Milestone 2: Deep Learning and Convolutional Neural Networks for Malaria Parasite Detection

By: Srikant Kumar Kalaputapu

# Introduction

In this experiment, we created and compared different neural network models to find the best option for predicting the presence of the Malaria parasite in images of red blood cells.

# Data Analysis

We used the tensorflow keras module to create a sequential model with various layers. We used the adam optimizer for our models instead of gradient descent as the adam optimizer updates learning rate dynamically. Gradient descent (GD) is an iterative first-order optimization algorithm used to find a local minimum/maximum of a given function. This method is commonly used to minimize the cost/loss function of the CNN. This leads to a better fit in the model.

## Model 0: Basic Convolutional Neural Network

In this model we did a simple CNN using ReLU activation functions, Dropout Layers and Max Pooling Hidden layer.

ReLU helps to prevent the exponential growth in the computation required to operate the neural network and prevents vanishing gradients. The Dropout layer is a mask that nullifies the contribution of some neurons towards the next layer and leaves unmodified all others – preventing overfitting. Max Pooling is a down sampling technique that reduces the size of the feature map.

Results:

We got an accuracy of 0.9842307567596436 after 15 epochs with an F1 score of 0.98, a recall of .99 and a precision of .98 for class 1 (parasitized). The confusion matrix shows we had 25 False positives, and 16 false negatives.

The accuracy plots show the validation accuracy is slightly higher than the train accuracy however the training accuracy follows the same trend as the validation accuracy (very similar at around the first epoch onwards)

The accuracy, F1, recall and precision is extremely good and the accuracy plots don’t show too much overfitting.

### Model 1: CNN with extra Relu layers

We added extra Relu layers.

Results:

We got an accuracy of 0.9819231033325195 after 3 epochs with an F1 score of .98, class 1 recall of .99 and class 1 precision of .97 **(slightly lower than model 0).** We had more False positives than model 0 (33), but less False negatives (14).

The accuracy plots show the validation accuracy is slightly higher than the train accuracy however the training accuracy follows the same trend as the validation accuracy (very similar at around the first epoch onwards)

Overall the accuracy has gone down slightly compared to model 0, however we had less False negatives – which is a benefit.

### Model 2: CNN with LeakyReLU and BatchNormalization

LeakyRelu layers prevent the dead neuron problem (vanishing gradient) by replacing zero values during backpropagation gradient descent. We use Batch normalization to improve optimization by regularization.

We got an accuracy of 0.9792307615280151 after 5 epochs with an F1 score of .98, class 1 recall of .97 (lower than the other models) and class 1 precision of .98. We had more False negatives (33) but less False positives (21).

The accuracy plots are all over the place with validation accuracy sometimes being lower – indicating possible overfitting.

### Model 3: CNN with LeakyReLU, BatchNormalization and Data Augmentation

Using the train\_generator and validation\_generator for the images with new orientations.

We got an accuracy of 0.9561538696289062 after 5 epochs with an F1 score of 0.95, class 1 recall of .92 and a class 1 precision of .99. The False Negatives increased tremendously to 105 while the False positives decreased further than the other models to 9.

The accuracy plot is much better than model 2.

Overall this model had less accuracy, F1, and recall than the other models, but the accuracy plot showed a closer training accuracy to validation accuracy than the other plots.

### Model 4: Pre-trained model (VGG16)

We got an accuracy of 0.9415384531021118 (the worst so far) after 7 epochs with an F1 score of .94, class 1 recall of .96 and a class 1 precision of .93. We had more false negatives than the base model, but less than the data augmentation model 3. This model had the greatest number of false positives at 94.

The accuracy graphs show less overfitting in this model compared to model 3, but worse than model 0 and 1.

## Conclusions

The model with the best accuracy and the least number of False negatives was Model 1: Basic CNN with extra ReLU layers. I think this model is a good final solution because its accuracy plots showed a similar trend between the training accuracy and the validation accuracy. The other model’s had higher False negatives which I believe we need to reduce considering missing a Malaria diagnosis could lead to death.

## Bibliography

<https://towardsdatascience.com/everything-you-need-to-know-about-gradient-descent-b1d9edfce0b5#:~:text=Gradient%20Descent%20is%20an%20algorithm,minimize%20the%20model's%20cost%20function>

<https://medium.com/@cdabakoglu/what-is-convolutional-neural-network-cnn-with-keras-cab447ad204c>

[https://www.baeldung.com/cs/ml-ReLU-dropout-layers](https://www.baeldung.com/cs/ml-relu-dropout-layers)

<https://ai.stackexchange.com/questions/9563/is-pooling-a-kind-of-dropout>