605.649 — Introduction to Machine Learning

Programming Project #3

The purpose of this assignment is to give you a chance to get some hands-on experience learning decision trees for classification and regression. This time around, we are not going to use anything from the module on rule induction; however, you might want to examine the rules learned for your trees to see if they "make sense." Specifically, you will be implementing a standard univariate (i.e., axis-parallel) decision tree and will compare the performance of the trees when grown to completion on trees that use either early stopping (for regression trees) or reduced error pruning (for classification trees).

Let's talk about the numeric attributes. There are two ways of handling them. The first involves discretizing (binning), similar to what you were doing in earlier assignments. This is *not the preferred approach*, so we ask that you avoid binning these attributes. Instead, the second and preferred approach is to sort the data on the attribute and consider possible binary splits at midpoints between adjacent data points. Note that this could lead to a lot of possible splits. One way to reduce that is to consider midpoints between data where the class changes. For regression, there is no corresponding method, so you should consider splits near the middle of the sorted range and not consider all possible.

For decision trees, it should not matter whether you have categorical or numeric attributes, but you need to remember to keep track of which is which. In addition, you need to implement that gain-ratio criterion for splitting in your classification trees. Go ahead and eliminate features that act as unique identifiers of the data points.

For this assignment, you will use three classification datasets and three regression data sets that you will download from the UCI Machine Learning Repository or from Blackboard, namely:

1. Breast Cancer [Classification]

   `https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Original%29`

   This breast cancer databases was obtained from the University of Wisconsin

2. Car Evaluation [Classification]

   `https://archive.ics.uci.edu/ml/datasets/Car+Evaluation`

   The data is on evaluations of car acceptability based on price, comfort, and technical specifications.

3. Image Segmentation [Classification]

   `https://archive.ics.uci.edu/ml/datasets/Image+Segmentation`

   The instances were drawn randomly from a database of 7 outdoor images. The images were hand segmented to create a classification for every pixel.

4. Abalone [Regression]

   `https://archive.ics.uci.edu/ml/datasets/Abalone`

   Predicting the age of abalone from physical measurements.

5. Computer Hardware [Regression]

   `https://archive.ics.uci.edu/ml/datasets/Computer+Hardware`

   The estimated relative performance values were estimated by the authors using a linear regression method. The gives you a chance to see how well you can replicate the results with these two models.

6. Forest Fires [Regression]

   `https://archive.ics.uci.edu/ml/datasets/Forest+Fires`

   This is a difficult regression task, where the aim is to predict the burned area of forest fires, in the northeast region of Portugal, by using meteorological and other data .

For this project, the following steps are required:

- Download the six (6) data sets from the UCI Machine Learning repository. You can find this repository at `http://archive.ics.uci.edu/ml/`. The data sets are also available in Blackboard. All of the specific URLs are also provided above.

- Implement the ID3 algorithm for classification decision trees using gain-ratio as the splitting criterion.

- Implement reduced-error pruning to be used as an option with your implementation of ID3.

- Run your ID3 implementation on each of the three classification data sets, comparing both pruned and unpruned versions of the trees. These runs should be done with 5-fold cross-validation so you can compare your results statistically. You should pull out 10% of the data to be used as a validation set (similar to what we did for tuning) and then use the remaining 90% for cross validation. You should use classification error for your loss function.

- Implement the CART algorithm for regression decision trees using mean squared error as the splitting criterion.

- Incorporate a cut-off threshold for early stopping. If the threshold is set to zero, this should indicate no early stopping.

- Run your CART implementation on each of the three regression data sets, comparing both full and stopped versions of the trees. You will need to tune the stopping threshold and should use the same procedure for extracting a validation set to serve as your tuning set. The runs should be done with 5-fold cross-validation so you can compare your results statistically. You should use mean squared error for your loss function.

- Write a very brief paper that incorporates the following elements, summarizing the results of your experiments. Your paper is required to be at least 5 pages and no more than 10 pages using the JMLR format You can find templates for this format at `http://www.jmlr.org/format/format.html`. The format is also available within Overleaf.

  1. Title and author name
  2. Problem statement, including hypothesis, projecting how you expect each algorithm to perform
  3. Brief description of your experimental approach, including any assumptions made with your algorithms
  4. Presentation of the results of your experiments
  5. A discussion of the behavior of your algorithms, combined with any conclusions you can draw
  6. Summary
  7. References (Only required if you use a resource other than the course content.)

- Submit your fully documented code, the outputs from running your programs, and your paper.

- For the video, the following constitute minimal requirements that must be satisfied:

  - The video is to be no longer than 5 minutes long.
  - The video should be provided in mp4 format. Alternatively, it can be uploaded to a streaming service such as YouTube with a link provided.
  - Fast forwarding is permitted through long computational cycles. Fast forwarding is *not permitted* whenever there is a voice-over or when results are being presented.
  - Be sure to provide verbal commentary or explanation on all of the elements you are demonstrating.
  - Provide sample outputs from one test set on one fold for a classification tree and a regression tree.
  - Show a sample classification tree without pruning and with pruning as well as a sample regression tree without early stopping and with early stopping.
  - Demonstrate the calculation of information gain, gain ratio, and mean squared error.

  - Demonstrate a decision being made to prune a subtree (pruning) and a decision being made to stop growing a subtree (early stopping).
  - Demonstrate an example traversing a classification tree and a class label being assigned at the leaf.
  - Demonstrate an example traversing a regression tree and a prediction being made at the leaf.
  - Show the average performance over the five folds on a classification data set (with and without pruning) and on a regression data set (with and without early stopping).

Your grade will be broken down as follows:

- Code structure – 10%
- Code documentation/commenting – 10%
- Proper functioning of your code, as illustrated by a 5 minute video – 30%
- Summary paper – 50%