**MARKET BASKETS AND INSIGHTS:**

| **NAME : Prakash R NM ID : au513521106026**  **COLLAGE: AMCET** |
| --- |

**Phase 3**

**1 Introduction**

Hi! In this kernel we are going to use the Apriori algorithm to perform a Market Basket Analysis. A Market what? Is a technique used by large retailers to uncover associations between items. It works by looking for combinations of items that occur together frequently in transactions, providing information to understand the purchase behavior. The outcome of this type of technique is, in simple terms, a set of rules that can be understood as “if this, then that”. For more information about these topics, please check in the following links:

Market Basket Analysis

Apriori algorithm

Association rule learning

First it’s important to define the Apriori algorithm, including some statistical concepts (support, confidence, lift and conviction) to select interesting rules. Then we are going to use a data set containing more than 6.000 transactions from a bakery to apply the algorithm and find combinations of products that are bought together. Let’s start!

**2 Association rules**

The Apriori algorithm generates association rules for a given data set. An association rule implies that if an item A occurs, then item B also occurs with a certain probability. Let’s see an example,

Transaction Items

t1 {T-shirt, Trousers, Belt}

t2 {T-shirt, Jacket}

t3 {Jacket, Gloves}

t4 {T-shirt, Trousers, Jacket}

t5 {T-shirt, Trousers, Sneakers, Jacket, Belt}

t6 {Trousers, Sneakers, Belt}

t7 {Trousers, Belt, Sneakers}

**2.1 Support**

Support is an indication of how frequently the item set appears in the data set.

supp(X⇒Y)=|X∪Y|n

In other words, it’s the number of transactions with both X

and Y

divided by the total number of transactions. The rules are not useful for low support values. Let’s see different examples using the clothing store transactions from the previous table.

supp(T-shirt⇒Trousers)=37=43%

supp(Trousers⇒Belt)=47=57%

supp(T-shirt⇒Belt)=27=28%

supp({T-shirt,Trousers}⇒{Belt})=27=28%

**2.2 Confidence**

For a rule X⇒Y

, confidence shows the percentage in which Y

is bought with X

. It’s an indication of how often the rule has been found to be true.

conf(X⇒Y)=supp(X∪Y)supp(X)

For example, the rule T-shirt⇒Trousers

has a confidence of 3/4, which means that for 75% of the transactions containing a t-shirt the rule is correct (75% of the times a customer buys a t-shirt, trousers are bought as well). Three more examples:

conf(Trousers⇒Belt)=4/75/7=80%

conf(T-shirt⇒Belt)=2/74/7=50%

conf({T-shirt,Trousers}⇒{Belt})=2/73/7=66%

**2.3 Lift**

The lift of a rule is the ratio of the observed support to that expected if X

and Y

were independent, and is defined as

lift(X⇒Y)=supp(X∪Y)supp(X)supp(Y)

Greater lift values indicate stronger associations. Let’s see some examples:

lift(T-shirt⇒Trousers)=3/7(4/7)(5/7)=1.05

lift(Trousers⇒Belt)=4/7(5/7)(4/7)=1.4

lift(T-shirt⇒Belt)=2/7(4/7)(4/7)=0.875

lift({T-shirt,Trousers}⇒{Belt})=2/7(3/7)(4/7)=1.17

**2.4 Conviction**

The conviction of a rule is defined as

conv(X⇒Y)=1−supp(Y)1−conf(X⇒Y)

It can be interpreted as the ratio of the expected frequency that X

occurs without Y

if X

and Y

were independent divided by the observed frequency of incorrect predictions. A high value means that the consequent depends strongly on the antecedent. Let’s see some examples:

conv(T-shirt⇒Trousers)=1−5/71−3/4=1.14

conv(Trousers⇒Belt)=1−4/71−4/5=2.14

conv(T-shirt⇒Belt)=1−4/71−1/2=0.86

conv({T-shirt,Trousers}⇒{Belt})=1−4/71−2/3=1.28

3 Loading Data

First we need to load some libraries and import our data. We can use the function read.transactions() from the arules package to create a transactions object.

# Load libraries

library(tidyverse) # data manipulation

library(arules) # mining association rules and frequent itemsets

library(arulesViz) # visualization techniques for association rules

library(knitr) # dynamic report generation

library(gridExtra) # provides a number of user-level functions to work with "grid" graphics

library(lubridate) # work with dates and times

# Read the data

trans <- read.transactions("../input/BreadBasket\_DMS.csv", format="single", cols=c(3,4), sep=",", rm.duplicates=TRUE)

**4 Data Dictionary**

The data set contains 15.010 observations and the following columns,

Date. Categorical variable that tells us the date of the transactions (YYYY-MM-DD format). The column includes dates from 30/10/2016 to 09/04/2017.

Time. Categorical variable that tells us the time of the transactions (HH:MM:SS format).

Transaction. Quantitative variable that allows us to differentiate the transactions. The rows that share the same value in this field belong to the same transaction, that’s why the data set has less transactions than observations.

Item. Categorical variable containing the products.

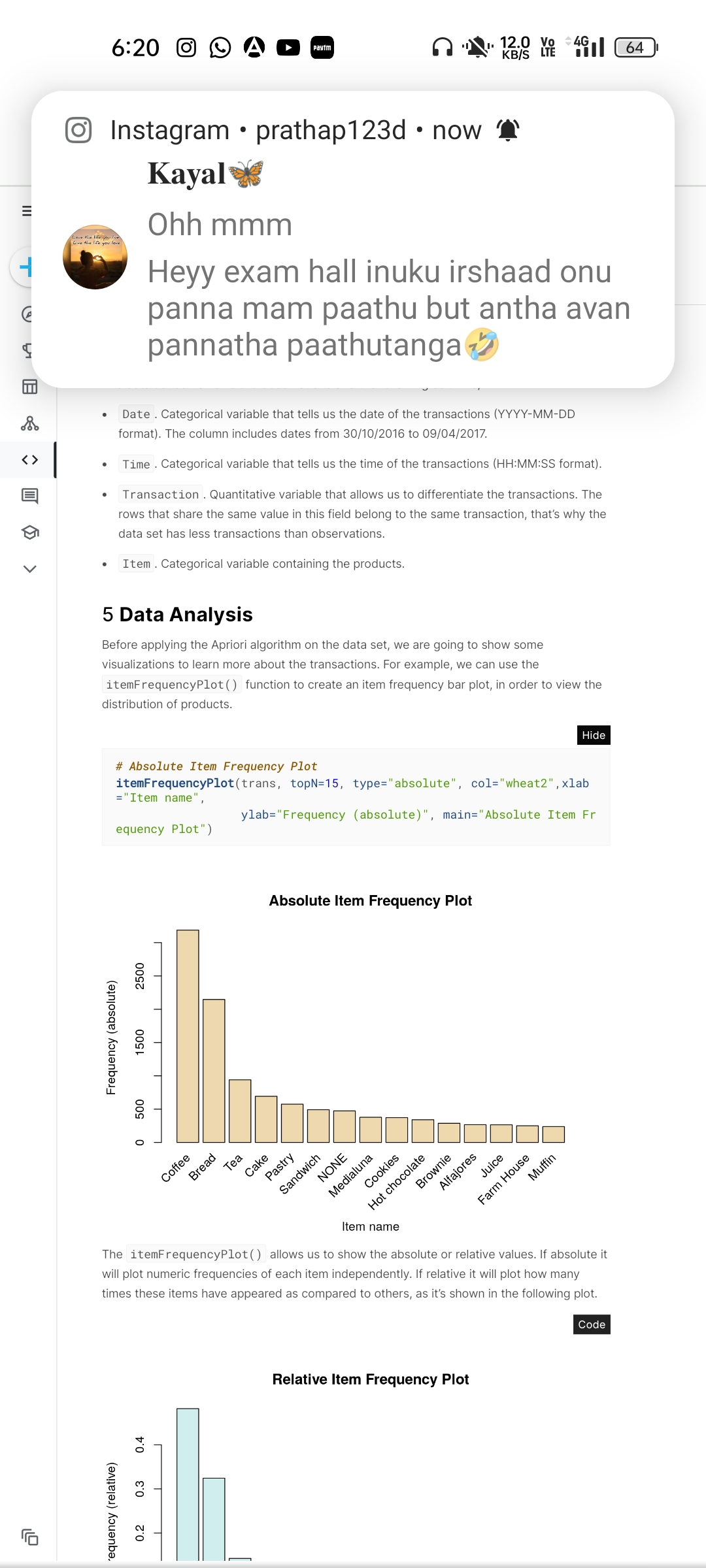
**5 Data Analysis**

Before applying the Apriori algorithm on the data set, we are going to show some visualizations to learn more about the transactions. For example, we can use the itemFrequencyPlot() function to create an item frequency bar plot, in order to view the distribution of products.

# Absolute Item Frequency Plot

itemFrequencyPlot(trans, topN=15, type="absolute", col="wheat2",xlab="Item name",

ylab="Frequency (absolute)", main="Absolute Item Frequency Plot")..



**CONCLUSION:**

In conclusion, market basket insights represent a valuable tool for businesses seeking to enhance their operations, boost customer satisfaction, and increase revenue. Through the meticulous analysis of transactional data, organizations can uncover hidden patterns and associations among products, paving the way for informed decision-making and strategic actions.

By leveraging market basket insights, businesses can implement a range of strategies, from cross-selling and product bundling to optimizing product placement and personalized marketing. These tactics not only increase sales but also enrich the overall shopping experience for customers.