

Article

A Multi-Objective Prediction XGBoost Model for Predicting Ground Settlement, Station Settlement, and Pit Deformation Induced by Ultra-Deep Foundation Construction

Guangkai Huang^{1,2}, Zhijian Liu^{1,3,*}, Yajian Wang^{1,4}  and Yuyou Yang¹¹ School of Engineering and Technology, China University of Geosciences (Beijing), Haidian District, Beijing 100083, China² China Railway 18th Bureau Group Co., Ltd., Hexi District, Tianjin 300222, China³ China Railway 14th Bureau Group Co., Ltd., Lixia District, Jinan 250014, China⁴ National Center for Materials Service Safety, University of Science and Technology Beijing (USTB), Beijing 100083, China

* Correspondence: 2023020022@cugb.edu.cn

Abstract: Building a deep foundation pit in urban centers frequently confronts issues such as closeness to structures, high excavation depths, and extended exposure durations, making monitoring and prediction of the settlement and deformation of neighboring buildings critical. Machine learning and deep learning models are more popular than physical models because they can handle dynamic process data. However, these models frequently fail to establish an appropriate balance between accuracy and generalization capacity when dealing with multi-objective prediction. This work proposes a multi-objective prediction model based on the XGBoost algorithm and introduces the Random Forest Bayesian Optimization method for hyperparameter self-optimization and self-adaptation in the prediction process. This model was trained with monitoring data from a deep foundation pit at Luomashi Station of Chengdu Metro Line 18, which are characterized by a sand and pebble stratum, cut-and-cover construction, and a depth of 45.5 m. Input data of the model included excavation rate, excavation depth, construction time, shutdown time, and dewatering; output data included settlement, ground settlement, and pit deformation at an operating metro station only 5.7 m adjacent to the ongoing pits. The training effectiveness of the model was validated through its high R^2 scores in both training and test sets, and its generalization ability and transferability were evaluated through the R^2 calculated by deploying it on adjacent monitoring data (new data). The multi-objective prediction model proposed in this paper will be promising for monitoring the data processing and prediction of settlement of surrounding buildings for ultra-deep foundation pit engineering.

Keywords: foundation pit; existing subway station; multi-objective prediction; XGBoost; shutdown time



Citation: Huang, G.; Liu, Z.; Wang, Y.; Yang, Y. A Multi-Objective Prediction XGBoost Model for Predicting Ground Settlement, Station Settlement, and Pit Deformation Induced by Ultra-Deep Foundation Construction. *Buildings* **2024**, *14*, 2996. <https://doi.org/10.3390/buildings14092996>

Academic Editor: Elena Mele

Received: 7 August 2024

Revised: 9 September 2024

Accepted: 19 September 2024

Published: 21 September 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the development of urbanization, shallow underground space has almost been used up, which prompts new underground construction to go deeper [1–3]. The construction of deep foundation pits is necessary for building underground infrastructure such as metro stations, underground car parks, etc. [4,5]. However, future deep foundation pits will almost inevitably be constructed in city centers, facing the challenge of protecting the adjacent existing metro lines, stations, or buildings from being disturbed [6,7]. Predicting the settlement of critical infrastructure and ground surface around the foundation pit is usually an important guide to construction safety [8,9].

Two types of methods are currently used to predict settlement in pit construction: one is the first-principle model based on the physical laws and the other is the data-driven model based on machine learning or artificial neural networks [10–12]. The first-principle models

are usually theoretical or semi-theoretical formulations based on knowledge of theoretical mechanics, geotechnics, and hydraulics. In practical engineering, these theories are usually implemented using numerical methods such as finite element and finite difference to calculate the ground settlement caused by pit excavation and dewatering [11,13].

By employing long-term in situ monitoring and numerical simulation using Midas GTS finite element software, Liu et al. examined the deformation characteristics of a complex deep foundation pit in Taizhou. Their findings revealed horizontal displacements and settlements within safe limits, providing valuable data and insights for future construction projects in similar geological conditions [14]. Lin et al. simulated deep foundation pit excavation by MIDAS/GTS software, compared the calculated and measured data of pile deformation and internal support axial force, and analyzed the effect of the excavation method on the deformation of the supporting structure to optimize the excavation method [13]. Zhang simulated the construction process of the Xujiapeng foundation pit using Plaxis finite element software with the HSS soil model, observing a “concave” surface settlement pattern around the pit with a value of approximately 30mm, and a maximum diaphragm wall deformation of 32.29 mm under various loads [15]. Dong et al. investigated the effects of foundation pit excavation on adjacent building foundations, verifying a numerical model through experimental comparison and by studying the surface settlement distribution and its relationship with excavation depth and additional loads [16].

These numerical simulation models are constrained by parameter sensitivity, static computation processes, and limited parameter adaptability in predicting ground and adjacent building settlements due to foundation pit excavation, which reduces the accuracy and applicability of the predictions; hence, data-driven models are increasingly becoming the focus of research, owing to their advantages in handling complex nonlinear problems, dynamic parameter updating, and adaptive adjustments. Various models have been proposed and validated to predict the settlement of the surrounding ground surface and structures due to foundation pit construction, for example, the neural network model, time series model, regression analysis model, grey prediction and support vector machine model, etc. [17–21].

In 1995, Goh et al. demonstrated that neural networks can capture the non-linear relationship between variables in pit excavation and, when trained on sufficient sample data, can obtain a more accurate prediction of perimeter wall displacements [22]. BP (Backpropagation) Neural Networks are developed from the basic neural network structure, which optimizes the network weights by introducing a backpropagation algorithm [23]. Wu et al. used a genetic algorithm (GA) to optimize the BP neural network to cope with its defect of often getting stuck in the local optimal solution and constructed a GA–BP prediction model to predict the surface settlement of the Zhoushan foundation pit project [24]. Li et al. employed LSTM networks to predict deep foundation pit deformation, emphasizing the critical role of the Adam optimization algorithm in achieving high prediction accuracy. The research validates the LSTM network’s utility in deformation data analysis for deep foundation pits [18]. Zhang et al. introduced the spatiotemporal deep hybrid prediction model (STdeep model), a spatiotemporal deep hybrid prediction model, to achieve more precise surface settlement predictions around subway foundation pits. The model, which incorporates convolutional long short-term memory neural networks and gated recurrent units within a dual-stream structure, was shown to effectively capture the dynamic spatiotemporal relationships of settlement data, exhibiting optimal performance and robust stability in two practical cases [25].

These deep learning or machine learning algorithms have made significant progress in solving practical engineering problems; however, these algorithms have challenges with hyperparameter adaptation, model generalization, and multi-objective prediction when predicting deep foundation pit construction settling.

Addressing these challenges, this paper establishes a multi-objective prediction model by integrating the Random Forest Bayesian optimization process into the XGBoost algorithm. It achieves automatic adjustment of hyperparameters based on dynamic monitoring

data during pit excavation and can simultaneously output multi-objective prediction results, including ground settlement, pit deformation, and displacement of the adjacent subway station.

This paper is organized as follows: Section 2 describes the fundamentals of the XGBoost algorithm. Section 3 describes the training process and algorithmic implementation of the proposed model, including engineering cases, dataset preparation, and algorithmic frameworks. Section 4 discusses and evaluates the predictive performance and generalization ability of the trained model by deploying it on new data.

2. Methods

The Extreme Gradient Boosting (XGBoost) algorithm was chosen to predict ground settlement and pile displacement caused by deep foundation pit construction in the metro. This is because it can comprehensively learn and abstract data patterns, effectively handle a large number of complex features, and provide high efficiency by utilizing parallel computing [26,27]. It improves prediction accuracy by integrating many decision trees, with a new tree added at each step to minimize the objective function (consisting of the training loss and the canonical term). The gradient boosting method is used to create new decision trees by calculating the first- and second-order derivative information of the loss function, iteratively optimizing the model and gradually reducing the prediction error, which improves the accuracy and stability of the prediction results [27,28].

The basic unit construct of the XGBoost algorithm is CART (Classification and Regression Tree). Whether performing classification or regression tasks, XGBoost uses CART as the base learner.

The CART algorithm is a binary recursive partitioning technique, where the dataset is represented by “nodes”, which can only be divided into two categories, parent and child nodes. Each divided child node can continue splitting (recursively) until the set maximum depth of the tree is reached. Therefore, the cost function (as Equation (1)) of CART is the sum of the mean squared errors of all nodes [27,28].

$$J(k, t_k) = \frac{m_{left}}{m} MSE_{left} + \frac{m_{right}}{m} MSE_{right} \quad (1)$$

where the mean square error (MSE) represents the error at each leaf node and can be calculated by Equation (2):

$$MSE_{node} = \sum_{i \in node} (\hat{y}_{node} - y^{(i)})^2 \quad (2)$$

where $\hat{y}_{node}$ is the predicted output of the node and $y^{(i)}$ is the actual value, which can be derived from Equation (3).

$$\hat{y}_{node} = \frac{1}{m_{node}} \sum_{i \in node} y^{(i)} \quad (3)$$

where m_{node} is the count of instances in the node.

In terms of dealing with overfitting, XGBoost includes built-in regularization, which tends to make it a better learner than other gradient-boosting machines. This regularization term helps to smooth the final learned weights to avoid overfitting.

The simplified version of the process is as follows:

- Step 1: Start with a single tree. Initially, in the absence of any predictors, the mean of the target variable is assumed as the prediction for all observations.
- Step 2: Compute the residuals. The residuals are computed as the actual target variable minus the predicted target variable. The aim is to prepare a model to predict this residual.
- Step 3: Build a decision tree. The next step involves building a decision tree to predict the residuals instead of the actual target variable.
- Step 4: Update Predictions. The new prediction will be equal to the old prediction plus learning rate times residuals predicted by the decision tree.

- Step 5: Iterate. Repeat steps 2 to 4 until the maximum number of iterations is reached or the addition of new trees does not reduce the residuals significantly anymore.

In consideration of the culmination of the task yielding K -trained trees, with T representing the rudimentary tree model, the predictive outcomes are computed by aggregating the tree models (as depicted in Equation (4)):

$$\hat{y}_i = \sum_{k=1}^K f_k(x_i), f_k \in T \quad (4)$$

Here, $f_k(x_i)$ symbolizes the score attributed to each leaf node and T exemplifies the comprehensive basic tree model.

Equation (5) articulates the objective function:

$$Obj = \sum_i L(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \quad (5)$$

where L is the loss function that represents the gap between the predicted value and the real value; Ω is the regularization function that is used to control overfitting, which is expressed by Equation (6).

$$\Omega(f) = \gamma N + \frac{1}{2} \lambda \|w\|^2 \quad (6)$$

where N is the number of leaves on each tree and w is the weight of each tree's leaves; γ and λ are controlling factors employed to prevent overfitting.

The prediction after the t th iteration is denoted as $\hat{y}_i^{(t)}$. It can be expressed as Equation (7):

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i) \quad (7)$$

Thus, the objective function can be calculated as Equation (8):

$$Obj^{(t)} = \sum_i L(\hat{y}_i^{(t-1)} + f_t(x_i), y_i) + \Omega(f) \quad (8)$$

The second-order Taylor expansion of the objective function is approximately written as Equation (9):

$$Obj^{(t)} = \sum_i [L(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f) \quad (9)$$

where g_i is the first-order derivative of the loss function and h_i denotes its second-order derivative. They can be obtained by Equations (10) and (11), respectively.

$$g_i = \partial_{\hat{y}_i^{(t-1)}} L(y_i, \hat{y}_i^{(t-1)}) \quad (10)$$

$$h_i = \partial_{\hat{y}_i^{(t-1)}}^2 L(y_i, \hat{y}_i^{(t-1)}) \quad (11)$$

Since the residuals $L()$ of the $(t - 1)$ th tree have little influence on the change of the objective function, Equation (9) can be simplified to

$$Obj^{(t)} = \sum_i \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \quad (12)$$

All instances within the same leaf node can be reassembled. For example, Equation (12) for the j th leaf node can be written as Equation (13):

$$\begin{aligned} Obj^{(t)} &= \sum_i \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \frac{1}{2} \lambda \sum_{j=1}^N w_j^2 \\ &= \sum_{j=1}^N \left[\left(\sum_{i \in I_j} g_i \right) w_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) w_j^2 \right] + \gamma N \end{aligned} \quad (13)$$

The optimal weights are acquired as Equation (14) while the derivative of the objective function is equal to zero.

$$w_j^* = - \frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda} \quad (14)$$

Then, the objective function is expressed as Equation (15):

$$Obj^{(t)}(q) = - \frac{1}{2} \sum_{j=1}^T \frac{\left(\sum_{i \in I_j} g_i \right)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma N \quad (15)$$

XGBoost adopts a greedy algorithm to improve computational efficiency, which is achieved by performing two loops. One is for each split point of each feature and finds the one with the largest gain as the split point. The other iterates through all the features to find the feature with the greatest gain. The information gain of the objective function can be expressed as Equation (16):

$$Gain = \frac{1}{2} \left[\frac{\left(\sum_{i \in I_L} g_i \right)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{\left(\sum_{i \in I_R} g_i \right)^2}{\sum_{i \in I_R} h_i + \lambda} - \frac{\left(\sum_{i \in I} g_i \right)^2}{\sum_{i \in I} h_i + \lambda} \right] - \gamma \quad (16)$$

where I_L represents all left nodes after each split and I_R represents all right nodes.

3. Case Study

Chengdu Metro Line 18 Luomashi Station is a four-line interchange station located in the central area of the city, and its excavation area is adjacent to the existing pre-existing station, requiring settlement control to a very small extent. Settlement control is very challenging for this project and involves two aspects: firstly, the sand pebble–mudstone composite ground interface stagnant water evacuation drying process easily leads to sediment loss; secondly, Supporting the ultra-deep foundation pits constructed by open cut methods involves complex steps, resulting in a long time of exposure of the pit, and all these aspects contribute to the aggravation of settlement.

3.1. Engineering Case

The deep foundation pit of Luomashi Station of Line 18 in Chengdu, Sichuan Province, China was used as a case study. It is an interchange station of Lines 1, 4, 10, and 18, and it is a six-storey underground double-column, three-span island-type station with a total length of 214.6 m, a standard section width of 27.8 m, and an open-cut method of construction with a maximum depth of 45.5 m. The project was carried out by the open-cut method of construction.

Its dimensions and geometry are shown in Figure 1. It can be seen that in the existing Line 1 station (below Renmin Middle Road) and the existing Line 18 new pit and parallel laying, the minimum distance is only 5.7 m, resulting in the existing station being completely in the pit excavation within the scope of influence.

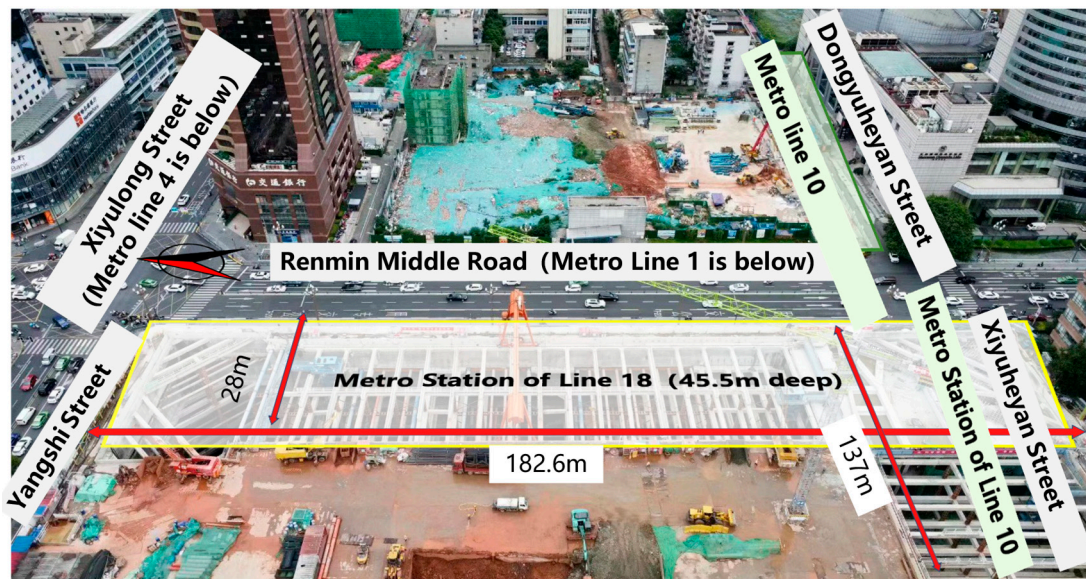


Figure 1. A view of Luomashi Station of Chengdu Metro Line 18.

The stratigraphy within the project area is—in vertical sections from top to bottom—plain fill, silty clay, fine sand, medium dense pebbles, dense pebbles, and bedrock of strongly and moderately weathered mudstone. The distribution of stratigraphic thickness and geotechnical characteristics are given in Table 1 and shown in Figure 2.

Table 1. Soil thickness and composition.

Soil Type	Geological Era	Thickness (m)	Depth of Burial (m)	Composition and Properties
Plain fill	Q_4^{ml}	0.5–2.7	—	Consists of clayey soil, concrete rubble, sand, pebbles
Silty clay	Q_4^{al}	1.0–4.4	0.5–4.0	Contains ferromanganese nodules and a few calcium nodules with hard plasticity
Fine sand	Q_4^{al}	0–3.5	3.3–10.5	Distribution matches moist to saturated, loose to slightly dense
Medium dense pebbles	Q_4^{al}	0.9–5.2	3.0–7.9	Mainly medium dense with a distribution of wet to saturated; the filler is mainly fine sand and rounded gravel
Dense pebbles	Q_3^{fgl+al}	19.2–29.3	5.5–9.0	Saturated, dense, grain size 20–200 mm, filler mainly medium and fine sand
Strongly weathered mudstone	K_2g	0–2.3	28.2–36.1	The rock is soft, joints are developed, and the core is mostly crumbly or pie-shaped
Moderately weathered mudstone	K_2g	—	29.1–36.5	The rock is hard, with a small amount of sand, and the weathering fissures are more developed

As illustrated in Figure 2, the groundwater level is 2–7.1 m; the interface of pebbles and mudstone is located at about 28.5–30 m below the ground surface; and there is a perched water layer at the interface, which is difficult to drain.

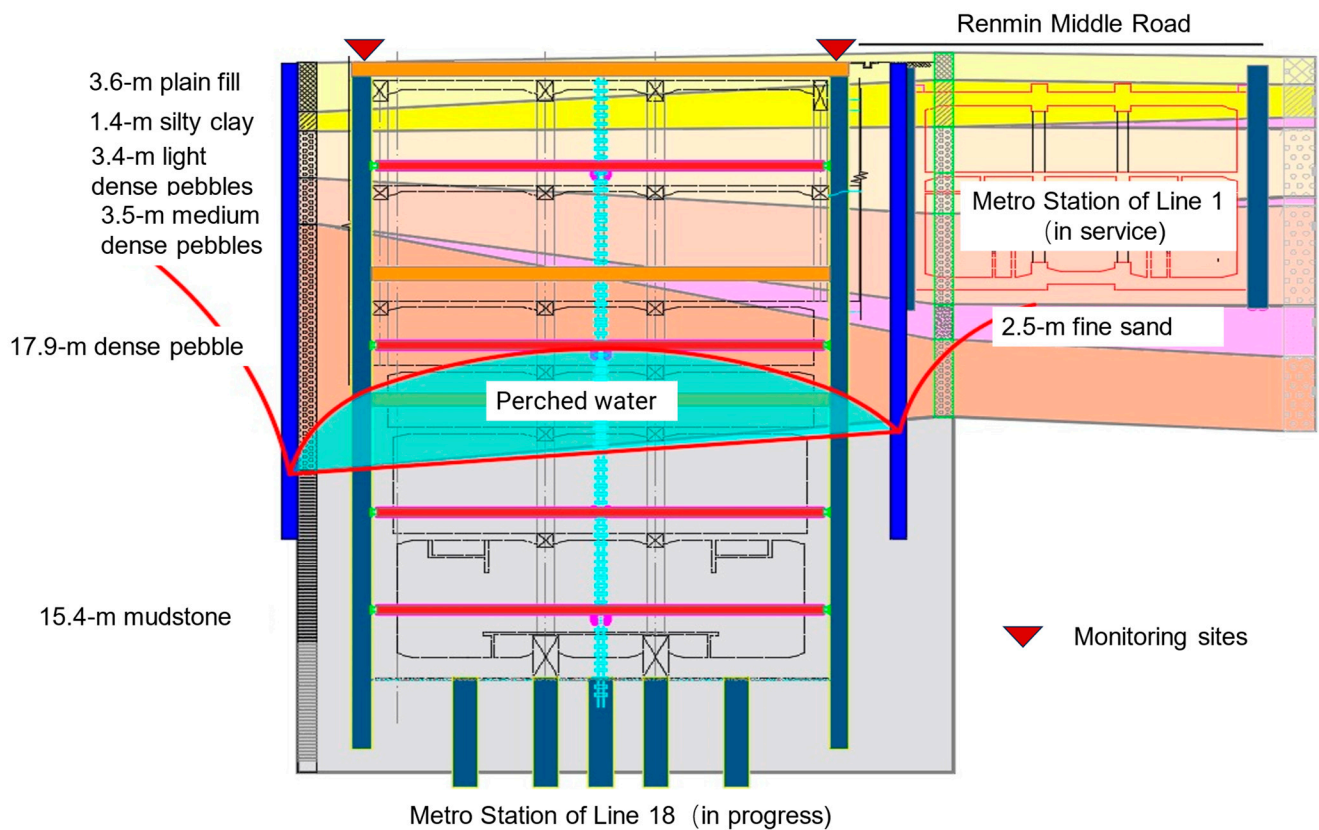


Figure 2. Geological profile of the foundation pit of Luomashi Station.

3.2. Datasets and Preprocessing

3.2.1. Input Datasets

The datasets consist of the data reflecting the construction process and the monitoring data. The input data reflecting the construction process consist of construction time, earthwork volume, excavation depth, groundwater level decline, excavation rate, and shutdown time.

All the data related to pit excavation are a time series. Thus, the construction time is chosen as an input attribute, which is the number of days until the start of construction, ranging from 0 to 676 days for the entire duration of the project.

Excavation progress is defined in this paper to describe the ratio of the cumulative amount of earth excavated on the day and the total amount of earth required, which lies between 0 and 1 and is expressed as a percentage.

Excavation depth reflects the elevation of the excavation depth in the cross-section area corresponding to the monitoring point location as the output data, which has similar physical significance to the project progress, with the difference that the excavation depth represents the workload in the longitudinal direction whereas the excavation progress is the overall progress of the project in both the vertical and horizontal directions.

Excavation rate is a very important factor affecting the disturbance settlement of deep foundation pit construction, which involves parameters with time-dependent characteristics such as soil strain rate, reaction time of supporting structures, and soil drainage conditions. Therefore, the excavation rate—the ratio of excavation depth to construction time—is used as one of the input characteristics in this project.

Groundwater descending depth for ground settlement and soil displacement may not be so significant for pebbles, but the Luomashi Station local stratigraphy is interspersed with a layer of powdery fine sand, which is prone to cause water gushing and sand gushing. In addition, as the hydraulic gradient increases during pumping down, fine particles are carried out by the water, resulting in submerged erosion and piping,

which can easily lead to the rearrangement of the coarse particles to compress the density and, thus, cause subsidence.

The duration of the shutdown, i.e., the length of time that excavation is interrupted due to the influence of cross-construction, machine failure, extreme weather, epidemic, etc., has a relatively important influence on the ground settlement and pit deformation.

3.2.2. Output Datasets

The output data include the settlement of the existing station, deformation of the bracing piles, and ground settlement. All of these use cumulative values during whole construction.

As shown in Figure 2, the monitoring sites of enclosure piles were installed at the top of the crown beam by drilling out a hole of about 10 cm and putting the prism connecting rod into the hole. The gap is filled with an anchoring agent, and the prism can be accessed during monitoring. The total station measurement technique is used to determine the coordinate change of the monitoring point to find the displacement vector of the point, and the horizontal and vertical displacements of the pile can be obtained by decomposing the displacement vector into specific horizontal and vertical directions, respectively.

As shown in Figure 3, the external wall of the station is at a vertical distance of 10 m from the edge of the pit. The monitoring points for station are placed every 10 m on the roadbed. The monitoring data from DM2_left of cross-section A-A were used as the output data. The surface monitoring sites are deployed on the ground beyond 1 m from the edge of the pit, where the monitoring data of DB-14 are taken as the output data.

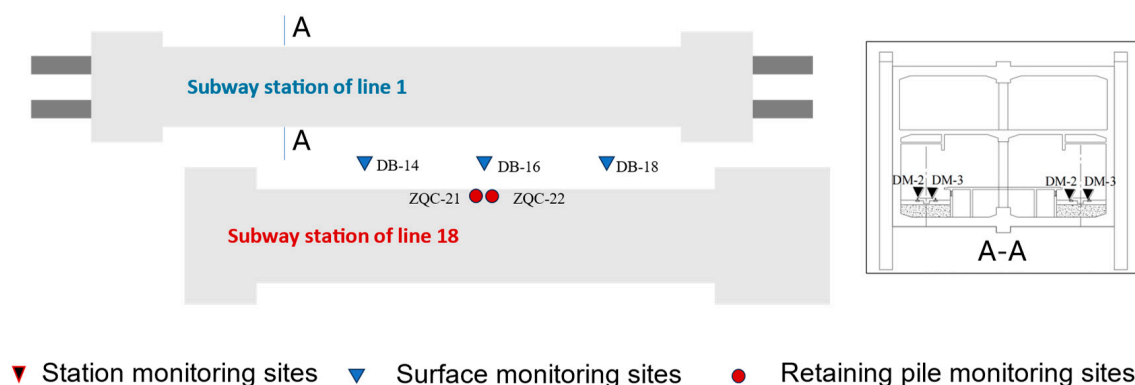


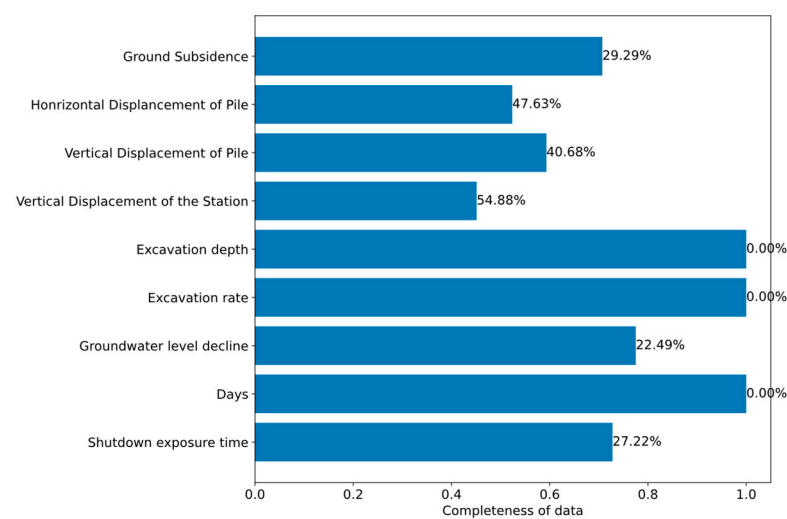
Figure 3. Monitoring sites for new data to be deployed into the model.

3.2.3. Data Preprocessing and Imputation

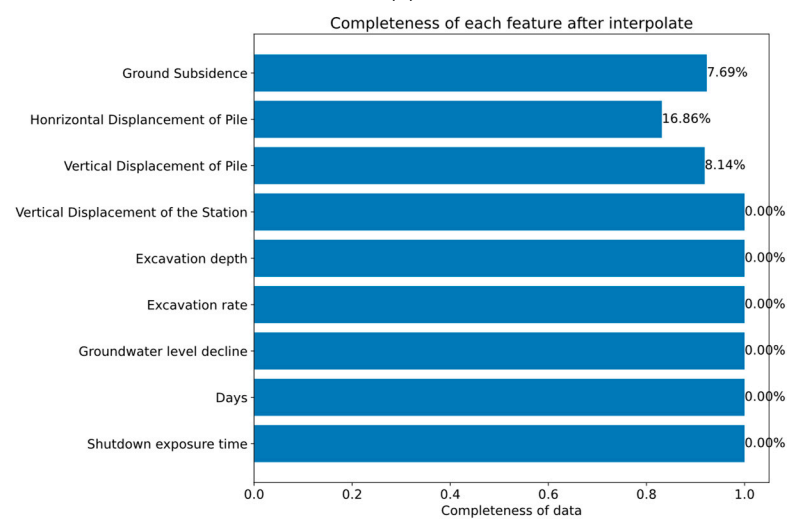
The progress of the project is usually affected by climate and epidemics, resulting in some of the monitoring data being missing. Since the change in the project data is a continuous process, this paper uses the interpolation method to supplement the missing values.

All of the data manipulation and project coding were conducted in Jupyter Notebook, utilizing Python version 3.8. The main objective of data preprocessing, also known as feature engineering, is to facilitate a more effective “understanding” of the data by machine learning algorithms, thereby enhancing the quality of the training process.

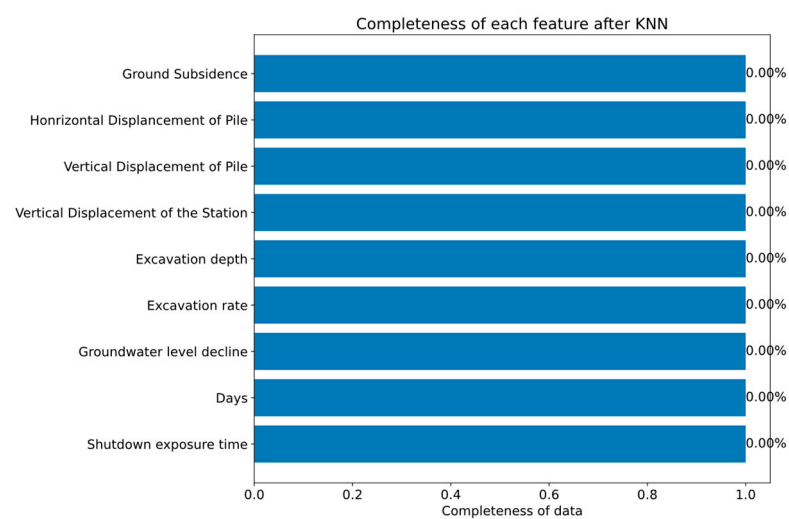
The *missingno* package has been put into use for the visual inspection of missing values within attributes, which is demonstrated in Figure 4. As shown in Figure 4a, excavation depth, excavation rate, excavation progress, and time form a total of 676 sets of data with no missing values, while surface settlement is missing by 29.29%, horizontal displacement of bracing piles is missing by 47.63%, vertical displacement of bracing piles is missing by 40.68%, vertical displacement of the station is missing by 54.88%, dewatering depth is missing by 22.49%, and shutdown time is missing by 27.22%.



(a)



(b)



(c)

Figure 4. Filling of missing values, where (a) are the original data, (b) are the data processed by median interpolation, and (c) are the data filled by using the K-Nearest Neighbor Algorithm.

As shown in Figure 4b, after filling in the missing values using the interpolation method, there are still 7.69%, 16.66%, and 8.14% missing values for surface settlement, horizontal pile displacement, and vertical pile displacement, respectively, which is due to the defects of the interpolation method that cannot fill in the data at the beginning and at the end of the data.

The XGBoost algorithm can handle missing values through a specialized mechanism, but this is limited to missing feature values (input data). However, the data in this article are all missing target values (output data), which is something the XGBoost algorithm cannot handle. Therefore, KNNImputer is further employed here to predict these missing values. The reason for using the K-Nearest Neighbor Algorithm (KNN) to predict the missing values after the interpolation method is explained as follows:

Data interpolation is a method of prediction based on existing parts of the data, which utilizes the relationships between the data to estimate missing values, whereas the KNN method predicts based on the similarity of the data. Interpolation and KNN emphasize the inherent trends and similarities in the data, respectively, and they complement each other to make predictions more comprehensive and accurate.

The interpolation method mainly deals with sequence-type data, and it is very effective to use interpolation when data are missing in the time series. However, in this case, some data cannot be effectively interpolated by the data before and after, so the KNN algorithm is introduced to fill in effectively based on feature similarity.

KNN is extremely sensitive to missing values, and the original data in this paper have an excessive number of missing values; thus, the similarity based on this feature is likely to be inaccurate. Interpolation is performed first to reduce the number of missing values, making subsequent applications of the KNN method more accurate.

The scientific basis for using the missingno package and the KNN algorithm for data imputation lies in their ability to accurately fill in missing data by leveraging the similarity between data points, combining iterative processes, and optimizing the selection of neighbors, thereby enhancing the accuracy and efficiency of data mining [29,30].

3.3. Modelling Process

3.3.1. Data Analysis

Before the training task of multiple regression, the relationship between each variable is analyzed. Figure 5 displays the Pearson Correlation Coefficient (PCC), which is a measure of the amount of linear correlation between two variables. Pearson's correlation coefficient (PCC) ranges from -1 to 1 . Its value approaching -1 and 1 indicates that the two variables are highly dependent but with negative and positive effects, respectively [31].

As can be seen from Figure 5, the linear correlation between surface and station settlement and excavation progress and excavation time is good, ranging between 0.5 and 0.8 . The PCC coefficient of excavation depth and excavation progress is equal to 1 , and they are completely equivalent, so the excavation progress is excluded from the dataset when training the model. The PCC values of pile deformation with other factors are all 0 – 0.31 , proving that it has a poor linear correlation with other factors.

As can be seen in Table 2, there are 676 sets of data for construction time, but most of the factors have less data because the monitoring or construction do not take place daily. The range of excavation depth is 0 – 45.5 m, and the range of excavation rate is 0 – 0.19 m/day, which is converted to the equivalent value of excavation for the whole section—which is a stepped excavation—and the depth of excavation is much more than that. The depth of groundwater drop is 0 – 17.24 m. The duration of the work stoppage is 0 – 57 days. The vertical displacement of the existing station is 0 – 3.53 mm. The vertical and horizontal displacements of the pile varied from 0 to 12.77 mm and 0 to 1.4 mm, respectively. Ground settlement is in the range of 0 – 16.87 mm.

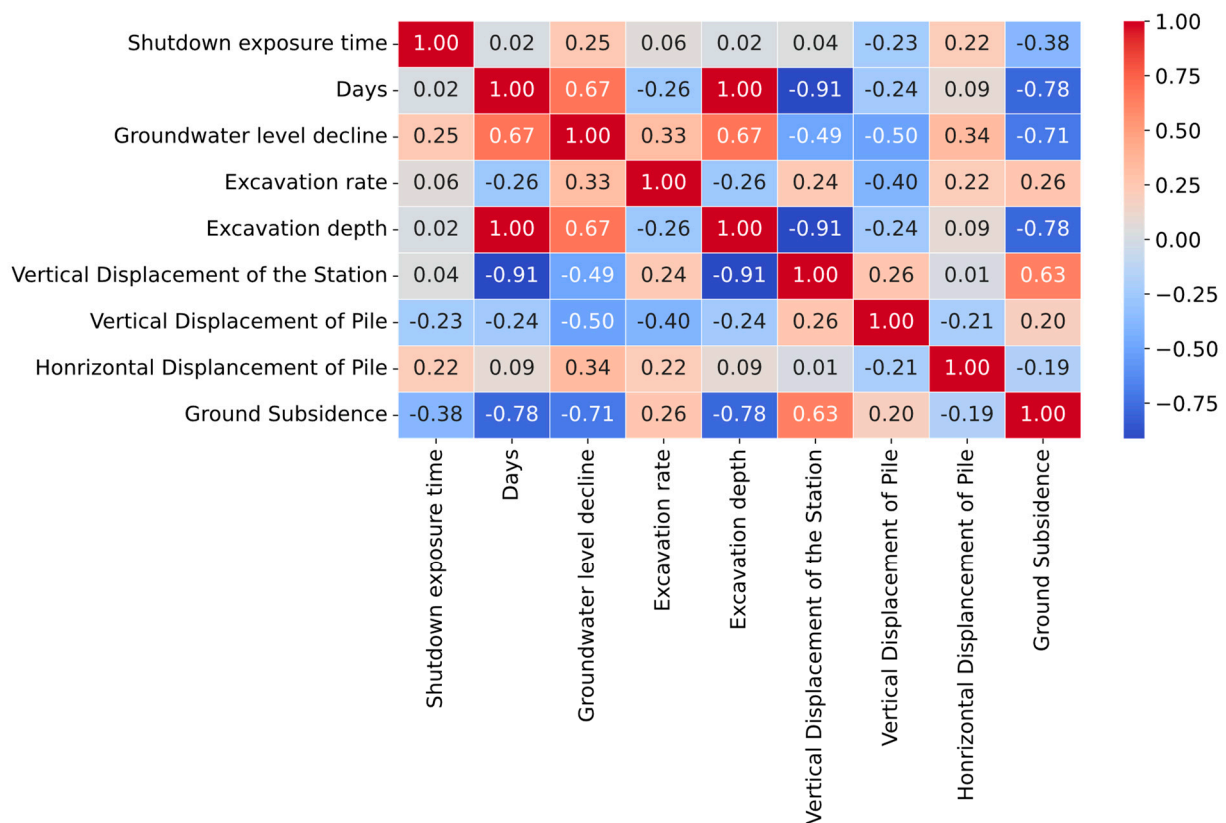


Figure 5. Pearson correlations of each pair of attributes.

Table 2. Contributors of the database and its range of values.

Input						Output			
Construction Time (Day)	Excavation Progress (%)	Excavation Depth (m)	Excavation Rate (m/Day)	Groundwater Level Decline (m)	Shutdown Explosion Time (Day)	Vertical Displacement of the Station (mm)	Vertical Displacement of the Pile (mm)	Horizontal Displacement of Pile (mm)	Ground Subsidence (mm)
0–676	0–100	0–45.5	0–0.19	0–17.24	0–57	0–3.53	0–12.77	0–1.4	0–16.87

3.3.2. Data Split

Shutdown time, construction days, excavation progress, dewatering depth, and excavation rate are defined as inputs, while the station vertical displacement, pile vertical displacement, pile horizontal displacement, and ground settlement are used as outputs.

Before model training, the dataset is divided into the training set and test set. In this paper, there are multiple objective variables to prepare the corresponding training set and test set for each objective variable separately; this process is achieved by creating a loop function. For each target variable, the test set and the training set were randomly placed into two dictionaries, *test_sets* and *train_sets*, from the dataset at a ratio of 8:2, respectively. The random_state was set as 42. These processes were looped four times until the dataset was divided for each target variable.

3.3.3. Model Training

The process of hyperparametric optimization consists of three steps: model initialization, Bayesian optimization, and return of hyperparametric optimization results to the model. Model initialization and hyperparameter search threshold determination are empirical and require several trial runs. Hyperparameter optimization is set to be completed through 200 iterations of training, and if the model loss function converges early,

the early_stopping mechanism is used to complete the model training task. The optimal parameters are then returned to the model.

As shown in Figure 6, model training and optimization are performed independently for each target variable in this study, whose specific details were described in Table 3. The hyperparameters that require optimization for the proposed model include max_depth, eta, subsample, colsample_bytree, reg_lambda, min_child_weight, and booster. Among these, max_depth and eta are critical in controlling the complexity and learning rate of the model, while subsample and colsample_bytree play pivotal roles in enhancing the model's generalization capability through regularization. Furthermore, reg_lambda and min_child_weight are essential for preventing overfitting by penalizing large weights and ensuring robust node splits, respectively. The choice of booster determines the underlying algorithmic approach to optimizing the objective function. The roles of these hyperparameters were summarized in a previously published article by Wang et al. [28]. The specific steps are as follows:

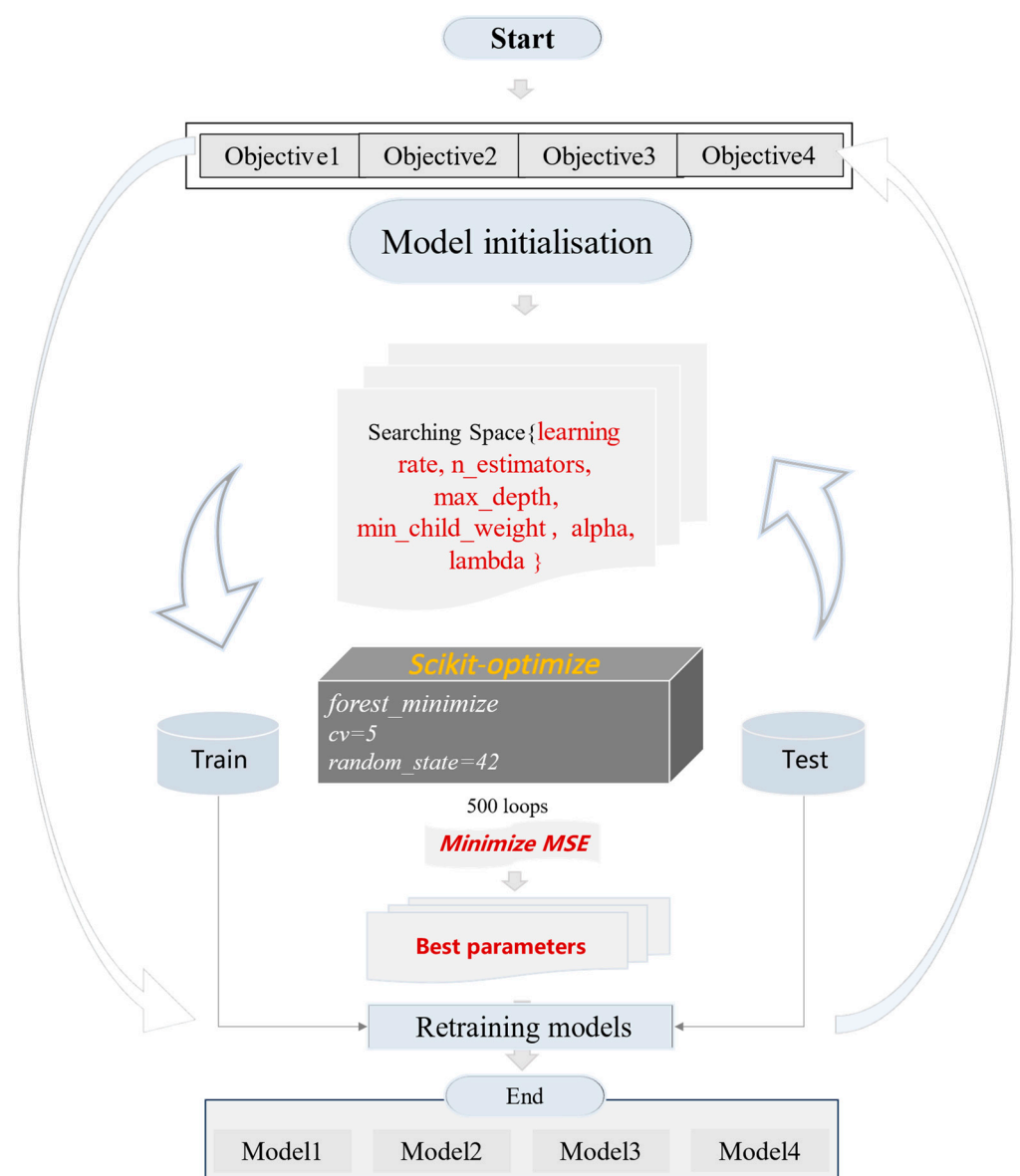


Figure 6. Framework diagram of a multi-objective prediction model based on the XGBoost algorithm.

Table 3. Execution steps for the multi-objective prediction algorithm.

Step	Action	Details
1	Initialize XGBRegressor	Set initial parameters: learning rate, n_estimators, max_depth, min_child_weight, alpha, lambda
2	Define Objective Function	Function to evaluate model performance (MSE) based on input parameters
3	Set Parameter Space	Define the range of values for each hyperparameter to be optimized
4	Initialize Bayesian Optimization	Use Bayesian Optimization with the defined objective function and parameter space
5	Perform Optimization	Run the optimization process with a specified number of initial points and iterations
6	Retrieve Best Parameters	Extract the hyperparameters that yield the best performance from the optimization results
7	Train Final Model	Train the XGBoost model using the optimal hyperparameters on the training data
8	Predict and Evaluate	Make predictions on the test set and evaluate the model's performance
9	Loop for Each Objective Variable	Repeat steps 1–8 for each objective variable to be predicted

Firstly, the XGBRegressor was initialized and some basic start-up parameters were set, such as learning rate, n_estimators, max_depth, min_child_weight, alpha, lambda, etc.

Then, the objective function for Bayesian optimization was set. This is a function that takes the parameters of the XGBoost model as inputs and a certain performance metric of the model (i.e., MSE) as output.

Next, Bayesian optimization is used to search for the best model parameters that minimize this objective function on both training and testing sets. During the search process, the algorithm automatically adjusts the hyperparameters using the forest_minimize algorithm. The performance of the model is evaluated using 5-fold cross-validation. After running the hyperparameters through regression and prediction, the algorithm determines which hyperparameters are most likely to improve the score and attempts to use them in the next run. Finally, the model is retrained using the optimal parameters found.

Each objective variable goes through the above process independently, resulting in a series of optimized XGBoost models. Each step in the above process uses a programming loop structure. It is possible to view this procedure as training several models since it involves integrating XGBoost into a looping framework to train one target at a time. For this reason, the training outputs for each target in this research are referred to as Models 1, 2, 3, and 4.

All of the remaining parameters are set to default values. In addition, n_estimators is initially set to 200, and an early_stopping_rounds of 40 is called to avoid overfitting. The model is checked for underfitting or overfitting by examining the mean absolute error (MAE) evolution curves of both the test and training sets during learning. As shown in Figure 7, the MAEs of both training and test sets decreased rapidly within the first 50 loops, indicating the improving performance of the model. The test curve reached a minimum MAE of 0.67 after about 75 iterations and then there was a slight rally, whereas that of the training curve continuously decayed to almost 0.5 under 200 loops. The MAE on the test set was always kept at a low level after 75 loops, implying that no overfitting of the model occurred. To preserve the generalization ability of the model, the iteration corresponding to the minimal variance between the MAEs of the training and test sets was set to be the moment to stop early. The hyperparameter optimization results are given in Table 4.

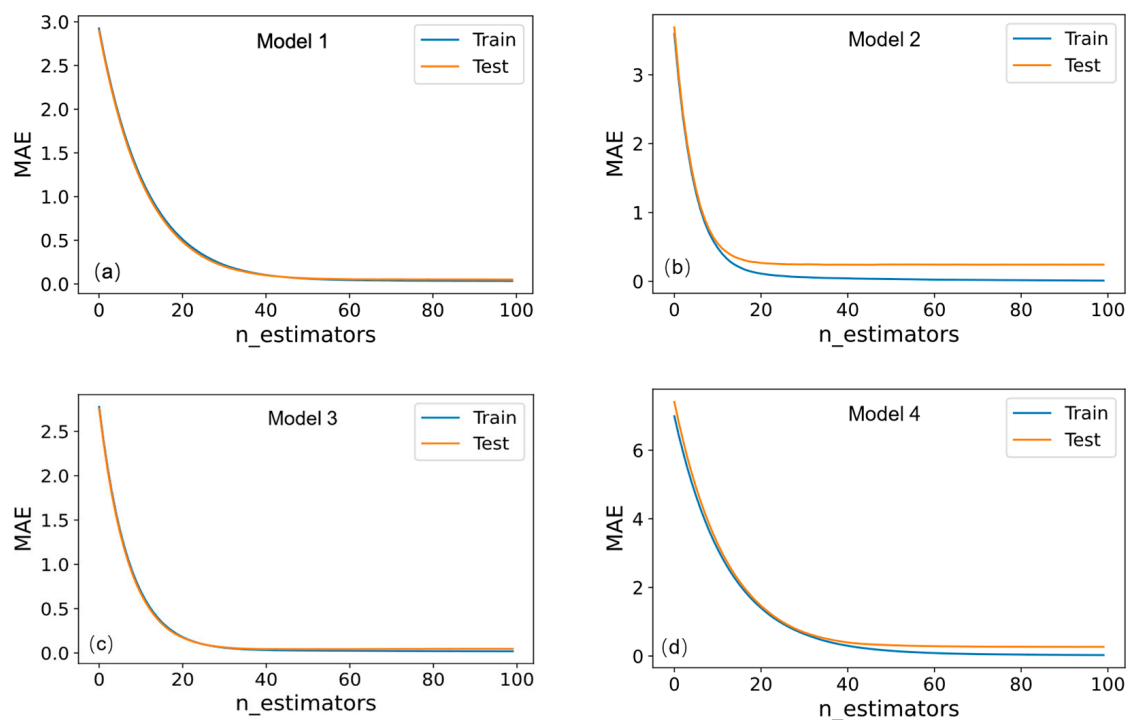


Figure 7. Learning curves for the model training process, where (a) is the learning process for predicting station settlement, (b) is the learning process for predicting vertical displacement of enclosure piles, (c) is the learning process for predicting horizontal displacement of enclosure piles, and (d) is the learning process for predicting ground settlement.

Table 4. Hyperparametric self-optimization results for the trained model.

Models	Max_depth	Learning Rate (eta)	Subsample	Colsample_bytree	Reg_lambda	Min_child_weight	Booster
Model1	3	0.084	0.990	0.645	0.982	0	gbtree
Model2	9	0.188	0.673	0.690	0.141	2	gbtree
Model3	14	0.131	0.615	0.882	0.714	5	gbtree
Model4	8	0.110	0.627	0.679	0.134	0	Dart

4. Results and Discussions

4.1. Training Results of the Model

To assess the proposed model's performance, the predicted values and the monitoring values on both the training and test sets were compared. R^2 scores were calculated to measure the predictive accuracy. Figure 8a–d illustrate the relationships between predicted and actual values for the XGBoost model applied to the four datasets. In these figures, the line represents perfect agreement between predicted and observed values. Scatter points closer to this line indicate higher accuracy, with points above the line showing over-prediction and those below it indicating under-prediction.

As illustrated in Figure 8a, the R^2 scores for Model 1 on the training and test sets are 0.972 and 0.962, respectively. Figure 8b shows that the R^2 scores for Model 2 are 0.989 on the training set and 0.918 on the test set. Figure 8c indicates that the R^2 scores for Model 3 are 0.924 for the training set and 0.883 for the test set. Lastly, Figure 8d reveals that the R^2 scores for Model 4 are 0.993 and 0.912 on the training and test sets, respectively.

The high R^2 scores on both the training and test sets indicate not only a good regularity in the data samples but also reveal that the model is well-trained, achieving balanced performance in both prediction accuracy and generalization.

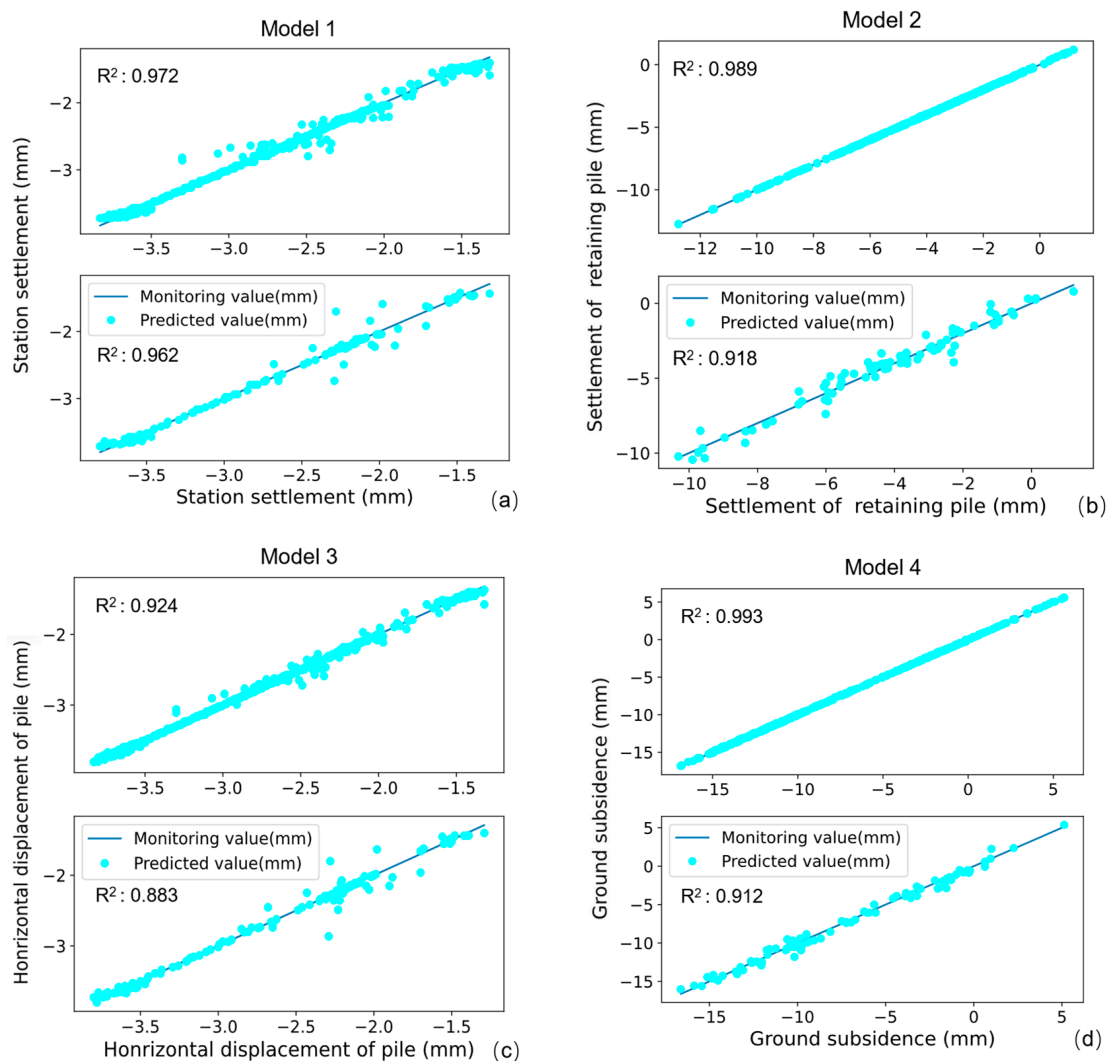


Figure 8. Comparison of predicted and actual values on the training and test sets for four models: (a) Model 1 predicting station settlement, (b) Model 2 predicting retaining pile settlement, (c) Model 3 predicting horizontal displacement of retaining piles, and (d) Model 4 predicting ground settlement.

4.2. Predictive Performance Assessment Based on the New Datasets

The model is built based on training on a small number of monitoring point data. This section aims to test the generalization ability of the model, which is accomplished by deploying the trained model to a new dataset of other monitoring points. As shown in Figure 3, station monitoring data DM2_left are the original training data for M1, and DM3_left, DM2_right, and DM3_right were used as new data to test the performance of the M1 model. Vertical and horizontal displacements of enclosure pile ZQC-21 are the original training data for M2 and M3, respectively, Vertical and horizontal displacements of ZQC-22 were used to test the M2 and M3 models, respectively. Ground monitoring data DB14 and DB16 are the original training data, whereas DB18 were used as new data to test the predictive ability of the M4 model.

As shown in Figure 9, the combined scores of the model for the new data samples including DM3_left, DM2_right, and DM3_right are 0.92, 0.88, and 0.88, respectively. This indicates, on the one hand, that the overall settlement pattern of the station is more consistent and, on the other hand, that the generalization of the model is good.

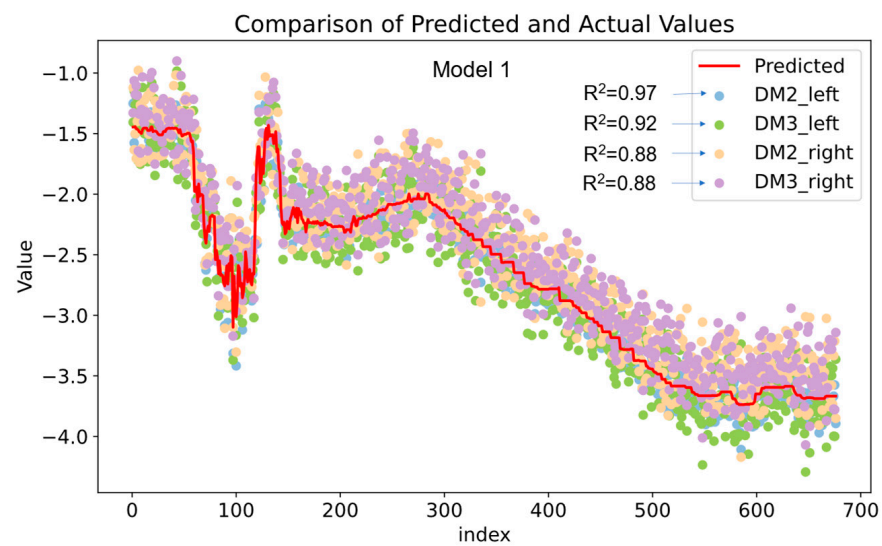


Figure 9. Comparison of monitoring values and predicted values of settlement for Line 1 stations.

As shown in Figure 10, Model 2 has an R^2 score of 0.95 for the full-sample data at ZQC21, while the full-sample R^2 score for the new data, ZQC22, is 0.88. Due to the proximity of the locations of the two piles, the trends of their data are very consistent and have close values. This also shows that the model has good transferability and generalization when dealing with data from the same situation.

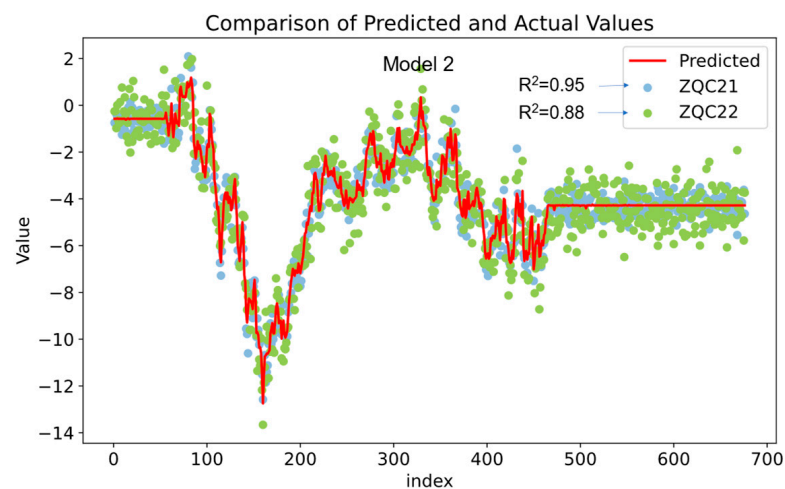


Figure 10. Comparison of measured and predicted settlement of enclosure piles, including in the new data ZQC22.

As shown in Figure 11, the horizontal displacement of the bracing piles exhibits complex fluctuating variations due to the cyclic action of support and excavation during construction. Such data are a huge challenge for the model, and for XGBoost, which can cope with complex forms of data deformation, but predicting these data requires metrics related to the erection and removal of supports, which is missing from the data in this paper.

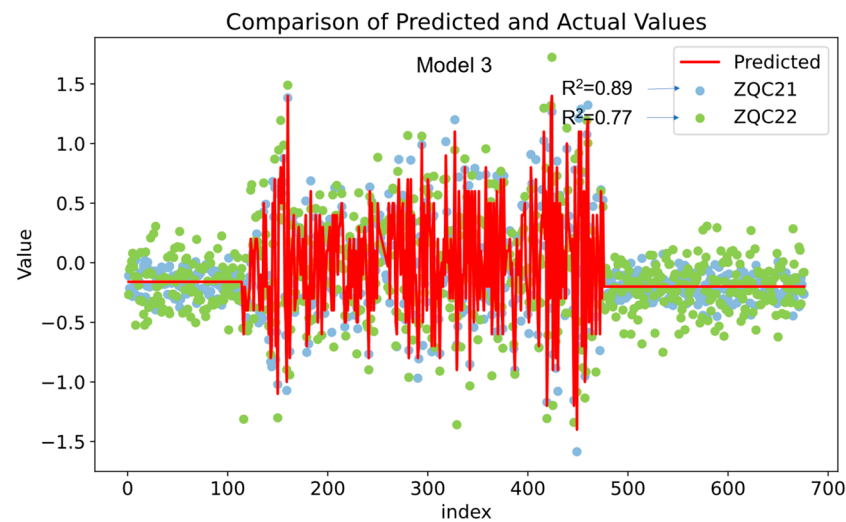


Figure 11. Comparison of predicted and measured horizontal displacement of enclosure piles.

As shown in Figure 12, the full-sample R^2 scores for Model 4 for the new data DB16 and DB18 are 0.90 and 0.89, respectively. These new measurement points and the measurement points used for training are all 1 m away from the edge of the pit, so it can be concluded that the settlement caused by pit excavation is more uniform. The higher R^2 scores imply that Model 4 can capture the characteristics of the data better, making it perform well on similar data samples.

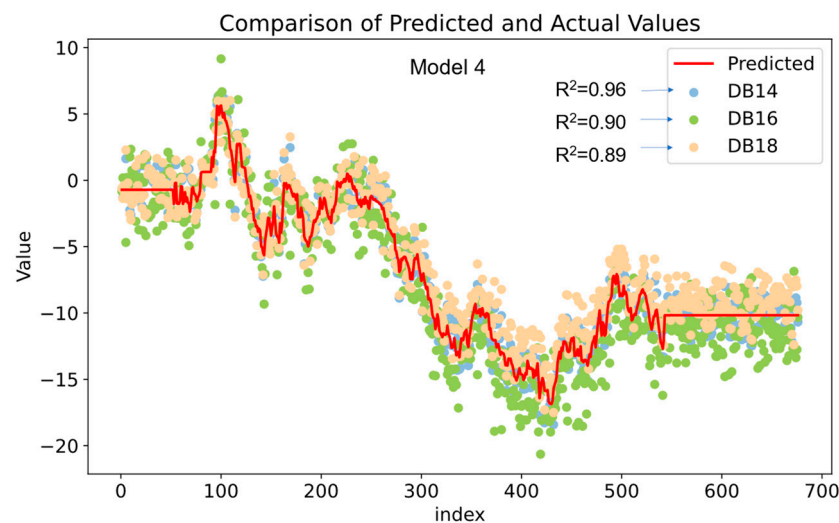


Figure 12. Comparison of predicted and measured ground settlement values.

4.3. Importance of Attributes

To quantify the relative importance of each feature in terms of predictive performance, a feature selection strategy based on tree-based machine learning algorithms was employed. Specifically, the F-score for each feature was calculated, which is obtained by averaging the gains of all nodes in which this feature participates in the splits. The gain can be calculated by Equation (16). It reflects the additional information increment that a specific feature brings relative to the baseline model. By accumulating the gain of each feature in every tree, a global measurement index was obtained, allowing for the determination of which feature contributes the most to the final prediction results across the entire model [32].

A higher F-score value indicates that the feature has a greater influence on generating the prediction results. This metric standard aids in identifying which features are the most influential factors in the model. By calculating and sorting the F-scores, a thorough

understanding of the role size of each feature in the model prediction process was achieved. This section calculates the F-scores of the four models to provide guidance for subsequent data preprocessing and analysis tasks.

From Figure 13a, it can be seen that the excavation rate is the most important influencing factor for the ground settlement around the pit. This makes sense because the rate of excavation is a combination of several factors reflecting the depth of excavation, groundwater drawdown, time, soil rheology, etc. This can be explained by the fact that pit construction is a dynamic process over time and settlement increases with time. The magnitude of settlement around the pit is not only related to the excavation rate but also closely related to factors such as soil stress adjustment, creep, consolidation, changes in the force of the supporting structure, and changes in the water level caused by time factors during the construction process. With the passage of time, the combined effect of these factors will lead to the accumulation and increase in settlement. Shutdown exposure time ranks third among the factors affecting the settlement around the pit, which may be due to a variety of reasons: on the one hand, after excavation of the pit, if the work stoppage leads to the failure of timely implementation of support measures, the sidewall of the pit will be in an unsupported or insufficiently supported state, which will lead to greater deformation of the sidewall soil due to the action of self-weight and external loading; on the other hand, if rainfall or other water intrusion is encountered, the pit without timely support may be softened and the strength of the soil reduced due to the action of water, thus causing greater settlement. Excavation depth is ranked fourth, but the importance of excavation depth should not be ignored, which only indicates that the above factors may have a more direct and rapid impact on pit stability and settlement than excavation depth. Groundwater level decline is ranked last; this is because the stratum in this case is sand and pebble, the consolidation induced by lowering groundwater is not very significant, and some suitable control measures were made against quicksand and pipe surges.

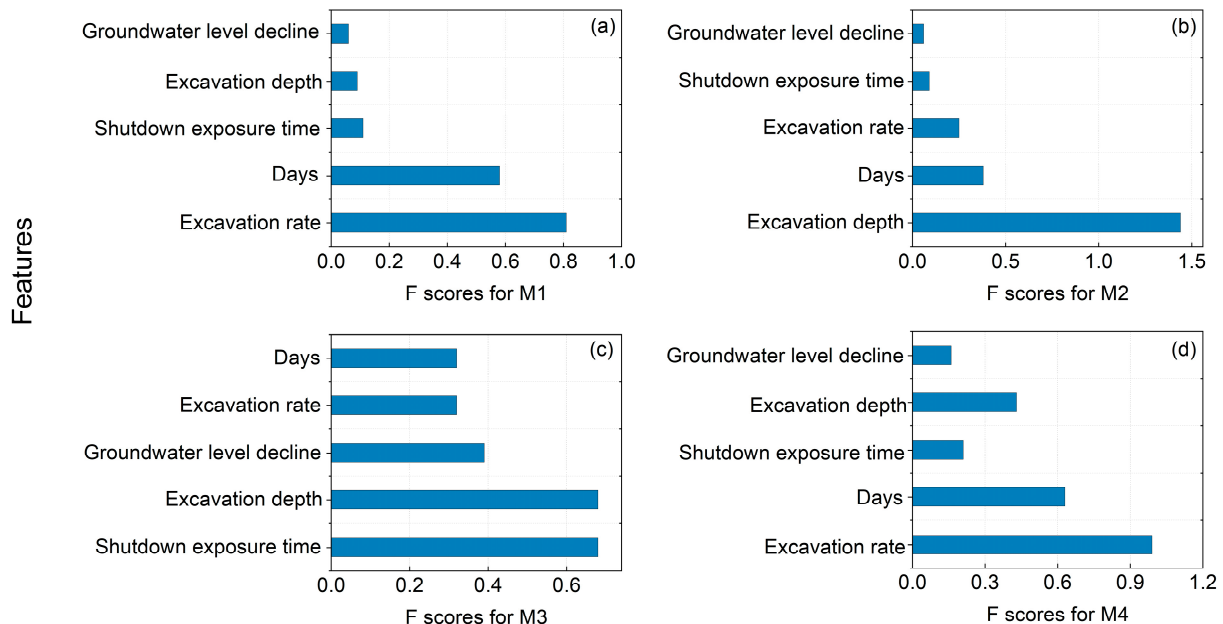


Figure 13. Importance rank of influencing factors for all models, where (a) represents the station, (b) represents the settlement of the enclosing piles, (c) represents the horizontal displacement of the enclosing piles, and (d) represents the ground settlement.

From Figure 13b, the excavation depth is sorted first because the pit excavation directly affects the stress state of the enclosing pile body. The days is ranked second, which is because the creep of the soil body may also lead to a certain vertical displacement of the support structure. Excavation rate is ranked third. The excavation rate mainly affects the stress release pattern of the soil body and the deformation behavior of the soil body

in the short term, and the effect on the pile body is not significant. Since the supporting structure has been stabilized, it is less affected by the shutdown time compared to the soil. The stratum is a sand and pebble layer, and the drop in water level has less effect on the supporting piles.

As can be seen from Figure 13c, both the shutdown time and the excavation depth have a significant effect on the horizontal displacement of the enclosing piles, which can be interpreted as the downtime on the one hand that reflects the rate of erection of the horizontal support, and the depth of excavation directly increases the earth pressure on the enclosing piles. The excavation rate, days, and groundwater level decline had little effect on the horizontal displacement of the enclosing piles as it was well balanced after the erection of the horizontal supports.

From Figure 13d, it can be seen that the order of importance of the factors affecting the ground settlement is consistent with the station settlement, and the explanation of its mechanism can also refer to Figure 13a.

5. Conclusions

Monitoring data during deep foundation pit construction can help guide construction management decisions to prevent excessive ground settlement. The case studied in this paper is an open-cut foundation pit with a depth of more than 45 m, whose excavation and supports involved are time-consuming, and the exposure time of the foundation pit is much longer than that of previous foundation pit projects. This generates a large amount of data, which are difficult to analyze manually. To this end, this paper develops a framework for the XGBoost algorithm that can perform multi-objective outputs and introduces a Bayesian optimization algorithm with a random forest search process to optimize the hyperparameters. The algorithm was initially deployed on monitoring data from Chengdu Rail Transit Line 18 stations to determine hyperparameters before completing model training. The training process was analyzed based on the learning curve of the model, and the trained model was evaluated by comparing the R^2 scores in the test set and the training set. The trained model was then also deployed to new station monitoring data to evaluate the migratability and generalization ability of the model. The main contributions of this study are as follows:

- (1) A multi-objective prediction model was built based on the XGBoost algorithm, which enables multi-objective prediction of ground settlement, pit deformation, and settlement of existing buildings during pit excavation. This process is implemented by embedding XGBoost into a looping framework to train one target at a time so that it may be interpreted as training more than one model, which is why in this paper the training outputs for each target are named Models 1, 2, 3, and 4.
- (2) The scores of each training objective on both the test set and the training set are high, for example, the scores on the training set and the test set for predicting station settlement are 0.992 and 0.982, respectively; further, the scores on the training set and the test set for predicting the settlement of the bracing piles are 0.999 and 0.968, respectively, which is because the monitoring data generated from the pit excavation process are a time-series and linearly varying process, with a high degree of regularity.
- (3) Model generalization is evaluated by deploying the trained model to new data, which are sourced from the monitoring sites close to the monitoring sites of the training data. Model 1 and Model 2 both have R^2 scores greater than or equal to 0.88 on the new dataset, Model 3 has an R^2 score of 0.77 on the new dataset, and Model 4 has an R^2 score greater than or equal to 0.89 on the new dataset, which suggests that the models in this paper are well suited to the studied case.

Author Contributions: Conceptualization, Z.L. and Y.Y.; methodology, G.H.; software, G.H.; validation, Y.W. and Y.Y.; formal analysis, G.H.; investigation, G.H.; resources, G.H.; data curation, G.H.; writing—original draft preparation, G.H.; writing—review and editing, Y.W. and Y.Y.; visualization,

Y.W.; supervision, Y.W. and Y.Y.; project administration, Z.L.; funding acquisition, Y.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Fundamental Research Funds for the Central Universities, grant number 2-9-2023-027.

Data Availability Statement: Interested readers who wish to access the data are requested to contact the authors via email. We will review and provide the data in accordance with our data use agreement and privacy protection policies.

Acknowledgments: We appreciate all the staff involved in the construction of Luomashi Station of Chengdu Railway Line 18.

Conflicts of Interest: Author Guangkai Huang is employed by the company China Railway 18th Bureau Group Co., Ltd. Author Zhijian Liu is employed by the company China Railway 14th Bureau Group Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as potential conflicts of interest.

References

1. Lin, D.; Broere, W.; Cui, J. Underground space utilisation and new town development: Experiences, lessons and implications. *Tunn. Undergr. Space Technol.* **2022**, *119*, 104204. [\[CrossRef\]](#)
2. He, R.; Tiong, R.L.K.; Yuan, Y.; Zhang, L. Enhancing resilience of urban underground space under floods: Current status and future directions. *Tunn. Undergr. Space Technol.* **2024**, *147*, 105674. [\[CrossRef\]](#)
3. Wang, Y.; Yang, Y.; Su, F.; Wang, L. Multiscale Analytical Method and Its Parametric Study for Lining Joint Leakage of Shield Tunnel. *Appl. Sci.* **2020**, *10*, 8528. [\[CrossRef\]](#)
4. Qi, Y.; Jia, F.; Li, W.; Shi, L.; Qin, X.; He, Y.; Li, S. Optimization and energy consumption analyses of the support system of a super large deep foundation pit in the Xi'an Metro. *Environ. Earth Sci.* **2024**, *83*, 140. [\[CrossRef\]](#)
5. Lin, D.; Zhou, Z.; Weng, M.; Broere, W.; Cui, J. Metro systems: Construction, operation and impacts. *Tunn. Undergr. Space Technol.* **2024**, *143*, 105373. [\[CrossRef\]](#)
6. Dobraszczyk, P. *Future Cities: Architecture and the Imagination*; Reaktion books: London, UK, 2019.
7. Tao, L.; Hu, G.; Jiao, B.; Guo, F.; Zhao, X.; Zhang, Y. Research on the Interaction and Deformation Control of the Adjacent Deep Foundation Pits Excavation. *J. Munic. Technol.* **2024**, *42*, 81–88, 98. [\[CrossRef\]](#)
8. Li, D.; Liao, F.; Wang, L.; Lin, J.; Wang, J. Multi-Stage and Multi-Parameter Influence Analysis of Deep Foundation Pit Excavation on Surrounding Environment. *Buildings* **2024**, *14*, 297. [\[CrossRef\]](#)
9. Chen, Y. Prediction Method for Additional Settlement of Buildings Adjacent to the Deep Foundation Pits. *J. Munic. Technol.* **2023**, *41*, 102–105. [\[CrossRef\]](#)
10. Wu, X.; Feng, Z.; Liu, J.; Chen, H.; Liu, Y. Predicting existing tunnel deformation from adjacent foundation pit construction using hybrid machine learning. *Autom. Constr.* **2024**, *165*, 105516. [\[CrossRef\]](#)
11. Zhang, W.; Zhang, R.; Wu, C.; Goh, A.T.C.; Lacasse, S.; Liu, Z.; Liu, H. State-of-the-art review of soft computing applications in underground excavations. *Geosci. Front.* **2020**, *11*, 1095–1106. [\[CrossRef\]](#)
12. Liu, J.; Shi, C.; Cao, C.; Lei, M.; Wang, Z. Improved analytical method for pile response due to foundation pit excavation. *Comput. Geotech.* **2020**, *123*, 103609. [\[CrossRef\]](#)
13. Lin, P.; Liu, P.; Ankit, G.; Singh, Y.J. Deformation Monitoring Analysis and Numerical Simulation in a Deep Foundation Pit. *Soil Mech. Found. Eng.* **2021**, *58*, 56–62. [\[CrossRef\]](#)
14. Liu, J.; Ye, J.; Shen, X.; Yu, J.; Wu, T.; Yuan, J.; Ye, Q.; Wang, S.; He, H. In-Situ Monitoring and Numerical Analysis of Deformation in Deep Foundation Pit Support: A Case Study in Taizhou. *Appl. Sci.* **2023**, *13*, 6288. [\[CrossRef\]](#)
15. Zhang, J. Finite Element Analysis of Ultra-deep Foundation Pit Covered Top-Down Excavation Based on PLAXIS. *IOP Conf. Ser. Earth Environ. Sci.* **2019**, *300*, 022158. [\[CrossRef\]](#)
16. Dong, J.; Bai, Q.; Zhao, W.; Wang, B. Test Study on the Influence of Foundation Pit Excavation on the Surface Settlement of Sandy Soil Natural Foundation of Adjacent Buildings. *Buildings* **2023**, *13*, 1293. [\[CrossRef\]](#)
17. Yuan, Z.; Gao, L.; Chen, H.; Song, S. Study on Settlement of Self-Compacting Solidified Soil in Foundation Pit Backfilling Based on GA-BP Neural Network Model. *Buildings* **2023**, *13*, 2014. [\[CrossRef\]](#)
18. Li, H.; Zhao, Z.; Du, X. Research and Application of Deformation Prediction Model for Deep Foundation Pit Based on LSTM. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 9407999. [\[CrossRef\]](#)
19. Zhou, Y.; Li, S.; Zhou, C.; Luo, H. Intelligent Approach Based on Random Forest for Safety Risk Prediction of Deep Foundation Pit in Subway Stations. *J. Comput. Civ. Eng.* **2019**, *33*, 05018004. [\[CrossRef\]](#)
20. Zhou, Y.; Su, W.; Ding, L.; Luo, H.; Love, P.E.D. Predicting Safety Risks in Deep Foundation Pits in Subway Infrastructure Projects: Support Vector Machine Approach. *J. Comput. Civ. Eng.* **2017**, *31*, 04017052. [\[CrossRef\]](#)
21. Zhang, C.; Li, J.-z.; He, Y. Application of optimized grey discrete Verhulst-BP neural network model in settlement prediction of foundation pit. *Environ. Earth Sci.* **2019**, *78*, 441. [\[CrossRef\]](#)

22. Goh, A.T.C.; Wong, K.S.; Broms, B.B. Estimation of lateral wall movements in braced excavations using neural networks. *Can. Geotech. J.* **1995**, *32*, 1059–1064. [[CrossRef](#)]
23. Huang, H.-X.; Li, J.-C.; Xiao, C.-L. A proposed iteration optimization approach integrating backpropagation neural network with genetic algorithm. *Expert Syst. Appl.* **2015**, *42*, 146–155. [[CrossRef](#)]
24. Wu, H.; Bian, K.; Qiu, J.; Ye, X.; Chen, C.; Fu, B. The prediction of foundation pit based on genetic back propagation neural network. *J. Comput. Methods Sci. Eng.* **2019**, *19*, 707–717. [[CrossRef](#)]
25. Zhang, W.-S.; Yuan, Y.; Long, M.; Yao, R.-H.; Jia, L.; Liu, M. Prediction of surface settlement around subway foundation pits based on spatiotemporal characteristics and deep learning models. *Comput. Geotech.* **2024**, *168*, 106149. [[CrossRef](#)]
26. Bentéjac, C.; Csörgő, A.; Martínez-Muñoz, G. A comparative analysis of gradient boosting algorithms. *Artif. Intell. Rev.* **2021**, *54*, 1937–1967. [[CrossRef](#)]
27. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
28. Wang, Y.; Su, F.; Guo, Y.; Yang, H.; Ye, Z.; Wang, L. Predicting the microbiologically induced concrete corrosion in sewer based on XGBoost algorithm. *Case Stud. Constr. Mater.* **2022**, *17*, e01649. [[CrossRef](#)]
29. Fouad, K.M.; Ismail, M.M.; Azar, A.T.; Arafa, M.M. Advanced methods for missing values imputation based on similarity learning. *PeerJ Comput. Sci.* **2021**, *7*, e619. [[CrossRef](#)] [[PubMed](#)]
30. Ren, L.; Wang, T.; Seklouli, A.S.; Zhang, H.; Bouras, A. A review on missing values for main challenges and methods. *Inf. Syst.* **2023**, *119*, 102268. [[CrossRef](#)]
31. Benesty, J.; Chen, J.; Huang, Y. On the importance of the Pearson correlation coefficient in noise reduction. *IEEE Trans. Audio Speech Lang. Process.* **2008**, *16*, 757–765. [[CrossRef](#)]
32. Ben Jabeur, S.; Stef, N.; Carmona, P. Bankruptcy prediction using the XGBoost algorithm and variable importance feature engineering. *Comput. Econ.* **2023**, *61*, 715–741. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.