Instructions

You can run the program in a python 2.7 interpreter. (I used PyDev for Eclipse)

The program takes 3 files as input: a datafile and two clustering assignments. The file names can be specified at the beginning of the source code. The datafiles for the wine and community datasets are located in files named wnorm.csv and comm.csv respectively. The clustering assignments are located in the wnorm-clusterings and comm-clusterings folders. Each folder has several clusterings with different numbers of clusters and different seeds. You could use any pair of clusterings from each folder.

The results of my experiments are located in the comm-matrix.xlsx and wnorm-matrix.xslx files. These files contain tables with disagreement distances based on membership and based on my distance measure, as well as comparisons of results for different seeds/clusters numbers.

Wine Processing

I downloaded the file and modified the names and dates files into a csv file so I could work with the data. The data points were all on different scales, and since my measure used distance, I normalized the data within excel ( for each column).

Note: I used the following commands to process the data

cat wine-clus-1.arff | gawk '/@data/,/177/' | tail -n +2 | cut -d"," -f1,16   
cat wine-clus-1.arff | gawk '/@data/,/177/' | tail -n +2 > wine-clus-1.csv

Communities Processing

I downloaded the file and converted it into proper format to import into weka. The data for this file was already normalized.  
I removed community name from the file as it was nominal, and removed fold, since it didn’t seem to have an effect on the clusters. I applied an Attribute Selection filter using the BestFirst Algorithm and chose the following algorithms for clustering.   
Selected attributes: 6,7,17,42,48,54,72,75,94,105 : 10  
                     racepctblack  
                     racePctWhite  
                     pctWWage  
                     MalePctDivorce  
                     PctKids2Par  
                     PctIlleg  
                     PctPersDenseHous  
                     HousVacant  
                     NumStreet  
                     LemasTotReqPerPop  
I noticed that LemasTotReqPerPop consisted of about 3/4ths missing values so I removed it, which strongly improved the results of the clustering.

Note: I used the following commands to process the output data.   
cat comm-best9f.arff | grep "@attribute" | cut -d" " -f2 | tr "\n" "," | gawk '{print $0}' | sed 's/,$//' > com9b.csv  
cat comm-best9f.arff | gawk '/@data/,EOF' | tail -n +2 >> com9b.csv