**Brain Tumour Classification with**

**kNN, Logistic Regression and Neural Network**

**Chua Yeong Hui, Lee Yu Xuan Olivia, Tan Kia Leng, Wong Xin Ying**

**Introduction**

Our machine learning (ML) model classifies a brain tumor as benign or malignant. This work would contribute to the medical field by automating the process of identifying malignant tumors, making diagnosis quicker and less susceptible to human error.

The AI and ML techniques we intend to use independently are K-Nearest Neighbours (kNN), Logistic Regression (LR), and Neural Network (NN). For the neural network, we used Convolutional Neural Network (CNN). We used KNN to account for non-linear relationships and LR to account for linear relationships in the dataset. However, LR is limited as it assumes no multicollinearity between feature variables (Rout, 2022) which is not common in complex real-life occurrences such as in brain tumors. Therefore, we implemented CNN too since it specializes in computer vision tasks such as image classification (Varghese, 2018). However, their disadvantage is the need for a large training dataset, which we would address using data augmentation. After implementing all aforementioned models, we would evaluate the models independently and determine the best model.

We investigated a version of CNN known as Very Deep Convolutional Networks for Large-Scale Image Recognition (VGG16) (Abdalslam & Klymentiev, 2019). VGG16 has many convolution layers, making it time-consuming to run (Thakur, 2019). As such, VGG16 is not viable for our project. However, we decided to adopt the cropping technique used in this work. Another work investigated was a CNN model with data augmentation (M K, 2020). However, an Adam optimiser was used here, which could be a drawback despite its fast computation time due to the inability to generalise as well as other optimisers such as Stochastic Gradient Descent (SGD). We referenced the data augmentation portion in our project.

**Dataset**

The dataset used is the Brain Tumour Dataset. The dataset has a mix of grayscale and RGB images of different sizes. In the malignant folder, there was a non-image DS file that should not be part of the dataset for image classification. The issues were corrected during data processing. We experimented with the raw dataset and an improved dataset in this work.

Improved Data Processing

The objective of improved data processing (Abdalslam & Klymentiev, 2019) is to remove all unwanted objects from images, such as annotations (eg. Malignant Image 6) by cropping. By delineating the brain in the improved dataset, it will be easier for our models to carry out classification. It requires four steps - 1. Converting all images to grayscale, 2. Applying binary thresholding, 3. Finding the contours of the brain and 4. Cropping the images.

We set all images to grayscale for thresholding. Since some were already in grayscale, an if-else statement was used to identify non-grayscale images before applying Gaussian blurring to remove Gaussian noise from the images. Through binary thresholding, the outline of the brain will be segmented to white (LearnOpenCV, 2021), facilitating contour detection in the next step. We found the contours of the thresholded images and the extreme points (top, bottom, left, right) along the contours. The grayscale images were cropped, resized to 244x244 and converted to arrays based on contours. The benign and malignant arrays are concatenated into X\_raw\_improved. Y\_improved is an array of 0s and 1s that correspond to the number of benign and malignant images respectively.

For both improved and original data processing methods, we carried out train-test splits before applying data augmentation on the training sets. Lastly, we normalised the training and testing x for the original data as a form of feature scaling to improve numerical stability.

**Methods**

KNN: kNN is a classification technique used to classify data based on the assumption that similar data points are close to each other (Harrison, 2018). Since we needed to classify images into benign and malignant datasets, we used kNN to compare the similarity of two images by using a similarity metric on their respective representations in numpy arrays. The default minkowski metric gave highest accuracy when k=1, leading to risk of overfitting. We therefore used the cosine metric which is more appropriate for our high-dimensional dataset.

Logistic regression: We used binomial LR to classify the validation data into 2 classes of diagnosis— Malignant & Benign. LR models are usually computationally less intensive than predictive models built using neural networks. In addition, the interpretation of LR model coefficients could serve as indicators of feature importance. Hence, LR may be preferred if tumour prediction is satisfactory. Our LR model's novelty lies in using data augmentation which increases the training dataset by generating new data points from existing training data.. Currently, data augmentation is more commonly used for deep learning (Seita, n.d). Using data augmentation with LR makes the training dataset richer and reduces the risk of overfitting.

Neural Network: NNs are specialised to handle nonlinear and complex relationships between input and output, which is ideal for our project involving brain tumors. Conv2D was used as our inputs are images with 2 spatial dimensions. Unlike previous works, our NN uses cropped versions of the dataset together with normalisation prior to training, and during training through BatchNormalisation. Batch Normalisation ensures that the mean and standard deviation of layer inputs stay constant, as computations in the convolution layers and changes in weights during training may cause changes in input distribution. We also implemented the SGD optimiser instead of Adam, which uses random batches of data instead of the all data for each iteration as in Gradient Descent (GD). We chose SGD for its lower computation time than GD, and higher generalisability than Adam (Gupta, 2022).

**Results & Discussions**

KNN: We evaluated the kNN model based on accuracy and balanced accuracy. This distinction led to us more judiciously using data augmentation to ensure a similar number of benign and malignant datasets. We also plotted the graph to see which k-value is most ideal for kNN.

Logistic regression: To evaluate the LR models, we utilised confusion matrix, accuracy, balanced accuracy, F1 score, precision, and recall. F1 score is crucial for monitoring FN. The cost of FN is high as we do not want to predict a tumour as benign when it is malignant (Huilgol, 2019). Hence, F1 score is more relevant than accuracy scores for our project. We also monitored recall as it is important when FN cost is high. After comparing the performance matrices between original and improved data (Appendix), both F1 score and recall increased when the improved dataset was used. Most importantly, the number of FN decreased from 13 to 10. The methods to get the improved dataset helped to better account for possible variability in the raw dataset, leading to the improvement. Other fine-tuning methods include the changes made to the default parameters of the LR model fitting function.

Neural Network: In order to evaluate our CNN model, we implemented the binary cross entropy loss function (in accordance with our binary variables of benign tumor or malignant tumor) and an accuracy metric. We fine-tuned performance by tweaking the number of layers (between 2 to 3), kernel size (between 3x3 and 5x5), padding (between valid and same), and optimizers (between SGD and Adam).

Our model functioned best with 3 convolutional layers and 1 dense layer. We opted for kernels of 3x3 as smaller kernel numbers are more discriminative, have lower computation cost, and reduce the number of weight parameters. We found that this indeed produced the best results. We found that “valid” padding worked better than “same” padding. With our dataset, our area of interest lies closer to the centre, hence there is no need to preserve edge pixels. Hence, there is no need for “same” padding. For optimisers, we discovered that SGD indeed performed better than Adam, as hypothesised earlier.

Initially, we found that the model produced an increasing loss and decreasing accuracy for the testing dataset, a sign of overfitting. We countered this by implementing Spatial Dropout, which drops a predetermined number of feature maps from the convolutional layer. Our results showed that the improved dataset yielded a lower accuracy despite a general increasing trend, and a higher loss (Appendix 2a, 2b). In contrast, the original dataset yielded a higher accuracy and lower loss (Appendix 3a, 3b)

Conclusion: From the results obtained by the 3 types of models, NN is the most effective model for brain tumour images classification, though it requires a high computation time and a larger training dataset. KNN comes a close second and can be used when there are fewer training datasets.

**References**

Abdalslam, L., & Klymentiev, R. (2019, June 21). *Brain tumor MRI classification: VGG16*. Kaggle. Retrieved October 31, 2022, from https://www.kaggle.com/code/loaiabdalslam/brain-tumor-mri-classification-vgg16

Gupta, A. (2022, May 24). *A comprehensive guide on Deep learning optimizers*. Analytics Vidhya. Retrieved October 31, 2022, from https://www.analyticsvidhya.com/blog/2021/10/a-comprehensive-guide-on-deep-learning-optimizers/

Huilgol, P. (2019) *Accuracy vs. F1-score*, *Medium*. Analytics Vidhya. Available at: https://medium.com/analytics-vidhya/accuracy-vs-f1-score-6258237beca2 (Accessed: October 31, 2022).

*Image Thresholding in opencv*. LearnOpenCV. (2021). Retrieved October 31, 2022, from https://learnopencv.com/opencv-threshold-python-cpp/#-binary-thresholding

M K, G. (2020, June 5). *Classification of brain MRI as tumor/non tumor*. Medium. Retrieved October 31, 2022, from https://towardsdatascience.com/classification-of-brain-mri-as-tumor-non-tumor-d48838ccc162

Rout, A. R. (2022, August 23). *Advantages and disadvantages of logistic regression. GeeksforGeeks.* https://www.geeksforgeeks.org/advantages-and-disadvantages-of-logistic-regression/ (Accessed: October 31, 2022).

Seita, D. (n.d.). *1000x faster data augmentation*. The Berkeley Artificial Intelligence Research Blog. Retrieved October 31, 2022, from https://bair.berkeley.edu/blog/2019/06/07/data\_aug/

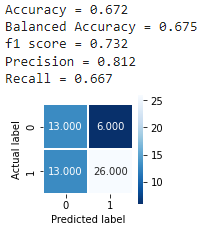
Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2015). Rethinking the inception architecture for computer vision. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/cvpr.2016.308

Thakur, R. (2019, August 6). *Step by step VGG16 implementation in Keras for beginners*. Medium. Retrieved October 31, 2022, from https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c

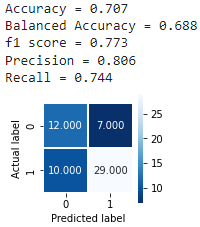
*Thresholding – Image Processing with Python.* DataCarpentry. (n.d.). Retrieved October 31, 2022, from https://datacarpentry.org/image-processing/07-thresholding/#:~:text=Thresholding%20is%20a%20type%20of,is%20simply%20black%20and%20white.

Varghese, D. (2018, December 7). *Comparative study on classic machine learning algorithms*. Medium. Retrieved October 31, 2022, from https://towardsdatascience.com/comparative-study-on-classic-machine-learning-algorithms-24f9ff6ab222

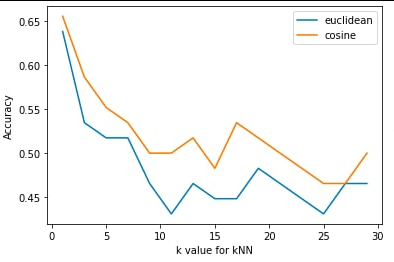
**Appendix**

****

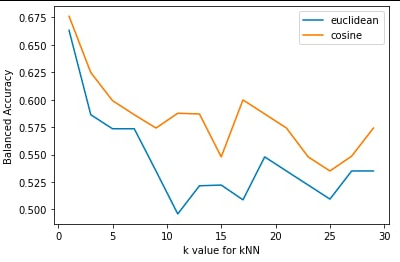
1a: for model\_ori

****

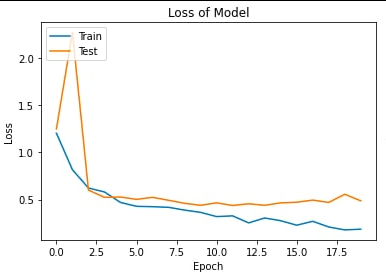
1b: for model\_imp



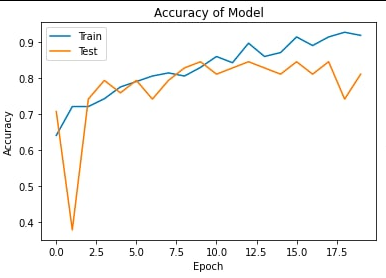
2a: Loss for NN using improved data



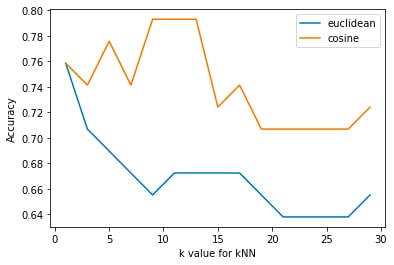
2b: Accuracy for NN using improved data



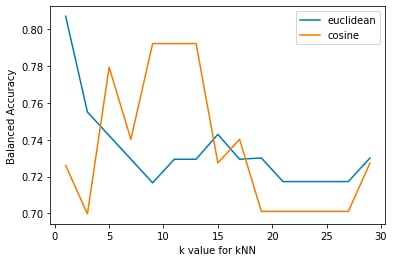
3a: Loss for NN using original data



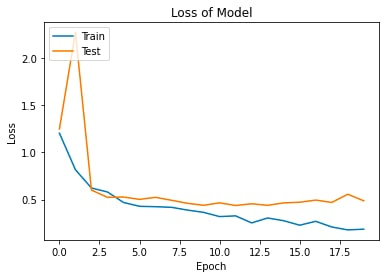
3b: Accuracy for NN using original data

****

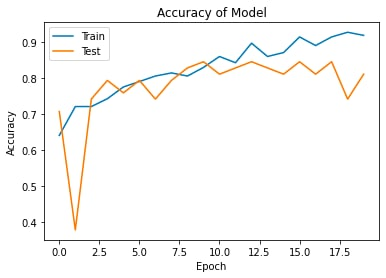
kNN accuracy using original data

****

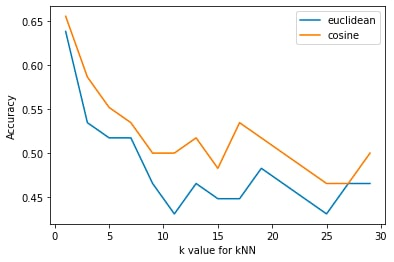
kNN balanced accuracy using original data

****

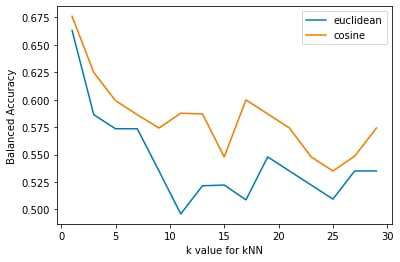
NN for original data

****

NN accuracy for original data

****

Knn accuracy for improved data

****