

assignment 3 - Data exploration

October 10, 2022

Table of Contents

1 Load Libraries and Datasets

1.1 Load Libraries

1.2 Load Datasets

1.3 Preparing Data

2 Descriptive Statistics

2.1 View Datasets

2.2 Shape of Datasets

2.3 Features Overview

2.3.1 types of features

2.3.2 Statistical Description and Boxplots

2.3.2.1 TransactionDT

2.3.2.2 ProductCD

2.3.2.3 TransactionAmt

2.3.2.4 card brand (card4)

2.3.2.5 Card type (card6)

3 Skewness

3.1 TransactionDT

3.2 TransactionAmt

3.3 Card1

3.4 Card2

3.5 Card3

3.6 Card5

4 Class Distribution

4.1 Card Issuer

- 4.2 Card Type
- 4.3 Product CD
- 4.4 Device Type
- 4.5 isFraud
- 4.6 Email Domain
- 5 Correlations between features
 - 5.1 Correlation Coffeicents
 - 5.2 Chi-square test
 - 5.2.1 ProductCD and Card4
 - 5.2.2 ProductCD and Card5
 - 5.2.3 ProductCD and Card6
 - 5.2.4 Card4 and Card5
 - 5.2.5 Card4 and Card6
 - 5.2.6 Card5 and Card6

1 Load Libraries and Datasets

1.1 Load Libraries

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

1.2 Load Datasets

```
[2]: train_transaction = pd.read_csv('input/train_transaction.csv')
train_identity = pd.read_csv('input/train_identity.csv')

test_transaction = pd.read_csv('input/test_transaction.csv')
test_identity = pd.read_csv('input/test_identity.csv')
```

Reduce memory since the datasets would take a lot

```
[3]: ## Function to reduce the DF size
def reduce_mem_usage(df, verbose=True):
    numerics = ['int16', 'int32', 'int64', 'float16', 'float32', 'float64']
    start_mem = df.memory_usage().sum() / 1024**2
    for col in df.columns:
        col_type = df[col].dtypes
```

```

        if col_type in numerics:
            c_min = df[col].min()
            c_max = df[col].max()
            if str(col_type)[:3] == 'int':
                if c_min > np.iinfo(np.int8).min and c_max < np.iinfo(np.int8).
→max:
                    df[col] = df[col].astype(np.int8)
                elif c_min > np.iinfo(np.int16).min and c_max < np.iinfo(np.
→int16).max:
                    df[col] = df[col].astype(np.int16)
                elif c_min > np.iinfo(np.int32).min and c_max < np.iinfo(np.
→int32).max:
                    df[col] = df[col].astype(np.int32)
                elif c_min > np.iinfo(np.int64).min and c_max < np.iinfo(np.
→int64).max:
                    df[col] = df[col].astype(np.int64)
            else:
                if c_min > np.finfo(np.float16).min and c_max < np.finfo(np.
→float16).max:
                    df[col] = df[col].astype(np.float16)
                elif c_min > np.finfo(np.float32).min and c_max < np.finfo(np.
→float32).max:
                    df[col] = df[col].astype(np.float32)
                else:
                    df[col] = df[col].astype(np.float64)
            end_mem = df.memory_usage().sum() / 1024**2
            if verbose: print('Mem. usage decreased to {:.2f} Mb ({:.1f}% reduction)'.
→format(end_mem, 100 * (start_mem - end_mem) / start_mem))
        return df

```

```

[4]: ## Reducing memory of train sets
train_transaction = reduce_mem_usage(train_transaction)
train_identity = reduce_mem_usage(train_identity)

```

Mem. usage decreased to 542.35 Mb (69.4% reduction)

Mem. usage decreased to 25.72 Mb (43.0% reduction)

```

[5]: ## Reducing memory of test sets
test_transaction = reduce_mem_usage(test_transaction)
test_identity = reduce_mem_usage(test_identity)

```

Mem. usage decreased to 472.59 Mb (68.9% reduction)

Mem. usage decreased to 25.44 Mb (42.7% reduction)

1.3 Preparing Data

merge *train_transaction* and *train_identity*, *test_transaction* and *test_identity*

```
[6]: train = pd.merge(train_transaction, train_identity, on='TransactionID',
    →how='left')
test = pd.merge(test_transaction, test_identity, on='TransactionID', how='left')
```

data descripton of *train* and *test*

```
[7]: train.info()
test.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 590540 entries, 0 to 590539
Columns: 434 entries, TransactionID to DeviceInfo
dtypes: float16(353), float32(45), float64(1), int16(1), int32(2), int8(1),
object(31)
memory usage: 653.9+ MB
<class 'pandas.core.frame.DataFrame'>
Int64Index: 506691 entries, 0 to 506690
Columns: 433 entries, TransactionID to DeviceInfo
dtypes: float16(346), float32(53), int16(1), int32(2), object(31)
memory usage: 565.4+ MB
```

output the new csv files

```
[8]: train.to_csv('data/train.csv')
test.to_csv('data/test.csv')

[9]: del train_transaction, train_identity, test_transaction, test_identity
gc.collect()
```

[9]: 0

2 Descriptive Statistics

2.1 View Datasets

- load datasets

```
[2]: # If you have run 1.1 and 1.2, you don't need to run this cell
train = pd.read_csv('data/train.csv')
test = pd.read_csv('data/test.csv')
```

- train

```
[9]: pd.set_option('display.max_columns', None)
pd.DataFrame(train)
```

```
[9]:      Unnamed: 0  TransactionID  isFraud  TransactionDT  TransactionAmt  \
0              0      2987000         0         86400         68.50
1              1      2987001         0         86401         29.00
```

2	2	2987002	0	86469	59.00
3	3	2987003	0	86499	50.00
4	4	2987004	0	86506	50.00
...
590535	590535	3577535	0	15811047	49.00
590536	590536	3577536	0	15811049	39.50
590537	590537	3577537	0	15811079	30.95
590538	590538	3577538	0	15811088	117.00
590539	590539	3577539	0	15811131	280.00

	ProductCD	card1	card2	card3	card4	card5	card6	addr1	\
0	W	13926	NaN	150.0	discover	142.0	credit	315.0	
1	W	2755	404.0	150.0	mastercard	102.0	credit	325.0	
2	W	4663	490.0	150.0	visa	166.0	debit	330.0	
3	W	18132	567.0	150.0	mastercard	117.0	debit	476.0	
4	H	4497	514.0	150.0	mastercard	102.0	credit	420.0	
...	
590535	W	6550	NaN	150.0	visa	226.0	debit	272.0	
590536	W	10444	225.0	150.0	mastercard	224.0	debit	204.0	
590537	W	12037	595.0	150.0	mastercard	224.0	debit	231.0	
590538	W	7826	481.0	150.0	mastercard	224.0	debit	387.0	
590539	W	15066	170.0	150.0	mastercard	102.0	credit	299.0	

	addr2	dist1	dist2	P_emaildomain	R_emaildomain	C1	C2	C3	C4	\
0	87.0	19.0	NaN	NaN	NaN	1.0	1.0	0.0	0.0	
1	87.0	NaN	NaN	gmail.com	NaN	1.0	1.0	0.0	0.0	
2	87.0	287.0	NaN	outlook.com	NaN	1.0	1.0	0.0	0.0	
3	87.0	NaN	NaN	yahoo.com	NaN	2.0	5.0	0.0	0.0	
4	87.0	NaN	NaN	gmail.com	NaN	1.0	1.0	0.0	0.0	
...	
590535	87.0	48.0	NaN	NaN	NaN	2.0	1.0	0.0	0.0	
590536	87.0	NaN	NaN	gmail.com	NaN	1.0	1.0	0.0	0.0	
590537	87.0	NaN	NaN	gmail.com	NaN	1.0	1.0	0.0	0.0	
590538	87.0	3.0	NaN	aol.com	NaN	1.0	1.0	0.0	0.0	
590539	87.0	NaN	NaN	gmail.com	NaN	2.0	1.0	0.0	0.0	

	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	D1	D2	D3	\
0	0.0	1.0	0.0	0.0	1.0	0.0	2.0	0.0	1.0	1.0	14.0	NaN	13.0	
1	0.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	1.0	0.0	NaN	NaN	
2	0.0	1.0	0.0	0.0	1.0	0.0	1.0	0.0	1.0	1.0	0.0	NaN	NaN	
3	0.0	4.0	0.0	0.0	1.0	0.0	1.0	0.0	25.0	1.0	112.0	112.0	0.0	
4	0.0	1.0	0.0	1.0	0.0	1.0	1.0	0.0	1.0	1.0	0.0	NaN	NaN	
...	
590535	1.0	0.0	0.0	0.0	2.0	0.0	1.0	0.0	3.0	2.0	29.0	29.0	30.0	
590536	0.0	1.0	0.0	0.0	1.0	0.0	1.0	0.0	1.0	1.0	0.0	NaN	NaN	
590537	1.0	1.0	0.0	0.0	1.0	0.0	1.0	0.0	1.0	1.0	0.0	NaN	NaN	
590538	0.0	3.0	0.0	0.0	2.0	0.0	1.0	1.0	5.0	1.0	22.0	22.0	0.0	

590539	1.0	1.0	0.0	0.0	2.0	0.0	1.0	0.0	1.0	1.0	0.0	NaN	0.0
--------	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	M1	\
0	NaN	NaN	NaN	NaN	NaN	NaN	13.0	13.0	NaN	NaN	NaN	0.0	T	
1	0.0	NaN	NaN	NaN	NaN	NaN	0.0	NaN	NaN	NaN	NaN	0.0	NaN	
2	0.0	NaN	NaN	NaN	NaN	NaN	0.0	315.0	NaN	NaN	NaN	315.0	T	
3	94.0	0.0	NaN	NaN	NaN	NaN	84.0	NaN	NaN	NaN	NaN	111.0	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	56.0	56.0	NaN	NaN	NaN	56.0	T	
590536	0.0	NaN	NaN	NaN	NaN	NaN	0.0	0.0	NaN	NaN	NaN	0.0	T	
590537	0.0	NaN	NaN	NaN	NaN	NaN	0.0	0.0	NaN	NaN	NaN	0.0	T	
590538	22.0	0.0	NaN	NaN	NaN	NaN	22.0	22.0	NaN	NaN	NaN	22.0	T	
590539	1.0	0.0	NaN	NaN	NaN	NaN	1.0	0.0	NaN	NaN	NaN	1.0	T	

	M2	M3	M4	M5	M6	M7	M8	M9	V1	V2	V3	V4	V5	V6	\
0	T	T	M2	F	T	NaN	NaN	NaN	1.0	1.0	1.0	1.0	1.0	1.0	
1	NaN	NaN	M0	T	T	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	T	T	M0	F	F	F	F	F	1.0	1.0	1.0	1.0	1.0	1.0	
3	NaN	NaN	M0	T	F	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
590535	T	T	M0	T	F	F	F	T	1.0	1.0	1.0	0.0	0.0	1.0	
590536	F	F	M0	F	T	F	F	F	1.0	1.0	1.0	1.0	1.0	1.0	
590537	F	F	NaN	NaN	T	NaN	NaN	NaN	1.0	1.0	1.0	1.0	1.0	1.0	
590538	T	T	M0	F	T	NaN	NaN	NaN	1.0	1.0	1.0	2.0	2.0	1.0	
590539	F	F	NaN	NaN	T	F	F	F	1.0	1.0	1.0	1.0	1.0	1.0	

	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20	\
0	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
1	NaN	NaN	NaN	NaN	NaN	0.0	0.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
2	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
3	NaN	NaN	NaN	NaN	NaN	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
590535	1.0	1.0	1.0	0.0	0.0	2.0	2.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
590536	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
590537	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
590538	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	2.0	2.0	
590539	1.0	1.0	1.0	1.0	1.0	2.0	2.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	

	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30	V31	V32	V33	V34	\
0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
1	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
2	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
3	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

...	
590535	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	
590536	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
590537	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	
590538	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	
590539	0.0	0.0	2.0	2.0	1.0	1.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	
	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44	V45	V46	V47	V48	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	0.0	0.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	
2	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	
3	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	
590537	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	
590538	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	
590539	2.0	2.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	2.0	2.0	1.0	1.0	1.0	
	V49	V50	V51	V52	V53	V54	V55	V56	V57	V58	V59	V60	V61	V62	\
0	NaN	NaN	NaN	NaN	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
1	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
2	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
3	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
590535	NaN	NaN	NaN	NaN	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
590536	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
590537	1.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
590538	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	2.0	2.0	
590539	1.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
	V63	V64	V65	V66	V67	V68	V69	V70	V71	V72	V73	V74	V75	V76	\
0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	
1	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
2	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	
3	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
590535	0.0	0.0	1.0	1.0	1.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	2.0	2.0	
590536	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	
590537	0.0	0.0	1.0	1.0	1.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
590538	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	
590539	0.0	0.0	1.0	1.0	1.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	2.0	2.0	
	V77	V78	V79	V80	V81	V82	V83	V84	V85	V86	V87	V88	V89	V90	\

0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0
1	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0
2	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0
3	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
...
590535	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	1.0
590536	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0
590537	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	1.0
590538	1.0	1.0	0.0	0.0	0.0	2.0	2.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0
590539	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	2.0	2.0	1.0	0.0	1.0

	V91	V92	V93	V94	V95	V96	V97	V98	V99	V100	V101	V102	\
0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	1.0	
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
3	0.0	0.0	0.0	0.0	1.0	48.0	28.0	0.0	10.0	4.0	1.0	38.0	
4	NaN	NaN	NaN	NaN	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
...	
590535	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	1.0	0.0	0.0	0.0	
590536	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
590537	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
590538	0.0	0.0	0.0	0.0	1.0	4.0	1.0	1.0	1.0	1.0	0.0	3.0	
590539	1.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	

	V103	V104	V105	V106	V107	V108	V109	V110	V111	V112	V113	\
0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
1	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
2	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
3	24.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
4	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
...	
590535	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
590536	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
590537	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
590538	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
590539	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	

	V114	V115	V116	V117	V118	V119	V120	V121	V122	V123	V124	\
0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
1	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
2	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
3	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
4	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
...	
590535	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
590536	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	

590537	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
590538	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
590539	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

	V125	V126	V127	V128	V129	V130	V131	V132	V133	\
0	1.0	0.00	117.00	0.00	0.0	0.00	0.0	0.0	117.0	
1	1.0	0.00	0.00	0.00	0.0	0.00	0.0	0.0	0.0	
2	1.0	0.00	0.00	0.00	0.0	0.00	0.0	0.0	0.0	
3	1.0	50.00	1758.00	925.00	0.0	354.00	135.0	50.0	1404.0	
4	1.0	0.00	0.00	0.00	0.0	0.00	0.0	0.0	0.0	
...	
590535	1.0	0.00	47.95	0.00	0.0	47.94	0.0	0.0	0.0	
590536	1.0	0.00	0.00	0.00	0.0	0.00	0.0	0.0	0.0	
590537	1.0	0.00	0.00	0.00	0.0	0.00	0.0	0.0	0.0	
590538	1.0	117.00	1035.50	117.00	117.0	117.00	117.0	0.0	918.5	
590539	1.0	279.95	279.95	279.95	0.0	0.00	0.0	0.0	0.0	

	V134	V135	V136	V137	V138	V139	V140	V141	V142	V143	\
0	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	NaN	
1	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	NaN	
2	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	NaN	
3	790.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	NaN	
4	0.0	0.00	0.00	0.00	0.0	0.0	0.0	0.0	0.0	6.0	
...	
590535	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	NaN	
590536	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	NaN	
590537	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	NaN	
590538	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	NaN	
590539	0.0	279.95	279.95	279.95	NaN	NaN	NaN	NaN	NaN	NaN	

	V144	V145	V146	V147	V148	V149	V150	V151	V152	V153	V154	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	18.0	140.0	0.0	0.0	0.0	0.0	1803.0	49.0	64.0	0.0	0.0	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	V155	V156	V157	V158	V159	V160	V161	V162	V163	V164	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	0.0	0.0	0.0	0.0	15560.0	169690.8	0.0	0.0	0.0	515.0
...
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

	V165	V166	V167	V168	V169	V170	V171	V172	V173	V174	V175	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	5155.0	2840.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	V176	V177	V178	V179	V180	V181	V182	V183	V184	V185	V186	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	V187	V188	V189	V190	V191	V192	V193	V194	V195	V196	V197	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	V198	V199	V200	V201	V202	V203	V204	V205	V206	V207	V208	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	V209	V210	V211	V212	V213	V214	V215	V216	V217	V218	V219	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	V220	V221	V222	V223	V224	V225	V226	V227	V228	V229	V230	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	V231	V232	V233	V234	V235	V236	V237	V238	V239	V240	V241	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	
...	

590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

	V242	V243	V244	V245	V246	V247	V248	V249	V250	V251	V252	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	V253	V254	V255	V256	V257	V258	V259	V260	V261	V262	V263	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	V264	V265	V266	V267	V268	V269	V270	V271	V272	V273	V274	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	V275	V276	V277	V278	V279	V280	V281	V282	V283	V284	V285	\
0	NaN	NaN	NaN	NaN	0.0	0.0	0.0	1.0	1.0	0.0	0.0	

1	NaN	NaN	NaN	NaN	0.0	0.0	0.0	1.0	1.0	0.0	0.0
2	NaN	NaN	NaN	NaN	0.0	0.0	0.0	1.0	1.0	0.0	0.0
3	NaN	NaN	NaN	NaN	1.0	28.0	0.0	0.0	0.0	0.0	10.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0
...
590535	NaN	NaN	NaN	NaN	0.0	0.0	0.0	1.0	1.0	0.0	1.0
590536	NaN	NaN	NaN	NaN	0.0	0.0	0.0	1.0	1.0	0.0	0.0
590537	NaN	NaN	NaN	NaN	0.0	0.0	0.0	1.0	1.0	0.0	0.0
590538	NaN	NaN	NaN	NaN	1.0	1.0	0.0	2.0	7.0	1.0	5.0
590539	NaN	NaN	NaN	NaN	1.0	1.0	0.0	2.0	2.0	0.0	0.0

	V286	V287	V288	V289	V290	V291	V292	V293	V294	V295	V296	\
0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	1.0	0.0	0.0	
1	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	
2	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	
3	0.0	4.0	0.0	0.0	1.0	1.0	1.0	1.0	38.0	24.0	0.0	
4	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	
...	
590535	0.0	0.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	
590536	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	
590537	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	
590538	0.0	1.0	1.0	1.0	1.0	2.0	1.0	0.0	11.0	0.0	0.0	
590539	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	1.0	

	V297	V298	V299	V300	V301	V302	V303	V304	V305	V306	V307	\
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.00	117.00	
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.00	0.00	
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.00	0.00	
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	50.00	1758.00	
4	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.00	0.00	
...	
590535	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.00	47.95	
590536	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.00	0.00	
590537	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.00	0.00	
590538	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	117.00	2903.50	
590539	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	1.0	279.95	279.95	

	V308	V309	V310	V311	V312	V313	V314	V315	V316	\
0	0.00	0.0	0.00	0.0	0.0	0.00	0.00	0.00	0.0	
1	0.00	0.0	0.00	0.0	0.0	0.00	0.00	0.00	0.0	
2	0.00	0.0	0.00	0.0	0.0	0.00	0.00	0.00	0.0	
3	925.00	0.0	354.00	0.0	135.0	0.00	0.00	0.00	50.0	
4	0.00	0.0	0.00	0.0	0.0	0.00	0.00	0.00	0.0	
...	
590535	0.00	0.0	47.94	0.0	0.0	47.94	47.94	47.94	0.0	
590536	0.00	0.0	0.00	0.0	0.0	0.00	0.00	0.00	0.0	
590537	0.00	0.0	0.00	0.0	0.0	0.00	0.00	0.00	0.0	

590538	117.00	117.0	669.50	0.0	117.0	317.50	669.50	317.50	0.0	
590539	279.95	0.0	0.00	0.0	0.0	0.00	0.00	0.00	0.0	

	V317	V318	V319	V320	V321	V322	V323	V324	V325	V326	\
0	117.0	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	
1	0.0	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	
2	0.0	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	
3	1404.0	790.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	
4	0.0	0.0	0.00	0.00	0.00	0.0	0.0	0.0	0.0	0.0	
...	
590535	0.0	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	
590536	0.0	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	
590537	0.0	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	
590538	2234.0	0.0	0.00	0.00	0.00	NaN	NaN	NaN	NaN	NaN	
590539	0.0	0.0	279.95	279.95	279.95	NaN	NaN	NaN	NaN	NaN	

	V327	V328	V329	V330	V331	V332	V333	V334	V335	V336	V337	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	V338	V339	id_01	id_02	id_03	id_04	id_05	id_06	id_07	id_08	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	0.0	0.0	0.0	70787.0	NaN	NaN	NaN	NaN	NaN	NaN	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	id_09	id_10	id_11	id_12	id_13	id_14	id_15	id_16	id_17	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

4	NaN	NaN	100.0	NotFound	NaN	-480.0	New	NotFound	166.0
...
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

	id_18	id_19	id_20	id_21	id_22	id_23	id_24	id_25	id_26	id_27	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	542.0	144.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	id_28	id_29	id_30	id_31	id_32	id_33	\
0	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	
4	New	NotFound	Android 7.0	samsung browser	6.2	32.0	2220x1080
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	

	id_34	id_35	id_36	id_37	id_38	DeviceType	\
0	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	
4	match_status:2	T	F	T	T	mobile	
...	
590535	NaN	NaN	NaN	NaN	NaN	NaN	
590536	NaN	NaN	NaN	NaN	NaN	NaN	
590537	NaN	NaN	NaN	NaN	NaN	NaN	
590538	NaN	NaN	NaN	NaN	NaN	NaN	
590539	NaN	NaN	NaN	NaN	NaN	NaN	

```

                                DeviceInfo
0                                NaN
1                                NaN
2                                NaN
3                                NaN
4    SAMSUNG SM-G892A Build/NRD90M
...
590535                           NaN
590536                           NaN
590537                           NaN
590538                           NaN
590539                           NaN

```

[590540 rows x 435 columns]

- test

```
[10]: pd.DataFrame(test)
```

```

[10]:      Unnamed: 0  TransactionID  TransactionDT  TransactionAmt  ProductCD  \
0              0      3663549      18403224          31.95          W
1              1      3663550      18403263          49.00          W
2              2      3663551      18403310         171.00          W
3              3      3663552      18403310         285.00          W
4              4      3663553      18403317          67.94          W
...          ...          ...          ...          ...          ...
506686      506686      4170235      34214279          94.70          C
506687      506687      4170236      34214287          12.17          C
506688      506688      4170237      34214326          49.00          W
506689      506689      4170238      34214337         202.00          W
506690      506690      4170239      34214345          24.34          C

      card1  card2  card3      card4  card5  card6  addr1  addr2  dist1  \
0      10409  111.0  150.0      visa  226.0  debit  170.0  87.0    1.0
1       4272  111.0  150.0      visa  226.0  debit  299.0  87.0    4.0
2       4476  574.0  150.0      visa  226.0  debit  472.0  87.0  2636.0
3      10989  360.0  150.0      visa  166.0  debit  205.0  87.0   17.0
4      18018  452.0  150.0  mastercard  117.0  debit  264.0  87.0    6.0
...      ...      ...      ...      ...      ...      ...      ...      ...
506686  13832  375.0  185.0  mastercard  224.0  debit  284.0  60.0   NaN
506687   3154  408.0  185.0  mastercard  224.0  debit    NaN    NaN   NaN
506688  16661  490.0  150.0      visa  226.0  debit  327.0  87.0   NaN
506689  16621  516.0  150.0  mastercard  224.0  debit  177.0  87.0   NaN
506690   5713  168.0  144.0      visa  147.0  credit    NaN    NaN   NaN

      dist2  P_emaildomain  R_emaildomain  C1  C2  C3  C4  C5  C6  C7  \
0      NaN      gmail.com      NaN  6.0  6.0  0.0  0.0  3.0  4.0  0.0
1      NaN      aol.com      NaN  3.0  2.0  0.0  0.0  0.0  1.0  0.0

```


2	NaN	hotmail.com		NaN	2.0	2.0	0.0	0.0	0.0	5.0	0.0
3	NaN	gmail.com		NaN	5.0	2.0	0.0	0.0	1.0	1.0	0.0
4	NaN	gmail.com		NaN	6.0	6.0	0.0	0.0	2.0	5.0	0.0
...
506686	NaN	gmail.com	gmail.com		1.0	1.0	0.0	1.0	0.0	1.0	1.0
506687	157.0	hotmail.com	hotmail.com		1.0	3.0	0.0	1.0	0.0	1.0	1.0
506688	NaN	hotmail.com		NaN	1.0	1.0	0.0	0.0	0.0	1.0	0.0
506689	NaN	hotmail.com		NaN	1.0	1.0	0.0	0.0	0.0	1.0	0.0
506690	NaN	hotmail.com	hotmail.com		1.0	1.0	0.0	1.0	0.0	1.0	1.0

	C8	C9	C10	C11	C12	C13	C14	D1	D2	D3	D4	D5	\
0	0.0	6.0	0.0	5.0	1.0	115.0	6.0	419.0	419.0	27.0	398.0	27.0	
1	0.0	2.0	0.0	1.0	1.0	12.0	2.0	149.0	149.0	7.0	634.0	7.0	
2	0.0	4.0	0.0	2.0	0.0	22.0	2.0	137.0	137.0	10.0	97.0	10.0	
3	0.0	2.0	0.0	2.0	0.0	7.0	4.0	42.0	42.0	41.0	242.0	41.0	
4	0.0	5.0	0.0	6.0	0.0	14.0	6.0	22.0	22.0	0.0	22.0	0.0	
...	
506686	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	NaN	NaN	0.0	NaN	
506687	2.0	0.0	2.0	1.0	1.0	3.0	1.0	16.0	16.0	16.0	16.0	16.0	
506688	0.0	1.0	0.0	1.0	0.0	1.0	1.0	0.0	NaN	NaN	0.0	NaN	
506689	0.0	1.0	0.0	1.0	0.0	1.0	1.0	0.0	NaN	NaN	0.0	NaN	
506690	1.0	0.0	1.0	1.0	1.0	1.0	1.0	0.0	NaN	NaN	0.0	NaN	

	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15	M1	\
0	NaN	NaN	NaN	NaN	418.0	203.0	NaN	NaN	NaN	409.0	T	
1	NaN	NaN	NaN	NaN	231.0	634.0	NaN	NaN	NaN	634.0	T	
2	NaN	NaN	NaN	NaN	136.0	136.0	NaN	NaN	NaN	97.0	T	
3	NaN	NaN	NaN	NaN	242.0	242.0	NaN	NaN	NaN	242.0	T	
4	NaN	NaN	NaN	NaN	22.0	22.0	NaN	NaN	NaN	22.0	T	
...	
506686	0.0	NaN	NaN	NaN	0.0	NaN	0.0	NaN	NaN	0.0	NaN	
506687	16.0	16.0	NaN	NaN	0.0	NaN	16.0	0.0	0.0	16.0	NaN	
506688	NaN	NaN	NaN	NaN	0.0	0.0	NaN	NaN	NaN	0.0	T	
506689	NaN	NaN	NaN	NaN	0.0	0.0	NaN	NaN	NaN	0.0	T	
506690	0.0	NaN	29.95	0.9585	0.0	NaN	0.0	0.0	0.0	0.0	NaN	

	M2	M3	M4	M5	M6	M7	M8	M9	V1	V2	V3	V4	V5	V6	\
0	T	F	NaN	NaN	F	T	T	T	1.0	1.0	1.0	1.0	1.0	1.0	
1	F	F	M0	NaN	F	NaN	NaN	NaN	1.0	1.0	1.0	1.0	1.0	1.0	
2	T	F	M0	F	F	F	F	F	1.0	1.0	1.0	1.0	1.0	1.0	
3	T	T	NaN	NaN	T	NaN	NaN	NaN	1.0	1.0	1.0	1.0	1.0	1.0	
4	T	T	NaN	NaN	F	F	T	T	1.0	1.0	1.0	1.0	1.0	1.0	
...	
506686	NaN	NaN	M2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	NaN	NaN	M2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506688	T	T	M0	F	F	F	T	T	1.0	1.0	1.0	1.0	1.0	1.0	
506689	T	T	M0	F	F	F	F	F	1.0	1.0	1.0	1.0	1.0	1.0	

506690	NaN	NaN	M2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20	\
0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
1	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
2	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
3	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
4	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
...	
506686	NaN	NaN	NaN	NaN	NaN	0.0	0.0	1.0	1.0	1.0	0.0	0.0	1.0	1.0	
506687	NaN	NaN	NaN	NaN	NaN	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
506688	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
506689	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
506690	NaN	NaN	NaN	NaN	NaN	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30	V31	V32	V33	V34	\
0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
1	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
2	0.0	0.0	2.0	2.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
3	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	
4	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	
...	
506686	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	
506687	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	
506688	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
506689	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
506690	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	
	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44	V45	V46	V47	V48	\
0	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	
1	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	
2	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	
3	2.0	2.0	2.0	2.0	0.0	0.0	1.0	0.0	0.0	2.0	2.0	2.0	2.0	1.0	
4	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	
...	
506686	0.0	0.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	
506687	0.0	0.0	2.0	2.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	
506688	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	
506689	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	
506690	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	
	V49	V50	V51	V52	V53	V54	V55	V56	V57	V58	V59	V60	V61	V62	\
0	1.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
1	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
2	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
3	1.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
4	1.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	

...
506686	0.0	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	1.0	1.0
506687	0.0	1.0	1.0	1.0	0.0	0.0	2.0	2.0	1.0	1.0	1.0	1.0	1.0	1.0
506688	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0
506689	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0
506690	0.0	1.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

	V63	V64	V65	V66	V67	V68	V69	V70	V71	V72	V73	V74	V75	V76	\
0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
1	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	
2	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	2.0	
3	0.0	0.0	1.0	1.0	1.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
4	0.0	0.0	1.0	1.0	1.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
...	
506686	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	
506687	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	
506688	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	
506689	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	
506690	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	0.0	0.0	

	V77	V78	V79	V80	V81	V82	V83	V84	V85	V86	V87	V88	V89	V90	\
0	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	1.0	
1	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	
2	1.0	2.0	0.0	0.0	0.0	1.0	2.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	
3	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	1.0	
4	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	1.0	
...	
506686	1.0	1.0	1.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	
506687	2.0	2.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	
506688	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	
506689	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	
506690	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	

	V91	V92	V93	V94	V95	V96	V97	V98	V99	V100	V101	V102	V103	\
0	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	
1	0.0	0.0	0.0	0.0	0.0	4.0	1.0	0.0	4.0	1.0	0.0	0.0	0.0	
2	0.0	0.0	0.0	0.0	0.0	5.0	0.0	0.0	3.0	0.0	0.0	0.0	0.0	
3	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
4	1.0	0.0	0.0	0.0	1.0	3.0	1.0	1.0	3.0	1.0	0.0	0.0	0.0	
...	
506686	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
506687	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
506688	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	
506689	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	
506690	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

	V104	V105	V106	V107	V108	V109	V110	V111	V112	V113	V114	\
--	------	------	------	------	------	------	------	------	------	------	------	---

0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
1	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
2	0.0	2.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
3	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
4	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
...
506686	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
506687	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
506688	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
506689	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
506690	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

	V115	V116	V117	V118	V119	V120	V121	V122	V123	V124	V125	\
0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
1	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
2	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
3	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
4	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
...	
506686	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
506687	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
506688	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
506689	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
506690	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	

	V126	V127	V128	V129	V130	V131	V132	V133	V134	V135	\
0	0.00	47.95	0.00	0.00	47.95	0.00	0.0	0.0	0.0	0.0	
1	0.00	280.00	77.00	0.00	280.00	77.00	0.0	0.0	0.0	0.0	
2	0.00	968.00	0.00	0.00	705.00	0.00	0.0	0.0	0.0	0.0	
3	0.00	0.00	0.00	0.00	0.00	0.00	0.0	0.0	0.0	0.0	
4	67.95	183.85	67.95	67.94	183.85	67.95	0.0	0.0	0.0	0.0	
...	
506686	0.00	0.00	0.00	0.00	0.00	0.00	0.0	0.0	0.0	0.0	
506687	0.00	0.00	0.00	0.00	0.00	0.00	0.0	0.0	0.0	0.0	
506688	0.00	0.00	0.00	0.00	0.00	0.00	0.0	0.0	0.0	0.0	
506689	0.00	0.00	0.00	0.00	0.00	0.00	0.0	0.0	0.0	0.0	
506690	0.00	0.00	0.00	0.00	0.00	0.00	0.0	0.0	0.0	0.0	

	V136	V137	V138	V139	V140	V141	V142	V143	V144	V145	V146	\
0	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	263.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

506688	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
506689	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
506690	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

	V147	V148	V149	V150	V151	V152	V153	V154	V155	V156	V157	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	V158	V159	V160	V161	V162	V163	V164	V165	V166	V167	V168	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.0	0.0	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.0	0.0	

	V169	V170	V171	V172	V173	V174	V175	V176	V177	V178	V179	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	

	V180	V181	V182	V183	V184	V185	V186	V187	V188	V189	V190	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
...
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
506687	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
506690	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0

	V191	V192	V193	V194	V195	V196	V197	V198	V199	V200	V201	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	

	V202	V203	V204	V205	V206	V207	V208	V209	V210	V211	V212	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

	V213	V214	V215	V216	V217	V218	V219	V220	V221	V222	V223	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	

	V224	V225	V226	V227	V228	V229	V230	V231	V232	V233	V234	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	0.0	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	

	V235	V236	V237	V238	V239	V240	V241	V242	V243	V244	V245	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	0.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0	1.0	1.0	

	V246	V247	V248	V249	V250	V251	V252	V253	V254	V255	V256	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	

	V257	V258	V259	V260	V261	V262	V263	V264	V265	V266	V267	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	

506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
506687	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
506690	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0

	V268	V269	V270	V271	V272	V273	V274	V275	V276	V277	V278	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

	V279	V280	V281	V282	V283	V284	V285	V286	V287	V288	V289	\
0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	
1	0.0	1.0	0.0	0.0	0.0	0.0	4.0	0.0	1.0	0.0	0.0	
2	0.0	0.0	0.0	0.0	0.0	0.0	4.0	0.0	0.0	0.0	0.0	
3	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	
4	1.0	1.0	0.0	1.0	1.0	1.0	3.0	1.0	1.0	1.0	1.0	
...	
506686	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
506687	0.0	0.0	1.0	2.0	2.0	0.0	2.0	0.0	0.0	2.0	2.0	
506688	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
506689	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
506690	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	

	V290	V291	V292	V293	V294	V295	V296	V297	V298	V299	V300	\
0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
1	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
2	1.0	2.0	1.0	0.0	0.0	0.0	2.0	0.0	2.0	0.0	0.0	
3	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
4	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
...	
506686	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
506687	1.0	3.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
506688	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
506689	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
506690	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

	V301	V302	V303	V304	V305	V306	V307	V308	V309	\
0	0.0	0.0	0.0	0.0	1.0	0.00	47.9500	0.00	0.00	

1	0.0	0.0	0.0	0.0	1.0	0.00	280.0000	77.00	0.00
2	0.0	0.0	0.0	0.0	1.0	0.00	1321.0000	0.00	0.00
3	0.0	0.0	0.0	0.0	1.0	0.00	0.0000	0.00	0.00
4	0.0	0.0	0.0	0.0	1.0	67.95	183.8500	67.95	67.94
...
506686	0.0	0.0	0.0	0.0	1.0	0.00	0.0000	0.00	0.00
506687	0.0	1.0	2.0	1.0	1.0	0.00	31.7237	0.00	0.00
506688	0.0	0.0	0.0	0.0	1.0	0.00	0.0000	0.00	0.00
506689	0.0	0.0	0.0	0.0	1.0	0.00	0.0000	0.00	0.00
506690	0.0	1.0	1.0	1.0	1.0	0.00	0.0000	0.00	0.00

	V310	V311	V312	V313	V314	V315	V316	V317	V318	\
0	47.9500	0.00	0.00	0.00	0.00	0.00	0.0	0.0	0.0	
1	280.0000	0.00	77.00	0.00	0.00	0.00	0.0	0.0	0.0	
2	1058.0000	0.00	0.00	0.00	0.00	0.00	0.0	0.0	0.0	
3	0.0000	0.00	0.00	282.50	282.50	282.50	0.0	0.0	0.0	
4	183.8500	67.94	67.95	67.94	183.90	67.94	0.0	0.0	0.0	
...	
506686	0.0000	0.00	0.00	0.00	0.00	0.00	0.0	0.0	0.0	
506687	31.7237	0.00	0.00	31.72	31.72	31.72	0.0	0.0	0.0	
506688	0.0000	0.00	0.00	0.00	0.00	0.00	0.0	0.0	0.0	
506689	0.0000	0.00	0.00	0.00	0.00	0.00	0.0	0.0	0.0	
506690	0.0000	0.00	0.00	0.00	0.00	0.00	0.0	0.0	0.0	

	V319	V320	V321	V322	V323	V324	V325	V326	V327	V328	V329	\
0	0.0	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	0.0	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	0.0	263.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	0.0	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	0.0	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	0.0	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	0.0	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506688	0.0	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	0.0	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	0.0	0.0	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

	V330	V331	V332	V333	V334	V335	V336	V337	V338	V339	id-01	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	-45.0	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
506690	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	-10.0

	id-02	id-03	id-04	id-05	id-06	id-07	id-08	id-09	id-10	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	266704.0	NaN	NaN	-3.0	-10.0	NaN	NaN	NaN	NaN	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	692090.0	0.0	0.0	0.0	0.0	NaN	NaN	0.0	0.0	

	id-11	id-12	id-13	id-14	id-15	id-16	id-17	id-18	id-19	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	100.0	NotFound	27.0	NaN	New	NotFound	225.0	15.0	176.0	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	100.0	NotFound	27.0	NaN	Found	Found	225.0	NaN	266.0	

	id-20	id-21	id-22	id-23	id-24	id-25	id-26	id-27	id-28	id-29	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
...	
506686	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506687	507.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	New	NotFound	
506688	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506689	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
506690	127.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Found	Found	

	id-30	id-31	id-32	id-33	id-34	id-35	id-36	id-37	\
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

4	NaN		NaN	NaN	NaN	NaN	NaN	NaN	NaN
...
506686	NaN		NaN	NaN	NaN	NaN	NaN	NaN	NaN
506687	NaN	chrome 43.0 for android	NaN	NaN	NaN	F	F	T	
506688	NaN		NaN	NaN	NaN	NaN	NaN	NaN	NaN
506689	NaN		NaN	NaN	NaN	NaN	NaN	NaN	NaN
506690	NaN	samsung browser 8.2	NaN	NaN	NaN	F	F	T	

	id-38	DeviceType	DeviceInfo
0	NaN	NaN	NaN
1	NaN	NaN	NaN
2	NaN	NaN	NaN
3	NaN	NaN	NaN
4	NaN	NaN	NaN
...
506686	NaN	NaN	NaN
506687	F	mobile ALE-L23 Build/Huawei	ALE-L23
506688	NaN	NaN	NaN
506689	NaN	NaN	NaN
506690	F	mobile	SAMSUNG

[506691 rows x 434 columns]

2.2 Shape of Datasets

```
[11]: print('train shape is {}'.format(train.shape))
      print('test shape is {}'.format(test.shape))
```

train shape is (590540, 435)

test shape is (506691, 434)

2.3 Features Overview

2.3.1 types of features

```
[12]: # pd.set_option('display.max_rows', None)
      sm = pd.DataFrame(train).dtypes
```

```
[13]: pd.set_option("display.precision", 1)
      pd.DataFrame(train.describe())
```

```
[13]:
```

	Unnamed: 0	TransactionID	isFraud	TransactionDT	TransactionAmt	\
count	590540.0	5.9e+05	5.9e+05	5.9e+05	590540.0	
mean	295269.5	3.3e+06	3.5e-02	7.4e+06	135.0	
std	170474.4	1.7e+05	1.8e-01	4.6e+06	239.2	
min	0.0	3.0e+06	0.0e+00	8.6e+04	0.3	
25%	147634.8	3.1e+06	0.0e+00	3.0e+06	43.3	
50%	295269.5	3.3e+06	0.0e+00	7.3e+06	68.8	

75%	442904.2	3.4e+06	0.0e+00	1.1e+07	125.0
max	590539.0	3.6e+06	1.0e+00	1.6e+07	31940.0

	card1	card2	card3	card5	addr1	addr2	dist1 \
count	590540.0	581607.0	588975.0	586281.0	524834.0	524834.0	238269.0
mean	9898.7	362.6	153.2	199.3	290.7	86.8	118.5
std	4901.2	157.8	11.3	41.2	101.7	2.7	371.9
min	1000.0	100.0	100.0	100.0	100.0	10.0	0.0
25%	6019.0	214.0	150.0	166.0	204.0	87.0	3.0
50%	9678.0	361.0	150.0	226.0	299.0	87.0	8.0
75%	14184.0	512.0	150.0	226.0	330.0	87.0	24.0
max	18396.0	600.0	231.0	237.0	540.0	102.0	10290.0

	dist2	C1	C2	C3	C4	C5	C6 \
count	37627.0	590540.0	590540.0	5.9e+05	590540.0	590540.0	590540.0
mean	231.9	14.1	15.3	5.6e-03	4.1	5.6	9.1
std	529.1	133.6	154.7	1.5e-01	68.8	25.8	71.5
min	0.0	0.0	0.0	0.0e+00	0.0	0.0	0.0
25%	7.0	1.0	1.0	0.0e+00	0.0	0.0	1.0
50%	37.0	1.0	1.0	0.0e+00	0.0	0.0	1.0
75%	206.0	3.0	3.0	0.0e+00	0.0	1.0	2.0
max	11624.0	4684.0	5692.0	2.6e+01	2252.0	349.0	2252.0

	C7	C8	C9	C10	C11	C12	C13 \
count	590540.0	590540.0	590540.0	590540.0	590540.0	590540.0	590540.0
mean	2.8	5.1	4.5	5.2	10.2	4.1	32.5
std	61.7	95.4	16.7	95.6	94.3	86.7	129.4
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	0.0	1.0	0.0	1.0
50%	0.0	0.0	1.0	0.0	1.0	0.0	3.0
75%	0.0	0.0	2.0	0.0	2.0	0.0	12.0
max	2256.0	3332.0	210.0	3256.0	3188.0	3188.0	2918.0

	C14	D1	D2	D3	D4	D5	D6 \
count	590540.0	589271.0	309743.0	327662.0	421618.0	280699.0	73187.0
mean	8.3	94.3	169.6	28.3	140.0	42.3	69.8
std	49.5	157.7	177.3	62.4	191.1	89.0	143.7
min	0.0	0.0	0.0	0.0	-122.0	0.0	-83.0
25%	1.0	0.0	26.0	1.0	0.0	1.0	0.0
50%	1.0	3.0	97.0	8.0	26.0	10.0	0.0
75%	2.0	122.0	276.0	27.0	253.0	32.0	40.0
max	1429.0	640.0	640.0	819.0	869.0	819.0	873.0

	D7	D8	D9	D10	D11	D12	D13 \
count	38917.0	74926.0	74926.0	514518.0	311253.0	64717.0	61952.0
mean	41.6	146.1	0.6	124.0	146.6	54.0	17.9
std	99.7	231.7	0.3	182.6	186.0	124.3	67.6

min	0.0	0.0	0.0	0.0	-53.0	-83.0	0.0
25%	0.0	1.0	0.2	0.0	0.0	0.0	0.0
50%	0.0	37.9	0.7	15.0	43.0	0.0	0.0
75%	17.0	188.0	0.8	197.0	274.0	13.0	0.0
max	843.0	1708.0	1.0	876.0	670.0	648.0	847.0

	D14	D15	V1	V2	V3	V4	V5 \
count	62187.0	501427.0	3.1e+05	311253.0	311253.0	311253.0	311253.0
mean	57.7	163.7	1.0e+00	1.0	1.1	0.8	0.9
std	136.3	202.7	7.4e-03	0.2	0.3	0.4	0.5
min	-193.0	-83.0	0.0e+00	0.0	0.0	0.0	0.0
25%	0.0	0.0	1.0e+00	1.0	1.0	1.0	1.0
50%	0.0	52.0	1.0e+00	1.0	1.0	1.0	1.0
75%	2.0	314.0	1.0e+00	1.0	1.0	1.0	1.0
max	878.0	879.0	1.0e+00	8.0	9.0	6.0	6.0

	V6	V7	V8	V9	V10	V11	V12 \
count	311253.0	311253.0	311253.0	311253.0	311253.0	311253.0	514467.0
mean	1.0	1.1	1.0	1.0	0.5	0.5	0.6
std	0.2	0.3	0.2	0.2	0.5	0.6	0.5
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	1.0	1.0	1.0	1.0	0.0	0.0	0.0
50%	1.0	1.0	1.0	1.0	0.0	0.0	1.0
75%	1.0	1.0	1.0	1.0	1.0	1.0	1.0
max	9.0	9.0	8.0	8.0	4.0	5.0	3.0

	V13	V14	V15	V16	V17	V18	V19 \
count	514467.0	5.1e+05	514467.0	514467.0	514467.0	514467.0	514467.0
mean	0.6	1.0e+00	0.1	0.1	0.1	0.1	0.8
std	0.5	2.2e-02	0.3	0.3	0.4	0.4	0.4
min	0.0	0.0e+00	0.0	0.0	0.0	0.0	0.0
25%	0.0	1.0e+00	0.0	0.0	0.0	0.0	1.0
50%	1.0	1.0e+00	0.0	0.0	0.0	0.0	1.0
75%	1.0	1.0e+00	0.0	0.0	0.0	0.0	1.0
max	6.0	1.0e+00	7.0	15.0	15.0	15.0	7.0

	V20	V21	V22	V23	V24	V25	V26 \
count	514467.0	514467.0	514467.0	514467.0	514467.0	514467.0	514467.0
mean	0.8	0.1	0.1	1.0	1.1	1.0	1.0
std	0.5	0.3	0.4	0.2	0.3	0.2	0.2
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	1.0	0.0	0.0	1.0	1.0	1.0	1.0
50%	1.0	0.0	0.0	1.0	1.0	1.0	1.0
75%	1.0	0.0	0.0	1.0	1.0	1.0	1.0
max	15.0	5.0	8.0	13.0	13.0	7.0	13.0

	V27	V28	V29	V30	V31	V32	V33 \
--	-----	-----	-----	-----	-----	-----	-------

count	5.1e+05	5.1e+05	514467.0	514467.0	514467.0	514467.0	514467.0
mean	7.8e-04	8.3e-04	0.4	0.4	0.1	0.1	0.1
std	2.9e-02	3.1e-02	0.5	0.6	0.4	0.4	0.3
min	0.0e+00	0.0e+00	0.0	0.0	0.0	0.0	0.0
25%	0.0e+00	0.0e+00	0.0	0.0	0.0	0.0	0.0
50%	0.0e+00	0.0e+00	0.0	0.0	0.0	0.0	0.0
75%	0.0e+00	0.0e+00	1.0	1.0	0.0	0.0	0.0
max	4.0e+00	4.0e+00	5.0	9.0	7.0	15.0	7.0

	V34	V35	V36	V37	V38	V39	V40 \
count	514467.0	421571.0	421571.0	421571.0	421571.0	421571.0	421571.0
mean	0.1	0.5	0.6	1.1	1.2	0.2	0.2
std	0.4	0.5	0.5	0.7	0.9	0.5	0.5
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	1.0	1.0	0.0	0.0
50%	0.0	1.0	1.0	1.0	1.0	0.0	0.0
75%	0.0	1.0	1.0	1.0	1.0	0.0	0.0
max	13.0	3.0	5.0	54.0	54.0	15.0	24.0

	V41	V42	V43	V44	V45	V46	V47 \
count	4.2e+05	421571.0	421571.0	421571.0	421571.0	421571.0	421571.0
mean	1.0e+00	0.2	0.2	1.1	1.1	1.0	1.0
std	2.7e-02	0.4	0.4	0.6	0.7	0.2	0.2
min	0.0e+00	0.0	0.0	0.0	0.0	0.0	0.0
25%	1.0e+00	0.0	0.0	1.0	1.0	1.0	1.0
50%	1.0e+00	0.0	0.0	1.0	1.0	1.0	1.0
75%	1.0e+00	0.0	0.0	1.0	1.0	1.0	1.0
max	1.0e+00	8.0	8.0	48.0	48.0	6.0	12.0

	V48	V49	V50	V51	V52	V53	V54 \
count	421571.0	421571.0	421571.0	421571.0	421571.0	513444.0	513444.0
mean	0.4	0.4	0.2	0.2	0.2	0.6	0.6
std	0.5	0.5	0.4	0.4	0.4	0.5	0.5
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	0.0	0.0	0.0	0.0	0.0	1.0	1.0
75%	1.0	1.0	0.0	0.0	0.0	1.0	1.0
max	5.0	5.0	5.0	6.0	12.0	5.0	6.0

	V55	V56	V57	V58	V59	V60	V61 \
count	513444.0	513444.0	513444.0	513444.0	513444.0	513444.0	513444.0
mean	1.1	1.1	0.1	0.1	0.1	0.1	0.8
std	0.4	0.7	0.3	0.4	0.4	0.4	0.4
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	1.0	1.0	0.0	0.0	0.0	0.0	1.0
50%	1.0	1.0	0.0	0.0	0.0	0.0	1.0
75%	1.0	1.0	0.0	0.0	0.0	0.0	1.0

max	17.0	51.0	6.0	10.0	16.0	16.0	6.0
-----	------	------	-----	------	------	------	-----

	V62	V63	V64	V65	V66	V67	V68 \
count	513444.0	513444.0	513444.0	5.1e+05	513444.0	513444.0	5.1e+05
mean	0.9	0.1	0.1	1.0e+00	1.0	1.0	5.3e-04
std	0.5	0.4	0.4	1.8e-02	0.2	0.2	2.4e-02
min	0.0	0.0	0.0	0.0e+00	0.0	0.0	0.0e+00
25%	1.0	0.0	0.0	1.0e+00	1.0	1.0	0.0e+00
50%	1.0	0.0	0.0	1.0e+00	1.0	1.0	0.0e+00
75%	1.0	0.0	0.0	1.0e+00	1.0	1.0	0.0e+00
max	10.0	7.0	7.0	1.0e+00	7.0	8.0	2.0e+00

	V69	V70	V71	V72	V73	V74	V75 \
count	513444.0	513444.0	513444.0	513444.0	513444.0	513444.0	501376.0
mean	0.4	0.4	0.1	0.1	0.1	0.2	0.5
std	0.5	0.6	0.4	0.4	0.4	0.4	0.5
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	0.0	0.0	0.0	0.0	0.0	0.0	1.0
75%	1.0	1.0	0.0	0.0	0.0	0.0	1.0
max	5.0	6.0	6.0	10.0	7.0	8.0	4.0

	V76	V77	V78	V79	V80	V81	V82 \
count	501376.0	501376.0	501376.0	501376.0	501376.0	501376.0	501376.0
mean	0.6	1.1	1.1	0.1	0.1	0.2	0.8
std	0.5	0.5	0.8	0.4	0.4	0.5	0.4
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	1.0	1.0	0.0	0.0	0.0	1.0
50%	1.0	1.0	1.0	0.0	0.0	0.0	1.0
75%	1.0	1.0	1.0	0.0	0.0	0.0	1.0
max	6.0	30.0	31.0	7.0	19.0	19.0	7.0

	V83	V84	V85	V86	V87	V88	V89 \
count	501376.0	501376.0	501376.0	501376.0	501376.0	5.0e+05	5.0e+05
mean	0.9	0.1	0.1	1.1	1.1	1.0e+00	9.0e-04
std	0.5	0.4	0.4	0.4	0.5	2.7e-02	3.2e-02
min	0.0	0.0	0.0	0.0	0.0	0.0e+00	0.0e+00
25%	1.0	0.0	0.0	1.0	1.0	1.0e+00	0.0e+00
50%	1.0	0.0	0.0	1.0	1.0	1.0e+00	0.0e+00
75%	1.0	0.0	0.0	1.0	1.0	1.0e+00	0.0e+00
max	7.0	7.0	7.0	30.0	30.0	1.0e+00	2.0e+00

	V90	V91	V92	V93	V94	V95	V96 \
count	501376.0	501376.0	501376.0	501376.0	501376.0	590226.0	590226.0
mean	0.4	0.4	0.2	0.2	0.1	1.0	3.0
std	0.5	0.6	0.4	0.4	0.3	21.0	40.2
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0

25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
75%	1.0	1.0	0.0	0.0	0.0	0.0	1.0
max	5.0	6.0	7.0	7.0	2.0	880.0	1410.0

	V97	V98	V99	V100	V101	V102	V103	\
count	590226.0	5.9e+05	590226.0	590226.0	590226.0	590226.0	590226.0	
mean	1.7	6.2e-02	0.9	0.3	0.9	1.8	1.3	
std	27.7	2.8e-01	2.7	0.9	20.6	35.9	25.7	
min	0.0	0.0e+00	0.0	0.0	0.0	0.0	0.0	
25%	0.0	0.0e+00	0.0	0.0	0.0	0.0	0.0	
50%	0.0	0.0e+00	0.0	0.0	0.0	0.0	0.0	
75%	0.0	0.0e+00	1.0	0.0	0.0	0.0	0.0	
max	976.0	1.2e+01	88.0	28.0	869.0	1285.0	928.0	

	V104	V105	V106	V107	V108	V109	V110	\
count	5.9e+05	590226.0	590226.0	5.9e+05	5.9e+05	590226.0	5.9e+05	
mean	8.5e-02	0.3	0.2	1.0e+00	1.0e+00	1.0	1.0e+00	
std	6.5e-01	3.4	1.8	2.0e-02	8.1e-02	0.1	9.7e-02	
min	0.0e+00	0.0	0.0	0.0e+00	0.0e+00	0.0	0.0e+00	
25%	0.0e+00	0.0	0.0	1.0e+00	1.0e+00	1.0	1.0e+00	
50%	0.0e+00	0.0	0.0	1.0e+00	1.0e+00	1.0	1.0e+00	
75%	0.0e+00	0.0	0.0	1.0e+00	1.0e+00	1.0	1.0e+00	
max	1.5e+01	99.0	55.0	1.0e+00	7.0e+00	7.0	7.0e+00	

	V111	V112	V113	V114	V115	V116	V117	\
count	5.9e+05	5.9e+05	5.9e+05	590226.0	590226.0	590226.0	5.9e+05	
mean	1.0e+00	1.0e+00	1.0e+00	1.0	1.0	1.0	1.0e+00	
std	7.1e-02	8.5e-02	7.5e-02	0.1	0.2	0.1	3.5e-02	
min	0.0e+00	0.0e+00	0.0e+00	0.0	0.0	0.0	0.0e+00	
25%	1.0e+00	1.0e+00	1.0e+00	1.0	1.0	1.0	1.0e+00	
50%	1.0e+00	1.0e+00	1.0e+00	1.0	1.0	1.0	1.0e+00	
75%	1.0e+00	1.0e+00	1.0e+00	1.0	1.0	1.0	1.0e+00	
max	9.0e+00	9.0e+00	9.0e+00	6.0	6.0	6.0	3.0e+00	

	V118	V119	V120	V121	V122	V123	V124	\
count	5.9e+05	5.9e+05	5.9e+05	5.9e+05	5.9e+05	590226.0	590226.0	
mean	1.0e+00	1.0e+00	1.0e+00	1.0e+00	1.0e+00	1.0	1.1	
std	4.1e-02	3.6e-02	4.2e-02	6.7e-02	4.9e-02	0.2	0.4	
min	0.0e+00	0.0e+00	0.0e+00	0.0e+00	0.0e+00	0.0	0.0	
25%	1.0e+00	1.0e+00	1.0e+00	1.0e+00	1.0e+00	1.0	1.0	
50%	1.0e+00	1.0e+00	1.0e+00	1.0e+00	1.0e+00	1.0	1.0	
75%	1.0e+00	1.0e+00	1.0e+00	1.0e+00	1.0e+00	1.0	1.0	
max	3.0e+00	3.0e+00	3.0e+00	3.0e+00	3.0e+00	13.0	13.0	

	V125	V126	V127	V128	V129	V130	V131	\
count	590226.0	590226.0	590226.0	590226.0	590226.0	590226.0	590226.0	

mean	1.1	130.0	336.6	204.1	8.8	92.2	31.1
std	0.3	2347.0	4238.7	3010.3	113.8	316.0	161.2
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	1.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	1.0	0.0	0.0	0.0	0.0	0.0	0.0
75%	1.0	0.0	108.0	0.0	0.0	59.0	0.0
max	13.0	160000.0	160000.0	160000.0	55140.0	55140.0	55140.0

	V132	V133	V134	V135	V136	V137	V138 \
count	590226.0	590226.0	590226.0	590226.0	590226.0	590226.0	8.2e+04
mean	103.5	204.9	146.0	17.3	38.8	26.4	3.6e-02
std	2266.1	3796.3	2773.0	293.8	451.8	348.3	4.3e-01
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0e+00
25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0e+00
50%	0.0	0.0	0.0	0.0	0.0	0.0	0.0e+00
75%	0.0	0.0	0.0	0.0	0.0	0.0	0.0e+00
max	93736.0	133915.0	98476.0	90750.0	90750.0	90750.0	2.2e+01

	V139	V140	V141	V142	V143	V144	V145	V146 \
count	81945.0	81945.0	8.2e+04	8.2e+04	81951.0	81951.0	81951.0	81945.0
mean	1.1	1.1	3.8e-02	4.9e-02	8.4	3.7	22.1	0.2
std	1.3	1.5	2.2e-01	3.1e-01	55.3	10.5	64.4	0.7
min	0.0	0.0	0.0e+00	0.0e+00	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0e+00	0.0e+00	0.0	0.0	0.0	0.0
50%	1.0	1.0	0.0e+00	0.0e+00	0.0	0.0	0.0	0.0
75%	1.0	1.0	0.0e+00	0.0e+00	0.0	0.0	1.0	0.0
max	33.0	33.0	5.0e+00	9.0e+00	869.0	62.0	297.0	24.0

	V147	V148	V149	V150	V151	V152	V153	V154 \
count	81945.0	81945.0	81945.0	81951.0	81951.0	81951.0	81945.0	81945.0
mean	0.2	0.8	0.8	277.6	6.5	9.4	0.8	0.8
std	0.8	0.6	0.6	829.6	15.2	21.6	0.5	0.5
min	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0
25%	0.0	0.0	0.0	1.0	1.0	1.0	0.0	0.0
50%	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
75%	0.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
max	26.0	20.0	20.0	3388.0	57.0	69.0	18.0	18.0

	V155	V156	V157	V158	V159	V160	V161 \
count	81945.0	81945.0	81945.0	81945.0	81951.0	81951.0	81945.0
mean	0.8	0.8	0.8	0.8	2719.3	47453.2	4.8
std	0.6	0.6	0.7	0.7	8355.3	142076.1	58.9
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	1.0	1.0	1.0	1.0	0.0	0.0	0.0
75%	1.0	1.0	1.0	1.0	0.0	0.0	0.0
max	24.0	24.0	24.0	24.0	55140.0	641511.4	3300.0

	V162	V163	V164	V165	V166	V167	V168 \
count	81945.0	81945.0	81951.0	81951.0	81951.0	139631.0	139631.0
mean	6.6	5.5	877.9	2239.9	359.5	3.9	5.9
std	69.2	63.1	6049.2	8223.3	1244.5	42.2	54.0
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
75%	0.0	0.0	0.0	0.0	0.0	0.0	1.0
max	3300.0	3300.0	93736.0	98476.0	104060.0	872.0	964.0

	V169	V170	V171	V172	V173	V174	V175 \
count	139819.0	139819.0	139819.0	139631.0	1.4e+05	139819.0	139819.0
mean	0.2	1.4	1.7	0.1	5.5e-02	0.1	0.2
std	0.9	1.8	2.4	0.9	2.6e-01	0.4	0.9
min	0.0	0.0	0.0	0.0	0.0e+00	0.0	0.0
25%	0.0	1.0	1.0	0.0	0.0e+00	0.0	0.0
50%	0.0	1.0	1.0	0.0	0.0e+00	0.0	0.0
75%	0.0	1.0	1.0	0.0	0.0e+00	0.0	0.0
max	19.0	48.0	61.0	31.0	7.0e+00	8.0	14.0

	V176	V177	V178	V179	V180	V181	V182 \
count	139631.0	139631.0	139631.0	139631.0	139819.0	139631.0	139631.0
mean	1.4	3.5	6.6	4.9	0.9	0.3	0.9
std	1.8	41.3	69.4	50.5	6.1	1.2	5.8
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	1.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	1.0	0.0	0.0	0.0	0.0	0.0	0.0
75%	1.0	0.0	0.0	0.0	0.0	0.0	0.0
max	48.0	861.0	1235.0	920.0	83.0	24.0	83.0

	V183	V184	V185	V186	V187	V188	V189 \
count	139631.0	139819.0	139819.0	139631.0	139631.0	139819.0	139819.0
mean	0.5	0.1	0.2	1.1	1.8	1.0	1.0
std	2.9	0.5	0.7	1.2	9.9	0.7	0.8
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	1.0	1.0	1.0	1.0
50%	0.0	0.0	0.0	1.0	1.0	1.0	1.0
75%	0.0	0.0	0.0	1.0	1.0	1.0	1.0
max	41.0	16.0	31.0	38.0	218.0	30.0	30.0

	V190	V191	V192	V193	V194	V195	V196 \
count	139631.0	139631.0	139631.0	139631.0	139819.0	139819.0	139631.0
mean	1.2	1.1	1.2	1.1	0.9	1.0	1.1
std	1.5	0.7	2.6	1.7	0.3	0.4	1.1
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	1.0	1.0	1.0	1.0	1.0	1.0	1.0

50%	1.0	1.0	1.0	1.0	1.0	1.0	1.0
75%	1.0	1.0	1.0	1.0	1.0	1.0	1.0
max	42.0	21.0	44.0	37.0	7.0	16.0	38.0

	V197	V198	V199	V200	V201	V202	V203	\
count	139819.0	139819.0	139631.0	139819.0	139819.0	139631.0	139631.0	
mean	0.9	1.0	1.3	1.1	1.2	444.1	1078.3	
std	0.3	0.5	1.7	1.3	1.4	4683.8	9105.6	
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
25%	1.0	1.0	1.0	1.0	1.0	0.0	0.0	
50%	1.0	1.0	1.0	1.0	1.0	0.0	0.0	
75%	1.0	1.0	1.0	1.0	1.0	0.0	30.9	
max	14.0	21.0	45.0	45.0	55.0	104060.0	139777.0	

	V204	V205	V206	V207	V208	V209	V210	\
count	139631.0	139631.0	139631.0	139631.0	139819.0	139819.0	139819.0	
mean	687.0	18.1	6.2	72.3	8.9	35.0	14.4	
std	6049.0	266.6	191.5	925.6	59.0	250.8	86.5	
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
50%	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
75%	20.0	0.0	0.0	0.0	0.0	0.0	0.0	
max	104060.0	55140.0	55140.0	55140.0	3300.0	8050.0	3300.0	

	V211	V212	V213	V214	V215	V216	V217	\
count	139631.0	139631.0	139631.0	139631.0	139631.0	139631.0	130430.0	
mean	385.1	766.0	536.3	38.4	133.2	71.1	1.1	
std	4541.8	7496.1	5471.7	571.8	1040.5	680.3	9.5	
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
50%	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
75%	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
max	92888.0	129006.0	97628.0	104060.0	104060.0	104060.0	303.0	

	V218	V219	V220	V221	V222	V223	V224	\
count	130430.0	130430.0	141416.0	141416.0	141416.0	1.3e+05	130430.0	
mean	1.7	1.4	0.2	1.3	1.4	9.3e-02	0.4	
std	13.9	12.0	1.0	2.7	2.9	4.0e-01	3.4	
min	0.0	0.0	0.0	0.0	0.0	0.0e+00	0.0	
25%	0.0	0.0	0.0	1.0	1.0	0.0e+00	0.0	
50%	0.0	0.0	0.0	1.0	1.0	0.0e+00	0.0	
75%	1.0	1.0	0.0	1.0	1.0	0.0e+00	0.0	
max	400.0	378.0	25.0	384.0	384.0	1.6e+01	144.0	

	V225	V226	V227	V228	V229	V230	V231	\
count	130430.0	130430.0	141416.0	130430.0	130430.0	130430.0	130430.0	
mean	0.2	0.2	0.1	1.4	1.6	1.5	0.8	

std	1.3	2.8	2.0	1.4	4.3	2.1	8.9
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	1.0	1.0	1.0	0.0
50%	0.0	0.0	0.0	1.0	1.0	1.0	0.0
75%	0.0	0.0	0.0	1.0	1.0	1.0	0.0
max	51.0	242.0	360.0	54.0	176.0	65.0	293.0

	V232	V233	V234	V235	V236	V237	V238 \
count	130430.0	130430.0	141416.0	130430.0	130430.0	130430.0	141416.0
mean	1.0	0.9	2.1	0.2	0.3	0.3	0.1
std	10.5	10.1	11.3	0.9	2.2	1.7	0.6
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
75%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
max	337.0	332.0	121.0	23.0	45.0	39.0	23.0

	V239	V240	V241	V242	V243	V244	V245 \
count	141416.0	1.3e+05	1.3e+05	130430.0	130430.0	130430.0	141416.0
mean	0.1	1.0e+00	1.0e+00	1.1	1.2	1.1	0.9
std	0.6	5.0e-02	2.3e-02	0.7	1.4	0.7	1.8
min	0.0	0.0e+00	0.0e+00	0.0	0.0	0.0	0.0
25%	0.0	1.0e+00	1.0e+00	1.0	1.0	1.0	1.0
50%	0.0	1.0e+00	1.0e+00	1.0	1.0	1.0	1.0
75%	0.0	1.0e+00	1.0e+00	1.0	1.0	1.0	1.0
max	23.0	7.0e+00	5.0e+00	20.0	57.0	22.0	262.0

	V246	V247	V248	V249	V250	V251	V252 \
count	130430.0	130430.0	130430.0	130430.0	141416.0	141416.0	130430.0
mean	1.2	1.0	1.1	1.0	0.8	0.8	1.0
std	1.0	0.3	0.9	0.5	0.5	0.5	0.4
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	1.0	1.0	1.0	1.0	1.0	1.0	1.0
50%	1.0	1.0	1.0	1.0	1.0	1.0	1.0
75%	1.0	1.0	1.0	1.0	1.0	1.0	1.0
max	45.0	18.0	36.0	22.0	18.0	18.0	24.0

	V253	V254	V255	V256	V257	V258	V259 \
count	130430.0	130430.0	141416.0	141416.0	130430.0	130430.0	141416.0
mean	1.2	1.1	0.8	0.8	1.3	1.3	1.0
std	3.6	1.4	0.9	1.0	1.3	2.0	2.1
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	1.0	1.0	1.0	1.0	1.0	1.0	1.0
50%	1.0	1.0	1.0	1.0	1.0	1.0	1.0
75%	1.0	1.0	1.0	1.0	1.0	1.0	1.0
max	163.0	60.0	87.0	87.0	48.0	66.0	285.0

	V260	V261	V262	V263	V264	V265	V266 \
count	130430.0	130430.0	130430.0	130430.0	130430.0	130430.0	130430.0
mean	1.0	1.1	1.0	117.4	201.7	153.5	9.2
std	0.4	1.3	0.6	1294.9	2284.8	1605.5	208.1
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	1.0	1.0	1.0	0.0	0.0	0.0	0.0
50%	1.0	1.0	1.0	0.0	0.0	0.0	0.0
75%	1.0	1.0	1.0	0.0	33.6	20.9	0.0
max	8.0	49.0	20.0	153600.0	153600.0	153600.0	55140.0

	V267	V268	V269	V270	V271	V272	V273 \
count	130430.0	130430.0	130430.0	141416.0	141416.0	141416.0	130430.0
mean	36.5	18.8	6.0	7.7	9.4	8.5	73.8
std	644.9	311.3	207.4	65.5	74.1	69.7	935.2
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
75%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
max	55140.0	55140.0	55140.0	4000.0	4000.0	4000.0	51200.0

	V274	V275	V276	V277	V278	V279	V280 \
count	130430.0	130430.0	130430.0	130430.0	130430.0	590528.0	590528.0
mean	107.2	88.9	31.8	52.0	42.3	1.1	2.0
std	1258.7	1072.9	615.7	732.1	660.6	21.0	27.9
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
75%	0.0	0.0	0.0	0.0	0.0	0.0	1.0
max	66000.0	51200.0	104060.0	104060.0	104060.0	880.0	975.0

	V281	V282	V283	V284	V285	V286	V287 \
count	5.9e+05	589271.0	589271.0	5.9e+05	590528.0	5.9e+05	590528.0
mean	8.8e-02	0.8	1.0	8.9e-02	1.2	3.1e-02	0.4
std	5.1e-01	0.9	1.6	3.4e-01	3.3	1.9e-01	1.1
min	0.0e+00	0.0	0.0	0.0e+00	0.0	0.0e+00	0.0
25%	0.0e+00	0.0	0.0	0.0e+00	0.0	0.0e+00	0.0
50%	0.0e+00	1.0	1.0	0.0e+00	0.0	0.0e+00	0.0
75%	0.0e+00	1.0	1.0	0.0e+00	1.0	0.0e+00	0.0
max	2.2e+01	32.0	68.0	1.2e+01	95.0	8.0e+00	31.0

	V288	V289	V290	V291	V292	V293	V294 \
count	589271.0	589271.0	590528.0	590528.0	590528.0	590528.0	590528.0
mean	0.2	0.2	1.1	1.7	1.2	0.9	2.3
std	0.4	0.6	0.8	16.3	3.8	20.6	39.5
min	0.0	0.0	1.0	1.0	1.0	0.0	0.0
25%	0.0	0.0	1.0	1.0	1.0	0.0	0.0
50%	0.0	0.0	1.0	1.0	1.0	0.0	0.0

75%	0.0	0.0	1.0	1.0	1.0	0.0	0.0
max	10.0	12.0	67.0	1055.0	323.0	869.0	1286.0

	V295	V296	V297	V298	V299	V300	V301 \
count	590528.0	589271.0	5.9e+05	590528.0	590528.0	5.9e+05	5.9e+05
mean	1.4	0.3	8.9e-02	0.3	0.2	4.6e-02	5.2e-02
std	26.0	3.3	6.3e-01	3.2	1.7	2.9e-01	3.2e-01
min	0.0	0.0	0.0e+00	0.0	0.0	0.0e+00	0.0e+00
25%	0.0	0.0	0.0e+00	0.0	0.0	0.0e+00	0.0e+00
50%	0.0	0.0	0.0e+00	0.0	0.0	0.0e+00	0.0e+00
75%	0.0	0.0	0.0e+00	0.0	0.0	0.0e+00	0.0e+00
max	928.0	93.0	1.2e+01	93.0	49.0	1.1e+01	1.3e+01

	V302	V303	V304	V305	V306	V307	V308 \
count	590528.0	590528.0	590528.0	5.9e+05	590528.0	590528.0	590528.0
mean	0.3	0.3	0.3	1.0e+00	139.7	408.7	230.4
std	0.5	0.6	0.5	2.6e-03	2348.8	4392.0	3021.9
min	0.0	0.0	0.0	1.0e+00	0.0	0.0	0.0
25%	0.0	0.0	0.0	1.0e+00	0.0	0.0	0.0
50%	0.0	0.0	0.0	1.0e+00	0.0	0.0	0.0
75%	0.0	0.0	0.0	1.0e+00	0.0	151.4	36.0
max	16.0	20.0	16.0	2.0e+00	108800.0	145765.0	108800.0

	V309	V310	V311	V312	V313	V314	V315 \
count	590528.0	590528.0	590528.0	590528.0	589271.0	589271.0	589271.0
mean	11.0	118.2	4.2	39.2	21.4	43.3	26.8
std	116.3	353.0	102.4	172.1	95.9	173.6	116.9
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
75%	0.0	107.9	0.0	0.0	0.0	0.0	0.0
max	55140.0	55140.0	55140.0	55140.0	4816.0	7520.0	4816.0

	V316	V317	V318	V319	V320	V321	V322 \
count	590528.0	590528.0	590528.0	590528.0	590528.0	590528.0	82351.0
mean	109.8	247.6	162.2	18.4	42.1	28.3	6.2
std	2270.0	3980.0	2793.3	332.3	473.5	382.1	56.0
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
75%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
max	93736.0	134021.0	98476.0	104060.0	104060.0	104060.0	880.0

	V323	V324	V325	V326	V327	V328	V329	V330 \
count	82351.0	82351.0	8.2e+04	82351.0	82351.0	82351.0	82351.0	82351.0
mean	13.1	9.2	5.8e-02	0.9	0.3	0.3	1.3	0.8
std	106.7	73.6	3.0e-01	4.0	1.4	1.6	8.8	4.7

min	0.0	0.0	0.0e+00	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0e+00	0.0	0.0	0.0	0.0	0.0
50%	0.0	0.0	0.0e+00	0.0	0.0	0.0	0.0	0.0
75%	1.0	0.0	0.0e+00	0.0	0.0	0.0	0.0	0.0
max	1411.0	976.0	1.2e+01	44.0	18.0	15.0	99.0	55.0

	V331	V332	V333	V334	V335	V336	V337 \
count	82351.0	82351.0	82351.0	82351.0	82351.0	82351.0	82351.0
mean	721.7	1375.8	1014.6	9.8	59.2	28.5	55.4
std	6217.2	11169.3	7955.7	243.9	387.6	274.6	668.5
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
50%	0.0	0.0	0.0	0.0	0.0	0.0	0.0
75%	0.0	25.0	0.0	0.0	0.0	0.0	0.0
max	160000.0	160000.0	160000.0	55140.0	55140.0	55140.0	104060.0

	V338	V339	id_01	id_02	id_03	id_04	id_05 \
count	82351.0	82351.0	144233.0	140872.0	6.6e+04	6.6e+04	136865.0
mean	151.2	100.7	-10.2	174716.6	6.0e-02	-5.9e-02	1.6
std	1095.0	814.9	14.3	159651.8	6.0e-01	7.0e-01	5.2
min	0.0	0.0	-100.0	1.0	-1.3e+01	-2.8e+01	-72.0
25%	0.0	0.0	-10.0	67992.0	0.0e+00	0.0e+00	0.0
50%	0.0	0.0	-5.0	125800.5	0.0e+00	0.0e+00	0.0
75%	0.0	0.0	-5.0	228749.0	0.0e+00	0.0e+00	1.0
max	104060.0	104060.0	0.0	999595.0	1.0e+01	0.0e+00	52.0

	id_06	id_07	id_08	id_09	id_10	id_11	id_13 \
count	136865.0	5155.0	5155.0	7.5e+04	74926.0	140978.0	127320.0
mean	-6.7	13.3	-38.6	9.1e-02	-0.3	99.7	48.1
std	16.5	11.4	26.1	9.8e-01	2.8	1.1	11.8
min	-100.0	-46.0	-100.0	-3.6e+01	-100.0	90.0	10.0
25%	-6.0	5.0	-48.0	0.0e+00	0.0	100.0	49.0
50%	0.0	14.0	-34.0	0.0e+00	0.0	100.0	52.0
75%	0.0	22.0	-23.0	0.0e+00	0.0	100.0	52.0
max	0.0	61.0	0.0	2.5e+01	0.0	100.0	64.0

	id_14	id_17	id_18	id_19	id_20	id_21	id_22	id_24 \
count	80044.0	139369.0	45113.0	139318.0	139261.0	5159.0	5169.0	4747.0
mean	-344.5	189.5	14.2	353.1	403.9	368.3	16.0	12.8
std	93.7	30.4	1.6	141.1	152.2	198.8	6.9	2.4
min	-660.0	100.0	10.0	100.0	100.0	100.0	10.0	11.0
25%	-360.0	166.0	13.0	266.0	256.0	252.0	14.0	11.0
50%	-300.0	166.0	15.0	341.0	472.0	252.0	14.0	11.0
75%	-300.0	225.0	15.0	427.0	533.0	486.5	14.0	15.0
max	720.0	229.0	29.0	671.0	661.0	854.0	44.0	26.0

id_25	id_26	id_32
-------	-------	-------

count	5132.0	5163.0	77586.0
mean	329.6	149.1	26.5
std	97.5	32.1	3.7
min	100.0	100.0	0.0
25%	321.0	119.0	24.0
50%	321.0	149.0	24.0
75%	371.0	169.0	32.0
max	548.0	216.0	32.0

2.3.2 Statistical Description and Boxplots

TransactionDT

```
[14]: print('type of TransactionDT is ' + str(train["TransactionDT"].dtypes))
```

type of TransactionDT is int64

The *TransactionDT* feature is a timedelta from a given reference datetime (not an actual timestamp).

```
[15]: sm = pd.DataFrame(
    [
        ['count', train['TransactionDT'].count(), test['TransactionDT'].count()],
        ['min', train.TransactionDT.min(), test.TransactionDT.min()],
        ['max', train.TransactionDT.max(), test.TransactionDT.max()],
        ['range', train.TransactionDT.max() - train.TransactionDT.min(), test.
→ TransactionDT.max() - test.TransactionDT.min()],
        ['mid-range', (train.TransactionDT.max() - train.TransactionDT.min())/2,
→ (test.TransactionDT.max() - test.TransactionDT.min())/2],
        ['mean', train.TransactionDT.mean(), test.TransactionDT.mean()],
        ['median', train.TransactionDT.median(), test.TransactionDT.median()],
        ['std.dev', train.TransactionDT.std(), test.TransactionDT.std()],
        ['variance', train.TransactionDT.var(), test.TransactionDT.var()],
        ['mode', train.TransactionDT.mode(), test.TransactionDT.mode()],
        ['25%', train.TransactionDT.quantile(0.25), test.TransactionDT.quantile(0.
→ 25)],
        ['50%', train.TransactionDT.quantile(0.50), test.TransactionDT.quantile(0.
→ 50)],
        ['75%', train.TransactionDT.quantile(0.75), test.TransactionDT.quantile(0.
→ 75)],
        ['IQR', train.TransactionDT.quantile(0.75) - train.TransactionDT.quantile(0.
→ 25), test.TransactionDT.quantile(0.75) - test.TransactionDT.quantile(0.25)]
    ],
    columns=['property', 'value_of_train', 'value_of_test']
)
# sm['value_of_train'] = sm.value_of_train.astype(int)
# sm['value_of_test'] = sm.value_of_test.astype(int)
sm
```



```

[15]:      property                                value_of_train \
0      count                                590540
1      min                                  86400
2      max                                  15811131
3      range                                15724731
4  mid-range                                7862365.5
5      mean                                  7372311.3
6      median                                7306527.5
7      std.dev                               4617223.6
8  variance                                21318754202165.2
9      mode  0      9474817
Name: TransactionDT, dtype: int64
10      25%                                3027057.8
11      50%                                7306527.5
12      75%                                11246620.0
13      IQR                                8219562.2

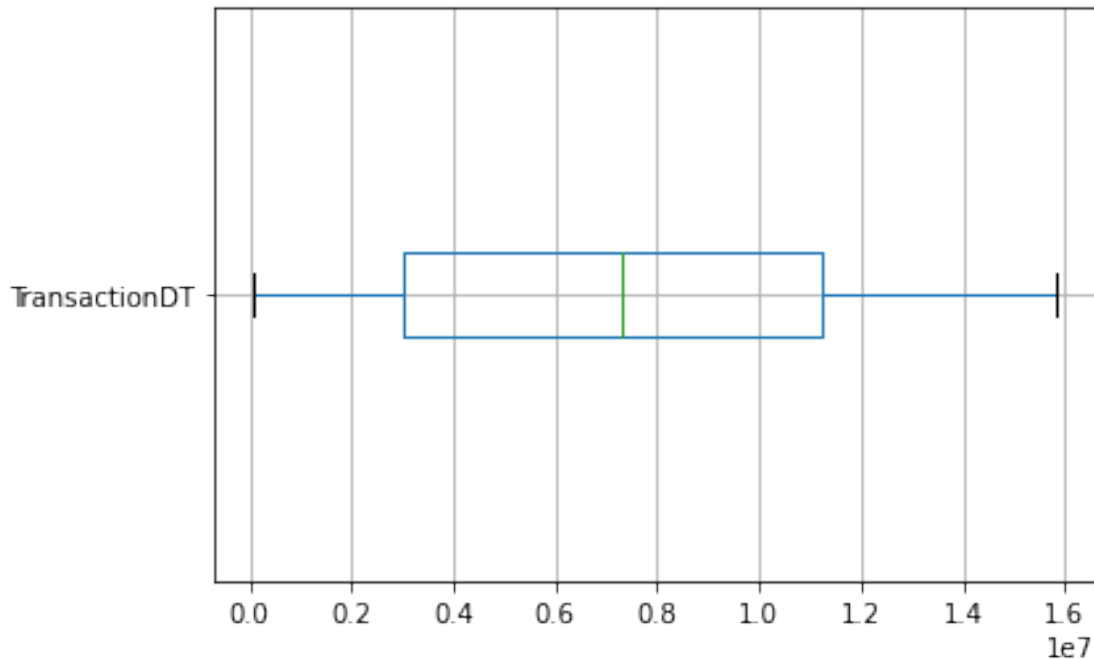
                                value_of_test
0                                506691
1                                18403224
2                                34214345
3                                15811121
4                                7905560.5
5                                26929937.2
6                                27204658.0
7                                4756506.7
8                                22624356079919.6
9  0      19601418
1      26576684
2      31790084
3      ...
10                                22771540.5
11                                27204658.0
12                                31348560.5
13                                8577020.0

```

```

[16]: train[['TransactionDT']].boxplot(vert = False)
      plt.show()

```



```
[17]: print("train: min = " + str(train.TransactionDT.min()) + ", train: max = " +
        str(train.TransactionDT.max()))
        print("test: min = " + str(test.TransactionDT.min()) + ", test: max = " +
        str(test.TransactionDT.max()))
```

```
train: min = 86400, train: max = 15811131
test: min = 18403224, test: max = 34214345
```

```
[18]: time_span = (test.TransactionDT.max() - train.TransactionDT.min()) / 60 / 60 / 24
        print("time span of the total datasets is: " + str(round(time_span)) + 'days')
```

```
time span of the total datasets is: 395days
```

ProductCD

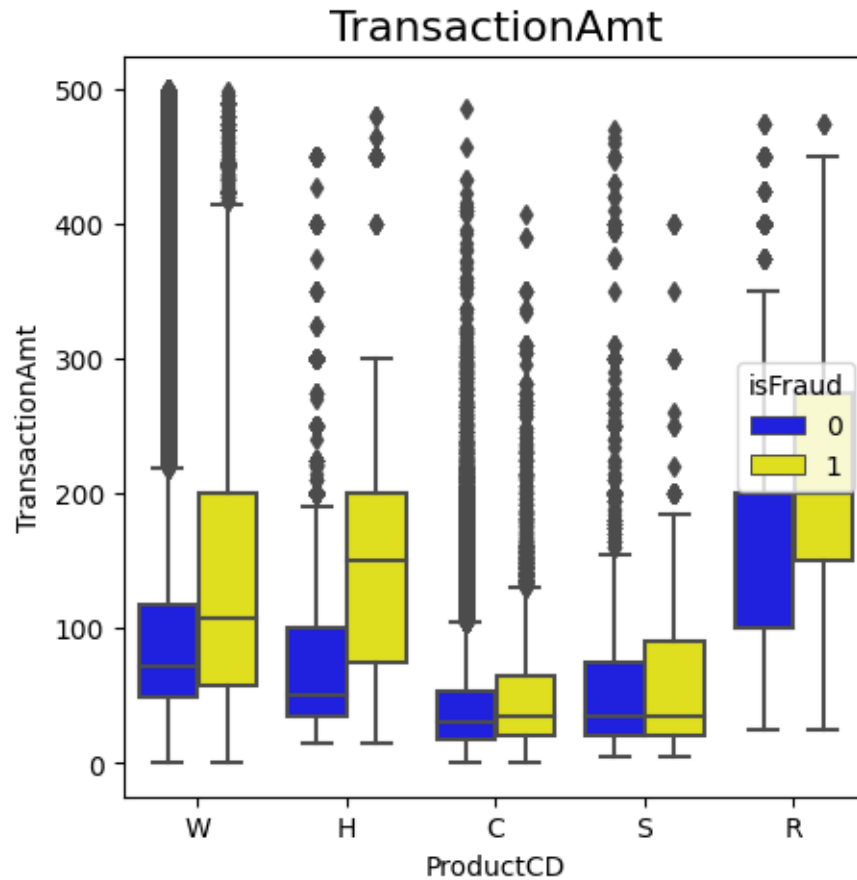
```
[19]: print('type of ProductCD is ' + str(train["ProductCD"].dtypes))
```

```
type of ProductCD is object
```

Products in this dataset come under five broad categories: W, H, C, S, and R.

We will visualize this feature in relation with the target, *isFraud*.

```
[20]: fig, ax = plt.subplots(figsize=(5, 5))
        plot = sns.boxplot(x="ProductCD", y="TransactionAmt", hue = "isFraud",
        data=train.query("TransactionAmt < 500"), palette=["blue", "yellow"], ax=ax).
        set_title('TransactionAmt', fontsize=16)
        plt.show()
```



Except for C and S, the non-fraudulent mean, first quartile, and third quartile of the products are higher than the fraudulent ones.

TransactionAmt

```
[21]: print('type of TransactionAmt is ' + str(train["TransactionAmt"].dtypes))
```

```
type of TransactionAmt is float64
```

This is the amount of money transferred during the transaction, which is a continuous variable. We will visualize this feature in relation with the target - *isFraud*.

```
[22]: sm = pd.DataFrame(
    [
        ['count', train['TransactionAmt'].count()],
        ['min', train.TransactionAmt.min()],
        ['max', train.TransactionAmt.max()],
        ['range', train.TransactionAmt.max() - train.TransactionAmt.min()],
        ['mid-range', (train.TransactionAmt.max() - train.TransactionAmt.min())/
→2],
        ['mean', train.TransactionAmt.mean()],
```

```

        ['median',train.TransactionAmt.median()],
        ['std.dev',train.TransactionAmt.std()],
        ['variance',train.TransactionAmt.var()],
        ['mode',train.TransactionAmt.mode()],
        ['25%',train.TransactionAmt.quantile(0.25)],
        ['50%',train.TransactionAmt.quantile(0.50)],
        ['75%',train.TransactionAmt.quantile(0.75)],
        ['IQR',train.TransactionAmt.quantile(0.75)-train.TransactionAmt.
→quantile(0.25)]
    ],
    columns=['property','value']
)
sm

```

```

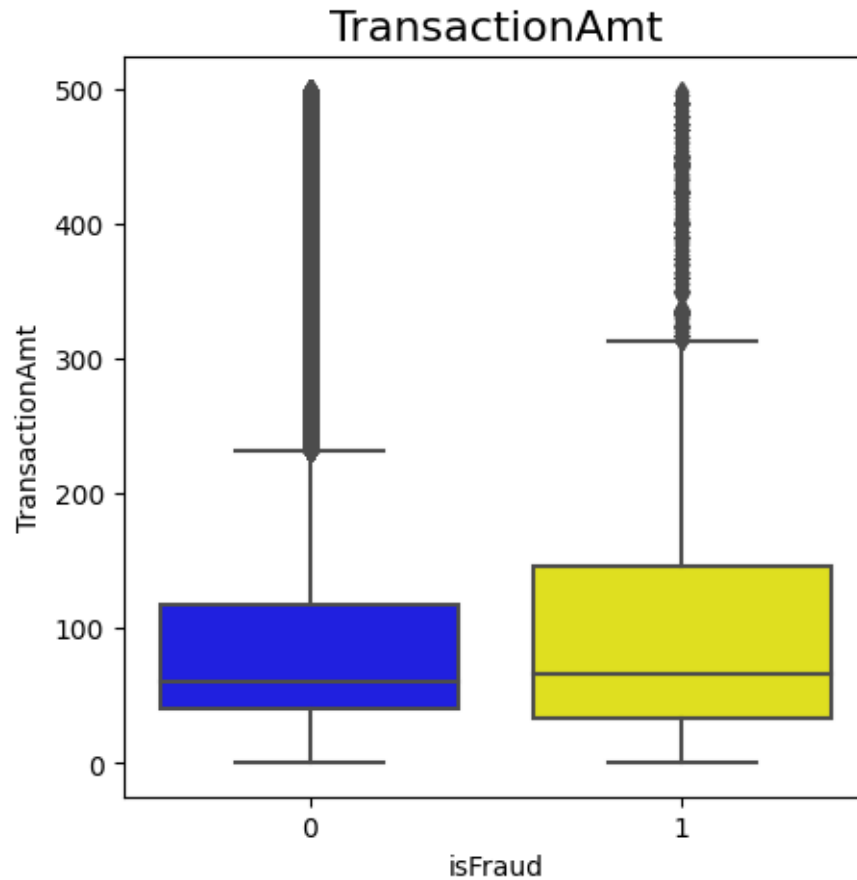
[22]:
      property      value
0      count    590540
1         min         0.3
2         max    31940.0
3        range   -54460.0
4   mid-range    15969.9
5         mean     135.0
6        median     68.8
7      std.dev    239.2
8   variance    57196.9
9      mode 0    59.0
Name: TransactionAmt, dtype: float64
10      25%         43.3
11      50%         68.8
12      75%        125.0
13      IQR         81.7

```

```

[23]: fig, ax = plt.subplots(figsize=(5, 5))
plot = sns.boxplot(x="isFraud", y="TransactionAmt", data=train.
→query("TransactionAmt < 500"), palette=["blue", "yellow"], ax=ax).
→set_title('TransactionAmt', fontsize=16)
plt.show()

```



The above box plot implies that transactions with higher prices are more likely to be fraudulent. This can be deduced from the fact that the yellow box's mean value is higher than the blue box's. The third quartile of the yellow box is significantly higher than that of the blue box, despite the first quartiles of the two distributions being very similar. This difference serves as additional proof that the higher the transaction amount, the more likely the transaction to be fraudulent.

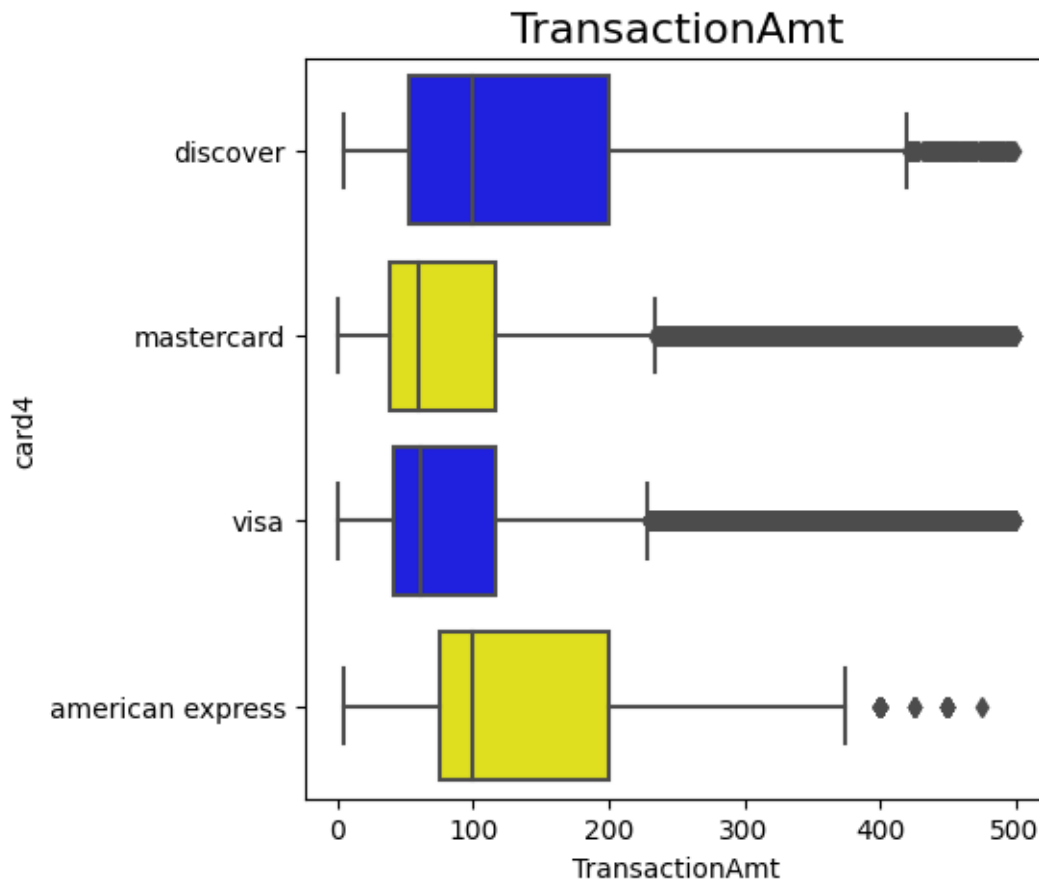
card brand (card4)

```
[24]: print('type of card4 is ' + str(train["card4"].dtypes))
```

type of card4 is object

The *card4* refers to the brand of the card, which are discover, mastercard, visa, and american express.

```
[25]: fig, ax = plt.subplots(figsize=(5, 5))
plot = sns.boxplot(x="TransactionAmt", y="card4", data=train.
    →query("TransactionAmt < 500"), palette=["blue", "yellow"], ax=ax).
    →set_title('TransactionAmt', fontsize=16)
plt.show()
```



The distributions for fraudulent and legitimate Mastercard and Visa cards are remarkably similar. Furthermore, it is evident from this box plot that Discover and American Express cards are frequently used for larger transaction amounts. They have far higher means than the other two card brands.

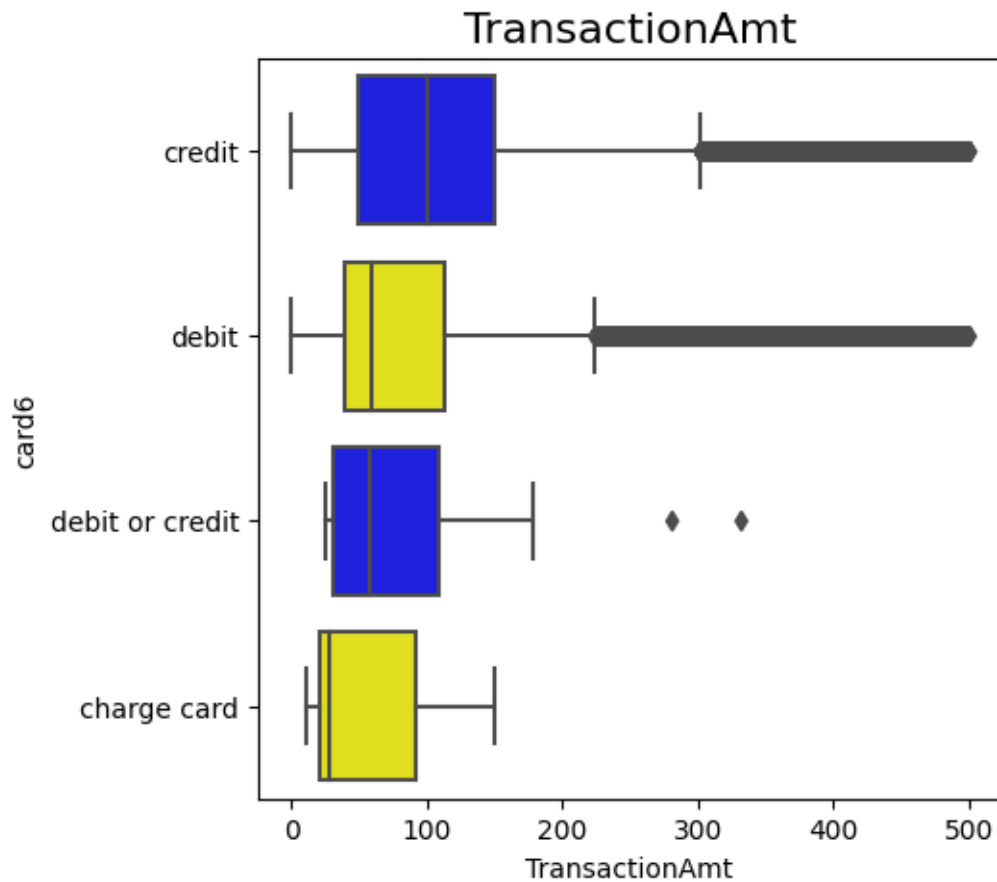
Card type (card6)

```
[26]: print('type of card6 is ' + str(train["card6"].dtypes))
```

type of card6 is object

The *card6* refers to the btype of the card, which are credit and debit.

```
[27]: fig, ax = plt.subplots(figsize=(5, 5))
plot = sns.boxplot(x="TransactionAmt", y="card6", data=train.
    →query("TransactionAmt < 500"), palette=["blue", "yellow"], ax=ax).
    →set_title('TransactionAmt', fontsize=16)
plt.show()
```



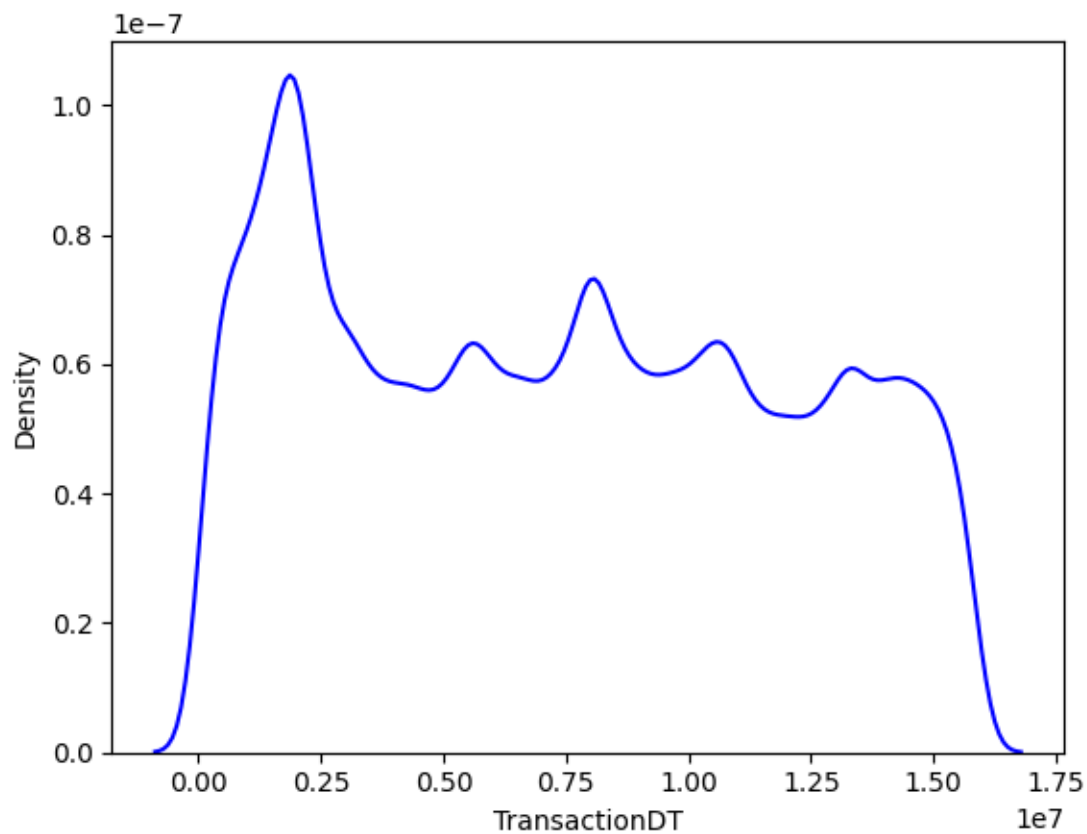
It can be seen from the boxplot above, the average transaction value of credit cards is much higher compared to debit cards.

3 Skewness

3.1 TransactionDT

```
[35]: sns.kdeplot(train['TransactionDT'], color = 'b')
      print('skewness')
      print(train[['TransactionDT']].skew())
```

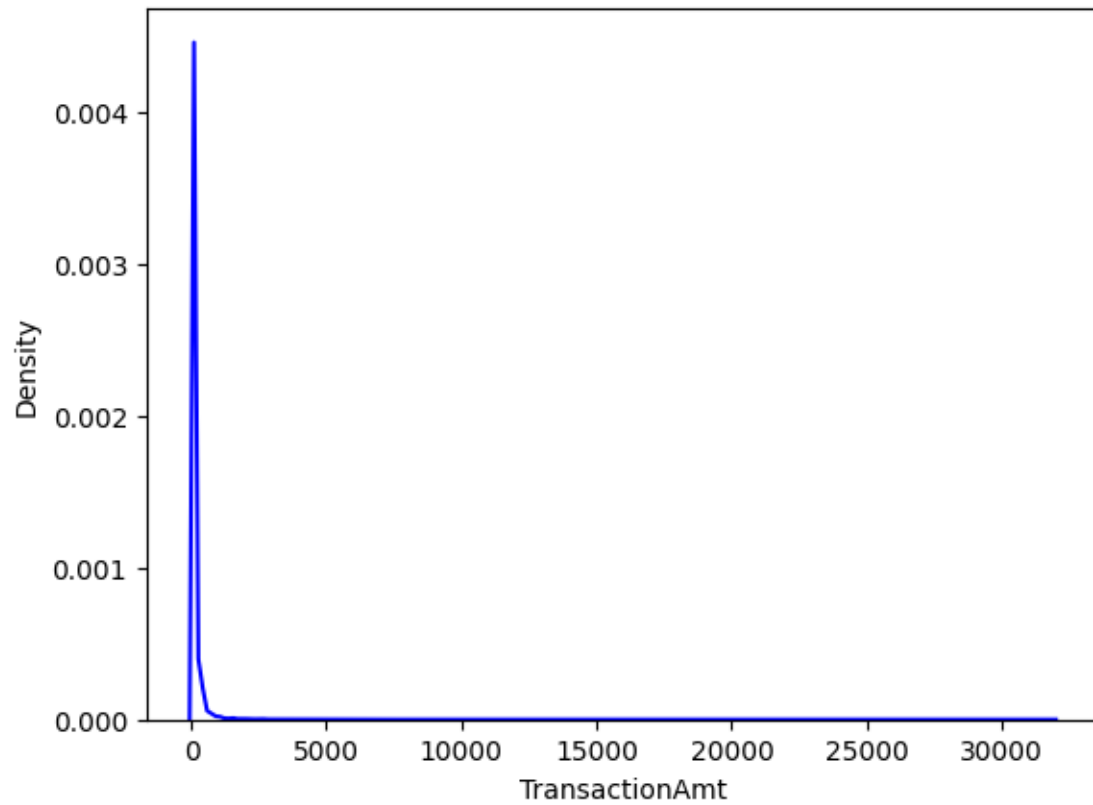
```
skewness
TransactionDT    0.1
dtype: float64
```



3.2 TransactionAmt

```
[36]: sns.kdeplot(train['TransactionAmt'], color = 'b')  
      print('skewness')  
      print(train[['TransactionAmt']].skew())
```

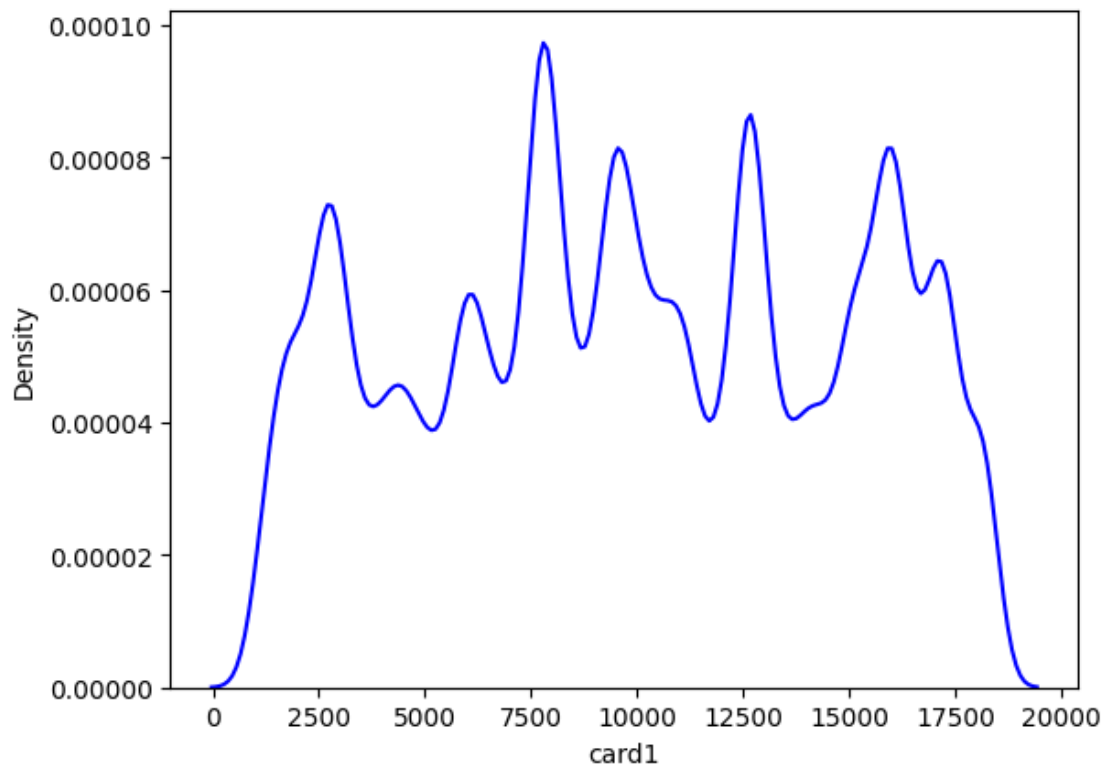
```
skewness  
TransactionAmt    14.4  
dtype: float64
```

3.3 Card1

```
[37]: sns.kdeplot(train['card1'], color = 'b')  
      print('skewness')  
      print(train[['card1']].skew())
```

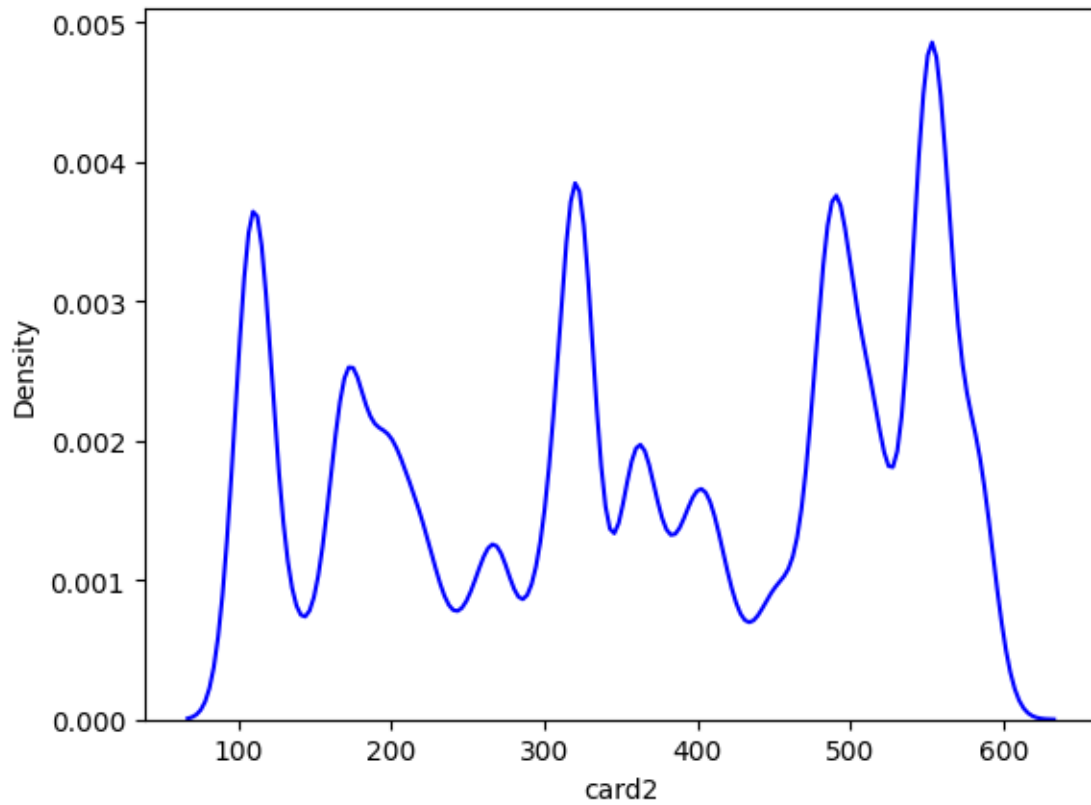
```
skewness  
card1    -4.1e-02  
dtype: float64
```



3.4 Card2

```
[38]: sns.kdeplot(train['card2'], color = 'b')  
      print('skewness')  
      print(train[['card2']].skew())
```

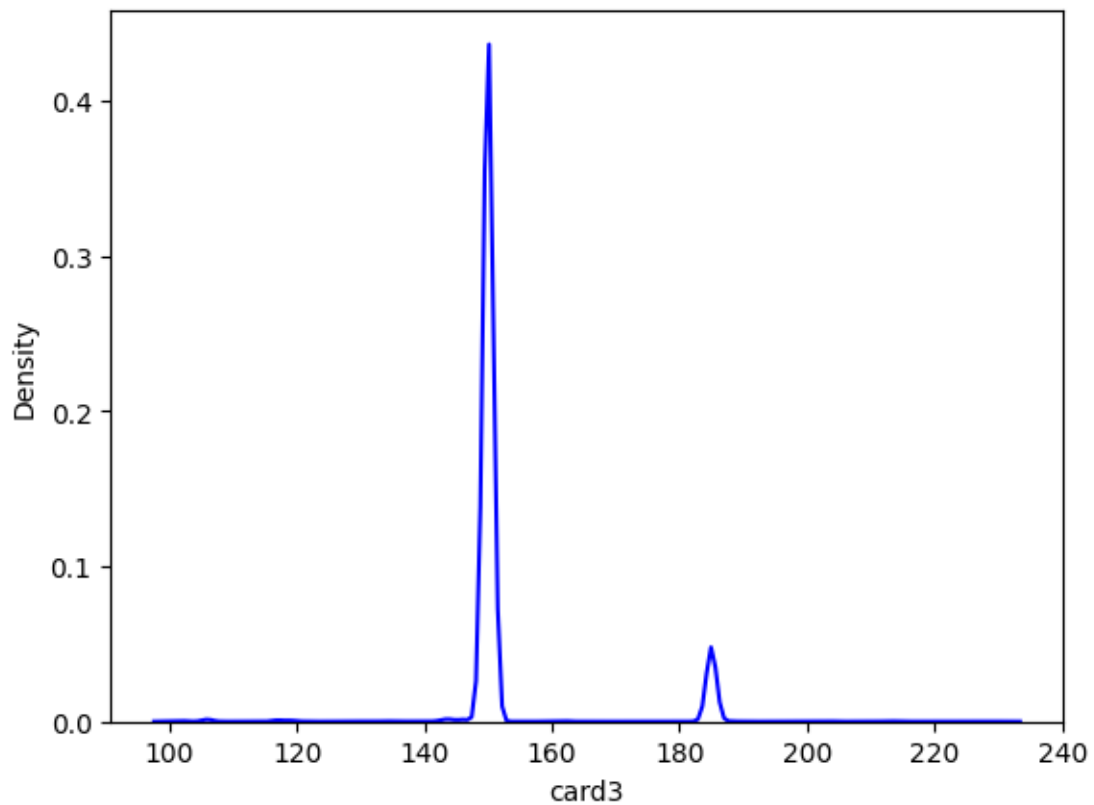
```
skewness  
card2    -0.2  
dtype: float64
```



3.5 Card3

```
[39]: sns.kdeplot(train['card3'], color = 'b')  
      print('skewness')  
      print(train[['card3']].skew())
```

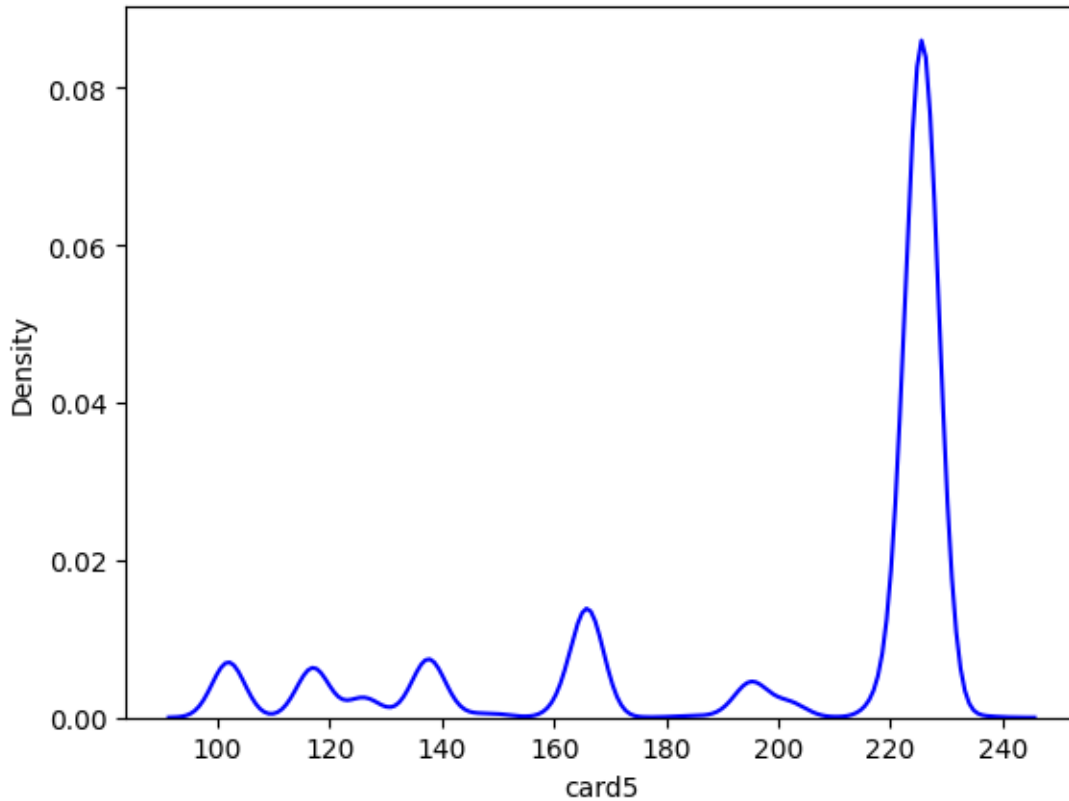
```
skewness  
card3    2.0  
dtype: float64
```



3.6 Card5

```
[40]: sns.kdeplot(train['card5'], color = 'b')  
      print('skewness')  
      print(train[['card5']].skew())
```

```
skewness  
card5    -1.2  
dtype: float64
```



4 Class Distribution

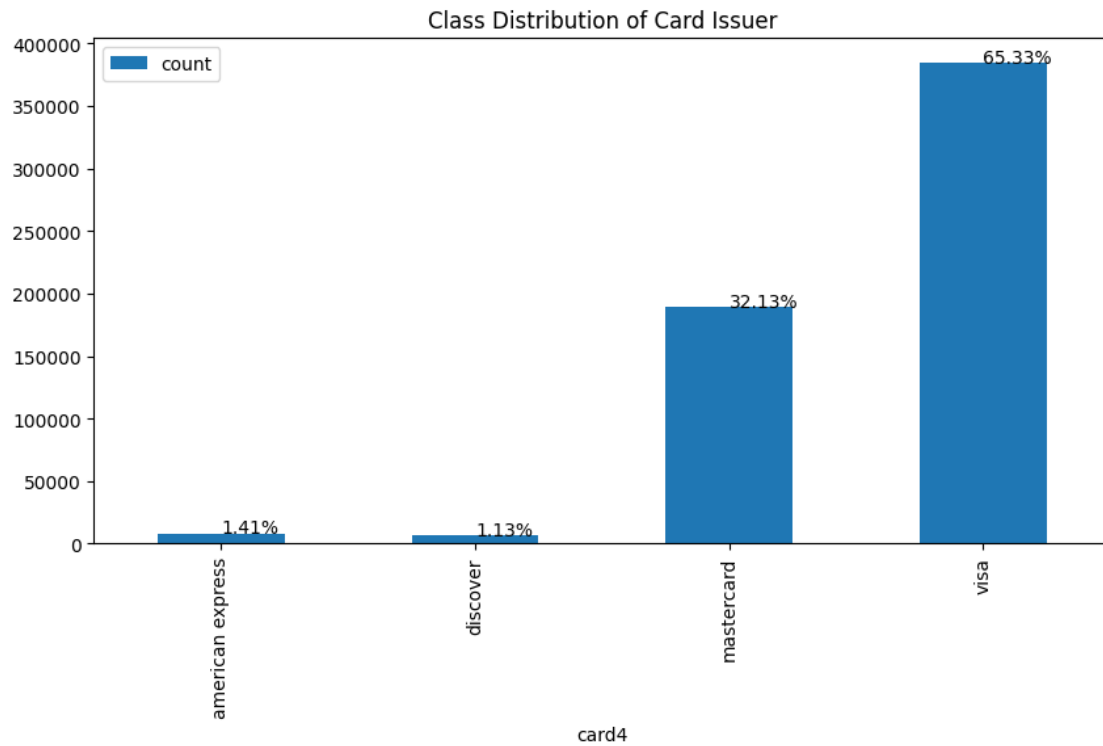
4.1 Card Issuer

```
[ ]: class_counts = pd.DataFrame(train.groupby('card4').size(),columns=['count']).
    →reset_index()
class_counts['percentage']=class_counts['count']/class_counts['count'].sum()*100
print(class_counts)
ax = class_counts.plot(x='card4',y='count',kind='bar',title='Class Distribution_
    →of Card Issuer',figsize=(10,5));
[ax.text(i, class_counts.loc[i]['count'],
    '{:.2f}%'.format(class_counts.loc[i].percentage)) for i in class_counts.
    →index];

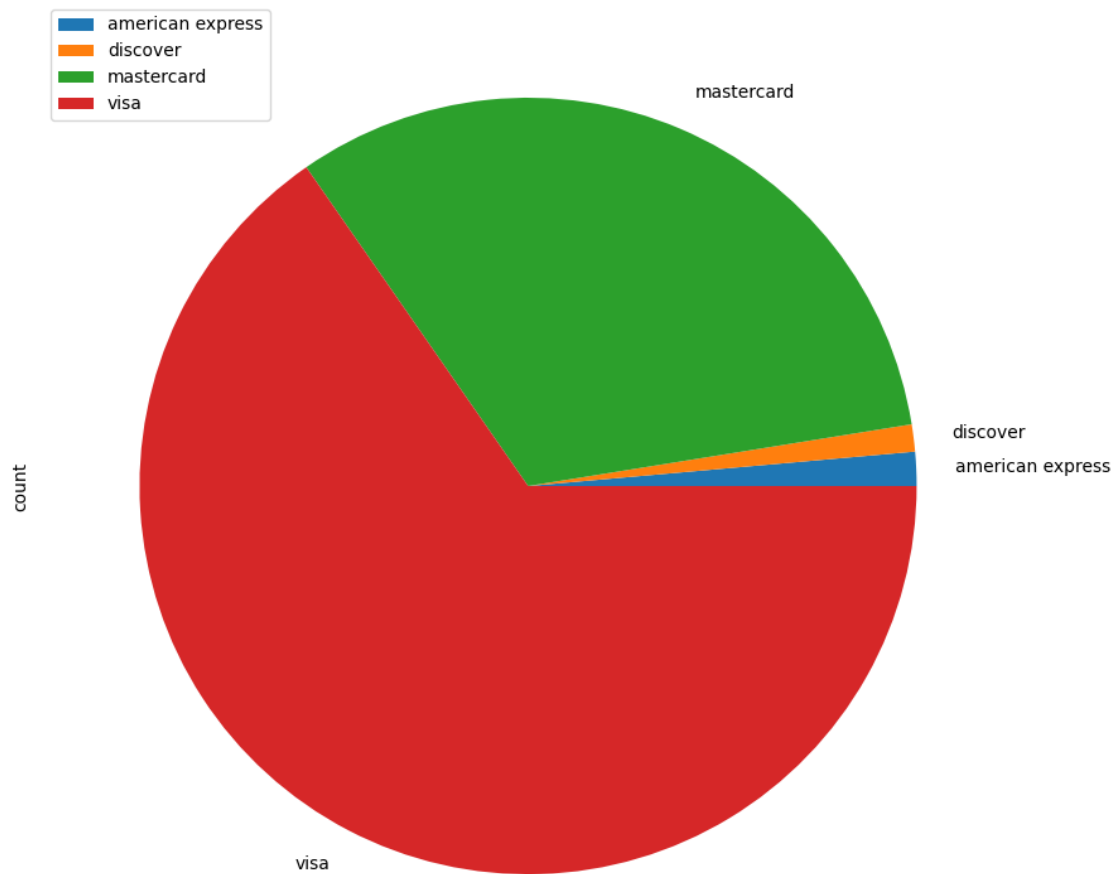
class_counts = pd.DataFrame(train.groupby('card4').size(),columns=['count']).
    →reset_index()
#class_counts = class_counts[class_counts['count']>=10].reset_index()
class_counts['percentage']=class_counts['count']/class_counts['count'].sum()*100
ax = class_counts.plot(x='card4',y='count',kind='pie',title='Class Distribution_
    →of Card Issuer',figsize=(16,10),
```

```
subplots=True, labels=class_counts.card4);
```

	card4	count	percentage
0	american express	8328	1.414011
1	discover	6651	1.129273
2	mastercard	189217	32.127146
3	visa	384767	65.329571



Class Distribution of Card Issuer



```
[ ]: for i in class_counts.index:
    if (class_counts.loc[i].percentage/100)>=0.25:
        print(class_counts.loc[i]['card4'] + " is balanced")
    elif (class_counts.loc[i].percentage/100)>=0.1:
        print(class_counts.loc[i]['card4'] + " is Slightly Imbalance")
    else:
        print(class_counts.loc[i]['card4'] + " is Severely Imbalance")
```

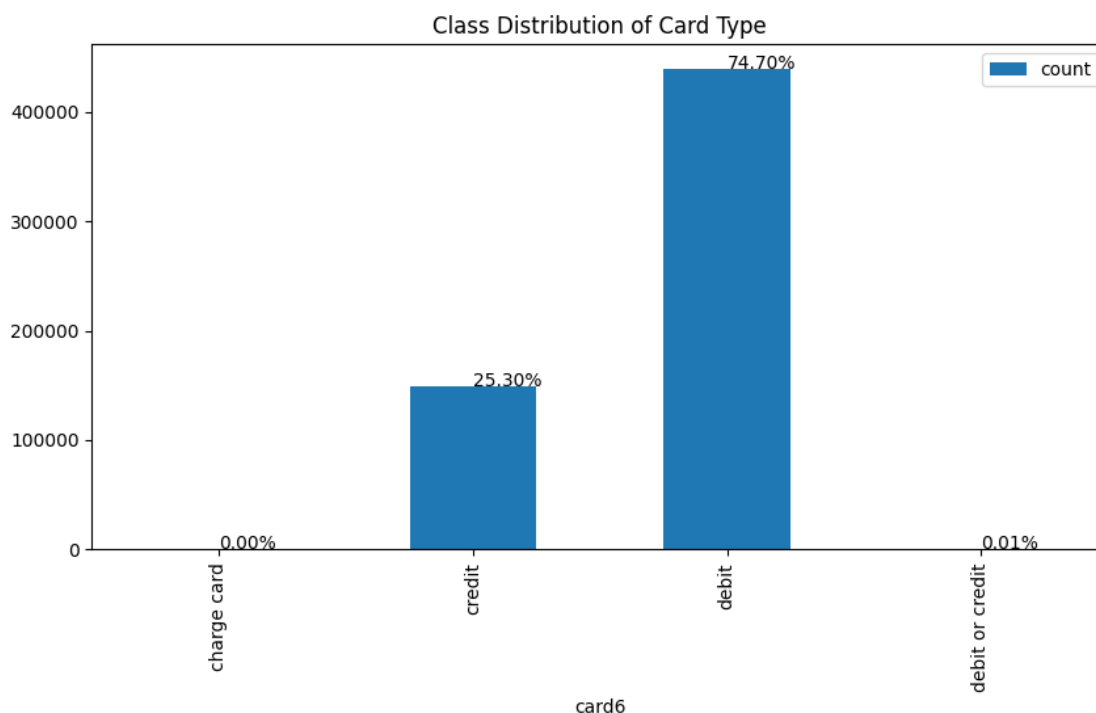
```
american express is Severely Imbalance
discover is Severely Imbalance
mastercard is balanced
visa is balanced
```

4.2 Card Type

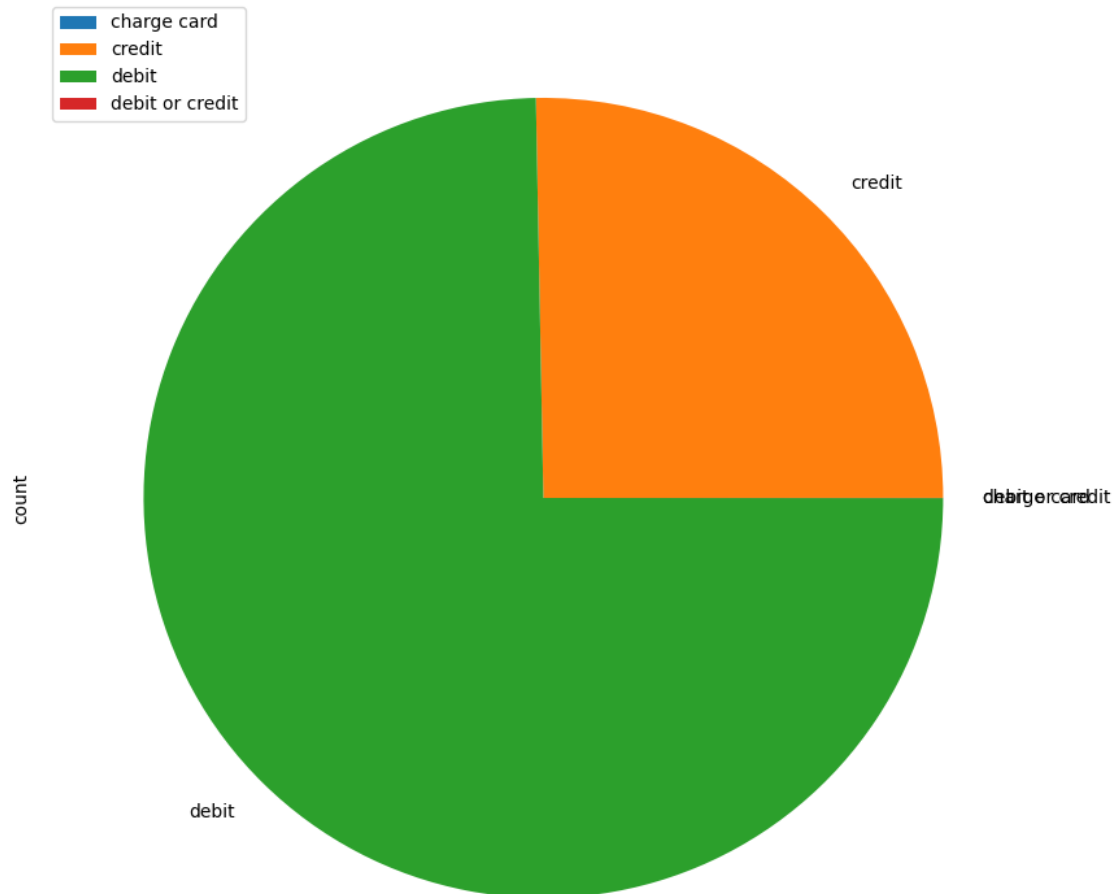
```
[ ]: class_counts = pd.DataFrame(train.groupby('card6').size(),columns=['count']).
    →reset_index()
class_counts['percentage']=class_counts['count']/class_counts['count'].sum()*100
print(class_counts)
ax = class_counts.plot(x='card6',y='count',kind='bar',title='Class Distribution_
    →of Card Type',figsize=(10,5));
[ax.text(i, class_counts.loc[i]['count'],
    '{:.2f}%'.format(class_counts.loc[i].percentage)) for i in class_counts.
    →index];

class_counts = pd.DataFrame(train.groupby('card6').size(),columns=['count']).
    →reset_index()
#class_counts = class_counts[class_counts['count']>=10].reset_index()
class_counts['percentage']=class_counts['count']/class_counts['count'].sum()*100
ax = class_counts.plot(x='card6',y='count',kind='pie',title='Class Distribution_
    →of Card Type',figsize=(16,10),
    subplots=True, labels=class_counts.card6);
```

	card6	count	percentage
0	charge card	15	0.002547
1	credit	148986	25.296068
2	debit	439938	74.696291
3	debit or credit	30	0.005094



Class Distribution of Card Type



```
[ ]: for i in class_counts.index:
    if (class_counts.loc[i].percentage/100)>=0.25:
        print(class_counts.loc[i]['card6'] + " is balanced")
    elif (class_counts.loc[i].percentage/100)>=0.1:
        print(class_counts.loc[i]['card6'] + " is Slightly Imbalance")
    else:
        print(class_counts.loc[i]['card6'] + " is Severely Imbalance")
```

```
charge card is Severely Imbalance
credit is balanced
```

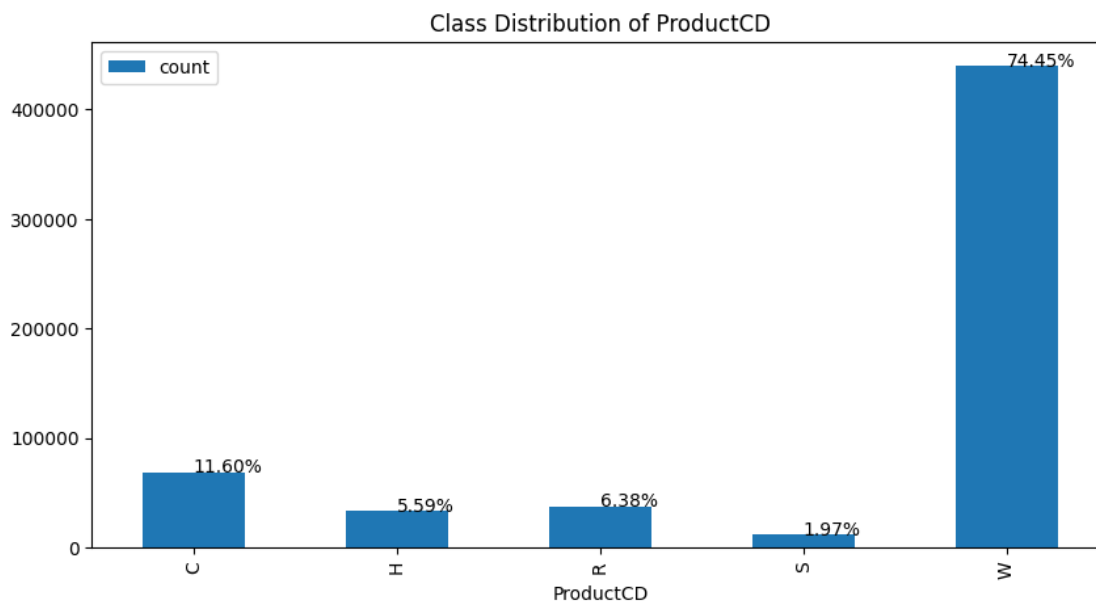
debit is balanced
debit or credit is Severely Imbalance

4.3 Product CD

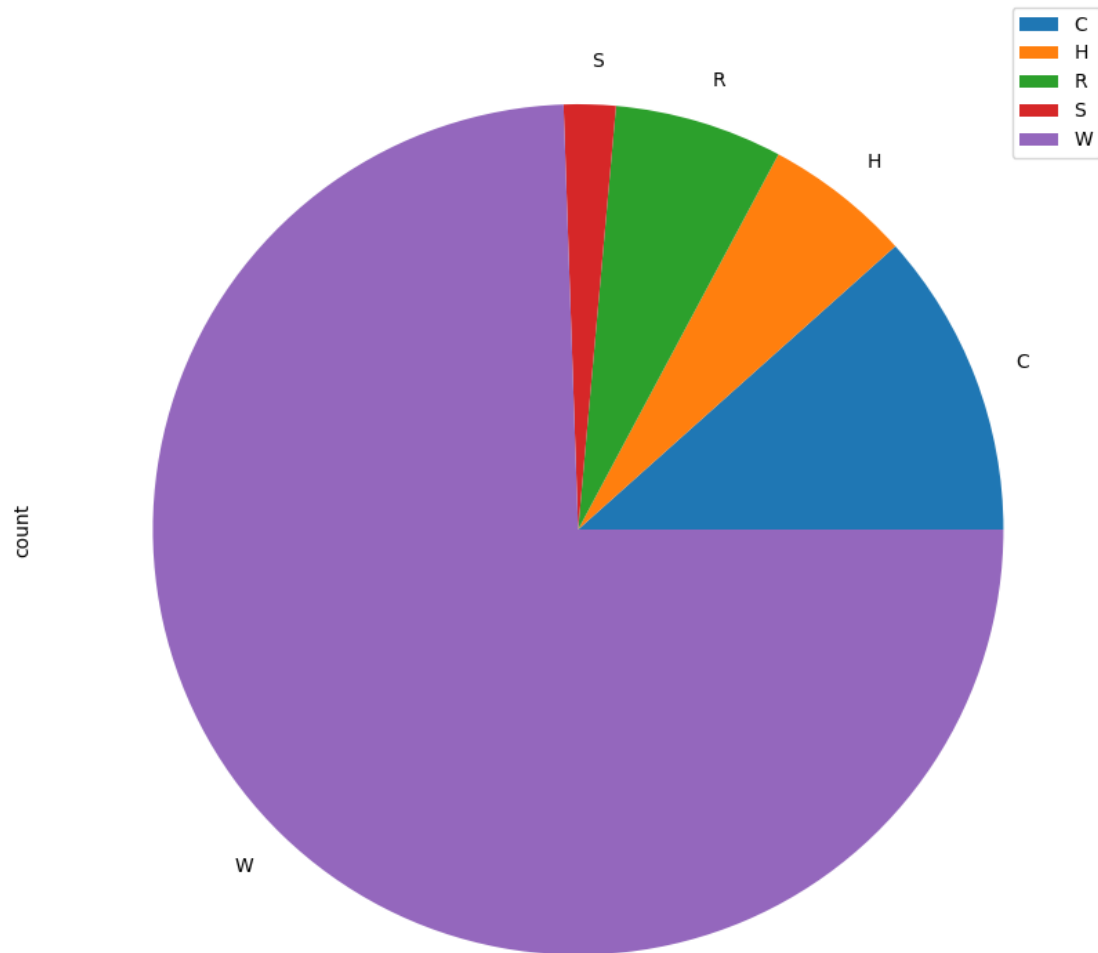
```
[ ]: class_counts = pd.DataFrame(train.groupby('ProductCD').size(),columns=['count']).
    →reset_index()
class_counts['percentage']=class_counts['count']/class_counts['count'].sum()*100
print(class_counts)
ax = class_counts.plot(x='ProductCD',y='count',kind='bar',title='Class_
    →Distribution of ProductCD',figsize=(10,5));
[ax.text(i, class_counts.loc[i]['count'],
    '{:.2f}%'.format(class_counts.loc[i].percentage)) for i in class_counts.
    →index];

class_counts = pd.DataFrame(train.groupby('ProductCD').size(),columns=['count']).
    →reset_index()
#class_counts = class_counts[class_counts['count']>=10].reset_index()
class_counts['percentage']=class_counts['count']/class_counts['count'].sum()*100
ax = class_counts.plot(x='ProductCD',y='count',kind='pie',title='Class_
    →Distribution of ProductCD',figsize=(16,10),
    subplots=True, labels=class_counts.ProductCD);
```

	ProductCD	count	percentage
0	C	68519	11.602770
1	H	33024	5.592170
2	R	37699	6.383818
3	S	11628	1.969045
4	W	439670	74.452196



Class Distribution of ProductCD



```
[ ]: for i in class_counts.index:
    if (class_counts.loc[i].percentage/100)>=0.25:
        print(class_counts.loc[i]['ProductCD'] + " is balanced")
    elif (class_counts.loc[i].percentage/100)>=0.1:
        print(class_counts.loc[i]['ProductCD'] + " is Slightly Imbalance")
    else:
        print(class_counts.loc[i]['ProductCD'] + " is Severely Imbalance")
```

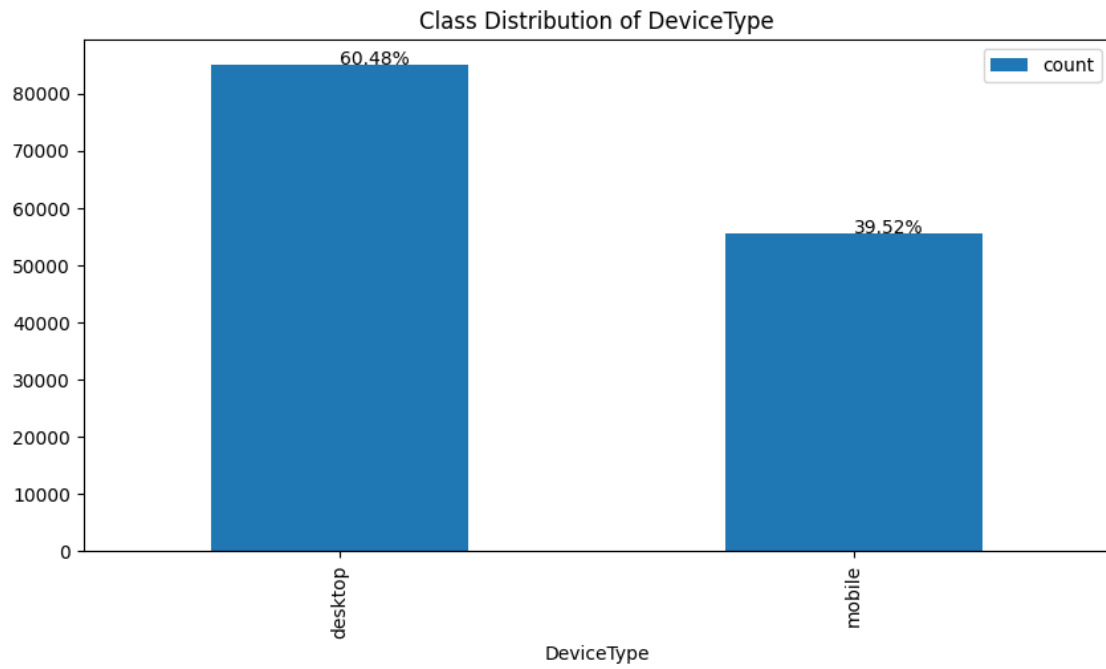
C is Slightly Imbalance
H is Severely Imbalance
R is Severely Imbalance
S is Severely Imbalance
W is balanced

4.4 Device Type

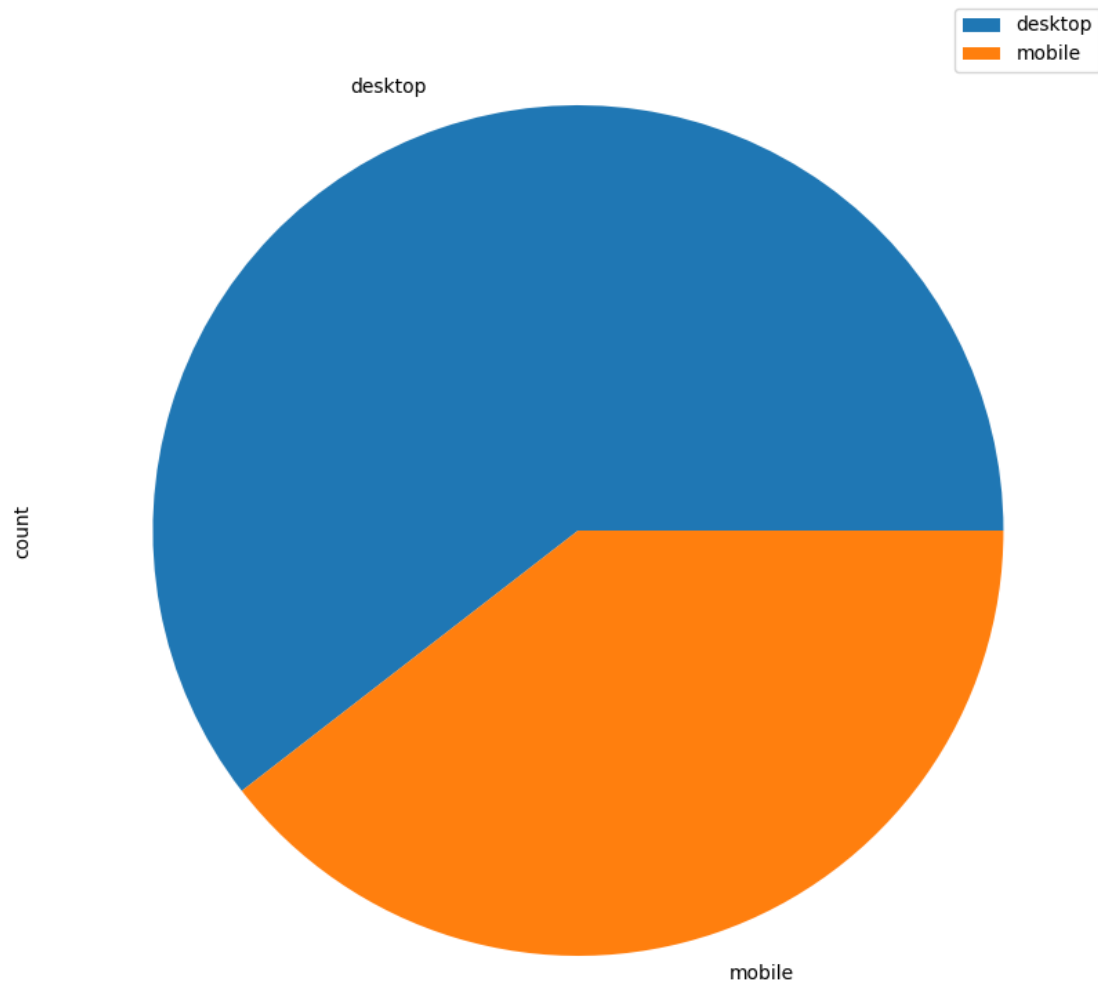
```
[ ]: class_counts = pd.DataFrame(train.groupby('DeviceType').
    ↳size(),columns=['count']).reset_index()
class_counts['percentage']=class_counts['count']/class_counts['count'].sum()*100
print(class_counts)
ax = class_counts.plot(x='DeviceType',y='count',kind='bar',title='Class_
    ↳Distribution of DeviceType',figsize=(10,5));
[ax.text(i, class_counts.loc[i]['count'],
    ' {:.2f}%'.format(class_counts.loc[i].percentage)) for i in class_counts.
    ↳index];

class_counts = pd.DataFrame(train.groupby('DeviceType').
    ↳size(),columns=['count']).reset_index()
#class_counts = class_counts[class_counts['count']>=10].reset_index()
class_counts['percentage']=class_counts['count']/class_counts['count'].sum()*100
ax = class_counts.plot(x='DeviceType',y='count',kind='pie',title='Class_
    ↳Distribution of DeviceType',figsize=(16,10),
    subplots=True, labels=class_counts.DeviceType);
```

	DeviceType	count	percentage
0	desktop	85165	60.48221
1	mobile	55645	39.51779



Class Distribution of DeviceType



```
[ ]: for i in class_counts.index:
    if (class_counts.loc[i].percentage/100)>=0.25:
        print(class_counts.loc[i]['DeviceType'] + " is balanced")
    elif (class_counts.loc[i].percentage/100)>=0.1:
        print(class_counts.loc[i]['DeviceType'] + " is Slightly Imbalance")
    else:
        print(class_counts.loc[i]['DeviceType'] + " is Severely Imbalance")
```

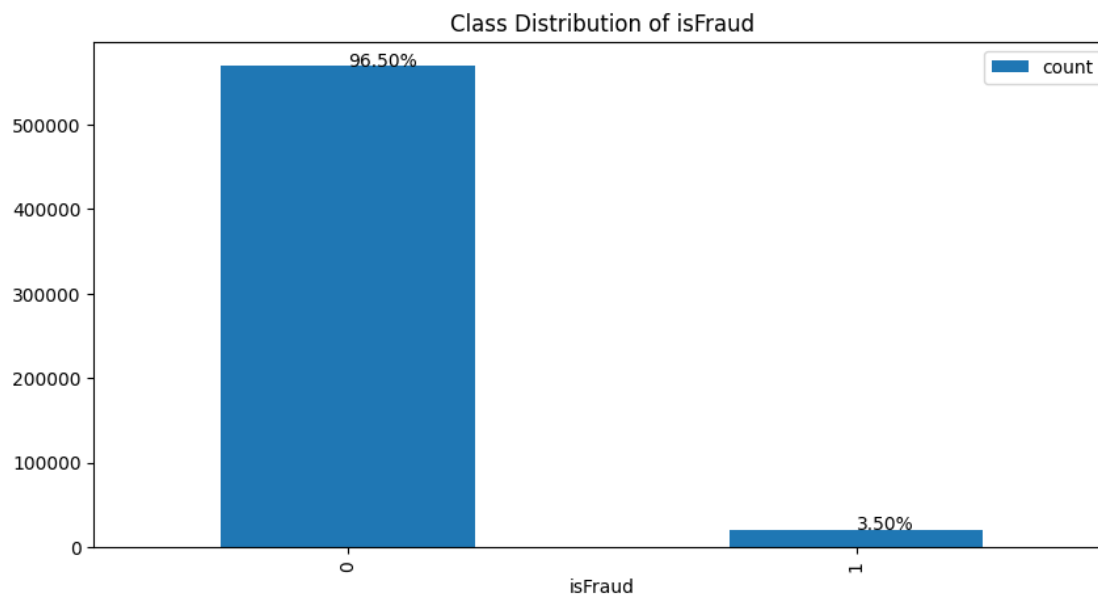
desktop is balanced
mobile is balanced

4.5 isFraud

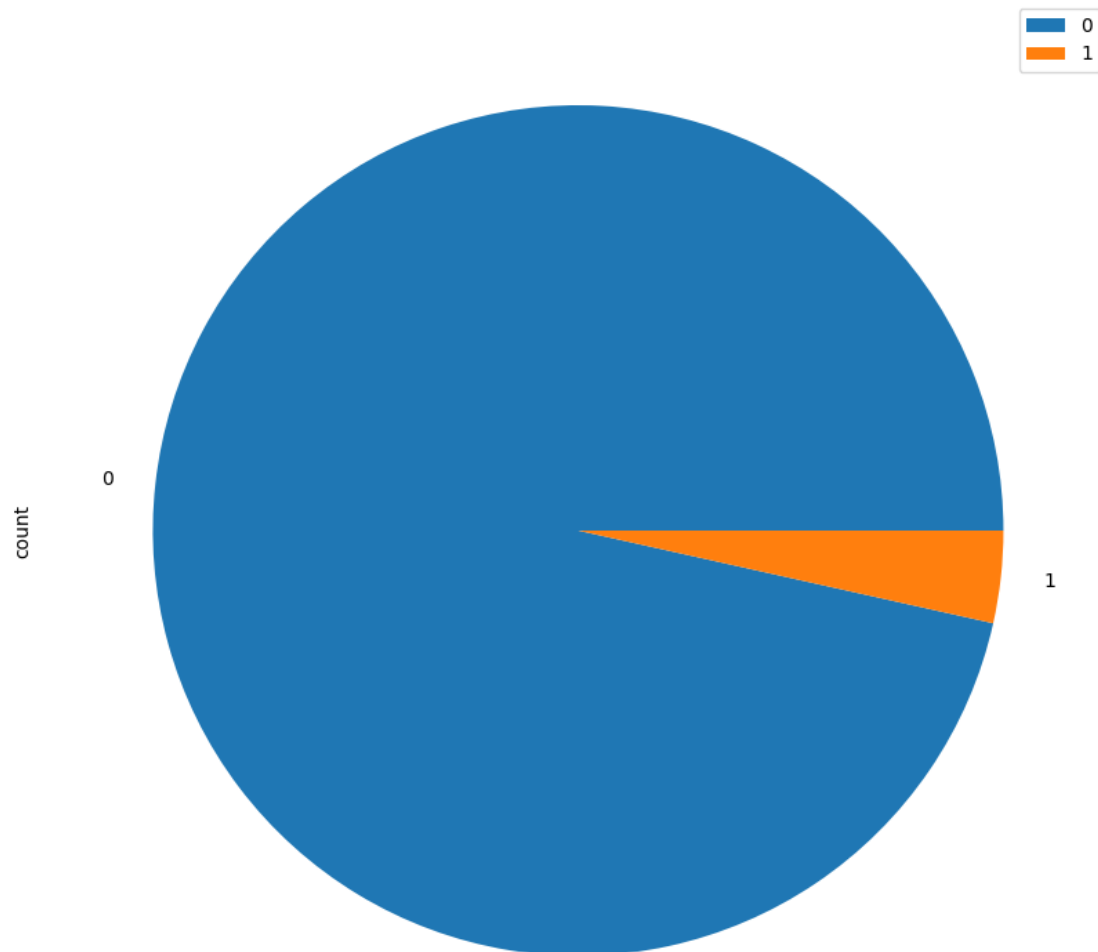
```
[ ]: class_counts = pd.DataFrame(train.groupby('isFraud').size(),columns=['count']).
    →reset_index()
class_counts['percentage']=class_counts['count']/class_counts['count'].sum()*100
print(class_counts)
ax = class_counts.plot(x='isFraud',y='count',kind='bar',title='Class_
    →Distribution of isFraud',figsize=(10,5));
[ax.text(i, class_counts.loc[i]['count'],
    '{:.2f}%'.format(class_counts.loc[i].percentage)) for i in class_counts.
    →index];

class_counts = pd.DataFrame(train.groupby('isFraud').size(),columns=['count']).
    →reset_index()
#class_counts = class_counts[class_counts['count']>=10].reset_index()
class_counts['percentage']=class_counts['count']/class_counts['count'].sum()*100
ax = class_counts.plot(x='isFraud',y='count',kind='pie',title='Class_
    →Distribution of isFraud',figsize=(16,10),
    subplots=True, labels=class_counts.isFraud);
```

	isFraud	count	percentage
0	0	569877	96.500999
1	1	20663	3.499001



Class Distribution of isFraud



```
[ ]: for i in class_counts.index:
    if (class_counts.loc[i].percentage/100)>=0.25:
        print(class_counts.loc[i]['isFraud'].astype(str) + " is balanced")
    elif (class_counts.loc[i].percentage/100)>=0.1:
        print(class_counts.loc[i]['isFraud'].astype(str) + " is Slightly␣
→ Imbalance")
    else:
        print(class_counts.loc[i]['isFraud'].astype(str) + " is Severely␣
→ Imbalance")
```


0.0 is balanced
1.0 is Severely Imbalance

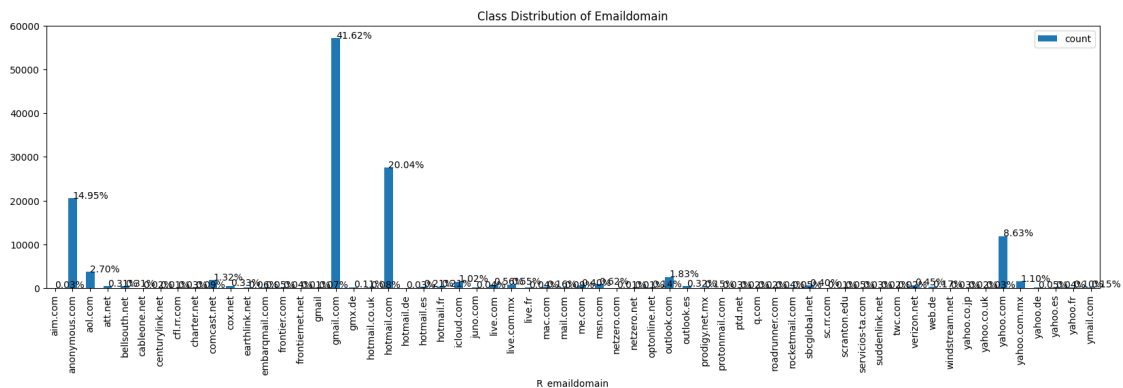
4.6 Email Domain

```
[ ]: class_counts = pd.DataFrame(train.groupby('R_emaildomain').
    ↳size(),columns=['count']).reset_index()
class_counts['percentage']=class_counts['count']/class_counts['count'].sum()*100
print(class_counts)
ax = class_counts.plot(x='R_emaildomain',y='count',kind='bar',title='Class_
    ↳Distribution of Emaildomain',figsize=(20,5));
[ax.text(i, class_counts.loc[i]['count'],
    ' {:.2f}%'.format(class_counts.loc[i].percentage)) for i in class_counts.
    ↳index];

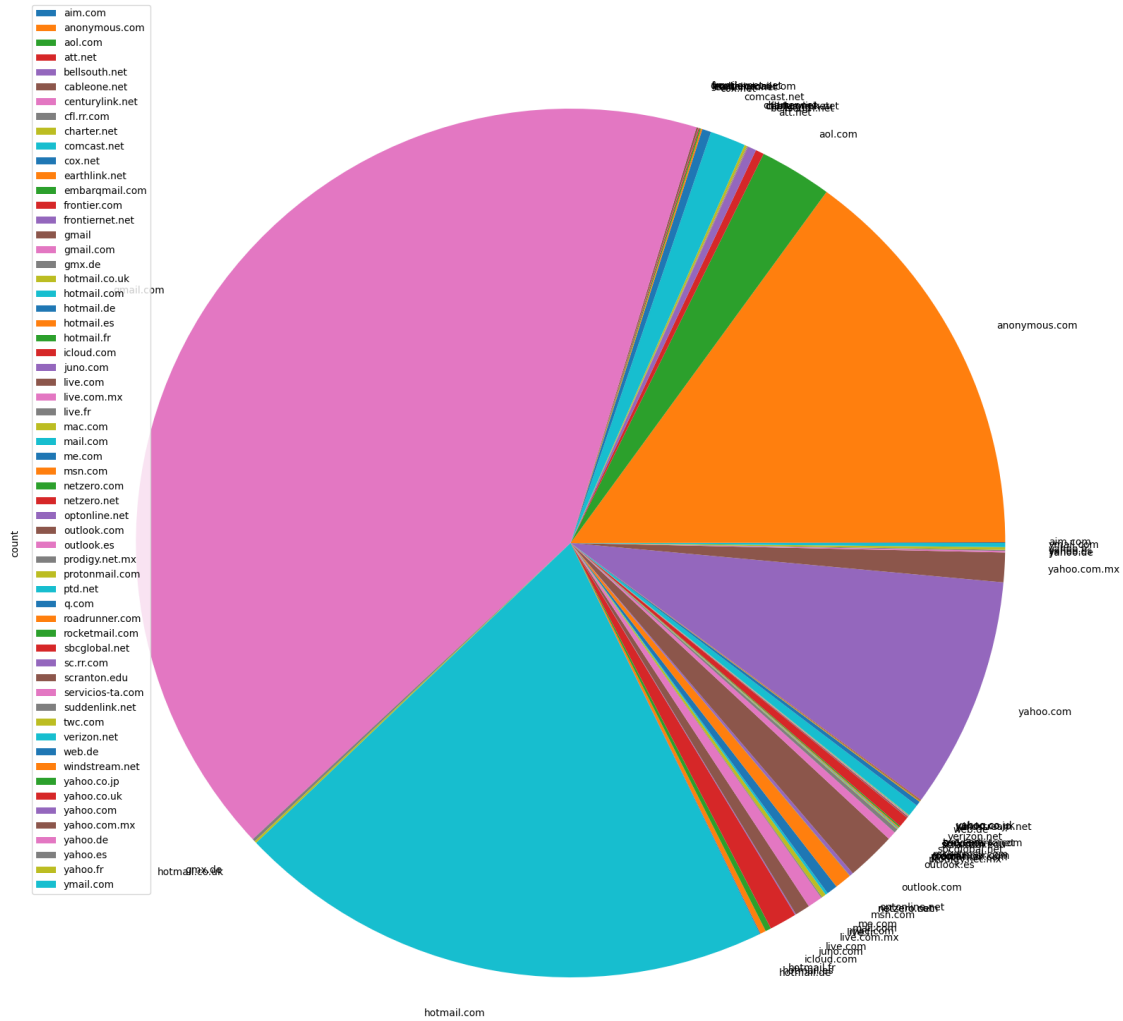
class_counts = pd.DataFrame(train.groupby('R_emaildomain').
    ↳size(),columns=['count']).reset_index()
#class_counts = class_counts[class_counts['count']>=10].reset_index()
class_counts['percentage']=class_counts['count']/class_counts['count'].sum()*100
ax = class_counts.plot(x='R_emaildomain',y='count',kind='pie',title='Class_
    ↳Distribution of Emaildomain',figsize=(26,20),
    subplots=True,labels=class_counts.R_emaildomain);
```

	R_emaildomain	count	percentage
0	aim.com	36	0.026222
1	anonymous.com	20529	14.952910
2	aol.com	3701	2.695734
3	att.net	430	0.313203
4	bellsouth.net	422	0.307376
5	cableone.net	27	0.019666
6	centurylink.net	12	0.008741
7	cfl.rr.com	37	0.026950
8	charter.net	127	0.092504
9	comcast.net	1812	1.319824
10	cox.net	459	0.334326
11	earthlink.net	79	0.057542
12	embarqmail.com	68	0.049530
13	frontier.com	52	0.037876
14	frontiernet.net	14	0.010197
15	gmail	95	0.069196
16	gmail.com	57147	41.624724
17	gmxd.de	147	0.107072
18	hotmail.co.uk	105	0.076480
19	hotmail.com	27509	20.037002
20	hotmail.de	42	0.030592
21	hotmail.es	292	0.212687
22	hotmail.fr	293	0.213415
23	icloud.com	1398	1.018275

24	juno.com	53	0.038604
25	live.com	762	0.555025
26	live.com.mx	754	0.549198
27	live.fr	55	0.040061
28	mac.com	218	0.158787
29	mail.com	122	0.088862
30	me.com	556	0.404979
31	msn.com	852	0.620580
32	netzero.com	14	0.010197
33	netzero.net	9	0.006555
34	optonline.net	187	0.136207
35	outlook.com	2507	1.826048
36	outlook.es	433	0.315388
37	prodigy.net.mx	207	0.150775
38	protonmail.com	41	0.029864
39	ptd.net	27	0.019666
40	q.com	25	0.018209
41	roadrunner.com	53	0.038604
42	rocketmail.com	69	0.050258
43	sbcglobal.net	552	0.402066
44	sc.rr.com	8	0.005827
45	scranton.edu	63	0.045888
46	servicios-ta.com	35	0.025493
47	suddenlink.net	25	0.018209
48	twc.com	29	0.021123
49	verizon.net	620	0.451596
50	web.de	237	0.172626
51	windstream.net	47	0.034234
52	yahoo.co.jp	33	0.024037
53	yahoo.co.uk	39	0.028407
54	yahoo.com	11842	8.625474
55	yahoo.com.mx	1508	1.098397
56	yahoo.de	75	0.054628
57	yahoo.es	57	0.041518
58	yahoo.fr	137	0.099788
59	ymail.com	207	0.150775



Class Distribution of Emaildomain



```
[ ]: for i in class_counts.index:
    if (class_counts.loc[i].percentage/100)>=0.25:
        print(class_counts.loc[i]['R_emaildomain'] + " is balanced")
    elif (class_counts.loc[i].percentage/100)>=0.1:
        print(class_counts.loc[i]['R_emaildomain'] + " is Slightly Imbalance")
    else:
        print(class_counts.loc[i]['R_emaildomain'] + " is Severely Imbalance")
```

aim.com is Severely Imbalance
anonymous.com is Slightly Imbalance
aol.com is Severely Imbalance
att.net is Severely Imbalance
bellsouth.net is Severely Imbalance
cableone.net is Severely Imbalance
centurylink.net is Severely Imbalance
cfl.rr.com is Severely Imbalance
charter.net is Severely Imbalance
comcast.net is Severely Imbalance
cox.net is Severely Imbalance
earthlink.net is Severely Imbalance
embarqmail.com is Severely Imbalance
frontier.com is Severely Imbalance
frontiernet.net is Severely Imbalance
gmail is Severely Imbalance
gmail.com is balanced
gmx.de is Severely Imbalance
hotmail.co.uk is Severely Imbalance
hotmail.com is Slightly Imbalance
hotmail.de is Severely Imbalance
hotmail.es is Severely Imbalance
hotmail.fr is Severely Imbalance
icloud.com is Severely Imbalance
juno.com is Severely Imbalance
live.com is Severely Imbalance
live.com.mx is Severely Imbalance
live.fr is Severely Imbalance
mac.com is Severely Imbalance
mail.com is Severely Imbalance
me.com is Severely Imbalance
msn.com is Severely Imbalance
netzero.com is Severely Imbalance
netzero.net is Severely Imbalance
optonline.net is Severely Imbalance
outlook.com is Severely Imbalance
outlook.es is Severely Imbalance
prodigy.net.mx is Severely Imbalance
protonmail.com is Severely Imbalance
ptd.net is Severely Imbalance
q.com is Severely Imbalance
roadrunner.com is Severely Imbalance
rocketmail.com is Severely Imbalance
sbcglobal.net is Severely Imbalance
sc.rr.com is Severely Imbalance
scranton.edu is Severely Imbalance
servicios-ta.com is Severely Imbalance
suddenlink.net is Severely Imbalance

twc.com is Severely Imbalance
 verizon.net is Severely Imbalance
 web.de is Severely Imbalance
 windstream.net is Severely Imbalance
 yahoo.co.jp is Severely Imbalance
 yahoo.co.uk is Severely Imbalance
 yahoo.com is Severely Imbalance
 yahoo.com.mx is Severely Imbalance
 yahoo.de is Severely Imbalance
 yahoo.es is Severely Imbalance
 yahoo.fr is Severely Imbalance
 ymail.com is Severely Imbalance

5 Correlations between features

5.1 Correlation Coffeicents

The correlation coefficient is a statistical measure of the strength of a linear relationship between two variables

```
[3]: #finding coeffeicients between few numeric data columns
numeric_data = pd.DataFrame(train, columns = ['TransactionDT', 'TransactionAmt',
→ 'V33',
→ 'V12', 'C11', 'C2', 'V51', 'D4', 'D3',
→ 'D15'])
numeric_data.corr(method = 'pearson')
```

```
[3]:
```

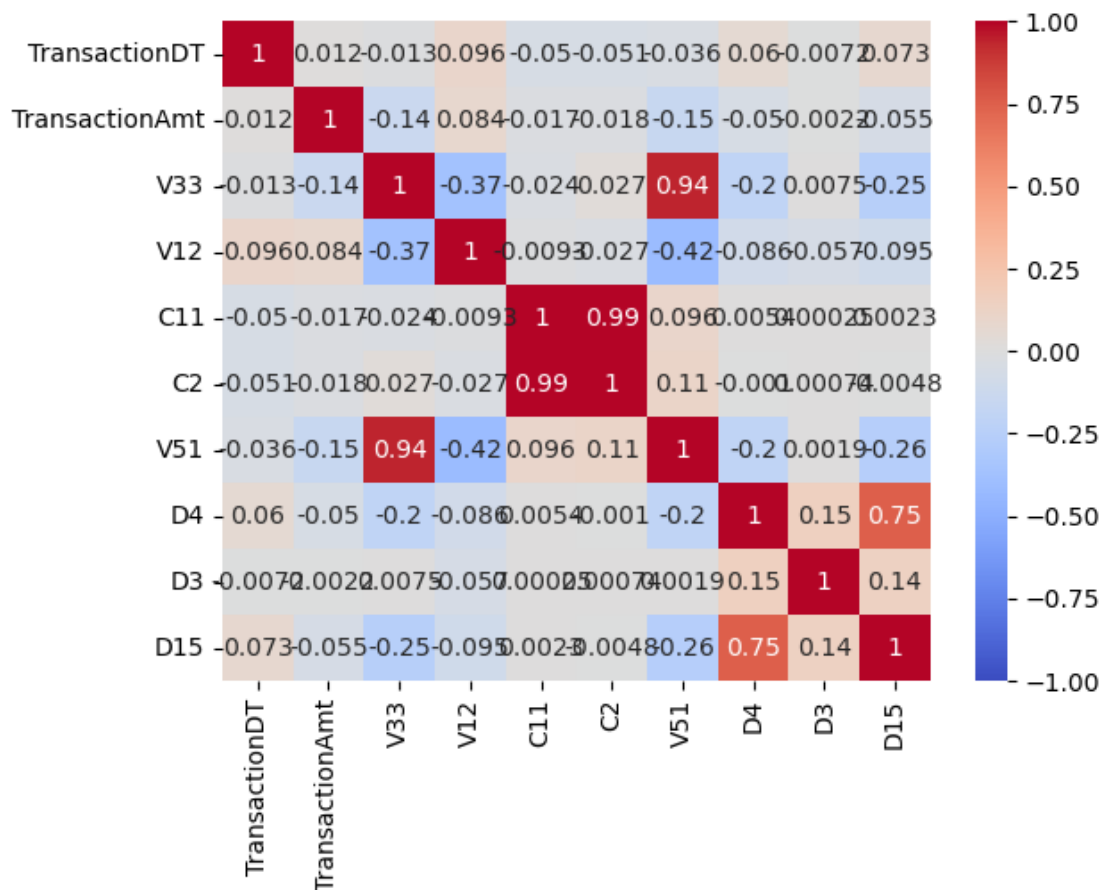
	TransactionDT	TransactionAmt	V33	V12	C11	\
TransactionDT	1.000000	0.011920	-0.013207	0.095749	-0.050181	
TransactionAmt	0.011920	1.000000	-0.135737	0.083628	-0.017313	
V33	-0.013207	-0.135737	1.000000	-0.369410	-0.024444	
V12	0.095749	0.083628	-0.369410	1.000000	-0.009336	
C11	-0.050181	-0.017313	-0.024444	-0.009336	1.000000	
C2	-0.051126	-0.018468	0.027042	-0.027223	0.993897	
V51	-0.035868	-0.149464	0.940497	-0.417592	0.095588	
D4	0.059797	-0.050345	-0.197543	-0.086478	0.005410	
D3	-0.007200	-0.002165	0.007453	-0.056569	0.000252	
D15	0.072791	-0.054733	-0.251661	-0.094844	0.002289	

	C2	V51	D4	D3	D15
TransactionDT	-0.051126	-0.035868	0.059797	-0.007200	0.072791
TransactionAmt	-0.018468	-0.149464	-0.050345	-0.002165	-0.054733
V33	0.027042	0.940497	-0.197543	0.007453	-0.251661
V12	-0.027223	-0.417592	-0.086478	-0.056569	-0.094844
C11	0.993897	0.095588	0.005410	0.000252	0.002289
C2	1.000000	0.110859	-0.001038	0.000743	-0.004773
V51	0.110859	1.000000	-0.197530	0.001905	-0.259066

D4	-0.001038	-0.197530	1.000000	0.149038	0.751546
D3	0.000743	0.001905	0.149038	1.000000	0.141511
D15	-0.004773	-0.259066	0.751546	0.141511	1.000000

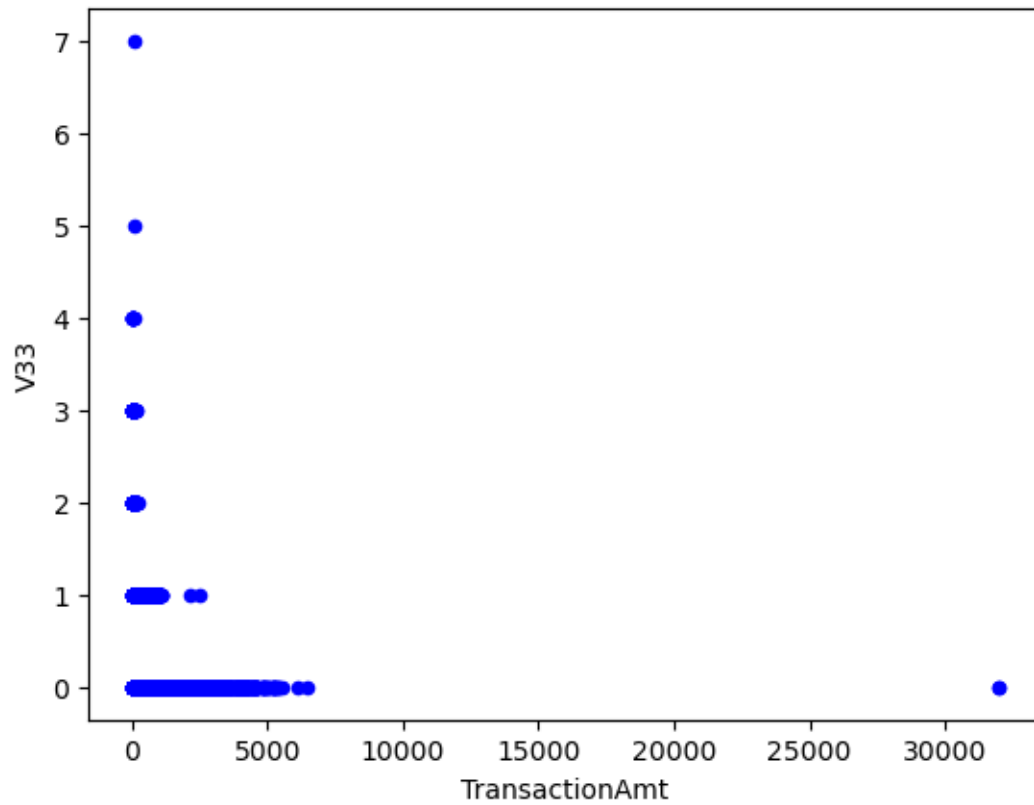
```
[4]: #A heat map is a data visualization technique that shows magnitude of a
      ↪phenomenon as color in two dimensions.
      sns.heatmap(numeric_data.corr(), vmin=-1, vmax=1, cmap = 'coolwarm', annot =
      ↪True )
```

```
[4]: <AxesSubplot:>
```



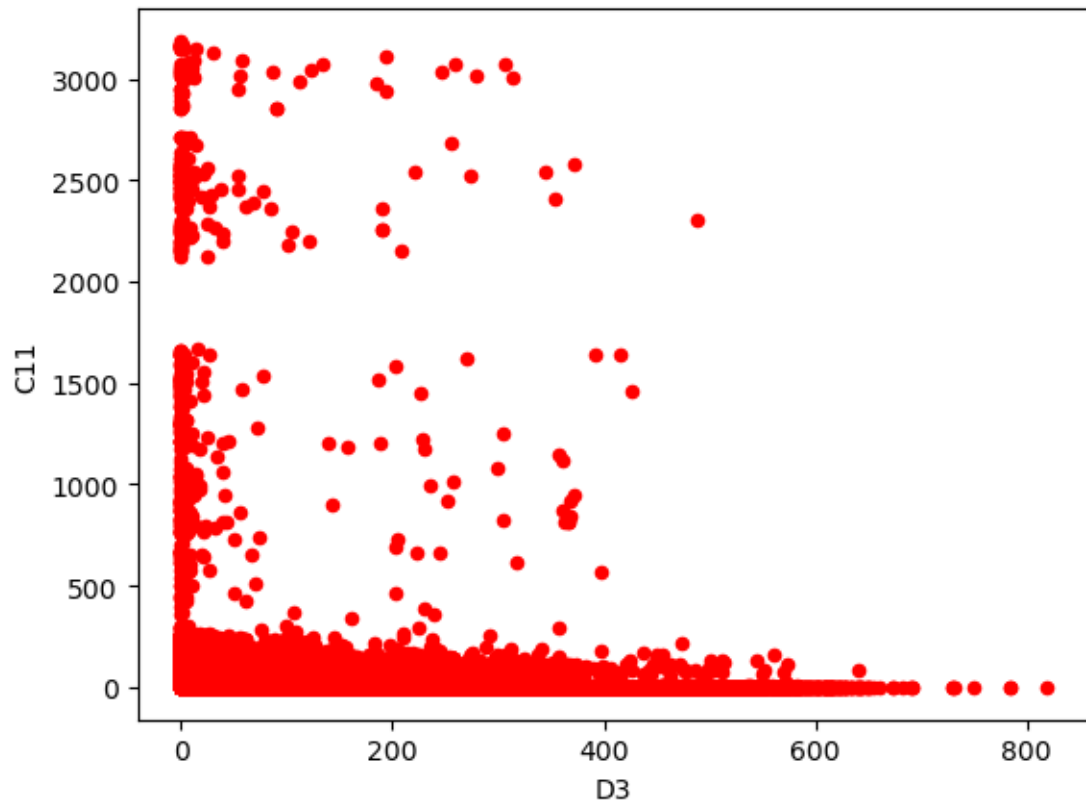
```
[5]: #pandas scatter plot between two variables
      numeric_data.plot(x='TransactionAmt',y='V33',kind='scatter',color = 'Blue')
```

```
[5]: <AxesSubplot:xlabel='TransactionAmt', ylabel='V33'>
```

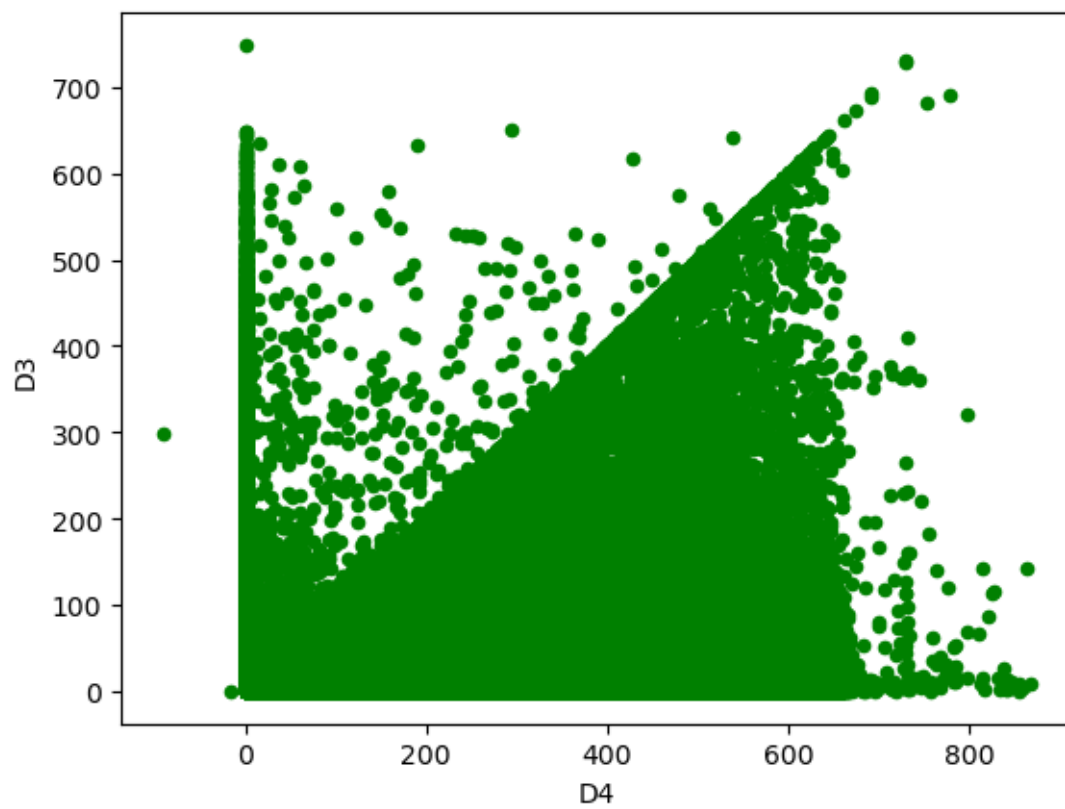


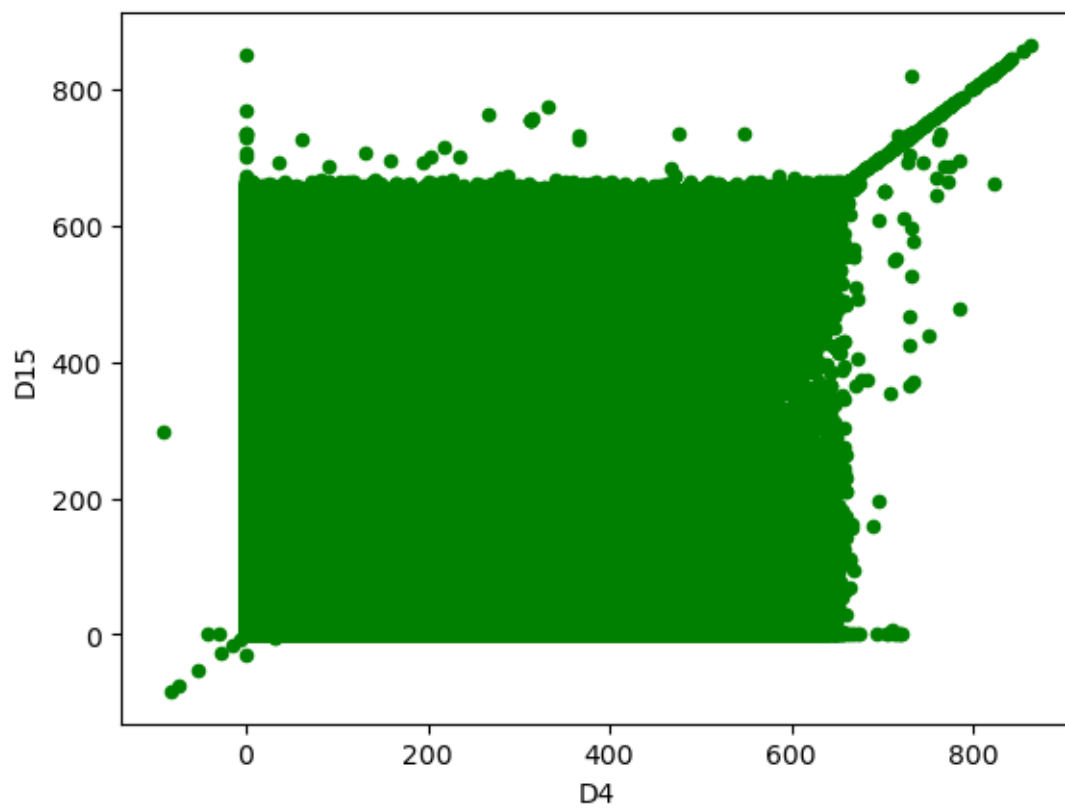
```
[6]: #pandas scatter plot between two variables  
numeric_data.plot(x='D3',y='C11',kind='scatter',color = 'Red')
```

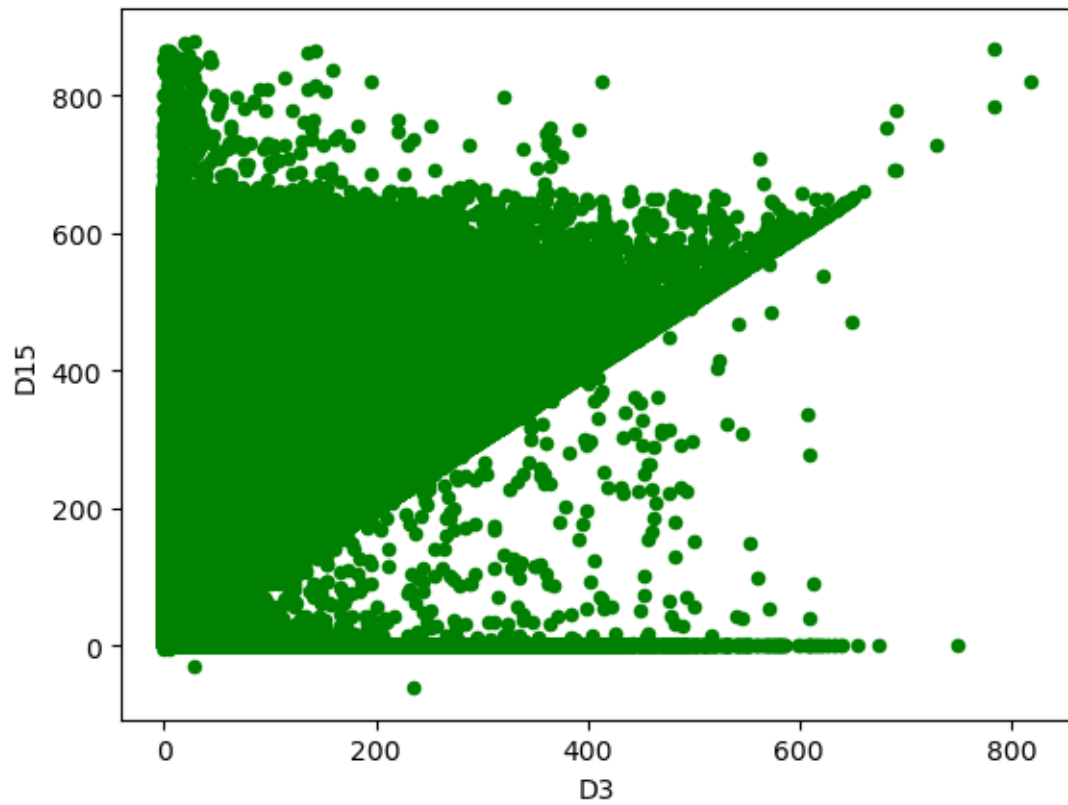
```
[6]: <AxesSubplot:xlabel='D3', ylabel='C11'>
```



```
[7]: cols = ['D4', 'D3', 'D15']
for i in range(len(cols)):
    for j in range(i, len(cols)):
        if(i != j):
            numeric_data.plot(x=cols[i], y=cols[j], kind='scatter',
                               ↪color='green')
```

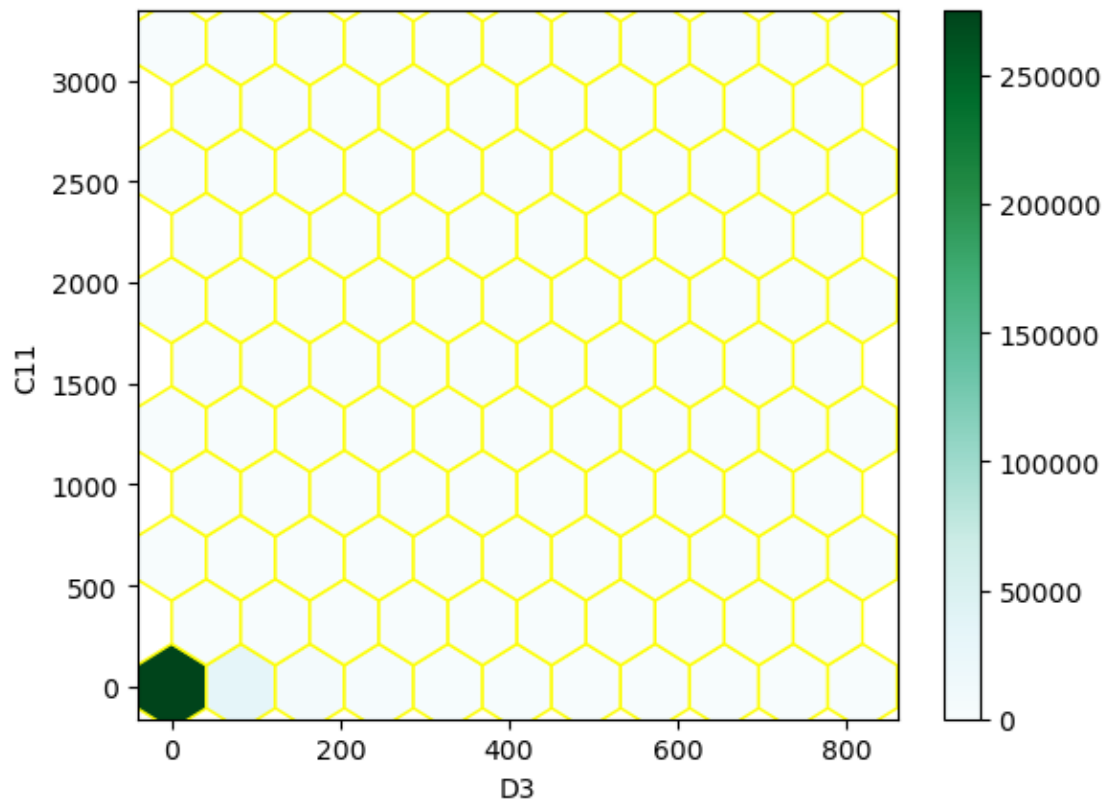







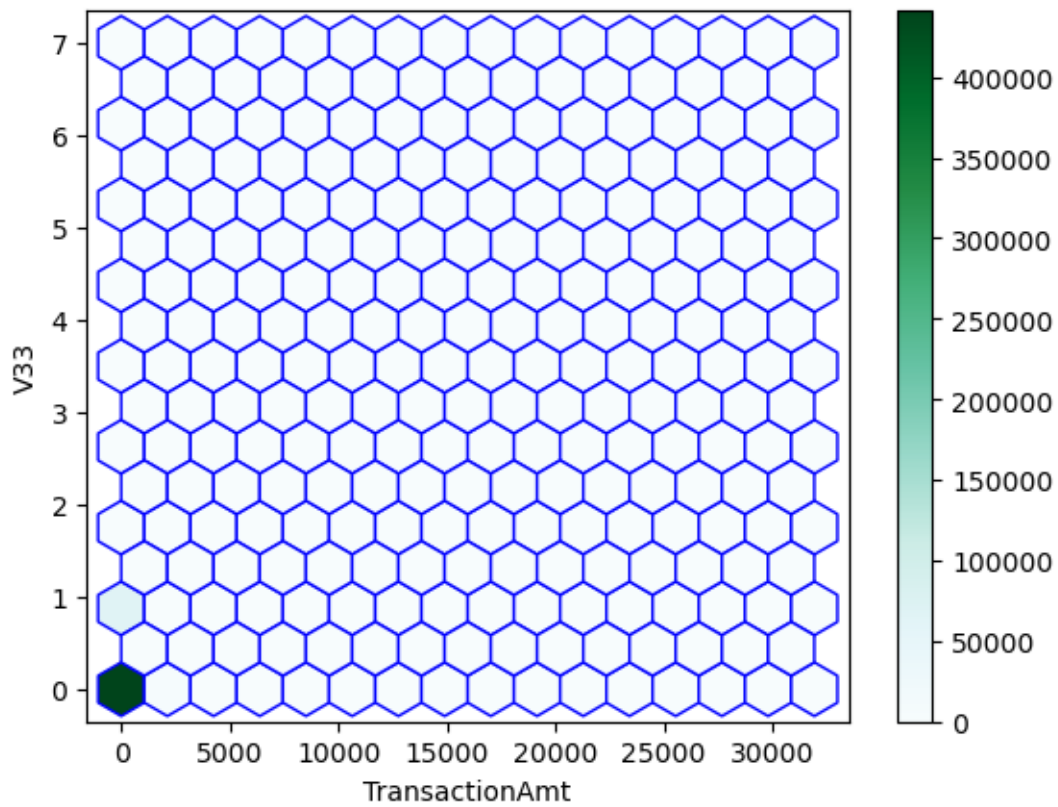
```
[8]: #pandas hexbin plot between two variables  
numeric_data.plot(x='D3',y='C11',kind='hexbin',color = 'Yellow', gridsize =10)
```

```
[8]: <AxesSubplot:xlabel='D3', ylabel='C11'>
```

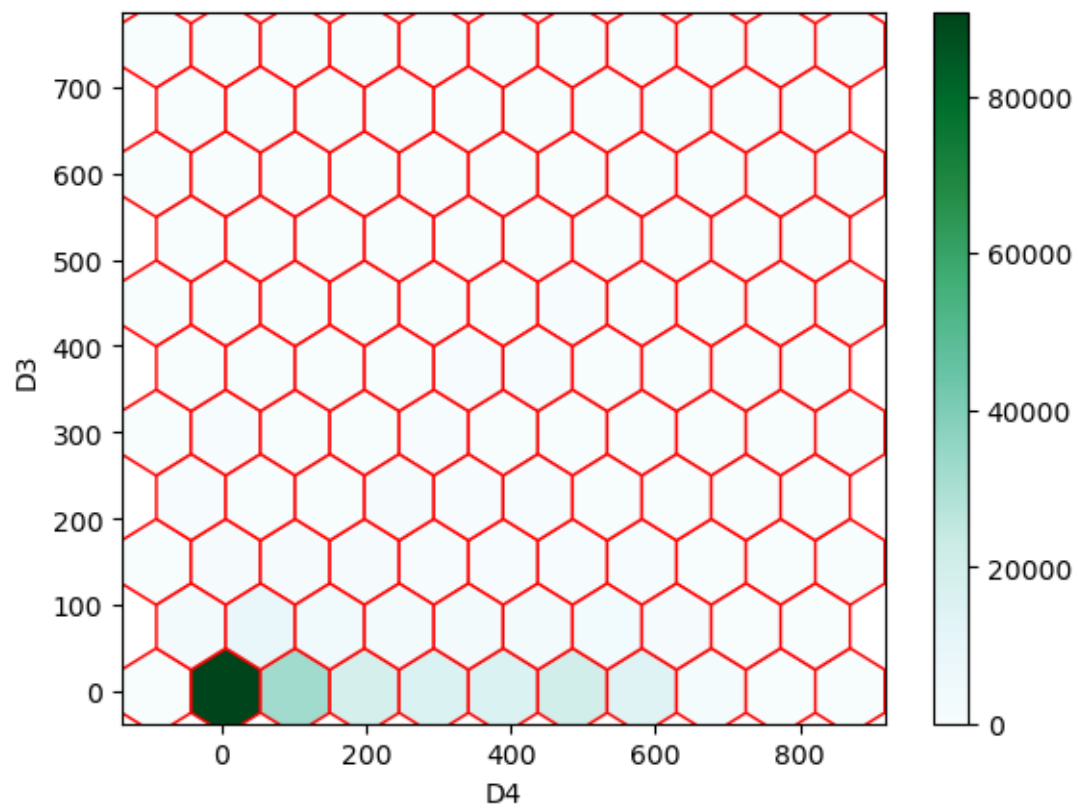


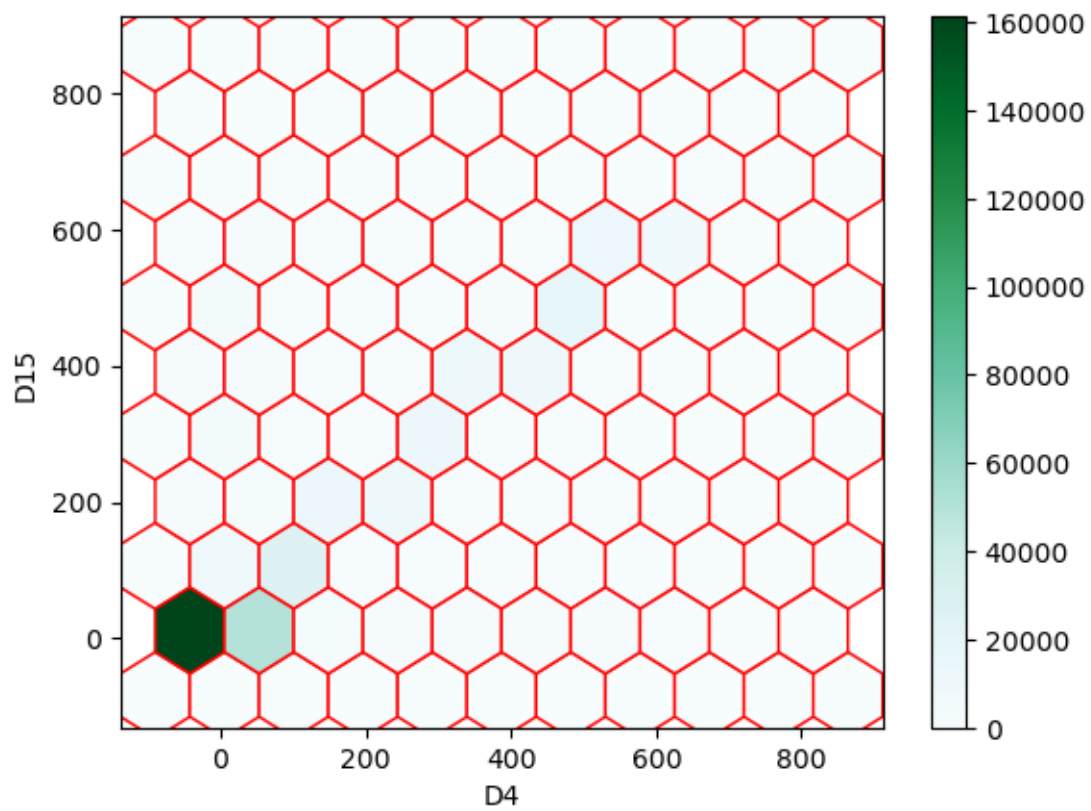
```
[9]: #pandas hexbin plot between two variables
numeric_data.plot(x='TransactionAmt',y='V33',kind='hexbin',color = 'Blue',
→gridsize=15)
```

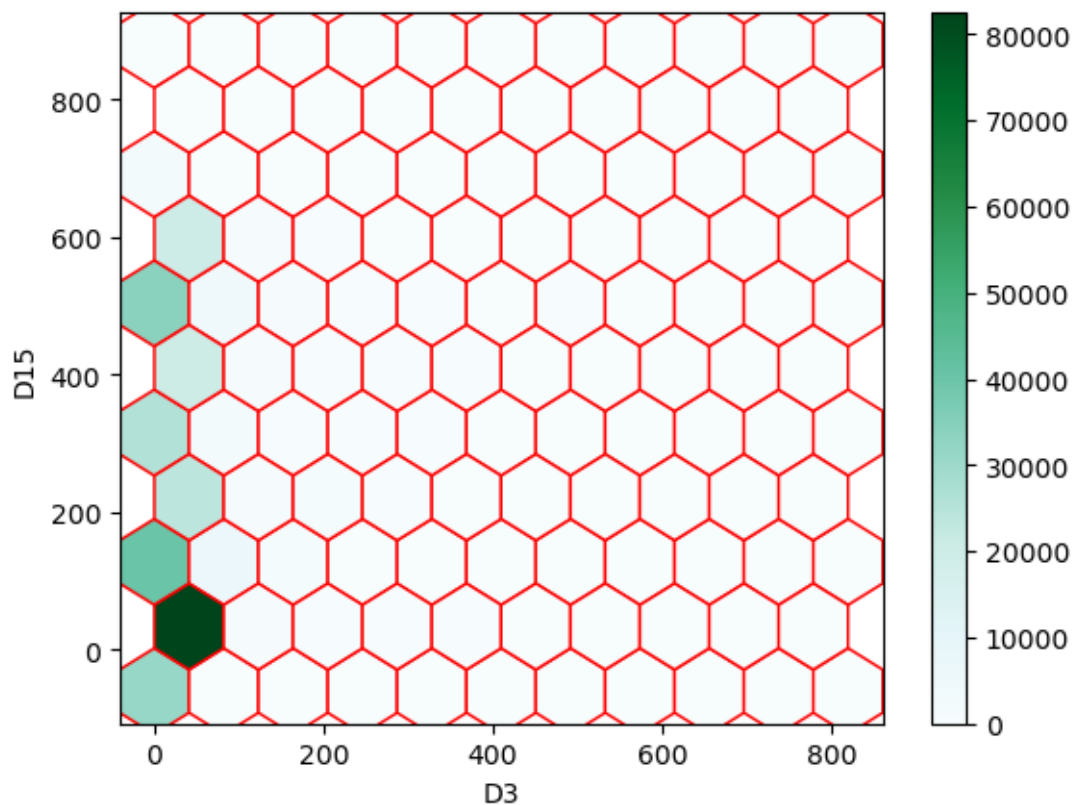
```
[9]: <AxesSubplot:xlabel='TransactionAmt', ylabel='V33'>
```



```
[10]: cols = ['D4', 'D3', 'D15']
      for i in range(len(cols)):
          for j in range(i, len(cols)):
              if(i != j):
                  numeric_data.plot(x=cols[i], y=cols[j], kind='hexbin', color='red',
→gridsize=10)
```







5.2 Chi-square test

The chi-square test is to test if two categorical features are independent, or not. The null hypothesis (H_0) is that two observed features are independent. The alternative hypothesis (H_a) is that the two observed features are dependent.

```
[11]: # the significant value that will be used is 0.05
      sig_value = 0.05
```

5.2.1 ProductCD and Card4

H_0 : ProductCD and Card4 are independent

H_a : ProductCD and Card4 are dependent

```
[12]: pd.DataFrame(train, columns = ['ProductCD', 'card4'])
```

```
[12]:
```

	ProductCD	card4
0	W	discover
1	W	mastercard
2	W	visa
3	W	mastercard


```

4          H  mastercard
...          ...
590535      W      visa
590536      W  mastercard
590537      W  mastercard
590538      W  mastercard
590539      W  mastercard

```

[590540 rows x 2 columns]

```
[13]: table = pd.crosstab(train['ProductCD'], train['card4'])
table
```

```
[13]: card4      american express  discover  mastercard    visa
ProductCD
C              2          0      27418    40904
H             1566       603       7938    22907
R             6138      1089       7432    23039
S              622       580       4042     6384
W              0      4379     142387   291533

```

```
[14]: observed_values = table.values
observed_values
```

```
[14]: array([[ 2,  0, 27418, 40904],
 [1566, 603, 7938, 22907],
 [ 6138, 1089, 7432, 23039],
 [  622,  580, 4042,  6384],
 [    0, 4379, 142387, 291533]], dtype=int64)
```

After finding the observed values, we will calculate the chi2 test statistic from the above table. Then we'll calculate the critical value and compare the values.

```
[15]: import scipy.stats as sp
from scipy.stats import chi2_contingency

#chi2 test stat, p-value, degrees of freedom expected frequencies
chi2_stat, p, dof, expected_freq = sp.chi2_contingency(table)
print('chi2 test stat =', round(chi2_stat, 4))
print('degrees of freedom =', dof)
```

```
chi2 test stat = 78171.3003
degrees of freedom = 12
```

```
[16]: from scipy.stats import chi2

crit_value = chi2.ppf(1 - sig_value, dof)
print("critical value =", round(crit_value, 4))
```

```

print()

if (chi2_stat >= crit_value):
    print("Because the chi2 test statistic is at least as large as the critical_
    ↪value,",
          "we reject the null hypothesis.")
    print("ProductCD and card4 are dependent features")
else:
    print("Because the chi2 test statistic is less than as the critical value,",
          "we do not reject the null hypothesis.")
    print("ProductCD and card4 are independent features")

```

critical value = 21.0261

Because the chi2 test statistic is at least as large as the critical value, we reject the null hypothesis.

ProductCD and card4 are dependent features

ProductCD and Card5 H_0 : ProductCD and Card5 independent

H_a : ProductCD and Card5 are dependent

```
[17]: pd.DataFrame(train, columns = ['ProductCD', 'card5'])
```

```
[17]:
```

	ProductCD	card5
0	W	142.0
1	W	102.0
2	W	166.0
3	W	117.0
4	H	102.0
...
590535	W	226.0
590536	W	224.0
590537	W	224.0
590538	W	224.0
590539	W	102.0

[590540 rows x 2 columns]

```
[18]: table = pd.crosstab(train['ProductCD'], train['card5'])
table
```

```
[18]:
```

card5	100.0	101.0	102.0	104.0	105.0	106.0	107.0	109.0	111.0	\
ProductCD										
C	4	7	3175	0	0	0	1	0	4	
H	149	0	2577	4	1	0	0	0	13	
R	40	0	2654	17	0	2	0	0	0	
S	11	0	2111	2	1	3	0	0	0	
W	774	0	18588	0	0	0	11	3	8	

card5	112.0	...	228.0	229.0	230.0	231.0	232.0	233.0	234.0	235.0	\
ProductCD		...									
C	3	...	62	0	4	0	61	15	0	0	
H	0	...	21	33	0	0	4	0	1	0	
R	0	...	125	5	0	0	0	0	0	0	
S	0	...	27	20	0	0	0	0	0	0	
W	0	...	13	1990	0	2	4	4	0	6	

card5	236.0	237.0
ProductCD		
C	0	0
H	175	7
R	434	39
S	38	3
W	0	0

[5 rows x 119 columns]

```
[19]: observed_values = table.values
```

After finding the observed values, we will calculate the chi2 test statistic from the above table. Then we'll calculate the critical value and compare the values.

```
[20]: #chi2 test stat, p-value, degrees of freedom expected frequencies
chi2_stat, p, dof, expected_freq = sp.chi2_contingency(table)
print('chi2 test stat =', round(chi2_stat, 4))
print('degrees of freedom =', dof)
```

chi2 test stat = 364652.2048
degrees of freedom = 472

```
[21]: crit_value = chi2.ppf(1 - sig_value, dof)
print("critical value =", round(crit_value, 4))
print()

if (chi2_stat >= crit_value):
    print("Because the chi2 test statistic is at least as large as the critical_
    value,",
          "we reject the null hypothesis.")
    print("ProductCD and card5 are dependent features")
else:
    print("Because the chi2 test statistic is less than as the critical value,",
          "we do not reject the null hypothesis.")
    print("ProductCD and card5 are independent features")
```

critical value = 523.6487

Because the chi2 test statistic is at least as large as the critical value, we

reject the null hypothesis.
ProductCD and card5 are dependent features

5.2.2 ProductCD and Card6

H_0 : ProductCD and Card6 are independent

H_a : ProductCD and Card6 are dependent

```
[22]: pd.DataFrame(train, columns = ['ProductCD', 'card6'])
```

```
[22]:      ProductCD  card6
0           W  credit
1           W  credit
2           W  debit
3           W  debit
4           H  credit
...         ...    ...
590535        W  debit
590536        W  debit
590537        W  debit
590538        W  debit
590539        W  credit
```

[590540 rows x 2 columns]

```
[23]: table = pd.crosstab(train['ProductCD'], train['card6'])
table
```

```
[23]: card6      charge card  credit  debit  debit or credit
ProductCD
C           12   27551   40763           0
H            0   17656   15367           0
R            3   26499   11192           0
S            0    6527    5100           0
W            0   70753  367516          30
```

```
[24]: observed_values = table.values
observed_values
```

```
[24]: array([[ 12, 27551, 40763,    0],
 [   0, 17656, 15367,    0],
 [   3, 26499, 11192,    0],
 [   0,  6527,  5100,    0],
 [   0, 70753, 367516,  30]], dtype=int64)
```

After finding the observed values, we will calculate the chi2 test statistic from the above table. Then we'll calculate the critical value and compare the values.

```
[25]: #chi2 test stat, p-value, degrees of freedom expected frequencies
chi2_stat, p, dof, expected_freq = sp.chi2_contingency(table)
print('chi2 test stat =', round(chi2_stat, 4))
print('degrees of freedom =', dof)
```

```
chi2 test stat = 87807.5884
degrees of freedom = 12
```

```
[26]: crit_value = chi2.ppf(1 - sig_value, dof)
print("critical value =", round(crit_value, 4))
print()

if (chi2_stat >= crit_value):
    print("Because the chi2 test statistic is at least as large as the critical_
    value,",
          "we reject the null hypothesis.")
    print("ProductCD and card6 are dependent features")
else:
    print("Because the chi2 test statistic is less than as the critical value,",
          "we do not reject the null hypothesis.")
    print("ProductCD and card6 are independent features")
```

```
critical value = 21.0261
```

Because the chi2 test statistic is at least as large as the critical value, we reject the null hypothesis.

ProductCD and card6 are dependent features

5.2.3 Card4 and Card5

H_0 : Card4 and Card5 are independent

H_a : Card4 and Card5 are dependent

```
[27]: pd.DataFrame(train, columns = ['card4', 'card5'])
```

```
[27]:
```

	card4	card5
0	discover	142.0
1	mastercard	102.0
2	visa	166.0
3	mastercard	117.0
4	mastercard	102.0
...
590535	visa	226.0
590536	mastercard	224.0
590537	mastercard	224.0
590538	mastercard	224.0
590539	mastercard	102.0

[590540 rows x 2 columns]

```
[28]: table = pd.crosstab(train['card4'], train['card6'])
table
```

```
[28]: card6          charge card  credit  debit  debit or credit
card4
american express      3    8175    144          0
discover              0    6304    347          0
mastercard            0   50772  138415         30
visa                 12   83732  301023          0
```

```
[29]: observed_values = table.values
observed_values
```

```
[29]: array([[ 3,  8175,  144,  0],
        [ 0,  6304,  347,  0],
        [ 0, 50772, 138415, 30],
        [12, 83732, 301023,  0]], dtype=int64)
```

After finding the observed values, we will calculate the chi2 test statistic from the above table. Then we'll calculate the critical value and compare the values.

```
[30]: #chi2 test stat, p-value, degrees of freedom expected frequencies
chi2_stat, p, dof, expected_freq = sp.chi2_contingency(table)
print('chi2 test stat =', round(chi2_stat, 4))
print('degrees of freedom =', dof)
```

```
chi2 test stat = 43314.2477
degrees of freedom = 9
```

```
[31]: crit_value = chi2.ppf(1 - sig_value, dof)
print("critical value =", round(crit_value, 4))
print()

if (chi2_stat >= crit_value):
    print("Because the chi2 test statistic is at least as large as the critical_
    →value,",
          "we reject the null hypothesis.")
    print("Card4 and card5 are dependent features")
else:
    print("Because the chi2 test statistic is less than as the critical value,",
          "we do not reject the null hypothesis.")
    print("Card4 and card5 are independent features")
```

```
critical value = 16.919
```

Because the chi2 test statistic is at least as large as the critical value, we reject the null hypothesis.

Card4 and card5 are dependent features

5.2.4 Card4 and Card6

H_0 : Card4 and Card6 are independent

H_a : Card4 and Card6 are dependent

```
[32]: pd.DataFrame(train, columns = ['card4', 'card6'])
```

```
[32]:      card4  card6
0    discover  credit
1  mastercard  credit
2      visa    debit
3  mastercard  debit
4  mastercard  credit
...      ...    ...
590535    visa    debit
590536  mastercard  debit
590537  mastercard  debit
590538  mastercard  debit
590539  mastercard  credit
```

[590540 rows x 2 columns]

```
[33]: table = pd.crosstab(train['card4'], train['card6'])
table
```

```
[33]: card6      charge card  credit  debit  debit or credit
card4
american express      3    8175    144              0
discover              0    6304    347              0
mastercard            0   50772  138415             30
visa                 12   83732  301023              0
```

```
[34]: observed_values = table.values
observed_values
```

```
[34]: array([[ 3,  8175,  144,  0],
        [ 0,  6304,  347,  0],
        [ 0, 50772, 138415, 30],
        [12, 83732, 301023,  0]], dtype=int64)
```

After finding the observed values, we will calculate the chi2 test statistic from the above table. Then we'll calculate the critical value and compare the values.

```
[35]: #chi2 test stat, p-value, degrees of freedom expected frequencies
chi2_stat, p, dof, expected_freq = sp.chi2_contingency(table)
print('chi2 test stat =', round(chi2_stat, 4))
print('degrees of freedom =', dof)
```

```
chi2 test stat = 43314.2477
degrees of freedom = 9
```

```
[36]: crit_value = chi2.ppf(1 - sig_value, dof)
print("critical value =", round(crit_value, 4))
print()

if (chi2_stat >= crit_value):
    print("Because the chi2 test statistic is at least as large as the critical_
    value,",
          "we reject the null hypothesis.")
    print("Card4 and card6 are dependent features")
else:
    print("Because the chi2 test statistic is less than as the critical value,",
          "we do not reject the null hypothesis.")
    print("Card4 and card6 are independent features")
```

```
critical value = 16.919
```

Because the chi2 test statistic is at least as large as the critical value, we reject the null hypothesis.

Card4 and card6 are dependent features

5.2.5 Card5 and Card6

H_0 : Card5 and Card6 are independent

H_a : Card5 and Card6 are dependent

```
[37]: pd.DataFrame(train, columns = ['card5', 'card6'])
```

```
[37]:
```

	card5	card6
0	142.0	credit
1	102.0	credit
2	166.0	debit
3	117.0	debit
4	102.0	credit
...
590535	226.0	debit
590536	224.0	debit
590537	224.0	debit
590538	224.0	debit
590539	102.0	credit

[590540 rows x 2 columns]

```
[38]: table = pd.crosstab(train['card5'], train['card6'])
table
```



```
[38]: card6  charge card  credit  debit  debit or credit
card5
100.0      0      0      978      0
101.0      0      0      7      0
102.0      0  28611      494      0
104.0      0      23      0      0
105.0      0      2      0      0
...      ...      ...      ...      ...
233.0      0      0      19      0
234.0      0      0      1      0
235.0      0      6      0      0
236.0      0    647      0      0
237.0      0     49      0      0
```

[119 rows x 4 columns]

```
[39]: observed_values = table.values
```

After finding the observed values, we will calculate the chi2 test statistic from the above table. Then we'll calculate the critical value and compare the values.

```
[40]: #chi2 test stat, p-value, degrees of freedom expected frequencies
chi2_stat, p, dof, expected_freq = sp.chi2_contingency(table)
print('chi2 test stat =', round(chi2_stat, 4))
print('degrees of freedom =', dof)
```

```
chi2 test stat = 236793.9479
degrees of freedom = 354
```

```
[41]: crit_value = chi2.ppf(1 - sig_value, dof)
print("critical value =", round(crit_value, 4))
print()

if (chi2_stat >= crit_value):
    print("Because the chi2 test statistic is at least as large as the critical_
    ↪value,",
          "we reject the null hypothesis.")
    print("Card5 and card6 are dependent features")
else:
    print("Because the chi2 test statistic is less than as the critical value,",
          "we do not reject the null hypothesis.")
    print("Card5 and card6 are independent features")
```

```
critical value = 398.8739
```

Because the chi2 test statistic is at least as large as the critical value, we reject the null hypothesis.

Card5 and card6 are dependent features