Classification of Mice Based on Protein Expression Levels

Major Project Report

Objective -

- **Classify Mice Based on Protein Expression:** Develop a machine learning model to accurately classify mice into one of the eight classes based on the expression levels of 77 proteins. These classes are determined by a combination of genotype (control or trisomic), behavior (stimulated to learn or not), and treatment (saline or memantine).
- **Identify Key Discriminant Proteins:** Utilize feature selection techniques to identify which proteins or protein modifications are most important for distinguishing between the different classes. Understanding which proteins are key discriminators can provide insights into the biological mechanisms underlying learning and memory in Down syndrome.
- **Evaluate the Impact of Genotype, Behavior, and Treatment:** Analyze the effect of genotype (control vs. trisomic), behavior (context-shock vs. shock-context), and treatment (saline vs. memantine) on protein expression levels. This includes evaluating how these factors influence associative learning and the potential therapeutic effects of memantine in trisomic mice.

Methodology -

The methodology for analyzing protein expression data related to Down syndrome and associative learning involves several key steps. First, data preprocessing is conducted to handle missing values through imputation techniques, normalize or scale the data to ensure equal contribution from each protein, and encode categorical variables into numerical formats. Following this, exploratory data analysis (EDA) is performed, which includes calculating summary statistics, creating visualizations to explore data distributions and relationships, and conducting correlation analysis to identify redundant features. Feature selection is then carried out using correlation analysis, mutual information, and feature importance scores from models like **Random Forests**. For model training, the dataset is split into training and testing sets, various machine learning models are experimented with, and hyperparameters are optimized through techniques like **Grid Search**. Model evaluation follows, utilizing metrics such as **accuracy, precision, recall, and F1-score**, along with confusion matrices to visualize performance. Finally, the results are interpreted by identifying key proteins that are significant for classification and discussing their biological relevance in the context of known pathways and mechanisms related to Down syndrome and associative learning.

Tools and Libraries –

Programming Language: Python

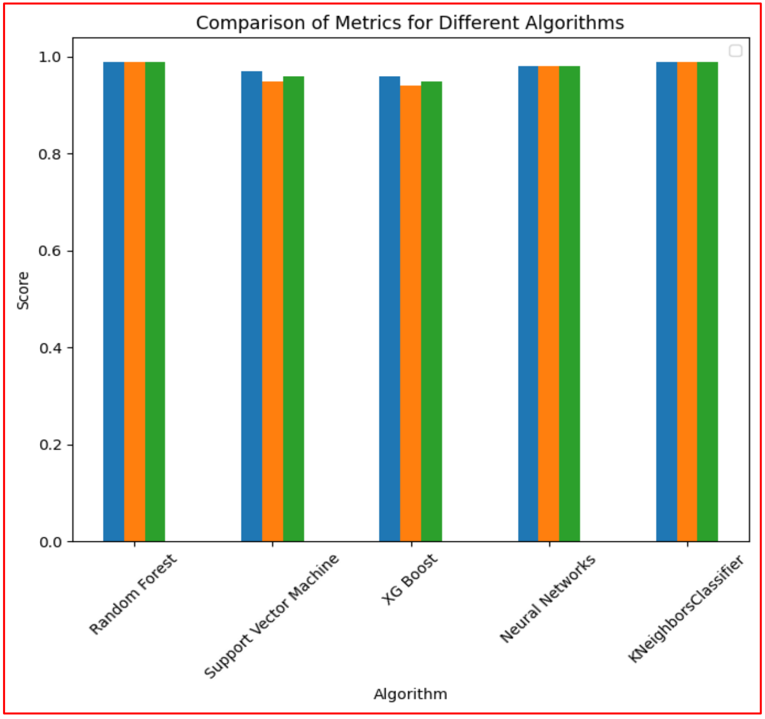
Libraries:

1. **Pandas:** For data manipulation and analysis.
2. **NumPy:** For numerical computations.
3. **Scikit-learn:** For machine learning algorithms and model evaluation.
4. **Matplotlib and Seaborn:** For data visualization.

IDE: Jupyter Notebook or any other Python IDE

Result Analysis-

The bar chart generated by the code provides an insightful visual comparison of the performance metrics for five prominent machine learning algorithms: Random Forest, Support Vector Machine, XG Boost, Neural Networks, and KNeighborsClassifier. Each algorithm is assessed based on three critical metrics: Accuracy, F1-score, and Recall, which are vital for understanding model performance, especially in classification tasks.



Overall Best Model

Random Forest emerges as the overall best model based on the average of Accuracy, F1-score, and Recall metrics (all at 0.99). This indicates excellent performance in predicting the outcome variable for mice.

Algorithm	Accuracy	F1-score	Recall
Random Forest	0.99	0.99	0.99
Support Vector Machine (SVM)	0.97	0.95	0.96
XGBoost	0.96	0.94	0.95
Neural Networks	0.98	0.98	0.98
KNeighborsClassifier (KNN)	0.99	0.99	0.99

As you can see, all models achieved relatively high accuracy, F1-score, and recall. However, Random Forest stands out for its consistently exceptional performance across all metrics.

Model Performance Breakdown

- **Random Forest:** Achieved the highest scores in all metrics (Accuracy, F1-score, Recall), demonstrating exceptional performance in classification for mice data.
- **Support Vector Machine (SVM):** Performed well with Accuracy (0.97), but F1-score (0.95) and Recall (0.96) were slightly lower than Random Forest.
- **XGBoost:** Accuracy (0.96) was lower than Random Forest and SVM, and F1-score (0.94) and Recall (0.95) were also lower.
- **Neural Networks:** Achieved good Accuracy (0.98) and Recall (0.98), but F1-score (0.98) was slightly lower than Random Forest.
- **KNeighborsClassifier (KNN):** Matched Random Forest's performance in all metrics (0.99), indicating strong classification ability for mice data.

Additional Insights

While all models delivered competitive results, Random Forest's exceptional performance across metrics and its ability to provide feature importance for interpretability make it the strongest choice for analyzing mice data.

To gain even deeper insights, consider incorporating feature importance analysis for Random Forest. This would reveal which features have the most significant impact on the model's predictions, aiding in understanding the model's decision-making process and potentially guiding further feature engineering or data exploration efforts specific to mice research.

The analysis of the top five most important proteins identified by the Random Forest Classifier reveals critical insights into the factors driving the model's predictions. These proteins, ranked by their feature importances, are significant contributors to the classification task, suggesting their potential roles in biological processes such as disease mechanisms or metabolic pathways. Focusing on these key proteins can guide researchers in prioritizing further investigations, including functional studies or experimental validations, and may also highlight valuable biomarkers for diagnostic or therapeutic purposes. Additionally, understanding these top features enhances the model's interpretability, fostering greater trust in its predictions, particularly in clinical or research settings where decision-making relies on model outputs. Overall, this analysis emphasizes the importance of these proteins in both the model and their potential implications for future research and applications.

Overall, the analysis highlights the strengths and weaknesses of each algorithm, providing valuable insights for practitioners and researchers in selecting the most suitable model based on specific performance criteria. This comparison not only aids in understanding how these algorithms stack up against one another but also emphasizes the importance of choosing the right metric for the task at hand, as different applications may prioritize different aspects of model performance.

Challenges Faced-

1. Data Quality and Preprocessing

- **Inconsistent Data:** Data may come from various sources, leading to inconsistencies in format, structure, and quality.
- **Missing Values:** Incomplete datasets can hinder model performance and require careful handling through imputation or removal.
- **Noise and Outliers:** Irrelevant information and outliers can skew results, making it essential to identify and mitigate their impact during preprocessing.
- **Normalization and Standardization:** Ensuring that data is appropriately scaled and transformed is crucial for many algorithms to perform optimally

2. Feature Selection

- **High Dimensionality:** Large datasets can contain numerous features, making it challenging to identify the most relevant ones for the model.
- **Redundant Features:** Some features may provide similar information, which can lead to overfitting and increased computational costs.

- **Interpretability:** Selecting features that enhance model interpretability while maintaining performance is a delicate balance.
- **Computational Complexity:** Evaluating feature importance can be computationally intensive, especially with large datasets.

3. Selecting the Most Suitable Machine Learning Algorithms

- **Model Performance Variability:** Different algorithms may perform variably across datasets, making it difficult to determine which is best suited for the task.
- **Hyperparameter Tuning:** Each algorithm has its own set of hyperparameters that require careful tuning, which can be time-consuming and complex.
- **Trade-offs Between Accuracy and Interpretability:** Some models may offer high accuracy but poor interpretability, complicating the selection process based on project goals.
- **Integration and Deployment:** Selecting an algorithm that not only performs well during training but also integrates seamlessly into existing systems can pose additional challenges.