FUNDAMENTALS OF MACHINE LEARNING IN DATA SCIENCE

CSIS 3290

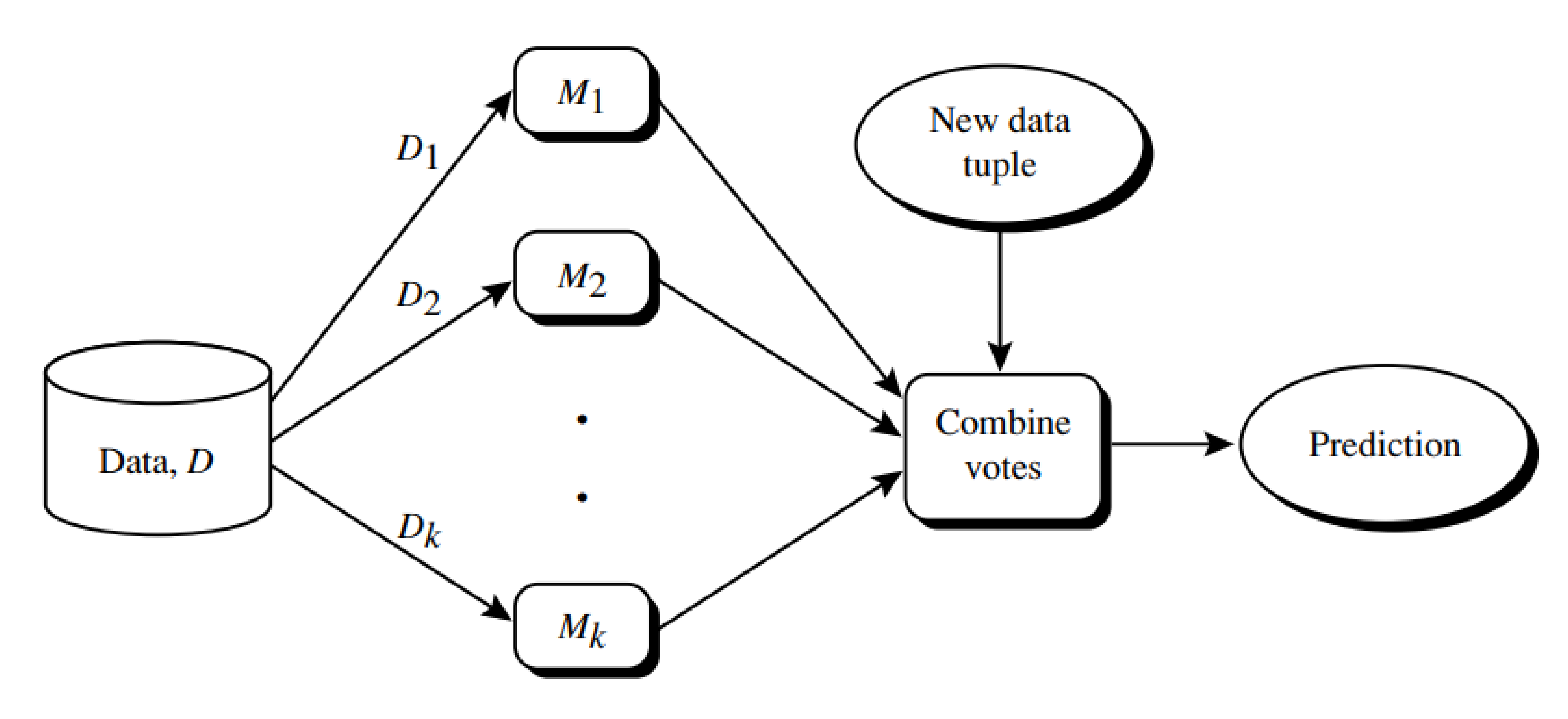
CLASSIFICATION: ENSEMBLE METHODS

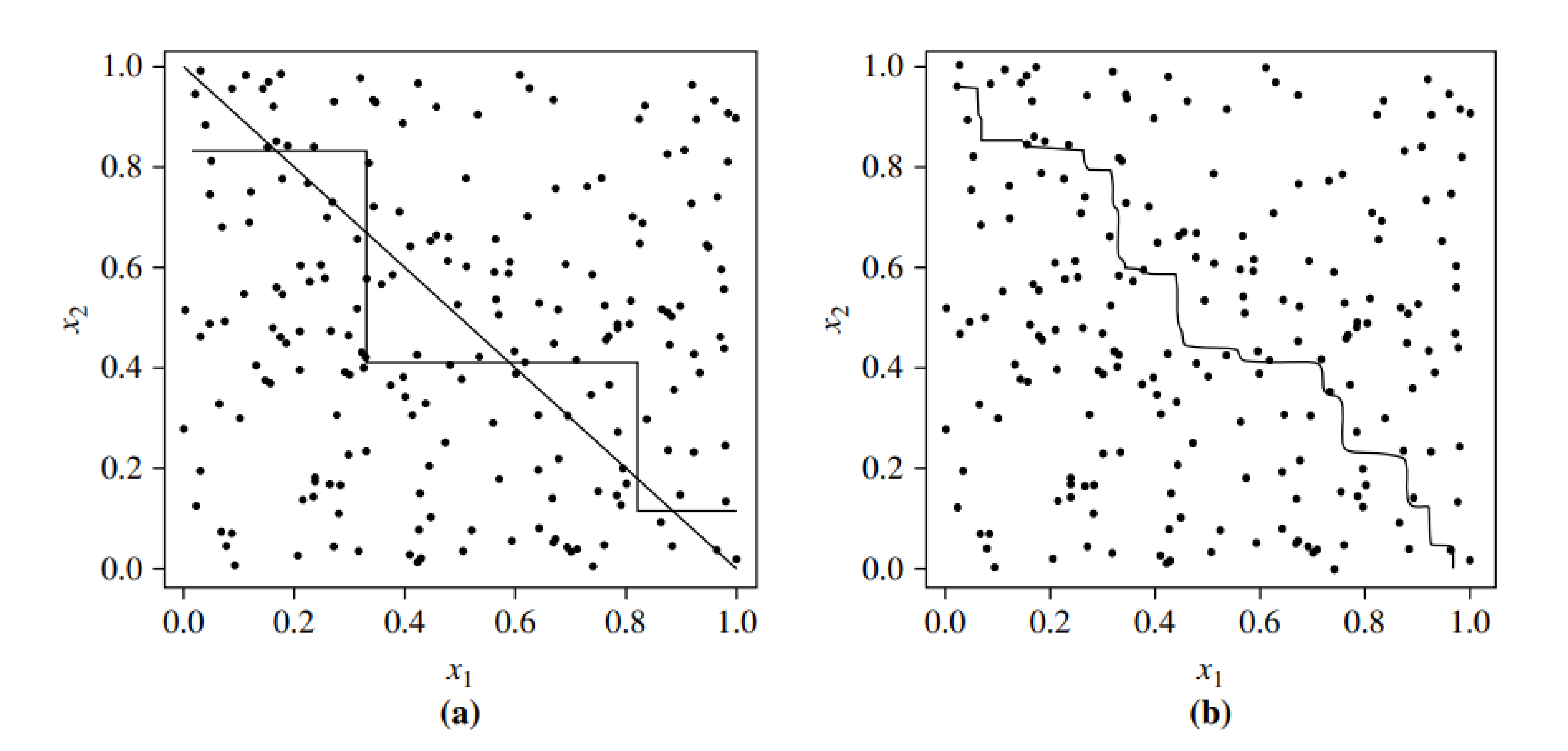
FATEMEH AHMADI

Ensemble Methods

- ✓ **Bagging**, **Boosting**, and **Random Forests** are examples **of Ensemble Methods**.
- ✓ An ensemble method combines a series of k learned models (or base classifiers), $M1$, $M2$,..., Mk , with the aim of creating an improved composite classification model, $M*$.
- ✓ A given data set, D, is used to create k training sets, $D1$, $D2$,..., Dk , where Di ($1 \leq i \leq k - 1$) is used to generate classifier Mi.
- ✓ Given a new data tuple to classify, the base classifiers each vote by returning a class prediction. The ensemble returns a class prediction based on the votes of the base classifiers.

Ensemble Methods



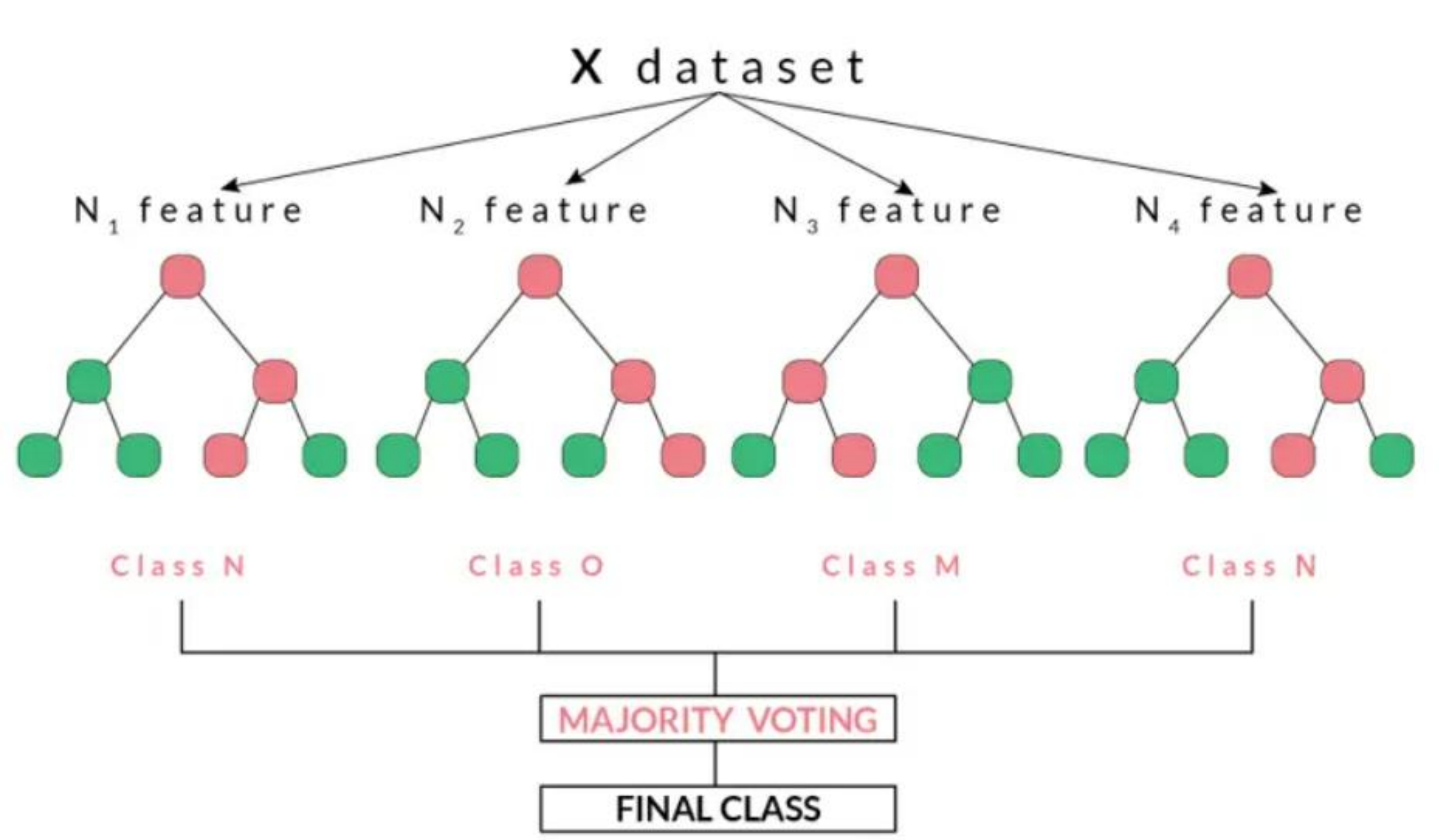


To help illustrate the power of an ensemble, consider **a simple two-class problem** described by two attributes, *x1* and *x2*. The problem has a **linear decision boundary**. Decision boundary by (a**) a single decision tree** and (b) an **ensemble of decision trees** for a **linearly separable problem** (i.e., where the actual decision boundary is a straight line). The decision tree struggles with approximating a linear boundary. The decision boundary of the ensemble is closer to the true boundary

Random Forest

- ✓ Imagine that each of the classifiers in the ensemble is a **decision tree classifier** so that the **collection of classifiers is a "forest**." The individual decision trees are generated **using a random selection of attributes at each node to determine the split**.
- ✓ More formally, each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest.
- ✓ During classification, each tree votes and the most popular class is returned.

Random Forest



Bagging

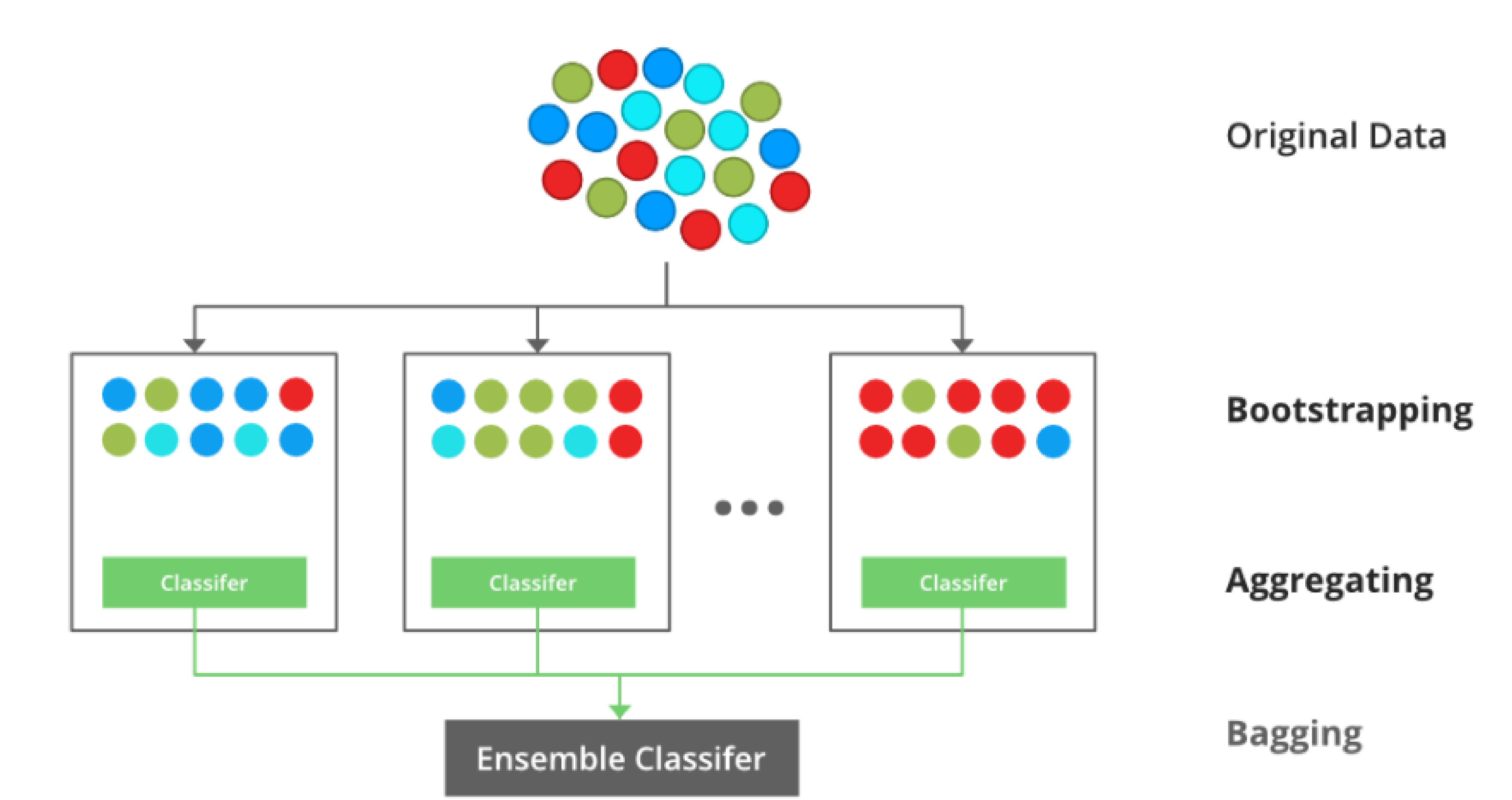
- ✓ Given a set, D, of d tuples, bagging works as follows. For iteration $i(i = 1, 2, \ldots, k)$, a training set, D_i , of d tuples is **sampled with replacement** from the original set of tuples, D.

- ✓ Note that the term **bagging stands for Bootstrap Aggregation**. Each training set is a bootstrap sample. Because sampling with replacement is used, some of the original tuples of D may not be included in D_i , whereas others may occur more than once. A classifier model, M_i , is learned for each training set, D_i.

- ✓ To classify an unknown tuple, X, each classifier, M_i , returns its class prediction, which counts as **one vote**. The bagged classifier, $M*$, counts the votes and assigns the class with the most votes to X.

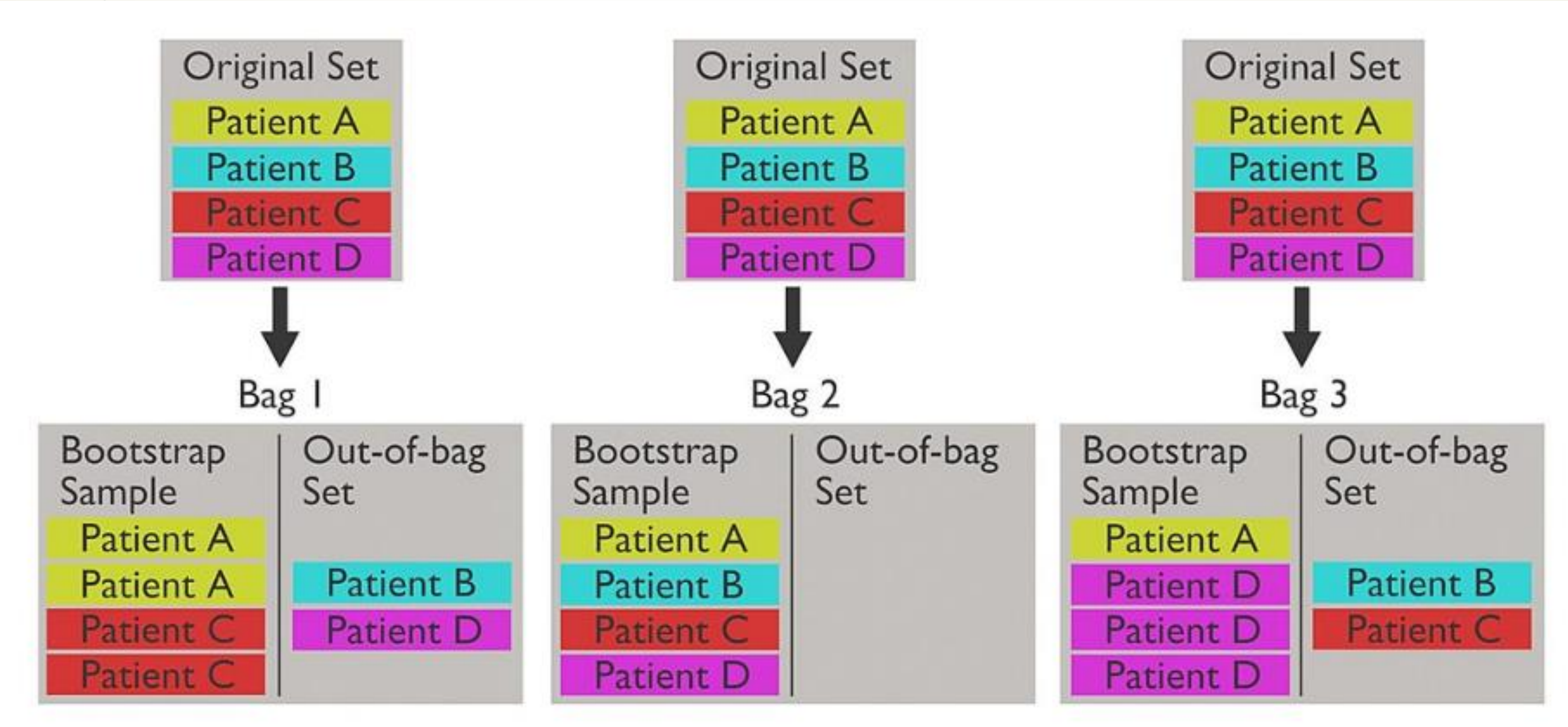
- ✓ **Bagging** can be applied to the prediction of continuous values by taking the average value of each prediction for a given test tuple.

Bagging



A **bootstrap sample** is a **smaller sample that is “bootstrapped” from a larger sample**. Bootstrapping is a type of resampling where large numbers of smaller samples of the same size are repeatedly drawn, **with replacement**, from a single original sample.

OOB Error Rate (Out-Of-Bag)



https://en.wikipedia.org/wiki/Out-of-bag_error#Out-of-bag_dataset

Boosting

- In **Boosting**, weights are also assigned to each training tuple. A series of k classifiers is iteratively learned.

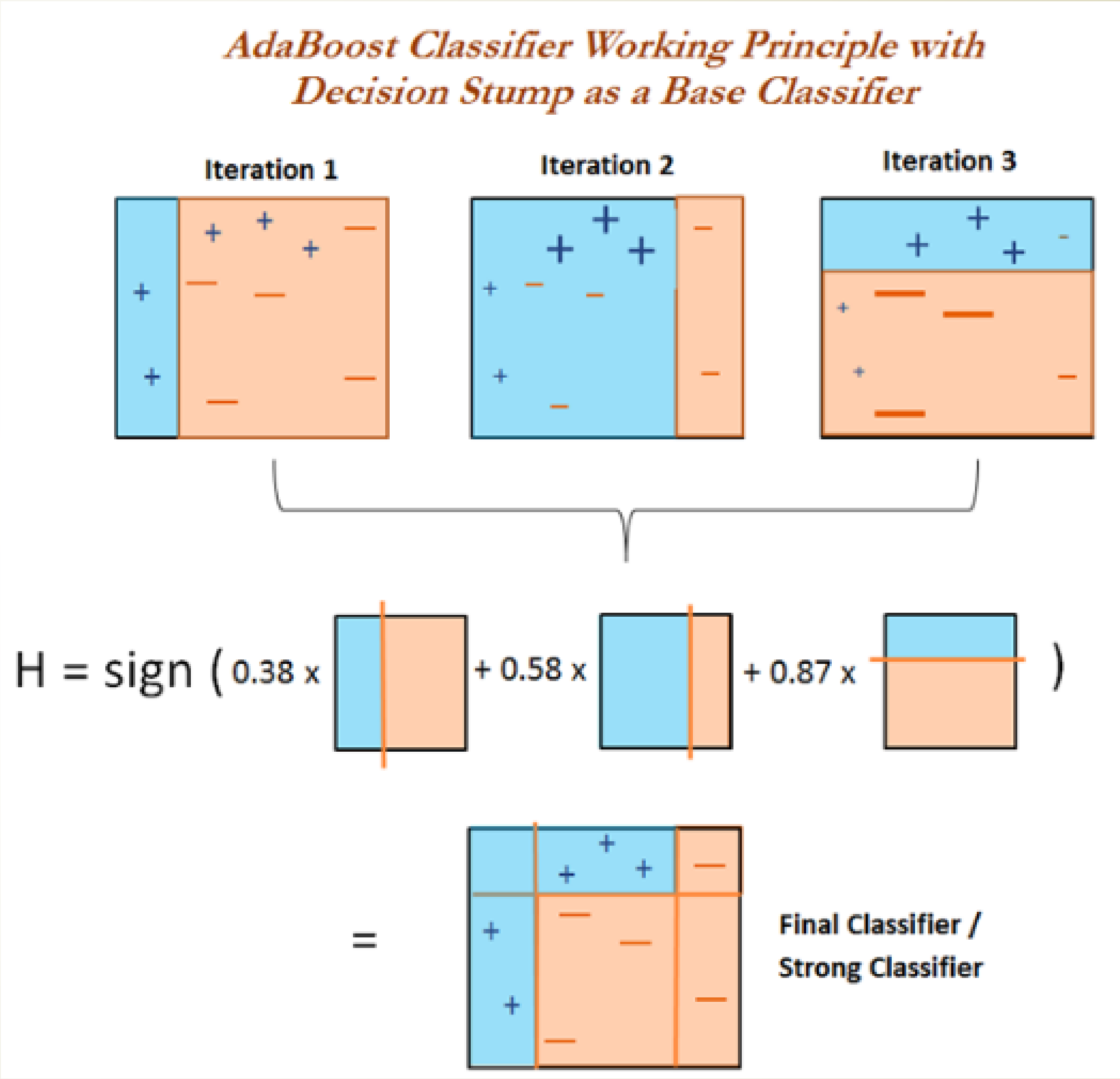
- After a classifier, M_i , is learned, the weights are updated to allow the subsequent classifier, M_{i+1}, to "pay more attention" to the training tuples that were misclassified by M_i.

- The final boosted classifier, $M*$, combines the votes of each individual classifier, where the weight of each classifier's vote is a function of its accuracy.

Ada Boost

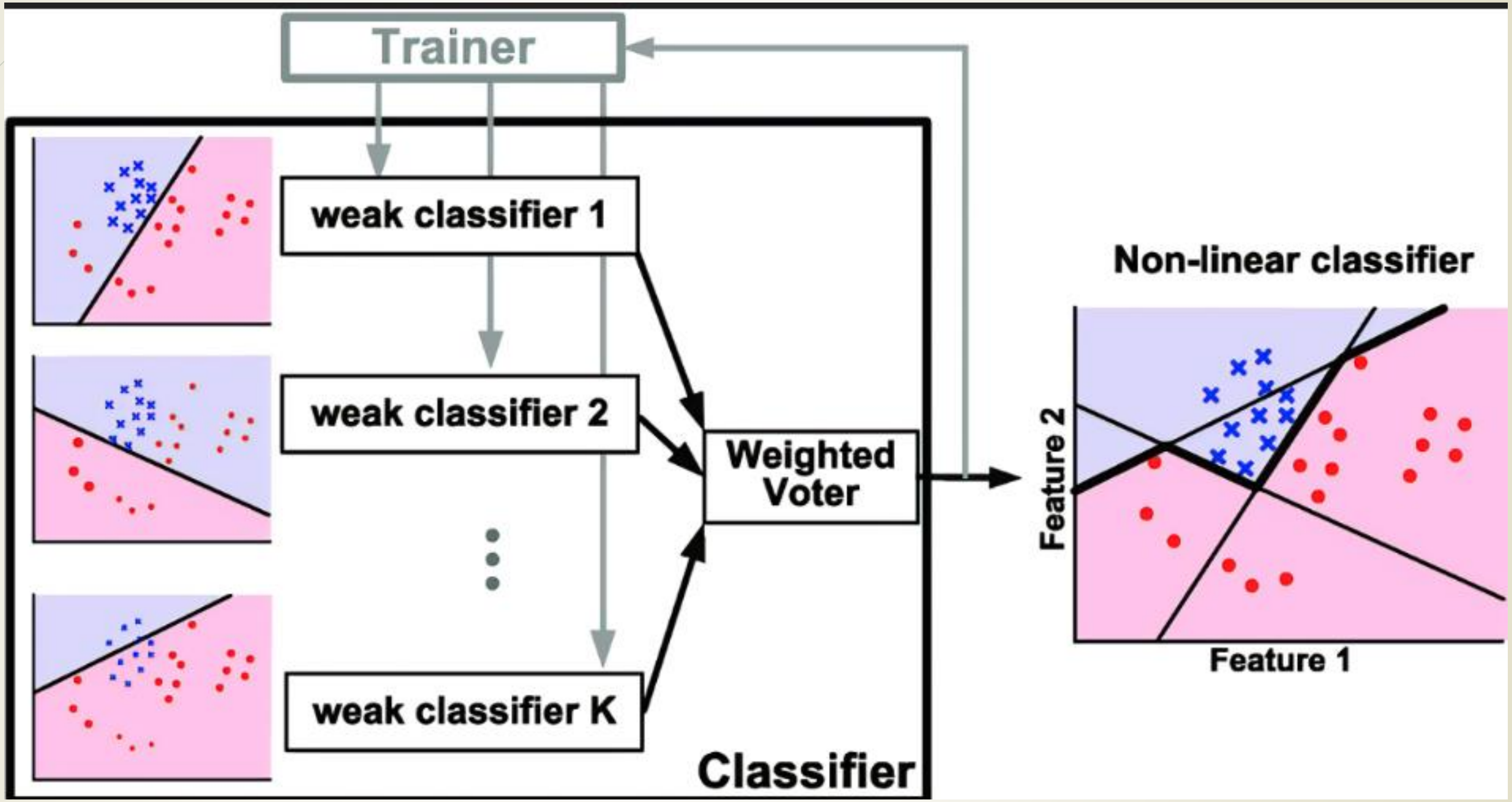
- **AdaBoost** (short for **Adaptive Boosti**ng) is a popular boosting algorithm. Suppose we want to boost the accuracy of a learning method. We are given *D*, a data set of *d* class-labeled tuples, $(X1, y1),(X2, y2),...,(Xd, yd)$, where yi is the class label of tuple Xi . Initially, AdaBoost assigns each training tuple **an equal weight of 1/d**.
- Generating *k* classifiers for the ensemble requires *k* rounds through the rest of the algorithm. In round *i*, the tuples from *D* are sampled to form a training set, *Di* , of size *d*. **Sampling with replacement** is used—the same tuple may be selected more than once. Each tuple's chance of being selected is based on its weight. A classifier model, *Mi*, is derived from the training tuples of *Di*. Its error is then calculated using *Di* as a test set. The weights of the training tuples are then adjusted according to how they were classified.
- If a tuple was incorrectly classified, its weight is increased. If a tuple was correctly classified, its weight is decreased. A tuple's weight reflects how difficult it is to classify— the higher the weight, the more often it has been misclassified.

Boosting



https://www.bing.com/images/search?view=detailV2&ccid=bU6AVMuL&id=D817DFE521B12BAACFD0DA8BB0879F78CD2A05E1&thid=OIP.bU6AVMuL1w4DsSkBwexjRwHaHG&mediaurl=https%3a%2f%2fstatic.packt-cdn.com%2fproducts%2f9781788295758%2fgraphics%2fimage_04_046-1.png&cdnurl=https%3a%2f%2fth.bing.com%2fth%2fid%2fR.6d4e8054cb8bd70e03b12901c1ec6347%3frik%3d4QUqzXifh7CL2g%26pid%3dImgRaw%26r%3d0&exph=594&expw=620&q=boosting+and+adaboost+in+machine+learning&simid=608047162160935230&FORM=IRPRST&ck=CA7A23785B7BD14CACF45EB7CD28BAEA&selectedIndex=3&ajaxhist=0&ajaxserp=0

Boosting



https://www.bing.com/images/search?view=detailV2&ccid=H6baP7Og&id=7692F611DA9A66EF23322EA543CE21DFB3820D22&thid=OIP.H6baP7OgjLma4oMRlTEqzgHaD7&mediaurl=https%3a%2f%2fwww.researchgate.net%2fprofile%2fZhuo-Wang-36%2fpublication%2f288699540%2ffigure%2ffig9%2fAS%3a668373486686246%401536364065786%2fIllustration-of-AdaBoost-algorithm-for-creating-a-strong-classifier-based-on-multiple.png&cdnurl=https%3a%2f%2fth.bing.com%2fth%2fid%2fR.1fa6da3fb3a08cb99ae2831195312ace%3frik%3dIg2Cs98hzkOlLg%26pid%3dImgRaw%26r%3d0%26sres%3d1%26sresct%3d1%26srh%3d688%26srw%3d1300&exph=372&expw=702&q=boosting+and+adaboost+in+machine+learning&simid=607992470065583027&FORM=IRPRST&ck=E95E58CB9898E7BB324309BD4913D9DA&selectedIndex=4&ajaxhist=0&ajaxserp=0

Reference

Data Mining, Concepts and Techniques, Jiawei Han, Micheline Kamber, Jian Pei. MK. Chapter 8.