* Definition: Construct a set of classifiers from the training data, Exploit “diversity” in data, models, parameters. It predicts class label of a test case by aggregating predictions made by multiple classifiers and let them vote!
* A diagram of a training data set

  Description automatically generated

Generating a set of base classifiers

* Pro: Often reduce variance, reduce bias by improving predictive performance
* Con: Usually produces output that is difficult to understand
* It works well when base classifiers are independent each other, base classifiers perform better than random guessing.
* “A set of base classifiers may be generated by”
  + Sampling training examples: Train k classifiers on k subsets drawn from the
  + training set
  + Using different learning models: Use all the training examples, but apply different learning algorithms
  + Sampling features: Train k classifiers on k subsets of features drawn from the feature space
* Manipulating training sets:

This approach would be

helpful when your model

is complex, easy to overfit, such as decision tree

* + Bagging (Bootstrap Aggregation): Several training sets with the same size; Generating new datasets of size n by sampling from the original dataset with replacement (Bootstrap sample); Combining predictions by voting/averaging Each model receives equal weight; Can be applied to regression (AVG) and classification (VOTING)
  + Boosting: Train classifiers in a sequence; A new classifier should focus on those cases which were misclassified in the previous round — hard examples; Combine the classifiers on the final prediction (like bagging) but each base classifier has a weight.
  + Example: Ada Boosting, Initially, set uniform weights on all the records; At each round, create a bootstrap sample based on the weights, train a classifier on the sample and apply it on the original training set, records that are wrongly classified will have their weights increased, records that are classified correctly will have their weights decreased, If the error rate is higher than 50%, start over; Finally, combining all the classifiers
* Manipulating input features:
  + Random Forest: Bagging + Random Features
  + Improved Accuracy: Incorporate more diversity and reduce variances
  + The key idea is to build numerous decision trees during training and output the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random forests introduce extra randomness when growing trees; instead of searching for the most important feature while splitting a node, it searches for the best feature among a random subset of features.