

Machine Learning Program assignment #2

0416303 楊博凱

Implementation :

在這次作業中，一開始我先 import 許多會用到的 module 到程式中，再來利用 with open() as csvfile 將 wine data 讀進 python 中。利用 split 將 data 整理過後將其轉為 numpy array 並分離 target feature 與 descriptive feature；並將數字部分從 string 轉成 float。

在進入 for 迴圈之前先建立 algorithm 與 metric lists，並令 kFold number 為 5。跑過 6 次 for 迴圈計算所要求的六種不同算法與距離，而當此計算之距離為 cosine 時將 data 轉成 normalized data 並使用 euclidean 作為距離計算方式，則可以得到 cosine distance。最後在進入 5 次 for 迴圈取 kFold 與 substitution 之平均值，算出結果。我的演算法中，以 KNeighborsClassifier 與 StratifiedKFold 搭配 5-Fold 與 30 weighted neighbors 作為計算條件。

Results :

Resubstitution 準確度：大約 92% K-Fold 準確度：大約 61% 距離比較：manhattan > Euclidean > cosine

```
*****kd_tree & cosine*****kd_tree & euclidean*****kd_tree & manhattan*****
Time:
0.0107569694519
***** Resubstitution Result *****
Confusion Matrix:
[[ 16  0  0  3  0  0  0]
 [ 0 136 12 12  1  0  0]
 [ 0  0 1332 114  9  0  0]
 [ 0  0  73 2095 28  0  0]
 [ 0  0 11 87 781  0  0]
 [ 0  0  1 16 4 152  0]
 [ 0  0  0  1  0  0  4]]
Accuracy Score:
0.92229481421
***** 10-Fold Result *****
Confusion Matrix:
[[ 0  0  0  3  0  0  0]
 [ 0  5 12 12  1  0  0]
 [ 0  0 166 114  9  0  0]
 [ 0  0  73 336 28  0  0]
 [ 0  0 11 87 77  0  0]
 [ 0  0  1 16 4 12  0]
 [ 0  0  0  1  0  0  0]]
Accuracy Score:
0.610838445808

Time:
0.0117794036865
***** Resubstitution Result *****
Confusion Matrix:
[[ 16  0  1  2  0  0  0]
 [ 0 135 14 13  0  0  0]
 [ 0  0 1330 118  8  0  0]
 [ 0  0  75 2101 21  0  0]
 [ 0  0 15 84 780  0  0]
 [ 0  0  3 16 2 152  0]
 [ 0  0  0  1  0  0  4]]
Accuracy Score:
0.922784810127
***** 10-Fold Result *****
Confusion Matrix:
[[ 0  0  1  2  0  0  0]
 [ 0  4 14 13  0  0  0]
 [ 0  0 164 118  8  0  0]
 [ 0  0  75 342 21  0  0]
 [ 0  0 15 84 76  0  0]
 [ 0  0  3 16 2 12  0]
 [ 0  0  0  1  0  0  0]]
Accuracy Score:
0.613292433538

Time:
0.0113987922668
***** Resubstitution Result *****
Confusion Matrix:
[[ 16  0  1  2  0  0  0]
 [ 0 135 15 11  0  0  0]
 [ 0  0 1326 122  8  0  0]
 [ 0  0  70 2105 22  0  0]
 [ 0  0 16 80 782  0  0]
 [ 0  0  2 18 2 151  0]
 [ 0  0  0  1  0  0  4]]
Accuracy Score:
0.923111474071
***** 10-Fold Result *****
Confusion Matrix:
[[ 0  0  1  2  0  0  0]
 [ 0  4 15 11  0  0  0]
 [ 0  0 160 122  8  0  0]
 [ 0  0  70 346 22  0  0]
 [ 0  0 16 80 78  0  0]
 [ 0  0  2 18 2 11  0]
 [ 0  0  0  1  0  0  0]]
Accuracy Score:
0.614928425358

*****brute & cosine*****brute & euclidean*****brute & manhattan*****
Time:
0.0069953918457
***** Resubstitution Result *****
Confusion Matrix:
[[ 16  0  1  2  0  0  0]
 [ 0 136 14 11  2  0  0]
 [ 0  0 1331 114 11  0  0]
 [ 0  0  76 2094 27  0  0]
 [ 0  0 13 86 780  0  0]
 [ 0  0  2 17 3 152  0]
 [ 0  0  0  1  0  0  4]]
Accuracy Score:
0.921518987342
***** 10-Fold Result *****
Confusion Matrix:
[[ 0  0  1  2  0  0  0]
 [ 0  5 14 11  2  0  0]
 [ 0  0 165 114 11  0  0]
 [ 0  0  76 335 27  0  0]
 [ 0  0 13 86 76  0  0]
 [ 0  0  2 17 3 12  0]
 [ 0  0  0  1  0  0  0]]
Accuracy Score:
0.606952965235

Time:
0.0070650775452
***** Resubstitution Result *****
Confusion Matrix:
[[ 16  0  1  2  0  0  0]
 [ 0 134 13 14  0  0  0]
 [ 0  0 1332 115  8  0  0]
 [ 0  0  76 2096 24  0  0]
 [ 0  0 13 86 779  0  0]
 [ 0  0  3 16 2 153  0]
 [ 0  0  0  1  0  0  4]]
Accuracy Score:
0.922253981217
***** 10-Fold Result *****
Confusion Matrix:
[[ 0  0  1  2  0  0  0]
 [ 0  3 13 14  0  0  0]
 [ 0  0 166 115  8  0  0]
 [ 0  0  76 337 24  0  0]
 [ 0  0 13 86 75  0  0]
 [ 0  0  3 16 2 13  0]
 [ 0  0  0  1  0  0  0]]
Accuracy Score:
0.61063394683

Time:
0.00508456230164
***** Resubstitution Result *****
Confusion Matrix:
[[ 16  0  2  2  0  0  0]
 [ 0 134 15 13  0  0  0]
 [ 0  0 1332 119  4  0  0]
 [ 0  0  72 2102 22  0  0]
 [ 0  0 18 86 775  0  0]
 [ 0  0  2 15 4 153  0]
 [ 0  0  0  0  0  0  4]]
Accuracy Score:
0.922131482238
***** 10-Fold Result *****
Confusion Matrix:
[[ 0  0  2  2  0  0  0]
 [ 0  3 15 13  0  0  0]
 [ 0  0 166 119  4  0  0]
 [ 0  0  72 343 22  0  0]
 [ 0  0 18 86 71  0  0]
 [ 0  0  2 15 4 13  0]
 [ 0  0  0  0  0  0  0]]
Accuracy Score:
0.610020449898
```

Using library :

csv 、 numpy 、 sklearn 、 random 、 timeit

```
1 import csv
2 import numpy as np
3 from sklearn.neighbors import KNeighborsClassifier
4 from sklearn.metrics import confusion_matrix
5 from sklearn.metrics import accuracy_score
6 from sklearn import cross_validation
7 from sklearn.model_selection import StratifiedKFold
8 import random
9 import timeit
10 from sklearn.preprocessing import normalize
11 from sklearn import preprocessing
```

Environment & Language :

```
kai@kai-UX305FA:~$ sudo lsb_release -a
[sudo] password for kai:
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 16.04.2 LTS
Release:        16.04
Codename:       xenial
```

```
kai@kai-UX305FA:~$ python
Python 2.7.12 (default, Nov 19 2016, 06:48:10)
[GCC 5.4.0 20160609] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
```