**HOMEWORK 2 REPORT**

Furkan Doğan / 2098937

1. **K-Nearest Neighbors Algorithm**
   1. Shortcoming of Euclidean distance is it doesn’t work well in high dimensions and for categorical variables. The drawback of Euclidean distance is that it ignores the similarity between attributes. Each attribute is treated as totally different from all of the attributes.
   2. Because even some data vectors have no attribute values in common, they have a smaller distance than the other pair of data vectors containing the same attribute values.
   3. We can normalize the dataset before using because without normalizing it some of classes are aligned in the direction of axis with small ranges and this leads to incorrect classification.
2. **A screenshot of a cell phone

   Description automatically generatedA screenshot of a cell phone

   Description automatically generatedA screenshot of a cell phone

   Description automatically generatedA screenshot of a cell phone

   Description automatically generatedHierarchical Agglomerative Clustering**

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

**Single-Linkage:** As we can see from above plots, single linkage criterion is suitable for first 3 dataset but not 4th dataset because since it takes the next cluster from min difference it always grows up in one direction. So, there is space between clusters it is okay to use this one. But in the 4th dataset all clusters are almost connected, so single linkage cannot separate that clusters.

**Complete-Linkage:** Since complete linkage works with minimum of most far elements of clusters, this method usually produces tighter clusters than single-linkage, but these tight clusters can end up very close together and we can see this in our plots. It may be suitable for 4th dataset but not for rest because for the first 3 dataset clusters ended up very close to each other’s.

**Average-Linkage:** Except the first dataset average linkage is suitable for given datasets. As we can see from below plots it separated clusters in very good form except the first one because average linkage is not suitable for that dataset since there cluster inside cluster.

**Centroid-Linkage:**  Centroid linkage is suitable for dataset3 and dataset4. Since it takes the center of cluster in each level it is not suitable for dataset1 because there is cluster in cluster in that dataset. Also, not suitable for dataset2 because when center is taken even the particles are closer the other cluster it doesn’t matter.

1. **Decision Trees**

Node Count for Information Gain is: 579

Accuracy for Information Gain is: 93.92%

Node Count for Gain Ratio is: 586

Accuracy for Gain Ratio is: 94.24%

Node Count for Average Gini Index is: 608

Accuracy for Average Gini Index is: 93.6%

Node Count for pre-pruned Average Gini Index is: 74

Accuracy for Average Gini Index is: 91.04%

Node Count for post-pruned Average Gini Index is: 47

Accuracy for Average Gini Index is: 89.6%

As we can see with above results Gain Ratio has better results than Information Gain because since our dataset have multi-valued attributes Information Gain bias multi-valued attributes and Gain Ratio reduces this bias. So, Gain ratio is more reliable than Information Gain.

On the other hand, Average Gini Index could be alternative of Information Ratio which uses Gini Index instead of entropy. And different than Information Gain Gini index is minimized instead of maximizing Gini gain.

For pruning’s while node counts are reduced significantly accuracy just decreased slightly. And in the pre.png, post.png files you can see the resulting pre-pruned and post-pruned trees.

For the pre-pruning chi-square test used and calculated dependency of nodes with estimated and observed values and it reduces number of nodes significantly.

A picture containing table

Description automatically generatedA screenshot of a cell phone

Description automatically generatedFor error reduced post-prune I replaced the subtrees with a node when accuracy of spitted train data is getting higher or same. As you can see in below snippets of tree taken from pre.png and post.png for every possibility of parents output is always spec\_prior so we can cut the parents sub-tree and add one output instead of it.

Figure Post-Pruned Tree

Figure Pre-Pruned Tree