

PAPER • OPEN ACCESS

Justification of the possibility of application of a neural network for predicting methane release during underground mining of gas-bearing coal seams

To cite this article: M V Dvoryanchikov and L D Pavlova 2021 *IOP Conf. Ser.: Earth Environ. Sci.* **823** 012033

View the [article online](#) for updates and enhancements.



ECS **240th ECS Meeting**
Digital Meeting, Oct 10-14, 2021
We are going fully digital!
Attendees register for free!
REGISTER NOW

Justification of the possibility of application of a neural network for predicting methane release during underground mining of gas-bearing coal seams

M V Dvoryanchikov and L D Pavlova

Siberian State Industrial University, 42 Kirova str., Novokuznetsk, 654007, Russia

E-mail: kicksaflips@gmail.com

Abstract. The mathematical apparatus of neural networks is presented and its application is substantiated for solution of the problem of predicting methane release in underground coal mining. An example of using a neural network for a regression problem in the Matlab software package is given.

1. Introduction

When carrying out underground work, safety comes first. Gas drainage of coal seams is used to combat gas releases. To make a decision to continue or stop the work, it is important to recognize and predict a gas-dynamic phenomenon timely, to take into account many factors. When the number of properties of the studied phenomenon increases to tens, it becomes very difficult to establish any connections between them. To solve such problems, a special mathematical apparatus has been developed, called a neural network [1].

2. Basics of neural networks

Artificial neural networks that simulate the operation of biological neural networks are statistical learning models that are used in machine learning. These networks are presented as systems of interconnected “neurons” that send messages to each other. The connections within the network can be systematically adjusted based on inputs and outputs, making them ideal for supervised learning. Due to its structure, a program with a neural network acquires the ability to analyze and even memorize various information, just like the human brain does [2].

Neural networks are used to solve completely different kinds of problems: recommending systems, forecasting, determination of the position of objects from video images, in unmanned vehicles, when predicting the failure of devices, etc. [3-4].

The smallest element of a neural network is a neuron (figure 1). It receives a vector of input data, performs calculations on it and passes it on. There are 3 main types of neurons: input, hidden (there may be several layers) and output. In the case of an input neuron: input = output. In other cases, the input receives the total information from the previous layer, it is normalized using the activation function $f(x)$ and enters the output field.



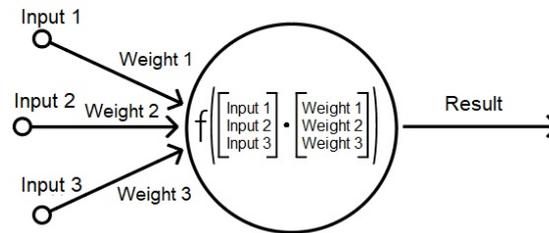


Figure 1. Schematic representation of a neuron.

A synapse is a connection between two neurons. It stores a weight that changes the information passing through it.

The activation function $f(x)$ is used to convert the sum of the input data to the normalized range.

The input signals are combined by means of the scalar product with the corresponding weights of the neuron to obtain the resulting signal, which is modified by the activation function $f(x)$.

An epoch is an iteration during which the entire training set passes through the network with the necessary correction of the weights.

The construction and configuration of such a network is carried out through training based on the available information. To do this, you need to know the correct results for certain inputs so that you can use the difference between the actual and expected outputs to find errors and correct the weights. The network needs to be told enough correct inputs and outputs so that it can then use it to process new inputs.

3. The task of predicting methane release based on a neural network

3.1. Formulation of the problem

The purpose of constructing a neural network in this work is to predict methane release during underground mining of gas-bearing coal seams based on field observations in the mine conditions of a particular mining area, the main properties of the gas-bearing geomass and its state parameters.

The initial data are:

- Measurements of the operation of directional drilling wells.
- Instruction on control of mine air composition, determination of gas content and establishment of mine categories for methane and/or carbon dioxide.
- Mathematical apparatus of neural networks.

The criterion for the development efficiency is the issuance of recommendations for making management decisions based on forecasting the future methane concentration.

3.2. Design stages

When building a neural network, the following stages can be distinguished:

1. Data collection, formation of initial data tables of the so-called database.
2. Primary data processing – removal of fluctuations, reduction to one dimension, removal of rows with empty values or, if possible, restoration of their values.
3. The use of simple models, graphs to highlight general patterns (figure 2).
4. Normalization of data for a neural network, division into training, debugging and test samples [5].
5. Selection of the type of network, its parameters.
6. Training on a training set.
7. Verification and correction on the test sample, analysis of the results.
8. Supplement with new data and repetition of stages.

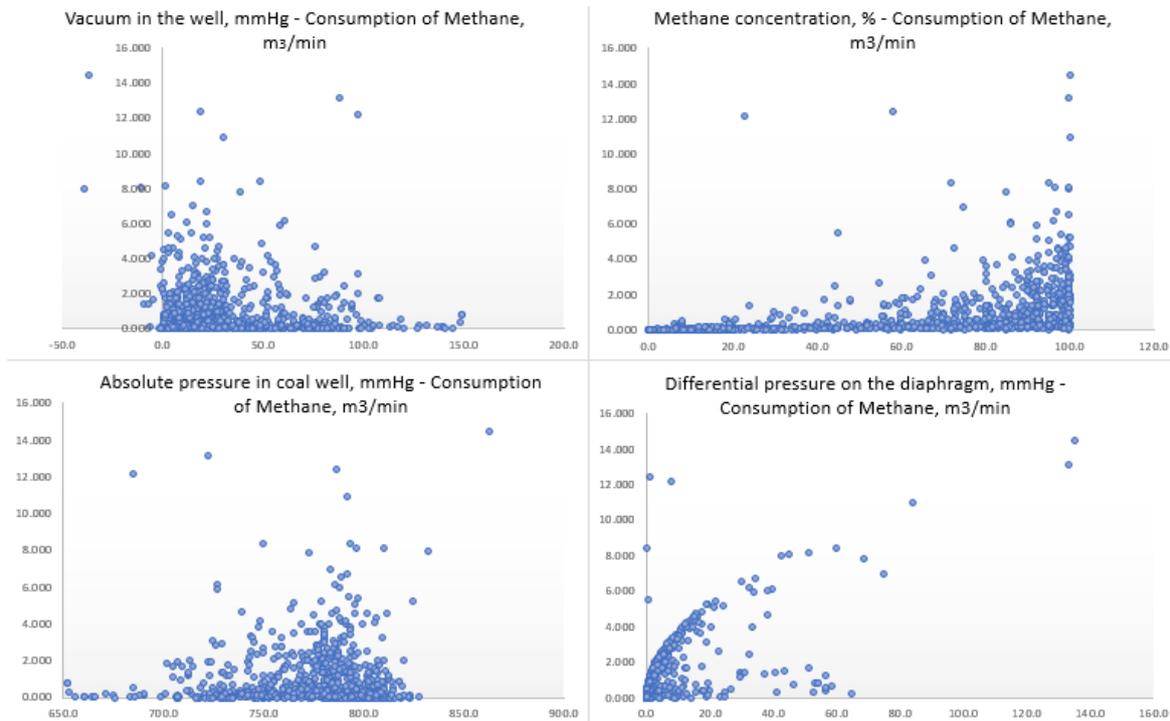


Figure 2. Visualization of a part of the original dataset.

4. Software platforms for designing neural networks

In practice, in most cases in machine learning the neural networks are not built from scratch. This is usually done with highly optimized platforms that do all the hard work (table 1).

Table 1. Software tools for machine learning.

Name of the software	Purpose
TensorFlow	Machine learning based on data streams
Deeplearning	A framework for deep learning in Java
Keras	High-level interface, a set of abstractions for quickly building neural networks regardless of the computational backend
Caffe	Parallel multiprocessor computations of multidimensional data arrays
PyTorch	Addressing the problems of computer vision and natural language processing
Microsoft Cognitive Toolkit	Representation of a neural network as a series of computational steps using a directed graph
Matlab	A software package for solution of technical problems includes tools for the synthesis and analysis of neural networks

5. Solution of the regression problem using Matlab

The Deep Learning Toolbox in Matlab can be used to solve the regression problem using a neural network. After training, the neural network must be able to reproduce the output from the new input. Based on the identification of the object, it will be possible to proceed to the forecasting task in the future.

Let the input set be 1000 well measurements by 7 parameters:

- orifice diameter, mm;
- methane concentration,%;
- depression at the well, Mmhg;

- absolute pressure in the well P, MmHg;
- absolute pressure in the working P atmTM, MmHg;
- temperature of a methane air mixture, C;
- pressure drop across the orifice, MmHg.

The output sample is two parameters with 1000 values:

- consumption of methane-air mixture, m³/min;
- methane consumption, m³/min.

Samples are split in percentage terms: 70% – training sample, 15% – debug sample and 15% – test sample.

The results of training a neural network using the Levenberg-Marquardt algorithm with back propagation of an error are shown in figure 3 [6].

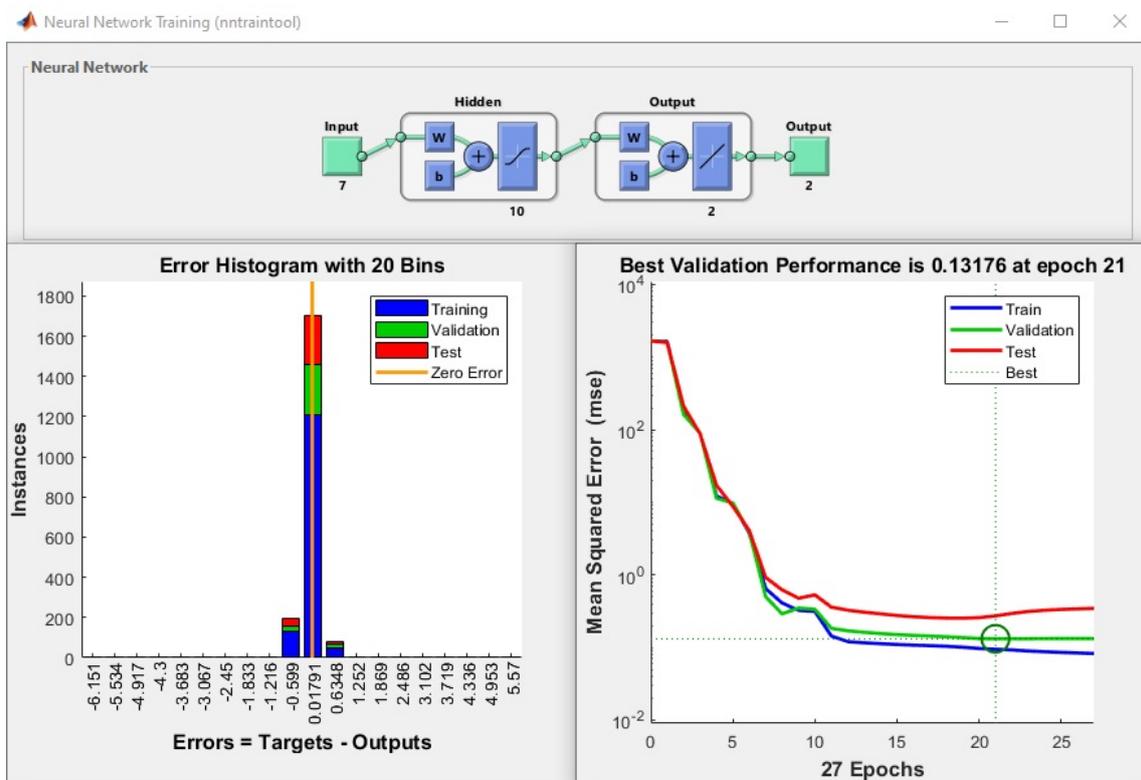


Figure 3. Results of calculations of a neural network.

The structure of a neural network is described by the vector of input values, 10 hidden and 2 output neurons receiving the result.

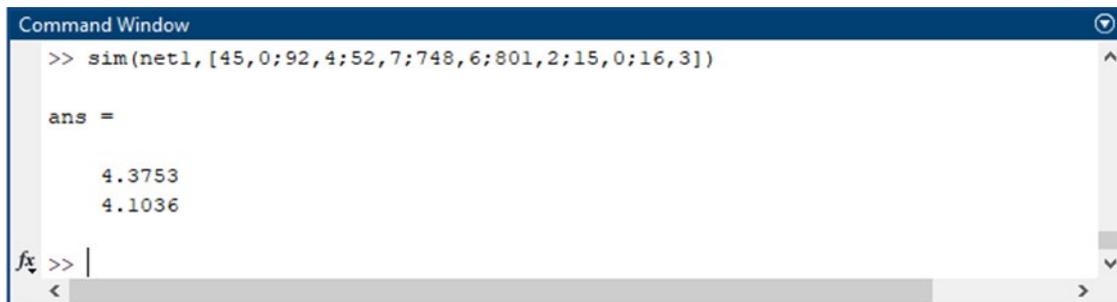
Vector \bar{x} of 7 parameters is fed to the **Input**. In the hidden layer of 10 neurons (**Hidden**), the vector is multiplied by weight matrix W_1 of size 7x10, the bias vector b of size 10 is added to the result, and the result is passed to the nonlinear activation function $f(x)$ (1). In the second block of neurons (**Output**), the weight matrix W_2 , the bias vector b_1 , are applied to the vector, and without the nonlinear function in (**Output**), the result vector \bar{y} (2) is obtained.

$$\bar{h} = f \left([x_1 \dots x_7] \begin{bmatrix} w_{1,1}^{(1)} & \dots & w_{1,10}^{(1)} \\ \vdots & \ddots & \vdots \\ w_{7,1}^{(1)} & \dots & w_{7,10}^{(1)} \end{bmatrix} + [b_1^{(1)} \dots b_{10}^{(1)}] \right) \quad (1)$$

$$\bar{y} = [h_1 \dots h_{10}] \begin{bmatrix} w_{1,1}^{(2)} & w_{2,1}^{(2)} \\ \vdots & \vdots \\ w_{1,10}^{(2)} & w_{2,10}^{(2)} \end{bmatrix} + [b_1^{(2)} \ b_2^{(2)}] \quad (2)$$

The right graph in figure 3 shows the dependence of the mean square error on training iterations. The error is shown for 3 datasets: trainer (blue line), debug (green line) and test (red line). Learning stops after the error on the debug set stops decreasing (21 epochs in figure 3). Also figure 3 shows a histogram of errors distribution, where they are calculated as the difference between the actual values (target) and the results of calculations of the neural network.

If necessary, check for specific values using the matlab console can be performed (figure 4).



```

Command Window
>> sim(net1, [45, 0; 92, 4; 52, 7; 748, 6; 801, 2; 15, 0; 16, 3])

ans =

    4.3753
    4.1036

fx >> |
< >

```

Figure 4. Check of values in matlab console.

Output 1: 4.3753; Target value 1: 4.5514; The error is: 0.1761

Output 2: 4.1036; Target value 2: 4.2055; The error is: 0.1019

6. Conclusions

Solution of problems using neural networks is a kind of black box in which it is difficult to establish the significance of the parameters. To achieve the specified accuracy, very large datasets are usually required, which are difficult to find and process. When building a network, the choice of a structure that provides an optimal solution to the problem is also of great complexity. Nevertheless, they are used to solve a wide range of problems when it is difficult to establish an explicit dependence of input-output parameters, and they have the ability to learn and retrain.

When predicting the state of methane in a mine, taking into account a large number of measured indicators, it is advisable to use the apparatus of neural networks. An important condition for ensuring effective gas release management is the correct forecast of methane release at the extraction area and the permissible load on the production face by gas factor.

References

- [1] Kopets D 2020 *Classical Computer Science Problems in Python* (St. Petersburg: Peter) p 256
- [2] Miller S 2014 *Mind: How to Build a Neural Network (Part One)* <http://stevenmiller888.github.io/mind-how-to-build-a-neural-network/>
- [3] Korablina T V et al 2018 *Proc. of VII Int. Sci. Practical Conf. on Fundamental Scientific Research on Theoretical and Practical Aspects* (Kemerovo: ZapsibNC) vol 2 pp 73–77
- [4] Trofimov V B 2016 *Proc. of IV All-Russian Sci. Practical Conf. on Modeling and High Technology Information Technologies in Technical and Socio-economic Systems* (Novokuznetsk) part 2 pp 54–59
- [5] Andrew N G *Machine Learning Yearning. Technical Strategy for AI Engineers, In the Era of Deep Learning* <https://github.com/ajaymache/machine-learning-yearning>
- [6] Marquardt D 1963 *SIAM Journal on Applied Mathematics* **11(2)** 431–441