**1. What is the difference between Machine Learning and Deep Learning?**

[**Machine Learning**](https://intellipaat.com/blog/what-is-machine-learning/) forms a subset of Artificial Intelligence, where we use statistics and algorithms to train machines with data, thereby, helping them improve with experience.

[**Deep Learning**](https://intellipaat.com/blog/what-is-deep-learning/) is a part of Machine Learning, which involves mimicking the human brain in terms of structures called neurons, thereby, forming [**neural networks**](https://intellipaat.com/blog/tutorial/machine-learning-tutorial/neural-network-tutorial/).

### ****2. What is a perceptron?****

A perceptron is similar to the actual neuron in the human brain. It receives inputs from various entities and applies functions to these inputs, which transform them to be the output.

A perceptron is mainly used to perform binary classification where it sees an input, computes functions based on the weights of the input, and outputs the required transformation.

### ****3. Why is Fourier transform used in Deep Learning?****

Fourier transform is an effective package used for analyzing and managing large amounts of data present in a database. It can take in real-time array data and process it quickly. This ensures that high efficiency is maintained and also makes the model more open to processing a variety of signals.

### ****4. What are the steps involved in training a perception in Deep Learning?****

There are five main steps that determine the learning of a perceptron:

1. Initialize thresholds and weights
2. Provide inputs
3. Calculate outputs
4. Update weights in each step
5. Repeat steps 2 to 4

**5. What are autoencoders?**

Autoencoders are artificial neural networks that learn without any supervision. Here, these networks have the ability to automatically learn by mapping the inputs to the corresponding outputs.

Autoencoders, as the name suggests, consist of two entities:

* Encoder: Used to fit the input into an internal computation state
* Decoder: Used to convert the computational state back into the output