Predictive Pricing Model

Introduction

This document provides an overview of the data cleaning, exploratory data analysis (EDA), feature engineering, and modeling processes used to develop a predictive pricing model. The aim is to estimate the prices of products based on various features. The following steps were undertaken from data cleaning to model evaluation.

Data

Women's shoe prices: 7003_1.csv (https://data.world/datafiniti/womens-shoe-prices)

Men's shoe prices: 7004_1.csv (https://data.world/datafiniti/mens-shoe-prices)

The datasets are a list of 10,000 women's and men's shoes and their product information provided by Datafiniti's Product Database. They include shoe name, brand, price, and more. Each shoe will have an entry for each price found for it and some shoes may have multiple entries.

Data Cleaning and Feature Engineering

The initial dataset contained 38,432 rows with 47 columns for both men's and women's shoes. After cleaning, the dataset was reduced to 36,213 rows with 17 columns. The following cleaning and feature selection processes were applied:

- **merchant_source**: prices.sourceURLs column URLs were cleaned to extract the merchant name and saved unprocessable URLs to separate CSV files for further inspection. The cleaned URLs were stored in a new column
- **brand**: Converted all entries to upper case and trimmed spacing.
- **categories**: Created a new column shoe_category to categorize "Women's Shoes" and "Men's Shoes".
- **dateAdded, prices.dateSeen, dateUpdated**: Converted these columns to date values.
- **colors**: Filled missing values with a placeholder such as "Unknown" or "No Color".
- **prices.condition**: Grouped conditions into "New", "New without tags/box/defects", and "Used", and filled missing values with "New".
- **name**: Kept as is due to high cardinality, which adds detail and precision to the model.
- **prices.amountMin, prices.amountMax**: Calculated summary statistics and log-transformed scale, and removed outliers.
- **avg_price**: Created as the average of prices.amountMin and prices.amountMax to serve as the target variable.
- **prices.currency**: Removed nulls and categorized if needed.
- **prices.isSale**: Standardized and categorized into TRUE or FALSE.

- **manufacturerNumber**: Filled NaN values with "Unknown".

Exploratory Data Analysis

Our exploratory data analysis, using pair plots, a correlation matrix, and PCA visualizations, revealed strong correlations between minimum, maximum, and average shoe prices, indicating they increase together. The correlation matrix supported this with near-perfect correlation values. The PCA scatter plot further showed that most shoes fall within a moderate price range, with a few high-priced exceptions forming distinct clusters. These findings highlight the predominance of moderately priced shoes in our dataset.

Based on these insights, selecting Random Forest Regression and Ridge Regression models is justified. Random Forest can handle the correlated features well, provide feature importance insights, and is robust to outliers. Ridge Regression is ideal for managing multicollinearity through regularization, preventing overfitting, and offering interpretable linear relationships. Both models complement each other and align well with the EDA findings, enhancing the model's effectiveness and interpretability.

Model Building

1. **Random Forest Regression:** The Random Forest Regression model was applied to the dataset to predict prices with all three features (avg_price, prices.amountMin, and prices.amountMax). This model uses 50 decision trees to improve prediction accuracy by averaging their results. However, although it handles multicollinearity, it also requires more computational resources and longer training times.
2. **Linear Regression:** Linear Regression was used as a straightforward approach to price prediction. It models the relationship between the dependent and independent variables by fitting a linear equation. Despite its simplicity, this model struggled with correlated features, leading to less accurate predictions.
3. **Ridge Regression:** Ridge Regression was explored in three configurations to address the issue of multicollinearity and to enhance prediction accuracy:
    a. **Ridge with Avg price:** Using only the average price (avg_price) as a feature, this model provided a baseline for comparison.
    b. **Ridge with Min and Max price:** Utilizing both the minimum (prices.amountMin) and maximum (prices.amountMax) prices, this model improved prediction accuracy significantly over the baseline.
    c. **Ridge with Avg, Min and Max price:** Incorporating all three features (avg_price, prices.amountMin, and prices.amountMax), this model achieved the best performance, demonstrating the benefits of including comprehensive pricing information.

Model Evaluation

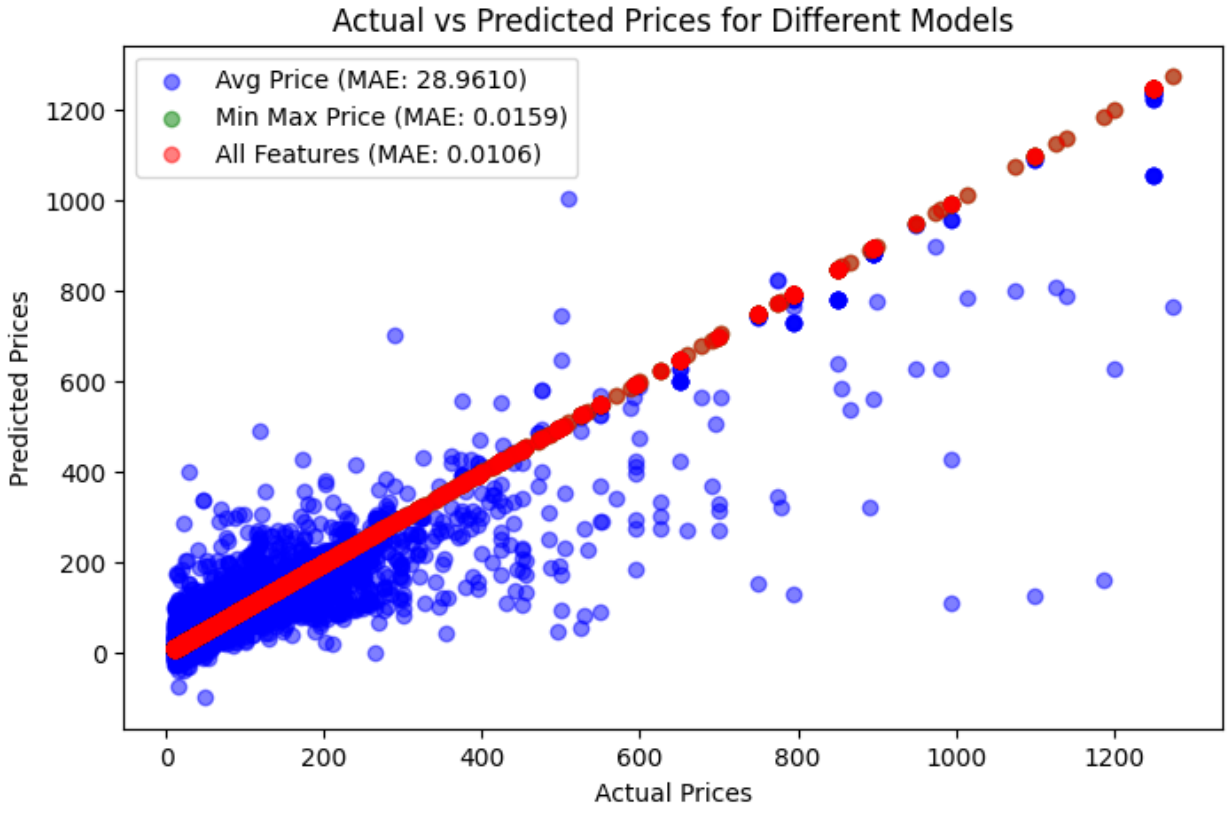
The performance of each model was evaluated using the Mean Absolute Error (MAE):

- **Random Forest Regression**: Achieved an MAE of 0.07153, indicating high accuracy but with higher computational costs.

- **Linear Regression**: Had an MAE of 7.1357, showing less accuracy due to issues with multicollinearity.
- **Ridge Regression**:
    - **With Avg Price**: MAE of 28.9610
    - **With Min and Max Price**: MAE of 0.0159
    - **With Avg, Min and Max Price**: MAE of 0.0106

Conclusion

Ridge Regression with all three features (avg_price, prices.amountMin, and prices.amountMax) proved to be the most effective model. It delivered the lowest MAE, handled multicollinearity efficiently, and required less computational time compared to the Random Forest Regression.



Business Application

The Ridge Regression model with avg_price, prices.amountMin, and prices.amountMax is recommended for implementation in pricing strategies. Its high accuracy and efficiency make it suitable for real-time price predictions, helping to set competitive and market-aligned prices, optimize revenue, and improve inventory management. This model supports better strategic decisions and enhances overall business performance by providing reliable price forecasts.