Reliability Analysis of Rock Slope Using Soft Computing Techniques

Prithvendra Singh 1)*, Deepak Kumar 2) and Pijush Samui 3)

1) National Institute of Technology Patna, Bihar, India. * Corresponding Author.
   E-Mail: prithvendra.ce17@nitp.ac.in
2) National Institute of Technology Patna, Bihar, India.
   E-Mail: decage007@gmail.com
3) Associate Professor, Department of Civil Engineering, National Institute of Technology Patna,
   Bihar, India. E-Mail: pijush@nitp.ac.in

ABSTRACT
Probability of safety (reliability) analysis is a major concern of any structure, especially of rock mechanics. This paper used different machine learning (ML) techniques (cubist model, extreme learning machine (ELM) and multivariate adaptive regression splines (MARS)) for reliability analysis of rock slopes. The performance of these ML models was assessed using different statistical parameters, such as Nash-Sutcliff coefficient (NS), coefficient of determination ($R^2$), root mean square error (RMSE), variance account factor (VAF), expanded uncertainty (U95), mean absolute error (MAE), … etc. A comparative study was performed to test the adaptability of cubist, ELM and MARS models. It is evident from the results that MARS model shows excellent results in terms of fitness parameters. This study reflects that cubist, ELM and MARS models are well capable of predicting the reliability of slope in terms of the factor of safety (FOS) of rock slope considering statistical predictands.

KEYWORDS: Reliability analysis, Rock slope, Cubist model, ELM, MARS.

INTRODUCTION
Human intervention in the form of construction is increasing significantly over the elapse of time. Random change in climate (i.e., continuous increase in temperature) favours the occurrence of natural disasters in the form of landslide (Liu and Chen, 2007; Zhou et al., 2010). Soil slope failure cannot be determined using slip surface; whereas rock slopes fail generally by slipping along the surface which is structurally weak (Khalokakaie and Zare Naghadehi, 2012; Pariseau, Puri and Schmelter, 2008; Youssef, Pradhan and Al-Harthi, 2015).

Rock slope failure is primarily controlled by weak surfaces, which depends upon two factors: the enhancement of hydrostatic pressure as a result of seepage flow and the reduction in shear strength of the weak surface. These two factors disturb the force balance of the rock slope (Park, West and Woo, 2005; Tan et al., 2013; Topal, 2007). In the case of disturbance and uncontrolled situation, rock slope failure occurs. Further study on failure of rocks has been carried out (Li et al., 2015). In past decades, study of rock slopes (reliability analysis) mainly concentrated on homogeneous slopes. Many assessment methods for homogeneous slopes were proposed (Jimenez-Rodriguez and Sitar, 2007; Wang et al., 2013; Wei et al., 2014). Most of these methods concentrated on Rosenbleuth method, Monte Carlo method and first-order second- moment (FOSM) method (Duzgun, Yucemen and Karpuz, 2003; Ganji and Jowkarshorijeh, 2012; Ge, Tu and Qin, 2011).
Rosenbleuth method, yielding of stability of slope using reliability index has been done using specific samples at points with stipulated rules (Park, Um et al., 2012; Sun, Zeng and Ding, 2008). Few researchers used Gauss process-based approach for landslide displacement and analysis (Liu, Xu and Shao 2012). While using FOSM method in the calculation of reliability index, partial derivative of the performance function is required (Liang and Xue-Song, 2012).

The relationship between hydrostatics and shear strength plays a vital role in slope reliability (Kourosh, Mosrafa and Heydari, 2011; Li, Zhao and Ru, 2013). There are various deterministic methods, like finite element method, limit equilibrium method, finite difference method and discrete element method, which are available to study rock slope stability (Reale et al., 2015). The above methods of rock slope characterization can only be used to determine safety factor, deformation distribution characteristics and stress (Li, Wang and Cao, 2014; Park, Um et al., 2012; Park et al., 2005). These deterministic methods have limitations, as these methods do not account for the degree of uncertainty during the rock slope stability characterization process (Yang, Zhi-gang, Tong-chun Li and Miao-lin Dai, 2009) and the calculated factor of safety is not suitable for the characterization of stability of the slope as index (Jiang et al., 2014; Li, Tang and Phoon, 2015). Considering the above limitations, several researchers have shifted their focus towards the probabilistic theory to analyze the slope stability problem and calculate the safety factor for rock slope (Gravanis, Pantelidis and Griffiths, 2014; Park et al., 2016; Pathak and Nilsen, 2004). Reviewing recent literature on artificial intelligence (AI) and its applications in various fields of civil engineering motivated the authors to adopt different machine learning techniques (Baecher and Christian, 2003; Duzgun et al., 2003; Ganji and Jawkarshorijeh, 2012; Ge et al., 2011; Harr, 1987; Liang and Xue-song, 2012). In this research, a relatively new machine learning technique called cubist model has been used to analyze rock stability. In addition, a comparative study has been carried out using two other machine learning models (ELM and MARS).

Safety factor \( F_s \) of slopes having plane failure in rocks has been determined using equations of force balance, using Mohr- Coulomb criteria (Zhou et al., 2017) as in Equation (1).

\[
F_s = \frac{cA_2 + (W \cos \beta + A_1 \sigma \alpha \tan \varphi)}{W \sin \beta - A_1 \sigma \alpha \sin \alpha}
\] (1)

where \( W, \alpha, \beta, \varphi \) and \( c \) are the mass of sliding block, intersection angle of slip surface \( B \) with horizontal plane, intersection angle of slip surfaces \( B \) and \( A \), internal friction angle and cohesion of slipping surface, respectively. \( A_1 \) is the area of rear edge slip surface \( A \) and \( A_2 \) is the area of bottom slip surface \( B \). The total mass weight has been obtained through geometric analyses and mechanical calculation as 17,420 kN/m, \( A_1 \) is 35 m, \( A_2 \) is 50 m, \( \alpha \) is 40° and \( \beta \) is 32° as from the work of Zhou et al. (2017).

After calculating the factor of safety, performance function (PF) has been obtained using Equation (2).

\[
PF = F_s - 1
\] (2)

Performance function (PF) follows normal distribution and the values of PF and \( \beta \) (reliability index) can be calculated using Equation (3).

\[
\beta = \frac{\mu(PF)}{\sigma(PF)}
\] (3)

where \( \mu(PF) \) is the mean and \( \sigma(PF) \) is the standard deviation of the performance function PF.

THEORETICAL BACKGROUND OF MODEL USED

**CUBIST**

Cubist, generally a rule-based model, is an extension of Quinlan’s M5 model tree (Quinlan, 2014). Cubist uses the method of recursive partitioning. It splits
training cases just like growing a decision tree. It is based on the minimization of intra subset alteration of class value in spite of maximization. Similar to regression trees (Breiman, 2017), it also uses the same divide and conquer rule.

Suppose that we have to construct a model tree for set M of training cases. Now set M will be split according to the outcome of the test. Suppose that M_i represents the subset of i outcomes and sd (M_i) is the standard deviation; then, the expected reduction in the value of error can be obtained by using Equation (4).

$$\Delta \text{error} = sd(M) - \sum \frac{|M_i|}{|M|} \times sd(M_i)$$  \hspace{1cm} (4)

As the test proceeds, the model picks the outcome having maximum reduction in the value of error. Cubist model also uses committees, where iterative model trees are created in sequence, which have boosting-like schemes. As the initial tree grows, alteration in the final cubist model occurs by estimation of error through the processes of linear modelling, linear model simplification, pruning and smoothing. Interested readers can find more details on CUBIST in Kuhn et al. (2016).

**Extreme Learning Machine (ELM)**

Huang (2003) proposed extreme learning machine (ELM) for extensive single hidden layer feedforward neural networks (SLFNs). ELM has gathered more focus than any other machine learning model since the development of machine learning techniques, because it performs generalization at an astonishingly fast speed, needless of mathematically predetermined internal knowledge.

ELM uses datasets in two forms; i.e., training datasets and testing datasets. First of all, ELM model is developed using a training dataset and then verified using a testing dataset. The training and testing datasets have been normalized before using ELM. An enormously fast cognizing algorithm for SLFNs having N’ cryptic neurons can be proposed using ELM, where N’ ≤ N, considering that N is the number of training samples.

**SLFN Approximation Problem**

For N distinct random samples \((x_i, t_i)\), where \(x_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^T \in \mathbb{R}^n\) and \(t_i = [t_{i1}, t_{i2}, \ldots, t_{im}]^T \in \mathbb{R}^m\), standard SLFNs with N’ hidden neurons and activation function \(g(x)\) are mathematically modelled as in Equation (5).

$$\sum_{i=1}^{N'} \beta_{ij} g(w_i \cdot x_j + b_j) = O_j, \ j = 1, 2, \ldots, N$$  \hspace{1cm} (5)

where \(w_i = [w_{i1}, w_{i2}, \ldots, w_{im}]^T\) is the weight vector which connects the \(i^{th}\) hidden neuron and the input neurons, while the weight vector which connects the \(i^{th}\) hidden neuron and the output neurons is \(\beta_i = [\beta_{i1}, \beta_{i2}, \ldots, \beta_{im}]^T\). \(w_i \cdot x_j\) denotes the inner product of \(w_i\) and \(x_j\).

These N samples can be approximated using standard SLFNs with N’ hidden neurons with the activated function \(g(x)\),

$$\sum_{j=1}^{m} \| O_j - t_j \| = 0$$  \hspace{1cm} (6)

i.e., there exist \(\beta_i\), \(w_i\) and \(b_i\), such that:

$$\sum_{i=1}^{N'} \beta_{ij} g(w_i \cdot x_j + b_j) = O_j, \ j = 1, 2, \ldots, N$$  \hspace{1cm} (7)

The above equations can compactly be written as:

$$H\beta = T$$  \hspace{1cm} (8)
where,
\[
H(\ldots, w_N, b_1, \ldots, b_N, x_1, \ldots, x_N) = \begin{bmatrix}
g(w_1 \cdot x_i + b_1) & \ldots & g(w_N \cdot x_i + b_N) \\
\vdots & \ddots & \vdots \\
g(w_1 \cdot x_N + b_1) & \ldots & g(w_N \cdot x_N + b_N)
\end{bmatrix}
\]

(9)

\[
\beta = \begin{bmatrix}
\beta_1^T \\
\vdots \\
\beta_N^T
\end{bmatrix}
\quad T = \begin{bmatrix}
t_1^T \\
\vdots \\
t_N^T
\end{bmatrix}
\quad \text{and} \quad t_i^T_{N\times m}
\]

(10)

The \(i\)th hidden neuron’s output vector is the \(i\)th column of \(H\) with respect to inputs. \(x_1, x_2, \ldots, x_N\).
Interested readers can find more details in Shah et al. (2014).

**MARS**

MARS, developed by Friedman, is used for developing relations among a group of predictors and subjected variables. MARS uses the strategy of divide and conquer, partitioning training data groups in distinct domains, where every domain gets its own regression string. MARS is dependent on collection and analysis of data procurement.

Generally, if \(B\) is the basis function and \(P\) stands for the predictor variables, then the predictor of MARS model can be represented using Equation (11).

\[
y = \beta_0 + \sum_{j=1}^{p} \sum_{b=1}^{B} [\beta_{jb}(+)Max(0, x_j - H_{bj}) + \beta_{jb}(-)Max(0, H_{bj} - x_j)]
\]

(11)

If \(\beta\) coefficients of univariate basis functions \(Max(0, x - H)\) and \(Max(0, H - x)\) are 0, they need not to be present.

In MARS algorithm, first of all, from the initial constant basis function, a leading stepwise quest takes place for the basis function. Splitting minimizes “lack of fit” criteria chosen from all possible splits. The process repeats as long as the model clauses the values of certain maximum numbers of pre-decided basis functions (LeBlanc and Tibshirani, 1994). Interested readers can find more information in Yuvaraj et al. (2013).

**METHODOLOGY**

**Dataset and Model Development**

In this paper, the generation of data has been carried out according to the database of Zhou et al. (2017). The data contains information on coefficient of cohesion \((c; \text{kPa})\), angle of internal friction \((\phi; ^\circ)\) and residual tensile strength \((\sigma_t; \text{kPa})\). A total number of 80 geotechnical datasets have been used in this study to predict the reliability of rock slope. Each of the datasets has been divided into two categories while developing the model. The division of data depends on the problem and on the total number of available datasets. In this study, 70% (i.e., 56) datasets have been adopted for training the model and the remaining 30% (i.e., 24) for testing the model. Prior to using the dataset, it has been normalized using Equation (12).

\[
X_{\text{normalized}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}}
\]

(12)

where \(X_{\text{normalized}}\) is the normalized value, \(X_{\text{min}}\) is the minimum value from all the events, \(X_{\text{max}}\) is the maximum value from all the events and \(X\) is the particular value of that parameter.

Cubist model has been developed using R script, while ELM and MARS have been developed using MATLAB. Models drawn have a structure of input matrix \((x)\) defined by \(x = \text{coefficient of cohesion, angle of internal friction and residual tensile strength, while the factor of safety is considered as the target variable (y). All the machine models have used trial and error basis to optimize their prediction.}

A graph representing 80 datasets (each of \(\sigma_t\) (kPa), \(c\) (kPa), \(\phi\) and \(F_s\)) has been drawn and is shown in Figure 1.
**Performance Measures**

Regression calculates the strength of relationship between different predictors and predictand variables. The fitness and adequacy of the model are justified using various statistical approaches, e.g., weighted mean absolute percentage error (WMAPE); Nash-Sutcliffe efficiency (NS) (Nash and Sutcliffe, 1970); root mean square error (RMSE); variance account factor (VAF); coefficient of determination ($R^2$); adjusted determination coefficient (Adj. $R^2$); performance index (PI); bias factor (Armstrong and Collopy, 1992; Wang et al., 2013); root mean square error to observation’s standard deviation ratio (RSR); normalized mean bias error (NMBE); mean absolute percentage error (MAPE); mean absolute error (MAE); mean bias error (MBE); sample testing of observed vs. predicted value was performed by Legates & McCabe’s Index (LMI) (Legates and McCabe, 2013; Legates and McCabe, 1999); relative percent difference (RPD); Willmott’s index of agreement (WI) (Willmott, 1984); expanded uncertainty (U95); t-statistic (Stone 1993). While importing data, observed values are mentioned as $d_i$, whereas predicted values are mentioned as $y_i$. SD is the standard deviation, N is the number of datasets, p is the number of variables, n is the number of event inputs and $d_{mean}$ is the average of observed values.

1. **Weighted mean absolute percentage error (WMAPE)** can be calculated using Equation (13).

\[
WMAPE = \frac{\sum_{i=1}^{n} \frac{|d_i - y_i|}{d_i} \times d_i}{\sum_{i=1}^{n} d_i}
\]  

(13)

2. **Nash-Sutcliffe efficiency (NS)** is the ratio of residual error variance to the measured variance in observed data (Nash and Sutcliffe, 1970) and can be represented using Equation (14).

\[
NS = 1 - \frac{\sum_{i=1}^{n} (d_i - y_i)^2}{\sum_{i=1}^{n} (d_i - d_{mean})^2}
\]  

(14)
3. Root mean square error (RMSE) can be calculated as shown in Equation (15).

\[ \text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{n} (d_i - y_i)^2} \] (15)

4. Variance account factor can be obtained using Equation (16).

\[ \text{VAF} = \left(1 - \frac{\text{var}(d_i - y_i)}{\text{var}(d_i)}\right) \times 100 \] (16)

5. Coefficient of determination \((R^2)\) value can be obtained using Equation (17).

\[ R^2 = \frac{\sum_{i=1}^{n} (d_i - d_{mean})^2 - \sum_{i=1}^{n} (d_i - y_i)^2}{\sum_{i=1}^{n} (d_i - d_{mean})^2} \] (17)

6. Adjusted determination coefficient \((\text{Adj. } R^2)\) can be calculated using Equation (18).

\[ \text{Adj } R^2 = 1 - \frac{(n-1)}{(n-p-1)}(1 - R^2) \] (18)

7. The values of performance index (PI) and the bias factor (Kung et al., 2007; Prasomphan and Machine, 2013) are calculated using Equations (19) and (20).

\[ \text{PI} = \text{adj } R^2 + 0.01 \times \text{VAF} - \text{RMSE} \] (19)

\[ \text{Bias Factor} = \frac{1}{N} \sum_{i=1}^{n} \frac{y_i}{d_i} \] (20)

8. RSR includes the benefits of both error index statistics and normalization factor (Moriasi et al., 2007). The most favorable value of RSR is zero. RSR can be calculated using Equation (21).

\[ \text{RSR} = \frac{\text{RMSE}}{\sqrt{\frac{1}{N} \sum_{i=1}^{n} (d_i - d_{mean})^2}} \] (21)

9. NMBE calculates the model strength to forecast values away from mean value (Srinivasulu and Jain, 2006). A positive NMBE represents an over-predicted and a negative NMBE represents an under-predicted model. NMBE is given by Equation (22).

\[ \text{NMBE} = \frac{1}{N} \sum_{i=1}^{n} \left(\frac{y_i - d_i}{d_i}\right) \times 100 \] (22)

10. Mean absolute percentage error (MAPE) relates residual error of each event with the observed value (Armstrong and Collopy, 1992). It is a dimensionless quantity and a lower value of MAPE represents superior performance of the model and vice versa. MAPE is obtained using Equation (23).

\[ \text{MAPE} = \frac{1}{N} \sum_{i=1}^{n} \left|\frac{d_i - y_i}{d_i}\right| \times 100 \] (23)

11. Relative percent difference (RPD), suggested by Viscarra Rossel, is used to evaluate the efficiency of the model (Viscarra Rossel, McGlynn and McBratney, 2006). RPD value can be obtained using Equation (24).

\[ \text{RPD} = \frac{\text{SD}}{\text{RMSE}} \] (24)

12. Willmott’s Index of agreement \((\text{WI})\) value ranges between 0 and 1 (Willmott, 1981, 1982, 1984). The value of 0 represents no match at all, while the value of 1 indicates a perfect match. WI can be calculated using Equation (25).
\[ WI = 1 - \left[ \frac{\sum_{i=1}^{N} (d_i - y_i)^2}{\sum_{i=1}^{N} |y_i - d_{mean}| + |d_i - d_{mean}|^2} \right], \quad (25) \]

\[ 0 < WI \leq 1 \]

13. Mean absolute error (MAE) can be obtained using Equation (26).

\[ MAE = \frac{1}{N} \sum_{i=1}^{N} |y_i - d_i| \quad (26) \]

14. Mean bias error (MBE) can be obtained using Equation (27).

\[ MBE = \frac{1}{N} \sum_{i=1}^{N} (y_i - d_i) \quad (27) \]

15. Legate and McCabe’s Index, LMI, varies from 0 to 1. For better model fit, the value of LMI should be closer to 1 (Legates and Davis, 1997; Legates and McCabe, 2013; Legates and McCabe, 1999). LMI can be calculated using Equation (28).

\[ LMI = 1 - \left[ \frac{\sum_{i=1}^{N} |d_i - y_i|}{\sum_{i=1}^{N} |d_i - d_{mean}|} \right], \quad 0 < LMI \leq 1 \quad (28) \]

16. U95 compares the actual deviation amidst calculated value and measured value and thus examines short-term performance of the model. The formula in Equation (29) provides uncertainty at 95% confidence level. 1.96 is the coverage factor subsequent to confidence level. Expanded uncertainty (U95) (Behar, Khellaf and Mohammedi, 2015; Gueymard, 2014) can be formulated as:

\[ U_{95} = 1.96(SD^2 + RMSE^2)^{1/2} \quad (29) \]

17. t-statistic determines whether the prediction of the model at a particular confidence level is statistically significant or not. It is one of the most widely accepted parameters. t-statistic compares sample mean(s) to the null hypothesis. Smaller values of t-statistic indicate superior performance of the model. The value of zero accepts the null hypothesis, while an increase in the absolute value of t-statistic corresponds to an increase in the sample data and null hypothesis. t-statistic (Stone, 1993) can be given by Equation (30).

\[ t-stat = \sqrt{\frac{(N-1)MSE^2}{RMSE^2 - MBE^2}} \quad (30) \]

RESULTS AND DISCUSSION

In this research, three soft computing techniques (cubist model tree, ELM and MARS) were used for the assessment of rock slope reliability analysis. A comparison has been carried out between them considering different statistical indicators, like variance account factor, RMSE, MAPE, R², relative percentage difference (RPD) (Viscarra Rossel et al., 2006), receiver operating characteristic (ROC), …etc. All the three models showed significant results in terms of all fitness parameters. Coefficient of determination (R²) values for all the models were close to 1 in both training and testing cases. Adjusted determination coefficient (Adj. R²) values were almost equal to 1 in both training and testing cases. A value of NS closer to one indicates the accuracy of the model. The value of RSR is good for all the models. All the models gave values closer to zero for mean absolute error (MAE) in both training and testing cases, which shows the excellency of the models used. Values of MBE were almost zero in all the training and testing outputs using different models taken into consideration. All models showed good results of LMI as well. Values of different statistical parameters are shown in Table 1.
Table 1. Statistical parameters of cubist, ELM and MARS models

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Cubist</th>
<th>ELM</th>
<th>MARS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Training</td>
<td>Testing</td>
<td>Training</td>
</tr>
<tr>
<td>NS</td>
<td>0.999</td>
<td>0.999</td>
<td>0.999</td>
</tr>
<tr>
<td>RMSE</td>
<td>0.0045</td>
<td>0.006</td>
<td>0.006</td>
</tr>
<tr>
<td>R²</td>
<td>0.999</td>
<td>0.999</td>
<td>0.999</td>
</tr>
<tr>
<td>Adj R²</td>
<td>0.999</td>
<td>0.998</td>
<td>0.999</td>
</tr>
<tr>
<td>PI</td>
<td>1.994</td>
<td>1.99</td>
<td>1.991</td>
</tr>
<tr>
<td>Bias Factor</td>
<td>0.999</td>
<td>0.999</td>
<td>1</td>
</tr>
<tr>
<td>RSR</td>
<td>0.021</td>
<td>0.033</td>
<td>0.029</td>
</tr>
<tr>
<td>NMBE</td>
<td>-0.041</td>
<td>-0.013</td>
<td>-0.005</td>
</tr>
<tr>
<td>MAPE</td>
<td>0.244</td>
<td>0.316</td>
<td>0.387</td>
</tr>
<tr>
<td>RPD</td>
<td>46.932</td>
<td>30.288</td>
<td>33.522</td>
</tr>
<tr>
<td>WI</td>
<td>0.999</td>
<td>0.999</td>
<td>0.999</td>
</tr>
<tr>
<td>MAE</td>
<td>0.003</td>
<td>0.004</td>
<td>0.005</td>
</tr>
<tr>
<td>MBE</td>
<td>-0.0006</td>
<td>-0.0002</td>
<td>0</td>
</tr>
<tr>
<td>LMI</td>
<td>0.98</td>
<td>0.97</td>
<td>0.97</td>
</tr>
<tr>
<td>$U_{95}$</td>
<td>0.42</td>
<td>0.36</td>
<td>0.42</td>
</tr>
<tr>
<td>t-stat</td>
<td>0.93</td>
<td>0.148</td>
<td>0.009</td>
</tr>
<tr>
<td>β</td>
<td>1.774</td>
<td>2.162</td>
<td>1.772</td>
</tr>
</tbody>
</table>

A graph between actual and predicted values of factor of safety of training and testing cases for cubist model has been drawn, as shown in Figure 2.

All the values of training and testing data are close to the predicted line, which indicates a high prediction ability.

Figure (2): Actual vs. predicted values of training and testing data using cubist model
A graph between actual and predicted values of factor of safety of training and testing cases for ELM model has been drawn, as illustrated in Figure 3.

All the values of training and testing data are close to the predicted line, which shows that the model prediction ability is high.

Figure (3): Actual vs. predicted values of training and testing data using ELM model

A graph between actual and predicted values of factor of safety of training and testing cases for MARS model has been drawn, as shown in Figure 4.

All the values of training and testing data are close to the predicted line, which shows that the model prediction ability is high.

Figure (4): Actual vs. predicted values of training and testing data using MARS model
Taylor diagram provides a path to plot three different statistics on a 2D graph indicating the closeness of pattern to the observations (Taylor, 2001). For each model, three statistics as correlation coefficients are related to the azimuthal angle, represented by black dotted contours generating from the origin, root mean square error, represented by red contours and the third one is the standard deviation of the stipulated pattern, represented by proportioning through radial distance from the origin. Taylor diagram makes it convenient to differentiate between different models and to distinguish which model is the best compared to the other models.

Taylor diagram for training cases of different models has been plotted and the outcome is as shown in Figure 5.

![Taylor Diagram (Training)](image)

**Figure (5): Taylor diagram (training)**

From the curve in Figure 5, it is clear that the correlation coefficient of each model is greater than 0.99, the root mean square error value is close to zero and the standard deviation of each stipulated model is 1. All the models are giving values close to each other and accumulating to make a point.

Taylor diagram for testing cases of different models has been plotted and the outcome is as shown in Figure 6.
From Figure 6, it can be seen that all the models give values close to each other; hence, also in the case of testing of dataset, values from these models accumulate to make a point.

ROC curve gives an idea about the performance of the models used. The area covered on x-axis represents the area under curve of the model. Higher area under curve represents a good performance of the model. Different curves have been drawn using training and testing values of the models used and the outcome is as shown in Figure 7.
From the ROC curve (Figure 7), we can conclude that MARS model in both training and testing is showing the best performance among the three models. Reliability index (β) of observed data for training is 1.7713 and for testing is 2.1889. A comparison between reliability values of training cases has been carried out, as shown in Figure 8.

A comparison between reliability values of testing cases has been carried out as shown in Figure 9.
CONCLUSIONS

In this article, the verification of usefulness of three models of advanced soft computing (MARS, ELM and cubist) has been carried out for calculating the reliability of rock slope and evaluating the reliability obtained. Since all the methods used are data-specific, hence based on their efficiency on the utilized data, different outputs were obtained. Here, for the study of slope failure and geological condition, a total of 80 cases were used. Soft computing parameters, like R², Adj.R², MAE, MBE, AUC, … etc., were used for assessing the performance of soft computing models. From the abstracted results (from ROC curves), the largest value of area under curve (AUC) was obtained for MARS model, followed by cubist model and then by ELM model. All these models give prominent results to use them for reliability analysis of rock slope, but if we have to choose one from these models, then MARS model comes out to be the best. Training and testing data accumulated up while graphing Taylor and ROC curves. This article gives an idea of using computational methods and models in forecasting and analyzing the reliability of rock slope and can also be used in the field of geotechnical engineering while calculating other parameters.

REFERENCES


Li, Shaojun, Hong-Bo Zhao, and Zhongliang Ru. (2013). “Slope reliability analysis by updated support vector machine and Monte Carlo simulation”. Natural Hazards, 65 (1), 707-722.


