MACHINE LEARNING (ML-17)

Dr. NEERAJ GUPTA, Department of CEA, GLA University, Mathura

AGENDA

- Ensemble Methods

(1) Max Voting ←

(2) Averaging ←

(3) Weighted Averaging ←

1	2	3	4	5	
4	4	2	4	5	3
4	4	2	4	5	

→ $\frac{19}{5} = 3.8$

$$W_1 x_1 + W_2 x_2 + W_3 x_3 + W_4 x_4 + W_5 x_5$$

Movie director

↓

short movie → public

ask your friends – 5 (4, 4, 2, 4, 5)

50 peoples →

Advanced Ensemble tech.

- Bagging
- Boosting

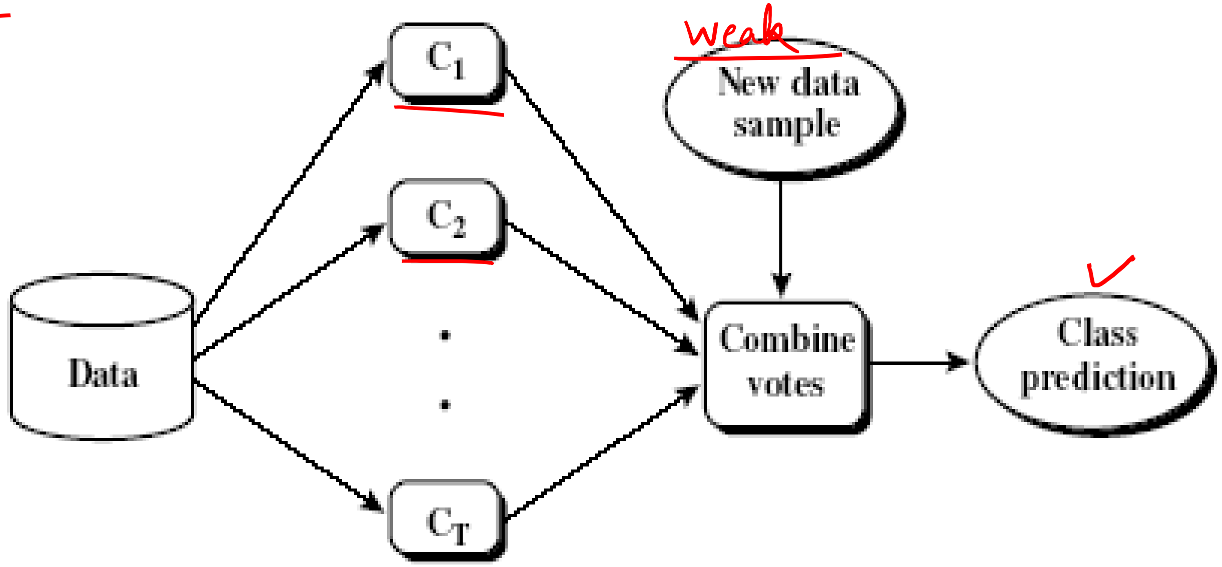
INTRODUCTION

Ensemble methods

- Use a combination of models to increase accuracy
- Combine a series of k learned models, M1, M2, …, Mk, with the aim of creating an improved model M*

strong

weak



INTRODUCTION

Two most popular ensemble methods are bagging and boosting.

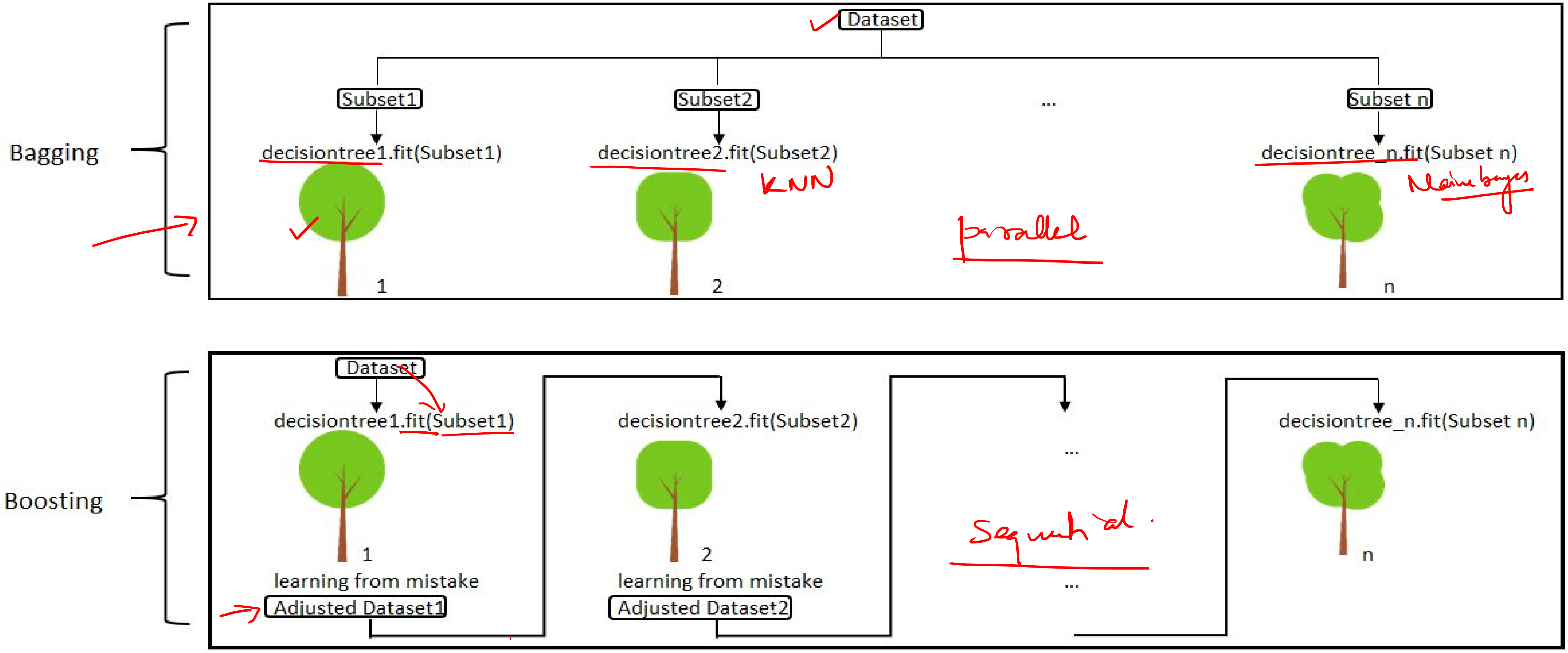
Bagging: Training a bunch of individual models in a parallel way. Each model is trained by a random subset of the data.

- averaging the prediction over a collection of classifiers

Boosting: Training a bunch of individual models in a sequential way. Each individual model learns from mistakes made by the previous model.

- weighted vote with a collection of classifiers

BAGGING VS BOOSTING



BAGGING: BOOSTRAP AGGREGATION

Analogy: Diagnosis based on multiple doctors' majority vote

Training

- Given a set D of d tuples, at each iteration i, a training set Di of d tuples is sampled with replacement from D (i.e., bootstrap)
- A classifier model Mi is learned for each training set Di

Classification: classify an unknown sample X

- Each classifier Mi returns its class prediction
- The bagged classifier M* counts the votes and assigns the class with the most votes to X

BAGGING: BOOSTRAP AGGREGATION

Prediction: can be applied to the prediction of continuous values by taking the average value of each prediction for a given test tuple

Accuracy

- Often significantly better than a single classifier derived from D
- For noise data: not considerably worse, more robust
- Proved improved accuracy in prediction

BOOSTING

Analogy: Consult several doctors, based on a combination of weighted diagnoses—weight assigned based on the previous diagnosis accuracy

How boosting works?

- Weights are assigned to each training tuple
- A series of k classifiers is iteratively learned
- After a classifier Mi is learned, the weights are updated to allow the subsequent classifier, Mi+1, to pay more attention to the training tuples that were misclassified by Mi
- The final M* combines the votes of each individual classifier, where the weight of each classifier's vote is a function of its accuracy

Boosting algorithm can be extended for numeric prediction

Comparing with bagging: Boosting tends to have greater accuracy, but it also risks overfitting the model to misclassified data

Bagging vs Boosting

greater accuracy

overfitting

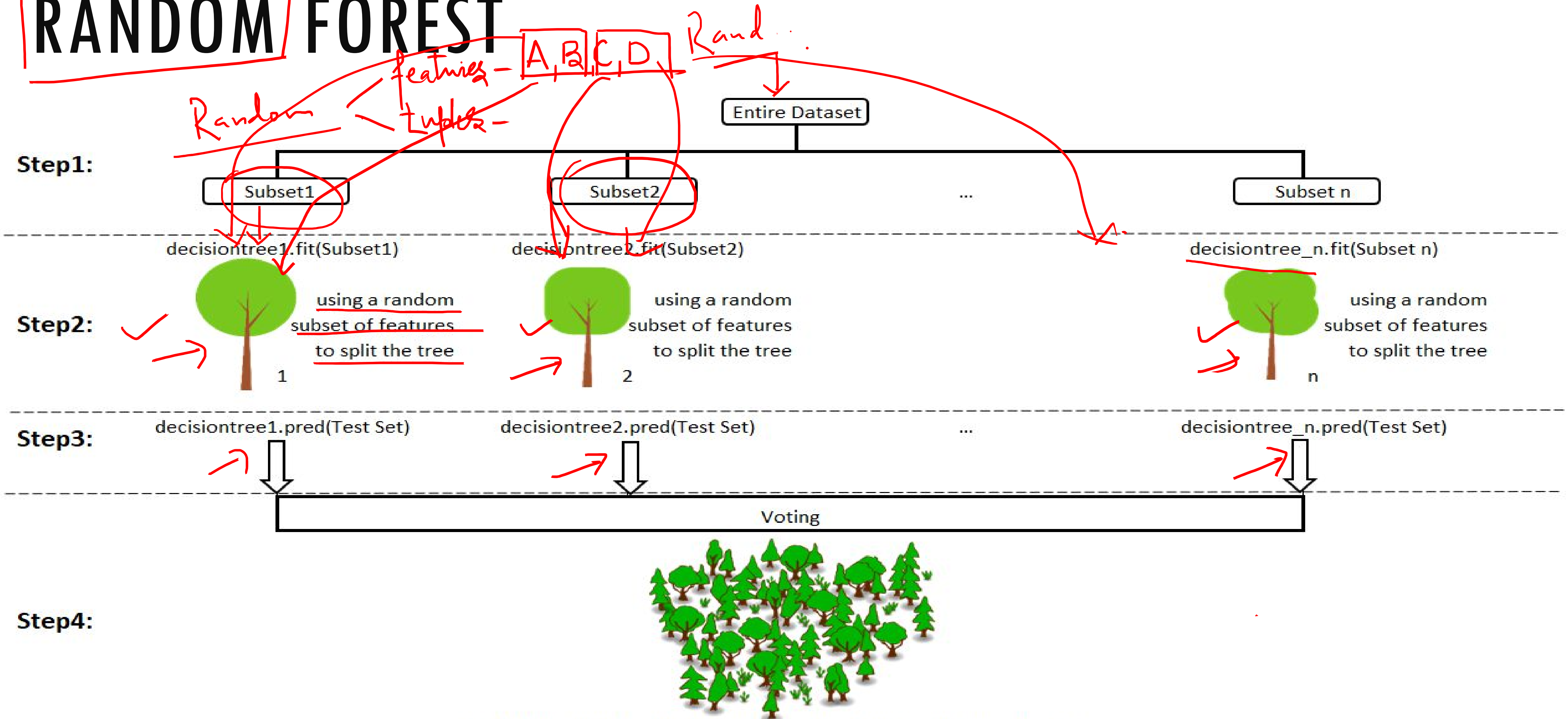
RANDOM FOREST

Random forest is an ensemble model using bagging as the ensemble method and decision tree as the individual model.

Problem with Bagging

- Works by reducing the variance
- Its possible of tree to be correlated, since presence of indicative features would lead to similar split in each tree

RANDOM FOREST



Final Prediction: Use the Majority Vote for Each Candidate in the Test set

RANDOM FOREST (BREIMAN 2001)

Random Forest:

- Each classifier in the ensemble is a decision tree classifier and is generated using a random selection of attributes at each node to determine the split
- During classification, each tree votes and the most popular class is returned

Two Methods to construct Random Forest:

- Forest-RI (random input selection): Randomly select, at each node, F attributes as candidates for the split at the node.
- Forest-RC (random linear combinations): Creates new attributes (or features) that are a linear combination of the existing attributes (reduces the correlation between individual classifiers)

Comparable in accuracy to Adaboost, but more robust to errors and outliers

Insensitive to the number of attributes selected for consideration at each split, and faster than bagging or boosting

GLA UNIVERSITY MATHURA
Accredited with A Grade by NAAC

ADABOOST (FREUND AND SCHAPIRE, 1997)

Given a set of d class-labeled tuples, (X1, y1), …, (Xd, yd)

W

Initially, all the weights (wi) of tuples are set the same (1/d)

Generate k classifiers in k rounds. At round i,

- Tuples from D are sampled (with replacement) to form a training set Di of the same size
- Each tuple's chance of being selected is based on its weight
- A classification model Mi is derived from Di
- Its error rate is calculated using Di as a test set
- If a tuple is misclassified, its weight is increased, o.w. it is decreased

Error rate: err(Xj) is the misclassification error of tuple Xj. Classifier Mi error rate is the sum of the weights of the misclassified tuples:

$$error(M_i) = \sum_{j}^{d} w_j \times err(\mathbf{X_j})$$

The weight of classifier Mi's vote (αM) is $\log \frac{1 - error(M_i)}{error(M_i)}$

REFERENCES

Han, Jiawei, Jian Pei, and Micheline Kamber. Data mining: concepts and techniques. Elsevier, 2011.

T. M. Mitchell, Machine Learning. McGraw-Hill Science, 1997.

THANKS


Keep Learning
Keep Growing


Dr. Neeraj Gupta
Assistant Professor, Dept. of CEA
neeraj.gupta@gla.ac.in