INTRO TO DATA SCIENCE

DECISION TREES & RANDOM FORESTS

Decision Trees

Ensemble Methods

Random Forests

DECISION TREES

DECISION TREES: CHARACTERISTICS

- Non-parametric:
  - no parameters, no distribution assumptions
- Hierarchical:
  - consists of a sequence of questions which yield a class label when applied to any record
- Variable Size:
  - Any boolean functions can be represented
- Deterministic:
  - For the same set of features the tree will assign the same label
- Discrete and Continuous Parameters:
  - Binning and Threshold

Table 4.1. The vertebrate data set.

Name	Body Temperature	Skin Cover	Gives Birth	Aquatic Creature	Aerial Creature	Has Legs	Hiber-nates	Class Label
human	warm-blooded	hair	yes	no	no	yes	no	mammal
python	cold-blooded	scales	no	no	no	no	yes	reptile
salmon	cold-blooded	scales	no	yes	no	no	no	fish
whale	warm-blooded	hair	yes	yes	no	no	no	mammal
frog	cold-blooded	none	no	semi	no	yes	yes	amphibian
komodo dragon	cold-blooded	scales	no	no	no	yes	no	reptile
bat	warm-blooded	hair	yes	no	yes	yes	yes	mammal
pigeon	warm-blooded	feathers	no	no	yes	yes	no	bird
cat	warm-blooded	fur	yes	no	no	yes	no	mammal
leopard shark	cold-blooded	scales	yes	yes	no	no	no	fish
turtle	cold-blooded	scales	no	semi	no	yes	no	reptile
penguin	warm-blooded	feathers	no	semi	no	yes	no	bird
porcupine	warm-blooded	quills	yes	no	no	yes	yes	mammal
eel	cold-blooded	scales	no	yes	no	no	no	fish
salamander	cold-blooded	none	no	semi	no	yes	yes	amphibian

DECISION TREES

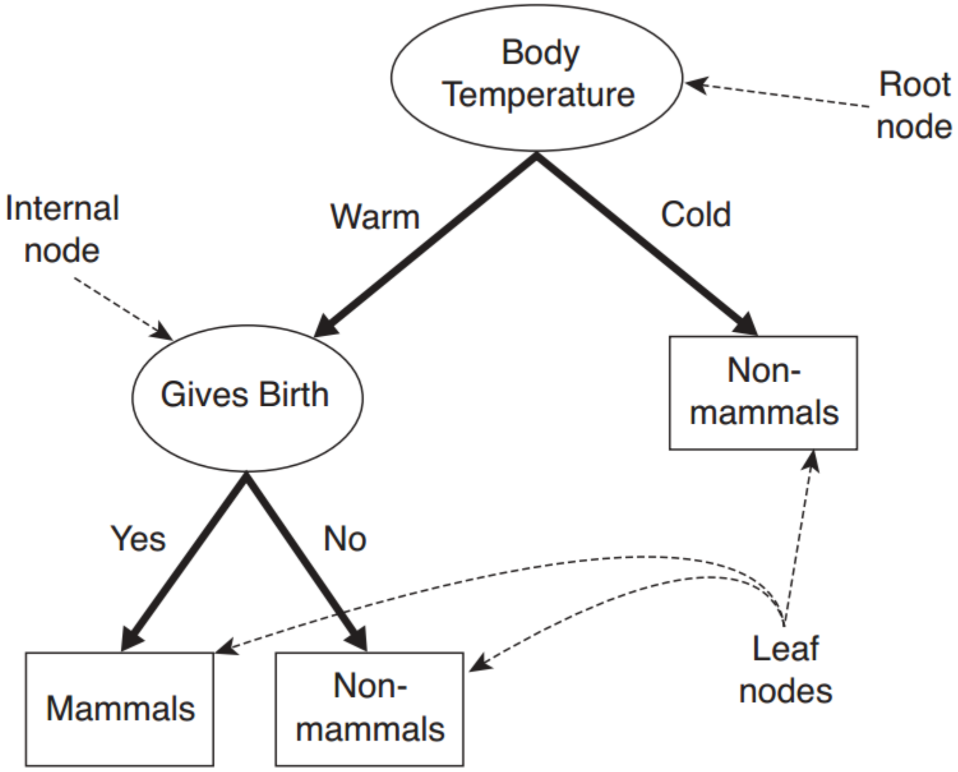
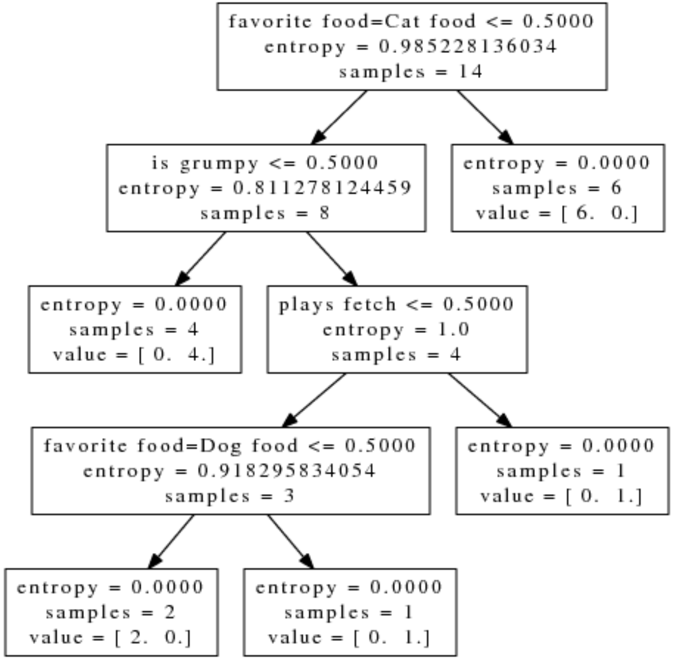


Figure 4.4. A decision tree for the mammal classification problem.

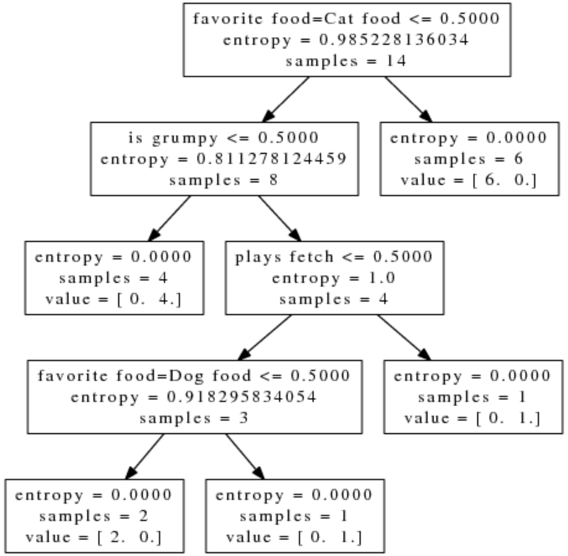
DECISION TREES

- **Decision trees** are tree-like graphs that can be used as non-linear models in classification and regression tasks.



DECISION TREES

- The branches of a decision tree specify the shortest sequences of explanatory variables that can be examined in order to estimate the value of a response variable.



20 QUESTIONS

- The **answerer** chooses an object but does not reveal the object to the questioners.

- The object should be a common noun, such as “guitar” or “sandwich”, but not “1969 Gibson Les Paul Custom” or “North Carolina.”

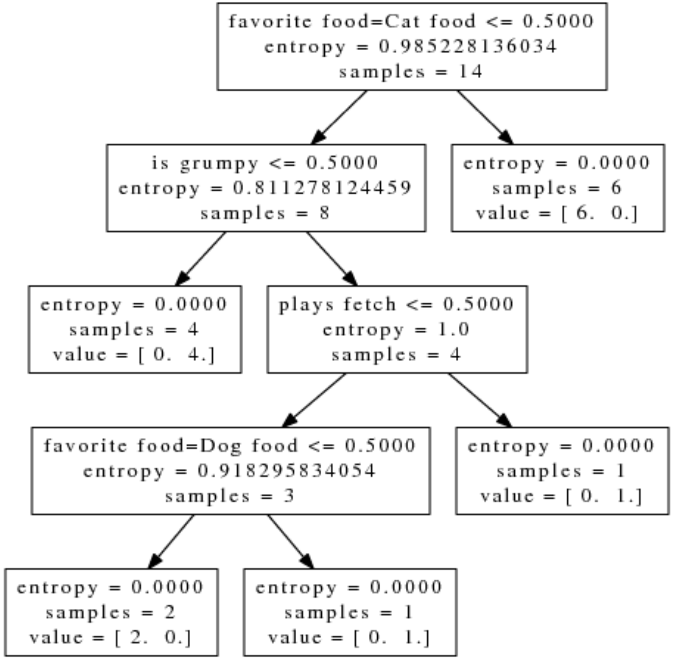
- The **questioners** must guess the object by asking as many as twenty questions that can be answered with “yes,” “no,” or “maybe.”

DECISION TREES

- Decision trees are learned by recursively splitting the set of training instances into subsets based on instances' values for the explanatory variables.

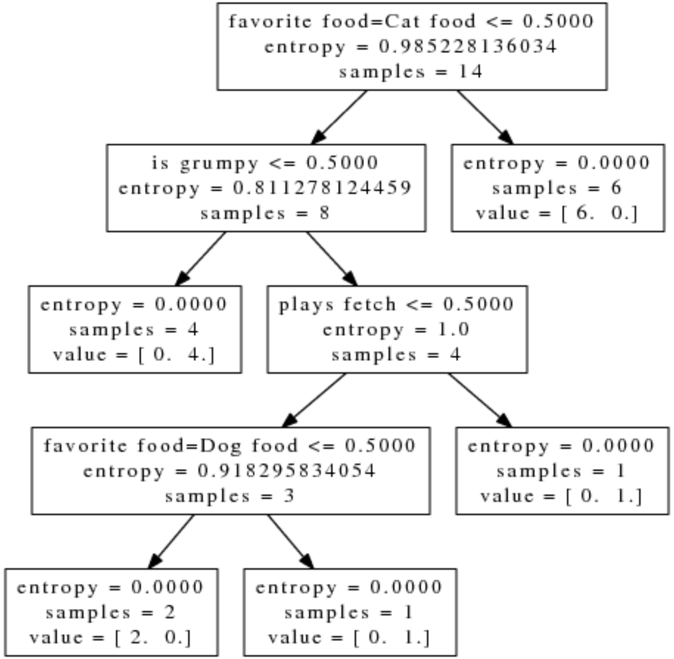
DECISION TREES

- Each interior node of a decision tree tests an explanatory variable.



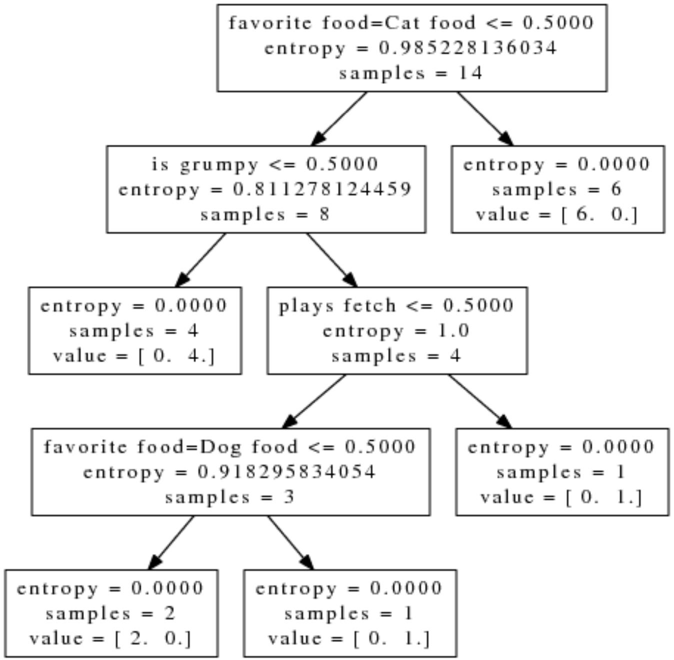
DECISION TREES

- These nodes are connected by edges that specify the possible outcomes of the tests.



DECISION TREES

- The training instances are divided into subsets based on the outcomes of the tests.



DECISION TREES: ALGORITHMS

What are all the various decision tree algorithms and how do they differ from each other?

ID3, C4.5, C5.0 and CART

Scikit-learn uses an optimised version of the CART algorithm.

DECISION TREES: ALGORITHMS

ID3 (Iterative Dichotomiser 3) was developed in 1986 by Ross Quinlan. The algorithm creates a multiway tree, finding for each node (i.e. in a greedy manner) the categorical feature that will yield the largest information gain for categorical targets. Trees are grown to their maximum size and then a pruning step is usually applied to improve the ability of the tree to generalise to unseen data.

DECISION TREES: ALGORITHMS

C4.5 is the successor to ID3 and removed the restriction that features must be categorical by dynamically defining a discrete attribute (based on numerical variables) that partitions the continuous attribute value into a discrete set of intervals. C4.5 converts the trained trees (i.e. the output of the ID3 algorithm) into sets of if-then rules. These accuracy of each rule is then evaluated to determine the order in which they should be applied. Pruning is done by removing a rule's precondition if the accuracy of the rule improves without it.

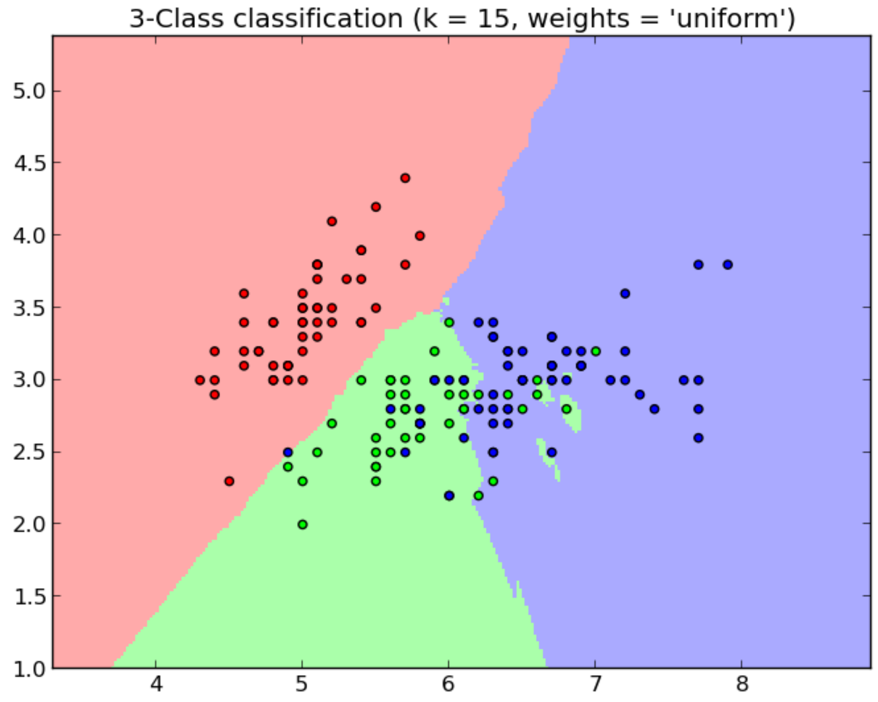
DECISION TREES: ALGORITHMS

C5.0 is Quinlan's latest version release under a proprietary license. It uses less memory and builds smaller rulesets than C4.5 while being more accurate.

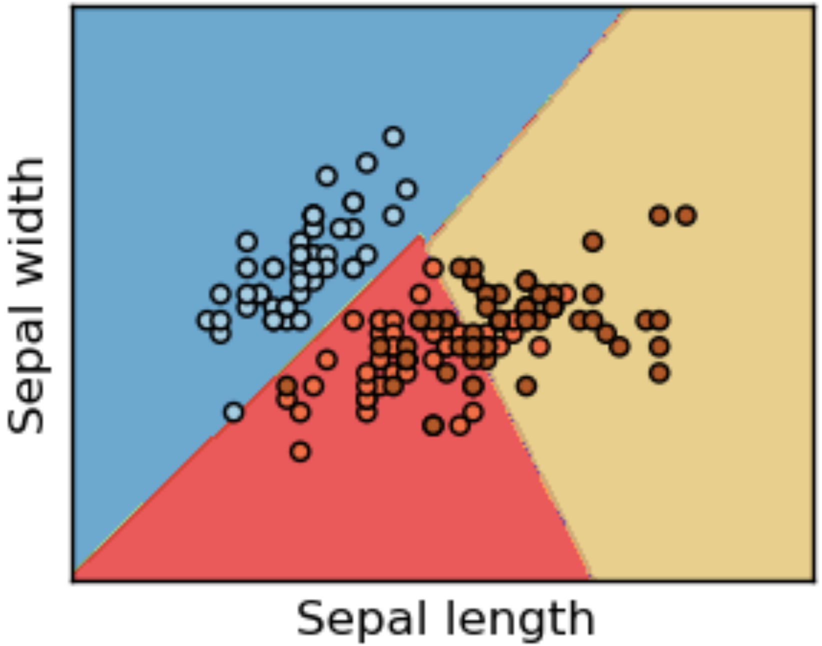
DECISION TREES: ALGORITHMS

CART (Classification and Regression Trees) is very similar to C4.5, but it differs in that it supports numerical target variables (regression) and does not compute rule sets. CART constructs binary trees using the feature and threshold that yield the largest information gain at each node.

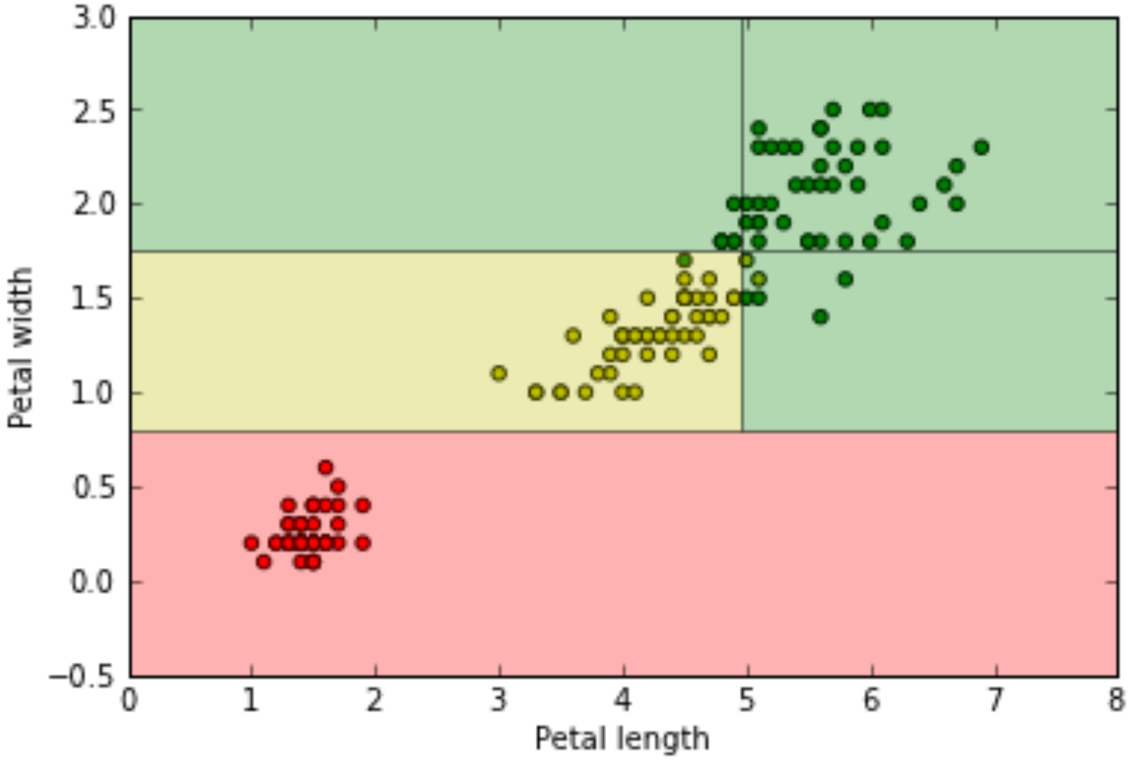
KNN DECISION BOUNDARIES



LOGISTIC REGRESSION DECISION BOUNDARIES



DECISION TREE DECISION BOUNDARIES



GROWING DECISION TREES

- Finding the globally-optimal tree is intractable.

GROWING DECISION TREES

- Finding the globally-optimal tree is intractable.
- ID3 is a greedy search algorithm.

GROWING DECISION TREES

- Finding the globally-optimal tree is intractable.
- ID3 is a greedy search algorithm.
- Other algorithms can be used to grow decision trees

GROWING DECISION TREES

- Finding the globally-optimal tree is intractable.
- ID3 is a greedy search algorithm.
- Other algorithms can be used to grow decision trees
- Other measures can be used to quantify uncertainty

ADVANTAGES OF DECISION TREES

- Features do not need to be scaled or mean-centered

ADVANTAGES OF DECISION TREES

- Features do not need to be scaled or mean-centered
- Some learning algorithms can tolerate missing features

ADVANTAGES OF DECISION TREES

- Features do not need to be scaled or mean-centered
- Some learning algorithms can tolerate missing features
- Implicit feature selection

ADVANTAGES OF DECISION TREES

- Features do not need to be scaled or mean-centered
- Some learning algorithms can tolerate missing features
- Implicit feature selection
- Eager learners

DISADVANTAGES OF DECISION TREES

- Prone to over-fitting

DISADVANTAGES OF DECISION TREES

- Prone to over-fitting
- Prediction can be slower than linear models

DISADVANTAGES OF DECISION TREES

- Prone to over-fitting
- Prediction can be slower than linear models
- Eager learners

REGULARIZATION

- Over-fitting can be controlled by **pruning** trees, or by setting limits on the depth of the tree or the sizes of the subsets.

APPLICATIONS OF DECISION TREES

- Never used

APPLICATIONS OF DECISION TREES

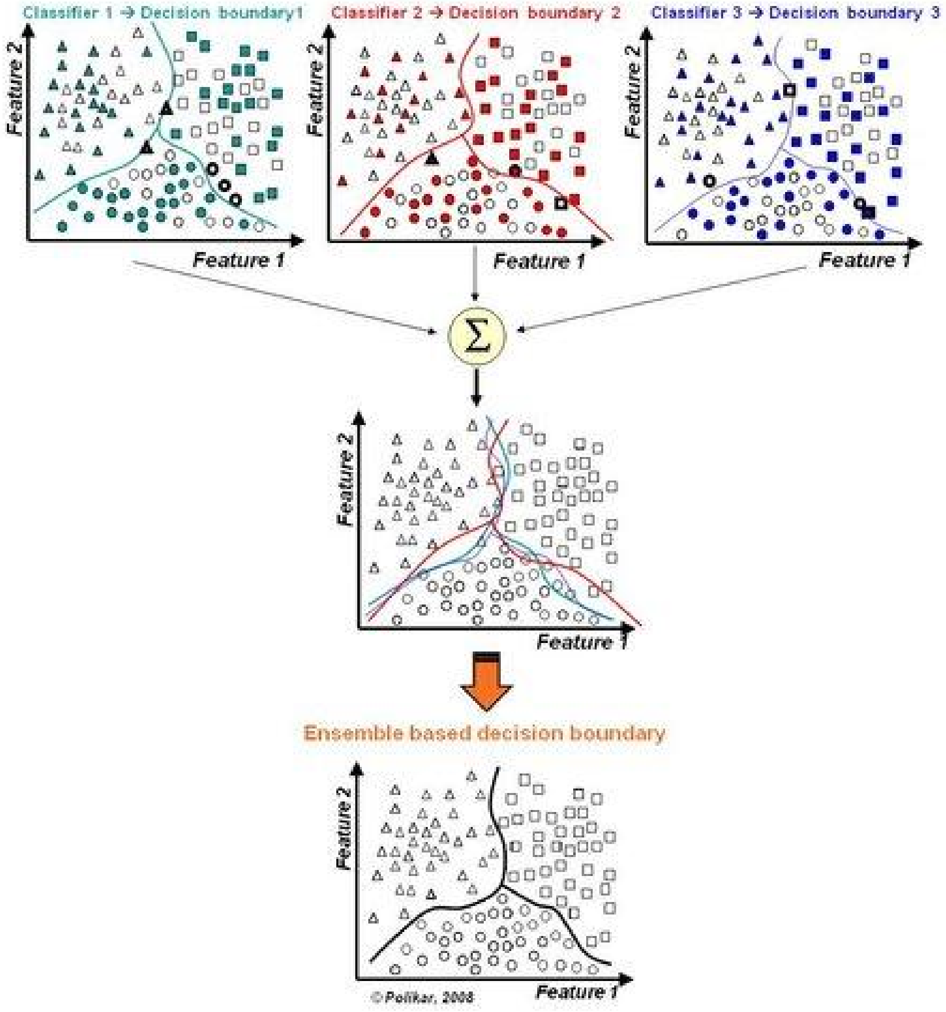
- Never used

¯_(ツ)_/¯

Ensemble Learning

ENSEMBLE LEARNING

- ***Ensemble learning*** methods combine a set of models to produce an estimator that has better predictive performance than its individual components.


Classifier 1 → Decision boundary1
Classifier 2 → Decision boundary 2
Classifier 3 → Decision boundary 3
Feature 2
Feature 1
Σ
Ensemble based decision boundary
© Polikar, 2008

ENSEMBLE LEARNING: BAGGING

Bagging (bootstrap aggregating) is a method that involves manipulating the training set by resampling.

We learn ***k*** base classifiers on ***k*** different samples of training data.

These samples are independently created by resampling the training data using uniform weights (eg, a uniform sampling distribution).

ENSEMBLE LEARNING: BAGGING

Bagging reduces the variance in our generalization error by aggregating multiple base classifiers together.

Since each sample of training data is equally likely, bagging is more robust against overfitting with noisy data.

ENSEMBLE LEARNING: BOOSTING

Boosting is an iterative procedure that adaptively changes the sampling distribution of training records at each iteration.

The first iteration uses uniform weights (like bagging). In subsequent iterations, the weights are adjusted to emphasize records that were misclassified in previous iterations.

The final prediction is constructed by a weighted vote (where the weights for a *bc* depends on its training error).

RANDOM FORESTS

ENSEMBLE LEARNING

- A **random forest** is a collection of decision trees that have been trained on randomly selected subsets of the training instances and explanatory variables.

ENSEMBLE LEARNING

- A **random forest** is a collection of decision trees that have been trained on randomly selected subsets of the training instances and explanatory variables.

- scikit-learn's implementations return the majority/mean of the trees' predictions for classification/regression.

USES OF RANDOM FORESTS

- Widely-applicable
  - Wisdom of Crowds
  - IBM’s WATSON
  - Nate Silver’s election models
  - Kaggle competitions...

RANDOM FORESTS IN SCIKIT-LEARN

- RandomForestClassifier, RandomForestRegressor
- **`fit()`** learns the tree
- **`predict()`** tests the instance using the tree to estimate the value of the response variable.
- **`score()`** makes predictions for the test instances and compares them to the ground truth