

# **MARKET BASKET ANALYSIS USING APRIORI ALGORITHM**

A

Project Report

Submitted In Partial Fulfillment of the

Requirements For the Award Of  
**Bachelor of Technology**

Submitted By-

**SUVENDU SEKHAR ROY**

## **TABLE OF CONTENTS**

<b>S.NO.</b>	<b>Name of the Topic</b>	<b>Page No.</b>
<b>1.</b>	<b>Abstract</b>	<b>4</b>
<b>2.</b>	<b>Introduction</b>	<b>4</b>
<b>3.</b>	<b>Algorithms</b>	<b>5</b>
<b>3.1</b>	<b>Association Rule</b>	<b>5</b>
<b>3.2</b>	<b>Apriori Algorithm</b>	<b>5</b>
<b>4.</b>	<b>Problem Statement</b>	<b>6</b>
<b>5.</b>	<b>Objectives</b>	<b>6</b>
<b>6.</b>	<b>Background Dataset</b>	<b>6</b>
<b>7.</b>	<b>What is Machine Learning and AI</b>	<b>6</b>
<b>8.</b>	<b>System Analysis</b>	<b>7-8</b>
<b>8.1</b>	<b>Identification of Need</b>	<b>7</b>
<b>8.2</b>	<b>Feasibility Study</b>	<b>7</b>
<b>8.3</b>	<b>Work Flow</b>	<b>8</b>
<b>8.4</b>	<b>Hardware and Software Requirements</b>	<b>8</b>
<b>9.</b>	<b>System Design</b>	<b>7-</b>
<b>9.1</b>	<b>Gantt Chart</b>	<b>9</b>
<b>9.2</b>	<b>Sequence Diagram</b>	<b>10</b>
<b>9.3</b>	<b>Activity Diagram</b>	<b>11</b>
<b>9.4</b>	<b>Class Diagram</b>	<b>12</b>
<b>9.5</b>	<b>Use Case Diagram</b>	<b>13</b>
<b>10.</b>	<b>User Inreface Design</b>	<b>14</b>
	<b>Snapshots</b>	<b>14-17</b>

<b>11.</b>	<b>Implementation and Testing</b>	<b>18-</b>
<b>11.1</b>	<b>Introduction</b>	<b>18</b>
<b>11.2</b>	<b>Objective of Testing</b>	<b>18</b>
<b>11.3</b>	<b>Process Overview</b>	<b>18</b>
<b>11.4</b>	<b>Test Cases</b>	<b>19</b>
<b>11.5</b>	<b>Testing Step</b>	<b>20</b>
<b>11.6</b>	<b>Validation</b>	<b>21</b>
<b>12</b>	<b>Future Scope &amp; Further Enhancements</b>	<b>22</b>
<b>13.</b>	<b>Limitations</b>	<b>22</b>
<b>14.</b>	<b>Conclusion</b>	<b>22</b>
<b>15.</b>	<b>Bibliography/Reference</b>	<b>22</b>

## 1. ABSTRACT

The main objective of mining Data is to transform it into knowledge. This paper explores the basics of data mining with its architecture in brief. It also focuses on the patterns which can be mined so as to convert data in to knowledge. The topic elaborates one of the techniques to analyze the data from large amount of data sets. The analysis explained here is market basket analysis which is basically used by marketer to improve the performance of their business. It can be done by analyzing the available data in such way that frequent item set can be found and can be analyzed to define an association rule. One of the algorithm which helps in finding association rule for frequent item set is Apriori algorithm. It is explained at the end with a case study in this paper.

## 2. INTRODUCTION

Machine learning is helping the retail industry in many ways. From forecasting the sales performance to identifying the prospective buyers, there are a lot of applications of machine learning in the retail industry. Market basket analysis is one of the key applications of machine learning in retail. By analysing the past buying behaviour of customers, one can find out which are the products that are bought together by the customers. For example, bread and butter are sold together, baby diapers and baby massage oil are sold together, etc. That means one can analyze the association among products. If the retails management can find this association, while placing the products in the shop, these associated products can be put together. Or, when seeing that a customer is buying a product, the salesman can offer the associated product to the customer.

This process of analyzing the association is called the Association Rule Learning and analyzing the products bought together by the customers is called the Market Basket Analysis. In this article, we will discuss the association rule learning method with a practical implementation of market basket analysis in python. We will use the Apriori algorithm as an association rule method for market basket analysis.

### 3. ALGORITHMS

For our project, we focused on two main algorithms for recommendations: **Association Rule Mining & Apriori Algorithm**

#### 3.1 Association Rule Mining:

Association Rule Mining is used when you want to find an association between different objects in a set, find frequent patterns in a transaction database, relational databases or any other information repository. The applications of Association Rule Mining are found in Marketing, Basket Data Analysis (or Market Basket Analysis) in retailing, clustering and classification.

Association Rule-based algorithms are viewed as a two-step approach:

**1.Frequent Itemset Generation:** Find all frequent item-sets with support  $\geq$  pre-determined min\_support count

**2.Rule Generation:** List all Association Rules from frequent item-sets. Calculate Support and Confidence for all rules. Prune rules that fail min\_support and min\_confidence thresholds.

#### 3.2 Apriori Algorithm:

Apriori algorithm assumes that any subset of a frequent itemset must be frequent. It's the algorithm behind Market Basket Analysis.

Say, a transaction containing {Grapes, Apple, Mango} also contains {Grapes, Mango}. So, according to the principle of Apriori, if {Grapes, Apple, Mango} is frequent, then {Grapes, Mango} must also be frequent. In order to find out interesting rules out of multiple possible rules from this small business scenario, we will be using the following matrices:

**1.Support:** Its the default popularity of an item. In mathematical terms, the support of item **A** is nothing but the ratio of transactions involving **A** to the total number of transactions.

$$\text{Support}(\mathbf{A}) = \frac{(\text{Transactions involving } \mathbf{A})}{(\text{Total transaction})}$$

**2. Confidence:** Likelihood that customer who bought both **A** and **B**. Its divides the number of transactions involving both **A** and **B** by the number of transactions involving **B**.

$$\text{Confidence } (\mathbf{A} \Rightarrow \mathbf{B}) = \frac{(\text{Transactions involving both } \mathbf{A} \text{ and } \mathbf{B})}{(\text{Transaction involving only } \mathbf{A})}$$

**3. Lift:** Increase in the sale of **A** when you sell **B**.

$$\text{Lift } (\mathbf{A} \Rightarrow \mathbf{B}) = \frac{\text{Confidence } (\mathbf{A}, \mathbf{B})}{\text{Support } (\mathbf{B})}$$

So, likelihood of a customer buying both **A** and **B** together is 'lift-value' times more than the chance if purchasing alone.

- **Lift** ( $\mathbf{A} \Rightarrow \mathbf{B}$ ) = 1 means that there is no correlation within the itemset.
- **Lift** ( $\mathbf{A} \Rightarrow \mathbf{B}$ ) > 1 means that there is a positive correlation within the itemset, i.e., products in the itemset, **A**, and **B**, are more likely to be bought together.
- **Lift** ( $\mathbf{A} \Rightarrow \mathbf{B}$ ) < 1 means that there is a negative correlation within the itemset, i.e., products in itemset, **A**, and **B**, are unlikely to be bought together.

## 4. PROBLEM STATEMENT

Through in depth research and observations carried on supermarket we have discovered that retailers are willing to know what product is purchased with the other or if a particular products are purchased together as a group of items. Which can help in their decision making with respect to placement of product , determining the timing and extent of promotions on product and also have a better understanding of customer purchasing habits by grouping customers with their transactions.

This project is aimed at designing and implementing a well-structured market basket analysis software tool to solve the problem stated above.

## 5. Objectives:

- a) To identify the frequent items from the transaction on the basis of support and confidence
- b) To generate the association rule from the frequent item sets.

## 6. BACKGROUND DATASET

In this implementation, we have used the Market Basket Optimization dataset that is publicly available on [Kaggle](#). This dataset comprises the list of transactions of a retail company over the period of one week. It contains a total of 7501 transaction records where each record consists of the list of items sold in one transaction. Using this record of transactions and items in each transaction, we will find the association rules between items.

## 7. WHAT IS ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING?

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves.

The process of learning begins with observations or data, such as examples, direct experience, or instruction, in order to look for patterns in data and make better decisions in the future based on the examples that we provide. The primary aim is to allow the computers learn automatically without human intervention or assistance and adjust actions accordingly.

The word Artificial Intelligence comprises of two words “Artificial” and “Intelligence”. Artificial refers to something which is made by human or non natural thing and Intelligence means ability to understand or think. There is a misconception that Artificial Intelligence is a system, but it is not a system .AI is implemented in the system. There can be so many definition of AI, one definition can be “It is the study of how to train the computers so that computers can do things which at present human can do better. Therefore It is a intelligence where we want to add all the capabilities to machine that human contain.

## 8. SYSTEM ANALYSIS

### 8.1 IDENTIFICATION OF NEED

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minutest detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organization are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analysis and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is a loop that ends as soon as the user is satisfied with the proposal.

### 8.2 FEASIBILITY STUDY

Feasibility study is made to see if the project on completion will serve the purpose the organization for the amount of work

Effort and time spent on it : Feasibility study lets the developer foresee the future of the project and the usefulness. A feasibility study of a system proposal is according to its workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus when a new application is proposed it normally goes through a feasibility study before it is approved for development.

The document provides the feasibility of the project that is being designed and lists various areas that were considered very carefully during the feasibility study of this project such as Technical, Economic and operational feasibilities.

**Technical Feasibility:** This project is technically feasible as all it has got to do to extract tweets, is to get the proper credentials from the Developer's console provided by Twitter. After the credentials are obtained, i.e. the Access Token Key, Access Token Secret Key, Consumer Key and Consumer Secret key, Twitter gives us access to its tweets. Hence we get a sufficiently large dataset to conduct sentiment analysis. Also the range of tweets obtained is limited to 300 tweets per page, which ensures that the results do not go out of bounds. Thus, this is technically feasible.

**Economic Feasibility:** This project work is economically feasible as it does not take into account any additional costs. Whatever data is extracted, it is done without any charges. Twitter provides free use of this data that is non-encrypted and publicly available for analysis purpose. Hence, this work is economically feasible as well.

**Operational Feasibility :** This is operationally feasible as well. As already mentioned, it takes in 300 tweets per page as that is the limit set by Twitter. Therefore it is operationally feasible as well. The system won't hang when getting the results.

## **8.3 WORK FLOW**

The steps of working of the apriori algorithm can be given as:-

1. Define the minimum support and confidence for the association rule
2. Take all the subsets in the transactions with higher support than the minimum support
3. Take all the rules of these subsets with higher confidence than minimum confidence
4. Sort the association rules in the decreasing order of lift.
5. Visualize the rules along with confidence and support.

## **8.4 HARDWARE AND SOFTWARE REQUIREMENTS**

### **➤ Hardware Requirements**

- Standard computer with at least i3 processor
- Standard computer with 2GB of RAM
- Standard computer with 100GB of free space

### **➤ Software Requirements**

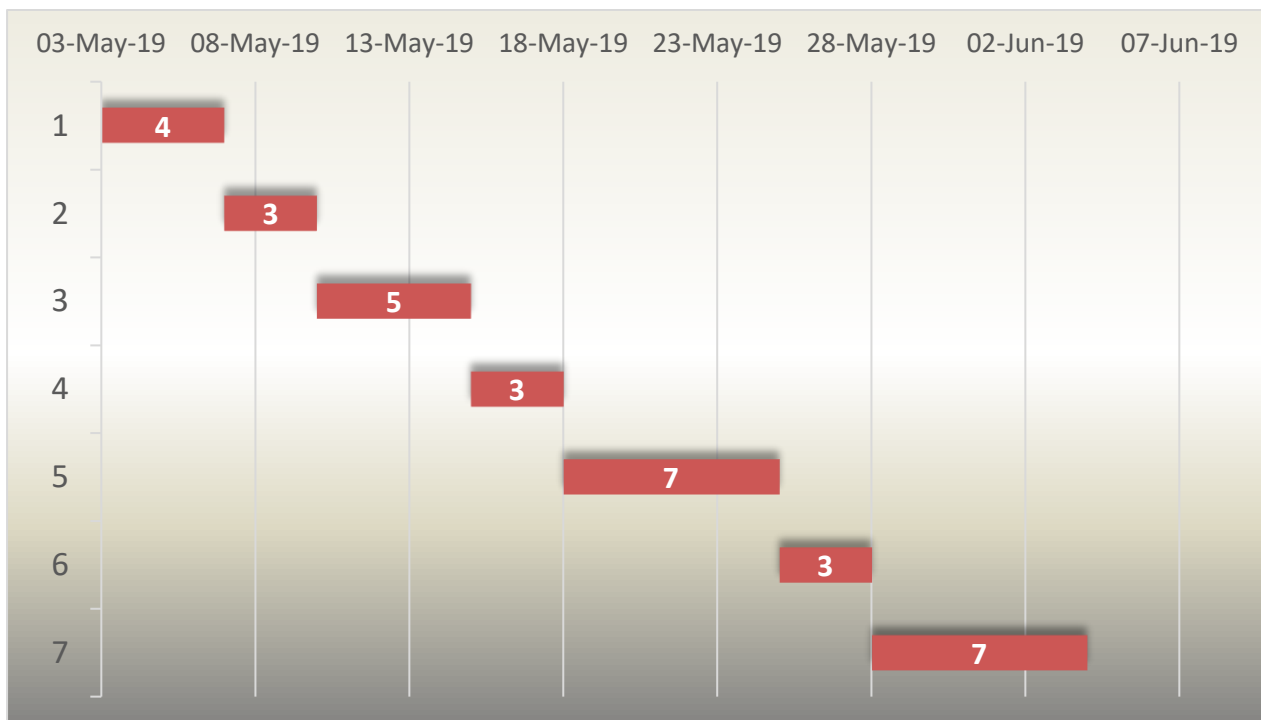
- python 3.7
- Anaconda-3.5
- Ms Office
- Notepad++



## 9. SYSTEM DESIGN

### 9.1 GANTT CHART

SI No.	TASK	START DATE	END DATE	DURATION
1	FEASIBILITY STUDY	3-May-19	6-May-19	4
2	REVIEW PREVIOUS WORKS	7-May-19	9-May-19	3
3	PLANNING	10-May-19	14-May-19	5
4	DESIGN	15-May-19	17-May-19	3
5	CODING	18-May-19	24-May-19	7
6	TESTING	25-May-19	27-May-19	3
7	DOCUMENTATION	28-May-19	3-Jun-19	7



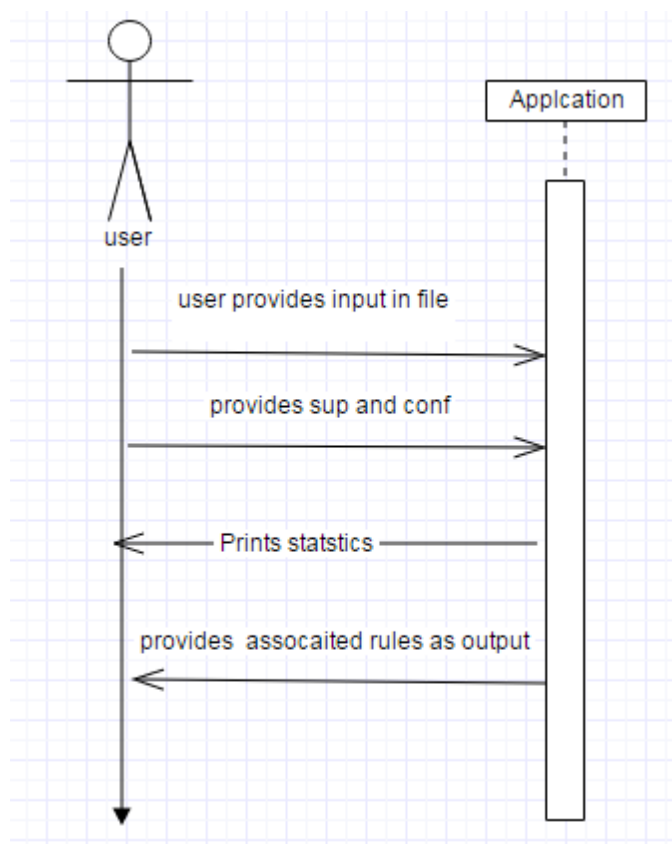
## GANTT CHART

## 9.2 SEQUENCE DIAGRAM

Sequence diagrams can be useful reference diagrams for businesses and other organizations.

Try drawing a sequence diagram to:

- Represent the details of a UML use case.
- Model the logic of a sophisticated procedure, function, or operation.
- See how tasks are moved between objects or components of a process.
- Plan and understand the detailed functionality of an existing or future scenario.



**Fig:Sequence Diagram**

This figure shows the sequence diagram of this application.

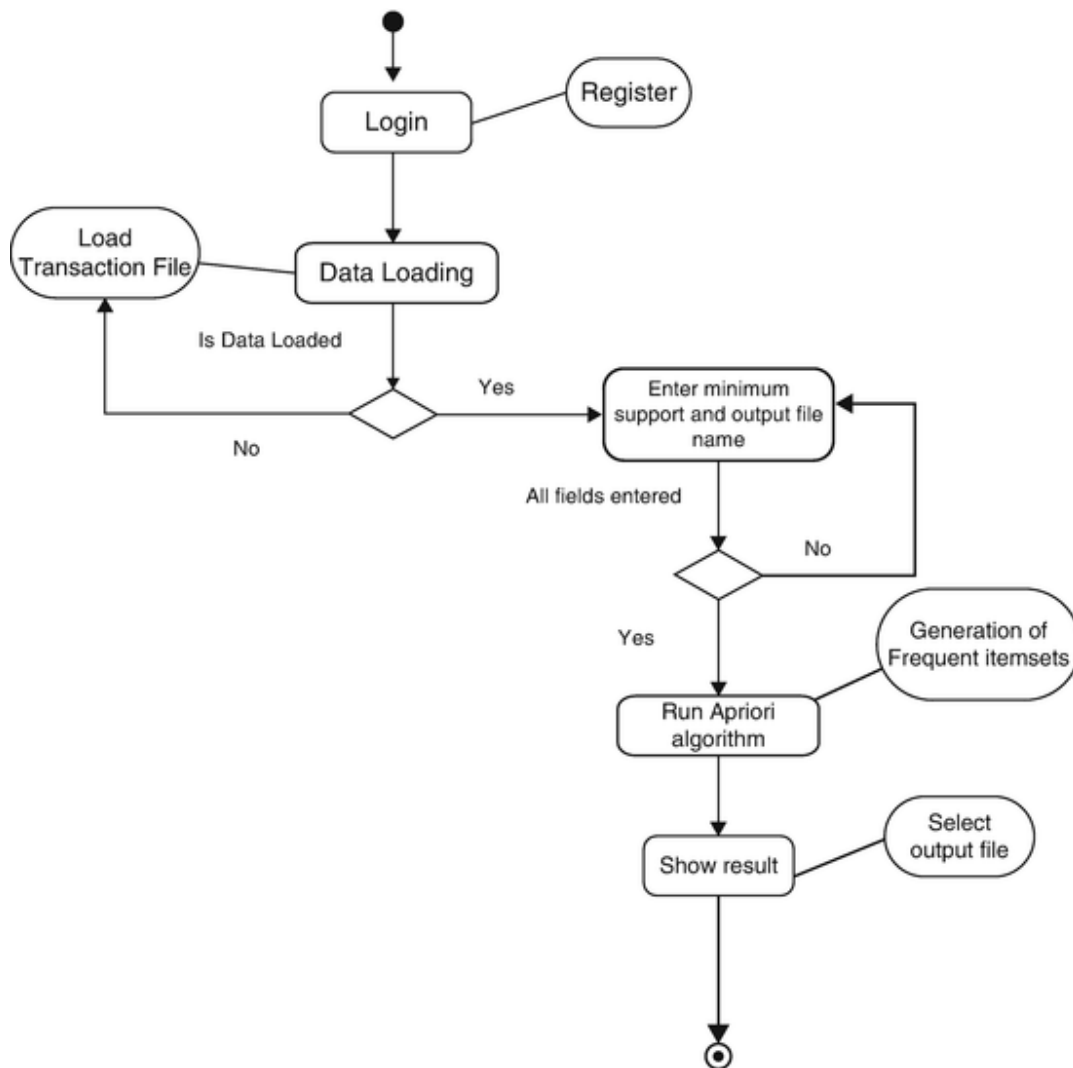
The user needs to choose the input file that is going to be processed. The file should contain the data in integer where the row represents the items and column represents transactions. Confidence and support should be provided by the user. After all the input is given the application process the data and provide the output to the user.

## 9.3 ACTIVITY DIAGRAM

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

### Purpose of Activity Diagrams:

The basic purposes of activity diagrams is similar to other four diagrams. It captures the dynamic behavior of the system. Other four diagrams are used to show the message flow from one object to another but activity diagram is used to show message flow from one activity to another.



**Fig:Activity Diagram**

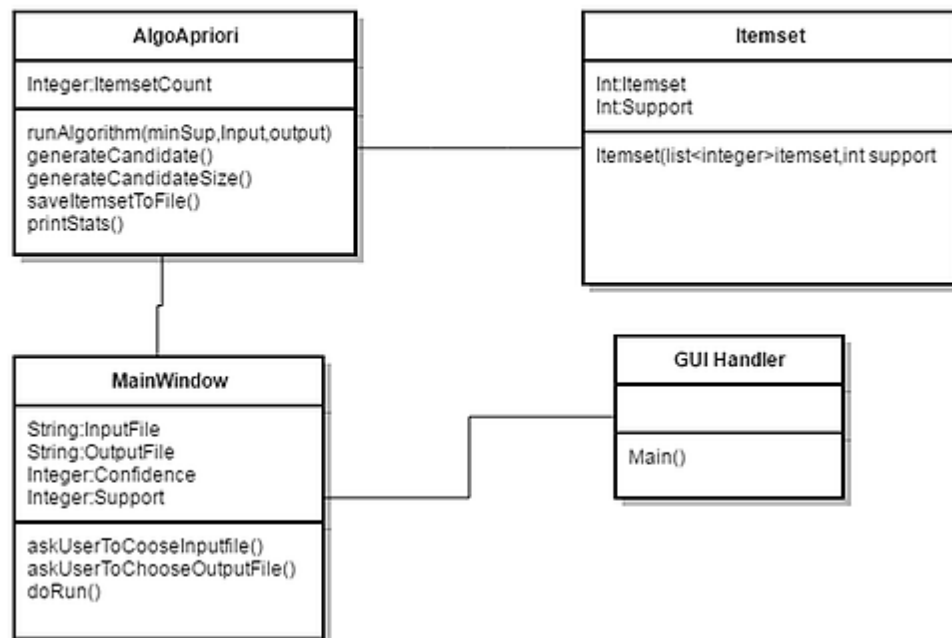
## 9.4. CLASS DIAGRAM

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

### Purpose of Class Diagrams

- Class diagram is a static diagram and it is used to model the static view of a. The static view describes the vocabulary of the system.
- Class diagram is also considered as the foundation for component and deployment diagrams. Class diagrams are not only used to visualize the system static view of the system but they are also used to construct the executable code for forward and reverse engineering of any system.
- Generally, UML diagrams are not directly mapped with any object-oriented programming languages but the class diagram is an exception.
- Class diagram clearly shows the mapping with object-oriented languages such as Java, C++, etc. From practical experience, class diagram is generally used for construction purpose.



**Fig: Class Diagram**

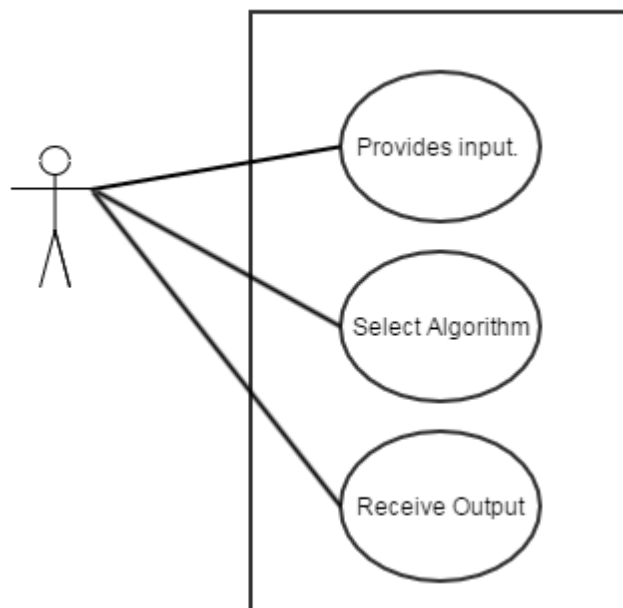
As shown in Figure, there are three main classes used in the application

The main Window class is used to present the user interface for choosing the input file and output file as desired by the user.

The Algo Apriori is the class that performs all the calculations once the data is provided by the user. It generates the candidate item sets and determines the size of the item sets. Finally the statistics are provide to the user in the same GUI and output is written to the desired file.

The item set class stores the items as the array of integer and provides the support of the respective item from the given input data.

## 9.5. USE CASE DIAGRAM:



**Fig: Use Case Diagram**

The user will provide input to the application. The input is a text file where items are mapped into integer's value. The row represents the items that were purchased in one single transaction and column represents the transaction.

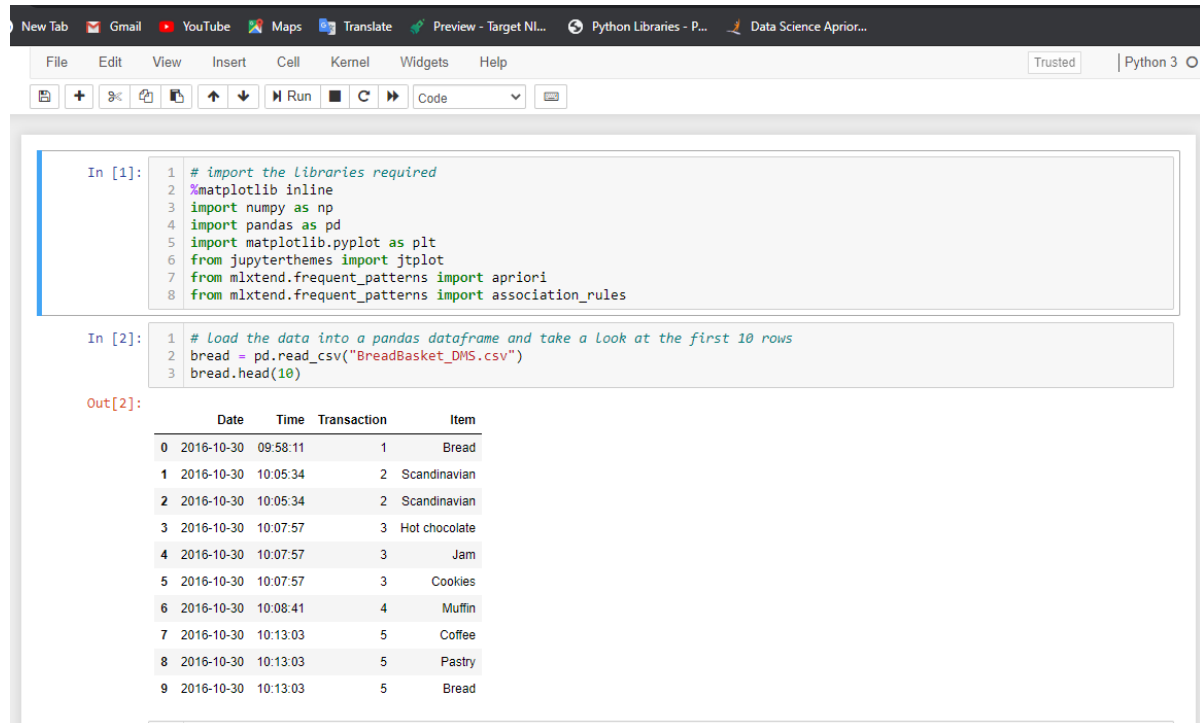
The user then can select the algorithm to run and provide the necessary parameters i.e. confidence and support.

After processing the user will receive output in to the desired path that the user wants. The output will be a text file containing association rules.

## 10. USER INTERFACE DESIGN

1. The design of user interfaces for machines and software, such as computers, home appliances, mobile devices, and other electronic devices, with the focus on maximizing the user experience. The goal of user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals (user-centered design).
2. Good user interface design facilitates finishing the task at hand without drawing unnecessary attention to it. Graphic design and typography are utilized to support its usability, influencing how the user performs certain interactions and improving the aesthetic appeal of the design; design aesthetics may enhance or detract from the ability of users to use the functions of the interface. The design process must balance technical functionality and visual elements (e.g., mental model) to create a system that is not only operational but also usable and adaptable to changing user needs.

### SNAPSHOTS:



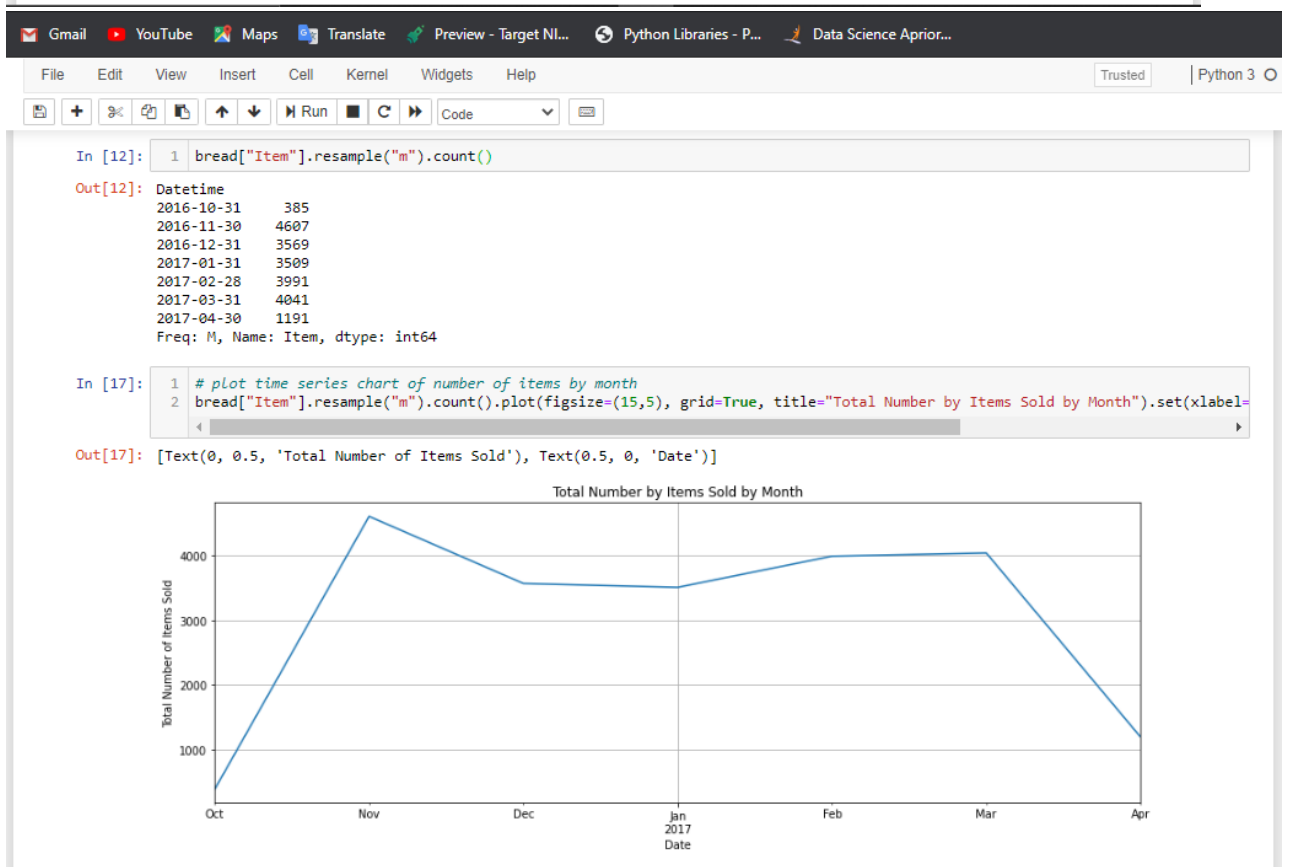
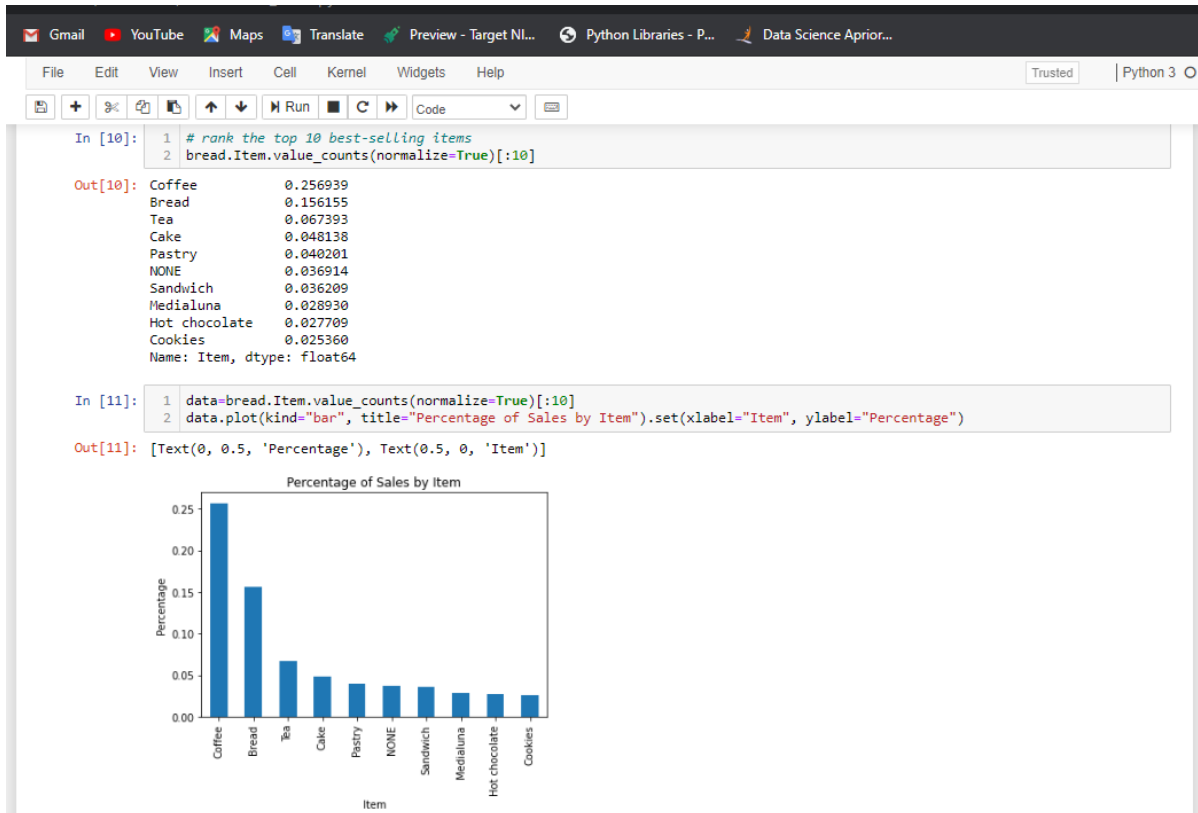
The screenshot shows a Jupyter Notebook interface with a dark theme. The top bar includes tabs for 'New Tab', 'Gmail', 'YouTube', 'Maps', 'Translate', 'Preview - Target NL...', 'Python Libraries - P...', and 'Data Science Aprior...'. Below the tabs is a menu bar with 'File', 'Edit', 'View', 'Insert', 'Cell', 'Kernel', 'Widgets', and 'Help'. A toolbar contains icons for file operations, a 'Run' button, and a 'Code' dropdown menu. The notebook area displays two input cells and one output cell.

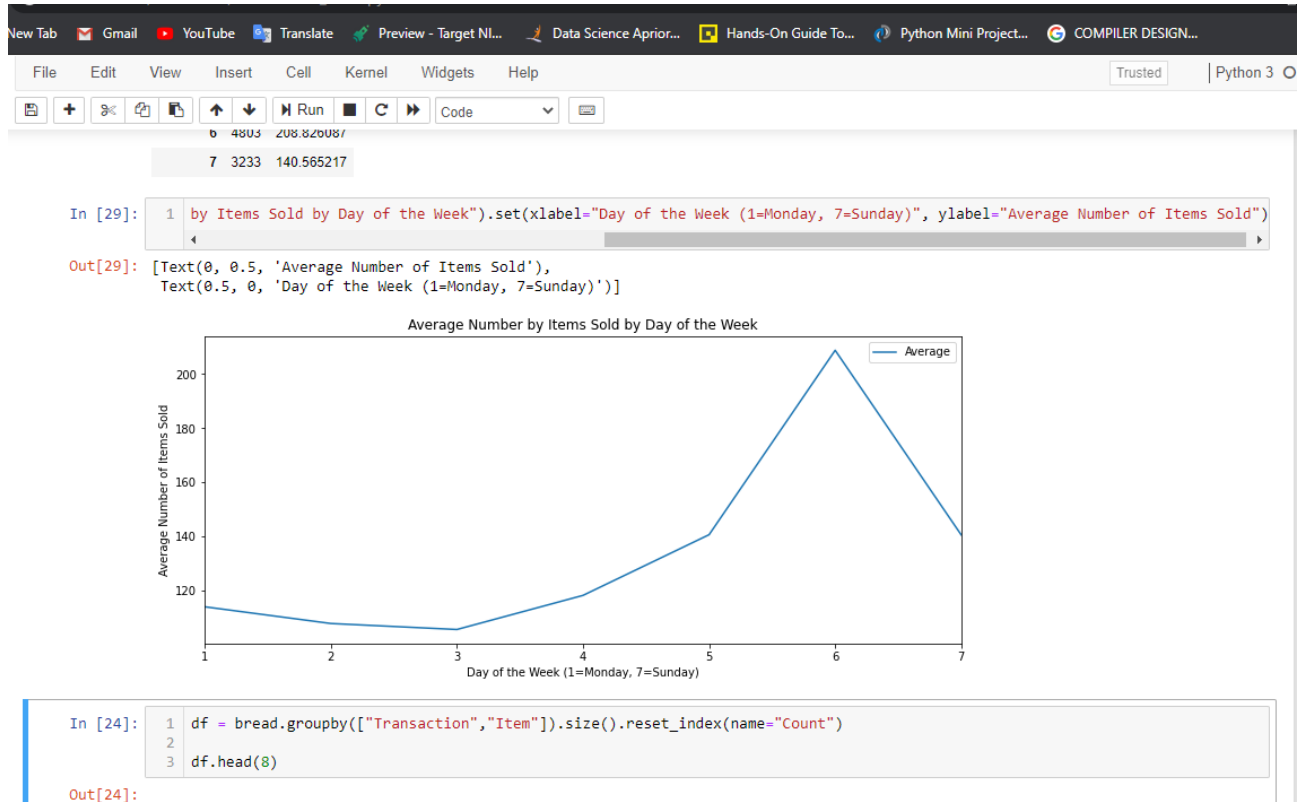
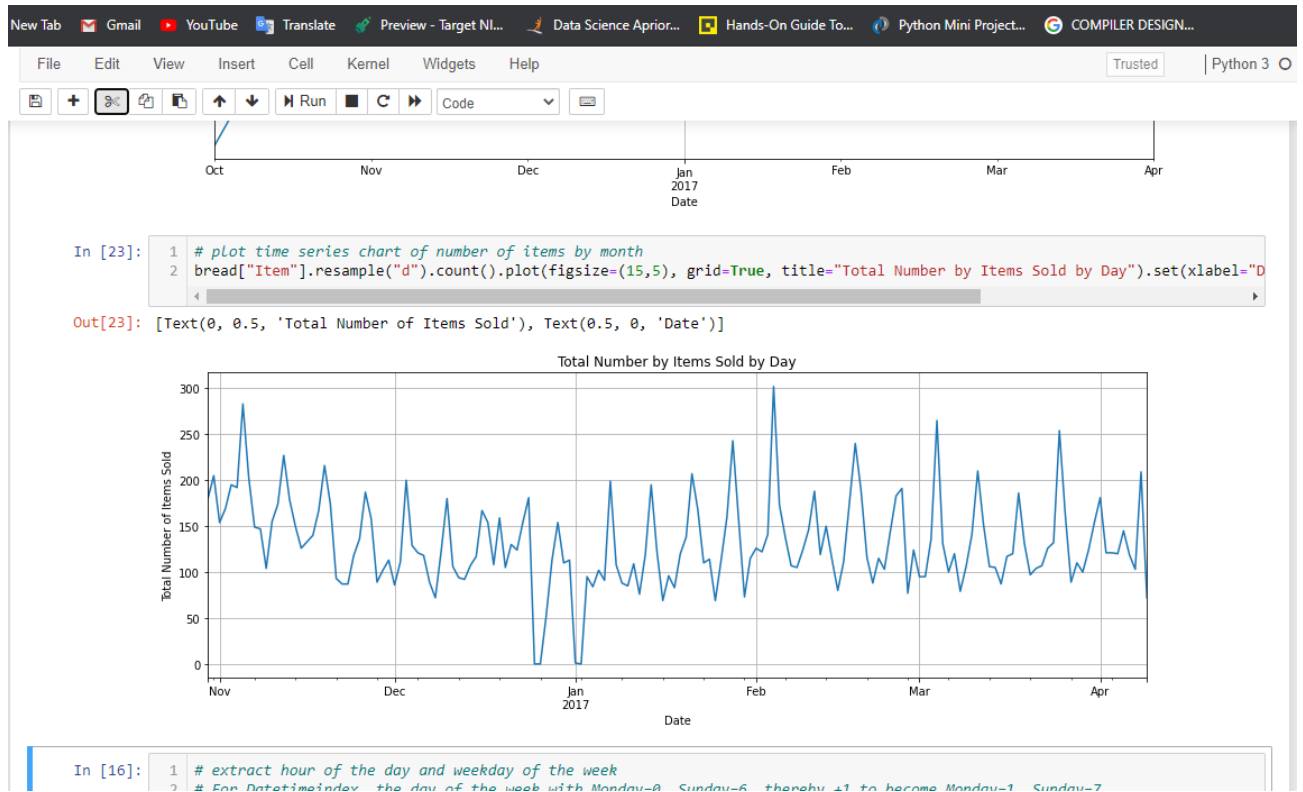
```
In [1]: 1 # import the libraries required
        2 %matplotlib inline
        3 import numpy as np
        4 import pandas as pd
        5 import matplotlib.pyplot as plt
        6 from jupyterthemes import jtplot
        7 from mlxtend.frequent_patterns import apriori
        8 from mlxtend.frequent_patterns import association_rules
```

```
In [2]: 1 # Load the data into a pandas dataframe and take a look at the first 10 rows
        2 bread = pd.read_csv("BreadBasket_DMS.csv")
        3 bread.head(10)
```

Out[2]:

	Date	Time	Transaction	Item
0	2016-10-30	09:58:11	1	Bread
1	2016-10-30	10:05:34	2	Scandinavian
2	2016-10-30	10:05:34	2	Scandinavian
3	2016-10-30	10:07:57	3	Hot chocolate
4	2016-10-30	10:07:57	3	Jam
5	2016-10-30	10:07:57	3	Cookies
6	2016-10-30	10:08:41	4	Muffin
7	2016-10-30	10:13:03	5	Coffee
8	2016-10-30	10:13:03	5	Pastry
9	2016-10-30	10:13:03	5	Bread







Gmail YouTube Maps Translate Preview - Target NL... Python Libraries - P... Data Science Aprior...

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

In [53]:

```
1 rules = association_rules(frequent_itemsets, metric="lift", min_threshold=1) #model
2 rules.sort_values("confidence", ascending = False, inplace = True)
3 rules.head(20)
```

Out[53]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
32	(Toast)	(Coffee)	0.033365	0.475081	0.023502	0.704403	1.482699	0.007651	1.775789
31	(Spanish Brunch)	(Coffee)	0.018046	0.475081	0.010807	0.598837	1.260494	0.002233	1.308493
18	(Medialuna)	(Coffee)	0.061379	0.475081	0.034939	0.569231	1.198175	0.005779	1.218561
25	(Pastry)	(Coffee)	0.085510	0.475081	0.047214	0.552147	1.162216	0.006590	1.172079
1	(Alfajores)	(Coffee)	0.036093	0.475081	0.019515	0.540698	1.138116	0.002368	1.142861
17	(Juice)	(Coffee)	0.038296	0.475081	0.020460	0.534247	1.124537	0.002266	1.127031
22	(NONE)	(Coffee)	0.079005	0.475081	0.042073	0.532537	1.120938	0.004539	1.122908
26	(Sandwich)	(Coffee)	0.071346	0.475081	0.037981	0.532353	1.120551	0.004086	1.122468
7	(Cake)	(Coffee)	0.103137	0.475081	0.054349	0.526958	1.109196	0.005350	1.109667
29	(Scone)	(Coffee)	0.034309	0.475081	0.017941	0.522936	1.100729	0.001642	1.100310
13	(Cookies)	(Coffee)	0.054034	0.475081	0.028014	0.518447	1.091280	0.002343	1.090053
15	(Hot chocolate)	(Coffee)	0.057916	0.475081	0.029378	0.507246	1.067704	0.001863	1.065276
5	(Brownie)	(Coffee)	0.039765	0.475081	0.019515	0.490765	1.033013	0.000624	1.030799
20	(Muffin)	(Coffee)	0.038191	0.475081	0.018676	0.489011	1.029321	0.000532	1.027260
3	(Pastry)	(Bread)	0.085510	0.324940	0.028958	0.338650	1.042194	0.001172	1.020731
34	(NONE)	(Tea)	0.079005	0.141643	0.018256	0.231076	1.631394	0.007066	1.116309
11	(Cake)	(Tea)	0.103137	0.141643	0.023607	0.228891	1.615972	0.008999	1.113146
36	(Sandwich)	(Tea)	0.071346	0.141643	0.014269	0.200000	1.412000	0.004164	1.072946
8	(Hot chocolate)	(Cake)	0.057916	0.103137	0.011331	0.195852	1.897010	0.005358	1.115019
10	(Tea)	(Cake)	0.141643	0.103137	0.023607	0.166667	1.615972	0.008999	1.076235

In [ ]: 1

## **11. IMPLEMENTATION AND TESTING**

A software system test plan is a document that describes the objectives, scope, approach and focus of software testing effort. The process of preparing a test plan is a usual way to think the efforts needed to validate the acceptability of a software product. The complete document will help people outside the test group understand the "WHY" and "HOW" product validation. It should be thorough enough to be useful but not so thorough that no one outside the test group will read it.

### **11.1 Introduction**

Testing is the process of running a system with the intention of finding errors. Testing enhances the integrity of a system by detecting deviations in design and errors in the system. Testing aims at detecting error-prone areas. This helps in the prevention of errors in a system. Testing also adds value to the product by conforming to the user requirements.

The main purpose of testing is to detect errors and error prone areas in a system. Testing must be through well planned. A partially tested system is to detect errors and error prone areas in a system. Testing must be through well planned. A partially tested system is as bad as an untested system. And the price of an untested and under tested system is high.

### **11.2 Objectives Of Testing**

The objective our test plan is to find and report as many bugs as possible to improve the integrity of our program. Although exhaustive testing is not possible, we will exercise a broad range of tests to achieve our goal. Our user interface to utilize these functions is designed to be user-friendly and provide easy manipulation of the tree. The application will only be used as a demonstration tool, but we would like to ensure that it could be run from a variety of platforms with little impact on performance or usability.

### **11.3 Process Overview**

The following represents the overall flow of the testing process:

- Identify the requirements to be tested. All test cases shall be derived using the current Program Specification.
- Identify which particular test(s) will be used to test each module.
- Review the test data and test cases to ensure that the unit has been thoroughly verified and that the test data and test cases are adequate to verify proper operation of the unit.

## 11.4 Test Cases

A test case is a document that describe an input, action, or event and expected response, to determine if a feature of an application is working correctly. A test case should contain particular such as test case identifier, test condition, input data.

Requirement expected results. The process of developing test cases can help find problems in the requirements or design of an application since it requires completely thinking through the operations of the application.

Transaction	Item
1	Bread
2	Scandinavian
2	Scandinavian
3	Hot chocolate
3	Jam
3	Cookies
4	Muffin
5	Coffee
5	Pastry
5	Bread
6	Medialuna
6	Pastry
6	Muffin
7	Medialuna
7	Pastry
7	Coffee
7	Tea
8	Pastry
8	Bread
9	Bread
9	Muffin
10	Scandinavian
10	Medialuna

ID	Bread	Scandinavian	Hot-Chocolate	Jam	Cookies	Muffin	Coffee	Pastry	Tea	Medialuna
1	1	-	-	-	-	-	-	-	-	-
2	-	1	-	-	-	-	-	-	-	-
3	-	-	1	1	1	-	-	-	-	-
4	-	-	-	-	-	1	-	-	-	-
5	1	-	-	-	-	-	1	1	-	-
6	-	-	-	-	-	1	-	1	-	1
7	-	-	-	-	-	-	1	1	1	1
8	1	-	-	-	-	-	-	1	-	-
9	1	-	-	-	-	1	-	-	-	-
10	-	1	-	-	-	-	-	-	-	1

$$\text{Support of bread} = \frac{\text{Transactions involving bread}}{\text{Total transaction}} = \frac{4}{10} = 0.4$$

$$\text{Confidence (bread} \Rightarrow \text{pastry)} = \frac{\text{Transactions involving bread and pastry}}{\text{Transactions involving only bread}} = \frac{2/10}{4/10} = 0.5$$

$$\text{lift (bread} \Rightarrow \text{pastry)} = \frac{\text{Confidence (bread} \Rightarrow \text{pastry)}}{\text{Support of pastry}} = \frac{2/4}{4/10} = 1.25$$

Here lift (bread $\Rightarrow$ pastry) is 1.25(>1). So, bread and pastry can be associate with each other.

## 11.5 Testing Steps

### ● Unit Testing

Unit testing focuses efforts on the smallest unit of software design. This is known as module testing. The modules are tested separately. The test is carried out during programming stage itself. In this step, each module is found to be working satisfactory as regards to the expected output from the module.

### ● Integration Testing

Data can be lost across an interface. One module can have an adverse effect on another, sub functions, when combined, may not be linked in desired manner in major functions. Integration testing is a systematic approach for constructing the program structure, while at the same time conducting test to uncover errors associated within the interface.

## 11.6 Validation

At the culmination of the integration testing, Software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of software test begin in validation testing. Validation testing can be defined in many ways, but a simple definition is that the validation succeeds when the software functions in a manner that is expected by the customer. After validation test has been conducted, one of the three possible conditions exists.

- a) The function or performance characteristics confirm to specification and are accepted.
- b) A deviation from specification is uncovered and a deficiency lists is created.
- c) Proposed system under consideration has been tested by using validation test and found to be working satisfactory

Tested By:		SUVENDU SEKHAR ROY
Test Type		Unit Testing
Test CaseNumber		
TestCaseName		Market Basket Analysis
Test Case Description		The user should enter name of the products. The system checks to which products are associate with each other. The system after the analysis the top 20 products, of the genres based on the top ratings.
Item(s) to be tested		
	Verification of recommendation provided by the system.	
Specifications		
Input		Expected Output/Result
Enter the name of a particular item		Output contain most associated item

## 12. FUTURE SCOPE AND FURTHER ENHANCEMENTS

**Market basket analysis** is one of the best possible ways to find out which items can be bought together with another item. **Market Basket Analysis** helps us to identify the purchasing behavior of the customer. By mining the data from the huge transaction database shop managers can study the behavior or buying habits of the customer to increase the sale. These proposed techniques are much useful in both supermarket and agriculture.

The future work would be some modification of other algorithms to generate the association rules that can be adopted on existing recommendation system to make them functionally more effective. Better and effective rule mining techniques can be used for better performance of the recommendation system.

## 13. Limitations

- a. The application will be desktop and will not be available online.
- b. Input to the application will be a file which contains integer values representing the list of items, the integer values will be mapped manually.

## 14. CONCLUSION:

The Apriori algorithm effectively generates highly informative frequent item sets and association rules for the data of the supermarket. The frequent data items are generated from the given input data and based on the frequent item sets strong association rules were generated. The input data given to the application is used as the integer value mapped from the transaction database. The mapping is done manually. If database converter is made then the system will work effectively for any format of data. The application can be efficiently used by using more efficient algorithm rather than Apriori Algorithm in future.

## 15. BIBLIOGRAPHY

- I. [intellipaat.com/blog/data-science-apriori-algorithm/#What-Is-an-Apriori-Algorithm](http://intellipaat.com/blog/data-science-apriori-algorithm/#What-Is-an-Apriori-Algorithm)
- II. pandas: powerful Python data analysis toolkit (Release 1.0.3) by Wes McKinney and the Pandas Development Team
- III. Python Tutorial (Release 3.5.2) by Guido van Rossum and the Python development team
- IV. [https://en.wikipedia.org/wiki/Apriori\\_algorithm#:~:text=Apriori%20is%20an%20algorithm%20for,sufficiently%20often%20in%20the%20database.](https://en.wikipedia.org/wiki/Apriori_algorithm#:~:text=Apriori%20is%20an%20algorithm%20for,sufficiently%20often%20in%20the%20database.)
- V. <https://www.youtube.com/watch?v=-oD50HSBBBI>
- VI. <https://images.app.goo.gl/wCxm9rUGrBDWaDqS9>