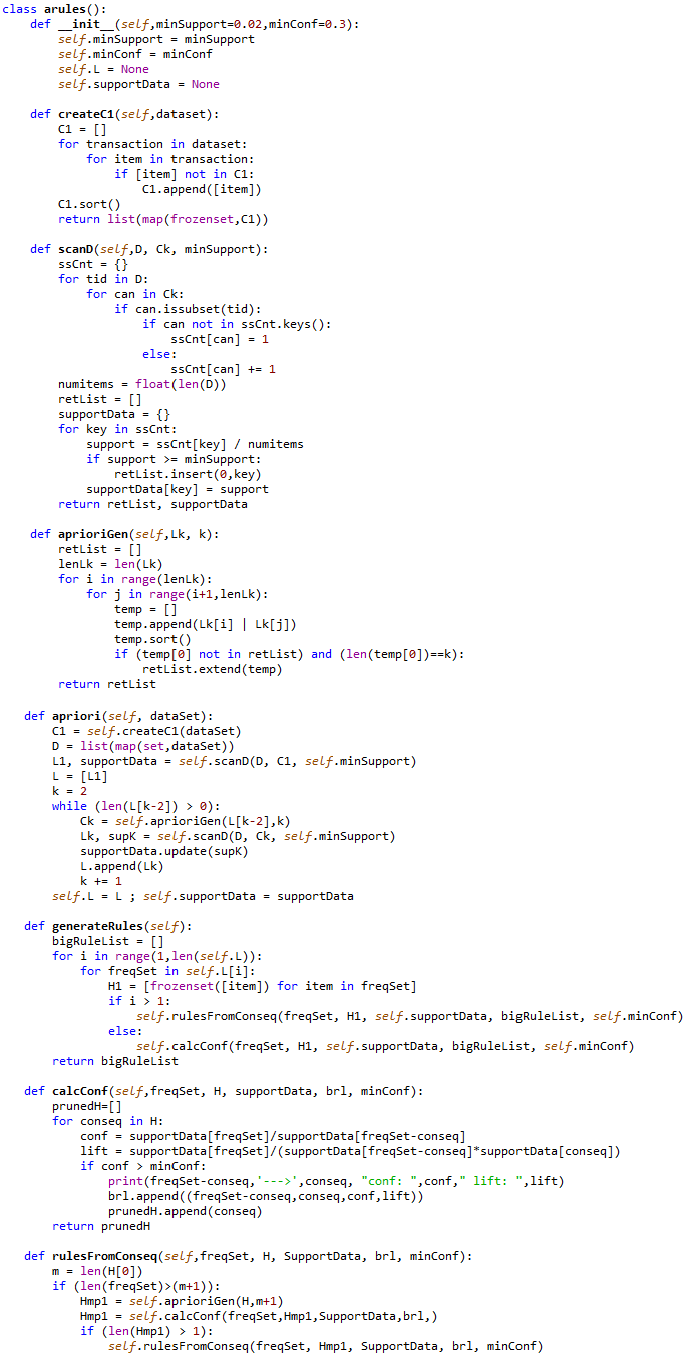
About a month ago, my parents and I had a heated discussion on possible ways to boost sales at their restaurant. The usual and somewhat mundane tactics were brought up including a lunch special, loyalty program, a combo menu and etc. Among the possible candidates, we saw eye to eye on a quickest and simplest way being creating a combo menu, using existing food items. My parents had a seemingly good idea on what items to include in a combo menu and how much to charge based on their years of experiences and intuition. I was not mindlessly against their idea but wondering if it would be the optimal decision to bundle item A and B in the combo menu with 15 % discounts as they initially thought. Specifically, I was wondering why they chose item A and B instead of C and D and why discounts should be 15% rather than 5%, 10%, or 20%.

The idea of a combo menu immediately reminded me of a popular method called ‘Association Rule’, aka. ‘Market Basket Analysis’. This method is widely known as the ‘beer and diaper’ story where a supermarket discovered that male shoppers who bought diapers had high propensity to also buy beer. I decided to use the association rule to discover regularities in customers’ buying patterns to validate my parents’ initial thoughts on bundling specific items. The general approach and the steps of this method are as follows:

1. The input to the model is in the form of transactional data from POS (Point of Sales) which is usually comprised of two variables, transaction id (invoice number) and items being purchased under each transaction id.
2. Using this input, a set of frequent items is being generated based on a measure called ‘Support’ which is merely the number of transactions that have a specific item divided by the total transaction. For example, assuming that the input dataset consists of 100 transactions (invoices) and that an item A appears on only 65 invoices, the ‘Support’ of the item A comes out to be 0.65, 65 divided by 100.
3. Any sets of items that have lower Support than the minimum threshold set by the user are not included in the frequent item list. The logic behind it is to ignore a set of items that only appear few times.
4. Based on the set of frequent items generated from the previous steps, a measure called ‘Confidence’ is computed for each permutation of the set of frequent items. For example, assuming that the set of frequent items contains only item A and B, a confidence of item A with respect to Item B is computed by the number of transactions that contain both item A and B divided by the number of transactions that only contains item A. Conversely, a confidence of item B with respect to Item A is also computed by the number of transactions that contain both item A and B divided by the number of transactions that only contains item B.
5. Rules are established if a confidence computed in the previous step is greater than the minimum confidence threshold set by the user. For example, if the minimum confidence threshold is 0.5 and the confidence of item A with respect to item B (computed in step 4) is 0.65, it gets included in the final association rules as its confidence of 0.65 is greater than the minimum threshold of 0.5. The confidence of 0.65 of this rule indicates that if a customer purchased item A, there is 65% probability that the same customer would purchase item B.

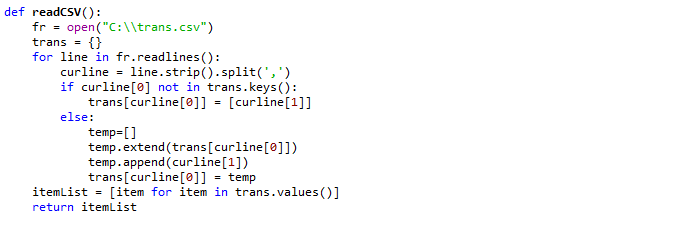
For those of who are interested in nuts and bolts of this method, below is the complete Python codes for association rules using the Apriori algorithm.



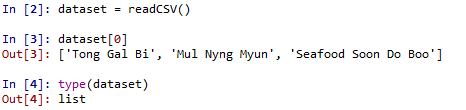
Before running the above model to find assocaiton rules, a proper format of the data should be fed into the model. The initial format of the dataset (which has been already wangled using R) has two columns, transaction id (invoice number) and items being purchased under each transaction id as shown below.



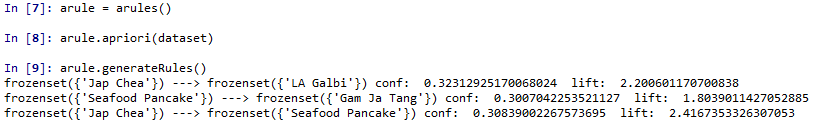
To read and convert the dataset into a proper format for the model, the code below has been written and run.



Now, the dataset has been read and converted to the proper list format as shown below.



Then, an instance of the arule class is created and the dataset is fed into the instance to create frequent items and then establish association rules. With a minimum support of 0.02 and a minimum confidence of 0.3, three rules have been generated as shown below.



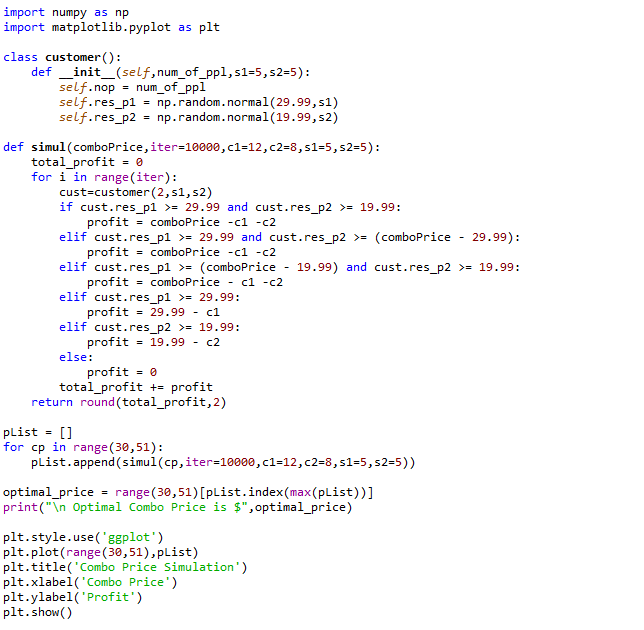
* Rule 1: Customers who order ‘Jap Chea’ have a 32 percent (confidence) chance to also order ‘LA Galbi’
* Rule 2: Customers who order ‘Seafood Pancake’ have a 30 percent (confidence) chance to also order ‘Gam Ja Tang’
* Rule 3: Customers who order ‘Jap Chea’ have a 31 percent (confidence) chance to also order ‘Seafood Pancake’

Among the three rules, the 2nd rule was actually my parents initially thought they would bundle together as a combo menu. It seems to be a manifest outcome given there are only about 60 items in the restaurant and my parents know their business like the back of their hands. Regardless, it was worth confirming their idea using a more systematic approach and data. Also it will be much more helpful to find hidden patterns if there are hundreds or thousands of items like a grocery store.

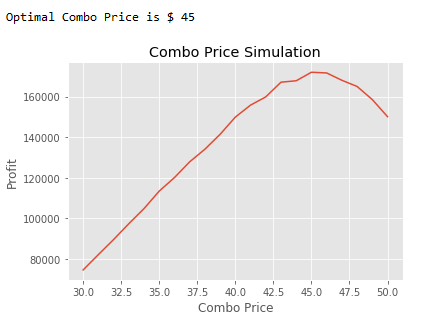
Now that I have confirmed my first question: which items to bundle, the remaining question is how much to charge for that bundle. Given the current price of ‘Seafood Pancake ($19.99) and ‘Gam Ja Tang’ ($29.99), would the 15% discounts on the bundle (or $42.5) that my parent initially had in mind be the optimal decision or should it be something else? To answer this question, a few assumptions were made and a simulation was performed to find the optimal bundling price that could yield the most profits to the business.

1. Customers’ reservation price for each item follows the normal distribution with the mean of the current price and standard deviation of $5
2. Customer would purchase the combo if their reservation prices for both items are greater than the individual price. For example, if a customer was willing to pay up to $32 for the $29.99 item and up to $22 for the $19.99 item, a reasonable customer would purchase the combo.
3. If a customer’s reservation price for one item is greater than the individual price but less for the other item, the customer would select the combo only if the difference between the bundling price and the individual price of the item with the higher reservation price is greater than the reservation price of the other item. For example, if a customer was willing to pay up to $32 for the $29.99 item but only up to $15 for the $19.99 item, a reasonable customer would purchase the combo only if the difference between the bundling price and $29.99 is lower than $15 (the reservation price of the $19.99 item).
4. Otherwise, a customer would only purchase either of the two items or none based on their reservation prices.

The simple Python implementation for this simple simulation is as follows.



The final outcome of the simulation based on 10,000 interations is shown below.



The above simulation chart depicts that the profit is maximized when the bundling price is at $45, which is equated to about a 10% discount of the total value ($29.99 + $19.99). Other simulations with different standard deviations on the reservation price resulted in the optimal price revolving around $45 as well. Given the psychological pricing where the impact of each digit of a price on customer’s perception diminishes as the digit moves from left to right (pricing that ends in 9 or 99), the final bundling price I would suggest comes out to be $44.99 which differs from the initial intuition of the management of $42.5.

In summary, a method called ‘Association Rules’ has been used to identify items that are likely to be purchased together at a restaurant and then the model result has validated the owner’s decision on which items to bundle for a special combo menu. Subsequently, a simulation method has been used to identify the optimal combo price to maximize profits. The final bundling price (about 10% discount) suggested by the model has come out to be different from what the owner initially had in mind (15% discount). Should the management disregard their intuition and go with the price suggested by the model? Well, I would say ‘Yes’ but at the end of the day it will be the management’s decision on how much they are willing to utilize analytical findings in making a final call in conjunction with their intuition.