**EDA-Elo Merchant Category Recommendation**

**上原　蔵人**

**2024-10-5**

1. 各ファイルに含まれる行数・列数、データ型、欠損値

以下は各ファイルのデータの行数・列数とデータの型

train.csv

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 201917 entries, 0 to 201916

Data columns (total 6 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 first\_active\_month 201917 non-null object

1 card\_id 201917 non-null object

2 feature\_1 201917 non-null int64

3 feature\_2 201917 non-null int64

4 feature\_3 201917 non-null int64

5 target 201917 non-null float64

dtypes: float64(1), int64(3), object(2)

memory usage: 9.2+ MB

test.csv

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 123623 entries, 0 to 123622

Data columns (total 5 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 first\_active\_month 123622 non-null object

1 card\_id 123623 non-null object

2 feature\_1 123623 non-null int64

3 feature\_2 123623 non-null int64

4 feature\_3 123623 non-null int64

dtypes: int64(3), object(2)

memory usage: 4.7+ MB

historical\_transactions.csv

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 29112361 entries, 0 to 29112360

Data columns (total 14 columns):

# Column Dtype

--- ------ -----

0 authorized\_flag object

1 card\_id object

2 city\_id int64

3 category\_1 object

4 installments int64

5 category\_3 object

6 merchant\_category\_id int64

7 merchant\_id object

8 month\_lag int64

9 purchase\_amount float64

10 purchase\_date object

11 category\_2 float64

12 state\_id int64

13 subsector\_id int64

dtypes: float64(2), int64(6), object(6)

memory usage: 3.0+ GB

new\_merchant\_transactions.csv

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 1963031 entries, 0 to 1963030

Data columns (total 14 columns):

# Column Dtype

--- ------ -----

0 authorized\_flag object

1 card\_id object

2 city\_id int64

3 category\_1 object

4 installments int64

5 category\_3 object

6 merchant\_category\_id int64

7 merchant\_id object

8 month\_lag int64

9 purchase\_amount float64

10 purchase\_date object

11 category\_2 float64

12 state\_id int64

13 subsector\_id int64

dtypes: float64(2), int64(6), object(6)

memory usage: 209.7+ MB

merchants.csv

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 334696 entries, 0 to 334695

Data columns (total 22 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 merchant\_id 334696 non-null object

1 merchant\_group\_id 334696 non-null int64

2 merchant\_category\_id 334696 non-null int64

3 subsector\_id 334696 non-null int64

4 numerical\_1 334696 non-null float64

5 numerical\_2 334696 non-null float64

6 category\_1 334696 non-null object

7 most\_recent\_sales\_range 334696 non-null object

8 most\_recent\_purchases\_range 334696 non-null object

9 avg\_sales\_lag3 334683 non-null float64

10 avg\_purchases\_lag3 334696 non-null float64

11 active\_months\_lag3 334696 non-null int64

12 avg\_sales\_lag6 334683 non-null float64

13 avg\_purchases\_lag6 334696 non-null float64

14 active\_months\_lag6 334696 non-null int64

15 avg\_sales\_lag12 334683 non-null float64

16 avg\_purchases\_lag12 334696 non-null float64

17 active\_months\_lag12 334696 non-null int64

18 category\_4 334696 non-null object

19 city\_id 334696 non-null int64

20 state\_id 334696 non-null int64

21 category\_2 322809 non-null float64

dtypes: float64(9), int64(8), object(5)

memory usage: 56.2+ MB

以下は各ファイルのカラムの欠損値

Train Null Values:

first\_active\_month 0

card\_id 0

feature\_1 0

feature\_2 0

feature\_3 0

target 0

dtype: int64

Test Null Values:

first\_active\_month 1

card\_id 0

feature\_1 0

feature\_2 0

feature\_3 0

dtype: int64

History Null Values:

authorized\_flag 0

card\_id 0

city\_id 0

category\_1 0

installments 0

category\_3 178159

merchant\_category\_id 0

merchant\_id 138481

month\_lag 0

purchase\_amount 0

purchase\_date 0

category\_2 2652864

state\_id 0

subsector\_id 0

dtype: int64

New Merchant Null Values:

authorized\_flag 0

card\_id 0

city\_id 0

category\_1 0

installments 0

category\_3 55922

merchant\_category\_id 0

merchant\_id 26216

month\_lag 0

purchase\_amount 0

purchase\_date 0

category\_2 111745

state\_id 0

subsector\_id 0

dtype: int64

Merchants Null Values:

merchant\_id 0

merchant\_group\_id 0

merchant\_category\_id 0

subsector\_id 0

numerical\_1 0

numerical\_2 0

category\_1 0

most\_recent\_sales\_range 0

most\_recent\_purchases\_range 0

avg\_sales\_lag3 13

avg\_purchases\_lag3 0

active\_months\_lag3 0

avg\_sales\_lag6 13

avg\_purchases\_lag6 0

active\_months\_lag6 0

avg\_sales\_lag12 13

avg\_purchases\_lag12 0

active\_months\_lag12 0

category\_4 0

city\_id 0

state\_id 0

category\_2 11887

dtype: int64

1. ターゲット変数の分布

以下はtrain.csv内にあるターゲット変数の分布。外れ値が負の方向にあるが、大体のデータは0を中心とした尖った正規分布の形になっている。

グラフ, ヒストグラム

自動的に生成された説明

1. カテゴリ変数とターゲットの関係

以下の3つ図はtrain.csvの中にある特微量とターゲットの関係を箱ヒゲ図にしたもの。

2章で見たように負の方向に外れ値が存在するが、0を中心にまとまっている。その傾向はどの特微量も同じようだ。

ダイアグラム

低い精度で自動的に生成された説明

グラフ, 箱ひげ図

自動的に生成された説明

グラフ, 箱ひげ図

自動的に生成された説明

1. 新しい特微量の生成と結合

以下のコードはカードIDを軸に取引データを集約している。取引回数、取引額の合計、平均、最小値・最大値などを特徴量として追加しています。

|  |
| --- |
| # カードIDごとに取引データを集約  history\_agg = history.groupby('card\_id').agg({  'purchase\_amount': ['sum', 'mean', 'min', 'max', 'count'],  'month\_lag': 'mean'  }).reset\_index()  new\_merchant\_agg = new\_merchant.groupby('card\_id').agg({  'purchase\_amount': ['sum', 'mean', 'min', 'max', 'count'],  'month\_lag': 'mean'  }).reset\_index() |

以下は上記で作った新しい特微量を’card\_id’をキーにしてtrainに結合したもの。

|  |
| --- |
| # データの結合  train\_merged = pd.merge(train, history\_agg, on='card\_id', how='left')  train\_merged = pd.merge(train\_merged, new\_merchant\_agg, on='card\_id', how='left', suffixes=('\_history', '\_new')) |

train\_mergedデータで'first\_active\_month'のみ時系列データだったため、扱いやすいように時間の情報を個別に取得してint型の新しい特微量を作った。

|  |
| --- |
| # 例: 'first\_active\_month'を日付型に変換し、各成分を抽出  train\_merged['first\_active\_month'] = pd.to\_datetime(train\_merged['first\_active\_month'])  train\_merged['year'] = train\_merged['first\_active\_month'].dt.year  train\_merged['month'] = train\_merged['first\_active\_month'].dt.month  train\_merged['day'] = train\_merged['first\_active\_month'].dt.day  train\_merged['dayofweek'] = train\_merged['first\_active\_month'].dt.dayofweek |

以下はtrain\_mergedの特微量の情報。

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 201917 entries, 0 to 201916

Data columns (total 22 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 first\_active\_month 201917 non-null datetime64[ns]

1 card\_id 201917 non-null object

2 feature\_1 201917 non-null int64

3 feature\_2 201917 non-null int64

4 feature\_3 201917 non-null int64

5 target 201917 non-null float64

6 purchase\_amount\_sum\_history 201917 non-null float64

7 purchase\_amount\_mean\_history 201917 non-null float64

8 purchase\_amount\_min\_history 201917 non-null float64

9 purchase\_amount\_max\_history 201917 non-null float64

10 purchase\_amount\_count\_history 201917 non-null int64

11 month\_lag\_mean\_history 201917 non-null float64

12 purchase\_amount\_sum\_new 179986 non-null float64

13 purchase\_amount\_mean\_new 179986 non-null float64

14 purchase\_amount\_min\_new 179986 non-null float64

15 purchase\_amount\_max\_new 179986 non-null float64

16 purchase\_amount\_count\_new 179986 non-null float64

17 month\_lag\_mean\_new 179986 non-null float64

18 year 201917 non-null int32

19 month 201917 non-null int32

20 day 201917 non-null int32

21 dayofweek 201917 non-null int32

dtypes: datetime64[ns](1), float64(12), int32(4), int64(4), object(1)

memory usage: 30.8+ MB

1. 相関関係の分析

この章では4章で作ったtrain\_mergedの’card\_id’を取り除いたnew\_train\_mergedの特微量の相関関係を可視化する。

以下は全ての特微量のヒートマップ。

グラフ

自動的に生成された説明

かなり見にくいため下記に図を2つ補足する。

以下は相関が高い上位10個の特微量のヒートマップ。

グラフ, ツリーマップ図

自動的に生成された説明

以下は相関が高い上位5個と低い下位5個を合わせたヒートマップ。

グラフ, ツリーマップ図

自動的に生成された説明

targetとの相関が一番高いのは’month\_lag\_mean\_month’だが、その数値は0.062でかなり小さい。