# PREDICTIVE ANALYSIS FOR RETAIL DEMAND FORECASTING


By Naveen, Shinjini Pal &
Swathi N

# PROBLEM STATEMENT: RETAIL DEMAND FORECASTING

- Retail businesses face challenges in managing inventory efficiently. Overstocking leads to increased holding costs, while understocking results in lost sales and customer dissatisfaction.
- This project aims to:
  - **Predict demand** for retail products using historical sales data.
  - **Optimize inventory levels** to balance supply and demand.
  - **Incorporate external factors** (e.g., weather, seasonality, competitor pricing) for better accuracy.
  - **Enhance decision-making** in pricing, promotions, and stock replenishment.
- By leveraging **predictive analytics**, retailers can **minimize waste, maximize profits, and improve customer satisfaction**.

# CHALLENGES IN RETAIL DEMAND FORECASTING

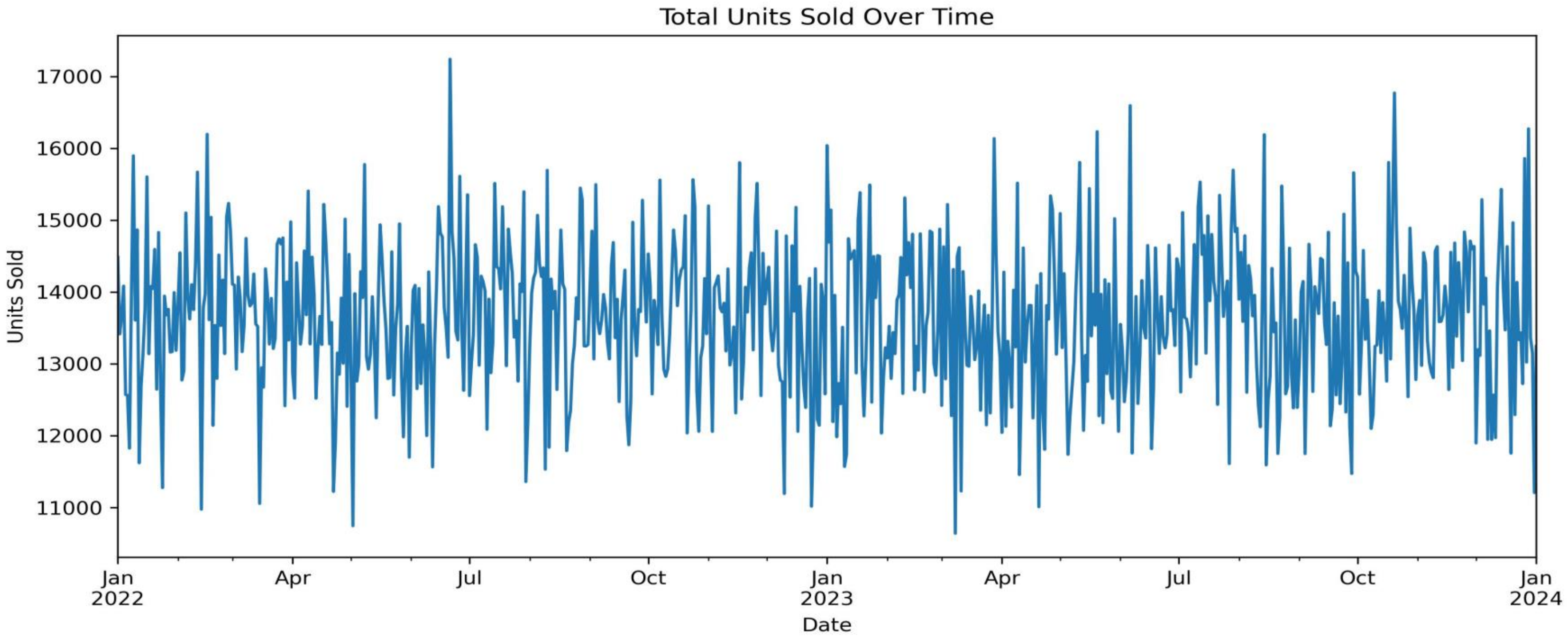
- **Data Quality:** Inaccurate or incomplete data hampers forecasting.
- **Seasonality:** Demand changes with seasons and holidays.
- Promotions: Special offers can disrupt forecast accuracy.
- **External Factors:** Economic and environmental changes are unpredictable.
- **Granularity:** Balancing forecast detail with accuracy.
- **Model Complexity:** Selecting and tuning the right forecasting model.
- **Inventory Alignment**: Matching demand with inventory levels

## FORECASTING IS THE ART OF SAYING WHAT WILL HAPPEN AND THEN EXPLAINING WHY IT DIDN'T!

# RETAIL STORE INVENTORY DATASET

- The dataset contains **73,100 rows and 15 columns** related to retail inventory, sales, and demand forecasting.
- Key columns include:
  - **Date, Store ID, Product ID, Category, Region**
  - **Inventory Level, Units Sold, Units Ordered, Demand Forecast**
  - **Price, Discount, Competitor Pricing**
  - **External Factors:** Weather Condition, Holiday/Promotion, and Seasonality

**Exploratory Data Analysis (EDA) –**
✅ Basic statistics & missing values check
✅ Visualizations (sales trends, correlations, category-wise demand)
✅ Outlier detection



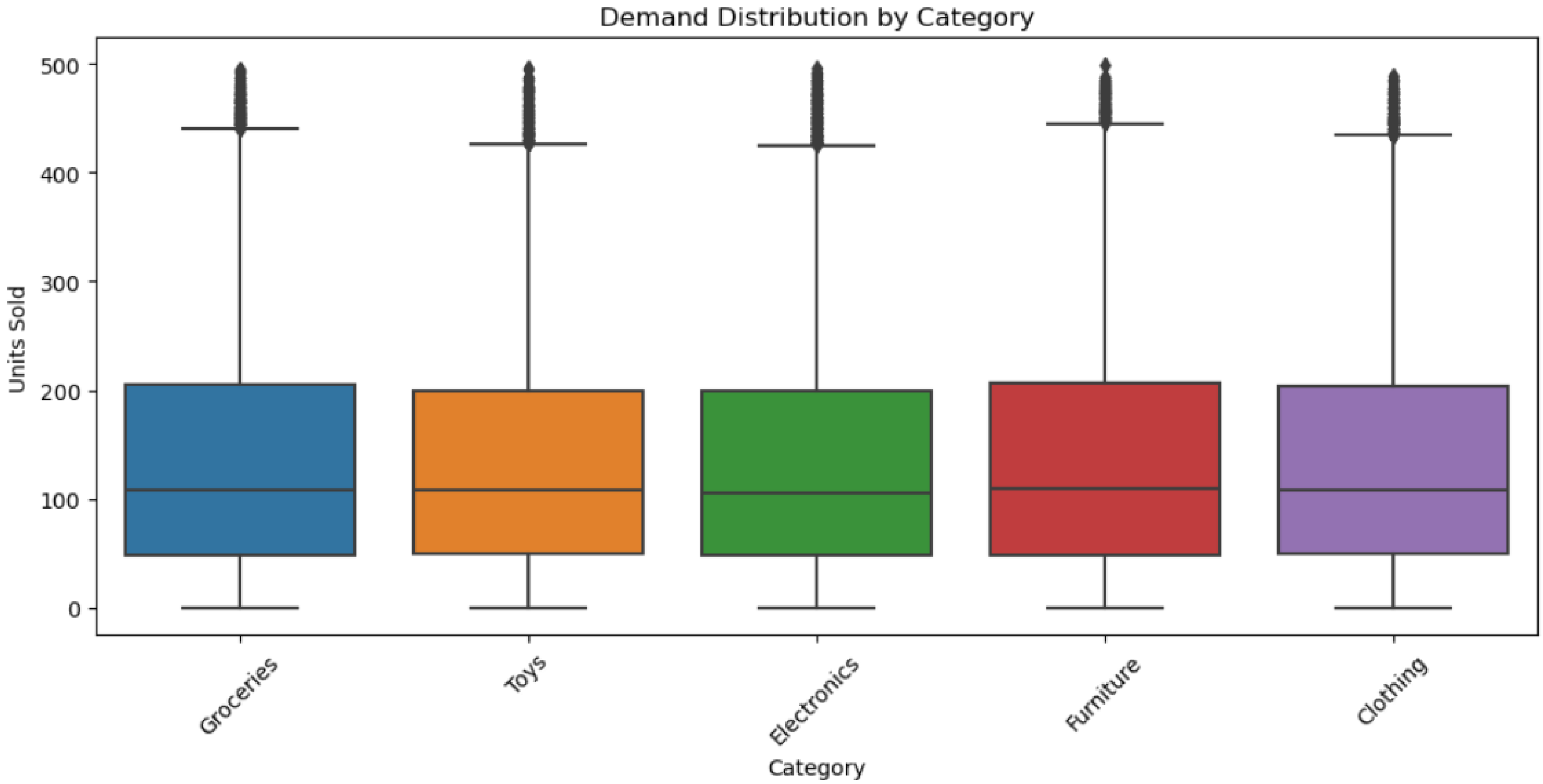
Total Units Sold Over Time
17000
16000
15000
14000
13000
12000
11000
Units Sold
Jan
2022
Apr
Jul
Oct
Jan
2023
Apr
Jul
Oct
Jan
2024
Date

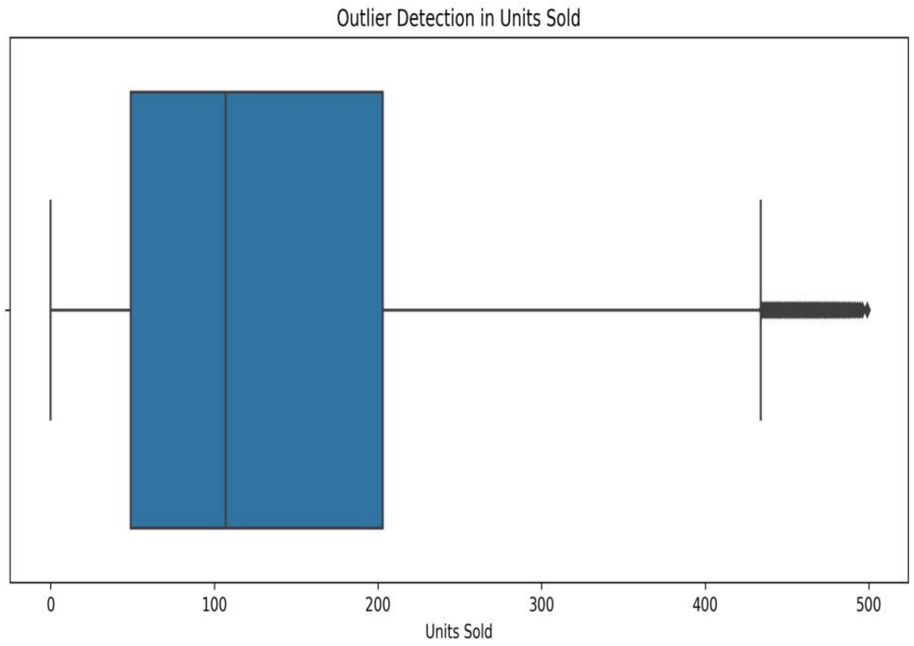
```
Summary Statistics:
       Inventory Level    Units Sold  Units Ordered  Demand Forecast  \
count     73100.000000  73100.000000   73100.000000     73100.000000
mean        274.469877    136.464870     110.004473       141.494720
std         129.949514    108.919406      52.277448       109.254076
min          50.000000      0.000000      20.000000        -9.990000
25%         162.000000     49.000000      65.000000        53.670000
50%         273.000000    107.000000     110.000000       113.015000
75%         387.000000    203.000000     155.000000       208.052500
max         500.000000    499.000000     200.000000       518.550000

              Price      Discount  Holiday/Promotion  Competitor Pricing
count  73100.000000  73100.000000       73100.000000        73100.000000
mean      55.135108     10.009508           0.497305           55.146077
std       26.021945      7.083746           0.499996           26.191408
min       10.000000      0.000000           0.000000            5.030000
25%       32.650000      5.000000           0.000000           32.680000
50%       55.050000     10.000000           0.000000           55.010000
75%       77.860000     15.000000           1.000000           77.820000
max      100.000000     20.000000           1.000000          104.940000
```

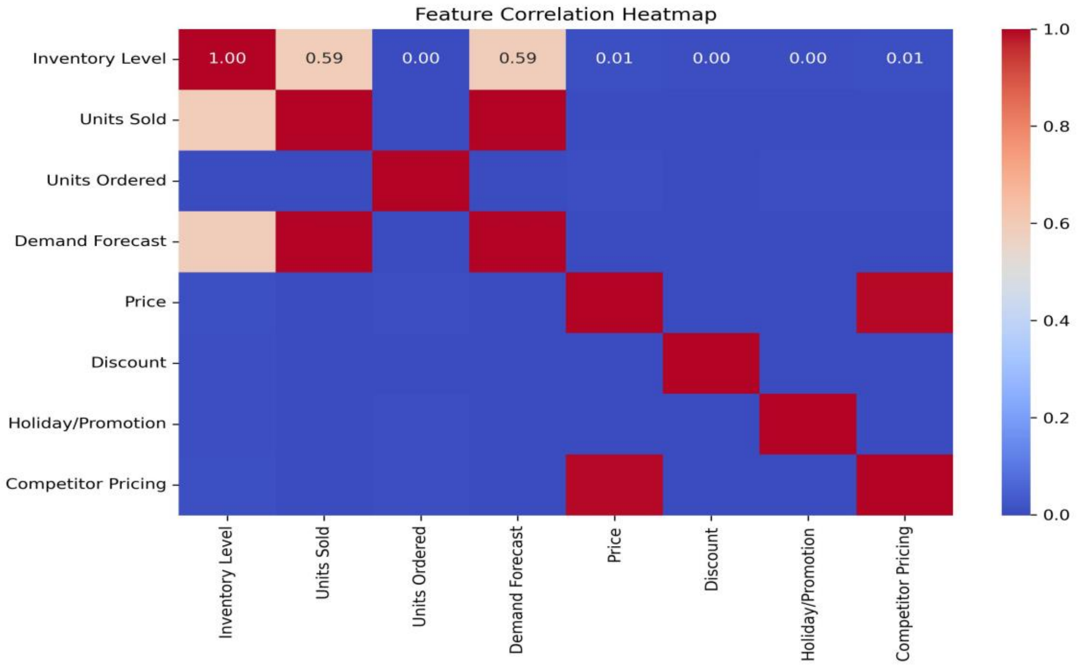
```
Dataset Overview:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 73100 entries, 0 to 73099
Data columns (total 15 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Date                73100 non-null  object
 1   Store ID            73100 non-null  object
 2   Product ID          73100 non-null  object
 3   Category            73100 non-null  object
 4   Region              73100 non-null  object
 5   Inventory Level     73100 non-null  int64
 6   Units Sold          73100 non-null  int64
 7   Units Ordered       73100 non-null  int64
 8   Demand Forecast     73100 non-null  float64
 9   Price               73100 non-null  float64
 10  Discount            73100 non-null  int64
 11  Weather Condition   73100 non-null  object
 12  Holiday/Promotion   73100 non-null  int64
 13  Competitor Pricing  73100 non-null  float64
 14  Seasonality         73100 non-null  object
dtypes: float64(3), int64(5), object(7)
memory usage: 8.4+ MB
```

```
Missing Values:
Date                  0
Store ID              0
Product ID            0
Category              0
Region                0
Inventory Level       0
Units Sold            0
Units Ordered         0
Demand Forecast       0
Price                 0
Discount              0
Weather Condition     0
Holiday/Promotion     0
Competitor Pricing    0
Seasonality           0
dtype: int64
```

```
Missing Values:
```




Outlier Detection in Units Sold
0
100
200
300
400
500
Units Sold


Feature Correlation Heatmap
Inventory Level
Units Sold
Units Ordered
Demand Forecast
Price
Discount
Holiday/Promotion
Competitor Pricing
1.00
0.59
0.00
0.59
0.01
0.00
0.00
0.01
1.0
0.8
0.6
0.4
0.2
0.0

- **Training & Evaluation**
  - **Train/Test Split:** 80/20 ratio.
  - **Performance Metrics:** MAE, RMSE, R^2 Score.
  - **Cross-Validation:** k-Fold validation to improve generalization.
- **6. Deployment Strategy**
  - **Containerization:** Deploying the model with Docker.
  - **API Development:** Flask/FastAPI for model inference.
  - **CI/CD Pipeline:** Automating model updates.

- **Data Preprocessing**
  - **Handling Missing Values:** Imputation methods.
  - **Feature Engineering:** Creating new features like demand trends.
  - **Normalization & Encoding:** Scaling numerical features, one-hot encoding categorical features.
- **4. Model Selection**
- **Why Random Forest?**
  - Handles non-linearity and feature interactions.
  - Robust against overfitting with ensemble learning.
  - Handles missing values well.
- **Hyperparameter Tuning:** Grid Search, Random Search.