Feature Selection Using L1 Regularization on Shallow Neural Networks for Glioma Grading from Molecular and Demographic Data

Juan Felipe Herrera Poveda

***Abstract*—Gliomas are a type of brain tumors that tend to affect the surrounding brain tissue and that can be classified depending on its aggressiveness. This classification play a crucial role on defining treatment plans for patients and provides a broader view on the current state, future steps and possible outcomes for the patients that are diagnosed with these types of tumors. One of the most popular methods for this classification process uses the molecular features from the medical histology of the tissues obtained from the resection process along with some demographic data from the patients. In this histological process, assessing the molecular state of the tumors can get expensive and reducing the amount of molecular features become a necessity. In this paper, a feature selection technique is proposed for finding an optimal subset with the minimum possible amount of molecular features, using shallow neural networks and L1 regularization on the TCGA dataset retrieved from the UCI Machine Learning Repository**

***Index Terms*—Feature selection, regularization, deep learning, gliomas, shallow neural networks, lasso feature selection, mRMR.**

I. Introduction

ONE of the crucial steps in designing an effective treatment plan for patients with any kind of tumor is determining its aggressiveness, usually through the usage of histological data. This information guides therapeutic decisions, minimizing unnecessary treatment toxicity, providing valuable insights into a patient's prognosis and hopefully resulting in a better outcome for patients.

This study focuses on a specific type of tumors called gliomas, the most prevalent brain and spinal cord tumors, for which there has been an extensive research regarding the best way of classifying them and arising in the utility of combining molecular and demographic data obtained during tumor resection and diagnosis [1].

Glioma grading typically categorizes tumors based on their aggressiveness. Glioblastoma multiforme (GBM) represents the most aggressive form, while low-grade gliomas (LGG) are considered one of the least aggressive. This process usually relies on using molecular data collection, which involves analyzing the expression levels of specific molecules within the tumor tissue. While valuable, this process can be time-consuming and expensive depending on the amount of molecular markers studied. Resulting in the necessity of minimizing the amount of molecular markers to study without losing too much information that would prevent a correct grading of the tumor. Therefore, the key objective of this study is to identify a minimal set of molecular markers while retaining sufficient accuracy for glioma grading into the two classes described above.

This problem has been approached before, primarily using machine learning and feature selection techniques not only for finding the optimal subset of features but also resulting in model proposals that use a small amount of molecular markers that yield in a good classification of gliomas. This approach has resulted in some new novel feature selection techniques and in public datasets like the one used in this work, available at the UC Irvine Machine Learning Repository, specifically, the one provided under the name *Glioma Grading Clinical and Mutation Features* [2] presented at the article that introduced this dataset [3].

In this paper, a process for feature selection is proposed, using regularization on shallow neural networks and a variation of forward selection technique in order to achieve an optimal set with the minimum amount of features and hence a minimum amount of molecular markers to be studied, resulting in an accuracy of around 84% with one feature and a maximum accuracy of 88% with eight features. The result of the experimentation process is compared to classical feature selection techniques such as LASSO, mRMR and with the GradWise feature selection method [4] proposed by some of the authors of the original dataset.

II. Method Description

The method proposed for the feature selection involves creating a shallow neural network with an architecture similar to the next one

- Dense layer with the same amount of units as the amount of features in the dataset, including a kernel L1 regularizer (i.e. a layer's weights L1 regularizer). This layer recieves the input.
- A hidden layer with a small amount of units connected to the first layer and the output layer describe in the next item.
- An output layer with one unit and a sigmoid activation function.

With the neural network created, the model is trained with the whole set of features and after the training process is done,

the weights of the first layer (the layer with the regularizer) are extracted. Since this weights were regularized with an L1 regularization, retrieving them provide an insight of the importance of each of the features on the model's performance, where the least important ones will have a value close to zero while the most important ones will have a higher value.

The L1 regularization involves adding a penalty function for the specific weights of the layers for whom the kernel regularizer is added, in general terms, the L1 regularization is added as a new term on the loss function, that will be computed as

$$\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{p} x_{ij}W_j\right)^2 + \lambda\sum_{j=1}^{p}|W_j| \tag{1}$$

where λ corresponds to the regularization term, a value set during the regularization definition in the model's layer and for whom the specific value for this work is shown in the experimentation section.

After the weights extraction process, the original set of features is sorted according to their respective importance retrieved from the layer's weights and then a modified forward feature selection technique is applied according to the importance order of the features. This means that with the same architecture described above, a first model is created and trained with the most important feature, then a second model is created and trained with the two most important features and so on until fifteen models are created and trained; with the first one being trained with the most important feature and the last one being trained with the fifteen most important features. The maximum number of features is set to fifteen deliberately so the value is not very close to 23 features, which is the total number of features in the dataset.

III. Experimental Process

A. Dataset details

The dataset chosen includes 23 features, including 20 molecular features and 3 demographical features. These features are described below

- IDH1: Isocitrate dehydrogenase (1: Mutated, 0: Not mutated).
- TP53: Tumor protein p53 (1: Mutated, 0: Not mutated).
- ATRX: ATRX chromatin remodeler (1: Mutated, 0: Not mutated).
- PTEN: Phosphatase and tensin homolog (1: Mutated, 0: Not mutated).
- EGFR: Epidermal growth factor receptor (1: Mutated, 0: Not mutated).
- CIC: Capicua transciptional repressor (1: Mutated, 0: Not mutated).
- MUC16: mucin 16, cell surface associated (1: Mutated, 0: Not mutated).
- PIK3CA: phosphatidylinositol-4,5-bisphosphate 3-kinase catalytic subunit alpha (1: Mutated, 0: Not mutated).
- NF1: neurofibromin 1 (1: Mutated, 0: Not mutated).
- PIK3R1: phosphoinositide-3-kinase regulatory subunit 1 (1: Mutated, 0: Not mutated).
- FUBP1: far upstream element binding protein (1: Mutated, 0: Not mutated).
- RB1: RB transcriptional corepressor 1 (1: Mutated, 0: Not mutated).
- NOTCH1: Notch receptor 1 (1: Mutated, 0: Not mutated).
- BCOR: BCL6 corepressor (1: Mutated, 0: Not mutated).
- CSMD3: CUB and Sushi multiple domains 3 (1: Mutated, 0: Not mutated).
- SMARCA4: SWI/SNF related, matrix associated, actin dependent regulator of chromatin, subfamily a, member 4 (1: Mutated, 0: Not mutated).
- GRIN2A: Glutamate ionotropic receptor NMDA type subunit 2A (1: Mutated, 0: Not mutated).
- IDH2: isocitrate dehydrogenase (1: Mutated, 0: Not mutated).
- FAT4: FAT atypical cadherin 4 (1: Mutated, 0: Not mutated).
- PDGFRA: platelet-derived growth factor receptor alpha (1: Mutated, 0: Not mutated).
- Gender: (0: male, 1: female)
- Age (Age): Age of the patient when diagnosed.
- Race: (0: white, 1: black or african american, 2: asian, 3: american indian or alaska native).

And that where each element is labeled as Low-Grade glioma (LGG) or Glioblastoma Multiforme (GBM) wit classes 0 and 1 respectively. The used dataset includes 839 examples and 23 features, each example with their corresponding label.

B. Experimental setup

Following the method described in the previous section, the shallow neural network is created according to the proposed architecture, specifically, for the general model (i.e. the model created for the feature importance extraction) the architecture is

1) First dense layer with 23 units, linear activation, kernel L1 regularizer with the L1 parameter set to 0.01 and an input shape tuple of (None, 23)
2) Hidden layer with 15 units, a relu activation and a L1 regularizer as well with the same parameter as the first layer

3) Output layer with one unit.

The model is then compiled with an Adam optimizer using a learning rate of 0.001, and a loss function corresponding to Binary Cross-Entropy. The model creation, compilation and further training is performed using Tensorflow library [5].

For the training process and for ensuring certain randomness and multiple experiments, a 10-Fold Cross-Validation is applied since the neural network is small enough so performing this cross-validation does not have a high cost. The original dataset is split into train and test datasets with 85% and 15% respectively. The further cross-validation split is performed on the training split while the test dataset remains the same for model evaluation after the CV process is done.

After the training process is completed, the average metrics through the K-Fold Cross-Validation are reported and the model with the best performance is saved in order to extract the final features importance.

Then, for the forward feature selection process, the models are created using the same architecture that the one used for the general model, except for the fact that the number of units in the first layer correspond to the number of features that are received by the neural network in each case. This feature set is initialized as an empty set and gets recomputed after the training of each model, as follows

$$S = S \cup \{X_i\} \, \forall i \in \{1, 2, ..., 15\} \tag{2}$$

where X corresponds to the set of features sorted by the feature importance extracted from the general model and where the increment of i is done once the model with the previous set S has finished. The architecture for each of the models trained with the set of features S is

1) First dense layer with $|S|$ units, linear activation, kernel L1 regularizer with the L1 parameter set to 0.01 and an input shape tuple of (None, $|S|$)

2) Hidden layer with 15 units, a relu activation and a L1 regularizer as well with the same parameter as the first layer

3) Output layer with one unit.

For the training process of these models, 10-Fold Cross-Validation is performed as well.

After the fifteen models have been successfully trained, the models' performances are reported using different metrics such as precision, recall, f1, accuracy and ROC AUC. Metrics described here for clarification purposes

$$\begin{aligned} Precision &= \frac{TruePositive}{TruePositive + FalsePositive} \\ Recall &= \frac{TruePositive}{TruePositive + FalseNegative} \\ F1 &= \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \\ Accuracy &= \frac{TP + TN}{TP + TN + FP + FN} \end{aligned} \tag{3}$$

And for computing the ROC Curve, two other metrics are needed, that correspond to True Positive Rate (TPR) and False Positive Rate (FPR). Then, the area under the curve (AUC) is computed as

$$\int_0^1 TPR(FPR^{-1}(x))dx \tag{4}$$

where

$$\begin{aligned} TPR &= \frac{TP}{TP + FN} \\ FPR &= \frac{FP}{TN + FP} \end{aligned} \tag{5}$$

The optimal set of features is then selected based on the highest accuracy among the models. Also, the minimum set is highlighted for showing the results obtained for that particular model (one feature).

Once the metrics are retrieved, and in order to perform some comparison between the proposed method and the ones proposed on the state of the art, three other methods are recreated for this purpose, specifically, the LASSO feature selection method that involves using a lasso regression and then extracting the coefficients for observing the importance of the features which is the same concept that the L1 regularization but with the last one being applied to the shallow neural network and the first one being used with a Lasso regression which for this case is performed using the LASSO model from scikit-learn [6]. The second comparison is with the minimum redundancy maximum relevance (mRMR) proposed in [7] where an algorithm capable of ranking the features based on relevance (correlation between the feature and the target) and redundancy (feature correlation with the selected features for previous iterations). This method is available using the python library *pymrmr* [8]. And last, one of the methods proposed by the authors of the dataset and the original paper, called GradWise [4] which corresponds to a rank-based feature selection algorithm that uses LASSO and mRMR feature selection techniques and a feature-weighting technique that for this specific recreation resulted in the same values as the ones described in the paper because of the general better performance of the LASSO feature selection technique over the mRMR which corresponds to increasing the count of each feature by two whenever it is selected by the LASSO feature

selection and by one when it is chosen by mRMR. The general process of this method is described in the steps below

1) For each feature, a variable is created initialized with a value of zero that will hold the times that the feature was chosen (weight).
2) K-Folds are generated (in the paper 5-Folds were used and for this work 10-Folds are used)
3) For each fold:
   a) LASSO feature selection is applied on the fold's data
   b) mRMR feature selection is applied on the fold's data
   c) For all the chosen features, their corresponding variable is incremented by two if it was chosen by LASSO and by one if it was chosen by mRMR (three if it was chosen by both).
4) The features are then sorted according to their weight. The maximum weight for the 5-Fold case is 15 (a feature was chosen by both methods in all five folds) and for the 10-Fold, the maximum weight is 30 (a feature was chosen by both methods in all ten folds).
5) Iteratively, from the maximum weight down to zero independent of the steps (a step of 5 is chosen for this work, i.e. {30, 25, ..., 5, 0}):
   a) Features that have an equal or greater weight that the one selected are chosen as the new feature subset.
   b) A model is trained with the chosen feature subset.
   c) Model performance is measured.

Once the metrics for the four methods have been gathered, the metrics obtained are shown.

IV. Results

A. Proposed Method

After the training process of the general model, the best model got the following metrics on the test dataset

Table I: Metrics of the general model

Metric	Value
Precision	0.8393
Recall	0.8868
F1	0.8624
Accuracy	0.8810
ROC AUC	0.8817

And the confusion matrix 1.

While for the learning process of the best model the loss and accuracy behave as shown in figure 2.

With the trained model, the weights are extracted from the first layer and then the weight for each feature is shown in the histogram at figure 3.

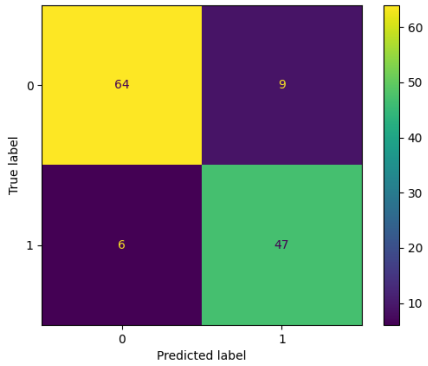

Figure 1: General model's confusion matrix

Figure 2: General model's loss and accuracy

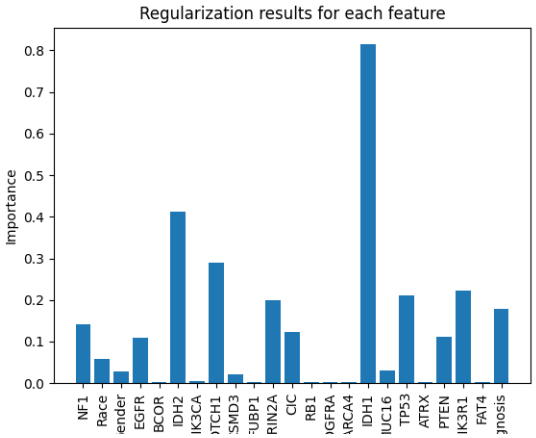

Figure 3: Proposed method feature importance

After this feature importance extraction, the models with the corresponding subsets are performed. The metrics on the test dataset are reported in table II. The metrics reported show that the best accuracy metric is achieved with 8 features, specifically with the feature set {NF1, IDH1, TP53, GRIN2A, NOTCH1, PIK3R1, IDH2, Age} and smaller features set not so far from that metric, for example, showing an accuracy of 84.1% with one feature {IDH1} and for a subset of 2 to 5 features an accuracy of 87.3% with the features shown for that number of features on the table. The metrics versus the feature set size is shown on figure 4.

Table II: Metrics for each feature subset from the proposed method

Number of Features	Features	Accuracy	Precision	Recall	F1
1	IDH1	0.8413	0.7620	**0.9057**	0.8276
2	IDH2, IDH1	0.8730	0.8136	0.9057	0.8571
3	NOTCH1, IDH2, IDH1	0.8730	0.8136	0.9057	0.8571
4	PIK3R1, NOTCH1, IDH2, IDH1	0.8730	0.8136	0.9057	0.8571
5	TP53, PIK3R1, NOTCH1, IDH2, IDH1	0.8730	0.8136	0.9057	0.8571
6	GRIN2A, TP53, PIK3R1, NOTCH1, IDH2, IDH1	0.8730	0.8136	0.9057	0.8571
7	Age, GRIN2A, TP53, PIK3R1, NOTCH1, IDH2, IDH1	0.8730	0.8421	0.9057	0.8727
8	NF1, IDH1, TP53, GRIN2A, NOTCH1, PIK3R1, IDH2, Age	**0.8810**	0.8545	0.8868	0.8704
9	CIC, NF1, IDH1, TP53, GRIN2A, NOTCH1, PIK3R1, IDH2, Age	0.8730	0.8571	0.9057	**0.8807**
10	PTEN, CIC, NF1, IDH1, TP53, GRIN2A, NOTCH1, PIK3R1, IDH2, Age	0.8810	0.8363	0.8680	0.8518
11	EGFR, PTEN, CIC, NF1, IDH1, TP53, GRIN2A, NOTCH1, PIK3R1, IDH2, Age	0.8651	**0.8704**	0.8868	0.8785
12	Race, EGFR, PTEN, CIC, NF1, IDH1, TP53, GRIN2A, NOTCH1, PIK3R1, IDH2, Age	0.8571	0.8545	0.8868	0.8704
13	MUC16, Race, CIC, EGFR, PTEN, NF1, IDH1, TP53, GRIN2A, NOTCH1, PIK3R1, IDH2, Age	0.8650	0.8421	0.9057	0.8727
14	Gender, MUC16, Race, CIC, EGFR, PTEN, NF1, IDH1, TP53, GRIN2A, NOTCH1, PIK3R1, IDH2, Age	0.8571	0.8393	0.8868	0.8624
15	CSMD3, Gender, MUC16, Race, CIC, EGFR, PTEN, NF1, IDH1, TP53, GRIN2A, NOTCH1, PIK3R1, IDH2, Age	0.8571	0.8393	0.8868	0.8624

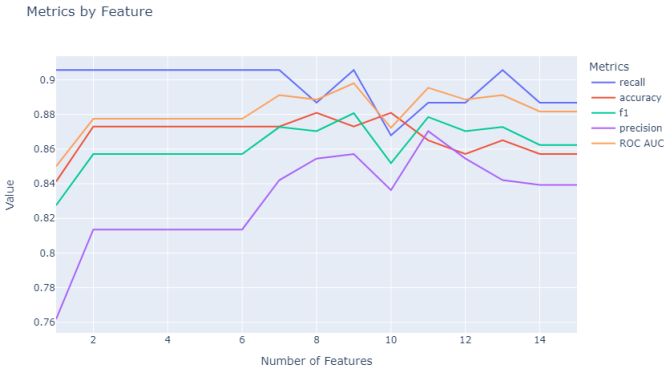

Figure 4: Metrics vs. feature set size

B. *LASSO Feature Selection*

Using the LASSO model from scikit-learn and a K-Fold CV with K=10, the features' importance are shown in the histogram below for which the fifteen most important features

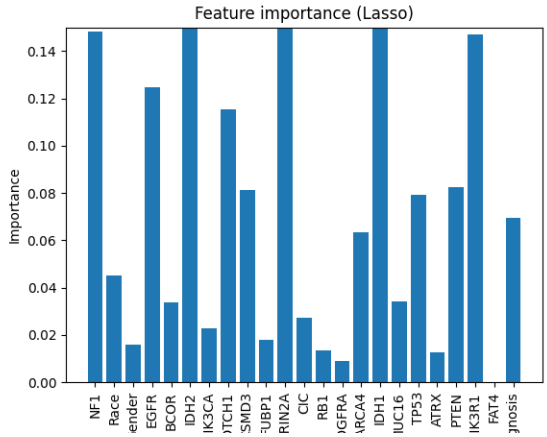

Figure 5: Lasso feature importance

were chosen as the feature subset for training a new shallow neural network, achieving the metrics shown in table III, for which an accuracy of 81.75% was achieved with the feature set {IDH1, IDH2, GRIN2A, NF1, PIK3R1, EGFR, NOTCH1, PTEN, CSMD3, TP53, Age, SMARCA4, Race, MUC16, BCOR}, a set that is similar to the one obtained for the 15 features subset of the proposed method, except for the feature {CIC} present in the proposed method set but not in the LASSO set. Instead, the {BCOR} feature is present. And since the neural network architecture is very similar then this different feature might be causing the almost 4 percentage points difference.

Table III: Metrics for Lasso features subset

Metric	Value
Precision	0.7188
Recall	0.9020
F1	0.8000
Accuracy	0.8175
ROC AUC	0.8310

The confusion matrix obtained for the Lasso model is shown in figure 7 for the model evaluation over the test dataset

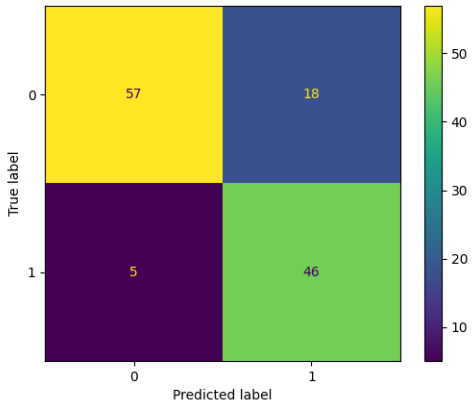

Figure 6: Lasso model confusion matrix

C. *mRMR Feature Selection*

For the feature selection using mRMR, a process similar to that one proposed in GradWise was followed, for each fold, 15 features were retrieved using the mRMR feature selection and at the end of the 10 folds, the fifteenth ones that were chosen the most times are the final subset of features.

This process resulted in the subset of features {Age, PIK3R1, PIK3CA, PDGFRA, NF1, IDH2, MUC16, FAT4, CSMD3, BCOR, SMARCA4, Race, RB1, NOTCH1, GRIN2A } for which a new neural network was built and training was applied also with 10-Fold CV, which for the best model resulted in the following metrics

Table IV: Metrics for mRMR features subset

Metric	Value
Precision	0.7500
Recall	0.7637
F1	0.7568
Accuracy	0.7857
ROC AUC	0.7832

and the confusion matrix

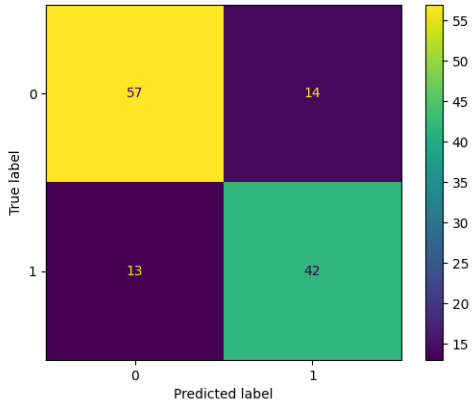

Figure 7: mRMR model confusion matrix

for which can be concluded that the performance was not bad but had a lower performance that the one obtained for the proposed method and the LASSO method.

D. GradWise

In order to perform the GradWise process, as described in the paper, it is necessary to verify between the two feature selection methods to be applied (Lasso and mRMR in this case as well as in the paper) which of the two has a higher performance (measured by accuracy) in order to assign the greatest increment (a value of 2). In this case, a similar result was achieved with the one achieved in the original paper, resulting in a higher accuracy in the Lasso model hence assigning an increment of 2 to every feature selected by Lasso and an increment of only 1 for those selected by the mRMR method.

Once the corresponding increment values were defined for each method, the process was applied as described before, resulting in the weights for each feature shown in table V

And when iterating through the minimum weights ({30, 25, 20, 15, 10, 5, 0}) the metrics were obtained and are shown in table VI. Showing the highest accuracy with 14, 20 and 21 features, while for the smallest set of features (4 features) the accuracy is around 79%.

The plot between the metrics and minimum weight chosen is shown on figure (Lower weight means higher features set's size) 8

Table V: Weights for each feature using GradWise

Feature	Weight
PTEN	30
Age	30
IDH2	30
PIK3R1	30
RB1	29
NF1	28
SMARCA4	27
GRIN2A	26
FAT4	24
EGFR	24
NOTCH1	24
MUC16	20
TP53	20
IDH1	20
CSMD3	18
PDGFRA	15
ATRX	14
Race	14
BCOR	12
CIC	10
PIK3CA	5
FUBP1	0
Gender	0

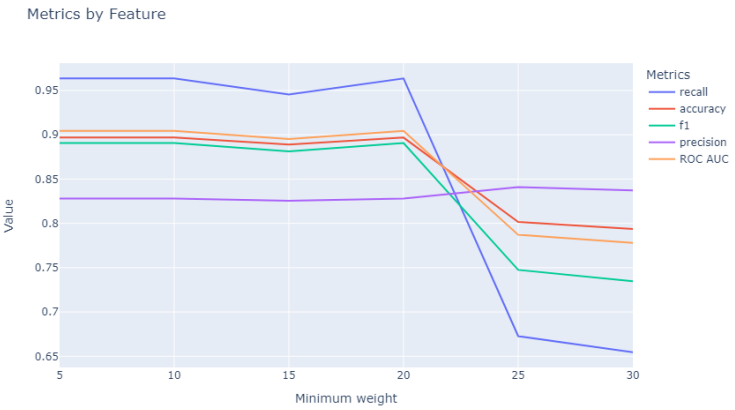

Figure 8: Metrics vs. weight

V. Conclusions

Between the different comparisons made among the different features selection techniques it can be seen that with the proposed method an accuracy of around 87% was achieved using only 2 to 5 features reaching a peak of 88% with 8 features and that with a single feature (IDH1) an accuracy of around 84% was obtained, this specific molecular marker is usually responsible of the most aggressive mutations of gliomas and is very well known that this specific marker is probably one of the most important when it comes to watching gliomas' aggressiveness, to the point where some novel treatments include medications that specifically block tumors growth for those tumors that present IDH1 or IDH2 mutations so the results obtained in this work show that studying the mutations of these two molecules provide a huge impact on the classification between GBMs and LGGs.

The proposed method has shown a better performance on smaller features subsets compared to the ones obtained with

Table VI: Metrics for each minimum weight, GradWise

Min weight	Number of features	Features	Precision	Recall	F1	Accuracy
30	4	PTEN,Age,IDH2,PIK3R1	0.8372	0.6545	0.7347	0.7937
25	8	PTEN,Age,IDH2,PIK3R1,RB1,NF1,SMARCA4,GRIN2A	0.8409	0.6727	0.7474	0.8015
20	14	PTEN,Age,IDH2,PIK3R1,RB1,NF1,SMARCA4,GRIN2A,FAT4,EGFR,NOTCH1,MUC16,TP53,IDH1	0.8281	0.9636	0.8907	0.8968
15	16	PTEN,Age,IDH2,PIK3R1,RB1,NF1,SMARCA4,GRIN2A,FAT4,EGFR,NOTCH1,MUC16,TP53,IDH1,CSMD3,PDGFRA	0.8254	0.9454	0.8813	0.8888
10	20	PTEN,Age,IDH2,PIK3R1,RB1,NF1,SMARCA4,GRIN2A,FAT4,EGFR,NOTCH1,MUC16,TP53,IDH1,CSMD3,PDGFRA,ATRX,Race,BCOR,CIC	0.8281	0.9636	0.8908	0.8968
5	21	PTEN,Age,IDH2,PIK3R1,RB1,NF1,SMARCA4,GRIN2A,FAT4,EGFR,NOTCH1,MUC16,TP53,IDH1,CSMD3,PDGFRA,ATRX,Race,BCOR,CIC,PIK3CA	0.8281	0.9636	0.8907	0.8968

the other methods while the highest accuracy of around 89% was obtained with the GradWise method using 14, 20 and 21 features.

In the same way, it is evident that for the proposed, LASSO and GradWise methods, the selected features sets were somehow similar between them except for certain features that depending on the technique were considered way more important than others. This can provide an insight of a new feature selection technique for further works that include the methods here discussed. Also, since IDH1 and IDH2 were the most common molecular markers selected as the most important features, further work on optimal features subsets that do not include these two molecules can provide an insight on how the other molecules impact the classification process.

REFERENCES

[1] H. Jiang, Y. Cui, J. Wang, and S. Lin, "Impact of epidemiological characteristics of supratentorial gliomas in adults brought about by the 2016 world health organization classification of tumors of the central nervous system," *Oncotarget*, vol. 8, no. 12, pp. 20 354–20 361, 2017. [Online]. Available: https://www.oncotarget.com/article/13555/

[2] E. Tasci, K. Camphausen, A. V. Krauze, and Y. Zhuge, "Glioma Grading Clinical and Mutation Features," UCI Machine Learning Repository, 2022, DOI: https://doi.org/10.24432/C5R62J.

[3] E. Tasci, Y. Zhuge, H. Kaur, K. Camphausen, and A. V. Krauze, "Hierarchical voting-based feature selection and ensemble learning model scheme for glioma grading with clinical and molecular characteristics," *International Journal of Molecular Sciences*, vol. 23, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:253653992

[4] E. Tasci, S. Jagasia, Y. Zhuge, K. Camphausen, and A. V. Krauze, "Gradwise: A novel application of a rank-based weighted hybrid filter and embedded feature selection method for glioma grading with clinical and molecular characteristics," *Cancers*, vol. 15, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:262190708

[5] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[6] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[7] C. Ding and H. Peng, "Minimum redundancy feature selection from microarray gene expression data," *Journal of bioinformatics and computational biology*, vol. 3, no. 02, pp. 185–205, 2005.

[8] F. G. Brundu, "Pymrmr: Python3 binding to mrmr feature selection algorithm," 2017. [Online]. Available: https://pypi.org/project/pymrmr/