An efficient novel approach to E-commerce retail price optimization through machine learning

Yerragudipadu Subbarayudu[1*], *G* Vijendar Reddy[2], *M* Vamsi Krishna Raj[3], *K* Uday[4], *MD* Fasiuddin[5], and *P* Vishal[6]
[1]Department of Information Technology, Gokaraju Rangaraju Institute of Engineering and Technology, Bachupally, Hyderabad, Telangana, India

Abstract. Businesses can use price optimization to discover the most profitable price point by using customer and market data to drive their decisions. The optimal price points will result in the company making the most money possible, but they may also be created to help the company expand into untapped markets or increase its market share, for example Businesses can use machine learning to price products and services to maximise sales or profitability by using data instead of educated guesswork. When utilised for price optimization, ML-based algorithms can be used to forecast demand for a particular product as well as the ideal price and how buyers will respond to specific pricing. Pricing decisions can be made more accurately using machine learning, which will boost a company's revenue.

1 Introduction

A common issue in the management of numerous product and service businesses is price optimization. Economics has long established that a product's price has an impact on its demand. A reduced price for a consumer good typically results in greater demand for the product. But when numerous products are available, the situation is frequently more complicated because of the relationships between the different products. We frequently employ demand functions to quantify how product demands are influenced by their prices. [1].

* Corresponding mail id: subbu.griet@gmail.com

Customer demand is influenced by the item's life expectancy as well as its cost. Because cost has a significant impact on demand, companies like Ford and Dell Computer use dynamic pricing in addition to their distribution and production strategies to increase profitability [2,3]. When a new item enters the market, buyers want to buy it because of its features, because interest in items like technology and fashion design fades after a while [4]. Despite this, interest in the previous model continues to dwindle as innovation advances and new aspects are added to it [5]. Regardless of its transaction worth, an item's interest is influenced by its deal volume and general attention. A powerful strategy is to consider the dynamic cost function of demand, where the demand varies in response to client demand. These characteristics served as the impetus for the study, which found that a dynamic pricing technique is more effective at raising revenue than adopting a fixed sales price [6]. Even if they are making losses, companies that follow this price-cultivation strategy do not decrease their product pricing to boost sales because they view such losses as investments in price development. Price cultivation aims to develop client assessments of products and attach a reference price in their thoughts. Prior to making speculations about adopting the cost development practise, organisations must consider the constant refreshing of reference prices for such exquisite items in the minds of customers, as well as provider rivalry (contest among brands) and retailer rivalry (contest within brands) in the item market [7-9] [10]. Price elasticity of demand is a notion that gauges how much demand will fluctuate in response to a change in a product's price. This idea can be applied to price determination depending on demand. Regression to estimate the demand for a product based on historical sales data and use the machine learning model to do so, machine learning models can be employed for this purpose. To establish the best price, the price is calculated using the predictions and price elasticity. This aids companies in setting product prices as profitably as possible while also giving them a competitive edge in a crowded market.

2 Research Review

The following definition is a broad one for the dynamic pricing issue: Adaptively change prices over time to maximise predicted profits when faced with a fixed set of products to offer and a predetermined sales outlook. It is necessary to consider factors such as uncertain client demand, constant competition, shifting markets, and remaining inventory levels. Since cost is an important marketing factor and has a big effect on revenue and income, dynamic pricing is important in revenue management, particularly in e-commerce [11]. A significant outstanding issue in revenue management is stochastic dynamic pricing when there is competition and insufficient demand information. Although the problem has great practical implications, it seems to be inherently difficult. It is difficult to I forecast sales probabilities using private market data that can be observed, and (ii) develop methods that enable automated price reactions that are optimum and take the shortest amount of time to compute [12]. Because of this, managers are forced to restrict the range of pricing options, for example by employing deterministic or highly stylized demand models [13], monopolistic situations, or by pricing less frequently. The quality of strategies is negatively impacted by this simplicity. The basic problem still exists in the size of the solution space, necessitating a workaround for the dimensionality curse. We develop price reaction techniques for the Amazon Market-place in this research. We want to address the following presumptions: restricted demand information, strategies of unknown competitors, and market data that is only partially observable [14].

3 Problem Statement

Optimizing a price of a product is important for good sales of a product. Products need to be priced well to have an advantage in the market with competition. Earlier, retail price optimization was done by managers based on their knowledge, this can be improved by using historical sales data to gain insights from it and make a better decision. With the help of machine learning, models can be taught on data and then used to successfully determine an ideal price for a product. By taking into account the various factors that influence decisions, a machine learning model trained with all that data can identify patterns and provide predictions that can be used for optimally pricing a model. Right pricing is important as it has a huge impact on the product sales

4 Proposed Architecture

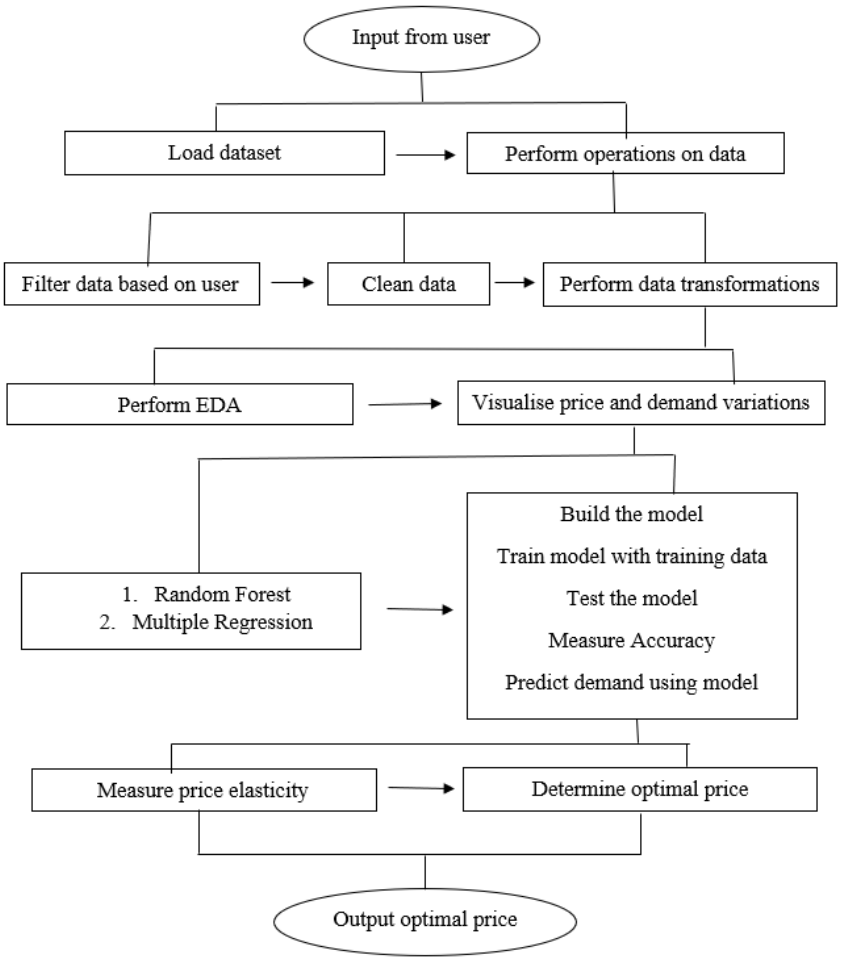

Fig 1: System Architecture

Fig (1), The architecture describes the working of the project, where the input is taken from user and the data set is loaded and pre-processing techniques are performed on the da-

taset. Then the data is investigated through EDA filtered, cleaned and transformed. The data is used to train, test and measure accuracy of the algorithms. Finally, demand is predicted using model. The price elasticity and optimal price are determined to get the output optimal price.

5 Dataset

The dataset is taken from Kaggle [15]. It contains data about retail products including product price, product rating score, freight price, category, quantity sold, etc. The dataset contains data about the competitors, i.e., their product price, freight price and product rating score, and various products and their categories, information about the product like product description. The data contains attributes of various types: categorical attributes, numerical attributes, nominal attributes, etc. Different features from the dataset that can have an effect on the price of a product to build a model are taken into account and is used for training and testing a machine learning model to optimize the price of a product.

6 Preprocessing

Pre-processing of data includes steps like data cleaning, data transformation, data reduction etc. The data used is pre-processed before it is used for training the machine learning model. Missing values, if any, are removed from the data. The data contains categorical and nominal attributes and are transformed into numerical attributes using label encoding. The label encoder transforms the string categorical attributes into numerical attributes.

7 Methods

7.1 Random Forest Regression model

A Random Forest regression model is an algorithm that fits numerous grouping decision trees on various sub-sets of the data and using averaging to enhance prediction accuracy and prevent over-fitting. If bootstrap=True (the default), the sub-set size is restricted by the max_samples boundary; otherwise, the whole dataset is used to construct each tree. [16]. Random Forest is an ensemble technique which is capable of performing tasks of both regression and classification using multiple decision trees and a technique called Bootstrap and Aggregation, commonly known as bagging. The main idea here is to combine many decision trees to determine the final outcome rather than relying on individual decision trees. Random Forest has more than one decision tree as base learning model [17].

Decision Trees are used to deal with both regression and classification concerns. They stream outwards like trees, and in the regression situation, they start at the base of the tree and follow splits in view of variable results until reaching a leaf node and finding the result. Underneath is an illustration of a decision tree:

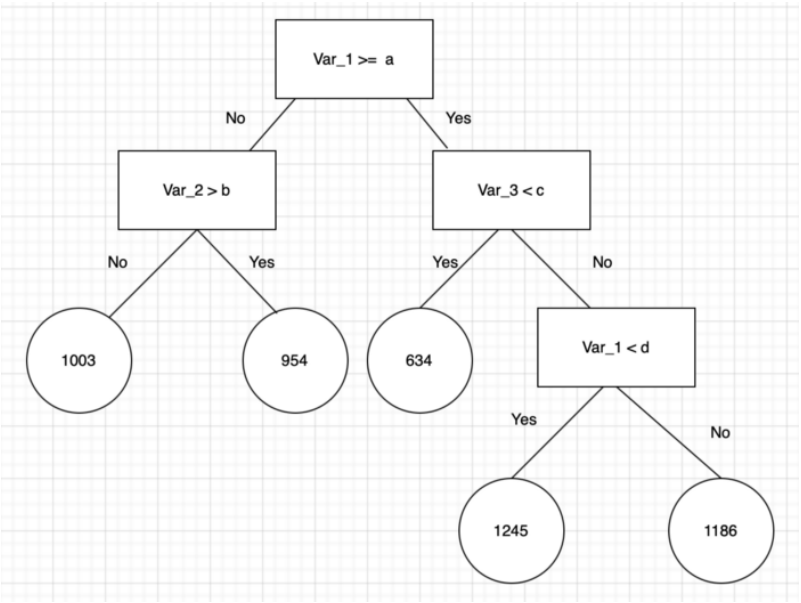

Fig 2. Decision tree

An essential decision tree chart is shown above, which starts with the Var_1 and separates depending on specified rules. When 'yes,' the decision tree goes down the addressed path; when 'no,' the decision tree goes down the opposite path. This cycle is repeated till the decision tree reaches the leaf node and the subsequent outcome is selected. In the preceding model, the advantages could be representative of any numerical or categorical value.

Ensemble learning is the process of using multiple models, trained on comparable data, and averaging the outcomes of each model to provide a more spectacular predictive/classification result. Our expectation, and the necessity, is that the mistakes of each model (for this situation decision tree) are autonomous and unique from tree to tree.

Bootstrapping is the process of randomly evaluating different dataset subsets across a certain number of cycles and parameters. The mean value of these results is then calculated for all of them to produce an even more impressive result. A practical application of the ensemble model is bootstrapping.

Bootstrapping algorithm combines ensemble learning techniques with decision tree system to generate many randomly chosen decision trees from the data, averaging the results to produce another conclusion that frequently leads to reliable predictions.

We will use the Sklearn module, specifically the RandomForestRegressor feature, to create a random forest regression model. The documentation for the RandomForestRegressor lists different options for model parameters. Below are some of the important parameters:

- n_estimators — number of decision trees used in the model.
- measure — this option allows you to select the rule (misfortune capability) that will be utilised to determine model results. Error methods like mean squared error (MSE) and mean absolute blunder (MAE) can be used, MSE is the default value.
- max_depth — this specifies each tree's maximum potential depth.

- max_features — maximum number of features that the model will evaluate when deciding on a split.

- bootstrap — default value is True, which means the model adheres to bootstrapping standards (described previously).
- max_samples — If bootstrapping is enabled, this boundary is ignored; it has no effect otherwise. In case of True, this value determines the maximum size of each example for each tree.
- Other important parameters are min_samples_split, min_samples_leaf, n_jobs

rf = RandomForestRegressor (n_estimators = 300, max_features = 'sqrt', max_depth = 5, random_state = 18).fit(x_train, y_train)[18].

7.2 Multiple Regression Model

Multiple Regression is an algorithm which models the connection between one dependent variable and more than one independent variable. Several conditions that impact the reliant variable can be controlled through multiple regression analysis. Regression analysis is a technique for examining the connection between independent factors and dependent factors.

Let k to address the quantity of factors meant by x1, x2, .., xk. For this method, we expect that there are k autonomous factors x1, x2, ..,, xk that we can set, then, they probabilistically decide a result Y. Moreover, we expect that Y is nearly dependent on the variables as indicated by

$$Y = \beta 0 + \beta 1x1 + \beta 2x2 + \cdots + \beta kxk + \varepsilon \quad (1)$$

- The variable yi is the predicted or dependent
- The slope of y depends on the y-intercept, i.e. y will be β0, when xi and x2 are both zero
- β1 and β2, the regression coefficients represent change in y as a result of one-unit changes in xi1 and xi2.
- βp is all independent variables' slope coeffiecient
- ε term represents the model's random error (residual).

where ε is standard error, this is the same as simple linear regression, except k doesn't have to be 1.

There are n observations, n generally greater than k. For observation i, we set the independent variables to the values xi1, xi2, xi3 ..., xik and measure a value yi for the random variable Yi.

Hence, the equation of the model is:

$$Yi = \beta 0 + \beta 1xi1 + \beta 2xi2 + \beta 3xi3 + \ldots + \beta kxik + i, for\ i = 1, 2, \ldots, n \quad (2)$$

where the errors i are independent standard variables and with mean zero and the same unknown variance of σ2 for each.

Altogether the multiple linear regression model has k + 2 unknown parameters:

$$\beta 0, \beta 1, \beta 2, \ldots, \beta k, and\ \sigma 2 \quad (3)$$

when k = 1, we found least squares line equation to be y = βˆ 0 +βˆ 1x and it was a line in the plane R2.

Now, with k greater than or equal to 1, the least squares hyperplane is,

$$y = \beta\hat{}\,0 + \beta\hat{}\,1x1 + \beta\hat{}\,2x2 + \cdots + \beta\hat{}\,kxk \text{ in } Rk+1 \quad (4)$$

The method to find the estimators βˆ 0, βˆ 1, βˆ 2, … , and βˆ k is the same. Take the partial derivatives of the squared error.

After that is solved, the fitted values will be

$$y\hat{}\,i = \beta\hat{}\,0 + \beta\hat{}\,1xi1 + \cdots + \beta\hat{}\,kxik \quad (5)$$

for i = 1, . . ., n that should be close to the actual values yi[19].

8 Experiment Results and Discussion

Results for example of garden tools category products
Price elasticity is -1.2855714285714284

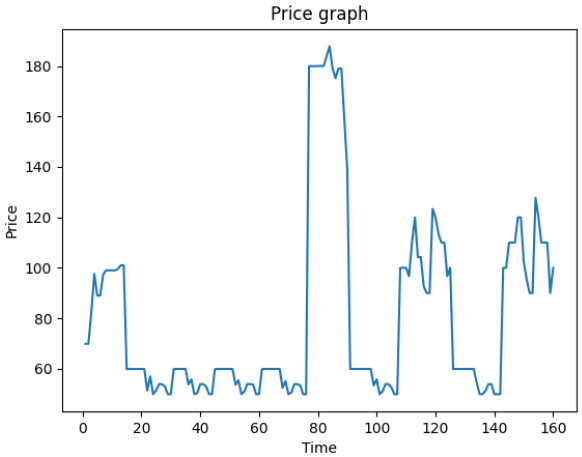

Fig.3 Example 1 demand graph

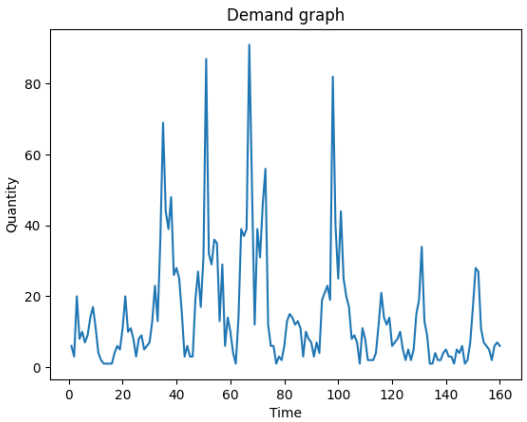

Fig.4 Example 2 price graph

Fig (3,4): The price graph shows price over time, whereas the quantity graph shows demand over time.

Table 1. Example-1 scores

Model	R-squared score	Mean squared error (MSE)	Max error	Predicted demand	Optimized price
Random Forest	98.51%	1.95%	5.82%	7	84.07
Multiple Linear Regression	87.57%	16.28%	16.94%	4	125.43

For an example garden_ tools of these are the graphs, accuracy, MSE, max error rate, prediction and optimized price obtained from the models. Demand and price graphs are plotted and model is trained and tested and used to make prediction of demand. Price elasticity is calculated and along with the predicted demand, price is optimized.

Results for example of computers_ accessories category products
Price elasticity is 56.35

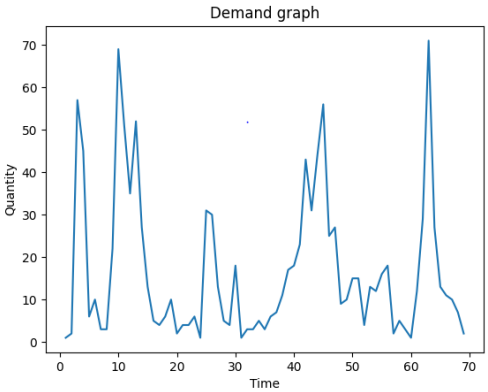

Fig.5 Example 2 demand graph

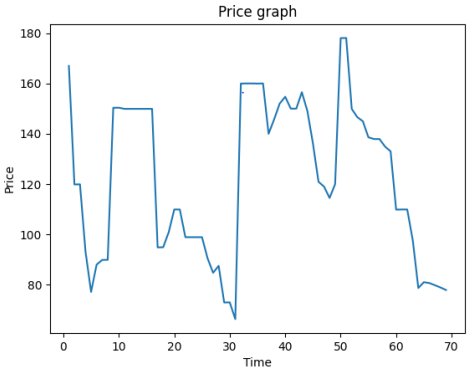

Fig.6 Example 2 price graph

Fig (5,6): The quantity over time shows demand graph, whereas price over time shows price graph.

Table 2. Example-2 scores

Model	R-squared score	Mean squared error (MSE)	Max error	Predicted demand	Optimized price
Random Forest	92.25%	32.23%	17.89%	2	77.83
Multiple Linear Regression	91.65	34.73%	14.76%	8	81.86

For an example of computer_ accessories these are the graphs, accuracy, MSE, max error rate, prediction and optimized price obtained from the models. Demand and price graphs are plotted and model is trained and tested and used to make prediction of demand. Price elasticity is calculated and along with the predicted demand, price is optimized.

9 Conclusion and future scope

Machine learning can be used to perform retail price optimization and improve the business of a company. Using a machine learning model can provide a lot of benefits like being able to process large amounts of historical sales' data and to find hidden patterns in the data which might not have been found by people. Random forest regression model and multiple linear regression model were used for this purpose. Random forest model has higher R-squared score than multiple regression model. Random forest model has lower Mean Squared Error than multiple regression model

Random forest model has lower Max Error than multiple regression model. Because of its superior performance, the random forest regression model can be used to optimize a product's price.

Price optimization takes into account many factors like competition, market trend, etc. and can be done using many different concepts. Better price optimization can be done by taking into account other features which affect price of a product. By providing data about such factors, better models can be built which helps improve accuracy. Therefore, using more data and taking into account factors like market situation, etc. improvements can be made for better optimization of price.

References

1. Wang, Xiaojie, et al. "Price optimization with practical constraints." *arXiv preprint arXiv:2104.09597* (2021)
2. Taleizadeh, A.A.; Niaki, S.T.A.; Seyedjavadi, S.M.H. Multi-product multi-chance-constraint stochastic inventory control problem with dynamic demand and partial back-ordering: A harmony search algorithm. J. Manuf. Syst. (2012), 31, 204–213

3. Tan, B.; Karabati, S. Retail inventory management with stock-out-based dynamic demand substitution. Int. J. Prod. Econ. (2013), 145, 78–87
4. Hu, X.; Wan, Z.; Murthy, N.N. Dynamic pricing of limited inventories with product returns. Manuf. Serv. Oper. Manag. (2019), 21, 501–518
5. ABC News. Amazon Error May End “Dynamic Pricing”. (2000)
6. Jani, Mrudul Y., et al. "Optimal Pricing Policies with an Allowable Discount for Perishable Items under Time-Dependent Sales Price and Trade Credit." *Mathematics* 10.11 (2022): 1948
7. Fumagalli, C.; Motta, M. Exclusive Dealing and Entry, When Buyers Compete. Am. Econ. Rev. (2006), 96, 785–795
8. Fumagalli, C.; Motta, M.; Persson, L. On the Anticompetitive Effect of Exclusive Dealing When Entry by Merger is Possible. J. Ind. Econ. (2009), 57, 785–811
9. Chen, Po-Yu. "Dynamic Sales Price Control Model for Exclusive Exquisite Products within a Time Interval." *Processes* 9.10 (2021): 1717
10. Phillips, R. L. Pricing and Revenue Optimization. Stanford University Press, (2005)
11. https://www.profitero.com/2013/12/profitero-reveals-thatamazon-com-makes-more-than-2-5-million-price-changes-every-day
12. Schlosser, R. Dynamic Pricing with Time-Dependent Elasticities. Journal of Revenue and Pricing Management 14, 365–383, (2015)
13. Schlosser, Rainer, and Martin Boissier. "Dynamic pricing under competition on online marketplaces: A data-driven approach." *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. (2018)
14. https://www.kaggle.com/datasets/suddharshan/retail-price-optimization
15. https://towardsdatascience.com/random-forest-regression-5f605132d19d
16. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html
17. https://www.geeksforgeeks.org/random-forest-regression-in-python
18. https://towardsdatascience.com/random-forest-regression-5f605132d19d
19. https://www.simplilearn.com/what-is-multiple-linear-regression-in-machine-learning-article