

Neural networks with dimensionality reduction for predicting temperature change due to plastic deformation in a cold rolling simulation

Chun Kit Jeffery Hou  and Kamran Behdinan

Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, ON, Canada

Research Article

Cite this article: Hou CKJ, Behdinan K (2023). Neural networks with dimensionality reduction for predicting temperature change due to plastic deformation in a cold rolling simulation. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **37**, e1, 1–13. <https://doi.org/10.1017/S0890060422000233>

Received: 22 October 2021
Revised: 8 August 2022
Accepted: 16 November 2022

Key words:

Artificial neural networks; cold rolling; dimensionality reduction; finite element analysis; machine learning

Author for correspondence:

Chun Kit Jeffery Hou,
E-mail: jhou@mie.utoronto.ca

Abstract

Cold rolling involves large deformation of the workpiece leading to temperature increase due to plastic deformation. This process is highly nonlinear and leads to large computation times to fully model the process. This paper describes the use of dimension-reduced neural networks (DR-NNs) for predicting temperature changes due to plastic deformation in a two-stage cold rolling process. The main objective of these models is to reduce computational demand, error, and uncertainty in predictions. Material properties, feed velocity, sheet dimensions, and friction models are introduced as inputs for the dimensionality reduction. Different linear and nonlinear dimensionality reduction methods reduce the input space to a smaller set of principal components. The principal components are fed as inputs to the neural networks for predicting the output temperature change. The DR-NNs are compared against a standalone neural network and show improvements in terms of lower computational time and prediction uncertainty.

Introduction

Cold rolling is a sheet metal forming process used to strengthen material properties through plastic deformation of the microstructure. This procedure produces parts with good dimensional accuracy and clean surface finishes. Typically, cold rolling is limited to compression ratios up to 50%. Thus, to accumulate high plastic strain in cold rolling, multiple passes are required to compress the thickness of the workpiece rather than performing it in one single, large compression stage. The plastic deformation and friction between the workpiece and rollers cause the workpiece temperatures to rise. This can lead to deterioration of the rolls or softening of the workpiece. Pietrzyk and Lenard (1991) investigated the heating of aluminum strips during deformation in cold rolling. Their analytical model accurately predicted the final temperatures with an error of below 1.5°C compared with experimental results. Other authors have extended their work to develop thermo-mechanical models for measuring temperatures in rolling processes (Chang, 1998; Luo, 1998).

Numerical methods have been used to analyze parameters' effects (Hossain *et al.*, 2016) and investigate phenomenon that would otherwise be difficult to build experiments for (Kostiv *et al.*, 2012). In the context of cold rolling, Khan *et al.* (2004) created a coupled thermo-mechanical heat transfer model based on the finite volume method. The study proved the impact of roll speed and heat transfer on heat flow distribution in both the roll and workpiece. Yadav *et al.* (2014) established an efficient FEM method for estimating temperature distributions in roll and strip in the order of seconds, which could be applied to both cold and hot rolling. Simplified mathematical models for heat transfer analysis were developed and validated against experimental results.

FEM has also been used for evaluating other important phenomena in cold rolling. Hwu and Lenard (1988) developed a finite element solution for evaluating the effects of work-roll deformation and friction on resulting strain fields in flat rolling. Liu *et al.* (1985) developed an elastic-plastic FEM for cold rolling of copper strips to determine stress and deformation outside the nominal contact. More recently, Mehrabi *et al.* (2015) studied chattering vibration in cold rolling using an implicit FEM method. From validating their FEM with experimental results, they determined the effects of friction and reduction on chatter and provided new design considerations in future rolling processes. FEMs are powerful tools for conducting engineering analyses and obtaining data on experiments which would otherwise be expensive to conduct. However, solving nonlinear problems is time-consuming and predictions become more difficult to obtain with higher complexity models. Artificial neural networks are a viable solution to these computational demands, which have not been explored for cold rolling simulations involving temperature rise due to plastic deformation.

Data mining has become a rapidly growing field in recent years due to a rise in data generation volume. This has led to a growth in size and complexity in data processing which

© The Author(s), 2023. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited.



continues to be a growing problem when efficiency and computation speed is important. High-dimensional data exists where the number of data features is on the order of the number of samples or observations (Narisetty, 2020). These datasets can be computationally expensive to generate mapping functions between input and output. Thus, reducing the number of features, or the input dimensionality, can greatly simplify the learning and accelerate the training phase of regression and classification models for identifying patterns between features and output. Dimensionality reduction has been used extensively as a feature extraction tool for surrogate models, such as artificial neural networks (ANNs) and finite element analysis (FEA) research. Kashid and Kumar (2012) reviewed the current state-of-the-art applications of ANNs to sheet metal work. They critically reviewed the use of ANNs for reducing computation time and on bending operations. Their conclusions discuss the drawbacks on backpropagation in ANNs such as long training times and large data requirements in several works. Shahani *et al.* (2009) used FEM with ANNs to predict influential parameters in a 2D aluminum hot rolling process. The study used an updated Lagrangian method formulation to solve a thermo-mechanical analysis and determine the effects of process parameters. ANNs have also been optimized for predicting springback in sheet metal forming (Liu *et al.*, 2007; Ruan *et al.*, 2008; Spathopoulos and Stavroulakis, 2020). For instance, Spathopoulos *et al.* (Spathopoulos and Stavroulakis, 2020) used a Bayesian regularized backpropagation neural network for predicting springback in the S-Rail metal forming process. In cold rolling, ANNs have been explored for predicting new mechanical properties (Ghaisari *et al.*, 2012) such as yield and ultimate tensile strength, degree of void closure (Chen *et al.*, 2011), defect detection (Yazdchi and Mahyari, 2008), and rolling force (Lin, 2002). However, dimensionality reduction has not been applied to ANNs for improving the computation time and training of these surrogate models in cold rolling.

In terms of combining dimensionality reduction with neural networks, several authors have investigated these integrated methods for reducing computational cost of designing reduced-order models. Higdon *et al.* (2008) applied PCA for non-probabilistic, dimensionality reduction in Gaussian processes for uncertainty quantification of material properties. PCA has also improved optimization of surrogate models in calibration (Kamali *et al.*, 2007), aerodynamics (Kapsoulis *et al.*, 2018; Tao *et al.*, 2020), and air pollution (Olvera *et al.*, 2012). Straus *et al.* (Straus and Skogestad, 2017) developed extended surrogate model fitting methods by incorporating PLS regression for reducing the number of independent variables in an ammonia process case study. The nonlinear surrogate models were fitted to the PLS latent variables and improved the surrogate model fit by a factor of two. Finally, PCA's applicability is limited to linearly separable datasets and projections onto a linear subspace. Thus, kernel PCA (kPCA)-based surrogate models have been used to reduce complexity in high-dimensionality datasets such as water quality forecasting (Zhang *et al.*, 2020) and intrusion detection (Hongwei and Liang, 2008). However, DR-NNs have yet to be explored in prediction for cold rolling and other sheet metal applications.

In this work, we investigated the use of dimensionality reduction methods such as principal component analysis (PCA), partial least squares (PLS), and kPCA with neural network surrogate models for reducing computation times and uncertainty in predicting the change in temperature due to plastic deformation in a two-stage cold rolling FEM simulation. The paper is organized in the following manner. Section "Dimensionality Reduction"

begins with a mathematical summary of the explored dimensionality reduction methods and neural network used and describes the general structure of the DR-NNs. Next, Section "Test Case: Finite Element Model" describes the FEA two-stage cold rolling test case used to evaluate the performance of our DR-NNs against standard neural networks. Section "Results and Discussion" shows the prediction error and computation time results of the DR-NNs and neural network and discusses the results and usefulness of dimensionality reduction for neural networks, as well as the potential for future research. Finally, we conclude the work and summarize our key findings.

Dimensionality reduction

Principal component analysis

PCA is the most popular unsupervised learning method for linear dimensionality reduction. This method has been applied to a variety of engineering simulation problems to reduce complexity, ease computations, and extract key information for simplified reconstruction of datasets with quantitative variables. The general procedure for PCA is shown in Figure 1 (Vimala devi, 2009; Jolliffe and Cadima, 2016).

If we consider a dataset with n input samples with k features, excluding the output labels, PCA first normalizes the dataset for comparable values across features. We normalize the dataset to prevent features with larger magnitudes from skewing the variance calculated by the PCA procedure. The goal of PCA is to identify the linear combinations of the columns in matrix X which result in maximum variance. These linear combinations can be expressed as shown in Eq. (1), where c is a vector of constants with vectors x .

$$\sum_{j=1}^n c_j x_j = Xc. \quad (1)$$

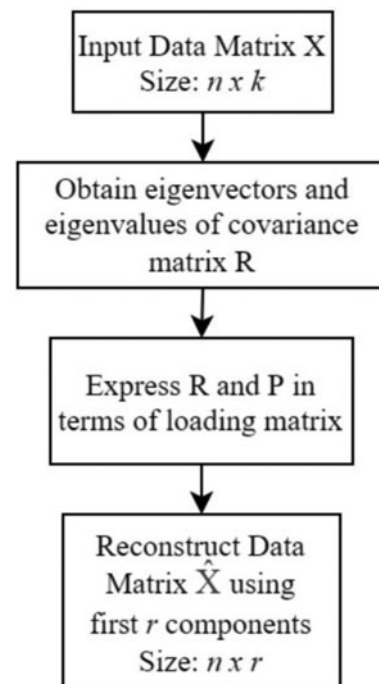


Fig. 1. PCA procedure.

The variance of the linear combinations can be obtained as a function the sample covariance matrix R . We can further express this variance in terms of a Lagrange multiplier, expressed as an eigenvalue, where the goal is to maximize the variance. Any constant c multiplied by its transpose equals one. This defines c as a set of unit vectors. Altogether, the simplified relation between the variance of the dataset and eigenvalue is shown in Eq. (2).

$$\text{var}(Xc) = cRc^T = \Lambda c^T c = \Lambda. \tag{2}$$

The covariance matrix, R , is computed for the input data and singular value decomposition is used to produce a diagonal matrix of principal component eigenvalues, Λ , and associated loading vectors, C , as shown in Eq. (3).

$$R = C\Lambda C^T. \tag{3}$$

The elements of the covariance matrix indicate the correlation of the component with the corresponding input variables. The eigenvalues are proportional to the level of variance captured by the associated normalized principal component's eigenvector. Thus, the resulting principal components are linear expressions of the initial input data and are independent from one another. Taking advantage of the diagonal properties of the eigenvector matrix, R can be rewritten in terms of the principal component loading matrix, L , with dimensionality $r \times k$, where $r < k$, as shown in Eq. (4).

$$R = LL^T. \tag{4}$$

The principal components P can be found by multiplying the input data matrix X with the loading matrix, as shown in Eq. (5).

$$P = XL. \tag{5}$$

The number of retained r components are selected based on captured variance of the full data. A good rule-of-thumb is to

keep enough components to retain 70%–80% of the total variance. Once the number of components is determined, the principal component matrix is multiplied with the loading matrix, which reconstructs the input features to $n \times r$ dimensions, as shown by Eq. (6).

$$\hat{X} = PL^T. \tag{6}$$

Figure 2 shows an example of the variance captured by each principal component for a typical dataset. The curve indicates the cumulative variance that is captured with each principal component, while the bars show the contributed variance by each principal component.

While the PCA procedure is useful for linear dimensionality reduction by seeking feature combinations for maximum variance, its application is limited to a linear order of simplification. For high-dimensionality datasets, nonlinear methods can produce a more insightful view of higher-order relations within the dataset structure.

Kernel principal component analysis

Linear dimensionality methods such as PCA and PLS are limited to datasets which are linearly separable. As such, nonlinear dimensionality reduction methods such as kernel PCA can be considered for inseparable dataset cases. kPCA is a classical approach for nonlinear dimensionality reduction. The method projects the linearly inseparable data into a higher-dimensional Hilbert space through a kernel function $k(x_i, x_j)$ where the PCA method is performed afterwards. The kPCA method is more commonly used for classification problems, such as face recognition. While the nonlinearity of the kPCA method makes it more computationally expensive to perform than linear PCA, the method captures nonlinearities in the dataset through transforming the data into a higher-dimensional space representation, which can ease the neural network training process thereafter. An ongoing challenge with this method is the trivial kernel selection task that is largely problem-dependent.

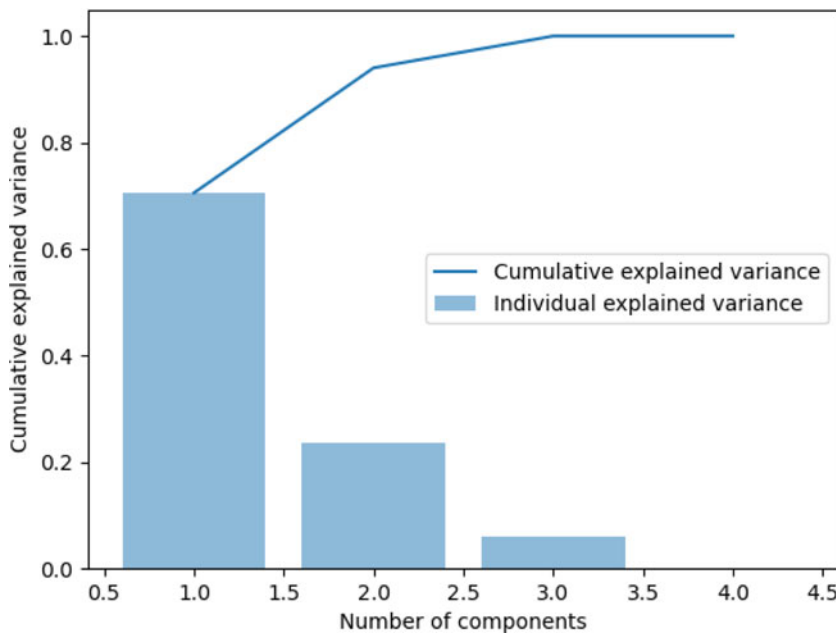


Fig. 2. Variance versus principal components. Cumulative variance line and contributed variance by each subsequent principal component as bars (Chaves et al., 2012).

Partial least squares

Unsupervised methods such as PCA perform dimensionality reduction using only the input features and ignoring the output labels. Alternative methods which consider the correlation between the input/independent variables and output/dependent variables have shown to perform better than PCA for capturing information in the reduced components (Maitra and Yan, 2008). Supervised methods also produce more suitable topology representations of input–output maps compared with unsupervised methods (Lataniotis *et al.*, 2018).

The PLS method is a supervised, linear dimensionality reduction method that has been implemented for multivariate calibration and classification problems (Wold *et al.*, 2001; Hussain and Triggs, 2010). Unlike PCA which calculates hyperplanes for maximum variance in the input features, PLS transforms the input and response/output variables to a new feature space based on maximum covariances and finds a regression mapping function relating them. PLS regression reduces the dimensionality through fitting multiple response variables into a single model. This is analogous to a multilayer perceptron, where the hidden layer nodes can be determined by the number of retained PLS components after applying the reduction (Hsiao *et al.*, 2003). Furthermore, the multivariate consideration does not assume fixed predictors, which makes PLS robust in measuring uncertainty. Equations (7) and (8) define the underlying PLS model.

$$X = AC^T + E, \quad (7)$$

$$Y = BD^T + F. \quad (8)$$

Here, X and Y are the non-reduced predictors and responses, respectively. A and B are the projections of X and Y , C and D are the respective orthogonal loading matrices, and E and F are error terms. In the PLS algorithm, the covariance of A and B is maximized when projecting X and Y into the new feature space. The number of PLS components to keep is traditionally determined based on minimum mean squared error or cross-validation (Kvalheim *et al.*, 2018).

Artificial neural networks

ANNs are multilayer algorithms, based on the biological learning process of the human brain, that perform high-complexity regression and nonlinear classification analyses. ANNs have been employed as data-fitting surrogate models for complex system output prediction (Papadopoulos *et al.*, 2018; Shahriari *et al.*, 2020).

ANNs are typically used to identify complex mapping functions between input features and outputs. Figure 3 shows the general structure of the ANN with one hidden layer. Traditional ANNs contain one input and output layer with one or more hidden layers. The number of nodes in the input N and output layers M are dependent on the number of dataset features and labels, respectively. Careful consideration must be exercised for determining the number of hidden layers and the number of nodes Q . The structure is chosen based on the complexity and size of the problem. A small number of hidden nodes causes the model to predict outputs with poor accuracy, while having too many nodes lead to overfitting the training data and poor generalization of unseen data. Good practices have been developed for

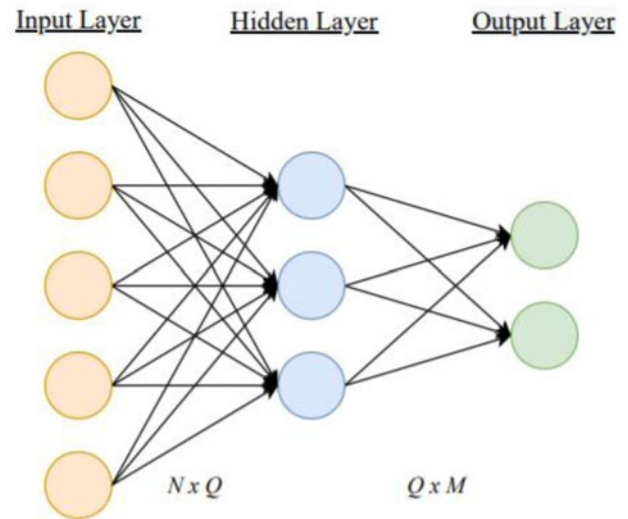


Fig. 3. Structure of an artificial neural network.

selecting the number of hidden nodes. Generally, the rule-of-thumb is for the number of hidden nodes to be between the number of input and output nodes. Trial-and-error is also an option. In this study, trial-and-error was used in this work to determine the size of the hidden layers, as the rule-of-thumb method produced inadequate predictions.

The input dataset is typically normalized prior to training the neural network for improving the learning process and convergence. For a given sample, a normalized input feature vector x is passed into the input layer of the ANN. An activation function φ transforms the i th input entry, with weights $w_{i,j}$ and biases b_j connected to the j th output node of vector u , as shown in Eq. (9). Another activation function is applied to the hidden layer vector values to generate the prediction vector y , described by Eq. (10). This is repeated for subsequent samples to generate a set of predictions.

$$u_j = \varphi_1 \left(\sum_{i=1}^N w_{i,j}^1 x_i + b_j^1 \right), \quad (9)$$

$$y_k = \varphi_2 \left(\sum_{l=1}^M w_{l,k}^1 u_l + b_k^1 \right). \quad (10)$$

The ReLU activation function, as shown in Eq. (11) and Figure 4, is commonly chosen for each layer because of its good convergence properties and reduced likelihood of reaching a vanishing gradient.

$$\varphi(z) = \max(0, x). \quad (11)$$

Finally, ANNs are trained by minimizing a cost function through a back-propagation optimization algorithm. The mean squared error (MSE) function is often chosen as the cost function in regression tasks, as shown by Eq. (12). Here, \hat{y} denotes the

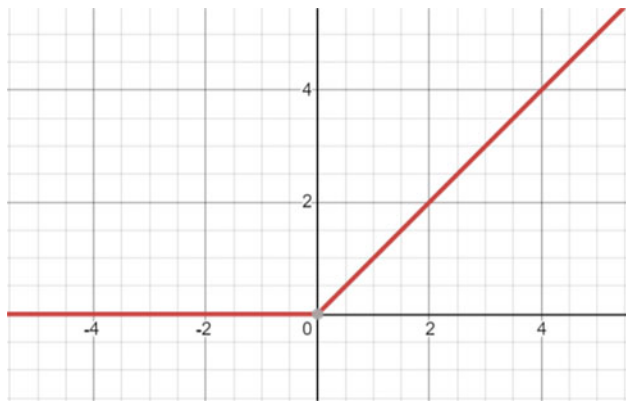


Fig. 4. ReLU activation function.

prediction from the DR-NN and y is the labeled output.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y - \hat{y})^2. \tag{12}$$

Proposed models

The algorithm architecture in this paper combined dimensionality reduction methods with neural networks for more efficient computation and improved prediction accuracy. Specifically, PCA, PLS, and kPCA-based neural networks were constructed and compared against a standalone neural network in terms of prediction accuracy, computation time, and error uncertainty.

First, the dataset was split into training and test sets. 80% of the dataset was used for training, while the remaining 20% was reserved for testing the trained DR-NNs. Next, we split the training data into initial training and validation sets. The initial training dataset was used to fit the DR-NNs, while the validation set was used to evaluate the error of the model fit as the

hyperparameters were being tuned. Finally, the test dataset was used for final evaluation of error in the trained DR-NNs.

Figure 5 shows the structure of the DR-NNs. Dimensionality reduction was applied to the raw input to reduce 16 features to a smaller set of d features. The reduced input is subsequently fed into the two layer, 64 node hidden layers to predict the final output average temperature change. The ReLU activation function was applied to each hidden layer for its good convergence behavior. For consistency, the same hidden layer structure was used in each experiment.

The DR methods reduced the raw dataset to a smaller set of features, which were fed into the neural network for training. Mini-Batch Gradient Descent was selected as the optimization method for computing functional gradients based on the ANN’s weights and biases. In this work, a batch size of 32 samples was used and the DR-NN weights and biases were optimized to minimize MSE. Specifically, the weights and biases were updated with every 32 training samples passed forward and back through the neural network. The training loop continued for a specified number of epochs, the number of times the entire training dataset is explored by the ANN, and a validation dataset was used for validating the model’s accuracy with the optimized weights and biases.

Test case: finite element model

Experimental setup

The two-stage rolling simulation was implemented in ABAQUS/CAE using an explicit solver. The system consisted of four rollers and the workpiece as shown in Figure 6. The dimensions shown are in millimeters and were constant across all trials.

The model is based on the thick bar rolling example in the ABAQUS Examples Manual v 6.12. The key differences between the model shown in Figure 6 and the benchmark example include the additional roller, additional materials, and the inclusion of adiabatic heating effects. At a room temperature of 293.15 K, the forming process was assumed to occur quickly such that the

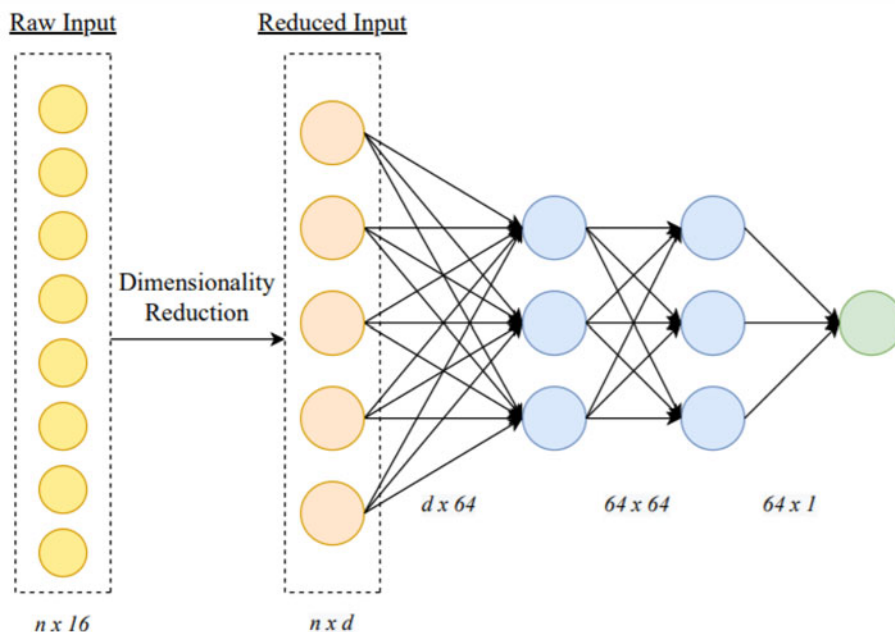


Fig. 5. DR-NN structure. Dimensionality reduction reduces raw input to a smaller set of input nodes.

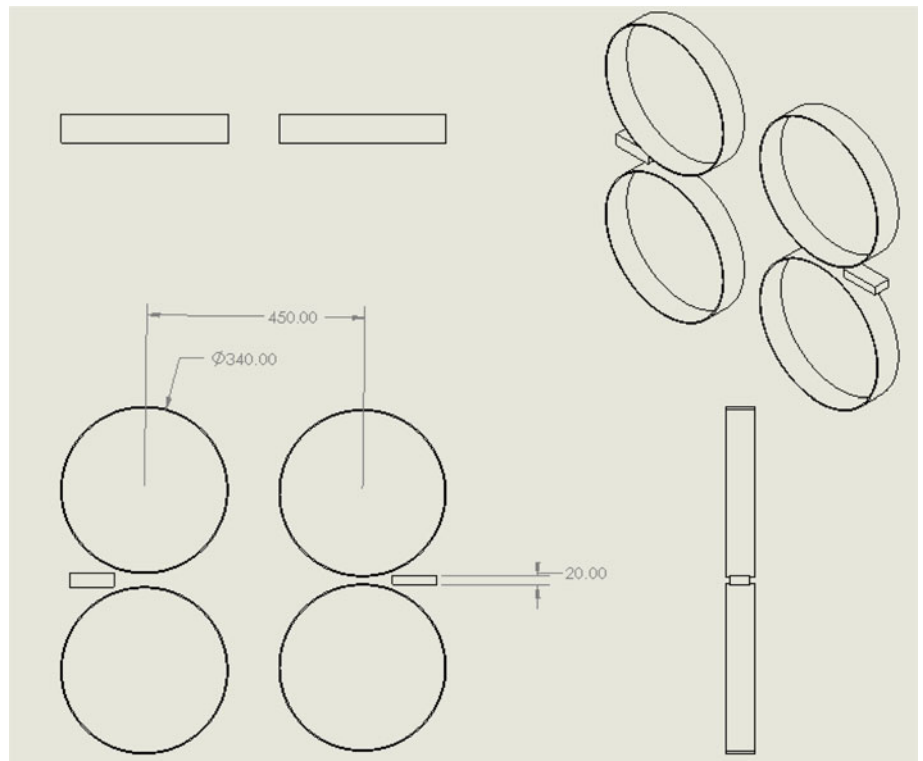


Fig. 6. Rolling simulation assembly.

generated heat does not dissipate through the material. Thus, a dynamic, explicit analysis with the inclusion of adiabatic heating effects was conducted to obtain the change in output surface temperature. An experimental displacement control was imposed by setting the second roller at a height of 20 cm from the center of the workpiece to ensure a constant thickness output. For data generation, the python scripting feature in ABAQUS was used to iterate the input variables which affect the output temperature. The dataset was shuffled for training purposes and a total of 1095 samples were acquired for training and testing the DR-NNs.

Boundary conditions

To reduce simulation times, a quarter model was developed with symmetric boundary conditions on two planes. Two 170 mm radii rollers compress the workpiece to a thickness of 20 mm. The rotation speed of each roller is equal to the feed velocity to avoid slippage in each trial.

Material properties and design variables

Four materials were tested altogether with the material properties shown in Table 1. Table 2 shows the range of values taken by the variable inputs of the experiment. The default inelastic heat fraction value of 0.2 was used in all simulations. Steel and aluminum were chosen as their friction properties in cold rolling are well studied in the literature. The true stress versus true strain plasticity data were obtained from experiments for each material from Kopp and Dohmen (1990), Alves *et al.* (2011), Deng *et al.* (2013), and Johnston *et al.* (2016)

A total of 700 sample points were used for training the DR-NNs and 176 samples were used for training validation. The first roller's height range was chosen based on critical reduction ratios in cold rolling to avoid slipping. Friction in cold rolling has been shown to vary with material, feed velocity, and compression ratio (Lenard, 2014). Friction in cold rolling tends to decrease with feed velocity due to the time-dependent nature of adhesion bond formation. However, with sufficient roll surface roughness,

Table 1. Material properties

Material property	Steel 572 (Kopp and Dohmen, 1990)	Al 2011 (Alves <i>et al.</i> , 2011)	Steel 1018 (Johnston <i>et al.</i> , 2016)	Al 1050 (Deng <i>et al.</i> , 2013)
Young's modulus (GPa)	200	68.3	205	73
Conductivity (W/m K)	51	235	51.9	157
Yield stress (MPa)	160.72	150	241.3	60
Specific heat (J/kg K)	490	910	486	901
Density (g/cm ³)	7.8	2.7	7.87	2.79

Table 2. Variables with range of values in the dataset

Variable	Range
Workpiece width	37–42 mm
Workpiece height	34–42.4 mm
Feed velocity	0.95–1.34 m/s
Roller 1 height	6.8–8.8 mm

friction increases with feed velocity in aluminum cold rolling (Lenard, 2004). In this study, we have assumed the rollers have high roughness due to the absence of lubrication in the FEA model. In general, the friction coefficient tends to decrease with reduction in low-carbon steels, regardless of lubrication. However, this behavior changes with softer materials because of reduced normal pressure during contact. Lim and Lenard (1984) studied the effects of reduction ratio on friction coefficients in aluminum alloys. Their results showed that the friction coefficient increases with reduction due to a faster rate of real contact area and rate of asperity flattening.

An approximate friction model based on the combined findings in Lim and Lenard (1984) and Lenard (2004, 2014) was developed and shown in Eqs (13) and (14). Here, v denotes the feed velocity and r is the reduction ratio. The constants were determined based on best-fit curves in Microsoft Excel for Al 6061 and low-carbon steel experiments. This study assumed that steel 572 and 1018 follow the same friction model, as did Al 1050 and 2011. Furthermore, the models are only valid for steel with reduction ratios greater than 10%. Further research can be conducted to relax this assumption.

For steels (Lim and Lenard, 1984; Lenard, 2004, 2014):

$$\mu = -\frac{0.1}{1300}v + 0.0866(r)^{-0.727} + 0.082. \quad (13)$$

For aluminum (Lim and Lenard, 1984; Lenard, 2004, 2014):

$$\mu = \frac{0.02}{500}v + 0.4r + 0.13. \quad (14)$$

The friction models were applied as explicit general contacts in the ABAQUS rolling simulation with penalty-based properties.

Overall, 16 input variables were identified: 3 geometric, 1 velocity, 2 friction measures at each roller, and 10 material properties with plasticity data found in Kopp and Dohmen (1990), Alves *et al.* (2011), Deng *et al.* (2013), and Johnston *et al.* (2016). Other variables such as roll surface roughness and strain rate were not included in this study.

Measurement methods and meshing

Figure 7 shows the typical output change in temperature distribution of the quarter-model workpiece after passing through both stages of the roller system.

A total of 1280 C3D8RT elements were used in the workpiece mesh and a mass scaling factor of 2758.5 was used for stabilizing the time increments. This value was chosen based on information provided in the ABAQUS Examples Manual. For this research, only the results of the central elements of the top surface were used in validating the ML models. In ABAQUS, eight elements were selected near the center of the workpiece as shown in Figure 8.

The temperatures of these elements were averaged to a single value for prediction. These surface elements were chosen as measurement points because friction is prominent at the surface, the regional temperatures are similar, and mesh refinement does not significantly affect the results. We demonstrated the latter by comparing Figure 8 with a refined mesh shown in Figure 9. The difference in output temperature change is less than 1%, indicating good convergence of the FEM and validity of the selected mesh fineness.

Results and discussion

The following dimensionality-reduced neural networks were implemented in Python 3.8.10 using the Scikit-learn library (Pedregosa *et al.*, 2011) for dimensionality reduction and TensorFlow library (Abadi *et al.*, 2015) for regression with neural networks.

PCA-NN parameters

Figure 10 shows the results of the validation MSE after training the PCA-NN with different number of retained components.

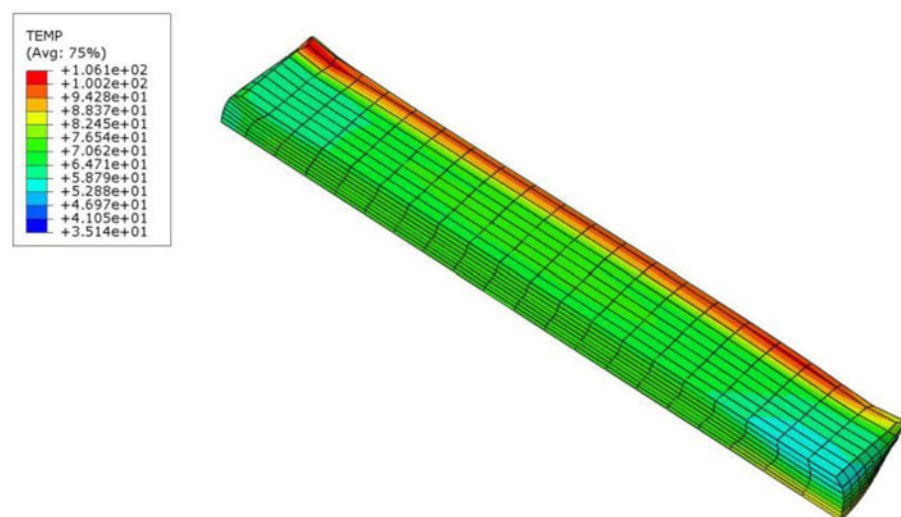


Fig. 7. Change in temperature distribution of the quarter model.

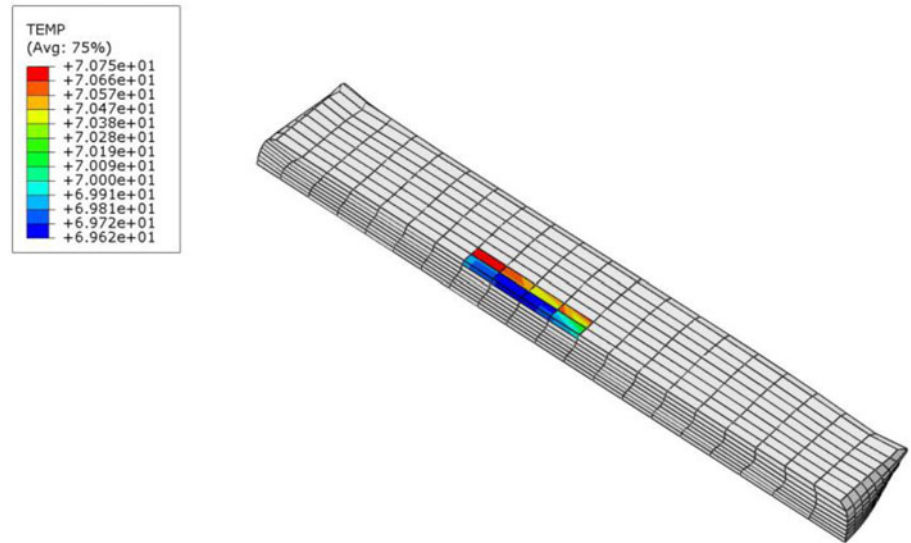


Fig. 8. Eight surface elements for average temperature on the quarter model.

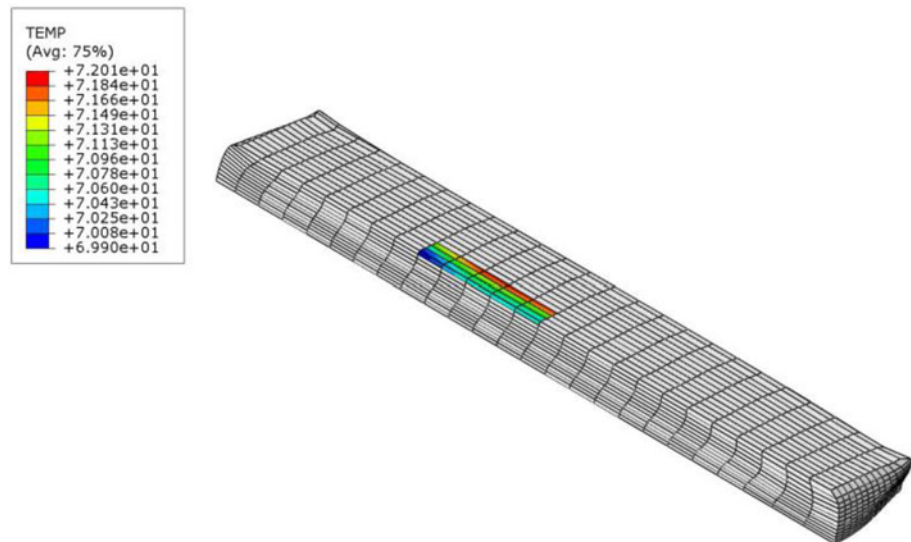


Fig. 9. Refined mesh shows little change in results.

The MSE values were averaged across 40 runs. From observation, we selected seven components for the final PCA-NN reduction.

PLS-NN parameters

Figure 11 shows the results of validation MSE after training the PLS-NN with different number of components retained after dimensionality reduction with PLS across 40 runs. As shown, a minimum MSE occurred with eight components, which is the number we have chosen to use in our PLS-NN model for comparison.

Kernel PCA-NN parameters

In this study, we investigated radial basis functions, cosine, hyperbolic tangent, and polynomial kernels. After comparing the performance of these kernels, a cosine kernel (Ezuko and Zareian, 2019), as shown in Eq. (15), was selected in this work due to its small error convergence when combined with the neural

network. Mathematically, the cosine kernel measures the similarity between two non-zero vectors where the output is the angle between them. This is equivalent to the dot product relation between two vectors and the angle between them.

$$k(x_i, x_j) = \frac{x_i x_j^T}{|x_i| |x_j|}. \quad (15)$$

Figure 12 shows the results of validation MSE after training the kPCA-NN with different number of components retained after kPCA across 40 runs. For the comparison with the neural network and other DR-NNs, we chose to retain eight components based on the results of the figure above.

Training results

Table 3 shows the training hyperparameters used in each model. The DR-NN hyperparameters were selected based on the lowest converging error and the models were implemented in Visual

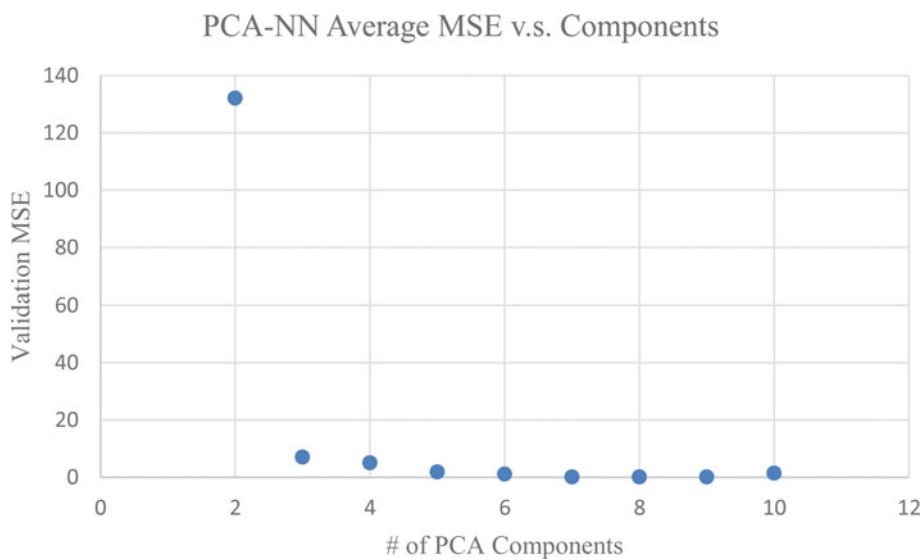


Fig. 10. PCA-NN mean squared error versus number of components retained.

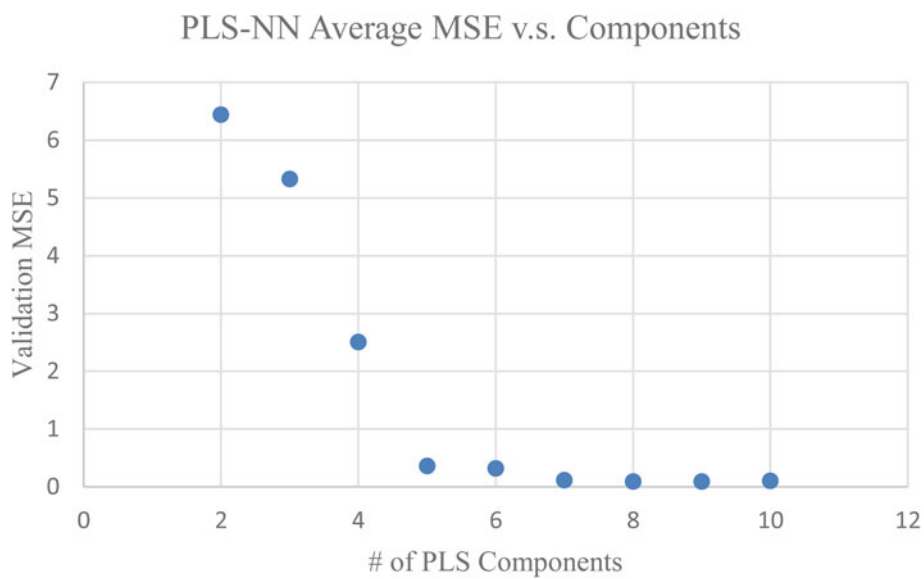


Fig. 11. PLS-NN mean squared error versus number of components retained.

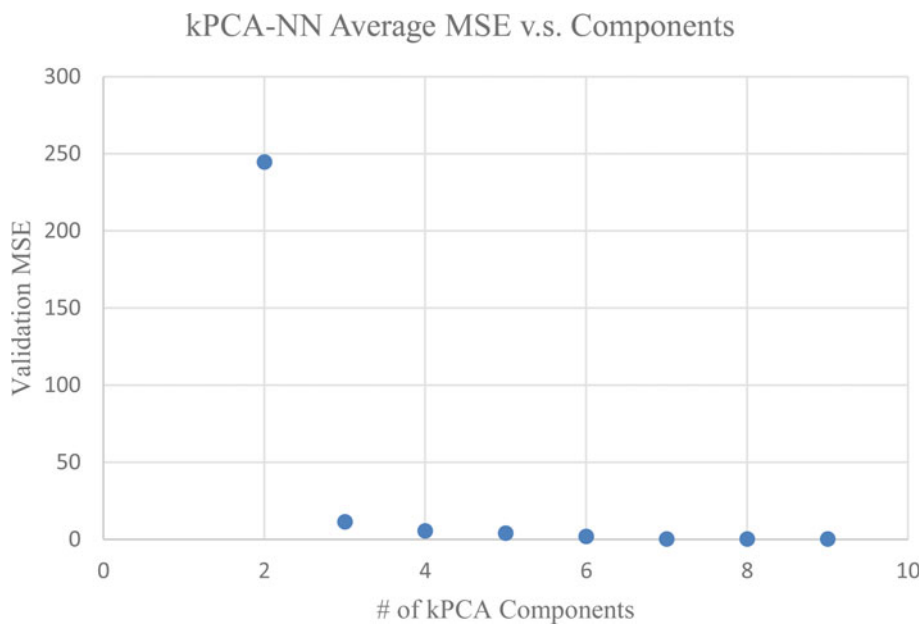


Fig. 12. kPCA-NN mean squared error versus number of components retained using a cosine kernel.

Table 3. Hyperparameters in each DR-NN model

	NN	PCA-NN	PLS-NN	kPCA-NN
Dimensionality	16	7	8	8
Learning rate	0.01	0.01	0.01	0.008
Batch size	32	32	32	32
Epochs	50	70	70	80
# of parameters	5346	4752	4818	4818

Table 4. Temperature change predictions compared with FEM

	NN	PCA-NN	PLS-NN	kPCA-NN	FEM
Temperature change predicted	52.68	52.85	52.69	52.52	52.79

Studio Python. The dimensionality was reduced from 16 dimensions to 8 in the DR-NNs. The computations were performed on a computer with an i7 processor with 4 cores running at 1.8 GHz on Windows 10.

Table 4 shows a sample prediction made by each DR-NN after training. As shown, both the neural network and DR-NNs predict the temperature change with good accuracy.

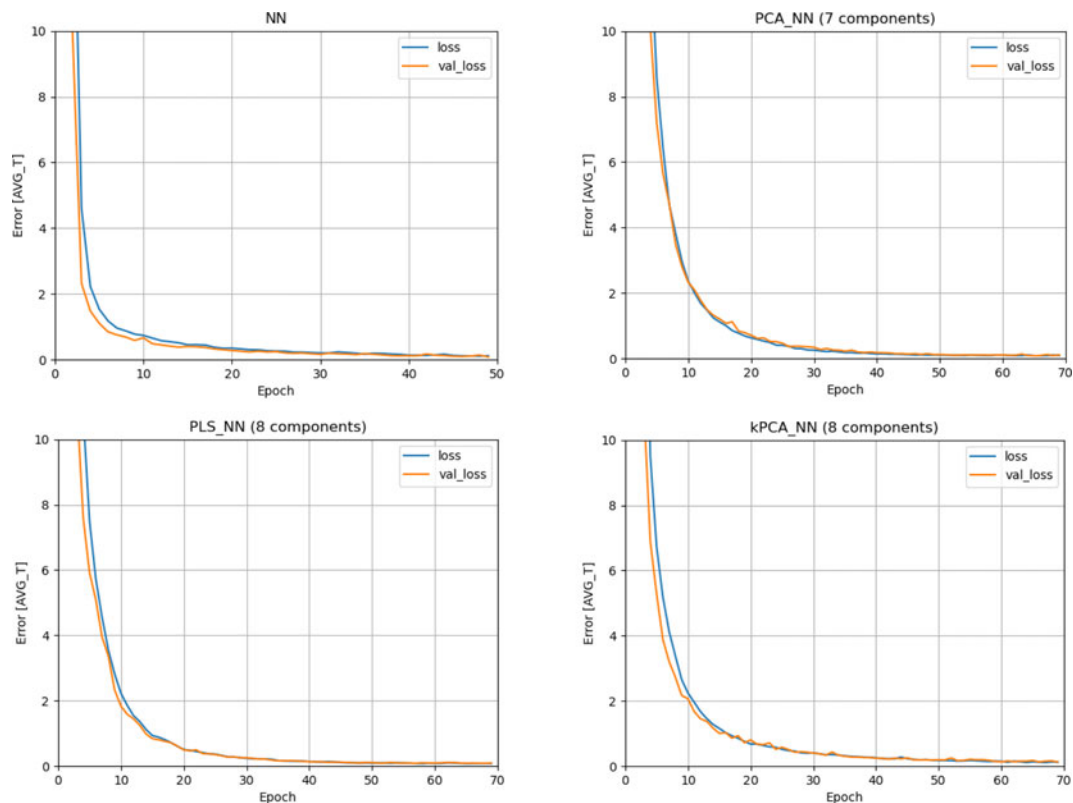
Figure 13 shows the training and validation loss when fitting the DR-NNs and NN. At higher learning rates, the error does not converge to a steady global value. This is because the gradient magnitude calculated is too large and causes the weight changes to oscillate rather than converge. Furthermore, the PCA- and

PLS-based neural networks improved the convergence and prediction uncertainty, showing less oscillatory behavior toward the end of the training epochs. The reduced uncertainty was also evident in the smaller whiskers and box size in the error boxplots for the PCA- and PLS-NNs compared with the neural network in Figures 14 and 15.

The computation times and errors below were compared against FEM results, which took 2.5 min to reach a prediction. Overall, the DR-NNs require more epochs to reach a converging error compared with the standard neural network. However, due to the lower dimensionality of the DR-NNs, the overall training times are reduced by 0.3–0.5 s, as shown in Figure 15, while maintaining comparable accuracy to the neural network, shown in Figure 14.

In total, 5300 hyperparameters, namely weights and biases, were trained in the neural network without dimensionality reduction. The number of weights and biases to be trained was reduced to 4800 in DR-NNs, requiring less storage space. Overall, both the PCA- and PLS-NNs improved upon the training time with little loss of prediction. The kPCA-NN also showed improvement in computation/training time but sacrificed much more prediction accuracy.

The DR-NNs successfully reduced the computation time, storage space, and the uncertainty in prediction error. While the prediction error in all DR-NNs were comparable to the standard neural network, the reduced computation time and storage requirements make DR-NNs a viable substitute for the lengthy FEA simulations. The complexity of the investigated cold rolling problem is low, thus the scale of time and space saved with dimensionality reduction was around 0.3–0.8 s. However, DR-NNs can be extended to high-dimensionality FEM problems

**Fig. 13.** Training and validation MSE loss for DNN and DR-NNs.

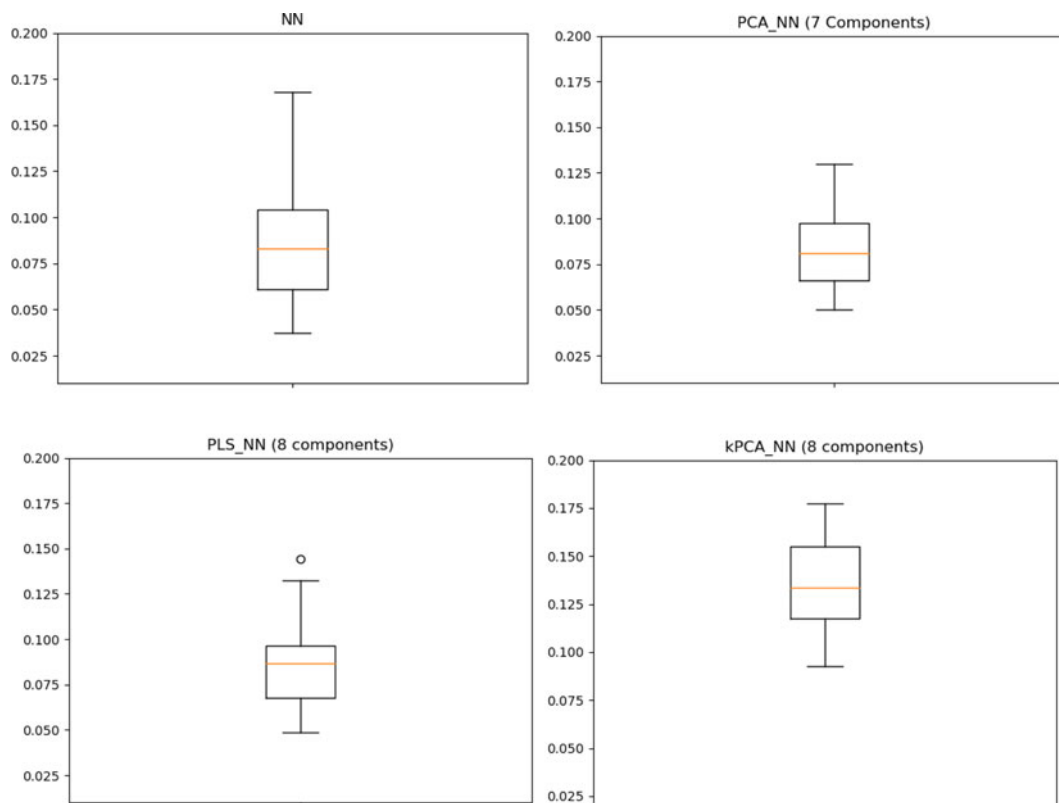


Fig. 14. MSE box plots averaged across 40 runs for DNN and DR-NNs.

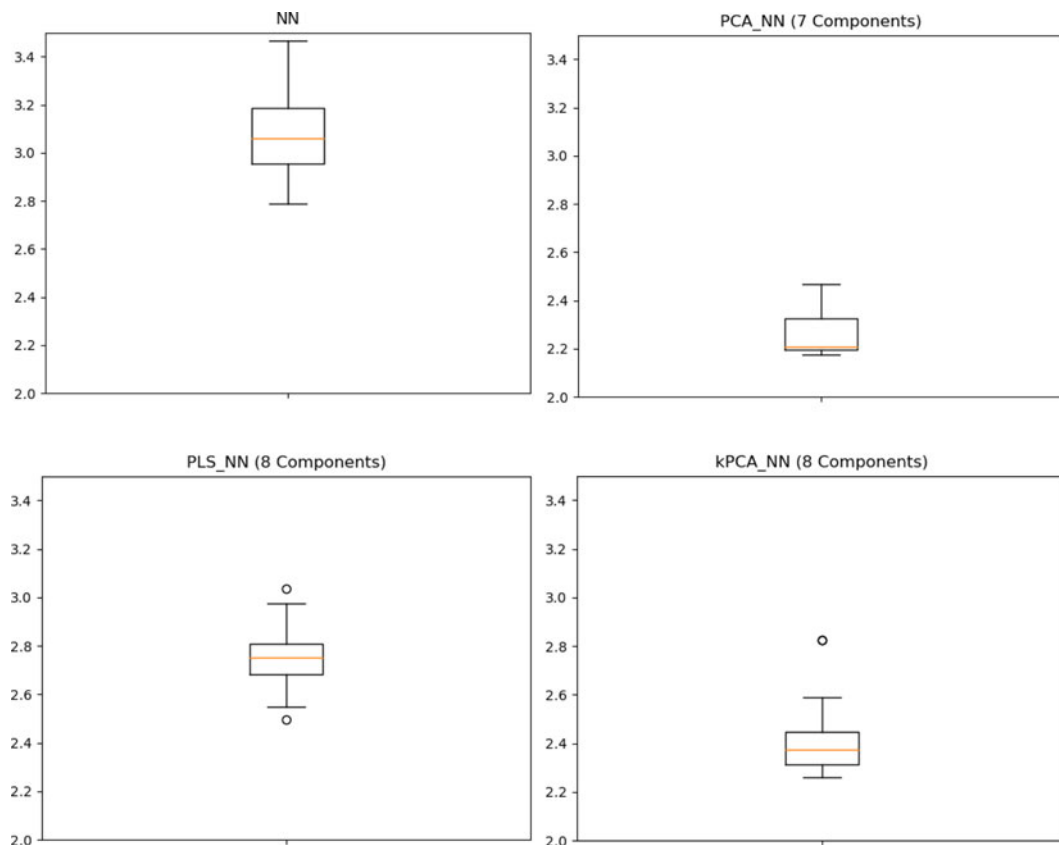


Fig. 15. Computation times in seconds averaged across 40 runs for DNN and DR-NNs.

involving thousands of inputs, where training the hyperparameters of full neural network can become cumbersome such as the case in imaging (Fan and Li, 2006). All in all, these methods can be applied to more complex FEA problems to give accurate assessments of output parameters in the order of seconds, compared to hours with the full simulation run.

In the future, there are improvements that can improve upon the saved computation times and storage requirements with DR-NNs. Here, we conducted trial-and-error tests to determine hyperparameters such as learning rate, batch size, and the reduced dimensionality. Optimization of the DR-NN hyperparameters can further reduce the training time and dimensionality of the problem. In terms of future work in improving the cold rolling simulation, the inclusion of parameters such as roll surface roughness and strain rate-dependent material data can yield more accurate and complex training data for assessing the DR-NNs.

Conclusion

In this paper, several dimensionality reduction techniques combined with a neural network to reduce the computation time and space required to store the trained model hyperparameters are presented. The methods were compared for predicting the output change in temperature in a two-stage cold rolling simulation. Overall, the DR-NNs successfully reduced the training time, storage space, and prediction uncertainty compared with the standalone neural network with minimal loss of accuracy. This proves the viability of using these methods to overcome the long simulation times in other FEA problems and ease the training procedure for neural networks in more complex problems.

Conflict of interest. The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements. This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) under grant number RGPIN-217525.

Data availability. The data that support the findings of this study are openly available in Mendeley Data at <http://doi.org/10.17632/wcp8wrhdyj.1>.

References

- Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado G, Davis A, Dean J, Devin M, Ghemawat S, Goodfellow I, Harp A, Irving G, Isard M, Jia Y, Jozefowicz R, Kaiser L, Manjunath K, Levenberg J, Mane D, Monga R, Moore S, Murray D, Olah C, Schuster M, Shien J, Steiner B, Sutskever I, Talwar K, Tucker P, Vanhoucke V, Vasudevan V, Viegas F, Vinyals O, Warden P, Wattenberg M, Wicke M, Yu Y and Zheng X (2015) TensorFlow: large-scale machine learning on heterogeneous distributed systems. Retrieved from <https://www.tensorflow.org/>.
- Alves L, Nielsen C and Martins P (2011) Revisiting the fundamentals and capabilities of the stack compression test. *Experimental Mechanics* **51**, 1565–1572. doi:10.1007/s11340-011-9480-5
- Chang D (1998) An efficient way of calculating temperatures in the strip rolling process. *Journal of Manufacturing Science and Engineering* **120**, 93–100. doi:10.1115/1.2830115
- Chaves R, Ramirez J, Gorritz J and Illan I (2012) Effective diagnosis of Alzheimer's disease by means of large margin-based methodology. *BMC Medical Informatics and Decision Making* **12**. doi:10.1186/1472-6947-12-79
- Chen J, Chandrashekhara K, Mahimkar C, Lekakh SN and Richards V (2011) Void closure prediction in cold rolling using finite element analysis and neural network. *Journal of Materials Processing Technology* **211**, 245–255. doi:10.1016/j.jmatprotec.2010.09.016
- Deng G, Lu C, Su L and Li J (2013) A study on the aluminum alloy AA1050 severely deformed by non-equal channel angular pressing. *Advanced Materials Research* **651**, 442–447. doi:10.4028/www.scientific.net/AMR.651.442
- Ezukwoke K and Zareian S (2019) Kernel methods for principal component analysis (PCA): a comparative study of classical and kernel PCA. doi:10.13140/RG.2.2.17763.09760.
- Fan J and Li R (2006) Statistical challenges with high dimensionality: feature selection in knowledge discovery. *Proceedings of the International Congress of Mathematicians* **3**, 595–622.
- Ghaisari J, Jannesari H and Vatani M (2012) Artificial neural network predictors for mechanical properties of cold rolling products. *Advances in Engineering Software* **45**, 91–99. doi:10.1016/j.advengsoft.2011.09.016
- Higdon D, Gattiker J, Williams B and Rightley M (2008) Computer model calibration using high dimensional output. *Journal of the American Statistical Association* **103**, 570–583. doi:10.1198/01621450700000888
- Hongwei D and Liang W (2008) Research on intrusion detection based on KPCA-BP neural network. *2018 IEEE 18th International Conference on Communication Technology (ICCT)*. Chongqing, China: IEEE. doi:10.1109/ICCT.2018.8600090
- Hossain K, Rafiei S, Lachemi M, Behdinan K and Anwar M (2016) Finite element modeling of impact shear resistance of double skin composite wall. *Thin-Walled Structures* **107**, 101–118. doi:10.1016/j.tws.2016.06.002
- Hsiao TCR, Lin CW and Chiang HK (2003) Partial least-squares algorithm for weights initialization of backpropagation network. *Neurocomputing*, 237–247. doi:10.1016/S0925-2312(01)00708-1
- Hussain S and Triggs B (2010) Feature sets and dimensionality reduction for visual object detection. Proceedings of the British Machine Vision Conference. Aberystwyth. doi:10.5244/C.24.112
- Hwu Y and Lenard J (1988) A finite element study of flat rolling. *Transactions of the ASME Journal of Engineering Materials and Technology* **110**, 22–26. doi:10.1115/1.3226004
- Johnston N, Muha S, Grandov A and Rossovsky D (2016) Tensile test 1018 fully annealed steel. Available at <https://portfolium.com/entry/tensile-test-1018-fully-annealed-steel> (retrieved 22 September 2021).
- Jolliffe I and Cadima J (2016) Principal component analysis: a review and recent developments. *Philosophical Transactions A* 20150202. doi:10.1098/rsta.2015.0202
- Kamali M, Ponnambalam K and Soulis E (2007) Integration of surrogate optimization and PCA for calibration of hydrologic models: a WATCLASS case study. *2007 IEEE International Conference on Systems, Man and Cybernetics*. Montreal, Canada.
- Kapsoulis D, Tsiakas K, Trompoukis XA and Giannakoglou K (2018) A PCA-assisted hybrid algorithm combining EAs and adjoint methods for CFD-based optimization. *Applied Soft Computing* **73**, 520–529. doi:10.1016/j.asoc.2018.09.002
- Kashid S and Kumar S (2012) Applications of artificial neural network to sheet metal work – a review. *American Journal of Intelligent Systems* **2**, 168–176. doi:10.5923/j.ajis.20120207.03
- Khan O, Jamal A, Arshed G, Arif A and Zubair S (2004) Thermal analysis of a cold rolling process – a numerical approach. *Numerical Heat Transfer* **46**, 613–632. doi:10.1080/10407780490487803
- Kopp R and Dohmen P (1990) Simulation und planung von Walzprozessen mit Hilfe der Finite-Elemente-Methode (FEM). *Stahl und Eisen* **7**, 131–136.
- Kostiv O, Behdinan K and Hashemi S (2012) Finite element analysis of friction-assisted powder compaction process. *SAE International Journal of Materials and Manufacturing* **5**, 333–344. doi:10.4271/2012-01-0051
- Kvalheim O, Arneberg R, Grung B and Kvalheim T (2018) Determination of optimum number of components in partial least squares regression from distributions of the root-mean-squared error obtained by Monte Carlo resampling: determination of optimum number of components in PLS regression. *Journal of Chemometrics* **32**, e2993. doi:10.1002/cem.2993
- Lataniotis C, Marelli S and Sudret B (2018) Extending classical surrogate modelling to high dimensions through supervised dimensionality reduction: a data-driven approach. *International Journal for Uncertainty Quantification*. doi:10.1615/Int.J.UncertaintyQuantification.2020031935

- Lenard J** (2004) The effect of roll roughness on the rolling parameters during cold rolling of an aluminium alloy. *Journal of Material Processing Technology* **132**, 133–144. doi:10.1016/j.jmatprotec.2004.03.026
- Lenard J** (2014) Tribology. In *Primer on Flat Rolling*. Waterloo, Ontario: Elsevier, pp. 193–266.
- Lim L and Lenard G** (1984) Study of friction in cold strip rolling. *Journal of Engineering Materials and Technology* **106**, 139–146. doi:10.1115/1.3225688
- Lin J** (2002) Prediction of rolling force and deformation in three-dimensional cold rolling by using the finite-element method and a neural network. *The International Journal of Advanced Manufacturing Technology* **20**, 799–806. doi:10.1007/s001700200219
- Liu C, Hartley P, Sturgess C and Rowe G** (1985) Elastic-plastic finite-element modelling of cold rolling of strip. *International Journal of Mechanical Sciences* **27**, 531–541. doi:10.1016/0020-7403(85)90043-8
- Liu W, Liu Q, Ruan F, Liang Z and Qiu H** (2007) Springback prediction for sheet metal forming based on GA-ANN technology. *Journal of Materials Processing Technology* **187–188**, 227–231. doi:10.1016/j.jmatprotec.2006.11.087
- Luo C** (1998) A thermal model for the foil rolling process. *Journal of Materials Processing Technology* **74**, 158–173. doi:10.1016/s0924-0136(97)00264-1
- Maitra S and Yan J** (2008) Principle component analysis and partial least squares: two dimension reduction techniques for regression.
- Mehrabi R, Salimi M and Ziaei-Rad S** (2015) Finite element analysis on chattering in cold rolling and comparison with experimental results. *Journal of Manufacturing Science and Engineering* **137**, 016013. doi:10.1115/1.4030379
- Narisetty NN** (2020) Bayesian model selection for high-dimensional data. In *Handbook of Statistics*, Vol. **43**. Champaign, IL, USA, pp. 207–248. doi:10.1016/bs.host.2019.08.001
- Olvera H, Garcia M, Li W, Yang H, Amaya M, Myers O, Burchiel S, Berwick M and Pingitore N Jr.** (2012) Principal component analysis optimization of a PM2.5 land use regression model with small monitoring network. *Science of the Total Environment* **425**, 27–34. doi:10.1016/bs.host.2019.08.001
- Papadopoulos V, Soimiris G, Giovanis D and Papadrakakis M** (2018) A neural network-based surrogate model for carbon nanotubes with geometric nonlinearities. *Computational Methods in Applied Mechanics and Engineering* **328**, 411–430. doi:10.1016/j.cma.2017.09.010
- Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R., Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M and Duchesnay E** (2011) Scikit-learn: machine learning in python. *Journal of Machine Learning Research* **12**, 2825–2830.
- Pietrzyk M and Lenard G** (1991) Deformation heating during cold rolling of aluminum strips. *Journal of Engineering Materials and Technology* **113**, 69–74. doi:10.1115/1.2903384
- Ruan F, Feng Y and Liu W** (2008) Springback prediction for complex sheet metal forming parts based on genetic neural network. Second International Symposium on Intelligent Information Technology Application. NW Washington, DC, USA. doi:10.1109/IITA.2008.425
- Shahani A, Setayeshi S, Nodamaie S, Asadi M and Rezaie S** (2009) Prediction of influence parameters on the hot rolling process using finite element method and neural network. *Journal of Materials Processing Technology* **209**, 1920–1935. doi:10.1016/j.jmatprotec.2008.04.055
- Shahriari M, Pardo D and Moser B** (2020) A deep neural network as surrogate model for forward simulation of borehole resistivity measurements. *Procedia Manufacturing* **42**, 235–238. doi:10.1016/j.promfg.2020.02.075
- Spathopoulos S and Stavroulakis G** (2020) Springback prediction in sheet metal forming, based on finite element analysis and artificial neural network approach. *Applied Mechanics* **1**, 97–110. doi:10.3390/applmech1020007
- Straus J and Skogestad S** (2017) Use of latent variables to reduce the dimension of surrogate models. *Computer Aided Chemical Engineering* **40**, 445–450. doi:10.1016/B978-0-444-63965-3.50076-3
- Tao J, Sun G, Guo L and Wang X** (2020) Application of a PCA-DBN-based surrogate model to robust aerodynamic design optimization. *Chinese Journal of Aeronautics* **33**, 1573–1588. doi:10.1016/j.cja.2020.01.015
- Vimala devi MR** (2009) Multifocus image fusion using the combination of PCA and Wavelet transform. TIPA-2009 - Trend in Image Processing and Pattern Analysis. Solapur, India.
- Wold S, Sjostrom M and Eriksson L** (2001) PLS-Regression: a basic tool of chemometrics. *Chemometrics and Intelligent Laboratory Systems*, 109–130. doi:10.1016/S0169-7439(01)00155-1
- Yadav V, Singh A and Dixit U** (2014) An approximate method for computing the temperature distributions in roll and strip during rolling process. *Journal of Engineering Manufacture* **228**, 1118–1130. doi:10.1177/0954405413498871
- Yazdchi M and Mahyari A** (2008) Detection and classification of surface defects of cold rolling mill steel using morphology and neural network. *Computational Intelligence for Modelling Control & Automation*. Vienna. doi:10.1109/CIMCA.2008.130
- Zhang Y-F, Fitch P and Thorburn P** (2020) Predicting the trend of dissolved oxygen based on the kPCA-RNN model. *Water* **12**. doi:10.3390/w12020585

Chun Kit Jeffery Hou received the BSc. Degree in Mechanical Engineering from the University of British Columbia in 2020. He is currently working toward the MASc. degree with the Advanced Research Laboratory for Multifunctional Lightweight Structures, Department of Mechanical and Industrial Engineering, University of Toronto, ON, Canada. His research interests include machine learning, finite element analysis, mechanics of materials, and failure analysis.

Dr. Kamran Behdinan (PhD) is the founding director and principal investigator of the U of T Advanced Research Laboratory for Multifunctional Lightweight Structures. His research interests include Design and Development of Light-weight Structures and Systems for biomedical, aerospace, automotive, and nuclear applications, Multidisciplinary Design Optimization of Aerospace and Automotive systems, Multi-scale Simulation of Nano-structured Materials and Composites. He has also published more than 370 peer-reviewed journal and conference papers, and 9 book chapters. He has been the recipient of many prestigious awards and recognitions such as the Research Fellow of Pratt and Whitney Canada and Fellows of the CSME, ASME, the Canadian Academy of Engineering, EIC, AAAS, as well as the Associate Fellow of AIAA.