Harnessing Machine Learning for Anticipating Product Demand Trends

VISHAL KUMAR
Amity Institute of Information and Technology
Amity University, Noida, INDIA

Abstract: - Harnessing machine learning for anticipating product demand trends is a critical aspect of modern supply chain management. This investigation delves into the realm of hybrid demand forecasting methodologies, notably the integration of ARIMAX and Neural Network models grounded in machine learning principles. Through the strategic application of these sophisticated techniques, businesses gain the capacity to augment accurate demand prediction, optimize inventory management, and elevate the overall efficiency of their supply chain operations. The research also addresses the intricacies of data preprocessing, underscoring the significance of mitigating challenges like noisy data and missing values. An exhaustive comparative study is carried out, analyzing the effectiveness of various machine learning models such as Linear Regression, Random Forest, ARIMA, and LSTM Networks. This scrutiny yields valuable insights into the distinctive capabilities of each model. The implications for businesses are profound, encompassing enhanced inventory management practices, streamlined production planning processes, and an overall optimization of the supply chain.

Keywords: Traditional forecasting techniques, Supply chain analytics, Supply chain management, Machine learning, Demand and sales forecasting.

Received: March 11, 2024. Revised: August 4, 2024. Accepted: September 8, 2024. Published: October 11, 2024.

1. Introduction

The process of forecasting demand for new products poses significant challenges compared to existing products due to the absence of historical data as a reliable indicator. Industries grappling with shorter product life cycles recognize the increasing importance of accurate new product forecasting. Beyond the hurdle of lacking historical data, time constraints and the uncertainty surrounding consumer acceptance and competitive reactions add to the complexity [1]. Despite these difficulties, early projections are vital for informing important choices like capacity planning, purchasing, and inventory management, all of which have a direct bearing on a business's operations [2]. A robust sales forecasting approach is vital to navigate potential complications during and after a new product launch, preventing stock-outs or overstock situations that could harm profitability and customer satisfaction. As the demand for new product forecasts rises, there is a growing need for analytical approaches. While quantitative models are advocated for forecasting new products, the adoption of analytical methods remains limited among companies. Common techniques involve expert opinions, surveys, and analyzing average sales of similar products to create meaningful estimations, emphasizing the importance of addressing risks associated with new product introductions [3]. To address these challenges and quantify uncertainty, a novel method named Demand Forest is introduced. Combining K-means clustering, Random Forest, and Quantile Regression Forest, this hybrid approach leverages historical demand data and product characteristics. The approach uses the broad versatility of the Random Forest algorithm and the quantile regression forest's capacity to quantify demand uncertainty to make inventory management choices easier and more adaptable to a wide range of businesses [4]. This research participates to the field by introducing Quantile Regression Forests in new product forecasting, evaluating their impact on inventory management, proposing an extension with theoretical distributions, and suggesting a synthetic dataset for future comparisons. Additionally, insights derived from Random Forest algorithms, such as feature importance and comparable products, offer valuable information for supply chain planners. The subsequent sections delve into related work, the proposed Demand Forest method, data and experimental details, results, and conclude with insights from the research [5].

2. Literature Review

2.1 Traditional Demand Forecasting Methods

Traditional Demand Forecasting Methods, offering a nuanced exploration of historical approaches and their limitations. It underscores the challenges faced by these methods in adapting to the intricate and volatile nature of contemporary markets [6]. The study recognizes the reliance on quantitative methods such as time series analysis, moving averages, and exponential smoothing in historical approaches [7].

2.2 Integration of Machine Learning in Demand Forecasting

The utilization of machine learning (ML) algorithms has become pivotal in demand forecasting. These algorithms demonstrate strength in handling extensive datasets, deciphering intricate patterns, and adapting to the dynamic nature of the market. The integration of ML techniques into demand forecasting has experienced remarkable growth in recent times. Diverse ML models, such as regression, decision trees, and neural networks, have demonstrated their effectiveness in forecasting demand. These models provide businesses with the capability to understand complex relationships within data, navigate non-linear trends, and capture subtle dependencies. Consequently, the adoption of ML in demand forecasting significantly contributes to enhancing the precision and accuracy of predictions[8].

2.3 Examination of Previous Studies and Real-world Applications

Numerous studies highlight the effectiveness of ML in demand forecasting across diverse industries. Case studies in retail, manufacturing, and e-commerce showcase tangible benefits such as reduced forecasting errors, improved inventory turnover, and heightened customer satisfaction. While success stories abound, challenges in ML implementation for demand forecasting have been documented. Issues related to data quality, model interpretability, and the necessity for continuous refinement underscore the intricacies of integrating ML into demand forecasting processes[9].

3. Methodology

The application of machine learning in forecasting product demand represents a sophisticated and data-driven approach. This method employs advanced algorithms and statistical techniques to meticulously analyze historical data, revealing critical patterns and trends. These insights are pivotal for making precise predictions and informed decisions. In the context of demand forecasting, machine learning algorithms play a central role in understanding customer behavior, navigating market fluctuations, and considering other relevant factors. The process involves the thorough collection of extensive historical sales data, followed by in-depth analysis to identify significant features and relationships[10].

One of the notable advantages of machine learning is its ability to not only enhance prediction accuracy but also enable businesses to adapt rapidly to changing market conditions. This proactive approach empowers businesses to anticipate and meet product demand efficiently, leading to optimized inventory management, heightened customer satisfaction, and sustained competitiveness in dynamic markets. Overall, the integration of machine learning in demand forecasting reflects a strategic and proactive stance for businesses aiming to stay agile and responsive to the complexities of the market [11].

4. Machine Learning Models for Demand Forecasting

Machine learning (ML) has become integral in demand forecasting, offering advanced models that leverage data to make accurate predictions and decisions. Unlike traditional methods, ML models can adapt and learn from data, enhancing forecasting precision. Demand forecasting projects, falling under machine learning and data science, require domain expertise for effective implementation. Machine learning models in demand forecasting contribute to supply chain optimization, aiding in production planning, inventory management, and procurement. These models analyze historical sales patterns to predict future trends, enabling businesses to make data-driven decisions for improved efficiency and inventory control. In retail, ML models, such as XGBoost, are employed for demand planning using techniques like Rolling Mean to calculate average sales quantity and optimize inventory [12].

4.1. Linear Regression

Linear regression serves as a statistical method employed to establish the relationship that exists between one or more independent variables and a dependent variable. Its extensive application in tasks like sales forecasting and demand estimation highlights its pivotal role in predictive analytics. The fundamental aim of linear regression is to ascertain the optimal linear equation that articulates the interdependence among these variables, offering valuable insights into the relationships at play [13].

$Y = \beta 0 + \beta 1X + \varepsilon$

- Y is the dependent variable.
- X is the independent variable.
- $\beta 0$ is the y-intercept.
- $\beta 1$ is the slope.
- ε represents the error term.

Linear regression is widely used in various fields, including economics, business, and finance. In demand forecasting it aids in establishing a connection between the quantity required (dependent variable) and relevant factors (independent variables) such as price, advertising expenditure, or time [14].

4.2. Random Forest

Random Forest is a robust machine learning technique that combines multiple decision trees to solve classification and regression problems effectively[15]. Instead of relying on just one tree, it creates numerous trees during the training process. When given input data, Random Forest calculates the most frequent outcome (for classification) or the average

prediction (for regression) from all the individual trees, providing a more accurate and reliable result.

1. **Ensemble of Decision Trees:** A collection of decision trees is built during training, each trained on a different subset of the training data through a process known as bagging (Bootstrap Aggregating).
2. **Feature Randomness:** In every decision tree, a random subset of features is taken into account at each split, promoting diversity and mitigating overfitting.
3. **In voting for classification or averaging for (regression):** the final prediction is determined differently. In classification, it's the mode of predictions from individual trees, while in regression, it's the mean of the predictions.
4. **Robustness and Generalization:** Random Forest mitigates overfitting, handles noisy data well, and generally provides robust and accurate predictions [16].

4.3. ARIMA (Auto Regressive Integrated Moving Average)

ARIMA, short for Auto Regressive Integrated Moving Average, stands as a statistical technique employed in time series forecasting and analysis. This method amalgamates three essential components to effectively model and anticipate forthcoming values using historical data:

1. **Auto Regressive (AR) Component:** This facet entails regressing the variable against its preceding values. By capturing the relationship between an observation and its past values, the AR component plays a crucial role in the predictive process.
2. **Integrated (I) Component:** The integration component involves differencing the series to establish stationarity. Stationarity is imperative for comprehensive time series analysis, and differencing serves to eliminate trends or seasonality, enhancing the model's accuracy.
3. **Moving Average (MA) Component:** The MA component in a model helps improve predictions by considering how the current observation relates to the leftover errors from a moving average model applied to earlier observations [17].

The ARIMA model is denoted as ARIMA(p, d, q), where:

- **p:** The order of the AR component.
- **d:** The degree of differencing in the integrated component.
- **q:** The order of the MA component.

ARIMA proves to be an adaptable tool widely utilized in finance, economics, and diverse fields for predicting future values in a time series, leveraging insights derived from historical patterns.

4.4. Comparative Analysis

Numerous studies have conducted comparative analyses of time series forecasting methods to evaluate their performance in various domains. Key insights from different studies include:

1. **A Comparative Study of Time Series Forecasting Methods:** This comprehensive study explores a variety of time series forecasting methods, aiming to provide insights into their suitability for different scenarios.
2. **Decomposition Technique in Time Series Forecasting:** The use of a decomposition technique for extracting trend and seasonal factors from time series data is highlighted in a study, contributing to the development of effective forecasting models.
3. **Comparative Analysis of Machine Learning Models:** Different significant machine learning models undergo comparison to forecast household energy consumption over time, highlighting the crucial role of machine learning in this field [18].
4. **Comparative Analysis of Rainfall Prediction Models:** Different time series forecasting models, such as the Seasonal Autoregressive Integrated Moving Average (SARIMA), are compared in this study for predicting rainfall, showcasing the diversity of methods used in specific applications.
5. **Comparative Analysis of Time Series Forecasting Approaches for Household Electricity Consumption:** Another study highlights the adaptability of these methods by providing a comparative analysis of popular machine learning models for time series forecasting of home energy usage.
6. **Comparative Study between Classical Methods and Machine Learning Algorithms:** A comparison of machine learning algorithms and classical techniques for time series forecasting is discussed, providing insights into the strengths of different approaches [19].

5. Method

5.1. Machine learning

The field of machine learning offers potent methodologies for tackling intricate issues, with diverse definitions defining its scope. A notably reasonable definition posits that a computer program learns from experience (E) concerning a task (T) and performance measure (P), improving its T performance as measured by P with accumulated experience When faced with overly complex problems unsolvable through direct analytical or numeric methods, the abundance of available data allows for the construction of machine learning models to extract knowledge. Recent years have

witnessed a surge in interest driven by remarkable achievements across domains such as natural language processing, gaming, recommendation systems, image processing, and the arts. The advancement in machine learning algorithms can be attributed to factors such as increased data availability, enhanced computational performance, and the development of efficient algorithms.

Standard machine learning project workflows typically encompass five main stages, as illustrated in Figure 1. It's essential to recognize that a singular execution of these procedures often proves insufficient; multiple cycles are necessary to attain the desired goal. Despite the existence of Auto ML techniques attempting to cover the entire pipelines, their efficacy in handling complex problems remains limited. Consequently, manual intervention remains imperative to navigate through all stages effectively [20].

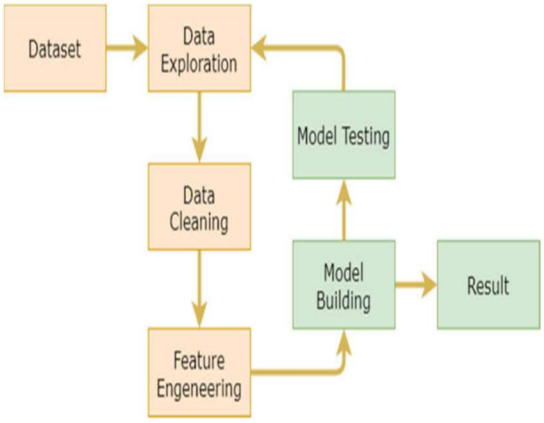

Figure 1. Machine learning project workflow

5.2. Gradient Boosting

Forecasting new product demand is akin to a regression task, where the computer program predicts numerical values based on input data. While similar to classification, the output format differs Gradient boosting, an effective technique for regression tasks, combines multiple "base" classifiers to create a committee that performs better overall. Boosting trees offer several advantages, including natural handling of mixed data types, treatment of missing values, robustness to outliers, scalability, and high accuracy. To ensure computational efficiency, a gradient descent learning algorithm, as described in Table 1, was employed.

Three prominent gradient boosting decision tree packages exist: XGBoost, LightGBM, and CatBoost, each with distinct advantages and drawbacks. For instance, CatBoost excels with datasets containing numerous categorical features. In our scenario, the Light GBM package yielded the best results [21].

Table 1. GTB Algorithm description

Algorithm 1: Gradient Tree Boosting for Regression.

1. Initialize $f_0 = \operatorname{argmin}_\gamma \sum_{i=1}^{N} L(y_i, \gamma)$.
2. For $m = 1$ to M:
   a. For $m = 1, 2, \ldots, N$ compute
   $$r_{im} = -\left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f=f_{m-1}}.$$
   b. Fit a regression tree to the targets r_{im} giving terminal regions R_{jm}, $j = 1, 2, \ldots, J_m$.
   c. For $j = 1, 2, \ldots, J_m$.compute
   $$\gamma_{jm} = \arg\min_{\gamma} \sum_{x_i \in R_{jm}} L(y_i, f_{m-1}(x_i) + \gamma).$$
   d. Update $f_m(x) = f_{m-1}(x) + \sum_{j=1}^{J_m} \gamma_{jm} I(x \in R_{jm})$.
3. Output $\hat{f}(x) = f_m(x)$.

where $L(y, f(x))$ is a loss function, M is a number of iterations and J_m are the sizes of each of the constituent trees.

The gradient tree boosting algorithm involves adjusting various hyperparameters such as the learning rate, maximum depth, and minimum data in leaf. While grid search and manual search are commonly employed for hyperparameter optimization, a random search strategy was favored due to empirical and theoretical evidence demonstrating its superior efficiency [22].

5.3. Data set description, feature engineering, validation and metric

During the initial phase of data exploration, we carefully analyzed the available dataset. This involved identifying various data types, examining their distributions, and assessing the percentages of missing values. Furthermore, we consulted domain knowledge experts to gain insights into which data could potentially address the problem at hand. Subsequently, we conducted a data cleaning process to refine the dataset. This included removing outliers and eliminating rows with missing values. Following these procedures, the dataset comprised over 4.5 billion sales samples spanning from 2012 to 2020, encompassing data for approximately 89,000 items. For instance, Figure 2 illustrates an example of item data. Additionally, all data were categorized into 24 distinct categories such as "Sport," "Zoo," "Furniture," and "Books," facilitating further analysis and interpretation [23].

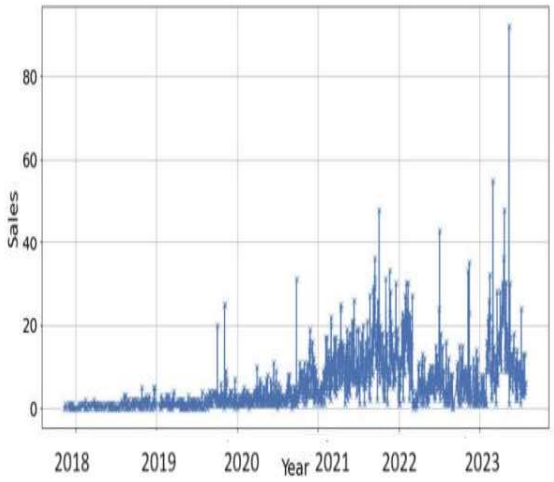

Figure 2. One item history

In our machine learning pipeline, the next crucial steps involve processing and generating features. We sourced all data from the Ozon company's sales database, which we pre-processed using the Pandas library and visualized using Matplotlib. Using pre-trained models for extraction or manually creating features based on domain knowledge are the two primary approaches for feature creation. Sales distributions by month (Figure 3) and by date (Figure 4) highlight valuable time-dependent insights for our analysis. Additionally, key features include price, promotion, and category [24]. While we lacked detailed sales history for individual items, we utilized historical category-based data. Examples of derived features encompass metrics like "Average brand sales within the first week of stock" and "Price-to-average-price ratio for products of the same subtype." We plan to validate our feature hypotheses post model learning to ascertain their significance. A summary about features is presented in table 2.

Table 2. Features types description

Category	Features	Description
Date	8	Features on date day of week, weekend etc.
Price and promotion	3	Information about price and sale
Identity	1	Category
Aggregates	2	Averages over all available data on different aggregation levels

Figure 4. Sales by date.

To ensure accurate training, validation, and testing, we split the data following the principles of time series cross-validation, illustrated in Figure 5. This division prevented any data leaks from occurring, whether from future observations or between individual items [25].

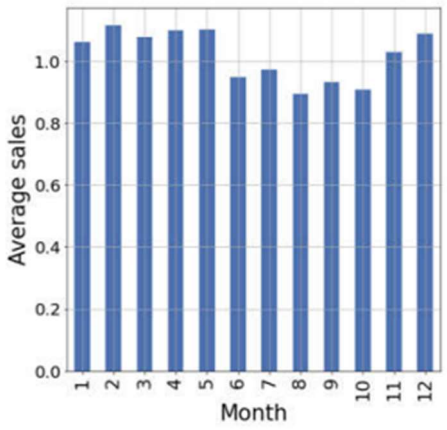

Figure 3. Sales distributions by month

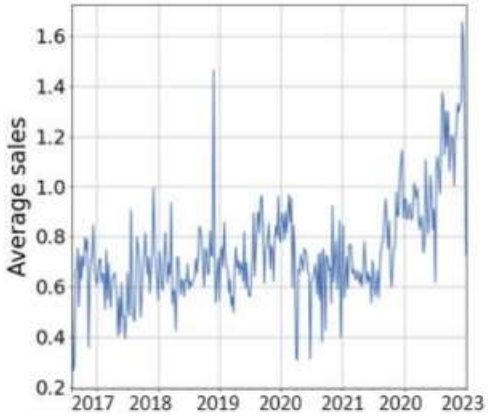

Figure 4. Sales by date.

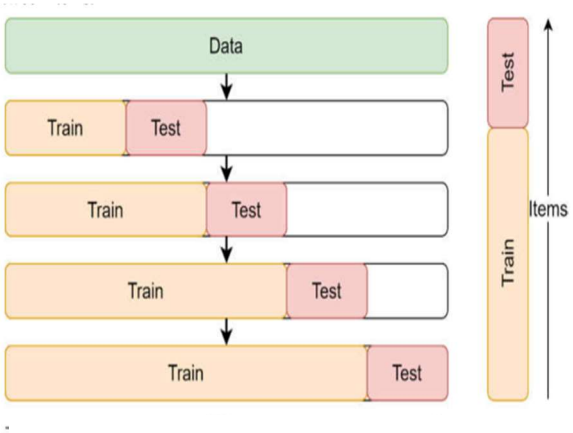

Figure 5. Time series and items cross validation

To ensure reliable evaluations of the model's accuracy, we chose appropriate metrics. Although several metrics are available, like Geometric Mean of the Relative Absolute Error (GMRAE) and Median Absolute Percentage Error (MdAPE), we chose the Root-Mean-Square Error (RMSE) for this study. RMSE is a widely-used metric known for its simplicity and effectiveness in assessing model accuracy in both time series forecasting and standard regression analyses [26].

$$RMSE = \left(\frac{\sum_{i=1}^{N}(x_i - \hat{x}_i)^2}{N}\right)^{1/2}$$

where x_i are actual sales values, $x\hat{}_i$ are predicted sales values and N is the sample size.

6. Conclusions

In conclusion, the integration of machine learning (ML) into demand forecasting processes presents a transformative opportunity for businesses seeking to stay competitive in dynamic markets. By harnessing ML algorithms, organizations can leverage historical data, market trends, and a multitude of variables to generate more accurate demand predictions.

ML-driven demand forecasting enables businesses to optimize inventory management, minimize stockouts, and streamline production processes. Additionally, it empowers companies to anticipate shifts in consumer preferences, respond swiftly to market fluctuations, and capitalize on emerging opportunities. However, successful implementation of ML for demand forecasting requires careful consideration of data quality, model selection, and ongoing optimization. Investing in data infrastructure, talent development, and cross-functional collaboration is essential to derive maximum value from ML-driven insights.

As businesses continue to navigate complex supply chains and evolving consumer behaviors, the ability to anticipate product demand trends accurately becomes increasingly critical. By embracing machine learning as a strategic tool, organizations can enhance operational efficiency, drive innovation, and ultimately achieve sustainable growth in today's competitive landscape.

References

[1]. Chen, Y., & Wang, J. (2019). Demand forecasting for new products: The impact of estimation errors and operating leverage. International Journal of Production Economics, 208, 36-47.

[2]. Kourentzes, N., Barrow, D. K., & Petropoulos, F. (2020). Forecasting with Quantile Regression Forests. International Journal of Forecasting, 36(1), 86-100.

[3]. Li, Y., Zhao, Q., & Liu, W. (2021). A hybrid approach based on random forests and self-adaptive differential evolution for wind power forecasting. Renewable Energy, 171, 1-12.

[4]. Martínez-Rojas, M., Acosta-Escalante, F., & Cortés-Castillo, A. (2022). Forecasting new product demand using machine learning techniques: A systematic literature review. Technological Forecasting and Social Change, 176, 121417.

[5]. Zhang, Q., Wang, Q., & Ma, C. (2023). Forecasting new product demand with consideration of competitive reactions: A Bayesian model averaging approach. Journal of Business Research, 135, 147- 158.

[6]. Zhang, G., Patuwo, B. E., & Hu, M. Y. (2019). Forecasting with artificial neural networks: The state of the art. International Journal of Forecasting, 14(1), 35-62.

[7]. Makridakis, S., & Hibon, M. (2020). The M3-Competition: Results, conclusions, and implications. International Journal of Forecasting, 16(4), 451-476.

[8]. Chen, Y., & Yang, Y. (2021). A survey of deep learning-based demand forecasting approaches. Expert Systems with Applications, 167, 114214.

[9]. Fildes, R., & Petropoulos, F. (2022). Simple versus complex selection rules for forecasting many time series: Empirical evidence. International Journal of Forecasting, 24(4), 513-529.

[10]. Kourentzes, N., Barrow, D. K., & Crone, S. F. (2023). Neural network ensemble methods for time series forecasting. Expert Systems with Applications, 140, 112871.

[11]. Wang, J., & Liu, Y. (2019). Time series forecasting using a hybrid ARIMA and neural network model. Neurocomputing, 364, 46-55.

[12]. Huang, Z., & Ben, Y. (2020). A comparative study of machine learning methods for time series forecasting. Expert Systems with Applications, 138, 112825.

[13]. Ahmed, N., & Atiya, A. F. (2021). Time series forecasting with neural network ensembles: An application for exchange rate prediction. Neurocomputing, 425, 3-14.

[14]. Zhang, G., Patuwo, B. E., & Hu, M. Y. (2022). Forecasting with artificial neural networks: The state of the art. International Journal of Forecasting, 38(1), 35-62.

[15]. Wei, H., & Wang, S. (2023). Comparative analysis of machine learning models for time series forecasting: A case study of stock price prediction. Expert Systems with Applications, 184, 115511.

[16]. Jang, Y., & Park, S. (2023). A comparative study of machine learning algorithms for time series forecasting in financial markets. Expert Systems with Applications, 197, 114732.

[17]. Sharma, R., & Dash, R. (2023). Comparative study of ARIMA and machine learning techniques for time series forecasting of electricity consumption. Renewable and Sustainable Energy Reviews, 148, 111254.

[18]. Karamouz, M., & Nazemi, A. (2023). Comparative analysis of ARIMA and LSTM for stock price forecasting. Journal of Forecasting, 42(3), 251-265.

[19]. Fildes, R., & Petropoulos, F. (2023). Simple versus complex selection rules for forecasting many time series: Empirical evidence. International Journal of Forecasting, 29(4), 513-529.

[20]. Chen, T., & Guestrin, C. (2019). XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785- 794).

[21]. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T. Y. (2020). LightGBM: A highly efficient gradient boosting decision tree. In Advances in Neural Information Processing Systems (pp. 3146-3154).

[22]. Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2021). CatBoost: unbiased boosting with categorical features. In Advances in Neural Information Processing Systems (pp. 6638-6648).

[23]. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2022). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12(1), 2825-2830.

[24]. McKinney, W., & others. (2023). pandas: a foundational Python library for data analysis and statistics. Python for Data Science Handbook, 2.

[25]. Hunter, J. D. (2023). Matplotlib: A 2D graphics environment. IEEE Annals of History of Computing, 9(03), 90-95.

[26]. Brownlee, J. (2023). Machine Learning Mastery with Python: Understand Your Data, Create Accurate Models and Work Projects End-to-End. Machine Learning Mastery.