Received: 9 September 2016 | Revised: 27 July 2017 | Accepted: 27 July 2017

DOI: 10.1002/asmb.2273

SPECIAL ISSUE PAPER

WILEY

Risk assessment of failure of rock bolts in underground coal mines using support vector machines

Peng Jiang[1] | Peter Craig[2,3] | Alan Crosky[1] | Mojtaba Maghrebi[4,5] | Ismet Canbulat[2] | Serkan Saydam[2]

[1]School of Materials Science and Engineering, UNSW Australia, New South Wales, Australia

[2]School of Mining Engineering, UNSW Australia, New South Wales, Australia

[3]Jennmar Australia, New South Wales, Australia

[4]School of Civil and Environmental Engineering, UNSW Australia, New South Wales, Australia

[5]Department of Civil Engineering, Ferdowsi University of Mashhad, Razavi Khorasan, Iran

Correspondence
Peng Jiang, School of Materials Science and Engineering, UNSW Australia, New South Wales, Australia.
Email: peng.jiang@student.unsw.edu.au

Funding information
Australian Research Council, Grant/Award Number: LP 100200238, LP 140100153 and LP 100200238

In recent years, there has been an increasing incidence of failure of rock bolts due to stress corrosion cracking and localized corrosion attack in Australian underground coal mines. Unfortunately, prediction of the risk of failure from results obtained from laboratory testing is not necessarily reliable because it is difficult to properly simulate the mine environment. An alternative way of predicting failure is to apply machine learning methods to data obtained from underground mines. In this paper, support vector machines are built to predict failure of bolts in complex mine environments. Feature transformation and feature selection methods are applied to extract useful information from the original data. A dataset, which had continuous features and spatial data, was used to test the proposed model. The results showed that principal component analysis-based feature transformation provides reliable risk prediction.

KEYWORDS
feature learning, rock bolt, stress corrosion cracking, support vector machine, underground mining

1 | INTRODUCTION

Rock bolts are common components for supporting the roof and stabilizing the rockmass in underground coal mines. The bolts are typically 1.8 m long and 22 mm in diameter, produced from carbon or low alloy steel. Recently, the occurrence of catastrophic failure of rock bolts has been rising in some underground coal mines in Australia.[1,2] One potential consequence of such failures is roof fall, one of the major safety risks in underground coal mines.

Based on the recent investigation,[3] most of failures take the forms of stress corrosion cracking (SCC) and localized corrosion attack. The former is one of the most dangerous types of failures, often leading to catastrophic failures at a low stress state without any obvious signs. The latter, localized corrosion, occurs in the form of massive loss in local areas of the bolt, due to the increasing concentration of stress in its vicinity.

In order to uncover the hidden patterns behind corrosion behavior, parametric statistical models such as those given in Refs[4-7] have been developed. Some of these models can find a joint probability distribution over a few parameters, but none are able to effectively take all the potential contributors into consideration. In this sense, artificial intelligence (AI) has been applied to conduct reliability analysis in such nonlinear and complex systems.[8-12] Shi et al[13] used an artificial neural network (ANN) model to predict the crack growth rate in Type 304 stainless steel. The predicted values of crack

growth rate were in good agreement with the experimental values. In addition, the sensitivity analysis showed that the temperature and conductivity contributed the most to the crack growth rate. Kamrunnahar et al[14] developed an ANN model using the actual measurements of corrosion weight loss data of Alloy 22 to predict the future corrosion weight loss of this material. Results demonstrated a good agreement between the predicted and measured values under similar environmental and sampling conditions. Jiménez et al[15] presented an automatic model based on ANN to predict pitting corrosion behavior of austenitic stainless steel, and the results revealed the excellent classification performance compared with k-nearest neighbor (KNN) and classification tree (CT).

However, the AI techniques mentioned earlier are based on ANN, which suffers from overfitting, a serious shortcoming leading to the poor generalization. Moreover, a large number of samples are often required to boost the prediction performance. To overcome the disadvantages of ANN, support vector machine (SVM) has been developed,[16] which presents a more promising generalization ability through implementing the structural risk minimization principle. In this study, 3 SVM models with different feature processing techniques are developed to assess failure risk of rock bolts in underground coal mines. The results obtained from feature processing can help to improve data collection procedures and monitoring process.

This paper has 4 sections. Section 2 gives an introduction to kernel methods, SVM, and 2 feature processing techniques, followed by metrics for evaluating the performance of models. Section 3 discusses the results of the models tested on a real-world dataset. The conclusions from this study are presented in the last section.

2 | KERNEL METHODS

Kernel methods is a memory-based machine learning algorithm for the ability to extract important features shared by the training samples. It uses an inferred latent representation based on the original input, providing a fast and reliable solution to those problems. In this section, a brief introduction to kernels and how the SVM applies kernels to solve prediction problems are provided. Two feature processing techniques are also introduced.

2.1 | Introduction to kernels

Consider a training dataset (x, y), where x is the input vector for each pattern from $\mathcal{X}\in\mathbb{R}^n$ and y is the corresponding target class in $\mathcal{Y}\in\mathbb{R}$. One important application of machine learning algorithms is to predict the corresponding class y given some new input x. This requires similarity measures in $\mathcal{X}$, which is usually a function:

$$k : X \times X \to \mathbb{R}, \left(x, x^{'}\right) \mapsto k\left(x, x^{'}\right) \tag{1}$$

for all the $x, x^{'}$, satisfying:

$$k\left(x, x^{'}\right) = \phi(x), \phi\left(x^{'}\right) \tag{2}$$

where ϕ is a mapping from the original $\mathcal{X}$ to some dot product feature space $\mathcal{H}$. k is the similarity measure, which is usually called a *kernel*. The dot product is the projection of $\phi(x)$ into $\phi(x^{'})$, which measures the similarity between x and $x^{'}$ in this new feature space.

Positive definite kernels enable one to calculate the dot product without calculating the explicit forms of the new feature map, making projecting the original space into infinite dimensional feature space possible. In addition, this approach is often computationally much cheaper than the explicit computation, which is known as the *kernel trick*. To give a clear definition of positive definite kernels, Gram matrix and positive definite matrix are introduced here. Consider a kernel k and the inputs $x_1,..., x_n \in\mathcal{X}$. The Gram matrix refers to the $n \times n$ matrix $\mathbf{K}$ which has entries:

$$\mathbf{K}_{ij} = k\left(x_i, x_j\right) \tag{3}$$

The positive definite matrix is a real symmetric $n \times n$ matrix $\mathbf{K}$ if for all $c_1,..., c_n \in \mathbb{R}$:

$$\sum_{i,j=1}^{n} c_i c_j \mathbf{K}_{ij} \geq 0 \tag{4}$$

If the equality in Equation 4 occurs only when $c_1 = ... = c_n = 0$, then the matrix is called *strictly positive definite*. A positive definite kernel is the function that always gives rise to a positive definite Gram matrix for elements in $\mathcal{X}$. As is mentioned, the most important application of a positive definite kernel is that it can replace the mapping in calculating the dot product.

2.2 | Radial basis function

A kernel function[17] is defined to be a real-valued function measuring the similarity between 2 arguments, which is symmetric and non-negative. The Gaussian kernel is defined by:

$$k(x,x') = \exp\left(-\frac{1}{2}(x-x')^T\Sigma^{-1}(x-x')\right) \tag{5}$$

This kernel can be written as the following form if Σ is diagonal:

$$k(x,x') = \exp\left(-\frac{1}{2}\sum_{j=1}^{D}\frac{1}{\sigma_j^2}\left(x_j-x_j'\right)^2\right) \tag{6}$$

where σ_j defines the characteristic length scale of dimension j. If Σ is spherical, the isotropic kernel can be obtained as follows:

$$k(x,x') = \exp\left(-\frac{\|x-x'\|^2}{2\sigma^2}\right) \tag{7}$$

Here, σ^2 is the bandwidth and can be simplify using the following equation:

$$k(x,x') = \exp\left(-\gamma\|x-x'\|^2\right) \tag{8}$$

where γ must be greater than 0. This kernel is an example of a radial basis function (RBF), which is one of the most widely used kernels due to the good performance in handling non-linear problems. The function space defined by the RBF is much larger than other kernels. Generally, it is more flexible because it allows infinite complexity of models within its function space.

2.3 | Introduction to support vector machines

Introduced by Vapnik,[16] SVM applies the structural risk minimization principle to minimize the generalization error. Through finding an optimal separating hyperplane, SVM shows the good generalization ability in solving classification problems. Figure 1 illustrates the optimal hyperplane separates one category (represented by circles) from the other category (represented by triangles).

The input samples lying on the hyperplane satisfy the following equation:

$$w^T \cdot x + b = 0 \tag{9}$$

where w denotes the norm vector of regression coefficients controlling the smoothness of the model, and b is a bias term. As seen in Figure 1, the hyperplane lies in the middle of the margin, and the data points nearest from the optimal hyperplane are support vectors. Let the 2 categories y_i be 1 and -1. The optimal hyperplane can be found by solving the following equation:

$$y_i\left(w^T \cdot x_i + b\right) - 1 \geq 0 \tag{10}$$

The classification performance can be improved as the margin increases. In order to maximize the margin, SVM attempts to solve the following optimization problem:

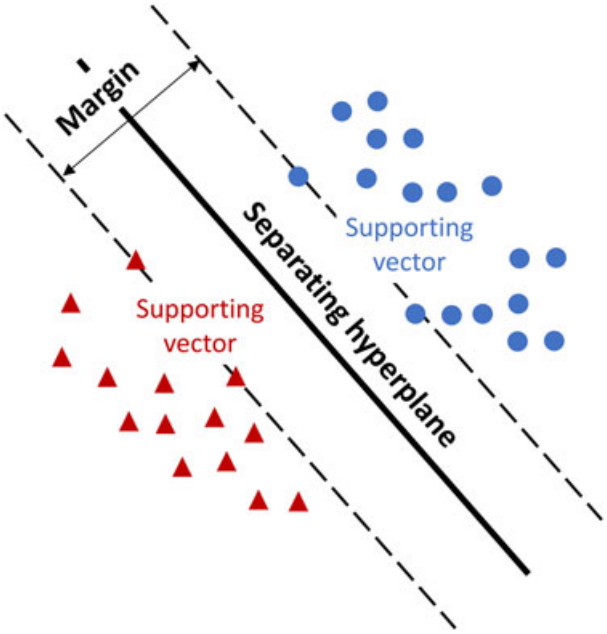

FIGURE 1 Schematic view of the optimal separating hyperplane [Colour figure can be viewed at wileyonlinelibrary.com]

$$Min\frac{1}{2}\|w\|^2 + C \cdot \sum_{i=1}^{n} \xi_i \tag{11}$$

where C is a regularized factor which determines the degree of penalized loss when an error occurs. ξ is an non-negative slack variable. The non-linearity classification or regression problems can be processed by mapping the input features to a higher-dimensional feature space from the original space by the kernel function:

$$f(x) = w^T \cdot \phi(x) + b \tag{12}$$

2.4 | Principal component analysis

For many real-life engineering problems, the information collected can be highly correlated. The reason for this is that 2 or more variables convey the same valuable information, which brings redundant and irrelevant information to the data. In order to obtain a more cost-effective model and remove the noise, principal component analysis (PCA)[18] is often used as a preprocessing method for processing features.

Consider m data points $\{x_1, \ldots, x_m\}$ in $\mathbb{R}^n$. These variables should be standardized before applying PCA. To represent the original m data points with a lower-dimensional version, an encoded vector $c_i \in \mathbb{R}^l$ is needed for each $x_i \in \mathbb{R}^l$. This transformation allows less storage memory if l is smaller than n. Generally, PCA transform x_i into c_i using a linear transformation matrix U^T:

$$c_i = U^T x_i \tag{13}$$

U is a $m \times m$ orthogonal matrix. Its columns u_i is the eigenvector of sample covariance matrix:

$$S = \frac{1}{m}\sum_{i=1}^{m} x_i {x_i}^T \tag{14}$$

The eigenvectors are associated with eigenvaluesλ_i:

$$\lambda_i u_i = S u_i \tag{15}$$

PCA firstly solves the above eigenvalue problem. Then, c_i can be obtained from the orthogonal transformations of x_i. These new variables c_i are called principal components. The maximum variance lies on the first principal component,

the second maximum variance on the second principal component, and so on. Because all principal components are orthogonal to each other, PCA actually projects the original data into a new space with the principal components as the basis. This projection reorganizes the structure of data in a way that best shows the valuable information.

2.5 | Gradient tree boosting

Unlike PCA which transforms the original data to a new feature space, gradient tree boosting (GTB) directly calculates the relative importance of the features, enabling one to screen out the unimportant features without changing the original data structures. GTB uses classification and regression tree (CART) as the based learners, as shown in Figure 2. CART partitions the input feature space recursively into a series of rectangles, and then defines a simple model in each one. One advantage of GTB is that it can be customized to particular needs of applications by allowing optimization of various arbitrary differentiable loss function.[19]

GTB is a tree-based ensemble model. Each tree is a weak classifier with multiple features as a binary splitting node that splits the input into 2 subsets. Let t be an internal node and s_t be its binary split according to a threshold value of 1 feature. t_L and t_R denote 2 children nodes, which divide a sample of size N_t into N_{tL} and N_{tR}, respectively. A predicted output is obtained from propagating 1 instance through the tree until it reaches a final node. A tree is constructed through learning the training set using a recursive procedure so that an optimized split s_t maximizes the decrease of some impurity measure $i(t)$:

$$\Delta \mathrm{i}(s,t) = i(t) - p_L i(t_L) - p_R i(t_R) \tag{16}$$

where $p_L = N_{tL}/N_t$ and $p_R = N_{tR}/N_t$. Gini impurity is often used as the impurity measure function $i(t) = 1 - \sum p^2(c)$, where the sum is over the labels c at the node, and $p(c)$ denotes the fraction of samples with label c that reach the node.

Similar to other boosting methods, GTB attempts to improve the prediction performance by building a vast number of weak base learners. Given a training dataset $(x_1, y_1), \ldots, (x_N, y_N)$, where x denotes the input variables of the dataset while y denotes 2 corresponding target classes in $y \in \mathcal{Y} = \{0, 1\}$. Each input data point x has p dimensions or features $(X_1, \ldots, X_p)$. Let $L(y, F(x))$ be the differentiable loss function of the model and M be the number of boosting trees. At each stage $1 \le m \le M$, a model F_m is built and generates residual errors $L(y, F_m(x))$. In the next boosting stage, $F_{m+1}(x) = F_m(x) + h(x)$ is built iteratively to compensate the residual errors produced by $F_m(x)$. It is evident that it improves the prediction performance by adding an estimator $h(x)$, which is obtained from fitting $h(x)$ to the residual $L(y, F(x))$. The sum of the residuals is $J = \sum_{i=1}^{N} L(y_i, F(x_i))$. Its derivatives can be obtained by:

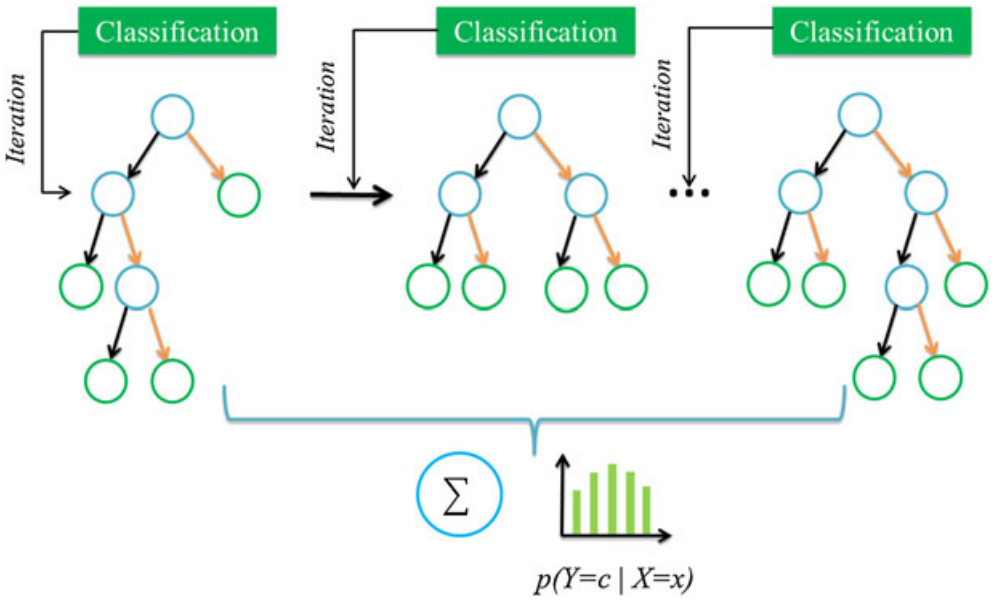

FIGURE 2 Schematic view of the boosting methods [Colour figure can be viewed at wileyonlinelibrary.com]

$$\frac{\partial J}{\partial F(x_i)} = \frac{\partial \sum_{i=1}^{N} L(y_i, F(x_i))}{\partial F(x_i)} = \frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \tag{17}$$

The concept of gradient is more universal and useful than the concept residuals. This is because it allows the usage of other loss functions. Steps for building GTB are as follows:

1. A constant function F_0 is initialized:

$$F_0(x) = \arg\min_{\gamma} \sum_{i=1}^{N} L(y_i, \gamma) \tag{18}$$

where γ is an initial guess which equals to $\frac{\sum_{i=1}^{N} y_i}{N}$.

2. $F_m(x)$ is then estimated and added to $F_{m-1}(x)$ iteratively:
2.1. Gradient r_i is calculated as:

$$r_i = -\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\bigg|_{F(x)=F_{m-1}(x)} \tag{19}$$

2.2. A base learner $h_m(x)$ is estimated by fitting it to r_i using the training set $(x_1, r_1), \ldots, (x_N, r_N)$. A weight γ_m is calculated by:

$$\gamma_m = \arg\min_{\gamma} \sum_{i=1}^{N} L(y_i, F_{m-1}(x_i) + \gamma h_m(x_i)) \tag{20}$$

Here, γ_m is used for adjusting the relative importance of h_m. $F_m(x)$ is updated by:

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x) \tag{21}$$

3. After iteration, an ensemble model can be obtained by:

$$F_M(x) = F_0(x) + \sum_{m=i}^{M} (\gamma_m h_m(x)) \tag{22}$$

Breiman[20] proposed a feature importance measuring method by averaging weighted Gini impurity ($p(t)\Delta i(s_t, t)$) decreases for all nodes t where the feature X_j is used. The Gini importance (GI)[21]is calculated by:

$$GI(X_j) = \frac{1}{N_T} \sum_{T} \sum_{t \in T: v(s_t)=X_j} p(t)\Delta \mathrm{i}(s_t, t) \tag{23}$$

where $p(t)$ is the fraction of samples reaching t and $v(s_t)$ is the feature used in split $s(t)$. By normalizing the feature importance among p features, relative feature importance can be obtained.

2.6 | Performance evaluation

For a binary classification problem, the predicted results involve True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). True (T) and False (F) refer to the actual values of our samples while Positive (P) and Negative (N) refer to the predicted results by the model. TP and TN denote samples that are correctly classified as positive

and negative respectively. Similarly, FP and FN denote samples that are misclassified as positive and negative respectively. Classification tasks often adopt a receiver operating characteristic (ROC) to quantify the classification performance. A ROC can be obtained through the tradeoff between True Positive Rate (TPR) and False Positive Rate (FPR), which are:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{P}} \cdot 100\% \tag{24}$$

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{N}} \cdot 100\% \tag{25}$$

For each observation, a probability value is assigned to classify it into different classes. Through connecting all of the values, which are in an increasing sequence, the ROC can be plotted. Furthermore, the area under the curve (AUC) can be estimated and used as a measure for quantifying the classification performance. The closer AUC comes to one, the better classification performance is.

3 | CASE STUDY

To investigate the effects of environmental variables and mechanical states on the premature failure of the bolts in underground coal mines, an Australian Research Council (ARC) and industry funded linkage research project has been granted at UNSW Australia.

3.1 | Database

Several underground coal mines with known prematurely failed rock bolts are investigated. Variables representing the environmental and geotechnical information are collected and used to build the database.[2] One dataset is extracted from the database, which contains 160 samples of roof panels collected from 1 mine site in New South Wales (NSW), Australia. This dataset contains 80 failures, which take the forms of SCC and localized corrosion attack. The difference in the fracture surface between the 2 types of failures can be seen in Figure 3.

Twelve features including 4 continuous features (length of roadway, dripper flow rate, age of bolts, and stress factor) and 8 categorical features (3 zones and 5 locations) were used as the input to predict the failure in this area (Table 1). Feature 1 represents the length of roadway. Feature 2 is the flow rate of ground water dripping from bolts. Feature 3 is the service age of bolts. Feature 4 is a normalized stress factor. The above 4 features are continuous values. Features 5 to 7 are 3 spatial indicators (3 zones), represented by binary number (0, 1). The bit value (0, 1) determines whether this sample is from that zone. Zone 1 has slightly higher calcium and magnesium with low sulfur, probably influenced by the connection to the distant old working by a fault zone. Zone 2 appears influenced by very close (<250 m) old flooded workings, providing higher sulfur. Zone 3 is relatively far away from the old workings, presenting the consistent flow

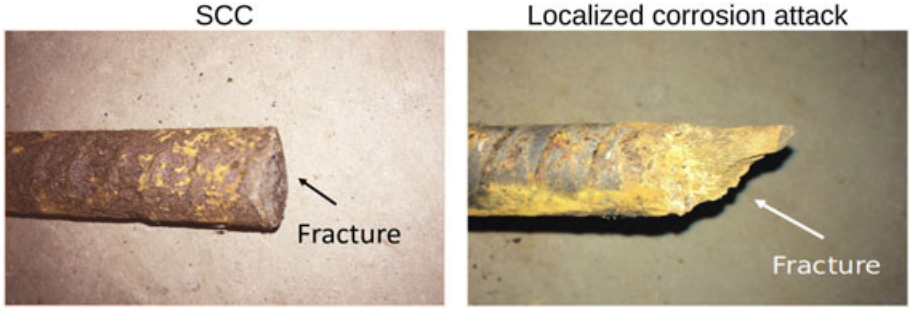

FIGURE 3 The left graph is side view of failed rock bolt due to SCC; the fracture occurs perpendicular to the axis of the bolt. The right graph is side view of failed bolt due to localized corrosion attack; severe section loss occurs on the surface of fracture [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 1 Input features for modeling

No. of feature	Feature	Range
1	Length of roadway	5–95 m
2	Dripper flow rate	0–261 mL/h
3	Age of bolts	2–13 years
4	Stress factor	0.05–0.9
5	Zone 1	(0,1)
6	Zone 2	(0,1)
7	Zone 3	(0,1)
8	Head gate 1	(0,1)
9	Cut-through 1	(0,1)
10	Head gate 2	(0,1)
11	Cut-through 2	(0,1)
12	Head gate 3	(0,1)

rate, low sulfur, and lower calcium or magnesium. Five spatial indicators, Features 8 to 12, represent 5 gate roads in the mine. Features 8, 10, and 12 are 3 headings. Features 9 and 11 are 2 cut-throughs between headings. The 5 features are also represented by binary numbers. A more detailed description can be found in Figure S1 in supporting information.

The roof panels, which had bolts fallen, were labeled with high risk of failure (failed), and those with no fallen bolts were labeled with low risk (unfailed). After extensive training, the SVM model is programmed to assign a high risk or low risk of failure to a roof panel given a set of new features. This dataset is balanced, which means it contains 50% high-risk samples and 50% low-risk samples. In order to get a better performance, normalization is conducted to scale all samples to the range [0, 1] by the following formula:

$$X_{i^{(m)}}(k) = \frac{x_{i^{(m)}} - \min_{n=1,N}\left\{x_{i^{(n)}}\right\}}{\max_{n=1,N}\left\{x_{i^{(n)}}\right\} - \min_{n=1,N}\left\{x_{i^{(n)}}\right\}} \tag{26}$$

3.2 | Results and discussion

LIBSVM,[22] a Python-based machine learning library, is used to build SVM models. This library allows a probabilistic prediction for each predicted class. Scikit-learn[23] is used to implement PCA and GTB models. Ten-fold cross validation is conducted to verify the performance: 9 of the 10-folds are used for training, and the remaining are used to assess the predicting its performance on out-of-sample data. Table 2 shows the PCA results of the 12 input features. It is evident that the cumulative variance ratio obtained from the first 6 principal is around 0.93, containing most information in the original dataset. In addition, the last 2 principal components contribute marginally to the variance, demonstrating the existence of the redundancy in the dataset.

In order to reduce noise and redundancy in data, only several components are selected to replace the original data in the following modeling. There are various rules of thumb on selection. One rule is the "90% rule"[24]; present the top k components that contain 90% cumulatively of the total variation. Another rule is a scree plot of eigenvalues, as shown in Figure 4. The aim is to observe an "elbow-effect".[25] If a dramatic drop in the variation is observed in the $k + 1$component, then the first k components are selected. It is apparent that a significant change point (elbow) can be observed in the second component in Figure 4. But the variance ratio obtained from the first component is only around 0.3, which does not convey enough information. The second change point can be observed in the seventh component. Based on 2 selection rules, the top 6 components are selected, which explains 93% of the information.

To compare the classification performance between feature transformation and feature selection, GTB is employed to select the same number of features. In this work, 100 trees are built to form a strong GTB classifier. Figure 5 shows the rank of relative importance of the features obtained from GTB. According to the results calculated from GTB, 4 continuous features, length of roadway, dripper flow rate, age of bolts, and stress factor, are the most important predictors. Two spatial indicators, Feature 8 and Feature 10, also contribute significantly to correct classification. However, the types of

TABLE 2 PCA of the 12 input features

Principal component	Percentage of variance	Percentage of cumulative variance
1	0.308	0.308
2	0.187	0.495
3	0.144	0.638
4	0.117	0.756
5	0.091	0.847
6	0.084	0.931
7	0.040	0.971
8	0.017	0.987
9	0.007	0.994
10	0.006	1.000
11	0	1.000
12	0	1.000

ground water are 3 of the least important predictors, indicating that the difference in water chemistry of the 3 water types is not a powerful discriminating variable for classification. Further data collection in this mine site should focus on the features found to be most important. Like PCA, the top 6 features are used as the input for SVM modeling.

Based on results of PCA and GTB, the types of groundwater in this database evidently reveal limited information. In fact, sampling groundwater was performed following a procedure of collection, storage, and transport back to an accredited laboratory within 2 days. Limited by the environment and techniques, the groundwater chemistry may not represent the general information in that particular area. In future data collection procedure, a more precise method for sampling and analyzing the groundwater chemistry is needed. The cut-throughs were noted as not having groundwater dripping from bolts. Therefore, redundancy may come from the Feature 2 (dripper flow rate) and Features 8 to 12 (5 spatial indicators) because they may convey the same information.

Grid search is conducted to optimize the hyperparameters. The parameter searching can be found in Figure S2 in supporting information. The AUC obtained from the PCA feature transformation, GTB feature selection, and the conventional SVM based on the optimized hyperparameters can be seen in Figure 6. PCA (feature transformation) combined with SVM (PCA-SVM), and the conventional SVM both achieved an average AUC of 0.85, compared with an AUC of 0.82 for GTB (feature selection) combined with SVM (GTB-SVM). The optimized models are then applied on the test

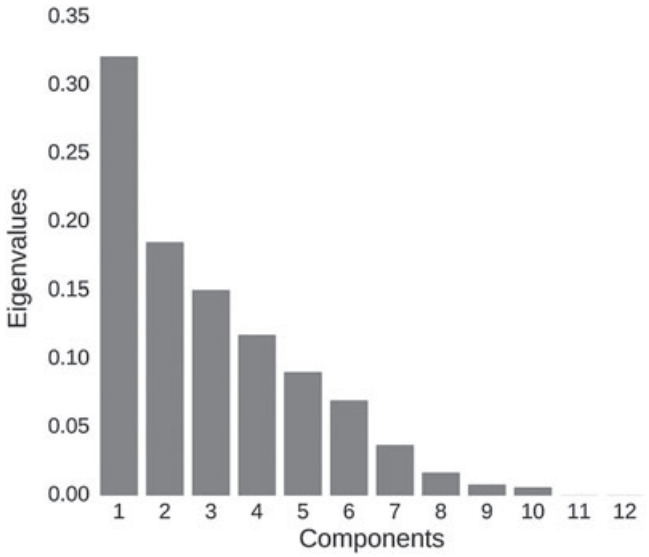

FIGURE 4 Rank of eigenvalues of principal components

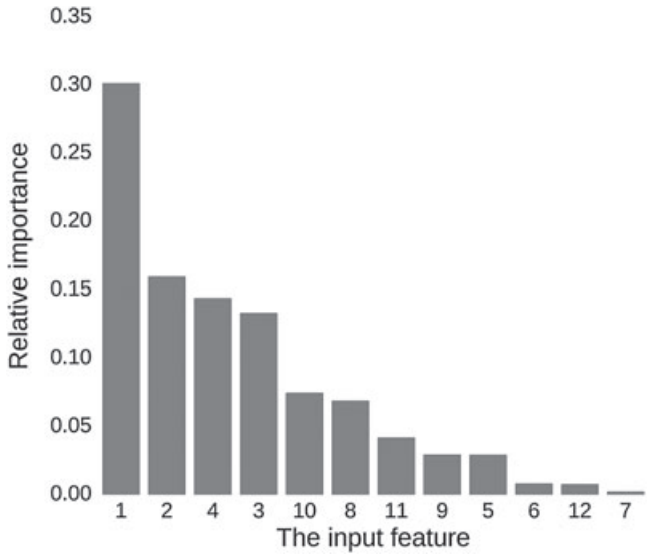

FIGURE 5 Rank of relative importance of the features. Features 1 to 4 are length of roadway, dripper flow rate, age of bolts, and stress factor respectively; features 5 to 7 are 3 zones with different water types; features 8 to 12 are spatial indicators

set. An AUC of 0.83, 0.86, and 0.81 are obtained from conventional SVM, PCA-SVM, and GTB-SVM, respectively, as shown in the lower-right graph in Figure 6 and Table 3.

Based on the results, feature transformation outperforms the feature selection regarding the classification performance. Using only 6 features, PCA-SVM model even achieves a higher AUC than conventional SVM, demonstrating

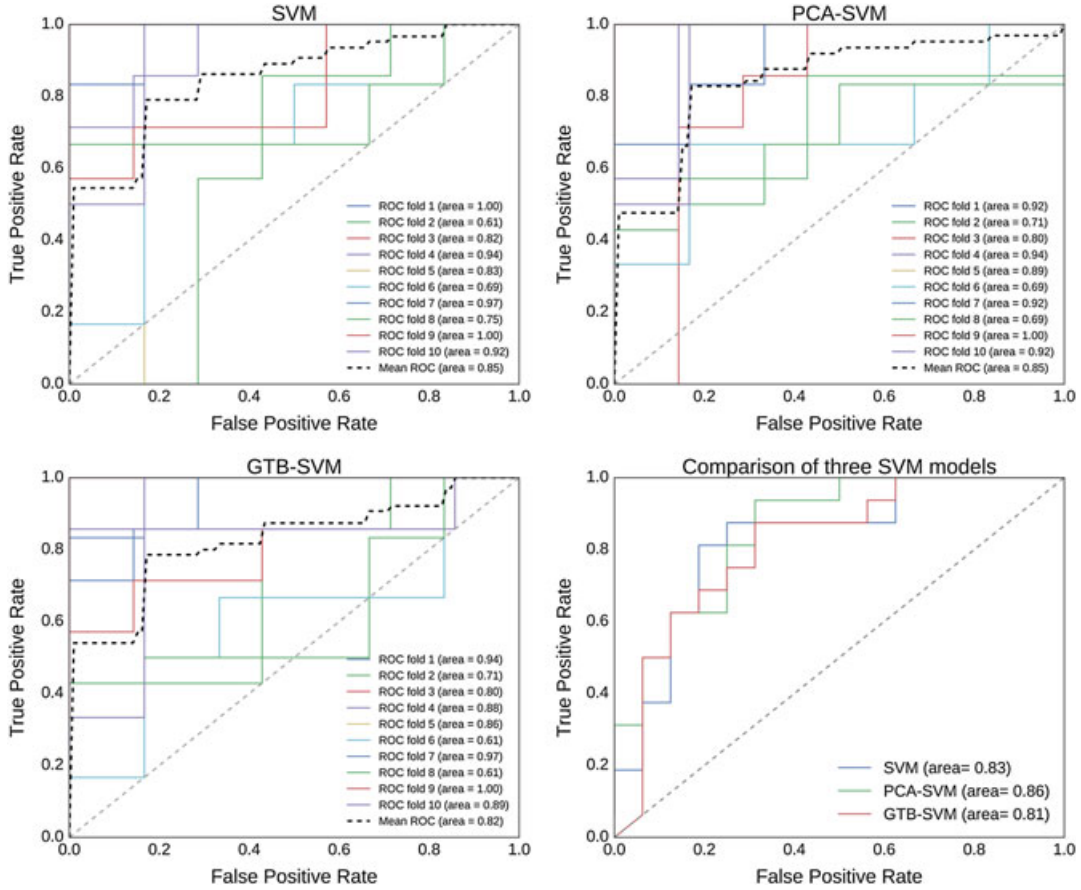

FIGURE 6 The 10-folds AUC of conventional SVM (upper left), PCA-SVM (upper right), and GTB-SVM (lower left) on the training set, while the lower-right graph is comparison of AUC of the above 3 models on the test set [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 3 Classification performance of 3 models

Models	Training	Testing
SVM	0.85	0.83
PCA-SVM	0.85	0.86
GTB-SVM	0.82	0.81

its efficiency in information reconstruction. Feature selection by GTB is able to rank the original input features according to the contribution of each feature to the classification. However, with only the top 6 features (in relative importance), the classification performance could not be boosted, which means that other less important features still contain a considerable number of useful information.

4 | CONCLUSION

In this study, we proposed SVM models for the assessing the risk of failure of rock bolts in underground coal mines. The classification performances of conventional SVM, PCA combined with SVM, and GTB combined with SVM have been compared. Specifically, the combined PCA-SVM model has significant advantages over the other 2 methods. PCA-SVM model reduced the number of the features to 6 from 12 and achieved an AUC of 0.85 on the training set and an AUC of 0.86 on the test set. GTB reduced the dimensions by feature selection, resulting in an AUC of 0.82 and an AUC of 0.81 on the training set and test set, respectively, demonstrating a considerable number of information loss after the feature selection. The dataset used in this study consists of the cross-sectional data (continuous) and the spatial data (discrete). In addition, the results showed that PCA, feature transformation, can achieve a better extraction on the complex data than feature selection. A further study will be conducted to test non-linear feature transformation methods like autoencoders, as well as more complex datasets.

ACKNOWLEDGEMENT

This work was supported by the Australian Research Council Linkage grant Project LP 100200238 and LP 140100153.

REFERENCES

1. Crosky A, Smith B, Hebblewhite B. Failure of rockbolts in underground mines in Australia. *Practical Failure Analysis*. 2003;3(2):70-78.
2. Craig P, Serkan S, Hagan P, et al. Investigations into the corrosive environments contributing to premature failure of Australian coal mine rock bolts. *Int J Min Sci Technol*. 2016;26(1):59-64.
3. Craig P, Saydam S, Hagan P, Hebblewhite B, Vandermaat D, Crosky A, Elias E. An investigation of the corrosive impact of groundwater on rock bolts in underground coalmines. *AusRock 2014: Third Australian Ground Control in Mining Conference* 2014:277–286.
4. Galvele Je R. A stress corrosion cracking mechanism based on surface mobility. *Corros Sci*. 1987;27(1):1-33.
5. Galvele JR. Surface mobility mechanism of stress-corrosion cracking. *Corros Sci*. 1993;35(1):419-434.
6. Gest RJ, Troiano AR. Stress corrosion and hydrogen embrittlement in an aluminum alloy. *Corrosion*. 1974;30(8):274-279.
7. Nordsveen M, Nešic S, Nyborg R, Stangeland A. A mechanistic model for carbon dioxide corrosion of mild steel in the presence of protective iron carbonate films—part 1: Theory and verification. *Corrosion*. 2003;59(5):443-456.
8. Caleyo F, Valor A, Alfonso L, Vidal J, Perez-Baruch E, Hallen JM. Bayesian analysis of external corrosion data of non-piggable underground pipelines. *Corros Sci*. 2015;90:33-45.
9. Mabbutt S, Picton P, Shaw P, Black S. Review of artificial neural networks (ANN) applied to corrosion monitoring. *J Phys Conf Ser*. 2012;2012: 012114
10. Parthiban T, Ravi R, Parthiban GT, Srinivasan S, Ramakrishnan KR, Raghavan M. Neural network analysis for corrosion of steel in concrete. *Corros Sci*. 2005;47(7):1625-1642.
11. Kurz D, Lewitschnig H, Pilz J. Failure probability estimation with differently sized reference products for semiconductor burn-in studies. *Appl Stoch Model Bus Ind*. 2015;31(5):732-744.
12. Pao HK, Lee YJ, Huang CY. Statistical learning methods for information security: fundamentals and case studies. *Appl Stoch Model Bus Ind*. 2015;31(2):97-113.

13. Shi J, Wang J, Macdonald DD. Prediction of crack growth rate in type 304 stainless steel using artificial neural networks and the coupled environment fracture model. *Corros Sci*. 2014;89:69-80.

14. Kamrunnahar M, Urquidi-Macdonald M. Prediction of corrosion behaviour of alloy 22 using neural network as a data mining tool. *Corros Sci*. 2011;53(3):961-967.

15. Jiménez-Come M, Turias I, Trujillo F. An automatic pitting corrosion detection approach for 316L stainless steel. *Mater Des*. 2014;56:642-648.

16. Cortes C, Vapnik V. Support-vector networks. *Mach learn*. 1995;20(3):273-297.

17. Murphy KP. Machine learning: a probabilistic perspective; 2012.

18. Wold S, Esbensen K, Geladi P. Principal component analysis. *Chemom Intel Lab Syst*. 1987;2(1–3):37-52.

19. Elith J, Leathwick JR, Hastie T. A working guide to boosted regression trees. *J Anim Ecol*. 2008;77(4):802-813.

20. Breiman L, Friedman J, Stone CJ, Olshen RA. *Classification and Regression Trees*. London, England: CRC press; 1984.

21. Louppe G, Wehenkel L, Sutera A, Geurts P. Understanding variable importances in forests of randomized trees. *Adv Neural Inf Proces Syst*. 2013;2013:431-439.

22. Chang C-C, Lin C-J. LIBSVM: a library for support vector machines. *ACM Trans Intell Syst Technol (TIST)*. 2011;2(3):27

23. Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: machine learning in python. *J Mach Learn Res*. 2011;12:2825-2830.

24. Jolliffe I. *Principal Component Analysis*. Hoboken, New Jersey, United States: Wiley Online Library; 2002.

25. Zuur AF, Ieno EN, Smith GM. Principal component analysis and redundancy analysis. *Anal Ecol Data*. 2007;193-224.

SUPPORTING INFORMATION

Additional Supporting Information may be found online in the supporting information tab for this article.

How to cite this article: Jiang P, Craig P, Crosky A, Maghrebi M, Canbulat I, Saydam S. Risk assessment of failure of rock bolts in underground coal mines using support vector machines. *Appl Stochastic Models Bus Ind*. 2017;1–12. https://doi.org/10.1002/asmb.2273