

# 金融科技與文字探勘

## 野村題目一 組別二

K O L、網路情緒指標與投資人行為之早期預測

### 組員：

台大財金所碩一	謝侑軒
台大農經系大六	莊浩平
師大數學所碩一	蔡承男
政大企管系大三	曾琮傑
政大科智所碩一	劉珈卉

**指導老師：**石百達、張智星 老師

**指導業師：**Tony / 黃俊哲



# Agenda

01

資料簡介

02

PTT論壇分析

03

強基金論壇分析

04

研究發現

05

應用程式與其他



# 資料簡介



# 資料簡介

PTT的主要使用者為年輕群眾，強基金的使用者則多為青壯年，也因此我們預期將此兩種資料來源合併分析，我們認為可以得到更完整的輿情結果。

## 以下是我們資料集的選取原因：

- PTT的Stock版
  - ✓ 因為在Stock版為熱門看板，相比於Fund版，討論比較熱烈，而且也比較多意見領袖
- 強基金論壇
  - ✓ 由於此論壇專注討論基金相關議題，所以我們資料涵蓋此論壇各個討論版



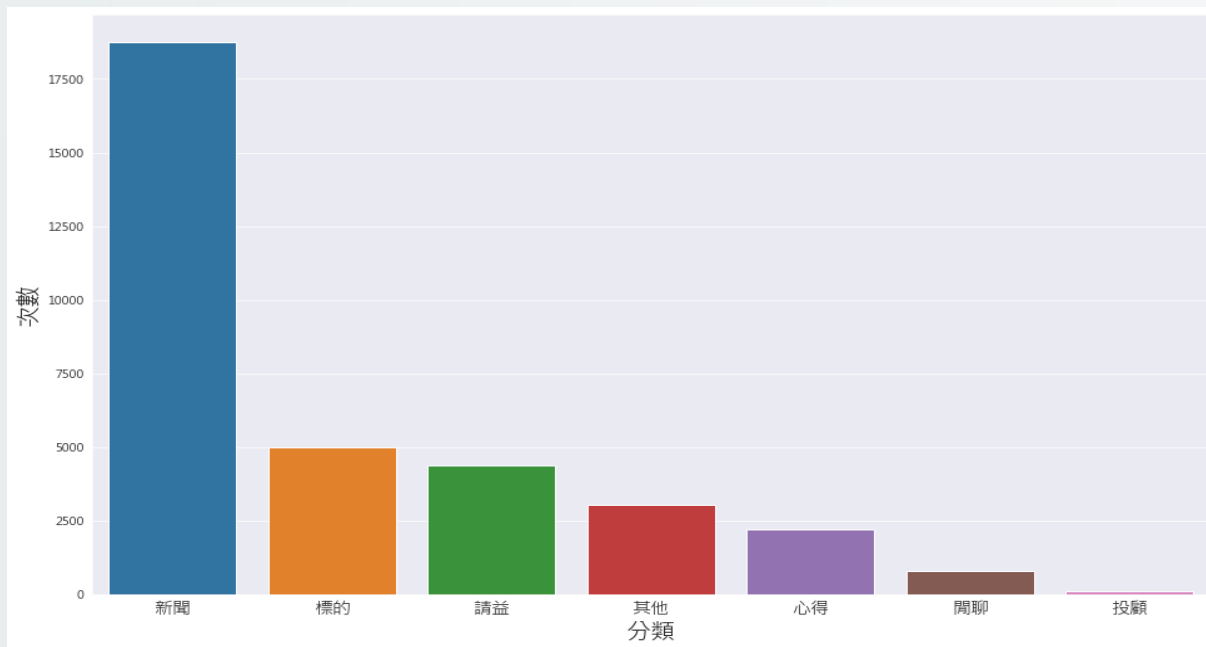
# PTT資料簡介

資料區間：2018/02/01 – 2020/05/01

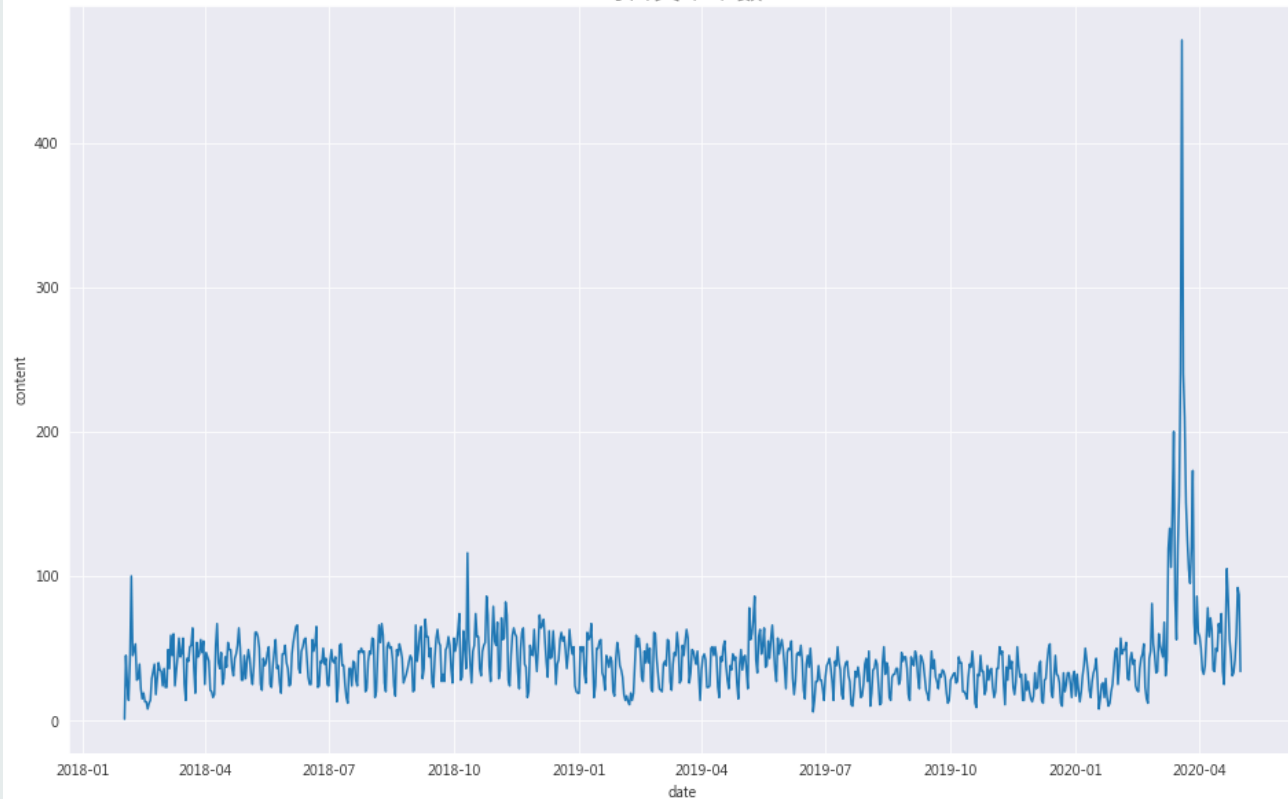
資料筆數：34354

資料來源：PTT Stock版

- 資料特色：各個文章有類別標籤(如右圖)
- 採用文章正文



每日文章筆數



## 統計量

count	821.000000
mean	41.844093
std	27.711054
min	1.000000
25%	27.000000
50%	39.000000
75%	50.000000
max	471.000000

# 強基金論壇資料簡介

資料區間：2011/01/01 – 2020/05/09

資料筆數：54985

資料來源：強基金論壇

強基金論壇的特點是，留言回復也很有資訊含量，且專注於基金的討論，年齡層相較PTT更成熟一些

- 論壇資料：主要還是以強基金論壇作為文字資料以及模型 fitting、training 的來源
- 該論壇運行時間約從 2011 年開始
- 文章數量約為 8000 篇
- 3000 位不重複使用者的發言紀錄
- 不計文章正文，約有 55000 篇回文

Re: 摩根士丹利印度基金-27 %，該續扣養大？還是該痛下決心贖回？

jett

發表於：2020-06-16, 08:08



天使人

刪除

註冊時間:

2016-11-18, 07:43

文章: 400

定期定額很久了吧？我猜低檔應該沒加扣，報酬率才會如此，不是基金的問題，是你操作習慣的問題

一堆資深強友正在討論加碼印度基金，你選擇在此時退出，反而可能賣在低點

股災買了東協、泰國、印尼、印度、馬來西亞、巴西、南韓基金，績效不錯

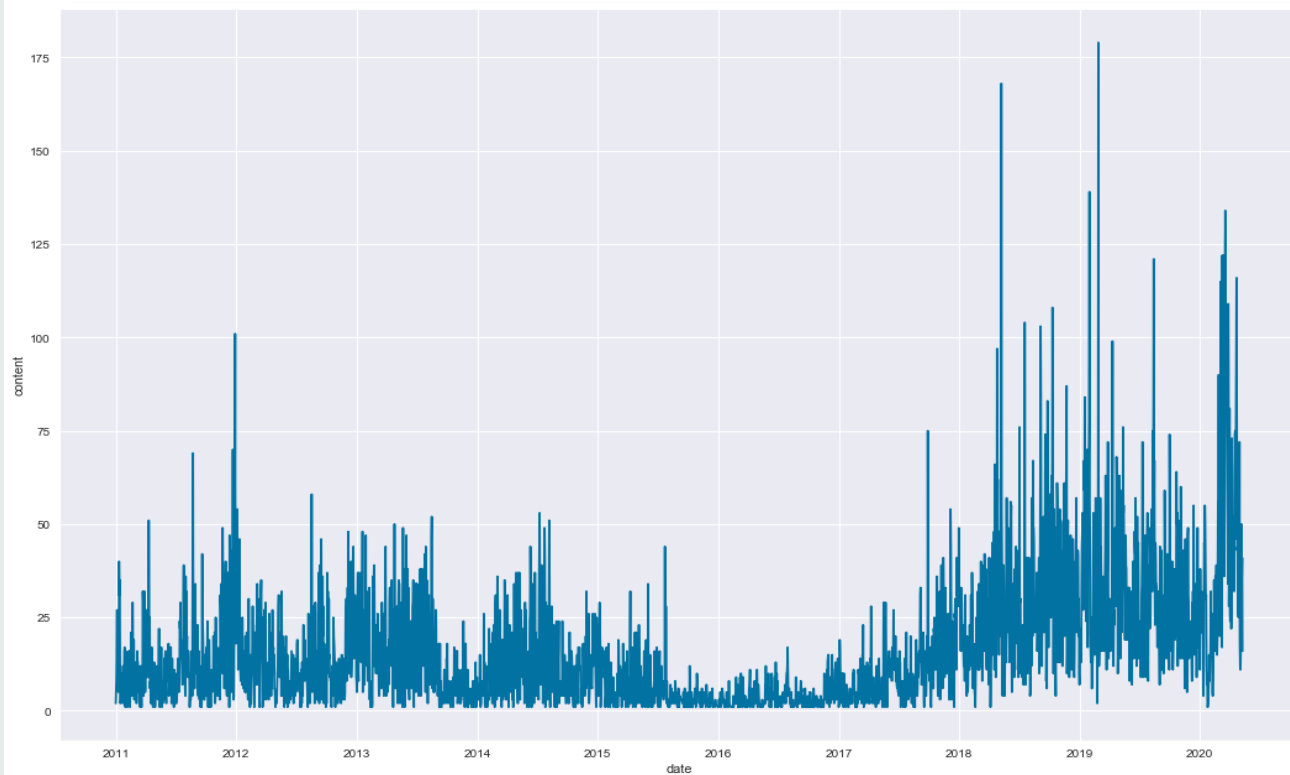
<https://forum.fundhot.com/viewtopic.php?t=8230>

當然如果你困擾到睡不著，也沒太多錢可加碼，就贖回吧

重來一次，請看FBI逢低加扣，效果才會好

「修正是被圍困的城堡，城外的人想衝進去，城裡的人想逃出來」

每日文章筆數



## 統計量

count	3235.000000
mean	16.997218
std	17.228887
min	1.000000
25%	5.000000
50%	12.000000
75%	24.000000
max	179.000000





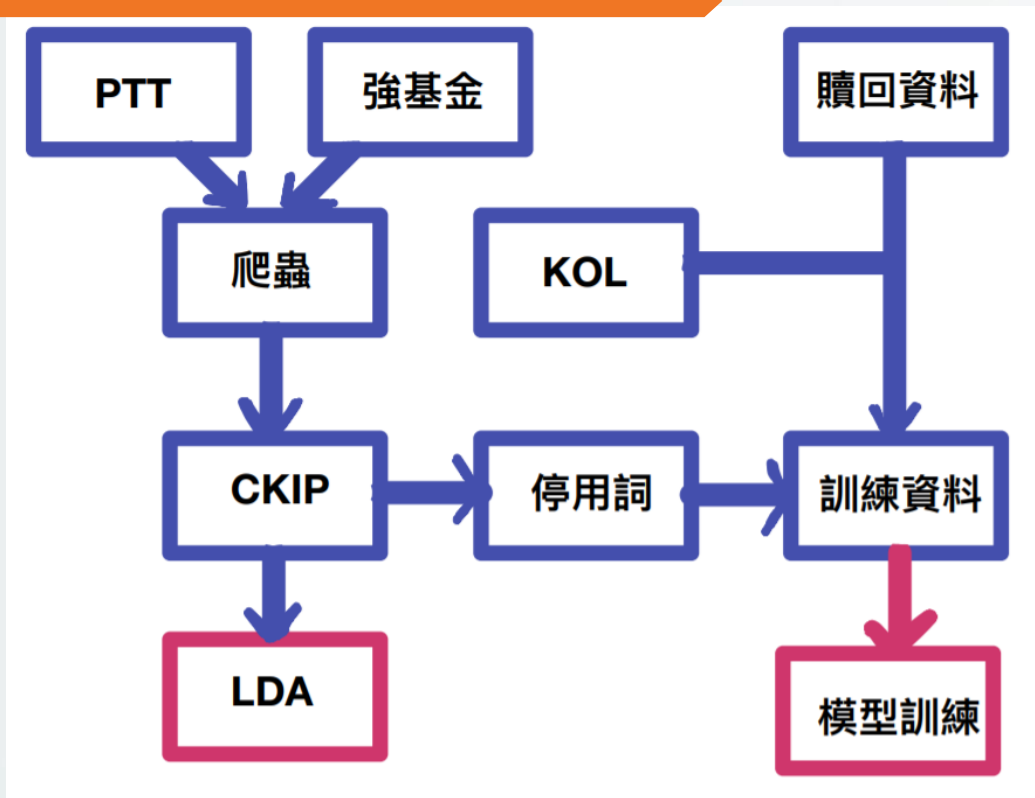
# PTT論壇分析



# 研究議題

透過KOL的文本來預測贖回率是否上升

- 指標建立
  - KOL定義
  - 贖回率指標
- 研究方法
  - 斷詞
  - 找尋主題
  - 找尋關鍵字
  - 預測
- 模型結果
  - 字典法+傳統機器學習
  - 自然語言處理-BERT



# 指標建立

- KOL定義

1. 指標一：文章數前十名
2. 推文超過30的文章數前十名
3. 扣除”新聞”及”其他”後，發文數最多
4. 利用PageRank概念找出分數最高的人做KOL

- 贖回率定義

1. 野村投信的「股票型基金」平均贖回率是否高於上個月
2. 野村投信的「台灣股票型基金」平均贖回率是否高於上個月
3. 野村投信的「全部基金」平均贖回率是否高於上個月

# 版上KOL

## 指標一：文章數前十名

author  
nightwing 849  
CLV518 805  
ESunBoy 573  
Su22 535  
coconing 506  
cjol 495  
zxcvxx 462  
mayingnine 386  
l75cm 354  
anti87 330

大多為新聞，並非個人見解

	time	author	type	title	score	filename
6760	2018-07-26 09:39:57	nightwing	新聞	陸最大黃豆進口商較產 董事長曾為山東自	31	pttdata\M.1532569201.A.EA6.json \n1.原文連結: \nhttps://tinyurl.com/y8dt原...
6860	2018-07-28 16:01:19	nightwing	新聞	陸P2P保雷雷 釐千萬金融難民	59	pttdata\M.1532764883.A.1AF.json \n1.原文連結: \nhttps://tinyurl.com/yccdm6ng\n\n\n2...
6893	2018-07-30 08:52:12	nightwing	新聞	傳中國官方要擴大基建 降低貿易戰衝	77	pttdata\M.1532911936.A.43A.json \n1.原文連結: \nhttps://tinyurl.com/y99lyrtj\n\n\n2...
6939	2018-07-31 08:50:01	nightwing	新聞	經濟轉型 中國首現17年來半年期貿易赤字	17	pttdata\M.1532998204.A.E8F.json \n1.原文連結: \nhttps://tinyurl.com/y89c原...
6990	2018-08-01 11:15:37	nightwing	新聞	騰訊市值蒸發逾4兆台幣 全球股市最大苦主	33	pttdata\M.1533093341.A.9F4.json \n1.原文連結: \nhttps://tinyurl.com/yd4n8q4\n\n2.原...
...	...	...	...	...	...	...
29337	2020-03-09 08:00:52	nightwing	新聞	因應武漢肺炎 歐洲央行要員工在家上班	6	pttdata\M.1583712055.A.A11.json \n1.原文連結: \n https://tinyurl.com/tv8u
29595	2020-03-11 07:40:29	nightwing	新聞	武漢肺炎》蘋果歐洲總部失守 1員工確診	5	pttdata\M.1583883632.A.7BC.json https://tinyurl.com/vo7pc4n\n\n2....
33431	2020-04-16 06:13:28	nightwing	新聞	疫情重挫新加坡淡馬錫 華一季慘賠235億	148	pttdata\M.1586988810.A.D4A.json \n1.原文連結: \n https://tinyurl.com/s5t4
33495	2020-04-17 06:16:50	nightwing	新聞	油價戰新苦主！新加坡石油巨頭驟降900億	24	pttdata\M.1587075413.A.6DD.json \n1.原文連結: \n https://tinyurl.com/ybhgg8s\n\n\n2...
33698	2020-04-21 07:31:57	nightwing	新聞	星石油巨擘申請破產！傳虧損8億美元虧損	74	pttdata\M.1587425520.A.E4A.json \n1.原文連結: \n https://tinyurl.com/yc8zq36k\n\n\n2...

	time	author	type	title	score	filename
8356	2018-09-03 14:52:59	CLV518	標的	大差 多	12	pttdata\M.1535957581.A.6A8.json \n1. 標的: 大盤\n\n2. 分類: 多\n\n3. 分析/正文: 這一根回檔也...
8632	2018-09-08 17:03:15	CLV518	新聞	貿易戰驟加2670億 川普實現四大戰略	55	pttdata\M.1536397398.A.D3D.json \n1.原文連結: \nhttp://www.epochtimes.com/b
8638	2018-09-08 19:23:23	CLV518	新聞	川普: 希望取消對中印等國的經濟援助	37	pttdata\M.1536405805.A.D8A.json \n1.原文連結: \nhttp://www.epochtimes.com/b
8645	2018-09-09 00:38:56	CLV518	新聞	貿易戰恐使經濟下滑 中共二個月發債萬億	11	pttdata\M.1536424738.A.064.json \n1.原文連結: \nhttp://www.epochtimes.com/b
8658	2018-09-09 10:45:57	CLV518	新聞	將中共定為貨幣操縱? 川普當局雙再出招	19	pttdata\M.1536461160.A.EEE.json \n1.原文連結: \nhttp://www.epochtimes.com/b
...	...	...	...	...	...	...
24716	2019-10-15 14:39:01	CLV518	新聞	大陸9月進出口超預期雙降	8	pttdata\M.1571121543.A.E5E.json \n1.原文連結: \nhttp://www.epochtimes.com/b
24724	2019-10-15 20:09:59	CLV518	新聞	大陸9月CPI上漲3% 豬肉價格大漲近七成	15	pttdata\M.1571141403.A.F33.json \n1.原文連結: \nhttp://www.epochtimes.com/b
24748	2019-10-16 12:36:31	CLV518	新聞	權美國退出萬國郵聯 北京同意明年逐郵費	12	pttdata\M.1571200593.A.1A0.json \n1.原文連結: \nhttps://www.epochtimes.com/t
24785	2019-10-17 12:15:11	CLV518	新聞	9月外儲量跌 專家: 資金加速離開中國大陸	9	pttdata\M.1571285714.A.83B.json \n1.原文連結: \nhttps://www.epochtimes.com/t
24794	2019-10-17 16:35:40	CLV518	新聞	姆欽: 美中在確定初步協議文本 含知識產權	9	pttdata\M.1571301344.A.77A.json \n1.原文連結: \nhttps://www.epochtimes.com/t

# 版上KOL

## 指標二：

## 推文超過30的文章數前十名

author	
coconing	460
vendan5566	264
nightwing	217
hito21	165
justforsing	149
CLV518	121
tim0259	120
EDFR	113
ESunBoy	111
zesonpso	100

## 為公告性質的文章居多

	time	author	type	title	score	filename
179	2018-02-06 15:02:44	coconing	其他	107年02月06日 三大法人買賣金額統計表	111	pttdata\M.1517900566.A.0B4.json
348	2018-02-09 15:01:28	coconing	其他	107年02月09日 三大法人買賣金額統計表	88	pttdata\M.1518159691.A.0B8.json
454	2018-02-12 14:59:35	coconing	其他	107年02月12日 三大法人買賣金額統計表	126	pttdata\M.1518418779.A.A3D.json
600	2018-02-21 15:01:56	coconing	其他	107年02月21日 三大法人買賣金額統計表	51	pttdata\M.1519196519.A.01F.json
630	2018-02-22 15:01:09	coconing	其他	107年02月22日 三大法人買賣金額統計表	29	pttdata\M.1519282873.A.BEE.json
...	...	...	...	...	...	...
33826	2020-04-22 15:05:34	coconing	其他	109年04月22日 三大法人買賣金額統計表	152	pttdata\M.1587539137.A.216.json
34060	2020-04-27 15:02:59	coconing	其他	109年04月27日 三大法人買賣金額統計表	176	pttdata\M.1587970981.A.4FE.json
34099	2020-04-28 15:06:38	coconing	其他	109年04月28日 三大法人買賣金額統計表	93	pttdata\M.1588057621.A.37D.json
34177	2020-04-29 14:59:03	coconing	其他	109年04月29日 三大法人買賣金額統計表	115	pttdata\M.1588143546.A.3A4.json
34267	2020-04-30 14:59:56	coconing	其他	109年04月30日 三大法人買賣金額統計表	144	pttdata\M.1588229999.A.8B3.json

	time	author	type	title	score	filename
436	2018-02-12 08:30:37	vendan5566	閒聊	2018/02/12 盤中間聊	625	pttdata\M.1518395440.A.F2A.json
493	2018-02-14 08:30:26	vendan5566	閒聊	2018/02/14 盤中間聊	207	pttdata\M.1518568229.A.F53.json
589	2018-02-21 08:30:03	vendan5566	閒聊	2018/02/21 盤中間聊	587	pttdata\M.1519173005.A.F0D.json
627	2018-02-22 14:00:53	vendan5566	閒聊	2018/02/22 盤後閒聊	215	pttdata\M.1519279255.A.8A7.json
653	2018-02-23 08:30:03	vendan5566	閒聊	2018/02/23 盤中間聊	577	pttdata\M.1519345805.A.31B.json
...	...	...	...	...	...	...
32284	2020-03-27 13:39:43	vendan5566	心得	早知道今天這樣，我就	39	pttdata\M.1585287585.A.074.json
32877	2020-04-06 09:51:17	vendan5566	新聞	科技大廠第一家！鴻海宣布警報 景點員工在	34	pttdata\M.1586137879.A.66B.json
32984	2020-04-07 20:56:36	vendan5566	參選	vendan5566	138	pttdata\M.1586264198.A.862.json
33149	2020-04-10 10:38:43	vendan5566	新聞	中環3/24~4/9處分台灣高鐵股票 虧損829	31	pttdata\M.1586486325.A.5FE.json
34046	2020-04-27 10:02:49	vendan5566	標的	1227佳格	23	pttdata\M.1587952971.A.F9B.json

# 版上KOL

指標三：

扣除「新聞」及「其他」  
後，發文數最多

author	
vendan5566	264
hito21	175
justforsing	137
hrma	128
tim0259	114
epson5566	102
j2708180	100
zesonpso	90
Ejaculation	76
zakjudelo	61

文章數大幅減少，比起NLP方法  
較適合使用字典法

	time	author	type	title	score	filename
9641	2018-10-01 05:18:14	epson5566	標的	2330 台積電 空	39	pttdata\M.1538342296.A.937.json
9796	2018-10-04 00:59:41	epson5566	標的	9955 佳能多	7	pttdata\M.1538585983.A.AD6.json
9859	2018-10-05 01:52:46	epson5566	標的	2330 中性偏空	7	pttdata\M.1538675568.A.EEC.json
10000	2018-10-08 10:03:57	epson5566	標的	2330 台積電 多	54	pttdata\M.1538964240.A.424.json
10800	2018-10-23 14:40:30	epson5566	請益	中國的美元儲備是不是有點少 啊?	10	pttdata\M.1540276832.A.AA1.json
...	...	...	...	...	...	...
32746	2020-04-02 19:15:18	epson5566	心得	整理一些昨日美股收盤後的期權 倉位	11	pttdata\M.1585826120.A.4F4.json
32831	2020-04-05 00:34:56	epson5566	心得	週五美股REITs暴跌	63	pttdata\M.1586018098.A.22B.json
32838	2020-04-05 12:46:54	epson5566	心得	美國PPP計畫的混亂 可能會衝擊 小型企業	13	pttdata\M.1586062018.A.3C1.json
33512	2020-04-17 11:16:25	epson5566	心得	盤勢的一些看法	16	pttdata\M.1587093387.A.CB2.json
33930	2020-04-24 00:39:31	epson5566	心得	整理一些體質良好的偏股資料	120	pttdata\M.1587659973.A.D33.json

	time	author	type	title	score	filename
8104	2018-08-27 11:15:17	hito21	標的	元大滬深300正2	44	pttdata\M.1535339720.A.B5C.json
9278	2018-09-21 23:25:44	hito21	標的	長盛航	24	pttdata\M.1537543547.A.439.json
10252	2018-10-11 23:26:30	hito21	標的	元大黃金S&P正2	3	pttdata\M.1539271593.A.E72.json
10515	2018-10-17 16:37:56	hito21	標的	00672L 元大S&P原油正 2	8	pttdata\M.1539765479.A.FE0.json
10997	2018-10-26 13:23:53	hito21	標的	00669R+00671R 美國 反t	4	pttdata\M.1540531436.A.6A2.json
...	...	...	...	...	...	...
27593	2020-01-23 22:49:03	hito21	標的	道瓊03(UDH0) 月刊~	69	pttdata\M.1579790945.A.3A8.json
28790	2020-02-26 19:04:01	hito21	標的	道瓊03(UDH0) 月報~	106	pttdata\M.1582715043.A.F0E.json
31885	2020-03-23 22:59:55	hito21	標的	微型小道瓊06	51	pttdata\M.1584975599.A.E0D.json
32527	2020-03-30 10:25:00	hito21	標的	00637L 元大滬深300正 2	13	pttdata\M.1585535102.A.5A7.json
33898	2020-04-23 16:32:55	hito21	標的	微型小道瓊06	60	pttdata\M.1587630778.A.526.json

Epson5566的  
文章種類分配性質理想

type	
心得	85
標的	10
請益	7

Hito21 的  
標的文最多

type	
標的	174
標的 美國道瓊(*UDU9)	1

# 版上KOL

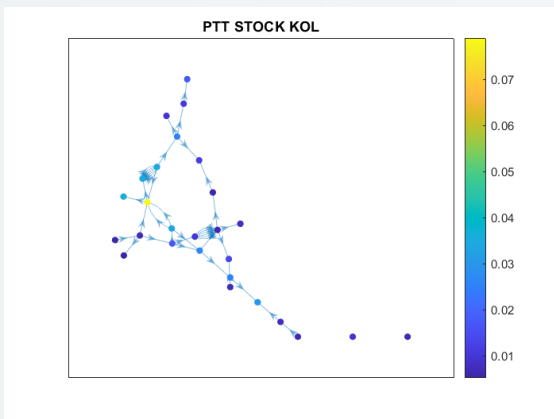
## 指標四：

利用PageRank概念

找出分數最高的人做KOL

(以「心得」與「標的」文發文者為主)

文章數大幅減少，比起NLP方法  
較適合使用字典法



1 Name	2 PageRank	3 InDegree	4 OutDegree
'zesonpso'	0.0790	813	2
'henin2003'	0.0362	60	0
'LittlePong'	0.0349	125	2
'zzz499'	0.0346	22	6
'civelant'	0.0344	501	3
'drakd4d'	0.0304	76	0
'finalwave'	0.0266	260	1
'jazz8010...	0.0249	42	1
'optimize'	0.0249	218	3
't200830...	0.0175	147	0

# 可以運用的資料子集

- Epson5566的心得文 (指標三)
- Hito21 的標的文 (指標三)
- Zesonpso的心得文 (指標四)
- 所有的心得文
- 通通一起跑





# 找尋贖回率指標


根據贖回率相關指標，才能接下來找尋有影響力的關鍵字，或是進行模型預測

## 贖回率指標建立步驟

1. 收集國內基金贖回率資料
2. 鎖定「野村」的「證券型」基金
3. 根據每個月結算資料做算術平均
4. 計算每月差異
5. 將大於上月的月份標出

```
eqt_fund_name = [  
    "野村環球基金",  
    "野村優質基金",  
    "野村台灣高股息基金",  
    "野村全球品牌基金",  
    "野村中國機會基金",  
    "野村歐洲中小成長基金",  
    "野村日本領先基金",  
    "野村e科技基金",  
    "野村高科技基金",  
    "野村中小基金",  
    "野村鴻運基金",  
    "野村台灣通籌基金",  
    "野村成長基金",  
    "野村積極成長基金",  
    "野村大俄羅斯基金",  
    "野村巴西基金",  
    "野村印尼潛力基金",  
    "野村印度潛力基金",  
    "野村泰國基金",  
    "野村新馬基金",  
    "野村全球高股息基金",  
    "野村亞太高股息基金"  
]
```

1



	ym	redemption	delta	signal
60	2019-11-30	0.040687	0.009933	True
61	2019-12-31	0.045125	0.004438	True
62	2020-01-31	0.063234	0.018109	True
63	2020-02-29	0.063234	0.000000	False
64	2020-03-31	0.114438	0.051203	True

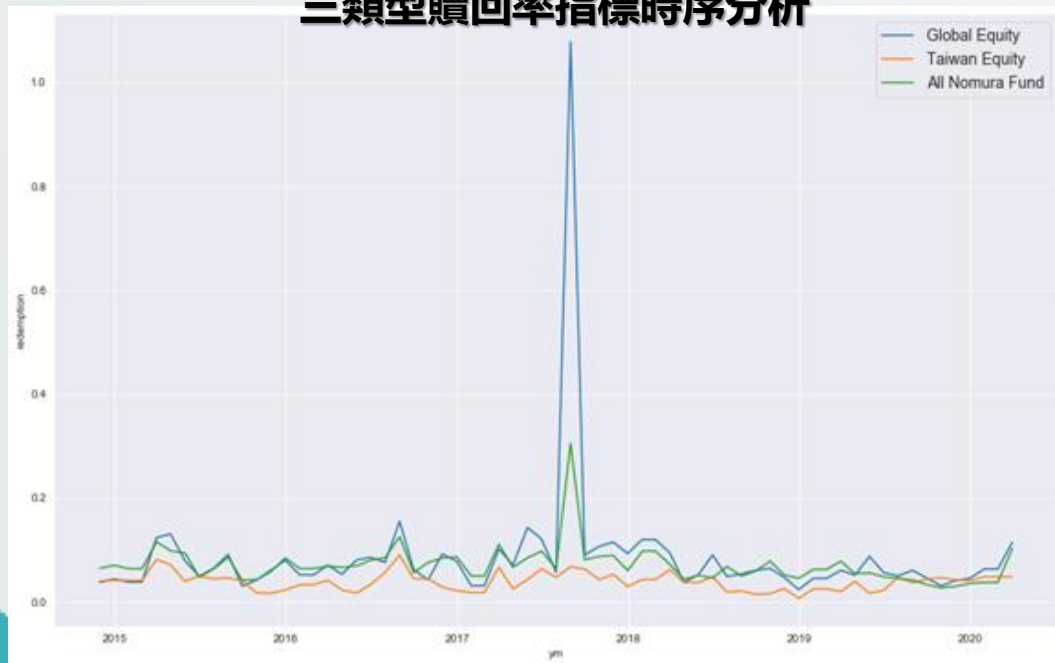
# 找尋贖回率指標

除了野村全部股票型基金，我們也使用同樣方法製作以下兩種資料的贖回率指標

1. 野村投信的「台灣股票型基金」
2. 野村投信的「全部基金」

## 三類型贖回率指標時序分析

```
eqt_fund_name = [  
    # "野村環球基金",  
    # "野村優質基金",  
    "野村台灣高股息基金",  
    # "野村全球品牌基金",  
    # "野村中國機會基金",  
    # "野村歐洲中小成長基金",  
    # "野村日本領先基金",  
    "野村e科技基金",  
    "野村高科技基金",  
    "野村中小基金",  
    "野村鴻運基金",  
    "野村台灣運籌基金",  
    "野村成長基金",  
    "野村積極成長基金",  
    # "野村大俄羅斯基金",  
    # "野村巴西基金",  
    # "野村印尼潛力基金",  
    # "野村印度潛力基金",  
]
```



可以看到大致上贖回率的走勢都滿接近的，除了在2017下半年，全球股票型基金贖回率指標有出現一個很大的跳點

在接下來的分析，我們先是使用指標一的全球股票型基金來分析，之後會探討改成台灣股票型基金的狀況。

# 研究方法

- 斷詞
  - 中研院-CKIP Tagger
- 找尋主題
  - 文件主題模型 - LDA(Latent Dirichlet Allocation)
- 找尋關鍵字
  - TF-IDF
  - 統計方法
- 預測是否影響贖回率
  - 字典+機器學習分類模型
  - NLP方法-Google BERT(Bidirectional Encoder Representations from Transformers)

# CKIP

使用CKIP的開源套件，將文章轉換成以下形式，  
並且使用Stopping Words字典，將無意義字詞去除

## 詞庫小組

CHINESE KNOWLEDGE AND INFORMATION PROCESSING

中研院資訊所、語言所於民國七十五年成立一個跨所合作的中文計算語言研究小組，共同合作建構中文自然語言處理的資源與研究環境，為國內外中文自然語言處理及其相關研究提供基本的研究資料與知識架構。代表性研究成果包括中文詞知識庫、語料庫及中文處理技術等。網際網路產生大量資訊，但缺乏有效的自動化分析方法及技術足以快速處理。為了達到智慧型的資訊處理，知識為本的訊息處理成為目前研究的核心焦點，本計劃進行五個主要研究方向：深度學習、知識表達、自然語言理解、知識擷取、聊天機器人。



\n大家戴口罩，保持社交距離。\\n\\n很多時尚奢侈品是不是都失去了炫富的意義？\\n\\n然後去全球化，反中，\\n\\n很多東西的生產效率都沒辦法達到疫情前的程度。\\n\\n在地化反脆弱，\\n\\n又多了很多安全性的額外成本。\\n\\n大家可能會比較腳踏實地的在乎生活實業，\\n\\n比較沒預算在形而上的精神層面。\\n\\n打算停業的大億麗緻也算壯士斷腕。\\n\\n紓困貸款，也只是讓業主越欠越多而已。\\n\\n一年的疫情損失，不知道正常的十年賺不賺得回來。\\n\\n除非是政府願意補助幾成的員工薪資，業主可能還願意賭一下。\\n\\n股市因為 QE 印鈔票，\\n\\n就很難講會上還會下了。公司不賺錢，但錢也不值錢。\\n\\n只是我覺得現金購買力，在一些基本生活用品上會降低。\\n\\n像食物會變貴，藥品會變貴甚至買不到。\\n\\n現在最值得的投資，大概只有健康跟家庭了。\\n\\n--\\n\\n手機的 google map 的探照燈方向就是你的正前方\\n\\n.....好像很多人都不知道\\n\\n--\\n\\n✖ 編輯: whkuo (61.231.200.192 臺灣), 05/01/2020 15:34:14\\n'



[ '戴', '口罩', '保持', '社交', '距離', '很多', '時尚', '奢侈品', '失去', '炫富', '意義', '全球化', '反中', '很多', '東西', '生產', '效率', '沒辦法', '達到', '疫情', '前', '程度', '地化', '反', '脆弱', '很多', '安全性', '額外', '成本', '可能', '會', '比較', '腳踏實地', '在乎', '生活', '實業', '比較', '沒', '預算', '形而上', '精神', '層面', '打算', '停業', '大億', '麗緻', '算', '壯士斷腕', '紓困', '貸款', '業主', '越', '欠', '越', '年', '疫情', '損失', '知道', '正常', '十', '年', '賺', '賺', '回來', '政府', '願意', '補助', '幾成', '員工', '薪資', '業主', '可能', '願意', '賭', '一下', '股市', 'QE', '印', '鈔票', '難', '講會', '會', '公司', '賺錢', '錢', '值錢', '覺得', '現金', '購買力', '基本', '生活', '用品', '會', '降低', '食物', '會', '變', '貴', '藥品', '會', '變', '貴', '買', '現在', '值得', '投資', '大概', '健康', '家庭', '手機', 'google map', '探照燈', '方向', '正前方', 'n', '...', '...', '好像', '很多', '知道', '編輯', 'whkuo' ]

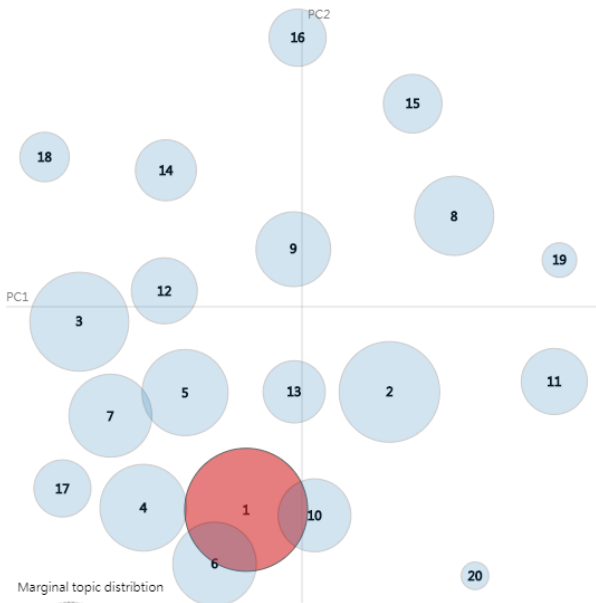
# LDA

屬於非監督式學習，透過生成模型的機制，在一系列文件中萃取出抽象的「主題」

右圖為使用全部資料下去做LDA的主題，可以看見**主題1**似乎和**中美貿易戰**很有關係

Selected Topic:  Previous Topic Next Topic Clear Topic

Intertopic Distance Map (via multidimensional scaling)



Marginal topic distribution

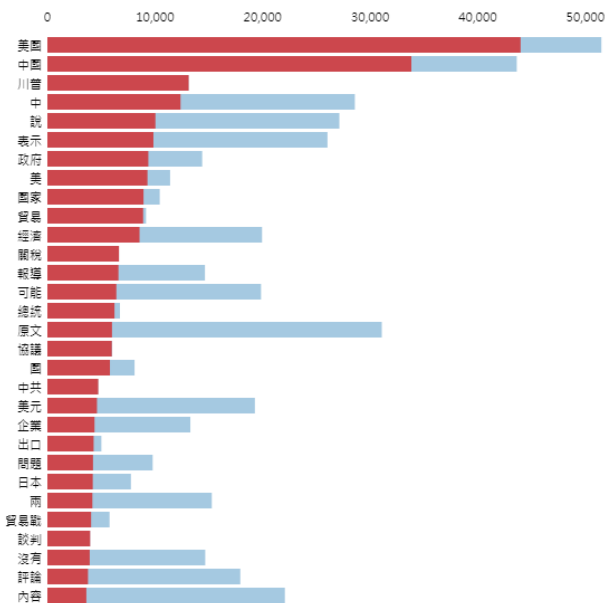


Slide to adjust relevance metric:(2)

$\lambda = 1$

0.0 0.2 0.4 0.6 0.8 1.0

Top-30 Most Relevant Terms for Topic 1 (13.6% of tokens)



Overall term frequency

Estimated term frequency within the selected topic

1. saliency(term w) = frequency(w) \* [sum\_t p(t | w) \* log(p(t | w)/p(t)) for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) =  $\lambda * p(w | t) + (1 - \lambda) * p(w | t)/p(w)$ ; see Sievert & Shirley (2014)

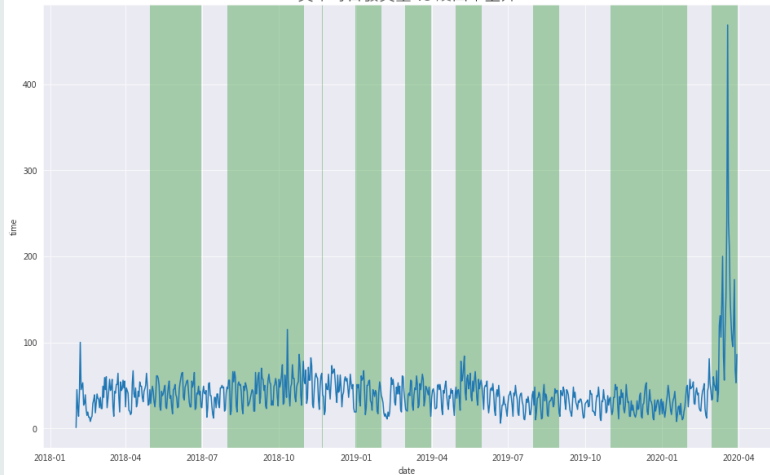
# 資料探勘

## 使用贖回率指標做資料探勘

```
signal  
False 14358  
True 18115
```

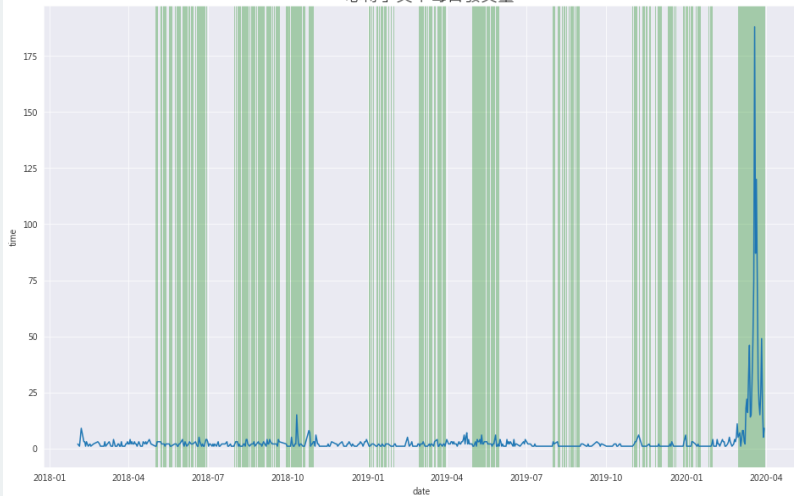
底色為贖回率增加的時間

文章每日發文量 vs 贖回率上升

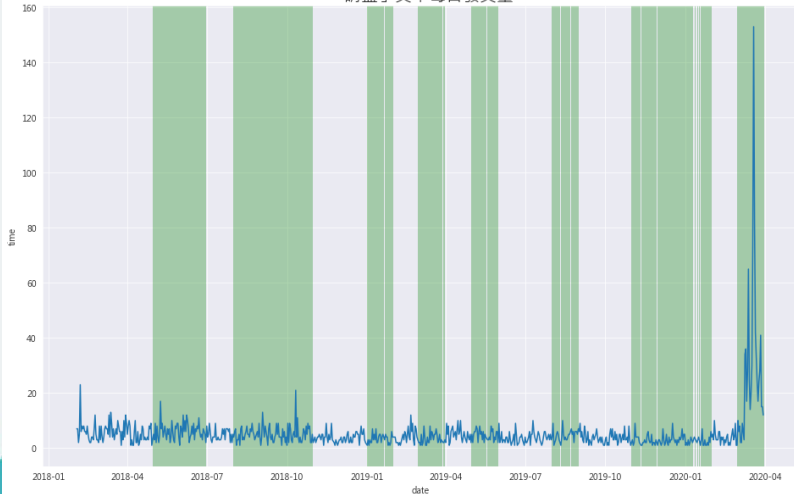


	type	signal	time
其他	False	1261	
	True	1621	
心得	False	526	
	True	1472	
投顧	False	36	
	True	50	
新聞	False	8161	
	True	9722	
標的	False	2323	
	True	2421	
請益	False	1661	
	True	2448	
閒聊	False	390	
	True	381	

「心得」文章每日發文量



「請益」文章每日發文量



# 找尋關鍵字

以贖回率指標回頭找有顯著差異的用字

以epson5566為例

```
In [7]: data = df[df['author']=='hito21']

In [8]: sub_data = data[['remainder_sentence', 'signal']]

In [117]: sub_data
```

	remainder_sentence	signal
8104	[-----]	True
9278	[-----]	True
10252	[標的, 元大, S&P, 黃金, 正, 2\n, 分類, 空, 請益, 心得, 短多, 不...	True
10515	[先, 損出, 趨勢, 轉, 大, 低點, 接, ~, 感恩, ..., 空, 低點, 接回...	True
10997	[獲利, 達標, 出場, ~, 感謝, ..., 標的, 00671, R+, 00669, ...]	True

```
In [7]: def find_key_word_table(sub_data, top_range=80, top_select=20):
        pos_dict = dict()
        neg_dict = dict()

        for seg, signal in zip(sub_data['remainder_sentence'], sub_data['signal']):
            if signal == True:
```

```
In [32]: reg_data
```

	中國	美	關稅	貿易	囉	川普	市場	日本	協議	...	沒	數據	news	說	盤整	進場	空手	破	建議	signal
4	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	True	False	False	False	1
8	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	True	False	False	False	1
52	False	False	False	False	False	False	False	False	True	False	...	True	False	False	False	True	False	False	False	1
15	False	False	True	False	False	False	False	False	False	...	False	False	False	False	False	True	False	False	False	1
37	False	False	True	False	False	False	False	False	False	...	True	False	False	False	False	True	False	False	True	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
58	False	True	False	False	True	True	True	True	True	...	False	False	False	False	False	False	False	False	False	1
33	False	False	False	False	False	True	True	False	True	False	...	False	False	False	False	False	False	False	False	1
30	False	True	False	False	False	True	True	True	True	False	...	True	False	False	True	False	True	False	True	0
85	True	True	True	False	False	False	False	False	True	False	...	True	True	False	True	False	True	False	False	1
27	False	True	True	False	False	False	False	False	False	False	...	True	False	False	False	True	False	False	False	1

## 方法一：

將限定的資料子集選出後，先篩選出在贖回率上升以及沒上升時各自的前80個用字。再鎖定各自前20個字，彼此對照是否有出現在對方80個字的空間中，如果有則刪去，以找出在各自情況下最具代表性的用字。



# 找尋關鍵字 TF-IDF

```
matrix.toarray().shape
```

```
(96, 2312)
```

以epson5566為例

```
from sklearn.feature_extraction.text import TfidfVectorizer

# corpus = ["welcome to stackoverflow my friend", "my friend"]
# corpus = np.array(data['remainder_sentence'].apply(lambda x: x + " "))
corpus
vectorizer = TfidfVectorizer()
matrix = vectorizer.fit_transform(corpus)
print(matrix)
```

```
(0, 489)    0.05607410220584747
(0, 139)    0.09324305250856824
(0, 1332)   0.05901854389058382
(0, 2058)   0.06068055359266021
(0, 1627)   0.06931394944093751
(0, 1900)   0.09324305250856824
(0, 898)    0.09324305250856824
(0, 1399)   0.06068055359266021
(0, 406)    0.07574092647648596
(0, 766)    0.0854982383607639
(0, 13)     0.048329288058043124
```

```
words = vectorizer.get_feature_names()
for i in range(len(corpus)):
    print('----Document {}----'.format(i))
    for j in range(len(words)):
        if matrix[i,j] > 1e-5:
            # print(words[j].encode('utf-8'), matrix[i,j])
            print(words[j], matrix[i,j])
```

```
----Document 94----
1月期 0.1774983562833317
3月期 0.1774983562833317
下降 0.13094093536633694
亞洲 0.16804189165917965
公債 0.12754885421911924
加上 0.2488165552262192
區域 0.16804189165917965
商品 0.1896898595785012
```

```
In [183]: for i in b.index:
          print(words[i])

          words[j]
```

```
5566
編輯
epson
2019
整理
118
166
數據
資產
利率
美國
資料
03
營收
負債
指數
```

## 方法二：

使用Scikit-Learn建立文章的TF-IDF矩陣，分數越高表示某個字在該篇文章中的重要性以及代表性越高，因此我們將各個字在各文章的分數加總，取出TF-IDF最高的前50個字當作標準。

```
a = pd.DataFrame(matrix.toarray()).sum()
b = a.sort_values(ascending=False)[:50]
```



# 找尋關鍵字

依照正負分別使用TF-IDF

以epson5566為例

落在	目前	長期	上市	分析	未來	30	機制	美元	signal
True	False	False	False	True	False	False	True	False	1
False	False	True	False	True	False	False	True	False	1
True	False	True	False	True	False	False	True	False	1
False	False	True	False	True	False	False	True	False	1
False	False	False	False	False	False	False	False	True	1
...	...	...	...	...	...	...	...	...	...
False	True	True	False	False	True	False	False	False	1
False	True	False	False	False	False	False	False	False	1
False	False	False	False	False	True	False	False	False	1
False	True	False	False	False	False	False	False	True	1
False	False	False	False	False	False	True	True	False	1

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	Logistic Regression	0.7643	0.7708	0.900	0.7600	0.8178	0.4858
1	Extreme Gradient Boosting	0.7476	0.8083	0.875	0.7467	0.8022	0.4485
2	Gradient Boosting Classifier	0.7452	0.7542	0.800	0.7683	0.7733	0.4680
3	K Neighbors Classifier	0.7024	0.6729	0.875	0.7271	0.7818	0.3235
4	Random Forest Classifier	0.6810	0.7521	0.600	0.8667	0.6752	0.4030
5	Extra Trees Classifier	0.6810	0.7167	0.650	0.8017	0.7083	0.3664
6	Ridge Classifier	0.6762	0.0000	0.800	0.6933	0.7352	0.2914
7	Ada Boost Classifier	0.6714	0.6042	0.775	0.7033	0.7316	0.2854
8	SVM - Linear Kernel	0.6429	0.0000	0.825	0.6438	0.7102	0.2389
9	Decision Tree Classifier	0.6381	0.6417	0.600	0.7517	0.6560	0.2822
10	Quadratic Discriminant Analysis	0.6262	0.5792	0.775	0.6588	0.6949	0.1608
11	Linear Discriminant Analysis	0.6119	0.5458	0.725	0.6650	0.6809	0.1609
12	Naïve Bayes	0.5095	0.5354	0.425	0.5950	0.4650	0.0654

## 方法三：

先依照label分類，再使用Scikit-Learn分別建立文章的TF-IDF矩陣，分數越高表示某個字在該篇文章中的重要性以及代表性較高。因此我們將各個字在各label下的分數加總，手動去除無意義和重複出現字詞後，各取出TF-IDF最高的前25個字當作標準。



# 找尋關鍵字

依照T-test顯著程度

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	Naive Bayes	0.8190	0.7917	0.900	0.8300	0.8579	0.6008
1	Ada Boost Classifier	0.7857	0.7833	0.875	0.8150	0.8334	0.5017
2	Gradient Boosting Classifier	0.7833	0.8521	0.900	0.7983	0.8406	0.4857
3	Logistic Regression	0.7690	0.7792	0.875	0.7867	0.8213	0.4909
4	Extra Trees Classifier	0.7690	0.8250	0.900	0.7717	0.8272	0.4777
5	Ridge Classifier	0.7381	0.0000	0.825	0.7650	0.7860	0.4403
6	SVM - Linear Kernel	0.7000	0.0000	0.725	0.6917	0.6933	0.4004
7	Decision Tree Classifier	0.6952	0.6542	0.825	0.7283	0.7645	0.3096
8	Random Forest Classifier	0.6905	0.7062	0.775	0.7667	0.7546	0.3003
9	K Neighbors Classifier	0.6690	0.6812	0.975	0.6543	0.7810	0.1864
10	Extreme Gradient Boosting	0.6500	0.7417	0.750	0.7050	0.7152	0.2523
11	Quadratic Discriminant Analysis	0.6381	0.6000	0.800	0.6971	0.7340	0.1957
12	Linear Discriminant Analysis	0.5786	0.6146	0.600	0.6983	0.6226	0.1373

## 方法四：

將限定的資料子集選出後，依照每個字為單位，統計在各篇文章出現的次數，並標註是正向文章還是負面文章，之後就可以將此兩個序列做2樣本T檢定。(用Y=1次數減去Y=0次數)得到T-ratio以後，我們再將T-ratio依照絕對值之後的值排序，找出影響力強烈的字詞。

以epson5566為例

	t-ratio	p-value	abs_ratio
關稅	-5.239830	1.644165e-07	5.239830
白宮	-4.408803	1.051665e-05	4.408803
盤	-4.312124	1.634414e-05	4.312124
買入	4.200072	2.694381e-05	4.200072
應該	3.944542	8.056790e-05	3.944542
美股盤	-3.912335	9.209019e-05	3.912335
協議	-3.811656	1.389724e-04	3.811656
美股	-3.804072	1.432920e-04	3.804072
指數	-3.786857	1.535711e-04	3.786857
評論	-3.774198	1.615685e-04	3.774198
新高	-3.657921	2.557412e-04	3.657921
發文	3.638000	2.763223e-04	3.638000
道瓊	-3.629329	2.857563e-04	3.629329
底部	3.615507	3.014217e-04	3.615507
ADR	-3.596216	3.246330e-04	3.596216
進口	-3.574429	3.528566e-04	3.574429
美	-3.528269	4.204013e-04	3.528269
指	-3.436644	5.916685e-04	3.436644
言論	-3.435087	5.950743e-04	3.435087
週	-3.254811	1.138931e-03	3.254811

# 預測資料集

```
In [32]:
```

		中國	美	關稅	貿易	囉	川普	市場	日本	協議	...	沒	數據	news	說	盤整	進場	空手	破	建議	signal
4		False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	True	False	False	False	1
8		False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	True	False	False	False	1
52		False	False	False	False	False	False	False	True	False	...	True	False	False	False	False	True	False	False	False	1
15		False	False	True	False	False	False	False	False	False	...	False	False	False	False	False	True	False	False	False	1
97		False	False	True	False	False	False	False	False	False	...	True	False	False	False	False	True	False	False	True	1
...		...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
68		False	True	False	False	True	True	True	True	True	...	False	False	False	False	False	False	False	False	False	1
93		False	False	False	False	False	True	True	False	True	...	False	False	False	False	False	False	False	False	False	1
90		False	True	False	False	False	True	True	True	True	...	True	False	False	True	False	True	False	False	True	0
85		True	True	True	False	False	False	False	True	False	...	True	True	False	True	False	True	False	False	False	1
27		False	True	True	False	False	False	False	False	False	...	True	False	False	False	False	True	False	False	False	1

篩選關鍵字之後，我們把個文章是否出現該文字以True跟False表示，並且刪除掉完全沒有出現關鍵字的文章，用以當作我們用來跑模型的Data。



# 分類模型集成

PYCARET

使用PyCaret可以輕鬆的比較各個分類模型的預測力，挑選出適當的模型。

An open source **low-code** machine learning library.



	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	Extra Trees Classifier	0.8381	0.8792	0.875	0.8883	0.8667	0.6613
1	Ridge Classifier	0.7952	0.0000	0.850	0.8550	0.8153	0.5759
2	Ada Boost Classifier	0.7929	0.8792	0.825	0.8517	0.8288	0.5701
3	Gradient Boosting Classifier	0.7643	0.8458	0.850	0.7800	0.8079	0.4954
4	Logistic Regression	0.7357	0.8167	0.875	0.7743	0.7975	0.4208
5	Decision Tree Classifier	0.7095	0.7125	0.700	0.8067	0.7316	0.4113
6	Linear Discriminant Analysis	0.7071	0.8250	0.700	0.8100	0.7259	0.4007
7	SVM - Linear Kernel	0.7024	0.0000	0.700	0.8350	0.7102	0.4128
8	Naive Bayes	0.6595	0.7958	0.575	0.8921	0.6392	0.3357
9	Random Forest Classifier	0.6595	0.7771	0.700	0.7150	0.7020	0.2959
10	Extreme Gradient Boosting	0.6476	0.7583	0.750	0.6905	0.7092	0.2512
11	Quadratic Discriminant Analysis	0.6024	0.5375	0.875	0.6314	0.7206	0.0678
12	K Neighbors Classifier	0.6000	0.6000	0.725	0.6833	0.6817	0.1304

# NLP-BERT

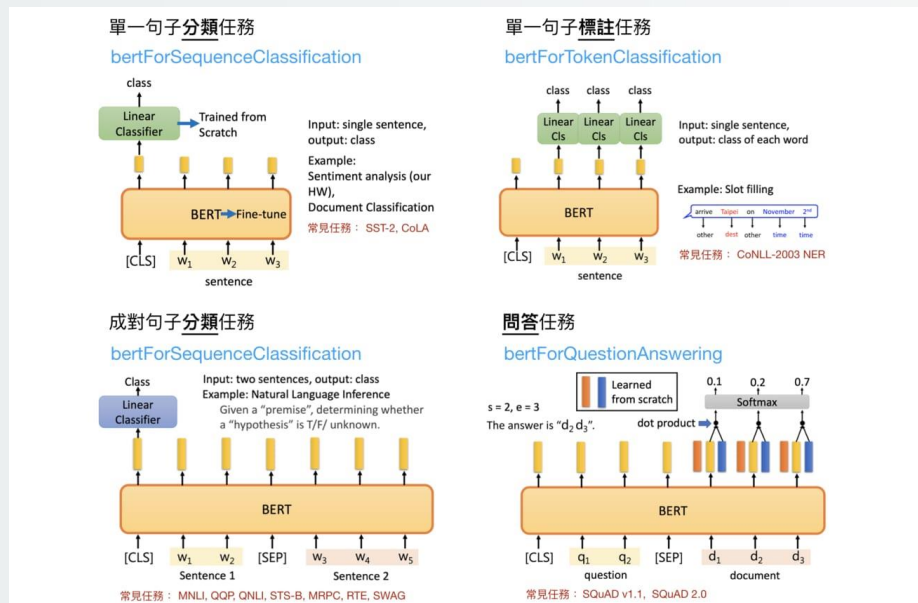
是 Google 以無監督的方式利用大量無標註文本「煉成」的語言代表模型

Attention，是編碼器和解碼器之間的接口。有了它，解碼器接收到的，就不僅僅是一個統一的向量表征了，還有來自編碼器每個時間步的向量表征

BERT 論文的作者們使用 Transformer Encoder、大量文本以及兩個預訓練目標，事先訓練好一個可以套用到多個 NLP 任務的 BERT 模型

遷移學習

站在巨人的肩膀上 省錢省力



圖片來源:李宏毅教授講解 BERT 的投影片

<https://ek21.com/news/tech/67141/>

[https://leemeng.tw/attack\\_on\\_bert\\_transfer\\_learning\\_in\\_nlp.html](https://leemeng.tw/attack_on_bert_transfer_learning_in_nlp.html)



# 模型結果

- 字典+機器學習分類模型
  - 關鍵字方法四效果最好
  - 關鍵字方法一效果次之
- NLP方法-Google BERT
  - 針對「心得」文章訓練效果最好
  - 針對「新聞」文章訓練效果最好



# 找尋關鍵字

## 方法一 以贖回率指標回頭找有顯著差異的用字

以epson5566為例

```
In [7]: data = df[df['author']=='hito21']
```

```
In [8]: sub_data = data[['remainder_sentence','signal']]
```

```
In [117]: sub_data
```

	remainder_sentence	signal
8104	[-----...]	True
9278	[-----...]	True
10252	[標的, 元大, S&P, 黃金, 正, 2\n, 分類, 空, 請益, 心得, 短多, 不...	True
10515	[先, 損出, 趨勢, 轉, 太, 低點, 接, ~, 感恩, ..., 空, 低點, 接回...	True
10997	[獲利, 達標, 出場, ~, 感謝, ..., 標的, 00671, R+, 00669, ...]	True
...	...	...
26968	[標的, 道瓊, 03, UDHO, 分類, 空, 請益, 心得, 請益, 分享, 夜盤, ...]	True
27593	[標的, 道瓊, 03, UDHO, or, 美國, 道瓊, 指數, ~, 分類, 空, ...]	True
28790	[標的, 道瓊, 03, UDHO, 月報, or, 美國, 道瓊, 指數, ~, 分...	False
31885	[標的, 微型, 道瓊, 06, 分類, 空, 請益, 心得, 請益, 分享, 美國, DJ...	True
32527	[標的, 00637, L, 元大滬, 深, 300, 正, 2\n, 分類, 空, 請益...	True

```
In [7]: def find_key_word_table(sub_data, top_range=80, top_select=20):  
        pos_dict = dict()  
        neg_dict = dict()  
  
        for seg, signal in zip(sub_data['remainder_sentence'], sub_data['signal']):  
            if signal == True:
```

```
In [32]: reg_data
```

	中國	美	關稅	貿易	囉	川普	市場	日本	協議	...	沒	數據	news	說	盤整	進場	空手	破	建議	signal
4	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	True	False	False	False	1
8	False	False	False	False	False	False	False	False	False	...	False	False	False	False	False	True	False	False	False	1
52	False	False	False	False	False	False	False	True	False	...	True	False	False	False	False	True	False	False	False	1
15	False	False	True	False	False	False	False	False	False	...	False	False	False	False	False	True	False	False	False	1
37	False	False	True	False	False	False	False	False	False	...	True	False	False	False	False	True	False	False	True	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
58	False	True	False	False	True	True	True	True	True	...	False	False	False	False	False	False	False	False	False	1
33	False	False	False	False	False	True	True	False	True	...	False	False	False	False	False	False	False	False	False	1
90	False	True	False	False	False	True	True	True	True	...	True	False	False	True	False	True	False	False	True	0
85	True	True	True	False	False	False	False	True	False	...	True	True	False	True	False	True	False	False	False	1
27	False	True	True	False	False	False	False	False	False	...	True	False	False	False	False	True	False	False	False	1

# 機器學習預測

## 方法一

```
In [111]: data = df[df['author']=='epson5566']  
reg_data = find_key_word_table(data, 50, 50)
```

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	Extra Trees Classifier	0.8381	0.8792	0.875	0.8883	0.8667	0.6613
1	Ridge Classifier	0.7952	0.0000	0.850	0.8550	0.8153	0.5759
2	Ada Boost Classifier	0.7929	0.8792	0.825	0.8517	0.8288	0.5701
3	Gradient Boosting Classifier	0.7643	0.8458	0.850	0.7800	0.8079	0.4954
4	Logistic Regression	0.7357	0.8167	0.875	0.7743	0.7975	0.4208
5	Decision Tree Classifier	0.7095	0.7125	0.700	0.8067	0.7316	0.4113
6	Linear Discriminant Analysis	0.7071	0.8250	0.700	0.8100	0.7259	0.4007
7	SVM - Linear Kernel	0.7024	0.0000	0.700	0.8350	0.7102	0.4128
8	Naive Bayes	0.6595	0.7958	0.575	0.8921	0.6392	0.3357
9	Random Forest Classifier	0.6595	0.7771	0.700	0.7150	0.7020	0.2959
10	Extreme Gradient Boosting	0.6476	0.7583	0.750	0.6905	0.7092	0.2512
11	Quadratic Discriminant Analysis	0.6024	0.5375	0.875	0.6314	0.7206	0.0678
12	K Neighbors Classifier	0.6000	0.6000	0.725	0.6833	0.6817	0.1304

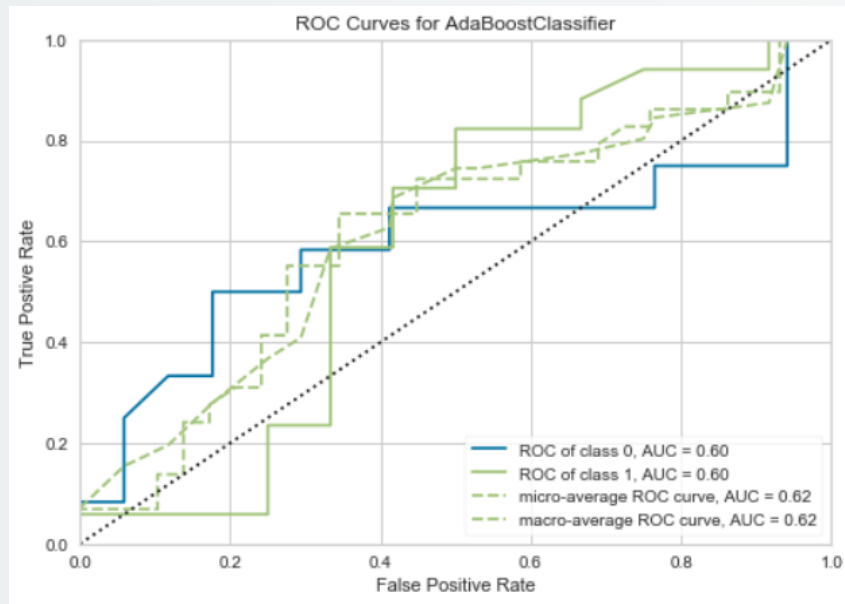
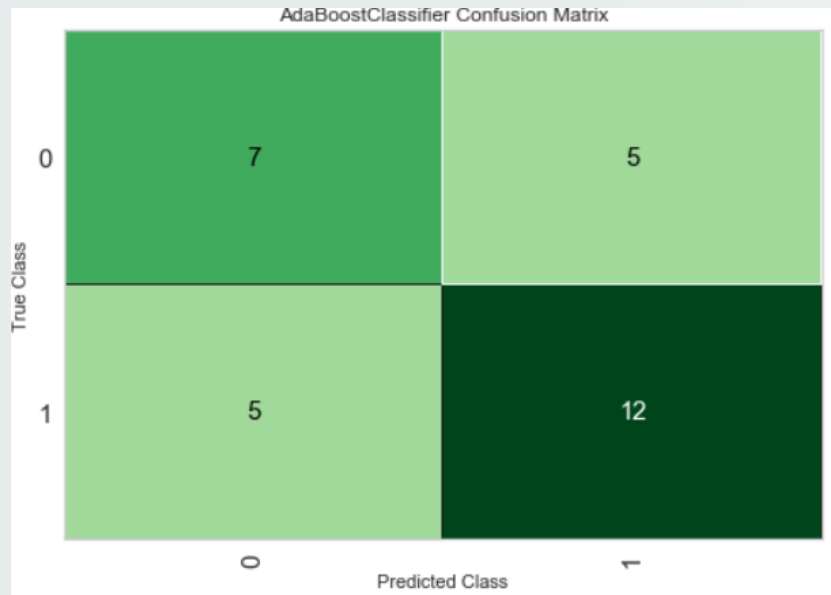
```
In [124]: ada = create_model('ada')
```

	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	0.5714	0.8333	0.7500	0.6000	0.6667	0.0870
1	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
2	0.7143	0.8333	1.0000	0.6667	0.8000	0.3636
3	0.7143	0.5833	0.7500	0.7500	0.7500	0.4167
4	0.7143	0.8333	0.7500	0.7500	0.7500	0.4167
5	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
6	0.7143	0.8333	0.7500	0.7500	0.7500	0.4167
7	0.8333	1.0000	0.7500	1.0000	0.8571	0.6667
8	0.8333	0.8750	0.7500	1.0000	0.8571	0.6667
9	0.8333	1.0000	0.7500	1.0000	0.8571	0.6667
Mean	0.7929	0.8792	0.8250	0.8517	0.8288	0.5701
SD	0.1282	0.1242	0.1146	0.1545	0.1033	0.2728



# 機器學習預測

## Adaboost相關指標



以個人的文章來預測的話，樣本數可能過少

# 找尋關鍵字

## 方法二 以TF-IDF找關鍵的用字

## 以epson5566為例

...	股票	已經	06	分析	認為	長期	影響	公司	殖利率	上市
...	False	False	False	True	False	True	False	False	False	False
...	False	False	False	True	False	True	False	False	False	False
...	False	False	False	True	False	True	False	False	False	False
...	False	False	False	True	False	True	True	False	False	False
...	False	False	False	True	False	True	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...
...	False	False	False	True	False	True	False	False	False	False
...	False	False	False	True	False	True	True	True	False	False
...	True	False	False	True	False	True	False	False	False	False
...	False	True	True	True	False	True	True	False	False	False
...	False	False	False	True	False	True	False	False	False	False

直接用TF-IDF而沒有做  
類似方法一的分類，預測  
可能效果較差

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	Extreme Gradient Boosting	0.6365	0.6563	0.6452	0.6582	0.6439	0.2743
1	Random Forest Classifier	0.6122	0.6125	0.5976	0.6470	0.6057	0.2264
2	Logistic Regression	0.5859	0.6457	0.5333	0.6279	0.5534	0.1798
3	K Neighbors Classifier	0.5788	0.5973	0.6286	0.5990	0.6041	0.1575
4	Extra Trees Classifier	0.5788	0.5666	0.5643	0.6333	0.5759	0.1600
5	Quadratic Discriminant Analysis	0.5776	0.5605	0.3881	0.6750	0.4800	0.1659
6	Ridge Classifier	0.5609	0.0000	0.5333	0.6052	0.5486	0.1298
7	Ada Boost Classifier	0.5532	0.5787	0.5643	0.5868	0.5614	0.1101
8	Decision Tree Classifier	0.5372	0.5452	0.5500	0.5921	0.5411	0.0789
9	Gradient Boosting Classifier	0.5372	0.5777	0.5190	0.5724	0.5249	0.0821
10	Naive Bayes	0.5365	0.5782	0.1952	0.7333	0.2895	0.0936
11	Linear Discriminant Analysis	0.5276	0.5679	0.4690	0.5939	0.4938	0.0670
12	SVM - Linear Kernel	0.4962	0.0000	0.6929	0.4786	0.5529	-0.0143

# 找尋關鍵字

## 方法三:依照正負分別使用TF-IDF

## 以epson5566為例

落在	目前	長期	上市	分析	未來	30	機制	美元	signal
True	False	False	False	True	False	False	True	False	1
False	False	True	False	True	False	False	True	False	1
True	False	True	False	True	False	False	True	False	1
False	False	True	False	True	False	False	True	False	1
False	False	False	False	False	False	False	False	True	1
...	...	...	...	...	...	...	...	...	...
False	True	True	False	False	True	False	False	False	1
False	True	False	False	False	False	False	False	False	1
False	False	False	False	False	True	False	False	False	1
False	True	False	False	False	False	False	False	True	1
False	False	False	False	False	False	True	True	False	1

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	Logistic Regression	0.7643	0.7708	0.900	0.7600	0.8178	0.4858
1	Extreme Gradient Boosting	0.7476	0.8083	0.875	0.7467	0.8022	0.4485
2	Gradient Boosting Classifier	0.7452	0.7542	0.800	0.7683	0.7733	0.4680
3	K Neighbors Classifier	0.7024	0.6729	0.875	0.7271	0.7818	0.3235
4	Random Forest Classifier	0.6810	0.7521	0.600	0.8667	0.6752	0.4030
5	Extra Trees Classifier	0.6810	0.7167	0.650	0.8017	0.7083	0.3664
6	Ridge Classifier	0.6762	0.0000	0.800	0.6933	0.7352	0.2914
7	Ada Boost Classifier	0.6714	0.6042	0.775	0.7033	0.7316	0.2854
8	SVM - Linear Kernel	0.6429	0.0000	0.825	0.6438	0.7102	0.2389
9	Decision Tree Classifier	0.6381	0.6417	0.600	0.7517	0.6560	0.2822
10	Quadratic Discriminant Analysis	0.6262	0.5792	0.775	0.6588	0.6949	0.1608
11	Linear Discriminant Analysis	0.6119	0.5458	0.725	0.6650	0.6809	0.1609
12	Naive Bayes	0.5095	0.5354	0.425	0.5950	0.4650	0.0654

	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	0.7143	0.8333	1.0000	0.6667	0.8000	0.3636
1	0.1429	0.1667	0.2500	0.2500	0.2500	-0.7500
2	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
3	0.7143	0.4167	1.0000	0.6667	0.8000	0.3636
4	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
5	0.8571	1.0000	1.0000	0.8000	0.8889	0.6957
6	0.7143	0.6667	1.0000	0.6667	0.8000	0.3636
7	0.6667	0.8750	0.7500	0.7500	0.7500	0.2500
8	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
9	0.8333	0.7500	1.0000	0.8000	0.8889	0.5714
Mean	0.7643	0.7708	0.9000	0.7600	0.8178	0.4858
SD	0.2415	0.2695	0.2291	0.2150	0.2092	0.4972

預測效果較穩定，且用計量學理常見的Logit就可以得到滿好的AUC預測結果，第二名則是機器學習競賽常見的XGBoost

# 找尋關鍵字

## 方法四:使用T-test顯著性篩選關鍵字

## 以epson5566為例

	t-ratio	p-value	abs_ratio
關稅	-5.239830	1.644165e-07	5.239830
白宮	-4.408803	1.051665e-05	4.408803
盤	-4.312124	1.634414e-05	4.312124
買入	4.200072	2.694381e-05	4.200072
應該	3.944542	8.056790e-05	3.944542
美股盤	-3.912335	9.209019e-05	3.912335
協議	-3.811656	1.389724e-04	3.811656
美股	-3.804072	1.432920e-04	3.804072
指數	-3.786857	1.535711e-04	3.786857
評論	-3.774198	1.615685e-04	3.774198
新高	-3.657921	2.557412e-04	3.657921
發文	3.638000	2.763223e-04	3.638000
道瓊	-3.629329	2.857563e-04	3.629329
底部	3.615507	3.014217e-04	3.615507
ADR	-3.596216	3.246330e-04	3.596216
進口	-3.574429	3.528566e-04	3.574429
美	-3.528269	4.204013e-04	3.528269
指	-3.436644	5.916685e-04	3.436644
言論	-3.435087	5.950743e-04	3.435087
週	-3.254811	1.138931e-03	3.254811

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	Naive Bayes	0.8190	0.7917	0.900	0.8300	0.8579	0.6008
1	Ada Boost Classifier	0.7857	0.7833	0.875	0.8150	0.8334	0.5017
2	Gradient Boosting Classifier	0.7833	0.8521	0.900	0.7983	0.8406	0.4857
3	Logistic Regression	0.7690	0.7792	0.875	0.7867	0.8213	0.4909
4	Extra Trees Classifier	0.7690	0.8250	0.900	0.7717	0.8272	0.4777
5	Ridge Classifier	0.7381	0.0000	0.825	0.7650	0.7860	0.4403
6	SVM - Linear Kernel	0.7000	0.0000	0.725	0.6917	0.6933	0.4004
7	Decision Tree Classifier	0.6952	0.6542	0.825	0.7283	0.7645	0.3096
8	Random Forest Classifier	0.6905	0.7062	0.775	0.7667	0.7546	0.3003
9	K Neighbors Classifier	0.6690	0.6812	0.975	0.6543	0.7810	0.1864
10	Extreme Gradient Boosting	0.6500	0.7417	0.750	0.7050	0.7152	0.2523
11	Quadratic Discriminant Analysis	0.6381	0.6000	0.800	0.6971	0.7340	0.1957
12	Linear Discriminant Analysis	0.5786	0.6146	0.600	0.6983	0.6226	0.1373

```
lr = create_model("lr")
```

	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	0.5714	1.0000	0.2500	1.0000	0.4000	0.2222
1	0.7143	0.8333	0.7500	0.7500	0.7500	0.4167
2	0.8571	0.9167	0.7500	1.0000	0.8571	0.7200
3	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
4	0.7143	1.0000	0.5000	1.0000	0.6667	0.4615
5	0.8571	1.0000	1.0000	0.8000	0.8889	0.6957
6	0.5714	0.7500	0.7500	0.6000	0.6667	0.0870
7	0.7143	0.6667	1.0000	0.6667	0.8000	0.3636
8	0.5714	0.5833	0.7500	0.6000	0.6667	0.0870
9	0.5000	0.2222	0.6667	0.5000	0.5714	0.0000
Mean	0.7071	0.7972	0.7417	0.7917	0.7267	0.4054
SD	0.1514	0.2399	0.2250	0.1870	0.1630	0.3070

效果和方法一差不多好，但是標準差較高，  
不過比較能有一致性的判斷標準

# 字典法小結

- 方法一、四的效果預測效果還不錯
- 方法二沒有依照Y的類別做分類，而是直接把TF-IDF高的字挑出來訓練，預測效果直觀的發現的確是不好。
- 方法三將方法一和二兩者結合，但是效果卻沒有直接使用方法一好。針對Epson5566的時候，雖然AUC也很高，但是經過K-fold交叉驗證後，預測的標準差較大。而針對心得文子集時，也有一些預測力不足的問題。
- 而方法四雖然沒有方法一好，但其實兩者相去不遠，而且也比較不需要人工調校參數

**針對PTT的KOL個人發文關鍵字篩選的最佳實務應該是：**  
**第四種的T-test方法**



# BERT

- 1735s 121ms/step - loss: 0.6964 - sparse\_categorical\_accuracy: 0.5139 - val\_loss: 0.7155 - val\_sparse\_categorical\_accuracy: 0.4129  
- 1733s 121ms/step - loss: 0.6900 - sparse\_categorical\_accuracy: 0.5321 - val\_loss: 0.6451 - val\_sparse\_categorical\_accuracy: 0.7137  
- 1735s 121ms/step - loss: 0.6828 - sparse\_categorical\_accuracy: 0.5527 - val\_loss: 0.6693 - val\_sparse\_categorical\_accuracy: 0.6153

以“新聞”做資料子集

以“閒聊”做資料子集

```
model.fit(  
    X,  
    y,  
    epochs=EPOCHS,  
    batch_size=BATCH_SIZE,  
    validation_split=0.2  
)
```

16 samples, validate on 155 samples

=====] - 80s 130ms/step - loss: 0.7818 - sparse\_categorical\_accuracy: 0.4919 - val\_loss: 0.7114 - val\_sparse\_categorical\_accuracy: 0.5032  
=====] - 75s 122ms/step - loss: 0.6365 - sparse\_categorical\_accuracy: 0.6429 - val\_loss: 0.8705 - val\_sparse\_categorical\_accuracy: 0.3742

以“心得”做資料子集

```
model.fit(  
    X,  
    y,  
    epochs=EPOCHS,  
    batch_size=BATCH_SIZE,  
    validation_split=0.2  
)
```

samples, validate on 400 samples

=====] - 194s 121ms/step - loss: 0.6636 - sparse\_categorical\_accuracy: 0.6302 - val\_loss: 0.3885 - val\_sparse\_categorical\_accuracy: 0.9425  
=====] - 194s 121ms/step - loss: 0.5775 - sparse\_categorical\_accuracy: 0.6871 - val\_loss: 0.3808 - val\_sparse\_categorical\_accuracy: 0.8925

選對資料子集預測力有顯著提升





# K O L 部分

Epson5566

```
model.fit(  
    X,  
    y,  
    epochs=EPOCHS,  
    batch_size=BATCH_SIZE,  
    validation_split=0.2  
)  
  
76 samples, validate on 20 samples  
/2  
=====] - 10s 131ms/step - loss: 0.6643 - sparse_categorical_accuracy: 0.5526 - val_loss: 0.7842 - val_sparse_categorical_accuracy: 0.3000  
/2  
=====] - 10s 132ms/step - loss: 0.6210 - sparse_categorical_accuracy: 0.6842 - val_loss: 0.6811 - val_sparse_categorical_accuracy: 0.5500  
callbacks.CallbackList: History at 0:764881s0fb00>
```



Hito21

```
model.fit(  
    X,  
    y,  
    epochs=EPOCHS,  
    batch_size=BATCH_SIZE,  
    validation_split=0.2  
)  
  
89 samples, validate on 35 samples  
  
=====] - 18s 129ms/step - loss: 0.7128 - sparse_categorical_accuracy: 0.5180 - val_loss: 0.9952 - val_sparse_categorical_accuracy: 0.0857  
=====] - 18s 126ms/step - loss: 0.6543 - sparse_categorical_accuracy: 0.6187 - val_loss: 1.0614 - val_sparse_categorical_accuracy: 0.0857  
callbacks.CallbackList: History at 0:764881s0fb00>
```



# BERT小結

- 使用BERT來進行KOL講話的預測效果不太好，原因可能是因為資料筆數少，或是選到的KOL不夠有解釋力，以及個人的習慣用字可能也會影響到訓練的結果。
- 反觀如果使用以「心得」集結各家說法，得出來的預測力較高，並且可以看到預測效果遠高於「閒聊」。
- 相比於每天高頻的日常文章，心得文的發文頻率較低，和每月公布一次的贖回率的對應相性可能比較好

**使用BERT進行PTT預測的最佳實務是：**  
**針對同類型文章跨作者來進行訓練及預測**







# 強基金論壇分析



# 研究方法

- 除了 PTT 部分的研究方法，由於強基金論壇是我們更重視的資料來源，因此我們除了沿用在 PTT 的方法之外，也加入了另一種方法。
  - 目的
    - 實際連結詞彙與買賣超的關係，並量化出每個詞彙對於贖回情形的影響，進而瞭解各文章屬於買、賣超的情緒傾向
    - 瞭解各論壇在輿論所扮演的腳色，區分在預測未來市場時各論壇所隱含的資訊
      - 例如如果發現強基金論壇的文章比較偏向投資標的的討論（而且容易在），而 PTT 的發文比較是屬於情緒發洩為主（較容易察覺出負面情緒）。則假設目前預測未來市場為看多，則應該以監測強基金的輿論為主，能夠比較精準地掌握大眾目前的投資意向。
  - 資料
    - 以建議的贖回率的替代變數：0050 每日買賣超作為研究的主要變數。也希望藉由比較精細的日資料可以獲得比較精準並細緻的結果



# 研究流程

- 訓練集與測試集的切分以 2020 年為準（2020 年以前的資料會被歸類為訓練集，2020 年以後的資料會被歸類為測試集）
  - 為避免模型學習到「事件」，而不是依據長期、持續出現的「情緒詞」做為買賣超情形的判斷
  - 例如測試集的選擇為完全隨機，「肺炎」將會連結到市場的賣超情形，不過未來「肺炎」再次主導市場的可能性不一定很大，但目前會讓我們模型的結果過度樂觀

以 0050 的「一般戶買賣超情形」做為基金贖回率的替代指標，並定義每日為買超日或賣超日

初步找出「訓練資料集」中，高頻在買超日、賣超日出現的詞彙（並以 **proportion t-test**、**OLS test** 分別進行統計顯著性的驗證）

根據各詞彙影響買賣超的程度，綜合考量各詞彙 **TF-IDF** 之權重，為每個詞彙定義出【買超分數】以及【賣超分數】以評估各個詞彙傾向買超行為以及賣超行為的情形

將「訓練集」計算出各詞彙的買超、賣超分數套用在「驗證集」上，並以之計算出各文章的【買超分數】以及【賣超分數】來代表該文章的情形分數，並進行後續的因果關聯驗證（**OLS**、**Quantile Regression**）



# 研究發現



# 研究發現

- PTT部分

- 如果要針對特定KOL分析，使用字典法效果較好
- 在我們採用的方法中，使用BERT對「心得」文章訓練效果最好

- 強基金部分

- 計量方法部分

- 【買超分數】其實較能夠較好地描述市場情形
    - 【賣超分數】的表現則較差

- 機器學習方法部分

- 用字典法和NLP表現都不如計量方法理想

# PTT部分



# 穩健性測試

延續前面PTT的結果，右圖為對「epson5566」使用T-test尋找關鍵字的結果，我們想測試如果在不同的資料集，或是不同的贖回率指標能不能一樣有良好的預測結果。

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	Naive Bayes	0.8190	0.7917	0.900	0.8300	0.8579	0.6008
1	Ada Boost Classifier	0.7857	0.7833	0.875	0.8150	0.8334	0.5017
2	Gradient Boosting Classifier	0.7833	0.8521	0.900	0.7983	0.8406	0.4857
3	Logistic Regression	0.7690	0.7792	0.875	0.7867	0.8213	0.4909
4	Extra Trees Classifier	0.7690	0.8250	0.900	0.7717	0.8272	0.4777
5	Ridge Classifier	0.7381	0.0000	0.825	0.7650	0.7860	0.4403
6	SVM - Linear Kernel	0.7000	0.0000	0.725	0.6917	0.6933	0.4004
7	Decision Tree Classifier	0.6952	0.6542	0.825	0.7283	0.7645	0.3096
8	Random Forest Classifier	0.6905	0.7062	0.775	0.7667	0.7546	0.3003
9	K Neighbors Classifier	0.6690	0.6812	0.975	0.6543	0.7810	0.1864
10	Extreme Gradient Boosting	0.6500	0.7417	0.750	0.7050	0.7152	0.2523
11	Quadratic Discriminant Analysis	0.6381	0.6000	0.800	0.6971	0.7340	0.1957
12	Linear Discriminant Analysis	0.5786	0.6146	0.600	0.6983	0.6226	0.1373

## 不同資料集 - Zesonpso的文章

這位是用Page Rank方法選出來的KOL，我們用此資料集依照相同方法論實作模型訓練。

可以看到雖然AUC沒有前一位KOL高，但是也仍有6.5成以上的AUC

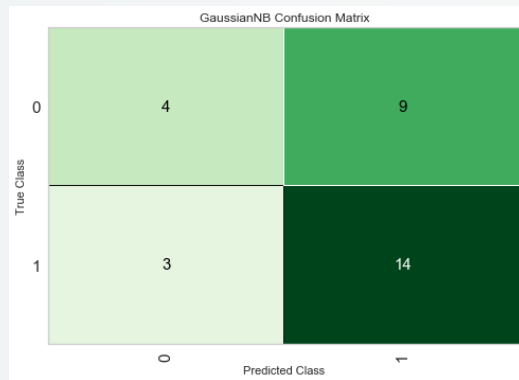
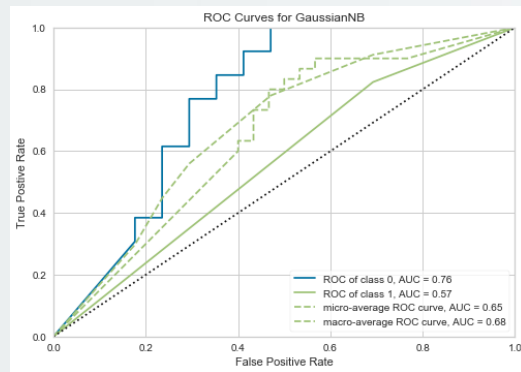
	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	Naive Bayes	0.7119	0.6667	1.0000	0.6693	0.8001	0.3544
1	Logistic Regression	0.6524	0.6889	0.6917	0.7033	0.6849	0.2933
2	Gradient Boosting Classifier	0.6381	0.6306	0.7417	0.6567	0.6944	0.2486
3	Extra Trees Classifier	0.6381	0.6667	0.7000	0.6583	0.6594	0.2736
4	Ada Boost Classifier	0.6143	0.6917	0.6500	0.6567	0.6460	0.2199
5	Random Forest Classifier	0.6095	0.7264	0.5917	0.6850	0.6226	0.2208
6	K Neighbors Classifier	0.5952	0.6139	0.7167	0.6100	0.6513	0.1610
7	Decision Tree Classifier	0.5952	0.5875	0.6750	0.6317	0.6417	0.1760
8	Linear Discriminant Analysis	0.5833	0.6583	0.6667	0.6450	0.6417	0.1302
9	SVM - Linear Kernel	0.5810	0.0000	0.6917	0.6143	0.6320	0.1279
10	Ridge Classifier	0.5667	0.0000	0.5917	0.6067	0.5913	0.1299
11	Quadratic Discriminant Analysis	0.5476	0.5458	0.5250	0.5817	0.5377	0.0911
12	Extreme Gradient Boosting	0.5238	0.5361	0.5667	0.5750	0.5566	0.0352



# Naïve Bayes

對Naïve Bayes模型做10-fold

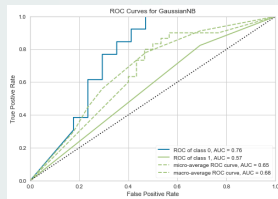
	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	0.7143	0.6667	1.0	0.6667	0.8000	0.3636
1	0.7143	0.6667	1.0	0.6667	0.8000	0.3636
2	0.7143	0.6667	1.0	0.6667	0.8000	0.3636
3	0.7143	0.6667	1.0	0.6667	0.8000	0.3636
4	0.7143	0.6667	1.0	0.6667	0.8000	0.3636
5	0.5714	0.5000	1.0	0.5714	0.7273	0.0000
6	0.7143	0.6667	1.0	0.6667	0.8000	0.3636
7	0.8571	0.8333	1.0	0.8000	0.8889	0.6957
8	0.5714	0.5000	1.0	0.5714	0.7273	0.0000
9	0.8333	0.8333	1.0	0.7500	0.8571	0.6667
Mean	0.7119	0.6667	1.0	0.6693	0.8001	0.3544
SD	0.0868	0.1054	0.0	0.0654	0.0466	0.2158



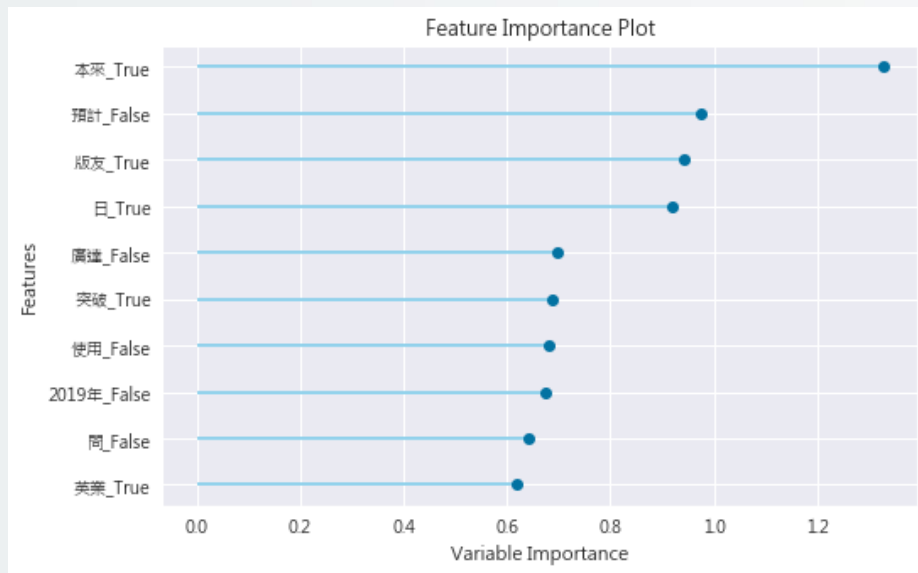
- 可以看到AUC仍有不錯的表現，但會遭遇到樣本資料太少的問題
- 因此雖然Zesonpso的發文回響熱烈，但是文章數量本身較少，難以當作預測的主力KOL

# Logit

對Logit模型做10-fold



	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	0.7143	0.5833	0.5000	1.0000	0.6667	0.4615
1	0.5714	0.6667	0.7500	0.6000	0.6667	0.0870
2	0.8571	1.0000	1.0000	0.8000	0.8889	0.6957
3	0.4286	0.5000	0.5000	0.5000	0.5000	-0.1667
4	0.7143	0.8333	0.7500	0.7500	0.7500	0.4167
5	0.4286	0.4167	0.5000	0.5000	0.5000	-0.1667
6	0.7143	0.7500	0.7500	0.7500	0.7500	0.4167
7	0.5714	0.5000	0.5000	0.6667	0.5714	0.1600
8	0.8571	0.7500	1.0000	0.8000	0.8889	0.6957
9	0.6667	0.8889	0.6667	0.6667	0.6667	0.3333
Mean	0.6524	0.6889	0.6917	0.7033	0.6849	0.2933
SD	0.1445	0.1795	0.1865	0.1439	0.1319	0.2946



- 可以看到AUC仍有不錯的表現，但和Naïve Bayes一樣會遭遇到樣本資料太少的問題
- 這邊的Features的資料集較少，所以也選出比較難直觀解釋的詞

# 更改預測的贖回率 - 野村台灣型股票型基金贖回率

這是一樣使用epson5566  
的心得文的狀況，並且也  
一樣使用T-test尋找關鍵字。

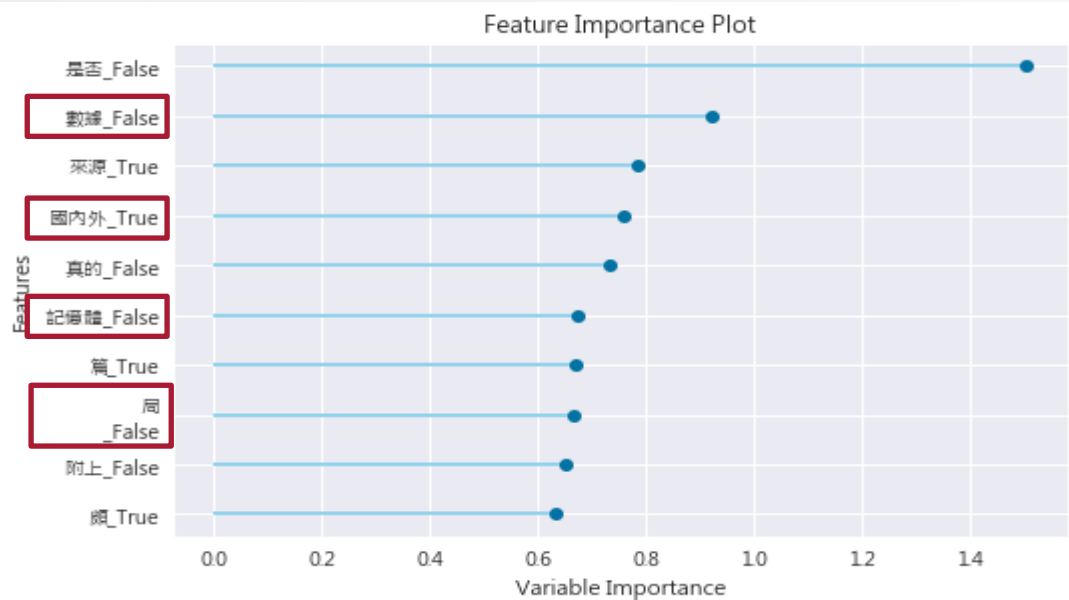
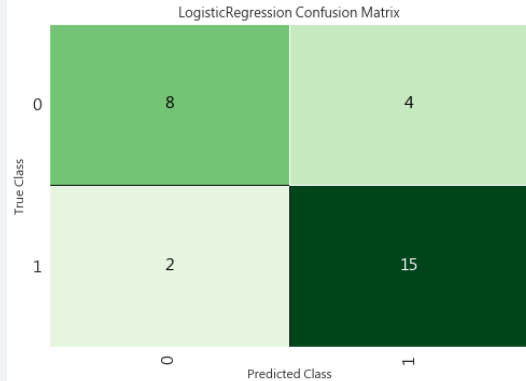
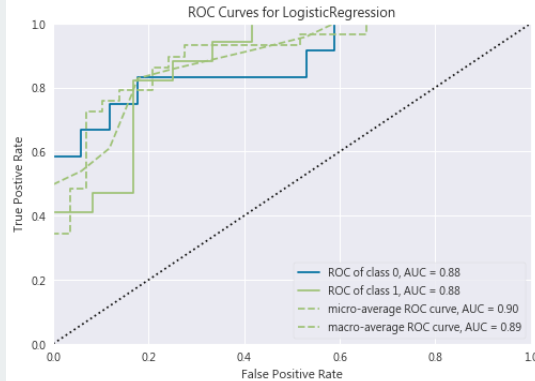
	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	Naive Bayes	0.8190	0.7917	0.900	0.8300	0.8579	0.6008
1	Ada Boost Classifier	0.7857	0.7833	0.875	0.8150	0.8334	0.5017
2	Gradient Boosting Classifier	0.7833	0.8521	0.900	0.7983	0.8406	0.4857
3	Logistic Regression	0.7690	0.7792	0.875	0.7867	0.8213	0.4909
4	Extra Trees Classifier	0.7690	0.8250	0.900	0.7717	0.8272	0.4777
5	Ridge Classifier	0.7381	0.0000	0.825	0.7650	0.7860	0.4403
6	SVM - Linear Kernel	0.7000	0.0000	0.725	0.6917	0.6933	0.4004
7	Decision Tree Classifier	0.6952	0.6542	0.825	0.7283	0.7645	0.3096
8	Random Forest Classifier	0.6905	0.7062	0.775	0.7667	0.7546	0.3003
9	K Neighbors Classifier	0.6690	0.6812	0.975	0.6543	0.7810	0.1864
10	Extreme Gradient Boosting	0.6500	0.7417	0.750	0.7050	0.7152	0.2523
11	Quadratic Discriminant Analysis	0.6381	0.6000	0.800	0.6971	0.7340	0.1957
12	Linear Discriminant Analysis	0.5786	0.6146	0.600	0.6983	0.6226	0.1373

# Logit

對Logit模型做10-fold

```
lr = create_model('lr')
```

	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	0.5714	1.0000	0.2500	1.0000	0.4000	0.2222
1	0.7143	0.8333	0.7500	0.7500	0.7500	0.4167
2	0.8571	0.9167	0.7500	1.0000	0.8571	0.7200
3	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
4	0.7143	1.0000	0.5000	1.0000	0.6667	0.4615
5	0.8571	1.0000	1.0000	0.8000	0.8889	0.6957
6	0.5714	0.7500	0.7500	0.6000	0.6667	0.0870
7	0.7143	0.6667	1.0000	0.6667	0.8000	0.3636
8	0.5714	0.5833	0.7500	0.6000	0.6667	0.0870
9	0.5000	0.2222	0.6667	0.5000	0.5714	0.0000
Mean	0.7071	0.7972	0.7417	0.7917	0.7267	0.4054
SD	0.1514	0.2399	0.2250	0.1870	0.1630	0.3070

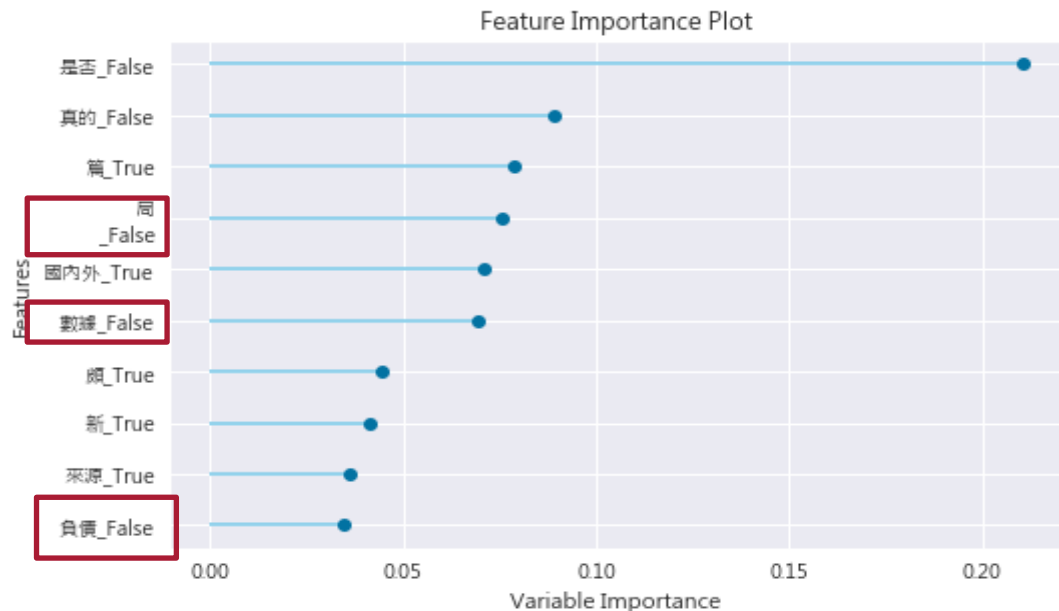
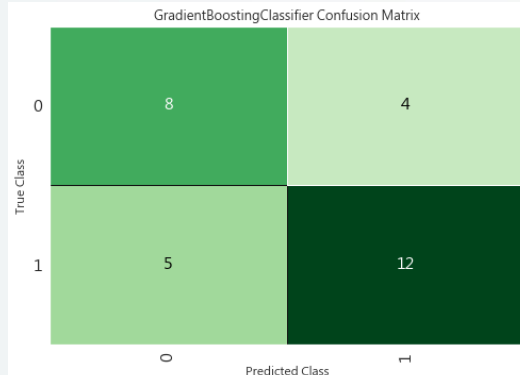
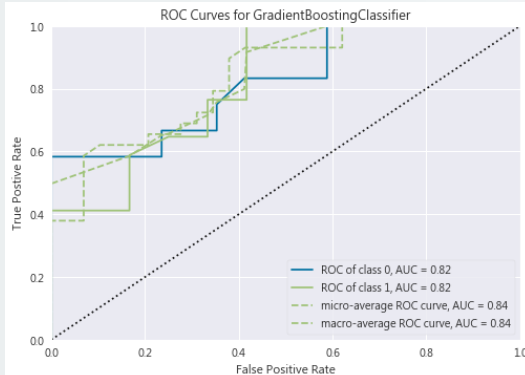


# Gradient Boosting

對GB模型做10-fold

```
gbc = create_model(gbc)
```

	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	0.8571	1.0000	1.0000	0.8000	0.8889	0.6957
1	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
2	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
3	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
4	0.7143	0.8333	1.0000	0.6667	0.8000	0.3636
5	0.7143	0.9167	0.7500	0.7500	0.7500	0.4167
6	0.7143	0.8333	0.7500	0.7500	0.7500	0.4167
7	0.5000	0.4375	0.7500	0.6000	0.6667	-0.2857
8	0.6667	0.7500	0.7500	0.7500	0.7500	0.2500
9	0.6667	0.7500	1.0000	0.6667	0.8000	0.0000
Mean	0.7833	0.8521	0.9000	0.7983	0.8406	0.4857
SD	0.1637	0.1689	0.1225	0.1425	0.1169	0.4182



# 不同資料集 - 整體心得文

	交易	操作	獲利	分析	比較	其實	經濟	最近	反彈	美股	疫情	公園	一直	停損	看到	存股	signal
25	False	False	False	False	False	False	False	False	False	False	False	False	False	False	True	False	0
32	False	False	True	False	False	False	False	False	True	False	False	False	False	False	False	False	0
77	False	False	True	True	False	False	True	False	False	False	False	False	False	False	False	False	0
86	False	False	False	False	False	False	False	True	False	False	False	False	False	False	False	False	0
92	False	False	False	True	True	True	False	False	False	False	False	False	False	False	True	False	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
32535	False	False	True	True	True	False	False	True	True	False	True	False	False	True	False	False	1
32543	True	False	False	False	False	False	False	True	False	False	True	False	False	False	True	True	1
32549	False	True	False	False	False	False	False	False	True	True	True	False	False	False	False	False	1
32563	False	False	False	False	False	True	False	False	True	True	False	False	False	False	False	False	1
32587	False	True	False	True	False	False	False	False	False	True	False	False	True	False	True	False	1

1998 rows × 17 columns

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	Logistic Regression	0.7404	0.6767	0.9718	0.7499	0.8465	0.0868
1	Ridge Classifier	0.7368	0.0000	0.9767	0.7452	0.8453	0.0578
2	Extreme Gradient Boosting	0.7353	0.6763	0.9534	0.7532	0.8415	0.1022
3	Ada Boost Classifier	0.7339	0.6813	0.9621	0.7486	0.8419	0.0762
4	Linear Discriminant Analysis	0.7339	0.6737	0.9553	0.7513	0.8409	0.0908
5	Gradient Boosting Classifier	0.7318	0.6726	0.9408	0.7553	0.8378	0.1119
6	Random Forest Classifier	0.7060	0.6039	0.8874	0.7562	0.8163	0.1003
7	SVM - Linear Kernel	0.6923	0.0000	0.8738	0.7561	0.7942	0.0581
8	Extra Trees Classifier	0.6910	0.5946	0.8660	0.7528	0.8051	0.0741
9	Decision Tree Classifier	0.6788	0.5527	0.8301	0.7574	0.7920	0.0932
10	K Neighbors Classifier	0.6624	0.5585	0.8262	0.7426	0.7808	0.0373
11	Quadratic Discriminant Analysis	0.6459	0.6788	0.6680	0.8183	0.7343	0.2176
12	Naive Bayes	0.6388	0.6748	0.6544	0.8198	0.7264	0.2114

效果並沒有原先好，有兩個可能理由，一是選擇的features數量相比於資料量較少，二是對於整個群體的發文來分析的話，比較難抓到特定的習慣用詞

# 小結

- 更換成另外一個Page Rank選出的KOL，也可以得到不錯的預測結果
- 更改預測的贖回率指標，不會影響原先預測的模型準確度
- 字典法針對特定的KOL預測時效果最好





# BERT

以「心得」做資料子集有最佳效果

		time
type	signal	
其他	False	1261
	True	1621
心得	False	526
	True	1472
投顧	False	36
	True	50
新聞	False	8161
	True	9722
標的	False	2323
	True	2421
請益	False	1661
	True	2448
閒聊	False	390
	True	381

```
model.fit(  
    X,  
    y,  
    epochs=EPOCHS,  
    batch_size=BATCH_SIZE,  
    validation_split=0.2  
)  
  
samples, validate on 400 samples  
  
=====] - 194s 121ms/step - loss: 0.6636 - sparse_categorical_accuracy: 0.6302 - val_loss: 0.3885 - val_sparse_categorical_accuracy: 0.9425  
=====] - 194s 121ms/step - loss: 0.5775 - sparse_categorical_accuracy: 0.6871 - val_loss: 0.3808 - val_sparse_categorical_accuracy: 0.8925
```

- 可以看到左邊的樣本數量分析中，心得文雖然有資料數目不均等的狀態，但是我們的預測效果仍然勝過壓單邊的狀況。
- 造成此結果的可能是因為原先BERT訓練時使用大量語料，因此如果綜合各個心得文行家的說法，會比單一KOL的可能會有特殊用語習慣時，更有預測的能力。

# 強基金部分 - 計量方法

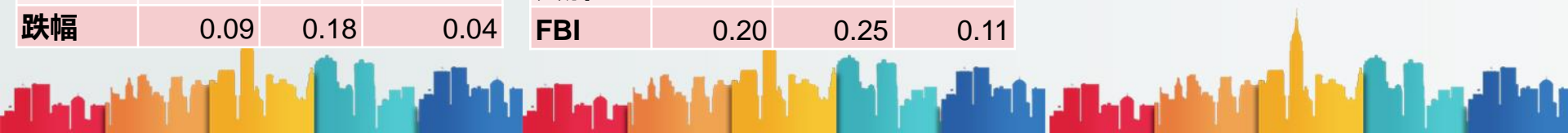


買超日常出現詞彙	買超分數(a)	TF-IDF index(b)	結果(a*b <sup>^</sup> .5)
現金	0.73	0.15	0.28
撲滿	0.24	0.32	0.13
沒錢	0.28	0.21	0.13
天然資源	0.23	0.28	0.12
組合	0.23	0.18	0.10
心情	0.18	0.28	0.10
跌破	0.12	0.59	0.09
加油	0.11	0.53	0.08
小弟	0.18	0.16	0.07
大跌	0.17	0.14	0.06
連動	0.14	0.18	0.06
明年	0.12	0.19	0.05
台股	0.09	0.22	0.04
事件	0.10	0.19	0.04
跌幅	0.09	0.18	0.04

賣超日常出現詞彙	賣出分數(c)	TF-IDF index(d)	結果(c*d <sup>^</sup> .5)
紅利	0.62	0.94	0.74
恭喜	0.60	0.93	0.72
NN	0.40	1.00	0.63
生技	0.38	0.89	0.55
贖回	0.17	0.66	0.28
大漲	0.25	0.52	0.26
謝謝	0.22	0.46	0.22
新年快樂	0.64	0.27	0.22
利安	0.72	0.24	0.21
表單	1.00	0.21	0.21
去年	0.13	0.57	0.20
強網	0.41	0.30	0.19
一支	0.26	0.28	0.14
天后	0.87	0.15	0.14
FBI	0.20	0.25	0.11

在此先列出強基金論壇在買超日以及賣超日常出現的詞彙

- 在買超日常常出現詞彙有：大跌、跌破等詞彙（逢低點進場）的事實。現金、沒錢（討論各自手頭上資產的流動性）等……。
- 而在賣超日容易出現的詞彙有恭喜、謝謝、贖回、大漲、去年、一支等詞彙（分享基金成效、描述購買基金的脈絡），另外也有天后、FBI等該網站的專門術語（FBI是該論壇專門用以評估相對低點的指數）



Y = 買賣超情形	Est.	S.E.	t-value	Pr(> t )
(Intercept)	2.354	0.041	57.20	0.00E+00
賣超分數	14.939	8.308	② 1.80	7.22E-02
買超分數	-12.698	3.025	① -4.20	2.74E-05

F-statistic: 10.62 on 2 and 5521 DF, p-value: 2.494e-05

首先使用購入以及贖回分數，針對驗證集 ( Validation sets ) 先進行 OLS 回歸分析，結果發現

- **【買超分數】**其實較能夠較好地描述市場情形 ( ①較高的 t-value 代表較好的顯著情形 )
    - 鎖定特定關鍵字：
      - 「大跌」 ( 逢低點進場 ) 、 「小弟」 ( 詢問投資目標的請教語氣 ) 、
      - 「現金」 ( 討論各自手頭上資產的流動性 ) ，其實可以對目前市場的買超情形有所瞭解。
  - **【賣超分數】**的表現則較差
    - ( ②統計顯著性僅在常使用的 0.05 邊緣，顯示這個變數並不是一個很強的指標 )
    - 買進基金時，需要請教彼此對特定投資標的的看法。但**基金贖回則是比較屬於個人停損點的評估**。考慮進該論壇性質，**強基金其實以詢問投資標的類文章為主，而抒發情緒類文章相對較少**，也造成預測贖回的結果沒有比預測購入時的結果來的好。
- 結論：較難單依賴論壇上的討論結果進行未來市場情形的預測



賣超情形	Est.	SE	t-v	Pr(> t )
(截距)	-3.57	0.041	-321.	<0.00
賣超分數	32.74	32.91	1.00	0.32
買超分數	0.27	0.35	0.77	0.44

買超情形	Est.	SE	t-v	Pr(> t )
(截距)	4.38	0.008	551.1	<0.00
賣超分數	6.30	2.77	2.27	0.02
買超分數	-3.31	0.55	-6.04	<0.00

- 這個模型的解釋能力在不同百分位的  $y$  上出現差異（在此取前 10% 買超情形以及前 10% 賣超情形進行觀察），在買超情形比較嚴重的情況下，這個模型的解釋能力會較好，③買賣超分數在市場氣氛是買超的時候，每個係數都變得更顯著了
- 代表握有近期的市場交易資訊，如果最近市場氛圍使散戶投資人的交易情形為買超，再使用這個模型找出買賣超傾向分數比較高的文章，可以較為準確地掌握市場脈動，並利未來進行相對的策略舉措



# 強基金部分 - 機器學習方法

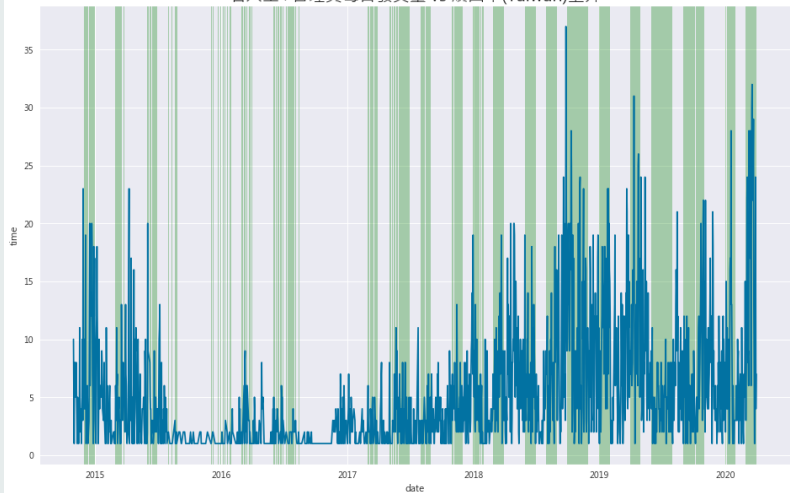


# 使用資料

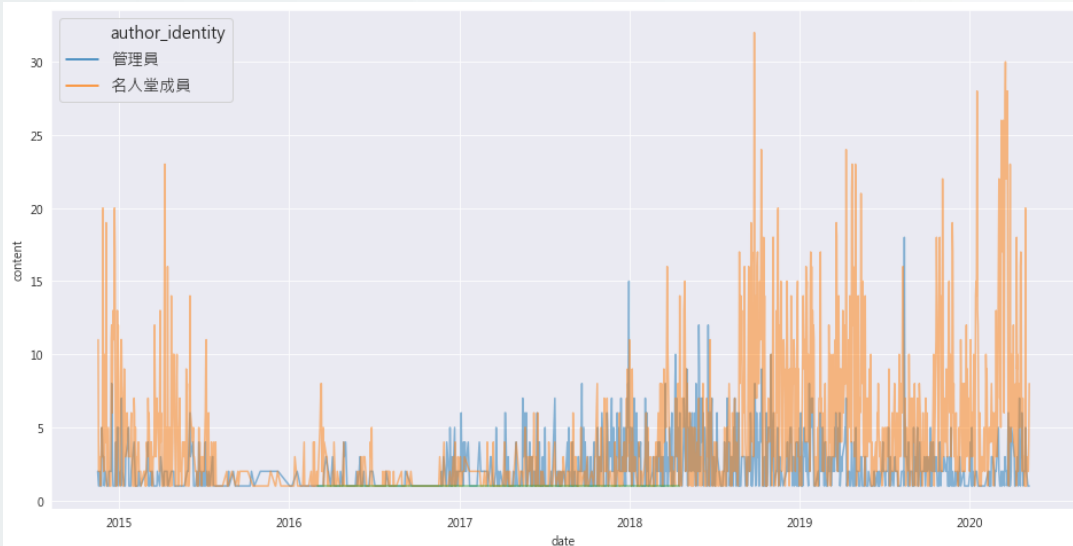
我們使用「名人堂成員和管理員」的文字資料，預測「野村的台灣股票型基金」贖回率上升現象

```
signal  
False 4270  
True 4609
```

名人堂+管理員每日發文量 vs 贖回率(Taiwan)上升



管理員與名人堂成員發文數比較



可以看到在贖回率上升與否下，發文文章數沒有特殊差異



# 訓練模型

可以看到AUC大部分表現都僅落在0.5~0.6之間。

依照Accuracy排序下，最佳的模型來自脊回歸以及Logit等傳統計量方法模型。

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	Ridge Classifier	0.5758	0.0000	0.7388	0.5786	0.6488	0.1326
1	Logistic Regression	0.5753	0.5993	0.7439	0.5776	0.6501	0.1309
2	Linear Discriminant Analysis	0.5751	0.6004	0.7388	0.5780	0.6484	0.1310
3	Ada Boost Classifier	0.5707	0.5998	0.7554	0.5724	0.6512	0.1196
4	Gradient Boosting Classifier	0.5702	0.5915	0.7997	0.5674	0.6637	0.1133
5	Extreme Gradient Boosting	0.5643	0.5856	0.8011	0.5629	0.6611	0.1003
6	Extra Trees Classifier	0.5530	0.5687	0.5880	0.5769	0.5819	0.1017
7	Random Forest Classifier	0.5449	0.5645	0.6184	0.5654	0.5905	0.0806
8	Decision Tree Classifier	0.5371	0.5451	0.5469	0.5655	0.5557	0.0729
9	SVM - Linear Kernel	0.5351	0.0000	0.6418	0.5987	0.5047	0.0588
10	K Neighbors Classifier	0.5285	0.5409	0.6331	0.5490	0.5875	0.0437
11	Naive Bayes	0.5036	0.5832	0.0886	0.7852	0.1589	0.0581
12	Quadratic Discriminant Analysis	0.4898	0.5875	0.0747	0.8016	0.1223	0.0321

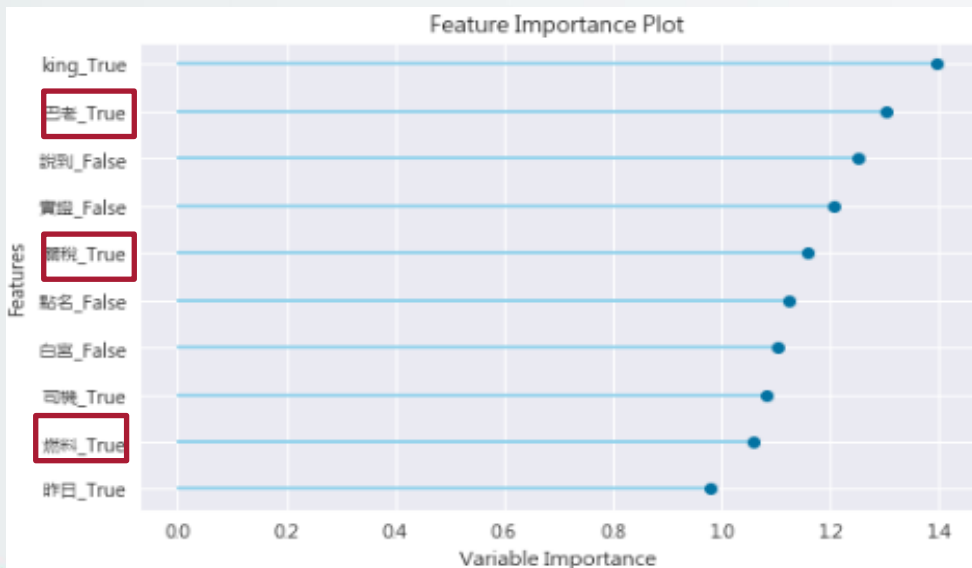
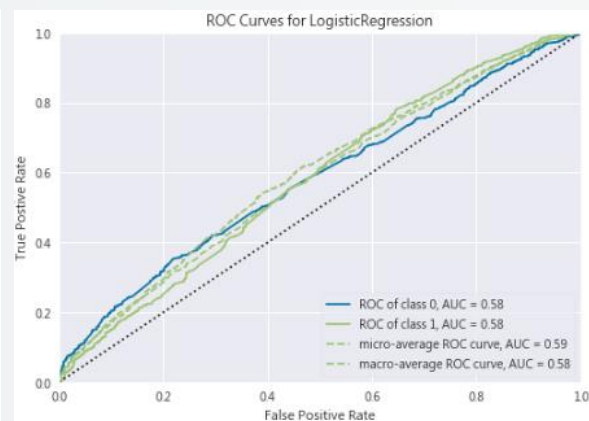
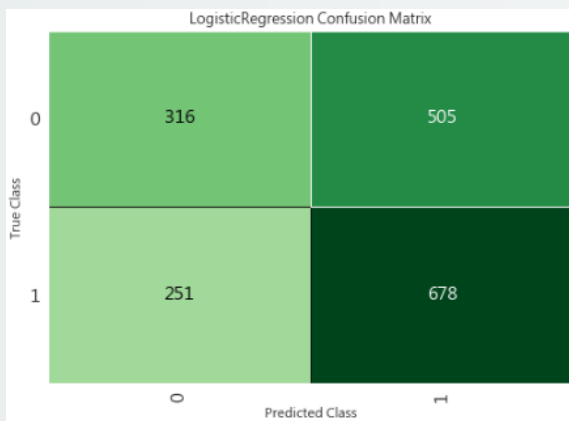


# 機器學習預測

對Logit模型做10-fold

```
lr = create_model(lr)
```

	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	0.5770	0.6064	0.7097	0.5833	0.6403	0.1387
1	0.6210	0.6644	0.7926	0.6099	0.6894	0.2241
2	0.5550	0.5855	0.7189	0.5632	0.6316	0.0903
3	0.5907	0.6034	0.7926	0.5850	0.6732	0.1577
4	0.5686	0.5855	0.7143	0.5762	0.6379	0.1194
5	0.5564	0.5688	0.7465	0.5625	0.6416	0.0888
6	0.5466	0.5673	0.7373	0.5556	0.6337	0.0687
7	0.6176	0.6263	0.7917	0.6064	0.6867	0.2177
8	0.5466	0.5784	0.7222	0.5552	0.6278	0.0725
9	0.5735	0.6066	0.7130	0.5789	0.6390	0.1315
Mean	0.5753	0.5993	0.7439	0.5776	0.6501	0.1309
SD	0.0257	0.0280	0.0334	0.0184	0.0223	0.0527



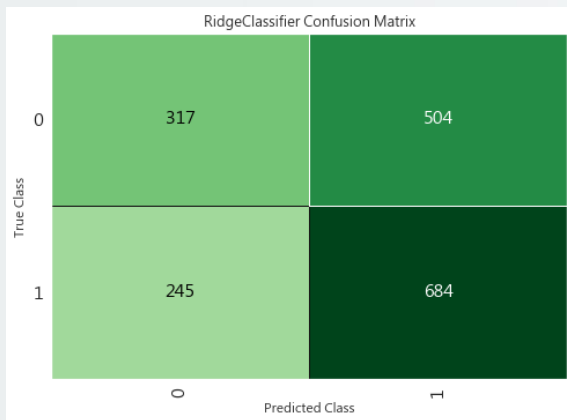
雖然模型本身預測效果不好，但是feature importance plot可以給我們一些有趣的insight，例如當論壇討論「巴老」(巴菲特)、「關稅」、「燃料」時，很有可能會影響贖回率上升

# 機器學習預測

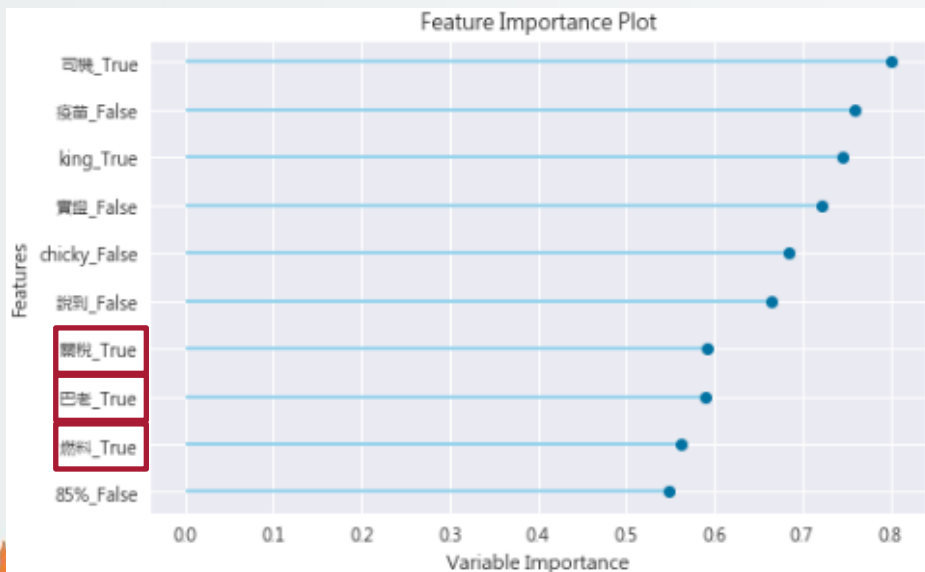
對Ridge模型做10-fold

```
rd = create_model('ridge')
```

	Accuracy	AUC	Recall	Prec.	F1	Kappa
0	0.5672	0.0	0.6959	0.5763	0.6305	0.1193
1	0.6210	0.0	0.7834	0.6115	0.6869	0.2250
2	0.5575	0.0	0.7097	0.5662	0.6299	0.0967
3	0.5956	0.0	0.7972	0.5884	0.6771	0.1678
4	0.5735	0.0	0.7097	0.5811	0.6390	0.1305
5	0.5613	0.0	0.7465	0.5664	0.6441	0.0995
6	0.5515	0.0	0.7419	0.5590	0.6376	0.0788
7	0.6127	0.0	0.7778	0.6043	0.6802	0.2086
8	0.5515	0.0	0.7222	0.5591	0.6303	0.0831
9	0.5662	0.0	0.7037	0.5736	0.6320	0.1168
Mean	0.5758	0.0	0.7388	0.5786	0.6488	0.1326
SD	0.0239	0.0	0.0347	0.0172	0.0219	0.0487



Ridge不能計算AUC



剛剛在Logit看到的Feature Importance也可以在這  
裡看到不謀而合的結果，另外像司機、King也和  
贖回率相關，但是比較難以直觀解釋

# BERT

Train on 7103 samples, validate on 1776 samples

Epoch 1/5

7103/7103 [=====] - 845s 119ms/step - loss: 0.6976 - sparse\_categorical\_accuracy: 0.5216 - val\_loss: 0.7183 - val\_sparse\_categorical\_accuracy: 0.4657

Epoch 2/5

7103/7103 [=====] - 840s 118ms/step - loss: 0.6873 - sparse\_categorical\_accuracy: 0.5433 - val\_loss: 0.7035 - val\_sparse\_categorical\_accuracy: 0.4690

Epoch 3/5

7103/7103 [=====] - 840s 118ms/step - loss: 0.6779 - sparse\_categorical\_accuracy: 0.5643 - val\_loss: 0.6953 - val\_sparse\_categorical\_accuracy: 0.5676

Epoch 4/5

7103/7103 [=====] - 836s 118ms/step - loss: 0.6733 - sparse\_categorical\_accuracy: 0.5709 - val\_loss: 0.6954 - val\_sparse\_categorical\_accuracy: 0.5619

Epoch 5/5

7103/7103 [=====] - 836s 118ms/step - loss: 0.6580 - sparse\_categorical\_accuracy: 0.6100 - val\_loss: 0.7013 - val\_sparse\_categorical\_accuracy: 0.5518

<keras.callbacks.callbacks.History at 0x7f48fd4fdd8>

可以看到經過5個Epoch之後，交叉驗證的Accuracy結果仍然維持在0.56上下，和字典法的結果差不多



# 小結

- PTT部分

- 如果要針對特定KOL分析，使用字典法效果較好
- 在我們採用的方法中，使用BERT對「心得」文章訓練效果最好

- 強基金部分

- 計量方法部分

- 【買超分數】其實較能夠較好地描述市場情形
    - 【賣超分數】的表現則較差

- 機器學習方法部分

- 用字典法和NLP表現都不如計量方法理想





# 應用程式與其他



# 應用程式

<http://3.134.79.174>

Made by plotly and javascripts



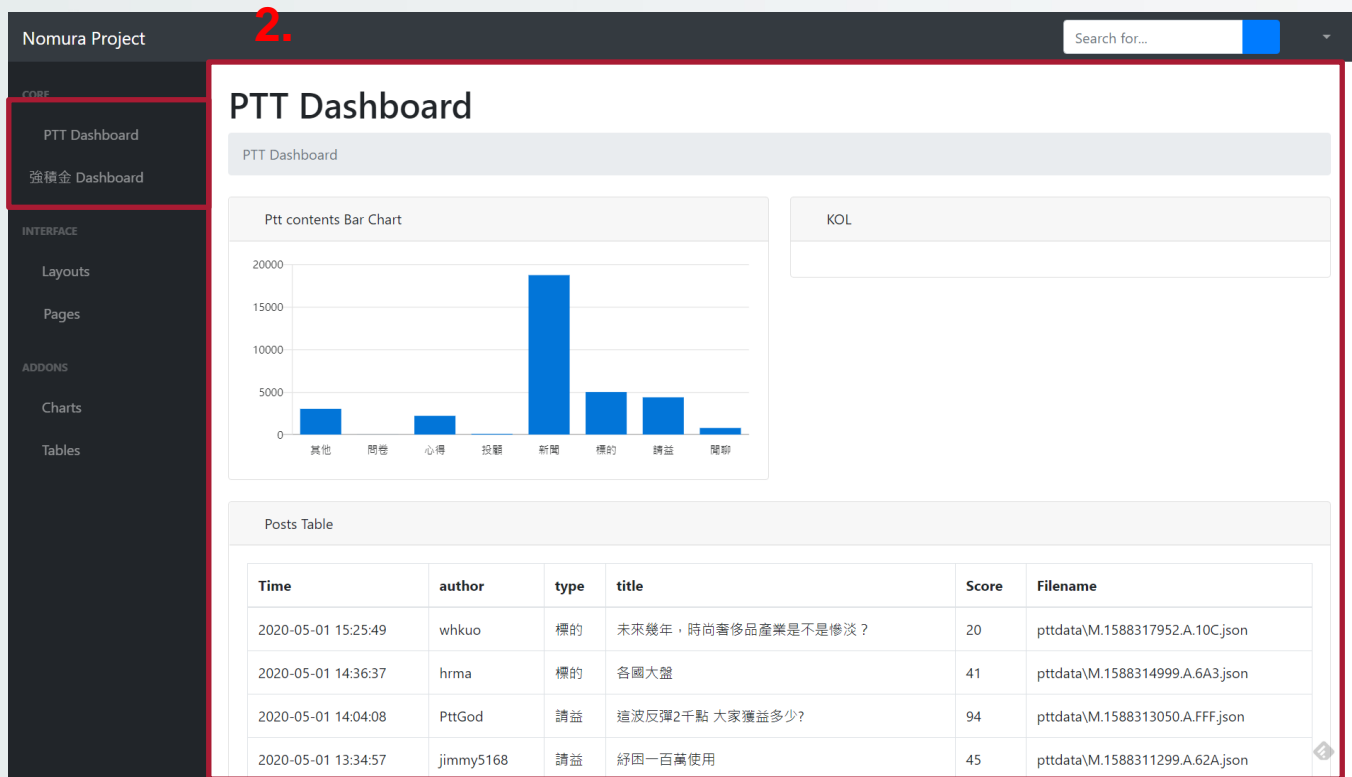


# 介面呈現

1.

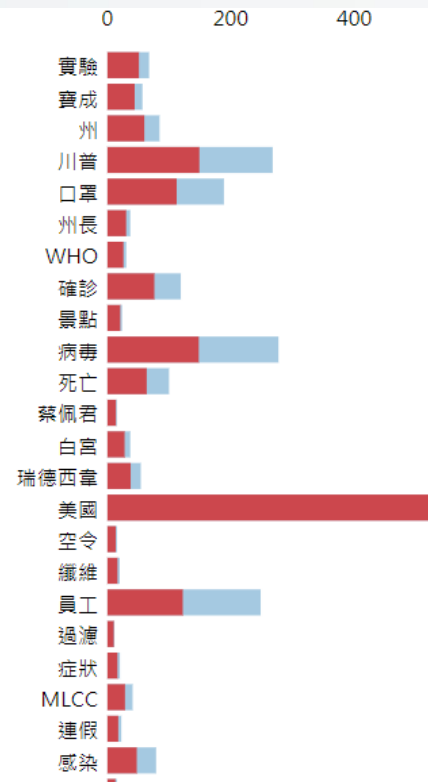
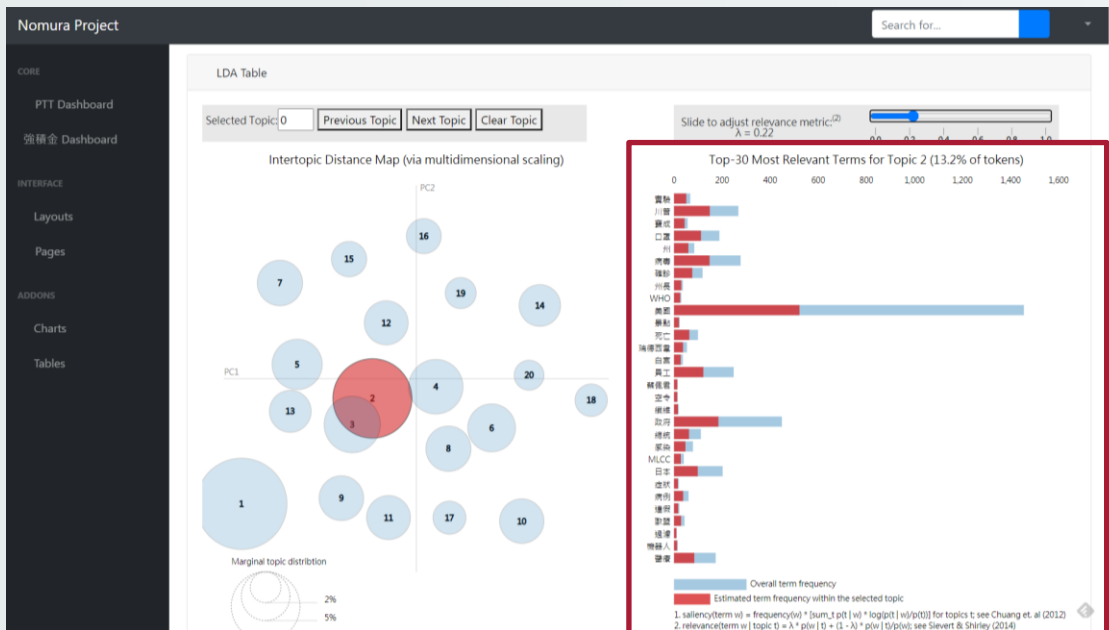
1. 選擇資料來源

2. 資料視覺化Dashboard



2.

# LDA



可以看到PTT的主題二應該是在討論Covid-19疫情相關的議題

# 組內分工

**謝侑軒：** PTT資料收集，PTT、強基金機器學習&NLP模型，贖回率指標、LDA，CKIP，PPT製作

**莊浩平：** 強基金、贖回率資料收集，強基金計量模型，PPT製作

**蔡承男：** Page Rank，PPT製作

**曾琮傑：** 前端介面

**劉珈卉：** 聯絡Mentor



**感謝您的聆聽**

