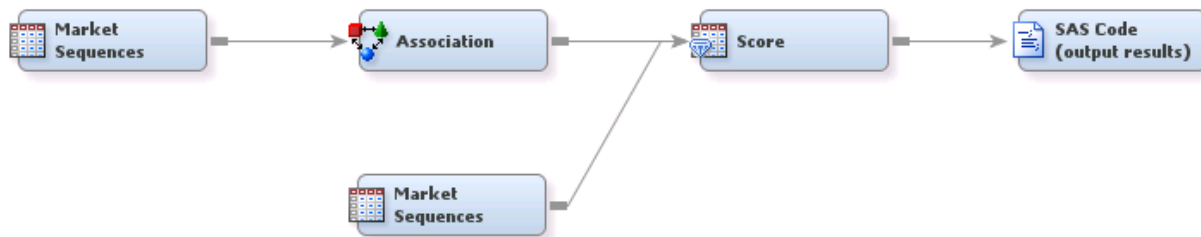# Association Discovery Using SAS® Enterprise Miner™

Brief Description of Association Discovery:

Association discovery, also known as market basket analysis, is the identification of items that occur together in a given event or record. The databases used for online transaction processing systems often provide the data sources for association discovery. Association discovery rules are based on the counts of the number of times items occur alone and in combination in the transaction records. Associations can be written in the form A➔B, where A (the left hand side) is called the antecedent and B (the right hand side) is called the consequent. Both sides of an association can contain more than one item. Identifying creditable associations between one or more items can help the business analyst make decisions such as when to distribute coupons, when to put a product on sale, or how to present items in store displays.

Data:

The MSEQ data set in the SAS library SAMPSIO is used to create the data source. This data set includes three variables (ACTION, CUSTOMER, and TIME) which take the following roles:

- ACTION: Target
- CUSTOMER: ID
- TIME: Sequence (not used in this analysis)

SAMPSIO.MSEQ2

	ACTION	CUSTOMER	TIME
1	new_car	1.0	109.0
2	new_home	1.0	109.0
3	baby	1.0	35.0
4	checking	1.0	337.0
5	close_loan	4.0	358.0
6	IRA	4.0	249.0
7	safe_deposit	4.0	166.0
8	open_loan	5.0	343.0
9	new_home	5.0	116.0
10	IRA	5.0	343.0
11	safe_deposit	5.0	170.0
12	money_market	5.0	170.0

Goal:

The goal is to identify the association between different actions by creating rules. These rules will then be used to make recommendations (to predict future actions) for each customer.

Market Sequences Node:

The data source is created by SAMPSIO.MSEQ and renamed Market Sequences.

Association Node:

The following properties are changed in the Association node.

Variables:

- The TIME variable is not used in the analysis.
  If the order of events is important for the analysis, you can take the action times into account. To make this change, click ⬚ for the Variables property, and set the **Use** column to Yes for the TIME variable.

Association:

- **Maximum Items property**: 2 (indicates the maximum size of the item set to be considered in an association)
- **Minimum Confidence Level:** 50 (specifies the minimum confidence level to be used to generate a rule)
- **Support Percentage:** 10 (specifies the minimum transaction frequency to support an association)

Rules:

- **Export Rule by ID:** Yes
- **Recommendation:** Yes

After you run the Association node, you can view its results by right-clicking the node and selecting **Results**. In the Results window, you can view the **Rules Table** by selecting **View >> Rules >> Rules Table**. The **Rules Table** (shown below) contains all the created rules along with the related statistics.

Rules Table

Relations	Expected Confidence(%)	Confidence(%)	Support(%)	Lift	Transaction Count	Rule	Left Hand of Rule	Right Hand of Rule	Rule Item 1	Rule Item 2	Rule Item 3	Rule Index	Transpose Rule
2	9.49	61.90	9.49	6.52	13.00	mutual_fun...	mutual_fund	new_job	mutual_fund	=========...	new_job	1	1
2	8.03	52.38	8.03	6.52	11.00	marriage =...	marriage	CD	marriage	=========...	CD	2	1
2	15.33	100.00	9.49	6.52	13.00	new_job ==...	new_job	mutual_fund	new_job	=========...	mutual_fund	3	1
2	15.33	100.00	8.03	6.52	11.00	CD ==> ma...	CD	marriage	CD	=========...	marriage	4	1
2	16.79	100.00	8.76	5.96	12.00	graduate =...	graduate	close_loan	graduate	=========...	close_loan	5	1
2	8.76	52.17	8.76	5.96	12.00	close_loan ...	close_loan	graduate	close_loan	=========...	graduate	6	1
2	13.87	67.86	13.87	4.89	19.00	open_loan ...	open_loan	savings	open_loan	=========...	savings	7	1
2	20.44	100.00	13.87	4.89	19.00	savings ==...	savings	open_loan	savings	=========...	open_loan	8	1
2	27.74	100.00	15.33	3.61	21.00	mutual_fun...	mutual_fund	credit_card	mutual_fund	=========...	credit_card	9	1
2	15.33	55.26	15.33	3.61	21.00	credit_card ...	credit_card	mutual_fund	credit_card	=========...	mutual_fund	10	1
2	27.74	100.00	9.49	3.61	13.00	new_job ==...	new_job	credit_card	new_job	=========...	credit_card	11	1
2	31.39	100.00	20.44	3.19	28.00	open_loan ...	open_loan	money_ma...	open_loan	=========...	money_ma...	12	1
2	20.44	65.12	20.44	3.19	28.00	money_ma...	money_ma...	open_loan	money_ma...	=========...	open_loan	13	1
2	31.39	100.00	13.87	3.19	19.00	savings ==...	savings	money_ma...	savings	=========...	money_ma...	14	1

Notice that Transpose Rule the (last column in the Rules Table) contains a value of 1 for all the rules. This value implies that all the rules will be used for recommendations. If you are not interested in all the rules, then you can interactively choose the rules by performing the following steps:

- Close the **Results** window.
- Click ⬛ for the **Rules** property (the third row under **Train** properties) of the Association node to open the Rules Selector table.
- Highlight all the rows in the table and set the Transpose Value to NO to indicate that none of the created rules are desired.
- Click the **Right Hand of Rule** column name to alphabetize the list of consequents.
- Highlight the rules whose Right Hand of Rule is either new_home or open_loan and set the Transpose Value to YES to indicate them as the desired rules.
- Click **OK** to continue. Now you can view the updated results by right-clicking the Association node and selecting **Results**.

Score Node:
The Score node uses the model created by the Association node to score on existing data. Here the Score node uses the rules created by the Association Node to recommend items to the customers. In the output data set of the score node, columns are binary variables for each rule and the rows represent customers. For each rule, a customer is assigned a recommendation value of 1 or 0. If a customer already has both the antecedent and the consequent of a rule, then the corresponding rule variable takes a value of 0 (rule not recommended). However, if the antecedent of a rule exists, but the consequent does not, then the rule variable takes a value of 1 (rule recommended).

SAS Code Node:
The SAS Code Node enables you to incorporate your SAS code into the SAS Enterprise Miner process flow diagrams. SAS Code node simplifies the output data set generated by the score code and yields the following table:

CUSTOMER	RULE_LABEL	RULE_ID
1	new_home ==> open_loan	RULE26
6	money_market ==> open_loan	RULE13
6	new_home ==> open_loan	RULE26
19	new_home ==> open_loan	RULE26
21	money_market ==> open_loan	RULE13
21	new_home ==> open_loan	RULE26
23	money_market ==> open_loan	RULE13
23	new_home ==> open_loan	RULE26
25	new_home ==> open_loan	RULE26
27	money_market ==> open_loan	RULE13
27	new_home ==> open_loan	RULE26

This SAS Code node can retain its functionality for similar examples. However, if you change the default value for the **Number to Keep** property of the Association node under **Rules,** you need to change it in the SAS Code node too. Click ⊡ for the **Code Editor** property of the SAS Code node, and edit the first line, '%let NUM=200', which specifies the maximum number of rules to keep (default for the Association node is 200).

Alternative Considerations:

The Association node also enables you to perform sequence discovery. Sequence discovery goes one step further than association discovery by taking into account the time of the actions. For example, a hypothetical sequence rule for this analysis could be "25 % of the customers who have a new baby will buy a new car in the next month". You can do sequence discovery for this analysis by using the TIME variable.

You could also perform this analysis by using the Market Basket node. This node does not enable you to do sequence discovery, but it can use the taxonomy data to generate rules at multiple levels. For more information about the Market Basket node see the relevant section available in SAS Enterprise Miner.