{

"cells": [

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"# Market Basket Analysis 101 with Real Example"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"### Association rules, Lift, Confidence, Support"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"## Table of Content\n",

"1. Introduction\n",

"2. What is Market Basket Analysis?\n",

"3. Import Library / Data\n",

"4. Data pre-processing\n",

"5. Association Rules in Python\n",

"6. Deriving Association Rules\n",

"7. Findings / Conclusion"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"## 1.0 Introduction"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"Another exciting topic in marketing analytics is Market Basket Analysis. This is the topic of this publication. At the beginning of this post I will be introducing some key terms and metrics aimed at giving a sense of what “association” in a rule means and some ways to quantify the strength of this association. Then I will show how to generate these rules from the dataset ‘Online Retail’ using the Apriori Algorithm.\n",

"\n",

"For this post the dataset Online Retail from the statistic platform “Kaggle” was used. You can download it from my “GitHub Repository”."

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"## 2.0 What is Market Basket Analysis?"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"Market Basket Analysis is a analysis technique which identifies the strength of association between pairs of products purchased together and identify patterns of co-occurrence.\n",

"\n",

"Market Basket Analysis creates If-Then scenario rules (association rules), for example, if item A is purchased then item B is likely to be purchased. The rules are probabilistic in nature or, in other words, they are derived from the frequencies of co-occurrence in the observations. Frequency is the proportion of baskets that contain the items of interest. The rules can be used in pricing strategies, product placement, and various types of cross-selling strategies."

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"How association rules work\n",

"\n",

"Association rule mining, at a basic level, involves the use of machine learning models to analyze data for patterns, or co-occurrences, in a database. It identifies frequent if-then associations, which themselves are the association rules.\n",

"\n",

"An association rule has two parts: an antecedent (if) and a consequent (then). An antecedent is an item found within the data. A consequent is an item found in combination with the antecedent.\n",

"\n",

"Association rules are created by searching data for frequent if-then patterns and using the criteria support and confidence to identify the most important relationships. Support is an indication of how frequently the items appear in the data. Confidence indicates the number of times the if-then statements are found true. A third metric, called lift, can be used to compare confidence with expected confidence, or how many times an if-then statement is expected to be found true.\n",

"\n",

"Association rules are calculated from itemsets, which are made up of two or more items. If rules are built from analyzing all the possible itemsets, there could be so many rules that the rules hold little meaning. With that, association rules are typically created from rules well-represented in data.\n",

"\n",

"More about association rule can be found on https://michael-fuchs-python.netlify.app/2020/09/15/marketing-market-basket-analysis/"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"# 3.0 Dive into code - Import Libraries / Data"

]

},

{

"cell\_type": "code",

"execution\_count": 9,

"metadata": {},

"outputs": [],

"source": [

"import gc\n",

"import os\n",

"import time\n",

"import numpy as np\n",

"import pandas as pd\n",

"import matplotlib.pyplot as plt\n",

"import seaborn as sns\n",

"import datetime as dt\n",

"import mlxtend.preprocessing\n",

"import mlxtend.frequent\_patterns"

]

},

{

"cell\_type": "code",

"execution\_count": 14,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>InvoiceNo</th>\n",

" <th>StockCode</th>\n",

" <th>Description</th>\n",

" <th>Quantity</th>\n",

" <th>InvoiceDate</th>\n",

" <th>UnitPrice</th>\n",

" <th>CustomerID</th>\n",

" <th>Country</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>536365</td>\n",

" <td>85123A</td>\n",

" <td>WHITE HANGING HEART T-LIGHT HOLDER</td>\n",

" <td>6</td>\n",

" <td>2010-12-01 08:26:00</td>\n",

" <td>2.55</td>\n",

" <td>17850.0</td>\n",

" <td>United Kingdom</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>536365</td>\n",

" <td>71053</td>\n",

" <td>WHITE METAL LANTERN</td>\n",

" <td>6</td>\n",

" <td>2010-12-01 08:26:00</td>\n",

" <td>3.39</td>\n",

" <td>17850.0</td>\n",

" <td>United Kingdom</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>536365</td>\n",

" <td>84406B</td>\n",

" <td>CREAM CUPID HEARTS COAT HANGER</td>\n",

" <td>8</td>\n",

" <td>2010-12-01 08:26:00</td>\n",

" <td>2.75</td>\n",

" <td>17850.0</td>\n",

" <td>United Kingdom</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>536365</td>\n",

" <td>84029G</td>\n",

" <td>KNITTED UNION FLAG HOT WATER BOTTLE</td>\n",

" <td>6</td>\n",

" <td>2010-12-01 08:26:00</td>\n",

" <td>3.39</td>\n",

" <td>17850.0</td>\n",

" <td>United Kingdom</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>536365</td>\n",

" <td>84029E</td>\n",

" <td>RED WOOLLY HOTTIE WHITE HEART.</td>\n",

" <td>6</td>\n",

" <td>2010-12-01 08:26:00</td>\n",

" <td>3.39</td>\n",

" <td>17850.0</td>\n",

" <td>United Kingdom</td>\n",

" </tr>\n",

" <tr>\n",

" <th>...</th>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541904</th>\n",

" <td>581587</td>\n",

" <td>22613</td>\n",

" <td>PACK OF 20 SPACEBOY NAPKINS</td>\n",

" <td>12</td>\n",

" <td>2011-12-09 12:50:00</td>\n",

" <td>0.85</td>\n",

" <td>12680.0</td>\n",

" <td>France</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541905</th>\n",

" <td>581587</td>\n",

" <td>22899</td>\n",

" <td>CHILDREN'S APRON DOLLY GIRL</td>\n",

" <td>6</td>\n",

" <td>2011-12-09 12:50:00</td>\n",

" <td>2.10</td>\n",

" <td>12680.0</td>\n",

" <td>France</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541906</th>\n",

" <td>581587</td>\n",

" <td>23254</td>\n",

" <td>CHILDRENS CUTLERY DOLLY GIRL</td>\n",

" <td>4</td>\n",

" <td>2011-12-09 12:50:00</td>\n",

" <td>4.15</td>\n",

" <td>12680.0</td>\n",

" <td>France</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541907</th>\n",

" <td>581587</td>\n",

" <td>23255</td>\n",

" <td>CHILDRENS CUTLERY CIRCUS PARADE</td>\n",

" <td>4</td>\n",

" <td>2011-12-09 12:50:00</td>\n",

" <td>4.15</td>\n",

" <td>12680.0</td>\n",

" <td>France</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541908</th>\n",

" <td>581587</td>\n",

" <td>22138</td>\n",

" <td>BAKING SET 9 PIECE RETROSPOT</td>\n",

" <td>3</td>\n",

" <td>2011-12-09 12:50:00</td>\n",

" <td>4.95</td>\n",

" <td>12680.0</td>\n",

" <td>France</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>541909 rows × 8 columns</p>\n",

"</div>"

],

"text/plain": [

" InvoiceNo StockCode Description Quantity \\\n",

"0 536365 85123A WHITE HANGING HEART T-LIGHT HOLDER 6 \n",

"1 536365 71053 WHITE METAL LANTERN 6 \n",

"2 536365 84406B CREAM CUPID HEARTS COAT HANGER 8 \n",

"3 536365 84029G KNITTED UNION FLAG HOT WATER BOTTLE 6 \n",

"4 536365 84029E RED WOOLLY HOTTIE WHITE HEART. 6 \n",

"... ... ... ... ... \n",

"541904 581587 22613 PACK OF 20 SPACEBOY NAPKINS 12 \n",

"541905 581587 22899 CHILDREN'S APRON DOLLY GIRL 6 \n",

"541906 581587 23254 CHILDRENS CUTLERY DOLLY GIRL 4 \n",

"541907 581587 23255 CHILDRENS CUTLERY CIRCUS PARADE 4 \n",

"541908 581587 22138 BAKING SET 9 PIECE RETROSPOT 3 \n",

"\n",

" InvoiceDate UnitPrice CustomerID Country \n",

"0 2010-12-01 08:26:00 2.55 17850.0 United Kingdom \n",

"1 2010-12-01 08:26:00 3.39 17850.0 United Kingdom \n",

"2 2010-12-01 08:26:00 2.75 17850.0 United Kingdom \n",

"3 2010-12-01 08:26:00 3.39 17850.0 United Kingdom \n",

"4 2010-12-01 08:26:00 3.39 17850.0 United Kingdom \n",

"... ... ... ... ... \n",

"541904 2011-12-09 12:50:00 0.85 12680.0 France \n",

"541905 2011-12-09 12:50:00 2.10 12680.0 France \n",

"541906 2011-12-09 12:50:00 4.15 12680.0 France \n",

"541907 2011-12-09 12:50:00 4.15 12680.0 France \n",

"541908 2011-12-09 12:50:00 4.95 12680.0 France \n",

"\n",

"[541909 rows x 8 columns]"

]

},

"execution\_count": 14,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"df = pd.read\_excel(\"./data/Retail.xlsx\", sheet\_name=\"Online Retail\")\n",

"df"

]

},

{

"cell\_type": "code",

"execution\_count": 15,

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"(541909, 8)"

]

},

"execution\_count": 15,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"df.shape"

]

},

{

"cell\_type": "code",

"execution\_count": 16,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>Quantity</th>\n",

" <th>UnitPrice</th>\n",

" <th>CustomerID</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>count</th>\n",

" <td>541909.000000</td>\n",

" <td>541909.000000</td>\n",

" <td>406829.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>mean</th>\n",

" <td>9.552250</td>\n",

" <td>4.611114</td>\n",

" <td>15287.690570</td>\n",

" </tr>\n",

" <tr>\n",

" <th>std</th>\n",

" <td>218.081158</td>\n",

" <td>96.759853</td>\n",

" <td>1713.600303</td>\n",

" </tr>\n",

" <tr>\n",

" <th>min</th>\n",

" <td>-80995.000000</td>\n",

" <td>-11062.060000</td>\n",

" <td>12346.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>25%</th>\n",

" <td>1.000000</td>\n",

" <td>1.250000</td>\n",

" <td>13953.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>50%</th>\n",

" <td>3.000000</td>\n",

" <td>2.080000</td>\n",

" <td>15152.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>75%</th>\n",

" <td>10.000000</td>\n",

" <td>4.130000</td>\n",

" <td>16791.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>max</th>\n",

" <td>80995.000000</td>\n",

" <td>38970.000000</td>\n",

" <td>18287.000000</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" Quantity UnitPrice CustomerID\n",

"count 541909.000000 541909.000000 406829.000000\n",

"mean 9.552250 4.611114 15287.690570\n",

"std 218.081158 96.759853 1713.600303\n",

"min -80995.000000 -11062.060000 12346.000000\n",

"25% 1.000000 1.250000 13953.000000\n",

"50% 3.000000 2.080000 15152.000000\n",

"75% 10.000000 4.130000 16791.000000\n",

"max 80995.000000 38970.000000 18287.000000"

]

},

"execution\_count": 16,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"df.describe()"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"# 4.0 Preprocess the dataset"

]

},

{

"cell\_type": "code",

"execution\_count": 17,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>InvoiceNo</th>\n",

" <th>StockCode</th>\n",

" <th>Description</th>\n",

" <th>Quantity</th>\n",

" <th>InvoiceDate</th>\n",

" <th>UnitPrice</th>\n",

" <th>CustomerID</th>\n",

" <th>Country</th>\n",

" <th>Is\_C\_Present</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>536365</td>\n",

" <td>85123A</td>\n",

" <td>WHITE HANGING HEART T-LIGHT HOLDER</td>\n",

" <td>6</td>\n",

" <td>2010-12-01 08:26:00</td>\n",

" <td>2.55</td>\n",

" <td>17850.0</td>\n",

" <td>United Kingdom</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>536365</td>\n",

" <td>71053</td>\n",

" <td>WHITE METAL LANTERN</td>\n",

" <td>6</td>\n",

" <td>2010-12-01 08:26:00</td>\n",

" <td>3.39</td>\n",

" <td>17850.0</td>\n",

" <td>United Kingdom</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>536365</td>\n",

" <td>84406B</td>\n",

" <td>CREAM CUPID HEARTS COAT HANGER</td>\n",

" <td>8</td>\n",

" <td>2010-12-01 08:26:00</td>\n",

" <td>2.75</td>\n",

" <td>17850.0</td>\n",

" <td>United Kingdom</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>536365</td>\n",

" <td>84029G</td>\n",

" <td>KNITTED UNION FLAG HOT WATER BOTTLE</td>\n",

" <td>6</td>\n",

" <td>2010-12-01 08:26:00</td>\n",

" <td>3.39</td>\n",

" <td>17850.0</td>\n",

" <td>United Kingdom</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>536365</td>\n",

" <td>84029E</td>\n",

" <td>RED WOOLLY HOTTIE WHITE HEART.</td>\n",

" <td>6</td>\n",

" <td>2010-12-01 08:26:00</td>\n",

" <td>3.39</td>\n",

" <td>17850.0</td>\n",

" <td>United Kingdom</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>...</th>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541904</th>\n",

" <td>581587</td>\n",

" <td>22613</td>\n",

" <td>PACK OF 20 SPACEBOY NAPKINS</td>\n",

" <td>12</td>\n",

" <td>2011-12-09 12:50:00</td>\n",

" <td>0.85</td>\n",

" <td>12680.0</td>\n",

" <td>France</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541905</th>\n",

" <td>581587</td>\n",

" <td>22899</td>\n",

" <td>CHILDREN'S APRON DOLLY GIRL</td>\n",

" <td>6</td>\n",

" <td>2011-12-09 12:50:00</td>\n",

" <td>2.10</td>\n",

" <td>12680.0</td>\n",

" <td>France</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541906</th>\n",

" <td>581587</td>\n",

" <td>23254</td>\n",

" <td>CHILDRENS CUTLERY DOLLY GIRL</td>\n",

" <td>4</td>\n",

" <td>2011-12-09 12:50:00</td>\n",

" <td>4.15</td>\n",

" <td>12680.0</td>\n",

" <td>France</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541907</th>\n",

" <td>581587</td>\n",

" <td>23255</td>\n",

" <td>CHILDRENS CUTLERY CIRCUS PARADE</td>\n",

" <td>4</td>\n",

" <td>2011-12-09 12:50:00</td>\n",

" <td>4.15</td>\n",

" <td>12680.0</td>\n",

" <td>France</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541908</th>\n",

" <td>581587</td>\n",

" <td>22138</td>\n",

" <td>BAKING SET 9 PIECE RETROSPOT</td>\n",

" <td>3</td>\n",

" <td>2011-12-09 12:50:00</td>\n",

" <td>4.95</td>\n",

" <td>12680.0</td>\n",

" <td>France</td>\n",

" <td>0</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>541909 rows × 9 columns</p>\n",

"</div>"

],

"text/plain": [

" InvoiceNo StockCode Description Quantity \\\n",

"0 536365 85123A WHITE HANGING HEART T-LIGHT HOLDER 6 \n",

"1 536365 71053 WHITE METAL LANTERN 6 \n",

"2 536365 84406B CREAM CUPID HEARTS COAT HANGER 8 \n",

"3 536365 84029G KNITTED UNION FLAG HOT WATER BOTTLE 6 \n",

"4 536365 84029E RED WOOLLY HOTTIE WHITE HEART. 6 \n",

"... ... ... ... ... \n",

"541904 581587 22613 PACK OF 20 SPACEBOY NAPKINS 12 \n",

"541905 581587 22899 CHILDREN'S APRON DOLLY GIRL 6 \n",

"541906 581587 23254 CHILDRENS CUTLERY DOLLY GIRL 4 \n",

"541907 581587 23255 CHILDRENS CUTLERY CIRCUS PARADE 4 \n",

"541908 581587 22138 BAKING SET 9 PIECE RETROSPOT 3 \n",

"\n",

" InvoiceDate UnitPrice CustomerID Country \\\n",

"0 2010-12-01 08:26:00 2.55 17850.0 United Kingdom \n",

"1 2010-12-01 08:26:00 3.39 17850.0 United Kingdom \n",

"2 2010-12-01 08:26:00 2.75 17850.0 United Kingdom \n",

"3 2010-12-01 08:26:00 3.39 17850.0 United Kingdom \n",

"4 2010-12-01 08:26:00 3.39 17850.0 United Kingdom \n",

"... ... ... ... ... \n",

"541904 2011-12-09 12:50:00 0.85 12680.0 France \n",

"541905 2011-12-09 12:50:00 2.10 12680.0 France \n",

"541906 2011-12-09 12:50:00 4.15 12680.0 France \n",

"541907 2011-12-09 12:50:00 4.15 12680.0 France \n",

"541908 2011-12-09 12:50:00 4.95 12680.0 France \n",

"\n",

" Is\_C\_Present \n",

"0 0 \n",

"1 0 \n",

"2 0 \n",

"3 0 \n",

"4 0 \n",

"... ... \n",

"541904 0 \n",

"541905 0 \n",

"541906 0 \n",

"541907 0 \n",

"541908 0 \n",

"\n",

"[541909 rows x 9 columns]"

]

},

"execution\_count": 17,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"'''\n",

"Create an indicator column stipulating whether the invoice number begins with 'C'\n",

"'''\n",

"\n",

"df['Is\_C\_Present'] = (\n",

" df['InvoiceNo']\n",

" .astype(str)\n",

" .apply(lambda x: 1 if x.find('C') != -1 else 0))\n",

"\n",

"df"

]

},

{

"cell\_type": "code",

"execution\_count": 10,

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"<bound method IndexOpsMixin.value\_counts of 0 0\n",

"1 0\n",

"2 0\n",

"3 0\n",

"4 0\n",

" ..\n",

"541904 0\n",

"541905 0\n",

"541906 0\n",

"541907 0\n",

"541908 0\n",

"Name: Is\_C\_Present, Length: 541909, dtype: int64>"

]

},

"execution\_count": 10,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"df.Is\_C\_Present.value\_counts"

]

},

{

"cell\_type": "code",

"execution\_count": 18,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>InvoiceNo</th>\n",

" <th>Description</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>536365</td>\n",

" <td>WHITE HANGING HEART T-LIGHT HOLDER</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>536365</td>\n",

" <td>WHITE METAL LANTERN</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>536365</td>\n",

" <td>CREAM CUPID HEARTS COAT HANGER</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>536365</td>\n",

" <td>KNITTED UNION FLAG HOT WATER BOTTLE</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>536365</td>\n",

" <td>RED WOOLLY HOTTIE WHITE HEART.</td>\n",

" </tr>\n",

" <tr>\n",

" <th>...</th>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541904</th>\n",

" <td>581587</td>\n",

" <td>PACK OF 20 SPACEBOY NAPKINS</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541905</th>\n",

" <td>581587</td>\n",

" <td>CHILDREN'S APRON DOLLY GIRL</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541906</th>\n",

" <td>581587</td>\n",

" <td>CHILDRENS CUTLERY DOLLY GIRL</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541907</th>\n",

" <td>581587</td>\n",

" <td>CHILDRENS CUTLERY CIRCUS PARADE</td>\n",

" </tr>\n",

" <tr>\n",

" <th>541908</th>\n",

" <td>581587</td>\n",

" <td>BAKING SET 9 PIECE RETROSPOT</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>530693 rows × 2 columns</p>\n",

"</div>"

],

"text/plain": [

" InvoiceNo Description\n",

"0 536365 WHITE HANGING HEART T-LIGHT HOLDER\n",

"1 536365 WHITE METAL LANTERN\n",

"2 536365 CREAM CUPID HEARTS COAT HANGER\n",

"3 536365 KNITTED UNION FLAG HOT WATER BOTTLE\n",

"4 536365 RED WOOLLY HOTTIE WHITE HEART.\n",

"... ... ...\n",

"541904 581587 PACK OF 20 SPACEBOY NAPKINS\n",

"541905 581587 CHILDREN'S APRON DOLLY GIRL \n",

"541906 581587 CHILDRENS CUTLERY DOLLY GIRL \n",

"541907 581587 CHILDRENS CUTLERY CIRCUS PARADE\n",

"541908 581587 BAKING SET 9 PIECE RETROSPOT \n",

"\n",

"[530693 rows x 2 columns]"

]

},

"execution\_count": 18,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"'''\n",

"Filter out all transactions having either zero or a negative number of items.\n",

"Remove all invoice numbers starting with 'C' (using columns 'Is\_C\_Present').\n",

"Subset the dataframe down to 'InvoiceNo' and 'Descritpion'.\n",

"Drop all rows with at least one missing value.\n",

"'''\n",

"\n",

"\n",

"df\_clean = (\n",

" df\n",

" # filter out non-positive quantity values\n",

" .loc[df[\"Quantity\"] > 0]\n",

" # remove InvoiceNos starting with C\n",

" .loc[df['Is\_C\_Present'] != 1]\n",

" # column filtering\n",

" .loc[:, [\"InvoiceNo\", \"Description\"]]\n",

" # dropping all rows with at least one missing value\n",

" .dropna()\n",

")\n",

"\n",

"df\_clean"

]

},

{

"cell\_type": "code",

"execution\_count": 19,

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"[['HAND WARMER UNION JACK', 'HAND WARMER RED POLKA DOT'], ['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', 'LOVE BUILDING BLOCK WORD', 'RECIPE BOX WITH METAL HEART', 'DOORMAT NEW ENGLAND']]\n"

]

}

],

"source": [

"'''\n",

"Transform the data into a list of lists called invoice\_item\_list\n",

"\n",

"'''\n",

"\n",

"invoice\_item\_list = []\n",

"\n",

"for num in list(set(df\_clean.InvoiceNo.tolist())):\n",

" # filter data set down to one invoice number\n",

" tmp\_df = df\_clean.loc[df\_clean['InvoiceNo'] == num]\n",

" # extract item descriptions and convert to list\n",

" tmp\_items = tmp\_df.Description.tolist()\n",

" # append list invoice\_item\_list\n",

" invoice\_item\_list.append(tmp\_items)\n",

"\n",

"print(invoice\_item\_list[1:3])"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"To be able to run any models the data, currently in the list of lists form, needs to be encoded and recast as a dataframe. \n",

"\n",

"Outputted from the encoder is a multidimensional array, where each row is the length of the total number of unique items in the transaction dataset and the elements are Boolean variables, indicating whether that particular item is linked to the invoice number that row presents. \n",

"\n",

"With the data encoded, we can recast it as a dataframe where the rows are the invoice numbers and the columns are the unique items in the transaction dataset."

]

},

{

"cell\_type": "code",

"execution\_count": 20,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>4 PURPLE FLOCK DINNER CANDLES</th>\n",

" <th>50'S CHRISTMAS GIFT BAG LARGE</th>\n",

" <th>DOLLY GIRL BEAKER</th>\n",

" <th>I LOVE LONDON MINI BACKPACK</th>\n",

" <th>I LOVE LONDON MINI RUCKSACK</th>\n",

" <th>NINE DRAWER OFFICE TIDY</th>\n",

" <th>OVAL WALL MIRROR DIAMANTE</th>\n",

" <th>RED SPOT GIFT BAG LARGE</th>\n",

" <th>SET 2 TEA TOWELS I LOVE LONDON</th>\n",

" <th>SPACEBOY BABY GIFT SET</th>\n",

" <th>...</th>\n",

" <th>returned</th>\n",

" <th>taig adjust</th>\n",

" <th>test</th>\n",

" <th>to push order througha s stock was</th>\n",

" <th>website fixed</th>\n",

" <th>wrongly coded 20713</th>\n",

" <th>wrongly coded 23343</th>\n",

" <th>wrongly marked</th>\n",

" <th>wrongly marked 23343</th>\n",

" <th>wrongly sold (22719) barcode</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>...</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>...</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>...</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>...</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>...</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" </tr>\n",

" <tr>\n",

" <th>...</th>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>20131</th>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>...</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" </tr>\n",

" <tr>\n",

" <th>20132</th>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>...</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" </tr>\n",

" <tr>\n",

" <th>20133</th>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>...</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" </tr>\n",

" <tr>\n",

" <th>20134</th>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>...</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" </tr>\n",

" <tr>\n",

" <th>20135</th>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>...</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" <td>False</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>20136 rows × 4077 columns</p>\n",

"</div>"

],

"text/plain": [

" 4 PURPLE FLOCK DINNER CANDLES 50'S CHRISTMAS GIFT BAG LARGE \\\n",

"0 False False \n",

"1 False False \n",

"2 False False \n",

"3 False False \n",

"4 False False \n",

"... ... ... \n",

"20131 False False \n",

"20132 False False \n",

"20133 False False \n",

"20134 False False \n",

"20135 False False \n",

"\n",

" DOLLY GIRL BEAKER I LOVE LONDON MINI BACKPACK \\\n",

"0 False False \n",

"1 False False \n",

"2 False False \n",

"3 False False \n",

"4 False False \n",

"... ... ... \n",

"20131 False False \n",

"20132 False False \n",

"20133 False False \n",

"20134 False False \n",

"20135 False False \n",

"\n",

" I LOVE LONDON MINI RUCKSACK NINE DRAWER OFFICE TIDY \\\n",

"0 False False \n",

"1 False False \n",

"2 False False \n",

"3 False False \n",

"4 False False \n",

"... ... ... \n",

"20131 False False \n",

"20132 False False \n",

"20133 False False \n",

"20134 False False \n",

"20135 False False \n",

"\n",

" OVAL WALL MIRROR DIAMANTE RED SPOT GIFT BAG LARGE \\\n",

"0 False False \n",

"1 False False \n",

"2 False False \n",

"3 False False \n",

"4 False False \n",

"... ... ... \n",

"20131 False False \n",

"20132 False False \n",

"20133 False False \n",

"20134 False False \n",

"20135 False False \n",

"\n",

" SET 2 TEA TOWELS I LOVE LONDON SPACEBOY BABY GIFT SET ... \\\n",

"0 False False ... \n",

"1 False False ... \n",

"2 False False ... \n",

"3 False False ... \n",

"4 False False ... \n",

"... ... ... ... \n",

"20131 False False ... \n",

"20132 False False ... \n",

"20133 False False ... \n",

"20134 False False ... \n",

"20135 False False ... \n",

"\n",

" returned taig adjust test to push order througha s stock was \\\n",

"0 False False False False \n",

"1 False False False False \n",

"2 False False False False \n",

"3 False False False False \n",

"4 False False False False \n",

"... ... ... ... ... \n",

"20131 False False False False \n",

"20132 False False False False \n",

"20133 False False False False \n",

"20134 False False False False \n",

"20135 False False False False \n",

"\n",

" website fixed wrongly coded 20713 wrongly coded 23343 \\\n",

"0 False False False \n",

"1 False False False \n",

"2 False False False \n",

"3 False False False \n",

"4 False False False \n",

"... ... ... ... \n",

"20131 False False False \n",

"20132 False False False \n",

"20133 False False False \n",

"20134 False False False \n",

"20135 False False False \n",

"\n",

" wrongly marked wrongly marked 23343 wrongly sold (22719) barcode \n",

"0 False False False \n",

"1 False False False \n",

"2 False False False \n",

"3 False False False \n",

"4 False False False \n",

"... ... ... ... \n",

"20131 False False False \n",

"20132 False False False \n",

"20133 False False False \n",

"20134 False False False \n",

"20135 False False False \n",

"\n",

"[20136 rows x 4077 columns]"

]

},

"execution\_count": 20,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"# Initialize and fit the transaction encoder\n",

"online\_encoder = mlxtend.preprocessing.TransactionEncoder()\n",

"online\_encoder\_array = online\_encoder.fit\_transform(invoice\_item\_list)\n",

"\n",

"# Recast the encoded array as a dataframe\n",

"online\_encoder\_df = pd.DataFrame(online\_encoder\_array, columns=online\_encoder.columns\_)\n",

"\n",

"# Print the results\n",

"online\_encoder\_df"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"# 5.0 Association Rules"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"The Apriori algorithm is one of the most common techniques in Market Basket Analysis to find the association between events\n",

"\n",

"It is used to analyze the frequent itemsets in a transactional database, which then is used to generate association rules between the products."

]

},

{

"cell\_type": "code",

"execution\_count": 21,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>support</th>\n",

" <th>itemsets</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>0.013359</td>\n",

" <td>(8)</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>0.015793</td>\n",

" <td>(14)</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>0.012465</td>\n",

" <td>(20)</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>0.017630</td>\n",

" <td>(21)</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>0.017978</td>\n",

" <td>(22)</td>\n",

" </tr>\n",

" <tr>\n",

" <th>...</th>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1849</th>\n",

" <td>0.011025</td>\n",

" <td>(1840, 1827, 1837, 1838)</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1850</th>\n",

" <td>0.011174</td>\n",

" <td>(2010, 2015, 2014, 2007)</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1851</th>\n",

" <td>0.010280</td>\n",

" <td>(2010, 2015, 2020, 2007)</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1852</th>\n",

" <td>0.010181</td>\n",

" <td>(2015, 2020, 2014, 2007)</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1853</th>\n",

" <td>0.010131</td>\n",

" <td>(3937, 706, 707, 2823, 3509)</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>1854 rows × 2 columns</p>\n",

"</div>"

],

"text/plain": [

" support itemsets\n",

"0 0.013359 (8)\n",

"1 0.015793 (14)\n",

"2 0.012465 (20)\n",

"3 0.017630 (21)\n",

"4 0.017978 (22)\n",

"... ... ...\n",

"1849 0.011025 (1840, 1827, 1837, 1838)\n",

"1850 0.011174 (2010, 2015, 2014, 2007)\n",

"1851 0.010280 (2010, 2015, 2020, 2007)\n",

"1852 0.010181 (2015, 2020, 2014, 2007)\n",

"1853 0.010131 (3937, 706, 707, 2823, 3509)\n",

"\n",

"[1854 rows x 2 columns]"

]

},

"execution\_count": 21,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"'''\n",

"Run the Apriori Algorithm with min\_support = 0.01 (by default 0.5)\n",

"'''\n",

"\n",

"apriori\_model = mlxtend.frequent\_patterns.apriori(online\_encoder\_df, min\_support=0.01)\n",

"apriori\_model"

]

},

{

"cell\_type": "code",

"execution\_count": 22,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>support</th>\n",

" <th>itemsets</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>0.013359</td>\n",

" <td>( SET 2 TEA TOWELS I LOVE LONDON )</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>0.015793</td>\n",

" <td>(10 COLOUR SPACEBOY PEN)</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>0.012465</td>\n",

" <td>(12 MESSAGE CARDS WITH ENVELOPES)</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>0.017630</td>\n",

" <td>(12 PENCIL SMALL TUBE WOODLAND)</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>0.017978</td>\n",

" <td>(12 PENCILS SMALL TUBE RED RETROSPOT)</td>\n",

" </tr>\n",

" <tr>\n",

" <th>...</th>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1849</th>\n",

" <td>0.011025</td>\n",

" <td>(JUMBO BAG RED RETROSPOT, JUMBO SHOPPER VINTAG...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1850</th>\n",

" <td>0.011174</td>\n",

" <td>(LUNCH BAG RED RETROSPOT, LUNCH BAG BLACK SKU...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1851</th>\n",

" <td>0.010280</td>\n",

" <td>(LUNCH BAG RED RETROSPOT, LUNCH BAG BLACK SKU...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1852</th>\n",

" <td>0.010181</td>\n",

" <td>(LUNCH BAG RED RETROSPOT, LUNCH BAG PINK POLKA...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1853</th>\n",

" <td>0.010131</td>\n",

" <td>(CHARLOTTE BAG PINK POLKADOT, STRAWBERRY CHARL...</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>1854 rows × 2 columns</p>\n",

"</div>"

],

"text/plain": [

" support itemsets\n",

"0 0.013359 ( SET 2 TEA TOWELS I LOVE LONDON )\n",

"1 0.015793 (10 COLOUR SPACEBOY PEN)\n",

"2 0.012465 (12 MESSAGE CARDS WITH ENVELOPES)\n",

"3 0.017630 (12 PENCIL SMALL TUBE WOODLAND)\n",

"4 0.017978 (12 PENCILS SMALL TUBE RED RETROSPOT)\n",

"... ... ...\n",

"1849 0.011025 (JUMBO BAG RED RETROSPOT, JUMBO SHOPPER VINTAG...\n",

"1850 0.011174 (LUNCH BAG RED RETROSPOT, LUNCH BAG BLACK SKU...\n",

"1851 0.010280 (LUNCH BAG RED RETROSPOT, LUNCH BAG BLACK SKU...\n",

"1852 0.010181 (LUNCH BAG RED RETROSPOT, LUNCH BAG PINK POLKA...\n",

"1853 0.010131 (CHARLOTTE BAG PINK POLKADOT, STRAWBERRY CHARL...\n",

"\n",

"[1854 rows x 2 columns]"

]

},

"execution\_count": 22,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"'''\n",

"Run the same model again, but this time with use\_colnames=True.\n",

"This will replace the numerical designations with the actual item names.\n",

"'''\n",

"\n",

"apriori\_model\_colnames = mlxtend.frequent\_patterns.apriori(\n",

" online\_encoder\_df, \n",

" min\_support=0.01,\n",

" use\_colnames=True\n",

")\n",

"\n",

"apriori\_model\_colnames"

]

},

{

"cell\_type": "code",

"execution\_count": 23,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>support</th>\n",

" <th>itemsets</th>\n",

" <th>length</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>0.013359</td>\n",

" <td>( SET 2 TEA TOWELS I LOVE LONDON )</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>0.015793</td>\n",

" <td>(10 COLOUR SPACEBOY PEN)</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>0.012465</td>\n",

" <td>(12 MESSAGE CARDS WITH ENVELOPES)</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>0.017630</td>\n",

" <td>(12 PENCIL SMALL TUBE WOODLAND)</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>0.017978</td>\n",

" <td>(12 PENCILS SMALL TUBE RED RETROSPOT)</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>...</th>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1849</th>\n",

" <td>0.011025</td>\n",

" <td>(JUMBO BAG RED RETROSPOT, JUMBO SHOPPER VINTAG...</td>\n",

" <td>4</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1850</th>\n",

" <td>0.011174</td>\n",

" <td>(LUNCH BAG RED RETROSPOT, LUNCH BAG BLACK SKU...</td>\n",

" <td>4</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1851</th>\n",

" <td>0.010280</td>\n",

" <td>(LUNCH BAG RED RETROSPOT, LUNCH BAG BLACK SKU...</td>\n",

" <td>4</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1852</th>\n",

" <td>0.010181</td>\n",

" <td>(LUNCH BAG RED RETROSPOT, LUNCH BAG PINK POLKA...</td>\n",

" <td>4</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1853</th>\n",

" <td>0.010131</td>\n",

" <td>(CHARLOTTE BAG PINK POLKADOT, STRAWBERRY CHARL...</td>\n",

" <td>5</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>1854 rows × 3 columns</p>\n",

"</div>"

],

"text/plain": [

" support itemsets length\n",

"0 0.013359 ( SET 2 TEA TOWELS I LOVE LONDON ) 1\n",

"1 0.015793 (10 COLOUR SPACEBOY PEN) 1\n",

"2 0.012465 (12 MESSAGE CARDS WITH ENVELOPES) 1\n",

"3 0.017630 (12 PENCIL SMALL TUBE WOODLAND) 1\n",

"4 0.017978 (12 PENCILS SMALL TUBE RED RETROSPOT) 1\n",

"... ... ... ...\n",

"1849 0.011025 (JUMBO BAG RED RETROSPOT, JUMBO SHOPPER VINTAG... 4\n",

"1850 0.011174 (LUNCH BAG RED RETROSPOT, LUNCH BAG BLACK SKU... 4\n",

"1851 0.010280 (LUNCH BAG RED RETROSPOT, LUNCH BAG BLACK SKU... 4\n",

"1852 0.010181 (LUNCH BAG RED RETROSPOT, LUNCH BAG PINK POLKA... 4\n",

"1853 0.010131 (CHARLOTTE BAG PINK POLKADOT, STRAWBERRY CHARL... 5\n",

"\n",

"[1854 rows x 3 columns]"

]

},

"execution\_count": 23,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"'''\n",

"Add an additional column to the output of apriori\_model\_colnames that contains the size of the item set.\n",

"This will help with filtering and further analysis. \n",

"'''\n",

"\n",

"apriori\_model\_colnames['length'] = (\n",

" apriori\_model\_colnames['itemsets'].apply(lambda x: len(x))\n",

")\n",

"\n",

"apriori\_model\_colnames"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"#### Examine one case"

]

},

{

"cell\_type": "code",

"execution\_count": 24,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>support</th>\n",

" <th>itemsets</th>\n",

" <th>length</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>0.01763</td>\n",

" <td>(12 PENCIL SMALL TUBE WOODLAND)</td>\n",

" <td>1</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" support itemsets length\n",

"3 0.01763 (12 PENCIL SMALL TUBE WOODLAND) 1"

]

},

"execution\_count": 24,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"apriori\_model\_colnames[\n",

" apriori\_model\_colnames['itemsets'] == frozenset(\n",

" {'12 PENCIL SMALL TUBE WOODLAND'})]"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"The output gives us the support value for ‘12 PENCIL SMALL TUBE WOODLAND’. The support value says that this specific item appears in 1,76% of the transactions."

]

},

{

"cell\_type": "code",

"execution\_count": 25,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>support</th>\n",

" <th>itemsets</th>\n",

" <th>length</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>836</th>\n",

" <td>0.020759</td>\n",

" <td>(ALARM CLOCK BAKELIKE PINK, ALARM CLOCK BAKELI...</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>887</th>\n",

" <td>0.020362</td>\n",

" <td>(CHARLOTTE BAG SUKI DESIGN, CHARLOTTE BAG PINK...</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>923</th>\n",

" <td>0.020610</td>\n",

" <td>(STRAWBERRY CHARLOTTE BAG, CHARLOTTE BAG SUKI ...</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1105</th>\n",

" <td>0.020560</td>\n",

" <td>(JUMBO BAG BAROQUE BLACK WHITE, JUMBO BAG PIN...</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1114</th>\n",

" <td>0.020908</td>\n",

" <td>(JUMBO SHOPPER VINTAGE RED PAISLEY, JUMBO BAG...</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1116</th>\n",

" <td>0.020957</td>\n",

" <td>(JUMBO BAG BAROQUE BLACK WHITE, JUMBO STORAGE...</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1129</th>\n",

" <td>0.020560</td>\n",

" <td>(JUMBO BAG RED RETROSPOT, JUMBO BAG ALPHABET)</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1137</th>\n",

" <td>0.020163</td>\n",

" <td>(JUMBO BAG APPLES, JUMBO BAG PEARS)</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1203</th>\n",

" <td>0.020709</td>\n",

" <td>(JUMBO SHOPPER VINTAGE RED PAISLEY, JUMBO BAG ...</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1218</th>\n",

" <td>0.020560</td>\n",

" <td>(JUMBO BAG RED RETROSPOT, JUMBO STORAGE BAG SK...</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1236</th>\n",

" <td>0.020610</td>\n",

" <td>(JUMBO BAG RED RETROSPOT, RECYCLING BAG RETROS...</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1328</th>\n",

" <td>0.020610</td>\n",

" <td>(LUNCH BAG APPLE DESIGN, LUNCH BAG BLACK SKULL.)</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1390</th>\n",

" <td>0.020610</td>\n",

" <td>(LUNCH BAG PINK POLKADOT, LUNCH BAG SUKI DESIGN )</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1458</th>\n",

" <td>0.020610</td>\n",

" <td>(WHITE HANGING HEART T-LIGHT HOLDER, NATURAL S...</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1581</th>\n",

" <td>0.020362</td>\n",

" <td>(SET OF 6 SPICE TINS PANTRY DESIGN, SET OF 3 C...</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1607</th>\n",

" <td>0.020163</td>\n",

" <td>(STRAWBERRY CHARLOTTE BAG, WOODLAND CHARLOTTE ...</td>\n",

" <td>2</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1615</th>\n",

" <td>0.020262</td>\n",

" <td>(WHITE HANGING HEART T-LIGHT HOLDER, WOODEN PI...</td>\n",

" <td>2</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" support itemsets length\n",

"836 0.020759 (ALARM CLOCK BAKELIKE PINK, ALARM CLOCK BAKELI... 2\n",

"887 0.020362 (CHARLOTTE BAG SUKI DESIGN, CHARLOTTE BAG PINK... 2\n",

"923 0.020610 (STRAWBERRY CHARLOTTE BAG, CHARLOTTE BAG SUKI ... 2\n",

"1105 0.020560 (JUMBO BAG BAROQUE BLACK WHITE, JUMBO BAG PIN... 2\n",

"1114 0.020908 (JUMBO SHOPPER VINTAGE RED PAISLEY, JUMBO BAG... 2\n",

"1116 0.020957 (JUMBO BAG BAROQUE BLACK WHITE, JUMBO STORAGE... 2\n",

"1129 0.020560 (JUMBO BAG RED RETROSPOT, JUMBO BAG ALPHABET) 2\n",

"1137 0.020163 (JUMBO BAG APPLES, JUMBO BAG PEARS) 2\n",

"1203 0.020709 (JUMBO SHOPPER VINTAGE RED PAISLEY, JUMBO BAG ... 2\n",

"1218 0.020560 (JUMBO BAG RED RETROSPOT, JUMBO STORAGE BAG SK... 2\n",

"1236 0.020610 (JUMBO BAG RED RETROSPOT, RECYCLING BAG RETROS... 2\n",

"1328 0.020610 (LUNCH BAG APPLE DESIGN, LUNCH BAG BLACK SKULL.) 2\n",

"1390 0.020610 (LUNCH BAG PINK POLKADOT, LUNCH BAG SUKI DESIGN ) 2\n",

"1458 0.020610 (WHITE HANGING HEART T-LIGHT HOLDER, NATURAL S... 2\n",

"1581 0.020362 (SET OF 6 SPICE TINS PANTRY DESIGN, SET OF 3 C... 2\n",

"1607 0.020163 (STRAWBERRY CHARLOTTE BAG, WOODLAND CHARLOTTE ... 2\n",

"1615 0.020262 (WHITE HANGING HEART T-LIGHT HOLDER, WOODEN PI... 2"

]

},

"execution\_count": 25,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"apriori\_model\_colnames[\n",

" (apriori\_model\_colnames['length'] == 2) & \n",

" (apriori\_model\_colnames['support'] >= 0.02) &\n",

" (apriori\_model\_colnames['support'] < 0.021)\n",

"]"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"### This dataframe contains all the item sets (pairs of items bought together) whose support value is in the range between 2% and 2.1% of transactions.\n",

"\n",

"### When you are filtering on support, it is important to specify a range instead of a sprecific value since it is quite possible to pick a value for which there are no item sets."

]

},

{

"cell\_type": "code",

"execution\_count": 26,

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"Text(0.5, 1.0, 'Support')"

]

},

"execution\_count": 26,

"metadata": {},

"output\_type": "execute\_result"

},

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 432x288 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"apriori\_model\_colnames.hist(\"support\", grid=False, bins=30)\n",

"plt.title(\"Support\")"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"# 4.0 Deriving Association Rules"

]

},

{

"cell\_type": "code",

"execution\_count": 27,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>antecedents</th>\n",

" <th>consequents</th>\n",

" <th>antecedent support</th>\n",

" <th>consequent support</th>\n",

" <th>support</th>\n",

" <th>confidence</th>\n",

" <th>lift</th>\n",

" <th>leverage</th>\n",

" <th>conviction</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>(ALARM CLOCK BAKELIKE CHOCOLATE)</td>\n",

" <td>(ALARM CLOCK BAKELIKE GREEN)</td>\n",

" <td>0.021255</td>\n",

" <td>0.048669</td>\n",

" <td>0.013756</td>\n",

" <td>0.647196</td>\n",

" <td>13.297902</td>\n",

" <td>0.012722</td>\n",

" <td>2.696488</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>(ALARM CLOCK BAKELIKE CHOCOLATE)</td>\n",

" <td>(ALARM CLOCK BAKELIKE RED )</td>\n",

" <td>0.021255</td>\n",

" <td>0.052195</td>\n",

" <td>0.014501</td>\n",

" <td>0.682243</td>\n",

" <td>13.071023</td>\n",

" <td>0.013392</td>\n",

" <td>2.982798</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>(ALARM CLOCK BAKELIKE ORANGE)</td>\n",

" <td>(ALARM CLOCK BAKELIKE GREEN)</td>\n",

" <td>0.022100</td>\n",

" <td>0.048669</td>\n",

" <td>0.013558</td>\n",

" <td>0.613483</td>\n",

" <td>12.605201</td>\n",

" <td>0.012482</td>\n",

" <td>2.461292</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>(ALARM CLOCK BAKELIKE GREEN)</td>\n",

" <td>(ALARM CLOCK BAKELIKE RED )</td>\n",

" <td>0.048669</td>\n",

" <td>0.052195</td>\n",

" <td>0.031784</td>\n",

" <td>0.653061</td>\n",

" <td>12.511932</td>\n",

" <td>0.029244</td>\n",

" <td>2.731908</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>(ALARM CLOCK BAKELIKE RED )</td>\n",

" <td>(ALARM CLOCK BAKELIKE GREEN)</td>\n",

" <td>0.052195</td>\n",

" <td>0.048669</td>\n",

" <td>0.031784</td>\n",

" <td>0.608944</td>\n",

" <td>12.511932</td>\n",

" <td>0.029244</td>\n",

" <td>2.432722</td>\n",

" </tr>\n",

" <tr>\n",

" <th>...</th>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>493</th>\n",

" <td>(CHARLOTTE BAG SUKI DESIGN, RED RETROSPOT CHAR...</td>\n",

" <td>(STRAWBERRY CHARLOTTE BAG, WOODLAND CHARLOTTE ...</td>\n",

" <td>0.016339</td>\n",

" <td>0.020163</td>\n",

" <td>0.010131</td>\n",

" <td>0.620061</td>\n",

" <td>30.752572</td>\n",

" <td>0.009802</td>\n",

" <td>2.578931</td>\n",

" </tr>\n",

" <tr>\n",

" <th>494</th>\n",

" <td>(STRAWBERRY CHARLOTTE BAG, CHARLOTTE BAG SUKI ...</td>\n",

" <td>(RED RETROSPOT CHARLOTTE BAG, CHARLOTTE BAG PI...</td>\n",

" <td>0.015048</td>\n",

" <td>0.025924</td>\n",

" <td>0.010131</td>\n",

" <td>0.673267</td>\n",

" <td>25.971094</td>\n",

" <td>0.009741</td>\n",

" <td>2.981264</td>\n",

" </tr>\n",

" <tr>\n",

" <th>495</th>\n",

" <td>(STRAWBERRY CHARLOTTE BAG, RED RETROSPOT CHARL...</td>\n",

" <td>(CHARLOTTE BAG SUKI DESIGN, CHARLOTTE BAG PINK...</td>\n",

" <td>0.016240</td>\n",

" <td>0.020362</td>\n",

" <td>0.010131</td>\n",

" <td>0.623853</td>\n",

" <td>30.638801</td>\n",

" <td>0.009800</td>\n",

" <td>2.604405</td>\n",

" </tr>\n",

" <tr>\n",

" <th>496</th>\n",

" <td>(STRAWBERRY CHARLOTTE BAG, CHARLOTTE BAG SUKI ...</td>\n",

" <td>(WOODLAND CHARLOTTE BAG, CHARLOTTE BAG PINK PO...</td>\n",

" <td>0.016587</td>\n",

" <td>0.019617</td>\n",

" <td>0.010131</td>\n",

" <td>0.610778</td>\n",

" <td>31.135784</td>\n",

" <td>0.009806</td>\n",

" <td>2.518831</td>\n",

" </tr>\n",

" <tr>\n",

" <th>497</th>\n",

" <td>(CHARLOTTE BAG SUKI DESIGN, RED RETROSPOT CHAR...</td>\n",

" <td>(STRAWBERRY CHARLOTTE BAG, CHARLOTTE BAG PINK ...</td>\n",

" <td>0.016687</td>\n",

" <td>0.018822</td>\n",

" <td>0.010131</td>\n",

" <td>0.607143</td>\n",

" <td>32.257067</td>\n",

" <td>0.009817</td>\n",

" <td>2.497544</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>498 rows × 9 columns</p>\n",

"</div>"

],

"text/plain": [

" antecedents \\\n",

"0 (ALARM CLOCK BAKELIKE CHOCOLATE) \n",

"1 (ALARM CLOCK BAKELIKE CHOCOLATE) \n",

"2 (ALARM CLOCK BAKELIKE ORANGE) \n",

"3 (ALARM CLOCK BAKELIKE GREEN) \n",

"4 (ALARM CLOCK BAKELIKE RED ) \n",

".. ... \n",

"493 (CHARLOTTE BAG SUKI DESIGN, RED RETROSPOT CHAR... \n",

"494 (STRAWBERRY CHARLOTTE BAG, CHARLOTTE BAG SUKI ... \n",

"495 (STRAWBERRY CHARLOTTE BAG, RED RETROSPOT CHARL... \n",

"496 (STRAWBERRY CHARLOTTE BAG, CHARLOTTE BAG SUKI ... \n",

"497 (CHARLOTTE BAG SUKI DESIGN, RED RETROSPOT CHAR... \n",

"\n",

" consequents antecedent support \\\n",

"0 (ALARM CLOCK BAKELIKE GREEN) 0.021255 \n",

"1 (ALARM CLOCK BAKELIKE RED ) 0.021255 \n",

"2 (ALARM CLOCK BAKELIKE GREEN) 0.022100 \n",

"3 (ALARM CLOCK BAKELIKE RED ) 0.048669 \n",

"4 (ALARM CLOCK BAKELIKE GREEN) 0.052195 \n",

".. ... ... \n",

"493 (STRAWBERRY CHARLOTTE BAG, WOODLAND CHARLOTTE ... 0.016339 \n",

"494 (RED RETROSPOT CHARLOTTE BAG, CHARLOTTE BAG PI... 0.015048 \n",

"495 (CHARLOTTE BAG SUKI DESIGN, CHARLOTTE BAG PINK... 0.016240 \n",

"496 (WOODLAND CHARLOTTE BAG, CHARLOTTE BAG PINK PO... 0.016587 \n",

"497 (STRAWBERRY CHARLOTTE BAG, CHARLOTTE BAG PINK ... 0.016687 \n",

"\n",

" consequent support support confidence lift leverage conviction \n",

"0 0.048669 0.013756 0.647196 13.297902 0.012722 2.696488 \n",

"1 0.052195 0.014501 0.682243 13.071023 0.013392 2.982798 \n",

"2 0.048669 0.013558 0.613483 12.605201 0.012482 2.461292 \n",

"3 0.052195 0.031784 0.653061 12.511932 0.029244 2.731908 \n",

"4 0.048669 0.031784 0.608944 12.511932 0.029244 2.432722 \n",

".. ... ... ... ... ... ... \n",

"493 0.020163 0.010131 0.620061 30.752572 0.009802 2.578931 \n",

"494 0.025924 0.010131 0.673267 25.971094 0.009741 2.981264 \n",

"495 0.020362 0.010131 0.623853 30.638801 0.009800 2.604405 \n",

"496 0.019617 0.010131 0.610778 31.135784 0.009806 2.518831 \n",

"497 0.018822 0.010131 0.607143 32.257067 0.009817 2.497544 \n",

"\n",

"[498 rows x 9 columns]"

]

},

"execution\_count": 27,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"'''\n",

"Generate derive association rules for the online retail dataset.\n",

"Here we use confidence as the measure of interestingness.\n",

"Set the minimum threshold to 0.6.\n",

"Return all metrics, not just support.\n",

"'''\n",

"\n",

"rules = mlxtend.frequent\_patterns.association\_rules(\n",

" apriori\_model\_colnames, \n",

" metric=\"confidence\",\n",

" min\_threshold=0.6, \n",

" support\_only=False\n",

")\n",

"\n",

"rules"

]

},

{

"cell\_type": "code",

"execution\_count": 28,

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"Number of Associations: 498\n"

]

}

],

"source": [

"print(\"Number of Associations: {}\".format(rules.shape[0]))"

]

},

{

"cell\_type": "code",

"execution\_count": 29,

"metadata": {},

"outputs": [

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 432x288 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"rules.plot.scatter(\"support\", \"confidence\", alpha=0.5, marker=\"\*\")\n",

"plt.xlabel(\"Support\")\n",

"plt.ylabel(\"Confidence\")\n",

"plt.title(\"Association Rules\")\n",

"plt.show()"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"### Initial Finding"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"There are not any association rules with both extremly high confidence and extremely high support.\n",

"\n",

"This make sense. If an item set has high support, the items are likely to appear with many other items, making the chances of high confidence very low."

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"## Conclusion"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"With this kind of analysis from the field of mareting you can now determine which products are most often bought in combination with each other. With this knowledge it is possible to arrange the products efficiently in the store. In the best case, products that are often bought together are positioned in the opposite direction in the store so that customers are forced to walk past as many other products as possible.\n",

"\n",

"Furthermore, one can now consider targeted discount campaigns. If you discount a product that is often bought in combination with others, you increase the chance of buying these products in combination, whereby a small discount is granted on only one."

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"References\n",

"https://michael-fuchs-python.netlify.app/2020/09/15/marketing-market-basket-analysis/"

]

}

],

"metadata": {

"kernelspec": {

"display\_name": "Python 3",

"language": "python",

"name": "python3"

},

"language\_info": {

"codemirror\_mode": {

"name": "ipython",

"version": 3

},

"file\_extension": ".py",

"mimetype": "text/x-python",

"name": "python",

"nbconvert\_exporter": "python",

"pygments\_lexer": "ipython3",

"version": "3.8.5"

}

},

"nbformat": 4,

"nbformat\_minor": 4

}