TRAINING SESSION ON FUNDAMENTALS OF DEEP LEARNING

Presented by:

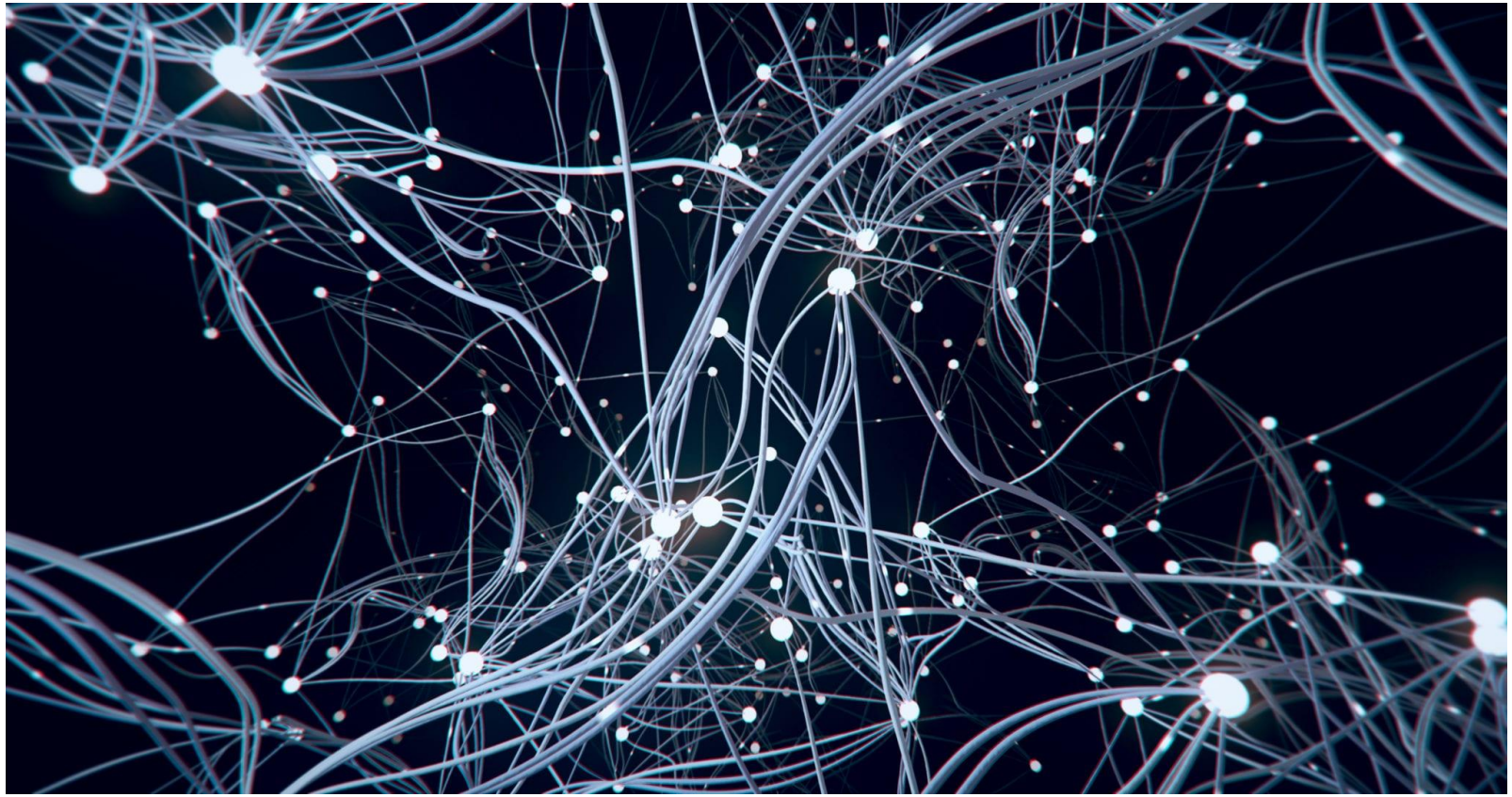
Dr. Bunil Kumar Balabantaray
Assistant Professor

Department of Computer Science and Engineering,
NATIONAL INSTITUTE OF TECHNOLOGY MEGHALAYA

Anmol Gautam, Swastik Jena (Member of AI Research Group,NIT Meghalaya)

AGENDA

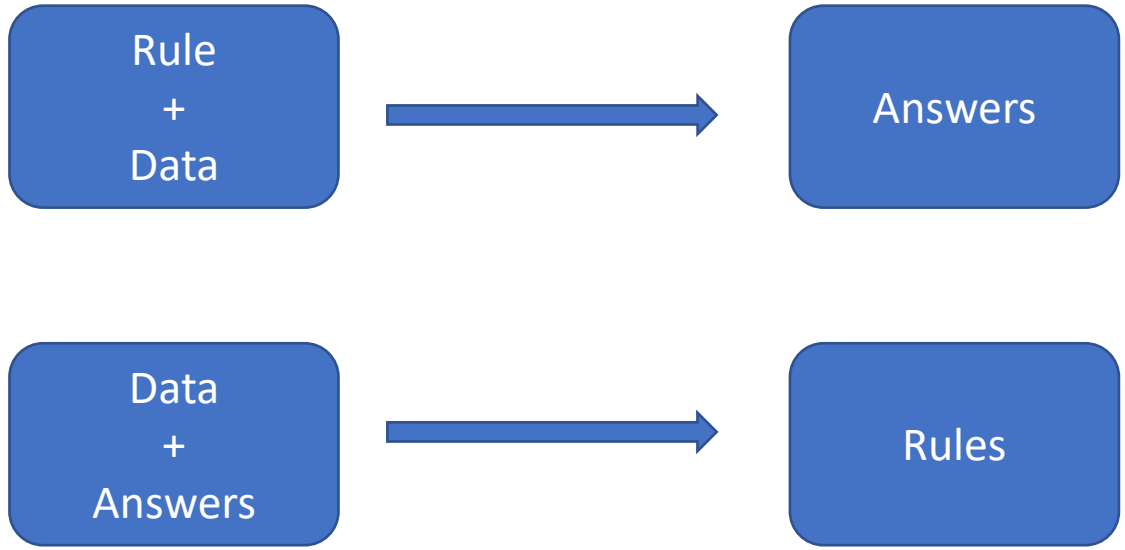
- INTRODUCTION
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION



AGENDA

- **INTRODUCTION**
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION

- Deep Learning is a programming paradigm
- Extract and transform data
- Training vs being explicitly programmed
- Learn by experience



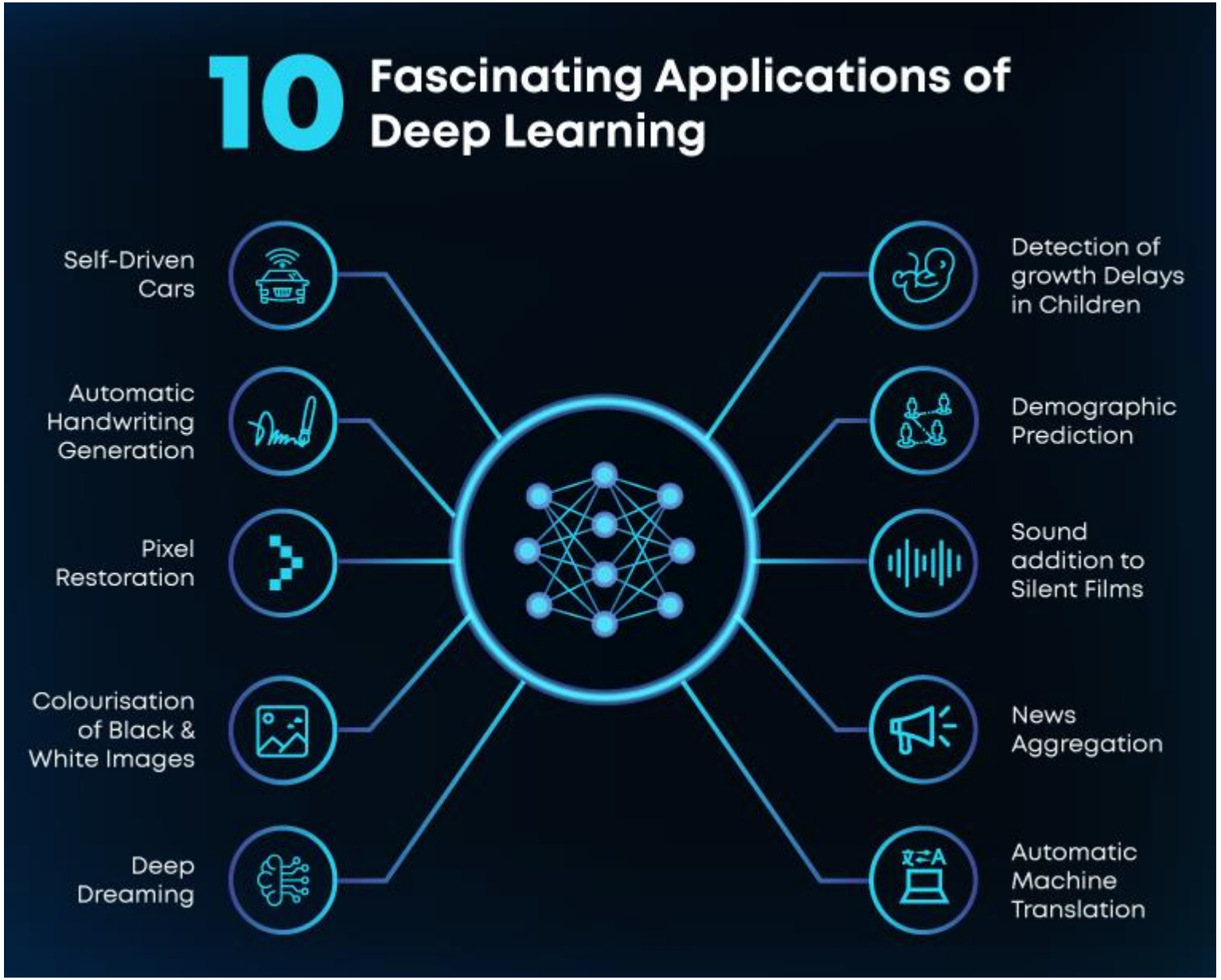
AGENDA

- **INTRODUCTION**
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION

- Therefore, ML model is trained and explicitly programmed.
- Learning is that part where based on inputs and expected output we tune the model i.e. update the parameters.
- Ultimately, we are transforming data to derive meaningful representations
- This meaningful representation comes from a finites set of possible solutions called hypothesis space.

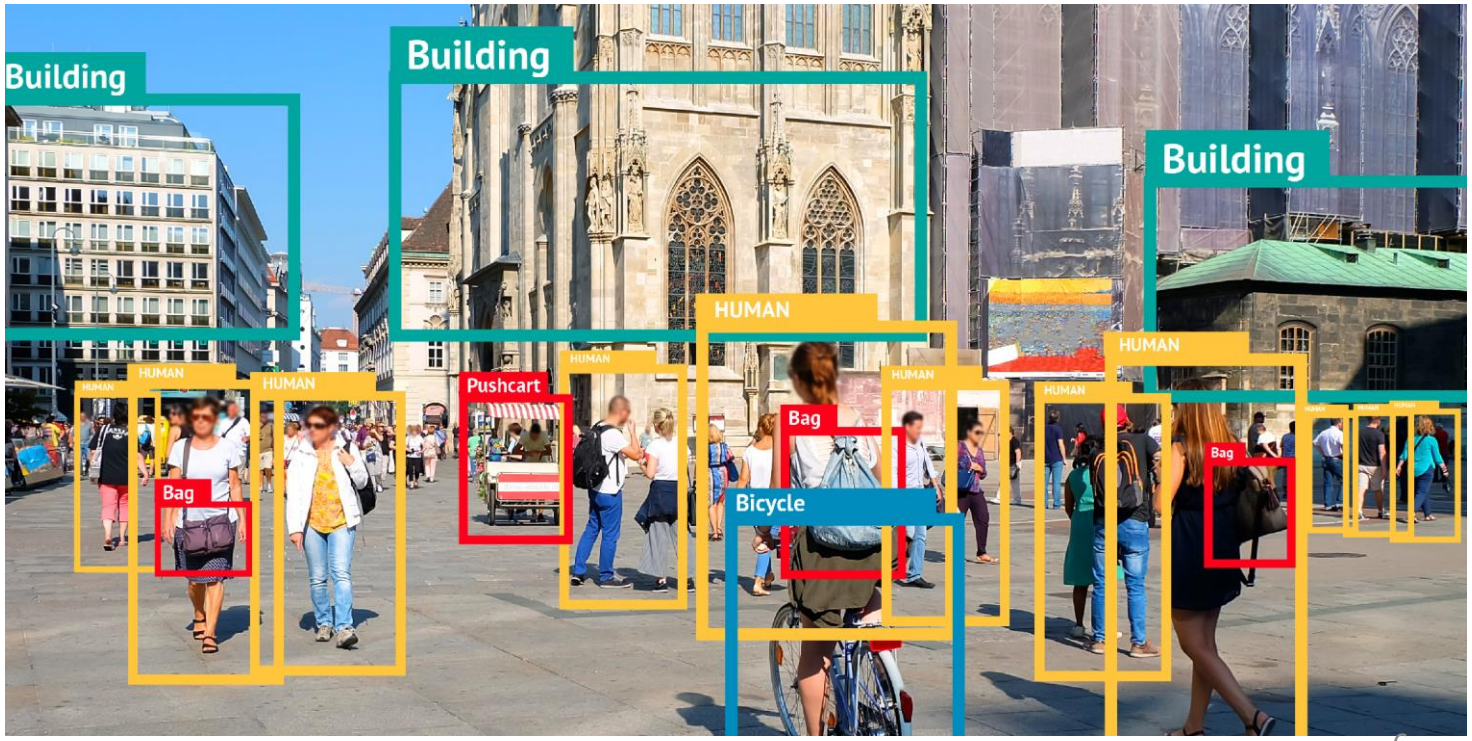
AGENDA

- **INTRODUCTION**
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION



AGENDA

- **INTRODUCTION**
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION

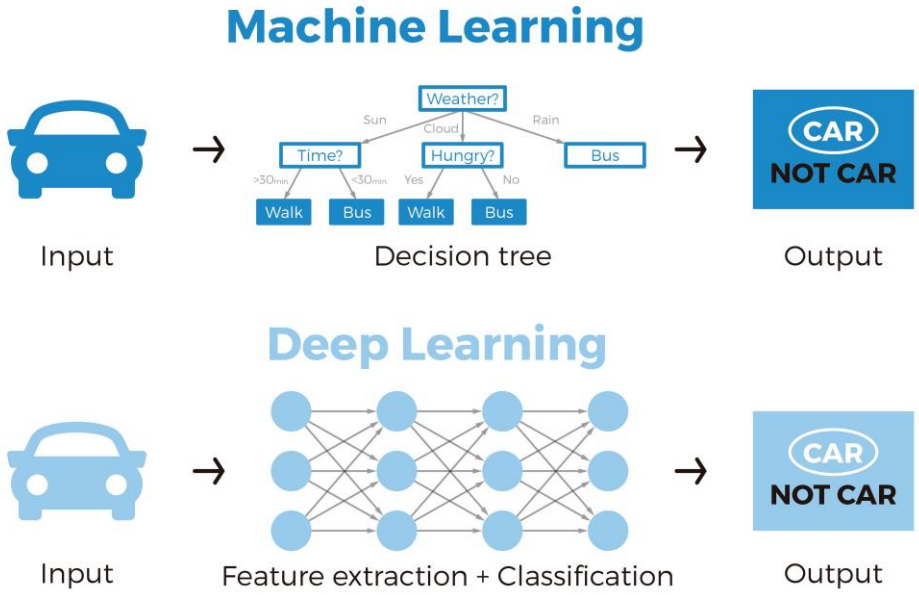


AGENDA

- INTRODUCTION
- **DEFINITION**
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION

- **Machine learning** is the science of getting computers to act without being explicitly programmed.

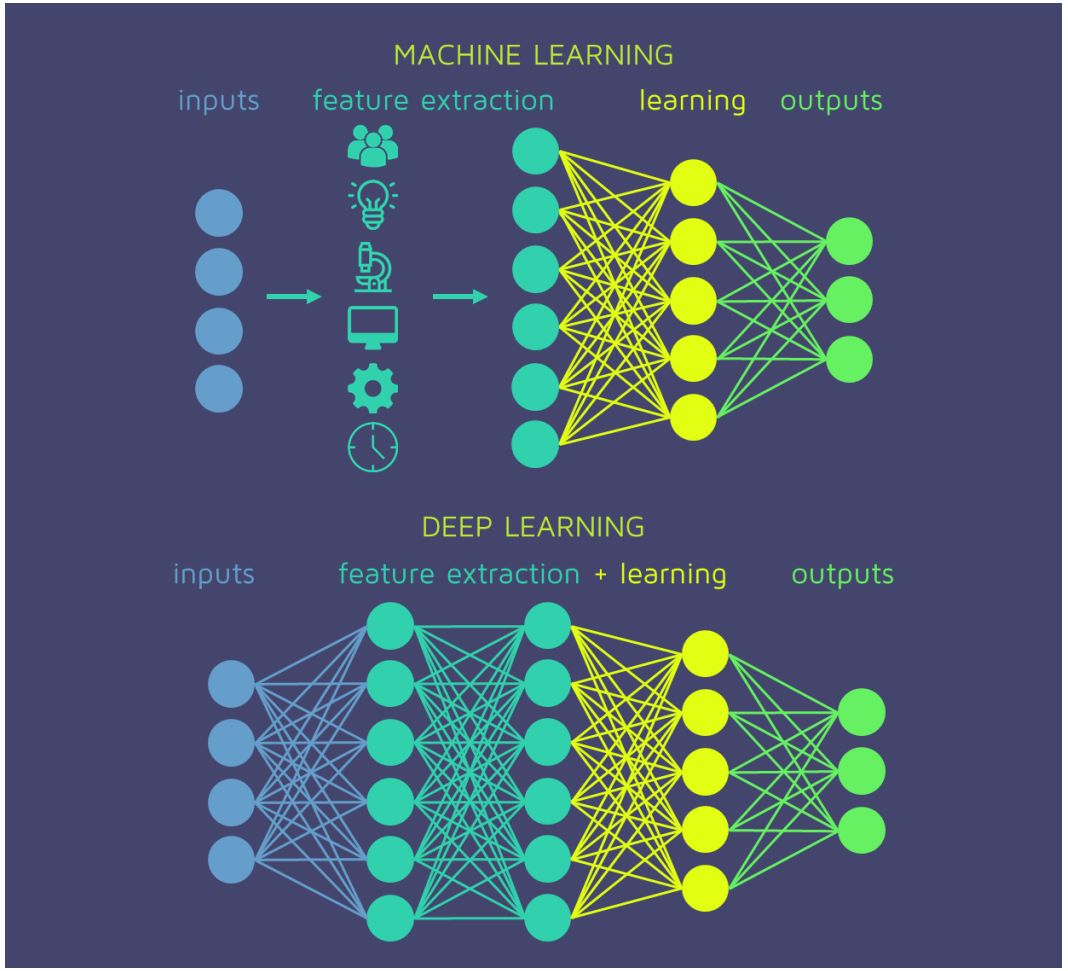
- **Deep learning** is a machine learning technique that teaches computers to do what comes naturally to humans: learn by example.



AGENDA

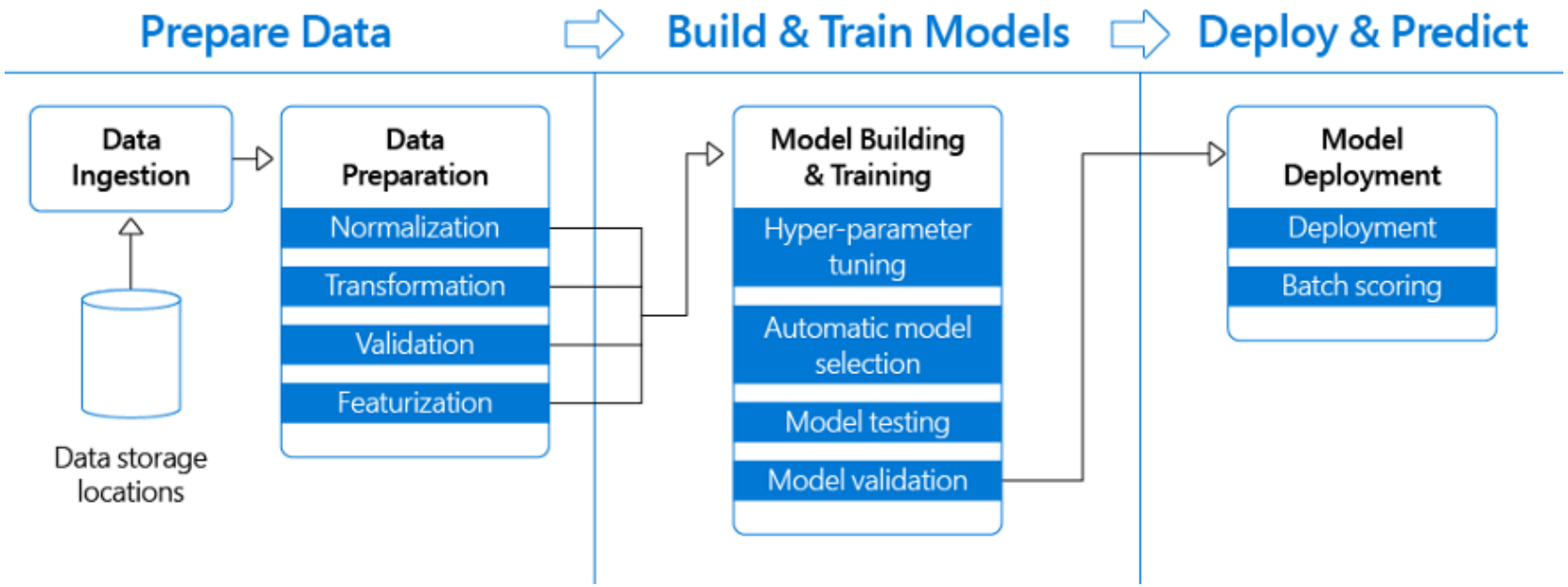
- INTRODUCTION
- **DEFINITION**
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION

- Deep learning requires large amounts of labeled data
- Deep learning requires substantial computing power



AGENDA

- INTRODUCTION
- **DEFINITION**
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION



AGENDA

- INTRODUCTION
- DEFINITION
- **HISTORY**
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION

- Arthur Samuel
- Worked on a different way to get computers to do a job.

"Programming a computer for such computations is, at best, a difficult task, not primarily because of any inherent complexity in the computer itself but, rather, because of the need to spell out every minute step of the process in the most exasperating detail. Computers, as any programmer will tell you, are giant morons, not giant brains"

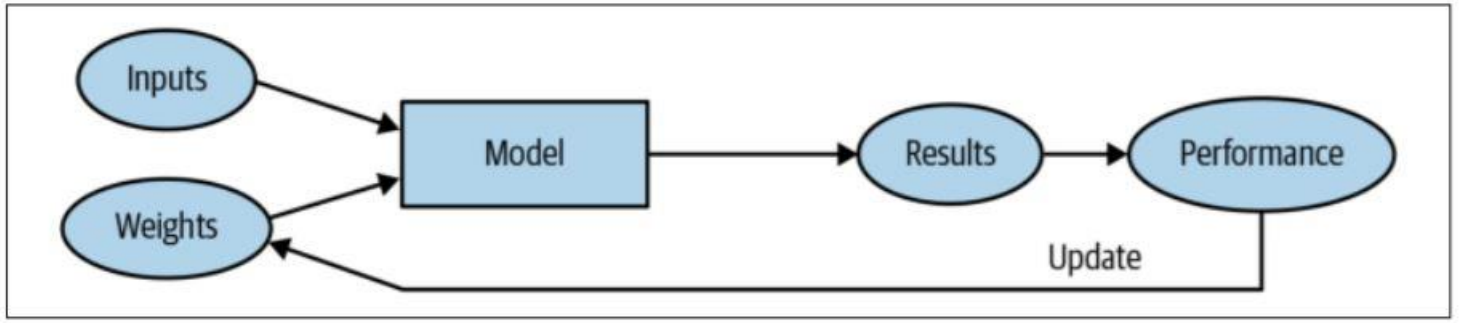
AGENDA

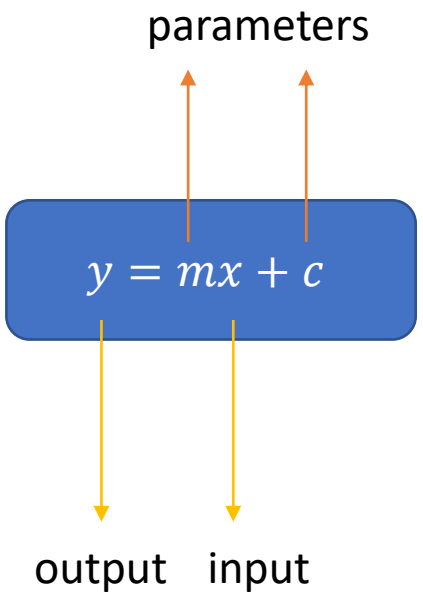
- INTRODUCTION
- DEFINITION
- **HISTORY**
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION

"Suppose we arrange for some automatic means of testing the effectiveness of any current weight assignment in terms of actual performance and provide a mechanism for altering the weight assignment so as to maximize the performance. We need not go into the details of such a procedure to see that it could be made entirely automatic and to see that a machine so programmed would **learn** from its **experience**"

AGENDA

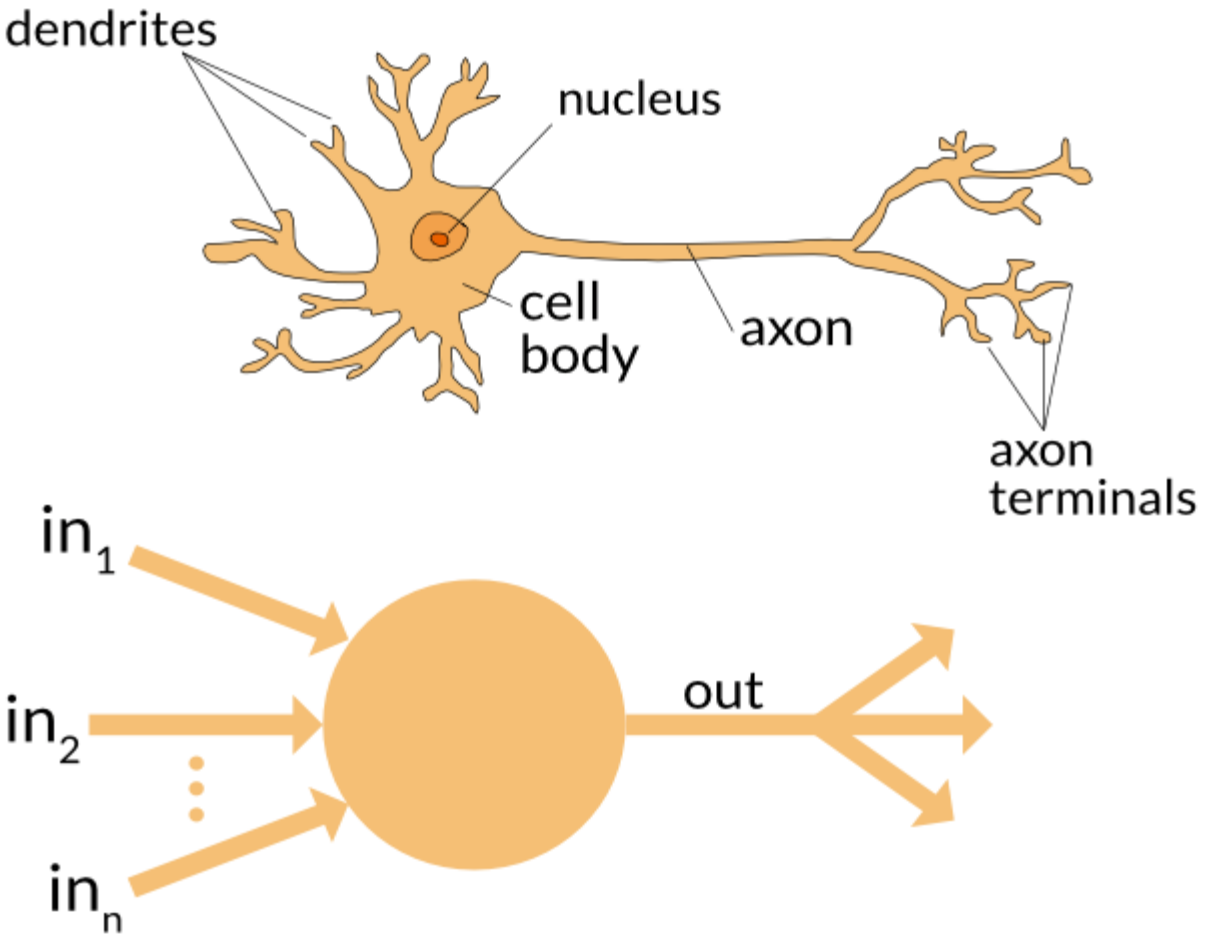
- INTRODUCTION
- DEFINITION
- **HISTORY**
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION





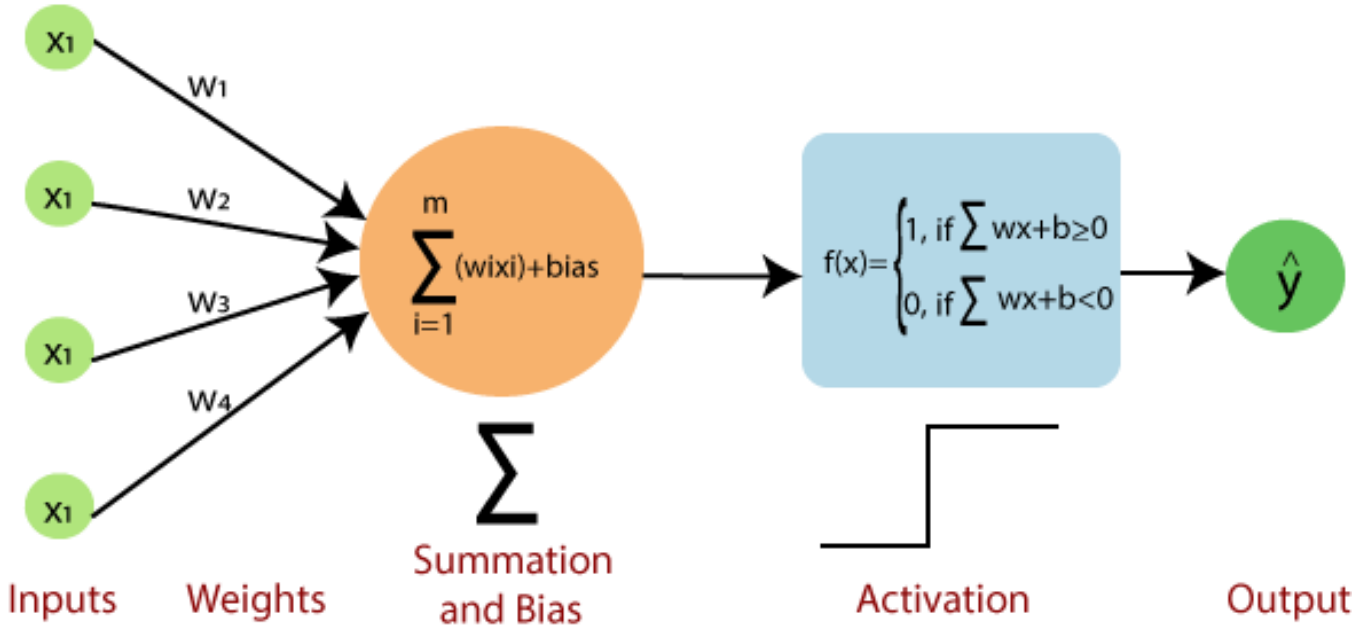
AGENDA

- INTRODUCTION
- DEFINITION
- HISTORY
- **PERCEPTRON MODEL**
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION



AGENDA

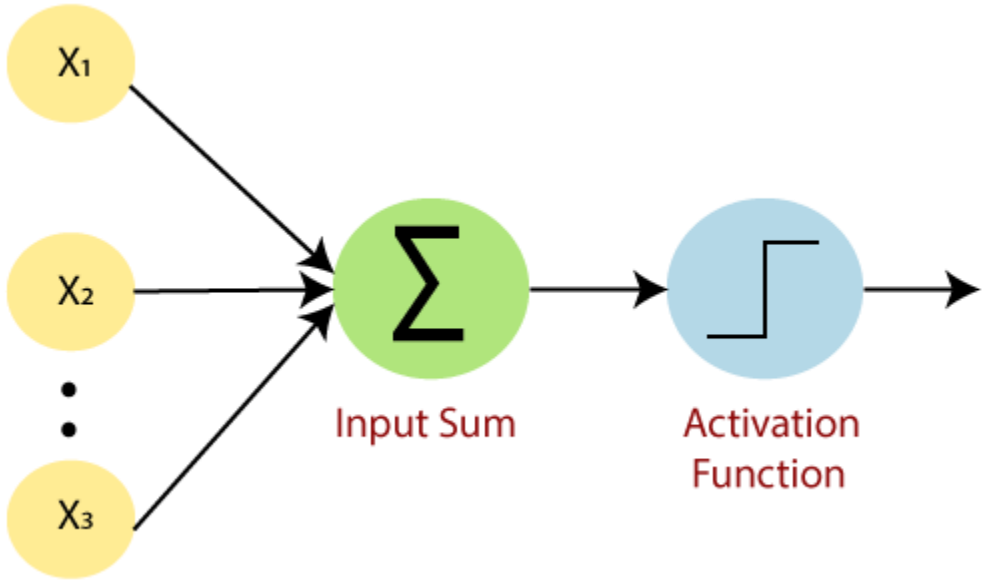
- INTRODUCTION
- DEFINITION
- HISTORY
- **PERCEPTRON MODEL**
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION



AGENDA

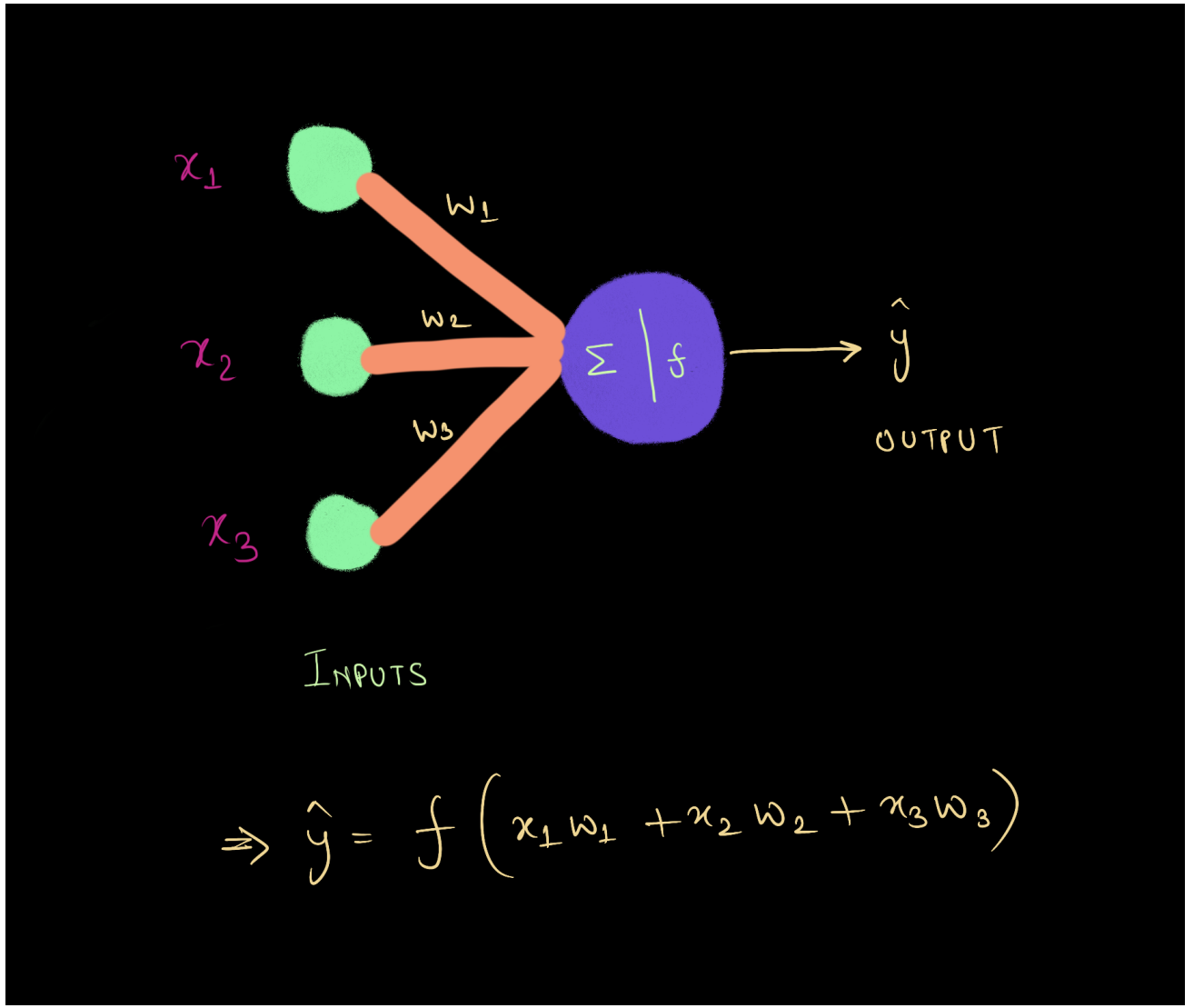
- INTRODUCTION
- DEFINITION
- HISTORY
- **PERCEPTRON MODEL**
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION

- **Weights** represents dimension or strength of the connection between units.
- **Bias** represents an additional parameter which task is to modify the output along with the weighted sum of the input to the other neuron
- **Activation Function** calculates a weighted sum and further adding bias with it to give the result.



AGENDA

- INTRODUCTION
- DEFINITION
- HISTORY
- **PERCEPTRON MODEL**
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION



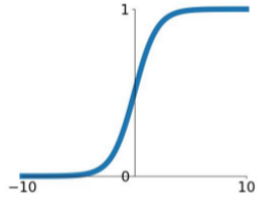
AGENDA

- INTRODUCTION
- DEFINITION
- HISTORY
- **PERCEPTRON MODEL**
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION

Activation Functions

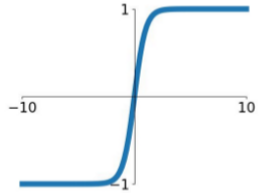
Sigmoid

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



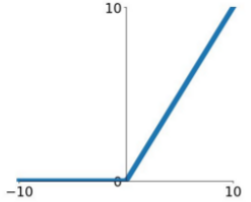
tanh

$\tanh(x)$



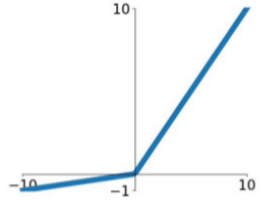
ReLU

$\max(0, x)$



Leaky ReLU

$\max(0.1x, x)$

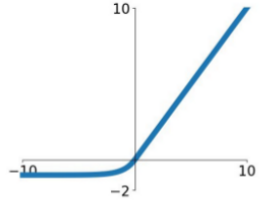


Maxout

$\max(w_1^T x + b_1, w_2^T x + b_2)$

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



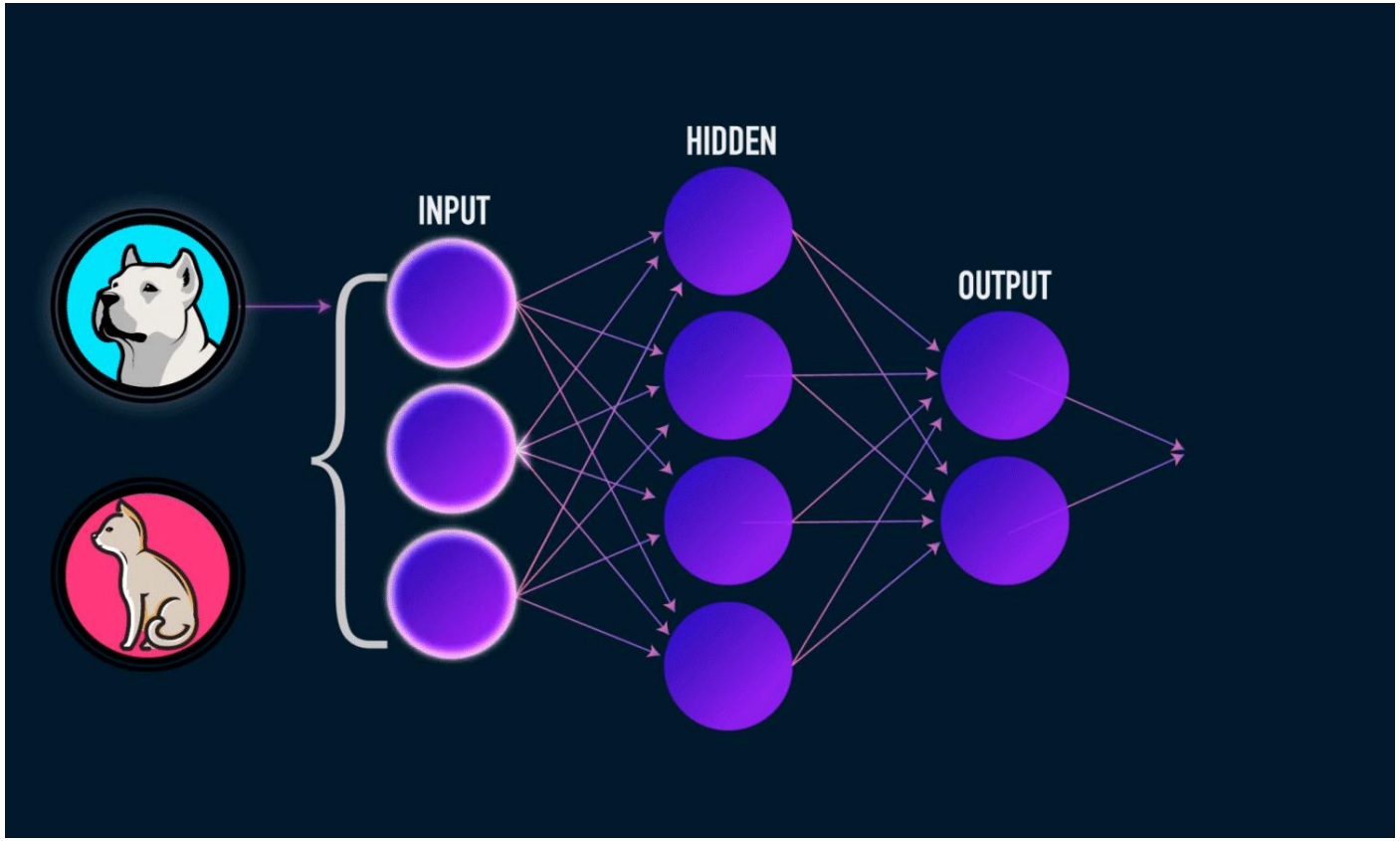
AGENDA

- INTRODUCTION
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- **FORWARD PROPOGATION**
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION

Computational Graph

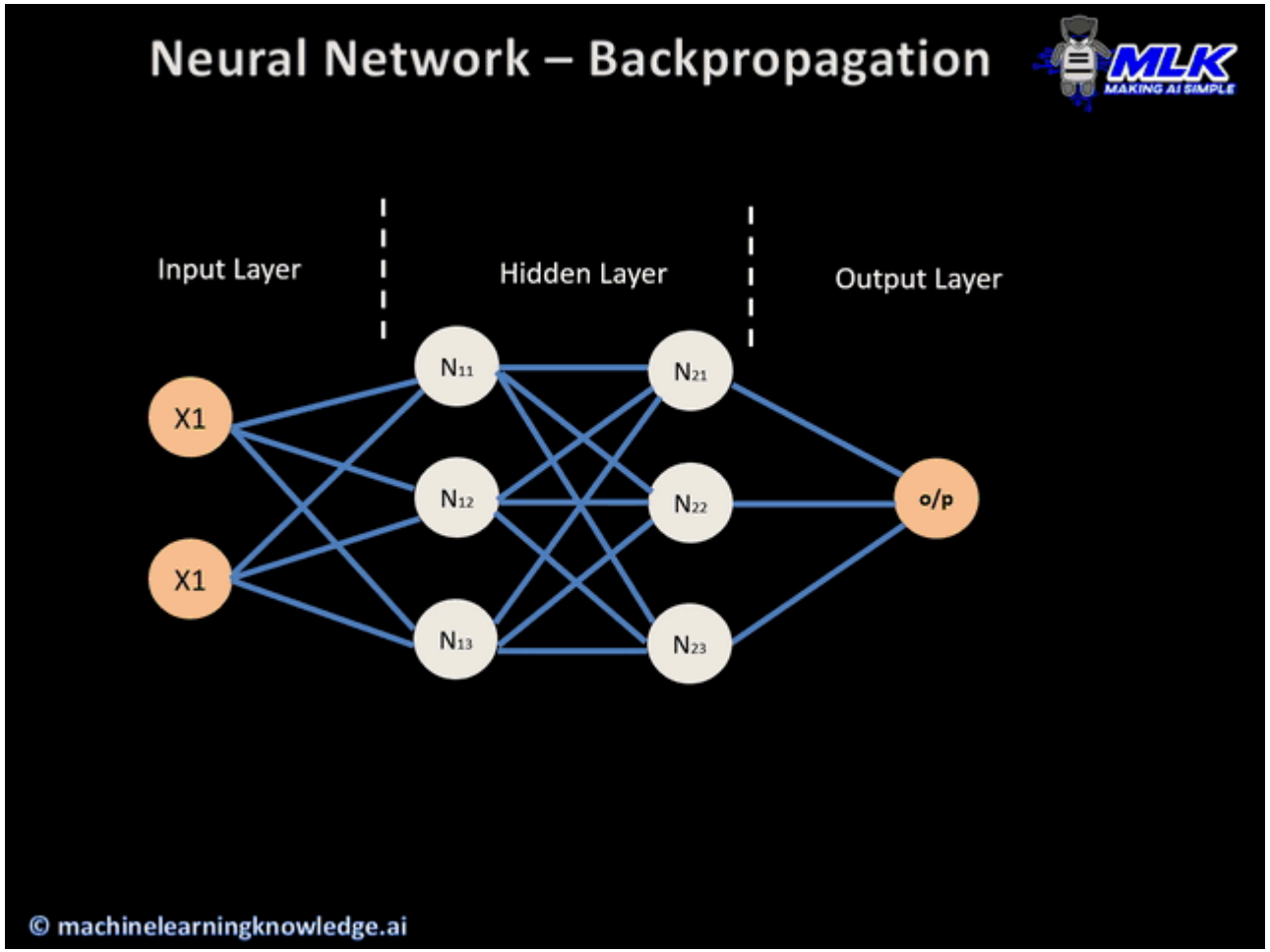
AGENDA

- INTRODUCTION
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- **FORWARD PROPOGATION**
- BACKWARD PROPOGATION
- LOSS
- OPTIMIZER
- HANDS ON SESSION



AGENDA

- INTRODUCTION
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- **BACKWARD PROPOGATION**
- LOSS
- OPTIMIZER
- HANDS ON SESSION



AGENDA

- INTRODUCTION
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- **LOSS**
- OPTIMIZER
- HANDS ON SESSION

This function will essentially calculate how poorly our model is performing by comparing what the model is predicting with the actual value it is supposed to output. If predicted output is very far off from actual output, the Loss value will be very high. However if both values are almost similar, the Loss value will be very low. Hence we need to keep a loss function which can penalize a model effectively while it is training on a dataset.

AGENDA

- INTRODUCTION
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- **OPTIMIZER**
- HANDS ON SESSION

- Goal is to minimize the loss function. Why?
- Well, because we want to get as accurate result as we can without memorizing the data.
- How to minimize?

MATHEMATICS

AGENDA

- INTRODUCTION
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- **OPTIMIZER**
- HANDS ON SESSION

- Lets see a story NOW……..

Friend A
Says
I have a function that I want to minimize such that I find those values for its variables
For which
This function gives
Minimum value

Friend B
Says
No worries Dude
If your function is differentiable
Then I have a tool that can tell you in which direction you can move to
Get these minimum values.

AGENDA

- INTRODUCTION
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- **OPTIMIZER**
- HANDS ON SESSION

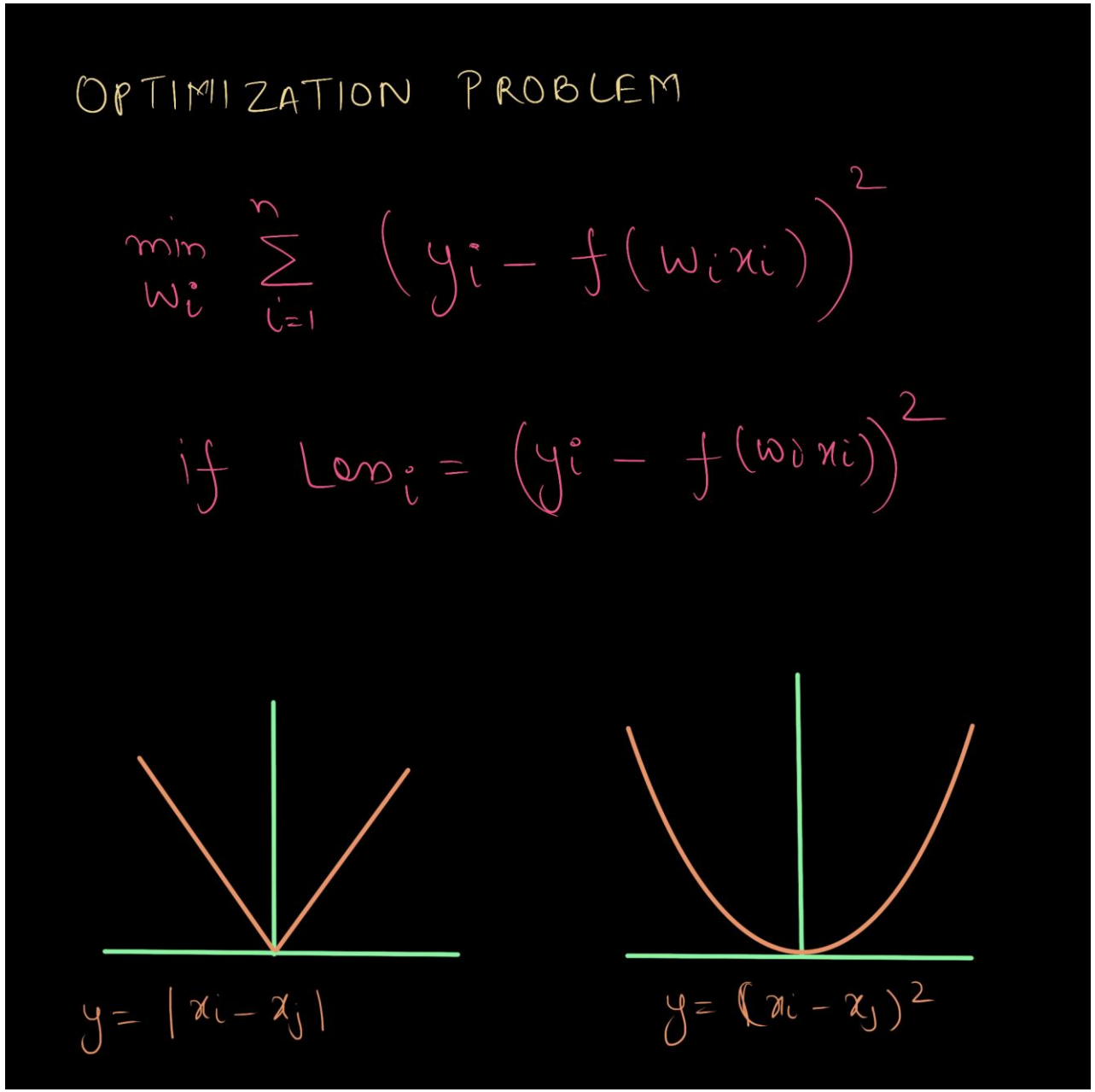
- Lets see a story NOW……..

LOSS Function

Gradient Descent

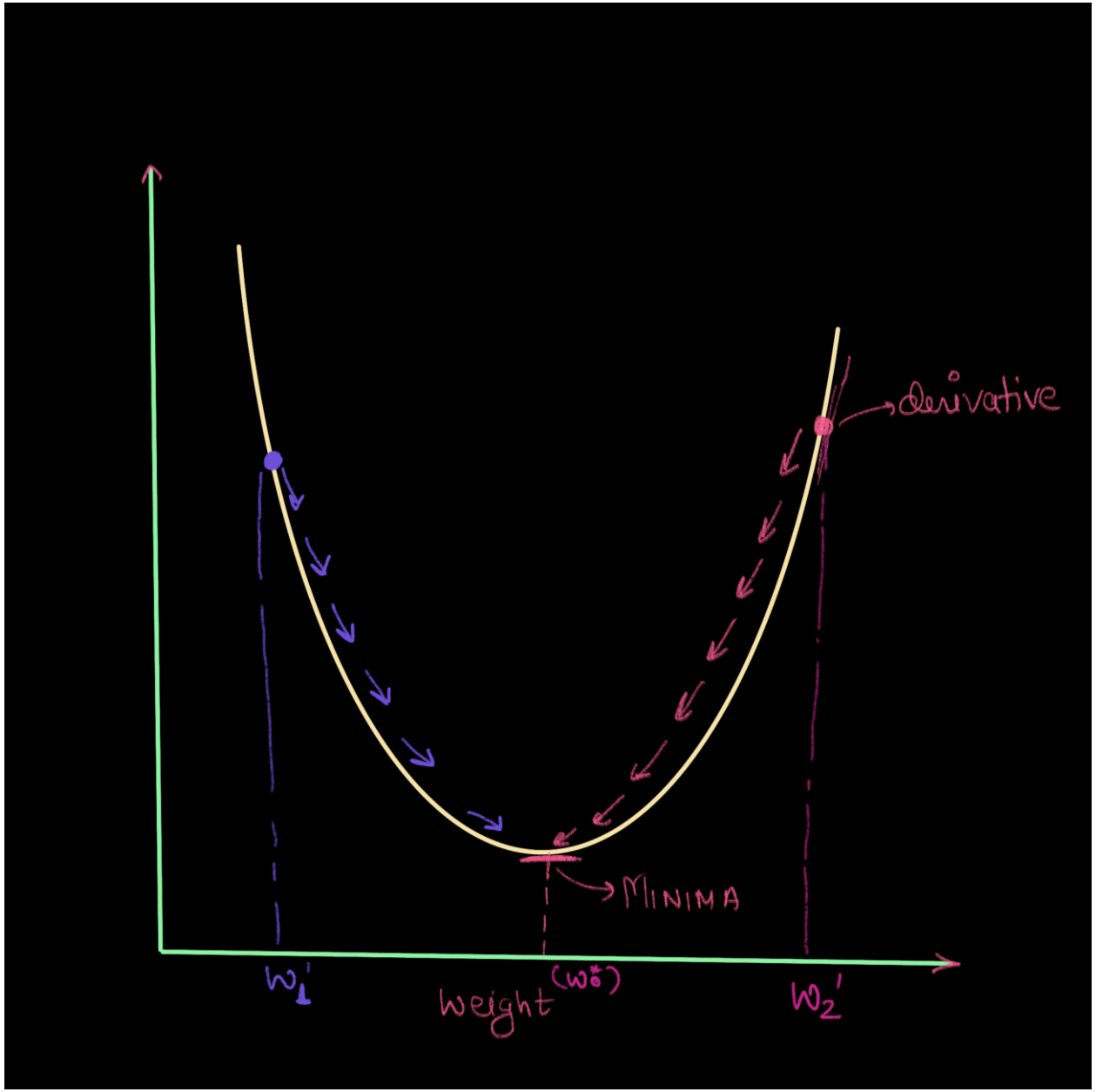
AGENDA

- INTRODUCTION
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- **OPTIMIZER**
- HANDS ON SESSION



AGENDA

- INTRODUCTION
- DEFINITION
- HISTORY
- PERCEPTRON MODEL
- FORWARD PROPOGATION
- BACKWARD PROPOGATION
- LOSS
- **OPTIMIZER**
- HANDS ON SESSION



THANK YOU….!