Thomas Cho

IEMS 308

Association Rules

**Executive Summary**

Dillard’s wants to increase sales, and they plan on rearranging the floors of the stores in order to better allow customers to buy more. Only 20 rearrangements can be made. Association rule is a great technique to discover item pairings and allow managers to strategically utilize capital to increase such item pairs.

**The goal of the analysis is to find the best item pairs that can best increase sales.**

The analysis tell us “If I buy item X, what other items Y do customers also buy”. Analysis was done on a subset of the data, particularly the transactions that occurred in store #4903, a store in Illinois. I chose a store in Illinois since we live in Illinois. With a minimum of 0.001 support and 0.65 confidence, what we see is shown in table 1 on page 3. What’s surprising is that the item pairs with the highest confidence is not a 1 to 1 pair but rather 2 to 2 pair. When customers bought items # 7808101 & #3631365, 91% would also buy #3524026 & #4108011. A very popular item is #4108011. A lot of customers buy that as a compliment to the first item. Assuming the SKU numbers are grouped by categories and so the closer the numbers are, the more similar, then it seems a lot of items are within the same or similar category. Except for a few exceptions such as #9, in which customers buy 9317426 and then 3524026 at a 77% rate. With more data on SKUs, then I would definitely look into #9. I give about 100 SKU, and so managers can freely manage and choose what items they want to move to maximize their profits.

**Methodology**

The first step is to load the data. Since the dataset is rather large, I looked at the transactions for one store. There is a crucial assumption that this store is a good generalization of all other Dillard’s stores. The reason for not random sampling is because I do not want to erase items in baskets. By keeping the data to one store, I know the baskets are not missing items.

The second step is to understand the data. The csv files were not given headers. Therefore, using the database schema and knowledge of the value ranges, ‘SKU’, ‘Store’, and ‘Trannum’ is deduced and needed in the transaction csv file for analysis.

The third step is preprocessing and using the Apriori algorithm to calculate the confidence of item pairs. The rows in the current transaction file represent one SKU. The rows are transformed to represent a “basket” or one transaction that a customer paid. This transformed is done by grouping the SKU numbers by date of purchase and transaction number. Together, the grouped SKUs represent one basket.

**Analysis**

*Understanding the data*

The data contains way too many rows and is too big (memory extensive) and time-consuming to analyze wholly. Pandas is used to load and read in chunks of 1000 and concatenated all rows in store #4903. This store contains 145,456 records. Secondly, the headers had to be deduced using the other CSV files. Skstinfo file shows ‘Store’ to have a range of 2-9909, and that translates to column 1 in transaction file. The Skuinfo file shows ‘SKU’ to have range of 3~ 9,000,000 and that translates to column 0 in transaction file. Using deduction and comparing variable types, ‘trannum’ is column 2 in the transaction file.

*Preprocessing*

I group the SKUs by ‘Trannum’ and ‘Date’. This is another assumption: a basket has to be on the same day and it has the same trannum; the trannum may reset and repeat on different days. Therefore, we group on two conditions to get the SKU baskets. After, I transform the series back into a data frame, in which I iterate through and create a list of only the SKUs. This list of baskets is what will go into the analysis.

*Apriori Algorithm*

The time it takes to iterate through all possible combinations would take forever. Therefore, we use the Apriori principle to reduce the time and complexity of our problem. If the item is infrequent, then the item’s subsets will be less than that. Therefore, if I provide a minimum frequency that I’m satisfied with then I can get rid of the items before calculating the different combinations. I use a Apriori implementation.\* The standard of 0.1-0.2 of minimum support was tested. However, almost no items met such requirements. I base that off the assumption that because the SKU values can be up to 9,900,000, the frequency of one item to be in between one and two million is too much. After iterating and playing with the values, 0.001 is a good choice, having enough items to start finding the combinations and not having too many items that it causes the analysis to fail. Once we found the items that meet the minimum support value, I calculate the confidence of the item pairs with a threshold of 0.65.

**Findings**

|  |  |  |  |
| --- | --- | --- | --- |
|  | IF | THEN | Confidence |
| 1 | 7808101, 3631365 | 3524026, 4108011 | 0.9091 |
| 2 | 803921, 2784759 | 3524026, 4108011 | 0.83333 |
| 3 | 4876578 | 3524026 | 0.8333 |
| 4 | 1298464 | 1308464 | 0.8333 |
| 5 | 1308464 | 1298464 | 0.8333 |
| 6 | 4876578 | 4108011 | 0.8333 |
| 7 | 3787564, 1832285 | 3524026, 4108011 | 0.8333 |
| 8 | 8142644 | 8122644 | 0.8125 |
| 9 | 9317426 | 3524026 | 0.7692 |
| 10 | 8132644 | 8142644 | 0.7692 |
| 11 | 2988370 | 2783996 | 0.7692 |
| 12 | 2726578, 726718 | 3524026, 4108011 | 0.7692 |
| 13 | 2267565 | 4108011 | 0.75 |
| 14 | 5980076 | 5930076 | 0.7368 |
| 15 | 7057566 | 4108011 | 0.7368 |
| 16 | 7808101, 1832285 | 3524026, 4108011 | 0.73333 |
| 17 | 803921, 2698353 | 3524026, 4108011 | 0.73333 |
| 18 | 5487088 | 2001637 | 0.7143 |
| 19 | 1943241 | 3524026 | 0.7143 |
| 20 | 5108107, 7808101 | 3978011, 4108011 | 0.7143 |

Table 1: Top 20 confidence

The table above shows you the item pairs and confidence values for the top 20 confidence values in store #4903. What’s interesting is that the first and highest confidence value has a 2 to 2 pair. Those who buy 7808101 & 3631365 have a 91% of buying 3524026 & 4108011. Therefore, now Dillard’s can utilize this knowledge and rearrange the store to make these items farther away to in order for customers to go through the whole store. The rest of the table is in the appendix. As you see in lines 4 & 5, they are the same items but flipped. Therefore, the first twenty does not translate to 20 rearrangements. Therefore, about 70 item pairs were calculated to sufficiently provide Dillard’s with about 100 SKU’s that can be moved around.

**Next steps**

Association rule is great to uncover relationship in the records. The limitations of this analysis are the memory and time to run the entire data frame. The assumption that one store is a good representation of all Dillard’s operations is very bold and not practical, or at least is prone to risk. That is why, next steps can be to look at all the data, and if that’s not possible, run this for several stores, and see underlying item pairs across the stores. Second, Apriori algorithm is not the most efficient algorithm, and therefore other algorithms can be used to test for patterns.

If we have more information on the SKU values, such as the item name and description can push this analysis further. The item names can be split into categories, and see the relationships of item categories.

\* The reference to the code is https://gist.github.com/marcelcaraciolo/1423287

**Appendix:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | IF | Then | Confidence |
| 1 | 7808101, 3631365 | 3524026, 4108011 | 0.9091 |
| 2 | 803921, 2784759 | 3524026, 4108011 | 0.83333 |
| 3 | 4876578 | 3524026 | 0.8333 |
| 4 | 1298464 | 1308464 | 0.8333 |
| 5 | 1308464 | 1298464 | 0.8333 |
| 6 | 4876578 | 4108011 | 0.8333 |
| 7 | 3787564, 1832285 | 3524026, 4108011 | 0.8333 |
| 8 | 8142644 | 8122644 | 0.8125 |
| 9 | 9317426 | 3524026 | 0.7692 |
| 10 | 8132644 | 8142644 | 0.7692 |
| 11 | 2988370 | 2783996 | 0.7692 |
| 12 | 2726578, 726718 | 3524026, 4108011 | 0.7692 |
| 13 | 2267565 | 4108011 | 0.75 |
| 14 | 5980076 | 5930076 | 0.7368 |
| 15 | 7057566 | 4108011 | 0.7368 |
| 16 | 7808101, 1832285 | 3524026, 4108011 | 0.73333 |
| 17 | 803921, 2698353 | 3524026, 4108011 | 0.73333 |
| 18 | 5487088 | 2001637 | 0.7143 |
| 19 | 1943241 | 3524026 | 0.7143 |
| 20 | 5108107, 7808101 | 3978011, 4108011 | 0.7143 |
| 21 | 2027565 | 4108011 | 0.7142 |
| 22 | 6706135 | 6656135 | 0.7142 |
| 23 | 5108107, 7808101 | 3524026, 4108011 | 0.7142 |
| 24 | 6432840 | 3524026 | 0.7059 |
| 25 | 1832285, 726718 | 3524026, 4108011 | 0.7059 |
| 26 | 1821637 | 2001637 | 0.6923 |
| 27 | 3978011, 3787564 | 3524026, 4108011 | 0.6875 |
| 28 | 4976322, 3978011 | 3524026, 4108011 | 0.6875 |
| 29 | 5778109 | 4108011 | 0.6842 |
| 30 | 3978011, 7808101 | 3524026, 4108011 | 0.6842 |
| 31 | 772991 | 792991 | 0.68 |
| 32 | 3427564 | 4108011 | 0.6667 |
| 33 | 5910076 | 5900076 | 0.6667 |
| 34 | 3448186 | 3524026 | 0.6667 |
| 35 | 803921, 264715 | 3524026, 4108011 | 0.6667 |
| 36 | 3524026, 2784759 | 803921, 4108011 | 0.6667 |
| 37 | 2698353, 3978011 | 3524026, 4108011 | 0.6667 |
| 38 | 5327384 | 3524026 | 0.66667 |
| 39 | 1226187 | 3524026 | 0.6522 |
| 40 | 2267565 | 3524026 | 0.65 |
| 41 | 4329285 | 4108011 | 0.6471 |
| 42 | 4008011 | 4108011 | 0.6333 |
| 43 | 5778109 | 3524026 | 0.6318 |
| 44 | 6328344 | 3524026 | 0.619 |
| 45 | 7808101 | 4108011 | 0.619 |
| 46 | 8122644 | 8142644 | 0.619 |
| 47 | 6707297 | 4108011 | 0.6154 |
| 48 | 448103 | 2783996 | 0.6111 |
| 49 | 2001637 | 5487088 | 0.6061 |
| 50 | 3844099 | 4108011 | 0.6 |
| 51 | 5778109 | 1832285 | 0.5789 |
| 52 | 3998011 | 3524026 | 0.5758 |
| 53 | 3968011 | 3524026 | 0.5714 |
| 54 | 2784759 | 4108011 | 0.5667 |
| 55 | 2127565 | 3524026 | 0.5556 |
| 56 | 3854099 | 4108011 | 0.5526 |
| 57 | 5557088 | 5487088 | 0.55 |
| 58 | 726718 | 3524026 | 0.5417 |
| 59 | 1821637 | 5487088 | 0.5384 |
| 60 | 7596135 | 6656135 | 0.53125 |
| 61 | 3874099 | 3524026 | 0.5294 |
| 62 | 5878946 | 3524026 | 0.5217 |
| 63 | 3390065 | 6490080 | 0.5217 |
| 64 | 9667426 | 3524026 | 0.5185 |
| 65 | 6468364 | 4108011 | 0.5185 |
| 66 | 2267565 | 3978011 | 0.5 |
| 67 | 4062567 | 3161221 | 0.5 |
| 68 | 2726578 | 3524026 | 0.5 |
| 69 | 2716578 | 4108011 | 0.5 |
| 70 | 2716578 | 3524026 | 0.5 |
| 71 | 5487088 | 1821637 | 0.5 |
| 72 | 4008011 | 3524026 | 0.5 |