Machine Learning Product Performance Analysis - Report

Introduction

This project analyzes supermarket product sales using preprocessing, K-means clustering (implemented from scratch), and regression models to predict product profit. The report documents methods, results, and recommendations.

Data Preprocessing

Missing value handling: dropped rows missing `product_name` and median-imputed numeric columns. Outlier detection: IQR method used; outliers capped at bounds. Normalization: Min-Max scaling applied because K-means uses Euclidean distance sensitive to scale.

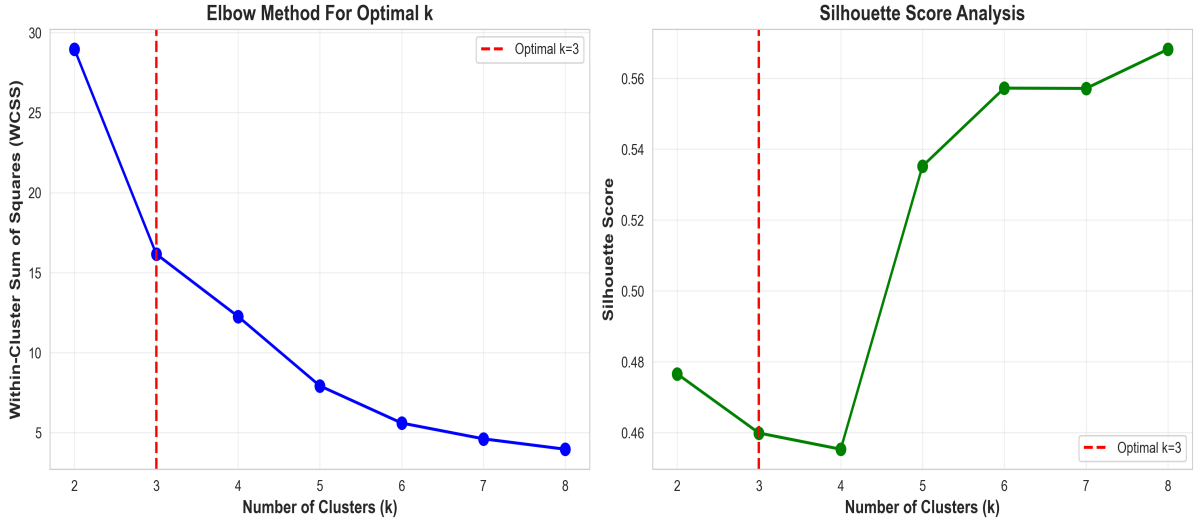
Statistic	price	cost	units_sold	promotion_frequency	profit
count	196.0	196.0	196.0	196.0	196.0
mean	0.423	0.401	0.381	0.282	0.454
std	0.267	0.279	0.253	0.321	0.225
min	0.0	0.0	0.0	0.0	0.0
25%	0.227	0.208	0.201	0.0	0.304
50%	0.327	0.299	0.335	0.2	0.442
75%	0.536	0.525	0.52	0.4	0.583
max	1.0	1.0	1.0	1.0	1.0

K-means Clustering Analysis

Implemented custom K-means with K-means++ initialization, iterative assignment and centroid updates, and vectorized computations. WCSS and silhouette scores were used to evaluate k.

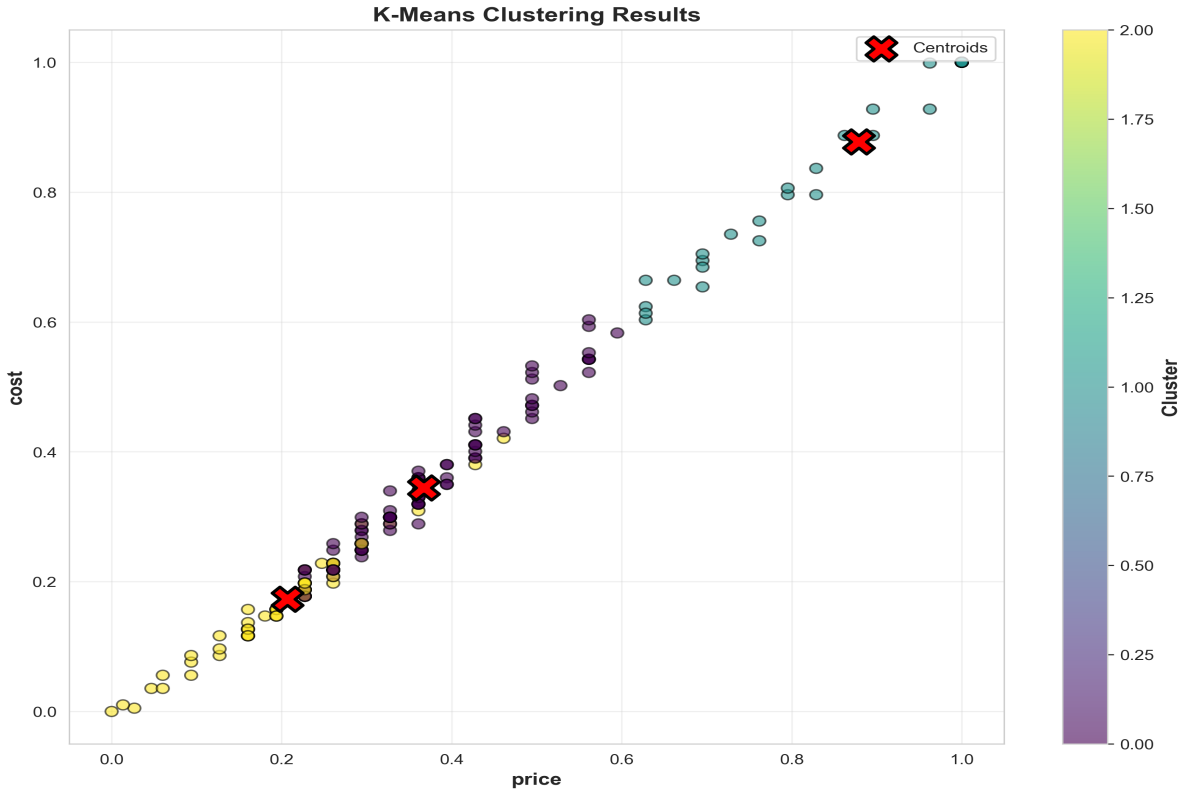
Elbow Method and Optimal k

WCSS values for k in [2, 3, 4, 5, 6, 7, 8]: [np.float64(28.96), np.float64(16.16), np.float64(12.26), np.float64(7.92), np.float64(5.59), np.float64(4.61), np.float64(3.96)]. Recommended k = 3. Silhouette at optimal k = 0.460.



Cluster Analysis, Interpretation and Naming

Cluster	Size	Avg Price	Avg Units	Avg Profit	Suggested Name	Insight
0	92	$6.08	226.7	$518.24	Mid-Range Steady	Mid-range products; monitor trends.
1	41	$13.74	78.4	$397.52	Premium Low-Volume	High price, low volume; consider targeted promotions.
2	63	$3.69	462.0	$708.26	Budget Best-Sellers	Low price, very high units sold; maintain stock and optimize margins.



Business Insights from Clustering

Examples: Budget Best-Sellers — keep stock and optimize margin; Premium Low-Volume — run targeted promotions; High-Value Populars — prioritize premium placement.

Regression Analysis

Models: Linear Regression (baseline) and Polynomial Regression (degree=2). Polynomial used to capture non-linear relationships.

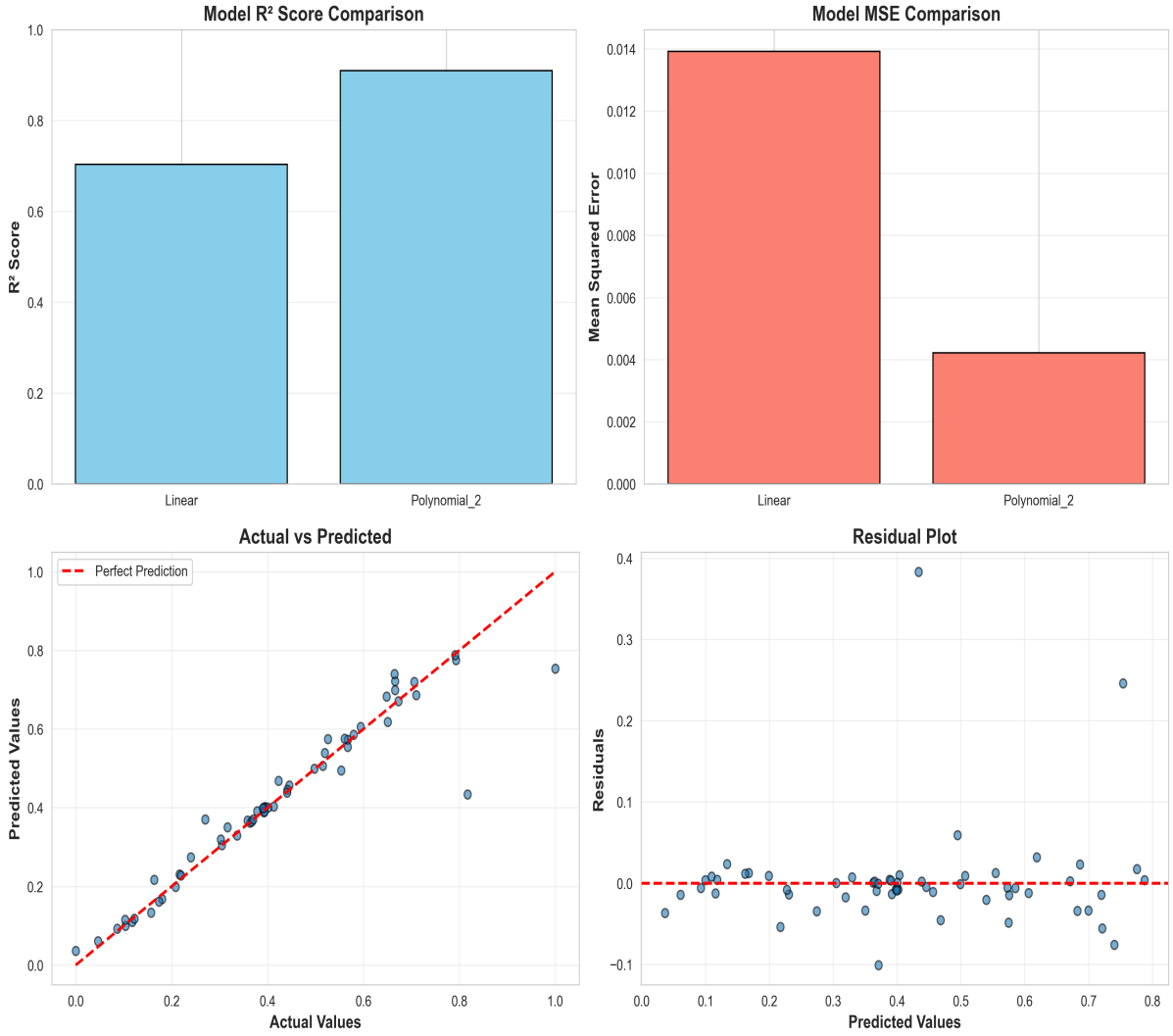
Training Process and Performance

Train/Test split: 137/59 (test_size=0.3). Metrics: MSE, MAE, R^2.

Model	Train MSE	Test MSE	Train MAE	Test MAE	Train R^2	Test R^2
Linear	0.0121	0.0139	0.0842	0.0820	0.7621	0.7031
Polynomial_2	0.0007	0.0042	0.0173	0.0281	0.9872	0.9100

Best Model Selection and Overfitting

Best model by test R^2: Polynomial_2 (test R^2=0.9100). Polynomial shows high complexity; check cross-validation and regularization to reduce overfitting.



Conclusion

Key findings: cleaned dataset (196 records); K=3 clusters with actionable segments; polynomial regression gives strong fit but risk of overfitting. Limitations: dataset size, feature set, potential overfitting. Improvements: CV, regularization, more data, feature engineering.

AI Tool Usage Summary

Generative AI assisted with code development, debugging, and report drafting. All final outputs were validated manually.