

# CS5102-Deep Learning

## KALEEM WAHEED 18L-1811 Project 2(B)

Preprocessing

- 1) Generate new images for Handle Class Balance Issue
- 2) Now each class have 1833 images
- 3) Remove noise/Irrelevant Images

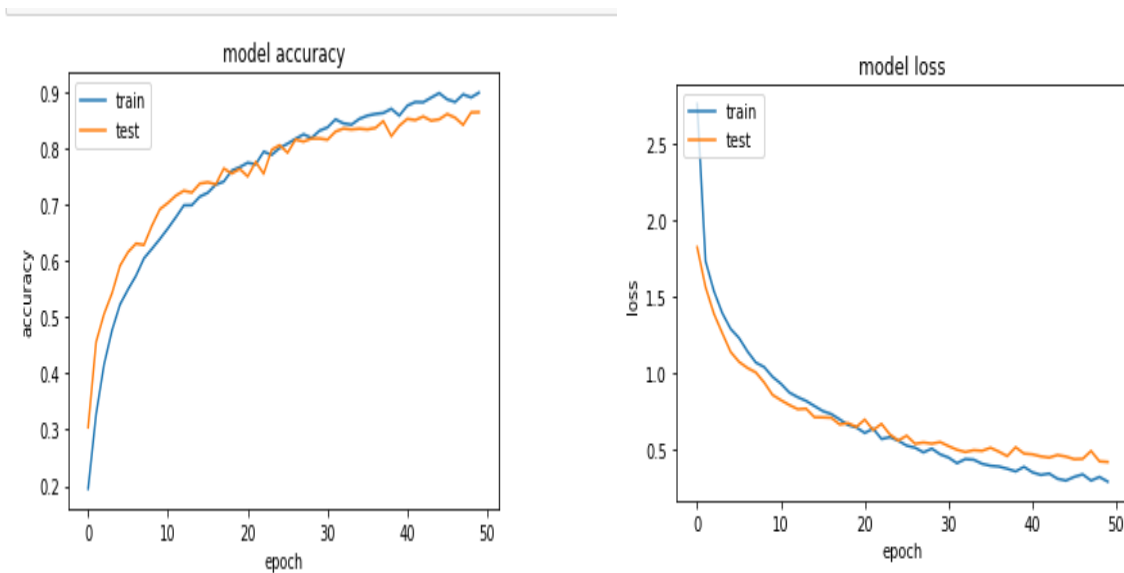
### 1. Classify the diseases using best deep learning architecture for classification

#### First Model

Building Convolutional Neural Network Model 1

Batch size 100, epoch 50, Adam optimizer Default Learning Rate

**Training Accuracy: 0.8980 validation accuracy: 0.8634**



## Second Model

I apply different model with changes in drop out for over fitting and increase layer and Epochs

Building Convolutional Neural Network Model Architecture 2

Use loss = 'categorical\_crossentropy'

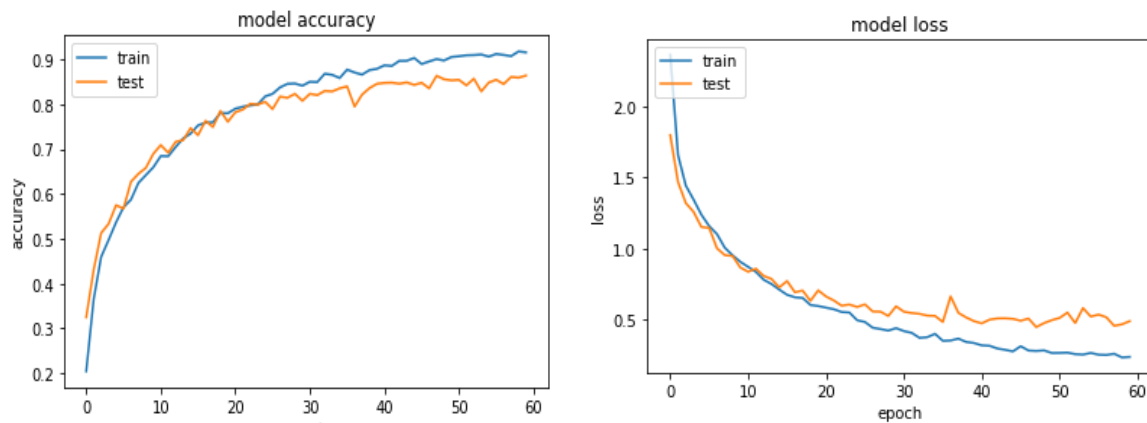
Batch size 100,

Epoch 60, Optimizer

Adam Default Learning Rate

Model 2 give better performance when increase Epochs

**Training Accuracy: 0.9159 validation accuracy: 0.8642**



## 2. Use transfer learning on ResNet, GoogleNet to retrain some part of the network

### ResNet Model 1

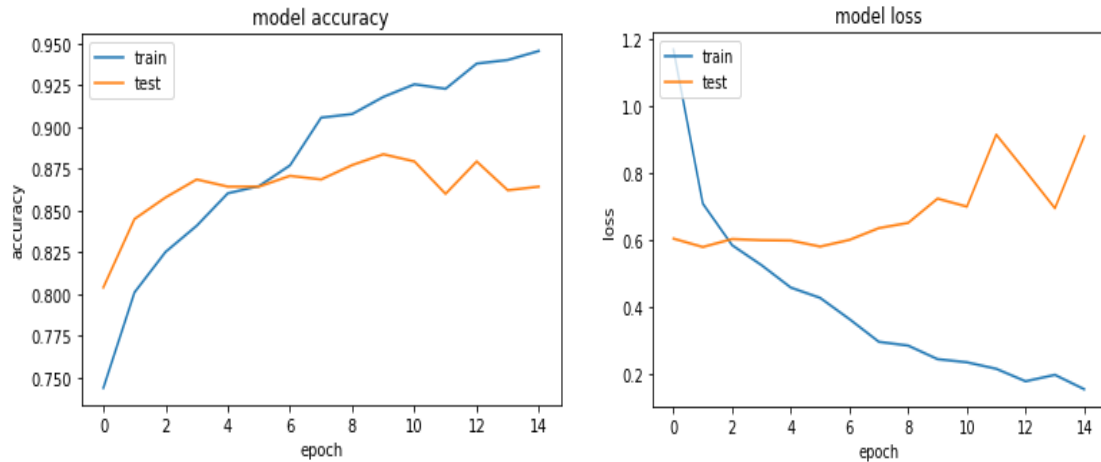
Keras ResNet50 Model 1 with Transfer learning

Input (shape=(64, 64, 3)), include\_top=False, weights='imagenet'

batch\_size=40, epochs=15 Dropout 0.6 and 0.4

Don't want to train all model only Last 5 layer retrain

**Training Accuracy: 0.9455 validation accuracy: 0.86**



## ResNet Model 2

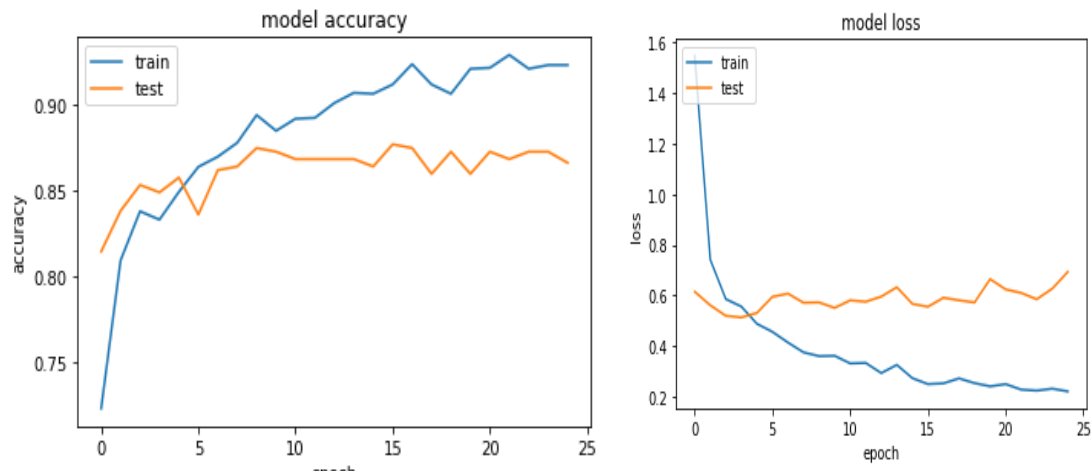
Input(shape=(64, 64, 3)),include\_top=False,weights='imagenet' batch\_size=38, epochs=25

Remove fully-connected 2nd last dense layer

loss='categorical\_crossentropy',optimizer='adam'

Don't want to train all model Last 3 layer retrain only due to limitation

**Training Accuracy: 0.92 validation accuracy: 0.86**



## GoogLeNet Model 1

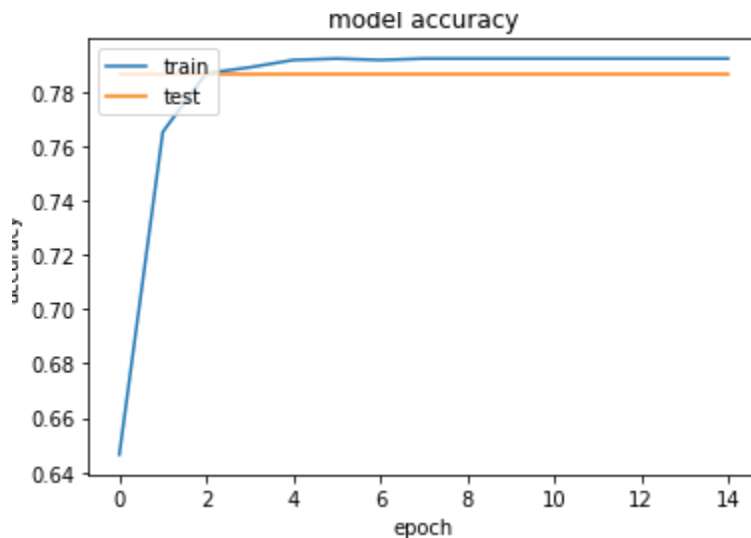
Input(shape=(150, 150, 3)), include\_top=False, weights='imagenet'

batch\_size=30, epochs=20 Remove fully-connected 2nd last dense layer

loss='categorical\_crossentropy', optimizer='adam'

Don't want to train all model Retrain only Last 3 layer

**Training Accuracy: 0.79 validation accuracy: 0.78**



## GoogleNet Model 2 (colab)

Input(shape=(150, 150, 3)), include\_top=False, weights='imagenet'

Decrease batch\_size=20, Decrease epochs=10 Add fully-connected 2nd last dense layer

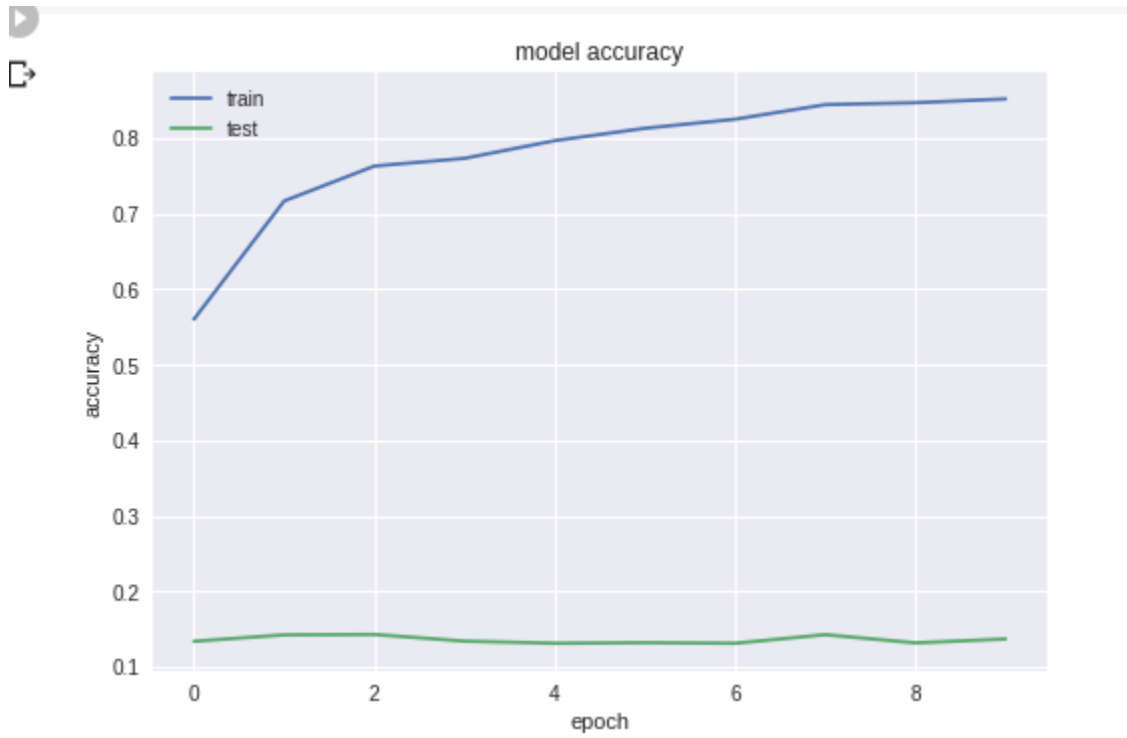
loss='categorical\_crossentropy', optimizer='adam'

Retrain Last 300 layer

I retrain last 300 layer but its validation accuracy is poor

It did not train

**Training Accuracy: 0.85 validation accuracy: 0.13**



**Generate adversarial network (GAN)**

**Using GAN Model generate 1 channel images.**

**It highlight skin disease.**

