

TUGAS KELOMPOK
PROGRAM STUDI INDEPENDEN
ORBIT FUTURE ACADEMY

Identitas Kelompok

Kelompok	: 10
Nama Anggota	: Halomoan Filipus Simarmata (Jupyter XXI) Diana Eka Riyani (Jupyter XXI) Nyayu Chika Marselina (Jupyter XXI) Sukma Imelda (Cordoba) Athiya Shinta Wulandari (Cordoba)
Coach	: Ipin Sugiyarto
Program	: Foundations of AI and Life Skills for Gen-Z
Hari, Tanggal	: Jumat, 1 April 2022

Tugas: Membuat model analisa market basket dengan dataset berbeda (dari beberapa model transaksi silahkan kalian pilih) proses sama seperti coding latihan. Buat kesimpulan barang apa yang cocok disandingkan berdasarkan kekuatan korelasi (lift). Laporrannya screenshoot coding & hasil serta penjelasan dari masing masing step.

Penyelesaian:

Groceries Market Basket Analysis

Market basket analysis adalah suatu metodologi untuk melakukan analisis buying habit konsumen dengan menemukan asosiasi antar beberapa item yang berbeda, yang diletakkan konsumen dalam shopping basket (keranjang belanja) yang dibeli pada suatu transaksi tertentu. Tujuan dari market basket analysis adalah untuk mengetahui produk-produk mana yang mungkin akan dibeli secara bersamaan.

Algoritma Apriori adalah suatu algoritma dasar yang diusulkan oleh Agrawal & Srikant pada tahun 1994 untuk penentuan frequent itemsets untuk aturan asosiasi boolean. Algoritma Apriori memberi kita sifat asosiatif dalam transaksi. Ini juga dikenal sebagai Aturan Asosiasi. Aturan asosiasi atau association rule adalah teknik untuk menemukan aturan asosiasi antara suatu kombinasi item. Terdapat 3 metrik untuk mengukur ketepatan aturan, yaitu:

a. Support

Support adalah indikasi seberapa sering kumpulan item muncul pada dataset. Berikut perhitungan nilai support:

$$supp(X \Rightarrow Y) = \frac{|X \cup Y|}{n}$$

b. Confidence

Confidence adalah suatu ukuran yang menunjukkan hubungan antar dua item secara conditional (berdasarkan suatu kondisi tertentu). Berikut perhitungan nilai confidence:

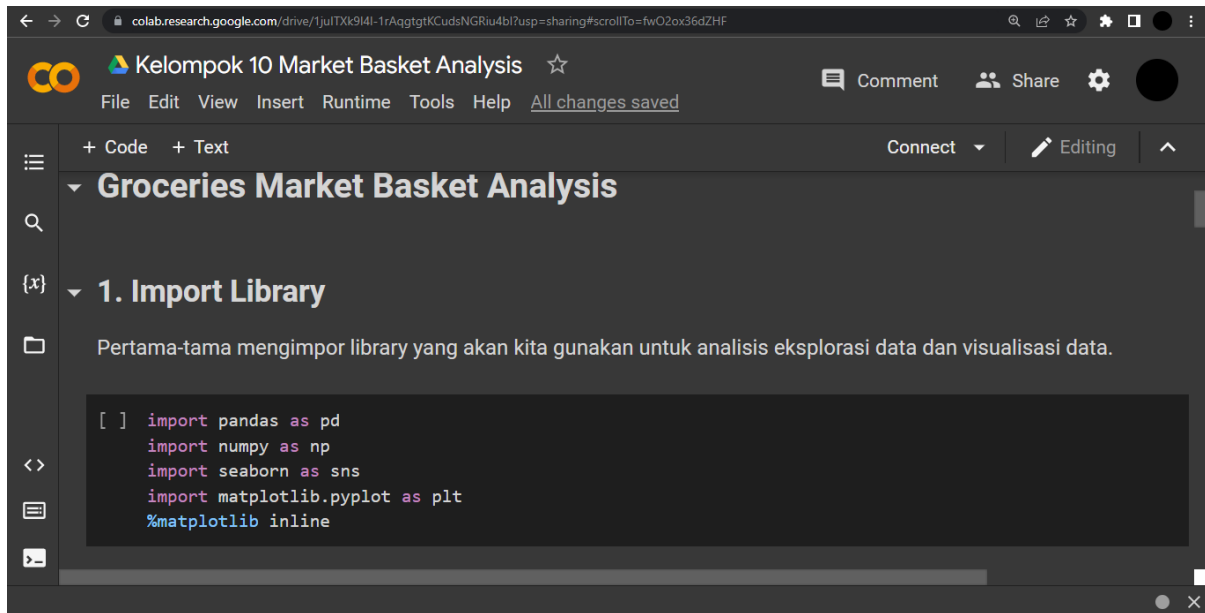
$$conf(X \Rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)}$$

c. Lift

Lift mengacu pada bagaimana peluang kedua item dibeli ketika item pertama dibeli. Berikut perhitungan nilai confidence:

$$lift(X \Rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)supp(Y)}$$

1. Import Library



Kelompok 10 Market Basket Analysis

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Groceries Market Basket Analysis

1. Import Library

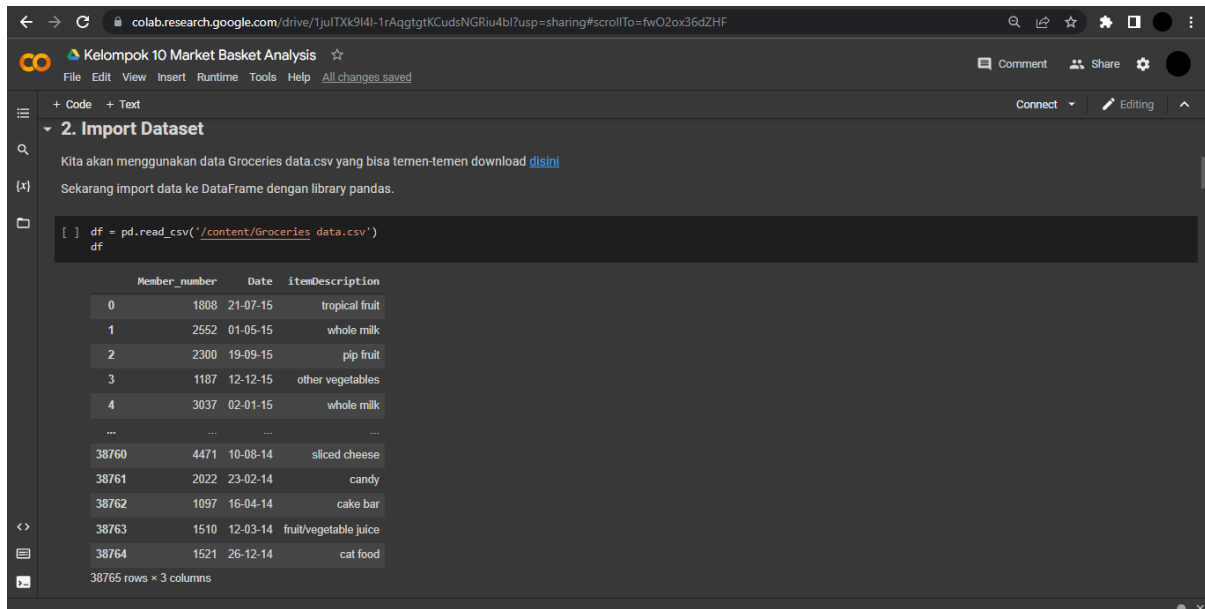
Pertama-tama mengimpor library yang akan kita gunakan untuk analisis eksplorasi data dan visualisasi data.

```
[ ] import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

2. Import Dataset

Kita akan menggunakan data Groceries data.csv yang bisa teman-teman download di <https://www.kaggle.com/datasets/rashikrahmanpritom/groceries-dataset-for-market-basket-analysismba?select=Groceries+data.csv>

Sekarang import data ke DataFrame dengan library pandas.



Kelompok 10 Market Basket Analysis

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Groceries Market Basket Analysis

2. Import Dataset

Kita akan menggunakan data Groceries data.csv yang bisa teman-teman download [disini](#)

Sekarang import data ke DataFrame dengan library pandas.

```
[ ] df = pd.read_csv('/content/Groceries data.csv')
df
```

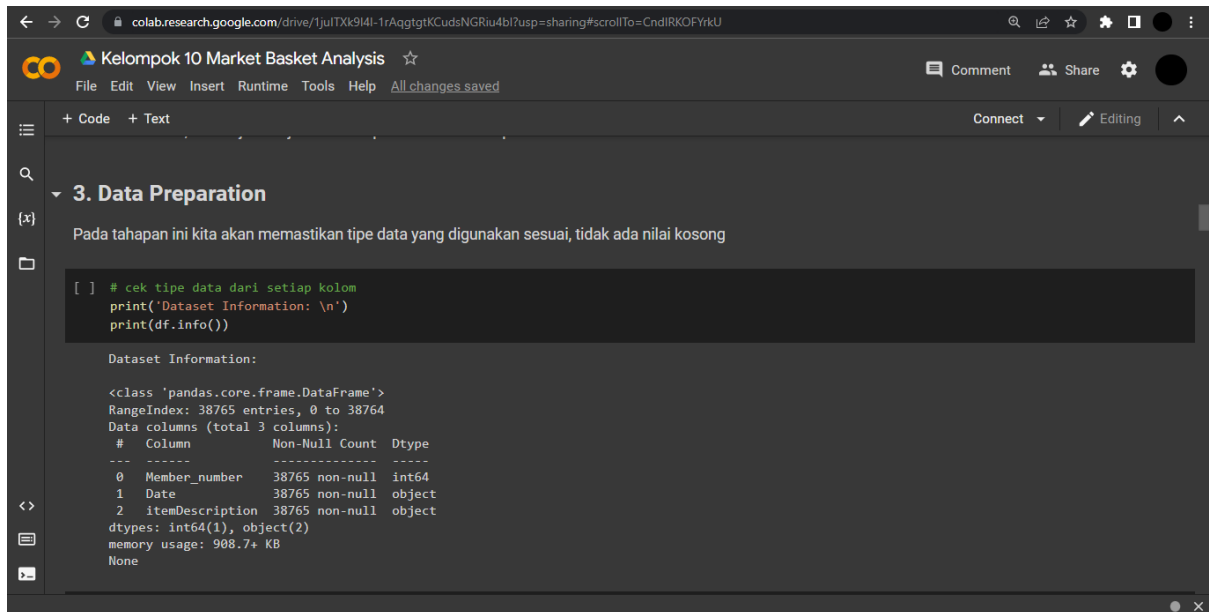
	Member_number	Date	itemDescription
0	1808	21-07-15	tropical fruit
1	2552	01-05-15	whole milk
2	2300	19-09-15	pip fruit
3	1187	12-12-15	other vegetables
4	3037	02-01-15	whole milk
...
38760	4471	10-08-14	sliced cheese
38761	2022	23-02-14	candy
38762	1097	16-04-14	cake bar
38763	1510	12-03-14	fruit/vegetable juice
38764	1521	26-12-14	cat food

38765 rows x 3 columns

Dari tabel diatas, menunjukkan jumlah data pada dataset terdapat 38765 baris dan 3 kolom.

3. Data Preparation

Pada tahapan ini kita akan memastikan tipe data yang digunakan sesuai, tidak ada nilai kosong.



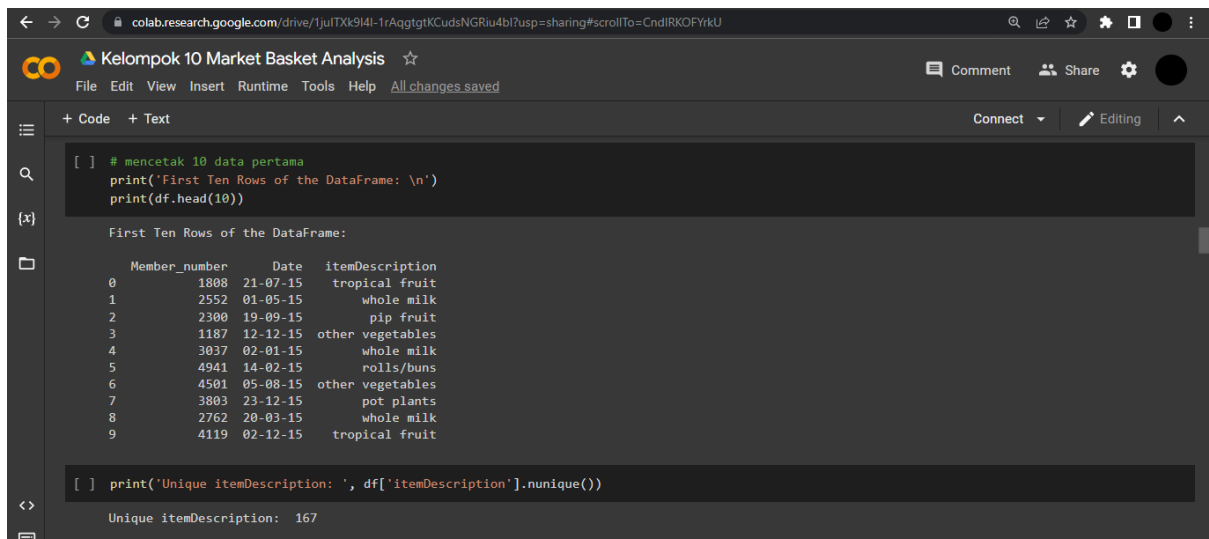
The screenshot shows a Google Colab notebook titled "Kelompok 10 Market Basket Analysis". The code cell contains the following Python code:

```
[ ] # cek tipe data dari setiap kolom
print('Dataset Information: \n')
print(df.info())
```

The output of the code is displayed below the code cell:

```
Dataset Information:

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 38765 entries, 0 to 38764
Data columns (total 3 columns):
#   Column          Non-Null Count  Dtype  
---  -
0   Member_number    38765 non-null  int64  
1   Date             38765 non-null  object  
2   itemDescription  38765 non-null  object  
dtypes: int64(1), object(2)
memory usage: 908.7+ KB
None
```



The screenshot shows the same Google Colab notebook. The code cell contains the following Python code:

```
[ ] # mencetak 10 data pertama
print('First Ten Rows of the DataFrame: \n')
print(df.head(10))
```

The output of the code is displayed below the code cell:

```
First Ten Rows of the DataFrame:

   Member_number    Date    itemDescription
0         1808  21-07-15    tropical fruit
1         2552  01-05-15      whole milk
2         2300  19-09-15        pip fruit
3         1187  12-12-15  other vegetables
4         3037  02-01-15      whole milk
5         4941  14-02-15      rolls/buns
6         4501  05-08-15  other vegetables
7         3803  23-12-15      pot plants
8         2762  20-03-15      whole milk
9         4119  02-12-15    tropical fruit
```

Below the output, there is another code cell:

```
[ ] print('Unique itemDescription: ', df['itemDescription'].nunique())
```

The output of this code is displayed below it:

```
Unique itemDescription: 167
```

```
colab.research.google.com/drive/1juITXk9l4I-1rAqgtgtKCudsNGRiu4bl?usp=sharing#scrollTo=SISaoS7YJvDu

Kelompok 10 Market Basket Analysis
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text
[ ] #mencetak data itemDescription yang unik (memiliki nilai itemDescription berbeda)
print('Unique itemDescription: ', df['itemDescription'].nunique())
print('\n', df['itemDescription'].unique())

Unique itemDescription: 167

['tropical fruit' 'whole milk' 'pip fruit' 'other vegetables' 'rolls/buns'
'pot plants' 'citrus fruit' 'beer' 'frankfurter' 'chicken' 'butter'
'fruit/vegetable juice' 'packaged fruit/vegetables' 'chocolate'
'specialty bar' 'butter milk' 'bottled water' 'yogurt' 'sausage'
'brown bread' 'hamburger meat' 'root vegetables' 'pork' 'pastry'
'canned beer' 'berries' 'coffee' 'misc. beverages' 'ham' 'turkey'
'curd cheese' 'red/blush wine' 'frozen potato products' 'flour' 'sugar'
'frozen meals' 'herbs' 'soda' 'detergent' 'grapes' 'processed cheese'
'fish' 'sparkling wine' 'newspapers' 'curd' 'pasta' 'popcorn'
'finished products' 'beverages' 'bottled beer' 'dessert' 'dog food'
'specialty chocolate' 'condensed milk' 'cleaner' 'white wine' 'meat'
'ice cream' 'hard cheese' 'cream cheese' 'liquor' 'pickled vegetables'
'liquor (spirits)' 'beer milk' 'candy' 'onions' 'hair spray'
'photo/film' 'domestic eggs' 'margarine' 'shopping bags' 'salt' 'oil'
'whipped/sour cream' 'frozen vegetables' 'sliced cheese' 'dish cleaner'
'baking powder' 'specialty cheese' 'salty snack' 'instant food products'
'pet care' 'white bread' 'female sanitary products' 'cling film/bags'
'soap' 'frozen chicken' 'house keeping products' 'spread cheese'
'decalifier' 'frozen dessert' 'vinegar' 'nuts/prunes' 'potato products'
'frozen fish' 'hygiene articles' 'artif. sweetener' 'light bulbs'
'canned vegetables' 'chewing gum' 'canned fish' 'cookware'
'semi-finished bread' 'cat food' 'bathroom cleaner' 'prosecco'
'liver loaf' 'zwieback' 'canned fruit' 'frozen fruits' 'brandy'
'baby cosmetics' 'spices' 'napkins' 'waffles' 'sauces' 'rum'
'chocolate marshmallow' 'long life bakery product' 'bags' 'sweet spreads'
'soups' 'mustard' 'specialty fat' 'instant coffee' 'snack products'
'organic sausage' 'soft cheese' 'mayonnaise' 'dental care'
'roll products' 'kitchen towels' 'flower soil/fertilizer' 'cereals'
'meat spreads' 'dishes' 'male cosmetics' 'candles' 'whisky' 'tidbits'
'cooking chocolate' 'seasonal products' 'liqueur' 'abrasive cleaner'
'symp' 'ketchup' 'cream' 'skin care' 'rubbing alcohol' 'nut snacks'
'cocoa drinks' 'softener' 'organic products' 'cake bar' 'honey' 'jam'
'kitchen utensil' 'flower (seeds)' 'rice' 'tea' 'salad dressing'
'specialty vegetables' 'pudding powder' 'ready soups' 'make up remover'
'toilet cleaner' 'preservation products']
```

```
colab.research.google.com/drive/1juITXk9l4I-1rAqgtgtKCudsNGRiu4bl?usp=sharing#scrollTo=CndIRKOFYrkU

Kelompok 10 Market Basket Analysis
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text
specialty vegetables' 'pudding powder' 'ready soups' 'make up remover'
'toilet cleaner' 'preservation products']

[ ] # Cek missing value dan zeros
print(df.isnull().sum().sort_values(ascending=False))

Member_number    0
Date              0
itemDescription    0
dtype: int64

Terlihat bahwa pada dataset tidak terdapat missing value dan zeros

[ ] # cek nilai "NONE" pada kolom itemDescription
print(df[df['itemDescription']=='NONE'])

Empty DataFrame
Columns: [Member_number, Date, itemDescription]
Index: []

Tidak terdapat nilai 'NONE' di dataset kita.
```

Seperti yang bisa kita lihat di atas, fitur Tanggal dan Waktu bukanlah tipe numerik. Untuk visualisasi dan pemahaman data yang lebih baik, kita bisa menambahkan beberapa fitur lagi ke DataFrame ini berdasarkan informasi dari dua fitur ini.

```
colab.research.google.com/drive/1juITXk9I4l-1rAqgtgtKCudsNGRiu4bl?usp=sharing#scrollTo=rPSVH7n1asdz

Kelompok 10 Market Basket Analysis
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text
[ ] # Year (ingat kembali penulisan function lambda)
df['Year'] = df['Date'].apply(lambda x: x.split("-")[0])
# Month
df['Month'] = df['Date'].apply(lambda x: x.split("-")[1])
# Day
df['Day'] = df['Date'].apply(lambda x: x.split("-")[2])

[ ] # cek kembali perubahan yang terjadi setelah nilai tahun, bulan dan hari ditampung di fitur baru (year, month, day)
print(df.info())
print(df.head())

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 38765 entries, 0 to 38764
Data columns (total 6 columns):
#   Column          Non-Null Count  Dtype
---  ---
0   Member_number    38765 non-null  int64
1   Date             38765 non-null  object
2   itemDescription  38765 non-null  object
3   Year             38765 non-null  object
4   Month            38765 non-null  object
5   Day              38765 non-null  object
dtypes: int64(1), object(5)
memory usage: 1.8+ MB
None
   Member_number  Date      itemDescription  Year  Month  Day
0           1888  21-07-15      tropical fruit    21     07   15
1           2552  01-05-15        whole milk     01     05   15
2           2380  19-09-15         pip fruit     19     09   15
3           1187  12-12-15      other vegetables    12     12   15
4           3037  02-01-15        whole milk     02     01   15
```

4. Visualisasi dan Memahami Data

Kita tahu bahwa dataset ini direkam dari 01/01/2014 hingga 30/12/2015. Sebelum kita masuk dalam pemodelan, kita harus mengeksplorasi dan memvisualisasikan penjualan dalam periode waktu ini. Bahan makanan apa yang paling banyak dibeli pelanggan? Bulan mana yang lebih sukses? Mari kita jawab ini secara visual.

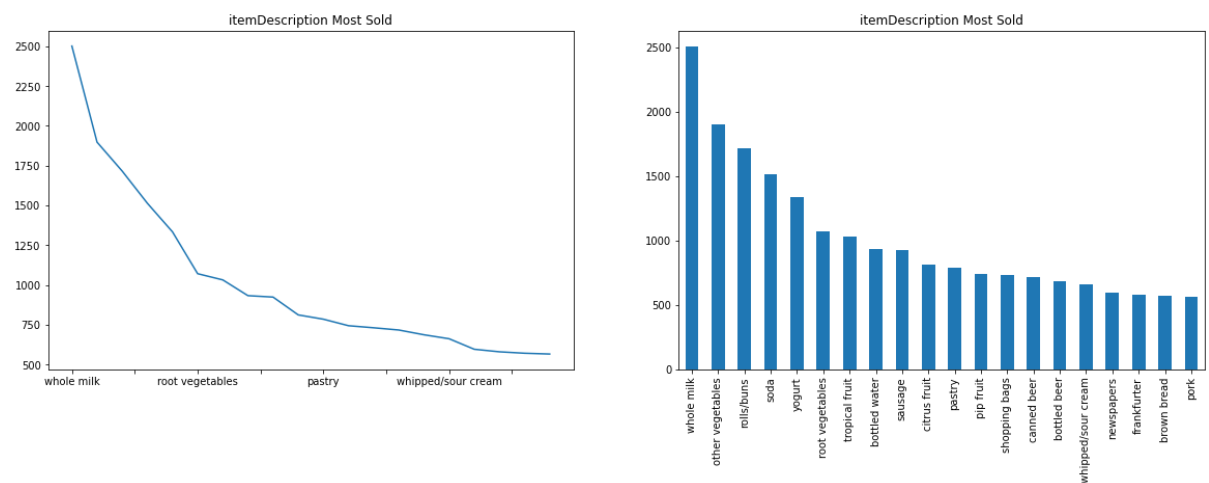
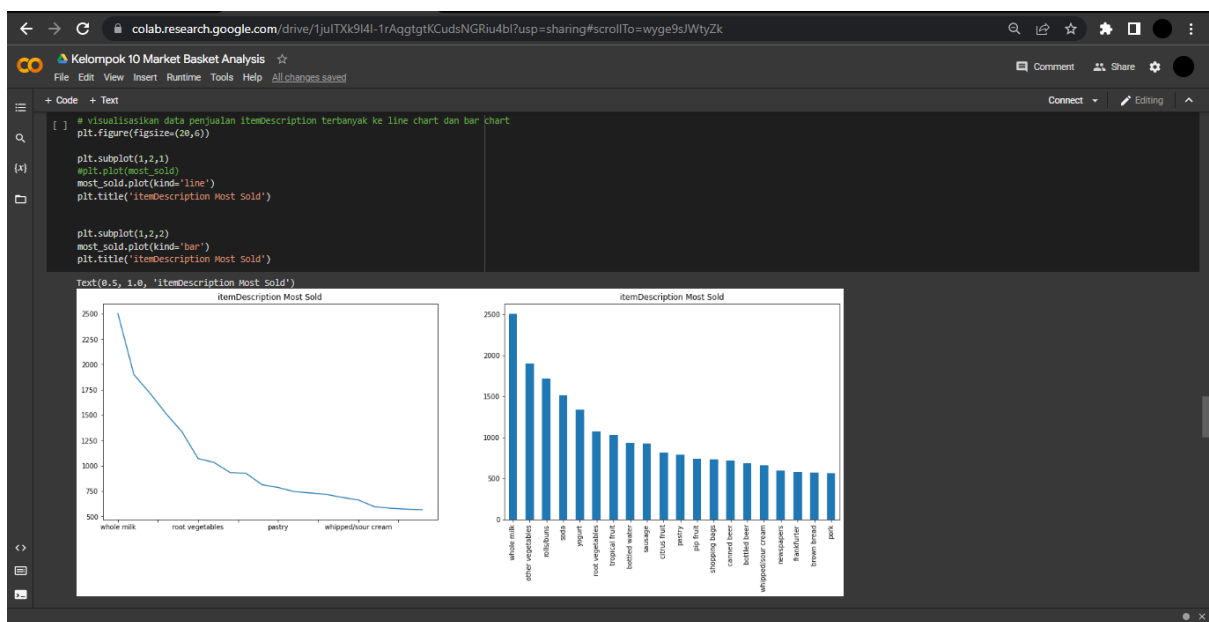
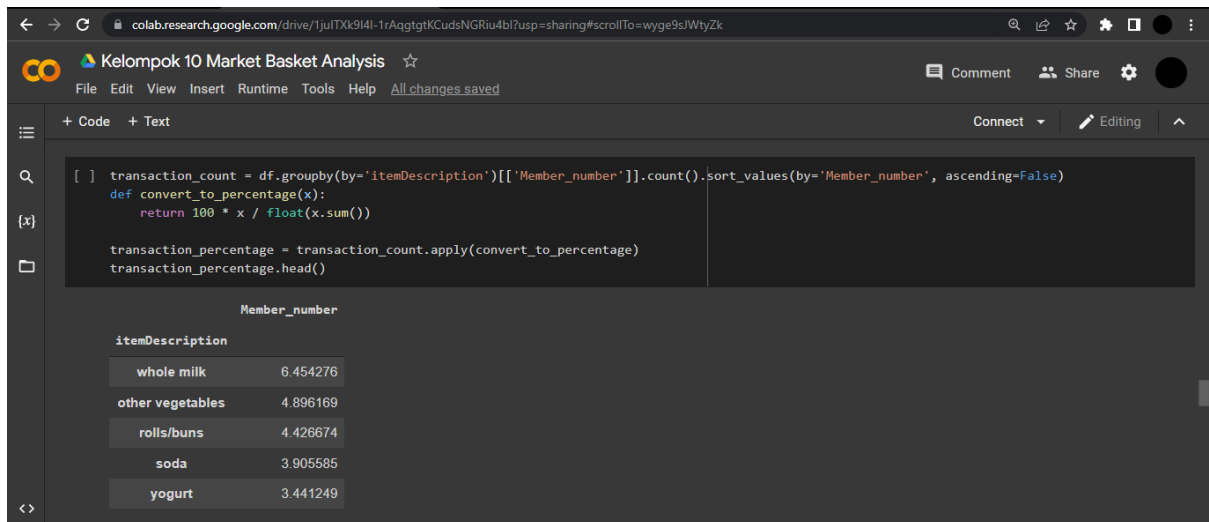
```
colab.research.google.com/drive/1juITXk9I4l-1rAqgtgtKCudsNGRiu4bl?usp=sharing#scrollTo=dJT5EJCeasS2

Kelompok 10 Market Basket Analysis
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text
[ ] # Mencetak 20 data penjualan itemDescription terbanyak
most_sold = df['itemDescription'].value_counts().head(20)

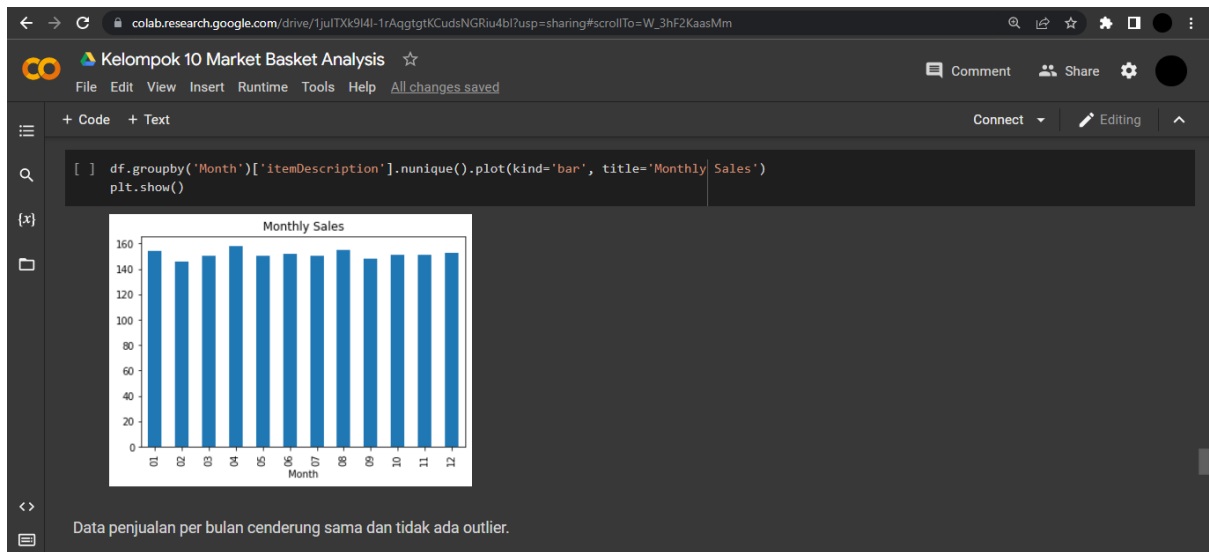
print('Most Sold itemDescription: \n')
print(most_sold)

Most Sold itemDescription:
whole milk          2502
other vegetables    1898
rolls/buns          1716
soda                1514
yogurt              1334
root vegetables     1871
tropical fruit       1832
bottled water        933
sausage             924
citrus fruit         812
pastry               785
pip fruit            744
shopping bags        731
canned beer          717
bottled beer         687
whipped/sour cream   662
newspapers           596
frankfurter          588
brown bread          571
pork                 566
Name: itemDescription, dtype: int64
```



Berdasarkan hasil visualisasi, whole milk adalah bahan makanan yang paling banyak terjual, diikuti oleh other vegetables, rolls/buns, soda, dan yogurt. Ini masuk akal

untuk toko groceries. Sekarang setelah kita mengetahui item mana yang paling populer, mari kita lihat bulan mana yang menghasilkan penjualan paling banyak.



The screenshot shows the same Google Colab notebook. The code cell contains the following Python code:

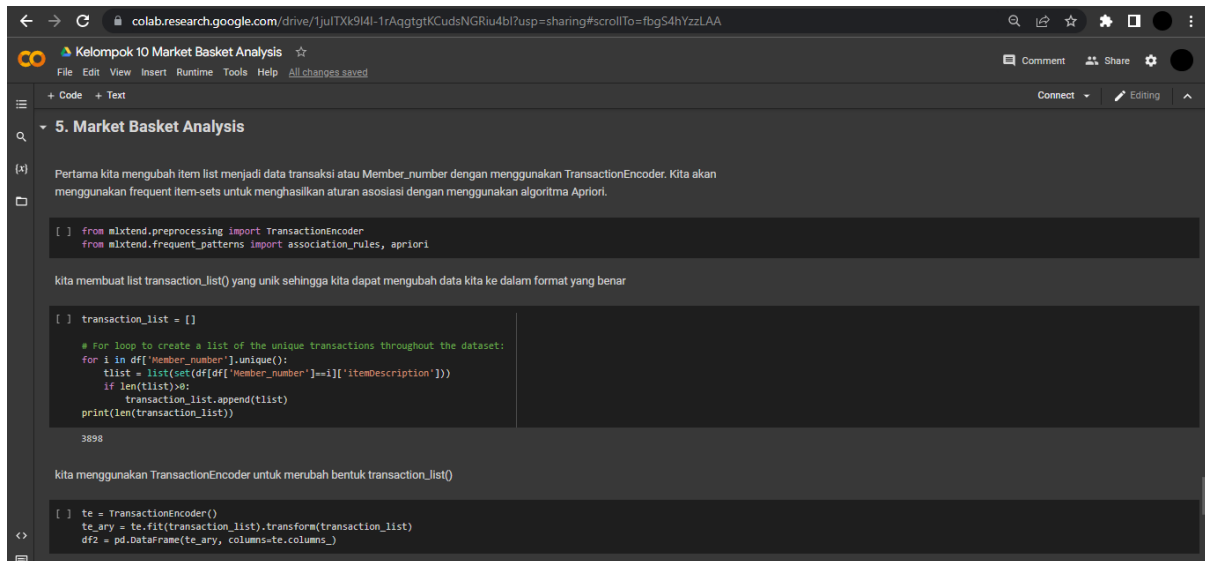
```
[ ] print(df.groupby('Month')['Day'].nunique())
```

The output is a table showing the number of unique days for each month:

Month	Day
01	2
02	2
03	2
04	2
05	2
06	2
07	2
08	2
09	2
10	2
11	2
12	2

Below the table, a text box states: "Pada semua bulan tercatat penjualan yang sama yaitu 2 hari."

5. Market Basket Analysis



```
[ ] from mlxtend.preprocessing import TransactionEncoder
from mlxtend.frequent_patterns import association_rules, apriori

kita membuat list transaction_list() yang unik sehingga kita dapat mengubah data kita ke dalam format yang benar

[ ] transaction_list = []

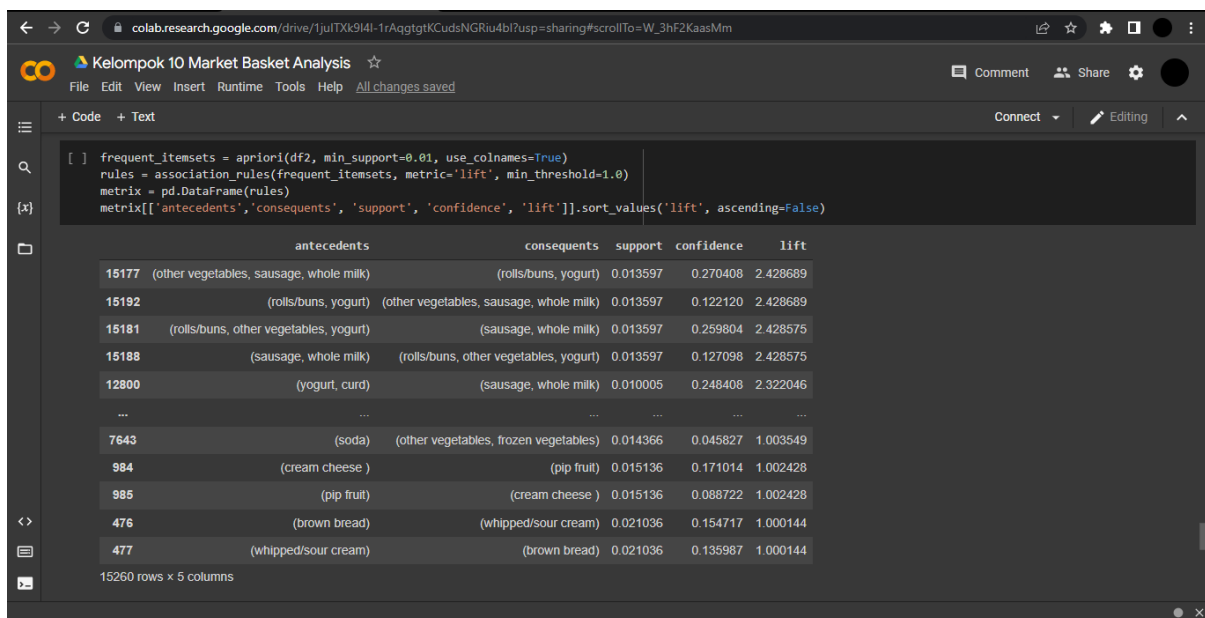
# For loop to create a list of the unique transactions throughout the dataset:
for i in df['Member_number'].unique():
    tlist = list(set(df[df['Member_number']==i]['itemDescription']))
    if len(tlist)>0:
        transaction_list.append(tlist)
    print(len(transaction_list))

3898

kita menggunakan TransactionEncoder untuk merubah bentuk transaction_list()

[ ] te = TransactionEncoder()
te_ary = te.fit(transaction_list).transform(transaction_list)
df2 = pd.DataFrame(te_ary, columns=te.columns_)
```

Sekarang kita terapkan Apriori. Kita akan menggunakan parameter `min_threshold` (nilai ambang batas yang ditentukan) dalam aturan asosiasi untuk metrik lift menjadi 1,0 karena jika kurang dari satu, maka kedua item tersebut kemungkinan tidak akan dibeli bersama. Kita akan mengurutkan nilai berdasarkan keyakinan untuk melihat kemungkinan suatu bahan makanan dibeli jika pendahulunya dibeli.



```
[ ] frequent_itemsets = apriori(df2, min_support=0.01, use_colnames=True)
rules = association_rules(frequent_itemsets, metric='lift', min_threshold=1.0)
matrix = pd.DataFrame(rules)
matrix[['antecedents', 'consequents', 'support', 'confidence', 'lift']].sort_values('lift', ascending=False)
```

	antecedents	consequents	support	confidence	lift
15177	(other vegetables, sausage, whole milk)	(rolls/buns, yogurt)	0.013597	0.270408	2.428689
15192	(rolls/buns, yogurt)	(other vegetables, sausage, whole milk)	0.013597	0.122120	2.428689
15181	(rolls/buns, other vegetables, yogurt)	(sausage, whole milk)	0.013597	0.259804	2.428575
15188	(sausage, whole milk)	(rolls/buns, other vegetables, yogurt)	0.013597	0.127098	2.428575
12800	(yogurt, curd)	(sausage, whole milk)	0.010005	0.248408	2.322046
...
7643	(soda)	(other vegetables, frozen vegetables)	0.014366	0.045827	1.003549
984	(cream cheese)	(pip fruit)	0.015136	0.171014	1.002428
985	(pip fruit)	(cream cheese)	0.015136	0.088722	1.002428
476	(brown bread)	(whipped/sour cream)	0.021036	0.154717	1.000144
477	(whipped/sour cream)	(brown bread)	0.021036	0.135987	1.000144

15260 rows x 5 columns

Kesimpulan

Dapat dilihat dari analisis di atas, di mana semakin tinggi nilai peningkatan, semakin kuat korelasi antar item. Data dengan jelas menunjukkan bahwa whole milk adalah item paling populer. Mari kita lihat korelasi item yang lebih menarik (format: *antecedent(s)* \Rightarrow *consequent*):

Other vegetables + sausage + whole milk \Rightarrow rolls/buns + yogurt

Rolls/buns + other vegetables + yogurt \Rightarrow sausage + whole milk

Yogurt + curd \Rightarrow whole milk + sausage

Bisnis selalu mencari cara untuk mengoptimalkan pengaturan mereka dan meningkatkan penjualan mereka. Karena kita sekarang mengetahui korelasi antara itemDescription dan kepentingan bersama pelanggan, bisnis dapat membuat keputusan berdasarkan temuan ini. Misalnya, toko groceries mungkin tertarik untuk mengadakan promosi itemDescription gratis, mengingat kemungkinan besar item lain dijual sebagai hasilnya (misalkan jika mereka mengadakan promo buy 1 get 1 khusus special event, itu mungkin tidak hanya menarik pelanggan baru yang sering datang, tetapi juga ada peluang yang sangat bagus bahwa pelanggan masih akan menghabiskan uang untuk whole milk).