S^3Baggingによる高速な分類器生成

寺邊正大[†] 鷲尾隆[††] 元田浩[††]

データマイニングの過程では，意思決定者がデータから必要とする知識を得ることができるまでに，分類器の生成を繰り返さなければならないことが多い．よって，データマイニングツールには大規模データから正確な知識を抽出するだけでなく，意思決定者の要求に応えるべく高速に抽出することも要請される．高速に分類器を生成する方法としてはサブサンプリングを行うことにより学習データの事例数を減らすことが考えられる．しかしながら，一般的には学習データの事例数が減少すると分類器の分類精度が劣化する．本論文では，サブサンプリングとコミッティ学習手法の１種である Bagging を組み合せた新しい分類器生成手法である S^3Bagging (**S**mall **S**ub**S**ampled **Bagging**) を提案する．S^3Bagging では，複数の分類器をサブサンプリングされた学習データを用いて並列に生成し，生成された分類器を用いてコミッティ分類器を構成する．これにより，サブサンプリングによる最終的な分類精度の劣化を防ぐことができる．また，S^3Bagging の性能について実験を通じて確認した結果を合わせて示す．

Fast Classifier Generation by S^3Bagging

Masahiro Terabe,[†] Takashi Washio[††] and Hiroshi Motoda[††]

In the data mining process, it is often necessary to induce classifiers iteratively until the human analysts complete extracting valuable knowledge from data. Therefore, the data mining tools have to extract accurate knowledge from a large amount of data quickly enough in response to the human demand. One of the approaches to answer this request is to reduce the training data size by subsampling. In many cases, the accuracy of the induced classifiers becomes worse when the training data is subsampled. We propose S^3Bagging (**S**mall **S**ub**S**ampled **Bagging**) that adopts both subsampling and a method of committee learning, i.e., Bagging. S^3Bagging can induce classifiers efficiently by reducing the training data size by subsampling and parallel processing. Additionally, the accuracy of the classifiers is maintained by aggregating the result of each classifier through the Bagging process. The performance of S^3Bagging is investigated by carefully designed experiments.

1. はじめに

近年，大規模データから有用な知識を発掘するデータマイニングの研究が盛んである [9]．データマイニングは，金融や流通・小売などビジネス分野から，製造，通信などの産業分野に至るまで広範な分野で現実問題に適用されている [20]．

意思決定者がデータマイニングツールを用いて知識抽出を行う過程では，必要とする知識の抽出が完了するまでに，データからの分類器生成を繰り返さなければならない場合が多い [1]．このため，意思決定者の利便性を考えると，データマイニングツールには高い精度の知識を抽出するだけでなく，高速に知識を抽出し意思決定者に提示することが要請される．

分類器の分類精度を向上させるための研究については，機械学習の分野でコミッティ学習 (committee learning) に関する研究が盛んである [23]．一般的な分類器生成手法では，元学習データの全ての事例を使って学習を行うことにより，１つの分類器を生成する．これに対してコミッティ学習の枠組みでは，以下の手順によりコミッティ分類器とよばれる分類器を生成する．まず準備された元学習データからサンプリングすることにより，事例構成を多少異ならせた複数の学習データを生成する．次に，各学習データを用いて学習を行い，学習データごとに１つのメンバー分類器を生成する．コミッティ学習により生成されるコミッティ分類器とは，これらメンバー分類器の集まりとして構成されるものである．コミッティ分類器ではメンバー分類器が自らの分類結果にもとづいて投票を行い，最も多くの投票を獲得したクラスをコミッティ分類器の

† （株）三菱総合研究所
Mitsubishi Research Institute, Inc.
†† 大阪大学 産業科学研究所
I.S.I.R, Osaka University

分類結果とする．

コミッティ学習手法の主なものに，Boosting[8),19)] と Bagging[2)] がある．Boosting は，次のような手順で系列的にメンバー分類器を生成し，これをもとにコミッティ分類器を構成する．まず，元学習データの事例ごとに付与されたサンプリング確率にもとづいて重み付きサンプリングを行い，学習データを 1 つ準備する．次に，この学習データから分類器を生成する．そして，生成した分類器による学習データの分類結果をもとに，分類を誤った事例がよりサンプリングされやすくなるようにサンプリング確率を更新する．ここで，収束条件が充たされた場合には分類器の生成を終了し，それまでに生成された分類器をメンバー分類器とするコミッティ分類器を構成する．一方，収束条件を充たさない場合には，さらに分類器の生成を繰り返す．Boosting と一般的な分類器生成手法を比較すると，Boosting により生成された分類器の方が分類精度が優れることが報告されている．一方，Boosting はメンバー分類器を系列的に生成することが必要であるために，一般的な分類器生成手法に比べて多くの処理時間を要する．

一方，Bagging では，まず元学習データに対して復元抽出法によるサンプリングを行い，複数の学習データを準備する．そして各学習データからメンバー分類器を生成し，これらをメンバー分類器としたコミッティ分類器を構成するものである．ここで，サンプリングからメンバー分類器を生成する過程は，独立して実行が可能である．Bagging は，Boosting に比べると生成されるコミッティ分類器の分類精度の面で劣ることが多い．しかしながら，データを問わず安定して一般的な分類器生成手法よりも分類精度の良い分類器を生成することができる．さらには Boosting と異なり，コミッティ分類器を構成するメンバー分類器を並列に生成することができる．このため，並列処理を行えば一般的な分類器生成手法と処理時間が変わらないという特長をもつ．

データマイニングにおいて大規模データから分類器を高速に生成する方法としては，大規模データを高速に処理することができる学習アルゴリズムを開発することの他に，学習データに対してサブサンプリングを行うことにより学習データの事例数を小さくすることが有効である．ここでサブサンプリングとは，元の学習データの事例数よりも少ない事例数を抽出（サンプリング）してデータを生成することを言う．

サブサンプリングによる分類器の分類精度への影響については，Catlett がサンプリング手法としてランダムサンプリングや層別サンプリングを適用した場合について実験を通じた研究を行っている [6)]．この実験では，学習アルゴリズムに代表的な決定木学習アルゴリズムである C4.5[16)] を用いている．C4.5 が数値属性の多い大規模データを扱った場合に処理速度が遅くなることに注目し，サブサンプリングを行うことにより処理時間が大幅に削減できることを確認している．一方，現在では並列計算が可能な計算機環境が身近なものになっており [12)]，並列処理を導入することにより，さらに高速に精度の高い分類器を得ることができる手法の開発が可能となっている [14)]．

本論文では並列処理が可能な環境において，大規模データから高速に，かつ分類精度の高い分類器を生成することができる手法として，サブサンプリングと Bagging を併用する S^3Bagging(**S**mall **S**ub**S**ampled **Bagging**) について提案し，性能評価を行う．S^3Bagging は，サブサンプリングを行い学習データの事例数を減らすことにより，分類器の生成に要する時間を学習データの全事例を用いる場合に比べて大幅に短縮する．また，サブサンプリングを行うことによる分類精度の低下を，Bagging と同様の方法でコミッティ分類器を構成することにより補うことを狙っている．また，Bagging のメンバー分類器の生成を並列に行えるという特長を活かして，メンバー分類器の生成を並列に行うことにより分類器生成を高速化している．

本論文は，以下のように構成される．まず，2 章では提案手法に関連する Bagging とサンプリングについて説明する．次に，3 章で S^3Bagging について提案する．さらに，4 章と 5 章で S^3Bagging の性能を評価するために実験を行った結果を示し，この結果をもとに 6 章で考察を行う．

2. Bagging とサンプリング

2.1 Bagging

Bagging とは，Breiman[2)] により提案されたコミッティ学習手法の 1 つであり，図 1 に示すようにメンバー分類器の生成を並列分散的に行なうことができる．まず，準備されている元学習データ D から復元抽出法 (sampling with replacement) によりサンプリングを行い，元学習データと同じ事例数 $|D|$ をもつ複数の学習データ D_t $(t = 1, \ldots, T)$ を生成する．ここで T は，メンバー分類器数である．復元抽出法とは，既にサンプリングされた事例についても，再びサンプリングの対象とする方法である．よって，復元抽出法によるサンプリングでは，同じ事例が複数回サンプリ

```
Bagging(D, T)
    INPUT:
        D: 元学習データ,
        T: メンバー分類器数.
    OUTPUT:
        C: コミッティ分類器.
        {
            for each t from 1 to T
            {
                /* 復元抽出法により |D| 個の
                事例をサンプリングした学習
                データ D_t を作成する */
                D_t := Samping(D);
                /* メンバー分類器 C_t を生成
                する */
                C_t := Learning(D_t);
            }
            /* 生成された T 個の分類器により
            コミッティ分類器 C を構成する */
            C := {C_t : t = 1, ..., T};
        }
        return C;
```

図 1 Bagging のアルゴリズム
Fig. 1 Algorithm for Bagging.

ングされることがある．このため，高い確率で生成された学習データ間で事例構成が異なる．次に，これらの各学習データを用いて同種の学習アルゴリズムによりメンバー分類器 C_t $(t = 1, \ldots, T)$ を生成し，これらの集合としてコミッティ分類器を構成する．そして，新規の事例 x を分類する場合には，各メンバー分類器による分類結果 $C_t(x)$ について多数決をとってコミッティ分類器の分類結果とする．

ここでサンプリングから各メンバー分類器の生成までの過程は，それぞれ独立に行えるために並列処理が可能である．よって並列処理が可能な計算機環境下では，Boosting など系列的に学習データのサンプリングと分類器の生成を行わなければいけない手法に比べて処理時間の面で優れる．

Bagging は他のコミッティ学習手法と同様に，一般的な分類器生成手法よりも高い精度の分類器を得られることが知られている．また，学習器に代表的な決定木アルゴリズムである CART[2] や C4.5 を用いた場合 [17]，さらにはニューラルネットワークを用いた場合 [13] についても，Bagging により分類精度が改善されることが実験を通じて明らかにされている．

2.2 サンプリング

データマイニングの分野では，以下の内容を目的としたサンプリングの研究が行われてきた．

- **分類精度の向上:** 学習（アルゴリズム）に適した事例を選択的にサンプリングする．学習に適した事例のみを使うことにより，生成される分類器の精度を高めることができる．事例選択手法に関する研究の多くは，主に分類器の精度の向上を狙ったものである．
- **学習時間の短縮:** サブサンプリングにより学習データの事例数を減らす．学習データの事例数が減ることにより，分類器の生成に要する時間を短縮することができる．

前者の目的で用いられるサンプリング手法は，事例の特徴を分類するために学習を行なうものなど，処理に時間を要するものが多い [18]．一方，後者については，統計解析の分野で用いられるランダムサンプリングや層別サンプリング [11] などが用いられる．ランダムサンプリングとは，元学習データからランダムにサンプリングする事例を選択するものである．層別サンプリングとは，元学習データの事例をクラスなどの特徴にもとづいて部分データ（階層）に分割しておき，それぞれの部分データからサンプリングを行うものである．これらの手法は，多くの処理時間を要さない．

Bagging で用いられているサンプリングは，復元抽出を行ないながらランダムサンプリングを行なっている．また，Boosting で用いられている重み付きサンプリングとは，各事例に与えられたサンプリング確率にもとづいて，抽出する事例を選択するものである．これらのサンプリング手法は非常に短い処理時間で行えるものである．

3. S^3 Bagging

高速に精度の高い分類器を生成する手法として，Bagging とサブサンプリングを併用する S^3Bagging を提案する．S^3Bagging は，Bagging のサンプリング部分にサブサンプリングを適用している．S^3Bagging のアルゴリズムを図 2 に示す．

まず，元学習データ D の r%の事例数分をサブサンプリングした学習データ D_t を T 個生成する．T はメンバー分類器数である．D_t を生成するためのサブサンプリングは，復元抽出法を用いて行う．ここで，サブサンプリング率 $r < 100$ であるので，サンプリングされた学習データの事例数は元学習データの事例数 $|D|$ よりも小さい．次に各学習データ D_t から分類器 C_t を生成する．この生成された T 個のメンバー分類器 C_t により構成される C が，S^3Bagging により生成されるコミッティ分類器である．コミッティ分類器による新規の事例 x を分類は，Bagging と同様に各メ

```
S^3Bagging(D, T, r)
    INPUT:
        D: 元学習データ,
        T: メンバー分類器数,
        r: サブサンプリング率 (%)
        (0 < r < 100).
    OUTPUT:
        C: コミッティ分類器.
        {
            / * 以下を並列処理する */
            for each t from 1 to T
            {
                /* |D| · r/100 個の事例を
                復元抽出法によりサブサン
                プリングする */
                D_t = SubSampling(D, r);
                /* メンバー分類器 C_t
                を生成する */
                C_t := Learning(D_t);
            }
            /* 生成された T 個の分類器により
            コミッティ分類器 C を構成する */
            C := {C_t : t = 1, ..., T};
        }
        return C;
```

図 2 S^3Bagging のアルゴリズム
Fig. 2 Algorithm for S^3Bagging.

ンバー分類器による分類結果 $C_t(x)$ について多数決をとる．

ここで，各サブサンプリングとメンバー分類器の生成は，独立に行われるので並列処理が可能であることに注目されたい．サブサンプリングとメンバー分類器生成の処理に要する時間を $proc\_time_t(t = 1, \ldots, T)$ とする．このとき T 個のサブサンプリングとメンバー分類器の生成が並列に行われる場合には，コミッティ分類器の構成までに要する時間 $proc\_time_c$ は，$proc\_time_c = \max_t proc\_time_t$ である．よって，S^3Bagging において分類器生成に要する処理時間は，サブサンプリングを行ってメンバー分類器を 1 つ生成するのに要する時間とほぼ同じである．また，Bagging と異なりサブサンプリングを行っているため，メンバー分類器の生成に要する時間も一般的な分類器生成手法のように全学習データを用いる場合に比べて短縮することができる．

S^3Bagging は，各メンバー分類器を生成するために準備するサンプリングにおいて，サブサンプリングを行い学習データの事例数を減少させる点で Bagging と異なる．Bagging の枠組みでは，元学習データと同じ事例数をサンプリングすることを前提としており，

表 1 実験データ 1
Table 1 The specification of data sets for experiment 1.

データ名	事例数	属性		クラス数
		数値属性	離散属性	
BREAST-W	699	9	–	2
CREDIT-A	690	6	9	2
CREDIT-G	1,000	7	13	2
HEPATITIS	155	6	13	2
PIMA	768	8	–	2
SICK	3,772	7	22	2
TIC-TAC-TOE	958	–	9	2
VOTE	435	10	16	2

サブサンプリングを行った場合については研究が行われていない．これに対して提案する S^3Bagging では，Bagging を行うことにより分類精度については 1 つの分類器を生成する場合に比べて向上させながら，サブサンプリングとメンバー分類器生成の並列化により処理速度を大幅に短縮することを狙っている．

S^3Bagging のアルゴリズムを特徴づける主なパラメータは，サブサンプリング率 r とメンバー分類器数 T である．サブサンプリング率を大きくすると，学習データの事例数が増えるために分類器の生成に要する時間が増加するが，サブサンプリングによる分類精度の劣化は防ぐことができる．一方，サブサンプリング率を小さくすると，逆に学習データの事例数が減ることにより，一般に分類器の生成に要する時間は短縮されるが分類精度は劣化すると考えられる．また，メンバー分類器数は，コミッティ分類器の分類精度に影響を与える可能性がある．

次章以後では，これらのパラメータが S^3Bagging の性能に与える影響を確認するために実験を行った結果を示す．

4. 実験 1： サブサンプリング率と分類精度

4.1 実 験 内 容

S^3Bagging においてサブサンプリング率を小さく設定することによる分類精度の劣化と，コミッティ学習による分類精度向上の効果について確認することを目的として実験を行った．学習器のアルゴリズムには，決定木アルゴリズムの代表的なものである Quinlan による C4.5 Release 8[16] を用いた．C4.5 の実行時のオプションは，生成される決定木に影響を与えるものについては全てデフォルトのものを用いた．また，以下の分類精度の評価は全て枝刈り後の決定木を用いた．なお，S^3Bagging および Bagging におけるコミッティ分類器を構成するメンバー分類器数 T は，一般的な設定として $T = 10$ とした [17]．

実験で生成した各分類器は，以下の 5 つの方法で学習したものである．

- S^3 Bagging (コミッティ): S^3Bagging により生成されたコミッティ分類器，
- S^3 Bagging (メンバー)：S^3Bagging においてコミッティ分類器を構成する各メンバー分類器，
- Bagging (コミッティ): 通常の Bagging により生成されるコミッティ分類器，
- Bagging (メンバー)：通常の Bagging においてコミッティ分類器を構成する各メンバー分類器，
- All：全学習データから生成された分類器．一般的な分類器の生成手法であり，評価の基準とする．

なお，メンバー分類器に関する実験結果は，コミッティ分類器を構成する各メンバー分類器の平均をとったものである．

4.2 実験データ

UCI Machine Learning Repository[5] のデータの中から，Quinlan[17]，Reinartz[18] が実験に用いているものを参考に選択した．各データの特徴を表 1 にまとめる．実験は，10 分割の交差検定により行った [21]．S^3Bagging と Bagging の実験では，各アルゴリズム中のサンプリングによる影響を考慮するために，交差検定に使う元データから，それぞれ 10 回サンプリングをして分類器を生成し，平均をとっている．

また，コミッティ分類器内のメンバー分類器による分類結果間の多様性を見るために，以下のようなエントロピー E を計算した．

$$E = -\sum_{i=1}^{|C|} p(c_i) \log_2 p(c_i) \qquad (1)$$

ここで，$p(c_i)$ はコミッティ内である事例についてクラス $C = c_i$ と分類したメンバー分類器の割合である．今回実験 1 で用いた実験データでは全てクラス数が 2 であるため $0 \leq E \leq 1$ であり，メンバーが全て同じ分類結果を出したときに 0 となり，各クラスと分類したメンバー分類器が同数のときに 1 となる．

4.3 実 験 結 果

4.3.1 サブサンプリング率と分類精度

サブサンプリング率の設定とコミッティ分類器の分類精度の関係を図 3 に示す．全てのデータについて，サブサンプリング率を大きくすることにより，分類精度が良くなっている．ただし，多くのデータでは，20%より大きいサブサンプリング率にしても分類精度の改善はなくなっている．

次に，全てのデータを使って 1 つの分類器を生成する一般的な分類器生成手法により生成された分類器

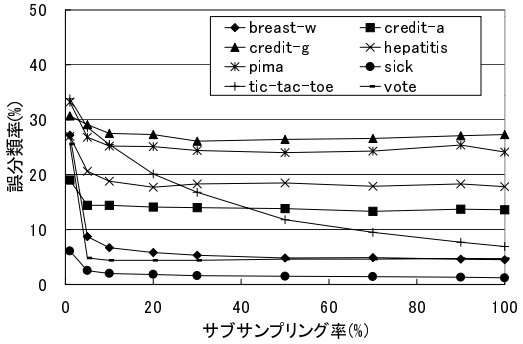


図 3 サブサンプリング率とコミッティ分類器の分類精度（誤分類率）

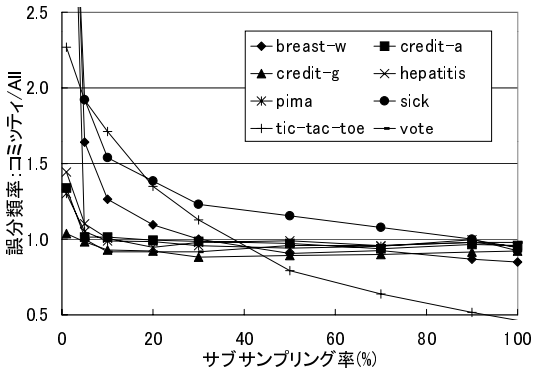


図 4 コミッティ分類器の誤分類率 (B) と全学習データから導出された分類器の誤分類率 (C) の比率 (B/C)

Fig. 4 The ratio(B/C) of the error rate of committee classifier(B) to the error rate of classifier(C) induced from all training data.

(All) と，S^3Bagging により生成されたコミッティ分類器の分類精度について比較した．S^3Bagging により生成されたコミッティ分類器の誤分類率 (B) と一般的な手法を使って生成した分類器 (All) の誤分類率 (C) の比率 (B/C) を図 4 に示す．全てのデータにおいて，サブサンプリング率を高く設定すれば S^3Bagging を行うことにより分類精度を改善することができることがわかる．

メンバー分類器の各サブサンプリング率における分類精度を図 5 に示す．この図からわかるように，サブサンプリングを行うことにより生成されるメンバー分類器の分類精度は悪くなる．特にサブサンプリング率が 1%のときのように極端に学習事例が少なくなる場合には，分類精度が大幅に悪化する．これは，学習データの事例数が正しく学習を行うのに必要な数に対して不足しているためである．

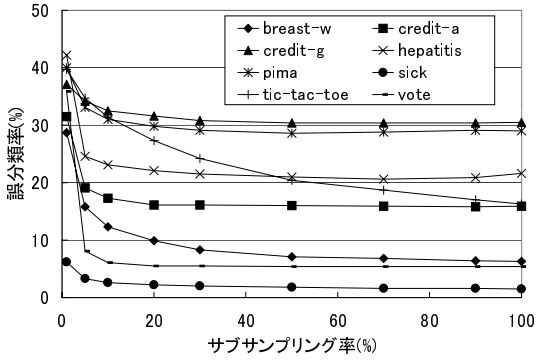


図 5 サブサンプリング率とメンバー分類器の分類精度（誤分類率）
Fig. 5 The subsampling rate vs. prediction accuracy of committee member classifier.

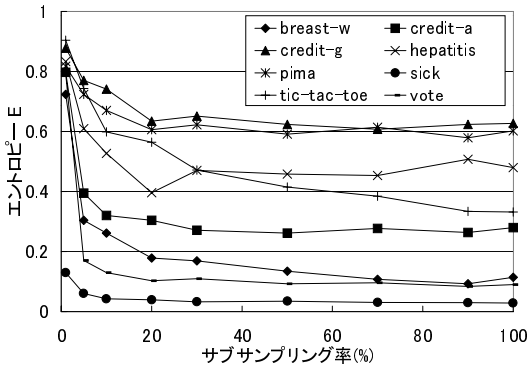


図 7 サブサンプリング率とメンバー分類器による分類結果の多様性（エントロピー）
Fig. 7 The subsampling rate vs. entropy of classfication by committee member classifiers.

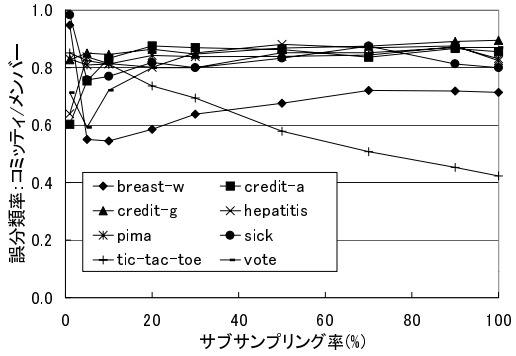


図 6 コミッティ分類器の誤分類率 (B) とメンバー分類器の誤分類率 (A) の比率 (B/A)
Fig. 6 The ratio(B/A) of the error rate of committee classifier(B) to the error rate of member classifier(A).

4.3.2 コミッティ学習による分類精度の改善

次に，コミッティ学習においてコミッティ分類器を構成することによる分類精度の改善について確認する．コミッティ分類器の誤分類率 (B) とメンバー分類器の誤分類率 (A) の比率 (B/A) を図 6 に示す．この比率は，コミッティ学習による分類精度の改善効果を示している．比率が 1 より小さいとき，コミッティ学習により分類精度が改善されている．実験結果より比率は全て 1 より小さくなっており，コミッティ学習を行うことにより分類精度を改善することができることが確認できた．また TIC-TAC-TOE を除いては，サブサンプリング率が 5%以上ではサブサンプリング率が小さいほどコミッティ学習による分類精度の改善が大きい．このようにコミッティ学習は，Bagging のみでなくサブサンプリングを行う S^3Bagging においても分類精度の改善に効果がある．また，分類精度の改善効果は，サブサンプリング率が比較的小さいときに大きい．

コミッティ分類器を構成するメンバー分類器による分類結果の多様性について式 (1) で定義したエントロピーを計算した結果を図 7 に示す．ここで示すエントロピーとは，全テスト事例について計算した値の平均である．コミッティ学習を通じてコミッティ分類器を構成することにより，メンバー分類器に比して分類精度を十分に改善するためには，以下の条件を充たすことが必要である．

(1) コミッティ分類器を構成するメンバー分類器が多様であること．
(2) メンバー分類器の分類精度が一定の水準以上であること．

（1）の条件については図 7 から，サブサンプリング率が 20%以上の場合には，CREDIT-G，PIMA，TIC-TAC-TOE がエントロピーが大きい☆ことがわかる．(2) の条件については，図 5 から TIC-TAC-TOE ではサブサンプリング率が 20%のときには誤分類率が 27.3%であり，サブサンプリング率を大きくすることによりさらに分類精度が高まっている．これに対して，CREDIT-G，PIMA ではそれぞれメンバー分類器の分類精度が 30%前後と悪く，サブサンプリング率を大きくしても改善されていない．このメンバー分類器の分類精度の違いが，TIC-TAC-TOE と CREDIT-G，PIMA の間のコミッティ学習による分類精度の改善効果の違いに表れていると考えられる．

また図 6 で確認できるように，BREAST-W ではサブサンプリング率が大きくなるにつれてコミッティ学習

☆ ここでは，エントロピーが 0.3 以上を基準にしている

表 2 実験データ 2：大規模データ
Table 2 The specification of data sets for experiment 2.

データ名	事例数	属性		クラス数
		数値属性	離散属性	
CENSUS	299,285	7	33	2
LED(10%)	500,000	–	7	10
WAVEFORM	300,000	21	–	3

による分類精度の改善が小さくなっている．これは，メンバー分類器の分類精度は良くなっているが，図 7 のようにメンバー分類器間の多様性が小さくなっていることによるものと考えられる．

5. 実験 2: 大規模データを用いた実験

5.1 実験内容

サブサンプリング率と S^3Bagging の性能の関係とメンバー分類器数が分類精度に与える影響について確認するために，大規模データを使った実験を行った．S^3Bagging の性能については，生成されるコミッティ分類器の分類精度と処理速度について確認した．

処理時間 (CPU Time) とは，サンプリングに要する時間と分類器の生成に要する時間の和である．また，S^3Bagging や Bagging などのコミッティ学習では，各メンバー分類器の生成は並列に行えるものとして，コミッティを構成する各メンバー分類器の生成に要した時間のうち最も長いものをコミッティ分類器の生成に要した時間としている．なお，実験に用いた計算機は，OS が Linux OS，CPU が PentiumIII 700MHz，メインメモリが 256M のものである．

実験に用いるデータとして，UCI KDD Repository[4] から CENSUS-INCOME（以下，CENSUS と略す）を，UCI Machine Learning Repository[5] から LED(10%) と WAVEFORM を選択した．データの特徴を表 2 にまとめる．データを選択する際には，Provost[15] らが実験に用いているものを参考にした．LED(10%) と WAVEFORM は，Repository で指定した事例数のデータを生成するプログラムが配布されている．また，LED は，データ生成の際にノイズを加える事例の割合を指定することができる．事例にノイズを加える際には，ランダムに 1 つの属性を選び，その属性値を他の属性値に変更するという操作を行う．“LED(10%)”という表記は，ノイズが 10%の事例に加えられていることを表している．今回の実験では，ノイズを加える事例の割合を過去の研究でデフォルトとして用いられている 10%としている．

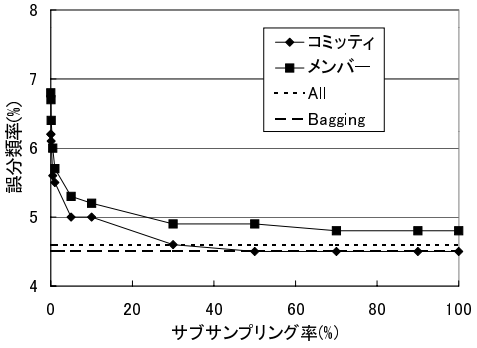


図 8 サブサンプリング率と誤分類率：CENSUS
Fig. 8 The subsampling rate vs. error rate: CENSUS.

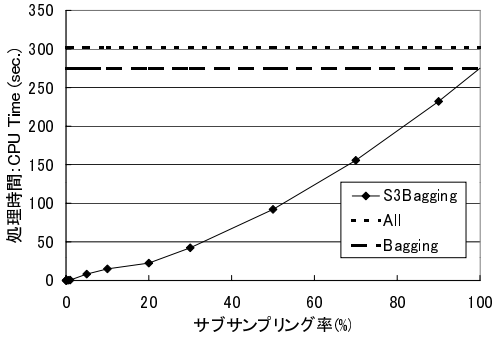


図 9 サブサンプリング率と処理時間：CENSUS
Fig. 9 The subsampling rate vs. processing time: CENSUS.

5.2 実験結果

5.2.1 サブサンプリング率と S^3Bagging の性能

各データの実験結果のうち分類精度について，図 8，10，12 に示す．LED(10%) については，1%以下という比較的小さいサブサンプリング率で分類精度の変化が大きく，その後一定となったため，ここではサブサンプリング率が 5%以下の部分のみを示した．実験 1 の結果として図 6 で確認されたのと同様に，メンバー分類器よりもコミッティ分類器の方が常に分類精度が良い．すなわち，コミッティ学習により分類精度が改善されている．また，コミッティ分類器の分類精度は，メンバー分類器よりも小さいサブサンプリング率で一定となっている．サブサンプリング率が CENSUS では 30%，LED(10%)，WAVEFORM では 0.5%を超えると，S^3Bagging により生成したコミッティ分類器の方が全学習データを用いて生成した分類器 (All) よりも分類精度が良くなる．また，Bagging と比較すると，サブサンプリング率が CENSUS では 50%，LED(10%) では 0.5%，WAVEFORM では 5%をそれぞれ超えると

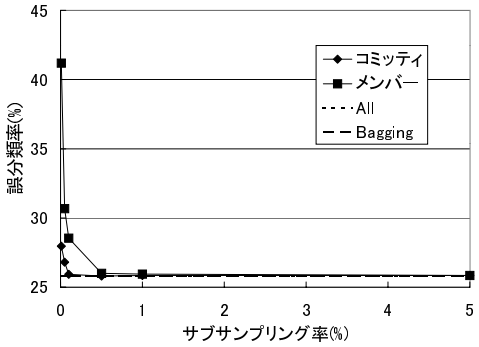


図 10 サブサンプリング率と誤分類率：LED(10%):
Fig. 10 The experimental result of LED(10%): error rate.

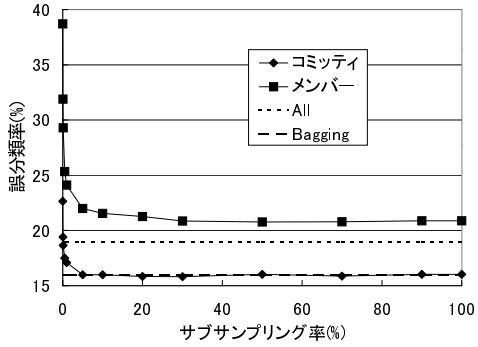


図 12 サブサンプリング率と誤分類率：WAVEFORM
Fig. 12 The subsampling rate vs. error rate: WAVEFORM.

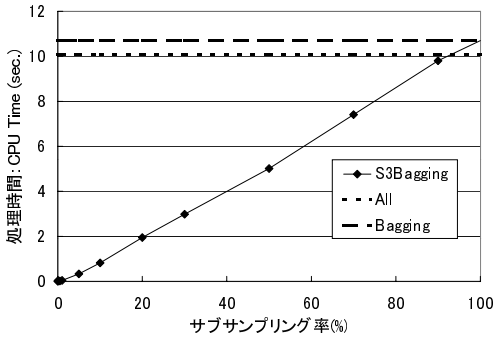


図 11 サブサンプリング率と処理時間：LED(10%)
Fig. 11 The subsampling rate vs. processing time: LED(10%).

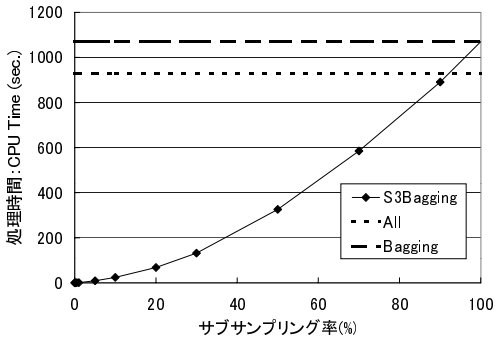


図 13 サブサンプリング率と処理時間：WAVEFORM
Fig. 13 The subsampling rate vs. error rate: WAVEFORM.

Bagging による分類器よりも S^3Bagging の分類器による分類器の方が分類精度が良くなる．

次に処理時間に関する各データの実験結果について，図 9，11，13 に示す．CENSUS において，Bagging にはサンプリングに要する時間があるにも関わらず All の方が処理時間が短いのは，Bagging で生成されたメンバー分類器が All のものよりも単純なものであったため分類器の生成に要する時間が短かったことによる．処理速度は，サブサンプリング率にほぼ比例している．このことから，サブサンプリング率を低くすることにより処理時間が大幅に短縮することができることが分かる．例えば，CENSUS では，サブサンプリング率が 5%のときで S^3Bagging に要する時間は 8.3 秒程度であり，一般的な分類器生成手法 (All)，および Bagging に要する処理時間の 3%弱である．

5.2.2 メンバー分類器数の分類精度への影響

次に，コミッティ分類器を構成するメンバー分類器数 T を変化させた場合について，各サブサンプリング率におけるコミッティ分類器の分類精度への影響について確認した．メンバー分類器の数は，$T = 10, 20, 30$ の 3 とおりに設定した．各データについての実験結果を図 14，15，16 に示す．図中では，それ以上サブサンプリング率を高くしてもコミッティ分類器の分類精度の改善がなくなる点より小さい部分を中心に示している．

何れのデータの場合も，メンバー分類器数を増やすことによりコミッティ分類器の分類精度が良くなっている．LED(10%) と WAVEFORM の場合では，特にサブサンプリング率が小さい場合に，メンバー分類器数を増やすことによる分類精度の改善が大きい．このようにサブサンプリング率が小さく，メンバー分類器の分類精度が悪い場合には，メンバー分類器数を増やすことによりコミッティ分類器のより良い分類精度を得ることができる可能性がある．S^3Bagging では，各メンバー分類器の生成を並列で行える場合には，処理速度はメンバー分類器数に関わらず一定である．よって，計算機環境が許す限りのメンバー分類器数でコミッティ分類器を構成することが，より良い分類精度の分類器を得るために有効である．

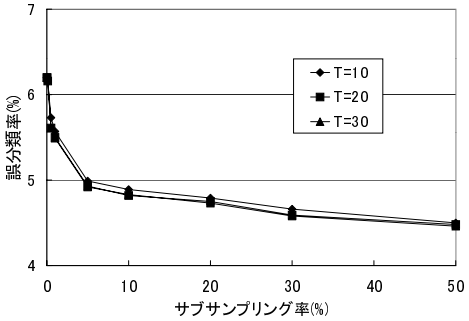


図 14 メンバー分類器数の分類精度への影響:CENSUS
Fig. 14 The effect of the number of committee member classfiers on the prediction accuracy: CENSUS.

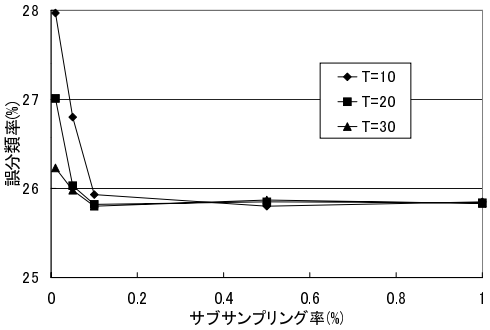


図 15 メンバー分類器数の分類精度への影響:LED(10%)
Fig. 15 The effect of the number of committee member classfiers on the prediction accuracy: LED(10%).

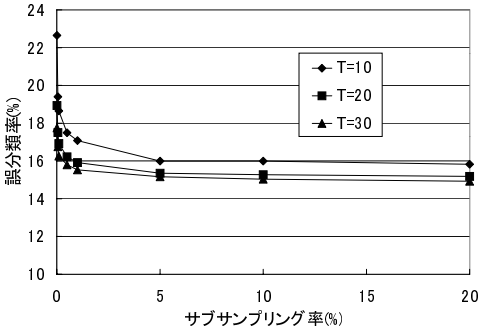


図 16 メンバー分類器数の分類精度への影響:WAVEFORM
Fig. 16 The effect of the number of committee member classfiers on the prediction accuracy: WAVEFORM.

6. 考　　察

4 章の実験結果から，10%程度の低いサブサンプリング率でサンプリングを行った場合でも，S^3Bagging が採用しているコミッティ学習を行うことによって，全学習データより分類器を生成した場合からの分類精度の劣化を防ぐことができることが確認された．また，5 章の実験結果からは，S^3Bagging が採用しているサブサンプリングを行うことにより，分類器の生成に必要な処理時間を大幅に削減できることを確認した．また，メンバー分類器数を増やすことにより，コミッティ分類器の分類精度の改善を期待できることを確認した．

データマイニングの過程では，一度の知識抽出で必要な知識を取り出せることは少なく，抽出された知識を分析者が吟味しながら知識抽出を重ねなければいけない場合が一般的である．このように，意思決定者とのインタラクションの中で知識抽出を行うことを考えると，短い時間で知識を提示することは，意思決定者にその後の知識抽出のきっかけを与えるという点で非常に重要である．よって，S^3Bagging のように抽出される分類器の質の劣化を抑えながら，大規模データから高速に分類器を生成するような枠組みは，データマイニングツールの実用性の観点から有用である．

分類器の生成に必要な処理時間のコストと，分類精度により代表される分類器の質との間のバランスという観点から 4 章および 5 章の実験結果を見ると，データにより異なるが，サブサンプリング率で 10%–30%程度，あるいは学習データの事例数で数百から数千ぐらいに設定するのが適切であると考えられる．ただし，これは今回の実験で用いた C4.5 を学習器として適用した場合の結果にもとづいたものであり，ニューラルネットワークなど他のアルゴリズムを学習器に適用した場合については，実験を通じた検討が必要である．

サブサンプリング率については，データに対して事前に与えられる知識から適切に設定することは難しい．一方，サンプリング中に，サンプリングを打ちきるサブサブサンプリング率やサンプリング数を決定するには，事例の内容について十分に吟味すれば可能である．しかしながら，一般的には事例の吟味に多くの処理時間を要する．

筆者等は，処理時間が少なく，かつサンプリングの打ちきり点を決定するための参考指標として，次のようなサンプリングされた事例の記述長である $DL_{sampled\_data}$ を採用することについて検討した．

$$DL_{sampled\_data}(r) = \sum_{i=1}^{N} \log_2 nvc_i(r) \qquad (2)$$

ここで，N は属性数，$nvc_i(r)$ はサブサンプリング率が r の時点までにサンプリングされたデータ中に現れた異なる (属性 A_i の属性値, クラス値）の組の種類数である．これは，サブサンプリング率が r の時点まで

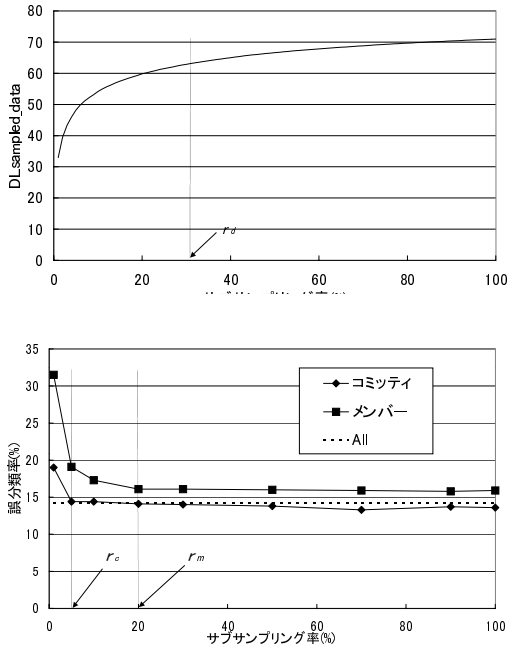


図 17 記述長 $DL_{sampled\_data}$（上）と誤分類率（下）：CREDIT-A
Fig. 17 The subsampling rate vs. the description length $DL_{sampled\_data}$(upper) and error rate(lower): CREDIT-A.

にサンプリングされたデータに含まれる属性値とクラス属性の組み合わせのパターンを記述するために必要な記述長である．この記述長の計算は，各属性の属性値とクラス値の組を数え上げるだけなので，わずかな処理時間で計算することができる．

各サブサンプリング率における記述長 $DL_{sampled\_data}$ とコミッティ分類器の誤分類率の変化について，ここでは一般的な例として CREDIT-A の例を図 17 に示す．多くのデータでは図 17 のように，記述長の増加量が小さくなるサブサンプリング率 r_d，メンバー分類器の誤分類率の改善がなくなるサブサンプリング率 r_m，およびコミッティ分類器の誤分類率がなくなるサブサンプリング率 r_c の間に以下のような関係が見られる．

- 記述長の増加量が小さくなるとメンバー分類器の誤分類率の改善も小さくなっている ($r_m \leq r_d$)．
- コミッティ学習による誤分類率の改善は，図 8, 10, 12 にも示したようにメンバー分類器よりも小さいサブサンプリング率で改善がなくなっている ($r_c < r_m$)．
- よって，記述長の増加量が小さくなった時点では，コミッティ分類器の誤分類率の改善もなくなっている ($r_c < r_m \leq r_d$)．

このように記述長は $DL_{sampled\_data}$ と増加量が小さくなる点 r_d は，サンプリングを打ちきる点，すなわちコミッティ分類器の誤分類率の改善が十分に小さくなっている点を判断するのに有効である．記述長の増加量が小さくなった点 r_d については，実験から記述長の 2 階微分が -0.01 より小さくなった点をもって判断するのが適当であることが確認された．表 3 にこの判断基準にもとづいた場合のサンプリング打ち切り点を実験から求めた結果を示す．

しかしながら，この記述長は定義のように各属性とクラス間のパターンの多様性について計算されるものであり，属性間の相関については反映されない．よっ

表 3 $DL_{sampled\_data}$ の 2 階微分値から判断されるサンプリング打ち切り点 r_d
Table 3 The finishing point of sampling r_d derived from the second derivative coefficient of $DL_{sampled\_data}$.

データ名	r_d
BREAST-W	30%
CREDIT-A	31%
CREDIT-G	11%
HEPATITIS	36%
PIMA	37%
SICK	33%
TIC-TAC-TOE	9%
VOTE	11%

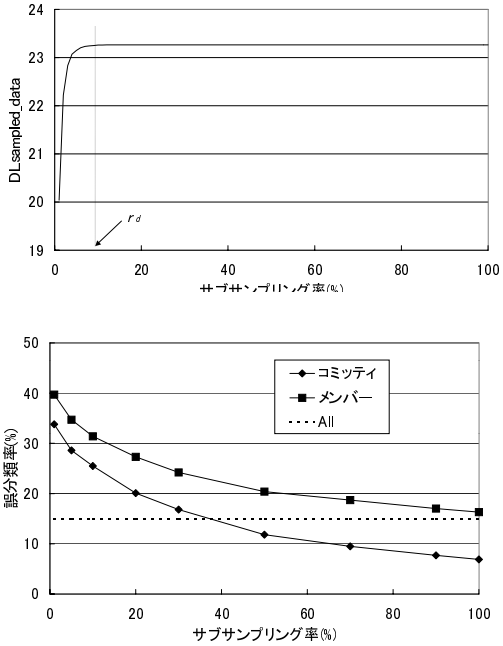


図 18 記述長 $DL_{sampled\_data}$ と誤分類率 (TIC-TAC-TOE)
Fig. 18 The subsampling rate vs. the description length $DL_{sampled\_data}$(upper) and the error rate(lower): TIC-TAC-TOE.

表 4 メンバー分類器による分類結果の一致率 (VOTE, S^3Bagging のサブサンプリング率：10%)
Table 4 The rate of identical classifications among committee member classifiers (VOTE, subsampling rate: 10%).

		All	
		正答	誤答
S^3Bagging (10%)	正答	96.0%	81.7%
	誤答	78.2%	86.0%

て，例えば属性間に強い属性が存在することが知られている [10] TIC-TAC-TOE の場合は，図 18 のように誤分類率の改善が小さくなる前に記述長は一定となる．属性間に相関があることが分っている場合には，サブサンプリング率を高めに設定することが必要である．

S^3Bagging により生成されるコミッティ分類器の内容を，そのままルール表現すると非常に複雑なものとなるため理解するのが困難である [7]．しかしながらコミッティ分類器からはデータマイニングに有用な情報として，分類器および分類結果の他にコミッティ分類器を構成する各メンバー分類器による分類結果も得ることができる．

例として，サブサンプリング率が 10%の S^3Bagging により生成されたコミッティ分類器と全学習データから生成された分類器 (All) による分類が，それぞれ正答および誤答であった場合ごとにコミッティ分類器を構成するメンバー分類器による分類結果が一致していた割合（一致率）を表 4 に示す．実験の設定は，実験 1 と同じであり，実験データには VOTE を用いている．S^3Bagging と All がともに正答している事例については，ほぼ全てのメンバー分類器が同じ分類結果を出している．これに対して，何れかのメンバー分類器が誤っているような分類が難しい事例については，メンバー分類器間で分類結果が異なっている場合が多い．

このように，メンバー分類器間の分類結果の一致率を見ることにより，分類が困難である特徴的な事例を抽出することができる．生成されたコミッティ分類器の情報だけでなく，合わせてメンバー分類器間の分類結果の一致率を適切な形で提示することができれば，データマイニングを行う意思決定者に示唆を与えられると考えられる．メンバー分類器による分類結果を含むコミッティ分類器の提示方法の検討については，今後の課題としたい．

7. 関 連 研 究

Weiss らはサブサンプリング率と得られる分類器の分類精度の関係について，学習器に様々な学習アルゴリズムを用いた実験を行っている [22]．その中で，最も良い分類精度が得られるサンプリング率が学習アルゴリズムごとに異なることを示している．また，コミッティ学習に Arcing[3] を用いて，サブサンプリングした学習データから生成した分類器によりコミッティ分類器を構成した場合の分類精度の変化を確認する実験も行っている．実験結果として，サブサンプリングを行った場合でもコミッティ学習を行うことにより分類精度の劣化を抑えることができることを示しているが，処理時間の点については言及していない．

Reinartz は，サンプリングによる分類精度の向上と新たに必要となる計算時間の両面から，サンプリング手法の有効性について実験を通じた研究を行っている [18]．ただし，実験で用いられているデータは小規模なものであり，大規模データに適当した場合の処理時間と分類精度については議論されていない．

8. お わ り に

本論文では，大規模なデータから高速に分類器を生成する手法として，サブサンプリングと Bagging を併用する S^3Bagginig を提案し，テストデータを用いた実験を通じて性能を確認した．

S^3Bagging では，サブサンプリングにより学習データの事例数を減らし，かつコミッティを構成する分類器の生成を並列に行うため，処理時間はサブサンプリング率にほぼ比例する程度である．大規模データを用いた実験からは，サブサンプリング率を 20%程度に設定した場合には，全学習データを用いる場合に比べて計算時間を 20%程度に抑えることが可能であることを確認した．

また，S^3Bagging は，Bagging の方法によりコミッティ学習を行うことにより，サブサンプリングによる分類精度の低下を抑えることができる．実験結果から，サブサンプリング率が 20%程度にした場合に誤分類率の劣化は多くの場合 20%程度に抑えることができることを確認した．

さらに，コミッティ分類器を構成するメンバー分類器数を増やすことにより，分類精度を改善できることを確認した．この効果は，特にサブサンプリング率が小さい場合に顕著である．

このように，提案手法は並列処理が可能な計算機環境下において，大規模データから高速に，かつ分類精度の高い分類器を生成するのに有効である．

今後の検討課題として，以下のようなものを考えている．

- 並列計算機への実装: S^3Bagging を並列計算機に

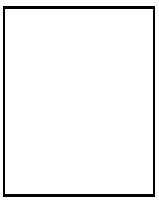
実装し，評価実験を行う．

- 属性間の相関を考慮したサンプリング率決定のための指標の検討:サブサンプリング率を決定する際の参考指標として提案した記述長 $DL_{sampled\_data}$ は，属性間の相関を考慮していないものである．注目する属性数を絞り込むことにより処理時間を小さくしながら，属性間の相関も考慮することができる指標について検討する必要がある．
- S^3Bagging により抽出された情報の効果的な提示方法の検討：6 章で述べたように，S^3Bagging では抽出されるコミッティ分類器以外にも，メンバー分類器による分類結果などデータマイニングを進める上で有用な情報が含まれている．これら情報の意思決定者への効果的な提示方法について検討していきたい．

参 考 文 献

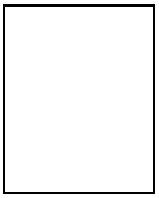
1) Addriaans, P. and Zantinge, D.: *Data Mining*, Addison Wesley (1996).
2) Breiman, L.: Bagging Predictors, *Machine Learning*, **24** (2), pp.123–140 (1996).
3) Breiman, L.: Bias, variance, and arcing classifiers, Technical Report 460 UC-Berkeley, Berkeley, CA. (1996).
4) Bay, S. D.: The UCI KDD Archive, `http://kdd.ics.uci.edu`, Irvine, CA: University of California, Department of Information and Computer Science (1999).
5) Blake, C., Keogh, E. and Merz, C.J.: UCI Repository of machine learning databases, `http://www.ics.uci.edu/~mlearn /MLRepository.html`, Irvine, CA: University of California, Department of Information and Computer Science (1998).
6) Catlett, J.: Megainduction: a Test Flight, In *Proceedings of the Eighth International Workshop on Machine Learning*, pp.596–599. (1991)
7) Dietterich, T.G.: Machine Learning Research: Four Current Directions, *AI Magazine*, **18** (4), pp.97-136 (1997).
8) ヨアブ　フロインド，ロバート　シャピリ，安部直樹（訳）：ブースティング入門，人工知能学会誌，**14** (5), pp.771–780 (1999).
9) 河野浩之：データベースからの知識発見の現状と動向，人工知能学会誌，**12** (4)，pp.497–504 (1997).
10) Matheus, C.J., Adding Domain Knowledge to SBL thorough Feature Construction, In *Proceedings of the Eighth National Conference on Artificial Intelligence*, pp.803–808 (1990).
11) 森口 繁一（編）：新編 統計的方法 改訂版，日本規格協会 (1989).
12) 野澤 恵： Linux Cluster, *bit*, **31** (9), pp.3–13 (1999).
13) Opitz, D. and Maclin, R.: Popular Ensemble Methods: An Empirical Study, *Journal of Artificial Intelligence Research*, **11**, pp.169–198 (1999).
14) Provost, F. and Kolluri, V.: A Survey of Methods for Scaling Up Inductive Algorithms, *Knowledge Discovery and Data Mining*, **3** (2), pp.131–169 (1999).
15) Provost, F., Jensen, D., and Oates, T.: Efficient Progressive Sampling, In *Proc. of the Fifth International Conference on Knowledge Discovery and Data Mining*, pp.23–32 (1999).
16) Quinlan, R.: *C4.5: Programs for Machine Learning*, Morgan Kaufmann (1993).
17) Quinlan, R.: Bagging, boosting , and C4.5., In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pp.725–730 (1996).
18) Reinartz, T.: Similarity-Driven Sampling for Data Mining, Zytkow, J.M. and Quafafou, M. (eds.). *Principles of Data Mining and Knowledge Discovery: Second European Symposium PKDD '98*, pp.423–431 (1998).
19) Shapire, R.E.: A brief introduction to boosting, In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, pp.1401–1406 (1999).
20) 鷲尾 隆：ビジネスにおけるデータマイニングの現在・未来，情報処理，**42** (5)，pp.467–471 (2001).
21) Weiss, S.M. and Kulikowski, C.A.: *Computer Systems That Learn*, Morgan Kaufmann (1991).
22) Weiss, S.M. and Indurkhya, N.: *Predictive Data Mining – a practical guide –*, Morgan Kaufmann (1998).
23) Zheng, Z.: Generating Classifier Committees by Stochastically Selecting both Attributes and Training Examples, In *Proceedings of the Third Pacific Rim Conference on Artificial Intelligence*, pp.12–33 (1998).

(平成 10 年 5 月 1 日受付)
(平成 10 年 5 月 31 日採録)



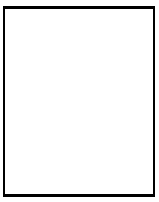
寺邊　正大（正会員）

1970 年生．1993 年京都大学工学部精密工学科卒業．1995 年京都大学大学院工学研究科修士課程修了．同年（株）三菱総合研究所入社．現在，総合安全研究センターで，データマイニング，機械学習，マルチエージェント技術の大規模システム運転支援・診断への適用，およびヒューマン・マシンシステムの設計に関する研究に従事．2000 年計測自動制御学会学術奨励賞受賞．人工知能学会，計測自動制御学会，日本原子力学会，AAAI，各会員．



鷲尾　　隆（正会員）

1960 年生．1983 年東北大学工学部原子核工学科卒業．1988 年東北大学大学院原子核工学専攻博士課程修了．工学博士．1988 年から 1990 年にかけてマセチューセッツ工科大学原子炉研究所客員研究員．1990 年（株）三菱総合研究所入社．1996 年退社．現在，大阪大学産業科学研究所助教授（知能システム科学研究部門）原子力システムの異常診断手法に関する研究，定性推論に関する研究を経て，現在は人工知能の基礎研究，特に科学的知識発見，データマイニングなどの研究に従事．著書に “*Expert Systems Applications within the Nuclear Industry*”, American Nuclear Society,「知能工学概論」: 第 2 章エージェント（共著，廣田　薫　編，昭晃堂）など．人工知能学会，計測自動制御学会，日本ファジイ学会，日本原子力学会，AAAI，各会員．



元田　　浩（正会員）

1943 年生．1965 年東京大学工学部原子力工学科卒業．1967 年東京大学大学院原子力工学専攻修士課程修了．同年（株）日立製作所入社．同社中央研究所，原子力研究所，エネルギー研究所，基礎研究所を経て 1995 年退社．現在，大阪大学産業科学研究所教授（知能システム科学研究部門）原子力システムの設計，運用，制御に関する研究，診断型エキスパート・システムの研究を経，現在は人工知能の基礎研究，特に機械学習，知識獲得，知識発見などの研究に従事．工学博士．認知科学会，人工知能学会，情報処理学会，日本ソフトウエア科学会，AAAI, IEEE Computer Society，各会員．