For office use only
T1 ________________
T2 ________________
T3 ________________
T4 ________________

Team Control Number
2019330

Problem Chosen
C

For office use only
F1 ________________
F2 ________________
F3 ________________
F4 ________________

2020
MCM/ICM
Summary Sheet

(Your team's summary should be included as the first page of your electronic submission.)
Type a summary of your results on this page. Do not include the name of your school, advisor, or team members on this page.

As e-commerce increasingly becomes the primary mode of American consumerism, analytical insights into platform-level success have become the Age of the Internet's iteration of strategic management. For this project, we were provided Amazon data for three product types: microwaves, hair dryers, and pacifiers. We cleaned each dataset to ensure ease of review sentiment analysis. We then extracted product characteristics from the product titles and engineered boolean features for them. The cleaned data was then used to investigate seasonality and other time-series characteristics that might have occurred in our data.

After time-series analysis, we used a combination of supervised and unsupervised learning methods to discover the most important product characteristics and how these relate to product ratings. The learning methods used include linear regression, gradient descent, sentiment analysis, and clustering models. We conducted a sensitivity analysis to confirm that our conclusions correspond to the real world and that results are both scalable and repeatable.

From these models, we were finally able to extract meaningful insights for each product with respect to both marketing and product design. These insights allowed for the crafting of an ideal product, with features and a meaningful listing name derived from data-driven results. Along with identifying optimal product design, we developed lists of terms and trends in user behavior to earliest identify a successful or failing product.

Dear Sunshine Company Marketing Director,

We've been tasked to utilize consumer online reviews to both inform your sales strategy and help identify specific product characteristics that consumers want to see. We understand that the Sunshine Company is currently developing plans to sell three new products: a pacifier, a hair dryer, and a microwave. We created a series of models such as text-based sentiment analysis and clustering to address the unstructured text format, and believe we have enough insight to provide you with the ideal set of products that will really shine.

Before jumping straight into recommendations, we wanted to give an overview of what consumer purchasing trends in the three product markets looked like. From time-based measures, we can identify points throughout the year when consumers most often leave reviews. Because these reviews are usually verified purchases, we are confident that the uptick in reviews at certain points in time accurately reflects consumer purchasing behavior. However, the only time-based relationship we observed in the data is a direct relationship between the review of products and consumer purchasing trends. This demonstrates that time is not a very good measure of product feature success.

For each product, we have created for you an ideal product name, information on what features the product should contain, and a list to track its success and mitigate failures. We have opted to keep the Sunshine brand name in every product as we noticed that reviews containing the word "Shine" averaged a high star rating and had positive sentiments, an indication of smart branding!

Pacifiers The ideal title character count for pacifiers falls within the 40-60 character range. The pacifier should aim to be BPA-free, graspable for the baby, soft, and gender neutral. Animals are the highest predictor of a high star rating and should also be included in some way. Success metrics to track would be the presence of these words within reviews: son, daughter, month, hold, cute, or love. Failing measures are: hard, nose, boy, disappointed, expected, wrong, and bad. To maximize consumer confidence and provide keywords for SEO, the ideal product name for a Sunshine pacifier would be: "Sunshine 0+ months pacifier bpa-free, many animals designs".

Hair Dryers The ideal title character count for hair dryers sits at 60 characters. The ideal dryer has ionic tendencies, finds a balance between power and noise level, and is incredibly durable and safe. Tourmaline is the ideal material due to its ionic properties and light weight. Success measures follow the presence of: love, powerful, easy, convenient, fast, and quickly. Failing measures are: waste, disappointed, return, warranty, died, and heat-related words(fire/burned/sparks/hot). To maximize consumer confidence, the ideal product name for a hair dryer would be: "Sunshine Ionic Pro – Quiet and Powerful, Light and Portable".

Microwaves There is no ideal character count for microwaves. Consumers who purchase microwaves consistently look for size, effectiveness, ease of use, and the overall wattage. Microwaves should be powerful and durable at the same time. Success measures of microwaves are: easy, simple, fits, compact, and effective. Failing measures are: junk, errors, repair, fire, broke, warranty, stopped, and lasted. The ideal name for a Sunshine microwave is: “Sunshine Perfect Fit – Powerful, Easy Clean Microwave.

With all of these product recommendations, we recommend continuing to monitor and adjust your products using data from reviews. This would not only allow Sunshine to consistently monitor product reviews for those success and failure measures, but also would enable the development of new insights into your products and industries as a whole. Using data from reviews helps with effectively navigating consumer emotions and truly pushes forward a product that the customer is asking for.

We hope that you find these recommendations useful and we thank you for using our services.

Sincerely,
Team #2019330

Sell This, Not That!

Identifying Key Product Characteristics through Sentiment Analysis, Regression, and Clustering

---

Contents

1 Introduction **1**
- 1.1 Background . . . 1
- 1.2 Restatement of the Problem . . . 1

2 Assumptions **1**

3 Data Exploration **1**
- 3.1 Data Cleaning . . . 2
- 3.2 Feature Engineering . . . 3

4 Methodology **3**
- 4.1 Sankey Diagrams . . . 3
- 4.2 Time Series Analysis . . . 5
- 4.3 Modeling . . . 6
  - 4.3.1 Linear Modeling with Multiple Regression . . . 6
  - 4.3.2 Sentiment Analysis . . . 7
  - 4.3.3 Clustering . . . 8

5 Analytic Insights **11**
- 5.1 Pacifier . . . 11
- 5.2 Hair Dryer . . . 12
- 5.3 Microwave . . . 14
- 5.4 Other Modeling Insights . . . 15

6 Sensitivity Analysis **16**

7 Model Strengths and Weaknesses **17**

8 Conclusion **18**

9 Appendix **20**

1 Introduction

1.1 Background

Product reviews and ratings provide quantitative and qualitative feedback pertaining to the reception of their products. If these untidy reviews and ratings can be transformed into meaningful insights, companies can design and market new products better aligned with what customers actually want. We were provided review and rating data for three different products: microwaves, hair dryers, and pacifiers, in which each product was rated and reviewed by real customers. These reviewers are able to mark reviews as helpful, be verified for their item purchase, and distinguished as a Vine voice through longitudinal providing of thoughtful product reviews. We modeled these features and others created throughout the duration of the competition to determine relative importance in the success or failure of product brands.

1.2 Restatement of the Problem

We were tasked with discovering key design features that could predict the success or failure of a product. We calculated a sentiment score for each product and review and compared these scores against ratings over time. We also performed sentiment analysis on review contents to find the words that corresponded to good and great reviews. These words could subsequently be associated with tangible or intangible design features.

2 Assumptions

Within this solution we are making the following assumptions:

- All reviews are indeed made by real amazon consumers
- The individual reviewer's opinions are treated as factual and correct
- There is innate subjectivity in all reviews that cannot be completely accounted for
- Unstructured text cannot be transformed into a fully clean dataset within a short time period
- Sentiment measures do not yet fully comprehend natural language and its complexity
- Online reviews follow a bimodal distribution rather than a normal one, where highest density occurs on the extremes of 1 and 5 with a general skew towards positive reviews [1]

3 Data Exploration

We were presented with three datasets for this competition: hair_dryer.tsv, pacifier.tsv, and microwave.tsv. No outside datasets were used during the data cleaning and engineering process. The initial for the datasets were as follows: hair_dryer.tsv contained 11,470 rows and 15 columns; pacifiers.tsv contained 18,939 rows and 15 columns, microwave.tsv contained 1,615 rows and 15 columns. All columns were congruent across each dataset.

We utilized the statistical programming language R for all steps in the data science pipeline, including data cleaning, exploratory analysis, modeling, and sensitivity analysis.

3.1 Data Cleaning

The cleaning process was generally similar across all datasets. We first decided to remove all reviews that were unrelated in product, eg. other baby toys or formula in pacifiers, flat irons and beauty supplies in hair dryers, and microwave cavity paint in microwaves. In order to do this we identified key words within each dataset that signified a certain product and filtered out the rows that did not contain that data through string matching.

Much of the cleaning process was spent cleaning unstructured text in the form of user reviews. User reviews contain significant inherent difficulty for modeling, as incongruity exists in word choice for the same meaning (100%, 10/10, one-hundred percent, etc.). There are many characters present that created errors in exporting the data, such as quotation marks encoded as 34; and line breaks encoded as
. The text also contained hyperlinks to other Amazon products and video or image links that proved challenging to systematically remove .

The brunt of issues encountered in cleaning appeared in the form of odd characters, typos, unique forms of emotional expression, and reviews in different languages. These issues were addressed via grep() and gsub() functions in combination with regular expressions to remove or replace all pattern matches. For example the following regular expression finds instances of letters occurring in tandem 3+ times ("ooo," "aaaaaaa") and replaces them with a single version of the characters:

```
dryer$review_body<-gsub("([[:alpha:]])\\1{2,}","\\1",dryer$review_body)
```

In some portions of the data, it was easier to manually transform the unstructured text. For example, two reviews of the hair_dryers data included the word "good" as "gooo o o oo oo o o o ooo o od". The infrequency of occurrence and difficulty of regular expressions in detecting abnormally spaced values like these led us to re-code them by hand to "good".

Reviews that appeared in Spanish were not translated in full due to their infrequency and the absence of a coherent R language translator for the appearing cases. We translated specific words that would have an effect on the overall sentiment of the product review such as "bueno" to "good", "muy" to "very", and "excelente" to "excellent". The rest of the review body would be ignored by the sentiment analysis model.

We acknowledge that the methodology of cleaning we used will not clean all text cells to 100%. Given the duration of the competition and the significant amount of energy required to perfectly clean the reviews, we exerted due diligence given the problem's time constraints.

3.2 Feature Engineering

We augmented the existing datasets by engineering boolean values that would assist in linear modeling and sentiment analysis. To do so, we identified unique key terms and categories pulled from the product_title fields whose occurrences were documented in separate fields were then coded into new boolean values. A full list of the title-generated features is provided in the Appendix's Data Dictionary.

We made the decision to combine the review_headling and review_body columns into a singular column titled user_experiece. Our rationale for doing so was that we wanted to understand the underlying sentiment of user feedback. Both features were reflective of the feedback and oftentimes said similar things. Condensing the columns allowed us to more efficiently perform sentence and word level sentiment analysis over the user-provided feedback.

4 Methodology

4.1 Sankey Diagrams

We built sankey diagrams to identify how product ratings change over time. On the left column we placed the proportion of 1-5 star reviews for the first year that the product was sold. On the right column, we placed the proportion of 1-5 star reviews for the last year that the product was sold. The curves in between the bars show what proportion of each rating stayed the same or changed across the interval.

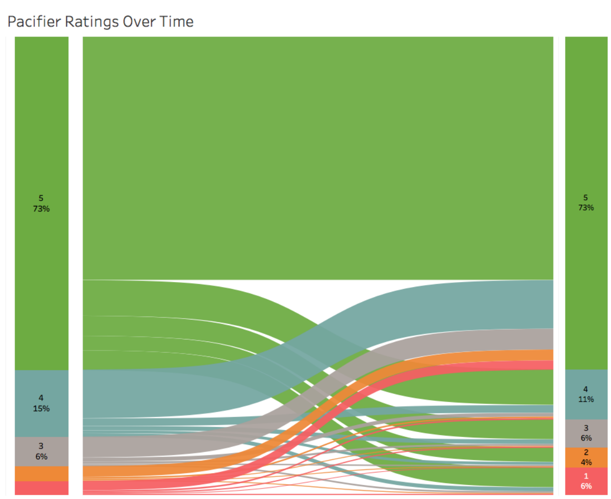

Figure 1: Demonstrates the Pacifier change in product star rating over time

In the Pacifier Ratings Sankey, the proportion of five star reviews has remained constant despite shifts in review rating. At the end of the sale period, 73% of five star reviews had not changed rating since the beginning of the sale period. The rest of the five star reviews (26.91%) came from the other ratings.

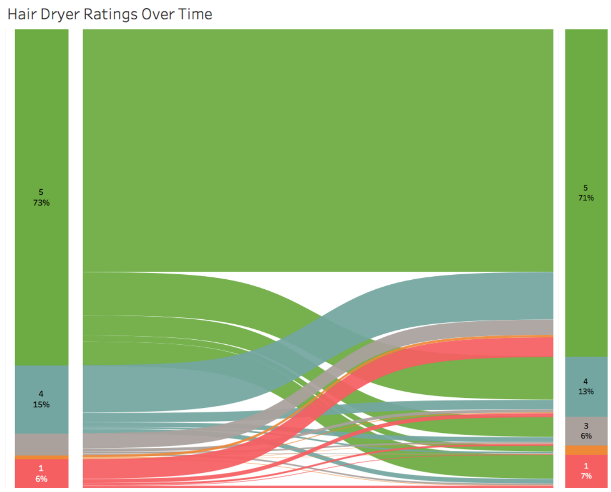

Figure 2: Demonstrates the Hair Dryer change in product star rating over time

In Hair Dryer Ratings Sankey, the rating proportions remained fairly steady over the time interval. Approximately 70% of both 3 and 4 star ratings transitioned to 5 star ratings, while 12.86% of ratings dropped out of the 5 star category. Although there was a lot of movement in ratings, the proportion of ratings generally stayed the same.

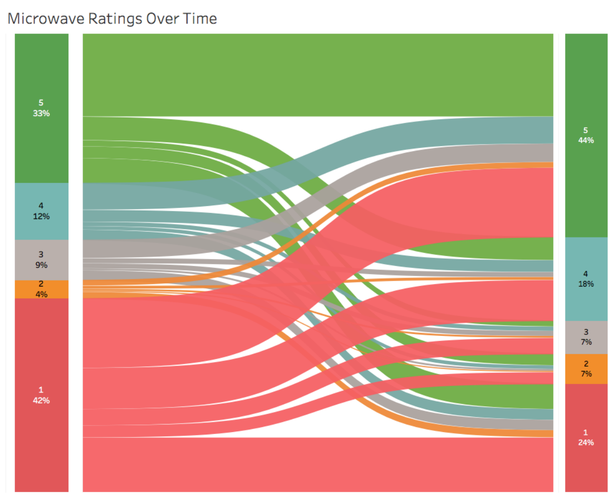

Figure 3: Demonstrates the Microwave change in product star rating over time

In the Microwave Ratings Sankey, we observed that a large proportion of one-star ratings (57.22%) at the beginning of a product's run were pulled up to four- or five- star ratings by the end of the time period. Nearly half of the 5 star ratings and a quarter of the 1 star ratings stayed consistent. The 2, 3, and 4 star reviews retained only 6%, 12%, and 21% of the same reviews respectively come period end.

For all three sankey diagrams, the mean appears to converge at a four-star rating as opposed to a three-star rating. Given our assumption about the polarity of leaving reviews when either an extremely positive or negative experience occurs, a convergence at a four-star rating is aligned with ratings consistently recorded at the maximum star value.

4.2 Time Series Analysis

All product data sets have at least 9 years of data. Earlier years such as 2002 and 2003 have far less data than years like 2013 and 2014. Over these years, we can find strong evidence of seasonality in each product data set.

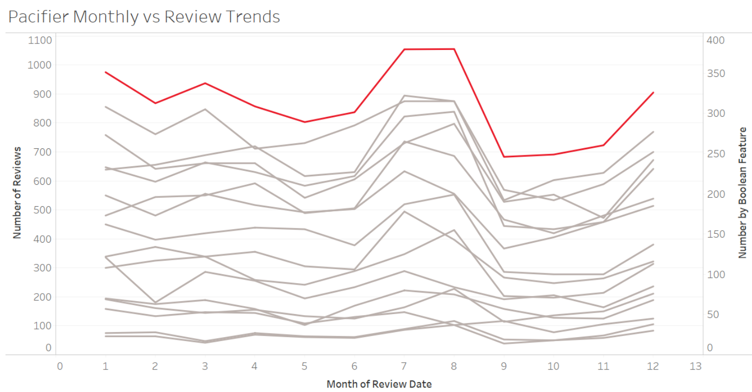

Figure 4: Trend of total number of pacifier reviews versus the month overlayed by specific product features per month

In this visualization of the number of pacifier reviews per month we can see that there is a peak of product reviews in July and August. According to the Center for Disease Control, August is the peak month for births. When the CDC adjusts for seasonality January is the peak month for births.[2] We can see a high number of reviews in January as well. Thus we find clear evidence of seasonality in the pacifier product data. We also compare the number of reviews for all pacifiers per month to the number of reviews for pacifiers containing important product features per month. These are the features that we determined could have an impact on the success of each product title. We see that the reviews per product features per month follows the same seasonality. Every product characteristic seems to follow the same seasonality.

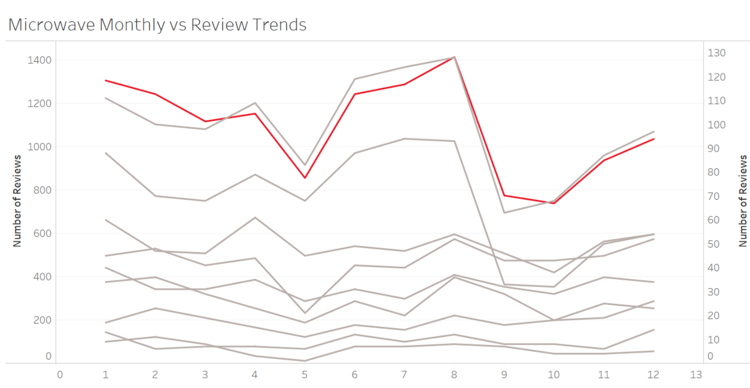

Figure 5: Trend of total number of microwave reviews versus the month overlayed by specific product features per month

The number of reviews in the microwave data also peaks in August. The reviews take on an upwards trend from June to August, but taper off quickly right after. According to movinglabor.com, August is the busiest month of the year for moving.[3] When moving, people may want a new microwave for their new residence which could explain the high peak in reviews during the summer. A high number of reviews in January and February may be explained by microwave Christmas gifts. Like the pacifier data the number of reviews by boolean feature follows the seasonality of the number of microwave reviews.

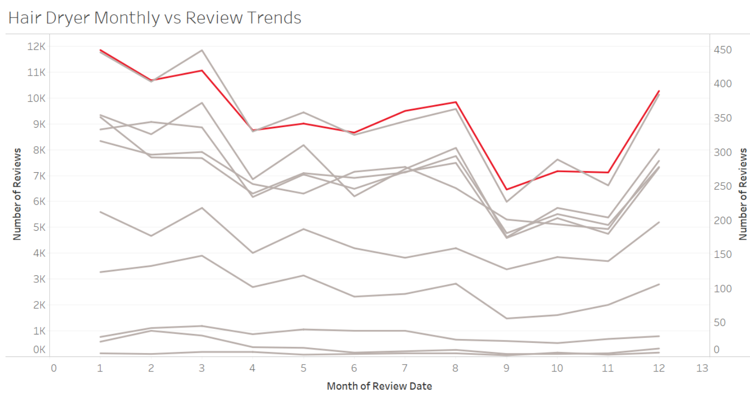

Figure 6: Trend of total number of hair dryer reviews versus the month overlayed by specific product features per month

Hair dryers do not exhibit a lot of seasonality outside of the holiday season. The number of reviews trends upwards during Christmas time due to the volume of hair dryers being bought as gifts. The word Christmas is even mentioned in the review text and on average actually holds a 5 star rating across all reviews. Reviews taper off towards the beginning of the winter season, however. According to an article from the New York Times, cold weather can have a negative effect on hair and we assume that hair dryers can potentially exacerbate the problem.[4] Like microwave and pacifier, hair dryer boolean fields followed the seasonality found with the total number of reviews. Three of these fields do not appear to follow any seasonality.

Because the review trends of all feature booleans closely mimic that of the total number of reviews, time based measures are not very effective in determining whether a product's reputation is increasing or decreasing. The ultimate predictor of review count simply falls to general consumer purchasing behavior trends. Hence we did not pursue the analysis of time-based measures further.

4.3 Modeling

4.3.1 Linear Modeling with Multiple Regression

Multiple regression is one of the most powerful, yet simple, models available out-of-the-box. It generalizes a linear relationship between the target value (y) and explanatory features of the data (x). We used supervised machine learning methods like regression and gradient descent in order to see if star rating could be optimized based on features of the product and product title.[5] Accuracy, in this problem measured using root mean squared logarithmic error (RMSLE) is calculated by measuring the distance between the predicted y value and its actual counterpart:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon \tag{1}$$

Utilizing a basic multiple linear regression model, we regressed on y as star_value, and utilized the boolean features that we developed along with related numeric features like vine, helpful_votes, and verified_purchase as explanatory variables. The data was split into train and test sets using a random sampling of the data at a 75:25 split. Using the pacifier dataset as an example, with all explanatory variables present, the RMSLE of the model was 0.2575517, with the other datasets demonstrating similar values. In an attempt to reduce the error measure further, we employed stepwise feature selection to identify the strongest features. The forward selection was able to slightly improve model performance, with a resultant RMSLE of 0.257479.

Prior to running the regressions, we hypothesized that the strongest models on these datasets would most likely be unsupervised models given the nature and structure of the data. The persistently high RMSLE values provided credence to this hypothesis and allowed us to direct the brunt of effort upon both sentiment analysis and clustering algorithms.

4.3.2 Sentiment Analysis

To prepare for sentiment analysis we made the decision to combine many different word tenses contained in the user_feedback variable. To do so, we utilized a bag of words technique and identified words of similar tenses as well as words of similar meaning. Different permutations of hair dryer such as hair dryer, hairdryer, dryer, and even misspellings such as hair drier and hairdrier were all combined into the blanket term dryer. The same was done with different words such as love, loves, and loved. This provided a more significant word frequency count which had a great effect on our sentiment and clustering methods.

We created two separate sentiment-based measures: a word-level measure and a sentence-level measure. To get sentiment scores on a sentence level, we utilized the R package "sentimentR" to parse the sentences from the user_feedback column and return an overall sentiment score per sentence. We then joined the sentence level scores on review_id to the original datasets.

For word-level analysis, we utilized the bag of words method to transform the user_feedback column into a table of individual words and their counts sans stop words (a, and, the, but, etc.). We then filtered by number of occurrences to limit the amount of results that appeared within our word data as this transformation would pick up words that only occurred once. The hair_dryer dataset was filtered on n >= 100 and contained 326 observations in the words data table. The pacifier dataset was also filtered on n >= 100 and contained 231 observations. The microwaves dataset was filtered on n >= 40 and contained 207 observations in the words table.

4.3.3 Clustering

Clustering is one of the most common forms of unsupervised machine learning used to create groupings in data where labeled groups (such as those in classification problems) exist. Given the unstructured nature of product reviews, clustering by the previously created bag-of-words granular text data allows for a quantifiable relationship to be drawn between frequency of word appearance in user feedback and average product star rating.

Hierarchical Clustering

Utilizing both k-means and hierarchical clustering was integral to understanding trends in user feedback, as both algorithms utilize very different methodologies to determine clusters. Hierarchical clustering compensates for bias in k-selection by allowing the user to manually "cut" the tree at any desired level of k from 1 to number of records (n).

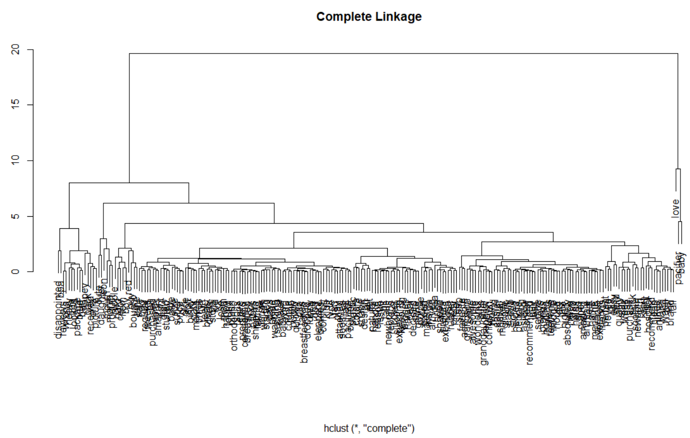

Figure 7: Complete linkage hierarchical cluster of Pacifier

The primary decision is between linkage methods, the process through which the algorithm determines which clusters should be merged at each step. We selected the Complete linkage method, which calculates the shortest distance between clusters by the two furthest elements in the respective clusters furthest away from each other:

$$max\{d(a, b) : a \,\epsilon A, b \,\epsilon B\} \tag{2}$$

Utilizing a secondary clustering method provided additional context into how our existing data could be divided into interpretable categories. However, the only signifying difference between the k-means and hierarchical outcomes was the merging of the final two clusters into a single low-frequency amalgam with the highly rated words split into their own individual clusters. For interpretability purposes, having a grouping which consists of only single values is not helpful in determining trends in consumer behavior. We elected to focus on k-means clustering. We provide the code in the event a more robust bisection of groups in the future warrants the comparison of k-means and hierarchical clustering results.

K-Means Clustering

The k-means approach to clustering uses the sum of squared Euclidean distances between records and its related centroid.[6] Unlike hierarchical clustering, the user must manually set the number of clusters (k) which are used to initialize the number of centroids and perform all subsequent calculations:

$$W(C_k) = \sum_{x_i \epsilon C_k} (x_i - \mu_k)^2 \tag{3}$$

We used the elbow method of determining the k which best optimizes the total withinness sum of squares (detailed further in the Sensitivity Analysis section) as k=4 for microwave and pacifier, and k=3 for hairdryer. Through our set number of configurations (50), the following clusters were created.

According to our analysis, the boundaries of each cluster represent their relative positioning on the product rating-frequency continuum. Doing so allows for both summary-level and word-level insights.

The first (or first two in the case of k=4 clustering) cluster represents high frequency, high average star_rating values which signify a thriving and well-received product. The middle (or third cluster for k=4) represents high-average low-frequency values. These values may prove to be part of the first group upon inclusion of new data, however the provided dataset demonstrated a high average star_rating over an average of less than 1000 records in each dataset. Finally, the final cluster serves as the diametric opposite to the first entry, occurring less-frequently than the centroid mean yet signifying a significantly lower average star_rating when mentioned by users.

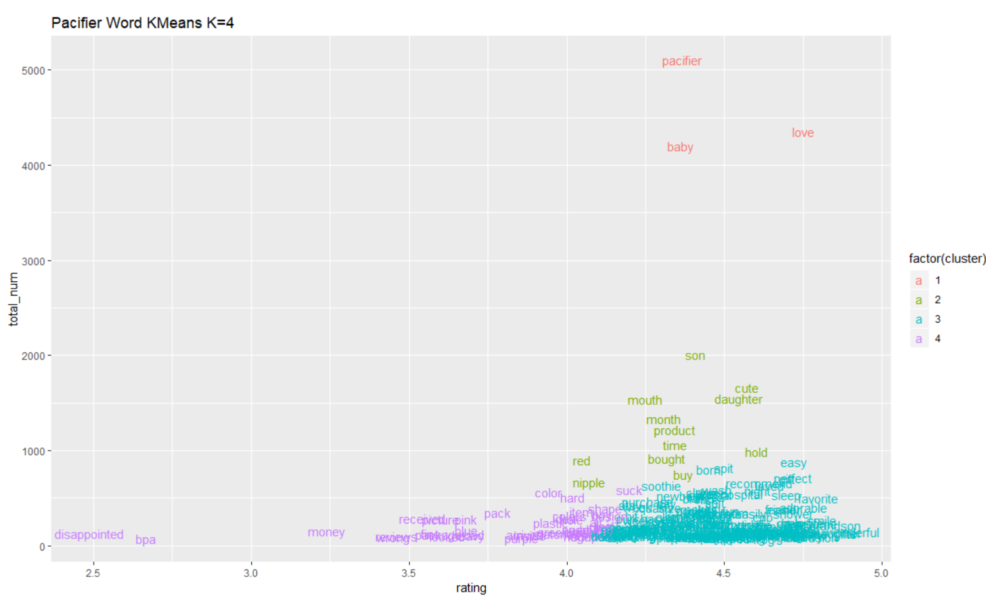

Figure 8: K-Means Pacifier k=4

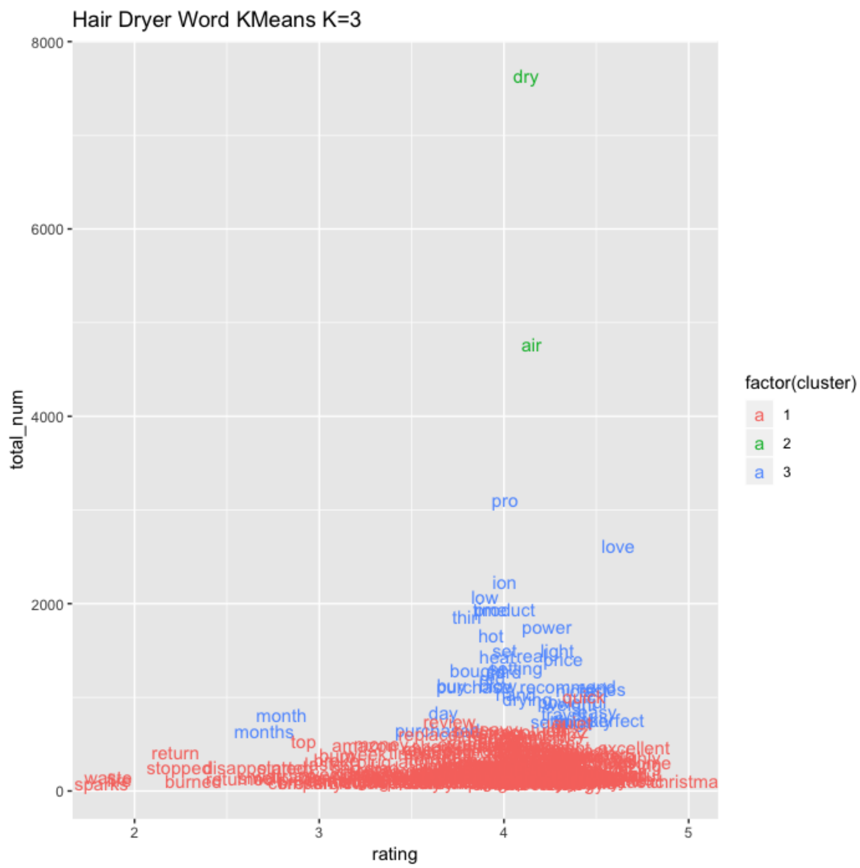

Figure 9: K-Means Hair Dryer k=3

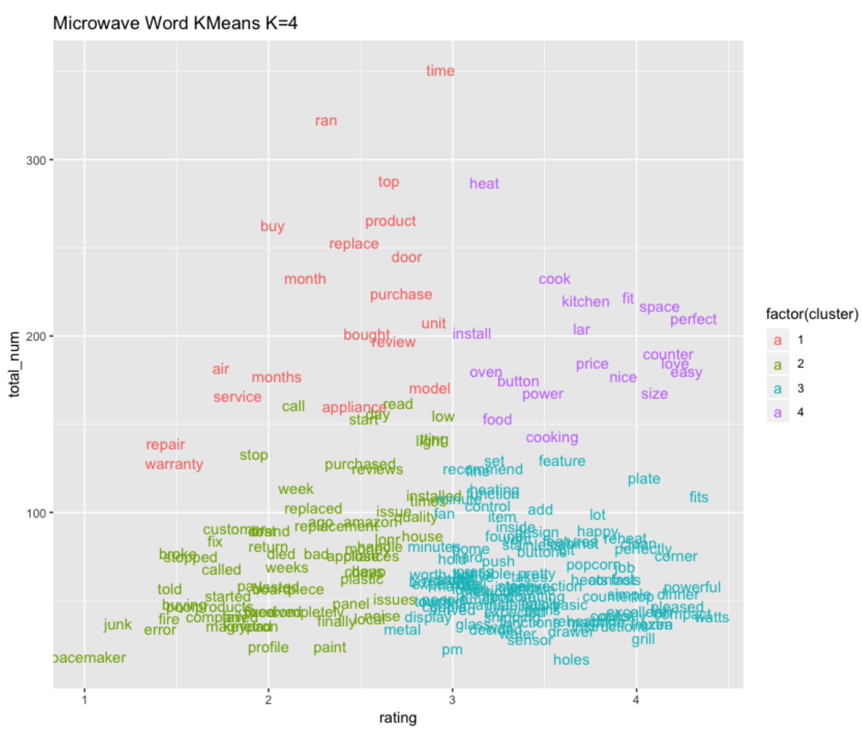

Figure 10: K-Means Microwave k=4

5 Analytic Insights

5.1 Pacifier

Important Insights Found in Modeling

From the models that we have developed, we identified a series of product features that are unique and have a significant impact on the relationship between product features and the overall star value. From the linear regression p values, the most powerful product features were animal designs/attachments, accessibility for various age ranges, and bpa-free products. Animals displayed a confidence interval of 5.00E-18 and 89.95% of animal-related reviews were either four- or five- stars. Due to the sensitivity of premature infants and distinction between stage 1 (babies without teeth) and stage 2 (babies teething or with fully developed teeth), references to specific age ranges also led to a 20.37% increase in star review.[7]

One of the common conundrums pacifier manufacturers encounter is deciding between describing their product as a 'pacifier' or 'nipple,' and the decision to include both values is negatively related to star outcome. When deciding on terminology, 'pacifier' is overwhelmingly the better option, not only for its appearance in 3,502 five-star reviews but also as it prevents confusion from other nipple maternity products such as bottle nipples and nipple pads for new mothers.

The Ideal Product

The ideal pacifier can be described in merely 40-60 characters. As more and more descriptive adjectives and search engine optimization (SEO) key words are added to product titles, enthusiasm from reviewers wane. Similarly, the economic focus on organic goods particularly with newborn children whose immune systems have just developed has led to a sensitivity to certain compounds, particularly BPA. BPA-free pacifiers are a necessity, as they are one of the highest leverage points bringing the centroid mean of the lowest cluster down. To take this a step further, many of the negative reviews of pacifiers mention specific materials, as measured by the title_material boolean. Developing an all-natural and bpa-free pacifier would satisfy current appetites for eco-consumerism and stave off negative ratings.

User experience is particularly important for pacifiers, as poor product design can lead to a crying infant and the adverse effect of its stated purpose. Critical in a positive parental experience is the "hold" or grasp for the baby, as reviews describing the ergonomic grip of a pacifier or difficulty infants have ripping the pacifier from their mouths held an average star rating of 4.6. Manufacturers like Sunshine should take caution in merely widening the plastic base of the pacifier, as blocking the nasal airflow of infants is cited in many of the one- and two- star reviews appearing in the data.

The societal trend towards gender-neutral children's toys has led to a rejection of traditional blue and pink color assignment to infant accessories by both households and retailers.[8] This sociocultural pattern can likewise be observed in our modeling, as both terms appear in the poorest performing cluster at an average star rating a full one-star below the data's mean. Mentioning the softness of a pacifier positively relates to the product opinion, similarly illustrated by the appearance of "hard" describing pacifier texture in the lowest-performing cluster.

Using these characteristics as a blueprint, we have designed for this and the following 2 products mockups for what the data-driven ideal product title would be:

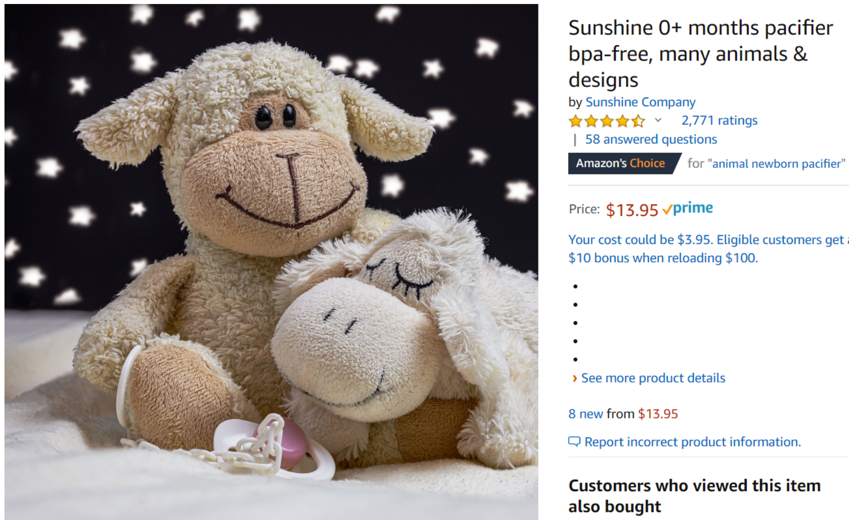

Figure 11: The ideal pacifier product [9]

How to Track Success and Failure

Beyond selecting the ideal product characteristics, we utilized the best and worst clusters of words to curate lists of key indicators of a products success or failure on the market. In addition to the features and product characteristics discussed above, two values in each list deserve additional attention. One of the telltale signs of a lack of interest in a product, particularly with children too puerile to conceal their true emotions, is rejection of that good in favor of a more desirable one. A rise in frequency of "prefers" and "expected" indicate a need to re-assess certain aspects of the product design, as competitive advantage by other manufacturers stands to inhibit future customers.

Similarly powerful in predicting success, however, is the use of "son" or "daughter" in the review body. Connecting the review back to the family unit using personal language indicates satisfaction in this case, as the more impersonal "boy" appears as one of the words representing failure. All of these factors converge to inform the following list of make or break product keywords: Data measures to track success: son, daughter, month, hold , cute, love Data measures to track failing: material (including plastic, rubber, bpa, and the word itself), hard, nose, boy, disappointed, prefers, expected, wrong, bad

5.2 Hair Dryer

Important Insights Found in Modeling

From our linear models, the most significant features within the dataset are the powerful, ionic, and quiet features. This is unique in that people seem to rate both power and quiet as important features, yet inherently by maximizing power you inevitably increase noise level. It seems that finding a balance is necessary, and the perfect balance of power and noise are what consumers are looking for.

From the clustering methods we have discovered clusters of values that we can use in tandem with the significant linear features. There are some unique insights however; within the lowest scoring cluster lies catastrophic product failures such as sparking, fire, and burned. However, even though these characteristics are events and characteristics that should not even appear within a hair dryer, they are still rated above the minimum value of 1 star on average. From the other cluster, we can identify a few words that are consistently mentioned in user reviews that on average have high ratings over a high amount of views: pro, ion, power, and light. Along with these, we know that similar values to the ones mentioned also trend very highly on the rating scale.

The Ideal Product

Given the insights that we've gathered, we can at this point construct what the ideal hair dryer would be to consumers. The ideal product title contains the words ionic, pro/professional, and quiet. Moreover, the ideal length of the title is around 60 characters long. Consumers are looking for a hair dryer that finds the perfect balance between convenience, power, and durability. When it comes to convenience, consumers are looking for products that are lightweight and easy to handle, quiet, and portable. With power, they are looking for hair dryers that provide fast hair drying capabilities, as well as strong airflow to aid in drying times. Finally with durability, consumers are looking for products that last a long time, feel premium, and come included with sufficient lengthy warranty. An ideal inclusion in the ideal product is Tourmaline, which is quickly growing in popularity. Tourmaline possesses natural ionic properties and is quite light in weight. [10]

The ideal product must also absolutely avoid demonstrating any anti-features that will drastically lower the product's average review rating. Products must be safe, avoiding qualities such as potential fire hazards, handling elements that are improperly insulated or that get too hot, and sparking from within the heating element. Furthermore products that develop a smell from the heating element are to be avoided. Finally, products that feel cheap and do not provide a "high quality" feel are to be avoided.

The ideal product may also utilize terms to populate its description. The ideal description should play up the capabilities of the device but may also use words related to the following terms: gift, convenience, and professional. Gifts is a category that finds itself in high ratings in a good amount of reviews, especially when paired with daughter or wife. However, it is most likely wise not to silo the product into a gender due to the current political environment. Furthermore, the highest consistently rated word is "Christmas" with an average of 5 stars over all reviews. Identifying how this product would make a great gift will no doubt increase positive reviews. Convenience and professional are also both terms that can be elaborated on further within the description, as various forms of both words are rated highly.

With this in mind, we are able to develop an ideal product title for Sunshine Company to release on Amazon:

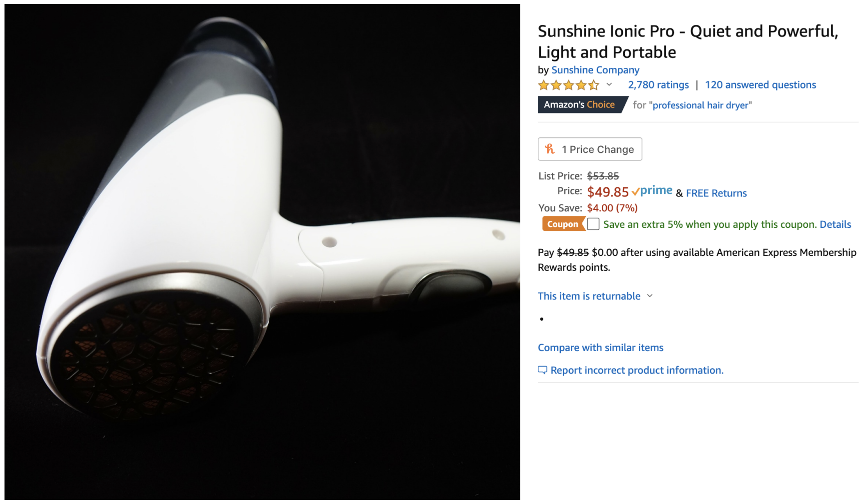

Figure 12: The ideal hair dryer product [11]

How to Track Success and Failure

Within hair dryer reviews, a series of success and failure indicators appear that demonstrate whether or not Sunshine's released product is trending towards success or failure.

Data measures to track success: love, powerful, easy, convenient, fast, quickly. Data measures to track failure: waste, disappointed, return, warranty, died, and fire/burned/sparks/hot (anything heat related but not heat or dries).

5.3 Microwave

Important Insights Found in Modeling

The most significant product features mentioned in the microwave product title were color and location-based features. If the product title mentions stainless (p value: 2.58E-08), the mean star rating is 2.4. However, when stainless is not mentioned the star rating jumps to 3.7 (a 26% difference). Stainless appears in nearly twice as many one- to two-star reviews as four- to five- star reviews. Over the range microwaves appear to be less popular than other microwave locations. Over the range microwaves garnered two times more low reviews than high reviews (p value: 0.000223). When range was mentioned in the product title, the average star rating was 2.6, but when range was not mentioned the average star rating jumps up to 3.7 (a difference of 22%).

The Ideal Product

Customers who purchase microwaves frequently mention attributes such as size, effectiveness, ease of use, and durability in their reviews. Some of the most used words in highly rated reviews are: fits, compact, space, and size. Customers want to be assured that what they are buying uses space economically. Microwaves must also be effective, as "powerful" and "watts" are among the top-rated words. Customers should find using the microwave to be intuitive; reviewers like "easy" and "simple". Proper microwave size, effectiveness, and ease of use can make a microwave "perfect".

What customers do not want in a microwave is "junk". They do not want a microwave that commits "errors", needs "repair", or catches "fire". Customers want a microwave that is durable, safe and will last a long time. "Warranty" is mentioned in low star reviews as well as "broke", "stopped", "months", and "lasted". Moreover, "customer" and "service" appear in low-star reviews. A good backup plan is needed for when things go awry. The warranty should be a better length than competitors' warranties. Customer service should be emphasized, so that when customers interact with the company they leave with a positive impression. "Brand" is included in low star reviews. It may be worth going above and beyond in terms of durability, warranty, and service in order to build a good brand reputation.

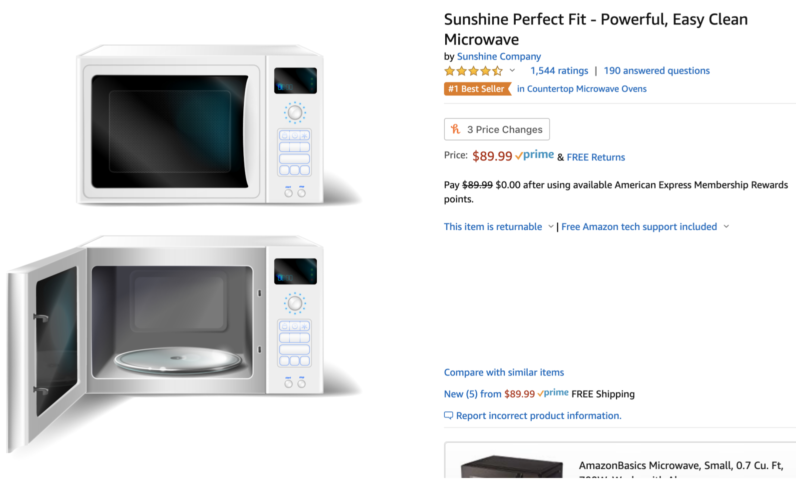

Figure 13: The ideal microwave product [12]

How to Track Success and Failure

Data measures to track success: fits, perfect, powerful, easy, pleased, and love Data measures to track failure: junk, error, repair, fire, broke, poor

5.4 Other Modeling Insights

One of the most critical insights we gleaned from our modeling of the data was a better understanding of the impact helpful votes had on product success. When creating the monthly aggregated version of the data, we engineered two features which for each month determines 1) the maximum helpful votes on a single review in that month and 2) the star rating of that maximum review. When summing the maximum review's star rating over time, the lower totals indicate products with consistently "helpful" one-star reviews whereas the upper bound values often have four- or five- star reviews as the most helpful.

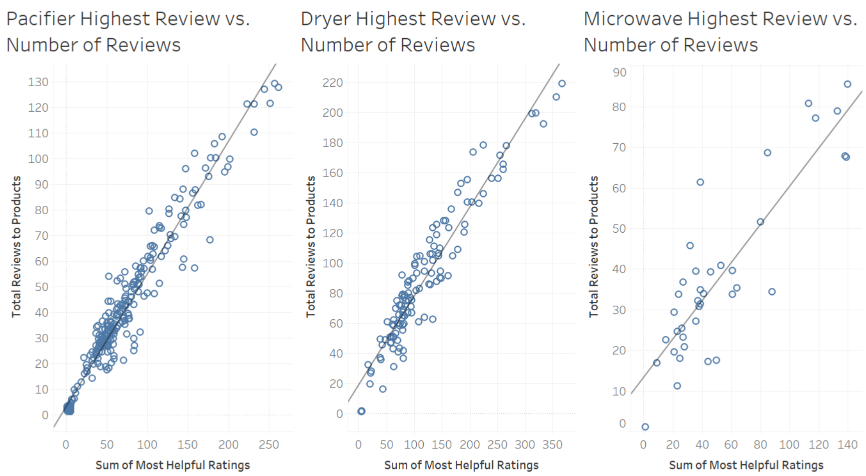

Figure 14: This figure shows the relationship between the the sum of the most helpful voted reviews versus the number of reviews

When plotting this feature against the number of new monthly reviews on a product, a surprisingly linear relationship appears. More five-star reviews being deemed most helpful leads to an increase in the quantity of product reviews. Rather than time-based measures, trending each month's most helpful rating could serve as a forecasting method for brand reputations's increase or decrease.

6 Sensitivity Analysis

When conducting any modeling efforts, the most quintessential step in the process is conducting due diligence in measuring the impact sensitivity has upon model performance. Because our efforts were largely focused on unsupervised machine learning and qualitatively interpreting mathematically-derived patterns, many of the traditionally employed methods of sensitivity analysis including machine learning explainability could not be used. Despite this, we employed several tactics to validate the performance of our modeling and ensure repeatability with future data.

To measure the performance of our sentiment analysis, we looked to the provided quantitative data to help better support our calculated scores. Through plotting visualizations within Tableau, we saw that the review sentiments closely mirrored the user provided star ratings. This validated that our generated sentiment scores were accurate in representing the positive or negative emotion that the user conveyed in their written review. Doing so helped support our word-level sentiment analysis, as we could be sure that the words recorded accurately represented the emotions of the provided user feedback.

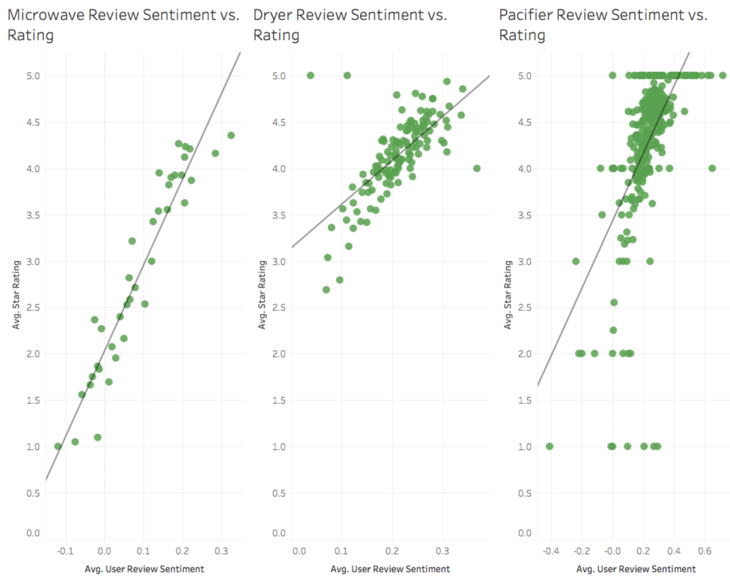

Figure 15: This figure shows the relationship between review sentiment over time

To validate the selection of k for the k-means clustering, we utilized the elbow method to identify the most appropriate value. The elbow method plots the total withinness sum of squares (TWSS) vs. the number of clusters k in the dataset. The goal is to minimize the TWSS, however the graph trends towards zero and reaches zero at the point which k clusters = the number of data points. The elbow in the graph represents the point at which the TWSS is low, but provides a useful amount of identifiable clusters.

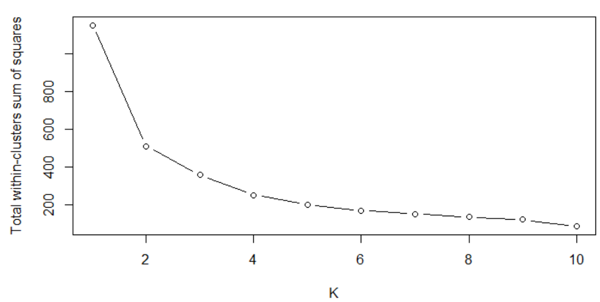

Figure 16: This figure shows minimization of the TWSS in an elbow chart

Selecting the elbow removes subjectivity from k-decisionmaking and ensures our visualized word distributions accurately reflect mathematically-optimal groupings in the data.

7 Model Strengths and Weaknesses

Providing a linear model in addition to the unsupervised modeling allowed for a high-level understanding of how product design features affect its ultimate star rating. With feature rescaling, introduction of different modeling types or other supervised methods, the performance illustrated could be greatly improved. Given the greater emphasis on unsupervised modeling, however, our modeling represented the minimum viable product to justify our focus on clustering and sentiment analysis. The performance was certainly below ideal, as the Rsquared values rarely explained more than 10% of the data and the RMSLE averaged a substantial 0.29. Suggestions for future work pertaining to the supervised modeling of star ratings is included in the following Conclusion section.

Sentiment Analysis is a useful tool when emotion plays a role in data. Though no model can fully interpret human language and human language nuances, they can rate text on a positive-negative scale. Because of the relationship between sentiments and rating established through sensitivity analysis, we could trust that the bag of words generated by sentiment analysis is relevant and utilizes the word-level granularity data in clustering. Yet, we must acknowledge the shortcomings of sentiment analysis. Sentiment analysis algorithms cannot take into account slang, like "this hair dryer is so fire". In this sentence, fire is a very positive adjective. However, sentiment analysis will score it negatively because, outside of slang usage, fire and hair dryer do not go together.

Clustering methods highlight not only which high frequency words appear in positive reviews, but also which high frequency words appear in negative reviews. This enables a compiling of positive features and anti-features. One weakness of clustering is the subjectivity it necessarily entails. Though there are methods for determining the ideal number of clusters, all of these must be interpreted by the modeler and the recommended number may not even make the most sense of the data.

8 Conclusion

After modeling the Amazon review data, we gained a more nuanced understanding of how time, product features, helpfulness of votes, and other factors affect the success or failure of e-commerce sales. In an oversaturated field in which hundreds of competitors without imperfect imitability must vie for the same base of customers, any factors which can provide competitive advantage either save or destroy brands. Heeding the product recommendations provided and taking a data-driven strategy approach to releasing and tracking these three household goods will help ensure Sunshine Company's dominance in the homegoods and baby markets.

Beyond the work conducted to complete this research analysis, more can be done to better understand and forecast customer perspectives. Building off the gradient descent model developed and the idea of using product features to forecast product rating, better linearized models should be created to understand at a more granular level how each feature affects a product's star rating. With its ability to manipulate layers and directions, developing a neural network-based solution could provide the computation power to better understand the relationship between star_rating and the data's quantitative features. The core of our analysis lies within text mining and sentiment analysis, and, as such, efforts to improve the initial quality and accuracy of data before modeling would expedite the cleaning process and improve the insights it could provide. Being able to deal with special case issues like letter doubling, or run records through an improved spell-check system than the ones we utilized could further enrich the level of insights able to be derived from the data.

Of paramount importance, however, is the continuation and monitoring of the insights provided in this report. User behavior data thrives off of longitudinal analysis, and measuring the reviews of customers from here on out is the only way to label and measure performance in the way a supervised machine learning solution would be able to. Additionally, growing the number of tracked boolean features and augmenting the data further will only lead to the data insights growing richer over time. Companies like Sunshine need streamlined, proven, calculation-based strategies to drive product development and growth, and tools like gradient descent, sentiment analysis, and clustering combined with the data analytics techniques outlined above are able to provide just that.

9 Appendix

References

[1] Verena schoenmüller, oded netzer, and florian stahl. the extreme distribution of online reviews: Prevalence drivers and implications, february 2019. http://www.columbia.edu/~on2110/Papers/Schoenmueller_netzer_stahl_2018.pdf,.

[2] Births: final data for 2018, november 2019.https://www.cdc.gov/nchs/data/nvsr/nvsr68/nvsr68_13_tables-508.pdf.[Online;accessed9-March-2020].,.

[3] Brandon. when is peak moving season? april 2019.http://help.movinglabor.com/start/peak-moving-season.[Online;accessed9-March-2020].,.

[4] Bee shapiro. dry winter hair is the worst, february 2018. https://www.nytimes.com/2018/02/20/style/dry-winter-hair-solutions.html.[Online;accessed9-March-2020].,.

[5] Ethen liu. gradient descent with linear regression, october 2015. http://ethen8181.github.io/machine-learning/linear_regression/linear_regession.html.[Online;accessed8-March-2020].,.

[6] K-means cluster analysis. https://afit-r.github.io/kmeans_clustering.[Online;accessed8-March-2020].,.

[7] Bcutie pat round. https://www.ryanandrose.co/products/cutie-pat-pacifier-and-teether.[Online;accessed9-March-2020].,.

[8] Robin levinson-king. children's toys: The backlash against pink and blue branding, december 2018. https://www.bbc.com/news/world-us-canada-46613032.[Online;accessed9-March-2020],.

[9] Myriam zilles. teddy bears sleep star pacifier symbol picture, february 2020.https://pixabay.com/photos/teddy-bears-sleep-star-pacifier-4841062/. [Online; accessed 9-March-2020].,.

[10] Global hair dryer market 2018-2022 | growing popularity of tourmaline hair dryers to boost growth | technavio, december 2019.https://apnews.com/d97ad79705f849548bb25cfc5530a203.[Online;accessed9-March-2020].,.

[11] Hans braxmeier. hairdryer hair dryer device budget home appliance, april 2014.https://pixabay.com/photos/hairdryer-hair-dryer-device-budget-295618/.[Online;accessed9-March-2020].,.

[12] vectorpocket. white microwave oven with empty glass plate inside, with open and close door free vector, march 2018.https://www.freepik.com/free-vector/white-microwave-oven-with-empty-glass-plate-inside-with-open-close-door_2238357.htm#page=1&query=microwave&position=0.[Online;accessed9-March-2020].,.

Data Dictionary

The follow data dictionary was created by consolidating the one provided by the competition with the newly engineered features used in the modeling
\\
customer_id (string): Random identifier that can be used to aggregate reviews written by
a single author.

review_id (string): The unique ID of the review.

product_id (string): The unique Product ID the review pertains to.

product_parent (string): Random identifier that can be used to aggregate reviews for the same product.

product_title (string): Title of the product.

product_category (string): The major consumer category for the product.

star_rating (int): The 1-5 star rating of the review.

helpful_votes (int): Number of helpful votes.

total_votes (int): Number of total votes the review received.

vine (string): Customers are invited to become Amazon Vine Voices based on the trust that they have earned in the Amazon community for writing accurate and insightful reviews. Amazon provides Amazon Vine members with free copies of products that have been submitted to the program by vendors. Amazon doesn't influence the opinions of Amazon Vine members, nor do they modify or edit reviews.

verified_purchase (string): A Y indicates Amazon verified that the person writing the review purchased the product at Amazon and didn't receive the product at a deep discount.

review_date (date): The date the review was written. Converted using lubridate package in R.

title_length (int): The number of characters in product_title before data cleaning.

is_branded (boolean): Whether or not a brand name, either household or Amazon-specific, appears in the title.

title_{color, etc.} (boolean): Dataset-specific boolean values determining if key words which would not be identified by the sentiment analysis algorithms and

```
    could be used in linearized modeling.

Pacifier:
- baby_age (boolean): Was the acceptable age range for the product (0-6 months)
    mentioned in the title?
- newborn_safe (boolean): Was the product mentioned as being safe for 0-month
    babies or infants?
- material (boolean): Was a material (latex, rubber, plastic, etc.) mentioned in
    the title?
- bpa (boolean): Does the product mention being bpa-free in the title?
- novelty (boolean): Does the product title mention a sports team, fantasy
    creature, or other novelty genre?
- animals (boolean): Does the product title mention an animal pattern or animal
    attached to the pacifier?
- nipple (boolean): Does the product title refer to nipple?
- pacifier (boolean): Does the product title refer to pacifier?
- nip_and_pacifier (boolean): Does the product title refer to both nipple and
     pacifier ?
- color (boolean): Does the product title refer to the items color?
- blue_pink (boolean): Does the product title specifically refer to the items
    color as blue or   pink  ?
- natural (boolean): Does the product title refer to its constitution as
    all-natural?
- soothe (boolean): Does the product title refer to its soothing nature?
- orthodontic (boolean): Does the product title refer to its orthodontic nature?
- soft (boolean): Does the product title refer to its soft nature?
- baby (boolean): Does the product title refer to baby?

Microwave:
- cubic_feet (int): How many cubic feet were mentioned in the title?
- has_dimensions (boolean): Were the dimensions of the microwave mentioned in the
    title?
- watts (int): How many watts were mentioned in the title?
- has_watts (boolean): Was the wattage of the microwave mentioned in the title?
- counter (boolean): Was  counter  mentioned in the title?
- range (boolean): Was  range  mentioned in the title?
- drawer (boolean): Was  drawer  mentioned in the title?
- location (boolean): Was  counter ,   range   , or   drawer   mentioned in the
    title?
- stainless (boolean): Was  stainless  mentioned in the title?
- color_or_stainless: Does the product title refer to the microwaves color
    (including stainless)?
- oven (boolean): Was  oven  mentioned in the title?
- convection (boolean): Was  convection  mentioned in the title?

Hairdryer:
- portable (boolean): Does the product title refer to its portable nature?
```

```
- professional (boolean): Does the product title refer to its professional nature?
- ceramic (boolean): Was  ceramic  mentioned in the title?
- tourmaline (boolean): Was  tourmaline  mentioned in the title?
- titanium (boolean): Was  titanium  mentioned in the title?
- ionic (boolean): Was  ionic  mentioned in the title?
- has_color (boolean): Does the product title refer to the hair dryers color?
- powerful (boolean): Does the product title refer to its powerful nature?
- quiet (boolean): Does the product title refer to its quiet nature?
- ergonomic (boolean): Does the product title refer to its ergonomic nature?


review_length (int): The number of characters in the now-consolidated feature
   review_body before data cleaning.

user_feedback (large string): An aggregated large text input which contains both
   the review_headline and review_body for more concise modeling of the users
   experience.

user_review_sentiment (num): A review_id-level score of the reviews sentiments
   pertaining to the product.

product_review_sentiment (num): A product-level aggregation of reviews sentiment
   scores.

product_title_average (num): A review_id-level score of the product titles
   sentiments.

parent_star_rating (num): A product-level aggregation of star ratings across the
   datas reviewers.
```

R Script

```r
#This is a heavily modified script that does not include: cleaning script, gradient
#descent, time & sankey data generation, a lot of the different iterations of
#ggplot visualization scripts written

#List of used packages
#readr,readxl,dplyr,data.table,stringr,tidytext,topicmodels,lexicon,textclean,sentimentr,
#lexicon,tidyr,psych,cluster,factoextra,lubridate,MASS,Metrics
#1. Cleaning

#Remove commas and parentheses
pacifier$product_title<-gsub(",|\\(|\\)|\\.)","",x=pacifier$product_title,ignore.case
   = TRUE)

#Method for extracting boolean values
```

```
pacifier$title_baby_age<-grepl("month|[[:digit:]]
   m|[[:digit:]]m",x=pacifier$product_title,ignore.case = TRUE)

#Convert all new boolean features to zeroes and ones
pacifier[,c(14:31)]<-lapply(pacifier[,c(14:31)], as.integer)

#Calculate Review length
pacifier$review_length<-nchar(pacifier$review_body)

#Omit emojis from review body and head
pacifier$review_body<-iconv(pacifier$review_body, from = "latin1", to =
   "ascii",sub = "byte")
pacifier$review_body<-replace_emoticon(pacifier$review_body, emoticon_dt =
   lexicon::hash_emojis)
pacifier$review_headline<-iconv(pacifier$review_headline, from = "latin1", to =
   "ascii",sub = "byte")
pacifier$review_headline<-replace_emoticon(pacifier$review_headline, emoticon_dt =
   lexicon::hash_emojis)
#Remove exteranous emojis that could not be translated by the lexicon
pacifier$review_body<-gsub("<.*?\\>","",x=pacifier$review_body)
pacifier$review_body<-gsub(">","",x=pacifier$review_body)
pacifier$review_headline<-gsub("<.*?\\>","",x=pacifier$review_headline)
pacifier$review_headline<-gsub(">","",x=pacifier$review_headline)

#Merge review body and headline columns
#If the first 30 characters of both headline and body are the same, remove the
   headline data
pacifier$review_headline[which(substr(pacifier$review_body,1,30)==substr(pacifier$review_headlin
#Add a period to the end of features w/ no end punctuation
pacifier$review_headline<-trimws(pacifier$review_headline)
pacifier$user_feedback<-NA
pacifier$user_feedback[-grep("[[:punct:]]$",pacifier$review_headline)]<-paste(pacifier$review_he
   = ". ")
pacifier$user_feedback[grep("[[:punct:]]$",pacifier$review_headline)]<-paste(pacifier$review_hea
   = " ")
pacifier<-pacifier[,-c(10:11)]
pacifier$user_feedback<-gsub("NA\\.","",pacifier$user_feedback)
pacifier$user_feedback<-gsub("([[:punct:]])\\1{1,}", "\\1", pacifier$user_feedback)


#2 Feature Importance and Linear Modeling
feature_table <- data.table(feature = "", doesnt_contain = "", contains = "",
                           fourtofive_star = "", onetotwo_star = "", percent_high =
                              "")
for (i in 14:29) {
  feature_table <- feature_table %>%
    add_row(
```

```
      feature = colnames(pacifier)[i],
      doesnt_contain = as.numeric(pacifier %>%
                              filter(pacifier[,i] == 0) %>%
                              summarize(mean(star_rating))),
      contains = as.numeric(pacifier %>%
                        filter(pacifier[,i] == 1) %>%
                        summarize(mean(star_rating))),
      fourtofive_star = as.numeric(pacifier %>%
                              filter(pacifier[,i] == 1) %>%
                              summarize(sum(star_rating==5 | star_rating==4))),
      onetotwo_star = as.numeric(pacifier %>%
                            filter(pacifier[,i] == 1) %>%
                            summarize(sum(star_rating==1 | star_rating==2))),
      percent_high = 100* (fourtofive_star / as.numeric(pacifier %>%
                                                   filter(pacifier[,i] == 1) %>%
                                                   summarize(n())))
    )
}
feature_table <- feature_table[-1,]
feature_table[,2:6]<-lapply(feature_table[,2:6],as.numeric)

#Linear Modeling
set.seed(75)
train_ind <- sample(seq_len(nrow(pacifier)), size = floor(0.75 * nrow(pacifier)))
reg <- pacifier[,c(5:9,11:12,14:29,34)]
train <- reg[train_ind, ]
test <- reg[-train_ind, ]

ols <- lm(star_rating ~ ., data = train)
lm.fitted <- abs(predict(ols,newdata=test))
error<-rmsle(test$star_rating,lm.fitted)
error
summary(lm)
plot(lm)

stepback <- stepAIC(ols, direction="backward")
summary(stepback)
plain<-lm(star_rating~1, data=train)
forward <- stepAIC(plain, direction="forward",scope =list(upper=ols, lower=plain))
summary(forward)
lm.fitted <- abs(predict(forward,newdata=test))
error<-rmsle(test$star_rating,lm.fitted)
error

pval<-as.data.frame(summary(forward)$coefficients[,4])
pval$feature<-rownames(pval)
colnames(pval)<-c("p_value","feature")
```

```
feature_table<-feature_table%>%left_join(pval)%>%filter(!is.na(p_value))

#3. Sentiment Analysis

#Collect review-level average sentiment scores
sentences <- pacifier %>%select(review_id,user_feedback) %>% mutate(sentences =
    get_sentences(user_feedback))
sentiment <-
    sentiment_by(sentences$sentences,by=sentences$review_id)%>%select(review_id,
    ave_sentiment)
colnames(sentiment)[2]<-"user_review_sentiment"
pacifier <- pacifier%>%left_join(sentiment, by="review_id")

#Word level distribution bag of words
words <- pacifier %>%
  unnest_tokens(word, user_feedback) %>%
  select(review_id, word) %>%
  count(word, sort = T) %>%
  anti_join(stop_words) %>%
  filter(n>=100) #n=40 for microwave
words <- words[-grep("\\d",words$word),]
summary(words$n)

#Claculate average star ratings per word
star <- data.frame(word = words$word)
for (i in 1:length(words$word)) {
  temp <- pacifier %>% filter(grepl(words$word[i], user_feedback))
  star[i,"rating"] <- mean(temp$star_rating)
}
rm('temp')
words <- words %>% left_join(star, by="word")

#How many reviews is the word in, how many 5 star reviews, and how many 1 star
    reviews?
appearances <- data.table(word = "", five_star = "", one_star = "", total_num = "")
for (i in 1:length(words$word)) {
  appearances <- add_row(appearances,
                        word = words$word[i],
                        five_star = as.numeric(pacifier %>%
                                             filter(grepl(words$word[i],
                                                 user_feedback, ignore.case = T))
                                                 %>%
                                             filter(star_rating > 4.00) %>%
                                             summarize(n())),
                        one_star = as.numeric(pacifier %>%
                                            filter(grepl(words$word[i],
                                                user_feedback, ignore.case = T))
```

```
                                                    %>%
                                                filter(star_rating < 2.00) %>%
                                                summarize(n())),
                        total_num = as.numeric(pacifier %>%
                                                 filter(grepl(words$word[i],
                                                     user_feedback, ignore.case = T))
                                                     %>%
                                                 summarize(n()))
  )
}
appearances <- appearances[-1,]
appearances$five_star <- as.numeric(appearances$five_star)
appearances$one_star <- as.numeric(appearances$one_star)
appearances$total_num <- as.numeric(appearances$total_num)
words <- words %>% left_join(appearances, by="word")

#4. Clustering for user_feedback

#KMeans
clustering.words<-data.matrix(words)
clustering.words<-clustering.words[,-1]
row.names(clustering.words)<-words$word
clustering.words<-scale(clustering.words)

#Use elbow method to determine optimal k value
k_tot.wthnss <- sapply(1:10,function(x){kmeans(clustering.words, x,
    nstart=50)$tot.withinss})
k_tot.wthnss
plot(1:10, k_tot.wthnss,type="b", pch = 1,xlab="K",ylab="Total within-clusters sum
    of squares")

#Create Kmeans and plot
k4<-kmeans(clustering.words,centers=4,nstart = 50)
#Text
words %>% as_tibble() %>% mutate(cluster = k4$cluster, words =
    row.names(clustering.words)) %>%
  ggplot(aes(rating, total_num, color = factor(cluster), label = words))+
      geom_text()+ labs(title = "Pacifier Word KMeans K=4")
#Points and Ellipses
words %>% as_tibble() %>% mutate(cluster = k4$cluster, words =
    row.names(clustering.words)) %>%
  ggplot(aes(rating, total_num, color = factor(cluster), label = words)) +
      geom_point()+
  stat_ellipse(aes(x=rating, y=total_num,color=factor(cluster)),type = "norm")+
      labs(title = "Pacifier KMeans Clusters K=4")

#Cbind k-means values to the dataset
```

```
words<-cbind(words,k4$cluster)
#Make successful and failing lists
successful<-words%>%filter(`k4$cluster`%in% c(1,2))
failing<-words%>%filter(`k4$cluster`==4)
successful<-successful[,-7]
failing<-failing[,-7]

#Hierarchical
hc.complete<-hclust(dist(clustering.words), method="complete")
hc.average<-hclust(dist(clustering.words), method="average")
hc.single<-hclust(dist(clustering.words), method="single")
hc.centroid<-hclust(dist(clustering.words), method="centroid")
k4_centroid<-cutree(hc.centroid, k=4)
k4_complete<-cutree(hc.complete, k=4)
k4_average<-cutree(hc.average,k=4)
k4_single<-cutree(hc.single,k=4)
#Plot complete hierarchical cluster
plot(hc.complete,main="Complete Linkage",xlab="",ylab="")
```