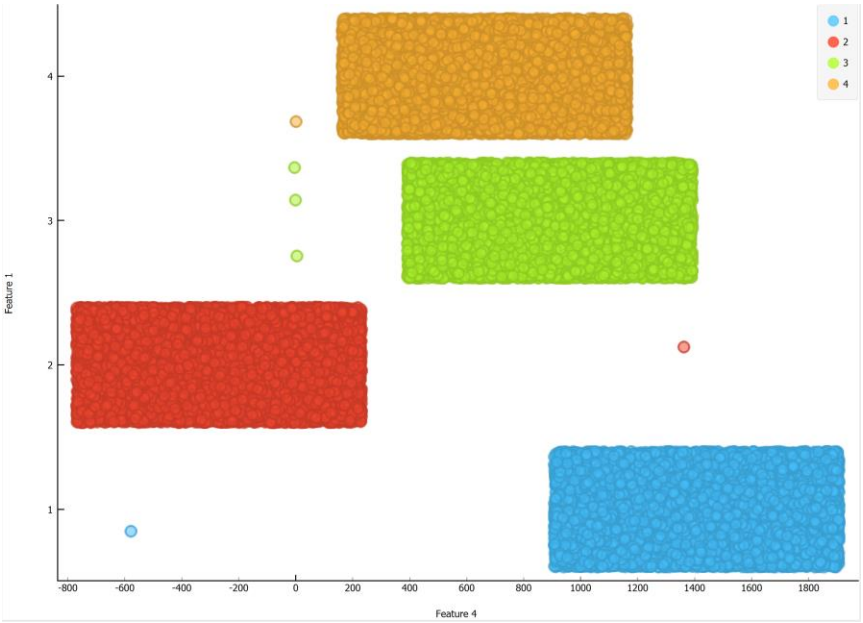
Project 3: outlier detection

Haijin He

1, First I tried scatter plot to explore the data.

Below is a figure of Figure 4 plot against the class. The isolated points are obviously outliers. These outliers can be seen in various plots using feature against class. But as you can see from this plot, there are more than 5 outliers. The outliers from scatter plot is manually put down into ***scatterplotresult.txt***



2, To find these outliers using algorithms, I first try distance based outlier detection.
(1)
Assumptions:
Since data have 4 classes, assume each class is generated through different mechanisms, so I run algorithm on each class.
for simplicity, I first run on each features.
result:
I run for a r=10, pi=0.0004(4 points in a 10000 sample dataset).
(the parameter can be set in a relatively large range to achieve similar result)
Hit 8 outliers.
The result is saved in "***DistanceOutlierUni.csv***"
It is easy to see that 5 of them are clearly outliers. 3 of them are composed of integers, and the other 2 is close to them in distance, and have index number that is dividable by 5000.
The other 3 I also consider them outliers, since they deviate from main population a lot.

	F1	F2	F3	F4	F5	F6	F7	F8	id
3	1	1	1	1	1	1	1	1	999
3	0.53766	1.83388	2.25885	0.86217	0.31876	1.30769	0.43359	0.3426	4999
1	81.5004	635.777	576.507	1674.58	563.458	853.242	975.856	608.32	6520
4	1	2	3	4	5	6	7	8	14999
3	8	7	6	5	4	3	2	1	24999
2	801.378	639.284	1364.20	126.288	746.055	252.981	672.228	161.42	28953
2	0.22279	1.95272	0.86331	0.08802	0.23293	0.04135	0.42199	3.3322	34999
3	407.624	422.829	904.234	386.064	568.928	570.661	1077.36	414.11	38128

(2) multi-variate distance based outlier detection

I apply multi-variate version of distance based outlier detection to data set without breaking into classes.

First on un-normalized data.

The result is not very good.

Then I tried to normalize data by feature.

The parameters for distance based outlier detection:

r=0.0025

pi=0.0002 (8 points in 40000 samples)

Here the r has to set carefully, r=0.005 will yield no outlier, and r=0.002 will give too many outliers.

The result is automatically saved to "***DistanceOutlierMulti.csv***", and attached below.

Exactly 5 outlier is found by this method. And It looks like there are the outliers.

	F1	F2	F3	F4	F5	F6	F7	F8	id
3	1	1	1	1	1	1	1	1	999
3	0.53766	1.83388	-2.25885	0.86217	0.31876	-1.30769	0.4335	0.3426	4999
4	1	2	3	4	5	6	7	8	14999
3	8	7	6	5	4	3	2	1	24999
2	0.22279	1.95272	0.86331	0.08802	0.23293	0.04135	0.4219	3.3322	34999

3, Cluster based outlier detection
On data without normalization, I also used hierarchical clustering with average distance, assuming the data set will form 5 clusters, with 4 clusters being the original classes, and outliers forming the smallest cluster.
But due to the long running time, I could not finish it. I tried on smaller batches like 4000, and the 5 outliers found in distance based scan will form the smallest cluster.

Files:

Outlier.py the script the runs outlier detection.
The 2 outlier detection functions are DBOutlierByFeature(), and GloabalDBOutlier(), called separately by outlier1() and outlier2(). Normalization function is normalizeByFeature().

DistanceOutlierMulti.csv result from multi-variate distance based outlier detection

DistanceOutlierUni.csv result from single variate distance based outlier detection

Data.csv original data converted to csv.

convertToCSV.py script that converts original data file to csv format.

scatterplotresult.txt result by looking at scatter plot of features.