What is a Neural Network?

A neural network, more formally known as an artificial neural network, is a machine learning model which based on the structure and functioning of biological neural networks. A neural network can be developed through supervised learning, unsupervised learning, and reinforcement learning. Typically, neural networks are organized in a number of layers which are classified as the *Input Layer*, *Hidden Layer(s)*, and *Output Layer*. Each of these layers is described below:

1. **Input Layer:** This is the layer that deals with the external environment to feed the input into the neural network model. Each independent attribute (variable) in the dataset which has an impact on the output is a neuron in the input layer.

2. **Hidden Layer(s):** This is the layer in which the inputs are processed. It is an intermediary layer between the input layer and output layer in which an activation function, such as the sigmoid function, is applied to process the data fed to the hidden layer from the previous layer. Theoretically, this is the layer which is responsible for extracting the meaning and required features from the input layer. It is important to mention that there can be more than one hidden layer in a neural network, depending on the requirements and complexity of the dataset. In each hidden layer, the number of neurons can vary. There is no definite formula (or method) known to determine the exact number of hidden layers or neurons per hidden layer in a neural network.

3. **Output Layer:** This is the layer where the processed data is transmitted from the neural network. The number of neurons in the output layer depend on the type of analysis that the neural network is performing and it can vary depending on the desired outcome from the neural network.

In a neural network, the communication between the various layers is carried out through connections between the neurons. Each connection has a weight assigned to it which is typically assigned randomly in the beginning but is then modified and *improved* through a learning rule. An example of a commonly used learning rule is the learning rule in which the weights are recalculated through the backward propagation of the error in the output value. A neural network in which backpropagation occurs is known as a backpropagation-al neural network (BPNN).

Artificial Neural Network (Image):
https://cdn-images-1.medium.com/max/1600/1*hczvrCYgU_JQt5sx-UGM1A.gif

As with other machine learning models, a neural network is first trained using a part of the dataset which is referred to as the training dataset. To train the neural network model, the weights are updated several times by applying a learning rule until the weights and outputs are accurate to a satisfactory level.

There are numerous applications of neural networks which include the recognition of patterns or discovery of regularities within a dataset. Generally, a neural network performs very well when the dataset is very diverse, complex, or involves a high number of variables. It is best suited for problems where the relationship between variables is vague and dynamic (non-linear) in nature.

What is Keras?

Keras is an open source machine learning library which has been written in Python. The library is widely used to model neural networks and enable experimentation on neural networks. Keras is commonly used as a wrapper to Tensorflow. It is highly popular because of it is modular, extensible, and user-friendly.

The library contains a number of implementations of all the basic building blocks of a neural network. These include the layers, activation functions, learning rules, optimizers, and objectives in addition to a number of utilities for handling text-based and image-based data. Through Keras, neural network models can be developed on the web, on a Java Virtual Machine (JVM), and smartphones (such as iOS and Android).

The benefit of using Keras is that it simplifies the process of building a neural network. Without Keras, individual components of the neural network have to be developed from scratch and then have to be adjusted according to the problem. Keras enables us to work directly on the neural network and hence making the code simpler, efficient, and cleaner to write, maintain, and troubleshoot.

Here are the advantages of using the Keras machine learning library:

1. Neural network models developed on Keras can be easily transformed into products that can be released on a range of platforms such as mobile and web.
2. There is support for a number of backend engines compatible with Keras which include the Tensorflow backend and CNTK backend.
3. It is developer-centric and not machine-centric. It is easy to use and flexible at the same time.
4. The research community and industry are widely using Keras and it is supported by key companies such as Google and Microsoft.

Should information about Keras features be added here? Or a tutorial about building a neural network with Keras.

Fraudulent Credit Card Transactions:

The problem of credit card fraud has been around since a long time but it has grown substantially over time. One primary reason for the growth of this