EB5202 MTECH EBAC WEB ANALYTICS

PRODUCT RECOMMENDATION FOR E-COMMERCE SYSTEM

TEAM ANALYTICS SQUAD
CHOKKALINGAM SHANMUGASIVA (A0178230J)
MADAN KUMAR MANJUNATH (A0178237W)
NIU XUETONG (A0178464R)
SONG YUHAN (A0178350A)
YESUPATHAM KENNETH RITHVIK (A0178448M)

1.0 EXECUTIVE SUMMARY

1.1 Problem Description

Due to the explosive growth of data volume, a task of finding correlations between items in a dataset have received considerable attention, resulting in a variety of algorithms for both common and specialist mining tasks. In this project, a dataset of logs from e-commerce website containing the collection of sessions is leveraged to discover the associated items and buying habits of customers.

1.2 Business Goal

Our team is trying to measure the purchasing behavior of customers on the internet. We are aiming at discovering insights based on items which are clicked and bought by user and the click sequence. After identifying the association rules, the following questions can be answered:

1) What items are generally viewed together and can be recommended to a user?
2) Is the user going to buy items in a particular session?
3) Is there any pattern for click sequence in a session?

1.3 Summary of Method and Result

We initially had a dataset of 33 million clicks corresponding to 9.25 million user sessions and 1 million items being bought in 509,696 sessions. In order to down-sample the dataset, we decided to delete all user sessions that had category '0' item click in them as this represented missing category value. This still gave us 17 million clicks which we further down-sampled randomly to 1 million clicks corresponding to 286,780 sessions. We further divided them into a stratified training and test set with 80:20 split ratio. The data from the buy dataset was merged with the clicks dataset to get the end result of a session, if it was a buy or no buy session.

We used the train dataset to generate our association and sequence mining analysis. We made use of the libraries available in R- arules to perform association mining and arulesSequences to perform sequence analysis.

Our first goal was to recommend products to users based on the items that they have already clicked on or viewed. For this we used only the item ID in our baskets and no other feature to give us the rules that contained the items in the RHS as well as the LHS of the rule. The rules were generated using the apriori algorithm. These rules were later used to recommend items to users based on their clicks basket. We were able to achieve a precision of 41.01% using a set of 78 rules that were generated. The number of unique items in the dataset were 20,295. The minimum support and confidence we set were 0.002 and 0.34 respectively.

Our second objective was to predict whether a user that is using our platform is going to buy something in that session based on the items that he has already clicked on. We used the apriori algorithm to generate the rules for this as well. We used the items clicked on, the time of the day that the items were viewed as well as the day of the week that the items were being clicked on as features to generate the rules. We restricted the RHS of the algorithm to contain only BUY or NO-BUY terms and the LHS can contain the other features. We were able to generate 317 rules with a minimum support and confidence of 0.001 and 0.1. This gave us an accuracy of 75% and recall of 42.97%.

Our final goal was to analyze the sequence of steps or clicks that users generally follow while browsing our website. We used the cspade algorithm to generate the sequences and rule-induction to generate rules from these sequences. The rules that were generated were used to recommend items to users in a session. Then testing of these rules is done on the test dataset by achieving a precision of 20.01% while recommending items to users.

2.0 DATA PREPARATION

2.1 DATA UNDERSTANDING

Content of Data

The data was collected from an e-retail website and was posted by the RecSys Challenge 2015. The two datasets contain different records of events: one is the click events with the name of "yoochoose-clicks.dat", the other one is the buy event with the name of "yoochoose-buys.dat". Each record/line in the file has the following fields:

Click Event Dataset	
Column	Description
Session ID	The id of the session. In one session there are one or many clicks.
Timestamp	The time when the click occurred.
Item ID	The unique identifier of the item.
Category	The category of the item

Buy Event Dataset	
Column	Description
Session ID	The id of the session. In one session there are one or many buying events.
Timestamp	The time when the buy occurred.
Item ID	The unique identifier of item.
Price	The price of the item.
Quantity	How many of this item were bought.

Table1. Datasets Description

EDA of Data

Exploratory data analysis (EDA) is done to discover some main characteristics of user purchase behaviors along with sessions and product categories. According to the buying ratio changed with different time slots shown in Fig 1, consumers' buying peaks are concentrated in the afternoons between 12:00pm to 18:00pm as well as the weekends reaching almost 8% buying ratios.

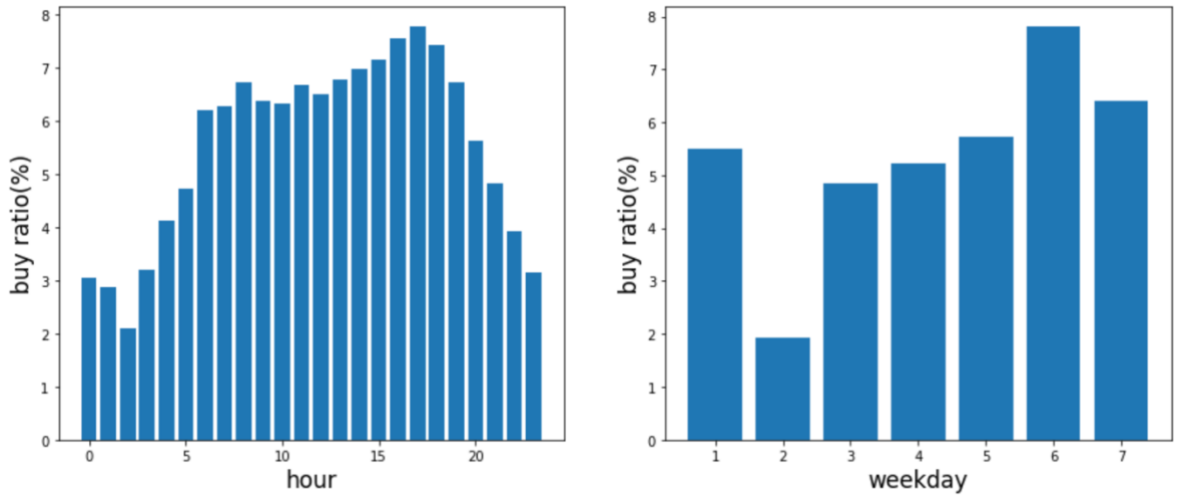

Fig 1. Buying Ratio Averaged for Time

Meanwhile, whether a consumer purchases a certain category of products is proportional to the number of times a product page is clicked on, that is, if a customer visits webpages related to such a category of products more times, he may have a greater willingness to purchase this category of products. As shown in Fig 2., Category 3 and 10 have relative higher buying ratio and more clicks simultaneously.

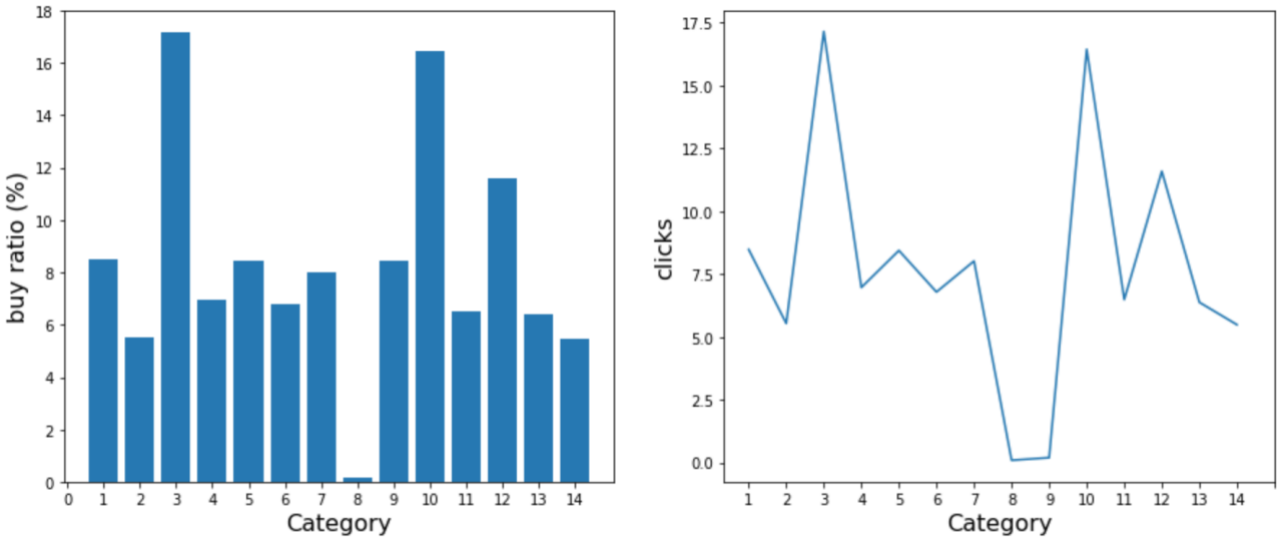

Fig 2. Buying Ratio Averaged for Category

In most of sessions, only 1 or 2 categories of products are purchased by consumers (Fig 3.) indicating currently online shopping customers have so clear shopping goals that most of the time they only focus on browsing webpages related to what they want to buy.

Most of sessions only contain less than 50 clicks (Fig 4.), which may also implicit that a large number of on-line shopping customers are so goal-oriented that they don't repeat to find what they want to buy many times.

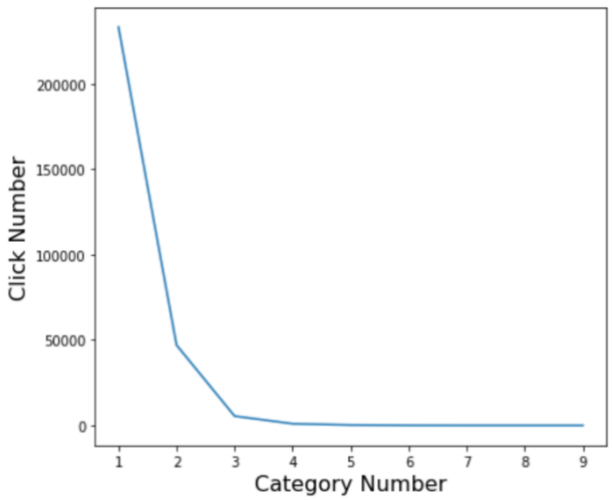

Fig 3. Buying Ratio Averaged for Category Number

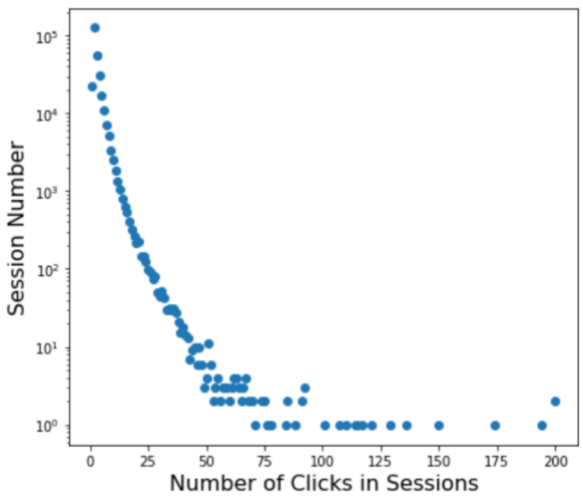

Fig 4. Distribution of Number of Clicks

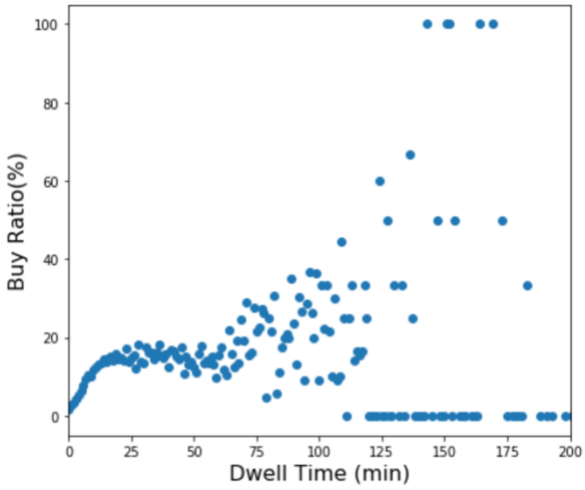

Fig 5. Buying Ratio Averaged for Dwell Time

Considering residence time on the website for customers in each session (Fig 5.), most consumers spend less than 125 minutes. However, it also indicates a slightly non-distinctive trend that the buying ratio increases with the dwell time on the websites.

2.2 DATA PREPROCESSING

The data preprocessing procedures are conducted in Python.

Merging Two Datasets

In order to find the click linkages between items and result of buy or not, we first merge the two datasets together and group them based on the "Session ID". We also add one new column called "status" to distinguish the sessions that end with a purchase to the ones that do not end with a purchase. All the sessions in the clicks dataset that end with a purchase have a corresponding entry in the buy dataset, hence the status of these sessions are set to true.

Feature Generation

The column of "Timestamp" contains the time value of year, month, week, day and the exact time. The most valuable information for the timestamp are day and hour for the macro level analysis. Then the two new columns are generated to help do the association and sequence analysis.

Data Cleaning

As for the category of the items, the "0" and "null" category don't contain the sensible inference of our association analysis goal so we delete the session with these two categories. For the cleaned dataset, it contains total 17 million records.

Down Sampling

Due to the large number of records after cleaning the data, we need to do the down sampling to further load the data into R. The down sampling is randomly selected 1 million records from the cleaned data and all the analysis and modeling is based this down sampled dataset.

Training and Testing Dataset Splitting

Before training the model, we split the data into training and testing. Because for the association rule analysis, we don't need to have the balance dataset, so we keep the original ratio of data of "buy" or "not

buy" for both the training and testing dataset. We use the Python Sklearn package of "train_test_split" function to do the random split.

After all the data preprocessing steps, the overall training data for building the model are 801621records and the testing data are 108379 records. Also, the distribution of the "buy" and "not buy" in the original dataset maintains in the training dataset.

3.0 MODELLING AND TESTING

3.1 MODEL BUILDING

Environment Used

We made use of R programming language and environment to do our analysis. There are a lot of libraries present in R that helps us derive these rules. Some of the packages that we used were "arules" and "arulesSequence". These packages give us association rules and sequences respectively.

Libraries Created

Some of the functions were not readily available to perform the testing of the rules generated. We have created helper functions that helped us validate our rules. These functions use items available in the whole session or basket to recommend items to the user and not only based on single items in the basket. This allows us to generate rules that have multiple items in the LHS of the rules. The LHS of the rules can also have other dimensions or features instead of a single type of feature as we use list matching instead of string matching to verify our generated rules.

Item Recommendation

Our first goal was to recommend items to users based on the items that they have previously clicked on. We have followed two ways of doing this recommendation- association rule mining and sequence rule mining. We have also made use of different combinations of features that we have generated to generate these rules and also the accuracy or precision that they give.

Association Rule Mining

For recommending items to users we create a dataset that only contains the item ID and the session ID to create the baskets. This data is then fed to the apriori algorithm to generate rules with different minimum values of support and confidence. The rules that were generated are of the form- if user clicks on item1 and item 6, then he is most likely to click on item 9 also, hence you recommend that item to him. We were able to generate a different set of rules by giving different sets of support and confidence values. The results obtained and their precision are reported more elaborately in the subsequent session. Some of the sample rules that were generated shown below. These form a subset of the generated rules for representational purposes.

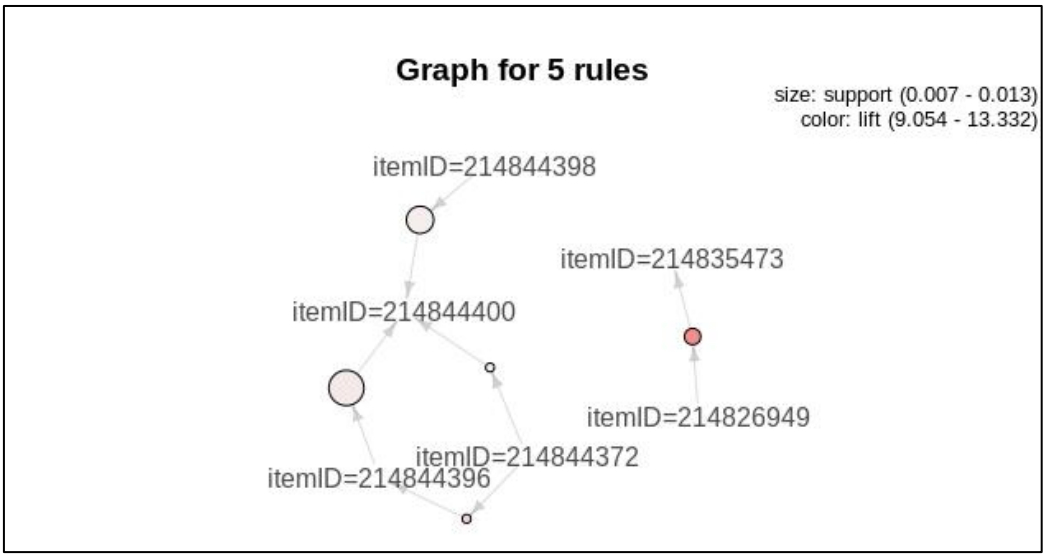

Fig 6. Item Recommendation

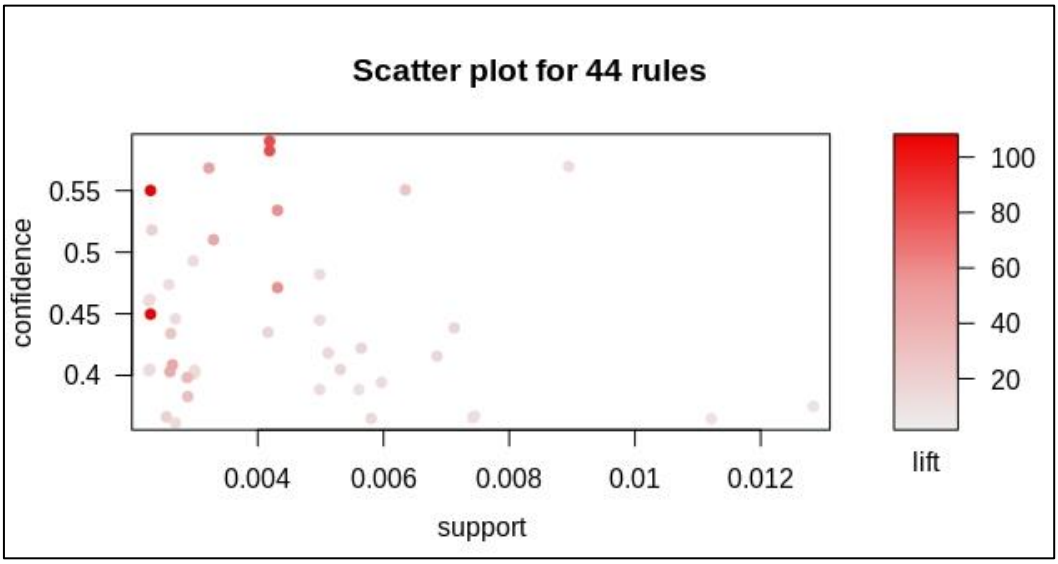

Fig 7. Scatterplot of Parameters

Sequence Analysis

In our dataset we also have the timing of the clicks. We can use the timing sequence to generate event ID's within a session for the clicks. This can later be supplied to the cspade algorithm to generate sequences that usually occur together. These results can be used to analyze the paths that are most frequently followed during a session on the platform. They represent the items that are related to each other.

Some of the sequences are shown below

```
<{itemID=214844400},{itemID=214844400}> => <{itemID=214844400}>
<{itemID=214844400},{itemID=214844396}> => <{itemID=214844400}>
<{itemID=214844398},{itemID=214844396}> => <{itemID=214844400}>
```

Fig 8. Sequence Rules

It can be seen that item 214844400 appears in a lot of sequences and also that people click back on this item after clicking on other items.

Purchase Prediction using Association Rules

For this task we not only make use of the items in our session or basket but also the other features that we have extracted earlier.
Some of the features that were used are:

- The hour of the day during which the clicks are made, as users are more prone to buy during certain periods of time.
- The day of the week during which the session happens.
- The category of the items being viewed.

Some of the features that could also be generated to enhance the model would be:

- The average time between item clicks for a session.
- The duration of the session on the platform.

These features were supplied to the algorithm and we restricted the terms on the LHS of the rules to only be these features while the terms on the RHS would only be, if the user purchases an item or not. We used different parameters for our minimum support and confidence and checked which ones gave us the best results. We also tried generating rules with different sets of features and also what they are trying to predict.

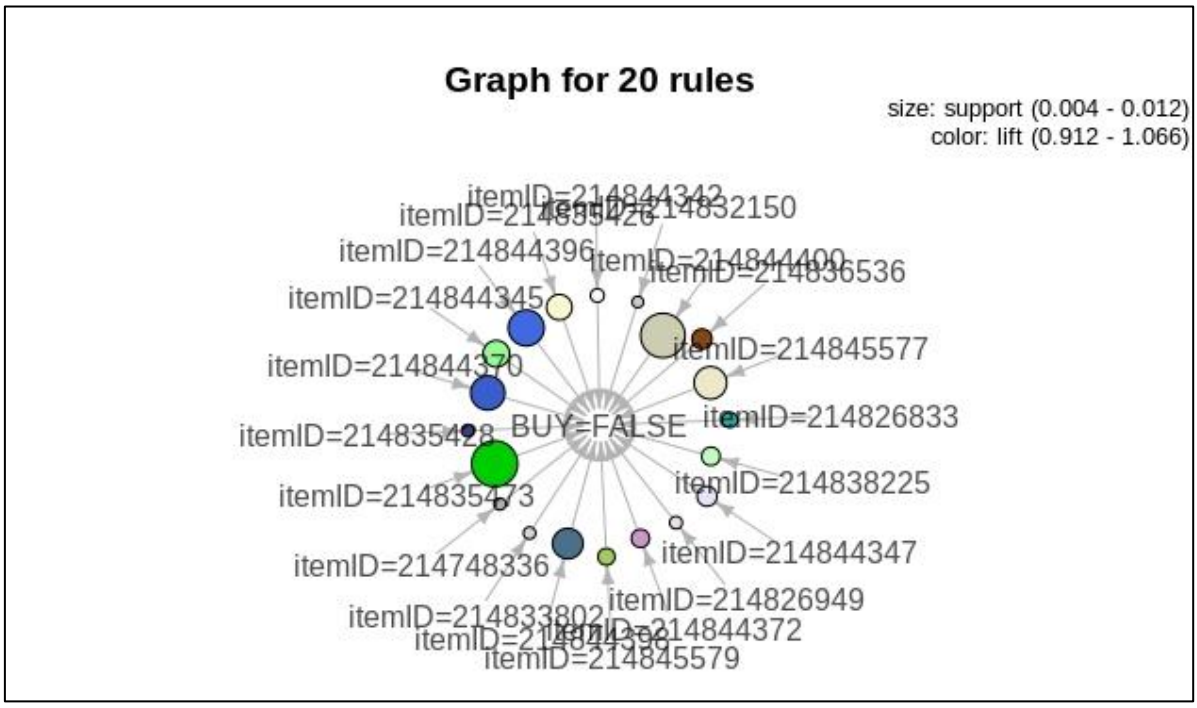

Fig 9. Rules to Predict When User won't Buy

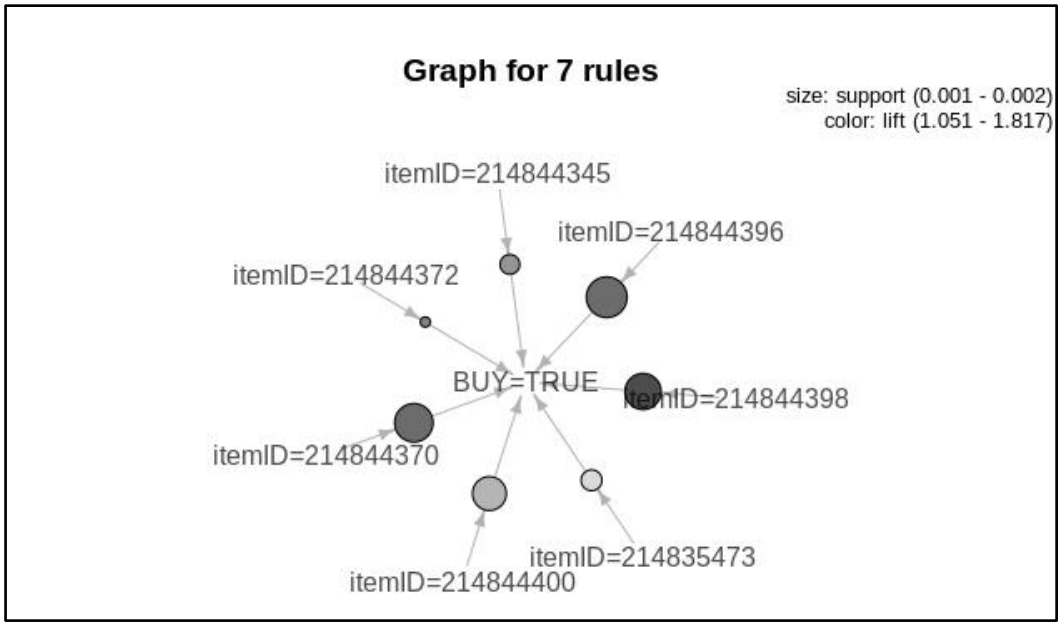

Fig. 10 Rules that Predict Only Based on Items

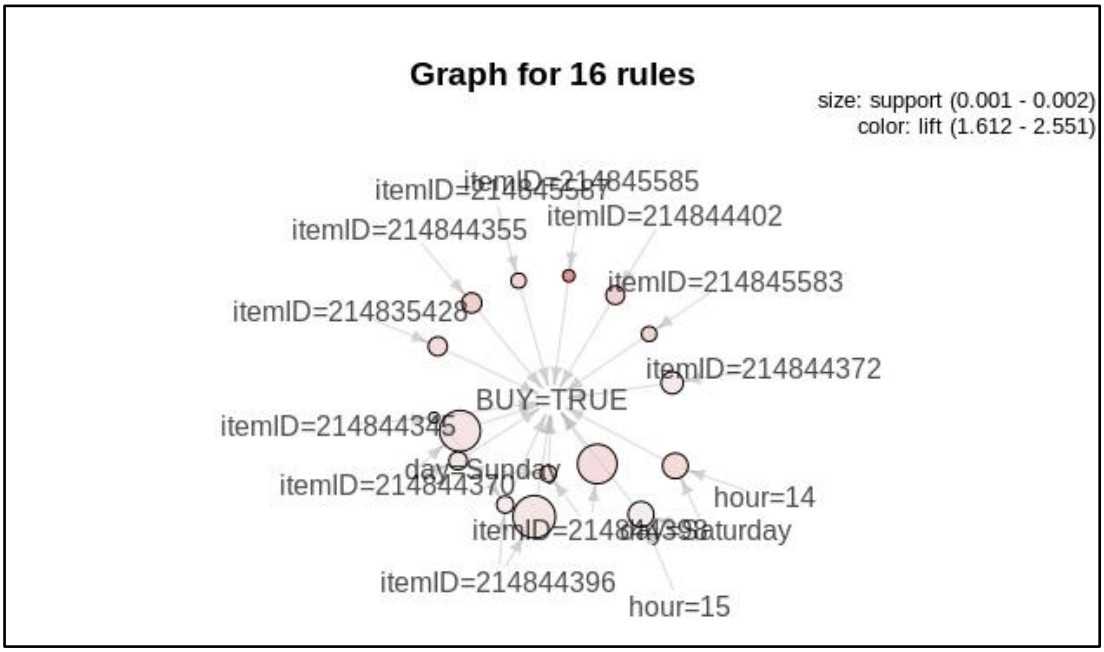

Fig 11. Using Different Features

The metrics and results of these models are explained in detail later.

3.2 MODEL TESTING

3.2.1 Next Item Prediction

First model was built to recommend products to users based on their click history. That is to Identify the items that sell in conjunction with other items on an everyday, promotional, and cyclic basis, as well as co-occur between purchases over time.

Apriori algorithm was used to generate these rules. This model uses item ID in each session to generate the rules. For the apriori algorithm each session is taken as a basket and click events as taken as products. There is a high imbalance in the datasets as very few sessions have a buy event. Thus, the rules generated matches very few sessions. However, for the given business case "precision" is the most significant feature.
Precision helps us to understand how relevant are the fired rules.

Rule Pruning	Value	Metrics	Value
Support	**0.0022**	Sessions with matching rules	**19639**
Confidence	**0.3444**	Correctly Predicted Sessions	**8054**
Min Length	**2**	Precision	**41.0 %**

3.2.2 Sequence Mining for Item recommendation

As click events are sequential in nature, sequential rule mining (cspade algorithm) was implemented to find the most relevant item we can recommend to the customer. First sequence mining was used to find the most frequent patterns in the click/view history. Then rule induction was done over these patterns to generate rules to predict the most likely product a customer would visit/buy.

Compared to apriori algorithm, sequential mining was able to match majority of the sessions. However, the precision value was very low.

Rule Pruning	Value	Metrics	Value
Support (Sequence mining)	**0.001**	Sessions with matching rules	**2493559**
Confidence (Rule Induction)	**0.01**	Correctly Predicted Sessions	**50600**
Min Length	**2**	Precision	**20.3%**

3.2.3 Prospective buyer prediction

The second objective was to identify those sessions that retraces frequently occurring patters that will lead to a "buy" event. That is to identify a prospective customer before the session ends and target them to increase the conversion rate. Apriori algorithm will be used to mine association rules.

Prediction based on Item Id:

- **No of rules:** 71

- **Input feature:** Item Id and Session status

Initially the item ids and the status of the sessions those items were present was used to generate association algorithm. Thus, we will be generating associations to understand the Item ids which leads to a prospective buy.

Rule Pruning	Value	Metrics	Value
Support	**0.0015**	Accuracy	**0.8018**
Confidence	**0.125**	Recall	**0.3809**
Min Length	**2**	Precision	**0.112**

(I) Prediction based on All features after pruning:

By including more features like the time of day and day of week, we improved the model. This generates 848 rules. Then different subsets of rules are generated by applying rule pruning. This is done by varying the values of support, confidence and list cutoffs. As the distribution of the rules were in a very narrow range, even a slight increment in the cutoff drastically reduces the no of rules.

- **No of rules:** 18
- **Input feature:** Item Id and generated features such as time of day and day of week with session status

Rule Pruning	Value	Metrics	Value
Support	**0.0019**	Accuracy	**0.8328**
Confidence	**0.1302**	Recall	**0.2759**
Min Length	**2**	Precision	**0.1058**

(II) Prediction based on All features after pruning:

- **No of rules:** 317
- **Input feature:** Item Id and generated features such as time of day and day of week with session status

Rule Pruning	Value	Metrics	Value
Support	**0.001**	Accuracy	**0.7576**
Confidence	**0.1**	Recall	**0.4296**
Min Length	**2**	Precision	**0.1002**

We found the best result by taking 317 rules. This is based on the Recall metrics. In the given business case it is most important to maximize the prospect conversion rate with minimum spend. Thus, recall is more important than accuracy as a validation metrics.

4.0 CONCLUSION

From this project we were able to get a practical experience of how association mining and sequence mining can be used to perform market basket analysis. Then fine tune or prune the huge list of rules that we got are taken by varying the support and confidence. We were also able to see how the use of different features can improve our model or also harm it.

Deploying to Production

- Our item recommendation can be deployed to production with a few more improvements as it has a precision of 40% and predicts the correct item out of a possible 20,000 items to predict.
- We also made 19,639 predictions out of the 57,000 sessions, which is a low and has to be improved further.
- We need to further improve our purchase prediction system to accurately predict our buying customers.
- Currently our system has very low precision (around 10%), which means only 1 in 10 customers that we predict are going to buy, actually end up buying.
- Our recall is also around 40%. We need to improve this further as we do not want to miss out on actual customers by predicting that they are not going to buy.
- In our business case, it is valuable to have a good recall as it gives us a measure of how accurately we are predicting our true customers. Most of the users (95%) on platform do not end up purchasing an item on our platform.