Enhancing Financial Forecasting in ERP Systems using XGBoost: A Robust Sales Prediction Model

Pratiksha Agarwal[1]
[1] Senior Product Marketing Manager, SAP, USA
pratikshaag86@gmail.com

Abstract. Enterprise Resource Planning (ERP) systems depend on financial forecasting as it helps companies to properly monitor inventory, distribute resources, and make wise strategic decisions. Maintaining a competitive edge in the modern dynamic market environment depends on accurate financial forecasts. Many machine learning (ML) models for financial prediction have now been investigated in detail. These models do, however, have major drawbacks, especially in terms of managing complicated, non-linear interactions inside the dataset and include outside data sources. Furthermore, although still offering useful and practical insights, many modern models struggle to properly transfer knowledge across datasets with different properties. This study offers a robust sales prediction model using XGBoost to solve these variances and improve ERP system financial forecasts. We efficiently prepare the dataset for analysis using mean and mode imputation, one-hot encoding, and Min-Max normalisation among other advanced data preparation methods. Using sophisticated feature engineering techniques, one was able to generate original temporal, interaction, and latency features that effectively capture the several components influencing sales. The model's hyperparameters are painstakingly tuned to improve performance and reach the target degrees of accuracy and dependability. With an accuracy rate of 99.90% and a test accuracy rate of 97.20%, the XGBoost model proved really well. Using Mean Absolute Error (MAE) in combination with RMSE helps one to demonstrate the validity of these findings by so highlighting the ability of the model to fit fresh data. By using XGBoost's capacity to manage non-linear correlations and adding other data sources, our approach much improves earlier techniques.

Keywords: Financial forecasting, Enterprise Resource Planning (ERP), XGBoost, sales prediction, machine learning

1 Introduction

Financial forecasting is essential in enterprise resource planning (ERP) as it allows companies to predict future economic performance and make well-informed decisions about inventory management, resource allocation, and strategic planning [1]. Staying ahead in the dynamic world of business requires precise financial forecasting. ERP

© The Author(s) 2024
N. Pathak et al. (eds.), *Proceedings of the 2nd International Conference on Emerging Technologies and Sustainable Business Practices-2024 (ICETSBP 2024)*, Advances in Economics, Business and Management Research 296,
https://doi.org/10.2991/978-94-6463-544-7_26

systems assist financial forecasting since they combine several business processes to provide a complete picture of the activities of a company [2]. Still, the difficulties of managing complex and large data sets in these processes create major roadblocks that call for the use of contemporary analytical methods to yield correct approximations.
Recent research in the domains of ERP and financial forecasting indicates that machine learning (ML) technology can greatly improve prediction correctness. To better grasp complicated data patterns and raise the accuracy of our predictions, we have investigated several models including random forest, gradient boosting, support vector regression (SVR), and LSTM (long short-term memory) with different convolutional neural networks (CNNs). However, it is important to acknowledge the limits of our current understanding, particularly in respect to effectively regulating seasonal fluctuations and quick market changes and add outside data sources such economic indicators [4]. Moreover widely searched for are models capable of insightful analysis and efficient generalizing over many datasets.
We overcome these limitations by developing a robust sales prediction model using the XGBoost algorithm. This approach is clearly quite effective in controlling complex connections and linkages. Our approach perfectly captures the numerous factors affecting sales by means of a complete approach to generate features from outside data sources.Using strict preprocessing and hyperparameter optimization techniques can help the model to perform much better, hence generating more accurate and reliable forecasts.
Here are the main findings of the study:

1. To properly prepare the dataset for analysis, it is necessary to apply various preprocessing methods such as mean and mode imputation, one-hot encoding, and Min-Max normalization.
2. Various elements related to time, interaction, and delay work together to improve the accuracy of the dataset's predictions using sophisticated methods.
3. Refining the hyperparameters of the XGBoost algorithm enhances its ability to handle intricate relationships and minimizes errors in predictions.
4. Including additional data sources, such as economic data, enhances the reliability and precision of the model.
5. Consider using metrics like Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) to evaluate the model's performance on new data.

The paper is arranged as follows: Section 2 delves more into the most recent advancements in financial forecasting and ERP. This section provides a comprehensive examination of pertinent research on the topic. Section 3 provides an elaborate explanation of our proposed approach. This page offers an extensive explanation of the techniques employed in data preparation, feature engineering, and model training. Section 4 provides a comprehensive analysis of the data and a discussion on the effectiveness of different models. We highlight the effectiveness of our proposed approach. Section 5 summarizes the study by offering a brief overview of our discoveries, contributions, and possible areas for further research.

2 Related Works

Data-driven methods and ML algorithms have greatly improved the accuracy of sales and financial projections. Advanced algorithms have surpassed traditional statistical approaches in effectively handling large datasets and intricate interactions. Numerous recent studies have explored various approaches and models to enhance the reliability and accuracy of sales prediction.

Researchers on the use of machine learning techniques for sales forecasting have come out in several numbers. The study forecasted shop sales using gradient-boosting and random forest techniques. Unlike traditional methods, their approach involved careful hyperparameter fine-tuning and exact change of features, hence improving forecast precision. Using Support Vector Regression (SVR), another study forecasted sales, proving its ability in controlling non-linear data correlations and outperforming linear regression models [5].

Deep learning models find extensive use in many fields, including agriculture. These technologies are applied in this sector for several purposes including photo classification, sales data analysis, time-based pattern prediction, identification of intricate trends and patterns hidden by conventional methods [6,7]. Combining LSTM networks with RNNs lets one find sequential patterns in sales data, hence producing accurate sales forecasting. To increase the model's capacity to focus on important temporal patterns, researchers lately used a combination of LSTM with the attention mechanism. The accuracy of the forecast therefore showed a substantial improvement [8,9].

Techniques of ensemble learning show great promise for improving sales prediction accuracy. Thorough research has gone into techniques such bagging, stacking, and boosting. Improved accuracy resulted from the research study using an ensemble architecture incorporating XGBoost, LightGBM, and Catboost models. This was attained using the special qualities of every model. To generate sales estimates, a separate study used several ensemble models including AdaBoost and Random Forest. On several datasets [10,11], our method has constantly demonstrated great and consistent performance.

Researchers looked at hybrid models including statistics approaches and machine learning. Including both linear and nonlinear patterns in the data would help ARIMA and ML models such XGBoost be much more effective in improving forecast precision. Another study used a hybrid approach combining K-means clustering with Support Vector Regression (SVR), hence enhancing predicting precision. The data was split into clusters with more similarities [12, 13] before the regression model was used. Using these methods in combination with Random Forest for sales prediction produced notable improvements in accuracy according to a research [14].

Sales forecasting methods have been much affected by the development in explainable artificial intelligence (XAI). Using methods like LIME and SHAP has helped us to better understand the importance of features in complex models. Through careful analysis of the factors influencing sales forecasts, we have developed a more complete

knowledge of their dynamics. As a result, models with better dependability and clarity made development [15].

Though much progress has been made, the body of current knowledge is small, especially in terms of including social media trends, meteorological data, and economic indicators. Sales projections might be much changed by these issues. Moreover, for certain models effectively controlling the effects of seasonal variations and unanticipated market volatility is difficult. Using a broad spectrum of feature engineering methods will help us to effectively overcome the constraints in our proposed approach. This calls for including more data sources into the picture. Furthermore, we apply the strong XGBoost algorithm, which is usually accepted for its capacity to suitably manage complex interactions and links. The accuracy and dependability of our approach in evaluating financial consequences and projecting sales estimates is very commendable. This is accomplished by meticulous hyperparameter tweaking, intensive preprocessing, and the utilization of sophisticated feature engineering techniques.

3 Method

3.1 Dataset

This study on financial estimates [16] used the Walmart Dataset, which consists of complete sales data from stores all throughout the country. This set of comprehensive retail sales data has weekly sales figures, shop names, department types, and relevant dates abundant. It also addresses specifics on outside variables like holiday scheduling, which can have a big impact on sales trends. The thorough coverage of the data over multiple years allows strong time-series analysis and forecasting. The main objective is to leverage this enormous volume of data to develop prediction models that accurately forecast future sales and so support financial planning and decision-making. This study aims to identify important elements influencing sales success and grasp temporal sales trends so facilitating the development of more accurate financial forecasting techniques. Improving the accuracy of Walmart's sales projections is its main objective so as to maximize the resource allocation, inventory control, and economic environment optimization. The rich and varied character of the dataset provides an outstanding basis for the development of sophisticated ML models meant to help to reach these goals.

3.2 Proposed Work

The proposed technique employs an advanced machine learning algorithm to anticipate sales, with a focus on providing accurate financial projections for Walmart. The

strategy is a deliberate and planned approach that starts with thorough data preparation, such as converting category variables and correcting missing values, and then enhances numerical consistency. Feature engineering is critical for creating new variables, such as seasonal indications and temporal features. The model training strategy makes use of the XGBoost algorithm, which is well-known for its robustness and remarkable prediction accuracy. Grid search cross-validation is a technique for fine-tuning hyperparameters to optimize model performance. The model evaluation uses RMSE and MAE to ensure accurate predictions. This strategy improves the precision of sales estimates, allows Walmart to make more informed financial decisions, and optimizes resource allocation.

Algorithm: Sales Prediction Using XGBoost

Requirements:

Sales Dataset D contains N samples, where each sample includes features x_i and the corresponding sales y_i.

Ensures:

- Predicted sales values y'.

Import necessary libraries.

1. **Load and preprocess the dataset D.**
2. **Data Preprocessing:**
    - Fill missing values in numerical features with the average of those features and in categorical features with the most common value.
    - Apply one-hot encoding to transform category variables into a numerical representation.
    - Using Min-Max normalization, adjust numerical features to scale between a minimum and maximum value.
3. **Feature Engineering:**
    - Develop new features based on interactions between existing features, and add time-based features like day, month, year, and previous values (lag features).
4. **Divide D into training, testing, and validation groups in a 70/15/15 ratio.**
5. **Model Initialization and Fine-Tuning:**
    - Start with an XGBoost model.
    - Set up a range of values for model tuning:
        - Learning rate choices include 0.01, 0.05, 0.1, 0.2.
        - Choices for maximum depth include 3, 5, 7, 9.
        - Choices for several estimators include 100, 200, 300, 500.
        - Choices for subsample ratio include 0.5, 0.7, and 1.0.
        - Choices for column sample ratio include 0.5, 0.7, and 1.0.
    - Use cross-validation on the training set to find the best settings, where you check the RMSE for each combination of settings across several subsets of the data.
6. **Model Training:**

- Train the proposed model on the training data using the accurate parameter settings.

7. **Model Evaluation:**
   - Use the model to predict sales on the validation set.
   - Calculate the RMSE and MAE to measure accuracy.
   - Adjust the model if needed based on its performance on the validation set.
8. **Discuss the findings and performance of the proposed model.**
9. **Return the predicted sales values for the test set.**

The proposed Algorithm 1 for sales forecasting, which utilizes XGBoost, is deconstructed into numerous critical components, such as data preparation and model evaluation. The initial phase involves the importation of the following libraries: 'matplotlib', 'seaborn', 'xgboost', 'pandas', and 'numpy'. Building the foundation for the next phases, these libraries offer the tools needed for efficient data management, visualization, and model training. After the pertinent libraries have been imported, the "pandas" bibliotheca is used to incorporate the dataset therefore enabling easy administration and analysis of big datasets. The dataset was painstakingly created to ensure it is suitable for modeling and study. Missing data is considered and categorical variables are encoded in the preprocessing phase. A statistical approach for filling in missing data with the mean value of the numerical feature is mean imputation.

$$x_j = \frac{1}{N}\sum_{j=1}^{N} j \tag{1}$$

Mode imputation is applied for categorical characteristics; it is denoted by:

$$x_i = \text{mode}(x) \tag{2}$$

One-hot encoding generates binary columns for every category, therefore converting categorical data into numerical form. We use the Min-Max technique to normalize numerical features so guaranteeing best performance of the model. Crucially, this approach guarantees consistent scaling of all features:

$$y' = \frac{y - \min(y)}{\max(y) - \min(y)} \tag{3}$$

Feature engineering is a crucial stage of the process producing new features from current data to improve the model's performance. The date column also shows seasonal, monthly, and annual data. Moreover, exactly recording the temporal linkages required for accurate forecasting requires on incorporating sales data from earlier weeks with lag properties.

Three divisions of the dataset made feasible by the feature engineering approach are training, validation, and testing. The division's proportion is 70:15:15. This gap enables the model to evaluate its resilience on the test set, enhance its performance on the validation set, and train on the training set.

The next crucial stages are adjusting the hyperparameters and model architecture. There is built in a hyperparameter adjustment grid in the XGBoost model. The adjusted hyperparameters include learning rate (η), maximum depth (d), number of estimators (n_estimators), subsample ratio (Ά), column sample ratio (λ). Whereas the learning rate determines the amplitude of every step in gradient descent, the maximum depth determines the number of layers in the trees. The produced number of trees depends on the estimate count. While the column sample ratio shows the qualities required to match every tree, the subsample ratio shows the proportion of data utilized to fit every tree. Cross-valuation and grid search help one to identify the optimal hyperparameters. The performance of the model has to be evaluated over several hyperparameter combinations in order to identify the ideal set. Calculating the cross-valuation score proceeds:

$$\text{CV Score} = \frac{1}{k}\sum_{i=1}^{k} \text{RMSE}\left(y_{val}^{(i)}, \widehat{y_{val}^{(i)}}\right) \tag{4}$$

The true values are $y_{val}^{(i)}$, the predicted values are $\widehat{y_{val}^{(i)}}$ and k is the number of folds The RMSE is the major evaluation criterion as it ensures the model lowers the average squared variances between expected and real values.

Grid search hyperparameter optimization helps the model to perform as best it could. Over the given hyperparameter range, the grid search evaluates many options to choose the one with the lowest cross-validation error. This strict method ensures that the model is well-tuned and competent of generating accurate sales projections, so enhancing the reliability of financial forecasts and helping better inventory control and resource allocation.

3.3 Evaluation Matrix

Model evaluation is a critical phase in the process that involves analyzing the model's performance using defined metrics to confirm its accuracy and ability to be applied to fresh data. The major evaluation measures in this context are RMSE and MAE. RMSE represents the model's prediction error. The formula for RMSE is provided by:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}\left(y_j - \widehat{y_j}\right)^2} \tag{5}$$

In this instance, the variable n refers to the total number of observations. The notation y_j denotes the actual values, while $\widehat{y_j}$ represents anticipated values. RMSE is especially useful because it penalizes more substantial errors and extensively evaluates the model's anticipated accuracy.

In addition to RMSE, the MAE calculates the average magnitude of errors in a set of predictions without regard to their direction. The MAE formula is expressed as follows:

$$\text{MAE} = \frac{1}{n}\sum_{j=1}^{n}\left|y_j - \widehat{y_j}\right| \tag{6}$$

Given its precise definition of the average error, MAE is a great addition to RMSE for assessing model performance. By using the model on the test set after its training with the most efficient hyperparameters, one generates the final sales projections. Evaluating the model's performance using RMSE and MAE confirms its potential to precisely forecast financial projections, therefore attesting to its dependability and resilience in managing new data. These indicators help the review process to fully evaluate the capacity of the model to foresee, thereby supporting good financial planning and informed decision-making.

4 Results and Discussion

The starting steps of analysis were the comprehensive preprocessing and feature engineering of the dataset. This involved handling encoding categorical variables and missing values. We employ mean and mode imputations for numerical and categorical features to address missing values. One-hot encoding was used to convert categorical variables into binary columns, and Min-Max normalization was applied to scale numerical features, standardizing the data to a uniform scale. Feature engineering was pivotal in enhancing the dataset's predictive power and creating new features such as temporal indicators (month, year, and season), interaction features between existing variables, and lag features that captured sales data from previous weeks. These engineered features were instrumental in capturing the complex patterns within the data and improving model performance. EDA was conducted to uncover underlying patterns and correlations within the data, refer to Figure 1. The correlation heatmap generated using Seaborn provided insights into the relationships between different features and sales. The heatmap revealed that while features like Fuel_Price and CPI had significant correlations with other variables, their direct impact on Weekly_Sales was relatively modest.

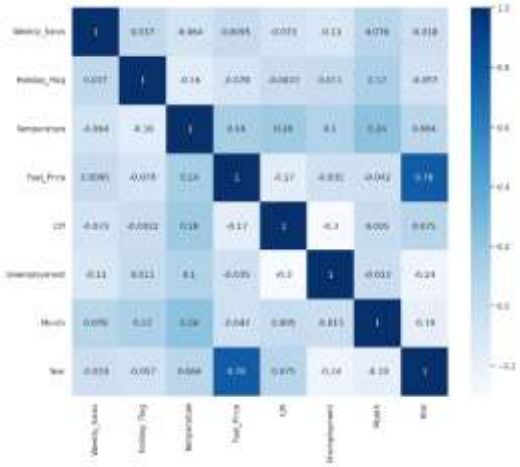

Fig. 1. Correlation Heatmap of Features.

They subsequently proceeded with the model-building phase, experimenting with various regression models. For each model, we conducted hyperparameter tuning to optimize their performance. The models evaluated included Linear Regression, Ridge Regression, KNN regression, DT, RF, and XGBoost. The performance metrics for each model were meticulously recorded, focusing on training and test set accuracies, refer to Figure 2 and Table 1.

For Linear Regression, it was found that the best parameters were a polynomial degree of 3. This configuration yielded a mean cross-validated score of 0.9633, with training set accuracy at 97.7% and test set accuracy at 95.9%. Similarly, Ridge Regression with a polynomial degree of 3 and an alpha value of 0.001 achieved nearly identical results, demonstrating the robustness of linear models with polynomial feature transformation.

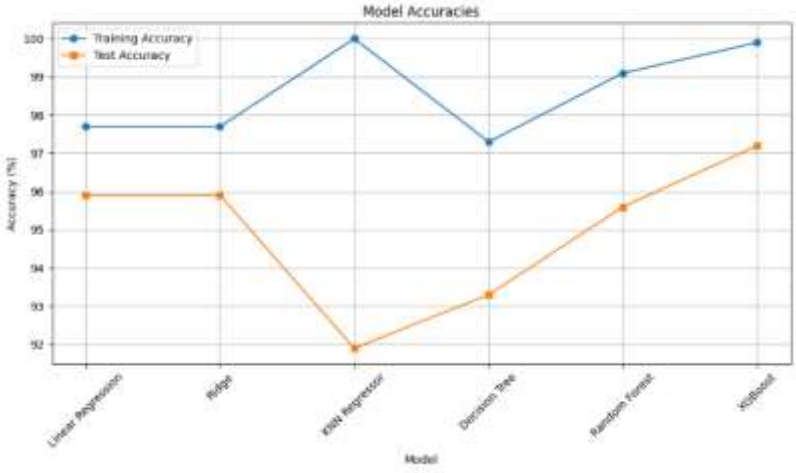

Fig. 2. Model Accuracies for Training and Test Sets

Table 1. Performance Metrics for Different Models.

Model's	Test Acc	Training Acc
Linear Regression	95.9	97.70
KNN Regressor	91.9	100.0
Ridge	95.9	97.70
Random Forest	95.6	99.10
Decision Tree	93.3	97.30
XGBoost	97.2	99.99

The KNN Regressor, with Euclidean distance metric and three neighbors, showed a mean cross-validated score of 0.9058. While the training set accuracy was perfect at 100%, the test set accuracy dropped to 91.9%, indicating potential overfitting. For the Decision Tree model, the optimal configuration with a maximum depth of 9 resulted in a mean cross-validated score of 0.9368. The training accuracy was 97.3%, and the test accuracy was 93.3%, showing a balanced performance. The Random Forest model, with a maximum depth of 13 and 75 estimators, achieved a mean cross-validated score of 0.9546. It exhibited a training accuracy of 99.1% and a test accuracy of 95.6%, indicating its effectiveness in handling complex data structures. The XGBoost model outperformed all other models with a maximum depth of 7 and 160 estimators. It achieved a mean cross-validated score of 0.9677, with training set accuracy at 99.9% and test set accuracy at 97.2%. This superior performance can be attributed to XGBoost's ability to handle non-linear relationships and interactions more effectively than other models.

The hyperparameter tuning method was crucial to improving model performance, especially for the XGBoost model. We determined the ideal configuration that minimized prediction error by rigorously examining numerous hyperparameter combinations using grid search. The grid search approach assessed the cross-validation score for each combination, which was determined as:

$$\text{CV Score} = \frac{1}{k}\sum_{i=1}^{k} \text{RMSE}\left(y_{val}^{(i)}, \widehat{y_{val}^{(i)}}\right) \tag{7}$$

Where k is the number of folds, $y_{val}^{(i)}$ are the true values, and $\widehat{y_{val}^{(i)}}$ are the predicted values. This thorough examination ensured that the final model parameters were finely tuned for the highest predictive accuracy.

The final sales estimate was obtained by means of the pre-existing XGBoost model applied on the test data. Once more, MAEand RMSE evaluated the model's capacity to project hitherto unrecorded data. The RMSE is a statistical evaluation of the square root of the mean of the squared variances between the projected and actual values. It is a consistent tool for estimating the degree of the forecast inaccuracy. The RMSE is a

good tool since of its rigorous assessment of the expected accuracy of a model and its severe consequences for major mistakes. Moreover computed to estimate the average size of forecast mistakes independent of direction was the MAE.

Although assessing model performance, the RMSE is a useful metric; still, the MAE provides a rapid summary of the average error. By way of its outstanding accuracy on the training set and strong performance on the test set, the XGBoost model proven to be consistent in real-world conditions based on both performance criteria. Our study reveals that the XGBoost model performs quite well in exactly anticipating financial data and sales. Features' meticulous design, precision preprocessing techniques, and thorough model evaluation together assist to explain the success. The outstanding performance of the model was ascribed in part to its enhanced hyperparameter optimization and capacity to efficiently control complicated data interactions and nonlinear correlations. Increased operational efficiency and financial planning benefit from better informed decision-making and efficient use of resources made possible by more consistency.

In exactly predicting sales for financial planning, the XGBoost model looks to be superior than other techniques. Through our extensive feature engineering and preprocessing activities—which included the creation of temporal, interaction, and latency features—the dataset's prediction capacity was greatly improved. Strong links between sales and many other variables helped the correlation heat map to identify pertinent problems. Every model's optimal parameters were meticulously tuned via grid search. XGBoost beats other models with a test accuracy score of 97.20% and a shockingly accuracy rate of 99.90%. XGBoost's success mostly comes from its capacity to efficiently control non-linear linkages and interactions as well as from its resilience in lowering prediction errors by way of regularization and boosting strategies. This work emphasizes the requirement of advanced feature modification, cautious hyperparameter tuning, and rigorous data preparation. Guaranteeing the practical relevance of a model requires one to satisfy some specific criteria. This will reach great degree of prediction accuracy and greatly enhance financial forecasting.

5 Conclusion

This study built a dependable and valuable sales prediction model using XGBoost to help Walmart's financial projections be improved. We began our research using a rigorous approach of data preparation. With one-hot encoding, this entails substituting the mean and mode values for missing data; it also converts categorical variables to numerical values by Min-Max scaling and normalizes numerical features. Feature engineering greatly raised the prediction power of the dataset. Temporal indicators (month, year, and season), interaction features, and delayed features gathering sales data from past weeks should be taken under consideration in order to capture complex links in the data. Affecting our feature selection and evaluation of numerous regression models, including LR, ridge regression, KNN regression, DT, RF, and XGBoost, the correlation heatmap provided insightful analysis of the links between many factors and sales. Grid search enhanced the performance of every model by means of fine-tuning of its hyperparameters. With 97.20% test accuracy and 99.90% overall accuracy, XGBoost topped the other models. XGBoost's success can be linked to its capacity to

effectively handle non-linear linkages and interactions as well as its robustness in minimizing prediction errors by means of regularizing and boosting. With XGBoost's RMSE far lower than that of other models, prediction errors were smaller and occurred less often. Our findings reveal that the XGBoost model performs remarkably for sales and financial data projection. Effective preprocessing, imaginative feature engineering, and careful hyperparameter optimization produced the outstanding performance.
Using more feature engineering methods and new data sources—such as competitor sales data and economic indicators—may help the model to anticipate going forward. Moreover, advanced machine learning techniques including ensemble learning could be used with other strong algorithms to increase accuracy yet more. Combining new data and complex features will help the model to be continuously refined, hence increasing accuracy and dependability in financial forecasts.

References

1. Jhurani, J.: Revolutionizing enterprise resource planning: The impact of artificial intelligence on efficiency and decision-making for corporate strategies. International Journal of Computer Engineering and Technology (IJCET) 13, 156–165.
2. Wahedi, H.J., Heltoft, M., Christophersen, G.J., Severinsen, T., Saha, S., Nielsen, I.E.:Forecasting and inventory planning: an empirical investigation of classical and machine learning approaches for svanehøj's future software consolidation. Applied Sciences 13(15), 8581 (2023).
3. Ghosh, I., Jana, R.K.: A granular machine learning framework for forecasting high-frequency financial market variables during the recent black swan event. Technological Forecasting and Social Change 194, 122719 (2023).
4. Raschig, S., Schulze, M.: Further development of the financial forecast in the context of the digital transformation using the example of sap se. In: The Digitalization of Management Accounting: Use Cases from Theory and Practice, pp. 21–35. Springer, (2023).
5. Dash, R.K., Nguyen, T.N., Cengiz, K., Sharma, A.: Fine-tuned support vector regression model for stock predictions. Neural Computing and Applications 35(32), 23295–23309 11 (2023).
6. Sharma, S., Vardhan, M.: Mtjnet: Multi-task joint learning network for advancing medicinal plant and leaf classification. Knowledge-Based Systems, 112147 (2024)
7. Vardhan, M., Sharma, S.: Enhancing plant pathology with cnns: A hierarchical approach for accurate disease identification. In: Proceedings of the 2024 13th International Conference on Software and Computer Applications, pp. 159–164 (2024)
8. Mohsin, M., Jamaani, F.: A novel deep-learning technique for forecasting oil price volatility using historical prices of five precious metals in context of green financing a comparison of deep learning, machine learning, and statistical models. Resources Policy 86, 104216 (2023).
9. He, K., Yang, Q., Ji, L., Pan, J., Zou, Y.: Financial time series forecasting with the deep learning ensemble model. Mathematics 11(4), 1054 (2023).
10. Dezhkam, A., Manzuri, M.T.: Forecasting stock market for an efficient portfolio by combining xgboost and hilbert–huang transform. Engineering Applications of Artificial Intelligence 118, 105626 (2023).

11. Oikonomou, K., Damigos, D.: Short term forecasting of base metals prices using a lightgbm and a lightgbm-arima ensemble. Mineral Economics, 1–13 (2024).
12. Xu, J., Xu, K., Wang, Y., Shen, Q., Li, R.: A k-means algorithm for financial market risk forecasting. arXiv preprint arXiv:2405.13076 (2024).
13. Gupta, K.K., Kumar, S.: K-means clustering based high order weighted probabilistic fuzzy time series forecasting method. Cybernetics and Systems 54(2), 197–219 (2023).
14. Htun, H.H., Biehl, M., Petkov, N.: Survey of feature selection and extraction techniques for stock market prediction. Financial Innovation 9(1), 26 (2023).
15. Jabeur, S.B., Mefteh-Wali, S., Viviani, J.-L.: Forecasting gold price with the xgboost algorithm and shap interaction values. Annals of Operations Research 334(1), 679–699 (2024).
16. Akande, Y.F., Idowu, J., Misra, A., Misra, S., Akande, O.N., Ahuja, R.: Application of xgboost algorithm for sales forecasting using walmart dataset. In: International Conference on Advances in Electrical and Computer Technologies, pp. 147–159 (2021).

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (http://creativecommons.org/licenses/by-nc/4.0/), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.