SPRINT5

目的はなにか

スクラッチを通して決定木を理解する

複雑なアルゴリズムの実装に慣れる

このスライドは？

ここでは、決定木の

基本的な知識を学びましょう

決定木とはなにか

単純な識別規則を組み合わせて、実数値の特徴量に対して軸平行な超平面（識別境界）を得る手法。
具体的には、ある特徴量の値としきい値の**大小関係を判断する過程**を木構造で表現したもの。

与えられた条件は何か

決定木においては以下が仮定されている。

① 解析対象のデータの分布を仮定しない。
② 複数のステップ関数(1)で構成されている。

(1) 入力がある値より大きければ1を返し、そうでなければ0を返す関数

この課題の対象者

① scikit-learnの分類モデルを用いて、学習、推定するコードが書ける方(1)

(1) sprint5 SVMスクラッチを解いた方

この後の流れ

決定木の問題設定を知る

① 特徴量の任意の値を分割の「しきい値」とする（各特徴量ごとに行う）
② ①のしきい値でサンプルを分割する（各特徴量ごとに行う）
③ 分割後のグループごとのサンプルのgini不純度の合計と、分割前の全サンプルのgini不純度の差異を求める
④ その差異が最大になるものを根ノードの分割判定基準とする

Iris data

いまここにIrisデータセットがあるとしよう。

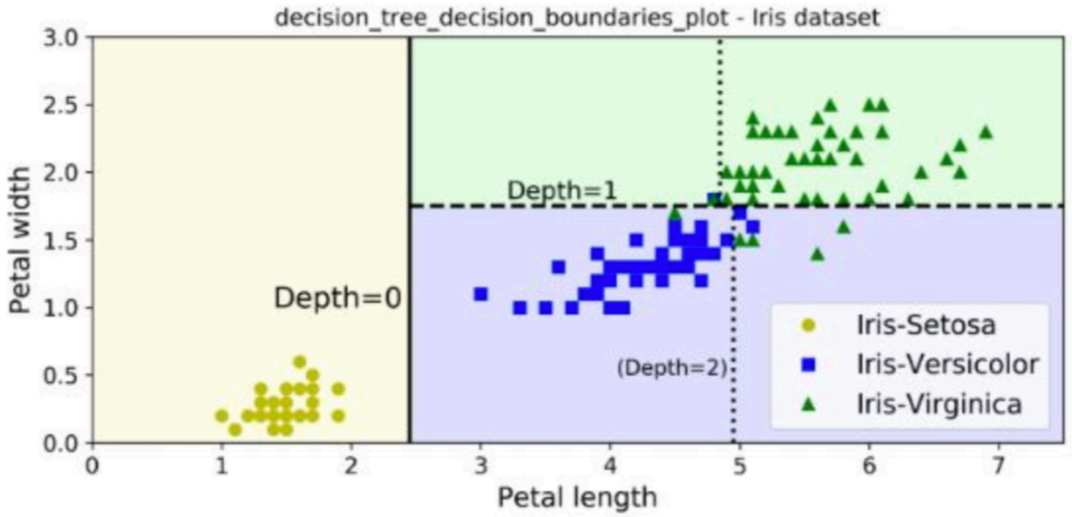
データ点はあらかじめクラスごとに色分けされている。

ある特徴量X_1 (petallength)と特徴量X_2（petalwidth）を選び、

二変数間の関係 をプロットしてみよう。

今回はIris-setosaとIris-versicolor、Iris-Versinicaのクラスを分類するための識別境界（境界線）が引けると嬉しい。

決定木は軸平行な決定境界で分割を繰り返すというが、どのようにして分割点を選択するのだろうか。



この後の流れ

決定木の問題設定を知る

① 特徴量の任意の値を分割の「しきい値」とする（各特徴量ごとに行う）
② ①のしきい値でサンプルを分割する（各特徴量ごとに行う）
③ 分割後のグループごとのサンプルのgini不純度の合計と、分割前の全サンプルのgini不純度の差異を求める
④ その差異が最大になるものを根ノードの分割判定基準とする

分割規則

決定木は、分割候補点をある分割指数で評価することによって選択している。

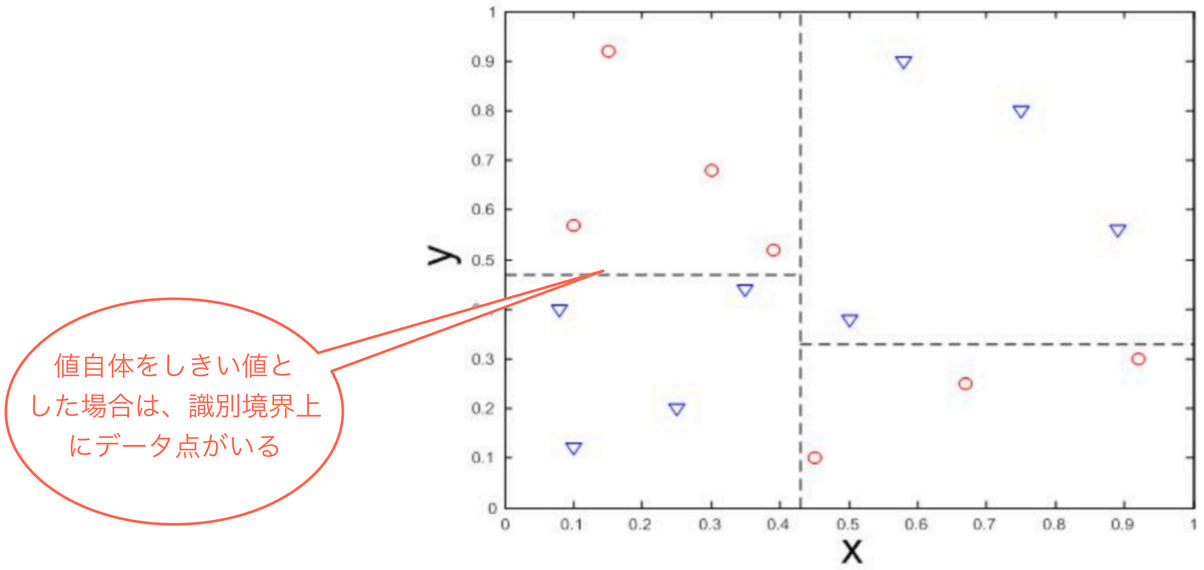
このとき、識別境界はd次元空間の特徴軸に直交する。

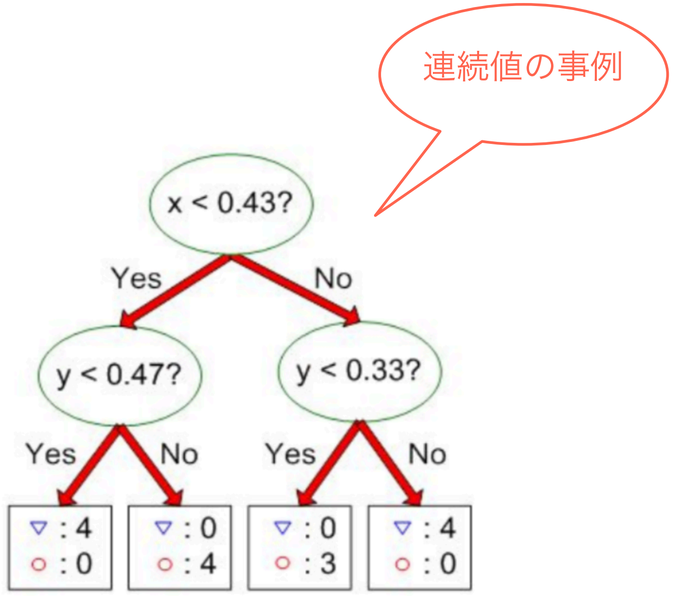
特徴量が連続値のとき

訓練データ数がn 個のとき、 n-1 個の離散的な分割**候補点**が存在する

特徴量が名義尺度・順序尺度のとき

ちょうどカテゴリー数分の分割**候補点**が存在する





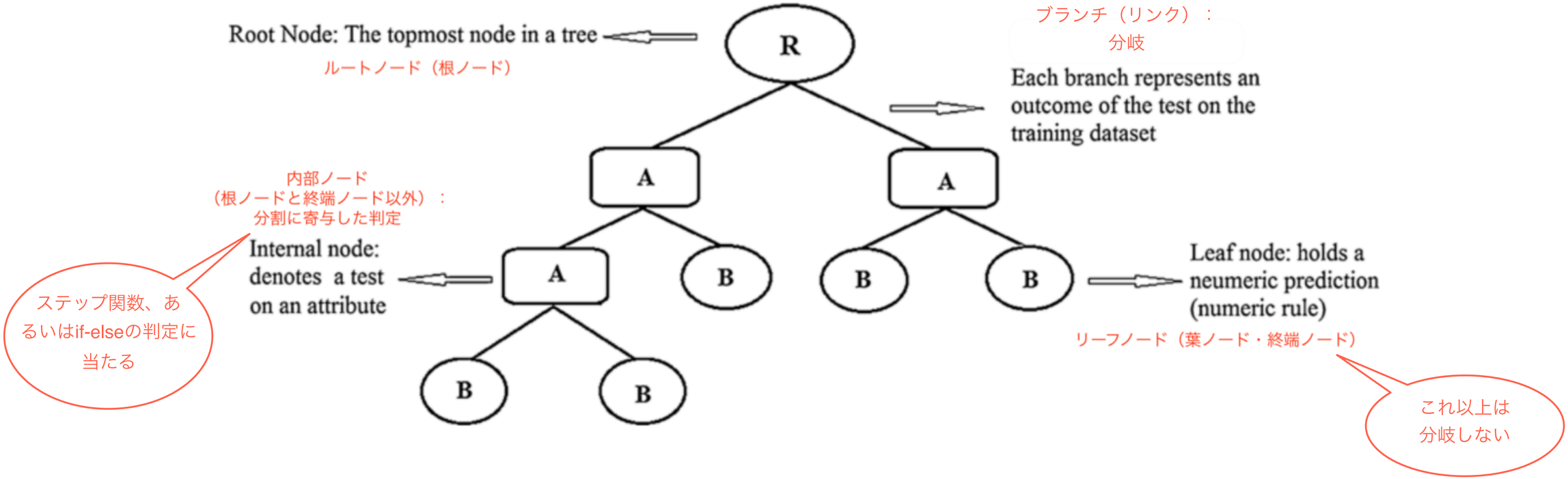
木構造のアルゴリズム

今回は、**2**分木である **CART(classification and regression tree)**というアルゴリズムを用いる。

このアルゴリズムは、分類と回帰のどちらにも対応している。

(1) 他のアルゴリズムとしては、C4.5というN分木を生成するアルゴリズムがある。こちらは分類問題にのみ適応可能である。

各ノードが属性（特徴量名）の判定を表し、各リンク（ブランチ）が分岐を表し、各リーフが結果（カテゴリ値または連続値）を表す。



この後の流れ

決定木の問題設定を知る

① 特徴量の任意の値を分割の「しきい値」とする（各特徴量ごとに行う）
② ①のしきい値でサンプルを分割する（各特徴量ごとに行う）
③ 分割後のグループごとのサンプルのgini不純度の合計と、分割前の全サンプルのgini不純度の差異を求める
④ その差異が最大になるものを根ノードの分割判定基準とする

目的関数 (分類の場合)

CART の分類モデルには、分割指数として**ジニ不純度(Gini Impurity)** または**交差エントロピー**を用いる。

今回は**ジニ不純度 (ノード t における誤り率)** に基づいて分割を行う。

ジニ不純度は、以下のように定式化される。

ジニ係数(Gini index)とも呼ばれる

あるノードから取り出したサンプルについて

それが i 番目のクラスであるときは1を、それ以外のクラスであるときは0とする

試行（これをベルヌーイ試行という）を考えたとき、

ジニ不純度は、そのノードでのサンプルのクラスが異なる（1のクラスと0のクラスのサンプルがほぼ同程度存在する、つまり、偏りが小さい）**確率**といえる。

また、ベルヌーイ分布におけるすべてのクラスの分散の和に相当する。

$$I(t) = 1 - \sum_{i=1}^{K} P^2(C_i|t) = 1 - \sum_{i=1}^{K} \left(\frac{N_{t,i}}{N_{t,al}} \right)^2$$

$$= \sum_{i=1}^{K} P(C_i|t)(1 - P(C_i|t))$$

式の意味：

ノード t でクラス i が選ばれる確率Pと、クラス i 以外が選ばれる

確率 1-P を掛け合わせる計算をKクラス分行い、足し合わせている。

手計算でジニ不純度の求め方を確認しよう

右図は、分割前と分割後におけるクラス数を示した2つの決定木（A, B）である。AとBそれぞれにおけるジニ不純度を計算してみよう。

他の事例はこちら。

https://www.randpy.tokyo/entry/decision_tree_theory

根ノード

$$I_G(D_p) = 1 - \left(\left(\frac{40}{80}\right)^2 + \left(\frac{40}{80}\right)^2\right) = 1 - (0.5^2 + 0.5^2) = 0.5$$

$$A: I_G(D_{left}) = 1 - \left(\left(\frac{30}{40}\right)^2 + \left(\frac{10}{40}\right)^2\right) = 1 - \left(\frac{9}{16} + \frac{1}{16}\right) = \frac{3}{8} = 0.375$$

$$A: I_G(D_{right}) = 1 - \left(\left(\frac{10}{40}\right)^2 + \left(\frac{30}{40}\right)^2\right) = 1 - \left(\frac{1}{16} + \frac{9}{16}\right) = \frac{3}{8} = 0.375$$

$$A: I_G = 0.5 - \frac{40}{80} \times 0.375 - \frac{40}{80} \times 0.375 = 0.125$$

最後は**情報利得**（分割前後の差異を評価。**大きい**方が嬉しい）の計算だよ。

$$B: I_G(D_{left}) = 1 - \left(\left(\frac{20}{60}\right)^2 + \left(\frac{40}{60}\right)^2\right) = 1 - \left(\frac{9}{16} + \frac{1}{16}\right) = 1 - \frac{5}{9} = 0.44$$

$$B: I_G(D_{right}) = 1 - \left(\left(\frac{20}{20}\right)^2 + \left(\frac{0}{20}\right)^2\right) = 1 - (1 + 0) = 1 - 1 = 0$$

こちらも

$$B: I_G = 0.5 - \frac{60}{80} \times 0.44 - 0 = 0.5 - 0.33 = 0.17$$

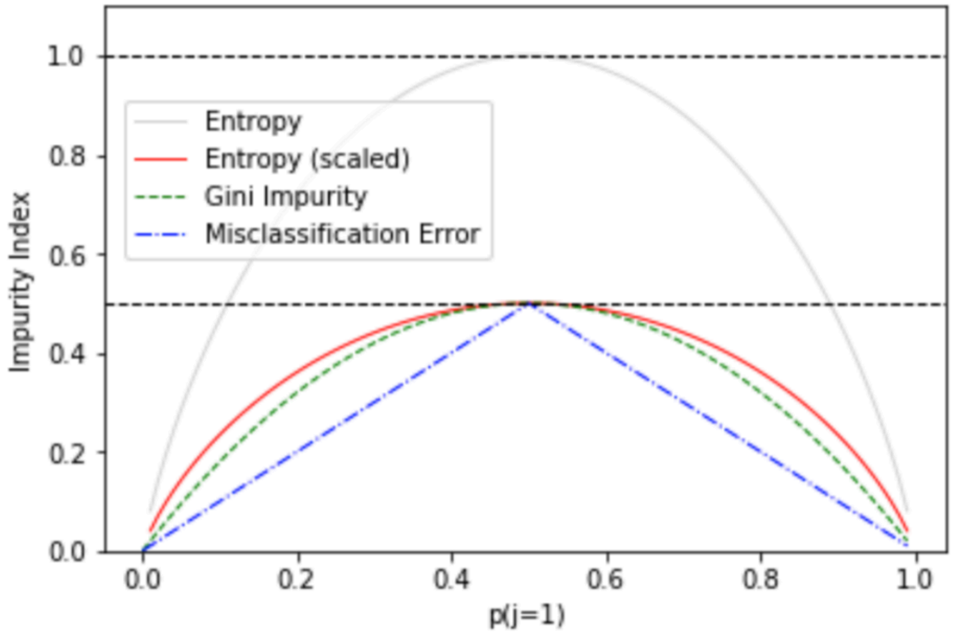
ジニ不純度の取りうる範囲

ノードt でクラス j が選ばれる確率（横軸）と、不純度指標（縦軸）の関係は、以下のようなグラフで表すことができる。クラスの確率0.5のとき、ジニ不純度は最大値0.5をとる。

完全に分割されるとき、不純度は0となる。

下のグラフのコードはこちら。

https://www.bogotobogo.com/python/scikit-learn/scikt_machine_learning_Decision_Tree_Learning_Informatioin_Gain_IG_Impurity_Entropy_Gini_Classification_Error.php



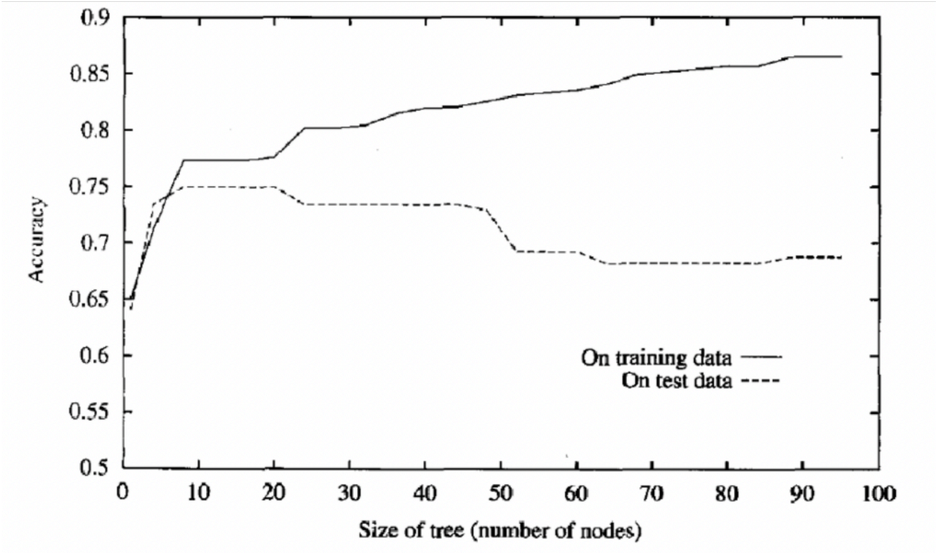
不純度は**小さい**方が嬉しい！

決定木の問題点

分割を繰り返しモデルの複雑さが増すと、訓練データに過剰適合しやすくなる。
すると、得られる訓練データが大きく異なる場合、学習後に得られるツリー構造もまた大きく異なってしまう。
このとき、そのモデルは**分散(バリアンス)が高い**とされる。

訓練データから、ランダムに標本再抽出(ブートストラップ標本)を行って、それぞれに対して決定木を当てはめ、複数の決定木の結果に対して多数決を行う、バギングという手法がある。

一つの決定木からの結果の**不安定さを補う**という発想からなる。



いつ分岐をやめるの？

特徴量が多ければ、多数の分割が発生し、結果として巨大なツリーが作成される。そのようなツリーは複雑で、過剰適合につながる恐れがある。

過剰適合を回避する一つの方法は、各リーフで使用する入力データの**最小数**を設定することである。

また別の方法にとして、重要度の**低い**判定を削除する「**枝借り**（剪定）」という手法もある。これによって外れ値の影響を避け、過剰適合を防ぐことができる。

この後の流れ

決定木の問題設定を知る

① 特徴量の任意の値を分割の「しきい値」とする（各特徴量ごとに行う）
② ①のしきい値でサンプルを分割する（各特徴量ごとに行う）
③ 分割後のグループごとのサンプルのgini不純度の合計と、分割前の全サンプルのgini不純度の差異を求める
④ その差異が最大になるもの（情報利得の最大化）を根ノードの分割判定基準とする