知能システム開発特論:
第6回

データマイニング:
決定木、

ダビド (david@iwate-pu.ac.jp)
IPU
25 January 2021

データマイニングは機械学習か？

- 分類器
- クラスター器
- 回帰器
- ニュラルネット
- PCA
- 決定木器
- …

知識を獲得したい！！！

学習の手法

- 情報から従来使えそうな知識をどんな方法で見つけ出す
- 例題（事例）からアルゴリズムに基づいて概念記述（structural descriptions)を取得
- 概念記述には明示的にパターンを定義される

データマイニングの機能

- 分類と予測
  - –予測を目的として、クラスや概念を区別するための記述を求める。
  - –例国を気候により分類する。自動車を燃費で分類する。
  - –表現: 決定木、分類規則、ニューラルネットワーク
  - –予測:未知の、あるいは欠落した数値(missing values)を予測する。
- クラスター分析
  - –分類カテゴリが未知: 新しいクラスを作るためにグループ化する。たとえば、分布パターンを見つけるために顧客の住居をクラスター化する。
  - –クラスタリングの原理: クラス内の類似性を最大化し、同時に、クラス間の類似性を最小にする。

決定木

- データの分類を多段階で繰り返し実施する場合に有用な木構造を指す。

- 意図決定に利用されるが、未知のデータに対するクラス属性の値を決定することが可能である。
- データセットDが与える時、n サンプルのx_iと$y_i \epsilon \{c_1, c_2, \ldots, c_k\}$、$X_j$が量的属性または質的属性
- 決定木分類器は新たなサンプルx　の$\hat{\mathrm{y}}$クラスを予測する。
- 決定木の構築にあたり、決定木の分岐ノードの分割属性を選択するため評価基準として情報利得を導入する

決定木

- 決定木分類器は軸並行超平面(axis-parallel hyperplane)を用いてデータ空間Rをデータ空間$R1, R2$に再帰的に分割する。

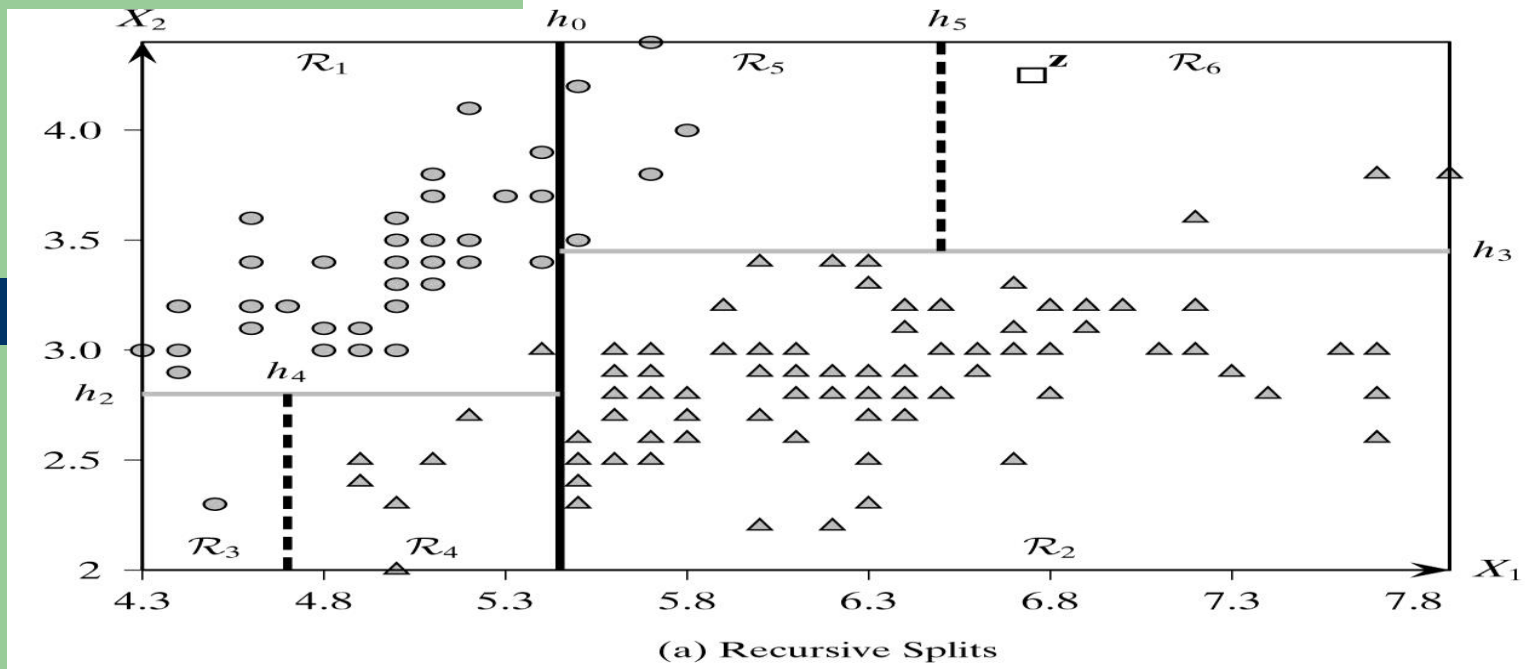
- 決定木分類器は再帰的にテストデータを半空間に分類し、葉ノードまで行う。葉ノードのラベルがクラスとなる。

Irisデータセットの例

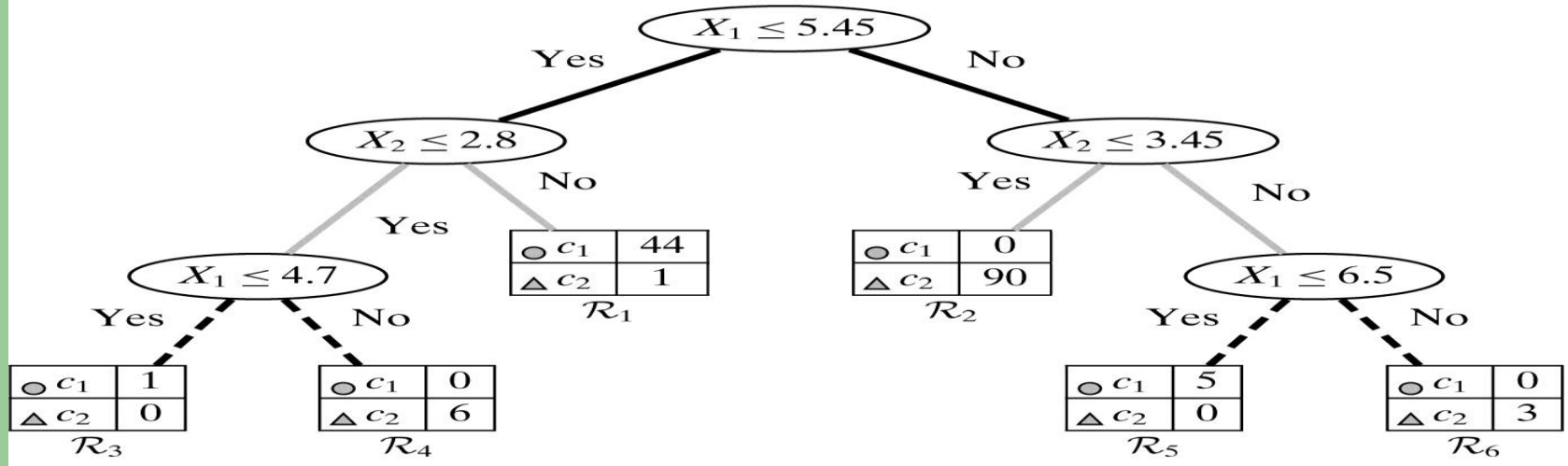
- 2次元のデータセット、属性Iris-sepal-lengthとIris-sepal-width、クラスc_1はIris-Setosa、

  クラスc_2はその他。
- $n_1 = 50, n_2 = 100$
- $R = range(X_1) \times range(X_2) = [4.3,7.9] \times [2.0,4.4]$
- h_0, h_1, h_3, h_4超平面
- $R_1, R_2, R_3, R_4, R_5, R_6$データ空間と決定木モデル
- $\mathbf{x} = (6.75, 4.25)^T$ =z を与える時、クラスを予測する

決定木分類の例



(a) Recursive Splits



(b) Decision Tree

決定木

- 軸並行超平面(axis-parallel hyperplane)

$$h(\mathrm{x}): w^T \mathrm{x} + b = 0$$

- 分ける点(split points)

$$X_j \leq v, v = -b$$

- データパーティション(data partition)$R\ to\ R_Y\ and\ R_N$

$$D_Y = \{\mathrm{x} | \mathrm{x} \in D, x_j \leq v\}$$

$$D_N = \{\mathrm{x} | \mathrm{x} \in D, x_j > v\}$$

- 純度(Purity)

$$purity(D_j) = \max_i \{\frac{n_{ji}}{n_j}\}$$

-

Irisデータセットの例

- 決定木から決定ルール

$\mathcal{R}_3$: If $X_1 \leq 5.45$ and $X_2 \leq 2.8$ and $X_1 \leq 4.7$, then class is c_1, or

$\mathcal{R}_4$: If $X_1 \leq 5.45$ and $X_2 \leq 2.8$ and $X_1 > 4.7$, then class is c_2, or

$\mathcal{R}_1$: If $X_1 \leq 5.45$ and $X_2 > 2.8$, then class is c_1, or

$\mathcal{R}_2$: If $X_1 > 5.45$ and $X_2 \leq 3.45$, then class is c_2, or

$\mathcal{R}_5$: If $X_1 > 5.45$ and $X_2 > 3.45$ and $X_1 \leq 6.5$, then class is c_1, or

$\mathcal{R}_6$: If $X_1 > 5.45$ and $X_2 > 3.45$ and $X_1 > 6.5$, then class is c_2

1. 評価基準

- 情報利得の計算のため、利用される尺度:
  - エントロピー

$$H(D) = -\sum_{i=1}^{k} P(c_i|D) \log_2 P(c_i|D)$$

情報利得(Information Gain):

$$Gain(D, D_Y, D_N) = H(D) - H(D_Y, D_N)$$

$$H(D_Y, D_N) = \frac{n_Y}{n} H(D_Y) + \frac{n_N}{n} H(D_N)$$

1. 評価基準

- 情報利得の計算のため、利用される尺度：
  - ジニ係数

$$G(D) = 1 - \sum_{i=1}^{k} P(c_i|D)^2$$

ジニ係数(Gini Index)の分割点：

$$G(D_Y, D_N) = \frac{n_Y}{n} G(D_Y) + \frac{n_N}{n} G(D_N)$$

  - CART（Classification And Regression Trees）

$$CART(\mathbf{D}_Y, \mathbf{D}_N) = 2\frac{n_Y}{n}\frac{n_N}{n}\sum_{i=1}^{k}\left|P(c_i|\mathbf{D}_Y) - P(c_i|\mathbf{D}_N)\right|$$

Irisデータセットの例

- 2次元のデータセット、属性Iris-sepal-lengthとIris-sepal-width、クラスc_1はIris-Setosa、

  クラスc_2はその他。
- $n_1 = 50, n_2 = 100$
- $\mathbf{x} = (6.75, 4.25)^T$ を与える時、クラスを予測する
- $\widehat{P}(c_1) = \frac{1}{3}, \widehat{P}(c_2) = \frac{2}{3}, N_{v1} = 45, N_{v2} = 7$

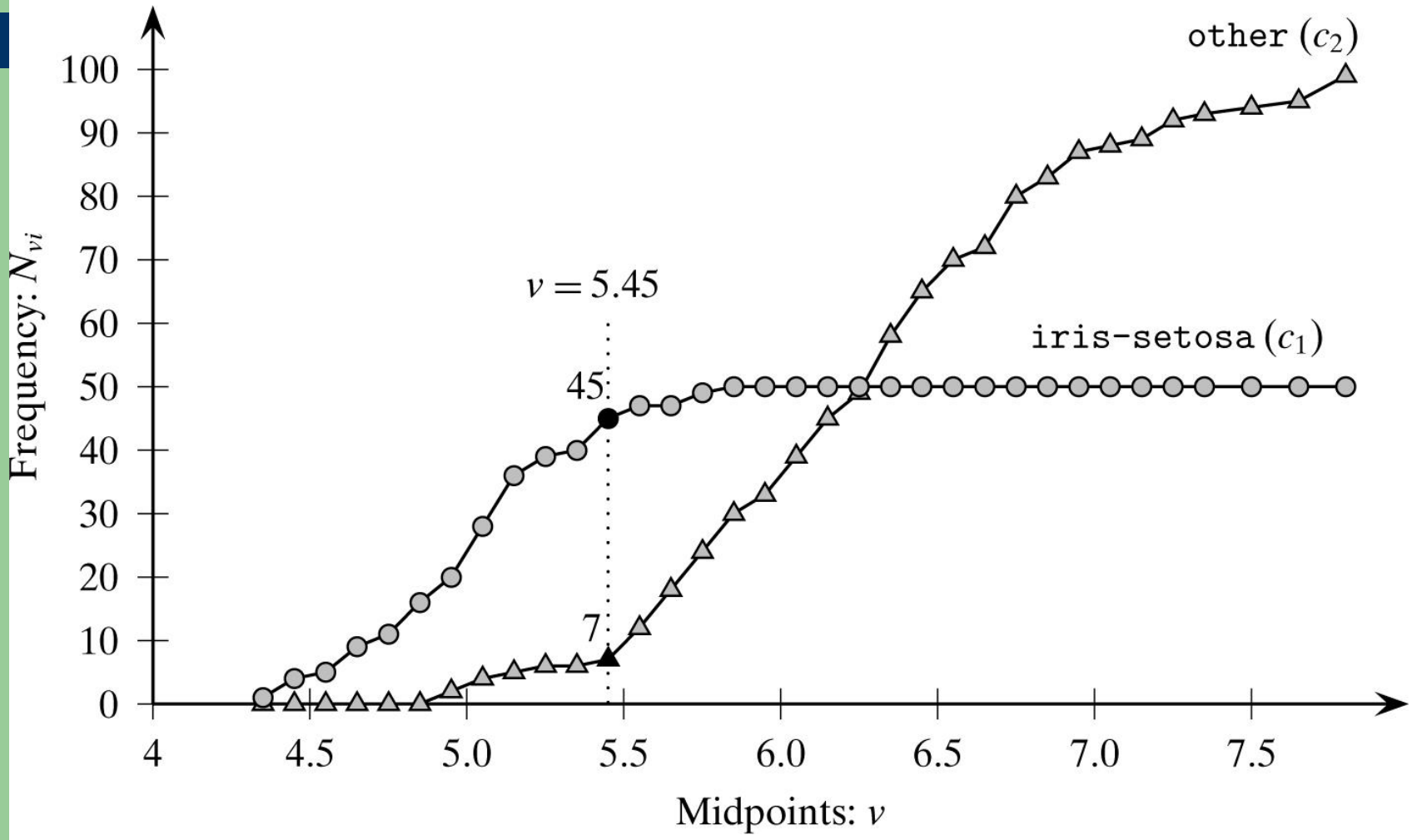
$$H(\mathbf{D}) = -\left(\frac{1}{3}\log_2\frac{1}{3} + \frac{2}{3}\log_2\frac{2}{3}\right) = 0.918$$

$$\hat{P}(c_1|\mathbf{D}_Y) = \frac{N_{v1}}{N_{v1}+N_{v2}} = \frac{45}{45+7} = 0.865$$

$$\hat{P}(c_2|\mathbf{D}_Y) = \frac{N_{v2}}{N_{v1}+N_{v2}} = \frac{7}{45+7} = 0.135$$

-

クラスc1,c2について、sepal_lengthの頻度グラフ



Irisデータセットの例

$$\hat{P}(c_1|\mathbf{D}_N) = \frac{n_1 - N_{v1}}{(n_1 - N_{v1}) + (n_2 - N_{v2})} = \frac{50 - 45}{(50 - 45) + (100 - 7)} = 0.051$$

$$\hat{P}(c_2|\mathbf{D}_N) = \frac{n_2 - N_{v2}}{(n_1 - N_{v1}) + (n_2 - N_{v2})} = \frac{(100 - 7)}{(50 - 45) + (100 - 7)} = 0.949$$

We can now compute the entropy of the partitions $\mathbf{D}_Y$ and $\mathbf{D}_N$ as follows:

$$H(\mathbf{D}_Y) = -(0.865 \log_2 0.865 + 0.135 \log_2 0.135) = 0.571$$

$$H(\mathbf{D}_N) = -(0.051 \log_2 0.051 + 0.949 \log_2 0.949) = 0.291$$

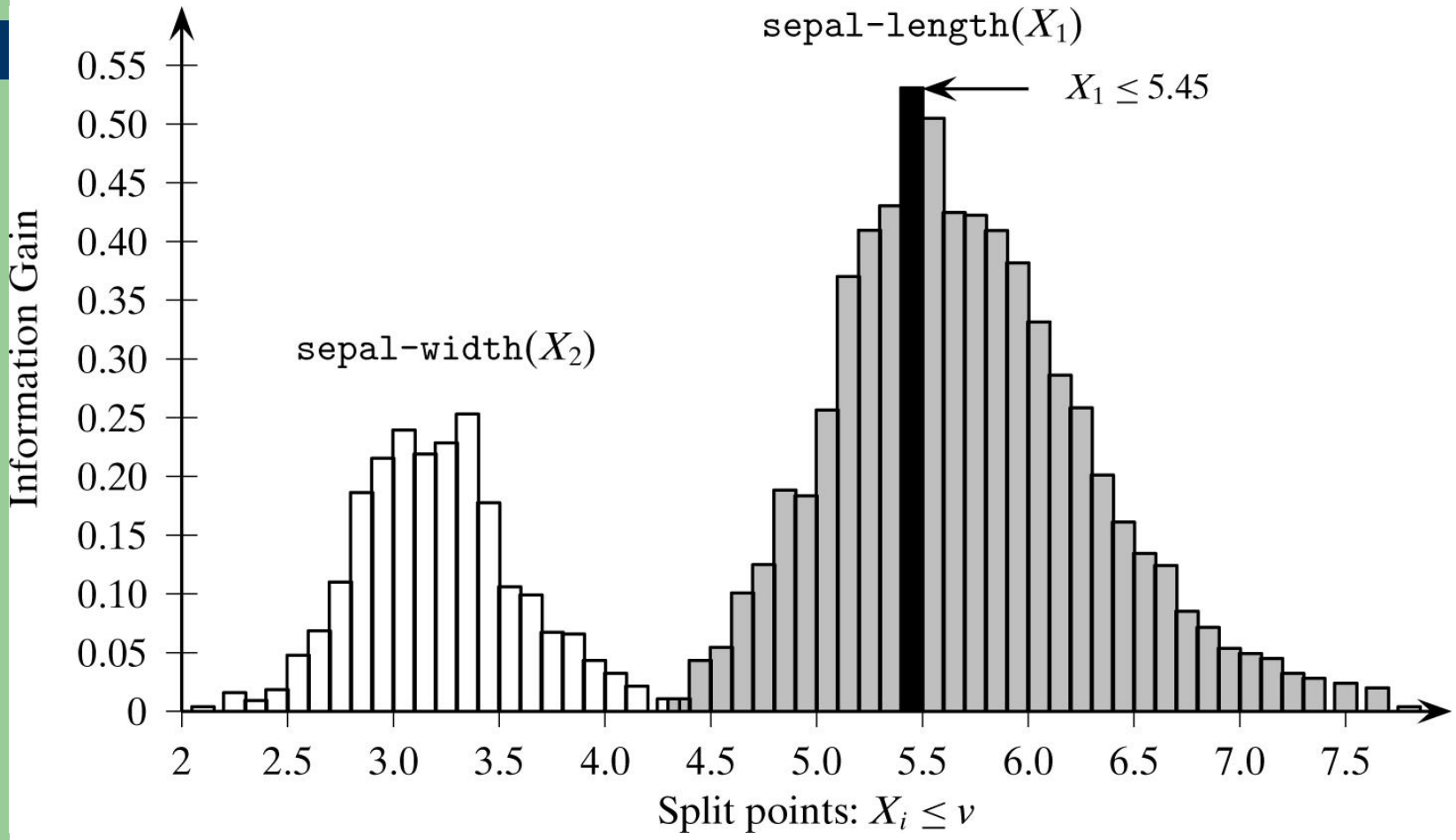
The entropy of the split point $X \leq 5.45$ is given via Eq. (19.4)

$$H(\mathbf{D}_Y, \mathbf{D}_N) = \frac{52}{150} H(\mathbf{D}_Y) + \frac{98}{150} H(\mathbf{D}_N) = 0.388$$

where $n_Y = |\mathbf{D}_Y| = 52$ and $n_N = |\mathbf{D}_N| = 98$. The information gain for the split point is therefore

$$Gain = H(\mathbf{D}) - H(\mathbf{D}_Y, \mathbf{D}_N) = 0.918 - 0.388 = 0.53$$

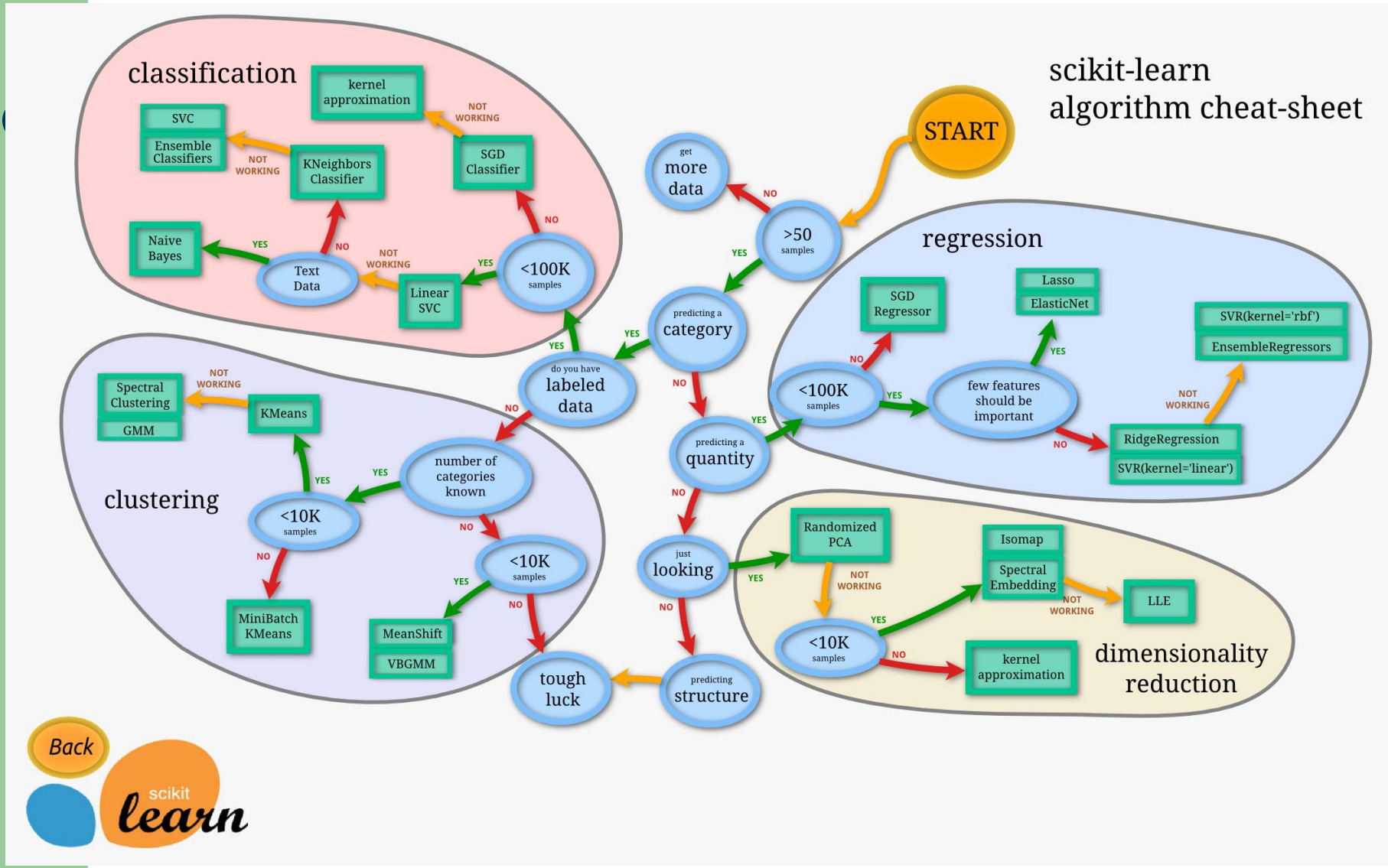
情報利得のバーグラフ


sepal-length(X_1)
$X_1 \leq 5.45$
sepal-width(X_2)
Information Gain
0.55
0.50
0.45
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0
2
2.5
3.0
3.5
4.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5
Split points: $X_i \leq v$

Split Point

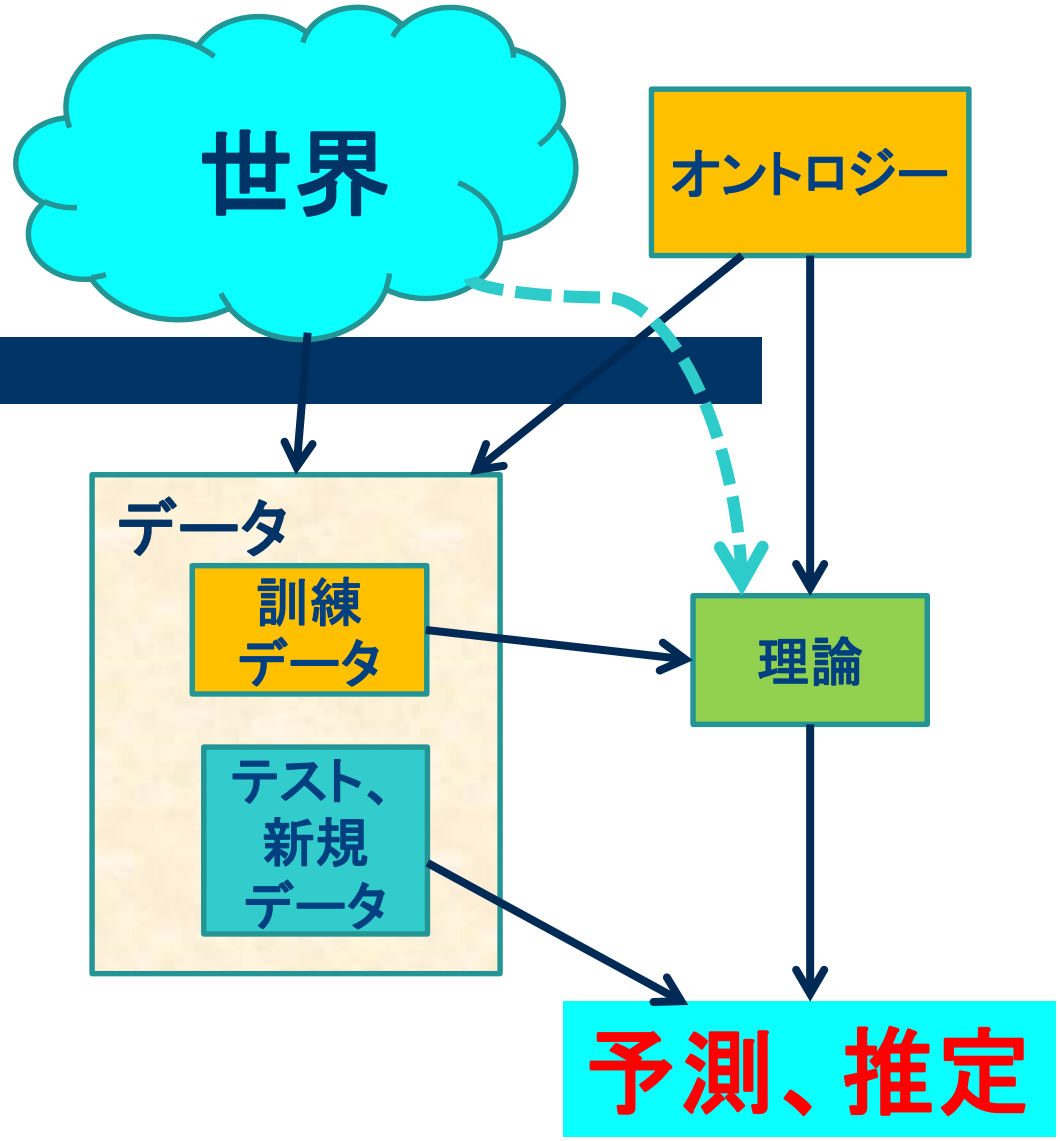
- 情報利得の大きい方から分割点を行うの方がベストである。
- $X_1 \leq 5.45$
- スライド8のような木を得られる

分類器/クラスタ分析器/回帰など



まとめ

- 決定木分類器
- 例題



参考文献

- Data Mining (Mohamed Zaki and Wagner Meira)

Equation 6-1. Gini impurity

$$G_i = 1 - \sum_{k=1}^{n} {p_{i,k}}^2$$

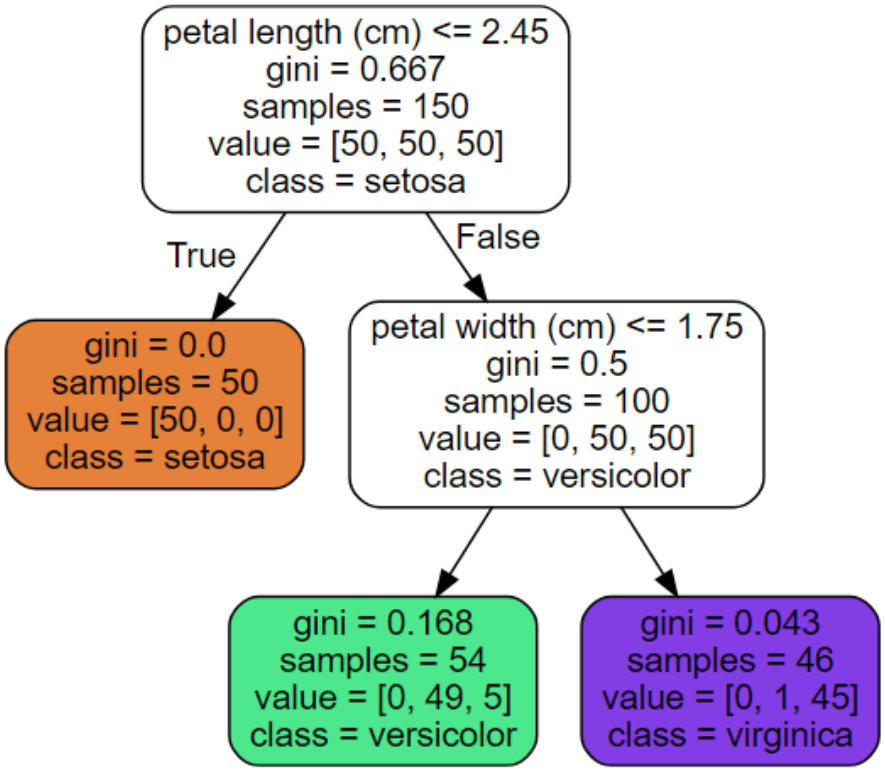
Equation 6-2. CART cost function for classification

$$J(k, t_k) = \frac{m_{\text{left}}}{m} G_{\text{left}} + \frac{m_{\text{right}}}{m} G_{\text{right}}$$

where $\begin{cases} G_{\text{left/right}} \text{ measures the impurity of the left/right subset,} \\ m_{\text{left/right}} \text{ is the number of instances in the left/right subset.} \end{cases}$

Equation 6-3. Entropy

$$H_i = -\sum_{\substack{k=1 \\ p_{i,k} \neq 0}}^{n} p_{i,k} \log_2 \left(p_{i,k}\right)$$


petal length (cm) <= 2.45
gini = 0.667
samples = 150
value = [50, 50, 50]
class = setosa
True
False
gini = 0.0
samples = 50
value = [50, 0, 0]
class = setosa
petal width (cm) <= 1.75
gini = 0.5
samples = 100
value = [0, 50, 50]
class = versicolor
gini = 0.168
samples = 54
value = [0, 49, 5]
class = versicolor
gini = 0.043
samples = 46
value = [0, 1, 45]
class = virginica

Equation 6-4. CART cost function for regression

$$J(k, t_k) = \frac{m_{\text{left}}}{m}\text{MSE}_{\text{left}} + \frac{m_{\text{right}}}{m}\text{MSE}_{\text{right}} \quad \text{where} \begin{cases} \text{MSE}_{\text{node}} = \sum_{i \in \text{node}} \left(\hat{y}_{\text{node}} - y^{(i)}\right)^2 \\ \hat{y}_{\text{node}} = \frac{1}{m_{\text{node}}} \sum_{i \in \text{node}} y^{(i)} \end{cases}$$

Equation 7-1. Weighted error rate of the j^{th} predictor

$$r_j = \frac{\sum_{\substack{i=1 \\ \hat{y}_j^{(i)} \neq y^{(i)}}}^{m} w^{(i)}}{\sum_{i=1}^{m} w^{(i)}}$$ where $\hat{y}_j^{(i)}$ is the j^{th} predictor's prediction for the i^{th} instance.

Equation 7-2. Predictor weight

$$\alpha_j = \eta \log \frac{1 - r_j}{r_j}$$

Equation 7-3. Weight update rule

for $i = 1, 2, \cdots, m$

$$w^{(i)} \leftarrow \begin{cases} w^{(i)} & \text{if } \hat{y}_j^{(i)} = y^{(i)} \\ w^{(i)} \exp\left(\alpha_j\right) & \text{if } \hat{y}_j^{(i)} \neq y^{(i)} \end{cases}$$

Then all the instance weights are normalized (i.e., divided by $\sum_{i=1}^{m} w^{(i)}$).

Equation 7-4. AdaBoost predictions

$$\hat{y}(\mathbf{x}) = \underset{k}{\operatorname{argmax}} \sum_{\substack{j=1 \\ \hat{y}_j(\mathbf{x}) = k}}^{N} \alpha_j$$ where N is the number of predictors.