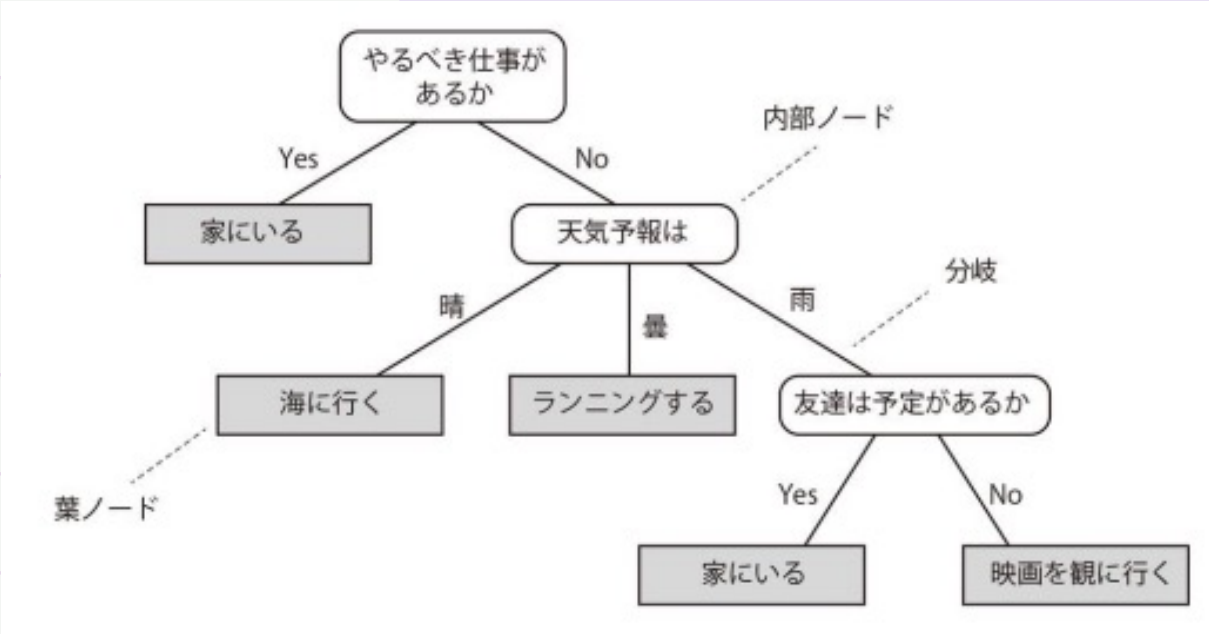
決定木:Decision Tree

概容

- ある日の行動を決定する。

「トレーニングデータの特徴量に基づいて質問の答えを学習する。」



ちょっとわかりずらいので、、、

概容

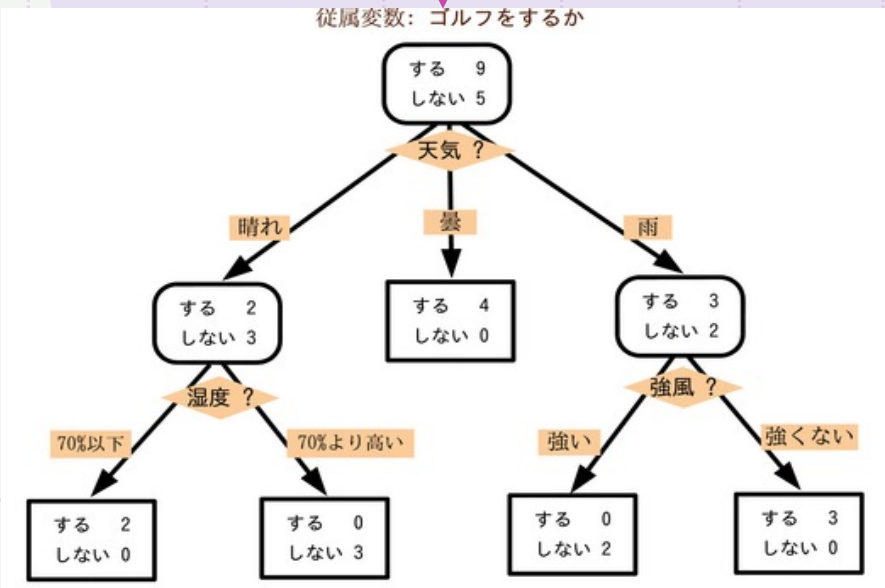
- ある日の行動を決定する。

「トレーニングデータの特徴量に基づいて質問の答えを学習する。」

ゴルフクラブ来場状況

独立変数				従属変数
天気	気温(℃)	湿度(%)	風が強いか	ゴルフをするか
晴れ	29	85	強くない	しない
晴れ	27	90	強い	しない
曇	28	78	強くない	する
雨	21	96	強くない	する
雨	20	80	強くない	する
雨	18	70	強い	しない
曇	18	65	強い	する
晴れ	22	95	強くない	しない
晴れ	21	70	強くない	する
雨	24	80	強くない	する
晴れ	24	70	強い	する
曇	22	90	強い	する
曇	27	75	強くない	する
雨	22	80	強い	しない

アルゴリズムを考案してみましょう



概容

Step1：
集合Cのすべてのデータが同一クラスならクラスノードを作り停止．それ以外なら，属性の選択基準により一つの属性 A を選択し判別ノードを作る．

Step2：
属性Aの属性値によりCを部分集合C1,C2,・・・，Cnに分けてノードを作り属性値の枝を張る．

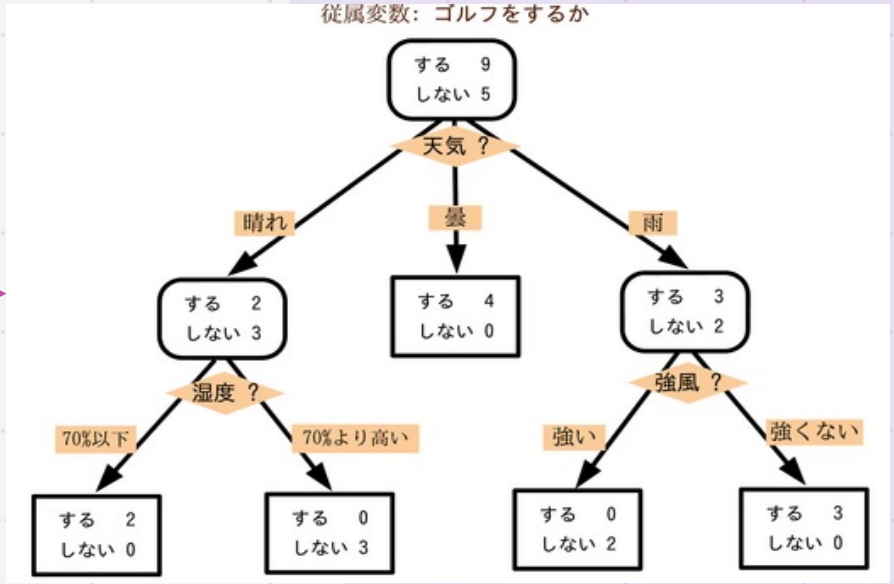
Setp3：
ノードCi（1≦i≦n）について，アルゴリズムを再起的に適用する．

- ある日の行動を決定する。

「トレーニングデータの特徴量に基づいて質問の答えを学習する。」

ゴルフクラブ来場状況

独立変数				従属変数
天気	気温(℃)	湿度(%)	風が強いか	ゴルフをするか
晴れ	29	85	強くない	しない
晴れ	27	90	強い	しない
曇	28	78	強くない	する
雨	21	96	強くない	する
雨	20	80	強くない	する
雨	18	70	強い	しない
曇	18	65	強い	する
晴れ	22	95	強くない	しない
晴れ	21	70	強くない	する
雨	24	80	強くない	する
晴れ	24	70	強い	する
曇	22	90	強い	する
曇	27	75	強くない	する
雨	22	80	強い	しない



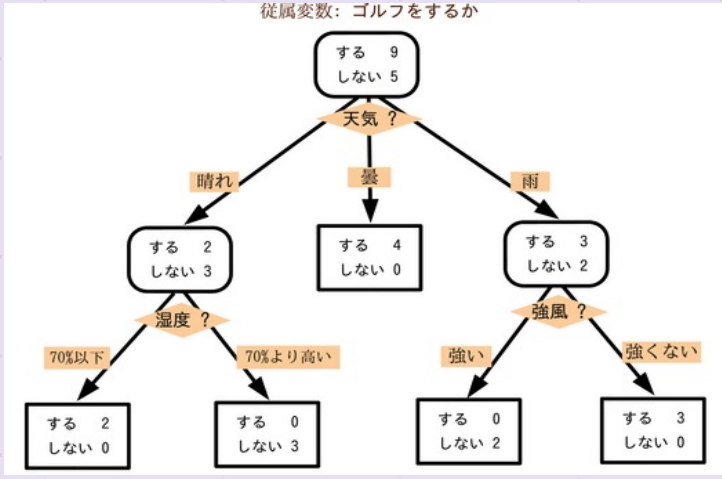
属性の選択基準

情報利得(Information gain:IG)を最大にする

Step1：
集合Cのすべてのデータが同一クラスならクラスノードを作り停止．それ以外なら，
属性の選択基準により一つの属性Aを選択し判別ノードを作る．

Step2：
属性Aの属性値によりCを部分集合C1,C2,・・・，Cnに分けてノードを作り属性値の枝を張る．

Setp3：
ノードCi（1≦i≦n）について，アルゴリズムを再起的に適用する．



$$IG(D_p, f) = I(D_p) - \sum_{j=1}^{m} \frac{N_j}{N_p} I(D_j)$$

ここで、fは分割を行う特徴量であり、D_pは親のデータセット、D_jはj番目の子ノードのデータセットである。Iは**不純度**(impurity)※35を数値化したものであり、N_pは親ノードのデータ点の総数、N_jはj番目の子ノードのデータ点の個数である。このように、情報利得は「親ノードの不純度」と「子ノードの不純度の合計」との差にすぎない。つまり、子ノードの不純度が低いほど、情報利得は大きくなる。

属性の選択基準

$$IG(D_p, f) = I(D_p) - \sum_{j=1}^{m} \frac{N_j}{N_p} I(D_j)$$

ここで、fは分割を行う特徴量であり、D_pは親のデータセット、D_jはj番目の子ノードのデータセットである。Iは**不純度**(impurity)[※35]を数値化したものであり、N_pは親ノードのデータ点の総数、N_jはj番目の子ノードのデータ点の個数である。このように、情報利得は「親ノードの不純度」と「子ノードの不純度の合計」との差にすぎない。つまり、子ノードの不純度が低いほど、情報利得は大きくなる。

不純度の指標

多くの実装で
エントロピー or Gini不純度
が用いられている

$$I_H(t) = -\sum_{i=1}^{c} p(i|t) \log_2 p(i|t)$$

entropy

$$I_G(t) = \sum_{i=1}^{c} p(i|t)(1 - p(i|t)) = 1 - \sum_{i=1}^{c} p(i|t)^2$$

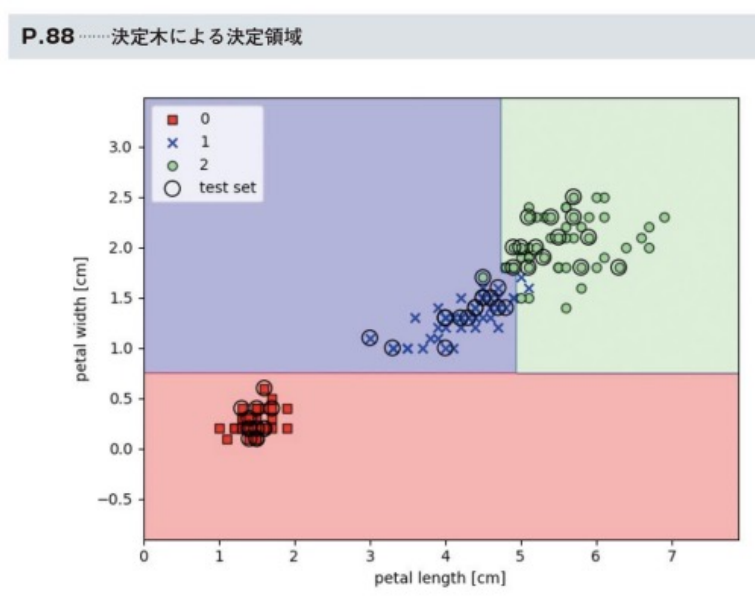
Gini 不純度

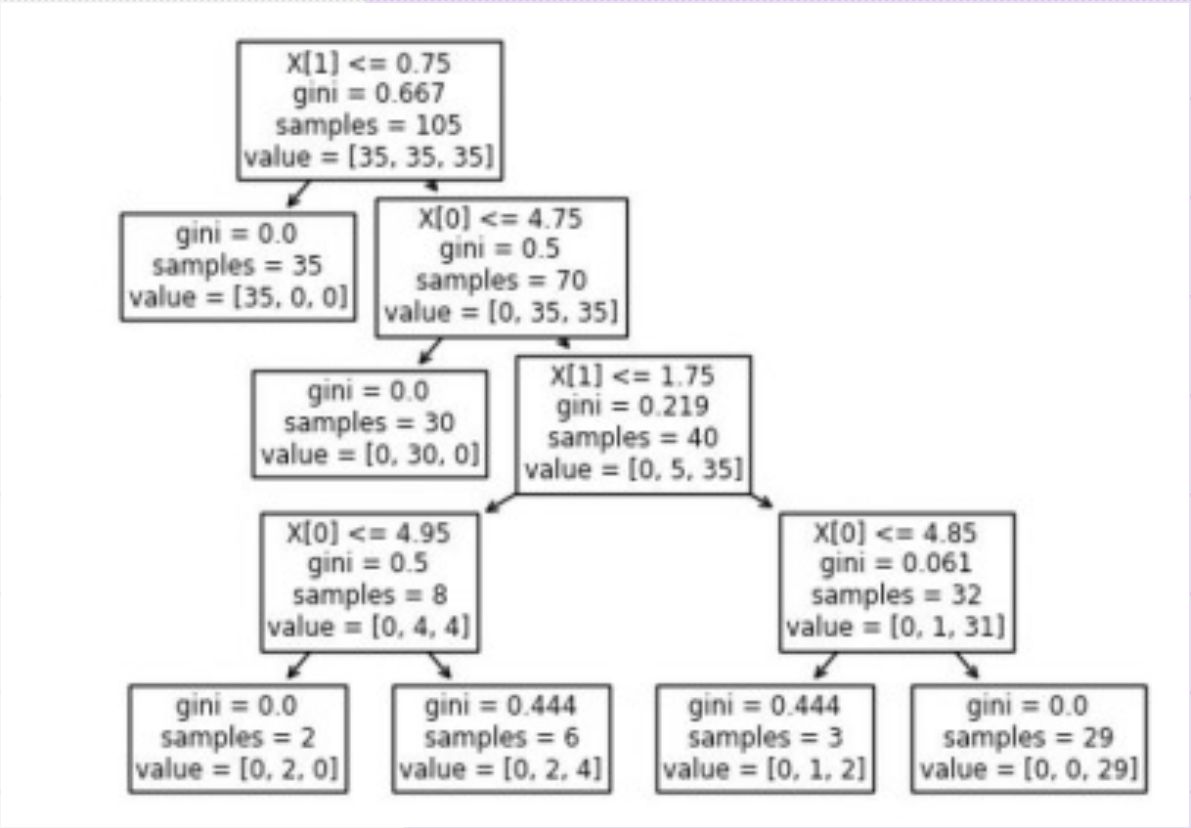
$$I_E(t) = 1 - \max\{p(i|t)\}$$

分類誤差

$p(i|t)$は、特定のノードtにおいてクラスiに所属しているデータ点の割合を表す。

特徴



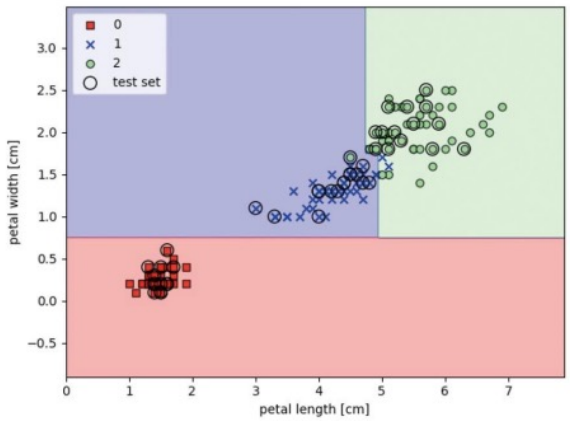


意味解釈可能性（Interpretability）
得られた結果の意味解釈ができるかどうか？

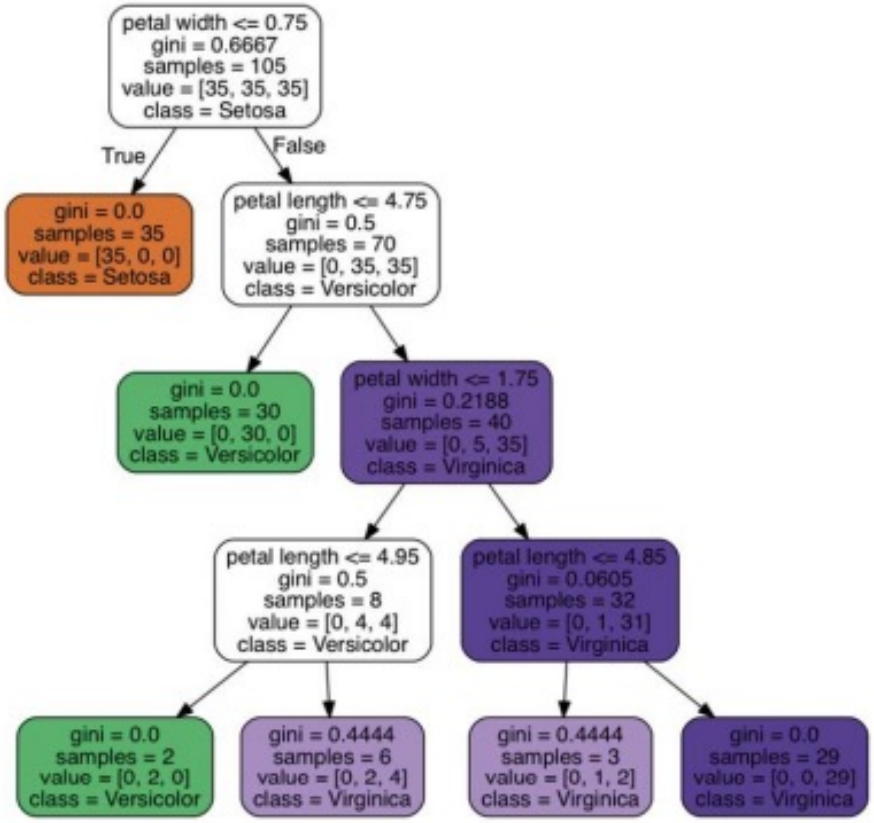
意味解釈可能性（Interpretability）

得られた結果の意味解釈ができるかどうか？

P.88 ……決定木による決定領域



P.90 ……分割条件を示す決定木の画像



Random Forest

決定木で多数決？（アンサンブル学習）

1. サイズ n のランダムな**ブートストラップ**標本を復元抽出[※41]する（訓練データセットから n 個のデータ点をランダムに選択する）。
2. ブートストラップ標本から決定木を成長させる。各ノードで以下の作業を行う。
   - 2.1 d 個の特徴量をランダムに非復元抽出[※42]する。
   - 2.2 たとえば情報利得を最大化することにより、目的関数に従って最適な分割となる特徴量を使ってノードを分割する。
3. 手順 1 ～ 2 を k 回繰り返す。
4. 決定木ごとの予測をまとめ、**多数決**に基づいてクラスラベルを割り当てる。多数決については、第 7 章で詳しく説明する。

復元抽出と非復元抽出

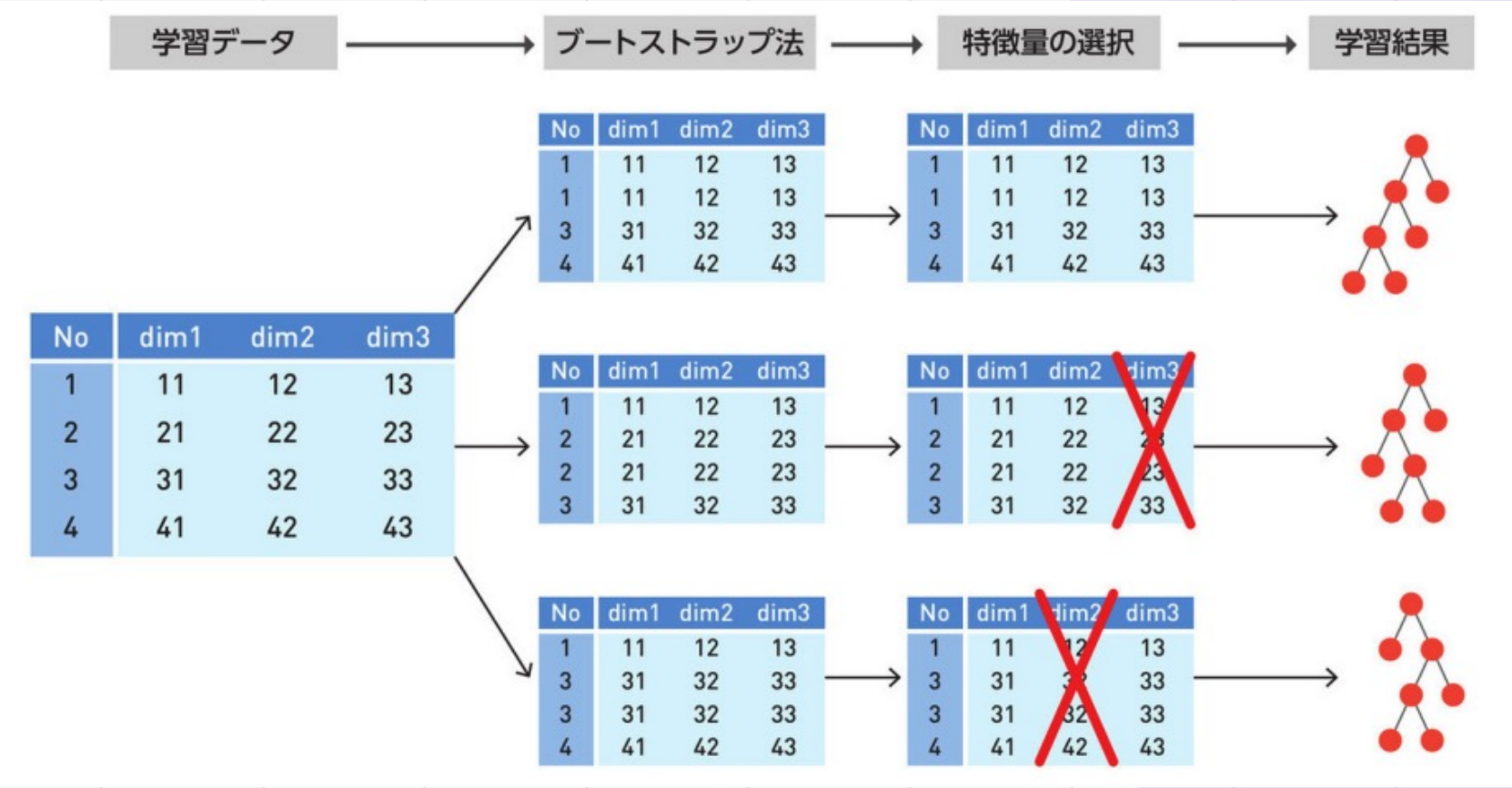
復元抽出と非復元抽出

復元抽出 (sampling with replacement) と**非復元抽出** (sampling without replacement) という言葉になじみがないかもしれないので、簡単な思考実験をしてみよう。くじ引きゲームで、つぼの中から数字を適当に引くとしよう。0、1、2、3、4の5つの数字の入ったつぼを用意し、1回につき1つの数字を引く。1回目につぼから特定の数字を引く確率は5分の1である。さて、非復元抽出では、1回ごとにつぼに数字を戻さない。したがって、次の回に残りの数字の中から特定の数字を引く確率は前の回によって決まる。たとえば、つぼの中に残っている数字が0、1、2、4であるとすれば、次の回に0を引く可能性は4分の1になる。

これに対し、復元抽出では、引いた数字を常につぼに戻すため、特定の数字を引く確率は毎回変化しない。毎回同じ数字を引く可能性もある。つまり、復元抽出では、サンプル (数字) は独立しており、共分散は0である。たとえば、数字を5回引いた結果は次のようになるかもしれない。

- ランダムな非復元抽出：2、1、3、4、0
- ランダムな復元抽出：1、3、3、4、1

Random Forestの直感的理解



多数決（majority voting）の性能

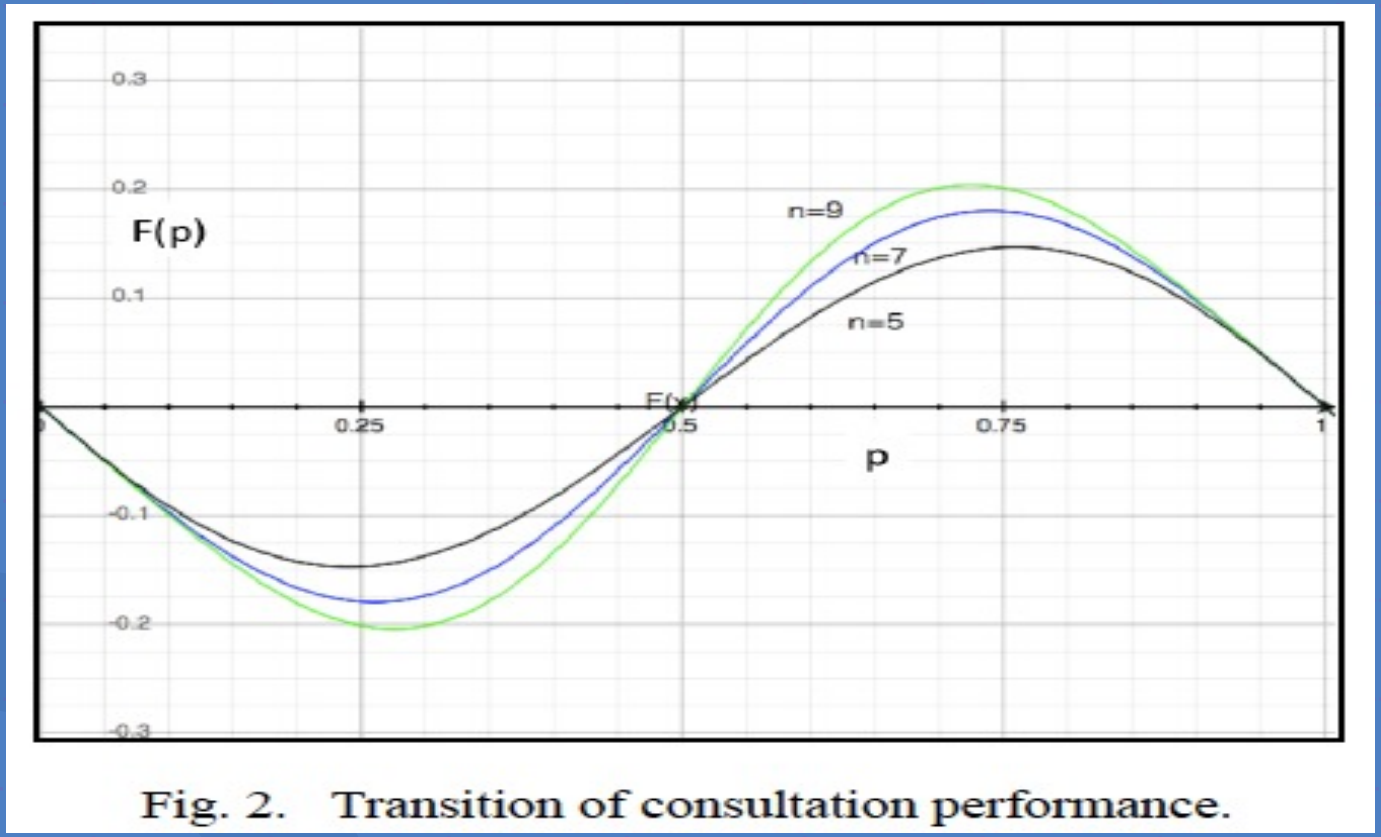


Fig. 2. Transition of consultation performance.

多数決の性能

正答率５０％の人がいくら集まっても、正答率は上がらない。

正答率１００％の人に多数決は必要ない。

一定程度の正答率の高さをもつ人が複数集まることによって多数決が有効に働く

Theoretical Value of consultation

- Number of members to participate in the consultation

  n

- Accuracy rate of the answer for each member

  p

Accuracy rate of the consultation Q(p)?

$$Q(p) = p^n + {}_nC_{n-1}\,p^{n-1}(1-p) + {}_nC_{n-2}\,p^{n-2}(1-p)^2 + \cdots + {}_nC_{\frac{n+1}{2}}\,p^{\frac{n+1}{2}}(1-p)^{\frac{n-1}{2}} \qquad (1)$$

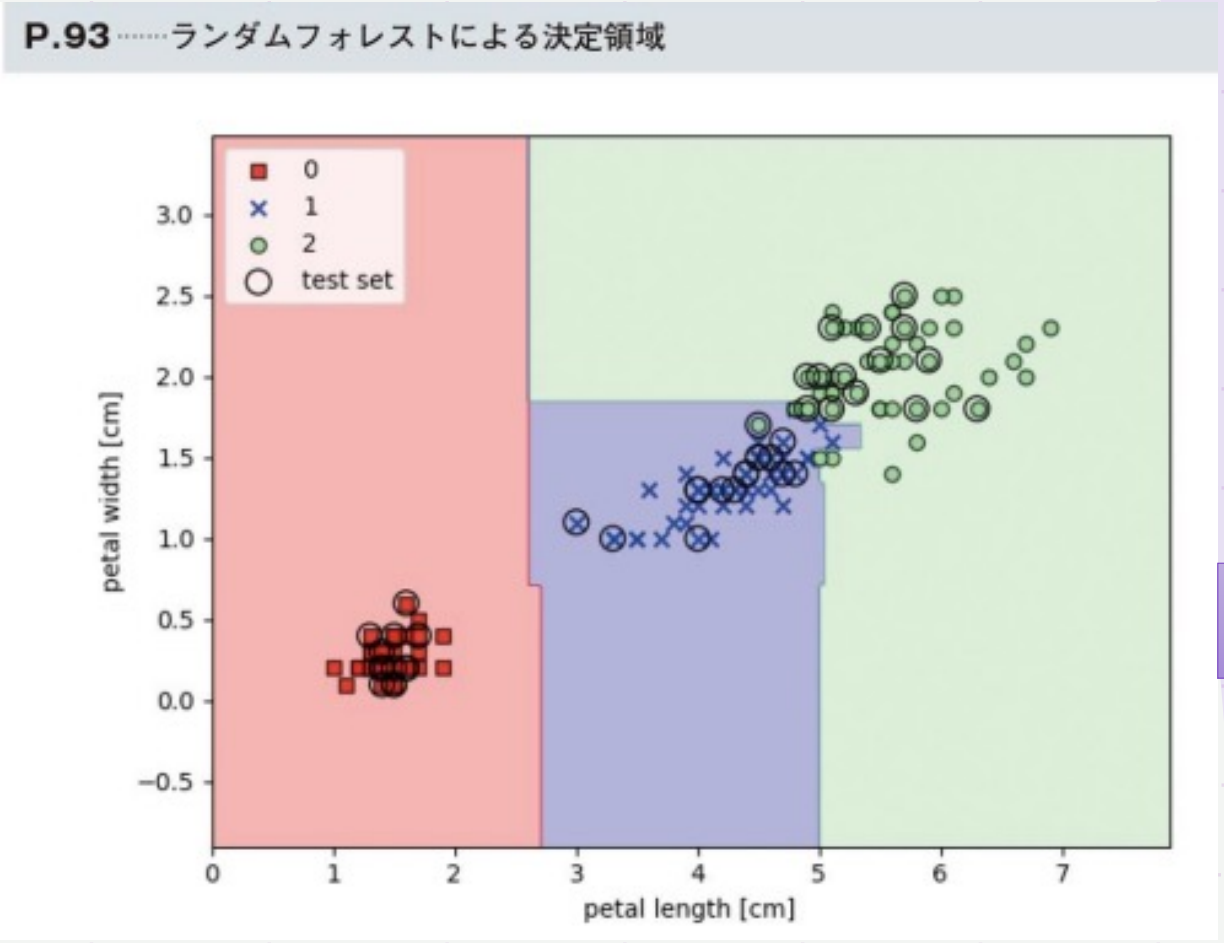
where n is odd number.

Performance of consultation F(p) is,

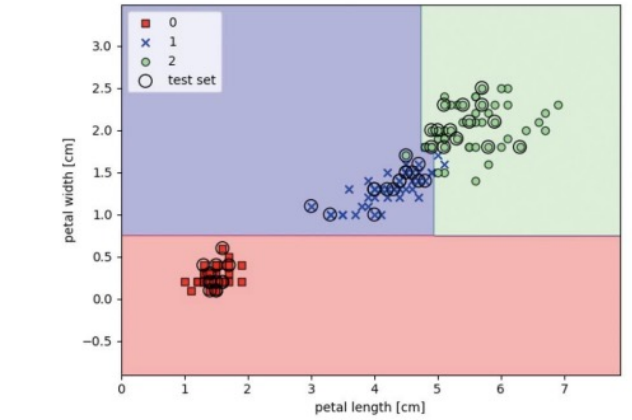
$$F(p) = Q(p) - p$$

Random Forestの実行結果

P.93……ランダムフォレストによる決定領域



P.88……決定木による決定領域



実行結果について、バイアスとバリアンスの観点から考察してみよう。