**Work Journal**

Team: The Deep Divers

Members: Haoda Song, Siyuan Li, Yuyang Li

# **Info Summary**

## **Overview and requirements**

<https://docs.google.com/document/d/1vLa1HNtYBDl7JfoD9_n_HuVW-bvGhZTb/edit#>

## **Google Drive Data Access**

<https://drive.google.com/drive/u/0/folders/0AMaE9IE5oMG-Uk9PVA>

## **Goal**

Can you identify a problem with a cassava plant using a photo from a relatively inexpensive camera? This competition will challenge you to distinguish between several diseases that cause material harm to the food supply of many African countries. In some cases the main remedy is to burn the infected plants to prevent further spread, which can make a rapid automated turnaround quite useful to the farmers ([Kaggle](https://www.kaggle.com/c/cassava-leaf-disease-classification/data)).

## **Github**

<https://github.com/siyuanli1202/DATA2040_Midterm_Project>

## **Google Colab**

<https://colab.research.google.com/drive/1zWUkOCkTl1btajaMNX8lJwosYJJadAOT#scrollTo=YEThTavokywC>

## **Medium**

<https://yuyang-li1.medium.com/cassava-leaf-disease-classification-part-i-d0ddc94f1635>

## **Kaggle**

<https://www.kaggle.com/haodasong01/data2040-deep-divers>

# **WEEK 1 (02/18 - 02/21)**

## **To-do List:**

* + Complete Blog Post #1. (Due at 02/21)
  + Data prepossessing. Training and validation split for model fit. *(****Haoda Song, Siyuan Li)***
  + EDA (Many previous amazing EDA could be found [here](https://www.kaggle.com/c/cassava-leaf-disease-classification/notebooks?competitionId=13836&sortBy=voteCount)). Summarized label distributions and demonstrated sample images for clarification. ***(Haoda Song, Siyuan Li)***
  + Baseline Model Identification. Constructed CNN model structure and hypertune the parameters in hidden layers for model improvement. Achieved the current best training accuracy as 74.63% and validation accuracy as 73.86%. ***(Haoda Song, Yuyang Li, Siyuan Li)***
  + Analysis of Baseline model results. Build data visualization of training and validation accuracy trends over epochs. ***(Yuyang Li, Haoda Song)***
  + Other stuff: Cited all resources used, included code fragments from notebooks on Kaggle. Discussion of model training history using Tensorboard.dev. Researched on advanced DL methods for future direction. ***(Haoda Song, Yuyang Li, Siyuan Li)***
* **Next Steps:**
  + Work on solving the data imbalance issue. ***(Haoda Song)***
  + Implement EfficientNet and GoogleNet algorithms with Keras package.(***Yuyang Li, Siyuan Li)***

# **WEEK 2 (02/21 - 02/28)**

## **To-do List:**

* + Reloaded the image data in format of *‘tfrecord’*. *(****Haoda Song, Yuyang Li)***
  + Reproduced everything we have done in the previous week, which included data preprocessing, augmentation and baseline model. *(****Haoda Song, Siyuan Li)***
  + Used fine-tuning method to transfer the baseline model to different model structures, including the structures that inspired by ResNet and GoogLeNet. ***(Haoda Song, Yuyang Li, Siyuan Li)***
  + Implemented the BatchNormalization and Dropout to prevent overfitting of deep NN architectures. ***(Haoda Song, Yuyang Li, Siyuan Li)***
  + Researched on the algorithm and structure of EfficientNet. ([Link of this](https://www.kaggle.com/c/cassava-leaf-disease-classification/notebooks?competitionId=13836&sortBy=scoreDescending)) ***(Haoda Song, Yuyang Li, Siyuan Li)***
* **Next Steps:**
  + Build self-designed neural network models based on different architectures in order to achieve better prediction performance.
  + Continue the research on Cropping, and use majority voting in the test set to classify images.

# 

# 

# 

# 

# 

# **WEEK 3 (02/29 - 03/07)**

## **To-do List:**

* + Continue to use the fine-tuning method to transfer the baseline model to different model structures and tune the hyperparameters, including the structures that inspired by ResNet and GoogLeNet. ***(Haoda Song, Yuyang Li, Siyuan Li)***
  + Continue to research the different NN architectures.
* **Next Steps:**
  + Preparing Midway Blogs and updating the kaggle notebook.
  + Clearly state the reasoning of hyperparameters and architectures chosen.
  + Build the visualization of the final architecture that is determined to use.