ID3 does not guarantee to construct an optimal Decision Tree

Md Abdul Kader
Machine Learning Problem

• Given example Dataset.
• Learn a system (i.e., Decision Tree) to predict unknown instance.
### Example Data Set (Animal)

<table>
<thead>
<tr>
<th>Has Leg</th>
<th>Lives underground</th>
<th>Prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td>YES</td>
<td>NO</td>
<td>Cat</td>
</tr>
<tr>
<td>NO</td>
<td>YES</td>
<td>Earthworm</td>
</tr>
<tr>
<td>NO</td>
<td>NO</td>
<td>Snake</td>
</tr>
</tbody>
</table>
Decision tree

Does it have legs?

Yes → It is cat

No → Does it live underground?

Yes → It is an earthworm.

No → It is snake
Big Question

• How can we construct the Decision Tree from dataset so that we can predict any instance (vector of attributes value) as quickly as possible?

• **Answer:** we need optimal decision tree (DT).
Cost of a Tree

• Sum of all length of path.
• PL=Path Length.

• Sum(PL_i)=5
Optimal Decision Tree (DT)?

• Generated DT from dataset that gives minimum cost.

• Selecting the best DT from all possible trees.

• **BUT** finding the minimum cost (best) tree is \(NP-Complete\) [1] problem. [Proved in May 1976]
ID3?

• ID3 is an algorithm invented by Ross Quinlan used to generate a decision tree from a dataset.
• It uses good Heuristic (Information gain) to construct DT with less cost.
• Iterative Dichotomiser 3
• Typically used in the machine learning and natural language processing domains.
Advantages of ID3

• It uses greedy approach. So it is really fast.
• It constructs optimal Decision Tree for more than 50% of all datasets.
• [According to my experiments].
Goals of my project

• *First goal*: To show that ID3 does not always construct an optimal DT.

• *Second goal*: How frequent does it produce optimal Decision Tree?
Experiment

• I wrote three programs
  – Calculate the cost of generated DT by ID3
  – Calculate the cost of optimal DT
    • Generating all possible trees and selecting best one.
  – Random Dataset generator for small number of attributes
Calculation of cost of DT by ID3

```cpp
ii ID3length(vii db, int mask){
    if(isAllSameClass(db) || mask==-(1<<((C-1)))){
        return ii(0,1); //sum, no of leaf
    }
    //cout<<mask<<endl;
    double mg=-1,t;
    int at=0;
    for(int i=0;i<C-1;i++) // for each attributes;
        if(!(mask&(1<<i))){
            t=IG(db,i);
            if(t>mg){
                mg=t; at=i;
            }
        }
    //cout<<at<<" .";
    ii a,b;
    a=ID3length(project(db,at,0),mask|(1<<at)); a.first+=a.second;
    b=ID3length(project(db,at,1),mask|(1<<at)); b.first+=b.second;
    ii r;
    r.first=a.first+b.first;
    r.second=a.second+b.second;
    return r;
}
```
Calculation of cost of optimal DT

```c
ii minCostTree(vii db, int mask, int lb){

  if(isAllSameClass(db)|| mask==-(1<<(C-1))){
    return ii(0,1); //sum of len , no of leaf
  }
  //showDB(db);
  int na=db[0].size()-1;
  int att=-1;
  ii a,b,r,mn(INF,0):
  for(int i=0;i<na;i++)
    if(!(mask&(1<<i))){
      a=minCostTree(project(db,i,0),mask|(1<<i),lb+1);   a.first+=a.second;
      b=minCostTree(project(db,i,1),mask|(1<<i),lb+1);   b.first+=b.second;
      r.first=a.first+b.first;
      r.second=a.second+b.second;
      if(mn.first>r.first){
        att=i;
      }
      mn=min(r,mn);
    }
  return mn;
}
```
Randomly generated dataset

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>+</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>+</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>+</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>+</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>+</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>+</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-</td>
</tr>
</tbody>
</table>
Generated DT by ID3

Cost = Total length of path = 25
Generated optimal DT

Cost = Total length of path = 12
• Now we know ID3 does not produce the optimal decision tree.
• **BUT** how frequent does it produce optimal decision tree?
<table>
<thead>
<tr>
<th>Number of dataset</th>
<th>ID3 is optimal</th>
<th>ID3 is not optimal</th>
<th>Optimal in %</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>27</td>
<td>23</td>
<td>54%</td>
</tr>
<tr>
<td>50</td>
<td>32</td>
<td>18</td>
<td>64%</td>
</tr>
<tr>
<td>50</td>
<td>33</td>
<td>17</td>
<td>66%</td>
</tr>
<tr>
<td>100</td>
<td>61</td>
<td>39</td>
<td>61%</td>
</tr>
<tr>
<td>100</td>
<td>52</td>
<td>49</td>
<td>52%</td>
</tr>
<tr>
<td>150</td>
<td>83</td>
<td>67</td>
<td>55%</td>
</tr>
</tbody>
</table>

*Note: Number of attributes is less than 10*
Conclusion

• ID3 does not produce optimal decision tree but gives a good approximation.
• For more than 50% of datasets, it produces optimal Decision tree.
References

• [1] Laurent Hyafil, Ronald L. Rivest. “Constructing optimal binary decision trees is NP-complete”
Q&A