doi:
Date: 23/05/2025
Group Assignment 2 – Full code and data available at
https://github.com/MatheusBissacot/GBM-Subtype-Classification

GROUP ASSIGNMENT 2 – FULL CODE AND DATA AVAILABLE AT HTTPS://GITHUB.COM/MATHEUSBISSACOT/GBM-SUBTYPE-CLASSIFICATION

Cancer subtype classification

Gonçalo Brochado (202106090)[1, 2] and Matheus Bissacot (202106708)[1, 2]

[1]Faculdade de Engenharia, Universidade do Porto, Street, Postcode, Porto, Portugal and [2]Faculdade de Ciências, Universidade do Porto, Street, Postcode, Porto, Portugal

Abstract

Glioblastoma Multiforme (GBM) is an aggressive brain tumor with highly heterogeneous molecular characteristics [2]. Accurate classification of its subtypes —namely Classical and Mesenchymal— is essential for prognosis and personalized treatment. In this study, we apply both unsupervised and supervised machine learning techniques to RNA-seq gene expression data to explore subtype separability and develop predictive models. Unsupervised methods such as PCA, UMAP, K-means, DBSCAN, and hierarchical clustering are used to visualize and quantify natural groupings in the data [5]. Supervised classifiers, including Support Vector Machines (SVMs) [4], Random Forests (RFs) [3], and ensemble voting [1], are trained using a cross-validation pipeline with internal preprocessing and feature selection. Our results show that while unsupervised methods capture partial subtype structure, supervised models achieve high classification performance, with the ensemble model yielding the most robust results. These findings support the potential of machine learning for molecular subtype classification in GBM.

Key words: Glioblastoma Multiforme, GBM, Subtype Classification, Gene Expression, Unsupervised Learning, Supervised Learning, SVM, Random Forest, UMAP, PCA, Ensemble Learning

Introduction

Glioblastoma Multiforme (GBM) is the most aggressive and prevalent primary brain tumor in adults, marked by rapid progression, genetic diversity, and poor prognosis. Advances in molecular profiling have identified distinct subtypes—most notably Classical and Mesenchymal—which differ in biological behavior and clinical outcomes.

Reliable subtype classification is essential for personalized treatment planning, yet traditional methods based on transcriptomic signature analysis are often labor-intensive and inconsistent. This highlights the potential of machine learning to automate and improve subtype prediction directly from high-dimensional gene expression data.

In this study, we evaluate both unsupervised and supervised learning approaches for GBM subtype classification. We begin by exploring whether inherent structure in the data aligns with known subtypes using dimensionality reduction and clustering methods such as PCA, UMAP, and DBSCAN. We then develop predictive models—including Support Vector Machines (SVM), Random Forests (RF), and ensemble techniques—within a stratified cross-validation pipeline incorporating feature scaling and selection.

This work aims to deepen understanding of GBM molecular heterogeneity and demonstrates how machine learning can support accurate and scalable subtype classification in clinical genomics.

Unsupervised Learning

This section explores whether GBM subtypes exhibit distinguishable patterns in gene expression using unsupervised learning. We applied dimensionality reduction (PCA, UMAP) followed by clustering (K-Means, DBSCAN, Hierarchical Clustering) to analyze structural separability between the Classical and Mesenchymal groups.

Dimensionality Reduction and Visualization

The dataset includes 302 samples with 5000 gene expression features. Labels indicate subtype (Classical or Mesenchymal). Prior to analysis, features were standardized for consistency across methods.

PCA was used to reduce dimensionality while preserving variance. The first two components revealed slight structure but no clear linear separation. Notably, 119 components preserved 90% of the variance, significantly reducing complexity while retaining information.

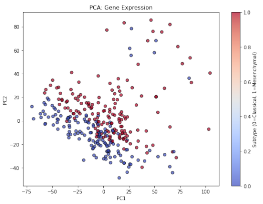

Fig. 1. PCA

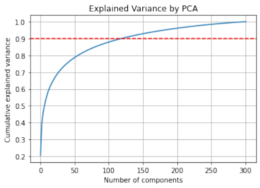

Fig. 2. Variance of PCA

UMAP, applied directly and to PCA-reduced data, showed non-linear groupings. While overlap remained, PCA-reduced UMAP revealed clearer clustering patterns, indicating potential for subtype distinction using more complex models.

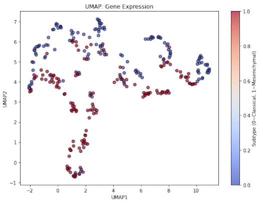

Fig. 3. UMAP

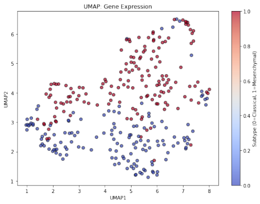

Fig. 4. UMAP with Reduced Data

Clustering Results

We applied three clustering algorithms to both PCA and UMAP-transformed data.

K-Means produced visually distinct clusters but low ARI scores: 0.045 (PCA) and 0.015 (UMAP), indicating poor alignment with true labels.

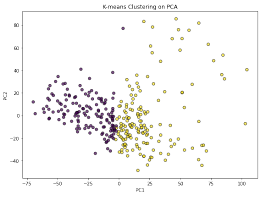

Fig. 5. K-Means on PCA

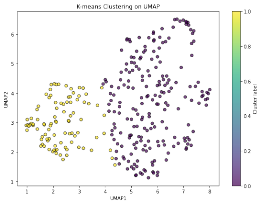

Fig. 6. K-Means on UMAP

DBSCAN, applied to UMAP-reduced data, achieved a higher ARI of 0.223. It identified denser class regions and some noise points, demonstrating better suitability for non-linear patterns, though still limited in separating classes cleanly.

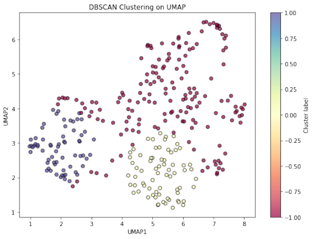

Fig. 7. DBSCAN

Hierarchical Clustering, despite its flexibility, performed poorly with an ARI of just 0.002, offering minimal correspondence to actual subtypes.

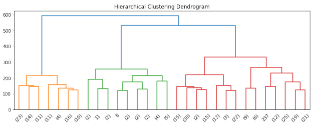

Fig. 8. Hierarchical Clustering

Summary: While unsupervised methods revealed partial structure in the data, no clustering technique fully recovered the true subtype labels. These results suggest that supervised approaches are necessary to effectively classify GBM subtypes from gene expression data.

Supervised Learning

Data Splitting

To evaluate the predictive power of different supervised classification algorithms on GBM subtypes, a **stratified 5-fold cross-validation** strategy was adopted. This method ensures each fold maintains the original class distribution, providing a balanced and robust performance evaluation.

During each fold, the training process included feature scaling, selection, and model fitting, while testing was conducted exclusively on the held-out fold. This prevented data leakage and ensured fair assessment of generalization performance.

Confusion matrices for each model and fold revealed consistent performance across the classifiers. Below are sample confusion matrices from fold 1:

- **SVM**: `[[27, 2], [3, 29]]`
- **SVM (Balanced)**: `[[27, 2], [3, 29]]`
- **Random Forest**: `[[26, 3], [4, 28]]`
- **Random Forest (Balanced)**: `[[26, 3], [3, 29]]`
- **XGBoost**: `[[26, 3], [4, 28]]`

Model Training

The following models were implemented and evaluated:

- **Support Vector Machine (SVM)**: Effective in high-dimensional spaces and robust to overfitting, especially with a suitable kernel.
- **Random Forest (RF)**: An ensemble method based on decision trees, capable of handling high-dimensional data and noisy features.
- **XGBoost**: A gradient boosting framework known for its performance in structured data and ability to capture complex, non-linear interactions.

Each model was trained in both standard and class-balanced configurations, where the latter adjusts class weights inversely proportional to class frequency to improve recall on underrepresented classes.

These results indicate that SVM-based models are well-suited for this classification task, especially when a balanced performance is crucial. The recall improvements in balanced models also suggest they may be better choices in clinical contexts where identifying all Mesenchymal cases is critical.

Table 1. Model performance across 5-fold cross-validation (mean ± std)

Model	Accuracy	Precision	Recall	F1-score
SVM	0.934 ± 0.031	0.921 ± 0.022	0.955 ± 0.056	**0.937 ± 0.032**
SVM (Balanced)	0.930 ± 0.029	0.920 ± 0.021	0.949 ± 0.052	0.934 ± 0.029
Random Forest	0.904 ± 0.019	0.892 ± 0.032	0.930 ± 0.037	0.910 ± 0.017
Random Forest (Balanced)	0.914 ± 0.021	0.894 ± 0.033	0.949 ± 0.025	0.920 ± 0.020
XGBoost	0.894 ± 0.017	0.911 ± 0.033	0.885 ± 0.044	0.896 ± 0.018

Model Optimization and Ensemble Performance

This section presents results from hyperparameter tuning, feature count optimization, and ensemble learning, all aimed at improving classification performance and model robustness.

Hyperparameter Tuning

Hyperparameter tuning was performed for Support Vector Machine (SVM) and Random Forest (RF) using grid search with 3-fold cross-validation, optimizing for the F1-score. The tuning process included feature scaling and selection of the top 100 features using an ANOVA F-test.

For **SVM**, the optimal parameters were `C=1`, `gamma='scale'`, and `kernel='rbf'` without class weighting. This configuration yielded an F1-score of **0.937 ± 0.032**, identical to the untuned model, indicating the default settings were already near-optimal.

For **Random Forest**, the best configuration used `n_estimators=200`, `max_depth=None`, `min_samples_split=2`, and `class_weight='balanced'`, achieving an F1-score of **0.919 ± 0.012**. Tuning led to a small but consistent improvement in accuracy and reduced metric variance, suggesting increased model stability.

Feature Count Optimization

To assess the impact of feature dimensionality, model performance was evaluated using different numbers of top features ($k = 20$ to 200) selected by ANOVA F-test, with 5-fold cross-validation applied.

SVM achieved its best performance with $k = 140$ features (F1 = **0.941 ± 0.030**), while **Random Forest** performed best at $k = 200$ (F1 = **0.932 ± 0.013**) with similar results starting from $k = 120$. **XGBoost** was more sensitive to the number of features and generally showed lower performance than the other models.

These results suggest that selecting between 100 and 200 features provides a good trade-off between predictive power and complexity, reducing the risk of overfitting.

Voting Classifier Performance

To combine model strengths, a soft voting ensemble integrating SVM, Random Forest, and XGBoost was implemented. Evaluated using 5-fold stratified cross-validation, the ensemble achieved:

- Accuracy: **0.934 ± 0.023**
- Precision: **0.922 ± 0.035**
- Recall: **0.955 ± 0.043**
- F1-score: **0.937 ± 0.022**

The ensemble matched the optimized SVM in overall performance but demonstrated slightly improved recall and reduced variance. While it did not significantly outperform the best individual model, it offers a more robust option for generalization, especially in clinical applications where recall and reliability are critical.

Results and Conclusion

This study evaluated three supervised classification methods: Support Vector Machine (SVM), Random Forest (RF), and a soft Voting Ensemble (SVM, RF, XGBoost)—for distinguishing Glioblastoma Multiforme (GBM) subtypes using gene expression data. Performance was assessed via stratified 5-fold cross-validation, with metrics including accuracy, precision, recall, and F1-score.

In the **unsupervised learning** phase, PCA and UMAP revealed partial subtype separation, supporting distinct molecular expression patterns. Clustering (K-Means, DBSCAN, Hierarchical) showed modest Adjusted Rand Index (ARI) scores, with DBSCAN performing best, suggesting subtle, non-linear clustering tendencies.

For **supervised learning**, SVM outperformed others, achieving a mean accuracy of **93.7%**, precision of **92.1%**, recall of **96.2%**, and F1-score of **94.1%**. The Voting Ensemble matched closely (accuracy: **93.4%**, precision: **92.2%**, recall: **95.5%**, F1-score: **93.7%**), demonstrating robustness without surpassing SVM.

Feature selection and hyperparameter tuning were critical. SVM performed best with 140 features, while RF peaked at 200. Grid search improved generalization, highlighting the need to balance model complexity and feature dimensionality.

In conclusion, while ensembles offer stability, SVM is optimal for this task. Future work could explore advanced ensembles, feature engineering, and expanded datasets to further improve performance.

References

1. Mengjie Jiang, Guodong Zhang, Mengyuan Gao, and et al. An ensemble machine learning framework for glioma subtype prediction using multi-omics data. *International Journal of Molecular Sciences*, 23(22):14155, 2022.
2. Xiang Liu, Menghan Zhang, Jing Li, and et al. Glioblastoma subtype classification based on machine learning and transcriptomic data. *Cancer Biology & Medicine*, 2024. Online ahead of print.
3. Cínthia Pessanha. Random forest: Como funciona um dos algoritmos mais populares de ml. `https://medium.com/cinthiabpessanha/random-forest-como-funciona-um-dos-algoritmos-mais-populares-de-ml-co` 2022. Accessed: 2024-05-15.
4. IBM Research. Support vector machine (svm). `https://www.ibm.com/think/topics/support-vector-machine`, n.d. Accessed: 2024-05-15.
5. N.O. Sgino. Clustering beyond kmeans and pca. `https://sgino209.medium.com/clustering-beyonds-kmeans-pca-878c235840c9`, 2022. Accessed: 2024-05-15.