Christopher Morgan
CS478: Brother Christophe
Project 1: Decision Tree

Correctly implement the ID3

- **Some mechanism to handle continuous-valued attributes.**
  My algorithm will simply divide a continuous-valued attribute in two. The attribute is split along the mean of its data values.

- **You should be able to get ~68% predictive accuracy on lenses data with cross-validation.**

	Using Accuracy:
Rep: 0, Fold: 0, Accuracy: 0.5	Rep: 0, Fold: 0, Accuracy: 1
Rep: 0, Fold: 1, Accuracy: 0	Rep: 0, Fold: 1, Accuracy: 0.5
Rep: 0, Fold: 2, Accuracy: 1	Rep: 0, Fold: 2, Accuracy: 1
Rep: 0, Fold: 3, Accuracy: 1	Rep: 0, Fold: 3, Accuracy: 0.5
Rep: 0, Fold: 4, Accuracy: 1	Rep: 0, Fold: 4, Accuracy: 0
Rep: 0, Fold: 5, Accuracy: 0.5	Rep: 0, Fold: 5, Accuracy: 0.5
Rep: 0, Fold: 6, Accuracy: 1	Rep: 0, Fold: 6, Accuracy: 1
Rep: 0, Fold: 7, Accuracy: 0.5	Rep: 0, Fold: 7, Accuracy: 1
Rep: 0, Fold: 8, Accuracy: 1	Rep: 0, Fold: 8, Accuracy: 0.5
Rep: 0, Fold: 9, Accuracy: 0.5	Rep: 0, Fold: 9, Accuracy: 1
Mean predictive accuracy: **0.7**	Mean predictive accuracy: **0.7**

Use your ID3 algorithm on the Iris problem.

Information Gain	Accuracy
Node: Property: petallength	Node: Property: petalwidth
Node: Property: petalwidth	Node: Property: petallength
Node: Class: 0	Node: Class: 0
Node: Property: sepalwidth	Node: Property: sepallength
Node: Class: 1	Node: Property: sepalwidth
Node: Class: 0	Node: Class: 1
Node: Property: petalwidth	Node: Class: 0
Node: Property: sepalwidth	Node: Class: 1
Node: Property: sepallength	Node: Property: petallength
Node: Class: 1	Node: Property: sepallength
Node: Class: 1	Node: Property: sepalwidth
Node: Property: sepallength	Node: Class: 1
Node: Class: 1	Node: Class: 1
Node: Class: 1	Node: Property: sepalwidth
Node: Property: sepalwidth	Node: Class: 1
Node: Class: 2	Node: Class: 1
Node: Property: sepallength	Node: Property: sepallength
Node: Class: 2	Node: Property: sepalwidth
Node: Class: 2	Node: Class: 2
	Node: Class: 2
	Node: Class: 2

- **Compare this tree with the one obtained with information gain as the splitting criterion.**

  The main order of the Entropy tree was to choose petalLength, petalWidth, and then sepalWidth in that order. However, the Accuracy tree favored petalWidth, petalLenght, then sepalLength in that order. Showing that the accuracy algorithm chose petalWidth over petalLength. In addition to the order of attributes chosen, the switch in algorithms also created a slightly larger tree when accuracy was used. The Entropy tree consisted of 19 nodes where the Accuracy tree was made up of 21. There was also a very small increase in the number branching nodes, the information gain tree has 9 branching nodes and the accuracy tree has 10.

- **Evaluate predictive accuracy using 10-fold cross-validation for information gain and accuracy.**

Rep: 0, Fold: 0, Accuracy: 1
Rep: 0, Fold: 1, Accuracy: 1
Rep: 0, Fold: 2, Accuracy: 1
Rep: 0, Fold: 3, Accuracy: 1
Rep: 0, Fold: 4, Accuracy: 0.933333
Rep: 0, Fold: 5, Accuracy: 0.866667
Rep: 0, Fold: 6, Accuracy: 1
Rep: 0, Fold: 7, Accuracy: 0.933333
Rep: 0, Fold: 8, Accuracy: 0.933333
Rep: 0, Fold: 9, Accuracy: 0.933333
Mean predictive accuracy: **0.96**

Using Accuracy:
Rep: 0, Fold: 0, Accuracy: 1
Rep: 0, Fold: 1, Accuracy: 0.866667
Rep: 0, Fold: 2, Accuracy: 1
Rep: 0, Fold: 3, Accuracy: 0.933333
Rep: 0, Fold: 4, Accuracy: 0.866667
Rep: 0, Fold: 5, Accuracy: 1
Rep: 0, Fold: 6, Accuracy: 1
Rep: 0, Fold: 7, Accuracy: 0.933333
Rep: 0, Fold: 8, Accuracy: 1
Rep: 0, Fold: 9, Accuracy: 1
Mean predictive accuracy: **0.96**

- **Compare the results.**
  On average the two different branching criteria appear to create models that perform about the same, with a mean accuracy of .96 over 10 folds on the Iris data set. Using Information Gain the algorithm produced 5 perfect runs where as the Accuracy driven algorithm produced 6. In the end they appear equally effective for learning on the Iris data set.

Repeat the experiment with the Voting problem.

Information Gain	**Accuracy**
Node: Property: 'physician-fee-freeze'	Node: Property: 'physician-fee-freeze'
Node: Property: 'synfuels-corporation-cutback'	Node: Property: 'handicapped-infants'
Node: Property: 'crime'	Node: Property: 'religious-groups-in-schools'
Node: Property: 'anti-satellite-test-ban'	Node: Class: 0
...	Node: P: 'water-project-cost-sharing'
Node: Property: 'anti-satellite-test-ban'	...
Node: Class: 0	Node: Property: 'water-project-cost-sharing'
Node: Property: 'el-salvador-aid'	Node: Property: 'anti-satellite-test-ban'
...	...
	Node: Class: 0
Node: Class: 0	Node: Property: 'handicapped-infants'
Node: Property: 'synfuels-corporation-cutback'	Node: Property: 'water-project-cost-sharing'
Node: Property: 'duty-free-exports'	N: P: 'adoption-of-the-budget-resolution'
N: P: 'adoption-of-the-budget-resolution'	N: P: 'religious-groups-in-schools'
...	...
Node: Property: 'immigration'	Node: Property: 'el-salvador-aid'
...	Node: Class: 1
Node: P: 'adoption-of-the-budget-resolution'	N: P: 'religious-groups-in-schools'
Node: Property: 'el-salvador-aid'	...
Node: Class: 0	Node: Property: 'el-salvador-aid'
Node: Property: 'immigration'	...
Node: P: 'superfund-right-to-sue'	Node: P: 'synfuels-corporation-cutback'
Node: Class: 0	Node: Class: 1

N: P: 'anti-satellite-test-ban' ... Node: Class: 1 Node: Property: 'anti-satellite-test-ban' Node: Class: 0 Node: Class: 1	Node: Property: 'mx-missile' Node: P: 'water-project-cost-sharing' ... Node: P: 'water-project-cost-sharing' ...

- **Compare this tree with the one obtained with information gain as the splitting criterion.**
  The trees differ in attribute selection, number of branches, depth to first classification, and number of nodes. Looking at attribute selection, both trees begin by splitting the data on the physician-fee-freeze attribute, but from there the trees diverge and do not follow the same attribute selection. The information gain tree follows synfuels-corporation-cutback, crime, duty-free-exports, and adoption-of-budget-restoration, where the accuracy tree follows handicapped-infants, religious-groups-in-school, water-project-cost-sharing, and synfuels-corporation-cutback.

  Along this line of divergent attributes, the accuracy driven algorithm requires a lot more branches to classify all of the data. The accuracy tree consists of 68 different attribute selections, or branching points, where as the information gain tree has slightly fewer with 54 different attribute selections. The accuracy tree also took longer to find its first classification. The accuracy tree found its first classification three nodes(attribute selections) deep, where as the information gain tree found its first classification two nodes deep.

  Given that the accuracy tree already had to dive deeper to find its first classification, it is not surprising that the accuracy tree also contained more nodes that the information gain tree. The accuracy tree contained 137 nodes where as the information gained tree consisted of 109 nodes.

- **Evaluate predictive accuracy using 10-fold cross-validation for information gain and accuracy.**

	Using Accuracy:
Rep: 0, Fold: 0, Accuracy: 0.883721	Rep: 0, Fold: 0, Accuracy: 0.930233
Rep: 0, Fold: 1, Accuracy: 0.906977	Rep: 0, Fold: 1, Accuracy: 0.906977
Rep: 0, Fold: 2, Accuracy: 0.976744	Rep: 0, Fold: 2, Accuracy: 0.976744
Rep: 0, Fold: 3, Accuracy: 0.883721	Rep: 0, Fold: 3, Accuracy: 0.906977
Rep: 0, Fold: 4, Accuracy: 0.930233	Rep: 0, Fold: 4, Accuracy: 0.930233
Rep: 0, Fold: 5, Accuracy: 0.906977	Rep: 0, Fold: 5, Accuracy: 0.930233
Rep: 0, Fold: 6, Accuracy: 0.953488	Rep: 0, Fold: 6, Accuracy: 0.930233
Rep: 0, Fold: 7, Accuracy: 0.953488	Rep: 0, Fold: 7, Accuracy: 0.860465
Rep: 0, Fold: 8, Accuracy: 0.930233	Rep: 0, Fold: 8, Accuracy: 0.883721
Rep: 0, Fold: 9, Accuracy: 0.953488	Rep: 0, Fold: 9, Accuracy: 0.930233
Mean predictive accuracy: **0.927907**	Mean predictive accuracy: **0.918605**

- **Compare the results.**
  The information gain model is 0.01 more accurate that the accuracy model. This seems to give backing to the idea that improving accuracy it a lot harder task than simply decreasing entropy.

- **Describe and justify the method you used to handle missing values.**
  For each missing attributes I calculate the majority value for that attribute from the original data set. I then replace the missing value with the majority for that attribute. This seems reasonable given that the majority value, for a missing attribute, is a simple best guess for what it might be given our current data set.

Extend your algorithm so that, when accuracy is the splitting criterion, it may use up to 2 conditions in the tests at each node (e.g., attrX = Vx and attrY = Vy). You may choose to make that an user-specified option.

- **Induce a decision tree using the entire dataset with this extended algorithm for both the Iris problem and the Voting problem. Give a visual representation of the trees.**

Iris

```
Node: Property: sepallength & sepalwidth
    Node: Property: petallength
        Node: Property: petalwidth
            Node:  Class: 0
            Node:  Class: 1
        Node: Property: petalwidth
            Node:  Class: 1
            Node:  Class: 2
    Node:  Class: 0
    Node: Property: petallength & petalwidth
        Node:  Class: 1
        Node:  Class: 2
        Node:  Class: 2
        Node:  Class: 2
    Node: Property: petallength & petalwidth
        Node:  Class: 1
        Node:  Class: 2
        Node:  Class: 2
        Node:  Class: 2
```

Voting

```
Node: Property: 'physician-fee-freeze' & 'superfund-right-to-sue'
    Node: Property: 'handicapped-infants'
        Node: Property: 'water-project-cost-sharing' & 'aid-to-nicaraguan-contras'
            ~Class Nodes
        Node: Property: 'el-salvador-aid' & 'religious-groups-in-schools'
            Node: Property: 'adoption-of-the-budget-resolution' & 'crime'
                ~Class Nodes
            Node:  Class: 0
            Node:  Class: 0
            Node:  Class: 0
    Node: Property: 'handicapped-infants' & 'crime'
        Node: Property: 'education-spending'
            Node:  Class: 0
            Node:  Class: 1
        Node: Property: 'water-project-cost-sharing'
            Node:  Class: 0
            Node: Property: 'adoption-of-the-budget-resolution' & 'mx-missile'
                 ...
                Node: Property: 'el-salvador-aid' & 'synfuels-corporation-cutback'
                    ...
        Node:  Class: 0
        Node:  Class: 0
    Node: Property: 'water-project-cost-sharing' & 'adoption-of-the-budget-resolution'
```

```
    ~Class Nodes
  Node: Property: 'adoption-of-the-budget-resolution' & 'synfuels-corporation-cutback'
    Node: Property: 'water-project-cost-sharing' & 'duty-free-exports'
      ~Class Nodes
    Node: Property: 'anti-satellite-test-ban' & 'education-spending'
      Node: Property: 'handicapped-infants'
        ...
      Node: Property: 'handicapped-infants'
        ...
      Node:  Class: 0
      Node:  Class: 1
    Node: Property: 'anti-satellite-test-ban' & 'immigration'
      ~Class Nodes
    Node: Property: 'water-project-cost-sharing'
      Node:  Class: 1
      Node:  Class: 0
```

- **Compare them with those obtained above.**
  The effects of allowing the model to represent up to two attributes at once has similar effects in both the iris data the voting data with respect to branch nodes and number of nodes. The iris model decreased in number of branching nodes from 10 branch nodes to 6. The voting data model decreased in number of branching nodes from 68 nodes to 35. There was a similar effect to the overall number of nodes, where the number of nodes in the iris model went from 21 nodes to 19. The voting data also showed a decrease in the number of nodes from 137 nodes to 97.

  However, the addition of allowing the model to select up to two attributes at a time caused the order of attributes selected, or path followed, to differ between the two data sets. The iris data showed a flip in path of attributes followed; where as the voting data follow a roughly similar path. The iris data with only one attribute selection followed a path of selecting on petalwidth and petallength, followed by selections on sepallength and sepalwidth. However, the iris data, with two attributes selected, followed the path of sepallength and spealwidth first followed by petalwidth and petallength. However, the voting data models, the ones using single and up to two attribute selection, both roughly followed the attribute path of physician-fee-freeze, handicap-infants, and water-project-cost-sharing.

  All in all it appears that allowing the accuracy-training algorithm to select on multiple attributes helped to decrease the difficulty of the problem, in the sense of decreasing the size of the model trees.

- **Explain why it may be necessary to thus extend the decision tree learning algorithm when using accuracy as the splitting criterion (and why the extension is of little value when information gain is the splitting criterion).**
  It is necessary because splitting using accuracy is a much harder task. It requires finding an attribute to split on that will change the majority of an example set versus a spilt that just reduces the number of instances that differ. Thus, by giving the algorithm more attributes to select from, it makes it easier to find a split that finds a meaningful change in the majority of a set of data. However, a entropy driven algorithm only seeks to change the distribution of a data set along an attribute split and it is likely that just the act of splitting the data set alone will be sufficient to cause a change in entropy, therefore, splitting along multiple attributes with entropy wont help the algorithm much, if any.