PREDICTING CONSUMER PRODUCTS' PRICES AND PERFORMANCE

Damilola G. Efuwape

Price and demand are generally believed to move in opposite directions. Changes in price usually cause a movement along the demand curve, but not a complete shift. Understanding how a potential price change could affect product sales and vice-versa is very important in developing good strategies to improve sales and increase profit. In this project, I built some machine learning models that will help predict future prices and quantity sales of products based on price changes, using patterns of known changes in prices and quantity demanded.

Problem Statement

At what prices will consumer products be sold and what is the likely performance of such products after anticipated price changes?

Audience

This project was carried out to help manufacturers and management of retail stores track, analyze, assess and determine stock prices of products, as well as, make likely predictions of products' sales based on anticipated price changes.

Data

The data that was used for this project is Marian Svatco's CSV-formatted " Retail Store Sales Transactions (Scanner Data)" from Kaggle. According to the website, this dataset shows details of consumer goods' sales obtained by scanning the bar codes of individual products at electronic points of sale in a retail store. It contains eight features with detailed information about products that were sold. These information include their quantities and prices, amongst other things. There are one hundred and thirty-one thousand, seven hundred and six rows in the dataset. The link to the original dataset is:
https://www.kaggle.com/marian447/retail-store-sales-transactions

Data Wrangling

After importing the necessary packages, I changed the "Unnamed:0" column to "Row_Id". Then, I checked if there are any missing values and what percentage of each column is missing. If there are missing values, I wanted to order them in increasing order and then present them in a single table. Fortunately, there are no missing values in the dataset. Next, I checked for outliers and discovered there are just three of them in the dataset. These outliers were replaced by the median values of their respective columns. The dataset contains some features that are not relevant to my desired predictions, so, I reduced the features to just four, namely: Date, SKU, Quantity and Sales_Amount. SKU (product) and Quantity became my target feature for predicting Sales Price, while Sales_Amount and SKU were used to predict Quantity Sales.

Exploratory Data Analysis

In this section, I checked for duplicate values, but found none. Then, I did sales summaries by date for Sales_Amount, Quantity and SKU. This was done to determine the total price, quantity sales and SKU sales for each day. The values of Quantity and SKU are different because SKU signifies the unique items sold, while Quantity is the total count of items sold, including items that are sold more than once.
Any of the dependent variables (“Sales_Amount" for price prediction and “Quantity” for quantity prediction) or independent variables (“Sales_Amount”/“Quantity” and “SKU”) that was not already an integer type was converted thus because linear regression, which is among the models used in this project, is supported only on integer-type variables. I also checked for the product categories with the highest sale, as well as, the products with the highest sale and the highest patronizing customers. The top five categories are N8U, R6E, LPF, P42, U5F with 10913, 5099, 5062, 4836 and 4570 sales respectively. The top five products sold are UNJKW (2007), COWU2 (791), OV1P9 (737), M6J9W (698) and C6TXL (689). The top five patronizing customers are customers with IDs: 1660, 1665, 17104, 1685 and 16905 with 228, 222, 218, 191 and 179 patronages, respectively. Finally, some data visualizations were done to see the relationships among the variables and data profiling was also done to get comprehensive information about the dataset and check for further defects. Different types of charts were plotted to basically check the same thing: Variable Relationships!

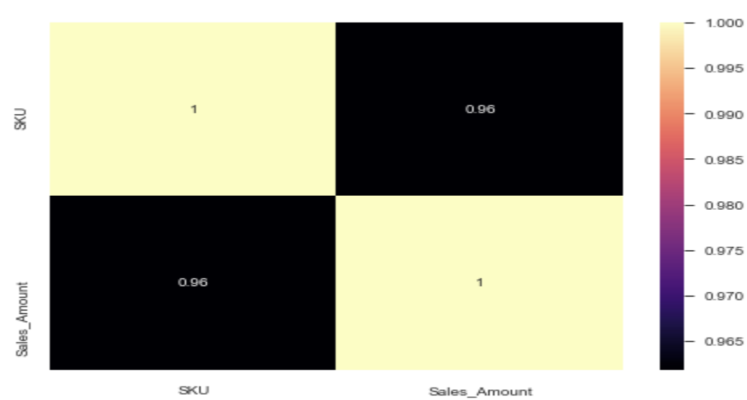

Fig. 1

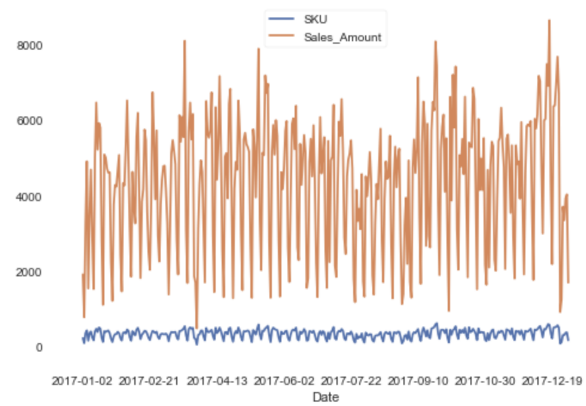

Fig. 2: SKU & Price Distribution

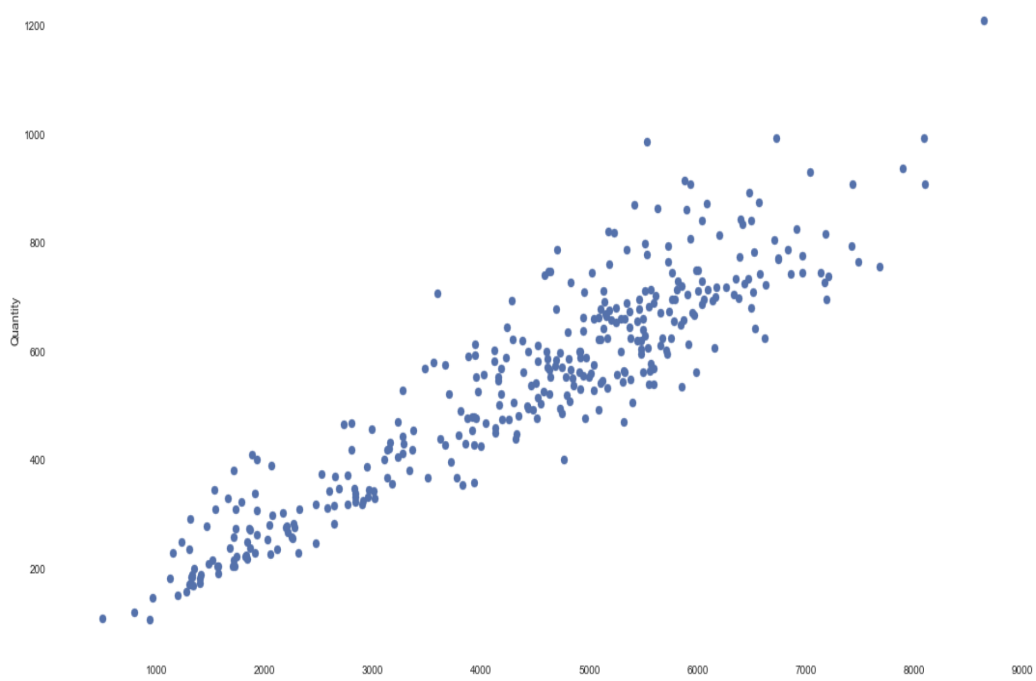

Fig. 3: Quantity Distribution1

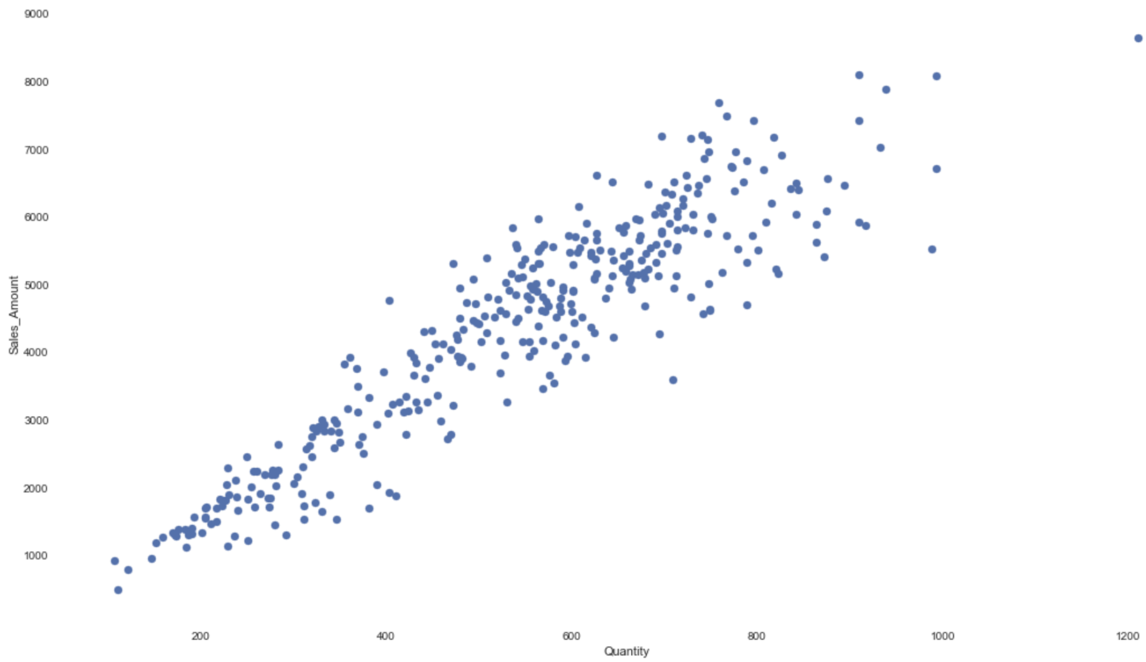

Fig. 4: Sales Amount Distribution1

Fig. 5: Sales Amount Distribution2

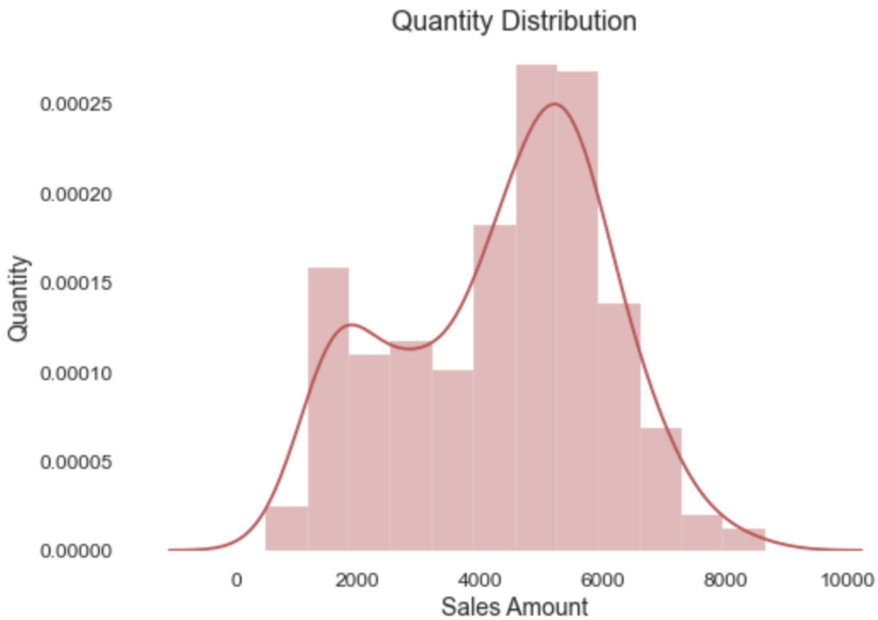

Fig. 6: Quantity Distribution2

Figure 1 shows the relationship that exists among the four variables.
Figure 2 shows there is no particular pattern in the number of unique products sold or their prices.
Figures 3 and 4 show that most times, there is a positive correlation between price and quantities of products sold.
Figures 5 and 6 are histograms that show an irregular pattern in price and quantities sold.

Modeling

To predict sale prices and quantities of products (SKU), five Linear Regression algorithms and a Random Forest model were used through the scikit-learn package. These algorithms are Ordinary Least Square (OLS), Bayesian, Ridge, ElasticNet and Lasso.
To evaluate the models, the “variance_score” and the r2_score” metric functions from the scikit-learn package in python were used. For the model to be considered effective, the variance score must be between 0.60 (60%) and 1 (100%), and the R-Squared score, which is the most popular evaluation metric for regression models, and is a measurement of how well the dependent variable (in this case, “Sales_Amount” for price prediction and “Quantity” for quantity prediction) explains the variance of the independent variable (“Sales_Amount”/“Quantity” and SKU), must be greater than 0.60 (60%). In fact, it should be more than 0.70.
After deployment, every model had a variance and R-Squared scores of either approximately 0.92 (92%) or 0.94 (94%) for predicting sales price of products, and Variance and R-Squared scores of 0.87 (87%) for predicting quantity of products that will be sold based on product prices. This suggests that the models perform well on the dataset.

Price Prediction

Comparing all the evaluation metrics from the models, the Bayesian regression algorithm is the most suitable model for predicting product prices on the basis of both Variance and R-Squared scores.

Bayesian Model Prediction:

	Actual Price	Predicted Price
0	2321	2774.001849
1	4810	4453.223470
2	4186	4574.860794
3	4683	4959.492662
4	3274	3469.214090
...	...	...
68	6965	6737.025019
69	5307	4664.989723
70	1837	1855.342174
71	4194	4108.317574
72	5530	6073.967646

Fig. 7: Actual Vs Predicted Prices

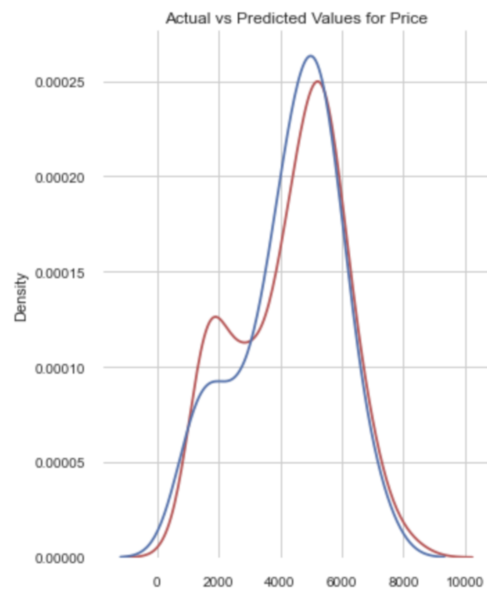

Fig. 8: Distribution of Actual Vs Predicted Prices

NB: Actual price is in red and the predicted is blue.

The second best performing model on the basis of both metrics is the ElasticNet model.

ElasticNet Model Prediction:

	Actual Price	Predicted Price
0	2321	2775.128331
1	4810	4451.506462
2	4186	4574.742749
3	4683	4961.383247
4	3274	3469.674994
...	...	...
68	6965	6741.544740
69	5307	4666.514021
70	1837	1855.160944
71	4194	4109.716476
72	5530	6073.071957

Fig. 9: Actual Vs Predicted Prices

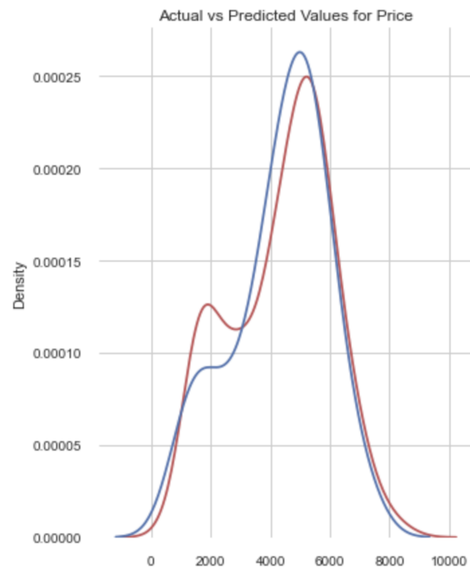

Fig. 10: Distribution of Actual Vs Predicted Prices

The worst performing model on both evaluation metrics for price prediction, however, is the Random Forest model.

Quantity Prediction

In contrast, the Random Forest model is the best performing model for sales quantity prediction on the basis of both Variance and R-Squared scores.

Random Forest Model Prediction:

	Actual Quantity	Predicted Quantity
0	310	341.772
1	588	535.530
2	569	565.993
3	575	569.007
4	414	435.188
...	...	...
68	747	812.701
69	545	587.913
70	220	231.626
71	476	522.831
72	779	711.406

Fig. 11: Actual Vs Predicted Prices

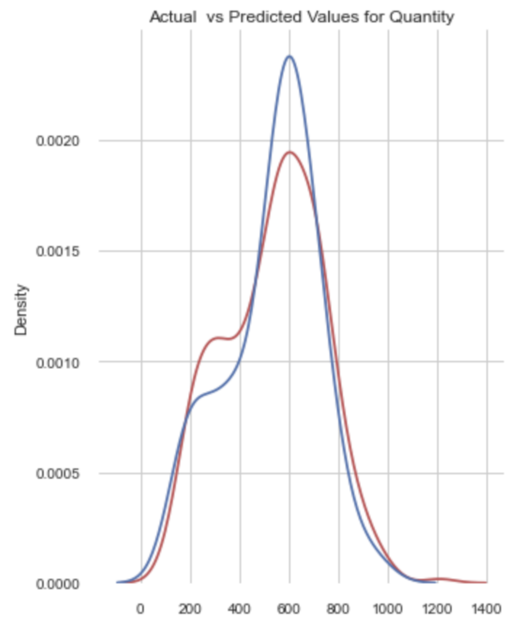

Fig. 12: Distribution of Actual Vs Predicted Prices

The second best performing model on both metrics is the Bayesian model.

Bayesian Model Prediction:

	Actual Quantity	Predicted Quantity
0	310	346.983849
1	588	556.877119
2	569	547.800612
3	575	601.929785
4	414	432.660742
...	...	...
68	747	829.505296
69	545	602.147665
70	220	255.694508
71	476	518.203667
72	779	707.980708

Fig. 13: Actual Vs Predicted Prices

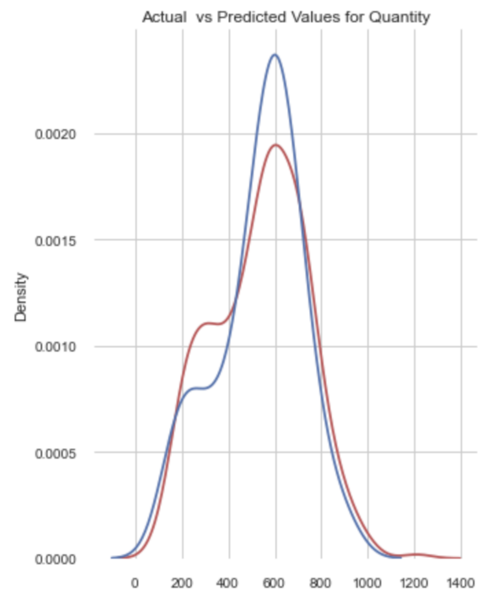

Fig. 14: Distribution of Actual Vs Predicted Prices

The worst performing model on both bases is the Ordinary Least Square (OLS) model.

Conclusion

After evaluation, it can be concluded that the Bayesian or ElasticNet regression models should be used for price prediction and the Random Forest or Bayesian models should be used to predict sales quantity in this case.
These models have the tendency to make near-perfect predictions, which would help decision makers make the best possible business decisions for maximum profitability.

Future Improvements

Other powerful models like the Neural Networks, Time Series, along with models like the Boosted decision tree model, the Poisson regression model, etc, should be used to make these predictions and their performances compared to the ones of the models used in this project.
Also, I believe limited data information restricted the flow of this project and other things that could have been predicted. Actual products and category names should be provided in the data to help in predicting how change in prices of substitute products can affect sales of certain goods. This will also help to adequately predict customers' purchase patterns.