Data Mining Phase 3

Faridreza Momtazzandi 9812762601 Alireza Nourbakhsh 9812762496

After all the knowledge we got from our previous two phases now we have enough information to finally get some specific insight on how we can use our analysis result.

In this phase we're asked to find out about three different objectives in our datasets.

1. Incoming rate of assets

Our first task is to Find the ratio of incoming assets based on year which can be Fiscal year or normal year.

As we said before we'll be focusing on INOUT and INOUTLINE datasets which mostly has the information on ingoing and outgoing assetss. So let's figure out which year we want to work on.

If we want to use a fiscal year, unfortunately, most of our records don't have value and are null so another way we can deal with year is to use CREATED feature which we have in both INOUT and INOUTLINE so let's get to coding for a bit.

In [1]:
```
import pandas as pd
```

In [2]:
```
inout_df = pd.read_csv('E:\ce\data mining\DataSets\INOUT.csv' , low_memory = False)
inoutline_df = pd.read_csv('E:\ce\data mining\DataSets\INOUTLINE.csv' , low_memory
```

In [3]:
```
year_2015 = 0
year_2016 = 0
year_2017 = 0
year_2018 = 0
year_2019 = 0
year_2020 = 0
year_2021 = 0
year_2022 = 0

for ind in inout_df.index:
    if '2015' in inout_df['CREATED'][ind]:
        year_2015 += 1
    elif '2016' in inout_df['CREATED'][ind]:
        year_2016 += 1
    elif '2017' in inout_df['CREATED'][ind]:
        year_2017 += 1
    elif '2018' in inout_df['CREATED'][ind]:
        year_2018 += 1
    elif '2019' in inout_df['CREATED'][ind]:
        year_2019 += 1
    elif '2020' in inout_df['CREATED'][ind]:
        year_2020 += 1
    elif '2021' in inout_df['CREATED'][ind]:
        year_2021 += 1
    elif '2022' in inout_df['CREATED'][ind]:
        year_2022 += 1
print('\033[94m Lets see how much asset came in on each year and how we documented
print('\033[90m')
print('in 2015:' , year_2015)
print('in 2016:' , year_2016)
print('in 2017:' , year_2017)
print('in 2018:' , year_2018)
print('in 2019:' , year_2019)
print('in 2020:' , year_2020)
print('in 2021:' , year_2021)
print('in 2022:' , year_2022)
```

```
 Lets see how much asset came in on each year and how we documented it in INOUT dat
aset:

in 2015: 1071
in 2016: 947
in 2017: 1291
in 2018: 1208
in 2019: 851
in 2020: 992
in 2021: 727
in 2022: 512
```

In [4]:
```
print('\033[94m Now we can do the same thing for our INOUTLINE dataset:')
print('\033[90m')
year_2013 = 0
year_2014 = 0
year_2015 = 0
year_2016 = 0
year_2017 = 0
year_2018 = 0
year_2019 = 0
year_2020 = 0
year_2021 = 0
year_2022 = 0

for ind in inoutline_df.index:
    if '2013' in inoutline_df['CREATED'][ind]:
        year_2013 += 1
    elif '2014' in inoutline_df['CREATED'][ind]:
        year_2014 += 1
    elif '2015' in inoutline_df['CREATED'][ind]:
        year_2015 += 1
    elif '2016' in inoutline_df['CREATED'][ind]:
        year_2016 += 1
    elif '2017' in inoutline_df['CREATED'][ind]:
        year_2017 += 1
    elif '2018' in inoutline_df['CREATED'][ind]:
        year_2018 += 1
    elif '2019' in inoutline_df['CREATED'][ind]:
        year_2019 += 1
    elif '2020' in inoutline_df['CREATED'][ind]:
        year_2020 += 1
    elif '2021' in inoutline_df['CREATED'][ind]:
        year_2021 += 1
    elif '2022' in inoutline_df['CREATED'][ind]:
        year_2022 += 1
print('in 2013:' , year_2013)
print('in 2014:' , year_2014)
print('in 2015:' , year_2015)
print('in 2016:' , year_2016)
print('in 2017:' , year_2017)
print('in 2018:' , year_2018)
print('in 2019:' , year_2019)
print('in 2020:' , year_2020)
print('in 2021:' , year_2021)
print('in 2022:' , year_2022)
```

Now we can do the same thing for our INOUTLINE dataset:

```
in 2013: 1
in 2014: 0
in 2015: 3326
in 2016: 4448
in 2017: 8245
in 2018: 6228
in 2019: 5374
in 2020: 9191
in 2021: 5227
in 2022: 61370
```

After getting all the incoming assets based on year we can analyze the result:

First, after a quick look at the result, we see almost all the assets in both datasets had been registered after 2015 and only one asset was registered before that so we can guess it's a noise and a mistake and probably best to just ignore it.

Also, we can get some useful information about our datasets such as the years which had the most incoming assets which are 2017 and 2018 in the INOUT data set and by a long difference 2022 in the INOUTLINE dataset.

2. Count and Value of assets based on Holding

Now in the second part of this phase, we need to get statistical knowledge on how each of our holdings is doing so we're trying to examine both INOUT and INOUTLINE databases on the value and count of registered data.

For this part, we're trying to look for the ACCT_AC_HOLDING_ID feature which is the id number of the holding responsible for some assets and we examine it to see how each record of this feature represents a holding. We can also use the VAHED_MALI feature to check what accounting unit did the accounting for these assets.

In [5]:
```
print('\033[94m All the holdings in INOUT dataset asset counts are:')
print('\033[90m')
print(inout_df['ACCT_AC_HOLDING_ID'].value_counts())

print('\033[94m All the accounting units in INOUT dataset asset counts are:')
print('\033[90m')
print(inout_df['VAHED_MALI'].value_counts())

print('\033[94m All the accounting units in INOUTLINE dataset asset counts are:')
print('\033[90m')
print(inoutline_df['VAHED_MALI'].value_counts())
```

All the holdings in INOUT dataset asset counts are:

```
1.0     7180
3.0      289
4.0       86
5.0       24
21.0      19
Name: ACCT_AC_HOLDING_ID, dtype: int64
```

All the accounting units in INOUT dataset asset counts are:

```
200000138.0    6472
469638358.0     154
200000141.0      59
200000130.0      46
469638568.0      43
200000192.0      41
469638115.0      36
200000114.0      24
469678455.0      18
210000001.0      15
200000142.0      15
200000679.0       8
200000491.0       7
200000696.0       5
469718709.0       5
200000538.0       4
200000529.0       3
1.0               1
200000599.0       1
200000523.0       1
200000143.0       1
469638210.0       1
200000531.0       1
469638544.0       1
200000471.0       1
Name: VAHED_MALI, dtype: int64
```

All the accounting units in INOUTLINE dataset asset counts are:

```
200000138.0    77242
200000141.0     7226
469678455.0     3421
210000001.0     2104
469638358.0      671
469638568.0      221
200000143.0      189
200000679.0      158
200000192.0      151
469638115.0       76
200000130.0       70
210000112.0       63
200000114.0       58
200000529.0       40
200000142.0       34
200000538.0       25
200000491.0       20
469637613.0       16
```

```
200000546.0       11
200000696.0       10
200000135.0        9
200000548.0        9
469638544.0        6
200000523.0        5
200000542.0        5
200000531.0        5
469718709.0        5
200000471.0        5
200000550.0        3
200000484.0        3
200000525.0        1
200000599.0        1
469638210.0        1
Name: VAHED_MALI, dtype: int64
```

Let's calculate the values for each holding in INOUT dataset and 5 most accounting units in both datasets, but unfortunately, there is no book value in INOUT dataset so we have to move on to INOUTLINE dataset:

In [6]:
```
accounting_unit_200000138 = 0
accounting_unit_469638358 = 0
accounting_unit_200000141 = 0
accounting_unit_200000130 = 0
accounting_unit_469638568 = 0

print('\033[94m With the code down below we try to calculate the sum of book values
print('But unfortunatly as we try to get book values we see most of them are NaN or
print('so there is no usefull data to get from them we print accounting_unit_469638

print('\033[90m')

for ind in inoutline_df.index:
    if (inoutline_df['VAHED_MALI'][ind] == 200000138):
        accounting_unit_200000138 = accounting_unit_200000138 + inoutline_df['BOOKV
    elif inoutline_df['VAHED_MALI'][ind] == 469638358:
        accounting_unit_469638358 = accounting_unit_469638358 + inoutline_df['BOOKV
    elif inoutline_df['VAHED_MALI'][ind] == 200000141:
        accounting_unit_200000141 = accounting_unit_200000141 + inoutline_df['BOOKV
    elif inoutline_df['VAHED_MALI'][ind] == 200000130:
        accounting_unit_200000130 = accounting_unit_200000130 + inoutline_df['BOOKV
    elif inoutline_df['VAHED_MALI'][ind] == 469638568:
        accounting_unit_469638568 = accounting_unit_469638568 + inoutline_df['BOOKV
        print(inoutline_df['BOOKVALUE'][ind])
```

With the code down below we try to calculate the sum of book values base on accounting units
But unfortunatly as we try to get book values we see most of them are NaN or null so there is no usefull data to get from them we print accounting_unit_469638568 book values to show this

```
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
```

```
0.0
0.0
0.0
0.0
0.0
0.0
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
```

```
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
0.0
```

```
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
nan
```

```
nan
nan
nan
```

3. Draft or finalized accounting document

For this part, we need to check INOUT dataset and look for the C_DOCSTATUS_ID feature. This feature has three values: 3000025 for finalized, 3000006 for drafted, and 3000018 for waiting. We need to check each one of them to understand the state of each asset's accounting document

In [7]:
```
print(inout_df['C_DOCSTATUS_ID'].value_counts())
```

```
3000025    7583
3000006      14
3000018       1
6000035       1
3309          1
Name: C_DOCSTATUS_ID, dtype: int64
```

As we saw in the result of the code above 7583 assets have been finalized and 14 assets are drafted and only 1 asset is in the waiting process, also 2 assets have the C_DOCSTATUS_ID value of 6000035 and 3309 which we don't know the meaning of so we can assume are mistaken.

The only thing left in this phase is to get the list of these three states of accounting documents which we get in the below codes and after putting them into separated data frames we print the newly made data frames.

In [8]:
```
fialized = inout_df.loc[inout_df["C_DOCSTATUS_ID"] == 3000025]
print(fialized)
drafted = inout_df.loc[inout_df["C_DOCSTATUS_ID"] == 3000006]
print(drafted)
waiting = inout_df.loc[inout_df["C_DOCSTATUS_ID"] == 3000018]
print(waiting)
```

```
       INOUT_ID  AD_CLIENT_ID  AD_ORG_ID ISACTIVE          CREATED  CREATEDBY  \
0     469637755     104000002          0        Y   6/28/2015 7:50  210619032
1     469637756     104000002          0        Y   6/28/2015 7:50  210619032
2     469637757     104000002          0        Y   6/28/2015 7:50  210619032
3     469637758     104000002          0        Y   6/28/2015 7:50  210619032
4     469637759     104000002          0        Y   6/28/2015 7:50  210619032
...         ...           ...        ...      ...              ...        ...
7593  469647795     104000002          0        Y   9/8/2022 10:05  469637406
7594  469647814     104000002          0        Y   9/12/2022 9:16  469637406
7595  469647815     104000002          0        Y  9/12/2022 10:21  469637406
7597  469647817     104000002          0        Y  9/12/2022 11:22  469637406
7598  469647818     104000002          0        Y  9/12/2022 12:59  469637406

              UPDATED  UPDATEDBY  NAME DESCRIPTION  ... SUB_REF_ORG  STRING3  \
0     11/8/2017 11:15  210033614   NaN          69  ...         NaN      NaN
1      6/28/2015 7:50  210619032   NaN          73  ...         NaN      NaN
2      6/28/2015 7:50  210619032   NaN         206  ...         NaN      NaN
3      6/28/2015 7:50  210619032   NaN        9405  ...         NaN      NaN
4      6/28/2015 7:50  210619032   NaN         346  ...         NaN      NaN
...               ...        ...   ...         ...  ...         ...      ...
7593   9/8/2022 10:07  469637406   NaN         NaN  ...   265272107    24/15
7594   9/12/2022 9:17  469637406   NaN         NaN  ...   265272107    24/15
7595  9/12/2022 10:24  469637406   NaN         NaN  ...   265272107        1
7597  9/12/2022 11:33  469637406   NaN         NaN  ...   265272107    24/72
7598  9/12/2022 13:01  469637406   NaN         NaN  ...   265272107      NaN

      M_INOUT_AMVAL_ID DOCUMENTNO1 LOCATIONS_ID  ACCT_AC_HOLDING_ID  \
0                  NaN        69.0          NaN                 1.0
1                  NaN        73.0          NaN                 1.0
2                  NaN       206.0          NaN                 1.0
3                  NaN      9405.0          NaN                 1.0
4                  NaN       346.0          NaN                 1.0
...                ...         ...          ...                 ...
7593               NaN         NaN    1000050.0                 1.0
7594               NaN         NaN    1000050.0                 1.0
7595               NaN         NaN    1000050.0                 1.0
7597               NaN         NaN    1000196.0                 1.0
7598               NaN         NaN    1000050.0                 1.0

       VAHED_MALI  ACCT_AC_JOURNAL_ID  BASEINFO_RECORDID    C_YEAR_ID
0     200000138.0         470900574.0                NaN          NaN
1     200000138.0         470900574.0                NaN          NaN
2     200000138.0         470900574.0                NaN          NaN
3     200000138.0         470900574.0                NaN          NaN
4     200000138.0         470900574.0                NaN          NaN
...           ...                 ...                ...          ...
7593  200000138.0                NaN        469638651.0  470737412.0
7594  200000138.0                NaN        469638651.0  470737412.0
7595  200000138.0                NaN        469638651.0  470737412.0
7597  469638568.0             2443.0        469638671.0  470737412.0
7598  200000138.0                NaN        469638651.0  470737412.0

[7583 rows x 43 columns]
       INOUT_ID  AD_CLIENT_ID  AD_ORG_ID ISACTIVE          CREATED  \
1466  469639217     104000002          0        Y   7/12/2016 15:09
2269  469640047     104000002          0        Y     3/7/2017 8:58
```

2270	469640048	104000002	0	Y	3/7/2017 9:01
2271	469640049	104000002	0	Y	3/7/2017 9:15
2990	469640796	104000002	0	Y	10/18/2017 8:19
3462	469641285	104000002	0	Y	2/25/2018 12:32
4361	469642220	104000002	0	Y	12/8/2018 8:17
5616	469643504	104000002	0	Y	6/8/2020 8:50
6663	469644634	104000002	104000002	Y	7/3/2021 14:01
6894	469645454	104000002	104000002	Y	11/27/2021 10:35
7522	469647353	104000002	0	Y	8/3/2022 8:34
7580	469647760	104000002	0	Y	9/4/2022 15:12
7596	469647837	104000002	0	Y	9/13/2022 10:18
7599	469647835	104000002	0	Y	9/13/2022 10:01

	CREATEDBY	UPDATED	UPDATEDBY	NAME
1466	210032662	9/3/2016 11:21	210032662	NaN
2269	210032662	3/7/2017 8:58	210032662	NaN
2270	210032662	3/7/2017 9:04	210032662	NaN
2271	210032662	3/7/2017 9:15	210032662	NaN
2990	210032662	10/18/2017 8:19	210032662	NaN
3462	210032662	2/25/2018 12:32	210032662	NaN
4361	210032662	12/8/2018 8:17	210032662	NaN
5616	469638254	6/8/2020 8:50	469638254	NaN
6663	469637406	8/7/2021 8:24	469637406	NaN
6894	469637406	11/27/2021 13:26	469637406	NaN
7522	210032977	8/3/2022 8:34	210032977	NaN
7580	210032977	9/4/2022 15:12	210032977	NaN
7596	210033167	9/13/2022 10:18	210033167	NaN
7599	210033167	9/13/2022 10:01	210033167	NaN

	DESCRIPTION	...	SUB_REF_ORG
1466	???? ??????? ????? ?? ?????? ??????? ????? ???...	...	NaN
2269	NaN	...	NaN
2270	NaN	...	NaN
2271	NaN	...	NaN
2990	NaN	...	NaN
3462	NaN	...	NaN
4361	NaN	...	NaN
5616	NaN	...	NaN
6663	NaN	...	265272107
6894	NaN	...	265272107
7522	NaN	...	265272107
7580	NaN	...	265272107
7596	NaN	...	265272107
7599	NaN	...	265272107

	STRING3	M_INOUT_AMVAL_ID	DOCUMENTNO1	LOCATIONS_ID	ACCT_AC_HOLDING_ID
1466	NaN	NaN	NaN	NaN	1.0
2269	NaN	NaN	NaN	NaN	1.0
2270	NaN	NaN	NaN	NaN	1.0
2271	NaN	NaN	NaN	NaN	1.0
2990	NaN	NaN	NaN	NaN	1.0
3462	NaN	NaN	NaN	NaN	1.0
4361	NaN	NaN	NaN	NaN	1.0
5616	NaN	NaN	NaN	NaN	1.0
6663	NaN	NaN	NaN	NaN	1.0
6894	24/52	NaN	NaN	1000194.0	1.0

```
7522      NaN              NaN         NaN          NaN                1.0
7580      NaN              NaN         NaN          NaN                3.0
7596      NaN              NaN         NaN          NaN                4.0
7599      NaN              NaN         NaN          NaN                4.0

       VAHED_MALI  ACCT_AC_JOURNAL_ID  BASEINFO_RECORDID    C_YEAR_ID
1466  200000138.0                 NaN                NaN          NaN
2269  200000138.0                 NaN                NaN          NaN
2270  200000138.0                 NaN                NaN          NaN
2271  200000138.0                 NaN                NaN          NaN
2990          NaN                 NaN                NaN          NaN
3462          NaN                 NaN                NaN          NaN
4361          NaN                 NaN                NaN          NaN
5616          NaN                 NaN                NaN          NaN
6663  200000138.0                 NaN        469638726.0  469637411.0
6894  200000141.0                 NaN        469638588.0  469637411.0
7522  200000138.0                 NaN        469638651.0          NaN
7580  469638358.0                 NaN        469638630.0          NaN
7596  200000529.0                 NaN        469638805.0          NaN
7599  200000529.0                 NaN        469638805.0          NaN

[14 rows x 43 columns]
     INOUT_ID  AD_CLIENT_ID  AD_ORG_ID ISACTIVE          CREATED  CREATEDBY  \
19  210032665     104000002          0        Y  12/26/2013 9:49  210035294

            UPDATED  UPDATEDBY  NAME DESCRIPTION  ... SUB_REF_ORG  STRING3  \
19  12/26/2013 9:50  210035294   NaN         NaN  ...         NaN      NaN

    M_INOUT_AMVAL_ID DOCUMENTNO1 LOCATIONS_ID  ACCT_AC_HOLDING_ID  VAHED_MALI  \
19               NaN         NaN          NaN                 1.0         NaN

    ACCT_AC_JOURNAL_ID  BASEINFO_RECORDID  C_YEAR_ID
19                 NaN                NaN        NaN

[1 rows x 43 columns]
```

As we finished this phase we saw how much our previous two phases helped us in understanding the core of our datasets and also to clean it and get the most knowledge on what are the ups and downs and quality of what we have our hands-on, only after those steps we had the courage and enough resource to analyze our datasets and know the answers to some real questions about assets.