NEXUS INFO

Project 1: Stock Market Prediction

Project Requirements:

1. Exploratory Data Analysis (EDA):

- Perform EDA on the stock market dataset to understand its structure and characteristics.
- Visualize key statistics and trends in stock prices.

2. Predictive Modeling:

- Utilize machine learning techniques to predict stock prices.
- Implement regression models to forecast future stock values.

3. Documentation:

- Document your approach, methodologies, and insights gained from the stock market dataset.
- Provide clear explanations for the chosen predictive models.

Approach and Methodologies

Exploratory Data Analysis (EDA):

Dataset Overview:

I began by loading the stock market dataset, which contains information about stock prices, including the opening, high, low, closing prices, adjusted close prices, and trading volume. The dataset also includes the date column as the index.

Data Cleaning:

After loading the dataset, I checked for missing values using isnull().sum() and found no missing values in the dataset.

Visualization:

To gain insights into the distribution and variability of stock prices, I visualized histograms and box plots for the numerical columns (open, high, low, close, adjclose, volume). These visualizations provided a clear understanding of the data distribution and potential outliers.

Predictive Modelling:

Target Creation:

To create a target variable for classification, I generated a new column named tomorrow, which represents the opening stock price of the next trading day. This was achieved by shifting the open column by one day. Missing values in the tomorrow column were filled with the mode value.

Model Selection and Training:

1. *Random Forest Classifier:*
    - I utilized the Random Forest classifier from scikit-learn to predict whether the opening stock price of the next trading day (tomorrow) would be higher than the current day's opening price (open).
    - The model was trained on features (open, high, low, close, adjclose, volume), and the target variable (target).
    - Achieved an accuracy of 0.76 on the testing set.

2. *Logistic Regression:*
    - I employed Logistic Regression to perform the same binary classification task.
    - Trained the model using the same features and target variable.
    - Attained an accuracy of 0.55 on the testing set.

3. *XGBoost Classifier:*
    - Implemented the XGBoost classifier for predicting the binary target variable.
    - Trained the model on a subset of features (open, high, low, close, adjclose, volume), considering the relevance of these features in stock price prediction.
    - Obtained an accuracy of 0.74 on the testing set.

4. *Support Vector Classifier (SVC):*
    - Utilized the Support Vector Classifier to classify whether the opening stock price of the next trading day would be higher than the current day's opening price.
    - Trained the model on features (open, high, low, close, adjclose, volume).
    - Achieved an accuracy of 0.55 on the testing set.

Project 2: Breast Cancer Prediction

Project Requirements:

1. Data Preprocessing:

- Clean and preprocess the Breast Cancer Wisconsin (Diagnostic) dataset.
- Handle missing values, outliers, and any other inconsistencies in the data.

2. Feature Selection and Engineering:

- Identify relevant features for breast cancer prediction.
- Create new features or transformations that might enhance the predictive model's performance.

3. Machine Learning Model (SVM):

- Implement a Support Vector Machine (SVM) model for classifying tumors into malignant or benign.
- Train and evaluate the model on the Breast Cancer dataset.

4. Documentation:

- Document your data preprocessing, feature selection, and machine learning model implementation.
- Explain the model's performance metrics and any challenges faced during the analysis.

Approach and Methodologies

Data Preprocessing:

Cleaning and Preprocessing:

- The Breast Cancer Wisconsin (Diagnostic) dataset was loaded and inspected for any missing values using **isnull().sum()**. No missing values were found.
- The column "Unnamed: 32" was dropped as it contained no useful information.
- The target column "diagnosis" was renamed to "target" for clarity.
- The target variable was encoded into binary labels: 'B' (benign) as 0 and 'M' (malignant) as 1.
- A count plot was created to visualize the distribution of the target variable.

Feature Selection and Engineering:

Outlier Detection:

- Outliers were detected using the Local Outlier Factor (LOF) algorithm.
- Outliers were visualized using a scatter plot and identified using a threshold outlier score.
- Outliers were removed from the feature dataset.

Dimensionality Reduction (PCA):

- Principal Component Analysis (PCA) was applied to reduce the dimensionality of the feature dataset.
- The optimal number of components was determined based on the cumulative explained variance ratio.
- The feature dataset was transformed using PCA with the optimal number of components.

Machine Learning Model (SVM):

Support Vector Machine (SVM):

- A Support Vector Machine (SVM) classifier was implemented for classifying tumors into malignant or benign.
- The dataset was split into training and testing sets.
- The SVM classifier was trained on the training data and evaluated on the testing data.
- The accuracy of the SVM model was calculated, and the classification report was generated.

Random Forest Classifier:

- The Random Forest classifier was applied for tumor classification.
- The dataset was split into training and testing sets.
- The Random Forest classifier was initialized and trained on the training data.
- Predictions were made on the testing data using the trained model.
- The accuracy of the Random Forest model was calculated.
- The classification report was generated to assess the model's performance.

XGBOOST Classifier:

- The XGBoost classifier was utilized for tumor classification.
- Similar to other models, the dataset was split into training and testing sets.
- The XGBoost classifier was initialized and trained on the training data.
- Predictions were made on the testing data using the trained XGBoost model.
- The accuracy of the XGBoost model was calculated.
- A classification report was generated to evaluate the model's performance.