I have implemented Apriori algorithm with F1 and Fk-1. I converted categorical and continuous data sets into sparse matrix. For a categorical column with four attribute value, I created four columns in sparse matrix and assigned 1 to one of the attribute present in a data point and rest as 0. For continuous column I created 2 columns in sparse matrix with values less than equal to mean and greater than mean. I implemented Apriori initially with naïve method i.e. to check support for 3 different items sets (ab,bc,cd) my program had to traverse all transactions 3 times. As the complexity of that algorithm was very high then I again implemented Apriori with hashing thus my program only need to traverse transactions only once to find support of any number of item sets. Eventually my program became very fast.

Initially I picked these 3 data sets:

- Car Evaluation (1729 data points, 25 attributes)
- Nursery (12960 data points, 32 attributes)
- Mushrooms (8124 data points, 121 attributes)

Number of attributes and data points in Mushrooms made it difficult for to work with it that is why I picked 1 other data set:

- Abalone (4177 data points, 20 attributes)

But in the end my program worked well so here I am presenting results from majorly from 4 data sets excluding Mushrooms on some occasions as it takes a lot of time to present output.

(b)

After observing the below mentioned result I have understood that F(k-1) X F(k-1) method is efficient than F(k-1) X F(1) because it generates less number of frequent item set candidates. Savings are more significant at less support level (1%) as the value of k in k – itemsets goes high in case of less support.

Intuitively we can say that F(k-1) X F(k-1) just generates those candidates whose subsets of k-2 elements are already frequent but this is not the case with F(k-1) X F(1).

For example:

We already know that 1,2,3,4, (1,2),(1,3) are frequent.

F(k-1) X F(k-1): It will generate just (1,2,3)

But

F(k-1) X F(1) : It will generate (1,2,3), (1,2,4), (1,3,4). SO already a difference of two.

By observing result we can also say that with decease in support, savings tends to increase or Minimum support is inversely proportional to Savings.

Support 10%:

Data Set	F(k-1)F(1) Total Candidate	F(k-1)F(k-1)Total Candidate	Savings
Car Evaluation	393	350	43
Nursery	1167	926	241

Abalone	3467	1827	1640

Support 5%:

Data Set	F(k-1)F(1) Total Candidate	F(k-1)F(k-1)Total Candidate	Savings
Car Evaluation	1796	1644	152
Nursery	3998	3615	383
Abalone	5178	3123	2055

Support 1%:

Data Set	F(k-1)F(1) Total Candidate	F(k-1)F(k-1)Total Candidate	Savings
Car Evaluation	11803	6956	4847
Nursery	42770	29784	12986
Abalone	11388	8814	2574

(c) :

I have found that

Frequent Sets > Frequent Closed Item Sets > Maximal Frequent Item Sets, in all of the cases. After giving it a little more thought I can say that:

Frequent Sets >= Frequent Closed Item Sets >= Maximal Frequent Item Sets.

By observing the data we can also say that Number of all the above mentioned parameters in this question tends to increase with decrease in the support. Or in other words, Minimum Support is inversely proportional to number of frequent item sets, Frequent Closed and Maximal Frequent Item Sets.

Support 10%:

Data Sets		Frequent Item Sets	Frequent Closed Item Sets	Maximal Frequent Item Sets
Car Evaluation	F(k-1)F(1)	86	73	43
	F(k-1)F(k-1)	86	73	43
Nursery	F(k-1)F(1)	169	145	115
	F(k-1)F(k-1)	169	145	115
Abalone	F(k-1)F(1)	1365	1197	47
	F(k-1)F(k-1)	1365	1197	47

Support 5%:

Data Sets		Frequent Item Sets	Frequent Closed Item Sets	Maximal Frequent Item Sets
Car Evaluation	F(k-1)F(1)	349	309	195
	F(k-1)F(k-1)	349	309	195
Nursery	F(k-1)F(1)	623	537	377
	F(k-1)F(k-1)	623	537	377
Abalone	F(k-1)F(1)	2246	1905	59
	F(k-1)F(k-1)	2246	1905	59

Support 1%:

Data Sets		Frequent Item Sets	Frequent Closed Item Sets	Maximal Frequent Item Sets
Car Evaluation	F(k-1)F(1)	2291	1950	1000
	F(k-1)F(k-1)	2291	1950	1000
Nursery	F(k-1)F(1)	8094	6800	3781
	F(k-1)F(k-1)	8094	6800	3781
Abalone	F(k-1)F(1)	5361	4420	336
	F(k-1)F(k-1)	5361	4420	336

(d) Confidence base pruning Savings:

From the below mentioned results I have observed that:

Minimum support is inversely proportional to savings and confidence is directly proportional to Savings.

Thus we can do a lot of savings with confidence based pruning.

Support 10% and Confidence 90%:

Data Sets	Brute Force	Generated Association Rules	Savings
Car Evaluation	170	13	157
Nursery	326	24	302
Abalone	42608	12568	30040

Support 10% and Confidence 80%:

Data Sets	Brute Force	Generated Association Rules	Savings
Car Evaluation	170	15	155
Nursery	326	24	302
Abalone	42608	20296	22312

Support 10% and Confidence 60%:

Data Sets	Brute Force	Generated Association Rules	Savings
Car Evaluation	170	30	140
Nursery	326	26	300
Abalone	42608	25356	17252

Support 5% and Confidence 90%:

Data Sets	Brute Force	Generated Association Rules	Savings
Car Evaluation	1076	42	1034
Nursery	2314	88	2226
Abalone	90850	19085	71765

Support 5% and Confidence 80%:

Data Sets	Brute Force	Generated Association Rules	Savings
Car Evaluation	1076	42	1034
Nursery	2314	90	2224
Abalone	90850	31086	59764

Support 5% and Confidence 60%:

Data Sets	Brute Force	Generated Association Rules	Savings
Car Evaluation	1076	112	964
Nursery	2314	117	2197
Abalone	90850	40541	50309

Support 1% and Confidence 90%:

Data Sets	Brute Force	Generated Association Rules	Savings
Car Evaluation	19164	350	18814
Nursery	93334	1354	91980
Abalone	216448	25221	191227

Support 1% and Confidence 80%:

Data Sets	Brute Force	Generated Association Rules	Savings
Car Evaluation	19164	411	18753
Nursery	93334	1428	91906
Abalone	216448	42838	173610

Support 1% and Confidence 60%:

Data Sets	Brute Force	Generated Association Rules	Savings
Car Evaluation	19164	783	18381
Nursery	93334	2175	91159
Abalone	216448	63421	153027

(e)

It was tedious task to fetch meaningful rules from a lot of association rules as many of the rules have confidence of 100%. To get as much knowledge as possible from rules, I finalized just 1 rule from a itemset instead of multiple rules from same item set. The rule I choose is of highest confidence.

For example:

We have a frequent item set as (a,b,c). SO this item set can create 8 rules. But I choose just 1 rule with highest confidence from this item set. Then after getting all the association rules, I arranged them in decreasing order of confidence and then just selected top 15 rules from the list.

I also took care to include the best rule in terms of number of attributes present in consequent. For example if we have rules:

a->b

a-> b,c

a->b,c,d

Then I selected the rule a->b,c,d and discarded others to present the maximum knowledge of data in top 10 association rules.

Going forward I selected top 10 rules from this list of top 15 confident rules. To select top 10 I used support or lift. For example if we are working with support and confidence then I finalized top 10 rules using support count or If we are working with lift then I used lift to select top 10 rules.

I even selected the support value to be very low (1%) to get rules for classes which are very less if their confidence is high.

Data Set Car Evaluation:

Minimum Support 1% and Minimum Confidence 90%:

Top 10 Association rules

Support Confidence Association Rule

[576, 100.0, 'Safety_low ----> unacc ']

[576, 100.0, 'Persons_2 ----> unacc ']

[192, 100.0, 'LuggageBoot_big , Safety_low ----> unacc ']

[192, 100.0, 'Persons_2 , LuggageBoot_med ----> unacc ']

[192, 100.0, 'Persons_more , Safety_low ----> unacc ']

[144, 100.0, 'BP_v-high , Persons_2 ----> unacc ']

[144, 100.0, 'Doors_5-more , Persons_2 ----> unacc ']

[144, 100.0, 'BP_low , Safety_low ----> unacc ']

[144, 100.0, 'MP_med , Persons_2 ----> unacc ']

[144, 100.0, 'MP_v-high , Persons_2 ----> unacc ']

Some rules makes sense:

As first rules says that If safety is low then car is very bad. Unacc= Very bad. Support count of the rule is 576 (25%) and confidence is 100%.

Second rule: If a car is for 2 persons then it is very bad.

As the Unacc class was present in most of the datapoints so all the rules have considered just that class. I tried to get some rules for classes which are less present by reducing support but it didn’t work.

Minimum Support 1% and Minimum Confidence 80%:

Top 10 Association rules

Support Confidence Association Rule

[576, 100.0, 'Safety_low ----> unacc ']

[576, 100.0, 'Persons_2 ----> unacc ']

[192, 100.0, 'LuggageBoot_big , Safety_low ----> unacc ']

[192, 100.0, 'Persons_2 , LuggageBoot_med ----> unacc ']

[192, 100.0, 'Persons_more , Safety_low ----> unacc ']

[144, 100.0, 'BP_v-high , Persons_2 ----> unacc ']

[144, 100.0, 'Doors_5-more , Persons_2 ----> unacc ']

[144, 100.0, 'BP_low , Safety_low ----> unacc ']

[144, 100.0, 'MP_med , Persons_2 ----> unacc ']

[144, 100.0, 'MP_v-high , Persons_2 ----> unacc ']

I am getting mostly same rules even with increasing confidence.

Minimum Support 1% and Minimum Confidence 60%:

Top 10 Association rules

Support Confidence Association Rule

[576, 100.0, 'Safety_low ----> unacc ']

[576, 100.0, 'Persons_2 ----> unacc ']

[192, 100.0, 'LuggageBoot_big , Safety_low ----> unacc ']

[192, 100.0, 'Persons_2 , LuggageBoot_med ----> unacc ']

[192, 100.0, 'Persons_more , Safety_low ----> unacc ']

[144, 100.0, 'BP_v-high , Persons_2 ----> unacc ']

[144, 100.0, 'Doors_5-more , Persons_2 ----> unacc ']

[144, 100.0, 'BP_low , Safety_low ----> unacc ']

[144, 100.0, 'MP_med , Persons_2 ----> unacc ']

[144, 100.0, 'MP_v-high , Persons_2 ----> unacc ']

Nursery Data Set:

In the second data set I got some interesting rules. This data is of a school admissions of nursery.

Minimum Support 1% and Minimum Confidence 90%:

Top 10 Association rules

Support Confidence Association Rule

[4320, 100.0, 'health_not_recom ----> class_not_recom ']

[2160, 100.0, 'finance_inconv , health_not_recom ----> class_not_recom ']

[2160, 100.0, 'finance_convenient , health_not_recom ----> class_not_recom ']

[1440, 100.0, 'housing_less_conv , health_not_recom ----> class_not_recom ']

[1440, 100.0, 'housing_critical , class_not_recom ----> health_not_recom ']

[1440, 100.0, 'housing_convenient , class_not_recom ----> health_not_recom ']

[1440, 100.0, 'parents_pretentious , health_not_recom ----> class_not_recom ']

[1080, 100.0, 'children_1 , health_not_recom ----> class_not_recom ']

[1080, 100.0, 'children_more , class_not_recom ----> health_not_recom ']

[1080, 100.0, 'form_completed , health_not_recom ----> class_not_recom ']

Most the rules are biased towards health not recommend as the support for same is very high.

- First rule with 100 % confidence and 4320 Support says:
  - If health of a child in not recommended then it is not recommended to give admission to that child to nursery as per data.
- If housing is less convenient or critical then in that case also admission is not recommended as is explained by 3rd and 4th rule.
- Mostly data is stating that is health of child is not recommended (not well) then that child is denied admission.

Minimum Support 1% and Minimum Confidence 80%:

Top 10 Association rules

Support Confidence Association Rule

[4320, 100.0, 'health_not_recom ----> class_not_recom ']

[2160, 100.0, 'finance_inconv , health_not_recom ----> class_not_recom ']

[2160, 100.0, 'finance_convenient , health_not_recom ----> class_not_recom ']

[1440, 100.0, 'housing_less_conv , health_not_recom ----> class_not_recom ']

[1440, 100.0, 'housing_critical , class_not_recom ----> health_not_recom ']

[1440, 100.0, 'housing_convenient , class_not_recom ----> health_not_recom ']

[1440, 100.0, 'parents_pretentious , health_not_recom ----> class_not_recom ']

[1080, 100.0, 'children_1 , health_not_recom ----> class_not_recom ']

[1080, 100.0, 'children_more , class_not_recom ----> health_not_recom ']

[1080, 100.0, 'form_completed , health_not_recom ----> class_not_recom ']

Again the results are dominated by health_not_recommeded attribute.

Minimum Support 1% and Minimum Confidence 60%:

Top 10 Association rules

Support Confidence Association Rule

[4320, 100.0, 'health_not_recom ----> class_not_recom ']

[2160, 100.0, 'finance_inconv , health_not_recom ----> class_not_recom ']

[2160, 100.0, 'finance_convenient , health_not_recom ----> class_not_recom ']

[1440, 100.0, 'housing_less_conv , health_not_recom ----> class_not_recom ']

[1440, 100.0, 'housing_critical , class_not_recom ----> health_not_recom ']

[1440, 100.0, 'housing_convenient , class_not_recom ----> health_not_recom ']

[1440, 100.0, 'parents_pretentious , health_not_recom ----> class_not_recom ']

[1080, 100.0, 'children_1 , health_not_recom ----> class_not_recom ']

[1080, 100.0, 'children_more , class_not_recom ----> health_not_recom ']

[1080, 100.0, 'form_completed , health_not_recom ----> class_not_recom ']

Abalone Data Set:

This data is predicting the age of abalone from measurements. I have divided the data using mean into less and equal to and greater than mean.

Minimum Support 1% and Minimum Confidence 90%:

Top 10 Association rules

Support Confidence Association Rule

[432, 100.0, 'Sex_M , Shucked_>0.359 , Rings_11-20 ----> Whole_>0.829 ']

[418, 100.0, 'Sex_F , Shucked_>0.359 , Rings_11-20 ----> Diam_>0.408 ']

[229, 100.0, 'Length_<=0.524 , Height_<=0.140 , Rings_11-20 ----> Shucked_<=0.359 ']

[224, 100.0, 'Diam_<=0.408 , Height_<=0.140 , Rings_11-20 ----> Shucked_<=0.359 ']

[147, 100.0, 'Sex_M , Whole_<=0.829 , Rings_11-20 ----> Shucked_<=0.359 ']

[69, 100.0, 'Sex_I , Length_<=0.524 , Rings_11-20 ----> Whole_<=0.829 ']

[69, 100.0, 'Sex_I , Length_<=0.524 , Rings_11-20 ----> Viscera_<=0.181 ']

[68, 100.0, 'Sex_I , Diam_<=0.408 , Rings_11-20 ----> Whole_<=0.829 ']

[60, 100.0, 'Sex_I , Whole_>0.829 , Rings_11-20 ----> Length_>0.524 ']

[60, 100.0, 'Sex_I , Whole_>0.829 , Rings_11-20 ----> Diam_>0.408 ']

Top rule says that if sex is male and shucked_length > 0.359 and belongs to Ring class of 11-20 then whole weight is > 0.829 grams with support 423 and confidence of 100%.

Similarly 2nd rule says if sex if female and shucked_length > 0.359 and ring class is 11-20 then diameter of abalobe is > 0.408

Minimum Support 1% and Minimum Confidence 80%:

Top 10 Association rules

Support Confidence Association Rule

[432, 100.0, 'Sex_M , Shucked_>0.359 , Rings_11-20 ----> Whole_>0.829 ']

[229, 100.0, 'Length_<=0.524 , Height_<=0.140 , Rings_11-20 ----> Shucked_<=0.359 ']

[224, 100.0, 'Diam_<=0.408 , Height_<=0.140 , Rings_11-20 ----> Shucked_<=0.359 ']

[147, 100.0, 'Sex_M , Whole_<=0.829 , Rings_11-20 ----> Shucked_<=0.359 ']

[114, 100.0, 'Sex_F , Diam_<=0.408 , Rings_11-20 ----> Shucked_<=0.359 ']

[69, 100.0, 'Sex_I , Length_<=0.524 , Rings_11-20 ----> Whole_<=0.829 ']

[69, 100.0, 'Sex_I , Length_<=0.524 , Rings_11-20 ----> Viscera_<=0.181 ']

[68, 100.0, 'Sex_I , Diam_<=0.408 , Rings_11-20 ----> Whole_<=0.829 ']

[60, 100.0, 'Sex_I , Whole_>0.829 , Rings_11-20 ----> Length_>0.524 ']

[60, 100.0, 'Sex_I , Whole_>0.829 , Rings_11-20 ----> Diam_>0.408 ']

Minimum Support 1% and Minimum Confidence 60%:

Top 10 Association rules

Support Confidence Association Rule

[147, 100.0, 'Sex_M , Whole_<=0.829 , Rings_11-20 ----> Shucked_<=0.359 ']

[117, 99.15254237288136, 'Height_<=0.140 , Shucked_>0.359 , Shell_>0.239 ----> Diam_>0.408 ']

[117, 99.15254237288136, 'Height_<=0.140 , Shucked_>0.359 , Shell_>0.239 ----> Length_>0.524 , Diam_>0.408 ']

[114, 100.0, 'Sex_F , Diam_<=0.408 , Rings_11-20 ----> Shucked_<=0.359 ']

[69, 100.0, 'Sex_I , Length_<=0.524 , Rings_11-20 ----> Whole_<=0.829 ']

[68, 100.0, 'Sex_I , Diam_<=0.408 , Rings_11-20 ----> Whole_<=0.829 ']

[60, 100.0, 'Sex_I , Whole_>0.829 , Rings_11-20 ----> Length_>0.524 ']

[60, 100.0, 'Sex_I , Whole_>0.829 , Rings_11-20 ----> Diam_>0.408 ']

[59, 100.0, 'Sex_M , Whole_<=0.829 , Shell_>0.239 ----> Shucked_<=0.359 ']

[55, 100.0, 'Sex_I , Viscera__>0.181 , Rings_11-20 ----> Length_>0.524 ']

(f)

I used list values of 0.8, 1 and 1.5.

Lift value of 1 suggests that Antecedent and consequent is independent of each other while values greater than 1 suggests positive co-relation present in antecedent and consequent.

Car Evaluation Data Set:

Minimum Support 1% and Minimum Lift 1:

Top 10 Association rules

Lift Confidence Association Rule

[1.428099173553719, 100.0, 'Safety_low ----> unacc ']

[1.428099173553719, 100.0, 'Persons_2 ----> unacc ']

[1.428099173553719, 100.0, 'BP_v-high , Persons_2 ----> unacc ']

[1.428099173553719, 100.0, 'BP_high , MP_v-high ----> unacc ']

[1.428099173553719, 100.0, 'MP_v-high , Persons_2 ----> unacc ']

[1.428099173553719, 100.0, 'Persons_2 , LuggageBoot_big ----> unacc ']

[1.428099173553719, 100.0, 'Persons_2 , Safety_med ----> unacc ']

[1.428099173553719, 100.0, 'BP_v-high , MP_v-high ----> unacc ']

[1.428099173553719, 100.0, 'Doors_3 , Persons_2 ----> unacc ']

[1.428099173553719, 100.0, 'BP_v-high , MP_high ----> unacc ']

Minimum Support 1% and Minimum Lift 1.5:

Top 10 Association rules

Lift Confidence Association Rule

[3.375, 75.0, 'BP_high , Persons_4 , Safety_high ----> acc ']

[2.7692307692307696, 92.3076923076923, 'BP_med , Safety_high , unacc ----> Persons_2 ']

[2.7692307692307696, 92.3076923076923, 'MP_low , Safety_high , unacc ----> Persons_2 ']

[2.7692307692307696, 92.3076923076923, 'MP_med , Safety_high , unacc ----> Persons_2 ']

[2.7692307692307696, 92.3076923076923, 'BP_low , Safety_high , unacc ----> Persons_2 ']

[2.6666666666666665, 88.88888888888889, 'BP_low , Persons_4 , unacc ----> Safety_low ']

[2.4, 80.0, 'MP_low , Persons_4 , unacc ----> Safety_low ']

[2.4, 80.0, 'BP_med , Persons_4 , unacc ----> Safety_low ']

[2.4, 80.0, 'MP_med , Persons_4 , unacc ----> Safety_low ']

[2.4, 80.0, 'BP_low , Persons_more , unacc ----> Safety_low ']

Minimum Support 1% and Minimum Lift 0.8:

Top 10 Association rules

Lift Confidence Association Rule

[4.0, 100.0, 'MP_med , good ----> BP_low ']

[4.0, 100.0, 'BP_med , good ----> MP_low ']

[3.0, 100.0, 'MP_med , v-good ----> Safety_high ']

[3.0, 100.0, 'Doors_4 , v-good ----> Safety_high ']

[3.0, 100.0, 'Persons_more , v-good ----> Safety_high ']

[3.0, 100.0, 'BP_med , v-good ----> Safety_high ']

[3.0, 100.0, 'LuggageBoot_small , good ----> Safety_high ']

[3.0, 100.0, 'Persons_4 , v-good ----> Safety_high ']

[3.0, 100.0, 'LuggageBoot_big , v-good ----> Safety_high ']

[3.0, 100.0, 'MP_low , v-good ----> Safety_high ']

I believe that I have found some more relevant rules in case of list as stated above:

Rule 1: If the maintenance of car is medium and car is termed as good then buying price of car is low.

Rule 3: If the maintenence price is medium and class is v-good then safety of the car is high.

Data Set Nursery:

Minimum Support 1% and Minimum Lift 0.8:

Top 10 Association rules

Lift Confidence Association Rule

[3.0, 100.0, 'housing_less_conv , health_not_recom ----> class_not_recom ']

[3.0, 100.0, 'children_1 , class_not_recom ----> health_not_recom ']

[3.0, 100.0, 'finance_inconv , class_not_recom ----> health_not_recom ']

[3.0, 100.0, 'has_nurs_very_crit , class_not_recom ----> health_not_recom ']

[3.0, 100.0, 'children_more , health_not_recom ----> class_not_recom ']

[3.0, 100.0, 'finance_convenient , health_not_recom ----> class_not_recom ']

[3.0, 100.0, 'form_completed , class_not_recom ----> health_not_recom ']

[3.0, 100.0, 'has_nurs_less_proper , class_not_recom ----> health_not_recom ']

[3.0, 100.0, 'has_nurs_proper , class_very_recom ----> health_recommended ']

[3.0, 100.0, 'housing_convenient , class_not_recom ----> health_not_recom ']

Minimum Support 1% and Minimum Lift 1:

Top 10 Association rules

Lift   Confidence   Association Rule

[3.0, 100.0, 'health_not_recom ----> class_not_recom ']

[3.0, 100.0, 'housing_less_conv , health_not_recom ----> class_not_recom ']

[3.0, 100.0, 'form_completed , health_not_recom ----> class_not_recom ']

[3.0, 100.0, 'housing_critical , class_not_recom ----> health_not_recom ']

[3.0, 100.0, 'housing_convenient , class_not_recom ----> health_not_recom ']

[3.0, 100.0, 'parents_pretentious , health_not_recom ----> class_not_recom ']

[3.0, 100.0, 'form_foster , health_not_recom ----> class_not_recom ']

[3.0, 100.0, 'has_nurs_proper , class_spec_prior ----> parents_great_pret ']

[3.0, 100.0, 'social_problematic , class_not_recom ----> health_not_recom ']

[3.0, 100.0, 'form_incomplete , class_not_recom ----> health_not_recom ']

Minimum Support 1% and Minimum Lift 1.5:

Top 10 Association rules

Lift   Confidence   Association Rule

[3.0, 100.0, 'health_not_recom ----> class_not_recom ']

[3.0, 100.0, 'housing_less_conv , health_not_recom ----> class_not_recom ']

[3.0, 100.0, 'form_completed , health_not_recom ----> class_not_recom ']

[3.0, 100.0, 'housing_critical , class_not_recom ----> health_not_recom ']

[3.0, 100.0, 'housing_convenient , class_not_recom ----> health_not_recom ']

[3.0, 100.0, 'parents_pretentious , health_not_recom ----> class_not_recom ']

[3.0, 100.0, 'form_foster , health_not_recom ----> class_not_recom ']

[3.0, 100.0, 'has_nurs_proper , class_spec_prior ----> parents_great_pret ']

[3.0, 100.0, 'social_problematic , class_not_recom ----> health_not_recom ']

[3.0, 100.0, 'form_incomplete , class_not_recom ----> health_not_recom ']

Data Set Abalone:

Minimum Support 1% and Minimum Lift 1:

Top 10 Association rules

Lift Confidence Association Rule

[2.0905905905905904, 100.0, 'Sex_M , Shucked_>0.359 , Rings_11-20 ----> Whole_>0.829 ']

[2.0905905905905904, 100.0, 'Sex_M , Shucked_>0.359 , Shell_>0.239 ----> Whole_>0.829 ']

[2.0905905905905904, 100.0, 'Sex_M , Height_<=0.140 , Shucked_>0.359 , Shell_>0.239 ----> Whole_>0.829 ']

[1.916934373565856, 100.0, 'Sex_I , Length_<=0.524 , Rings_11-20 ----> Whole_<=0.829 ']

[1.916934373565856, 100.0, 'Sex_I , Height_<=0.140 , Shell_<=0.239 , Rings_11-20 ----> Whole_<=0.829 ']

[1.8630686886708296, 100.0, 'Sex_I , Whole_>0.829 , Rings_11-20 ----> Length_>0.524 , Diam_>0.408 ']

[1.8614081996434937, 100.0, 'Sex_M , Whole_<=0.829 , Rings_11-20 ----> Shucked_<=0.359 ']

[1.8614081996434937, 100.0, 'Length_<=0.524 , Whole_<=0.829 , Shell_>0.239 ----> Shucked_<=0.359 ']

[1.8614081996434937, 100.0, 'Sex_M , Whole_<=0.829 , Shell_>0.239 ----> Shucked_<=0.359 ']

[1.8614081996434937, 100.0, 'Length_<=0.524 , Height_<=0.140 , Rings_11-20 ----> Shucked_<=0.359 ']

Minimum Support 1% and Minimum Lift 0.8:

Top 10 Association rules

Lift Confidence Association Rule

[2.0905905905905904, 100.0, 'Sex_M , Shucked_>0.359 , Rings_11-20 ----> Whole_>0.829 ']

[2.0905905905905904, 100.0, 'Sex_M , Shucked_>0.359 , Shell_>0.239 ----> Whole_>0.829 ']

[2.0905905905905904, 100.0, 'Sex_M , Height_<=0.140 , Shucked_>0.359 , Shell_>0.239 ----> Whole_>0.829 ']

[1.916934373565856, 100.0, 'Sex_I , Length_<=0.524 , Rings_11-20 ----> Whole_<=0.829 ']

[1.916934373565856, 100.0, 'Sex_I , Height_<=0.140 , Shell_<=0.239 , Rings_11-20 ----> Whole_<=0.829 ']

[1.8630686886708296, 100.0, 'Sex_I , Whole_>0.829 , Rings_11-20 ----> Length_>0.524 , Diam_>0.408 ']

[1.8614081996434937, 100.0, 'Sex_M , Whole_<=0.829 , Rings_11-20 ----> Shucked_<=0.359 ']

[1.8614081996434937, 100.0, 'Length_<=0.524 , Whole_<=0.829 , Shell_>0.239 ----> Shucked_<=0.359 ']

[1.8614081996434937, 100.0, 'Sex_M , Whole_<=0.829 , Shell_>0.239 ----> Shucked_<=0.359 ']

[1.8614081996434937, 100.0, 'Length_<=0.524 , Height_<=0.140 , Rings_11-20 ----> Shucked_<=0.359 ']

Minimum Support 1% and Minimum Lift 1.5:

Top 10 Association rules

Lift   Confidence   Association Rule

[2.0905905905905904, 100.0, 'Sex_M , Shucked_>0.359 , Rings_11-20 ----> Whole_>0.829 ']

[2.0905905905905904, 100.0, 'Sex_M , Shucked_>0.359 , Shell_>0.239 ----> Whole_>0.829 ']

[2.0905905905905904, 100.0, 'Sex_M , Height_<=0.140 , Shucked_>0.359 , Shell_>0.239 ----> Whole_>0.829 ']

[1.916934373565856, 100.0, 'Sex_I , Length_<=0.524 , Rings_11-20 ----> Whole_<=0.829 ']

[1.916934373565856, 100.0, 'Sex_I , Height_<=0.140 , Shell_<=0.239 , Rings_11-20 ----> Whole_<=0.829 ']

[1.8630686886708296, 100.0, 'Sex_I , Whole_>0.829 , Rings_11-20 ----> Length_>0.524 , Diam_>0.408 ']

[1.8614081996434937, 100.0, 'Sex_M , Whole_<=0.829 , Rings_11-20 ----> Shucked_<=0.359 ']

[1.8614081996434937, 100.0, 'Length_<=0.524 , Whole_<=0.829 , Shell_>0.239 ----> Shucked_<=0.359 ']

[1.8614081996434937, 100.0, 'Sex_M , Whole_<=0.829 , Shell_>0.239 ----> Shucked_<=0.359 ']

[1.8614081996434937, 100.0, 'Length_<=0.524 , Height_<=0.140 , Rings_11-20 ----> Shucked_<=0.359 ']