multiply

w0

Constant

wn-1

**…**

w1

output

**…**

wn

inputs

weights

activation function

Weighted sum

output

**Artificial Intelligence**

Visual perception

Natural language processing

Intelligent robot

Automatic programming

Automatic reasoning

Knowledge representation

**…**

**…**

**Machine learning**

Principal components analysis

Support vector machine

Linear/logistic regression

K-nearest neighbors

K-Means

Decision trees

Random forest

**Neural Networks**

**…**

**…**

Boltzmann neural networks

MPL

**Deep Learning**

**…**

**…**

CNN

RNN

GAN

DBN

**…**

**Input Layer**

**Output Layer**

**Hidden Layers**

**mathematics behind AI on Digit Recognition**

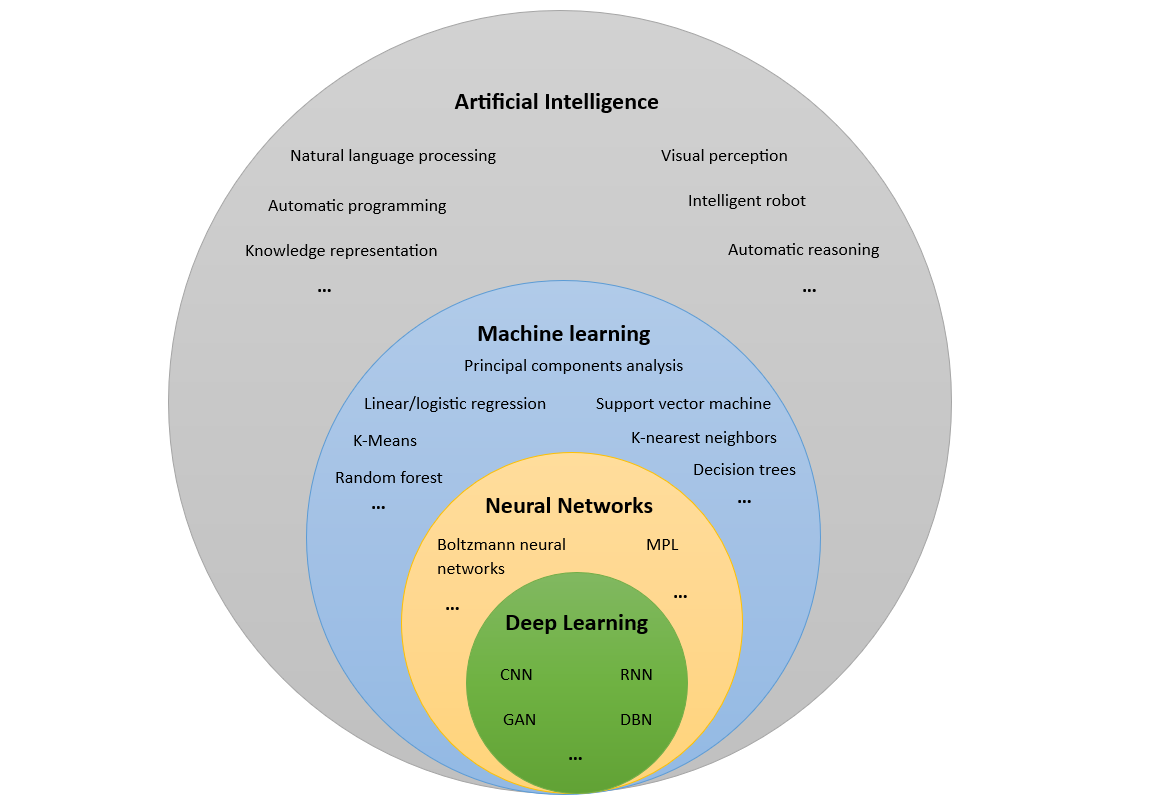
# Introduction:

*Hi everyone! I’m* [*devloker*](https://www.linkedin.com/in/dev-loker)*, and today I’m excited to share a project I’ve been working on: a digit recognition system implemented using pure math functions in Python. This project aims to help beginners grasp the mathematics behind AI and digit recognition without relying on high-level libraries like TensorFlow or PyTorch.*

You can find the complete code on my [GitHub repository](https://www.github/devloker/digits-recognition).

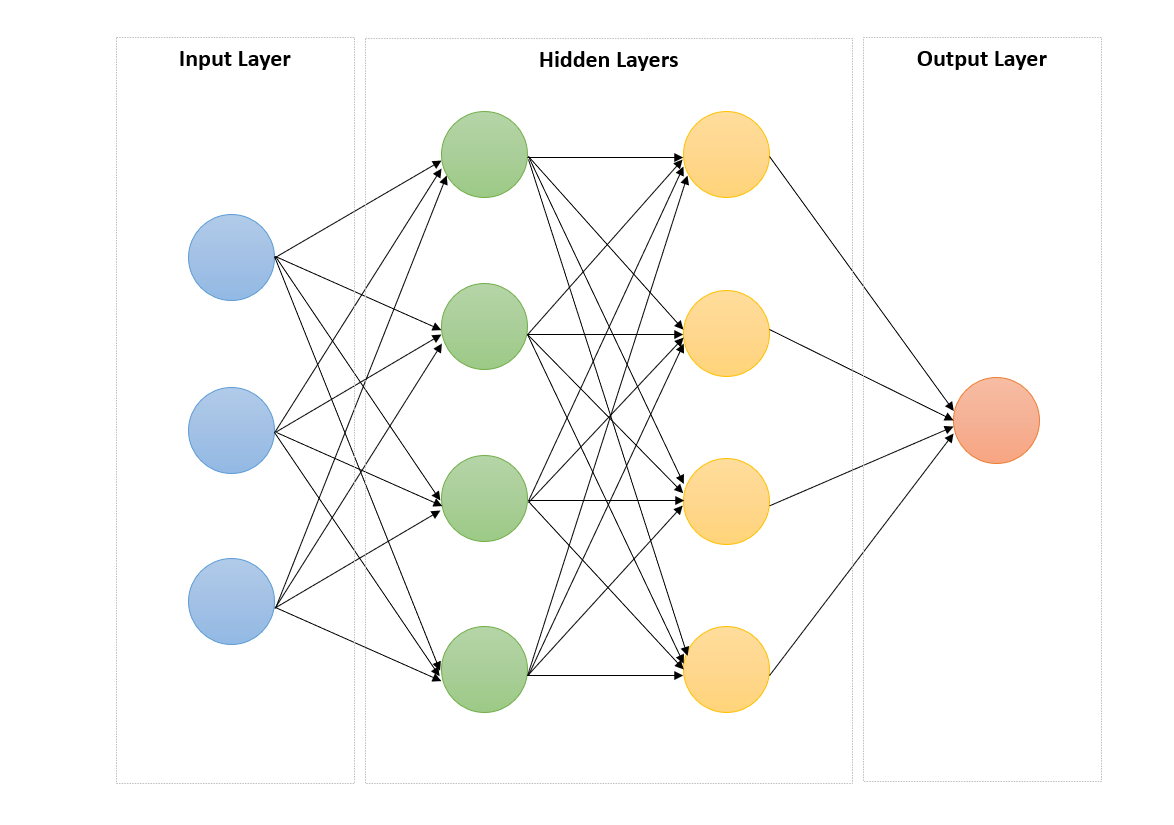
# Fundamental concepts in AI world

Artificial Intelligence (AI): Artificial Intelligence (AI) is a broad field of computer science focused on creating systems capable of performing tasks that typically require human intelligence. These tasks include problem-solving, understanding natural language, recognizing patterns, and making decisions. AI can be categorized into several subdomains, each with its own focus and techniques.



Machine Learning (ML): Machine Learning is a subset of AI that emphasizes the development of algorithms and statistical models that enable computers to learn from and make predictions or decisions based on data. Rather than being explicitly programmed to perform a task, ML models are trained using large datasets, allowing them to improve their performance over time.

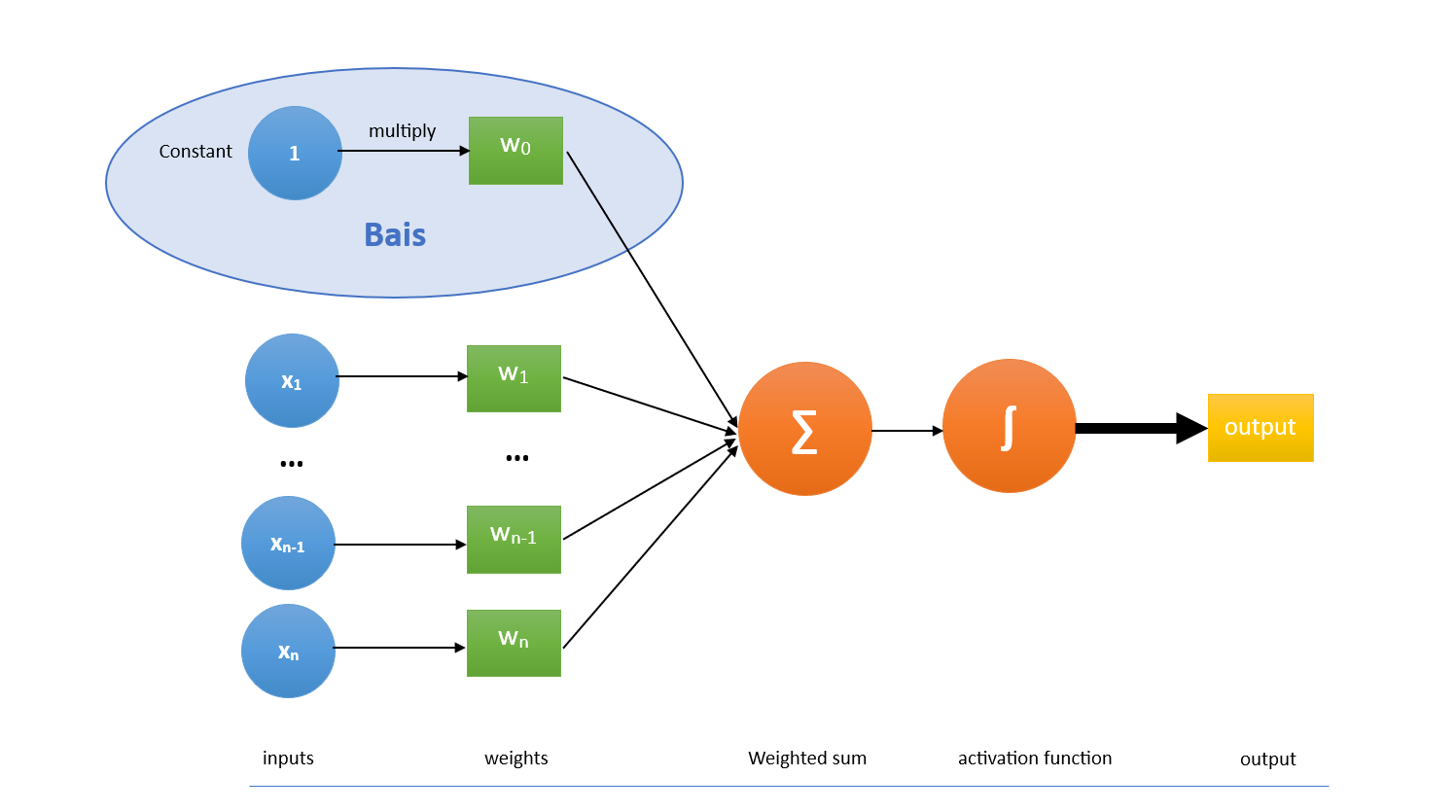
Artificial Neural Networks (ANN): Artificial Neural Networks are a type of machine learning model inspired by the structure and function of the human brain. They consist of layers of interconnected nodes (or neurons), each performing simple computations.



## Neuron

Neurons are the basic building blocks of artificial neural networks, inspired by biological neurons in the human brain. In AI, a neuron is a mathematical function that receives one or more inputs, applies weights to these inputs, sums them up, applies an activation function, and produces an output. In the context of artificial neural networks, a neuron performs the following operations:

* Input Features: The neuron takes multiple input features, each input represents a characteristic or attribute of the input data which represented as x1 ,x2​ , ... , xn​. In the context of digit recognition, these could be the pixel values of an image.
* Weights: Each input feature is associated with a weight w1, w2​, ...,wn​. which indicates the importance of the feature in making the prediction. During training, these weights are adjusted to learn the optimal values.
* Summation Function: Each input is multiplied by its weight, the weighted inputs are summed together, often with an added bias term: z = sum(xi \* wi for xi, wi in zip(x, w)) + b
* Bias: The bias b is an additional parameter that allows to make adjustments that are independent of the input, which helps the model make accurate predictions.
* Activation Function: This function decides whether the neuron should be fired or not based on weighted sum, introducing non-linearity to the model. Common activation functions include softmax, sigmoid, tanh, and ReLU (Rectified Linear Unit).
* Output: the neuron's output is the obtained result after applying the activation function. This output can be fed as input to the next layer of neurons or can be the final output in the case of the output layer, the final output represents the decision or prediction based on the input and the weights, in digits recognition problem, the output might represent a predicted class (from 0 to 9).



These operations work together to enable a neuron to learn and make predictions, while a single neuron can only solve linearly separable problems, combining multiple neurons into layers allows the creation of more complex models capable of solving non-linear problems. This structure forms the basis of multi-layer neural networks used in deep learning.

Deep Learning (DL):

Deep Learning is a subfield of machine learning that focuses on neural networks with many layers (hence "deep" networks). These networks are capable of learning from vast amounts of data and can model complex, high-dimensional patterns. Deep learning has been particularly successful in fields like speech & image recognition, natural language processing, medical diagnosis, and game playing. These models require vast amounts of data and computational power to train effectively but can achieve remarkable accuracy and performance.

Deep learning models consist of multiple layers of neurons. The common types of layers include:

* Input Layer: The first layer, which receives the initial data (e.g., pixel values of an image).
* Hidden Layers: Intermediate layers that transform the input of previous layer into more abstract representations through weighted connections and activation functions.
* Output Layer: The final layer, which produces the final prediction or classification (e.g., the probabilities of each digit in digit recognition).

Training deep networks involves adjusting the weights and biases of the network to minimize the error in predictions. This is done using backpropagation and optimization algorithms like gradient descent.

* **Forward Propagation**: Calculate the output of the network for given inputs.
* **Loss Function**: Measure the error between the predicted output and the actual output.
* **Backword Propagation**: Compute the gradient of the loss function with respect to each weight and bias, propagating the error backward through the network.
* **Weight Update**: during training, the perceptron learns by adjusting its weights and bias based on the difference between the predicted output and the true output.

Types of Deep Neural Networks:

* **Feedforward Neural Networks (FNNs)**: The simplest type where connections between the nodes do not form a cycle.
* **Convolutional Neural Networks (CNNs)**: Primarily used for image processing, recognizing patterns using convolutional layers.
* **Recurrent Neural Networks (RNNs)**: Suitable for sequence data like time series or text, where outputs from previous steps are fed as inputs to the next step.
* **Generative Adversarial Networks (GANs)**: Consist of two networks (generator and discriminator) that compete against each other, useful for generating synthetic data.

# Digits recognition process

**Digit Recognition**: Digit recognition is a classic application of neural networks where the goal is to correctly identify handwritten digits (0-9) from images. This task is commonly tackled using the MNIST dataset, which contains thousands of labeled images of handwritten digits.

**Accuracy**: In the context of digit recognition and neural networks, accuracy is a key metric used to evaluate the performance of the model. It represents the proportion of correctly predicted digits out of the total number of predictions made. High accuracy indicates that the model is effectively learning and generalizing from the training data to make correct predictions on unseen data.

**The Process**: The digit recognition process involves several key steps:

1. **Data Preparation**: Normalizing and reshaping the input images for the neural network.
2. **Model Architecture**: Defining the structure of the neural network, including the number of layers and nodes.
3. **Training**: Feeding the training data through the network, computing the loss (error), and adjusting the weights through backpropagation.
4. **Evaluation**: Testing the trained model on unseen data to assess its accuracy and performance.

By understanding these fundamental concepts, we can appreciate the complexity of digit recognition using neural networks. As we delve into the implementation details, we'll see how mathematical functions and operations come together to create a model capable of accurately identifying handwritten digits.