

	Model1	Model2	Model3	WeightAveragePrediction
Weight	0.4	0.3	0.3	
Prediction	45	40	60	48

Model1	Model2	Model3	AveragePrediction
45	40	65	50

Model1	Model2	Model3	VotingPrediction
1	0	1	1

Data	Weights
1	w1
2	w2
3	w3
4	w4
5	w5
6	w6
7	w7
8	w8
9	w9
10	w10
11	w11
12	w12
13	w13
14	w14
15	w15

All weights are equal

Sample Data1
1
2
3
5
6
7
10

Train Model 1

Predict on all the data & generate output

Classification Result
How many wrong classified

1
4
6
Weak Learner

Given Test Data

0

Weights
w1
w2
w3
w4
w5
w6
w7
w8
w9
w10
w11
w12
w13
w14
w15

Make high weights

Make high weights

Make high weights

Sample Data2
1
4
6
10
11
12
15

Train Model 2

Predict on all the data & generate output

Classification Result
How many wrong classified

2
8
14
Weak Learner

Given Test Data

1

Updating Weight- Make high weights

Weights
w1
w2
w3
w4
w5
w6
w7
w8
w9
w10
w11
w12
w13
w14
w15

Make high weights

Make high weights

Make high weights

Make high weights

Make high weights

Make high weights

Make high weights

Randomly Sampling

Sample Data..n
2
8
9
10
12
13
14

Train Model..n

Predict on all the data & generate output

Classification Result
How many wrong classified

3
5
15
Weak Learner

Given Test Data

1

Boosting Technique

This iteration is performed untile all misclassification convert into correct classification

Why make high weights due to they will get select in the next prediction
Probability of getting selected high
Updating Weight- Make high

This process is run till min (as much as it can be possible)

Using majority vote & consider 1
This is a Strong Learner

Testing the model
Prediction