主題:繪製生活機能與安全的指標

組長:

統計四 C 408650033黃翔筠 01 號

組員:

統計四 A 408650108顏 豊 03 號

統計四 B 408650777張洧承 08 號

數資三 409200119阮柏誠 15 號

目錄

[**第一章 緒論** 5](#_Toc122637390)

[**1.1 前言** 5](#_Toc122637391)

[**1.2 研究動機** 5](#_Toc122637392)

[**1.3 研究目的** 5](#_Toc122637393)

[**1.4 研究流程** 6](#_Toc122637394)

[**第二章 資料介紹** 7](#_Toc122637395)

[**2.1 變數介紹** 8](#_Toc122637396)

[**2.2 資料處理** 8](#_Toc122637397)

[**2.3 敘述統計量** 9](#_Toc122637398)

[**第三章 研究分析** 11](#_Toc122637399)

[**3.1 各變數討論** 11](#_Toc122637400)

[3.1.1 安全上層面 11](#_Toc122637401)

[3.1.2 便利性層面 12](#_Toc122637402)

[**3.2 分群** 13](#_Toc122637403)

[**3.3 總結** 14](#_Toc122637404)

[**第四章 繪製指標** 15](#_Toc122637405)

[**4.1 迴歸分析** 15](#_Toc122637406)

[4.1.1 模型假設 15](#_Toc122637407)

[4.1.2 配適所有解釋變數 16](#_Toc122637408)

[4.1.3 配適距離變數 17](#_Toc122637409)

[4.1.4 比較兩者優劣 17](#_Toc122637410)

[**4.2 隨機森林** 18](#_Toc122637411)

[4.2.1 第一種模型 18](#_Toc122637412)

[4.2.2 第二種模型 18](#_Toc122637413)

[4.2.2 第三種模型 19](#_Toc122637414)

[4.2.4 比較三種模型優劣 20](#_Toc122637415)

[**4.3 XGBoost模型** 20](#_Toc122637416)

[4.3.1 第一種模型 20](#_Toc122637417)

[4.3.2 第二種模型 21](#_Toc122637418)

[4.3.3 第三種模型 21](#_Toc122637419)

[4.3.1 比較三種模型優劣 21](#_Toc122637420)

[**4.4 總結** 22](#_Toc122637421)

[**第五章 結論** 23](#_Toc122637422)

[**5.1 結論** 23](#_Toc122637423)

[**5.2 範例** 23](#_Toc122637424)

[**附錄** 24](#_Toc122637425)

[**程式碼** 24](#_Toc122637426)

[Orange3: 24](#_Toc122637427)

[SAS box-cox model: 24](#_Toc122637428)

[R: 0](#_Toc122637429)

圖目錄

[圖1研究流程圖 6](#_Toc122637430)

[圖2新北房價分層設色圖 7](#_Toc122637431)

[圖3 960筆房屋每單位價格之盒方圖與直方圖 9](#_Toc122637432)

[圖4 960筆各變數之盒方圖 10](#_Toc122637433)

[圖5 建築材料和有無警衛對每單位房價影響的熱圖 11](#_Toc122637434)

[圖6有無警衛對每單位房價影響 12](#_Toc122637435)

[圖7房價和距離變數的散佈圖 13](#_Toc122637436)

[圖8房價和距離變數的相關係數熱圖 13](#_Toc122637437)

[圖9距離變數的原始資料熱圖 13](#_Toc122637438)

[圖10距離變數的分群熱圖(Pearson) 14](#_Toc122637439)

[圖11迴歸模型假設檢定 15](#_Toc122637440)

[圖12殘差圖 16](#_Toc122637441)

[圖13 選取比較圖 16](#_Toc122637442)

[圖14 第一種模型變數的重要性指標 18](#_Toc122637443)

[圖15 第二種模型變數的重要性指標 19](#_Toc122637444)

[圖16 第三種模型變數的重要性指標 19](#_Toc122637445)

[圖17 第三種模型變數的重要性指標 20](#_Toc122637446)

[圖18 第三種模型變數的重要性指標 21](#_Toc122637447)

[圖19 第三種模型變數的重要性指標 21](#_Toc122637448)

[圖20 第三種模型變數的重要性指標 23](#_Toc122637449)

表目錄

[表1變數說明 8](#_Toc122637450)

[表2 960筆房屋連續型統計量 10](#_Toc122637451)

[表3 建材種類下anova表 11](#_Toc122637452)

[表4 給定有無警衛下的anova表 12](#_Toc122637453)

[表5 所有模型配適結果 16](#_Toc122637454)

[表6 距離變數配適結果 17](#_Toc122637455)

[表7 兩模型比較結果 17](#_Toc122637456)

[表8 兩模型比較結果 20](#_Toc122637457)

[表9 兩模型比較結果 22](#_Toc122637458)

[表10 兩模型比較結果 22](#_Toc122637459)

**第一章 緒論**

**1.1 前言**

主要的資料來源為台灣內政部不動產成交案件實際資訊資料供應系統，時間是從2014年到2022年，近9年來的房價資訊，其反應變數為房屋每平方公尺價格，解釋變數有建材種類、有無保安、屋齡。至於額外的解釋變數為google map的資訊補充，其資料為距離變數(公尺)，其中以買房地點為出發點到公車站、便利商店、郵局、警察局、學校、醫院的距離，距離計算採用的方式為實際上在走的最短路線。

**1.2 研究動機**

主要族群是針對一般的普羅大眾，在資料部分有充分的考慮到人的四個階段，分別為出生、幼年、壯年與老年的時期考量，其選擇新北市地區的原因有兩個，其一為新北市是台灣人口最多的縣市，其二為地理接壤許多縣市，為重要交通樞紐。我們的主題為繪製生活機能與安全指標，提供大眾一個參考依據以及兼顧到價位的考量。

**1.3 研究目的**

目的主要可以分為兩大類，分別為生活機能和安全。在生活機能的部分，以

便利商店和郵局為生活上便利的探討，至於學校為在教育上的考量，也進一步去研究傳統的觀念對於現代人是否有改變，最後是醫院的部分為健康需求的考量對於房價是否有影響。在安全的部分，以警察局和有無保安作為心理上安全感去看對於房價之間的變化，至於屋齡和建材的部分，由於台灣位於歐亞大陸以及太平洋海板塊的交界處，因此地震相較於其他國家是較為頻繁，所以屋齡和建材就是不錯的指標作為物理上的安全，再去探討對房價影響程度。

最後，繪製出生活機能指標是可以作為判斷房價價格高低的依據，有別於其他模型要取得房屋本身的資訊，我們的指標只要透過查詢房屋與標的物距離，就可以進行房價預測，甚至還沒蓋的預售物都可以給出房價的高低，對於建商和購屋者都是不錯的選擇工具。

**1.4 研究流程**

首先在第一章的時候，詳述了我們的研究動機和目的要繪製出生活指標，提供大眾一個參考生活機能的依據，第二章會介紹資料的型態以及做資料探索式分析，嘗試去解釋一些所呈現上的資訊，而第三章第一節會透過變異數分析或是視覺化呈現的方式，去檢驗在安全上或是機能上是否對房價有影響，第二節會嘗試觀察距離變數間是否有族群的存在。第四章，目標是想用生活指標去判別房價高低，有別於傳統模型需要很多變數解釋最大變異，而且生活指標的變數也相對好取得，所以這一章將會證明生活指標是跟房價是有相關，而且比傳統的模型更優秀。因此，我們會先配適所有變數模型，會嘗試一般迴歸、隨機森林、XGBoosting，再利用測試集來進行選出預測能力最好的模型，第二步就是繪製生活指標，只配適距離變數的情況之下的模型，去比較與所有變數模型的預測能力和解釋程度會不會差太多。第五章，為最後的結論的部分，主要會總結以上的資訊重點，以及開始介紹儀表板使用和利用甚麼模型去做房價的計算。

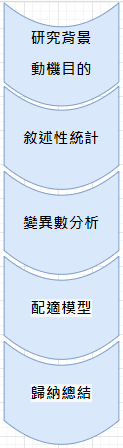


圖1研究流程圖

**第二章 資料介紹**

在進入資料介紹之前，首先要先了解收集對象，而我們所收集的資料都來自新北市，由下方圖2可以看出新北市各行政區的房價熱圖，其中灰色的區域代表資料本身來源的不足，並不足以畫出熱圖。回到主題，我們的目的是想透過房價來配適生活機能指標，因此房價就相當的重要，從行政區劃分的分層設色圖可以知道越靠近中心位置其每單位房價就越來越高，如:永和、中和和板橋等，最低則是萬里。若單純從圖形看是可以發現一些行政區房價比較是有不同的，從分層設色圖是可以當作初步認識資料對象還不錯的方式。

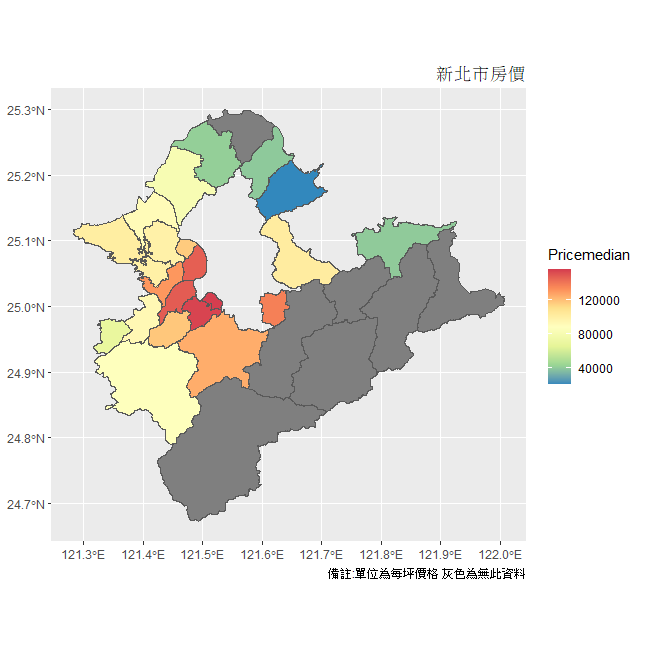


圖2新北房價分層設色圖

**2.1 變數介紹**

由表2來進行變數上的說明，首先定義房價的計算式為房屋價值扣除車位價值再除以總建物面積單位，若直接用房屋的總價值去衡量不太符合實際情況，因為建物面積的懸殊差距會導致房價的差距也進一步拉大，因此透過上面手法得到單純的建物每平方公尺的價格，有助於接下來配適生活機能指標的權重，第二個變數為屋齡，而屋齡的計算方式為建造完成日期減去完成交易日，若為負值屬於正常現象，因為市面上有預售屋的發行，因此交易會比實際建造完成日期來得早，單位是以日為計算，主要原因是更好去精確計算，因為若是要以年計算的話會有年要以365天或是366天計算的問題。第三個距離變數，標的物有公車站、便利商店、醫院、郵局、警察局和學校等，定義為由出發地到實際標的物的距離，使用了GoogleMap進行輔助，單位統一以公尺計算。第四個變數為建材，經整理後分別為鋼骨構造SC、鋼骨鋼筋混凝土SRC、鋼筋混凝土RC、加強磚造等。

表1變數說明

|  |  |  |
| --- | --- | --- |
| **變數** | **單位** | **說明** |
| 房價 | 元/每平方公尺 | 反應變數 |
| 屋齡 | 天 | 建造完成日期扣掉完成交易日期 |
| 距離變數 | 公尺 | 從出發地到實際到標的物的距離 |
| 建材 |  | 主要分為SC、SRC、RC、磚頭等分類 |
| 有無警衛 |  | 分為有警衛和沒有警衛 |

**2.2 資料處理**

進行資料處理第一個面對到的問題是遺失值的處理，在所有992筆資料裡面，唯有屋齡的變數有32筆的遺失值，這些遺失值是不可考的，對於遺失值處置，我們不採取插補的方法。原因分為兩點，第一點由於我們的遺失值占資料集的比重占不到5%，若是直接進行插補對執行後的結果影響不大。第二點，若是直接去做插補，我們也無法確定插補過後的結果是否正確，只能比較插補是否有影響，而這並不是我們所關心的議題。

由底下圖3可以看出房屋每平方公尺價格的分佈呈現為右偏分佈，有六筆的明顯離群值，以房價高到低進行排序，地點分別為新北市中和區連城路１２７巷１８號、新北市林口區文化二路一段１２６號、新北市永和區中正路６６６巷２９號、新北市新莊區福德二街７８號、新北市三重區大同南路１２９巷１０號、新北市土城區廣興街３４號等，其中位數為每平方公尺111546元。

第二步離群值的處理上，以最右邊的三點最為嚴重，因此我們想去找可歸屬原因，進一步去合理的剔除資料，避免影響最後的預測結果。第一筆資料新北市中和區連城路，並沒有找到可能的可歸屬原因，若是將此資料放入模型將會嚴重破壞模型的一致性，所以應該拉出來個別討論。第二筆為新北市林口區文化二路一段，由於地理位置過於特殊隔壁就是商場和電影院，但這些是在我們的考慮範圍之外，因此給予剔除。最後是新北市永和區中正路為私人高級別墅，因此無法和一般的房價進行比較應該刪除。經整理後，房屋每平方公尺價格的分佈有近似對稱情形。

|  |  |
| --- | --- |
|  |  |

圖3 960筆房屋每單位價格之盒方圖與直方圖

**2.3 敘述統計量**

反應變數為每單位的總價格，其資料中位數(111319)與平均數(118987)相差約7500左右，分佈呈現稍微右偏，並且由圖4我們可以看出一共有3點離群值的產生。

解釋變數部分，第一個為交易日到建造日的日期，其中最小值為-1056，有負數的原因是因為先交易後才建造完成的，就是市面上所販售的預售屋，資料方面中位數(7280)與平均數(7303)相近，分布應較為對稱。第二個部分為房屋距離有關的資料，分別有公車、便利商店、郵局、警察局、學校和醫院等，分布上呈現稍右偏，從圖4可以發現距離變數都有離群值的存在發生，數量上也不小想猜彼此之間可能有族群分佈，所以在第三章的時候會透過分群手法，找出潛在的族群。

表2 960筆房屋連續型統計量

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 每單位房價  (新台幣) | 交易日扣除建造日 | 公車 | 商店 | 郵局 | 警局 | 學校 | 醫院 |
| 最小值 | 15811 | -1056 | 78 | 66 | 60 | 36 | 18 | 38 |
| Q1 | 87651 | 2189 | 1700 | 1300 | 1400 | 1300 | 1300 | 1800 |
| 平均數 | 118987 | 7303 | 2918 | 2776 | 3006 | 2701 | 2686 | 3419 |
| 中位數 | 111319 | 111319 | 2600 | 2200 | 2500 | 2400 | 2300 | 2700 |
| Q3 | 118987 | 11203 | 3800 | 3800 | 3900 | 3600 | 3500 | 4400 |
| 最大值 | 261037 | 20728 | 10800 | 12700 | 19600 | 13700 | 13200 | 13000 |

|  |  |
| --- | --- |
| **每單位房價** | **距離變數** |
|  |
| **交易日到建造日** |
|  | 圖4 960筆各變數之盒方圖 |

**第三章 研究分析**

**3.1 各變數討論**

3.1.1 安全上層面

對於建築材料和有無警衛對每單位房價的影響，在這裡我們利用熱圖的手法去簡單分析，在建立在不同建築材料或有無警衛的條件下，是否對每單位房價有著明顯的視覺化的差異，在圖一的圖形中建築材料變數有特別，以順序的方式排序過，根據前面新北市建築執照工程造價標準表，從高到低排序分別為SC鋼骨結構、SRC鋼骨鋼筋混凝土構造、RC鋼筋混凝土、Brick磚頭等，首先由圖5 a的圖中，可以發現到給定在有警衛的條件之下，建築材料從左到右每單位房價有逐漸遞增的現象發生，但在沒警衛的條件下，就沒有現象趨勢去歸納。但是圖5 a中的個數，在沒警衛情況下個數有偏少的狀況發生，由於資料樣本數所受限所導致的，意味著不太適合以警衛特徵去做切割，因此決定以單一建材種類去分析較為恰當，由圖5 b可以發現SC鋼骨結構和SRC鋼骨鋼筋混凝土構造對RC鋼筋混凝土和Brick磚頭是有差異，透過變異數分析可以發現到p-value遠小於α值0.05，並且我們有足夠證據證明至少有一種建材種類下的每單位房價有顯著差異與圖5 b表現一致。

|  |  |
| --- | --- |
| **圖5 a 有無警衛和建材種類交叉分析** | **圖5 b建材種類** |

圖5 建築材料和有無警衛對每單位房價影響的熱圖

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 自由度 | SS | MS | F | P值 |
| 建物種類 | 3 |  |  | 18.26 | 0.0001<α=0.05 |
| 殘差 | 953 |  |  |  |  |

表3 建材種類下anova表

若只單純觀察警衛的條件之下，有警衛的房屋平均每單位房價為122120元比沒警衛的房屋平均每單位109307來的高，有無警衛房屋的分配都呈現些許的右偏，兩筆資料看起來很接近，因此我們就利用變異數分析手法去檢驗在有無警衛的條件下每單位房價是否真的有差異。而檢驗結果發現，在給定顯著水準在之下，在不同有無警衛的條件下，房屋的每單位房價在平均數和變異數都有顯著差異，推論有無警衛的條件對每單位房價有顯著差異，而有警衛房屋相較無警衛的分散程度較高。

|  |  |
| --- | --- |
|  |  |

圖6有無警衛對每單位房價影響

表4 給定有無警衛下的anova表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 自由度 | SS | MS | F | P值 |
| 警衛 | 1 |  |  | 15.1 | 0.0001<α=0.05 |
| 殘差 | 955 |  |  |  |  |

3.1.2 便利性層面

從圖7和圖8六個距離變數和房價的散佈圖和相關係數可以發現，這些距離變數皆集中在一公里到四公里內，還有郵局、警察局、學校、便利商店和公車變數和每單位房價呈現低度負相關，而醫院變數相較其他變數可以看得出跟每單位房價有中度負相關，所以每當郵局、警察局、學校、便利商店、公車和醫院距離房屋越遠房價會隨之增加，因此我們打算在下一節的部分進行分群族群劃分。

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

圖7房價和距離變數的散佈圖

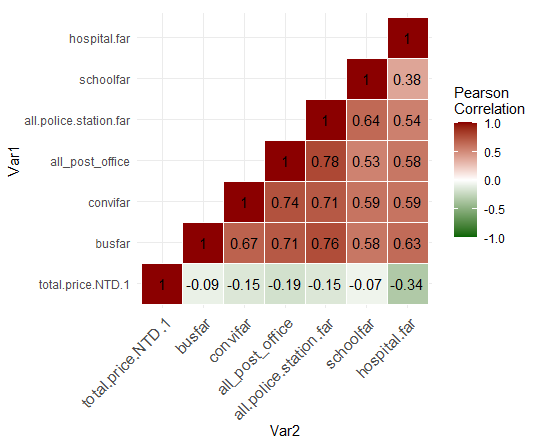
****

圖8房價和距離變數的相關係數熱圖

**3.2 分群**

圖9利用k-means對原始資料的距離變數做族群的集中共分為20群，再利用歐式的最大值距離去畫出樹狀，從圖和樹狀分的結果可以發現到大致上可以分為三群，分別為高機能族群、中機能族群、低機能族群等，但是唯有醫院變數相對於其他五個的變數有些許格格不入，從歐式距離的樹狀分法也可以發現這一點。另外，圖10利用皮爾森相關係數分群來看距離變數之間的關係，反而得到不一樣的結果學校距離變數和另外五個的距離變數比較不一樣。

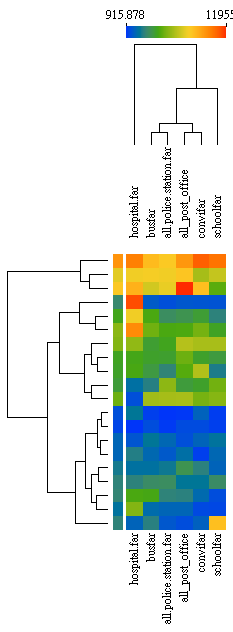


圖9距離變數的原始資料熱圖

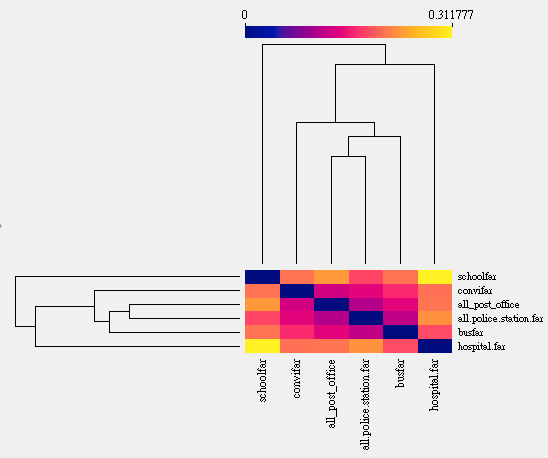


圖10距離變數的分群熱圖(Pearson)

**3.3 總結**

透過各變數的討論裡，先從安全上層面的著手可以知道對於建材的選擇以及警衛的有無對於房價是有顯著的差異，但建材不是每種都有差異，而是SC鋼骨結構和SRC鋼骨鋼筋混凝土構造對RC鋼筋混凝土和Brick磚頭是兩兩之間有差異，其次是便利性的層面，唯一找出特別變數是醫院相較其他距離變數可以看得出跟每單位房價有中度負相關為-0.3384，其餘距離變數跟房價關係就沒有這麼明顯。

最後，利用分群的手法想試圖找出潛在的族群分布，而從首先利用k-means分群和歐式距離畫樹狀，可以知道可以分為高中低三種機能的族群分布，再觀察距離變數可以得知醫院變數和其他變數的差異性是比較大的，最後，換成皮爾森相關係數畫樹狀，反而得到截然不同的結果，學校的趨勢長相跟別人的差異反而比較大。

**第四章 繪製指標**

這一章主要會做三種方式去做配適模型，分別為一般的迴歸模型、隨機森林和XGBoost。除此之外，做法會設置好分訓練集和測試集，其比例分別為8比2，資料來源為資料清理後的957筆的資料，再以訓練集訓練模型，再以測試集測試模型的優劣。

**4.1 迴歸分析**

4.1.1 模型假設

首先我們會逐一檢驗迴歸模型的三個假設，分別為殘差服從常態性、殘差項之間要互相獨立、變異數同質性等。第一個為殘差服從常態性，我們利用Shapiro-Wilk檢定以及搭配圖形的方式，查看P值是否有足夠證據拒絕常態性假設，在這次的模型裡雖然點大部分都落在直線上，但是頭尾部分有偏離的現象，P值也小於0.05，因此處理手法上採用SAS套件裡的box-cox方法去做轉換，其λ選取由程式決定最佳值，在以下模型都選用λ=0.5做為調整，第二個殘差項之間要互相獨立，從殘差圖並沒有看到任何可疑的曲線趨勢，第三個為變異數同質性假設，從下方圖9殘差圖可以發現沒有特別的突兀的點，而P值為0.8985大於α值0.05，因此不拒絕H0 ，所以沒有明顯證據證明誤差項變異數無均質性。此外，我們還有另外做共線性檢驗的步驟，避免遇到解釋變數間有高度相似使模型失效，而結果可以得知變異數膨脹因子均小於10，因此各變數之間都無共線性的問題。

|  |  |
| --- | --- |
| (a)VIF 變異數膨脹係數 | (b)常態性檢定 |
| (c) 變異數均值檢定 |

圖11迴歸模型假設檢定

|  |  |
| --- | --- |
|  |  |

圖12殘差圖

4.1.2 配適所有解釋變數

針對所有解釋變數下去配適模型，利用向前、逐步和向後選取法，進行選模型。由於逐步選取法的兩種判定係數比另外兩個還高，因此最後選出的模型為逐步的選取法。再進行box-cox的模型轉換，模型如下表5，由此可知為醫院、屋齡、鋼筋、警察局和郵局對於房價是成反比，但其中警察局和郵局是不顯著的，還有是否選鋼筋為建材影響房價比重較大，而公車和便利商店對於房價是成正比關係。

|  |  |  |
| --- | --- | --- |
| 向前 | 逐步 | 向後 |

圖13 選取比較圖

表5 所有模型配適結果

|  |  |  |
| --- | --- | --- |
|  | 係數 | P值 |
| 截距 | 862.073938 |  |
| 醫院 | -0.028524 | <.0001 |
| 屋齡 | -0.007728 | <.0001 |
| 公車距離 | 0.019445 | <.0001 |
| 建材(鋼筋) | -92.915140 | <.0001 |
| 警察局 | -0.006234 | 0.1261 |
| 郵局 | -0.002763 | 0.3755 |
| 便利商店 | 0.007280 | 0.0190 |
|  | | |

4.1.3 配適距離變數

只挑選出距離變數作為預測的解釋變數，模型如下表6，由此可知為醫院、學校、警察局、郵局對於房價是成反比，意味著距離離得越近房屋價格越高，但其中學校和郵局是不顯著的，而醫院的影響是相對較大的，而公車和便利商店對於房價則是成正比關係，但便利商店不顯著。

表6 距離變數配適結果

|  |  |  |
| --- | --- | --- |
|  | 係數 | P值 |
| 截距 | 720.646018 |  |
| 醫院 | -0.026699 | <.0001 |
| 公車距離 | 0.023433 | <.0001 |
| 學校 | -0.000421 | 0.8876 |
| 警察局 | -0.009658 | 0.0336 |
| 郵局 | -0.002763 | 0.4251 |
| 便利商店 | 0.004920 | 0.1534 |
|  | | |

4.1.4 比較兩者優劣

在兩個模型的比較上可以發現對於全部變數經過篩選過後的模型，判定係數和MSE比單用距離變數配適房價模型的還好，但只用距離變數配適房價模型的MAPE是比較小的，所以box-cox轉換後全部變數的模型比較好。

表7 兩模型比較結果

|  |  |  |
| --- | --- | --- |
| 比較 配適模型 | 全部變數 | 距離變數 |
| 判定係數 | 0.1460404 | 0.08071148 |
| MSE | 1748593229 | 1882362625 |
| MAPE | 0.3380373 | 0.3358576 |

**4.2 隨機森林**

主要配適的模型有三種，第一種為將所有的變數皆加進來，有建材種類、有為警衛、屋齡、距離變數和新北市行政區的變數等、第二種為行政區的變數、距離變數和屋齡以及第三種為只有六個距離變數，最後以MAPE和MSE作為比較優劣，除此之外會進一步去分析模型裡變數的重要性以至於分出以上三種的模型分析。

4.2.1 第一種模型

第一種隨機森林模型的進行步驟，先設定模型中的結點每次抽取以10個變數都下去考慮去建每500棵決策樹，而結果從圖14看出無論是%IncMSE和不純度指標比例很重要的變數從高到低排序分別為屋齡、醫院距離、新北市行政區和公車站距離，而其他的變數與前三高的變數的不純度指標就有一點差距，最讓人意外的是建材種類和警衛有無，反而對模型的影響最低的，所以下一個模型將此兩個剃除，最後比較模型的依據MAPE和MSE分別為19.87%和766701223。

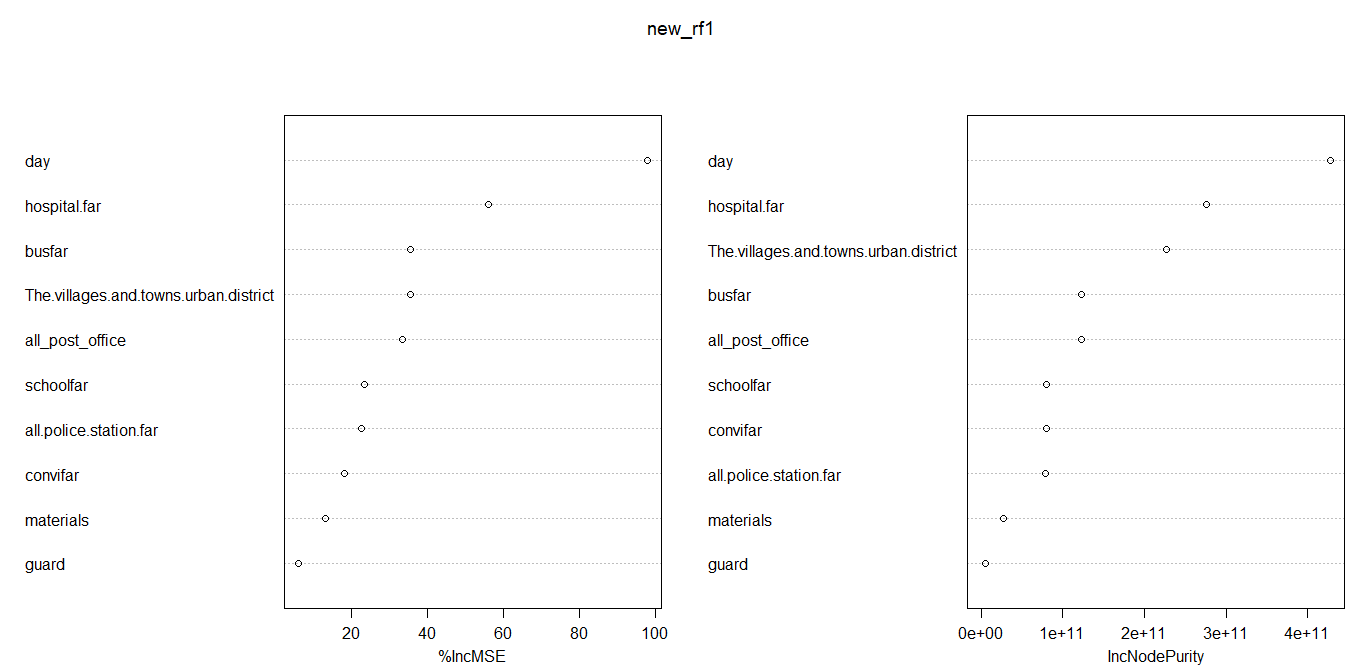


圖14 第一種模型變數的重要性指標

4.2.2 第二種模型

第二種隨機森林模型的進行步驟，一樣先設定模型中的結點每次抽取以8個變數都下去考慮去建每500棵決策樹，而結果從圖15發現無論是%IncMSE或是不純度指標比例很重要的變數一樣是屋齡、醫院距離和新北市行政區等，與上個模型結果沒有差異，最後比較模型的依據MAPE和MSE分別為19.80%和756755738，發現剔除建材種類和警衛有無變數後的結果預測的精確度有上升，所以可以推論建材種類和警衛有無的確對預測房價並沒有幫助。

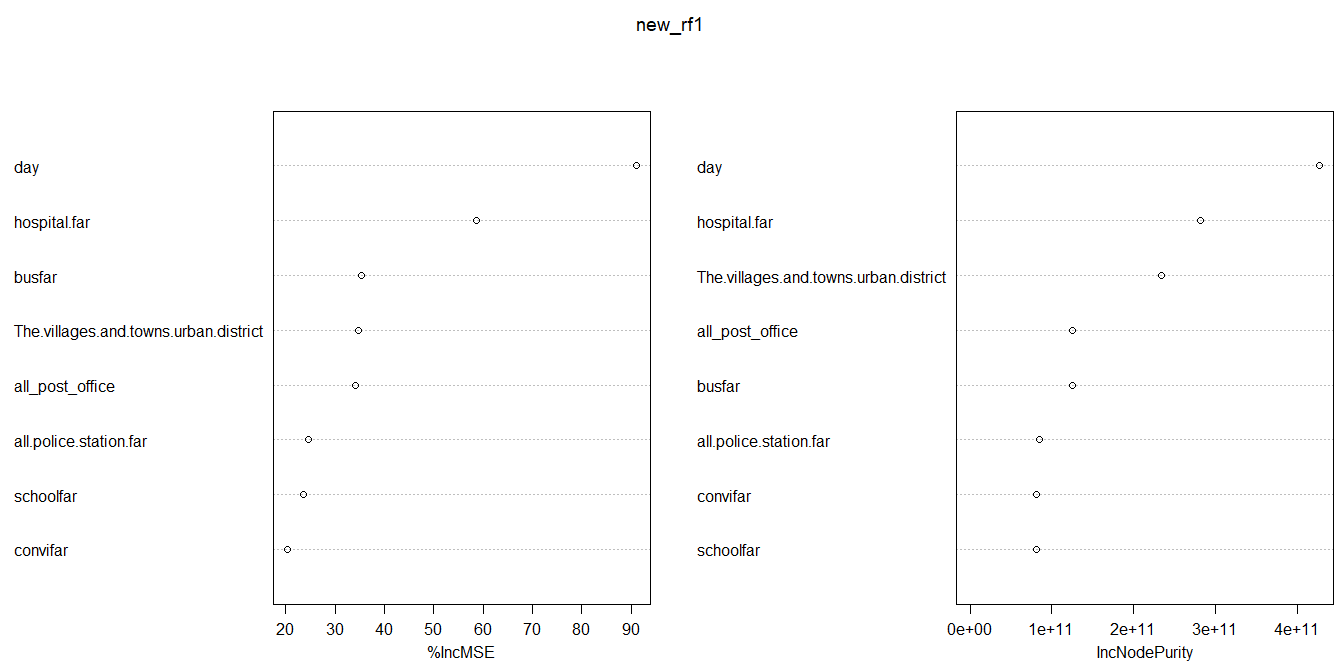


圖15 第二種模型變數的重要性指標

4.2.2 第三種模型

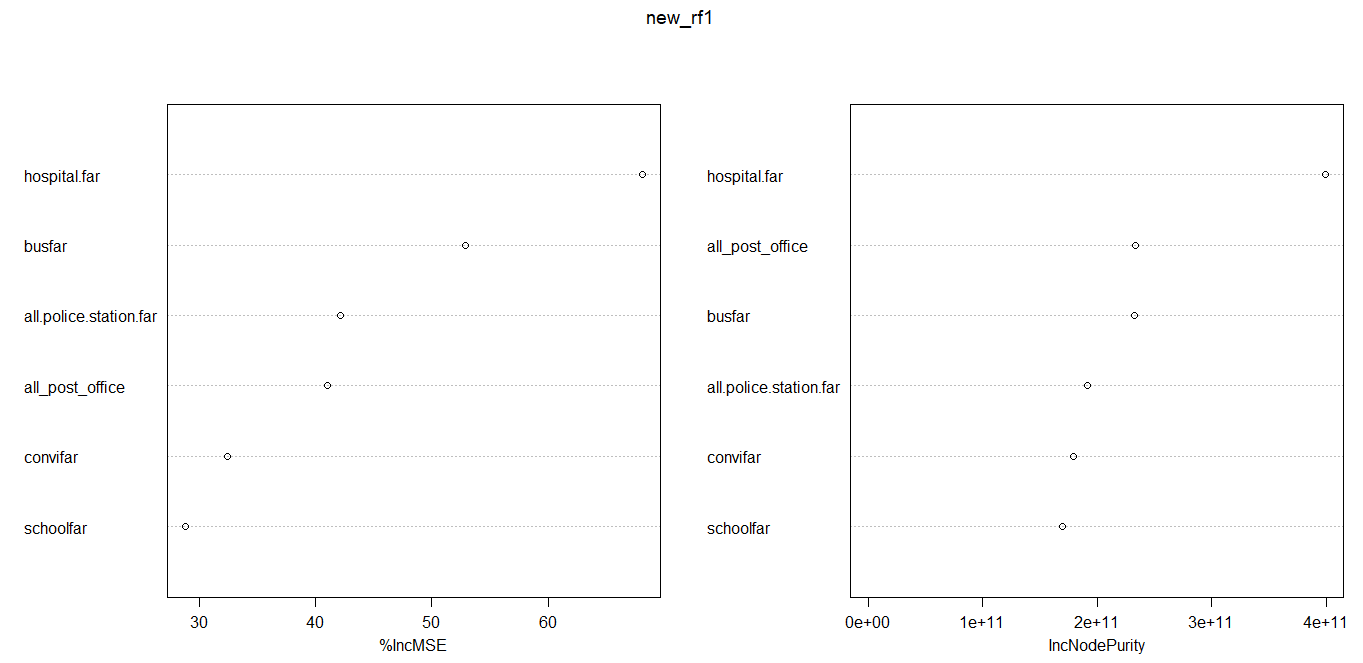
第三種隨機森林模型的進行步驟，設定模型中的結點每次抽取以6個變數都下去考慮去建每500棵決策樹，而結果從圖16看出無論是%IncMSE和不純度指標比例，很重要的變數還是有醫院距離和公車站距離，最後比較模型的依據MAPE和MSE分別為24.40%和967011400，發現只有距離變數其預測結果的精確度有下降，所以可以推論行政區和屋齡對預測房價有一定重要性，而MAPE的數值24.40還算差強人意。

圖16 第三種模型變數的重要性指標

4.2.4 比較三種模型優劣

做完三個模型後，整理成表8可以篩選出到最佳的模型為第二種模型，其變數有屋齡、距離變數和行政區等，而貢獻度對大的變數為屋齡、醫院和行政區等，以及確認出建材種類和警衛有無變數對於模型幫助性不大。

表8 兩模型比較結果

|  |  |  |  |
| --- | --- | --- | --- |
|  | 第一種 | 第二種 | 第三種 |
| MSE | 766701223 | 756755738 | 967011400 |
| MAPE | 19.87% | 19.80% | 24.40% |

**4.3 XGBoost模型**

XGBoost主要配適的模型有三種，第一種為將所有的變數皆加進來，有建材種類、有為警衛、屋齡、距離變數和新北市行政區的變數等、第二種為行政區的變數、距離變數和屋齡以及第三種為只有六個距離變數，最後以MAPE和MSE作為比較優劣，除此之外會進一步去分析模型裡變數的重要性以至於分出以上三種的模型分析，而XGBoost套件裡採用的分析模式為最小平方法、遞迴次數為150次、使用的手法是使用預設的tree。

4.3.1 第一種模型

第一種模型結果從圖17得知相對重要指標比例的變數排序從高到低分別為屋齡、醫院距離、新北市行政區，而其他的變數與前三高的變數的比例，就有開始有些許差異，而是建材種類和警衛有無為相對不重要的，所以下個模型將此兩個剃除，再去比較模型的MAPE和MSE是否有提升，最後分別第一種模型的MAPE和MSE分別為19.41%和768379226。

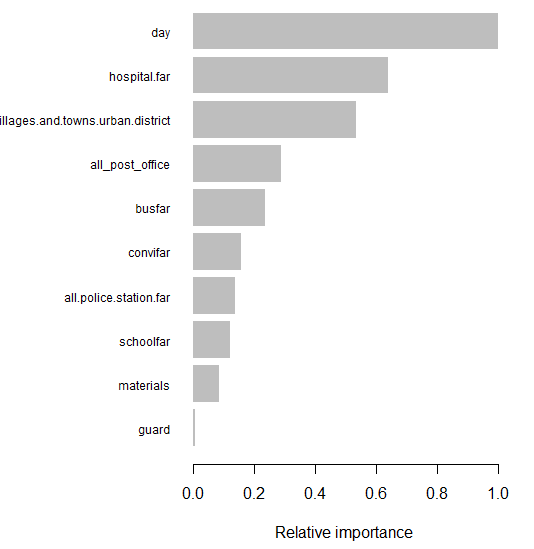


圖17 第三種模型變數的重要性指標

4.3.2 第二種模型

第二種模型扣除警衛和建材種類剩下8個變數，結果從圖18得知相對重要指標比例的變數排序從高到低分別為屋齡、醫院距離、新北市行政區，與上一個模型結果無異，但是模型的MAPE和MSE為20.40%和87310837皆提升了一些，所以代表建材種類和警衛是有貢獻的，只是相對貢獻程度比較低。

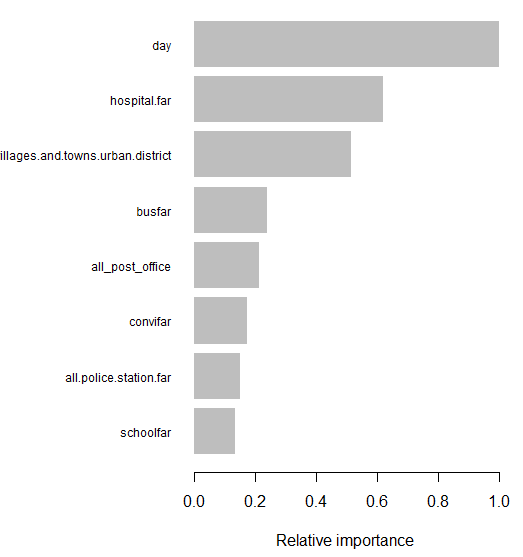


圖18 第三種模型變數的重要性指標

4.3.3 第三種模型

第三種模型為六個距離變數，結果從圖19知道相對重要指標比例的變數最為重要的是醫院，其他變數的重要性就都差不多，還可以發現跟相關係數的強弱的表現很類似，因為只有醫院與房價的相關係數比較高，其餘變數的相關係數為低相關，最後第三種的模型的MAPE和MSE為21.32%和905889792。

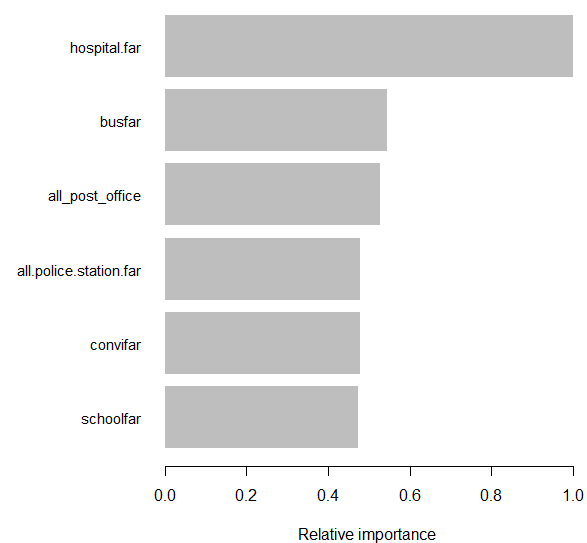


圖19 第三種模型變數的重要性指標

4.3.1 比較三種模型優劣

做完三個模型後，整理成表9可以篩選出到最佳的模型為第一種模型，其變數有屋齡、距離變數、行政區、警衛和建材等，但是我認為第二種的模型仍然可以考慮，因為兩者的MSE和MAPE的差異性並不大。

表9 兩模型比較結果

|  |  |  |  |
| --- | --- | --- | --- |
|  | 第一種 | 第二種 | 第三種 |
| MSE | 768379226 | 873108379 | 905889792 |
| MAPE | 19.41% | 20.40% | 21.32% |

**4.4 總結**

從一般迴歸、隨機森林和XGBoost的模型裡利用MSE和MAPE挑出最佳的模型，經整理後的表10可知以第二種隨機森林以及第一種XGBoost為最佳的MSE和MAPE，再以變數個數去比較以第二種隨機森林比較小，所以最後房價的預測模型的選擇為有行政區的變數、距離變數和屋齡的隨機森林，為什麼以變數個數最小作為選擇的依據，是因為第一種隨機森林模型還是第二種XGBoost模型MSE和MAPE的差異性不大，能用較少的資訊達到最達的目的是我們想要的，何況從前面不管是隨機森林還是XGBoost都顯示建材和警衛重要性是變數裡最低的。

除此之外，若是套用到現實生活中的例子，屋齡和建材的資訊收集上可能會有困難，所以我們另外找出只有距離變數最佳的模型為XGBoost其MAPE也不差，所以下一章介紹我們的預測網站的時候，使用的計算模型將是第二種隨機森林和XGBoosting兩種方式。

表10 兩模型比較結果

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Cox-box | | 隨機森林 | | | XGBoost | | |
| 所有 | 距離 | 一 | 二 | 三 | 一 | 二 | 三 |
| 變數個數 | 7 | 6 | 10 | 8 | 6 | 10 | 8 | 6 |
| MSE | 1748106 | 1882106 | 766106 | 756106 | 967106 | 768106 | 873106 | 905106 |
| MAPE | 33.80% | 33.58% | 19.87% | 19.80% | 24.40% | 19.41% | 20.40% | 21.32% |

**第五章 結論**

**5.1 結論**

針對以上探索式分析和模型的細節做個彙總，首先從敘述性統計的部分和直方圖，看出房價的分布近似鐘型分佈以及有離群值的存在發生，第二點在做安全上的研究的時候透過圖5和檢定發現警衛和建材對房價有顯著的差異，但後面做模型的時候卻被否定，判斷可能的原因是因為其他變數在解釋房價能力比較好，而在便利的層面上，醫院對房價的相關程度相較其他距離變數比較高，在後面不管是一般迴歸、隨機森林還是XGBoosting都證明依據的確對房價解釋能力，第三點利用k-means和階層式分群的方法，了解到k-means的分群方法距離變數可以分為三個族群分為遠、中、近的距離，但唯有醫院色階有點不同，從歐式距離的方式畫樹更可以知道醫院是與其他變數距離上是最遠的，而變數之間的關係若改用皮爾森相關係數的時候反而得到截然不同的結果，變成學校和其他變數的趨勢有很大不同。最後，在模型選擇上一般迴歸、隨機森林和XGBoosting裡，最好的模型為第二種隨機森林，若是要兼顧實用性只有距離變數模型，就是以XGBoosting的第三種為最好。

**5.2 範例**

從上一章得出的結論，若單純採用只有六個距離變數的部分，就採用XGBoosting計算，使用方法為填入左側6個距離變數，若是想更進階計算的更精準，可以再填入屋齡和行政區。實測一個範例，若有一個房屋地址為新北市新莊區青山路一段64之1號，其醫院、警察局、公車站、便利商店、郵局和學校其距離分為10900、3400、10800、12400、3400和13200公尺先輸入左側，右側就會得出簡易版的價格319683.381606914 NTD，再多輸入屋齡6138日和行政區新莊的話就會得到用隨機森林計算進階版的房價334877.443470746 NTD，與實際價格309964.759682差異不大。



圖20 第三種模型變數的重要性指標

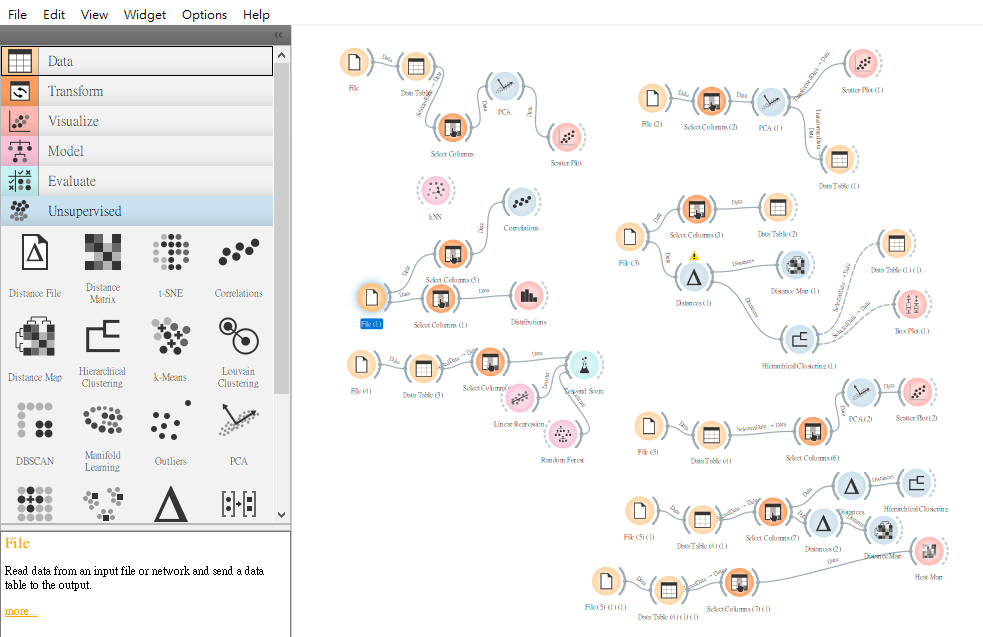
**附錄**

[](https://xiangyun2582.shinyapps.io/project/)

<https://xiangyun2582.shinyapps.io/project/>

**程式碼**

Orange3:



SAS box-cox model:

**PROC** **IMPORT** OUT= WORK.train80

DATAFILE= "C:\Users\Huang\Desktop\資料探勘\報告\train80.csv"

DBMS=CSV REPLACE;

GETNAMES=YES;

DATAROW=**2**;

**RUN**;

**PROC** **IMPORT** OUT= WORK.test20

DATAFILE= "C:\Users\Huang\Desktop\資料探勘\報告\test20.csv"

DBMS=CSV REPLACE;

GETNAMES=YES;

DATAROW=**2**;

**RUN**;

**proc** **print** data=test20;

**run**;

/\* 迴歸分析配適模型 \*/

**proc** **sort** data=train80 out=out;

by materials ;

**proc** **print**;

**run**;

**data** train;

set out;

by materials ;

if materials=**1** then do

ma1=**1**;ma2=**0** ; ma3=**0** ;end;

if materials=**2** then do

ma1=**0**;ma2=**1** ; ma3=**0** ;end;

if materials=**3** then do

ma1=**0**;ma2=**0** ; ma3=**1** ;end;

if materials=**4** then do

ma1=**0**;ma2=**0** ; ma3=**0** ;end;

**proc** **print**;**run**;

**proc** **reg** data=train;

model total\_price\_NTD\_1=day busfar convifar all\_post\_office all\_police\_station\_far schoolfar hospital\_far guard ma1 ma2 ma3/partial VIF;

plot rstudent. \*(predicted. day busfar convifar all\_post\_office all\_police\_station\_far schoolfar hospital\_far guard ma1 ma2 ma3 obs.);

plot NPP. \*RSTUDENT.;

output out=set1 residual=residual rstudent=rstudent student=student h=h press=press;

**proc** **print**;

var residual student h press rstudent;

**proc** **univariate** plot normal;

var residual;

**run**;

/\*Brown-Forsythe test\*/

**DATA** SET2;/\*Brown-Forsythe test\*/

SET SET1;

IF \_N\_ /**2**=int(\_N\_/**2**) THEN GROUP=**1**;

ELSE GROUP=**2**;

**PROC** **GLM**;

CLASS GROUP;

MODEL RESIDUAL=GROUP;

MEANS GROUP/HOVTEST=BF;

**RUN**;

\*forward;

**proc** **reg** data=train;

model total\_price\_NTD\_1=day busfar convifar all\_post\_office all\_police\_station\_far schoolfar hospital\_far guard ma1-ma3 /CLB selection=forward;

**run**;

\*stepwise;

**proc** **reg** data=train;

model total\_price\_NTD\_1=day busfar convifar all\_post\_office all\_police\_station\_far schoolfar hospital\_far guard ma1-ma3 /CLB selection=stepwise;

**run**;

**proc** **transreg** data=train test;

model BoxCox(total\_price\_NTD\_1) = identity(hospital\_far day busfar materials all\_police\_station\_far convifar all\_post\_office);

**run**;

\*backward;

**proc** **reg** data=train;

model total\_price\_NTD\_1=day busfar convifar all\_post\_office all\_police\_station\_far schoolfar hospital\_far guard ma1-ma3/CLB selection=backward;

**run**;

**proc** **transreg** data=train test SS2;

model BoxCox(total\_price\_NTD\_1) = identity(day busfar convifar all\_post\_office all\_police\_station\_far hospital\_far ma3);

**run**;

**proc** **transreg** data=train80 OUTTEST=m1 TEST SS2;

model BoxCox(total\_price\_NTD\_1) = identity(hospital\_far busfar all\_police\_station\_far convifar all\_post\_office schoolfar);

**run**;

R:

####爬蟲 抓距離資料#####

## package 匯入

check.packages <- function(pkg) {

# STEP 1 檢查機器裡有沒有我現在要用的套件

new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]

if (length(new.pkg)) {

#if>0執行

# 安裝

install.packages(new.pkg, dependencies = TRUE)

}

## STEP 2 我要用的套件 讀近來

sapply(pkg, require, character.only = TRUE)

}

# 我要用的

pkg <- c(

"data.table", "dplyr", "jiebaR", "textTinyR", "remoji", "colortools",

"showtext", "wordcloud", "tm", "tmcn", "Rwordseg", "stringr", "tidytext",

"ggplot2", "word2vec", "bigstep",'stringr',"rvest","RSelenium","xml2"

)

check.packages(pkg)

my\_data <- read.csv("目標資料集.csv")

convinience <- c(rep(0,992)) %>% as.data.frame()

for (i in 1:nrow(my\_data)) {

convinience[i,1]<-paste0(my\_data[i,2],"便利商店")

}

paste0(my\_data[i,2],"便利商店")

location <- c(rep(0,992)) %>% as.data.frame()

for (i in 1:992) {

location[i,1] <- paste0('https://www.google.com/maps/dir/',my\_data[i,4],"/",convinience[i,1])

}

location[2,1]

####進入到爬網階段####----

####動態網頁:jquery,Ajex####

library(RSelenium)

remDr=rsDriver(browser = c("firefox"),port=sample(1000:8000,1))

driver=remDr[["client"]]

driver$open()

# 瀏覽 Google 首頁 test

#driver$navigate("https://www.google.com.tw/")

convifar<- c(rep(0,992)) %>% as.data.frame()

for (i in 1:4) {

driver$navigate(location[i,1])

webpage <- read\_html(driver$getPageSource()[[1]][1])

box <- webpage%>% html\_nodes(css ='#section-directions-trip-0 .fontBodyMedium') %>% html\_text()

convifar[i,1] <- paste0(trimws(box[1]))

Sys.sleep(0.5)

}

#my\_data <- cbind(my\_data,convifar)

#names(my\_data)[36] <- c("convifar")

#my\_data <- my\_data[,-1]

#write.csv(my\_data,"目標資料集2.csv",fileEncoding = "big-5")

my\_data[35]my\_data[1,35]

for (i in 1:992) {

a[i,1] <- nchar(my\_data[i,35])

}

a<- c(rep(0,992)) %>% as.data.frame()

distinct(a)

my\_data <- cbind(my\_data,1:992)

names(my\_data)[36] <- c("num")

adjust <- my\_data %>% arrange(desc(my\_data[35])) %>% select(num)

adjust <- adjust[1:68,1]

View(my\_data %>% arrange(desc(my\_data[35])))

#box <- webpage%>% html\_nodes(css ='#section-directions-trip-0 .fontBodyMedium') %>% html\_text()

#convifar[5,1] <- paste0(trimws(box[1]))

#section-directions-trip-0 .XdKEzd

##.ivN21e div

#section-directions-trip-0 .fontBodyMedium div

####繪製地圖####

library(dplyr)

library(sf)

newcity.map <-

st\_read("C:/Users/Huang/Desktop/研究/bigdata/mapdata202209220431/TOWN\_MOI\_1100415.shp") %>% filter(COUNTYNAME == "新北市")

#print(newcity.map)

dim(newcity.map)

names(newcity.map)

plot(newcity.map[1])

plot(st\_geometry(newcity.map))

newcity.map$TOWNNAME

my.newcity.map <- newcity.map[c("TOWNNAME", "geometry")]

my.newcity.map$TOWNNAME <- as.character(newcity.map$TOWNNAME)

head(my.newcity.map)

y <-

mydata10 %>% select(The.villages.and.towns.urban.district, total.price.NTD.1) %>% group\_by(The.villages.and.towns.urban.district) %>% summarise(Pricemedian=median(total.price.NTD.1))

my.taiwan.map.data <- left\_join(my.newcity.map,y,

by = c("TOWNNAME" = "The.villages.and.towns.urban.district"))

library(ggplot2)

ggplot(data = my.taiwan.map.data) +

geom\_sf(aes(fill = Pricemedian))+

scale\_fill\_distiller(palette = "Spectral")+

ggtitle("新北市房價")+

theme(plot.title = element\_text(hjust = 1))+

labs(caption = "備註:單位為每坪價格 灰色為無此資料")

####相關係數熱圖####

mydata <- mtcars[, c(1,3,4,5,6,7)]

head(mydata)

cormat <- round(cor(mydata),2)

library(reshape2)

melted\_cormat <- melt(cormat)

head(melted\_cormat)

melted\_cormat <- melt(cormat)

head(melted\_cormat)

# Get lower triangle of the correlation matrix

get\_lower\_tri<-function(cormat){

cormat[upper.tri(cormat)] <- NA

return(cormat)

}

# Get upper triangle of the correlation matrix

get\_upper\_tri <- function(cormat){

cormat[lower.tri(cormat)]<- NA

return(cormat)

}

upper\_tri <- get\_upper\_tri(cormat)

upper\_tri

# Melt the correlation matrix

library(reshape2)

melted\_cormat <- melt(upper\_tri, na.rm = TRUE)

# Heatmap

library(ggplot2)

ggplot(data = melted\_cormat, aes(Var2, Var1, fill = value))+

geom\_tile(color = "white")+

scale\_fill\_gradient2(low = "darkgreen", high = "darkred", mid = "white",

midpoint = 0, limit = c(-1,1), space = "Lab",

name="Pearson\nCorrelation") +

theme\_minimal()+

theme(axis.text.x = element\_text(angle = 45, vjust = 1,

size = 12, hjust = 1))+

coord\_fixed()

data <- read.csv("目標資料集10.csv")

names(data)

mydata <- data[, c(1,5,6,7,8,9,10)]

head(mydata)

cormat <- round(cor(mydata),2)

library(reshape2)

melted\_cormat <- melt(cormat)

head(melted\_cormat)

melted\_cormat <- melt(cormat)

head(melted\_cormat)

upper\_tri <- get\_upper\_tri(cormat)

upper\_tri

# Melt the correlation matrix

library(reshape2)

melted\_cormat <- melt(upper\_tri, na.rm = TRUE)

# Heatmap

library(ggplot2)

ggplot(data = melted\_cormat, aes(Var2, Var1, fill = value))+

geom\_tile(color = "white")+

geom\_text(aes(Var2, Var1, label = value), color = "black", size = 4)+

scale\_fill\_gradient2(low = "darkgreen", high = "darkred", mid = "white",

midpoint = 0, limit = c(-1,1), space = "Lab",

name="Pearson\nCorrelation") +

theme\_minimal()+

theme(axis.text.x = element\_text(angle = 45, vjust = 1,

size = 12, hjust = 1))+

coord\_fixed()

####保安和警衛熱圖和箱型圖####

mydata10 <- read.csv("目標資料集10.csv")

windows()

ggplot(mydata10, aes(x = mydata10[, 1])) +

geom\_boxplot(

fill = "slateblue",

alpha = 0.2,

size = 0.1,

outlier.shape = 19,

outlier.size = 2,outlier.color = "red"

)+

labs(x = '每坪價格',

y = '',

title = '房價的盒方圖') +

labs(caption = "註記:為已去除遺失值後960筆")

adjdata <-mydata10 %>% arrange(mydata10[,1])

adjdata <-adjdata[1:957,]

ggplot(adjdata, aes(x = adjdata[, 1])) +

geom\_boxplot(

fill = "slateblue",

alpha = 0.2,

size = 0.1,

outlier.shape = 19,

outlier.size = 2,outlier.color = "red"

)+

labs(x = '每坪價格',

y = '',

title = '房價的盒方圖')

summary(mydata10)

distinct(adjdata [11])

set.seed(3347)

names(adjdata)

adjdata1 <- adjdata %>% group\_by(materials,guard) %>% summarise(price=mean(total.price.NTD.1)) %>%as.data.frame()

adjdata2 <- dcast(adjdata1,materials~guard)

row.names(adjdata2) <- adjdata2$materials

adjdata2 <- adjdata2[,-1]

windows()

class(adjdata2)

adjdata2 <- as.matrix(adjdata2)

# Dummy data

x <- adjdata1[1]

y <- adjdata1[2]

for (i in 1:8) {

if(adjdata1[i,2]==1){

adjdata1[i,4]="有保安"

}

if(adjdata1[i,2]==0){

adjdata1[i,4]="沒保安"

}

}

for (i in 1:8) {

if(adjdata1[i,1]==1){

adjdata1[i,5] <- "SRC"

}

if(adjdata1[i,1]==2){

adjdata1[i,5] <- "SC"

}

if(adjdata1[i,1]==3){

adjdata1[i,5] <- "RC"

}

if(adjdata1[i,1]==4){

adjdata1[i,5] <- "磚造"

}

}

data <- cbind(X=x, Y=y)

data$Z <- adjdata1[,3]

# Color Brewer palette

library(viridis)

library(hrbrthemes)

adjdata1$V5 <- factor(adjdata1$V5, ordered = TRUE, levels = c("SRC", "SC", "RC", "磚造"))

# Give extreme colors:

ggplot(adjdata1, aes(adjdata1$V5, adjdata1$V4 ,fill= adjdata1$price)) +

geom\_tile() +

scale\_fill\_gradient(low="white", high="blue") +

theme\_ipsum()+ggtitle("保全與建築類別對房價的影響")

library(dplyr)

library(ggplot2)

data <- read.csv("C:/Users/Huang/Desktop/資料探勘/報告/目標資料集10.csv",fileEncoding = "big-5")

names(data)

distinct(data[11])

x=data.frame()

for (i in 1:957) {

if(data[i,11]==1){

x[i,1]="SC"

x[i,2]=18440/7020

}

if(data[i,11]==2){

x[i,1]="SRC"

x[i,2]=15050/7020

}

if(data[i,11]==3){

x[i,1]="RC"

x[i,2]=11180/7020

}

if(data[i,11]==4){

x[i,1]="Brick"

x[i,2]=7020/7020

}

}

for (i in 1:957) {

if(data[i,12]==1){

x[i,3]="有警衛"

}

if(data[i,12]==0){

x[i,3]="沒警衛"

}

}

####heatmap----

library(reshape2)

names(x) <- c("materials","weights","guard")

allmatrix <- cbind(x,data[,1])

allmatrix <- allmatrix[,-2]

names(allmatrix)[3] <- "y"

allmatrix <- allmatrix %>% select(guard,materials,y) %>% group\_by(guard,materials)

allmatrix.dcast <- dcast(allmatrix,materials~guard)

table(allmatrix$guard,allmatrix$materials)

single <- cbind(x,data[,1])

single <- single[ ,-2]#run兩次

names(single)[2] <- "y"

single <- data.frame(single ,hei=rep(1,957)) %>% group\_by(materials) %>% summarise(mean=mean(y))

single <- data.frame(single ,hei=rep(1,4))

single$materials <- factor(single$materials, levels = c("SC","SRC","RC","Brick"))

##single單一材料變數

ggplot(single, aes(x = materials, y = hei , fill = mean)) +

geom\_tile(colour = "white", size = 0.25) + # 繪製熱圖

scale\_y\_discrete(expand = c(0, 0)) + # 移除多餘空白

scale\_x\_discrete(expand = c(0, 0)) + # 移除多餘空白

coord\_fixed() + # 設定 X 與 Y 軸等比例

scale\_colour\_brewer(palette = "Greens") + # 設定色盤

theme(

legend.text = element\_text(face = "bold"),

# 說明文字用粗體

axis.ticks = element\_line(size = 0.5),

# 座標軸上的刻度寬度

plot.background = element\_blank(),

# 移除背景

panel.border = element\_blank(),

# 移除邊框

axis.text.x = element\_text(

angle = 90,

vjust = 0.5,

hjust = 1

) # X 軸文字轉向

)+xlab("建築材料的種類") + ylab("") + labs(title = "建築材料對每坪房價影響的熱圖", caption = "備註:建材價格比 SC:SRC:RC:Brick=2.63:2.14:1.59:1")

goal <- cbind(x,data[,1]);names(goal)[4] <-c("y")

goal <- goal %>% group\_by(materials,guard) %>% summarise(mean=mean(y))

#a <- goal[,-3]

goal.melt <- dcast(goal,materials~guard)

rownames(goal.melt) <- goal.melt$materials

#goal.melt <- goal.melt[,-1] %>% as.matrix()

library(RColorBrewer)

##交叉分析----

goal$materials <-

factor(goal$materials, levels = c("SC", "SRC", "RC", "Brick"))

windows()

text <- c(2, 41, 686, 191, 30, 1, 15, 1)

goaltext <- cbind(goal, text)

names(goaltext)[4] <- "text"

ggplot(goaltext, aes(x = materials , y = guard , fill = mean)) +

geom\_tile(colour = "white", size = 0.25) + # 繪製熱圖

scale\_y\_discrete(expand = c(0, 0)) + # 移除多餘空白

scale\_x\_discrete(expand = c(0, 0)) + # 移除多餘空白

coord\_fixed() + # 設定 X 與 Y 軸等比例

scale\_fill\_gradientn(colours = brewer.pal(7, "YlGnBu")) + # 設定色盤

theme(

legend.text = element\_text(face = "bold"),

# 說明文字用粗體

axis.ticks = element\_line(size = 0.5),

# 座標軸上的刻度寬度

plot.background = element\_blank(),

# 移除背景

panel.border = element\_blank(),

# 移除邊框

axis.text.x = element\_text(

angle = 90,

vjust = 0.5,

hjust = 1

) # X 軸文字轉向

) + xlab("建築材料的種類") + ylab("") + labs(title = "建築材料和有無警衛對每坪房價影響的熱圖", caption = "圖中的數字為個數 \n 備註:建材價格比 SC:SRC:RC:Brick=2.63:2.14:1.59:1")+geom\_text(label = goaltext$text)

####模型####

test20 <- fread("test20.csv",header = T)

train80 <- fread("train80.csv",header = T)

train <- train80 %>% select(total.price.NTD.1,hospital.far,day,busfar,materials,all.police.station.far,convifar,all\_post\_office)

rc=data.frame()

for (i in 1:nrow(train)) {

if (train[i,5]==3){

rc[i,1]=1

}

else{

rc[i,1]=0

}

}

train <- cbind(train,rc)

test=data.frame()

for (i in 1:nrow(test20)) {

if (test20[i,11]==3){

test[i,1]=1

}

else{

test[i,1]=0

}

}

test=cbind(test20,test)

names(test)[13] <- "rc"

names(train)[9] <- "rc"

boxcox <- function(hospital\_far,day,busfar,rc,all\_police\_station\_far,convifar,all\_post\_office){

lambday <- 862.073938-0.028524\*hospital\_far-0.007728\*day+0.019445\*busfar-92.915140\*rc-0.006234\*all\_police\_station\_far+0.007280\*convifar--0.002763\*all\_post\_office

fity <- (lambday\*0.5+1)^2

return(fity)

}

names(test20.x )

test20.x <- test[,c(10,4,5,13,8,6,7)] %>% data.frame()

fittedy <- data.frame()

summary(train80)#ymean118763

for (loop in 1:1) {

SSE <- 0

SST <- 0

MAPE <- 0

for (i in 1:nrow(test20)) {

fittedy[i, 1] <- boxcox(test20.x[i, 1],

test20.x[i, 2],

test20.x[i, 3],

test20.x[i, 4],

test20.x[i, 5],

test20.x[i, 6],

test20.x[i, 7])

SSE <- SSE + (test20[i, 1] - fittedy[i, 1]) ^ 2

SST <- SST + (test20[i, 1] - 118763) ^ 2

MAPE <- MAPE+(abs(test20[i, 1]-fittedy[i, 1])/test20[i, 1])/nrow(test20)

}

Rsquared <- 1 - SSE / SST %>% as.numeric()

MSE <- SSE /(nrow(test20)-7) %>% as.numeric()

MAPE <- MAPE %>% as.numeric()

print(c(Rsquared=Rsquared,MSE=MSE,MAPE=MAPE))

}

##單一只有跑距離變數

distance <- function(hospital\_far,busfar,all\_police\_station\_far,convifar,all\_post\_office,schoolfar){

lambday <- 720.646018-0.026699\*hospital\_far+0.023433\*busfar-0.009658\*all\_police\_station\_far+0.004920\*convifar-0.002707\*all\_post\_office-0.000421\*schoolfar

fity <- (lambday\*0.5+1)^2

return(fity)

}

test.x <- test20[,c(4,5,6,7,8,9,10)] %>% data.frame()

names(test.x)

fittedy <- data.frame()

for (loop in 1:1) {

SSE <- 0

SST <- 0

MAPE <- 0

for (i in 1:nrow(test20)) {

fittedy[i, 1] <- distance(test.x[i, 7],

test.x[i, 2],

test.x[i, 5],

test.x[i, 3],

test.x[i, 4],

test.x[i, 6])

SSE <- SSE + (test20[i, 1] - fittedy[i, 1]) ^ 2

SST <- SST + (test20[i, 1] - 118763) ^ 2

MAPE <- MAPE+(abs(test20[i, 1]-fittedy[i, 1])/test20[i, 1])/nrow(test20)

}

Rsquared <- 1 - SSE / SST %>% as.numeric()

MSE <- SSE /(nrow(test20)-7) %>% as.numeric()

MAPE <- MAPE %>% as.numeric()

print(c(Rsquared=Rsquared,MSE=MSE,MAPE=MAPE))

}

####Xgboosting####

require(xgboost)

train <- read.csv("train80orange.csv",fileEncoding = "utf-8")

test<- read.csv("test20orange.csv",fileEncoding = "utf-8")

####六個變數----

x.train <- data.matrix(train[,c(5,6,7,8,9,10)])

y.train <- train[,1]

x.test <- data.matrix(test[,c(5,6,7,8,9,10)])

y.test <- test[,1]

xgb\_train = xgb.DMatrix(data = x.train, label = y.train)

xgb\_test = xgb.DMatrix(data = x.test, label = y.test)

bst <- xgboost(data = xgb\_train, max.depth = 6, nrounds = 150,

objective = "reg:squarederror",eval\_metric="rmse")

#bst

pred\_y = predict(bst, x.test)

importance\_matrix <- xgb.importance(colnames(x.train), model = bst)

xgb.plot.importance(importance\_matrix, rel\_to\_first = TRUE, xlab = "Relative importance")

windows()

mse = mean((y.test - pred\_y)^2)

mse

MAPE <- mean(abs((y.test - pred\_y)/y.test))

MAPE

####行政+日期 all----

x.train <- data.matrix(train[,c(2,4,5,6,7,8,9,10,11,12)])

y.train <- train[,1]

x.test <- data.matrix(test[,c(2,4,5,6,7,8,9,10,11,12)])

y.test <- test[,1]

xgb\_train = xgb.DMatrix(data = x.train, label = y.train)

xgb\_test = xgb.DMatrix(data = x.test, label = y.test)

bst <- xgboost(data = xgb\_train, max.depth = 6, nrounds = 150,

objective = "reg:squarederror",eval\_metric="rmse")

#bst

pred\_y = predict(bst, x.test)

importance\_matrix <- xgb.importance(colnames(x.train), model = bst)

xgb.plot.importance(importance\_matrix, rel\_to\_first = TRUE, xlab = "Relative importance")

windows()

mse = mean((y.test - pred\_y)^2)

mse

MAPE <- mean(abs((y.test - pred\_y)/y.test))

MAPE

####扣除 警衛和建材----

x.train <- data.matrix(train[,c(2,4,5,6,7,8,9,10)])

y.train <- train[,1]

x.test <- data.matrix(test[,c(2,4,5,6,7,8,9,10)])

y.test <- test[,1]

xgb\_train = xgb.DMatrix(data = x.train, label = y.train)

xgb\_test = xgb.DMatrix(data = x.test, label = y.test)

bst <- xgboost(data = xgb\_train, max.depth = 6, nrounds = 150,

objective = "reg:squarederror",eval\_metric="rmse")

#bst

pred\_y = predict(bst, x.test)

importance\_matrix <- xgb.importance(colnames(x.train), model = bst)

xgb.plot.importance(importance\_matrix, rel\_to\_first = TRUE, xlab = "Relative importance")

windows()

mse = mean((y.test - pred\_y)^2)

mse

MAPE <- mean(abs((y.test - pred\_y)/y.test))

MAPE

####爬蟲 抓距離資料#####

## package 匯入

check.packages <- function(pkg) {

# STEP 1 檢查機器裡有沒有我現在要用的套件

new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]

if (length(new.pkg)) {

#if>0執行

# 安裝

install.packages(new.pkg, dependencies = TRUE)

}

## STEP 2 我要用的套件 讀近來

sapply(pkg, require, character.only = TRUE)

}

# 我要用的

pkg <- c(

"data.table", "dplyr", "jiebaR", "textTinyR", "remoji", "colortools",

"showtext", "wordcloud", "tm", "tmcn", "Rwordseg", "stringr", "tidytext",

"ggplot2", "word2vec", "bigstep",'stringr',"rvest","RSelenium","xml2"

)

check.packages(pkg)

my\_data <- read.csv("目標資料集.csv")

convinience <- c(rep(0,992)) %>% as.data.frame()

for (i in 1:nrow(my\_data)) {

convinience[i,1]<-paste0(my\_data[i,2],"便利商店")

}

paste0(my\_data[i,2],"便利商店")

location <- c(rep(0,992)) %>% as.data.frame()

for (i in 1:992) {

location[i,1] <- paste0('https://www.google.com/maps/dir/',my\_data[i,4],"/",convinience[i,1])

}

location[2,1]

####進入到爬網階段####----

####動態網頁:jquery,Ajex####

library(RSelenium)

remDr=rsDriver(browser = c("firefox"),port=sample(1000:8000,1))

driver=remDr[["client"]]

driver$open()

# 瀏覽 Google 首頁 test

#driver$navigate("https://www.google.com.tw/")

convifar<- c(rep(0,992)) %>% as.data.frame()

for (i in 1:4) {

driver$navigate(location[i,1])

webpage <- read\_html(driver$getPageSource()[[1]][1])

box <- webpage%>% html\_nodes(css ='#section-directions-trip-0 .fontBodyMedium') %>% html\_text()

convifar[i,1] <- paste0(trimws(box[1]))

Sys.sleep(0.5)

}

#my\_data <- cbind(my\_data,convifar)

#names(my\_data)[36] <- c("convifar")

#my\_data <- my\_data[,-1]

#write.csv(my\_data,"目標資料集2.csv",fileEncoding = "big-5")

my\_data[35]my\_data[1,35]

for (i in 1:992) {

a[i,1] <- nchar(my\_data[i,35])

}

a<- c(rep(0,992)) %>% as.data.frame()

distinct(a)

my\_data <- cbind(my\_data,1:992)

names(my\_data)[36] <- c("num")

adjust <- my\_data %>% arrange(desc(my\_data[35])) %>% select(num)

adjust <- adjust[1:68,1]

View(my\_data %>% arrange(desc(my\_data[35])))

#box <- webpage%>% html\_nodes(css ='#section-directions-trip-0 .fontBodyMedium') %>% html\_text()

#convifar[5,1] <- paste0(trimws(box[1]))

#section-directions-trip-0 .XdKEzd

##.ivN21e div

#section-directions-trip-0 .fontBodyMedium div

####繪製地圖####

library(dplyr)

library(sf)

newcity.map <-

st\_read("C:/Users/Huang/Desktop/研究/bigdata/mapdata202209220431/TOWN\_MOI\_1100415.shp") %>% filter(COUNTYNAME == "新北市")

#print(newcity.map)

dim(newcity.map)

names(newcity.map)

plot(newcity.map[1])

plot(st\_geometry(newcity.map))

newcity.map$TOWNNAME

my.newcity.map <- newcity.map[c("TOWNNAME", "geometry")]

my.newcity.map$TOWNNAME <- as.character(newcity.map$TOWNNAME)

head(my.newcity.map)

y <-

mydata10 %>% select(The.villages.and.towns.urban.district, total.price.NTD.1) %>% group\_by(The.villages.and.towns.urban.district) %>% summarise(Pricemedian=median(total.price.NTD.1))

my.taiwan.map.data <- left\_join(my.newcity.map,y,

by = c("TOWNNAME" = "The.villages.and.towns.urban.district"))

library(ggplot2)

ggplot(data = my.taiwan.map.data) +

geom\_sf(aes(fill = Pricemedian))+

scale\_fill\_distiller(palette = "Spectral")+

ggtitle("新北市房價")+

theme(plot.title = element\_text(hjust = 1))+

labs(caption = "備註:單位為每坪價格 灰色為無此資料")

####相關係數熱圖####

mydata <- mtcars[, c(1,3,4,5,6,7)]

head(mydata)

cormat <- round(cor(mydata),2)

library(reshape2)

melted\_cormat <- melt(cormat)

head(melted\_cormat)

melted\_cormat <- melt(cormat)

head(melted\_cormat)

# Get lower triangle of the correlation matrix

get\_lower\_tri<-function(cormat){

cormat[upper.tri(cormat)] <- NA

return(cormat)

}

# Get upper triangle of the correlation matrix

get\_upper\_tri <- function(cormat){

cormat[lower.tri(cormat)]<- NA

return(cormat)

}

upper\_tri <- get\_upper\_tri(cormat)

upper\_tri

# Melt the correlation matrix

library(reshape2)

melted\_cormat <- melt(upper\_tri, na.rm = TRUE)

# Heatmap

library(ggplot2)

ggplot(data = melted\_cormat, aes(Var2, Var1, fill = value))+

geom\_tile(color = "white")+

scale\_fill\_gradient2(low = "darkgreen", high = "darkred", mid = "white",

midpoint = 0, limit = c(-1,1), space = "Lab",

name="Pearson\nCorrelation") +

theme\_minimal()+

theme(axis.text.x = element\_text(angle = 45, vjust = 1,

size = 12, hjust = 1))+

coord\_fixed()

data <- read.csv("目標資料集10.csv")

names(data)

mydata <- data[, c(1,5,6,7,8,9,10)]

head(mydata)

cormat <- round(cor(mydata),2)

library(reshape2)

melted\_cormat <- melt(cormat)

head(melted\_cormat)

melted\_cormat <- melt(cormat)

head(melted\_cormat)

upper\_tri <- get\_upper\_tri(cormat)

upper\_tri

# Melt the correlation matrix

library(reshape2)

melted\_cormat <- melt(upper\_tri, na.rm = TRUE)

# Heatmap

library(ggplot2)

ggplot(data = melted\_cormat, aes(Var2, Var1, fill = value))+

geom\_tile(color = "white")+

geom\_text(aes(Var2, Var1, label = value), color = "black", size = 4)+

scale\_fill\_gradient2(low = "darkgreen", high = "darkred", mid = "white",

midpoint = 0, limit = c(-1,1), space = "Lab",

name="Pearson\nCorrelation") +

theme\_minimal()+

theme(axis.text.x = element\_text(angle = 45, vjust = 1,

size = 12, hjust = 1))+

coord\_fixed()

####保安和警衛熱圖和箱型圖####

mydata10 <- read.csv("目標資料集10.csv")

windows()

ggplot(mydata10, aes(x = mydata10[, 1])) +

geom\_boxplot(

fill = "slateblue",

alpha = 0.2,

size = 0.1,

outlier.shape = 19,

outlier.size = 2,outlier.color = "red"

)+

labs(x = '每坪價格',

y = '',

title = '房價的盒方圖') +

labs(caption = "註記:為已去除遺失值後960筆")

adjdata <-mydata10 %>% arrange(mydata10[,1])

adjdata <-adjdata[1:957,]

ggplot(adjdata, aes(x = adjdata[, 1])) +

geom\_boxplot(

fill = "slateblue",

alpha = 0.2,

size = 0.1,

outlier.shape = 19,

outlier.size = 2,outlier.color = "red"

)+

labs(x = '每坪價格',

y = '',

title = '房價的盒方圖')

summary(mydata10)

distinct(adjdata [11])

set.seed(3347)

names(adjdata)

adjdata1 <- adjdata %>% group\_by(materials,guard) %>% summarise(price=mean(total.price.NTD.1)) %>%as.data.frame()

adjdata2 <- dcast(adjdata1,materials~guard)

row.names(adjdata2) <- adjdata2$materials

adjdata2 <- adjdata2[,-1]

windows()

class(adjdata2)

adjdata2 <- as.matrix(adjdata2)

# Dummy data

x <- adjdata1[1]

y <- adjdata1[2]

for (i in 1:8) {

if(adjdata1[i,2]==1){

adjdata1[i,4]="有保安"

}

if(adjdata1[i,2]==0){

adjdata1[i,4]="沒保安"

}

}

for (i in 1:8) {

if(adjdata1[i,1]==1){

adjdata1[i,5] <- "SRC"

}

if(adjdata1[i,1]==2){

adjdata1[i,5] <- "SC"

}

if(adjdata1[i,1]==3){

adjdata1[i,5] <- "RC"

}

if(adjdata1[i,1]==4){

adjdata1[i,5] <- "磚造"

}

}

data <- cbind(X=x, Y=y)

data$Z <- adjdata1[,3]

# Color Brewer palette

library(viridis)

library(hrbrthemes)

adjdata1$V5 <- factor(adjdata1$V5, ordered = TRUE, levels = c("SRC", "SC", "RC", "磚造"))

# Give extreme colors:

ggplot(adjdata1, aes(adjdata1$V5, adjdata1$V4 ,fill= adjdata1$price)) +

geom\_tile() +

scale\_fill\_gradient(low="white", high="blue") +

theme\_ipsum()+ggtitle("保全與建築類別對房價的影響")

library(dplyr)

library(ggplot2)

data <- read.csv("C:/Users/Huang/Desktop/資料探勘/報告/目標資料集10.csv",fileEncoding = "big-5")

names(data)

distinct(data[11])

x=data.frame()

for (i in 1:957) {

if(data[i,11]==1){

x[i,1]="SC"

x[i,2]=18440/7020

}

if(data[i,11]==2){

x[i,1]="SRC"

x[i,2]=15050/7020

}

if(data[i,11]==3){

x[i,1]="RC"

x[i,2]=11180/7020

}

if(data[i,11]==4){

x[i,1]="Brick"

x[i,2]=7020/7020

}

}

for (i in 1:957) {

if(data[i,12]==1){

x[i,3]="有警衛"

}

if(data[i,12]==0){

x[i,3]="沒警衛"

}

}

####heatmap----

library(reshape2)

names(x) <- c("materials","weights","guard")

allmatrix <- cbind(x,data[,1])

allmatrix <- allmatrix[,-2]

names(allmatrix)[3] <- "y"

allmatrix <- allmatrix %>% select(guard,materials,y) %>% group\_by(guard,materials)

allmatrix.dcast <- dcast(allmatrix,materials~guard)

table(allmatrix$guard,allmatrix$materials)

single <- cbind(x,data[,1])

single <- single[ ,-2]#run兩次

names(single)[2] <- "y"

single <- data.frame(single ,hei=rep(1,957)) %>% group\_by(materials) %>% summarise(mean=mean(y))

single <- data.frame(single ,hei=rep(1,4))

single$materials <- factor(single$materials, levels = c("SC","SRC","RC","Brick"))

##single單一材料變數

ggplot(single, aes(x = materials, y = hei , fill = mean)) +

geom\_tile(colour = "white", size = 0.25) + # 繪製熱圖

scale\_y\_discrete(expand = c(0, 0)) + # 移除多餘空白

scale\_x\_discrete(expand = c(0, 0)) + # 移除多餘空白

coord\_fixed() + # 設定 X 與 Y 軸等比例

scale\_colour\_brewer(palette = "Greens") + # 設定色盤

theme(

legend.text = element\_text(face = "bold"),

# 說明文字用粗體

axis.ticks = element\_line(size = 0.5),

# 座標軸上的刻度寬度

plot.background = element\_blank(),

# 移除背景

panel.border = element\_blank(),

# 移除邊框

axis.text.x = element\_text(

angle = 90,

vjust = 0.5,

hjust = 1

) # X 軸文字轉向

)+xlab("建築材料的種類") + ylab("") + labs(title = "建築材料對每坪房價影響的熱圖", caption = "備註:建材價格比 SC:SRC:RC:Brick=2.63:2.14:1.59:1")

goal <- cbind(x,data[,1]);names(goal)[4] <-c("y")

goal <- goal %>% group\_by(materials,guard) %>% summarise(mean=mean(y))

#a <- goal[,-3]

goal.melt <- dcast(goal,materials~guard)

rownames(goal.melt) <- goal.melt$materials

#goal.melt <- goal.melt[,-1] %>% as.matrix()

library(RColorBrewer)

##交叉分析----

goal$materials <-

factor(goal$materials, levels = c("SC", "SRC", "RC", "Brick"))

windows()

text <- c(2, 41, 686, 191, 30, 1, 15, 1)

goaltext <- cbind(goal, text)

names(goaltext)[4] <- "text"

ggplot(goaltext, aes(x = materials , y = guard , fill = mean)) +

geom\_tile(colour = "white", size = 0.25) + # 繪製熱圖

scale\_y\_discrete(expand = c(0, 0)) + # 移除多餘空白

scale\_x\_discrete(expand = c(0, 0)) + # 移除多餘空白

coord\_fixed() + # 設定 X 與 Y 軸等比例

scale\_fill\_gradientn(colours = brewer.pal(7, "YlGnBu")) + # 設定色盤

theme(

legend.text = element\_text(face = "bold"),

# 說明文字用粗體

axis.ticks = element\_line(size = 0.5),

# 座標軸上的刻度寬度

plot.background = element\_blank(),

# 移除背景

panel.border = element\_blank(),

# 移除邊框

axis.text.x = element\_text(

angle = 90,

vjust = 0.5,

hjust = 1

) # X 軸文字轉向

) + xlab("建築材料的種類") + ylab("") + labs(title = "建築材料和有無警衛對每坪房價影響的熱圖", caption = "圖中的數字為個數 \n 備註:建材價格比 SC:SRC:RC:Brick=2.63:2.14:1.59:1")+geom\_text(label = goaltext$text)

####模型####

test20 <- fread("test20.csv",header = T)

train80 <- fread("train80.csv",header = T)

train <- train80 %>% select(total.price.NTD.1,hospital.far,day,busfar,materials,all.police.station.far,convifar,all\_post\_office)

rc=data.frame()

for (i in 1:nrow(train)) {

if (train[i,5]==3){

rc[i,1]=1

}

else{

rc[i,1]=0

}

}

train <- cbind(train,rc)

test=data.frame()

for (i in 1:nrow(test20)) {

if (test20[i,11]==3){

test[i,1]=1

}

else{

test[i,1]=0

}

}

test=cbind(test20,test)

names(test)[13] <- "rc"

names(train)[9] <- "rc"

boxcox <- function(hospital\_far,day,busfar,rc,all\_police\_station\_far,convifar,all\_post\_office){

lambday <- 862.073938-0.028524\*hospital\_far-0.007728\*day+0.019445\*busfar-92.915140\*rc-0.006234\*all\_police\_station\_far+0.007280\*convifar--0.002763\*all\_post\_office

fity <- (lambday\*0.5+1)^2

return(fity)

}

names(test20.x )

test20.x <- test[,c(10,4,5,13,8,6,7)] %>% data.frame()

fittedy <- data.frame()

summary(train80)#ymean118763

for (loop in 1:1) {

SSE <- 0

SST <- 0

MAPE <- 0

for (i in 1:nrow(test20)) {

fittedy[i, 1] <- boxcox(test20.x[i, 1],

test20.x[i, 2],

test20.x[i, 3],

test20.x[i, 4],

test20.x[i, 5],

test20.x[i, 6],

test20.x[i, 7])

SSE <- SSE + (test20[i, 1] - fittedy[i, 1]) ^ 2

SST <- SST + (test20[i, 1] - 118763) ^ 2

MAPE <- MAPE+(abs(test20[i, 1]-fittedy[i, 1])/test20[i, 1])/nrow(test20)

}

Rsquared <- 1 - SSE / SST %>% as.numeric()

MSE <- SSE /(nrow(test20)-7) %>% as.numeric()

MAPE <- MAPE %>% as.numeric()

print(c(Rsquared=Rsquared,MSE=MSE,MAPE=MAPE))

}

##單一只有跑距離變數

distance <- function(hospital\_far,busfar,all\_police\_station\_far,convifar,all\_post\_office,schoolfar){

lambday <- 720.646018-0.026699\*hospital\_far+0.023433\*busfar-0.009658\*all\_police\_station\_far+0.004920\*convifar-0.002707\*all\_post\_office-0.000421\*schoolfar

fity <- (lambday\*0.5+1)^2

return(fity)

}

test.x <- test20[,c(4,5,6,7,8,9,10)] %>% data.frame()

names(test.x)

fittedy <- data.frame()

for (loop in 1:1) {

SSE <- 0

SST <- 0

MAPE <- 0

for (i in 1:nrow(test20)) {

fittedy[i, 1] <- distance(test.x[i, 7],

test.x[i, 2],

test.x[i, 5],

test.x[i, 3],

test.x[i, 4],

test.x[i, 6])

SSE <- SSE + (test20[i, 1] - fittedy[i, 1]) ^ 2

SST <- SST + (test20[i, 1] - 118763) ^ 2

MAPE <- MAPE+(abs(test20[i, 1]-fittedy[i, 1])/test20[i, 1])/nrow(test20)

}

Rsquared <- 1 - SSE / SST %>% as.numeric()

MSE <- SSE /(nrow(test20)-7) %>% as.numeric()

MAPE <- MAPE %>% as.numeric()

print(c(Rsquared=Rsquared,MSE=MSE,MAPE=MAPE))

}

####Xgboosting####

require(xgboost)

train <- read.csv("train80orange.csv",fileEncoding = "utf-8")

test<- read.csv("test20orange.csv",fileEncoding = "utf-8")

####六個變數----

x.train <- data.matrix(train[,c(5,6,7,8,9,10)])

y.train <- train[,1]

x.test <- data.matrix(test[,c(5,6,7,8,9,10)])

y.test <- test[,1]

xgb\_train = xgb.DMatrix(data = x.train, label = y.train)

xgb\_test = xgb.DMatrix(data = x.test, label = y.test)

bst <- xgboost(data = xgb\_train, max.depth = 6, nrounds = 150,

objective = "reg:squarederror",eval\_metric="rmse")

#bst

pred\_y = predict(bst, x.test)

importance\_matrix <- xgb.importance(colnames(x.train), model = bst)

xgb.plot.importance(importance\_matrix, rel\_to\_first = TRUE, xlab = "Relative importance")

windows()

mse = mean((y.test - pred\_y)^2)

mse

MAPE <- mean(abs((y.test - pred\_y)/y.test))

MAPE

####行政+日期 all----

x.train <- data.matrix(train[,c(2,4,5,6,7,8,9,10,11,12)])

y.train <- train[,1]

x.test <- data.matrix(test[,c(2,4,5,6,7,8,9,10,11,12)])

y.test <- test[,1]

xgb\_train = xgb.DMatrix(data = x.train, label = y.train)

xgb\_test = xgb.DMatrix(data = x.test, label = y.test)

bst <- xgboost(data = xgb\_train, max.depth = 6, nrounds = 150,

objective = "reg:squarederror",eval\_metric="rmse")

#bst

pred\_y = predict(bst, x.test)

importance\_matrix <- xgb.importance(colnames(x.train), model = bst)

xgb.plot.importance(importance\_matrix, rel\_to\_first = TRUE, xlab = "Relative importance")

windows()

mse = mean((y.test - pred\_y)^2)

mse

MAPE <- mean(abs((y.test - pred\_y)/y.test))

MAPE

####扣除 警衛和建材----

x.train <- data.matrix(train[,c(2,4,5,6,7,8,9,10)])

y.train <- train[,1]

x.test <- data.matrix(test[,c(2,4,5,6,7,8,9,10)])

y.test <- test[,1]

xgb\_train = xgb.DMatrix(data = x.train, label = y.train)

xgb\_test = xgb.DMatrix(data = x.test, label = y.test)

bst <- xgboost(data = xgb\_train, max.depth = 6, nrounds = 150,

objective = "reg:squarederror",eval\_metric="rmse")

#bst

pred\_y = predict(bst, x.test)

importance\_matrix <- xgb.importance(colnames(x.train), model = bst)

xgb.plot.importance(importance\_matrix, rel\_to\_first = TRUE, xlab = "Relative importance")

windows()

mse = mean((y.test - pred\_y)^2)

mse

MAPE <- mean(abs((y.test - pred\_y)/y.test))

MAPE

#隨機森林10個變數的程式碼:

rm = list(ls = all(T))

setwd("D:/資料探勘百靈杯")

library(randomForest)

library(dplyr)

set.seed(1)

data = read.csv("目標資料集9.csv")

train\_data = read.csv("train80.csv")

test\_data = read.csv("test20.csv")

names(data)

names(test\_data)

names(train\_data)

train\_data = subset(train\_data,select = -c(land.sector.position.building.sector.house.number.plate))

test\_data = subset(test\_data,select = -c(land.sector.position.building.sector.house.number.plate))

set.seed(1)

new\_rf1 <- randomForest(total.price.NTD.1~.,data = train\_data,mtry = 10,importance = TRUE)

yhat <- predict(new\_rf1,newdata = test\_data)

mean((yhat-test\_data$total.price.NTD.1)^2) #mse

mean(abs((test\_data$total.price.NTD.1-yhat)/test\_data$total.price.NTD.1)) #MAPE

#隨機森林8個變數的程式碼:

rm = list(ls = all(T))

setwd("D:/資料探勘百靈杯")

library(randomForest)

library(dplyr)

set.seed(1)

data = read.csv("目標資料集9.csv")

train\_data = read.csv("train80.csv")

test\_data = read.csv("test20.csv")

names(data)

train\_data = subset(train\_data, select = -c(land.sector.position.building.sector.house.number.plate,materials,guard))

test\_data = subset(test\_data, select = -c(land.sector.position.building.sector.house.number.plate,materials,guard))

names(test\_data)

names(train\_data)

set.seed(1)

new\_rf1 <- randomForest(total.price.NTD.1~.,data = train\_data,mtry = 8,importance = TRUE)

yhat <- predict(new\_rf1,newdata = test\_data)

mean((yhat-test\_data$total.price.NTD.1)^2) #mse

mean(abs((test\_data$total.price.NTD.1-yhat)/test\_data$total.price.NTD.1)) #MAPE

#隨機森林6個程式碼:

rm = list(ls = all(T))

setwd("D:/資料探勘百靈杯")

library(randomForest)

library(dplyr)

set.seed(1)

data = read.csv("目標資料集9.csv")

train\_data = read.csv("train80.csv")

test\_data = read.csv("test20.csv")

names(data)

train\_data = subset(train\_data, select = -c(land.sector.position.building.sector.house.number.plate,materials,guard))

test\_data = subset(test\_data, select = -c(land.sector.position.building.sector.house.number.plate,materials,guard))

names(test\_data)

names(train\_data)

set.seed(1)

new\_rf1 <- randomForest(total.price.NTD.1~.,data = train\_data,mtry = 8,importance = TRUE)

yhat <- predict(new\_rf1,newdata = test\_data)

mean((yhat-test\_data$total.price.NTD.1)^2) #mse

mean(abs((test\_data$total.price.NTD.1-yhat)/test\_data$total.price.NTD.1)) #MAPE

#蒐集郵局的程式碼

getwd()

setwd("D:\\資料探勘百靈杯")

rm(list = ls(all=T))

## package 匯入

check.packages <- function(pkg) {

# STEP 1 檢查機器裡有沒有我現在要用的套件

new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]

if (length(new.pkg)) {

#if>0執行

# 安裝

install.packages(new.pkg, dependencies = TRUE)

}

## STEP 2 我要用的套件 讀近來

sapply(pkg, require, character.only = TRUE)

}

# 我要用的

pkg <- c(

"data.table", "dplyr", "jiebaR", "textTinyR", "remoji", "colortools",

"showtext", "wordcloud", "tm", "tmcn", "Rwordseg", "stringr", "tidytext",

"ggplot2", "word2vec", "bigstep",'stringr',"rvest","RSelenium","xml2"

)

check.packages(pkg)

my\_data <- read.csv("目標資料集3.csv")

post\_office<- c(rep(0,992)) %>% as.data.frame()

for (i in 1:nrow(my\_data)) {

post\_office[i,1]<-paste0(my\_data[i,1],"郵局")

}

location <- c(rep(0,992)) %>% as.data.frame()

for (i in 1:992) {

location[i,1] <- paste0('https://www.google.com/maps/dir/',my\_data[i,3],"/",post\_office[i,1])

}

####進入到爬網階段####----

####動態網頁:jquery,Ajex####

library(RSelenium)

remDr=rsDriver(browser = c("firefox"),port=sample(1000:8000,1))

driver=remDr[["client"]]

driver$open()

# 瀏覽 Google 首頁 test

#driver$navigate("https://www.google.com.tw/")

post\_office\_far<- c(rep(0,992)) %>% as.data.frame()

for (i in 1:992) {

driver$navigate(location[i,1])

webpage <- read\_html(driver$getPageSource()[[1]][1])

box <- webpage%>% html\_nodes(css ='#section-directions-trip-0 .fontBodyMedium') %>% html\_text()

post\_office\_far[i,1] <- paste0(trimws(box[1]))

Sys.sleep(0.5)

}

#-----------------------

#利用PYTHON的字串功能看有幾筆錯誤

##post\_office\_far

write.csv(post\_office\_far,file="D:\\資料探勘百靈杯\\post\_office.csv",row.names = FALSE)

## import

#改完後傳回R並存入error\_value

error\_value <- read.csv("error.csv",head = F)

error\_value = error\_value[,1]+1

#把有錯的值改空白

for (i in error\_value){

#cat(post\_office\_far[,1][i])

post\_office\_far[,1][i] = " "

}

post\_office\_far

error\_value

#-----------------------------

#看有錯的location

for (i in error\_value){

cat(location[,1][i],"\n")

}

#-------------------------------------------

#發現大多如下，所以分成幾個去存入新的距離

#section-directions-trip-1 .fontBodyMedium

#section-directions-trip-1 .fontBodyMedium div

#section-directions-trip-0 .fontBodyMedium div

#section-directions-trip-0 .fontBodyMedium div

remDr=rsDriver(browser = c("firefox"),port=sample(1000:8000,1))

driver=remDr[["client"]]

driver$open()

# 瀏覽 Google 首頁 test

#driver$navigate("https://www.google.com.tw/")

post\_office\_far\_1<- c(rep(0,992)) %>% as.data.frame()

for (i in error\_value) {

driver$navigate(location[i,1])

webpage <- read\_html(driver$getPageSource()[[1]][1])

box <- webpage%>% html\_nodes(css ='#section-directions-trip-1 .fontBodyMedium') %>% html\_text()

post\_office\_far\_1[i,1] <- paste0(trimws(box[1]))

Sys.sleep(0.5)

}

#----------------

post\_office\_far\_2<- c(rep(0,992)) %>% as.data.frame()

for (i in error\_value) {

driver$navigate(location[i,1])

webpage <- read\_html(driver$getPageSource()[[1]][1])

box <- webpage%>% html\_nodes(css ='#section-directions-trip-1 .fontBodyMedium div') %>% html\_text()

post\_office\_far\_2[i,1] <- paste0(trimws(box[1]))

Sys.sleep(0.5)

}

#----------------

post\_office\_far\_3<- c(rep(0,992)) %>% as.data.frame()

for (i in error\_value) {

driver$navigate(location[i,1])

webpage <- read\_html(driver$getPageSource()[[1]][1])

box <- webpage%>% html\_nodes(css ='#section-directions-trip-0 .fontBodyMedium div') %>% html\_text()

post\_office\_far\_3[i,1] <- paste0(trimws(box[1]))

Sys.sleep(0.5)

}

#------------------------------------------

#看結果

#有用的很多個

new\_number\_1 <- c()

new\_value\_1 <- c()

for (i in 1:992){

if(post\_office\_far\_1[,1][i] > 0 && post\_office\_far\_1[,1][i] != "NA"){

new\_number\_1 <- append(new\_number\_1,i)

new\_value\_1 <- append(new\_value\_1,post\_office\_far\_1[,1][i])

}

}

new\_number\_1

new\_value\_1

#---------------------------------

#沒用

new\_number\_2 <- c()

new\_value\_2 <- c()

for (i in 1:992){

if(post\_office\_far\_2[,1][i] > 0 && post\_office\_far\_2[,1][i] != "NA"){

new\_number\_2 <- append(new\_number\_2,i)

new\_value\_2 <- append(new\_value\_2,post\_office\_far\_2[,1][i])

}

}

new\_number\_2

new\_value\_2

#---------------------------------

#有用的只有2個

new\_number\_3 <- c()

new\_value\_3 <- c()

for (i in 1:992){

if(post\_office\_far\_1[,1][i] > 0 && post\_office\_far\_3[,1][i] != "NA"){

new\_number\_3 <- append(new\_number\_3,i)

new\_value\_3 <- append(new\_value\_3,post\_office\_far\_3[,1][i])

}

}

new\_number\_3

new\_value\_3

#-------------------------------------

#將距離的值存入原本的data.frame

k = 1

for (i in new\_number\_1){

post\_office\_far[i,1] = new\_value\_1[k]

k = k + 1

}

#---------------------------------

#把已輸入進去的位置刪掉

new\_error\_value <- error\_value

for (i in 1:length(error\_value)){

for (j in 1:length(new\_number\_1)){

if (error\_value[i] == new\_number\_1[j]){

new\_error\_value[i] = NA

}

}

}

new\_error\_value <- na.omit(new\_error\_value)

#計算剩下幾個error

length(new\_error\_value)

#----------------

#剩下不多，手動輸入

for (i in new\_error\_value){

cat("第",i,"個",location[,1][i],'\n\n')

}

post\_office\_far[7,1] <- "2.3 公里"

post\_office\_far[84,1] <- "900 公尺"

post\_office\_far[87,1] <- "1.1 公里"

post\_office\_far[100,1] <- "900 公尺"

post\_office\_far[102,1] <- "1.0 公里"

post\_office\_far[106,1] <- "900 公尺"

post\_office\_far[109,1] <- "1.0 公里"

post\_office\_far[122,1] <- "900 公尺"

post\_office\_far[147,1] <- "1.0 公里"

post\_office\_far[161,1] <- "850 公尺"

post\_office\_far[164,1] <- "4.8 公里"

post\_office\_far[284,1] <- "1.0 公里"

post\_office\_far[292,1] <- "850 公尺"

post\_office\_far[298,1] <- "350 公尺"

post\_office\_far[314,1] <- "3.8 公里"

post\_office\_far[315,1] <- "9.7 公里"

post\_office\_far[328,1] <- "850 公尺"

post\_office\_far[341,1] <- "650 公尺"

post\_office\_far[377,1] <- "1.0 公里"

post\_office\_far[398,1] <- "1.7 公里"

post\_office\_far[427,1] <- "3.3 公里"

post\_office\_far[464,1] <- "700 公尺"

post\_office\_far[523,1] <- "1.0 公里"

post\_office\_far[559,1] <- "3.9 公里"

post\_office\_far[564,1] <- "700 公尺"

post\_office\_far[567,1] <- "750 公尺"

post\_office\_far[569,1] <- "1.2 公里"

post\_office\_far[573,1] <- "11.7 公里"

post\_office\_far[621,1] <- "1.5 公里"

post\_office\_far[646,1] <- "750 公尺"

post\_office\_far[702,1] <- "1.4 公里"

post\_office\_far[708,1] <- "800 公尺"

post\_office\_far[839,1] <- "1.3 公里"

post\_office\_far[848,1] <- "650 公尺"

post\_office\_far[850,1] <- "1.0 公里"

post\_office\_far[989,1] <- "1.4 公里"

post\_office\_far[990,1] <- "1.4 公里"

post\_office\_far

write.csv(post\_office\_far,file="D:\\資料探勘百靈杯\\all\_post\_office.csv",row.names = FALSE)

#---------------------------寫入後發現有誤的

#233

#402

#429

#721

#740

#771

#806

#855

#856

location[,1][771]

location[,1][806]

location[,1][855]

location[,1][856]

#-------------------------手動改正

post\_office\_far[233,1] <- "2.3 公里"

post\_office\_far[402,1] <- "850 公尺"

post\_office\_far[429,1] <- "850 公尺"

post\_office\_far[721,1] <- "700 公尺"

post\_office\_far[740,1] <- "850 公尺"

post\_office\_far[771,1] <- "900 公尺"

post\_office\_far[806,1] <- "1.0 公里"

post\_office\_far[855,1] <- "900 公尺"

post\_office\_far[856,1] <- "900 公尺"

#交易年月日-建築完成年月

getwd()

setwd("D:\\資料探勘百靈杯")

rm(list = ls(all = T))

library(dplyr)

library(lubridate)

Mydata <- read.csv("目標資料集.csv")

names(Mydata)

#將建造時間存入construction time

for (i in 1:length(Mydata)){

if (colnames(Mydata)[i] == "construction.to.complete.the.years")

output = i

}

construction\_time <- Mydata[output]

#將交易時間存入construction time

for (i in 1:length(Mydata)){

if (colnames(Mydata)[i] == "transaction.year.month.and.day")

output = i

}

transaction\_time <- Mydata[output]

#將空缺值排除(建造時間有空缺值)

na\_sum <- sum(is.na(construction\_time))

#將沒有空缺值的建造時間存入new\_construction\_time

new\_construction\_time <-na.omit(construction\_time) %>% as.data.frame()

#把沒有空缺值編號存入後面交易時間可以把建造時間沒有空缺值的值與交易時間相減

new\_rownames <- rownames(new\_construction\_time) %>% as.numeric()

#因為建造時間有空缺值導致不能直接讓交易時間-建造時間，所有將可以相減的提出來，也就是交易時間提出來並存入new\_transaction\_time

new\_transaction\_time <- c()

k = 0

for (i in new\_rownames){

new\_transaction\_time <- append(new\_transaction\_time,((transaction\_time[,1])[i]))

}

new\_transaction\_time <- new\_transaction\_time %>% as.data.frame()

#用來看是否有對到

nrow(new\_transaction\_time)

nrow(new\_construction\_time)

#因為很多直接使用ymd會出錯，先將他們同時加19110000，可以變成2022之類的，再用ymd處理

new\_construction\_time <- new\_construction\_time[,1]+19110000

new\_transaction\_time <- new\_transaction\_time[,1]+19110000

#ymd

#將結果處存在output

output <- c()

for (i in 1:length(new\_transaction\_time)){

output <- append(output,ymd(new\_transaction\_time[i]) - ymd(new\_construction\_time[i]))

}

output %>% as.data.frame()

output

#負的是因為建造時間筆交易時間晚，然後第992筆可能有問題

#建造日期有空缺值的是哪幾行並存入error\_value

error\_value <- c()

for (i in 1:nrow(construction\_time)){

if ((is.na(construction\_time))[i] == T){

error\_value <- append(error\_value,i)

}

}

error\_value

#-------------

target\_number <- c()

for (i in error\_value){

target\_number <- append(target\_number,Mydata[,4][i])

}

target\_number

#主要建材計算

getwd()

setwd("D:\\資料探勘百靈杯")

rm(list = ls(all=T))

library(readxl)

library(dplyr)

Mydata <- read.csv("目標資料集.csv")

class(Mydata)

#用csv讀空格會變成點

for (i in 1:length(Mydata)){

if (colnames(Mydata)[i] == "main.building.materials")

output = i

}

#所有總類

category <- distinct(Mydata[output]) %>% as.data.frame()

category <- category[,1]

category

#計算各個類別的個數

number <- c()

for (i in 1:length(category)){

number <- append(number, nrow(Mydata %>% filter(Mydata[output] == category[i])))

}

#將總類與個數存入

main\_building\_materials\_data\_frame <- data.frame(x = category,y = number)

#觀看

View(main\_building\_materials\_data\_frame)

#總數

sum(number)