MARKET BASKET ANALYSIS

A PROJECT REPORT

Submitted by

AJAY KUMAR K B	**113320104003**
JAGADEESAN D	**113320104040**

In partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

In

COMPUTER SCIENCE AND ENGINEERING

VELAMMAL INSTITUTE OF TECHNOLOGY,

CHENNAI 601 204

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2023

BONAFIDE CERTIFICATE

Certified that this project report "**MARKET BASKET ANALYSIS"** is the bonafide work of "**AJAY KUMAR K B 113320104003, JAGADEESAN D 113320104040"** who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr. V. P. Gladis Pushparathi, M.Tech.,Ph.D .,	**Mr. J .A. Jevin., M.E .,**
HEAD OF THE DEPARTMENT	ASSISTANT PROFESSOR
Computer Science and Engg.,	Computer Science and Engg.,
Velammal Institute of Technology	Velammal Institute of Technology
Velammal Garden Panchetti	Velammal Garden Panchetti
Chennai-601 204.	Chennai-601 204.

MARKET BASKET ANALYSIS

VIVA-VOCE EXAMINATION

The viva-voce examination of this project work was done as a part of the Bachelor's Degree in Computer Science and Engineering held on ________________________

AJAY KUMAR K B	**113320104003**
JAGADEESAN D	**113320104040**

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We are personally indebted to many who had helped us during the course of this project work. Our deepest gratitude to the **God Almighty**.

We are greatly and profoundly thankful to our beloved Chairman **Thiru.M.V.Muthuramalingam** for facilitating us with this opportunity. Our sincere thanks to our respected Director **Thiru. M. V. M. Sasi Kumar** who took keen interest on us. We are also thankful to our Advisors **Prof. K. Razak**, **Shri. M. Vasu** and our Principal **Dr. N. Balaji** and our Vice Principal **Dr.S.Soundararajan** for their never-ending encouragement which accelerates us towards innovation.

We are extremely thankful to our Head of the Department **Dr.V.P.Gladis Pushparathi**, Project Coordinator **Ms. Pranamita Nanda,** Assistant Professor, Computer Science and Engineering Department for their valuable teachings and suggestions. From the bottom of our heart, we would like to thank our guide **Mr.Jevin** who has been the pillar of this project without whom we would not have been able to complete the project successfully.

The Acknowledgment would be incomplete if we would not mention word of thanks to our **Parents, Teaching and Non-Teaching Staffs, Administrative Staffs and Friends** who had motivated and lend their support throughout the project. Thank you one and all.

TABLE OF CONTENTS

ABSTRACT

Market Basket Analysis is one of the key techniques used by large retailers to uncover associations between items. It works by looking for combinations of items that occur together frequently in transactions. To put it another way, it allows retailers to identify relationships between the items that people buy. Association Rules are widely used to analyze retail basket or transaction data, and are intended to identify strong rules discovered in transaction data using measures of interestingness, based on the concept of strong rules. The outcome of this type of technique is, in simple terms, a set of rules that can be understood as "if this, then that". Association rule mining finds interesting associations and relationships among large sets of data items. This rule shows how frequently an itemset occurs in a transaction. The project is developed by applying and utilizing the Association rules on the data item list. Here, Association rule is used in such a way that by importing the Apriori module. Apriori module is the concept from Machine Learning. Here, Machine Learning concept is utilized using Python to achieve the aspect of the Project. Given a set of transactions, we can find rules that will predict the occurrence of an item based on the occurrences of other items in the transactions. Here before applying the rule to the data set, the basic definitions such as Support count, Frequent Itemset are ensured from the data set that is passed to the system. Once these terms are ensured from the data set, then the Evaluation Metrics such as Support, Confidence and Lift value are noticed from the data set. Once done, then the Association rule from Apriori module is applied to the dataset. This is termed as applying data mining to the dataset. Then the Market Basket is analyzed for the most frequent item/product bought by the customer.

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Problem Statement

Market Basket Analysis is one of the key techniques used by large retailers to uncover associations between items. It works by looking for combinations of items that occur together frequently in transactions. Though, the customer buys the items/product from the market, there is inconvenient for the shop keeper to know what the customers prefer more to buy. In order to confirm that, the shop keeper needs an analysis software/system that evaluates the market basket from the customer and lists out the various products/items are on demand by the customers. By doing such thing, it helps and it is convenient to the shop keeper to have the stock of those preferred items for the customer in advance. So that, it gives profit to the shop keeper than before.

1.2 Project Scope and Objective:

1.2.1Scope of the Project:

The scope of the project is to address the above-mentioned issue and solve it accordingly. Our solution for the above problem is that, we create a project that utilizes the Apriori module from Machine Learning domain. From the Apriori module, we use Association rule that is applied to the dataset provided to the system to find the most purchased product by the customer through analyzing the market basket.

1.2.2 Objective of the Project:

Our main objective is to provide the system for the Market Basket Analysis so that it helps to the shop keeper to analyze the customers' shopping basket to view the details of the most frequently purchased products/items. To achieve this, Association rules of Apriori module from the Machine Learning is used and applied to the dataset using Python. Then the system lists the frequently purchased products.

CHAPTER 2

OVERALL DESCRIPTION

2.1 Project Specification:

Our project is the Market Basket Analysis for the Shop keeper to analyze the frequently purchased items by the customer. Market Basket Analysis is one of the key techniques used by large retailers to uncover associations between items. It works by looking for combinations of items that occur together frequently in transactions. To put it another way, it allows retailers to identify relationships between the items that people buy.

As stated earlier**,** our project is the Market Basket Analysis for the Shop keeper to analyze the frequently purchased item by the customers. First of all, minimum of 6 months is required to ensure the demand for the products by the customer. Then, based on these 6 months period, the frequently purchased products are found. From these, the shop keeper ensures the frequently purchased products and have the stock accordingly. Then, from this time, the Market Basket of the customers are analyzed time to time to keep on updating the frequently purchased items to make profit by having the stock of the products all the time. For, this the purchase history by the customer is collected from the shop keeper database or from the cashier. This purchase history is passed as an input dataset to the system. This dataset i.e., purchase history is analyzed in such a way that the system produces the output dataset that consists of the frequently purchased products by the customer. This project helps in two ways. One is that, by ensuring the frequently purchased products, one can have the stocks of those products preliminarily. The other merit is that, if the mart is whole sale market, definitely, the whole sale market sells the products at less than maximum retail price MRP, by ensuring the frequently bought items/products one can view them and increase the selling price to little bit closer to MRP as it sells the products less than MRP. By this way, it ensures profit to the shop keeper. Then, if the shop keeper ensures both the merits/conclusions of the project, the shop keeper would make use of this project/system efficiently at the fullest. Association Rules are widely used to analyze retail basket or transaction data, and are intended to identify strong rules discovered in transaction data using measures of interestingness, based on the concept of strong rules. The outcome of this type of technique is, in simple terms, a set of rules that can be understood as “if this, then that”. Association rule mining finds interesting associations and relationships among large sets of data items. This rule shows how frequently an itemset occurs in a transaction.

Association analysis is also known as affinity analysis or association rule mining, a method commonly used for market basket analysis. ARM is currently the most suitable method for analysis of big market basket data but when there is a large volume of sales transaction with high number of products, the data matrix to be used for association rule mining usually ends up large and sparse, resulting in longer time to process data. Association rules provide information of this type in the form of "IF-THEN" statements. There are three indexes which are commonly used to understand the presence, nature and strength of an association rule.

Lift is obtained first because it provides information on whether an association exist or not or if the association is positive or negative. If the value for lift suggests that there is an existence of association rule, then we obtain the value for support.

Support of an item or itemset is the fraction of transactions in our dataset that contain that item or itemset. It is an important measure because a rule that have low support may occur simply by chance. A low support rule may also be uninteresting from a business perspective because it may not be profitable to promote items that are seldom bought together. For these reasons, support is often used to eliminate uninteresting rules.

Confidence is defined as the conditional probability that shows that the transaction containing the LHS will also contain RHS. Association analysis results should be interpreted with caution. The 16 inference made by an association rules does not necessarily imply causality. Instead, it suggests a strong co-occurrence relationship between the items in the antecedent and consequent of the rule.

Confidence and support measure the strength of an association rule. Since the transactional database is quite large, there is a higher risk of getting too many unimportant and rules which may not be of our interest. To avoid these kinds of errors we commonly define a threshold of support and confidence prior to the analysis, so that only useful and interesting rules are generated in our result.

Causality:

Ideally, we would like to know that in an association rule the presence of an item/ itemset causes another item/ itemset to be bought. However, "causality" is an elusive concept. Nevertheless, for market-basket data, the following test suggests what causality means. If we lower the price of diapers and raise the price of beer, we can lure diaper buyers, who are more likely to pick up beer 17 while in the store, thus covering our losses on the diapers. That strategy works because diapers cause beer. However, working it the other way around, running a sale on beer and raising the price of diapers, will not result in beer buyers buying diapers in any great numbers, and we lose money.

Frequent Itemset:

In many (but not all) situations, we only care about association rules or causalities involving sets of items that appear frequently in baskets. For example, we cannot run a good marketing strategy involving items that no one buys anyway. Thus, much data mining starts with the assumption that we only care about sets of items with high support; i.e., they appear together in many baskets. We then find association rules or causalities only involving a high-support set of items. The consequent must appear in at least a certain percent of the baskets, called the support threshold.

Time Period:

Each transaction takes place in a single time period where W = {W1, W2, W3, …, Wt} is the set of all the time periods under consideration. Only one transaction occurs per store, per user in a time period.

Transaction:

In general terms transaction is an agreement, contract, understanding, or transfer of cash or property that takes place between two parties and establishes a legal obligation. In accounting terms events that initiates a change in the asset, liability, or net worth account. Transactions first entry are made in journal and then posted to ledger. From there they move to other accounting books like profit and loss account, balance sheet, etc. In banking a transaction would be an activity performed by the account holder at his/her request which is affecting a bank account. In the language of commerce, a transaction will be considered exchange of goods or services between a buyer and a seller. It has 3 components:

1. Transfer of goods or services and money,
2. Transfer of title which may or may not be accompanied by a transfer of possession,
3. Transfer of exchange rights.

In the field of marketing, a typical transaction consists of a set of products purchased by a customer at a retail store or on a website. These transactions contain all the information about each specific transaction which make up the data entered into the database. These can include information on the customer, information of what products were purchased in what quantity, information on time of purchases, information on if the companies marketing strategies are attracting customers or not, etc. Also, a transaction can take place at one point in time or over time and could involve a day, a quarter, a fiscal year, or even longer periods. Because they are not limited to an event.

Long Tail Effect:

This term often refers to data products purchase in supermarkets describing their distribution as a long tail in which a small number of products is purchased more frequently whereas a large one is purchased less frequently. This phenomenon creates data sparsity problem and worsens even more their elaboration. Studying such data with MBA would be practically helpful because its transactional data and best approach to work with transactional data is to do market basket analysis.

Mining association rules is a clearly defined task. The objective there is to generate all rules of the form $X \Rightarrow Y$ which are above some given support and confidence thresholds. The problem of evaluation and validation is thus reduced to one of correctness and efficiency. Correctness in this case is unambiguous. Any algorithm is required to return *the* set of rules meeting the given support and confidence criteria. Here, we the algorithm called Apriori Algorithm of Machine learning that facilitates the data mining.

APRIORI ALGORITHM:

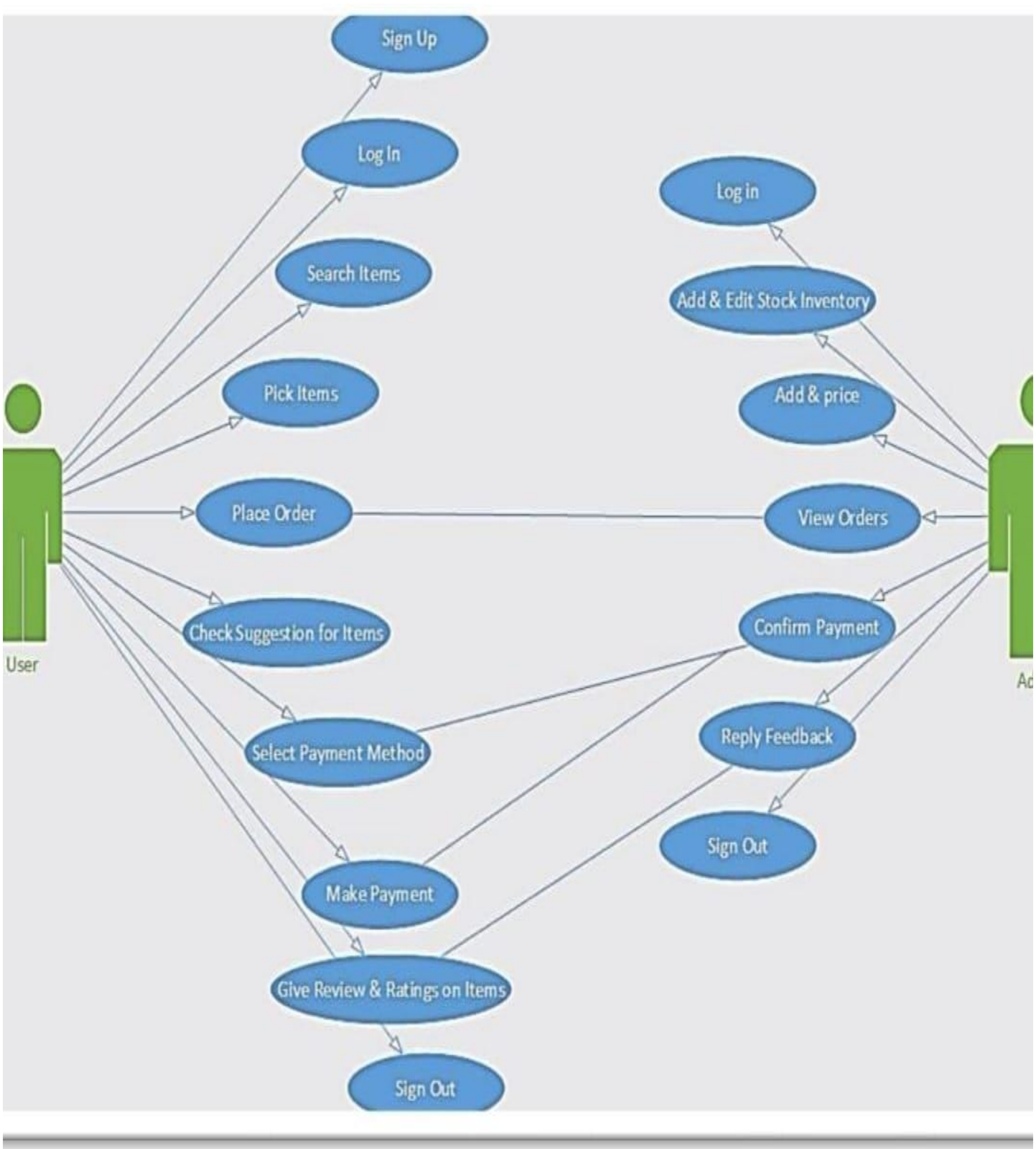
This part will explain how the algorithm that will be running behind the python libraries for Market Basket Analysis. This will help the companies to understand their clients more and analyze their data more closely and attentively. Rakesh Agrawal proposed the Apriori algorithm which was the first associative algorithm proposed and future developments in association, classification, associative classification algorithms have used it as a part of the technique.

Association rule mining is seen as a two-step approach:

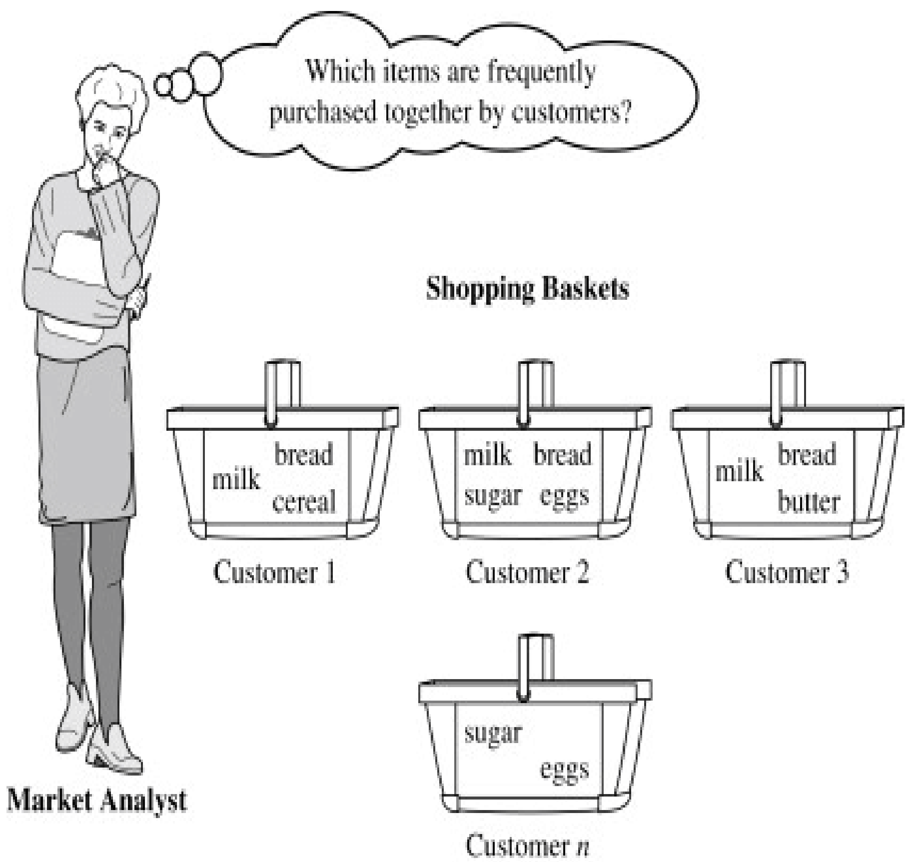
1. Frequent Itemset Generation: Find all frequent item-sets with support >= pre-determined minimum support count. In frequent mining usually the interesting associations and correlations between item sets in transactional and relational databases are found. In short, Frequent Mining shows which items appear together in a transaction or relation. The discovery of frequent item sets is accomplished in several iterations. Counting new candidate item-sets from existing item sets requires scanning the entire training data. In short it involves only two important steps:
    a. Pruning
    b. Joining

2. Rule Generation: List all association rules from frequent item-sets. Calculate Support and Confidence for all the rules. Prune rules which fail minimum support and minimum confidence thresholds.

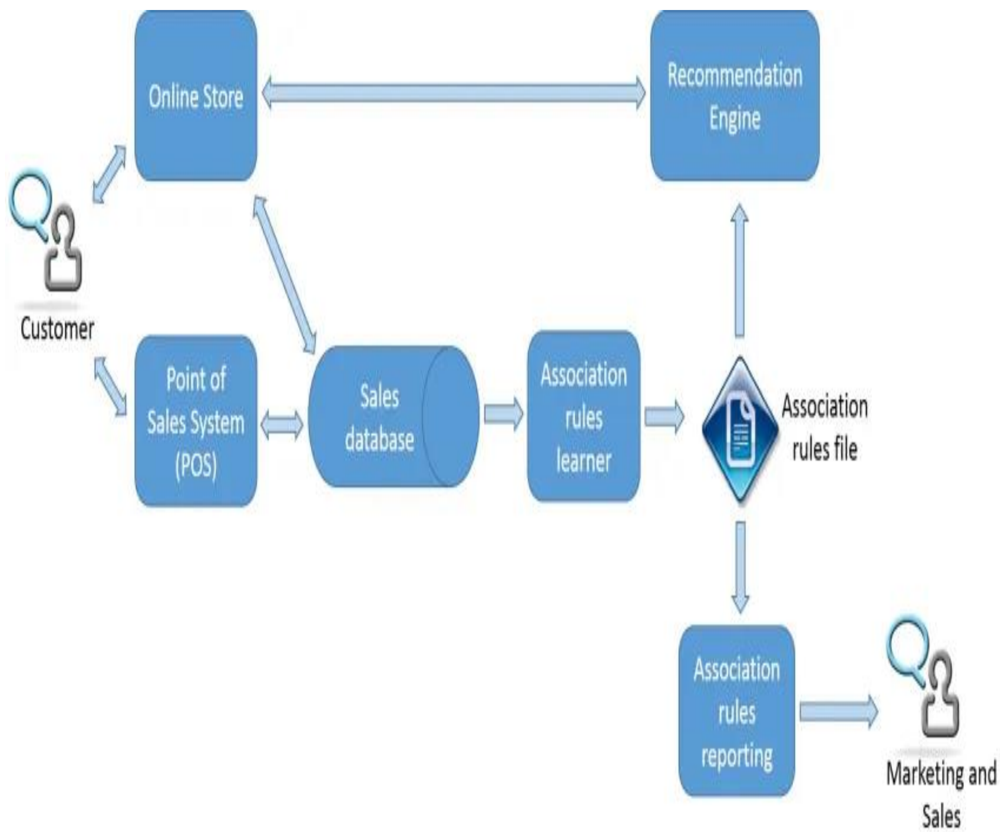
2.1 USE CASE DIAGRAM



2.2 DESIGN


Which items are frequently purchased together by customers?
Shopping Baskets
milk
bread
cereal
Customer 1
milk
bread
sugar
eggs
Customer 2
milk
bread
butter
Customer 3
sugar
eggs
Customer n
Market Analyst

2.3 SYSTEM ARCHITECTURE



CHAPTER 3

EXTERNAL INTERFACE REQUIREMENTS

3.1 REQUIREMENTS:

Here, the project deals with the Machine Learning concept. Here, it is better to use the Python programming language that facilitates the best utilization of the Machine Learning modules available in it. As described, here we use the Apriori module for the implementation. From this, Association rules are applied to the data set that is passed to the system. the first and foremost requirement for the project is that the dataset. The dataset enables us to test the efficiency of the project and makes us to utilize the project for the better use. There are some of the preliminary conditions to the dataset to be passed as the input to the system. They are, product id, product name and product purchased count. Similarly, the format of the output data set is that, the product id, product name, product purchased count and at last the product percent that contributes the most frequent one from the customers' basket. The other requirements are a scanner to read the dataset of purchased items, a system to enter or store the details of the products purchased from the shop and a system that contains this project to analyze the dataset i.e., purchased products history to obtain the result. The system should reach the minimum requirement that should be able to run the project successfully from it.

3.2 HARDWARE INTERFACE:

A barcode reader, also called a price scanner or point-of-sale (POS) scanner, is a hand-held or stationary input device used to capture and read information contained in a barcode. If the scanner is not used in the super market, the manual entry of the product details is utilized by the cashier. This enables the shop keeper to keep track of the purchase history of the customer. Then a system to evaluate the billing. After this, the scanned details are passed to the system to analyze the dataset to produce the most frequent purchased products.

3.3 SOFTWARE INTERFACE

Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Here, as stated Machine Learning concept is used i.e., Association rule from the Apriori module available from the Python Machine Learning library. It is used to make use of it to find the association between the products that facilitates us to find the most frequently purchased products by the customers from the Market.

CHAPTER 4

TESTING

4.1TEST PLAN:

The testing is done to determine whether the analysis system functions properly or not and to address the issues occur while functioning of the analysis system. If any issues found while testing, these are to be rectified and tested again whether it functions properly or not.

4.2TEST PROCEDURE:

- At first, the necessary modules are installed to the system to proceed with the project since, these modules play the back bone for the implemented project. Once all the required modules from the Python library is installed, it is tested that the project is suited to use these modules for the project.

- Once the modules are imported, the necessary functions are altered in such a way that it is used for the project. Then, the project is tested to rectify the utilities used from the modules.

- Then the data set is imported to the project, by using the Association rule from Apriori module of Machine Learning is applied to the project. Then it is tested that it shows the expected dataset i.e., the frequently purchased products by the customer from the shop.

4.3 TEST REPORT:

TEST ID	TEST NAME	EXPECTED RESULTS	ACTUAL RESULT	STATUS
1	Importing necessary modules	Imported successfully.	Expected result achieved	pass
2	Adding functionality from the modules	Successfully added.	Expected result achieved	pass
3	Dataset is converted to input form	Converted.	Expected result achieved	pass
4	Initializing the steps to read dataset.	Dataset is passed to the project.	Expected result achieved	pass
5	Steps to apply the Apriori module to dataset.	Success for above test Id and then applying the function.	Expected result achieved	pass
6	Overall testing	The project should produce the output dataset.	Run time error	fail
6	Overall testing	The project should produce the dataset.	Expected result achieved	pass

CHAPTER 5

FUTURE ENHANCEMENTS

In future, we can have the mobile application to the Market basket Analysis. It facilitates the new UI design to the market analyst. By doing this, we can have the interaction-based system for the application so that it makes easier to pass the dataset and obtain the relationships of frequently bought products. The interaction between the application and analyst i.e., user becomes the easy way to make use of the application. The UI facilitates the developer to express the application to the users in different way. For this, we need to have a framework that acts as an intermediate between the application and the user i.e., Market Analyst. This framework is designed in such way that it facilitates the above-mentioned enhancements.

CHAPTER 6

CONCLUSION

Here we conclude that, Our Market Basket Analysis system for Market Analyst facilitates us to analyze and predict the most frequently purchased products by the customer. As the analysis system using data mining of Machine learning, it is need not to monitor the application all the time since, it establishes the frequently bought products as the dataset automatically when the input dataset is passed to the application. It also shows that how market basket analysis also has its applicability in many domains and applying market basket analysis considering time as an important factor will be able to solve several problems in much effective and efficient way. We not only did comparison of association rules but also found top association rules during the 3 months of period which can help the company is inventory management as well as which categories of products should be kept close to each other to upscale the purchase. Mining into their data provides managers with a unique over view of what is happening with their business so that they can implement strategies efficiently and can move faster than their competitors.

REFERENCES

https://sci-hub.se/https://ieeexplore.ieee.org/document/9410737

https://www.techtarget.com/searchbusinessanalytics/definition/association-rules-in-data-mining

https://www.geeksforgeeks.org/implementing-apriori-algorithm-in-python

SAMPLE CODING

```
import pandas as p
from mlxtend.frequent_patterns import apriori, association_rules
d=p.read_csv('grocer.csv',sep='delimeter',engine='python',header=None)
a=[j.to_list() for (i,j) in d.iterrows()]
e=[]
m=[]
for i in a:
    for j in i:
        j=j.split(",")
        m.append(j)
        for k in j:
            e.append((k))
q={}
for i in e:
    if i not in q:
        q[i]=1
    else:
        q[i]+=1
z= sorted(q.items(), key=lambda x: x[1], reverse=True)
c=0
bb={}
for i in z:
    bb[i[0]]=i[1]
    c+=1
print(bb)
```

```
dict={"item_name":[i for i,j in bb.items()],"item_count":[j for i,j in
bb.items()]}
dd=p.DataFrame(dict)
print(type(e))
association_rules=(apriori(m,min_support=0.004,min_confidence=0.3,min_lift=3,min_
length=1.2))
association_results = list(association_rules)
print("There are {} Relation derived.".format(len(association_results)))
for item in association_results:
    pair = item[0]
    items = [x for x in pair]
    print("Rule: " + items[0] + " -> " + items[1])
    print("Support: " + str(item[1]))
    print("Confidence: " + str(item[2][0][2]))
    print("Lift: " + str(item[2][0][3]))
    print("=====================================")
dd['item_percent']=dd["item_count"]/len(e)
print(dd)
r=round(dd[:20]['item_percent'].sum()*100,2)
print(r)
```

SCREENSHOTS

{'mineral water': 1788, 'eggs': 1348, 'spaghetti': 1306, 'french fries': 1282, 'chocolate': 1230, 'green tea': 991, 'milk': 972, 'ground beef': 737, 'frozen vegetables': 715, 'pancakes': 713, 'burgers': 654, 'cake': 608, 'cookies': 603, 'escalope': 595, 'low fat yogurt': 574, 'shrimp': 536, 'tomatoes': 513, 'olive oil': 494, 'frozen smoothie': 475, 'turkey': 469, 'chicken': 450, 'whole wheat rice': 439, 'grated cheese': 393, 'cooking oil': 383, 'soup': 379, 'herb & pepper': 371, 'honey': 356, 'champagne': 351, 'fresh bread': 323, 'salmon': 319, 'brownies': 253, 'avocado': 250, 'hot dogs': 243, 'cottage cheese': 239, 'tomato juice': 228, 'butter': 226, 'whole wheat pasta': 221, 'red wine': 211, 'yogurt cake': 205, 'light mayo': 204, 'energy bar': 203, 'ham': 203, 'energy drink': 200, 'pepper': 199, 'vegetables mix': 193, 'cereals': 193, 'muffins': 181, 'oil': 173, 'french wine': 169, 'fresh tuna': 167, 'strawberries': 160, 'meatballs': 157, 'almonds': 153, 'parmesan cheese': 149, 'mushroom cream sauce': 143, 'rice': 141, 'protein bar': 139, 'mint': 131, 'white wine': 124, 'pasta': 118, 'light cream': 117, 'carrots': 115, 'black tea': 107, 'tomato sauce': 106, 'fromage blanc': 102, 'gums': 101, 'eggplant': 99, 'extra dark chocolate': 90, 'melons': 90, 'yams': 86, 'body spray': 86, 'magazines': 82, 'barbecue sauce': 81, 'cider': 79, 'nonfat milk': 78, 'candy bars': 73, 'zucchini': 71, 'whole weat flour': 70, 'salt': 69, 'blueberries': 69, 'green grapes': 68, 'flax seed': 68, 'antioxydant juice': 67, 'bug spray': 65, 'bacon': 65, 'green beans': 65, 'clothes accessories': 63, 'toothpaste': 61, 'shallot': 58, 'strong cheese': 58, 'spinach': 53, 'gluten free bar': 52, 'pet food': 49, 'sparkling water': 47, 'soda': 47, 'mayonnaise': 46, 'chili': 46, 'pickles': 45, 'burger sauce': 44, 'mint green tea': 42, 'hand protein bar': 39, 'salad': 37, 'shampoo': 37, 'corn': 36, 'cauliflower': 36, 'asparagus': 35, 'sandwich': 34, 'babies food': 34, 'dessert wine': 33, 'ketchup': 33, 'oatmeal': 33, 'chocolate bread': 32, 'chutney': 31, 'mashed potato': 31, 'tea': 29, 'bramble': 14, 'cream': 7, 'napkins': 5, 'water spray': 3, ' asparagus': 1}
<class 'list'>
There are 17 Relation derived.
Rule: escalope -> mushroom cream sauce
Support: 0.005732568990801226
Confidence: 0.3006993006993007
Lift: 3.790832696715049

```
========================================
Rule: pasta -> escalope
Support: 0.005865884548726837
Confidence: 0.3728813559322034
Lift: 4.700811850163794
========================================
Rule: ground beef -> herb & pepper
Support: 0.015997866951073192
Confidence: 0.3234501347708895
Lift: 3.2919938411349285
========================================
Rule: ground beef -> tomato sauce
Support: 0.005332622317024397
Confidence: 0.3773584905660377
Lift: 3.840659481324083
========================================
Rule: pasta -> shrimp
Support: 0.005065991201173177
Confidence: 0.3220338983050847
Lift: 4.506672147735896
========================================
Rule: ground beef -> cooking oil
Support: 0.004799360085321957
Confidence: 0.5714285714285714
Lift: 3.2819951870487856
========================================
Rule: ground beef -> eggs
Support: 0.0041327822956939075
Confidence: 0.3297872340425532
Lift: 3.3564912381997174
========================================
Rule: ground beef -> spaghetti
Support: 0.008665511265164644
Confidence: 0.31100478468899523
Lift: 3.165328208890303
========================================
Rule: olive oil -> milk
Support: 0.004799360085321957
Confidence: 0.4235294117647058
Lift: 3.2684095860566447
```

==
Rule: mineral water -> shrimp
Support: 0.007199040127982935
Confidence: 0.3050847457627118З
Lift: 3.200616332819722
==
Rule: tomatoes -> spaghetti
Support: 0.006665777896280496
Confidence: 0.3184713375796179
Lift: 3.341053850607991
==
Rule: ground beef -> grated cheese
Support: 0.005332622317024397
Confidence: 0.3225806451612903
Lift: 3.283144395325426
==
Rule: ground beef -> mineral water
Support: 0.006665777896280496
Confidence: 0.39062500000000006
Lift: 3.975682666214383
==
Rule: ground beef -> herb & pepper
Support: 0.006399146780429276
Confidence: 0.3934426229508197
Lift: 4.004359721511667
==
Rule: ground beef -> shrimp
Support: 0.005999200106652446
Confidence: 0.5232558139534884
Lift: 3.005315360233627
==
Rule: tomatoes -> spaghetti
Support: 0.004399413411545127
Confidence: 0.6111111111111112
Lift: 3.5099115194827295
==
Rule: ground beef -> mineral water
Support: 0.004399413411545127
Confidence: 0.3666666666666667
Lift: 3.7318407960199007

```
=========================================
        item_name  item_count  item_percent
0    mineral water        1788      0.060893
1             eggs        1348      0.045908
2        spaghetti        1306      0.044478
3     french fries        1282      0.043660
4        chocolate        1230      0.041889
..             ...         ...           ...
115        bramble          14      0.000477
116          cream           7      0.000238
117        napkins           5      0.000170
118    water spray           3      0.000102
119      asparagus           1      0.000034

[120 rows x 3 columns]
56.54

Process finished with exit code 0
```