Machine Learning HW 1 Report

組員：0516025張智閔、0516032邱繼聖、0516049吳柏劭、0516215林亮穎、0516220李元毓

1. **What environments the members are using?**

OS: Windows 10

Language: Python 3.6

Packeages:

(1) numpy 1.15.2

(2) Pandas 0.23.4

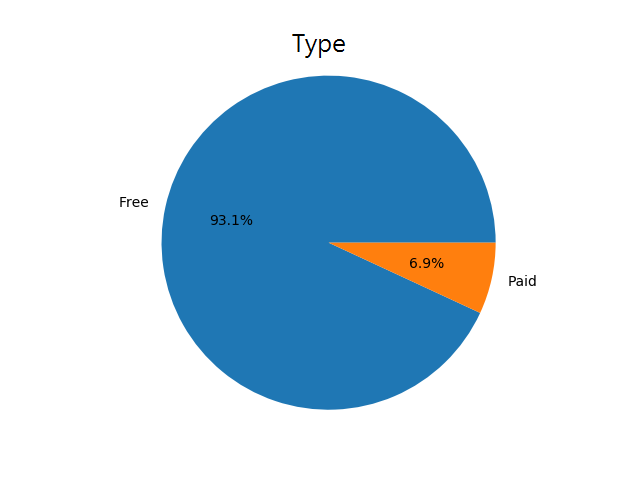
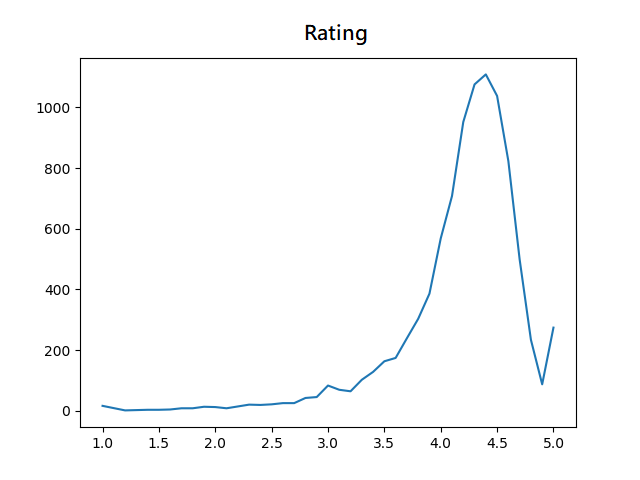
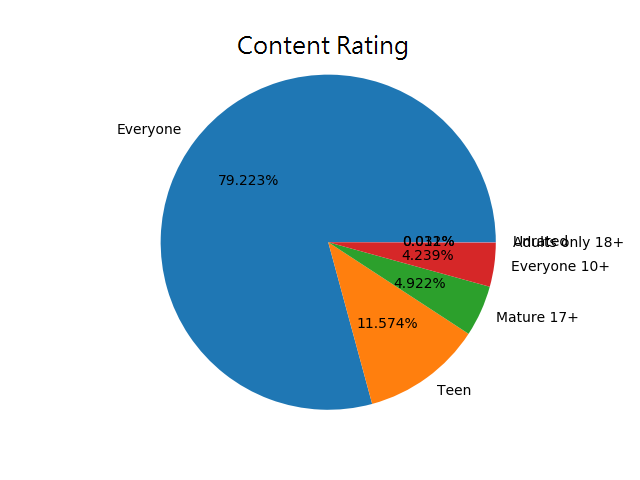
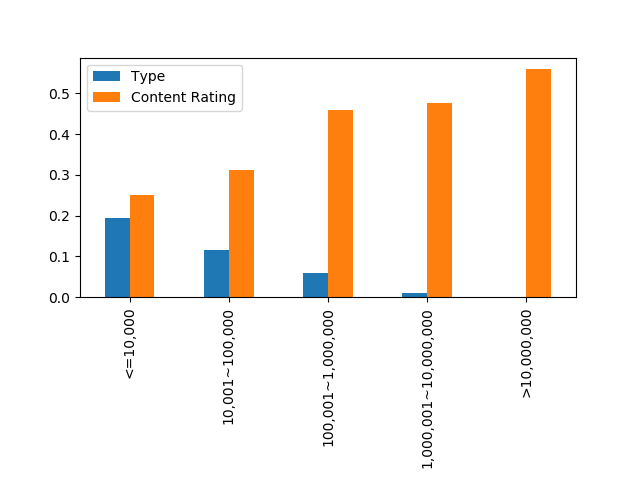
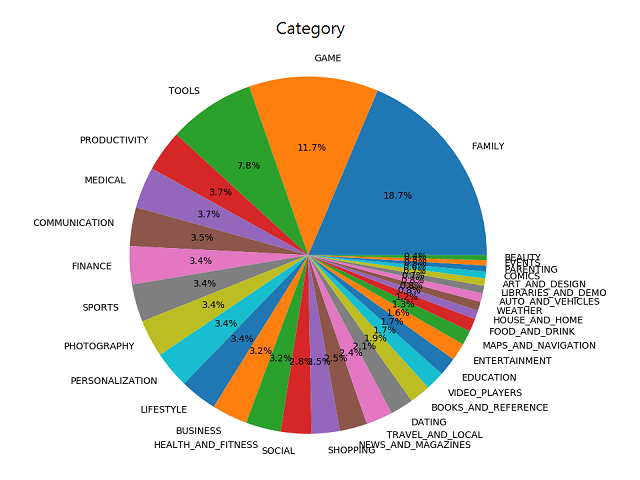
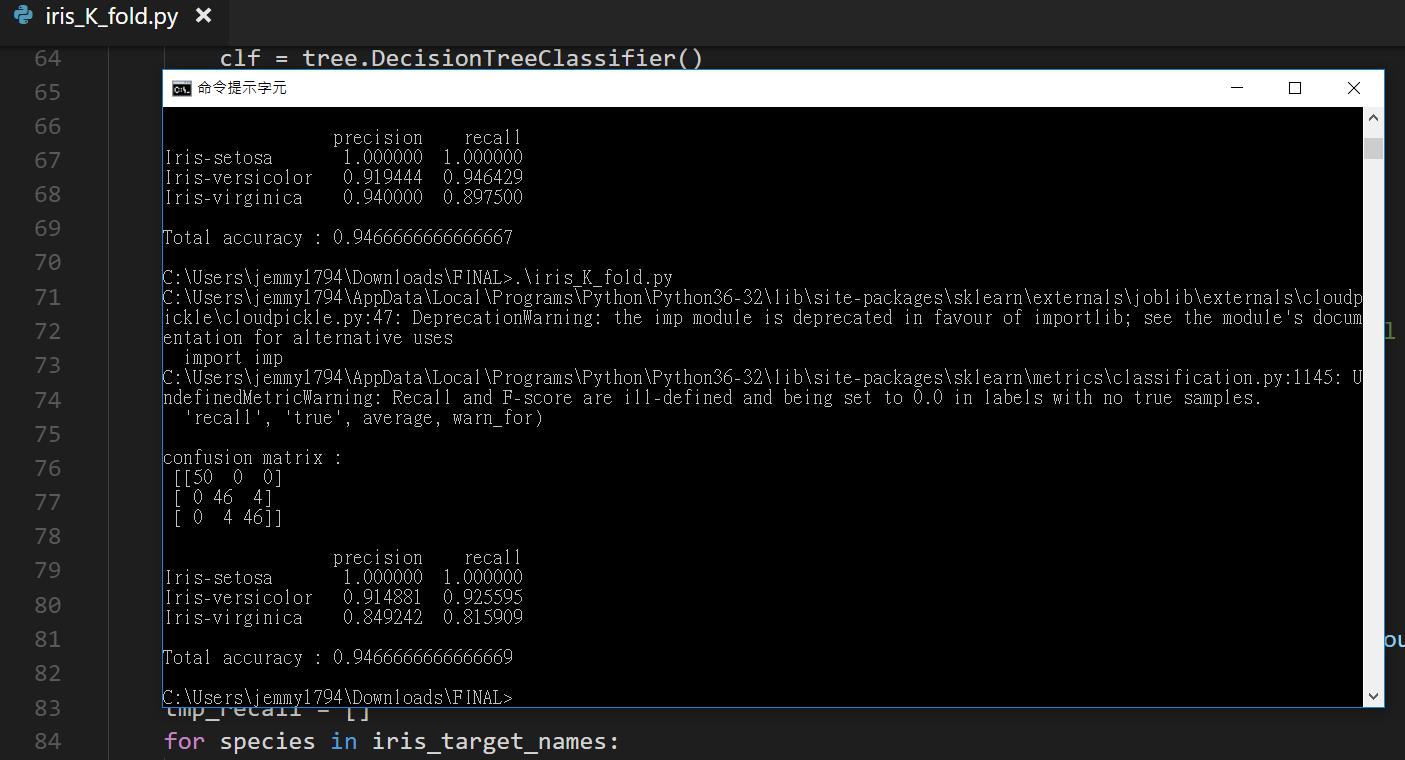
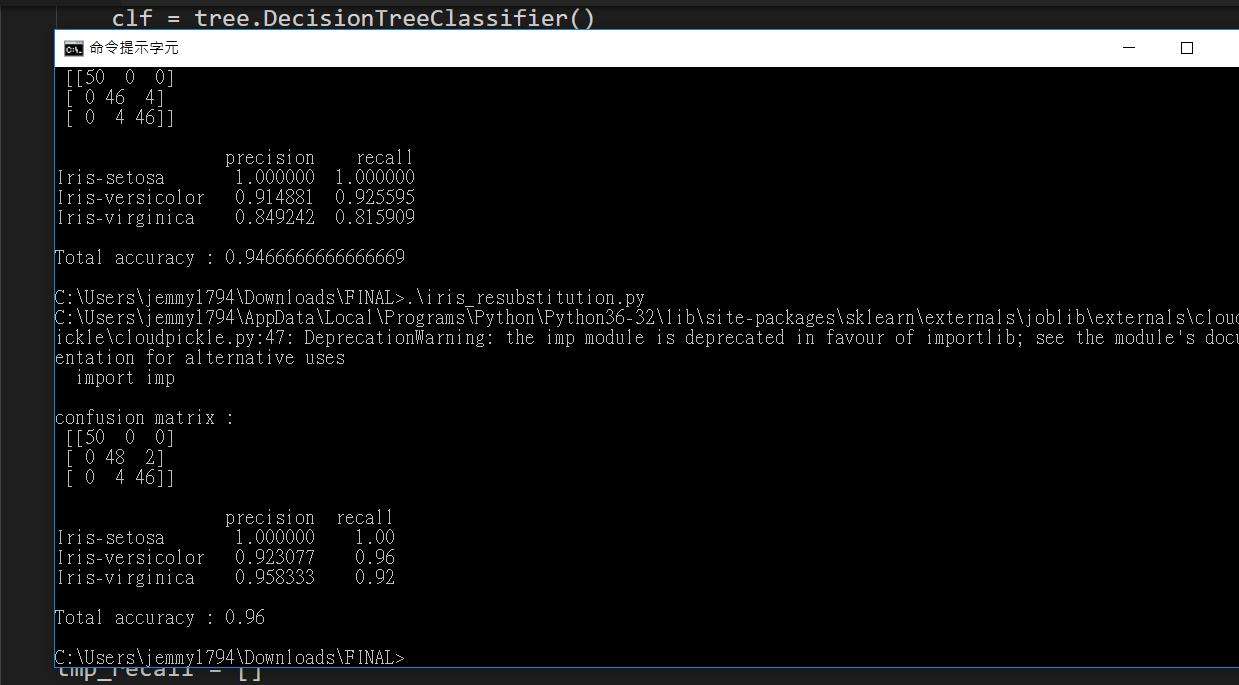
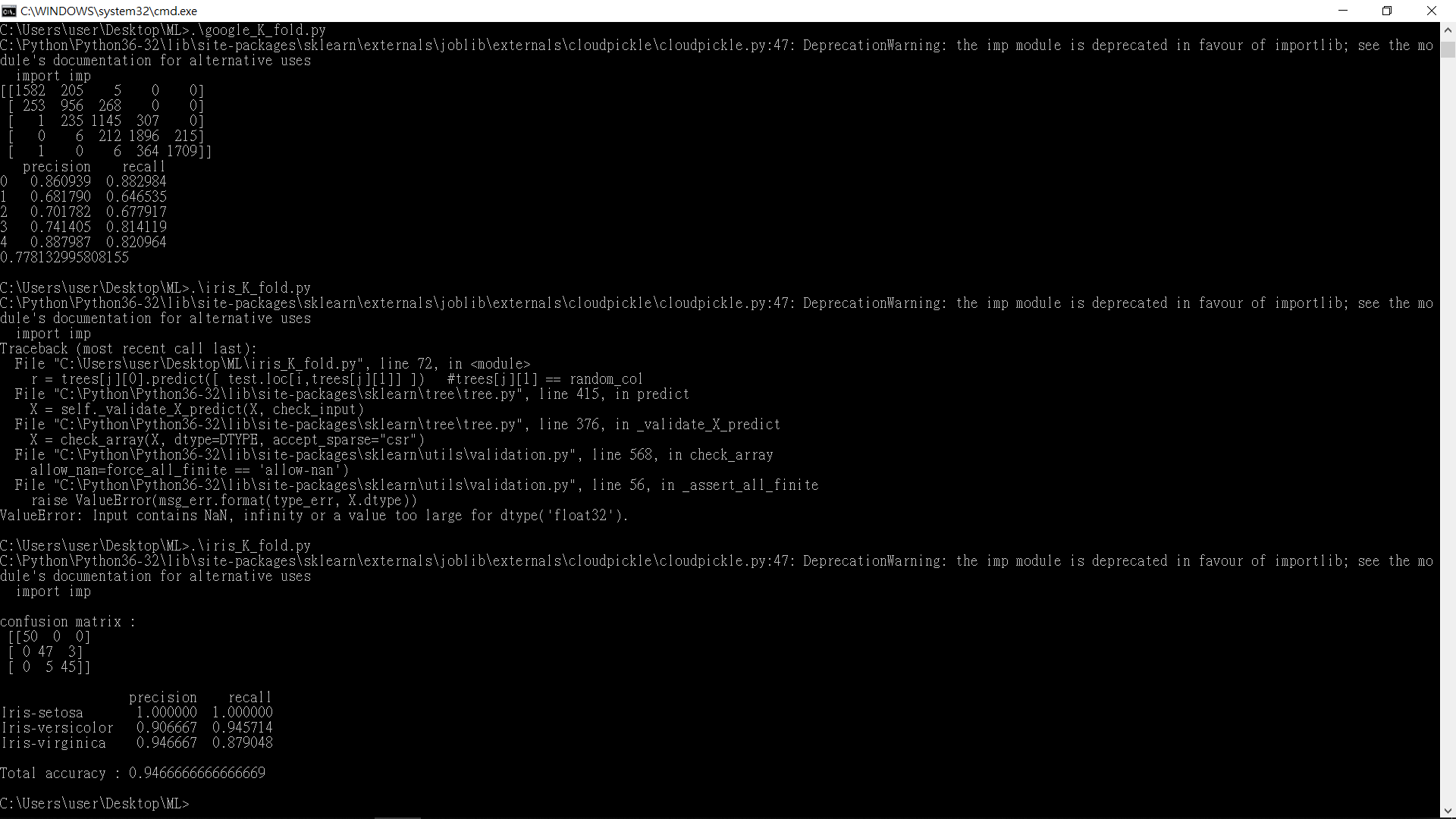
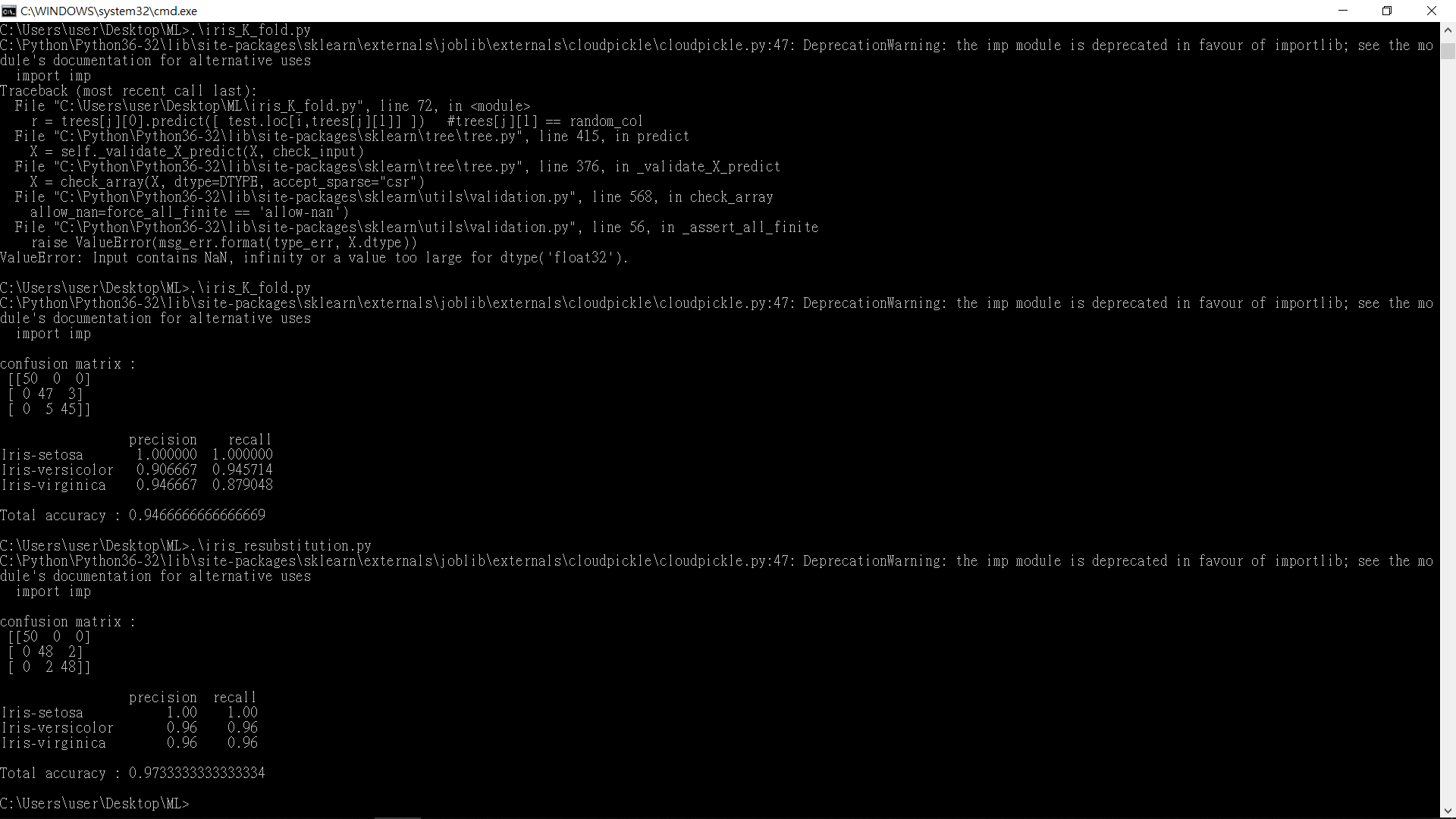
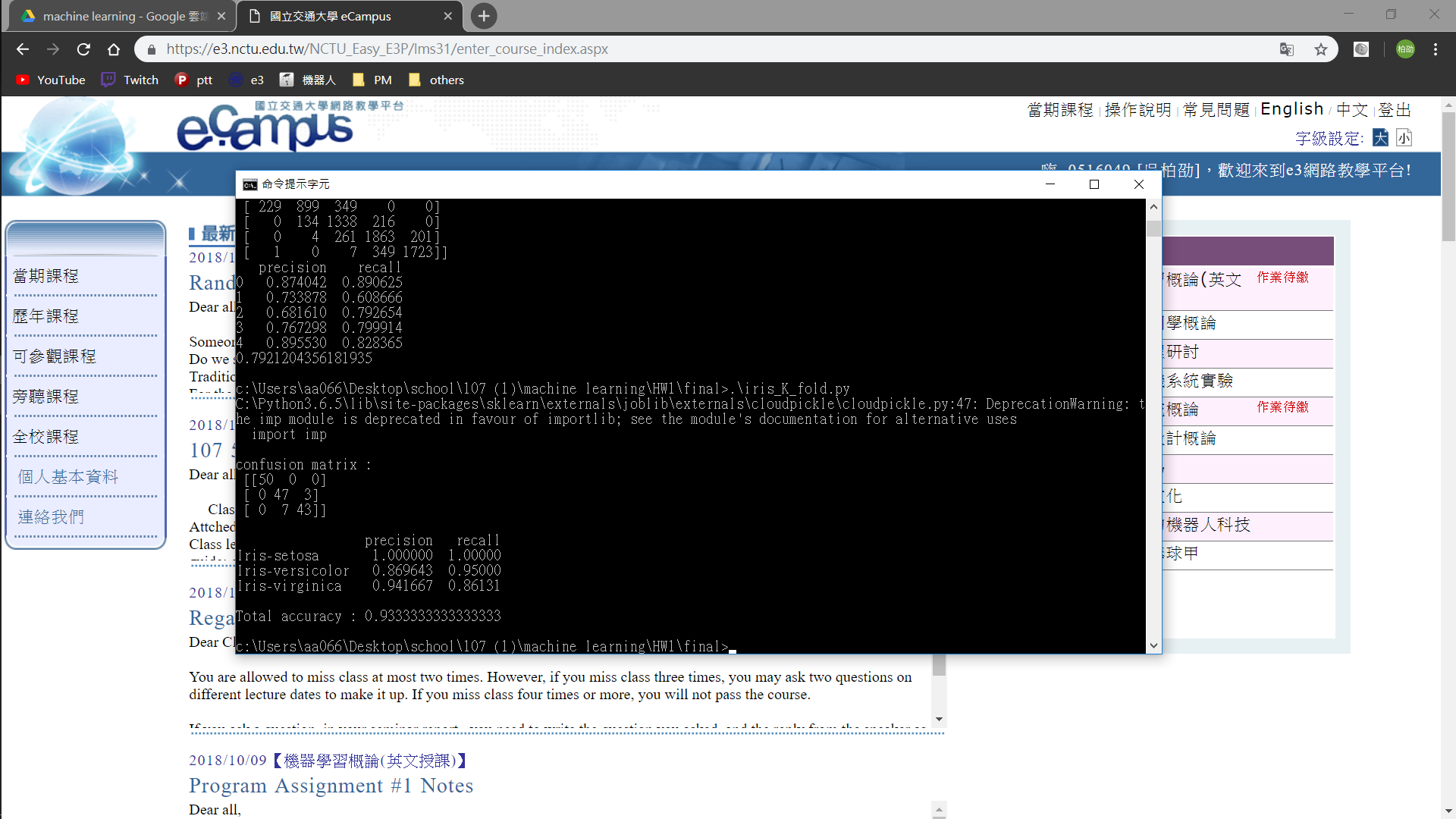
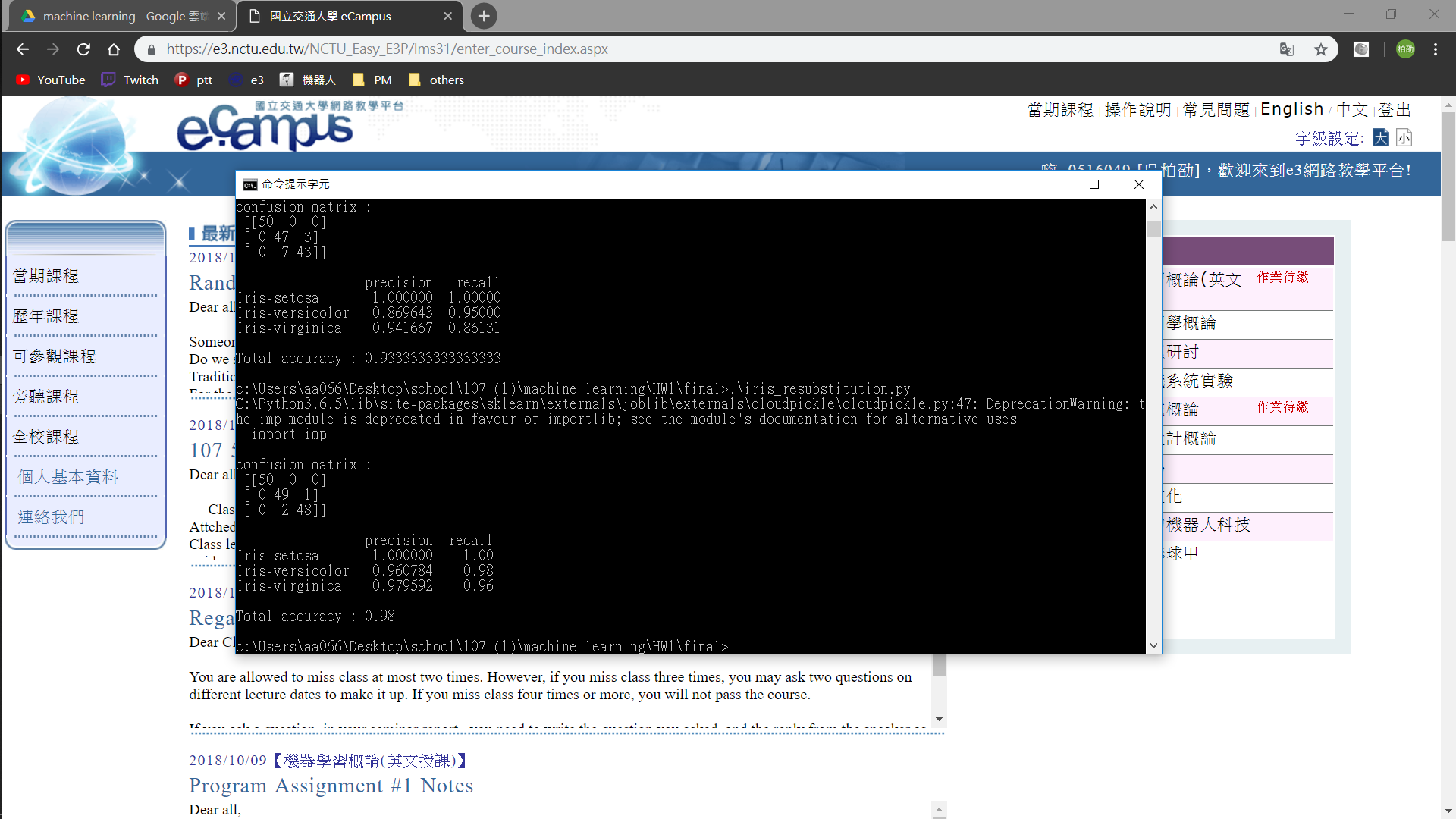
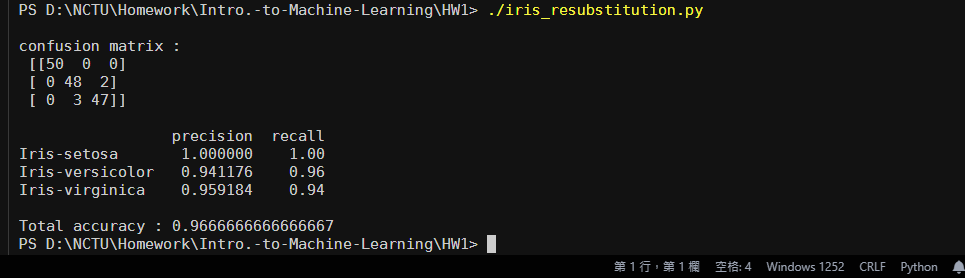
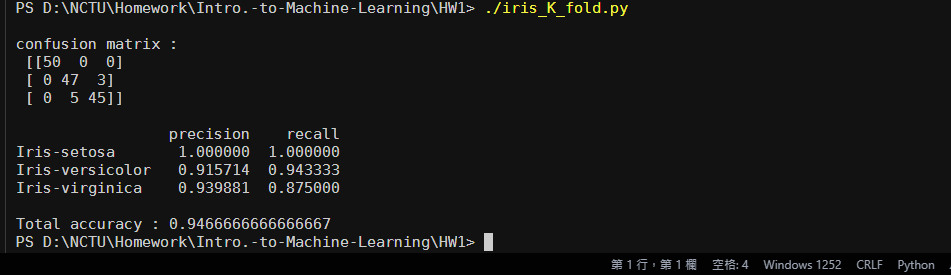
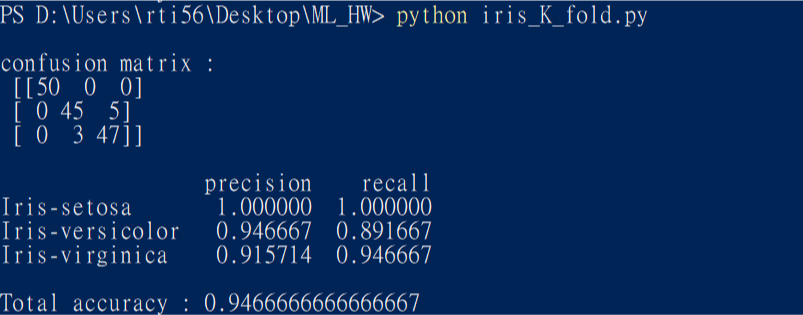
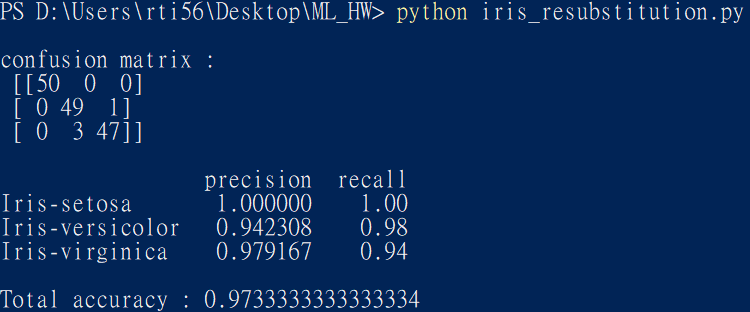
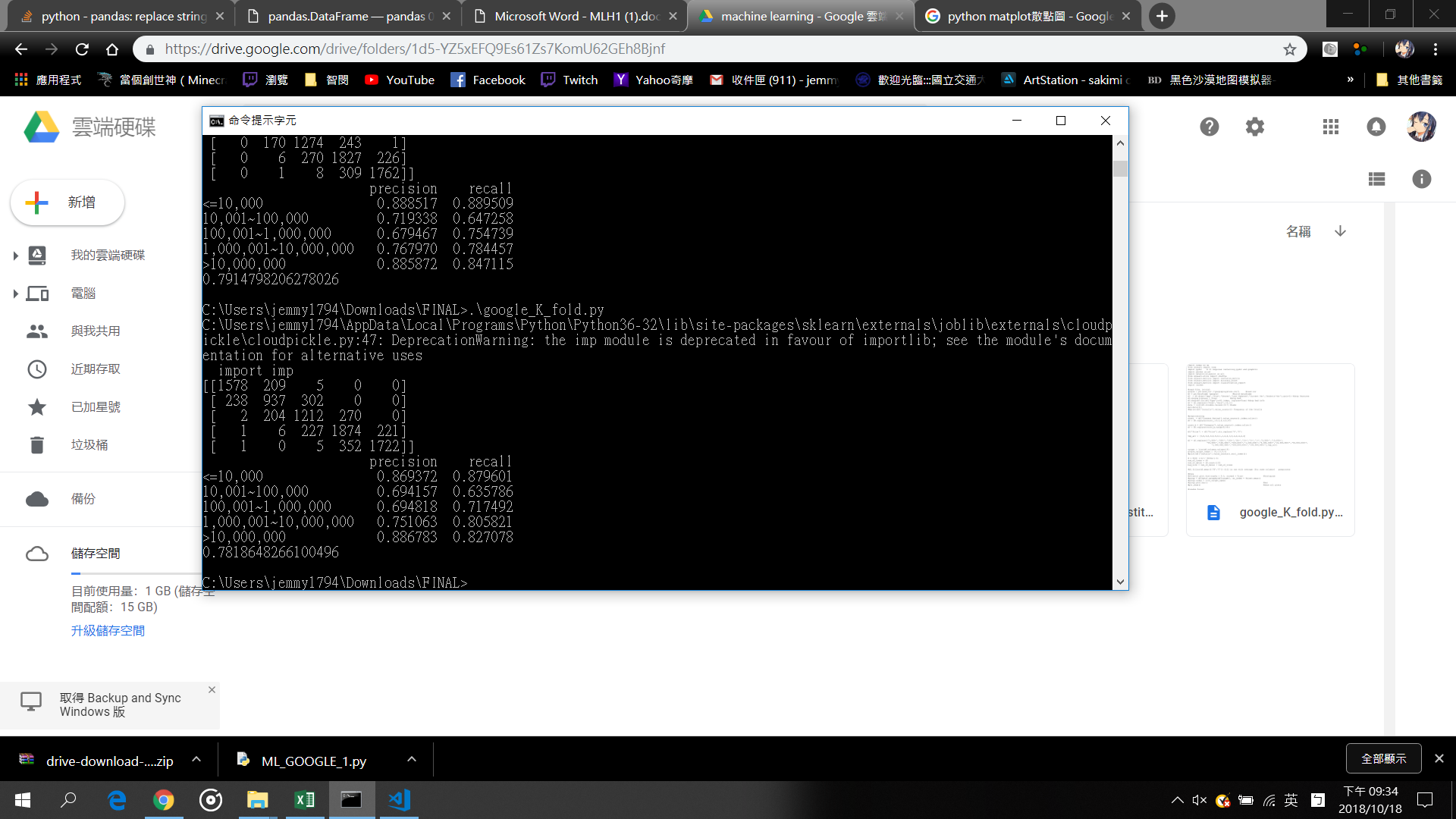
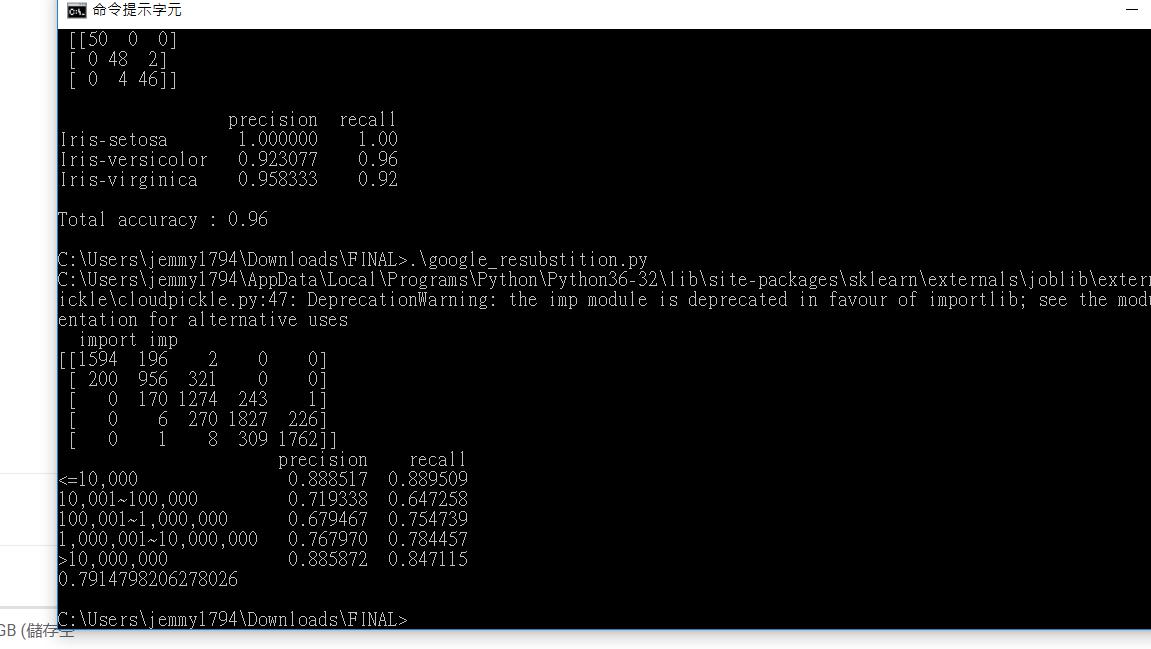
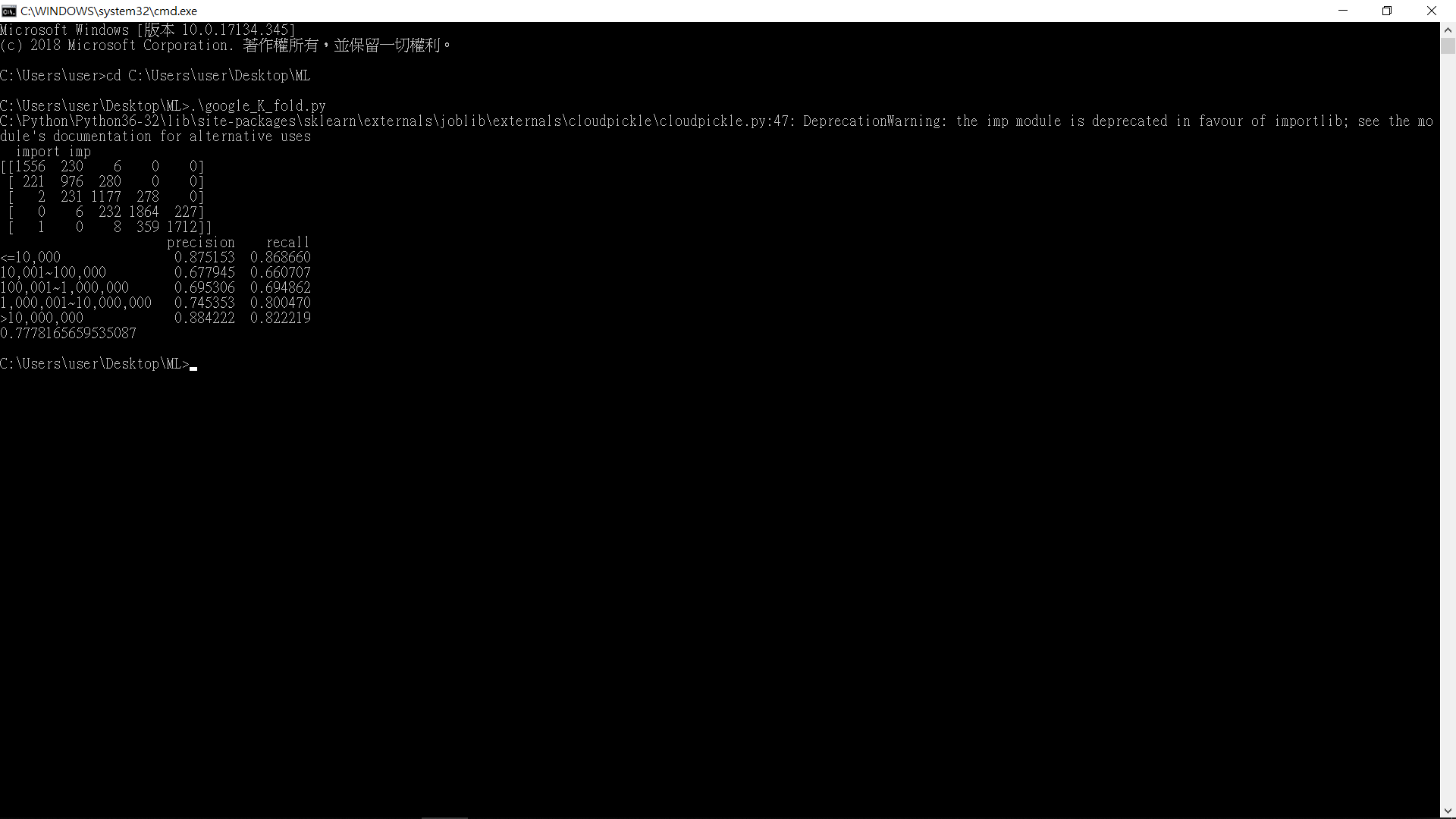
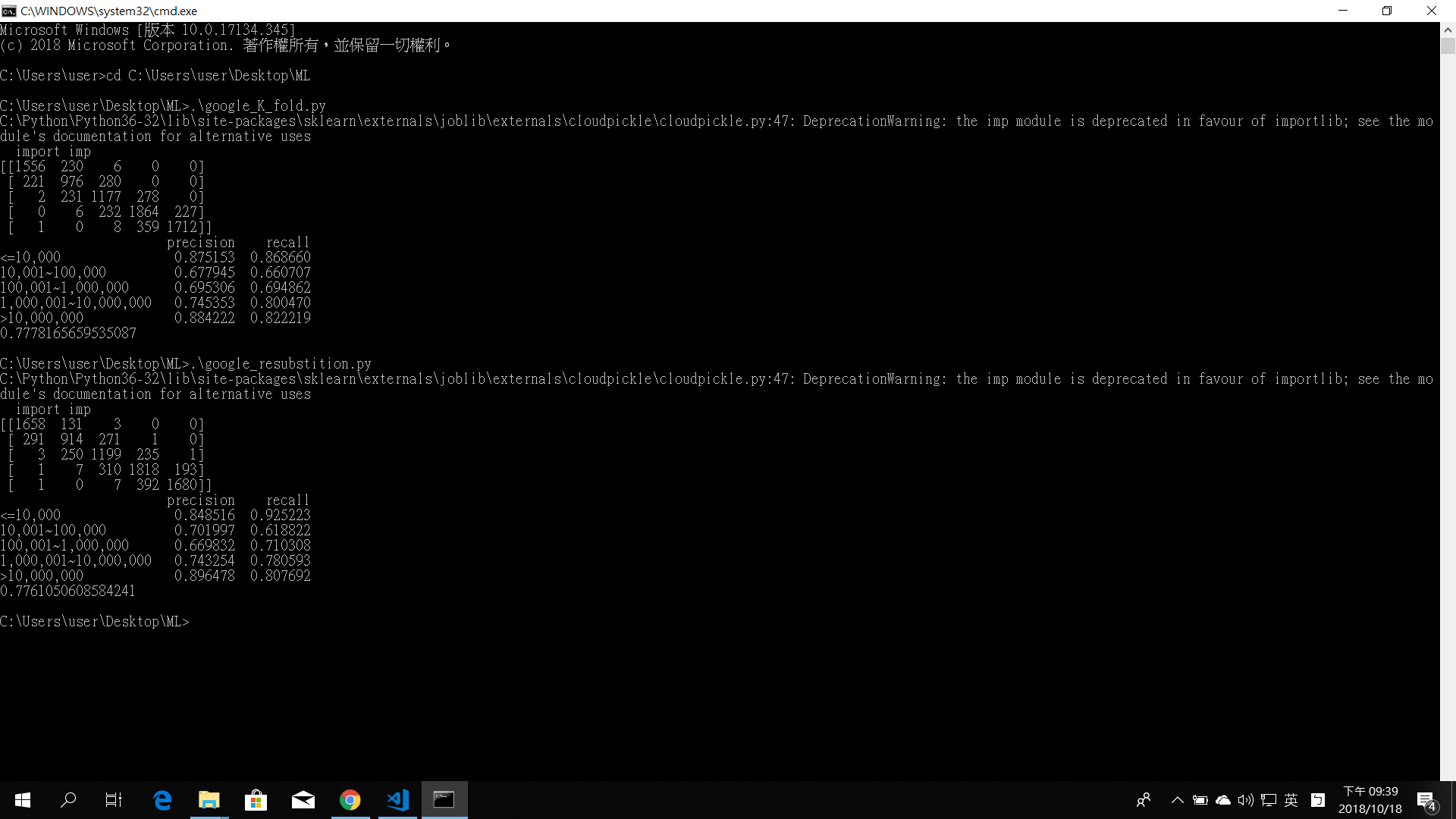
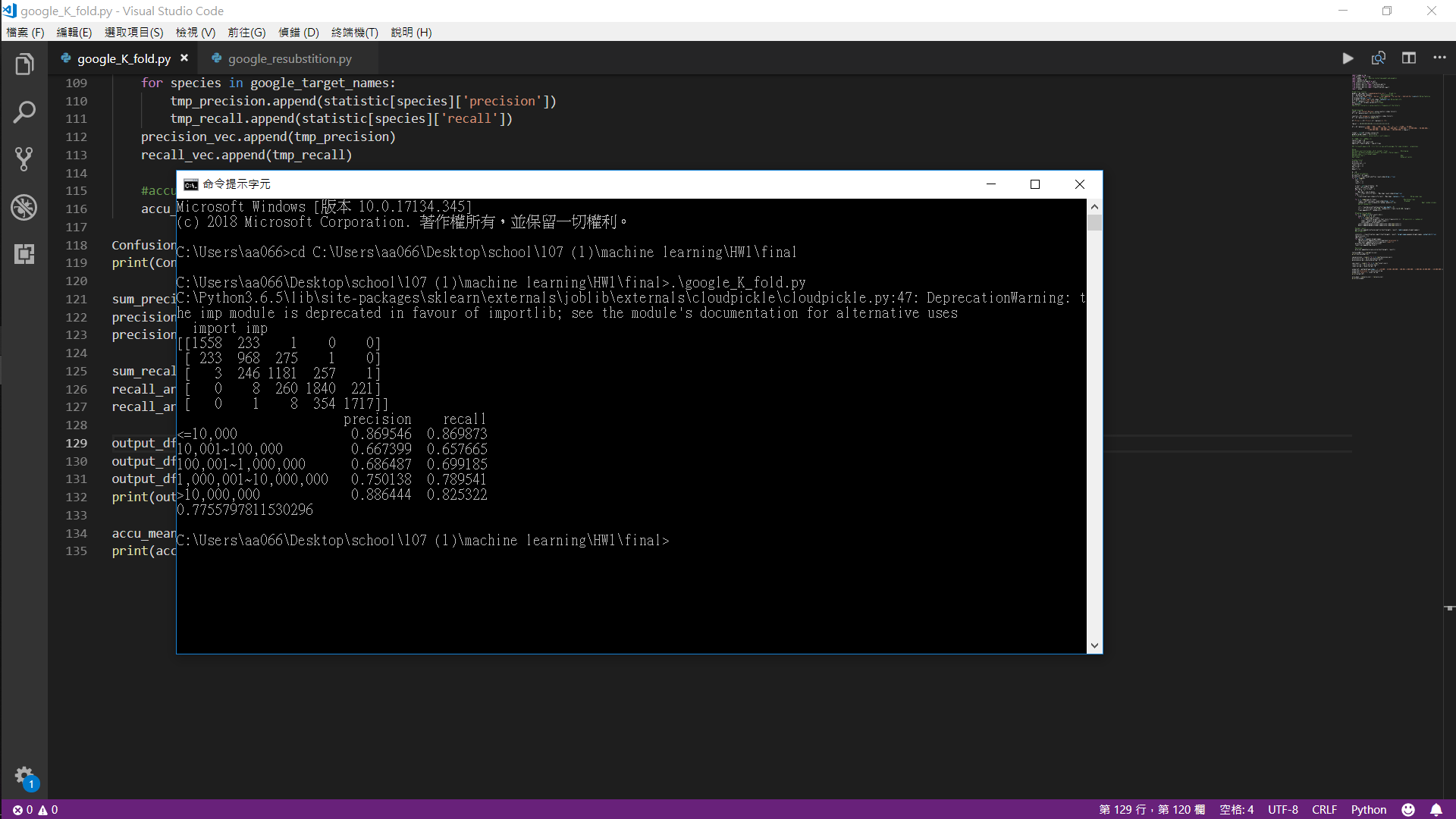
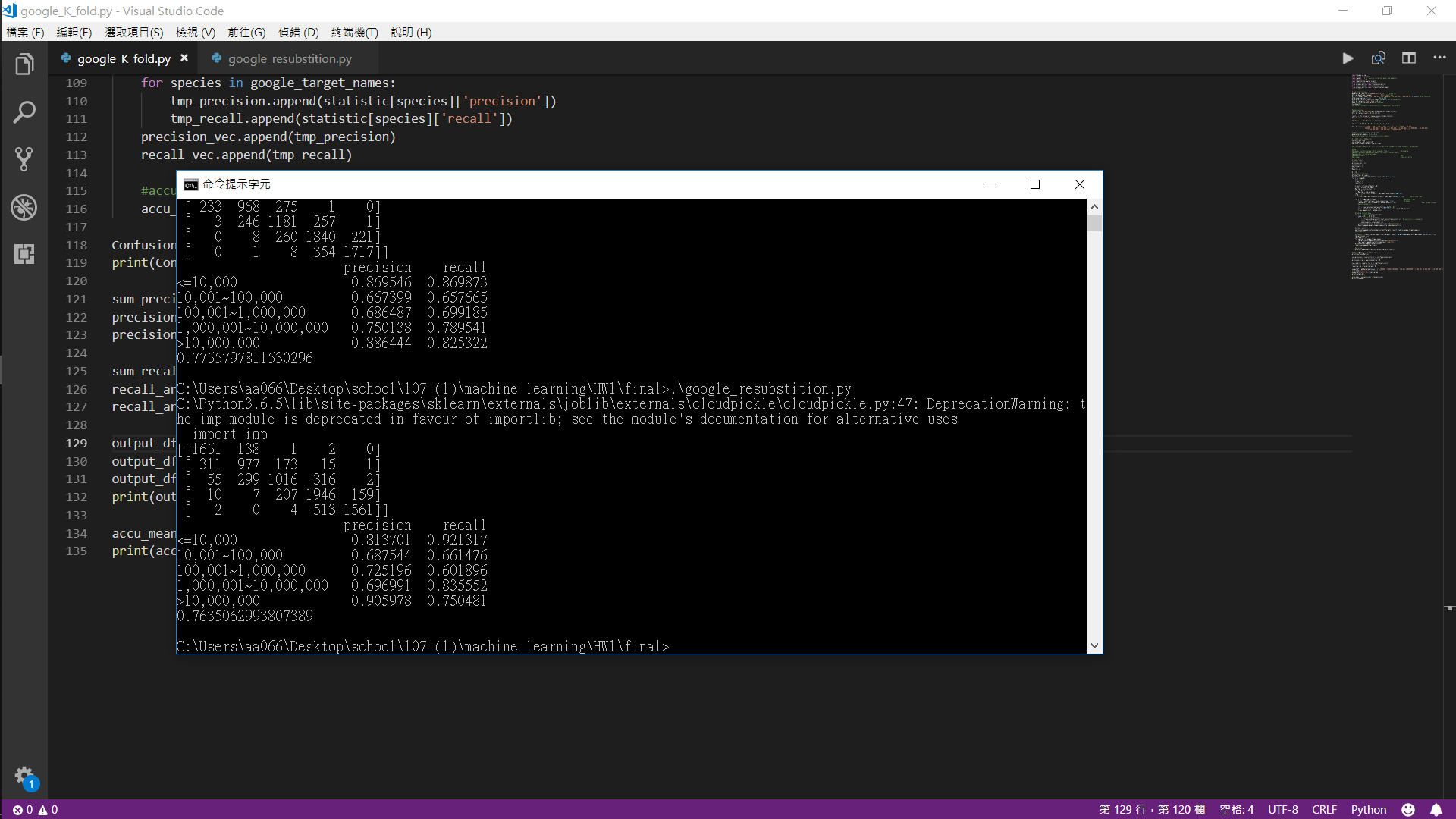
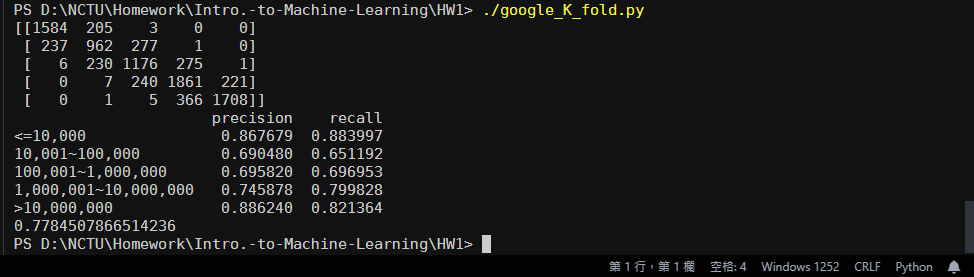
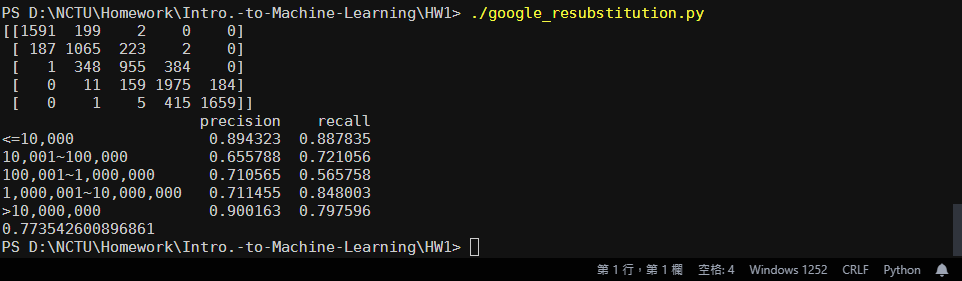
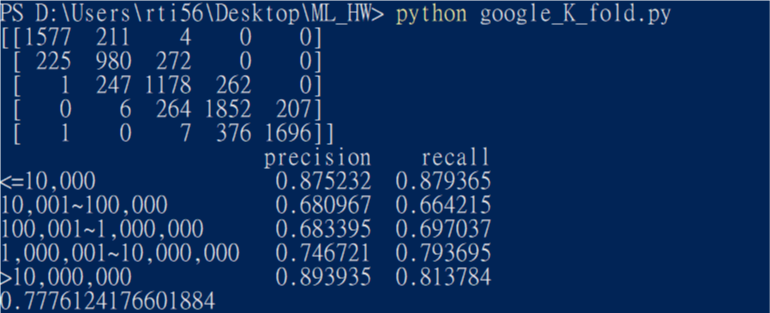
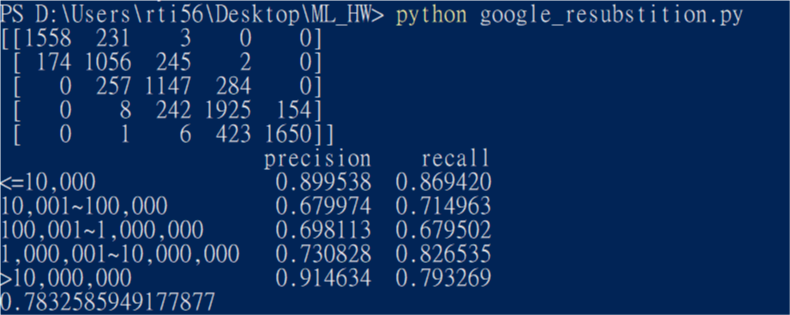
(3) scikit-learn 0.20.0

(4) scipy 1.1.0

(5) pydot 1.2.4

(6) matplotlib 3.0.0

(7) graphviz 2.3.8

1. **Basic statistic visualization of the data**▪Iris  
     
     
     
     
   ▪Google Play Store Apps  
   ****
2. **Data preprocessing methods**▪Iris  
   將品種設為target，petal width、petal length、sepal width、sepal length這四樣設為feature（這四樣feature通通為數字所以也不需要特別做處理）。  
   ▪Google Play Store Apps  
   將Installs設為target，並且再將Installs分類為＂<=10,000＂、＂10,001~100,000＂、＂100,001~1,000,000＂、＂1,000,001~10,000,000＂、＂>10,000,000＂，Category、Rating、Reviews、Type、Price、Content Rating這六樣設為feature(Category、Type以及Content Rating原屬字串，將其變為數字型式，如Type的Free跟Paid改為0、1，Category跟Content Rating依其種類分為0~種類總數)。
3. **How you generate decision tree and random forest models**▪Decision Tree  
   decision tree的部分是直接套用sklearn裡面的DecisionTreeClassifier()、fit()等函式去做生成和訓練的部分  
   ▪Random Forest  
   至於random forest，由於不能使用現成的函式，所以我們用了一個for loop去生成多棵decision tree，值得注意的是每一棵的decision tree的feature數目及種類是隨機的，這部分做法是先用shuffle打亂feature的順序，並生成(～總feature個數)個亂數去擷取隨機數目的feature，之後用sklearn裡預設的decision tree函式去建樹並從剛剛做好的隨機數目的feature中取部分data(Iris隨機取30個，Google Play Store Apps隨機取100個)作為training data去訓練模型。等到for loop結束後，再開一個2層的for loop將testing data一個一個餵給每一棵樹，然後再從眾多樹的預測結果中選擇得票率最高者作為最後的預測結果。
4. **The performance**▪Iris  
   最終的accuracy都有九成三以上。  
   以下為各組員執行結果(上圖為resubstitution，下圖為K-fold)：  
   0516025張智閔  
     
   0516032邱繼聖  
     
   0516049吳柏劭  
     
     
     
     
     
     
   0516215林亮穎  
     
   0516220李元毓  
     
     
   ▪Google Play Store Apps  
   最終的accuracy大概落在七成六至七成九之間。  
   以下為各組員執行結果(上圖為resubstitution，下圖為K-fold)：  
   0516025張智閔  
     
   0516032邱繼聖  
     
   0516049吳柏劭  
     
     
     
     
     
     
   0516215林亮穎  
     
   0516220李元毓  
   
5. **Conclusion**Iris的部分accuracy都還算高，主要也是因為Iris的data還算好預判，當選中某些feature時3種類型的Iris的分布可說是壁壘分明(可見visualization的Iris部分)，僅有少數data會跑到別的類型的涵蓋區，所以誤判率不高。  
     
   而Google Play Store Apps的部分accuracy就比Iris的部分來的少，主要是因為資料方面跟Iris比相對比較複雜，分布上也比較沒有Iris那樣來的壁壘分明，所以accuracy就比Iris來的少了。  
   這次作業我們花了很多心力在做，可以說是嘔心瀝血之作，雖然途中面臨到些許困難，但是憑著組員間的互相合作、協力分工，總算是完成了此次作業。