ML Assignment Report - Sean Tan (s3806690), Duc Tran (s3752703)

In this assignment we have considered the use of 3 different models - a simple NN model as well as two CNN models, namely leNet and VGG. We have used these 3 models for both the isCancerous classification and the cellType classification, although not exactly the same as we've implemented different methods of validations and regularizations.

1) isCancerous classification

For cancerous cell classification, we decide to use hold out validation (also cross validation for the base nn model), the ratio is 18:6:6 (train:validation:testing). The reason for such low ratio is because of computation limitations, it simply would take too long to complete all the training and predicting. Also 2 of the CNN models use data augmentation so it can compensate for the small training data. For parameter tuning, regularization and drop out is the method of choice for all 3 models , due to their simplicity and ease of implementation. Cross validation is not used for the CNN models since through trial and error has shown that it is ineffective and time consuming, probably due to these 2 models complexity.

<u>Evaluation:</u>

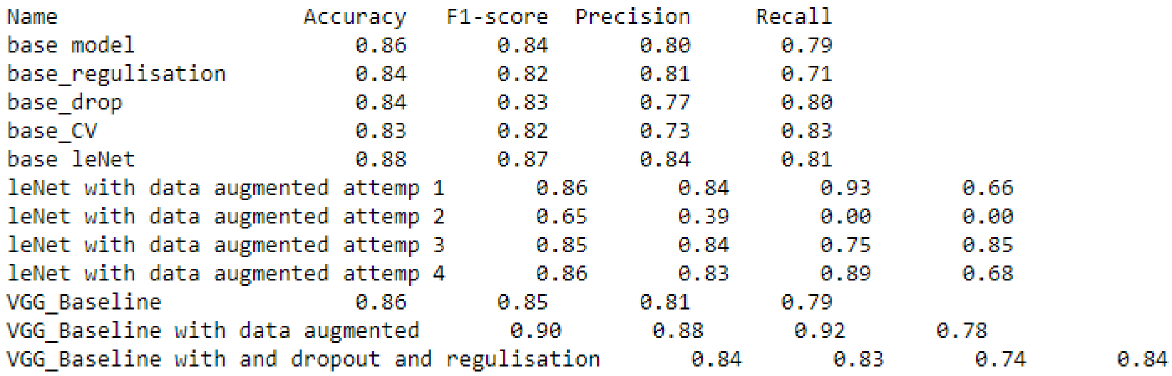
Simple NN model: It is observed that the regularised model actually performs worse than the base and dropout model. It has slightly lower accuracy score and appear to be a little overfitted

leNet : All leNet models use the same regularisation and dropout method, the only difference is how the data is augmented. It is shown in the notebook that more rotation, width and height shift result in better accuracy and fit. Model leNet 1,2,5 performed better than the rest with leNet 5 not susceptible to over or underfitting.

VGG: It is observed that the data augmentation technique used in leNet 5 increases the performance of VGG. While regularisation and drop out did nothing to improve the performance, and thus we concluded that we have reached the diminishing returns of regularization.

<u>Ultimate Judgement:</u>

From this table VGG_Base line with data augmentation seems to be the best model for cancer screening task in terms of f1-score which is 0.88.



```
Name                   Accuracy   F1-score  Precision     Recall
base model                 0.86       0.84       0.80       0.79
base_regulisation          0.84       0.82       0.81       0.71
base_drop                  0.84       0.83       0.77       0.80
base_CV                    0.83       0.82       0.73       0.83
base leNet                 0.88       0.87       0.84       0.81
leNet with data augmented attemp 1       0.86       0.84       0.93       0.66
leNet with data augmented attemp 2       0.65       0.39       0.00       0.00
leNet with data augmented attemp 3       0.85       0.84       0.75       0.85
leNet with data augmented attemp 4       0.86       0.83       0.89       0.68
VGG_Baseline               0.86       0.85       0.81       0.79
VGG_Baseline with data augmented       0.90       0.88       0.92       0.78
VGG_Baseline with and dropout and regulisation       0.84       0.83       0.74       0.84
```

2) cellType classification

For the cellType classification problem, we decided to use hold-out validation by using train-test-split from sklearn to split our data. The reason is due to the limitations of the processing speed of our notebooks whereby for the 2 CNN models, we had to further minimize the size of our train, validation and test sets by 30% (thus using only 70% of the data set) to have arguably appropriate waiting time for each model fitting (approx 20-30 mins per fit). Of course we had also done a plot comparison analysis between the original and the shrunk set to make sure that the shrunk set was a good representation of the original set whereby it had almost the same distribution of elements for each cellType and isCancerous category. For regularization on all 3 models, we used Dropout layer and added L2 regularizers to our Conv2D and Dense layers. We use the Hyperband tuner from Keras Tuner to assist in picking the best value of our lambda and Dropout value, as well as our Dense layer units and activation function. In our CNN models, we also used data augmentation such as rotations and width/height shifts to regularize our model's performance.

Evaluation

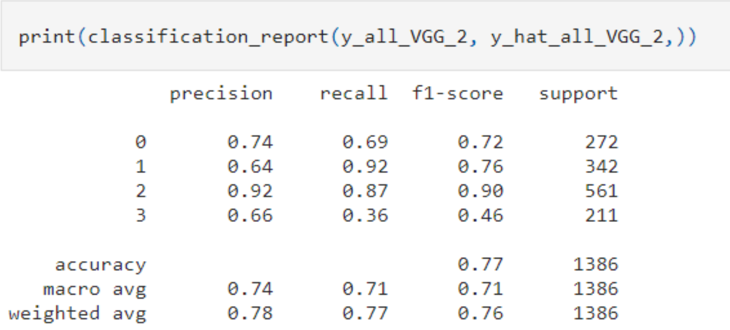
Simple NN model: We observe that the performance of the regularized model actually had a slightly lower performance compared to the initial model without regularization. We concluded that it was due to pure coincidence that the initial model was more optimized to predict the test set as it was slightly over-fitting the train model as compared to the regularized model.

leNet & VGG CNN models: We have used the same optimizing strategies for both of these models, namely data augmentation and Hyperband tuning from Keras Tuner. We observed that data augmentation has greatly regularized and improved the performance of the models. The hyperparameter tuning using Hyperband tuner did not improve the performance and thus we concluded that we've reached the diminishing returns of further tuning and hyperparameter optimizations.

Ultimate Judgement

We choose the VGG model with data augmentation only as our final and best model. It has the best and most balanced performance for the test set in predicting all classes of cellTypes, and also properly regularized to the validation set with no over-fitting.

Image below is the classification report of the our final model chosen:



```
print(classification_report(y_all_VGG_2, y_hat_all_VGG_2,))
```

```
              precision    recall  f1-score   support

           0       0.74      0.69      0.72       272
           1       0.64      0.92      0.76       342
           2       0.92      0.87      0.90       561
           3       0.66      0.36      0.46       211

    accuracy                           0.77      1386
   macro avg       0.74      0.71      0.71      1386
weighted avg       0.78      0.77      0.76      1386
```

'fibroblast' : 0, 'inflammatory' : 1, 'epithelial' : 2, 'others' : 3

Independent Evaluation

In K. Sirinukunwattana's research paper, he had done models with cellType (nucleus) classification and also a combined performance on nucleus detection and classification, with the conclusion of having just purely cellType classification yielding a better performance. Image below is the performances of his models which can be compared with our best model for cellType classification.

From the f1-score charts, we roughly estimate the softmax CNN + NEP model (which is the best model) have the following f1-scores:
0 - fibroblast : 0.72 vs 0.72(our model)
1 - inflammatory : 0.795 vs 0.76(our model)
2 - epithelial : 0.88 vs 0.90(our model)
3 - others/miscellaneous : 0.53 vs 0.47(our model)
Weighted average f1-score: 0.784 vs 0.76(our model)

From here we have observed that their model has a better overall performance on all categories namely 'inflammatory' and 'others' while our model very slightly neglected these categories in favor of the performance for categorizing 'epithelial'. Their model's weighted average is 0.024 (2.4%) higher than our model. We conclude that their use of NEP (Neighboring Ensemble Predictor) has helped them to achieve this which based on spatial ensembling leverages all relevant patch-based predictions in the local neighborhood of the nucleus to be classified (*K. Sirinukunwattana et al. 2016*). Further research also led us to believe that they had come up with the NEP algorithm themselves and not generally used nor available in keras or sklearn, thus with our limited knowledge on this field we are unable to replicate such an algorithm to be used on our model.

Appendix

References:

K. Sirinukunwattana, S. E. A. Raza, Y. Tsang, D. R. J. Snead, I. A. Cree and N. M. Rajpoot, ”Locality Sensitive Deep Learning for Detection and Classification of Nuclei in Routine Colon Cancer Histology Images,” in IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1196-1206, May 2016, doi: 10.1109/TMI.2016.2525803. 6