**CLASSIFICATION OF BRAIN GLIOMA TYPES IN HUMANS USING MACHINE LEARNING ALGORITHMS**

Obot Enomfon and Obianuju Okafor

Faculty of Engineering, Environment and Computing, Coventry University

MSc Data Science Coventry, England

[obote@coventry.ac.uk](mailto:obote@coventry.ac.uk)

[okaforo7@coventry.ac.uk](mailto:okaforo7@coventry.ac.uk)

**Abstract: In this paper, the primary objective is to perform the classification of gliomas using machine learning algorithms without conducting molecular tests. We want to predict if a patient has Lower-Grade Glioma (LGG) or Glioblastoma Multiforme (GBM) using the status of certain genes as well as clinical features. Molecular testing in addition to being expensive, can give inaccurate results based on the complexity of the biopsy which can lead to confusion and errors if misinterpreted or misapplied. The primary objective is to enhance efficiency and reduce costs associated with molecular testing by identifying the best subset of clinical features and mutation genes for the glioma grading process. This project makes use of scientific libraries and the Python programming language to apply classification methods using the Glioma Grading Clinical Mutation Features dataset that was obtained from the UCI website.**

***Keywords: classification; machine learning; glioma grading clinical mutation features dataset; python.***

1. INTRODUCTION

Glioma is a tumour that begins in the brain or spinal cord. Gliomas are made up of cells that resemble glial cells, which are normal brain tissue. Glial cells support and enhance the function of nerve cells [[1](https://www.mayoclinic.org/diseases-conditions/glioma/symptoms-causes/syc-20350251)].

In the 2016 World Health Organisation (WHO) Classification of Tumours of the Central Nervous System, glioblastoma, oligodendroglioma, and astrocytoma are the three types of adult diffuse gliomas. Gliomas can be classified into I to IV stages. Stages I and II represent low-grade gliomas (LLG), while stages III and IV represent high-grade gliomas (Glioblastoma multiforme grade, GBM) [[2](https://doi.org/10.3390/cells10113169)].

Brain and central nervous system cancer ranks 12th among the main causes of cancer deaths (2.5% of all cancers) and 19th among the most common malignancies (1.9% of all cancers), according to Global Cancer Observatory (GLOBOCAN) 2020 estimates. These figures indicate that brain and central nervous system cancer contributes significantly to the global burden of disease [[5](https://doi.org/10.3322/caac.21660)].

Based on WHO projections, there will be roughly 70,000 deaths and 85,000 new cases of brain and central nervous system cancer in the European Region in 2030 compared to the 82,000 cases and 58,000 deaths from brain cancer reported for both sexes combined in 2015[[6](https://doi.org/10.1016/j.heliyon.2023.e18222)].

The WHO classification of CNS tumours formulates a paradigm for how CNS tumour diagnoses should be organized in the molecular era by using molecular characteristics in addition to histology to classify several tumour entities for the first time [[3](https://doi.org/10.1007/s00401-016-1545-1)].  
Molecular testing is a collection of tests that examines a specimen’s genetic code [[4](https://www.medicalnewstoday.com/articles/molecular-testing)]. One major challenge is the possibility of erroneous outcomes if misconstrued. Our options for solving this issue include machine learning techniques. The accurate classification of Brain Gliomas would be a significant advantage for cancer research.

Considering the most recent data on the incidence, mortality, and association between brain and central nervous system cancer and global socioeconomic development [[7](https://doi.org/10.3390/ijms232214155)], finding the best way to detect the type of glioma early and effectively is important with the aim of using machine learning techniques.

This paper discusses and compares the results of applying five Machine Learning techniques to find the best accuracy score in predicting brain glioma types in humans.  
Machine learning describes a set of procedures that include developing and accessing algorithms that aid in pattern identification, classification, and prediction using models made from data.   
Without being specifically taught to do so, machine learning algorithms create the model using sample data, or "training data," to make predictions [[2](https://doi.org/10.3322/caac.21660)].

1. DATASET

The dataset was obtained from the UCI Machine Learning repository (Appendix A). It contains 839 instances each containing 23 attributes with no missing values. 3 clinical features and the 20 most frequently mutated genes from the TCGA-LGG and TCGA-GBM brain glioma projects are taken into consideration in this dataset.

Appendix A shows the summary of the TCGA dataset describing the feature and the feature role.

The authors proposed the use of Ensemble Machine Learning techniques to analyse the given dataset. The model consisted of soft-voting-based ensemble learning, employed by four dimensionality reduction methods to conduct a voting-based ensembled feature selection and five supervised models, with a total of sixteen combination sets. This proposed method achieved 87.6% accuracy rates on the TCGA dataset [[7](https://doi.org/10.3390/ijms232214155)].

1. DATA PREPARATION
2. Class Imbalance

A model that primarily predicts the majority class may have a high accuracy score when dealing with data that exhibits an imbalanced class distribution. This is suspicious because basic classification methods may find it difficult to identify and accurately predict occurrences from the minority class if 99% of the data belongs to the majority class [[9](ISBN:%20978-1-449-36941-5),[10](file:///C:\Users\eobot\OneDrive\Desktop\Importance%20of%20feature%20scaling.%20(n.d.-b).%20Scikit-learn.%20https:\scikit-learn.org\stable\auto_examples\preprocessing\plot_scaling_importance.html)]. To avoid this situation, a class imbalance check was conducted on the dataset. The code was obtained from [[9](ISBN:%20978-1-449-36941-5)] and modified to suit the dataset. After conducting the analysis, it was confirmed that the difference in the dataset classes is negligible.

A graph of a number of blue squares

Description automatically generated

Figure 1. Bar graph showing the class distribution.

1. Feature Encoding

Machine learning algorithms only work with binary values, however categorical data can be represented as binary values. To do this categorical data has to be encoded into binary values. In this paper, the dataset imported from the UCI website was already converted to binary values, hence feature encoding is not required.

1. Data Normalization and Scaling

Normalization is a rescaling of the data from the original range so that all values are within the new range of 0 and 1. It requires that the minimum and maximum observable values can be accurately estimated from available data [[11](https://machinelearningmastery.com/standardscaler-and-minmaxscaler-transforms-in-python/)].

Scaling features in a dataset before lowering dimensionality can yield components with the same order of magnitude [[10](https://scikit-learn.org/stable/auto_examples/preprocessing/plot_scaling_importance.html)] hence, normalization was used to scale the dataset so that the values for all features in the dataset would fall within the range of 0 and 1.

1. Feature Selection

Feature Reduction is reducing the number of features to be processed while avoiding information loss, and it expedites data processing by allowing users to reduce data to two or three dimensions for visualization. There are different methods of feature selection [[18](file:///C:\Users\eobot\OneDrive\Desktop\ISBN-13%20:%20978-1783988365)]. In the paper, two approach were made, one machine learning techniques were applied with feature reduction and another without feature reduction and PCA was used for feature reduction.

1. EXPERIMENTAL SETUP AND MACHINE LEARNING CLASSIFICATION TECHNIQUES

The dataset used for this project has been initially cleaned, it had no missing values and has been transformed into binary values hence feature encoding wasn’t required.

A class balance check was done to ensure that there was no class imbalance, while the two classes in the dataset used where not perfectly balanced, the difference between both classes was negligible. Thus, no data balance technique was applied for this project. However, data normalization was used to scale the dataset ensuring that the values for all features in the dataset fall within the range of 0 and 1 and thus guaranteeing that the applied techniques will work correctly as the features will be weighted equally.   
To begin, the environment must first import libraries with specified functions to perform necessary activities (see Appendix B). The classification methods rely on the scikit-learn library, which includes Python-specific functions.  
Second, the Pandas software was used to import the dataset in CSV format and assign it to a variable. The following stage is the preparation stage, which in this work includes the class balance check, data splitting, and data scaling.

The data needs to be randomly split into training and testing sets using scikit-learn function train\_test\_split which separates it into variables for attributes and labels being trained and assessed. In this paper, 80% of the data was used for training and 20% for testing (Appendix C).

After all the previous steps, modelling with different machine learning algorithms take place, each classification algorithm has an accuracy test score.

This project implements five classification methods as stated below:

* K-Nearest Neighbours (KNN)
* Decision Tree
* Gradient Boost
* Logistics Regression
* Support Vector Machine (SVM).

This paper analysed the dataset with the application of PCA and without the application of PCA.

Figures 2 and 3 show the cumulative explained variance of the dataset for the feature reduction using PCA and no feature reduction, respectively.

A graph with a red line

Description automatically generated

Figure 2. Line graph showing the variance of the dataset explained by PCA reduction features.

A graph with red dots

Description automatically generated

Figure 3. Line graph showing the variance of the dataset explained by all the features.

**KNN**

K-Nearest Neighbours employs closeness to classify or forecast how to group a single data point. In the k-NN algorithm, the k value defines the number of neighbours that will be checked to classify a given query point. Defining k can require some balancing, higher values of k may result in reduced variance and higher bias, while lower values may have low bias and higher variance, therefore the choice of k will mostly depend on the input data. Overall, it is recommended to have an odd number for k to avoid ties in classification. In this project, the K-value = 5 [[14](https://www.ibm.com/topics/knn#:~:text=The%20k%2Dnearest%20neighbors%20(KNN)%20algorithm%20is%20a%20non,used%20in%20machine%20learning%20today)].

**SVM**

The Support Vector Machine algorithm uses a collection of training samples, each set labelled as belonging to one of the two categories, to group data. Afterward, the algorithm creates a model by adding more values to either group. However, when the data set has additional noise, such overlapping target classes, SVM is less effective. [[12](https://www.sas.com/en_gb/insights/articles/analytics/machine-learning-algorithms.html),[13](https://doi.org/10.1007/s42979-021-00592-x)].

**Decision Tree**

A decision tree begins with a root node that has no incoming branches. The internal nodes, sometimes referred to as decision nodes, receive input from the outgoing branches of the root node. Both node types perform assessments based on available attributes to create homogeneous subsets, which are represented by leaf nodes or terminal nodes. The decision tree’s complexity plays a major role in classifying all the data points as homogenous sets or not [[15](https://www.ibm.com/topics/decision-trees#:~:text=A%20decision%20tree%20is%20a,internal%20nodes%20and%20leaf%20nodes)].

**Logistics Regression**

Logistics regression is a statistical technique used for building machine learning models with binary dependent variables. It is used to describe data and the relationship between one dependent and one or more dependent variables which can be nominal ordinal or of interval type. It is used in predictive modelling where the model determines the mathematical probability of whether an instance falls into a particular category or not [[16](https://www.simplilearn.com/tutorials/machine-learning-tutorial/logistic-regression-in-python),[17](https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-logistic-regression/#:~:text=Practices%20for%202022-,What%20Is%20Logistic%20Regression%3F,1%2C%20or%20true%2Ffalse)].

**Gradient Boost**

Gradient Boosting is an ensemble technique that builds a more potent model by combining several decision trees. It creates a final model by combining several separate models, usually decision trees. To minimize loss and increase model performance and generalization, it computes the second-order gradients of the loss function (L1 and L2) thus reducing over-fitting. Gradient-boosted trees frequently employ very shallow trees, ranging in depth from one to five, which reduces the memory requirements of the model and speeds up prediction times. [[9](ISBN:%2020978-1-449-36941-5),[13](https://doi.org/10.1007/s42979-021-00592-x)].

V. DATABASE AND CODE

All the files needed to perform the experimentation have been uploaded to GitHub profile (Appendix D).

The files included in the repository are:

* Jupyter Notebook
* Dataset (glioma.csv)

VI. EXPERIMENT RESULTS

The models were trained using all the features of the dataset and then trained again after PCA dimension reduction was applied to the dataset leaving us with nineteen features that explained about 93% of the variance in the dataset. The results were estimated using accuracy scores to determine the performance of the different machine learning techniques applied. Confusion Matrices, and model metrics for all the models were also obtained.

Model evaluation metrics include:

* Accuracy
* Precision
* Specificity
* Sensitivity / Recall
* F1 Score

**KNN Confusion Matrix**

A chart of a confused matrix

Description automatically generated

Figure 4. With feature reduction

A chart of a confused matrix

Description automatically generated

Figure 5. Without feature reduction

**Decision Tree Confusion Matrix**

A chart with different colored squares

Description automatically generated

Figure 6. With feature reduction

A chart with different colored squares

Description automatically generated

Figure 7. Without feature reduction

**Gradient Boost Confusion Matrix**

A chart with different colored squares

Description automatically generated

Figure 8. With feature reduction

A chart with different colored squares

Description automatically generated

Figure 9. Without feature reduction

**Support Vector machines Confusion Matrix**

A chart of confusion matrix

Description automatically generated

Figure 10. With feature reduction

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | KNN | Decision Tree | Gradient Boost | LR | SVM |
| Accuracy | 85.1 | 84.5 | 85.1 | 86.9 | 86.9 |
| Precision | 82.8 | 73.0 | 82.9 | 82.0 | 82.0 |
| F1 Score | 86.7 | 77.4 | 84.5 | 86.9 | 86.9 |
| Sensitivity/Recall Score | 91.1 | 82.3 | 86.1 | 92.4 | 92.4 |
| Specificity | 83.1 | 73.0 | 84.3 | 82.0 | 82.0 |

A chart of confusion matrix

Description automatically generated

Figure 11. Without feature reduction

**Logistics Regression Confusion Matrix**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | KNN | Decision Tree | Gradient Boost | LR | SVM |
| Accuracy | 86.9 | 83.9 | 83.9 | 87.5 | 86.9 |
| Precision | 81.6 | 79.5 | 82.5 | 82.2 | 82.0 |
| F1 Score | 85.5 | 81.4 | 83.0 | 87.5 | 86.9 |
| Sensitivity/Recall Score | 89.8 | 83.5 | 83.5 | 93.6 | 86.9 |
| Specificity | 82.0 | 80.8 | 84.2 | 84.2 | 84.2 |

A chart with different colored squares

Description automatically generated

Figure 12. With feature reduction

A chart with different colored squares

Description automatically generated

Figure 13. Without feature reduction

The accuracy and model metrics scores have been compared in the tables below. Training the models without any feature reduction gave the best results.

Table 1. Accuracy and Model metrics scores with feature reduction

Table 2. Accuracy and Model metrics scores without feature reduction

VII. CONCLUSION AND RECOMMENDATION

From the tables above, the Logistic Regression model performed best with both PCA feature reduction applied and without PCA feature reduction applied. The best trained model displayed good performance; however deep learning approaches can help it achieve even greater optimization.

Future works could involve the application of Neural network techniques to both augment the dataset and process the dataset to generate an overall improved model and better results.

VIII. ETHICAL CONSIDERATIONS

The dataset used was obtained from a publicly available UCI machine learning repository. However, the dataset was originally obtained from TCGA project being carried out by the National Cancer Institute. Both organizations are guided by certain ethical policies which include but are not limited to protecting the privacy of participants donating specimens, obtaining consent from participants when collecting specimens, establishing controlled assess for collected data, ensuring that data subjects understand the risks and benefits of participation.

Hence, the data used was obtained within acceptable ethical boundaries [19]. In this report, no data or result was fabricated, neither was any change made or data purposefully omitted to misrepresent results, misappropriate the ideas, writings, research, or findings of others [20].

**REFERENCE**

[1] *Glioma - Symptoms and causes - Mayo Clinic*. (2024b, March 7). Mayo Clinic. https://www.mayoclinic.org/diseases-conditions/glioma/symptoms-causes/syc-20350251

[2] Wu, Y., Guo, Y., Ma, J., Sa, Y., Li, Q., & Zhang, N. (2021). Research progress of Gliomas in machine Learning. *Cells*, *10*(11), 3169. https://doi.org/10.3390/cells10113169

[3] Louis, D. N., Perry, A., Reifenberger, G., Von Deimling, A., Figarella‐Branger, D., Cavenee, W. K., Ohgaki, H., Wiestler, O. D., Kleihues, P., & Ellison, D. W. (2016). The 2016 World Health Organization Classification of Tumors of the Central Nervous System: a summary. *Acta Neuropathologica*, *131*(6), 803–820. https://doi.org/10.1007/s00401-016-1545-1

[4] Villines, Z. (2022, November 30). *What is molecular testing?* https://www.medicalnewstoday.com/articles/molecular-testing

[5] Sung, H., Ferlay, J., Siegel, R. L., Laversanne, M., Soerjomataram, I., Jemal, A., & Bray, F. (2021). Global Cancer Statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for thirty-six cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, *71*(3), 209–249. https://doi.org/10.3322/caac.21660

[6] Ilić, I., & Ilić, M. (2023). International patterns and trends in the brain cancer incidence and mortality: An observational study based on the global burden of disease. *Heliyon*, *9*(7), e18222. https://doi.org/10.1016/j.heliyon.2023.e18222

[7] Taşçı, E., Zhuge, Y., Kaur, H., Camphausen, K., & Krauze, A. (2022). Hierarchical Voting-Based Feature Selection and Ensemble Learning Model Scheme for Glioma Grading with Clinical and Molecular Characteristics. *International Journal of Molecular Sciences*, *23*(22), 14155. https://doi.org/10.3390/ijms232214155

[8] GfG. (2024, March 18). *How to handle imbalanced classes in Machine Learning*. GeeksforGeeks. https://www.geeksforgeeks.org/how-to-handle-imbalanced-classes-in-machine-learning/

[9] Müller, A. C., & Guido, S. (2016). *Introduction to Machine Learning with Python: A Guide for Data Scientists*. O’Reilly Media.

[10] *Importance of feature scaling*. (n.d.-b). Scikit-learn. https://scikit-learn.org/stable/auto\_examples/preprocessing/plot\_scaling\_importance.html

[11] Brownlee, J. (2020b, August 27). *How to use StandardScaler and MinMaxScaler Transforms in Python*. MachineLearningMastery.com. https://machinelearningmastery.com/standardscaler-and-minmaxscaler-transforms-in-python/

[12] *A guide to the types of machine learning algorithms*. (n.d.-b). SAS UK. https://www.sas.com/en\_gb/insights/articles/analytics/machine-learning-algorithms.html

[13] Sarker, I. H. (2021). Machine learning: algorithms, Real-World applications, and research directions. *SN Computer Science*, *2*(3). https://doi.org/10.1007/s42979-021-00592-x

[14] *What is the k-nearest neighbors algorithm? | IBM*. (n.d.). https://www.ibm.com/topics/knn#:~:text=The%20k%2Dnearest%20neighbors%20(KNN)%20algorithm%20is%20a%20non,used%20in%20machine%20learning%20today

[15] *What is a Decision Tree? | IBM*. (n.d.). https://www.ibm.com/topics/decision-trees#:~:text=A%20decision%20tree%20is%20a,internal%20nodes%20and%20leaf%20nodes

[16] Banoula, M. (2023, November 7). *An introduction to logistic regression in Python*. Simplilearn.com. https://www.simplilearn.com/tutorials/machine-learning-tutorial/logistic-regression-in-python

[17] *Logistic Regression: equation, assumptions, types, and best practices*. (2022, April 18). Spiceworks. https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-logistic-regression/#:~:text=Practices%20for%202022-,What%20Is%20Logistic%20Regression%3F,1%2C%20or%20true%2Ffalse

[18] Hackeling, G. (2014). *Mastering Machine Learning with Scikit-Learn*.

[19] *TCGA Ethics and Policies*. (2019, March 6). Ccg - National Cancer Institute. https://www.cancer.gov/ccg/research/genome-sequencing/tcga/history/ethics-policies

[20] *Principles of Ethical conduct // compliance // UCI*. (n.d.-b). https://www.compliance.uci.edu/ethical-values.php

APPENDIX

APPENDIX A-DATASET TABLE AND LINK

Link: <https://archive.ics.uci.edu/dataset/759/glioma+grading+clinical+and+mutation+features+dataset>

| **Variable Name** | **Role** | **Description** |
| --- | --- | --- |
| Grade | Target | Glioma grade class information (0 = "LGG"; 1 = "GBM") |
| Gender | Feature | Gender (0 = "male"; 1 = "female") |
| Age\_at\_diagnosis | Feature | Age at diagnosis with the calculated number of days |
| Race | Feature | Race (0 = "white"; 1 = "black or african American"; 2 = "asian"; 3 = "american indian or alaska native") |
| IDH1 | Feature | isocitrate dehydrogenase (NADP(+))1 (0 = NOT\_MUTATED; 1= MUTATED) |
| TP53 | Feature | tumor protein p53 (0 = NOT\_MUTATED; 1 = MUTATED) |
| ATRX | Feature | ATRX chromatin remodeler (0 = NOT\_MUTATED; 1 = MUTATED) |
| PTEN | Feature | phosphatase and tensin homolog (0 = NOT\_MUTATED; 1 = MUTATED) |
| EGFR | Feature | epidermal growth factor receptor (0 = NOT\_MUTATED; 1 = MUTATED) |
| CIC | Feature | capicua transcriptional repressor (0 = NOT\_MUTATED; 1 = MUTATED) |
| MUC16 | Feature | mucin 16, cell surface associated (0 = NOT\_MUTATED; 1 = MUTATED) |
| PIK3CA | Feature | phosphatidylinositol-4,5-bisphosphate 3-kinase catalytic subunit alpha (0 = NOT\_MUTATED; 1 = MUTATED) |
| NF1 | Feature | neurofibromin 1 (0 = NOT\_MUTATED; 1 = MUTATED) |
| PIK3R1 | Feature | phosphoinositide-3-kinase regulatory subunit 1 (0 = NOT\_MUTATED; 1 = MUTATED) |
| FUBP1 | Feature | far upstream element binding protein 1 (0 = NOT\_MUTATED; 1 = MUTATED) |
| RB1 | Feature | RB transcriptional corepressor 1 (0 = NOT\_MUTATED; 1 = MUTATED) |
| NOTCH1 | Feature | notch receptor 1 (0 = NOT\_MUTATED; 1 = MUTATED) |
| BCOR | Feature | BCL6 corepressor (0 = NOT\_MUTATED; 1 = MUTATED) |
| CSMD3 | Feature | CUB and Sushi multiple domains 3 (0 = NOT\_MUTATED; 1 = MUTATED) |
| SMARCA4 | Feature | SWI/SNF related, matrix associated, actin dependent regulator of chromatin, subfamily a, member 4 (0 = NOT\_MUTATED; 1 = MUTATED) |
| GRIN2A | Feature | glutamate ionotropic receptor NMDA type subunit 2A (0 = NOT\_MUTATED; 1 = MUTATED) |
| IDH2 | Feature | isocitrate dehydrogenase (NADP(+)) 2 (0 = NOT\_MUTATED; 1 = MUTATED) |
| FAT4 | Feature | FAT atypical cadherin 4 (0 = NOT\_MUTATED; 1 = MUTATED) |
| PDGFRA | Feature | platelet-derived growth factor receptor alpha (0 = NOT\_MUTATED; 1 = MUTATED) |

Table 3. Description of the Dataset features and the feature roles

APPENDIX B-SCIENTIFIC LIBRARIES

A screenshot of a computer program

Description automatically generated

Figure 14. Scientific Libraries used in this project

APPENDIX C-DATA SPLITTING

A screenshot of a computer code

Description automatically generated

Figure 15. Code for data splitting.

APPENDIX D-LINK TO FULL EXPERIMENT CODE

<https://github.com/Uju024/-CLASSIFICATION-OF-BRAIN-GLIOMA-TYPES-IN-HUMANS-USING-MACHINE-LEARNING-ALGORITHMS->