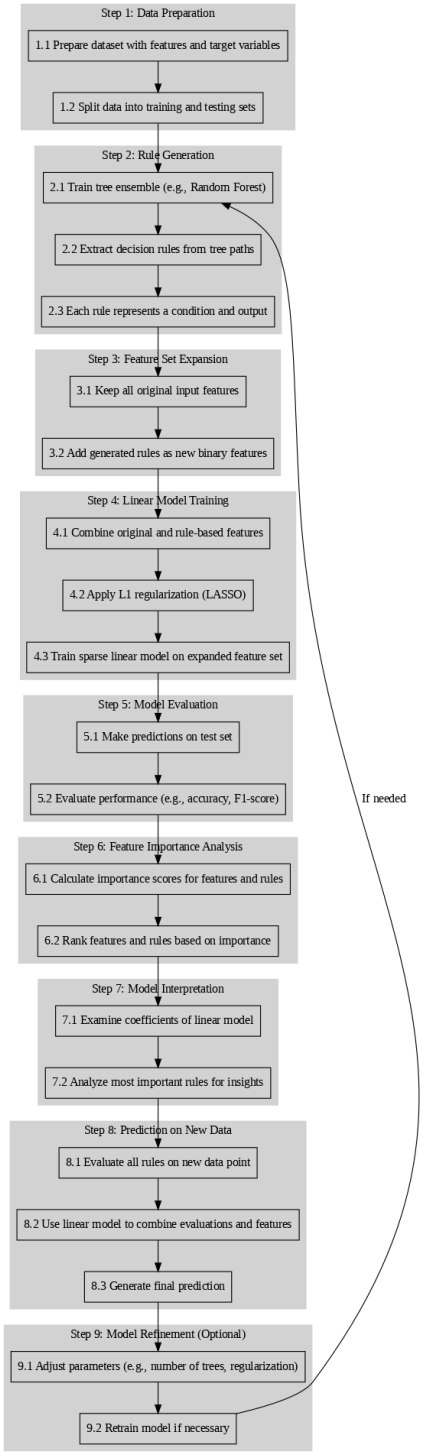
Algorithms Explanation - Vihaan Nama

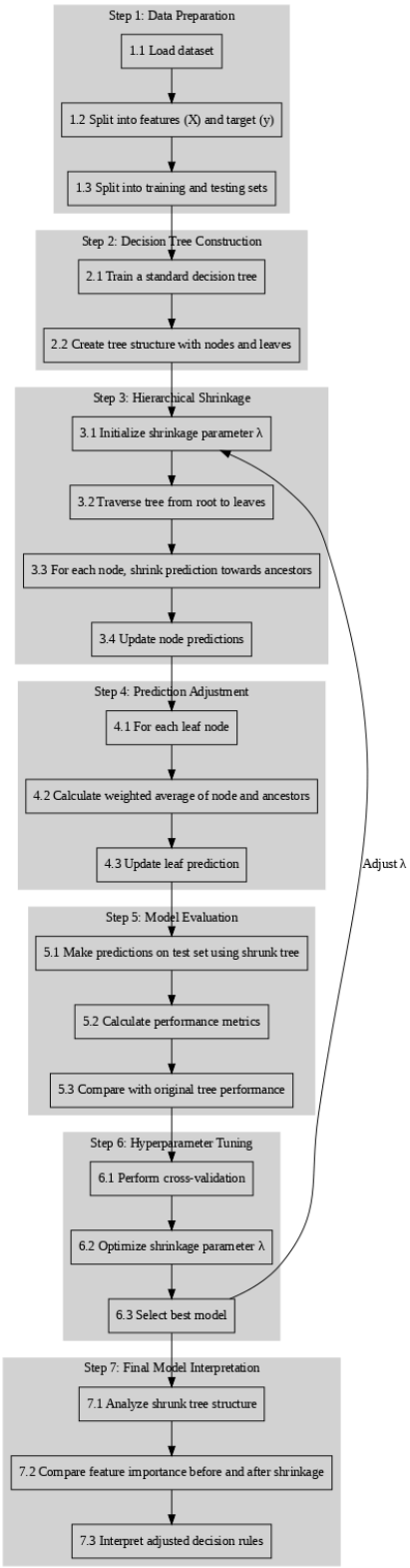
1) Rule Fit Algorithm



Explanation -

- First you pre-process your data then split it into train and test sets.
- Next, you train an ensemble method such as the Random Forest Classifier.
- Now you extract the decision rules from the tree paths.
- Next we start expanding the feature set. We do this by - keeping the original input features and adding the binary rules as new features - this means we create each rule as a column in our data set and add 0 or 1 for the data points that conform with this rule.
- Once we add all the rules and the input features as features we now perform LASSO regression on the model so we can only keep the rules that are necessary - the features which don't add any benefit are given a coefficient of 0.
- Now we train any sparse linear model on the LASSO regressed expanded feature set - and make predictions on the test set - we obtain precision, recall, F1 and other metrics.
- Calculate the importance scores for the features and rules based on various metrics like the gini index -> the most important rules are then analysed for insights .
- Evaluate the rules on new data that was not yet fed to the model.
- Then we generate the final predictions and adjust the parameters if necessary.
- Retrain the model if it is not performing accurately.

2) Hierarchical Shrinkage applied on decision tree

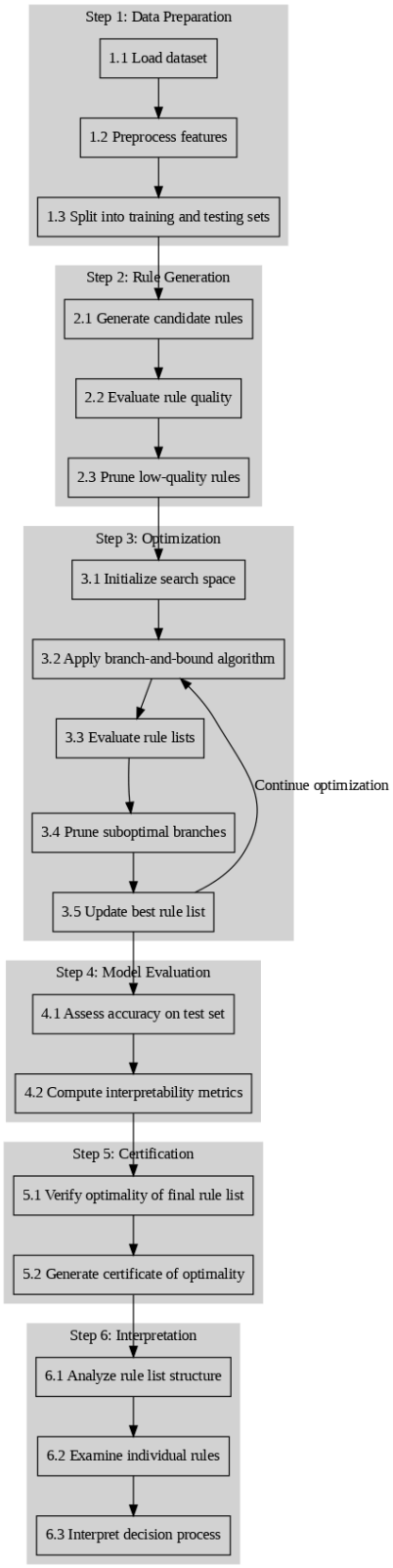


Explanation -

- In the beginning you go about the work normally -> split the data into train and test and then create a standard decision tree.
- We now initialise a shrinking parameter lambda and start traversing the tree from the root to leaves.
- For each node we shrink the predictions towards the ancestors - this means -
    - Ancestors: In a decision tree, the ancestors of a node are all the nodes above it in the path from the root to that node. The root node is the ancestor of all other nodes.
    - Sample means: At each node, there's a sample mean of the target variable for all the data points that reach that node.
    - Shrinking towards: This means adjusting the prediction at a node to be a weighted average between its own prediction and the predictions of its ancestors.
    - The process:
        - Start with the prediction at a leaf node (usually the mean of the target variable for samples in that leaf).
        - Adjust this prediction by combining it with the sample means of its parent node, grandparent node, and so on, up to the root.
        - The amount of influence each ancestor has depends on the number of samples at that ancestor and a regularization parameter.
    - Purpose: This shrinkage helps to regularize the model by incorporating more global information (from higher up in the tree) into the local predictions (at the leaves). It can help reduce overfitting, especially in areas of the feature space with few samples.
    - Effect: Predictions become "smoother" and less prone to abrupt changes between neighboring leaves. This often leads to better generalization, especially for smaller datasets or deep trees.

- For each leaf node we calculate the weighted average of the node and its ancestors and update the leaf prediction respectively.
- After this we perform cross validation and anlyse the shrinking parameter lambda -> then select the best model

3) Optimal Rule list Classifier



Explanation -

- We load the dataset and perform a train test split
- Next we generate candidate rules, evaluate the rule quality, and then prune the low quality rules.
- Now we start with the process of optimization -
    - We initialise a search space
    - Apply a common branch and bound algorithm
    - Evaluate Rule lists
    - Prune sub optimal branches
    - Update the best rule fit

  This is done in repeat until no more optimization is required.
- Now we perform various metric tests to check if the model is working correctly
- We now finalize the optimal rule set and generate a certificate of optimality