

第四組資料分析與R語言期末報告

影響大安區**YouBike2.0**缺車風險之可能因素

107208058 / 經濟四 / 廖珮嫻

107208068 / 經濟四 / 江馥羽

108305096 / 企管三 / 江彤恩

107306064 / 資管四 / 嚴怡詠

目錄

目錄	1
研究動機	2
介紹	3
資料來源與變數定義及說明	4
分析方法	6
結論	11
參考文獻	12
附錄	13

研究動機

政治大學位於一個交通相對難以到達的地區，除了住宿外，選擇通勤的人大多是使用捷運搭乘到動物園站，再轉公車或腳踏車才能抵達學校；或是較晚才回到文山區，出捷運站後公車已過了末班車，僅能使用腳踏車做為代步工具，否則就要徒步行走近三十分鐘才能抵達政大。組內幾乎都是通勤生，偶較早出門、天氣又風光明媚之時，有時會想騎著腳踏車享受一段愜意的通勤路，然而每每出捷運站、走到公共自行車站時，美好的幻想總被現實潑一桶冷水：車柱全部是空的！沒有任何一輛自行車可供租用！不知是再稍早的一些人和自己打著相同的算盤，所以早就都被借走了，還是控制中心沒有預計到該站會有大量的借車需求，而未即時補充足夠的腳踏車，趁興而來，卻敗興而歸，即使天氣再好，上空卻好似盤旋著一朵灰色烏雲，讓原本愉悅的心情也沈重起來。再想到高中時，自己常常會睡到最後一刻才跳起床準備出門，雖然捷運站只有十分鐘左右的路程，如果能搶到自行車便可節省下五分鐘，五分鐘看似微不足道，但對於快要遲到的人來說，這一分一秒都是場戰鬥，是必須斤斤計較的。無奈人算不如天算，通勤的交通交鋒時段，腳踏車可謂一輛難求，時常只能眼巴巴望著空無一車的Ubike站點，然後繼續趕路。

「不能夠遲到」無論是對於學生亦或上班族，都是十分重要的，所以我們認為能夠確保站點隨時有車，無「缺車風險」是非常需要在意的一項指標，倘若不會再發生想借車時沒車可借的窘境，說不定能不再為搶不到腳踏車而預留多一些的步行時間，將鬧鐘再往後延遲一點也說不定。

起初在到處搜尋有趣的資料來做期末報告時，便打開了「政府資料開放平台」大海撈針，看看有沒有好玩的、有意思的資料拿來使用，比如看到了動物園的動植物統計，對那驚人的數字訝異，或是看到整個文山區的無障礙餐廳數量之稀少，最後翻到了統計雙北公共自行車，也就是Ubike的借還資料，便想起前面種種我們曾經歷過的窘境，最終決定拿這筆資料進行分析，製作期末報告。無奈於資料的龐大，平常主要用於文書處理的電腦跑不動如此大量的資料分析，且適逢政大的Ubike站點要從1.0更新至2.0，所以我們決定使用目前數量相對較少的2.0站點資料來做分析。目前台北市共設有1094個2.0站點，原先打算針對信義區，但其商業活動佔比較高，尤其是晚上是許多人會前去消費娛樂的繁華地帶，可能影響的變數較多不易探討，故改以有住宅、亦有商業活動的大安區，其站點共有144個，做為我們的主要探討對象。Ubike 2.0在大安區的缺車風險，便是此份報告欲嘗試探討的主題。

介紹

我們希望透過真實的借還車資料，去分類、分析哪些特定站點比較熱門，常常會出現大量租車需求、大量還車需求，以及從中推測可能的因素。

不同時間段是第一項我們認為可能影響的因素。推測平日的上下班通勤時段易有相對較多的Ubike需求，因為捷運站間的距離較遠時，或許會以Ubike做為中間移動的代步工具；故推測通勤時段可能會出現大量公共自行車的需求。

再者是平日與假日的差別，前述的通勤時段主要針對平日而言，假日因沒有明顯上下班的時間段，故若該日為假日，通勤時段並不會出現大量需求。假日時，公共自行車則可能成為休閒娛樂的一環，或許是騎在河堤邊享受沿途的風光，或是做為抵達部分易達性相對較低的觀光景點用的代步工具。故推測假日需要大量公共自行車可能不是受時間段所影響，而是特定站點。

最後是周邊設施，附近是否有重要設施，比如是否有捷運站、公園、學校，這類的設施定義相對明確，資料也較容易取得，所以暫且不考慮其他雖具重要性，卻難以定義的設施。我們認為若站點附近有捷運站、公園、學校，或許會讓該站點的Ubike使用率增加，故這也是應考慮的因素。

「是否有足夠的車輛供大家借還」是相對模糊，僅以文字說明的概念，故我們嘗試以數據的方式定義，導入「缺車風險」的概念，希望能夠以明確的標準去界定該站的車輛是否充足。

資料來源與變數定義及說明

從政府公開資料平台上面下載下來的檔案，是以月份為單位的檔案，我們使用了最新的檔案，2021年11月單月的資料，光是該檔，就已經有一百萬筆的借還紀錄，其中含有的資訊有：租借的日期時間(rent_time)、租借的站點名稱(rent_station)、歸還的日期時間(return_time)、歸還的站點名稱(return_station)、總租借時長(rent)，其中租借的日期時間是以一小時為單位，紀錄借出與歸還各在哪個時間區段裡。這是關於Ubike 2.0的借還紀錄，以及每筆資料有包含的資訊。

礙於設備問題，我們聚焦於「大安區」和「Ubike 2.0站點」進行討論，故我們需要先從一百萬筆資料中篩選出所有Ubike 2.0站點，以及「借出在大安區」、「還車在大安區」、「借還都在大安區」這三類資料。

首先，從維基百科上找到Ubike所有站點，其中紅色標示為Ubike 2.0的站，將該表格複製下來，先在excel做簡單的資料整理，將錯誤格式校正，且避免直接丟進R時資料過於龐大而跑不動。變數包含租賃站名稱(name)、設站位置(location)、鑽孔數量(quant)、啟用日期(date)以及周邊設施(nearby)，並透過excel內建的公式COUNTIF，若nearby欄位有關鍵字「捷運」、「公園」、「小學」、「國民中學」、「高級中學」、「大學」，若有就給1，若無就給0。最後再使用公式IF((OR))將小學、國中、高中、大學四項數量加總，設成新欄位「學校」。自此我們就將先前推測可能影響Ubike使用的因素填進去，並設定好變數。

整理好Ubike 2.0站點資料後，接下來還要處理從政府公開資料平台上面下載下來的實際租借資料檔案。因為此報告嘗試討論的是大安區，故將所有非大安區的借還資料都刪除，再將剩餘資料分成三類：「只有借出在大安區」，名稱為rent_daan；「只有還車在大安區」，名稱為return_daan；以及「借還車都在大安區」，名稱為both_daan。透過excel內建的公式代勞完成了所有分類，雖然因為總共有一百萬筆資料，所以跑完花了不少時間，但除去我們不需要的資料之後，終於可以把不再那麼大筆的資料放進去R語言裡面。

```
only_rent <- read_excel('202111_YouBike2.xlsx','rent_daan')
only_return <- read_excel('202111_YouBike2.xlsx','return_daan')
both <- read_excel('202111_YouBike2.xlsx','both_daan')
```

將excel讀進R以後，將資料分別命名為：「只有借車在大安區」，名稱為only_rent；「只有還車在大安區」，名稱為only_return；「借還車都在大安區」，名稱為both。

接著我們使用「dplyr」及「tidyr」的套件，將原始excel資料中的NA值及問號刪除，接著將only_rent及both的資料合併，取得在大安區的所有車站之借車數據，亦將only_return及both的資料合併，取得在大安區的所有車站之還車數據，然後依序按照站牌、時間做出排序，並計算所有大安區站牌在各個時段的借還車數量，我們也將原本資料中的時間區間從一小時改為兩小時，

此外，我們的缺車風險計算方式為：在同一時段內，若借出車輛 - 還車車輛 < 該站點所有車柱*0.6，就將該地點該時段列為有缺車風險。

為了分析方便，我們最後將借、還車的數據以時間、地點取聯集的方式合併成一張表格，如下圖所示：

station	date	time	rent_value	holidays	return_value	quan	mrt	park	school	elementary	junior	senior	university	lack
瑞公公園	2021-11-02	00:00	5	0	1	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-02	02:00	3	0	0	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-02	04:00	3	0	0	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-02	06:00	3	0	1	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-02	08:00	4	0	6	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-02	10:00	3	0	3	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-02	12:00	5	0	8	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-02	14:00	10	0	9	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-02	16:00	15	0	12	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-02	18:00	13	0	15	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-02	20:00	11	0	10	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-02	22:00	4	0	3	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-03	00:00	3	0	1	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-03	02:00	2	0	0	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-03	04:00	1	0	0	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-03	06:00	2	0	2	20	1	1	0	0	0	0	0	0
瑞公公園	2021-11-03	08:00	10	0	6	20	1	1	0	0	0	0	0	0

圖 1

分析方法

1. 模型假設

為了分析缺車風險的影響因素，在此我們將「該日為假日或非假日」與「該站點的周邊設施」列為可能影響借還車的因素，並將周邊設施以捷運、公園及學校的有無來分開討論。由於應變數與自變數皆為二元變數，意即我們可將資料轉換為0或1，因此我們採用羅吉斯迴歸的分析方法，以下為本文所建構的羅吉斯迴歸模型：

$$\frac{p(x)}{1-p(x)} = Y_{it} = \beta_0 + \beta_1 X_{1t} + \beta_2 X_{2t} + \beta_3 X_{3t} + \beta_4 X_{4t}$$

式中，應變數 Y_{it} 代表某停車站缺車風險的有無， $Y_{it} = 0$ 表示該站無缺車風險， $Y_{it} = 1$ 則表示有缺車風險。自變數 X_{1t} 代表該日是否為假日，若 $X_{1t} = 0$ 為非假日，若 $X_{1t} = 1$ 則為假日。而 X_{2t} 代表該站周邊是否有捷運， X_{3t} 代表公園的有無， X_{4t} 代表學校的有無。若周邊無該設施，以 $X_t = 0$ 表示；若有，則以 $X_t = 1$ 表示。 β_0 為模型的常數項；($i=1,2,3,4$) 為模型的係數項； t 則代表2021年11月某日某時間段。

2. 資料訓練及驗證

我們以5-fold cross validation的方法將我們的資料分為test_data及train_data，並使用五次迴圈的方式將每次train_data得出的模型對test_data進行預測，最後將進行五次預測後的test_data加總，得出預測值與實際值的異同，並計算出準確度。

我們亦對原始資料進行羅吉斯迴歸分析， $H_0: \beta_i = 0$ ， $H_1: \beta_i \neq 0$ ，在 $\alpha = 0.05$ 的信心水準下，假日、學校、及捷運站之變數皆小於0.05，拒絕虛無假設，說明有足夠證據顯示大安區站牌在假日或附近有學校、捷運站的地方對於缺車風險有顯著影響。

Coefficients:					
	Estimate	Std. Error	z value	Pr(> z)	
(Intercept)	-3.81970	0.07090	-53.877	< 2e-16	***
holidays	-0.55994	0.10892	-5.141	2.74e-07	***
mrt	0.52361	0.09492	5.516	3.46e-08	***
park	0.04494	0.11133	0.404	0.686463	
school	-0.46030	0.12176	-3.780	0.000157	***

圖2

由於我們無法明確證實公園一變數對缺車風險是否有顯著影響，故我們將公園之變數從模型中剔除，估計出最終的模型如下：

$$Y_{it} = \beta_0 + \beta_1 X_{1t} + \beta_2 X_{2t} + \beta_4 X_{4t}$$

$$= -3.81 - 0.56 X_{1t} + 0.52 X_{2t} - 0.47 X_{4t}$$

Coefficients:				
	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-3.80758	0.06400	-59.495	< 2e-16 ***
holidays	-0.56036	0.10892	-5.145	2.68e-07 ***
mrt	0.51967	0.09441	5.505	3.70e-08 ***
school	-0.47072	0.11891	-3.959	7.54e-05 ***

圖3

3. 實證結果

- 使用羅吉斯迴歸進行機器學習的預測結果

	0	1
0	26695	538

圖4

我們將cut-off設為0.5，意即若預測出有缺車風險的機率大於0.5就設定為1(有缺車風險)，若小於0.5則設定為0(沒有缺車風險)，結果預測有缺車風險之機率皆趨近於0。

- 假日與平日的**YouBike**使用量差異
由於收集到的平日天數比假日天數來的多，平日的使用總量也像對來的多許多。因此，我們將平日、假日的租借總次數除以天數，得到平日、假日個時段的平均租借次數，發現平日的使用量比假日多，其中在8:00~10:00 & 18:00~20:00 平日的平均租借次數比假日來的多，而在10:00~16:00的時段則是假日的平均租借次數比平日更多。平日的使用高峰時段為8:00~10:00 & 18:00~20:00，而假日的使用高峰時段則為12:00~14:00 & 16:00~18:00。
- 各時段的**YouBike**使用量
我們發現，8:00~10:00 & 18:00~20:00 為租借youbike2.0的尖峰時段，或許是因為學生族和上班族的上下課/班的緣故，導致這兩個時段的租借次數高於其他時段。下班時間過後，隨著時間越晚，YouBike 的使用量越少，在凌晨2:00~6:00 的使用量最低。

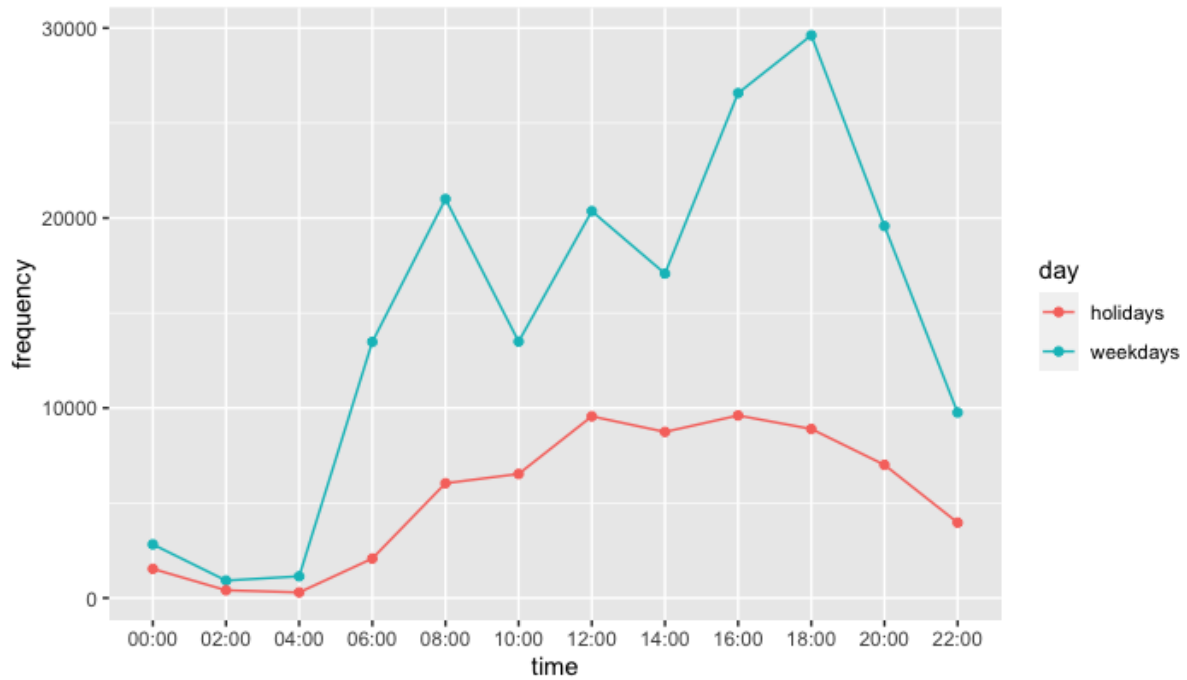


圖5 平日及假日各時段總租借次數

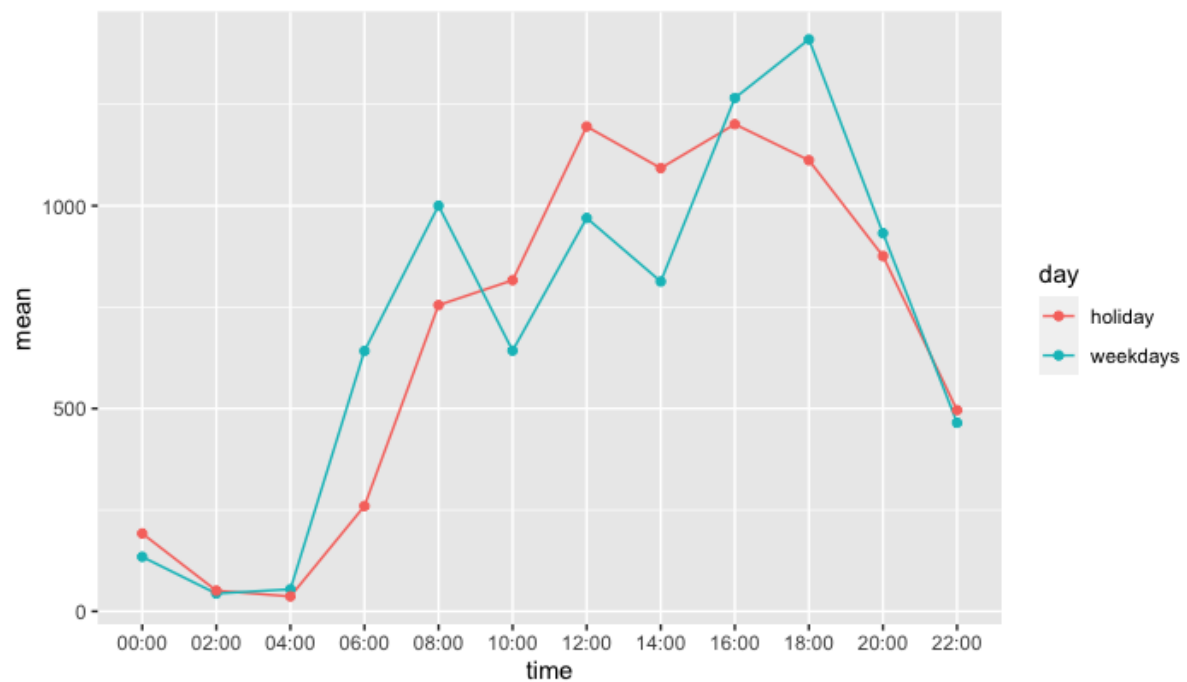


圖6 平日及假日各時段平均租借次

- 最常缺車的站點

由下圖排名可得知，缺車風險較高之車站集中於捷運大安站及科技大樓站一區，該地區位於台北市的精華地區，附近有捷運、大安森林公園及多間高中與大學，因此平日上班族及學生非常多，假日也有不少人會到大安森林公園，導致該區的缺車風險較高。

▲	station	sum(lack)
1	捷運大安站(4號出口)	24
2	捷運科技大樓站	23
3	敦化南路二段103巷口	22
4	捷運公館站(2號出口)	21
5	敦化基隆路口	19
6	捷運大安森林公園站(5號出口)	19
7	羅斯福路三段311號前	18
8	嘉興公園	16
9	銘傳國小側門	16
10	復興南路二段273號前	15
11	建國南路一段279巷口	13
12	羅斯福路三段245號前	13
13	仁愛安和路口	13
14	大安高工	11
15	臺大醫學院附設癌醫中心	10
16	信義建國路口	10
17	和平復興路口西北側	9
18	捷運大安站(2號出口)	9
19	臨江街夜市(通化街101巷口)	9
20	新生南路三段66號前	9

圖7 前20大YouBike2.0缺車車站之排名

- 最常缺車的時段

我們將出現缺車風險時段的次數加總，發現8:00~10:00的時段最容易發生缺車的狀況，其次則是16:00~18:00。其中再分別查看平日和假日的缺車時段，發現平日的缺車時段和上述相同，而假日則是20:00~22:00的缺車風險最高，其次為中午12:00~14:00。

time	total
08:00	117
16:00	95
06:00	74
18:00	71
20:00	67
10:00	34
12:00	32
14:00	28
22:00	17
00:00	3
02:00	1

圖8 各時段出現缺車風險的次數

time	total
08:00	102
16:00	81
06:00	74
18:00	65
20:00	45
10:00	24
12:00	14
14:00	12
22:00	11
00:00	3
02:00	1

圖9 平日各時段出現缺車風險的次數

time	total
20:00	22
12:00	18
14:00	16
08:00	15
16:00	14
10:00	10
18:00	6
22:00	6

圖10 假日各時段出現缺車風險的次數

結論

透過羅吉斯迴歸分析，我們得知「該日為平日或假日」與「周邊是否有捷運或學校」將對缺車風險造成顯著影響。而平日的缺車情形主要發生於早上 8:00 ~ 10:00 及 下午16:00 ~ 18:00，也就是上下班的尖峰時段；假日則為晚上 20:00 ~ 22:00 及中午 12:00 ~ 14:00。此外，捷運大安站及科技大樓站一區是最常發生缺車情形的區域，該區人潮眾多，鄰近捷運、學校與公園，同排名次之的敦化南路站、公館站等，皆屬於大安區之精華地段。

然而，我們使用羅吉斯迴歸進行機器學習的預測結果為皆無缺車風險，此結果與我們一開始的預設相左，上網查詢一些資料後，發現我們的原始資料可能有「資料不平衡」的情況。在我們的資料中，有缺車風險的欄位僅佔沒有缺車風險的欄位的2.01%，因為資料比例的嚴重失衡，無論我們設定的變數對於缺車風險有無顯著影響，預測出有缺車風險的機率皆會趨近於0。而我們可能可以透過under sampling, over sampling, SMOTE, GAN等方法來改善資料不平衡的情況，使我們的預測更準確。

參考文獻

1. 政府資料開放平臺
<https://data.gov.tw>
2. 臺北市公共自行車2.0租借紀錄
<https://data.gov.tw/dataset/150635>
3. youbike2.0臺北市公共自行車即時資訊
<https://data.gov.tw/dataset/137993>
4. 臺北市公共自行車租賃系統站點列表
<https://zh.wikipedia.org/wiki/臺北市公共自行車租賃系統站點列表>
- 5.【QA】如何處理不平衡資料(imbalanced data)?
<https://reurl.cc/b2jVYy>

附錄

```
getwd()
library(readxl)
library(dplyr)
library(tidyr)
library(lubridate)
setwd("C:/Users/USER/Dropbox/我的電腦 (LAPTOP-5PL8Q2NT)/Downloads")

#載入EXCEL資料, 分別為「只有借車在大安」、「只有還車在大安」及「借還車都在大安」
only_rent <- read_excel('202111_YouBike2(2).xlsx','rent_daan')
only_return <- read_excel('202111_YouBike2(2).xlsx','return_daan')
both <- read_excel('202111_YouBike2(2).xlsx','both_daan')

#去除?及NA值、分別取得大安區各站的借車及還車數據、按照站名及時間依序排列
arr_rent <- rbind(only_rent,both)%>% arrange(rent_station, rent_time)
arr_rent$rent_station <- gsub("[?]", "溜", arr_rent$rent_station)
arr_rent[is.na(arr_rent)] <- 0
arr_return <- rbind(only_return,both)%>% arrange(return_station, return_time)
arr_return$return_station <- gsub("[?]", "溜", arr_return$return_station)
arr_return$holidays[is.na(arr_return$holidays)] <- 0

#將時間資料合併為2小時為一區間, 並計算大安區各站各時段的借車及還車總數
rent <- arr_rent %>% select(rent_time, rent_station, holidays) %>% mutate(rent_time =
as.POSIXct(rent_time), value = rep(1, nrow(arr_rent))) %>% group_by(rent_station,holidays,
rent_time_2hr = floor_date(rent_time, "2 hour")) %>% summarize(rent_value = sum(value))
return <- arr_return %>% select(return_time, return_station, holidays) %>% mutate(return_time =
as.POSIXct(return_time), value = rep(1, nrow(arr_return))) %>% group_by(return_station,holidays,
return_time_2hr = floor_date(return_time, "2 hour")) %>% summarize(return_value = sum(value))

#為了將rent 和return的資料作成同張table,
#抓出rent和return資料的站名和時間, 做出所有站名和時間的總表,
#並刪除重複值
rent_part <- rent %>% select(rent_station, rent_time_2hr,holidays)
colnames(rent_part) <- c("station", "time", "holidays")
rent_part
return_part <- return %>% select(return_station, return_time_2hr, holidays)
colnames(return_part) <- c("station", "time", "holidays")
return_part
full_test <- rbind(rent_part, return_part)
full_station_time <- full_test[!(duplicated(full_test[c("station","time")]) |
duplicated(full_test[c("station","time")])), ] %>% arrange(station,time)
#full_station_time 是rent 和return裡面出現的所有站名和時間的table

#利用總表join rent和return的資料
full_join_rent <- left_join(full_station_time, rent, by = c("station" = "rent_station", "time" =
"rent_time_2hr"))
```

```

full_join_all <- left_join(full_join_rent, return, by = c("station" = "return_station", "time" =
"return_time_2hr"))

#加上是否鄰近捷運、學校等資料並將NA改為0
variables <- read_excel('第四組_資料_廖珮嫻、江馥羽、嚴怡詠、江彤恩.xlsx', 'return_') %>%
select(station, quan, mrt, park, school, elementary, junior, senior, university)
full <- left_join(full_join_all, variables, by = "station")
full[is.na(full)] <- 0
full <- subset(full, select = -holidays.y)
full <- full %>% select(-holidays.x)
#將time的日期和時段分開
full <- separate(full, time, c("date", "time"), " ")
full$time <- gsub('.{3}$', "", full$time)

#平日/假日租借總次數
library(ggplot2)
holidays <- full[which(full$holidays==1), ]
weekdays <- full[which(full$holidays==0), ]
holidays_freq <- holidays %>% group_by(time) %>% summarise(sum(rent_value))
day <- c(rep("weekdays", 12), rep("holidays", 12))
frequency <- cbind(rbind(weekdays %>% group_by(time) %>% summarise(sum(rent_value))),
holidays_freq), day)
colnames(frequency) <- c("time", "frequency", "day")
ggplot(frequency, aes(x = time, y = frequency, group = day)) + geom_line(aes(color=day)) +
geom_point(aes(color=day))

#平日/假日平均租借次數
mean <- rep(0,24)
for ( i in 1:24){
  if (day[i] == "weekdays") {
    mean[i] <- frequency$frequency[i]/21
  }else {
    mean[i] <-frequency$frequency[i]/8
  }
}
fre_mean <- cbind(frequency, mean)
ggplot(fre_mean, aes(x = time, y = mean, group = day)) + geom_line(aes(color=day)) +
geom_point(aes(color=day))

#設定各車站和時段的缺車風險
lack <- filter(full, rent_value - return_value > quan*0.6)
data <- cbind(full, lack)

#缺車風險之車站排名
lack_station <- filter(data, lack=="1")
lack_station_freq <- data %>% group_by(station) %>% summarise(sum(lack))
order <- lack_station_freq[order(lack_station_freq[,2], decreasing = TRUE), ]
order

```

```

#各時段出現缺車風險的次數
lack_ <- lack %>% group_by(time) %>% summarise(total = n())
#假日各時段出現缺車風險的次數
lack_hol <- lack %>% filter(holidays == 1) %>% group_by(time) %>% summarise(total = n())
#平日各時段出現缺車風險的次數
lack_weekday <- lack %>% filter(holidays == 1) %>% group_by(time) %>% summarise(total = n())

#羅吉斯迴歸分析
#估計模型
mod1 <- glm( lack ~ holidays + mrt + park + school, data= data, family="binomial")
summary(mod1)

#資料訓練及驗證
set.seed(123)
data_random <- data[sample(1:n, size=n, replace = F), ]
list1 <- 1:5446
list2 <- 5447:10892
list3 <- 10893:16338
list4 <- 16339:21784
list5 <- 21785:27233
list6 <- list(list1, list2, list3, list4, list5)
logit_mod <- list()
logit_prob <- list()
logit_pred <- list()
cf.mat <- list()

#5-fold cross validation
for(i in 1:5) {
  data_sample <- as.numeric(list6[[i]])
  test_data <- data_random[data_sample, ]
  train_data <- data_random[-data_sample, ]
  logit_mod[[i]] <- glm( lack ~ holidays + mrt + school, data= train_data,
family=binomial(link="logit"))
  logit_prob[[i]] <- predict(logit_mod[[i]], newdata = test_data, type="response")
  logit_pred[[i]] <- ifelse(logit_prob[[i]] > 0.5, 1, 0)
  cf.mat[[i]] <- table( logit_pred[[i]], data$lack[data_sample])
}
cm_sum <- cf.mat[[1]]+cf.mat[[2]]+cf.mat[[3]]+cf.mat[[4]]+cf.mat[[5]]
cm_sum

#最終估計模型
mod2 <- glm( lack ~ holidays + mrt + school, data= data, family="binomial")
summary(mod2)

```