DOI: 10.21062/mft.2025.017 http://www.journalmt.com

Demonstration of Neural Network in Prediction of Bearing Lifetime

Eliška Cézová (0000-0002-1000-3411)
Department of Designing and Machine Components, Faculty of Mechanical Engineering, Czech Technical University in Prague, Technická 4, 166 07 Prague 6, Czech Republic. E-mail: eliska.cezova@fs.cvut.cz

The topic of this paper is the application of machine learning and neural networks in engineering, specifically in the prediction of the lifetime of bearings operating in different conditions. In addition, the basics of machine learning are introduced, giving an idea of the importance of input data quality for model training. It also presents the elements of neural network training to be used in other projects. The article is supplemented by a source code examples written using only the Python language, and some other popular libraries, like the NumPy, Matplotlib, Tensorflow, Keras, and Scikit-learn. The main advantage of the libraries used is that they are freely available and widely used, bringing variety of sophisticated tools for general use.

Keywords: Experiment, Bearing, Neural network, Deep learning, Python

1 Introduction

In engineering, machine learning can be used to analyse data, optimise processes and improve product design and other applications. Using neural networks in engineering allows to predict, analyze and optimize various states of machine parts. In this paper, the focus will be on life prediction of bearings operating under different conditions. Thus, the effect of various factors on the wear level of bearings will be analyzed, which can be further used to optimize maintenance and prevent failures. By using a neural network, complex data can be processed to identify patterns that might not be otherwise considered, which serves to estimate bearing life more accurately. When manufacturing machines, different types of their parts can break down, which quite often affects the maintenance costs of the machines. Higher maintenance costs directly impact the efficiency and profitability of running the machines.

The cause of bearing failure may be incorrect or unprofessional assembly, poor bearing selection, manufacturing error of the parts to be connected, poor lubrication or contaminants entering the bearings. Bearings rotate during manufacturing at high speeds and without sufficient lubrication, dry metal to metal contact can occur, causing wear, heating and possible contamination of the lubricant. Therefore it is necessary to check the correct type, volume and interval of the regeneration of the lubricant. If the bearing becomes contaminated, it is necessary to check the seals. Other undesirable factors affecting wear for the bearing are the vibrations or overloading [1].

2 Description of the neural network

A neural network is a mathematical model inspired by the biological nervous system that consists of interconnected units called neurons. This network is used to solve various tasks in machine learning and artificial intelligence, and is able to adapt itself based on experience and training. Neural networks are used, for example, for image recognition, language translation, prediction, etc.

A neuron in a neural network is the basic building unit that processes input data and transmits information through connections with other neurons. Each neuron has its own input weight vector which changes during the learning of the network. Neurons communicate with each other through activation functions that determine what information to pass on in the network.

Machine learning is a branch of artificial intelligence that deals with the development of techniques and algorithms that allow computers to learn independently and improve their performance based on experience and data. This approach is used in many applications, such as prediction, image recognition, deep learning and many more. Machine learning uses various methods including neural networks, decision trees, linear models, etc.

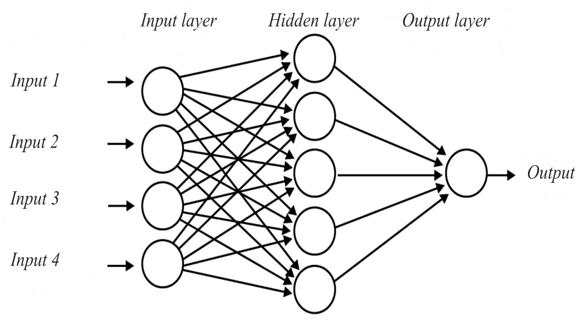

***Fig. 1** Arranging neurons into layers in a feedforward neural network*

The basic building block of neural networks is the processing layer, which can be considered as a data filter, see Fig. 1. A layer is a data processing module that receives one or more tensors as input and has one or more tensors as output. Some layers are stateless, but more often layers have state: layer weights, one or more tensors learned by stochastic gradient descent, which together contain the knowledge of the network [2,3].

3 Using libraries in Python

While implementing machine learning in Python, certain libraries need to be loaded first. The numpy library was used to generate the data. The matplotlib.pyplot library was used to create the graphs for this paper. Another library is tensorflow which is used for machine learning see Figure 2. Keras is a superstructure of tensorflow. Sklearn is a library for machine learning and data preprocessing [3,4,5,6].

The next step is to load the training and test data, see Figure 3. The training data and test data do not come from real measurements, but are only demonstration data.

In the next section of this article, the focus will be on generating data for training the model. It will be the data for bearings speed, load, temperature, vibration, operating time and lifetime, see Figure 4.

```
import numpy as np
import matplotlib.pyplot as plt
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
```

Fig. 2 *Sample source code – use of libraries*

```
train_data = np.loadtxt('train_data_realistic.txt', delimiter=',')
test_data = np.loadtxt('test_data_realistic.txt', delimiter=',')
```

Fig. 3 *Sample source code – data loading*

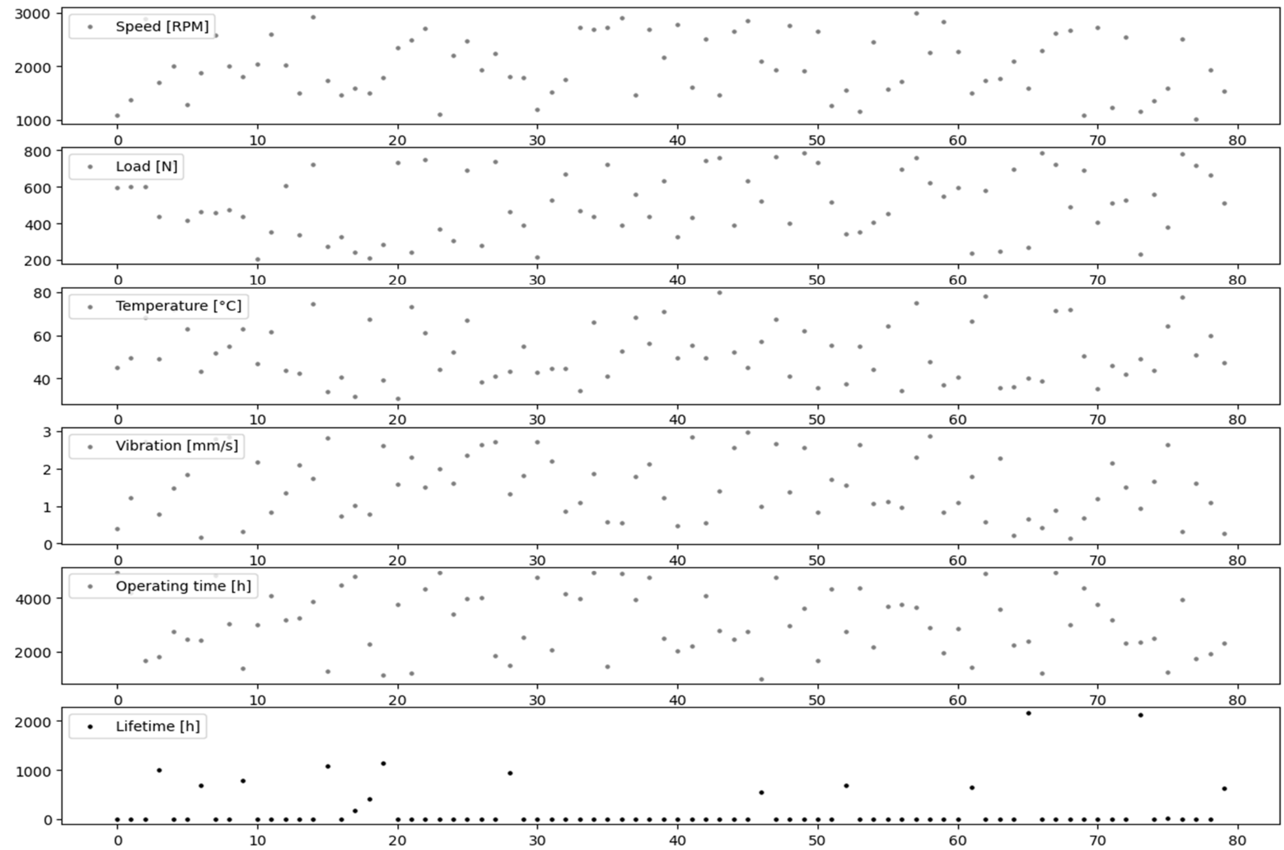

Fig. 4 *Generated data for bearing lifetime*

First, the training data was loaded and normalized. After the data normalization, the data was split into training and validation data. A model was generated, see Figure 5. The model consists of a string and several layers. Each layer is applied to the input data by several simple tensor operations and these operations include weight tensors. The weight tensors, which are attributes of the layers, are where the knowledge of the model persists.

The Relu activation function activates a neuron if it is positive, otherwise it generates zero. Relu is the most popular activation function in deep learning, it deals with tensor operations.

For the demonstration, an input layer with 64 neurons, a hidden layer with 32 neurons and an output layer containing 1 neuron was used - lifetime estimation.

```
model = Sequential([
Dense(64, activation='relu', input_shape=(X_train_scaled.shape[1],)),
Dense(32, activation='relu'),
Dense(1)
])
```

Fig. 5 *Source code sample – creating a model*

The training model was set up for 256 epochs, i.e. 256 iterations over all training and validation samples in mini-batches of 16. (Each iteration over all training data is called an epoch). Increasing the number of epochs beyond this point would be meaningless.

As it can be seen, the training loss decreases with each epoch and the training accuracy increases with each epoch, i.e., the value we are trying to minimize should decrease with each iteration. To avoid overtraining, epochs can be terminated earlier.

A network with a mean squared error (MSE) loss function was constructed, see Figure 6, i.e., the square of the difference between the prediction and the target. This is a commonly used loss function for regression problems. The training metric labeled MAE, which stands for mean absolute error, is the absolute value of the difference between the prediction and the target.

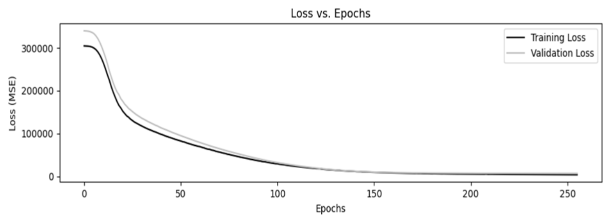

Fig. 6 *Plot of the loss function*

The MSE is calculated by calculating the deviation between the actual value and the predicted value for each data point, squaring this deviation to remove the effect of negative values (since the deviation can be either positive or negative), and finally summing all the squared deviations and averaging them. This average provides a value that expresses how well the model estimates the true values.

The MSE shows how much the model predictions deviate from the actual values. The lower the MSE, the better the model, because it means that the model is more accurate in its predictions.

When training a machine learning model, we use MSE as a criterion for model optimization. The goal is to minimize the MSE during training, which means that the model learns to better approximate the true values. In the model optimization process (e.g., using gradient descent), we try to find parameters that minimize this value. MSE can be very sensitive to outliers in the data.

Figure 7 shows the predicted data and the actual data. The actual data is usually known in advance and is used to evaluate how well the model is performing.

Predicted data is the result of a machine learning model that is trained on historical data and then makes predictions (forecasts) for unknown or future values. The model makes estimates or predictions based on the input data.

A graph of the linear relationship between predicted and actual data is often used to visualize how well a machine learning model predicts outcomes based on actual values. Such a graph can provide useful insight into how close the model's predictions are to the actual values and how strong the linear correlation between them is.

The linear dependency can be seen from the graph, if we see that the points form a straight line or a narrow band around the straight line, it means that there is a strong linear correlation between the actual and predicted values. Otherwise it would be linear independence.

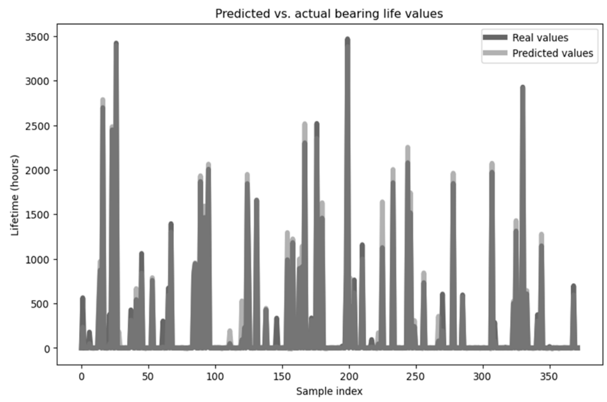

Fig. 7 *Display of predicted and actual data*

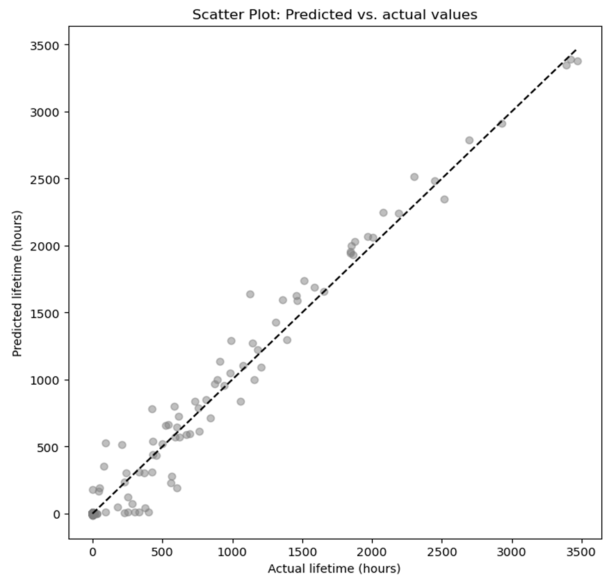

Fig. 8 *Dependence of predicted values on actual values*

In this paper, a linear model is used which assumes a linear relationship between predicted and actual values on Figure 8. The aim is to find the best linear approximation.

4 Conclusion

This paper presents a simplified example of the fundamental use of neural networks and demonstrates the application of free software for this purpose. Using this approach, it is also possible to predict the behavior of critical components in the development of a new generation of high-speed spindles [1]. The primary contribution of this article is to provide essential information that enables readers to try the software independently, without the need to purchase additional software tools. This article serves as an introductory guide to neural networks. For more in-depth discussions of neural network theory, readers are referred to literature sources [3] and [5]. Other professional literature that deals with a similar topic and was used in this work is listed in the sources [6,7,8,9].

Acknowledgement

This work was partly funded by the Czech Ministry of Education under the Institutional support for the development of the research organization No.: RVO12000 for Faculty of Mechanical Engineering of the Czech Technical University in Prague.

References

[1] ONDRUŠKA, J., BÁBICS, L., BOŠANSKÝ, J., PROTASOV, R. (2024). Development hybrid high-speed electro pneumatic spindles drive unit. pp. 49-56. In: *Strojnicky časopis = Journal of Mechanical engineering.* Vol. 74, no. 2 (2024), ISSN 0039-2472

[2] BORTNOWSKI, P., DOROSZUK, B., KROL, R., MARASOVA, D., MORAVIC, M., OZDOBA, M. (2023). Forecasting Blockades of Conveyor Transfer Points Based on Vibrodiagnostics. *Meas. J. Int. Meas.* Confed. 216, 112884. https://doi.org/10.1016/j.measurement.2023.112884.

[3] PECINOVSKÝ, R. (2021). *Python – Complete language manual for the version 3.10*, Grada, Prague. ISBN 978-80-271-3442-7 (in Czech)

[4] CÉZOVÁ, E. (2024). Automated Processing of Experimental Data and Reports Using Free Software, the 62th Conference of Machine Design Departments, Proceedings of the 62nd International Conference of Machine Design Departments (ICMD 2022), pp. 376-387, In: *Springer Nature*, ISBN 978-94-6463-423-5, ISSN 2589-4943, DOI 10.2991/978-94-6463-423-5_42.

[5] CHOLLET, F. (2023). *Deep learning in language Python*, Grada, Prague. ISBN 978-80-271-5133-2 (in Czech)

[6] CÉZOVÁ, E. (2022). Methods and Means in Teaching of Experimental Methods – 60th annual Conference on experimental stress analysis, *EAN 2022 – Conference Proceedings*,

[7] VONDRÁŠEK, D., HADRABA, D., MATĚJKA, R., LOPOT, F., SVOBODA, M. JELEN, K. (2018). Uniaxial tensile testing device for measuring mechanical properties of biological tissues with stress-relaxation test under a confocal microscope. pp. 866 - 872, In: *Manufacturing Technology*, Vol. 18, No. 5. ISSN 1213-2489

[8] BITTNER, V., JEŽDÍK, R., KUBOVÝ, P., LOPOT, F., STOČEK, O., HAVLÍČEK, M., SVOBODA, M., JELEN, K. (2019). Possibilities of Using Tram Windscreen Impact Tests in Analysis of Human-Machine Accidents. pp. 912 – 916, In: *Manufacturing Technology*, Vol. 19, No. 6. ISSN: 1213 – 2489.

[9] SVOBODA, M., SCHMID, V., SAPIETA, M., JELEN, K., LOPOT, F. (2019). Influence of the damping system on the vehicle vibrations. pp. 1034 – 1040, In: *Manufacturing Technology*, Vol. 19, No. 6. ISSN: 1213 – 2489