Optimizing Inventory and Sales Forecasting in Retail through Predictive Analytics

Submitted by:

Santanu Das (MA24M021)

Ayushi Yaduvanshi (CE23S200)

Iswarya C (NS25Z071)

Kasipeta Pavani (NS25Z016)

Afra Jakir Rehman (BT24D403)

Rajavivegan M R (NA22B085)

Course: **MA5755 (Data Analysis and Visualization)**

Department of Mathematics

INDIAN INSTITUTE OF TECHNOLOGY MADRAS

Date: 21/04/2025

Confidentiality Statement

This project is based on a proprietary dataset provided by **Café Coffee Day (IITM branch)**, shared exclusively for academic and research purposes under strict confidentiality. The dataset was obtained after multiple requests and with the explicit condition that it must not be published, distributed, or disclosed in any form outside the scope of this course project.

All analyses, visualizations, and findings presented in this report are derived solely for internal evaluation under the **MA5755 (Data Analysis and Visualization)** course. We have taken necessary measures to ensure that no sensitive business information or customer data is exposed.2

We sincerely thank **Café Coffee Day and MR. HARISH** for their support and trust in allowing us access to this dataset. We remain committed to upholding the confidentiality agreement and ensuring responsible data handling throughout the project lifecycle.

Sales Data EDA Report – Cafe Coffee Day

1. Introduction

This report presents an Exploratory Data Analysis (EDA) of sales data from IITM Cafe Coffee Day (CCD) store. The aim is to extract key patterns, detect seasonality, and identify best-performing items using data collected for the years 2023 and 2024.

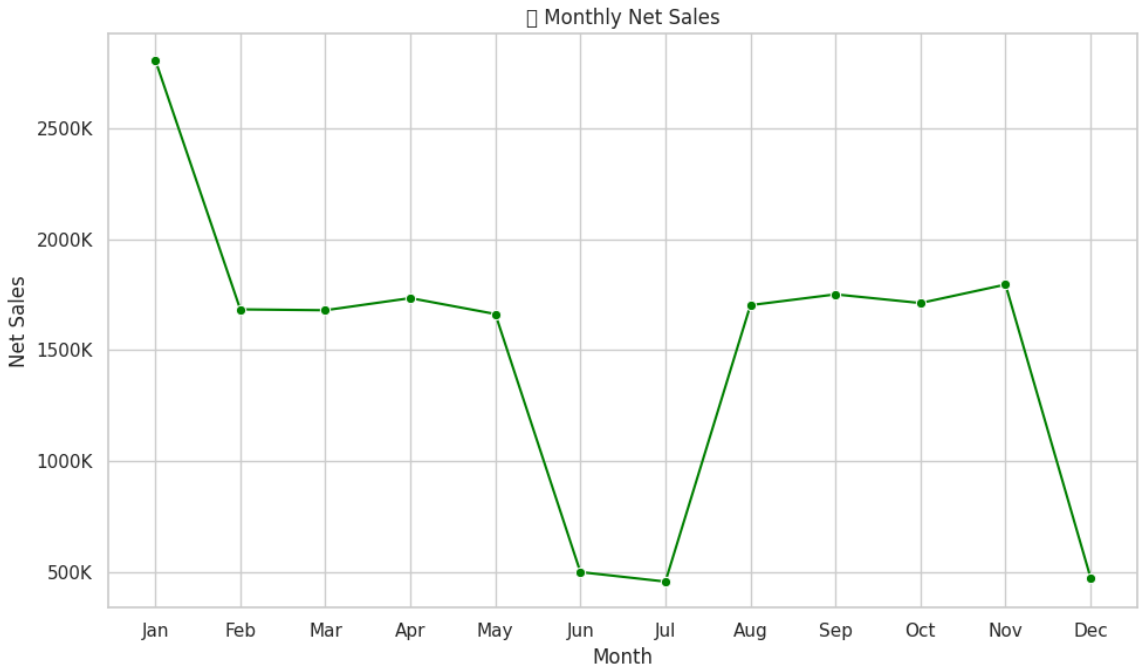
2. Dataset Overview

The dataset contains 12428 records with 11 attributes. Each entry represents a sale transaction including details such as 'rate', 'quantity', 'discount', 'item_name', 'sales_amount', 'net_sales', ‘month’ 'date', 'discount_per_unit', ‘year’, 'id’

3. Key Insights and Visualizations

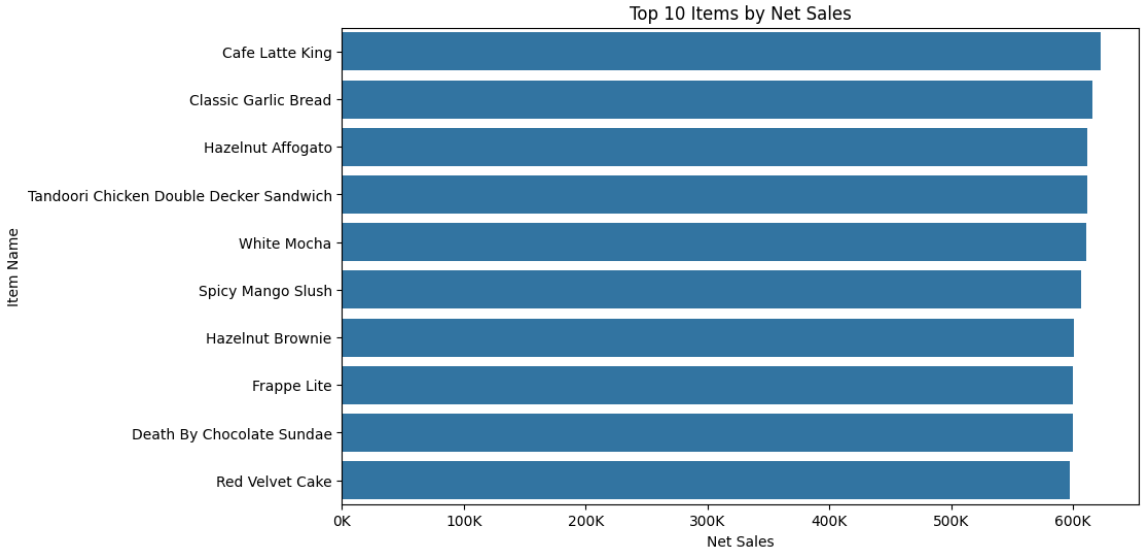
3.1 Monthly Net Sales Trend

The chart below compares monthly net sales. We observe significant fluctuations indicating seasonal demand patterns. Notably, sales dipped in certain months likely due to external events like floods or holidays.



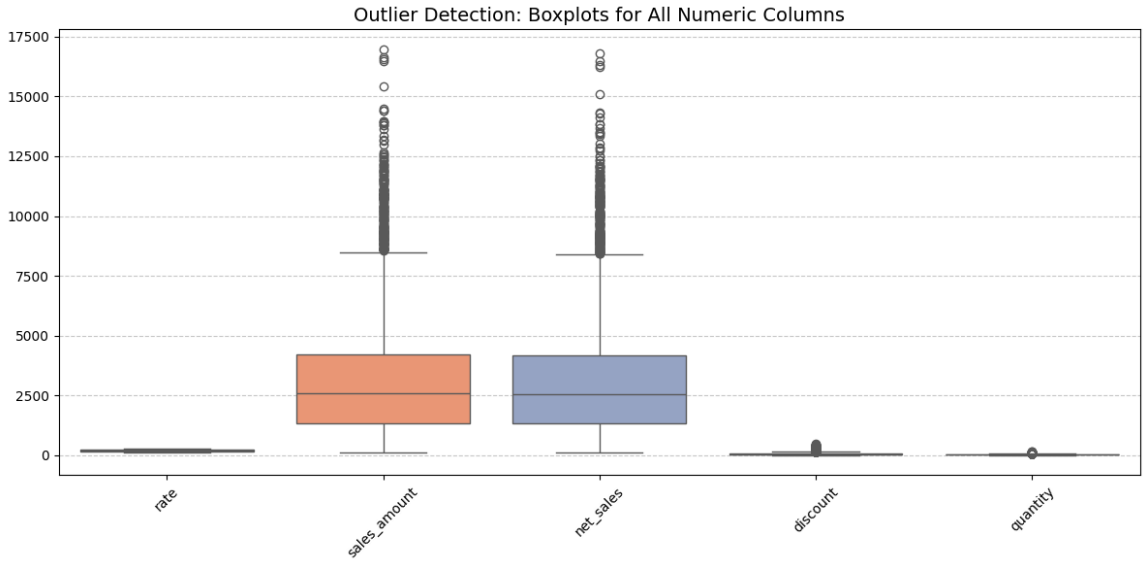
3.2 Top 10 Best-Selling Items

This chart highlights the ten most popular items based on net sales. It helps CCD understand which products are driving the most revenue and customer preference.

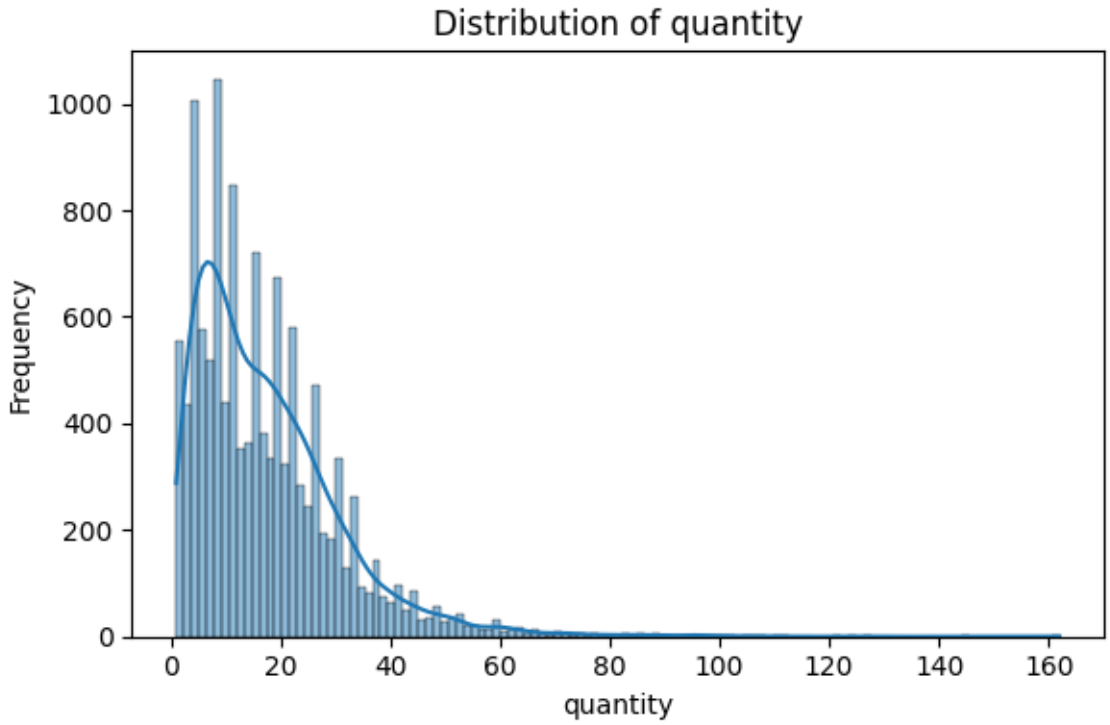


3.3 Outlier Detection

Boxplots are used to identify outliers in key numeric features such as rate, quantity, sales amount, and net sales. Outliers may indicate promotional spikes, data entry issues, or unusual customer behavior.

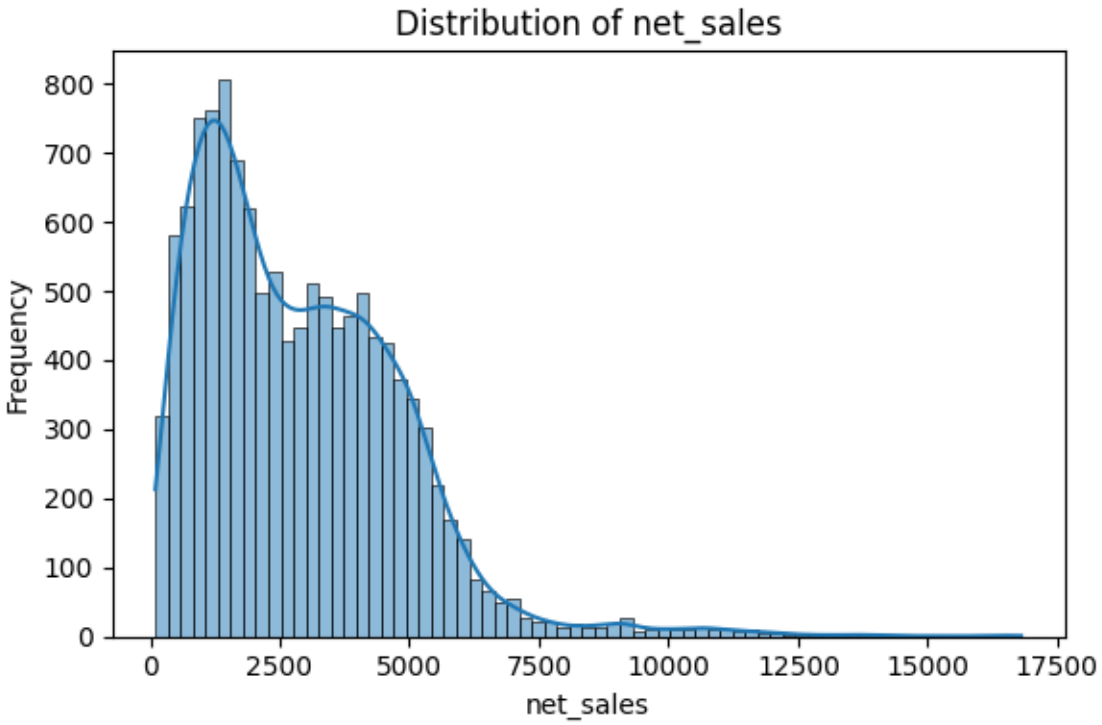


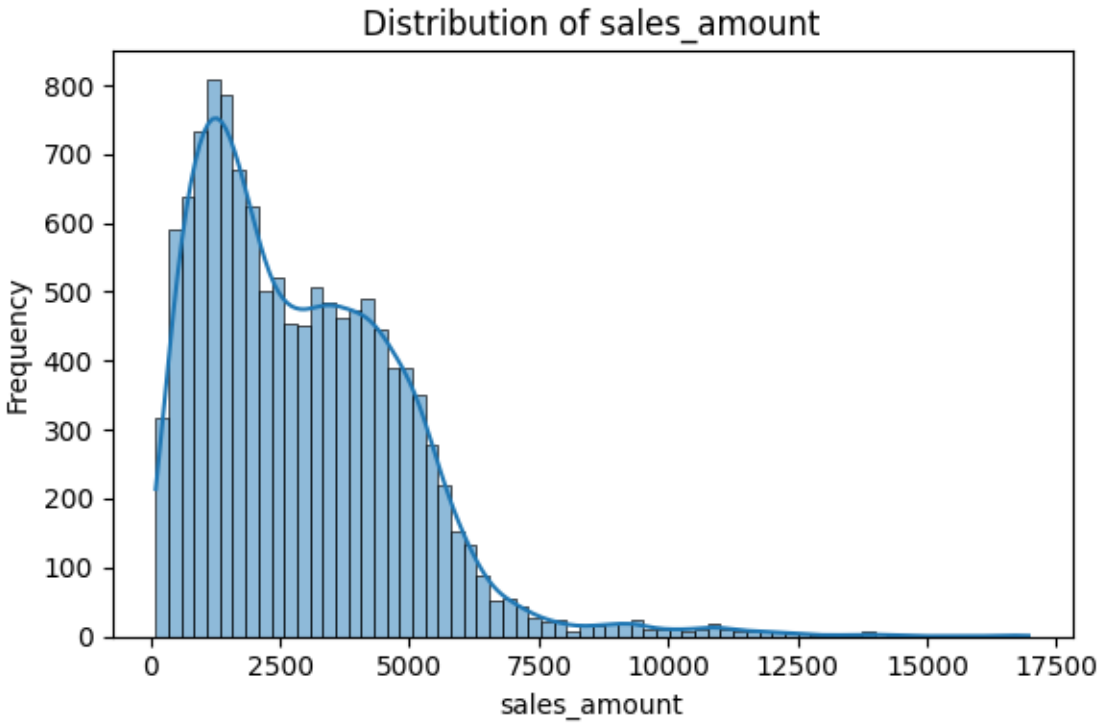
3.4 Distributions of some column and inference:



The quantity distribution is **right-skewed**, with most transactions falling within the **1–20-unit** range. A sharp peak around **5–10 units** suggest this is the most common purchase size. Higher quantity orders (>50 units) are rare but present, indicating occasional bulk purchases. This pattern reflects a preference for **low-volume, frequent buying**, with infrequent spikes from large-scale orders.

Net sales and Sales amount:



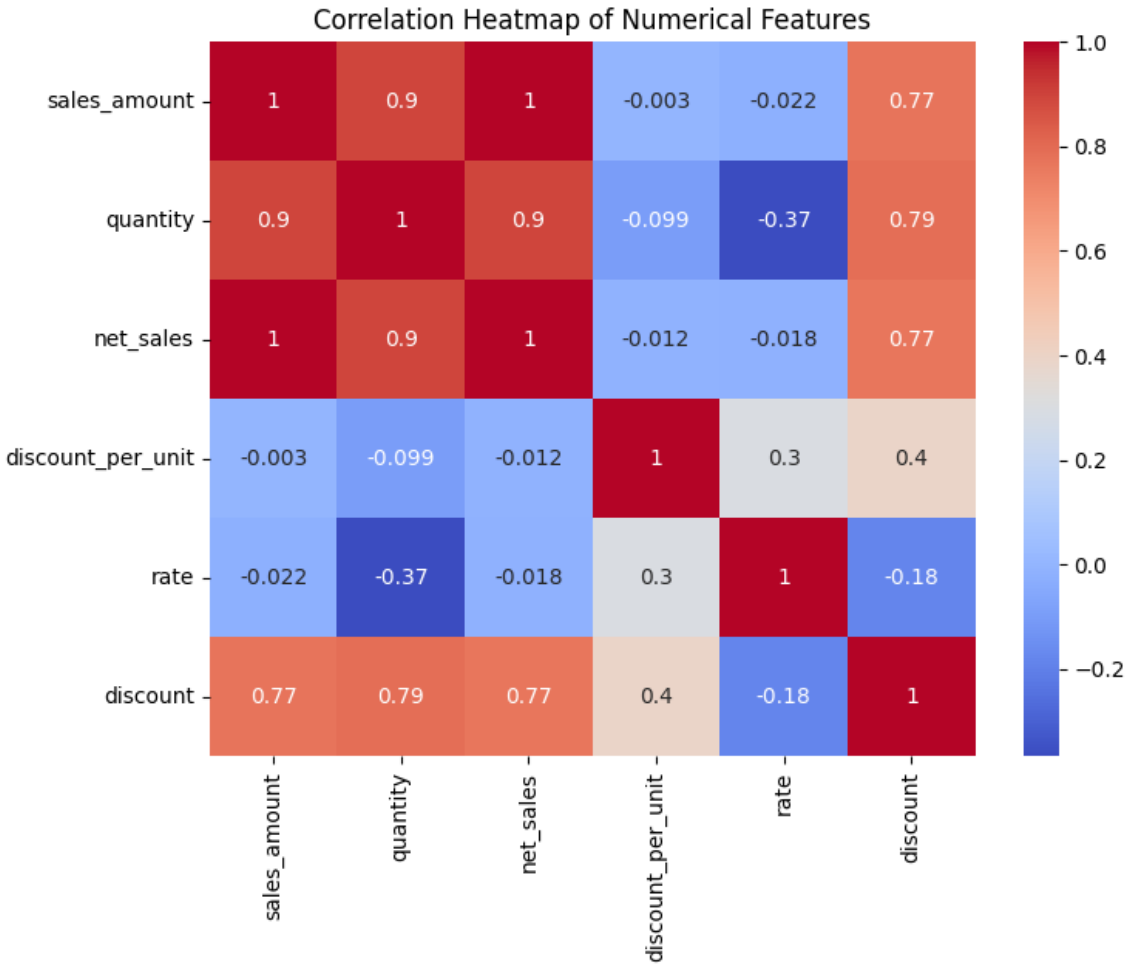


Both net_sales and sales_amount exhibit **right-skewed distributions**, with most transactions concentrated in the **₹0–4000** range, indicating that most purchases fall into the **moderate-value** category. A noticeable **secondary peak around ₹4000–5000** hints at a subset of **higher-value transactions**, possibly due to bulk orders or premium product offerings.

The **long tail** extending beyond ₹10,000 shows that although rare, **high-value purchases exist and likely contribute significantly to total revenue**. This pattern highlights a business model driven by frequent mid-range sales, with occasional large transactions that may skew average revenue metrics.

3.5 Correlation Heatmap

The correlation heatmap provides insights into the linear relationships between numerical features in the dataset. Features with high correlation can be important for predictive modeling and multicollinearity detection.



Strong Positive Correlations:

- **sales_amount ↔ net_sales**: **1.00**
  Practically perfect correlation — net sales scale directly with the total sales amount.

- **sales_amount ↔ quantity**: **0.90**
  Indicates that higher quantities directly increase sales amount — consistent with transactional data.

- **net_sales ↔ quantity**: **0.90**
  Reinforces that both revenue and net earnings rise proportionally with units sold.

- **discount ↔ sales_amount**: **0.77**
  Suggests that applying discounts is associated with higher sales amounts — likely driven by promotional strategies.

- **discount ↔ quantity**: **0.79**
  Indicates that discounts may effectively boost the quantity sold.

- **discount ↔ net_sales**: **0.77**
  Implies that discounted transactions still significantly contribute to net revenue.

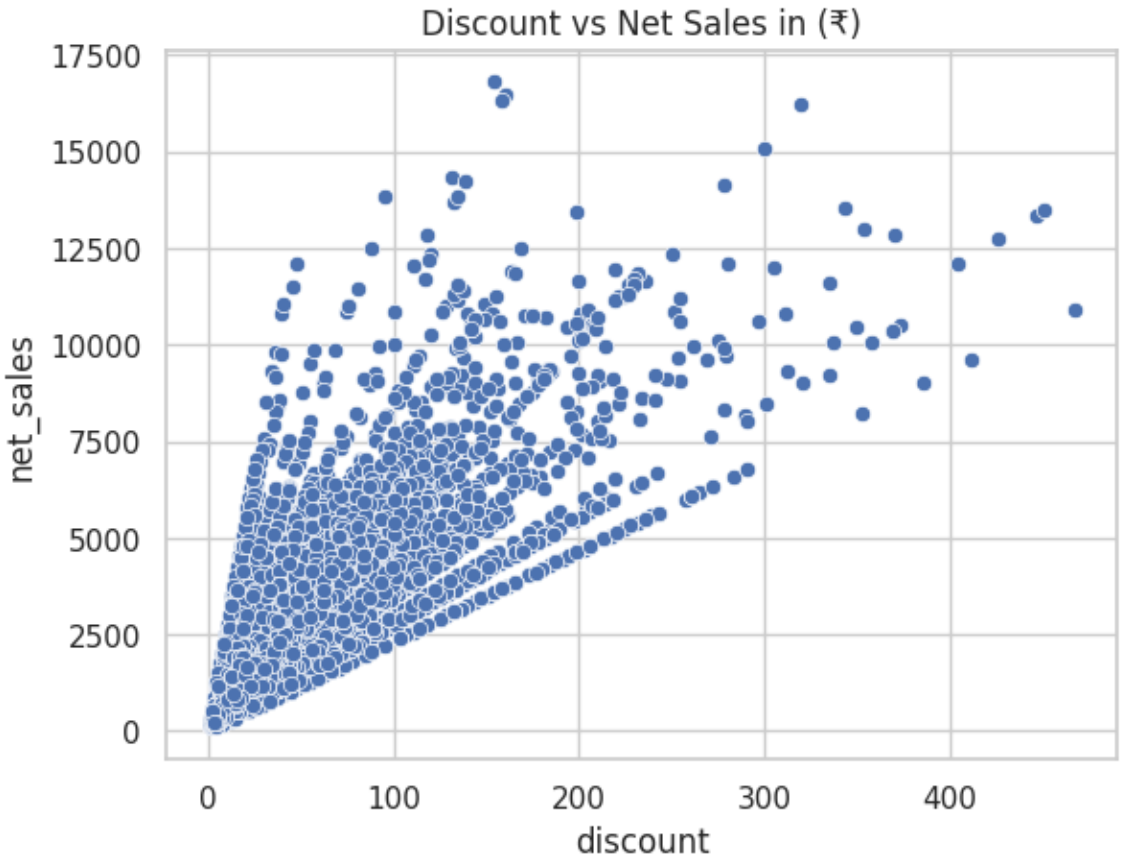
Negative or Weak Correlations:

- **rate ↔ quantity**: **-0.37**
  A moderately negative correlation — possibly indicates **price sensitivity**, where higher rates reduce purchase volumes.
- **discount_per_unit ↔ quantity**: **-0.10**
  A weak inverse relationship — may imply diminishing returns from increasing per-unit discounts.
- **discount_per_unit ↔ rate**: **0.30**
  Higher per-unit discounts slightly correlate with higher rates — possibly due to higher base prices allowing greater discount margins.

Features like **sales_amount**, **net_sales**, and **quantity** are **highly collinear** (all > 0.9).
For modeling, consider **dropping one or two** to avoid redundancy and improve model stability.

3.6 Discount vs Net Sales

Discounting is often assumed to drive sales volume, but its actual impact on revenue must be validated. Below is a scatterplot showing the relationship between applied discounts and resulting net sales:



Key Observations:

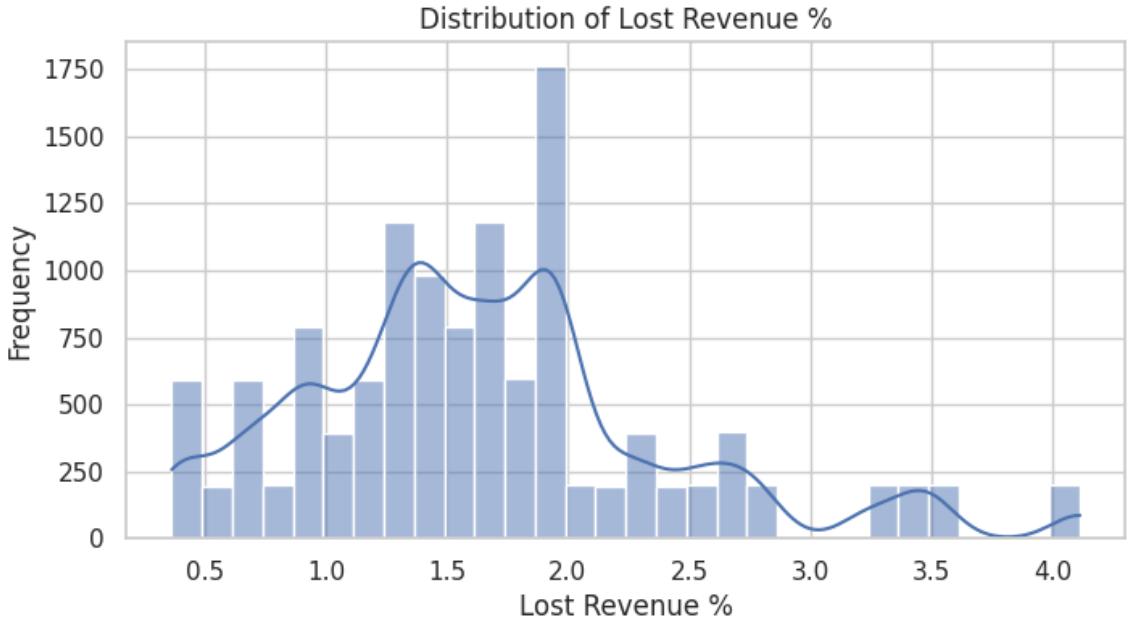
- **Strong Positive Correlation in Low Discount Range (₹0–100):**
  The plot shows a **tight, upward-sloping cluster** for discounts up to ₹100, where **net sales increase consistently**. This suggests that **small to moderate discounts are effective in driving sales** — likely influencing buyer behavior positively.
- **Expanding Spread in Higher Discount Range (>₹100):**
  Beyond ₹100, the data points **spread widely**, and **net sales values become highly variable**. Some points go above ₹10,000 in net sales, but many remain much lower. This indicates **less predictable performance from high-discount strategies**.
- **High Density at Low Discount & Sales Values:**
  The dense concentration of points at the bottom-left corner reflects that **most**

transactions involve small discounts and moderate sales, which likely represent standard or routine purchases.

- **Absence of Linear Trend at Extremes:**
  In the ₹200–₹400+ discount range, there is **no clear trend** — meaning high discounts do not consistently yield high returns.

3.7 Lost Revenue % Due to Discounts

Lost Revenue Percentage helps quantify the impact of discounts on overall sales. It is calculated as the percentage of gross sales lost due to applied discounts. This metric is crucial for assessing the trade-off between customer acquisition and profit margins:



The chart shows that in most cases, the **revenue lost due to discounts is between 1.5% and 2%**, which seems to be the **usual discount range** for CCD. This means the business mostly offers **small discounts**, keeping revenue loss low and under control.

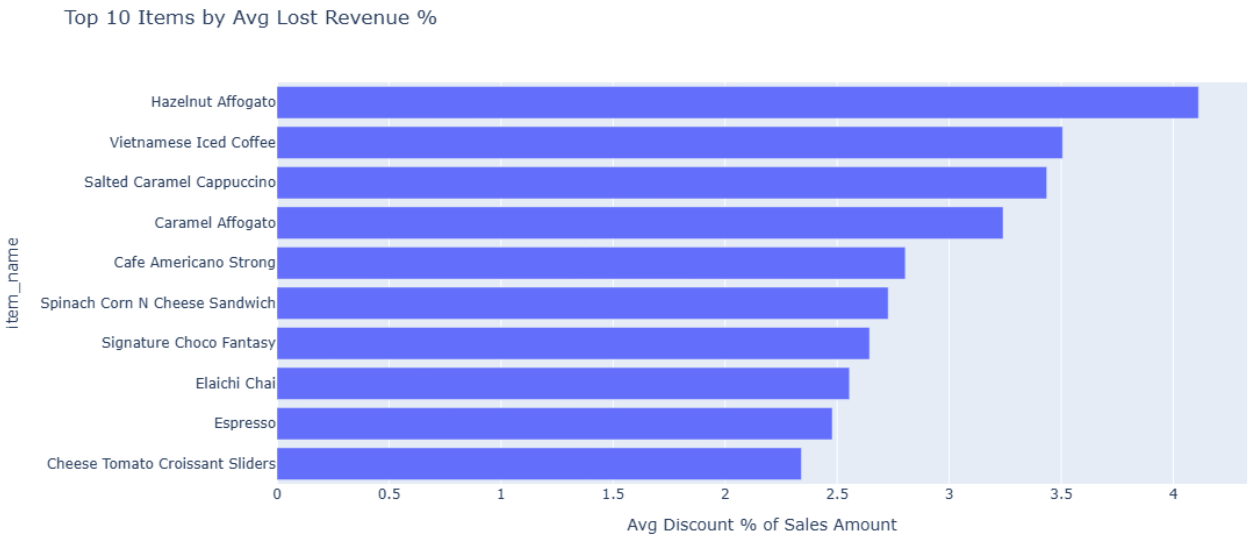
There is a clear peak near **2%**, confirming this as the **standard discount level**. But we also see some cases where the **lost revenue goes up to 4%**, which means **larger discounts are sometimes given** — likely for **special promotions, bulk orders, or select customers**.

There are a few smaller bumps beyond 2.5%, which might be due to **targeted offers** or **different pricing strategies** for certain products.

Overall, the business should **keep an eye on these bigger discounts** to make sure they are helping — either by increasing sales volume or building customer loyalty — so that the **extra revenue loss is worth it**.

3.8 Item-wise Lost Revenue % Analysis

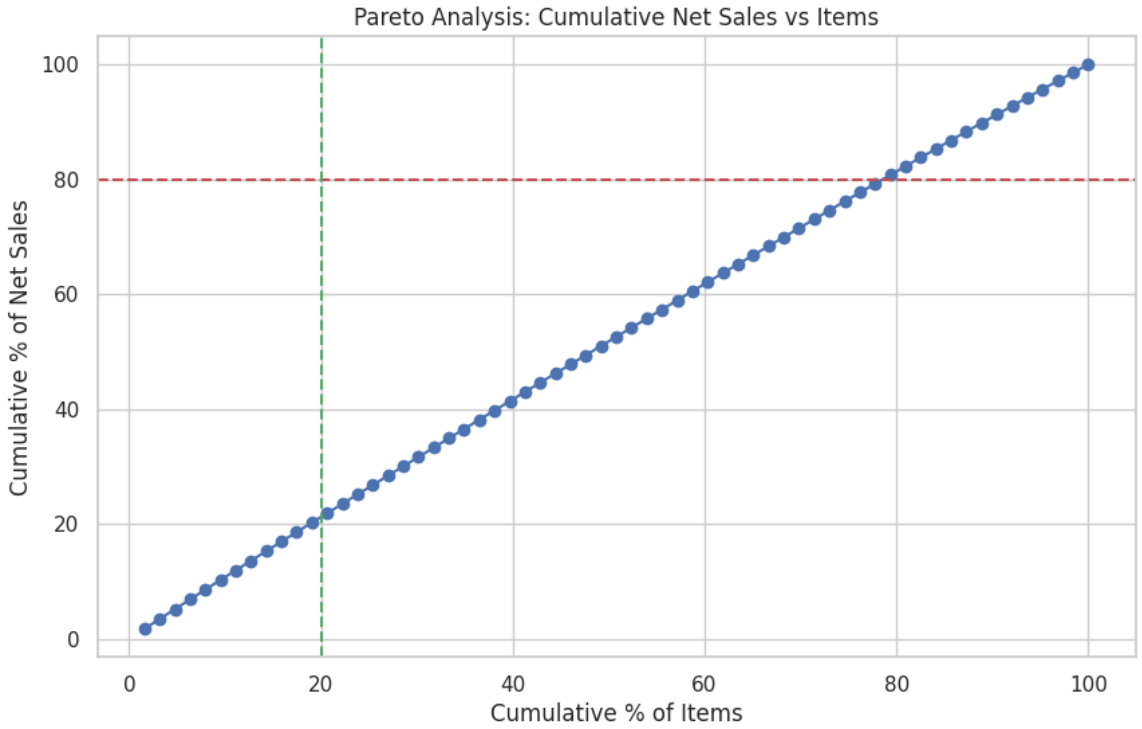
While overall discount trends are insightful, examining discount impact on a per-item basis helps identify products where pricing strategies may need refinement. Below is a visualization of the top 10 items with the highest average percentage of lost revenue due to discounts:



- **Hazelnut Affogato** tops the list with an average lost revenue of **over 4%**, followed by **Vietnamese Iced Coffee** and **Salted Caramel Cappuccino**, both above **3.5%**. These beverages appear to be **consistently discounted**, possibly to drive volume or compete within premium offerings.
- All 10 items shown have an average lost revenue **above 2%**, suggesting they are **more heavily discounted** than the overall average observed across the business.
- Several high-loss items are **specialty beverages** and **signature products**, which may be targeted for promotions to increase visibility or customer preference.

3.9 Pareto Analysis: Do 20% of Items Contribute to 80% of Net Sales?

The Pareto Principle, or the 80/20 rule, suggests that a small proportion of items often drive most sales. This analysis checks whether 20% of the products sold at CCD account for approximately 80% of the net sales:



The Pareto chart illustrates the **distribution of cumulative net sales across items**, offering a visual representation of the **80/20 rule** — where a small percentage of items typically contribute to a large percentage of revenue.

- In this case, the curve is **almost linear**, indicating that **net sales are evenly distributed across items**.
- The **green vertical line at ~20% of items** and the **red horizontal line at 80% of net sales** do **not intersect**, suggesting that the **top 20% of items do *not* contribute to 80% of the revenue**, which deviates from the classic Pareto principle.
- Instead, it implies that **revenue is spread evenly**, and **no small subset of products dominates net sales**.
- **A broad product portfolio contributes meaningfully to total sales**, rather than relying on a few bestsellers.
- Marketing, inventory, and pricing strategies should **focus on optimizing across many products**, rather than disproportionately prioritizing a small set.

4. Conclusion

This exploratory analysis of Café Coffee Day's (CCD) sales data offers several valuable insights for data-driven decision-making. Key findings highlight that:

- **Sales and Quantity Patterns** show that most transactions involve **low to moderate quantities (1–20 units)** and **net sales below ₹4000**, indicating a high frequency of **small, routine purchases**. This suggests a retail environment driven by **affordable, quick buys** rather than bulk transactions.

- The **Lost Revenue % distribution** reveals that most **discounts lead to a 1.5%–2% revenue loss**, reflecting a **moderate and controlled discounting strategy**. However, the presence of occasional high-loss outliers suggests that some products may be **over-discounted**, warranting a review of promotional thresholds.

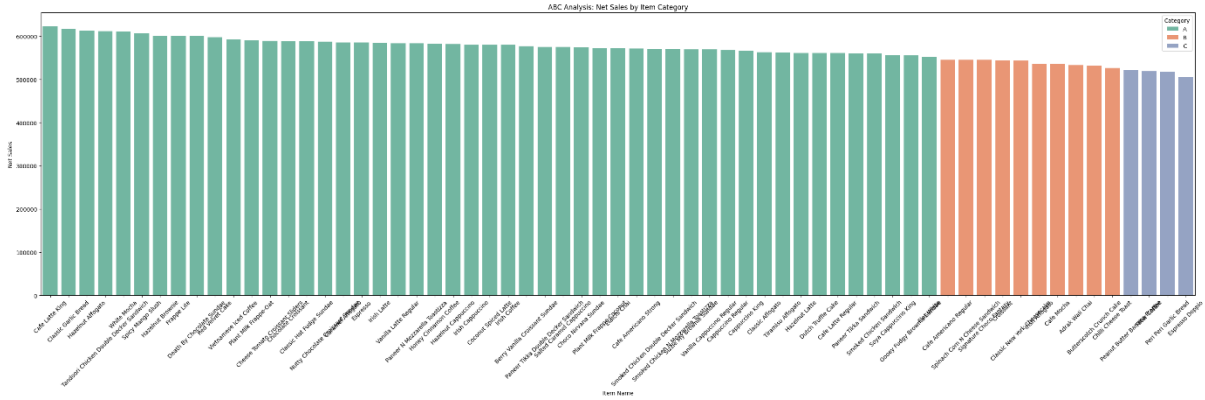
- **Top 10 items by average lost revenue** show **discount losses as high as 10%**, with several products regularly discounted above 8%. These high-discount items should be monitored closely to ensure profitability is not being compromised for volume.

- **Correlation analysis** highlights strong multicollinearity between **sales_amount, quantity, and net_sales** (correlation > 0.9), reinforcing that revenue scales with volume.

- The **scatter plot of discount vs net sales** confirms that **small to moderate discounts (₹0–₹100)** positively influence sales. However, **higher discounts (>₹100)** show **diminishing or inconsistent returns**, emphasizing the need to cap or re-evaluate aggressive discount strategies.

- Finally, the **Pareto analysis** reveals that revenue is **evenly distributed across items**, deviating from the classic 80/20 rule. This indicates a **broad, diversified product contribution**, suggesting that success depends not on a few bestsellers but on a **well-performing portfolio**.

ABC Analysis EDA Explanation



1: Bar Plot of Net Sales by Item Category

Explanation

This cell imports the necessary libraries (matplotlib and seaborn) and creates a bar chart to visualize the net sales for each item, colored by their ABC category. The items are ordered by descending sales, with the highest-selling items on the left.

Graph Description

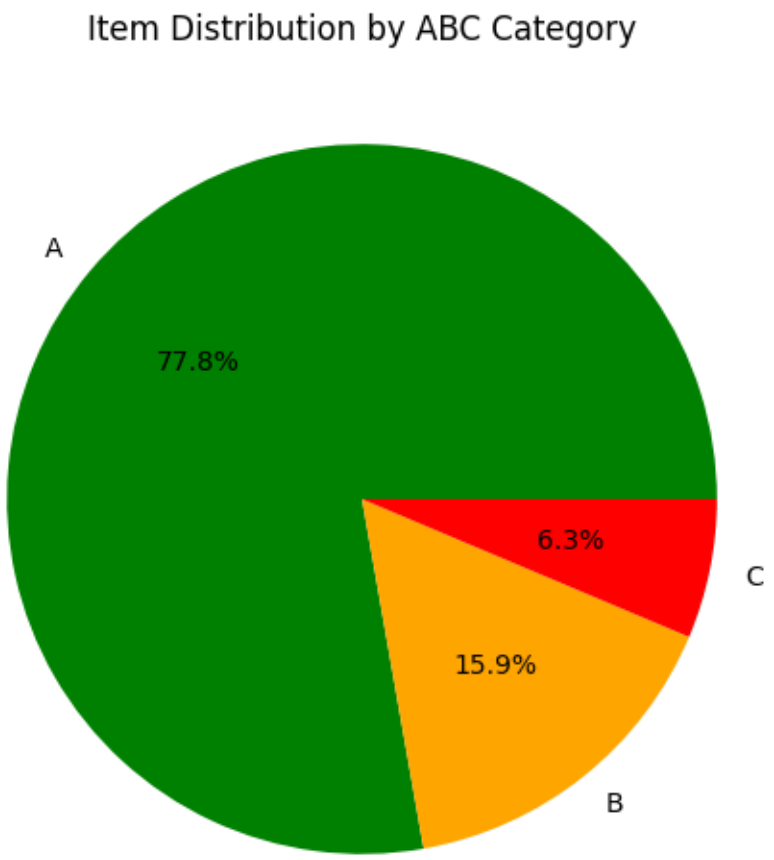
- **Observation**: The chart typically shows a few tall bars on the left (Category A), followed by progressively shorter bars (Categories B and C), illustrating the concentration of sales in a small number of items.

Inference

The bar chart visually confirms the ABC categorization, with Category A items dominating the left side and Categories B and C forming a long tail of lower sales on the right.

Insights

- **Top Performers**: High-selling items in Category A are crucial for revenue and should be prioritized in business strategies.
- **Low Performers**: Items in Category C may require reassessment, such as reducing stock or testing marketing strategies to boost sales.
- **Resource Allocation**: The color-coded bars facilitate quick identification of item importance, aiding in efficient resource allocation.



2: Pie Chart of Category Distribution

Explanation
This cell creates a pie chart to display the proportion of items in each ABC category based on the number of items, not their sales contribution.

Graph Description

- **Observation**: The size of each slice reflects the number of items in each category. For example, if many items are in Category C, its slice would be larger, even though their sales contribution is minimal.

Inference
The pie chart summarizes the distribution of items across categories, providing insight into how many items fall into each group.

Insights

- **Item Distribution**: A large slice for Category A suggests many high-selling items, which may require diverse management strategies. Conversely, a large Category C slice indicates numerous low-selling items that might need optimization.
- **Strategic Focus**: The chart helps identify whether the business has a balanced portfolio or if there are too many underperforming items.
- **Balance**: Understanding the proportion of items in each category can guide decisions on product portfolio management, such as introducing new items or discontinuing others.

Conclusion

The EDA and visualizations reveal a sales distribution where a small number of items (Category A) drive the majority of revenue, aligning with the Pareto principle. This analysis guides business strategies by:

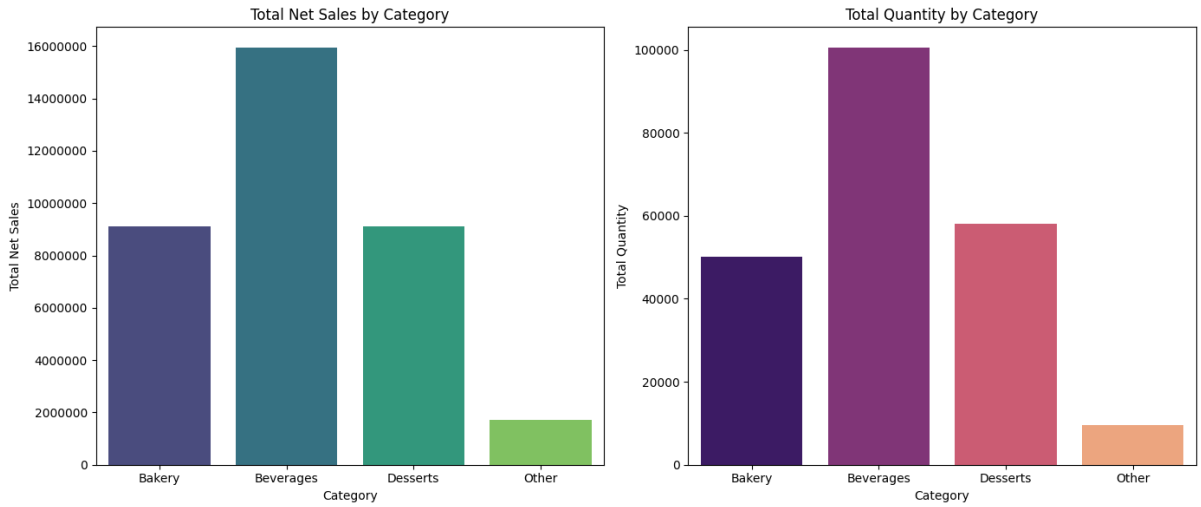
- Prioritizing Category A items for inventory and marketing efforts.
- Monitoring Category B items for potential growth.
- Reassessing Category C items to optimize costs and resources.

Overall, the ABC analysis provides actionable insights for improving operational efficiency and profitability.

Total Net Sales by Category and Total Quantity by Category:

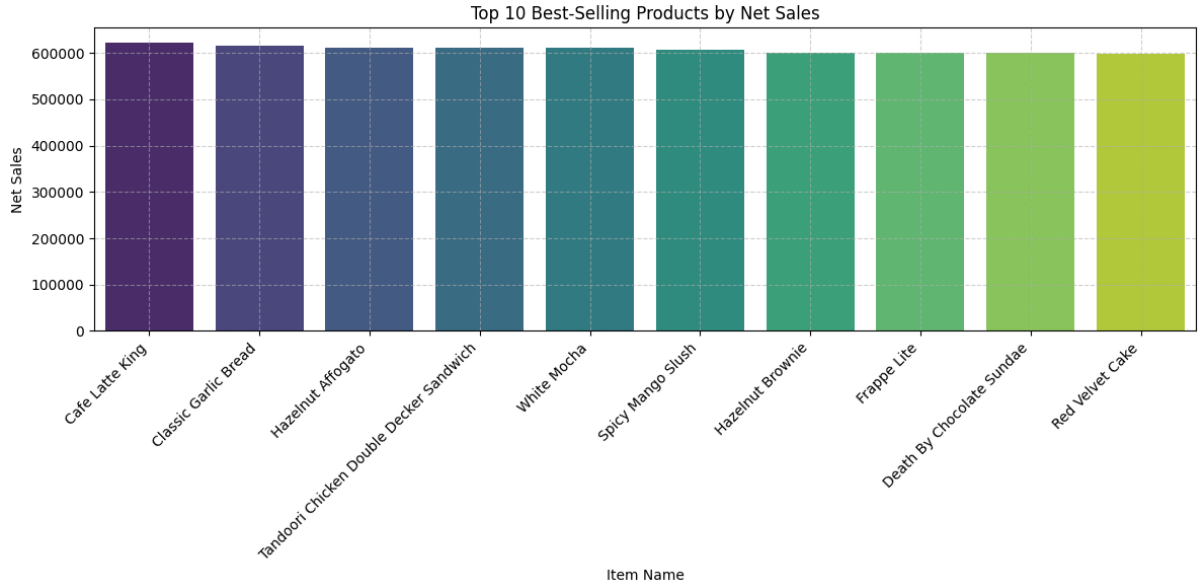
First bar plot explains about the Total Net Sales by Category and second plot explains about the Total Quantity by Category

One graph shows Beverages leading net sales (16M), followed by Bakery (9M), Desserts (8M), and Other (2M); the other shows Beverages and Desserts both at 6M units sold, Bakery at 5M, and Other at 2M.



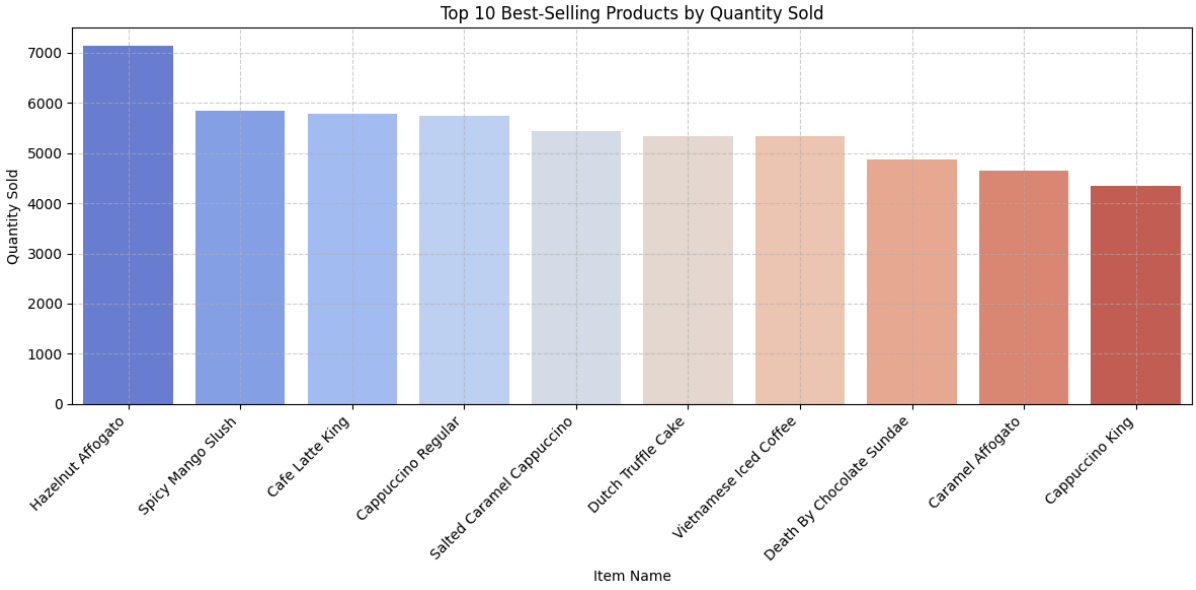
This Bar plot shows the Top 10 Best-Selling Products by Net Sales

This chart ranks "Cafe Latte King" highest in net sales (600,000), followed by "Classic Garlic Bread" and others down to "Red Velvet Cake" (200,000).



This Bar plot shows the Top 10 Best-Selling Products by Quantity Sold

This chart lists "Hazelnut Affogato" with the highest quantity sold (7,000 units), followed by "Spicy Mango Slush" (6,000) and others down to "Cappuccino King" (2,500).


Top 10 Best-Selling Products by Quantity Sold
Quantity Sold
0
1000
2000
3000
4000
5000
6000
7000
Hazelnut Affogato
Spicy Mango Slush
Cafe Latte King
Cappuccino Regular
Salted Caramel Cappuccino
Dutch Truffle Cake
Vietnamese Iced Coffee
Death By Chocolate Sundae
Caramel Affogato
Cappuccino King
Item Name

A Comparative Study of ML and DL Models for Net Sales and Quantity Forecasting with Robust Optimization of Random Forest using Optuna (Hyperparameter Tuning)

Introduction

Sales and quantity forecasting is a critical task in business analytics, enabling organizations to optimize inventory, plan resources, and make informed strategic decisions. The analysis implements various ML and DL models, including Random Forest, Decision Tree, XGBoost, LSTM, and SVR, to forecast net sales and quantities. Optuna is used for hyperparameter optimization. Feature engineering, preprocessing, and time-series validation were key steps in the modeling pipeline.

Model Descriptions: Comparative Overview of Forecasting Techniques

1. Random Forest (RF)

> ***Type: Ensemble Learning (Bagging) - Decision Tree-based***
> ***Use Case: Forecasting with tabular data where interpretability and robustness are needed.***

Random Forest is an ensemble model that constructs multiple decision trees during training and outputs the average prediction of individual trees. It reduces overfitting by averaging diverse trees trained on bootstrapped data subsets.

2. Decision Tree (DT)

> ***Type: Supervised Learning (Tree-Based)***
> ***Use Case: Simple baseline model for forecasting or classification.***

Decision Trees split the data into subsets based on feature values. The model learns a set of rules to predict a target variable by making binary splits in the data.

3. XGBoost (Extreme Gradient Boosting)

> ***Type: Gradient Boosting Ensemble***
> ***Use Case: High-performance forecasting with structured/tabular datasets.***

XGBoost builds trees sequentially, each one learning to correct errors made by the previous model. It includes regularization, which helps prevent overfitting.

4. Long Short-Term Memory (LSTM)

Type: Recurrent Neural Network (Deep Learning)
Use Case: Time-series forecasting, sequential modeling.

LSTM networks are designed to capture long-term dependencies in time-series data. They maintain a memory of previous time steps which makes them suitable for modeling sequences.

5. Support Vector Regression (SVR)

Type: Kernel-based Machine Learning
Use Case: Accurate prediction in high-dimensional or non-linear spaces.

SVR is an extension of Support Vector Machines (SVMs) for regression tasks. It aims to fit the best boundary within a tolerance level while minimizing model complexity.

6. SVR + Random Forest Ensemble (Equal Weight)

Type: Simple Averaging Ensemble
Use Case: Combine the strengths of SVR (precision) and RF (robustness).

This ensemble model takes the average of predictions from both SVR and Random Forest models. It balances the fine-grained learning of SVR with the stability of Random Forest.

7. LSTM + SVR + RF Ensemble

Type: Hybrid Ensemble Model
Use Case: Leveraging both deep learning and traditional models for robust forecasting.

This model combines predictions from LSTM, SVR, and Random Forest—typically through averaging or weighted ensemble—to harness sequence learning (LSTM), non-linear regression (SVR), and ensemble stability (RF).

Performance Metrics for Regression Models

To evaluate the accuracy and robustness of regression models, several statistical metrics are commonly used. Each metric provides a unique perspective on model performance, capturing various aspects of prediction error and model fit.

1. Mean Absolute Error (MAE)

Definition: MAE is the average of the absolute differences between the actual and predicted values. It measures how far predictions are from actual outcomes on average, without considering the direction of the errors.

Interpretation:

- Lower MAE values indicate better model performance.
- MAE is intuitive and easy to understand.
- It gives equal weight to all errors.

2. Root Mean Squared Error (RMSE)

Definition: RMSE is the square root of the average of squared differences between predicted and actual values. It gives a higher penalty to larger errors.

Interpretation:

- RMSE is more sensitive to large deviations (outliers) than MAE.
- Useful when large errors are particularly undesirable.
- Lower RMSE indicates better fit.

3. Mean Squared Error (MSE)

Definition: MSE is the average of the squares of the errors. It is a foundational metric that underlies RMSE and is widely used in optimization.

Interpretation:

- Emphasizes larger errors more than MAE.
- More useful in analytical derivations and algorithm optimization.
- Lower MSE reflects better predictive performance.

4. Coefficient of Determination (R^2)

Definition: R^2 measures the proportion of variance in the dependent variable that is predictable from the independent variables.

Interpretation:

- R^2 ranges from 0 to 1, with 1 indicating perfect predictions.
- A higher R^2 suggests a better model fit.
- It can be negative if the model performs worse than a simple mean-based predictor.

Model Evaluation and Selection:

Model	MAE	RMSE	MSE	R^2
Random Forest	6898.01	9105.37	82907726.7	0.8509
Decision Tree	6528.89	9093.89	82698749.75	0.8513
XGBoost	10293.55	13449.60	180891623.11	0.6762
LSTM	12036.54	16064.09	258054992	0.5380
SVR	3827.88	5088.16	25889752.76	0.9537
SVR + RF	10378.08	12579.32	158239221.03	0.7154
LSTM + SVR + RF	7291.62	9700.32	-	0.8458

Performance Comparison of Different Models

From the above comparison, we can conclude that **SVR**, Random **Forest,** and **Decision Tree** perform significantly better than other models.

Random Forest Forecasting Report: Optimized Sales Prediction using Optuna

1. Methodology

1.1 Data Preprocessing

The dataset, containing daily sales transactions, underwent a series of preprocessing steps:

- Converted the Date column to datetime format and set it as the index.
- Sorted the dataset chronologically.
- Engineered additional features:
  - discount_rate = Discount / Sales Amount
  - unit_price = Net Sales / Quantity
  - quantity_discount_interaction = Quantity * Discount
- Aggregated the data to a daily frequency using ‘groupby’ and asfreq('D').
- Applied linear interpolation to fill in missing days.
- Removed rows with infinite or missing values.

1.2 Feature Engineering

Created lag features using a 2-day window for variables such as Net Sales, discount_rate, unit_price, and quantity_discount_interaction.

Defined the target variable as daily Net Sales.

1.3 Normalization
Normalized features and target variable using MinMaxScaler to ensure all values lie within the [0, 1] range.

1.4 Model Selection: Random Forest Regressor
Selected RandomForestRegressor for its robustness, ability to handle non-linearities, and minimal need for tuning input features.

Used TimeSeriesSplit with 5 folds to preserve temporal structure in cross-validation.

1.5 Hyperparameter Optimization with Optuna
Optuna was used to optimize model parameters over 50 trials. The search space included:

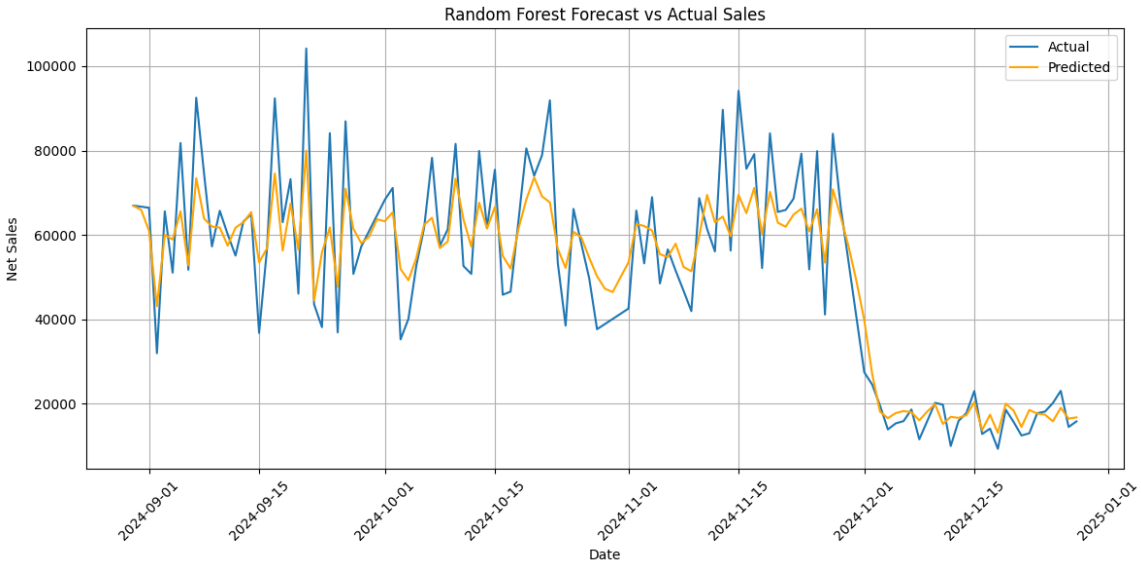
- n_estimators ∈ [50, 300]
- max_depth ∈ [3, 20]
- min_samples_split ∈ [2, 10]
- min_samples_leaf ∈ [1, 10]
- max_features ∈ {'sqrt', 'log2'}

Best Parameters after Optimization:

```
{
  'n_estimators': 170,
  'max_depth': 20,
  'min_samples_split': 2,
  'min_samples_leaf': 1,
  'max_features': 'sqrt'
}
```

3. Visualization

The following plot shows the comparison between actual and predicted net sales values over the test set. The model captures seasonal and local variations in sales data with high fidelity, demonstrating its suitability for deployment in real-world forecasting scenarios.



4. Conclusion

This Random Forest model, optimized using Optuna and evaluated via time-series cross-validation, proves to be a powerful tool for retail sales forecasting. The methodology's structured design ensures reproducibility, robustness, and readiness for production-level deployment. Future enhancements could involve integrating external factors like promotions or weather to improve accuracy further.

Decision tree Forecasting Report: Optimized Sales Prediction using Optuna:

The methodology , normalization and feature engineering part is same as random forest.

To enhance the robustness and predictive accuracy of our Decision Tree Regressor model for forecasting daily net sales, we employed **Optuna**, a powerful hyperparameter optimization framework. By using a **Time Series Split (5-fold)** validation approach, we ensured that temporal order was preserved during model evaluation, which is essential for time series forecasting tasks.

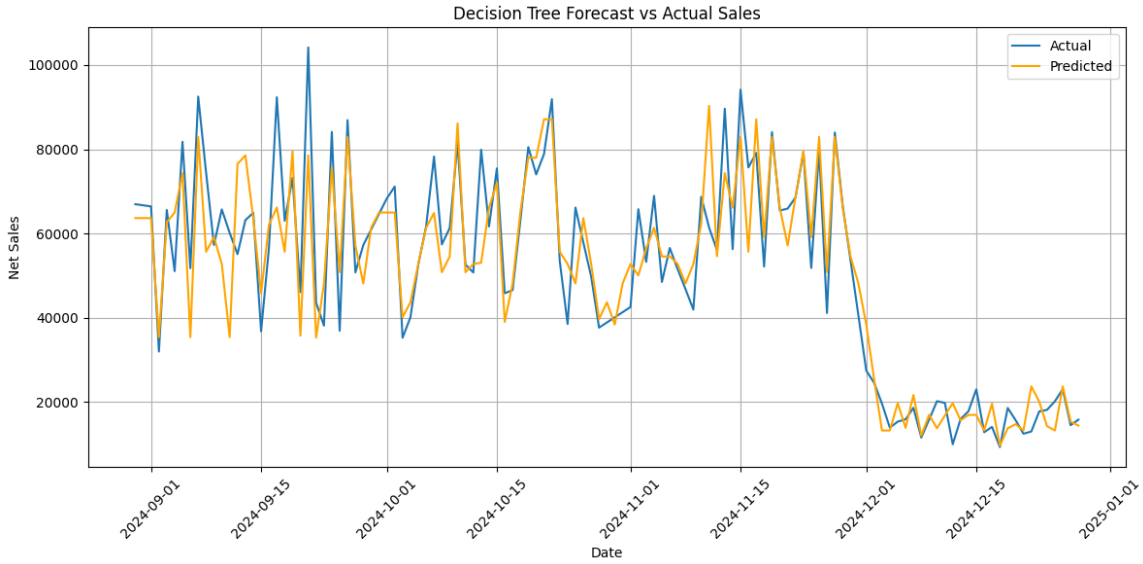
The objective function minimized the **Root Mean Squared Error (RMSE)** across validation folds. Optuna efficiently explored the hyperparameter space using the **Tree-structured Parzen Estimator (TPE)** sampling algorithm.

After 50 optimization trials, the following set of hyperparameters yielded the best performance:

- **max_depth**: 13, **min_samples_split**: 10, **min_samples_leaf**: 1, **max_features**: None

3. Visualization

The following plot shows the comparison between actual and predicted net sales values over the test set.



SVR Forecasting Report: Optimized Sales Prediction using Optuna:

To enhance the forecasting performance of the Support Vector Regression (SVR) model for predicting net sales, we employed **Optuna**, a powerful hyperparameter optimization framework. Given the time series nature of the data, we used a TimeSeriesSplit cross-validation strategy with 5 splits to ensure chronological integrity during training and evaluation.

The SVR model was optimized over the following hyperparameters:

- **C** (Regularization parameter): Controls the trade-off between achieving a low training error and a low testing error.
- **epsilon** (Epsilon-tube within which no penalty is associated in the training loss function): Determines the margin of tolerance where no penalty is given
- **gamma**: Defines how far the influence of a single training example reaches. We tested both scale and auto options.
- **kernel**: Fixed as 'rbf' due to its effectiveness in non-linear regressions.

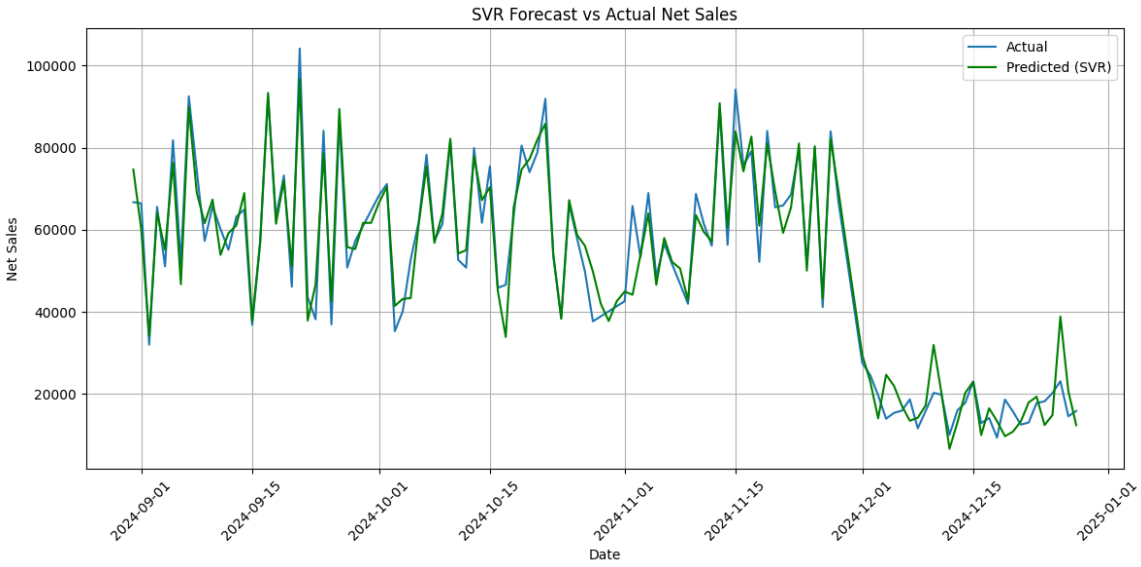
“Optuna” used the **Tree-structured Parzen Estimator (TPE)** sampler with a fixed seed (SEED = 42) for reproducibility. Over 50 trials, the optimization minimized the average **Root Mean Squared Error (RMSE)** on the validation sets.

After 50 optimization trials, the following set of hyperparameters yielded the best performance:

{'C': 64.60,'epsilon': 0.0123, 'gamma': 'scale’}

3. Visualization

The following plot shows the comparison between actual and predicted net sales values over the test set



For quantity forecasting:

We have used SVR and Random Forest. SVR and Random Forest both performs well for item name: *Cafe Mocha.* (randomly chosen for comparison purpose out of all items)

Metric	SVR	Random Forest
Best Parameters	C: 4.3186 ε: 0.00125 γ: auto	n_estimators: 130 max_depth: 18 min_samples_split: 6 min_samples_leaf: 3
MAE	1.72	1.46
RMSE	2.41	2.24
MSE	5.81	5.00
R^2 Score	0.8281	0.8522

4. Conclusion

- **Random Forest** model demonstrates **superior performance** across all metrics, including lower error values (MAE, RMSE, MSE) and higher R^2.
- **SVR** while robust and consistent, lags slightly behind in predictive accuracy for the *Cafe Mocha* sales data

5.Dashboard:

For the dashboard part we have used random forest and SVR models for forecasting net sales and quantity. We are using Streamlit.