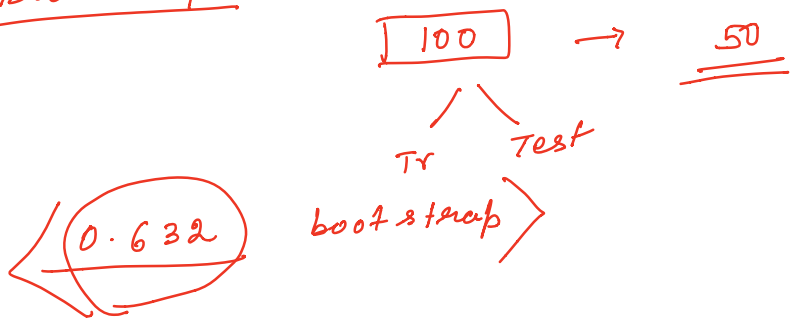
Lec-27 16/03/21 → Bootstrap
→ ROC curve.

Bootstrap.



Suppose we are given a data set of 'd' tuples (d = 20 if)
The data set is sampled 'd' times with replacement resulting in a bootstrap sample or training set of 'd' samples.
Trying out several times, it is observed that 63.2% of original data will end up in bootstrap and remaining 36.8% will form the test set.

prob. that each tuple will be selected = 1/d
" " " " not being " = (1 - 1/d)
As 'd' times the process is repeated,
the prob. that tuple not being selected $= \left(1 - \frac{1}{d}\right)^d$
If d is large, the prob. approaches $e^{-1} = 0.368$
Thus, 36.8% of tuples will not be selected for training and thereby end up in test set. The remaining 63.2% will form the training set.
→ The sampling procedure is repeated K-times.

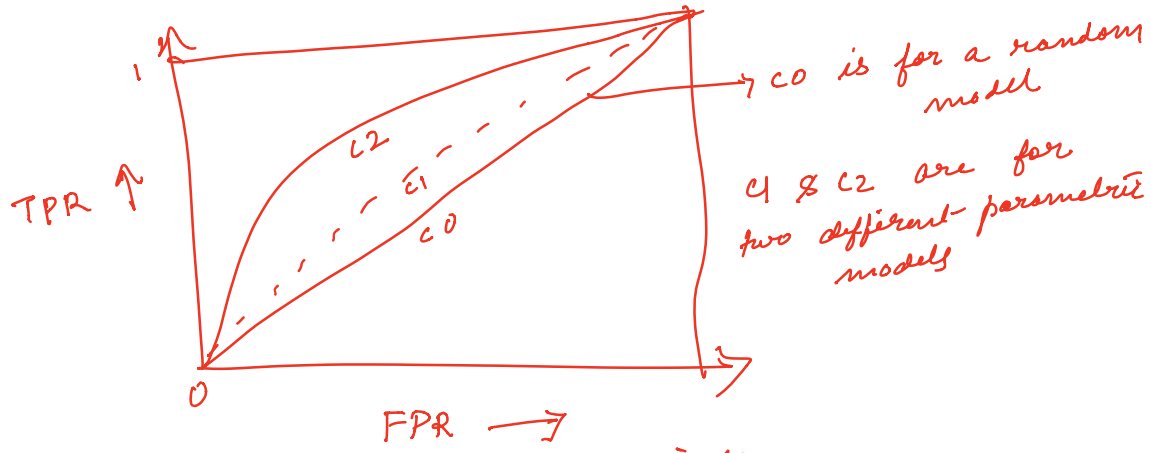
$$Acc(M) = \sum_{i=1}^{k} (0.632) \times Acc(M_i)_{\text{test set}} + 0.368 \times Acc(M_i)_{\text{Train}}$$

Model accuracy over test set

train set

ROC curve : (Receiver operating characteristic curve)

→ The relative increase of FPR with an increase in TPR is captured in ROC of a model.



(0,0) represents all cases to be negative.

(1,1) " " " " positive

(0,1) " perfect classifier

(1,0) point represents incorrect classifications for every positive that exists.

The diagonal line from (0,0) to (1,1) represents a random model where we can expect an equal no. of true & false cases.

8.6
Han Kamber
3rd Edn

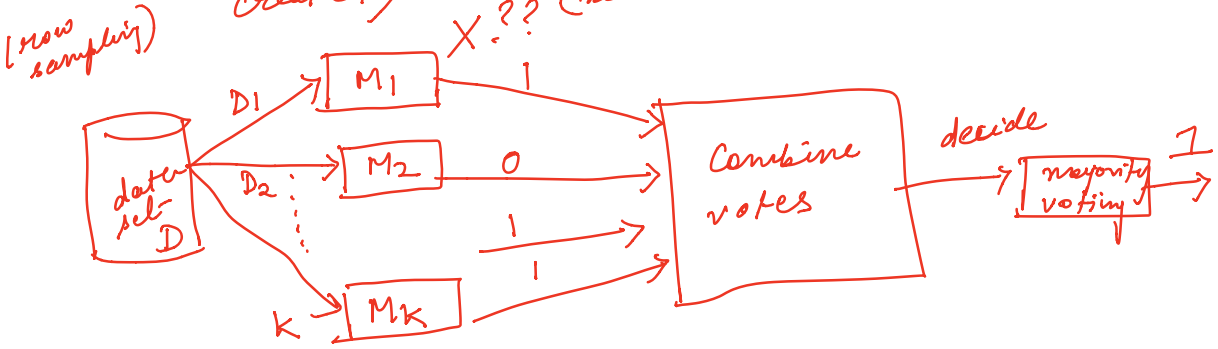
Techniques to improve classification accuracy (Ensemble approaches)

- Bagging (Bootstrap aggregation)
  - 1) Random forest
- Boosting (in syllabus)
  - 1) Adaboost
  - 2) Gradient Boost
  - 3) Xg Boost (Extreme Gradient Boost)

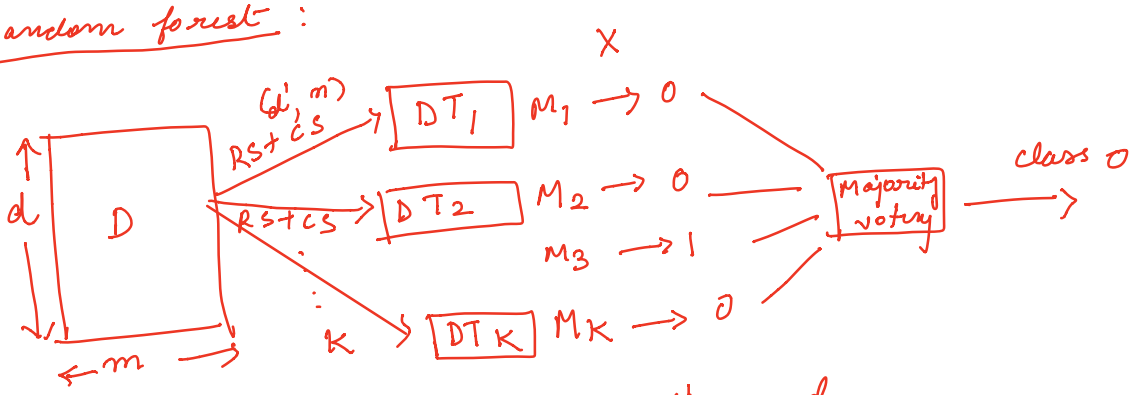
Bagging :

$M_1, M_2 \cdots M_K \rightarrow M_*$ (composite model)

(K no. of base models are created)



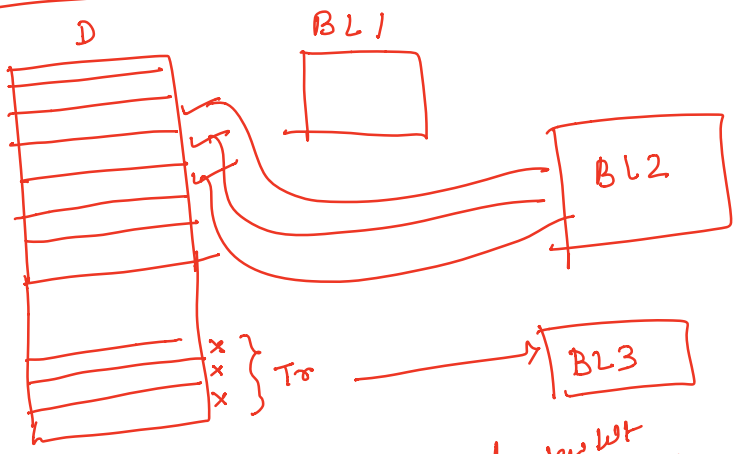
Random forest :



Row sampling (d' rows) s.t $d' << d$

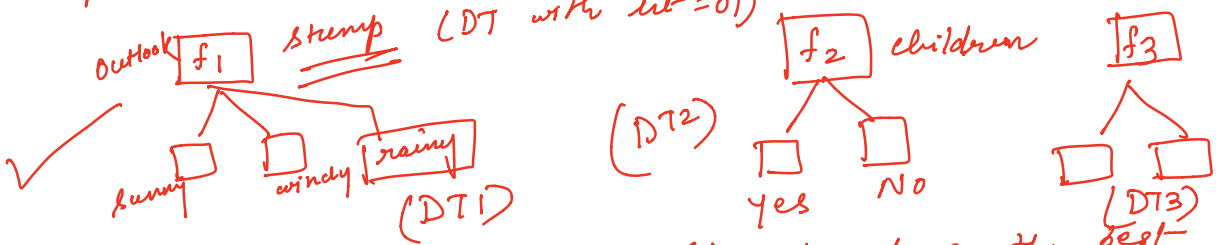
column sampling (n columns) s.t $n << m$

Adaboost : base learner



	f_1	f_2	f_3	O/P	Step1 sample wt $w = w_t \cdot 1/n$	New Wt	Normalize wts	
1	✓			0	1/7	0.05	0.05 / 0.649 →	0.077
2	✓			1	1/7	0.05	0.05 / 0.649 →	0.077
3	X			0	1/7	0.349	0.349 / 0.649 →	0.53
4	✓			1	1/7	0.05	⋮	⋮
5	✓			0	1/7	0.05	⋮	0.07
6	✓			1	1/7	0.05	⋮	
7	✓			0	1/7	0.05		
					Σ = 1	Σ = 0.649	Σ = 1	

step2: create the base learners (here it is DT)



(use min entropy, Gini to choose the best stump)

step3: Evaluate error and performance of stump

Total Error = (1/7)

performance of stump = $\frac{1}{2} \log_e \left(\frac{1 - TE}{TE} \right)$

$$= \frac{1}{2} \log_e \left(\frac{1 - 1/7}{1/7} \right)$$

$$= \frac{1}{2} \log_e (6)$$

$$= 0.896$$

Step 4 : update wts. s.t increase wts. of misclassified record and decrease wts. of correct classified record.

i) Incorrectly classified record

New wt = old wt $\times$ $e^{\text{total Error}}$

$$= \frac{1}{7} \times e^{0.895}$$

$$= 0.349$$

ii) correctly classified record

$$= \frac{1}{7} \times e^{-(0.895)}$$

$$= 0.05$$

Step 5 : Normalize the wts. so that the summation = 1 $\left(\sum wt = 01 \right)$

Step 6 : **create buckets** :

0.07	0 – 0.07
0.07	0.07 – 0.14
0.53	0.14 – 0.67
0.07	0.67 – 0.74
0.07	0.74 – 0.81
0.07	0.81 – 0.88
0.07	0.88 – 0.95

Generate random no 8 times

0.43	rec 03
0.31	rec 03
0.68	rec 04
0.58	rec 03

Accordingly, the records are selected and the same process continues till the classification accuracy is attained as per user specified level.