WOA-XGBoost, GWO-XGBoost and BO-XGBoost models to predict blast-induced ground vibration. Engineering with Computers, 38(5), 4145-4162.

Ramhormozi, R.S., Mozhdehi, A., Kalantari, S., Wang, Y., Sun, S. & Wang, X. (2022). Multi-task graph neural network for truck speed prediction under extreme weather conditions. In: Proceedings of the 30th International Conference on Advances in Geographic Information Systems (pp. Article 93). Seattle, Washington: Association for Computing Machinery.

Rana, A., Bhagat, N.K., Jadaun, G.P., Rukhaiyar, S., Pain, A. & Singh, P.K. (2020). Predicting blast-induced ground vibrations in some Indian tunnels: A comparison of decision tree, artificial neural network and multivariate regression methods. Mining, Metallurgy & Exploration, 37(4), 1039-1053.

Ransom, K.M., Nolan, B.T., Stackelberg, P.E., Belitz, K. & Fram, M.S. (2022). Machine learning predictions of nitrate in groundwater used for drinking supply in the conterminous United States. Science of The Total Environment, 807, 151065.

Ribeiro, M.H.D.M. & dos Santos Coelho, L. (2020). Ensemble approach based on bagging, boosting and stacking for short-term prediction in agribusiness time series. Applied Soft Computing, 86, 105837.

Rice, J.A. (1995). Mathematical Statistics and Data Analysis (2nd ed.). Belmont, CA.: Duxbury Press.

Rimélé, A., Dimitrakopoulos, R. & Gamache, M. (2020). A dynamic stochastic programming approach for open-pit mine planning with geological and commodity price uncertainty. Resources Policy, 65, 101570.

Ripley, B. & Venables, W. (2022). Package ‘nnet’. In: (pp. 11): CRAN.

Rodriguez, P.P. & Gianola, D. (2021). Package ‘brnn’. In: (pp. 23): CRAN.

Rodriguez-Galiano, V., Sanchez-Castillo, M., Chica-Olmo, M. & Chica-Rivas, M. (2015). Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines. Ore Geology Reviews, 71, 804-818.

Roh, H.J., Sahu Prasanta, K., Sharma, S., Datla, S. & Mehran, B. (2016). Statistical investigations of snowfall and temperature interaction with passenger car and truck traffic on primary highways in Canada. Journal of Cold Regions Engineering, 30(2), 04015006.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review, 65(6), 386-408.

Rumelhart, D.E., Hinton, G.E. & Williams, R.J. (1986). Learning internal representations by error propagation. In: Parallel distributed processing: explorations in the microstructure of cognition, vol. 1: foundations (pp. 318-362): MIT Press.

Russell, J.S. & Raftery, A.E. (2009). Performance of Bayesian model selection criteria for Gaussian mixture models. In: Despartment of Statistics, University of Washington.

Saadat, M., Khandelwal, M. & Monjezi, M. (2014). An ANN-based approach to predict blast-induced ground vibration of Gol-E-Gohar iron ore mine, Iran. Journal of Rock Mechanics and Geotechnical Engineering, 6(1), 67-76.

Sabniveesu, V., Kavuri, A., Kavi, R., Kulathumani, V., Kecojevic, V. & Nimbarte, A. (2015). Use of wireless, ad-hoc networks for proximity warning and collision avoidance in surface mines. International Journal of Mining, Reclamation and Environment, 29(5), 331-346.

Saeed, U., Jan, S.U., Lee, Y.D. & Koo, I. (2021). Fault diagnosis based on extremely randomized trees in wireless sensor networks. Reliability Engineering & System Safety, 205, 107284.

Sagberg, F., Selpi, Bianchi Piccinini, G.F. & Engström, J. (2015). A review of research on driving styles and road safety. Human Factors, 57(7), 1248-1275.

Sahari Moghaddam, A., Rezazadeh Azar, E., Mejias, Y. & Bell, H. (2020). Estimating stripping of asphalt coating using k-means clustering and machine learning–based classification. Journal of Computing in Civil Engineering, 34(1), 04019044.

Saini, L.M. (2008). Peak load forecasting using Bayesian regularization, Resilient and adaptive backpropagation learning based artificial neural networks. Electric Power Systems Research, 78, 1302-1310.

Santos, A., Figueiredo, E., Silva, M., Santos, R., Sales, C. & Costa, J.C.W.A. (2017). Genetic-based EM algorithm to improve the robustness of Gaussian mixture models for damage detection in bridges. Structural Control and Health Monitoring, 24(3), e1886.

Sattar, A.M.A., Ertuğrul, Ö.F., Gharabaghi, B., McBean, E.A. & Cao, J. (2019). Extreme learning machine model for water network management. Neural Computing and Applications, 31(1), 157-169.

Schexnayder, C., Weber, S.L. & Brooks, B.T. (1999). Effect of truck payload weight on production. Journal of Construction Engineering and Management, 125(1), 1-7.

Sekhar Roy, S., Roy, R. & Balas, V.E. (2018). Estimating heating load in buildings using multivariate adaptive regression splines, extreme learning machine, a hybrid model of MARS and ELM. Renewable and Sustainable Energy Reviews, 82, 4256-4268.

Sembakutti, D., Kumral, M. & Sasmito, A. (2017). Analysing equipment allocation through queuing theory and Monte-Carlo simulations in surface mining operations. International Journal of Mining and Mineral Engineering, 8, 56.

Shahin, M.A., Maier, H.R. & Jaksa, M.B. (2004). Data division for developing Nneural networks applied to geotechnical engineering. Journal of Computing in Civil Engineering, 18(2), 105-114.

Sheikhan, M., Bejani, M. & Gharavian, D. (2013). Modular neural-SVM scheme for speech emotion recognition using ANOVA feature selection method. Neural Computing and Applications, 23(1), 215-227.

Shi, J., Zhu, Y., Khan, F. & Chen, G. (2019). Application of Bayesian regularization artificial neural network in explosion risk analysis of fixed offshore platform. Journal of Loss Prevention in the Process Industries, 57, 131-141.

Shimizu, N. & Kaneko, H. (2020). Direct inverse analysis based on Gaussian mixture regression for multiple objective variables in material design. Materials & Design, 196, 109168.

Shirkhorshidi, A.S., Aghabozorgi, S., Wah, T.Y. & Herawan, T. (2014). Big data clustering: A review. In: B. Murgante, S. Misra, A.M.A.C. Rocha, C. Torre, J.G. Rocha, M.I. Falcão, D. Taniar, B.O. Apduhan & O. Gervasi, Computational Science and Its Applications - ICCSA 2014 (pp. 707-720). Cham: Springer International Publishing.

Siami-Irdemoosa, E. & Dindarloo, S.R. (2015). Prediction of fuel consumption of mining dump trucks: A neural networks approach. Applied Energy, 151, 77-84.

Silion, Ş. & Foşalău, C. (2014). Wet road surfaces detection by measuring the air humidity in two points. In: 2014 International Conference and Exposition on Electrical and Power Engineering (EPE) (pp. 744-747).

Simsekler, M.C.E., Rodrigues, C., Qazi, A., Ellahham, S. & Ozonoff, A. (2021). A comparative study of patient and staff safety evaluation using tree-based machine learning algorithms. Reliability Engineering & System Safety, 208, 107416.

Sinaga, K.P. & Yang, M.S. (2020). Unsupervised K-means clustering algorithm. IEEE Access, 8, 80716-80727.

Singh, S. & Yassine, A. (2018). Big data mining of energy time series for behavioral analytics and energy consumption forecasting. Energies, 11(2), 452.

Sleep, S., Laurenzi, I.J., Bergerson, J.A. & MacLean, H.L. (2018). Evaluation of variability in greenhouse gas intensity of Canadian oil sands surface mining and upgrading operations. Environmental Science & Technology, 52(20), 11941-11951.

Smith, S.D., Wood, G.S. & Gould, M. (2000). A new earthworks estimating methodology. Construction Management and Economics, 18(2), 219-228.

Song, S., Marks, E. & Pradhananga, N. (2017). Impact variables of dump truck cycle time for heavy excavation construction projects. Journal of Construction Engineering and Project Management, 7(2), 11-18.

Sonta, A.J., Simmons, P.E. & Jain, R.K. (2018). Understanding building occupant activities at scale: An integrated knowledge-based and data-driven approach. Advanced Engineering Informatics, 37, 1-13.

Soofastaei, A., Aminossadati, S.M., Kizil, M.S. & Knights, P. (2016). A discrete-event model to simulate the effect of truck bunching due to payload variance on cycle time, hauled mine materials and fuel consumption. International Journal of Mining Science and Technology, 26(5), 745-752.

Stringham, G. (2012). Chapter 2 - Energy Developments in Canada's Oil Sands. Developments in Environmental Science, 11, 19-34.

Su, J., Wang, Y., Niu, X., Sha, S. & Yu, J. (2022). Prediction of ground surface settlement by shield tunneling using XGBoost and Bayesian Optimization. Engineering Applications of Artificial Intelligence, 114, 105020.

Sun, Y., Li, G., Zhang, N., Chang, Q., Xu, J. & Zhang, J. (2021). Development of ensemble learning models to evaluate the strength of coal-grout materials. International Journal of Mining Science and Technology, 31(2), 153-162.

Sun, X., Zhang, H., Tian, F. & Yang, L. (2018). The use of a machine learning method to predict the real-time link travel time of open-pit trucks. Mathematical Problems in Engineering, 2018, 4368045.

Svenson, G. & Fjeld, D. (2017). The impact of road geometry, surface roughness and truck weight on operating speed of logging trucks. Scandinavian Journal of Forest Research, 32(6), 515-527.

Tadeusiewicz, R. (2015). Neural networks in mining sciences - general overview and some representative examples. Archives of Mining Sciences, 60(4), 971-984.

Tan, Q., Wei, Y., Wang, M. & Liu, Y. (2014). A cluster multivariate statistical method for environmental quality management. Engineering Applications of Artificial Intelligence, 32, 1-9.

Thai, D.K., Tu, T.M., Bui, T.Q. & Bui, T.T. (2021). Gradient tree boosting machine learning on predicting the failure modes of the RC panels under impact loads. Engineering with Computers, 37(1), 597-608.

Tian, W., Liu, Y., Heo, Y., Yan, D., Li, Z., An, J. & Yang, S. (2016). Relative importance of factors influencing building energy in urban environment. Energy, 111, 237-250.

Tibshirani, R., Walther, G. & Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 63(2), 411-423.

Ticknor, J.L. (2013). A Bayesian regularized artificial neural network for stock market forecasting. Expert Systems with Applications, 40(14), 5501-5506.

Tien Bui, D., Hoang, N.-D. & Samui, P. (2019). Spatial pattern analysis and prediction of forest fire using new machine learning approach of Multivariate Adaptive Regression Splines and Differential Flower Pollination optimization: A case study at Lao Cai province (Viet Nam). Journal of Environmental Management, 237, 476-487.

Trivedi, R., Singh, T.N. & Raina, A.K. (2014). Prediction of blast-induced flyrock in Indian limestone mines using neural networks. Journal of Rock Mechanics and Geotechnical Engineering, 6(5), 447-454.

Tsanas, A. & Xifara, A. (2012). Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. Energy and Buildings, 49, 560-567.

Tu, L., Lv, Y., Zhang, Y. & Cao, X. (2021). Logistics service provider selection decision making for healthcare industry based on a novel weighted density-based hierarchical clustering. Advanced Engineering Informatics, 48, 101301.

Upadhyay, S., Tabesh, M., Badiozamani, M. & Askari-Nasab, H. (2020). A simulation model for estimation of mine haulage fleet productivity. In: E. Topal, Proceedings of the 28th International Symposium on Mine Planning and Equipment Selection - MPES 2019 (pp. 42-50). Cham: Springer International Publishing.

Usama, M., Qadir, J., Raza, A., Arif, H., Yau, K.l.A., Elkhatib, Y., Hussain, A. & Al-Fuqaha, A. (2019). Unsupervised machine learning for networking: techniques, applications and research challenges. IEEE Access, 7, 65579-65615.

Vapnik, V.N. & Lerner, A.Y. (1963). Recognition of patterns with help of generalized portraits. Avtomat. i Telemekh, 24(6), 774-780.

Virupakshappa, K. & Oruklu, E. (2019). Unsupervised machine learning for ultrasonic flaw detection using Gaussian mixture modeling, K-means clustering and mean shift clustering. In: 2019 IEEE International Ultrasonics Symposium (IUS) (pp. 647-649).

Vitale, M., Proietti, C., Cionni, I., Fischer, R. & De Marco, A. (2014). Random forests analysis: A useful tool for defining the relative importance of environmental conditions on crown defoliation. Water, Air, & Soil Pollution, 225(6), 1992.

Wackerly, D., Mendenhall, W. & Scheaffer, R.L. (2014). Mathematical statistics with applications: Cengage Learning.

Wang X. & Hamilton, H.J. (2005). A comparative study of two density-based spatial clustering algorithms for very large datasets. In: B. Kégl & G. Lapalme, Advances in Artificial Intelligence (pp. 120-132). Berlin, Heidelberg: Springer Berlin Heidelberg.

Wang, J., Lu, S., Wang, S. & Zhang, Y. (2021). A review on extreme learning machine. Multimedia Tools and Applications, 81, 41611-41660.

Wang, C., Peng, G. & De Baets, B. (2022). Embedding metric learning into an extreme learning machine for scene recognition. Expert Systems with Applications, 203, 117505.

Webb, B.W., Clack, P.D. & Walling, D.E. (2003). Water-air temperature relationships in a Devon river system and the role of flow. Hydrological Processes, 17(15), 3069-3084.

Wei, L. (1990). Empirical Bayes test of regression coefficient in a multiple linear regression model. Acta Mathematicae Applicatae Sinica, 6(3), 251-262.

Wen, Y., Liu, X., Xin, Q., Wu, J., Xu, X., Pei, F., Li, X., Du, G., Cai, Y., Lin, K., Yang, J. & Wang, Y. (2019). Cumulative effects of climatic factors on terrestrial vegetation growth. Journal of Geophysical Research: Biogeosciences, 124(4), 789-806.

Wheeler, D. & Wilkinson, C. (2004). From calm to storm: The origins of the Beaufort Wind Scale. The Mariner's Mirror, 90(2), 187-201.

Wu, C.L., Chau, K.W. & Li, Y.S. (2009). Predicting monthly streamflow using data-driven models coupled with data-preprocessing techniques. Water Resources Research, 45(8).

Wu, L., Hu, C. & Liu, W.V. (2020). Forecasting the deterioration of cement-based mixtures under sulfuric acid attack using support vector regression based on Bayesian optimization. SN Applied Sciences, 2(12), 1970.

Xie, X., Wu, T., Zhu, M., Jiang, G., Xu, Y., Wang, X. & Pu, L. (2021). Comparison of random forest and multiple linear regression models for estimation of soil extracellular enzyme activities in agricultural reclaimed coastal saline land. Ecological Indicators, 120, 106925.

Xing, F., Huang, H., Zhan, Z., Zhai, X., Ou, C., Sze, N.N. & Hon, K.K. (2019). Hourly associations between weather factors and traffic crashes: Non-linear and lag effects. Analytic Methods in Accident Research, 24, 100109.

Xu, Z. & Saleh, J.H. (2021). Machine learning for reliability engineering and safety applications: Review of current status and future opportunities. Reliability Engineering & System Safety, 211, 107530.

Xue, Bai, C., Qiu, D., Kong, F. & Li, Z. (2020). Predicting rockburst with database using particle swarm optimization and extreme learning machine. Tunnelling and Underground Space Technology, 98, 103287.

Xue, Y., Liu, Y., Xiong, Y., Liu, Y., Cui, X. & Lei, G. (2021). A data-driven shale gas production forecasting method based on the multi-objective random forest regression. Journal of Petroleum Science and Engineering, 196, 107801.

Yang, J., Ning, C., Deb, C., Zhang, F., Cheong, D., Lee, S.E., Sekhar, C. & Tham, K.W. (2017). K-shape clustering algorithm for building energy usage patterns analysis and forecasting model accuracy improvement. Energy and Buildings, 146, 27-37.

Ye, L., Zhang, Y., Zhang, C., Lu, P., Zhao, Y. & He, B. (2019). Combined Gaussian mixture model and cumulants for probabilistic power flow calculation of integrated wind power network. Computers & Electrical Engineering, 74, 117-129.

Yin, X., Liu, Q., Huang, X. & Pan, Y. (2022). Perception model of surrounding rock geological conditions based on TBM operational big data and combined unsupervised-supervised learning. Tunnelling and Underground Space Technology, 120, 104285.

Yiu, C.Y., Ng, K.K.H., Li, X., Zhang, X., Li, Q., Lam, H.S. & Chong, M.H. (2022). Towards safe and collaborative aerodrome operations: Assessing shared situational awareness for adverse weather detection with EEG-enabled Bayesian neural networks. Advanced Engineering Informatics, 53, 101698.

Yu, Z., Yousaf, K., Ahmad, M., Yousaf, M., Gao, Q. & Chen, K. (2020). Efficient pyrolysis of ginkgo biloba leaf residue and pharmaceutical sludge (mixture) with high production of clean energy: Process optimization by particle swarm optimization and gradient boosting decision tree algorithm. Bioresource Technology, 304, 123020.

Yuan, Z., Huang, H., Jiang, Y. & Li, J. (2021). Hybrid deep neural networks for reservoir production prediction. Journal of Petroleum Science and Engineering, 197, 108111.

Yuval & Hsieh, W.W. (2002). The impact of time-averaging on the detectability of nonlinear empirical relations. Quarterly Journal of the Royal Meteorological Society, 128(583), 1609-1622.

Zabin, A., González, V.A., Zou, Y. & Amor, R. (2022). Applications of machine learning to BIM: A systematic literature review. Advanced Engineering Informatics, 51, 101474.

Zhang, G., Chen, C.H., Cao, X., Zhong, R.Y., Duan, X. & Li, P. (2022). Industrial Internet of Things-enabled monitoring and maintenance mechanism for fully mechanized mining equipment. Advanced Engineering Informatics, 54, 101782.

Zhang, Y., Gao, S., Cai, P., Lei, Z. & Wang, Y. (2023). Information entropy-based differential evolution with extremely randomized trees and LightGBM for protein structural class prediction. Applied Soft Computing, 136, 110064.

Zhang, K., Ji, S., Zhang, Y., Zhang, J. & Pan, R. (2018). MEMS inertial sensor for strata stability monitoring in underground mining: An experimental study. Shock and Vibration, 2018, 4895862.

Zhang, S., Li, P., Zhang, L., Li, H., Jiang, W. & Hu, Y. (2016). Modified S transform and ELM algorithms and their applications in power quality analysis. Neurocomputing, 185, 231-241.

Zhang, R., Wu, C., Goh, A.T.C., Böhlke, T. & Zhang, W. (2021). Estimation of diaphragm wall deflections for deep braced excavation in anisotropic clays using ensemble learning. Geoscience Frontiers, 12(1), 365-373.

Zhong, H., Wang, J., Jia, H., Mu, Y. & Lv, S. (2019). Vector field-based support vector regression for building energy consumption prediction. Applied Energy, 242, 403-414.

Zhou, Y., Li, S., Zhou, C. & Luo, H. (2019). Intelligent approach based on random forest for safety risk prediction of deep foundation pit in subway stations. Journal of Computing in Civil Engineering, 33(1), 05018004.

Zhou, J., Li, X. & Mitri Hani, S. (2016). Classification of rockburst in underground projects: Comparison of ten supervised learning methods. Journal of Computing in Civil Engineering, 30(5), 04016003.

Zhou, H., Zhang, J., Zhou, Y., Guo, X. & Ma, Y. (2021). A feature selection algorithm of decision tree based on feature weight. Expert Systems with Applications, 164, 113842.

Zhu, M. & Xie, J. (2023). Investigation of nearby monitoring station for hourly PM2.5 forecasting using parallel multi-input 1D-CNN-biLSTM. Expert Systems with Applications, 211, 118707.

Zou, J., Han, Y. & So, S.-S. (2009). Overview of Artificial Neural Networks. In: D.J. Livingstone, Artificial Neural Networks: Methods and Applications (pp. 14-22). Totowa, NJ: Humana Press.