Deliverable 1: Final Year Dissertation

Sahil Pattni

BSc. Computer Science (Honours)

Supervisor: Neamat El Gayar

December 5, 2020

Declaration

I, Sahil Manojkumar Pattni, confirm that this work submitted for assessment is my own and is expressed in my own words. Any uses made within it of the works of other authors in any form (e.g., ideas, equations, figures, text, tables, programs) are properly acknowledged at any point of their use. A list of the references employed is included.

Signed: Sahil Manojkumar Pattni

Date: December 5, 2020

Abstract

In this digital age, data is being generated at an exponential rate, with data analytics being used by corporations and small businesses alike to produce insights, reduce costs, optimize operations and increase profits. Generating association rules allow us to find non-intuitive associations that bring new insights to the management, allowing them to leverage this information to maximize their profits. A prime example would be the *'Beers and Diapers'* urban legend, where a company looked at their point-of-sale data and found a strong association between beers and diapers being co-purchased, which seems rather unintuitive.

In this study, we will extract a minimum spanning tree (MST) from a data set using machine learning techniques. We will then use this minimum spanning tree to segment products together, and produce association rules and extract the most interesting ones. The resulting ruleset will be compared to rules generated by the established Apriori Algorithm [1], which is the foundation of association rule mining. Additionally, we will study how the grouping of the MST compares to clustering algorithms, and how the structure of our MST would change before and after promotional events, leading to insights that may help the management in such firms make better informed decisions about the type of promotions they would like to run.

Contents

1 Introduction

1.1 Context

For business which deal with the sale of a heterogeneous physical assets - such as groceries, hypermarkets and select retail outlets - operations such as inventory management and product placement play an instrumental role in determining the business' financial success. These involve asking questions such as:

- Which products should be placed at the entrance of the store? Which should be placed closer to the exit?
- Which products will benefit the most by being placed at eye-level?
- Which products should be placed next to each other to maximize the purchase volume?

One way to find optimal solutions for such questions is to employ the use of Association Rule Mining (also known as Market Basket Analysis). This set of techniques assess frequent itemsets (e.g. from sales data) and generate association rules between products. Several algorithms and techniques exist for association rule mining, such as the Apriori Algorithm and FP-Growth [2]. One problem with these algorithms is that they tend to generate an enormous amount of rules, of which many of the rules themselves are large This makes it grossly inconvenient for the end-user to retrieve any actionable information from the results.

What we will do is construct a network of products - where each vertice represents a product or product category, and an edge between two vertices represents the simultaneous occurrence of the the two products. The result is a graph (i.e. a network), whose architecture is visually informative of the products and their associations. The insights that can be gained from the MST can help the management of the mentioned businesses to maximize their profit.

1.2 Aims

The aim of this study is to study the effectiveness of a minimum spanning tree as a market basket analysis tool - from product clustering to association rule mining. Additionally, we will determine how the architecture of the MST will change during and after large events such as promotions. The model will be tested on a relatively large dataset of sales data.

1.3 Objectives

The research objectives for this project have been laid out below, in the order that they will be carried out.

1. Acquire a suitable dataset upon which the MST can be constructed.

2. Explore and evaluate MST extraction algorithms.

   (While Prim's [3][4] and Kruskal's [5] algorithms are the most commonly used methods to extract the MST from a graph, research has been conducted on more efficient ways to extract the MST using machine learning techniques such as Artificial Neural Networks and K-Nearest-Neighbors, and these will be discussed in the literature review (see Section 2.2).

3. Generate an undirected graph $G(E, V)$ where the edges E are the correlation values between the product vertices.

4. Extract a minimum spanning tree from this graph using the technique determined in step 2.

5. Analyze this MST and use its architecture to determine product clusters and generate association rules.

6. Generate association rules using the Apriori and FP-Growth algorithm and compare them in both their time complexity and their *'interestingness'* (i.e. the unintuitive rules they generate).

7. Validate the product grouping present in the MST against a clustering algorithm (e.g. such as the K-Means algorithm).

2 Background

2.1 Core Concepts

2.1.1 Graphs

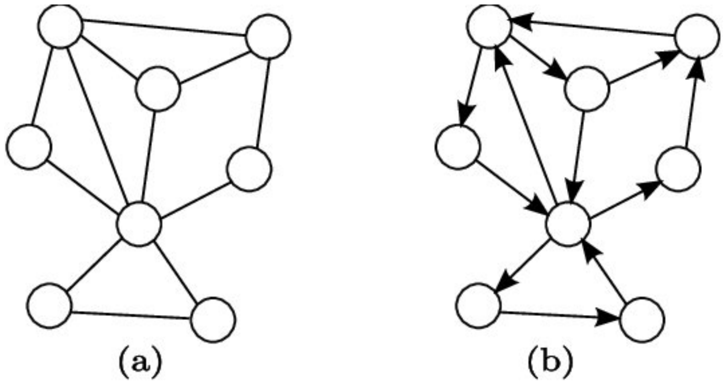

Figure 1: Undirected and Directed Graphs

In discrete mathematics and more specifically - graph theory, a graph is a data structure that contains a set of nodes (i.e. vertices) connected by lines (i.e. edges). These edges may be directed - such as in Figure 1:(a), or undirected - such as in Figure 1:(b). A graph G with a set of vertices V and a set of edges E can be represented via the notation $G = (V, E)$. For the purposes of our research, we will be focusing on undirected graphs, where the weight between two vertices is the same in both directions.

2.1.2 Minimum Spanning Trees

Given an undirected $G = (V, E)$, a *spanning tree* can be described as a subgraph that is a tree which includes all the vertices V of G with the minimum number of edges required. A *minimum spanning tree* (MST) is the spanning tree with the smallest sum of edge weights. This means that if the graph has n vertices, each spanning tree - including the minimum spanning tree - will have $n - 1$ edges.

2.1.3 Market Basket Analysis and Apriori Rule

Market Basket Analysis (MBA), also known as Affinity Analysis, is a data mining and analysis technique that identifies co-occurrence patterns between products purchased together, and produces association rules for these products as such: $\{A\} \rightarrow \{B\}$ which implies a strong relationship between the purchase of product A and product B. In the case of our *beers and pampers* example, the association rule could be represented as $\{Beers\} \rightarrow \{Pampers\}$. The contents of a purchase basket (i.e. the contents of a customer's basket when they check out) is called a *itemset*, which - as the name suggests - is a set of all the items in the basket. For example, if a customer had bought detergent, bread and soda, the itemset would be $\{bread,\ detergent,\ soda\}$. Association rules are generated by looking at different combinations of the itemset (e.g. $\{bread,\ soda\} \rightarrow \{detergent\}$ and $\{soda\} \rightarrow \{bread\}$). The equation [6] to calculate the number of rules for an itemset of length d is as follows :

$$number\ of\ rules = \sum_{k=1}^{d-1} \left(\binom{d}{k} \times \sum_{j=1}^{d-k} \binom{d-k}{j} \right) \tag{1}$$

Analyzing the equation, it becomes apparent that the problem with association rule generation from itemsets is that the number of rules produced grows exponentially with the size of the itemset, as we can see in Figure 2, using Equation 1.

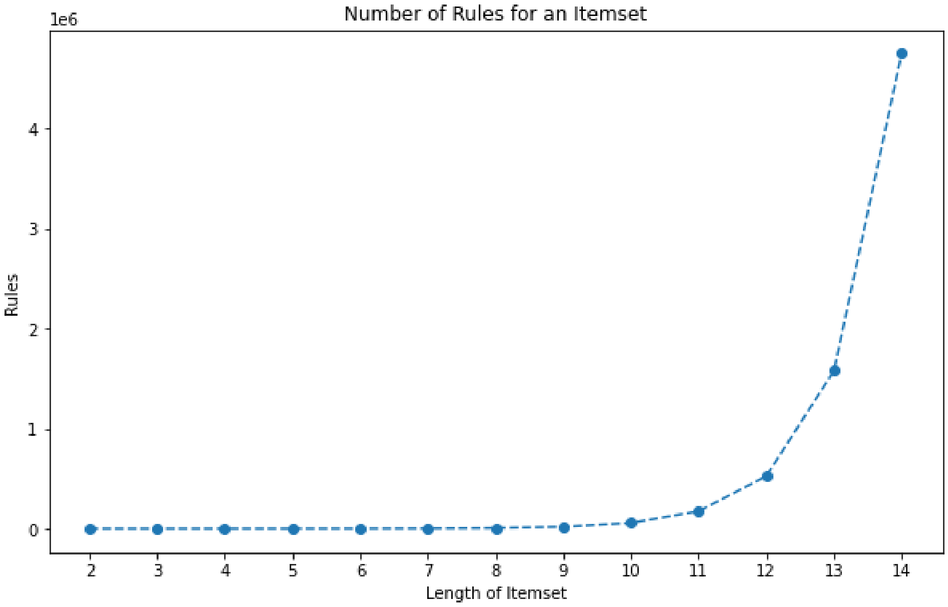

Figure 2: Number of Association Rules for an Itemset

Clearly, checking all itemsets in a database would be computationally expensive, and so we only look at *frequent itemsets* - itemsets that have a support value above a minimum threshold $min_{support}$. Even so, checking each itemset's support score via brute force is unacceptably time consuming, therefore we can optimize the search for frequent itemsets using the Apriori Principle [7] which states that *all subsets of a frequent itemset must be frequent.* To understand how the Apriori Principle can be applied here, we first look at three key metrics:

Support

The support of a set of products is the fraction of transactions in which these set of products are present. For example, for a list of transactions T, for the product X where $T(X)$ denotes that the set of transactions in which X was present:

$$\mathit{support}(\{A\}) = \frac{T(A)}{T}$$

and similarly,

$$\mathit{support}(\{A\} \rightarrow \{B\}) = \frac{T(A, B)}{T}$$

denotes the support for the rule $\{A\} \rightarrow \{B\}$. We can prune rules with a low support as they indicate a rule does not occur enough to draw any reasonable conclusions from.

Confidence

Confidence is the measure of how likely a product will be in a basket given that another product is in it. That is to say, the confidence of a rule $\{A\} \rightarrow \{B\}$ is the conditional probability that $\{B\}$ occurs in the basket given that $\{A\}$ is present. The confidence of a rule can be denoted as:

$$\mathit{confidence}(\{A\} \rightarrow \{B\}) = \frac{T(A, B)}{T(A)} \equiv \frac{\mathit{support}(\{A\} \rightarrow \{B\})}{\mathit{support}(\{A\})}$$

Lift

The lift of a rule $\{A\} \rightarrow \{B\}$ is the rise (i.e. **lift**) that $\{A\}$ gives to our $\mathit{confidence}(\{A\} \rightarrow \{B\})$.

$$\mathit{lift}(\{A\} \rightarrow \{B\}) = \frac{confidence(\{A\} \rightarrow \{B\})}{support(\{B\})}$$

To make this easier to understand, imagine that $\mathit{confidence}(\{A\} \rightarrow \{B\}) = 0.5$, and $\mathit{support}(\{B\}) = 0.4$. This means that the presence of $\{A\}$ increases the probability of $\{B\}$ being in the same basket by 25% ($\frac{0.5}{0.4} = 1.25$), therefore providing us with a lift value of 1.25. A lift value below 1 would indicate that the occurrence of $\{A\}$ in a basket decreases the likelihood of $\{B\}$ occurring in the same basket (i.e. a low product association).

As we established above, the Apriori Principle states that all subsets of a frequent itemset must be frequent. The Apriori Principle is a result of the *anti-monotone property of support* [8], which means $\{A, B, C\}$, $support(\{A, B\}) \geq support(\{A, B, C\})$. We can use this to prune the frequent itemsets much more efficiently, because if $support(\{A, B\}) < min_{support}$, then any itemset containing the set $\{A, B\}$ will also fall below $min_{support}$. Once the frequent itemsets have been pruned, association rules can be generated from the remaining itemsets. The resulting association rules can be even further pruned by removing those that fall below a confidence threshold $min_{confidence}$. Finally, the remaining rules can be ranked according to their lift to find the rules with the highest associations.

2.2 Related Work

2.2.1 Papers on Association Rules

R. Aggarwal et al. [7] proposed a novel algorithm to generate all statistically significant association rules between items in a database, laying the foundations for association rule mining. Given a set of items $I = I_1, I_2, I_3, ...I_m$, the authors define an association rule to be of the form $X \rightarrow I_j$ where X is a set of items such that $X \in I, I_j \notin X$. The hypothetical database stated was a list of transactions, T, where each transaction t was a binary vector of length m, representing a basket purchase, where $t[k] = 1$ if I_k been purchased in that basket. The authors stated that their methodology for association rule mining could be split into two steps: the generation of candidate itemsets, and the generation of statistically significant association rules from these itemsets.

To address the first subproblem, the authors provided the pseudo-code for their candidate itemset generation, where all itemsets possible were generated from tuples (samples) from the database, and those itemsets whose support score[1] is above the minimum support threshold are considered candidates (called *large itemsets*). Since a brute-force check would be sub-optimal (the authors note this could take up to 2^m passes of the database), the authors devised a methodology to check for candidate itemsets where on the k^{th} pass of the database, they would only check itemsets of length k, to see if they satisfied the support constraint. On the $(k+1)^{th}$ pass, they need only check those itemsets that are 1-extensions (i.e. itemsets extended by exactly one item) of the *large itemsets* found in the previous pass. This is because of what would

[1] see: Section 2.1.3

later be known as the *Apriori Principle*, where if an itemset Y is *large*, then all subsets of Y must also be *large*. Therefore, if they found the itemset $\{A, B\}$ was *small* (i.e. did not satisfy the support constraint), then sets containing $\{A, B\}$ (e.g. $\{A, B, C\}, \{A, B, D\}, \{A, B, C, D\}$) would also be small, and need not be checked. This means, however, that if an itemset I is *large*, then another pass over the dataset would be required to check the support of the subsets of I. To avoid this, the authors devised a measure to calculate the expected support, $\bar{s}$, of an itemset, and use this to measure the support of itemsets $I = (X + I_j)$ not only where I is expected to be large, but also where $X + I_j$ is expected to be small but X is expected to be large, further helping them prune the number of itemsets to check. The authors proceed to define a method that allows the algorithm to be more memory efficient[2]. The authors also defined method to further prune itemsets from the search, namely *remaining tuples optimization* and *pruning function optimization*.

To address the second subprobem, the authors proposed the following methodology: for each large itemset $Y = I_1, I_2, ...I_k, k \geq 2$ from the set of non-pruned large itemsets, generate a set of association rules of form $X \rightarrow I_j$ such that the consequent is I_j and the antecedent (i.e. the precedent set in the rule) is a subset X of Y such that X is of length $k - 1$ and $I_j \notin X$. Therefore, each large itemset will produce k rules. From the generated rules, the authors discarded those rules whose confidence scores[3] fell below the minimum confidence threshold c.

The authors tested their methodology on a sales dataset with $46,783$ transactions, with 63 *items* (in this case, the department from which the customer bought an item). They used a configuration of a minimum support threshold of 1% and a minimum confidence threshold of 50%. The rules produced seemed to follow with general intuition, such as:

$\{$Auto Accessories, Tires$\} \rightarrow \{$Automotive Services$\}$

Furthermore, the authors assessed the accuracy of their expectation method, by measuring the ratio of correctly estimated itemsets for both small and large, against various values for the minimum support threshold, and visualizing the result. To isolate the effect of their expectation method, they disabled their pruning optimization functions. They were able to conclude that their estimation accuracy was satisfactory, as their accuracy was above 98% for support thresholds except the first, where it was 96%. The authors also tested the effectiveness of their pruning optimization functions, namely the *remaining tuples* and the *pruning function* opti-

[2] This may no longer be required due to the advances in computational power in the the 27 years since this paper was written.

[3] see: Section 2.1.3

mization functions, against multiple minimum support threshold values. They were able to conclude that their pruning efficiency increased as the support threshold increased.

Critical Analysis

(To be completed)

2.2.2 Papers on Networks

H. K. Kim et al. [9] proposed a study of product networks to complement market basket analysis. Their methodology involved constructing two products: a market basket network (MBN) - which analyzes the relationships between products purchased together (i.e. in the same basket) - and a co-purchased product network (CPN) - which is a bipartite network associating customers with products they have purchased over time. The authors' reasoning for doing so is because they argue that market basket analysis does not analyze the relationships between products purchased over a period of time by a given customer. Using a dataset of 68,573 transactions from the sales data of a department store in South Korea, the authors constructed both networks, MBN1 and CPN1 respectively. Additionally, since both the resulting networks were too dense to analyze visually, they removed approximately 60% of the links from MBN1 (with values lower than 0.009) and named it MBN2, and removed all the links from CPN1 whose values were lower than the average, and called this new network CPN2. Furthermore, the authors created a third network - CPN3, by removing nodes from CPN2 to adjust its density to match that of MBN1. Analyzing them both, the authors noted that the CPN network was 20 times denser than its MBN counterpart.

M. A. Valle et al. [10] proposed a novel methodology to study the structure and behavior of consumer market baskets from the topology of a minimum spanning tree which represented the interdependencies between products, and use this information to complement the association rule generation process. The input to their proposed methodology was a correlation matrix between the set of all one-hot encoded purchase vectors such that each vector denoted the presence or non-presence of each product from the dataset in that vector. The dataset used for the MST construction was a list of $1,046,804$ transactions containing a set of $3,240$ unique products from a large supermarket chain branch in Santiago, Chile. When building this correlation matrix, the authors opted to use the Pearson's Coefficient (which is equivalent to the coefficient ϕ for binary data such as theirs) over the traditionally used Jaccard distance to compute the similarity between the binary product vectors, as the former provides both a positive and negative association between products. Additionally, they used the distance function $d_{ij} = \sqrt{2(1 - \phi_{ij}}$ to transform the correlation matrix into a distance measurement (i.e. the weight of the edges). The authors constructed a MST for 220 product subcategories, and noted that there was a significant level of grouping between product sub-categories that belonged to the same parent category. To remove edges from the MST that were not statistically significant, the authors used the mutual information [11] measure $\sum\limits_{x,y} log_2 \frac{r(x,y)}{p(x)q(y)}$ between product subcategories p and q, and were able to prune 14 edges, all of which were connected to a terminal node, therefore effectively pruning 14 vertices from the MST too. To identify the most influential regions of the MST, the authors defined an influence zone of distances that were in the 10^{th} percentile. To generate meaningful association rules, for each MST product i, the authors ran a search for the set of all association rules R_i such that $P_i \rightarrow P_j(i \neq q)$. Then from the resulting set of rules, they searched for rules that obeyed $P_i \rightarrow P_m$ where m a product node connected to the product i in the minimum spanning tree. For both resulting sets of rules for each product, the mean of their lift scores were observed, and the authors determined that the rules that were reinforced by the MST had a higher mean, and that a majority of these rules had a lift score above the 90^{th} percentile.

To identify the clusters each of the products should be identified under, the authors constructed a hierarchical tree using the average linkage clustering method, and by using an unspecified cut distance, they were able to produce 17 taxonomic groups (i.e. clusters). Cross-referencing their results with the actual parent categories of the products, they were able to conclude that the MST did indeed categorize the product sub-categories into clusters with a reasonable degree of accuracy. The authors then compared their MST to another methodology, namely the structured association map (SAM) [12], using the Jaccard distance as a measure of similarity, and were able to generate interesting 2x2 rules (i.e. $\{A, B\} \rightarrow \{C, D\}$), all with lift scores above 1.0, with one rule even having a lift score of 106.46. They concluded that while both approaches provided different information, they both visually identify the strongest relationships between the products, and provide useful information to reduce the search space for association rules.

Critical Analysis

The authors' approach seems to be novel, thorough and well structured. Their methodology successfully employed the use of minimum spanning trees to complement the association rule generation process with sound results. One caveat of their approach is that they only used the

MST to generate 1x1 rules (i.e. $\{A\} \rightarrow \{B\}$). Using the distance score in conjunction with the importance function they defined (i.e. $\sum\limits_{k \in K_u} \frac{1}{w_{uk}}$), they could have defined a system to produce $n \times n$ rules, then rank them using their respective lifts. Additionally, while the authors did cross reference their clustering results against the real parent categories of the products, they did not compare their results to that of a clustering algorithm (e.g. K-Means), which would have given a reasonable benchmark to compare the results of the MST clustering to. While 1x1 rules are easily understandable and tend to have high lift values when extracted from the MST, $n \times n$ rules would provide a layer of insight as to how a range of products (perhaps a cluster) related to another.

3 Requirements & Research

3.1 Requirements Analysis

This section will identify both the functional and non-functional requirements required for the implementation of the model proposed in this document. By stating the specific requirements of the model, it serves as a high-level view of how the model should function.

Table 1: Functional Requirements

ID	Category	Description
FR1	Dataset	The dataset must be a dataset of transactions.
FR2	Dataset	The dataset must have a unique identifier for each transaction.
FR4	Model	A list of binary purchase vectors must be extracted from the dataset.
FR5	Model	The model must return a storable data type representation of a MST.

Table 2: Non-Functional Requirements

ID	Category	Description
NFR1	Dataset	The dataset must contain at least 500,000 unique transactions.
NFR2	Hardware	The data inputs and the MST construction will run on a computer with at least 8 GB of ram, and a quad-core CPU with clock speed aabove 2.5GHz.

3.2 Research Methodology

The primary objective of this dissertation is to analyze the viability of a framework based on the minimum spanning tree as a 'one-in-all' tool for market basket analysis, from association rule generation to clustering. This section will describe how we plan to achieve these goals:
A candidate dataset has been identified: a dataset of $27,000,000$ transactions occurring across multiple branches of a Brazilian gas station store [13]. Each row in the dataset contains the purchase information for a particular product for a given transaction. The dataset has been checked for errors and inconsistencies, and the products have been aggregated (e.g. all the different brands of lubricant have been renamed to 'lubricant'). Additionally, since the trans-

action id is an unreliable identifier for a market basket[4], we have assigned each row a unique alphanumeric code consisting of the purchase's transaction number, city and date, which will act as a robust unique identifier for each basket. Further work is required on the dataset, such as the removal of redundant or unnecessary columns.

Once the dataset is prepared, we will need to extract a binary purchase matrix from the dataset. This will prove to be quite computationally expensive, as there are approximately $5,200,000$ unique baskets in our dataset. Even when only considering $473,641$ baskets with 40 unique products, the resulting matrix (see Fig. 3) took over 6 hours to construct. We will be looking into more efficient ways to generate out input data.

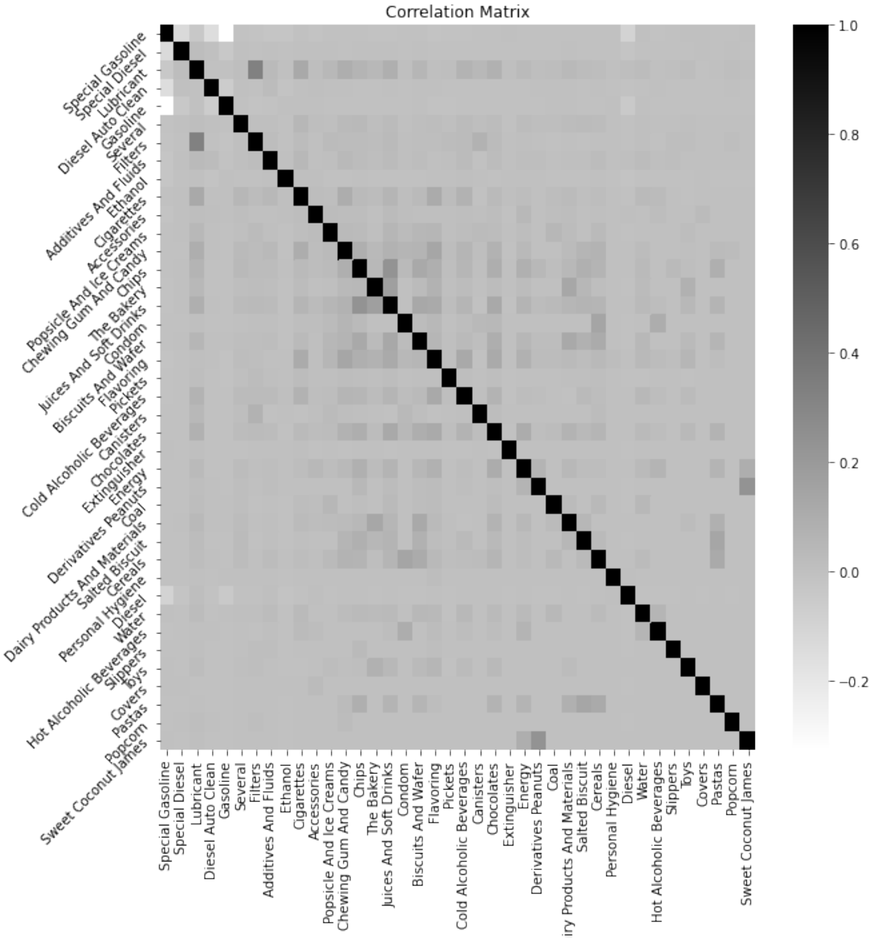

Figure 3: Correlation Matrix for 473,641 baskets

Once the correlation matrix has been generated and a distance function applied, a graph

[4] transaction numbers reset once they have reached an upper limit, transaction ids are not synced across stores so the same transaction id could refer to multiple transactions across different stores.

will have to be constructed. Given the size of our dataset, extracting a MST from our graph using traditional algorithms such as Prim's and Kruskal's may be ineffective, and we may have to look into more recent approaches [CITE NN / RANDOM FOREST MST EXTRACTION PAPERS]

References

[1] R. Agrawal and R. Srikant, “Fast algorithms for mining association rules,” *Proceedings of the 20th International Conference on Very Large Databases*, pp. 487–489, 1994.

[2] J. Han, J. Pei, and Y. Yin, “Mining frequent patterns without candidate generation,” 2, vol. 29, New York, NY, USA: Association for Computing Machinery, May 2000, pp. 1–12. DOI: 10.1145/335191.335372. [Online]. Available: https://doi.org/10.1145/335191.335372.

[3] R. C. Prim, “Shortest connection networks and some generalizations,” *Bell System Technical Journal*, vol. 6, pp. 1389–1401, 6 1957. DOI: 10.1002/j.1538-7305.1957.tb01515.x.

[4] V. Jarník, “O jistém problému minimálním,” *Práce Moravské Přírodovědecké Společnosti*, pp. 57–63, 1930.

[5] J. B. K. Jr., “On the shortest spanning subtree of a graph and the traveling salesman problem,” *Proceedings of the American Mathematical Society*, vol. 7, pp. 48–50, 1956. DOI: 10.1090/S0002-9939-1956-0078686-7.

[6] J. C. Pennsylvania. (2020). “Generating association rules,” Junita College Pennsylvania, [Online]. Available: http://faculty.juniata.edu/rhodes/ml/assocRules.html.

[7] R. Agrawal, T. Imieliński, and A. Swami, “Mining association rules between sets of items in large databases,” in *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD ’93, Washington, D.C., USA: Association for Computing Machinery, 1993, pp. 207–216, ISBN: 0897915925. DOI: 10.1145/170035.170072. [Online]. Available: https://doi.org/10.1145/170035.170072.

[8] P.-N. Tan, M. Steinbach, A. Karpatne, and V. Kumar, “Definition 6.2,” in *Introduction to Data Mining*, 2nd ed. 2019, ch. 6, p. 334, ISBN: 9780134080284.

[9] H. K. Kim, J. Kim, and Q. Y. Chen, “A product network analysis for extending the market basket analysis,” *Expert Systems with Applications*, vol. 39, pp. 7403–7410, 8 2012. DOI: 10.1016/j.eswa.2012.01.066.

[10] M. A. Valle, G. A. Ruz, and R. Morrśs, “Market basket analysis: Complementing association rules with minimum spanning trees,” *Expert Systems with Applications*, vol. 97, May 2018. DOI: 10.1016/j.eswa.2017.12.028.

[11] T. M. Cover and J. A. Thomas, *Elements of information theory.* 2006.

[12] J. W. Kim, “Construction and evaluation of structured association map for visual exploration of association rules,” *Expert Systems with Applications*, vol. 74, 2017. DOI: 10.1016/j.eswa.2017.01.007.

[13] Kaggle. (2020). “Sales data for a chain of brazilian stores,” [Online]. Available: `https://www.kaggle.com/marcio486/sales-data-for-a-chain-of-brazilian-stores`.