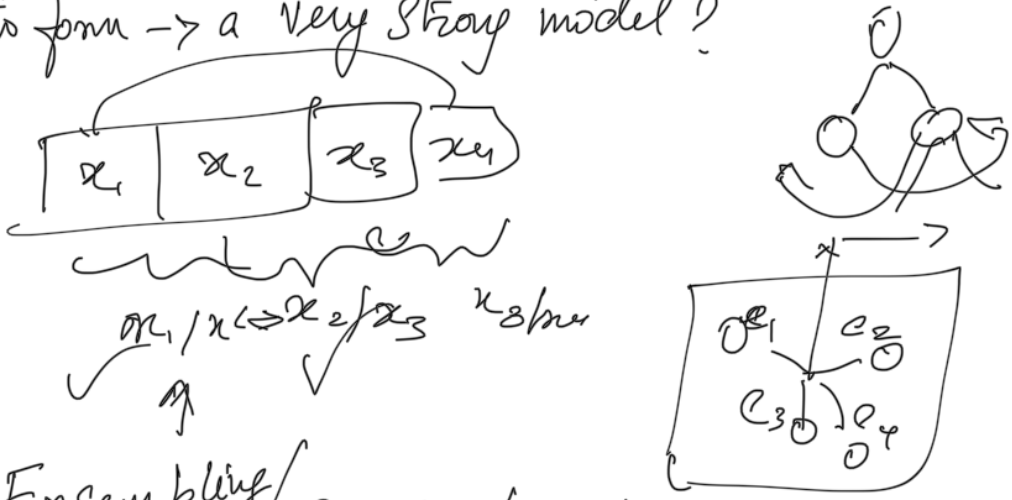
Combines multiple models / Multiple weak learners to form -> a very Strong model?



Ensembling / Ensemble Learning

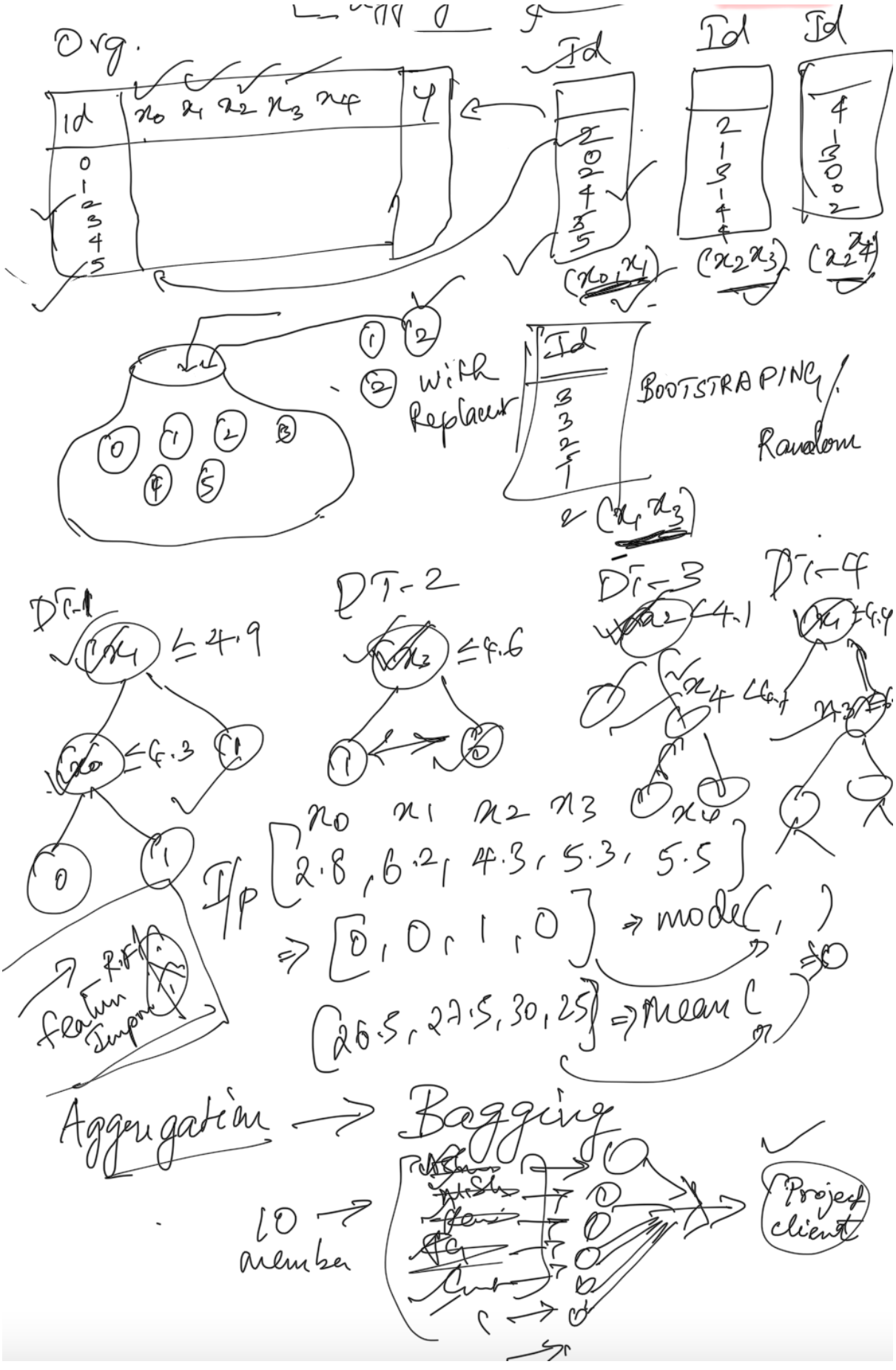
① Bagging ⚠ ② Boosting ⚠

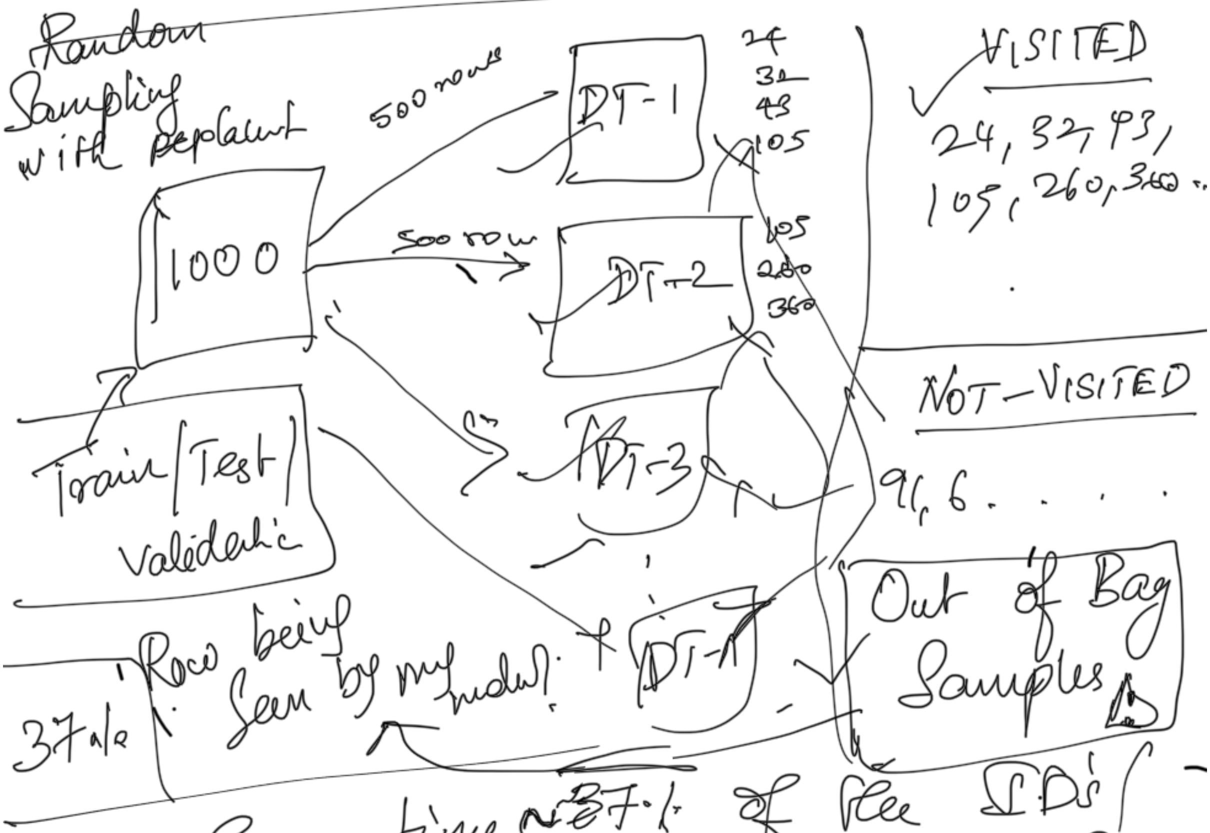
Highly Capable !!!

Bagging approach

① Random forest does not overfit.
② No assumption of Linear Relationship / Normal Dist.
③ less to No. feature Engineering.
④ Supports Multi-core processing -> n_jobs

Bagging — Bootstrap + aggregation — Random Forest


Orig.
Id
x0 x1 x2 x3 x4
y
(x0, x4)
(x2 x3)
(x2 x4)
with Replacement
BOOTSTRAPING / Random
DT-1
DT-2
DT-3
DT-4
≤ 4.9
≤ 4.3
≤ 4.6
≤ 4.1
≤ 4.9
≤ 4.7
I/p
[2.8, 6.2, 4.3, 5.3, 5.5]
[0, 0, 1, 0] → mode(,)
[26.5, 27.5, 30, 25] → mean
Feature Importance
Aggregation → Bagging
10 → members
Project client



Every time 37% of the DTs rows are never seen by the model !!

=> Out of Bag Score

Test	logistic	NB	KNN
6	1	0	1
20			
36			
40			

$\sqrt{n}$ / $\log(n)$

0.60, 0.70

⇒ HYPER-Parameter

values → used during the training to estimate the value of model param.

⇒ External to the model →

⇒ User

Model Parameters	Hyper param
Model Params	n_iter → Beforehand
y_i → during the training.	learn
Part of the model.	max_depth Ex
eg Slope, Intercept	num

Model Parameters	Hyper Param v. Tuning
(1) Estimated during Training proc	(1) This is set beforehand.
(2) Part of the model itself	(2) External value
eg) Intercept & Slope	eg) ⇒ n_iter, K, LearnRate

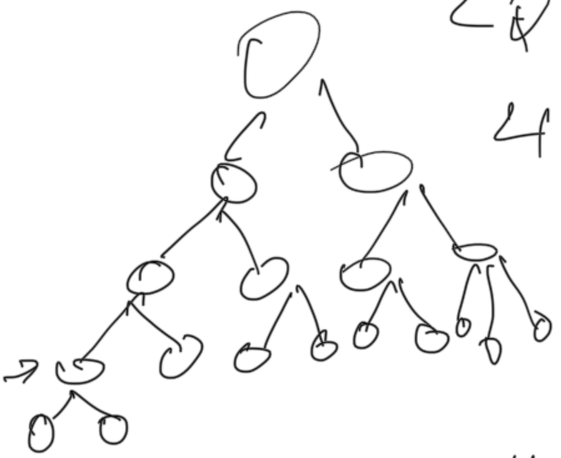
(1) n_estimators ⇒ No. of trees to (n) be present

(2) max-features ⇒ # of features for each tree

(0.30), (0.60), (0.50)

(3) max-depth ⇒ level or height of the tree

2
4
2
5



max dep = 20

max_dep = 5

Score

⇒ n_jobs ⇒ multi-core processing

⇒ OOB_score ⇒ prediction + Evaluation score on out-of-bag samples

rf.oob_score_

out of bag. default = false.

⇒ Bootstrap = True /

Random forest - very heavy slow algorithm