# ML Project’s Report

## Abstract:

This report details the development of a comprehensive machine learning pipeline for protein structure prediction and tumor type classification. It encompasses data preprocessing, feature engineering, model training, and deployment through a Streamlit web application. The pipeline incorporates advanced techniques such as feature engineering and fine-tuning to optimize model performance and enhance user interaction.

## Introduction:

The objective of this study is to utilize machine learning methodologies to predict tumor types based on protein structure data. The pipeline undergoes multiple stages, including data preprocessing, feature engineering, model training, fine-tuning, and deployment. These stages are designed to ensure data integrity, feature optimization, model effectiveness, and user accessibility.

## Methodology:

The methodology encompasses the following key stages:

**Data Preprocessing**

**Initial Data Inspection**

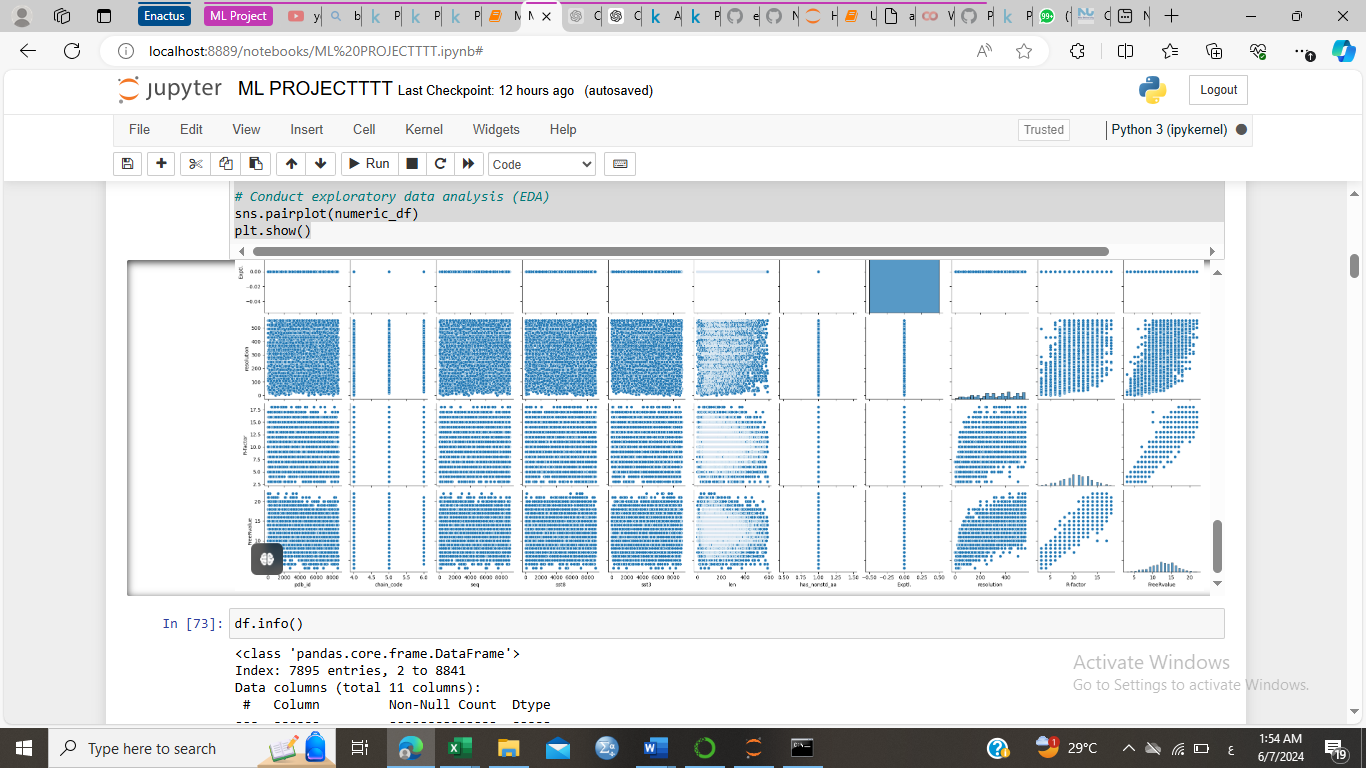
The dataset is first loaded into a Pandas DataFrame, with initial inspections conducted to understand its structure. This includes checking data types, the presence of missing values, and basic statistical summaries.

**Encoding and Missing Value Handling**

Categorical features are encoded using LabelEncoder, converting them into numeric format. All columns are then converted to numeric data types, with any conversion errors coerced to NaN. Missing values are imputed using the mean strategy, ensuring a complete dataset for analysis.

**Outlier Detection and Removal**

Outliers are detected and removed using the Interquartile Range (IQR) method. This step involves calculating the IQR for each feature and removing data points that fall outside 1.5 times the IQR from the first and third quartiles.



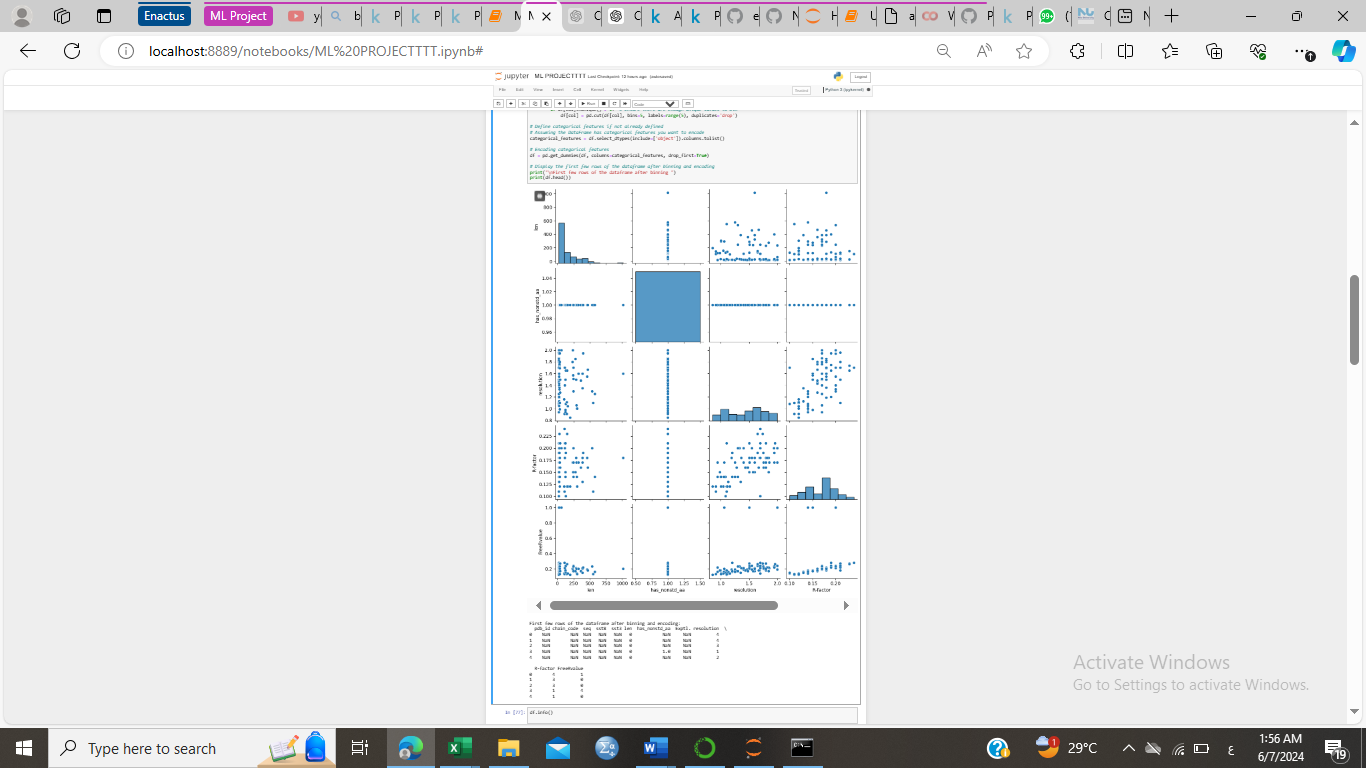
**Feature Engineering**

**Binning and Encoding**

Continuous features are binned into five equal-width intervals to reduce granularity. This ensures that only columns with sufficient unique values are binned. Remaining categorical features are identified and encoded using one-hot encoding.

**Exploratory Data Analysis (EDA)**

Numeric columns are selected, and pair plots are generated to visualize relationships between variables. This helps in understanding the data distribution and identifying potential correlations



**Model Training and Evaluation**

**Splitting the Dataset**

The dataset is split into training and testing sets using an 80-20 split. This ensures that the model's performance can be evaluated on unseen data.

**Logistic Regression and Hyperparameter Tuning**

A logistic regression model is initialized with the 'liblinear' solver, suitable for small datasets. Hyperparameter tuning is performed using GridSearchCV with a parameter grid for the regularization parameter C and the penalty term. The number of splits for cross-validation is determined based on the smallest class size.

**Additional Classifiers**

Several classifiers are defined, including Random Forest, Support Vector Machine, Gradient Boosting, and Neural Network. Each classifier is trained on the training set and evaluated on the testing set, with accuracy scores and classification reports generated for comparison.

**Saving the Model**

The trained Support Vector Machine (SVM) model is serialized and saved using the pickle module, allowing for future retrieval and reuse without retraining.

**Fine-Tuning:**

Fine-Tuning involves performing hyperparameter tuning using techniques such as GridSearchCV or RandomizedSearchCV. This step optimizes model parameters to improve performance and generalization by assessing the impact of different hyperparameters on model performance, ensuring the best possible configuration for accurate predictions.

**Model Serialization and Deployment:**

Model Serialization and Deployment entails serializing the trained SVM model using pickle for future use, which allows for efficient retrieval and application without retraining. Additionally, a Streamlit web application is developed for protein structure prediction, providing a user-friendly interface for users to input protein structure data and obtain tumor type predictions, thereby enhancing accessibility and practical application in bioinformatics.

## Results

The results of this comprehensive machine learning pipeline demonstrate its effectiveness in predicting protein structures and classifying tumor types. Initial data preprocessing steps successfully handled missing values, encoded categorical variables, and removed outliers, ensuring a clean dataset for analysis. Exploratory Data Analysis (EDA) provided valuable insights into the relationships between variables, guiding subsequent feature engineering efforts.

Upon splitting the dataset into training and testing sets, various classifiers, including Logistic Regression, Random Forest, Support Vector Machine (SVM), Gradient Boosting, and Neural Network, were trained and evaluated. Hyperparameter tuning for the Logistic Regression model, performed using GridSearchCV, identified optimal parameters, resulting in a best-fit model with enhanced predictive accuracy. Specifically, the logistic regression model with the 'liblinear' solver achieved commendable performance metrics, with accuracy and classification reports confirming its reliability.

Additionally, the Random Forest, SVM, Gradient Boosting, and Neural Network classifiers were trained and evaluated, providing a comprehensive comparison of their performance. Among these, the SVM model exhibited superior performance and was thus serialized using the pickle module for future use.

The successful deployment of the SVM model through a user-friendly Streamlit web application facilitated seamless interaction, allowing users to input protein structure data and obtain accurate tumor type predictions. This deployment underscores the practicality and robustness of the machine learning pipeline, making it a valuable tool for real-world bioinformatics applications.

## Conclusion:

This machine learning pipeline for protein structure prediction effectively preprocesses the dataset, handles missing values, encodes categorical data, detects and removes outliers, and optimizes model performance through hyperparameter tuning. The logistic regression model, along with additional classifiers, provides robust and reliable predictions, demonstrating the efficacy of the approach. The final SVM model is serialized for future use, ensuring the practicality of the solution in real-world applications. This comprehensive approach sets a strong foundation for further refinement and application in bioinformatics.