
Final Project Report: Avocado Sales Volume Prediction

Executive Summary

This project aimed to accurately forecast avocado sales volume using machine learning models, enabling data-driven inventory and marketing decisions. After thorough data preprocessing and exploratory analysis, we compared two main models: XGBoost optimized by Particle Swarm Optimization (PSO) and an LSTM neural network. The LSTM model outperformed XGBoost, achieving an R^2 score of 0.9209 on the test set. The final model was deployed via a FastAPI RESTful service, providing real-time sales forecasts accessible through a web interface.

1. Introduction

Forecasting sales volume is critical for efficient supply chain management and customer satisfaction. This project focused on predicting

avocado sales volume based on historical sales data spanning several years and geographic regions.

2. Data Collection and Preprocessing

Dataset Overview:

The data consists of sales records with features such as:

- date: Date of the sales record.
- average_price: Average price of avocados on that date.
- total_volume: Total volume sold.
- Codes like 4046, 4225, 4770: Representing different avocado sizes/types.
- total_bags, small_bags, large_bags, xlarge_bags: Quantities of bags sold.
- type: Avocado type (conventional or organic).
- year: Sales year.
- geography: Region of sales.

Preprocessing Steps:

- Handling missing values using interpolation.
- Encoding categorical features (type, geography) using one-hot encoding.
- Feature engineering including extracting seasonal indicators from the date.
- Normalizing numerical data to ensure scale uniformity.

3. Exploratory Data Analysis (EDA)

Key insights identified:

- Seasonal peaks in sales during holidays.
- Variations in sales volume across regions and avocado types.
- Correlations between price and sales volume trends.

Visual aids such as time series plots and correlation heatmaps were used to visualize these trends.

4. Model Selection and Optimization

Two main models were tested:

- XGBoost, a powerful gradient boosting model, with hyperparameters optimized via Particle Swarm Optimization (PSO).
- LSTM (Long Short-Term Memory) neural network, effective for capturing sequential dependencies in time-series data.

PSO Optimization Details:

- Tuned parameters: number of estimators, max depth, learning rate, and subsample ratio.
- The optimization process iterated through multiple candidate solutions to find the best hyperparameters, which were saved and reused for efficient model training.

5. Experiment Tracking with MLflow

MLflow was used for:

- Logging hyperparameters and model versions.

- Tracking evaluation metrics (MSE, MAE, R^2).
 - Storing artifacts like trained models and evaluation plots.
- This ensured reproducibility and easy comparison of experiments.

6. Model Deployment with FastAPI

The best model (LSTM) was deployed as a REST API using FastAPI, hosted on Railway.

Users can input 30 past sales volume values to receive forecasts on either a linear or logarithmic scale.

API link: [Avocado Sales Forecasting API](#)

7. Model Performance Comparison

The table below summarizes the evaluation metrics on the test set:

| Model | MSE | MAE | R^2 |
|---------------|-----------------------|------------|--------|
| XGBoost (PSO) | 2.22×10^{11} | 198,852.97 | 0.8965 |
| LSTM | 1.69×10^{11} | 169,825.36 | 0.9209 |

Final evaluation of the LSTM model:

- MSE: 169,268,756,844.33
- MAE: 169,825.36
- R^2 : 0.9209

The LSTM model's superior R^2 score and lower error values demonstrate its effectiveness in capturing temporal dependencies in sales data.

8. Business Impact

Accurate forecasting enables:

- Better inventory planning to reduce stockouts and overstock.
- Optimized marketing strategies based on predicted demand fluctuations.
- Increased customer satisfaction due to availability of products.

9. Challenges and Solutions

- Data Imbalance: Different regions and types had varying data densities; addressed with

careful preprocessing and feature engineering.

- Hyperparameter Optimization: PSO provided efficient tuning without exhaustive grid search.
- Deployment: Integrating the model into a RESTful API with FastAPI ensured accessibility and scalability.

10. Tools and Technologies

- Python for data processing and modeling.
- Pandas for data manipulation.
- XGBoost for gradient boosting model.
- TensorFlow/Keras for LSTM implementation.
- MLflow for experiment tracking.
- FastAPI for API deployment.
- Railway for hosting the API.

11. Future Work

- Incorporate external economic indicators and weather data to improve forecasting.
 - Explore other deep learning architectures such as Transformers.
 - Enhance the API with authentication and a user-friendly frontend interface.
-