# Brain Tumour Classification using 3 Machine Learning Models

**Group 41: Lim Zhi Chao, Ang Qi Wei Chester, Faith Arielle Peh, Tan Ze Qi Christian Lambayong**

## Introduction

Globally, brain cancer is a leading cause of mortality and is an important research topic in medical imaging (Maqsood, Damaševičius, and Maskeliūnas 2022). Brain cancer is caused by the formation of cancerous brain tumours in the brain.

The symptoms of this condition can vary greatly depending on the location, size and type of brain cells affected. The complexity and difficulty involved hence means that detecting and classifying whether brain tumours are malignant (cancerous) or benign (non-cancerous) accurately especially at the initial stages for optimal treatment is challenging (Asiri et al. 2023). Furthermore, manual detection of brain tumours by radiologists is a procedure which is monotonous and prone to errors (Maqsood, Damaševičius, and Maskeliūnas 2022).

Currently, Magnetic Resonance Imaging (MRI) is the diagnostic test used for the detection of brain tumours and to find out essential information about the brain tumour such as its shape, location and size (ZainEldin et al. 2022). The amount of data regarding patients with brain tumours has also increased dramatically in recent years, causing old procedures for diagnosis of brain tumours to become much more costly, in addition to its low accuracy (ZainEldin et al. 2022). The use of machine learning methods for the classification of brain tumours is hence proposed to improve the accuracy and precision of the classification of brain tumours (whether the brain tumour is malignant or benign).

Therefore, our project's aim is to classify the brain tumours in the given dataset with as high an accuracy rate as possible. We use three artificial intelligence methods, namely the k-Nearest Neighbours Algorithm (KNN), Logistic Regression, and Convolutional Neural Network (CNN), to attempt to explore ways to find the most accurate and precise method for the classification of brain tumours in the dataset.

## Related Works

There has been a variety of research done in regards to brain tumour classification in recent years, with many works exploring the use of Neural Networks to improve the process of identification of the brain tumour type.

One such work that advanced research on the subject studied Deep Learning and Sine-Cosine Fitness Grey Wolf Optimization for Brain Tumour Detection and Classification (ZainEldin et al. 2022). The proposed model is a CNN hyperparameters optimization using an adaptive dynamic sine-cosine fitness grey wolf optimizer algorithm. The results of the work done showed that the model as a classifier achieved a high accuracy of 99.98% using the BRaTS 2021 Task 1 dataset. In this work, the major limitations and drawbacks were the long time needed to process due to the large number of extra optimization steps. While our own project will not be attempting to develop our own model, studying such related works emphasised to us the importance of choosing appropriate hyperparameter values for success and gave us insights as to the different ways to evaluate the effectiveness of various models as applied and demonstrated in the work, such as through statistical methods as well as the confusion matrix.

Another related work attempted to classify brain tumours using a k-Nearest Neighbor-Genetic Algorithm and Support Vector Machine-Genetic Algorithm Methods (Wibowo, Rustam and Pandelaki, 2021). This work focused on classifying Glioblastoma, a common type of brain tumour resulting in fatal disease. The classification was carried out using the various features available in the dataset that represented information about the tumours. To prioritise more crucial features that would affect the classification results, the Genetic Algorithm, a search and optimization method inspired by the natural selection process, was implemented for the purposes of feature selection for the KNN and Support Vector Machine (SVM) to produce accurate classification. Using various features and the confusion matrix, the performance of the algorithms were compared to find the most optimal method between the modified KNN and SVM methods used. This work gave us insights into feature selection and the possible use of various methods in the feature selection process in the training process. These were points that helped us consider possible improvements to our KNN algorithm beyond the scopes of the project.

Another related work attempted to use logistic regression on top of a CNN model to achieve better classification results. With the use of logistic regression to enhance the classification process, a reported accuracy score of 0.995 which outperforms existing approaches. While we will not be attempting such novel approaches to classify brain tumours in our project, this work helped us understand the importance of knowing the various machine learning models well, and that there is potential to enhance the accuracy and efficiency of existing approaches via combining and fine-tuning existing models.

Overall, studying these various related works done on the subject helped us understand the importance of finding ways to study various approaches to classifying brain tumours, as well as going beyond what is learnt in the module to consider ways to improve existing solutions and approaches to solving issues like brain tumour classification.

## Machine Learning Models

### Dataset

For this section, a common processed dataset is used to run the KNN and Logistic Regression algorithms. Before training the model, the dataset was cleaned and standardised. For each image, black padding was removed. To remove black padding, the images were loaded in grey scale and had their contrasts enhanced to enhance visibility. The image foreground (brain) and background (black padding) were then identified and separated. After cropping out the black padding for each image, they were all resized to 224x224. Black padding does not contain useful information and it could potentially distract the model during training. Removing them allows for better focus on relevant features to distinguish between benign and malignant brain tumours. Having a consistent input size for all images reduces variability in the input data and as a result, reduces computational complexity.

Before using KNN and Logistic Regression, the cleaned image data is first normalised because it preserves relative relationships between the pixels and maintains structural characteristics. This ensures consistent data processing and improved classification accuracy of the KNN algorithm.

The original dataset is used to run the CNN algorithm as prior cleaning is not required in order to run the algorithm. The same pixel size 224x224 was used for all three models.

## Convolutional Neural Network (CNN)

### Methodology

The images are imported and resized to 224 by 224 pixels. The dataset is split into testing, validating and training with a ratio of 60:20:20. Create a data augmentation layer and a preprocessing layer. Data augmentation like random rotations, Gaussian Noise and random Zooms effectively create a larger dataset by transforming the images, creating new variations of the data. The preprocessing layer resizes the image pixel values between [-1,1], which is what the MobileNetv2 model requires. This rescaling is effectively the normalisation of the image data. We use MobileNetv2 for our feature extraction and create the classification layer to turn extracted features into a single prediction.

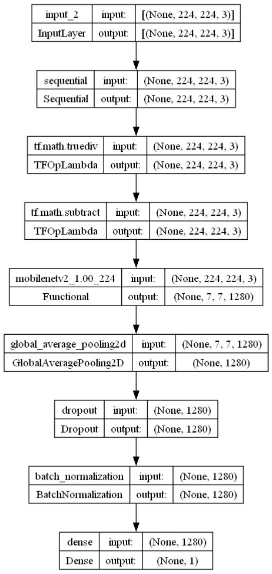


*Figure 1: CNN Model Flow*

After creating the model, we run the first training. After training, we fine-tune the model by unfreezing some layers of the feature extraction model and retraining the model. After fine-tuning, we hyperparameter tune the model to determine the optimal learning rate and the best epoch. We re-train the model with the optimal hyperparameter. Finally, we use the model to predict our test values.

### Results


*Figure 2: Accuracy and Validation Accuracy Graph. Loss and Validation Loss Graph*

After fine tuning and hyperparameter tuning, the graphs above were produced. The validate accuracy was generally higher than the training accuracy while the validate loss was generally lower than the training loss. This indicates that the model was rarely overfitting. These results were obtained with a learning rate of $1.13 \times 10^{-6}$. The low learning rate resulted in low increase in accuracy and decrease in loss. However, the aim was to get a good accuracy and not overfit which was achieved. The results maintained a high accuracy while also keeping a low FNR and FPR, as shown below.

**Accuracy:** 0.875
**False Negative Rate (FNR):** 0.09523809523809523
**False Positive Rate (FPR):** 0.18181818181818182
**Precision:** 0.9047619047619048

### Optimisation

Specificity versus Generalisation: To improve specificity of the model to the dataset, we fine tuned the model by training some layers of the pre-trained model. We noticed that between 10% to 20% of the layers should be unfreezed to allow a balance between specificity and generalisation.

Learning and Dropout rate : We used a hyperparameter tuner Hyperband to search for the optimal learning rate. We found that the learning rate range was between $1 \times 10^{-4}$ and$1 \times 10^{-6}$. Learning rate was reduced between trainings to prevent overfitting. Dropout rate was manually decided between the range of 0.0 to 0.5. We chose 0.3 after testing to prevent overfitting and maintain generalisation.

### Limitations

The dataset size was a limiting factor for the model. The small dataset size resulted in occasional overfitting. As there is very little training data, the model would converge very quickly and overfit. To reduce overfitting, we choose to use smaller learning rates and reduce the number of epoch training. Small datasets may not fully represent the diversity of the real-world data distribution. As a result, models trained on small datasets may struggle to generalise to new, unseen data.

## k-Nearest Neighbours (KNN) Algorithm

### Methodology

The normalised data is split into testing and training with a ratio 20:80. We then run the KNN classifier thrice. The first run uses the training data using N-Fold cross-validation and iterates over different values of K to find the optimal K based on its averaged accuracy score. Secondly, using the optimal K found, the algorithm then determines the best threshold value through evaluating recall score, TPR, FPR, FNR, TNR. Lastly, using the optimal K and threshold value, we test the whole dataset and compare the overall accuracy of the KNN. The whole cycle is run multiple times with different randomised seeds to find the average accuracy, which reduces bias.

### Results

**Optimal K**: After running multiple iterations with random seed, the most frequent values are K=11,13,15 , so we take the average hence **optimal K = 13.**
**Best Threshold**: With K=13, after computing the confusion matrix and plotting TPR,FPR,FNR and TNR (Figure 2) for better comparison. We can conclude that a threshold value of **Threshold = 0.5** will be good.

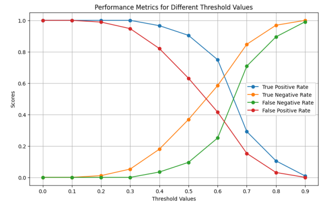


*Figure 3: Performance Metrics for Different Threshold Values*

**Validation Accuracy:** The final validation accuracy for the model is **74.9%** (3s.f)
**Validation Recall Score:** The final validation accuracy for the model is **89.0%** (3s.f)

### Optimisation

After the initial testing, we tried to optimise some variables to see if there are any improvements to the KNN algorithm.
1. **Different split ratio for test and train**: We tested different ratios of testing and training data. As seen in figure Z, we notice an optimal ratio for will be **between 0.16 and 0.22**.


*Figure 3: Accuracy and Recall scores for Different Test Split Ratios*

2. **Cleaned vs original dataset**: We decided to test the original uncleaned dataset to compare the difference in accuracy. After multiple iterations, the accuracy is **61.7%** (3s.f) and the recall **55.9%** (3s.f). We further explored using optimal K=13 and threshold = 0.5 from the cleaned dataset, which results in accuracy of **68.1%** (3s.f) and recall of **85.3%** (3s.f). Cleaning the dataset resulted in an increased accuracy and recall rate, which is important in the context of brain tumour classification.

### Limitations

KNN algorithm is sensitive to noisy data and outliers, making it susceptible to overfitting which can lead to misclassifications, which is a critical concern when dealing with brain tumour classification. Moreover, the computational demands of KNN can pose significant obstacles, especially with large-scale, high-resolution brain images, potentially delaying real-time diagnosis and treatments. Optimisation of search processes using specialised data structures and algorithms can be able to help enhance KNN algorithm's efficiency and accuracy in the classification of brain tumours.

## Logistic Regression

### Methodology

In the context of brain tumour classification, the objectives of maximising True Positives (TP) and minimising False Negatives (FN) in predictions are key. Hence, the model was trained with the objective of maximising the Recall Score, TP/(TP+FN). The normalised data was split into training and testing sets using a 8:2 ratio. Hyperparameters to maximise recall score for the model were then chosen. These parameters include penalty, C, solver, and maximum number of iterations. Penalty determines the type of regularisation used in the model, a technique to prevent overfitting. C determines the strength of regularisation. The solver determines the algorithm to be used in the optimization problem. It was found that the most ideal values for these parameters were “l1” for penalty, “10” for C, “liblinear” for solver, and the maximum number of iterations being 100. After fixing the model with these ideal parameters, we proceeded to find the ideal threshold value for the model. We evaluated the suitability of threshold values using the Receiving Operation Characteristic (ROC) curve and Area Under Curve (AUC) evaluation metric. The average threshold value found to maximise the AUC was 0.7. This threshold value of 0.7 provides an optimal balance in the objectives of maximising True Positives and minimising False Positives. The average AUC was found to be around 0.77. Finally, the model was fixed with the found hyperparameters and threshold value.

### Results

The model was then evaluated on its accuracy and recall score. Using N-Fold Cross Validation with the number of folds being 5, the predictions had an average accuracy of 0.75 and  average recall score of 0.85.

### Limitations

One limitation is that logistic regression models can be prone to overfitting, especially if the number of features is large compared to the number of samples. In the context of brain tumour classification, the number of features for classification can be vague. While regularisation techniques can be applied to mitigate overfitting, they might not always be sufficient, especially with limited data. Lastly, logistic regression models assume linearity between the input features and log-odds of the target variable. Complex images may cause such an assumption not to hold.

## Discussion of results

| Model | Validation accuracy (3s.f.) |
|---|---|
| Convolutional Neural Network (CNN) | 89.0% |
| k-Nearest Neighbours (KNN) | 74.9% |
| Logistic Regression | 75.0% |

Across the 3 models, CNN performed the best overall with validation accuracy as the comparison metric. This is likely because CNN is much more effective for image classification tasks as compared to KNN and Logistic Regression due to the use of the convolutional layers to extract key features in images and learn to recognise complex patterns. CNN is hence likely better suited for the task of classifying brain tumours as compared to KNN and Logistic Regression with its ability to use parameter sharing as well as other forms of optimisation to improve the accuracy of the model.

The Logistic Regression model performed better than KNN. This could be likely attributable to the fact that Logistic Regression is better able to handle irrelevant or redundant features by learning the underlying patterns in the presence of noise, as compared to KNN which is much more sensitive to noisy data.

## Limitations

A limitation of the 3 models are that the dataset used may not fully account for the full variation in brain tumour cases in reality. This may hence introduce biases in the results and models which would affect the accuracy of the models. Additionally, the imbalance in data with more malignant brain tumour data as compared to benign tumours may cause the models built to be biased as well.

## Conclusion

Overall, the CNN model had the best performance out of the 3 Machine Learning models. However, as compared to related works done for similar brain tumour classification tasks, the validation accuracy of our CNN model is lower than many of the current works done in this area of study. There are hence various possible areas of improvement and optimisation that are worth considering to add and work on in our models, such through the use of finding ways to further enhance the hyperparameter tuning process, as well as innovating and considering ways to combine other algorithms to increase the performance of the model.

## References

Putri Wibowo, V. V., Rustam, Z., & Pandelaki, J. (2021). Classification of brain tumour using K-nearest neighbour-genetic algorithm and support vector machine-genetic algorithm methods. *2021 International*

*Conference on Decision Aid Sciences and Application (DASA)*. https://doi.org/10.1109/dasa53625.2021.9682341

Saidani, O., Aljrees, T., Umer, M., Alturki, N., Alshardan, A., Khan, S. W., Alsubai, S., & Ashraf, I. (2023). Enhancing prediction of brain tumour classification using images and numerical data features. *Diagnostics*, *13*(15), 2544. https://doi.org/10.3390/diagnostics13152544

ZainEldin, H., Gamel, S. A., El-Kenawy, E. M., Alharbi, A. H., Khafaga, D. S., Ibrahim, A., & Talaat, F. M. (2022). Brain tumour detection and classification using deep learning and Sine-Cosine Fitness Grey Wolf optimization. *Bioengineering*, *10*(1), 18. https://doi.org/10.3390/bioengineering10010018