

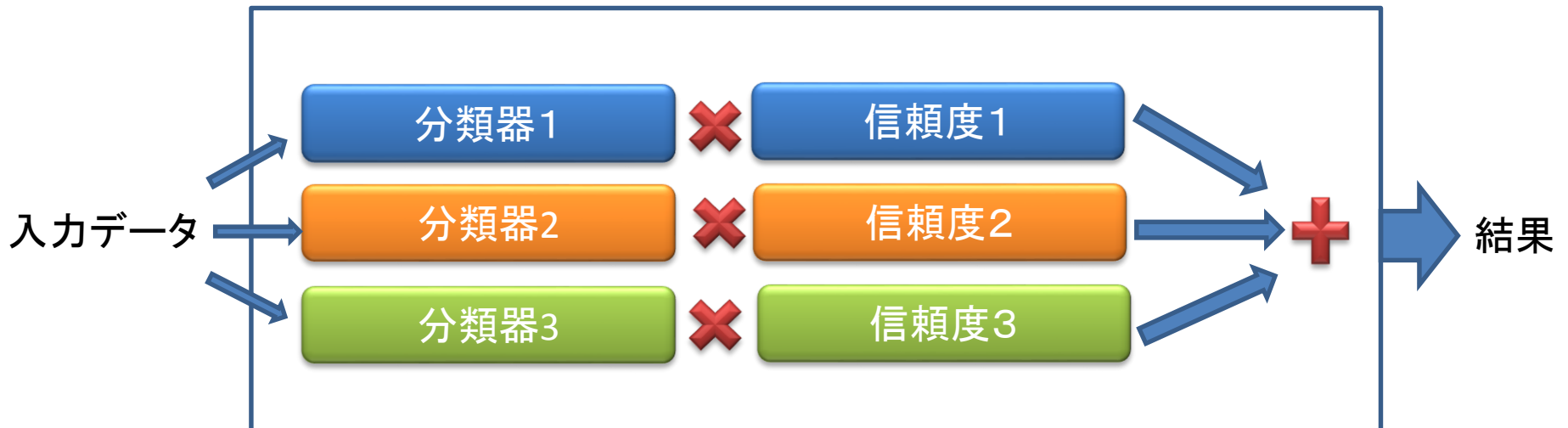
1.9.2. AdaBoost

1.9.2.1. Usage

Shinya Kunii

Adaboostとは

- バギングと並ぶ代表的なアンサンブル学習
- 弱分類器を複数作り、その出力と信頼度の積の和を計算する
- 学習過程でサンプルに重み付けを繰り返して、分類器の精度を上昇させる



アルゴリズム

- サンプルへの重み $D_t(x_i)$ を均一に初期化
- 以下の手続きを $t = 1, 2, \dots, T$ について反復
 - 1. 現在の $D_t(x_i)$ の重み付けを用いて, 弱学習器に分類器 $h_t(x)$ を生成させる.
 - 2. サンプルへの重み $D_t(x_i)$ で重み付けした, データ集合 $\{x_i\}$ に対する誤分類率を ϵ_t とする.

$$\beta_t = \epsilon_t / (1 - \epsilon_t)$$

- 3. サンプルへの重みを更新

$$D_{t+1}(x_i) = \begin{cases} \beta_t D_t(x_i), & y_i \neq h_t(x_i) \\ D_t(x_i), & y_i = h_t(x_i) \end{cases}$$

最後に $D_{t+1}(x_i)$ を正規化.

- 最終分類器は h_1, \dots, h_T がそれぞれ, $\log \frac{1}{\beta_t}$ で重み付けした投票で行う.

Usage

```
>>> from sklearn.cross_validation import cross_val_score
>>> from sklearn.datasets import load_iris
>>> from sklearn.ensemble import AdaBoostClassifier
>>> iris = load_iris()
>>> clf = AdaBoostClassifier(n_estimators=100)
>>> scores = cross_val_score(clf, iris.data, iris.target)
>>> scores.mean() 0.9...
```