Ensemble Techniques

one of the weakness of using a single decision tree is that, that decision tree can be highly sensitive to small changes in data.

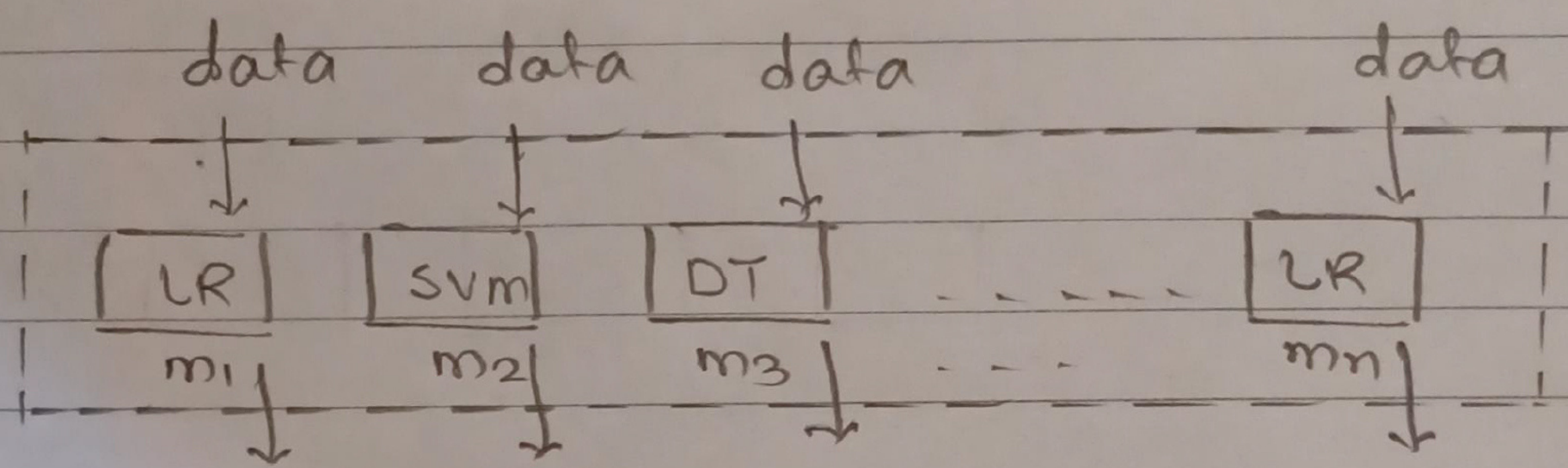
one solution to make the arrow less sensitive or more robust is to build not one decision tree, but to build a lot of decision tree and we call that a tree ensemble.

wisdom of crowd: refers to a phenomenon in which a group of people's collective opinion is more accurate than that of any individual member of the group.

Ensemble techniques are methods that involves training multiple models and combining their predictions to produce a more accurate result.

Komal Diwe

core idea: to combine the predictions of multiple models to produce a more accurate result.



condition:- Base model should be different. It can be can different in two forms.

① all machine learning model should be different on same data → algorithms

② or all base models should be same but on different samples of data.

after training several models independently and then, combining their predictions through a combination method such as voting, averaging, or weighting.

Komal Diwe

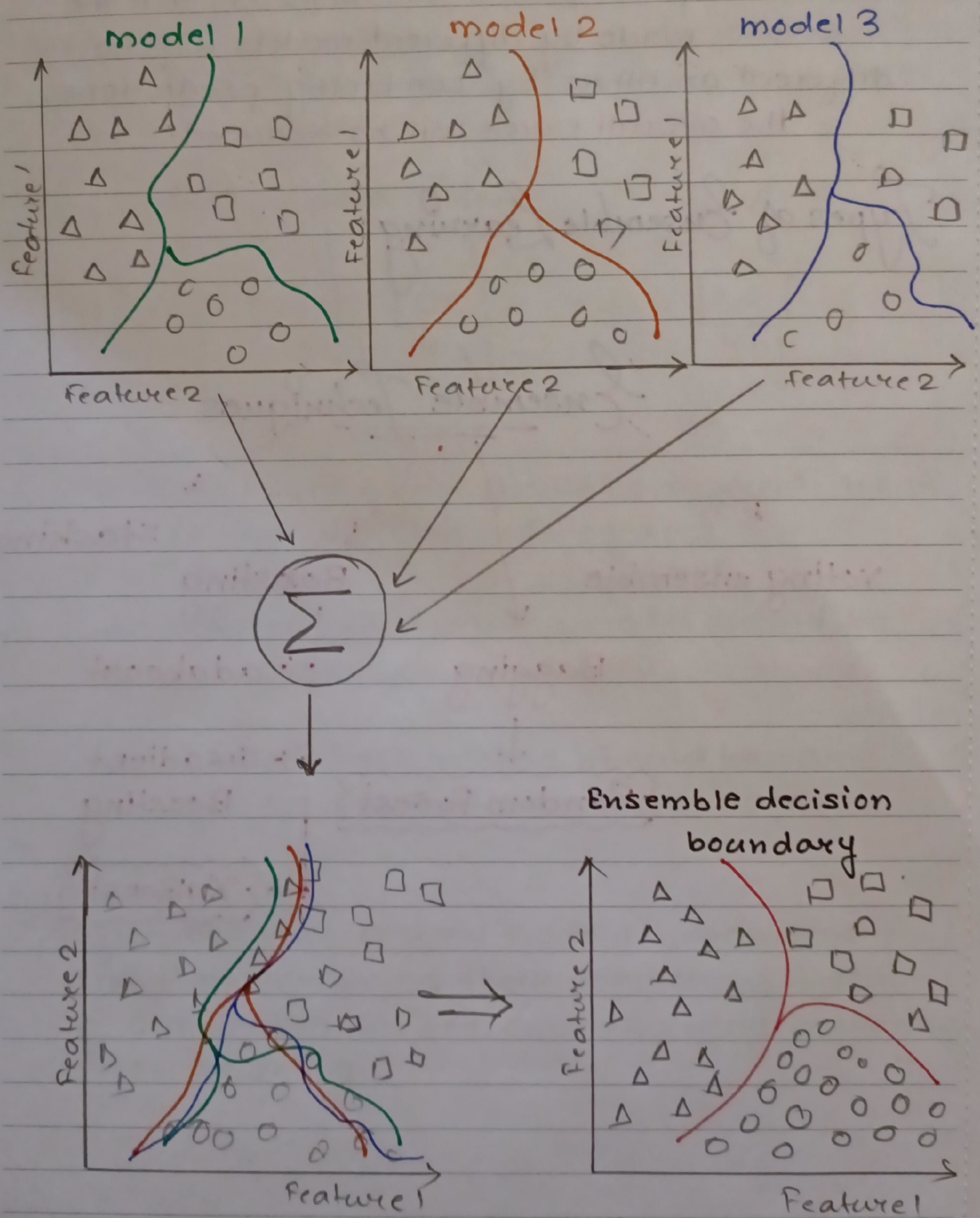
The assumption behind this method is that the errors made by different models will be different and thus by combining predictions, the overall error will be reduced.

Types of Ensemble Learning:

Ensemble Techniques

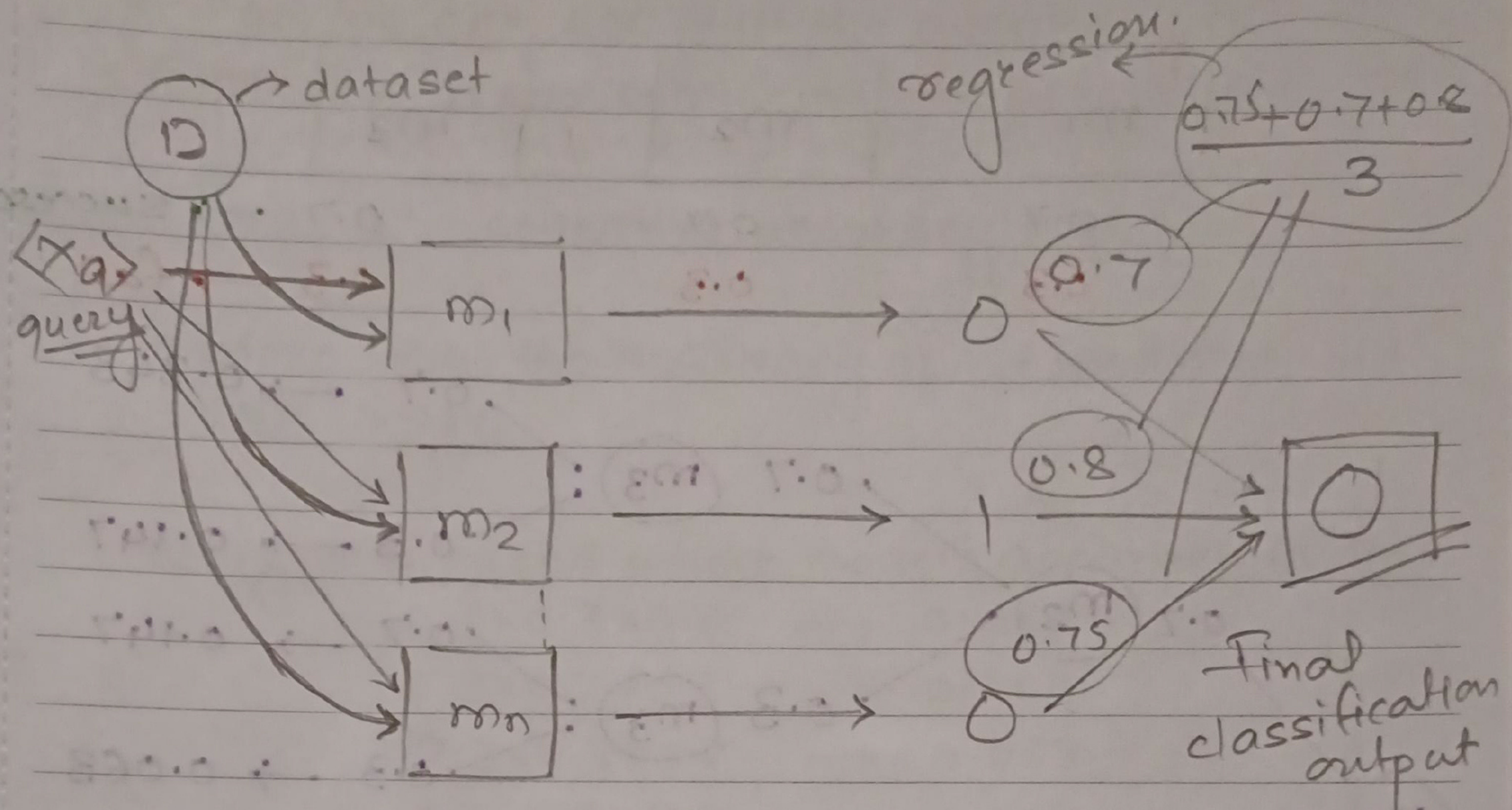
- voting ensemble
- Bagging → Random Forest
- Boosting
  - adaboost
  - Gradient Boosting
  - XgBoost
- Stacking

Komal Diwe


model 1
model 2
model 3
Feature 1
Feature 2
Feature 1
Feature 2
Feature 1
Feature 2
Σ
Feature 2
Feature 1
Ensemble decision boundary
Feature 2
Feature 1

Komal Diwe

Voting Ensemble



Here, we apply same dataset to all models. Models can be different or can be same.

Independently, all models are trained on same data.

The core intuition behind the voting ensemble technique is to combine the predictions of multiple models to produce a more robust and accurate final prediction.

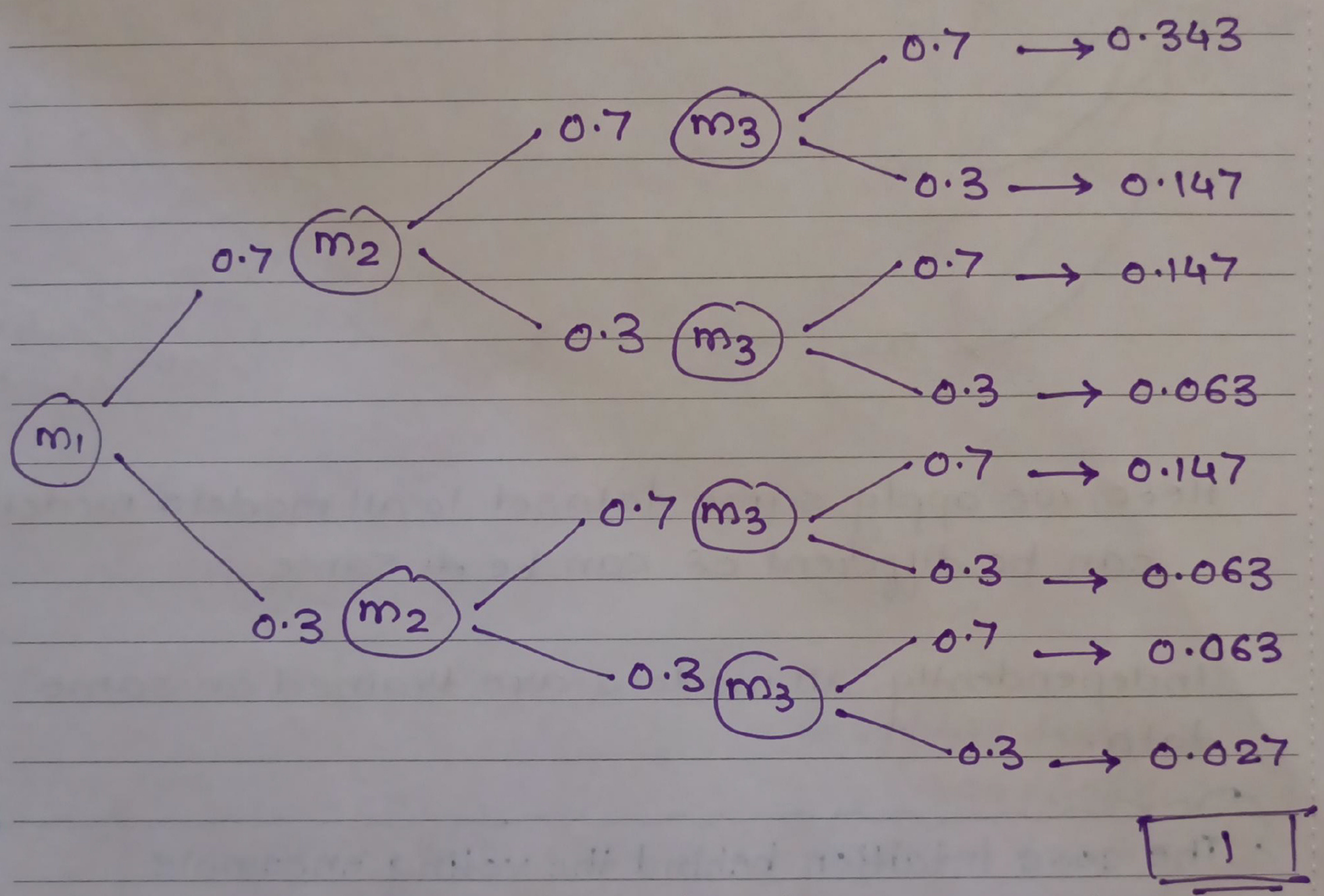
assumptions:-

1. models should be independent in nature.
2. accuracy of every model must be $\boxed{>50\%}$ otherwise resultant will be worst than the worst.

Komal Diwe

suppose, we have 3 models which have accuracy 0.7

m_1	m_2	m_3	
0.7	0.7	0.7	→ Success
0.3	0.3	0.3	→ fail



Now, suppose our voting classifier wants atleast two model's success accuracy.

then we get 4 possibilities:

1. $0.7 \times 0.7 \times 0.7 \rightarrow 0.343$
2. $0.7 \times 0.3 \times 0.7 \rightarrow 0.147$
3. $0.3 \times 0.7 \times 0.7 \rightarrow 0.147$
4. $0.7 \times 0.7 \times 0.3 \rightarrow 0.147$

0.784

78.4%

Komal Diwe

So, we can see that the accuracy is improved after combining the predictions.

∴ collective accuracy : 78.4%

To prove, that the accuracy of the model should be >50%.

we get 4 cases where model's accuracy is 30%.

$0.7 \times 0.3 \times 0.3 \rightarrow 0.063$

$0.3 \times 0.7 \times 0.3 \rightarrow 0.063$

$0.3 \times 0.3 \times 0.7 \rightarrow 0.063$

$0.3 \times 0.3 \times 0.3 \rightarrow 0.027$

$\boxed{0.216} \rightarrow 21.6\%$

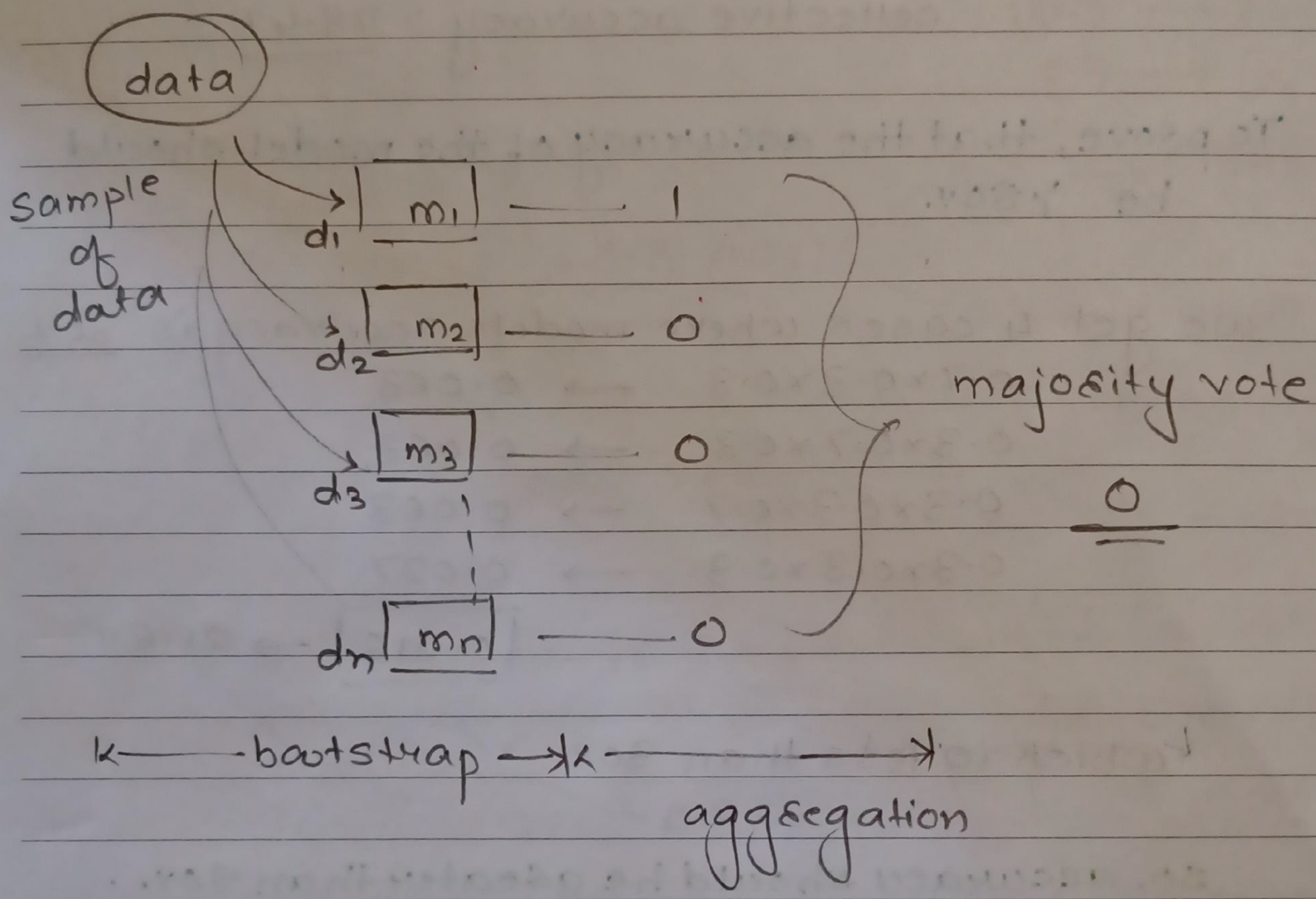
which is less than 30%.

so, accuracy should be greater than 50%.

Komal Diwe

Bagging.

Bootstraping + Aggregation = Bagging



· Every model gets different sample of data.

In this a random sample of data in a training set is selected with replacement - meaning that that individual datapoints can be choosen more than once.

Komal Dive

Bagging helps to reduce the variance of the model while keeping the bias roughly the same.

Bagging works by creating multiple versions of the same model, each trained on a different random subset of the training data.

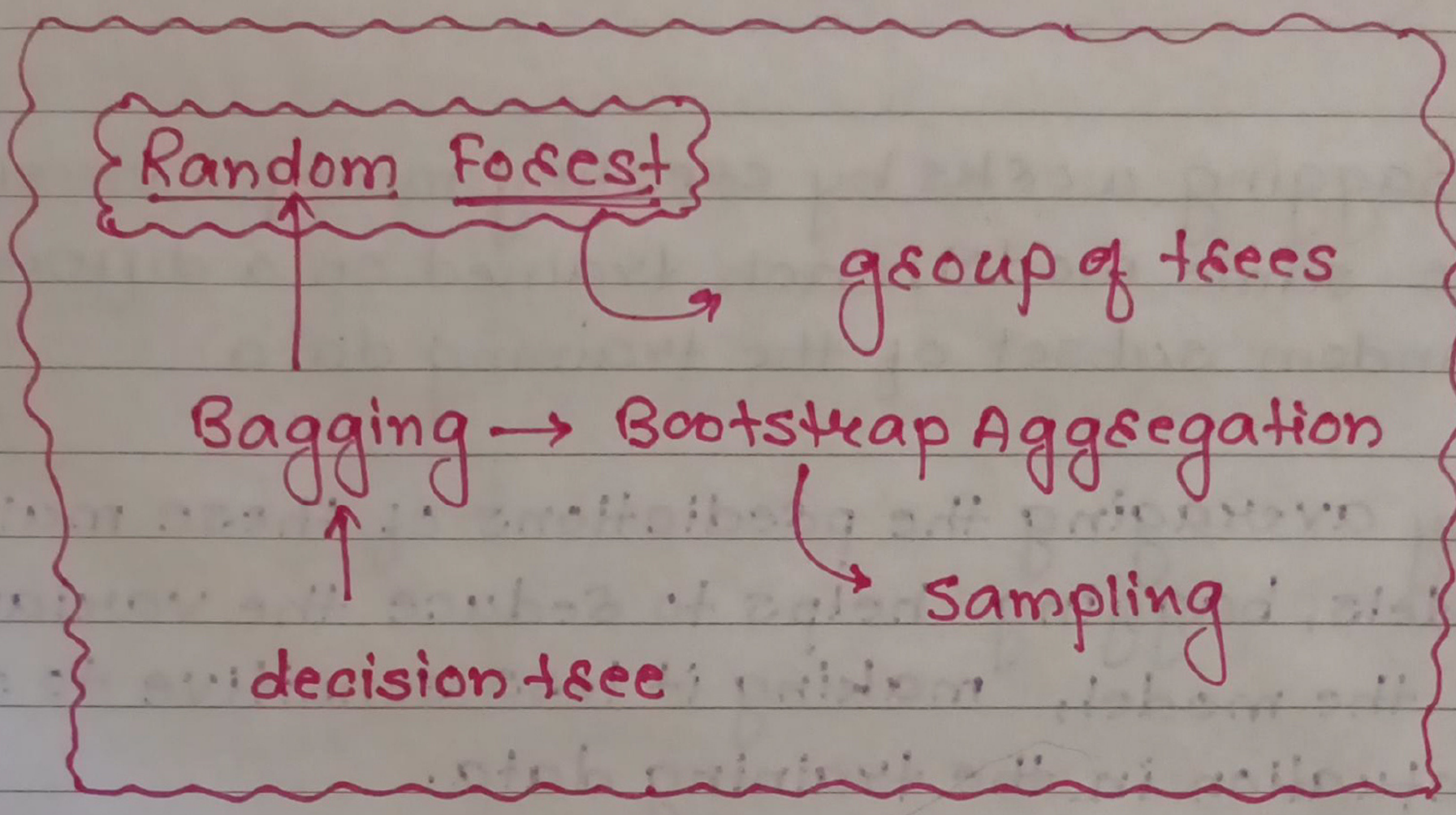
By averaging the predictions of these multiple models, bagging helps to reduce the variance of the model, making it less sensitive to small fluctuation in the training data.

However, since each model is trained on a different subset of the data, the bias of the model is not affected, as it uses the same assumptions and relationships as the original model.

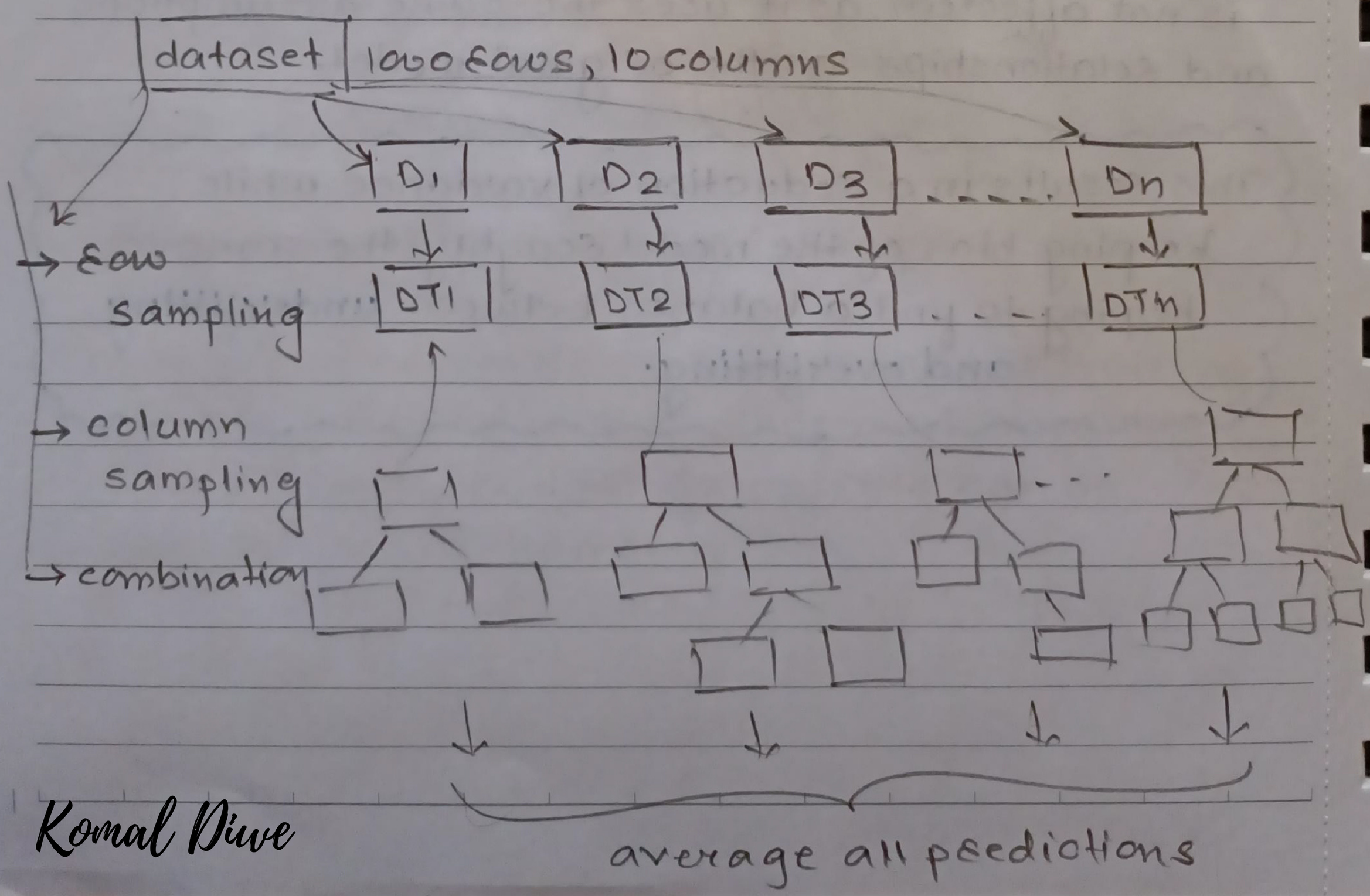
This results in a reduction of variance while keeping bias of the model roughly the same, helping to find a balance between underfitting and overfitting.

Komal Diwe

Random Forest



→ It is a bagging technique, where all base models are decision tree.



Komal Dive

Two characteristics of random forests allows a reduction in overfitting and the correlation between the trees.

The first is bagging, where individual decision trees are fitted following each bootstrap sample and then averaged afterwards.

Bagging significantly reduces the variance of the random forest versus the variance of any individual decision trees.

The second way Random forests reduce overfitting is that a random subset of features is considered at each split, preventing the important features from always being present at the tops of individual trees.

OOB Score (Out of Bag) Evaluation:

- while doing row sampling, feature sampling with replacement for each each model, so there is a chance that some set of values may not get selected.
- It is observed that nearly 37% of data is not selected.

Komal Dive

- we can use that unselected data points for testing and validation purpose.

- If you set OOB=True, then it will automatically handle missing data points for testing purpose.

: What is the difference between OOB score and validation score?

Since we have understood how OOB score is estimated let's try to comprehend how it differs from the validation score.

as compared to the validation score OOB score is computed on data that was not necessarily used in the analysis of the model.
whereas for calculation validation score, a part of the original training dataset is actually set aside before training the models.

Additionally, the OOB score is calculated using only a subset of DTs not containing the OOB sample in their bootstrap training dataset.
while the validation score is calculated using all the DTs of the ensemble.

Komal Diwe

In an ideal case, about 36.8% of the total training data forms the OOB sample.
This can be shown as follows:

If there are N rows in the training datasets
Then, the probability of not picking a row in a random draw is

$$\frac{N-1}{N}$$

using sampling-with-replacement the probability of not picking N rows in random draw is

$$\left(\frac{N-1}{N}\right)^N$$

which in the limit of large N becomes equal to

$$\lim_{N \to \infty} \left(1 - \frac{1}{N}\right)^N \simeq e^{-1} = 0.368$$

∴ about 36.8% of total training data are available as OOB sample ~~fore~~ for each DT and hence it can be used for evaluating or validating the random forest model.

Komal Dive