QMST 5367 Machine Learning

Plant Image Processing

Shainan Agrawal, Chi Bao, Christian Martin, Taja Castilleja

Hypothetical Business Scenario

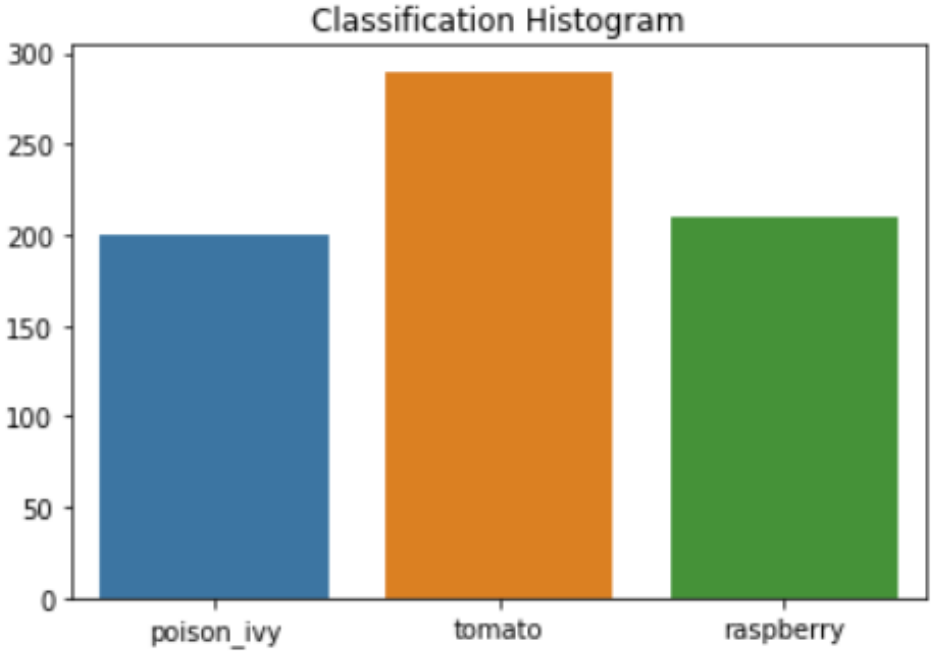
- Startup company developing an app to identify the plants using machine learning upon taking a photo
- Target customers: Girl/Boy Scout groups, avid hikers, forest rangers, hobbyist foragers, etc.
- We have premium and free services
  - Premium service: expanded plant image recognition
  - Free service: plant leaf recognition will be limited to 3 classifications
  - Tomato, raspberry, poison ivy

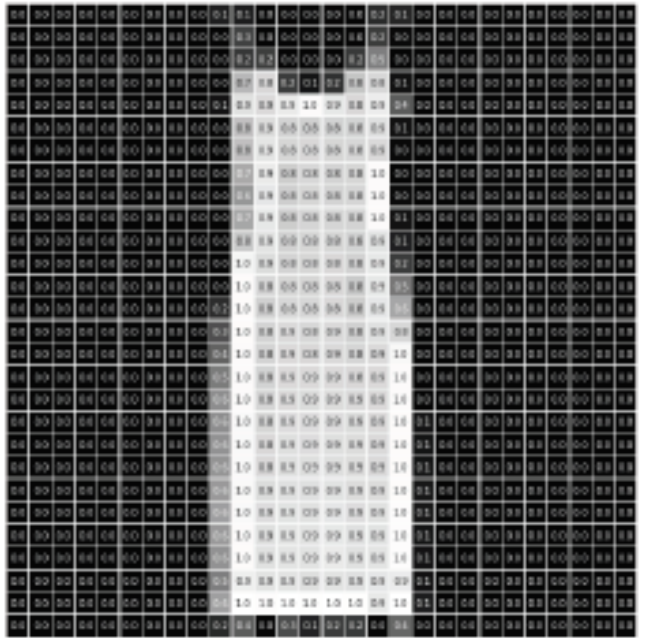
Datasets and ML Algorithm

- We gathered 200 - 300 images per classification: tomato, raspberry, poison ivy
- By training our model with the image datasets, we can accurately identify types of plants. This removes effort and time to research plant information and type
- App will use supervised machine learning since in the end the user knows the plant type. The algorithm will choose the best match using statistical significance

Image Visualization / Transformation

- Original image → Black and White image





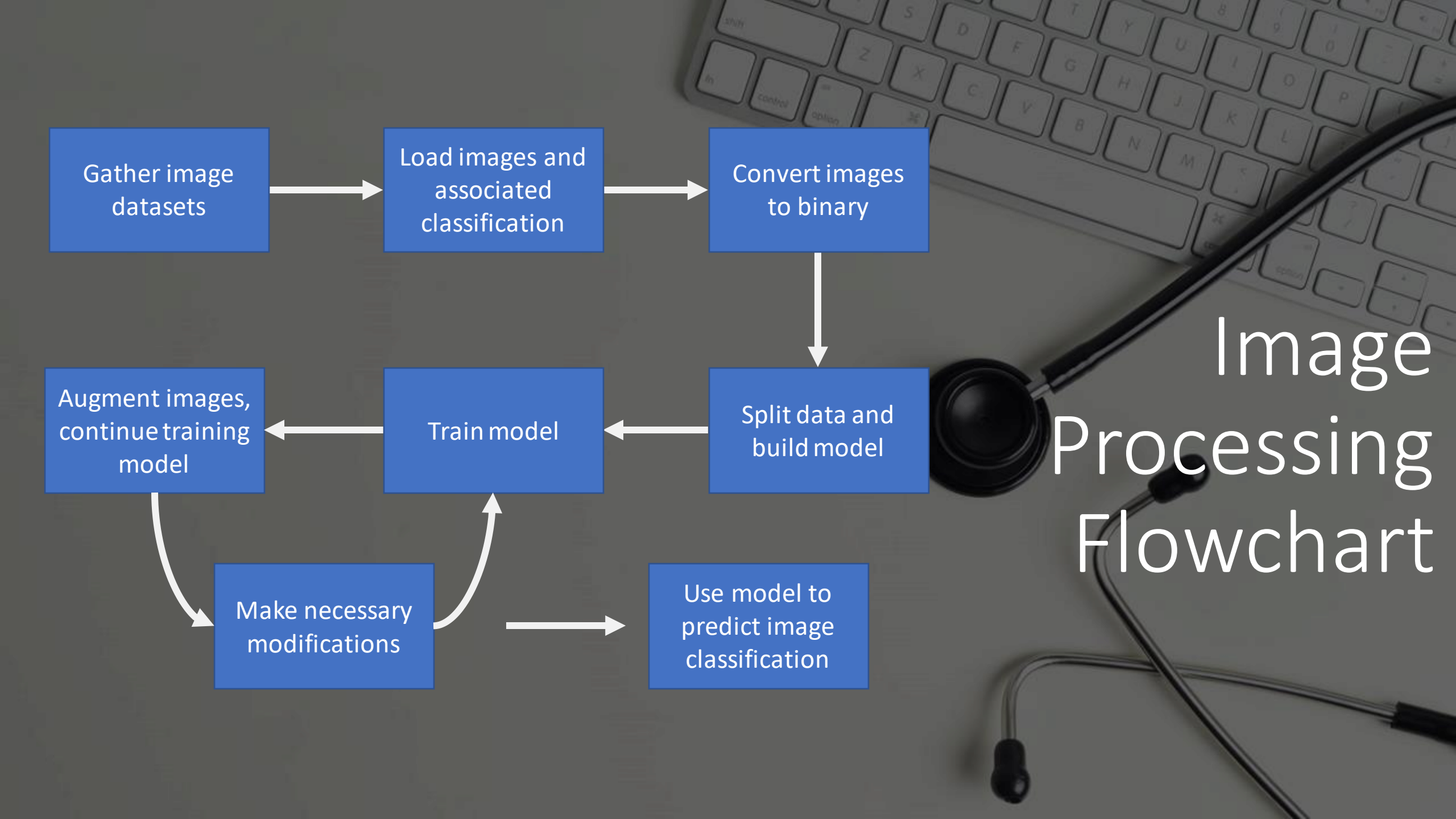
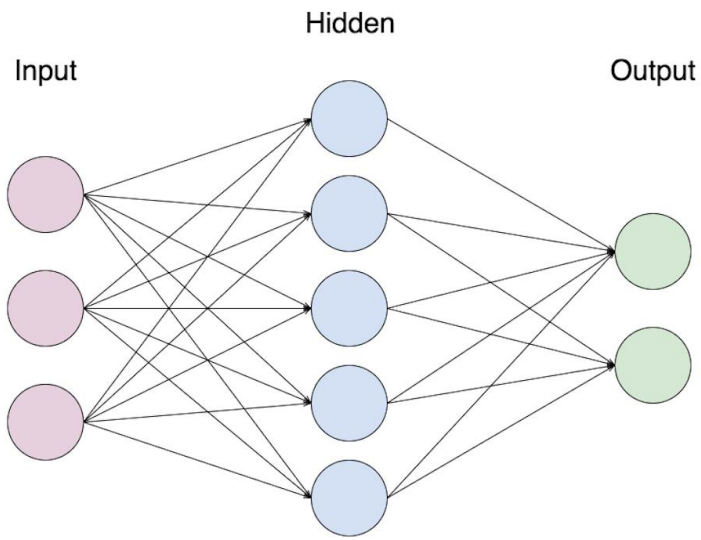
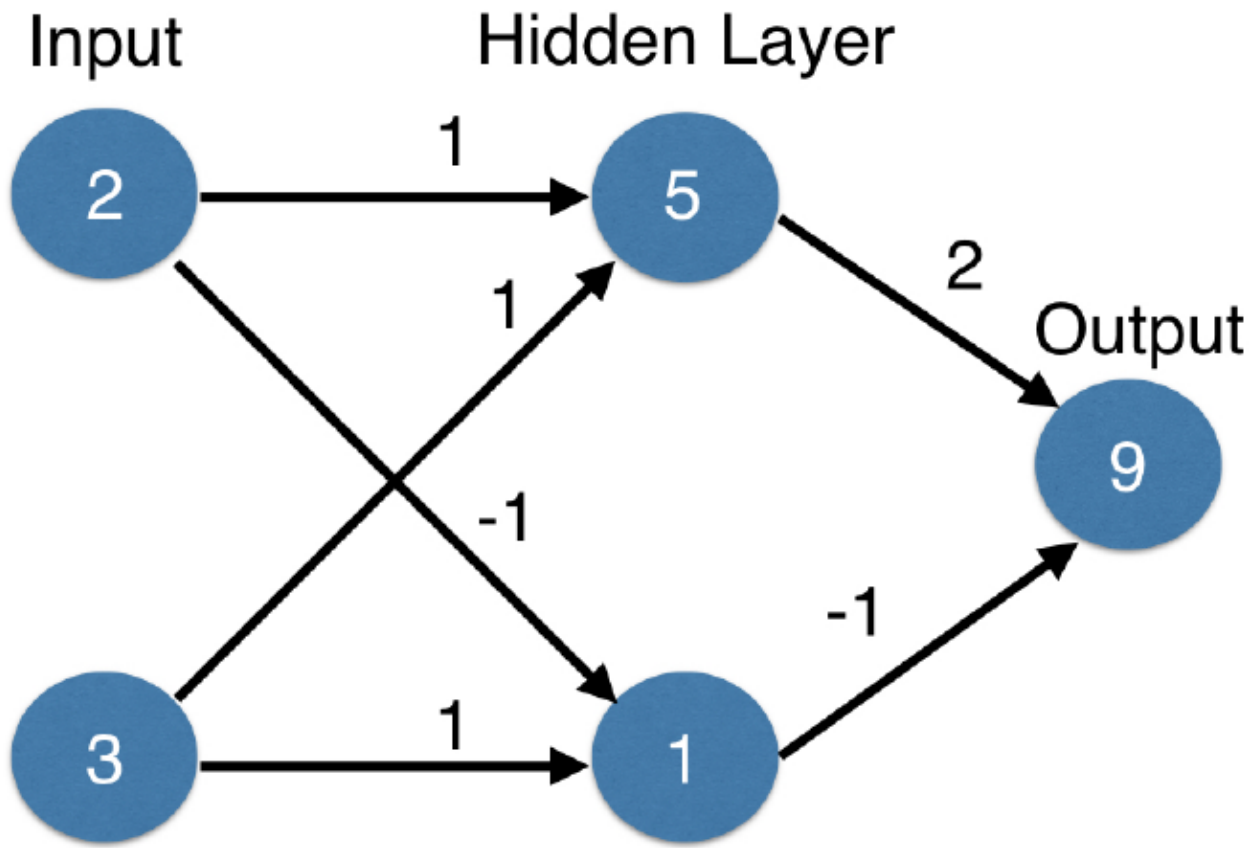


Image Processing Flowchart



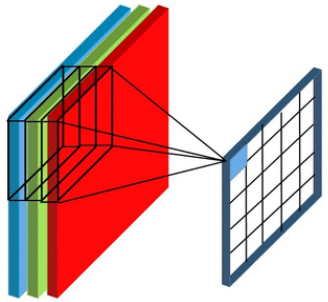
Layers of Model

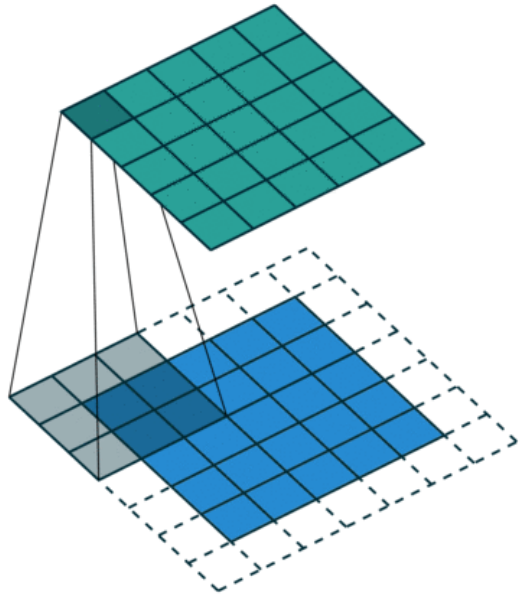
- Model consists of sequential layers of calculations – forming a neural network in order of input to output
  - Activation – determines output of neural network
  - Batch Normalization – helps prevent overfitting and normalize data
  - Flatten – before passing data to next layer we flatten to 1 dimension
  - Dense – takes outputs from previous layer and provides them for next layer



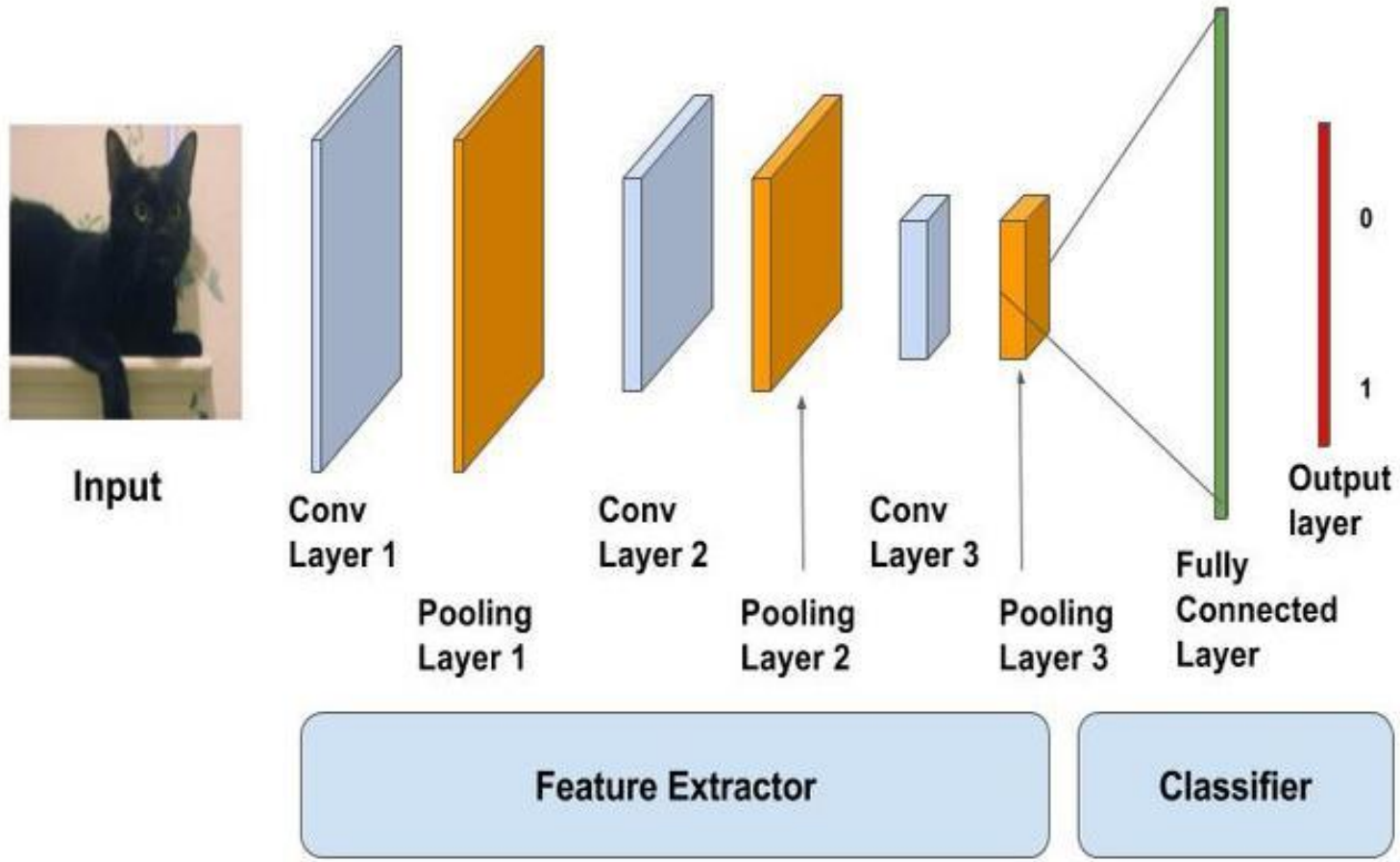
- Actual Value of Target: 13
- Error: Predicted - Actual = -4

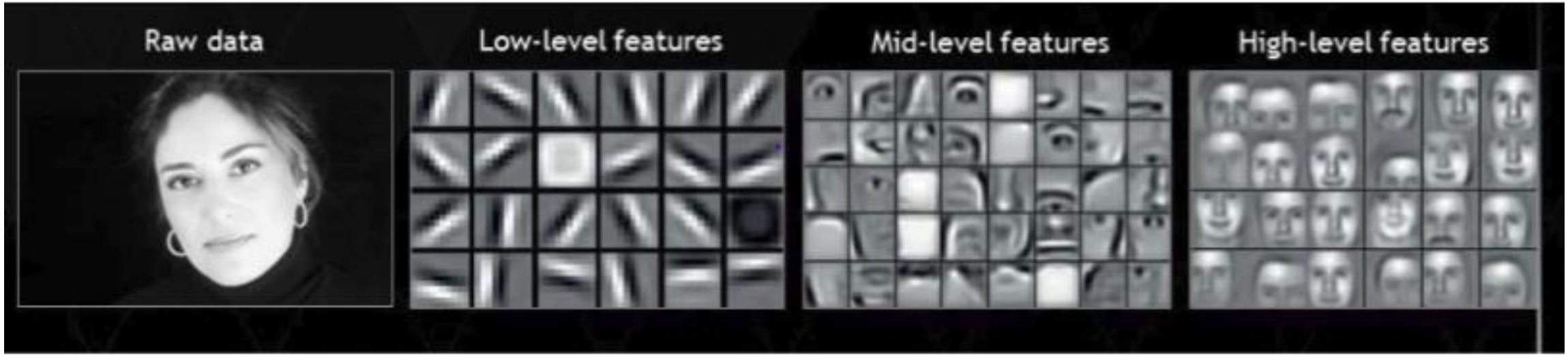
Convolutional Neural Network (CNN)





- CNNs have high accuracy for image processing and classification
- Used mainly used keras, sklearn, and cv2 packages to process images
  - Convert images to binary arrays for computer to interpret and analyze
  - Design model by adding layers to neural network
  - Optimize model


Input
Conv Layer 1
Pooling Layer 1
Conv Layer 2
Pooling Layer 2
Conv Layer 3
Pooling Layer 3
Fully Connected Layer
0
1
Output layer
Feature Extractor
Classifier


Raw data
Low-level features
Mid-level features
High-level features

Training Our Model

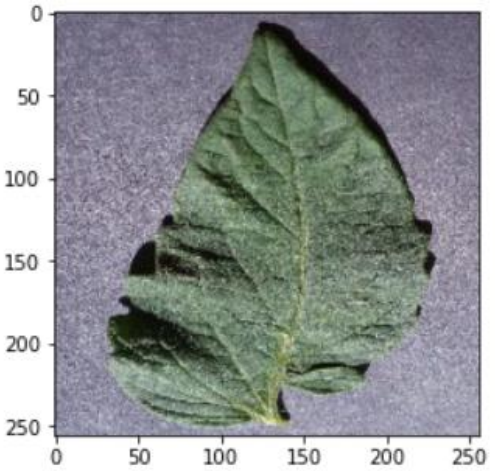
- Augment Data
- Split data into training and testing
- Fit model
- Evaluate model

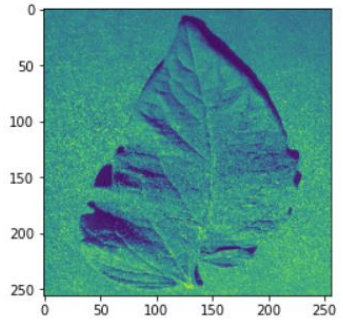
Model Optimization

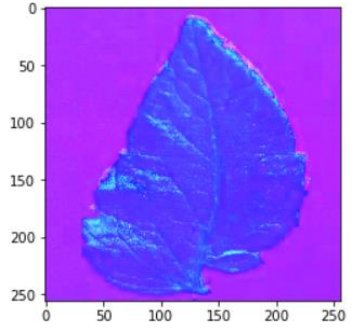
- Adam function is an optimization process that helps reduce prediction error
  - Adapts and molds the most accurate model by adjusting CNN weights
  - Has best accuracy in enhancing the CNN ability in classification
- Epoch is the number of times we pass the data through the model
  - Like solving same or similar problems again and again -- similar images easier to classify
  - It helps the model learn from a smaller datasets and with image transformations

Model Accuracy and Fit

- We are most concerned about the value accuracy of a model
  - Use statistical significance in app to identify plant leaf class
- Using image transformation and Adam optimization, we set epochs to 7 to achieve high model accuracy
- As a result, model is 84% accurate at classifying tomato, poison ivy, and raspberry leaf images

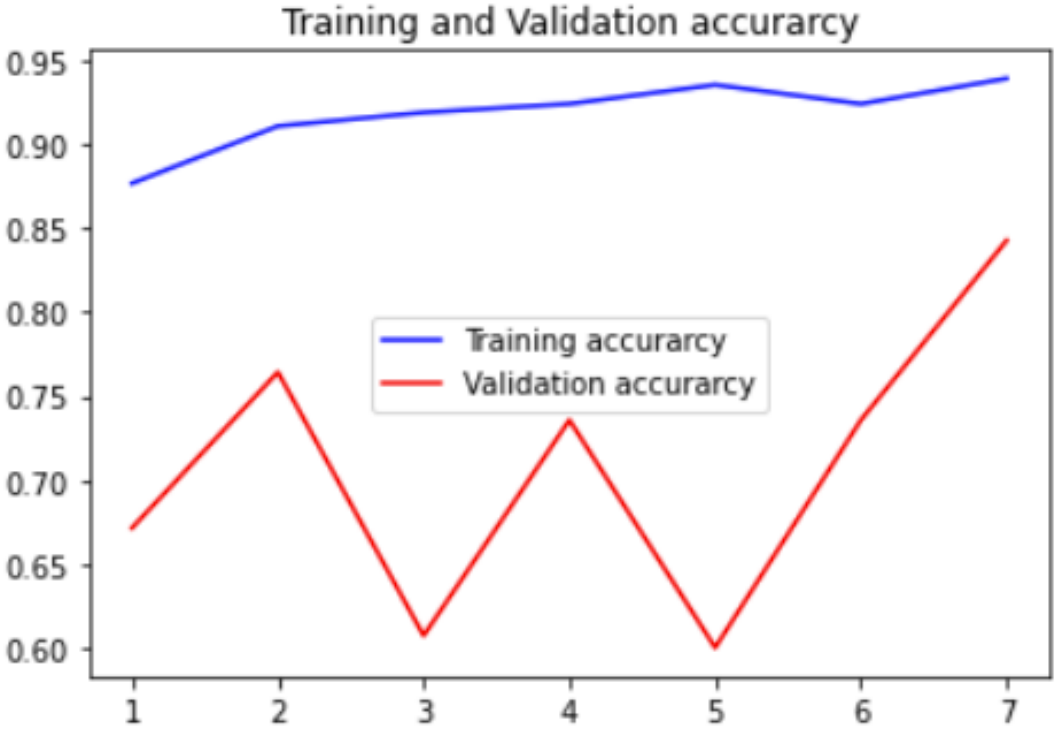






Model Accuracy and Loss

- Blue: Training data
- Red: Testing data



Validation

The training accuracy is higher than validation data

This is normal and expected

Model has seen the training data before and can easily identify images from training dataset

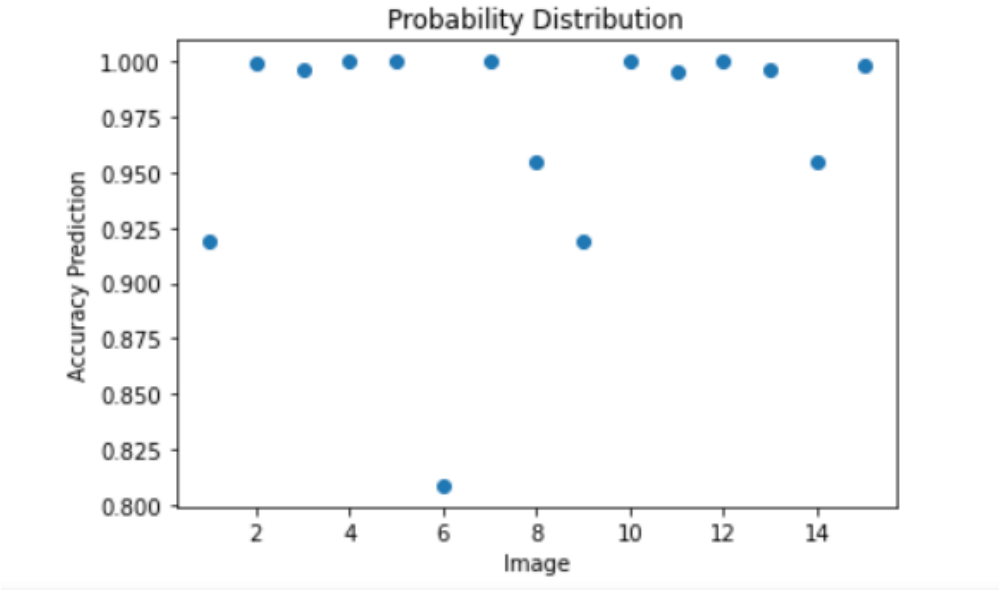
Training data loss is less than validation loss.

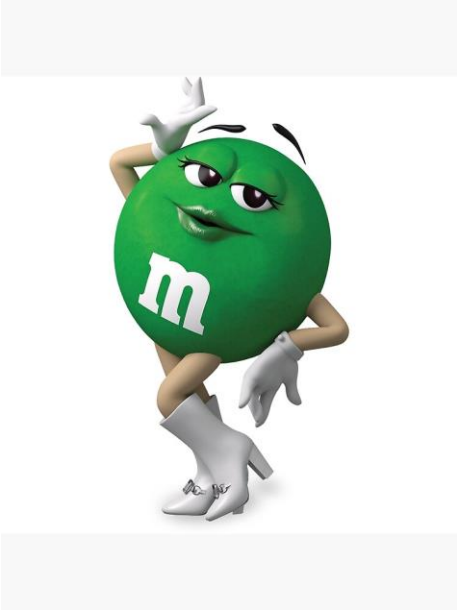
Normal and expected -- validation loss decreases with more epochs

Model has NOT seen the validation data before and is less familiar with those images

Prediction

- We have images that we did not run through our model
- We randomly selected an image from a set of 40 images
- The model accurately predicts the class of plant





Improvements

- Classify more plants
  - Currently have 3 classifications
  - Increase plant leaf classifications for premium service
- Add more images (more data) (more diverse environment)
- Augmenting the photos for training
- Merge Layers, Stack Models

Questions?