Rohan Agarwal

Market Basket Analysis of Dillard's Transactions

Executive Summary

Market basket analysis was performed on a data set of over 120 million transactions in over 450 stores for a national market chain named Dillard's. The goal of the analysis was to find the top 100 association rules to implement in single stores which involve shifting the position of one item closer to another commonly bought item.

The analysis began with sampling the data while accounting for two main factors affecting the variability of the nature of transactions – location of the store and time of year. Data across all times was selected, but in a subset of locations that followed the probability distribution of the entire dataset. Subsequently transaction lines were generated with a unique identifier based on transaction ID, sale date, store number, and register number.

After conducting association rule analysis across a subset of 40 stores, the rules generated for each store were combined and then sorted across a custom metric called benefit, which is the product of the probability of the combination of items occurring in a transaction with the price of the item on the antecedent side of the association rule in order to represent the financial value of the association rule.

The top 100 rules were selected based on the highest financial benefit, and characteristically seem to be item combinations with a low lift and low confidence – that is items that are purchased commonly, but still have scope for improved sales. Implementing the rules practically can be done by placing the constituent items together in the store, and the effectiveness can be measured in terms of increased sales.

Rohan Agarwal

Introduction

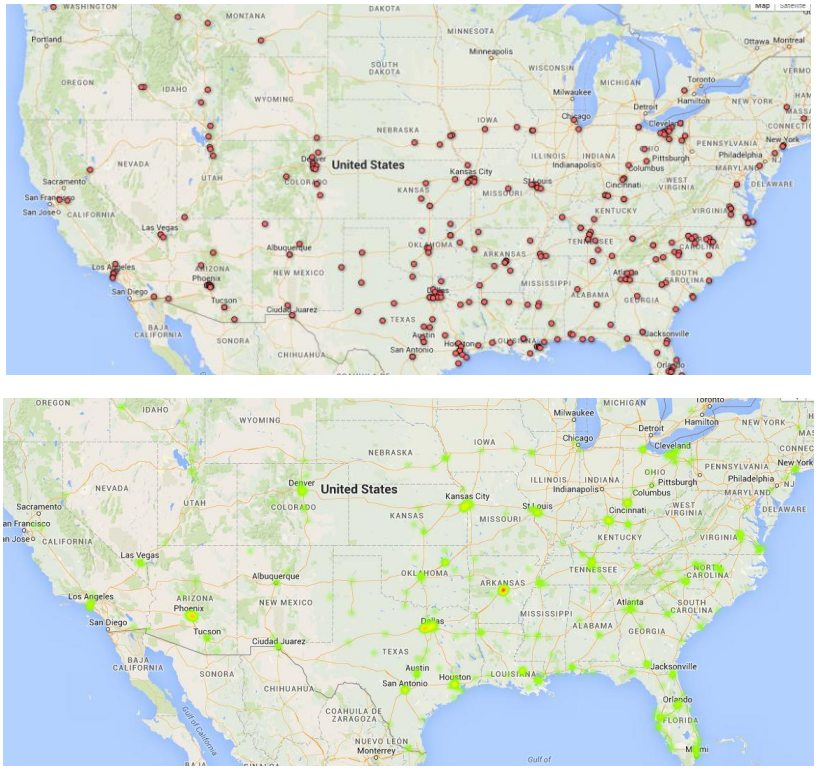
Given data crossing 120 million database rows, the goal of this analysis was to find commonly purchased items using market basket analysis with the concepts of support, confidence, and lift. The final consequence of doing so would be to rearrange the positioning of the items so that they are purchased more often, increasing revenues for Dillard's.

The goal was to find the best 100 rules that could be used to move an item from one department of one store to another department of the same store. However after subsequent analysis, all of the rules that were found with reasonably high support involved items from the same department. Therefore the goal then became to be to find 100 items in stores to place together, regardless of department, with one "move" constituting shifting the position of one item in one store.

Methodology

The data was extremely large, constituting over 12 gigabytes and over 120 million database rows. Because of this a sample of stores had to be taken; however the immediate problem is that it is impossible to know that the 100 moves are the best moves to make without analysing all of the data.

As a result my goal was to capture all of the variability in a sample, so that the 100 moves generated would also represent the distribution of demand across the country. Looking at the data, it was hypothesized that there are two main factors affecting transactions: the location of the store as well as seasonal purchases.



Figures 1 & 2: Mapping and Heat Plotting Stores

The method for sampling as a result was to:

1) Find the probabilistic distribution of stores across the country
2) Multiply the distribution by the number of stores to be sampled, in order to find out how many stores should be sampled from each state
3) Randomly select samples at a state level, across the entire year that the data spans, in order to capture time variability

Simply put,

$$Number\ of\ states\ to\ select = probability\ that\ a\ state\ occurs\ across\ all\ transactions * number\ of\ states\ to\ sample$$

Following the steps, the probability distribution of stores found across states is:

NJ	NY	WY	IN	CA	ID	IL	MT
0.000000000	0.000000000	0.003012048	0.006024096	0.009036145	0.009036145	0.009036145	0.009036145
NE	NV	IA	MS	NM	UT	KS	SC
0.012048193	0.012048193	0.015060241	0.018072289	0.018072289	0.018072289	0.024096386	0.024096386
AR	KY	VA	CO	GA	AL	OK	AZ
0.027108434	0.027108434	0.030120482	0.033132530	0.033132530	0.036144578	0.039156627	0.042168675
MO	LA	NC	TN	OH	FL	TX	
0.042168675	0.045180723	0.045180723	0.045180723	0.063253012	0.132530120	0.171686747	

Figure 3: Probability Distribution of Stores across States

Multiplying this distribution of locations by the store sample and rounding gives the number of stores to use from each state. With 40 stores, the number of stores to from each state is:

NJ	NY	WY	IN	CA	ID	IL	MT	NE	NV	IA	MS	NM	UT	KS	SC	AR	KY	VA	CO	GA	AL	OK	AZ	MO	LA	NC	TN	OH	FL	TX
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	3	5	7

Figure 4: Number of Stores to Select per State

With the pseudo-random sampling method to capture the variability across location and time, and a selection of 40 stores (shy of 10% sample), the result was a sample size of 11.6 million rows.

In order to process the data further, stores were iteratively analysed. For every store, the following seven steps occurred:

1) Selecting the transactions from one particular store out of the larger sample database.
2) Converting the data into transactions of the form {item a, item b…} => {item c, item d…}. This was achieved using commands part of the *arules* package in R, which implements the apriori and eclat algorithms. Note that a unique identifier was used for transactions, based on transaction number, store, register, and sale date.
3) Performing apriori analysis on the transactions in order to determine association rules. This was achieved using the *arules* package in R, with entered parameters of .0001 support and .0001 confidence. These were intentionally left low as executing the apriori algorithm over a single store's data has a much lower probability of a combinatorial explosion occurring.

4) Parsing the rules for further analysis, which consisted of cleaning the data frame of transactions with string manipulation, as well as making sure that all of the rules consisted of valid SKUs by checking against a separate database provided (skuinfo). If there were any invalid SKUs, they were discarded. At this point transactions with multiple items in the left side of the association rule were also discarded, as determining product placement across more than two items needs to be performed in multiple moves rather than a single one.
5) Finding the department for every item on each side of the association rules, by cross checking against the list of SKUs and adding the data to the data frame of rules.
6) Finding the price of the item on the right hand side of every association rule, which represents the gain in income from having the right-side item purchased given that the left-side purchase is made.
7) Calculating the projected benefit as the product of the support and the price of the item as a product of the support and price.

After processing the data to find rules with support, confidence, lift, department, and their prices, all of the rules for each store were combined into one data frame, and then the top 100 rules were selected based on their benefit. Note that the "benefit" for each item is:

$$benefit = support * price$$

Results

The plot below shows all of the rules generated by the sample below. Interestingly, many of the rules with high lift have high confidence but with very low benefit. This suggests that the "rare" transactions aren't necessarily for expensive items, but just for uncommon purchases.

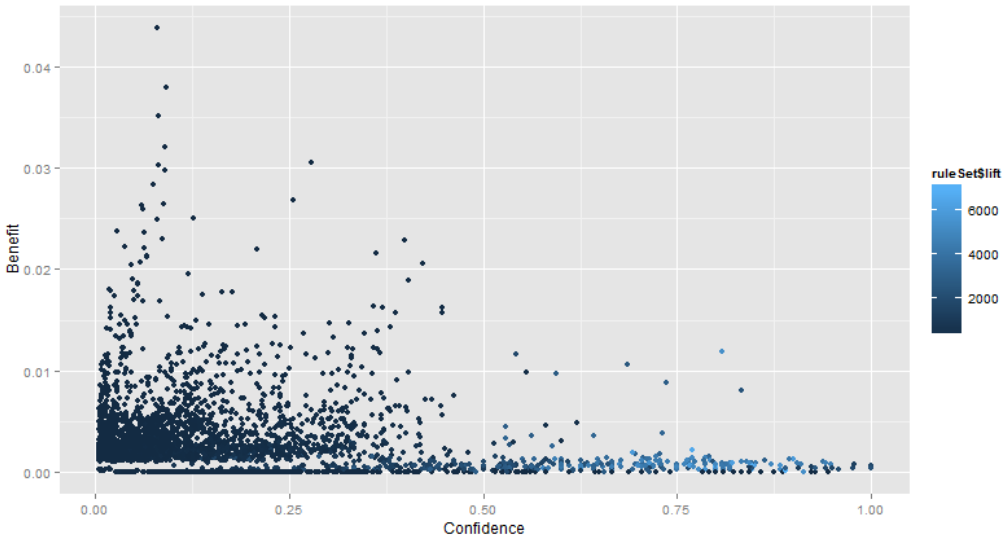


Figure 5: Results of Market Basket Analysis

Given that the top rules were selected by those having the highest benefit, the plot of the final selected rules is:

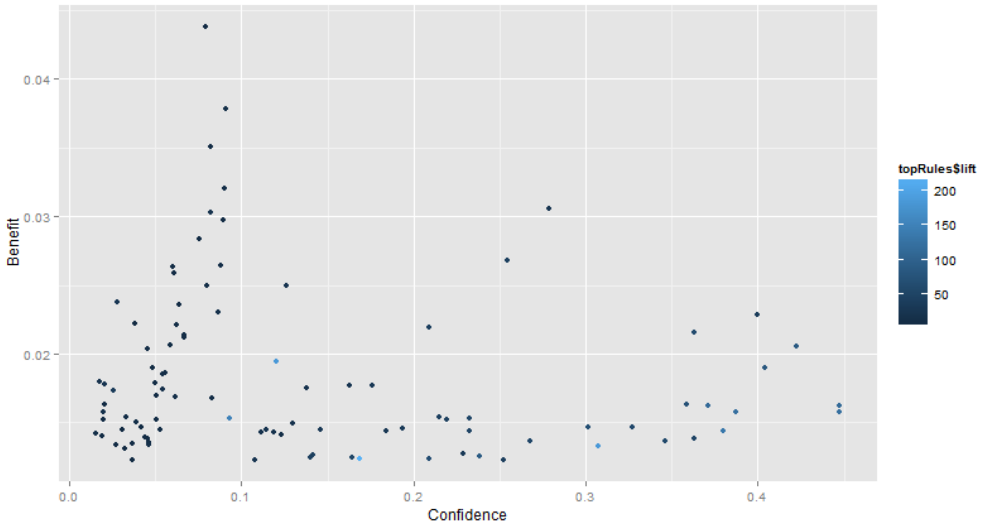


Figure 6: Plot of Final Selected Rules

Interestingly most of the rules have low confidence, below 20%, which means that there's a large scope for gain when the rules will be applied. Furthermore the lift also increases with confidence again with the benefit remaining constant, implying again that the rare transactions are not necessarily for expensive items.

In this case this is extremely beneficial – given that the set of rules has the highest benefit while still having low confidence and relatively low lift, the associated transactions have high scope for improvement while at the same time being common transactions. Appendix 1 contains the full set of rules with details.

Conclusion

By accurately sampling data, performing market basket analysis with the apriori algorithm, and selecting the best association rules with a weighted combination of transaction frequency and item revenue, a set of some of the most profitable product placements has been generated. A simple implementation of the association rules would be to place two products right next to each other on the shelves to boost sales. By measuring the increase in sales over time, one can quantify the success of the association rules. Given that the set of rules has been generated by sampling the variability of transactions, a good strategy would be to start by implementing the rules in the single stores that they were generated, and then over time rolling out to more stores in the chain based on success.

Rohan Agarwal

Appendix: Top 100 Rules

ruleLHS	ruleRHS	support	confidence	lift	departmentLeft	departmentRight	price	metric	store
4628597	4108011	0.002087	0.079279	7.237961	800	800	21	0.043824	604
4628597	4108011	0.001804	0.090833	14.93049	800	800	21	0.037874	5002
4628597	3524026	0.001594	0.082202	10.77224	800	800	22	0.035065	204
4628597	4108011	0.001524	0.090185	14.09297	800	800	21	0.032001	502
3978011	3524026	0.001389	0.278497	25.20581	800	800	22	0.030562	2209
4628597	4108011	0.001443	0.081883	22.58575	800	800	21	0.03031	6402
4628597	4108011	0.001417	0.089355	10.7985	800	800	21	0.029758	4804
4628597	4108011	0.001349	0.075234	8.187684	800	800	21	0.028336	2804
3978011	3524026	0.00122	0.254529	43.58937	800	800	22	0.026839	502
4628597	4108011	0.00126	0.087822	12.55484	800	800	21	0.026464	4407
4628597	3524026	0.001197	0.060276	10.64978	800	800	22	0.02633	5002
4628597	4108011	0.001232	0.060754	10.22071	800	800	21	0.025872	4604
3524026	3978011	0.001389	0.125731	25.20581	800	800	18	0.025005	2209
4628597	4108011	0.001187	0.079984	13.04335	800	800	21	0.024928	8702
4628597	5618966	0.000458	0.028128	28.39222	800	800	52	0.023804	3202
4628597	3524026	0.001072	0.063464	10.86845	800	800	22	0.023591	502
4628597	4108011	0.001096	0.086258	18.63281	800	800	21	0.023007	7307
3898011	3524026	0.00104	0.399504	36.15766	800	800	22	0.022886	2209
4628597	3524026	0.00101	0.038381	6.229498	800	800	22	0.022227	604
4628597	4108011	0.001051	0.062602	11.5476	800	800	21	0.022078	6204
3524026	3978011	0.00122	0.208922	43.58937	800	800	18	0.021959	502
3898011	3524026	0.000981	0.362761	62.12461	800	800	22	0.021586	502
4628597	4108011	0.001017	0.066634	11.34087	800	800	21	0.021361	3204
4628597	4108011	0.001009	0.066414	9.740596	800	800	21	0.021192	3704
4628597	3524026	0.000938	0.05843	11.87639	800	800	22	0.02063	2907
3898011	3978011	0.001142	0.422151	88.07735	800	800	18	0.020552	502
4628597	3524026	0.000926	0.045676	10.94665	800	800	22	0.020378	4604
8963391	5772500	0.000201	0.120499	184.4323	2200	2200	97	0.019483	2707
4628597	3524026	0.000863	0.048101	7.689231	800	800	22	0.018979	2804
3898011	3978011	0.001053	0.404467	81.08506	800	800	18	0.018958	2209
4628597	3524026	0.000846	0.055435	13.31826	800	800	22	0.018606	3602
4628597	4108011	0.000881	0.054156	12.424	800	800	21	0.018509	3202
4628597	5618966	0.000346	0.017836	20.27721	800	800	52	0.017983	204
4628597	3524026	0.000813	0.049958	11.98698	800	800	22	0.017887	3202
4628597	5618966	0.000343	0.020298	24.47859	800	800	52	0.017835	502
3978011	3524026	0.000806	0.176241	31.87244	800	800	22	0.017743	909
3978011	3524026	0.000804	0.162844	23.1464	800	800	22	0.017696	2004
3978011	3524026	0.000795	0.137788	19.29174	800	800	22	0.017484	1403
3978011	3524026	0.000795	0.137788	19.29174	800	800	22	0.017484	1403
4628597	4108011	0.000829	0.054348	18.00735	800	800	21	0.017412	3602
4628597	5618966	0.000333	0.025392	36.99082	800	800	52	0.017338	9202

4628597	3524026	0.00077	0.050472	10.6638	800	800	22	0.016951	3204
4628597	4108011	0.000804	0.06124	15.15758	800	800	21	0.016886	9202
4628597	4108011	0.000801	0.082474	13.36779	800	800	21	0.016826	3902
3898011	3978011	0.000906	0.358744	72.62764	800	800	18	0.016314	2004
4628597	9073382	0.000326	0.020288	9.017337	800	800	50	0.01628	2907
3898011	3968011	0.001472	0.447525	112.7531	800	800	11	0.016189	2907
3968011	3898011	0.001472	0.370796	112.7531	800	800	11	0.016189	2907
3898011	3968011	0.001434	0.447729	120.8319	800	800	11	0.01577	2707
3968011	3898011	0.001434	0.386916	120.8319	800	800	11	0.01577	2707
4628597	5618966	0.000303	0.019959	19.2783	800	800	52	0.01577	3704
3978011	3524026	0.000701	0.214506	43.34533	800	800	22	0.015426	509
4628597	264715	0.000497	0.032691	12.18743	800	800	31	0.015399	3704
8963391	656219	0.000155	0.092798	149.9837	2200	2200	99	0.015313	2707
3978011	3524026	0.000694	0.232143	55.01125	800	800	22	0.015277	3902
4628597	4108011	0.000727	0.050602	11.75875	800	800	21	0.015264	9704
4628597	3582465	0.000339	0.020041	10.73542	800	800	45	0.015238	502
3978011	3524026	0.00069	0.219008	44.5155	800	800	22	0.015186	2907
4628597	3524026	0.000682	0.038685	16.05598	800	800	22	0.015002	6402
3978011	3524026	0.000681	0.12963	20.72222	800	800	22	0.014989	2804
3898011	3978011	0.000817	0.326711	59.59594	800	800	18	0.014707	604
8798636	2783996	0.000698	0.301639	69.58994	2200	2200	21	0.014666	3204
4628597	3524026	0.000665	0.041919	8.542605	800	800	22	0.014625	4804
3898011	3524026	0.000664	0.193506	34.99474	800	800	22	0.014606	909
3524026	3978011	0.000806	0.14585	31.87244	800	800	18	0.014517	909
4628597	4108011	0.000691	0.052451	11.12615	800	800	21	0.014511	8407
4628597	4108011	0.000691	0.052451	11.12615	800	800	21	0.014511	8407
4628597	3978011	0.000806	0.030621	5.585599	800	800	18	0.014509	604
3524026	3978011	0.000804	0.114332	23.1464	800	800	18	0.014479	2004
3978011	3524026	0.000655	0.183767	30.79656	800	800	22	0.014413	3109
3898011	3978011	0.000798	0.232468	50.80083	800	800	18	0.014356	909
8147564	3949538	0.000287	0.379562	134.544	800	800	50	0.014354	604
3978011	3524026	0.000651	0.118832	19.28728	800	800	22	0.014332	604
3524026	3978011	0.000795	0.11127	19.29174	800	800	18	0.014305	1403
3524026	3978011	0.000795	0.11127	19.29174	800	800	18	0.014305	1403
4628597	5618966	0.000273	0.015474	15.64997	800	800	52	0.014183	6402
3978011	4108011	0.000674	0.12286	11.21686	800	800	21	0.014144	604
4628597	3751221	0.000326	0.01927	10.29857	800	800	43	0.014001	502
4628597	3524026	0.000635	0.044177	10.57751	800	800	22	0.01396	9704
8798636	5528349	0.000661	0.363363	51.39557	2200	2200	21	0.013872	3109
4628597	3978011	0.000768	0.045478	9.488487	800	800	18	0.013832	502
3898011	3978011	0.000761	0.346273	60.03593	800	800	18	0.013691	1403
3898011	3978011	0.000761	0.346273	60.03593	800	800	18	0.013691	1403
3898011	3524026	0.00062	0.267974	54.14957	800	800	22	0.013651	509
4628597	3524026	0.000619	0.046167	11.70809	800	800	22	0.013612	2707
4628597	3524026	0.000613	0.036473	10.20344	800	800	22	0.013475	6204

4628597	3524026	0.000608	0.046303	16.6847	800	800	22	0.013376	9202
3524026	3582465	0.000297	0.026901	10.81376	800	800	45	0.013375	2209
5772500	8963391	0.000201	0.30742	184.4323	2200	2200	66	0.013256	2707
4628597	4108011	0.000624	0.032183	6.762552	800	800	21	0.013104	204
3898011	3524026	0.000578	0.2287	32.50701	800	800	22	0.012711	2004
3524026	3978011	0.000701	0.141692	43.34533	800	800	18	0.012622	509
3978011	3898011	0.001142	0.238225	88.07735	800	800	11	0.01256	502
3524026	3978011	0.000694	0.164557	55.01125	800	800	18	0.012499	3902
3524026	3978011	0.00069	0.140304	44.5155	800	800	18	0.012425	2907
3898011	3978011	0.000687	0.208911	66.28281	800	800	18	0.012366	2907
8963391	656219	0.000125	0.168539	218.0518	2200	2200	99	0.012342	5002
3978011	3524026	0.00056	0.108059	24.33674	800	800	22	0.012315	3403
4628597	3524026	0.000559	0.03682	9.580811	800	800	22	0.012308	3704
8798636	5528349	0.000585	0.252459	42.17779	2200	2200	21	0.012275	3204