# Artificial Neural Network For Beginners

Admond Lee

# Little Bit About Me

## Admond Lee

Physics in NTU

Data Scientist

# Data Science Writer

M

Towards Data Science

TECHINASIA

AI TIME JOURNAL

KDnuggets

think/>
by UpCode Academy

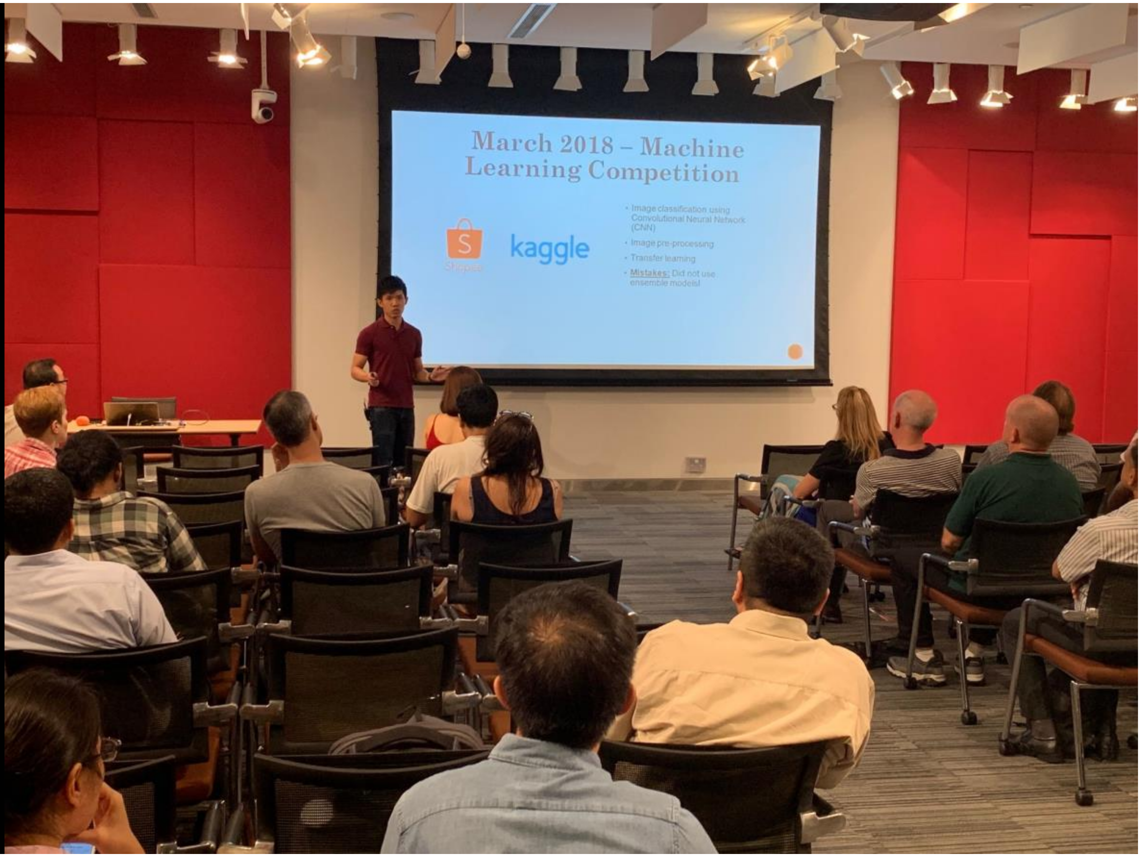
Experfy

# Data Science Consulting

- Helping digital marketing agencies and companies solve problems using data
- Data Science Consulting
  - ➢ Web Analytics
  - ➢ Attribution Modelling
  - ➢ Deep Learning / ML
  - ➢ Advanced Social Analytics

# Data Science Communicator

How to go into Data Science?
• Resume alone is not enough
• Online courses and certification are not enough
• School projects are not enough
• In other words, you HAVE TO stand o
• You need to have your:
✓ Portfolio (Personal projects, comp ions, internship if possible etc.)
✓ Personal branding (social media)
✓ And many more...


March 2018 – Machine Learning Competition
Shopee
kaggle
• Image classification using Convolutional Neural Network (CNN)
• Image pre-processing
• Transfer learning
• Mistakes: Did not use ensemble models!

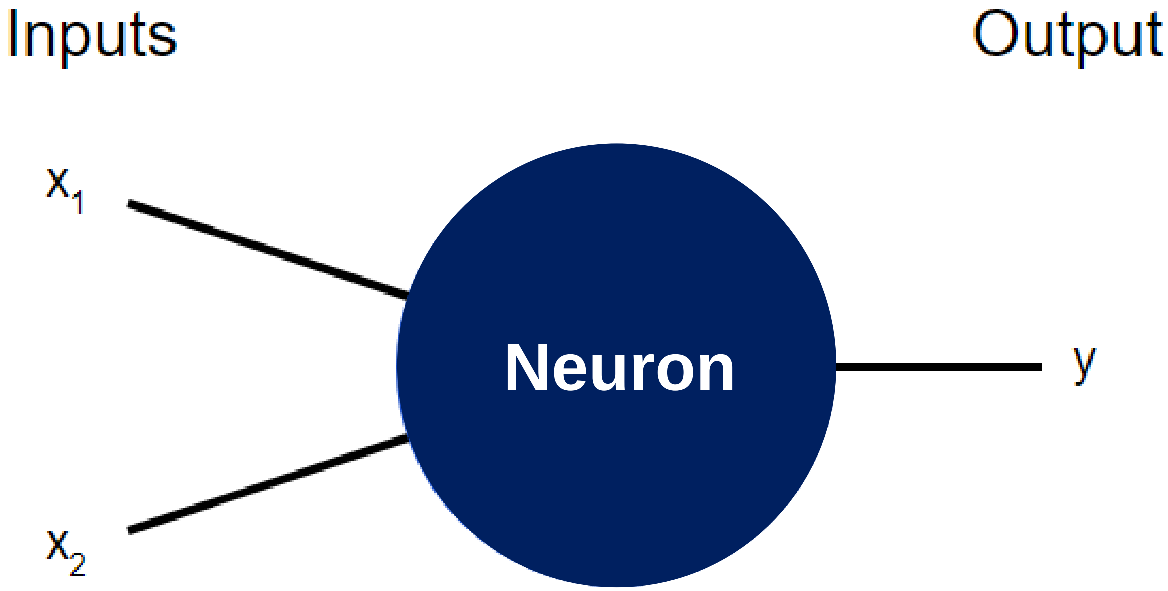
# What We'll Cover Today

What is Artificial Neural Network?

Build a simple ANN model
(for real)

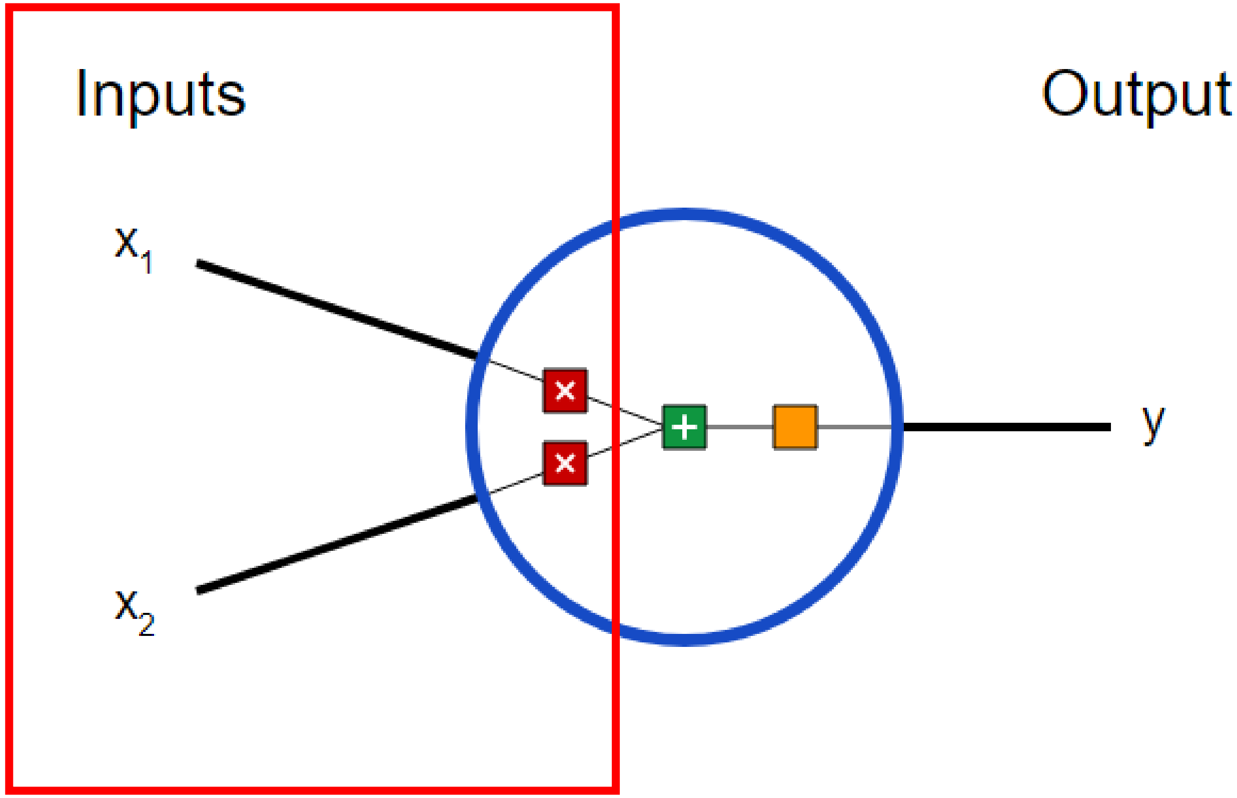
# What is Artificial Neural Network?

# Fundamental Building Blocks – Neurons



A neuron takes inputs, does some math with them, and produces one output

# Fundamental Building Blocks – Neurons



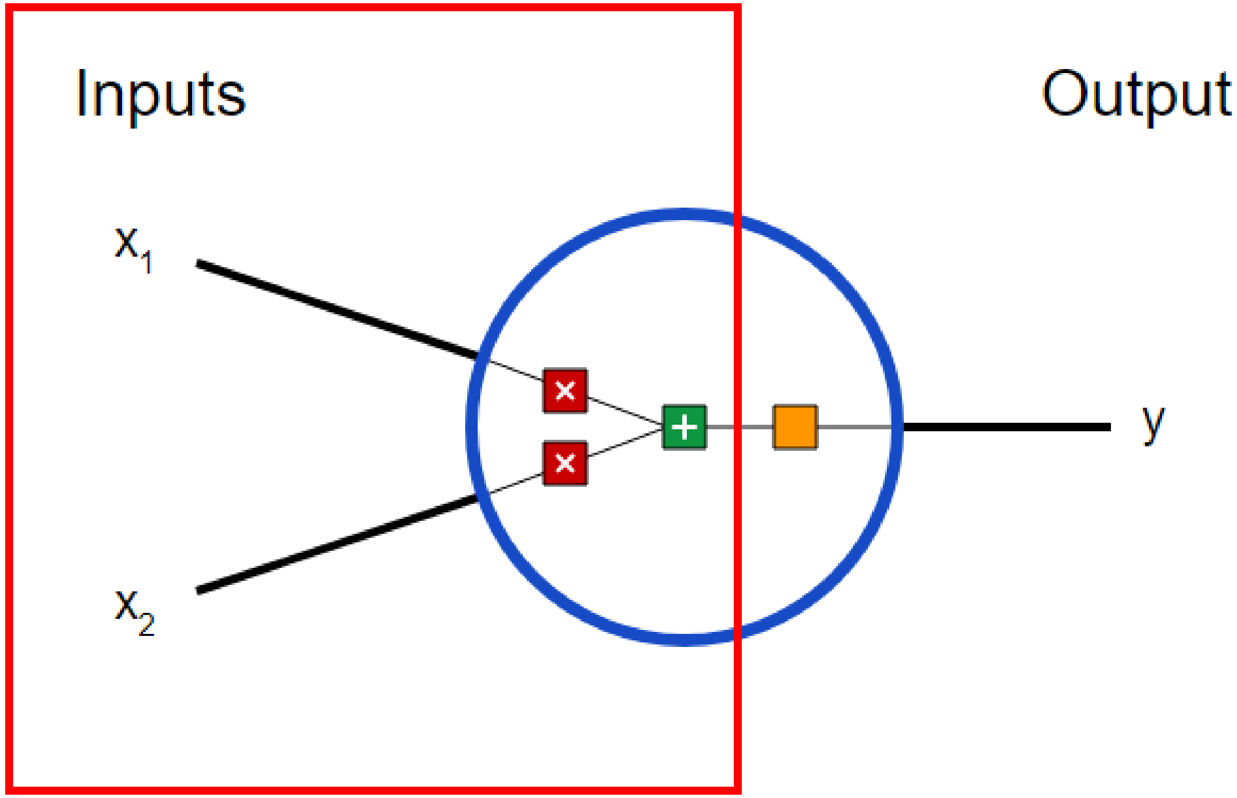
$x_1 \rightarrow x_1 * \boxed{w_1}$ weight

$x_2 \rightarrow x_2 * w_2$

The higher the weight:

- The faster it will trigger the activation function
- The more impact it will have on the network

# Fundamental Building Blocks – Neurons

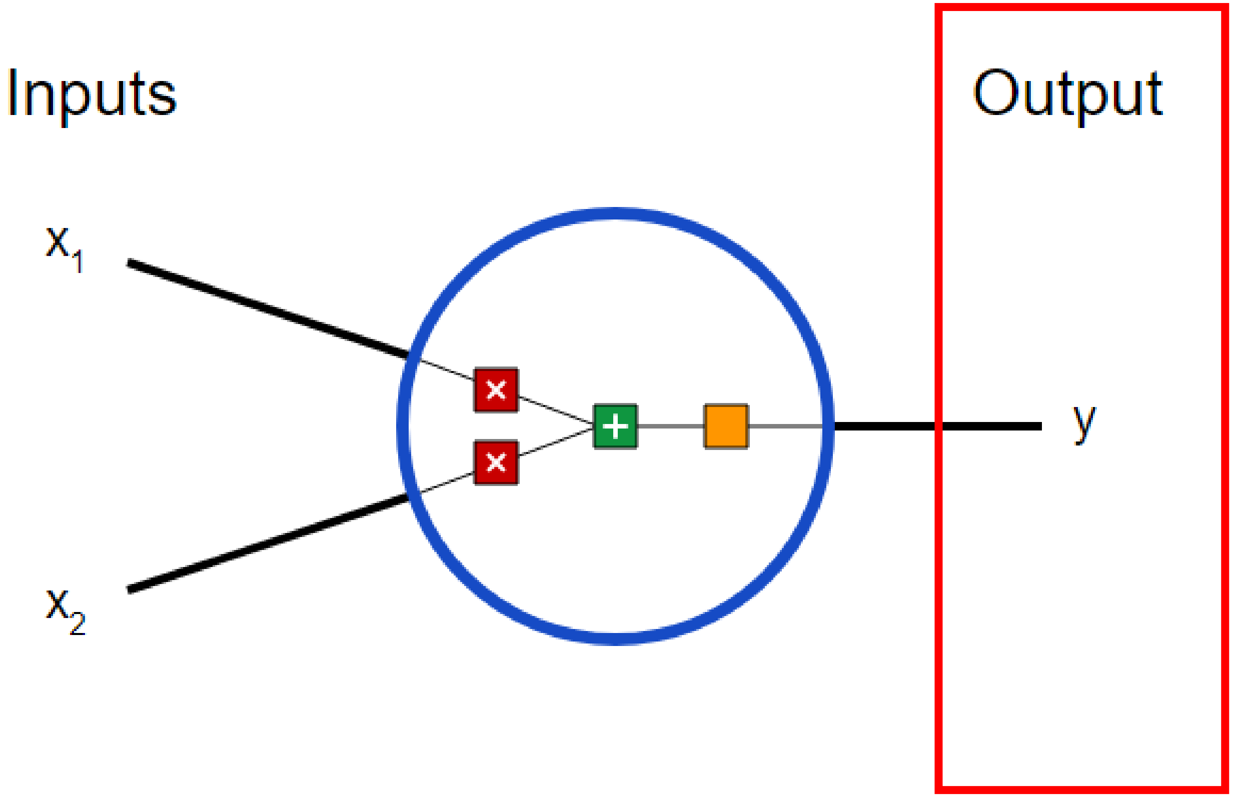


$$(x_1 * w_1) + (x_2 * w_2) + \boxed{b}$$

bias

Bias is used to delay the triggering of the activation function

# Fundamental Building Blocks – Neurons



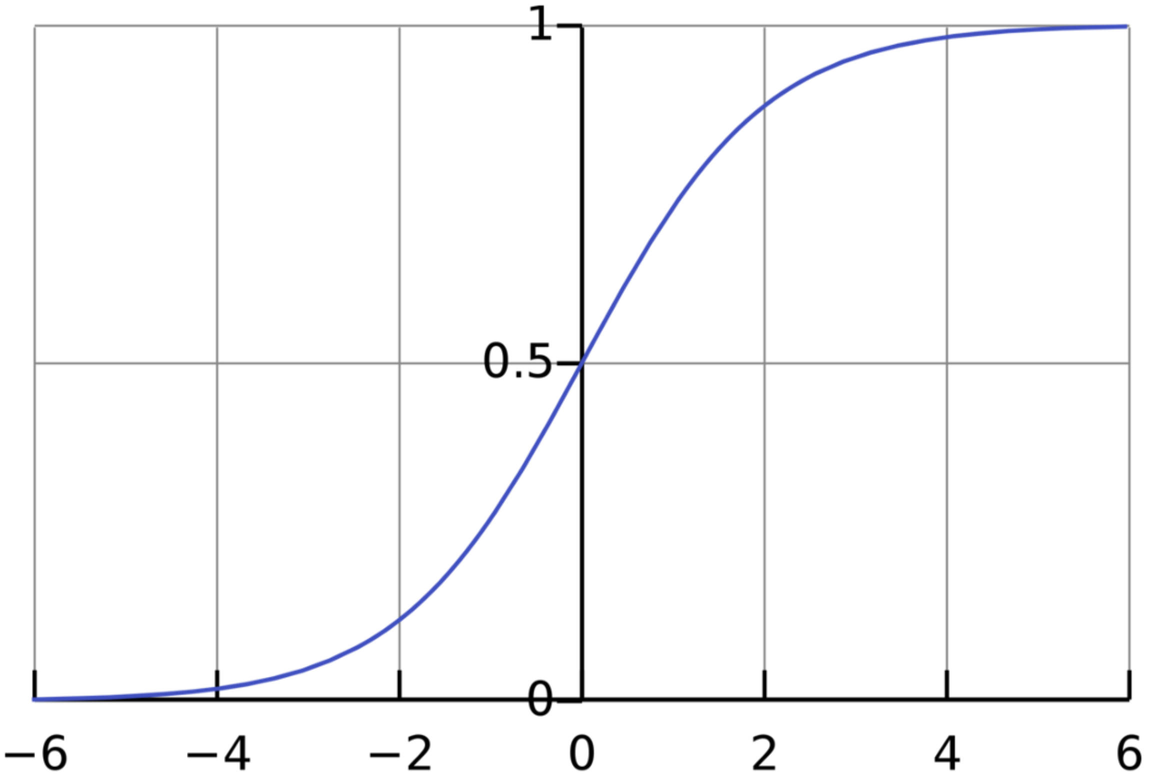
Activation function

$$y = f(x_1 * w_1 + x_2 * w_2 + b)$$

# Activation Function

- An activation function is used to turn an unbounded input into an output that has a nice, predictable form

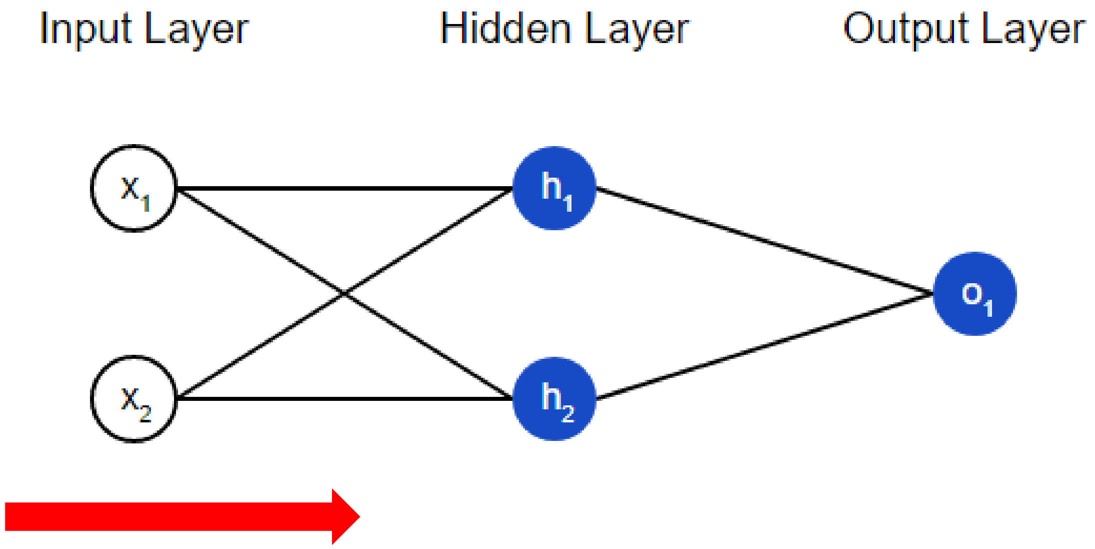
- It will decide whether a neuron should fire or not



Sigmoid function

# Neural Network

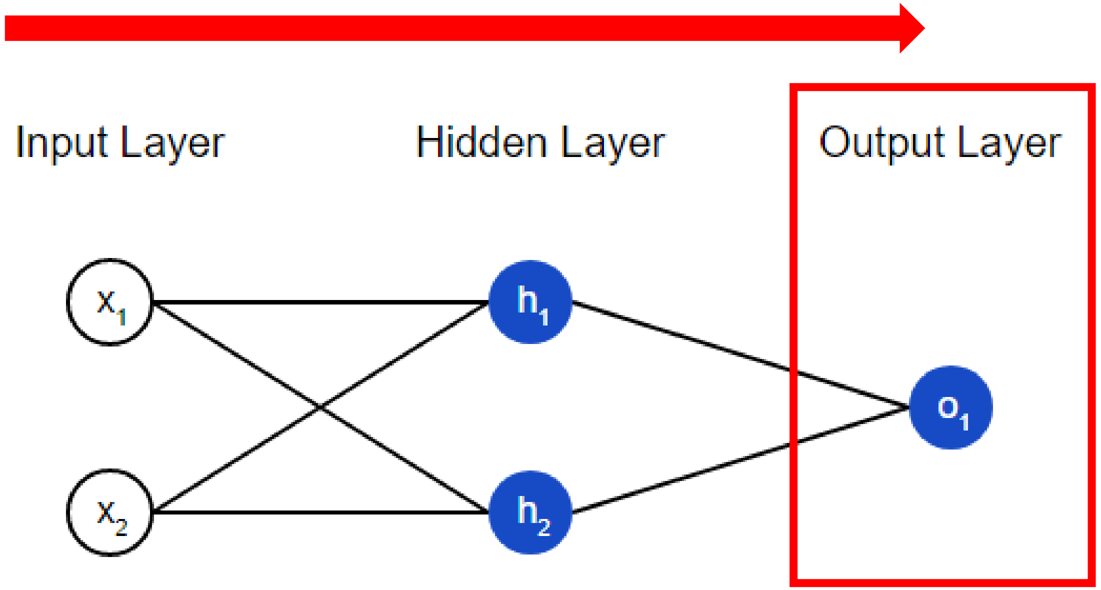
- A neural network is nothing more than a bunch of neurons connected together
- A neural network can have any number of layers with any number of neurons in those layers



Feedforward Neural Network

# Training a Neural Network

- Minimize the error (or loss) from predictions
- Lower loss → Better predictions
- Mean Squared Error (MSE) → $\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_{true} - y_{pred})^2$ **(Loss function)**

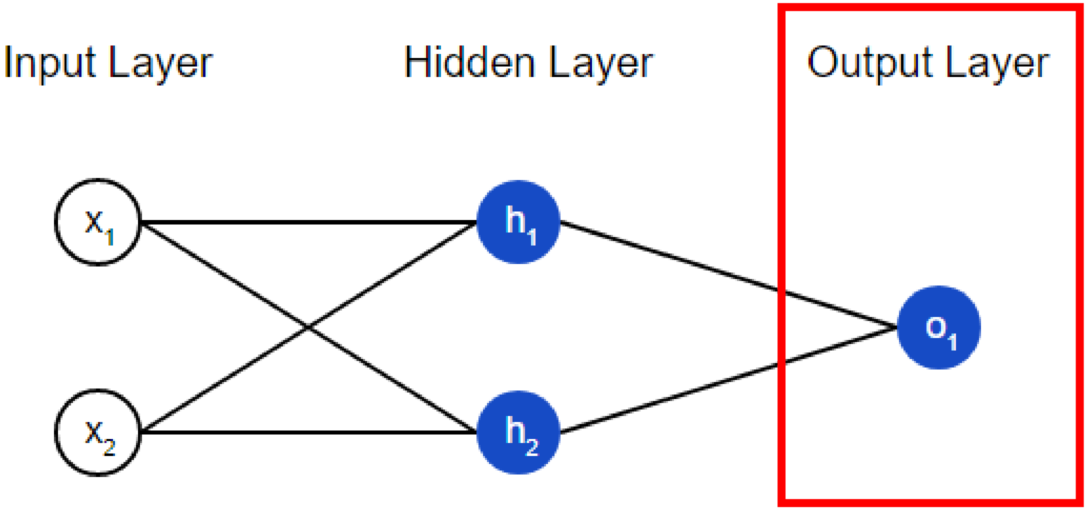


- Compare the actual and predicted values
- Compute loss function to find the error

$$y = f(x_1 * w_1 + x_2 * w_2 + b)$$

# Backpropagation

- AIM – Update the network’s weights and biases to minimize the error from predictions
- Lower loss → Better predictions
- Mean Squared Error (MSE) → $\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_{true} - y_{pred})^2$ **(Loss function)**



- How to know which weights to be updated
  - ➢ Partial derivative

- This system of calculating partial derivatives by working backwards is known as backpropagation

$$w_1 \leftarrow w_1 - \eta \frac{\partial L}{\partial w_1}$$

$$\boxed{\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial y_{pred}} * \frac{\partial y_{pred}}{\partial h_1} * \frac{\partial h_1}{\partial w_1}}$$

$$y = f(x_1 * w_1 + x_2 * w_2 + b)$$

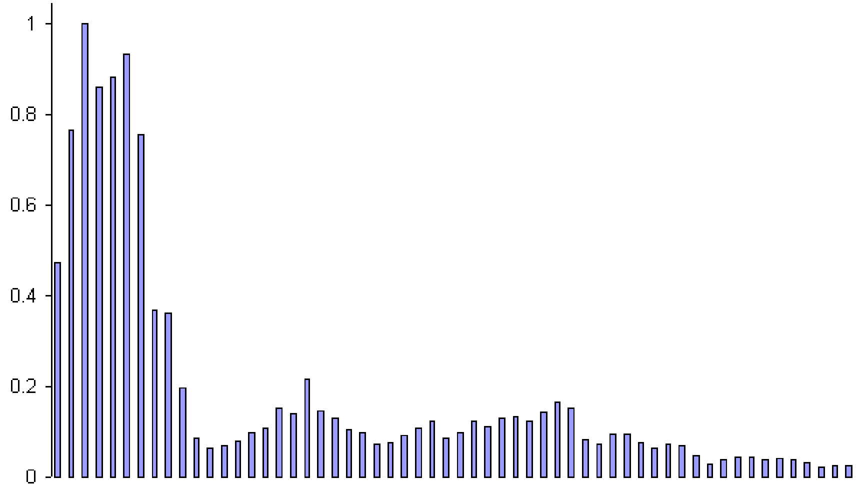
# Example – Voice Recognition

**Task**

- Learn to discriminate between two different voices saying “Hello”

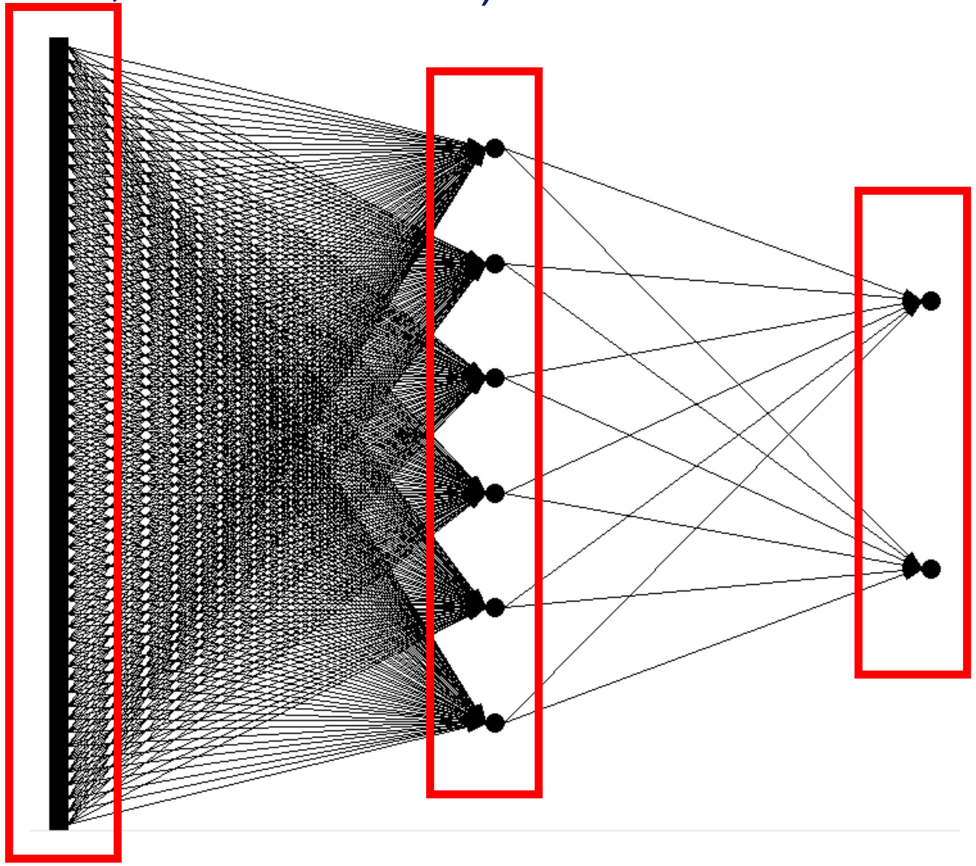
**Data**

- Sources
  - ➢ Steve
  - ➢ David
- Format
  - ➢ Frequency distribution (60 bins)



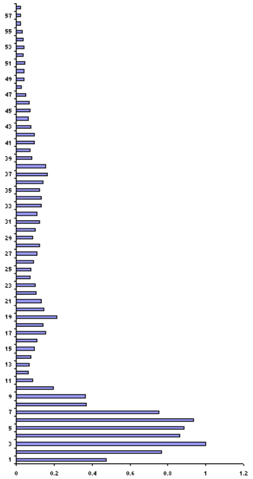
# Network architecture

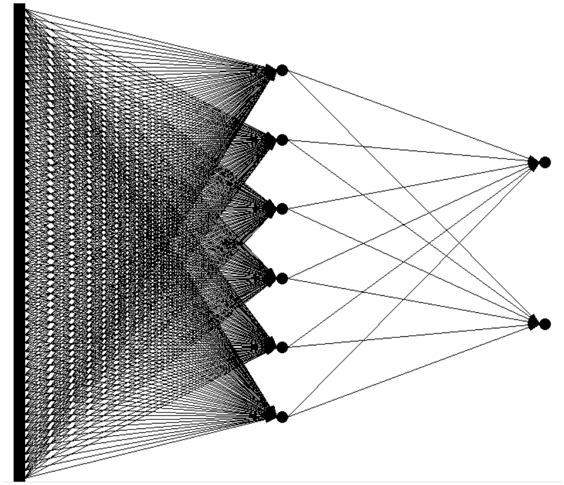
- Feedforward network
  - 60 input (one for each frequency bin)
  - 6 hidden
  - 2 output (0-1 for “Steve”, 1-0 for “David”)



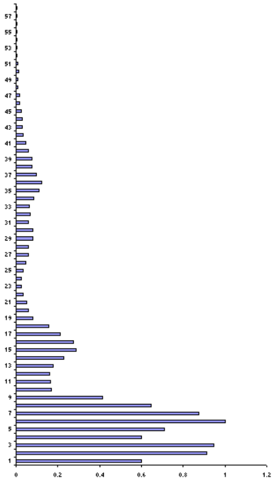
# Presenting the data

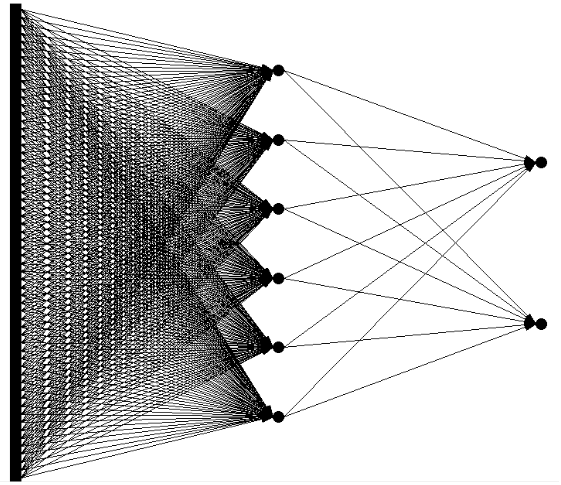
Steve





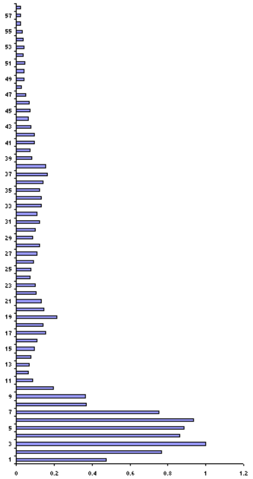
David

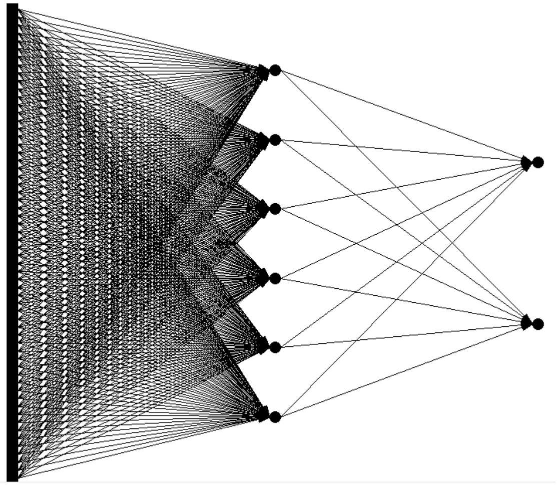




# Presenting the data (untrained network)

Steve

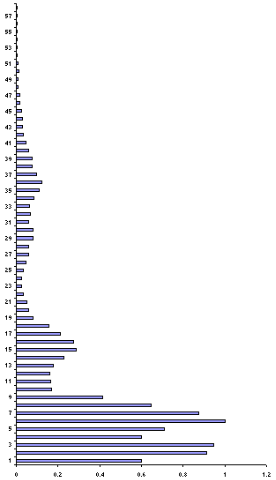


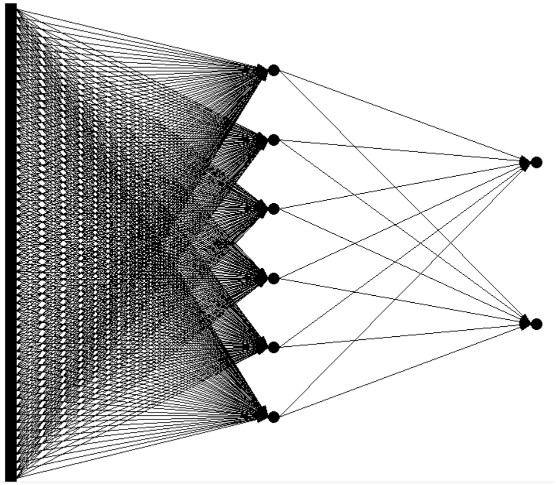


0.43

0.26

David



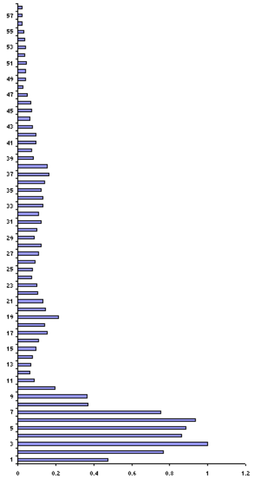


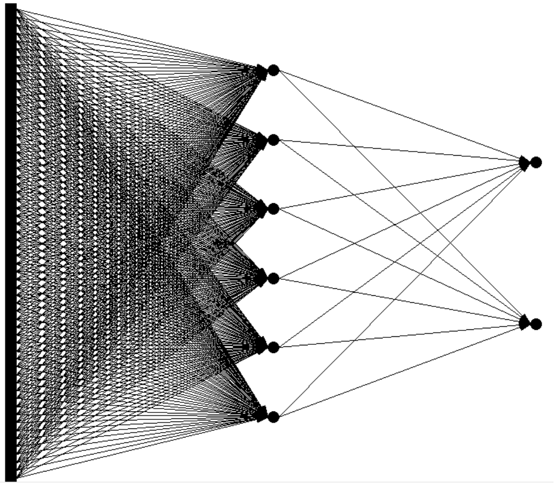
0.73

0.55

# Calculate error

Steve

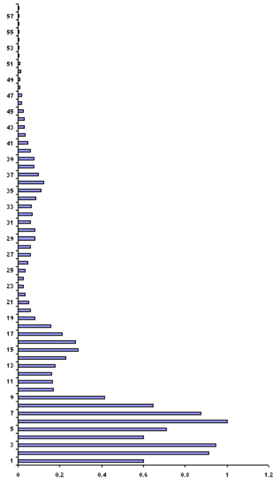


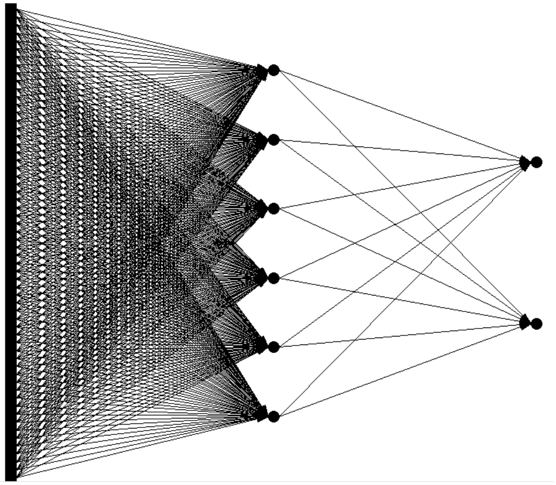


0.43 – 0 = 0.43

0.26 – 1 = 0.74

David



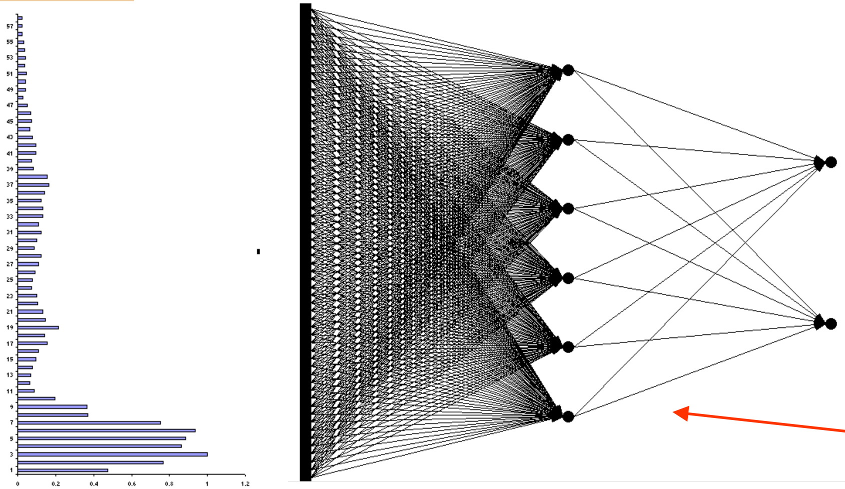


0.73 – 1 = 0.27

0.55 – 0 = 0.55

# Backpropagate error and adjust weights

Steve

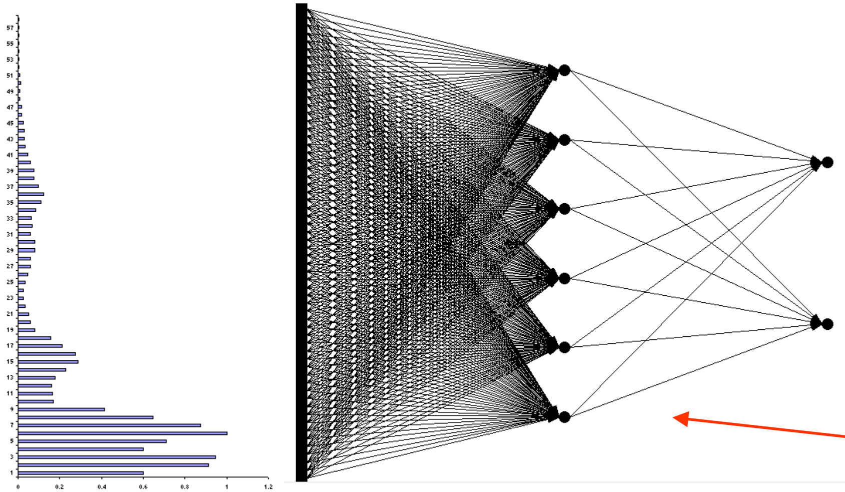


$|0.43 - 0|$ $= 0.43$

$|0.26 - 1|$ $= 0.74$

1.17

David



$|0.73 - 1|$ $= 0.27$

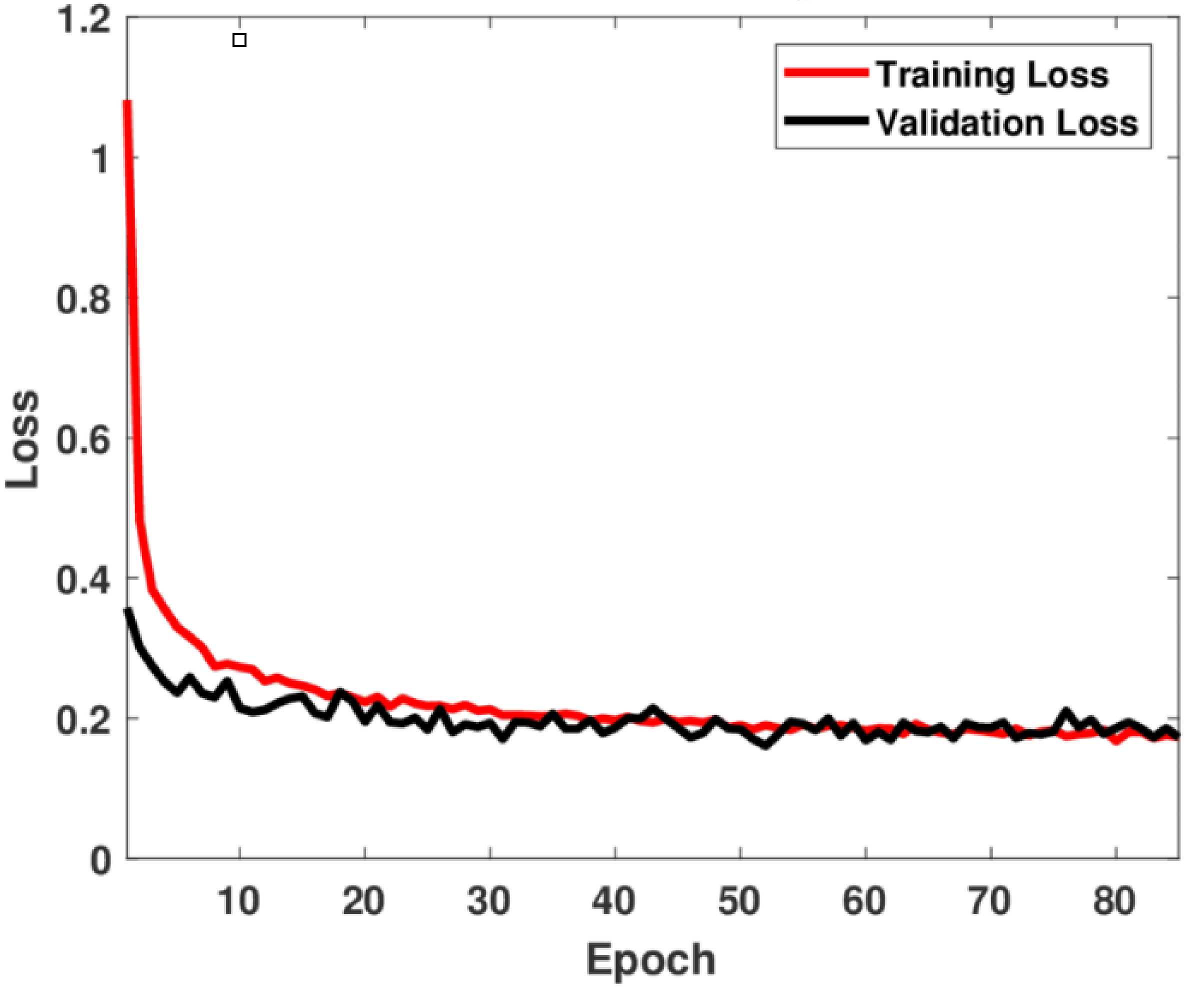
$|0.55 - 0|$ $= 0.55$

0.82

# Example – Voice Recognition

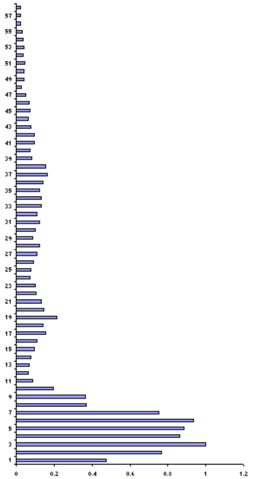
- Repeat process (sweep) for all training pairs
  - ➢ Present data
  - ➢ Calculate error
  - ➢ Backpropagate error
  - ➢ Adjust weights

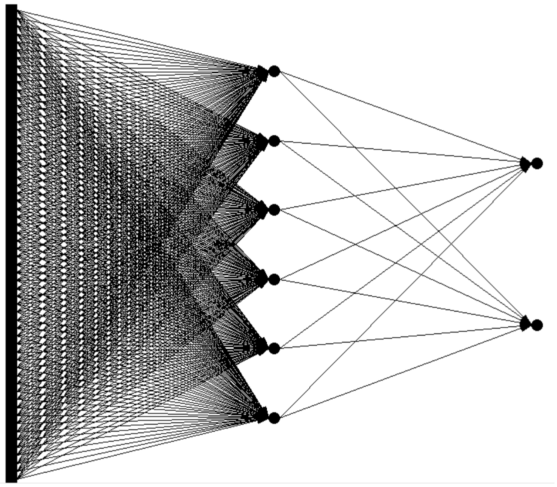
- Repeat process multiple times



# Presenting the data (trained network)

Steve

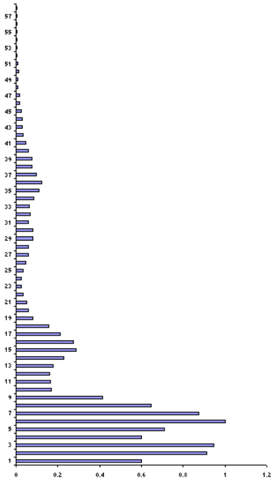


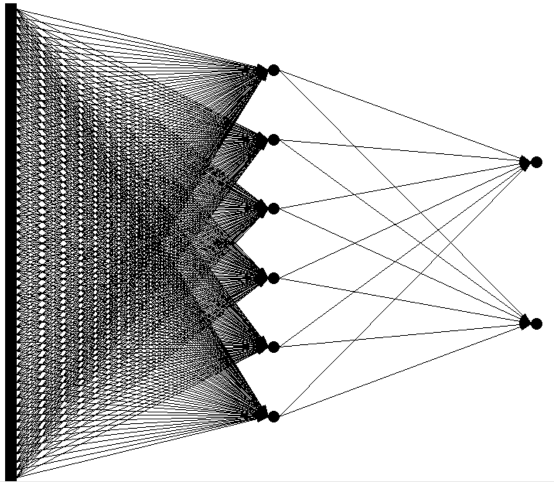


0.01

0.99

David





0.99

0.01

# Neural Network Demo

http://playground.tensorflow.org

Keras

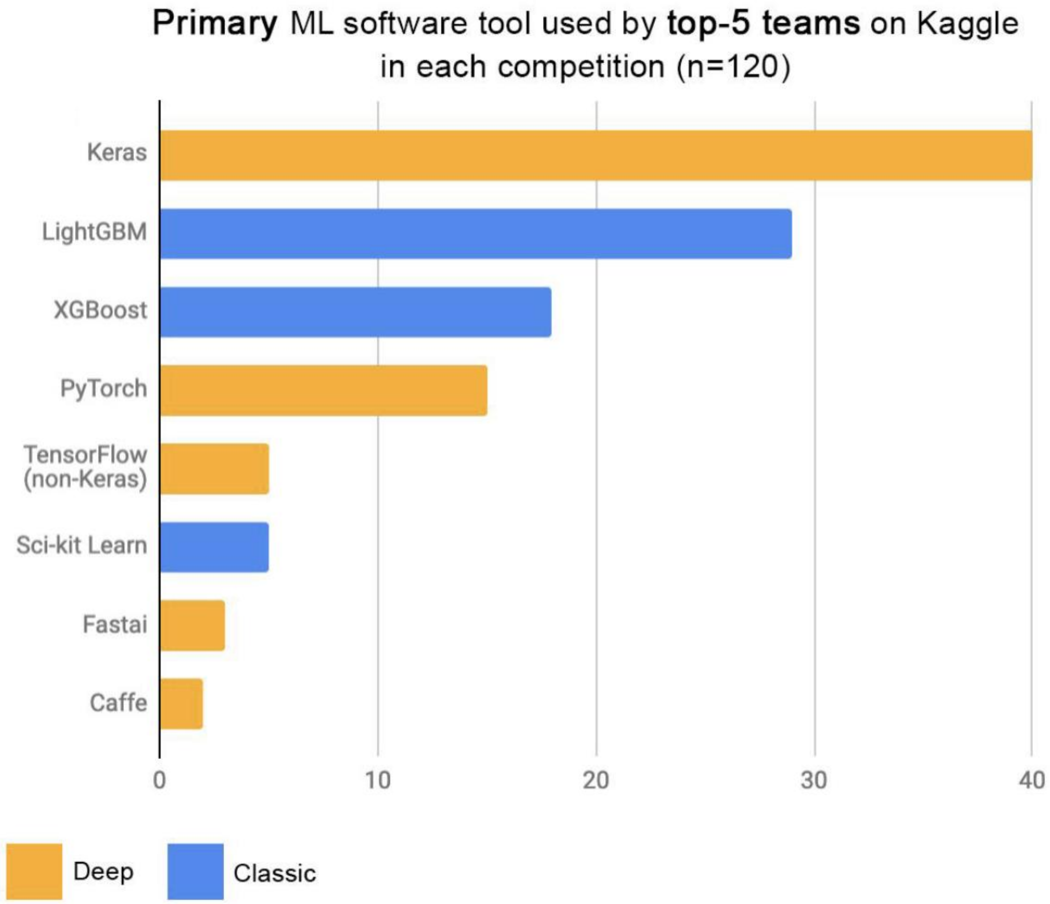
# Introduction to Keras

# What is Keras?

- **Keras** is a high-level deep learning API written in Python that can run on top of the deep learning libraries (TensorFlow, CNTK, Theano etc.)
  - ➢ Python deep learning library
  - ➢ Basically an API  “designed for human beings, not machines”
- **TensorFlow** is an open-source deep learning library developed by Google
- When TensorFlow 2.0 was released in late 2019, it adopted Keras as the official API for TensorFlow 2.0
  - ➢ Can import Keras directly from TensorFlow library

# Why Keras?

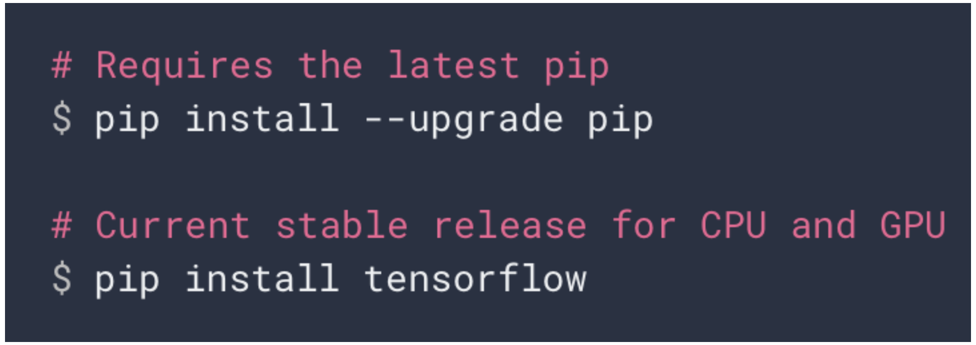
- Easy to use as it provides simple APIs to build deep learning models
- Provide clear & actionable error messages
- Easy to test new ideas, build prototype models, and deploy to production
- Strong multi-GPU & distributed training support
- Broad adoption in the industry and the research community



*Year: 2019*

# How to get Keras?

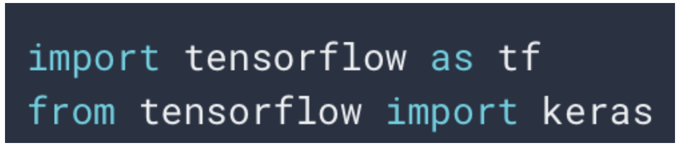
1. Install TensorFlow



```
# Requires the latest pip
$ pip install --upgrade pip

# Current stable release for CPU and GPU
$ pip install tensorflow
```

2. Import Keras



```
import tensorflow as tf
from tensorflow import keras
```

# Let's use Keras to build a simple ANN model and see how it works

# Let’s Connect!

in ***bit.ly/admond-linkedin***

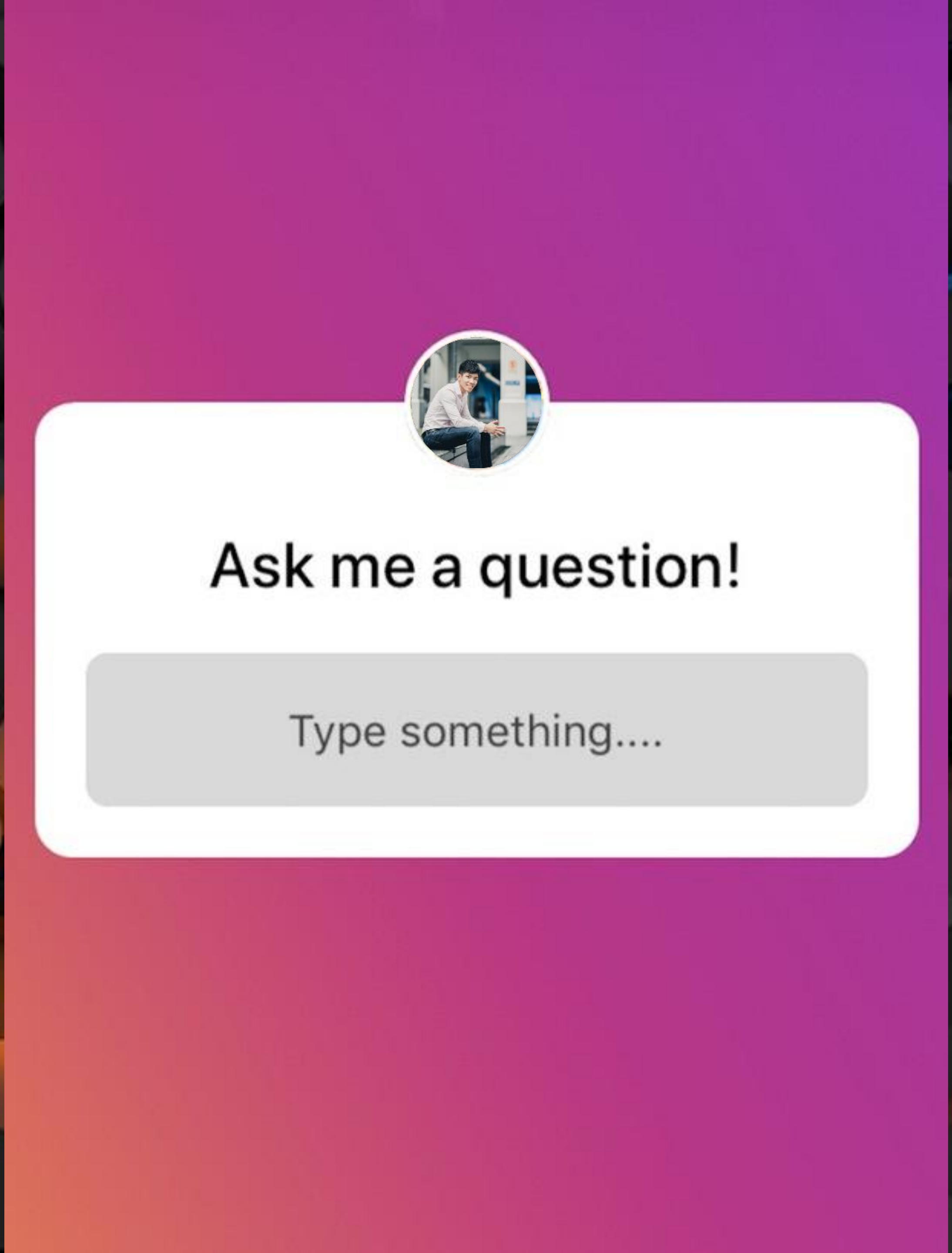
M ***bit.ly/admond-medium***

***bit.ly/admond-twitter***

***bit.ly/admond-instagram***

AL ***bit.ly/admond-website***

# THANK YOU!



Ask me a question!