Artificial Neural Networks

Dr. Muhammad Aqib

University Institute of Information Technology

PMAS-Arid Agriculture University Rawalpindi

Introduction to Neural Networks

Contents

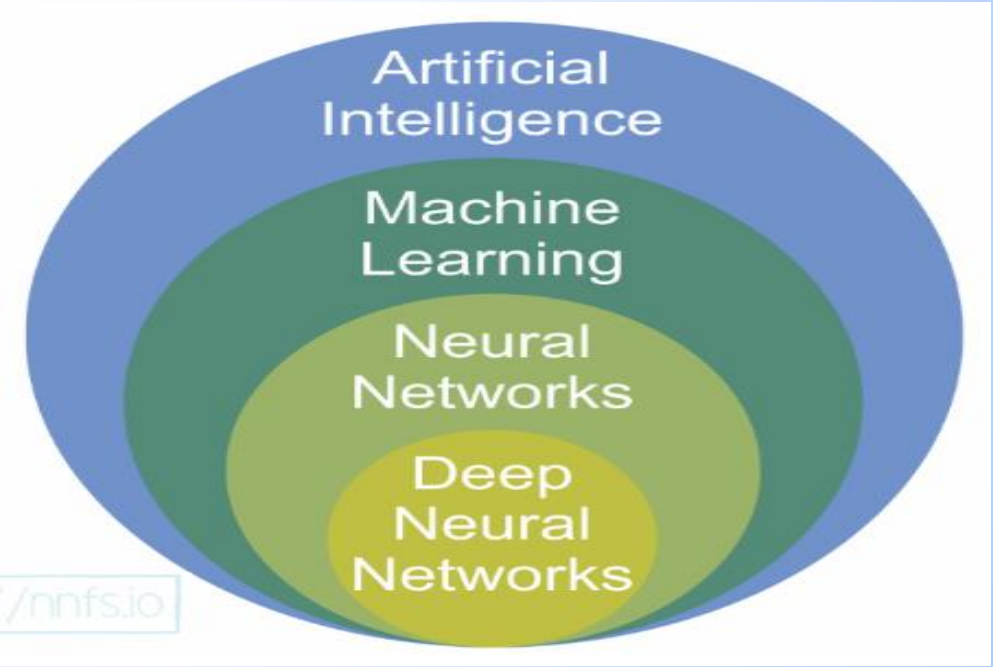
- Neural networks
- Brief history of neural networks
- Classification vs Regression
- Supervised vs Unsupervised learning
- Simple neural networks (diagrams / structure)
- Brief overview of layers
- Weights, biases and their impact
- Activation function
- Overfitting
- Generalization
- Loss
- Common tasks of neural networks

Neural Networks?

- Neural Networks (a.k.a Artificial Neural Networks) are type of machine learning with focus on imitating human brain.
- Also called as Deep Learning.
  - Deep Neural Networks comprise of two or more hidden layers
  - Hidden Layers are ones that Neural Network controls
- Most Neural Networks in use are form of Deep Learning.

Where Deep Learning Fits into?

- It can be seen that Neural Networks (NN) are Deeper Area of AI (subfield) focusing on brain imitation.



Brief History

- Since advent of computers, scientists have been trying to come up with machines that take input and output desired results for **Classification** and **Regression** problems.
- Classification problems are those in which inputs must be categorized into some predefined categories.
  - Like Cats/Dogs Classification – Where your input image must be classified into either Dog or Cat.
- Regression problems are those in which inputs are transformed into continuous outputs (used to answer questions like how much? how many?)
  - Stock Price Prediction – Where your inputs features are used to come up with stock price(s).

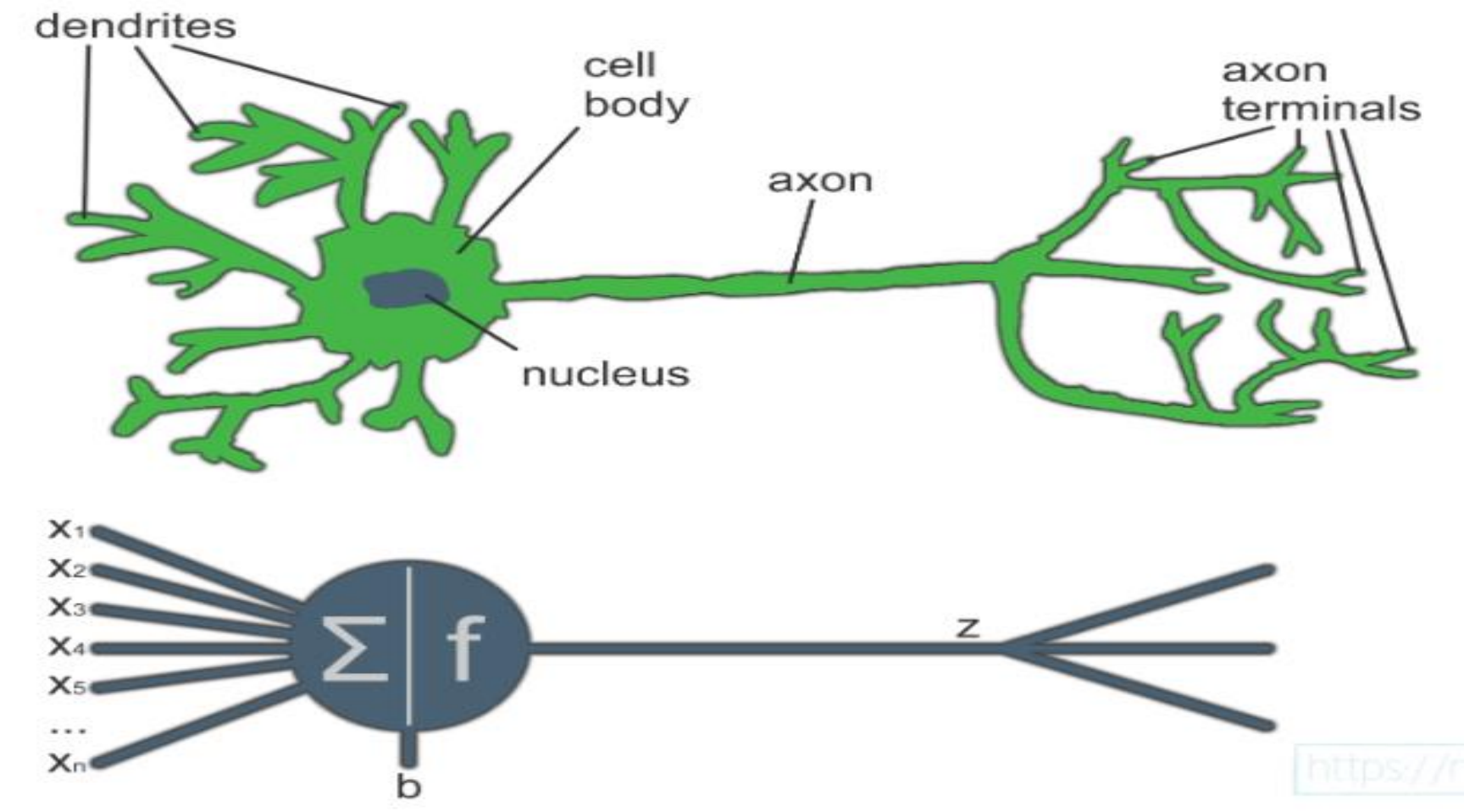
Brief History (cont.)

- In addition, there are ways of training / learning such as Supervised and **Unsupervised**.

- **Supervised ML:** When you have labeled data for training your algorithms.
  - X set of inputs with their Y outputs (labels)
  - X can also be called as feature
  - Labels are also referred as Targets / Ground-Truths

- **Unsupervised ML:** In this kind of approach, machine learning algorithms tends to find structure in data without knowing labels.

Neural Networks Revisited …



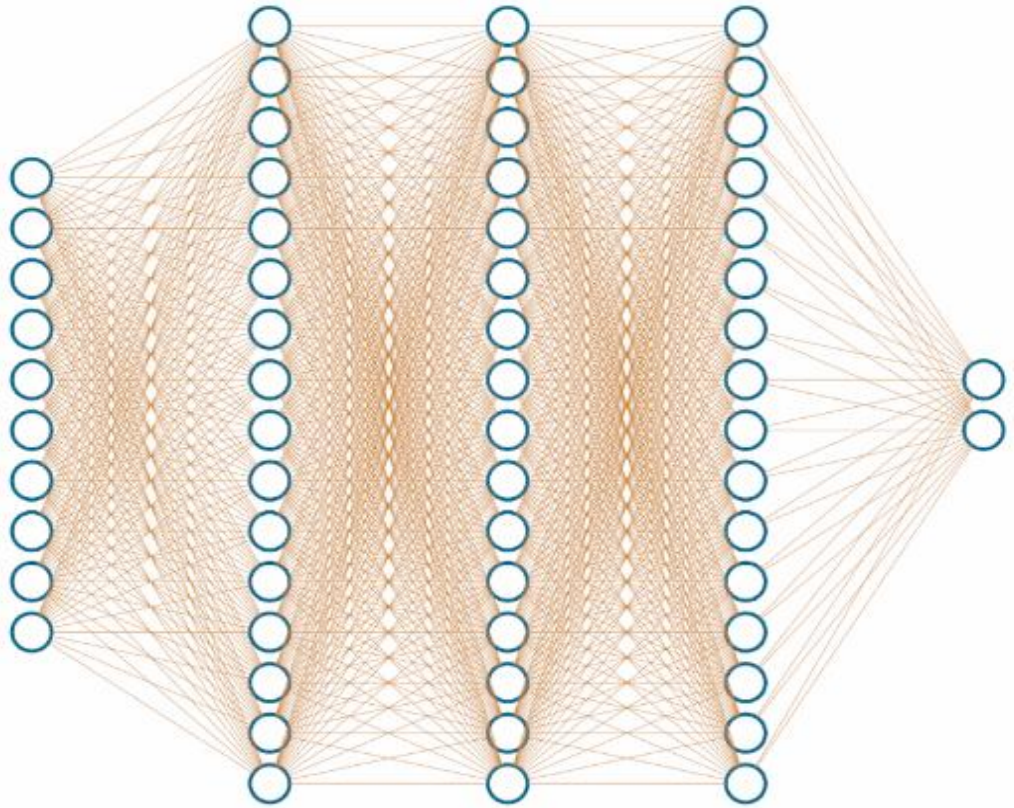
Neural Networks Revisited … (cont.)

- (Artificial) Neural Networks are inspired by human brain, translated to computer.
- There are neurons, activations, and lots of interconnectivity between both NN and human brain, even if underlying processes are quite different.
- Although single neuron itself is relatively useless
  - But … when combined with hundreds of thousands of other neurons, tremendous results can be achieved that outperform traditional Machine Learning methods.

Sample Neural Network

Layer sizes: 10, 16, 16, 16, 2

Weights:	704
Biases:	50
Params:	754



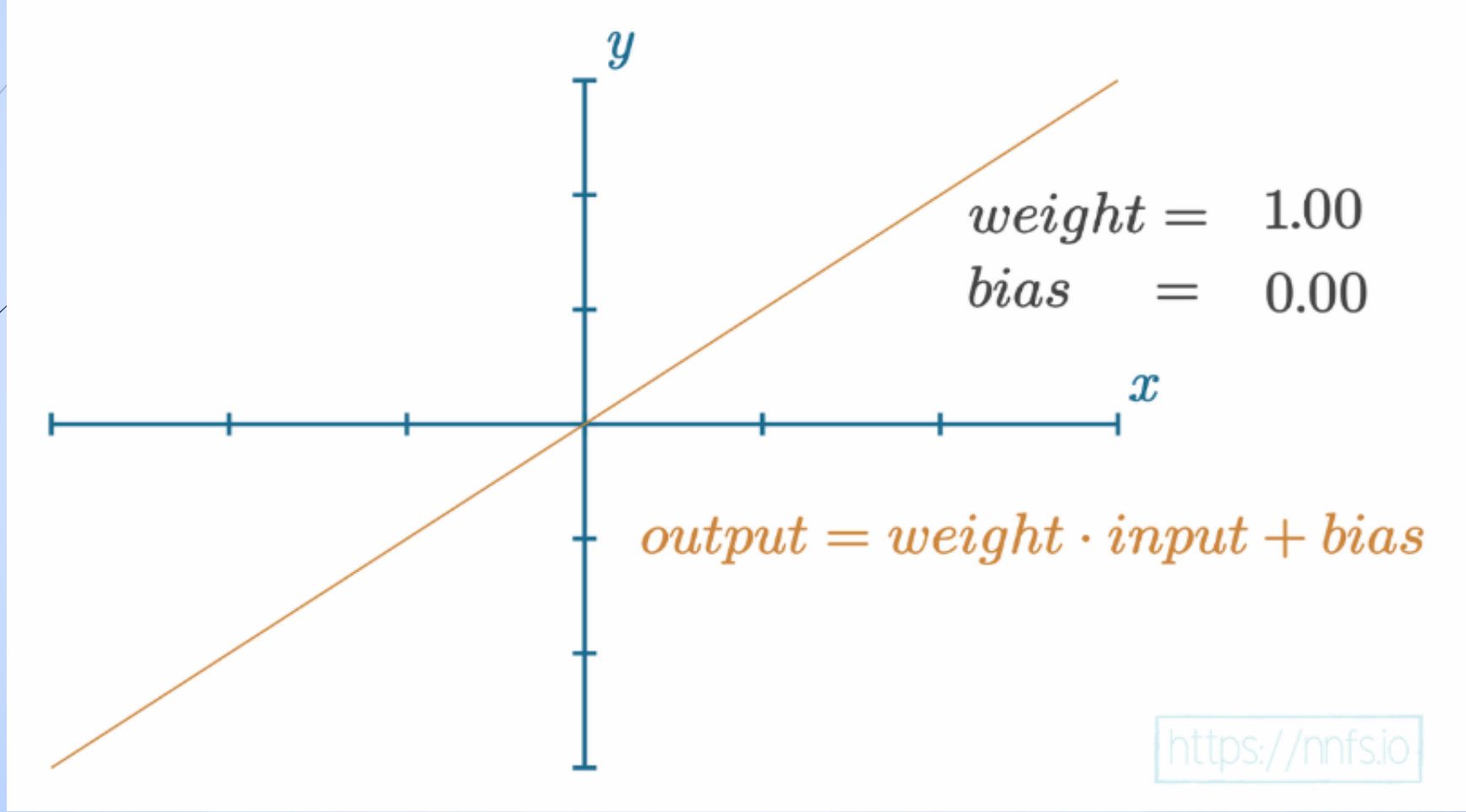
Neural Network – Dense Layer

- Dense Layers, consist of interconnected neurons.
- In Dense Layer, each neuron of given layer is connected to every neuron of next layer.
- In simple terms, output values of a layer becomes input to the next adjacent layer.
- Each connection between neurons has some weight associated.
- **Weights** are **trainable** factor; how much of this input to use?
- Weights get multiplied with Inputs and then summed up.
- **Bias** is another **trainable** factor, that is added afterwards.
- This whole process results in output of a single neuron.
  - output = sum(inputs * weights) + bias

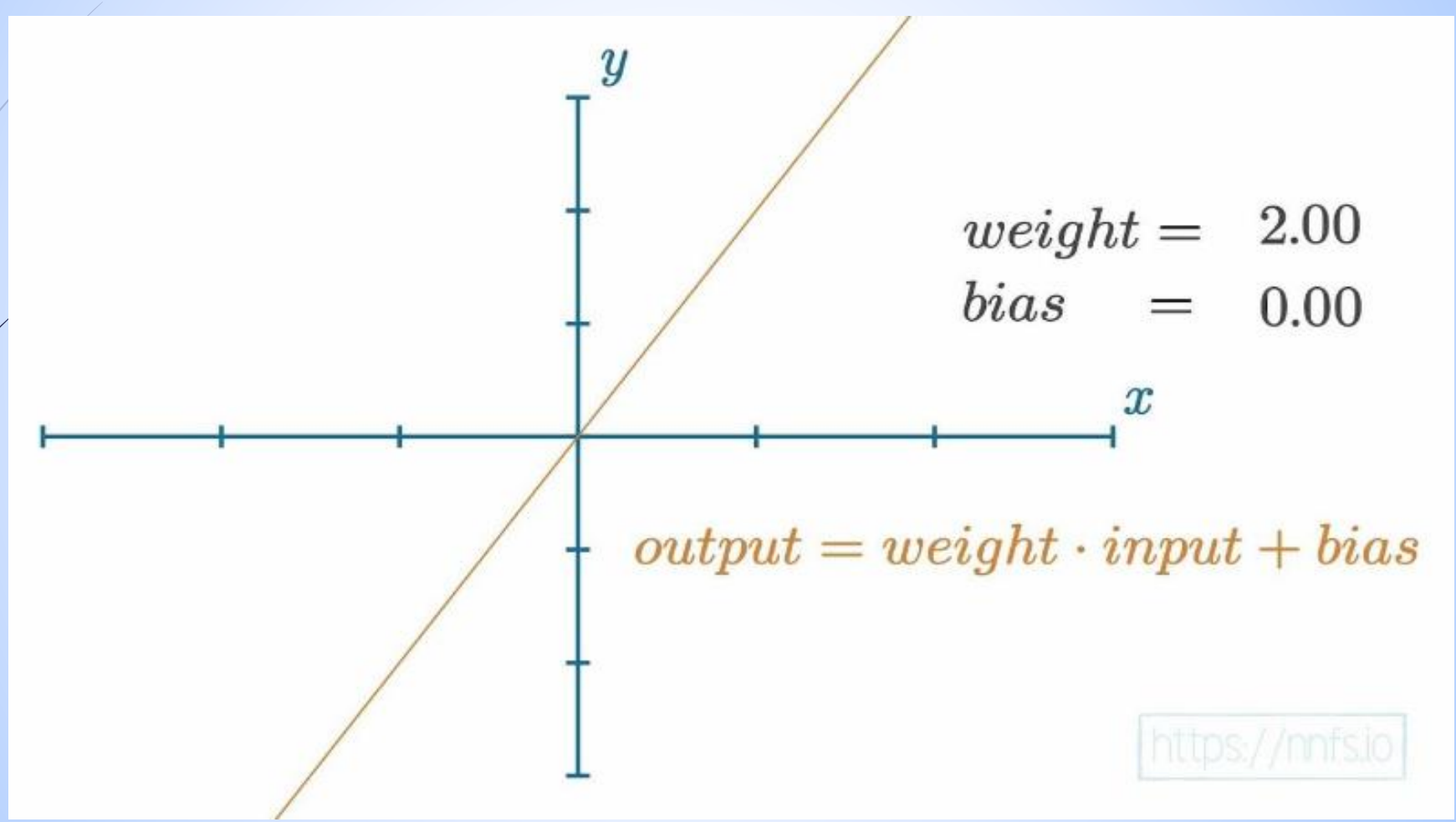
Impact of Weights / Bias

- Weights and Biases are kind of “knobs” that are tuned while training our models.
- Both of these impact neurons’ output, but in different ways …
- Both of these are tuned using **optimizers** during training.
- Adjusting weights will impact slope of output (function)
- Bias offsets the overall function
  - Confused? Let's check example on next slides :)

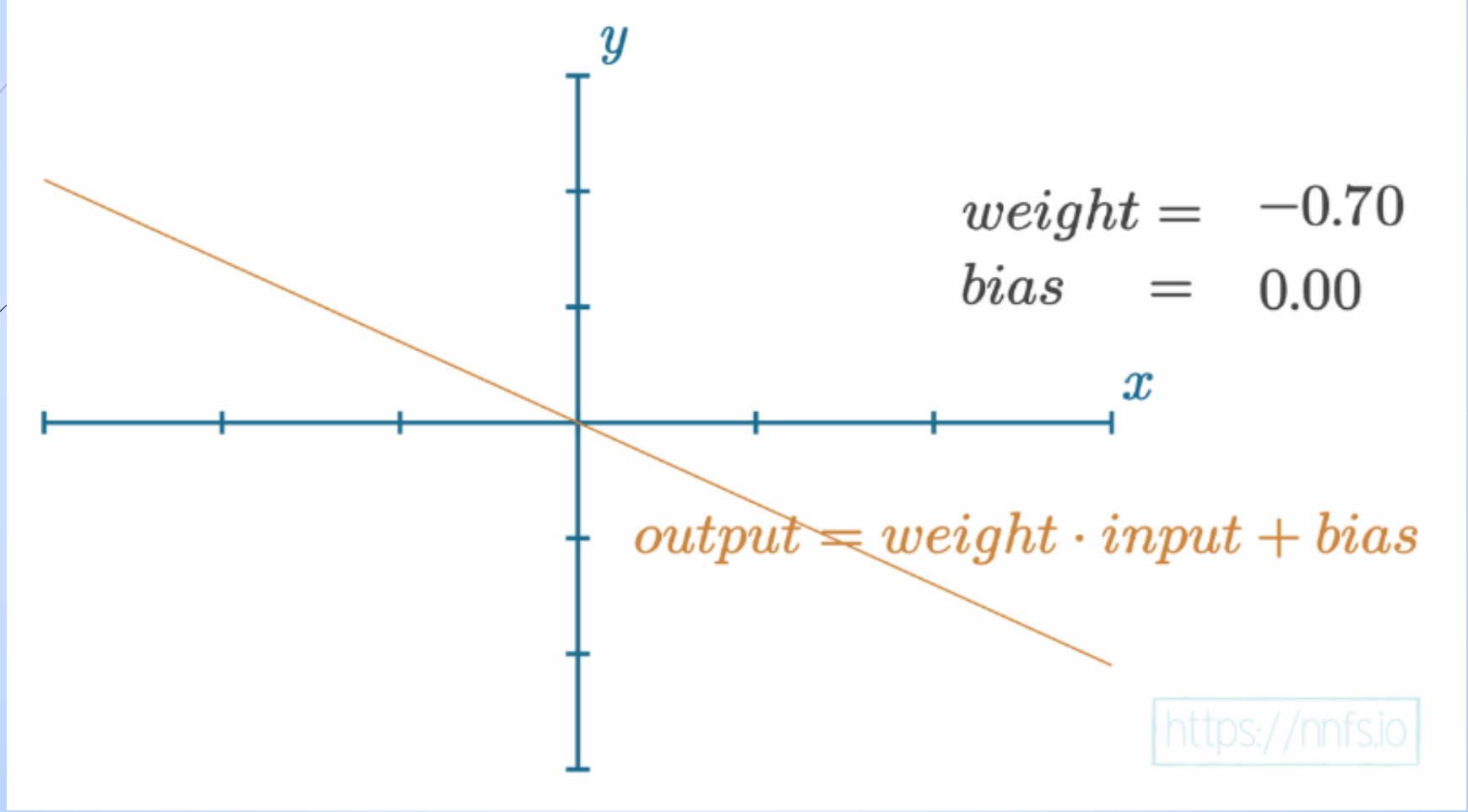
Impact of Weights



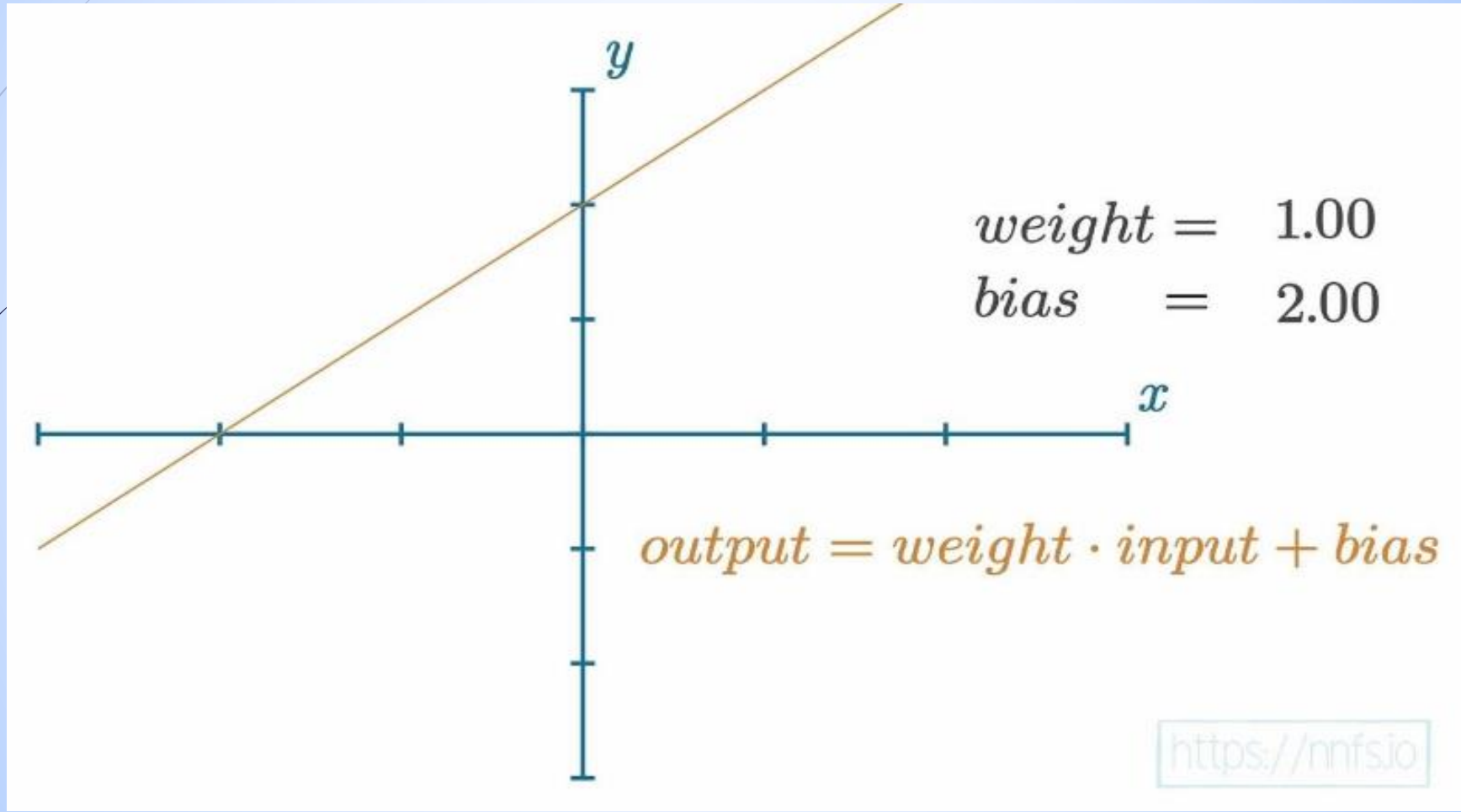
Impact of Weights (cont.)



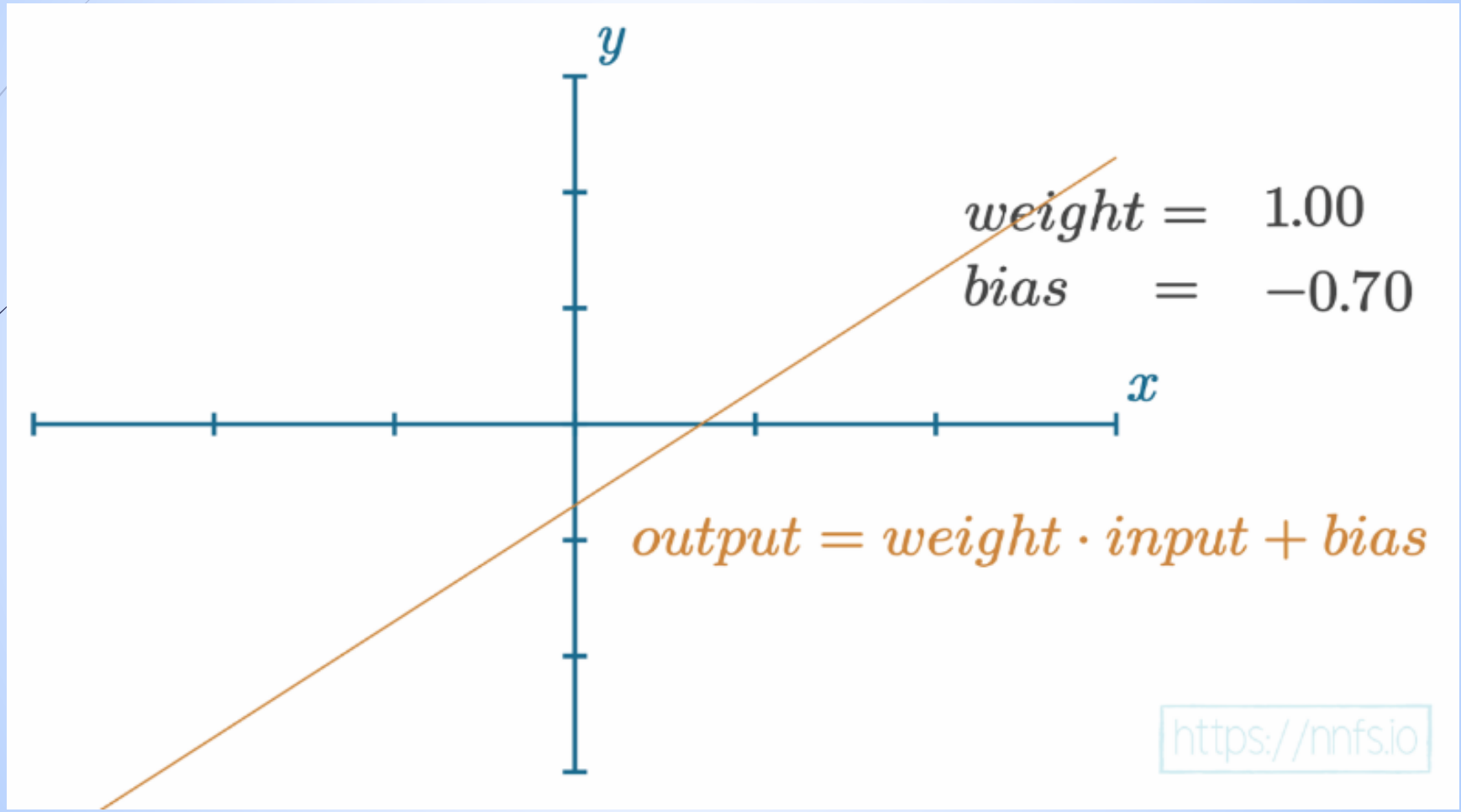
Impact of Weights (cont.)



Impact of Bias



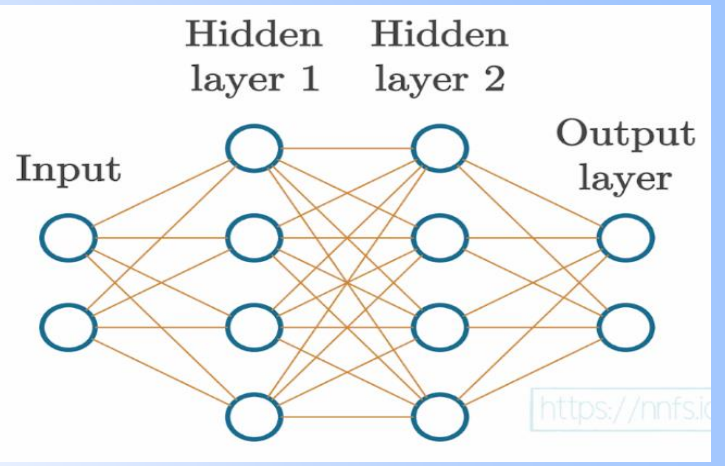
Impact of Bias (cont.)



Activation Function

- Activation functions mimic the neurons in brain, either "firing" or not.
- Simply put, getting excited on something or not.
- Neuron is activated/triggered if it meets criteria specified by activation function.
- Simple activation function can be stated like
  - Y = 1 if x > 0 ; Y = 0 if x <= 0
- One common activation function that neural networks use is Rectified Linear Unit (ReLU)

Another Simple Neural Network



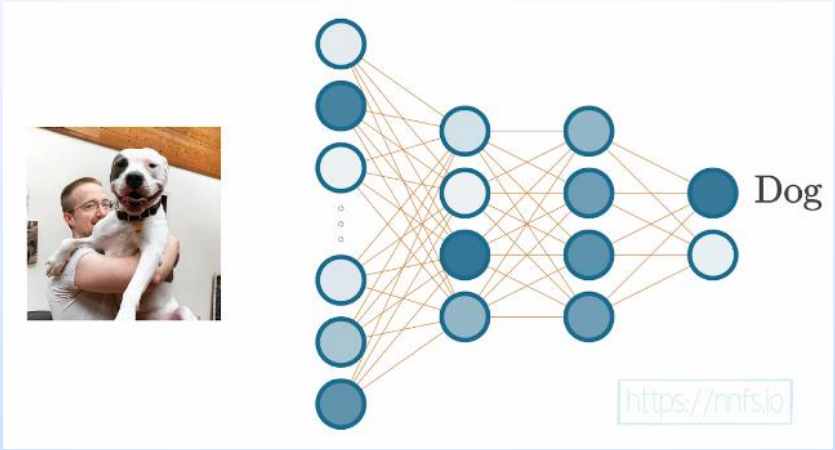
- Alongside hidden Layers, there are two layers named
  - Input & Output Layers
- Input layer represents actual data:
  - Pixel values from image etc.
- This "raw" pixel data can be passed into NN but typically preprocessing is done.
- Preprocessing is done through functions like normalization and scaling.
- It is common to preprocess data while retaining features and having values in similar ranges between 0 and 1 or 1 and -1.
- For achieving this task, either or both scaling and normalization are used.
- Output layer is whatever NN returns.

Classification Example

- In classification, the aim is to predict class of the given input.
- Output layer can have as many neurons as training dataset has classes.
- In case of binary classification, only a single neuron is enough.
  - Single neuron with output “Dog” or “Not Dog” etc
- Let's stick with example of Cats and Dogs classification with two neurons in output layer (as there are two classes).
- One neuron associated with “Dog” and the other with “Cat”.

Classification Example (cont.)

- For each passed image into the network, final output will have a calculated value in the "cat" and "dog" output neurons.
- Output neuron with highest score becomes class prediction for the image given as input.
- The end goal for neural networks is to adjust their weights and biases (parameters), so when applied to unseen example, they produce desired output(s).
- Weights and biases adjustment is what neural networks learn during training phase.



Overfitting

- In supervised approach, algorithms are trained with their associated desired outputs against every input.
- This is where **overfitting** issue arises!
- Overfitting is defined as: Model giving high accuracies on training but delivering poor results on unseen data.
- It happens when algorithm only learns to fit (get trained on) training data without actually "understanding" anything about input-output dependencies.
- It is because network basically "memorizes" training data :(

Generalization

- Instead of just "memorizing" all the training data …
  - Goal for model is to not only accurately predict on training data, but also on unseen data.
- Model fulfilling this goal is said to be doing Generalization.
  - Learning to fit the data instead of just memorizing it.
- But the next question is how do we achieve this goal? Any Guess?

Loss

- To train neural networks, we calculate how "wrong" they are in their predictions against actual labels / ground truths.
- This difference between predicted and ground truth on scale of how "much wrong" is basically an error called Loss.
- While training neural networks, the goal is to tune weights and biases in such a way that neural networks become less wrong (Lesser and lesser difference between prediction and target – the lesser the loss, the better the network)

Common Tasks of Neural Networks

- There are plenty of things being achieved with neural networks.

- Majority of things fall in given categories:
- Classification
- Regression
- Clustering
- …

Thank You