MARKET BASKET ANALYSIS

2019510075 ALPTUĞ TOPALHAN
2019510049 SEMİH FURKAN KARAMAN

CONTENT

ABOUT PROJECT

Our project is a Market Basket Analysis project that aims to uncover relationships between products by examining users' consumption habits.

The project has been developed using Python, and the dataset has been obtained from Kaggle.

ABOUT DATASET

	BillNo	Itemname	Quantity	Date	Price	CustomerID	Country
0	536365	WHITE HANGING HEART T-LIGHT HOLDER	6.0	01.12.2010 08:26	2,55	17850.0	United Kingdom
1	536365	WHITE METAL LANTERN	6.0	01.12.2010 08:26	3,39	17850.0	United Kingdom
2	536365	CREAM CUPID HEARTS COAT HANGER	8.0	01.12.2010 08:26	2,75	17850.0	United Kingdom
3	536365	KNITTED UNION FLAG HOT WATER BOTTLE	6.0	01.12.2010 08:26	3,39	17850.0	United Kingdom
4	536365	RED WOOLLY HOTTIE WHITE HEART.	6.0	01.12.2010 08:26	3,39	17850.0	United Kingdom
...	...	...	...	...	...	...	...
388018	581587	PACK OF 20 SPACEBOY NAPKINS	12.0	09.12.2011 12:50	0,85	12680.0	France
388019	581587	CHILDREN'S APRON DOLLY GIRL	6.0	09.12.2011 12:50	2,1	12680.0	France
388020	581587	CHILDRENS CUTLERY DOLLY GIRL	4.0	09.12.2011	4,15	12680.0	France

ABOUT DATASET

```
data.shape
```

```
(388023, 7)
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 388023 entries, 0 to 388022
Data columns (total 7 columns):
 #   Column      Non-Null Count   Dtype  
---  ------      --------------   -----  
 0   BillNo      388023 non-null  int64  
 1   Itemname    388023 non-null  object 
 2   Quantity    386083 non-null  float64
 3   Date        388023 non-null  object 
 4   Price       387635 non-null  object 
 5   CustomerID  388023 non-null  float64
 6   Country     388023 non-null  object 
dtypes: float64(2), int64(1), object(4)
memory usage: 20.7+ MB
```

Null values were observed in the dataset, and it was noticed that the 'Price' column is of object type.

```
data.isnull().sum()
```

```
BillNo          0
Itemname        0
Quantity     1940
Date            0
Price         388
CustomerID      0
Country         0
dtype: int64
```

PREPROCESSING

CHANGING DATA TYPE

The 'Price' column was converted from the object type to the float type. In the original data, numbers were written using commas, and they were converted to periods.

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 388023 entries, 0 to 388022
Data columns (total 7 columns):
 #   Column      Non-Null Count   Dtype  
---  ------      --------------   -----  
 0   BillNo      388023 non-null  int64  
 1   Itemname    388023 non-null  object 
 2   Quantity    388023 non-null  float64
 3   Date        388023 non-null  object 
 4   Price       388023 non-null  float64
 5   CustomerID  388023 non-null  float64
 6   Country     388023 non-null  object 
```

PREPROCESSING

HANDLING WITH NULL VALUES

The columns 'Price' and 'Quantity,' which contained null values, were filled with median values.

```
data.isnull().sum()
```

```
BillNo          0
Itemname        0
Quantity        0
Date            0
Price           0
CustomerID      0
Country         0
dtype: int64
```

VISUALIZATION

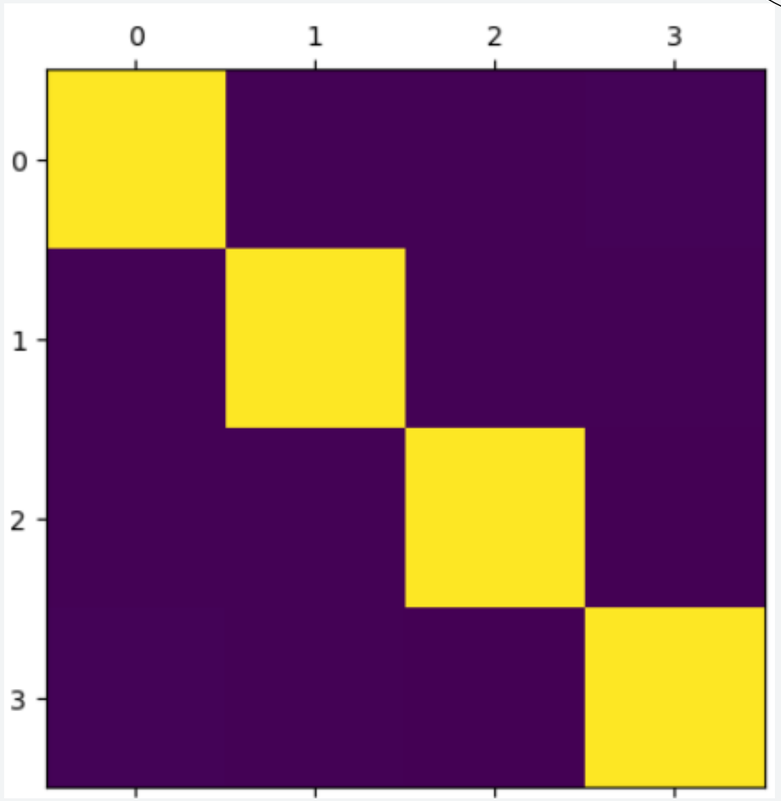
The statistical summary of numerical features in the dataset

```
data.describe().T
```

	count	mean	std	min	25%	50%	75%	max
BillNo	388023.0	560610.618886	13127.766961	536365.0	549225.00	561888.00	572131.00	581587.00
Quantity	388023.0	12.840837	182.596557	1.0	2.00	5.00	12.00	80995.00
Price	388023.0	3.077634	21.984452	0.0	1.25	1.93	3.75	8142.75
CustomerID	388023.0	15316.931710	1721.846964	12346.0	13950.00	15265.00	16837.00	18287.00

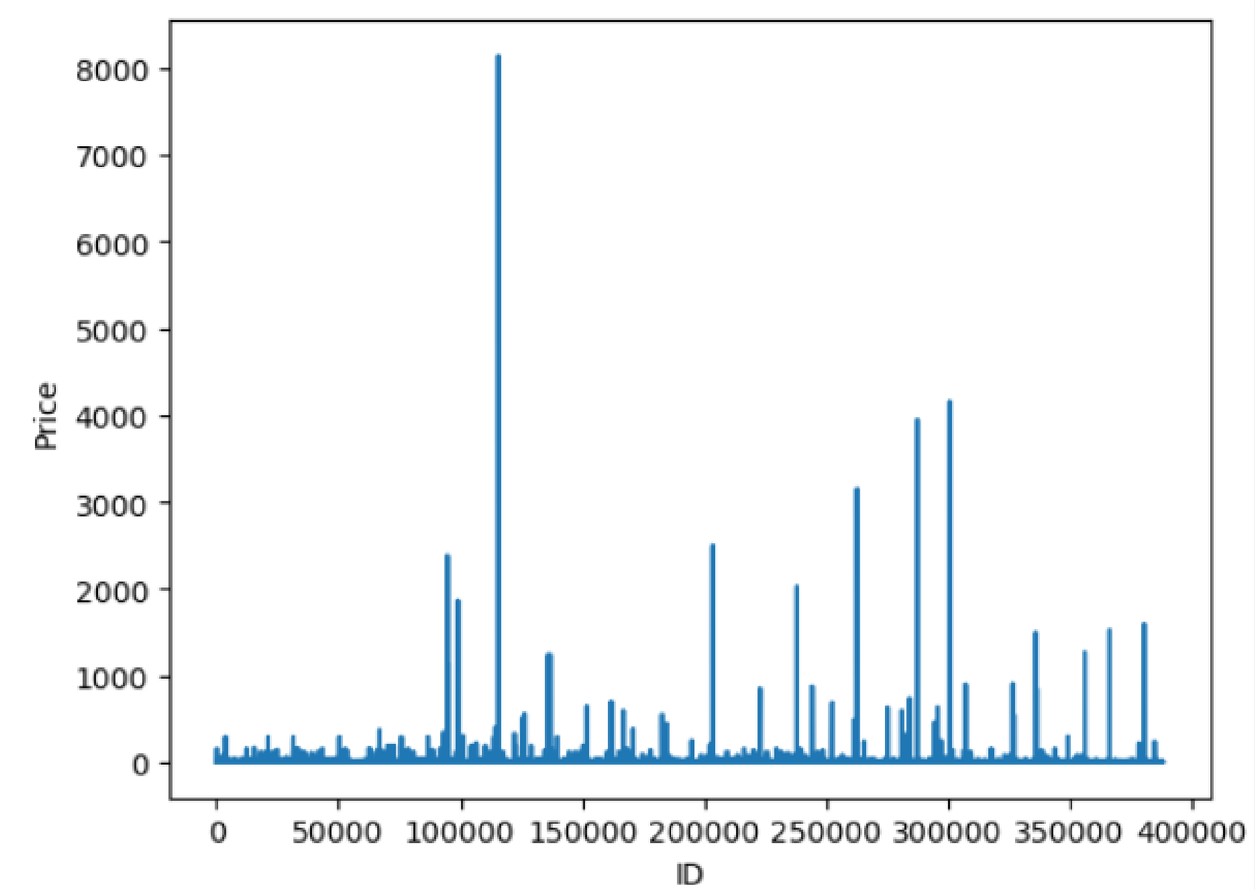
VISUALIZATION

	BillNo	Quantity	Price	CustomerID
BillNo	1.000000	-0.002486	-0.004803	-0.000335
Quantity	-0.002486	1.000000	-0.004368	-0.005976
Price	-0.004803	-0.004368	1.000000	-0.010244
CustomerID	-0.000335	-0.005976	-0.010244	1.000000



As the value approaches –1, there is a negative correlation between two variables (one variable decreases as the other increases). Approaching 1 indicates a positive correlation (both variables increase in the same direction). Close to 0 implies weak or no correlation between the two variables.

VISUALIZATION



Visualizing the data in the 'Price' column

CATEGORICAL VARIABLES

```
data["Itemname"].value_counts()
```

```
WHITE HANGING HEART T-LIGHT HOLDER    1976
REGENCY CAKESTAND 3 TIER              1643
JUMBO BAG RED RETROSPOT               1591
ASSORTED COLOUR BIRD ORNAMENT         1391
PARTY BUNTING                         1369
                                      ...
OCEAN STRIPE HAMMOCK                     1
PAINTED HEART WREATH WITH BELL           1
WEEKEND BAG VINTAGE ROSE PAISLEY         1
WRAP  PINK FLOCK                         1
PAPER CRAFT , LITTLE BIRDIE              1
Name: Itemname, Length: 3846, dtype: int64
```

```
data["Itemname"].value_counts(normalize=True)
```

```
WHITE HANGING HEART T-LIGHT HOLDER    0.005092
REGENCY CAKESTAND 3 TIER              0.004234
JUMBO BAG RED RETROSPOT               0.004100
ASSORTED COLOUR BIRD ORNAMENT         0.003585
PARTY BUNTING                         0.003528
                                         ...
OCEAN STRIPE HAMMOCK                  0.000003
PAINTED HEART WREATH WITH BELL        0.000003
WEEKEND BAG VINTAGE ROSE PAISLEY      0.000003
WRAP  PINK FLOCK                      0.000003
PAPER CRAFT , LITTLE BIRDIE           0.000003
```

Examining the values of the 'Itemname' column and exploring its normalized version

CATEGORICAL VARIABLES

EXAMINATION OF ITEMS

	ItemName	Count
0	WHITE HANGING HEART T-LIGHT HOLDER	1976
1	REGENCY CAKESTAND 3 TIER	1643
2	JUMBO BAG RED RETROSPOT	1591
3	ASSORTED COLOUR BIRD ORNAMENT	1391
4	PARTY BUNTING	1369
...	...	...
3841	OCEAN STRIPE HAMMOCK	1
3842	PAINTED HEART WREATH WITH BELL	1
3843	WEEKEND BAG VINTAGE ROSE PAISLEY	1
3844	WRAP PINK FLOCK	1

```
dataItems["percantage"]=dataItems["Count"]/dataItems["Count"].sum()
dataItems
```

	ItemName	Count	percantage
0	WHITE HANGING HEART T-LIGHT HOLDER	1976	0.005092
1	REGENCY CAKESTAND 3 TIER	1643	0.004234
2	JUMBO BAG RED RETROSPOT	1591	0.004100
3	ASSORTED COLOUR BIRD ORNAMENT	1391	0.003585
4	PARTY BUNTING	1369	0.003528
...	...	...	...
3841	OCEAN STRIPE HAMMOCK	1	0.000003
3842	PAINTED HEART WREATH WITH BELL	1	0.000003
3843	WEEKEND BAG VINTAGE ROSE PAISLEY	1	0.000003
3844	WRAP PINK FLOCK	1	0.000003

FEATURE ENGINEERING

	BillNo	ItemName	Quantity	Date	Price	CustomerID	Country	Year
0	536365	WHITE HANGING HEART T-LIGHT HOLDER	6.0	2010-12-01 08:26:00	2.55	17850.0	United Kingdom	2010
1	536365	WHITE METAL LANTERN	6.0	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	2010
2	536365	CREAM CUPID HEARTS COAT HANGER	8.0	2010-12-01 08:26:00	2.75	17850.0	United Kingdom	2010
3	536365	KNITTED UNION FLAG HOT WATER BOTTLE	6.0	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	2010
4	536365	RED WOOLLY HOTTIE WHITE HEART.	6.0	2010-12-01 08:26:00	3.39	17850.0	United Kingdom	2010

Extracting data from the 'Date' column to create a 'Year' column

DATA CLEANING

	BillNo	ItemName	Quantity	Year
0	536365	WHITE HANGING HEART T-LIGHT HOLDER	6.0	2010
1	536365	WHITE METAL LANTERN	6.0	2010
2	536365	CREAM CUPID HEARTS COAT HANGER	8.0	2010
3	536365	KNITTED UNION FLAG HOT WATER BOTTLE	6.0	2010
4	536365	RED WOOLLY HOTTIE WHITE HEART.	6.0	2010
5	536365	SET 7 BABUSHKA NESTING BOXES	2.0	2010
6	536365	GLASS STAR FROSTED T-LIGHT HOLDER	6.0	2010
7	536366	HAND WARMER UNION JACK	6.0	2010
8	536366	HAND WARMER RED POLKA DOT	6.0	2010

New dataframe after removing unnecessary columns

TRANSACTION MATRIX

ItemName	10 COLOUR SPACEBOY PEN	12 COLOURED PARTY BALLOONS	12 DAISY PEGS IN WOOD BOX	12 EGG HOUSE PAINTED WOOD	12 HANGING EGGS HAND PAINTED	12 IVORY ROSE PEG PLACE SETTINGS	12 MESSAGE CARDS WITH ENVELOPES	12 PENCIL SMALL TUBE WOODLAND	12 PENCILS SMALL TUBE RED RETROSPOT	12 PENCILS SMALL TUBE SKULL	...	ZINC STAR T-LIGHT HOLDER	ZINC SWEETHEART SOAP DISH
BillNo													
536365	0.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	...	0.0	0.0
536366	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
536367	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
536368	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
536369	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
581583	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
581584	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
581585	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
581586	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
581587	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0

18163 rows × 3846 columns

The "Transaction" matrix is a data structure representing customer purchases. Each row corresponds to a single transaction (BillNo), and the columns form a binary matrix where each column represents a product.

ALGORITHMS

In this project, three algorithms have been employed for association analysis. These are the Apriori, FP-Growth, and Eclat algorithms.

The mlxtend library was used for Apriori and FP-Growth algorithms, and the pyECLAT library was used for Eclat.

APRIORI

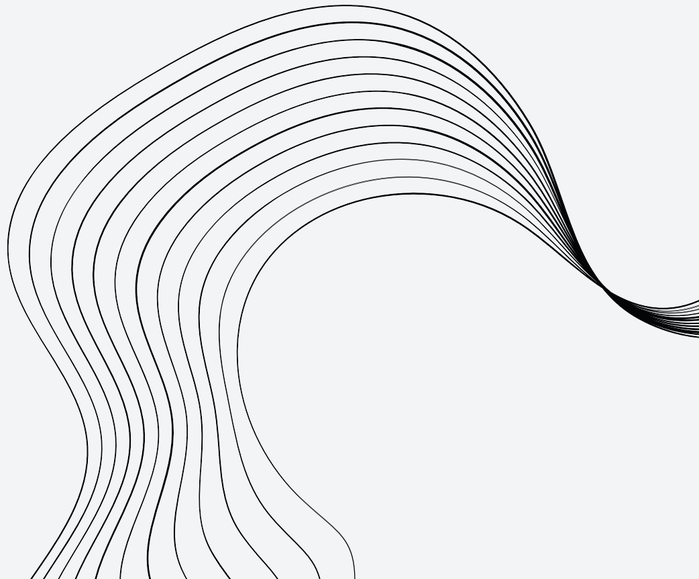
	support	itemsets
191	0.105654	(WHITE HANGING HEART T-LIGHT HOLDER)
152	0.089578	(REGENCY CAKESTAND 3 TIER)
80	0.086605	(JUMBO BAG RED RETROSPOT)
11	0.074767	(ASSORTED COLOUR BIRD ORNAMENT)
122	0.074437	(PARTY BUNTING)
...	...	...
123	0.020261	(PARTY METAL SIGN)
43	0.020206	(FELTCRAFT PRINCESS LOLA DOLL)
160	0.020206	(SET 2 TEA TOWELS I LOVE LONDON)
138	0.020206	(POTTERING IN THE SHED METAL SIGN)
193	0.020206	(WHITE WOOD GARDEN PLANT LADDER)

The Apriori function was invoked with the specified transaction table and minimum support threshold. The support values for the itemsets have been returned as a dataframe and sorted in descending order.

FP-GROWTH

	support	itemsets
0	0.105654	(WHITE HANGING HEART T-LIGHT HOLDER)
120	0.089578	(REGENCY CAKESTAND 3 TIER)
52	0.086605	(JUMBO BAG RED RETROSPOT)
2	0.074767	(ASSORTED COLOUR BIRD ORNAMENT)
164	0.074437	(PARTY BUNTING)
...	...	...
22	0.020261	(WOOD S/3 CABINET ANT WHITE FINISH)
130	0.020206	(WHITE WOOD GARDEN PLANT LADDER)
115	0.020206	(FELTCRAFT PRINCESS LOLA DOLL)
161	0.020206	(POTTERING IN THE SHED METAL SIGN)
17	0.020206	(SET 2 TEA TOWELS I LOVE LONDON)

FP-Growth is a frequent itemset mining algorithm utilized in market basket analysis and association rule discovery, distinguishing itself from Apriori by offering enhanced efficiency and requiring fewer passes through the dataset.



ECLAT

	0	1	2	3	4	5	6	7	8
0	WHITE HANGING HEART T-LIGHT HOLDER	WHITE METAL LANTERN	CREAM CUPID HEARTS COAT HANGER	KNITTED UNION FLAG HOT WATER BOTTLE	RED WOOLLY HOTTIE WHITE HEART.	SET 7 BABUSHKA NESTING BOXES	GLASS STAR FROSTED T-LIGHT HOLDER	None	None
1	HAND WARMER UNION JACK	HAND WARMER RED POLKA DOT	None	None	None	None	None	None	None
2	ASSORTED COLOUR BIRD ORNAMENT	POPPY'S PLAYHOUSE BEDROOM	POPPY'S PLAYHOUSE KITCHEN	FELTCRAFT PRINCESS CHARLOTTE DOLL	IVORY KNITTED MUG COSY	BOX OF 6 ASSORTED COLOUR TEASPOONS	BOX OF VINTAGE JIGSAW BLOCKS	BOX OF VINTAGE ALPHABET BLOCKS	HOME BUILDING BLOCK WORD
3	JAM MAKING SET WITH JARS	RED COAT RACK PARIS FASHION	YELLOW COAT RACK PARIS FASHION	BLUE COAT RACK PARIS FASHION	None	None	None	None	None
4	BATH BUILDING BLOCK	None	None	None	None	None	None	None	None

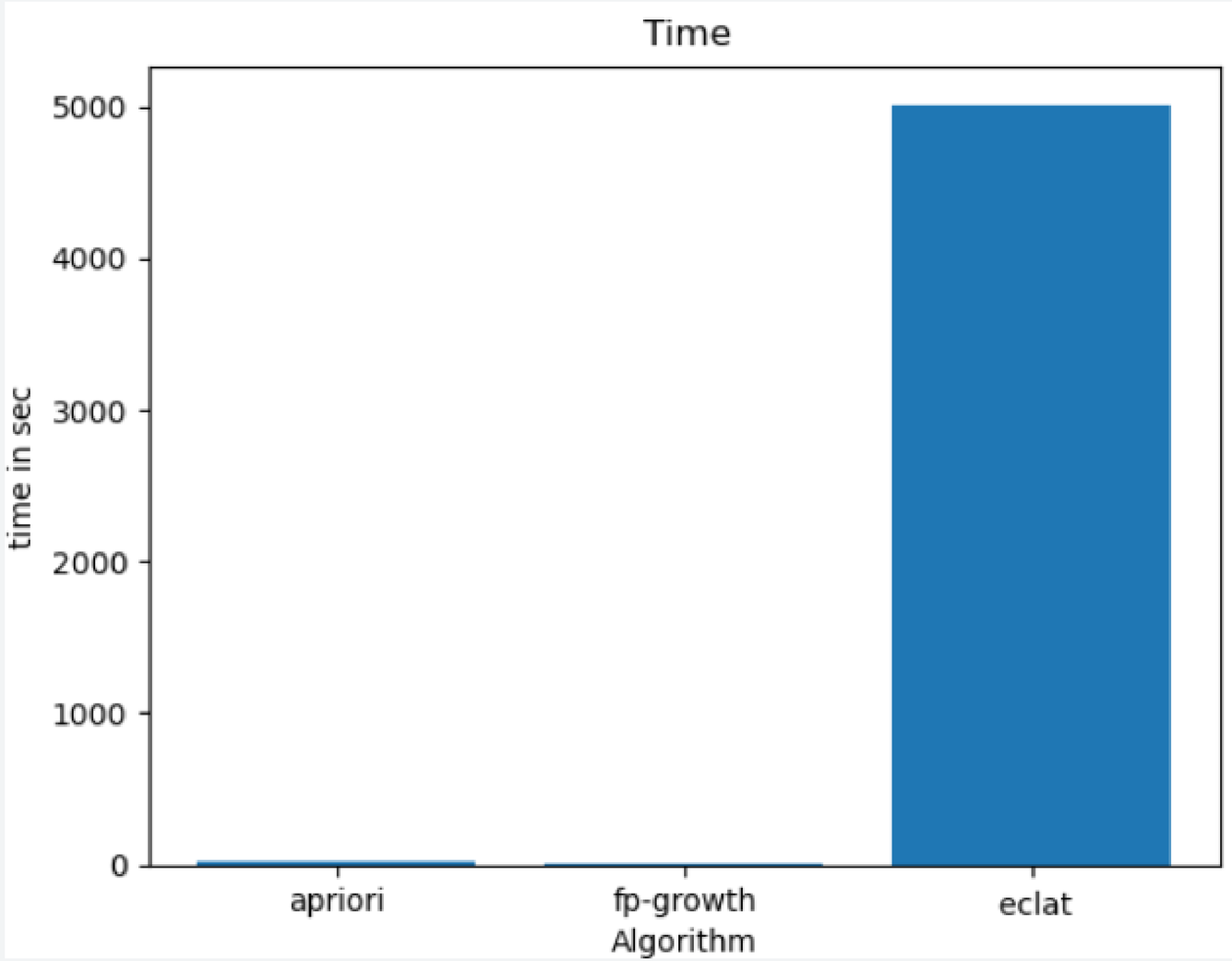
We realized that we needed to change the format of the dataset in order to use the pyECLAT library.

ECLAT

	Item	support
0	HEART OF WICKER SMALL & HEART OF WICKER LARGE	0.022023
1	PAPER CHAIN KIT VINTAGE CHRISTMAS & PAPER CHAI...	0.024445
2	JUMBO BAG RED RETROSPOT & JUMBO SHOPPER VINTAG...	0.021582
3	JUMBO BAG RED RETROSPOT & JUMBO STORAGE BAG SUKI	0.023730
4	JUMBO BAG RED RETROSPOT & JUMBO BAG PINK POLKADOT	0.029731
5	JUMBO BAG RED RETROSPOT & LUNCH BAG RED RETROSPOT	0.023069
6	JUMBO BAG RED RETROSPOT & JUMBO BAG STRAWBERRY	0.022408
7	SPACEBOY LUNCH BOX & DOLLY GIRL LUNCH BOX	0.022628
8	PARTY BUNTING & SPOTTY BUNTING	0.020922
9	LUNCH BAG APPLE DESIGN & LUNCH BAG RED RETROSPOT	0.021142
10	LUNCH BAG CARS BLUE & LUNCH BAG SPACEBOY DESIGN	0.021252
11	LUNCH BAG CARS BLUE & LUNCH BAG RED RETROSPOT	0.024721
12	LUNCH BAG CARS BLUE & LUNCH BAG SUKI DESIGN	0.021252

After creating a new dataframe, we utilized the necessary functions of the pyECLAT library to execute the ECLAT algorithm. The library returned a dataframe containing the support values of the itemsets.

TIME OF ALGORITHMS



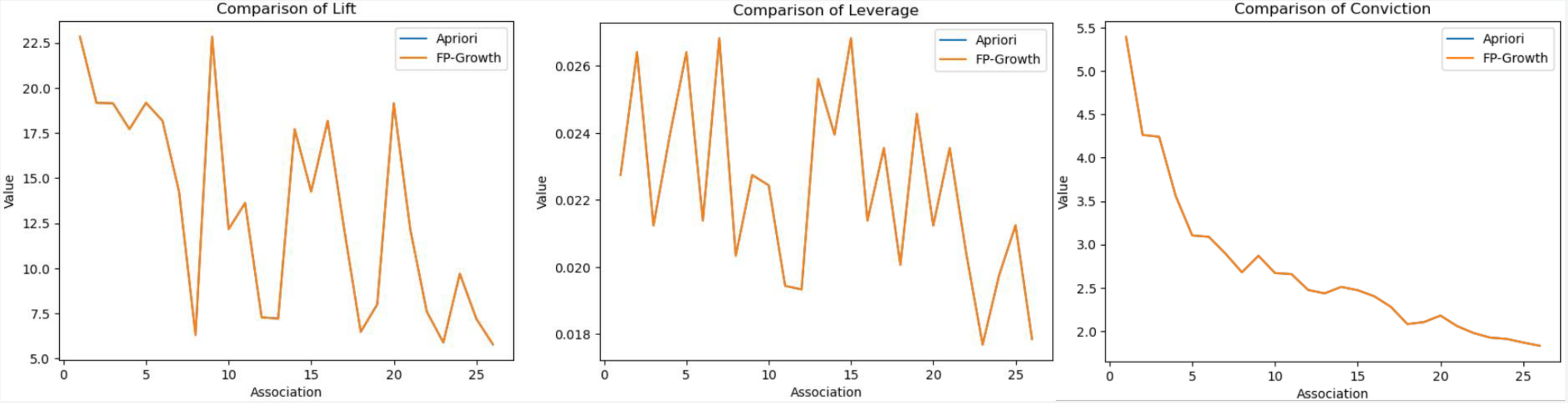
Most efficient algorithm is FP-Growth, less efficient is ECLAT algorithm.

ASSOCIATION RULES

antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	zhangs_metric
(PINK REGENCY TEACUP AND SAUCER)	(GREEN REGENCY TEACUP AND SAUCER)	0.028960	0.035952	0.023785	0.821293	22.844013	0.022743	5.394565	0.984743
(GREEN REGENCY TEACUP AND SAUCER)	(ROSES REGENCY TEACUP AND SAUCER)	0.035952	0.040412	0.027859	0.774885	19.174712	0.026406	4.262660	0.983196
(PINK REGENCY TEACUP AND SAUCER)	(ROSES REGENCY TEACUP AND SAUCER)	0.028960	0.040412	0.022408	0.773764	19.146976	0.021238	4.241541	0.976038
(GARDENERS KNEELING PAD CUP OF TEA)	(GARDENERS KNEELING PAD KEEP CALM)	0.034741	0.041238	0.025381	0.730586	17.716476	0.023949	3.558700	0.977515

This association rules function is created based on the confidence metric and with a min_threshold of 0.5.

ASSOCIATION RULES



Lift measures the impact of an association rule, conviction evaluates its ability to prevent false positives, and leverage assesses whether there is a genuine relationship between two items.

CHANGING PARAMETERS

The number of items obtained changes when the minimum support values of algorithms are altered, impacting the results of association rules function.

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	zhangs_metric
0	(REGENCY CAKESTAND 3 TIER, PINK REGENCY TEACUP...	(GREEN REGENCY TEACUP AND SAUCER)	0.013379	0.035952	0.012002	0.897119	24.953107	0.011521	9.370545	0.972942
1	(PINK REGENCY TEACUP AND SAUCER, ROSES REGENCY...	(GREEN REGENCY TEACUP AND SAUCER)	0.022408	0.035952	0.019986	0.891892	24.807707	0.019180	8.917442	0.981688
2	(GREEN REGENCY TEACUP AND SAUCER, PINK REGENCY...	(ROSES REGENCY TEACUP AND SAUCER)	0.013764	0.040412	0.012002	0.872000	21.577842	0.011446	7.496783	0.966966
3	(PINK REGENCY TEACUP AND SAUCER, REGENCY CAKES...	(GREEN REGENCY TEACUP AND SAUCER)	0.015801	0.035952	0.013764	0.871080	24.228834	0.013196	7.477884	0.974119
4	(POPPY'S PLAYHOUSE LIVINGROOM)	(POPPY'S PLAYHOUSE KITCHEN)	0.013434	0.018499	0.011452	0.852459	46.080991	0.011203	6.652395	0.991620
5	(PINK REGENCY TEACUP AND SAUCER, REGENCY CAKES...	(ROSES REGENCY TEACUP AND SAUCER)	0.015801	0.040412	0.013379	0.846690	20.951538	0.012740	6.259132	0.967560
6	(REGENCY TEA PLATE GREEN)	(REGENCY TEA PLATE ROSES)	0.013544	0.016572	0.011397	0.841463	50.775748	0.011172	6.203160	0.993765
7	(GREEN REGENCY TEACUP AND SAUCER, PINK REGENCY...	(ROSES REGENCY TEACUP AND SAUCER)	0.023785	0.040412	0.019986	0.840278	20.792868	0.019025	6.007856	0.975099
8	(SET/6 RED SPOTTY PAPER CUPS)	(SET/6 RED SPOTTY PAPER PLATES)	0.015306	0.017398	0.012608	0.823741	47.346860	0.012342	5.574762	0.994095
9	(GREEN REGENCY TEACUP AND SAUCER, REGENCY CAKE...	(ROSES REGENCY TEACUP AND SAUCER)	0.019160	0.040412	0.015746	0.821839	20.336598	0.014972	5.386076	0.969401
10	(PINK REGENCY TEACUP AND SAUCER)	(GREEN REGENCY TEACUP AND SAUCER)	0.028960	0.035952	0.023785	0.821293	22.844013	0.022743	5.394565	0.984743

antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	co
(PINK REGENCY TEACUP AND SAUCER)	(GREEN REGENCY TEACUP AND SAUCER)	0.028960	0.035952	0.023785	0.821293	22.844013	0.022743	
(GREEN REGENCY TEACUP AND SAUCER)	(ROSES REGENCY TEACUP AND SAUCER)	0.035952	0.040412	0.027859	0.774885	19.174712	0.026406	
(PINK REGENCY TEACUP AND SAUCER)	(ROSES REGENCY TEACUP AND SAUCER)	0.028960	0.040412	0.022408	0.773764	19.146976	0.021238	
(GARDENERS KNEELING PAD CUP OF TEA)	(GARDENERS KNEELING PAD KEEP CALM)	0.034741	0.041238	0.025381	0.730586	17.716476	0.023949	
(ROSES REGENCY TEACUP AND SAUCER)	(GREEN REGENCY TEACUP AND SAUCER)	0.040412	0.035952	0.027859	0.689373	19.174712	0.026406	
(DOLLY GIRL LUNCH BOX)	(SPACEBOY LUNCH BOX)	0.032869	0.037879	0.022628	0.688442	18.174674	0.021383	
(ALARM CLOCK BAKELIKE GREEN)	(ALARM CLOCK BAKELIKE RED)	0.043000	0.047074	0.028850	0.670935	14.252850	0.026826	
(RED HANGING HEART T-LIGHT HOLDER)	(WHITE HANGING HEART T-LIGHT HOLDER)	0.036283	0.105654	0.024170	0.666161	6.305096	0.020337	
(GREEN REGENCY TEACUP AND SAUCER)	(PINK REGENCY TEACUP AND SAUCER)	0.035952	0.028960	0.023785	0.661562	22.844013	0.022743	
(PAPER CHAIN KIT VINTAGE CHRISTMAS)	(PAPER CHAIN KIT 50'S CHRISTMAS)	0.037879	0.053020	0.024445	0.645349	12.171829	0.022437	

results for min_support 0.01 and 0.02

CONCLUSION

FP - Growth is the fastest algorithm
FP-growth and Apriori lead to the same average lift, support, confidence value.
The eclat algorithm is extremely slow.
Changing the parameters affects the number of items, thus altering the association rules; obtaining a sufficiently low support can lead to a more suitable table

REFERENCES

https://towardsdatascience.com/apriori-association-rule-mining-explanation-and-python-implementation-290b42afdfc6

https://www.javatpoint.com/fp-growth-algorithm-in-data-mining

https://pypi.org/project/pyECLAT/

https://www.datacamp.com/tutorial/association-rule-mining-python