

AdaBoost

→ Boosting technique

→ 3 points:

(i) It combines weak learners to make classifications.
The weak learners are almost always '1-layer deep' decision trees known as stumps.

(ii) Some stumps get more say than others.

(iii) Each stump is made by taking previous stumps' mistakes into account.

→ Description: (Classification)

↳ We get dataset. (x, y) .

↳ We assign equal weights to all (standardize weights to 1)
(for 8 entries - weight for each = $\frac{1}{8}$)

↳ Stumps are created for decision trees with different features.

↳ Stump with least Gini impurity / Entropy is selected.

↳ Wrongly classified data is marked.

↳ Total error = Sum of weights of wrongly classified data
(TE)

$$\left\{ \frac{1}{2} \times n = \text{max case} \right\}$$

↳ Amount of say for stump (AoS) = $\frac{1}{2} \log \left(\frac{1 - TE}{TE} \right)$

↳ Weight updates:

misclassified: New weight = sample weight $\times e^{AoS}$

not classified: New weight = sample weight $\times e^{-AoS}$

Standardize weights to 1

Continue this process of finding stumps using new data set.

All datasets have different weights. Use these as buckets.

1. $0 - 0.05$ $0.05 - 0.1$ $0.1 - 0.5$
↓ ↓ ↓
1 2 3

Random no. from 0-1. Pick that data entry. Misclassified data will get selected more. Use this to form data set of same size as original.

Repeat process of forming stumps.

HENCE

Lots of stumps with different AoS are created.

Sum up AoS for different classes.

Class with highest AoS wins / is the output.

NOTE

Instead of creating new dataset using weight buckets we can also use weighted Gini-impurity in stumps.

But the other process is generally preferred.

REGRESSION

Trees use Variance Reduction.

Weighted average of predicted values from stumps are taken to make prediction.