

京都大学 情報学科 3回生向け講義「人工知能」

# 様々な機械学習の方法

## 決定木 (Decision Trees)



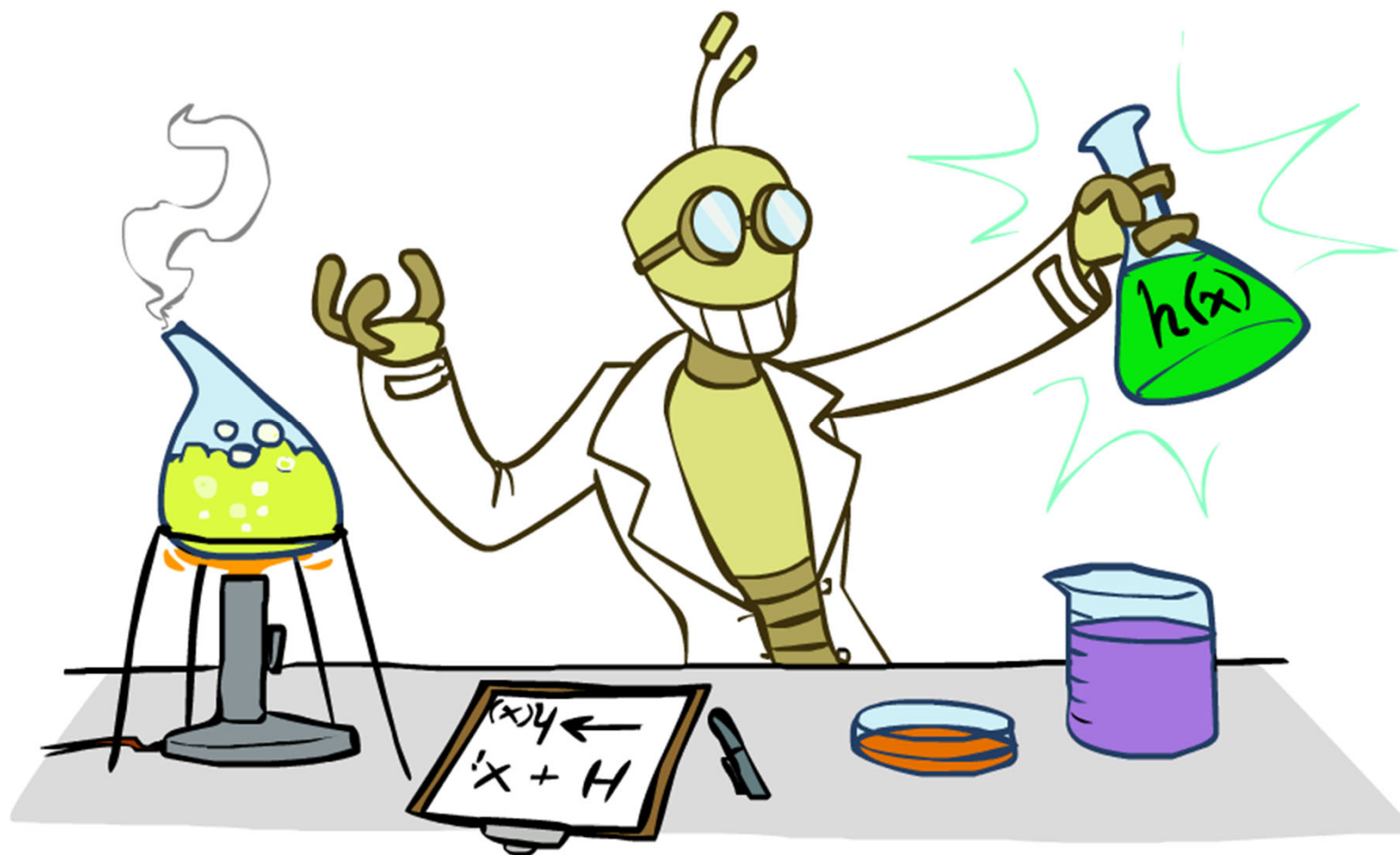
情報学研究科 教授 神田崇行

kanda@i.kyoto-u.ac.jp

本講義資料の無断複製、無断配布を禁止します

[Original slides were created by Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley. All CS188 materials are available at <http://ai.berkeley.edu>.]

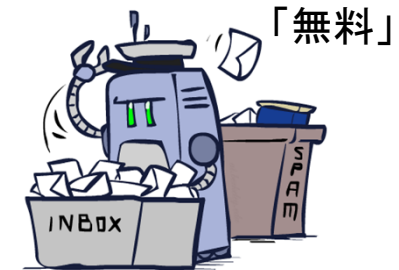
# 帰納学習 (Inductive Learning)



# 帰納学習

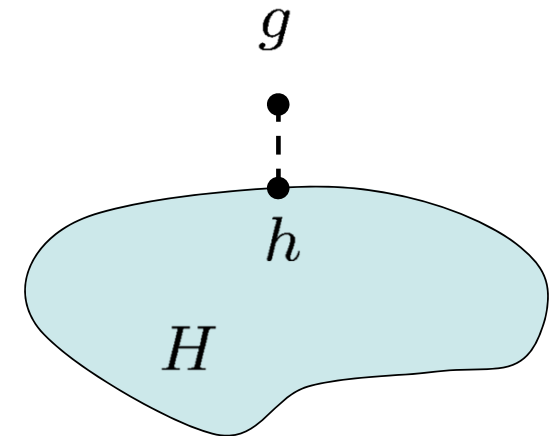
- 最もシンプルには: 複数の例から関数を学習する

- 目標の関数:  $g$
- 例: 入出力のペア  $(x, g(x))$
- E.g.  $x$  は電子メール、 $g(x)$  はスパムかどうか
- E.g.  $x$  は家、 $g(x)$  は売値



- 問題の定義:

- 仮説空間  $H$  があり
- 訓練例の集合  $x_i$  が与えられたとき
- $h \sim g$  となるような仮説  $h(x)$  をみつける



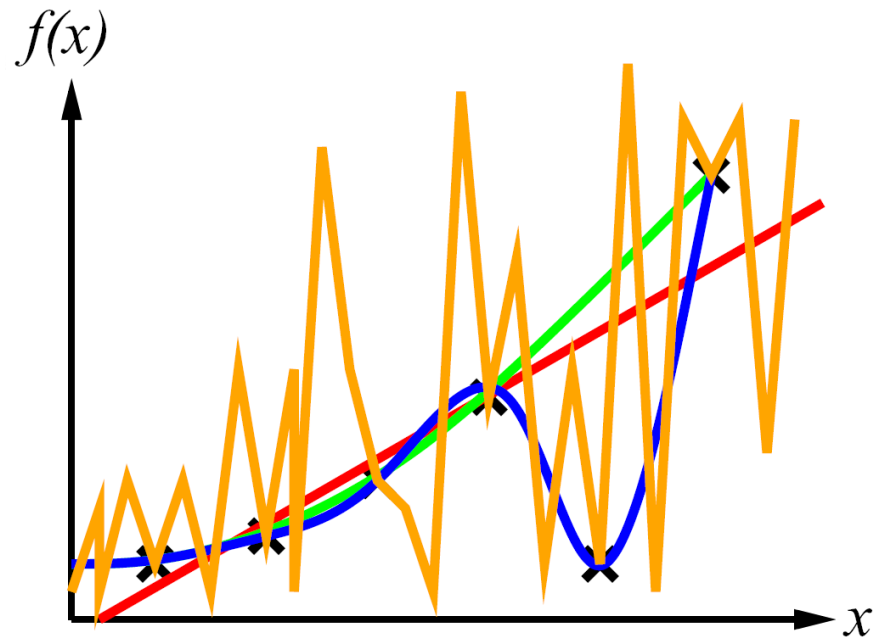
- 含まれるもの:

- 分類 (出力 = クラスのラベル)
- 回帰 (出力 = 実数)

- 単純ベイズ分類器などの分類器でどうやって実現? ( $H, h, g$ , etc.)

# 帰納学習

- 曲線のフィッティング (回帰, 関数近似):



- 整合性 (consistency) vs. 単純性 (simplicity)
- オッカムのカミソリ (Ockham's razor)

# 整合性 (consistency) vs. 単純性 (simplicity)

- 本質的なトレードオフ: バイアス(bias) vs. 分散 (variance)
- 通常のアлゴリズムは、デフォルトでは整合性を好みがち
- 「単純性」を扱うためのいくつかの方法：
  - 仮説空間 (hypothesis space) を小さくする
    - より多くの仮定を置く: e.g. 単純ベイズのように、独立性の仮定
    - 少なめの、質の良い特徴量 / 属性: 特徴量選択
    - 他の構造的な制約 (決定リスト vs 木)
  - 正則化 (Regularization)
    - スムージング: 観察が少ない場合を慎重に扱う
    - 一般化のためのパラメーター (決定木での枝刈りカットオフ)
    - 仮説空間は大きいままでも、周辺(outskirts)にたどり着きにくくする

※単純 ⇒ 人間可読性が高い ⇒ 説明できる(explainable) AI のアプローチの一種とも言える

# 決定木



# 例題

## ■ 特徴量（属性）

- 表現例 1 : TYPE=French
- 表現例 2 :  $f_{\text{TYPE=French}}(x) = 1$

| Example  | Attributes |            |            |            |             |               |             |            |                |               | Target          |
|----------|------------|------------|------------|------------|-------------|---------------|-------------|------------|----------------|---------------|-----------------|
|          | <i>Alt</i> | <i>Bar</i> | <i>Fri</i> | <i>Hun</i> | <i>Pat</i>  | <i>Price</i>  | <i>Rain</i> | <i>Res</i> | <i>Type</i>    | <i>Est</i>    | <i>WillWait</i> |
| $X_1$    | <i>T</i>   | <i>F</i>   | <i>F</i>   | <i>T</i>   | <i>Some</i> | <i>\$\$\$</i> | <i>F</i>    | <i>T</i>   | <i>French</i>  | <i>0-10</i>   | <i>T</i>        |
| $X_2$    | <i>T</i>   | <i>F</i>   | <i>F</i>   | <i>T</i>   | <i>Full</i> | <i>\$</i>     | <i>F</i>    | <i>F</i>   | <i>Thai</i>    | <i>30-60</i>  | <i>F</i>        |
| $X_3$    | <i>F</i>   | <i>T</i>   | <i>F</i>   | <i>F</i>   | <i>Some</i> | <i>\$</i>     | <i>F</i>    | <i>F</i>   | <i>Burger</i>  | <i>0-10</i>   | <i>T</i>        |
| $X_4$    | <i>T</i>   | <i>F</i>   | <i>T</i>   | <i>T</i>   | <i>Full</i> | <i>\$</i>     | <i>F</i>    | <i>F</i>   | <i>Thai</i>    | <i>10-30</i>  | <i>T</i>        |
| $X_5$    | <i>T</i>   | <i>F</i>   | <i>T</i>   | <i>F</i>   | <i>Full</i> | <i>\$\$\$</i> | <i>F</i>    | <i>T</i>   | <i>French</i>  | <i>&gt;60</i> | <i>F</i>        |
| $X_6$    | <i>F</i>   | <i>T</i>   | <i>F</i>   | <i>T</i>   | <i>Some</i> | <i>\$\$</i>   | <i>T</i>    | <i>T</i>   | <i>Italian</i> | <i>0-10</i>   | <i>T</i>        |
| $X_7$    | <i>F</i>   | <i>T</i>   | <i>F</i>   | <i>F</i>   | <i>None</i> | <i>\$</i>     | <i>T</i>    | <i>F</i>   | <i>Burger</i>  | <i>0-10</i>   | <i>F</i>        |
| $X_8$    | <i>F</i>   | <i>F</i>   | <i>F</i>   | <i>T</i>   | <i>Some</i> | <i>\$\$</i>   | <i>T</i>    | <i>T</i>   | <i>Thai</i>    | <i>0-10</i>   | <i>T</i>        |
| $X_9$    | <i>F</i>   | <i>T</i>   | <i>T</i>   | <i>F</i>   | <i>Full</i> | <i>\$</i>     | <i>T</i>    | <i>F</i>   | <i>Burger</i>  | <i>&gt;60</i> | <i>F</i>        |
| $X_{10}$ | <i>T</i>   | <i>T</i>   | <i>T</i>   | <i>T</i>   | <i>Full</i> | <i>\$\$\$</i> | <i>F</i>    | <i>T</i>   | <i>Italian</i> | <i>10-30</i>  | <i>F</i>        |
| $X_{11}$ | <i>F</i>   | <i>F</i>   | <i>F</i>   | <i>F</i>   | <i>None</i> | <i>\$</i>     | <i>F</i>    | <i>F</i>   | <i>Thai</i>    | <i>0-10</i>   | <i>F</i>        |
| $X_{12}$ | <i>T</i>   | <i>T</i>   | <i>T</i>   | <i>T</i>   | <i>Full</i> | <i>\$</i>     | <i>F</i>    | <i>F</i>   | <i>Burger</i>  | <i>30-60</i>  | <i>T</i>        |

代替案   バー   金曜  
(土曜)   空腹   既に客が居るか   雨   予約   推定待ち時間

## どのようなレストランなら「待つ」ことができるか？

上のサンプルからルールを考えてみよう。 e.g. 「値段が高いフランス料理」なら待つ？

「店がいっぱい(Pat=full)なら待たない？ 4/6」 「空腹(Hun=T)なら待つ？ 5/7」 「予約 (Res=T)なら待つ？ 3/5」

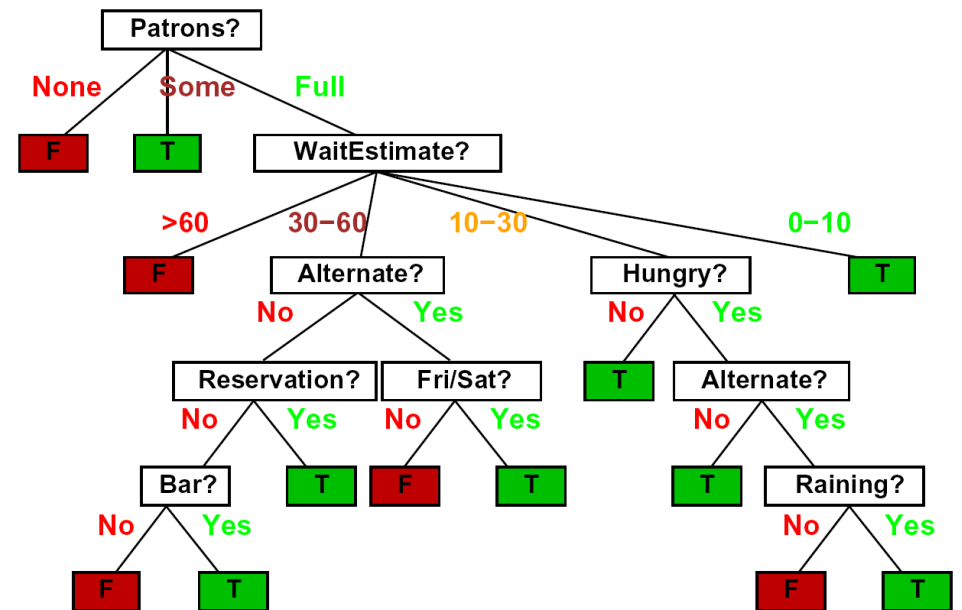
# 決定木

- 関数のコンパクトな表現:
  - 真理値表 (Truth table)
  - 条件付き確率表
  - 回帰分析の係数
- 真の関数
  - $H$  上に存在するなら、実現可能



[https://commons.wikimedia.org/wiki/File:Felix%27s\\_Bar\\_and\\_Restaurant\\_--\\_New\\_Orleans,\\_La..jpg](https://commons.wikimedia.org/wiki/File:Felix%27s_Bar_and_Restaurant_--_New_Orleans,_La..jpg)

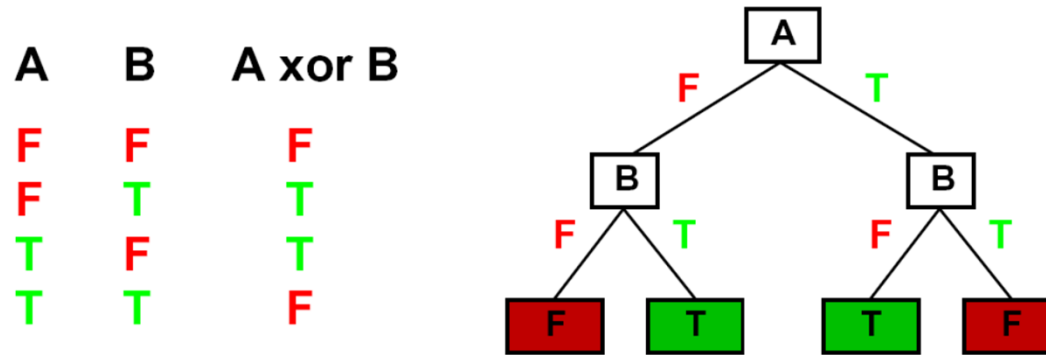
- 決定木
  - 「同定木」と呼ばれることも





# 決定木の表現能力

- 特徴に関するどのような関数も表現できる



$$P(C|A, B)$$

- ただし、なるべくコンパクトな木を得たい

# 分類法との比較

- 以下の特徴に関するパーセプトロンの表現能力は？

| Example | Attributes |            |            |            |             |               |             |            |               |              | Target          |
|---------|------------|------------|------------|------------|-------------|---------------|-------------|------------|---------------|--------------|-----------------|
|         | <i>Alt</i> | <i>Bar</i> | <i>Fri</i> | <i>Hun</i> | <i>Pat</i>  | <i>Price</i>  | <i>Rain</i> | <i>Res</i> | <i>Type</i>   | <i>Est</i>   | <i>WillWait</i> |
| $X_1$   | <i>T</i>   | <i>F</i>   | <i>F</i>   | <i>T</i>   | <i>Some</i> | <i>\$\$\$</i> | <i>F</i>    | <i>T</i>   | <i>French</i> | <i>0-10</i>  | <i>T</i>        |
| $X_2$   | <i>T</i>   | <i>F</i>   | <i>F</i>   | <i>T</i>   | <i>Full</i> | <i>\$</i>     | <i>F</i>    | <i>F</i>   | <i>Thai</i>   | <i>30-60</i> | <i>F</i>        |

- パーセプトロンでは、特徴の貢献の仕方は正か負のみ
  - もし、ある特徴の効果が他の特徴次第（e.g. XOR問題）であれば、新しく結合した特徴を足す必要があった
  - E.g. “PATRONS=full  $\wedge$  WAIT = 60”の特徴を新たに足すことで、これらの原始的な特徴の間の相互作用をパーセプトロンが扱えるようになる
- 決定木は、自動的に特徴/属性を結合する
  - 特徴は、木の別の枝において、異なった効果を持ちうる
- 証拠の相対的な重みのモデル化（単純ベイズ）と複雑な相互作用（決定木）の違い
  - ただし、もし相互作用が複雑すぎたら、適切な決定木が見つけれない場合も

# 仮説空間 (Hypothesis Spaces)

- $n$  個の Boolean 属性のもとで、何種類の決定木が作りうるか？

- =  $n$  属性のもとでの Boolean関数の数

- =  $2^n$  行ある真理値表の数

- =  $2^{(2^n)}$

- E.g., 6個の Boolean 属性に対しては,  
18,446,744,073,709,551,616 種類の決定木

- 深さ 1 の木の数は？ (決定株)?

- = 1 属性の Boolean 関数の数

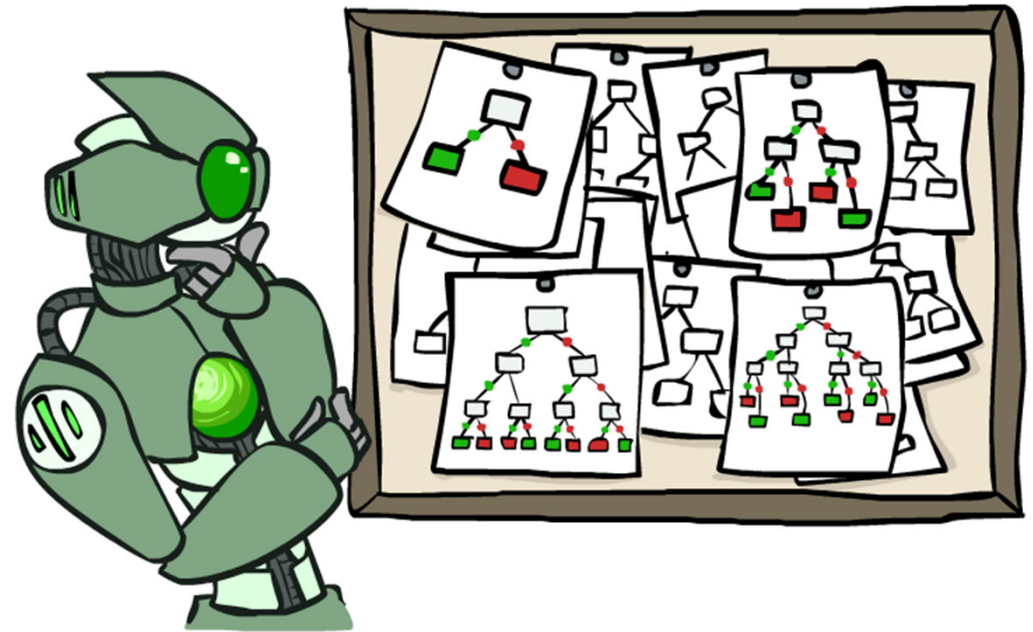
- = 2 行の真理値表  $\times n$  個

- =  $4n$

- E.g. 6個の Boolean 属性に対して、24 種類の決定株

- 仮説空間の表現力が増えると？

- 対象の関数が正しく表現される可能性が高まる (良い)
  - 訓練データ集合と一貫した仮説が増える (良くない, なぜか?)
  - 正しい予測ができる可能性が高まる (**バイアス(bias)** が少なくなる)
  - 間違った予測が起きる可能性も (**分散(variance)** の高まり)



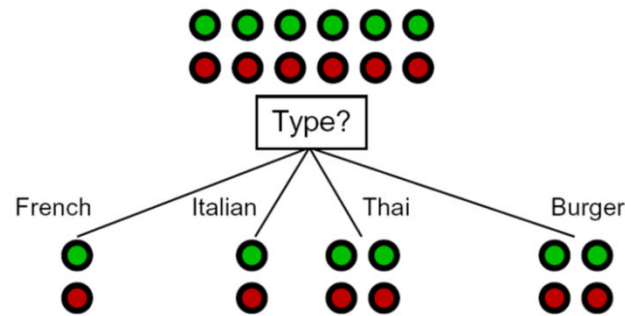
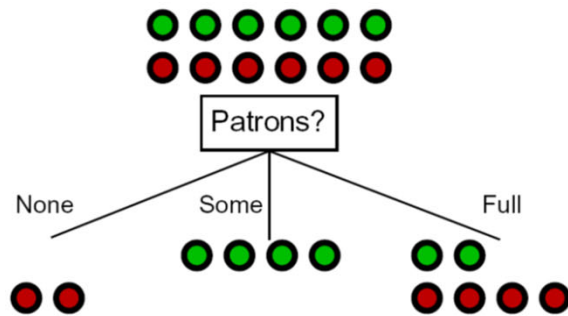
# 決定木の学習

- 目的: 訓練データと一貫した小さな木を見つける
- アイデア: (再帰的に) それぞれの部分木において “最も重要な” 属性を選択

```
function DTL(examples, attributes, default) returns a decision tree
  if examples is empty then return default
  else if all examples have the same classification then return the classification
  else if attributes is empty then return MODE(examples)
  else
    best ← CHOOSE-ATTRIBUTE(attributes, examples)
    tree ← a new decision tree with root test best
    for each value  $v_i$  of best do
      examplesi ← {elements of examples with best =  $v_i$ }
      subtree ← DTL(examplesi, attributes − best, MODE(examples))
      add a branch to tree with label  $v_i$  and subtree subtree
  return tree
```

# 属性の選択

- アイデア: 良い属性は、すべての例を（理想的には）「すべて正」「すべて負」となる部分木に分ける



- 結果が完全な分離にならなかったとしても、分割がどれくらい「良い」かの指標が必要

# エントロピーと情報 (Entropy and Information)

- 情報(量)によって答えが得られる
  - 事前に答えがより不確かであれば、答えを知るとより多くの情報が得られる
  - 単位: ビット (平均の通信量)
    - 事前確率が  $\langle 1/2, 1/2 \rangle$  の2択質問への解は?
    - 事前確率が  $\langle 1/4, 1/4, 1/4, 1/4 \rangle$  の4択質問への解は?
    - 事前確率が  $\langle 0, 0, 0, 1 \rangle$  の4択質問への解は?
    - 事前確率が  $\langle 1/2, 1/4, 1/4 \rangle$  の3択質問への解は?
- 確率  $p$  の均一分布の事象に必要なコード長は?  
 $\log 1/p$  の符号長 (A code of length)  
$$\sum p \log 1/p = \log 1/p$$

# エントロピー

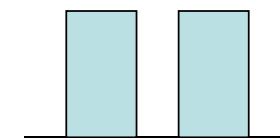
- 一般解: もし事前確率が  $\langle p_1, \dots, p_n \rangle$  であれば:

- 情報量の期待符号長

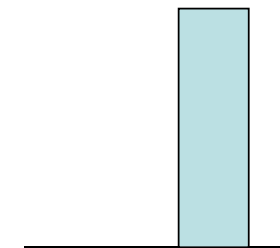
$$H(\langle p_1, \dots, p_n \rangle) = \sum_{i=1}^n -p_i \log_2 p_i$$

- あるいは分布の **エントロピー (entropy)** と呼ばれる

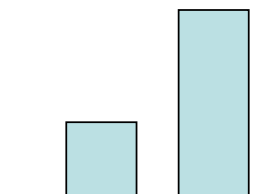
- より均一 = エントロピーはより高い
- 多くの値 = エントロピーは高い
- 多くのピーク = エントロピーは低い
- まれな値はほぼ関係しない



1 bit



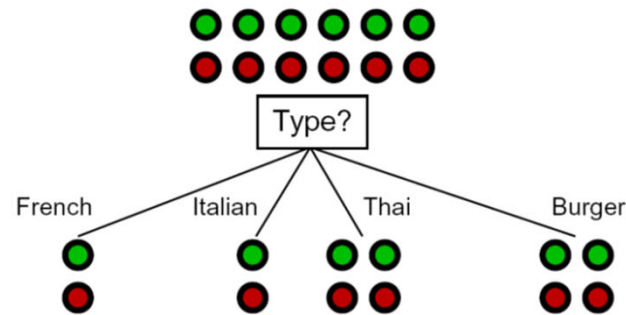
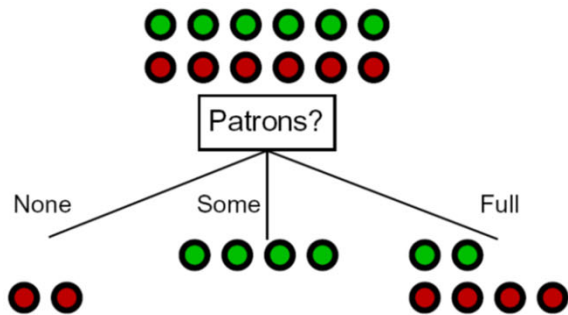
0 bits



0.5 bit

# 情報利得 (Information Gain)

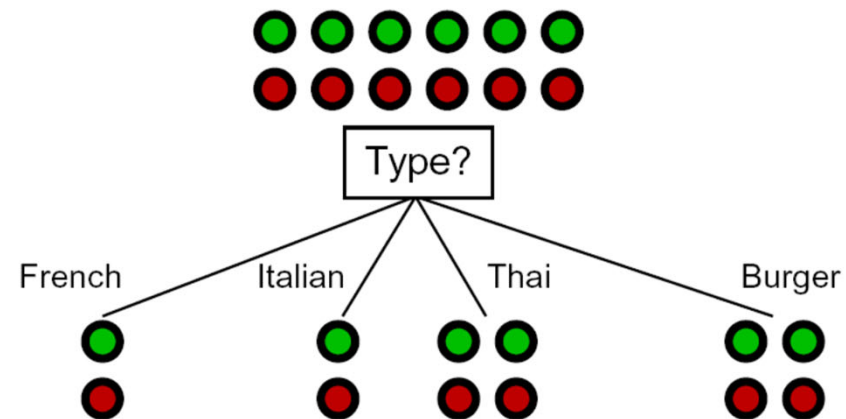
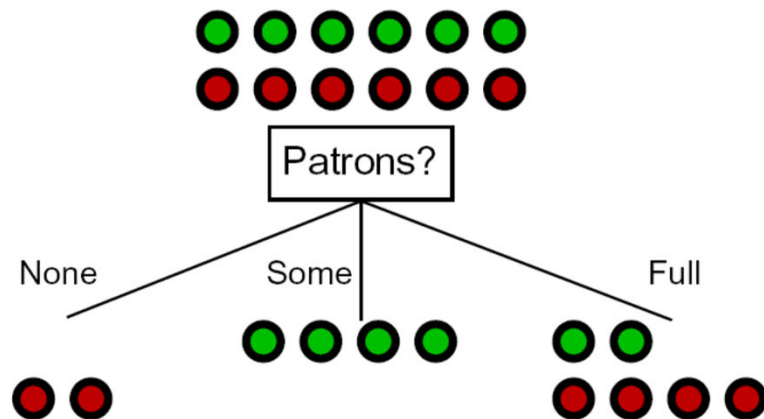
- 決定木に戻ると？
- 分割の度に、前後のエントロピーを比べる
  - 差分は **情報利得 (information gain)**
  - 問題: 分割後には複数の分布が存在する！
  - 解法: **期待エントロピー (expected entropy)** を用いる
    - ・・・ 例の数で重みづけしたもの





# 演習：情報利得と属性の選択

- 実際に以下の分割について、情報利得を計算した上で、どちらの分割が望ましいか選択しよう



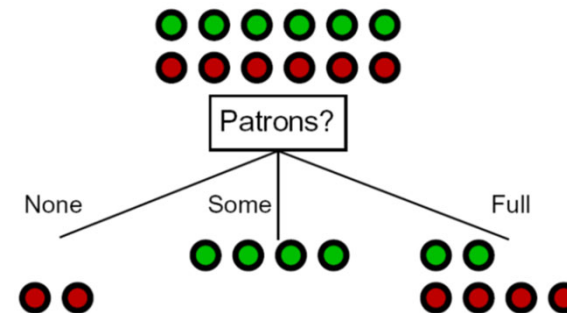
参考：エントロピー

$$H(\langle p_1, \dots, p_n \rangle) = \sum_{i=1}^n -p_i \log_2 p_i$$

$\log_2 3 = 1.58$  としてよい

# 次のステップ：再帰

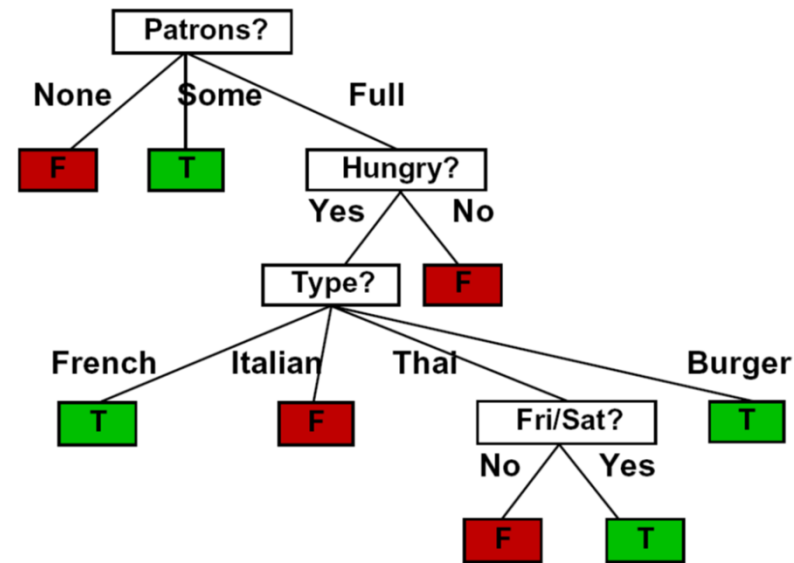
- 木をさらに伸ばしていく
- 2つの枝はこれで完了 (なぜか?)
- 枝 “full” についてはどうするか?
  - どのような例が含まれるか見てみよう...



| Example  | Attributes |     |     |     |      |        |      |     |         |       | Target<br>WillWait |
|----------|------------|-----|-----|-----|------|--------|------|-----|---------|-------|--------------------|
|          | Alt        | Bar | Fri | Hun | Pat  | Price  | Rain | Res | Type    | Est   |                    |
| $X_1$    | T          | F   | F   | T   | Some | \$\$\$ | F    | T   | French  | 0-10  | T                  |
| $X_2$    | T          | F   | F   | T   | Full | \$     | F    | F   | Thai    | 30-60 | F                  |
| $X_3$    | F          | T   | F   | F   | Some | \$     | F    | F   | Burger  | 0-10  | T                  |
| $X_4$    | T          | F   | T   | T   | Full | \$     | F    | F   | Thai    | 10-30 | T                  |
| $X_5$    | T          | F   | T   | F   | Full | \$\$\$ | F    | T   | French  | >60   | F                  |
| $X_6$    | F          | T   | F   | T   | Some | \$\$   | T    | T   | Italian | 0-10  | T                  |
| $X_7$    | F          | T   | F   | F   | None | \$     | T    | F   | Burger  | 0-10  | F                  |
| $X_8$    | F          | F   | F   | T   | Some | \$\$   | T    | T   | Thai    | 0-10  | T                  |
| $X_9$    | F          | T   | T   | F   | Full | \$     | T    | F   | Burger  | >60   | F                  |
| $X_{10}$ | T          | T   | T   | T   | Full | \$\$\$ | F    | T   | Italian | 10-30 | F                  |
| $X_{11}$ | F          | F   | F   | F   | None | \$     | F    | F   | Thai    | 0-10  | F                  |
| $X_{12}$ | T          | T   | T   | T   | Full | \$     | F    | F   | Burger  | 30-60 | T                  |

# 決定木の学習例

- これらの12の例から学習した決定木:



- 本質的には、シンプルというよりも、「正しさ」を重視した木
  - これよりも複雑な仮説はデータによってサポートされない
- この木は正しそうにも見えるが、誤っているかもしれない

## 例題 2 : 燃費 (MPG: Miles Per Gallon)

40 Examples

| mpg  | cylinders | displacement | horsepower | weight | acceleration | modelyear | maker   |
|------|-----------|--------------|------------|--------|--------------|-----------|---------|
| good | 4         | low          | low        | low    | high         | 75to78    | asia    |
| bad  | 6         | medium       | medium     | medium | medium       | 70to74    | america |
| bad  | 4         | medium       | medium     | medium | low          | 75to78    | europa  |
| bad  | 8         | high         | high       | high   | low          | 70to74    | america |
| bad  | 6         | medium       | medium     | medium | medium       | 70to74    | america |
| bad  | 4         | low          | medium     | low    | medium       | 70to74    | asia    |
| bad  | 4         | low          | medium     | low    | low          | 70to74    | asia    |
| bad  | 8         | high         | high       | high   | low          | 75to78    | america |
| :    | :         | :            | :          | :      | :            | :         | :       |
| :    | :         | :            | :          | :      | :            | :         | :       |
| :    | :         | :            | :          | :      | :            | :         | :       |
| bad  | 8         | high         | high       | high   | low          | 70to74    | america |
| good | 8         | high         | medium     | high   | high         | 79to83    | america |
| bad  | 8         | high         | high       | high   | low          | 75to78    | america |
| good | 4         | low          | low        | low    | low          | 79to83    | america |
| bad  | 6         | medium       | medium     | medium | high         | 75to78    | america |
| good | 4         | medium       | low        | low    | low          | 79to83    | america |
| good | 4         | low          | low        | medium | high         | 79to83    | america |
| bad  | 8         | high         | high       | high   | low          | 70to74    | america |
| good | 4         | low          | medium     | low    | medium       | 75to78    | europa  |
| bad  | 5         | medium       | medium     | medium | medium       | 75to78    | europa  |

mpg: 燃費 (miles per gallons)

cylinders: 気筒(数)

displacement: 総排気量

horsepower: 馬力

weight: 重さ

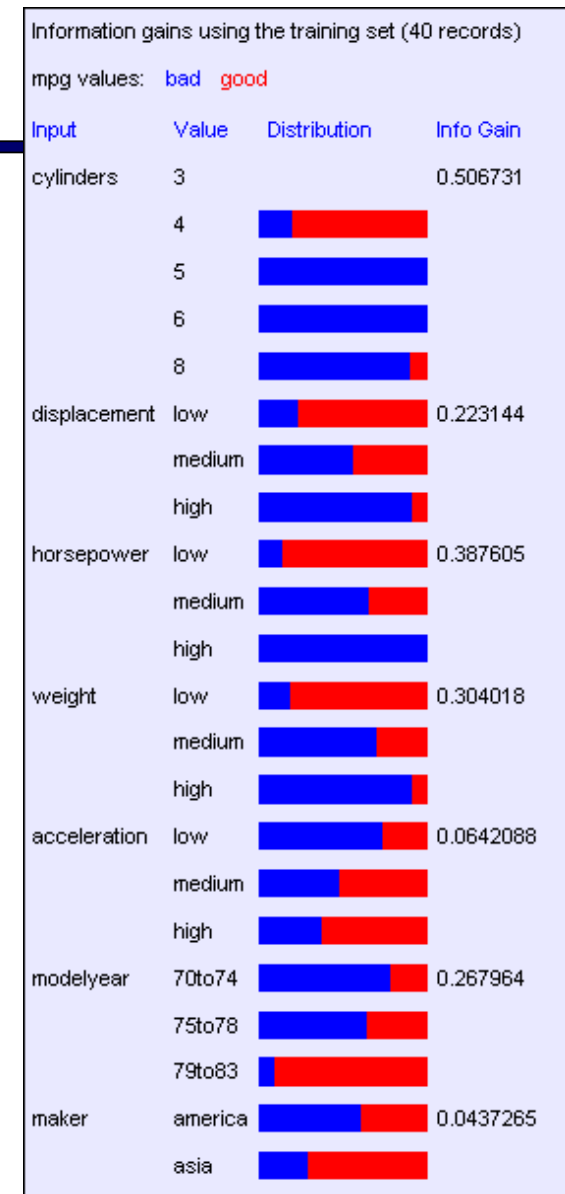
acceleration: 加速

modelyear: 製造年

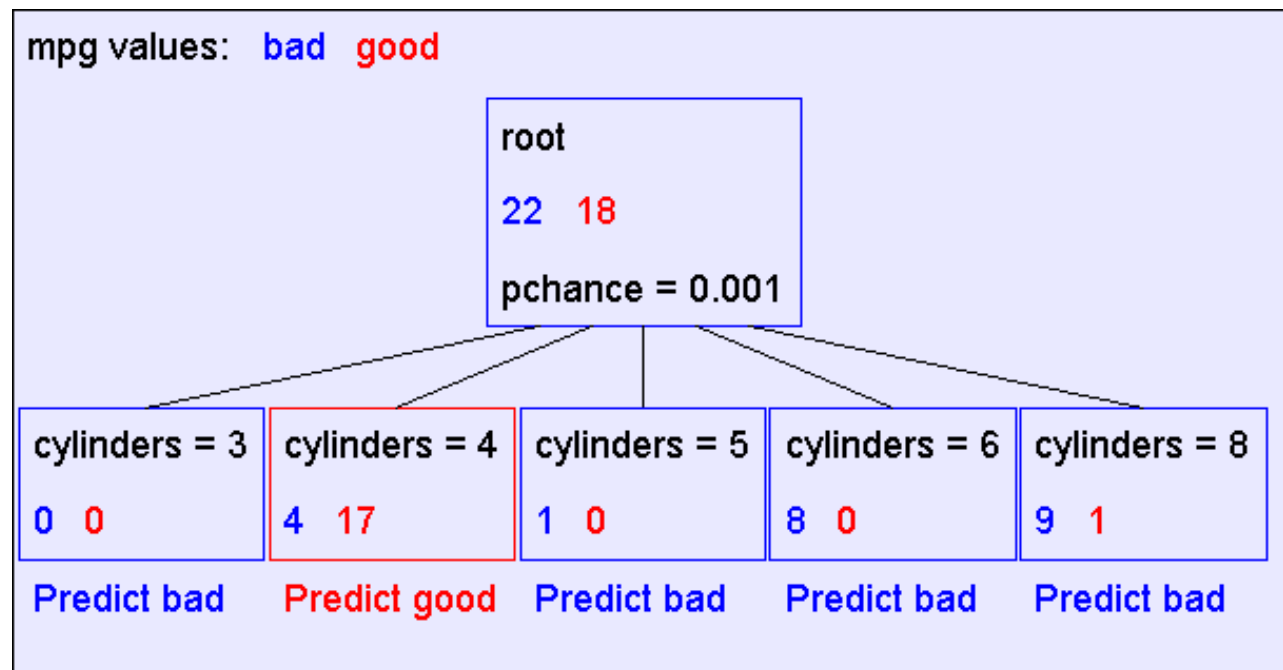
maker: 製造地域

# 最初の分割を見つける

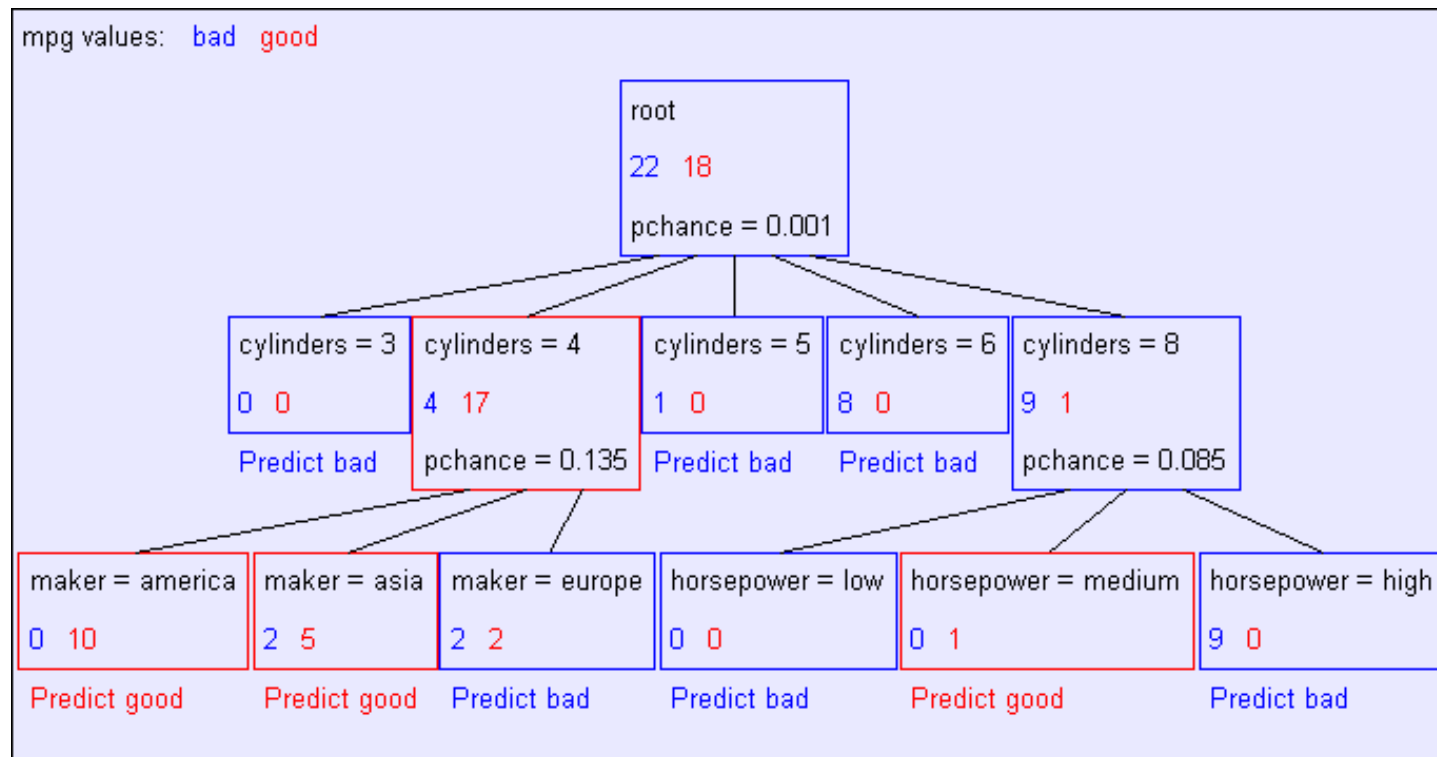
- 各属性について、情報利得をチェック
- 各属性は、目的変数と相関がある!
- どこで分割すべきか?



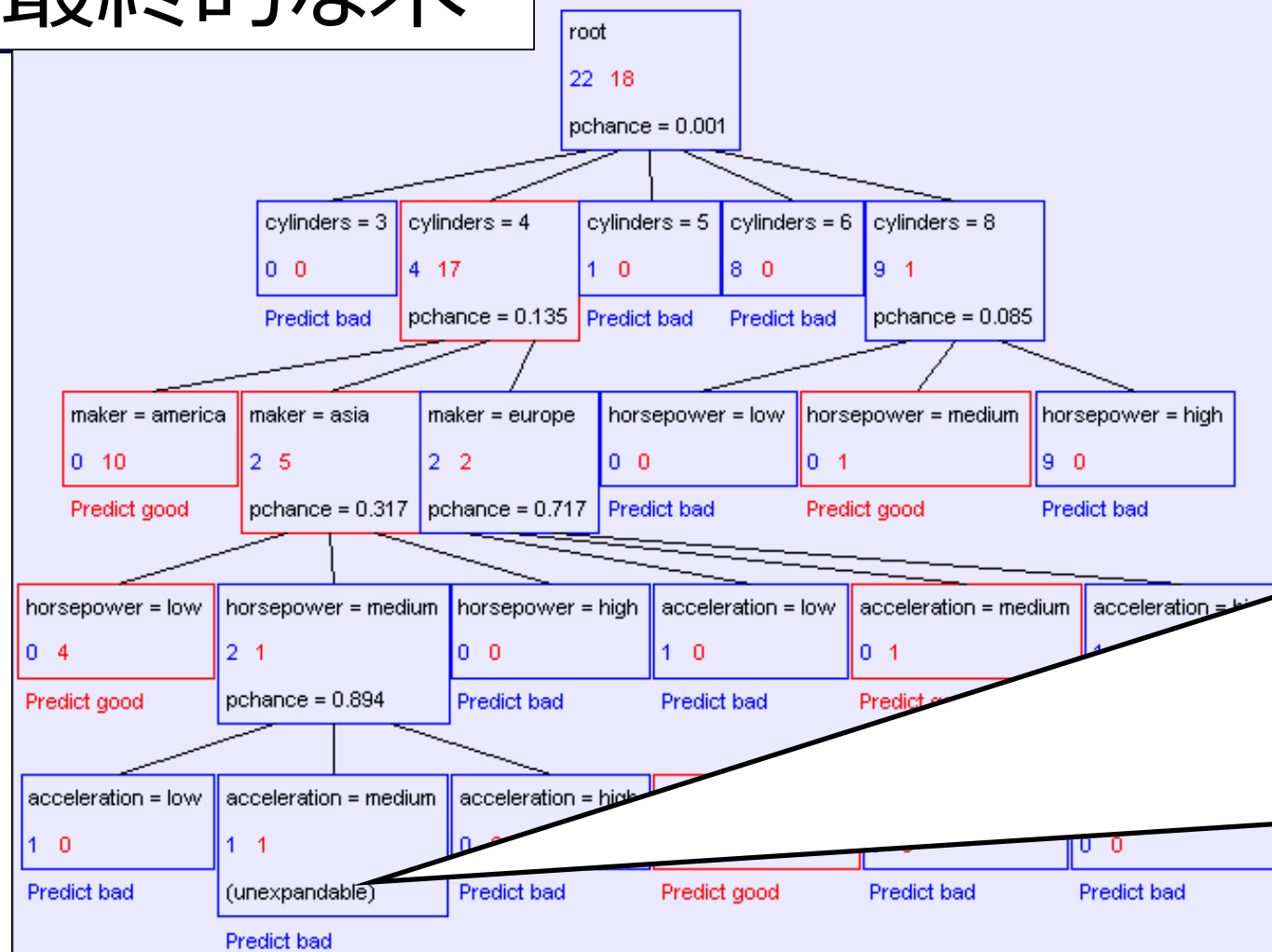
# 結果: 最初のレベル



## 2つ目のレベル



# 最終的な木



Information gains using the training set (2 records)

mpg values: bad good

| Input        | Value   | Distribution                      | Info Gain |
|--------------|---------|-----------------------------------|-----------|
| cylinders    | 3       |                                   | 0         |
|              | 4       | <div><div></div><div></div></div> |           |
|              | 5       |                                   |           |
|              | 6       |                                   |           |
|              | 8       |                                   |           |
| displacement | low     | <div><div></div><div></div></div> | 0         |
|              | medium  |                                   |           |
|              | high    |                                   |           |
| horsepower   | low     |                                   | 0         |
|              | medium  | <div><div></div><div></div></div> |           |
|              | high    |                                   |           |
| weight       | low     | <div><div></div><div></div></div> | 0         |
|              | medium  |                                   |           |
|              | high    |                                   |           |
| acceleration | low     |                                   | 0         |
|              | medium  | <div><div></div><div></div></div> |           |
|              | high    |                                   |           |
| modelyear    | 70to74  | <div><div></div><div></div></div> | 0         |
|              | 75to78  |                                   |           |
|              | 79to83  |                                   |           |
| maker        | america |                                   | 0         |
|              | asia    | <div><div></div><div></div></div> |           |
|              | europe  |                                   |           |



# 復習: オーバーフィッティング

---

- オーバーフィッティング:

- いつ訓練データからのパターンの（一般化できる）モデル化がおわるか？
- そして、いつ、ただのノイズのモデル化が始まるか？

- 以前も類似の問題があった:

- 単純ベイズ分類器: スムージングが必要であった
- パーセプトロン / ニューラルネットワーク:  
早期停止 (early stopping)

# MPG Training Error

mpg values: bad good

root  
22 18  
pchance = 0.001

|                | Num Errors | Set Size | Percent Wrong |
|----------------|------------|----------|---------------|
| Training Set 1 | 1          | 40       | 2.50          |
| Test Set       | 74         | 352      | 21.02         |

horsepower = high

Predict bad

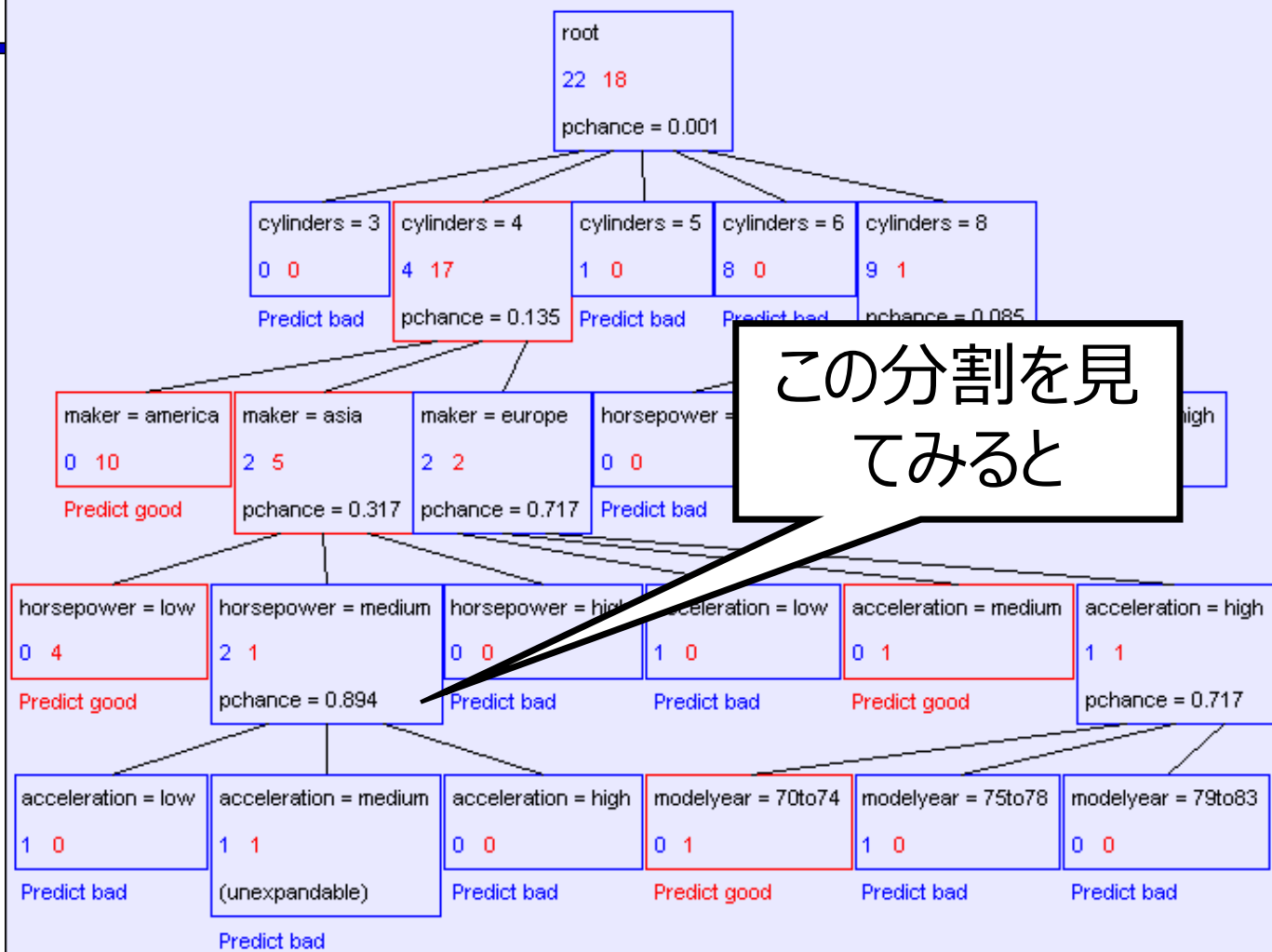
horsepower = low horsepower = medium horsepower = high acceleration = low acceleration = medium acceleration = high

訓練セットでのエラーよりもテストセットのエラーの方がずっと大きい...

...なぜか?

Predict bad (unexpandable) Predict bad Predict good Predict bad Predict bad

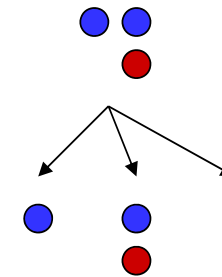
mpg values: bad good



# 分割の重要度 (Significance of a Split)

- 検討する事例:

- 4つのシリンダを持ち、Asia製、中程度の馬力の車は3種類
- 2種類は燃費(MPG)が悪い
- 1種類は燃費(MPG)が良い



- この3分割から何を期待するか？

- 例だけでなく、一般に、このように分割される？
- この3例以外の例は、他の枝にも分割されていく？

- これは偶然によるものかもしれないので、要素数がこれほど小さいなら分割すべきではないのでは？

- カイ二乗検定により、この完全な分割からの逸脱が偶然かどうかを推測できる

- 各分割は **重要度値 (significance value)**,  $p_{\text{CHANCE}}$  を持つ

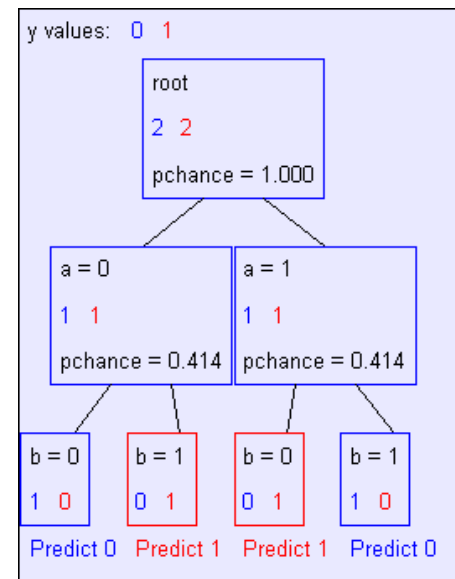
# 一般性の維持 (Keeping it General)

## ■ 枝刈り:

- 完全な決定木を作る
- 木の底（葉節点）から開始
- 以下を満たす分割を削除
  - $p_{\text{CHANCE}} > \text{Max} p_{\text{CHANCE}}$
- 木を上にとどってゆき、枝刈り可能な節点がなくなるまで続ける
- 注意: 重要度値が低くても、後で「回復」して枝刈りされない節点もある (e.g. XOR)

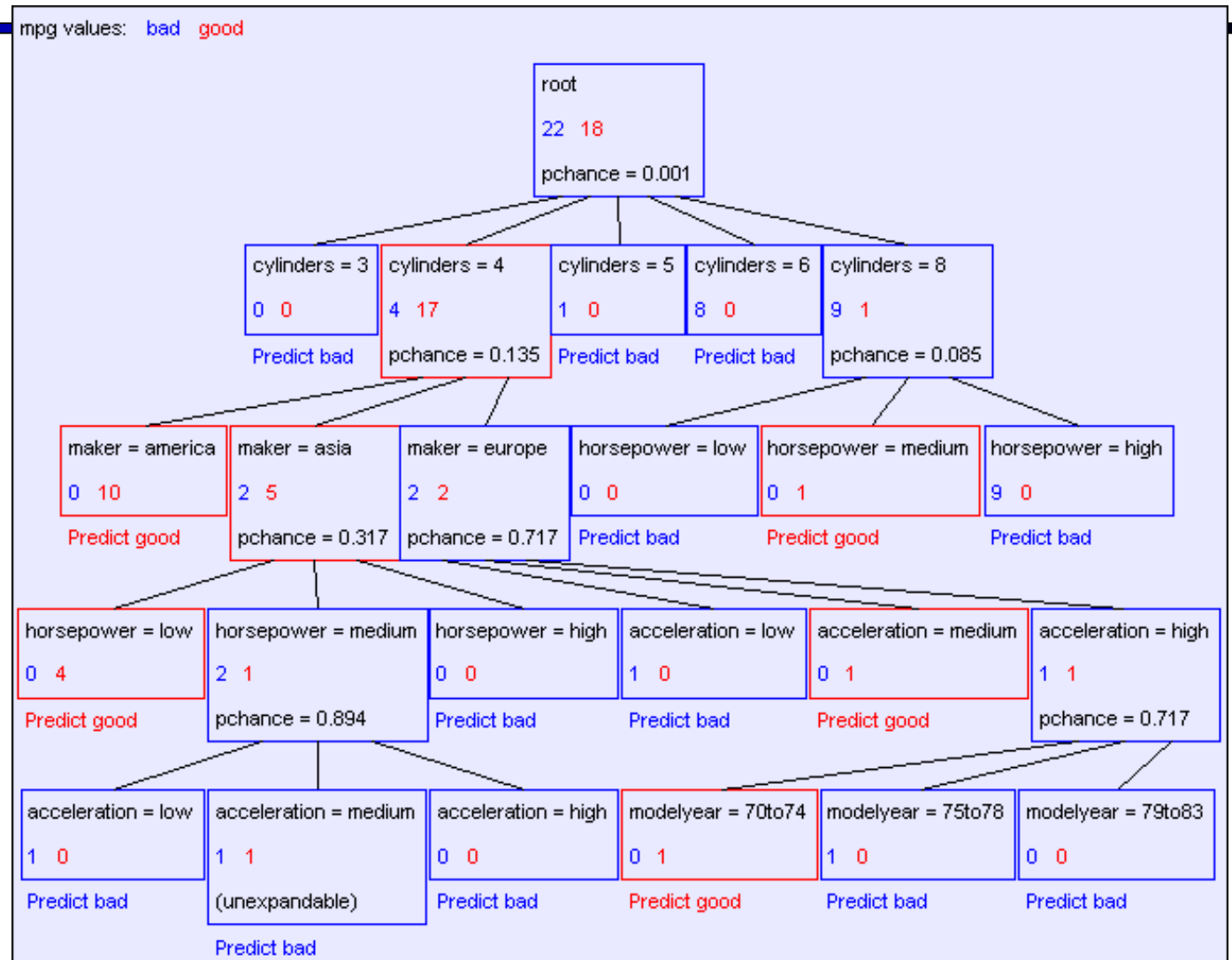
$$y = a \text{ XOR } b$$

| a | b | y |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

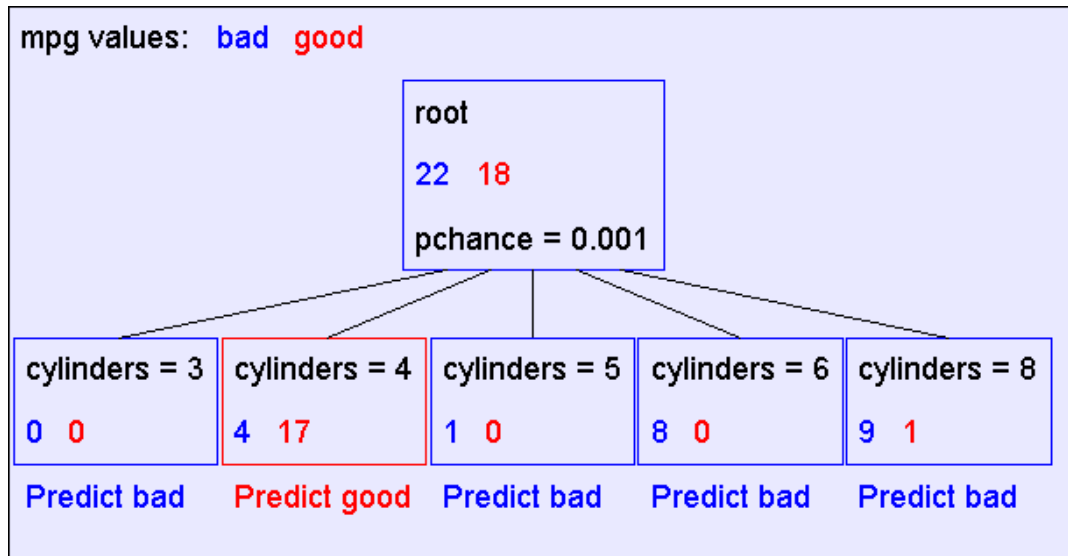


# 演習：枝刈り

- $\text{MaxP}_{\text{CHANCE}} = 0.20$  として  
枝刈りをしてみよう
- $\text{MaxP}_{\text{CHANCE}} = 0.05$  の場合  
はどうか？



# 枝刈りの効果

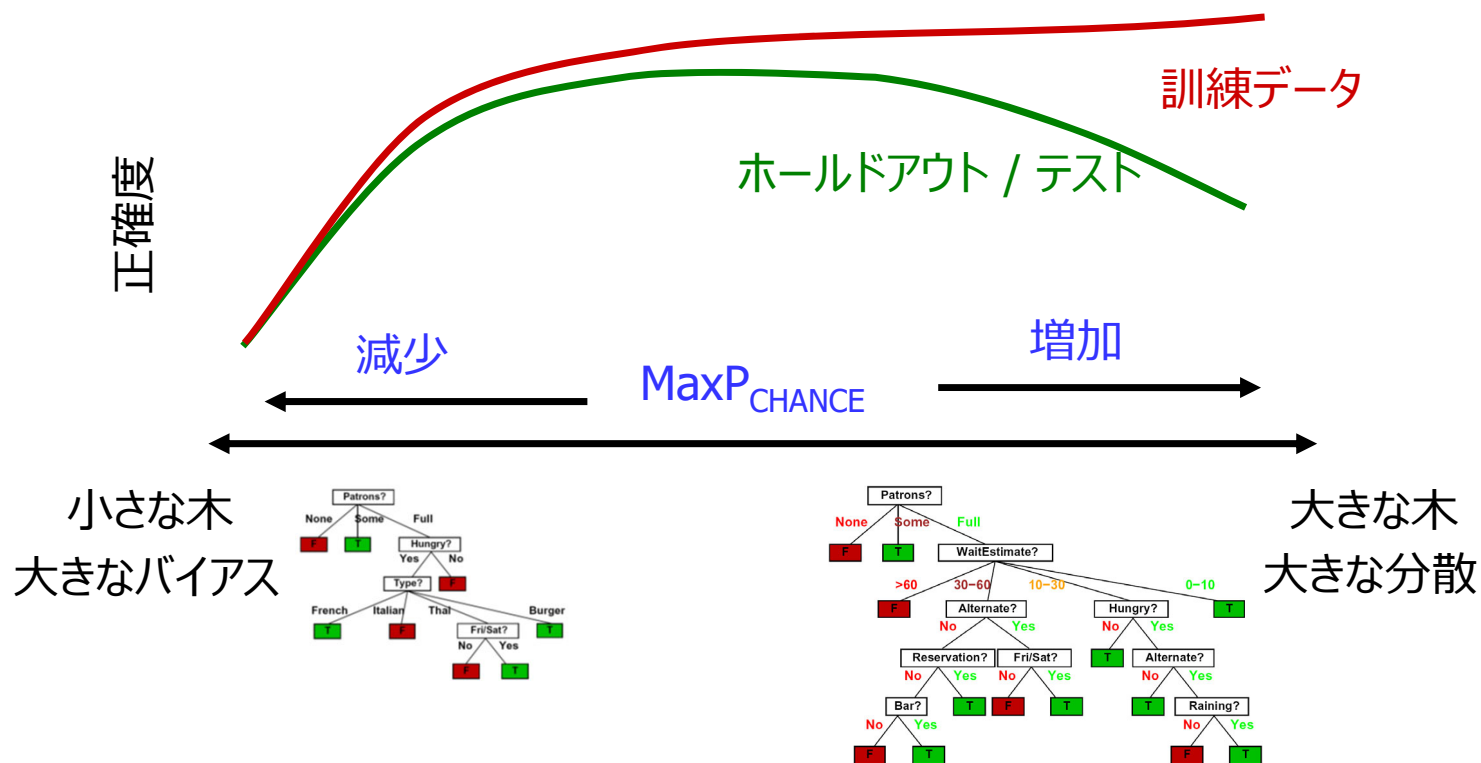


枝刈りしていない木  
よりもテストデータに  
対して高い性能

|              | Num Errors | Set Size | Percent Wrong |
|--------------|------------|----------|---------------|
| Training Set | 5          | 40       | 12.50         |
| Test Set     | 56         | 352      | 15.91         |

# 正則化 (Regularization)

- $\text{MaxP}_{\text{CHANCE}}$  は正則化 (regularization) パラメータ
- 一般的に, ホールドアウトデータを使って設定





# オーバーフィットを抑制する2つの方法

---

- 仮説空間の制約

- E.g. 木の最大の深さを制限する
- 分析がしやすい

- 仮説選択での正則化

- E.g. 枝刈りのカットオフ
- データがはっきりとしないような仮説を除外
- 実際に行われることが多い

# 発展的トピック

## ■ ランダムフォレスト (Random forest)

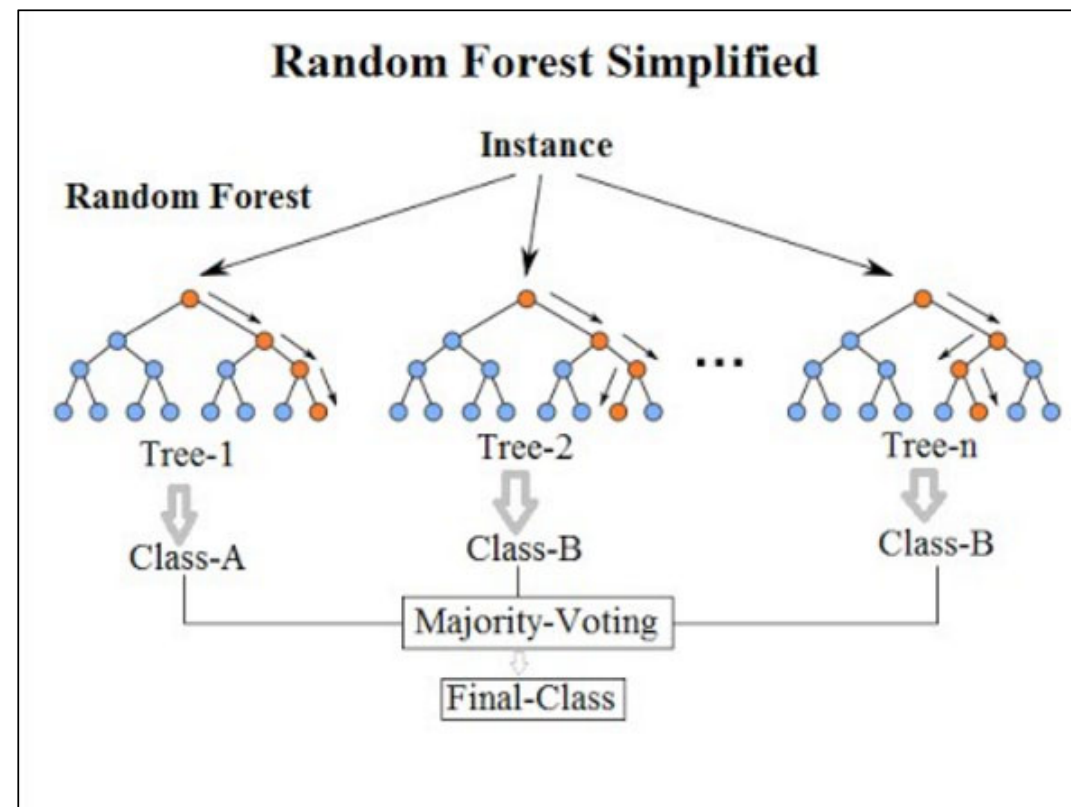
- 1つの決定木では過学習が起きやすい
- その代わりに、木自体を「サンプリング」により作れないか？



(ランダムに生成した) 決定木により  
「森」ができる



多数決で判別



[https://en.wikipedia.org/wiki/Random\\_forest](https://en.wikipedia.org/wiki/Random_forest) より

# 引用文献

---

- S. Russell and P. Norvig, Artificial Intelligence A Modern Approach, 3rd edition, Pearson Education Limited, 2016.
- UC Berkeley CS188 Intro to AI -- Course Materials  
<http://ai.berkeley.edu/>