MARKET BASKET ANALYSIS

MARKET BASKET ANALYSIS

Teknik Data Mining

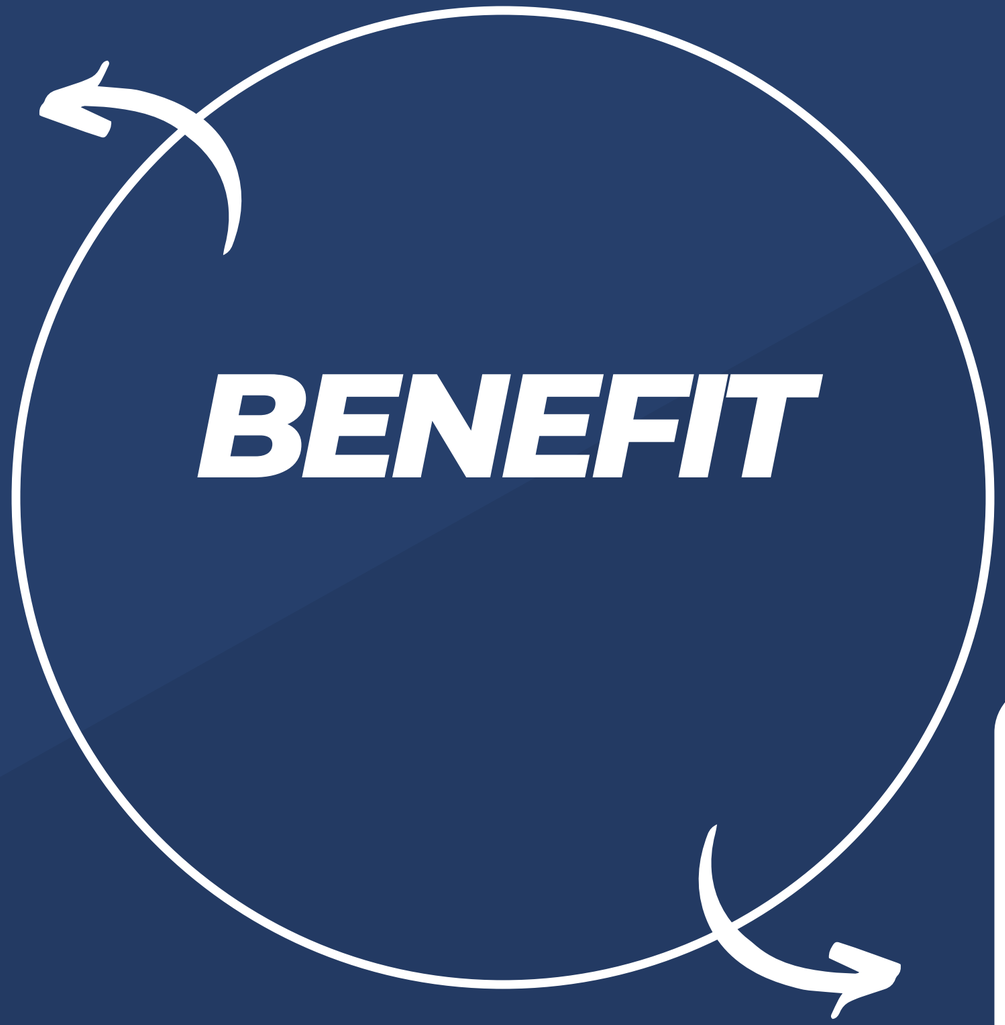
Tujuan : Menemukan asosiasi atau hubungan antara dua atau lebih produk

Konsep

Perhitungan frekuensi kemunculan produk-produk yang dibeli secara bersamaan dalam suatu transaksi atau order

Cross-selling

Strategi pemasaran yang bertujuan untuk menawarkan produk atau layanan tambahan kepada pelanggan yang sudah membeli produk utama.



Up-selling

Strategi penjualan yang mendorong pelanggan untuk membeli produk atau layanan yang lebih mahal atau lebih banyak

DATASET

Columns	Dtypes
Order_id	object
product_code	object
product_name	object
quantity	int64
order_date	object
price	float64
customer_id	float64

	order_id	product_code	product_name	quantity	order_date	price	customer_id
0	493410	TEST001	This is a test product.	5	2010-01-04 09:24:00	4.50	12346.0
1	C493411	21539	RETRO SPOTS BUTTER DISH	-1	2010-01-04 09:43:00	4.25	14590.0
2	493412	TEST001	This is a test product.	5	2010-01-04 09:53:00	4.50	12346.0
3	493413	21724	PANDA AND BUNNIES STICKER SHEET	1	2010-01-04 09:54:00	0.85	NaN
4	493413	84578	ELEPHANT TOY WITH BLUE T-SHIRT	1	2010-01-04 09:54:00	3.75	NaN
...	...	...	...	...	...	...	...
461768	539991	21618	4 WILDFLOWER BOTANICAL CANDLES	1	2010-12-23 16:49:00	1.25	NaN
461769	539991	72741	GRAND CHOCOLATECANDLE	4	2010-12-23 16:49:00	1.45	NaN
461770	539992	21470	FLOWER VINE RAFFIA FOOD COVER	1	2010-12-23 17:41:00	3.75	NaN
461771	539992	22258	FELT FARM ANIMAL RABBIT	1	2010-12-23 17:41:00	1.25	NaN
461772	539992	21155	RED RETROSPOT PEG BAG	1	2010-12-23 17:41:00	2.10	NaN

461773 rows × 7 columns

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 461773 entries, 0 to 461772
Data columns (total 7 columns):
 #   Column        Non-Null Count   Dtype
---  ------        --------------   -----
 0   order_id      461773 non-null  object
 1   product_code  461773 non-null  object
 2   product_name  459055 non-null  object
 3   quantity      461773 non-null  int64
 4   order_date    461773 non-null  object
 5   price         461773 non-null  float64
 6   customer_id   360853 non-null  float64
dtypes: float64(2), int64(1), object(4)
memory usage: 24.7+ MB
```

PREPROCESSING DATA

- Buat kolom ‘’date”

```
# Create column 'date'
df['date'] = pd.to_datetime(df['order_date']).dt.date
df['date'] = pd.to_datetime(df['date'])
```

- Check Null Values

```
# check null values
df.isnull().sum()
```

```
order_id              0
product_code          0
product_name       2718
quantity              0
order_date            0
price                 0
customer_id      100920
date                  0
dtype: int64
```

- Menghapus Null values di ‘customer_id’ dan ‘product_id’

```
# delete null values in 'customer_id' and 'product_name'
df = df[~df['customer_id'].isna()]
df = df[~df['product_name'].isna()]
```

PREPROCESSING DATA

- Ubah tipe data ‘customer_id’ menjadi string

```
# convert 'customer_id' to string
df['customer_id'] = df['customer_id'].astype(str)
```

- Membuat semua product_name berhuruf kecil

```
# lowercase 'product_name'
df['product_name'] = df['product_name'].str.lower()
```

- Menghapus semua baris dengan product_code atau product_name test

```
# delete all rows with product_code or product_name 'test'
df = df[(~df['product_code'].str.lower().str.contains('test')) |
                    (~df['product_name'].str.contains('test '))]
```

- Menghapus baris dengan status cancelled, yaitu yang order_id-nya diawali 'C'

```
# delete rows with cancelled status, namely those whose order_id starts with 'C'
df = df[df['order_id'].str[:1]!='C']
```

PREPROCESSING DATA

- Mengubah nilai quantity yang negatif menjadi positif karena nilai negatif tersebut hanya menandakan order tersebut cancelled

```
# change the negative quantity value to positive
# because the negative value only indicates that the order is cancelled
df['quantity'] = df['quantity'].abs()
```

- Menghapus baris dengan price bernilai negatif

```
# delete rows with negative price values
df = df[df['price'] > 0 ]
```

- Membuat nilai amount, yaitu perkalian antara quantity dan price

```
# create an amount value, which is the multiplication of quantity and price
df['amount'] = df['quantity'] * df['price']
```

PREPROCESSING DATA

- Mengganti product_name dari product_code yang memiliki beberapa product_name dengan salah satu product_name-nya yang paling sering muncul

```
# replace the product_name of a product_code that has multiple product_names with the one that appears most often
most_freq_product_name = df.groupby(['product_code', 'product_name'], as_index=False) \
    .agg(order_cnt=('order_id', 'nunique')) \
    .sort_values(['product_code', 'order_cnt'], ascending=[True, False])

most_freq_product_name['rank'] = most_freq_product_name.groupby('product_code')['order_cnt'] \
    .rank(method='first', ascending=False)

most_freq_product_name = most_freq_product_name[most_freq_product_name['rank'] == 1] \
    .drop(columns=['order_cnt', 'rank'])

df = df.merge(most_freq_product_name.rename(columns={'product_name': 'most_freq_product_name'}),
              how='left', on='product_code')

df['product_name'] = df['most_freq_product_name']
df = df.drop(columns='most_freq_product_name')
```

- Menghapus Outliers

```
# delete outliers
df = df[(np.abs(stats.zscore(df[['quantity','amount']]))<3).all(axis=1)]
df = df.reset_index(drop=True)
df
```

SETELAH PREPROCESSING DATA

	order_id	product_code	product_name	quantity	order_date	price	customer_id	date	amount
0	493414	21844	red retrospot mug	36	2010-01-04 10:28:00	2.55	14590.0	2010-01-04	91.80
1	493414	21533	retro spot large milk jug	12	2010-01-04 10:28:00	4.25	14590.0	2010-01-04	51.00
2	493414	37508	new england ceramic cake server	2	2010-01-04 10:28:00	2.55	14590.0	2010-01-04	5.10
3	493414	35001G	hand open shape gold	2	2010-01-04 10:28:00	4.25	14590.0	2010-01-04	8.50
4	493414	21527	red retrospot traditional teapot	12	2010-01-04 10:28:00	6.95	14590.0	2010-01-04	83.40
...	...	...	...	...	...	...	...	...	...
350087	539988	84380	set of 3 butterfly cookie cutters	1	2010-12-23 16:06:00	1.25	18116.0	2010-12-23	1.25
350088	539988	84849D	hot baths soap holder	1	2010-12-23 16:06:00	1.69	18116.0	2010-12-23	1.69
350089	539988	84849B	fairy soap soap holder	1	2010-12-23 16:06:00	1.69	18116.0	2010-12-23	1.69
350090	539988	22854	cream sweetheart egg holder	2	2010-12-23 16:06:00	4.95	18116.0	2010-12-23	9.90
350091	539988	47559B	tea time oven glove	2	2010-12-23 16:06:00	1.25	18116.0	2010-12-23	2.50

350092 rows × 9 columns

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 350092 entries, 0 to 350091
Data columns (total 9 columns):
 #   Column        Non-Null Count   Dtype
---  ------        --------------   -----
 0   order_id      350092 non-null  object
 1   product_code  350092 non-null  object
 2   product_name  350092 non-null  object
 3   quantity      350092 non-null  int64
 4   order_date    350092 non-null  object
 5   price         350092 non-null  float64
 6   customer_id   350092 non-null  object
 7   date          350092 non-null  datetime64[ns]
 8   amount        350092 non-null  float64
dtypes: datetime64[ns](1), float64(2), int64(1), object(5)
memory usage: 24.0+ MB
```

DATASET BASKET

Membuat DataFrame ‘basket’

```
basket = pd.pivot_table(df,
                        index='order_id',
                        columns='product_name',
                        values = 'product_code',
                        aggfunc='nunique',
                        fill_value=0)
```

product_name	10 colour spaceboy pen	12 ass zinc christmas decorations	12 coloured party balloons	12 daisy pegs in wood box	12 egg house painted wood	12 ivory rose peg place settings	12 message cards with envelopes	12 mini toadstool pegs	12 pencil small tube woodland	12 pencils small tube posy	...	zinc heart lattice charger large	zinc heart lattice charger small	zinc heart lattice double planter	zinc heart lattice planter bowl	: h lat t-l ho
order_id																
493414	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
493427	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
493428	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
493432	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
493433	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
539981	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
539982	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
539985	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
539987	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
539988	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	

Encode DataFrame basket dengan nilai True untuk semua nilai di atas 0 dan False untuk semua nilai 0

```
def encode(x):
    if x==0:
        return False
    if x>0:
        return True

basket_encode = basket.applymap(encode)
basket_encode
```

product_name	10 colour spaceboy pen	12 ass zinc christmas decorations	12 coloured party balloons	12 daisy pegs in wood box	12 egg house painted wood	12 ivory rose peg place settings	12 message cards with envelopes	12 mini toadstool pegs	12 pencil small tube woodland	12 pencils small tube posy	...	zinc heart lattice charger large	zinc heart lattice charger small	zinc heart lattice double planter	zinc heart lattice planter bowl	: h lat t-l ho
order_id																
493414	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
493427	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
493428	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
493432	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
493433	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
539981	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
539982	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
539985	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
539987	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
539988	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F

DATASET BASKET

Mengambil transaksi dengan banyaknya produk unik lebih dari 1 saja

```
basket_filter = basket_encode[(basket_encode>0).sum(axis=1)>1]
basket_filter
```

product_name	10 colour spaceboy pen	12 ass zinc christmas decorations	12 coloured party balloons	12 daisy pegs in wood box	12 egg house painted wood	12 ivory rose peg place settings	12 message cards with envelopes	12 mini toadstool pegs	12 pencil small tube woodland	12 pencils small tube posy	...	zinc heart lattice charger large	zinc heart lattice charger small	zinc heart lattice double planter	zinc heart lattice planter bowl	h lat t-l ho
order_id																
493414	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
493427	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
493428	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
493432	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
493433	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
539978	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
539981	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
539982	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
539985	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F
539988	False	False	False	False	False	False	False	False	False	False	...	False	False	False	False	F

APRIORI ALGORITHM

Apriori Algorithm adalah salah satu algoritma yang digunakan dalam **association rule learning** untuk menemukan hubungan atau pola tersembunyi dalam dataset yang besar. Algoritma ini banyak digunakan dalam **market basket analysis.**

Tujuan : Menemukan asosiasi antara item yang sering dibeli bersama dalam suatu transaksi.

IMPLEMENTASI APRIORI ALGORITHM

Install Libarary

```
!pip install mlxtend
```

Import Libararies

```
from mlxtend.frequent_patterns import apriori
from mlxtend.frequent_patterns import association_rules
```

IMPLEMENTASI APRIORI ALGORITHM

Buat list frequent itemset (kumpulan produk yang sering dibeli)

```
frequent_itemset = apriori(basket_filter,
                           min_support=.01,
                           use_colnames=True).sort_values('support', ascending=False).reset_index(drop=True)

frequent_itemset['product_cnt'] = frequent_itemset['itemsets'].apply(lambda x: len(x))
frequent_itemset
```

	support	itemsets	product_cnt
0	0.177953	(white hanging heart t-light holder)	1
1	0.099440	(regency cakestand 3 tier)	1
2	0.096841	(jumbo bag red retrospot)	1
3	0.079912	(pack of 72 retro spot cake cases)	1
4	0.078512	(assorted colour bird ornament)	1
...	...	...	...
1189	0.010064	(key fob , front door , key fob , shed, key f...	3
1190	0.010064	(wrap,suki and friends)	1
1191	0.010064	(retrospot padded seat cushion)	1
1192	0.010064	(pack of 6 pannetone gift boxes, pack of 6 bir...	2
1193	0.010064	(jumbo storage bag suki, lunch bag red spotty,...	3

1194 rows × 3 columns

IMPLEMENTASI APRIORI ALGORITHM

Hitung nilai support, confidence, dan lift dari setiap pasangan produk yang mungkin

```
product_association = association_rules(
                                        frequent_itemset,
                                        metric='confidence',
                                        min_threshold=0.7
                                    ).sort_values(
                                        ['support', 'confidence'],
                                        ascending=[False, False]
                                    ).reset_index(drop=True)

product_association
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	representativity	leverage	conviction	zhangs_metric	jaccard	certa
0	(red hanging heart t-light holder)	(white hanging heart t-light holder)	0.058784	0.177953	0.042455	0.722222	4.058510	1.0	0.031995	2.959371	0.800671	0.218525	0.662
1	(sweetheart ceramic trinket box)	(strawberry ceramic trinket box)	0.049254	0.074980	0.037457	0.760487	10.142533	1.0	0.033764	3.862089	0.948103	0.431644	0.741
2	(toilet metal sign)	(bathroom metal sign)	0.026993	0.040589	0.021728	0.804938	19.831353	1.0	0.020632	4.918499	0.975918	0.473837	0.796
3	(red retrospot sugar jam bowl)	(red retrospot small milk jug)	0.023660	0.037123	0.016796	0.709859	19.121592	1.0	0.015917	3.318652	0.970669	0.381818	0.698
4	(painted metal pears assorted)	(assorted colour bird ornament)	0.021927	0.078512	0.016596	0.756839	9.639738	1.0	0.014874	3.789618	0.916356	0.197933	0.736

KESIMPULAN

RED HANGING HEART T-LIGHT HOLDER

- Dibeli dalam 5,87% dari total transaksi.
- 72% pembeli juga membeli WHITE HANGING HEARTS T-LIGHT HOLDER.
- Keduanya memiliki asosiasi kuat, dengan peluang dibeli bersamaan 4x lebih besar dibanding dibeli secara terpisah.
- **Strategi**: Letakkan berdekatan atau tawarkan dalam bundling.

SWEETHEART CERAMIC TRINKET BOX

- Dibeli dalam 4,9% dari total transaksi.
- 76% pembeli juga membeli STRAWBERRY CERAMIC TRINKET BOX.
- Keduanya memiliki asosiasi kuat, dengan peluang dibeli bersamaan 10x lebih besar dibanding dibeli secara terpisah.
- **Strategi**: Letakkan berdekatan atau tawarkan dalam bundling.

KESIMPULAN

TOILET METAL SIGN

- Dibeli dalam 2,69% dari total transaksi.
- 80% pembeli juga membeli BATHROOM METAL SIGN.
- Keduanya memiliki asosiasi sangat kuat, dengan peluang dibeli bersamaan 19x lebih besar dibanding dibeli secara terpisah.
- **Strategi**: Letakkan berdekatan atau tawarkan dalam bundling.

RED RETROSPOT SUGAR JAM BOWL

- Dibeli dalam 2,36% dari total transaksi.
- 70,9% pembeli juga membeli RED RETROSPOT SMALL MILK JUG.
- Keduanya memiliki asosiasi kuat, dengan peluang dibeli bersamaan 19x lebih besar dibanding dibeli secara terpisah.
- **Strategi**: Letakkan berdekatan atau tawarkan dalam bundling.

THANK YOU

I look forward to working with you

CONTACT ME

kaylaaadra12@gmail.com

www.linkedin.com/in/kaylaalysa

www.github.com/kaylaalysa