Clickstream Project

Submitted by –

Mahathi Priya Appini (110162023),

Shriya Gupta (110168559)

ID3 Algorithm –

1. A ← the "best" decision attribute for next node
2. Assign A as decision attribute for node
3. For each value of A create new descendant node
4. Sort training examples to leaf node according to the attribute value of the branch
5. If all training examples are perfectly classified (same value of target attribute)
   STOP, else iterate over new leaf nodes.

Pseudocode for ID3 Implementation -

```
function ID3 (F: set of attributes, C: the class attribute, S: a training set)
  begin
         If S is empty, return a single node with value Failure;
         If S consists of records all with the same value for
           the class attribute,
           return a single node with that value;
         If F is empty, then return a single node with as value
           the most frequent of the values of the categorical attribute
           that are found in records of S; [note - accuracy will be low];
         Let D be the attribute with largest Gain(D,S)
           among attributes in F;
         Let {dj| j=1,2, .., m} be the values of attribute D;
         Let {Sj| j=1,2, .., m} be the subsets of S consisting
           respectively of records with value dj for attribute D;
         Return a tree with root labeled D and arcs labeled
           d1, d2, .., dm going respectively to the trees

             ID3(R-{D}, C, S1), ID3(R-{D}, C, S2), .., ID3(R-{D}, C, Sm);
  end ID3;
```

where

function Gain (X: attribute after which value is to be obtained; T: element to be identified)

Gain(X,T) = Info(T) - Info(X,T)

where

Info (T) = $-(p1*\log_2(p1) + p2*\log_2(p2) + .. + pn*\log_2(pn))$

Info (X,T) = Sum for i from 1 to n of $(|T_i|/|T|)*\text{Info}\ (T_i)$

(Here, n= 2, p1 = ratio of positives, p2 = ratio of negatives – interchangeable)

This is difference of entropy in moving from parent node to child node after split is applied on given attribute.

For our calculations, we have used split on mid value of range of attribute values

i.e.

Split criteria = Mid = (Min(|T|) + Max(|T|))/2

Observations

- ID3 approach is greedy – solution may be local maxima but may not be optimum. This can be improved by introducing backtracking to the tree.
- P value is inversely proportional to the Chi-square Threshold. Therefore, size and accuracy of the tree is directly proportional to the threshold value.
- The highest accuracy of 74.068% was observed for Chi Threshold of 1.

Output:

- Chi-Squared Stopping Criterion Threshold – 0.01:

  ```
  C:\Python34\python.exe ClickStreamTree.py 0.05 C: /trainfeat.csv C:/ trainlabs.csv C:/ testfeat.csv C:/testlabs.csv

  Percentage Accuracy : 36.0
  ```

- Chi-Squared Stopping Criterion Threshold – 0.05:

  C:\Python34\python.exe ClickStreamTree.py 0.05 C: /trainfeat.csv C:/ trainlabs.csv C:/ testfeat.csv C:/testlabs.csv

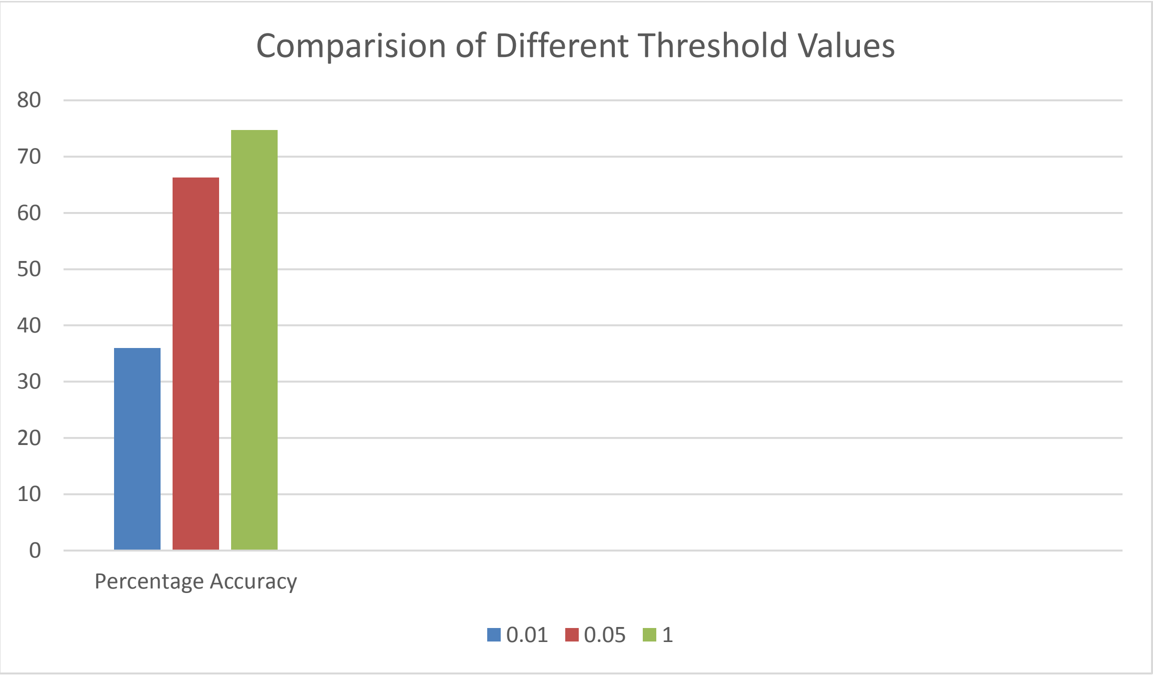
  Percentage Accuracy :66.218

- Chi-Squared Stopping Criterion Threshold – 1.00:

  C:\Python34\python.exe ClickStreamTree.py 1.00 C: /trainfeat.csv C:/ trainlabs.csv C:/ testfeat.csv C:/testlabs.csv

  Percentage Accuracy : 74.868

Comparisons of Accuracy of Different Threshold values



References

- http://web.cs.hacettepe.edu.tr/~ilyas/Courses/BIL712/lec02-DecisionTree.pdf
- http://www.onlamp.com/pub/a/python/2006/02/09/ai_decision_trees.html?page=3
- http://www.had2know.com/academics/chi-square-test-calculator.html
- http://www.cis.temple.edu/~giorgio/cis587/readings/id3-c45.html#3
- http://www.wikipedia.org