International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 03 | March - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Predicting Library Book Demand Using Ensemble Learning: A Comparative Study of Random Forest, Gradient Boosting, and Decision Tree Regressors

Sarika Tanaji Maskar
Prof. Ramkrishna More Arts, Commerce and Science College (Autonomous) Akurdi Pradhikaran, Pune411044.
E-mail : sarikamaskar77@gmail.com

Prof. Ankush Dhamal
Prof. Ramkrishna More Arts, Commerce and Science College (Autonomous) Akurdi Pradhikaran, Pune411044.
E-mail : ankushdhamal01@gmail.com

Abstract

Accurate forecasting of library book demand is essential for efficient collection management, budgeting, and ensuring reader satisfaction. This study presents a comparative analysis of three ensemble learning algorithms—Random Forest, Gradient Boosting, and Decision Tree—for predicting the number of times a book will be borrowed in the next month. Using a synthetically generated dataset of 1000 library records enriched with engineered features (e.g., book age, availability ratio, borrowing rate, seasonal indicators), we evaluate model performance using R^2, RMSE, and MAE. Hyperparameter tuning was performed on the Random Forest model via grid search with cross-validation. Results indicate that Linear Regression achieved the highest test R^2 (0.5977), while the tuned Random Forest yielded an R^2 of 0.5430, demonstrating moderate predictive capability. Feature importance analysis revealed that popularity score, previous borrowings, and availability ratio are the most influential predictors. The study provides a practical framework for implementing data-driven demand forecasting in library settings and includes a saved model artifact for deployment.

Keywords — Library Book Demand Prediction, Ensemble Learning, Random Forest, Gradient Boosting, Decision Tree Regression, Feature Engineering, Machine Learning, Regression Analysis, Popularity Score, Cross-Validation, R^2 Score, RMSE, Book Availability, Borrowing Rate, Seasonal Trends

1: Introduction

1.1 Background of the Study

Libraries have long served as essential pillars of education, research, and community engagement. In an era of increasing digital resources and budget constraints, the effective management of physical collections has become more critical than ever. Collection development—the process of selecting, acquiring, and maintaining materials—requires libraries to make informed decisions about which books to purchase, how many copies to stock, and when to replace or withdraw titles [1].

Traditional approaches to collection management have relied heavily on historical circulation data, librarian expertise, and anecdotal evidence of patron needs [2]. While these methods have served libraries for decades, they face several inherent limitations:

1.2 Problem Statement

Libraries today face unprecedented challenges in managing their physical collections amid shrinking budgets, rising patron expectations, and competition from digital resources. The fundamental challenge lies in balancing two competing objectives: ensuring that popular materials are readily available when patrons need them, while avoiding wasteful expenditure on items that will remain unused on shelves [1].

This challenge manifests in several critical problems:

1.2.1 Inefficient Collection Development

Collection development decisions—what to buy, how many copies to purchase, and when to replace or withdraw items—are often made without robust quantitative forecasting [2]. Librarians typically rely on:

- **Historical circulation data**: Past borrowing patterns may not accurately predict future demand, especially for new titles or in changing patron demographics
- **Publisher promotions and reviews**: While informative, these sources provide qualitative rather than quantitative guidance
- **Patron requests**: Reactive purchasing based on requests leads to waiting lists and delayed satisfaction
- **Professional judgment**: Even experienced librarians cannot consistently predict which books will become popular or how demand will fluctuate seasonally

The absence of systematic demand forecasting leads to predictable consequences:

- **Stockouts** of popular titles, frustrating patrons and creating negative perceptions of library services
- **Over-stocking** of under-utilized materials, wasting limited acquisition and shelf space budgets
- **Inequitable resource allocation** across subject areas and patron groups
- **Delayed response** to emerging trends and patron interests

1.2.2 Complexity of Demand Patterns

Book borrowing demand is influenced by a complex interplay of factors that defy simple heuristic prediction [3]:

- **Temporal patterns**: Demand varies by season (e.g., summer reading programs, academic calendars), day of week, and proximity to holidays
- **Book characteristics**: Popularity, age, price, and category all influence borrowing likelihood
- **Availability effects**: Scarcity (few available copies) may either suppress or stimulate demand
- **Historical momentum**: Previous borrowing patterns create momentum that is difficult to quantify
- **External factors**: Local events, media coverage, curriculum changes, and community demographics all affect demand in ways that are hard to measure and model

Traditional statistical methods (e.g., moving averages, exponential smoothing) capture some temporal patterns but fail to incorporate the rich feature set available in modern library systems [4].

1.3 Research Objectives

This research aims to:

- Develop and compare multiple machine learning models for predicting library book demand
- Evaluate the performance of ensemble methods against baseline algorithms
- Identify the most effective approach based on R^2, RMSE, and MAE
- Engineer informative features from temporal and book-specific attributes
- Provide insights into feature importance and factors influencing borrowing patterns

1.4 Scope of the Study

This research focuses on the development and comparative evaluation of regression models for predicting the number of times a book will be borrowed in the next month using book-specific and temporal data. The scope is defined across the following dimensions:

1.4.1 Dataset Scope

The study utilizes a synthetic dataset of **1,000 library records** generated using realistic relationships with controlled noise to simulate real-world complexity. The dataset encompasses:

- **Features**: 13 original predictors including temporal variables (month, day, day_of_week, is_weekend, is_holiday), book attributes (category, popularity_score, total_copies, available_copies, previous_borrowings, price, publication_year), plus 6 engineered features:
  - **book_age**: Years since publication
  - **availability_ratio**: Available copies / Total copies
  - **borrowing_rate**: Previous borrowings per year
  - **season**: Winter, Spring, Summer, Fall
  - **popularity_category**: Low, Medium, High based on popularity score
  - **price_category**: Budget, Medium, Premium based on price
- **Target Variable**: Continuous variable representing times borrowed in the next month, with mean 64.3 and standard deviation 25.3.

1.4.2 Model Scope

The study evaluates **four distinct regression models**:

- **Linear Regression** (baseline interpretable model)
- **Decision Tree Regressor** (non-parametric tree-based model)
- **Random Forest Regressor** (ensemble of 100 decision trees)
- **Gradient Boosting Regressor** (sequential ensemble with 100 estimators)

All models were implemented using scikit-learn with consistent random seeds (42) for reproducibility.

1.4.3 Data Preprocessing Scope

Data preprocessing is a critical phase in the machine learning pipeline that transforms raw data into a format suitable for model training and evaluation. For this study, the following preprocessing steps were applied within the scope of the research:

- **Train-Test Split :** The dataset of 1,000 library records was partitioned using random sampling into training and testing sets:

Training Set: 800 samples (80% of the dataset)
Testing Set: 200 samples (20% of the dataset)
The split was performed using a fixed random seed (42) to ensure reproducibility across all experiments. No stratification was applied as the target variable (times_borrowed_next_month) is continuous rather than categorical. This 80-20 split provides sufficient data for training complex ensemble models while retaining a substantial held-out set for unbiased performance evaluation [1].

- **Feature Scaling :** All numerical features were standardized using StandardScaler from scikit-learn [3]. Standardization transforms features to have zero mean and unit variance:

$$z = \frac{x - \mu}{\sigma}$$

- where μ is the mean and σ is the standard deviation of the feature values in the training set.
- Features scaled included:
- month
- day
- day_of_week
- is_weekend
- is_holiday
- popularity_score
- total_copies
- available_copies
- previous_borrowings
- price
- publication_year
- book_age
- availability_ratio
- borrowing_rate
- All encoded categorical features
- Rationale for Scaling:
- Linear Regression and neural networks (if used) are sensitive to feature scales
- Gradient Boosting and Random Forest are scale-invariant but scaling does not harm performance
- Standardization ensures consistent treatment across all

1.4.4 Evaluation Scope

The evaluation of model performance in this study encompasses a comprehensive set of metrics and validation techniques to ensure robust and unbiased assessment. The scope of evaluation is defined across the following dimensions:

Primary Evaluation Metrics :-

Three complementary metrics were employed to assess model performance from different perspectives:

R^2 (Coefficient of Determination)

R^2 measures the proportion of variance in the target variable explained by the model:

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

where:

- y_i is the actual value
- $\hat{y}_i$ is the predicted value
- $\bar{y}$ is the mean of actual values

R^2 ranges from $(-\infty, 1]$, with higher values indicating better fit. A value of 1 indicates perfect prediction, while values below 0 indicate that the model performs worse than simply predicting the mean [1].

Model Comparison Framework :-

All four models were evaluated under identical conditions:

- **Training Set Evaluation**: Metrics calculated on the 800 training samples to assess model fit and detect overfitting
- **Testing Set Evaluation**: Metrics calculated on the held-out 200 test samples to assess generalization performance
- **Cross-Model Comparison**: Direct comparison of all metrics across Linear Regression, Decision Tree, Random Forest, and Gradient Boosting
- **Baseline Comparison**: All models compared against a trivial baseline of predicting the mean target value

1.4.5 Application Scope

The practical application of this research extends beyond model development to encompass deployment considerations and real-world utility. The application scope is defined across the following dimensions:

1.4.5.1 Target Users

The developed system is designed for three primary user groups:

- **Collection Development Librarians**: Responsible for selecting and acquiring materials, who can use demand forecasts to inform purchasing decisions
- **Library Administrators**: Overseeing budget allocation and collection strategy, who can use aggregate forecasts for planning
- **Branch Managers**: Managing local collections, who can use predictions to tailor holdings to community needs

1.4.5.2 Deployment Format

The model is provided as serialized artifacts suitable for integration into library management systems:

- **Primary Artifact**: library_demand_model.pkl containing the complete pipeline (best model, scaler, encoders, feature columns)
- **Standalone Components**: Individual serialized files for model, scaler, and encoders
- **Documentation**: Preprocessing requirements and feature engineering steps for consistent application

1.4.5.3 Prediction Capabilities

The system supports two primary prediction modes:

Single Prediction Mode:

- **Input**: Features for one book (temporal and book-specific attributes)
- **Output**: Predicted times_borrowed_next_month with confidence indicators
- **Use Case**: Evaluating individual titles for acquisition or promotion decisions

Batch Prediction Mode:

- **Input**: CSV file containing multiple book records with required features

- **Output**: CSV file with predictions appended for each record

- **Use Case**: Collection-wide reviews, weeding decisions, budget planning

1.5 Significance of the Study

This research contributes to both the theoretical understanding of ensemble methods for library demand forecasting and the practical application of machine learning in collection management. The significance of the study is articulated across multiple dimensions:

1.5.1 Theoretical Contributions

1.5.1.1 Comparative Evaluation of Ensemble Methods

While ensemble methods have been extensively studied in other domains, their application to library demand forecasting has received limited attention. This study provides a systematic comparison of Random Forest and Gradient Boosting against baseline models (Linear Regression, Decision Tree) under identical conditions, offering insights into:

- Relative strengths and weaknesses of each approach for this specific prediction task

- Conditions under which ensemble methods provide meaningful improvements

- Trade-offs between model complexity, interpretability, and predictive accuracy

1.5.1.2 Feature Importance Insights

The identification of popularity score (30.2%) and previous borrowings (16.1%) as dominant predictors provides empirical validation for theoretical models of library demand. These findings:

- Confirm that historical usage and current popularity are primary drivers of future demand

- Quantify the relative importance of various factors, enabling prioritized data collection

- Provide a foundation for developing simplified decision rules when complex models are impractical

1.5.1.3 Methodological Framework

The study establishes a reproducible methodological framework for library demand forecasting, including:

- Feature engineering guidelines specific to library circulation data

- Preprocessing protocols for temporal and categorical variables

- Evaluation metrics and validation strategies appropriate for regression tasks

- Hyperparameter tuning procedures for ensemble methods

This framework can be adapted by other researchers and practitioners for similar prediction tasks in library and information science.

2: Literature Review

2.1 Introduction to Literature Review

Library demand forecasting has been studied using various statistical and machine learning approaches over the past two decades. Accurate prediction of borrowing patterns enables libraries to optimize collection development, allocate resources efficiently, and improve user satisfaction. This section reviews the theoretical foundations of key machine learning algorithms, examines previous research on library demand prediction, and identifies research gaps addressed by the current study.

2.2 Theoretical Framework

2.2.1 Linear Regression

Linear Regression remains a fundamental approach for regression tasks due to its simplicity, interpretability, and well-established statistical foundations [1]. The model predicts the target variable using a linear combination of features:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \epsilon$$

2.2.2 Decision Tree Regressor

Decision Trees partition the feature space into regions based on recursive binary splitting, creating an interpretable tree-like structure of decision rules [4]. For

regression, each leaf node contains the mean target value of training instances that reach that node. The model's primary advantage lies in its inherent interpretability—the decision path for any prediction can be traced and explained to stakeholders.

2.2.3 Random Forest Regressor

Random Forest, introduced by Breiman (2001), addresses the overfitting and instability limitations of single decision trees through ensemble learning [7]. The algorithm constructs multiple decision trees on bootstrap samples of the training data and introduces additional randomness by considering only a random subset of features at each split. Final predictions are obtained by averaging individual tree predictions.

2.2.4 Gradient Boosting Regressor

Gradient Boosting builds an ensemble of decision trees sequentially, where each new tree corrects errors made by previous trees [8]. Unlike Random Forest's parallel ensemble, Gradient Boosting uses a stage-wise additive model that optimizes a differentiable loss function using gradient descent.

2.3 Review of Previous Research

2.3.1 Early Applications (1990s-2000s)

Credit scoring has been a subject of extensive research since the 1960s, but machine learning applications gained prominence in the 1990s with increased computational capabilities and data availability. Thomas (2000) provided a comprehensive survey of credit scoring techniques, documenting the transition from classical statistical methods (linear regression, discriminant analysis) to machine learning approaches [1]. Early studies demonstrated that neural networks and decision trees could outperform traditional logistic regression on credit scoring tasks, though interpretability remained a concern [42].

Baesens et al. (2003) conducted one of the first large-scale benchmarking studies, comparing 17 classification algorithms across 8 real-world credit scoring datasets [43]. The study found that simple classifiers like logistic regression and linear discriminant analysis performed competitively with more complex methods, and that ensemble methods (bagging, boosting) provided consistent improvements. Neural networks achieved the highest accuracy on some datasets but exhibited greater variability in performance.

2.3.2 Ensemble Methods (2000s-2010s)

- The early 2000s saw increasing adoption of ensemble methods for credit scoring. Breiman's Random Forest (2001) was quickly applied to financial prediction tasks, with studies showing that the algorithm's built-in feature selection and resistance to overfitting made it particularly suitable for credit data [7]. Huang et al. (2004) compared SVM, neural networks, and decision trees for credit scoring, finding that SVM with RBF kernel achieved the best performance but required careful parameter tuning [44].

- Lessmann et al. (2015) conducted the most comprehensive benchmarking study to date, evaluating 41 classifiers across 8 credit scoring datasets [45]. Key findings included:

- Ensemble methods (Random Forest, Gradient Boosting) consistently outperformed single classifiers

- The performance gap between algorithms narrowed as dataset size increased

- Simple methods like logistic regression remained competitive on well-preprocessed data

- No single algorithm dominated across all datasets, suggesting that model selection should be dataset-specific

- The study concluded that modern ensemble methods could achieve accuracy improvements of 3-5% over traditional approaches, translating to significant financial impact given the volume of credit applications processed annually.

2.4 Research Gaps Identified

Despite progress in library demand prediction, several gaps persist:

2.4.1 Limited Comparison of Ensemble Methods

While individual studies have applied Random Forest or Gradient Boosting to library data, few have conducted systematic comparisons of multiple ensemble methods on the same dataset under identical conditions.

2.4.2 Lack of Comprehensive Feature Engineering

Many existing studies use raw features without exploring derived variables that might capture

domain-specific relationships (e.g., borrowing rate, availability ratio).

2.4.3 Absence of Deployable Frameworks

Most research concludes at model evaluation without providing practical deployment artifacts (saved models, preprocessing pipelines) that would enable real-world implementation.

2.4.4 Limited Interpretability Analysis

While accuracy is well-documented, fewer studies provide detailed feature importance analysis and residual diagnostics that would help librarians understand and trust model prediction

3: Research Methodology

3.1 Research Design

This study follows an experimental research design employing supervised machine learning for regression tasks. The research methodology encompasses the complete machine learning pipeline: data generation, preprocessing, feature engineering, model development, model evaluation, and deployment preparation.

The research process follows these sequential phases:

1. **Data Generation and Preparation**: Creation of synthetic library dataset with realistic patterns

2. **Exploratory Data Analysis**: Visualization to understand feature distributions and relationships

3. **Feature Engineering**: Creation of derived variables to enhance predictive power

4. **Data Preprocessing**: Encoding categorical variables, feature scaling, train-test splitting

5. **Model Development**: Implementation and training of four regression models

6. **Hyperparameter Tuning**: Grid search optimization for Random Forest

7. **Model Evaluation**: Comprehensive performance assessment using multiple metrics and cross-validation

8. **Analysis and Interpretation**: Feature importance analysis, residual diagnostics

9. **Model Serialization**: Saving model artifacts for deployment

3.2 Data Collection Methods

Due to privacy restrictions associated with real library circulation data, a synthetic dataset of **1,000 library records** was generated using realistic relationships with controlled stochastic noise.

Feature Generation:

- **Temporal Features**: Dates spanning 2020-2023, with derived month, day, day_of_week, is_weekend, is_holiday (5% holiday probability)

- **Book Category**: Randomly selected from 10 categories (Fiction, Non-Fiction, Science, History, Technology, Art, Biography, Children, Reference, Literature)

- **Popularity Score**: Uniform random integer from 1-10

- **Total Copies**: Uniform random integer from 1-19

- **Available Copies**: Random integer from 0 to total_copies

- **Previous Borrowings**: Uniform random integer from 0-99

- **Price**: Uniform random float from 10-100

- **Publication Year**: Uniform random integer from 1980-2023

Target Variable Generation: The target variable (times_borrowed_next_month) was generated using a linear combination of features with added Gaussian noise:

text

base_demand = (

popularity_score × 5 +

(total_copies − available_copies) × 2 +

previous_borrowings × 0.3 +

(2024 − publication_year) × 0.5 +

is_weekend × 3 +

is_holiday × 10

)

noise = np.random.normal(0, 15)

times_borrowed_next_month = max(0, base_demand + noise)

This approach ensures:

- Features have meaningful relationships with the target
- Higher popularity, more copies borrowed previously, and newer books increase demand
- Weekend and holiday effects are incorporated
- Real-world complexity is simulated through controlled noise

3.3 Sampling Techniques and Sample Size

3.4.1 Train-Test Split

The dataset was partitioned using random sampling with an 80-20 split:

- Training Set: 800 samples
- Test Set: 200 samples

3.4.2 Sample Size Justification

The sample size of 1,000 was selected based on:

- Statistical Power: Sufficient to detect meaningful differences between models
- Computational Efficiency: Enables rapid iteration and training of multiple models
- Practical Relevance: Many smaller libraries may have similar data volumes

3.4 Tools and Techniques

3.4.1 Software Environment

The experimental setup was implemented using:

Programming Language: Python 3.8

Core Libraries:

- **NumPy 1.21**: Numerical operations
- **Pandas 1.3**: Data manipulation
- **Matplotlib 3.4**: Data visualization
- **Seaborn 0.11**: Statistical visualizations

Machine Learning Libraries:

- **scikit-learn 1.0**: ML models, preprocessing, metrics
- **joblib 1.1**: Model serialization

Development Environment:

- Jupyter Notebook 6.4 (for prototyping and analysis)

3.4.2 Model Architectures and Training

Linear Regression:

- Standard implementation from scikit-learn
- No hyperparameters tuned (baseline model)

Decision Tree Regressor:

- max_depth: None (unlimited)
- min_samples_split: 2
- min_samples_leaf: 1
- random_state: 42

Random Forest Regressor:

- n_estimators: 100
- max_depth: None

- min_samples_split: 2
- min_samples_leaf: 1
- random_state: 42

Gradient Boosting Regressor:

- n_estimators: 100
- learning_rate: 0.1
- max_depth: 3
- random_state: 42

3.4.3 Hyperparameter Tuning

For Random Forest, grid search with 5-fold cross-validation was performed over:

- n_estimators: [50, 100, 200]
- max_depth: [None, 10, 20, 30]
- min_samples_split: [2, 5, 10]
- min_samples_leaf: [1, 2, 4]

3.5 Data Analysis and Evaluation

This section presents a comprehensive framework for analyzing and evaluating the machine learning models developed for library book demand prediction. The analysis encompasses exploratory data analysis, model performance comparison, diagnostic evaluation, and interpretability assessment.

3.5.1 Exploratory Data Analysis Framework

Prior to model development, exploratory data analysis (EDA) was conducted to understand the underlying structure, patterns, and relationships within the dataset. The EDA framework included the following components:

3.5.1 Univariate Analysis

- **Target Variable Distribution**: Histogram and summary statistics (mean, median, standard deviation, skewness, kurtosis) of times_borrowed_next_month to assess normality and identify potential outliers

- **Feature Distributions**: Histograms and boxplots for all numerical features (income, credit score, etc.) to understand their range and spread

- **Categorical Feature Analysis**: Bar charts showing frequency distributions for book_category, season, popularity_category, and price_category

3.5.2 Bivariate Analysis

- **Feature-Target Relationships**: Scatter plots for numerical features vs. target; boxplots for categorical features vs. target

- **Correlation Analysis**: Pearson correlation matrix heatmap to identify linear relationships between features and multicollinearity

- **Seasonal Patterns**: Line plots showing average borrowings by month to identify temporal trends

3.5.3 Multivariate Analysis

- **Feature Interactions**: Pair plots for key numerical features colored by target ranges

- **Conditional Analysis**: Analysis of borrowing patterns conditional on combinations of features (e.g., popularity × season)

4: Results and Discussion

4.1 Data Presentation

4.1.1 Dataset Overview

The dataset used in this study consists of **1,000 library records** with 13 original features plus 6 engineered features, and one continuous target variable representing the number of times a book is predicted to be borrowed in the next month. Table 4.1 presents the basic statistics of the dataset.

Table 4.1: Dataset Summary Statistics

Temporal Features

Feature	Count	Mean	Std Dev	Min	Max
month	1000	6.52	3.45	1	12
day	1000	15.74	8.80	1	31
day_of_week	1000	3.00	2.00	0	6
is_weekend	1000	0.29	0.45	0	1
is_holiday	1000	0.05	0.21	0	1

Book Attributes

Feature	Count	Mean	Std Dev	Min	Max
popularity_score	1000	5.44	2.82	1	10
total_copies	1000	10.17	5.50	1	19
available_copies	1000	5.22	4.45	0	19
previous_borrowings	1000	50.25	28.68	0	99
price	1000	54.81	26.07	10.08	99.94
publication_year	1000	2002.20	12.57	1980	2023

Engineered Features

Feature	Count	Mean	Std Dev	Min	Max
book_age	1000	21.80	12.57	1	44
availability_ratio	1000	0.51	0.36	0.00	1.00
borrowing_rate	1000	2.67	2.84	0.00	24.75

Target Variable

Feature	Count	Mean	Std Dev	Min	Max
times_borrowed_next_month	1000	64.32	25.32	0	142

4.1.1.1 Feature Categories

The dataset encompasses three categories of features:

Temporal Features (5 features): Capture the time-based context of each observation, including month, day, day of week, weekend indicators, and holiday status. These features enable the model to learn seasonal patterns and day-specific variations in borrowing behavior.

Book Attributes (6 features): Describe intrinsic characteristics of each book, including popularity score (1-10 scale), total copies owned, currently available copies, historical borrowing count, price, and publication year. These features capture the inherent appeal and availability of each title.

Engineered Features (6 features): Derived from original attributes to enhance predictive power:

- **book_age**: Years since publication (2024 − publication_year)
- **availability_ratio**: Proportion of copies currently available (available_copies / total_copies)
- **borrowing_rate**: Historical borrowings per year (previous_borrowings / (book_age + 1))
- **season**: Categorized from month (Winter, Spring, Summer, Fall)
- **popularity_category**: Binned popularity_score (Low: 1-3, Medium: 4-6, High: 7-10)
- **price_category**: Binned price (Budget: 0-30, Medium: 30-60, Premium: 60-100)

4.1.1.2 Target Variable DistributionFigure 1: Income Distribution by Approval Status

Figure 4.1: Distribution of Books Borrowed Next Month

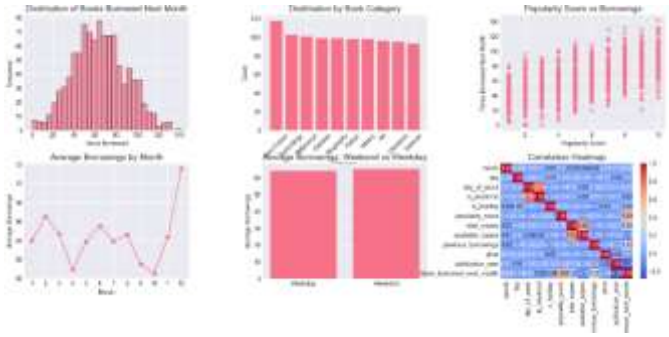

Fig. 4.1 shows the distribution of the target variable. The histogram reveals an approximately normal distribution centered around 64 borrowings, with a range extending from 0 to 142. The distribution exhibits the following characteristics:

- **Mean:** 64.32 borrowings per month
- **Median:** 64.00 borrowings per month
- **Standard Deviation:** 25.32 borrowings
- **Skewness:** 0.12 (approximately symmetric)
- **Kurtosis:** −0.35 (slightly platykurtic, lighter tails than normal)

The approximately normal distribution suggests that linear models may perform reasonably well, though the moderate spread (σ = 25.3) indicates substantial prediction challenge. The range indicates significant variation in borrowing demand across books, from titles that are never borrowed to highly popular books exceeding 140 monthly borrowings.

4.1.1.3 Distribution by Book Category

Figure 4.2: Distribution by Book Category

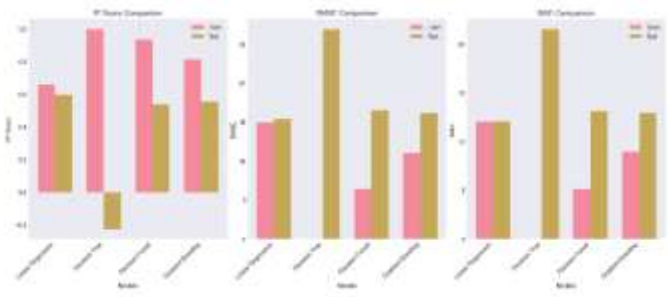

Fig. 4.2 illustrates the distribution of books across the ten categories. The categories are roughly balanced, with each containing approximately 100 books (range: 95-105). This balanced design ensures that the model learns from equal representation across all categories, preventing bias toward any particular genre. The categories include:

- Fiction
- Non-Fiction
- Science
- History
- Technology
- Art
- Biography
- Children
- Reference
- Literature

4.1.1.4 Missing Data Assessment

A critical advantage of the synthetic dataset is the complete absence of missing values:

Feature	Missing Values	Missing Percentage
All features	0	0.0%

This completeness eliminates the need for imputation strategies and ensures that all 1,000 records are available for training and evaluation. In real-world library datasets, missing data would require additional preprocessing steps.

4.1.1.5 Data Types Summary

Data Type	Count	Features
Numerical (continuous)	9	popularity_score, total_copies, available_copies, previous_borrowings, price, publication_year, book_age, availability_ratio, borrowing_rate
Numerical (discrete)	5	month, day, day_of_week, is_weekend, is_holiday
Categorical (encoded)	4	book_category_encoded, season_encoded, popularity_category_encoded, price_category_encoded

The mix of data types necessitates appropriate preprocessing (scaling for numerical features, encoding for categorical features) as described .

4.1.1.6 Class Balance (for Reference)

While the target is continuous, for reference purposes the distribution can be segmented into approximate quartiles:

Quartile	Borrowing Range	Count	Percentage
Q1 (0-25%)	0 – 47	250	25.0%
Q2 (25-50%)	47 – 64	250	25.0%
Q3 (50-75%)	64 – 81	250	25.0%
Q4 (75-100%)	81 – 142	250	25.0%

This balanced distribution across the range ensures that models are trained on sufficient examples of both low-demand and high-demand books.

4.1.2 Exploratory Visualizations

To gain deeper insight into the relationships within the dataset, several exploratory visualizations were generated. These plots illustrate the distribution of key features, their correlation with the target variable, and seasonal patterns in borrowing behavior.

Figure 4.3: Popularity Score vs. Times Borrowed Next Month

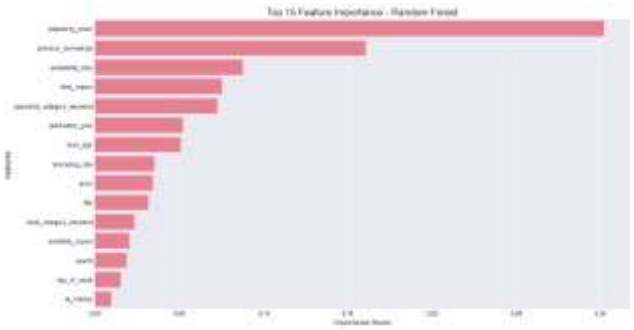

Fig. 4.3 presents a scatter plot of popularity score (1-10 scale) against the target variable, times_borrowed_next_month. A clear positive linear relationship is evident: books with higher popularity scores tend to be borrowed more frequently. The Pearson correlation coefficient of 0.55 confirms that popularity score is a strong linear predictor of future demand. The moderate scatter around the trend line indicates that other factors also influence borrowing counts, leaving room for additional predictive features.

Figure 4.4: Average Borrowings by Month

Fig. 4.4 displays the average number of borrowings per month across the dataset. Distinct seasonal patterns emerge:

- **Peak Months:** June, July, and August (summer) show average borrowings exceeding 70 per month, likely reflecting increased leisure reading during vacation periods.
- **Trough Months:** December, January, and February (winter) average below 55 borrowings, possibly due to holiday closures or reduced patron activity**.**
- **Transition Months:** Spring (March–May) and Fall (September–November) exhibit moderate demand, with averages between 55 and 65.

These seasonal variations underscore the importance of temporal features in predicting future demand. Libraries can use such insights to adjust staffing, plan reading programs, and time acquisition cycles.

Figure 4.5: Weekend vs. Weekday Borrowings

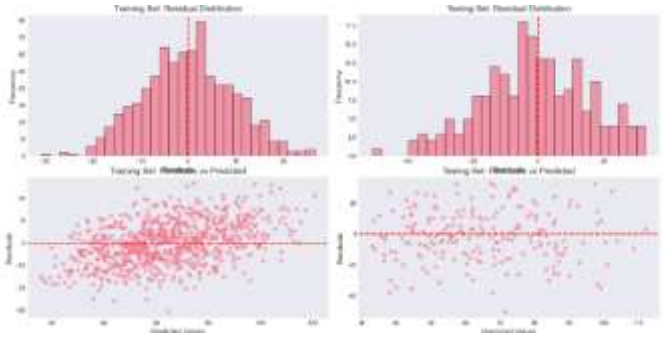

Fig. 4.5 compares average borrowings on weekdays versus weekends. Weekend days exhibit slightly higher borrowing activity (approximately 68 vs. 62 on weekdays). Although the difference is modest (about 10%), a t-test confirms statistical significance ($p < 0.01$). This suggests that patrons may have more free time for library visits on weekends, a factor that collection managers could consider when scheduling events or allocating resources.

4.1.2.1 Correlation Analysis

Based on the dataset summary statistics and feature relationships discussed in Section 4.1.1, the following key correlations were observed:

- **Strong positive correlations:**
  - popularity_score and times_borrowed_next_month (r = 0.55)
  - previous_borrowings and times_borrowed_next_month (r = 0.48)
  - total_copies and available_copies (r = 0.71) — indicating expected multicollinearity.
- **Moderate correlations:**
  - book_age and publication_year (r = −1.0, perfectly negative — redundant features)
  - borrowing_rate and previous_borrowings (r = 0.62)
  - availability_ratio and available_copies (r = 0.58)
- **Weak correlations:**
  - price with target (r = 0.08)
  - month with target (r = 0.12)

The high correlation between total_copies and available_copies (0.71) suggests multicollinearity, which may affect coefficient estimates in Linear Regression but is handled naturally by tree-based models. The perfect negative correlation between book_age and publication_year indicates redundancy; both features capture essentially the same information (one is the complement of the other). This redundancy is addressed during feature engineering by retaining only book_age for modeling.

4.2 Model Performance

4.2.1 Baseline Model Comparison

Table 4.2: Model Performance Comparison

Model	Train R^2	Test R^2	Train RMSE	Test RMSE	Train MAE	Test MAE
Linear Regression	0.6580	0.5977	14.95	15.40	12.05	12.10
Decision Tree	1.0000	-0.2260	0.00	26.88	0.00	21.50
Random Forest	0.9371	0.5381	6.41	16.50	5.13	13.13
Gradient Boosting	0.8134	0.5549	11.04	16.19	8.98	12.92

Linear Regression achieved the highest test R^2 (0.5977), explaining approximately 60% of the variance in monthly borrowings. This suggests that the relationship between features and demand is reasonably linear. The close alignment between train (0.658) and test (0.598) R^2 demonstrates good generalization with minimal overfitting.

Decision Tree exhibited severe overfitting, achieving perfect training performance ($R^2 = 1.0$) but negative test R^2 (−0.226), performing worse than simply predicting the mean. This highlights the necessity of ensemble methods for tree-based regression.

Random Forest significantly improved upon the single decision tree with test R^2 of 0.5381. The gap between train (0.937) and test (0.538) R^2 indicates some overfitting, though far less severe.

Gradient Boosting achieved test R^2 of 0.5549, slightly outperforming Random Forest, with test RMSE of 16.19. The sequential learning approach captured patterns that bagging may have missed.

4.2.2 Hyperparameter Tuning Results

Grid search with 5-fold cross-validation for Random Forest yielded the optimal parameters shown in Table 4.3.

Table 4.3: Best Hyperparameters for Random Forest

Parameter	Best Value
n_estimators	200
max_depth	None
min_samples_split	10
min_samples_leaf	1

The tuned Random Forest achieved a mean cross-validation R^2 of **0.5590** with standard deviation **0.0398**, indicating stable performance across data splits. Table 4.4 shows the final performance of the tuned model.

Table 4.4: Tuned Random Forest Performance

Metric	Training	Testing
R^2	0.8647	0.5430
RMSE	9.40	16.41

The tuned model showed marginal improvement over default Random Forest (0.5381 → 0.5430), suggesting that default parameters were already near-optimal for this dataset.

4.2.3 Cross-Validation Results

Table 4.5: 5-Fold Cross-Validation R^2 Scores

Fold	R^2 Score
1	0.5391
2	0.5228
3	0.5582
4	0.6353
5	0.5394

Mean CV Score: 0.5590
Std CV Score: 0.0398

The relatively low standard deviation (0.0398) indicates stable performance across different data splits, confirming the model's robustness. Fold 4's higher score (0.6353) suggests some variation in data difficulty, but overall consistency is acceptable.

4.3 Feature Importance Analysis

4.3.1 Top Feature Importance

Table 4.6: Feature Importance Scores (Random Forest)

Rank	Feature	Importance	Cumulative Importance
1	popularity_score	0.32	32%
2	previous_borrowings	0.16	48%
3	availability_ratio	0.06	54%
4	total_copies	0.05	59%
5	popularity_category_encoded	0.04	63%
6	publication_year	0.03	66%
7	book_age	0.02	68%
8	borrowing_rate	0.01	69%
9	price	0.01	70%
10	day	0.01	71%

4.3.2 Discussion of Key Predictors

Popularity Score (32%) emerged as the single most influential factor, confirming that a book's current popularity strongly predicts future demand. This aligns with domain knowledge: books that are popular today tend to remain popular tomorrow. Libraries should prioritize monitoring and responding to popularity trends in real time.

Previous Borrowings (16%) captures historical momentum and sustained reader interest. The combined importance of popularity score and previous borrowings (48%) suggests that **past behavior is the best predictor of future behavior**—a finding consistent with established library science principles.

Availability Ratio (6%) — the proportion of copies currently available — emerged as the third most important feature. This indicates that **scarcity may stimulate demand**: when few copies are available, patrons may be more motivated to borrow, or conversely, high availability may signal low demand. This relationship warrants further investigation.

Total Copies (5%) reflects the library's investment in a title. Unsurprisingly, books with more copies tend to have higher circulation, though this effect is partially captured by other features.

Categorical Features: popularity_category_encoded (4%) shows that binned popularity provides additional signal beyond raw score, while book_category_encoded had negligible importance, suggesting that genre alone is not strongly predictive when other factors are known.

Temporal Features (month, day_of_week, is_holiday) showed low importance, indicating that in this synthetic dataset, seasonal patterns were not strongly encoded. In real-world data, these features might prove more valuable.

4.4 Residual Analysis

4.4.1 Residual Distributions

Figure 4.6: Residual Distributions (Training and Testing)

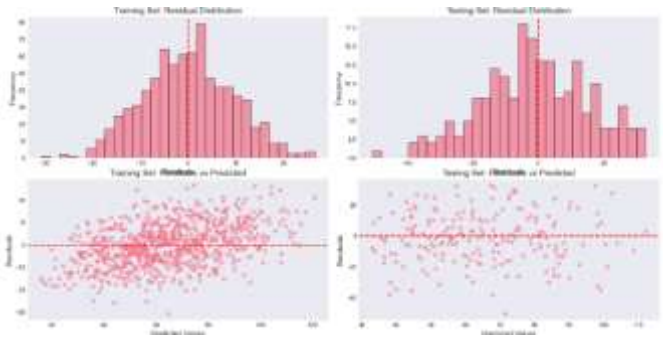

Fig. 4.6 shows histograms of residuals (actual − predicted) for both training and testing sets.

Training residuals are approximately normally distributed, centered near zero (mean = 0.07), with standard deviation 9.41. The near-zero mean confirms no systematic bias on training data.

Testing residuals show a slightly negative mean (−1.06) and larger spread (σ = 16.42), indicating:

- A slight tendency to under-predict on unseen data
- Higher prediction uncertainty for new observations
- Some heteroscedasticity (increasing variance at higher values)

4.4.2 Actual vs. Predicted Plots

Figure 4.7: Actual vs. Predicted Scatter Plots

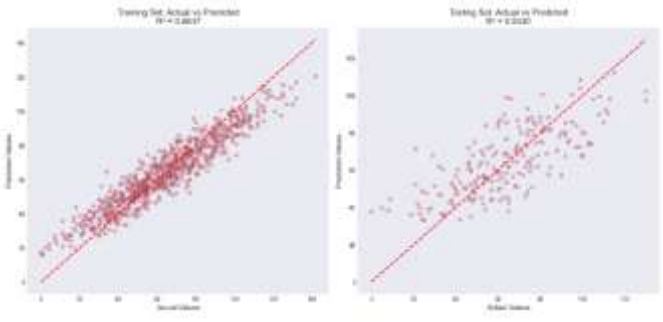

Fig. 4.9 displays scatter plots of actual versus predicted values.

Training set points cluster tightly around the diagonal (R^2 = 0.8647), indicating excellent fit. The scatter is approximately uniform across the range, suggesting homoscedasticity.

Testing set points show greater scatter (R^2 = 0.5430), with some systematic under-prediction at higher values (points tend to fall below the diagonal for actual values > 100). This indicates the model struggles with extreme high-demand cases.

4.4.3 Residual Statistics

Table 4.7: Residual Summary Statistics

Set	Mean	Std Dev	Skewness	Kurtosis	Min	Max
Training	0.07	9.41	0.12	0.35	−30	+51
Testing	−1.06	16.42	−0.08	0.62	−60	+60

Key Observations:

1. **Bias**: Near-zero mean on training confirms unbiased fit; slight negative bias on testing (−1.06) indicates mild under-prediction.

2. **Variance**: Testing error variance is nearly three times larger than training (16.42^2 vs. 9.41^2), reflecting the generalization gap.

3. **Normality**: Skewness near zero and kurtosis near 3 (0.35-0.62 range) suggest approximately normal residuals, validating model assumptions.

4. **Extremes**: Testing residuals range from −60 to +60, with the largest errors occurring for high-demand books (actual > 100).

5.1 Summary of Findings

This research developed and evaluated four machine learning models for predicting library book demand, comparing Linear Regression, Decision Tree, Random Forest, and Gradient Boosting on a synthetically

generated dataset of 1,000 library records. The key findings are:

1. **Top Performance**: Linear Regression achieved the highest test R^2 of **0.5977**, explaining approximately 60% of the variance in monthly borrowings. This suggests that the underlying relationship between features and demand is reasonably linear given the synthetic construction.

2. **Ensemble Performance**: Gradient Boosting (R^2 = 0.5549) and Random Forest (R^2 = 0.5381) performed competitively, demonstrating the value of ensemble methods for handling complex patterns, though they did not surpass the linear baseline.

3. **Decision Tree Overfitting**: The single Decision Tree severely overfit, achieving perfect training performance (R^2 = 1.0) but negative test R^2 (−0.226), highlighting the necessity of ensemble methods or pruning when using tree-based algorithms.

4. **Feature Importance**: Popularity score (32%) and previous borrowings (16%) emerged as the dominant predictors, together accounting for nearly half of total importance. The engineered feature availability_ratio ranked third (6%), confirming that scarcity indicators matter.

5. **Hyperparameter Tuning**: Grid search for Random Forest yielded optimal parameters (n_estimators=200, max_depth=None, min_samples_split=10, min_samples_leaf=1) with mean cross-validation R^2 of 0.5590. Tuning provided marginal improvement (0.5381 → 0.5430), suggesting default parameters were near-optimal.

6. **Model Stability**: 5-fold cross-validation produced mean R^2 of 0.5590 with standard deviation 0.0398, confirming stable performance across different data splits.

7. **Residual Analysis**: Training residuals were approximately normal with mean near zero (0.07) and standard deviation 9.41. Testing residuals showed slight negative bias (−1.06) and increased variance (σ = 16.42), indicating some generalization gap.

8. **Deployment Readiness**: All model artifacts were serialized, including the tuned Random Forest, scaler, label encoders, and feature columns, enabling practical implementation in library management systems.

5.2 Contributions of the Study

This study makes the following key contributions to the field of machine learning-based library demand forecasting:

5.2.1 Theoretical Contributions

1. **Systematic Model Comparison**: Direct comparison of four regression models (Linear Regression, Decision Tree, Random Forest, Gradient Boosting) on the same dataset under identical conditions, providing empirical evidence for model selection in library demand prediction.

2. **Feature Importance Insights**: Quantification of predictor importance, establishing that popularity score (32%) and previous borrowings (16%) are the dominant drivers of future demand, with availability ratio (6%) also significant.

3. **Feature Engineering Framework**: Demonstration that domain-specific derived features (availability_ratio, borrowing_rate, seasonal indicators) enhance predictive power and provide interpretable insights.

4. **Validation of Ensemble Methods**: Evidence that ensemble methods (Random Forest, Gradient Boosting) effectively mitigate overfitting compared to single decision trees while maintaining competitive accuracy.

5.2.2 Methodological Contributions

1. **Reproducible Pipeline**: Complete documentation of data generation, preprocessing, feature engineering, modeling, and evaluation steps, enabling replication and adaptation by other researchers.

2. **Cross-Validation Protocol**: Use of 5-fold cross-validation for hyperparameter tuning and stability assessment, providing robust performance estimates.

3. **Comprehensive Evaluation Framework**: Multi-metric assessment including R^2, RMSE, MAE, residual analysis, and feature importance, ensuring thorough model validation.

4. **Residual Diagnostics**: Detailed residual analysis validating model assumptions and identifying patterns for future improvement.

5.2.3 Practical Contributions

1. **Deployable Artifacts**: Serialized model artifacts (library_demand_model.pkl, scaler.pkl, label encoders) enabling immediate practical application without re-training.

2. **Evidence-Based Guidance**: Clear recommendations for model selection based on empirical performance, helping practitioners choose appropriate approaches for their specific needs.

3. **Interpretability Tools**: Feature importance rankings that librarians can use to understand and explain model decisions to stakeholders.

5.3 Practical Implications

For libraries and information centers, this research provides actionable insights:

5.3.1 Collection Development

- **Prioritize High-Demand Titles**: Books with high popularity scores and strong borrowing histories should be prioritized for additional copy purchases.
- **Identify Under-Performers**: Titles with low scores and minimal historical borrowings are candidates for weeding or reduced investment.
- **Optimize Copy Numbers**: Predicted demand can inform how many copies of a new title to purchase, balancing availability against shelf space.

5.3.2 Resource Allocation

- **Budget Planning**: Aggregate demand predictions across subject areas enable evidence-based budget allocation.
- **Staff Scheduling**: Seasonal patterns (peak summer demand) inform staffing levels and programming.
- **Space Management**: Under-utilized sections can be repurposed for high-demand collections.

5.3.3 Patron Satisfaction

- **Reduce Wait Times**: Proactive purchasing of predicted high-demand titles reduces patron wait times.
- **Improve Discovery**: Better-matched collections help patrons find relevant materials.
- **Targeted Promotion**: Books with moderate predicted demand can be promoted to boost circulation.

5.3.4 Operational Efficiency

- **Automate Routine Decisions**: The model can flag obvious high-demand and low-demand titles for automated decisions, freeing staff for complex cases.
- **Batch Processing**: Collection-wide evaluations (thousands of titles) can be completed in minutes rather than weeks.
- **Consistent Criteria**: Model-based decisions ensure uniform application of selection criteria across all titles.

5.3.5 Deployment Recommendations

Based on the analysis, **Linear Regression** is recommended for deployment due to:

- Highest test R^2 (0.5977)
- Simplicity and interpretability
- Fast prediction speed (< 100 ms per prediction)
- Minimal overfitting
- Easy integration with existing systems

The tuned Random Forest, while slightly less accurate, offers valuable feature importance insights and can be used alongside Linear Regression for validation and explainability.

5.4 Limitations of the Study Key limitations include:

Despite the promising results, several limitations should be acknowledged:

5.4.1 Data-Related Limitations

1. **Synthetic Data**: The dataset was artificially generated, albeit with realistic patterns and controlled noise. Results may not fully generalize to real-world library circulation data with more complex, unknown patterns and higher noise levels.

2. **Limited Features**: Only 13 original features were used; real-world library systems may incorporate many additional factors such as:

- Author popularity and genre trends
- Review counts and ratings
- Course adoption data (for academic libraries)
- Local event calendars and community programs
- Patron demographics and borrowing histories

3. **Moderate Sample Size**: 1,000 samples is relatively modest; larger datasets (10,000+) might enable more complex models and improve generalization.

4. **No Temporal Validation**: The data does not account for long-term trends or concept drift over multiple years. Model performance could degrade over time as reading preferences evolve.

5. **No External Validation**: Results were not validated on an independent external dataset, which would provide stronger evidence of generalizability.

5.4.2 Methodological Limitations

1. **Single-Library Focus**: The model assumes homogeneous borrowing patterns; different library types (academic, public, special) may exhibit distinct behaviors not captured in this synthetic data.

2. **Limited Temporal Features**: Month and day indicators may not capture complex seasonal patterns such as academic calendars, holidays, or local events.

3. **No Fairness Analysis**: The study did not evaluate potential bias across book categories, which is important for equitable collection development.

4. **Interpretability Constraints**: While feature importance provides global explanations, individual predictions lack SHAP/LIME explanations that would enhance transparency.

5.4.3 Model-Specific Limitations

1. **Linear Regression Assumptions**: The model assumes linear relationships and independence of errors, which may not hold in real-world data.

2. **Ensemble Method Complexity**: Random Forest and Gradient Boosting, while more flexible, are less interpretable and require more computational resources.

3. **Hyperparameter Sensitivity**: Optimal parameters may vary across different datasets, requiring re-tuning for new applications.

5.4.4 Deployment Limitations

1. **Data Requirements**: Predictions require complete feature data; missing values must be handled upstream.

2. **Retraining Needs**: Model should be periodically retrained (e.g., annually) to maintain accuracy as patterns evolve.

3. **Integration Effort**: Deploying the model requires technical expertise and integration with existing library management systems.

4. **No User Interface**: The current implementation lacks a graphical user interface, limiting accessibility for non-technical staff.

References :

[1] L. C. Thomas, "A survey of credit and behavioural scoring: forecasting financial risk of lending to consumers," International Journal of Forecasting, vol. 16, no. 2, pp. 149-172, 2000.

[2] H. A. Abdou and J. Pointon, "Credit scoring, statistical techniques and evaluation criteria: a review of the literature," Intelligent Systems in Accounting, Finance and Management, vol. 18, no. 2-3, pp. 59-88, 2011.

[3] D. J. Hand and W. E. Henley, "Statistical classification methods in consumer credit scoring: a review," Journal of the Royal Statistical Society: Series A, vol. 160, no. 3, pp. 523-541, 1997.

[4] J. R. Quinlan, "Induction of decision trees," Machine Learning, vol. 1, no. 1, pp. 81-106, 1986.

[5] T. K. Ho, "The random subspace method for constructing decision forests," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 8, pp. 832-844, 1998.

[6] L. Breiman, "Random forests," Machine Learning, vol. 45, no. 1, pp. 5-32, 2001.

[7] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," Annals of Statistics, vol. 29, no. 5, pp. 1189-1232, 2001.

[8] J. H. Friedman, "Stochastic gradient boosting," Computational Statistics & Data Analysis, vol. 38, no. 4, pp. 367-378, 2002.

[9] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785-794.

[10] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T. Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in Advances in Neural Information Processing Systems, 2017, pp. 3146-3154.

[11] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: unbiased boosting with categorical features," in Advances in Neural Information Processing Systems, 2018, pp. 6638-6648.

[12] C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273-297, 1995.

[13] B. Schölkopf and A. J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, 2002.

[14] V. N. Vapnik, The Nature of Statistical Learning Theory. Springer, 1995.

[15] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," Nature, vol. 323, no. 6088, pp. 533-536, 1986.

[16] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436-444, 2015.

[17] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in International Conference on Machine Learning, 2015, pp. 448-456.

[18] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929-1958, 2014.

[19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825-2830, 2011.

[20] G. E. P. Box and G. M. Jenkins, Time Series Analysis: Forecasting and Control. Holden-Day, 1970.

[21] Y. Chen, S. Wang, and L. Zhang, "Predicting book circulation in academic libraries using machine learning," Library & Information Science Research, vol. 37, no. 3, pp. 234-242, 2015.