Project Problem Statement

- Agriculture faces increasing challenges due to plant diseases and pest infestations, which significantly affect global food production and farmer income. Traditional methods of disease detection are slow, inaccurate, and rely heavily on manual inspection, making large-scale monitoring inefficient. This leads to excessive use of fertilizers, pesticides, and water, causing environmental damage and economic losses.
- To address this issue, there is a need for an intelligent, automated system that can accurately detect and classify crop diseases from leaf images using **Convolutional Neural Networks (CNNs)**. Such a system can promote **sustainable farming**, enable **early detection** of diseases, support **targeted treatment**, and reduce resource wastage — ensuring better productivity and eco-friendly agricultural practices.

Dataset:

• **Dataset Name:** PlantVillage Dataset
A Comprehensive Image Dataset for Plant Disease Classification and Detection

About Dataset:

This dataset contains over **54,000 labeled images** of healthy and diseased crop leaves across multiple plant species such as tomato, potato, maize, grape, and others. The images are categorized into various disease classes, making it ideal for **machine learning** and **computer vision** projects focused on **crop health monitoring** and **sustainable agriculture**.
It is widely used for building **image classification models** using CNNs or for developing **AI-powered solutions** that support early disease detection and sustainable farming practices.

Source: Kaggle

Plant Village Dataset

Next Steps:

1. **Collect & Prepare Dataset**
    - Download and organize the PlantVillage dataset from Kaggle.
    - Preprocess the images by resizing, normalizing, and splitting into training and testing sets.
2. **Train the CNN Model**
    - Use platforms like **Teachable Machine** or **TensorFlow/Keras** to train the Convolutional Neural Network.
    - Apply data augmentation to improve model accuracy and generalization.
3. **Evaluate & Test the Model**
    - Measure model performance using metrics such as accuracy, precision, recall, and F1-score.
    - Validate results with unseen test images.
4. **Build the Web Interface**
    - Create a simple web application where users can upload a leaf image and get real-time disease detection results.
5. **Test the Complete Application**
    - Test the system with different leaf samples to verify accuracy and response time.
6. **Deploy & Document**
    - Deploy the model and web interface online or locally.
    - Prepare project documentation including dataset details, model results, and sustainability impact.