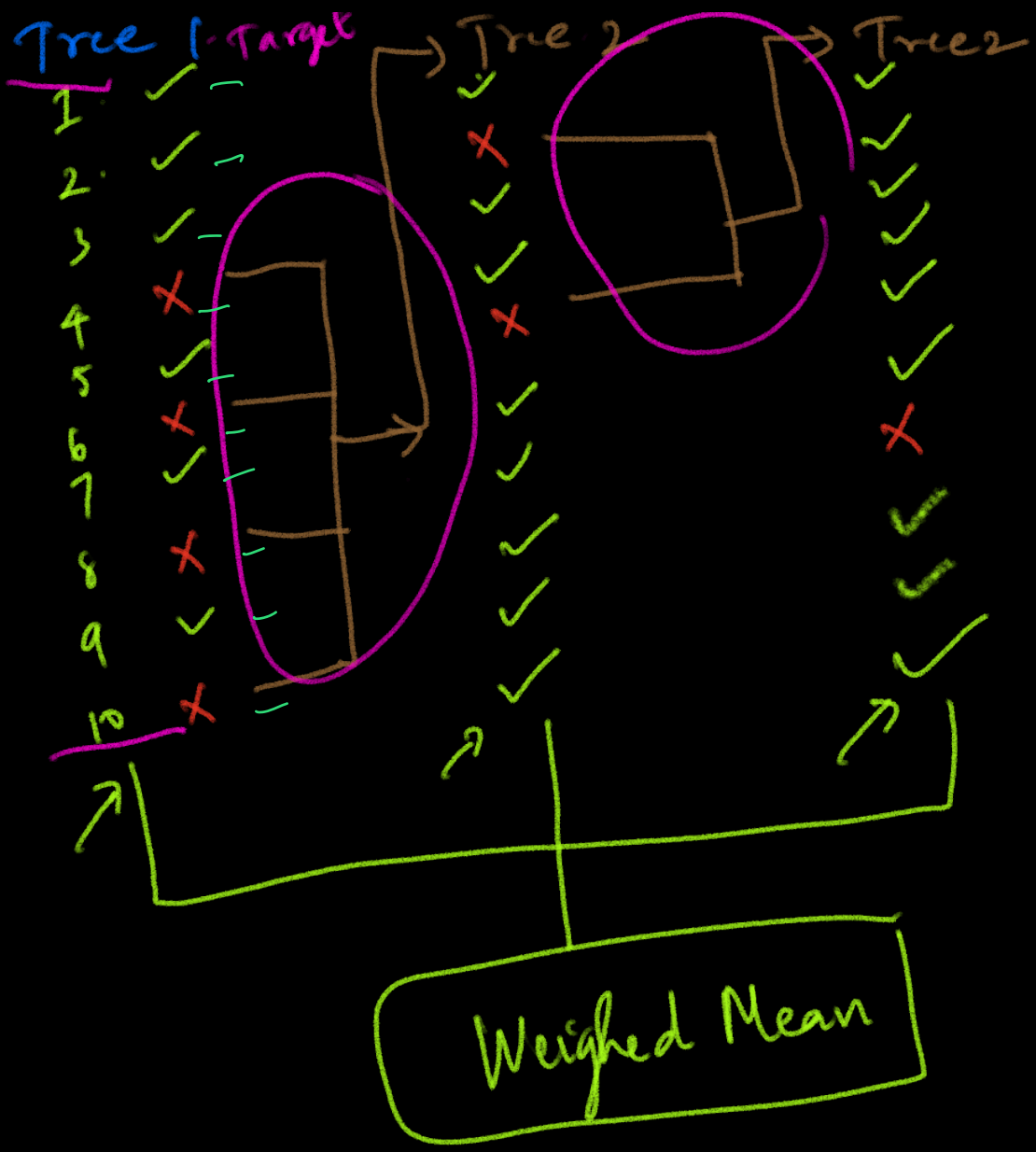

Tree 1 - Target
Tree 2
Tree 2
1
2
3
4
5
6
7
8
9
10
Weighed Mean

$$Y = mx + b + \varepsilon_1$$

($\varepsilon(mx+b)$)

$$\varepsilon_1 = Fx + z + \varepsilon_2$$

($\varepsilon(Fx+z)$)

$$\varepsilon_2 = Ax + D + \varepsilon_3$$

($+\varepsilon(Ax+D)$)

Gradient Boosting Method

$$Y = mx + b + Fx + z + Ax + D + \varepsilon_3$$

	Bagging	Boosting
objective	Prevent Overfitting Reduce Variance	Improve Accuracy
Base Learners	Independent models Decision Tree	weak lear Decision Tree
Training Process	Parallel training	Sequential training
Sample Selection	Random Sampling	weighted sampling based on errors
Predictions	Voting - Classificat Average - Regression	Weighed sum of base learne
Weighting of Data	All datapoints are equal weight	Data points have dofferent weights
Resampling	Bootstrap Sampling	NO
Example	Random Forest	Ada Boost Gradient Boost GBM Xgboost

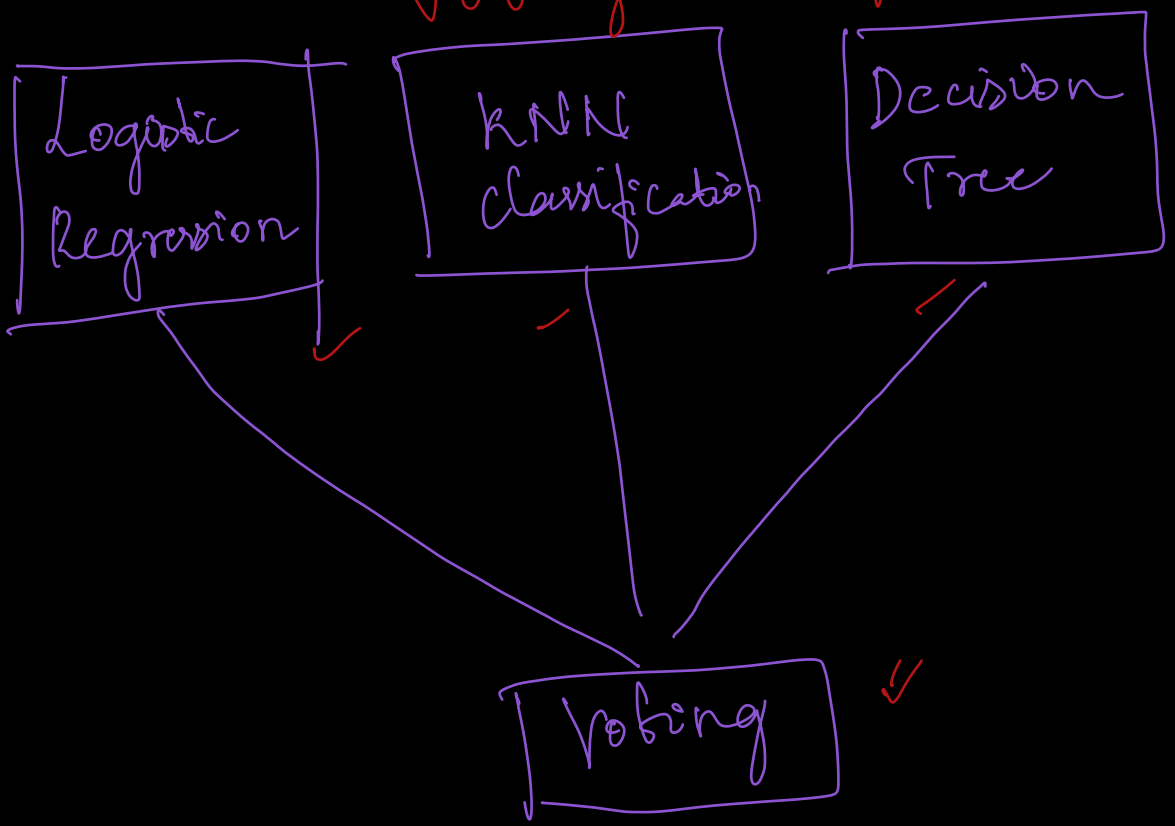
Limitations	Does not learn from mistake More space utilization May not perform well in Unbalanced sets less interpretable due of Multiple models	Sequential training can be slow More complex due to tuning Hyperparameters sensitive to noisy data and outliers
Handling Outliers	Robust	Sensitive

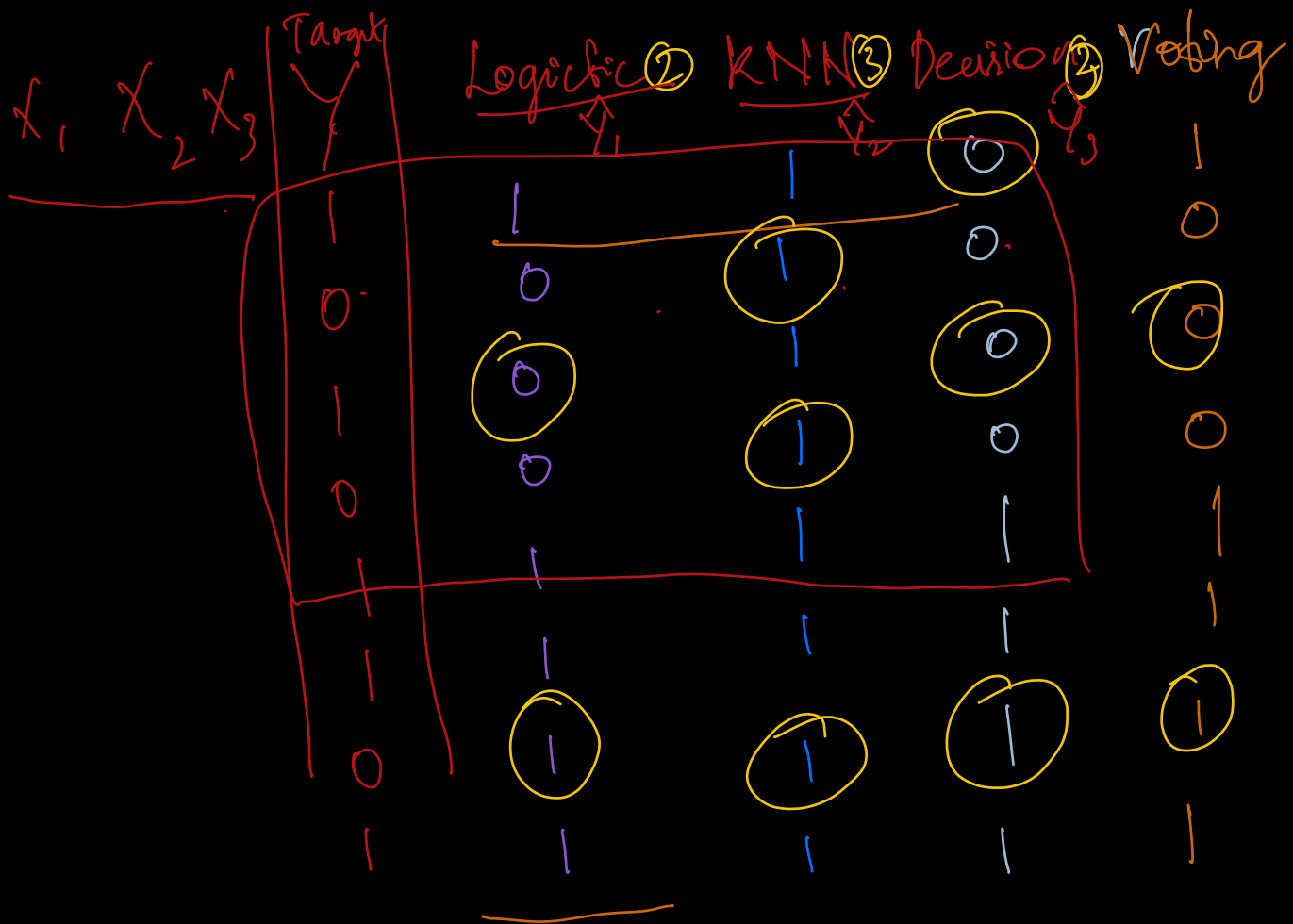
Advanced Ensemble Technique

Combining weak learners to become strong learners

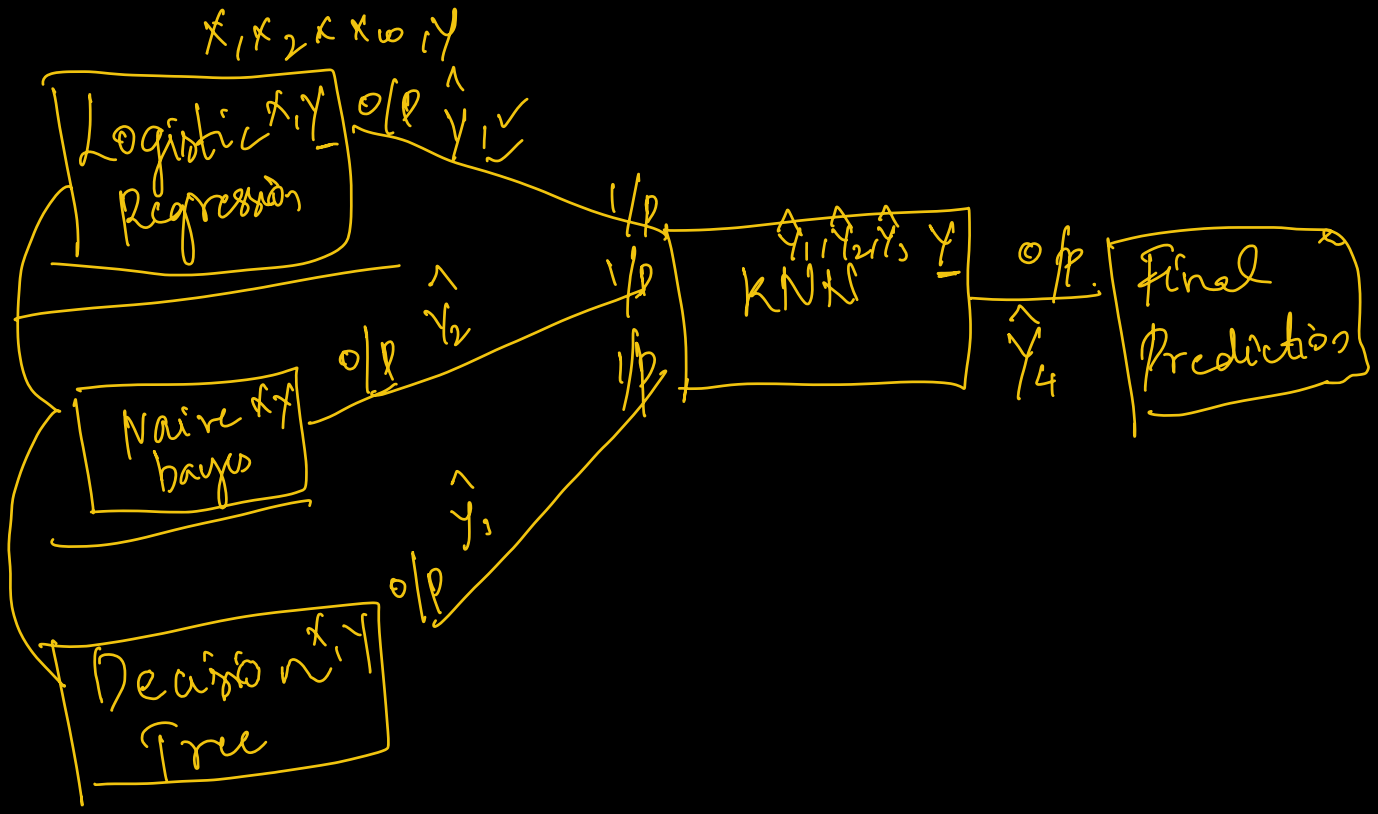
Combining strong learners to become even more stronger

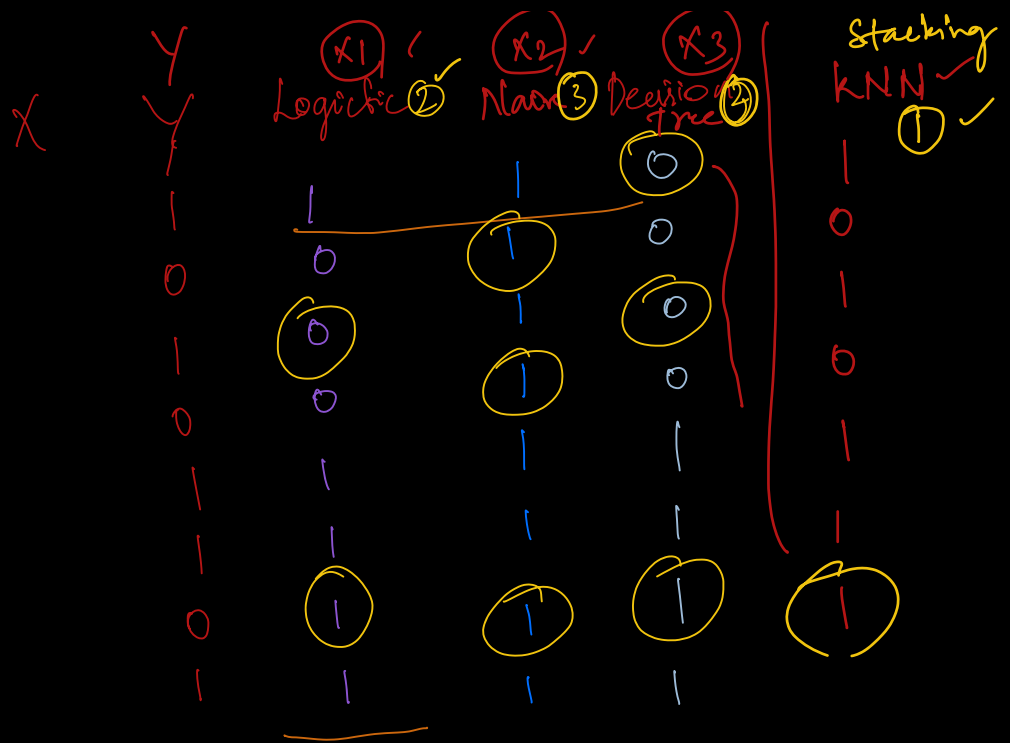
Voting Classification




X_1 X_2 X_3
Target
Y
Logistic ②
KNN ③
Decision ②
Voting
Y_1
Y_2
Y_3

Stacking Technique




X
Y
X1
X2
X3
Logistic
Naive
Decision Tree
Stacking
KNN