Feedforward Neural Networks

Harvey Alférez, Ph.D.

Agenda

- Introducing the **Feedforward Backpropagation Neural Network**
- Understanding the **Feedforward Algorithm**
- Understanding the **Backpropagation Algorithm**

Introduction

- The feedforward backpropagation neural network: One of the most common neural network architectures.
  - It can be applied to many different tasks.

Introduction (Cont.)

- "**Feedforward**" (alimentación hacia adelante) describes how this neural network processes and recalls patterns.
  - **Neurons** are ***only*** connected **foreword**.
  - Each layer of the neural network contains connections to the **next layer** (for example, from the input to the hidden layer), but there are no connections back.
    - This differs from the Hopfield neural network (fully connected and its connections are both forward and backward).

Introduction (Cont.)

- "**Backpropagation**" describes how this type of neural network is **trained**.
- **Backpropagation** is a form of **supervised training**.
  - The network must be provided with **both sample inputs and anticipated outputs**.
    - The **anticipated outputs** are compared against the **actual outputs for given input**.
  - Using the anticipated outputs, takes a calculated **error** and a**djusts the weights of the various layers backwards** from the output layer to the input layer.

Introduction (Cont.)

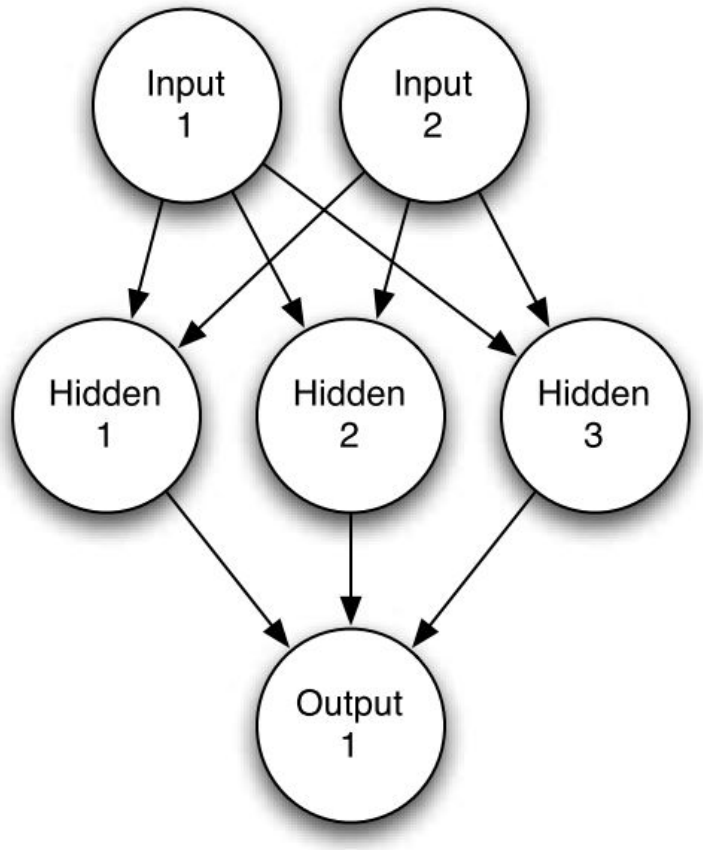
- The backpropagation and feedforward algorithms are often used together; however, this is by ***no*** means a requirement.

A Feedforward Neural Network

- It is **similar** to the types of neural networks that we have already examined.
- **Begins** with an **input layer**.
- The input layer may be connected to a hidden layer or directly to the output layer.
- If it is connected to a hidden layer, the hidden layer can then be connected to another hidden layer or directly to the output layer.
- In common use, **most neural networks will have one hidden layer**, and it is very rare for a neural network to have more than two hidden layers.

The Structure of a Feedforward Neural Network

Figure 5.1: A typical feedforward neural network (single hidden layer).



Choosing Your Network Structure

- There are **many ways** that **feedforward** neural networks can be constructed.
  - You must decide **how many neurons** will be **inside** the in**put and output layers.**
  - You must also decide **how many hidden layers** you are going to have and **how many neurons will be in each of them**.
- In this presentation we will cover some of the general “rules of thumb” that you can use to assist you in these decisions.
  - Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks (MIT Press, 1999).

The Input Layer

- The input layer is the conduit through which the **external environment presents a pattern** to the neural network.
- Once a pattern is presented to the input layer, the **output layer will produce another pattern**.

The Output Layer

- The output layer of the neural network is what actually **presents a pattern to the external environment**.
- To determine the **number of neurons to use in your output layer**, you must first consider the intended use of the neural network:
  - If the neural network is to be used to **classify** items into groups.
    - Then it is often preferable to have one output neuron for each group.
  - If the neural network is to perform **noise reduction** on a signal.
    - Then it is likely that the number of input neurons will match the number of output neurons.

The Output Layer (cont.)

- For example, consider a program that is used for **optical character recognition (OCR).**
- **Input layer:** To determine the **number of neurons** used for the OCR example.
  - The **number of input neurons** that we will use is the number of pixels that might represent any given character:
    - A 5x7 grid contains a total of 35 pixels.
    - Therefore, the OCR program has 35 input neurons.

The Output Layer (cont.)

- The **number of output neurons** will vary depending on how many characters the program has been trained to recognize.
  - The default training file that is provided with the OCR program is used to train it to recognize 26 characters.
- Using this file, the neural network will have 26 output neurons.

Solving the XOR Problem

- We will examine a simple neural network that will learn the XOR operator.
- XOR.java

Activation Functions

- Most neural networks pass the **output** of their layers through **activation functions**.
- The neural network program in the last section (XOR) used the **sigmoid activation function**.
  - The sigmoid activation function is the **default choice** for the FeedforwardLayer class.
- It is possible to use others.

```
network.addLayer(new FeedforwardLayer(new ActivationTANH(),2));
```

The **hyperbolic tangent** should be used, rather than the sigmoid function.

Activation Functions (Cont.)

- Activation functions:
  - Hyperbolic Tangent
  - Sigmoid
  - Linear

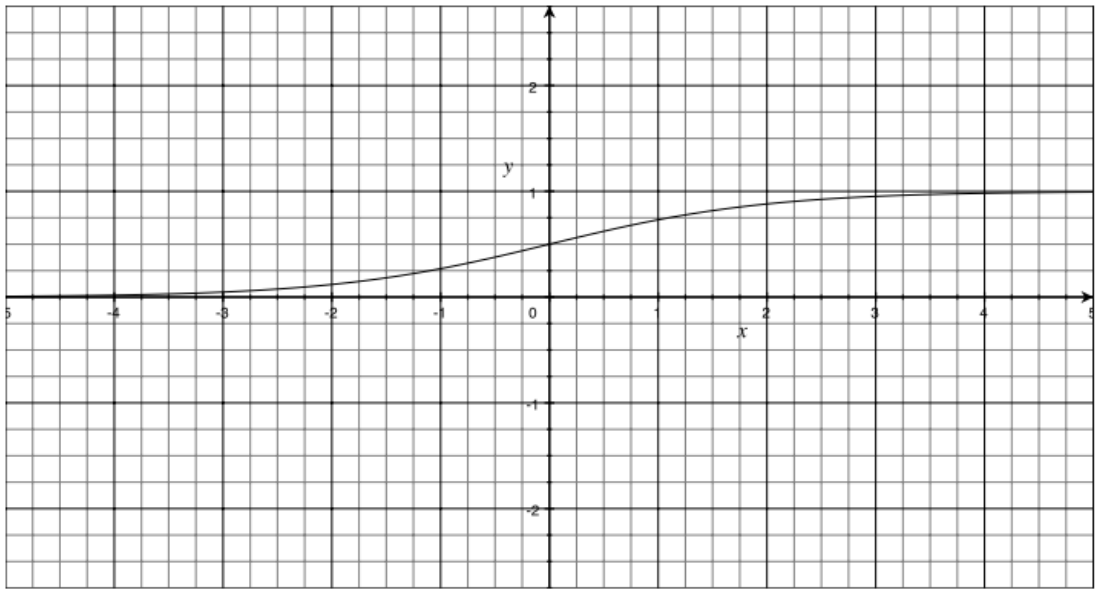
Using a Sigmoid Activation Function

Equation 5.1: The Sigmoid Function

$$f(x)=\frac{1}{1+e^{-x}}$$

- The term ***sigmoid*** means curved in two directions, like the letter “S.”

Figure 5.2: The Sigmoid function.



Using a Sigmoid Activation Function (Cont.)

- One important thing to note about the sigmoid activation function is that **it only returns positive values**.
  - If you need the neural network to return negative numbers, the sigmoid function will be unsuitable.

Using a Hyperbolic Tangent Activation Function

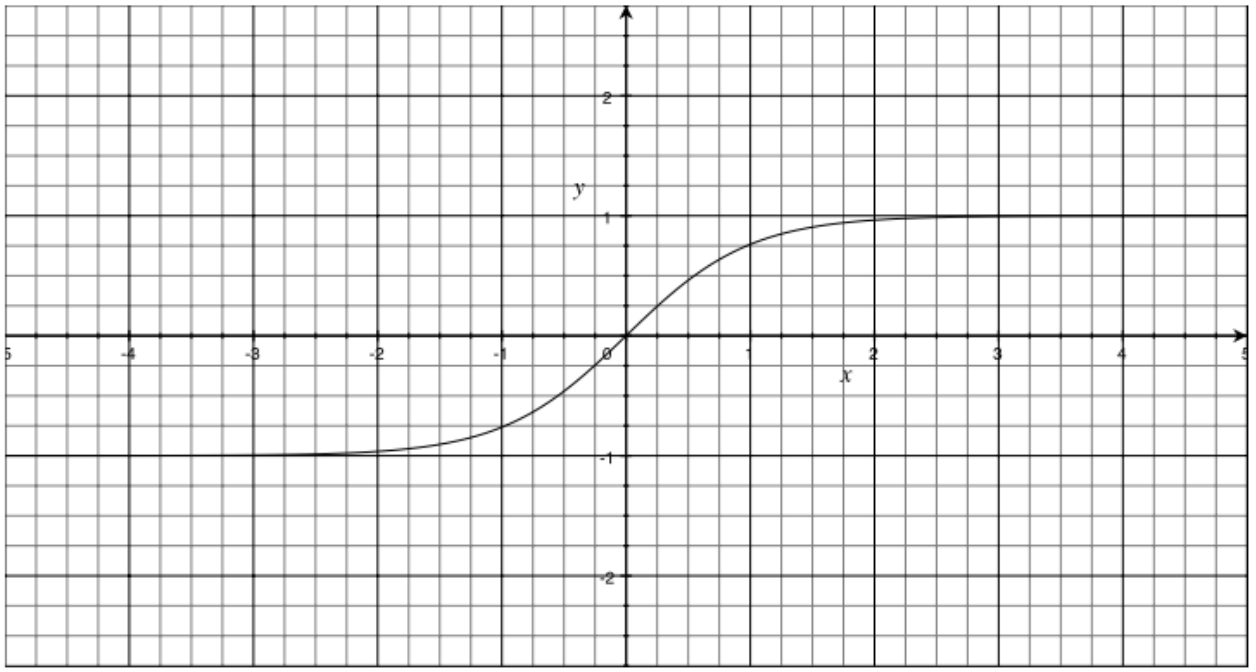
- It is possible to “move” the sigmoid function to a region of the graph so that it does provide negative numbers.
  - This is done using the hyperbolic tangent function.

Equation 5.2: The TANH Function

$$f(x)=\frac{e^{2x}-1}{e^{2x}+1}$$

Using a Hyperbolic Tangent Activation Function (Cont.)

Figure 5.3: The hyperbolic tangent function.



The Number of Hidden Layers

- There are two decisions that must be made regarding the hidden layers:
  - **how many hidden layers** to actually have in the neural network
  - **how many neurons** will be in each of these **layers**.
- Problems that require two hidden layers are rarely encountered.
- In fact, for many practical problems, there is no reason to use any more than one hidden layer.

The Number of Neurons in the Hidden Layers

- Deciding the **number of neurons in the hidden layers** is a **very important part** of deciding your overall neural network **architecture**.
  - Though these layers do not directly interact with the external environment, they have a tremendous influence on the final output.
  - Both the number of hidden layers and the number of neurons in each of these hidden layers must be carefully considered.

The Number of Neurons in the Hidden Layers (Cont.)

- Using **too few neurons** in the hidden layers will result in something called **underfitting**.
  - **Underfitting** occurs when there are too few neurons in the hidden layers to adequately detect the signals in a complicated data set.

The Number of Neurons in the Hidden Layers (Cont.)

- Using **too many neurons** in the hidden layers can result in several problems.
  - May result in **overfitting**: The neural network has so much information processing capacity that the limited amount of information contained in the training set is not enough to train all of the neurons in the hidden layers.
  - The amount of **training time** can increase to the point that it is impossible to adequately train the neural network.

The Number of Neurons in the Hidden Layers (Cont.)

- Rules-of-Thumb:
  - The number of hidden neurons should be between the size of the input layer and the size of the output layer.
  - The number of hidden neurons should be 2/3 the size of the input layer, plus the size of the output layer.
  - The number of hidden neurons should be less than twice the size of the input layer.

Summary

- The term “**feedforward**” refers to a method by which a neural network recognizes a pattern.
- The term “backpropagation” describes a process by which the neural network will be **trained**.

Summary (Cont.)

- A feedforward neural network is a network in which neurons are only connected to the next layer.
- There are no connections between neurons in previous layers or between neurons and themselves.
- Additionally, neurons are not connected to neurons beyond the next layer.
- As a pattern is processed by a feedforward design, the thresholds and connection weights will be applied.

Summary (Cont.)

- Neural networks can be trained using backpropagation.
- Backpropagation is a form of supervised training.
- The neural network is presented with the training data, and the results from the neural network are compared with the expected results.
- The difference between the actual results and the expected results is the error.
- Backpropagation is a method whereby the weights and input threshold of the neural network are altered in a way that causes this error to be reduced.