tabular_data_06

October 1, 2024

1 call utility functions to get the analysis file

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import os

import dhs_util
from dhs_util import *

os.chdir('/Users/yingli/Development/TopicsInDataScience/')
df = pd.read_csv('dhs_service_records_synthesized_final.csv')

df = dhs_preprocessing(df)
df, service_map = add_service_label(df)
df = add_age_bin(df)

recipient = get_recipient_attribute(df)
```

1.1 preparing data for association rule mining

- prepare the transaction, i.e., for each recipient, make a list that contains all the services the recipient used

[2]:
```
serv_list = []
for groups in df.groupby('id').groups.values():
    serv_list.append(df.loc[groups]['serv'].tolist())
```

[3]:
```
for i in range(10):
    print(serv_list[i])
```

```
['S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12',
'S12']
['S12']
['S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12']
['S12', 'S12', 'S12', 'S12']
['S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12',
```

```
'S12']
['S09', 'S09', 'S09', 'S09', 'S09', 'S09', 'S11', 'S11', 'S11', 'S11', 'S11',
'S11', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12',
'S12', 'S12']
['S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12',
'S12']
['S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12',
'S12']
['S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12']
['S11', 'S11', 'S11', 'S11', 'S11', 'S11', 'S11', 'S11', 'S11', 'S11', 'S11',
'S11', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12',
'S12', 'S12']
```

[4]:
```
# another way to do the same
serv_list_n = []
for groups in df.groupby('id').groups.values():
    serv_list_n.append(list(df.loc[groups]['serv'].to_numpy()))

# can check equality
serv_list == serv_list_n
```

[4]:
```
True
```

[5]:
```
for i in range(10):
    print(serv_list[i])
```

```
['S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12',
'S12']
['S12']
['S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12']
['S12', 'S12', 'S12', 'S12']
['S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12',
'S12']
['S09', 'S09', 'S09', 'S09', 'S09', 'S09', 'S11', 'S11', 'S11', 'S11', 'S11',
'S11', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12',
'S12', 'S12']
['S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12',
'S12']
['S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12',
'S12']
['S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12']
['S11', 'S11', 'S11', 'S11', 'S11', 'S11', 'S11', 'S11', 'S11', 'S11', 'S11',
'S11', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12', 'S12',
'S12', 'S12']
```

1.1.1 use mlxtend package

- package has full documentation
- license: permissive BSD, allows for usage, even commercially usable

1.1.2 in-class work

- read through doc to install
- read through examples

[6]:
```
from mlxtend.preprocessing import TransactionEncoder
from mlxtend.preprocessing import *
from mlxtend.frequent_patterns import association_rules
from mlxtend.frequent_patterns import fpgrowth
from mlxtend.frequent_patterns import apriori
from mlxtend.frequent_patterns import fpmax
from mlxtend.frequent_patterns import hmine
```

[7]:
```
# re-do the prep of list of services again
serv_list = []
for groups in df.groupby('id').groups.values():
    serv_list.append(df.loc[groups]['serv'].tolist())

# following the tutorial example
def oneHotCoding(serv_list):
    te = TransactionEncoder()
    te_ary = te.fit(serv_list).transform(serv_list)
    te_df = pd.DataFrame(te_ary, columns=te.columns_)
    return te_df

serv_oneHot = oneHotCoding(serv_list)
```

[8]:
```
serv_oneHot
```

[8]:

	S01	S02	S03	S04	S05	S06	S07	S08	S09	S10	\
0	False	False	False	False	False	False	False	False	False	False	
1	False	False	False	False	False	False	False	False	False	False	
2	False	False	False	False	False	False	False	False	False	False	
3	False	False	False	False	False	False	False	False	False	False	
4	False	False	False	False	False	False	False	False	False	False	
…	…	…	…	…	…	…	…	…	…	…	
533794	False	False	False	False	False	False	False	False	True	False	
533795	False	False	False	False	False	False	False	False	True	False	
533796	False	False	False	False	False	False	False	False	True	False	
533797	False	False	False	False	False	True	False	False	True	False	
533798	False	False	False	False	False	False	False	False	False	False	

	…	S13	S14	S15	S16	S17	S18	S19	S20	S21	\
0	…	False	False	False	False	False	False	False	False	False	
1	…	False	False	False	False	False	False	False	False	False	
2	…	False	False	False	False	False	False	False	False	False	
3	…	False	False	False	False	False	False	False	False	False	
4	…	False	False	False	False	False	False	False	False	False	

```
…       …    …      …      …      …      …      …      …      …      …
533794  …  False   True  False  False  False  False  False  False  False
533795  …  False   True  False  False  False  False  False  False  False
533796  …  False   True  False  False  False  False  False  False  False
533797  …  False  False  False  False  False  False  False  False   True
533798  …  False  False  False  False  False  False  False  False  False

          S22
0       False
1       False
2       False
3       False
4       False
…       …
533794  False
533795  False
533796  False
533797  False
533798  False

[533799 rows x 22 columns]
```

- using groupby([“id”,“serv”]) and then oneHotCoding seem to take very long time
- recall we have done something like this earlier, when we transformed df into a matrix for computing correlations
- that code is cleaned up and put in our dhs_util module as a function “get_id_service_matrix”

[9]:
```
df_id_serv = get_id_service_matrix(df) # this gives number of times the service␣
 ↪is used
df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value␣
 ↪into True or False
```

```
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S01, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
```

```
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S03, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S04, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
```

```
533798    False
Name: S05, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S02, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797     True
533798    False
Name: S06, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
```

```
533795    False
533796    False
533797    False
533798    False
Name: S07, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S08, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794     True
533795     True
533796     True
533797     True
533798    False
Name: S09, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
```

```
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S10, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797     True
533798    False
Name: S11, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         True
1         True
2         True
3         True
4         True
          …
533794    True
533795    True
533796    True
533797    True
533798    True
Name: S12, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
```

```
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S13, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794     True
533795     True
533796     True
533797    False
533798    False
Name: S14, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S15, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
```

```
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S16, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S17, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S18, Length: 533799, dtype: bool' has dtype incompatible with int64,
```

```
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S19, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S20, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
```

```
533797     True
533798    False
Name: S21, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
/var/folders/j8/w88cxj05115byx3n9mnk572w0000gn/T/ipykernel_53160/3490586201.py:2
: FutureWarning: Setting an item of incompatible dtype is deprecated and will
raise in a future error of pandas. Value '0         False
1         False
2         False
3         False
4         False
          …
533794    False
533795    False
533796    False
533797    False
533798    False
Name: S22, Length: 533799, dtype: bool' has dtype incompatible with int64,
please explicitly cast to a compatible dtype first.
  df_id_serv.iloc[:,1:23] = df_id_serv.iloc[:,1:23] > 0  # this converts value
into True or False
```

- this was much faster thant the list operation and mlxtend oneHotCoding

[10]:
```
df_id_serv
```

[10]:
```
serv        id    S01    S03    S04    S05    S02    S06    S07    S08    S09  \
0            1  False  False  False  False  False  False  False  False  False
1            2  False  False  False  False  False  False  False  False  False
2            3  False  False  False  False  False  False  False  False  False
3            4  False  False  False  False  False  False  False  False  False
4            5  False  False  False  False  False  False  False  False  False
…          …    …    …    …    …    …    …    …    …    …
533794  535604  False  False  False  False  False  False  False  False   True
533795  535605  False  False  False  False  False  False  False  False   True
533796  535606  False  False  False  False  False  False  False  False   True
533797  535607  False  False  False  False  False   True  False  False   True
533798  535608  False  False  False  False  False  False  False  False  False

serv    …    S13    S14    S15    S16    S17    S18    S19    S20    S21  \
0       …  False  False  False  False  False  False  False  False  False
1       …  False  False  False  False  False  False  False  False  False
2       …  False  False  False  False  False  False  False  False  False
3       …  False  False  False  False  False  False  False  False  False
4       …  False  False  False  False  False  False  False  False  False
…     …    …    …    …    …    …    …    …    …    …
```

```
533794  …  False   True  False  False  False  False  False  False  False
533795  …  False   True  False  False  False  False  False  False  False
533796  …  False   True  False  False  False  False  False  False  False
533797  …  False  False  False  False  False  False  False  False   True
533798  …  False  False  False  False  False  False  False  False  False

serv      S22
0       False
1       False
2       False
3       False
4       False
…       …
533794  False
533795  False
533796  False
533797  False
533798  False

[533799 rows x 23 columns]
```

- one difference is that this dataframe has the "id" as column, not index
- we could turn it into an index or just use the other columns

[11]:
```
apriori(df_id_serv.iloc[:,1:23], use_colnames=True, min_support=0.01)\
    .sort_values(by="support", ascending=False)
```

[11]:
```
     support              itemsets
4   0.941422                 (S12)
2   0.153844                 (S09)
14  0.139131            (S12, S09)
15  0.103528            (S14, S09)
6   0.103528                 (S14)
25  0.094436       (S12, S14, S09)
20  0.094436            (S12, S14)
3   0.040882                 (S11)
18  0.032106            (S12, S11)
10  0.031396                 (S19)
7   0.024002                 (S15)
21  0.022561            (S12, S15)
16  0.019431            (S15, S09)
13  0.018307            (S11, S09)
26  0.018280       (S12, S15, S09)
24  0.016508       (S11, S12, S09)
0   0.013687                 (S03)
9   0.013573                 (S18)
17  0.013468            (S18, S09)
23  0.013460            (S18, S14)
```

```
28  0.013460       (S18, S14, S09)
11  0.013260                 (S21)
8   0.012967                 (S17)
1   0.012649                 (S05)
12  0.012284            (S03, S12)
5   0.011915                 (S13)
19  0.011448            (S13, S12)
22  0.010384            (S18, S12)
27  0.010285       (S18, S12, S09)
29  0.010277       (S18, S12, S14)
30  0.010277  (S18, S12, S14, S09)
```

- compare

[12]:
```
apriori(serv_oneHot, min_support=0.01, use_colnames=True)\
    .sort_values(by="support", ascending=False)
```

[12]:
```
     support              itemsets
4   0.941422                 (S12)
2   0.153844                 (S09)
14  0.139131            (S12, S09)
15  0.103528            (S14, S09)
6   0.103528                 (S14)
25  0.094436       (S12, S14, S09)
20  0.094436            (S12, S14)
3   0.040882                 (S11)
18  0.032106            (S12, S11)
10  0.031396                 (S19)
7   0.024002                 (S15)
21  0.022561            (S12, S15)
16  0.019431            (S15, S09)
13  0.018307            (S11, S09)
26  0.018280       (S12, S15, S09)
24  0.016508       (S11, S12, S09)
0   0.013687                 (S03)
9   0.013573                 (S18)
17  0.013468            (S18, S09)
23  0.013460            (S18, S14)
28  0.013460       (S18, S14, S09)
11  0.013260                 (S21)
8   0.012967                 (S17)
1   0.012649                 (S05)
12  0.012284            (S03, S12)
5   0.011915                 (S13)
19  0.011448            (S13, S12)
22  0.010384            (S18, S12)
27  0.010285       (S18, S12, S09)
29  0.010277       (S18, S12, S14)
```

```
30  0.010277  (S18, S12, S14, S09)
```

[13]:
```
min_freq = 1000 # if we want to set threshold by frequency of the itemsets
min_support = min_freq/serv_oneHot.shape[0]
min_confidence = 0.6
min_rule_support = 0.2
min_lift = 0.15
```

[33]:
```
min_support
```

[33]:
```
0.001873364318779166
```

[14]:
```
apriori(serv_oneHot, min_support=min_support,use_colnames=True)\
    .sort_values(by="support", ascending=False)
```

[14]:
```
      support              itemsets
9    0.941422                 (S12)
6    0.153844                 (S09)
37   0.139131            (S12, S09)
11   0.103528                 (S14)
39   0.103528            (S14, S09)
..        …                     …
50   0.001931            (S18, S11)
131  0.001930  (S11, S18, S14, S09)
110  0.001930       (S18, S14, S11)
72   0.001875       (S14, S02, S09)
23   0.001875            (S02, S14)

[146 rows x 2 columns]
```

[15]:
```
freq_itemset_apriori =␣
 ↪apriori(serv_oneHot,min_support=min_support,use_colnames=True)
freq_itemset_apriori.describe()
```

[15]:
```
          support
count  146.000000
mean     0.017443
std      0.080320
min      0.001875
25%      0.002868
50%      0.004436
75%      0.009374
max      0.941422
```

[16]:
```
freq_itemset_fpgrowth =␣
 ↪fpgrowth(serv_oneHot,min_support=min_support,use_colnames=True)
freq_itemset_fpgrowth.describe()
```

[16]:
```
           support
count  146.000000
mean     0.017443
std      0.080320
min      0.001875
25%      0.002868
50%      0.004436
75%      0.009374
max      0.941422
```

[17]:
```
freq_itemset_fpmax =␣
 ↪fpmax(serv_oneHot,min_support=min_support,use_colnames=True)
freq_itemset_fpmax.describe()
```

[17]:
```
          support
count  22.000000
mean    0.003213
std     0.001915
min     0.001875
25%     0.002171
50%     0.002484
75%     0.003357
max     0.009342
```

[18]:
```
# compute and print the association rules
def serv_rules(freq_itemsets,metrics,threshold):
    asso_rules = association_rules(freq_itemsets, metric=metrics,␣
 ↪min_threshold=threshold)
    return asso_rules.sort_values(by='lift', ascending=False)[['antecedents',␣
 ↪'consequents', 'support', 'confidence', 'lift']]

rule_apriori = serv_rules(freq_itemset_apriori,"confidence",0.60)
rule_fpgrowth = serv_rules(freq_itemset_fpgrowth,"confidence",0.60)
```

[19]:
```
rule_fpgrowth
```

[19]:
```
          antecedents consequents   support  confidence       lift
126       (S06, S09)  (S12, S21)  0.002486    0.857789  96.948349
125       (S12, S06)  (S21, S09)  0.002486    0.722767  90.929116
209       (S02, S09)  (S03, S12)  0.002182    0.969218  78.903093
204            (S02)  (S03, S12)  0.003921    0.935628  76.168573
123  (S12, S06, S09)       (S21)  0.002486    0.987351  74.462712
..               …          …        …           …         …
117       (S21, S06)       (S12)  0.003215    0.701554   0.745206
83             (S17)       (S12)  0.008674    0.668882   0.710502
102            (S21)       (S12)  0.008848    0.667279   0.708799
26        (S19, S09)       (S12)  0.003432    0.651494   0.692032
```

```
175            (S05)       (S12)  0.008057    0.636996    0.676632

[212 rows x 5 columns]
```

[20]:
```
hmine(serv_oneHot,min_support=0.0001,use_colnames=True)
```

[20]:
```
      support         itemsets
0    0.000654            (S01)
1    0.000305       (S12, S01)
2    0.004191            (S02)
3    0.004142       (S03, S02)
4    0.000146  (S04, S03, S02)
..        …              …
930  0.000141       (S21, S19)
931  0.002471            (S20)
932  0.000126       (S20, S21)
933   0.01326            (S21)
934  0.002205            (S22)

[935 rows x 2 columns]
```

[21]:
```
fpmax(serv_oneHot,min_support=0.0001,use_colnames=True)
```

[21]:
```
     support                         itemsets
0   0.000199                       (S08, S12)
1   0.000305                       (S12, S01)
2   0.000124                       (S22, S12)
3   0.000133            (S02, S03, S12, S04)
4   0.000131       (S12, S04, S03, S14, S09)
..       …                                …
59  0.000122  (S12, S21, S11, S14, S09, S18)
60  0.000210  (S12, S21, S14, S09, S15, S18)
61  0.000204       (S12, S14, S09, S19, S18)
62  0.000169       (S12, S14, S09, S19, S15)
63  0.000247       (S12, S11, S14, S09, S19)

[64 rows x 2 columns]
```

[22]:
```
freq_itemset_fpgrowth = fpmax(serv_oneHot,min_support=0.0001,use_colnames=True)
asso_rules = association_rules(freq_itemset_fpgrowth, metric="support",␣
 ↪min_threshold=0.0003,support_only=True)
asso_rules.sort_values(by='lift', ascending=False)[['antecedents',␣
 ↪'consequents', 'support', 'confidence', 'lift']]
```

[22]:
```
                antecedents                consequents   support  \
0                      (S12)                      (S01)  0.000305
1                      (S01)                      (S12)  0.000305
```

```
2   (S12, S09, S14, S15, S18)                       (S17)  0.000326
3   (S12, S09, S14, S15, S17)                       (S18)  0.000326
4   (S12, S09, S14, S18, S17)                       (S15)  0.000326
..                        …                         …       …
59                    (S14)  (S12, S09, S15, S18, S17)  0.000326
60                    (S09)  (S12, S14, S15, S18, S17)  0.000326
61                    (S15)  (S12, S09, S14, S18, S17)  0.000326
62                    (S18)  (S12, S09, S14, S15, S17)  0.000326
63                    (S17)  (S12, S09, S14, S15, S18)  0.000326

    confidence  lift
0          NaN   NaN
1          NaN   NaN
2          NaN   NaN
3          NaN   NaN
4          NaN   NaN
..         …   …
59         NaN   NaN
60         NaN   NaN
61         NaN   NaN
62         NaN   NaN
63         NaN   NaN

[64 rows x 5 columns]
```

[23]:
```
def predict(antecedent, rules, consequents_only = False):
    # get the rules for this antecedent
    preds = rules[rules['antecedents'] == antecedent]
    if consequents_only:
        # a way to convert a frozen set with one element to string
        preds = preds['consequents'].apply(iter).apply(next)
    return preds
```

[24]:
```
rule_fpmax = association_rules(freq_itemset_fpmax, metric="confidence",␣
 ↪min_threshold=0.001, support_only=True)
```

[25]:
```
predict({"S06"}, rule_fpmax, consequents_only=False)
```

[25]:
```
   antecedents      consequents  antecedent support  consequent support  \
38       (S06)  (S12, S21, S09)                 NaN                 NaN

     support  confidence  lift  leverage  conviction  zhangs_metric
38  0.002486         NaN   NaN       NaN         NaN            NaN
```

[26]:
```
predict({"S09"}, rule_fpmax, consequents_only=False)
```

[26]:

	antecedents	consequents	antecedent support	consequent support	\
5	(S09)	(S02, S14)	NaN	NaN	
19	(S09)	(S03, S12, S02)	NaN	NaN	
25	(S09)	(S12, S10)	NaN	NaN	
39	(S09)	(S12, S21, S06)	NaN	NaN	
67	(S09)	(S12, S14, S07, S11)	NaN	NaN	
98	(S09)	(S18, S12, S16, S14)	NaN	NaN	
113	(S09)	(S13, S12, S14)	NaN	NaN	
129	(S09)	(S17, S15, S14)	NaN	NaN	
143	(S09)	(S12, S17, S15)	NaN	NaN	
157	(S09)	(S12, S17, S14)	NaN	NaN	
171	(S09)	(S12, S21, S15)	NaN	NaN	
185	(S09)	(S12, S21, S14)	NaN	NaN	
199	(S09)	(S18, S14, S11)	NaN	NaN	
213	(S09)	(S03, S12, S14)	NaN	NaN	
227	(S09)	(S12, S15, S11)	NaN	NaN	
241	(S09)	(S12, S15, S14)	NaN	NaN	
247	(S09)	(S19, S14)	NaN	NaN	
253	(S09)	(S12, S19)	NaN	NaN	

	support	confidence	lift	leverage	conviction	zhangs_metric
5	0.001875	NaN	NaN	NaN	NaN	NaN
19	0.002182	NaN	NaN	NaN	NaN	NaN
25	0.002754	NaN	NaN	NaN	NaN	NaN
39	0.002486	NaN	NaN	NaN	NaN	NaN
67	0.002055	NaN	NaN	NaN	NaN	NaN
98	0.002879	NaN	NaN	NaN	NaN	NaN
113	0.004436	NaN	NaN	NaN	NaN	NaN
129	0.001982	NaN	NaN	NaN	NaN	NaN
143	0.002482	NaN	NaN	NaN	NaN	NaN
157	0.003945	NaN	NaN	NaN	NaN	NaN
171	0.002167	NaN	NaN	NaN	NaN	NaN
185	0.002868	NaN	NaN	NaN	NaN	NaN
199	0.001930	NaN	NaN	NaN	NaN	NaN
213	0.003662	NaN	NaN	NaN	NaN	NaN
227	0.003130	NaN	NaN	NaN	NaN	NaN
241	0.009342	NaN	NaN	NaN	NaN	NaN
247	0.002093	NaN	NaN	NaN	NaN	NaN
253	0.003432	NaN	NaN	NaN	NaN	NaN

[27]:
```
predict({"S06"}, rule_fpgrowth)
```

[27]:

	antecedents	consequents	support	confidence	lift
115	(S06)	(S21)	0.004582	0.784477	59.162639

[28]:
```
serv_list =  ['S'+str(i).zfill(2) for i in range(1,23)]
for i in serv_list:
```

```
    print(i)
    if (len(predict({i},rule_fpgrowth))>0):
        print(i), print(predict({i},rule_fpgrowth))
```

```
S01
S02
S02
    antecedents consequents   support  confidence       lift
204       (S02)  (S03, S12)  0.003921    0.935628  76.168573
200       (S02)       (S03)  0.004142    0.988377  72.213908
201       (S02)       (S12)  0.003970    0.947251   1.006192
S03
S03
   antecedents consequents   support  confidence      lift
16       (S03)       (S12)  0.012284    0.897482  0.953326
S04
S05
S05
    antecedents consequents   support  confidence      lift
175       (S05)       (S12)  0.008057    0.636996  0.676632
S06
S06
    antecedents consequents   support  confidence       lift
115       (S06)       (S21)  0.004582    0.784477  59.162639
S07
S07
    antecedents consequents   support  confidence       lift
141       (S07)  (S12, S11)  0.005105     0.79795  24.853835
137       (S07)       (S11)  0.006398     1.00000  24.460386
138       (S07)       (S12)  0.005105     0.79795   0.847601
S08
S09
S09
   antecedents consequents   support  confidence      lift
15       (S09)  (S12, S14)  0.094436    0.613843  6.500073
9        (S09)       (S14)  0.103528    0.672938  6.500073
0        (S09)       (S12)  0.139131    0.904362  0.960634
S10
S10
   antecedents consequents   support  confidence      lift
43       (S10)       (S12)  0.005238     0.93637  0.994634
S11
S11
  antecedents consequents   support  confidence      lift
1       (S11)       (S12)  0.032106    0.785318  0.834183
S12
S13
S13
```

```
    antecedents consequents   support  confidence      lift
131       (S13)  (S12, S09)  0.008153    0.684277  4.918218
128       (S13)       (S09)  0.008419    0.706604  4.592976
127       (S13)       (S12)  0.011448    0.960849  1.020636
S14
S14
   antecedents consequents   support  confidence      lift
14       (S14)  (S12, S09)  0.094436    0.912184  6.556292
8        (S14)       (S09)  0.103528    1.000000  6.500073
10       (S14)       (S12)  0.094436    0.912184  0.968942
S15
S15
   antecedents consequents   support  confidence      lift
32       (S15)  (S12, S09)  0.018280    0.761630  5.474191
29       (S15)       (S09)  0.019431    0.809554  5.262157
28       (S15)       (S12)  0.022561    0.939978  0.998467
S16
S16
   antecedents      consequents   support  confidence      lift
50       (S16)       (S09, S14)  0.009371    0.998403  9.643824
46       (S16)            (S14)  0.009371    0.998403  9.643824
56       (S16)       (S12, S14)  0.006815    0.726148  7.689286
63       (S16)  (S09, S12, S14)  0.006815    0.726148  7.689286
45       (S16)            (S09)  0.009372    0.998603  6.490991
53       (S16)       (S12, S09)  0.006817    0.726347  5.220599
47       (S16)            (S12)  0.006830    0.727745  0.773027
S17
S17
   antecedents consequents   support  confidence      lift
83       (S17)       (S12)  0.008674    0.668882  0.710502
S18
S18
    antecedents      consequents   support  confidence      lift
181       (S18)       (S09, S14)  0.013460    0.991718  9.579254
177       (S18)            (S14)  0.013460    0.991718  9.579254
194       (S18)  (S09, S12, S14)  0.010277    0.757212  8.018229
187       (S18)       (S12, S14)  0.010277    0.757212  8.018229
176       (S18)            (S09)  0.013468    0.992271  6.449831
184       (S18)       (S12, S09)  0.010285    0.757764  5.446406
178       (S18)            (S12)  0.010384    0.765079  0.812685
S19
S20
S21
S21
    antecedents consequents   support  confidence      lift
102       (S21)       (S12)  0.008848    0.667279  0.708799
S22
```

[29]:
```
rule_fpgrowth.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 212 entries, 126 to 175
Data columns (total 5 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   antecedents  212 non-null    object
 1   consequents  212 non-null    object
 2   support      212 non-null    float64
 3   confidence   212 non-null    float64
 4   lift         212 non-null    float64
dtypes: float64(3), object(2)
memory usage: 9.9+ KB
```

[30]:
```
rule_fpgrowth.sort_values("support",ascending=False)
```

[30]:

	antecedents	consequents	support	confidence	lift
0	(S09)	(S12)	0.139131	0.904362	0.960634
8	(S14)	(S09)	0.103528	1.000000	6.500073
9	(S09)	(S14)	0.103528	0.672938	6.500073
14	(S14)	(S12, S09)	0.094436	0.912184	6.556292
11	(S12, S14)	(S09)	0.094436	1.000000	6.500073
..	…	…	…	…	…
196	(S18, S11)	(S14)	0.001930	0.999030	9.649879
197	(S18, S14, S11)	(S09)	0.001930	1.000000	6.500073
199	(S18, S11)	(S09, S14)	0.001930	0.999030	9.649879
210	(S02, S14)	(S09)	0.001875	1.000000	6.500073
211	(S02, S09)	(S14)	0.001875	0.832779	8.044016

[212 rows x 5 columns]

[31]:
```
df[(df.serv=="S09")].merge(df[(df.serv =="S12")], on = "id").id.nunique()/df.id.
↪nunique()
```

[31]:
```
0.1391310212270911
```

[32]:
```
predict({"S11", "S18"}, rule_fpgrowth)
```

[32]:

	antecedents	consequents	support	confidence	lift
196	(S18, S11)	(S14)	0.001930	0.99903	9.649879
199	(S18, S11)	(S09, S14)	0.001930	0.99903	9.649879
195	(S18, S11)	(S09)	0.001931	1.00000	6.500073

[ ]: