

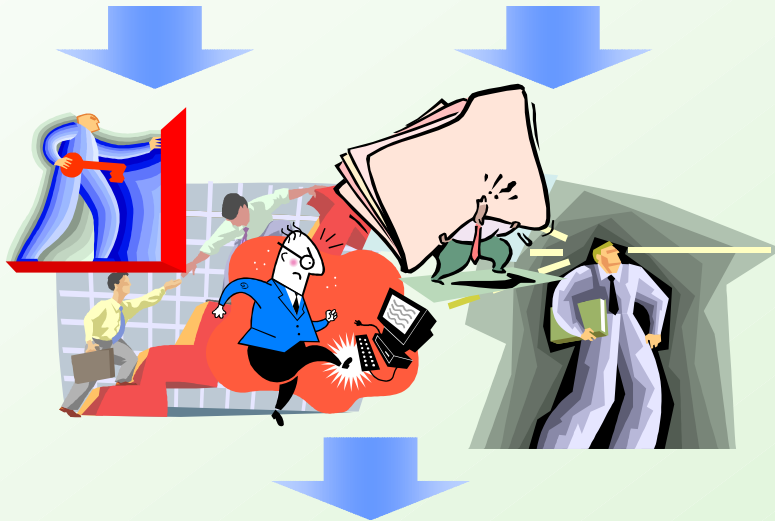
構成型メタ学習と 選択型メタ学習の比較評価

阿部秀尚, 山口高平
静岡大学

背景

データセット

学習アルゴリズム



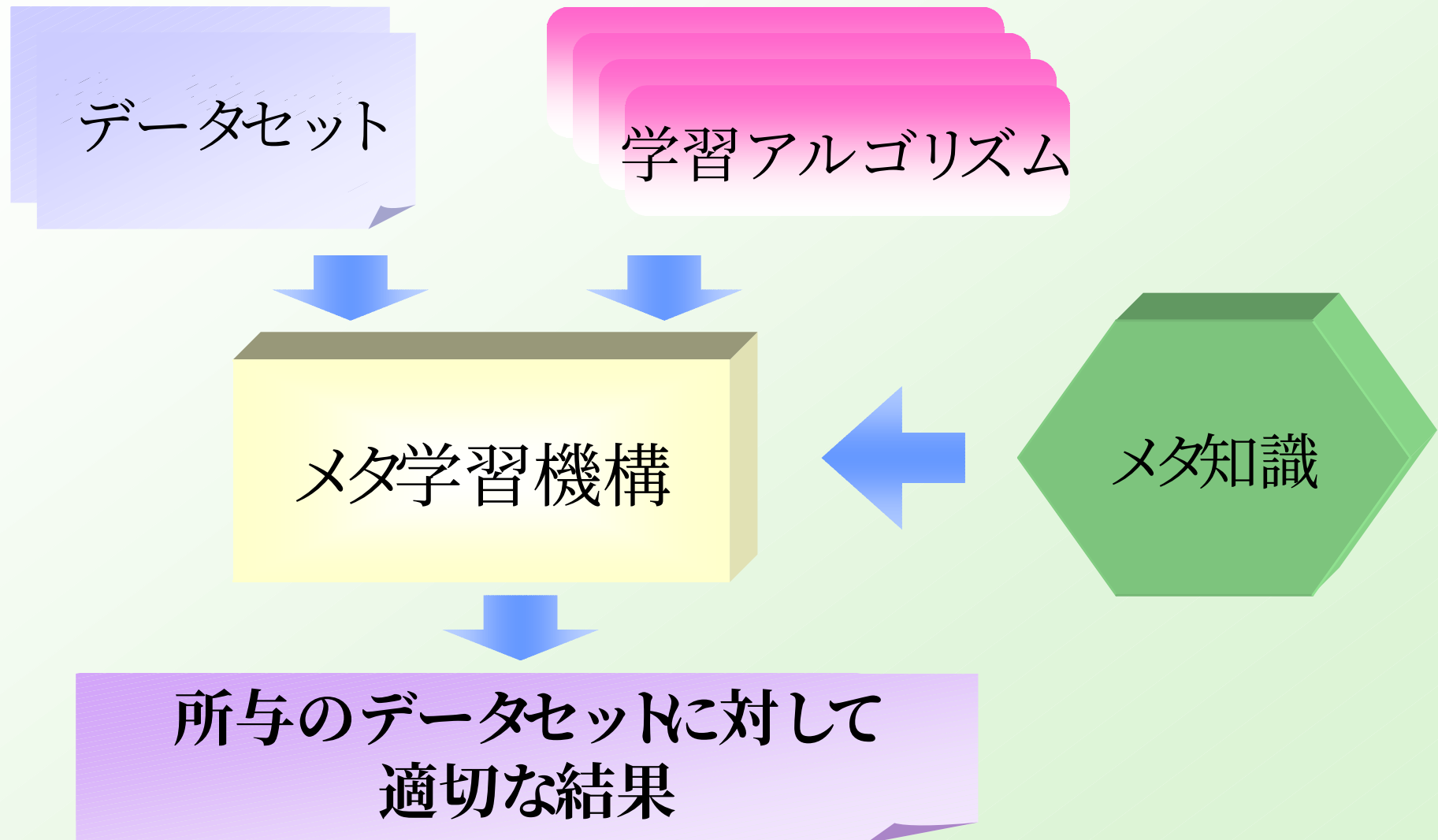
適切な
アルゴリズムの選定



高度な専門知識
試行錯誤
が必要！！
= コスト高

所与のデータセットに対して
適切な結果

背景：メタ学習

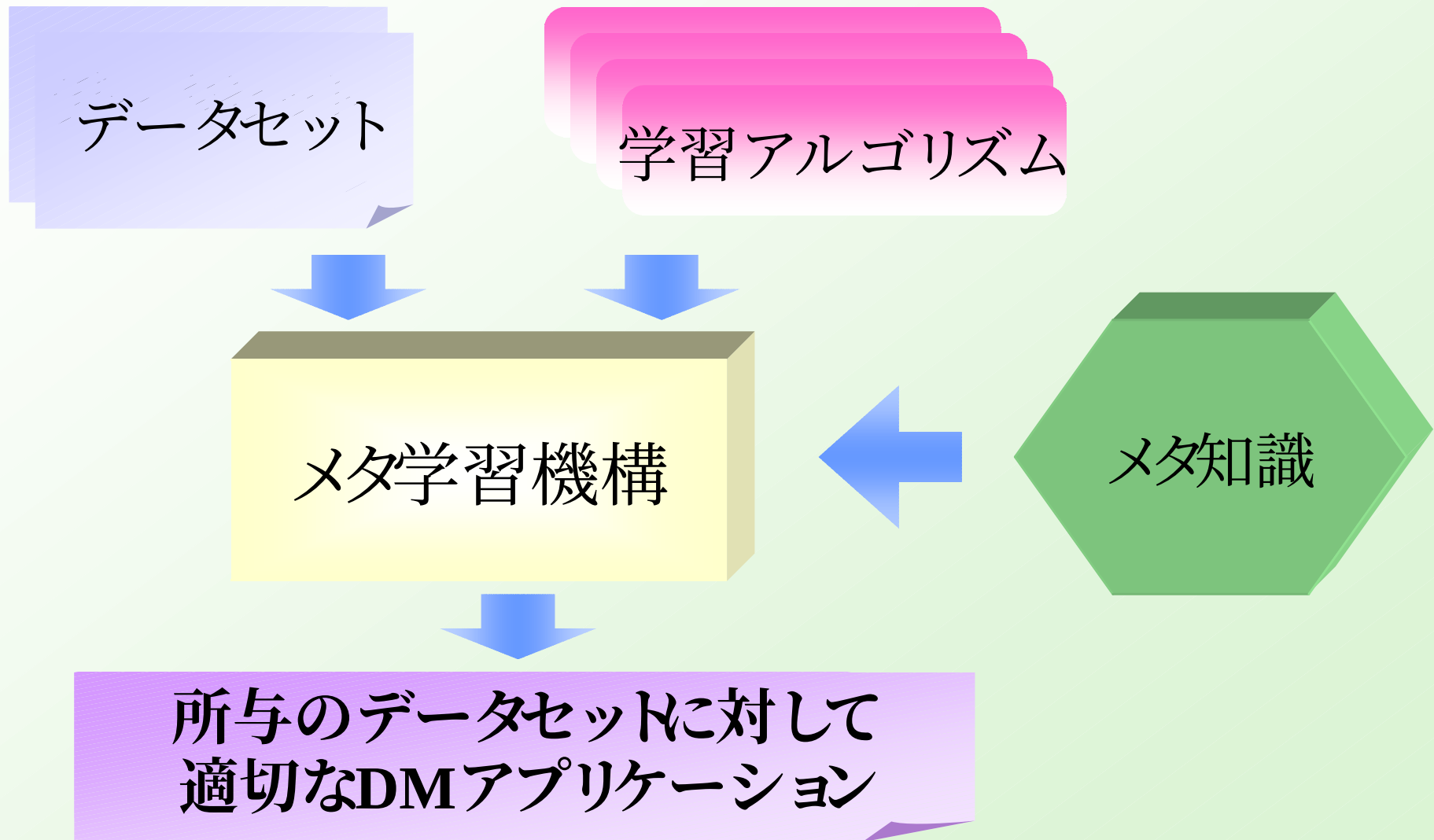


従来 選択型)のメタ学習

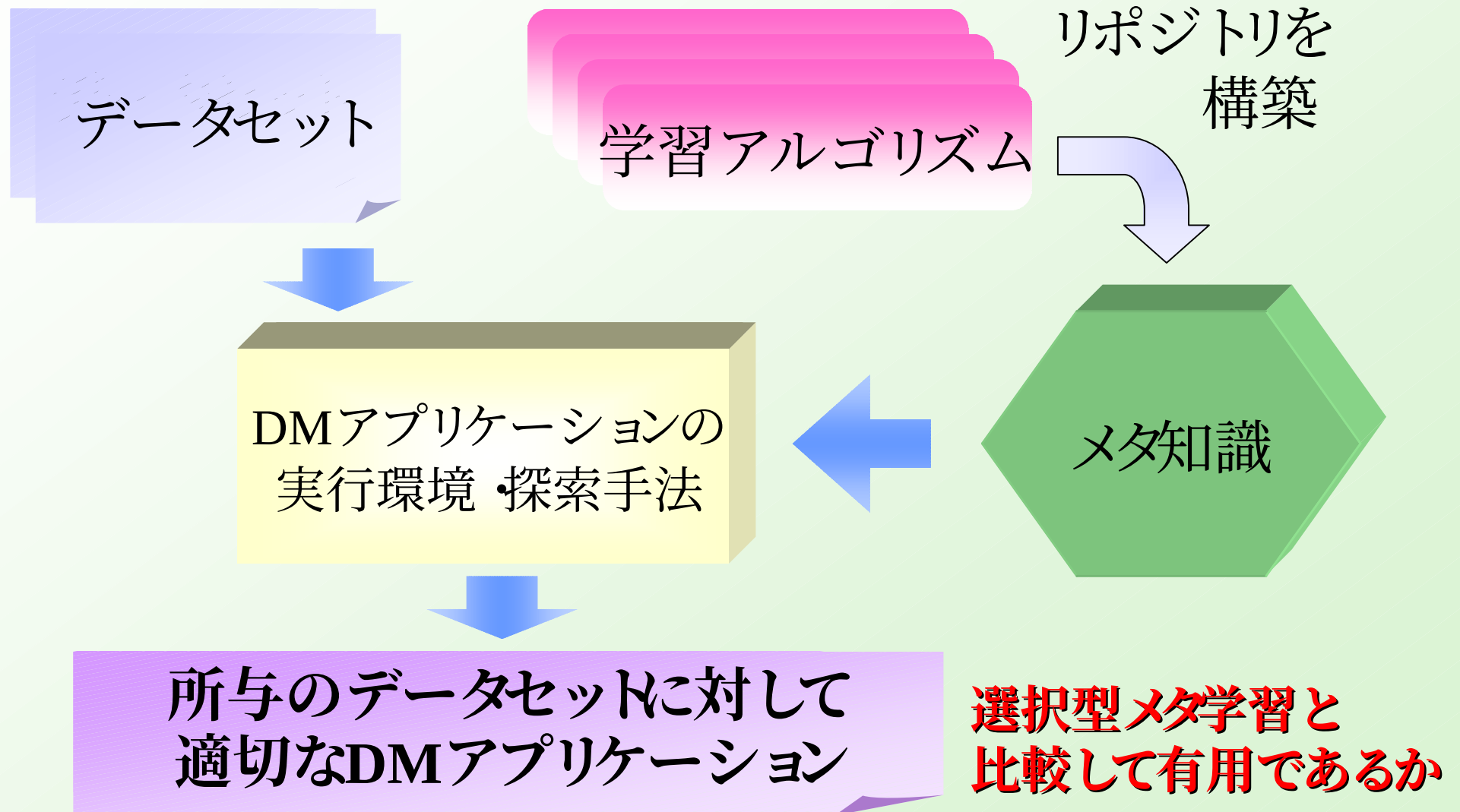
- データの分布を変えた結果を統合
 - ブートストラップサンプルによる分類器を統合(bagging)
 - 誤分類事例の重みを増やして学習した分類器を統合(boosting)
- 異なるアルゴリズムの結果を統合
 - 投票による統合(voting)
 - ベースレベル分類器の結果からメタレベル分類器を学習する(stack, cascading)

既存の学習アルゴリズムの性能が上がらない場合、満足する結果が得られないことがある

構成型メタ学習



構成型メタ学習



構成型メタ学習 :CAMLET

CAMLET=a Computer Aided Machine Learning Engineering Tool

•データセット (+ 検定法)

— 訓練データセット

— テストデータセット

•目標正解率

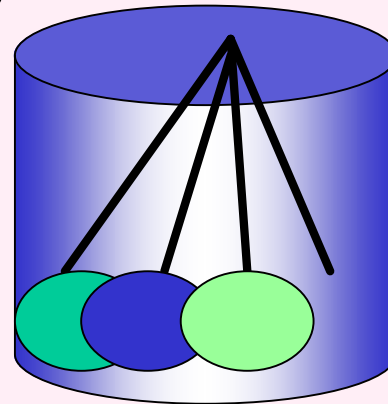
利用者



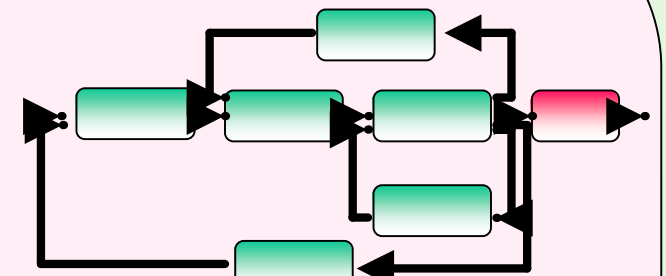
•正解率

•帰納(DM)アプリケーション

CAMLET

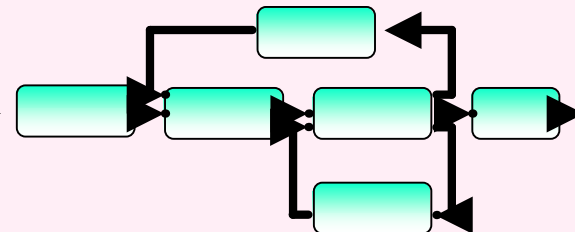


メソッドリポジトリ
データ構造階層



制御構造テンプレート

仕様空間を探索し、
最適な帰納アプリケーションを合成



学習アルゴリズムの分析

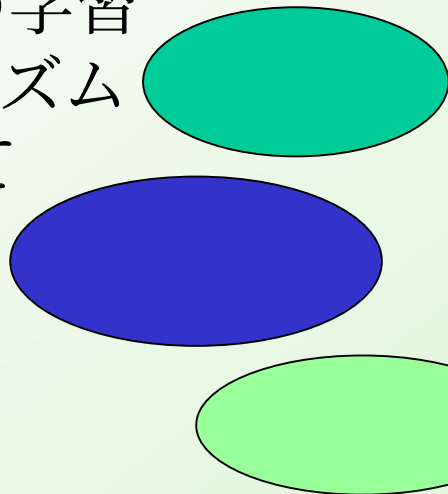
以下の 8種類の帰納学習アルゴリズムを分析

- バージョン空間法 候補削除アルゴリズム)
- AQ15 (スターアルゴリズム)
- ID3 決定木＋訓練データ更新)
- C4.5 決定木＋訓練データ更新＋枝刈り)
- Boosted C4.5 決定木＋訓練データ更新＋Voting)
- Bagged C4.5 決定木＋分類器統合)
- ニューラルネットワーク
- Classifier System(分類器評価＋分類器更新)

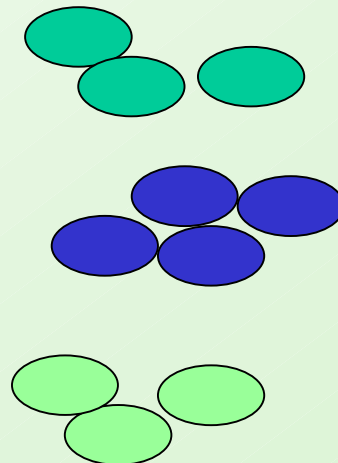
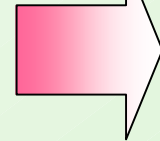
メソッドリポジトリの構築

- 機能や入出力のデータ構造 (データセット・分類器集合)の違いを考慮して, 機能部分であるメソッドを切り出す
- トップレベルのメソッドと制御構造を同定
- 各メソッドを入出力/参照データ構造・機能により階層化
- メソッドリポジトリの葉ノードは実装と対応する

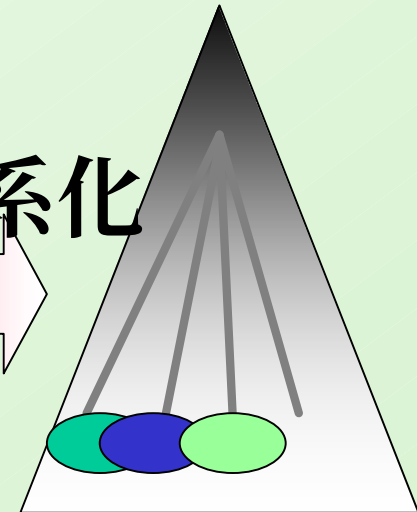
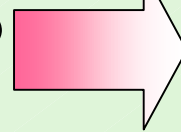
8種類の学習
アルゴリズム
について



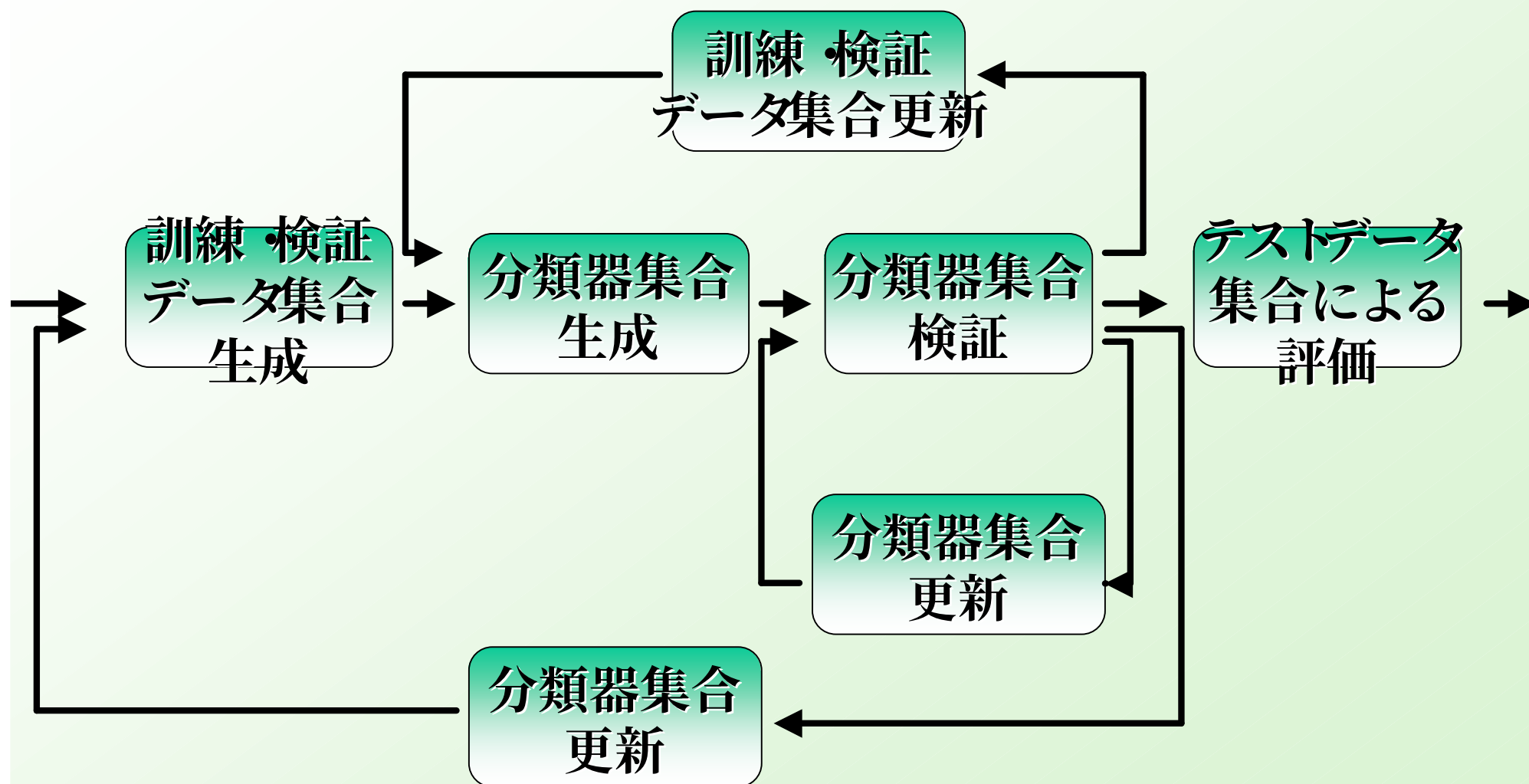
分析



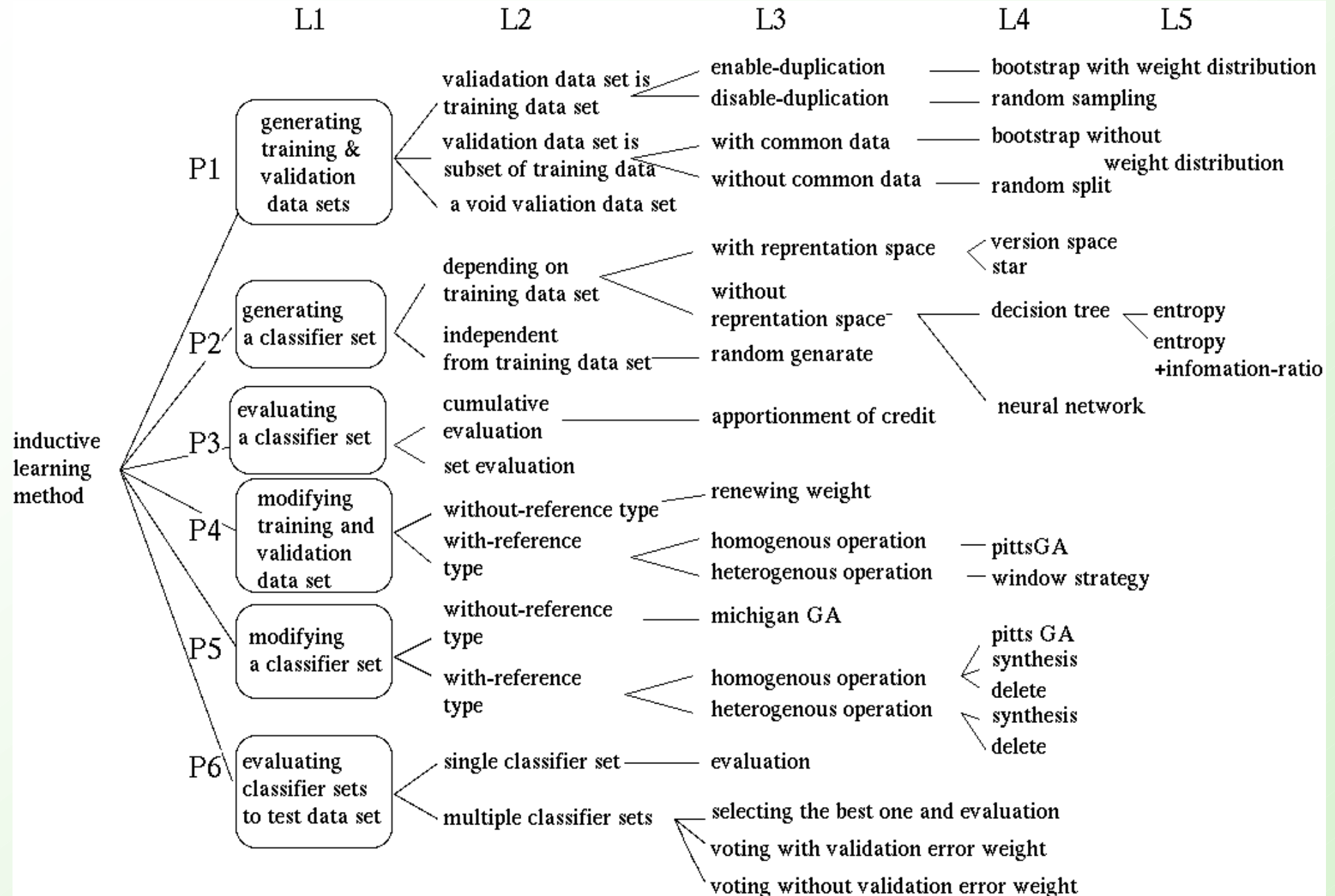
体系化



メソッドと制御構造

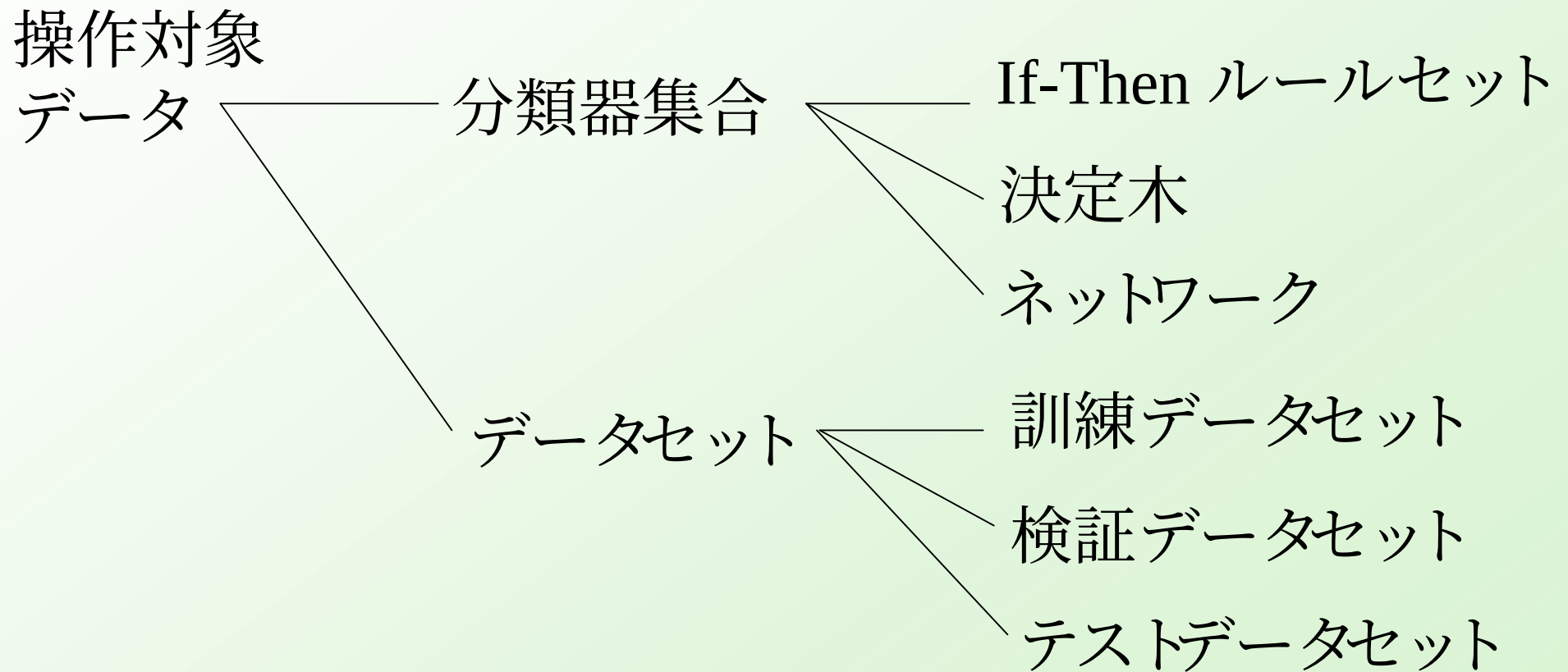


メソッドリポジトリ

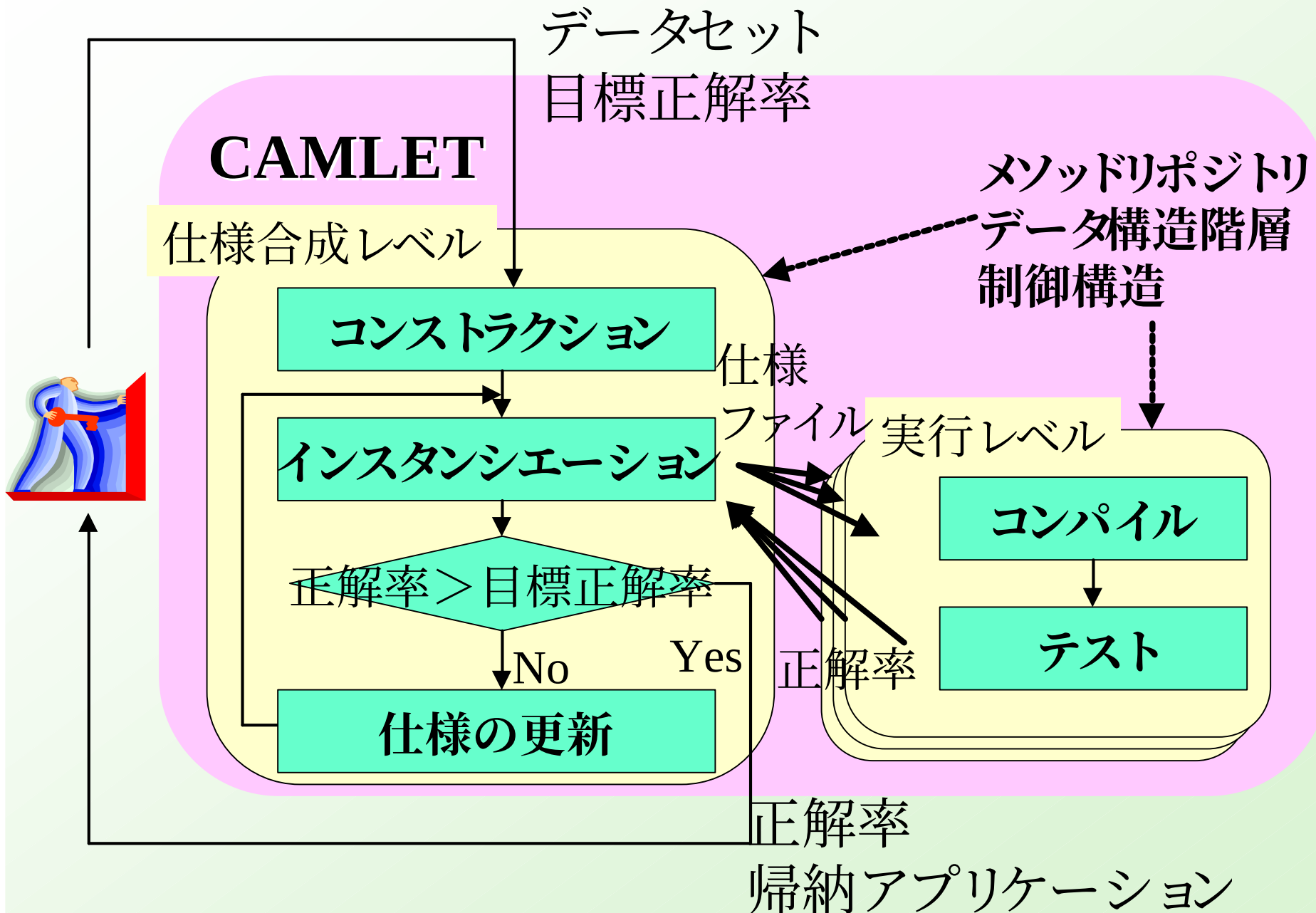


データ構造階層

帰納学習メソッドで操作対象となるデータ構造を階層化



CAMLETの基本動作

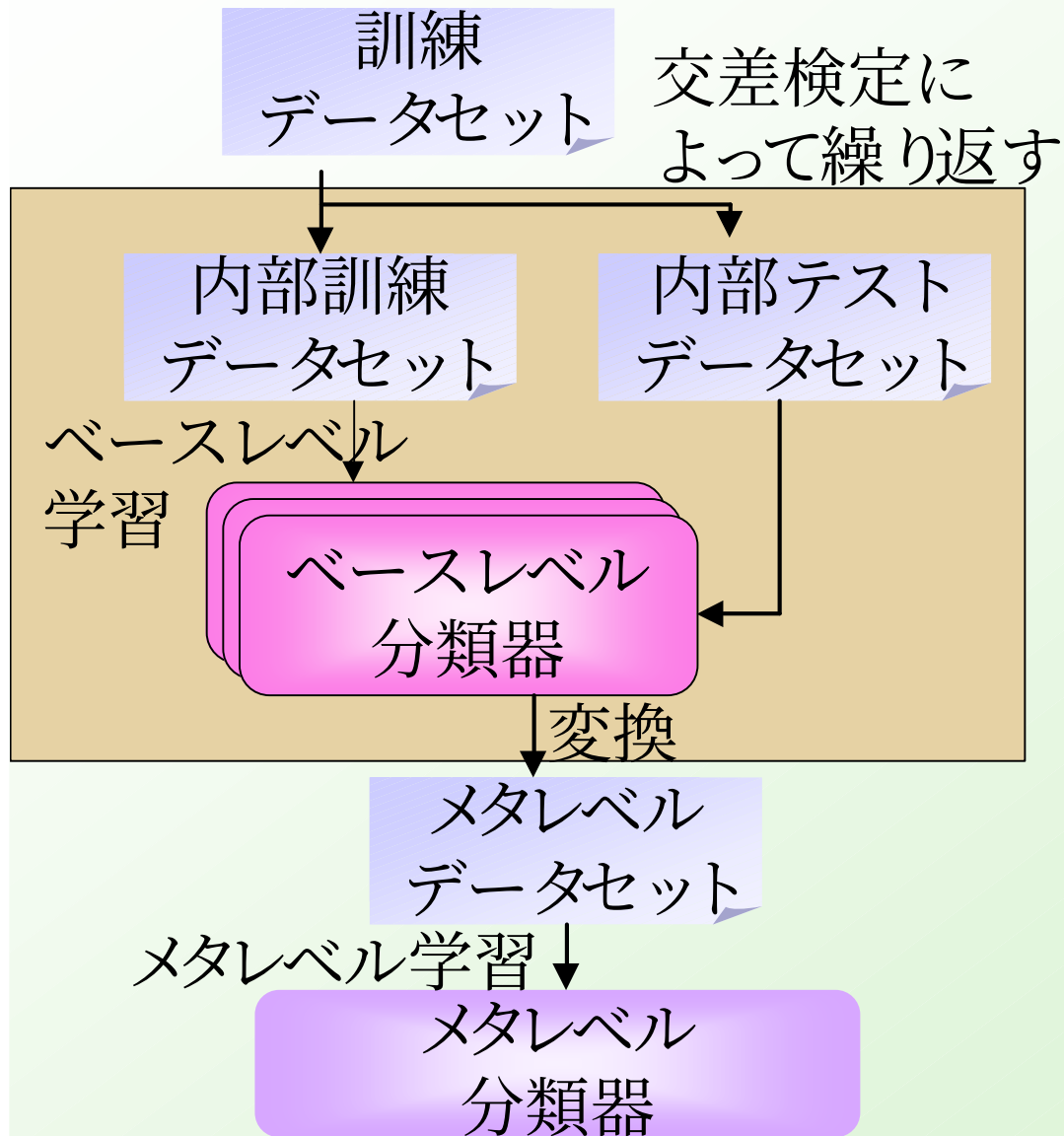


構成型メタ学習と選択型メタ学習の 比較実験

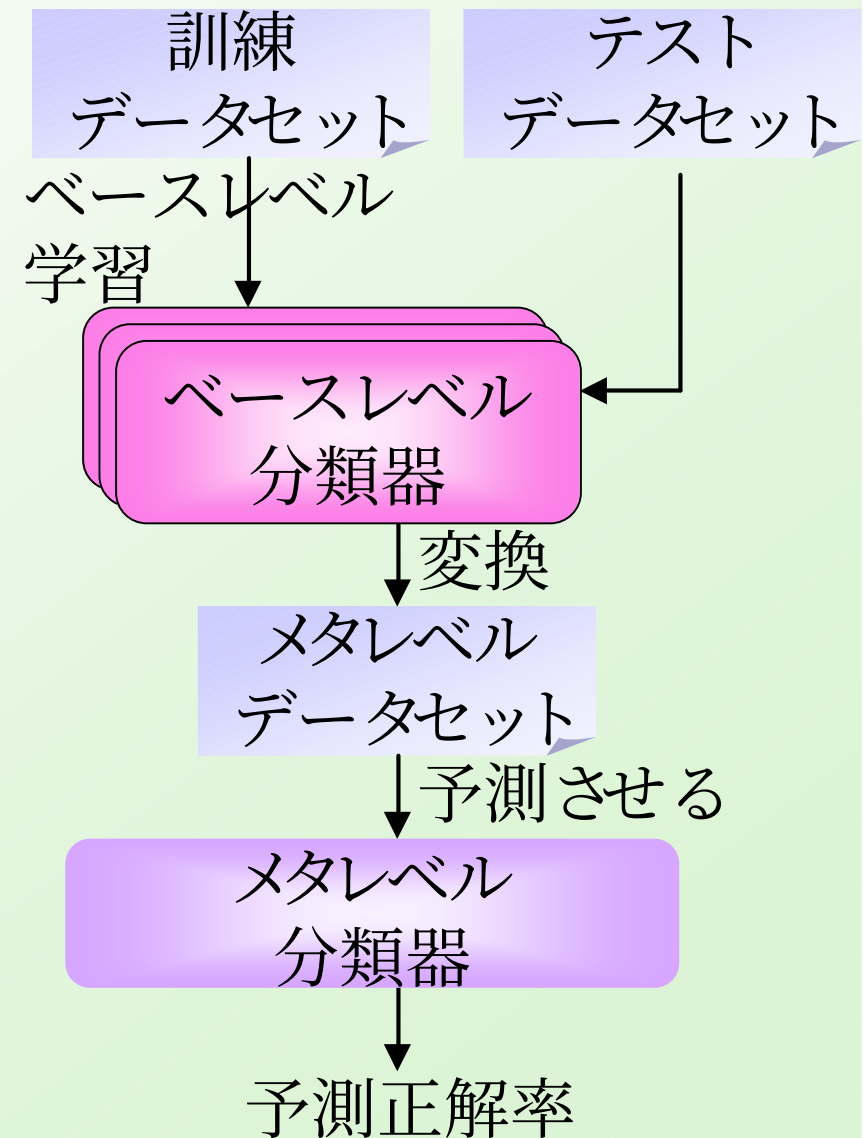
- 構成型メタ学習の代表としてCAMLETを用いる
- 選択型メタ学習の代表としてstackingを用いる
 - StackingはWekaに実装されたものを利用
- データセットはStatlogによる共通データセットを利用
- 比較は予測正解率 (テストデータセットに対する正解率) に対して行う

選択型メタ学習 : Stacking

学習過程



予測過程



メタデータセットへの変換 (stacking)

学習アルゴリズム 1によって分類器を得る

アルゴリズム 1
分類器

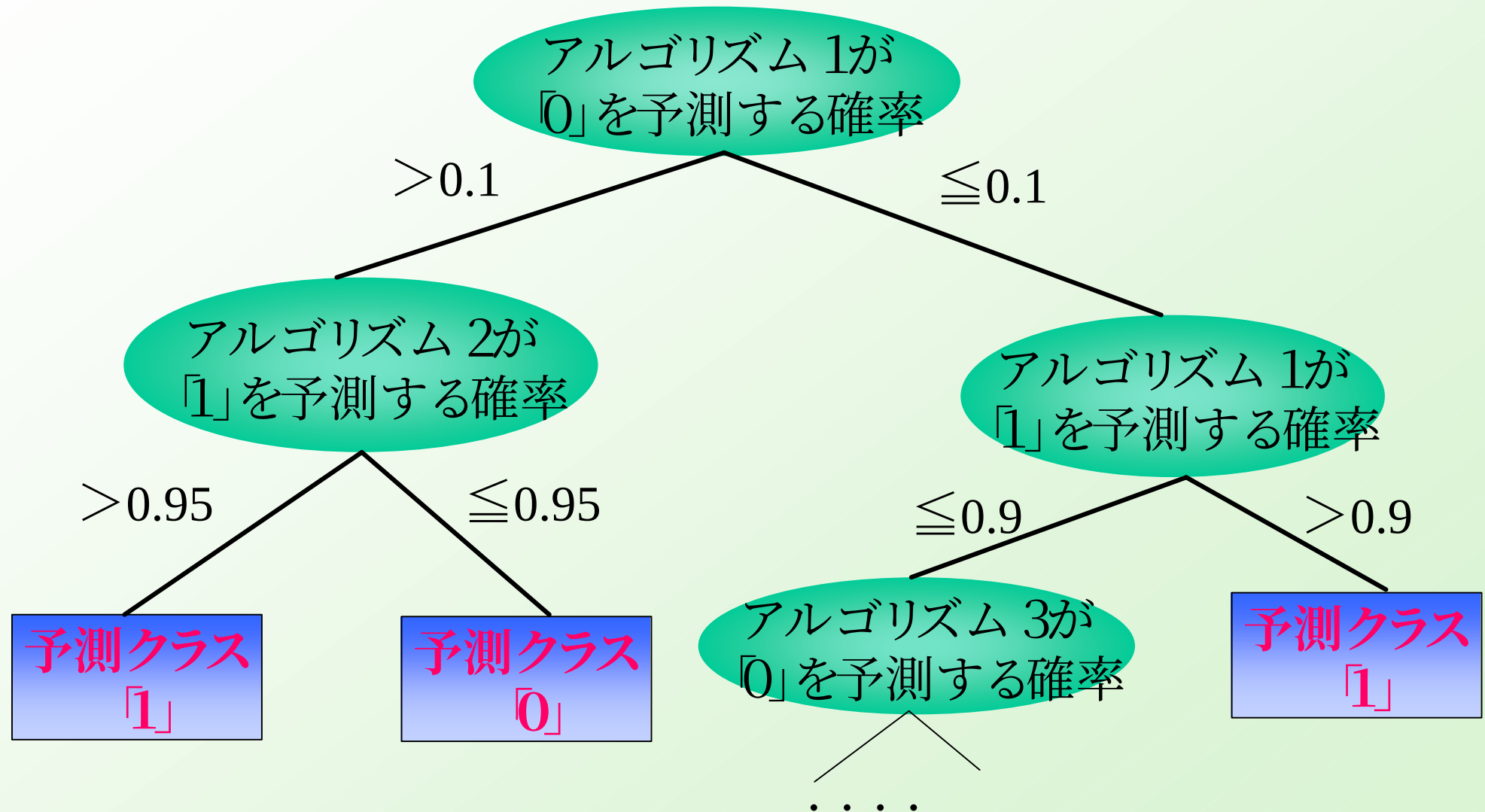
属性A	属性B	属性C	クラス
a1	0.3	c3	0
a2	5	c2	1
...	...	...	...

1. 各インスタンスに対して, 各分類器がクラス値を予測する確率 (数値属性)に変換する.
2. 変換されたインスタンスに元のクラスを付与する.

アルゴリズム 1		クラス
「0」を予測する確率	「1」を予測する確率	
0.99	0.01	0
0.4	0.6	1
...	...	...

アルゴリズム :A個
クラス :C個
メタ属性数 = $A \times C$ 個

メタレベル分類器 (分類モデル)



Stackingの設定

～ベースレベル学習アルゴリズム～

- J4.8 (枝刈りあり, 危険率0.25)
- IBk (5-nearest neighbours)
- Partルール学習器
- Naïve Bayes
- Bagged J4.8 (サンプル率55%, J4.8で5回学習)
- Bagged J4.8 (サンプル率55%, J4.8で10回学習)
- Boosted J4.8 (J4.8で5回まで学習)
- Boosted J4.8 (J4.8で10回まで学習)

Stackingの設定

～メタレベル学習アルゴリズム～

- J4.8 (枝刈りあり, 危険率0.25) :”J4.8”
- Naïve Bayes :”NB”
- 線形回帰による分類 (属性選択なし) :”LR”

→ ベースレベル学習アルゴリズム (8個)に対し,
それぞれのメタレベル学習アルゴリズムを適用

CAMLETへの入力と探索停止条件

- 要求する正解率は線形回帰による分類をメタレベル学習アルゴリズムとしたStackingの正解率
- 要求した正解率を満たした時点で探索を打ち切る
- 各データセットに対して, 100回で帰納アプリケーションの探索を打ち切る

Statlog共通データセット

Dataset	クラス数	属性数 (名義 :数値)	インスタンス数	評価法
australian	2	14 (6:8)	690	10CV
diabetes	2	8 (0:8)	768	10CV
dna	3	180 (180:0)	2,000/1,186	Test
german	2	20 (17:3)	1,000	10CV
heart	2	13 (6:7)	270	10CV
letter-liacc	26	16 (0:16)	16,000/4,000	Test
satimage	6	36 (0:36)	4,435/2,000	Test
segment	7	18 (0:18)	2,317	10CV
shuttle	7	9 (0:9)	43,500/14,500	Test
vehicle	4	18 (0:18)	846	10CV

10回交差検定のデータセットはWekaの
SplitDatasetFilterを使用して作成

ベースレベル学習アルゴリズムと 分析対象学習アルゴリズムの比較

Stacking

ベースレベル学習アルゴリズム

Algorithms	Avg.(10 data sets)
J4.8	84.84
IBk(5)	84.32
Part	84.10
NaiveBayes	76.53
Bagging(5)	85.29
Bagging(10)	86.17
Boosting(5)	85.56
Boosting(10)	85.94
Max	87.58

正解率 (%)

CAMLET

分析対象学習アルゴリズム

Algorithms	Avg.(10 data sets)
C4.5 DT	81.07
ID3 DT	81.76
NeuralNetwork	63.09
ClassifierSystems	64.85
Bagging(5)	84.28
Bagging(10)	85.17
Boosting(5)	84.52
Boosting(10)	85.59
Max	87.04

正解率 (%)

精度評価

データセット	Stacking			CAMLET
	J4.8	NB	LR	
australian	85.07	85.94	87.1	87.1
diabetes	73.31	74.86	76.17	76.18
dna	93.92	95.28	94.94	95.19
german	72.6	73.1	74.4	74.4
heart	80.74	82.22	82.96	83.3
letter-liacc	94.96	93.58	96.6	94.6
satimage	89.3	88.3	91.05	90.55
segment	97.84	97.62	98.14	97.66
shuttle	99.96	99.94	99.99	99.99
vehicle	71.98	75.17	74.69	75.18
Average	85.968	86.601	87.604	87.415

線形回帰による分類を利用したStackingとCAMLETの差は無い

→ CAMLETにはメタ学習として有用な性能がある

まとめ

- 選択型メタ学習(Stacking)と構成型メタ学習(CAMLET)を比較評価
- CAMLETがメタ学習として有用であることを確認
- stackingの3つの課題についてCAMLETの適用を考慮すると
 - CAMLETによるベースレベル／メタレベル学習アルゴリズムの選定
 - データの前処理を含めたデータマイニングプロセス全体の支援
- その他の評価基準 計算コスト, 理解容易性など)への対応