

Take-Home Task: Crop Disease Classification for Syngenta

Background

As a candidate for Syngenta's Data Science team, you'll work on a real-world agricultural problem: **identifying crop diseases from leaf images** to help farmers take timely action with appropriate crop protection products

Task Overview

Build a machine learning model to classify plant diseases using image data. This simulates Syngenta's work in digital agriculture and precision farming.

Dataset

Kaggle Dataset: [PlantVillage Dataset](#)

- 38 classes of plant diseases across multiple crops (tomato, potato, corn, etc.)
- ~54,000 images of healthy and diseased plant leaves
- Clean, labelled dataset perfect for image classification

(Alternative - Entry Level): [Plant Disease Recognition](#) - pre-processed and split into train/validation/test folders

Requirements

Part 1: Data Exploration

- Load and visualize sample images from 5 different disease classes
- Show class distribution (bar chart)
- Display 2-3 key insights about the dataset

Part 2: Model Building

Build a suitable model

Requirements:

- Train on subset (use 10-15 classes max to save time)
- Image augmentation (rotation, flip, zoom)
- Validation of performance

Part 3: Evaluation & Business Impact

- Confusion matrix
- Show 5 correct and 5 incorrect predictions with images
- **Business recommendation:** Which model would you deploy for Syngenta's mobile app for farmers? Why? (Consider accuracy, speed, model size)

Part 4: Bonus (Optional)

- Create a simple Gradio/Streamlit app for disease prediction
- Suggestions

Deliverables – To be sent until the day before the interview

- 1. Presentation – (20 min)**
- 2. Jupyter Notebook** with code, visualizations, and markdown explanations
- 3. README.md** with:
 - Setup instructions
 - Model performance summary table
 - Business recommendation (1 paragraph)
- 4. (Optional) Saved model file (.h5 or .pkl)**

Time Estimate

- Setup & exploration: 1 hour
- Model training: 3-4 hours
- Evaluation & write-up: 2 hours
- **Total: 6-7 hours**

Tech Stack Suggestion

- Python 3.8+
- TensorFlow/Keras or PyTorch
- pandas, matplotlib, seaborn
- scikit-learn