

# Introduction to ANN



# Understanding AI

## ARTIFICIAL INTELLIGENCE

Enabling computers to mimic human behaviour

## MACHINE LEARNING

Subset of AI that uses statistical tools for the computers to learn from data

BASED ON HUMANS BRAIN

NEURAL NETWORKS

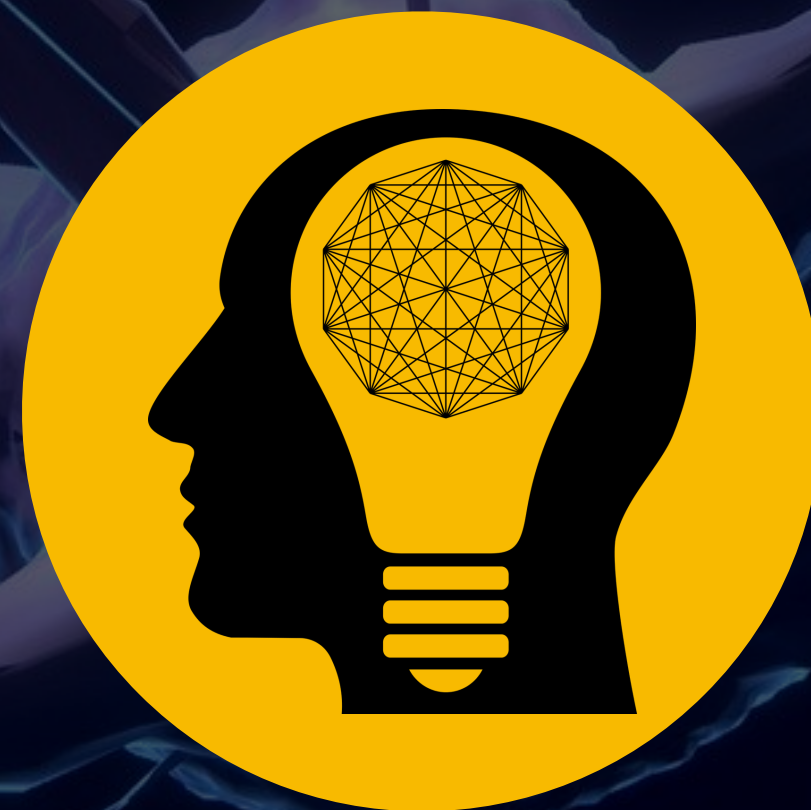
DEEP LEARNING





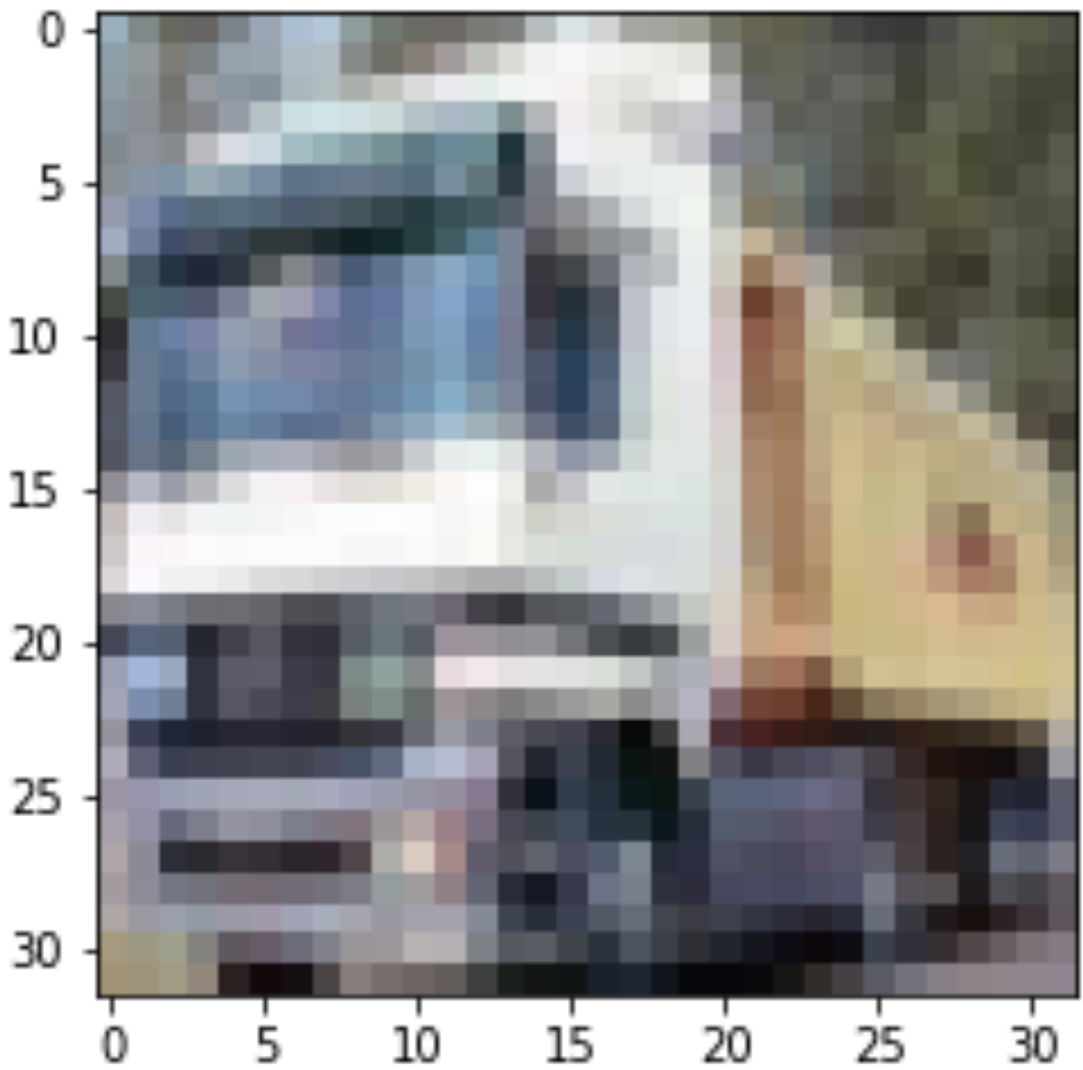
# About ANN

- ANN stands for Artificial Neural Network and in simple words it's a computational model that attempts to recognize patterns in a particular dataset through a process that mimics the way the human brain works in order to solve complex problems
- If a ANN consists of 2 or more hidden layers it becomes a deep neural network
- Nowadays we use them to solve many problems i.e. image recognition/ classification, making predictions (stock market, sales volume), creating clusters in a certain dataset and more





# ANN example



Shape: (32, 32, 3)

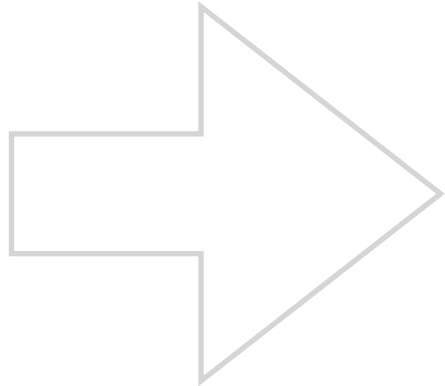
Height: 32

Width: 32

Channels: 3

Total Inputs = Height \* Width \* Channels

Total Inputs =  $32 * 32 * 3 = 784$



```
array([[154, 177, 187],
       [126, 137, 136],
       [105, 104, 95],
       ...,
       [ 91, 95, 71],
       [ 87, 90, 71],
       [ 79, 81, 70]],

      [[140, 160, 169],
       [145, 153, 154],
       [125, 125, 118],
       ...,
       [ 96, 99, 78],
       [ 77, 80, 62],
       [ 71, 73, 61]],

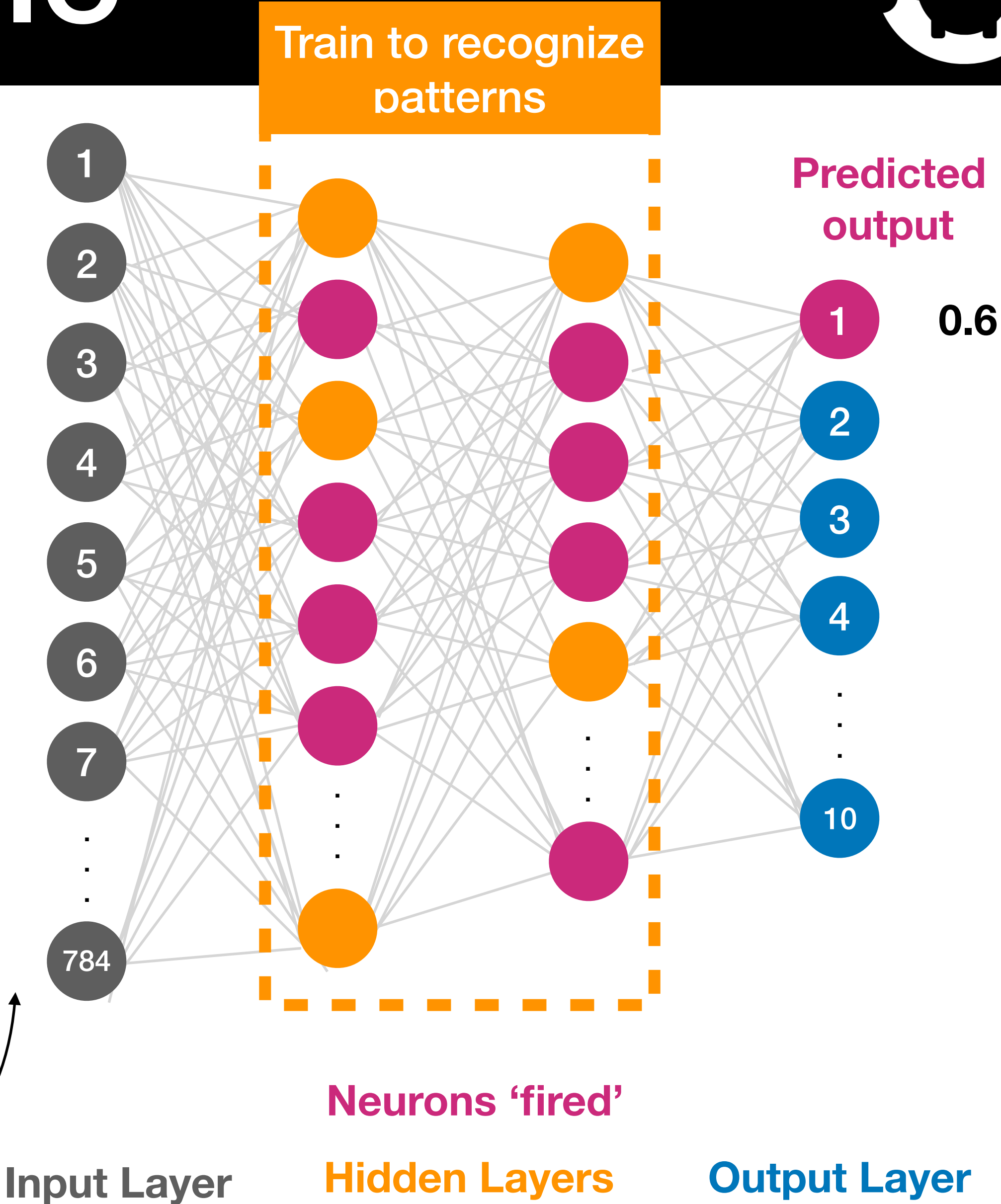
      [[140, 155, 164],
       [139, 146, 149],
       [115, 115, 112],
       ...,
       [ 79, 82, 64],
       [ 68, 70, 55],
       [ 67, 69, 55]],

      ...,

      [[175, 167, 166],
       [156, 154, 160],
       [154, 160, 170],
       ...,
       [ 42, 34, 36],
       [ 61, 53, 57],
       [ 93, 83, 91]],

      [[165, 154, 128],
       [156, 152, 130],
       [159, 161, 142],
       ...,
       [103, 93, 96],
       [123, 114, 120],
       [131, 121, 131]],

      [[163, 148, 120],
       [158, 148, 122],
       [163, 156, 133],
       ...,
       [143, 133, 139],
       [143, 134, 142],
       [143, 133, 144]]], dtype=uint8)
```



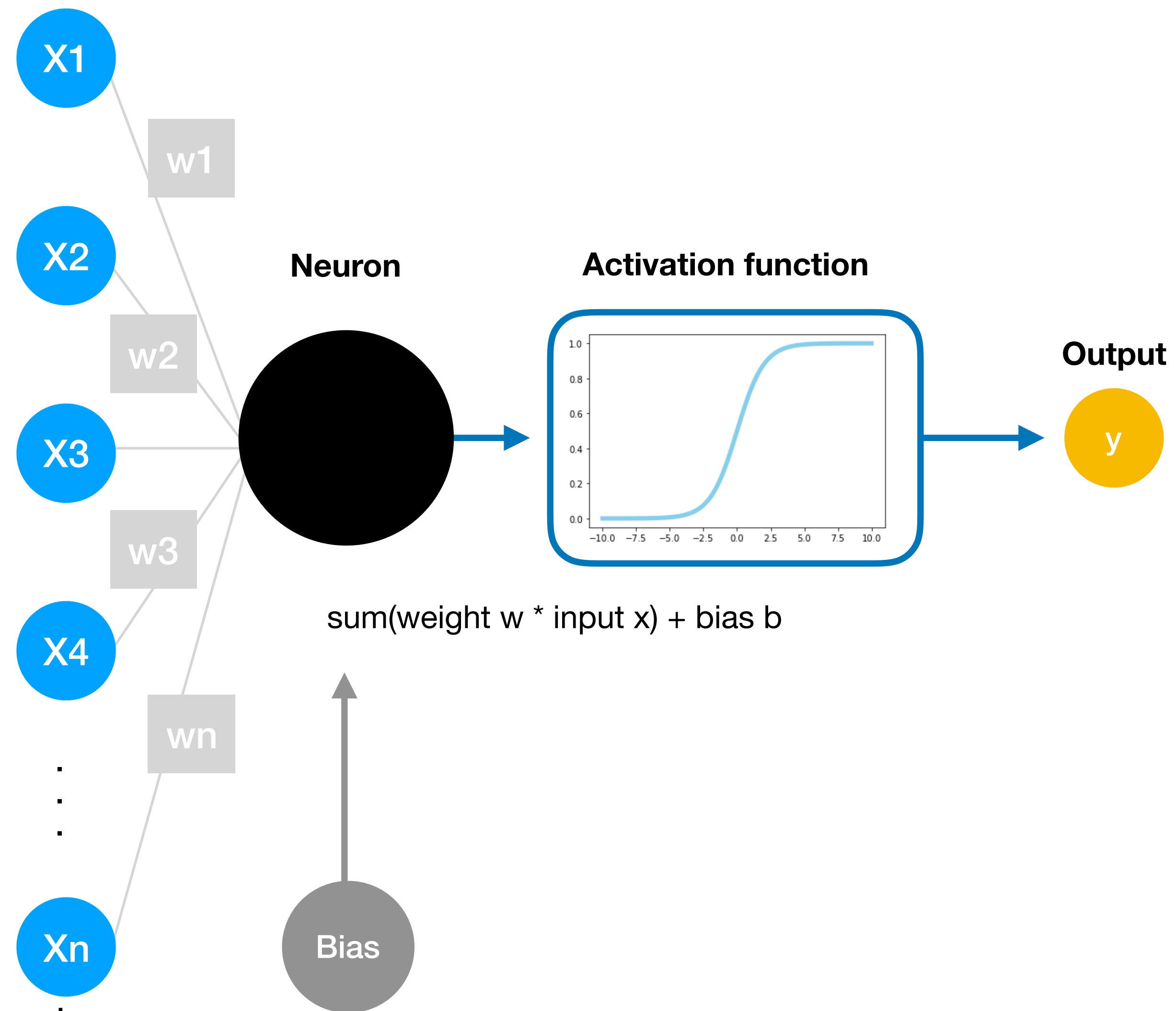
# How does it work?

Do NOT confuse  
with color channels

- Each pixel is an input to the neuron at the input layer
- Neurons of a particular layer are connected to neurons of the next one through channels with assigned weights, which determine a strength of connection between those neurons
- The inputs are multiplied by the weights and their sum is send as an input to the next layer
- Activation function determines whether a neuron should be activated or not by taking the sum of weights from the input and adding bias to it (a constant that helps the model perform better).
- If the sum is greater that a certain threshold it will be fired and the data will be transmitted to the next neuron in the next layer (forward propagation)
- At the end (output layer) we compare predicted output to the actual one and calculate error which enables to adjust model parameters (weights and biases) in so called backpropagation process
- We train our model on a dataset performing forward and backpropagation cycles until our model is capable of predicting correctly certain values and the weights are assigned



# Closer look



# CNN

- CNN stands for Convolutional Neural Network is a type of Deep Neural Network mainly used with the purpose of working with images (i.e. in order to perform face recognition or object classification)
- CNN are models with more advanced architecture having related to it topics such as pooling, filters etc.
- How a simple CNN constructed in Keras looks like and how to integrate it with Django will be presented at the end of this course



# Imagenet

- Imagenet is a huge online images of over 14 million high quality images at this moment, that are labeled and because of this fact they can be used to train Machine Learning Models
- It was designed by academics with the purpose for computer vision research
- Website: <http://www.image-net.org/>

IMAGENET

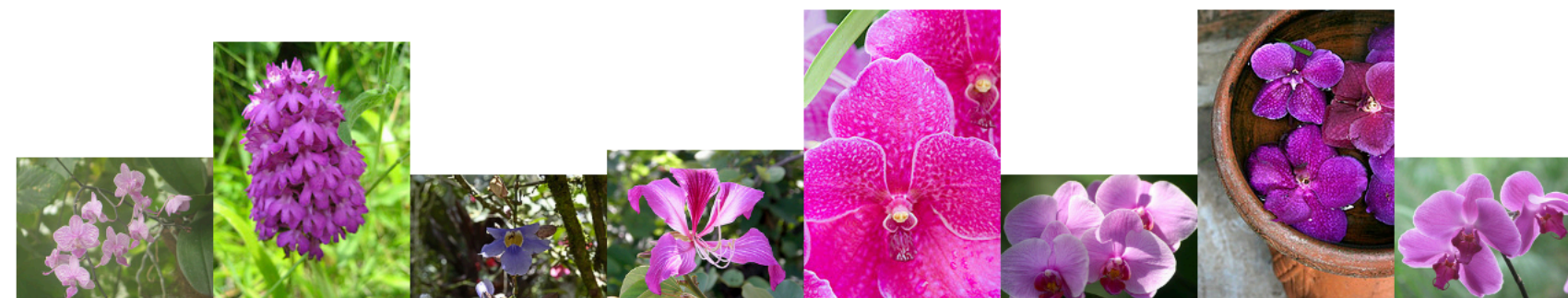
14,197,122 images, 21841 synsets indexed

[Explore](#) [Download](#) [Challenges](#) [Publications](#) [Updates](#) [About](#)

Not logged in. [Login](#) | [Signup](#)

**ImageNet** is an image database organized according to the **WordNet** hierarchy (currently only the nouns), in which each node of the hierarchy is depicted by hundreds and thousands of images. Currently we have an average of over five hundred images per node. We hope ImageNet will become a useful resource for researchers, educators, students and all of you who share our passion for pictures.

[Click here](#) to learn more about ImageNet, [Click here](#) to join the ImageNet mailing list.





# Keras pretrained models

- According to the Keras documentation:

*“Keras Applications are deep learning models that are made available alongside pre-trained weights. These models can be used for prediction, feature extraction, and fine-tuning. Weights are downloaded automatically when instantiating a model”*

- List of pretrained models: <https://keras.io/applications/>
- On the next slide we will create a solution for classifying images using InceptionResNetV2 and getting the weights from imagenet



# Example

```
# Imports
import numpy as np
import keras
import tensorflow as tf
from keras.preprocessing.image import img_to_array, load_img
from keras.applications.inception_resnet_v2 import InceptionResNetV2, decode_predictions, preprocess_input

# Instantiate the model
model = InceptionResNetV2(weights='imagenet')

# Load and resize image
img_path = "/full_path/some_img.jpg"
img = load_img(img_path, target_size=(299, 299))

# convert to array and expand the shape to 4 dimensions
img_arr = img_to_array(img)
img_arr = np.expand_dims(img_arr, axis=0)

# preprocess and make predictions
prep = preprocess_input(img_arr)
predictions = model.predict(prepare)
decode_predictions(predictions)

...
Result:
[[('n04517823', 'vacuum', 0.93994546),
 ('n03794056', 'mousetrap', 0.0035129644),
 ('n02123045', 'tabby', 0.0010646613),
 ('n04049303', 'rain_barrel', 0.00087809074),
 ('n03424325', 'gasmask', 0.00082691014)]]
...
```

Setting weights to imagenet, these weights will be downloaded the first time we will run our code and the model will be instantiated

Reshape off input image according to the keras documentation dedicated to our InceptionResNetV2 model



# Thank you

