Module 3

Here are five different machine learning analyses that we can perform on the dataset from `processed\_clinical\_data\_with\_idh\_status`:

1. Unsupervised Clustering of Molecular Subtypes

- Goal: Discover molecular subtypes of the cancer based on mutation patterns and clinical features.

- Features to Use: `Hugo\_Symbol` (one-hot encoded for multiple mutations), `Pathway`, `IDH Status`, `WHO Grade`, `Enhancing`, `Sample Type`, `TMB`, `Fraction Genome Altered`.

- Approach:Use algorithms like K-Means, Hierarchical Clustering, or DBSCAN for clustering. Apply dimensionality reduction techniques like PCA to visualize the clusters.

-Evaluation:Examine how clusters correlate with clinical features such as survival outcomes, IDH status, or WHO Grade.

### 2. \*\*Survival Prediction Using Cox Proportional Hazards Model\*\*

- \*\*Goal:\*\* Predict patient survival using clinical and molecular features.

- \*\*Features to Use:\*\* `Pathway`, `IDH Status`, `WHO Grade`, `Enhancing`, `TMB`, `Fraction Genome Altered`, and optionally encoded mutations (`Hugo\_Symbol`).

- \*\*Approach:\*\* Use Cox Proportional Hazards regression to model the relationship between the features and overall survival.

- \*\*Evaluation:\*\* Assess the model's predictive ability using concordance index (C-index) and test for the proportional hazards assumption.

### 3. \*\*Classification of IDH Mutation Status\*\*

- \*\*Goal:\*\* Predict whether a patient has an IDH mutation based on other features.

- \*\*Features to Use:\*\* `Pathway`, `WHO Grade`, `Enhancing`, `TMB`, `Fraction Genome Altered`, and clinical data.

- \*\*Approach:\*\* Apply classification algorithms such as Random Forest, Gradient Boosting, or Support Vector Machines.

- \*\*Evaluation:\*\* Use metrics like accuracy, precision, recall, and ROC-AUC for model performance evaluation.

### 4. \*\*Predicting Recurrence Using Machine Learning\*\*

- \*\*Goal:\*\* Predict if a patient sample is classified as primary or recurrence based on molecular and clinical data.

- \*\*Features to Use:\*\* `Hugo\_Symbol` (encoded), `Pathway`, `IDH Status`, `WHO Grade`, `TMB`, `Fraction Genome Altered`, and `Enhancing`.

- \*\*Approach:\*\* Use classification algorithms such as Logistic Regression, XGBoost, or Neural Networks.

- \*\*Evaluation:\*\* Assess model performance using confusion matrices, ROC-AUC, and cross-validation techniques.

### 5. \*\*Feature Importance Analysis for Prognostic Factors\*\*

- \*\*Goal:\*\* Identify features that are most associated with overall survival or cancer subtypes.

- \*\*Features to Use:\*\* All available features.

- \*\*Approach:\*\* Apply techniques like Lasso Regression, Random Forest Feature Importance, or SHAP values to rank features by importance.

- \*\*Evaluation:\*\* Compare the top features with known prognostic markers and assess their relevance for patient outcomes.

These analyses cover both supervised (classification, regression) and unsupervised (clustering) approaches, providing a broad spectrum of insights from the dataset.

Here's the prompt for clustering analysis on primary tumor samples using the specified columns, while accounting for the presence of multiple mutations for each patient:

### Clustering Strategy

1. \*\*Data Preprocessing:\*\*

- \*\*Filter Primary Samples:\*\* Select only rows corresponding to primary tumor samples.

- \*\*Group Data by Patient ID:\*\* Since each patient may have multiple mutations, aggregate information while retaining the list of unique mutated genes for each patient.

- \*\*Handling Hugo\_Symbol:\*\* Convert the list of mutated genes for each patient into a binary representation (presence or absence of each gene) using one-hot encoding.

- \*\*Encode Categorical Variables:\*\* Convert categorical features (`Pathway`, `Enhancing`, `WHO Grade`, `IDH Status`) into numerical format using one-hot encoding or ordinal encoding as appropriate.

- \*\*Normalization:\*\* Scale numerical features (`Diagnosis Age`, `TMB (nonsynonymous)`, `Fraction Genome Altered`, etc.) to a standard range (e.g., 0-1 or z-score normalization).

- \*\*Handle Missing Data:\*\* Apply appropriate imputation techniques (e.g., mean/mode/median imputation) or drop rows/columns with excessive missing values.

2. \*\*Dimensionality Reduction:\*\*

- \*\*Apply PCA (Principal Component Analysis)\*\*: Use PCA to reduce the dimensionality of the data while retaining a significant portion of variance (e.g., 95%). This step helps to deal with the high-dimensional nature of the dataset, especially after one-hot encoding the genes.

3. \*\*Clustering:\*\*

- \*\*Algorithm Choice:\*\* Use K-Means, Agglomerative Clustering, or DBSCAN. Start with K-Means for initial clustering.

- \*\*Number of Clusters:\*\* Determine an optimal number of clusters (k) using techniques like the Elbow Method or Silhouette Score analysis.

4. \*\*Performance Evaluation:\*\*

- \*\*Cluster Quality Metrics:\*\* Calculate the Silhouette Score to evaluate how well-defined the clusters are.

- \*\*Cluster Distribution:\*\* Check if clusters differ in meaningful clinical or biological characteristics, such as WHO Grade or IDH Status distribution.

- \*\*Biological Validation:\*\* Compare clusters with known glioma subtypes or clinical outcomes to see if clusters reflect meaningful biological differences.

5. \*\*Post-Clustering Analysis:\*\*

- \*\*Association Testing:\*\* Use statistical tests to determine if certain clinical features or mutations are significantly enriched in specific clusters.

- \*\*Visualization:\*\* Plot the clusters using a 2D representation (e.g., PCA scatter plot or t-SNE) to assess the separation visually.

This strategy will help identify distinct molecular subtypes of glioma based on the chosen features while accounting for multiple mutations per patient.

After clustering  
Since you like the clustering results for both 4 and 10 clusters, the next steps will involve validating and interpreting these clusters to understand their biological or clinical significance. Here's a general plan:

**Step 1: Characterize the Clusters**

1. **Cluster Composition Analysis**: Examine the distribution of different features (e.g., IDH Status, WHO Grade, Pathway) across clusters to understand what distinguishes each cluster. You can use box plots, bar charts, or statistical tests to compare feature distributions across clusters.
2. **Summary Statistics**: Calculate the mean, median, or other statistics for continuous features within each cluster. This can help highlight differences in features like 'Diagnosis Age,' 'TMB (nonsynonymous),' or 'Fraction Genome Altered.'

**Step 2: Survival Analysis (Optional)**

If you have the 'Overall Survival (Months)' and 'Overall Survival Status' columns available, you can perform survival analysis to determine if certain clusters are associated with better or worse survival outcomes.

* **Kaplan-Meier Curves**: Generate Kaplan-Meier survival curves for each cluster to compare survival outcomes.
* **Log-Rank Test**: Perform a statistical test to check if survival distributions differ significantly across clusters.

**Step 3: Feature Importance Analysis**

1. **SHAP or Feature Importance with Tree-Based Models**: You can use a tree-based model (like Random Forest or Gradient Boosting) to identify which features are most important in distinguishing between clusters.
2. **Dimensional Contribution Analysis**: For the 30 PCA components, investigate which original features contributed most to the significant principal components used for clustering.

**Step 4: Biological Interpretation**

1. **Pathway Analysis**: Look at the most common pathways represented in different clusters to see if there are distinct biological processes driving the clustering.
2. **Literature Review**: For the key features that define each cluster, consider searching the literature to find if these features have been previously associated with specific tumor behaviors or patient outcomes.

**Step 5: Validation and Robustness Check**

1. **Stability of Clusters**: Test the stability of the clusters by running the clustering algorithm multiple times with different initializations or using different clustering methods (e.g., hierarchical clustering, DBSCAN).
2. **Cross-Validation**: If you want to ensure the clusters are not artifacts, consider splitting the data into subsets and performing clustering separately to see if similar patterns emerge.

Let me know which of these steps you'd like to proceed with, and I can provide the code and instructions accordingly.

291024  
**Prompt for Survival Prediction Ensemble Modeling**

**Task Overview**

The goal is to predict Overall Survival Status using an ensemble approach inspired by the glioma grading task in the paper, with a focus on accuracy and stability. We’ll apply individual models (Random Forest, SVM, KNN, and Gradient Boosting) and evaluate them with 10-fold cross-validation. Finally, we’ll explore combining these models into a voting ensemble to improve stability and predictive power.

**Detailed Steps and Control Points**

1. **Data Preparation**:
   * **Define Features and Target**:
     + Use the prediction DataFrame, retaining only the selected features (final\_selected\_features) and Patient ID.
     + Set Overall Survival Status as the target variable (y).
   * **QC Step**: Ensure there are no NaN values in the target variable.
2. **10-Fold Cross-Validation on Individual Models**:
   * **Random Forest**:
     + Run 10-fold cross-validation on Random Forest and record the accuracy for each fold, mean accuracy, and standard deviation.
   * **Support Vector Machine (SVM)**:
     + Repeat the process with SVM, ensuring it supports probabilistic output if needed for ensemble voting.
   * **K-Nearest Neighbors (KNN)**:
     + Evaluate KNN with cross-validation and record metrics.
   * **Gradient Boosting**:
     + Run cross-validation with Gradient Boosting and capture performance metrics.
   * **Control Step**: For each model, ensure that all cross-validation scores, mean accuracy, and standard deviation are recorded.
3. **Voting Ensemble Construction**:
   * **Ensemble Setup**:
     + Combine Random Forest, SVM, KNN, and Gradient Boosting using either:
       - **Hard Voting** (majority vote for each prediction) or
       - **Soft Voting** (average predicted probabilities across models, if supported).
   * **Cross-Validation on Ensemble**:
     + Run 10-fold cross-validation on the ensemble and capture mean accuracy and stability.
   * **Control Step**: Ensure ensemble cross-validation metrics are consistent and comparable to individual models.
4. **Evaluate and Compare Models**:
   * **Summary of Results**:
     + Compare individual models and ensemble models based on mean accuracy and standard deviation across all folds.
   * **Control Step**: Identify which approach provides the best trade-off between accuracy and stability.

Let me know if you need further specifications or if there’s anything to adjust based on your preferences.