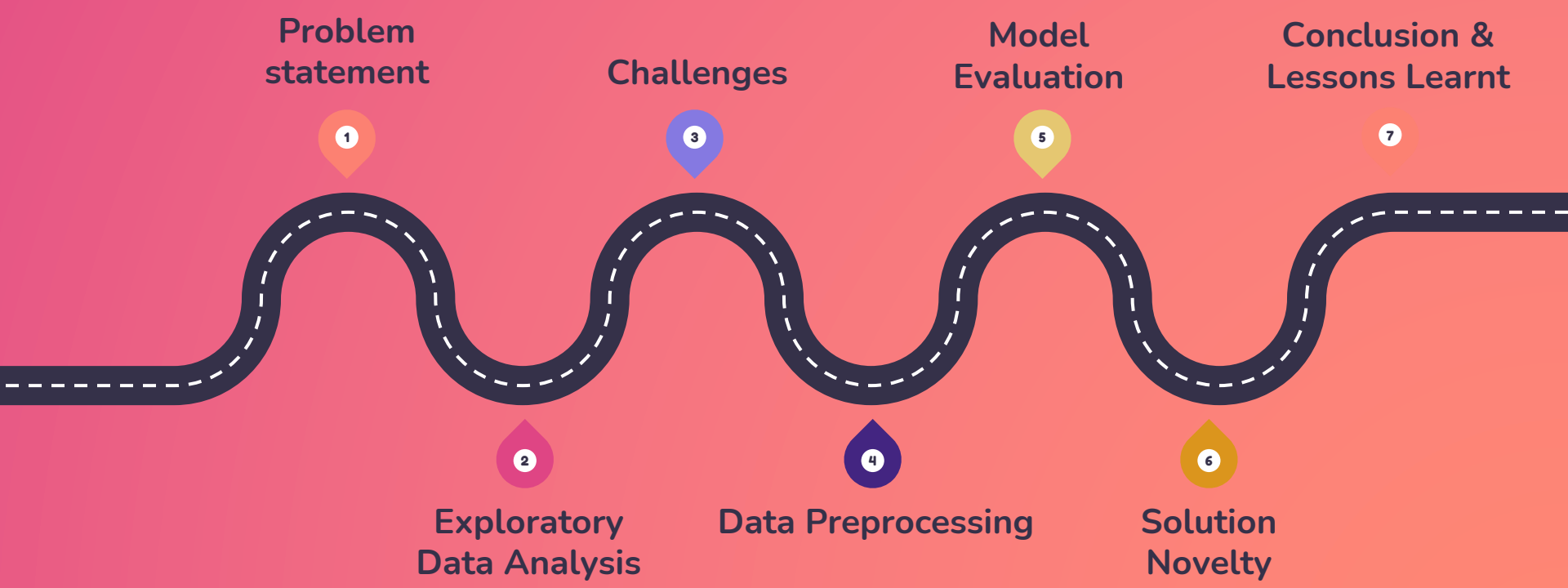
Kaggle - Plant Seedling Classification

Rank 54 / 833 (top 6.5%)

Bansal Aditya
Mantri Raghav
Gupta Jay
Dwivedee Lakshyajeet
Bhatia Ritik

CZ/CE4041 Machine Learning

Agenda


Problem statement
1
Exploratory Data Analysis
2
Challenges
3
Data Preprocessing
4
Model Evaluation
5
Solution Novelty
6
Conclusion & Lessons Learnt
7

Problem Statement

Predict the category (species) of a plant seedling with an RGB image of the plant.

Exploratory Data Analysis

Overview

960
Unique Plants

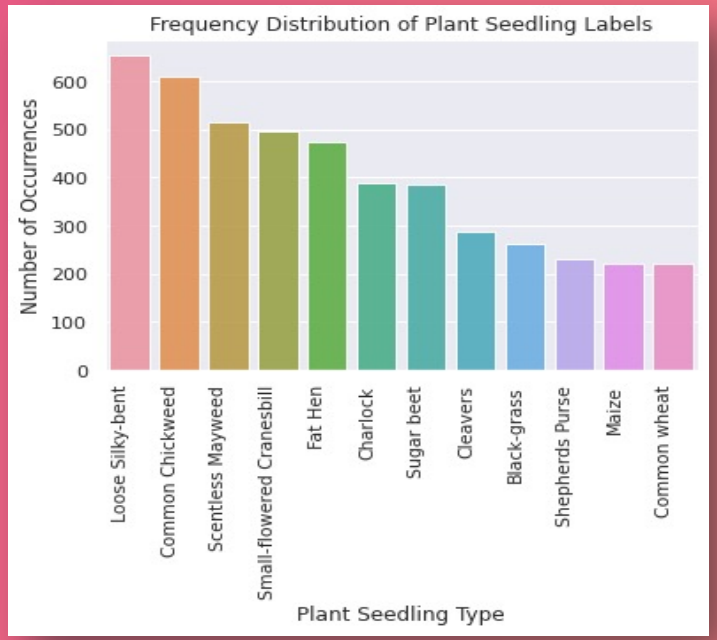
12
Different Species

Variable
Image Sizes

4750
Labelled Images (Train)

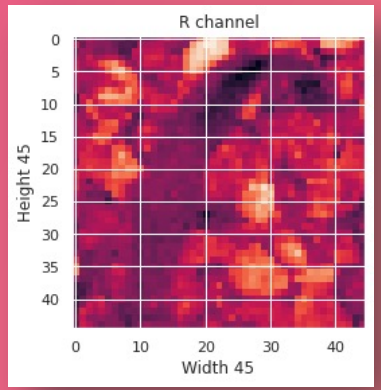
794
Labelled Images (Test)

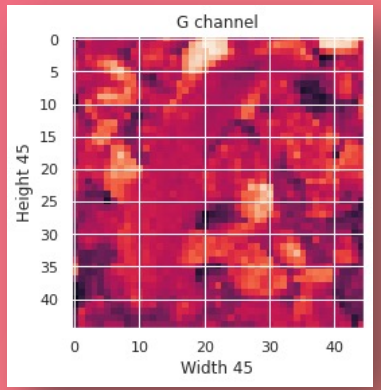
Data Distribution

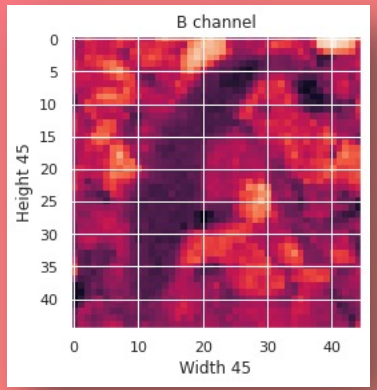


- Non-uniformity in the distribution.
- The number of training examples of some type of seedlings are more than double than some other classes.
- May result in slightly low model accuracy.

RGB Channel Analysis

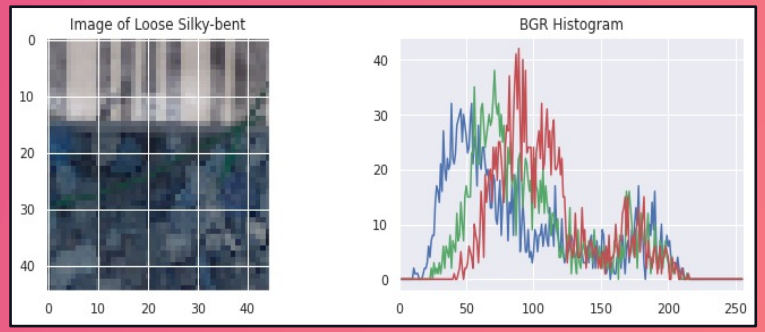


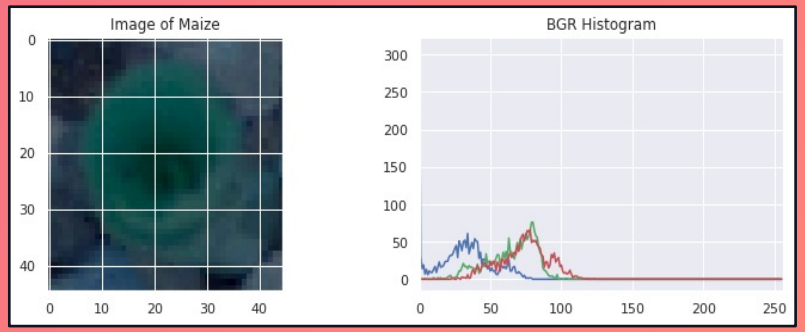


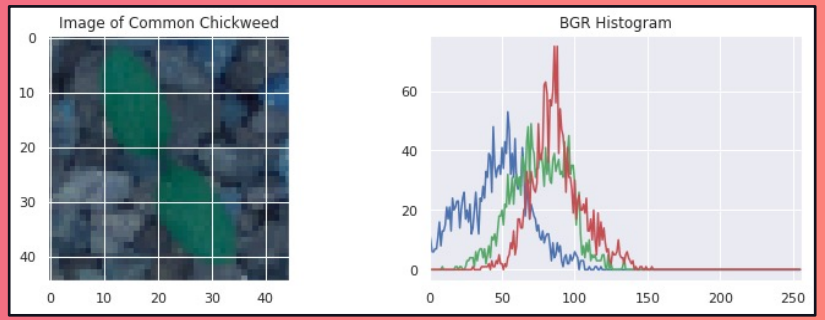


- All the images are coloured with three channels: red, blue, and green.
- Green channel encodes the most amount of information.
- In other channels, the location of the plant is comparatively darker.

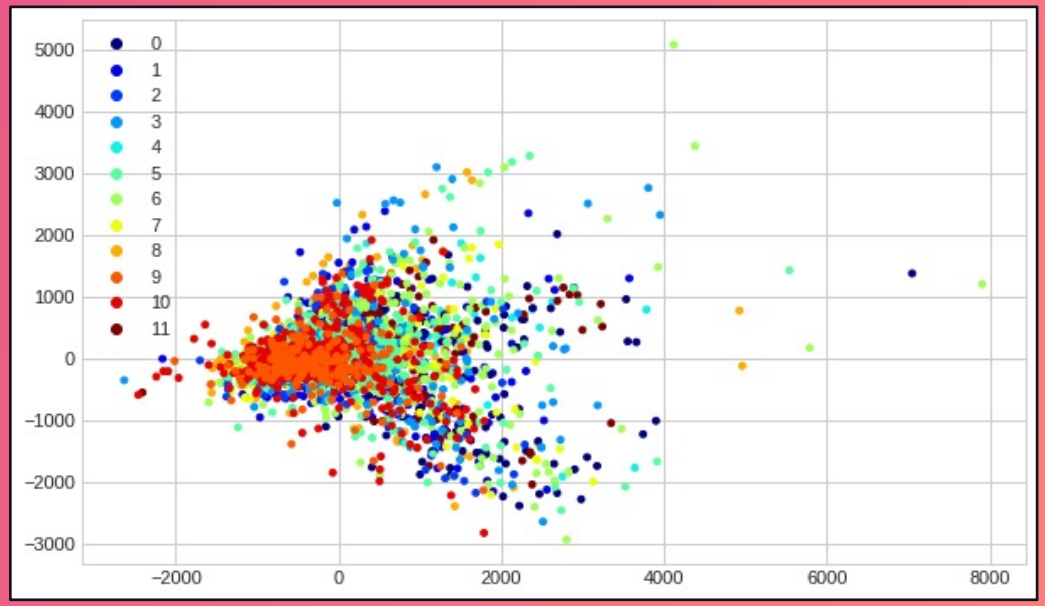
BGR Histograms



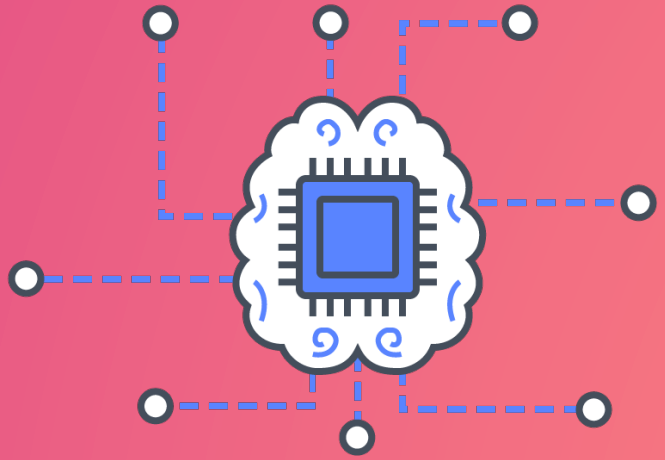




Principal Component Analysis (PCA)




```
Label Legend:
-------------
{'Black-grass': 0,
 'Charlock': 1,
 'Cleavers': 2,
 'Common Chickweed': 3,
 'Common wheat': 4,
 'Fat Hen': 5,
 'Loose Silky-bent': 6,
 'Maize': 7,
 'Scentless Mayweed': 8,
 'Shepherds Purse': 9,
 'Small-flowered Cranesbill': 10,
 'Sugar beet': 11
 }
```



Challenges of the Problem

Challenges

Small Dataset Size

Number of training example sets are not large (4750 images).

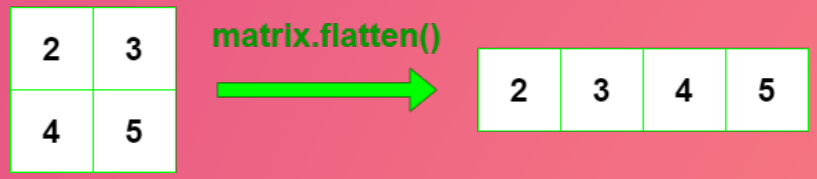
Similar classes

From PCA, Segregation between the different plant seedlings is minute

Model Selection

Difficult to identify the best model for the complex image data, which achieves high test accuracy

Data pre-processing



Resizing and Flattening

Image Augementation

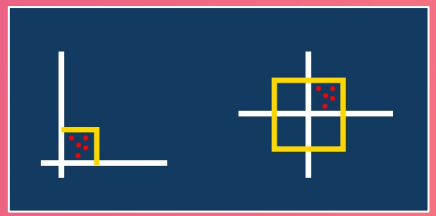
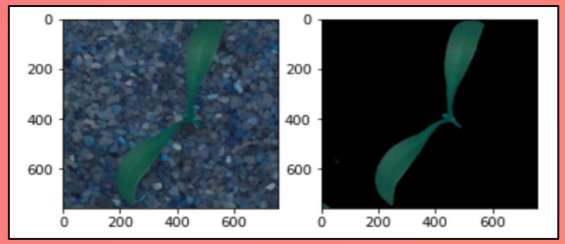


Image Normalization

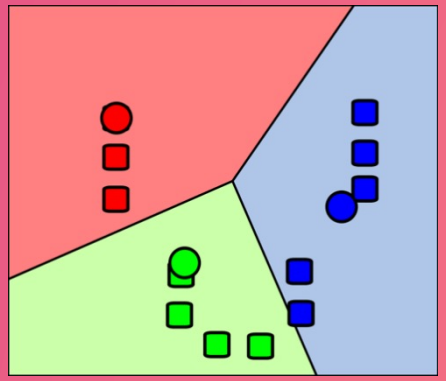


Feature Extraction

Model Evaluation

Approach 1
k-Means Clustering

k-Means Clustering



k-Means Clustering
(Source: Wikipedia)

- Due to its simplicity, established a **fine baseline**.
- The number is **deterministic** which is required by the algorithm.
- K-Means **scales up to large datasets** and guarantees **convergence**, irrespective of the dataset size.

29.116%
Train Accuracy

N/A **(Unsupervised Algorithm)**
Public Rank

Approach 2

k-Nearest Neighbors

k-Nearest Neighbours

0.52959
Private Kaggle F-Score

0.52959
Public Kaggle F-Score

- **Supervised Learning Algorithm,** i.e., utilizes labels of data.
- Relatively **computationally inexpensive**.
- Easy to implement for **multi-class classification** datasets.
- **Flexibility** to choose from a variety of distances to measure from: **Euclidian distance, Hamming distance, etc.**

	# Nearest Neighbors	Weight for Each Point	Distance Metric
Parameters	5	Uniform	Euclidian

Approach 3

Support Vector Machines

Support Vector machines

0.63979
Private Kaggle F-Score

0.63979
Public Kaggle F-Score

- Works well with **unstructured** and **semi-structured** data like images
- **Supervised learning** algorithm like K-Means
- Computationally inexpensive **(O(N2K))**
- **Flexibility** to choose from kernels e.g., Linear, rbf, poly, etc.

	Kernel	Gamma
Parameters	Linear	Auto

Approach 4

Convolutional Neural Network

Convolutional Neural Network

0.60831
Private Kaggle F-Score

0.60831
Public Kaggle F-Score

- Effectively captures **spatial information** by using convolutional operations in the images
- Need **minimal pre-processing**
- Requires **high computational power** and time

Model Architecture

- Three 2D convolutional layers, with a Max-pooling Layer,
- RELU activation for non-linearity
- Dropout layer to prevent over-fitting in the model.

	Epochs	Batch Size	Optimizer	Loss Function	Learning Rate	Validation Data set
Parameters	15	32	Adam	Sparse Categorical Cross Entropy	0.001	20% of train set

Approach 5
Xception

Xception

0.97607
Private Kaggle F-Score

0.97607
Public Kaggle F-Score

- **Transfer learning** on pre-trained model (ImageNet).
- Already learnt complex **feature extraction** from large dataset.
- **Learning rate scheduling** is used to prevent overshooting the minima
- **Image augmentation** used to virtually increase training set size
- **Test time augmentation** used to get multiple outputs for voting

	Epochs	Batch Size	Optimizer	Loss Function	Learning Rate	Learning Rate Decay
Parameters	30	16	Adam	Sparse Categorical Cross Entropy	0.001	0.9

Approach 6
Inception – ResNet – v2

Inception – ResNet – v2

0.98488
Private Kaggle F-Score

0.98488
Public Kaggle F-Score

- Combines best of **Inception v3** and **ResNet architectures**
- **Faster training** and **deeper network** due to ResNet, which skips immediate neurons

<u>Model Architecture</u>

- 2 Dense layers:
  - 1 with softmax activation
  - 1 with ReLU activation
- 2 Dropout layers *(probability = 0.5)*
- L2 regularization (*penalty* = 10^{-5})
- Base Inception – ResNet architecture

	Epochs	Batch Size	Optimizer	Loss Function	Learning Rate	Validation Data set
Parameters	100	16	Adam	Categorical Cross Entropy	10^{-10} to $4 * 10^{-5}$	1% of train set

Key Observations

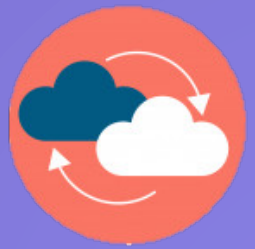
Exploratory Data Analysis helps

Importance of **Data Pre-processing**

Overfitting on training data reduces test accuracy

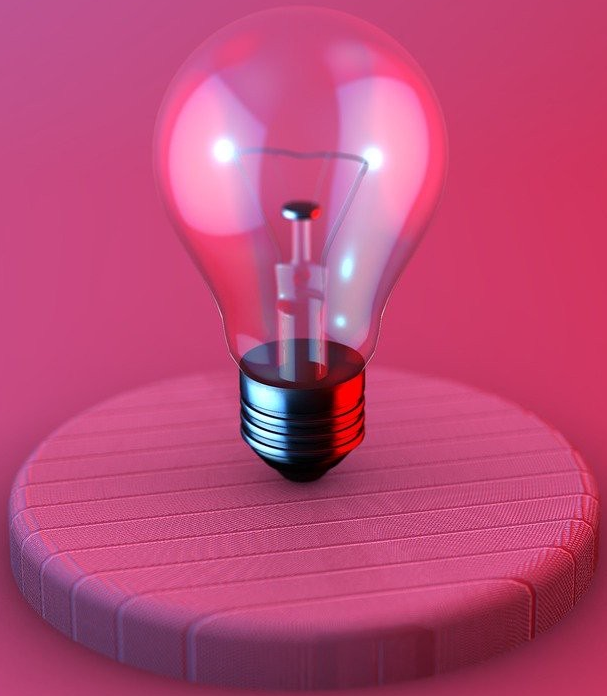
CNNs perform better than traditional ML

Cases where **Ensemble Learning** doesn't help



Transfer Learning on existing architectures

Solution Novelty



Solution Novelty

1 Ensemble learning

Voting for classification tasks to increase the accuracy by using outputs from multiple trained instances

2 Test-Time Augmentation

Creating augmented copies in the test set, making prediction and returning an ensemble

3 Keras Callbacks

Reduces learning rate when loss stops decreasing.

Stops training early when loss has converged.

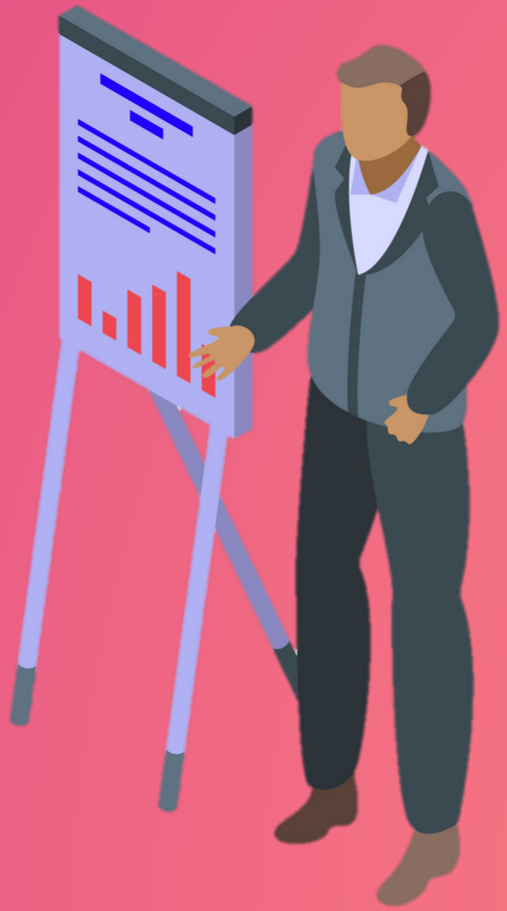
4 Dropouts & L2 Regularization

Regularisation to avoid overfitting.

- Dropouts
- L2 regularization

Leaderboard

Method	Score	Public Rank
kNN	0.52959	764
SVM	0.63979	752
CNN	0.60831	757
Xception Net	0.97607	193
Inception-ResNet-v2	0.98488	🥳 54



Lessons Learnt

Incremental Methodology

Tried six different approaches and tuned various hyperparameters.

Patience & Time

up to 15 GB Memory

used by Kaggle Notebook's GPU and TPUs

1.69 GB
Dataset Size

> 4 hours
Training Time / Model

Team synergy

Thank You!