

---

**Algorithm 1:** DRABAL for Multi-label Classification

---

**Input:** Data matrix  $D$ , Labels matrix  $L$ , Base Classifier  $C$

**Output:** BNsorted

```
/* Output: Nodes of BNsorted graph holds trained models for
   each label in  $L$  */
1 ActualScores = [[]]; // Initialize an empty matrix to hold
   predicted probability scores
2 BNgraph = learnBayesianNetworkStructure( $L$ ); // Any algorithm for
   learning a proper DAG structure characterizing distribution
   of  $L$  can be used.
3 BNsorted = topologicalSort( BNgraph );
4 for each node  $x \in$  BNsorted do
5   TrainData =  $D$ ; // each step re-initialize TrainData
6   parentlist = getParents( $x$ );
7   if length(parentlist)  $\neq$  0 then
8     for each parent  $p \in$  parentlist do
9       /* Add predicted probability scores stored in parent
          nodes to the original data matrix as an extra
          feature column */
10      TrainData = joinData(TrainData, ActualScores[:, $p.idx$ ])
11    end
12  end
13  clf =  $C$ .fit(TrainData,  $L$ ); // clf can be any trained classifier
14  ActualScores[:, $x.idx$ ] = clf.computeProbs(TrainData); //  $x.idx$  is
   the corresponding index of node  $x$  in matrix  $L$ 
15   $x$ .clf = clf; // store the trained model for node  $x$ 
15 end
```

---