DSA3101 Assignment 2 Report

Group 2: Cai Anqi, Chan Yu Hang, Chin Synn Khee Joash, Chua Cheng Ling, Chua Hua Ren, Clarence Ong

1. Introduction

In this report we will use the following techniques using transaction data collected over the past 3 years to help guide clients to make strategic business decisions to improve profitability.

- Customer Segmentation using K-Means
- Customer Segmentation using Hierarchical Clustering
- Principal Component Analysis (PCA)
- Recommendation Engines

We hope to be able to target these 4 key problem statements and give the client clear and concise recommendations to grow their business.

1. What are the different customer profiles in our customer database?
2. What are the most important variables for deriving actionable insights?
3. What products should we promote to our customers and why?
4. Which products do we want to recommend to your customers?

In the later parts we will be further exploring and elaborating on these ideas and key questions.

We are provided with 3 datasets, namely DSA3101_Hackathon_Categories_Information.csv , DSA3101_Hackathon_Data.csv and DSA3101_Hackathon_Panelists_Demographics.csv .

2. Imputation of Missing Data

In the previous assignment, our group noticed that there was 49,539 instances of missing/misrecorded data. Our team tried to use kNN imputation, however we were unable to do so and chose to remove the data instead as it was 3.8% of the dataset.

In our analysis this time, we want to create recommendations for individual customers. Dropping all data with 0s in any row would result in around 207 customers losing more than 10% of their purchase data, which could affect the accuracy of suggestions especially if there is some significance to the missing data. However incorrectly imputing data would also lead to inaccurate results. We thus attempt to take a conservative approach towards data imputation, the procedure of which is as follows:

1. We attempt to impute primarily the missing Volume values which account for most of the missing data, by finding candidate "matches" in the rest of the data.
2. We define a match as another row of purchase data having same Category, Pack Size and Spend.
3. For each row of missing data we find all candidate matches by performing a join on the complete data.
4. If there are no matches, we have no basis for imputation and thus drop the row.
5. If all candidates agree (same Volume), we term this as a consistent match and impute this value.

6. For inconsistent matches, we considered a few heuristics, such as taking the match from the same Panel ID, the match with the closest date, or the modal value of all matches. We ultimately chose to impute the modal value, **only if standard deviation of all matched volumes < 0.5**. Ultimately this is a heuristic, but we verified that within this margin, most matches agree thus the modal value is sufficient.
7. We drop rows where standard deviation >= 0.5.
8. From the full dataset, we then drop remaining rows with 0s in Pack Size and Spend, rather than repeat this procedure on the other columns. This is because a) there are only 3859 rows remaining with such missing data and b) repeated imputation means imputation based on imputed data which is more error-prone.

The figure below shows boxplots of the percentage of kept data per customer before and after imputation. For the 207 outliers below 90%, we can clearly see a general increase in the utilized data.

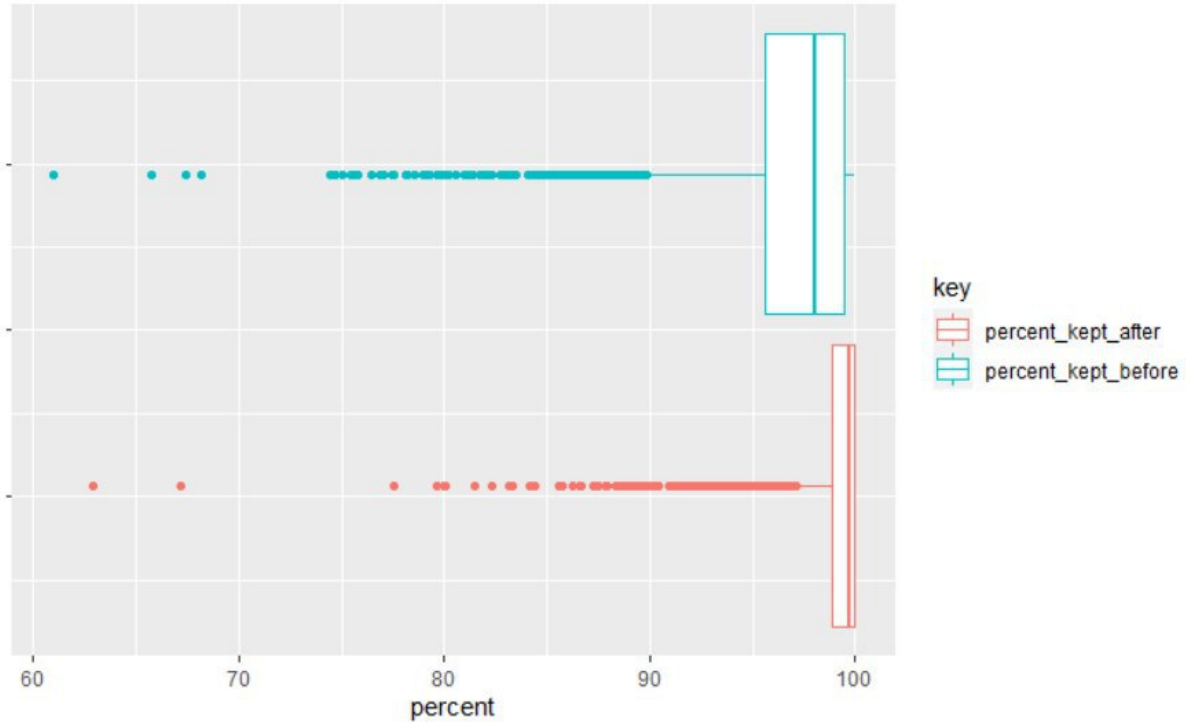


Figure 1

To evaluate the improvements from the imputation, we focus on the 207 customers with significant missing data. The mean increase in total rows kept after imputation was 9.1%. Also, the total customers who still had kept data < 90% decreased to 46. Overall, we conclude that the imputation is quite successful.

3. Hierarchal Clustering and Principal Component Analysis

In order to better understand some of the purchasing habits and patterns of the various customers, the team decided to utilise dimensionality reduction techniques such as Principal Component Analysis (PCA) and Unsupervised Hierarchal Clustering to aid us in our exploration.

Firstly, in the pre-processing stage, the team decided to focus our attention towards customers in the Central location of Malaysia, as the people there have statistically shown to be more affluent and able to afford a larger purchase then most others in the country, as found from our RFM analysis previously. Next, the team decided to represent each unique customer in the central area of Malaysia by the mean spending of that individual in each of the 62 item categories available in the supermarket, over the 3-year period in which data have been made available.

With each unique customer being represented by a 62-length long vector, the team proceeded to conduct our first trial of Principal Component Analysis to compress the 62 variables down to a significantly smaller number of principal components.

However, in doing so, the team noticed that there appears to be a low correlation between the various components in the input vectors, making it difficult to use PCA to significantly reduce the number of dimensions used to represent each customer. This was a severe problem faced by the team as the number of principal components needed to explain at least 70% of the variance in the data was greater than 50, making PCA a poor choice of dimensionality reduction for our current pre-processed dataset.

As such, the team decided to experiment with, and subsequently adopt a 2-layer dimensionality reduction technique that encompasses a first layer of hierarchal clustering, followed by the implementation of principal component analysis on the new pre-processed dataset.

Using Hierarchal Clustering, we first segmented the different categories into 7 larger overarching categories based on the Calories per 100g of each of the 62 original categories. This was done through the use of Euclidean distance to compare the Calories per 100g of the of the various categories and building the dendrogram in an agglomerative manner.

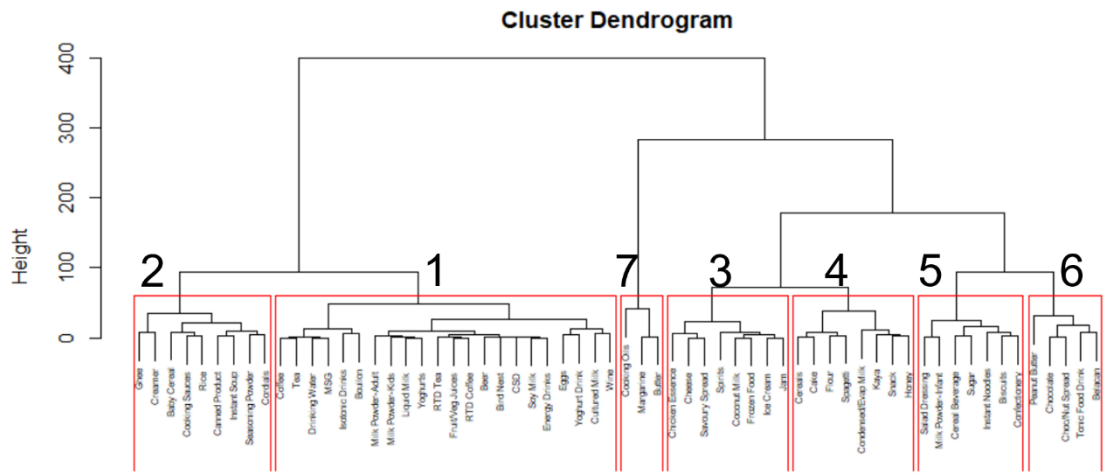


Figure 2

From Figure 1 above, we observe the generated dendrogram of the 62 categories, into 7 distinct clusters, where clusters with a smaller label number value represent categories with lower values of Calories per 100g, and that of a larger label number represent categories with higher values of Calories per 100g. In this way, the team is able to better segmentize the mean spending of the various food items into a smaller number of distinct groups, according to the different ranges of calories that they belong to.

Having carried out the first level of dimensionality reduction through the use of hierarchal clustering, the team sought to challenge ourselves to further reduce the number of dimensions through the use of the Principal Component Analysis.

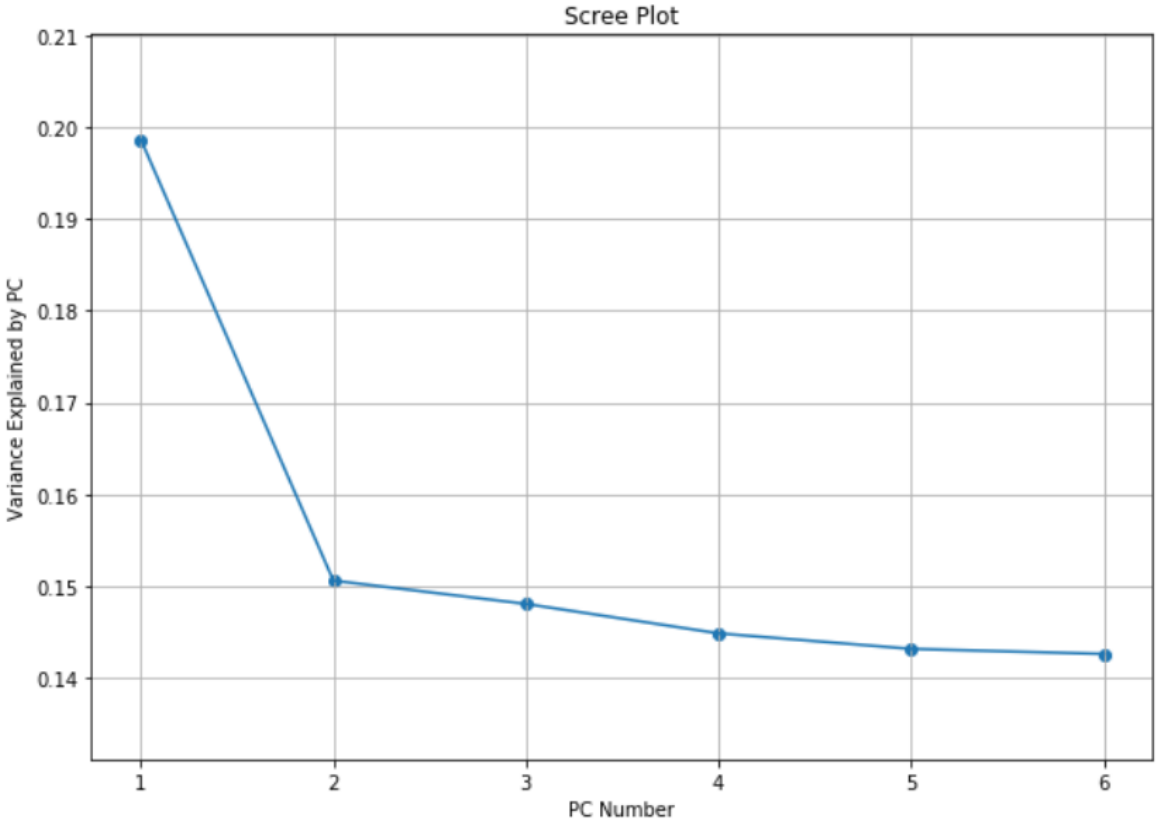


Figure 3

First, by observing the scree plot of the new pre-processed dataset, where each unique customer is now being represented by their mean spending across the 7 new larger cluster of categories, we observe that the desired number of principal components to use, as reflected by the elbow of the plot is 2.

rank	PC1	PC2
1		0.827071
2		
3		
4		
5		0.791783
6	0.711918	
7	0.741586	

Table 1

Using 2 Principal Components for our analysis, it appears that the correlation between the first principal component and the 6th and 7th cluster, as well the correlation between the second principal component and the 1st and 5th cluster seem to be significant and should be retained for our analysis.
On the other hand, with the correlation values for the 2nd, 3rd and 4th clusters not being reflected Table 1 on the right, the team then deemed these categories as being insignificant to be represented by the 2 principal components, and proceeded to remove them from our table of analysis.

rank	PC1	PC2
1		0.811288
5		0.825791
6	0.855963	
7	0.823697	

Table 2

Delving deeper into the representations of the 2 principal components, the team was able to establish that Principal Component 1 is represented by items of relatively high calories like Chocolate, Peanut Butter and Butter which can be formalized as our comfort food.
On the other hand, Principal Component 2 can be represented by items of low calories like soy milk, fruits and vegetable, and items that are rich in carbohydrates, like biscuits, instant noodles and cereal beverages.

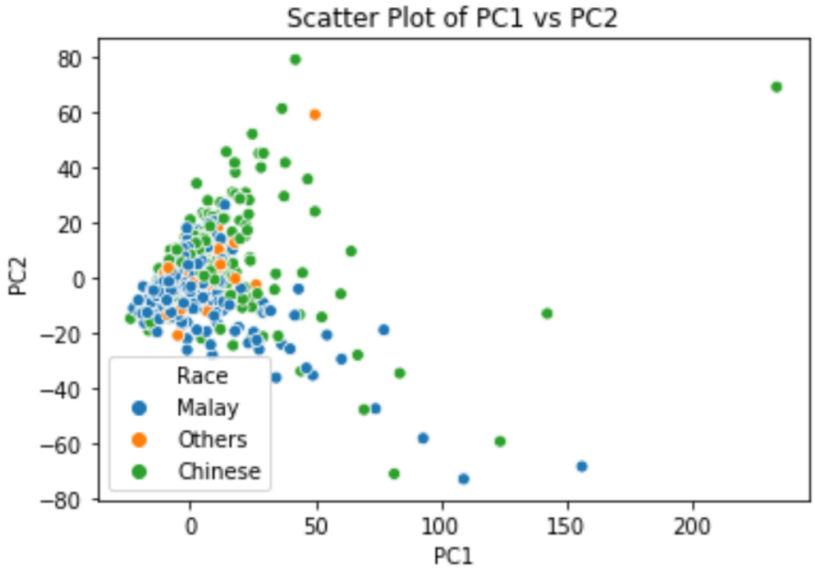


Figure 4

Mapping each unique customer in the Central area of Malaysia, represented by the 2 principal components onto a 2D plot, the team found that there seems to be a relationship between the particular racial community that the individual belongs to, and his/her purchasing habits.

From Figure 3, we observe that for customers in the Malay community, there seems to exist a negative linear relationship between the 2 principal components. This would suggest that customers in this community would tend to purchase items belonging to different principal components in a substitutional manner. Where a high purchase of items in a certain principal component would tend to result in a lower purchase of the other another.

On the other hand, customers in the Chinese community seems to showcase varying purchasing habits, with some choosing to purchase their items in a substitutional manner, while most others seem to be purchasing items belonging to different principal components in a complementary manner. Where a high purchase of items in a certain principal component would tend to also result in a higher purchase of the other another.

As such, having established a strong relationship between the racial community of the individual customer and their purchasing habits, the team will be able to utilise this information to improve the accuracy of our recommendation engine.

By segmentizing customers first based on their race and running our recommendation engine algorithm on the different racial groups, we will be able to better recommend items that are not only of interest to, but also in line with the purchasing habits of the individual customer.

4. K-Means Clustering with RFM

Using R, we first computed the annual average spend of each customer.

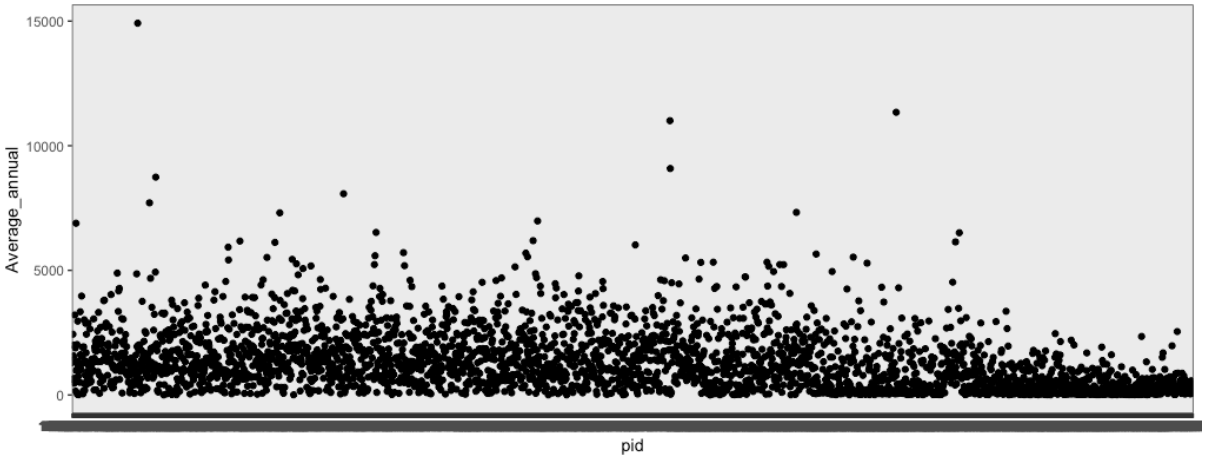


Figure 5

From the scatter plot, we identified some outliers and decided to remove them so as to not affect the clustering. The outliers are customers with annual average spend of more than 10000 Ringgit.

We then used the idea of RFM modelling and generated the data used for K-Means clustering.

Data summary:

1**) Recency**: The number of weeks since the customer's last transaction. The reference date is taken to be one week after the last transaction among all the transactions.

2) **Monetary:** The annual average spend of the customer.

3) **Frequency:** The annual average number of transactions of the customer.

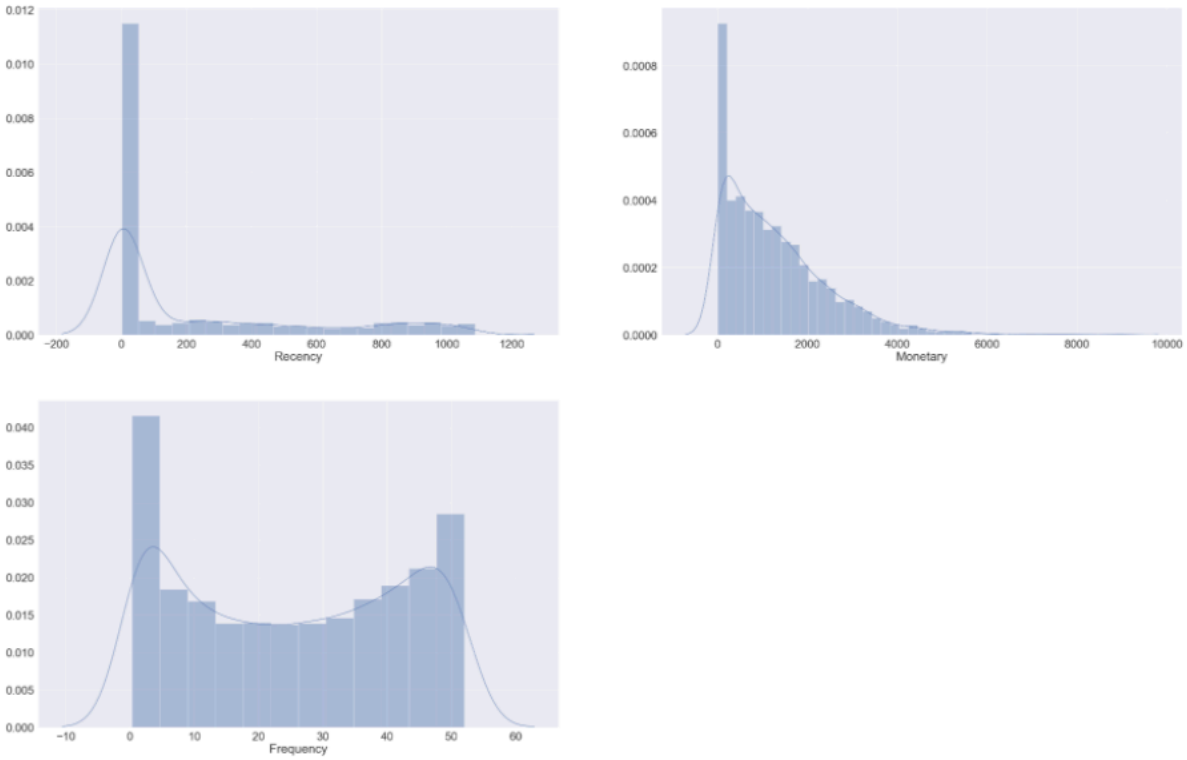


Figure 6

From the plot and the data summary, it is observed that most of the customers have spent at the supermarket recently. Most of the customers have average spend of less than 2000 Ringgit annually. As for annual average number of visits to the supermarket, it is more uniformly distributed, although there is a relatively large number of customers who visit the supermarket less frequently.

4.1 Choosing the appropriate number of clusters

4.1.1. Elbow method with Total WCSS vs Number of Clusters

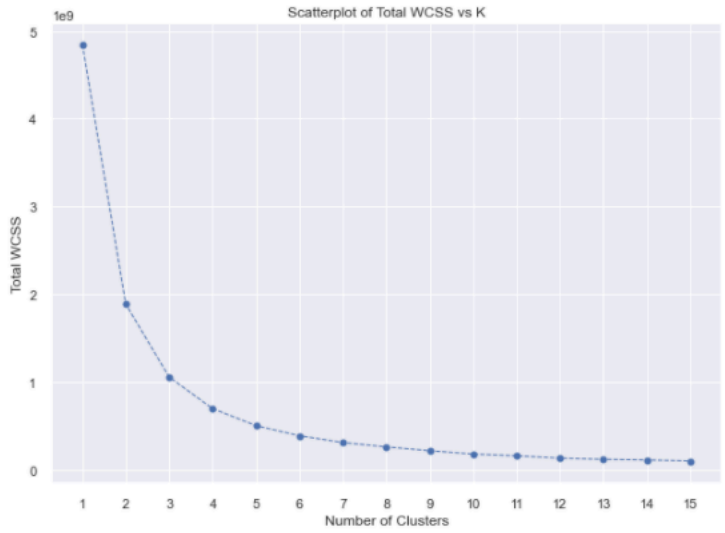


Figure 7

Using the kneed package to locate the elbow point of the curve, we find that the elbow point is at 4. We will then proceed to do K-means clustering based on the RFM score for the customers for 4 and 5 clusters.

4.1.2 K-means using RFM data - 4 clusters

We conduct K-means clustering using the RFM data with 4 clusters.

We plot a boxplot for each cluster and feature

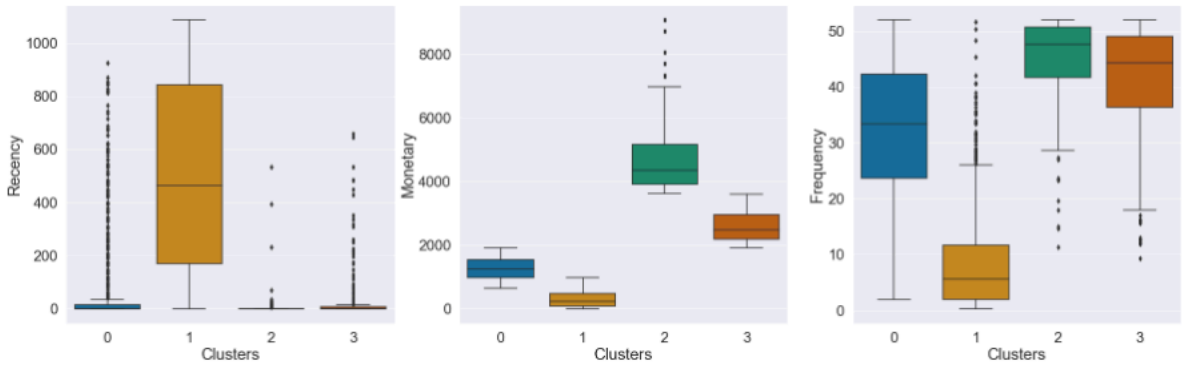
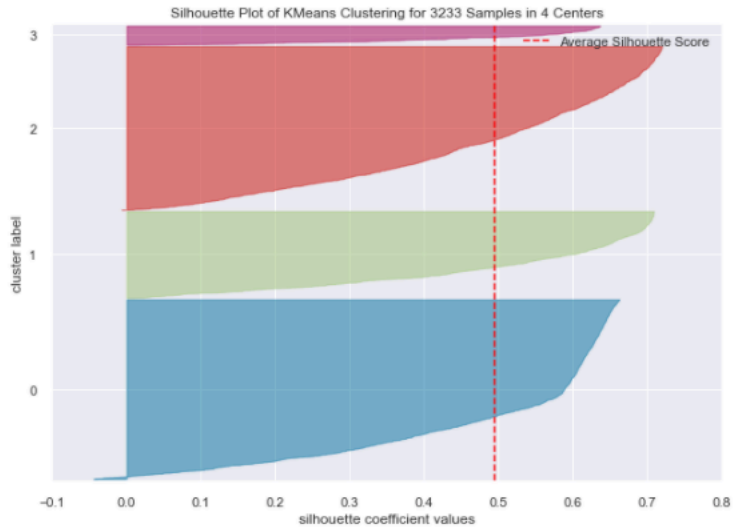


Figure 8




```
For n_clusters = 4 The average silhouette_score is: 0.4951807875617772
```

Figure 9

From figure 9, we can see that the silhouette score is 0.495. A larger silhouette score indicates that the object is well matched to its own cluster and poorly matched to neighbouring clusters.

4.1.3 K-means for Recency vs Monetary - 5 clusters

We conduct K-means clustering using the RFM data with 5 clusters.

We plot a boxplot for each cluster and feature

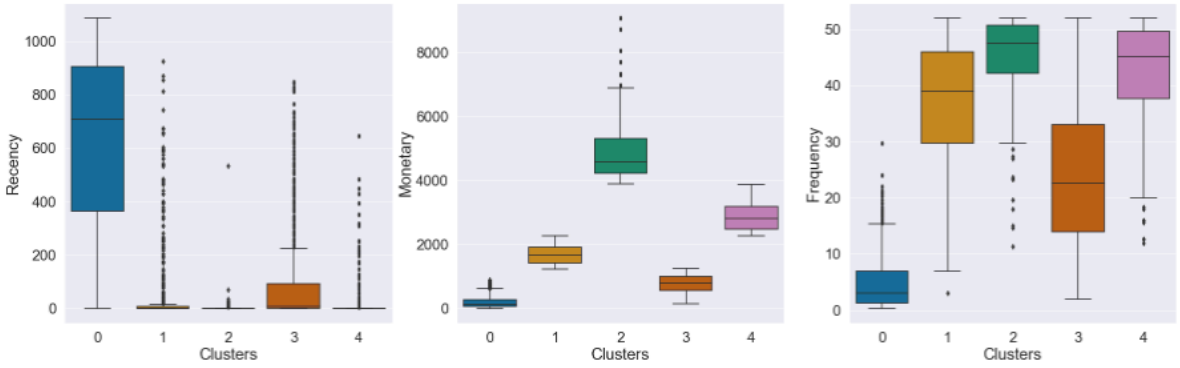


Figure 10

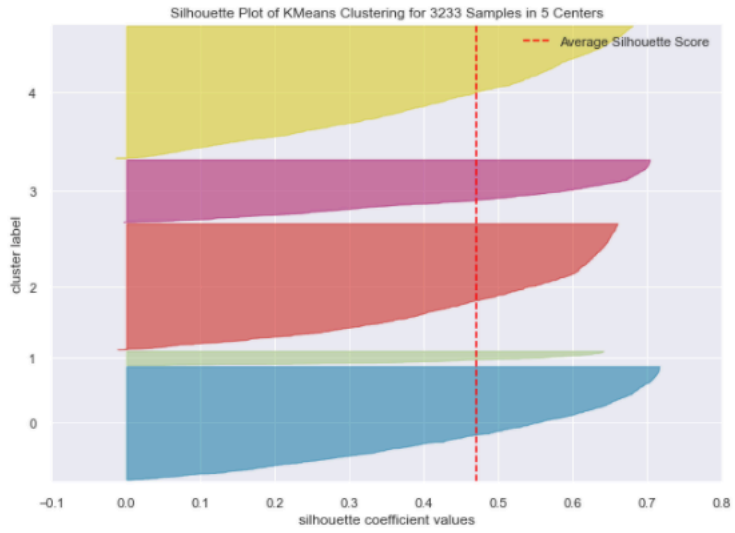


Figure 11

The dotted line in the plot represents the Average Silhouette Width for K = 5

With a higher silhouette score, it might be better to segment the customers into 4 clusters based on the RFM attributes.

4.2 Visualisation of the clustering

We now conduct a visualisation of the clustering for the customers with various RFM scores.

4.2.1 Recency VS Monetary

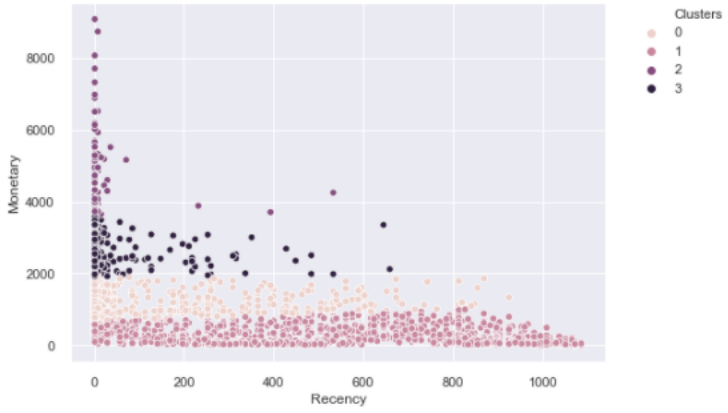


Figure 12

4.2.2 Frequency VS Monetary

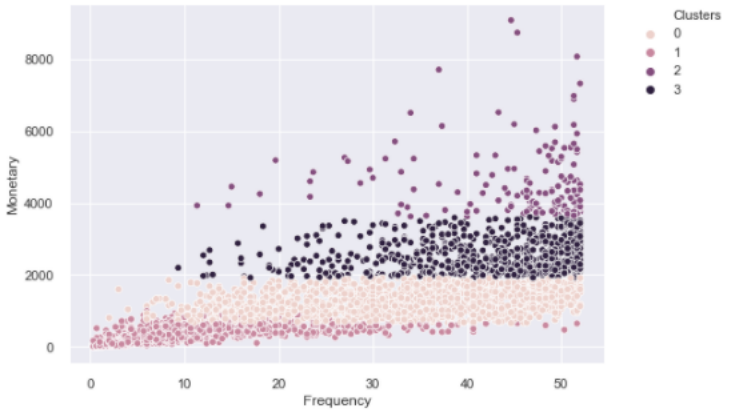


Figure 13

4.2.3 Recency VS Frequency

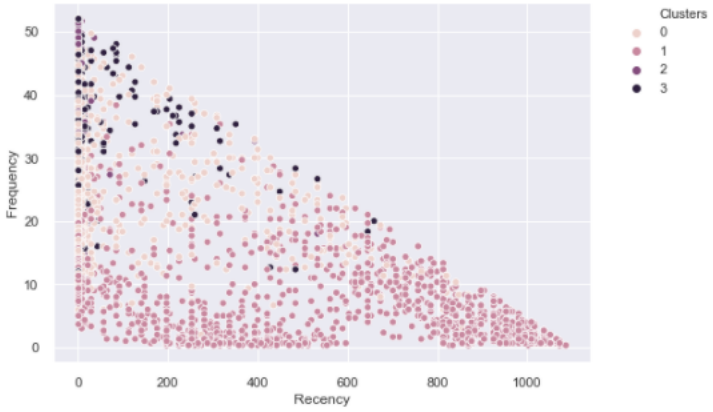


Figure 14

This plot does not show clear clustering of the customers.

4.3 Insights

From the scatter plots, we could derive the following customer profiles:

1) **Cluster 0:** This group of customers visit the supermarket least frequently and spend less (<1000) annually at the supermarket. Their last transactions took place over a very large duration of time.

2**) Cluster 1**: This group of customers visit the supermarket more frequently and spend more than cluster 0 customers. However, their average annual spend is still less than 2000 Ringgit. Their last transaction took place over a large duration of time as well.

3) **Cluster 2**: This group of customers visit the supermarket more frequently than cluster 1 and spend between 2000 to 4000 Ringgit on average per year. Most of them have visited the supermarket recently.

4) **Cluster 3**: This group of customers visit the supermarket most frequently and spend the most (>4000 Ringgit) on average per year. Their most recent transactions are also concentrated around 1, which is the last week in the dataset.

Therefore, we could see that K-Means clustering combined with RFM modelling can help us give suggestions on which segment outreach the supermarket should prioritise. Different campaigns should also be given to different customer profiles.

5. Recommendation Systems

A recommender system seeks to predict the rating given by a user for an item. However, in the dataset that we are provided, there is only information about the item categories (62 item categories). Our group feels that using item categories will not provide much meaningful and effective solutions (it is far more useful to recommend the clients on the actual item such as Carlsberg rather than the generic category of Beer). As a result, we attempted some data pre-processing to create arbitrary individual items. We thus define a unique item for that particular category as one which has a unique $\frac{Spend}{Volume}$ ratio for that particular category. With this approach, we generated a total of 36237 unique items.

We plan to create a web application that will be able to create our own recommendations based on these various methods:

5.1. Historical Purchases

Unlike many other recommendation systems like Netflix and Youtube, our group feels that the emphasis of our grocery recommendation system should be built upon the historical purchases of our customers. This is because of the high likelihood of repurchase of items due to various reasons such as habit, brand loyalty or tastes and preferences. Because of this, our recommendation engine places historical purchases over new recommendations at the top most priority.

5.2. Discounts and Promotions

As with most ecommerce platforms, items with discounts and promotions are usually listed at the top to catch the attention of consumers. Based on the product's stock inventory availability and whether or not the good is perishable, we can develop a marketing strategy around it to clear stock.

5.3. Historical Views

To improve further on the user's experience, we decide to include the historical views of the customers. Not only does it provides convenience, there is also a very high chance that they purchase the products given that they have viewed it before.

5.4. Trending Purchases

With the increasing awareness among consumers about the benefits of health ingredients, the market for them is growing, as consumers prefer balanced diets and food that not only satiates their hunger but is also nutritional. Our group hopes to capitalise on this trend or even create new trends to recommend healthier alternatives for the consumers. We also

understand that trends may be a double edged sword as seen from the lockdown period where many consumers fall into the herd mentality and mass buy essential goods like instant noodles and rice.

5.5. Collaborative Filtering

There are many different ways to build a collaborative filtering recommendation system but our group decided on using factor models:

$$R^{m\times n} \approx \hat{R}^{m\times n} = U^{m\times p}\left(V^{n\times p}\right)^{\mathsf{T}}$$

For m users and n items, we can construct a ratings matrix of size $m\times n$. This rating matrix $R^{m\times n}$ can then be approximated by our predicted rating matrix $R^{m\times n}$ of which can be decomposed into 2 smaller matrices, one for the users $U^{m\times p}$ and the other for the items $V^{n\times p}$
$n \times p$. We formulated this into a minimisation problem where our cost function is

$$Loss = \frac{1}{|S|}\sum_{(i,j)\in S}\left(r_{i,j} - u_i v_j^{\mathsf{T}}\right)^2 + \frac{\lambda}{m}\|U\|^2 + \frac{\lambda}{n}\|V\|^2$$

where S represents the set of (i,j) entries which user i has rated item j. Due to the large number of parameters, we also imposed regularisation penalties so that our model does not overfit the data. To speed up the computation for real time deployment, we chose to use alternative least squares method for our optimisation process. The evaluation metric used here is Root Mean Square Error (RMSE) which we seeks to optimise.

The advantage of using factor models is that we can obtain 2 smaller matrices, one for the users $U^{m\times p}$ and the other for the items $V^{n\times p}$ which provide us the latent variables associating with the users and items. These latent variables can provide us key insights to understanding consumer behaviours. A possible interpretation is as follows:

$$\hat{r}_{i,j} = \begin{bmatrix} \text{sugar level} \\ \text{calories} \\ \vdots \\ \text{cholesterol} \end{bmatrix}_u \cdot \begin{bmatrix} \text{sugar level} \\ \text{calories} \\ \vdots \\ \text{cholesterol} \end{bmatrix}_v$$

With these information, we can understand their dietary preferences and perhaps influence them to adopt a healthier living habits.

As the dataset provided to us lack information about the actual ratings, our group decided to construct pseudoratings in place of the actual ratings. After some exploration, we agreed to generate based on the idea of RFM modelling which we have learnt previously. We made some modifications to the original RFM to better suit our needs. Here, we encode the dates into integers from 0 (2017-06-25) to 155 (2020-06-14) for better representation. For a user i, and product j,

- $Recency_{i,j} = \frac{week}{155}$, where $week$ represents the most recent week of user i purchases item j.
- $Frequency_{i,j} = \frac{(activePeriod - diffWeekEMA)}{activePeriod}$, where $activePeriod$= date of last purchase−date of first purchase and $diffWeekEMA$ represents the Exponential Moving Average (EMA) of the difference in weeks of the k-th purchase and the $(k+1)$-th purchase (how many weeks are between each purchase of item j by user i). We assign the decay in terms of center of mass to be equals to 1.
- $Monetary_{i,j} = \frac{\text{spendEMA} - \text{minSpend}}{\text{maxSpend} - \text{minSpend}}$, where $minSpend$ and $maxSpend$ represents the minimum and maximum spend user i has spent for product j, and $spendEMA$ represents the Exponential Moving Average (EMA) of the amount spend of the k-th purchase and the $(k+1)$-th purchase. The decay in terms of center of mass is also assigned to be 1.

A more recent client will have a higher $week$ which equates to $Recency$ close to 1. The choice of using EMA is so that we place greater emphasis on the more recent customer behaviours. For example, a customer who is frequent in recent times will have lower $diffWeekEMA$ than a customer who is less frequent in recent times which results in the former having a higher $Frequency$ value than the latter. Similarly, a customer who spends more in recent times will have higher $spendEMA$ than a customer who spends less in recent times, resulting in higher $Monetary$ for the former. By doing this, we are able to reduce the impact caused by confounders like:

- Unfair comparison between customers who have been recording their purchases since 3 years ago and customers who have just started their recording process. This affects the $Frequency$ as naturally the latter will have a lower value if we were to just count by the flat number of weeks.
- Unfair comparison between high income earners and low income earners in which the high income earners have the ability to spend more than the low income earners resulting in high $Monetary$ value for them for most of the time. To curb this, we compare the weighted spending to the maximum spending by user i for product j and this will enable us to know if the consumer is spending within his/her capabilities. This also solves the issue of some products are inherently more expensive than others.

Finally, we combined them by taking the product. Since each of them are bounded between 0 and 1, the product naturally will also be bounded between 0 and 1. Finally we multiply the result by 10 for nicer inference.

5.5.1. User

Applying the nearest neighbour algorithm on our user matrix U, we find the top k users that have similar purchase patterns as the user. We then based our product recommendations on the users' recent purchases.

5.5.2. Item

Similarly, we can also do the same for the products. Using the nearest neighbour algorithm on the item matrix V, we find the top k products that have similar purchase patterns as the products bought by the user.

5.5.3. General

Apart from finding the similar users or items for user i, we can also retrieve the pseudoratings by taking the matrix multiplication of $u_i^{1\times p}$ and $(V^{n\times p})^T$ We then can take the top k highest pseudoratings among the n products and recommend to the customer.

5.5.4. Results

We split the data into about 75% training set and 25% testing set. We benchmarked our model against the testing set with 3 naive models namely, using the global mean to predict every item's ratings, using the panel id's mean to predict for the corresponding panel id's items' ratings and using the item's mean to predict for the corresponding item's ratings. Our results are as follows:

- Global Mean $=1.59$
- Panel Mean $=1.56$
- Product Mean $=1.55$ (2.6% improvement from global mean)

It is easy to see that given slightly more information (either knowing the panel/product mean) we can achieve lower RMSE which equates to better predictive performance. Using our matrix factorisation approach with regularisation $\lambda=1$ and number of latent factors $p=30$, we attained a RMSE of 1.52 which is an improvement from the best naive model (product mean) by 1.9%. It has been claimed that even as small an improvement as 1% RMSE results in a significant difference in the ranking of the "top-10" most recommended items for a user (Source: https://en.wikipedia.org/wiki/Netflix_Prize).

5.5.5. Others

In case the latent variables are unable to capture the features of the users and products, we also offer an additional strategy to complement with it - Content-based filtering:

- Upsell a luxurious or healthier **substitutes** to customers who fall into the higher income bracket or customers who does not have healthy BMIs.
- Cross-sell **complementary** goods such as pasta with pasta sauces.

6. Conclusion

1. Our team was able to reduce the missing data from 49,539 missing data to only 3859 missing data. This means that there will be less noise within the dataset and our team will have better results overall.

2. By utilising hierarchal clustering and principle component analysis, our team was able to identify demographical clusters (race) and are able to make good recommendations for these various groups.

3. We have identified a clear group of customers, using K-means clustering, that visit the supermarket the most frequently and we can create and curate specialised promotions to these various groups to increase sales.

4. We have created a deployable web application that uses various recommendation engine parameters. We are confident that the results of our recommendations engine will be accurate and we will need to deploy the engine to see the overall impact that the system can give.

7. Appendix

All codes can be found at the `code` folder in the zip file. Note: For `src > recommendations_system.ipynb`, the notebook uses an autodifferentiation library called JAX which is only available in *nix systems (Google Colaboratory has this package installed).