Reference: CS5228: Assignment

Decision Tree Regressor:

Input: Training Dataset D;

Output: A regression decision tree $f(x)$.

In the input space of the training dataset, each region is recursively divided into two sub-regions and the output values on each sub-region are determined to construct a binary decision tree.

Steps:

1. Select the optimal splitting variable j and the splitting threshold s to solve

$$\min_{j,s}\left[\min_{c_1}\sum_{x_i\in R_1(j,s)}(y_i-c_1)^2+\min_{c_2}\sum_{x_i\in R_2(j,s)}(y_i-c_2)^2\right]$$

   Traverse the splitting variable j and scan the splitting threshold s for the fixed splitting variable j, and determine the pair (j,s) that minimizes the expression. If we collect the values of variable j from all training sample and denote it as V_j, then the splitting threshold s is one element of V_j. For this assignment, if you find that every value in $[a_1,\ a_2)$ (where a_1 and a_2 are from V_j) can be a splitting threshold with the same minimum impurity, then a_1 should be selected as the threshold. You should not choose other thresholds, e.g., $(a_1+a_2)/2$. This is requirement is for automatic grading.

2. Split the region with the selected pair (j,s) and determine the corresponding output value:

$$\mathrm{R}_1(j,s)=\{x|x^{(j)}\le s\},\mathrm{R}_2(j,s)=\{x|x^{(j)}>s\}$$

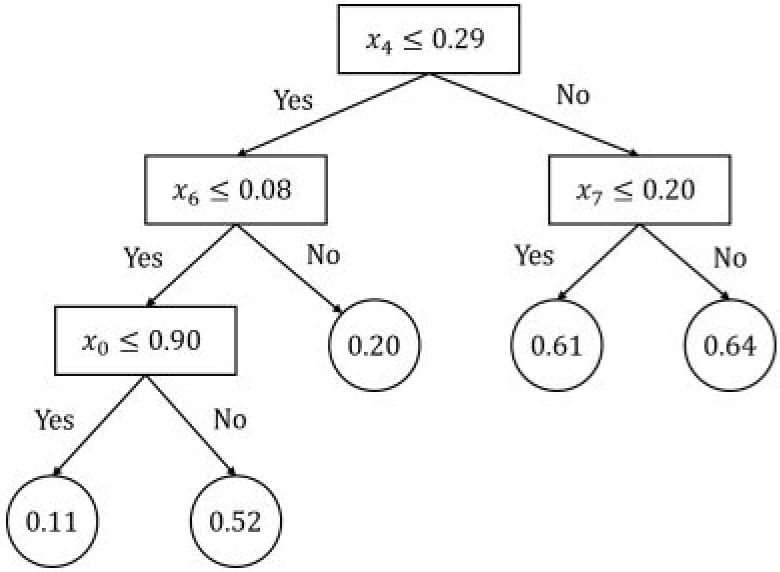
$$\hat{c}_m=\frac{1}{N_m}\sum_{x_i\in R_m}y_i,\qquad x\in R_m,\qquad m=1,2$$

3. Repeat the above two steps considering each resulting region as a parent node until the maximum depth of the tree is obtained.

4. The input space is divided into M regions $\mathrm{R}_1,R_2,\dots,R_M$, then the decision tree is

$$f(x)=\sum_{m=1}^{M}\hat{c}_m I(x\in R_m)$$

For example,



the above decision tree can be represented by

```
self.root = {"splitting_variable": 4,
             "splitting_threshold": 0.29,
             "left": {"splitting_variable": 6,
                      "splitting_threshold": 0.08,
                      "left": {"splitting_variable": 0,
                               "splitting_threshold": 0.90,
                               "left": 0.11,
                               "right": 0.52},
                      "right":0.20}
             "right": {"splitting_variable": 7,
                       "splitting_threshold": 0.20,
                       "left": 0.61,
                       "right": 0.64}
             }
```

Gradient Boosting Regressor:

In this task, the loss function is defined as

$$L(y, f(x)) = \frac{1}{2}(y - f(x))^2$$

Then, the negative gradient of $L(y, f(x))$ is

$$-\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)} = y_i - f(x_i)$$

The algorithm of Gradient Boosting Algorithm is summarized as follows:

Input: Training dataset $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}, x_i \in \mathcal{X} \in \mathbf{R}^n, y_i \in \mathcal{Y} \in \mathbf{R}^n$; Learning rate lr; Loss function $L(y, f(x))$.

Output: A regression decision tree $\hat{f}(x)$.

Steps:

1. Initialize $f_0(x) = argmin_c \sum_{i=1}^{N} L(y_i, c)$.
2. For $m = 1$ to M:
   a. For $i = 1,2, \dots, N$, compute the residual
   $$\gamma_{\mathrm{im}} = -\left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f(x)=f_{m-1}(x)}$$
   b. Fit a regression tree to the targets γ_{im} resulting in terminal regions $R_{mj}, j = 1,2, \dots, J_m$.
   c. For $j = 1,2, \dots, J_m$, compute
   $$c_{mj} = lr \times argmin_c \sum_{x_i \in R_{mj}} L(y_i, f_{m-1}(x_i) + c)$$
   d. Update $\mathrm{f_m}(x) = f_{m-1}(x) + \sum_{j=1}^{J_m} \mathrm{c}_{mj} I(x \in R_{mj})$
3. The regression tree is $\hat{f}(x) = \mathrm{f_M}(x) = f_0(x) + \sum_{m=1}^{M} \sum_{j=1}^{J_m} \mathrm{c}_{mj} I(x \in R_{mj})$