

An Unsupervised Learning Approach to Separating Products into Groups of Relative Expenditure Using Online Customer Reviews

Rory Tisdall

October 2021

School of Mathematics,

Cardiff University

A dissertation submitted in partial fulfilment of the

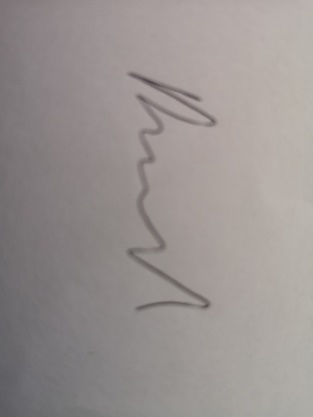
requirements for MSc (in Data Science and Analytics)

by taught programme.

|  |  |
| --- | --- |
| **CANDIDATE’S ID NUMBER** | **21010417** |
| **CANDIDATE’S SURNAME** | **Please circle as appropriate**  **Mr / Miss / Ms/ Mrs / Rev / Dr / Other Mr Tisdall** |
| **CANDIDATE’S FULL FORENAMES** | **Rory** |

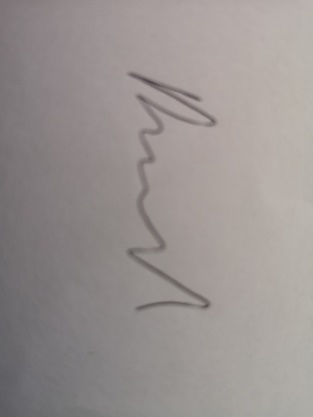
**DECLARATION**

This work has not previously been accepted in substance for any degree and is not concurrently submitted in candidature for any degree.

Signed  (candidate) Date 22/10/2021

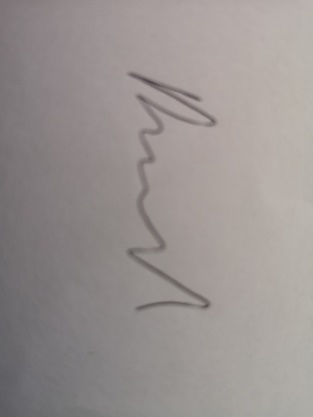
**STATEMENT 1**

This dissertation is being submitted in partial fulfilment of the requirements for the degree of MSc (insert MA, MSc,MBA, etc, as appropriate)

Signed  (candidate) Date 22/10/2021

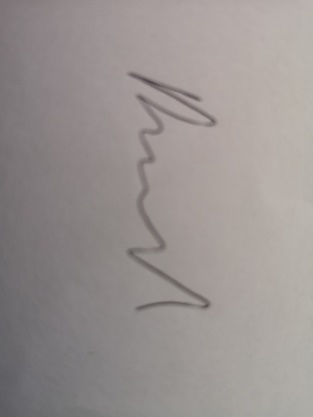
**STATEMENT 2**

This dissertation is the result of my own independent work/investigation, except where otherwise stated. Other sources are acknowledged by footnotes giving explicit references. A Bibliography is appended.

Signed  (candidate) Date 22/10/2021

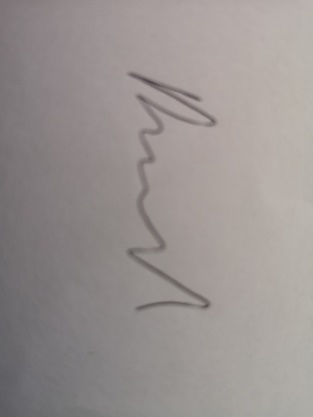
**STATEMENT 3 –**

I hereby give consent for my dissertation, if accepted, to be available for photocopying and for public viewing, and for the title and summary to be made available to outside organisations.

Signed  (candidate) Date 22/10/2021

**STATEMENT 4 - BAR ON ACCESS APPROVED**

I hereby give consent for my dissertation, if accepted, to be available for photocopying and for public viewing **after expiry of a bar on access approved by the Graduate Development Committee.**

Signed  (candidate) Date 22/10/2021

## **Executive Summary**

The consumer price index (CPI) tracks the variation in prices for household consumer goods, whereby the Office for National Statistics (ONS) have traditionally manually collected data for calculating the index. In recent years they have been moving towards the uses of alternate data sources, including shop scanner data and data web scraped from online stores. The large number of products available on online retail stores allows for the prices of more products to be collected, and the use of web scraping permits more frequent and efficient data collection than traditional manual collection. The issue with web scraping is that information on product expenditure cannot be collected, hence, the product weights cannot be calculated for CPI, this is a concern as product weights are needed to compute an accurate CPI. This paper explores a method that works towards proxying expenditure without the use of relevant historic data. Making use of online customer review data and unsupervised learning, this study attempts to separate products from four categories of consumer electronic goods into two groups of relative total expenditure using a density-based clustering approach (DBSCAN), whereby the group with the most sparsely spread data should contain the bestselling products. Scanner data with true expenditure values has been used to evaluate the models produced with a range of metrics. Overall, the method resulted in varying success for each of the four categories of electronic goods. The best F1 scores achieved without the use of true or past expenditure data, for the four datasets are 0.64, 0.61, 0.55 and 0.34.

## **Contents**

[List of Figures 5](#_Toc85792262)

[List of Tables 5](#_Toc85792263)

[1. Introduction 6](#_Toc85792264)

[2. Literature review 9](#_Toc85792265)

[***2.1 Web-scraping*** 9](#_Toc85792266)

[***2.2 Expenditure proxies*** 9](#_Toc85792267)

[***2.3 The effect of customer reviews on product sales*** 11](#_Toc85792268)

[***2.4 DBSCAN for outlier detection*** 13](#_Toc85792269)

[3. The data 14](#_Toc85792270)

[4. Methods 18](#_Toc85792271)

[***4.1 Data pre-processing*** 18](#_Toc85792272)

[***4.2 Feature engineering*** 18](#_Toc85792273)

[***4.3 Feature selection*** 19](#_Toc85792274)

[***4.3 Clustering*** 25](#_Toc85792275)

[5. Results 28](#_Toc85792276)

[***5.1 Evaluating with scanner data*** 28](#_Toc85792277)

[***5.2 Evaluating without scanner data*** 32](#_Toc85792278)

[*5.3 Combining results* 37](#_Toc85792279)

[6. Conclusion and future work 38](#_Toc85792280)

[***6.1 Conclusion*** 38](#_Toc85792281)

[***6.2 Future work*** 39](#_Toc85792282)

[***6.3 Reflection*** 40](#_Toc85792283)

[References 42](#_Toc85792284)

## **List of Figures**

[Figure 1: Histograms showing the distributions of normalised total sales from the scanner data. 15](#_Toc85791237)

[Figure 2: Histograms showing the distributions of the total volume of reviews from the web scraped data. 16](#_Toc85791238)

[Figure 3: Feature importances obtained from OLS. Note: coefficient values have been removed for readability, coefficient magnitudes should be taken as the length of the bars. 21](#_Toc85791239)

[Figure 4: Feature importances obtained from logistic regression. Note: coefficient values have been removed for readability, coefficient magnitudes should be taken as the length of the bars. 24](#_Toc85791240)

[Figure 5: Scatter plots showing the DBSCAN results for features (Rvol and Rvol/%Rec) parameterised using scanner data. 29](#_Toc85791241)

[Figure 6: Scatterplots showing the DBSCAN results for features (PC1 and PC2) parameterised using scanner data 31](#_Toc85791242)

[Figure 7: Metrics for DBSCAN results for features (Rvol and Rvol/%Rec) parameterised without scanner data. 33](#_Toc85791243)

[Figure 8: Metrics for DBSCAN results for features (PC1 and PC2) parameterised without scanner data. 35](#_Toc85791244)

## **List of Tables**

[Table 1: Pearson coefficients derived from correlating the total sales and review volume 17](#_Toc85791165)

[Table 2: P-values of the coefficients of each feature derived the multiple linear regression models. 22](#_Toc85791166)

[Table 3: The parameters which achieved the best mean F1 scores for each feature set. 28](#_Toc85791167)

[Table 4: Metrics for DBSCAN results for features (Rvol and Rvol/%Rec) parameterised using scanner data. 30](#_Toc85791168)

[Table 5: Metrics for DBSCAN results for features (PC1 and PC2) parameterised using scanner data. 32](#_Toc85791169)

[Table 6: Metrics for DBSCAN results for features (Rvol and Rvol/%Rec) parameterised without scanner data. 34](#_Toc85791170)

[Table 7: Metrics for DBSCAN results for features (PC1 and PC2) parameterised without scanner data. 36](#_Toc85791171)

[Table 8: The F1 scores achieved from the four models analysed. 37](#_Toc85791172)

## **1. Introduction**

The consumer price index (CPI) measures the rate of inflation of the prices of goods and services bought by households. In in the United Kingdom (UK), this index is produced monthly by the Office for National Statistics (ONS), the national statistical institute of the UK. The index is calculated by analysing price changes of items in the ‘basket of goods’, an imaginary basket which is made up of over 700 goods and services with their price changes sampled from approximately 20,000 UK outlets (ONS 2017). These goods and services are selected as they are most commonly bought by households across the country. In order to calculate the index, prices for each item in the basket of goods must be collected each month. Traditionally methods of price collection were and, in many cases, still are manually carried out by people via telephone surveys or collecting prices from shops in person. As one could imagine, this can be quite time consuming and expensive. Nowadays, most retail and service companies will collect data on their sales as the required technology to do so is easily accessible and there is increased awareness of the value of data. An example of this is retail scanner data, whereby data is collected whenever an item’s barcode is scanned at a till. Scanner data is highly effective for calculating CPI as it contains prices of all the items that have been purchased and gives the quantities of sales. This is useful for calculating CPI as a weight is given to each good and service in the ‘basket of goods’, whereby the weight shows their importance to the total expenditure of all the items in the basket (Gaf 2020). Unfortunately, scanner data is not readily available and must be provided by the owners, which could be seen as risky because the data contains valuable information which many retailers would choose to keep private. There is one freely available source to obtain the prices of products, the internet. Most retailers will have online-based stores, some being solely online stores, with online stores usually showing more products than their physical store counterparts. This provides an enormous number of products to choose from and their prices to collect via web scraping. Web scraping is the process of using a computer programmed web crawler to move through web pages and extract data from the web page’s HTML, generally text visible on web pages can be collected by a web crawler and stored as data in text files.

The use of web scraping to collect prices opens the potential to collect prices for a wider range of goods than more traditional methods, also it is far more appropriate for frequently collecting prices. It would be feasible to run a web crawler each day, but having people hand collect prices each day would not be. Using web scraped and scanner data allows more efficient identification of new potential products to enter the basket of goods in comparison to manual collection, as manual price collectors only collect the prices of specified items. There is one substantial drawback to using web scraped data for CPI, there is rarely, or if ever, any data displayed on a web page giving the quantities sold or relating to expenditure. Without this information no weights can be calculated for each item for CPI, and weights are key for producing an accurate price index. For example, a study completed by the ONS (Sands 2020) showed a CPI over a year for toothpaste with and without weights; the biggest difference between the two was over 20%. This highlights the importance of having expenditure information when producing CPI, and therefore, introduces the requirements of expenditure proxies (methods of approximating expenditure) when using web scraped data.

This study investigates how different qualities of a product displayed on a web page could potentially be useful in creating expenditure proxies, with a particular focus on product reviews. The idea arises from reviews generally being placed after a product has been purchased and reviews influencing potential buyers to purchase a product. Although the number of reviews or average rating could not give a direct insight on the number of sales of a product, as not everyone who buys a product will review it or give 5 stars, it could be used to separate products into groups of ones which are more likely to be top selling products and ones which are not. An unweighted price index calculated with only top selling products should be more representative to the true index weighted with actual expenditure than an unweighted price index calculated with every single product available, for in most categories of goods there are products which are considerably more frequently purchased than others.

This study aims to find out if web scraped products can be clustered into groups of popularity, the products tested here are tech goods from a leading retailer store in the UK. Web scraped data was used alongside scanner data containing the true sales quantities for each of the products, allowing the evaluation of all models created. The products are split into categories: laptops, phones, desktop computers and tablets; the price and review information were scraped once for each product in all the categories. The scanner data contains expenditure information for all products over the period of a year. Ideally, products could be split into groups according to estimated expenditure, though, in the scenario of multiple groups another problem would arise: which group represents what approximate proportion of total expenditure? To evade this, products will be split into only two groups; with the aim that the group corresponding to more expenditure should be obvious. To complete this, an unsupervised learning problem will be formulated with variables used from the products’ web page to cluster the products and then resultant clusters evaluated with the scanner data. Since the aim is to create only two clusters, the choice of clustering algorithm should be picked accordingly. Partitioning and agglomerative clustering algorithms allow the number of clusters produced to be specified as a parameter, however, these algorithms are sensitive to outliers and therefore, using them risks clustering all but one of the data points to a single cluster, and the other cluster to a single, extremely outlying point. It is important to note that outliers should be included in the modelling because they are more likely to be top-selling products. For this reason, the choice of algorithm is density-based spatial clustering with noise (DBSCAN) (Ester et al. 1996). This algorithm is known for being able to handle outliers, and although the number of clusters produced cannot be directly parameterised, the other parameters can be configured until only two clusters are produced.

Chapter two, the literature review, will explore web scraping, previous research into expenditure proxies, the effects of online reviews on sales and the use of DBSCAN for outlier detection. Chapter three, the data, will describe the datasets used in this study. Chapter four, the methods, will outline data pre-processing, feature engineering, feature selection and the techniques used to parameterise and select DBSCAN models. Chapter five, the results, will display and discuss the models created. Finally, Chapter six, will provide conclusions, areas for future work and a personal reflection.

## **2. Literature review**

### ***2.1 Web-scraping***

There has been a large shift to online shopping in the 21st century, with many retailers expanding or creating online stores; or only having online stores. This provides a large availability of open-access price data, which can be manually collected using web scraping. The ability to have a computer program that automatically collects text data contained on a web page is an effective method of obtaining data, however, there are several aspects regarding legality and ethics of web scraping that should be considered. The paper Legality and Ethics of Web Scraping (Kortov and Silva 2018) highlights the main areas to consider: terms of use, copyrighted material, purpose of web scraping, damage to the website and individual privacy. The ONS (2020) have published a web scraping policy which details the following: ‘minimise burden on website owners’, ‘respect the robots exclusion protocol’ and ‘abide by all applicable legislation and monitor the evolving legal situation’. To minimise burden on website owners, one must design the web crawler in such a way that it does not cause stress on the website’s server or interrupt public traffic to the website. Respecting the robots exclusion protocol involves notifying the website owners with a request to scrape their site along with the purpose of data collection. The last policy detail involves adhering to the Data Protection Act 2018. The web scraping activities carried out during this study have been done considering both the legal and ethical implications of web scraping, ensured by obeying the policies highlighted in the ONS (2020) web scraping policy.

### ***2.2 Expenditure proxies***

Expenditure is used to calculate product weights in order to compute a weighted CPI. Retail websites rarely display information on product expenditure and so it cannot be scraped, therefore, the exact expenditure cannot be known and can only be approximated. Web scraping to collect prices for CPI has only been adopted in recent years by government institutions and hence there is little research on the topic of expenditure proxies. Most related research requires historic data, or those without, have substantial flaws giving limited success for a general solution. It could be argued that an unweighted CPI can be utilised to negate the need of knowing the expenditure for products, but an unweighted index could have large differences compared to a weighted index (Sands 2020). Also, an unweighted index is unlikely to provide an accurate representation of how price changes will affect the population which is one of the key reasons for the interest in CPI.

This section of the literature review focuses on two expenditure proxy papers that have been found to be most relevant to this research, where there is limited or no use of historic data in modelling estimates for expenditure. The first paper to be reviewed was published by the ONS (Sands 2020). The study demonstrates fitting statistical distributions to the ranks at which products are displayed on an online shelf level. To calculate these distributions, summary statistics covering product sales quantities needed to be provided from the retailer. In the results section, there is a CPI time-series covering one year with CPI calculated from true expenditure, estimated weights and no weights. The estimated weighted CPI shows a similar trend to the true weighted CPI, although the inflation is slightly over-estimated. The trend of the unweighted CPI does not resemble that of the true weighted CPI, and inflation is substantially underestimated. The study shows good potential of this method, however, it relies on being provided with some information from the retailer and more so, page rankings are determined by the number of products sold. Some websites give the option on how to order products, a common option being popularity which is likely to correlate with product sales, however, not all retailers provide this option; even if popularity for one retailer is determined from product sales, this does not mean that it will be the case for other retailers. A general solution to proxy expenditure should be robust to variation in the design of retail websites, for example, the way in which shelf order is decided.

The second paper, Chessa and Griffionen (2019), focuses on comparing the number of prices scraped to sales quantities in scanner data. The entire website of a Dutch department store was scraped each day for 35 months, and the number of appearances of a product in separate areas of the website was also collected. It was found that the number of web scraped product prices and products sold in the scanner data showed high correlations. Monthly weighted price indices for men and women’s clothing were calculated from both the scanner data and web scraped data separately, it is noted that ‘clothing is a notoriously complex field in price index calculator, because product categories may be characterised by high churn rates’. The scanner data indices used sales quantities for each product as weights, and the web scraped data used average monthly prices for each product alongside the total number of prices scraped for that product, summed over the total number of products as weights. The price indices for both data sources are very similar month by month, with an average of 0.3% difference each year. These results demonstrate that, in this instance, this method has proved to be an effective proxy for expenditure. Although, the study used data from a single retailer so it may not be as effective with data from other retailers. The major drawback of this method is the extensive web scraping required to collect the data, scraping a whole website may take multiple hours depending on the number of products offered and the website design. This can be expensive, especially if multiple websites are being scraped, ruling this out as a possible method for some institutes. Another flaw is that this method relies on products being displayed in more than area on a website, some retailer websites might only have each distinct product displayed in a single area on their website, if so all products that are available on the website will have one price scraped for each day they are displayed; rendering this method ineffective.

### ***2.3 The effect of customer reviews on product sales***

Nowadays, it is rare to find online retailers that do not provide the option to leave a review on a purchased product. Many retailers will ask the customer to leave a review and sometimes offer incentives. Whilst customer reviews benefit the retailer in many ways, they also benefit potential new customers as these reviews are the opinion of past customers. Floyd et al. (2014) investigated how review ratings and the volume of reviews affect the unit sales of products. This was conducted through a meta-analysis of data from other research papers, which reviewed the following products: hotels, movies, books, digital cameras and video games consoles. Combining the data from each paper into a linear meta-analytic model, they found that review ratings were twice as important as review volume when modelling variations in sales, though review volume was still a significant variable. The authors noted that in other papers the volume of reviews was found to have a greater influence on the variation in product sales than review rating. Floyd et al.’s (2014) findings support the hypothesis of this study: products with a high review volume and average rating are likely to be top selling products, although the product reviews used in Floyd et al.’s (2014) study were from a wide range of categories, and not just tech goods. Another paper, by Ghose and Ipeirotis (2010), conducted similar research with review data from Amazon on entirely tech goods. This consisted of 144 video players, 109 digital cameras and 158 dvds; the Amazon sales rank was used as a proxy for the number of sales. Although, the main body of the paper is based on analysing the text in the reviews to determine their helpfulness, there is a section on predicting whether the number of sales are affected by the publishing of reviews. This was done by comparing the sales ranks between two time periods and the respective review volumes within these periods and training a random forest predictor which produced accurate results. Ghose and Ipeirotis (2010) found a high correlation between the reviews and sales, though stated ‘it is unclear whether the reviews influence sales, or whether the reviews are just a manifestation of the underlying sales trend`. Overall, Ghose and Ipeirotis’ (2010) paper provides some backing to the hypothesis of this study, as the products used in modelling are also tech goods and high correlation is found between the increase in reviews and the increase in sales; although, it was reported that causality cannot be proven between the two. A third paper, by Cui et al. (2012), made use of Amazon as the source of data for product reviews and sales similar to Ghose and Ipeirotis’ (2010) study. Cui et al. (2012) used data on consumer electronics and video games. Using regression analysis, they found that the volume of reviews had more impact on video games than it had on consumer electronics, however, review ratings were more influential on consumer electronics. It was found that the effect of the volume of reviews on product sales would decrease as a product became older, although, this could be due to newer products being more desirable. The findings of Floyd et al. (2014), Ghose and Ipeirotis (2010), and Cui et al. (2012) support the hypothesis of this study, nonetheless, the first paper had mixed data sources and the other two used the same data source with Amazon’s product rank as a proxy for expenditure instead of true sales quantities.

### ***2.4 DBSCAN for outlier detection***

Density based clustering methods are useful for finding outliers in data because they can handle arbitrary shaped clusters and are not sensitive to noise in the data (Aggarwal 2017). The algorithm can detect noise in data sets of varying shape which makes this algorithm a good-fit for a general solution for splitting a group of products into clusters to aid proxying expenditure. DBSCAN is an efficient algorithm and can be used on large datasets, although it does require two input parameters: maximum neighbourhood radius and the minimum number of samples in a cluster (Khan et al. 2014). The minimum number of samples in a cluster can be set high to ensure that only two clusters are being created: not noise and noise (outliers). Altering the maximum neighbourhood radius should be used to control the number of outliers as a lower value will decrease cluster size but increase the similarity of points within the cluster, therefore, increase the number of outliers found. Also, a study found that increasing the maximum neighbourhood radius decreased the number of clusters and highlighted the most anomalies in the data (Nowak-Brzezińska and Xięski 2017).

## **3. The data**

The data used in this study is from a popular general merchandise retailer based in the UK. The data can be broken down into four data sets, each a separate category of tech goods: laptops, desktop computers, mobile phones, and tablets. The data was collected by scraping each section of the website for the four categories; the following attributes for each product were obtained: product name, product identifier, price, shelf order, number of reviews, percentage of reviews which recommend the product, review text, review date and the average review rating. The size of the data sets can be defined by the number of products in the category of tech goods. Each dataset has no more than 300 products, the exact sizes cannot be stated as there is a confidentiality agreement with the data provider. From largest to smallest, the size of the data sets are ranked: phones, laptops, tablets and desktops. Scanner data for the retailer stretching over the period of one year has been provided by the retailer containing sales quantities. Products have been matched between the web scraped and scanner data sets so that expenditure models created can be evaluated with the true sale quantities. Any products that were not matched have not been included in this study.

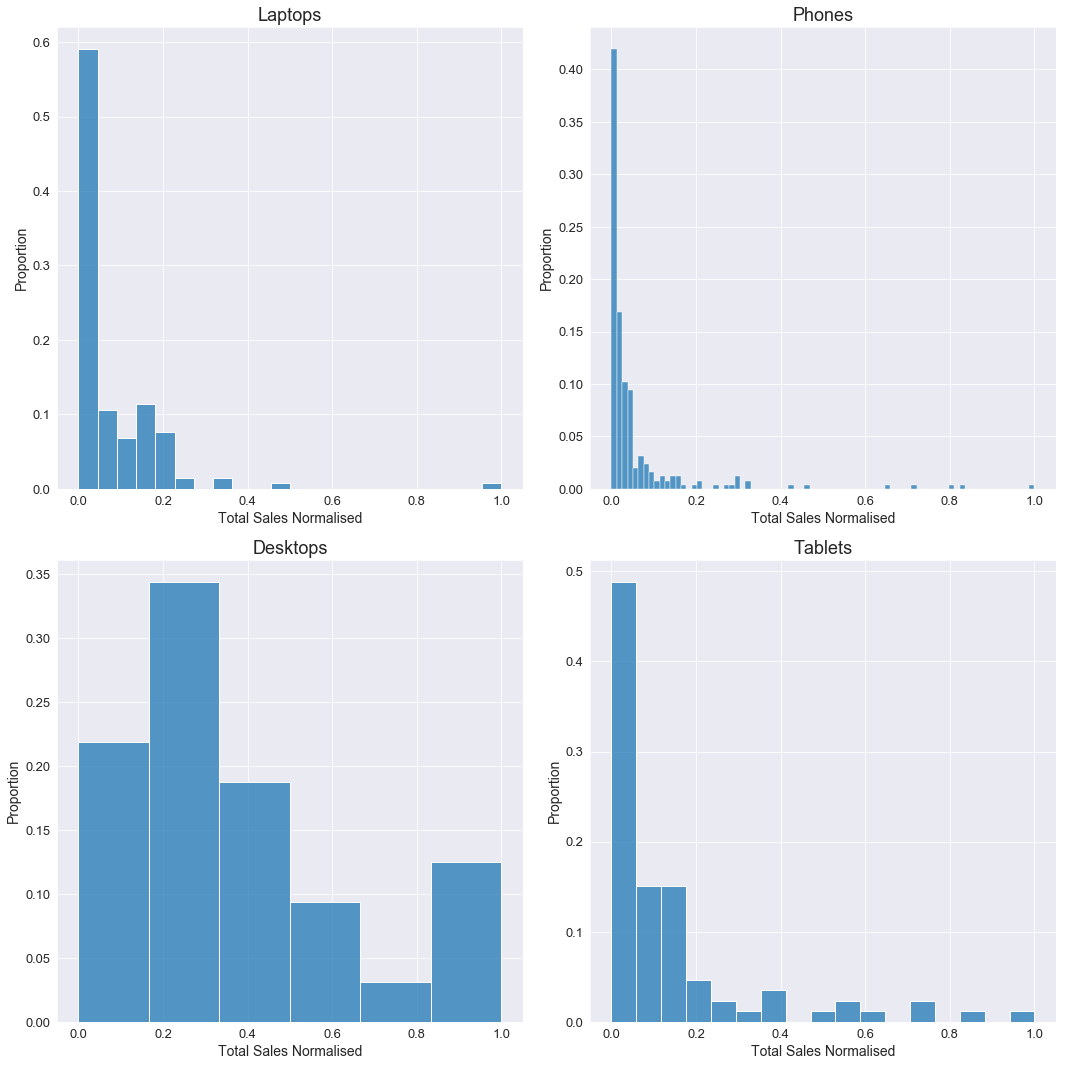


Figure 1: Histograms showing the distributions of normalised total sales from the scanner data.

The histograms from figure 1 show the quantity of sales distributions of products in each of the four categories from the scanner data. All of the categories, with the exception of desktops, are most similar to an exponential distribution, therefore, the majority of the sales come from only a handful of products. Desktops behave differently, the majority of products lie in the second lowest sales bin, and the bin with the highest number of sales represents 12% of all products; bearing in mind that desktops contain the lowest number of products by a significant amount compared to the other three categories.

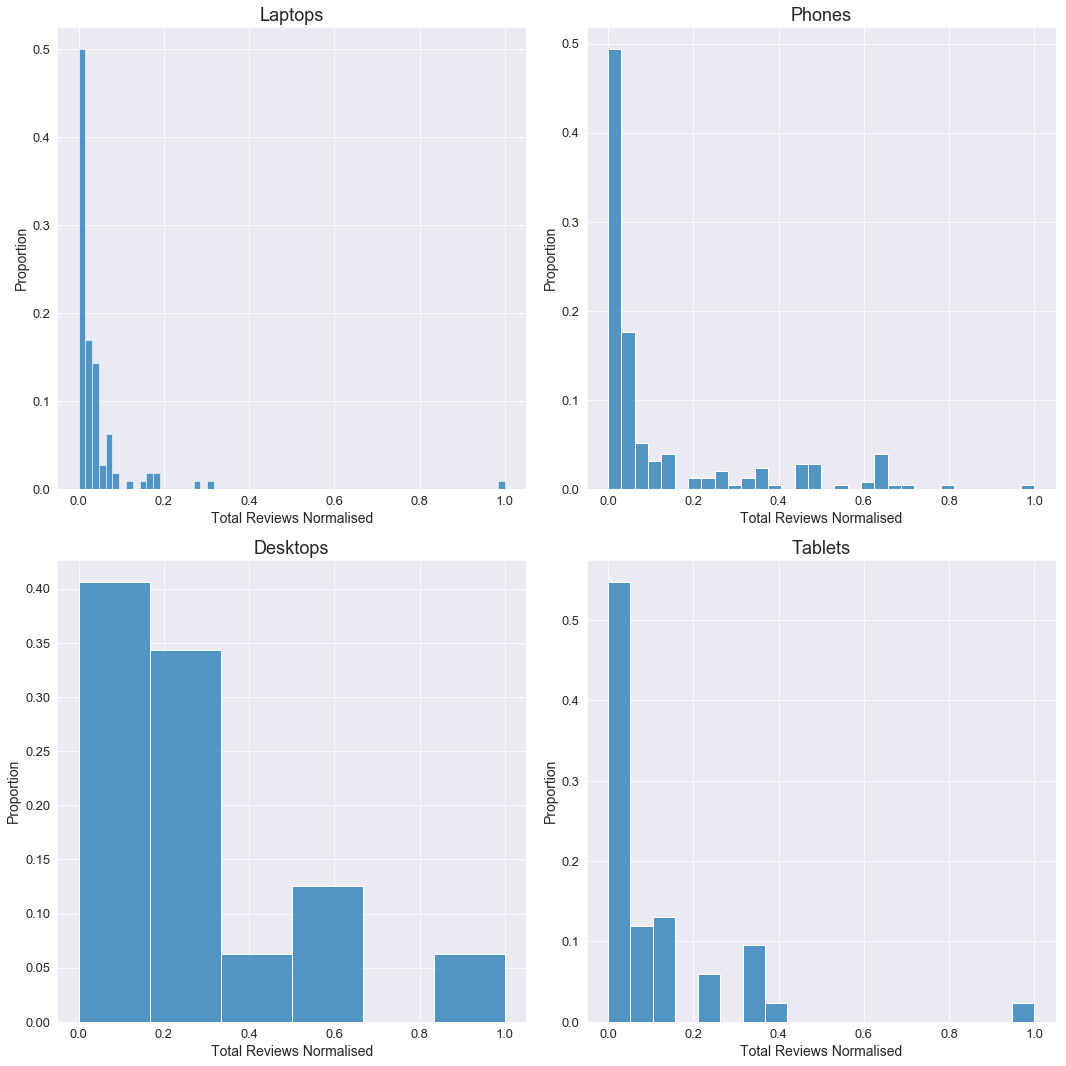


Figure 2: Histograms showing the distributions of the total volume of reviews from the web scraped data.

Figure 2 shows distributions of the number of reviews for each product, for each of the four categories from the web scraped data. The distributions are similar to those of total sales from the scanner data and are most like an exponential distribution; except for desktops, although now the majority of products lie in the lowest bin. For all categories, roughly half of the products do not have any reviews. The number of reviews are highly unbalanced in laptops and tablets, shown by the large gaps in the histograms with no proportion of the data, as the majority of reviews are on a small proportion of products in the data.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Category** | Laptops | Phones | Desktops | Tablets |
| **Pearson Coefficient** | 0.91 | 0.45 | 0.60 | 0.50 |

Table 1: Pearson coefficients derived from correlating the total sales and review volume

Table 1 describes the Pearson coefficients obtained by correlating the number of reviews and number of sales for each category. A coefficient of 1 means full positive correlation and -1 means full negative correlation. All categories show moderate positive correlation, except for laptops; showing highly positive. The relationship between the number of sales and reviews provides potential for exploitation, allowing products to be split into two groups, with one group containing most of the products and the other group containing just a handful of the total product; giving rise a hypothesis: the top selling-products will be contained in the smaller of the two groups.

There are no missing values in any of the data sets used in this study, except for the case when a product does not have any reviews. Without any reviews, there is no information on the percentage of reviews which recommend the product, review text, review date and the average review rating. Imputation of variables for products without any reviews will be explained in the next section.

## **4. Methods**

### ***4.1 Data pre-processing***

The first step in pre-processing was to clean the raw web scraped data, this involved un-nesting lists and filling empty numeric variables with 0, i.e when a product has no reviews, nothing is scraped and therefore that variable is left empty. The choice to impute the number of reviews as 0 is obvious, however, the decision to impute empty values for average star rating is more difficult; filling them with 0 would mean giving the products an average of 0 stars when nobody has actually rated the product with 0 stars, an unrated product would be more attractive than a product with many 0 star ratings. On the other hand, filling them with the mean or median would be misleading, so in this case they were filled with 0. The next step was to join the web scraped data to the scanner data, products which did not intersect between the two datasets were dropped, along with any duplicate products defined by having the same product id and number of reviews. The last step in pre-processing was to standardise the numeric variables with the Z-score, this transforms each variable to have a mean of 0 and a standard deviation of 1.

### ***4.2 Feature engineering***

Feature engineering involves the creation of additional features from the original features. Two new features were created from the review rating of each review: the number of negative reviews and the number of positive reviews. Negative reviews are important here as generally products are given good star ratings. When there is a large volume of reviews, negative reviews are unlikely to be reflected by the average star rating and hence a count of the number of negative reviews can be insightful (Chong et al. 2016). A positive review is classed as being 4 or 5 stars, a negative review is classed as 1 or 2 stars; reviews of 3 stars were classed as neutral, although a feature for the number of neutral reviews was not created. Chong et al. (2016) found that the relationship between the number of reviews and review sentiment could be used as a strong feature in predicting online sales via reviews. Following the findings from that study, the following features were created: (Rvol/Price), (Rvol/%Recommendation), (Rvol/PosRvol) and (Rvol/NegRvol). Where Rvolume is the total number of reviews, PosRvol is the total number of positive reviews and NegRvol is the total number of negative reviews.

Principal component analysis (PCA) is a technique that can be used to reduce the number of variables in a data set, whilst retaining important information. This is done by creating uncorrelated variables that maximise variance (Jolliffe and Cadima 2016). PCA can be useful to apply before clustering data as most clustering algorithms do not perform well with high dimensionality. PCA was done to find the primary (PC1) and secondary principal components (PC2), the input to PCA was ten features including the engineered features, the features that went into the engineered features and product price.

### ***4.3 Feature selection***

Feature selection is the process of investigating which features in a dataset are most important in solving the problem at hand. The importance can be measured by how much a feature influences finding the correct solution to the problem. When clustering is the chosen method of solving a problem, it is important to select a subset of features in the process of feature selection. Clustering data with too many features can lead to clusters that are poorly defined and do not give a meaningful representation of similarities and differences of the data points. This happens as true clusters frequently vary with respect to a small quantity of features, and therefore, using all the features can suppress the formation of the true clusters (Witten and Tibshirani 2010). The fewer features used in modelling make interpretation at the later stages more straightforward, as there are fewer model inputs to analyse against the model output. It would be challenging to create a visualisation which aids the interpretation of clusters when many features are input to the clustering model.

PCA is a useful method of feature selection for unsupervised problems as the method does not require priori knowledge of the target variable for prediction (Dash and Liu 2000). More common methods of feature selection require this priori knowledge as the importance of the features can be determined by analysing coefficients with each feature when fitting models to the target variable. Fortunately, access to the retailer’s scanner data provides knowledge of the target variable, the quantity of sales. This allows both supervised and non-supervised methods of feature selection to be applicable in this study. In order to compare the events of having access to scanner data and not having access to scanner data, features will be selected for clustering using a multiple linear regression model, and also with dimensionality reduction using PCA. The results of both models can then be compared to clarify the influence priori knowledge of the target variable has on determining which are the best features to use for clustering. Nevertheless, in most cases when target variable data is available supervised learning methods would be preferable over non-supervised methods.

Multiple linear regression is a technique used to approximate the relationship that occurs between multiple explanatory variables and the target variable. This technique attempts to fit a straight line through the data points.

For = observations:

= target variable, = explanatory variable, = y-intercept, = coefficients of slope and = error (residual).

Ordinary least squares (OLS) is an optimisation method used to estimate the parameters of linear regression by minimising the sum of squared residuals (Weisberg 2005). The residuals are the distance between each point of data and the straight line modelled to fit through the data points. The minimum sum of the squared residuals is found when the model which best represents the relationship between the explanatory variables and the target variable has been produced. Each explanatory variable in the multiple linear regression model will have a coefficient allocated; the magnitude of which corresponds to how much the target variable would change when the explanatory variable increases by 1 and the other explanatory variables are unchanged, the sign of the coefficient determines whether that change will be positive or negative.

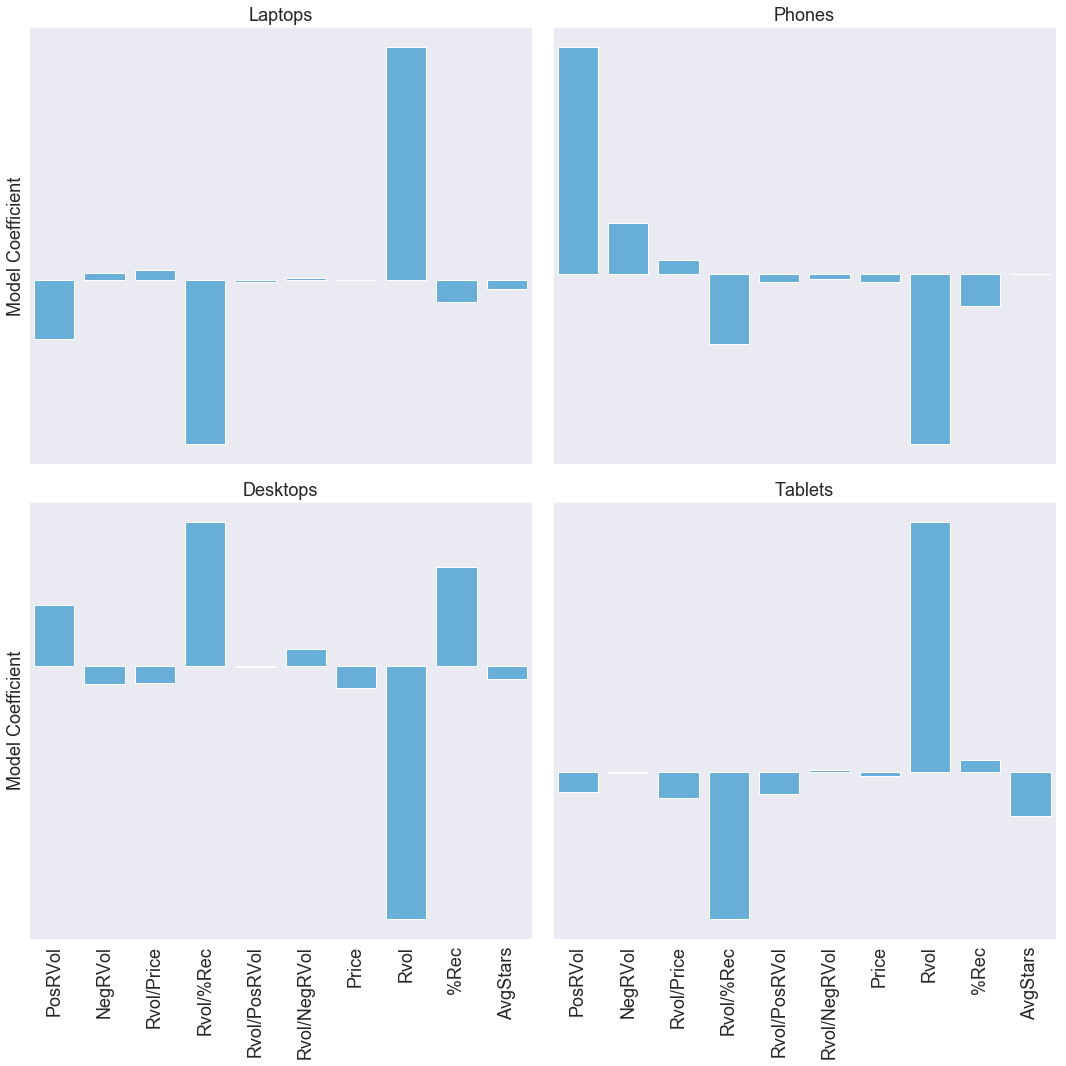


Figure 3: Feature importances obtained from OLS. Note: coefficient values have been removed for readability, coefficient magnitudes should be taken as the length of the bars.

Figure 3 shows that for all categories, Rvol is the most important variable when predicting the quantity of sales. For laptops and tablets the coefficient is positive, however, for phones and desktops the coefficient is negative; a high number of reviews would predict a low number of sales. This could suggest that the majority of the reviews are negative, in this case they are not, it is because there are products without any reviews with high sales quantities in both of these datasets. In fact, almost all the coefficient magnitudes are unstable between the four categories, with exception to Rvol/%rec. This variable remains important in all four datasets, although for desktops the coefficient is positive whereas it is negative for the other three. Overall, variables which have the highest average magnitude between all four categories, regardless of sign, are: Rvol, Rvol/%Rec and PosRvol. Regarding the p-values of each coefficient in table 2, there are no features which are consistently 95% significant in all four categories, although, price is in all categories but tablets. Rvol/Price has a p-value of 0 for both laptops and phones. This analysis of feature importance does not provide conclusive evidence on which are the best features to use in clustering.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feature** | **Laptops** | **Phones** | **Desktops** | **Tablets** |
| PosRVol | 0.029 | 0.031 | 0.73 | 0.72 |
| NegRVol | 0.011 | 0.00 | 0.66 | 0.73 |
| Rvol/Price | 0.00 | 0.00 | 0.76 | 0.065 |
| Rvol/%Rec | 0.003 | 0.42 | 0.20 | 0.57 |
| Rvol/PosR | 0.037 | 0.14 | 0.70 | 0.52 |
| Rvol/NegR | 0.050 | 0.088 | 0.60 | 0.84 |
| Price | 0.031 | 0.00 | 0.028 | 0.14 |
| Rvol | 0.024 | 0.36 | 0.40 | 0.45 |
| %Rec | 0.032 | 0.36 | 0.22 | 0.88 |
| AvgStars | 0.094 | 0.98 | 0.9 | 0.45 |

Table 2: P-values of the coefficients of each feature derived the multiple linear regression models.

Logistic regression, like multiple linear regression, is a method of regression analysis. However, logistic regression is used to predict a binary target variable, whereas linear regression predicts a continuous target variable. Since this study has transformed the problem of estimating expenditure (continuous) to estimating whether a product is a top-seller or not (binary), it would be appropriate to analyse feature importance with logistic regression as well as linear regression.

= probability of a 1, yields when =0 and determines the rate at which probability changes with changing a single unit.

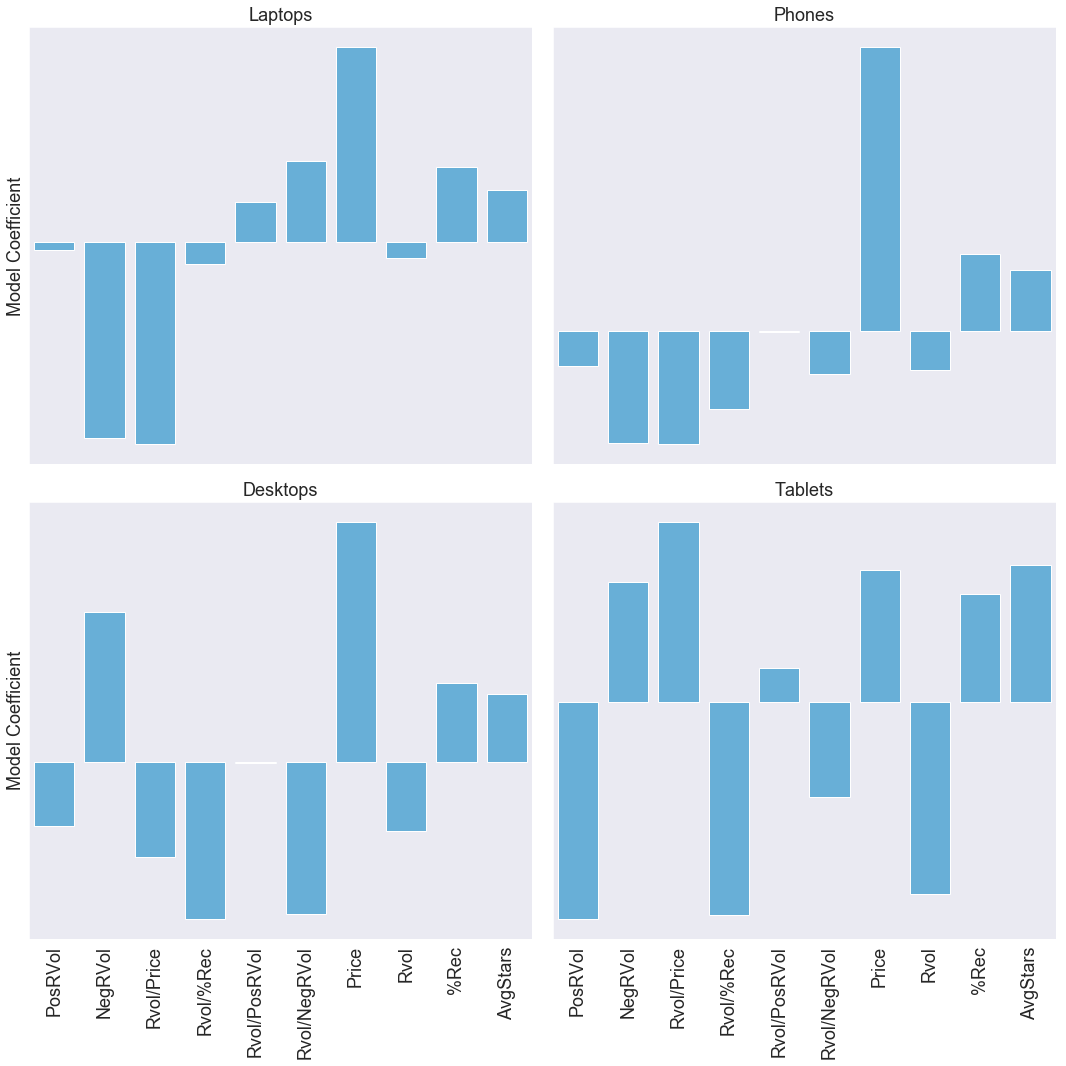


Figure 4: Feature importance’s obtained from logistic regression. Note: coefficient values have been removed for readability, coefficient magnitudes should be taken as the length of the bars.

The logistic regression model coefficients in figure 4 differ between categories, and it is difficult to see clear similarities yet there is interpretable information. Positive coefficients are in favour of producing an output of 1, which in this case means not a top-selling product, whereas negative coefficients are in favour of producing an output of 0: a top-selling product. For all categories Price has a large positive coefficient, meaning a product with a high price is less likely to be a top seller. The coefficients for Rvol and Rvol/&Rec are negative for all categories, although, the magnitude of the coefficients vary. Features AvgStars and %Rec have positive coefficients for all categories, these features would be expected to influence the purchase of a product, though not reflected by this model. This feature importance analysis has proved more useful than the previous, providing evidence that the features: Price, Rvol and Rvol/%Rec are the top three most suitable features to use in clustering.

### ***4.3 Clustering***

Clustering is a form of unsupervised learning, which is the process of finding similarities and differences in datasets without the use of training data (target variable). DBSCAN is an unsupervised learning clustering algorithm which derives clusters based on the density of data points (Ester et al. 1996). This algorithm was designed to perform effectively with large datasets and to have the ability to discover clusters of arbitrary shapes, something which other clustering methods do not perform well at. There are two input parameters to the DBSCAN algorithm: and . Each data point will have a neighbourhood with radius ; to be a neighbouring data point to another data point, a data point must lie within radius . For a group of neighbouring data points to become a cluster, there must be at least neighbouring data points. The code proposed by Ester et al (1996) states that the neighbourhood of a datapoint is defined as:

And that if:

Then data point is not inside a cluster and thus is classed as noise. The noise points are not necessarily clustered; a noise point could be of infinite distance from another noise point, however, they are all labelled the same. In most scenarios when using this algorithm, the noise points would be disregarded, however, in this study noise points could be considered as the objective. When parameterising the algorithm using a high value of and a small enough value for , it is possible to form just one cluster. Then the data will be split into two groups: data points in cluster 1 and data points classed as noise. This proves to be a method of separating the datasets into two groups, whereby if the hypothesis stated earlier is correct: the top selling-products will be contained in the smaller of the two groups; the smaller group will contain the top-selling products. When parameterising this algorithm, should scale with the number of products in a dataset because there cannot be a generic value for alternative sized datasets as the goal is to produce only one cluster. It was identified in the literature review that decreasing results in more data points being classed as outliers. Figure 1 demonstrates the importance of this, as only a small proportion of the products in each of the four datasets are top-sellers, and it is likely that datasets from other retailers will have a small proportion of top-sellers too, hence the value of should be chosen to control the number of data points classed as outliers.

A variety of feature combinations have been used to split the datasets into two groups using DBSCAN, these are: (PC1, PC2), (Rvol, Price, Rvol/%Rec), (Rvol, Price), (Rvol, Rvol/%Rec) and (Rvol/%Rec, Price). The true sales quantities from scanner data are used to evaluate the performance of each model. Although a rough method as the data does fit a normal distribution, the standard score (Z-score) for the quantity of sales was calculated to classify products. Products with a Z-score greater than or equal to 2 are classified as top-selling products.

Two approaches were conducted to find the optimal parameters, the first was to use a Monte Carlo grid search with 5000 iterations. With each search, the models were evaluated against the sales quantities in the scanner data using the following metrics: precision, recall and F1 score. The range tested was 0.2 to 2 in intervals of 0.05 and the range of was tested in proportion to the number of products in the dataset, proportions tested were: /2, /3, /4, /5, /6, /7 and /8. The outcome is the parameter set that has achieved the highest mean F1 score (this is calculated from the F1 scores attained from the four datasets) returned by the grid search, whereby the parameters tested are kept the same for each dataset.

The second approach uses the same grid search as the previous approach, however, instead of evaluating the models with the scanner data the grid search returns the model based on the proportion of the total number of products which are classed as outliers, whereby the grid search looks for /10 to /5 (10-20%) outliers, and picks the model with the lowest value of , which makes clustering easier and hence outliers more difficult to find.

## **5. Results**

### ***5.1 Evaluating with scanner data***

Firstly, this section focuses on comparing the performance of the DBSCAN algorithm with each feature combination, obtaining the parameters which give the best overall mean F1 score derived from the four datasets.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Features** |  |  | **Mean Precision** | **Mean Recall** | **Mean F1 Score** |
| Rvol, Price , Rvol/%Rec | 1.10 | /4 | 0.43 | 0.73 | 0.54 |
| Rvol, Price | 0.80 | /5 | 0.38 | 0.73 | 0.49 |
| Rvol, Rvol/%Rec | 0.95 | /5 | 0.60 | 0.68 | 0.62 |
| Rvol/%Rec, Price | 0.85 | /5 | 0.42 | 0.70 | 0.52 |
| PC1, PC2 | 1.10 | /7 | 0.70 | 0.41 | 0.52 |

Table 3: The parameters which achieved the best mean F1 scores for each feature set.

When studying table 3, it is evident that each set of features performs with varying success. The highest mean F1 score, 0.62, is obtained using features (Rvol and Rvol/%Rec) making this feature set the best balanced. However, this set of features does not have the highest mean precision or recall, although there is a small difference between the two metrics; other feature sets have considerably large differences between mean precision and mean recall. Feature sets (Rvol, Price and Rvol/%Rec) and (Rvol and Price) both scored the highest mean recall score of 0.73. Feature set (PC1, and PC2) scored the highest mean precision score of 0.70, which is significantly greater than the second highest precision score, 0.60, achieved by feature set (Rvol and Rvol/%Rec). A mean precision of 0.70 across the four datasets is a good score, meaning that there are significantly more true top-selling products being identified than not true top-selling products being falsely identified as top sellers. On the other hand, this feature set scored the lowest mean recall of 0.41, meaning that it misclassified many products as top sellers when they are not.

**Features (Rvol and Rvol/%Rec) parameterised using scanner data**

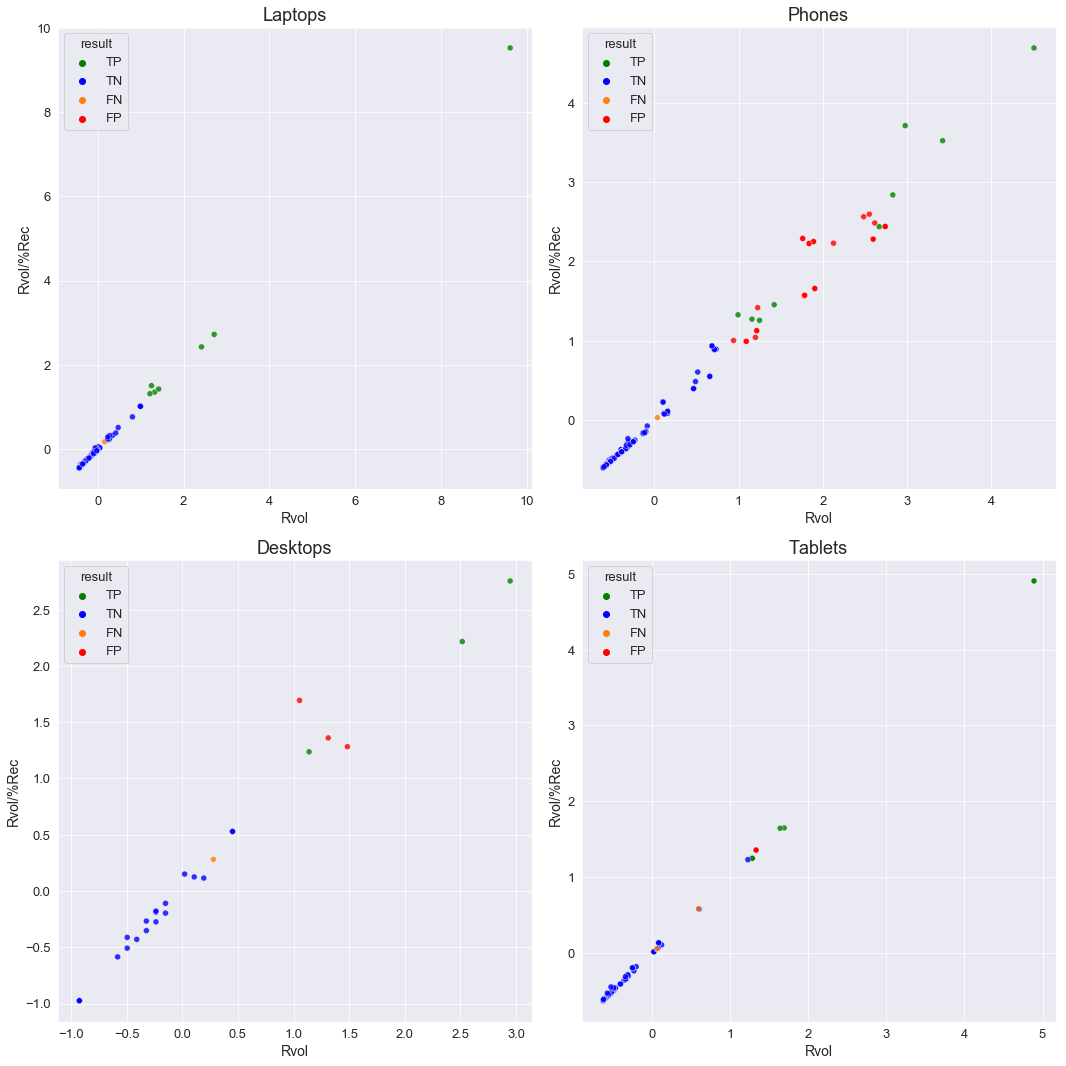


Figure 5: Scatter plots showing the DBSCAN results for features (Rvol and Rvol/%Rec) parameterised using scanner data.

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Precision** | **Recall** | **F1-Score** |
| laptops | 1.00 | 0.88 | 0.93 |
| phones | 0.26 | 0.61 | 0.37 |
| desktops | 0.50 | 0.60 | 0.55 |
| tablets | 0.64 | 0.64 | 0.64 |

Table 4: Metrics for DBSCAN results for features (Rvol and Rvol/%Rec) parameterised using scanner data.

Looking at figure 5 and table 4, it is immediately clear that laptops performed very well. All top-selling products were identified by the algorithm without a false positive, there was 1 false negative found, this was because this product was a top seller but did not have any reviews. Phones did not perform well here, with most top sellers being incorrectly identified given by the precision score of 0.26. A number of false positives can be seen at an Rvol of 1 or greater, many of these points are moderately densely populated and therefore would need a lower value of to class them as outliers, although, this may change some true negatives to false positives. Overall, this combination of selected features and parameters scored the highest mean F1-score because the algorithm performed with near perfection on laptops, moderately on desktops and tablets, and poorly on phones. These results show that this set of parameters does not provide a one-fits all solution to all four datasets, but it is possible to get very good performance from the algorithm. Figure 5 provides a clear representation of when the data points of bestselling products are sparsely distributed, the algorithm performs well; and poor performance occurs when they are more densely distributed.

**Features (PC1 and PC2) parameterised using scanner data**

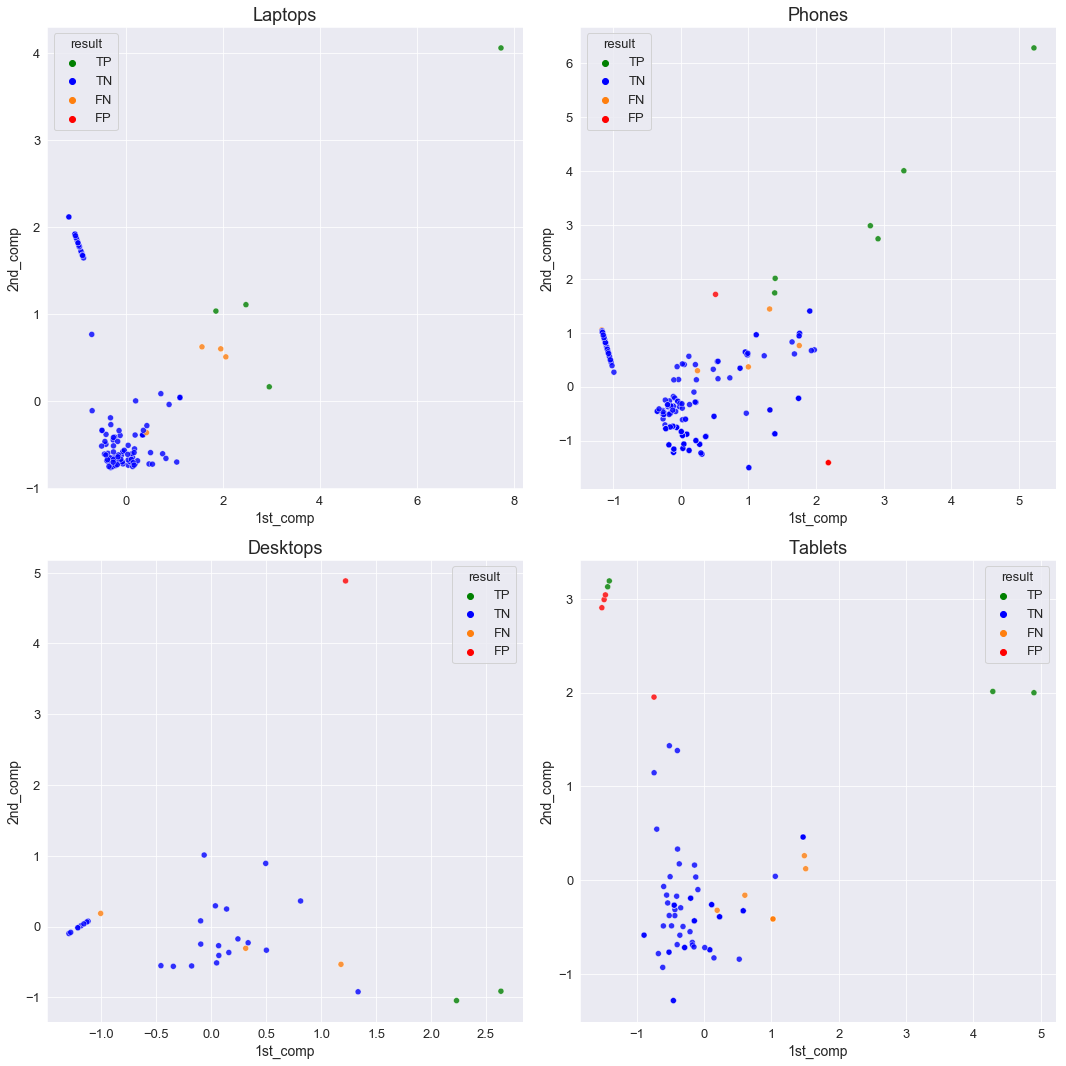


Figure 6: Scatterplots showing the DBSCAN results for features (PC1 and PC2) parameterised using scanner data

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | **Precision** | **Recall** | **F1-Score** |
| laptops | 1.00 | 0.50 | 0.67 |
| phones | 0.64 | 0.39 | 0.48 |
| desktops | 0.67 | 0.40 | 0.50 |
| tablets | 0.50 | 0.36 | 0.42 |

Table 5: Metrics for DBSCAN results for features (PC1 and PC2) parameterised using scanner data.

There are some similarities and differences from the performance given when using features (PC1 and PC2) compared to the previous feature set. Yet again the algorithm performs with complete precision on laptops, although now with four times as many false negatives. The precision of the algorithm on phones is significantly greater than the previous, but not without cost as the recall score is also less. The performance of the algorithm on desktops and laptops is slightly reduced than with the previous features. Focusing on only the precision and recall scores, when using the primary and secondary principal components (PC1 and PC2) of all the available features, DBSCAN performs with greater precision than it does recall, as there are no precision scores below 0.5, and no recall scores above 0.5.

### ***5.2 Evaluating without scanner data***

This section focuses on the results which are found without any reliance on using scanner data to find the best performance. Here models have been selected based on finding a certain proportion of outliers with a model that minimises the parameter . The two feature combinations (Rvol and Rvol/%Rec) and (PC1 and PC2) are analysed in this section.

**Features (Rvol and Rvol/%Rec) parameterised without using scanner data**

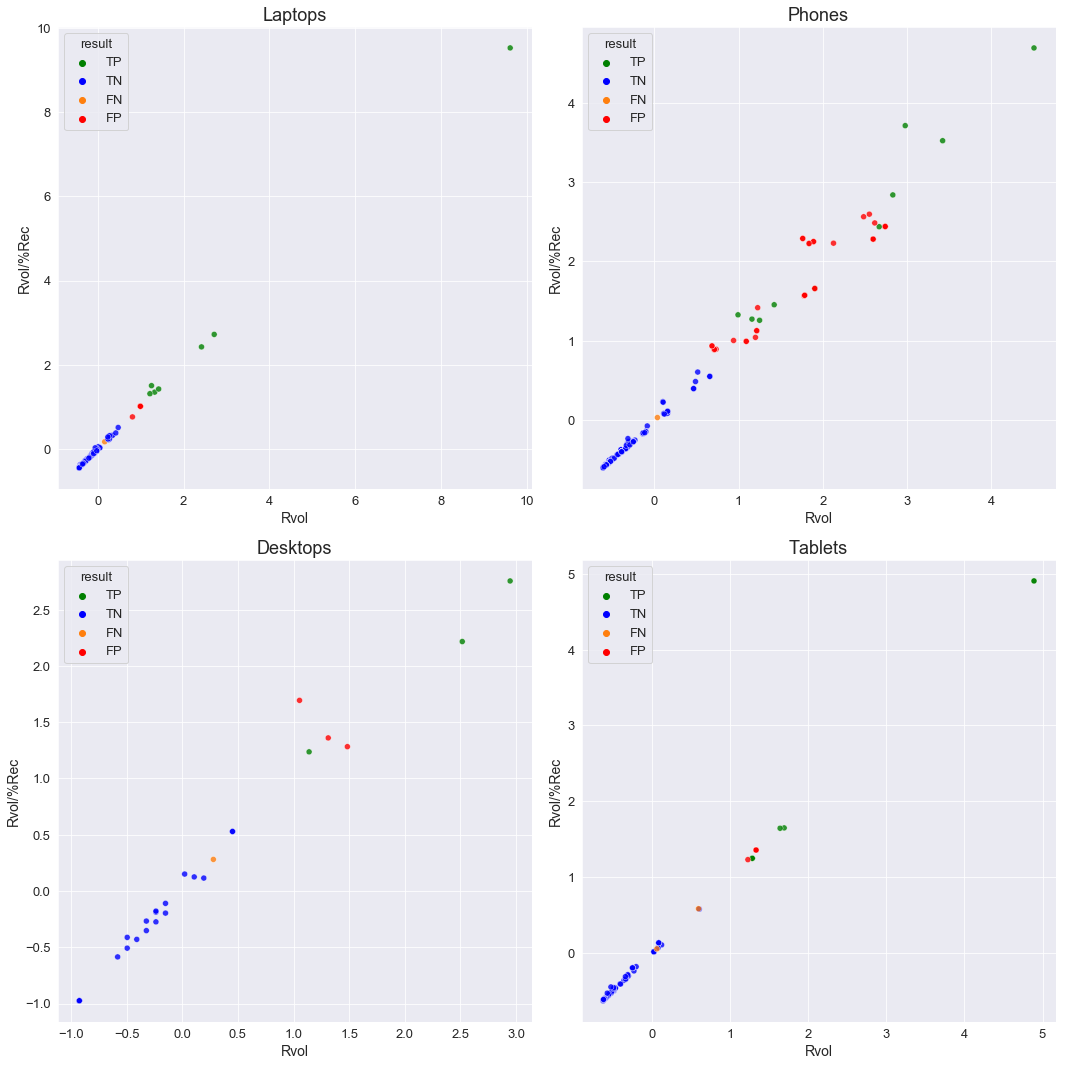
******

Figure 7: Metrics for DBSCAN results for features (Rvol and Rvol/%Rec) parameterised without scanner data.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Category** |  |  | **Outlier Proportion** | **Precision** | **Recall** | **F1-Score** |
| laptops | 0.35 | /7 | 10 | 0.50 | 0.88 | 0.64 |
| phones | 0.80 | /7 | 18 | 0.23 | 0.61 | 0.34 |
| desktops | 0.80 | /7 | 14 | 0.58 | 0.64 | 0.61 |
| tablets | 0.55 | /7 | 18 | 0.50 | 0.60 | 0.55 |

Table 6: Metrics for DBSCAN results for features (Rvol and Rvol/%Rec) parameterised without scanner data.

This section compares the performance of the models produced using the feature set (Rvol and Rvol/%Rec) with both approaches of parameterisation by studying the metrics in table 4 and table 6. Overall, the models parameterised using the proportion of outliers, present worse results than the previous. Notably laptops are still the best performing but with a precision score of 0.5, half the value of the previous. Again, the weakest results are produced by the phones dataset, with a slightly reduced precision score to the previous case. All the precision scores are reduced, with the exception of desktops where there is a 0.08 increase. The recall scores for laptops and phones are unchanged, those for desktops have increased by 0.04 and tablets decreased by 0.04. In general, the models produced with the second approach of parameterisation are not as effective at determining top-selling products as those obtained from the first approach, as there are an increased number of false positives, though an almost identical number of false negatives. This is because these models must produce a number of outliers that is at least a tenth of the total number of data points, thus potentially increasing the favour of classifying points as outliers, whereas the first approach of parameterisation has no bound on the number of outliers produced. In this respect, the first approach is less likely to produce as many false positives as the second.

**Features (PC1 and PC2) parameterised without using scanner data**

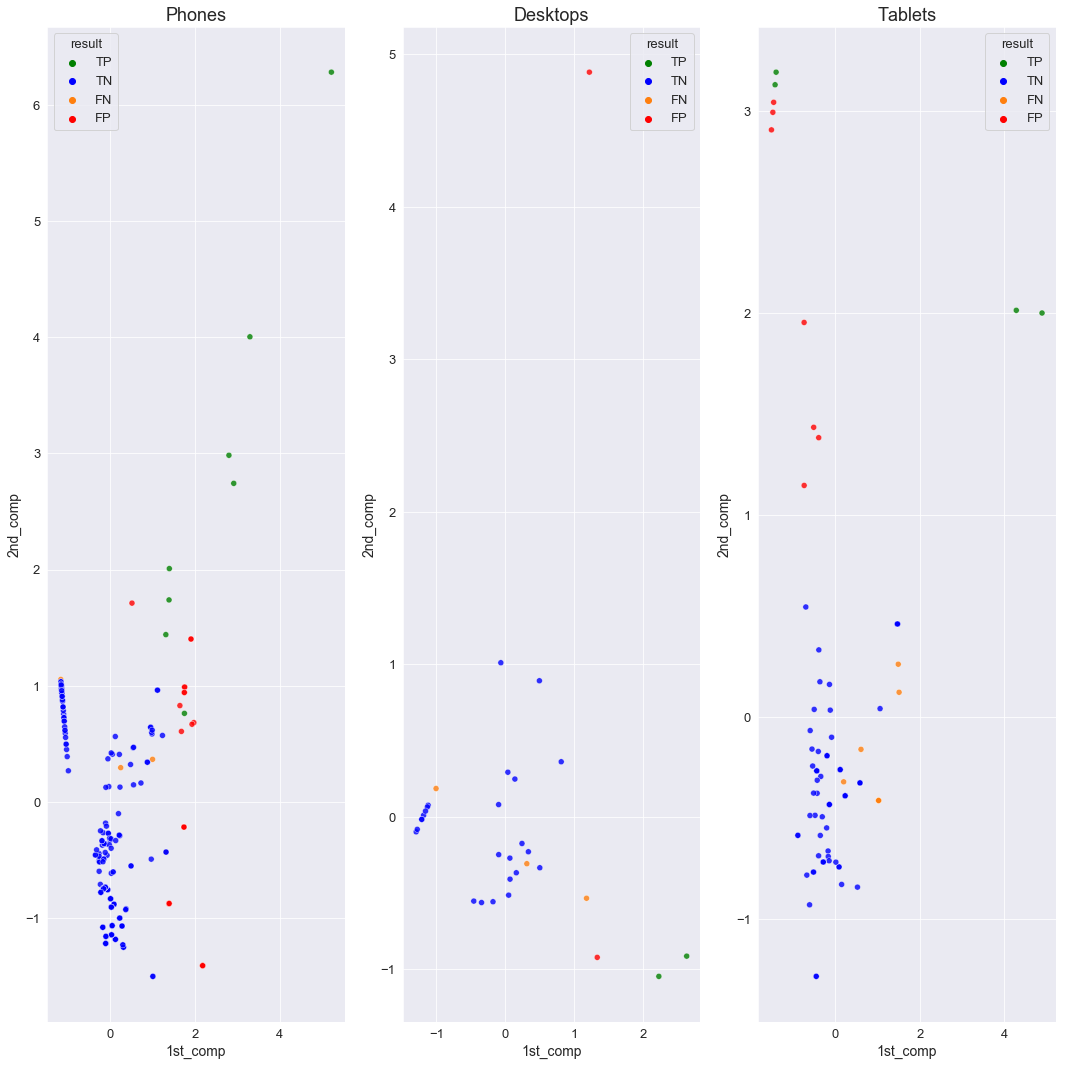
******

Figure 8: Metrics for DBSCAN results for features (PC1 and PC2) parameterised without scanner data.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Category** |  |  | **Outlier Proportion** | **Precision** | **Recall** | **F1-Score** |
| phones | 0.90 | /7 | 12 | 0.28 | 0.50 | 0.34 |
| desktops | 0.95 | /6 | 12 | 0.50 | 0.40 | 0.45 |
| tablets | 0.80 | /7 | 13 | 0.36 | 0.36 | 0.36 |

Table 7: Metrics for DBSCAN results for features (PC1 and PC2) parameterised without scanner data.

This section follows on from the previous, but now analysing models produced with the feature set (PC1, PC2). The first point to note is that no model could be produced for laptops with a number of outliers in the required range, and hence there are no results shown for that dataset. The metrics in table 7 show the lowest F1 scores from all the models analysed so far, performing less well than the counterpart models that were parameterised with the scanner data. The increased number of false positives have resulted in low precision scores, notably phones with a score of 0.28. The recall scores for desktops and tablets are the same as those for the counterpart models, and the recall score for phones is 0.11 higher, with half as many false negatives as the counterpart model. The graphs on figure 8 show that the false negatives are contained within the main cluster of points and therefore are impossible for this algorithm to detect them as outliers. Most false positives show similar properties to the true positives and so they have been classed as outliers, although they are not top-selling products.

## ***5.3 Combining results***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Rvol, Rvol/%Rec - scanner data | Rvol, Rvol/%Rec - no scanner data | PC1, PC2 - scanner data | PC1, PC2 - no scanner data |
| laptops | 0.93 | 0.64 | 0.67 | No results |
| phones | 0.37 | 0.34 | 0.48 | 0.34 |
| desktops | 0.55 | 0.61 | 0.50 | 0.45 |
| tablets | 0.64 | 0.55 | 0.42 | 0.36 |

Table 8: The F1 scores achieved from the four models analysed.

Overall, the models performed best when clustering with only two features. The best performance was achieved using features (Rvol and Rvol/%Rec) for all datasets except for phones, where the best performance was achieved using (PC1 and PC2). The F1 scores produced are lower when parameterising the model using the proportion of outliers in comparison to using true sales quantities from the scanner data, except for when using (Rvol, Rvol/%Rec) on desktops, the F1 score achieved is 0.06 higher than the scanner data equivalent. On the whole, the best F1 scores were achieved on the laptops data, although no model was found for the last case.

## **6. Conclusion and future work**

### ***6.1 Conclusion***

Finding the top sellers in each of the four datasets was completed with varying success. There was no single set of features combined with a fixed set of parameters which when used to cluster the data would produce the best results for all of the four datasets. Optimal results require parameterisation geared for that dataset, no best all-around set of parameters were found. As expected, the best results were found when using the expenditure information from the scanner data to parameterise the algorithm; however, the aim of this study is to look into proxying expenditure without the use of any data on expenditure. The scanner data was beneficial as it provided an opportunity to evaluate the models produced and was used to find the optimal features required for modelling. It was found that features (Rvol and Rvol/%Rec) produced the best results except for the phones dataset, where performance was significantly lower than performance for the other datasets, even though the phones dataset is considerably larger than the others. One of the reasons for poor performance is because the majority of expenditure was from a small number of the total products available and the algorithm was finding too many outliers in the data, and therefore, produced substantially more false positives than found with the other datasets. Nevertheless, just because features (Rvol and Rvol/%Rec) generally gave the best performance, it does not mean this would be replicated with a different retailer’s data. It is difficult to evaluate the impact of false negatives and false positives given by recall and precision respectively, without knowing which products are being misclassified, for example, a product with a very high number of sales being classified as a non-top seller would be more detrimental than a product slightly under the top seller threshold being classified as a top seller. If this method was used in a production environment, there would not be data on product expenditure, and hence no way to tell the difference.

There are three drawbacks of the methods demonstrated in this study which should be highlighted. Firstly, there are online retailers which do not facilitate the reviewing of products, and therefore, this method or one alike would be unusable with web scraped data from such a retailer. Secondly, a new product in an online store may sell more units than a product which is not new, but the number of reviews of the new product is likely to be significantly lower than that of the other; though this could be negated by only using the number of reviews in a certain timespan, i.e the past month. Thirdly, certain items might be less likely to be reviewed even if they are frequently purchased. Although the models produced in this study labelled outliers to top selling products with varying success, the findings have highlighted the potential usefulness that product reviews have on proxying the expenditure of products, an area that so far has received little research, whereby studies without the use of historic data have found limited success. The findings of this study could be a step in the right direction, providing a number of opportunities for further research on using customer reviews for proxying expenditure.

### ***6.2 Future work***

This section will be split into two areas: aspects where the current method could be improved and techniques which would follow the current method. There is potential to find new features to test in modelling, the text content of the reviews was not used in this study to create a sentiment feature using natural-language processing. Provided with review star ratings and an average star rating for each product, the need for review sentiment was overlooked; though Chong et al. (2016) found it was beneficial to use a review sentiment feature. In this study, PCA was carried out on one set of features only. Perhaps the models created using the PCA components would have performed better if PCA was completed on a lower number of features, if so, it would be greatly beneficial for finding the top-selling products for other online retailers as PCA is a non-supervised algorithm, and thus, does not require expenditure information. Different DBSCAN algorithms could be tested, here the DBSCAN method from the Scikit-Learn python library had been chosen, which is the standard method. There are more advanced versions of the algorithm presented by Khan et al. (2014), which have proved more effective than the standard method. The precision scores of the models parameterised without scanner data could be improved by using a lower outlier proportion threshold, here a range of 10 - 20% of outliers was chosen as this was the range that included the proportion of top sellers in the scanner datasets. Lowering this threshold would mean that only the most extreme outliers would be picked, lowering the chance of producing false positives. On the other hand, it could introduce more false negatives. As discussed in the previous section, reviews could be broken down by review date, either introducing a weight on the number of reviews (older reviews with less weight than newer reviews) or solely using the number of reviews posted in a recent timespan.

Frequent scraping of reviews would open more possibilities for analysis and feature creation. Firstly, new products can be identified and a feature assigned that relates to the age, i.e when it was first scraped. It would be expected that a new product without any reviews would sell more than an old product without any reviews. With access to time-series review data and scanner data for the same period, the true CPI and estimated CPI (using only products marked as outliers) can be calculated and compared. Experimentation with other methods of outlier detection could be combined with the method presented in this study, which may improve the robustness of the results. Additionally, a generalised supervised learning model could be trained using all categories of tech goods, this could uncover less obvious qualities that are shared between top sellers of different types of tech goods.

### ***6.3 Reflection***

This was the first time I had conducted a project in an area with limited research, proxying expenditure without historical data or direct links to sales quantities, i.e category best-sellers ranks. This presented a great challenge to me, without the availability of highly relevant previous research, it required me to research in different fields for each aspect of the project and synthesise my discoveries; in the process, I learned many valuable research skills. With little available past research on methods of approximating expenditure without historical data, many potential methods found became obsolete on application. I spent time learning to understand why these methods would not work to solve my problem and used this knowledge to search for alternative approaches. I found that initial processes and ideas can result in being inappropriate, and therefore, time should be allocated to allow the formulation of a fresher approach. Initially, I set out with the problem of approximating the relative expenditure of each product in its category, after many unsuccessful attempts I chose to convert the problem to separating products into two groups of relative expenditure. Here, I believe I learned the greatest lesson: no matter one’s desire to achieve a particular goal, sometimes it is best to rescope to one which progressively meets that goal.

## **References**

Aggarwal, C.C. 2017. An introduction to outlier analysis. In *Outlier analysis* (pp. 1-34). Springer, Cham.

Chessa, A.G. and Griffioen, R. 2019. Comparing price indices of clothing and footwear for scanner data and web scraped data. *Economie et Statistique* *509*(1), pp.49-68.

Chong, A.Y.L., Li, B., Ngai, E.W., Ch'ng, E. and Lee, F. 2016. Predicting online product sales via online reviews, sentiments, and promotion strategies: A big data architecture and neural network approach. *International Journal of Operations & Production Management*.

Cui, G., Lui, H.K. and Guo, X. 2012. The effect of online consumer reviews on new product sales. *International Journal of Electronic Commerce*, *17*(1), pp.39-58.

Dash, M. and Liu, H. 2000, April. Feature selection for clustering. In *Pacific-Asia Conference on knowledge discovery and data mining* (pp. 110-121). Springer, Berlin, Heidelberg.

*Data Protection Act* 2018. c. 12. Available at: <https://www.legislation.gov.uk/ukpga/2018/12/contents/enacted> [Accessed: 21/10/21].

Ester, M., Kriegel, H.P., Sander, J. and Xu, X. 1996, August. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd* (Vol. 96, No. 34, pp. 226-231).

Floyd, K., Freling, R., Alhoqail, S., Cho, H.Y. and Freling, T. 2014. How online product reviews affect retail sales: A meta-analysis. *Journal of Retailing*, *90*(2), pp.217-232.

Gaf, B. 2020. *Consumer Price Index Manual.* International Monetary Fund.

Ghose, A. and Ipeirotis, P.G., 2010. Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics. *IEEE transactions on knowledge and data engineering*, *23*(10), pp.1498-1512.

Jolliffe, I.T. and Cadima, J. 2016. Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *374*(2065), p.20150202.

Khan, K., Rehman, S.U., Aziz, K., Fong, S. and Sarasvady, S. 2014, February. DBSCAN: Past, present and future. In *The fifth international conference on the applications of digital information and web technologies (ICADIWT 2014)* (pp. 232-238). IEEE.

Krotov, V. and Silva, L. 2018. Legality and ethics of web scraping. *Emergent Research Forum (ERF).*

Nowak-Brzezińska, A. and Xięski, T. 2017. Outlier mining using the DBSCAN algorithm. *Journal of Applied Computer Science*, *25*(2), pp.53-68.

ONS. 2017. *Consumer Price Inflation*. Available at: <https://www.ons.gov.uk/economy/inflationandpriceindices/methodologies/consumerpriceinflationincludesall3indicescpihcpiandrpiqmi> [Accessed: 14/10/21].

ONS. 2020. *Web Scraping Policy.* Available at: <https://www.ons.gov.uk/aboutus/transparencyandgovernance/datastrategy/datapolicies/webscrapingpolicy> [Accessed: 09/09/21].

Sands, H. 2020. *Using Statistical Distribution to Estimate Weights for Web-scraped Price Quotes in Consumer Price Statistics.* Available at: <https://www.ons.gov.uk/economy/inflationandpriceindices/articles/usingstatisticaldistributionstoestimateweightsforwebscrapedpricequotesinconsumerpricestatistics/2020-09-01> [Accessed: 12/07/21].

Weisberg, S. 2005. *Applied linear regression* (Vol. 528). John Wiley & Sons.

Witten, D.M. and Tibshirani, R. 2010. A framework for feature selection in clustering. *Journal of the American Statistical Association*, *105*(490), pp.713-726.