Project title:

Mice Protein Expression

By INeuron

Team Members:

* Abhishek M Ganjigatti
* Manoj Kendri

Date : 19-01-2024

**Project link GitHub:** [**https://github.com/abhishek-ganjigatti/Mice-protein-expression-ml-project**](https://github.com/abhishek-ganjigatti/Mice-protein-expression-ml-project)

**Detailed Project Report (DPR)**

**1. Introduction:**

**1.1 Background and Context of the Project:**

The Mice Protein Expression Classification project aims to address the challenging task of identifying discriminant protein subsets between control and Down syndrome mice exposed to context fear conditioning. Leveraging machine learning, the project focuses on achieving a balance between accuracy and efficiency in the analysis validation process.

**1.2 Objectives and Goals:**

Develop a classification model for protein expression in mice.

The trade-off between accuracy and efficiency.

Identify discriminant protein subsets.

Implement a user-friendly API or interface for model testing.

**2. Problem Statement:**

**2.1 Problem Definition:**

The challenge involves classifying the protein expression levels in the cerebral cortex of mice across different classes (control and Down syndrome) exposed to context fear conditioning. The problem is framed as a multi-class classification issue with a substantial amount of data, requiring a thoughtful trade-off between accuracy and efficiency.

**2.2 Significance and Relevance:**

Understanding protein expression patterns in mice is crucial for unraveling associative learning processes. The project's outcomes could have implications for neuroscientific research and potentially contribute to advancements in understanding Down syndrome-related mechanisms.

**3. Data and Database:**

**3.1 Source of Data:**

The dataset, comprising 77 proteins' expression levels in 72 mice, was obtained from [Astra](https://astra.dev/ineuron). The dataset includes 15 measurements for each protein per mouse, resulting in a total of 1080 measurements per protein.

**3.2 Data Cleaning and Preprocessing:**

Missing values imputed using the median.

Categorical variables encoded using LabelEncoder.

The dataset divided into features and target variables for training.

**4. Methodology:**

**4.1 Machine Learning Algorithms:**

Support Vector Machine (SVM) with a linear kernel was chosen for its suitability in classification tasks. SVM offers a good balance between accuracy and efficiency.

**4.2 Pre-processing and Feature Engineering:**

Feature selection based on relevant protein columns.

Imputation of missing values using the median.

Label encoding for categorical variables.

**5. Model Training:**

**5.1 Splitting Data and Hyperparameter Tuning:**

The dataset is split into training and testing sets.

Hyperparameter tuning performed for SVM.

**5.2 Evaluation Metrics:**

Model performance evaluated using accuracy.

Additional metrics include precision, recall, and F1 score.

**6. Results:**

**6.1 Model Performance Metrics:**

The SVM model achieved an accuracy of [insert accuracy]. Precision, recall, and F1 score metrics provided a comprehensive evaluation of the model's performance.

**6.2 Confusion Matrix:**

A confusion matrix visualized the model's classification results, providing insights into true positives, true negatives, false positives, and false negatives.

**6.3 Comparison with Baseline Models:**

[Include information on any baseline models used for comparison and their performance.]

**7. Lessons Learned:**

**7.1 Technical Insights:**

Importance of feature engineering in protein expression data.

Efficient handling of missing values and categorical variables.

**7.2 Non-Technical Takeaways:**

Collaboration between domain experts and data scientists is crucial for effective problem-framing.

**8. Conclusion:**

**8.1 Key Findings:**

Successful development of a classification model.

A balanced trade-off between accuracy and efficiency.

**8.2 Contribution and Significance:**

The project contributes to the understanding of protein expression patterns in mice, particularly in the context of associative learning. The model can potentially aid in identifying biomarkers related to Down syndrome.

**9. Acknowledgments:**

The project acknowledges the contribution of [List individuals or organizations]. Their support and collaboration significantly impacted the project's success.

**10. Future Recommendations:**

**10.1 Areas for Improvement:**

Experiment with advanced machine learning models.

Explore additional features or engineering techniques.

**10.2 Future Work:**

Incorporate data from diverse sources for a more comprehensive model.

Continuous monitoring and updates to enhance model performance.

**11.References:**

Citations for any external resources, datasets, or libraries used.

* Siddhardhan (YouTube channel) - https://youtu.be/WLwjvWq0GWA?si=cDnYtO46e0cR-xtt

.

**12. Documentation Links:**

**High-Level Document:** <https://github.com/abhishek-ganjigatti/Mice-protein-expression-ml-project/blob/0838de66987f69e30a206bc78405fd7d6bf16864/High%20Level%20Documentation.docx>

**Architecture Document:** <https://github.com/abhishek-ganjigatti/Mice-protein-expression-ml-project/blob/0838de66987f69e30a206bc78405fd7d6bf16864/System_architecture.docx>

**Wireframe Document:** <https://github.com/abhishek-ganjigatti/Mice-protein-expression-ml-project/blob/e999408181bd223d42f5374686cad1d113edf8c2/WireFrame.docx>

13. Project Demo Video:

<https://github.com/abhishek-ganjigatti/Mice-protein-expression-ml-project/blob/2f39cbce47b3cef0711118c9084ca8ba94cc8f9d/project_demo_video1_20.20.22.mp4>

14. LinkedIn Post:

<https://www.linkedin.com/posts/abhishek-ganjigatti-01701b265_deploy-machine-learning-model-using-streamlit-activity-7155643814060134400-oJxs?utm_source=share&utm_medium=member_desktop>

This detailed project report provides a comprehensive overview of the Mice Protein Expression Classification project, covering its objectives, methodology, results, and future recommendations. The associated documentation links offer in-depth insights into the project's development and documentation.