УДК 004.032.26

BASICS OF ARTIFICIAL NEURAL NETWORKS (ANNs)

Stud. BIT 1-16 A. Lukashev
Language and scientific supervisor
associate professor Ya. Absaliamova
Kyiv National University of Technologies and Design

The aim of this research is to help the newcomers to the world of ANNs, the people who want to find out some basic information about ANN or just who want to start learning ANN Programming but do not know where to start. ***The tasks*** are to have a look at some of the most important basic terms of ANN Programming, find out some mathematical basis of ANN Programming and to prepare the reader for a deeper understanding of the ANN Programming. The research is written in a simple way, using only the most important terms you will definitely meet in ANN Programming, so it is easy enough for the beginners.

The study object is Artificial Neural Networks, their work and basics of their programming. Thus, the object of this research is a computing system inspired by the biological neural networks.

The methods of observation (having a look at the very basics of Artificial Neural Networks work, learning some examples) and description (describing some of the most important terms of ANN programming, having a look at some functions and sorting out how they work) are used in the paper.

The scientific novelty of this research is the fact that it is very important for the very beginners of ANN programming and was made mostly for them. This research will help them to fit in the world of ANNs easier, focusing on some basic terms and methods (functions) which “create” ANN programming, which will help beginners to understand how do ANNs work.

Results and discussion. The research itself starts with a representation of a ANN structural scheme. There the audience sees:

1) Typical ANN structural scheme. 3 types of neurons: an input neuron, an output neuron and hidden layer`s neurons.

2) A neuron divided into two halves, which is needed for representing that every neuron has both input and output signals.

3) Except for 2 halves we see some numbers there. This shows us, that input and output signals usually work only in [0 – -1] and [-1 – 1] diapasons of numbers. In case the numbers are bigger or smaller than [-1 – 1] we have to divide any of those numbers into 1 (or -1 in case when the number on input is smaller), – this process is called “normalization”.

4) The next layer is so-called hidden layer of neurons. Let`s imagine that each of those lines, which are going from one neuron to another, is signed with w1, w2, w3 etc. Those lines, or connections, are called synapses. And synapses have only 1 parameter - weights, which are some numbers that can and, actually, will be changing during the learning process of an ANN and when we want to change the task we have given to our ANN. As an example – a Network that does AND function and a Network that does OR function can have similar signal numbers (the number that goes to our input neuron) but they will have different weights in any way. Also, I will give you another example in order to show how information is divided in the input of the next neuron. Let`s say we have 3 neurons that are giving some signals to 1 neuron. As we already know, that means that we have 3 synopses, three lines. And each line has its own weight. And in the input of the neuron the biggest weight will be dominant.

Now about the hidden layer itself.

Hidden layer`s input signal will be equal a total of multiplications of l1 signal and w1 and l2w2 in other words – input of hidden neuron is a sum of all input neuron`s data multiplied on its own weights.

5) Activation functions (linear, sigmoid and hyperbolical tangent). Activation function is a way to normalize our input data. There is a plenty of AF actually, but most of ANNs usually use only 3 of them – the linear function, sigmoid and hyperbolical tangent, and the only difference between them is their diapason of numbers:

- ❖ The linear function (f(x) =x) is mainly used in cases when we either want to test our ANN or to give our data to the next neuron without any changes.
- ❖ Sigmoid is the most commonly used function of this three. It has diapason of [0-1] and most of examples in the net are shown with this particular function. Also it is called logistical function. But, if we have any number that can have a negative value, we will have to use another function.
- ❖ Hyperbolical tangent is used only in those cases when we have a negative value.

Now, as we have figured out the structure of a typical ANN, we need to talk about the main thing about ANN – the learning process. Here are some basic terms and their meanings:

- ▪ A train set – it is a sequence of data, which ANN uses.
- ▪ An iteration – it is a counter – an indicator – that shows how many train sets our ANN`s have made, passed in general.
- ▪ An era – it is an indicator that shows how many full train sets our ANN`s have completed in general. This number is set manually.

Finally, when a person is going through any kind of a learning process mistakes are inevitable. And that concerns ANNs too. While learning, ANNs will be making errors, an error in ANN programming is a percent value, which shows the divergence between the result you want and the result your get. An error appears in every era and every era it decreases. There are 3 basic ways to count out errors which ANN will be making during its learning process – MSE (mean squared error), Root MSE and Arctan. Of course, there are more methods, but these three are the most common of all. There are no limits in using these methods, so you can use them however you like and whenever you want, just use the one, that you like the most. The only difference between them is the way they count:

- Arctan works on the tenet of “The bigger difference – the bigger error is” and so it has the biggest error value of these three.

- Root MSE is an opposite to Arctan, so it has the smallest error value.

- And MSE has the best balance of all three, so that is why it is the most commonly used function.

Conclusion. Summarizing: we have a good starting point for the beginners which will help them to understand the basics of ANN programming. This research gathers up the information that is important for beginners and describes some of the most important terms of ANNs, and, as a result, we have a deeper understanding of ANN and greater chances to fit in the world of ANNs Programming.

Keywords: *Neural Network, Artificial Neural Network, Programming Technologies, Neurons.*

REFERENCES

1. Miller Steven. How to Build a Neural Network / Steven Miller. – 2015. – P. 1 – 2.
2. Haykin Simon. Neural Networks / Simon Haykin. – Ontario : Pearson. Prentice Hall. – 2006. – P. 4.
3. Rashid Tariq. Make Your Own Neutral Network / Tariq Rashid. – North Charleston : Create Space Independent Publishing Platform. – 2016. – P. 12 – 39.
4. Gurney Kevin. An Introduction to Neural Networks / Kevin Gurney. – London : UCL. – 1997. – P. 5 – 7.