Scikit Learn

A Quick Tour

Steps in Supervised Learning

- Selecting features and collecting labeled training examples
- Choosing a performance metric
- Choosing a learning algorithm and training a model
- Evaluating the performance of the model
- Changing the settings of the algorithm and tuning the model

Train & Test Splits

❏ Shuffle the data

❏ Stratify - same proportion of labels in both splits

Stratification is very important to train a balanced model

❏ Scale Features

Multi-Class using Perceptron

The Perceptron is a Binary Classifier

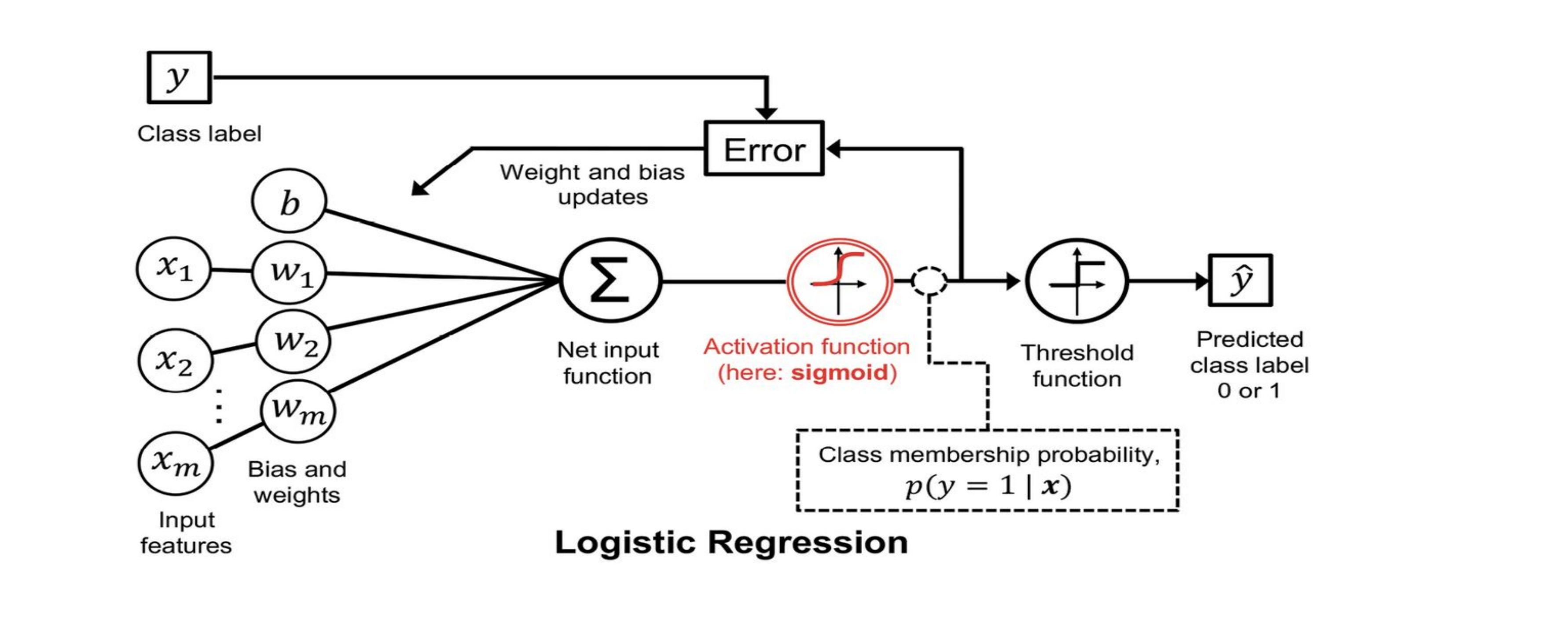
So, can we use it to do multi-class classification?

YES - we use a technique called One-vs-Rest (OvR)

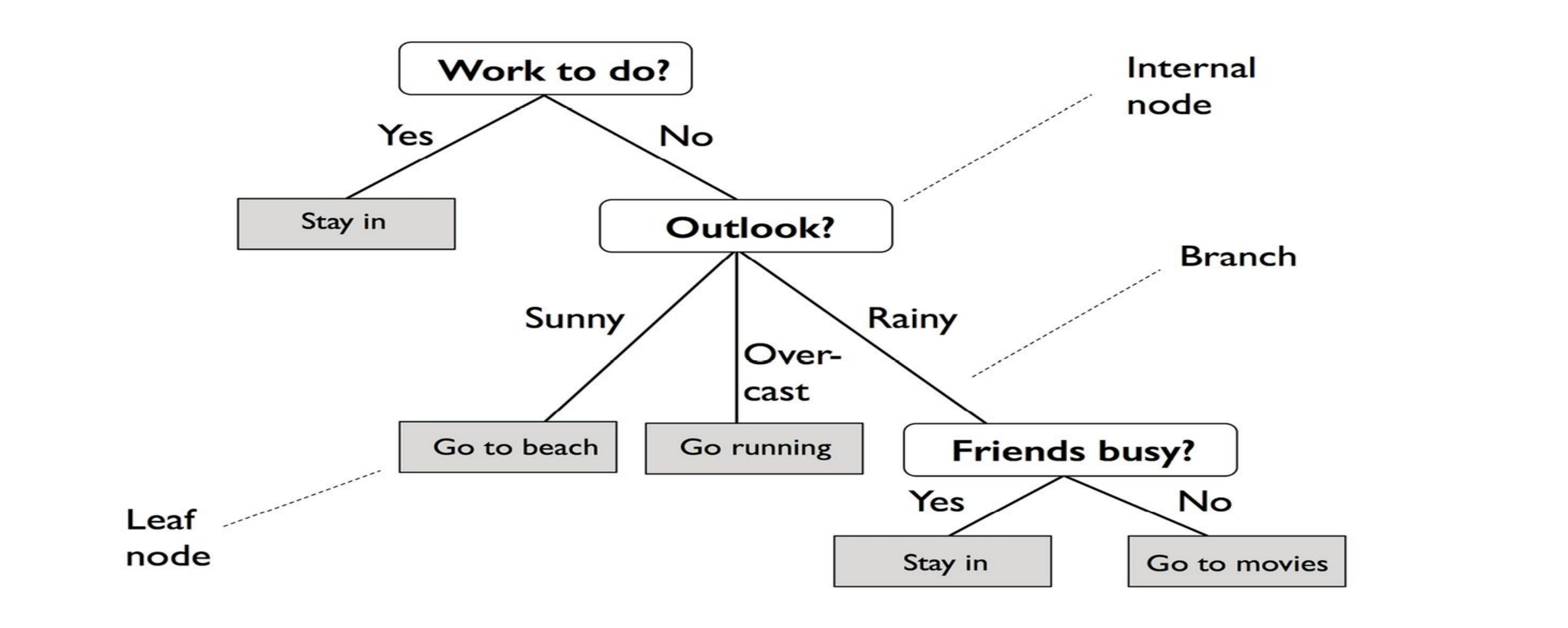
Basically we train many (= number of classes) models

Each model specializes in one class

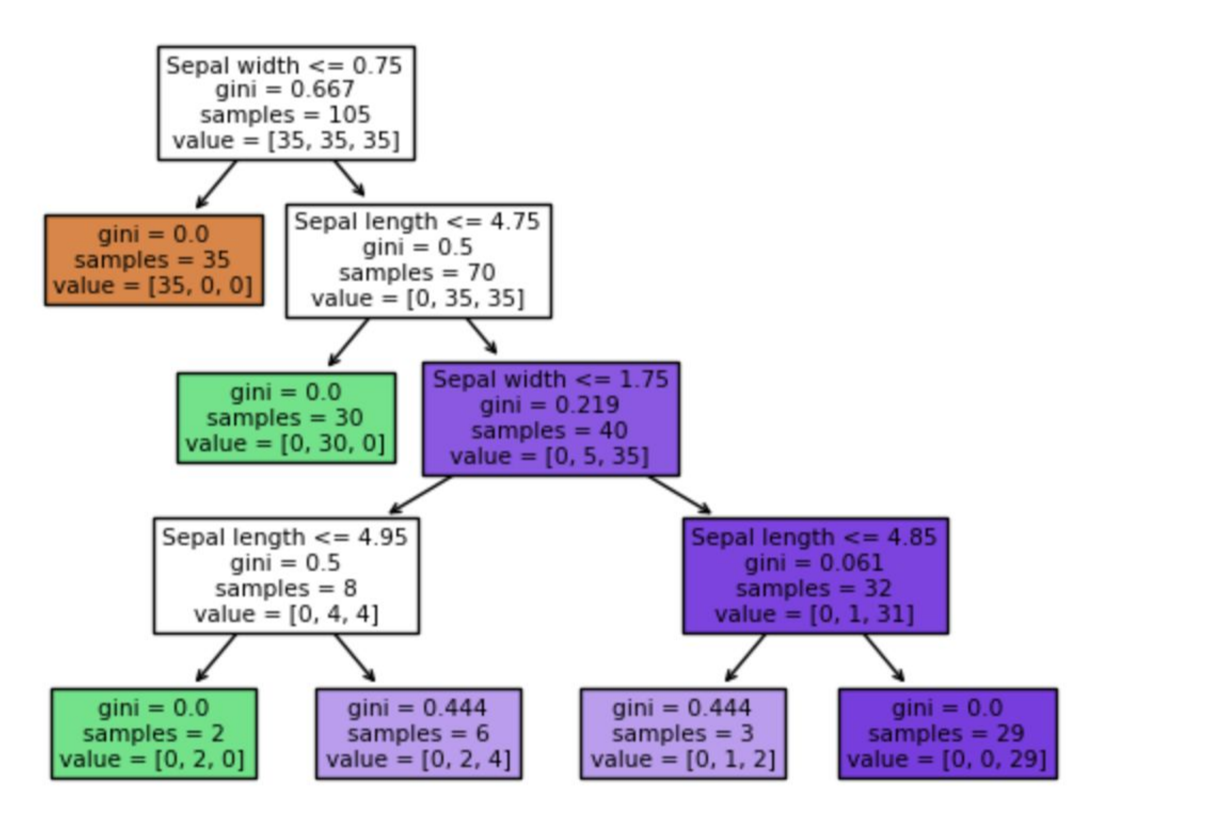
Logistic Regression



Decision Trees



Tree for the Iris DS



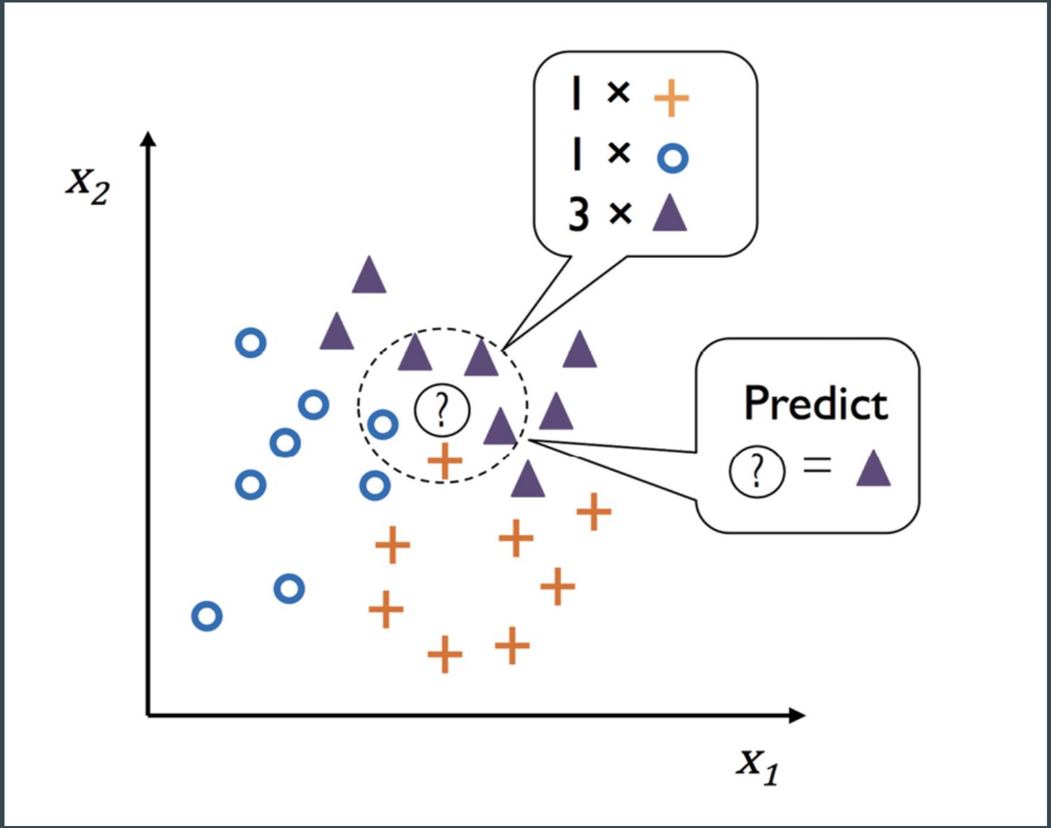
Random Forest

1. Draw a random **bootstrap** sample of size *n* (randomly choose *n* examples from the training dataset with replacement).
2. Grow a decision tree from the bootstrap sample. At each node:
   a. Randomly select *d* features without replacement.
   b. Split the node using the feature that provides the best split according to the objective function, for instance, maximizing the information gain.
3. Repeat *steps 1-2 k* times.
4. Aggregate the prediction by each tree to assign the class label by **majority vote** .

K-nearest Neighbors

1. Choose the number of *k* and a distance metric
2. Find the *k*-nearest neighbors of the data record that we want to classify
3. Assign the class label by majority vote

KNN Classification


x_2
1 × +
1 × ○
3 × ▲
?
Predict
? = ▲
x_1