Stock Prediction Using Regression

Project Overview

This project aims to predict stock closing prices using regression techniques. The primary dataset used is the NSE TATA GLOBAL dataset, which contains historical stock data for Tata Global Beverages Limited from the National Stock Exchange of India.

The project involves several key stages:

1. **Data Loading and Initial Exploration**: Understanding the dataset structure, features, and basic statistics.
2. **Exploratory Data Analysis (EDA)**: Visualizing trends, distributions, and potential outliers in the data.
3. **Feature Engineering and Scaling**: Selecting relevant features and scaling them to prepare for model training.
4. **Model Building**: Implementing and evaluating Linear Regression and Random Forest Regressor models.
5. **Cross-Validation**: Assessing model performance robustness.
6. **Hyperparameter Tuning**: Optimizing the Random Forest Regressor using Grid Search and Randomized Search techniques.
7. **Model Evaluation**: Measuring model performance using R-squared (R^2), Mean Squared Error (MSE), and Mean Absolute Error (MAE).
8. **Feature Importance**: Interpreting the model to understand which features contribute most to the predictions.

Dataset

- **Name**: NSE TATA GLOBAL (Tata Global Beverages Limited)
- **Source**: Tata Global Beverages Limited, National Stock Exchange of India
- **File**: `NSE-Tata-Global-Beverages-Limited.csv`

The dataset includes the following key columns:

- `Date`: The date of the stock record.
- `Open`: The opening price of the stock on that day.
- `High`: The highest price of the stock on that day.
- `Low`: The lowest price of the stock on that day.
- `Last`: The last traded price.
- `Close`: The closing price of the stock on that day (this is the target variable for prediction).
- `Total Trade Quantity`: The total volume of shares traded on that day.
- `Turnover (Lacs)`: The turnover in lakhs.

Methodology

1. Data Loading and Initial Exploration

The dataset was loaded using pandas. Initial exploration involved:

- Displaying the first few rows (`df.head()`).
- Getting a summary of the data types and non-null values (`df.info()`).
- Calculating descriptive statistics (`df.describe()`).

2. Exploratory Data Analysis (EDA)

Visualizations were created to understand the data better:

- **Closing Price Over Time**: A line plot showing the trend of the 'Close' price over the dates.
- **Volume Distribution**: A histogram showing the frequency distribution of 'Total Trade Quantity'.
- **Price Distribution and Outliers**: Boxplots for 'Open', 'High', 'Low', and 'Close' prices to identify distributions and potential outliers.

3. Feature Scaling

- **Features (X)**: 'Open', 'High', 'Low', 'Total Trade Quantity', 'Turnover (Lacs)'
- **Target (y)**: 'Close'
- The selected features were scaled using `StandardScaler` from scikit-learn to standardize their ranges, which helps in improving the performance of certain machine learning algorithms.

4. Cross-Validation Techniques

Cross-validation was performed on a `LinearRegression` model to assess its generalization ability.

- **Technique**: 5-fold cross-validation.
- **Scoring Metric**: R^2 score.
- The average R^2 score across the folds was reported.

5. Model Training and Evaluation

a. Linear Regression

- The scaled data was split into training (80%) and testing (20%) sets.
- A `LinearRegression` model was trained on the training data.
- Predictions were made on the test set.
- **Evaluation Metrics**:
  - R^2 Score
  - Mean Squared Error (MSE)
  - Mean Absolute Error (MAE)

b. Random Forest Regressor with Hyperparameter Tuning

i. Grid Search

- A `RandomForestRegressor` was chosen for its potential to capture non-linear relationships.
- **Hyperparameters Tuned**:
  - `n_estimators`: [50, 100, 150]
  - `max_depth`: [None, 10, 20]
- `GridSearchCV` with 5-fold cross-validation and R^2 scoring was used to find the best hyperparameter combination.
- The best estimator from Grid Search was then evaluated on the test set using R^2, MSE, and MAE.

ii. Randomized Search

- To explore a wider range of hyperparameters more efficiently, `RandomizedSearchCV` was employed.
- **Hyperparameters Tuned (distributions)**:

    - `n_estimators`: `randint(50, 200)`
    - `max_depth`: `randint(5, 30)`
- `RandomizedSearchCV` with 10 iterations, 5-fold cross-validation, and R^2 scoring was used.
- The best estimator from Randomized Search was evaluated on the test set using R^2, MSE, and MAE. This model generally showed the best performance among the tested configurations.

6. Model Interpretation and Feature Importance

- The feature importances were extracted from the best `RandomForestRegressor` model (obtained via Randomized Search).
- A bar plot visualized the importance of each feature ('Open', 'High', 'Low', 'Total Trade Quantity', 'Turnover (Lacs)').
- **Conclusion from Feature Importance**: The analysis suggested that features like 'Turnover (Lacs)' and 'Total Trade Quantity' had lower importance compared to 'Open', 'High', and 'Low' prices. This implies that a model might still perform well even if these less important features were removed, potentially simplifying the model.

Models Used

- Linear Regression
- Random Forest Regressor

Evaluation Metrics

The performance of the models was evaluated using:

- **R^2 Score (Coefficient of Determination)**: Indicates the proportion of the variance in the dependent variable that is predictable from the independent variables.
- **Mean Squared Error (MSE)**: Measures the average of the squares of the errors.
- **Mean Absolute Error (MAE)**: Measures the average of the absolute errors.

Results Summary (Illustrative - actual values are in the notebook)

Model Configuration	R^2 Score	MSE	MAE
Linear Regression	0.99942	1.61238	0.87037
Random Forest (Grid Search)	0.99980	0.53847	0.52718
Random Forest (Randomized Search)	0.99983	0.47173	0.48442

The Randomized Search for the Random Forest Regressor typically yielded the best performance metrics.

Conclusion

The project successfully demonstrated the process of building and evaluating regression models for stock price prediction. The Random Forest Regressor, particularly after hyperparameter tuning with Randomized Search, showed strong

predictive performance. Feature importance analysis provided insights into which factors were most influential in predicting the closing price, suggesting that 'Open', 'High', and 'Low' prices are key drivers.

How to Run

1. Ensure you have Python installed.
2. Install the required libraries:

   ```
   pip install pandas scikit-learn matplotlib seaborn
   ```

3. Download the dataset `NSE-Tata-Global-Beverages-Limited.csv` and place it in the same directory as the notebook `regression.ipynb`.
4. Open and run the `regression.ipynb` notebook in a Jupyter environment.