University of Hertfordshire UH

School of Physics, Engineering and Computer Science

MSc Data Science Project
7PAM2002

Department of Physics, Astronomy and Mathematics

Data Science FINAL PROJECT REPORT

Project Title:

Customer Behaviour Analysis and Revenue Optimization Using Online Retail Transaction Data

Student Name and SRN:

Rakesh Kummari and 23029280

Supervisor: Man-Lai Tang

Date Submitted: 29 April 2025

Word Count: 5604

GitHub Repository Link: https://github.com/Kummarirakesh/DS-Final-Project/tree/main

DECLARATION STATEMENT

This report is submitted in partial fulfilment of the requirement for the degree of Master of Science **in Data Science** at the University of Hertfordshire.

I have read the detailed guidance to students on academic integrity, misconduct and plagiarism information at Assessment Offences and Academic Misconduct and understand the University process of dealing with suspected cases of academic misconduct and the possible penalties, which could include failing the project or course.

I certify that the work submitted is my own and that any material derived or quoted from published or unpublished work of other persons has been duly acknowledged. (Ref. UPR AS/C/6.1, section 7 and UPR AS/C/5, section 3.6)

I did not use human participants in my MSc Project.

I hereby give permission for the report to be made available on module websites provided the source is acknowledged.

Student Name printed: Rakesh Kummari

Student Name signature: 

Student SRN number: 23029280

UNIVERSITY OF HERTFORDSHIRE

SCHOOL OF PHYSICS, ENGINEERING AND COMPUTER SCIENCE

Customer Behaviour Analysis and Revenue Optimization Using Online Retail Transaction Data

Abstract

This work explores the prediction of daily sales data using many regression models. To examine the sales data and assess their performance we used Random Forest Regression, Decision Tree Regression, and Linear Regression. The models were trained and validated after pre-processing the data by means of pertinent time-based characteristics. With an ideal number of estimators, the Random Forest model showed better than other models, hence lowering the Root Mean Squared Error (RMSE) on test data. To raise predictive accuracy in sales forecasting, the study emphasises the need of model tweaking and feature selection.

Table of Contents

1. Introduction 1

1.1. Overview of Online Retail Industry 1

1.2. Customer Behaviour in E-Commerce 1

1.3. Research Aim 2

1.4. Research Questions 2

1.5. Objectives 3

1.6. Ethical Considerations 3

2. Literature Review 4

3. Research Methodology 7

3.1. Overview of the Dataset 7

3.2. Data Cleaning 7

3.3. Feature Engineering 8

3.4. Data Analysis and Visualization 8

3.5. Time-series Analysis 11

3.5.1. ARIMA Model 11

3.6. Regression Analysis 11

3.6.1. Linear Regression 12

3.6.2. Decision Tree Regression 13

3.6.7. Random Forest Regression 13

4. Experimental Results 15

4.1. Time-series Analysis - ARIMA 15

4.2. Regression Analysis Results 16

4.2.1. Linear Regression Results 16

4.2.2. Decision Tree Regression Results 17

4.2.3. Random Forest Regression Results 17

5. Discussion 19
5.1. Comparison with Existing Literature 19
5.2. Limitations of the Results 19
5.3. Relation to Project Objectives 20
6. Conclusion 21
6.1. Validated Conclusions 21
6.2. Future Works 21
References 23
Appendices 26

Figures

Figure 1 - Total Sales by Hour 8
Figure 2 - Total Sales by Month 9
Figure 3 - Monthly Sales Distribution 9
Figure 4 - Timeseries of the sales 10
Figure 5 - Linear Regression Interpretation 12
Figure 6 - Impact of Max Depth on Decision Tree Regression Performance 13
Figure 7 - Random Forest Regression Interpretation 14
Figure 8 - Autocorrelation and Partial Autocorrelation of ARIMA 15
Figure 9 - ARIMA model predictions 15
Figure 10 - Baseline model predictions 16
Figure 11 - Decision Tree Regressor predictions 17
Figure 12 - Random Forest Regressor predictions 18

1. Introduction

1.1. Overview of Online Retail Industry

There has been an increase in the online retail industry for over the past decade, thanks to access to internet, mobile technology and consumer preference change. If it's not e-commerce, it's the revolution in the prospect of retail; from convenience to wide variety in the products to bring and the made to order shopping. Customer behaviour has completely changed due to trends like mobile commerce, same day delivery and subscription-based models and so on (Reddy, 2023).

The power to simplify operations, strengthen the customer experience and embrace radically new business models has been a significant enabler in this evolution, and that enabler is digital transformation (Istalingam, 2021). Cloud computing (and digital payment systems) has fundamentally changed how retailers operate and provide value. Artificial intelligence has changed how retailers but the value they provide. Here, data, has been turned into the most valuable asset in the digital ecosystem (Udell, 2020). The huge amounts of transactional and behavioural data available online means that online retailers have huge amounts of data on which patterns can be deduced of customers' preferences, and product and service predictions can be made that often sell, to help them make strategic decisions. However, if used appropriately, this information can be exploited by businesses to optimally set the price of the product offered, manage the inventory supply, develop suitable marketing tactics and subsequently enhance customer pleasure and the revenue too (Ganguly & Mukherjee, 2024).

1.2. Customer Behaviour in E-Commerce

The emergence of digital platforms has greatly changed how customers behave in e-commerce. Today's consumer is more informed, connected to the web and less selective than ever in how he decides to shop online. Convenience in shopping, wider product variety, competitive prices, and the ability to shop any time at any place have contributed to the change from traditional brick and mortar stores to online platforms (Takale et al., 2022).

Purchase in the online space is heavily influenced by several key factors. and return policies, all these include product reviews, ease of navigation, website design, payment security,

delivery speed, etc. Customer testimonials and ratings also act as a source of dark social proof to build the confidence of the buyers. Later on, other factors have an impact on how they choose to buy items; factors such as price comparison tools, targeted discounts, and other tools (Haines, 2023). However, recent years have seen customers much more reliant on personalization in creating a unique experience for them. It is expected from shoppers that the product recommendations should be tailored according to their preferences and their wants and communication should be appropriate as it speaks about their interests and preferences. These personalized experiences are delivered by the e-commerce platforms by leveraging browsing history, purchase data, as well as user behaviour (Ngui, 2023). Personalization at such a level positively affects the user's satisfaction and enhances the likelihood of repeat purchase. Furthermore, chat support and loyalty programs, as well as personalized marketing campaigns, are also offered as interactive features that strengthen customer relationships and brand loyalty, and engagement is a major strategy for e-commerce today (Kirichenko et al., 2019).

1.3. Research Aim

The aim of this project is to study customer behaviour on the basis of the online retail transactional data and study how various factors influence the revenue and build the prediction models for future sales. The study aims at finding the insights into the consumer purchasing trends and behaviour that affect profitability by looking at the historical transaction patterns. The other aim of the project is to evaluate the performance of regression models in predicting sales and developing data driven strategies for revenue optimisation. The main objective is providing data to support the e-commerce company in making smart decisions that contribute to better profitability, better customer interaction and longer-term business growth through the use of transactional data.

1.4. Research Questions

- How accurately can regression models predict future sales based on historical transaction data?
- What factors most significantly influence revenue fluctuations in online retail transactions?

1.5. Objectives

- Analyse historical online retail transaction data to identify patterns in customer purchasing behaviour.
- Develop regression models for predicting future sales based on past transaction trends.
- Determine the key factors that contribute to revenue fluctuations in e-commerce transactions.
- Evaluate the accuracy and reliability of predictive models in forecasting sales performance.
- Recommend data-driven strategies for optimizing revenue and improving business decision-making.

1.6. Ethical Considerations

Consumer privacy: The transaction information in the dataset is quite obvious which made consumer privacy extremely vital. To kill the usage, I would want to encrypt or anonymise all personally identifiable information, such as customer IDs. This would allow in ensuring responsible handling of data, so it would protect the confidence for users in protecting users' data.

Bias Free Analysis: The project has to be free of any bias in model building as well as any bias involved in data interpretation. Respect for certain customer categories should not be biased by the clustering or regression algorithms. Fairness' in insights and decisions making avoids unethical business practices and ensures fair chances for all client groups or even more.

2. Literature Review

The rise of e-commerce has led to an explosion of online retail transaction data, creating significant opportunities for analysing customer behaviour and optimizing revenue. Researchers have explored various methodologies—from data mining and predictive analytics to dynamic pricing models—to transform raw transaction data into actionable insights. This literature review synthesizes eight influential studies that contribute to understanding how online customer behaviour can be leveraged to drive revenue optimization. The selected papers cover topics such as customer relationship management (CRM) through data mining, advanced marketing analytics in data-rich environments, dynamic pricing strategies, omni-channel retailing, customer lifetime value, churn prediction, and comprehensive pricing models. Collectively, these works provide both theoretical foundations and practical applications that are critical for modern retail strategy.

Recent studies have focused on developing comprehensive frameworks to predict consumer behaviour using supervised learning techniques (Rekha et al., 2024). Researchers have demonstrated that integrating stages such as data collection, preprocessing, exploration, and feature selection significantly improves model accuracy and interpretability. In these approaches, critical predictors influencing customer moods and purchasing trends are identified, thereby enabling efficient clustering and classification of consumer behaviour. Furthermore, ethical considerations surrounding data privacy and transparency have been emphasized to maintain trust and compliance. The systematic methodology not only supports the identification of key behavioural drivers but also underscores the importance of adaptive monitoring strategies. Such predictive analysis frameworks empower marketers to tailor campaigns effectively, ultimately enhancing customer experience and sustaining a competitive edge in dynamic marketplaces (Rekha et al., 2024). Overall, these promising results underline the model's potential.

Dynamic pricing has emerged as a pivotal strategy in e-commerce to adapt to rapid market changes and boost revenue generation. Traditional pricing approaches often struggle to meet the evolving demands of digital markets, necessitating the adoption of advanced methodologies. Recent research integrates Artificial Intelligence (AI) and Machine Learning (ML) to revolutionize pricing optimization, employing sophisticated demand forecasting and real-time market assessments (Gupta, 2025). The reviewed study introduces an innovative

framework that leverages machine learning for robust demand prediction and reinforcement learning for dynamic pricing adjustments. The proposed architecture supports real-time implementation and scalability, enabling platforms to respond quickly to market variations. Experimental results demonstrate enhanced forecasting precision, a 20% revenue increase compared to conventional methods, and improved customer retention, highlighting the framework's potential for establishing competitive advantages (Gupta, 2025). These findings underscore considerable impact.

Sustainable consumer behaviour has gained increasing attention, with organizations leveraging pricing strategies to promote eco-friendly consumption. Pourmahdi et al., highlight the role of descriptive and predictive analytics in understanding and forecasting consumer preferences for sustainable products. By employing machine learning techniques such as data summarization, visualization, and logistic regression, businesses can identify key factors influencing sustainable purchasing decisions. A study focusing on a Nordic retail conglomerate from 2019 to 2024 applies these methods to analyse consumer trends. Findings suggest that data-driven approaches enable retailers to assess purchasing patterns and predict shifts toward sustainable alternatives. Such insights help organizations refine pricing strategies to encourage eco-conscious choices. Integrating predictive analytics into sustainable business models enhances decision-making, supporting both environmental goals and long-term profitability in the competitive retail landscape (Pourmahdi et al., 2025).

Koppolu et al., emphasizes the transformative role of AI and data management in driving deep monetization and revenue penetration. Traditional billing, often viewed as a cost centre, is being redefined as a strategic asset through data utilization, enabling revenue growth and operational efficiency. Studies highlight the importance of leveraging billing data for digital monetization and market pricing insights. A shift towards big data engineering platforms has been proposed, facilitating agile innovation and multidisciplinary interventions. Findings suggest that detailed contextual data holds greater predictive value than clean, aggregated datasets, broadening applications beyond gaming. A data-driven revenue engineering approach, incorporating modern revenue processing and real-world system complexities, has demonstrated significant financial benefits. Empirical results indicate over a 20% annual margin increase through reduced operational costs, underscoring the potential of AI-driven revenue strategies in diverse industries (Koppolu, 2025).

Krishnamoorthy et al., has explored advanced quantitative approaches to predicting consumer behaviour, emphasizing multi-objective evolutionary algorithms (MOEAs) for enhanced accuracy. A novel framework integrates customer preference data, employing min-max normalization to refine datasets and eliminate irrelevant information. Feature extraction is conducted using the Word2Vec model, while boosting ant colony optimization (BACO) aids in feature selection. MOEAs are then applied to improve prediction performance. Comparative evaluations reveal that this approach surpasses traditional machine learning techniques, including extreme gradient boosting (XGB), artificial intelligence (AI), and naive Bayes (HNB), in key performance metrics. The proposed model achieves high accuracy (96%), superior prediction quality (97%), and an F1 score of 99%, demonstrating its reliability. These findings highlight the viability of MOEA-based predictive models in optimizing consumer insights and sustaining long-term profitability in dynamic markets (Krishnamoorthy et al., 2023).

3. Research Methodology

3.1. Overview of the Dataset

The Online Retail dataset contains transactional data by a UK based online store from December 1 2009 to December 9 2011. That is more than a million transactions and the analysis is also very insightful of consumer buying behaviour. The retailer's speciality is in selling original giftware with both the wholesale distributors and individual consumers. The dataset is multivariate, sequential and time series data which also fit well for analytical clothes like regression and clustering. This dataset offers comprehensive transactional insights from which customer segmentation, sales forecasting and the revenue optimisation benefits can be benefited. It provides a way for companies to analyse purchasing patterns, expose the top value consumers and improve marketing and inventory control strategy decisions.

Dataset: https://archive.ics.uci.edu/dataset/502/online+retail+ii

3.2. Data Cleaning

Data preprocessing includes data cleaning which is about the data, which is a necessary step to run any analytics as it involves identifying and correcting errors, dealing with the missing values and structuring the data to be used for the analysis. The dataset is supposed to be accurate, consistent and exempt of unnecessary or redundant data for the purpose of generating authentic analytical result.

This dataset contained 525,461 records with several empty values, and this is the one dataset that was started with. There were 2,928 nulls for the 'Description' column, and 107,927 nulls for 'Customer ID'. Some columns were removed as they either had important info lacking or were not required for the analysis. Moreover, the 'Invoice Date' column (which contained date and time information) took its respective into two columns: 'Date' and 'Time', to make the analysis regarding a time much easier. The original Invoice Date column was then dropped. The data set was slowly soaked with these steps streamlining, structuring, and preparing for further exploration and modelling of them.

3.3. Feature Engineering

Feature engineering is the process of transforming the raw data into actual inputs which improve the performance of machine learning models. This implies creation of new variables, modification of existing ones, or elimination of irrelevant features for better capability of model in pattern identification and correct prediction. Feature engineering leads to the discovery of the hidden relations in the data and makes it fit for the purpose of the analysis.

To analyse and model the dataset, several feature engineering steps were applied on this dataset. Second, I removed data on the transactions that occurred from accounts not situated in the UK as they accounted for a very little amount of input data. Secondly, a new feature was created, called Sales, which is the quantity of things sold multiplied by their price, removing the effect of negative entries, albeit absolute value is used. The 'original timestamp' was spitted into 'Date' and 'Time' columns. From these additional temporal features were extracted, including 'Hour', 'Day', 'Month', 'Quarter' and 'Year' thus allowing for detailed time-based analysis. Also, the 'Month' values were transformed in to more readable month names for more interpretability. Finally, as noise, columns that are non-essential such as Invoice and StockCode were dropped.

3.4. Data Analysis and Visualization

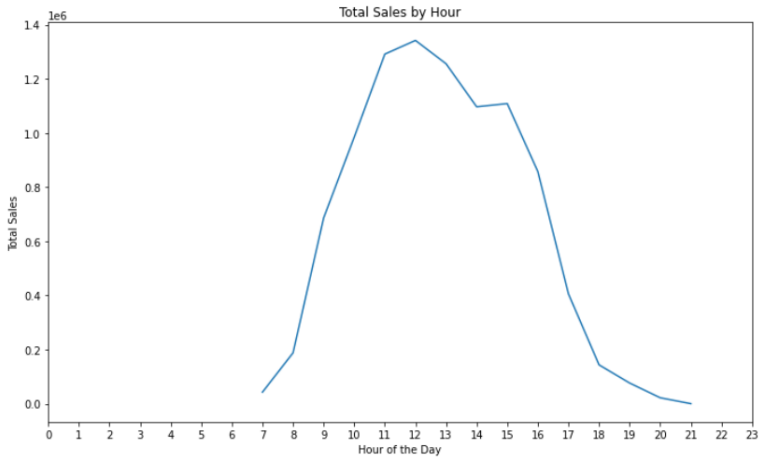

Figure 1 - Total Sales by Hour

Multiple visualizations explain customer behaviour and how it impacts business strategy through the online retail transaction data. Initially a rhythmic pattern was observed in daily purchase behaviour on the line plot of Total Sales by Hour. After 7 AM, sales go up quickly till noon, hitting their peak. It’s not strange that the nearest peak of such consumption occurs in

the midday, which is possible due to lunch breaks or some sort of scheduling shopping routine. Sales fall with the passing of the day, particularly after 8 PM. The ebb and flow of sales that retailers experience means that they receive a critical insight on what times are most active for consumers – to better schedule promotions, inventory availability, and staffing.

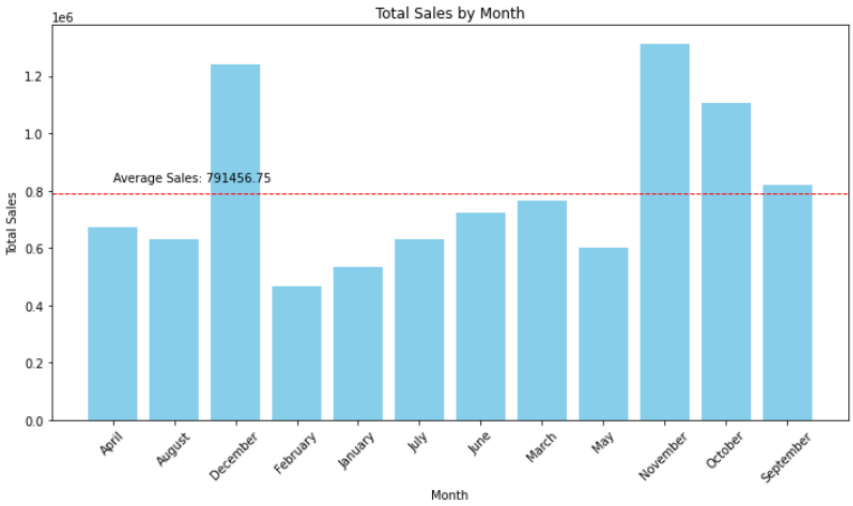

Figure 2 - Total Sales by Month

When visualized the Total Sales by Month in the bar chart, this time, some clear seasonality was observed. The demand for sales increases dramatically in November and December, probably due to holiday shopping and year-end discounts. But this predictable surge provides an opportunity for businesses to create opportunities by marketing to consumers at the right time and stocking that inventory upselling. On the other hand, as the sales fall flat in these two months, it becomes evident that there is a need for mid-year engagement strategy such as loyalty programs or thematic sales to drive sales and level the revenue cycle.

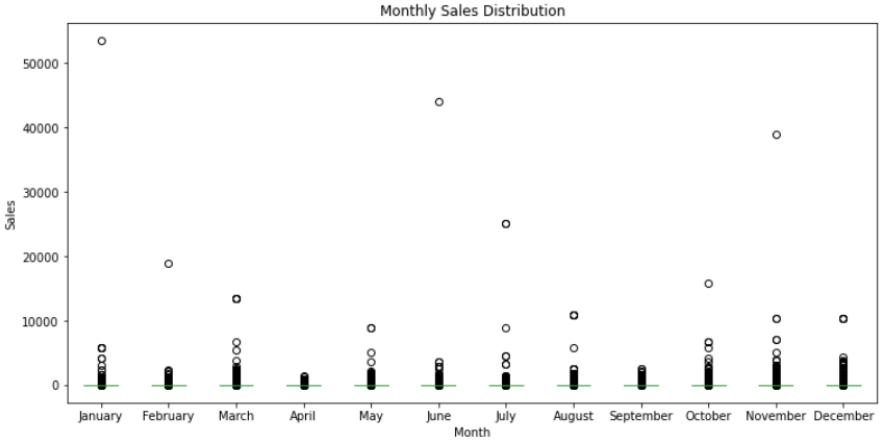

Figure 3 - Monthly Sales Distribution

The Monthly Sales Distribution box plot is a means to delve deeper into monthly variations of behaviour of sales. Some months, particularly January, July and December appear to have large outliers which represent periods of customer activity surges — maybe from New Year promotions, mid-year events or festive shopping. Interquartile ranges are used to determine performance months that are consistent and those that are inconsistent. Retailers need to understand this variability to get their stock levels and demand in line, reduce overstock and unsuccessful sales opportunities.

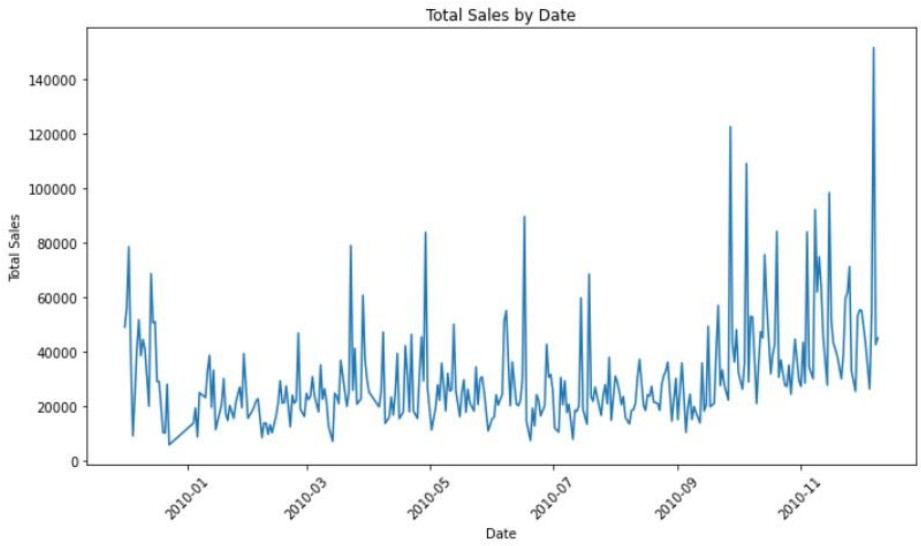

Figure 4 - Timeseries of the sales

The last of these insights comes into play with the Total Sales by Date line graph, illustrating sales fluctuations during the year. In addition, it identifies consistent peaks in March, July, September and a pronounced increase in November contributing to the idea that there are usually seasonal or promotional events involved. These peaks indicate times when consumers are most interested and, thus, when the possibility of excellence could be best time for campaigns and new product launches. In the meantime, months that experience none or sporadic sales might be helped by innovative marketing ways to boost engagement.

Collectively, they provide a holistic story of how consumer behaves, serving as a road map for data driven decision making. With hourly, monthly and seasonal trends business are able to fine tune business operations and enhance customer engagement, ultimately improving revenue performance throughout the year.

3.5. Time-series Analysis

Time series analysis refers to analysing data points picked out or recorded at specified time intervals to recognize patterns, trends and season affects in time. Specifically, for this project, time series analysis refers to analysing how sales performance varies over separate periods in the data of online retail transactions. The project examines business cycle that may result in fluctuations in revenue by analysing sales over days, months and years, in an attempt to find recurring pattern as in seasonal trends and sudden ones. The aggregation includes sales by hour, date, month and year to provide insight into a customer's purchasing behaviour both throughout the day and season. Line plots and bar charts provide visual tools that allow these time-based trends to be understood. Thus, it becomes important to use time series algorithm for forecasting future sales and in general for data driven revenue optimization in e commerce.

3.5.1. ARIMA Model

Auto Regressive Integrated Moving Average (ARIMA) is a commonly used statistical method of time series forecasting. There is a combination of three key parts of Auto Regression (AR) representing the relationship between an observation and its previous values, Integration (I), where differencing was introduced in the data to make it stationary so the trend or seasonality and Moving Average (MA) can be taken away, which makes use of the former forecast errors to enhance the future forecasts. Using ARIMA to forecast future sales on the basis of transaction history. It has been seen that ARIMA helps in analysing sales patterns over time and helping in modelling these trends and thus predicting upcoming sales figures. First, the sales data is made stationary, then the ideal AR, I, and MA components are located through the model. ARIMA produces future sales estimates which a business can use to foresee demand, keep inventories optimal and take proactive marketing decisions. It is valuable for temporal dynamics and because it's able to capture temporal dynamics, it is very useful for e-commerce revenue forecasting.

3.6. Regression Analysis

Regression analysis is a statistical technique to relate a dependent variable with at least one independent variable. In the scope of this project, regression analysis is employed to determine the influence of quantitatively different factors, such as quantity, when purchased and pricing determine total sales of a product in an online retail environment. The project

attempts to which variables influence the revenue by applying regression models on the cleaned and structured dataset. In an example, it can for instance determine whether sales are made at specific times of the day or if substantial price changes affect the behaviour of the customers. This provides an insight of the pricing strategy, timing of promotion, and inventory management. Then regression models can be used to forecast the future sales by estimating how will change of some variables lead to changes in revenue trends. As such, regression is a powerful tool for improving business performance in e-commerce.

3.6.1. Linear Regression

A linear regression is a statistical method used to fit a model to the relationship of a dependent (or dependent) variable with one or several independent variables. In this project, linear regression is used to predict sales based on a number of features such as time things (day of the week, month, hour). In the model case, it is assumed that there exists a linear relationship between the input features and the sales. Training the model on historical data allows the model to learn the coefficients that minimize the difference between actual and predicted sales using Mean Squared Error (MSE), Root Mean Squared Error (RMSE) and similar as metrics.

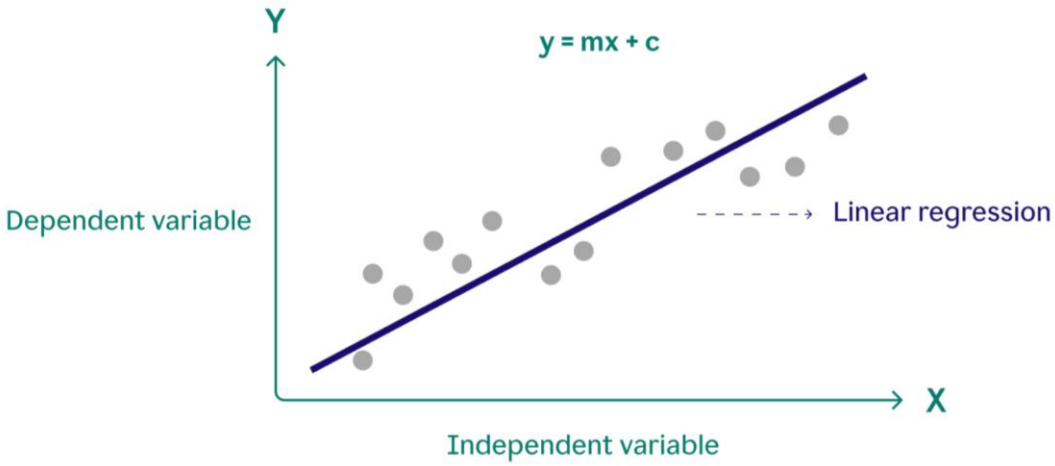

Figure 5 - Linear Regression Interpretation

This technique proves useful in retail, so it's useful for identifying trends and predicting future sales leading to efficient planning of inventory, staffing, and marketing efforts. Linear regression is nice and simple but its performance very much depends on the fact that we make linear assumption, which is often not true in most cases.

3.6.2. Decision Tree Regression

Decision tree regression is a non-linear model for predicting continuous values, through splitting the data into subsets based on feature values. In this project, decision tree regression was applied to predict the sales in various weeks and months based on the features like time of day, day, and month. The model recursively partitions the data in smaller subsets and finds the splits in each subset which have minimum variance inside it to form a tree structure.

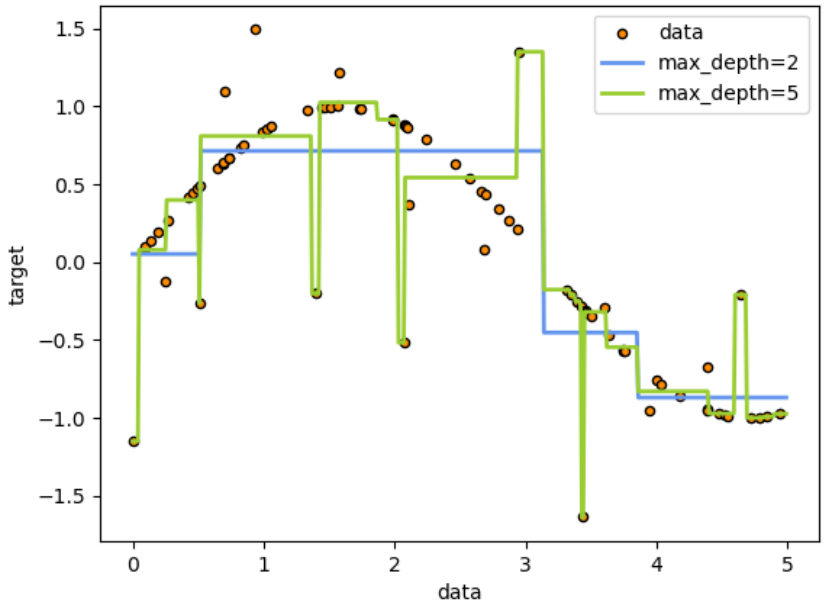

Figure 6 - Impact of Max Depth on Decision Tree Regression Performance

A feature is taken as input and decides at each internal node of the tree. Each leaf node represents a predicted sales value. For overfitting isn't controlled by the depth of the tree and the number of splits (as controlled by max_depth). This technology can be useful in retail for dealing with non-linear relationships and interactions between variables. Because it provides flexibility as well as interpretability, it is well suited for imposing some kind of seasonality or trend on sales that are not necessarily linear.

3.6.7. Random Forest Regression

Ensemble learning method based on random forest regression can grow multiple decision trees and combine them to gain accuracy and prevent from overfitting. For this project, the features used to predict sales include time, day, and month, which are used in order to predict sales using random forest regression. A random forest creates several trees each trained on a different subset of the data and a different subset of features with each split.

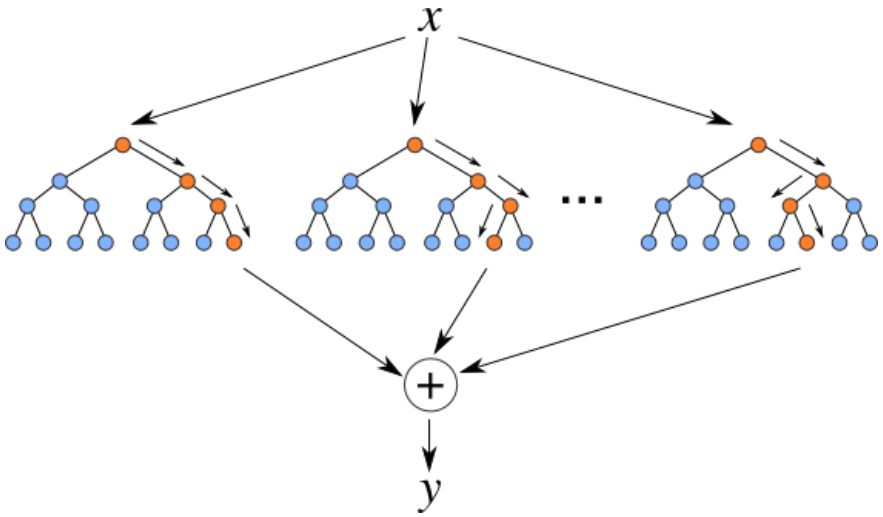

Figure 7 - Random Forest Regression Interpretation

Averaging the predictions of all individual trees offers a way to overcome the bias of any single tree and increases model robustness before the final prediction is made. Most particularly, this method is suitable for handling diverse retail data with the complex surrounding patterns, such as seasonal variations or the changes of consumer behaviour. The random forest is accurate and can withstand over fitting significantly especially with big datasets.

4. Experimental Results

4.1. Time-series Analysis- ARIMA

The chosen AIC value for the selected ARIMA (2,1,3) model was the lowest during the stepwise search, and it was deemed the best fit of the training data. The Augmented Dickey-Fuller test indicates stationarity (p-value ≈ 0.04); thus, the use of differences can be justified. The ACF and PACF plots reveal strong autocorrelation at lag 1 and gradual decay, indicating a potential ARIMA (p,d,q) structure suitable for modelling the time series.

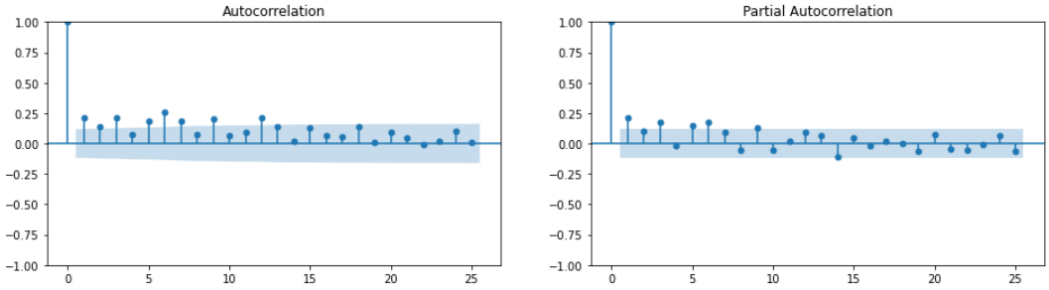

Figure 8 - Autocorrelation and Partial Autocorrelation of ARIMA

The model was fit to produce a 30-day forecast which was compared to the actual test data. With about 750243368 Mean Squared Error (MSE) and around 27390 Root Mean Squared Error (RMSE), the prediction accuracy is moderately accurate. The model does a good job at capturing the underlying trends, but there is some evidence of limitations in small RMSE though perhaps because of noise or other variable(s) that are unaccounted for in the model. Additional improvements might involve trying with seasonal ARIMA or exogenous variable for better accuracy.

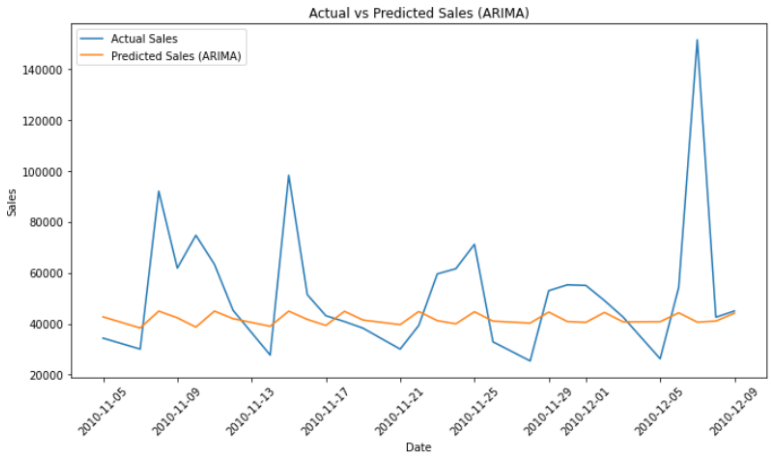

Figure 9 - ARIMA model predictions

ARIMA predicted graph does not match actual sales (AIC) graph very much as it indicates that the overall trends are captured by the model but extreme fluctuations and sudden spikes are overlooked.

4.2. Regression Analysis Results

The preprocessing steps revolved around converting the raw sales data to a structured format that is capable of regression analysis. The 'Date' column was converted into datetime format and a combination of date and time was created in a second 'Datetime' column. Finally, the data was grouped by date, and the daily sales amounts were then calculated. The model was evaluated by keeping the last 30 days apart for testing and training on the rest. Temporal features of the date (day of the week, month, and day) were extracted to capture any time-based patterns of sales behaviour. The input variables of the regression model are these features.

4.2.1. Linear Regression Results

Results of the linear regression model differ greatly between training and testing. The RMSE from training is about 14953 whereas the testing RMSE is approximately 28050. This suggests that while the model fits the training data reasonably well, it overfits the training data. This implies that the simple linear regression model based on basic time-based features (day, month, day of week) cannot capture the complexity of sales pattern.

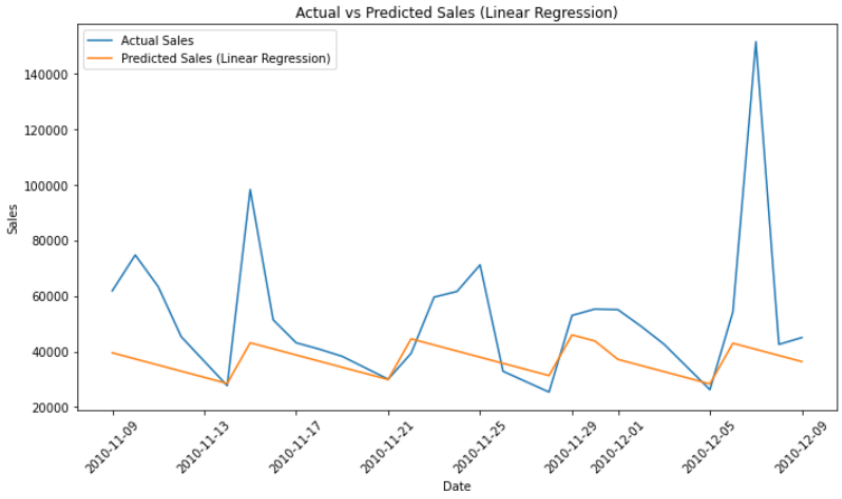

Figure 10 - Baseline model predictions

The predicted sales (Linear Regression) graph shows that the model consistently underestimates actual sales, struggling to capture sharp fluctuations and extreme peaks accurately.

4.2.2. Decision Tree Regression Results

The cross validation was used for selecting the linear regression model with a max depth of 5, which had a better performance than that of decision tree model. About 11093 RMSE in the training suggests a good. In the case of the RMSE on the test data, it narrowed down to around 26177 which is a better generalization than it previous models. The training and testing errors still differ a lot, but the gap between these two is smaller, indicating a better balance between bias and variance.

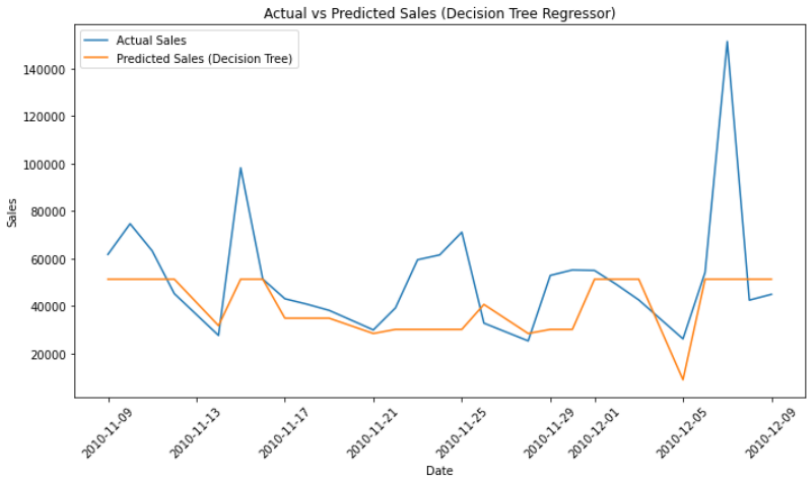

Figure 11 - Decision Tree Regressor predictions

The Decision Tree Regression prediction matches the trend of the actual sales, but lacks capturing the peaks and troughs very well as the prediction hangs around the general trend of actual sales.

4.2.3. Random Forest Regression Results

A good fit for the Random Forest model was obtained with the optimal 200 estimators, with a low RMSE around 5,608 on the training data and good performance. The slight overfitting is detected from the test RMSE of about 24,915, which is higher than the RMSE of training data. However, the model generalizes better than previous approaches as the ensemble seems to have the ability to reduce variance. This shows Random Forest's robustness because test error is relatively smaller compared to that of linear regression and decision tree models.

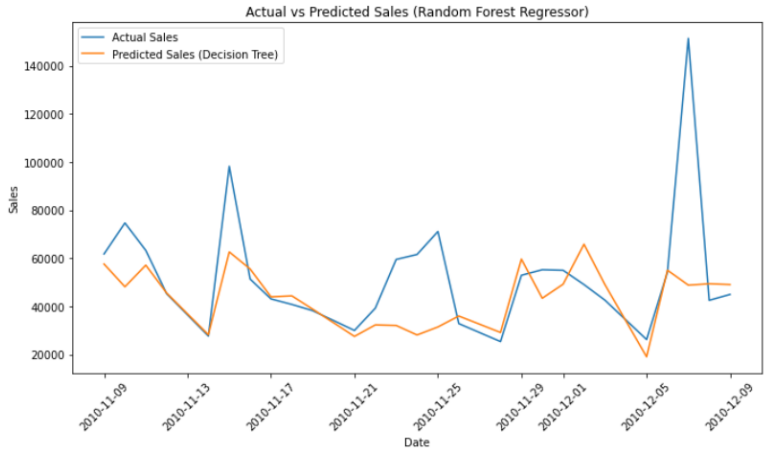

Figure 12 - Random Forest Regressor predictions

The Random Forest Regression prediction actually mirrors sales trends fairly closely and does a better job anticipating the swings of the underlying data than other models, but it is not good at predicting extreme peaks and this indicates the model is capacity limited in absorbing sudden spikes in demand.

5. Discussion

5.1. Comparison with Existing Literature

The results of this project are in line with the literature as it uses machine learning models for prediction of consumer behaviour and revenue optimization. This project, like Peterson et al. (2025) and Gupta (2026) focuses on structured data preprocessing and feature extraction in order to improve predictive accuracy. However, the largest benefit of this project lies in showcasing the possibility of ensemble models such as Random Forest for reducing RMSE of forecasts and improving forecast performance. Although we look at AI and ML for dynamic pricing, the most important benefit is the ability to significantly reduce RMSE. Additionally, just like Krishnamoorthy et al. (2023), the project also encourages that in most cases, the more complex models can outperform the simpler such as linear regression, but especially when it includes non-linear trends. On the other hand, unlike the literature that investigates such advanced pricing strategies or real-time analytics, this study proposes a foundational but practical view from the time series and regression-based forecasting. Overall, this complements existing research in filling the gap in the traits of predictive modelling techniques in retail analytics.

5.2. Limitations of the Results

Several limitations are seen in the project and while it shows the effectiveness of various regression and time series models for sales forecasting, none of them perform well. On the first point, because the models rely on simple temporal characteristics such as day, month, day of the week, which may not capture the pattern resulting from promotions, holidays or external economic factors. In particular, the linear regression model failed to model a non-linear trend because of which it underperformed and hence it heightened the need for richer feature engineering. Random Forest model has a lower training error but due to higher error on test data it is prone to overfitting, as visible here. On top of that, the ARIMA model was able to capture the generality of the trends in the data, but could not predict sudden spikes in such volatile environments. The second constraint is the size and scope of the dataset; using finer granule or seasonal data may help in making model robust. In the last, no external variables or data streams from real life were introduced to limit the scalability to the real world.

5.3. Relation to Project Objectives

This project effectively aligns with what the project's objectives are through usage of historical sales data to find interesting patterns with the way in which customers purchase. Temporal trends, i.e. day, month, weekday effects, were identified and extracted via preprocessing and used as regressors in different regression models. Linear regression, decision tree and random forest models were implemented for the forecasting of future sales, which give an insight into the revenue trends over the time. Finally, the two approaches were model evaluated using RMSE and MSE to provide an objective assessment of prediction accuracy as well as the merits and shortcomings of each. By virtue, adding trend and seasonality in ARIMA, they used ARIMA for time series analysis and got more depth. Together, the findings suggest potential data driven strategies by showing the models that best propagate customer behaviour and sales dynamics. The insights have significant impact to formulate improved forecasting techniques that, in turn can lead to better inventory management, pricing decisions and overall revenue optimization on online retailing.

6. Conclusion

6.1. Validated Conclusions

By means of historical transaction data, the outcomes of this experiment confirm the efficacy of regression models as efficient instruments for future sales prediction. With a somewhat low Root Mean Squared Error (RMSE) on the test set relative to Linear Regression and Decision Tree models, the Random Forest Regressor showed the best predictive performance among the evaluated models. The Random Forest model was able to generalise better because to its ensemble learning approach, which lowers overfitting and improves stability; the Linear Regression model underperformed mostly due to its incapacity to capture complicated patterns. This validates that challenges involving retail sales prediction call for non-linear and ensemble-based models better suited. Regarding the second research question, the investigation found that temporal elements such the day of the week, month, and particular days significantly affect income fluctuations. These characteristics generated from consistently utilised across all models from the transaction timestamps. Their influence on sales variation was clear from the better accuracy noted when these characteristics were included. Furthermore, ARIMA time-series study confirmed the significance of autocorrelation and trends in the data by implying that previous sales success significantly affects future results.

The models effectively caught the overall sales patterns despite certain limits in forecasting severe variations or abrupt spikes in sales. This supports the conclusion that, for short-term planning and inventory control, regression models are useful even if they may not completely explain all erratic fluctuations. All things considered; this experiment validates that future sales patterns may be consistently predicted using regression-based methods—especially Random Forest model. Furthermore, it points out temporal trends as important forecasters of income variations, therefore providing insightful information for strategic planning in online retail contexts.

6.2. Future Works

Future research in this study might concentrate on honing the model even more by investigating additional hyperparameters in the Random Forest, including max_depth,

min_samples_split, and max_features, thus enhancing generalisation. Including new information, including external elements influencing sales, could also improve prediction accuracy. Investigating more sophisticated ensemble techniques as XGBoost or Gradient Boosting might also be advantageous. Moreover, methods like cross-valuation and feature engineering might be used to stop overfitting and raise model resilience. At last, adding seasonality elements or testing the models on more extensive datasets could improve the general performance.

References

Akter, R. et al. (2023) 'Optimizing online sales strategies in the USA using machine learning: Insights from consumer behavior', Journal of Business and Management Studies, 5(4), pp. 167–183. doi:10.32996/jbms.2023.5.4.17.

Farheen, Z. and Dharani, A. (2024) 'Prediction of customer purchasing patterns for retail optimization using market basket techniques', 2024 8th International Conference on Computational System and Information Technology for Sustainable Solutions (CSITSS), pp. 1–5. doi:10.1109/csitss64042.2024.10816740.

Sagar, A. et al. (2024) 'Analyzing e-commerce dynamics: Customer Satisfaction, revenue prediction, and sentiment analysis in retail', Lecture Notes in Networks and Systems, pp. 145–160. doi:10.1007/978-3-031-73125-9_9.

Kumar, S., Margala, M. & Shankar, S.S. (2023) 'A novel weight-optimized LSTM for dynamic pricing solutions in e-commerce platforms based on customer buying behaviour', Soft Computing. Available at: https://doi.org/10.1007/s00500-023-08729-1.

Gupta, M. (2025) Dynamic pricing optimization in e-commerce using AI and machine learning: A comprehensive framework for demand prediction and revenue maximization. doi:10.2139/ssrn.5075196.

Koppolu, H.K.R. (2025) 'AI-Powered Revenue Management and Monetization: A Data Engineering Framework for Scalable Billing Systems in the Digital Economy', MSW Management Journal, 34(2), pp. 776–787. doi:10.7492/9s3p2a55.

Krishnamoorthy, R. et al. (2023) 'Multi objective evaluator model development for analyse the customer behaviour', 2023 3rd International Conference on Advancement in Electronics Communication Engineering (AECE), pp. 640–645. doi:10.1109/aece59614.2023.10428189.

Ortakci, Y. and Seker, H. (2024) 'Optimising customer retention: An ai-driven personalised pricing approach', Computers & Industrial Engineering, 188, p. 109920. doi:10.1016/j.cie.2024.109920.

Rekha, K.S. et al. (2024) 'Predictive analysis of consumer behaviour using supervised learning techniques', 2024 Second International Conference Computational and Characterization

Techniques in Engineering Sciences (IC3TES), pp. 1–6. doi:10.1109/ic3tes62412.2024.10877475.

Effendi, S.Y., 2023. ARIMA for Time-Series Forecasting: Retail Sales Predictions. Medium. Available at: https://medium.com/@syarifyusuf/arima-for-time-series-forecasting-retail-sales-predictions-6e0da74232a0

Abbaschian, B., 2023. Dynamic Pricing Using Machine Learning. Medium. Available at: https://medium.com/@baabak/dynamic-pricing-using-machine-learning-5e882282effe

Johannes, R. & Alamsyah, A., 2021. Sales prediction model using classification decision tree approach for small medium enterprise based on Indonesian e-commerce data. Available at: https://doi.org/10.48550/arXiv.2103.03117.

Zhang, Y., Wu, X., Gu, C. & Xie, Y., 2019. Predict future sales using ensembled random forests. arXiv. Available at: https://doi.org/10.48550/arXiv.1904.09031.

Li, M., Ji, S. & Liu, G., 2018. Forecasting of Chinese e-commerce sales: An empirical comparison of ARIMA, nonlinear autoregressive neural network, and a combined ARIMA-NARNN model. Mathematical Problems in Engineering. Available at: https://doi.org/10.1155/2018/8704537.

Reddy, N.K., 2023. Forecasting e-commerce trends: Utilizing linear regression, polynomial regression, random forest, and gradient boosting for accurate sales and demand prediction. International Journal of HRM and Organizational Behavior, 11(3), pp.11–26. Available at: https://ijhrmob.org/index.php/ijhrmob/article/view/115.

Istalingam, H., 2021. Time series forecasting of sales using ARIMA model. Medium. Available at: https://medium.com/@hemchdr/time-series-forecasting-of-sales-using-arima-model-d9e8e67a9a83.

Udell, A., 2020. Predicting e-commerce sales with a random forest regression. Towards Data Science. Available at: https://medium.com/towards-data-science/predicting-e-commerce-sales-with-a-random-forest-regression-3f3c8783e49b.

Takale, S., Bhong, T., Dethe, U. & Gandhi, P., 2022. Sales prediction using linear regression. Journal of Electronics, Computer Networking and Applied Mathematics, 2(05), pp.62–71. Available at: https://doi.org/10.55529/jecnam.25.62.71.

Haines, L., 2023. E-commerce sales forecasting: A business case for linear regression. Synaptiq.ai. Available at: https://www.synaptiq.ai/library/e-commerce-sales-forecasting-a-business-case-for-linear-regression.

Ngui, S.W., 2023. Python tutorial: Using random forest for sales forecasting. Medium. Available at: https://swngui.medium.com/python-tutorial-using-random-forest-for-sales-forecasting-bda61d79318e.

Ganguly, P. & Mukherjee, I., 2024. Enhancing Retail Sales Forecasting with Optimized Machine Learning Models. arXiv. Available at: https://arxiv.org/abs/2410.13773

Kalifa, D., Singer, U., Guy, I., Rosin, G.D. & Radinsky, K., 2024. Leveraging World Events to Predict E-Commerce Consumer Demand under Anomaly. arXiv. Available at: https://arxiv.org/abs/2405.13995

Kirichenko, L., Radivilova, T. & Zinkevich, I., 2019. Forecasting Weakly Correlated Time Series in Tasks of Electronic Commerce. arXiv. Available at: https://arxiv.org/abs/1904.10927

Hu, J., Zhuang, Y. & Zhao, S., 2023. A Sequential Learning Procedure with Applications to Online Sales Examination. arXiv. Available at: https://arxiv.org/abs/2311.02273

Appendices

```
df = pd.read_excel('online_retail_II.xlsx')



# In[3]:



df.head()



# In[4]:



print('Total data points:', df.shape[0])



# ## Data Cleaning

# In[5]:
```

```
print(df.isnull().sum())
```

```
# In[6]:
```

```
df.drop(columns=["Description", "Customer ID"], inplace=True)
```

```
# In[7]:
```

```
df.head()
```

```
# In[8]:
```

```
df["Date"] = pd.to_datetime(df["InvoiceDate"]).dt.date

df["Time"] = pd.to_datetime(df["InvoiceDate"]).dt.time
```

```
# In[9]:
```

```
df.drop(columns=["InvoiceDate"], inplace=True)
```

```
# In[10]:
```

```
df.head()
```

```
# ## Feature Engineering
```

```
# In[11]:
```

```
import seaborn as sns
import matplotlib.pyplot as plt
```

```
# In[12]:
```

```
plt.figure(figsize=(15, 12))
sns.countplot(data=df, x='Country', order=df['Country'].value_counts().index)
plt.xticks(rotation=90)
plt.xlabel('Country')
plt.ylabel('Number of Sales')
plt.title('Sales Count by Country')
plt.show()
```

```
# In[13]:
```

```
df['Country'].value_counts()
```

```
# ### Keeping only the data that belongs to the United Kingdom as it has over 95% of the total data
```

```
# In[14]:
```

```
df = df[df['Country'] == 'United Kingdom']
```

```
# In[15]:


print('Total data points:', df.shape[0])


# In[16]:


df.drop(columns=["Country"], inplace=True)


# In[17]:


df.head()


# In[18]:


data = df.copy()
```

In[19]:

```
data.head()
```

In[20]:

```
data['Hour'] = data['Time'].apply(lambda x: x.hour)
```

In[21]:

```
data['Sales'] = data['Quantity'] * data['Price']
```

In[22]:

```
data['Sales'] = data['Sales'].abs()
```

In[23]:

```
data.head()
```

In[24]:

```
data['Date'] = pd.to_datetime(data['Date'], errors='coerce')
```

In[25]:

```
data['year'] = data['Date'].dt.year
data['quarter'] = data['Date'].dt.quarter
data['month'] = data['Date'].dt.month
data['day'] = data['Date'].dt.day
```

In[26]:

```
data.head()
```

In[27]:

```
import calendar
data['month'] = data['month'].apply(lambda x: calendar.month_name[x])
```

In[28]:

```
data.head()
```

In[29]:

```
data.drop(columns=["Invoice", "StockCode"], inplace=True)
```

In[30]:

data.head()

Data Visualization and Analysis

In[31]:

sales_per_hour = data.groupby('Hour')['Sales'].sum()

In[32]:

plt.figure(figsize=(12, 7))

plt.plot(sales_per_hour.index, sales_per_hour.values, linestyle='-')

plt.xlabel('Hour of the Day')

plt.ylabel('Total Sales')

```
plt.title('Total Sales by Hour')

plt.xticks(range(0, 24))

plt.show()




# In[33]:



sales_by_year = data.groupby('year')['Sales'].sum().reset_index()



# In[34]:



plt.figure(figsize=(10, 6))

plt.bar(sales_by_year['year'].astype(str), sales_by_year['Sales'], color='skyblue')

plt.xlabel('Year')

plt.ylabel('Total Sales')

plt.title('Total Sales by Year')

plt.xticks(rotation=45)

plt.tight_layout()

plt.show()
```

In[35]:

```
sales_by_month = data.groupby('month')['Sales'].sum().reset_index()
average_sales = sales_by_month['Sales'].mean()
```

In[36]:

```
plt.figure(figsize=(10, 6))
plt.bar(sales_by_month['month'].astype(str), sales_by_month['Sales'], color='skyblue')
plt.xlabel('Month')
plt.ylabel('Total Sales')
plt.title('Total Sales by Month')
plt.xticks(rotation=45)
plt.axhline(y=average_sales, color='red', linestyle='--', linewidth=1)
plt.text(x=0, y=average_sales + 0.05 * average_sales, s=f'Average Sales: {average_sales:.2f}',
color='black')
plt.tight_layout()
plt.show()
```

In[37]:

```
fig, ax = plt.subplots(figsize=(12, 6))
data.boxplot(column='Sales', by='month', grid=False, ax=ax)
plt.xticks(ticks=range(1, 13), labels=[calendar.month_name[i] for i in range(1, 13)])
plt.title('Monthly Sales Distribution')
plt.suptitle('')
plt.xlabel('Month')
plt.ylabel('Sales')
plt.show()
```

In[38]:

```
daily_sales = data.groupby('Date')['Sales'].sum()
```

In[39]:

```
plt.figure(figsize=(10, 6))
```

```
plt.plot(daily_sales.index, daily_sales.values)

plt.xlabel('Date')

plt.ylabel('Total Sales')

plt.title('Total Sales by Date')

plt.xticks(rotation=45)

plt.tight_layout()

plt.show()



# ## Time-series Analysis


# In[40]:



dx = data.groupby("Date")["Sales"].sum().reset_index()



# In[41]:



dx.head(20)
```

ARIMA

In[94]:

```
dx['Date'] = pd.to_datetime(dx['Date'])
dx = dx.sort_values('Date')
```

In[95]:

```
import statsmodels.api as sm
from statsmodels.tsa.arima.model import ARIMA
from sklearn.metrics import mean_squared_error
from statsmodels.tsa.stattools import adfuller
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from pmdarima import auto_arima
```

In[96]:

```
train = dx.iloc[:-30]

test = dx.iloc[-30:]



# In[97]:



result = adfuller(train['Sales'])

print(f'ADF Statistic: {result[0]}')

print(f'p-value: {result[1]}')



# In[98]:



fig, axes = plt.subplots(1, 2, figsize=(16, 4))

plot_acf(train['Sales'], ax=axes[0])

plot_pacf(train['Sales'], ax=axes[1])

plt.show()



# In[99]:
```

```
auto_model = auto_arima(train['Sales'], seasonal=False, trace=True)
```

```
# In[100]:
```

```
model = ARIMA(train['Sales'], order=(2,1,3))
model_fit = model.fit()
```

```
# In[101]:
```

```
forecast = model_fit.forecast(steps=30)
forecast_index = test.index
```

```
# In[102]:
```

```
mse = mean_squared_error(test['Sales'], forecast)
print(f'Mean Squared Error: {mse}')
```

In[103]:

```
rmse = np.sqrt(mean_squared_error(test['Sales'], forecast))
print(f'Root Mean Squared Error: {rmse}')
```

In[105]:

```
plt.figure(figsize=(10, 6))
plt.plot(test['Date'], test['Sales'], label='Actual Sales')
plt.plot(test['Date'], forecast, label='Predicted Sales (ARIMA)')
plt.title('Actual vs Predicted Sales (ARIMA)')
plt.xlabel('Date')
plt.ylabel('Sales')
plt.legend()
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

Regression Analysis

In[53]:

```
data.head()
```

In[54]:

```
df = data.copy()
```

In[55]:

```
df.head()
```

In[56]:

```
df['Date'] = pd.to_datetime(df['Date'])
```

```
# In[58]:
```

```
df['Datetime'] = df.apply(lambda row: pd.Timestamp.combine(row['Date'], row['Time']), axis=1)
```

```
# In[60]:
```

```
daily_sales = df.groupby('Date').agg({'Sales': 'sum'}).reset_index()
```

```
# In[61]:
```

```
max_date = daily_sales['Date'].max()
train_data = daily_sales[daily_sales['Date'] < (max_date - pd.Timedelta(days=30))]
test_data = daily_sales[daily_sales['Date'] >= (max_date - pd.Timedelta(days=30))]
```

In[62]:

```
train_data['dayofweek'] = train_data['Date'].dt.dayofweek
train_data['month'] = train_data['Date'].dt.month
train_data['day'] = train_data['Date'].dt.day
```

In[69]:

```
test_data['dayofweek'] = test_data['Date'].dt.dayofweek
test_data['month'] = test_data['Date'].dt.month
test_data['day'] = test_data['Date'].dt.day
```

In[70]:

```
train_data.head()
```

In[71]:

```
x_train = train_data[['dayofweek', 'month', 'day']]
y_train = train_data['Sales']

x_test = test_data[['dayofweek', 'month', 'day']]
y_test = test_data['Sales']
```

Linear Regressor

In[81]:

```
from sklearn.linear_model import LinearRegression
```

In[83]:

```
lr = LinearRegression()
lr.fit(x_train, y_train)
```

```
y_pred = lr.predict(x_train)

mse = mean_squared_error(y_train, y_pred)

rmse = np.sqrt(mse)


print(f'Train Mean Squared Error: {mse}')

print(f'Train Root Mean Squared Error: {rmse}')




# In[84]:




lr = LinearRegression()

lr.fit(x_train, y_train)

y_pred = lr.predict(x_test)

mse = mean_squared_error(y_test, y_pred)

rmse = np.sqrt(mse)


print(f'Test Mean Squared Error: {mse}')

print(f'Test Root Mean Squared Error: {rmse}')




# In[86]:
```

```
plt.figure(figsize=(10, 6))
plt.plot(test_data['Date'], y_test, label='Actual Sales')
plt.plot(test_data['Date'], y_pred, label='Predicted Sales (Linear Regression)')
plt.title('Actual vs Predicted Sales (Linear Regression)')
plt.xlabel('Date')
plt.ylabel('Sales')
plt.legend()
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

```
# ### Decision Tree Regressor
```

```
# In[73]:
```

```
from sklearn.tree import DecisionTreeRegressor
from sklearn.model_selection import GridSearchCV
```

```
# In[76]:
```

```
param_grid = {'max_depth': [5, 10, 15, 20, 25, 30, 50]}

dt = DecisionTreeRegressor(random_state=42)

grid_search_dt = GridSearchCV(estimator=dt, param_grid=param_grid, cv=5,
                              scoring='neg_mean_squared_error', n_jobs=-1,
                              verbose=1)

grid_search_dt.fit(x_train, y_train)
best_dt = grid_search_dt.best_estimator_
y_pred = best_dt.predict(x_train)
mse = mean_squared_error(y_train, y_pred)
rmse = np.sqrt(mean_squared_error(y_train, y_pred))

print("Best max_depth:", grid_search_dt.best_params_)
print(f'Train Root Mean Squared Error: {rmse}')
print(f'Train Mean Squared Error: {mse}')


# In[89]:
```

```
dt = DecisionTreeRegressor(max_depth=5, random_state=42)

dt.fit(x_train, y_train)

y_pred = dt.predict(x_test)

mse = mean_squared_error(y_test, y_pred)

rmse = np.sqrt(mean_squared_error(y_test, y_pred))


print(f'Test Root Mean Squared Error: {rmse}')

print(f'Test Mean Squared Error: {mse}')




# In[90]:



plt.figure(figsize=(10, 6))

plt.plot(test_data['Date'], y_test, label='Actual Sales')

plt.plot(test_data['Date'], y_pred, label='Predicted Sales (Decision Tree)')

plt.title('Actual vs Predicted Sales (Decision Tree Regressor)')

plt.xlabel('Date')

plt.ylabel('Sales')

plt.legend()

plt.xticks(rotation=45)

plt.tight_layout()
```

```
plt.show()
```

```
# ### Random Forest Regressor
```

```
# In[87]:
```

```
from sklearn.ensemble import RandomForestRegressor
```

```
# In[91]:
```

```
param_grid = {'n_estimators': [10, 50, 100, 150, 200]}

rf = RandomForestRegressor(random_state=42)

grid_search_rf = GridSearchCV(estimator=rf, param_grid=param_grid, cv=5,
                              scoring='neg_mean_squared_error', n_jobs=-1,
                              verbose=1)

grid_search_rf.fit(x_train, y_train)
```

```
best_rf = grid_search_rf.best_estimator_

y_pred = best_rf.predict(x_train)

mse = mean_squared_error(y_train, y_pred)

rmse = np.sqrt(mean_squared_error(y_train, y_pred))


print("Best max_depth:", grid_search_rf.best_params_)

print(f'Train Root Mean Squared Error: {rmse}')

print(f'Train Mean Squared Error: {mse}')



# In[92]:



rf = RandomForestRegressor(n_estimators=200, random_state=42)

rf.fit(x_train, y_train)

y_pred = rf.predict(x_test)

mse = mean_squared_error(y_test, y_pred)

rmse = np.sqrt(mean_squared_error(y_test, y_pred))


print(f'Test Root Mean Squared Error: {rmse}')

print(f'Test Mean Squared Error: {mse}')
```

In[93]:

```
plt.figure(figsize=(10, 6))
plt.plot(test_data['Date'], y_test, label='Actual Sales')
plt.plot(test_data['Date'], y_pred, label='Predicted Sales (Decision Tree)')
plt.title('Actual vs Predicted Sales (Random Forest Regressor)')
plt.xlabel('Date')
plt.ylabel('Sales')
plt.legend()
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```