

## Article

# Intelligent Prediction of Maximum Ground Settlement Induced by EPB Shield Tunneling Using Automated Machine Learning Techniques

Syed Mujtaba Hussaine <sup>1</sup>  and Linlong Mu <sup>2,\*</sup><sup>1</sup> College of Civil Engineering, Tongji University, Shanghai 200092, China<sup>2</sup> Department of Geotechnical Engineering, Tongji University, Shanghai 200092, China

\* Correspondence: mulinlong@tongji.edu.cn; Tel.: +86-21-65982005

**Abstract:** Predicting the maximum ground subsidence ( $S_{max}$ ) in the construction of soil pressure balanced shield tunnel, particularly on soft foundation soils, is essential for safe operation and to minimize the possible risk of damage in urban areas. Although some research has been done, this issue has not been solved because of its complexity and many other influencing factors. Due to the increasing accuracy of machine learning (ML) in predicting surface deformation of shield tunneling and the development of automated machine learning (AutoML) technology. In the study, different ML prediction models were constructed using an open source AutoML framework. The prediction model was trained by the dataset, which contains 14 input parameters and an output (i.e.,  $S_{max}$ ). Different AutoML frameworks were employed to compare their validities and efficiencies. The performance of the model is estimated by contrasting the prediction accuracy parameters, including root mean square error (RMSE), mean absolute error (MAE) and determinant coefficient ( $R^2$ ). With a coefficient of determination ( $R^2$ ) of 0.808, MAE of 3.7, and RMSE of 5.2 on the testing dataset, the best prediction model i.e., extra tree regressor showed better performance, proving that our model has advantages in predicting  $S_{max}$ . Furthermore, the SHAP analysis reveal that the soil type (ST), torque ( $T_o$ ), cover depth (H), groundwater level (GW), and tunneling deviation have a significant effect on  $S_{max}$  compared to other model inputs.

**Keywords:** maximum surface settlement; tunneling; auto machine learning; feature selection; shapley additive explanations (SHAP) analysis

**MSC:** 65Z05

**Citation:** Hussaine, S.M.; Mu, L. Intelligent Prediction of Maximum Ground Settlement Induced by EPB Shield Tunneling Using Automated Machine Learning Techniques. *Mathematics* **2022**, *10*, 4637. <https://doi.org/10.3390/math10244637>

Academic Editors: Danial Jahed Armaghani, Hadi Khabbaz, Manoj Khandelwal, Niaz Muhammad Shahani and Ramesh Murlidhar Bhatawdekar

Received: 11 October 2022

Accepted: 1 December 2022

Published: 7 December 2022

Corrected: 12 April 2023

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

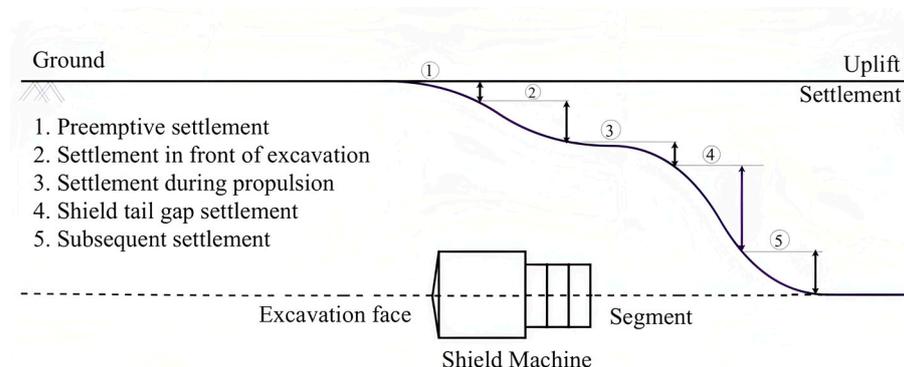


**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

With the acceleration of urban construction, the construction of subway networks has become one of the most practical methods to alleviate traffic jam and shortages of land resources [1–5]. These excavation systems are generally built as twin tunnels and the excavation is carried out through soft soils or weak rocks at shallow depths. For urban subway tunnels, the shield tunneling method (especially the earth pressure balance (EPB) shield tunneling) is one of the most widely used construction methods due to its little impact on the surrounding environment. The advantages of less influence and a high degree of mechanization are widely applied to the actual engineering projects. However, in weak strata, the shield tunneling method can still cause a lot of land subsidence [6,7]. The surface subsidence mechanism [8,9] and development process caused by shield tunneling is complex, which can be seen in Figure 1, including (1) preemptive settlement; (2) settlement in front of excavation; (3) settling during propulsion; (4) shield tail gap settlement; and (5) subsequent settlement. Each stage's surface subsidence involves geological conditions, shield parameters, on-site construction, and other factors. Predicting surface deforma-

tion during the shield construction process reasonably and accurately has always been a problematic issue in research.



**Figure 1.** Schematic diagram of longitudinal settlement caused by shield tunneling.

The ground settlement caused by shield tunneling, apart from empirical, traditional theoretical calculations, numerical simulations, and other research methods, has been analyzed [10,11]. The empirical formula [12,13] describes the general ground subsidence caused by shield tunneling because the geological conditions in different regions are quite different, and the numerical value of the parameters varies widely. However, the empirical models adopted in engineering often ignore the influence of parameters used to adjust the settlement during shield tunneling. Therefore, the accuracy of surface subsidence prediction based on the empirical formula method is unacceptable. Due to the limitations of empirical methods, many studies have proposed analytical methods to estimate the settlement induced by shield excavation [14,15].

In the analytical method, it is difficult for simplified computational models to accurately account for the complex interactions between shield and soil, which affects the application of the analytical method in practical engineering problems [16]. Compared with empirical and analytical methods, numerical simulation methods can simulate the dynamic construction process of shield tunnels and comprehensively consider the interaction between tunnel construction and soil layers [17]. However, calculating the numerical model is time-consuming, and the constitutive model is difficult to accurately simulate the response of the soil layer on the macroscopic scale [2,18,19].

In shield tunneling, which is a dynamic process, the tunneling parameters and geological parameters change in real-time, and the surface subsidence due to shield excavation can be predicted in real-time. The parameter adjustment plan can be given to guide the shield tunneling in an absolute sense. In the construction process, traditional methods are difficult to achieve in this regard. Machine learning algorithms have developed rapidly in recent years and are gradually being applied in geotechnical engineering due to their nonlinear solid fitting capabilities and the simultaneous consideration of the influence of multiple parameters [20]. Because machine algorithms can obtain accurate results quickly, machine learning algorithms provide new ideas for intelligently controlling the shield tunneling process. Regarding the prediction of surface subsidence caused by shield tunneling, the widely used machine learning algorithms include artificial neural networks (ANN) and support vector machines (SVM). Recent research shows that ML methods have great application prospects in analyzing complex geotechnical problems, such as deformation caused by landslide [21,22] and underground soil structure interaction caused by tunnel excavation [23]. In early investigations, Shi et al. [24] used the artificial neural network method (ANNs) to calculate the maximum ground settlement due to shield tunneling accurately. In addition, the same method is also used to calculate the width of the settlement tank induced by shield excavation. Suwansawat et al. [25] systematically expounded the application of artificial neural network method in earth pressure balance shield tunnel based on a substantial amount of measured engineering data. Santos et al. [26] obtained the correlation between the excavation parameters and the ground subsidence based on

the artificial neural network model, which fits the actual theoretical results. Many studies have combined a variety of optimization methods, for instance, genetic algorithms, particle swarm algorithms, with ANNs to optimize the accuracy of the prediction model [27,28].

However, a significant challenge in using ANN is to determine the optimal network framework [29]. In addition, due to its complex nature, the output from an ANN model is usually inexplicable; therefore, complicated ML models such as ANN are often referred to as a “black box” model. Zhang et al. [30] accurately predicted the development law of the ground subsidence due to shield excavation by integrating the wavelet function and the support vector machine algorithm. The study of machine learning methods to predict ground subsidence caused by shield tunneling is shown in Table 1.

**Table 1.** Development and Application of Machine Learning Algorithms in Shield Tunnels.

| Related Literature     | Method      | Output Parameters | Data Points |
|------------------------|-------------|-------------------|-------------|
| Shi (1998) [24]        | BP          | $S_c, S_i, S_f$   | 356         |
| Suwansawat (2006) [31] | BP          | $G$               | 49          |
| Santos (2008) [26]     | BP          | $G$               | 81          |
| Darabi (2012) [32]     | BP          | $G$               | 53          |
| Pourtaghi (2012) [33]  | Wavelet, BP | $G$               | 49          |
| Ahangari (2015) [28]   | ANFIS, GEP  | $G$               | 53          |
| Zhou (2016) [34]       | RF          | $G$               | 66          |
| Bouayad (2017) [27]    | ANFIS       | $G$               | 95          |
| Zhang (2017) [30]      | LSSVM       | $G$               | 55          |

Note:  $G$  = surface subsidence;  $S_c$  = Surface subsidence when passing through the monitoring section;  $S_i$  = Surface subsidence after the completion of the monitoring section segment assembly;  $S_f$  = Surface subsidence after stabilization.

Random Forest (RF) is another integrated ML algorithm that can process a large amount of data in a short time. The final prediction result integrates multiple embedded calculation results with high accuracy and is used to calculate the settlement caused by shield tunnel construction [34]. Shao et al. [35] optimized the ANN model through the particle swarm optimization (PSO) method and founded the optimum transfer speed of the screw conveyor to ensure the safety of the tunnel face. In order to guarantee the tunneling efficiency of the shield tunneling machine, Armaghani et al. [36] proposed the use of PSO-ANN and the Imperial Competitive Algorithm (ICA)-ANN method to estimate the tunnel speed of the shield tunneling facility. At the same time, the method of PSO-SVM is also applied to calculate and improve the tunnel parameters of the shield machine during the tunnel construction. At the same time, the method of PSO-SVM is also applied to calculate and improve the tunnel parameters of the shield machine during the tunnel construction [37]. However, there are many new machine learning algorithms at this stage, and the prediction performance of different algorithms is different.

Although abundant highly effective studies have been introduced above, there is still a lack of research on performance differences of different machine learning algorithms in predicting the maximum ground subsidence due to shield tunneling; secondly, the current research mainly focuses on the final output results, and there is a lack of research on the correlation between input and output parameters. Therefore, constructing an interpretable ML model can reveal the connection between input and output parameters, thereby helping engineering designers to make the best decisions to ensure that soil settlement is limited within the expected range throughout the construction process. At present, a feature selection method, that is, the Pearson correlation method, has been used to detect and control the influencing parameters of the surface settlement caused by the tunnel excavation process. But the defect of this method is that it can only consider the linear relationship between two parameters, while ignoring the influence of feature interaction between parameters [6]. So academia began to use explainable artificial intelligence (XAI) to study this problem. It allows humans to understand the output of complex ML models [38]. The Shapley Additive Interpretation (SHAP) proposed by [39], is one such XAI-based

algorithm. The SHAP method can measure how each input feature affects the dependent variable (output).

Owing to the importance of predicting settlements due to shield tunneling in geotechnical engineering, more and more people are trying to use machine learning algorithms to build predictive models that can accurately estimate influencing variables. Currently, selecting a suitable model requires the process of sample characterization, parameter fine-tuning, and configuration comparison. These steps are complicated and difficult for non-experts in machine learning to follow. For this reason, the research of automatic machine learning (AutoML) has attracted more and more attention. The advantage of AutoML is that it can automatically match the most suitable model and hyperparameters on the basis of complex datasets, thus simplifying the process of selecting the best model and optimize the performance of the model. On the whole, the structure of our study is organized as follows. Firstly, the database and data pre-processing methods we utilize are explained. Secondly, this study compares the differences of two feature selection methods (Pearson correlation and SHAP algorithm) in analyzing the same project datasets collected from two EPB tunnel projects completed in Hangzhou, China. The SHAP algorithm is applied to analyze the impact of the input feature parameters on the overall prediction results. In the end, considering the advantages of AutoML, based on the AutoML method, this research uses the PyCaret [40], a low-code machine learning library to construct a shield tunnel prediction model based on monitored data. Subsequently, a comparative analysis of various types of developed ML methods was accomplished to evaluate their performance and select the best-performing model in this problem, and remarkable conclusions are ultimately summarized.

## 2. Establishment of Surface Deformation Database for Shield Tunneling

### 2.1. Project Overview

The dataset used in this research was collected from two metro line tunneling projects in Hangzhou, China [41]. As shown in Figure 2, metro line two (project-1) was excavated from Gucui station to Xueyuan station, while metro line six (project-2) was excavated from Shangpu station to Heshan Road station. Figure 3 outlines the construction plan implemented during the excavation of Projects 1 and 2. The twin tunnels excavated for Project 1 (denoted as downlink and uplink in Figure 3a) were initiated in January 2016 and completed in June 2016. The twin tunnels considered in Project 2 (namely, the left and right tunnels in Figure 3b) commenced on 15 April 2017, and were completed on 15 October 2017. The downlink of Project 1 and both tunnels in Project 2 were excavated using two “Shichuandao” type EPB shield machines. In contrast, the “Kawasaki” EPB shield was used to excavate the uplink of Project 1. The inner diameter of each twin tunnel in both projects was 5.5 m, while the outer diameter was 6.2 m. The total excavation length of the twin tunnels for Project 1 was 1950 m, and that for Project 2 was 2486 m. Note to avoid any effect of secondary disturbance due to the second excavation; this analysis only considered the data from the first excavation of each project (i.e., downlink in Project 1 and left tunnel in Project 2).

### 2.2. Engineering Geology

To determine the geological conditions at the proposed site, the construction unit conducted comprehensive field and laboratory testing. The cross-sectional geological profile of the tunnel section observed in Projects 1 and 2 is shown in Figure 4, which shows the main soil layers of the site, including soil fill (mixed soil and pure soil), sandy silt, silty sandy silt, sandy silt, silt, boulders1, silty silt, silty clay, and boulders 2 observed at a depth of around 30 m. From Figure 4a, it can be seen that Project 1 started excavation from the downlink route, covering a depth of 10.6 m to 18.7 m. The soil layer of this route is mainly muddy silty clay and muddy clay. Project 2 is excavated from the left line and covers a depth of 9 m to 16.6 m, passing through sandy silt and silt layers, as shown in Figure 4b.

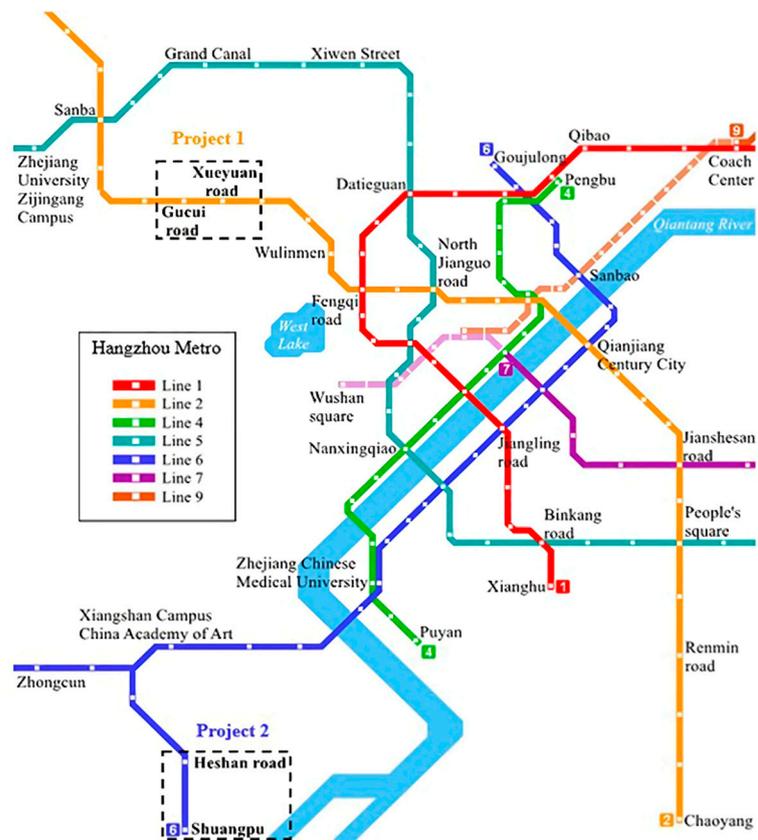


Figure 2. Hangzhou metro system map [41].

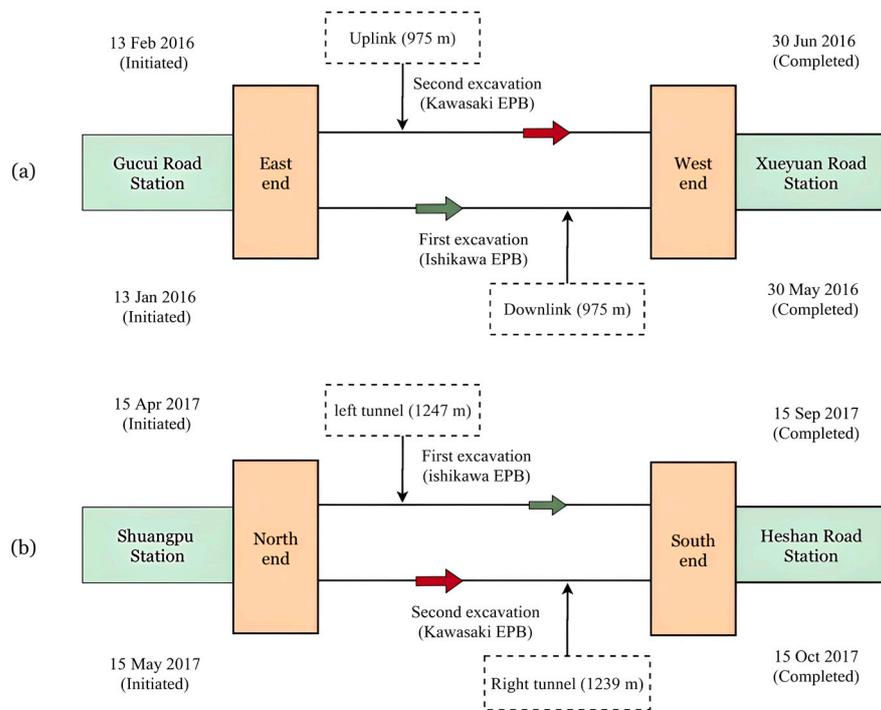
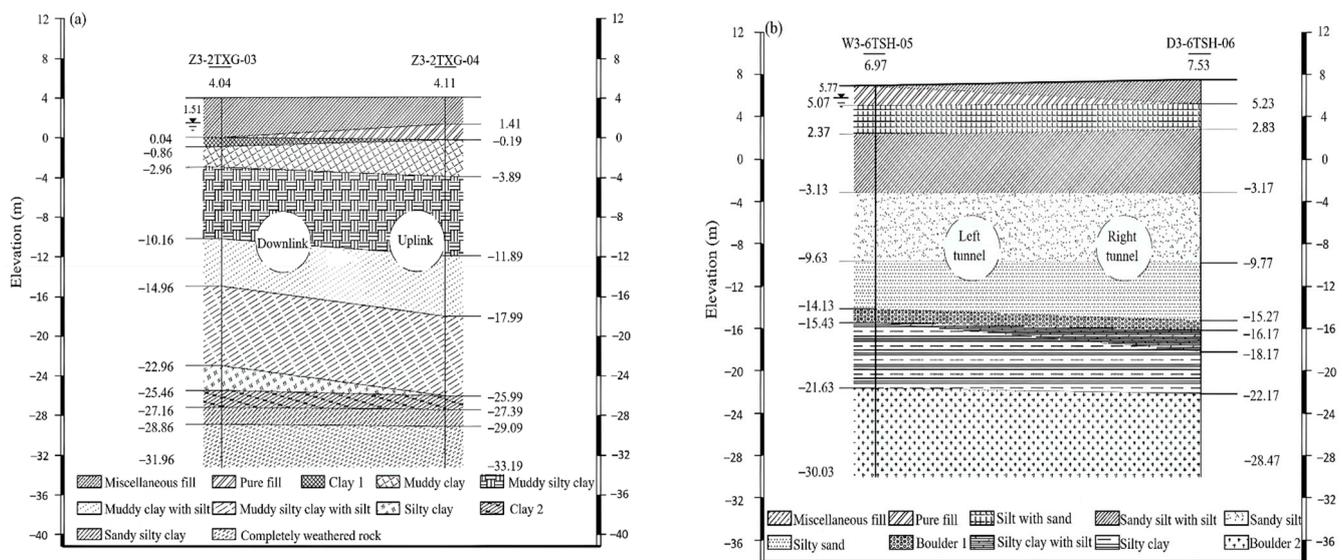


Figure 3. Construction plan of twin tunnels for (a) Project 1 and (b) Project 2 [41].



**Figure 4.** Cross-sectional geological profiles for (a) Project 1 and (b) Project 2 (unit: m) (Kannan-gara et al., 2022) [41].

Based on the Chinese National Standard (CNS) GB/T50123-1999 (standard for soil test methods) [42], the laboratory tests was carried out to measure the physical and mechanical properties of the soil layers of the project 1 and project 2, as shown in Table 2. The shear strength parameters (i.e.,  $c$  and  $\varphi$ ) of the soil can be measured through a series of direct shear tests. The direct shear tests require the soil samples to be pre-consolidated for 24 hours and sheared rapidly (0.8–1.2 mm/min) under undrained conditions. The average groundwater levels of Project 1 and Project 2 were  $-2.14$  m and  $-1.8$  m, respectively. It is worth noting that the groundwater levels remained stable during excavation.

**Table 2.** Soil physical properties of projects (1 and 2) [41].

| Project   | Soil Type                  | $\gamma$ (kN/m <sup>3</sup> ) | $\varphi^\circ$ | $c$ (kPa) | $G_s$ | $e$   |
|-----------|----------------------------|-------------------------------|-----------------|-----------|-------|-------|
| 1         | Miscellaneous fill         | (18)                          |                 |           |       |       |
|           | Pure fill                  | (18.5)                        |                 |           |       |       |
|           | Clay 1                     | 18.2                          | 10              | 12        | 2.74  | 1.095 |
|           | Muddy clay                 | 17.6                          | 13              | 10        | 2.73  | 1.247 |
|           | Muddy silty clay           | 17.6                          | 14              | 10        | 2.72  | 1.218 |
|           | Muddy clay with silt       | 17.5                          | 14              | 11        | 2.72  | 1.22  |
|           | Muddy silty clay with silt | 18.1                          | 18              | 12        | 2.71  | 1.067 |
|           | Silty clay                 | 17.6                          | 14              | 12        | 2.73  | 1.204 |
|           | Clay 2                     | 17.4                          | 12              | 15        | 2.74  | 1.243 |
|           | Sandy silty clay           | 20.2                          | 22              | 14        | 2.69  | 0.608 |
| 2         | Completely weathered rock  |                               |                 |           |       |       |
|           | Miscellaneous fill         | (18)                          |                 |           |       |       |
|           | Pure fill                  | (17.5)                        |                 |           |       |       |
|           | Silt with sand             | 19.4                          | 26              | 8         | 2.69  | 0.768 |
|           | Sandy silt with silt       | 19.5                          | 28              | 5.5       | 2.69  | 0.742 |
|           | Sandy silt                 | 19.7                          | 29              | 4.5       | 2.68  | 0.706 |
|           | Silty sand                 | 19.7                          | 31.5            | 4         | 2.68  | 0.687 |
|           | Boulder 1                  |                               | 36              | 5         |       |       |
|           | Silty clay with silt       | 17.1                          | 13              | 14        | 2.71  | 1.283 |
|           | Silty clay                 | 20.1                          | 21              | 28        | 2.71  | 0.66  |
| Boulder 2 |                            | 40                            | 6               |           |       |       |

Note:  $\gamma$  is the unit weight of soil,  $\varphi$  is the soil internal frictional angle,  $c$  is the cohesion of the soil,  $G_s$  is the specific gravity, and  $e$  is the void ratio. Data within round brackets are the empirical values.

### 2.3. Preliminary Selection of Input Parameters

Previous studies discovered that in the shield tunneling process, the main factors affecting the surface deformation could be roughly divided into three categories [31,43]: (1) tunnel geometric parameters (such as tunnel burial depth, shield diameter, section form, etc. [44,45]); (2) stratum parameters (such as cover soil type, face soil type, soil compressive modulus, elastic modulus, cohesion, internal friction angle, groundwater level, etc.); (3) shield construction parameters (shield thrust, advanced rate, shield attitude, cutter head torque, thrust, jack pressure, horizontal deviation (front), vertical deviation (front), horizontal deviation (back), vertical deviation (back) [46], grouting pressure, grouting volume, etc.). The 14 input features by their respective categories and the target variable (i.e.,  $S_{max}$ ) as shown in Table 3 are considered for the analysis.

**Table 3.** List of input features and target variable considered for analysis [41].

| Category                      | Parameters                          | Symbol    | Unit   |
|-------------------------------|-------------------------------------|-----------|--------|
| Tunnel geometry               | Cover depth                         | H         | m      |
| Geological conditions         | Soil type <sup>a</sup>              | ST        | -      |
|                               | Groundwater level                   | GW        | m      |
| Shield operational parameters | Face pressure (top)                 | FPt       | kPa    |
|                               | Face pressure (center) <sup>b</sup> | FPc       | kPa    |
|                               | Advance rate                        | AR        | mm/min |
|                               | Pitching angle                      | PA        | °      |
|                               | Thrust                              | Th        | kN     |
|                               | Torque                              | To        | kN m   |
|                               | Jack pressure                       | JP        | kPa    |
|                               | Horizontal deviation (front)        | HDf       | mm     |
|                               | Vertical deviation (front)          | VDf       | mm     |
|                               | Horizontal deviation (back)         | HDb       | mm     |
| Vertical deviation (back)     | VDb                                 | mm        |        |
| Target variable               | Maximum surface settlement          | $S_{max}$ | mm     |

<sup>a</sup> Categorical feature. <sup>b</sup> Computed by taking the average of face pressures recorded at left and right positions.

In order to observe the ground subsidence, an optical level (Suguang DS05, China, accuracy 0.5 mm/km) and an electronic level (Trimble DINI 03, USA, 0.3 mm/km) were used to measure the site subsidence. Surface settlements were measured twice daily, once at 8:00 a.m. and again at 4:00 p.m. The allowable values for the surface settlement and uplift were set at 35 mm and 10 mm, respectively.

Since the specifications of the entire tunnel are the same, the burial depth and diameter of the tunnel (D) are constantly changing. Considering that the buried depth and diameter of the tunnel will affect the development model of the stratum subsidence and the size of the final settlement during the shield tunnel process [31,47], these two parameters are selected as the only geometric parameters. Since the tunnels in this study were constructed by shield tunneling, their outer diameters are the same, both are 6.2 m, the influence of parameter D can be ignored.

Geological parameters include the depth of groundwater (GW) level, and physical and mechanical properties of rock and soil. In machine learning algorithms, the geological parameters need to be quantified. The physical and mechanical properties of the rock and soil layer, along with the thickness and location of the soil layer, will influence the subsidence induced by shield tunnel. Commonly used is the direct input of the soil layer  $c$ ,  $\varphi$  value [28,30], or directly using numbers to indicate the type of soil layer (ST) [35]. The soil types mainly traversed during the EPB shield excavation for the two projects in this study were “silty clay” and “silt sand”. For the convenience of description, they are coded as 0 and 1 respectively.

In this study, a total of 11 shield operating parameters were considered as the input features of the model. The four operational parameters, i.e., thrust, torque, tunneling rate, and jack pressure, affect the degree of disturbance to the stratum during the shield

tunneling process [48]. The soil pressure will affect the stability of the tunnel face [49,50]. Project 2 uses the “Ishikawa” EPB shield. In order to measure the surface pressure during its working process, three earth pressure gauges were installed on the top, left and right sides of the shield machine [51]. The “face pressure (top)” and “face pressure (center)” are used as input parameters to analyze the effect of face pressure on the settlement caused by excavation. The shield machine must advance strictly along the design route (DTA) during the working process. The attitude and position of the shield machine are described by vertical deviation (front), horizontal deviation (front), vertical deviation (back), horizontal deviation (back), rolling angle, and pitch angle [52]. The pitch and rolling angles describe the attitude of the shield machine relative to the horizontal and vertical axes, respectively. For each parameter taken into account in the dataset, the corresponding symbol and its unit are displayed in Table 3. It is to be noted that the data preparation process were carried out as recommended by kannangara et. al. [41] and the data is further refined as explained in Section 2.4 below.

2.4. Data Pre-Processing

A major problem of machine learning prediction models is that the learning curve is difficult to converge. In order to improve the probability of curve convergence, the data set must be preprocessed to reduce data inconsistency [53]. In the cause of probe critical information from the shield-soil interplay for surface subsidence prediction, a total of 264 data samples were collected, which were further divided into two subsets to evaluate a model’s generalization ability. Randomly select 80% of the samples in the constructed data sample library as the training and testing set of the model (211 observations per feature). It should be noted that the test set must be referred to evaluate the model’s behavior. The remaining 20% (53 data samples) have been retained from the basic dataset to be adopted for predictions, the data should not be confused with a training/test segmentation. According to the 264 surface subsidence measurement data chosen in this study, the input and output data of first 25 points are shown in Table 4. The limits of mentioned parameters to construct the predictive models for all 264 data samples, including average, standard deviation (Std.), maximum (Max.), minimum (Min.) and three percentiles (75%, 50%, and 25%) are summarized in Table 5.

Table 4. Dataset samples used for creating intelligent model.

| No. | Ring | H (m) | ST | GW (m) | FPt (kPa) | FPc (kPa) | AR (mm/min) | PA (°) | Th (kN) | To (kN/m) | JP (kPa) | HD (mm) | VDF (mm) | HD (mm) | VDB (mm) | Smax (mm) |
|-----|------|-------|----|--------|-----------|-----------|-------------|--------|---------|-----------|----------|---------|----------|---------|----------|-----------|
| 1   | 5    | 9.03  | 1  | 1.46   | 40        | 95        | 0           | −0.1   | 9345    | 1937      | 8700     | −34     | 53       | 27      | −53      | 4.65      |
| 2   | 9    | 9.05  | 1  | 1.57   | 0         | 70        | 7           | −0.22  | 27,124  | 1305      | 24,600   | −63     | −67      | 5       | −55      | 5.52      |
| 3   | 14   | 9.07  | 1  | 1.68   | 80        | 140       | 31          | 0      | 19,986  | 2310      | 17,700   | −80     | −43      | −23     | −62      | 40.11     |
| 4   | 18   | 9.09  | 1  | 1.79   | 110       | 180       | 59          | −0.1   | 16,804  | 1965      | 4700     | −78     | −43      | −46     | −48      | 8.8       |
| 5   | 22   | 9.1   | 1  | 1.9    | 110       | 180       | 45          | −0.2   | 20,275  | 1937      | 18,500   | −49     | −31      | −62     | −44      | 8.76      |
| 6   | 26   | 9.13  | 1  | 2.01   | 120       | 190       | 32          | −0.2   | 17,478  | 2529      | 16,075   | −37     | −17      | −54     | 42       | 18.67     |
| 7   | 30   | 9.25  | 1  | 2.12   | 110       | 180       | 34          | −0.6   | 18,907  | 2289      | null     | −31     | −45      | −37     | −15      | 16.16     |
| 8   | 34   | 9.36  | 1  | 2.22   | 110       | 195       | 30          | −0.77  | 17,459  | 2567      | 16,200   | −32     | −46      | −29     | −21      | 6.45      |
| 9   | 39   | null  | 1  | 2.33   | 120       | 190       | 42          | −0.7   | 19,564  | 2036      | 18,050   | −15     | −65      | −20     | −51      | 2.41      |
| 10  | 43   | 9.6   | 1  | 2.44   | 110       | 170       | 29          | −0.7   | null    | 2874      | 18,250   | −8      | −53      | −13     | −57      | 3.18      |
| 11  | 51   | 9.83  | 1  | 2.66   | 130       | 205       | 36          | −1     | 19,344  | 2853      | 17,800   | −1      | −55      | 5       | −46      | 1.58      |
| 12  | 55   | 9.94  | 1  | 2.49   | 130       | 205       | 48          | −1     | 19,726  | 2153      | 18,000   | 1       | −58      | 17      | −55      | 7.61      |
| 13  | 59   | 10.04 | 1  | 2.33   | 130       | 205       | 41          | −1     | 17,758  | 2250      | 16,300   | 14      | −47      | 16      | −49      | 10.12     |
| 14  | 64   | 10.14 | 1  | 2.16   | 130       | 200       | 38          | −1     | 18,297  | 2778      | 16,900   | −18     | −45      | −8      | −44      | 11.77     |
| 15  | 68   | 10.24 | 1  | 2      | 130       | 205       | 38          | −1.1   | 18,597  | 2657      | 16,975   | 4       | −35      | 7       | −31      | 12.97     |
| 16  | 72   | 10.34 | 1  | 1.84   | 120       | 190       | 35          | −1.2   | 18,693  | 2278      | 17,225   | −18     | −39      | 16      | −13      | 15.45     |
| 17  | 76   | 10.43 | 1  | 1.67   | 130       | 205       | 46          | −1.2   | 17,618  | 2095      | 15,750   | −42     | −44      | −5      | −14      | 21.3      |
| 18  | 80   | 10.53 | 1  | 1.51   | 130       | 200       | 45          | null   | 17,885  | 1953      | 15,775   | −31     | −47      | −29     | −19      | 16.11     |
| 19  | 84   | null  | 1  | 1.34   | 130       | 200       | 43          | −1.1   | 18,490  | 2567      | 16,900   | −15     | −47      | −30     | −33      | 11.6      |
| 20  | 89   | 10.73 | 1  | 1.21   | 140       | 205       | 44          | −1.1   | 18,923  | 2049      | 17,400   | −18     | −39      | −16     | −32      | 14.35     |
| 21  | 50   | 10.91 | 0  | 112.0  | 60        | 160       | 51          | −1.33  | 10,655  | 481       | 9500     | 13      | −55      | 10      | −4       | 12.1      |
| 22  | 55   | 11.05 | 0  | 240.0  | 50        | 170       | 50          | −1.42  | 11,270  | 506       | 10,100   | 29      | −48      | 42      | 2        | 16.7      |
| 23  | 85   | 11.89 | 0  | 12.2   | 50        | 190       | 62          | −1.49  | 10,307  | 518       | 9100     | 4       | −87      | 17      | −31      | 26.9      |
| 24  | 90   | 12.03 | 0  | 11.9   | 60        | 215       | 63          | −1.17  | 10,703  | 522       | 9525     | 21      | −69      | 35      | −60      | 28.5      |
| 25  | 100  | 12.31 | 0  | 32.4   | 40        | 170       | 57          | −1.31  | 12,307  | 569       | 10,875   | −22     | −66      | 47      | −40      | 40.2      |

**Table 5.** Descriptive statistical description of the dataset used.

| Parameter Count | Count | Mean Count | Std. Count | Min. Count | 25% Count | 50% Count | 75% Count | Max. Count |
|-----------------|-------|------------|------------|------------|-----------|-----------|-----------|------------|
| H               | 264   | 14.5       | 2.7        | 9.03       | 11.98     | 15.07     | 16.71     | 18.70      |
| ST              | 264   | 0.52       | 0.5        | 0          | 0         | 1         | 1         | 1          |
| GW              | 264   | 1.96       | 0.6        | 0.36       | 1.63      | 1.93      | 2.40      | 3.18       |
| FPt             | 264   | 122.6      | 62.12      | 0          | 70        | 110       | 182.5     | 230        |
| FPc             | 264   | 232.3      | 37         | 70         | 205       | 240       | 260       | 310        |
| AR              | 264   | 58.40      | 11.76      | 0          | 53        | 60        | 66        | 80         |
| PA              | 264   | −0.09      | 0.78       | −1.49      | −0.77     | −0.20     | 0.38      | 1.37       |
| Th              | 264   | 19,592.6   | 4404.27    | 0          | 17,194.0  | 19,331.0  | 23,280.0  | 27433.0    |
| To              | 264   | 1537.85    | 956.04     | 0          | 569.75    | 19,210.0  | 2481.5    | 3180       |
| JP              | 264   | 17,862.2   | 3992.54    | 25         | 15,750.0  | 17,850.0  | 21,131.25 | 24950.0    |
| HDf             | 264   | −8.74      | 23.70      | −80        | −22.25    | −12       | 2.25      | 69         |
| Vdf             | 264   | −47.14     | 39.57      | −125       | −76       | −48       | −14       | 36         |
| HDb             | 264   | 22.97      | 25.57      | −62        | 8         | 23        | 39.25     | 107        |
| VDb             | 264   | −25.07     | 35.80      | −126       | −51       | −26       | −4        | 54         |
| Smax            | 264   | 20.87      | 12.48      | 1.58       | 11.225    | 16.95     | 28082     | 55.30      |

For data cleansing, the shield parameters obtained from the shield site often contain many invalid data and cannot be used directly, so the data must be cleaned. PyCaret by default utilizes the drop\_duplicates () function for the cleaning process, which includes the removal of nulls and outlier rejection. Table 6 lists the data samples obtained after cleansing. We performed all analyses using the default settings; for example, the test/hold-out set was 80/20, with 10-fold cross-validation for model comparison. The preprocessing methods that were employed are discussed next.

**Table 6.** Cleaned dataset samples used for creating intelligent model.

| No. | Ring | H (m) | ST | GW (m) | FPt (kPa) | FPc (kPa) | AR (mm/min) | PA (°) | Th (kN) | To (kN/m) | JP (kPa) | HD (mm) | VDF (mm) | HD (mm) | VDB (mm) | Smax (mm) |
|-----|------|-------|----|--------|-----------|-----------|-------------|--------|---------|-----------|----------|---------|----------|---------|----------|-----------|
| 1   | 5    | 9.03  | 1  | 1.46   | 40        | 95        | 0           | −0.1   | 9345    | 1937      | 8700     | −34     | 53       | 27      | −53      | 4.65      |
| 2   | 9    | 9.05  | 1  | 1.57   | 0         | 70        | 7           | −0.2   | 27,124  | 1305      | 24,600   | −63     | −67      | 5       | −55      | 5.52      |
| 3   | 14   | 9.07  | 1  | 1.68   | 80        | 140       | 31          | 0      | 19,986  | 2310      | 17,700   | −80     | −43      | −23     | −62      | 40.11     |
| 4   | 18   | 9.09  | 1  | 1.79   | 110       | 180       | 59          | −0.1   | 16,804  | 1965      | 4700     | −78     | −43      | −46     | −48      | 8.8       |
| 5   | 22   | 9.1   | 1  | 1.9    | 110       | 180       | 45          | −0.2   | 20,275  | 1937      | 18,500   | −49     | −31      | −62     | −44      | 8.76      |
| 6   | 26   | 9.13  | 1  | 2.01   | 120       | 190       | 32          | −0.2   | 17,478  | 2529      | 16,075   | −37     | −17      | −54     | −42      | 18.67     |
| 7   | 30   | 9.25  | 1  | 2.12   | 110       | 180       | 34          | −0.6   | 18,907  | 2289      | 17,950   | −31     | −45      | −37     | −15      | 16.16     |
| 8   | 34   | 9.36  | 1  | 2.22   | 110       | 195       | 30          | −0.77  | 17,459  | 2567      | 16,200   | −32     | −46      | −29     | −21      | 6.45      |
| 9   | 39   | 9.48  | 1  | 2.33   | 120       | 190       | 42          | −0.7   | 19,564  | 2036      | 18,050   | −15     | −65      | −20     | −51      | 2.41      |
| 10  | 43   | 9.6   | 1  | 2.44   | 110       | 170       | 29          | −0.7   | 19,778  | 2874      | 18,250   | −8      | −53      | −13     | −57      | 3.18      |
| 11  | 51   | 9.83  | 1  | 2.66   | 130       | 205       | 36          | −1     | 19,344  | 2853      | 17,800   | −1      | −55      | 5       | −46      | 1.58      |
| 12  | 55   | 9.94  | 1  | 2.49   | 130       | 205       | 48          | −1     | 19,726  | 2153      | 18,000   | 1       | −58      | 17      | −55      | 7.61      |
| 13  | 59   | 10.04 | 1  | 2.33   | 130       | 205       | 41          | −1     | 17,758  | 2250      | 16,300   | 14      | −47      | 16      | −49      | 10.12     |
| 14  | 64   | 10.14 | 1  | 2.16   | 130       | 200       | 38          | −1     | 18,297  | 2778      | 16,900   | −18     | −45      | −8      | −44      | 11.77     |
| 15  | 68   | 10.24 | 1  | 2      | 130       | 205       | 38          | −1.1   | 18,597  | 2657      | 16,975   | 4       | −35      | 7       | −31      | 12.97     |
| 16  | 72   | 10.34 | 1  | 1.84   | 120       | 190       | 35          | −1.2   | 18,693  | 2278      | 17,225   | −18     | −39      | 16      | −13      | 15.45     |
| 17  | 76   | 10.43 | 1  | 1.67   | 130       | 205       | 46          | −1.2   | 17,618  | 2095      | 15,750   | −42     | −44      | −5      | −14      | 21.3      |
| 18  | 80   | 10.53 | 1  | 1.51   | 130       | 200       | 45          | −1.2   | 17,885  | 1953      | 15,775   | −31     | −47      | −29     | −19      | 16.11     |
| 19  | 84   | 10.63 | 1  | 1.34   | 130       | 200       | 43          | −1.1   | 18,490  | 2567      | 16,900   | −15     | −47      | −30     | −33      | 11.6      |
| 20  | 89   | 10.73 | 1  | 1.21   | 140       | 205       | 44          | −1.1   | 18,923  | 2049      | 17,400   | −18     | −39      | −16     | −32      | 14.35     |
| 21  | 50   | 10.91 | 0  | 112.0  | 60        | 160       | 51          | −1.33  | 10,655  | 481       | 9500     | 13      | −55      | 10      | −4       | 12.1      |
| 22  | 55   | 11.05 | 0  | 240.0  | 50        | 170       | 50          | −1.42  | 11,270  | 506       | 10,100   | 29      | −48      | 42      | 2        | 16.7      |
| 23  | 85   | 11.89 | 0  | 12.2   | 50        | 190       | 62          | −1.49  | 10,307  | 518       | 9100     | 4       | −87      | 17      | −31      | 26.9      |
| 24  | 90   | 12.03 | 0  | 11.9   | 60        | 215       | 63          | −1.17  | 10,703  | 522       | 9525     | 21      | −69      | 35      | −60      | 28.5      |
| 25  | 100  | 12.31 | 0  | 32.4   | 40        | 170       | 57          | −1.31  | 12,307  | 569       | 10,875   | −22     | −66      | 47      | −40      | 40.2      |

### 2.4.1. Data Normalization

Cleaned data is often different and affects the result of machine learning. In order to eliminate this influence and improve the convergence speed to a certain extent, it is

necessary to normalize the data. In statistics, the more commonly used normalization methods include dispersion standardization and Z-score standardization. Dispersion standardization is widely used in deformation prediction, and its data normalization interval may be different but mainly normalized to  $[-1, 1]$  or  $[0, 1]$ . For any parameter  $x$ , the normalized value is given as:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} (\bar{X}_{max} - \bar{X}_{min}) + \bar{X}_{min} \quad (1)$$

In the formula,  $X_{max}$ ,  $X_{min}$  is the maximum and minimum values of variable  $x$ , where  $\bar{X}_{max}$ ,  $\bar{X}_{min}$  is the maximum and minimum values of normalized variables  $X$ . For the normalization process of measured data, we employed PyCaret, which uses the "Zscore" function by default to normalize the data in the range of  $[0, 1]$ .

#### 2.4.2. Cross-Validation Method

Building a machine learning model is mainly composed of 3 phases: training, testing, and validation. The validation process mainly solves the problems of overfitting and under fitting in machine learning. Machine learning validation methods primarily include simple cross-validation (hold-out cross-validation),  $k$ -fold cross-validation, and leave-one-out cross-validation [54]. In order to improve the generalization performance of the ML model and overcome the deficiencies of data,  $k$ -fold cross-validation is the most popular cross-validation method used in the model training phase [55,56]. In order to test the performance of the entire prediction model more accurately, the original training data set constructed is stochastically divided into  $k$  parts. For each calculation,  $k-1$  subsets are provided for training, and the remaining subset is used for verification. This procedure is used to test the ability of the sub-models. Repeat the calculation  $k$  times so that each sub-dataset can be used as a validation. Summarize and calculate the average ability of  $k$  sub-models to measure the performance of the entire prediction model. The formula is shown as follows:

$$T = \frac{1}{k} \sum_{i=1}^k MSE_i \quad (2)$$

where,  $T$  = fitness function,  $MSE_i$  = prediction error for the  $i$ th validation set. The performance of the  $k$ -fold cross-validation method depends on the number of subsets. However, fewer subsets cannot eliminate the problem of overfitting or underfitting, which will affect the model's accuracy. Too many subsets will significantly increase the model's performance computation time. Considering the limited amount of data in this study, in order to obtain reliable results, we finally adopted the 10-fold cross-validation method.

### 3. Feature Selection

Feature selection plays a significant role in machine learning because it manually or automatically chooses the input features that contribute significantly to the target variable. It is a desirable step to consider when building an ML model [34]. After the primary selection of input parameters in the surface deformation prediction of shield tunneling, the model may still face the problem of having too many input parameters. In order to avoid the dimensionality and the occurrence of overfitting and improve the model's accuracy, it is necessary to rely on feature selection for input parameter further filtering. Tan et al. [57] used grey relational analysis and sorted them by the degree of relevance to determine the main factor influencing the amount of deformation. Moreover, the commonly used feature selection methods include Filter, Wrapper, the principal component analysis method, Sobol sensitivity analysis [58], recursive feature elimination, the tree model-based feature selection method, etc.

#### 3.1. Analysis 1: Pearson Correlation Method

Feature selection methods are numerous and complex in predicting the surface deformation of shield tunnels. The linear correlation between the  $x$  and  $y$  variables can

be measured by the Pearson correlation coefficient, whose formula is given in Equation (3) [59].

$$r = \frac{\sum(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{(\sum(X_i - \bar{X})^2)(\sum(Y_i - \bar{Y})^2)}} \tag{3}$$

In the formula,  $r$  represents the Pearson correlation coefficient;  $X_i$  and  $Y_i$  represent the values of the  $X$  and  $Y$  variables in the sample respectively;  $\bar{X}$  and  $\bar{Y}$  are the average values of the variable values.

The closer the absolute value of the correlation coefficient  $r$  is to 1, the stronger the linear correlation between the variables. When  $r = 0$ , it means that there is no linear correlation between the two variables. The correlation coefficient was calculated using the `corr(.)` function provided in the Pandas library, and the results are listed in Figure 5. In this study, the guidelines recommended by Zhang et al. [6], were used to select the characteristic variables. From the calculation results of the correlation coefficient, it can be seen that among the listed features, only ST is strongly linearly correlated with Smax ( $|r| = 0.63$ ), while FpT, Th, To, JP, and VdF are moderately correlated with Smax ( $|r| = 0.42\sim 0.56$ ). In addition, the parameters H, AR, HDb, and VDb are weakly correlated with Smax ( $|r| = 0.23\sim 0.36$ ), and the remaining characteristic variables GW, FpC, PA, and HDf show very weak correlations with Smax ( $|r| < 0.19$ ). In this analysis, the feature variables with a medium correlation with the output variable Smax ( $|r| \geq 0.4$ ) are selected as effective features for predicting Smax, and the rest of the input features are not used as effective variables for prediction analysis due to small correlation coefficient and weak correlation. There is some difficulty in using the Pearson correlation coefficient feature selection method when a large number of analyzed features are poorly correlated with the predictor variables, as shown in this study.

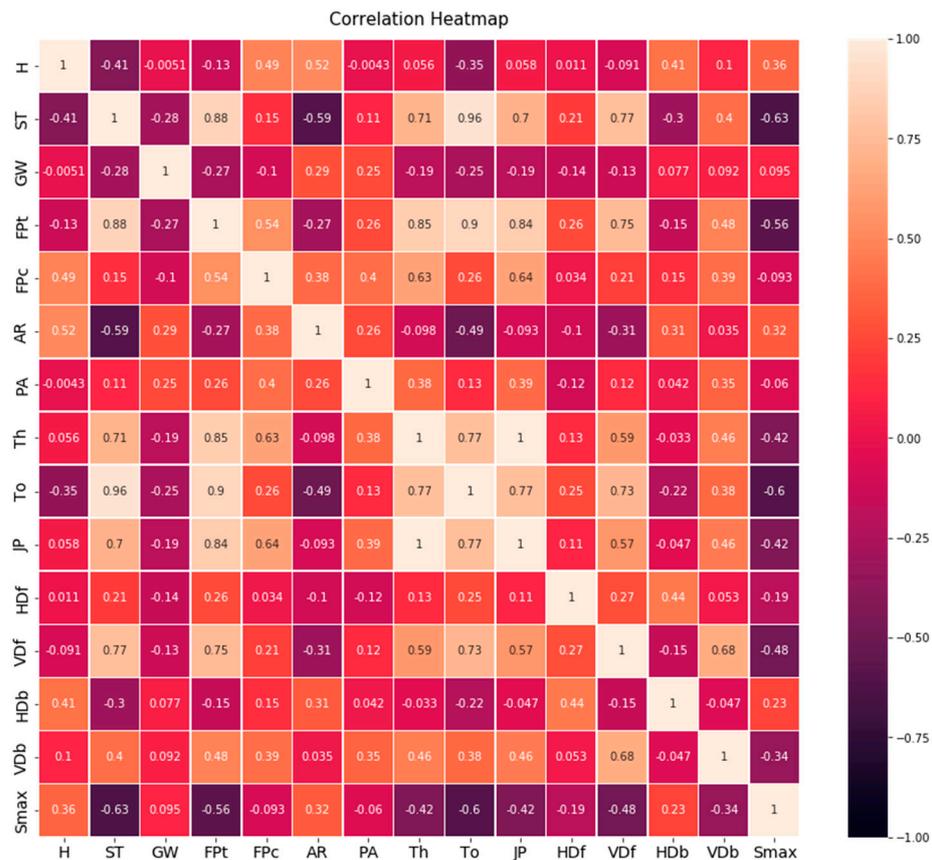


Figure 5. Inter-correlations of Pearson correlation coefficients among input data.

### 3.2. Analysis 2: Shapley Additive Explanations (SHAP)

Although the machine learning model based on the ensemble algorithm has relatively good performance, with the increase in model complexity, the interpretability of the model is reduced, which makes the regression model a black-box model. To solve the challenge of poor interpretability of the model, the SHAP framework is introduced to explain the model results and to provide support for the reliability of the model results. SHAP (Shapley additive explanations) is an interpretive framework proposed by Lundberg and Lee [39] for interpreting black-box models. The SHAP method is widely used in coalition game theory, which evaluates the degree of influence of input features on output parameters through Shapley values [60]. The basic method is to calculate the contribution value of each input feature and add the influence value of each feature to obtain the final prediction of the model [61].

For an ensemble tree model, when doing a regression task, the model outputs a probability value. SHAP can calculate the Shapley value to measure the influence value of each input variables to the final prediction. Assume that  $g$  represents the explanatory model,  $M$  represents the number of features, and  $z$  indicates whether the feature exists (value 0 or 1);  $\phi$  is the original value when all the inputs are absent, for each feature Shapley value, the formula can be given as follows:

$$g(z) = \phi_0 + \sum_i^M \varphi_i z_i \tag{4}$$

For each feature, the SHAP value describes the expected change in model predictions when conditioned on this feature. For each function, the SHAP value describes the feature’s contribution to the overall prediction outcome to account for the distinction between the average model calculation and the actual calculation. When  $i > 0$ ; it shows that this feature has an improving effect on the predicted value, and conversely when  $i < 0$ , it shows that this feature reduces the contribution. The model importance given by the regressor model only shows which input variable is essential but does not show how the variable influences the calculation results. The most significant superiority of the SHAP model is that it can show the influence of input variables in each data, as well as the positive or negative effect of this influence on the final prediction result.

Figure 6 is a summary graph of SHAP features, which analyzes factors affecting surface deformation according to feature importance.

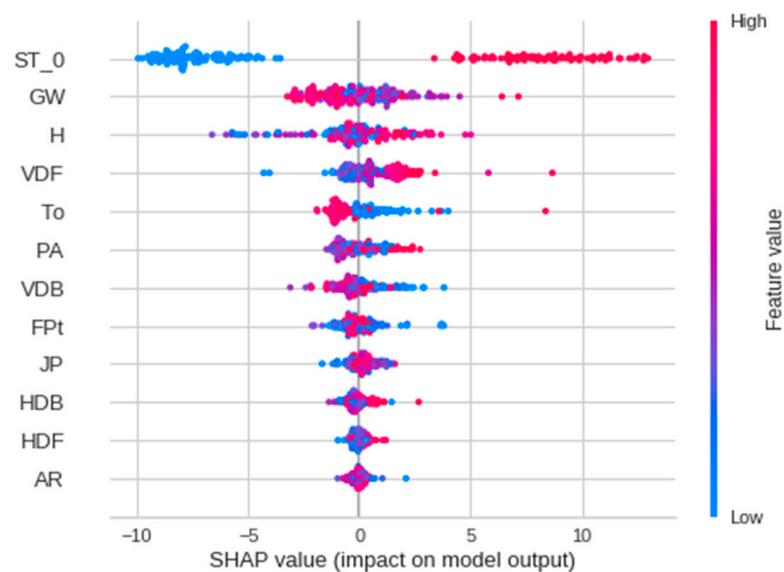


Figure 6. Summary plot obtained from SHAP analysis.

As shown in Figure 6 the soil type (ST), Torque (To) cover depth (H), ground water level (GW), and other characteristics have a significant effect on the model. The most important feature of SHAP that affects settlement prediction is the soil type (ST). Therefore, silty clay with a higher positive SHAP value has a greater influence on the model output result than silty sand. Torque (To) and Cover depth (H) in the current model (ET) also have a significant impact on predicting Smax. Positive SHAP values are observed when Torque values are low, while negative SHAP values are observed when Torque values are high, which means that a smaller torque will induce greater surface settlement. In the same way, it can be seen that when H is larger, the corresponding SHAP value is positive, which means that the output value of the prediction model will increase.

The SHAP values for PA, VDB, AR, JP, HDB, and HDF mainly converge near zero. The zero SHAP value stands that there is no effect on the model’s calculation. To better understand the dependency of each feature in the model’s output a simplified version of the above plot is shown in Figure 7. It can be found that in the current model, ST, To, H, GW, and VDF are the most important features in predicting Smax, while the importance of other features is less in comparison to ST.

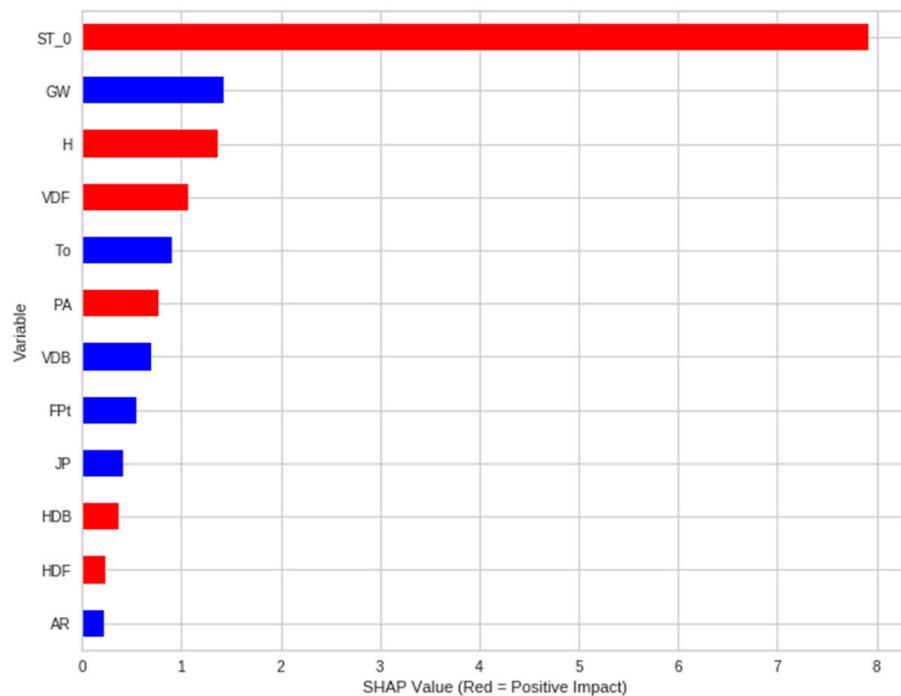


Figure 7. Feature Importance by SHAP Values of designed ET model.

Figure 8 shows the SHAP dependency graph between features ST, H, and GW, which have a high impact on the model and are selected to draw the SHAP feature dependency graph, where the third axis of the dependency graph is the categorical variable. Figure 8a, shows the correlation data of silty clay (labeled 0, represented by blue dots) and silty sand (labeled 1, represented by red dots). It can be found that the EPB operates at low VDF values while traversing the silty clay formation in Project 1, and calculates a large negative SHAP value. Conversely, when the TBM was operating at high VDF values while traversing the silty sand formation in Project 2, a large positive SHAP value was calculated.

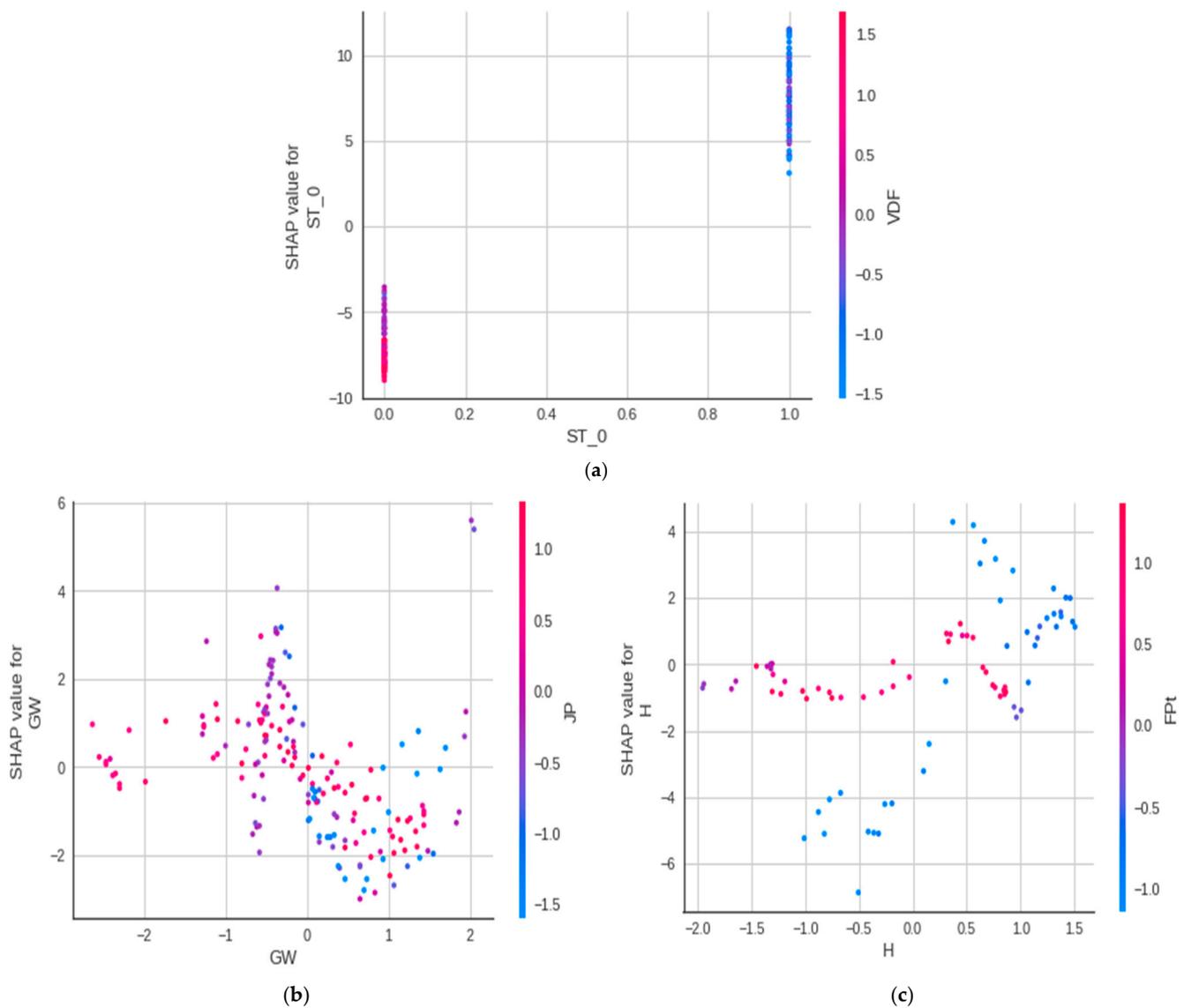


Figure 8. Dependency plots for (a) ST\_0, (b) GW, and (c) H.

Figure 8b shows how the SHAP value increases and then decreases as the GW value increases. Similarly, Figure 8c shows that SHAP values for H are primarily close to 0, corresponding to FPt, which means that the cover depth (H) has zero impact on the model’s output Smax when sufficient face pressure is present. Also, as the FPt values decrease at greater depths (>12 m), the Smax increases, indicating a larger positive SHAP value for H.

Therefore, as with analysis 1 and analysis 2, five variables (i.e. ST, To, H, GW, and VDF) were considered important for predicting tunnelling-induced settlements and are selected as final input parameters for building a ML models.

#### 4. Research Methodology

Despite the numerous research conducted in the past, it is essential to carry on with the ongoing efforts of developing newer and faster machine learning techniques that are more effective and can also be developed and deployed with ease. In this analysis, three commonly used statistical evaluation parameters, i.e., coefficient of determination ( $R^2$ ), mean absolute error (MAE), and root mean square error (RMSE), were used to evalu-

ate the accuracy of the calculation results generated by the intelligent method, as given by equations.

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i^{act} - y_i^{pred})^2}{\sum_{i=1}^N (y_i^{act} - \overline{y_i^{act}})^2} \quad (5)$$

$$MAE = \left(\frac{1}{n}\right) \sum_{i=1}^N |y_i^{act} - y_i^{pred}| \quad (6)$$

$$RMSE = \sqrt{\left(\frac{1}{N}\right) \sum_{i=1}^N (y_i^{act} - y_i^{pred})^2} \quad (7)$$

where  $y_i^{act}$  signifies the measured value of the  $i_{th}$  output feature,  $y_i^{pred}$  is the predicted value of the  $i_{th}$  output feature, and  $N$  is the number of data in the dataset. MAE, RMSE, and  $R^2$  represent the average value, standard deviation, and correlation degree of the difference between the measured value and the predicted value, respectively.

#### 4.1. Machine Learning Techniques

A new Python library (PyCaret) [40], offers a majority of machine learning techniques to construct a new prediction model. 21 ML algorithms were optimized through a comprehensive search of multiple ML methods, bypassing the whole dataset to the regression module of PyCaret (2.3.10), which divides the dataset into train and testing sets of 80% (211) and 20% (53) records, respectively, by calling the 'setup' function. 20% of the samples (53 data) are reserved from the original data set to demonstrate the predictive effect of the predict\_model() function. This process is independent of the train/test phase, since this particular split is done to simulate a real engineering environment. Another reason for this approach is that these 53 samples are not available when doing machine learning model building. In order to analyze and calculate the relationship between multiple input variables and output variable when using machine learning methods to build prediction models, regression analysis algorithms are often adopted [62,63]. Regression analysis statistics method determines the distribution relationship of data through known datasets, measures the contribution of input features to output features, and has been widely verified in ML methods [64,65]. Regression method can be used for making predictions on continuous data (time-series) in ML, especially when the regression relationship line of variables does not pass through the origin, regression analysis is more accurate. In addition, with the development of mathematical statistics theory, ML algorithm is often used in nonlinear regression estimation. Table 7 lists the regression estimators and other algorithms that were used in this study [40].

After performing the feature selection methods using analysis 1 and 2 as discussed in Section 3, all the models from the available machine learning libraries and frameworks were trained on datasets containing the selected features from Pearson correlation method and SHAP algorithm. Based on their  $R^2$  values, the top five models were selected for further optimization. The hyperparameter adjustment method is used to improve the  $R^2$  value of the selected model. Furthermore, tuned models were trained using 10-fold cross-validation to use all of the samples as training and testing, as the number of samples in the database is not enough. All of the tuned models were ensemble. Ensemble modeling is a technique in which various models are built to predict an output variable. This is accomplished through the use of various modeling methods or samples of training databases. The aggregated model then summarizes the predictions for each submodel, resulting in a single eventual prediction for the unknown data. The method of ensemble model can effectively reduce the generalization error of calculation, provided that the sub models built in the process of ensemble model are independent and diversified. The two most common methods in ensemble learning are bagging and boosting [66,67]. Stacking [68] is also a type of ensemble learning where predictions from multiple models are used as input features for a meta-model that predicts the final outcome. After the ensemble technique, the best of

all the models were calculated and selected using the AutoML function, improving the  $R^2$  value before determining the model for saving.

**Table 7.** Introduction to various ML algorithms (regression estimators).

| No. | Estimator                       | Description  |
|-----|---------------------------------|--|
| 1   | Extra tree Regressor            | A regressor with multiple decision trees, which is highly randomized, is only used in the ensemble methods.  |
| 2   | Random Forest Regressor         | The algorithm establishes multiple decision trees by randomly sampling, and obtains the overall regression prediction results by averaging the results of all trees.                     |
| 3   | Gradient Boosting Regressor     | An algorithm for combining multiple simple models into a composite model.  |
| 4   | Light Gradient Boosting Machine | The algorithm adopts a distributed gradient lifting framework based on decision tree algorithm, which can solve the problems encountered by GBDT in massive data.                        |
| 5   | AdaBoost Regressor              | This algorithm trains different weak regressors for the same training set and combines them to form a stronger final regressor.  |
| 6   | Extreme gradient boosting       | The algorithm is optimized on the framework of GBDT, which is efficient, flexible and portable.  |
| 7   | K neighbors Regressor           | A simple algorithm for predicting the target value on all available cases based on a similarity measure.   |
| 8   | Decision Tree Regressor         | A method of approximating the value of a discrete function. The induction algorithm is used to generate readable rules and decision trees, and the decision is used to analyze new data. |
| 9   | Support vector machine          | A generalized linear classifier for binary classification of data according to supervised learning.  |
| 10  | Bayesian Ridge                  | A probability model for estimating regression problems.  |
| 11  | Ridge Regression                | A biased estimation regression method dedicated to the analysis of collinearity data is essentially an improved least squares estimation method.   |
| 12  | CatBoost Regressor              | An algorithm based on symmetric decision tree, which can efficiently and reasonably handle categorical features.   |
| 13  | Linear Regression               | A linear approach that shows the relationship between a dependent variable and one or more independent variables.  |
| 14  | Least Angle Regression          | A statistical analysis method that uses regression analysis to determine the quantitative relationship between multiple variables.   |
| 15  | Huber Regressor                 | A linear regression that replaces the loss function of MSE with huber loss.  |
| 16  | Orthogonal Matching Pursuit     | A nonlinear adaptive algorithm using a super complete dictionary for signal decomposition.   |
| 17  | Elastic Net                     | A linear regression model applied to multiple correlated features.   |
| 18  | Lasso Regression                | A compressed estimate. It constructs a penalty function to obtain a more refined model, which is a biased estimate for processing data with complex collinearity.                        |
| 19  | Passive aggressive Regressor    | Online learning algorithms for both classification and regression.   |
| 20  | Random sample consensus         | An iterative method that estimates the parameters of a mathematical model from a set of observed data containing outliers that do not affect the estimates.                              |
| 21  | Theil-Sen regressor             | A robust model for fitting straight lines in nonparametric statistics.   |

## 5. Results and Discussion

The experimental work was performed by employing a Python library (PyCaret). The regression module of PyCaret is a supervised ML module that forecasts continuous values. It has over 21 ML algorithms and various plots to analyze the model's performance.

### 5.1. Experimental Design

Figure 9 depicts an experimental design flow diagram with seven major components: data collection, data pre-processing (data cleaning, normalization, and cross-validation), feature selection, hyper-parameter tuning, data partitioning, model development, model selection, and future prediction.

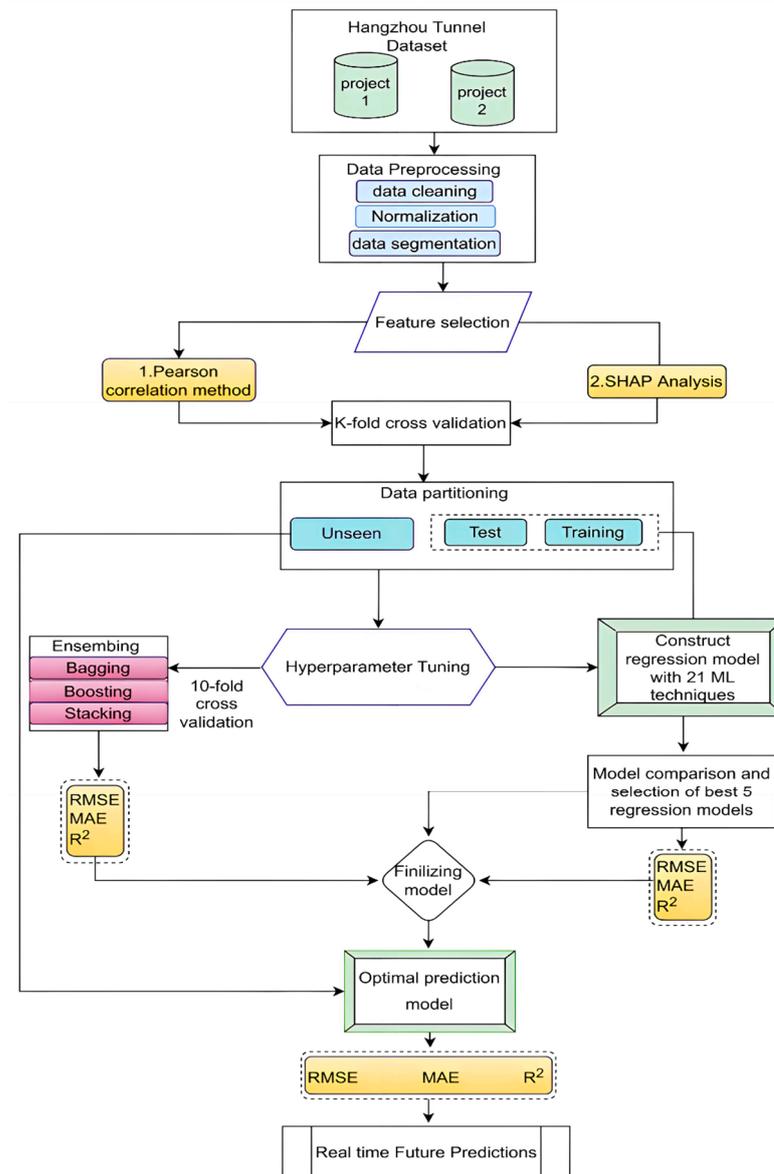


Figure 9. Flow chart of developed prediction model.

As described in Section 2.1, data from metro line tunneling projects in Hangzhou for predicting tunnel-induced settlements were collected. Data were cleaned first, which included the removal of nulls and outlier rejection. All the data are of integer datatype and were normalized to  $[-1, 1]$  as discussed in Section 2.4.2 and then divided into training and testing samples. As discussed in Section 3, feature selection methods using Pearson correlation and SHAP were applied to find relevant features. Five features (i.e. ST, To, H, GW, and VDF) were considered important for predicting Smax, and were selected as final input parameters for building ML models. ML models were then developed with 21 ML estimators, as explained in Section 4.1, and performance was recorded based on the MAE, RMSE, and  $R^2$  values the results that were obtained are presented in Table 8. Among the 21 developed ML models, the best five models were selected: the extra tree regressor, Random Forest Regressor, AdaBoost Regressor, Light Gradient Boosting Machine, and Gradient Boosting Regressor. All the best five selected models were then subjected to hyper-parameter tuning to maximize the model’s performance without overfitting by using the tune\_model function, which will automatically tune the hyper-parameters of a model using a random grid search on a pre-defined search space. Furthermore, the 10-fold cross-validation technique is utilized for a dynamic partitioning of data and for

improving the tuned model’s performance. The tuned models were then ensemble which is well known in improving the stability and accuracy of regression models (primarily tree-based) using various ensemble techniques; these include Bagging, Boosting, and Stacking. Table 8 presents the results obtained after adopting the corresponding techniques. Finally, forecasting was performed through the best-selected model (i.e., the extra trees regressor model). The model was also validated with unseen data for predictions to check the robustness of the model and it was found to be satisfactory.

**Table 8.** Statistical values of the 21 developed ML prediction models on Training and Test set.

| No. | Model                           | MAE      | R <sup>2</sup> | RMSE     | MAE   | R <sup>2</sup> | RMSE  |
|-----|---------------------------------|----------|----------------|----------|-------|----------------|-------|
|     |                                 | Training | Training       | Training | Test  | Test           | Test  |
| 1   | Extra tree Regressor            | 3.7      | 0.891          | 4.5      | 3.8   | 0.791          | 5.5   |
| 2   | Random Forest Regressor         | 4.2      | 0.857          | 5.0      | 4.3   | 0.753          | 6.1   |
| 3   | Gradient Boosting Regressor     | 4.3      | 0.846          | 5.1      | 3.8   | 0.788          | 5.6   |
| 4   | Light Gradient Boosting Machine | 4.5      | 0.826          | 5.5      | 3.97  | 0.762          | 6.0   |
| 5   | AdaBoost Regressor              | 4.4      | 0.834          | 5.2      | 5     | 0.736          | 6.4   |
| 6   | Extreme gradient boosting       | 4.3      | 0.845          | 5.2      | 5.1   | 0.742          | 6.41  |
| 7   | K neighbors Regressor           | 4.28     | 0.831          | 5.5      | 4.76  | 0.732          | 6.48  |
| 8   | Decision Tree Regressor         | 4.7      | 0.691          | 5.5      | 5.67  | 0.599          | 8.0   |
| 9   | Support vector machine          | 4.7      | 0.655          | 5.6      | 5.82  | 0.582          | 8.0   |
| 10  | Bayesian Ridge                  | 7.54     | 0.603          | 8.46     | 7.1   | 0.47           | 9.02  |
| 11  | Ridge Regression                | 7.59     | 0.602          | 8.48     | 6.80  | 0.51           | 8.74  |
| 12  | CatBoost Regressor              | 7.62     | 0.592          | 8.52     | 6.72  | 0.55           | 8.77  |
| 13  | Linear Regression               | 7.70     | 0.57           | 8.76     | 6.76  | 0.50           | 8.82  |
| 14  | Least Angle Regression          | 7.70     | 0.57           | 8.76     | 6.76  | 0.51           | 8.82  |
| 15  | Huber Regressor                 | 7.57     | 0.57           | 8.73     | 6.61  | 0.51           | 8.73  |
| 16  | Orthogonal Matching Pursuit     | 7.9      | 0.55           | 9.23     | 7.6   | 0.36           | 10.1  |
| 17  | Elastic Net                     | 8.1      | 0.52           | 9.31     | 7.62  | 0.40           | 9.6   |
| 18  | Lasso Regression                | 7.70     | 0.57           | 8.76     | 7.77  | 0.40           | 9.63  |
| 19  | Passive aggressive Regressor    | 8.1      | 0.42           | 10.44    | 8.56  | 0.19           | 11.20 |
| 20  | Random sample consensus         | 7.43     | -0.33          | 8.43     | 10.10 | -0.10          | 12.49 |
| 21  | Theil-Sen regressor             | 7.43     | -0.33          | 8.43     | 10.10 | -0.10          | 12.49 |

### 5.2. Performance Analysis

The model’s performance was analyzed across different aspects, as discussed below.

#### 5.2.1. Performance of Regression Models

The regression models for the given dataset were developed using PyCaret; for a diverse dataset, the coefficient of determination (R<sup>2</sup>), mean absolute error (MAE), and root mean square error (RMSE) are considered reliable statistics for evaluating the prediction model. Among the 21 different generated continuous models on the training set and the test set, the statistical significance of the best five selected models after being subjected to hyperparameter tuning, 10-fold cross-validation, and various ensemble techniques, giving their coefficient of determination (R<sup>2</sup>), the mean absolute error (MAE), and the root mean square error (RMSE), is given in Table 9 below. Based on the statistical values, it appears that the extra tree regressor (ET) outperformed in all cases at the training and testing stages, with an R<sup>2</sup> of 0.808, MAE of 3.7, and an RMSE of 5.2 on the test set. The extra tree regressor, which outperformed in all the cases, was finalized as the best model.

#### 5.2.2. Performance of the Extra Tree Regressor

The extra tree regressor model was also analyzed graphically using residual graphs, prediction error plots, and validation curve plots. Plotting uses the trained model object and generates a plot based on the testing dataset. Figure 10 depicts the plots between the experimental and predicted Smax as predicted by the generated models. The x-axis and y-axis represent the experimental and predicted values, respectively, and the blue and

green colors represent the training and testing sets, respectively. The black diagonal line represents the identity line.

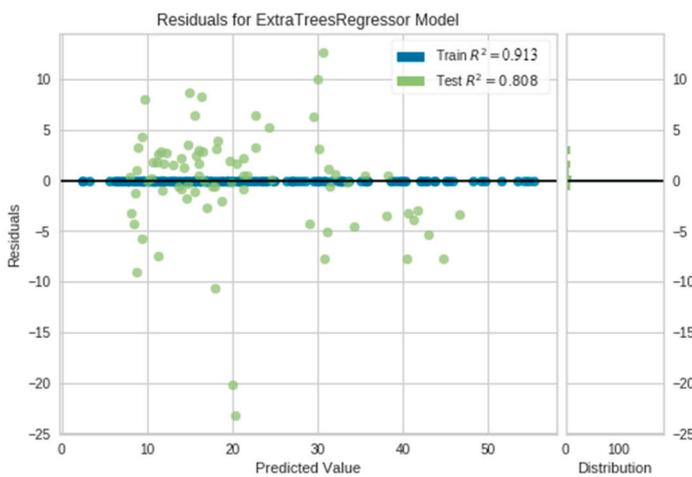
**Table 9.** Statistical values of the best five selected prediction Models on Training and Test set.

| No. | Model                           | MAE      | R <sup>2</sup> | RMSE     | MAE  | R <sup>2</sup> | RMSE |
|-----|---------------------------------|----------|----------------|----------|------|----------------|------|
|     |                                 | Training | Training       | Training | Test | Test           | Test |
| 1   | Extra tree Regressor            | 3.4      | 0.913          | 4.04     | 3.7  | 0.808          | 5.2  |
| 2   | Random Forest Regressor         | 4.2      | 0.861          | 5.0      | 4.3  | 0.786          | 5.4  |
| 3   | Gradient Boosting Regressor     | 4.3      | 0.854          | 5.1      | 3.8  | 0.792          | 5.5  |
| 4   | AdaBoost Regressor              | 4.4      | 0.849          | 5.1      | 5.0  | 0.763          | 5.9  |
| 5   | Light Gradient Boosting Machine | 4.5      | 0.842          | 5.5      | 3.9  | 0.778          | 6.0  |

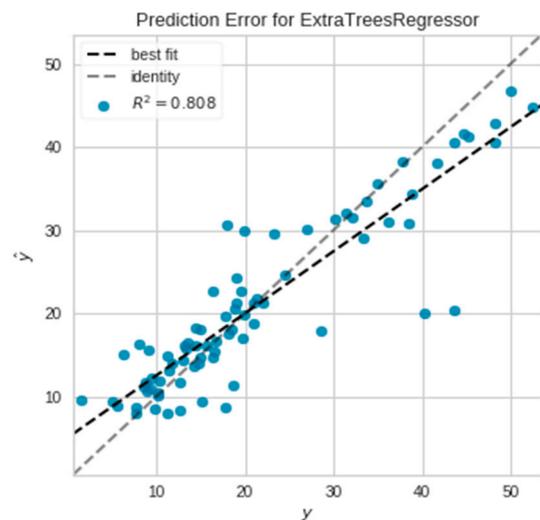
A prediction error plot compares actual targets to the values predicted by our model. This demonstrates the model’s variance. We can identify regression models using this plot by comparing them to the 45-degree slanting line and determining whether the prediction exactly matches the model.

A residual plot is a graphical representation of the relationship between an independent variable and its corresponding response variable. A residual value is a measure of how well a regression line fits the dataset, with a few data points fitting and others missing. The *x*-axis in the residual plot represents the residual values, and the *y*-axis represents the independent variable.

The validation curve is the learning curve calculated from a holdout validation dataset that gives an idea of how well the model is generalizing dataset. The validation curve plots the score over a varying hypermeter. It is more convenient to plot the influence of a single hypermeter on the training score and the validation score to determine whether the estimator is overfitting or underfitting for some hypermeter values. From Figure 10c, both the validation curves are becoming narrower with the increased value of max\_depth.



(a)



(b)

**Figure 10.** Cont.

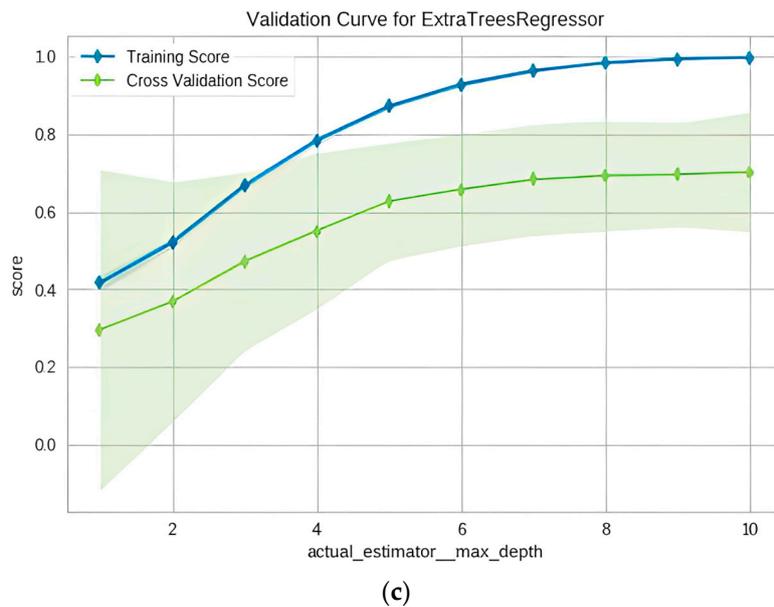


Figure 10. (a) Residuals for extra tree regressor; (b); Prediction error for extra tree regressor (c) Validation curve for extra tree regressor.

### 5.2.3. Prediction of Unseen Data

To finalize the model and predict based on unseen data (the 20% of data) that we detached at the start and never revealed to PyCaret. The `finalize_model()` function fits the model to the full dataset containing the test/holdout samples. The `predict_model()` function is employed to make predictions on the unseen data, this time we will pass the `data_unseen` parameter. `Data_unseen` is the variable created at the beginning and contains 20% (53 samples) of the original dataset that was never exposed to PyCaret. Although the model is same, we can see that  $R^2$  increased from 0.913 to 0.96 in the final ET model. This is because the final ET variable is trained on the entire dataset including the test/hold-out set. The plot of prediction error is shown in Figure 11. After testing the models on the unseen data subset, the results we obtained are summarized in Table 10 below. At the `unseen_data` stage, the model performed well with MAPE of 2.10,  $R^2$  of 0.961, and RMSE of 3.94.

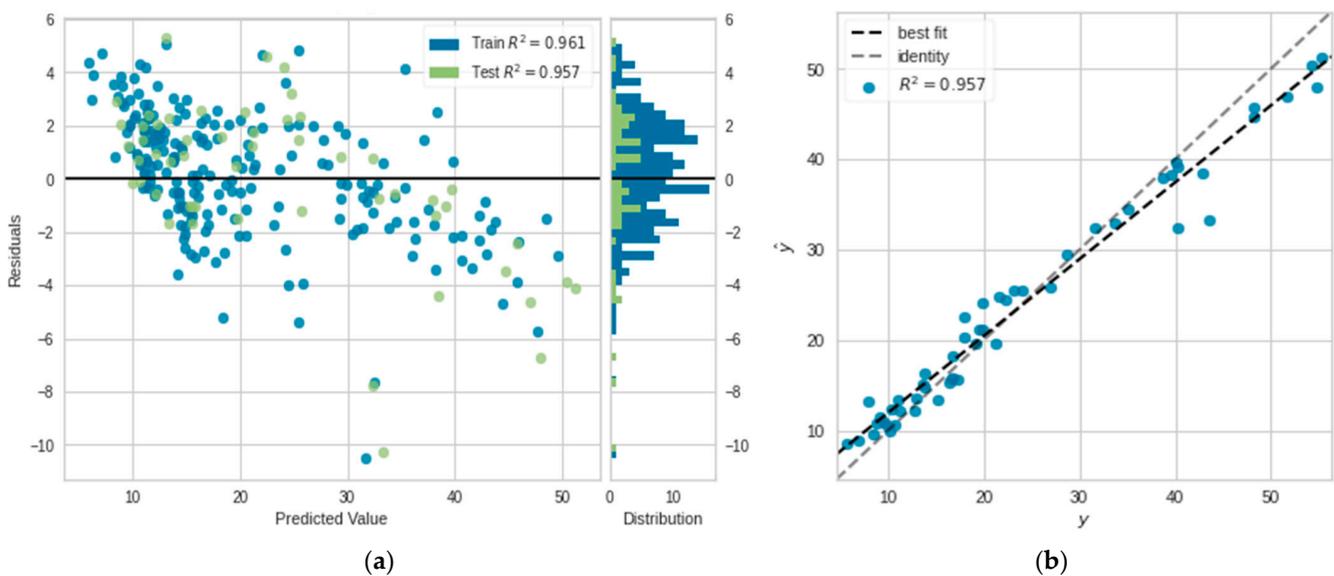


Figure 11. (a) Residuals for extra tree regressor; (b) Prediction error for extra tree regressor.

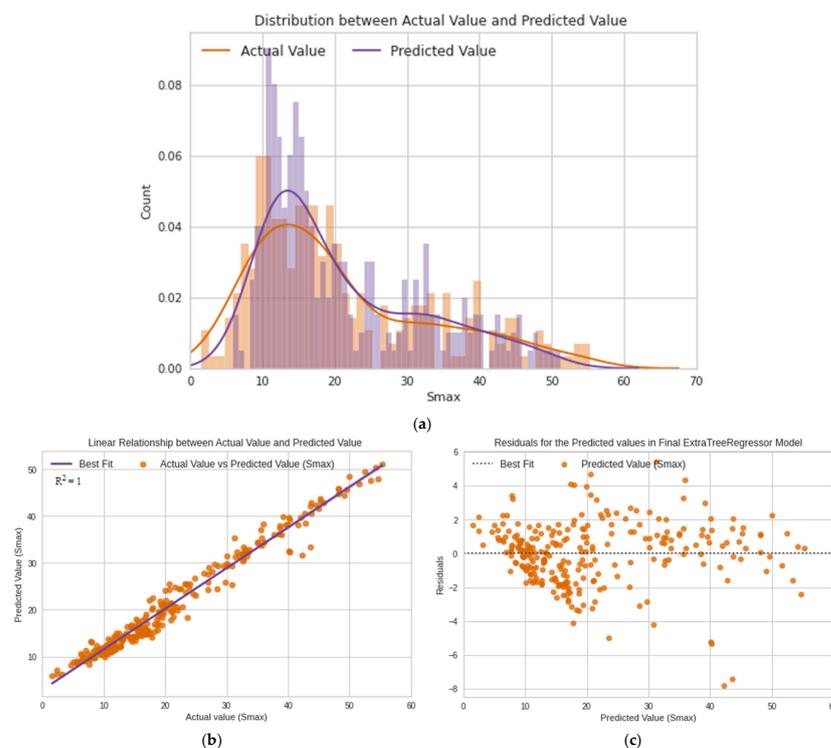
**Table 10.** Statistical values of the Generated prediction Models on unseen data.

|                      | MAE    | MSE     | RMSE   | R <sup>2</sup> | RMSLE  | MAPE   |
|----------------------|--------|---------|--------|----------------|--------|--------|
| Extra tree regressor | 2.1023 | 15.5794 | 3.9471 | 0.961          | 0.1664 | 0.1053 |

According to Table 10 and Figure 11, the mean absolute error (MAE) between predicted and measured maximum surface subsidence is less than 3%, indicating that the predictive performance of the model is acceptable and satisfactory for the given project. Given the statistical results and graphical plots, the models generated by PyCaret can be used to predict ground subsidence’s caused by shield tunneling.

5.3. Analysis of Model on Entire Dataset

As it is known that PyCaret wraps a number of machine learning frameworks and libraries, the model built by PyCaret is evaluated to learn about the details of the best algorithm selected by the AutoML function. The extra tree regressor is identified to be the best-selected model based on the statistical R<sup>2</sup> value of 0.961. Furthermore, our best model was finalized for deployment and saved for making new predictions over the whole dataset, including (training, test, and unseen\_data sets). An actual vs. predicted value plot is plotted for visualization as a histogram, as shown in Figure 12a, where the brown bars represent the actual values, the blue bars represent the predicted values, and the purple bars represent the error. A regression plot is plotted over the entire dataset to show the linear relationship between the Actual Value and the Predicted Value of Smax, and the dots are not far in the hyperplane of the linear line, which indicates that the regression model is good as shown in Figure 12b. Further, we can compare the predicted values and residuals in an error plot over the entire dataset, shown in Figure 12c. The statistical R<sup>2</sup> value of 1 and the actual vs. predicted value plot on our entire dataset indicate that the selected model i.e. extra tree regressor (ET) is highly significant in predicting the surface settlements induced by tunneling when compared to our other models.



**Figure 12.** (a) Distribution between Actual value and Predicted value; (b) Regression plot; (c) Residual plot.

## 6. Conclusions

This study systematically illustrates the process of application of Auto machine learning (AutoML)-based method to precisely predict tunneling-induced settlement using EPB shield machines. The 10-fold cross-validation method is utilized to overcome the scarcity of data and promote the robustness of the model. The coefficient of determination ( $R^2$ ), mean absolute error (MAE), and root mean square error (RMSE), are selected as three quantificational evaluation indices. Feature selection methods (i.e., Pearson correlation, and the SHAP framework) were employed to select features from a dataset with 14 input features (i.e., H, ST, GW, FPt, FPc, AR, PA, Th, To, JP, HDf, VDf, HDb, and VDb). Subsequently, AutoML-based models were built and trained on the selected features from the corresponding feature selection method. Then, the five best models were selected among 21 developed ML prediction models, and performances were compared by computing the  $R^2$ , RMSE, and MAE. According to the analysis, the extra tree regressor outperformed the other four models. Finally, the extra tree regressor model was used to make predictions on unseen data to simulate a real-life scenario and highlight the strengths of the model's predicted performance.

The following conclusions are provided based on the results of the model comparison and analysis:

- Feature selection is essential to address when predicting  $S_{max}$  due to shield tunneling. It is recommended to compare at least two feature selection methods, especially when there needs to be more information about the relationship between input and output parameters. Herein, H, ST, GW, FPt, PA, To, JP, VDF, and VDb significantly impact the maximum surface settlement caused by tunneling based on the features selected from the Pearson correlation method. However, deciding which feature to select may be challenging when there is a weak correlation with the desired output.
- SHAP-based feature selection algorithms comprehend the output of a complex ML model and facilitate model validation by allowing the user to investigate how various features contribute to the model's prediction. The SHAP analysis performed in this study revealed that the most critical parameters affecting tunneling-induced ground settlements were soil type (ST), torque (To), cover depth (H), groundwater level (GW), and tunneling deviation. These prudent factors identified by the model enable engineers and shield operators to reasonably manage shield operations.
- It is feasible and most reliable to calculate the maximum ground settlement ( $S_{max}$ ) during the construction of earth pressure balanced (EPB) shield tunneling by the proposed AutoML models. According to the statistical and graphical results, the extra-tree regressor's predictive ability is the best among all 21 AutoML models. Furthermore, the prediction results on unseen data indicate that the model's predicted performance is acceptable and within the project's tolerances. As a result, the prediction results generated from the AutoML-based extra tree regressor model are the most reliable, indicating that the model can be employed in real projects when completely-new deep excavation data are imported.

### *Limitations*

Because of the lack of a professional public database and the irregular quality of engineering data, this study excludes the meta-learning submodule in AutoML. More work should be done to collect similar data and create a database that can provide prior experience.

This study does not investigate the impact of tunneling operations with parameters related to grouting quality (e.g., large grout filling percentage and grouting pressure), which can significantly reduce settlements developed after the shield passing, as they were unavailable for Project 1. In order to enhance the effectiveness of the ML models, it is recommended to consider the effects of these parameters in future research.

**Author Contributions:** Conceptualization, S.M.H.; methodology, S.M.H.; software, S.M.H., L.M.; validation, S.M.H.; formal analysis, L.M.; investigation, S.M.H.; resources, L.M.; data curation, S.M.H.; writing—original draft preparation, S.M.H.; writing—review and editing, S.M.H.; visualization, L.M.; supervision, L.M.; project administration, L.M.; funding acquisition, L.M. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China under Grant No. 51738010, the Shanghai Natural Science Foundation (no. 22ZR1464600).

**Data Availability Statement:** The dataset used in this study is available on GitHub at <https://github.com/umgeotech/Database/tree/master/Surface%20Settlement> (accessed on: 6 September 2022). For any inquiries, please contact the corresponding author of the current manuscript.

**Acknowledgments:** The authors appreciate the five anonymous reviewers for the valuable suggestions that greatly improved this article. The dataset used in this research consists of 264 data samples adopted from the following study by Professor Wanhuan Zhou (State Key Laboratory of Internet of Things for Smart City and Department of Civil and Environmental Engineering, University of Macau, Macau, China) and her team, to whom we are grateful: Kannangara, K.P.M.; Zhou, W.H.; Ding, Z.; Hong, Z.H. Investigation of feature contribution to shield tunneling-induced settlement using Shapley additive explanations method. *J. Rock Mech. Geotech. Eng.* **2022**, *14*, 1052–1063.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Zhang, P. A novel feature selection method based on global sensitivity analysis with application in machine learning-based prediction model. *Appl. Soft Comput.* **2019**, *85*, 105859. [CrossRef]
- Chen, R.P.; Lin, X.T.; Kang, X.; Zhong, Z.Q.; Liu, Y.; Zhang, P.; Wu, H.N. Deformation and stress characteristics of existing twin tunnels induced by close-distance EPBS under-crossing. *Tunn. Undergr. Space Technol.* **2018**, *82*, 468–488. [CrossRef]
- Chen, D.F.; Feng, X.T.; Xu, D.P.; Jiang, Q.; Yang, C.X.; Yao, P.P. Use of an improved ANN model to predict collapse depth of thin and extremely thin layered rock strata during tunnelling. *Tunn. Undergr. Space Technol.* **2016**, *51*, 372–386. [CrossRef]
- Zhang, Z.; Huang, M. Geotechnical influence on existing subway tunnels induced by multilane tunneling in Shanghai soft soil. *Comput. Geotech.* **2014**, *56*, 121–132. [CrossRef]
- Jiang, M.; Yin, Z.Y. Influence of soil conditioning on ground deformation during longitudinal tunneling. *C. R. Mec.* **2014**, *342*, 189–197. [CrossRef]
- Zhang, W.G.; Li, H.R.; Wu, C.Z.; Li, Y.Q.; Liu, Z.Q.; Liu, H.L. Soft computing approach for prediction of surface settlement induced by earth pressure balance shield tunneling. *Undergr. Space* **2021**, *6*, 353–363. [CrossRef]
- Liang, R.Z.; Xia, T.D.; Lin, C.G.; Yu, F. Analysis of surface deformation and horizontal displacement of deep soil caused by shield driving. *J. Rock Mech. Eng.* **2015**, *34*, 583–593.
- Mair, R.J. Subsurface settlement profiles above tunnels in clays. *Geotechnique* **1993**, *43*, 315–320. [CrossRef]
- Standing, J.R. Greenfield ground response to EPBM tunnelling in London Clay. *Geotechnique* **2013**, *63*, 989–1007.
- Zhu, C.H.; Li, N. Estimation and regularity analysis of maximal surface settlement induced by subway construction. *Chin. J. Rock Mech. Eng.* **2017**, *1*, 3543–3560.
- Karakus, M.; Fowell, R.J. 2-D and 3-D finite element analyses for the settlement due to soft ground tunnelling. *Tunn. Undergr. Space Technol.* **2006**, *21*, 392. [CrossRef]
- Peck, R.B. Deep excavations and tunneling in soft ground. In Proceedings of the 7th ICSMFE, Mexico City, Mexico, 25–29 August 1969; pp. 225–290.
- Attewell, P.B.; Yeates, J.; Selby, A.R. Soil Movements Induced by Tunnelling and Their Effects on Pipelines and Structures. United States; 1986. Available online: <https://www.osti.gov/biblio/7052176> (accessed on 1 August 2022).
- Mindlin, R.D. Force at a point in the interior of a semi-infinite solid. *Physics* **1936**, *7*, 195–202. [CrossRef]
- Hagiwara, T.; Grant, R.J.; Calvello, M.; Taylor, R.N. The effect of overlying strata on the distribution of ground movements induced by tunnelling in clay. *Soils Found.* **1999**, *39*, 63–73. [CrossRef] [PubMed]
- Cheng, C.Y.; Dasari, G.R.; Chow, Y.K.; Leung, C.F. Finite element analysis of tunnel–soil–pile interaction using displacement controlled model. *Tunn. Undergr. Space Technol.* **2007**, *22*, 450–466. [CrossRef]
- Kasper, T.; Meschke, G. A 3D finite element simulation model for TBM tunnelling in soft ground. *Int. J. Numer. Anal. Methods Geomech.* **2004**, *28*, 1441–1460. [CrossRef]
- Ng, C.W.; Fong, K.Y.; Liu, H.L. The effects of existing horseshoe-shaped tunnel sizes on circular crossing tunnel interactions: Three-dimensional numerical analyses. *Tunn. Undergr. Space Technol.* **2018**, *77*, 68–79. [CrossRef]
- Jin, Y.F.; Zhu, B.Q.; Yin, Z.Y.; Zhang, D.M. Three-dimensional numerical analysis of the interaction of two crossing tunnels in soft clay. *Undergr. Space* **2019**, *4*, 310–327. [CrossRef]
- Qi, C.; Tang, X. Slope stability prediction using integrated metaheuristic and machine learning approaches: A comparative study. *Comput. Ind. Eng.* **2018**, *118*, 112–122. [CrossRef]

21. Zhang, L.; Shi, B.; Zhu, H.; Yu, X.B.; Han, H.; Fan, X. (PSO-SVM-based deep displacement prediction of Majiagou landslide considering the deformation hysteresis effect. *Landslides* **2021**, *18*, 179–193. [[CrossRef](#)]
22. Zhang, L.; Shi, B.; Zhu, H.; Yu, X.; Wei, G. A machine learning method for inclinometer lateral deflection calculation based on distributed strain sensing technology. *Bull. Eng. Geol. Environ.* **2020**, *79*, 3383–3401. [[CrossRef](#)]
23. Zhang, W.; Li, Y.; Wu, C.; Li, H.; Goh AT, C.; Liu, H. Prediction of lining response for twin tunnels constructed in anisotropic clay using machine learning techniques. *Undergr. Space* **2020**, *7*, 122–133. [[CrossRef](#)]
24. Shi, J.; Ortigao JA, R.; Bai, J. Modular neural networks for predicting settlements during tunneling. *J. Geotech. Geoenviron. Eng.* **1998**, *124*, 389–395. [[CrossRef](#)]
25. Suwansawat, S.; Einstein, H.H. Describing settlement troughs over twin tunnels using a superposition technique. *J. Geotech. Geoenviron. Eng.* **2007**, *133*, 445–468. [[CrossRef](#)]
26. Santos, O.J., Jr.; Celestino, T.B. Artificial neural networks analysis of Sao Paulo subway tunnel settlement data. *Tunn. Undergr. Space Technol.* **2008**, *23*, 481–491. [[CrossRef](#)]
27. Bouayad, D.; Emeriault, F. Modeling the relationship between ground surface settlements induced by shield tunneling and the operational and geological parameters based on the hybrid PCA/ANFIS method. *Tunn. Undergr. Space Technol.* **2017**, *68*, 142–152. [[CrossRef](#)]
28. Ahangari, K.; Moeinossadat, S.R.; Behnia, D. Estimation of tunnelling-induced settlement by modern intelligent methods. *Soils Found.* **2015**, *55*, 737–748. [[CrossRef](#)]
29. Goh, A.T.; Zhang, W.; Zhang, Y.; Xiao, Y.; Xiang, Y. Determination of earth pressure balance tunnel-related maximum surface settlement: A multivariate adaptive regression splines approach. *Bull. Eng. Geol. Environ.* **2018**, *77*, 489–500. [[CrossRef](#)]
30. Zhang, L.; Wu, X.; Ji, W.; AbouRizk, S.M. Intelligent approach to estimation of tunnel-induced ground settlement using wavelet packet and support vector machines. *J. Comput. Civ. Eng.* **2017**, *31*, 04016053. [[CrossRef](#)]
31. Suwansawat, S.; Einstein, H.H. Artificial neural networks for predicting the maximum surface settlement caused by EPB shield tunneling. *Tunn. Undergr. Space Technol.* **2006**, *21*, 133–150. [[CrossRef](#)]
32. Darabi, A.; Ahangari, K.; Noorzad, A.; Arab, A. Subsidence estimation utilizing various approaches—A case study: Tehran No. 3 subway line. *Tunn. Undergr. Space Technol.* **2012**, *31*, 117–127. [[CrossRef](#)]
33. Pourtaghi, A.; Lotfollahi-Yaghin, M.A. Wavenet ability assessment in comparison to ANN for predicting the maximum surface settlement caused by tunneling. *Tunn. Undergr. Space Technol.* **2012**, *28*, 257–271. [[CrossRef](#)]
34. Zhou, J.; Shi, X.; Du, K.; Qiu, X.; Li, X.; Mitri, H.S. Feasibility of Random-Forest approach for prediction of ground settlements induced by the construction of a shield-driven tunnel. *Int. J. Geomech.* **2016**, *17*, 04016129. [[CrossRef](#)]
35. Shao, C.; Lan, D. Optimal control of an earth pressure balance shield with tunnel face stability. *Autom. Constr.* **2014**, *46*, 22–29. [[CrossRef](#)]
36. Armaghani, D.J.; Mohamad, E.T.; Narayanasamy, M.S.; Narita, N.; Yagiz, S. Development of hybrid intelligent models for predicting TBM penetration rate in hard rock condition. *Tunn. Undergr. Space Technol.* **2017**, *63*, 29–32. [[CrossRef](#)]
37. Zhang, P.; Chen, R.P.; Wu, H.N. Real-time analysis and regulation of EPB shield steering using Random Forest. *Autom. Constr.* **2019**, *106*, 102860. [[CrossRef](#)]
38. Linardatos, P.; Papastefanopoulos, V.; Kotsiantis, S. Explainable AI: A review of machine learning interpretability methods. *Entropy* **2021**, *23*, 18. [[CrossRef](#)]
39. Lundberg, S.M.; Lee, S.I. A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems*. **2017**, Volume 30. Available online: <https://papers.nips.cc/paper/2017> (accessed on 1 August 2022).
40. Ali, M. (April 2022). PyCaret: An Open Source, Low-Code Machine Learning Library in Python. (PyCaret Version 2.3.5). Available online: <https://www.pycaret.org> (accessed on 1 August 2022).
41. Kannangara, K.P.M.; Zhou, W.H.; Ding, Z.; Hong, Z.H. Investigation of feature contribution to shield tunneling-induced settlement using Shapley additive explanations method. *J. Rock Mech. Geotech. Eng.* **2022**, *14*, 1052–1063.
42. GB/T50123-1999; Standard for Soil Test Method. China Planning Press: Beijing, China, 1999. (In Chinese)
43. Kim, C.Y.; Bae, G.J.; Hong, S.W.; Park, C.H.; Moon, H.K.; Shin, H.S. Neural network based prediction of ground surface settlements due to tunnelling. *Comput. Geotech.* **2001**, *28*, 517–547. [[CrossRef](#)]
44. Ding, L.; Wang, F.; Luo, H.; Yu, M.; Wu, X. Feedforward analysis for shield-ground system. *J. Comput. Civ. Eng.* **2013**, *27*, 231–242. [[CrossRef](#)]
45. Chen, R.; Meng, F.; Li, Z.; Ye, Y.; Ye, J. Investigation of response of metro tunnels due to adjacent large excavation and protective measures in soft soils. *Tunn. Undergr. Space Technol.* **2016**, *58*, 224–235. [[CrossRef](#)]
46. Feng, X.T.; Zhang, C.; Qiu, S.; Zhou, H.; Jiang, Q.; Li, S. Dynamic design method for deep hard rock tunnels and its application. *J. Rock Mech. Geotech. Eng.* **2016**, *8*, 443–461. [[CrossRef](#)]
47. Morovatdar, A.; Palassi, M.; Ashtiani, R.S. Effect of pipe characteristics in umbrella arch method on controlling tunneling-induced settlements in soft grounds. *J. Rock Mech. Geotech. Eng.* **2020**, *12*, 984–1000. [[CrossRef](#)]
48. Meng, F.Y.; Chen, R.P.; Kang, X. Effects of tunneling-induced soil disturbance on the post-construction settlement in structured soft soils. *Tunn. Undergr. Space Technol.* **2018**, *80*, 53–63. [[CrossRef](#)]
49. Dammyr, Ø.; Nilsen, B.; Gollegger, J. Feasibility of tunnel boring through weakness zones in deep Norwegian subsea tunnels. *Tunn. Undergr. Space Technol.* **2017**, *69*, 133–146. [[CrossRef](#)]

50. Qin, S.; Xu, T.; Zhou, W.H. Predicting pore-water pressure in front of a TBM using a deep learning approach. *Int. J. Geomech.* **2021**, *21*, 04021140. [[CrossRef](#)]
51. Kannangara, K.K.P.M.; Ding, Z.; Zhou, W.H. Surface settlements induced by twin tunneling in silty sand. *Undergr. Space* **2022**, *7*, 58–75. [[CrossRef](#)]
52. Zhou, C.; Xu, H.; Ding, L.; Wei, L.; Zhou, Y. Dynamic prediction for attitude and position in shield tunneling: A deep learning method. *Autom. Construct.* **2019**, *105*, 102840. [[CrossRef](#)]
53. Nawi, N.M.; Atomi, W.H.; Rehman, M.Z. The effect of data pre-processing on optimized training of artificial neural networks. *Procedia Technol.* **2013**, *11*, 32–39. [[CrossRef](#)]
54. Braga-Neto, U.; Hashimoto, R.; Dougherty, E.R.; Nguyen, D.V.; Carroll, R.J. Is cross-validation better than resubstitution for ranking genes. *Bioinformatics* **2004**, *20*, 253–258. [[CrossRef](#)]
55. Zhang, P.W. Hybrid meta-heuristic and machine learning algorithms for tunneling-induced settlement prediction: A comparative study. *Tunn. Undergr. Space Technol.* **2020**, *99*, 103383. [[CrossRef](#)]
56. Zhang, P.Y. Intelligent modelling of clay compressibility using hybrid meta-heuristic and machine learning algorithms. *Geosci. Front.* **2021**, *12*, 441–452. [[CrossRef](#)]
57. Tan, C.P. Surface subsidence prediction based on grey relational support vector machine. *J. Cent. South Univ. (Nat. Sci. Ed.)* **2012**, *43*, 632–637.
58. Cheng, Z.L.; Zhou, W.H.; Ding, Z.; Guo, Y.X. Estimation of spatiotemporal response of rooted soil using a machine learning approach. *J. Zhejiang Univ. Sci. A* **2020**, *21*, 462–477. [[CrossRef](#)]
59. Pearson, K. Notes on Regression and Inheritance in the Case of Two Parents. *Proc. R. Soc. Lond.* **1895**, *58*, 240–242. [[CrossRef](#)]
60. Štrumbelj, E.; Kononenko, I. Explaining prediction models and individual predictions with feature contributions. *Knowl. Inf. Syst.* **2014**, *41*, 647–665. [[CrossRef](#)]
61. Parsa, A.B.; Movahedi, A.; Taghipour, H.; Derrible, S.; Mohammadian, A.K. Toward safer highways, application of XGBoost and SHAP for real-time accident detection and feature analysis. *Accid. Anal. Prev.* **2020**, *136*, 105405. [[CrossRef](#)]
62. Siew, H.L.; Nordin, M.J. Regression techniques for the prediction of stock price trend. In Proceedings of the International Conference on Statistics in Science, Langkawi, Malaysia, 10–12 September 2012.
63. Jiang, P.; Chen, J. Displacement prediction of landslide based on generalized regression neural networks with K-fold cross-validation. *Neurocomputing* **2016**, *198*, 40–47. [[CrossRef](#)]
64. Handa, R. Prediction of Foreign Exchange Rate Using Regression Techniques. 2017. Available online: <https://www.semanticscholar.org/paper/PREDICTION-OF-FOREIGN-EXCHANGE-RATE-USING-Sharma/f3feac47eafb58a1c200082c895cd591b09e020a> (accessed on 1 August 2022).
65. Han, J.; Pei, J.; Kamber, M. *Data Mining: Concepts and Techniques*; Elsevier: Amsterdam, The Netherlands, 2011.
66. Breiman, L. Bagging predictors. *Mach. Learn.* **1996**, *24*, 123–140. [[CrossRef](#)]
67. Yoav Freund, R.E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139. [[CrossRef](#)]
68. Wang, G.; Hao, J.; Ma, J.; Jiang, H. A comparative assessment of ensemble learning for credit scoring. *Expert Syst. Appl.* **2011**, *38*, 223–230. [[CrossRef](#)]