Prediction of Fatigue Strength in Steels by Linear Regression and Neural Network

Takayuki Shiraiwa1, Yuto Miyazawa2 and Manabu Enoki

Department of Materials Engineering, The University of Tokyo, Tokyo 113-8656, Japan

This paper examines machine learning methods to predict fatigue strength with high accuracy using existing database. The fatigue database was automatically classified by hierarchical clustering method, and a group of carbon steels was selected as a target of machine learning. In linear regression analyses, a model selection was conducted from all possible combinations of explanatory variables based on cross validation technique. The derived linear regression model provided more accurate prediction than existing empirical rules. In neural network models, local and global sensitivity analyses were performed and the results of virtual experiments were consistent with existing knowledge in materials engineering. It demonstrated that the machine learning method provides prediction of fatigue performance with high accuracy and is one of promising method to accelerate material development. [doi:10.2320/matertrans.ME201714]

(Received May 29, 2018; Accepted June 18, 2018; Published July 23, 2018)

Keywords: steel, fatigue strength, linear regression, neural network, model selection, machine learning, materials integration

1. Introduction

Structural materials are mostly used for a long time and the time-dependent performances such as fatigue, creep and corrosion are directly related to the safety of our infrastructure. As a demand of new materials for specific applications increases and the cost of the experiments to evaluate the time-dependent performances rises, it is becoming increasingly important to reduce the number of experiments in the material development process. Against such a background, many attempts have been made to develop a methodology for predicting the fatigue performances by using physical models, numerical simulation, empirical rules and machine learning. In order to capture the complicated relationship between the manufacturing conditions and the performance, it is effective to decompose the material system into four principal elements of process, structure, properties and performance and to create models for each relationship.1) By integrating these models, the performance prediction can be realized.2–5) Despite a great deal of effort using computational simulations,6–8) there is still a lack of models that predict dislocation structure formation and microstructurally small crack growth under cyclic loading, and it is still challenging to develop the methodology to predict the total fatigue life with the current theoretical understanding. The need to bridge models of the elementary processes over different length scales and time scales makes the fatigue prediction more difficult. Although the empirical rules can be one way to obtain the missing models, the present measurement techniques do not provide sufficient data to derive the empirical models for dislocation and crack initiation.

In contrast, vast amounts of data on the process-performance relationship have been accumulated over the years for engineering applications. These data are not helpful for understanding the elementary processes of fatigue but are useful for interpolative prediction of performance within a limited range of materials. With the rapid improvement of computer performance and development of software, numerous machine learning algorithms have been proposed over several decades. In the traditional approach, we must select an appropriate learning algorithm to solve a problem from the set of the numerous algorithms, and choose the explanatory variables from many features in database. This traditional approach has resulted in successful predictions in various fields. In the field of mechanical properties, Mukherjee et al.9) proposed a multivariate regression model to predict the hardness of thermo-mехanically treated rebars as a function of chemical composition and tempering parameters. Ozerdem and Koluksa10) predicted multiple mechanical properties of copper alloy by using neural network. Collins et al.11) utilized Bayesian neural network to predict tensile properties of Ti–6Al–4V alloys with high accuracy. Also in the case of fatigue prediction, these approaches have been adopted. Malinov et al.12) developed neural network models to predict various mechanical properties including fatigue strength for titanium alloys. Agrawal et al.13) compared many kinds of machine learning algorithms for the prediction of fatigue strength in steels. Results of these researches showed that the accuracy of the prediction of fatigue strength was lower than that of other mechanical properties such as tensile strength and hardness.

This can be explained by the fact that fatigue involves more complex interactions between microstructure, stress and environment than static mechanical properties. A more sophisticated approach is required for fatigue prediction.

The purpose of this paper is to examine machine learning methods suitable for fatigue problems and to predict fatigue strength with high accuracy using existing database. The database contains fatigue properties on various kinds of materials. Since the mechanism of fatigue failure differs from material to material, empirical formulas should be derived separately for each material group. For this reason, the fatigue dataset was automatically classified by hierarchical clustering method, and a group of carbon steels was selected as a target of machine learning. After data preprocessing, there remained 393 samples with 21 explanatory variables such as chemical compositions, process parameters, heat treatment conditions and inclusion parameters, and one objective variable of fatigue strength. In linear regression analyses, a model

1Corresponding author, E-mail: shiraiwa@mme.m.t.u-tokyo.ac.jp
2Graduate Student, The University of Tokyo. Present address: IHI Corporation, Yokohama 235-8501, Japan.
selection was conducted from all possible combinations of explanatory variables based on cross validation technique in order to minimize the prediction error. In neural network models, a suitable network topology and an appropriate activation function were examined. Comparing the linear regression and neural network models, the advantage and disadvantage of each model were discussed. Also, by using these models, local and global sensitivity analyses were performed to evaluate the significance of each explanatory variable on the fatigue strength.

2. Methods

2.1 Data

Fatigue data sheet (FDS) provided from national institute of material science (NIMS) was used, which is one of the world’s largest fatigue databases and contains general steels, stainless steels, cast irons, aluminum alloys, nickel alloys, titanium alloys and magnesium alloys. In total, 689 types of metallic materials are linked to the features including fatigue strength, chemical compositions, processing parameters, heat treatment conditions, inclusion parameters and grain size. The chemical composition consists of 25 different elements: Fe, C, Si, Mn, P, S, Ni, Cr, Cu, Mo, Al, N, B, Nb, Ti, W, V, Co, Sb, O, Sn, Mg, Zn, Zr and H. The processing parameters include reduction ratio and heat treatment parameters in pre-heating, normalizing, quenching, tempering, annealing, solution and aging. The inclusion parameters are area fraction of inclusions deformed by plastic work (dA), inclusions occurring in discontinuous arrays (dB) and isolated inclusions (dC). The grain size is recorded for ferrite and prior austenite in several steels, and primary α and transformed β phase of titanium alloy. The fatigue strength was determined at 107 cycles in rotating bending fatigue tests under room temperature.

2.2 Hierarchical clustering and data preprocessing

The fatigue dataset was classified purely based on chemical composition by hierarchical clustering method in order to avoid the artificial categorization. Clustering is one of unsupervised learning methods and classifies data points into different data aggregates, that is, clusters. Among various clustering methods, hierarchical clustering is capable of grouping data points on various scales by creating a dendrogram, and is considered to be effective for classification of metal materials. Hierarchical clustering algorithms are categorized into agglomerative (bottom-up) and divisive (top-down). In the current study, this hierarchical agglomerative clustering was performed with the 689 material samples using the 25 chemical contents as explanatory variables.

It is obvious that the result of the clustering depends on the definitions of distance between data points (intra-cluster distance) and distance between clusters (inter-cluster distance). In order to determine the optimum definitions, clustering analyses were performed with all materials in the database by changing the definitions of intra-cluster and inter-clusters, and it was evaluated whether five kinds of metal materials (iron alloy, aluminum alloys, nickel alloys, titanium alloys and magnesium alloys) can be divided into each cluster. The mean, minimum and maximum values of the parameters used for the clustering analysis is shown in Table 1. Euclidean distance, standardized Euclidean distance and Mahalanobis distance were used as the intra-element distances. Given a dataset comprising n samples of $X = (x_1, x_2, \ldots, x_n)^T$ and each sample has $p$ explanatory variables $x_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^T$, the distances between the data points $x_i$ and $x_j$ are defined as follows:

$$d_{ij}^{\text{Euclid}} = \sqrt{\sum_{k=1}^{p} (x_{ik} - x_{jk})^2},$$

$$d_{ij}^{\text{standard Euclid}} = \sqrt{\sum_{k=1}^{p} \left(\frac{x_{ik} - x_{jk}}{\sigma_k}\right)^2},$$

$$d_{ij}^{\text{Mahalanobis}} = \sqrt{(x_i - x_j)^T \Sigma^{-1} (x_i - x_j)},$$

where $\sigma_k$ is the standard deviation of feature $k$, $C$ is the covariance matrix of the entire data $X$. The inter-cluster distance was calculated by shortest distance method, farthest distance method, group average method, Ward’s method and centroid distance method. In the shortest distance method, the distance between the closest two data points belonging to two clusters is measured. Conversely, the farthest distance method uses the distance between the most distant two data points. The group average method uses an average value of distances for all the combinations of two data points belonging to two clusters. The Ward’s method merges the clusters to minimize the difference between the

<table>
<thead>
<tr>
<th>Features</th>
<th>Mean value</th>
<th>Minimum value</th>
<th>Maximum value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fe (mass %)</td>
<td>85.6</td>
<td>0.003</td>
<td>99.1</td>
</tr>
<tr>
<td>C (mass %)</td>
<td>0.33</td>
<td>0</td>
<td>3.63</td>
</tr>
<tr>
<td>Si (mass %)</td>
<td>0.31</td>
<td>0</td>
<td>3.09</td>
</tr>
<tr>
<td>Mn (mass %)</td>
<td>0.73</td>
<td>0</td>
<td>1.85</td>
</tr>
<tr>
<td>P (mass %)</td>
<td>0.015</td>
<td>0</td>
<td>0.035</td>
</tr>
<tr>
<td>S (mass %)</td>
<td>0.011</td>
<td>0</td>
<td>0.031</td>
</tr>
<tr>
<td>Ni (mass %)</td>
<td>2.1</td>
<td>0</td>
<td>53.2</td>
</tr>
<tr>
<td>Cr (mass %)</td>
<td>3.2</td>
<td>0</td>
<td>25.3</td>
</tr>
<tr>
<td>Cu (mass %)</td>
<td>0.085</td>
<td>0</td>
<td>3.3</td>
</tr>
<tr>
<td>Mo (mass %)</td>
<td>0.30</td>
<td>0</td>
<td>9.2</td>
</tr>
<tr>
<td>Al (mass %)</td>
<td>3.4</td>
<td>0</td>
<td>98.4</td>
</tr>
<tr>
<td>N (mass %)</td>
<td>0.005</td>
<td>0</td>
<td>0.092</td>
</tr>
<tr>
<td>B (mass %)</td>
<td>0.0001</td>
<td>0</td>
<td>0.0056</td>
</tr>
<tr>
<td>Nb (mass %)</td>
<td>0.005</td>
<td>0</td>
<td>0.3</td>
</tr>
<tr>
<td>Ti (mass %)</td>
<td>3.2</td>
<td>0</td>
<td>99.9</td>
</tr>
<tr>
<td>W (mass %)</td>
<td>0.041</td>
<td>0</td>
<td>1.9</td>
</tr>
<tr>
<td>V (mass %)</td>
<td>0.15</td>
<td>0</td>
<td>4.4</td>
</tr>
<tr>
<td>Co (mass %)</td>
<td>0.18</td>
<td>0</td>
<td>12.2</td>
</tr>
<tr>
<td>Sn (mass %)</td>
<td>0.00004</td>
<td>0</td>
<td>0.002</td>
</tr>
<tr>
<td>O (mass %)</td>
<td>0.005</td>
<td>0</td>
<td>0.2</td>
</tr>
<tr>
<td>Mg (mass %)</td>
<td>0.00004</td>
<td>0</td>
<td>0.005</td>
</tr>
<tr>
<td>Zn (mass %)</td>
<td>0.37</td>
<td>0</td>
<td>95.9</td>
</tr>
<tr>
<td>Zr (mass %)</td>
<td>0.068</td>
<td>0</td>
<td>4.6</td>
</tr>
<tr>
<td>H (mass %)</td>
<td>0.0021</td>
<td>0</td>
<td>0.16</td>
</tr>
<tr>
<td>Fe (mass %)</td>
<td>0.0002</td>
<td>0</td>
<td>0.012</td>
</tr>
</tbody>
</table>
variance of the cluster after merging and the sum of variance of clusters before merging. The centroid method uses the distance between the centroids of clusters. Using three intra-cluster distances and five inter-cluster distances, clustering analyses were performed with a total of 15 algorithms. To assess the quality of the clustering, entropy was calculated. The entropy of each cluster, \( H(k) \), was defined by the following equation:

\[
H(k) = -\sum_{k=1}^{K} p_k \log p_k,
\]

where \( k \) is cluster index, \( K \) is number of clusters (\( K = 5 \)), \( |C(k,j)| \) is the number of data points belonging to cluster \( k \), \( |C(k,j)| \) is the number of data points corresponding to material \( j \) (iron alloy, aluminum alloys, etc.) in cluster \( k \). Low entropy indicates high quality clustering. The total entropy of five clusters was compared among the algorithms to determine the optimal definitions of intra-cluster distance and inter-cluster distance.

Using the determined definitions of distances, 630 types of iron alloys in the database were clustered. In the NIMS fatigue data sheet, iron alloys are classified into 15 categories of bearing steel, carbon steel for machine structural use, spring steel, stainless steel, tool steel and spheroidal graphite cast iron. To compare with these categories, the maximum number of clusters was set to 15. As the details are to be mentioned in section 3.1, 509 types of steels were classified into one cluster, and were used in the following data preprocessing.

### 2.3 Data preprocessing

In the fatigue database, not all features exist for all materials. In order to reduce the number of explanatory variables and not to reduce the number of material samples as much as possible, the data was preprocessed as follows:

1. Allo\-ying elements whose content is 0 mass\% in all 630 steels were excluded from the explanatory variables.
2. Material samples subjected to heat treatment other than quenching, tempering and normalizing were excluded. These three heat treatments have two variables of temperature \( T \) (K) and time \( t \) (min). These two variables were summarized into one variable of \( T\log(t) \). For materials not quenched or tempered, \( T\log(t) = 0 \) was assigned.
3. The grain size of ferrite and austenite were excluded from the explanatory variables because only either of them was recorded in the most cases. Since primary \( \alpha \) and transformed \( \beta \) phase of titanium alloy do not exist in steels, they were excluded.

After the preprocessing, there remained 21 explanatory variables of reduction ratio, \( \Delta A \), \( \Delta B \), \( \Delta C \), Fe, C, Si, Mn, P, S, Ni, Cr, Cu, Mo, Al, N, Ti, O, normalizing, quenching and tempering. The number of materials was 393. The mean, maximum, and minimum values of each explanatory variable are shown in Table 2. These 393 samples were randomly divided into training dataset with 360 samples and test dataset with 33 samples to evaluate the accuracy of linear regression model.

### 2.4 Linear regression model

Linear regression is one of the simplest and powerful regression method in which the relationship between explanatory and objective variables is modelled by a linear equation. Multivariate linear regression model can be expressed as follows:

\[
y = \sum_{i}^{p} a_i x_i,
\]

where \( y \) is objective variable, \( x_i \) is explanatory variable and \( a_i \) is constant coefficient for explanatory variables \( x_i \). Although more complex model gives more accurate prediction in the training dataset, if the model is too complex, the accuracy in the validation dataset decrease due to over-fitting. To avoid this problem, a variable selection from all \( 2^{21} - 1 \) combinations of explanatory variables was conducted to minimize the validation error based on cross validation technique. The procedure of the cross validation is illustrated in Fig. 1. At first, 360 training samples were divided for every ten into 36 groups. Linear regression coefficients were calculated for 35 groups except the first group and fatigue strength of first group was predicted by using these coefficients. Then, root mean squared error (RMSE) of this prediction was calculated. After that, 35 groups except the second group were used to calculate coefficients and RMSE was calculated for the second groups. Repeating these procedures, 36 values of RMSE were calculated and total

<table>
<thead>
<tr>
<th>Features</th>
<th>Mean value</th>
<th>Minimum value</th>
<th>Maximum value</th>
<th>Normalized ( a_i ), MPa</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fe (mass%)</td>
<td>97.23</td>
<td>95.11</td>
<td>99.67</td>
<td>-</td>
</tr>
<tr>
<td>C (mass%)</td>
<td>0.413</td>
<td>0.22</td>
<td>0.63</td>
<td>224</td>
</tr>
<tr>
<td>Si (mass%)</td>
<td>0.320</td>
<td>0.16</td>
<td>2.65</td>
<td>110</td>
</tr>
<tr>
<td>Mn (mass%)</td>
<td>0.829</td>
<td>0.21</td>
<td>1.6</td>
<td>68.5</td>
</tr>
<tr>
<td>P (mass%)</td>
<td>0.016</td>
<td>0.004</td>
<td>0.03</td>
<td>47</td>
</tr>
<tr>
<td>S (mass%)</td>
<td>0.014</td>
<td>0.003</td>
<td>0.06</td>
<td>-</td>
</tr>
<tr>
<td>Ni (mass%)</td>
<td>0.506</td>
<td>0.01</td>
<td>2.78</td>
<td>88.1</td>
</tr>
<tr>
<td>Cr (mass%)</td>
<td>0.537</td>
<td>0.01</td>
<td>1.12</td>
<td>134</td>
</tr>
<tr>
<td>Cu (mass%)</td>
<td>0.064</td>
<td>0.01</td>
<td>0.26</td>
<td>18.4</td>
</tr>
<tr>
<td>Mo (mass%)</td>
<td>0.069</td>
<td>0</td>
<td>0.24</td>
<td>34.6</td>
</tr>
<tr>
<td>Al (mass%)</td>
<td>0.001</td>
<td>0</td>
<td>0.64</td>
<td>137</td>
</tr>
<tr>
<td>N (mass%)</td>
<td>3.81×10^{-4}</td>
<td>0</td>
<td>0.015</td>
<td>144</td>
</tr>
<tr>
<td>Ti (mass%)</td>
<td>8.65×10^{-4}</td>
<td>0</td>
<td>0.01</td>
<td>24.7</td>
</tr>
<tr>
<td>O (mass%)</td>
<td>8.14×10^{-4}</td>
<td>0</td>
<td>0.003</td>
<td>-372</td>
</tr>
<tr>
<td>Inclusion ( \Delta A ) (%)</td>
<td>0.048</td>
<td>0</td>
<td>0.13</td>
<td>-</td>
</tr>
<tr>
<td>Inclusion ( \Delta B ) (%)</td>
<td>0.003</td>
<td>0</td>
<td>0.05</td>
<td>-</td>
</tr>
<tr>
<td>Inclusion ( \Delta C ) (%)</td>
<td>0.008</td>
<td>0</td>
<td>0.06</td>
<td>-</td>
</tr>
<tr>
<td>Reduction ratio</td>
<td>977</td>
<td>289</td>
<td>5530</td>
<td>-</td>
</tr>
<tr>
<td>Normalizing ( T\log(t) )</td>
<td>1681</td>
<td>1622</td>
<td>1733</td>
<td>-</td>
</tr>
<tr>
<td>Quenching ( T\log(t) )</td>
<td>1610</td>
<td>0</td>
<td>1681</td>
<td>890.9</td>
</tr>
<tr>
<td>Tempering ( T\log(t) )</td>
<td>1503</td>
<td>0</td>
<td>1695</td>
<td>-862.8</td>
</tr>
<tr>
<td>Fatigue strength (MPa)</td>
<td>499.9</td>
<td>225</td>
<td>906</td>
<td>-</td>
</tr>
</tbody>
</table>
RMSE was recorded as cross validation error (CVE). The cross validation was performed for all \(2^{21} - 1\) combinations of explanatory variables.

### 2.5 Neural network model

#### 2.5.1 Architecture

Artificial neural network is one way of machine learning which is inspired by biological neural networks. The architecture of artificial neural network used is depicted in Fig. 2(a). In each unit of hidden and output layers, calculation was conducted as following equation and the result was sent to next layer.

\[
h_j = \phi \left( \sum_k w_{jk} x_k + b_j \right),
\]

where \(x_k\) is the input variable from \(k\)th unit in input layer, \(w_{jk}\) and \(b_j\) are weight and bias, \(h_j\) is the output variable, and \(\phi\) is activation function. Changing the training algorithms, data normalization method, activation function and number of hidden layers (one or two layers), the prediction accuracy in the test dataset was compared. The number of units in each layer was fixed to be ten units on the first hidden layer and five units on the second hidden layer.

#### 2.5.2 Activation function

In the output layer, identity function \(\phi(x) = x\) was used as activation function. In hidden layers, three kinds of non-linear function were used as shown in the following equations and Fig. 2(b):

\[
\phi(x) = \frac{1}{1 + e^{-x}},
\]

\[
\phi(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}},
\]

\[
\phi(x) = \max(0, x).
\]

Equation (8) represents sigmoid function. It was commonly used early in neural network research because it is simple and easy to differentiate. Equation (9) is a hyperbolic tangent. Glorot and Bengio\(^{22}\) showed that the hyperbolic tangent is more suitable for neural network than the sigmoid function. Equation (14) is a ramp function. In neural network research, this function is called as "Rectified Linear Unit" or "ReLU."

Glorot and Bengio\(^{23}\) showed that ReLU is better than the hyperbolic tangent. LeCun \textit{et al.}\(^{24}\) reported that ReLU is the best activation function as of May 2015.

#### 2.5.3 Data normalization

When using sigmoid function or hyperbolic tangent as the activation function, output value converges to \(\pm 1\) or 0 as the input value moves far from zero. It means that the significance of input variable is almost eliminated. To solve this problem, the input and output variables should be normalized. In this research, three methods of normalization were examined. The first normalization method (hereinafter called "n1") is to normalize so that the average becomes zero and the variance becomes one as follows:

\[
x_i' = \frac{x_i - \bar{x}}{\sigma},
\]

where \(x_i'\) is the normalized variable, \(\bar{x}\) is the mean value and \(\sigma\) is standard deviation. LeCun \textit{et al.}\(^{25}\) suggested that this method was the best for neural network. The second and third method (hereinafter called "n2" and "n3") are to normalize so that the range of \(x_i'\) becomes \([0, 1]\) and \([0.1, 0.9]\), respectively:

\[
x_i' = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}},
\]

\[
x_i' = 0.1 + (0.9 - 0.1) \times \frac{x_i - x_{\min}}{x_{\max} - x_{\min}},
\]

where \(x_{\max}\) and \(x_{\min}\) are maximum and minimum value of each explanatory variable. Sobhani \textit{et al.}\(^{26}\) used eq. (13) for the neural network model.

#### 2.5.4 Training algorithm

Training of neural network is to properly set weights and biases. The backpropagation algorithm is one of the most common training methods. At the beginning of training
process, random values are assigned to weights and biases. The output value is calculated by using these initial weights and biases. Comparing the output value with experimental value, the bias of the output layer and the weight from the hidden layer are updated. Then, comparing the value of the hidden unit with the previous value, the bias of the hidden layer and the weight from the former layer are updated. In a similar manner, all weights and biases are updated. One cycle that updates weights and thresholds in all layers is called one epoch. Thus, the error in the output layer propagates to the input layer through the network, and hence this method is called as backpropagation.

The goal of the backpropagation is to minimize the error. In general, the error is the sum of squared errors between the predicted value and the actual value. To minimize the sum of squares, many algorithms have been proposed. The gradient descent method and the Gauss-Newton method are popular algorithm to solve this problem. The gradient descent method requires a long time to converge and the Gauss-Newton method converges quickly but can give inaccurate results. Consequently, the Levenberg-Marquardt (LM) method which is a combination of the gradient descent method and the Gauss-Newton method was adopted. Training was stopped when either the average error on the training datasets was less than a preset value, Marquardt parameter exceeded 1.0 × 1010, or the number of epochs exceeded 5000. The dataset was randomly divided into 70% for training, 15% for validation and 15% for testing when using LM method.

To prevent over-fitting, Bayesian framework has been proposed for the training in neural network. In the framework, the goal of training is to minimize the function $E(w)$ shown below, instead of sum squared error,

$$E(w) = \beta E_D(w) + \alpha E_w(w), \quad (14)$$

$$E_D(w) = \frac{1}{2} \sum_{i=1}^{n} (y_i - o_i)^2, \quad (15)$$

$$E_w(w) = \frac{1}{2} \sum_{i} w_i^2, \quad (16)$$

where $y_i$ is predicted value, $o_i$ is observed value (actual value), $w$ is vector of weights and biases. $E_D(w)$ is an error function similar to the total squared error in LM method. $E_w(w)$ is called a penalty term, which encourages to lower the weights and suppresses over learning. The calculation was performed based on LM method. The coefficients $\alpha$ and $\beta$ are hyper parameters to control the complexity of model. If $\beta$ is too large, the degree of freedom of the model becomes large and over-fitting occurs. Conversely, if $\alpha$ is too large, the weights and biases are not fitted to the data. To address this problem, the two parameters were statistically determined by Bayes’ theorem. The Bayes’ theorem is represented as follows:

$$p(w|D) = p(D|w)p(w) = p(D|w)p(w), \quad (17)$$

where $D$ is data set. When $w$ is the most probable, $p(D|w)$ becomes maximum. $p(w|D)$ is in proportion to two probabilities:

$$p(w|D) \propto p(D|w)p(w). \quad (18)$$

Assuming that these two probabilities are normal distributions,

$$P(D|w) = \prod_{k=1}^{n} p(o_k|x_k, w) = \frac{1}{Z_D} \exp \left\{ - \frac{1}{2 \sigma_w^2} \sum_{k=1}^{n} (y_k - o_k)^2 \right\}, \quad (19)$$

$$P(w) = \frac{1}{Z_w} \exp \left\{ - \frac{1}{2 \sigma_w^2} \|w\|^2 \right\} = \frac{1}{Z_w} \exp \left\{ - \frac{1}{\sigma_w^2} E_w \right\}, \quad (20)$$

$$p(w|D) \propto \frac{1}{Z_D Z_w} \exp \left\{ - \left( \frac{1}{\sigma_e^2} E_D + \frac{1}{\sigma_w^2} E_w \right) \right\}, \quad (21)$$

where $Z_D$ and $Z_w$ are constants for normalization, $\sigma_w$ is the standard deviation of actual output values, and $\sigma_e$ is the standard deviation of weights and biases. Maximizing $p(w|D)$ is equivalent to minimizing $(E_D/\sigma_e^2 + E_w/\sigma_w^2)$. Comparing with eq. (14), two coefficients $\alpha$ and $\beta$ are determined from the standard deviations. According to MacKay, there is no need to prepare validation data in the Bayesian framework. Therefore, the data set were chosen randomly for 85% for training and 15% for testing. These two training algorithm, LM method and LM method with Bayesian framework, were compared.

### 2.5.5 Evaluation

Training conditions are summarized in Table 3. The training algorithm was evaluated with model 1 and 2 in the table. All nine combinations of the three types of activation function and the three normalization method described in previous sections were compared using model 3 to 11. Since the result varied depending on the data dividing, ten training datasets were prepared by randomly dividing the entire data ten times. Each dataset was named as dataset number 1 to 10. In order to further improve the prediction accuracy, a second hidden layer was added to the selected neural network model (model 12). In each model, epoch number, correlation coefficient between actual value and predicted value of training data, validation data, test data and all data were recorded.

### Table 3 Training conditions for neural network models.

<table>
<thead>
<tr>
<th>Model number</th>
<th>Network architecture</th>
<th>Training algorithm</th>
<th>Activation function</th>
<th>Normalization method</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>H1</td>
<td>LM</td>
<td>Tanh</td>
<td>n1</td>
</tr>
<tr>
<td>2</td>
<td>H1</td>
<td>LM+Bayes</td>
<td>Tanh</td>
<td>n1</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>Sigmoid</td>
<td>n2</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td>n3</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td>n1</td>
</tr>
<tr>
<td>7</td>
<td>H1</td>
<td>LM+Bayes</td>
<td>Tanh</td>
<td>n2</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td>n3</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td>RelU</td>
<td>n2</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td>n3</td>
</tr>
<tr>
<td>12</td>
<td>H2</td>
<td>LM+Bayes</td>
<td>RelU</td>
<td>n1</td>
</tr>
</tbody>
</table>
2.5.6 Sensitivity analysis

Sensitivity analysis (SA) is to evaluate the contribution of input uncertainty to output uncertainty of model. The sensitivity is divided into local sensitivity and global sensitivity. The local sensitivity refers to the sensitivity at a fixed point in the parameter space, while global sensitivity refers to an integrated sensitivity over the entire input parameter space. In the linear regression model, the coefficient directly represents the local and global sensitivity, as shown in the eq. (6). In contrast to the linear regression model, weights and biases in neural network model have complicated influences on the sensitivity. The local sensitivity can be evaluated by changing only one of the explanatory variables and fixing remaining variables. Several researchers call the local SA in neural network model as “virtual experiment.” However, it does not represent the overall significance of the explanatory variables over the entire data. To evaluate the significance on fatigue strength over a wider range of materials, global SA is required. Using the created neural network model, global SA was performed by combined weight method, and fourier amplitude sensitivity test (FAST) method and Sobol’ method. Also, the local SA was conducted to examine the effect of carbon and manganese contents on the fatigue strength.

3. Results and Discussions

3.1 Hierarchical clustering

Total entropy of five clusters is shown in Fig. 3. Among the three intra-cluster distances of Euclidean, standardized Euclidean, and Mahalanobis distance, five metallic materials were successfully clustered with Euclidean distances. Of the five inter-cluster distances, the shortest distance method, the group average method, and the center of gravity method provided successful clustering. In the centroid method, the dendrogram partially reversed and intersected. This is because the centroid of cluster moved due to cluster merging and the distance between clusters does not monotonically decrease. Based on these results, the shortest distance method with Euclidean distance was used in the subsequent clustering.

3.2 Linear regression model

Result of variable selection with cross validation method is shown in Fig. 5. The horizontal axis shows the number of explanatory variables, $p$, and the vertical axis shows the minimum CVE for each number of explanatory variables. It can be seen that CVE reached a minimum at $p = 14$. Selected variables and calculated coefficients are shown in Table 2. Variables of C, Si, Mn, P, Ni, Cr, Cu, Mo, Al, N, Ti, O, quenching, tempering were selected. All of these variables are selected more than 85 times in the top 100 combinations. The normalized coefficients showed that two heat treatments have great significance on the fatigue strength. Also, the result showed that fatigue strength increases by adding elements of C, Mn and Cu, which is consistent with our

### Figure 3

Effect of distance definition on total entropy in hierarchical clustering of metallic materials.

### Figure 4

Results of hierarchical clustering of 630 iron alloys.

### Figure 5

Variable selection by cross validation method with linear regression model.
accurate prediction for wider range of materials than RMSE are shown in Table 4. The model provided more

\[ C \]

where

\[ d \]

contains 14 alloying elements and 2 tempering parameters.

hardness of the tempered martensite. The Mukherjee
derived the empirical model for predicting the Vickers

experimental fatigue strength. Correlation coe

hardening. Figure 6 shows the prediction result with

substitutional solid solution strengthening and precipitation

existing knowledge of solid solution strengthening,
substitutional solid solution strengthening and precipitation

hardening. Figure 6 shows the prediction result with

experimental fatigue strength. Correlation coefficient and

RMSE are shown in Table 4. The model provided more

accurate prediction for wider range of materials than

previously proposed linear regression model.\(^{13}\) These results

confirmed that variable selection is necessary for highly

accurate prediction of fatigue strength.

Thus, the variable combination giving the minimum CVE

was found by exhaustive search for all combinations. This method is also effective for comparing with existing

empirical rules. Igarashi \textit{et al.} proposed to plot the density

of state of CVE and compare various solutions of variable

selection on the density of state.\(^{38}\) They called the method as

exhaustive search with density of state (ES-DoS). The state

density for CVE of fatigue strength was plotted in Fig. 7. The solute

carbon content and the prior austenite grain size were ignored

in the mapping. The position of Umemoto’s model suggested a

possibility that the grain size and solute carbon content not

present in database are important on the fatigue prediction. Agrawal’s model with 21 explanatory variables showed

relatively high accuracy. However, compared with the best

model with 14 explanatory variables, the accuracy was

much lower. These results also demonstrated the necessity of

variable selection.

3.3 Neural network model

In the prediction result of model 1 and 2, the epoch number

was 17 and 454, the correlation coefficient (R) of the
test dataset was \( R = 0.9798 \) and 0.9864, respectively. The
training time of LM method with Bayesian inference (model 2) was about five seconds using PC with 2.3 GHz
octa cores and 16 GB memory. It was longer than the simple
LM method (model 1). From comparison of correlation
coefficients for the test dataset, it was confirmed that
prediction accuracy improves by adding Bayesian inference
to LM method. Therefore, in the following neural network
models, LM method with Bayesian inference was used as the training algorithm.

Figure 8 shows correlation coefficient of test dataset for

neural network models created by three kinds of activation
functions and three kinds of normalization methods. The
normalization method is different for each of the three plots.
The horizontal axis represents the dataset number. Overall,
the correlation coefficient scattered depending on the test
dataset. The method of n1–ReLU, n2–Tanh, n3–Tanh showed
relatively high correlation coefficients, among which n1–ReLU
has the smallest variance of the correlation coefficient.
Moreover, in the method of n1–ReLU, the training finished at
the smallest epoch number among three activation function.
These results agreed with the previous studies. In the
normalization method of n2 and n3, the variables are converted so as to fit within a certain range, whereas the

\[ \sigma_{TS} = 1301C + 1089C_s - 38.2Mn + 3.90Ni - 0.124d_p - 17.5I + 1008, \] (22)

where \( C \) is solute carbon content, \( d_p \) is prior austenite grain size (µm) and \( I \) is tempering parameter. Mukherjee \textit{et al.}\(^{9}\) derived the empirical model for predicting the Vickers

hardness of the tempered martensite. The Mukherjee’s model contains 14 alloying elements and 2 tempering parameters.

\[ \text{Agrawal et al.}^{13} \] derived a linear regression model of fatigue

strength using all the features of NIMS fatigue data sheet. Assuming a linear relationship between fatigue strength,
tensile strength and hardness, these existing models were

mapped on state of density of CVE (Fig. 7). The solute

carbon content and the prior austenite grain size were ignored in the mapping. The position of Umemoto’s model suggested a

possibility that the grain size and solute carbon content not

present in database are important on the fatigue prediction.

Agrawal’s model with 21 explanatory variables showed

relatively high accuracy. However, compared with the best

model with 14 explanatory variables, the accuracy was

much lower. These results also demonstrated the necessity of

variable selection.

Table 4 Correlation coefficient and root mean squared error (RMSE) of multivariate linear regression model (MR) and artificial neural network model (ANN).

<table>
<thead>
<tr>
<th>Model and dataset</th>
<th>Correlation coefficient</th>
<th>Root mean squared error (RMSE)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MR training</td>
<td>0.973</td>
<td>22.8</td>
</tr>
<tr>
<td>MR test</td>
<td>0.968</td>
<td>23.0</td>
</tr>
<tr>
<td>ANN training</td>
<td>0.991</td>
<td>13.4</td>
</tr>
<tr>
<td>ANN test</td>
<td>0.992</td>
<td>16.4</td>
</tr>
</tbody>
</table>
normalization method of n1 converts variables so that the average becomes zero. LeCun et al. suggested that if the average of the inputs is kept away from zero, the updating of the weight is biased toward a specific direction and the training takes long time. The ReLU function returns zero to the negative input value, and when combined with the normalization method of n1, the output of about half units in the hidden layer becomes zero. According to Glorot et al., the ReLU function enables sparse representation to invalidate a part of the hidden layer, and suppresses over-fitting by lowering the degree of freedom of the model.

To further improve the prediction accuracy, a second hidden layer was added to the neural network model (model 12). The prediction results of a neural network model consisting of two hidden layers and n1–ReLU method is shown in Fig. 9. In the models with single hidden layer (model 9) and double hidden layers (model 12), the epoch number was 112 and 135, the correlation coefficient (R) of the test dataset was R = 0.9886 and 0.9916, respectively. By adding the hidden layer the computation time increased slightly. It should be noted that as the number of hidden layers increases, an activation function is applied many times in the network, the gradient may disappear, and updating of the weight may not succeed. In the case of using the sigmoid function or Tanh, this gradient elimination problem can occur.

The results of global SA using combined weight method, FAST method and Sobol’ method are shown in Fig. 10. In all three methods, it was shown that quenching and tempering conditions are significant on the fatigue strength. In the combined weight method, not much difference was observed between the sensitivities of alloying elements and inclusions. Whereas FAST method and Sobol’ method showed that the alloying elements of Cu and Mo, and inclusion parameter dB has larger influence on fatigue strength. The dB represents an area fraction of inclusions occurring in discontinuous arrays (Alumina, etc.).

To examine the effect of carbon and manganese contents on the fatigue strength, local SA (virtual experiment) was conducted by changing C and Mn and fixing the remaining variables to the average value. As shown in Fig. 11, the predicted fatigue strength increased with C and Mn. This result was consistent with known physical phenomena that carbon and manganese exist as interstitial solid solution and substitutional solid solution in steels, respectively, and improve the strength of steels.

Only the first terms of alloying elements were used in this paper. There is a possibility that the prediction accuracy is further improved by adding lower and higher-order terms of C, and the cross terms between C and carbide-forming elements such as Cr and Mo into the explanatory variables. Also, it is one of promising approaches to select variables by changing combinations of explanatory variables in the neural network models. Since it is difficult to calculate all combinations of 20 or more explanatory variables with neural
network models from the viewpoint of computational cost, it may be effective to firstly select variables less than 20 with simple linear regression models and then select the variables in the neural network models.

4. Conclusion

The fatigue dataset provided by NIMS fatigue data sheet was automatically classified by hierarchical clustering method. The shortest distance method with Euclidean distance was suitable for categorizing metallic materials. A group of carbon steels was selected for linear regression and neural network analyses. In linear regression analyses, a model selection was conducted from all possible combinations of explanatory variables, and CVE reached a minimum when the number of variables is 14. It showed that two heat treatments of quenching and tempering have great significance on the fatigue strength. The frequency distribution of CVE in reference to ES-DoS method provided an overview of the relationship between the derived linear regression model and existing empirical rules. In neural network models, the model with double hidden layers, Bayesian framework, ReLU activation function and n1 normalization method provided the best prediction among 12 models. Using the neural network model, local and global SA was performed and the results were consistent with existing knowledge in materials engineering. There is a possibility that the prediction accuracy is further improved by adding lower and higher-order terms and the cross terms into the explanatory variables. These results demonstrated that the machine learning method provides prediction of fatigue performance with high accuracy and is one of promising method to accelerate material development.

Acknowledgments

This work was partially supported by Council for Science, Technology and Innovation (CSTI), Cross-ministerial Strategic Innovation Promotion Program (SIP), “Structural Materials for Innovation” (Funding agency: JST), Japan, and Society for the Promotion of Science KAKENHI (Grant Number 17K14832). The fatigue database used in this study was provided by National Institute for Materials Science (NIMS).

REFERENCES

15) M. Steinbach, G. Karypis and V. Kumar: KDD Workshop on Text
16) P.C. Mahalanobis: Proceedings of the National Institute of Sciences
    1409–1438.
21) C.M. Bishop: Pattern Recognition and Machine Learning, (Springer-
22) X. Glorot and Y. Bengio: Proceedings of the Thirteenth International
    Conference on Artificial Intelligence and Statistics, Proceedings of
23) X. Glorot, A. Bordes and Y. Bengio: Proceedings of the Fourteenth
    International Conference on Artificial Intelligence and Statistics,
25) Y. LeCun, L. Bottou, G.B. Orr and K.R. Müller: Neural Networks -
    Tricks of the Trade, vol. 1524, ed. by G. Orr and K. Müller, (Springer
    1373–1382.
31) M. Koda and T. Honma: Operations Research as a Management
34) R.I. Cukier, C.M. Fortuin, K.E. Shuler, A.G. Petschek and J.H.
    1140–1149.
38) Y. Igarashi, H. Takenaka, Y. Nakanishi-Ohno, M. Uemura, S. Ikeda and