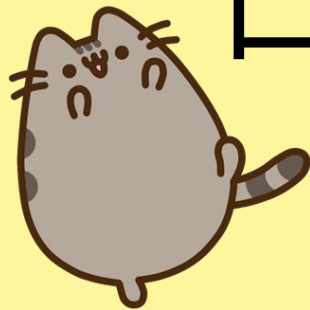


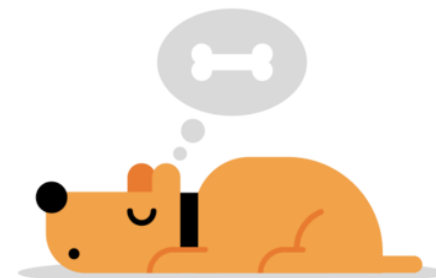
台灣寵物數量分析

李旻恩

2022/08/25



目錄

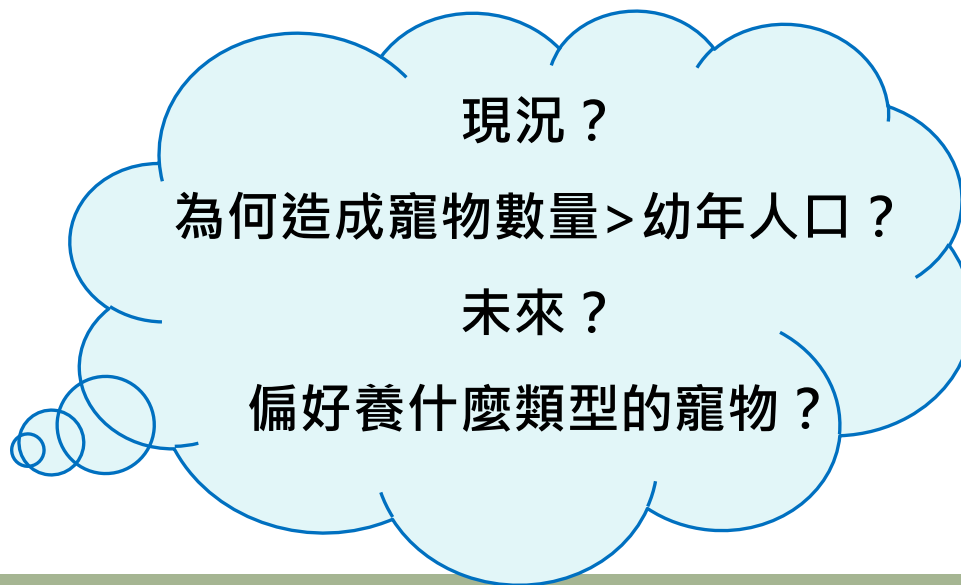


- ◆ 動機
- ◆ 圖表分析
- ◆ 結論
- ◆ 問題與解決

動機

- 根據內政部、農委會與時勢公司整理分析，**2021年全台犬貓數量達到295萬隻**，相比0~14歲幼年人口數僅289萬人，首度發生寵物數量比幼年人口數還多。(2018年)

<http://www.trendsightinc.com/insight/2020>黃金交叉，台灣貓狗數首次超過兒童/



現況？

為何造成寵物數量>幼年人口？

未來？

偏好養什麼類型的寵物？

圖表分析



使用資料：

- **寵物數量** – 寵物登記管理資訊網
- **出生人口** – 國家發展委員會 人口推估查詢系統
- **0~14歲人口** – 國家發展委員會 人口推估查詢系統
- **平均薪資** – 行政院主計總處 薪情平台
- **消費者物價指數** – 中華民國統計資訊網
- **銷售額與家數** – 中華民國財政部 財政統計資料庫查詢
- **抗憂鬱藥物使用人數** – 政府資料開放平台 (資料來源：健保醫療費用申報處方醫令暨明細檔)
- **家貓家犬數量** – 動物保護資訊網

PART 1：寵物VS人口

現在發展情形？

PART 2：原因

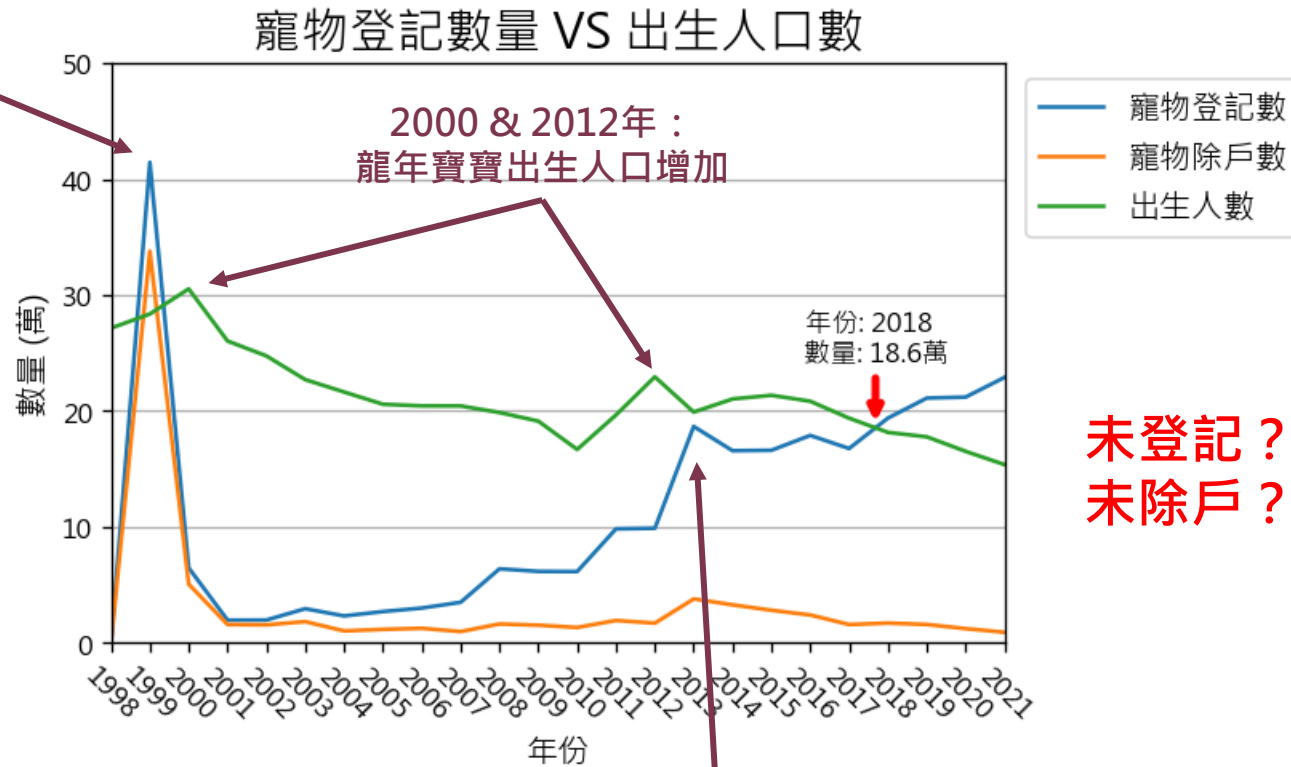
PART 3：未來預測

PART 4：貓 VS 狗

誰佔大宗？

圖表分析－寵物 VS 人口

全面上線



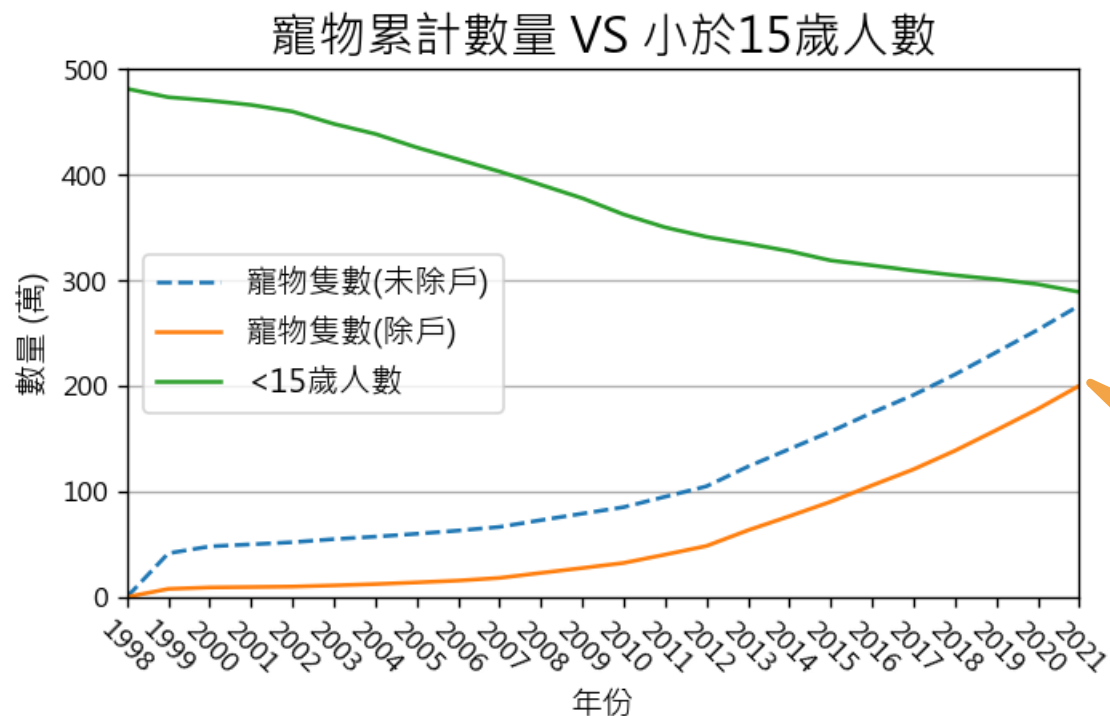
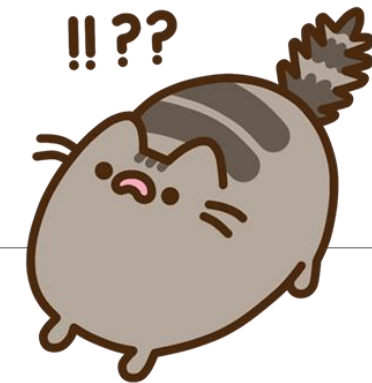
寵物登記數：晶片登記
寵物除戶數：登記失蹤>1年 or 死亡

未登記？
未除戶？

2018年寵物登記的數量
V
出生人口數

立法院經濟委員會於本（2013）年1月9日，審查通過「動物保護法」部分條文修正案，寵物出生後須依第十九條規定，植入晶片，辦理寵物登記

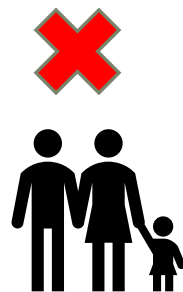
圖表分析－寵物 VS 人口



——▶ 2021年寵物的數量>小於15歲人口數？

去除掉除戶數後，實際寵物數量>小於15歲人口數的時間點應該更晚

圖表分析－原因



2017年

全球教育支出排行

排名	國家地區	花費(元)
1	香 港	4,070,559
2	阿 聯 酋	3,060,842
3	新 加 坡	2,184,921
4	美 國	1,800,691
5	台 灣	1,737,859
6	中 國	1,321,074
7	澳 洲	1,121,182
8	馬 來 西 亞	784,753
9	英 國	765,750
10	墨 西 哥	702,610

備註：每位子女由小學到大學的平均花費
資料來源：HSBC，時勢公司整理。

光教育費用
~170萬

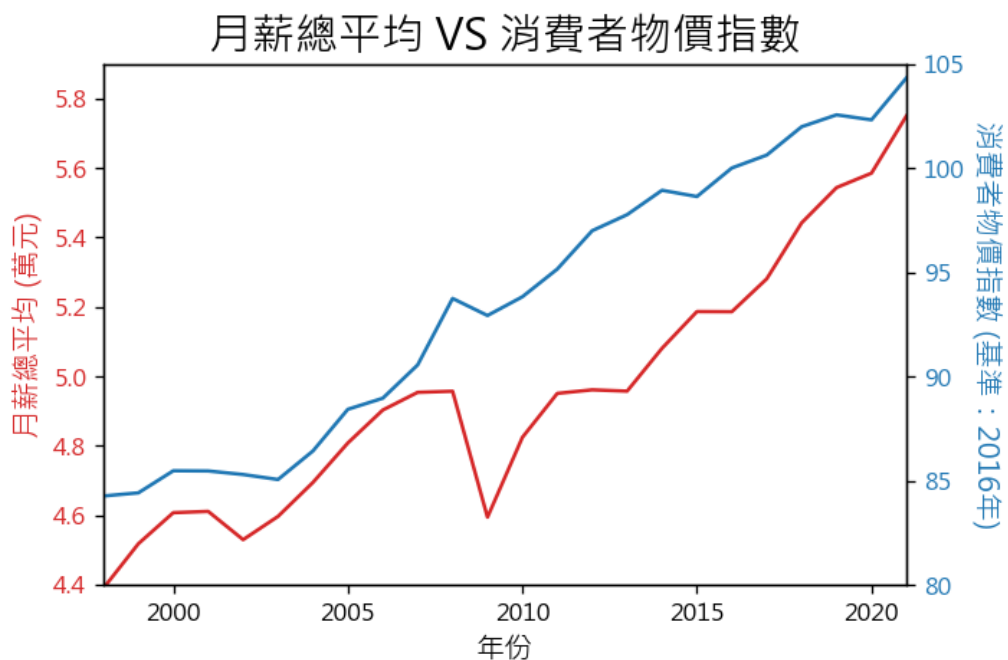


2019年
(富邦產險整理)

貓狗花費表格

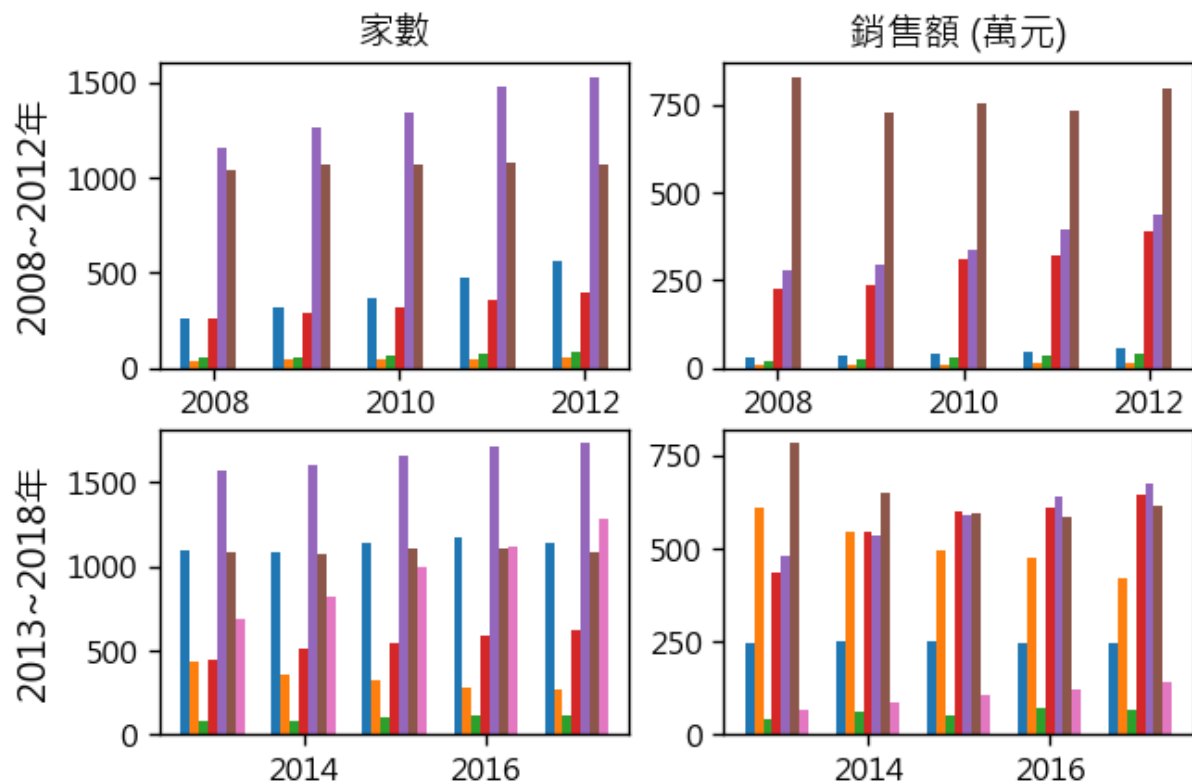
各種費用	養貓花費	養狗花費
晶片費用	植入費：300元 + 登記費用	
伙食費	一年大約5,000~15,000元	
施打疫苗	三合一疫苗500~900元	幼犬：三劑五合一疫苗+狂犬病疫苗3,000~3,900元 成犬：五合一疫苗+狂犬病疫苗3,000~3,900元 心絲蟲預防200~300元/月
其他開銷	3,000~5,000元/年	2,000~5,000元/年
醫療費	不一定，500~上萬元不等	
喪葬費	2,500~50,000元	

不含生病、意外、物品損毀
50~60萬

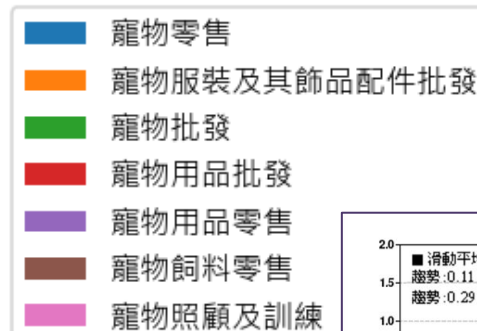


經濟壓力 + 時間 + 陪伴
“寧養寵物不養小孩”

圖表分析－延伸



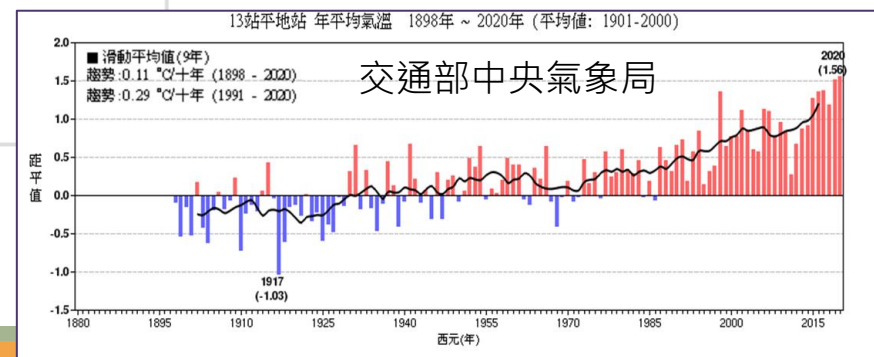
寵物數量增加，毛小孩相關產業看中商機，
各種類產業家數與銷售額逐年增長



寵物販賣
(變動不大)
推測市場飽和與推廣領養

寵物飼料零售
推測售寵物食品安全
問題事件影響後降低

寵物服裝與飾品
推測受氣溫增溫影
響需求降低



圖表分析－原因

回歸分析：(2012~2020年)

金錢 (總平均月薪，消費者物價指數)

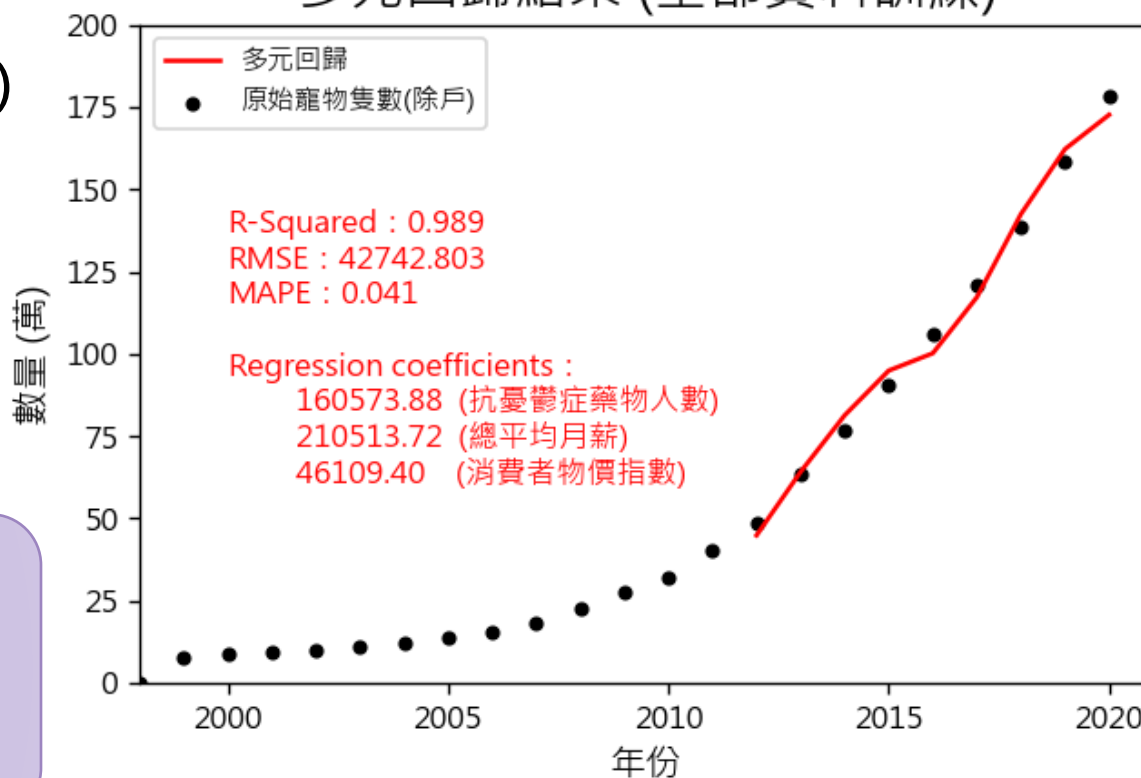
心理 (抗憂鬱症藥物人數)

1. 抗憂鬱藥物人數
2. 總平均月薪
3. 消費者物價指數

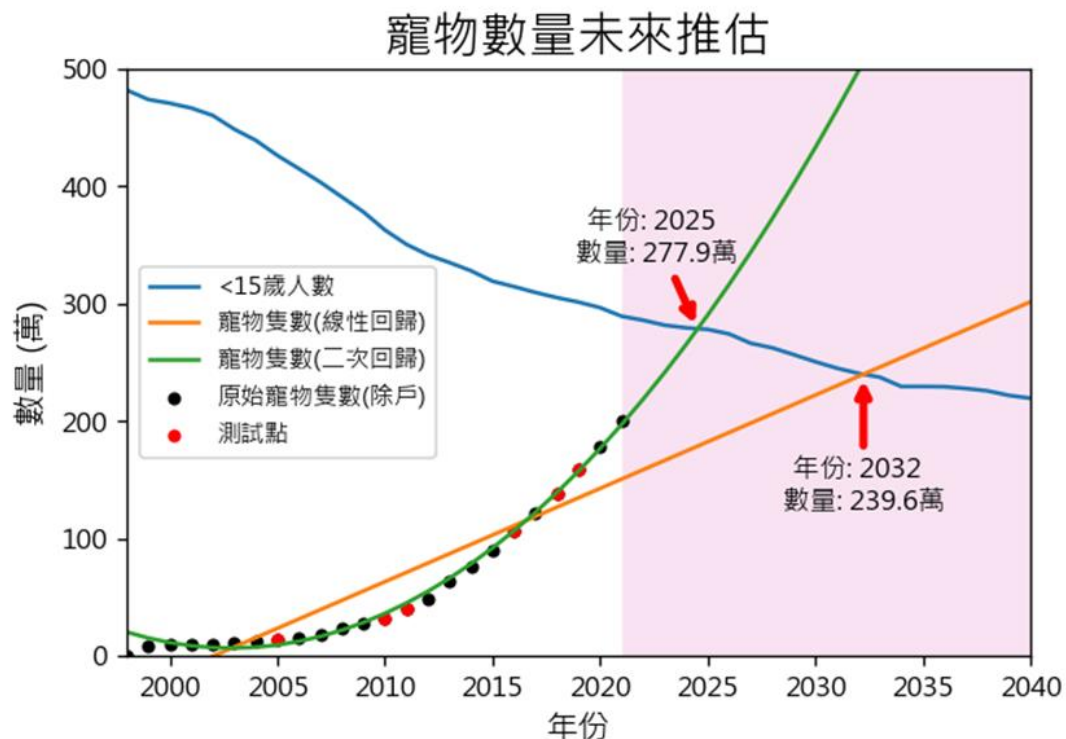
總平均月薪為最主要影響寵物數量的因素，服用
抗憂鬱藥物人數次之。

這三種自變數無未來推估資訊無法進行多元回歸預測...

多元回歸結果 (全部資料訓練)



圖表分析－未來預測



假設其他未考慮的因素變化率固定，以年份進行多項式回歸推估，推測未來2025年小於15歲人口將低於寵物的數量

回歸預測：1999~2021年 → 預測2021年以後
(同以 20% 測試資料計算得分結果)

線性回歸：(自變數為年份)

R-Squared(解釋力)：0.853

RMSE(誤差)：212703.528 (隻)

MAPE(誤差)：0.449

二次回歸：(自變數為年份)

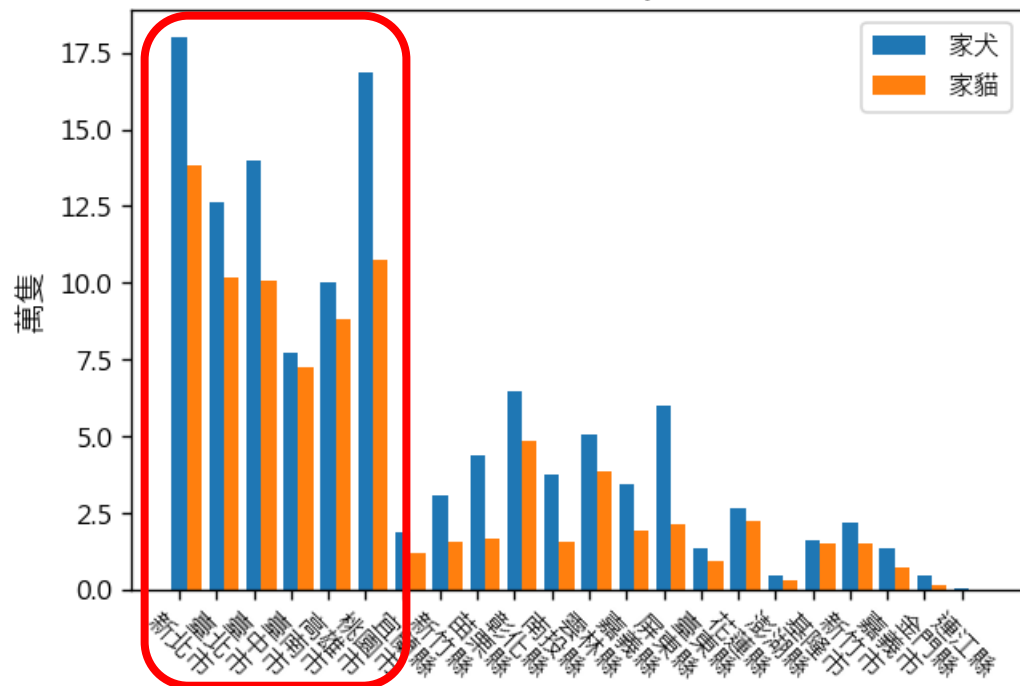
R-Squared(解釋力)：0.997

RMSE(誤差)：31292.816 (隻)

MAPE(誤差)：0.097

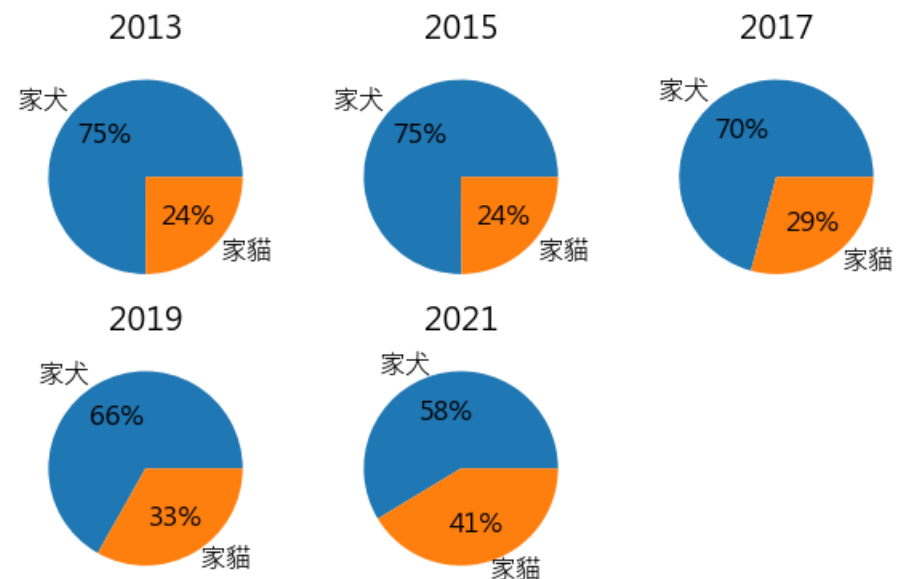
圖表分析 - PART 4 : 貓 VS 狗

2021年各縣市家貓/犬數量分布



都市化程度較高的區域，寵物的數量較多

每年飼養貓與狗數量分布



家犬的數量 > 家貓
但飼養貓的佔筆有逐年增長的趨勢

結論

- ◆ 登記寵物的數量隨著時間日益增長，隨著生育小孩的意願降低，在2018年時出生人口數就已經低於登記寵物的數量。
- ◆ 造成寧願養寵物不養小孩的原因，與經濟上的壓力高度相關，在此情況下心理上的陪伴需求增加，寵物數量逐漸升高，相關行業看準商機家數與銷售額逐年增長。
- ◆ 假設其他未考慮的因素變化率固定，以年份進行多項式回歸推估，推測未來2025年小於15歲人口將低於寵物的數量。
- ◆ 都市化程度較高的區域，寵物的數量較多，且大多數人偏好養狗，但飼養貓的佔比有逐年增長的趨勢。



問題與解決

```
from bs4 import BeautifulSoup as bs
from selenium import webdriver
from selenium.webdriver.common.by import By
# from selenium.webdriver.support.ui import
import time

pet_list = []

driver = webdriver.Chrome("chromedriver")
driver.get("https://www.pet.gov.tw/Web/0302.aspx?PG=1")
# 日期定位
select_start = driver.find_element(By.ID, "txtSDATE")
select_end = driver.find_element(By.ID, "txtEDATE")
year = [2020]
# 選擇日期
select_start.send_keys(str(year[0])+"/01/01")
select_end.send_keys(str(year[0])+"/12/31")
# 選擇選項"全部"
select_type = driver.find_element(By.CLASS_NAME, "circle")
select_type.click()
# 點擊搜尋
driver.find_element(By.ID, "aSearch").click()
driver.implicitly_wait(5)
# 資料解析
soup = bs(driver.page_source, "lxml")
table = soup.select_one("#CountTown")
print(table.prettify)
```

寵物登記管理資訊網
Pets Registration Information System

🔍 寵登資料查詢 📰 最新消息 🐾 寵物大小事 🌐 公開資訊 🌐 相關網站連結 ⚙️ 登入後台 📖 操作手冊

登記狀況

最後統計日:2022/08/24

每年 01/01 ~ 12/31

作業時間: ~ 2021/12/31

動物類別: ☒ 全部 ☐ 狗 ☐ 貓

▶ 動物登記日期 ▶ 登記數(A)為飼主戶籍 ▶ 轉讓數(C)為轉出飼主戶籍

縣市	登記單位數	登記數(A)	除戶數(B)	轉讓數(C)	變更數(D)	絕育數(E)	絕育除戶數(F)	免絕育數(G)	免絕育除戶數(H)	絕育率 (E-F)/(A-B)	繁殖管理率 (E-F)+(G-H)/(A-B)
新北市	465	417172	134624	103017	261422	207855	45955	3737	614	57.30	58.41
臺北市	317	308073	97505	61165	292690	175123	36683	1952	228	65.75	66.56
臺中市	326	382005	113287	70507	196230	263614	48678	16622	1284	79.99	85.69
臺南市	249	239695	76111	43163	94094	124108	22892	7535	711	61.87	66.05
高雄市	336	359416	98187	74927	146502	191234	32900	11513	968	60.61	64.65
桃園市	273	246728	42940	63758	145404	142346	15409	3054	412	62.29	63.59
宜蘭縣	40	53141	16859	9661	41737	24266	4414	2143	387	54.72	59.56
新竹縣	43	49686	14558	12372	27088	26136	3804	1836	196	63.57	68.24
苗栗縣	53	67870	18763	14405	39765	33259	6232	336	41	55.04	55.64

問題與解決

金門縣	2	20224	8041	4578	16107	8837	2052	2000	278	55.69	69.83
連江縣	1	1874	537	378	1814	860	137	8	0	54.08	54.67
合計	2649	2760258	760132	582817	1532871	1547414	271543	66489	7320	63.79	66.75

```
<bound method Tag.prettify of <table class="table table-bordered table-g13" id="CountTown">
<thead>
<tr>
<th>縣市</th>
<th>鄉鎮</th>
<th>登記單位數</th>
<th>登記數(A)</th>
<th>除戶數(B)</th>
<th>轉讓數(C)</th>
<th>變更數(D)</th>
<th>絕育數(E)</th>
<th>絕育除戶數(F)</th>
<th>免絕育數(G)</th>
<th>免絕育除戶數(H)</th>
<th>絕育率<br/>
<th>繁殖管理率<br/>

(E-F)/(A-B)</th>
(E-F)+(G-H)/(A-B)</th>

</tr>
</thead>
<tbody>
</tbody>
</table>>
```



無法直接檢視所以存取網頁原始碼

```
42 with open("test.html", "w", encoding="utf8") as file:
43     file.write(soup.prettify())
```

```
for (var key in json) {
    dal.ini(json[key]);
    $("#CountTown").append("<tr id='tr_' + key + '>'>
        + "<td data-th='縣市'><a href='void(0);' class='CountySpan' id=''" + dal.AreaID + "'>" + dal.AreaName + "</a></td>"
        + "<td data-th='鄉鎮'>" + dal.AreaName + "</td>"
        + "<td data-th='登記單位數'>" + dal.fld01 + "</td>"
        + "<td data-th='登記數(A)'>" + dal.fld02 + "</td>"
        + "<td data-th='除戶數(B)'>" + dal.fld03 + "</td>"
        + "<td data-th='轉讓數(C)'>" + dal.fld05 + "</td>"
        + "<td data-th='變更數(D)'>" + dal.fld06 + "</td>"
        + "<td data-th='絕育數(E)'>" + dal.fld04 + "</td>"
        + "<td data-th='絕育除戶數(F)'>" + dal.fld08 + "</td>"
        + "<td data-th='免絕育數(G)'>" + dal.fld07 + "</td>"
        + "<td data-th='免絕育除戶數(H)'>" + dal.fld10 + "</td>"
        + "<td data-th='絕育率(E-F)/(A-B)'>" + (((dal.fld04 - dal.fld08) / (dal.fld02 - dal.fld03)) * 100.0).toFixed(2) + "</td>"
        + "<td data-th='繁殖管理率(E-F)+(G-H)/(A-B)'>" + (((dal.fld04 - dal.fld08) + (dal.fld07 - dal.fld10)) / (dal.fld02 - dal.fld03)) * 100.0).toFixed(2) + "</td>"
        + "</tr>");
    dalS.fld01 += parseInt(dal.fld01);
    dalS.fld02 += parseInt(dal.fld02);
    dalS.fld03 += parseInt(dal.fld03);
    dalS.fld04 += parseInt(dal.fld04);
    dalS.fld05 += parseInt(dal.fld05);
    dalS.fld06 += parseInt(dal.fld06);
    dalS.fld07 += parseInt(dal.fld07);
    dalS.fld08 += parseInt(dal.fld08);
    dalS.fld10 += parseInt(dal.fld10);
}
```

但還是包在script中無解

問題與解決

```
from selenium import webdriver
from PIL import Image
import time
from selenium.webdriver.common.by import By

driver = webdriver.Chrome("chromedriver")
driver.get("https://www.pet.gov.tw/Web/0302.aspx?PG=1")

year = list(range(1998,2022))
# year =[1998]
def get_pic(year, path):
    for i in range(len(year)):
        print(year[i])
        filename = path+str(year[i])+".png"
        select_start = driver.find_element(By.ID,"txtSDATE")
        select_end = driver.find_element(By.ID,"txtEDATE")
        date = str(year[i])+"/01/01"
        select_start.send_keys(date)
        date = str(year[i])+"/12/31"
        select_end.send_keys(date)

        driver.find_element(By.ID, "aSearch").click()
        time.sleep(10)
        # 調整式窗大小與比例
        driver.execute_script("document.body.style.zoom='0.5';")
        driver.execute_script("var action=document.documentElement.scrollTop=350")
        # 截圖
        # driver.get_screenshot_as_file("2330.png")
        driver.get_screenshot_as_file(filename)
        webpage=Image.open(filename)
        # image_crop=webpage.crop(box=(left, upper, right, bottom))
        # 截圖範圍選取
        image_crop=webpage.crop(box=(15, 90, 1025, 750))
        # 存取截圖結果
        image_crop.save(filename)
        image_crop=Image.open(filename)
        # 調回式窗比例 (避免定位查詢難會出現問題)
        driver.execute_script("document.body.style.zoom='1';")
        # 清空時間選項
        select_start.clear()
        select_end.clear()
        time.sleep(10)
```

```
for j in driver.find_elements(By.CLASS_NAME, "form-check-label"):
    # print(i.text)
    if j.text == "全部":
        print(format(" 全部 ", "=^30"))
        j.click()
        path = "./data/pic/Pet_All_"
        get_pic(year, path)
    elif j.text == "狗":
        print(format(" 狗 ", "=^30"))
        j.click()
        path = "./data/pic/Pet_Dog_"
        get_pic(year, path)
    elif j.text == "貓":
        print(format(" 貓 ", "=^30"))
        j.click()
        path = "./data/pic/Pet_Cat_"
        get_pic(year, path)

driver.quit()
```

截圖結果

縣市	登記總數(只)	登記數(只)	給付數(只)	轉讓數(只)	棄養數(只)	收容數(只)	收容後放歸數(只)	生還數(只)	生還釋放數(只)
新北市	352	30242	1141	26945	49042	15641	177	192	9
臺北市	238	18628	517	14545	45881	11528	163	151	4
臺中市	256	32184	1138	19382	32191	23098	422	804	12
臺南市	184	18221	468	11227	10259	9791	95	685	14
高雄市	243	29743	1648	19570	22129	17979	816	687	16
桃園市	198	19187	533	17348	28807	10337	110	70	4
宜蘭縣	25	4077	123	2668	5977	2191	39	214	8
新竹縣	30	5129	161	3791	6477	3084	31	130	4
苗栗縣	35	5880	308	3674	7356	3247	64	29	0
彰化縣	78	14638	724	8491	8115	10016	444	168	2
南投縣	20	7981	262	2668	3442	6318	83	176	6
雲林縣	41	8334	507	3410	7269	6203	416	186	5
嘉義縣	14	5444	349	3171	2706	3132	32	54	0
屏東縣	93	7812	272	5786	5232	4302	78	41	2
臺東縣	7	4919	601	1191	2405	3270	524	46	2
花蓮縣	33	5510	62	1685	2451	3474	34	16	0
澎湖縣	10	1055	35	730	1289	488	12	89	6
基隆市	17	3816	139	2412	4059	2498	5	17	1
新竹市	37	3047	154	2664	2782	1611	18	41	1
嘉義市	33	1829	92	1620	1790	916	67	10	1
金門縣	1	1177	46	834	2233	706	22	268	14
連江縣	1	108	6	75	281	50	3	0	0
合計	1946	228961	9286	153887	252173	139880	3655	4074	111

手動抄寫每年合計資料，截圖方便確認資料...

未來可嘗試用深度學習模型來判斷表格內容...

問題與解決

營利事業家數及銷售額 - 第6次修訂.csv

營利事業家數及銷售額 - 第7次修訂.csv

統計期	行業別	家數總計	銷售額總計
97年			
97年	102年 4533-13寵物批發	84	405388
97年	102年 4552-19寵物服裝及其飾品配件批發	436	6076567
97年	102年 4699-22寵物用品批發	445	4359445
97年	102年 4852-20寵物零售	1096	2463810
97年	102年 4852-21寵物用品零售	1569	4781100
97年	102年 4852-22寵物飼料零售	1081	7821744
98年	102年 9630-16寵物繁殖	—	—
98年	102年 9690-21寵物照顧及訓練	688	663468
98年	103年 4533-13寵物批發	86	598709
98年	103年 4552-19寵物服裝及其飾品配件批發	356	5431603
98年	103年 4699-22寵物用品批發	507	5464693
98年	103年 4852-20寵物零售	1085	2521793
98年	103年 4852-21寵物用品零售	1597	5365566
98年	103年 4852-22寵物飼料零售	1069	6509259
99年	103年 9630-16寵物繁殖	—	—
99年	103年 9690-21寵物照顧及訓練	816	844957
99年	104年 4533-13寵物批發	99	536988
99年	104年 4552-19寵物服裝及其飾品配件批發	323	4951377
99年	104年 4699-22寵物用品批發	547	5989186
99年	104年 4852-20寵物零售	1141	2490745
99年	104年 4852-21寵物用品零售	1653	5914829
99年	104年 4852-22寵物飼料零售	1105	5926121
100年	104年 9630-16寵物繁殖	3 (D)	
100年	104年 9690-21寵物照顧及訓練	990	1055152
100年	105年 4533-13寵物批發	110	737168
100年	105年 4552-19寵物服裝及其飾品配件批發	281	4746615
100年	105年 4699-22寵物用品批發	587	6089885
100年	105年 4852-20寵物零售	1172	2482470
100年	105年 4852-21寵物用品零售	1715	6404629
100年	105年 4852-22寵物飼料零售	1102	5842786
101年	105年 9630-16寵物繁殖	3 (D)	
101年	105年 9690-21寵物照顧及訓練	1119	1216653
101年	106年 4533-13寵物批發	110	689198
101年	106年 4552-19寵物服裝及其飾品配件批發	265	4195136
101年	106年 4699-22寵物用品批發	615	6470262
101年	106年 4852-20寵物零售	1135	2438506
101年	106年 4852-21寵物用品零售	1728	6753867
101年	106年 4852-22寵物飼料零售	1086	6126026
106年	106年 9630-16寵物繁殖	5 (D)	
106年	106年 9690-21寵物照顧及訓練	1284	1437193

說明：(D)表示不陳示數值以保護個別資料。

兩個版本分開處理 (年份，排序，清理，重排，型態)

```

##% 6. 銷售額與家數 (六) (七)
sale6 = pd.read_csv(r"./data/營利事業家數及銷售額—第6次修訂.csv", encoding=("ANSI"))
sale6.dropna(inplace=True)
sale6["統計期"] = sale6["統計期"].str.replace("年", "").astype(int)+1911
sale6.sort_values(["行業別", "統計期"], inplace=True)
sale6.replace("(D)", np.nan, inplace=True)
sale6.dropna(inplace=True)
sale6.reset_index(drop=True, inplace=True)
sale6["<b>家數總計</b>"] = sale6["<b>家數總計</b>"].astype(int)
sale6["<b>銷售額總計</b>"] = sale6["<b>銷售額總計</b>"].astype(int)
sale7 = pd.read_csv(r"./data/營利事業家數及銷售額—第7次修訂.csv", encoding=("ANSI"))
sale7.dropna(inplace=True)
sale7["統計期"] = sale7["統計期"].str.replace("年", "").astype(int)+1911
sale7.sort_values(["行業別", "統計期"], inplace=True)
sale7.replace("(D)", np.nan, inplace=True)
sale7.replace("—", np.nan, inplace=True)
sale7.dropna(inplace=True)
sale7.reset_index(drop=True, inplace=True)
sale7["<b>家數總計</b>"] = sale7["<b>家數總計</b>"].astype(int)
sale7["<b>銷售額總計</b>"] = sale7["<b>銷售額總計</b>"].astype(int)

```


問題與解決

```
# predict ===== 1
# LinearRegression 1999~2021 --- 2050
year = pd.DataFrame(range(1998,2041)).values
x = pd.DataFrame(pet.loc[dy:,"年份"]).values
y = pd.DataFrame(pet_sum[dy:])
xTrain, xTest, yTrain, yTest = train_test_split(x, y, test_size=0.25, random_state=100)
lm_tt = LinearRegression(positive=True).fit(xTrain, yTrain) # 非負
pet_lm_predict_tt = lm_tt.predict(year)

# predict ===== 2
# https://accandrew2.pixnet.net/blog/post/359720873-python-%E8%AA%9E%E8%A8%80%E5%BB%BA%E7%AB%8B
# LinearRegression 1999~2021 --- 2050 (二次方)
poly = PolynomialFeatures(degree=2).fit(xTrain) # 2次方
x_new = poly.transform(xTrain)
lm2_tt = LinearRegression().fit(x_new, yTrain)
pet_lm2_predict_tt = lm2_tt.predict(poly.transform(year))
# 方法二
# from sklearn.pipeline import make_pipeline
# lm4_tt = make_pipeline(PolynomialFeatures(degree=2), LinearRegression()).fit(xTrain, yTrain)
# pet_lm4_predict_tt = lm4_tt.predict(year)
# pet_lm2_predict_tt = lm2_tt.predict(year)

# predict ===== 3
# 2012~2020 (金錢, 憂鬱)
# 無未來預測, 直接訓練全部
x_new2 = pd.DataFrame([data[1], income.loc[14:22, "月薪總平均"], index.loc[14:22, "總指數"]]).T
ss = StandardScaler().fit(x_new2)
x_new2 = ss.transform(x_new2)
lm3_tt = LinearRegression().fit(x_new2, y.loc[14-dy:22-dy])
pet_lm3_predict_tt = lm3_tt.predict(x_new2)
```

自變數(X) →	抗憂鬱症藥物人數	總平均月薪	消費者物價指數
回歸係數 (未標準化)	1.10	99.84	22858.53
回歸係數 (標準化)	160573.88	210513.72	46109.40

一元二次多項式回歸 – 多項式特徵陣列

多元回歸 – 標準化

x_new - NumPy object array

	0	1	2
0	1	1999	3.996e+06
1	1	2000	4e+06
2	1	2001	4.004e+06
3	1	2002	4.008e+06
4	1	2003	4.01201e+06
5	1	2004	4.01602e+06
6	1	2005	4.02002e+06
7	1	2006	4.02404e+06
8	1	2007	4.02805e+06
9	1	2008	4.03206e+06
10	1	2009	4.03608e+06
11	1	2010	4.0401e+06

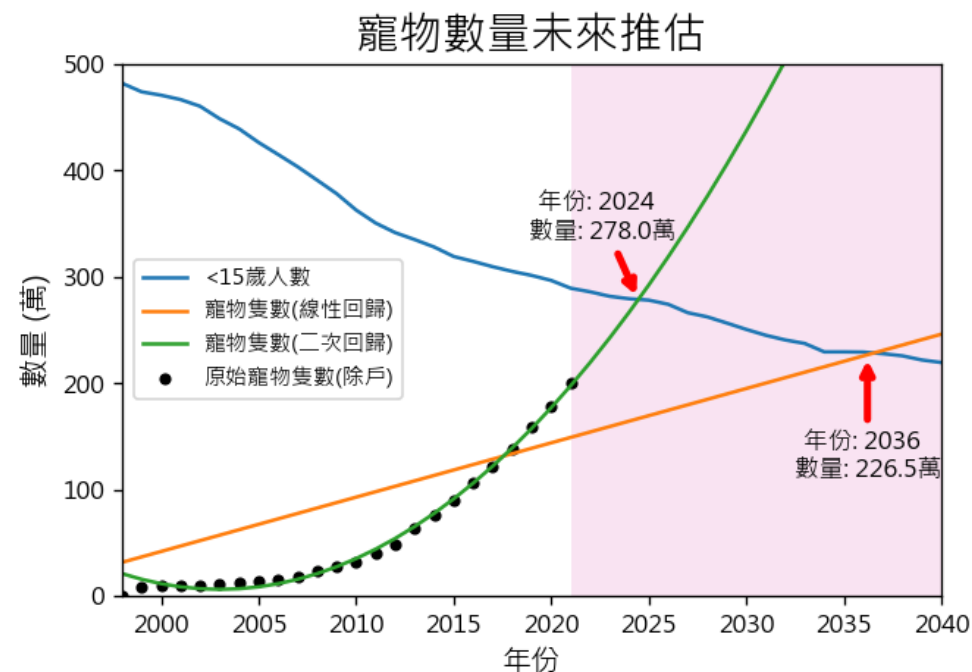
問題與解決

```
# %% function
# 兩條線的交點計算 (應有更好的方法)
def cross_point(line1,line2):
    # 取四點座標
    x1=line1[0]; y1=line1[1]; x2=line1[2]; y2=line1[3]
    x3=line2[0]; y3=line2[1]; x4=line2[2]; y4=line2[3]

    k1=(y2-y1)*1.0/(x2-x1) # 計算k1, 避免點均為整數, 需要進行浮點數轉化
    b1=y1*1.0-x1*k1*1.0 # 整型轉浮點型是關鍵
    if (x4-x3)==0: # L2直線斜率不存在操作
        k2=None
        b2=0
    else:
        k2=(y4-y3)*1.0/(x4-x3)#斜率存在操作
        b2=y3*1.0-x3*k2*1.0
    if k2==None:
        x=x3
    else:
        x=(b2-b1)*1.0/(k1-k2)
    y=k1*x*1.0+b1*1.0
    return [x,y]
```

$y = k_1x + b_1$
 $y = k_2x + b_2$ → 解聯立

```
plt.title("寵物數量未來推估", fontsize=18)
plt.broken_barh([(2021, 2040)], (0, 500), facecolors='tab:pink', alpha=0.2)
plt.plot(range(1998,1998+len(population.iloc[0])-1), population.iloc[0,1:]*1000/10000, label("<15歲人數")
plt.plot(year, pet_lm_predict_tt/10000, label="寵物隻數(線性回歸)")
plt.plot(year, pet_lm2_predict_tt/10000, label="寵物隻數(二次回歸)")
```

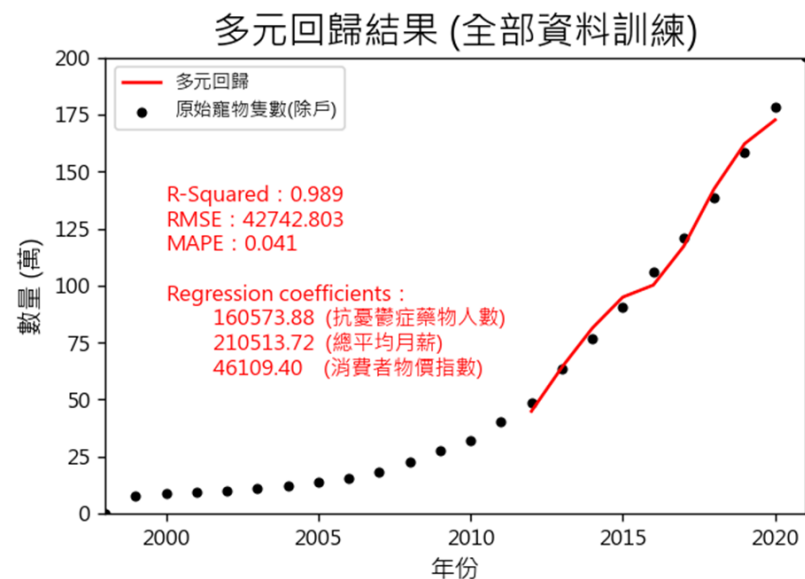


粉紅區域

問題與解決

```
t = f'''
R-Squared: {metrics.r2_score(y.loc[14-dy:22-dy], lm3_tt.predict(x_new2)):.3f}
RMSE: {(metrics.mean_squared_error(y.loc[14-dy:22-dy], lm3_tt.predict(x_new2)))*0.5:.3f}
MAPE: {metrics.mean_absolute_percentage_error(y.loc[14-dy:22-dy], lm3_tt.predict(x_new2)):.3f}

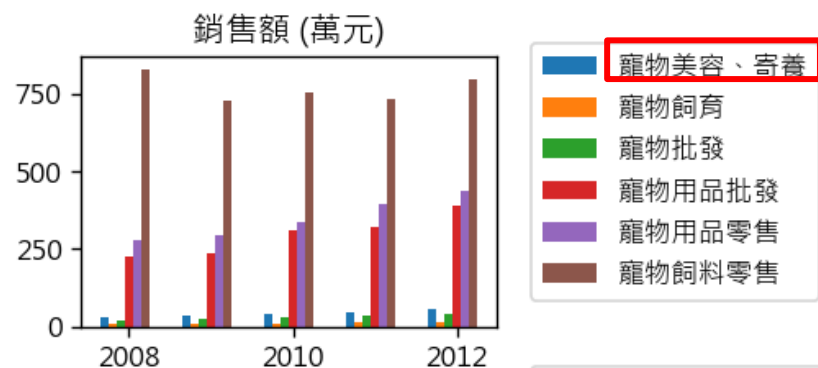
Regression coefficients:
    {lm3_tt.coef_[0][0]:.2f} (抗憂鬱症藥物人數)
    {lm3_tt.coef_[0][1]:.2f} (總平均月薪)
    {lm3_tt.coef_[0][2]:.2f} (消費者物價指數)
...
# print(t)
fig, ax = plt.subplots()
plt.title("多元回歸結果 (全部資料訓練)", fontsize=18)
plt.plot(range(2012,2021), pet_lm3_predict_tt/10000, label="多元回歸", color="red")
plt.scatter(pet["年份"], np.array(pet_sum)/10000, label="原始寵物隻數(除戶)", s=15, c="k")
plt.axis([1998,2021,0,200])
ax.text(2000, 50, t, color="red") ← 文字添加
# plt.grid()
plt.xlabel("年份", fontsize=font)
plt.ylabel("數量 (萬)", fontsize=font)
plt.legend(loc="upper left", fontsize=9)
plt.savefig("../figure/Pet_plot_11.png", dpi=120)
plt.show()
```



問題與解決

sale6 - DataFrame

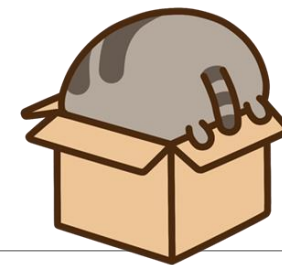
Index	西元(年)	統計期	行業別	家數總計	銷售額總計
0	2008	97年	9690-21寵物美容、寄養	259	295292



```
import re
width = 0.1
plt.suptitle("")
plt.subplot(2,2,1)
x = np.array(list(range(2008,2013)))-0.3
a = 0
for i in range(0,30,5):
    plt.bar(x+a, sale6["<b>家數總計</b>"][i:i+5], width=width, label=re.findall("[^0-9-]+",sale6["行業別"][i])[0])
    a += width
plt.ylabel("2008~2012年", fontsize=font)
plt.title("家數", fontsize=font)
```

正則取字串

心得



- ◆ 這次想說碰一下非專業領域的分析來練手，但需要花很多時間思考有甚麼資料可拿來做分析，像這次用抗憂鬱藥物使用人數來代表台灣人的心理狀態就感覺不太妥當，應該有其他指數可以來做代表。
- ◆ 時間要分配好，發散到最後想做的越來越多，資料蒐集的也越來越多，找資料與整理資料的時間也就被拉長，收斂成報告果然要果斷清楚一點，時間才不會被壓迫到很緊湊。
- ◆ 爬蟲練手撞牆，目前還無法處理如何爬取包在script中的資料，有嘗試看一下裡面的語法但還沒找到答案，雖然有想到奇葩方法，但依然未解決，反而增加課題。
- ◆ 機器學習目前學到的只是基礎，各種預測方式需要了解更多，可用的方式選擇才會更多，像是避免過度擬合的方法...等，同樣需要加強統計與各種得分的知識。

https://pyecontech.com/2019/12/28/python_regression/

參考資料補充

中央氣象局臺灣長期氣候變化全書下載：

https://www.cwb.gov.tw/V8/C/K/Encyclopedia/climate/climate7_all.html



END

