

AMAZON PRICING PROJECT



BY GETRUDE SHABIHA

Introduction



Business Problem: Predicting the best pricing for product by new amazon sellers

- Due to competition an amazon seller needs to list their product with the most competitive prices.
- When looking for the best product, a customer is influenced by factors like the price, discount, ratings and the rating count.



- Amazon is an online/ecommerce marketplace. It offers a diverse range of products, services like Amazon Prime, and innovative ventures like AWS and Kindle. Dominant force in the digital marketplace.
- Amazon FBA (Fulfillment by Amazon) was introduced in September 2006. It is a service offered by Amazon that allows sellers to store their products in Amazon's fulfillment centers. Amazon handles the storage, packaging, and shipping of the products, as well as provides customer service and returns processing. This service has been instrumental in enabling sellers to leverage Amazon's vast infrastructure and reach a broader customer base efficiently.
- The introduction of Amazon FBA has led to an increase in competition leading to fewer sales and revenue for the sellers. Naturally, an increase in supply competition leads to price reduction by sellers to attract more customers which in turn leads to lower revenue.

What I plan to achieve with my project

Help amazon sellers to set the best price and discount to get the highest possible ratings and rating count.

Primary goal of the project

- Assess which categories are likely to get the most reviews and the highest average rating.
- Assess the best discount percentage for the most reviews which translates to more sales.
- Identify the best percentage discount for each individual price point.

How it will help the business

- An amazon seller can influence the ratings and rating count by setting the best price and relative discount which lead to more sales.
- More sales means there is an opportunity to request for more reviews from customers which later loops back to more sales



Implications of this project

With this project New amazon sellers can:

- Know how to set the best discount amount that can attract the most customers sales and therefore more reviews
- Find out the general trend in pricing and discount of similarly categorized products.

Ultimately new sellers can leverage the data of existing sellers to better break into the market.



Describing the Dataset



Where has the data been gathered?

This dataset is a list of 1400+ amazon products which was scraped from the platform and posted on the kaggle data and machine learning platform. See link below;

<https://www.kaggle.com/datasets/karkavelrajaj/amazon-sales-dataset>

What are the input and output variables?

- The input variables are the actual price, discounted price and discount percentage.
- The output variable is the average rating which I am using to predict the performance of a product.



Descriptive Analysis



Number of rows and columns

```
In [3]: amazon_data.shape
```

```
Out[3]: (1465, 16)
```

- My initial dataset row and column size can be fetched using the “.shape function”
- As seen above by running the .shape function we can see that the dataset has a row count of 1465 and a column count of 16



The dataset has the following data columns:

- product_id - Product ID
- product_name - Name of the Product
- category - Category of the Product
- discounted_price - Discounted Price of the Product
- actual_price - Actual Price of the Product
- discount_percentage - Percentage of Discount for the Product
- rating - Rating of the Product
- rating_count - Number of people who voted for the Amazon rating
- about_product - Description about the Product



- user_id - ID of the user who wrote review for the Product
- user_name - Name of the user who wrote review for the Product
- review_id - ID of the user review
- review_title - Short review
- review_content - Long review
- img_link - Image Link of the Product
- product_link - Official Website Link of the Product



Percentage of missing Values

```
In [4]: ▶ print('Missing values:\n', amazon_data.isnull().sum())
```

```
Missing values:
product_id          0
product_name        0
category            0
discounted_price    0
actual_price        0
discount_percentage 0
rating              0
rating_count        2
about_product       0
user_id             0
user_name           0
review_id           0
review_title        0
review_content      0
img_link            0
product_link        0
dtype: int64
```

- In my dataset, there were only 2 missing values in the rating count column. This accounted for 0.13% of the total data in the dataset. I considered this an insignificant value and dropped it completely from the dataset.



Mean, Median, Mode and Variance table

	Statistic	Discounted Price	Actual Price	Discount Percentage	Rating
0	Mean	814.572412	1647.076866	54.313402	4.161237
1	Median	349.000000	999.000000	58.000000	4.200000
2	Mode	199.000000	999.000000	60.000000	4.300000
3	Variance	5.318606e+06	1.410994e+07	384.471824	0.667588

- The above table shows the mean, median and mode values from my dataset. To fetch the above results, I used the statistics package in my code.



Variance

- In this case, the variance represents the variability or spread of values within each column of the dataset. It is a measure of how much the values deviate from the mean.
- For my data, the discounted price and actual price had a high variance which indicates that some values are significantly further away from the mean. It is for this reason that we would rather focus on the discount percentage as a favorable input as the variance is significantly lower thus predictions are less likely to be skewed.

Outliers

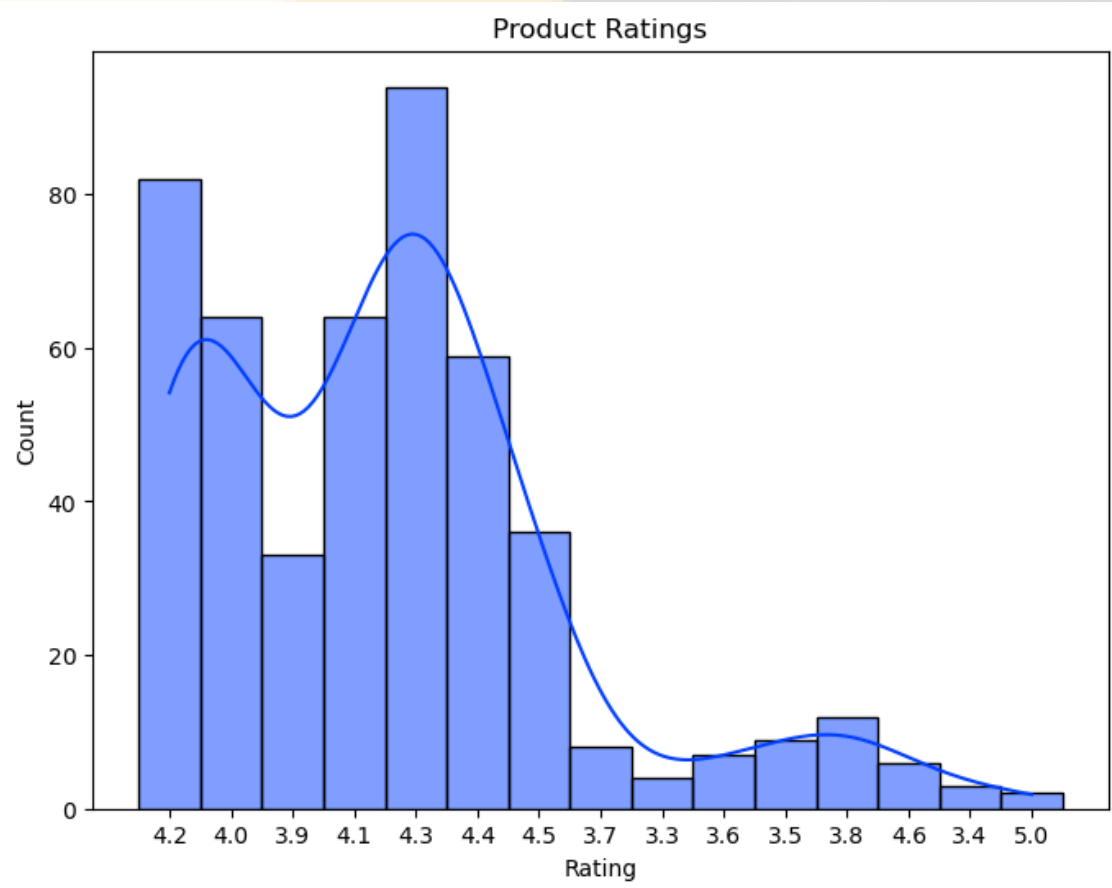
- Outliers included data that deviated significantly from the mean. I observed this as I noticed a huge price difference from categories.
- For this, I reduced the dataset to focus on a single category.



Data visualization

Correlating diagrams between Discount Percentage and Average Rating

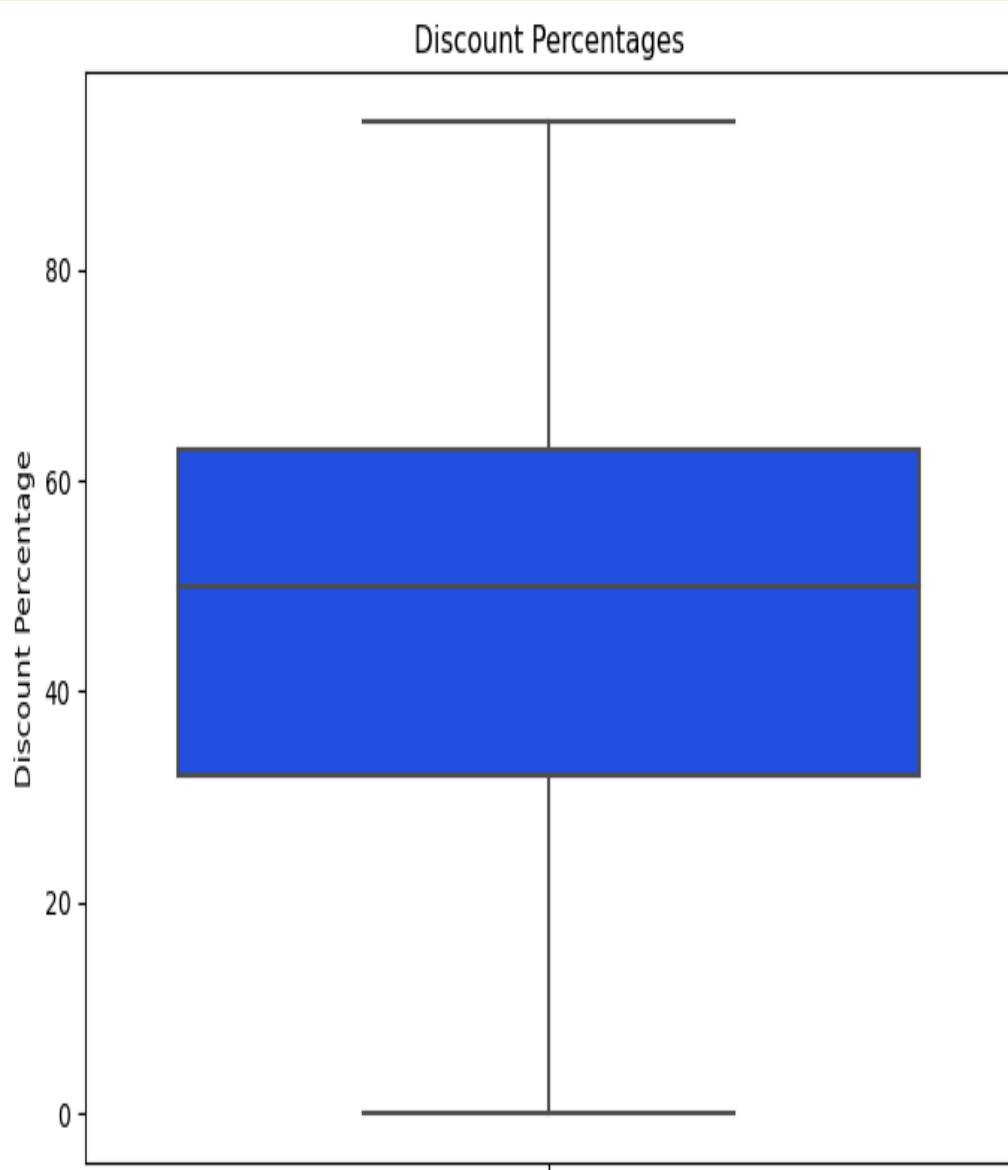
Histogram



- This histogram considers the rating values and plots them against the number of times they appear in the dataset.
- As evident from the histogram the average rating value is very high with rating values between 4 and 3 occurring most of the time
- After the rating value of 3 the occurrences drop drastically.
- This could skew our predictions later due to an excessively positive occurrence.

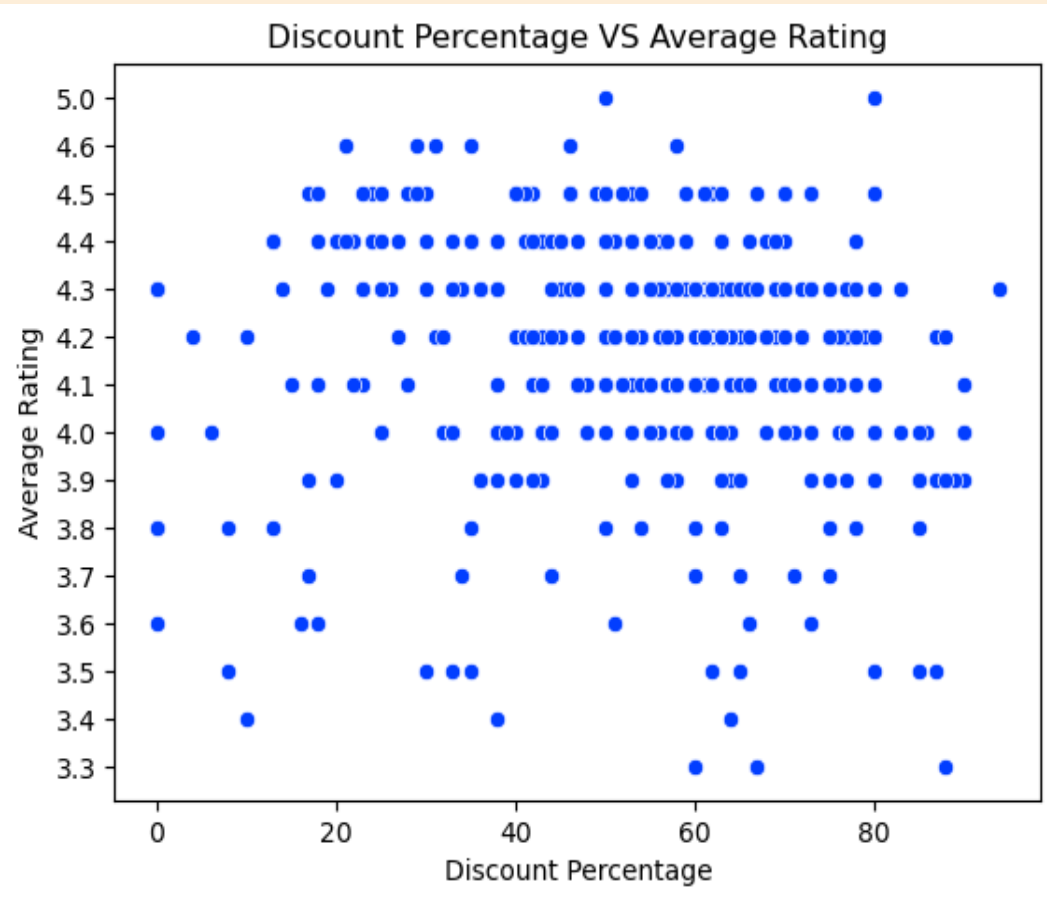


Box plot



- The boxplot displays the distribution of discount percentages for Amazon products.
- The y-axis represents the discount percentage, which indicates the amount of price reduction as a percentage of the original price.
- The box in the plot represents the interquartile range (IQR), which encompasses the middle 50% of the data. The line within the box represents the median discount percentage.
- The whiskers extend from the box and indicate the range of the data, excluding any outliers. Outliers are represented as individual points beyond the whiskers. 🔊

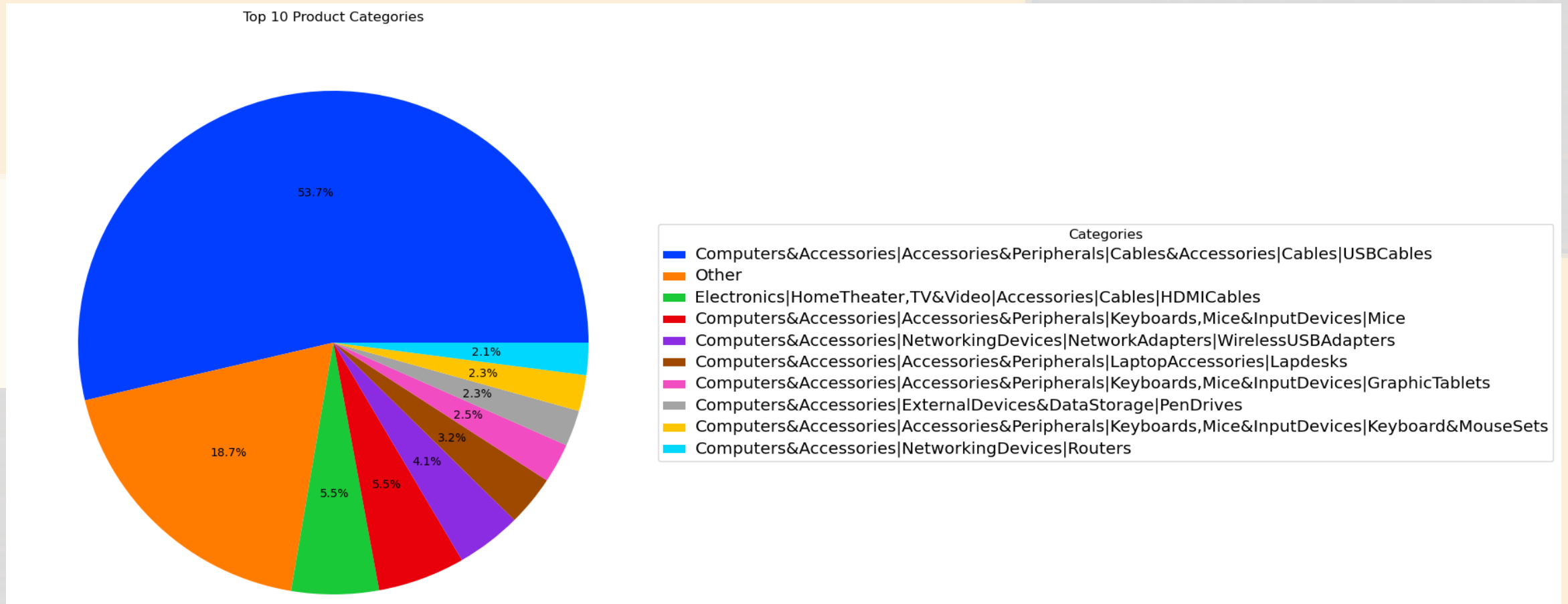
Scatterplot



- This scatterplot diagram shows the correlation between the discount percentage and the average rating.
- From the diagram, we can deduce that a higher discount percentage influences a higher average rating awarded by the customer.
- Looking at the chart, discount percentages above 40% show a high rating of above 3.9
- Products with a discount below 40% show a lower rating of below 3.9



Pie chart



- The above pie chart shows the category distribution of all the products in the dataset.
- Most of the products were placed in the Computers&Accessories|Accessories&Peripherals|Cables&Accessories|Cables|USBCables category
- The above visualized dataset is what I will process and feed into my model.

Data Preprocessing



Variable treatment

The following steps were taken to prepare my data:

- Removing null values in the dataset.
- Trimming down the dataset to focus on only one category.
- Variable selection by isolating columns that are needed and dropping any other column which may be considered an outlier.



```
In [4]: ▶ print('Missing values:\n', amazon_data.isnull().sum())
```

```
Missing values:
 product_id      0
product_name     0
category         0
discounted_price 0
actual_price     0
discount_percentage 0
rating           0
rating_count     2
about_product    0
user_id          0
user_name        0
review_id        0
review_title     0
review_content   0
img_link         0
product_link     0
dtype: int64
```

- In the sourced dataset we can print out the value count of columns that have null values.
- In this case, the dataset has two values in the rating count that are null.
- This accounts for 0.13% of the total data.
- To have a clean dataset, the null values will be removed.



amazon


```
In [5]: ▶ amazon_data.shape
```

```
Out[5]: (485, 16)
```

- In the sourced dataset, we can print out the value count of columns that have null values.
- In this case, the dataset has two values in the rating count that are null.
- To have a clean dataset, the null values will be removed.



Variable Selection

- Looking back, you can see that, my project aims to predict average ratings from a desirable variable that I will choose from my dataset.
- From the data, high average rating on a product is influenced from a customer's satisfaction with a product. We can, therefore, assume that a product's pricing will affect a customer's perception of value for their money.
- In the dataset, I have three columns that can help me achieve this. The columns are:
 - Actual product price
 - Discounted price
 - Discount percentage
- Using the actual and discounted price, I can calculate the discount percentage. At this stage, I verify that the corresponding values in the discount percentage column are correct 

Fitting Models

In this project, I have used 3 models on the dataset, that is;

- KNN Regression
- Decision Tree
- Linear regression

The dependent variable was “discounted price and actual price”

The independent variable was:

- Average rating

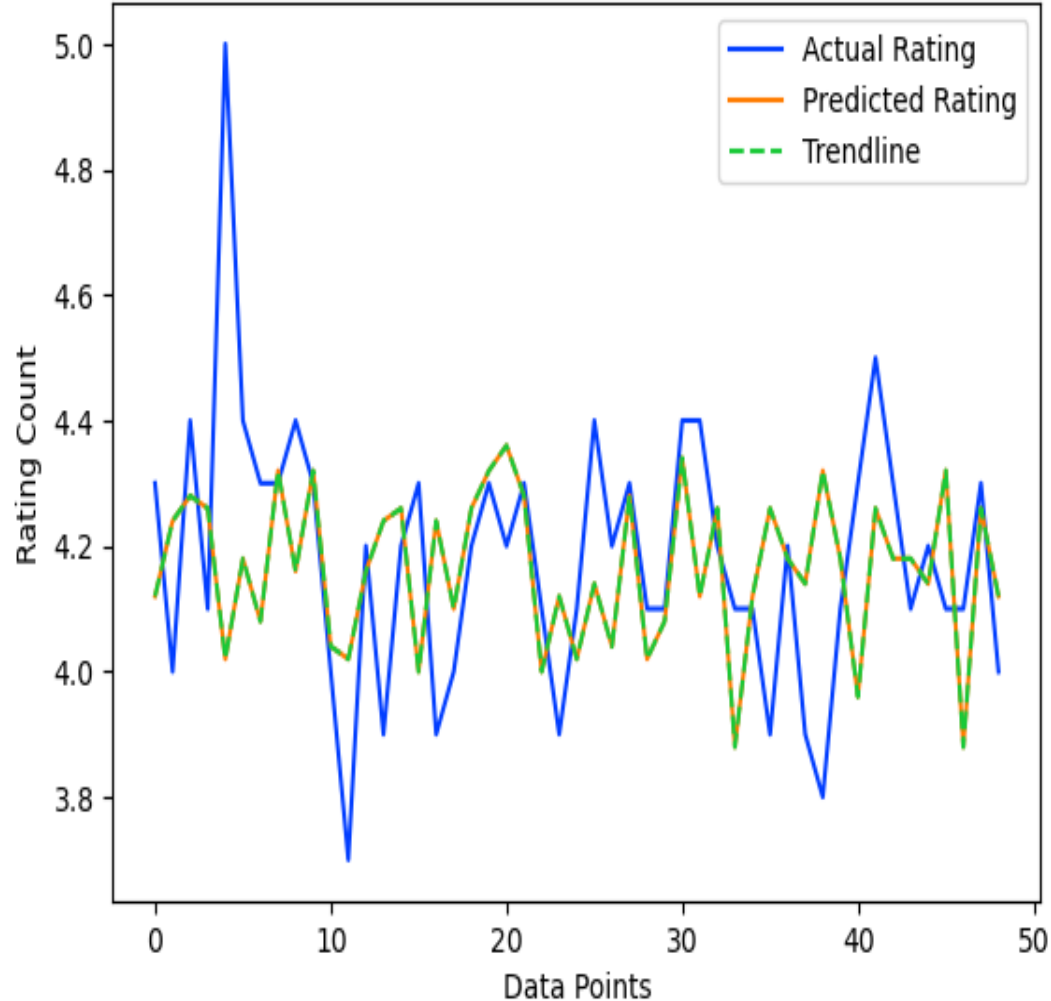
With the above variables, the goal was to:

- Predict the possible number of ratings that a seller can get on a given product.
- Identify the best performing model.



KNN Regression Model

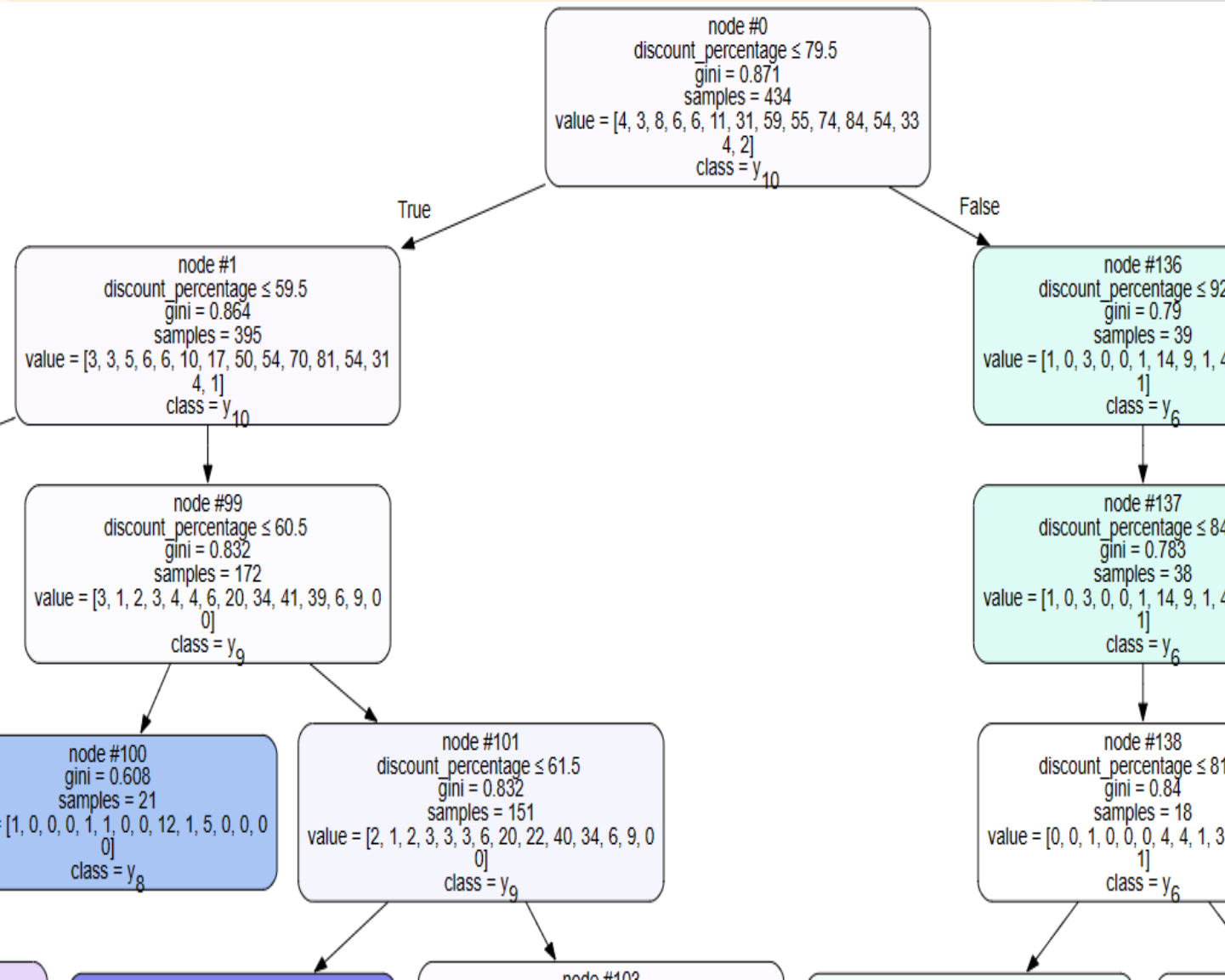
Actual vs. Predicted Rating with Trendline



- By observing the distance between the actual rating counts and the trendline, it's possible to assess the variability in the predictions. If the predicted rating counts are scattered around the trendline with significant deviations, it suggests that the model's predictions might have higher uncertainty or that there are additional factors not considered in the current model that influence the rating counts.
- The accuracy of the KNN regression model can be evaluated by examining the proximity of the predicted rating counts to the actual counts. The mean squared error (MSE) is one measure to quantify the overall model performance. A lower MSE indicates that the model is better at predicting the rating counts.
- This model scored an MSE of 0.08



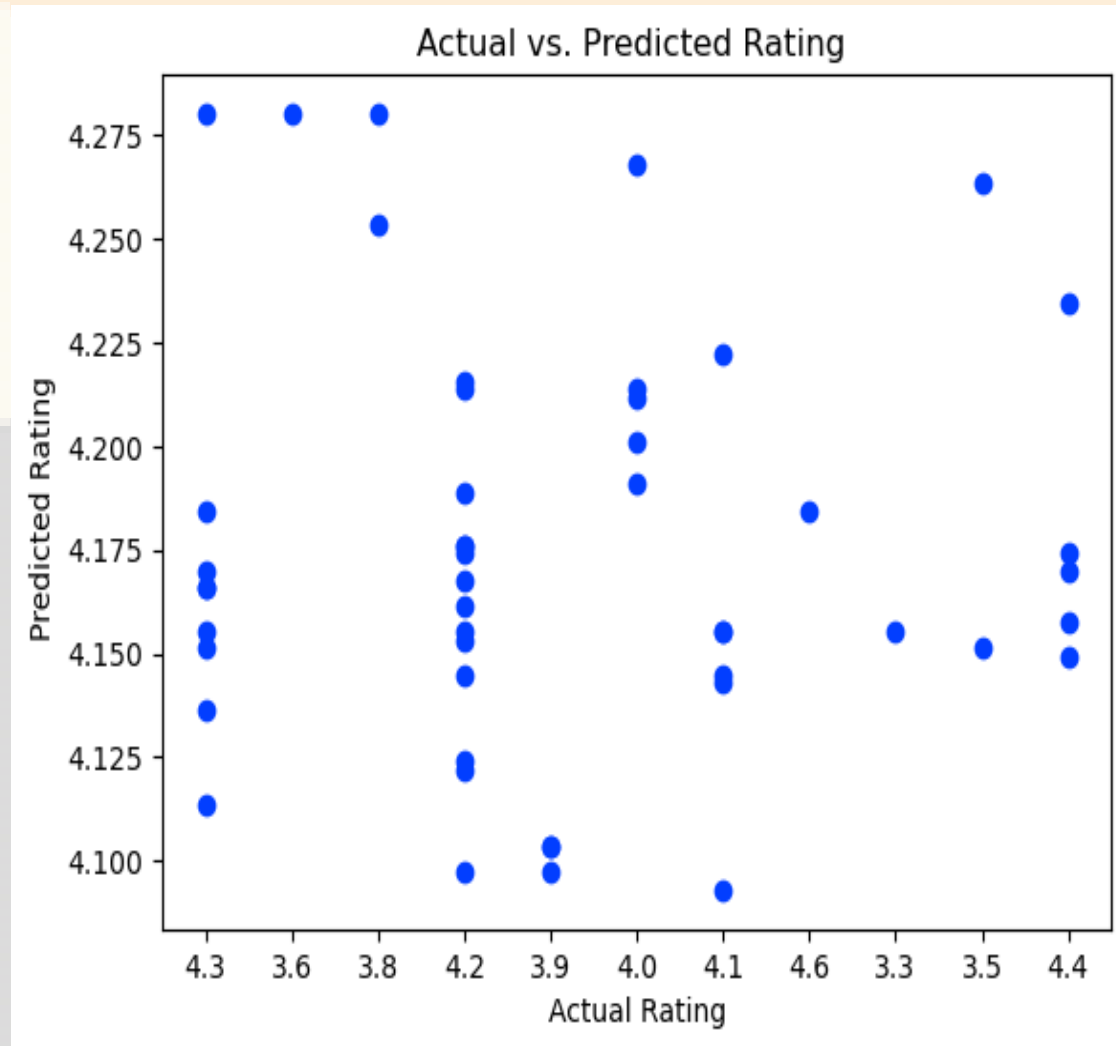
Decision tree sample preview



➤ This diagram shows a small representation of the tree model that was generated by the decision tree model.



Linear Regression scatter plot diagram



- The scatter plot shows a collection of points representing the relationship between the predicted rating and the actual rating.
- The plot reveals a generally positive relationship between actual prices and discounted prices, indicating that highly priced items tend to have higher discounts.
- There are some outliers visible in the plot, indicating a few instances where products had significantly higher or lower discounts compared to their actual prices.

Summary of the Models' Performance Indicator

- I was able to access the accuracy of the models through their inbuilt functions.
- For the decision tree, I used the accuracy indicator which is a function that ranges from 0 to 1 . Any value in between can be calculated to a percentage by using the formula “value X 100%”.
- The decision tree was the least accurate and inconsistent model with its accuracy score ranging between 20% and 40%.
- The accuracy score for both the linear and KNN regression models was measured by the Mean Squared Error value. In this case, the lower the value the more accurate the model is. However, this does not truly explain how accurate a model is when there is no comparison model.
- Both regression models scored below 0.09 which indicates their high level of accuracy.



Conclusion



Key Insights

The descriptive analytics showed that:

- Most of the products were placed in the Computers&Accessories|Accessories&Peripherals|Cables&Accessories|Cables|USBCables category
- Most of the average rating count was higher than 50,000

The Predictive model showed that:

- The average rating count was lower than the actual rating count.



Answering the business problem

- If a new seller wants to start selling on amazon, they must pay attention to the rating of similarly priced and discounted offers.
- This gives a slight indication of the potential market size and projected sales.

amazon

Was the goal achieved?

Using both regression models showed the most accurate data prediction on the average rating based on the discount.

We can confidently say that our models had significant accuracy that can be used by a seller to set their discounts.



Obstacles met

- Dataset cleaning and data preprocessing- This was remedied by reading documentation and tutorials online
- Syntax errors and warnings- Jupyter notebook has a very useful debug console that points out the problem which can be helpful when looking for the solution online
- Imbalanced dataset- most of the rating count predictions were inconsistent to the actual discount percentage of the product.



Possible changes that could have been made

- Feature Selection/Engineering: Carefully select or engineer relevant features that have a strong impact on the target variable. Removing irrelevant or redundant features can improve model accuracy by reducing noise and overfitting.
- Model Selection and Tuning: Choose an appropriate model for the problem and dataset. Consider different algorithms, such as random forests, or neural networks, and assess their performance using appropriate evaluation metrics. Tune hyper parameters using techniques like grid search or randomized search to find the optimal combination that maximizes accuracy.



- **Cross-Validation:** Utilize cross-validation techniques, such as k-fold cross-validation, to estimate the model's performance on unseen data. This helps to assess the generalization ability of the model and reduce overfitting. By validating the model's accuracy on multiple subsets of the data, we can obtain a more robust evaluation and make improvements accordingly.
- **Ensemble Methods:** Combining multiple models to create an ensemble that leverages the strengths of individual models. Techniques like bagging (e.g., random forests) or boosting (e.g., gradient boosting) can help improve accuracy by reducing bias or variance in the predictions.

