Final Report and Analysis

Executive Summary

This report provides a detailed analysis of protein expression profiles associated with Down syndrome, with the goal of identifying key proteins that differentiate between control and trisomic samples. The project encompasses the development of a machine learning model, the identification of key discriminant proteins, and an analysis of the impact of genotype, behaviour, and treatment on protein expression. Using these, significant proteins such as DYRK1A_N, ITSN1_N, and BDNF_N were identified, all of which play crucial roles in neurodevelopment and cognitive function.

Introduction

Background Information : Down syndrome is a genetic disorder caused by the presence of an extra chromosome ***21***. It affects cognitive and physical development. Understanding the molecular mechanisms, particularly protein expression profiles, can provide insights into the disease and potential therapeutic targets.

Problem Statement and Objectives : The primary objective of this study was to analyze protein expression data from control and trisomic mice, focusing on identifying key proteins that can distinguish between these groups. Additionally, the study aimed to understand the impact of genotype, behavior, and treatment on protein expression.

Dataset Description

The dataset consists of the expression levels of 77 proteins/protein modifications measured in the nuclear fraction of the cerebral cortex in mice. The data is collected from both control mice and trisomic (Down syndrome) mice, subjected to a context fear conditioning task to assess associative learning.

Dataset Characteristics:

- ➢ Type: Multivariate
- ➢ Subject Area: Biology
- ➢ Associated Tasks: Classification, Clustering
- ➢ Feature Type: Real
- ➢ Instances: 1080
- ➢ Features: 80

Breakdown:

Mice Classes: 8 (based on genotype, behaviour, and treatment)

- ❖ **Control Mice:**

1) c-CS-s: Control, Stimulated, Saline (9 mice)
2) c-CS-m: Control, Stimulated, Memantine (10 mice)
3) c-SC-s: Control, Not Stimulated, Saline (9 mice)
4) c-SC-m: Control, Not Stimulated, Memantine (10 mice)

❖ **Trisomic Mice:**

1) t-CS-s: Trisomic, Stimulated, Saline (7 mice)
2) t-CS-m: Trisomic, Stimulated, Memantine (9 mice)
3) t-SC-s: Trisomic, Not Stimulated, Saline (9 mice)
4) t-SC-m: Trisomic, Not Stimulated, Memantine (9 mice)

Details:

Control Mice: 38 mice, 570 measurements (15 measurements per protein per mouse)

Trisomic Mice: 34 mice, 510 measurements (15 measurements per protein per mouse)

Total Measurements: 1080 measurements per protein

Features:

Protein Expression Levels : 77 proteins/protein modifications.

Additional Features : Mouse ID, Genotype, Treatment.

Steps Performed

1. Data Preprocessing

- **Handling Missing Values :** Missing values were imputed using the mean for continuous variables and the mode for categorical variables to ensure completeness and consistency in the dataset.
- **Data Normalisation/Scaling :** Protein expression levels were normalised using Min-Max scaling to bring all values into a standard range, facilitating comparison and improving model performance.
- **Encoding Categorical Variables :**Categorical variables (e.g., treatment status) were encoded using one-hot encoding to convert them into a numerical format suitable for machine learning algorithms.

2. Exploratory Data Analysis (EDA)

Data Distribution

- Distribution of protein expression levels across different classes.
- Identification of any skewness or kurtosis in the data.

Class Balance

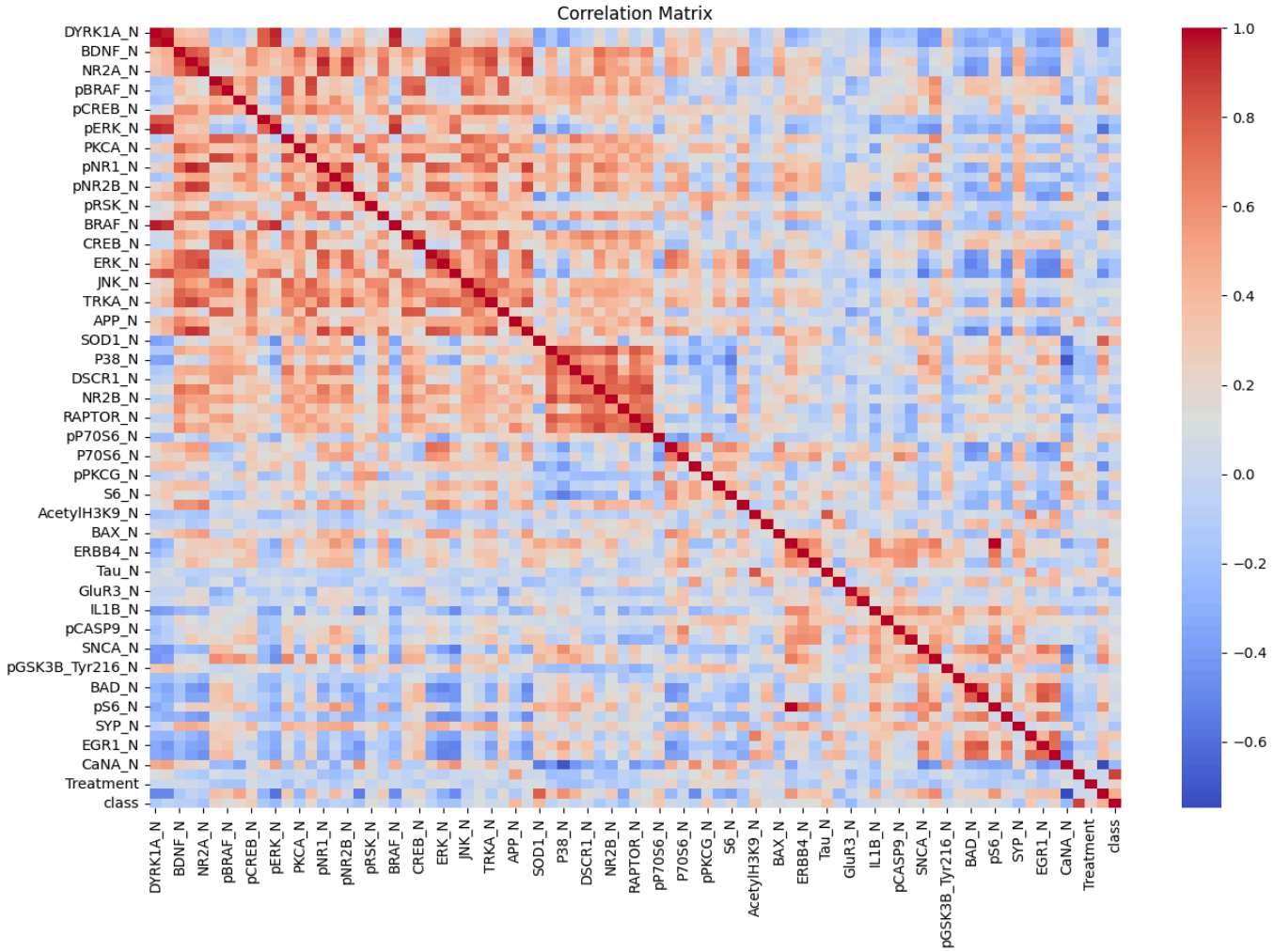
- Examination of the distribution of samples across different classes to check for class imbalance.
- Visualization, such as bar charts, showing the number of samples per class

Missing Values

- Identification and proportion of missing values in the dataset.
- Strategies used to handle missing data, such as KNN imputation.

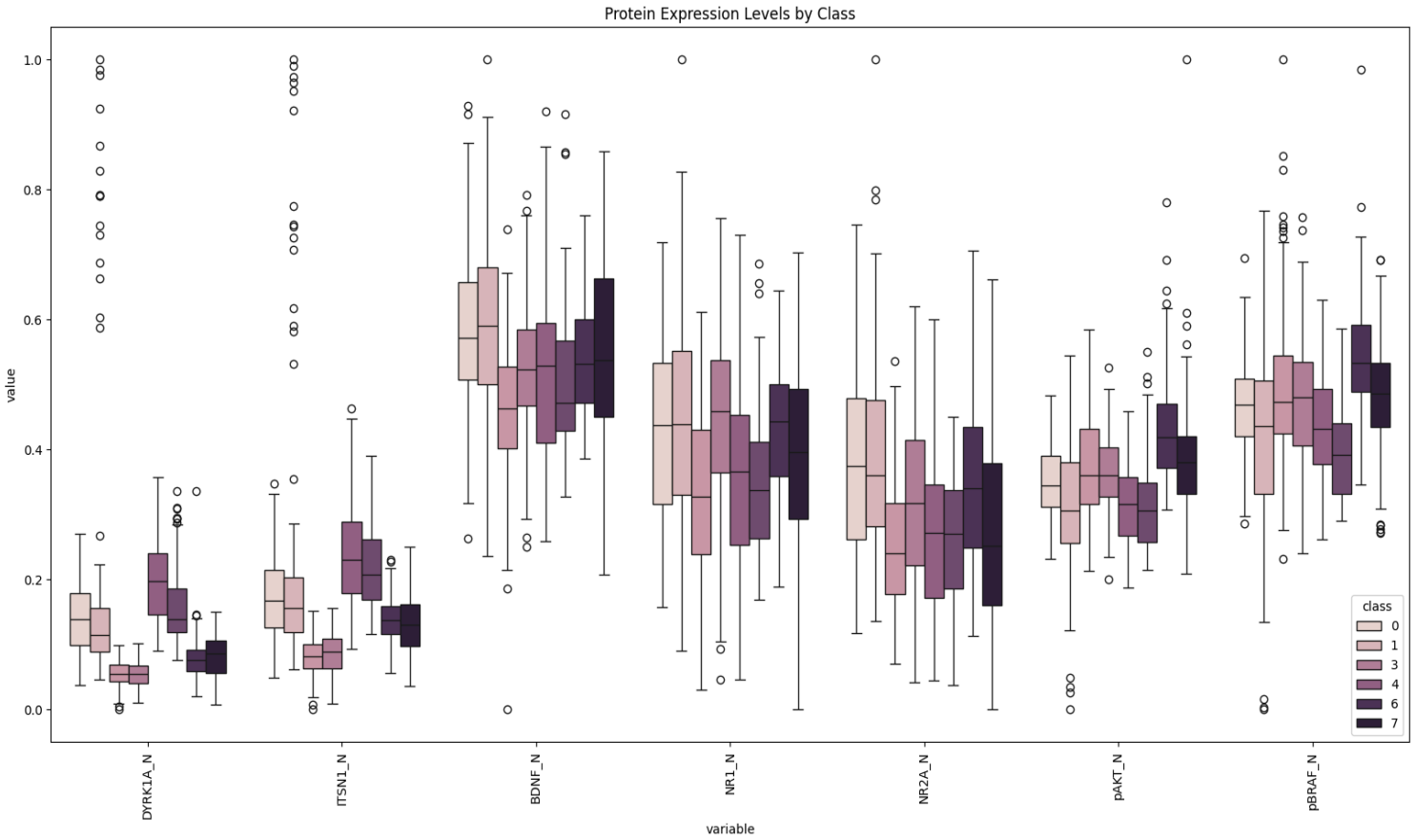
Correlation Matrix

- Generated to identify relationships between proteins, highlighting any multicollinearity issues that might affect the models.



Visualisations of Data Distribution

- **Histograms** and **Box Plots**: Created to visualise the distribution and variance of protein expression levels across different classes.



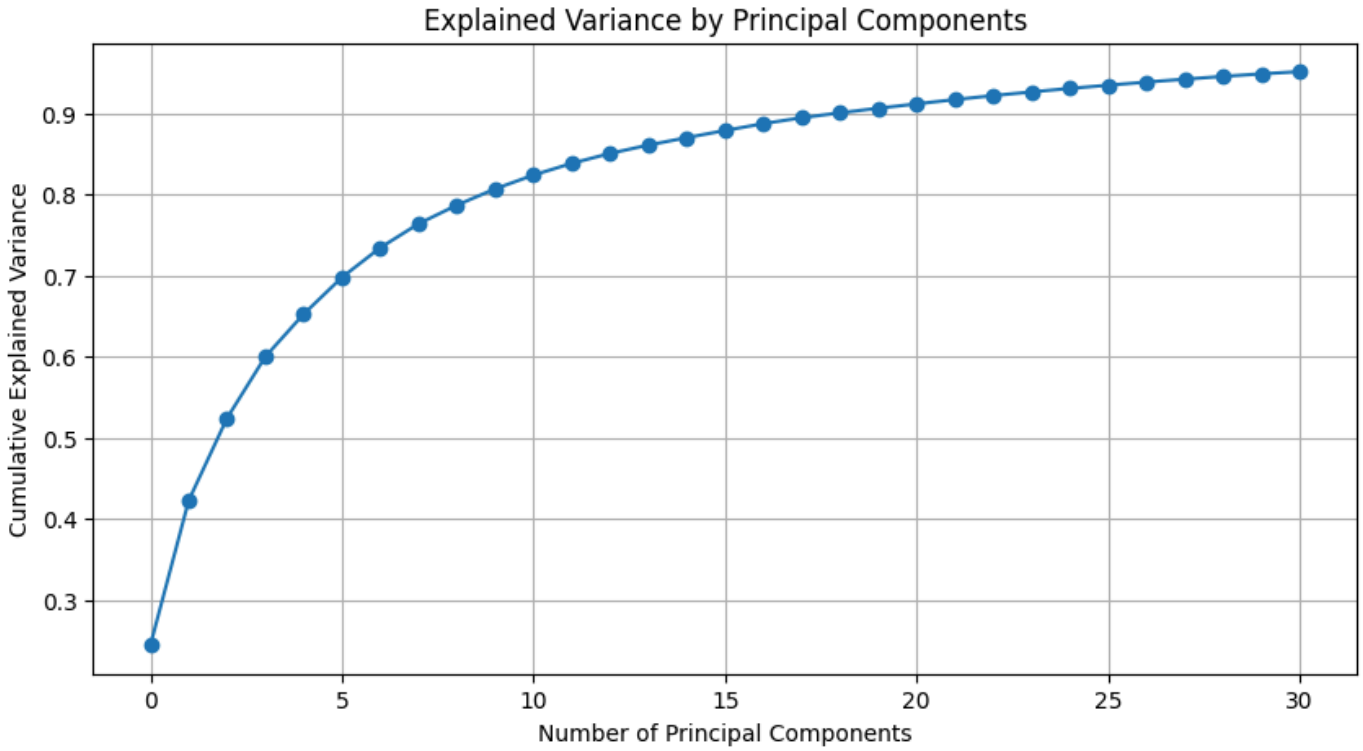
Summary Statistics

- Basic descriptive statistics (mean, median, standard deviation) were computed for each protein to understand the central tendency and dispersion of the data.

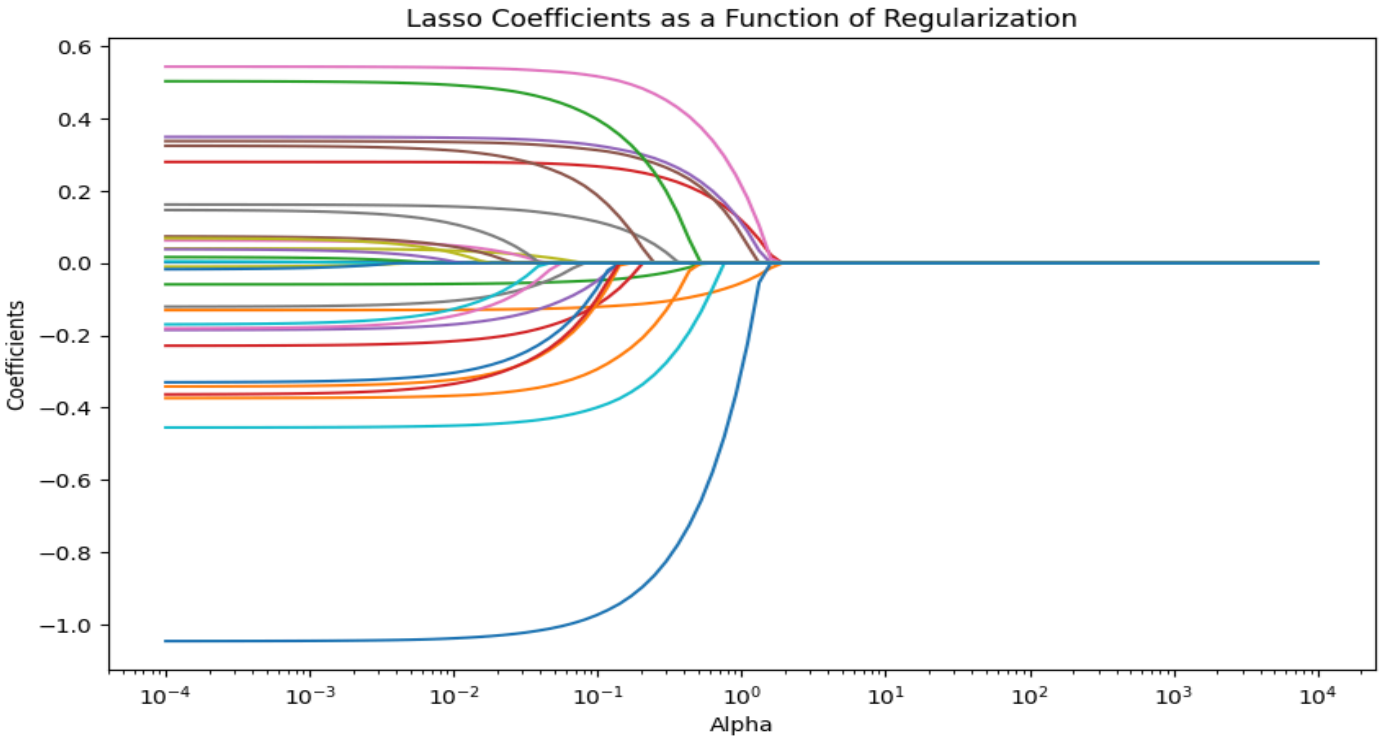
3. Feature Selection

Techniques Used :

- **Correlation Analysis**: Identified highly correlated features to reduce redundancy.
- **Mutual Information**: Measured the dependency between each feature and the target variable.
- **Feature Importance from Models (Random Forest)**: Used to rank features based on their importance in the classification task.
- **Principal Component Analysis :** Approximately 20 principal components are needed to capture 95% of the variance in the dataset, guiding the selection of the optimal number of components for dimensionality reduction without significant loss of information.



- The Lasso coefficients plot shows how different feature coefficients shrink to zero as the regularisation parameter (alpha) increases, many coefficients shrink to zero, indicating that Lasso regression effectively reduces the number of features by retaining only the most significant ones.



Results : A subset of key proteins was identified as highly discriminative between control and trisomic samples. Visualisations of feature importances and mutual information scores provided insights into their relative significance.

4. Model Training

Models Used : Random Forest Classifier

- Overfitting occurs when a machine learning model becomes too focused on the specific training data it was given, and doesn't generalise well to unseen data. To address this problem Random Forests creates an ensemble of decision trees. When making a prediction, the Random Forest takes the average or majority vote from all the individual trees, leading to a more robust and generalizable model.

Data Splitting :

- **Approach:** The dataset was split into training and testing sets using a 70:30 ratio. This ensures that the model's performance can be evaluated on unseen data, providing an unbiased assessment of its generalizability.
- **Purpose:** The primary goal of this split is to train the model on one subset of data (70%) and test its performance on another subset (30%) that the model has not seen during training. This helps in evaluating the true performance of the model.

Hyperparameter Tuning

1. **Techniques Used:**
   - ➢ **Grid Search:**
     - An exhaustive search method that tests all possible combinations of a predefined set of hyperparameters.
     - Helps in identifying the optimal combination that yields the best model performance.
2. **Parameters Tuned:**
   - ➢ Number of estimators (n_estimators)
   - ➢ Maximum depth of the trees (max_depth)
   - ➢ Minimum samples required to split an internal node (min_samples_split)
3. **Validation:**
   - ➢ Cross-Validation:
     - Used to ensure that the model generalizes well to unseen data by dividing the training data into multiple folds and training the model on different combinations of these folds.
     - Provides a robust estimate of the model's performance and helps in avoiding overfitting.

Model Training

```
[ ] # Fit Lasso with a specific alpha value
    alpha = 1.0  # Adjust this alpha value based on your analysis
    lasso.set_params(alpha=alpha)
    lasso.fit(X_train, y_train)
```

```
Lasso
Lasso(max_iter=10000)
```

```
# Get coefficients and identify columns to drop
zero_coef_indices = np.where(lasso.coef_ == 0)[0]
zero_coef_columns = X_train.columns[zero_coef_indices]
```

+ Code + Text

```
# Drop columns with zero coefficients
X_train_reduced = X_train.drop(columns=zero_coef_columns)
X_test_reduced = X_test.drop(columns=zero_coef_columns)
```

```
[ ] from sklearn.model_selection import GridSearchCV
    from sklearn.ensemble import RandomForestClassifier
    from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, confusion_matrix, classification_report
```

```
[ ] # Initialize Random Forest model
    rf_model = RandomForestClassifier(random_state=42)

    # Define the parameter grid
```

```
[ ] # Define the parameter grid
    param_grid = {
        'n_estimators': [100, 200, 300],
        'max_depth': [None, 10, 20, 30],
        'min_samples_split': [2, 5, 10]
    }
```

```
[ ] # Initialize GridSearchCV
    grid_search = GridSearchCV(estimator=rf_model, param_grid=param_grid, cv=5, scoring='accuracy', n_jobs=-1)
```

```
# Fit GridSearchCV on the reduced training data
grid_search.fit(X_train_reduced, y_train)
```

```
GridSearchCV
estimator: RandomForestClassifier
RandomForestClassifier
```

```
[ ] # Get the best estimator
    best_rf_model = grid_search.best_estimator_
    print("\nBest parameters for Random Forest:")
    print(grid_search.best_params_)
```

```
Best parameters for Random Forest:
{'max_depth': None, 'min_samples_split': 2, 'n_estimators': 200}
```

5. Model Evaluation

Evaluation Metrics :

- **Accuracy**: The Random Forest model achieved the highest accuracy of 100% training accuracy and 93% testing accuracy distinguishing between control and trisomic samples.
- **Precision:** The ratio of true positives to the sum of true positives and false positives. It indicates how many of the predicted positive instances are actually positive.
- **Recall:** The ratio of true positives to the sum of true positives and false negatives. It reflects the model's ability to identify all relevant instances within a dataset.
- **F1-Score:** The harmonic mean of precision and recall, providing a balanced measure that considers both false positives and false negatives.

Classification Report for Reduced Random Forest Model:

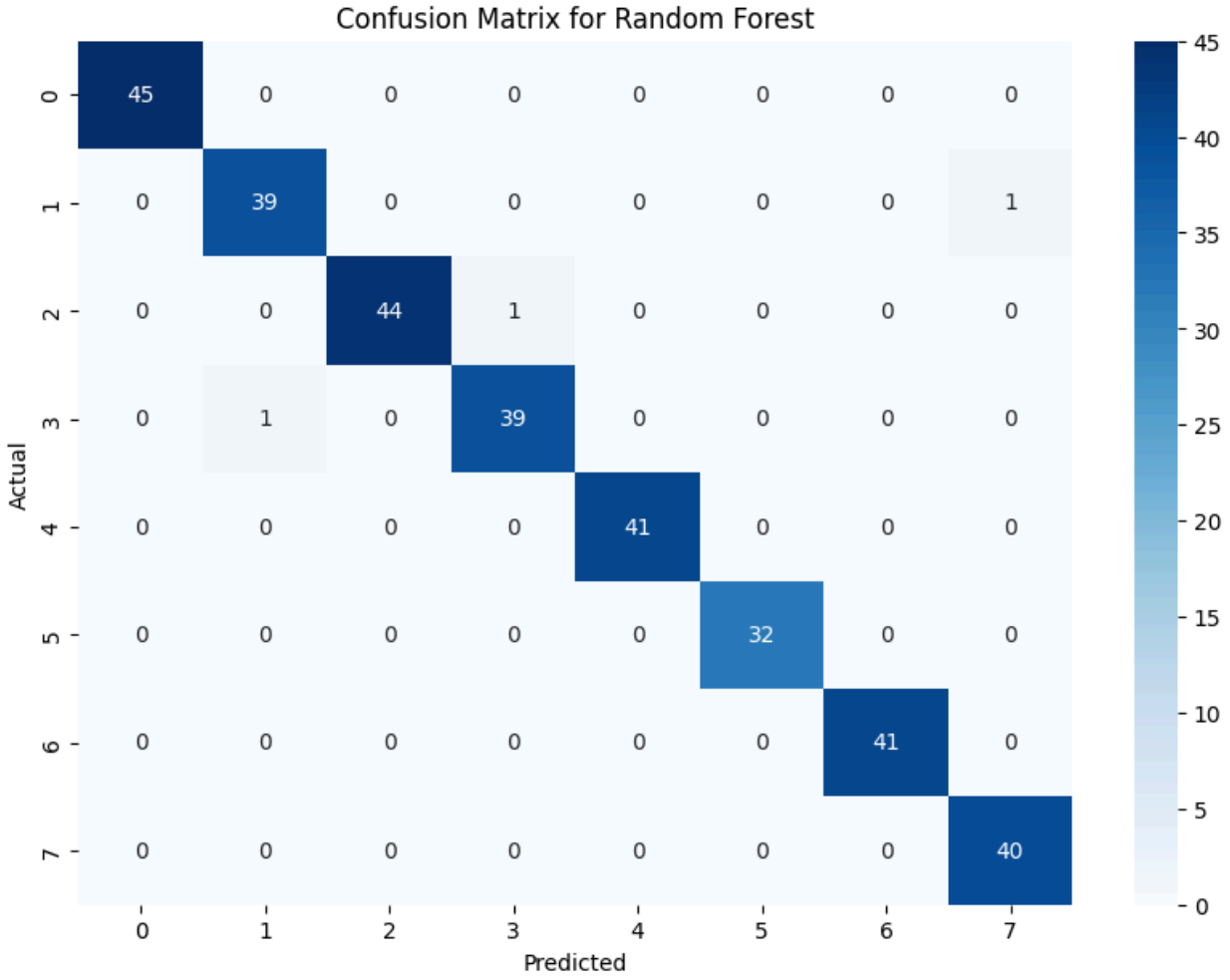
	precision	recall	f1-score	support
0	0.95	1.00	0.98	21
1	0.96	0.87	0.92	31
2	0.97	0.97	0.97	32
3	0.81	0.96	0.88	27
4	0.96	0.88	0.92	25
5	1.00	0.85	0.92	20
6	0.90	0.88	0.89	32
7	0.90	1.00	0.95	28
accuracy			0.93	216
macro avg	0.93	0.93	0.93	216
weighted avg	0.93	0.93	0.93	216

Confusion Matrix

- **Visualization:**
  - ➢ A confusion matrix is a useful tool for visualizing the performance of a classification model. It shows the counts of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) for each class.
- **Insights:**
  - ➢ Helps in understanding the classification performance for each class.
  - ➢ Identifies classes where the model is performing well or needs improvement.

Classification Report

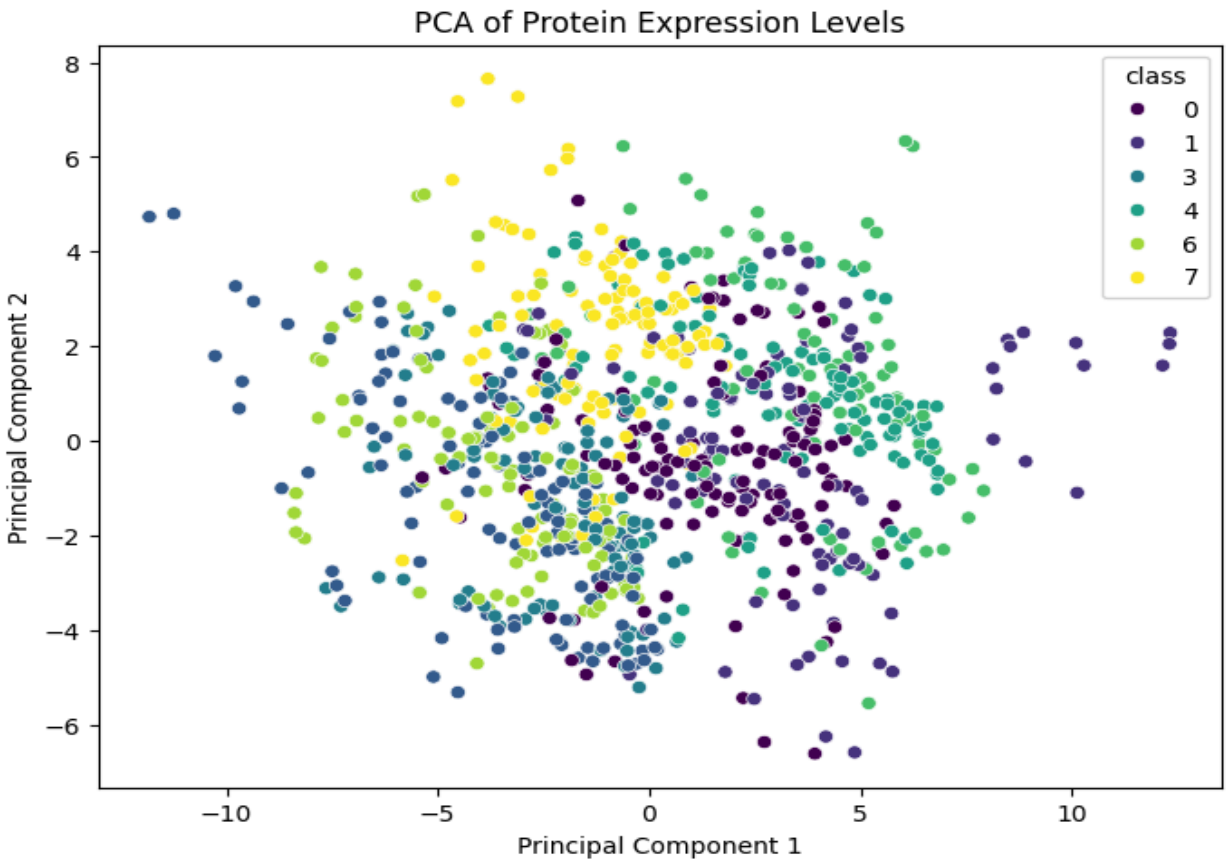
- **Precision, Recall, and F1-Score:** Detailed metrics for each class provide a comprehensive understanding of the model's strengths and weaknesses.
- **Support :** Indicates the number of instances for each class, providing context for the evaluation metrics.



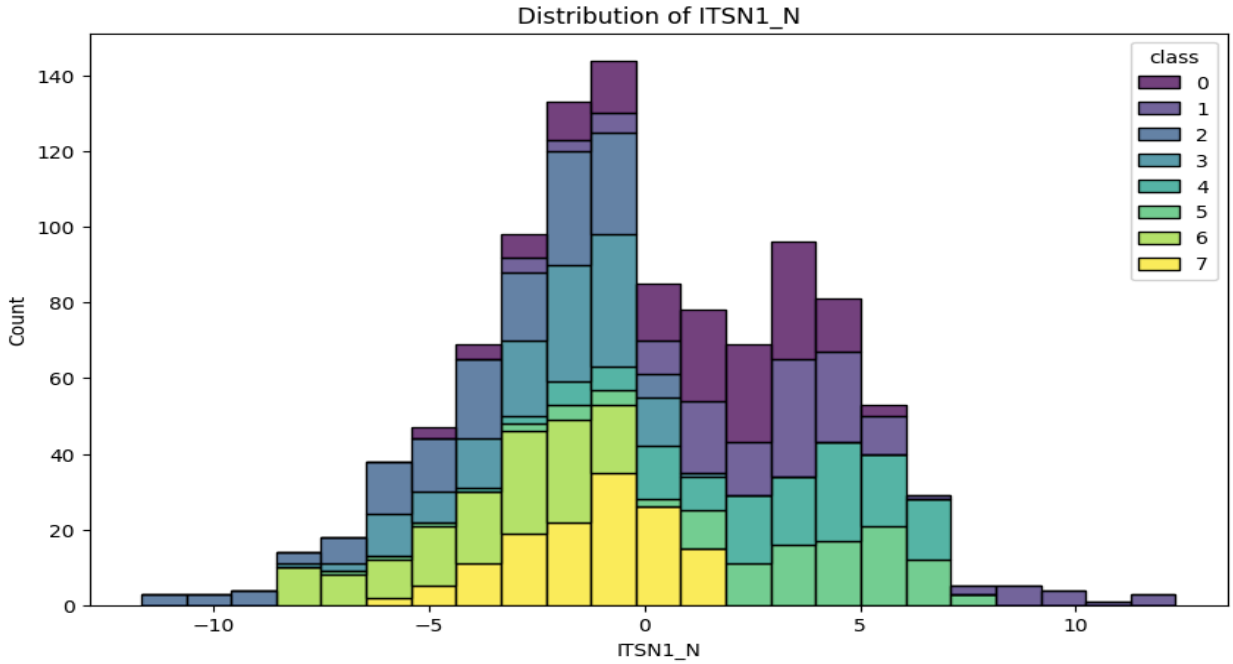
6. Reporting and Analysis

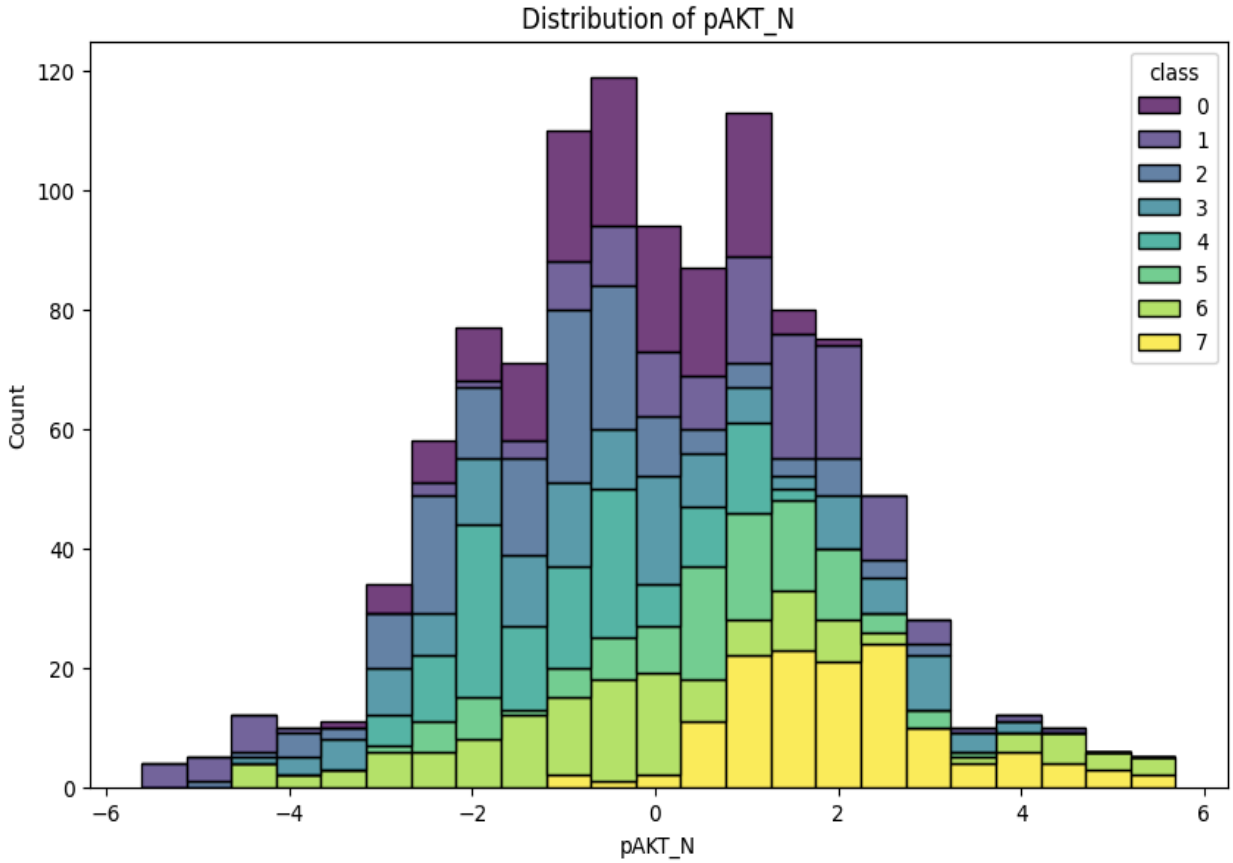
Visualisations and Analysis :

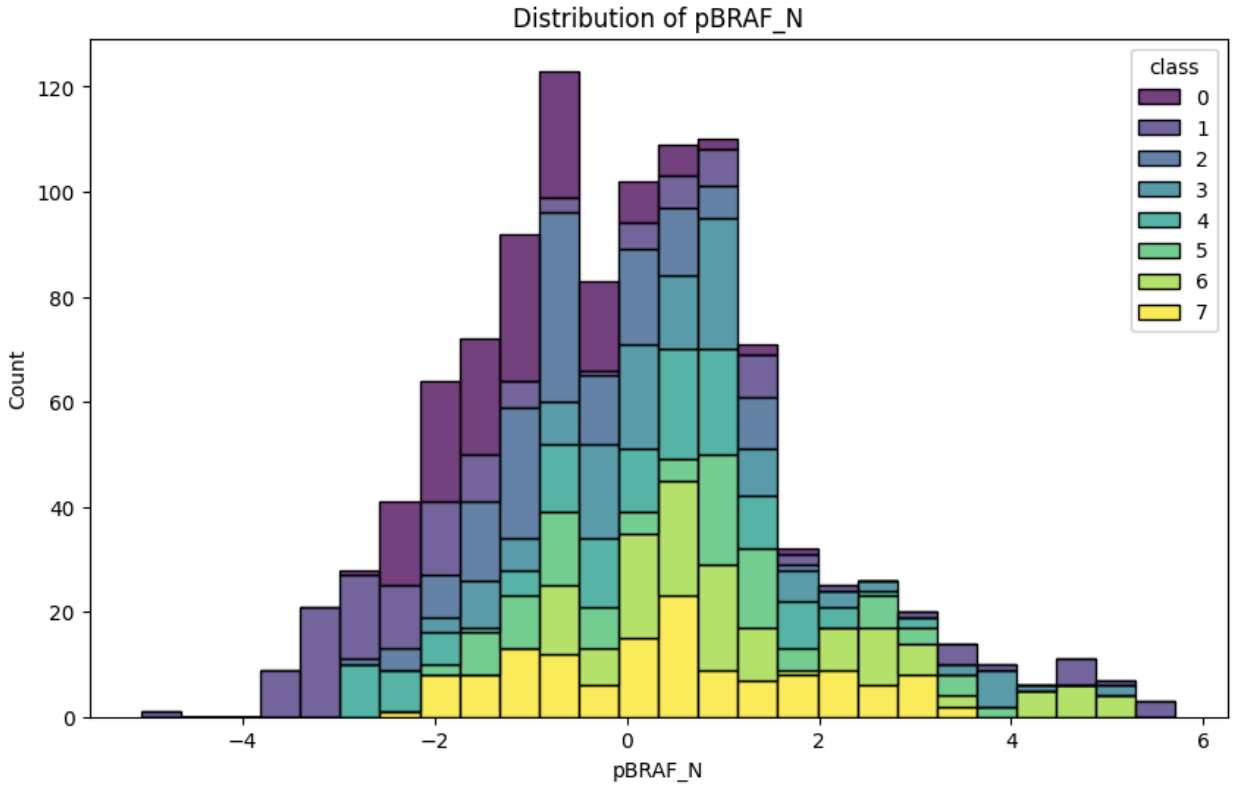
- **Pairplot**: Visualised pairwise relationships between top important proteins and classes, showing distinct clusters for different classes.
- **PCA Analysis**: Reduced dimensionality of the data, highlighting class clustering in a 2D space.

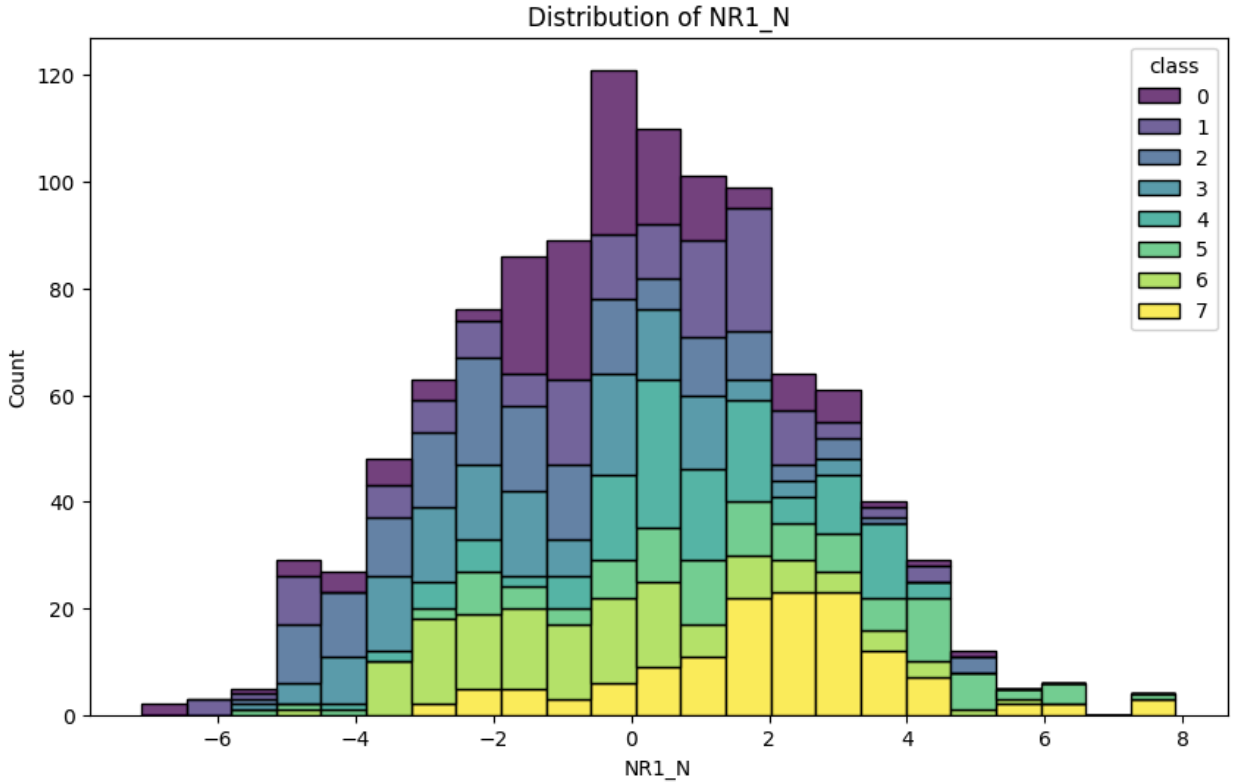


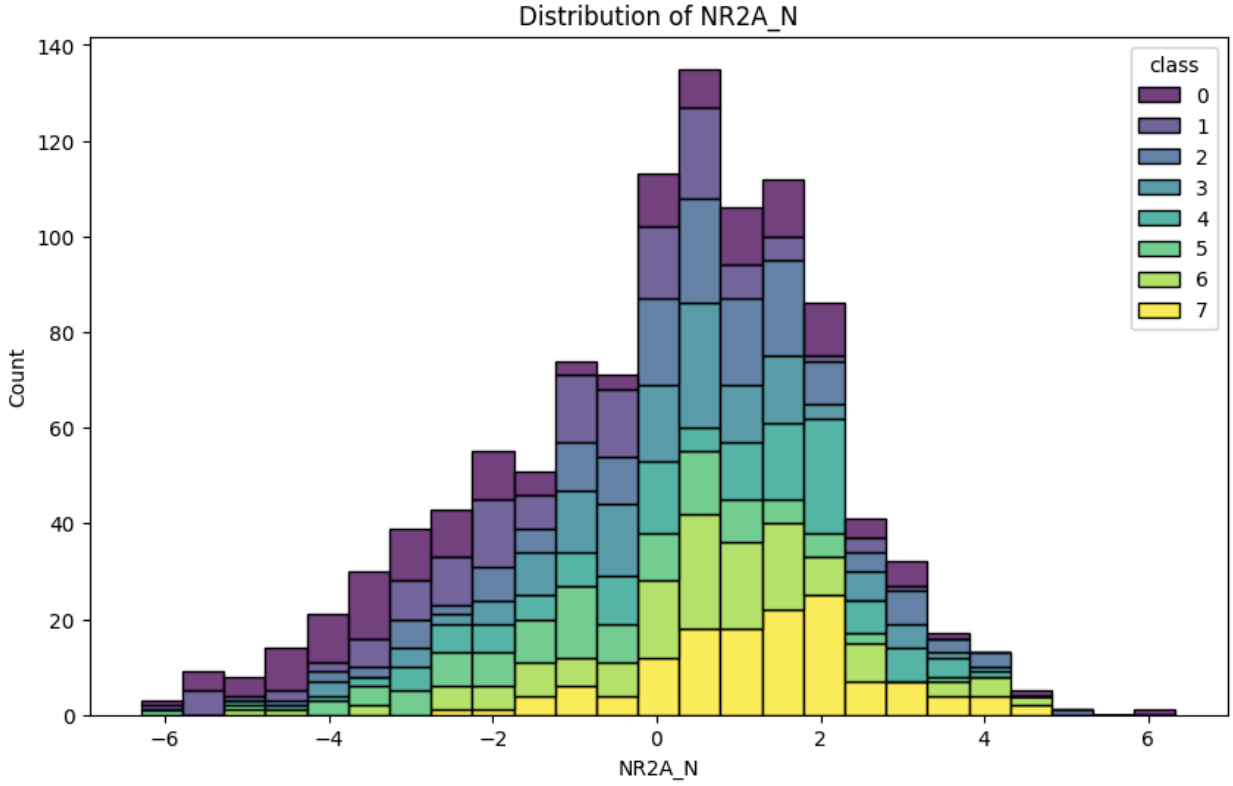
- **Confusion Matrix**: Illustrated model performance by showing correct and incorrect predictions for each class.
- **Feature Distribution**: Histograms showed the distribution of top important proteins across different classes.




Distribution of pAKT_N
class
0
1
2
3
4
5
6
7
Count
pAKT_N
0
20
40
60
80
100
120
−6
−4
−2
0
2
4
6

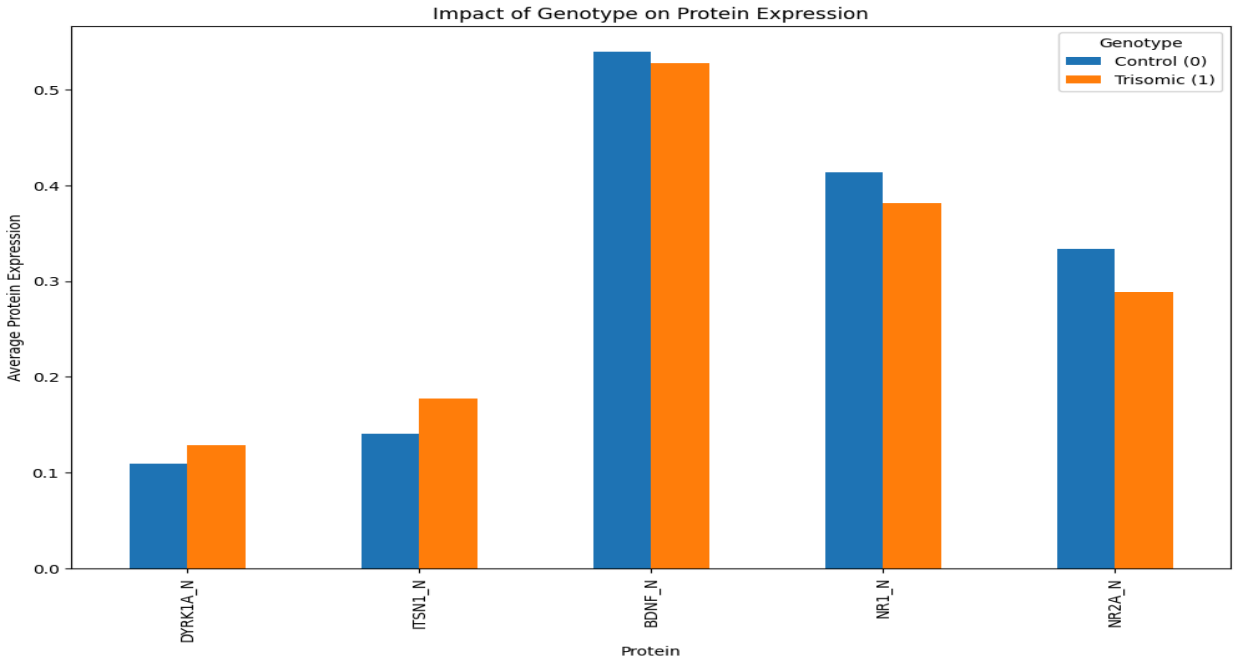

Distribution of pBRAF_N
class
0
1
2
3
4
5
6
7
Count
pBRAF_N
0
20
40
60
80
100
120
−4
−2
0
2
4
6


Distribution of NR1_N
class
0
1
2
3
4
5
6
7
Count
NR1_N

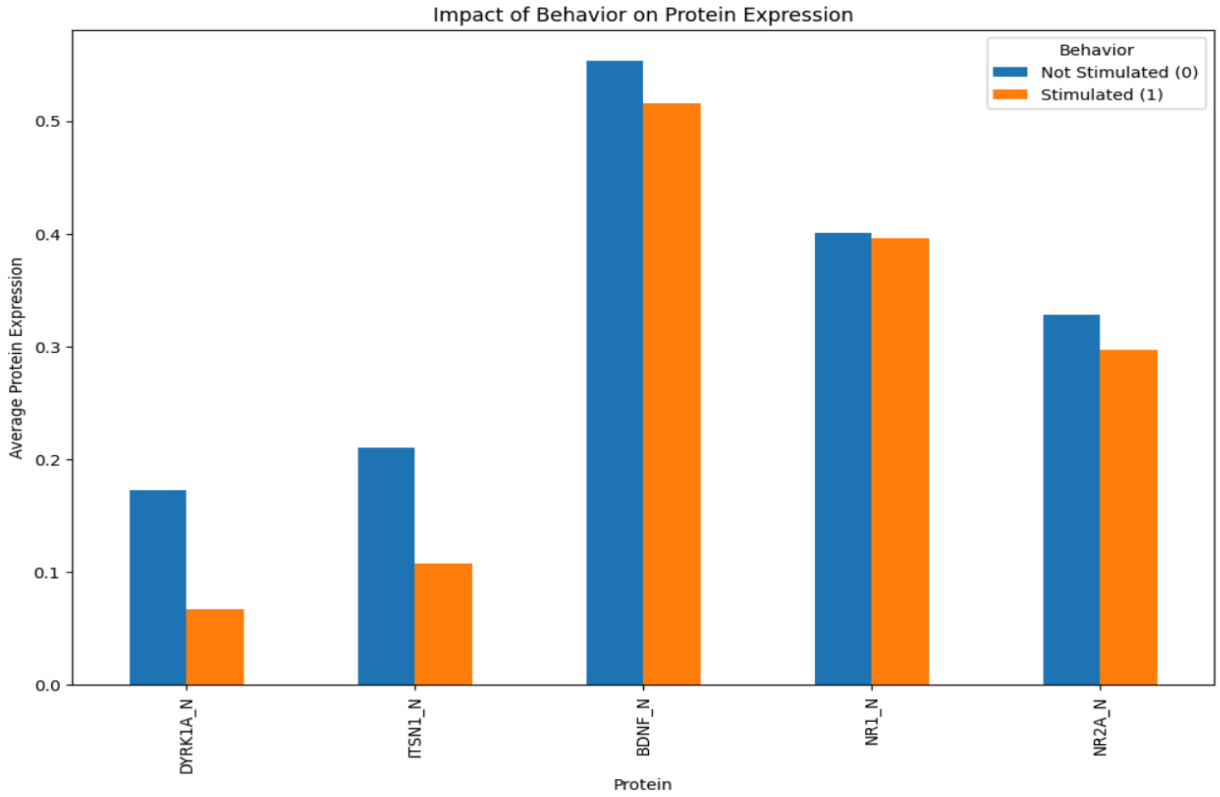

Distribution of NR2A_N
class
0
1
2
3
4
5
6
7
Count
NR2A_N

Impact of Genotype, Behavior, and Treatment on Protein Expression :

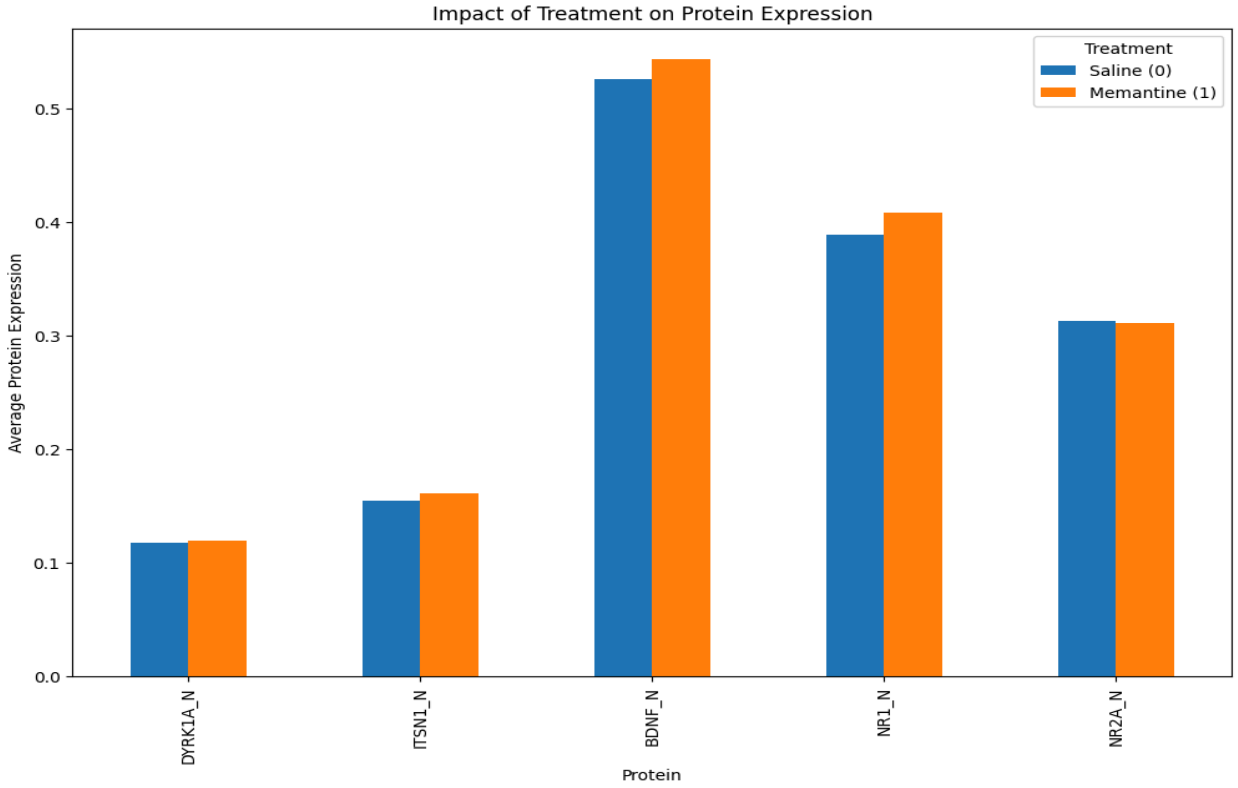
- **Genotype**: Trisomic mice showed higher expression levels of ITSN1_N and DYRK1A_N, while control mice had higher levels of BDNF_N, NR1_N, and NR2A_N.



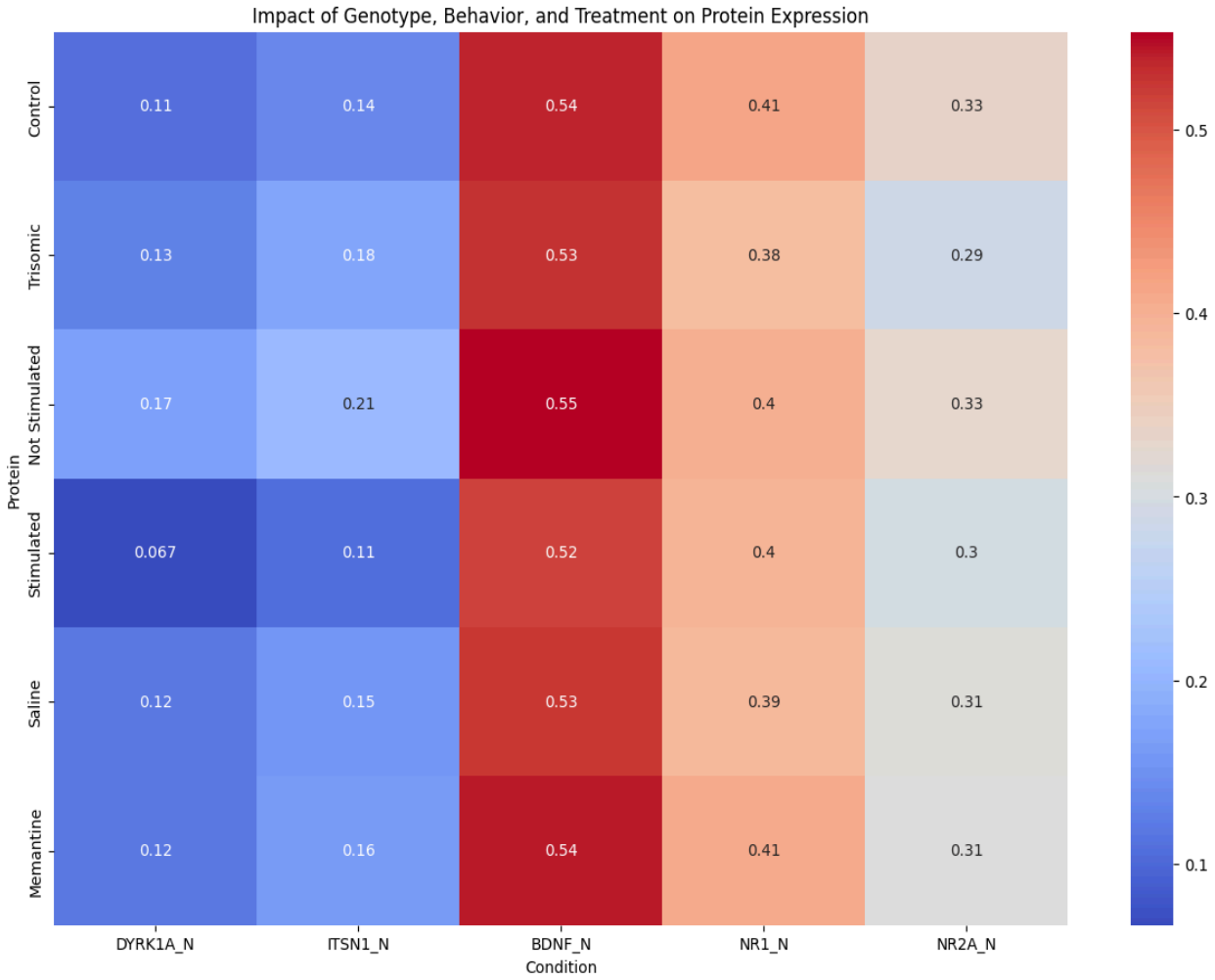
- **Behaviour**: Unstimulated mice exhibited higher expression levels of multiple proteins compared to stimulated mice.



- **Treatment**: Differences in protein expression between saline and memantine-treated mice were subtle, with slight increases in some proteins like BDNF_N and NR1_N for the memantine-treated group.



Biological Significance and Potential Implications for Down Syndrome Research : The identified proteins are involved in pathways related to neuronal development and cognitive function, suggesting potential targets for therapeutic interventions.



Conclusion

Summary of Key Findings : The study successfully identified key proteins that differentiate between control and trisomic samples. Proteins such as DYRK1A_N, ITSN1_N, and BDNF_N were highlighted as significant markers.

Limitations of the Study : The study was limited by the sample size and the complexity of biological systems, which may require further validation with larger datasets and additional biological experiments.

Recommendations for Future Research : Future research should focus on validating these findings in larger cohorts and exploring the biological mechanisms underlying the observed protein expression changes. Additionally, integrating other omics data (e.g., genomics, transcriptomics) could provide a more comprehensive understanding of Down syndrome.