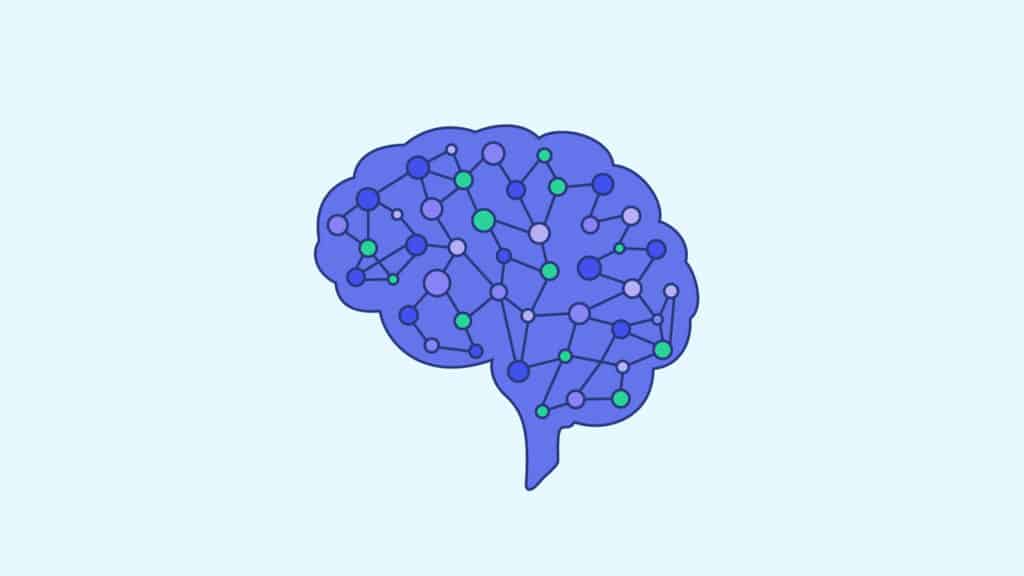
# Plant Disease Detection System for Sustainable Agriculture

## Problem Statement

To develop a CNN-based model capable of detecting and classifying plant diseases from images of leaves of various crops such as apples, cherry, grape, and corn. The model should accurately identify both healthy and diseased leaves while predicting the specific type of disease. The system will aid in precision agriculture by enabling early detection and effective disease management.

## Neural Network (NN)

The main idea behind Neural Networks (NN) is to mimic the human brain. There are three main types of NN:

  
- **Artificial Neural Network (ANN)**:

Processes large amounts of data.

- **Recurrent Neural Network (RNN)**:

Understands language and emotions.

- **Convolutional Neural Network (CNN)**:

Used for vision tasks (e.g., computer vision).

## Pipeline

The steps we follow throughout our project are:

### 1. Data Collection & Data Loading:



* We organize our dataset into three folders: train, test, and validation. Each folder contains subfolders representing categories (e.g., Category 1, Category 2, etc.).  
    
  - Train Folder: Used to train the model with all images.  
  - Valid Folder: Used during training to validate the model.  
  - Test Folder: Used to test the trained model.
* We use validation data for both validation and testing purposes.

**Platform: Google Colab.**

### 2. Uploading the Folder in Google Colab:

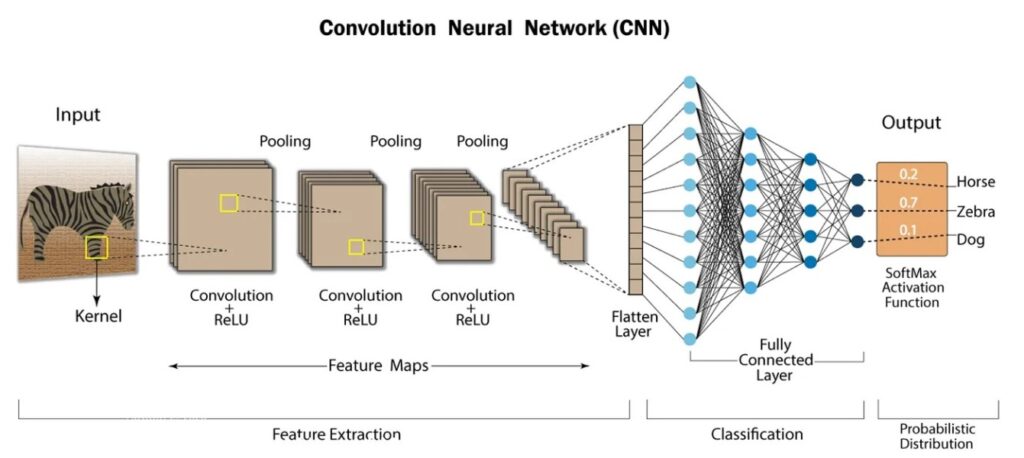


* Steps:  
  - Upload the dataset to Google Drive.  
  - Mount Google Drive on Google Colab.  
  - Unzip the dataset using specific Python code.

### 3. Image Processing & Augmentation:

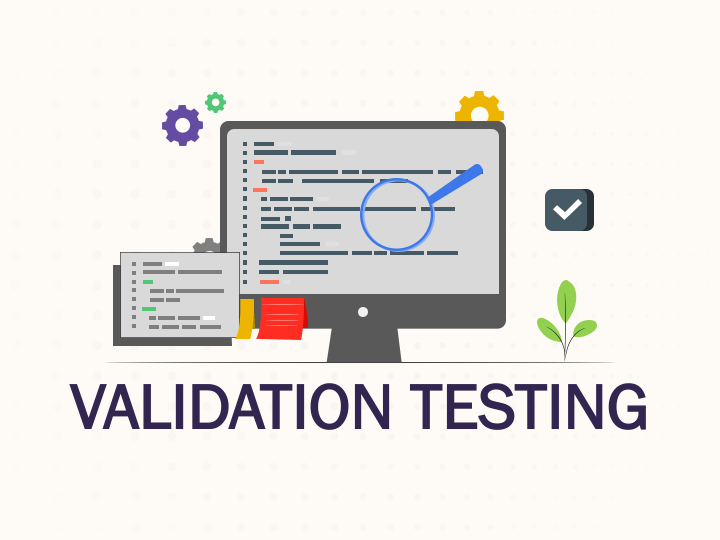
Images of plants may vary in dimensions (height, width). To standardize, we process and resize images accordingly.  
Image Augmentation: The model is trained on images from different views, angles, and zoom levels. We use Python code to perform augmentations like rotations, zooming, and angle shifts.

### 4. Building CNN Model



* We develop a CNN model tailored to classify plant diseases from images.

5. Testing / Evaluating the Model



* **After training, we evaluate the model performance using test data.**