CS-4641-Site

Amazon Underdogs

Group Members

Evan Rodgers, Matthew Gollins, John Heyerdahl, Mateo Mendoza

TA Mentor

Emanuel Goldin

Background & Literature Review

Amazon is the world's largest online retailer. Of the sellers, 60% of sales conducted through this platform have been through independent sellers, where most are small and medium sized businesses [1]. There is clear evidence that business growth is largely impacted by successful listing of products through online retailers, and providing a tool to optimize this listing of new products may expand growth of more small businesses in the economy [2]. Machine learning allows for this optimization to be far more accessible and cheap, leveling the playing field for small businesses [3].

Dataset Description

The dataset used in this project is data from Amazon product listings, containing over 1 million products across various categories includes pricing, reviews, and product characteristics, and sales

Dataset Link

Dataset Link

Problem Definition

Problem

Independent sellers and small businesses on Amazon may be under-optimizing their pricing strategies, leading to missed opportunities for maximizing profits. For a prospective small business looking to list a new product, there are several categories on Amazon's online store which the product may fit. By accidentally selecting an underperforming category out of several suitable choices, small businesses may be missing out on maximizing their profit margins.

Motivation

An implementation of machine learning on large product datasets may provide insight into how to price items and select a listing category to maximize profits and optimize customer satisfaction for business marketability. Small businesses are often dominated by larger companies, so insight into amazon marketing would allow for these small companies to leverage their flexibility to increase sales.

Methods

Data Preprocessing Methods

- **Data Cleaning**: To avoid erroneous model behavior due to invalid data, products in the dataset with null values for the 'sales', 'price', or 'listPrice' columns were removed from the dataset. Additionally, products with values of 0 in the 'price' or 'listPrice' columns were removed, while products with 0 sales were kept as this field reflects realistic data for business revenue analysis. Through this step, a reduced dataset with entirely valid data points allowed for seamless analysis with the applied Machine Learning Models.
- **Normalization**: Standardize some features such as price, so that models can make predictions without bias towards large values. This was done by normalizing all columns between 0 and 1. Features in the dataset, such as price. All features inputted into ML models in this analysis were first standardized. Namely, 'Percent Markup' and 'boughtInLastMonth' (Sales) were normalized by dividing each of these features of each product by the maximum feature value in the entire (preprocessed) dataset.
- **Feature engineering**: An additional dataset feature, 'Percent Markup', was created to provide a more insightful product feature and implement additional correlation in quantitative analysis for our ML model implementations. This feature was created by performing a simple linear combination of the 'price' and 'listPrice' features through the following equation: [Percent Markup = ('listPrice' - 'price') / 'price'].
- **Categorical Encoding**: With the resulting dataset cleaned, feature engineered, and normalized, the 'category' feature of the dataset was used to group the data based on this feature prior to feeding the data to our ML models. To create a representation of each product category for analysis, a "groupby" numpy operation condensed the data to include a representative product for each category based on the average feature values within the category.

ML Algorithms/Models

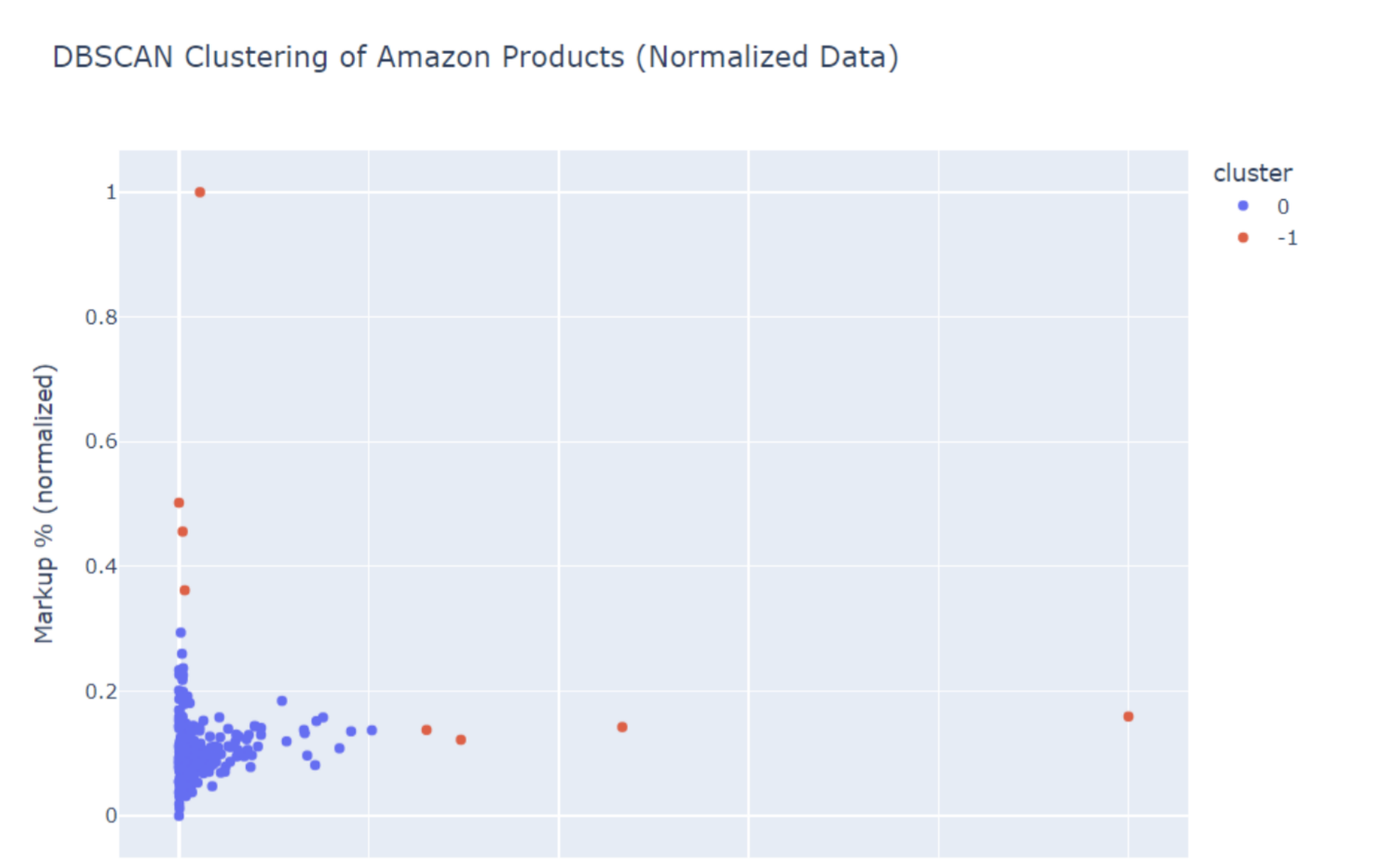
- **DBScan**: Detects outliers in the dataset and identifies anomalies within the product data. It also helps compare similarities between categories when paired with K-Means, revealing unexpected patterns like extreme markups.
- **KMeans**: Can hard cluster product categories based on attributes such as markup vs. sales and markup vs. reviews, simplifying data processing by grouping similar categories together. This could reduce the number of regressions needed and highlight trends across product types.
- **GMM**: Can soft cluster product categories by modeling the distribution of data points based on the relationship between attributes such as markup, sales, and average rating. This approach allows for more informed decisions on grouping, giving a clustering output in addition to KMeans.

Results & Discussion

DBScan

Results Summary & Visualization

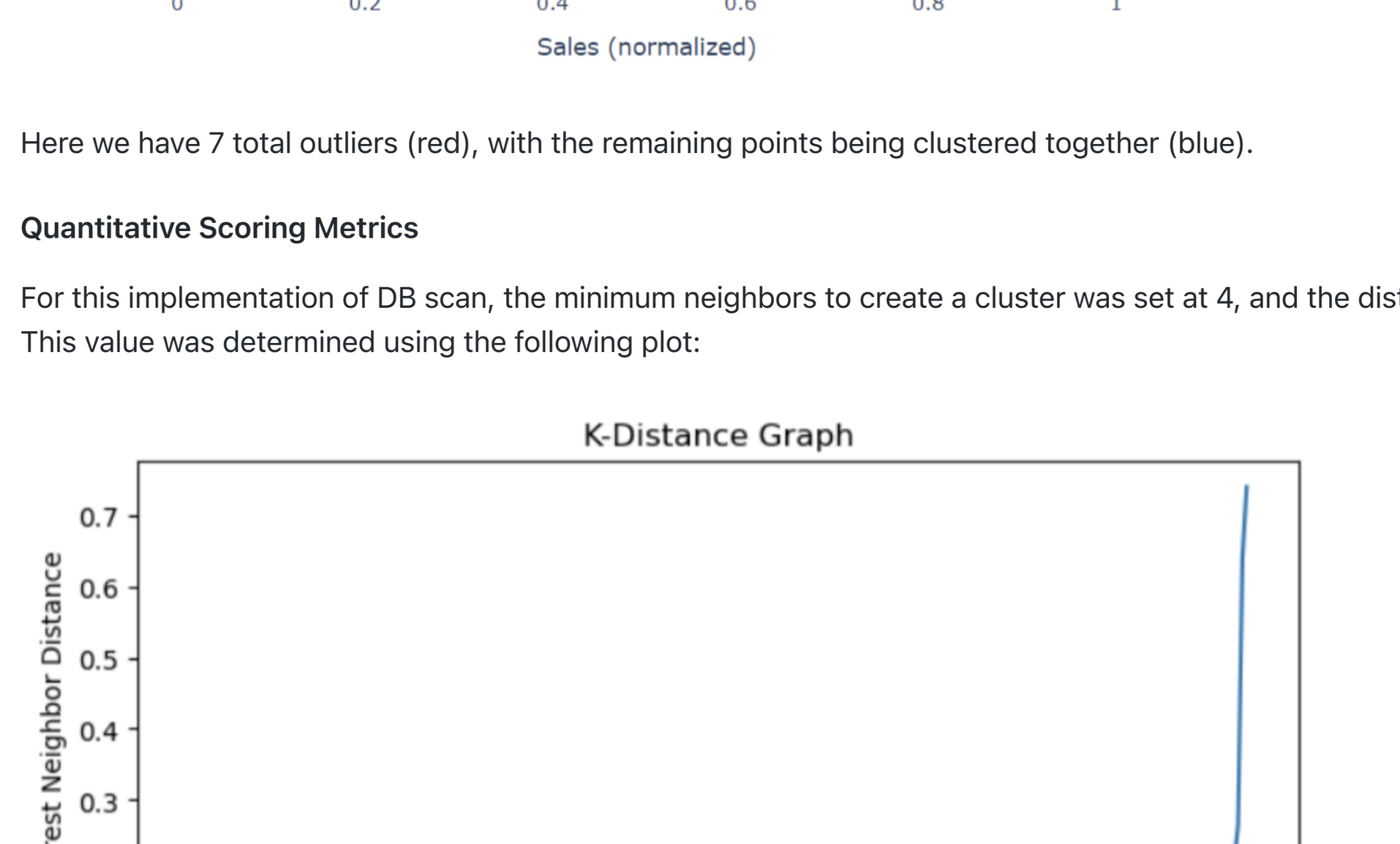
Using DBSCAN on the data resulted in one large cluster and many points classified as noise or outliers. This suggests that overall there is a dense region of categories that exhibit similar relationships between percent markup and sales, with a few categories with more extreme relationships that can be analyzed individually. Below is the plotted data clustered using DB scan.



Here we have 7 total outliers (red), with the remaining points being clustered together (blue).

Quantitative Scoring Metrics

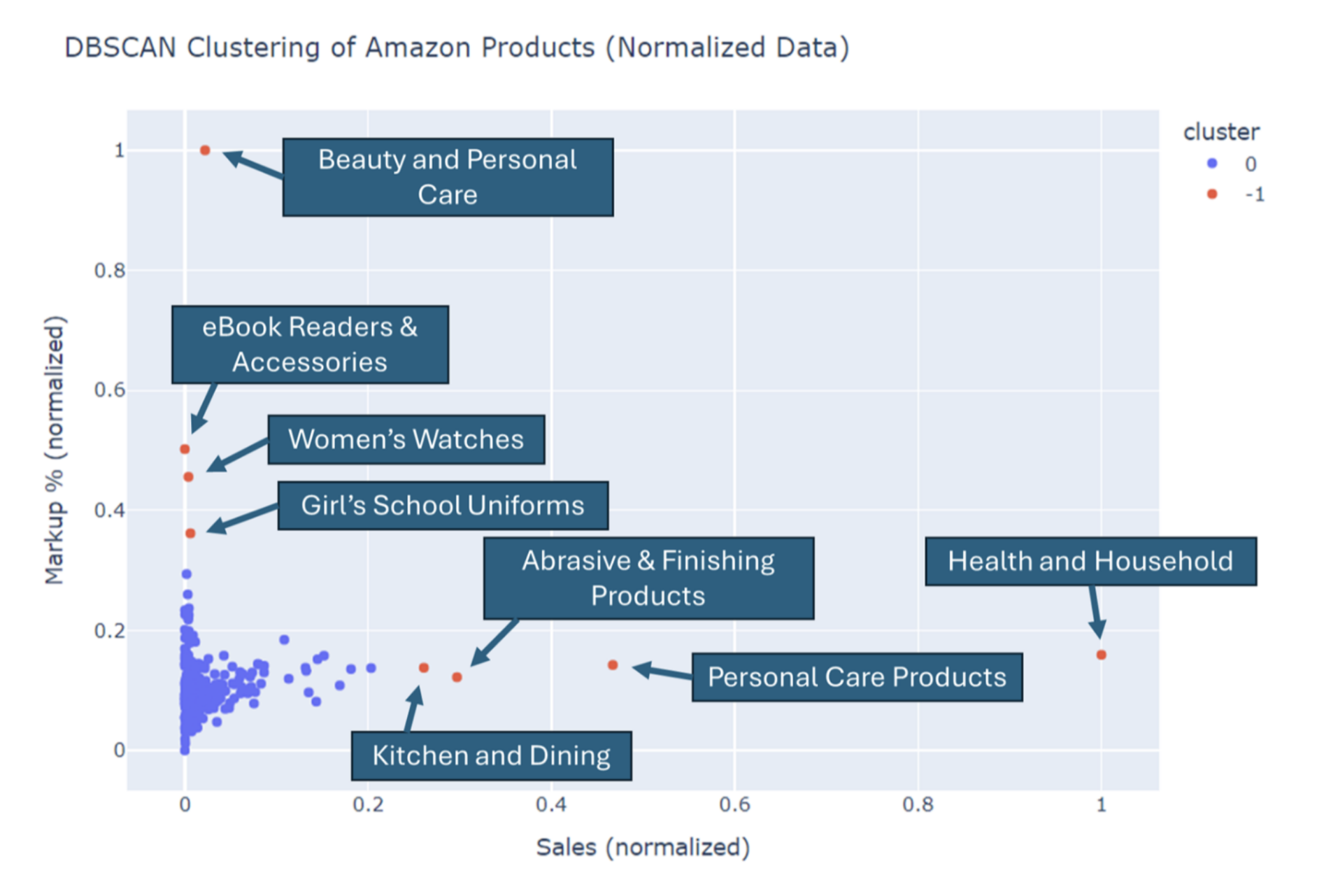
For this implementation of DB scan, the minimum neighbors to create a cluster was set at 4, and the distance (epsilon) was set at 0.5. This value was determined using the following plot:



As the plot shows, the distance to the 4th nearest neighbor is relatively constant for about 225 data points, after which there is a sharp increase. We want to identify the points after this sharp increase as outliers, while not identifying the points in the steady region. The epsilon value associated with the elbow of this plot is around 0.5, so this was chosen. Using this value ensures that we will be able to get the most information from the plot by ensuring that any identified outliers are much more extreme values than the clustered data points.

Discussion

The outliers identified by DB scan represent product categories that are most (or least) optimal to list products within. Below is the DB scan plot with all of the identified outliers labeled by their category.



This graph could be useful to sellers by informing them about which categories typically achieve higher sales, or do better at a higher markup. Most products could fall into multiple categories, so optimizing the category that a product is listed under could have large implications. For example, if a seller has a product that could be classified in either the "personal care products" or the "beauty and personal care products" sections, they should probably try to have it classified as the former. This is because although products in this category typically have a lower markup percentage, they have drastically more sales, which would lead to more profit. However, if the seller had a very limited supply of the product, then listing it under "Beauty and personal care" might be more beneficial, as these products are able to be marked up more, leading to more profit with fewer sales. In general, if it is possible to list a product under one of the four rightmost categories on the graph, then that is what the product should be listed under, assuming that the seller has enough supply to keep up with high sales. The outliers shown higher on the graph represent products that have higher markups than most, but probably will not result in high sales. These categories would be useful to sellers who have very few products that they want to maximize profit from, or even sellers of a single item.

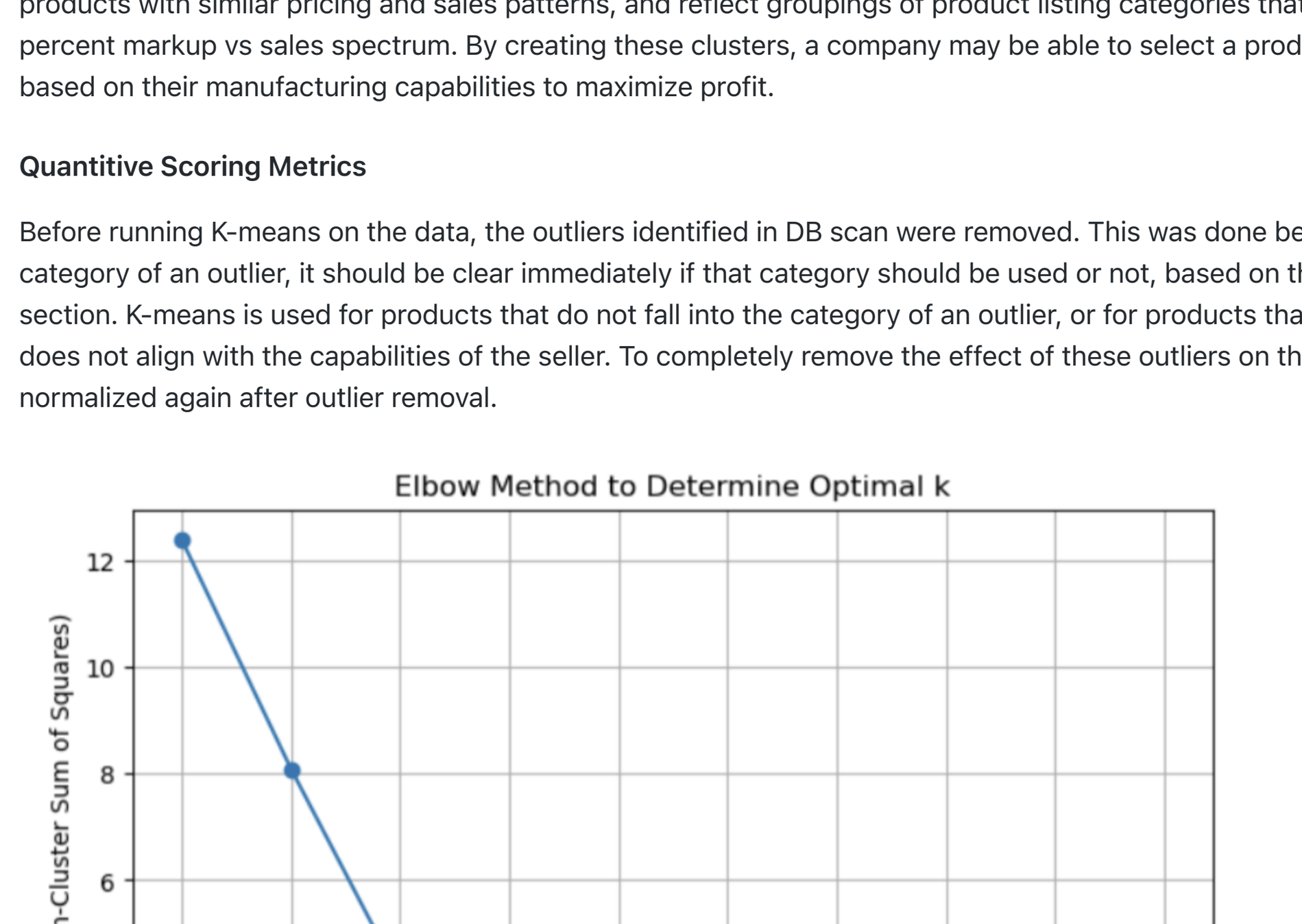
KMeans

Results Summary

K-Means was used to analyze the product categories in the dataset based on normalized sales and markup percentage. Using the elbow method, two different analysis were carried out: one using 3 clusters, and one using 7. The clusters represent distinct groups of products with similar pricing and sales patterns, and reflect groupings of product listing categories that perform in relevant areas of a percent markup vs sales spectrum. By creating these clusters, a company may be able to select a product category to list their product based on their manufacturing capabilities to maximize profit.

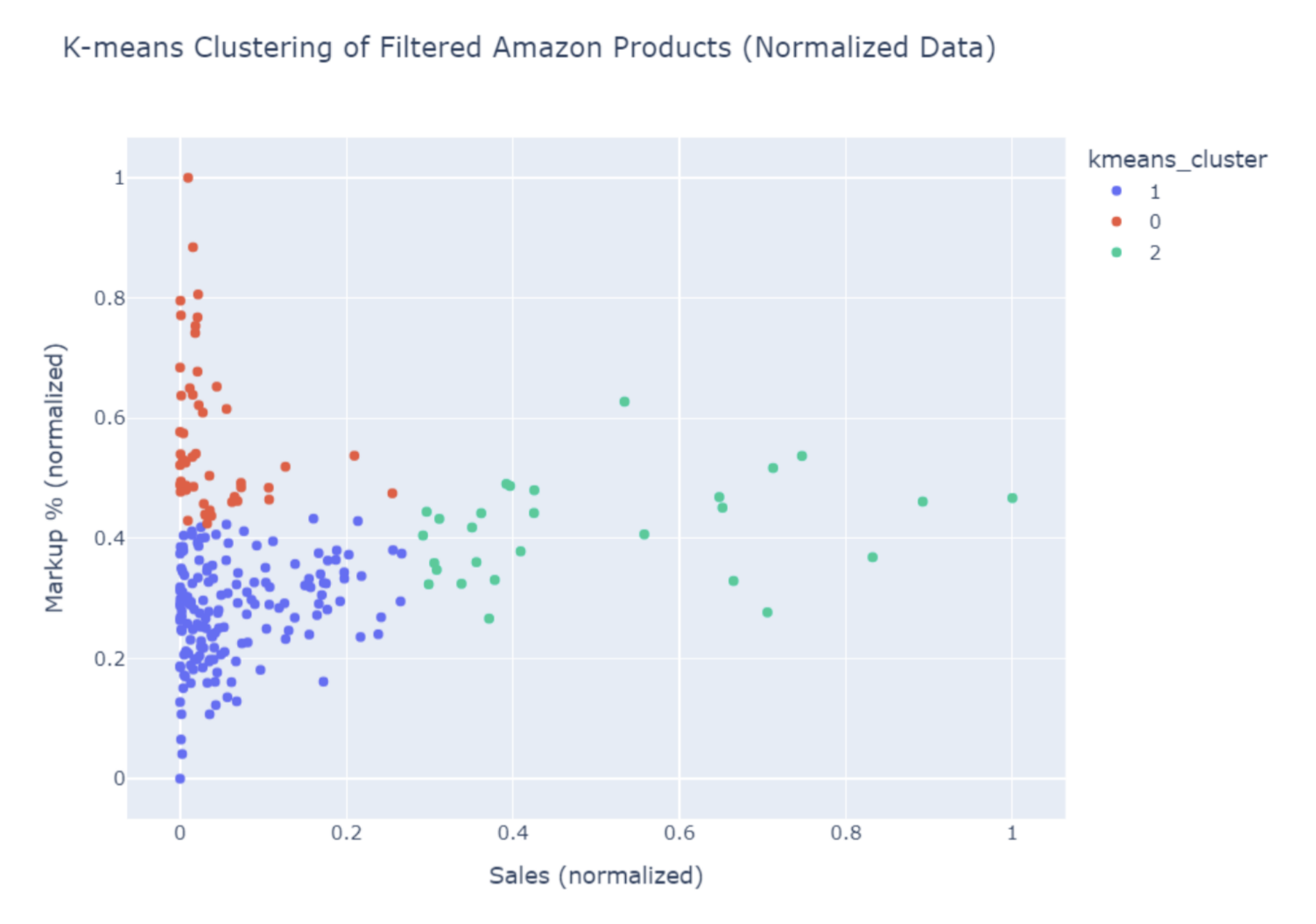
Quantitive Scoring Metrics

Before running K-means on the data, the outliers identified in DB scan were removed. This was done because if a product falls into the category of an outlier, it should be clear immediately if that category should be used or not, based on the discussion in the DB scan section. K-means is used for products that do not fall into the category of an outlier, or for products that fall into an outlier category that does not align with the capabilities of the seller. To completely remove the effect of these outliers on the remaining data, the data was normalized again after outlier removal.



The Elbow Method was used to evaluate the optimal number of clusters to be used in the KMeans analysis of the product data. Based on this test, two sharp gradient shifts from the graph were identified at (3) and (7) clusters - these were the two cluster quantities used in the KMeans algorithm for analysis.

Visualization & Discussion



Using 3 clusters in the initial use of KMeans highlights 3 different product category groupings which span different areas of the Markup Percentage vs Sales spectrum. Based on these groupings, a small business may be able to select a cluster to choose a product category from based on their manufacturing capabilities or individual product novelty. For example, a small business that has access to large scale manufacturing processes would be looking to maximize sales on amazon and would be looking at cluster 2 based on this KMeans analysis, as these product categories tend to achieve higher amounts of sales at a lower markup percentage. On the other hand, smaller companies that have a novel or highly advanced product worthy of a higher markup may look to select a product category from cluster 0 (red) to list their product under, as these categories prove to achieve a higher markup percentage while proportionally seeing less sales than cluster 2.

Applying KMeans with 7 clusters provides a more granular segmentation of product categories. With this analysis of the product categories, small businesses could make more targeted decisions based on their sales and markup objectives. Additionally, with more clusters, small companies can initially target high-performing clusters to choose a product category that best matches their market aspirations. If no suitable categories are found from the initially chosen cluster, they can expand their focus to lower-performing clusters. Overall, introducing more clusters in the K-Means analysis, informed from the elbow method, creates more of a balance between feasibility and market competitiveness.

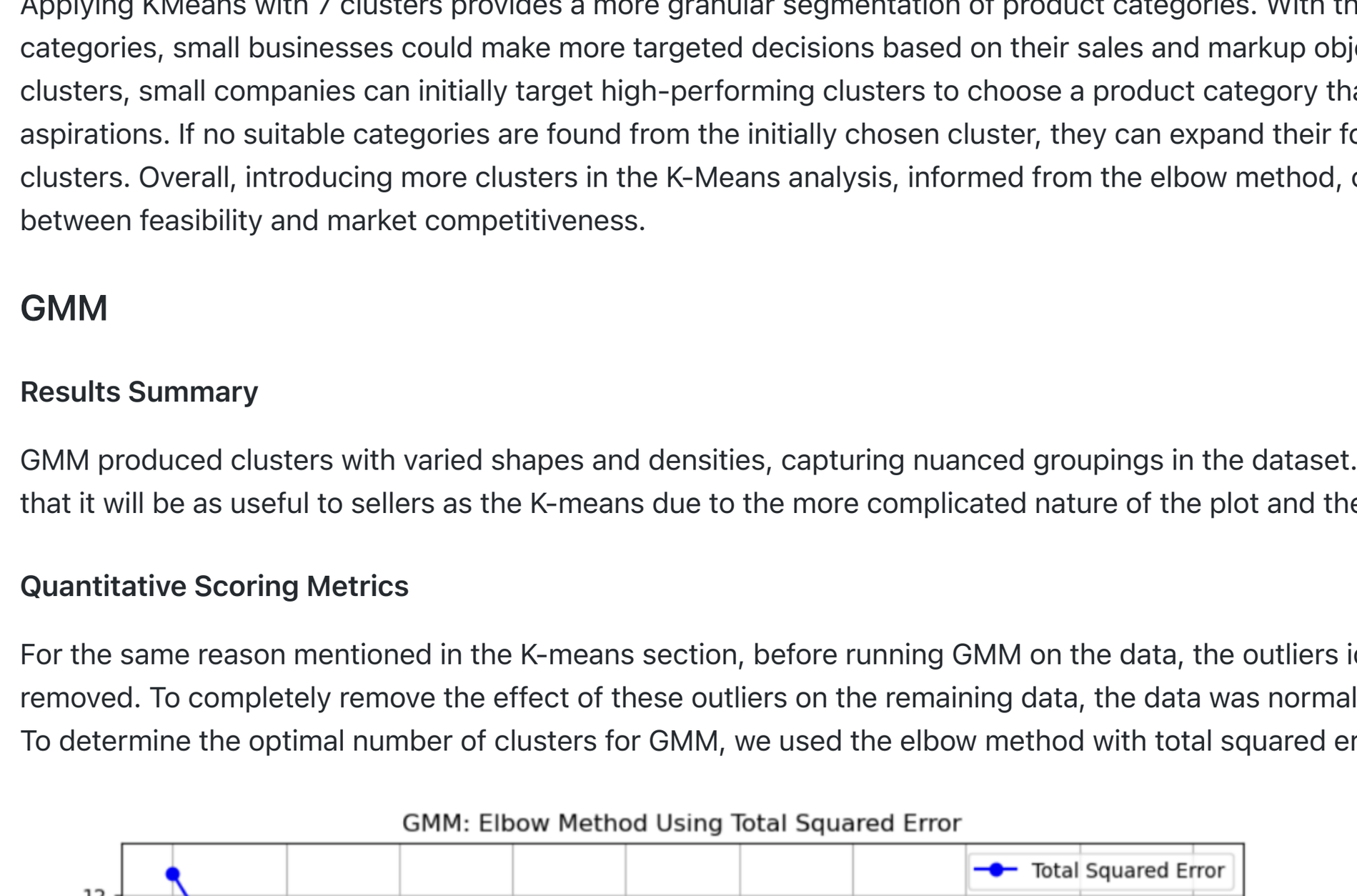
GMM

Results Summary

GMM produced clusters with varied shapes and densities, capturing nuanced groupings in the dataset. However, we do not anticipate that it will be as useful to sellers as the K-means due to the more complicated nature of the plot and the overlapping of clusters.

Quantitative Scoring Metrics

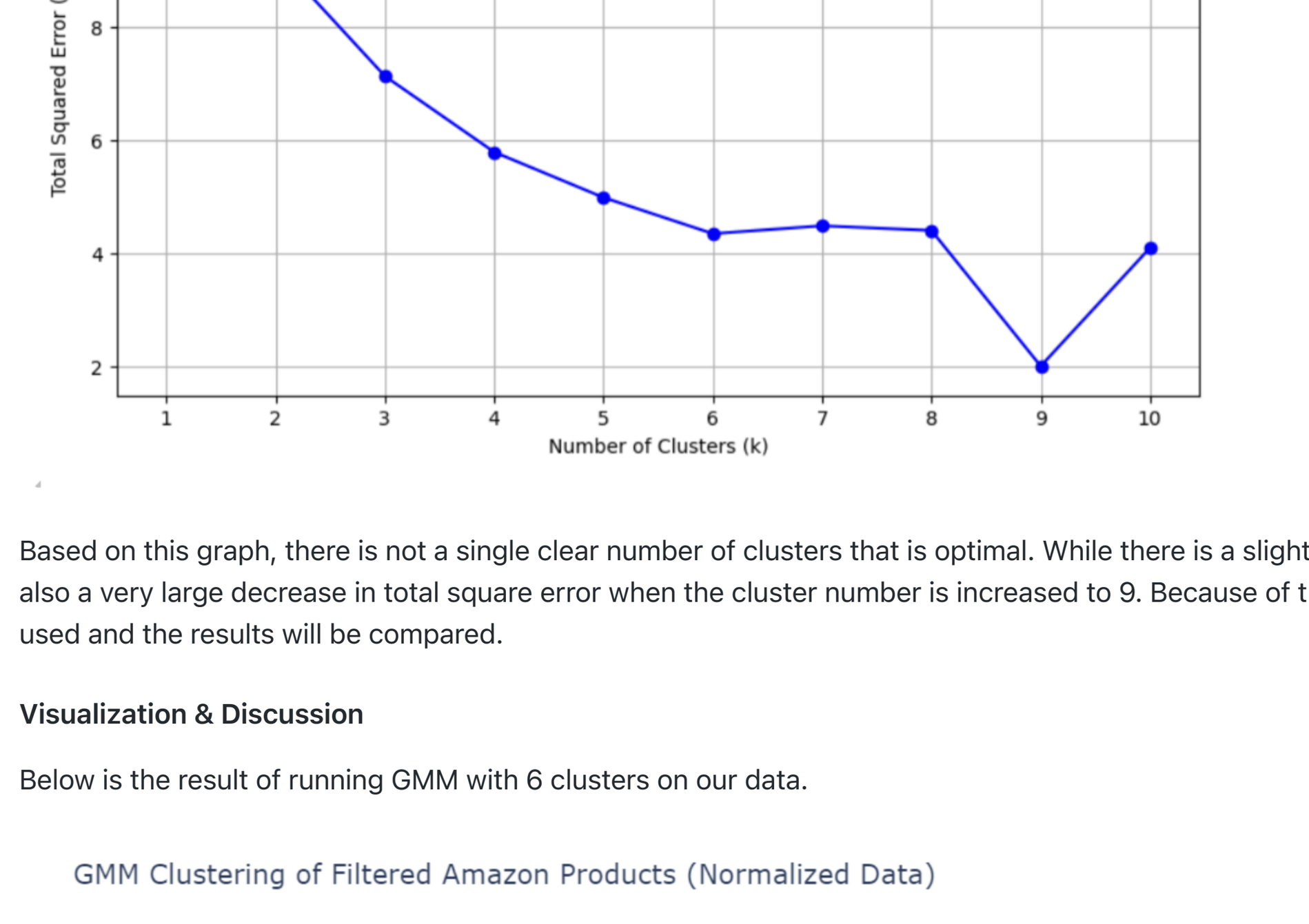
For the same reason mentioned in the K-means section, before running GMM on the data, the outliers identified in DB scan were removed. To completely remove the effect of these outliers on the remaining data, the data was normalized again after outlier removal. To determine the optimal number of clusters for GMM, we used the elbow method with total squared error, shown below.



Based on this graph, there is not a single clear number of clusters that is optimal. While there is a slight elbow at 6 clusters, there is also a very large decrease in total square error when the cluster number is increased to 9. Because of this, both 6 and 9 clusters will be used and the results will be compared.

Visualization & Discussion

Below is the result of running GMM with 6 clusters on our data.



This plot provides insights similar to K-means, and could be used in a similar way. A seller should identify a cluster that aligns with their manufacturing capabilities and provides the highest profit. For example, if a seller has high manufacturing capabilities, they would begin by looking in cluster 2. If a seller has lower manufacturing capabilities, they would begin in cluster 1 (as it would result in more profit than cluster 5). In general, sellers should aim for the cluster closest to the top right corner of the plot when choosing, probably favoring the right hand side over the top, as sales is more important that a slightly higher markup. The seller could then search this cluster for categories under which their product could fall. One benefit in using GMM over K-means for clustering is that it allows the clusters to not be strictly circular, capturing more nuances in the data. For example, sellers should typically try to avoid listing products in the categories in cluster 0. These products have a wide range of markup percentages, but get very low sales. GMM was able to capture this by creating a very thin and long cluster, which K-means would not be able to do.

As discussed earlier, using the elbow method for GMM identified two potential candidates for the optimal number of clusters: 6 and 9. Below is the result of running GMM with 9 clusters on our data.

Although this implementation has a lower total square error than 6 clusters, it is probably not as useful. It could be used in exactly the same way as previously described, but now some clusters are so small that it would be harder to find a category that a potential product could fit under. Additionally, because GMM is a probabilistic clustering algorithm, it is able to have some clusters overlapping, and even have clusters somewhat enclosed within other clusters. This behavior is not ideal for this application, as it would create confusion when attempting to identify an optimal cluster. This was not seen in the 6 cluster graph, but is seen to some extent in this graph, meaning that for our application, 6 clusters provides much more useful information.

Team Organization

Gantt Chart and Contribution Table

Gantt Chart Link

Name	Proposal Contributions
Matt	Updated Github Pages, Data Interpretation
John	Updated Data Preprocessing, Algoritms Implemenation
Evan	Updated DBScan, GMM, Kmeans, Jupyter Notebook
Mateo	Updated Presentation, Research, Misc Support

References

[1] Amazon, "Amazon 2023 Small Business Empowerment Report," About Amazon, 2023.

[2] C. Le, A. Mislove, and C. Wilson, "An Empirical Analysis of Algorithmic Pricing on Amazon Marketplace," in Proceedings of the 25th International Conference on World Wide Web, 2016, pp. 1339-1349.

[3] Tryolabs, "Price Optimization with Machine Learning," Tryolabs Blog, 2024