An Environment for Discovery of Associative Rules in Massive Datasets of Market Transactions

Helder Mateus dos Reis Matos[1], **Wilton Freitas Ribeiro**[1],
João Crisóstomo Weyl Albuquerque Costa[2], **Reginaldo Cordeiro dos Santos Filho**[1]

[1] Instituto de Ciências Exatas e Naturais – Universidade Federal do Pará (UFPA)
Belém – PA – Brazil

[2]Instituto de Tecnologia – Universidade Federal do Pará (UFPA)
Belém – PA – Brazil

{helder.matos, wilton.ribeiro}@icen.ufpa.br

{jweyl, regicsf}@ufpa.br

Abstract. *Associative Rule Mining is a data mining technique to extract sets of elements frequently associated with each other, originally developed in Market Basket Analysis (MBA) settings. A major concern with MBA associative rule mining is the availability of computational resources needed to process large collections of data, especially in time-dependent domains like markets. A knowledge-extraction-based environment is proposed to accommodate best practices to process massive MBA datasets, along with use cases of algorithms dedicated to generating associative rules. Market companies can adopt this environment to enhance marketing strategies, improve inventory management, and optimize business rules for maximum profit.*

1. Introduction

Market stores can greatly benefit from analysis of data generated in purchases, due to the volume and variability of items that can be bought. Market Basket Analysis (MBA) is a technique that helps market stores understand customer shopping patterns as well as the relationships between purchased products. Understanding this information is essential for increasing sales, improving the effectiveness of marketing and promotional campaigns, as well as aiding in strategic decision-making such as inventory management and shelf organization. Through data analysis, MBA enables market stores to adjust their sales strategies to meet the needs of their customers and thus improve profitability and loyalty.

This paper aims to describe the creation of an environment for Association Rule Mining (ARM) in MBA scenarios, highlighting computational resource management concerns that need to be taken into account when dealing with large volumes of data, especially in terms of memory availability. The construction of the environment revolves around the mining of a supermarket dataset, which is structured under the perspective of a process model designed for data-driven purposes. As a result, the rules extracted can be interpreted and validated by the supermarket chain that provided the dataset, and potentially be used to improve its logistics and management of products.

This paper is organized as follows: Section 2 reviews relevant literature. Section 3 details the methodology. Section 4 discusses the results. Section 5 provides final considerations and future prospects.

2. Related Work

The literature related to ARM highlights the concepts and tools to be used throughout the research.

Improvements in the use of Apriori on MBA scenarios were made in [Yoseph and Heikkilä 2020], which described preprocessing steps that reduce the dataset size, discovered meaningful rules by analyzing metrics, found trends, and popularity indexes in the itemsets, and measured the effect of the Pareto Rule on inventory categorization, by applying those techniques in a retailer dataset based on Kuwait.

In [Celik et al. 2020], a memory-efficient sampling method was developed and validated with banking datasets. Eight techniques were applied for determining sample size, using variables like minimum support, minimum confidence, and error probability. Results indicated that these techniques did not produce sufficiently similar samples compared to the full dataset for association mining. As expected, similarity error decreased with larger sample sizes when comparing results from samples versus the full dataset.

The application of associative rules was further explored in [Schonhorst et al. 2017], where consumer buying behavior patterns were collected from transactional data of supermarkets in Brazil, through Association Rules techniques. [Gino et al. 2023] explores network visualization of associative rules based on their social structure and temporal dynamics. [Kiani 2020] uses both Apriori and FP-Growth algorithms to analyze consumer behavior on a transactional database from market stores, considering the temporal variables that impact the display period of the products exhibited in the store. In [Mohapatra et al. 2021], aside from the application of association rule mining techniques such as Eclat and Apriori, advancements like collaborative filtering, reinforcement learning, and clustering methods were suggested for personalized recommendations and enhanced production strategies.

This paper introduces a methodology for generating associative rules in large MBA datasets, addressing sampling, memory management, and algorithm parameterization for mining frequent itemsets and creating rules. The techniques are validated using a supermarket chain dataset, facilitating discussions on real-world rule generation.

3. A methodological approach to generate associative rules in massive MBA datasets

This section describes the dataset used to validate the techniques to be employed and the relevant remarks on the construction of an environment for processing massive MBA data: basket creation, frequent itemset mining, and association rules extraction.

The source code for this project is publicly available[1] to facilitate transparency, reproducibility, and further research. It consists of a Python package containing modules for the processes described in this section.

3.1. Dataset description

The dataset is composed of 6.13 million samples of tickets from a supermarket chain based in the State of Pará, Brazil, emitted between January 1st, 2022, and December 31,

[1]Anonymous GitHub: `https://anonymous.4open.science/r/ruletune/`

2023. The data is stored in *Comma-Separated Values* (CSV) files. Some of the relational tables found include:

- Ticket: general information about purchases, such as date, total value and identification of items purchased. It is the central link between the tables;
- Item: an instance of a product bought in a ticket with a quantity multiplier, providing the value of a product at the time of the purchase;
- Product: general information about a product, its hierarchical aggregations, standard barcode numbering, and packaging method;
- Department: first level of hierarchical aggregation of products, such as groceries, home center, pharmaceuticals, etc. Products can be categorized into at least 40 different departments;
- Section: second level of hierarchical aggregation of products. Products can be categorized into at least 238 different sections;
- Group: third level of hierarchical aggregation of products. Products can be categorized into at least 1674 different groups;
- Subgroup: fourth level of hierarchical aggregation of products. Products can be categorized into at least 5478 different subgroups.

The dataset is not publicly available due to the supermarket chain's sensitive information it contains, such as receipts and client purchase patterns. However, the source code includes a well-known benchmarking dataset[2], following the format of typical MBA datasets used in associative rule mining frameworks. This inclusion eases the formatting of arbitrary datasets for reproducibility purposes, assisting prospective users of the proposed environment in preparing compatible datasets, enabling them to utilize most functionalities and parameters designed for processing massive MBA data.

Most of the steps required for preprocessing this dataset are listed in [Matos et al. 2024], including the analysis conducted to better understand attributes and their relationships, and the data preparation process that transforms the data into a suitable format for modeling techniques.

ABC Analysis is a key technique in the data preparation process that can be crucial for enhancing the creation of associative rules in MBA data. This inventory categorization method classifies each product, known as a *Stock Keeping Unit* (SKU), into one of three categories based on its cumulative contribution to total sales. Products in Class A are highly profitable, while those in Class C are less so.

The Percentual SKU is the key variable used in ABC classification, as expressed in Equation 1, where $PSKU_i$ is the percentual SKU of an arbitrary product i, CS_i is the cumulative total value up to this product, and T is the total value of all purchases. Thus, the minor the $PSKU$, the greater the contribution of the product to the total revenue.

$$PSKU\%_i = \frac{CS_i}{T} \quad (1)$$

The categorization of how much a product contributes to the generation of revenue in a given period also reveals what products are more likely to be bought in most of the

[2]Frequent Itemset Mining Dataset Repository: `http://fimi.uantwerpen.be/data/`

tickets, which is an important factor when analyzing items bought together. By considering only the transactions that include items of higher categories, one can greatly reduce the amount of data fed to associative rule learning algorithms.

3.2. An environment to mine associative rules in massive MBA data

An environment for the generation of associative rules is designed to embrace different forms of baskets and product categorization, along with memory consumption concerns that can be generalized to most MBA rule mining systems. The environment was implemented as a Python package, composed of three modules: creation of baskets, generation of frequent itemsets, and extraction of associative rules.

3.2.1. Creation of baskets

A basket is a data collection representing items bought in each transaction, typically as an $m \times n$ matrix, where m is the number of transactions and n is the number of unique SKUs. Figure 1 shows the process to create a sparse matrix from transactions. First, a table of items and SKUs is provided. Step 2 involves extracting sorted lists of unique tickets and SKUs. In Step 3, these lists' indexes are used to create row and column lists, along with a boolean list for matrix positions. Finally, a sparse matrix is created using these lists. The Compressed Sparse Row (CSR) class from SciPy can hold such matrices, which can be converted into sparse DataFrames for use in data mining libraries supported by Pandas.

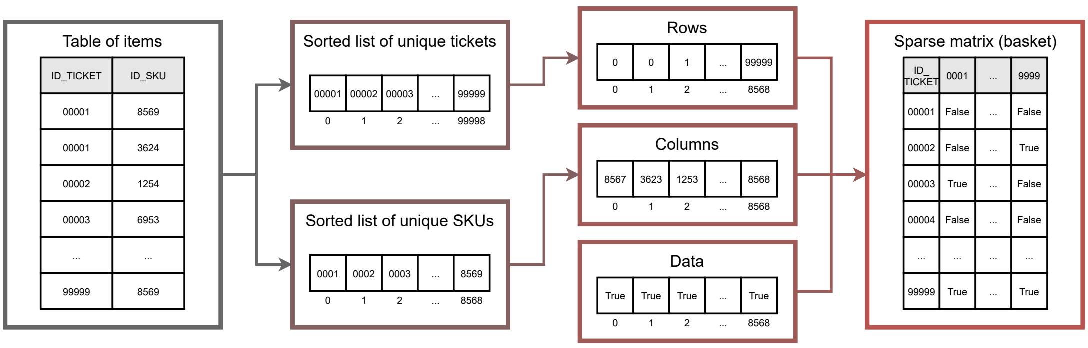

Figure 1. The steps to create a basket.

The performance of basket creation depends heavily on the domain where the associative rules are executed, specifically the table of items purchased, which determines the matrix's dimensionality. The best case scenario occurs in domains with many tickets and fewer unique SKUs, likely leading to more items bought together, resulting in stronger rules. On the other hand, domains with few transactions and many unique items tend to create few weak rules and should be avoided. Common domains for effective basket creation include:

- Hierarchical aggregation: categories of SKUs, such as departments and their subdivisions present in the aforementioned dataset, or attributes constructed for standardized systems, such as EAN, can reduce the number of unique products.
- Periods: day, week, month, year, etc. It can be used to reduce the number of transactions if the number of instances is huge.

- Locations: cities, sociodemographic regions, branches of the supermarket, etc.
- Payment: cash, credit card, instant payment, etc.

3.2.2. Generation of frequent itemsets

The generation of frequent itemsets (FI) is the most memory-intensive set of tasks to be addressed. There is a great variety of Frequent Itemsets Mining (FIM) algorithms, with different use cases and data complexity scenarios. A set of four popular FIM algorithms can be found in the Python library mlxtend [Raschka 2018], which will be used in the construction of the MBA rule mining environment:

- Apriori [Agrawal and Srikant 1994]: starts by counting occurrences of frequent itemsets of size 1. It then proceeds by generating candidate itemsets from the previously identified itemsets and computing their support. Only candidates with support meeting or exceeding the minimum support threshold are retained for the next step. Apriori introduced the candidate-generation-and-test approach for mining frequent itemsets, utilizing the anti-monotone property: any superset of a non-frequent itemset is also non-frequent.
- FP-Growth [Han et al. 2004]: optimizes the query of itemsets within the database by creating instances of an auxiliary data structure, called FP-Tree (Frequent-Pattern Tree). Then, a pattern growth process successively concatenates frequent itemsets of size 1, reducing the size of subsequent FP-trees by partitioning those structures based on a divide-and-conquer approach.
- FP-Max [Grahne and Zhu 2003]: is a variation of the FP-Growth algorithm designed to find maximal frequent itemsets, those that are not subsets of any larger frequent itemset. FP-Max is particularly effective for sparse datasets, as it employs an array to minimize the traversal effort required for FP-Trees.
- H-Mine [Pei et al. 2007]: extends pattern-growth algorithms by using a memory-based hyper-structure, H-Struct, and the partitioning and traversing of the subpartitions to find frequent itemsets. Then, global frequent patterns are constructed by combining the results of the recursive calls. H-Mine is designed to solve two problems: the unpredictable allocation size of data structure needed for mining frequent itemsets and the scalability for both dense and sparse datasets.

Itemsets can be compared by their support value, which can be calculated by Equation 2. Consider the itemset I and a basket B composed of m transactions T, the support of I is the fraction of transactions that have I as a subset.

$$support(I) = \frac{\mid \{T \in B \mid I \subseteq T\} \mid}{\mid B \mid} \tag{2}$$

Frequent itemsets are the prime source for generating associative rules and thus must contain sufficient sets to combine rules. When running an FIM algorithm, the minimum support is a common parameter to set a threshold for the combination of sets to be presented [Leskovec et al. 2019]. The value of minimum support depends on the dimensionality of the provided basket, the distribution of items inside the basket, and how many itemsets are needed to create important rules.

The main concern regarding the value of minimum support is how much memory will be used during the execution of an FIM algorithm, since low support thresholds result in a deeper exploration of itemsets. Consequently, the allocation of memory to hold itemsets also impacts the execution time of an FIM algorithm, with some running indefinitely despite not using all the memory. Moreover, there must be a balance between the exploration cutting provided by this parameter and the maximization of itemsets that will be used to extract rules.

To find the optimal minimum support value between 0 and 1, a sequential search is performed, starting with values to three decimal places and progressing to one decimal place, as shown in Figure 2. Each candidate value is tested with an FIM algorithm, with execution monitored for memory and time constraints. For instance, if a run uses up to 90% of memory and completes within 1 hour, it indicates that frequent itemsets were found efficiently. The goal is to identify the minimum support value that yields the highest number of frequent itemsets.

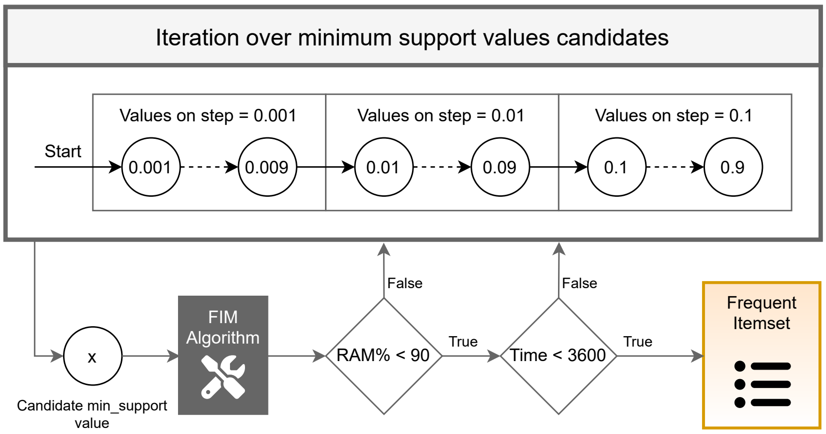

Figure 2. Search for optimal minimal support based on time and memory constraints.

3.2.3. Extraction of associative rules

The final step of the environment structure involves extracting rules that search for associations within the frequent itemsets. Figure 3 illustrates two conditions to verify the number of frequent itemsets and associative rules extracted. A list of frequent itemsets containing more than one item can be inputted into the association rule algorithm, also implemented in [Raschka 2018]. To enhance readability, SKU IDs in the rules can be replaced with descriptions using another data structure that translates IDs. If either the frequent itemsets or the association rules structures are empty, no rules are extracted.

3.3. Evaluation

This step verifies the fulfillment of the success criteria defined in business understanding over the models, generating improvement insights for the next iterations of the modeling step. A set of metrics are listed in Table 1, and can be obtained from the associative rules generated, which can be used to assess their usefulness.

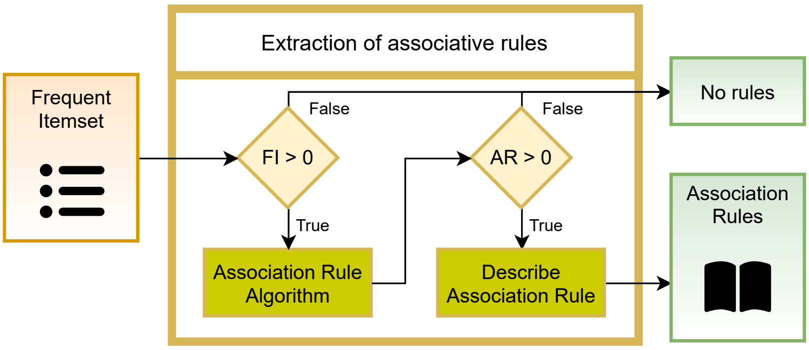

Figure 3. Extraction of associative rules.

Table 1. Metrics extracted from the generated rules.

Metric	Description	Formula	Interval	Paper
Support	Combines the support of the antecedent A and Consequent C.	$support(A \to C) = support(A \cup C)$	$[0, 1]$	[Agrawal et al. 1993]
Confidence	Probability of seeing C, given that A had occurred.	$confidence(A \to C) = \frac{support(A \cup C)}{support(A)}$	$[0, 1]$	[Agrawal et al. 1993]
Lift	Probability of occurrence of $A \to C$ if both were independent events.	$lift(A \to C) = \frac{confidence(A \to C)}{support(C)}$	$[0, \infty]$	[Brin et al. 1997]
Leverage	Measures how often A and C would appear together if they were independent.	$leverage(A \to C) = support(A \cup C) -$ $- support(A) \times support(C)$	$[-1, 1]$	[Piatetsky-Shapiro 1991]
Conviction	Measures how much the consequent is dependent on the antecedent.	$conviction(A \to C) = \frac{1 - support(C)}{1 - confidence(A \to C)}$	$[0, \infty]$	[Brin et al. 1997]
Zhang's metric	Measures association ($zhang > 0$) and dissociation ($zhang < 0$).	$num = confidence(A \to C) - confidence(A' \to C)$ $den = Max[confidence(A \to C), confidence(A' \to C)]$ $zhang(A \to C) = \frac{num}{den}$	$[-1, 1]$	[Yan et al. 2009]

4. Results and Discussion

The results are set to be discussed in two parts, representing the domains where the association rules can be explored: from an Universal Basket and from Hierarchical Aggregated Baskets, or from Departmental Baskets.

The dimension of the baskets used in ARM is given by the number of transactions and the number of unique SKUs found on these transactions, while a higher number of transactions and lower number of unique SKUs is desirable to discover better rules.

Figure 4 illustrates an ABC classification model for approximately 280 thousand SKUs. Class A, comprising 6% of items, contributes 80% of revenue, while Class C, with 83% of items, accounts for only 10% of revenue. The analysis effectively reduces product numbers by focusing on the most profitable items in classes A and B, leading to a sparse basket with 3.8 million instances and 16 thousand SKUs, compared to the original 43 million instances and 280 thousand SKUs.

4.1. Universal Basket

The first experiment was conducted in a basket containing all transactions available with a shape of 3,848,238 rows and 16,582 columns, allocated in 411.78 MB of RAM.

Table 2 shows the minimum support values found for each algorithm that max-

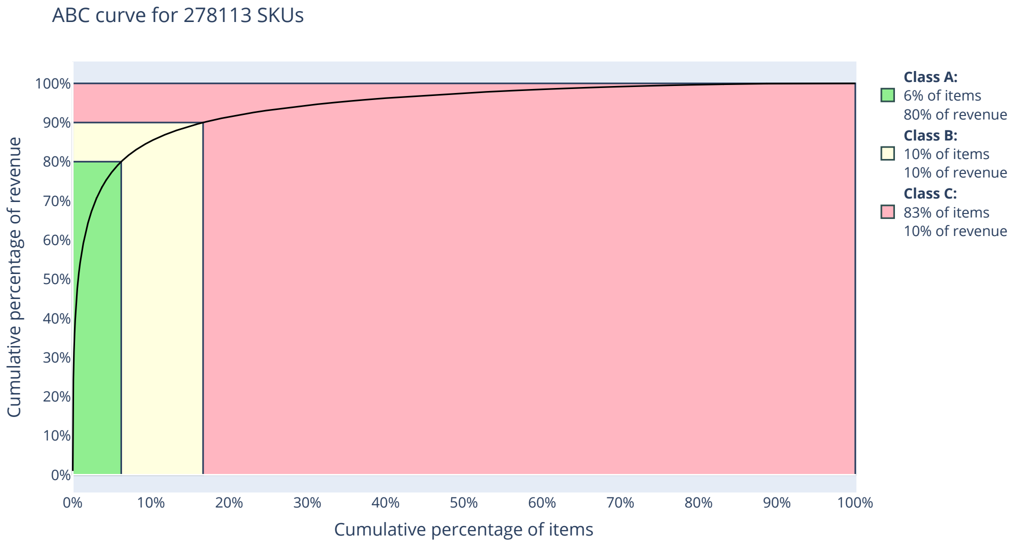

Figure 4. ABC curve that relates the cumulative percentage of revenue and items.

imizes the number of frequent itemsets (FIs) while avoiding memory errors. Apriori generates the fewest FIs due to the higher minimum support needed to execute without running out of memory. Both FP-based algorithms generate almost the same number of FIs with slight variations in minimum support. H-Mine operates at lower support values and returns the most FIs.

Table 2. Number of results for minimum supports applied for each algorithm in the universal basket.

Frequent Itemset Algorithm	Optimal Minimum Support	Number of Frequent Itemsets	Number of Associative Rules	Maximum Confidence
Apriori	0.02	54	38	0.634
FP-Growth	0.004	982	3230	0.839
FP-Max	0.003	1053	4432	-
H-Mine	0.003	1685	7010	0.859

Regarding rules, three algorithms computed rules and extracted their confidence, except for FP-Max, which ran out of memory while computing metrics other than support. Apriori not only generated the fewest rules, but its rule with the highest confidence was the lowest among the three algorithms that computed metrics. The best quantitative results were found by H-Mine, whose rule with the highest confidence reached 85.9%. Table 3 organizes H-Mine's 5 best rules for the universal basket, whose average confidence value revolves around 84.1%. Despite the higher confidence, all products in these rules are from the green groceries department, which are highly perishable, purchased in small quantities, and present in most tickets.

4.2. Hierarchical Aggregated Basket

A second basket domain was created by grouping products into unique identifiers based on hierarchical categories: Department, Section, Group, and Subgroup. We used Sub-

Table 3. Rules for the universal basket based on H-Mine's itemsets.

Antecedents	Consequents	Support	Confidence	Lift	Leverage	Conviction	Zhang's Metric
GREEN BELL PEPPER KG, WASHED POTATOES KG, CARROT KG, TOMATO KG	ONION KG	0.003440	0.859564	8.991461	0.003058	6.439946	0.892355
GREEN ONIONS, GREEN BELL PEPPER KG, CARROT KG, TOMATO KG	ONION KG	0.003451	0.849856	8.889914	0.003063	6.023566	0.891131
LIME KG, WASHED POTATOES KG, CARROT KG, TOMATO KG	ONION KG	0.003289	0.847916	8.869624	0.002918	5.946743	0.890710
GREEN ONIONS, LIME KG, CARROT KG, TOMATO KG	ONION KG	0.003159	0.847876	8.869203	0.002803	5.945169	0.890568
GREEN ONIONS, WASHED POTATOES KG, CARROT KG, TOMATO KG	ONION KG	0.004387	0.839867	8.785421	0.003888	5.647808	0.890828

group for more specific product labels. This aggregated basket is smaller than the universal basket, with 3,696,577 rows and 1,774 columns, totaling 373.16 MB. Tickets with products from the same subgroup were reduced to a single instance, as ARM focuses on label occurrence rather than quantity.

Table 4 shows the optimal minimum support and results for the four algorithms. Apriori ran up to a support of 0.03 and generated the lowest number of itemsets, while FP-based algorithms generated many more itemsets and rules, using a minimum support of 0.008. H-Mine stuck its execution indefinitely at a minimum support of 0.005.

Table 4. Number of results for minimum supports applied for each algorithm in the hierarchical aggregated basket.

Frequent Itemset Algorithm	Optimal Minimum Support	Number of Frequent Itemsets	Number of Associative Rules	Maximum Confidence
Apriori	0.03	103	318	0.88
FP-Growth	0.008	1371	9962	0.027
FP-Max	0.008	538	4078	-
H-Mine	0.005	-	-	-

As for the rules, Apriori's best rule has a confidence of 88%, while FP-Growth's rules were not satisfactory, and FP-Max didn't compute its metrics as it ran out of memory. Table 5 organizes Apriori's 5 best rules, which are mostly the same found for other algorithms. Once again, the subgroups belonging to the green groceries department dominated the most confident rules, emphasizing the need to strategically select market segments when creating baskets for processing.

4.3. Departmental Baskets

Lastly, an attempt to reduce the number of transactions is performed by choosing tickets that are more prominent to have products of a few categories of products. A selection of the tickets containing only products of the pharmaceutics department was conducted, since there are cashiers specialized in these products, with a great potential for inter-

Table 5. Rules for the hierarchical aggregated basket based on Apriori's itemsets.

Antecedents	Consequents	Support	Confidence	Lift	Leverage	Conviction	Zhang's Metric
FRESH REGIONAL FRUITS, FRESH REGIONAL GREENS, VEGETABLES, FRESH NATIONAL FRUITS	TUBERS/ROOTS/BULBS	0.030608	0.880643	4.676112	0.024062	6.800366	0.814455
FRESH REGIONAL FRUITS, FRESH REGIONAL GREENS, VEGETABLES	TUBERS/ROOTS/BULBS	0.036568	0.864009	4.587790	0.028597	5.968589	0.816591
FRESH REGIONAL FRUITS, VEGETABLES, FRESH NATIONAL FRUITS	TUBERS/ROOTS/BULBS	0.038671	0.850204	4.514483	0.030105	5.418505	0.815587
FRESH NATIONAL GREENS, VEGETABLES	TUBERS/ROOTS/BULBS	0.035716	0.841778	4.469744	0.027725	5.129959	0.810670
FRESH REGIONAL FRUITS, FRESH REGIONAL GREENS, TUBERS/ROOTS/BULBS, VEGETABLES	FRESH NATIONAL FRUITS	0.030608	0.837020	2.767492	0.019548	4.279988	0.662903

pretability of medicines bought together. The pharmaceutical basket has 183,450 transaction rows and 2,231 unique SKU columns, allocated in 14.67 MB.

Table 6 presents the minimum support values and results, with all four algorithms executed to the third decimal place. The number of frequent itemsets exceeded 600, except for Apriori, which was interrupted at a higher support value compared to the others. Apriori generated 14 rules, whereas the other three algorithms each produced 86 rules. The performance of this basket surpassed that of the previous two experiments, suggesting an optimal balance between the number of transactions and the associations extracted.

Table 6. Number of results for minimum supports applied for each algorithm in the pharmaceutical basket.

Frequent Itemset Algorithm	Optimal Minimum Support	Number of Frequent Itemsets	Number of Associative Rules	Maximum Confidence
Apriori	0.003	199	14	0.406
FP-Growth	0.001	695	86	0.762
FP-Max	0.001	666	86	-
H-Mine	0.001	695	86	0.762

Both FP-Growth and H-Mine's most confident rules were the same, at 76.2%. Table 7 organizes FP-Growth's 10 best rules for the pharmaceutics basket, with an average of) 30%.

Table 7. Rules for the pharmaceutical basket based on FP-Growth's itemsets

Antecedents	Consequents	Support	Confidence	Lift	Leverage	Conviction	Zhang's Metric
ALGESTONE 150 MG/ML + ESTRADIOL 10MG/ML	SYRINGE 3 ML 25X7	0.001226	0.762712	141.047875	0.001218	4.191497	0.994509
HYDROCHLOROTHIAZIDE EMS 25MG 30 TABS	LOSARTAN EMS 50MG 30 TABS	0.003794	0.406780	20.065536	0.003605	1.651541	0.959109
ATENOLOL EMS 25MG 30 TABS	LOSARTAN EMS 50MG 30 TABS	0.001532	0.263850	13.015122	0.001414	1.330880	0.928557
ENGOV 6 TABS	EPOCLER 1 FLASK 10ML	0.002873	0.263632	5.326350	0.002333	1.290800	0.821203

These rules can be explained as follows:

- If *Algestone + Estradiol* is purchased, so it is *Syringe*: related to a injectable contraceptive.
- If *Hydrochlorothiazide* is purchased, so it is *Losartan*: both are medicines for high blood pressure control.
- If *Engov* is purchased, so it is *Epocler*: both are medicines for the treatment of headache, heartburn, nausea, etc.

5. Conclusion

The paper focused on the development of an environment to generate associative rules in the context of Market Basket Analysis. There were key concerns to be considered when mining huge volumes of MBA data, including the complexity of data, the dimensionality of the baskets to be processed, and the importance of set domains of exploration that favor computing such datasets. The results have the potential to improve operational efficiency, strategic decision-making, and customer experience in a supermarket chain company.

When applying associative mining algorithms to high-dimensional data, addressing dataset complexity is essential for practitioners. Ensuring that the dataset is well-prepared and that the item domain is carefully refined can significantly alleviate the computational load of frequent itemset algorithms. This preparation not only enhances performance but also often leads to the generation of more interpretable and valuable rules.

In future work, the interpretability and usefulness of the associative rules are an emerging topic to be addressed, as the volume of rules created demands an automatic processing that put these to practice, enabling decision-making strategies. The impacts on the logistics methodology and management of supermarket companies regarding the implementation of the presented data mining tools in their routines should be presented, comparing impact factors such as revenue, ticket issuance, and product turnover before and after the application of these techniques.

References

Agrawal, R., Imieliński, T., and Swami, A. (1993). Mining association rules between sets of items in large databases. *SIGMOD Rec.*, 22(2):207–216.

Agrawal, R. and Srikant, R. (1994). Fast Algorithms for Mining Association Rules in Large Databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB '94, page 487–499, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Brin, S., Motwani, R., Ullman, J. D., and Tsur, S. (1997). Dynamic itemset counting and implication rules for market basket data. In *Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data*, SIGMOD '97, page 255–264, New York, NY, USA. Association for Computing Machinery.

Celik, O., Hasanbasoglu, M., Aktas, M. S., and Kalipsiz, O. (2020). Association Rule Mining on Big Data Sets. In Birant, D., editor, *Data Mining*, chapter 3. IntechOpen, Rijeka.

Gino, H., Pedro, D., Ponciano, J., Linhares, C., and Traina, A. (2023). Exploratory analysis on market basket data using network visualization. In *Anais do XII Brazilian*

Workshop on Social Network Analysis and Mining, pages 19–30, Porto Alegre, RS, Brasil. SBC.

Grahne, G. and Zhu, J. (2003). Efficiently Using Prefix-trees in Mining Frequent Itemsets. In *Workshop on Frequent Itemset Mining Implementations*.

Han, J., Pei, J., Yin, Y., and Mao, R. (2004). Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach. *Data Mining and Knowledge Discovery*, 8(1):53–87.

Kiani, G. H. (2020). Determining profitable products in the retail market with consideration of cash limitation and exhibition periods. *Journal of Retailing and Consumer Services*, 55:102079.

Leskovec, J., Rajaraman, A., and Ullman, J. D. (2019). *Mining of Massive Datasets*. Cambridge University Press, third edition.

Matos, H., Ribeiro, W., dos Santos Filho, R., and Costa, J. (2024). Descoberta de Padrões e Tendências de Compras em Supermercados Utilizando Análise de Cestas de Compras. In *CONTECSI USP - International Conference on Information Systems and Technology Management - ISSN 2448-1041*.

Mohapatra, D., Tripathy, J., Mohanty, K. K., and Nayak, D. S. K. (2021). Interpretation of Optimized Hyper Parameters in Associative Rule Learning using Eclat and Apriori. In *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*. IEEE.

Pei, J., Han, J., Lu†, H., Nishio, S., Tang, S., and Yang, D. (2007). H-Mine: Fast and space-preserving frequent pattern mining in large databases. *IIE Transactions*, 39(6):593–605.

Piatetsky-Shapiro, G. (1991). Discovery, analysis, and presentation of strong rules. In *Knowledge Discovery in Databases*.

Raschka, S. (2018). MLxtend: Providing machine learning and data science utilities and extensions to Python's scientific computing stack. *The Journal of Open Source Software*, 3(24).

Schonhorst, G., Paes, V., Balestrassi, P., Paiva, A., and Campos, P. (2017). Data mining association rules applied to supermarket transactional data modeling: a case study in brazil. In *International Joint Conference - ICIEOM-ADINGOR-IISE-AIM-ASEM (IJC 2017)*.

Yan, X., Zhang, C., and Zhang, S. (2009). Confidence metrics for association rule mining. *Applied Artificial Intelligence*, 23(8):713–737.

Yoseph, F. and Heikkilä, M. (2020). A new approach for association rules mining using computational and artificial intelligence. *Journal of Intelligent & Fuzzy Systems*, 39(5):7233–7246.