Optimizing Product Recommendation Decisions using Spatial Analysis

By

Raul A. Hincapie, Bachelor of Science Degree in Engineering Physics

A thesis submitted to the Graduate Committee of Ramapo College of New Jersey

In partial fulfillment of the requirements for the degree of

Master of Science in Data Science

May 2023

Committee Members:

Dr. Amanda Beecher, Advisor

Dr. Osei Tweneboah, Reader

Pablo Maldonado, Reader

Acknowledgements

I would like to thank my entire thesis committee: Dr. Amanda Beecher, Dr. Osei Tweneboah, and Pablo Maldonado. Thank you Dr. Amanda Beecher, for your unwavering support throughout this process, and Dr. Osei Tweneboah and Pablo Maldonado for their support and feedback on this research. I would also like to thank my parents, my sister, and my girlfriend for always pushing me every step of the way.

Table of Contents

Abstract

At a certain Consumer Packaged Goods (CPG) company, there was a need to coordinate between sales, geographic location, and demographic datasets to make better-informed business decisions. One area that required this type of coordination was the replacement process of a specific product being sold to a store. The need for this type of replacement arises when a product is not authorized to be sold at the store, out of stock, permanently discontinued, or not selling at the intended rate. Previously, the process at this company relied on instinctual decision-making when it came to product replacements, which showed a need for this protocol to be more data-driven.

The premise of this project is to create a data-driven product replacement process. It would be a type of system where the CPG company inputs a store and a product then it would output a product list with suitable replacement items. The replacement items would be based on stores similar to the input store using its sales, geographic location, and demographic portfolio. By identifying these similar stores, it is possible that the CPG company could also discover product opportunities or niches for a specific store or region. With a system like this, the company will increase their regional product knowledge based on geographical location as well as improve current and future sales. The system could also provide highly valuable information on its consumer preferences and behaviors, which could eventually help to understand future customers.

Background

In today's business environment, data has become a critical component for companies to make informed business decisions. In particular, CPG companies collect vast amounts of data that range from a store's buying behavior on a weekly or monthly basis to a log of its yearly historical sales. However, the effective utilization of this data has been a challenge for many companies. Research shows that "the average company analyzes only 37-40% of its data [meaning] that almost three-quarters of the data companies collect goes unanalyzed and, as a result, unused. This is an important statistic because it does companies no good to collect the data and then not use it to inform future decisions" (McCain, 2023). In the context of a specific CPG company whose data will be used in this report, they have recognized that this is a problem for them. This is part of an effort to address that problem.

One of the current practices within this company was the reliance on intuition and gut-feelings for their business decisions. While these methods may have worked in the past, they are rapidly becoming insufficient in today's data-driven business domain. In a 2021 article, it stated that after a team had talked to marketing and growth executives at major CPG companies around the world, they found that the executives had one goal: "[fulfill] an ambitious growth mandate [that] requires a marketing agenda that is far more sophisticated, predictive, and customized than ever before [where] marketers [would] now need to utilize data and analytics at scale to crack the code that enables more targeted and engaging interactions to shape consumer behavior." (Chen et al., 2021). Although their focus above was on marketing growth, it comes to show that CPG companies are starting to focus on data driven decision making skills. Additionally, the competition is adapting to this ideology at an increasing rate. According to Anne Grimmelt and Nobert Lurz, who both attended the CAGNY 2021 virtual conference, "two-thirds of CPG

companies say they have put data-driven marketing at the top of their agenda" (Chen et al., 2021). As two-thirds of their competition becomes data-driven focused, the CPG company needs to adapt to the evolving environment of data-driven decisions before it is left behind.

Data

For this CPG company to put their data to its optimal use, we must first look at the datasets. There are three sets of data that will be used in the system dated between January 2022 to December 2022 and reside in the U.S. Due to the data being proprietary, various masking steps were implemented. The three data sets give the sales data, the geographical location data, and the demographic of the region for each store.

Table 1: Detailed information on the three sets of data.

Data set information			
Dataset	***Sales***	***Geographic***	***Demographic***
Description	Percentage of total case sales by all 12 product categories by store.	Geographical data of each store, such as Address, City, State, and Zip Code.	(1) Four main demographic percentages per each store's region and (2) total population by ZIP code.
Source	CPG company's Online Analytical Processing (OLAP) cube used via Microsoft Excel's Pivot Table tool.	CPG company's Online Analytical Processing (OLAP) cube used via Microsoft Excel's Pivot Table tool.	CPG company's proprietary Geographical Information System (GIS) whose data comes from the Census Bureau.
Data Cleaning	- Stores that generated a negative percent share of sales in any of their categories were removed. - Stores that generated no sales were removed. - Warehouse and E-Commerce stores were removed.	- Removed P.O. Boxes. - Discrepancies with the ZIP code values and their representation.	- No cleaning was needed.

After the three sets of data were cleaned and properly structured, consolidation of the three datasets based on a store's main classification number. This was done via Excel's VLOOKUP function and placed into a new Excel sheet where it would contain the main classification number on the left followed by the geographical data, percentage share of sales by category, and lastly the four main demographic percentages. Due to the vast number of stores from the company's database, we wanted to group stores together to effectively analyze them. We began by considering their urbanicity class.

According to the USDA's Economic Research Service, the way the Census Bureau classifies these urbanicity classes are as follows:

- **Urbanized Area (or "urban"):** contains an urban nucleus of 50,000 or more people.
- **Urban Clusters (or "suburban"):** contains an urban nucleus of at least 2,500 but less than 50,000 people.
- **Rural Areas (or "rural"):** contains an urban nucleus of less than 2,500 people (Cromartie, 2019).

For correct classification of stores into the above classes (i.e., Urban, Suburban, and Rural), each store's respective ZIP code was used. By using a store's ZIP code and not their address, we are able to correctly represent the total population where the store resides. If we were to use the total population based on its address, the GIS' method of calculating demographic data would be misrepresenting the total population. Prior to this, we checked and ensured that the values within the ZIP code field were correct and up to date the demographic dataset.

While going through the ZIP code values, a proportionally small number of stores showed discrepancies in their ZIP code field. This varied from the ZIP code being labeled as "0"

or as “XXXXX”, where the possibility of these values being either a placeholder or an inaccurate data entry. To confirm the possibility and overall validity of the address data, we uploaded the addresses onto a Google Map through Google’s “Google My Maps”. To our discovery, results indicated that a substantial number of these stores had issues with their ZIP codes. This discrepancy was noted as location pins for stores appeared to be out of state and, in some cases, out of country. By inputting some of the correct ZIP codes using Google My Maps’ backend table, the original address would correctly pin the store which highlighted a fundamental issue: geocode misrepresentation. There was a high probability that due to the incorrect ZIP codes, both the GIS was acquiring the incorrect demographic data and the urbanicity classification was not representing the respected store.

The solution that arose was to geocode these addresses once again using Python’s Nominatim API. This API “is a tool to search [OpenStreetMap or] OSM data by name and address and to generate synthetic addresses of OSM points (reverse geocoding)” (Nominatim, 2023). The OSM dataset “is a global collaborative (crowd-sourced) dataset and project that aims at creating a free editable map of the world containing a lot of information about our environment” (Mann et al., 2023). For us to use this API within Python, we had to use GeoPy which is a Python client for several popular geocoding web services. Once GeoPy was imported and set with the Nominatim API, the next step was to extract the data from the cleaned address Excel sheet to then import into Python using Pandas’ read_excel function. This function converts the Excel sheet into a Pandas DataFrame, or a data structure within Python’s framework, in order to input the address data into the API. To successfully geocode a store location, it requires the street address, city, and state of the store. Additionally, to keep track of stores, the stores’ main classification number was included at the beginning of the data structure.

After several runs, the API was able to successfully geocode the addresses. The output was a DataFrame that contained three columns: store main classification number, ‘GeoCode Data’, and ‘Lat/Long’. ‘GeoCode Data’ contained the potential name of the location along with its address data (i.e., street address, city, state, and ZIP code), whereas ‘Lat/Long’ contained the latitude and longitude coordinates of the store. The ZIP code from the ‘GeoCode Data’ column was extracted and the DataFrame was restructured to contain only the store’s main classification number and its geocoded ZIP code. The DataFrame was then converted into an Excel sheet using Python’s ’to_excel’ function from the Pandas library. The remaining data was stored as an Excel workbook for future access.

Using Excel’s VLOOKUP function, we were able to successfully replace the misrepresented ZIP codes with the correctly geocoded ones within the consolidated file and the original address file. To confirm its validity, we uploaded the corrected addresses onto the Google My Maps tool. It resulted in correctly geocoding all stores with their pins showing the correct location. Since the ZIP codes were now correct, we were able to rerun the same GIS exercise previously mentioned to acquire the correct demographic percentages. In addition to this exercise, we were now able to continue classifying each store with their respected urbanicity class.

Exploratory Data Analysis

To classify each store with their respective urbanicity class, we used the demographic dataset's total population by ZIP code. Using the same prior VLOOKUP logic, we were able to automatically classify each store to their urbanicity class within the consolidated file. The results showed that 31% of stores reside in an **urban** area, 68% of stores reside in a **suburban** area, and 1% of stores reside in a **rural** area as shown in Figure 1.

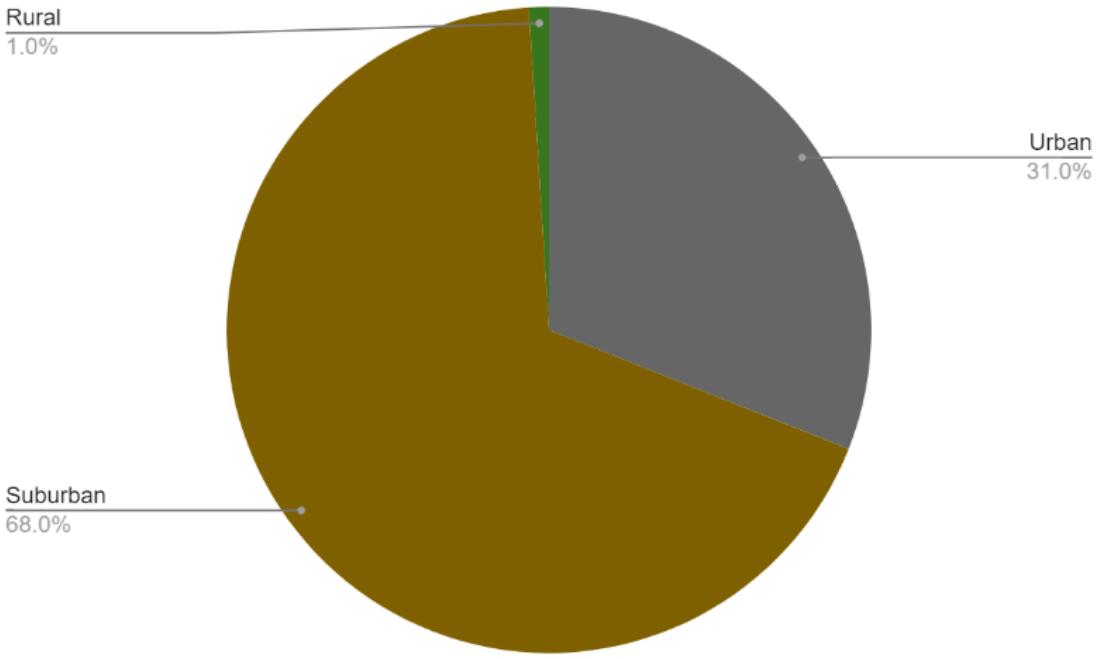

Figure 1: Urbanicity Distribution by Store

This was expected due to the large population range (2,500 to 50,000 residents) of the suburban class. We used Python's matplotlib library to plot each product category's store percentage sales by urbancity class, given in Figure 2. This would then give us a visual representation of the company's sales distribution across the twelve categories by urbanicity class.

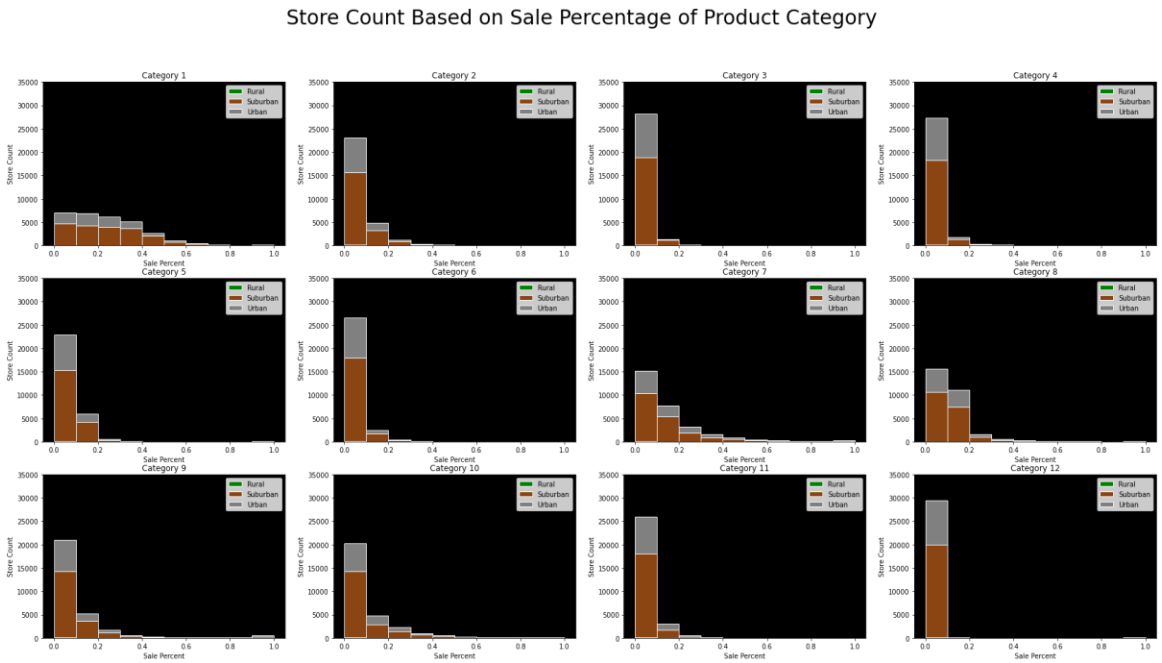

Figure 2: Sales distribution of all stores by all product categories.

Each plot contains all U.S. stores that this company sells products to. This means if one of the stores is purchasing 10% of Category 1 products, the remaining 90% of their sales must be scattered within one or more of the other 11 product categories. Looking at the graphs themselves, the horizontal axis represents the category's sales percentages of each store split into 10 bins: making the range 0% to 10%, 10% to 20%, 20% to 30%, 30% to 40%, 40% to 50%, 50% to 60%, 60% to 70%, 70% to 80%, 80% to 90%, and 90% to 100%.

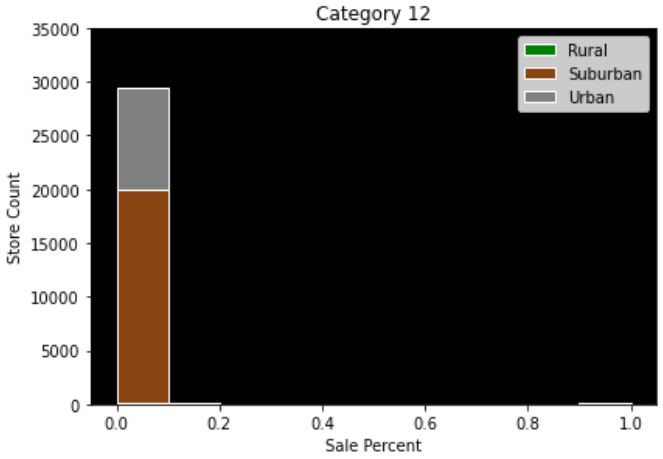

Figure 3: Sales distribution of all stores by Category 12.

In Figure 3, we see that most of Category 12's sales reside within the bucket of 0% to 10%. This means that the majority, if not all, of these stores were buying little to no products within this category, but instead they were buying products from others. On the other hand, if we look at Figure 4, Category 1's sales reside in more than just the bucket of 0% to 10%, but in 8 other bins.

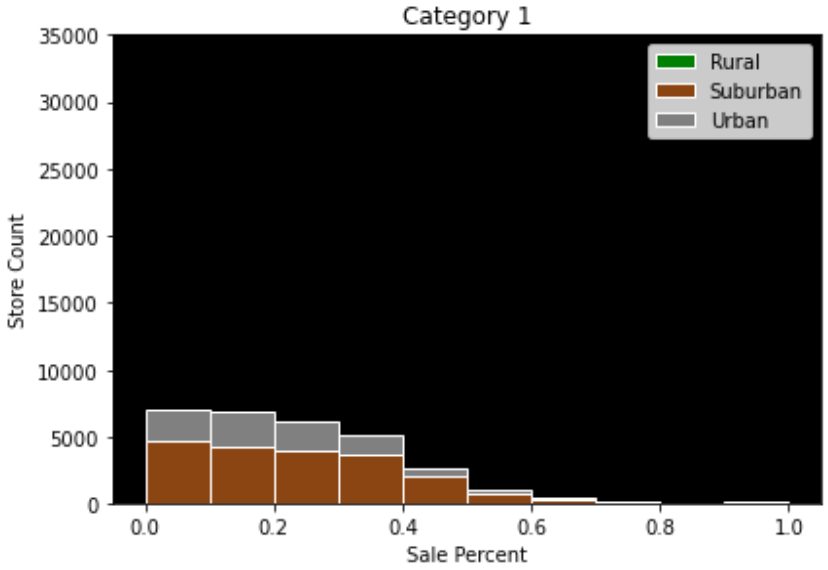

Figure 4: Sales distribution of all stores by Category 1.

This indicates that the category has better sales movement and distribution when in comparison to Category 12. The large variety of sales percentages also tells us there is a large variety of stores making their sales portfolio a certain percent of Category 1's products. In other words, stores who purchase little to none of this category's products reside within the 0% to 10% range. Meanwhile, stores whose majority of their sales come from Category 1 would reside within the 90% to 100% range. According to the company, categories like Category 1 are financially healthy to the company as the wide range of sales percentages indicate cases are being purchased from the category's products. In order for the category to maintain good health,

the category must maintain a positive case sales trend so the category must be constantly reviewed upon their product sales.

The vertical axis for these plots represents the number of stores that reside in each of these columns, but caution must be used when seeing a higher column on a leftmost bin; since this tells us the same conclusion of Category 12 from above: the majority of the category's sales are being bought 'little to none' by all of its stores. When it comes to the urbanicity classes of this visual, it showed us the same percentage trend we had found before. Therefore, for each column of each plot, it was 1/3 Urban and 2/3 Suburban.

Due to the small sample size of rural, it is visually insignificant in all the plots above, rendering it difficult to understand the proportion of stores associated with it. Therefore, we concluded that the visualization helped us understand what categories did best or worst in their sales distribution, but only for urban and suburban. For this reason, it is difficult to fully comprehend the complexity of the three urbanicity classes, making us consider a different type of analysis.

Figure 5 provides beneficial information to understand rural sales distribution and, concurrently, provide conceptualization of the sales percentages across all three urbanicity classes. To do this, we revisited the company's OLAP cube system via Microsoft Excel's Pivot Table tool to extract case sales by each store's main classification number. The next step was to then use the VLOOKUP function to place the case sales of each store within the consolidated Excel spreadsheet. By doing so, we were now able to reinitiate the percentage masking process by urbanicity class and would be done strictly within Python to create the visual.

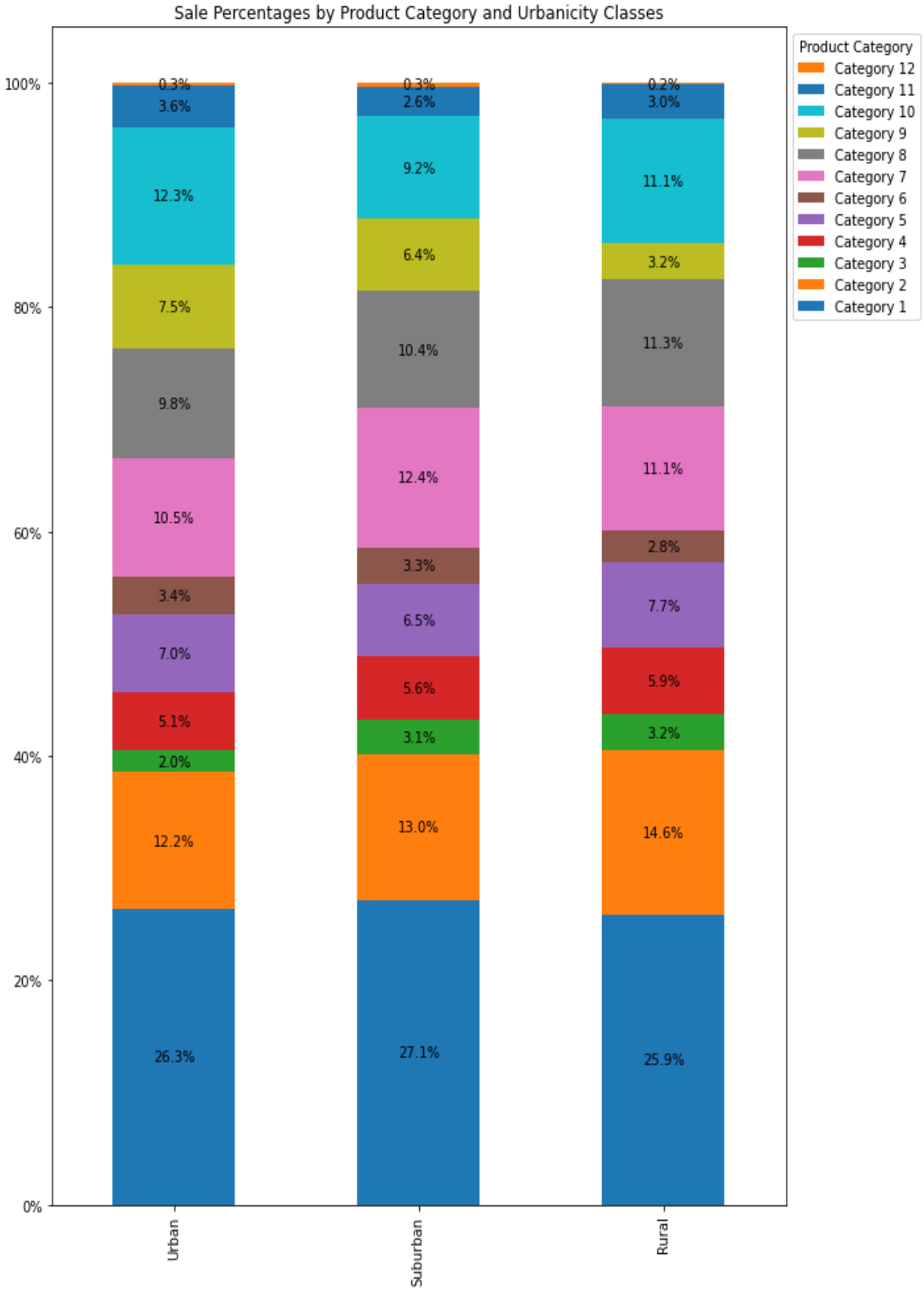

Figure 5: Sales distribution by urbanicity.

The horizontal axis represents the urbanicity classes, while the vertical axis represents the sales percentage of each product category by its urbanicity class. The initial takeaway from this visualization was an increased understanding of the sales distribution within the rural class. When in comparison to the other two urbanicity classes, visually we did not see a major percentage difference.

To confirm this, we tried using two statistical tests. The first test was a one-way Analysis of Variance (ANOVA) test which is a statistical procedure used to compare means of three or more groups (McDonough, 2023). In this case, we ran a one-way ANOVA test at a 95% confidence interval where,

- H_0 (or "null hypothesis"): There is no significant sales percentage difference between the three urbanicity classes.
- H_1 (or "alternate hypothesis"): There is a significant sales percentage difference between the three urbanicity classes.

Using Python's f_oneway function from SciPy's stat sub-module and the percentages of the urbanicity classes in Figure 5, we were able to find a p-value of exactly 1.0. This meant we can conclude that the difference between any of the three urbanicity class percentage arrays is statistically insignificant at the 5% significance level. Therefore, we cannot reject the null hypothesis that any of the three arrays are equal, as no statistically significant difference between urbanicity class percentage arrays was found. Even though it did output that there was no significant difference, we found an issue in our logic. Since each class's sales percentages add up to 100%, this would mean that each set of percentages has a mean of 1/12 (100% total for 12 categories of products). Therefore, when the ANOVA test tries to compare the means of each set,

it is comparing 1/12 to itself three times which makes sense why it would not find a significant difference between the three classes.

The second statistical test was the Chi-Squared Goodness-of-Fit test. A Chi-Squared Goodness-of-Fit test is used to assess if a sample of data came from a population with a specific distribution (NIST, 1989). In our case, the population with a specific distribution, or the observed set, would be the actual case count per each product category of a specific urbanicity class. The sample of the data, or expected set, would then be the sales percentages of another urbanicity class multiplied by the total amount of cases of the observed set. This means that both the observed and expected sets will only contain case count values rather than percentages, where the expected set is dependent on the total cases of the observed set.

Table 2: Combinations of observed and expected sets of urbanicity classes.

Observed	Expected
Urban	Suburban
Urban	Rural
Suburban	Urban
Suburban	Rural
Rural	Urban
Rural	Suburban

Using the different combinations within Table 2 above and the hypotheses of the ANOVA test, the Chi-Square Goodness-of-Fit test gave us a p-value of 0 for each combination, which meant we were to reject the null hypothesis and conclude that there is a difference between the classes. To confirm this even further, we used the same Chi-Square Goodness-of-Fit test, but instead of using actual total case count, we randomized the total. To our discovery, the

test is sample-size-sensitive since the smaller the sample size is, the more probable the combinations in Table 2 would output a large p-value. This type of discrepancy is problematic to having confidence in the results. Thus, this report focuses on the model creation and leave the statistical testing as future work.

Methodology

In order to pursue the path of model creation, it was imperative to first grasp two crucial topics: measures of closeness and clustering the data. These two components play important roles in the model creation and they both create the fundamental steps in analyzing and structuring our data effectively. Measures of closeness involve quantifying the similarities or dissimilarities between data points, providing insights into patterns and relationships that can significantly influence our model's performance. On the other hand, data clustering techniques aid in grouping similar data points together, helping us identify underlying structures and clusters within our dataset. By understanding these concepts, we lay the groundwork for the subsequent subsections in the Methodology section. In the following subsection, we delve deeper into the specific techniques of measuring closeness, which allow us to gain a comprehensive understanding of the relationships between data points in order to inform our model creation process.

Measures of closeness

While knowing that sales percentages by urbanicity class within the latter stage of the Exploratory Data Analysis were similar, coefficient of variation (or CV) was introduced to help further explain the difference between the data points. We calculated the coefficient of variation (CV) for each store’s twelve sales percentages. By definition, the CV “is a statistical measure of the dispersion of data points in a data series around the mean. The [coefficient] of variation

represents the ratio of the standard deviation to the mean, and it is a useful statistic for comparing the degree of variation from one data series to another" (Hayes, 2022). The formula is the standard deviation of a store's sales percentages divided by the mean of a store's sales percentages. In this context, coefficient of variation allows us to understand how varied the sales percentages are from the mean of each vector. Moreover, since they all sales vectors have the same mean of 1/12 due to each store's sales percentages totaling 100% based on 12 product categories, it means the CV is measuring the variability by standard deviation.

The closer two stores' CV were numerically, the less varied their sales percentages were since the mean is the same for all vectors and its dimensional values are all between 0 and 1. For example, the two stores with the lowest CVs from the entire dataset are shown in Table 3.

Table 3: Two stores with the lowest CV.

Category	Store X	Store Y
1	6.81%	15.8%
2	10.2%	9.97%
3	3.40%	7.76%
4	5.88%	7.76%
5	11.7%	10.0%
6	11.4%	2.86%
7	12.3%	9.28%
8	11.4%	9.09%
9	9.28%	11.3%
10	8.97%	8.16%
11	8.35%	7.79%
12	0.00%	0.00%

To determine further how numerically close each of the sales vectors were, three types of distance metrics were introduced.

- **Euclidean distance (or "L_2 Norm"):** By definition it is "the straight-line distance between two points" (Black, 2004). Based on our sales dataset, its range of values are from 0 to 1, where the closer you are to zero, the more similar the data points are.

    - **Pros:** It is the default distance measure for most clustering algorithms such as K-Means clustering.

    - **Cons:** Due to the high dimensionality of our dataset (i.e., the twelve product category sales percentages for each store), the Euclidean distance has the potential of outputting incorrect distances since naturally it works best with a two-dimensional dataset. The reason for this error potential is because of the Curse of Dimensionality. The Curse of Dimensionality describes the problems that can occur when classifying, organizing, and analyzing high-dimensional data sets (Selman, 2022). It is possible to perform a Principal Component Analysis (or PCA) in order to reduce the number of dimensions; however, in our case, each dimension, or sales percentage per product category, is needed for the analysis and cannot be reduced for the sake of Euclidian's effectiveness.

- **Manhattan distance (“L_1 Norm”):** By definition is “the distance between two points measured along axes at right angles. [It is also called] taxi cab metric, or city block distance” (Black, 2019). Based on our sales data set, its range of values are from 0 to 1, where the closer you are to zero, the more similar the data points are.
    - **Pros:** Compatible with multidimensional vectors.
    - **Cons:** Challenging to integrate this metric into a clustering algorithm that worked with the data.

The graphic below in Figure 6 is a geographical visual representation of the two metrics, where the straight line is the Euclidean distance, and the other line is the Manhattan distance.

Grigorev, 2016

Figure 6: Manhattan versus Euclidean distance.

- **Cosine Similarity:** By definition "is a metric used to measure the similarity of two vectors. Specifically, it measures the similarity in the direction or orientation of the vectors ignoring differences in their magnitude or scale" (Karabiber et al., 2022). The following formula is used:

  Vector 1: Store A's twelve sales percentages as a vector

  Vector 2: Store B's twelve sales percentages as a vector

$$\frac{Dot\ product\ of\ both\ \boldsymbol{Vector\ 1}\ and\ \boldsymbol{2}}{Norm\ of\ \boldsymbol{Vector\ 1}\ multiplied\ by\ Norm\ of\ \boldsymbol{Vector}} = Cosine\ Similarity\ of\ both\ \boldsymbol{Vector\ 1}\ and\ \boldsymbol{Vector\ 2}$$

  Its output is between -1 to +1, where if the formula's output approaches 1, the two vectors are similar and if it approaches -1, the two vectors are not similar.

  - **Pros:** Compatible with multidimensional vectors
  - **Cons:** Challenging to integrate this metric into a clustering algorithm that worked with the data.

In order to understand the output of these metrics based on our data, we selected the two vectors in Table 3. Using SciPy's Euclidean distance function called 'spatial.distance.euclidean', we were able to calculate the Euclidean distance within Python and received an output of 0.142. Then for Manhattan distance, we used SciPy's Manhattan distance function called 'spatial.distance.cityblock' and received an output of 0.348. Lastly, for Cosine Similarity, we used Python's NumPy library to calculate the dot product and norm of both stores to receive a Cosine Similarity of 0.898. Due to the two vectors having the lowest CVs in the data set, this indicates that the closer the distance metric outputs for other comparisons are to the values

above, the closer other comparison's sales percentages are to each other. The sales percentages for vectors were then plotted onto a radar chart in Figure 7.

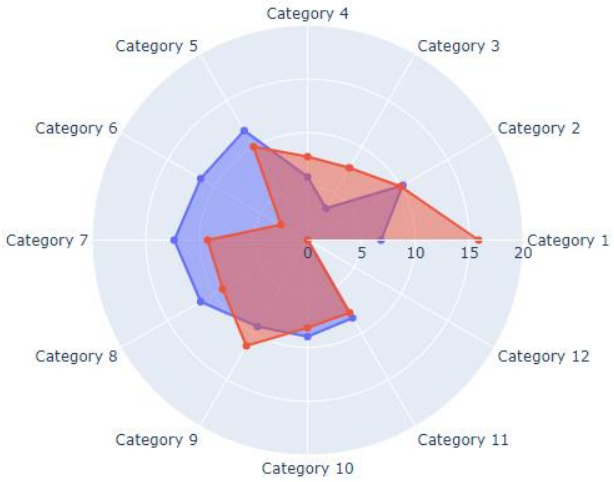

Figure 7: Radar chart of two stores with lowest CV.

The radar chart above demonstrates visually how close these two stores' sales percentages were to one another. The radius is scaled at the range of the sales percentages between the two stores above, while each point of the respected color indicates the sales percentage of the product category. This chart also better interprets the outputs from the distance metrics above since both Euclidean and Manhattan distance outputs were closer to zero, and Cosine Similarity was near to one. This explains why, for most of the categories on the chart, both stores' sales percentages were close. This also shows that even with the curse of dimensionality in place for the Euclidean distance, it was still able to accurately represent the sales-distance when in comparison to the Cosine Similarity metric.

Clustering the data

Our next step was to see how we could incorporate the sales percentage data using this metric via a clustering algorithm. By using a clustering algorithm, it will naturally give us some sort of grouping dependent on the criteria we provide it with. It works by computing the similarity distance between all pairs of data where the most widely used clustering algorithm is K-means; because of its efficiency, effectiveness, and simplicity (Google, 2022). To use it, we had to first figure out the right number of clusters using Python sklearn's Silhouette score metric. This metric is calculated using the mean intra-cluster distance and the mean nearest-cluster distance for each sample (Sklearn, 2023). After implementing it using Python's sklearn metric package, we can see the results in Figure 8.

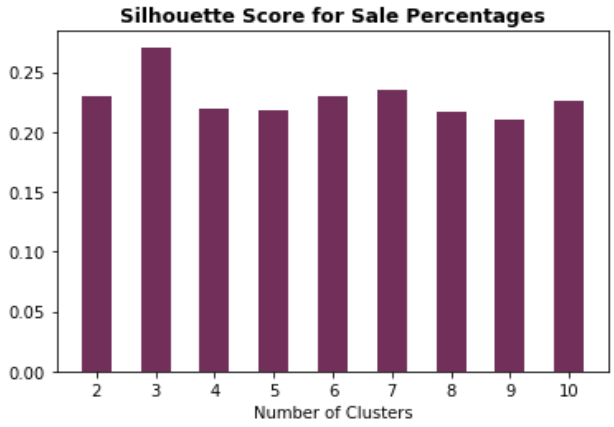

Figure 8: Finding the number of clusters for sales percentages.

According to sklearn's documentation, the best value is 1 and the worst value is -1 which means the greater the silhouette score, the better it will cluster the data (sklearn, 2023). Therefore, according to Figure 8, the optimal number of clusters for this dataset is 3. The sales percentages of all 12 categories of the three clusters are given in Figure 9.

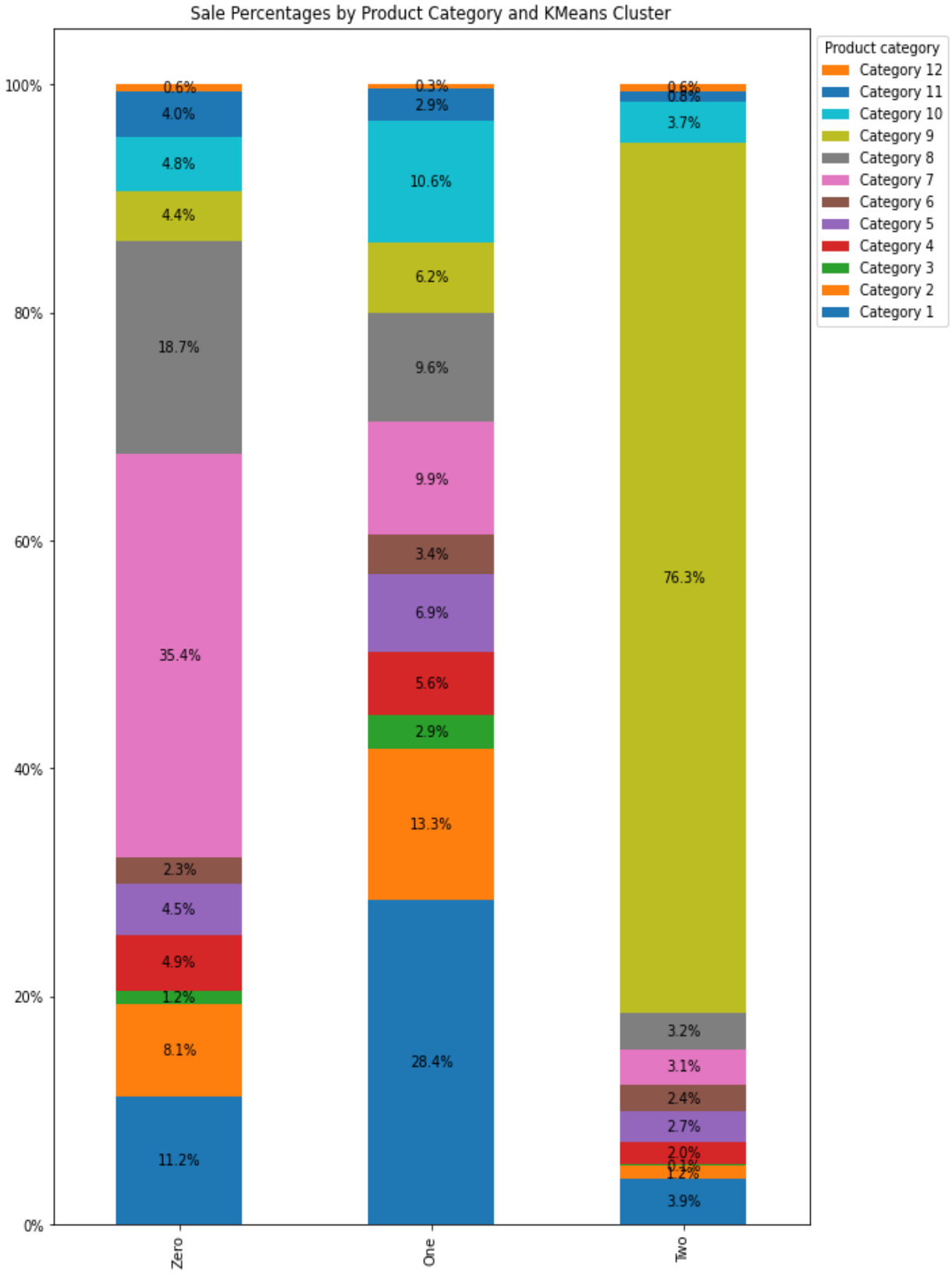

Figure 9: Sales percentage cluster.

The horizontal axis indicates the k-means cluster label, meanwhile the vertical axis tells us the sales percentage by category based on the cluster. It is clear from Figure 9 that Cluster Zero has a large percentage of Product Category 7; Cluster One has a slightly smaller percentage of Product Category 1; and Cluster Two has a large percentage of Product Category 9. Although it gave a visual difference in the sales data, the other categories' sales percentages were still a large proportion. Since the demographic data was tied with a store's main classification number in the consolidated file, we then looked at the sales cluster's demographic percentages. The results are in Figure 10.

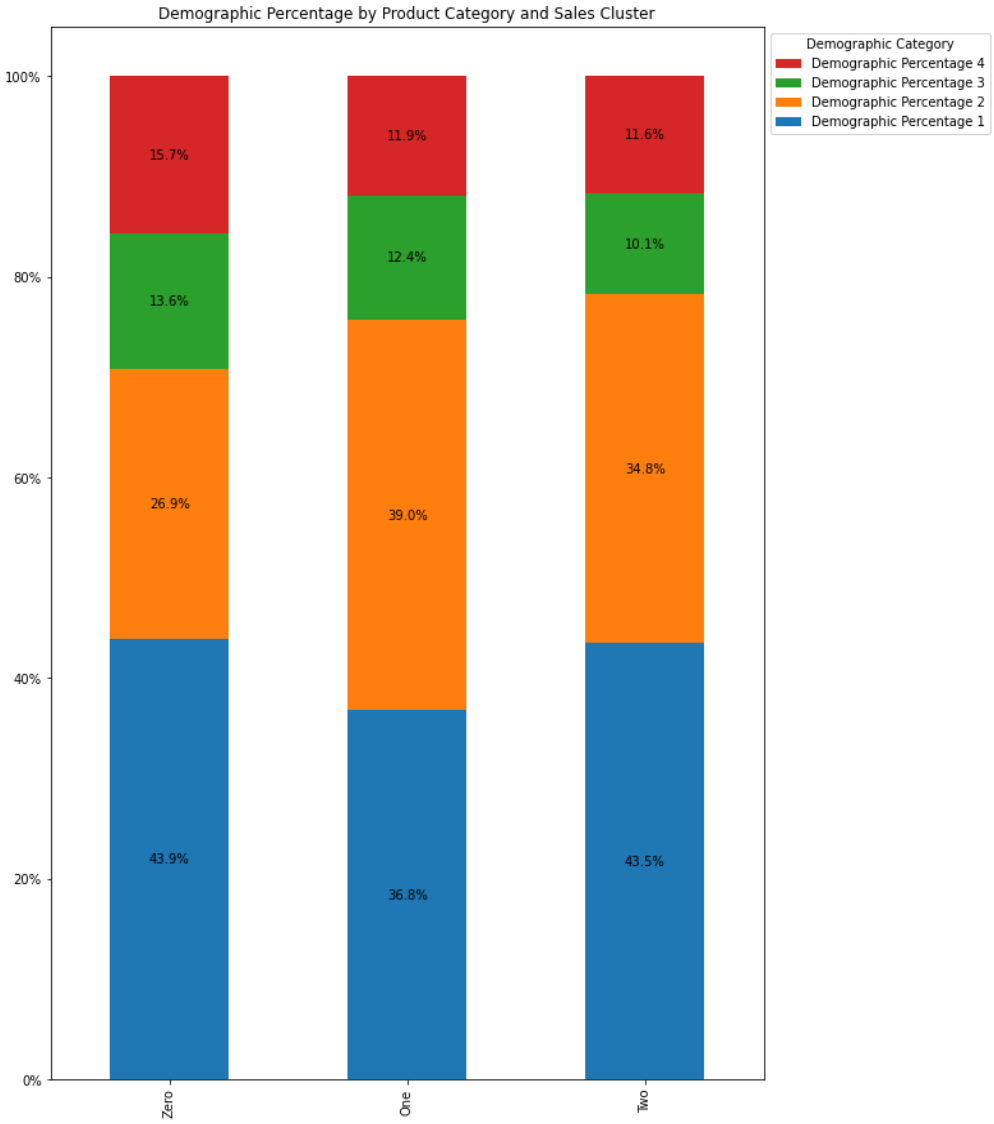

Figure 10: Sales clusters' four main demographics.

As previously shown, we were not able to statistically show that the differences between the cluster sales percentages were significant or not. We proceed with the last type of grouping by demographic percentages. We repeated the same procedure above by first determining the best number of clusters.

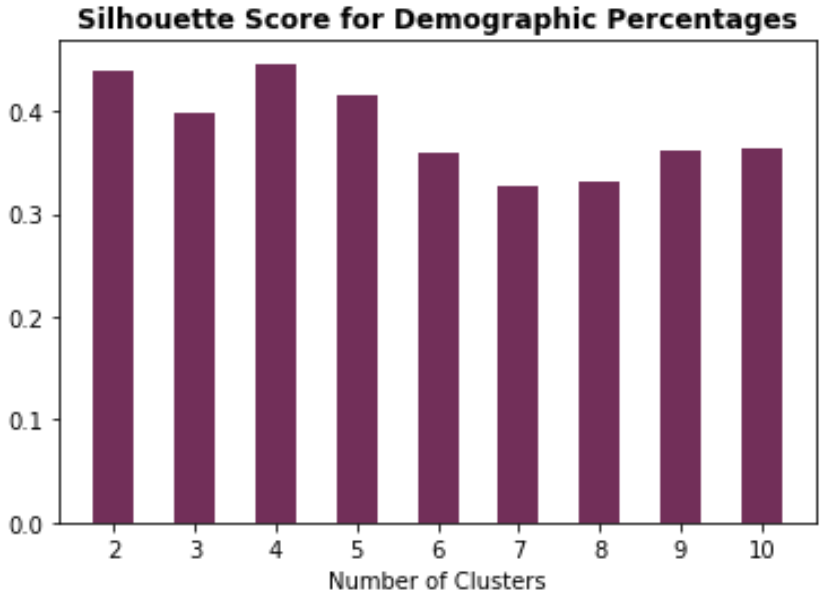

Figure 11: Finding the number of clusters for demographic percentages.

Four clusters were the optimal number of clusters for the demographic percentage data set. When clustering the demographic data set, there was a clear distinction between the different clusters, seen in Figure 12.

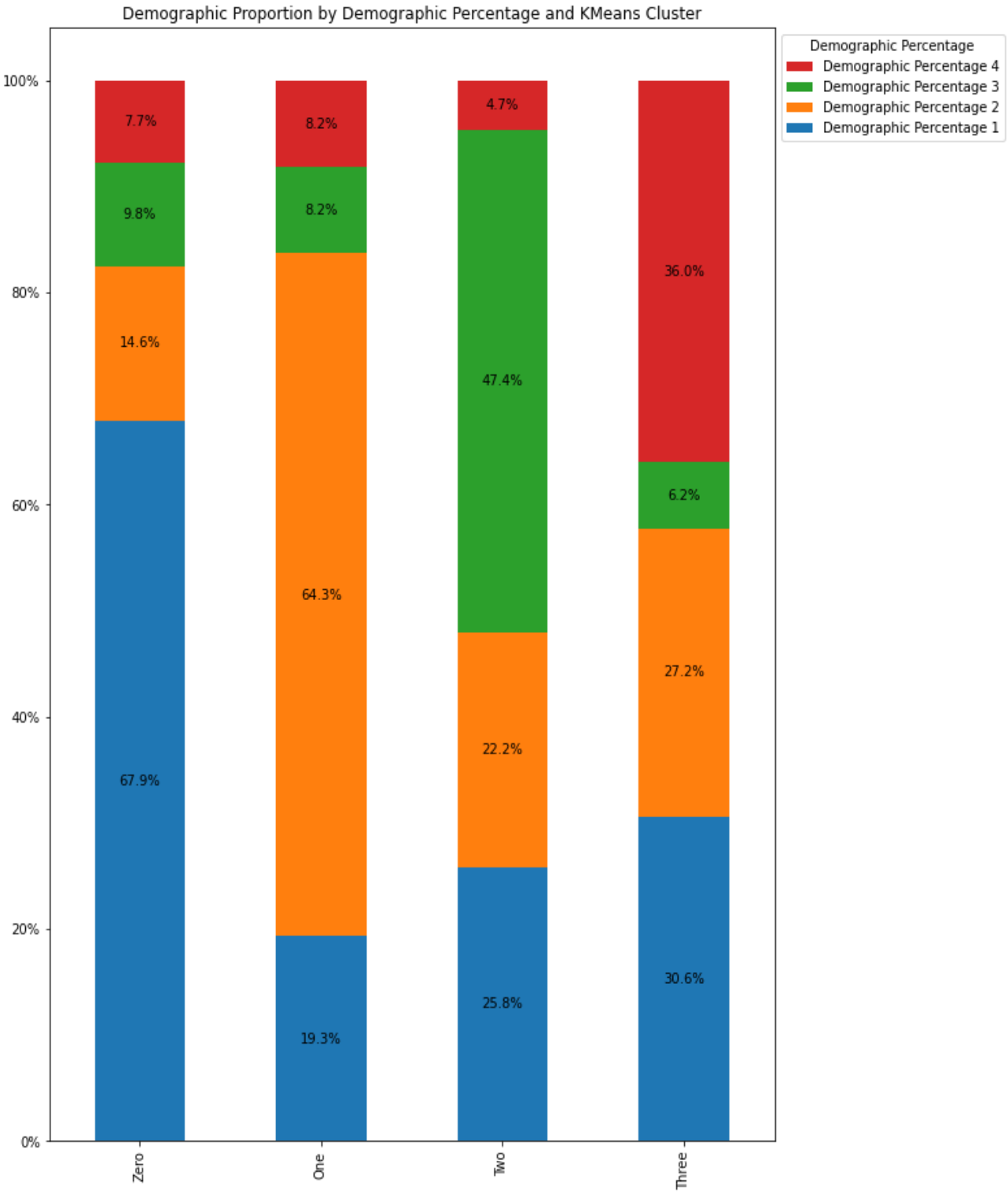

Figure 12: Demographic percentage cluster.

Cluster Zero has a large percentage of Demographic Percentage 1; cluster One has a large percentage of Demographic Percentage 2; cluster Two has a large percentage of Demographic Percentage 3; and cluster Three has a moderately large percentage of Demographic Percentage 4. This gives reason as to why the optimal number of clusters was four from the demographic silhouette score metric since the clustering algorithm clustered the demographic data by the most common demographic (Demographic Percentage 1 to 4). Although the algorithm did cluster by the most common demographic, it did help understand a type of demographic classification for each of the stores. Due to this fact, we analyzed the stores further by looking at the sales of each of the demographic clusters, which is presented in Figure 13.

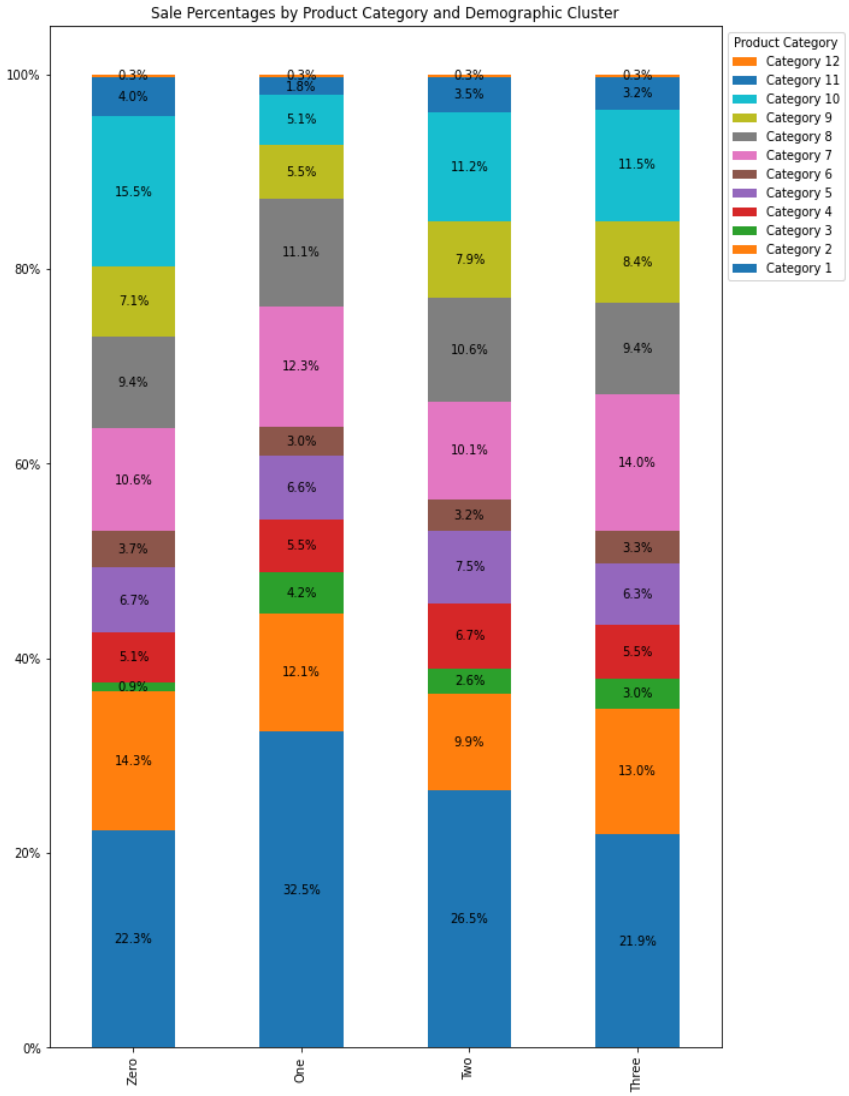

Figure 13: Demographic clusters' sales

Final Model

While the sales percentages in Figure 13 visually showed similarity, the initial demographical clustering of the latter grouping brought greater interest to the company than clustering by sales percentages. The model was then theorized to begin with a store's classification number input and a product/category in mind, demonstrated in Figure 14. By inputting a store and a product (or category) into the tool, a cosine similarity between the input stores' sales percentages and the rest of the data set's stores' sales percentages is computed. The top five stores with the greatest cosine similarity metric were considered 'sales-similar' to the input store. There is a demographic cluster check to ensure that the similar stores could also be demographically similar by existing in the same demographic cluster as the input store. If they do coexist in the same cluster, then the store is both sales-and-demographic similar. We then went back to the consolidated file and extracted the state where both the input store and its similar stores resided. To our discovery, some of the model tests showed similar stores to an input store but may reside within a different state as in the example in Figure 15.

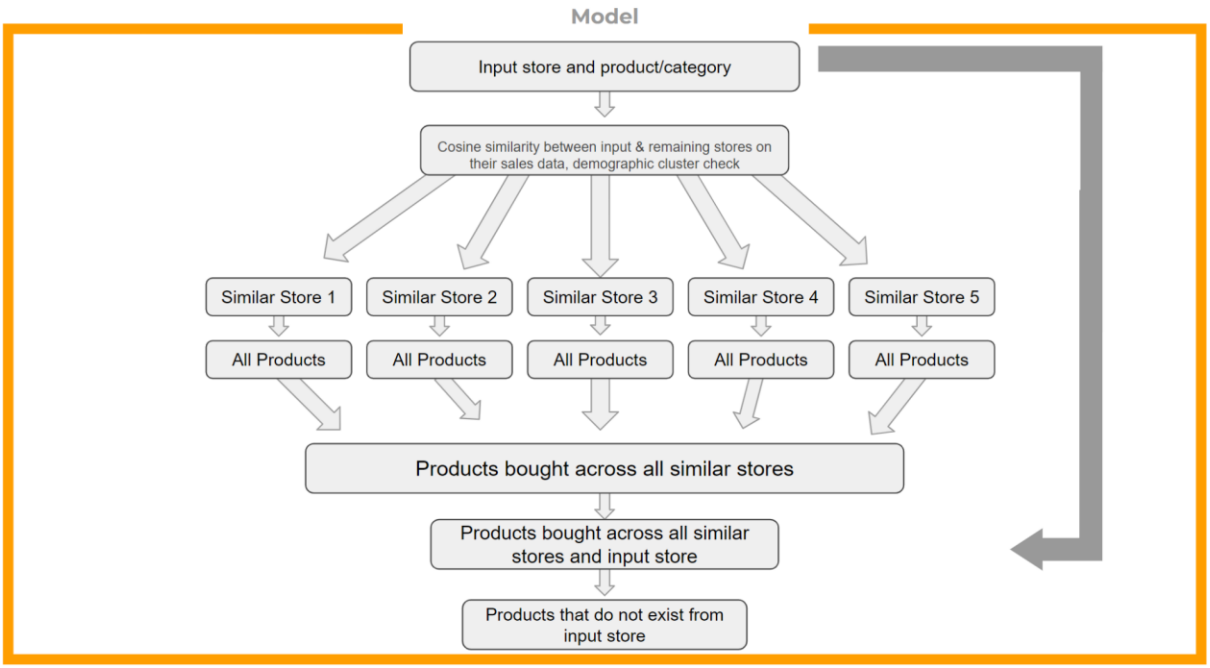

Figure 14: Final model

Results

Store A exists in Demographic Cluster Two in state X.

The following stores are similar based on sales from store A:

Store B exists in Demographic Cluster Two in state X.
Store C exists in Demographic Cluster Two in state X.
Store D exists in Demographic Cluster Two in state X.
Store E exists in Demographic Cluster Two in state X.
Store F exists in Demographic Cluster Two in state Y.

Figure 15: Example of initial output of the model.

Store A exists in Demographic Cluster Two residing in state X along with four sales-and-demographic similar stores. Although the last store resides in state Y, we proceeded with the analysis, since store F in state Y could provide different insights and products. Therefore, according to the model, we now had five similar stores in order to extract all of the similar store's product data. Creating a Python-Excel connection allowed us to interact with the OLAP cube's Pivot Table to automatically extract each similar store's product data. The product names were then extracted along with their sales ranking, derived from ranking the best-selling products (rank 1) to its least selling (rank N). From all the products across all the similar stores, we wanted to see the number of items that sold across all five. The first 24 products that are sold across all five similar stores from the example above are in Figure 16.

Product	Product category	Store 1	Store 2	Store 3	Store 4	Store 5
Product 1	Category 5	275	259	241	250	233
Product 2	Category 5	143	248	193	174	167
Product 3	Category 5	66	54	149	47	49
Product 4	Category 5	209	262	260	264	262
Product 5	Category 5	205	120	136	212	153
Product 6	Category 5	201	101	243	160	316
Product 7	Category 5	237	211	185	114	236
Product 8	Category 5	264	225	284	57	294
Product 9	Category 1	61	27	28	62	123
Product 10	Category 1	126	246	64	143	175
Product 11	Category 1	155	236	221	184	195
Product 12	Category 1	133	221	177	186	122
Product 13	Category 1	145	162	68	123	276
Product 14	Category 1	137	212	196	190	98
Product 15	Category 5	37	40	37	48	79
Product 16	Category 5	99	151	63	83	176
Product 17	Category 8	172	43	109	63	97
Product 18	Category 8	20	6	39	8	14
Product 19	Category 8	90	105	184	106	141
Product 20	Category 1	117	130	71	191	193
Product 21	Category 1	150	177	115	142	143
Product 22	Category 1	17	19	12	15	7
Product 23	Category 1	38	30	57	27	8
Product 24	Category 1	220	214	85	193	77

Figure 16: 24 out of the 164 products sold in all 5 similar stores.

The first column contains the name of a similar product - which has been masked - followed by the category and its ranking across all five similar stores. The closer the sales rank is to one, the better performing the product is to the store. The sales ranking inherently gives us the information that we need to inform ourselves on how well these similar products are performing in their respective stores.

The last step of the model was to extract the products that do not exist within the input store. By removing the items that exist in the input store from the items that exist in all similar stores, we were left with items that do not exist in the input store. The items that do not exist in store A of Figure 15's example is in Figure 17.

Product	Product Category	Store 1	Store 2	Store 3	Store 4	Store 5
Product 1	Category 5	275	233	241	250	233
Product 4	Category 5	209	262	260	264	262
Product 5	Category 5	205	153	136	212	153
Product 10	Category 1	126	175	64	143	175
Product 11	Category 1	155	195	221	184	195
Product 12	Category 1	133	122	177	186	122
Product 13	Category 1	145	276	68	123	276
Product 19	Category 8	90	141	184	106	141
Product 21	Category 1	150	143	115	142	143
Product 60	Category 11	158	287	143	192	287
Product 61	Category 3	92	205	145	78	205
Product 66	Category 2	12	20	4	19	20
Product 70	Category 2	45	66	249	82	66
Product 73	Category 2	2	4	1	1	4
Product 74	Category 3	49	100	102	65	100
Product 75	Category 3	190	185	126	124	185
Product 79	Category 3	309	260	219	255	260
Product 83	Category 7	226	329	265	201	329
Product 85	Category 7	132	211	192	162	211
Product 87	Category 7	63	39	88	40	39
Product 96	Category 1	134	60	134	120	60
Product 97	Category 5	281	137	138	232	137
Product 99	Category 4	229	280	247	214	280
Product 103	Category 10	211	194	214	206	194

Figure 17: 24 out of the 52 products nonexistent in input store A.

The first insight from Figure 16 was Product 73's sales ranking is low across all similar stores; it was also an item nonexistent in the input store. This made Product 73 a potential product replacement for any replacement request in store A's Product Category 2. Additionally, the CPG may want to consider selling Product 73 in Store A. Prior to this tool, this same scenario would rely solely on intuition, making it difficult to understand if the chosen recommended product would be successful. The tool minimizes this uncertainty by showing us which product should replace another product, along with its respective category and each of its sales ranking by similar store.

Discussion

The primary goal of this thesis was to discover the possibility of finding a natural grouping within any of the three data sets. For the geographical data set, despite each analysis conducted, the geographical urbanicity of each store did not provide useful results. Trying to cluster the entire data set by sales percentages did show some grouping but it was difficult to verify conclusively. Lastly, we tried to cluster the data by the four main demographic percentages. The silhouette score metric recommended the best number of clusters to be four, but this was expected since the clustering algorithm was only clustering the demographic data by the four main demographics. However, this made it easier to implement a classification by demographic of each store for the final model. The sales percentages for all the demographic clusters did show some similarity visually, so therefore we used the demographic clustering as an information item in the final model.

The final model consisted of inputting a store into the tool to find the most similar stores. First, the model checks the sales distance between the initial store and the rest of the data set to find the top five most sales-similar stores. After finding the most sales-similar stores, they are

checked for their demographic cluster and if they reside within the same cluster of the input store, they are both sales-and-demographically similar. There exists the possibility of having a similar store in both sales and demographics, but also exist in a different geographical state. This indicates that it is possible to have an item that replaces other items across multiple states due to their similarity.

Each similar store's products are then extracted from the OLAP cube using a Python-Excel connection, which was made to automate the data extraction process from Excel's Pivot Table tool. The items were then filtered to output products that were bought by all five similar stores. This helps us see what items are being bought across all five similar stores and its sales ranking in order to distinguish the best-selling 'similar' product. By combining the similar stores and the input stores' products, it gave us the opportunity to acquire a list of items that do not exist within the input store. To determine this, had to remove the similar items that included the input store from the first similar item list. Therefore, its output only contains items that exist within the five similar stores but do not exist in the input store. Combining the name of these nonexistent products with their respective category, in addition to their sales ranking across all five similar stores, gives us the option to analyze the product's sales rankings and determine what products are suitable as replacements.

Future Work

Upon reviewing the tool, we considered the tool's future work to be expanding the data sets currently used in order to increase the tool's prediction accuracy. The demographic data that consisted of four main demographic percentages could be broken down further to create a better understanding of the data's influence on branched demographics. Consequently, this would affect the demographic clustering used in the final model, but it would provide a better

understanding of the demographic clusters. In terms of the urbanicity classes' definitions, we decided it would be best to further divide the three urbanicity classes into five (i.e., urban, semi-urban, suburban, semi-rural, rural). There are different metrics the company's GIS could provide to accurately define these new classes with the help of the ZIP code population data set.

In terms of the sales data set, another piece of information that could be helpful for the company's decisions would be the footage size of the company's presence in a store. By having this information available, it would help to further understand if each of the five similar stores are also similar by their footage size. Having stores that are similar based on their sales, demographics, and footage size also gives insight on the number of products being bought by a store. Knowledge of the number of products being bought by a store also informs us of the reasoning for a potential product replacement.

Additionally, footage size is a necessary piece of data to better the tool's output; however, acquiring the footage size is difficult. This is because of the footage size's volatile nature (i.e., footage size could be increased or decreased for a store last minute). However, if the footage size is acquired effectively and efficiently, it could provide more accurate results. Another piece of data that could be provided in the output would be a product's information such as its size and price. Information on each of the products could heavily influence the company's decision on whether or not an item is suitable to be a replacement product.

Lastly, finding a statistical test that will validate whether or not the sales or demographic percentages' difference between the urbanicity class or cluster is statistically significant to each other would provide additional support for a clustering approach. The list of future work for the tool leads to a more refined data-driven product replacement process.

References

Black, P. E. (2004, December 17). Euclidean distance. Retrieved April 26, 2023, from https://xlinux.nist.gov/dads/HTML/euclidndstnc.html

Black, P. E. (2019, February 11). Manhattan distance. Retrieved April 26, 2023, from https://xlinux.nist.gov/dads/HTML/manhattanDistance.html

Chen, T., Choi, M., Henstorf, B., Jacobs, J., & See, E. (2021, September 23). The new marketing model for growth: How cpgs can crack the code. McKinsey & Company. Retrieved April 11, 2023, from https://www.mckinsey.com/capabilities/growth-marketing-and-sales/our-insights/the-new-marketing-model-for-growth-how-cpgs-can-crack-the-code#/

Contributors, G. P. (2023). *Welcome to GeoPy's documentation!* Welcome to GeoPy's documentation! - GeoPy 2.3.0 documentation. Retrieved April 15, 2023, from https://geopy.readthedocs.io/en/stable/

Cromartie, J. (2019, October 23). *What is rural?* USDA ERS - What is Rural? Retrieved April 11, 2023, from https://www.ers.usda.gov/topics/rural-economy-population/rural-classifications/what-is-rural.aspx

Google. (2022, July 18). Clustering algorithms. Google. Retrieved April 27, 2023, from https://developers.google.com/machine-learning/clustering/clustering-algorithms

Grigorev, A. (2016). What is the difference between Manhattan and euclidean distance measures?. Quora. https://www.quora.com/What-is-the-difference-between-Manhattan-and-Euclidean-distance-measures

Hayes, A. (2022, September 16). Co-efficient of variation meaning and how to use it. Investopedia. Retrieved April 26, 2023, from https://www.investopedia.com/terms/c/coefficientofvariation.asp

Karabiber, F. (2022). Cosine similarity. Learn Data Science - Tutorials, Books, Courses, and More. Retrieved April 26, 2023, from https://www.learndatasci.com/glossary/cosine-similarity/

Mann, M., Chao, S., Graesser, J., & Feldman, N. (2023). *Python Open Source Spatial Programming & Remote Sensing*. Accessing OSM Data in Python - Python Open Source Spatial Programming & Remote Sensing. Retrieved April 15, 2023, from https://pygis.io/docs/d_access_osm.html

McCain, A. (2023, March 1). *26 stunning big data statistics [2023]: Market size, trends, and facts*. Zippia. Retrieved April 9, 2023, from https://www.zippia.com/advice/big-data-statistics/

McDonough, M. (2023, April 28). ANOVA. Encyclopaedia Britannica. https://www.britannica.com/topic/variance-analysis-statistics

Mackenzie, R. J. (2021, November 26). *One-way vs two-way ANOVA: Differences, assumptions and hypotheses*. Informatics from Technology Networks. https://www.technologynetworks.com/informatics/articles/one-way-vs-two-way-anova-definition-differences-assumptions-and-hypotheses-306553

NIST, N. (Ed.). (1989). *Chi-Square Goodness-of-Fit Test*. 1.3.5.15. Chi-square goodness-of-fit test. https://www.itl.nist.gov/div898/handbook/eda/section3/eda35f.htm

Nominatim Overview. NOMINATIM 4.2.3. (2023). Retrieved April 15, 2023, from https://nominatim.org/release-docs/latest/

Selman, H. (2022, June 15). Machines are haunted by the curse of dimensionality. Dataconomy. Retrieved April 24, 2023, from https://dataconomy.com/2022/06/15/curse-of-dimensionality-machine-learning/

Sklearn. (2023). Sklearn.metrics.silhouette_score. scikit. Retrieved April 27, 2023, from https://scikit-learn.org/stable/modules/generated/sklearn.metrics.silhouette_score.html

Thorn, J. (2021, September 26). The surprising behaviour of distance metrics in high dimensions. Medium. Retrieved April 25, 2023, from https://towardsdatascience.com/the-surprising-behaviour-of-distance-metrics-in-high-dimensions-c2cb72779ea6

Trevino, A. (2016, December 6). Introduction to K-means Clustering. Oracle. Retrieved April 26, 2023, from https://blogs.oracle.com/ai-and-datascience/post/introduction-to-k-means-clustering