CNNs for image classification

Vadde Venkata Kamala Sindhoori
University of Adelaide
Adelaide, South Australia, 5000
a1905610@adelaide.edu.au

Abstract

This paper inspects various Convolutional Neural Network (CNN) algorithms for the image classifcation of CIFAR-10 dataset. The study begins with examining each of the four CNN algorithms : VGG-16, REsNet-18, ResNet-34, VGG-19 by building baseline models and performing hyper-parametric optimisation on these models. The findings of this paper compare different models in their performance for the image classification.

1. Introduction

CNNs are one of the most powerful concepts in Deep Learning which created a revolution in the areas of Image Classification, Object Recognition. The CIFAR-10 dataset is a widely used dataset which is used to study the performance of various models due to its diversity and well-balanced nature. Various CNN models like VGG-16,VGG-19, ResNet-18, ResNet-34 have demonstrated that they are exceptional in handling the CIFAR 10 dataset.

According to Simonyan & Zisserman (2014) [3], VGG models prioritised deeper architectures with smaller convolutional filters. Similarly as discussed by He et al.(2015) [2], ReNets use residual connections in mitigating the issue of vanishing gradients.

This paper involves individually examining different CNN models by building baseline models and performing hyper-parametric optimisation on them. The models are then compared on the basis of their performance.

In Section 2, previous works related to using various CNN models on CIFAR-10 dataset were discussed. Furthermore, in the section 3, the data used, its fetaures and the models used on the data were discussed. Finally, the section 4 involves the discussion the performance of various models on the data and which models perform best in the image classification of CIFAR-10 dataset.

2. Related Works

Several studies earlier have evaluated VGG and ResNet architectures. Simonyan & Zisserman (2014) [3] presented various VGG models which include VGG-16 and VGG-19 and emphasized that the VGG models improve feature learning by using deeper networks with small convolutional filters. From a study done by Chatfield (2014) [1], it can be inferred that the VGG models are well know for their robustness and good accuracy for small scale image datasets.

Furthermore He et al. (2015) [2] emphasized on the ability of ResNet models in solving the problem of vanishing gradient problem. Zagoruyko and Komodakis (2016) [5], increased the width of residual blocks in the ResNet models and achieved high and very good accuracy. Several studies demostrated the superior performance of VGG and ResNet architectures.

3. Methodolody

The complete code script for this assignment is available on Github ,for review purposes, please refer to the link [4]

3.1. Data

The assignment uses CIFAR-10 dataset which consists of 60,000 colour images in which 50,000 are used for training and the remaining 10,000 are used for testing purposes. Each image is of size 32x32 pixels.

3.2. Exploratory Data Analysis

3.2.1 Visualizing images

The images for each class in CIFAR-10 has been visualised. The samples are collected from each class, unnormalized and plotted in a grid. Some of the images are given in 1

3.2.2 Exploring Class distribution

The class distribution of the dataset is explored using bar chart which showed uniform distribution of classes over the dataset. The bar chart is given in 2

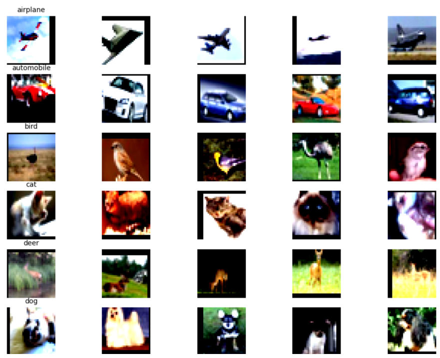


Figure 1. Visuals of Classes in CIFAR-10 dataset

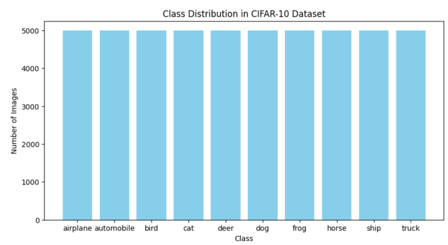


Figure 2. Class Distribution of CIFAR-10 dataset

3.3. VGG-16

VGG-16 is a deep neural netwok which is pre-trained in ImageNet and it consists of 16 layers of convolutional and fully connected layers.

3.3.1 Baseline model

The model is a pre-trained VGG-16 model. Techniques like Data Augmentation and color jittering are applied to the data for better generalization. In the code, dropout layers are included in order to avoid over-fitting. Early-stopping is also introduced in the code to check upon validation accuracy. The code uses learning rate scheduler which adjusts the learning rate while training.

3.3.2 Hyper-parametric optimisation model

In Hyperparametric optimisation, the model consists of customised classifier to prevent over-fitting. Optuna is used in the model. The parameters include ”dropout” which was set to different values which are in range 0.3 to 0.6. Another parameter ”lr” was in range 1e-5 to 1e-3. Finally, the last parameter which is weight decay was set to various values between 1e-6 and 1e-3. The optimal parameters are then picked and the test set is trained and analysed iin the basis of these parameters.

3.4. ResNet-18

ResNet-18 is a Convoloutional Neural Network which reduces vanishing gradient problem by using residual connections. It consists of 18 layers which include convolutional layers batch normalisation and shortcut layers.

3.4.1 Baseline model

The baseline model for ResNet-18 is trained for 15 epochs and the final-fully connected layer is modified for 10 classes. It is trained using Adam optimizer. Similar to VGG-16, early stopping is introduced in-order to prevent over-fitting.

3.4.2 Hyper-parametric optimisation model

The model uses optuna for optimising the parameters. The model consists of a configurable dropout layer before final fully connected layer. The code consists of three parameters which were assigned different values.They are dropout rate which range from 0.3 to 0.7, learning rate which is between 1e-5 and 1e-3, and finally weight decay whose range is 1e-6–1e-3.

3.5. ResNet-34

ResNet is a neural network which consists of 34 layers. To overcome the vanishing gradient problem it uses skipping connections.

3.5.1 Baseline model

The baseline model uses pre-trained ResNet-34 model. To reduce over-fitting, the final fully connected layer consists of an addition which is a dropout layer with rate 0.5. The baseline model uses Adam scheduler.

3.5.2 Hyper-parametric optimisation model

The hyper-parametric optimisation model for Res-Net 34 uses optuna and is pre-trained. The parameters which are to be tuned include dropout layer ranging between 0.3 and 0.7, learning rate which is from 1e-5 to 1e-3 and weight decay which lies between 1e-6 and 1e-3. The model also uses Adam optimiser along with a step based learning scheduler with a step size of 5 and with a gamma value of 0.5

3.6. VGG-19

VGG-19 is a convolutional nneural network which consists of 19 layers.

3.6.1 Baseline model

The code for baseline model of VGG-19 uses pre-trained models. The last two fully connected layers of the model includes increased dropout of 0.6 and batch normalisation to stop over-fitting. A learning rate scheduler is used to reduce learning rate by half with every increase in epoch.

3.6.2 Hyper-parametric optimisation model

Similar to other hyper-parametric optimisation models discussed above, the model for VGG-19 uses optuna for optimising paramters. The model is trained for 10 epochs during tuning. The model also consists of Adam Optimiser and LR scheduler. Early stopping is added inorder to prevent over-fitting. The parameters which are used for tuning include Dropout Rate varying between 0.3 and 0.7,learning rate which lies between 1e-5 and 1e-3 and weight decay which is from 1e-6 to 1e-3. After tuning, the best parameters are extracted and then used for retraining the model for 15 epochs.

4. Results and Evaluation

4.1. VGG-16

4.1.1 Baseline model

The learning curve for VGG-16 baseline model 3 indicates the progression of training and validation curves over 15 epochs. Both the training and validation accuracies increase rapidly at the beginning. At around epoch 5 both head toward covergence. As training accuracy increases the validation accuracy plateaus indicating that there is no overfitting.

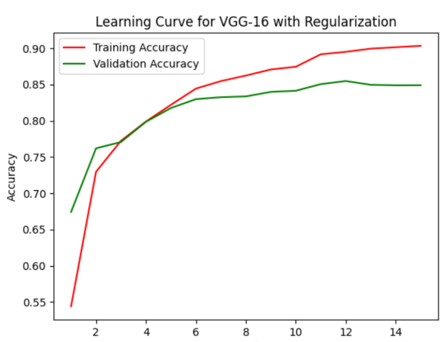


Figure 3. Learning curve for VGG-16 Baseline model

From the classification report 4, we can deduce that the overall accuracy for the VGG-16 baseline model is 85% . The average f1 score among all the classes also stands at 85%. The preicsion, recall and f1-scores of class 3 are less compared to other classes indicating that class 3 is difficult to classify.

Finally the confusion matrix for VGG-16 baseline model is 5

Classification Report:

	precision	recall	f1-score	support
0	0.84	0.87	0.86	1000
1	0.95	0.90	0.93	1000
2	0.83	0.83	0.83	1000
3	0.77	0.68	0.72	1000
4	0.82	0.84	0.83	1000
5	0.72	0.82	0.77	1000
6	0.91	0.88	0.89	1000
7	0.89	0.85	0.87	1000
8	0.93	0.92	0.93	1000
9	0.88	0.93	0.90	1000
accuracy			0.85	10000
macro avg	0.85	0.85	0.85	10000
weighted avg	0.85	0.85	0.85	10000

Figure 4. Classification report for VGG-16 Baseline model

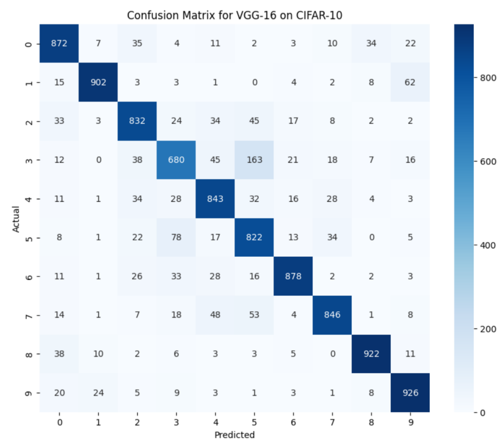


Figure 5. Confusion matrix for VGG-16 Baseline model

4.1.2 Hyperparametric optimisation model

The best hyperparametrs were found through optuna and are as follows: 'dropout': 0.33337000707810527, 'lr': 0.00022920450758813255, 'weight_decay': 7.145790730973122e-05.

From the learning curve of VGG-16 hyperparametric optimisation model 6, we can observe that the training accuracy starts slow in the beginning where as the validation accuracy curve increases rapidly at the beginning. There is a slight gap between training and validation accuracies approximately after epoch 5 indicating a chance of overfitting. However, the alignment of the curve in the further epochs indicates that the model could generalise well.

The overall accuracy after Hyper-parametric optimisation of VGG-16 model is 86% . Even though it is just 1% more than the baseline accuracy, because of computational resources restrcition, it can be considered good overall. Also from the classfication report 7, it can be deduced that the overall F1-score for the model is 86% and the precision and recall are 87% and 86% respectively.

The confusion matrix for VGG-16 hyper-parametric optimisation model is 8

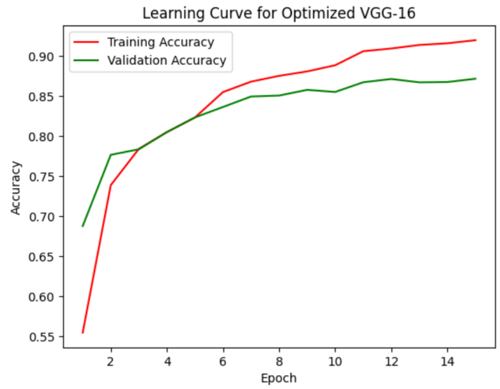


Figure 6. Learning curve for VGG-16 Hyperparametric optimisation model

Classification Report:

	precision	recall	f1-score	support
0	0.90	0.87	0.88	1000
1	0.94	0.92	0.93	1000
2	0.84	0.82	0.83	1000
3	0.71	0.76	0.74	1000
4	0.85	0.85	0.85	1000
5	0.75	0.83	0.79	1000
6	0.94	0.86	0.90	1000
7	0.90	0.87	0.89	1000
8	0.92	0.93	0.92	1000
9	0.89	0.92	0.90	1000
accuracy			0.86	10000
macro avg	0.87	0.86	0.86	10000
weighted avg	0.87	0.86	0.86	10000

Figure 7. Classification report for VGG-16 Hyperparametric optimisation model

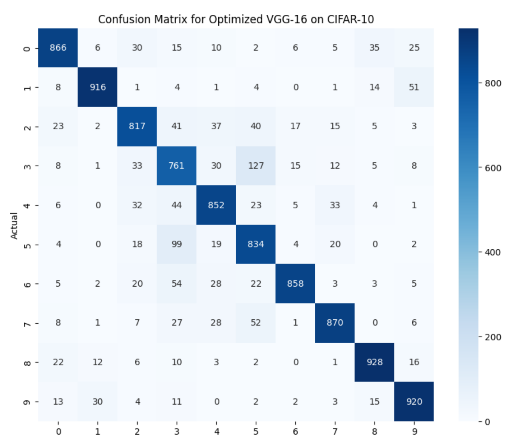


Figure 8. Confusion matrix for VGG-16 Hyper-paarmetric optimisation model

4.2. ResNet-34

4.2.1 Baseline model

The learning curve 9 for ResNet-34 baseline model indicates that the training accuracy starts low at the beginning where as the validation accuracy starts higher. The model has minimal over-fitting which is indicated by the gap between training and validation accuracies.

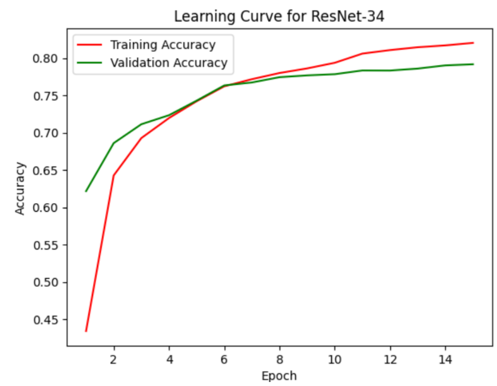


Figure 9. Learning curve for ResNet-34 Baseline model

The average accuracy for the baseline model is 79% which can be inferred from the classification report 10. Also, classes 1 and 9 have higher recall (0.88 and 0.84 respectively) and F1-scores (0.87 and 0.84 respectively) indicating that they have higher performance and better predictions compared to class 3 (precision: 0.63, F1-score:0.63) where the performance is not very good.

Classification Report:

	precision	recall	f1-score	support
0	0.82	0.83	0.82	1000
1	0.87	0.88	0.87	1000
2	0.78	0.71	0.74	1000
3	0.63	0.63	0.63	1000
4	0.74	0.78	0.76	1000
5	0.72	0.68	0.70	1000
6	0.81	0.86	0.84	1000
7	0.84	0.82	0.83	1000
8	0.84	0.87	0.86	1000
9	0.85	0.84	0.84	1000
accuracy			0.79	10000
macro avg	0.79	0.79	0.79	10000
weighted avg	0.79	0.79	0.79	10000

Figure 10. Classification report for ResNet-34 Baseline model

Finally the confusion matrix for ResNet-34 baseline model is given in figure 11.

4.2.2 Hyperparametric optimisation model

The best parameters for the ResNet-34 after performing hyper-parametric optimisation are:

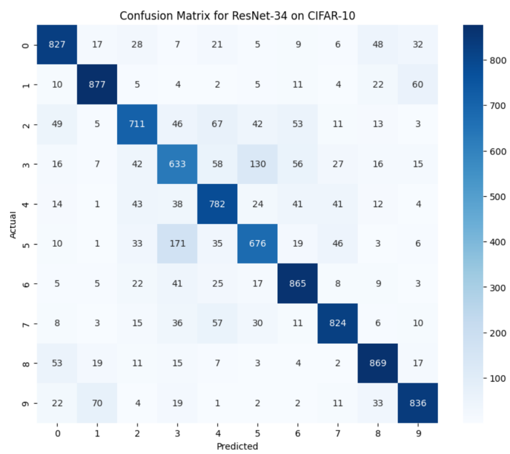


Figure 11. Confusion matrix for ResNet-34 Baseline model

'dropout_rate': 0.35357551522333724, 'learning_rate': 0.00030835054909216295, 'weight_decay': 0.0002513262903006411.

From the learning curve 12, the validation accuracy starts at a much higher rate compared to the training acuuracy due to regularisation techniques. But soon, the training accuracy show a rapid increase and the validation accuracy plateaus around 10 epochs indicating stabilization which means that the model generalizes well.

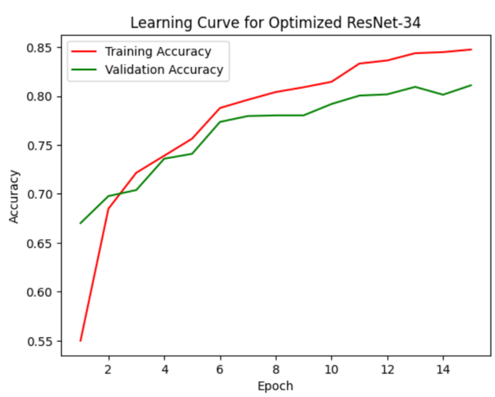
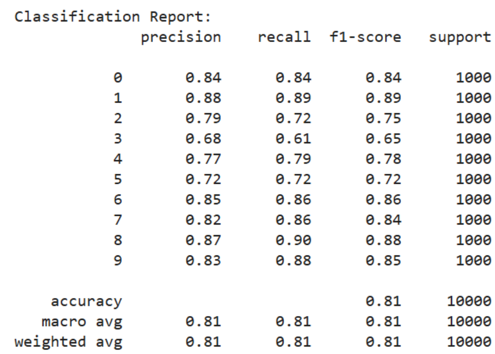


Figure 12. Learning curve for ResNet-34 Hyperparametric optimisation model

The classification report 13 for the hyper-parametric optimisation model indicates that the overall average accuracy is 81%. The average precision,recall and F1-scores also stand at 81% because of the well-balanced dataset. However, in particular, classes 1 and 8 show higher performance with high recall,f1-scores (0.89 and 0.90 repectively, 0.89 and 0.88 respectively) wherease classes 3 and 5 show not so good performance with their low recall and f1-scores (0.68 and 0.72 respectively, 0.65 and 0.72 respectively).

The confusion matrix for ResNet-34 hyper-parametric optimisation model is given in the figure 14



Classification Report:

	precision	recall	f1-score	support
0	0.84	0.84	0.84	1000
1	0.88	0.89	0.89	1000
2	0.79	0.72	0.75	1000
3	0.68	0.61	0.65	1000
4	0.77	0.79	0.78	1000
5	0.72	0.72	0.72	1000
6	0.85	0.86	0.86	1000
7	0.82	0.86	0.84	1000
8	0.87	0.90	0.88	1000
9	0.83	0.88	0.85	1000
accuracy			0.81	10000
macro avg	0.81	0.81	0.81	10000
weighted avg	0.81	0.81	0.81	10000

Figure 13. Classification report for ResNet-34 Hyperparametric optimisation model

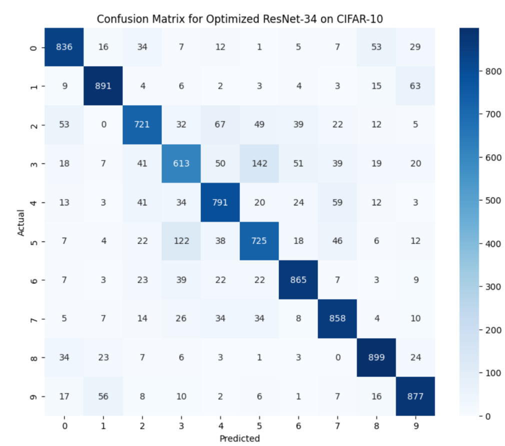


Figure 14. Confusion matrix for ResNet-34 Hyper-paarmetric optimisation model

4.3. ResNet-18

4.3.1 Baseline model

From the learning curve of ResNet-18 15 baseline model, we can see that the training curve is lower than the validation curve at the beginning but as the number of epochs increases and with more training both training accuracy and validation accuracy increase. The close movement of both the training and validation accuracy implies that there is no overfitting.

From the information provided by the classification report 16 of baseline model, we can infer that the average accuracy for the baseline model is 76%. Even the average precision, recall value and F1-score stand at 76%. The highest performance is observed with class 1 with high scores of recall and f1-score (0.85 and 0.85 respectively) which means it has the most accurate predictions.

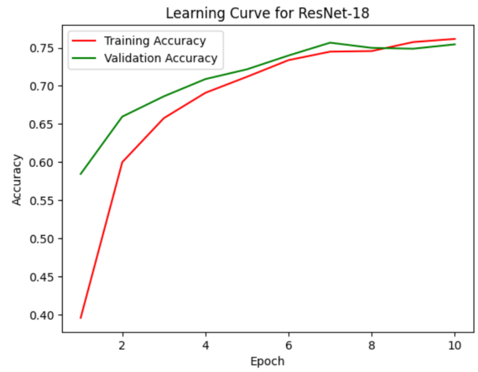


Figure 15. Learning curve for ResNet-18 Baseline model

Classification Report:

	precision	recall	f1-score	support
0	0.76	0.81	0.78	1000
1	0.85	0.85	0.85	1000
2	0.69	0.67	0.68	1000
3	0.61	0.57	0.59	1000
4	0.72	0.72	0.72	1000
5	0.69	0.68	0.68	1000
6	0.80	0.81	0.80	1000
7	0.79	0.81	0.80	1000
8	0.83	0.86	0.84	1000
9	0.83	0.80	0.81	1000
accuracy			0.76	10000
macro avg	0.76	0.76	0.76	10000
weighted avg	0.76	0.76	0.76	10000

Figure 16. Classification report for ResNet-18 Baseline model

Finally the confusion matrix for ResNet-18 baseline model is given in 17.

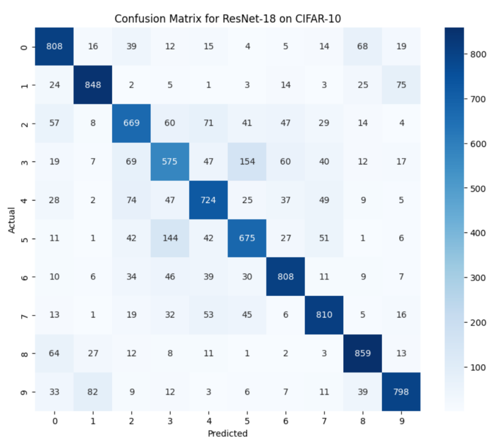


Figure 17. Confusion matrix for ResNet-18 Baseline model

4.3.2 Hyperparametric optimisation model

The best hyperparameters found for ResNet-18 are 'dropout_rate': 0.6166536528308428, 'learning_rate': 0.0004677538892259788, 'weight_decay': 1.8306657179446032e-06.

The learning curve for ResNet-18 Hyper-parametric optimisation model 18 indicates that the training accuracy increases steadily at the beginning. Validation accuracy also increases but quickly plateaus near epoch 6. The validation accuracy stabilises as the training accuracy increases. The divergence between both the curves is very less indicating that there is no over-fitting.

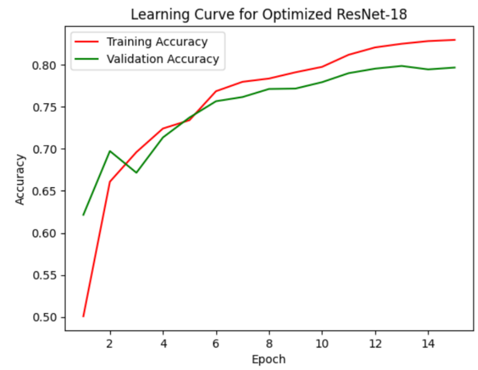


Figure 18. Learning curve for ResNet-18 Hyperparametric optimisation model

From the classification report of ResNet-18 Hyperparametric model 19, we can infer that the precision values range from 0.62 to 0.89 implying variable reliabilty. The F1-scores range from 0.65 to 0.87. The report also indicates overall well balanced performance of the classes. The overall accuracy for this model is 80%.

Classification Report:

	precision	recall	f1-score	support
0	0.79	0.85	0.82	1000
1	0.87	0.89	0.88	1000
2	0.76	0.74	0.75	1000
3	0.62	0.67	0.65	1000
4	0.78	0.78	0.78	1000
5	0.77	0.68	0.72	1000
6	0.84	0.84	0.84	1000
7	0.83	0.85	0.84	1000
8	0.89	0.84	0.87	1000
9	0.83	0.84	0.83	1000
accuracy			0.80	10000
macro avg	0.80	0.80	0.80	10000
weighted avg	0.80	0.80	0.80	10000

Figure 19. Classification report for ResNet-18 Hyperparametric optimisation model

The confusion matrix for ResNet-18 hyper-parametric optimisation model is given in the figure 20

4.4. VGG-19

4.4.1 Baseline model

In the initial stages of learning curve for baseline model 21, both training and validation accuracies increase rapidly and the validation accuracy ovetakes the training accuracy. At

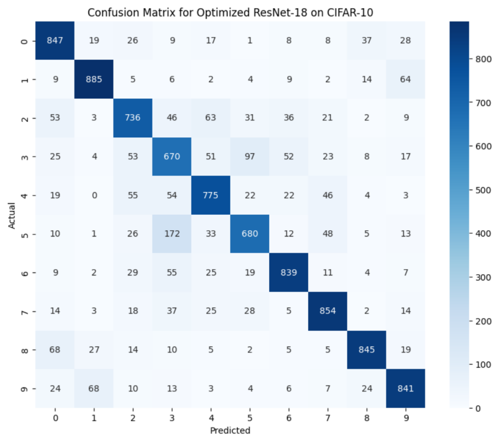


Figure 20. Confusion matrix for ResNet-18 Hyper-paarmetric optimisation model

around epoch 8, the training accuracy curve flattens and the validation accuracy curve stabilises. Both the curves have very minimal gap between them towards the end indicating a well-fitted model that would generalise well.

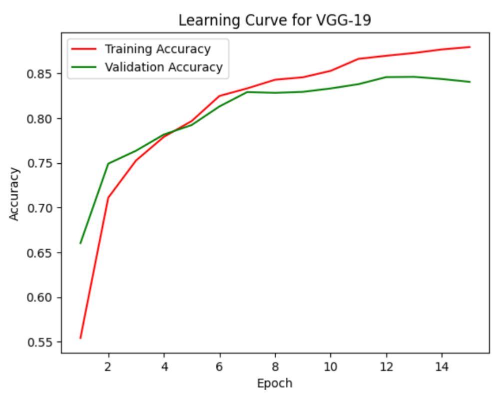


Figure 21. Learning curve for VGG-19 Baseline model

From the information provided by the classification report 22 of baseline model, we can understand that the overall accuracy for the model is 88% indicating strong perfromance. The recall, precision and F1-scores also stand at 81% because of well-balanced dataset.

Finally the confusion matrix for VGG-19 baseline model is given in 23.

4.4.2 Hyperparametric optimisation model

After performing hyper-parametrci optimisation on VGG-19 baseline model, the best hyperpaarameters are as follows: 'dropout_rate': 0.4, 'learning_rate': 0.0001, 'weight_decay': 1e-05.

For this model, the training accuracy curve increases rapidly from the beginning while the validation accuracy curve starts slow which can be seen in the learning curve 24. After epoch 6,the training accuarcy curve still rises whereas the validation accuaracy curve stabilises with slight fluctuations. The gap between the training and validation accuracy curves are minimum indicating a well-fitted model.

Classification Report:

	precision	recall	f1-score	support
0	0.91	0.86	0.88	1000
1	0.91	0.97	0.94	1000
2	0.90	0.84	0.87	1000
3	0.79	0.74	0.76	1000
4	0.83	0.90	0.86	1000
5	0.83	0.81	0.82	1000
6	0.93	0.92	0.92	1000
7	0.90	0.91	0.91	1000
8	0.92	0.95	0.94	1000
9	0.91	0.94	0.92	1000
accuracy			0.88	10000
macro avg	0.88	0.88	0.88	10000
weighted avg	0.88	0.88	0.88	10000

Figure 22. Classification report for VGG-19 Baseline model

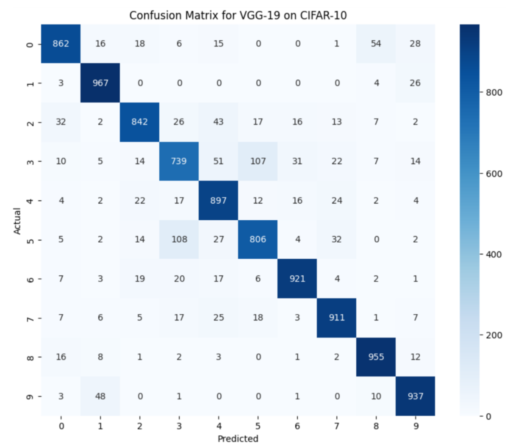


Figure 23. Confusion matrix for VGG-19 Baseline model

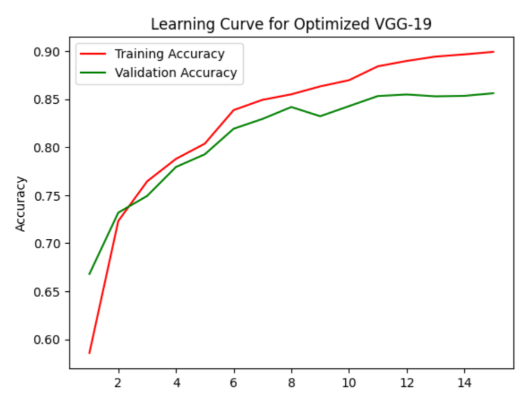


Figure 24. Learning curve for VGG-19 Hyperparametric optimisation model

From the classification report of VGG-19 Hyperpparametric optimisation model 25, we can infer that the overall accuracy is 90% which shows a very high performance compared to all other models. Even though there is slight difference between the values among classes, the overall precision, recall and F1-scores stand at 90%.

Classification Report:

	precision	recall	f1-score	support
0	0.93	0.90	0.91	1000
1	0.95	0.96	0.96	1000
2	0.87	0.90	0.88	1000
3	0.79	0.80	0.80	1000
4	0.86	0.89	0.87	1000
5	0.87	0.80	0.83	1000
6	0.94	0.93	0.93	1000
7	0.92	0.92	0.92	1000
8	0.92	0.96	0.94	1000
9	0.94	0.94	0.94	1000
accuracy			0.90	10000
macro avg	0.90	0.90	0.90	10000
weighted avg	0.90	0.90	0.90	10000

Figure 25. Classification report for VGG-19 Hyperparametric optimisation model

The confusion matrix for VGG-19 hyper-parametric optimisation model is given in the figure 26

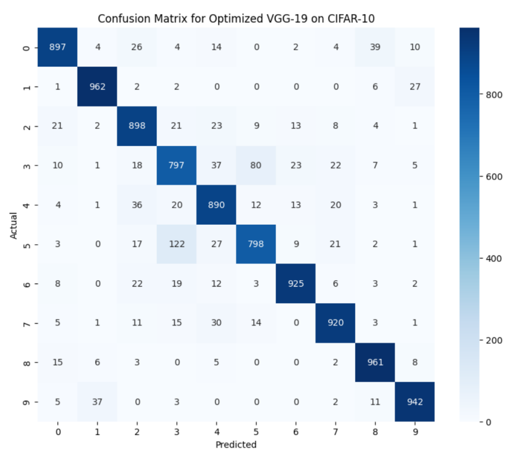


Figure 26. Confusion matrix for ResNet-18 Hyper-paarmetric optimisation model

5. Conclusion

Among all the models discussed, VGG-19 demonstrated highest performance in both baseline and hyper-parametric optimisation models with an accuracy of 88% and 90% respectively. Even the F1-score, recall and preicision values were very high compared to other models discussed. Both the models of VGG-19 neural network were very well-fitted with very low to no over-fitting. The VGG-19 model is closely followed by VGG-16 model whose performance was also better than ResNet-34 and ResNet-18 with baseline accuracy 85% and 86% after hyper-parametric optimisation. Both ResNet-18 and ResNet-34 showed similar performances with very close accuracy values. Therefore, because of it's high performance VGG-19 could be considered as one of the most powerful model which can be used for image classification for the CIFAR-10 dataset.

6. Summary and Ideas for Future work

The assignment evalueted 4 major CNN models VGG-16, VGG-19,ResNet-18 and ResNet-34 and compared their performance. Baseline models for each architecture were created using pre-trained models and to reduce iver-fitting techniques like data augmentation, early-stopping, dropout layers were used. Hyper-parametric optimisation using optuna was performed on each of the baseline models and each of there performance was evlauated on metrics like accuracy, F1-score, recall and precision. Results demonstrated that VGG-19 is the best among all models for image classification .

The following improvements can be made to the current studty to improve the results:

1. Model Ensemble: By combining both VGG and ResNet architectures, the robustness and the overall accuracy could improve.

2. Advanced Optimisers: We used Adam Optimiser in all of our models. However, an advanced optimiser like LAMB, SM3 could be experimented

3. Using Augmented Data strategies: Using advanced augmentation techniques like CutMix, MixUp can furter improve generalisation of the model.

References

[1] K Chatfield. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*, 2014. 1

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[3] Karen Simonyan. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[4] Venkata Kamala Sindhoori Vadde. Deep learning assignment 1. `https://github.com/Sindhu-Vadde/DeepLearning`, 2024. Accessed: 2024-10-05. 1

[5] Sergey Zagoruyko. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016. 1