**Plant Seedling Classification**

**Capstone: Machine Learning Engineer Nanodegree**

**Manvindra Singh**  
January 31st, 2018

## Project Report

### **Definition**

Project Overview: Can you differentiate a weed from a crop seedling? Many weed seedlings can be mistaken for crop seedling as they look alike, it becomes hard to differentiate weed from crop. They can be distinguished when they become mature plant but till then soil quality and crop yield will be impacted. For example, Carrot shows relatively slow development in early growth stages. Hence, competing weeds may overtake the crop plant and limit its access to resources such as sunlight, moisture, and nutrient. Thus, weeds may cause major yield losses, if uncontrolled.

Comparing few seedlings:

Wild Turnip can be mistaken for Charlock, which has more grass-green true leaves with many hairs. With Oil-seed Rape, which has almost smooth true leaves. Also, with Wild Radish, which has more grass-green true leaves with many hairs. In all crops in which this plant is found it is an injurious weed, which causes losses.

All the above weed looks alike to Shepherd’s-purse a member of the same mustard family. Shepherd’s purse has been used as a medicinal herb often recommended as a treatment for both internal and external bleeding.



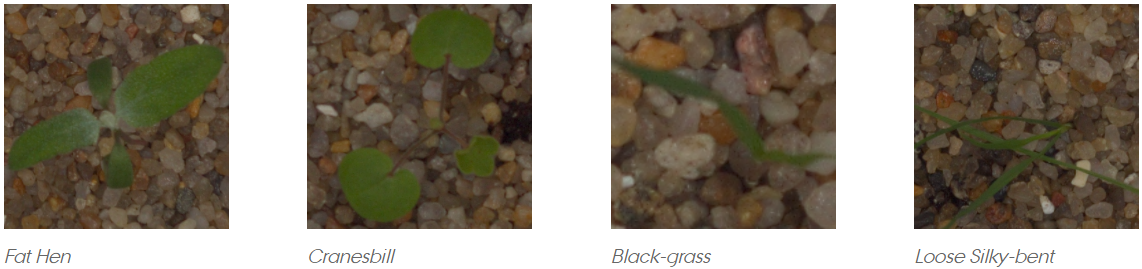
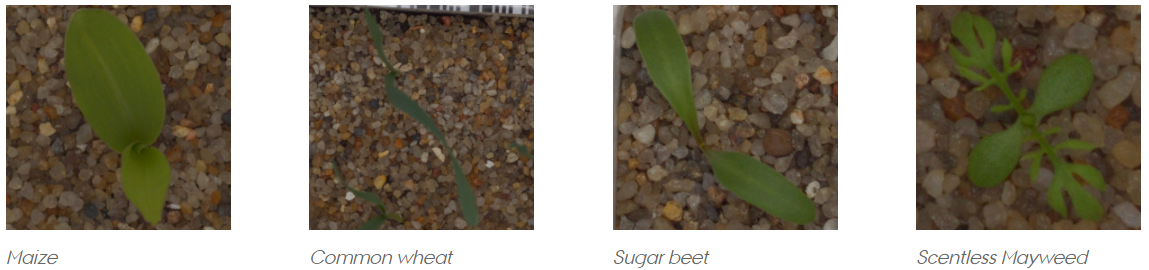
The ability to distinguish effectively can mean better crop yields and better stewardship of the environment. The goal of this project is to classify seedling images into its species. This problem is based on Kaggle Competition called “Plant Seedlings Classification”. ( <https://www.kaggle.com/c/plant-seedlings-classification> ).

The Aarhus University Signal Processing group, in collaboration with University of Southern Denmark, has released this dataset containing plants belonging to 12 species at several growth stages. I obtained the dataset through Kaggle which contains 4750 images in training set and 794 in test set. Number of images varies across species and a the size of images varies from 49 X 49 to 3457 X 3991 but most images have aspect ratio of 1.0

Problem Statement

As said earlier, many seedlings of crops look alike another crops. Its difficult to distinguish them with naked eye even for experienced farmers. Each seedling can have different growth rate, some may become larger than other seedling in short time. Seedling from a family may resemble other plant seedling in different growth stage. Maybe 4 months old seedling of a plant resemble 2-month-old seedling of another plant. Attempt here would be to classify seedlings images into its plant species.

Below are few images from seedling for each species in dataset



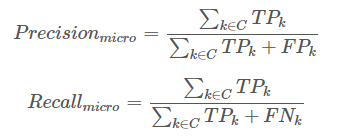
Some of the plant seedlings are similar in appearance. Shepherd’s purse is used as herbal medicine whereas it looks alike charlock is a common weed of cornfields.

Goal is to classify images of seedlings to its species. This make it a computer vision challenge. Deep learning techniques have been very popular and has outperformed traditional approaches in model performance, feature extractions. They have been used to win one biggest image classification competition -ImageNet. Convolutional neural networks(CNN) unlike other deep learning architect are designed to handle spatial information in images very well. I will build a simple CNN using 3 convolution layers each followed by max polling layers while also using different parameters settings. This CNN architecture will be made deepen by including more filters and will be made wider by adding more layers. Then I will also compare these models with models from ImageNet Challenge (Ex ResNet, VGG16, Xception ) using transfer learning technique. After these I will determine the best model using mean F-score.

Metrics

Micro-Averaged F1-score

F-score is weighted average of precision and recall given by following equations



Here

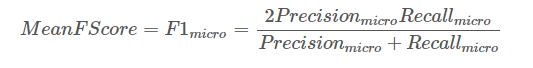
k is class

TP: True Positive, case where correct species predicted

FP: False Positive, where seeding incorrectly classified to a species

FN: False Negative, cases where seedling is not classified to its species

Once we have these, Mean F score calculated as



Since we have uneven class distribution and cost of False positive and False negative are very different, we should look at both Precision and Recall.

### **Analysis**

Data Exploration

The Aarhus University Signal Processing group, in collaboration with University of Southern Denmark, has released this dataset containing images of approximately 960 unique plants belonging to 12 species at several growth stages. Total number of training images are 4,750 and Test images are 794 without labels used only for submitting result to Kaggle leaderboard. I will use 4,750 training images to train, validate and test the model.

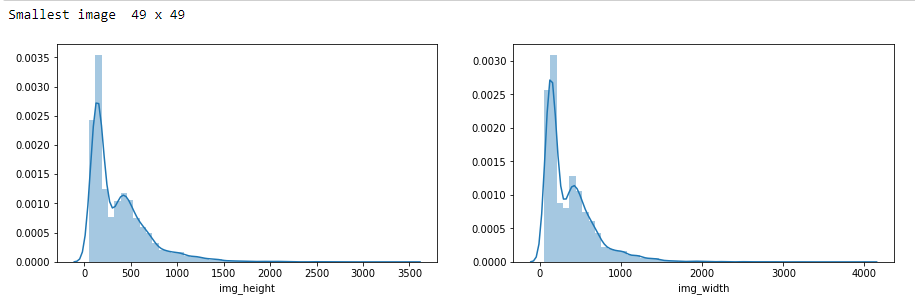
List of species along with number of images given in training set is in the following table

|  |  |
| --- | --- |
| Species | Number of Images |
| Sugar beet | 385 |
| Loose Silky-bent | 654 |
| Scentless Mayweed | 516 |
| Maize | 221 |
| Shepherds Purse | 231 |
| Cleavers | 287 |
| Charlock | 390 |
| Small-flowered Cranesbill | 496 |
| Fat Hen | 475 |
| Common Chickweed | 611 |
| Black-grass | 263 |
| Common wheat | 221 |
| Total | **4,750** |

Number of images in each category varies from 221 to 654. We can balance the dataset by taking only 221 images in each dataset but here we will not make full use of dataset. Or we can augment data and make each category have 654 images. This way all classes will have same number of training images.

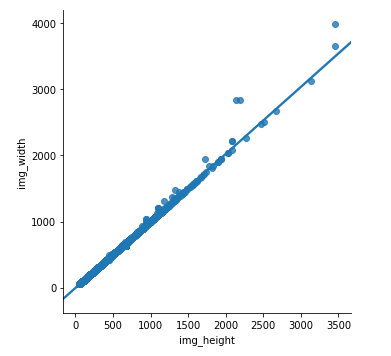
Exploring image size.

Fig 1: Distribution Plot of Image Height and Image Width.



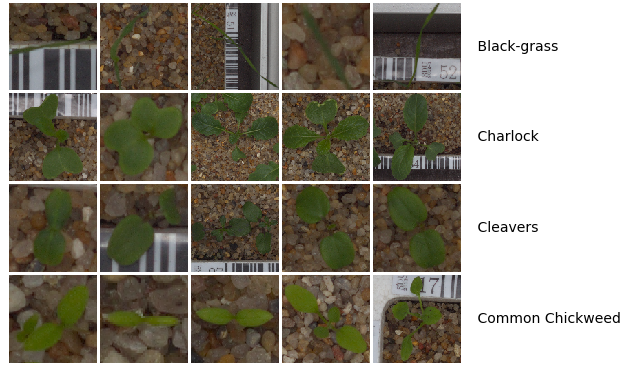
Smallest images in the data is 49\*49 and largest goes up to 3500px.

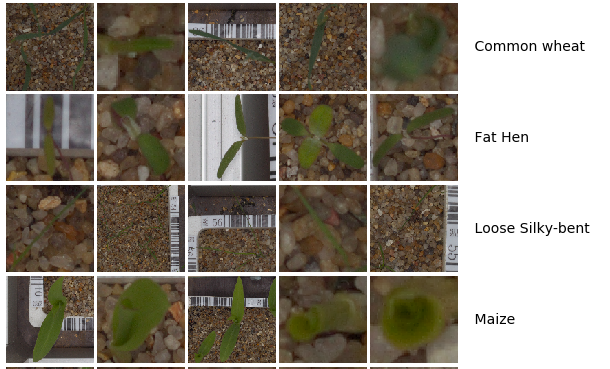
Fig 2: Aspect Ratio of Images

Expect than 4-5 images, rest have aspect ratio 1:1. We need to rescale the image for neural network training. As seen from distplot above, many images are around 100px-150 px. Smallest image is 49px X 49px. I am rescaling them such that I shouldn’t scale smaller images to much large size and loss details. Therefore, I choose 128px X 128px as size to rescale images for training.

Exploratory Visualization

Fig 3: 5 Samples images form each species







Input images are Color Images (RGB) containing image of seedling along with the box, soil, stones and scale markings. Black Grass and loose silky-bent seedling have so narrow leaves that majority of image is stone and box. All the plant leaves are green in color and background doesn’t have anything green that matches the seedling color. This will help us in removing the background box, sand, stone etc.

Algorithms and Techniques

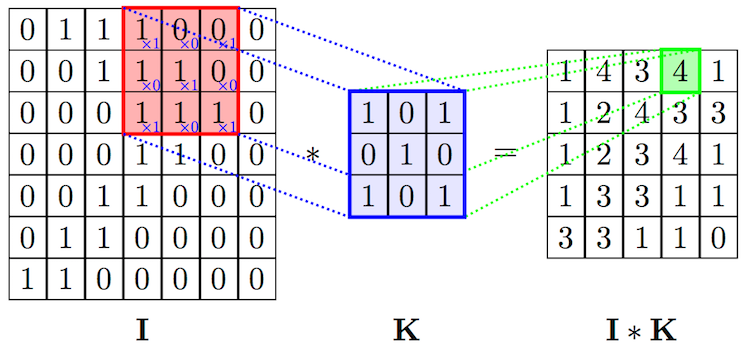
We are dealing with image classification, so this is generally treated as computer vision challenge. It’s easy for Human to classify images, for machine it can be uphill task. Neural Network are designed to mimic human brain learning process. Deep learning techniques have been very popular and has outperformed traditional approaches in model performance, feature extractions. I will be using Convolutional neural networks(CNN) to train the model. CNN unlike other deep learning architect are designed to handle spatial information in images very well. In neural networks, the input is a vector, in CNN the input is a multi-channeled image (Here it is 3 channel). CNN have been used to win biggest image classification competition -ImageNet which is like Olympics for computer vision expert

CNN consists of several layers. These layers can be of three types

1. **Convolutional(Conv)**: CNN has got its name from this layer. Primary function of this to extract features from image. Convolutional layers consist of rectangular grid of neurons. Each neuron takes input from rectangular section of previous layers.

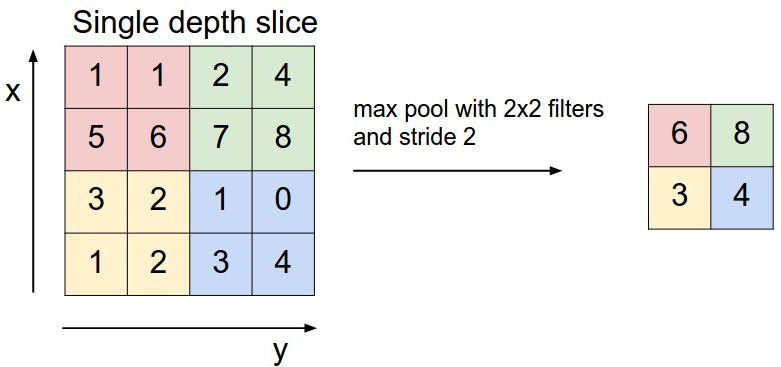
To build Conv Layer:

1. Image is converted into pixel values. (3 Matrix for each channel)
2. Make a filter that is also a matrix which will convolve(slide) over the image
3. We slide the filter over image and compute element wise multiplication and add the multiplication output to get the Convolved feature which will be also be a matrix



1. **Pooling Layer(Pool):** Each filter will result in feature map; many feature maps will increase the number of parameter. To reduce it, we have pooling layer. Pooling layer take small rectangular blocks from the convolutional layer and subsample it to product single output from that rectangular block. We can take maximum, average or linear combination of values in the block.

Fig 5: Max Pooling Layer (http://cs231n.github.io/convolutional-networks/)



Global Max pooling layers also exists in some network, its ordinary max pooling layer with pool size (rectangular blocks) equal the size of the input. In above diagram, max Global Max pooling layers will result in number 8 as output rather than 2 by 2 matrix.

1. **Fully Connected Layer(FC):** “Fully Connected” implies every neuron in the previous layer is connected to every neuron on next layer. It is the traditional multi-layer perceptron. Output from Conv-Pool layer above represent high-level features of the input image. FC layer will be used to use these features to classify input image into various species of plants. Fully connected layers are not spatially located anymore so there won’t be any more Conv-Pool layers after it.

Operations that are applied on layers and forms part of the network

1. Dropout: Dropout layer are used to prevent network from overfitting. Dropout alternately randomly disable neurons in training.
2. Activation: Apply nonlinear operation to output of Conv or FC layers.
   1. ReLU: Stands for Rectified Linear Unit. It is a nonlinear element wise operation and replace all negative pixel value in the feature map with 0.

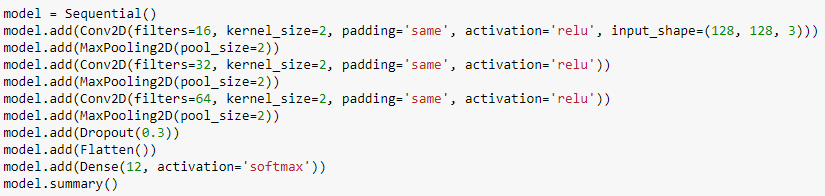
Output = max (0, input)

* 1. Softmax: Calculates probabilities of each target class over all possible target class. Sum of all probabilities will be equal to 1. Ratio of exponential of that input value to sum of exponential of all inputs value is the output of softmax.

1. Optimizers: Optimization function minimizes error function which is mathemical functional dependent on internal parameter (Weight and bias) here. There internal parameter are learned and updated in the direction of optimal solution, here it is minimizing the loss by network training process. I will try different optimizers and compare the result.

Benchmark

No benchmark is mentioned in Kaggle competition page. But we can still consider better than random chance of 1/12=0.08 as benchmark. But that is too less for any practical use. So, I have developed a simple CNN model with Three convolutional layers with ReLU activation each following with max pooling layers. I achieved mean F score of 0.657 and accuracy of 0.66. This accuracy is better than random chance of 1/12. I will use this basic CNN model with Mean F score of 0.66 as benchmark model.



I will improve the model by

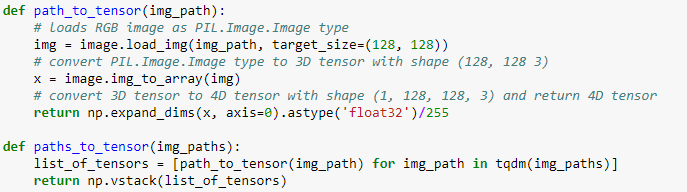
1. Data Augmentation
2. Making CNN denser: add more layers
3. Making CNN wider: add more filters
4. Using different optimization function

Then I will also use transfer learning by using pretrained model like ResNet50, VGG16, InceptionV3 and Xception. Finally, I will use the best model to predict images in test set and post the results on Kaggle to check if I achieved better score than benchmark

### **Methodology**

Data Preprocessing

As discusses earlier images are RGB channel and have different sizes. Also, I discussed that I will choose 128px X 128px as image size. I am using Tensorflow in backend of Keras, which require a 4D array as input. I will convert the image into Numpy array of dimension (Number of images, 3 , 128, 128) also called 4D Tensor ,suitable for supplying to a Keras CNN. Pixel values in 4D Tensor are convert to float32 format and normalized by diving by 255. Following function are used for this purpose.



For loading dataset, load\_dataset function is defined with take path of the dataset as input and return files path (path of image) and target(category of image species). This function also convert target into binary class matrix of (number of images, 12) dimension using np\_utils.to\_categorical.

