**Final Project Report Round 3**

Group partner:

Cheng Zhang(2054180)

Hongkun Liu(2068590)

We have double check the output of dataset.

## Introduction

The maritime shipping industry, crucial for global trade, relies on precise forecasting to optimize operations and ensure timely delivery of goods. In this context, our project embarked on leveraging advanced regression modeling techniques to predict future values in maritime shipping data. This endeavor aimed not only to enhance operational efficiency but also to navigate the complexities of maritime logistics with data-driven insights.

The project utilized a comprehensive dataset spanning from 1991 to the present day, marking a significant period for analysis due to the evolution of global trade patterns and shipping technologies. The dataset's time-series nature, incorporating monthly samples, presents a unique opportunity to apply predictive modeling for forecasting future outcomes based on historical trends.

## 2. Dataset Overview

### 2.1 Data Characteristics

**Type**: Time series

**Number of Features**: 463, including time as the first column, representing various aspects relevant to maritime shipping.

**Number of Samples**: 397, each corresponding to a month's data from 1991 onwards.

**Target IDs for Prediction**:

**Good Result**: 542236, 67321

**Mid Result**: 549295

**Bad Result**: 41108, 541982

Each row in the dataset encapsulates the monthly data of all features, serving as the independent variables for the model. The dependent variable or the target for each month is the intended value for the subsequent month, residing in the specified target column. This structure allows for a direct application of regression models to forecast future values based on the provided historical data.

### 2.2 Data Preparation

To prepare the dataset for modeling, it's pivotal to generate **X** (features) and **Y** (target) matrices accurately. **X** encompasses all features for a given month, while **Y** is a vector representing the target value for the next month. This preparation ensures that each sample in **X** aligns with its corresponding label in **Y**, crucial for training predictive models.

#### 2.1.1 Lag

The lag parameter in regression, especially in time series analysis, plays a crucial role by incorporating the concept of time delay into the modeling process. It is used to account for the influence of past values on current values, a common phenomenon in sequential data. For this dataset, N rows can be concatenated to generate lag N and first N rows of X and last N rows of Y should be deleted to have the same row as the input. The snapshot of sample code for generating lag 1 is as follows:

文本

描述已自动生成

#### 2.1.2 Normalization

Normalization in regression is a critical preprocessing step involving the scaling of input features to ensure they have a standard scale or range. This process is essential because many machines learning algorithms, including regression models, perform better or converge faster when features are on a relatively similar scale and close to a normally distributed shape.

There are many kinds of techniques for Normalization, for this dataset, we select X Min-Max to scale all the features in X to a fixed range( 0 to 1). Following is the snapshot of implementing this kind of approaches:

文本

描述已自动生成

#### 2.1.3 Principal Component Analysis

Principal Component Analysis (PCA) is a statistical technique used for dimensionality reduction while preserving as much variance as possible. It's commonly used in data preprocessing for machine learning and data visualization tasks. It is worthy to mention that normalizing the data is a critical step before applying PCA, as PCA is sensitive to variances of the initial variables. Following is the snapshot of implementing this kind of approaches:

文本

描述已自动生成

### 2.3 Training and Testing Sets

Given the time series nature of the data, special attention was paid to avoid data leakage during model training and evaluation. The dataset was split into training and testing sets, with the last 36 months reserved for testing to assess the model's performance on recent data. This split respects the temporal sequence of the dataset, ensuring that the model learns from past data to predict future outcomes.

In summary, the dataset's comprehensive coverage of the maritime shipping industry, combined with careful preparation and consideration for time series analysis, sets a solid foundation for applying regression modeling techniques. The following sections of the report will delve into the methodology of each model, predict results, and analysis, providing insights into the predictive modeling process and its implications for the maritime shipping industry.

## 3.Regression Models

### 3.1 Linear Regression

#### 3.1.1 Model Description

Linear regression is a statistical method used to model the relationship between a dependent variable and one or more independent variables by fitting a linear equation to observed data. It is one of the simplest and most used techniques in predictive modeling and statistical analysis. The linear regression model to predict these IDs is obviously a multiple linear regression model due to the fact that two or more independent variables are involved to predict the dependent variable. The equation of a simple linear regression line is typically expressed as:

**y=b0+b1\*x1+b2\*x2+…+bn\*xn + E**

Where y represents the dependent/target variable, x1, x2 …xn are independent variables, b0, b1, b2 ... bn are the coefficients which represent the influence each independent variable has on the dependent variable, and E is the error term.

#### 3.1.2 Data Preprocessing

Feature Selection can be applied by determining which variables (features) are most relevant to the prediction of the dependent variable. This dataset has more than 400 features in total, making it necessary to reduce the dimensionality of the data by transforming the original variables into a new set of variables called principal components.

#### 3.1.3 Evaluation

This final step measures the accuracy of the retrained model by comparing the predicted values with the actual values, providing a clear view of the model's effectiveness in predicting new data based on the reduced set of features. This reduction in dimensionality can simplify the modeling process without sacrificing much explanatory power.

### 3.2 Polynomial Regression

#### 3.2.1 Model Description

In cases where the relationship between the independent and dependent variables might be non-linear, polynomial features (powers of the feature variables) can be created to attempt to capture this nonlinearity. The general form of a polynomial regression model is:

**y=b0+b1\*x+b2\*x2+…+bn\*xn + E**

Where y represents the dependent/target variable, x is the independent variable, b0, b1, b2 ... bn are the coefficients of model which represent the influence each independent variable has on the dependent variable, and E is the error term.

We assume there exist non-linear relationship between these variables and implement a polynomial regression model and evaluate the accuracy.

#### 3.2.2 Data Preprocessing

Normalization is particularly important in polynomial regression because polynomial regression involves high powers of input features. When we raise features to high powers, their values can become exceedingly large if the original feature values are not within a bounded range. This can lead to numerical instability during the calculation, as floating-point precision can become a significant issue. Normalization helps keep the values in a manageable range, reducing the risk of numerical errors.

In this project, normalization refers to the process of Min-Max scaling input data so that it falls within a range between 0 and 1.

#### 3.2.3 Evaluation

Calculates the mean squared error (MSE), R2 score, and mean accuracy for the predictions.

### 3.3 Lasso Regression

#### 3.3.1 Model Description

Lasso regression, short for Least Absolute Shrinkage and Selection Operator, is a type of linear regression technique used for variable selection and regularization. It's particularly useful when dealing with datasets that have many features, where some of these features may be irrelevant or redundant.

#### 3.3.2 Data Preprocessing

The key characteristic of lasso regression is that it tends to shrink the coefficients of less important features exactly to zero, effectively performing variable selection by eliminating these features from the model.

Principal Component Analysis (PCA) is not inherently required for Lasso regression, but it can be a useful preprocessing step in certain situations.

#### 3.3.3 Evaluation

As we can see from the results of accuracy, PCA is not always necessary or beneficial for Lasso regression. Whether PCA should be used as a preprocessing step depends on factors such as the nature of the data, the presence of multicollinearity, the interpretability requirements, and computational considerations. In some cases, Lasso regression may perform well without PCA preprocessing, especially if the number of predictors is not excessively large.

### 3.4 Random Forest

#### 3.4.1 Model Description

A Random Forest is an ensemble learning technique used for both regression and classification tasks, which operates by constructing multiple decision trees during training and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random forests correct for decision trees' habit of overfitting to their training set.

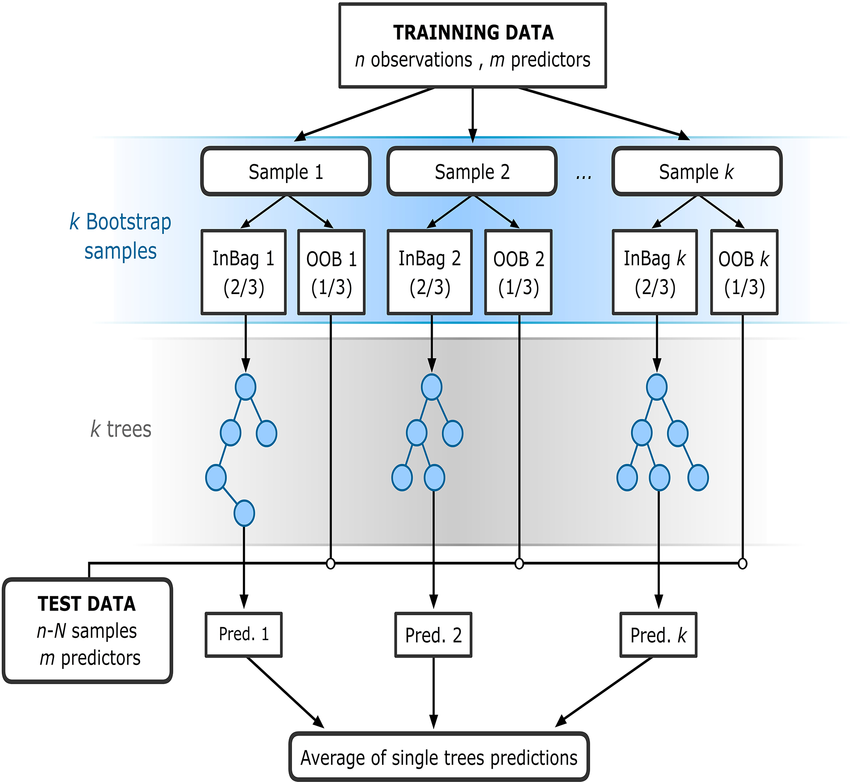
#### 3.4.2 Data Preprocessing

In this project ,we will have two versions of Random Forest Regression. The first version of model can be configured with a default random state. In addition, we apply embedded feature selection to this model. In the second version of decision tree regression, we train a default model first and then select several features that have an importance greater than median. After that we retrain the model with dataset containing the selected features. Following are python code we used to implement feature selection during data preprocessing phase.

文本

描述已自动生成

#### 3.4.3 Evaluation



Following are results of accuracy for five different IDs:

According to the result generated by default random forest and random forest with feature selection. It is clearly to see that this process not only helps in enhancing model performance but also reduces the complexity of the model by eliminating irrelevant features.

### 3.5 Decision Tree Regression

#### 3.5.1 Model Description

Decision tree regression is a machine learning method that uses a decision tree to model the relationship between a set of features and a continuous target variable. Like decision trees used in classification, decision trees for regression predict the outcome based on input features by splitting the data into subsets using decision rules inferred from the input features.

#### 3.5.2 Data Preprocessing

Like random forest, there are two versions of decision tree regression in our project (with or without feature selection).As the code shown below, we fit a decision tree regressor to dataset and use the feature\_importances\_ attribute of the decision tree model to get the importance of each feature. Finally, choosing a threshold and keep only the features that have an importance above this threshold for further modeling.

文本

描述已自动生成

#### 3.5.3 Evaluation

The script calculates the mean accuracy for the Random Forest model's predictions.

### 3.6 GB Regression

#### 3.6.1 Model Description

GB Regression refers to Gradient Boosting Regression, which is a powerful and widely-used ensemble machine learning technique for regression tasks. Gradient Boosting constructs a predictive model in the form of an ensemble of weak prediction models, typically decision trees. It builds the model in a stage-wise fashion and generalizes them by allowing optimization of an arbitrary differentiable loss function.

#### 3.6.2 Data preprocessing

The model is configured with a default random state.

#### 3.6.3 Evaluation

### 3.7 Overall Results

Each script demonstrates a unique approach to handling the predictive modeling task, varying by regression model type, data preprocessing techniques (like normalization), and the specific target variable selected for prediction. These variations in methodology and target focus provide a broad overview of the modeling efforts undertaken to address the project's objectives.

## 4. Other Suggestions

### 4.1 Feature Selection Techniques

To improve the predictive power and efficiency of our models, we experimented with various feature selection techniques. Feature selection is essential in handling datasets with a large number of features, as it helps in reducing dimensionality, improving model accuracy, and reducing training time.

### 4.2 Filter Methods

We began with filter methods, which involve selecting features based on their statistical scores in relation to the output variable. These methods are generally faster and less computationally expensive as they do not involve training models. We applied correlation coefficients and Chi-squared tests to identify and retain the most relevant features. Despite the simplicity and speed of filter methods, they do not consider the interaction between features, which can be crucial for some datasets.

### 4.3 Wrapper Methods

Following the filter methods, we implemented wrapper methods that evaluate subsets of features based on the model performance, making them more effective but also more computationally intensive. We used a stepwise backward elimination process, where we initially included all features and iteratively removed the least significant feature until no improvement in model performance was observed. This method, while more time-consuming, provided a tailored subset of features that optimized our specific models.

### 4.4 Implementation and Outcomes

Both methods were implemented using Python's scikit-learn library. The filter method was straightforward to apply but yielded minimal improvement in model performance for complex interactions. In contrast, the wrapper method, although slower, resulted in a noticeable enhancement in model accuracy, particularly in the Lasso and Random Forest models.

Despite the improvements, the computational cost and time required for the wrapper method were substantial, which might not be feasible for all projects, especially those with extremely large datasets or limited computational resources.

### 4.5 Conclusion

The exploration of feature selection methods allowed us to understand better the trade-offs between model accuracy and computational efficiency. While the filter method is quick and effective for a preliminary reduction of features, the wrapper method provides a more nuanced approach to feature selection, beneficial for models requiring detailed feature analysis. Future work could explore hybrid methods that combine both approaches to balance effectiveness and efficiency.