OCTOBER 2019 / INSEM (T1) Solution

F. Y. M. TECH. (Computer Engineering) (SEMESTER - I)

COURSE NAME: Machine Learning

COURSE CODE: CSPA11183A

(PATTERN 2018:R1)

Paper Code - P119-133 (T1)

SOLUTION OF PAPER

Q.1) a) Define Machine Learning? And give its applications? [6 marks]

Answer:

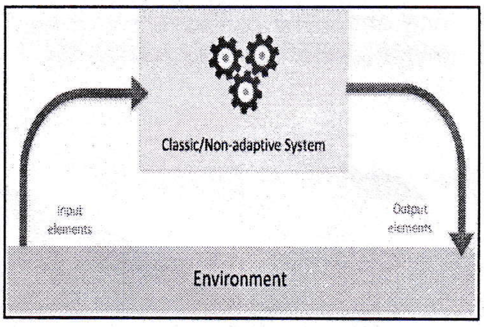
Definition of ML: Machine learning (ML) equips computers to learn and interpret without being explicitly programmed to do so. Here, as the "computers", also referred as the "models", are exposed to sets of new data, they adapt independently and learn from earlier computations to interpret available data and identify hidden patterns. This involves data analysis and automation of analytical model-building using numerous ML algorithms. ML enables computers and computing machines to search for and identify hidden insights, without being programmed for where to look for, when exposed to new data sets.

Applications of Machine Learning: The value of machine learning technology has been recognized by companies across several industries that deal with huge volumes of data. By leveraging insights obtained from this data, companies are able work in an efficient manner to control costs as well as get an edge over their competitors. This is how some sectors / domains are implementing machine learning -

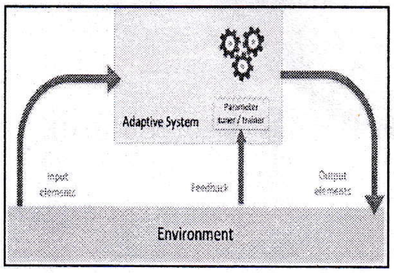
- Financial Services
- Marketing and Sales
- Government
- Healthcare
- Transportation
- Oil and Gas etc....

Q.1) b) Difference between classic and adaptive machines? [4 marks]

In the following figure, there's a generic representation of a classical system that receives some input values, processes them, and produces output results:



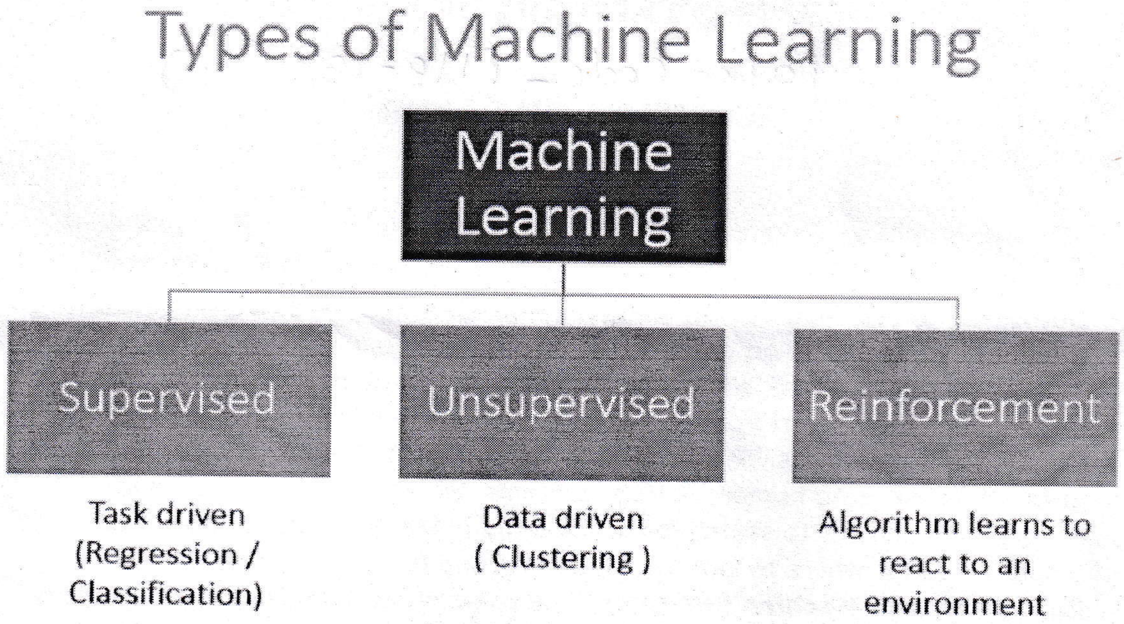
Here's a schematic representation of an adaptive system:



Solution

Q.2) Explain in detail with example? Different types of machine learning algorithms? [10 marks]

Answer:



Types of Machine Learning

Machine Learning

Supervised

Task driven (Regression / Classification)

Unsupervised

Data driven (Clustering)

Reinforcement

Algorithm learns to react to an environment

And explanation of all.

1. **Supervised Learning: *"The outcome or output for the given input is known before itself"*** and the machine must be able to map or assign the given input to the output. Multiple images of a cat, dog, orange, apple etc here the images are labelled. It is fed into the machine for training and the machine must identify the same. Just like a human child is shown a cat and told so, when it sees a completely different cat among others still identifies it as a cat, the same method is employed here.
2. **Unsupervised Learning:** *"The outcome or output for the given inputs is unknown",* here input data is given and the model is run on it. The image or the input given are grouped together here and insights on the inputs can be found here(which is the most of the real world data available). The main algorithms include *Clustering algorithms( ) and* learning algorithms.
3. **Reinforced Learning:** The machine is exposed to an *environment where it gets trained by trial and error method*, here it is trained to make a much specific decision. The machine learns from past experience and tries to capture the best possible knowledge to make *accurate decisions* based on the feedback received.

Q.3) a) what is Perceptron and Multilayer Perceptron? [6 marks]

Answer:

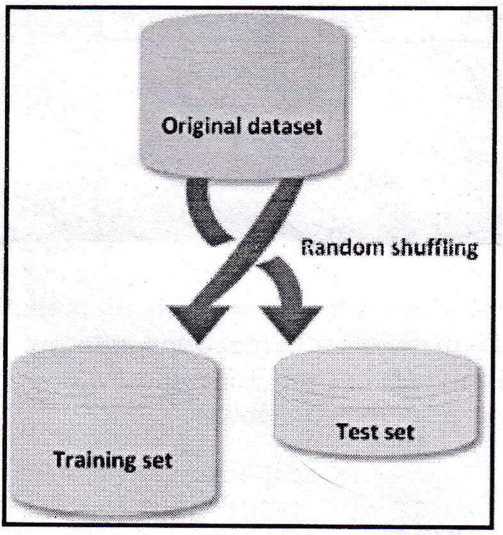
What Is a Perceptron?

A perceptron is a simple binary classification algorithm, proposed by Cornell scientist Frank Rosenblatt. It helps to divide a set of input signals into two parts—"yes" and "no". But unlike many other classification algorithms, the perceptron was usually between 0 and 1 or between -1 and 1. This allows for probability-based predictions or classification of items into multiple labels.

b) How to create training and test sets? [4 marks]

Answer:

When a dataset is large enough, it's a good practice to split it into training and test sets; the former to be used for training the model and the latter to test its performances. In the following figure, there's a schematic representation of this process:



There are two main rules in performing such an operation: Both datasets must reflect the original distribution, The original dataset must be randomly shuffled before the split phase in order to avoid a correlation between consequent elements. With scikit-learn, this can be achieved using the train_test_split() function:

```
from sklearn.model_selection import train_test_split
>>> X_train, X_test, Y_train, Y_test = train_test_split(X, Y,
test_size=0.25, random_state=1000)
```

The parameter test_size (as well as training_size) allows specifying the percentage of elements to put into the test/training set. In this case, the ratio is 75 percent for training and 25 percent for the test phase. Another important parameter is random_state which can accept a NumPy RandomState generator or an integer seed. In many cases, it's important to provide reproducibility for the experiments, so it's also necessary to avoid using different seeds and, consequently, different random splits

OR

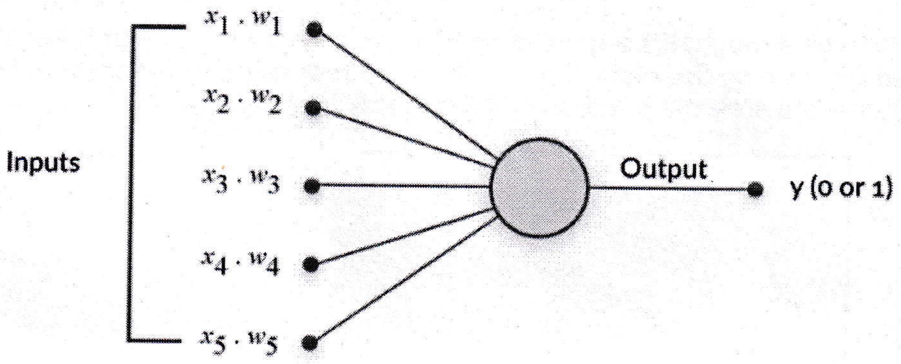
Q.4) How Does Back-Propagation in Artificial Neural Networks Work? [10 marks]

Answer:

Step 1: Explanation of How NN works:

modeled after the essential unit of the human brain—the neuron and has an uncanny ability to learn and solve complex problems.

Perceptron Input And Output

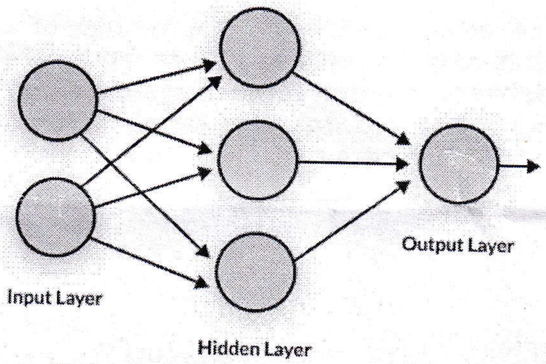


A perceptron is a very simple learning machine. It can take in a few inputs, each of which has a weight to signify how important it is, and generate an output decision of "0" or "1". However, when combined with many other perceptrons, it forms an **artificial neural network**. A neural network can, theoretically, answer any question, given enough training data and computing power.

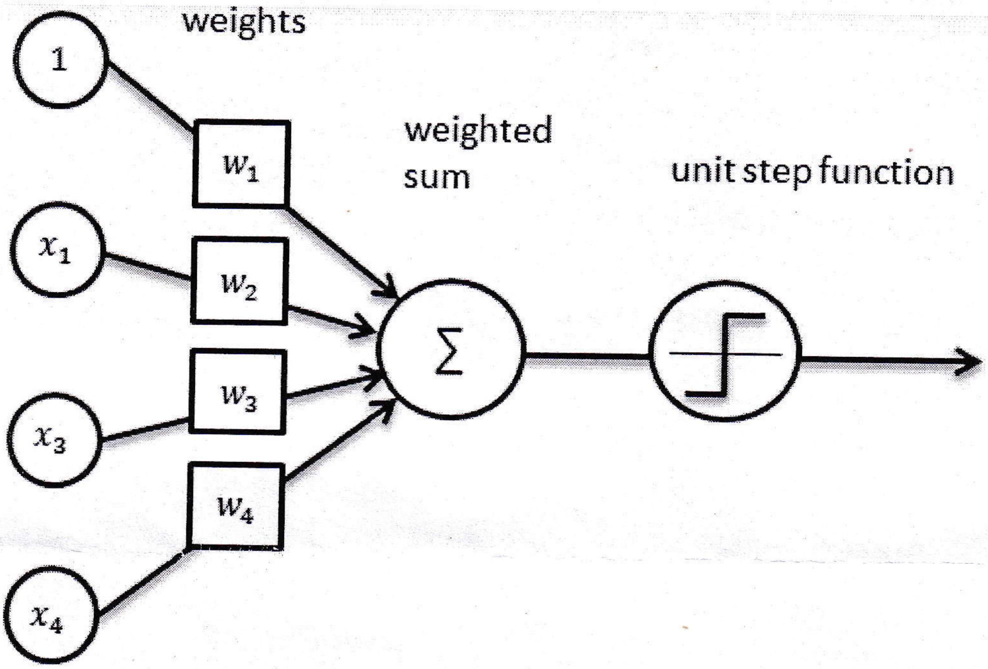
What is a Multilayer Perceptron?

A multilayer perceptron (MLP) is a perceptron that teams up with additional perceptrons, stacked in several layers, to solve complex problems. The diagram below shows an MLP with three layers. Each perceptron in the first layer on the left (the input layer), sends outputs to all the perceptrons in the second layer (the hidden layer), and all perceptrons in the second layer send outputs to the final layer on the right (the output layer).

Perceptron Input And Output



Each perceptron sends multiple signals, one signal going to each perceptron in the next layer. For each signal, the perceptron uses **different weights**. In the diagram above, every line going from a perceptron in one layer to the next layer represents a **different output**. Each layer can have a large number of perceptrons, and there can be multiple layers, so the multilayer perceptron can quickly become a very complex system. The multilayer perceptron has another, more common name—a **neural network**. A three-layer MLP, like the diagram above, is called a **Non-Deep** or **Shallow Neural Network**. An MLP with four or more layers is called a **Deep Neural Network**. One difference between an MLP and a neural network is that in the classic perceptron, the decision function is a step function and the output is binary. In neural networks that evolved from MLPs, **other** activation functions **can be used** which result in outputs of real values,



Step 2: Activation Functions
Step 3: Cost Function
Step 4: Forward Propagation Calculation
Step 5: Back-propagation Algorithm

****