金融科技
文字探勘與機器學習

玉山第三組

吳佳展、翁光昕、黃詩晴、游子有

理解状況

資料狀況 & 題目設定

- 資料：
  - 10萬個客戶基本屬性資料
  - 為期9個季（554個交易日）的所有客戶交易明細（購買標的、股數）
  - 違約註記（共96筆違約交割，發生於91位客戶身上）
  - 554交易日之中所有標的的資訊（開高低收量Beta）

- 題目設定：
  - 一開始鎖定在客戶的交易行為有改變
  - 後來鎖定在預測該客戶的交易是否有可能違約

target 鎖定在違約交割的classification

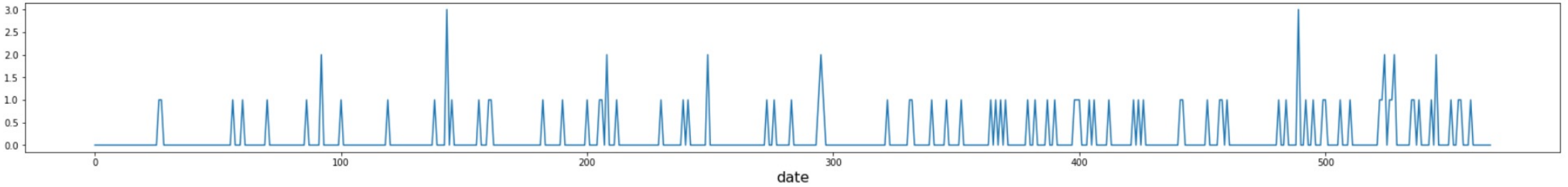
- 主觀認知：

1. 交易金額、交易頻率高者（計算每位客戶的交易次數、金額）
2. 有當沖、融資融券者
3. 年輕較缺乏經驗的用戶
4. ***自身經驗*** 算錯錢，自以為餘額夠

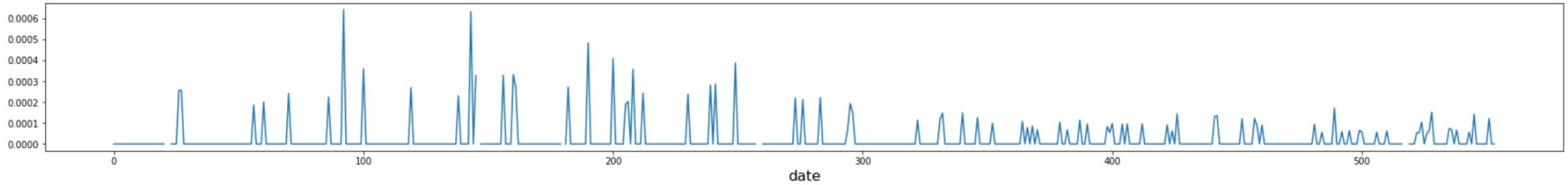
觀察資料

觀察違約交割的客戶分布狀況

- 在554個交易日裡面違約交割的次數分佈

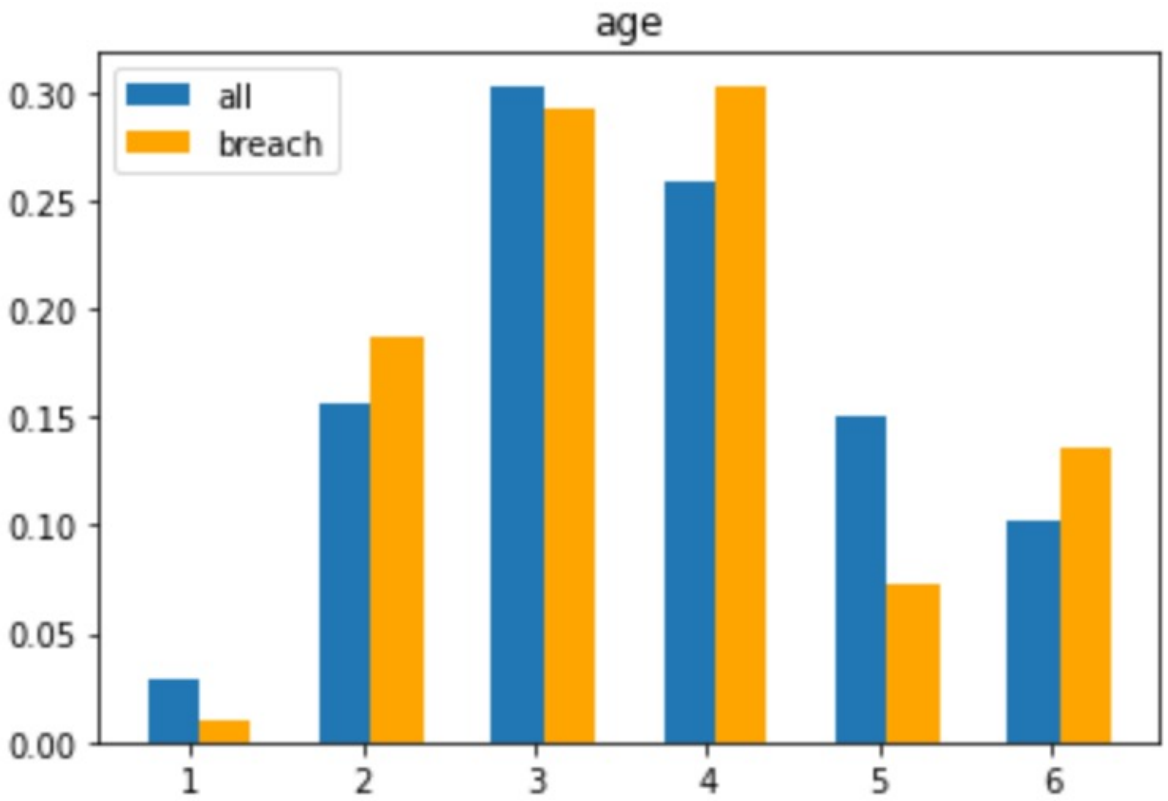


- 每天有違約交割的人口比例（分母為當天有下單者才計算）



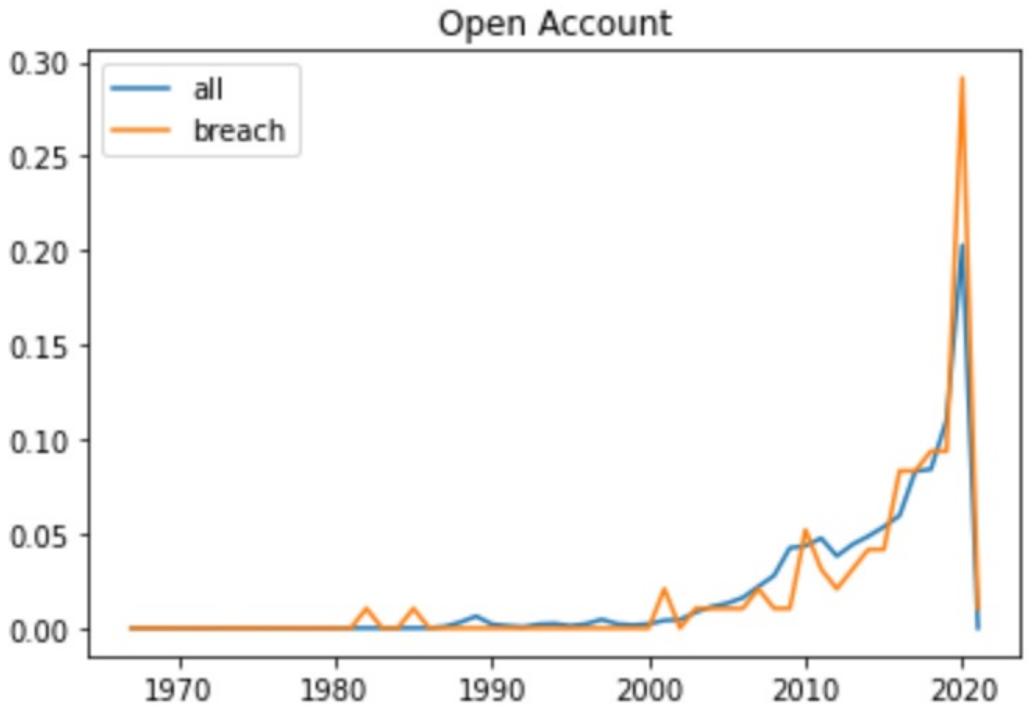
觀察違約交割的客戶分布狀況

- 觀察違約名單的年齡分佈，看起來年齡並不會明顯影響是否違約



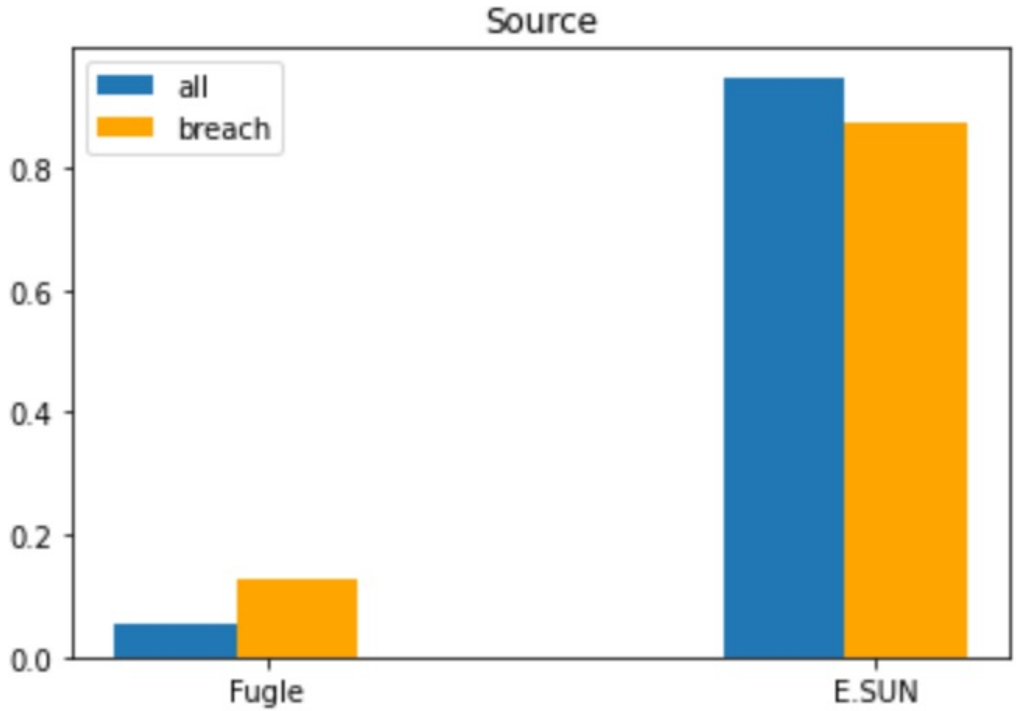
觀察違約交割的客戶分布狀況

- 開戶日期的比較，走勢大致相同，但看起來2020新用戶的違約比例確實有高一截



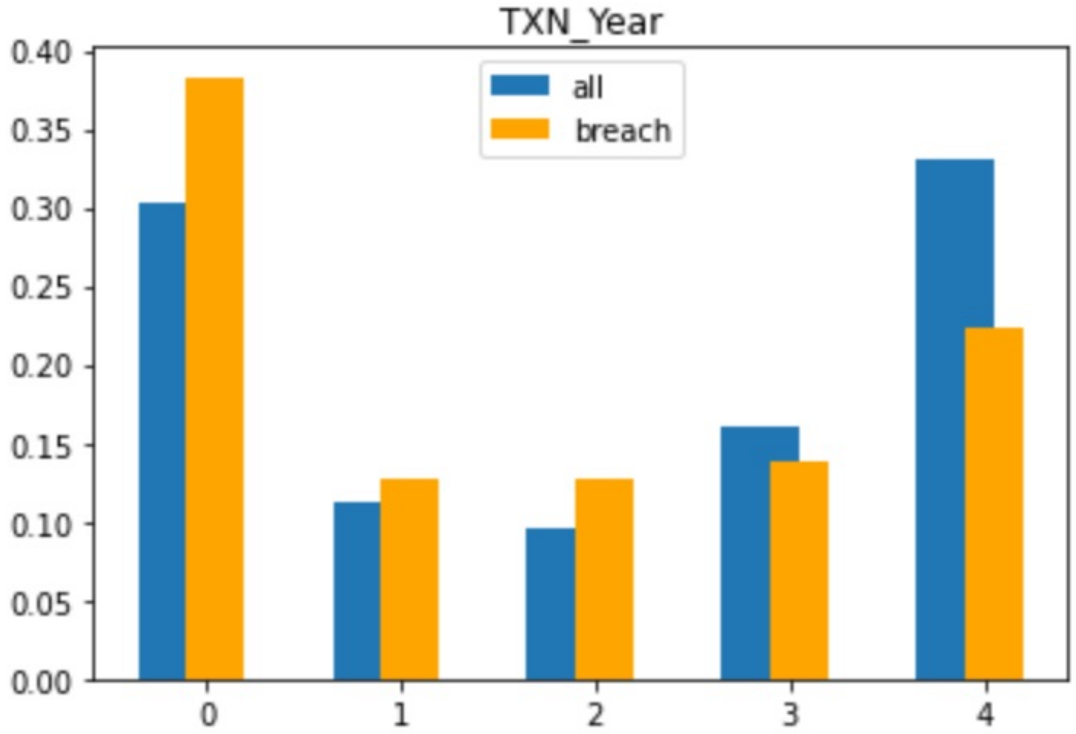
觀察違約交割的客戶分布狀況

- 證券戶的開設來源（玉證/富果），看起來也沒有明顯的差別



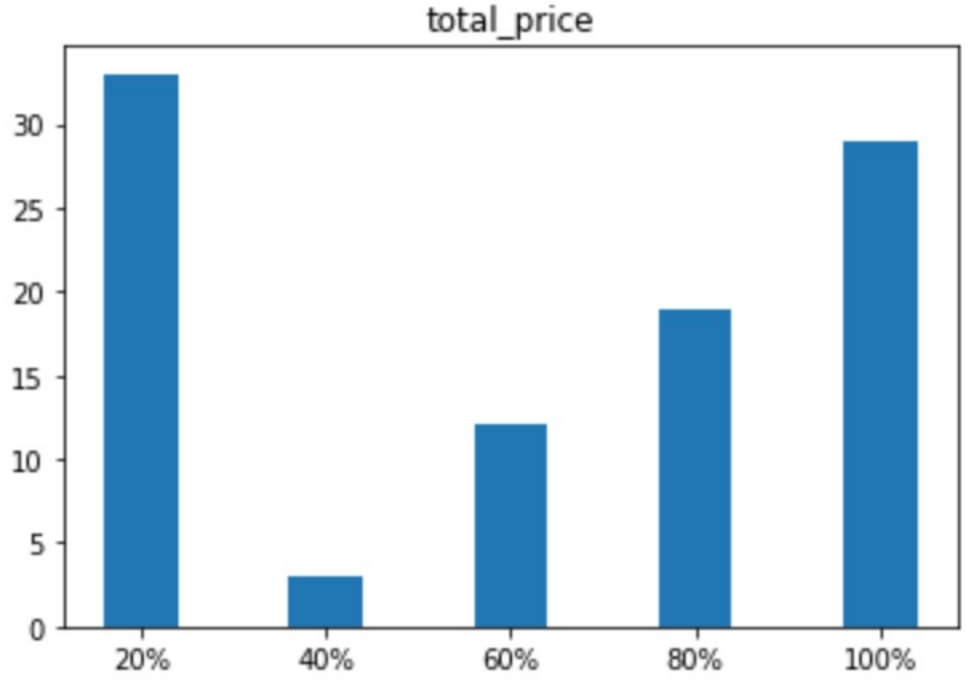
觀察違約交割的客戶分布狀況

- 交易經驗的年資，看起來交易年資未滿一年有比較高的違約比例



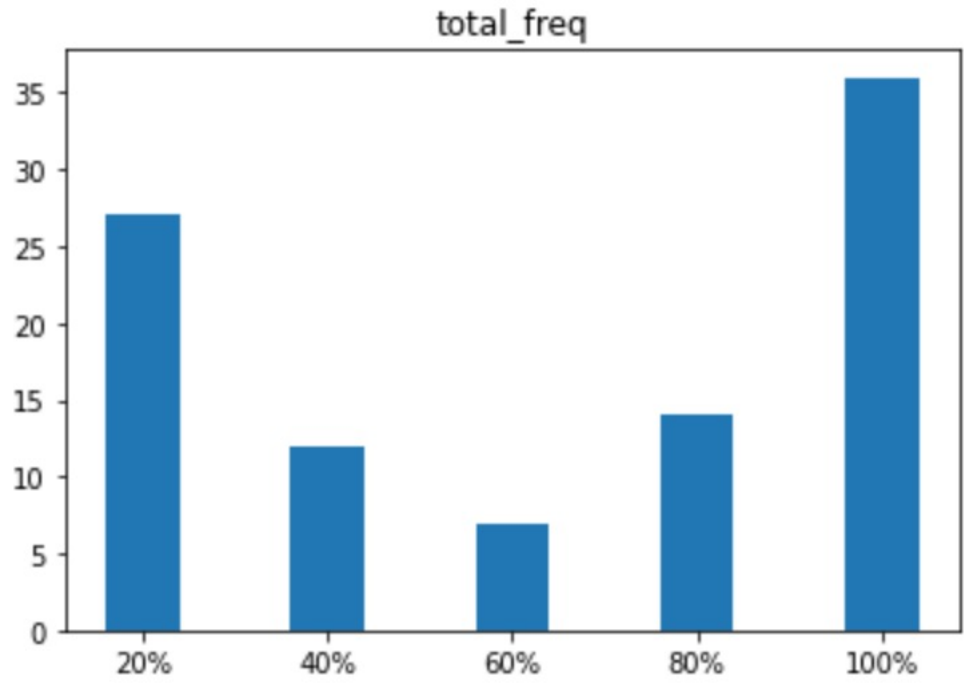
觀察違約交割的客戶分布狀況

- 累積交易金額，91名違約交割者：
  - 其中有超過30名他們的累積交易金額位在10萬母體樣本中的前20%
  - 另外有接近30名他們的累積交易金額位在10萬母體樣本中的後20%



觀察違約交割的客戶分布狀況

- 累積交易次數，91名違約交割者：
  - 其中有超過25名他們的累積交易次數位在10萬母體樣本中的前20%
  - 另外有超過35名他們的累積交易次數位在10萬母體樣本中的後20%



觀察違約交割的客戶分布狀況

- 主觀並且簡單的下個觀察結論：
  - 交易金額（次數）前段
  - 交易金額（次數）後段
  - 新開戶（交易年資未滿一年）

- 但畢竟違約樣本過少，看起來還是偏個案成分居多

資料結構與特徵擷取

資料到底長怎樣

CUST_NO	AGE	OPEN_ACCT	SOURCE	BREACH	BREACH_DATE	BREACH_RANK	TXN_YEAR	...	txn_Q8_freq	txn_Q8_day	txn_Q8_price	txn_Q9_freq	txn_Q9
7C5334D...	4	2005	B	1	546.0	1.0	4.0	...	0	0	0.00	226	
6B4F72E...	4	2016	B	1	544.0	1.0	3.0	...	32	15	4890980.00	34	
\B659144...	4	2021	B	1	539.0	1.0	0.0	...	0	0	0.00	20	
642E77C...	4	2016	B	1	529.0	1.0	3.0	...	0	0	0.00	2	
91AF368...	4	2001	B	1	525.0	2.0	4.0	...	46	18	2865900.00	50	
91AF368...	4	2001	B	1	524.0	1.0	4.0	...	46	18	2865900.00	50	
5F4503A...	4	2010	B	1	511.0	1.0	4.0	...	60	24	8050598.89	72	
A3CBB18...	4	2004	B	1	507.0	1.0	3.0	...	2	1	344000.00	8	
29740F52...	4	2016	B	1	490.0	1.0	0.0	...	20	14	5020974.00	60	
\06F287A...	4	2010	B	1	427.0	1.0	4.0	...	0	0	0.00	0	
D742CF0...	4	2018	B	1	391.0	1.0	2.0	...	0	0	0.00	0	
5F7FD40...	4	2018	B	1	383.0	1.0	2.0	...	14	12	660280.00	61	
29BEA5F...	4	2017	B	1	369.0	1.0	3.0	...	2	2	589800.00	0	
973ADFB...	4	2019	B	1	323.0	1.0	1.0	...	0	0	0.00	0	
7419C68...	4	2010	B	1	295.0	1.0	4.0	...	0	0	0.00	0	
ABFCDD...	4	2003	B	1	274.0	1.0	4.0	...	0	0	0.00	0	
73F15908...	2	2019	B	1	250.0	2.0	1.0	...	0	0	0.00	0	
FDDD458...	4	2014	B	1	242.0	1.0	2.0	...	0	0	0.00	0	
6017A4A...	4	2014	B	1	213.0	1.0	4.0	...	26	15	6620900.00	36	
73F15908...	2	2019	B	1	201.0	1.0	1.0	...	0	0	0.00	0	
CA920F4...	4	2009	B	1	157.0	1.0	4.0	...	0	0	0.00	0	

資料到底長怎樣

客戶 ID3	年齡	年資	購買過的股票	累積交易金額	累積交易次數	信用當沖	是否違約
2973	2	1	……	4837692	158	1	1
89671	1	2	……	198275	21	0	0
917	4	1	……	27582714	204	1	0
102769	3	4	……	3719572	40	0	0
93	4	3	……	4028962	38	0	0
64023	4	1	……	37161849	197	1	0
75820	1	2	……	836193	71	0	1

維度？客戶？時間序列？

客戶ＩＤ	交易日	基本屬性	……	累積交易金額	累積交易次數	是否違約
2973	1			0	0	0
	2			39710	1	0
	3			39710	1	0
	…..			…..	…..	1
	554			4837692	158	0
89671	1			0	0	0
	2			0	0	0
	3			21890	1	0
	…			…..	…..	0
	554			198275	21	0
917						
…..						

違約交割的比例到底有多低

- 10萬個客戶
- 554個交易日
- 剛剛的表格應該就會有 554*105,770 個row
- 五千多萬個樣本 卻只有 96 例違約
- 極度失衡的資料

- 認為：違約交割完全就是特殊個案

資料特徵擷取 Feature Selection

- 交易年資、交易次數、交易金額(收支)
- ***自身經驗***
  - 會違約基本上應該是兩種狀況
    1. 忘記有買股票
    2. 算錯錢
  - 怎樣會算錯錢：前兩天有交易
- 當天大盤走勢 & 當日漲停跌停股票個數

Training Data

- 考慮交割都是 T+2
  1. 當天收支金額、當天交易次數
  2. 昨日收支金額、昨日交易次數
  3. 前天收支金額、前天交易次數
- 並且只拿曾經有違約紀錄的客戶作為training data
  - training data : 300
  - validation data : 100
  - testing data : 154
  - 554*105,770 → 400*91
- 因為每個人的交易金額的區間不同
  - 針對金額有對每個人自己做Normalize（分母不應該用到後面資料、生哐平均）
  - 次數的部分就沒有了

Training Data

	pay	times	pay_1	times_1	pay_2	times_2	target
6739	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.0
6740	-2.071948	3.0	0.000000	0.0	0.000000	0.0	0.0
6741	0.000000	0.0	-2.071948	3.0	0.000000	0.0	0.0
6742	0.000000	0.0	0.000000	0.0	-2.071948	3.0	0.0
6743	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.0
6744	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.0
6745	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.0
6746	-6.876302	1.0	0.000000	0.0	0.000000	0.0	0.0
6747	-0.347212	4.0	-6.876302	1.0	0.000000	0.0	0.0
6748	0.000000	0.0	-0.347212	4.0	-6.876302	1.0	0.0
6749	3.370218	7.0	0.000000	0.0	-0.347212	4.0	0.0
6750	-0.132091	2.0	3.370218	7.0	0.000000	0.0	0.0
6751	100.000000	6.0	-0.132091	2.0	3.370218	7.0	0.0
6752	-90.992497	3.0	100.000000	6.0	-0.132091	2.0	0.0
6753	0.000000	0.0	-90.992497	3.0	100.000000	6.0	1.0
6754	0.000000	0.0	0.000000	0.0	-90.992497	3.0	0.0
6755	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.0
6756	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.0
6757	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.0
6758	0.000000	0.0	0.000000	0.0	0.000000	0.0	0.0

Testing Data

	pay	times	pay_1	times_1	pay_2	times_2
319	-30.735931	1.0	0.000000	0.0	0.000000	0.0
320	-2.272727	3.0	-30.735931	1.0	0.000000	0.0
321	1.082251	2.0	-2.272727	3.0	-30.735931	1.0
322	16.396104	1.0	1.082251	2.0	-2.272727	3.0
323	-54.653680	3.0	16.396104	1.0	1.082251	2.0
324	54.653680	3.0	-54.653680	3.0	16.396104	1.0
325	-3.896104	6.0	54.653680	3.0	-54.653680	3.0
326	-4.329004	2.0	-3.896104	6.0	54.653680	3.0
327	5.762987	4.0	-4.329004	2.0	-3.896104	6.0
328	0.000000	0.0	5.762987	4.0	-4.329004	2.0
329	-3.246753	2.0	0.000000	0.0	5.762987	4.0

模型架構與訓練

Model

- 標準 DNN 的 Regression model
  - 使用三層網路架構
  - Active function: ReLU
  - BatchNorm
  - Dropout
  - epoch : 300
  - optimizer : Adam

```
# Define your neural network here

self.net = nn.Sequential(
    nn.Linear(input_dim, 64),
    nn.BatchNorm1d(64),
    nn.ReLU(),
    nn.Dropout(0.3),
    nn.Linear(64,64),
    nn.BatchNorm1d(64),
    nn.ReLU(),
    nn.Dropout(0.3),
    nn.Linear(64, 1)
)
```

Why Regression ?

- 資料過度失衡，不易於使用傳統classification
- 利用regression找出哪些feature會違約的可能性有多大
- 沒有使用 RNN 的原因：
  1. 那架構好難寫
  2. 雖然是時間序列，但真的很吃歷史資料嗎？

模型表現結果

Testing 為有違約紀錄的所有客戶

- Testing data共有 154*91 = 14,014 筆
- Ground Truth 總共有 39 筆違約交割

```
acc(pred, answer,0.1)
```

```
There are  39  breach datas in groundtruth
We catch  29  breach datas only using  95  predictions
The total of data in this dataset are  14014
```

```
acc(pred, answer,0.01)
```

```
There are  39  breach datas in groundtruth
We catch  35  breach datas only using  355  predictions
The total of data in this dataset are  14014
```

```
acc(pred, answer,0.00058)
```

```
There are  39  breach datas in groundtruth
We catch  39  breach datas only using  13796  predictions
The total of data in this dataset are  14014
```

Testing 為有違約紀錄的所有客戶

id	tested_positive
260	0.012025
261	0.023594
324	0.015910
326	0.010076
332	0.011077
390	0.020716
391	0.012740
418	0.011099
419	0.022054
589	0.012254
590	0.018476
591	0.154359
702	0.031724
703	0.021702
704	0.015007
705	0.019041
706	0.036707
714	0.024104
715	0.018342
716	0.013294

id	tested_positive
717	0.014085
721	0.018102
723	0.015522
730	0.010370
731	0.026552
737	0.015637
739	0.015301
740	0.020359
741	0.010626
747	0.013075
749	0.027412
750	0.015252
751	0.014127
894	0.013068
895	0.035556
1153	0.061772
1156	0.012793
1159	0.021321
1161	0.013004
1162	0.084091

id	tested_positive
1165	0.046603
1166	0.017968
1168	0.011145
1169	0.012257
1257	0.011991
1289	0.012197
1290	0.023236
1294	0.010028
1316	0.015129
1319	0.026916
1338	0.010601
1377	0.024088
1382	0.012418
1403	0.010608
1404	0.013301
1413	0.012122
1836	0.014336
1913	0.017005
2719	0.010133
2722	0.040061

id	tested_positive
2733	0.016996
2734	0.017769
2735	0.013467
2738	0.011681
2744	0.052889
2749	0.011939
2750	0.053284
2751	0.033118
2752	0.017808
2753	0.156709
2763	0.012672
2764	0.095015
3215	0.025861
3217	0.015989
3218	0.011871

	target
261	1.0
419	1.0
591	1.0
706	1.0
894	1.0
895	1.0
1222	1.0
1413	1.0
2764	1.0
3219	1.0

Testing 為沒有違約紀錄的隨機抽樣客戶

- Testing data 共有 254*100 = 25,400 筆
- Ground Truth 總共有 0 筆違約交割

```
acc(pred, answer,0.1)
```

```
There are  0  breach datas in groundtruth
We catch  0  breach datas only using  269  predictions
The total of data in this dataset are  25400
```

```
acc(pred, answer,0.01)
```

```
There are  0  breach datas in groundtruth
We catch  0  breach datas only using  1243  predictions
The total of data in this dataset are  25400
```

Testing 為沒有違約紀錄的隨機抽樣客戶

- 雖然這些是沒有發生違約的資料，但或許他們的feature顯示出其實潛藏有違約的可能性，也有可能是因為專員有通知要補錢。
- 寧可 False Positive 大，也不要 False Negative 大。

Testing為違約和不違約的客戶各佔一半

- Testing data共有 154*100 = 15,400 筆
- Ground Truth 總共有 24 筆違約交割

```
acc(pred, answer,0.1)
```

```
There are  24  breach datas in groundtruth
We catch  18  breach datas only using  114  predictions
The total of data in this dataset are  14938
```

```
acc(pred, answer,0.01)
```

```
There are  24  breach datas in groundtruth
We catch  23  breach datas only using  569  predictions
The total of data in this dataset are  14938
```

把 Testing當天沒有交易的資料拔除

- 由於認為資料有太多的0會導致失真，所以在testing的時候嘗試把當天沒有交易的資料全數刪除，只保留當天有交易的資料丟進去model做測試。

把 Testing當天沒有交易的資料拔除

- Testing 為針對有違約紀錄的客戶，也就是91*154共14,014筆資料，但是刪除當日沒有交易的資料，因此剩餘 742筆資料
- 經過 model的表現如圖所示

```
acc(pred, answer,0.1)
```

```
There are  37  breach datas in groundtruth
We catch  10  breach datas only using  27  predictions
The total of data in this dataset are  742
```

```
acc(pred, answer,0.01)
```

```
There are  37  breach datas in groundtruth
We catch  30  breach datas only using  152  predictions
The total of data in this dataset are  742
```

把 Testing當天沒有交易的資料拔除

- Testing 為有無違約紀錄的客戶各取50名，也就是100*154共15,400筆資料，但是刪除當日沒有交易的資料，因此剩餘 1,201 筆資料
- 經過 model的表現如圖所示

```
acc(pred, answer,0.1)
```

```
There are  22  breach datas in groundtruth
We catch  7  breach datas only using  26  predictions
The total of data in this dataset are  1201
```

```
acc(pred, answer,0.01)
```

```
There are  22  breach datas in groundtruth
We catch  19  breach datas only using  188  predictions
The total of data in this dataset are  1201
```

未來規劃與新資料的預測

To Do

- 關於model的threshold，應該要針對企業對於False Positive與False Negative的成本去權衡出最適的threshold。
- Lift Chart
- AUROC
- 隨時學習最新資料
- 大盤走勢應該可以放進去當作feature
- 關於交易金額Normalize的部分應該要再修正

新資料預測力

- 不是太樂觀，因為並未考量到大盤以及各股狀況，甚至連客戶基本屬性資料都沒有，完全只有使用過去兩天內的交易狀況。
- 有可能model其實是overfit在這次的資料集。
- 2021.05的台股大崩盤可能會失準。