**Association Rules Project2**

Hao Zhu

**Executive summary:**

The Dillard’s has never rearranged the floors of the store before. They want to use the transaction data they collect to analyze the relationship between products so that they can rearrange the SKU to get the two or more which have strong relationship close and then improve the sales.

In this case, we use the Dillard’s point of sales (POS) data over a period of time to find the 100 SKUs that are best candidates to modify the planograms. The data set is quite large, so after importing it, we randomly choose 5000,000 transaction records to analyze this problem.

The algorithm we use is association rules learning. It is intended to identify strong rules discovered in databases using some measures of interestingness. We first review the summary of the transaction files to set the appropriate minimum support 0.000008 and confidence 0.01. We generate the 83 rules containing 90 SKUs, after removing the redundant rules, there is 45 rules remaining.

We choose 20 rules from all the rules we get by the order of the lift. The rules are quite strong, the lifts are all big enough although the support and confidence are relatively low due to the large number of datasets.

**Problem Statement:**

The retailer is interested in rearranging the floors of the stores. For budgetary reasons (manpower) they can only make at most 20 moves across the entire chain. (A move consists of moving one SKU to a different position.) The problem is how to find the 100 SKUs that are best candidates to modify the planograms from the transaction record data.

**Assumptions:**

* This is the first time the company is performing this analysis and thus it is highly unlikely that SKUs are already appropriately close to each other.
* Assume that the reason why the items are returned is not related to the association rules, so the return items are not included in our analysis .
* One package is calculated as one regardless of how many items in this package.
* We do not consider the influence produced by different areas and states, so all the customers and retail stores are relative.
* Assume same transaction code, store and date represent one transaction.
* The number of one SKU per transaction is recognized as one, situation that more purchase on the one SKU per transaction is not taken into consideration.
* The aim of the association rules is to increase number of sales not the profit.

**Methodology**

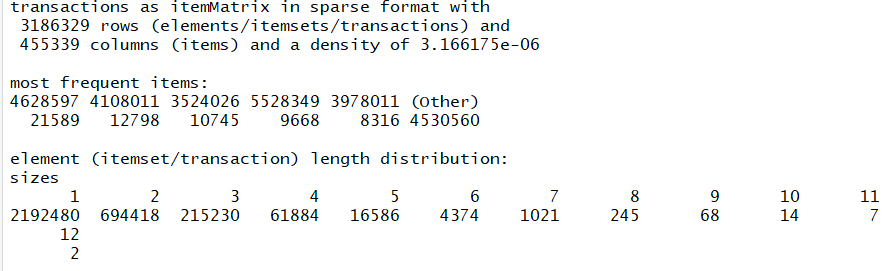
First, import the data, explore the features and preprocess the data, set up the table of SKU per transaction as basket exclude the return transaction. We use “TRANNUM”,” STORE”, “Sale Date” to identify the single transaction and combine SKU with same “TRANNUM” ,”STORE”, “Sale Date” as one transaction record.

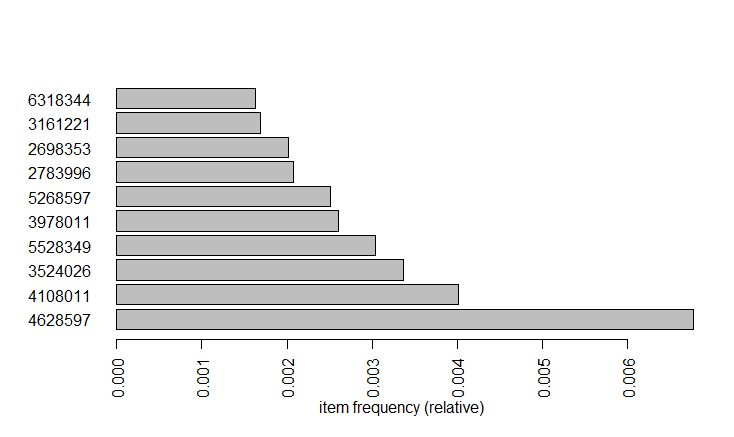
Next, in order to set the minimum support and confidence, we see the summary of the transaction and then set the support and confidence. After that, we can generate the association rules mining to find the best combinations. When we get the rules, remove redundant rules.

At last, visualize the association rules results.

**Analysis:**

After importing the transaction files, we summary it and plot the top10 support item.





From this summary, we appropriately set the minimum support as 0.000008 and the minimum confidence as 0.01, then perform the association rules mining. We generate the 83 rules containing 90 SKUs, after removing the redundant rules, there is 45 rules remaining. We inspect the rules by the order of the lift. As the problem required, we need to select the best 20 rules for floor rearrangement. So, the rules or SKUS below are what we advise company to rearrange.

lhs rhs support confidenc lift count

[1] {2657335} => {2587335} 8.473701e-06 0.13432836 1685.095840 27

[2] {6782521} => {6792521} 8.473701e-06 0.09782609 1078.567813 27

[3] {6600353} => {6420353} 8.159860e-06 0.09219858 1076.098953 26

[4] {4732521} => {4762521} 8.473701e-06 0.09121622 870.194237 27

[5] {4722472} => {4772472} 8.473701e-06 0.09030100 822.082016 27

[6] {6520353} => {6500353} 8.159860e-06 0.08099688 726.993585 26

[7] {6372521} => {6402521} 1.098443e-05 0.09408602 725.881401 35

[8] {8412644} => {8402644} 8.787542e-06 0.09491525 704.967895 28

[9] {8540723} => {8520723} 8.159860e-06 0.08666667 686.936600 26

[10] {1543503} => {1563503} 9.415224e-06 0.09036145 683.898563 30

[11] {768635} => {828635} 8.159860e-06 0.07142857 573.286974 26

[12] {6412521} => {6402521} 9.101383e-06 0.07323232 564.993403 29

[13] {8132644} => {8142644} 8.159860e-06 0.07027027 547.443032 26

[14] {6570353} => {6560353} 9.729064e-06 0.09198813 516.028958 31

[15] {7222521} => {6972521} 8.473701e-06 0.07068063 501.585157 27

[16] {4462521} => {4512521} 1.223979e-05 0.08904110 497.744256 39

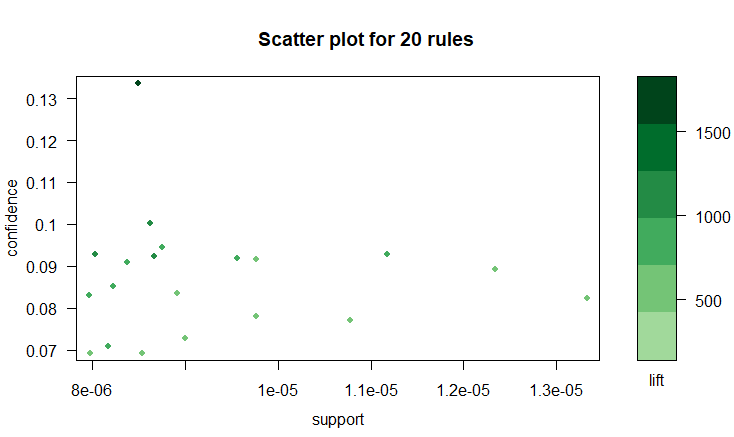
[17] {6742521} => {6752521} 1.318131e-05 0.08123791 493.048975 42

[18] {8542644} => {8522644} 9.729064e-06 0.07730673 478.300361 31

[19] {6972521} => {7232521} 1.067059e-05 0.07572383 468.506870 34

[20] {6320353} => {6340353} 9.101383e-06 0.08605341 465.525439 29

**Conclusion**

We use the package of “arulesViz” to visualize our selected rules. From the picture, we can find that the support is very small for most of the rules because the dataset is too large for the single SKU. Also, there is one rule that have relatively large confidence and lift compared to other rules, which means this rule is better than others, Those 20 rules’ lifts are all larger than 3, which means they are quite convincing. 

Although we provide the 20 rules, the rules still need to be confirmed due to the large and broad dataset. The relationship between two SKUs is difficult to make it confident due to the dataset now we used. However, we can use these rules to dig out the relationship between two SKUs which people do not recognize before.