- 1. 清洗 Accident.csv 的資料
- (1) A1/A2 資料合併成 A2

將 A1/A2 資料合併成一類(A2), A3 另成一類分析。

(2) 資料不列入

不將編號、天候名稱、光線名稱、路面狀況名稱、當事者性別名稱、車種名稱、保護裝備名稱及飲酒情形名稱列入資料分析內容,因為我認為編號對於分析結果應沒有幫助,只是普通的數字,其他天候、光線、路面狀況等名稱,因為他們有各自對應的代碼,所以不將這些說明列入分析中。

(3) 資料內容改動

因 accident 內的天候代碼與天候名稱對不上,例如天候代碼 8 對應的名稱就有「晴」與「暴雨」,無法做惡劣天氣的分類,因此我將每個代碼都只有對應一個天氣名稱。

天氣代碼	1	2	4	6	7	8
天氣名稱	暴雨	強風	霧或煙	雨	陰	晴

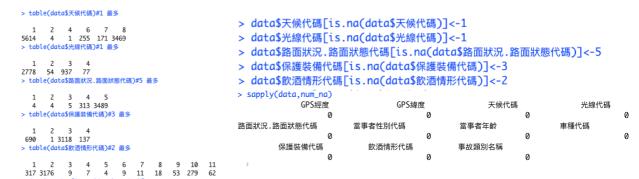
(4) 遺失資料處理

● 一開始匯入遺失的資料筆數:



● 將空白的資料以比數最多的資料取代:

找出比數最多者、將資料取代後,確認沒有遺漏的資料。



● 車種代碼為 factor,所以使用 dummy variable 分析:

```
車種代碼 = factor(data\$車種代碼)#車種代碼變dummy dummies = model.matrix(~車種代碼+0) dummies<-dummies[,-1] data1<-cbind(data,dummies)
```

- 2. 使用 SVM 方法及 Logistics 方法進行 10 次預測,預測目標為「事故類別」。
- 進行 10 次的 Logistics 方法與 SVM 方法預測事故類別: 在 logistics 中,我認為事故類別裡 A3 的發生次數會較多,所以機率應該較 A1,A2 高,所以當機率大於 0.5 則分類為 A3。

```
for(i in 1:10){
n<-0.1*nrow(data1)
index_d<-sample(1:nrow(data1),n)
train_d<-data1[-index_d,]
test_d<-data1[index_d,]
test_d<-data1[index_d,]
#logit

model_d<-glm(事故類別名稱)
p1<-predict(model_d,test_d,type = "response")
p1_result<-ifelse(p1>0.5,"A3","A2")
x<-mean(p1_result=etst_d5事故類別名稱)
accuracy.logit[i]<-x
#svm
model.svm<-svm(事故類別名稱-,,data=train_d)
resultd2<-predict(model.svm,test_d)
true_value.svm<-test_d5事故類別名稱
#table(true_value.svm,result02)
compare02<-
ifelse(result02==true_value.svm,1,0)
y<-sum(compare02)/length(compare02)
accuracy.svm[i]<-y
```

■ Logistics 和 SVM 的準確度:

這兩個方法預測事故種類的準確度都高達95%以上,在圖中基本上 logit 的準確度都比 SVM 高一些,而在第3,4,7,9 筆資料中, SVM 和 logit 的準確度一樣,所以在圖片中只有一個點。

> accuracy.logit

[1] 0.9730290 0.9636929 0.9512448 0.9678423 0.9678423 0.9616183 0.9564315 0.9564315 0.9636929 [10] 0.9688797

> accuracy.svm

[1] 0.9719917 0.9626556 0.9512448 0.9678423 0.9668050 0.9626556 0.9564315 0.9543568 0.9636929 [10] 0.9668050

