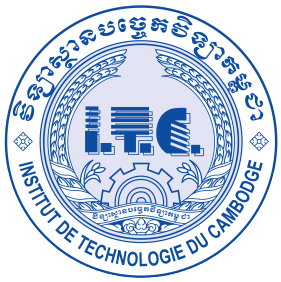


Prediction of Book Price

Programming for Data Science
2022-2023

3rd year Engineer's Degree in Data Science
Department of Applied Mathematics and Statistics
Institute of Technology of Cambodia

Group 2 members:

HONG Kimmeng	e20200559
KHON Yin Sakal	e20200425
KEO Vonmonyroth	e20200759
KHEANG Tongheang	e20200472
EAB Pisey	e20200994
HONG Kimleng	e20200766

Submission Date: 15 July, 2023

Lecturers:
Mr. Chan Sophal

Abstract

Background: The book industry has witnessed significant growth in recent years, and accurate prediction of book prices can be valuable for publishers, sellers, and consumers. This project aims to develop a predictive model for book prices based on various features such as author, category, publication year, book format

Objectives: The main objective of this project is to build regression models that can accurately predict book prices. Additionally, we aim to explore the relationships between book features and prices, identify influential factors, and assess the performance of different regression models.

Methodology: The methodology involves data preprocessing, data exploration and visualization, and the building of regression models. Multiple Linear Regression with Ridge and Lasso, Random Forest Regression, Neural Network Regression, Gradient Boosting Regression, and XGBoost Regression are utilized. The models are evaluated using metrics such as Mean Squared Error, Root Mean Squared Error, R-squared Score, Adjusted R-squared Score, and Mean Absolute Error. Techniques like GridSearchCV, cross-validation, learning curves, and train-test splits are employed to fine-tune the models and assess their generalization capabilities.

Results: The results of this project include the performance evaluation of the regression models on the test dataset, comparison of their predictive accuracy, and identification of influential factors affecting book prices. Insights into the relationships between book features and prices are provided, aiding in understanding the factors that contribute to pricing variations.

Conclusions: Through this project, we have developed and evaluated multiple regression models for book price prediction. The results demonstrate the efficacy of these models in accurately predicting book prices. Additionally, the insights gained from the analysis can provide valuable guidance for pricing strategies in the book industry. The developed models can be further refined and applied in real-world scenarios to assist publishers, sellers, and consumers in making informed decisions.

Key Words: book price, regression analysis, predictive modeling

1 Introduction

The book industry has witnessed significant growth in recent years, driven by the increasing popularity of reading and the availability of books in various formats. For publishers, sellers, and consumers, accurate prediction of book prices is crucial for making informed decisions related to pricing, marketing, and purchasing. However, determining the optimal price for a book can be challenging due to various factors such as author reputation, genre, publication year, and book condition. The aim of this project is to develop a predictive model for book prices using machine learning techniques. By leveraging the power of regression analysis, we can uncover the underlying relationships between book features and their corresponding prices. This model can assist publishers in setting competitive prices, guide sellers in pricing their inventory effectively, and help consumers make informed purchasing decisions.

In this project, we will explore and analyze a comprehensive dataset that includes various attributes of books such as author information, genre, publication details, and condition. Our goal is to build accurate regression models that can predict book prices based on these features. We will evaluate the performance of different regression algorithms and identify the most effective model for book price prediction. The findings from this project have practical implications for the book industry, providing valuable insights into the factors that influence book prices. By understanding these relationships, publishers and sellers can optimize their pricing strategies, while consumers can make informed choices when purchasing books. The remainder of this report is organized as follows: the Methodology section outlines the steps involved in data preprocessing, model building, and evaluation. The Results section presents the performance evaluation and analysis of the regression models. Finally, the Conclusion section summarizes the findings, discusses the implications, and provides recommendations for future work. By developing an accurate book price prediction model, we aim to contribute to the growth and efficiency of the book market, benefiting both industry stakeholders and book enthusiasts.

2 Data Description

We collected data from bookshop.org website of the popular book category. There are 2493 rows and 8 columns. Below are the description of each feature.

- Name: the title of the book which is a string
- Prices: the price for each book which is a float data type
- Page: the number of pages for each book which is written in from of integer
- Format: the layout type of book and it's also string data
- Pub Date: publication date
- Publisher: publisher of each book
- Dimension: the dimension of book which contain length, width, height and weight information of the book
- Categories: the categories of each book which is also a string data type

3 Methodology

3.1 Data Collection

We have scrapped the data from website bookshop.org/categories/m/popular-books uisng BeautifulSoup and undetected chromedriver tool.

3.2 Data Preprocessing

Data preprocessing is a crucial step in any data science project. It involves cleaning, transforming, and preparing the data for analysis. In this project, the following preprocessing techniques were applied:

3.2.1 Data Cleaning

- Removed duplicate entries: Duplicate entries can introduce bias and distort the analysis. Therefore, any duplicate records were identified and eliminated.
- Handling missing values: Missing data can affect the accuracy of the predictive model. Different strategies, such as imputation or removal, were employed to handle missing values based on the nature of the data.
- Outlier detection and treatment: Outliers can significantly impact the performance of a predictive model. Outliers were identified using boxplot. We used capping techniques which is the technique for replacing outliers with lower or upper bound of the boxplot.

3.2.2 Data Transformation

- Feature Engineering: This involves creating new feature or transforming the existing features to make them more suitable for the model. In this Data-set we have considered:
  - Removing language feature since almost all language of each collected book are written in English
  - Changing the format of publish date into day count since published.
  - Creating new feature, surface area, by multiplying length and width of the book.
  - Binning the Surface Area into Cover size type and Height into Thickness Type.
- Feature Encoding: It is a technique used to convert categorical variables into numerical labels, allowing the model to work with categorical data.
  - Target Encoding: Target encoding, also known as mean encoding or likelihood encoding, replaces each category or level of a categorical variable with the mean (or another statistical measure) of the target variable for that category. For our data-set, we chose the Target encoding to encode these categorical features (Format, Publisher, Author and Categories) since they are likely to have a relationship to the target variable (Price) and allow to encode the features in a way that preserves valuable information for predicting. One more reason for choosing these encoding technique rather than using One-hot encoding is to reduce the

high cardinality problem since there are large number of unique categories in our features.

- Ordinal Encoding : It is used to convert the categories variable with an inherent order or ranking as numerical. We used this encoding technique for Cover Size Type (small, medium, large) and Thickness Type (thin, medium, thick).

- Feature Scaling: Since different features may have different scales, it is important to bring them to a similar scale. We used standardization technique to ensure the features have a similar range. However, we only scaled the data with some models based on their evaluation metrics compared to unscaled data.

3.3 Data Exploration and Visualization

Data exploration is a critical step to gain insights into the dataset and understand the relationships between variables. Visualization techniques were employed to visually represent the data and identify patterns, trends, and correlations. The following methods were utilized:

3.3.1 Descriptive Statistics

Descriptive statistics, such as mean, median, mode, standard deviation, and quartiles, were calculated to summarize the dataset. These statistics provide a basic understanding of the data distribution and can help identify potential issues or interesting characteristics.

3.3.2 Univariate Analysis

Univariate analysis was conducted on individual features to explore their distributions. Histograms, box plots, and density plots were used to visualize the distribution of numerical variables. Bar charts and pie charts were used for categorical variables to understand their frequencies and proportions.

3.3.3 Bivariate Analysis

Bivariate analysis was performed to explore relationships between pairs of variables. Scatter plots or line plots were used to visualize the relationships and identify any correlations or trends.

3.3.4 Multivariate Analysis

Multivariate analysis was conducted to explore relationships between three or more variables. Techniques such as heatmap and pairplot were employed to identify complex patterns and dependencies.

3.4 Model Building

In the book price prediction project, several regression models were built to predict book prices based on the available features. The following models were employed:

3.4.1 Multiple Linear Regression

Multiple Linear Regression is a widely used regression technique that assumes a linear relationship between the dependent variable and multiple independent variables. In this model, the book price was predicted based on various features. To handle multicollinearity and perform feature selection, both Ridge and Lasso regularization techniques were incorporated within the Multiple Linear Regression framework. Ridge Regression added a penalty term to control model complexity and reduce overfitting, while Lasso Regression performed variable selection by forcing some coefficients to be exactly zero.

$$\text{Ridge} : \min_{\beta} \left(\sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right)$$

$$\text{Lasso} : \min_{\beta} \left(\sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right)$$

3.4.2 Random Forest Regression

Random Forest Regression is an ensemble learning method that combines multiple decision trees to make predictions. It handles nonlinear relationships and interactions between features effectively. The Random Forest model was used to capture complex patterns in the data and provide robust predictions by averaging the outputs of multiple trees.

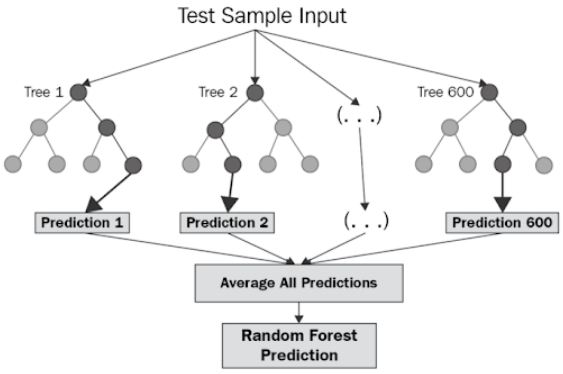


Figure 1: Random Forest Regression Process

3.4.3 Support Vector Regression

Support Vector Regression (SVR) is a regression technique that utilizes support vector machines (SVM) to model the relationships between the independent variables and the target variable. SVR aims to find the hyperplane that best fits the training data while minimizing the error within a specified margin. SVR is particularly effective in handling nonlinear relationships and can handle high-dimensional feature spaces efficiently.

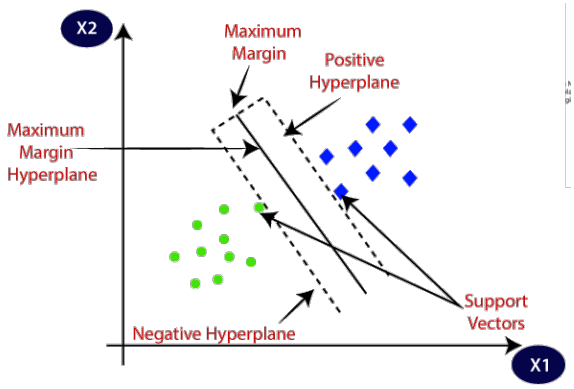


Figure 2: SVR Process

3.4.4 Neural Network Regression

Neural Network Regression is a powerful technique that can capture nonlinear relationships between features and the target variable. It consists of multiple layers of interconnected nodes (neurons) and can learn complex patterns in the data. A neural network model was trained using the book features as inputs and the book price as the output, with the aim of capturing intricate relationships and achieving high prediction accuracy.

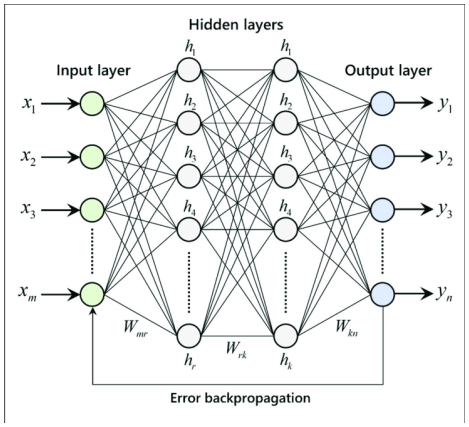


Figure 3: Neural Network Regression Process

3.4.5 Gradient Boosting Regression

Gradient Boosting Regression is another ensemble learning technique that combines multiple weak prediction models, typically decision trees, to form a strong predictive model. It builds the model in a stage-wise manner by sequentially adding new models that minimize the errors of the previous models. Gradient Boosting Regression was utilized to improve prediction accuracy by iteratively learning from the mistakes of the previous models.

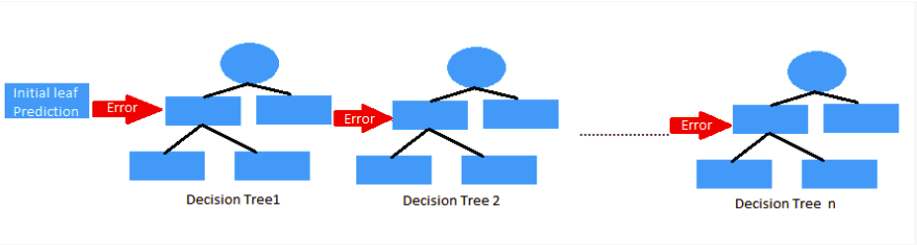


Figure 4: Gradient Boosting Regression Process

3.4.6 XGBoost Regression

XGBoost (Extreme Gradient Boosting) Regression is a highly efficient and popular gradient boosting framework that has demonstrated excellent performance in various machine learning tasks. It utilizes a gradient boosting algorithm that iteratively builds an ensemble of weak prediction models, such as decision trees, to minimize a specified loss function. XGBoost Regression was included in the model building process to leverage its powerful predictive capabilities and handle complex relationships in the book price prediction problem.

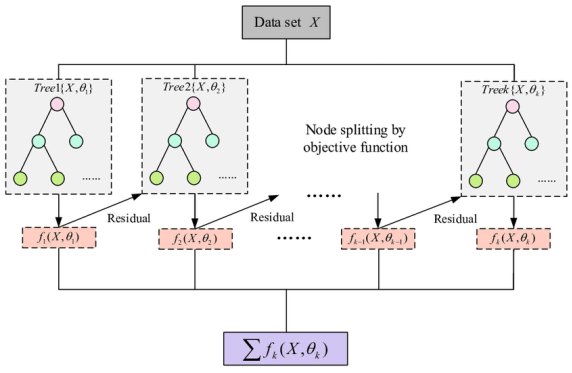


Figure 5: XGboost Regression Process

3.5 Model Evaluation

To assess the performance and effectiveness of the built regression models, the following evaluation metrics were employed:

- Mean Squared Error (MSE): MSE measures the average squared difference between the predicted book prices and the actual prices. It provides an overall measure of the model's accuracy, with lower values indicating better performance.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

- Root Mean Squared Error (RMSE): RMSE is the square root of the MSE and is commonly used to measure the average prediction error in the same units as the target variable. It provides a more interpretable measure of the model's performance.

$$RMSE = \sqrt{MSE}$$

- Mean Absolute Error (MAE): MAE measures the average absolute difference between the predicted book prices and the actual prices. It provides a robust measure of the model's performance that is less sensitive to outliers.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

- R-squared (R^2) Score: R-squared represents the proportion of the variance in the target variable that is predictable from the independent variables. It ranges from 0 to 1, with higher values indicating a better fit of the model to the data.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

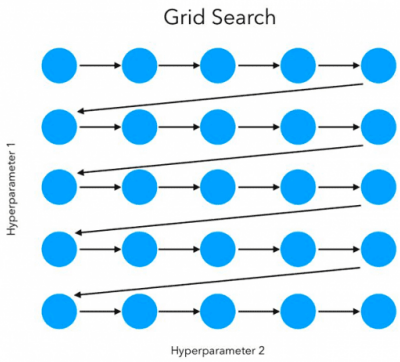
- Adjusted R-squared Score: Adjusted R-squared is a modified version of the R-squared score that takes into account the number of predictors in the model. It penalizes the addition of unnecessary predictors and provides a more reliable measure of the model's fit to the data.

$$Adjusted\ R^2 = 1 - \left(1 - R^2\right)\frac{n-1}{n-p-1}$$

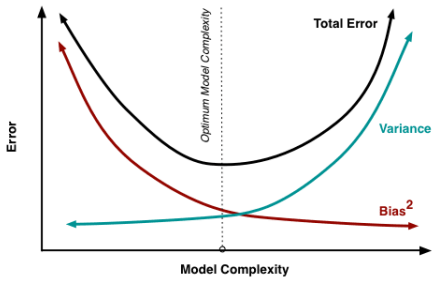
These evaluation metrics were used to compare and select the best-performing model for book price prediction. Additionally, techniques such as cross-validation and train-test splits were employed to assess the models' generalization capabilities and ensure reliable performance on unseen data. The chosen model was then used to make book price predictions on a separate test dataset to evaluate its real-world performance.

In addition to the model building phase, various techniques were employed to evaluate and fine-tune the performance of the regression models. These techniques included:

- Train-Test Split: A train-test split is a common practice in machine learning to evaluate model performance on unseen data. The dataset is divided into a training set and a separate test set. The training set is used to train the model, while the test set is used to evaluate its performance on new, unseen instances. Train-test split was performed to assess how well the models generalize to unseen data and provide an estimate of their real-world performance. We use 80:20 ratio train-split test in all of our models.
- GridSearchCV: GridSearchCV is a method for systematically evaluating different combinations of hyperparameters for a given model. By exhaustively searching through a predefined parameter grid, it helps identify the optimal hyperparameters that yield the best performance. GridSearchCV was used to fine-tune the hyperparameters of the regression models, such as the regularization strength in Ridge and Lasso Regression or the learning rate in Gradient Boosting Regression.
- Learning Curves: Learning curves provide valuable insights into a model's performance as the training data size increases. They plot the model's performance (e.g., MSE or R-squared) against the number of training instances. Learning curves help assess whether the model suffers from high bias (underfitting) or high variance (overfitting) and can guide decisions on data requirements, model complexity, or regularization. Learning curves were generated to analyze the models' behavior with different training data sizes and identify any issues related to bias or variance.



((a)) Grid Search



((b)) Learning Curve

Figure 6

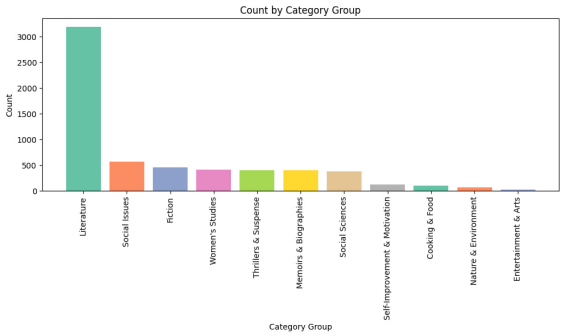
By utilizing these techniques, the project aimed to optimize the models' performance, identify the best hyperparameters, assess generalization capabilities, and understand the trade-off between bias and variance. These evaluation techniques provided valuable insights into the models' behavior and guided the selection of the final model for book price prediction.

4 Results

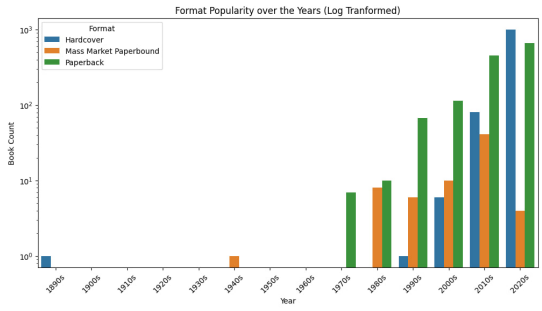
4.1 Exploratory Data Analysis

To study and observe the behaviour of data, attributes, relationship between attributes and target variable are graphically visualized, so that pattern of data is keenly studied and to explore the dependency of attribute so as to extract reliable features to develop a reliant model with robust features.

Exploratory Data Analysis (EDA) for categorical feature after encoding involves analyzing variables that represent qualitative characteristics or distinct categories. Involves the relationship with different categories. We can gain insights into the data, detect anomalies, and make informed decisions in subsequent stages of analysis and modeling.

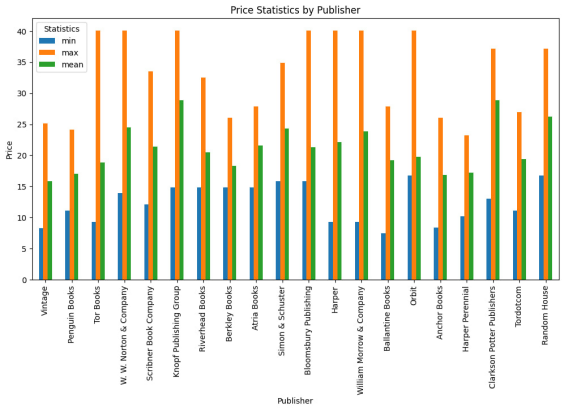


((a)) Group categories

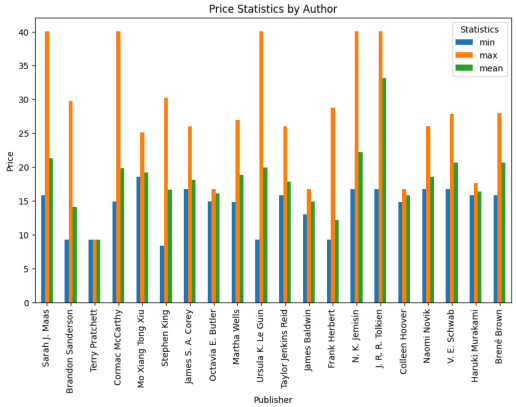


((b)) Format popularity over the year after transforming between 10 years

Figure 7: EDA on Format Feature and categories



((a)) Price Statistic by Publisher



((b)) Price Statistic by authors

Figure 8: EDA on Authors and Publisher Feature

According to the graph of figure(7.b) have been shown that: Format Hard cover can be higher than other format. In this case we can assume that hard cover is the most popular of Format feature. We can see that the bookshop mostly contain books in the last 50 years. Furthermore, paperback seem to be the most popular format book type from 1970 to 2010, while hardcover has increase it popularity sharply since 1990s until it become the most popular format book type in 2020. From figure(7.a), by categories have a lot of types in it. So we decided to combine small types to the categories as a group like figure(a). Most of the bookshop contains Literature books, while entertainment and art books have the least amount in the stores.

From Figure(8) : Belongs to above graph (a) and (b) have been shown that : Features (Author and Publisher) can assume that it can be affect to the Price Target variable. It is shown that Knopf Publishing Group is the most successfull publishing company by average, while Vintage company has the least selling book price overall. Moreover, it is also noticed that most books that are writting by author, J.R.R. Tolkien have the highest price in overall. In contrast, books written by author, Terry Pratchette, tends to have the least prices compared to other authors.

EDA for numerical features involves examining variables that represent quantitative measurements or continuous values. These features provide insights into the distribution, central tendency, variability and relationship within the dataset.

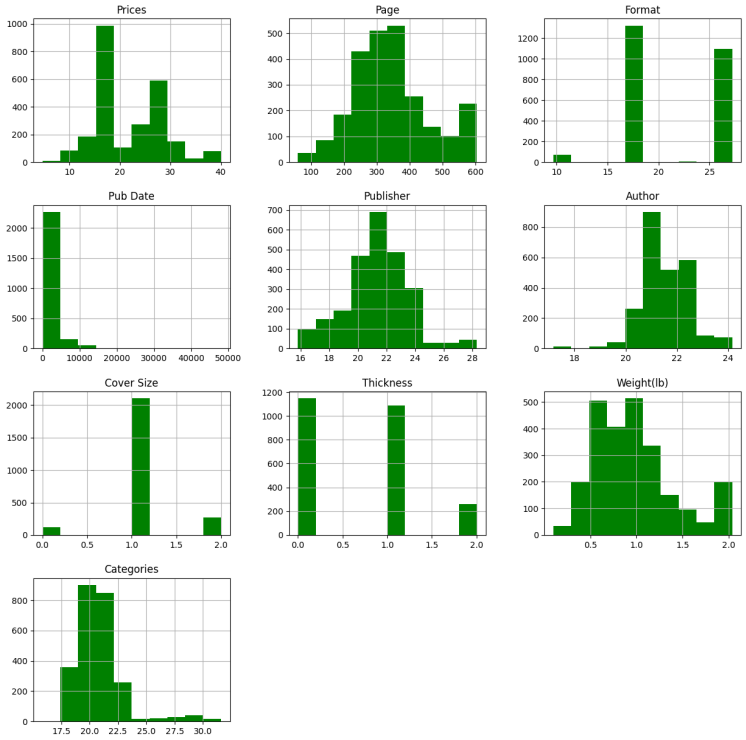


Figure 9: Histogram of Numerical variables

Histograms are commonly used to visualize the distribution of a numerical features like above. They display the frequency or count of data points falling within each predefined interval called bins. The x-axis represents the range of values, divided into bins, and y-axis represents the frequency or count. Histograms provide insights into shape of the distribution, such as whether it is symmetrical, skewed or multimodals.

4.2 Model Fitting

In this report, we delve into the realm of model fitting by testing various approaches, including Linear Regression, Random Forest, Support Vector Regression, MLP Regression, and Gradient Boosting. By exploring this ensemble of models, we aim to gain a comprehensive understanding of their strengths, weaknesses, and performance characteristics on the datasets to utltimately aid the selection of the most suitable model for the dataset prediciton tasks.

4.2.1 Multiple Linear Regression

Within this section, we embarked on an exploration of linear regression and its regularization counterparts, Lasso and Ridge techniques. Our objective was to assess the effectiveness of these approaches in predicting the target variable using the provided dataset.

After testing all three models, we concluded that Lasso regression yielded the best results. Therefore, in this section, we will narrow our focus solely on showcasing the performance and insights obtained from the Lasso regression model.

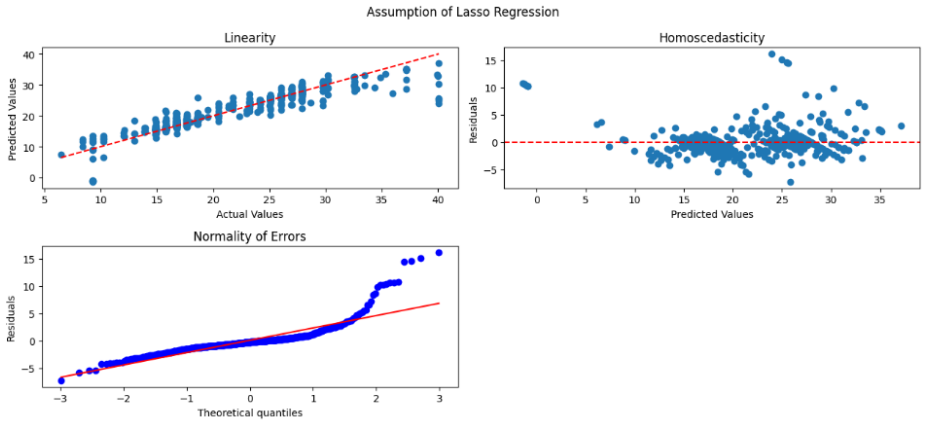


Figure 10: Linear Regression Model Diagnostic

4.2.2 Random Forest Regression

Next, we will employ the Random Forest Regressor algorithm by utilizing 100 decision trees, setting random state to 42, maximum depth of 10, max_features to auto minimum sample split and leaf of 10 and 4 respectively with bootstap enabled.

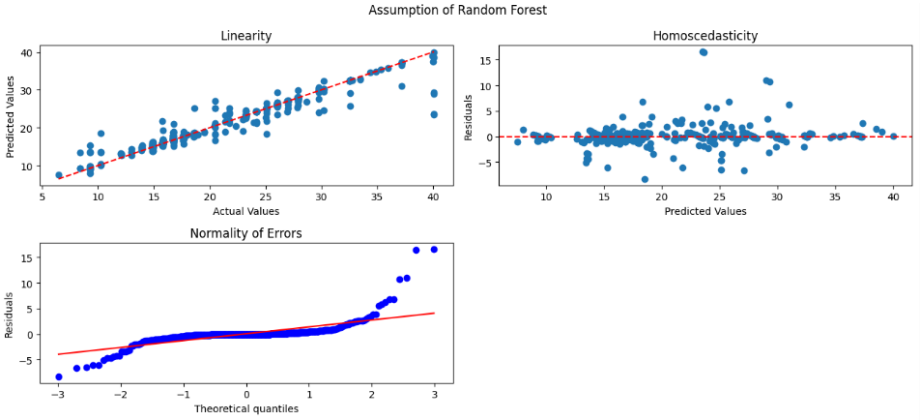


Figure 11: Random Forest Regression Model Diagnostic

4.2.3 Support Vector Regression

Continuing our analysis, we will now introduce Support Vector Regression as another powerful model for predicting our target variable. In this setup, we select the radial basis function kernel , the parameter gamma is set to 0.1, coef0 is set to 0.01 and C to 10 with epsilon of 0.5.

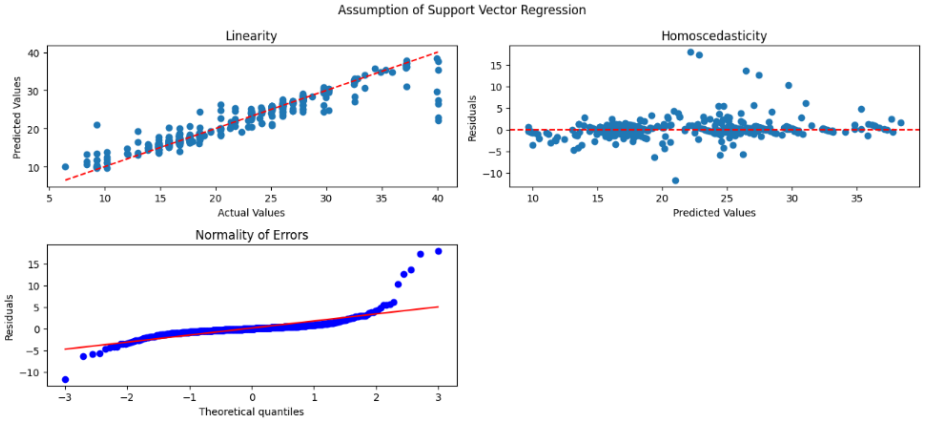


Figure 12: Support Vector Regression Model Diagnostic

4.2.4 Neural Network Regression

Moving forward, we will explore the Multilayer Perceptron Regressor. In this configuration, the model consist of two hidden layer with 100 and 50 neuron switch the activation function Hyperbolic Tangent and solver Adam Optimizer. The random state is also set to 42 and learning rate to constant.

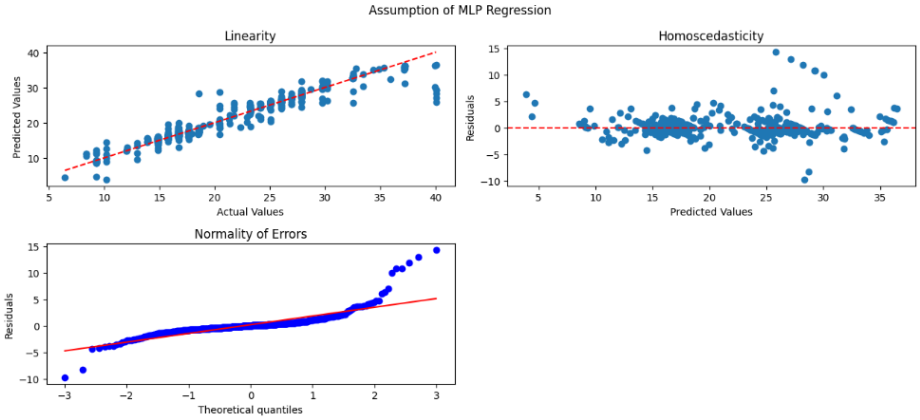


Figure 13: Mulitlayer Perception Regression Model Diagnostic

4.2.5 Gradient Boosting Regression

In the next analysis, we evaluated two popular gradient boosting algorithms: Extreme Gradient Boosting (XGBoost) and Gradient Boosting. After conducting experiments and comparing the performance of these models, we found that the Gradient Boosting algorithm achieved slightly better results compared to XGBoost on our specific dataset. As a result, we have decided to showcase the results solely for the Gradient Boosting model in this section.

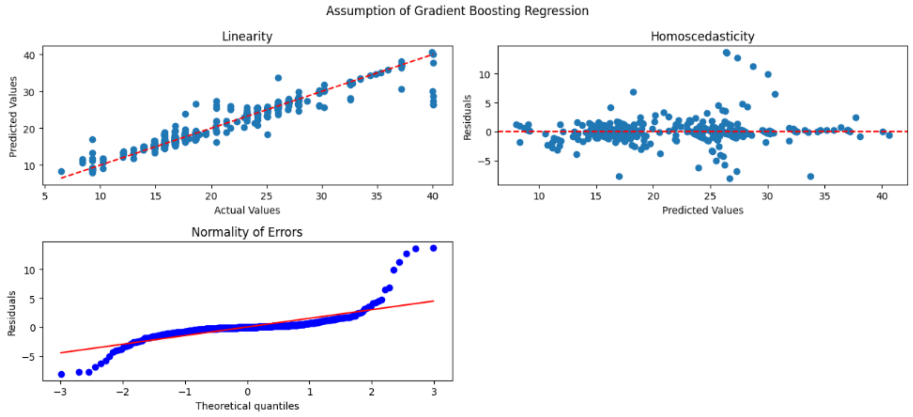


Figure 14: Gradient Boosting Regression Model Diagnostic

4.3 Model Comparison

In order to comprehensively evaluate and compare the performance of different models, We've created a table that presents an overview of their training and prediction results, along with the associated parameter settings. This table allows for a quick and concise comparison of the various models used in the analysis. This comparative analysis serves as a valuable reference, aiding in the selection of the most suitable model for the given dataset and problem domain.

Model	Train MSE	Train R2	Predict MSE	Predict R2	Parameter
Linear Regression	5.040807	0.883399	6.620211	0.854223	None
Ridge Regression	5.040807	0.883399	6.620117	0.854225	alpha = 0.1
Lasso Regression	5.096687	0.882107	6.637076	0.853852	alpha = 0.1
Random Forest Regression	0.457943	0.989407	3.056361	0.932699	n_estimators = 100, max_depth = 10, min_samples_split = 10, min_samples_leaf = 4, max_features = 'auto', bootstrap = True
Support Vector Regression	1.825713	0.957768	4.191129	0.907711	gamma = 0.1, coef0 = 0.01, C = 10, epsilon = 0.5
MultiLayer Perception	1.986323	0.954053	3.786791	0.916615	activation = 'tanh', solver = 'adam', learning_rate = 'constant', alpha = 0.0001
Gradient Boosting Regressor	0.611615	0.985852	3.445201	0.924136	learning_rate = 0.1, max_depth = 3, n_estimators = 300, alpha = 0.1
Extreme gradient Boosting	1.451656	0.966421	3.387203	0.925421	learning_rate = 0.1, max_depth = 3, n_estimators = 100, alpha = 0.1

Figure 15: Regression Model Perfomance Table

To further analyze the performance of the different models, we conducted a comparison of their learning curves. Learning curves provide insights into the relationship between model complexity and the amount of training data available. By plotting the training and validation error rates against the number of training instances, we can assess the models' generalization capabilities and identify any signs of underfitting or overfitting.
When interpreting the learning curve plots for negative mean squared error (MSE) and R2, we aim to observe specific trends that indicate the performance and generalization ability of the models.

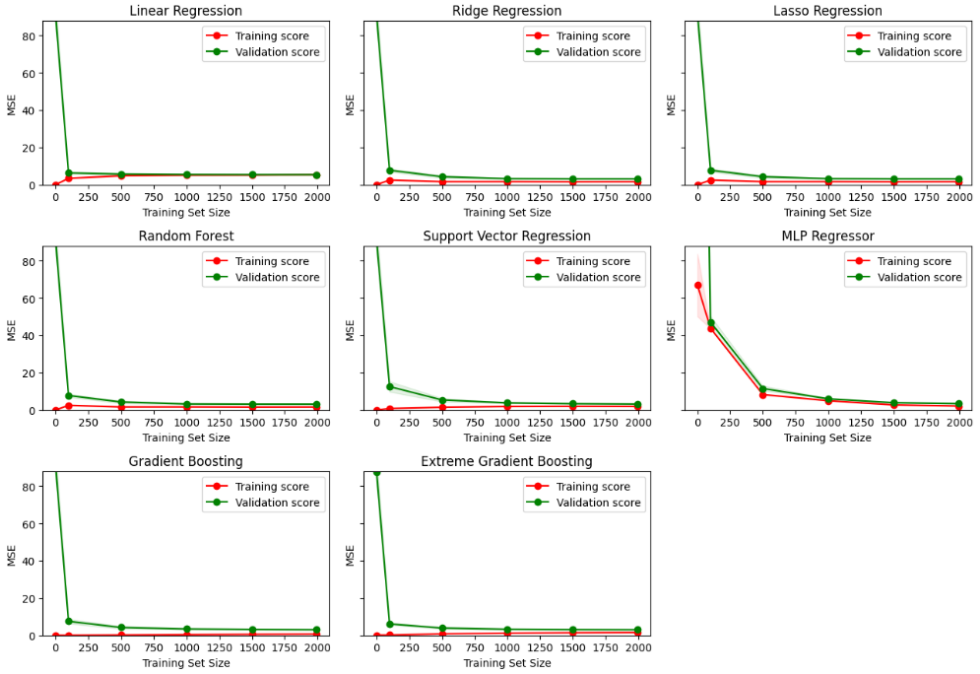


Figure 16: Negative MSE Learning Curve

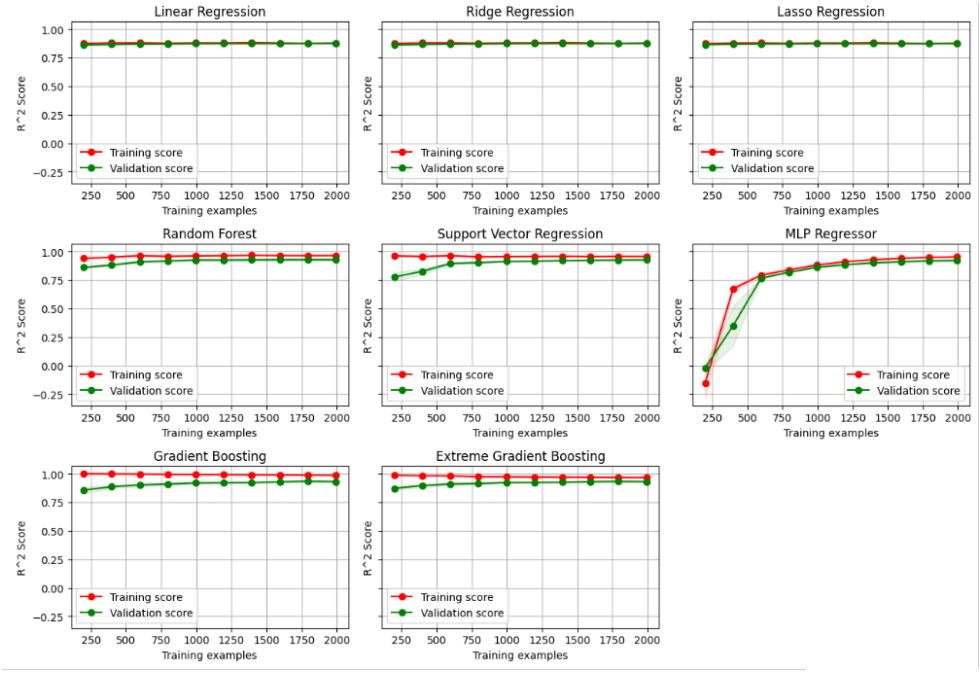


Figure 17: R2 Score Learning Curve

In summary, for negative MSE learning curves, we want to see decreasing trends, small gaps, and lower negative MSE values. For R2 learning curves, we look for increasing trends, small gaps, and higher R2 values. These trends indicate that the models are learning and performing well, both on the training data and on unseen validation data, ensuring their ability to make accurate predictions.

4.4 Construction of best fitting Model

Based on the learning curve analysis, We have decided to focus solely on Random Forest model for further analysis and reporting. By highlighting the performance of the chosen model, We aim to provide valuable insights into its predictive capabilities and its potential to achieve the desired outcome in future scenarios.

In our feature selection analysis for the random forest model, we investigated the impact of eliminating features with relatively lower importance scores. Surprisingly, we found that removing these features had minimal effect on the model's overall performance. Even though these features ranked lower in terms of importance, their exclusion did not result in a significant decrease in the model's predictive accuracy. This suggests that the random forest model is robust enough to handle variations in the importance of individual features.

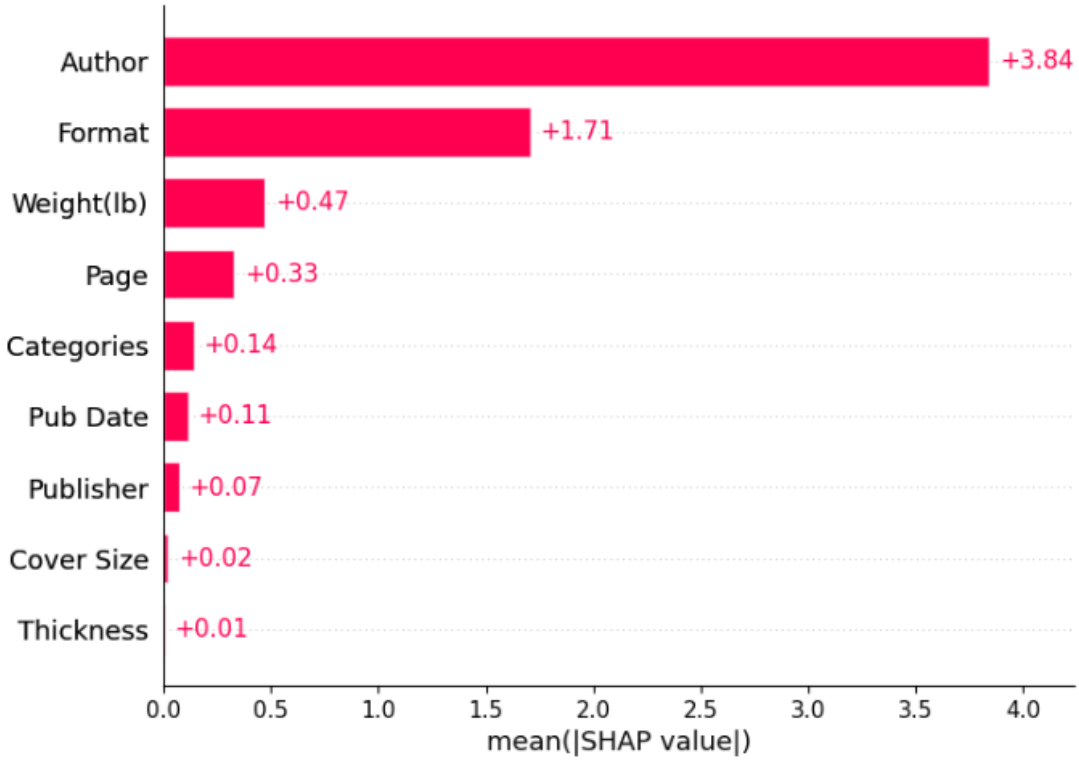


Figure 18: Feature Importance of Random Forest Regression

5 Conclusion

As a final result, we have chosen to include all available features in our final model. By doing so, we can leverage the collective knowledge embedded in the entire feature set, ensuring that no potentially valuable information is disregarded. This decision allows us to maintain the model's predictive power while preserving the integrity of the original feature set.

Acknowledgment

We would like to thank professor Chan Sophal who has guided and instructed us in this project.