



SmartStock Inventory Optimization for Retail Stores



CONTENT PAGE

1. Project Overview
2. Problem Statement & Goals
3. Methodology / Workflow
4. Machine Learning Model (LightGBM)
5. Results & Visualizations
6. Conclusion & Future Enhancements





Title & Overview

SmartStock Inventory Optimization for Retail Stores



Objective:

Develop an ML-based demand forecasting system to reduce inventory losses by minimizing overstock and stockout issues.

Tech Stack:

Python, LightGBM, Pandas, Matplotlib, Seaborn, Scikit-learn, Hugging Face Datasets.



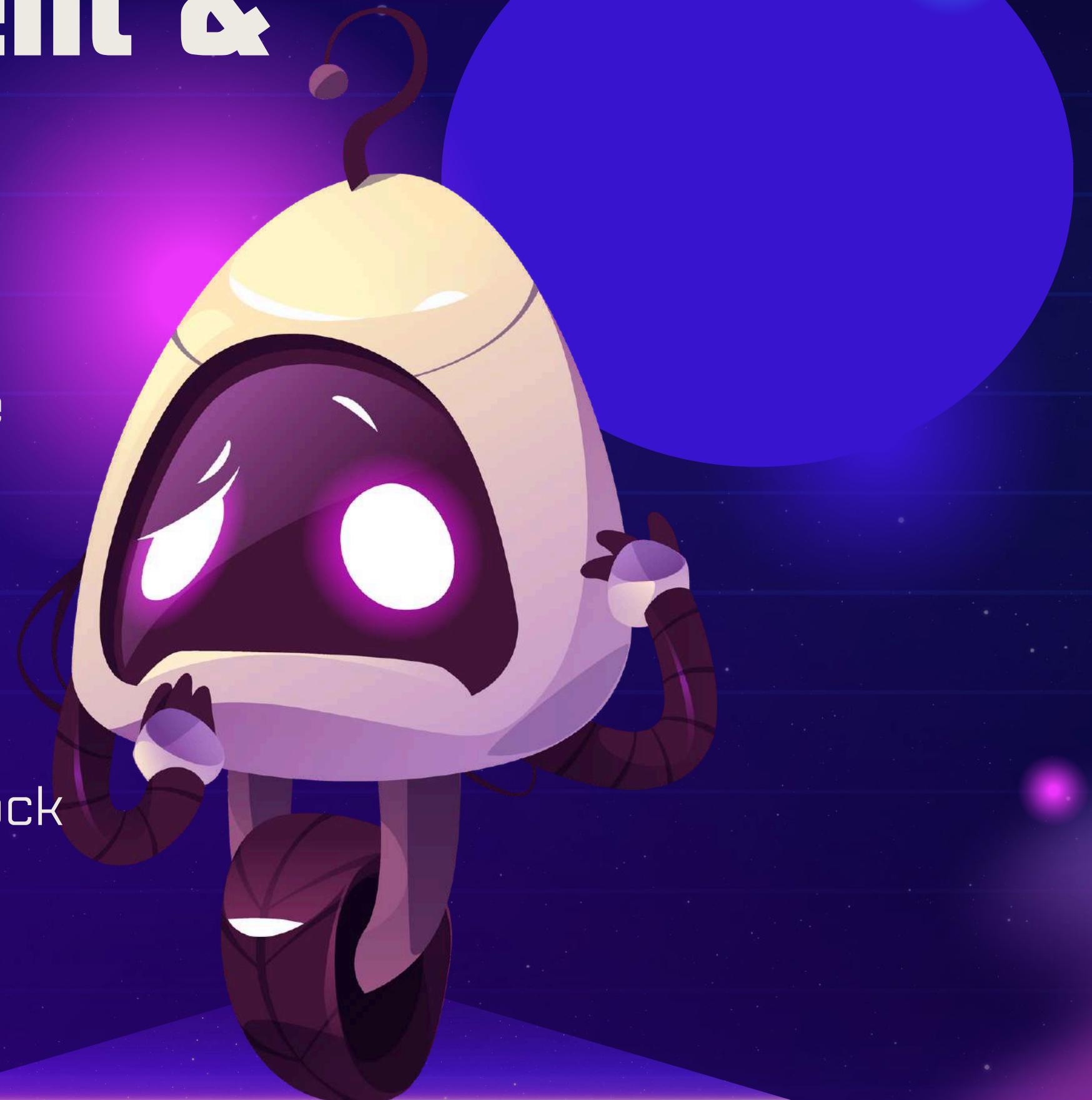
Problem Statement & Goals

Problem Statement:

- Retail businesses face losses due to:
- Overstock → Increased storage & wastage
- Stockout → Lost sales & poor customer experience

Goal:

- Build a predictive model that:
- Forecasts sales per store & product
- Classifies stock as Understock ●, Overstock ●, Balanced ●
- Helps retailers maintain optimal inventory levels



Methodology / Workflow

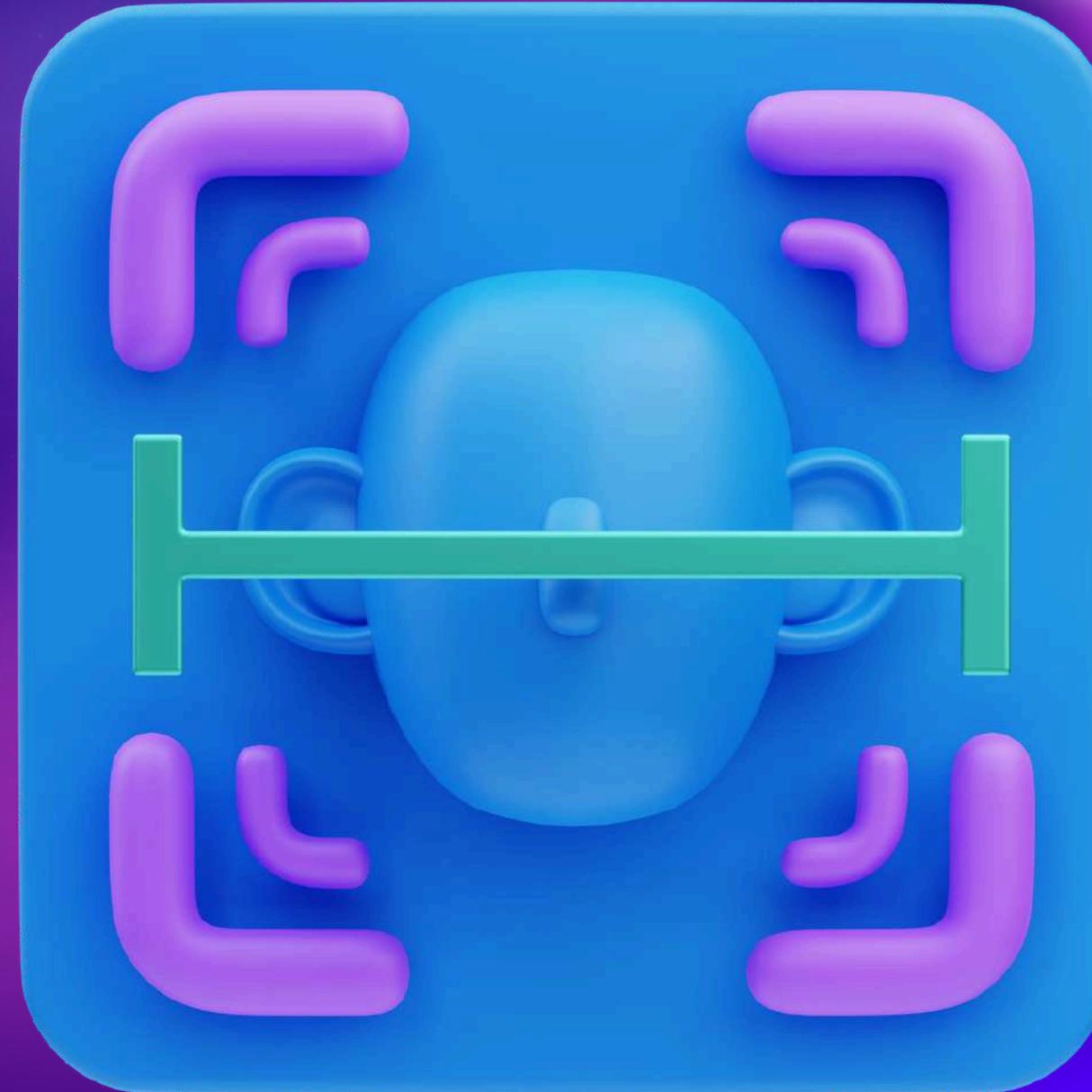
Data Pipeline Steps

1. **Data Loading:** Hugging Face dataset([store-sales-time-series-forecasting](#))
2. **Preprocessing:** Categorical encoding, date conversion
3. **Feature Engineering:**
 - Time features: day, month, weekday
 - Lag features: 7, 14, 28 days
 - Rolling features: moving averages & stds
4. **Model Training:** LightGBM regression
5. **Evaluation:** RMSE
6. **Reporting & Visualization**

Workflow Diagram

Data → Preprocess → Feature Engineering → Model Training → Forecast → Stock Classification → Report





Machine Learning Model (LightGBM)

Why LightGBM?

- Extremely fast
- Handles large tabular time-series
- Supports categorical features
- Excellent accuracy for demand forecasting

Key Parameters:

- num_leaves = 128
- learning_rate = 0.05
- objective = regression
- metric = RMSE
- early_stopping = 100 rounds

Evaluation:

Validation RMSE ≈ 216.0788047916286

Test RMSE ≈ 209.62396426002664



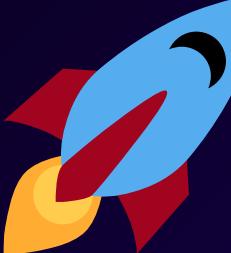
Results & Stock Visualizations

Generated Visuals:

- Train vs Validation error curve
- Prediction vs True values (test data)
- Category-wise stock status
- Store-wise stock summary (grouped bar chart)

Example Stock Classification:

Store	Product	Forecast	Est. Stock	Status
1	Snacks	220	150	UNDERSTOCK ●
2	Dairy	100	160	OVERSTOCK ●
3	Beverages	120	130	BALANCED ●



Conclusion & Future Enhancements

Conclusion

- Built an automated LightGBM-based forecasting pipeline
- Identifies overstock/understock conditions
- Provides decision-ready stock recommendations
- Helps reduce waste, improve sales, and maintain balanced inventory

Future Enhancements

- Real-time predictions (POS integration)
- AutoML & Neural Networks
- Deployment using FastAPI + React Dashboard





THANK YOU