**Geoffrey Pettet and Tim Darrah EDM Assignment Nov 28, 2017**

We did our data analysis in Python using the scipy, sklearn, and statsmodels libraries. Although RapidMiner was recommended, this approach provided more freedom and flexibility with our analysis. Visit <https://github.com/darrahts/PythonScripts/tree/master/DataAnalysis> for our code and other files. First, we carried out several trials with regression analysis, then we ran several trials with decision trees, and finally conducted clustering analysis using hierarchical clustering techniques.

1. **Regression Function to Predict Map Score**:

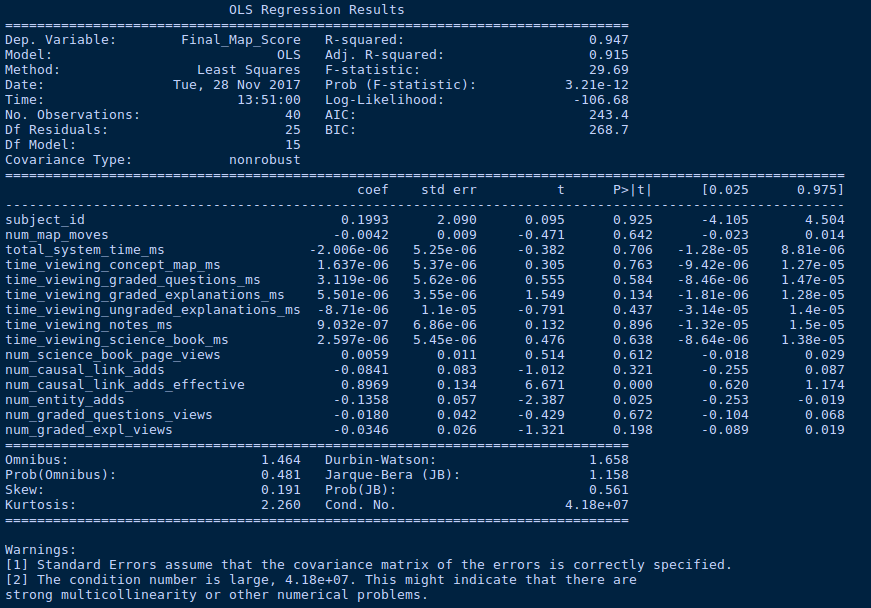
Our First goal when testing different linear regression techniques was to determine which subsets of features lead to the best performance when predicting the final map score. The subsets we tried were:

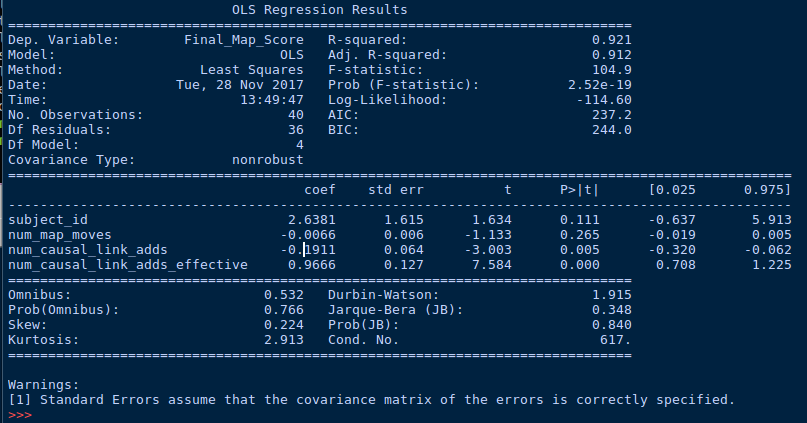
* Subset 1: (num\_map\_moves, time\_viewing\_concept\_map\_ms, num\_causal\_link\_adds, num\_causal\_link\_adds\_effective)
  + These represent the major features related to the concept map
  + The feature “time\_viewing\_concept\_map\_ms” and gave warnings for collinearity, and after experimentation we found that this was collinear with several other features. This makes sense since the other features deal with editing the concept map, and therefore time viewing the concept map is inherently related.
* Subset 2: (total\_system\_time\_ms, time\_viewing\_graded\_questions\_ms, time\_viewing\_notes\_ms, time\_viewing\_science\_book\_ms)
  + These represent the non-map related timing features
* Subset 3: (num\_map\_moves, time\_viewing\_concept\_map\_ms, time\_viewing\_graded\_questions\_ms, time\_viewing\_graded\_explanations\_ms, time\_viewing\_ungraded\_explanations\_ms, num\_causal\_link\_adds, num\_causal\_link\_adds\_effective)
  + This represents the most predictive of the second subset combined with the first subset
* Subset 4: (all features)
  + This is all features in the dataset, including a new feature representing whether the sample was a group or an individual (0 = individual, 1 = group), which we call subject\_id.
* Subset 5: (subject\_id, num\_map\_moves, num\_causal\_link\_adds, num\_causal\_link\_adds\_effective)
  + This dataset includes all of the features that seemed to have the greatest impact in the full feature subset analysis

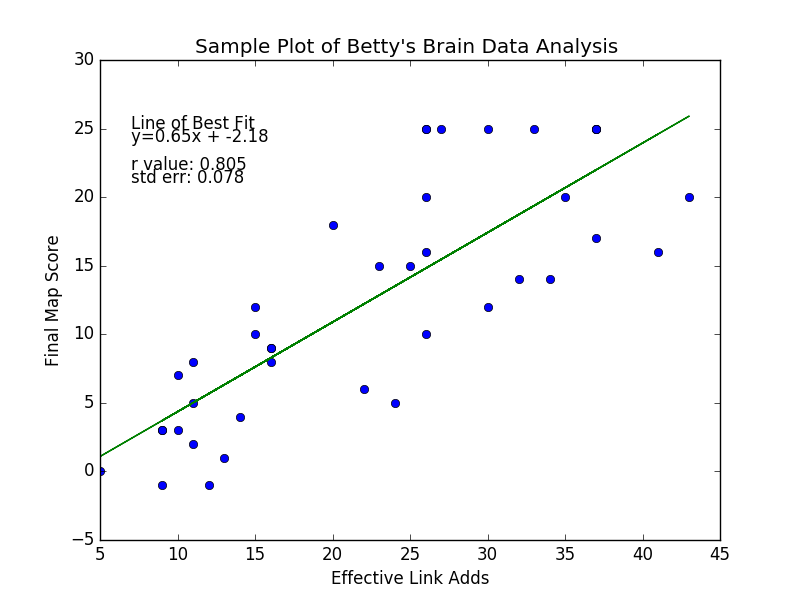
We performed classic linear regression using ordinary least squares (OLS) as a baseline. The results of the subsets are shown below.

|  |  |
| --- | --- |
| Feature Subset | Prediction Accuracy |
| Subset 1 | 91.6% |
| Subset 2 | 69.5% |
| Subset 3 | 92.9% |
| Subset 4 | 94.7%\* |
| Subset 5 | 92.1% |

This shows that utilizing the entire dataset provides the most accurate prediction results on the dataset. Below are the detailed results of this subset:

The above details show that while many of the features have a large impact (the subject\_id [i.e. group or individual] and the num\_causal\_link\_adds\_effective in particular), there are many that seem to have little correlation with the final map score. This includes many of the time based features. The second warning at the bottom also indicates that some features should be removed due to high collinearity. This implies that despite the full feature set having the highest accuracy, it might be best to remove these features to make the model more generalizable.

We do this in the 5th subset, which has a slightly lower accuracy of 92.1%. Despite this lower r2, This model is more generalizable:

The highest correlation by far appears to be the num\_causal\_link\_adds\_effective feature, which is backed up by plotting the feature values against the subjects’ final map scores:

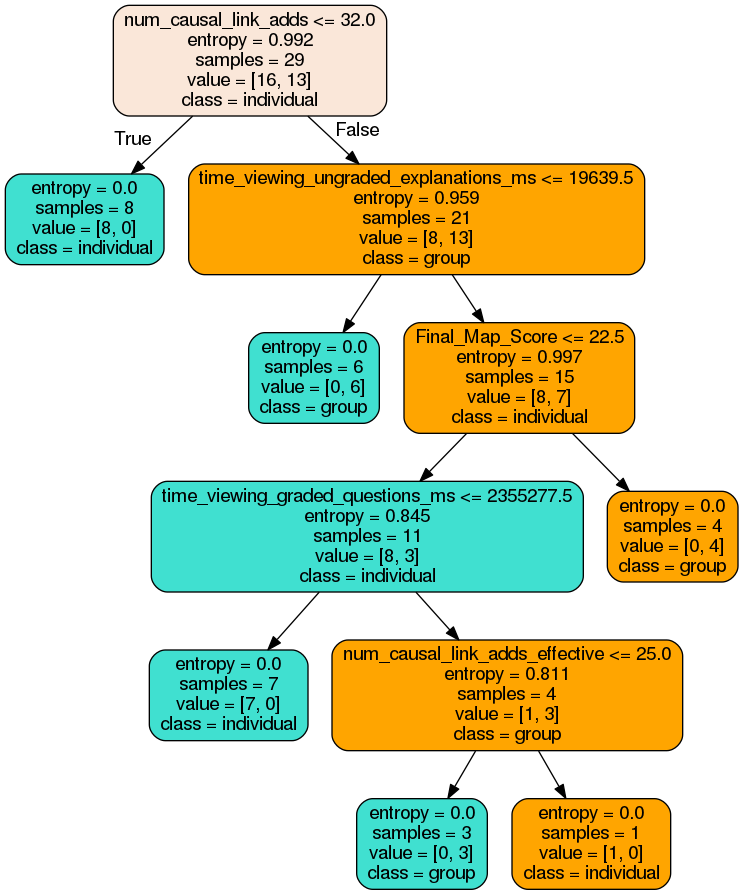
After seeing that the full data set gave the best accuracy when using OLS, we also tried a few other techniques using entire feature set. We tried the sklearn linear regression technique and Support Vector Regression, both of which had lower accuracies than the OLS approach (84.8%, and 77.4% respectively). This means that using OLS and the entire feature set gave the best results.

1. **Supervised Learning to Differentiate Two Groups**:

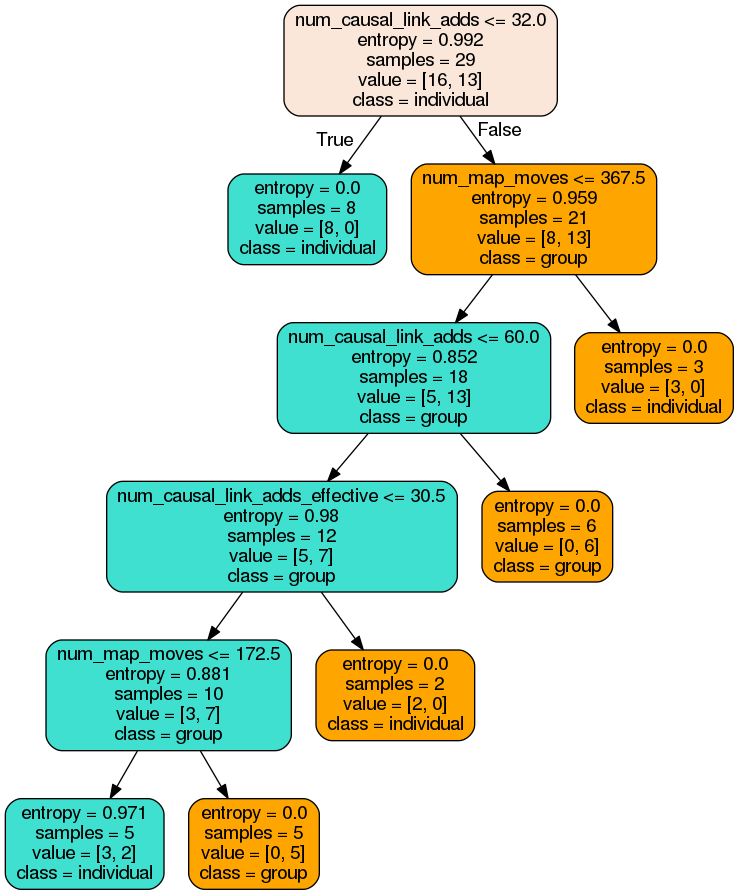
To predict if a sample was produced by an individual or a group, we use a decision tree classifier. When building our trees, we tested using two different splitting techniques (gini impurity, which measures how often a randomly chosen element from the set would be incorrectly labeled if it was randomly labeled according to the distribution of labels in the subset, and information gain (entropy), which uses the concept of entropy to split nodes based on increasing the structure of the classified data). We also tried removing some of the samples when training that we saw as outliers (A08, A13, G04, G08) to try and make the model more generalizable. We later decided that since we did not include this data in the testing/validation set, that the results were invalid. Then we ran the tests altering the feature set used.

We used cross validation when evaluating the techniques to avoid overfitting, and averaged the accuracies of the trees for our results. One of the sample created trees for each method and results are below:

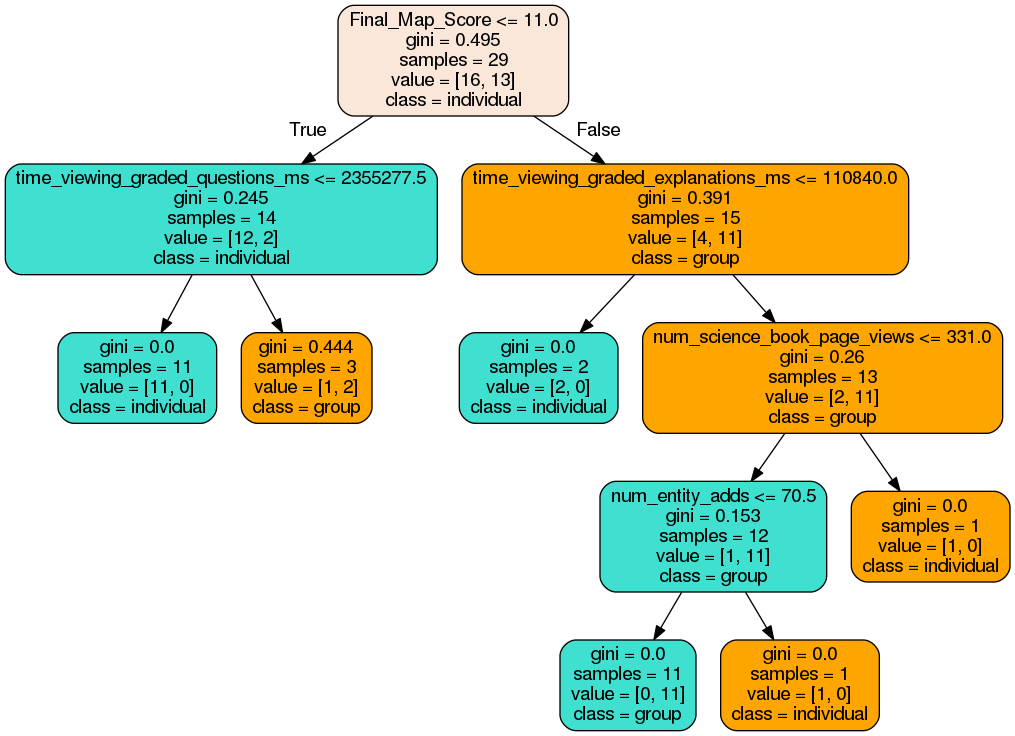
|  |  |
| --- | --- |
| Model | Average Accuracy |
| Entropy W/ Full Dataset | 73% |
| Entropy W/ features from subset 1 | 55% |
| Gini W/ Full Dataset | 82% |
| Gini W/ features from subset 1 | 64% |

Entropy with Full DataSet:

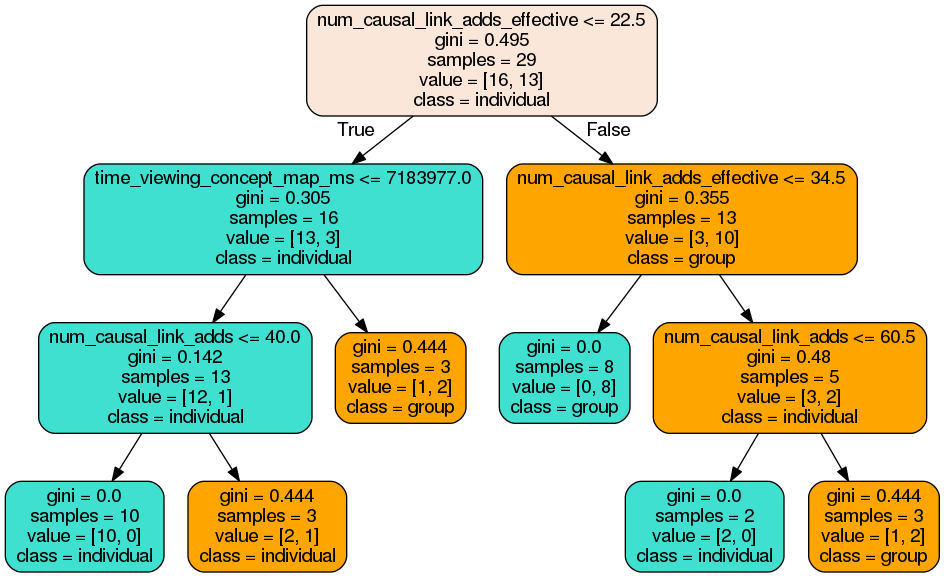
Entropy with subset 1 features:



Gini With Full Dataset



Gini with subset 1 features

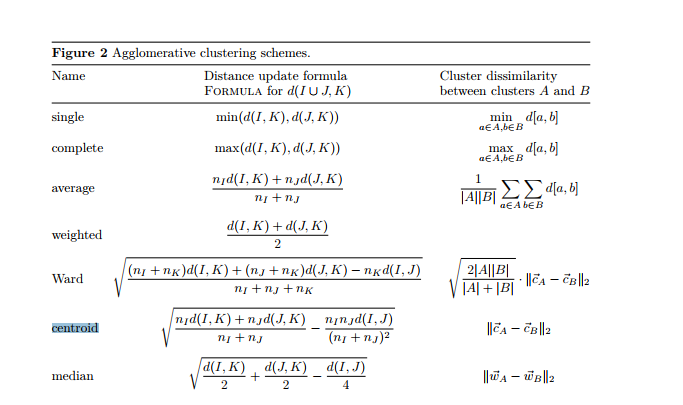


Our results also show that the gini technique not only leads to more accurate trees, but also less complex ones. Individuals are easily distinguished early on, whereas distinguishing the groups require more checking.

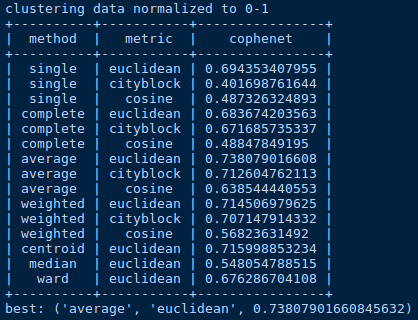
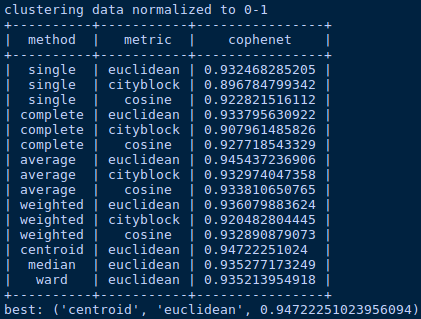
1. **Unsupervised Learning to Discover Structure in the Data**:

When clustering the values, we tried several different methods and metrics. The methods used were:

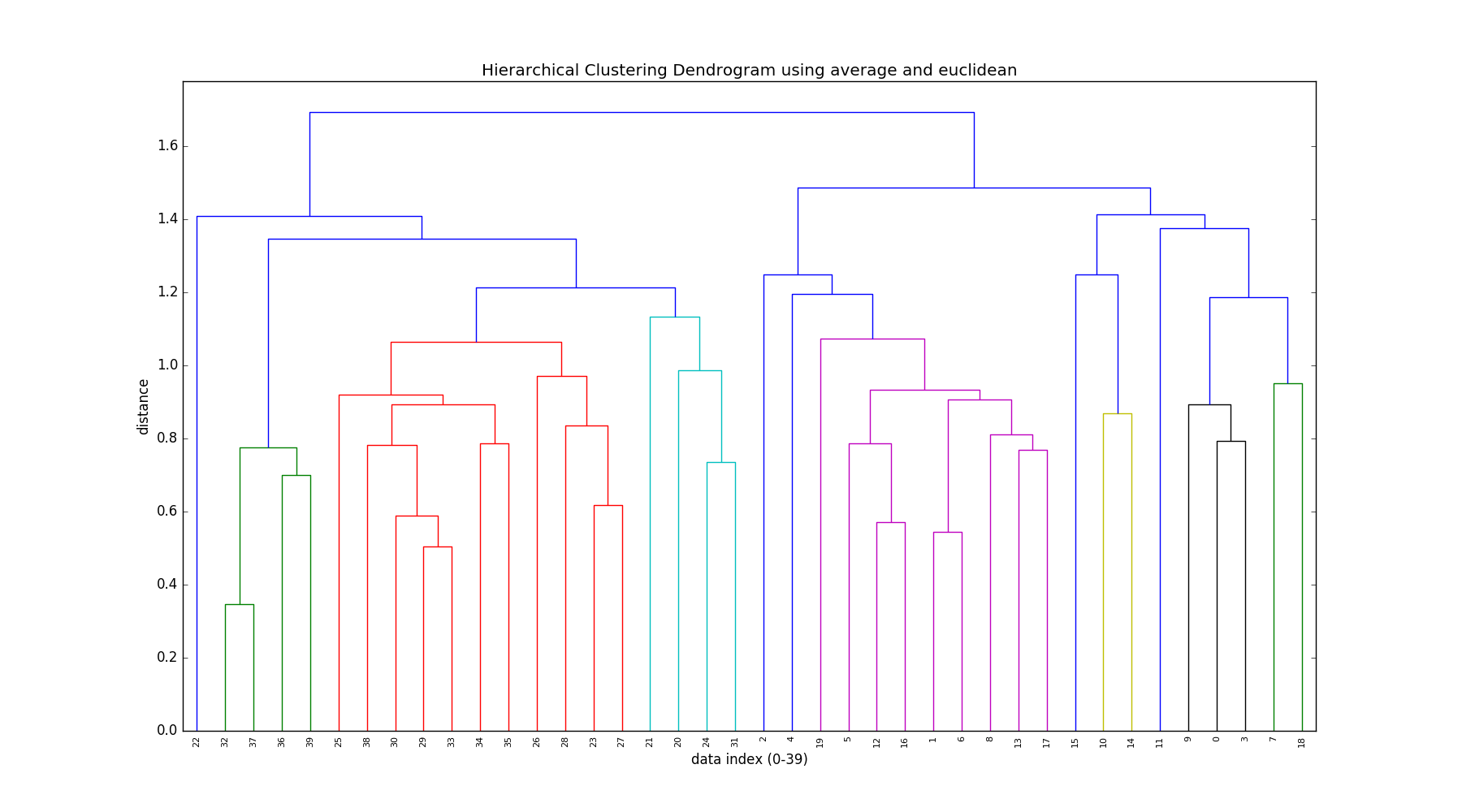
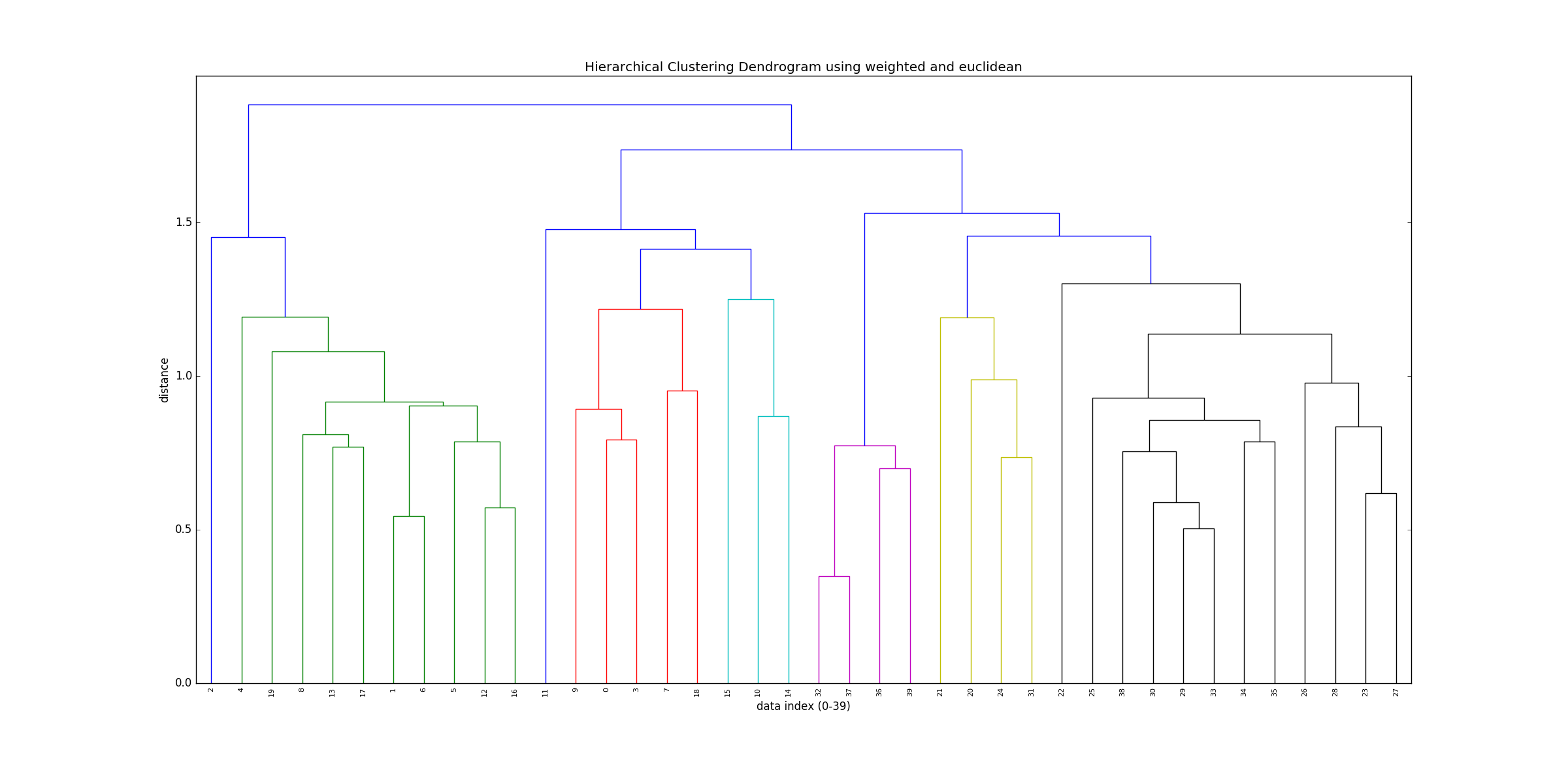
* Single-linkage: agglomerative method which combines two clusters that contain the closest two pairs in each step
* Complete: Agglomerative nearest-neighbor-chain method which combines the two clusters that have the closest ‘farthest neighbors’, i.e. the pair of samples in the two clusters that are the furthest apart
* Average: Agglomerative nearest-neighbor-chain method that averages the distances for each pair in two clusters to determine the distance between them
* Weighted: Agglomerative nearest-neighbor-chain method in which the distance is a weighted average based on the how representative each pair is for each cluster.
* Ward: Ward’s minimum variance nearest-neighbor-chain method
* Centroid: The distance between the cluster centroids is used.
* Median: The median pairs for the two clusters are used when determining dissimilarity

The specific details of the methods are shown in the following figure:

We also tested various distance measures including Euclidean, Manhattan (city block), and cosine. We measured the overall performance of the methods using their cophenet values, which measures how faithfully the clustering represents the dissimilarities between observations. The results are shown in the table below:

 subject\_id range (0,1) subject\_id range (-1, 1)

This shows that for this dataset the average, weighted, and centroid methods using the Euclidean distance metric split the samples into the best groups according to their dissimilarity. While the average method has the best cophenet by a small margin, we found that the clusters and dendrogram created by the weighted method were easier to parse and understand (due to the former having more groups that had less defining characteristics), so they are presented. Below are the dendrograms created by the average and weighted methods using the Euclidean metric for trials ran with subject\_id of either 0 or 1. It is interesting to note the difference in results when the value for the subject id is either -1 or 1. We believe this has to do with the rest of the dataset being restricted to 0-1, and thus the feature with -1 or 1 hold a higher weight. This dendrogram was not saved but all the methods essentially clustered around this feature primarily. The discussion will focus on the weighted method (first dendrogram), which created 6 optimal clusters of samples.



The groups are as follows:

|  |  |
| --- | --- |
| Clusters | Members |
| 1 | A03 A05 A20 A09 A14 A18 A02 A07 A06 A13 A17 |
| 2 | A12 A10 A01 A04 A08 A19 |
| 3 | A16 A11 A15 |
| 4 | 32 37 36 39 |
| 5 | G02 G01 G05 G12 |
| 6 | G03 G06 38 G11 G10 G13 G14 G15 G07 G09 G04 G08 |

There are many ways that these clusters can be described. The primary split that can be seen is that the clusters are firmly divided into clusters of individuals (clusters 1, 2 and 3) and clusters of groups (clusters 4, 5, and 6).

Regarding performance, clusters 2, 4, and 5 appear to contain the highest performing samples, with cluster 4 containing students with perfect final map scores. Cluster 1 low performing students, with all the negative final map scores contained within and a max score of only 9. The other clusters seem fairly mix in performance, indicating that other features describe them more.

Across all of the groups the “time-based” features seem to be fairly inconsistent within clusters, implying that they are not very important when differentiating samples.

A more detailed description of each cluster is below:

* Cluster 1
  + This is a cluster of all individual students that performed fairly poorly regarding final\_map\_score. They tended to make less effective causal links than other groups, and less overall causal links as well. They tended to have much fewer map moves than other individuals.
* Cluster 2
  + This is a cluster of individuals that performed well compared to other individual samples. They had higher causal links (both effective and total) than other individual samples.
* Cluster 3
  + This is a cluster of individuals that had medium performance. They are very similar to the second cluster, but with lower overall final\_map\_scores. This shows that they were making “better” map changes than the cluster 1 students, but still had lower performance for some reason.
* Cluster 4
  + This is a cluster of groups that performed very well. In fact, all the groups had perfect final\_map\_scores. They had lower total map moves than other groups, which points to them being more efficient with their moves, since their effective causal link additions is high.
* Cluster 5
  + This is a cluster of groups that performed well on average. They tended to have more total map views then the other group clusters, implying that they were more experimental. They also had more graded explanations viewed than other group clusters.
* Cluster 6
  + This is a cluster of groups that had a wide range of performance. This group tended to have lower causal link additions (both total and effective) and various numbers of map moves. This seems to be a catch all group for students that either didn’t perform well in groups.

Although these are excellent results, ultimately more data needs to be collected to generate journal-ready results and models.