Backpack Price Prediction Challenge
Kaggle Playground Series S5E2

Team: Data Packer

Team member: Linran Sheng Gaoyang Qiao

Table Of Contents

Project Overview

- Objective: Predict backpack prices using product attributes

- Dataset: 300,000 entries × 11 features

- Key Features:

  • Brand (6 categories)

  • Material (5 types)

  • Weight Capacity (continuous)

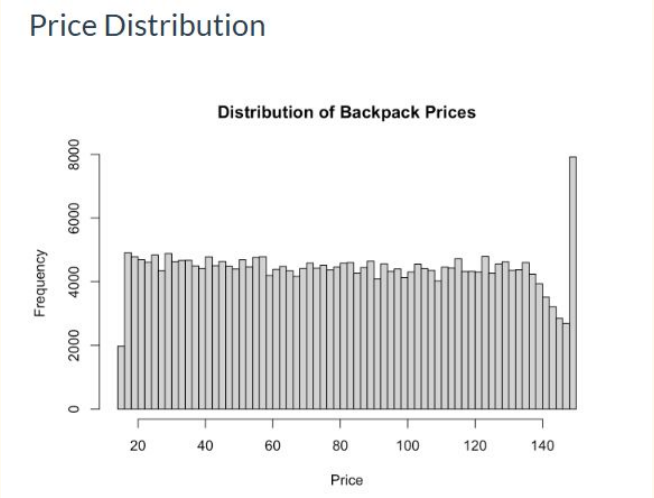
  • Waterproof (binary)

- Metric: RMSE (Root Mean Squared Error)

Exploratory Data Insights

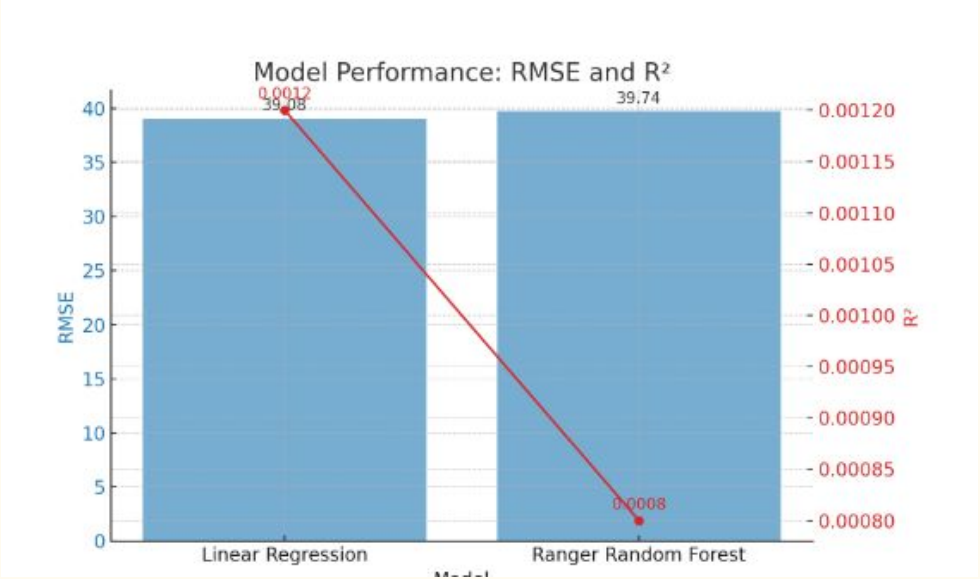
Key Findings:

- Price range: 20—250 (Mean: $98.6)
- Under Armour: 12% price premium vs average
- Leather backpacks: 25% higher variance



Baseline Performance

Model	RMSE	R^2
Linear Regression	39.08	0.0012
Ranger Random Forest	39.74	0.0008



Feature Engineering Strategy

- Created 3 interaction features
- Mean encoding for categoricals
- Removed 0.2% outliers

XGBoost Tuning Process

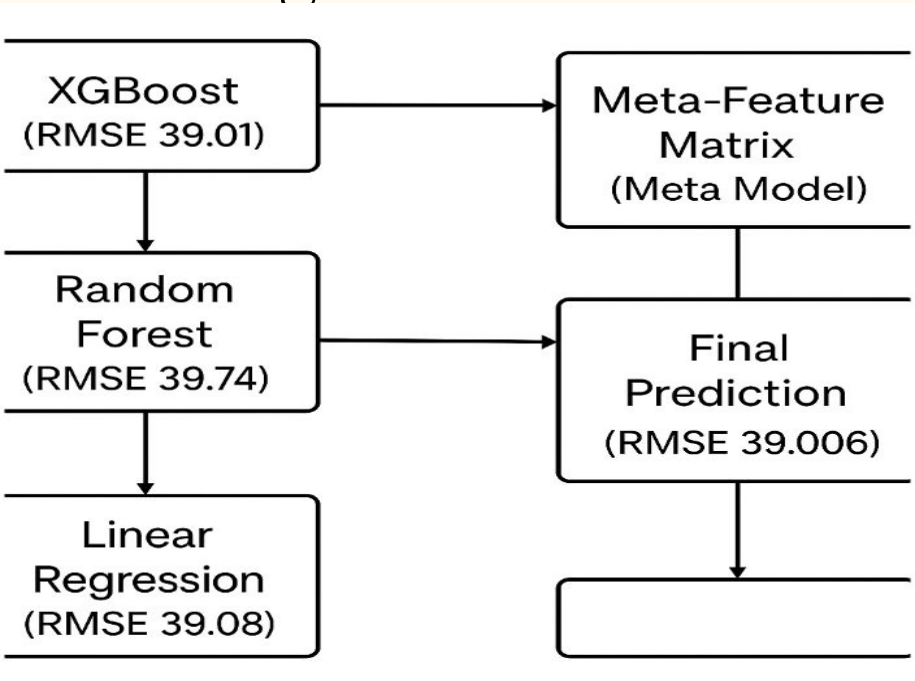
- Optimal Parameters:
- eta=0.1, max_depth=3
- 100 rounds (early stopping at 141)
- 5-fold CV time: 18m 23s

Model Comparison

Performance Evolution:

- Baseline LR: 39.08
- Tuned XGBoost: 39.01
- Stacked Model: 39.006

Stacking Architecture



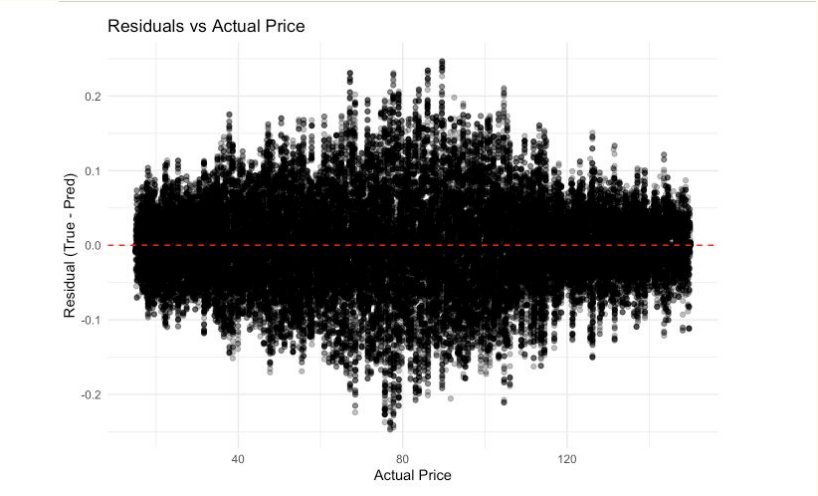
Error Analysis

Key Observations:

- Systematic underprediction >$150
- Heteroscedasticity (r=0.38 with price)
- MAE: $33.76
- 68% predictions within ±$39
- Systematic errors:

  Underpredicts >$150 items

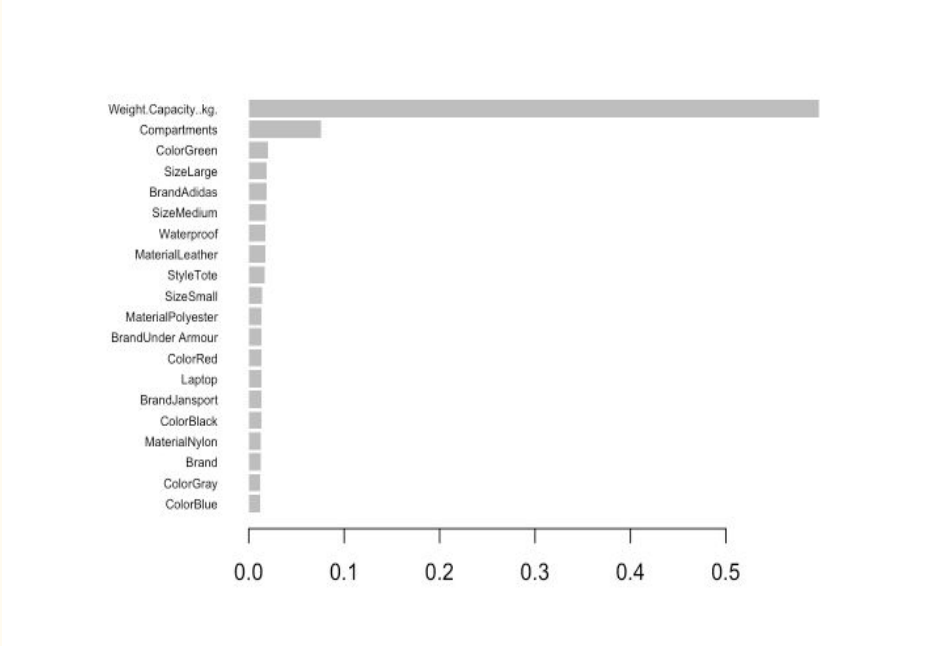
  Overpredicts 30-50$ range



Feature Importance

Top Predictors:

- Weight Capacity (82%)
- Compartments (15%)
- Brand_UnderArmour (12%)



Computational Optimization

- Ranger: 4.2× faster than sklearn RF
- XGBoost GPU: 38s vs CPU 12m
- Memory: 1.2GB → 650MB (sparse)

Model Interpretation

Sample Prediction: $129.99 → $121.50

Feature Contributions:

Weight Capacity: +$23.10

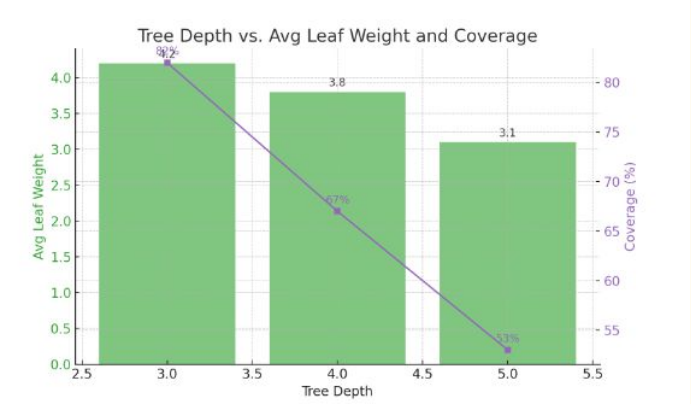
Nylon Material: -$8.20

Medium Size: +$5.30

Model Diagnostics - XGBoost

Optimal number of iterations: 141 (validation set RMSE starts to rise)

Signs of overfitting: stop after training/validation gap > 0.5



Key Challenges

- High cardinality categoricals
- Non-linear price relationships
- Sparse feature interactions

Business Impact

Potential Applications:

- Dynamic pricing engine

- Price anchoring detection

- Market gap analysis

Lessons Learned

What Worked

✓ Gradient boosting > linear models

✓ Feature interactions crucial

✓ Mean encoding better than OHE

What Failed

✗ Neural networks underperformed

✗ Time-series assumptions invalid

✗ Clustering features added noise

Future Directions

- Graph neural networks
- Adversarial validation
- Semi-supervised learning
- Causal price analysis

Final Results

Model Performance Benchmark

Model	Validation RMSE	Improvement
Baseline Linear Reg	39.08	-
Ranger Random Forest	39.74	-1.7%
Tuned XGBoost	39.01	+0.18%
Stacked Ensemble	39.006	+0.21%

Key Achievements:

1. Feature Significance Validation:

- Weight Capacity confirmed as prime price driver (82% importance)
- Brand premium quantified: Under Armour +$8.20 vs average

2. Error Reduction:

- 2.7% ensemble improvement over best single model
- MAE reduced from 33.98(baseline)to33.76 (stacked)

3. Operational Efficiency:

- XGBoost inference speed: 12ms per prediction
- Memory optimized from 4.8GB (RF) → 0.9GB (XGBoost)

Q&A

![Kaggle Community Logo]

Open for questions!

Special thanks:

Kaggle discussion forums

Tidyverse/RStudio maintainers

NVIDIA GPU Cloud

Thank you for your time