HW3

前言：

使用附檔的「Accident.csv」，透過各種資料處理的方式：包括(補值、刪值)、或是意義相同的 feature 擇其一，最後使用 k-means 進行分群。
此外，因為針對 categorical feature 如果採用 1,2,3…這種編碼方式，會產生有的類別距離較近、有的距離卻較遠這種不合理的結果，加上是以 K-means 來進行分群，會加劇這種偏差。因此本作業會採用 one-hot encoding 的方式，將類別型屬性處理成標準向量的那種形式(ex:1,0,0…0)。
然後將資料帶入程式碼前，先將 excel 的 feature 由中文改成英文

Feature 由中文轉成英文

’編號’：ID
’發生星期’：week
’ GPS 經度’：Longitude
’ GPS 緯度’：Latitude
’天候代碼’：Weather code
’天候名稱’：Weather name
’光線代碼’：Light code
’光線名稱’：Light name
’路面狀況-路面狀態代碼’：Road condition code
’路面狀況名稱’： Road condition name
’當事者性別代碼’：Gender code
’當事者性別名稱’：Gender name
’當事者年齡’：age
’車種代碼’：Vehicle type code
’車種名稱’：Vehicle type name
’保護裝備代碼’：Protection equipment code
’保護裝備名稱’：Protection equipment name
’飲酒情形代碼’：Drinking situation code
’飲酒情形名稱’：Drinking situation name
’事故類別名稱’：Accident category

程式碼：

```
setwd("C:/Users/Steven/Desktop/陽交109下/巨量資料分析/課程/單元5：相似度、鄰點、與聚類/範例程式與資料")
accs <- read.csv(file = "Accidents.csv", header=T, encoding='ANSI')
View(accs) # 開另一個視窗觀察csv檔的長相
str(accs) # 觀察此資料的結構以及各屬性的資料型態以及內容
class(accs) # 確認一下accs是不是dataframe
```

```r
# 將不重要的屬性、重複的屬性刪掉(像名稱跟代碼意義一樣就選擇把代碼去掉)
accs = accs[, c(6, 8, 10, 12, 13, 15, 17, 19, 20)]
# 觀察新資料的結構以及各屬性的資料型態以及內容
str(accs)
# 看一下剩餘特徵的資料分布，觀察有無空值、異常值、以及需要處理成數字的值
summary(accs)

# 把age的異常值直接改成NA
accs$age <- ifelse(accs$age>=0 , accs$age, NA)

# 從類別型屬性(共8個)下手，從中找到其空值的長相，有兩種分別是" "和""
unique(accs$Weather.name)
unique(accs$Light.name)
unique(accs$Road.condition.name)
unique(accs$Gender.name)
unique(accs$Vehicle.type.name)
unique(accs$Protection.equipment.name)
unique(accs$Drinking.situation.name)
unique(accs$Accident.category)

# 將" "和""帶換成空值
accs[accs == ""] <- NA
accs[accs == " "] <- NA
# 確認一下""和" "是否都轉成NA了
unique(accs$Weather.name)
unique(accs$Light.name)
unique(accs$Road.condition.name)
unique(accs$Gender.name)
unique(accs$Vehicle.type.name)
unique(accs$Protection.equipment.name)
unique(accs$Drinking.situation.name)
unique(accs$Accident.category)

# 算一下各個屬性共有幾筆NA
num_na <- function(x){sum(is.na(x))}
sapply(accs, num_na)
```

NA 太多，補值會產生太多偏差，所以選擇不補

```r
# 一共9643筆資料
# 但Drinking.situation.name、Protection.equipment.name、Light.name這三個特徵NA都過高，所以選擇刪除
accs = accs[, c(1, 3, 4, 5, 6, 9)]
# Vehicle.type.name、Gender.name、age因為空值很少但不好插補，選擇有NA的直接整筆刪除
accs <- accs[!is.na(accs$Vehicle.type.name),]
accs <- accs[!is.na(accs$Gender.name),]
accs <- accs[!is.na(accs$age),]
sapply(accs, num_na) # 只剩Weather.name、Road.condition.name有NA

# 接下來處理天氣狀況去插補路面的NA、或是一些明顯錯誤的值(比如暴雨地面不會乾燥)
accs$Road.condition.name <- ifelse(is.na(accs$Road.condition.name) &
                                   accs$Weather.name=='晴', '乾燥', accs$Road.condition.name)
accs$Road.condition.name <- ifelse(is.na(accs$Road.condition.name) &
                                   accs$Weather.name=='雨', '濕潤', accs$Road.condition.name)
accs$Road.condition.name <- ifelse(is.na(accs$Road.condition.name) &
                                   accs$Weather.name=='暴雨', '濕潤', accs$Road.condition.name)
accs$Road.condition.name <- ifelse(is.na(accs$Road.condition.name) &
                                   accs$Weather.name=='陰', '乾燥', accs$Road.condition.name)
accs$Road.condition.name <- ifelse(accs$Road.condition.name=='乾燥' &
                                   accs$Weather.name=='暴雨', '濕潤', accs$Road.condition.name)
accs$Road.condition.name <- ifelse(accs$Road.condition.name=='濕潤' &
                                   accs$Weather.name=='晴', '乾燥', accs$Road.condition.name)
# 剩下的NA直接整筆刪除
accs <- accs[!is.na(accs$Road.condition.name),]
accs <- accs[!is.na(accs$Weather.name),]
sapply(accs, num_na) # 發現都已補完
```

```r
# 再次確認一下剩餘類別屬性的值有哪些
unique(accs$Weather.name)
unique(accs$Road.condition.name)
unique(accs$Gender.name)
unique(accs$Vehicle.type.name)
unique(accs$Accident.category)

# 發現Vehicle.type.name值有點多
# 因此打算依據速度分成三種，人、慢車一組；機車一組;剩餘的一組
accs$Vehicle.type.name <- ifelse(accs$Vehicle.type.name=='人', '慢', accs$Vehicle.type.name)
accs$Vehicle.type.name <- ifelse(accs$Vehicle.type.name=='慢車', '慢', accs$Vehicle.type.name)
accs$Vehicle.type.name <- ifelse(accs$Vehicle.type.name=='機車', '中', accs$Vehicle.type.name)
accs$Vehicle.type.name <- ifelse(accs$Vehicle.type.name=='慢' |
                                 accs$Vehicle.type.name=='中', accs$Vehicle.type.name, '快')
unique(accs$Vehicle.type.name)

# 將類別型屬性轉成one-hot variable前要先將屬性轉成factor的型態(性別因為是二元所以不用)
accs$Weather.name <- as.factor(accs$Weather.name)
accs$Road.condition.name <- as.factor(accs$Road.condition.name)
accs$Vehicle.type.name <- as.factor(accs$Vehicle.type.name)
accs$Accident.category <- as.factor(accs$Accident.category)
# 確認是否轉成功
str(accs)
# 導入套件並做one-hot encoding
library(mltools)
library(data.table)
accs <- one_hot(as.data.table(accs))
View(accs)

# 最後處理gender這欄，男生:1 女生:0
accs$Gender.name <- ifelse(accs$Gender.name=='男', 1, 0)

# 資料正規化
accs_z <- sapply(accs, scale)
View(accs_z)
```

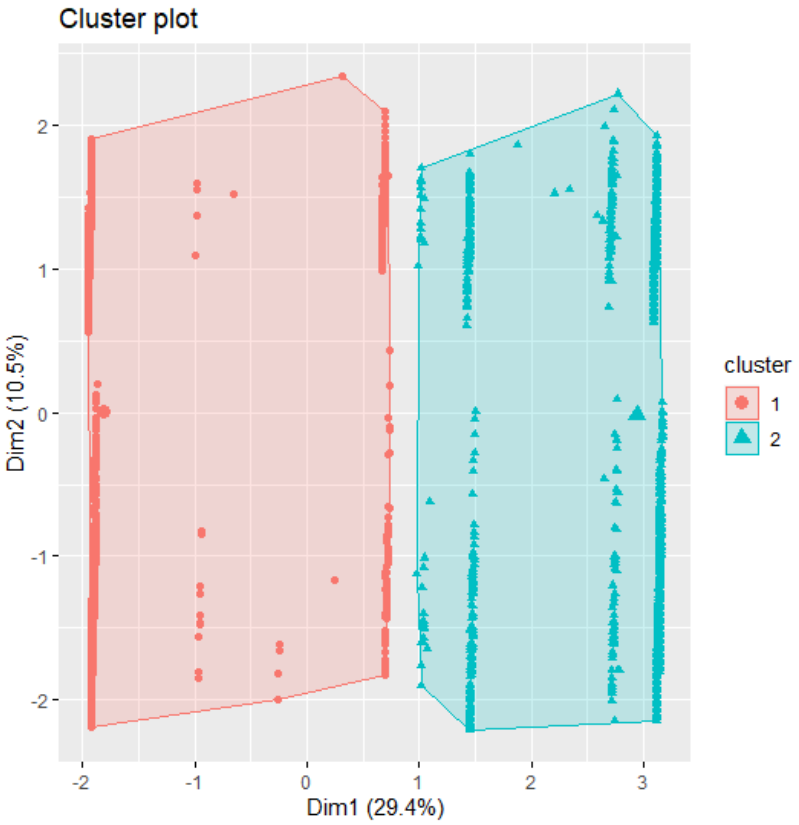
嘗試之後發現設定 k=2 的分群效果還可以

```r
# k-mean聚類分析，令k=2
kc <- kmeans(accs_z, iter.max=30, centers=2, nstart=10)
# 主成分分析
library(factoextra)
fviz_cluster(kc, geom="point", data=accs_z)
# 分群結果放入原本的資料集
accs$cluster <- kc$cluster

# 看一下accs的feature有哪些
names(accs)
# 查看各群的屬性值是否有差異
aggregate(data=accs, cbind(age, Gender.name, Weather.name_雨,
                           Weather.name_強風, Weather.name_陰,
                           Weather.name_暴雨, Vehicle.type.name_中,
                           Vehicle.type.name_快, Vehicle.type.name_慢,
                           Accident.category_A1, Accident.category_A2,
                           Accident.category_A3) ~ cluster, mean, na.rm=TRUE)
```

分群結果：

主成分分析結果



可以看出是存在某些屬性可以正確分兩群的

檢查不同群之間是否存在不同特性

cluster	Age	Gender.name	Weather_雨	Weather_強風
1	39.12712	0.6431253	0.0443411407	0
2	39.21640	0.6634671	0.0002908668	0.0008726003
cluster	Weather_陰	Weather_暴雨	Vehicle.type_中	Vehicle.type_快
1	0.0001787949	0.9554801	0.3797604	0.5692830
2	0.0456660849	0.0000000	0.3941245	0.5468296
cluster	Vehicle.type_慢	Accident.category_A1	Accident.category_A2	Accident.category_A3
1	0.05095655	0.0001787949	0.04183801	0.95798319
2	0.05904596	0.0029086678	0.91099476	0.08609657

發現塗色的這些欄位再兩群中差異較大，反而年齡性別這種的差異很小

結果與討論：

本次的資料集算是蠻有趣而且也是非常實用的資料集，尤其在台灣這種交通意外頻仍的地方，如果能夠正確分析意外發生的主要原因，那對未來立法者或是大眾來說都會有不小的助益。不過不同的是這次我們採用的是非監督式學習的方式，所以雖然能夠分群，但我們無法很明確的敘述每一群代表的意義。

我們一共分為兩群，由於前面提到，為了避免偏差，所以我們對類別型 feature 採取 one-hot encoding，所以會有雨、強風這些本來不存在的 feature 出現，觀察分群結果可以發現，相比其他 feature，雨、強風、暴雨等和 accident 的都是明顯的特徵(也就是在兩群中差異度大)。

而值的一提的是，這次資料處理過程中，並沒有把**保護裝備**、**飲酒情形**還有**光線**這些事前就覺得很重要的 feature 納入 K-means 作分析，原因是因為大家這幾欄的缺失值實在是太多，都超過原數據的一半，加上插補難度高，與其他欄關係不太高且缺失值太多導致用均值插補意義不大，權衡下覺得代表性可能會不足，所以決定捨棄這些 feature，若缺失值可以少一半，將其納入分析，可能可以得出在兩群中的差異性很大的結果，成為一個不錯的特徵。