You are to implement decision stumps (DS) and boosted decision stumps (BDS) for regression. Decision stumps are decision trees with one split. For further details, see below. Your programs should be written in R. You should apply your DS and BDS programs to the Boston data set to predict medv given lstat and rm. (In other words, medv is the label and lstat and rm are the attributes). There is no need to normalize the attributes, of course. Split the data set randomly into two equal parts, which will serve as the training set and the test set. Use your birthday (in the format MMDD) as the seed for the pseudorandom number generator.

1. **Train your DS implementation on the training set. Find the MSE on the test set.**

The Decision Stumps algorithm for regression is implemented in DS.R for the two attributes as mentioned in the function call.

The MSE on the test set obtained is 71.09.

Output:

```
[1] "The attribute with min RSS is:" "lstat"

[1] "Test MSE for Decision Stump:" " 71.0925722755505"
```

2. **Train your BDS implementation on the training set for learning rate η = 0.01 and B = 1000 trees. Find the MSE on the test set.**

The Boosted Decision Stumps algorithm for regression is implemented in BDS.R for the two attributes as mentioned in the function call.

The test MSE was calculated using min(RSS) of the

The MSE on the test set with learning rate, eta = 0.01 and number of trees, B = 1000 obtained is 140.256.

Output:

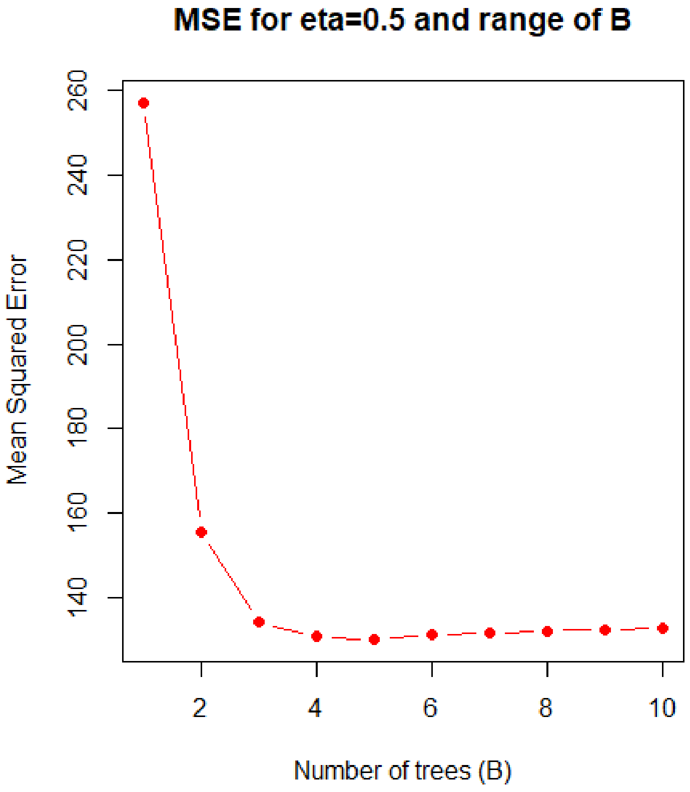
```
[1] "Test MSE for Boosted Decision Stump:"

[2] "140.255909419316"
```

3. **Plot the test MSE for a fixed value of η as a function of B ∈ [1, B0] (the number of trees) for as large B0 as possible. Do you observe overfitting?**

The test MSE was plotted for a fixed learning rate, eta = 0.5 and number of trees as in the below table and also depicted as a plot below.

Sl.no	Test MSE	eta(learning rate)	No. of tress (B)
1	257.1923	0.50	1
2	155.5552	0.50	2
3	134.2517	0.50	3
4	130.7857	0.50	4
5	130.0865	0.50	5
6	131.1882	0.50	6
7	131.5942	0.50	7
8	132.2185	0.50	8
9	132.4539	0.50	9
10	132.7147	0.50	10



In this task, the number of trees ranges from 1 to 10. The model with just 1 tree has a very large MSE and for B=2, the MSE decreases with a big margin.

We can observe that for just 10 observations, the MSE has lot of variation. After constructing a particular number of trees, the MSE tends to increase significantly.

General Observation:

In this boosted decision tree model, overfitting occurs as the model tends to reduce the training error at the cost of the increased test set error.

The test MSE increases after certain B (number of trees).

It is computationally complex when compared to other supervised learning algorithms.