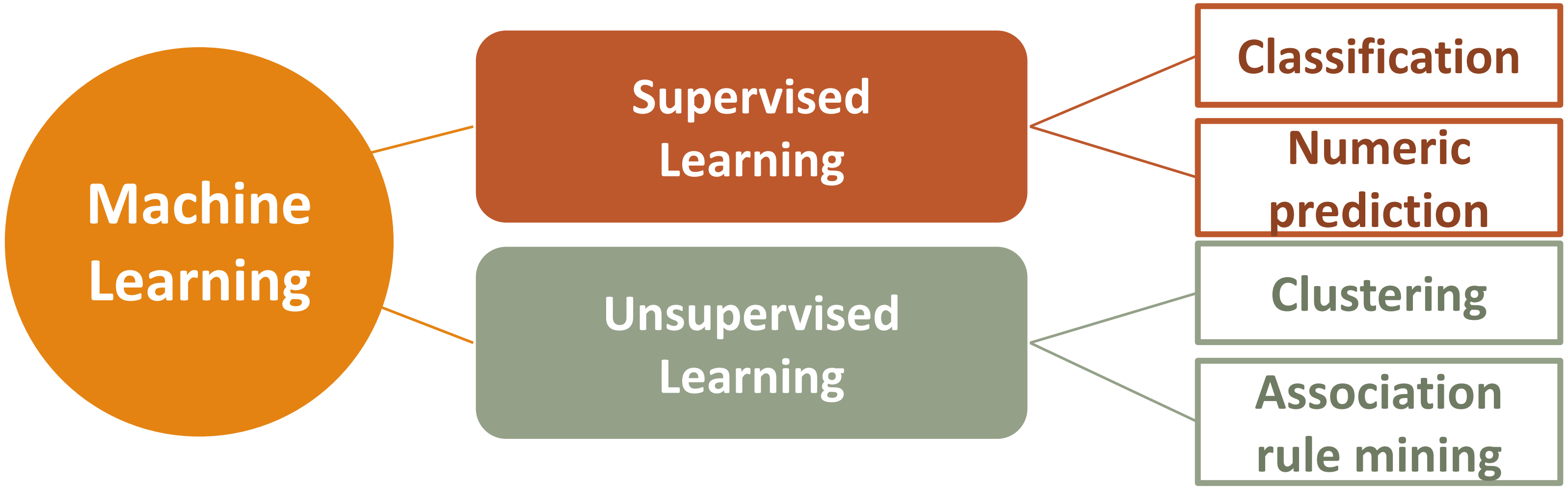
Lecture 11: Improving Model Performance

Data Mining Tasks



Data Mining Tasks

Model	Learning Task
Decision Trees	Classification
Naive Bayes	Classification
k-Nearest Neighbors	Classification
Linear Regression	Numeric Prediction
Regression Trees	Numeric Prediction
Model Trees	Numeric Prediction
Neural Networks	Classification/Numeric Prediction
Support Vector Machines	Classification/Numeric Prediction
K-means	Clustering
Apriori	Association Rule Mining

Outline

- Improve model performance with ensembles
  - Understanding ensembles
  - Bagging
  - Boosting
  - Random Forest

Improve Model Performance with Ensembles

- Netflix Prize
  - The Netflix Prize was an open competition for the best collaborative filtering algorithm to predict user ratings for films, based on previous ratings.
    - The competition was held by Netflix, an online DVD-rental and video streaming service, and the grand prize is US $1,000,000.
  - Business values:
    - Netflix has around 150 million subscribers globally.
    - Netflix claims its AI assisted recommendation system saves the company $1 billion per year.

Improve Model Performance with Ensembles

- How can we improve the model performance?

Improve Model Performance with Ensembles

- A model brings a unique bias to a learning task, it may readily learn one subset of examples, but have trouble with another.
  - Highly complex model, Clustering + classification, Ensemble-learning
- Combine several models to form a powerful team
  - Sports teams have players with complementary rather than overlapping skillsets
  - Machine learning algorithms utilize teams of complementary models

Understanding Ensembles

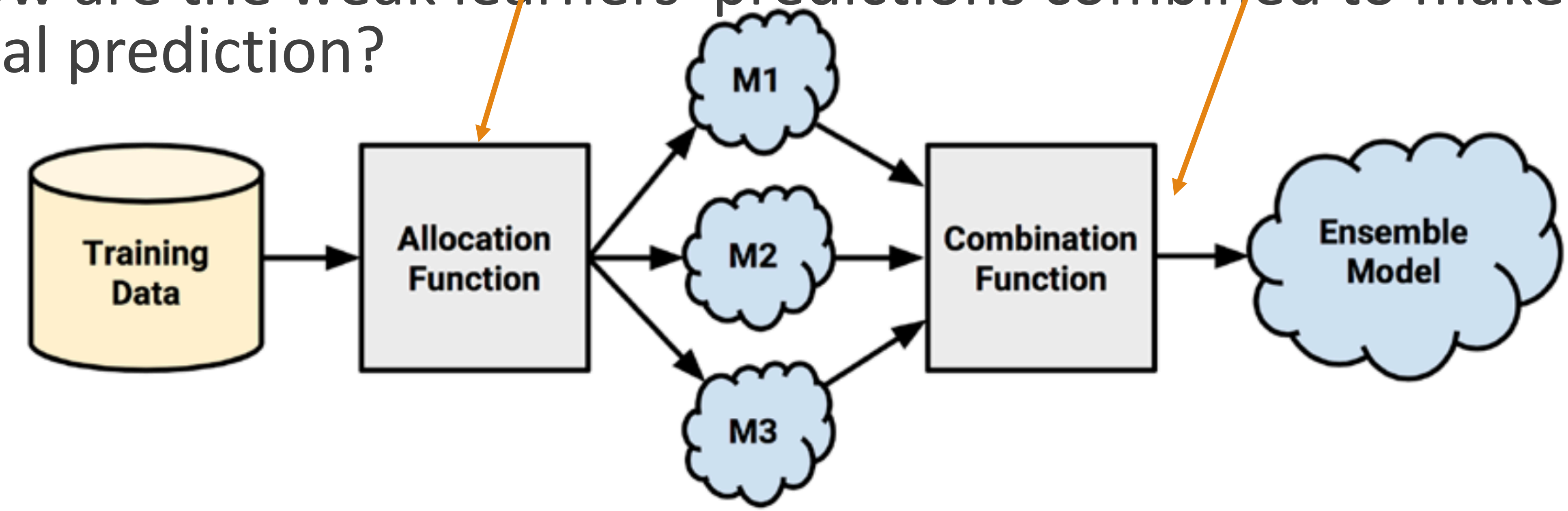
- Suppose you are a movie director and you have created a short movie on a very important and interesting topic. Now, you want to take preliminary feedback (ratings) on the movie before making it public. What are the possible ways you can do?

Improve Model Performance with Ensembles

- Ensembles methods: technique of combining and managing the predictions of multiple models.
  - Bagging
  - Boosting
  - Random Forest

Understanding Ensembles

- **Ensemble** method is based on the idea that by combining multiple weaker learners, a stronger learner is created.
  - How are the weak learning models chosen and/or constructed?
  - How are the weak learners' predictions combined to make a single final prediction?



Understanding Ensembles

- Ideal ensemble includes a diverse set of models, the **allocation function** can increase diversity by artificially varying the input data to bias the resulting learners.
- The **allocation function** dictates
  - How much of the training data each model receives.
  - Do they each receive the full training dataset or merely a sample?
  - Do they each receive every feature or a subset?

Understanding Ensembles

- After the models are constructed, they can be used to generate a set of predictions, which must be managed in some way.
- The **combination function** governs how disagreements among the predictions are reconciled.
  - Majority vote
  - Weighting each model's votes
  - Utilize another model to learn a combination function from various combinations of predictions.

Understanding Ensembles

- One of the benefits of using ensembles is that they may allow you to spend less time in pursuit of a single best model.
- Better generalizability to future problems
  - As the opinions of several learners are incorporated into a single final prediction, no single bias is able to dominate.
- Capture subtle patterns that a single global model might miss

Ensemble Method: Bagging

Ensemble Method: Bagging

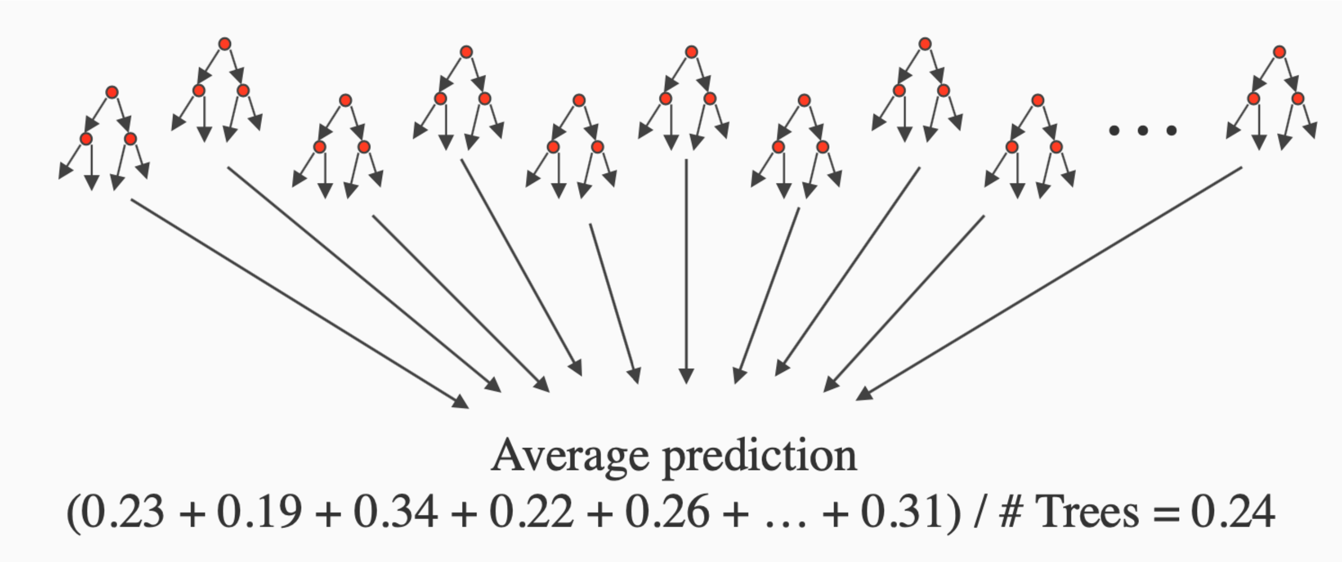
- Bagging generates a number of training datasets by bootstrap sampling the original training data.
  - Bagging perform quite well as long as it is used with relatively **unstable** learners
    - Unstable models are essential in order to ensure the ensemble's diversity in spite of only minor variations between the bootstrap training datasets

Ensemble Method: Bagging

- **Bootstrap sampling** refer to the statistical methods of using random samples of data to estimate the properties of a larger set.
  - The creation of several randomly selected training and test datasets, which are then used to estimate performance statistics
- Differences between bootstrap sampling and cross validation
  - Bootstrap allows examples to be selected multiple times through a process of **sampling with replacement.**
  - In cross validation, each example/instance will be selected for test exact once.

Ensemble Method: Bagging

- Bagged Decision Trees
  - Draw T bootstrap samples of data
  - Train trees on each sample
  - Average prediction of trees on out-of-bag samples



Ensemble Method: Boosting

Ensemble Method: Boosting

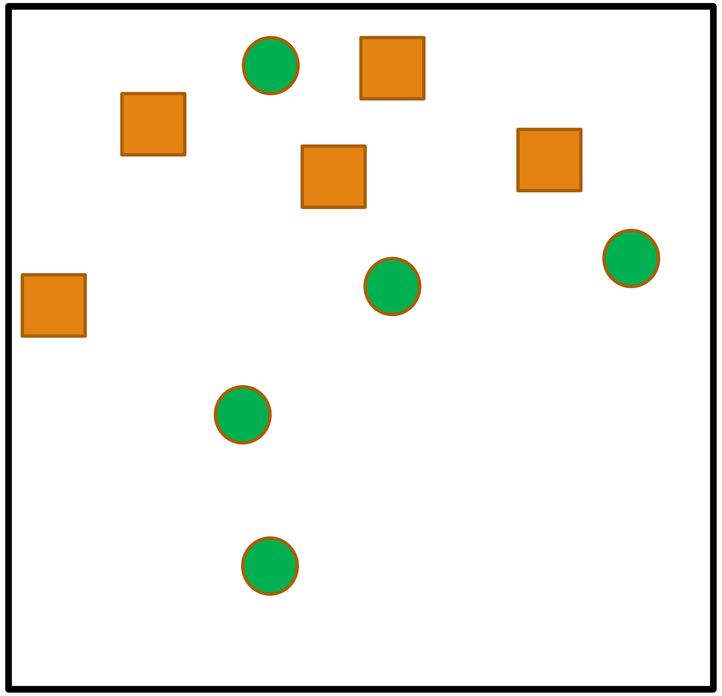
- Another common ensemble-based method is called **boosting** because it boosts the performance of weak learners to attain the performance of stronger learners.

Ensemble Method: Boosting

- **AdaBoost** or **adaptive boosting**
  - The algorithm is based on the idea of generating weak learners that iteratively learn a larger portion of the difficult-to-classify examples by paying more attention (that is, giving more weight) to frequently misclassified examples.

Ensemble Method: Boosting

- **AdaBoost** or **adaptive boosting**

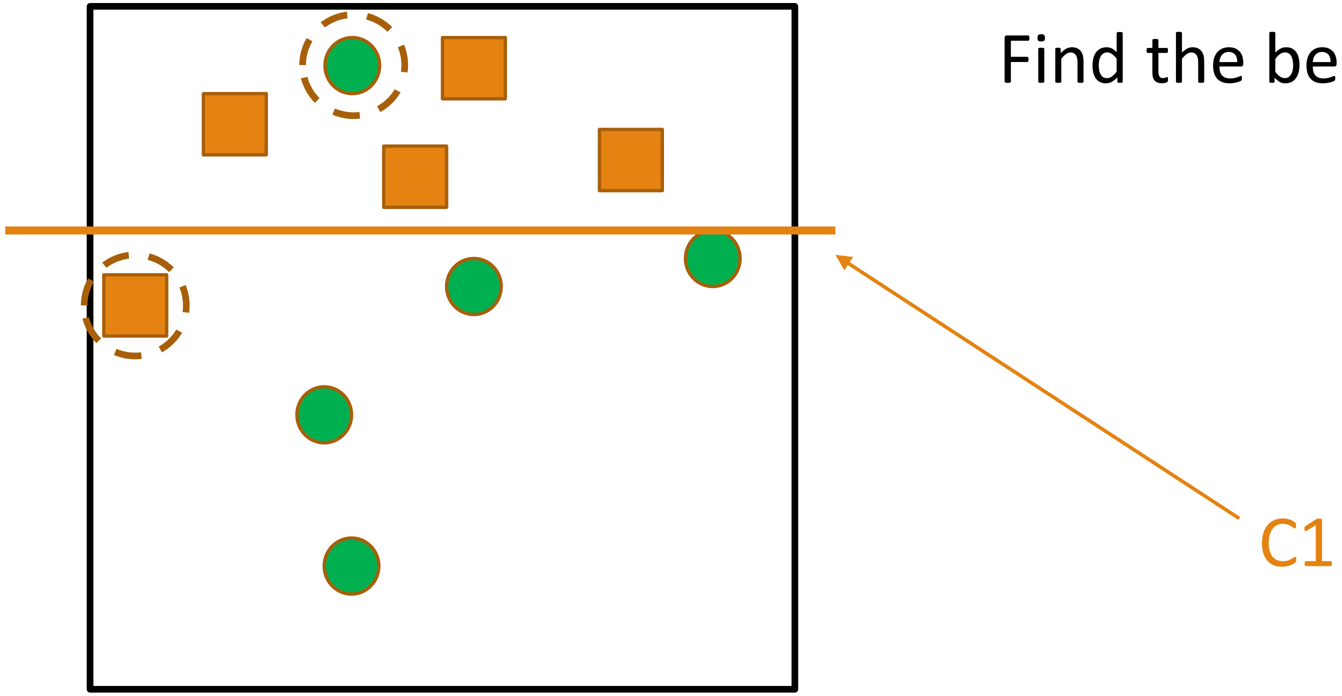


Each example/instance has the same importance

Weak Leaner: an axis parallel line

Ensemble Method: Boosting

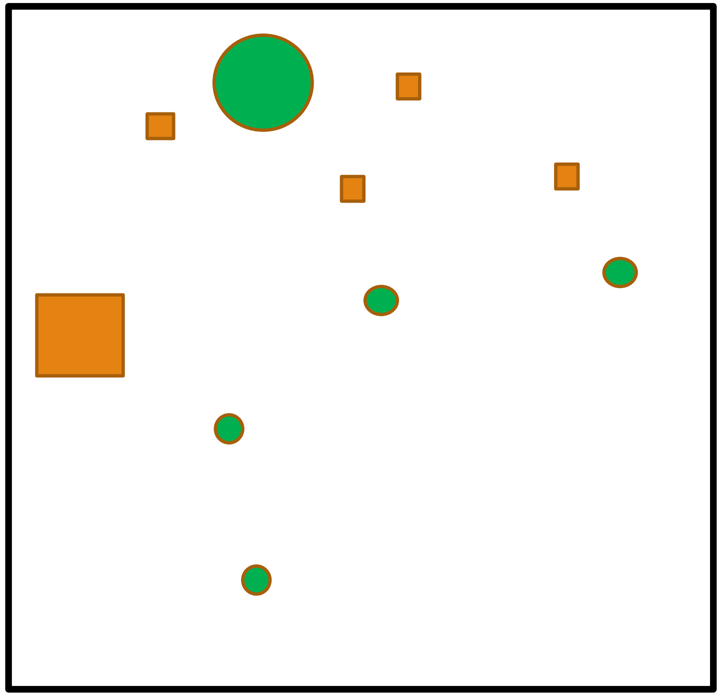
- **AdaBoost** or **adaptive boosting**



Find the best classifier on the dataset

Ensemble Method: Boosting

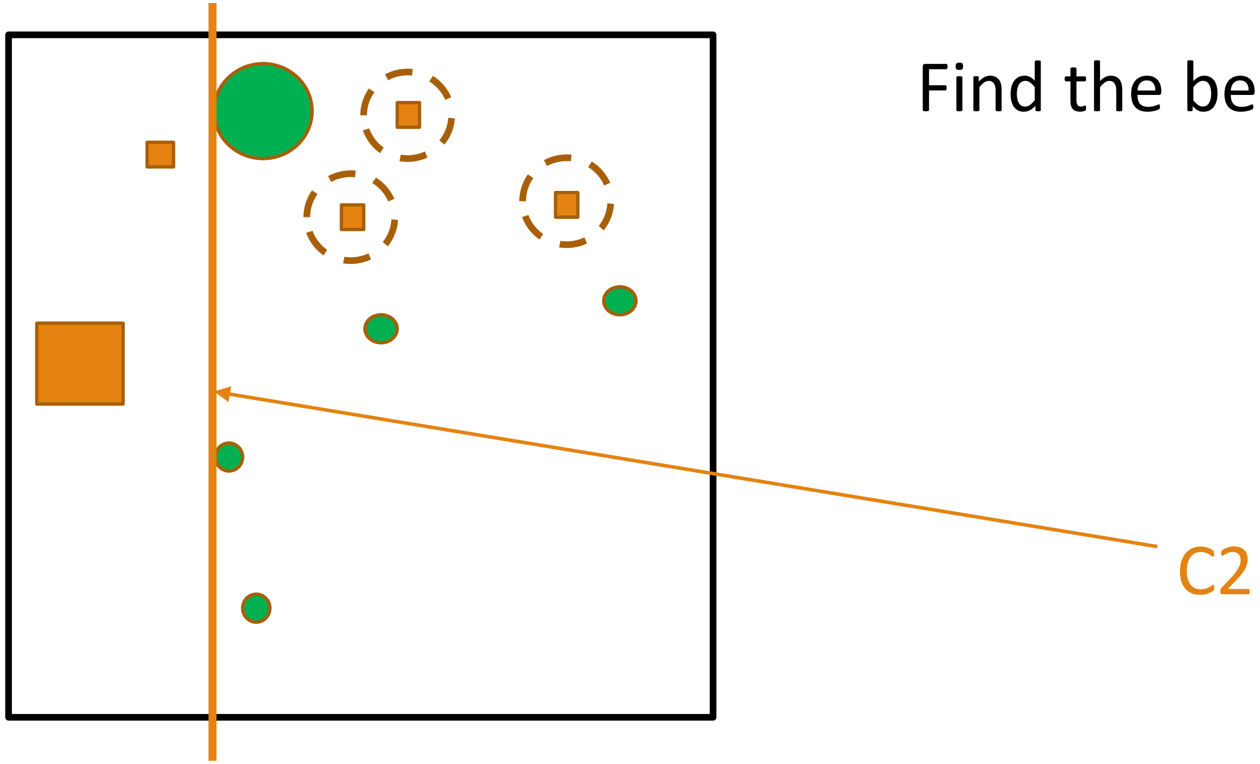
- **AdaBoost** or **adaptive boosting**



Increase the importance of the examples with mistakes and down-weight the examples that got correctly

Ensemble Method: Boosting

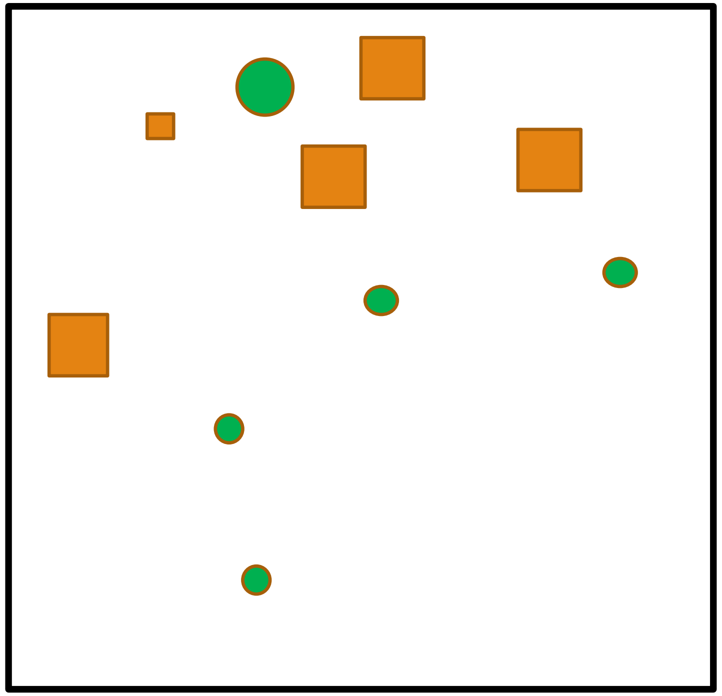
- **AdaBoost** or **adaptive boosting**



Find the best classifier on the dataset

Ensemble Method: Boosting

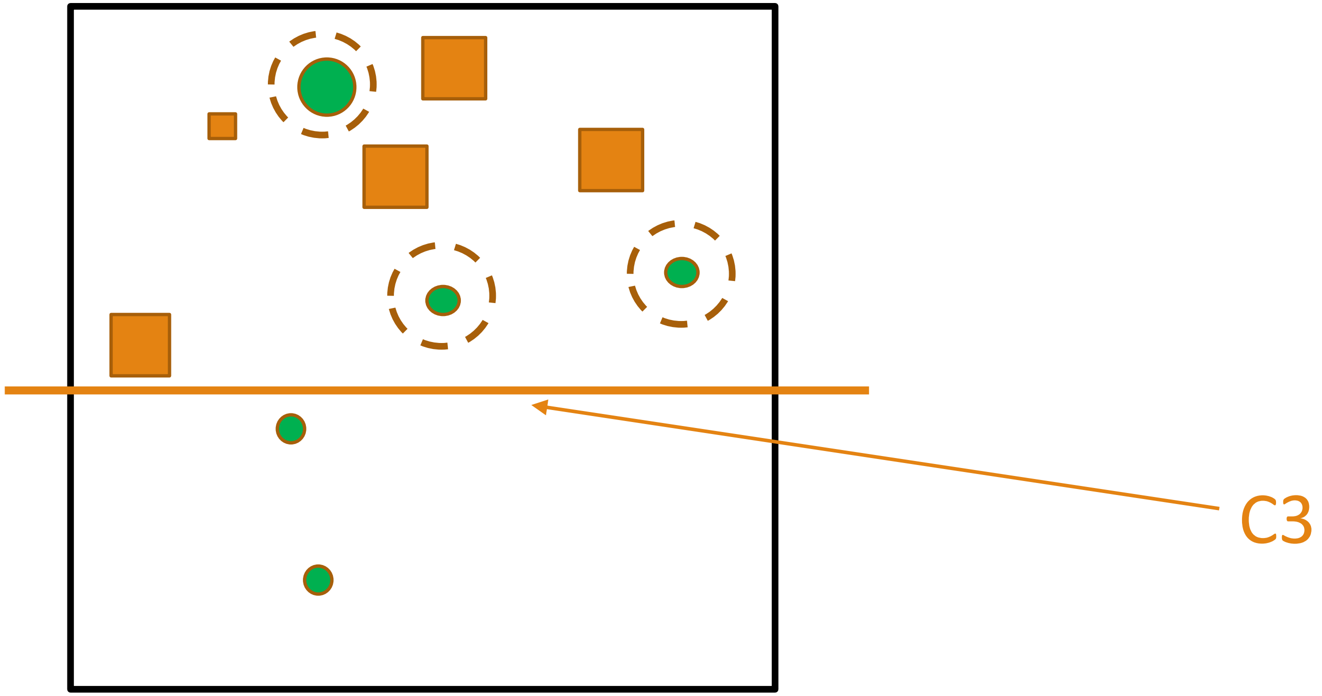
- **AdaBoost** or **adaptive boosting**



Increase the importance of the examples with mistakes and down-weight the examples that got correctly

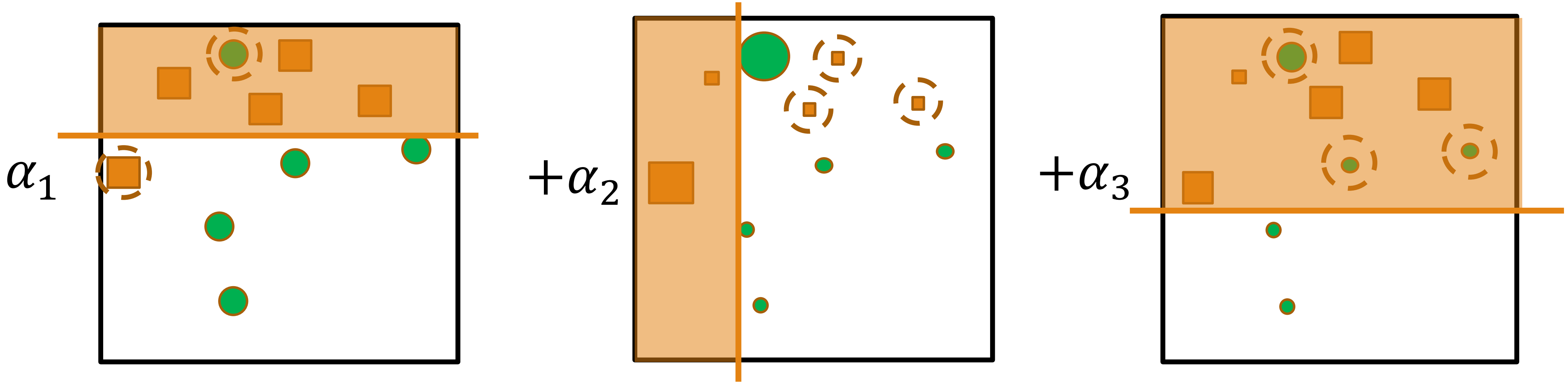
Ensemble Method: Boosting

- **AdaBoost** or **adaptive boosting**



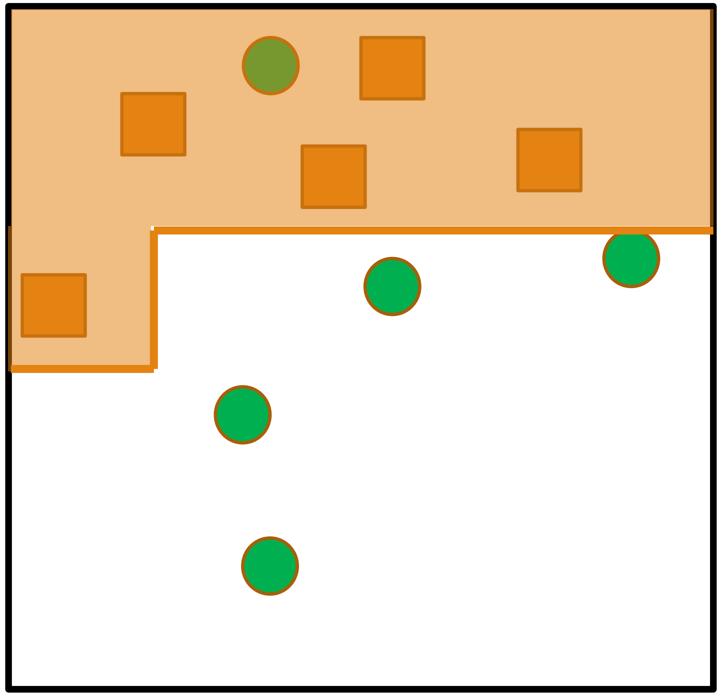
Ensemble Method: Boosting

- **AdaBoost** or **adaptive boosting**



Ensemble Method: Boosting

- **AdaBoost** or **adaptive boosting**



Ensemble Method: Boosting

- Initialization
  - Weigh all training samples equally
- Iteration Steps
  - Train model on (weighted) training set
  - Compute error of model on training set
  - Increase weights on training cases gets wrong
- Return final model
  - Carefully weighted prediction of each model

Ensemble Method: Random Forests

Random Forests

- Another ensemble-based method called **random forests** (or decision tree forests) focuses only on ensembles of decision trees.
- This method combines the base principles of bagging with random feature selection to add additional diversity to the decision tree models.
- After the ensemble of trees (the forest) is generated, the model uses a vote to combine the trees' predictions.

Random Forests

- As the ensemble uses only a small, random portion of the full feature set, random forests can handle extremely large datasets, where the so-called "curse of dimensionality" might cause other models to fail.

Random Forests

- Relative to other ensemble-based methods, random forests are quite competitive and offer key advantages relative to the competition.

Strengths	Weaknesses
• An all-purpose model that performs well on most problems • Can handle noisy or missing data as well as categorical or continuous features • Selects only the most important features • Can be used on data with an extremely large number of features or examples	• Unlike a decision tree, the model is not easily interpretable • May require some work to tune the model to the data