Ensemble Approaches

Dr. Shiple

Ensemble Philosophy

- Build many models and combine them
- Only through averaging do we get at the truth!
- It's too hard (*impossible*?) to build a single model that works best

Real Life example

- Suppose, you want to invest in a company XYZ. You are not sure about its performance though.
- So, you look for advice on whether the stock price will increase by more than 6% per annum or not?

The survey prediction

- Employee of Company XYZ:
  - In the past, he has been right 70% times.
- Financial Advisor of Company XYZ:
  - In the past, he has been right 75% times.
- Stock Market Trader:
  - In the past, he has been right 70% times.
- Employee of a competitor:
  - In the past, he has been right 60% times.

Summary

- • Use multiple learning algorithms (classifiers)
- Combine the decisions
- Can be more accurate than the individual classifiers
- Generate a group of base-learners
- Different learners use different
  - Algorithms
  - Hyperparameters
  - Representations (Modalities)
  - Training sets

- Difference in population
- Difference in hypothesis
- Difference in modeling technique
- Difference in initial seed

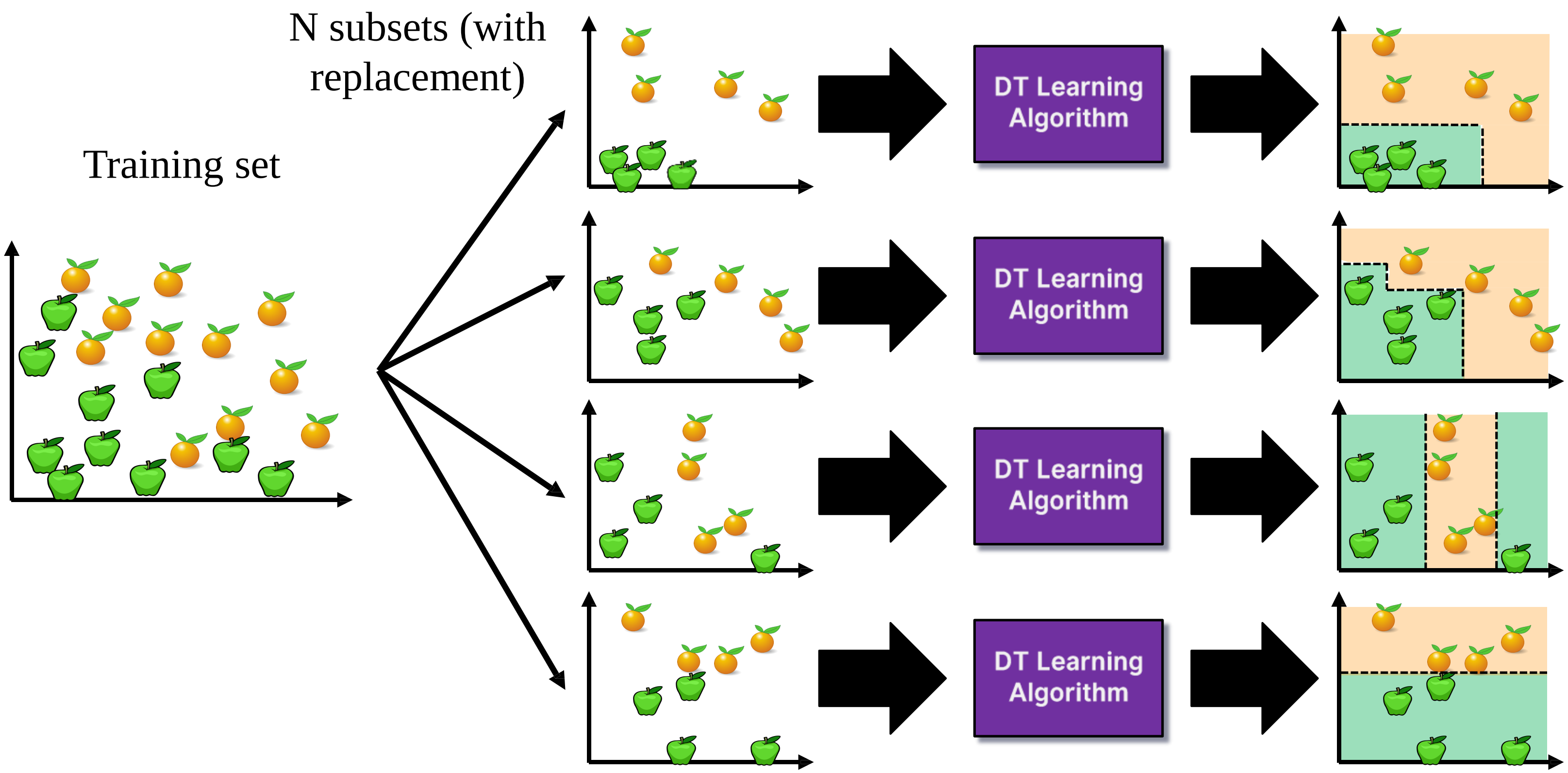
Why ensembles ?

- There are two main reasons to use an ensemble over a single model, and they are related; they are:
  - Performance: An ensemble can make better predictions and achieve better performance than any single contributing model.
  - Robustness: An ensemble reduces the spread or dispersion of the predictions and model performance.

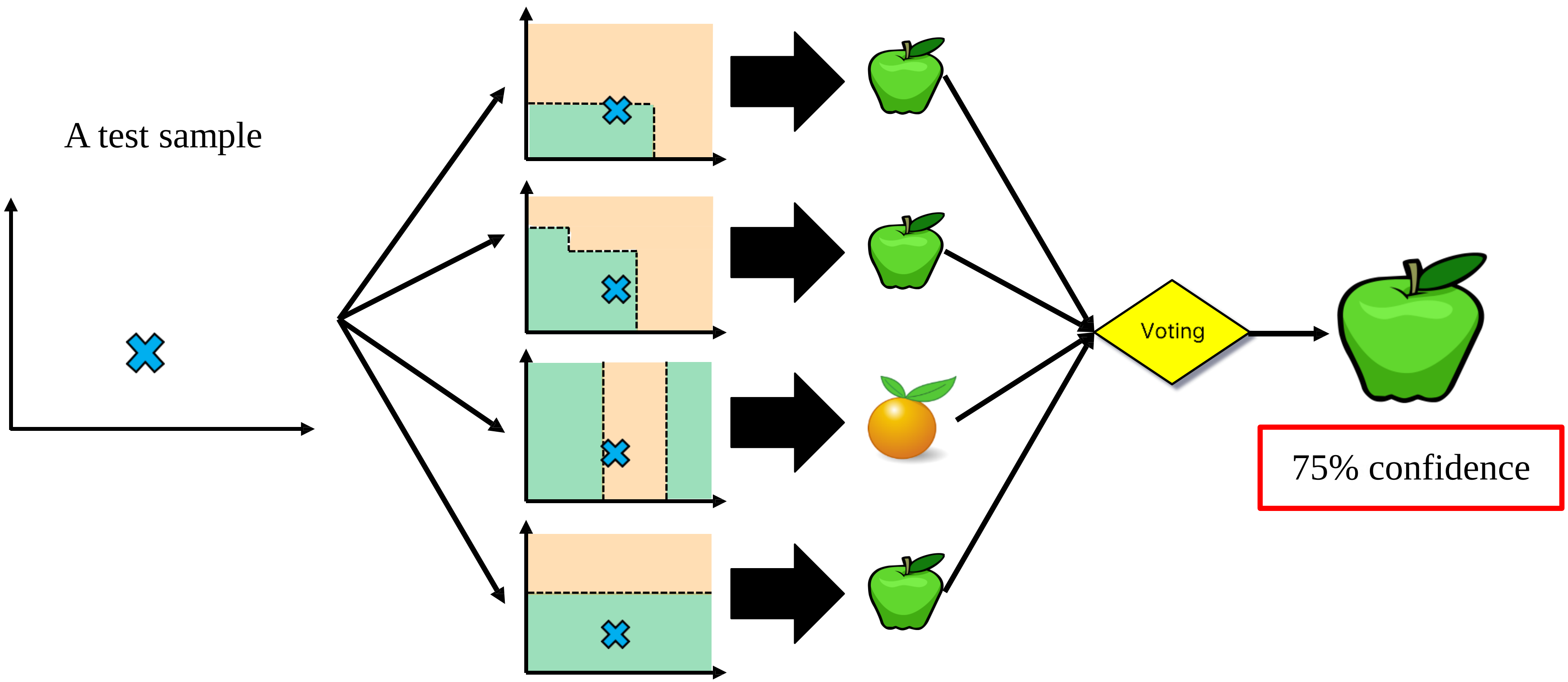
Ensemble Approaches

- Bagging (**B**ootstrap aggregating) (Unweighted Voting)
- Boosting (Weighted voting – based on accuracy)
- Staking (Learn the combination function)

Bagging at training time



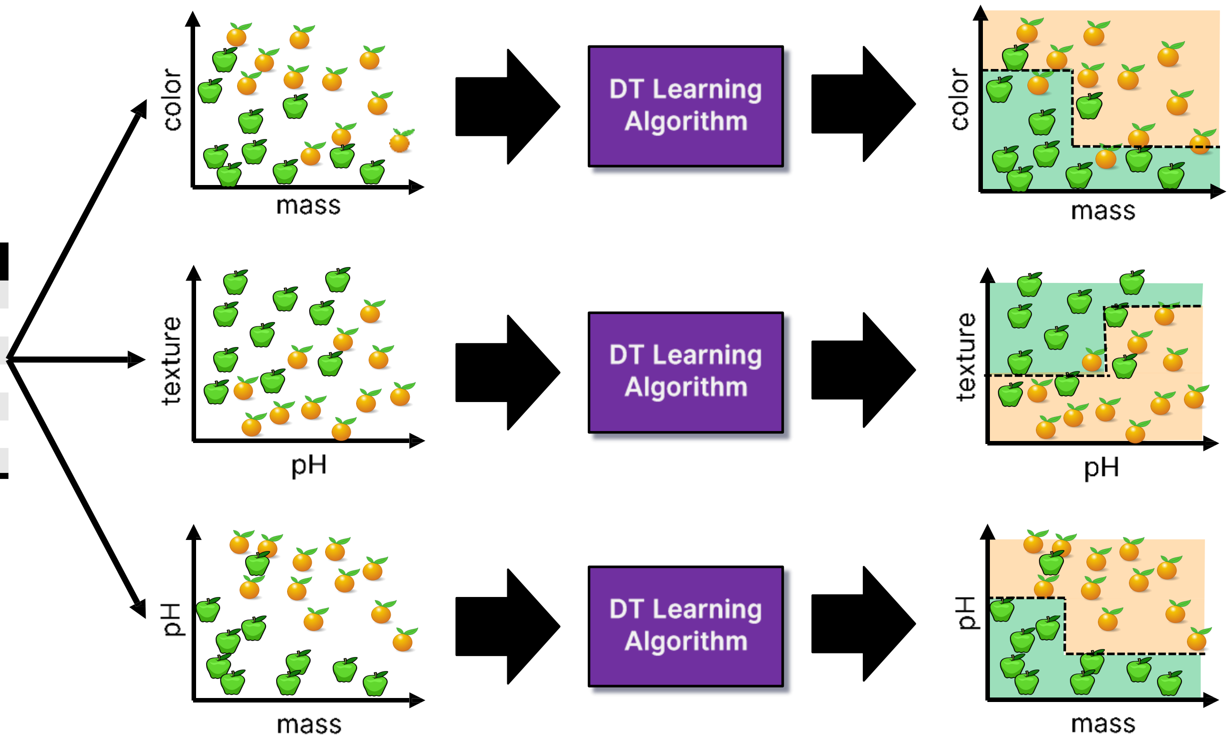
Bagging at inference time



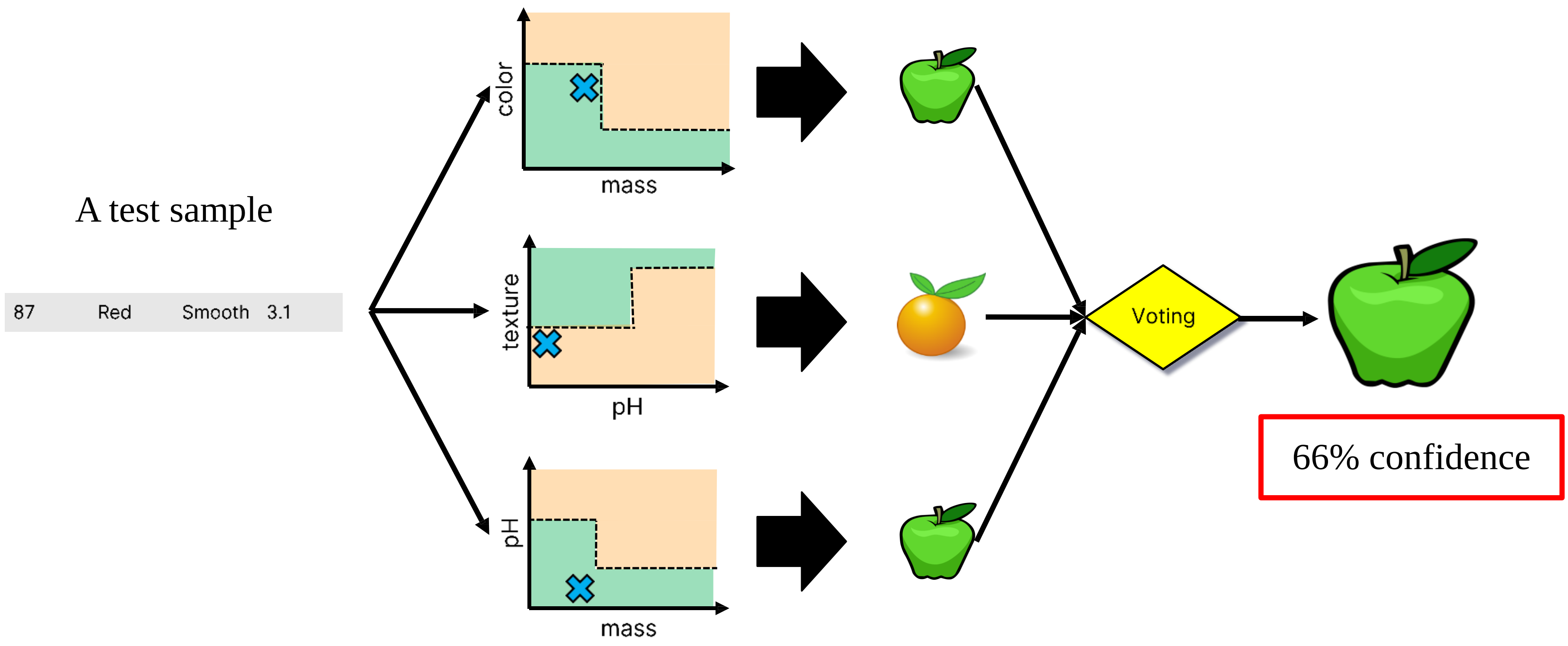
Random Subspace Method at training time

Training data

Mass (g)	Color	Texture	pH	Label
84	Green	Smooth	3.5	**Apple**
121	Orange	Rough	3.9	**Orange**
85	Red	Smooth	3.3	**Apple**
101	Orange	Smooth	3.7	**Orange**
111	Green	Rough	3.5	**Apple**
...				
117	Red	Rough	3.4	**Orange**



Random Subspace Method at inference time



1. Random forest is a type of supervised machine learning algorithm based on *ensemble learning*.
2. Ensemble learning is a type of learning where you join different types of algorithms or <u>same algorithm multiple times</u> to form a more powerful prediction model.
3. The random forest algorithm combines multiple algorithm of the same type i.e. multiple decision trees, resulting in a forest of trees, hence the name "Random Forest".
4. The random forest algorithm can be used for both regression and classification tasks.

The Bagging Algorithm

Given data: $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$

For $m = 1 : M$

- Obtain bootstrap sample D_m from the training data D
- Build a model $G_m(\mathbf{x})$ from bootstrap data D_m
- Dataset with replacement (meaning we can select the same value multiple times).

The Bagging Model

- Regression

$$\hat{y} = \frac{1}{M}\sum_{m=1}^{M} G_m(\mathbf{x})$$

- Classification:
  - Vote over classifier outputs $G_1(\mathbf{x}), \ldots, G_M(\mathbf{x})$

Boosting

- Boosting algorithms are a set of the low accurate classifier to create a highly accurate classifier.
- Low accuracy classifier (or weak classifier) offers the accuracy better than the flipping of a coin.
- This is done by building a model from the training data, then creating a second model that attempts to correct the errors from the first model. Models are added until the training set is predicted perfectly or a maximum number of models are added.
- Highly accurate classifier(or strong classifier) offer error rate close to 0. Boosting algorithm can track the model who failed the accurate prediction.
- Boosting algorithms are less affected by the overfitting problem.

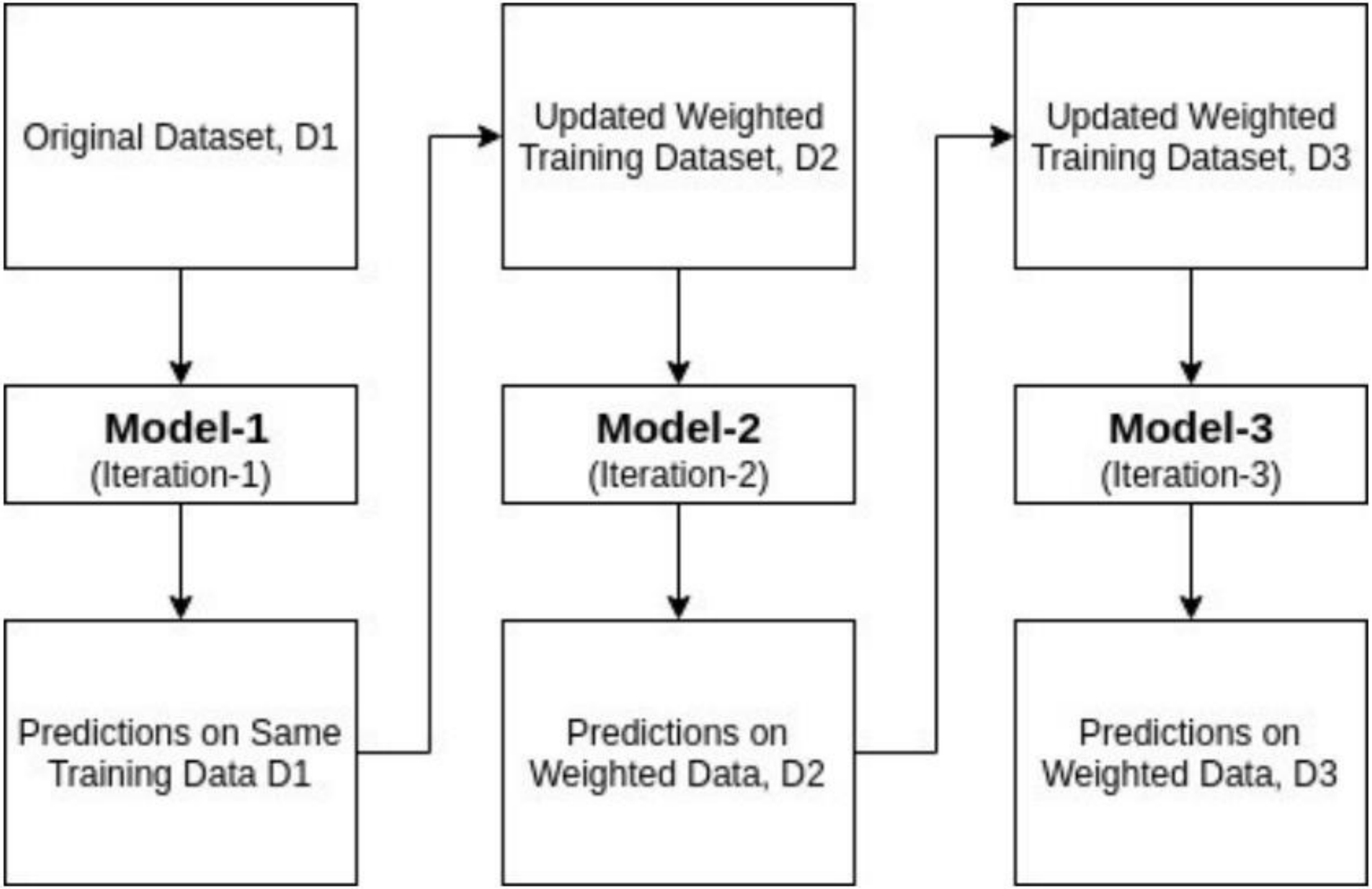
Boosting

- Models that are typically used in Boosting technique are:
  - XGBoost (Extreme Gradient Boosting)
  - GBM (Gradient Boosting Machine)
  - ADABoost (Adaptive Boosting)

Adaboost Summary

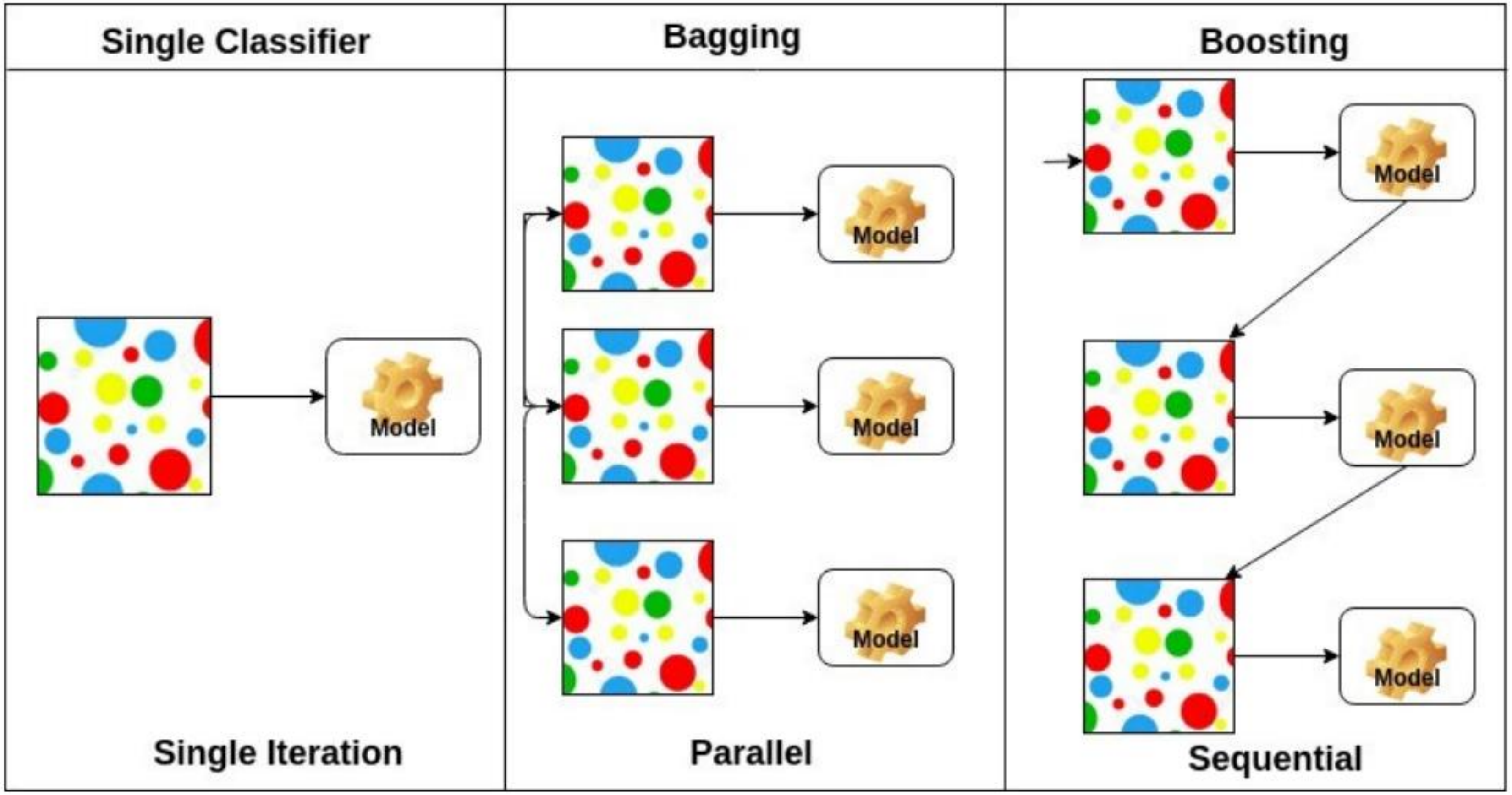
- Initially, Adaboost selects a training subset randomly.
- It iteratively trains the AdaBoost machine learning model by selecting the training set based on the accurate prediction of the last training.
- It assigns the higher weight to wrong classified observations so that in the next iteration these observations will get the high probability for classification.
- Also, It assigns the weight to the trained classifier in each iteration according to the accuracy of the classifier. The more accurate classifier will get high weight.
- This process iterate until the complete training data fits without any error or until reached to the specified maximum number of estimators.
- To classify, perform a "vote" across all of the learning algorithms you built.

Boosting



Original Dataset, D1
Updated Weighted Training Dataset, D2
Updated Weighted Training Dataset, D3
Model-1
(Iteration-1)
Model-2
(Iteration-2)
Model-3
(Iteration-3)
Predictions on Same Training Data D1
Predictions on Weighted Data, D2
Predictions on Weighted Data, D3

Boosting (Continued)

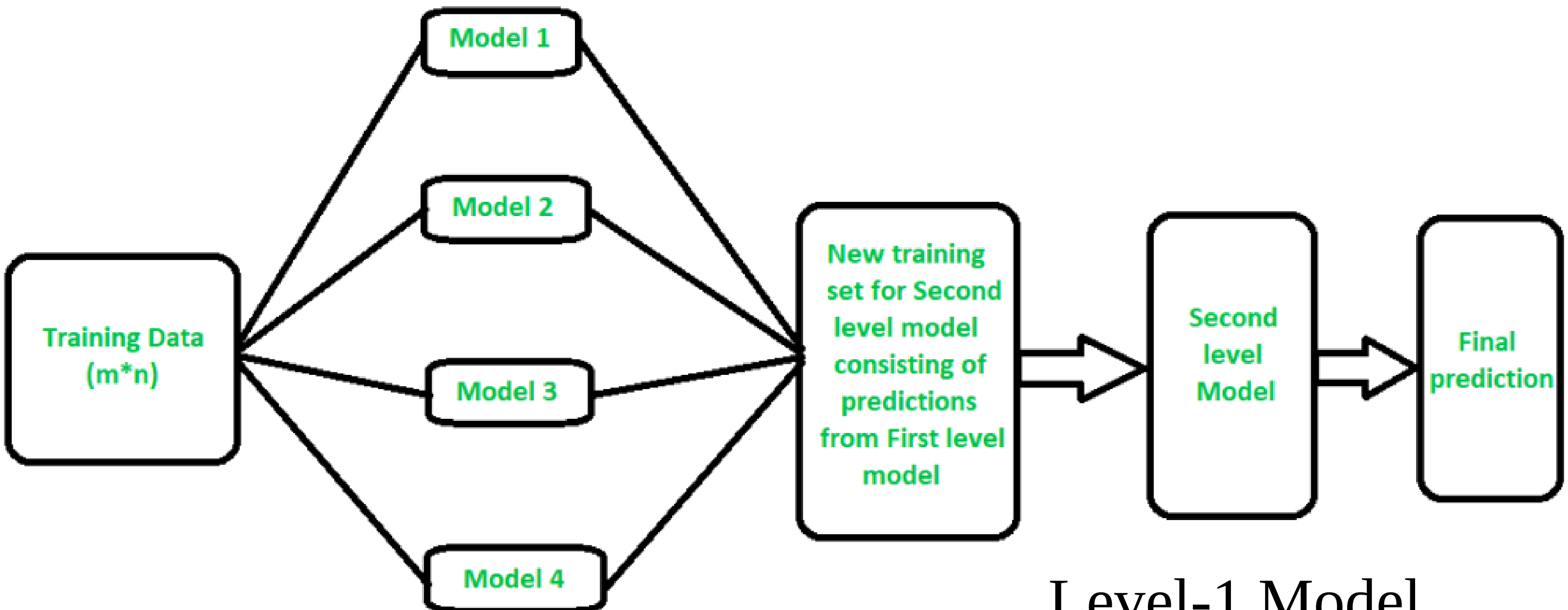


Stacking

- Stacked Generalization or "Stacking" for short is an ensemble machine learning algorithm.
- It involves combining the predictions from multiple machine learning models on the same dataset, like bagging and boosting. •
- Stacking addresses the question:
  - Given multiple machine learning models that are skillful on a problem, but in different ways, how do you choose which model to use (trust)?

Stacking II

- Unlike bagging, in stacking, the models are typically different (e.g. not all decision trees) and fit on the same dataset (e.g. instead of samples of the training dataset).
- Unlike boosting, in stacking, a single model is used to learn how to best combine the predictions from the contributing models (e.g. instead of a sequence of models that correct the predictions of prior models).



Level-1 Model
(Meta-Model)

Level-0 Models
(Base-Models)

Stacking Levels

- Level-0 Models (Base-Models): Models fit on the training data and whose predictions are compiled. provide the input and output pairs of the training dataset used to fit the meta-model.
- Level-1 Model (Meta-Model): Model that learns how to best combine the predictions of the base models.

Stacking levels

- The outputs from the base models used as input to the meta-model may be real value in the case of regression, and probability values, probability like values, or class labels in the case of classification.

Ref.

Tushar B. Kute,

Boosting Summary

- Good points
  - Fast learning
  - Capable of learning any function (given appropriate weak learner)
  - Feature weighting
  - Very little parameter tuning
- Bad points
  - Can overfit data
  - Only for binary classification
- Learning parameters (picked via cross validation)
  - Size of tree
  - When to stop
- Software
  - http://www-stat.stanford.edu/~jhf/R-MART.html