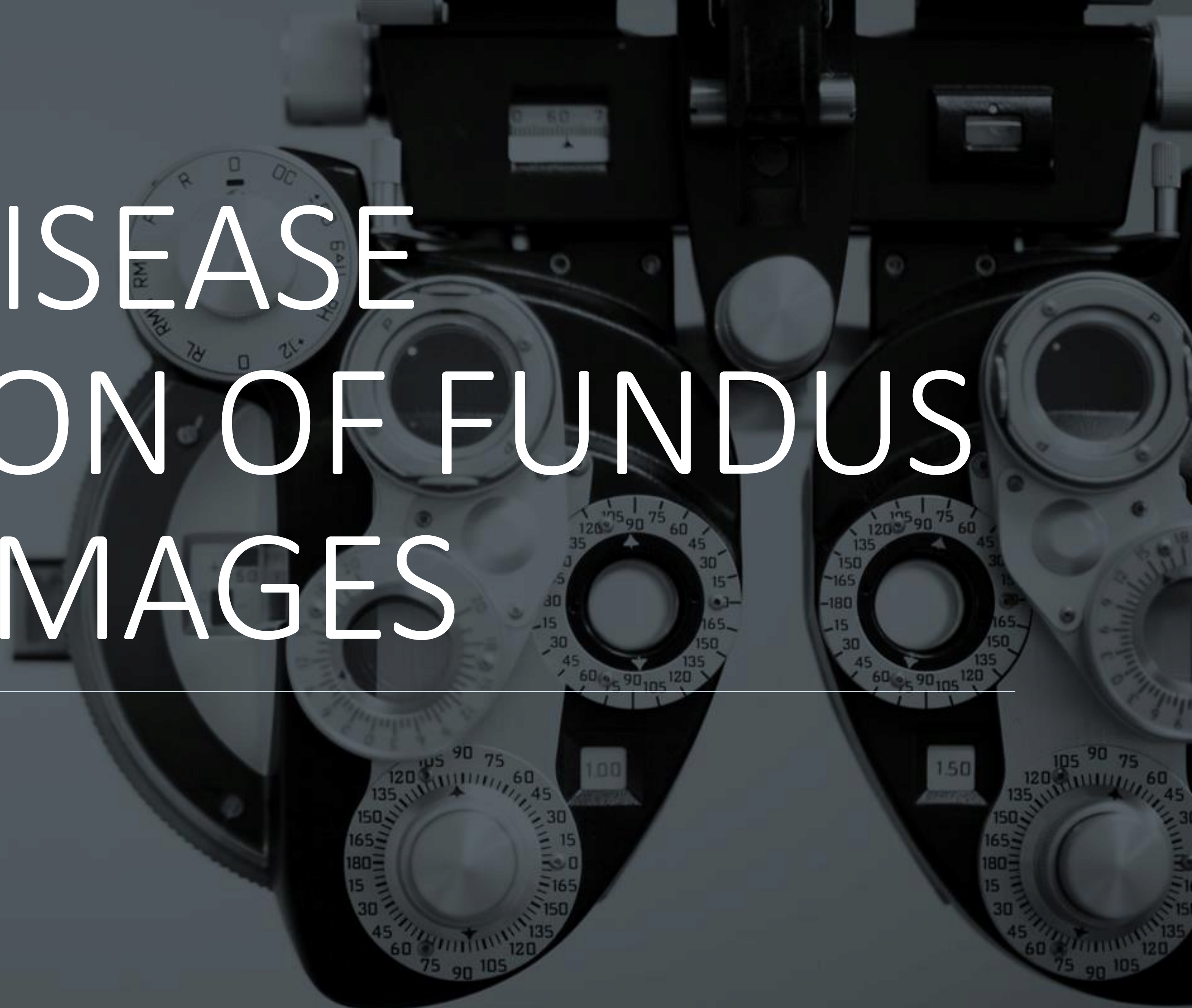


MULTI-DISEASE DETECTION OF FUNDUS RETINA IMAGES

JASJIV SINGH

UJJWAL MALIK

RISHABH SHARMA

Problem Statement

- The aim is to develop a generalized models for screening retina and detecting diseases to limit the adoption of computer-aided diagnostic tools.
- To train the neural network for the automated classification for 4 most common ocular diseases using retinal fundus images.
  - DR: Diabetic Retinopathy
  - MH: Media Haze
  - TSLN: Tessellation
  - ODC: Optic Disc Coloboma

Related Work

In-depth work on retina images is done using Convolution Neural Network in many papers:

- Multi-label classification of fundus images based on graph convolutional network (link)
- Multi-label Retinal Disease Classification Using Transformers (link)
- Hyper Parameter Tuned Deep Learning Based Lenet Architecture For Detection And Classification Of Diabetic Retinopathy Images (link)

Also, various CNN Models are used for this problem with the 20 multi-label ocular diseases are:

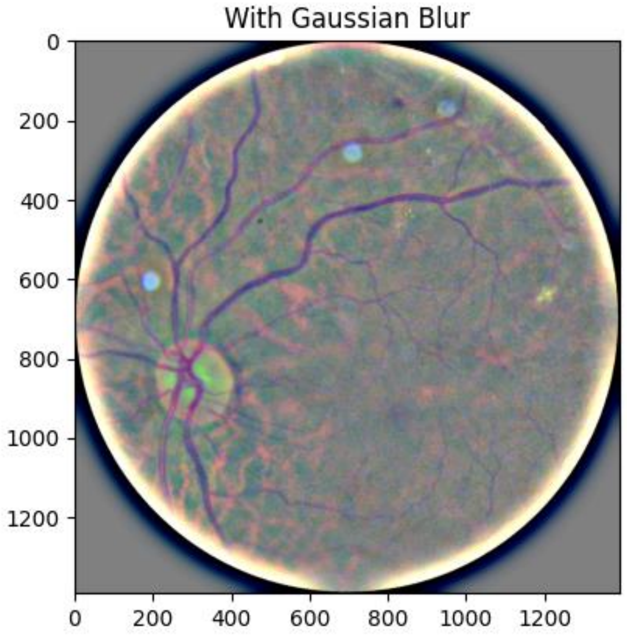
- ResNet50
- InceptionResNet
- Xception

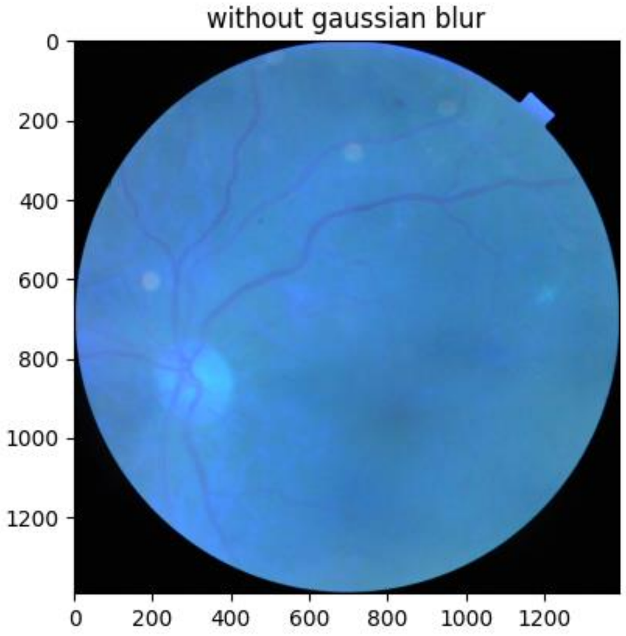
The main work done in the above papers is to augment the retina picture, reduce the noise and apply the model based on a particular retina disease.

Dataset Sourcing (1/2)

	DR	MH	TSLN	ODC
Number of Observations	508	419	354	251.0
Training	376	317	282	186
Evaluation	66	51	36	33
Testing	66	51	36	32

- The retina image dataset for 4 most common diseases is chosen for the model
- The dataset is cleaned by dropping the no diseases and other diseases classes.
- Filtered dataset is shown in the table.





Data Sourcing (2/2)

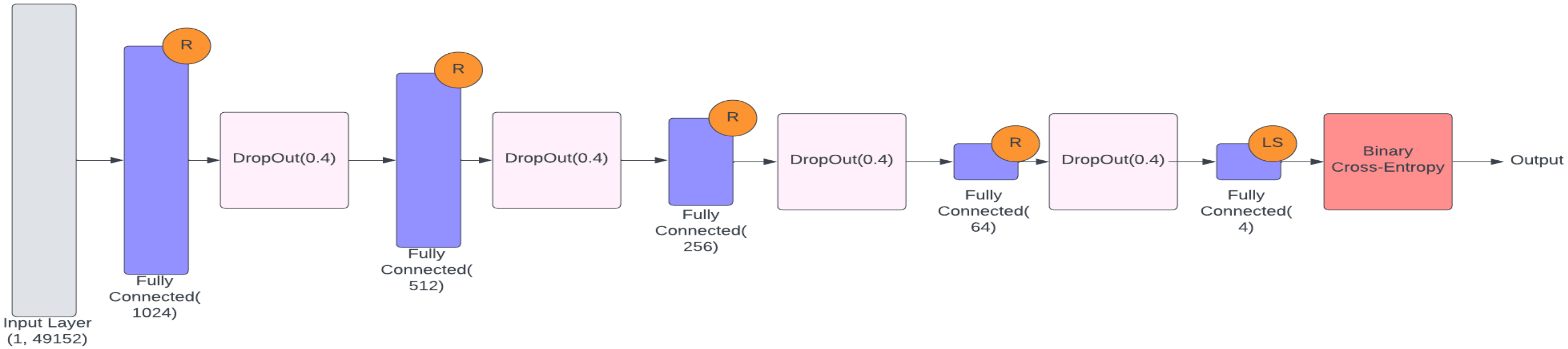
Each retina image is augmented in the following steps:-

- Each Image processing is done in the following steps using OpenCV:
  - Using Gaussian Blur to better expose the retina nerves.
  - Using circle crop to improve the model performance.
  - Resizing each image to 64 x 64 x 3
- All the observations of each dataset (Training, Validation, Testing) is stored in disk as numpy arrays.

Architecture (1/2) - MLP

Reasoning:

- The model consists of an input layer followed by 4 hidden n layers with ReLU activation function and an output layer used with the input RGB retina image size of 128 x 128 x 3
- As the dataset is multi-label, we are using sigmoid with log loss to calculate the probability of each disease and setting the argmax threshold of 0.5 for each label to compare the accuracies.
- The given figure shows the parameters used in building the architecture.



Architecture (1/2) – MLP (Continued)

Hyperparameters:

- Learning rate: 0.001
- Epochs: 30
- Batch Size: 64 with stochastic mini-batching
- Fully connected with Xavier initialization
- Adams optimization with weights
  - decay rates $\rho 1$ and $\rho 2$: 0.9 and 0.999 respectively with constant

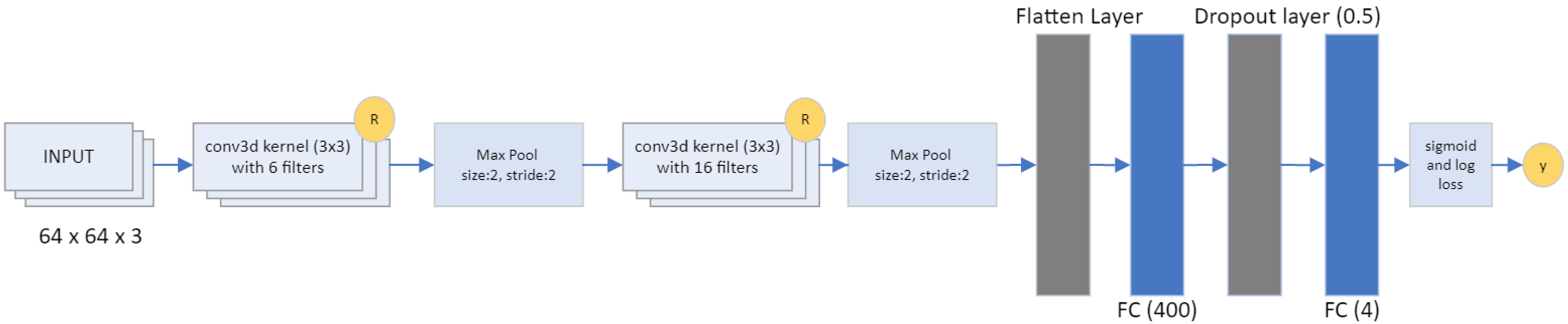
Reasoning

- Learning rate: Faster Learning rates were not able to converge properly on validation data
- Epochs: Overfitting was observed for complete data after 30 epochs
- Batch Size: maximum accuracy is observed with training, evaluation and testing dataset.
- Xavier and Adams optimization: helps in converging faster compared to standard weights. ADAM is chosen for better computation.

Architecture (2/2) - CNN

Reasoning:

- To keep the model simplified, a variant of LeNet architecture is used with the input RGB retina image size of 64 x 64 x 3
- As the dataset is multi-label, we are using sigmoid with log loss to calculate the probability of each disease and setting the argmax threshold of 0.5 for each label to compare the accuracies.
- The kernel size and filter are chosen based on the LeNet architecture and trail/error using small number of rows.



Architecture (2/2) – CNN (Continued)

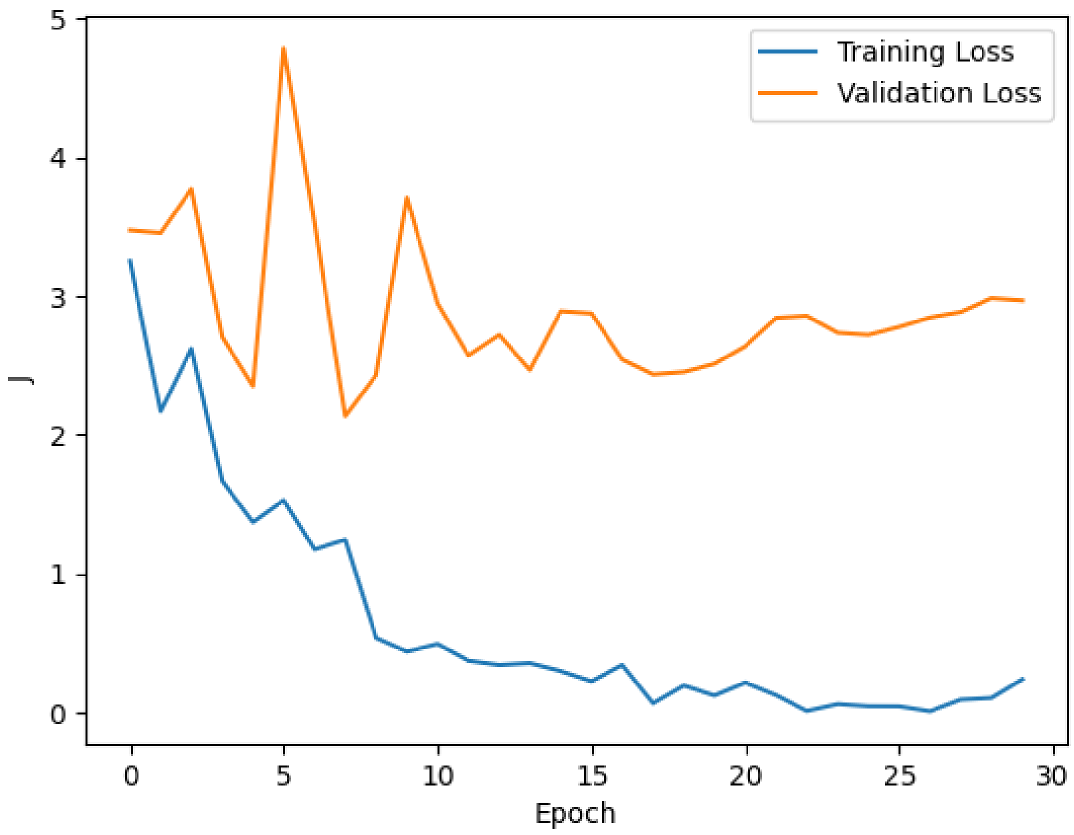
Hyperparameters:

- Learning rate: 0.0001
- Epochs: 5
- Batch Size: 4 with stochastic mini-batching
- Fully connected with Xavier initialization
- Adams optimization with weights
  - decay rates $\rho 1$ and $\rho 2$: 0.9 and 0.999 respectively with constant

Reasoning

- Learning rate: standard learning rate for image classification
- Epochs: Overfitting was observed after 5 epochs
- Batch Size: maximum accuracy is observed with training, evaluation and testing dataset.
- Xavier and Adams optimization: helps in converging faster compared to standard weights

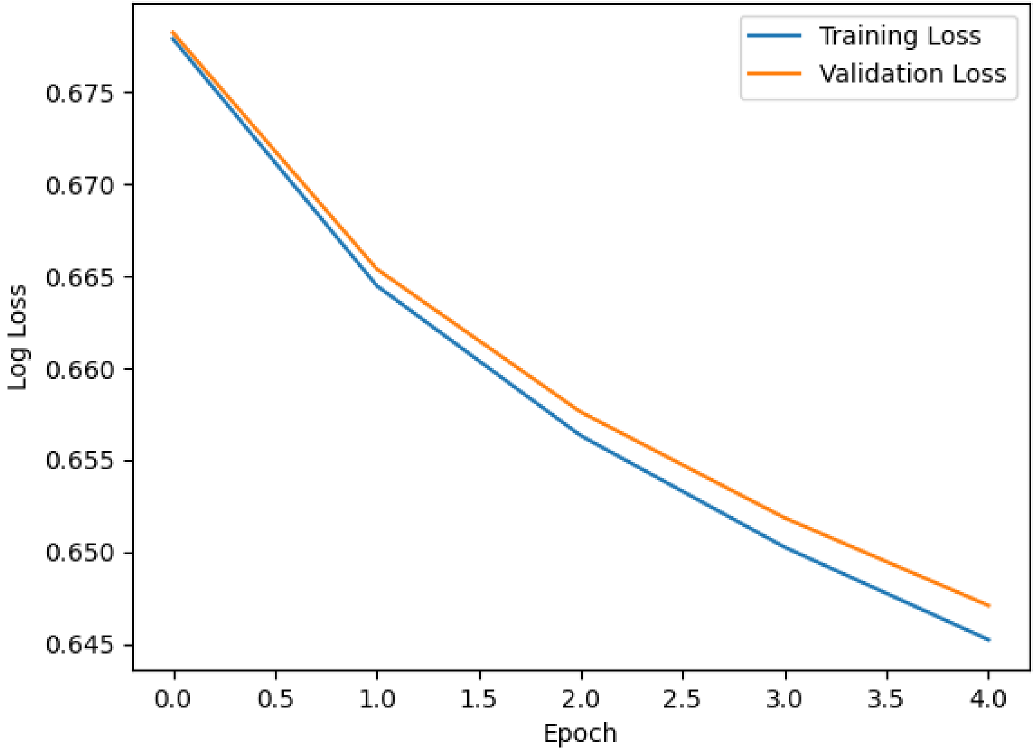
Observations (1/2) – MLP



The observation show that, the model is overfitting

Training Accuracy: 85%
Validation Accuracy: 65%

Observations (2/2) – CNN



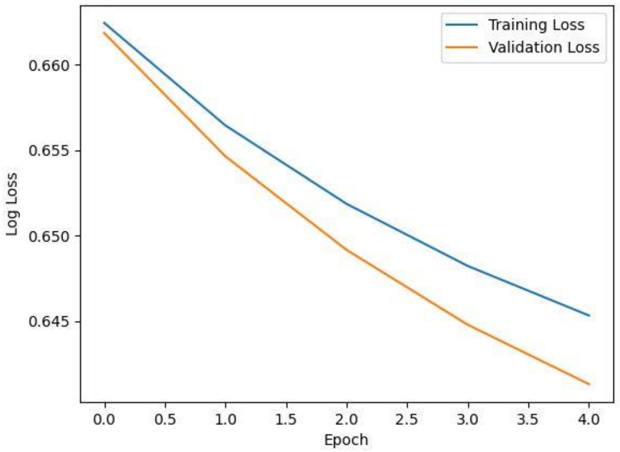
Batch: 8, Training: 83.9%, Validation: 68.5%

The observation shows that:

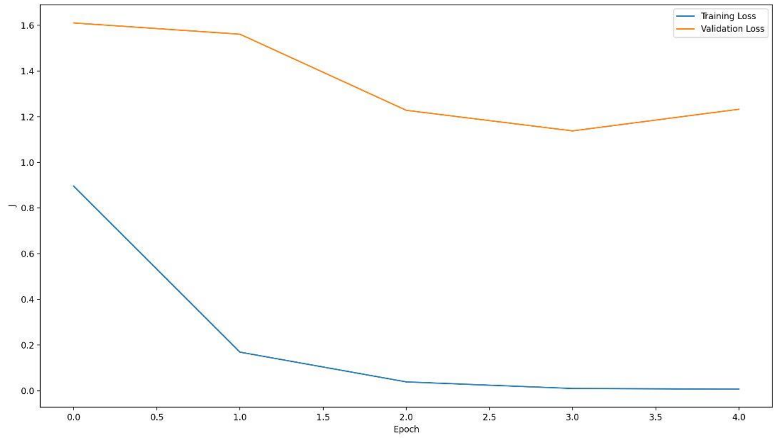
- Having a batch size of 8 helped in more generalized results.
- The convergence could happen if the model ran more than 5 epochs.
- Due to a long processing time (~3 hrs), we ran the model for 5 epochs.

Observations (2/2) – CNN with 100 observations

The observation shows the model trained with 100 images, also validated and tested with 100 images.



Batch size: 8, Training: 99, Validation: 93



Batch size: 4, Training: 99% Validation: 93

Future Work

Future extension could be:

- Expanding the model using miniVGG (VGG7) or VGG16 for a 512x512 image which should help us detect more details.
- Clear the noise and segmentation of the retina image dataset using the combination of CLAHE and Gaussian blur in openCV. This might help us reduce the size and information in images.
- Adding more layers and complexity will help us with more accurate predictions for all the 20 available classifications.
- Extend similar architecture for other applications in medical field.