

# 教師あり学習の導入による Mondrian Forests の効率化

村田 隆英<sup>†</sup> 木村 昭悟<sup>††</sup> 牛久 祥孝<sup>††</sup> 山下 隆義<sup>†</sup> 山内 悠嗣<sup>†</sup>  
藤吉 弘亘<sup>†</sup>

<sup>†</sup> 中部大学 〒487-8501 愛知県春日井市松本町 1200

<sup>††</sup> 日本電信電話株式会社 コミュニケーション科学基礎研究所 〒243-0198 神奈川県厚木市森の里若宮 3-1

E-mail: †{mryua@vision., yamashita@, yuu@vision., hf@}cs.chubu.ac.jp,

††{kimura.akisato, ushiku.yoshitaka}@lab.ntt.co.jp

あらまし Random Forests のオンライン学習法である Mondrian Forests は、学習時に、木構造全体を再構築するのではなく、学習済みデータと追加データの差異に基づいて更新が必要と判定された任意のノードのみを更新することで、計算コストを抑えた効率の良いオンライン学習を実現している。しかし、Mondrian Forests は学習時に教師ラベルを用いないため、不要なノードを追加することがあり、木構造が肥大化し、木構造の構築に必要なメモリ量が増加する問題がある。本研究では、オンライン学習である Mondrian Forests の効率的なフレームワークに、教師あり学習を導入し、木構造の構築に必要なノード数を削減することで、メモリ量を抑えた効率的な学習法を提案する。提案手法では、教師ラベル情報を用いた分岐関数の設計とノードの追加判定を導入する。評価実験により、提案手法は従来法と比較してノード数を約 68%削減できることを確認した。

キーワード Random Forests, Mondrian Forests, 機械学習, オンライン学習

## Efficient Mondrian Forests by Introducing Supervised Learning

Ryuei MURATA<sup>†</sup>, Akisato KIMURA<sup>††</sup>, Yoshitaka USHIKU<sup>††</sup>, Takayoshi YAMASHITA<sup>†</sup>, Yuji YAMAUCHI<sup>†</sup>, and Hironobu FUJIYOSHI<sup>†</sup>

<sup>†</sup> Chubu University, 1200, Matsumoto, Kasugai, Aichi, 487-8501 Japan

<sup>††</sup> Communication Science Laboratories NTT Corporation, 3-1, Morinosato Wakamiya, Atsugi, Kanagawa, 243-0198 Japan

E-mail: †{mryua@vision., yamashita@, yuu@vision., hf@}cs.chubu.ac.jp,

††{kimura.akisato, ushiku.yoshitaka}@lab.ntt.co.jp

**Abstract** Mondrian Forests is an online learning method based on framework of Random Forests. At the online learning, Mondrian Forests add a node where is judged as necessary for updating by based on difference between trained data and additional data. This makes fast online training of decision trees. Since Mondrian Forests use may label information at the updating, tree structure contain unnecessary nodes. Therefore, this paper presents an efficient framework of online learning for Mondrian Forests by introducing supervised learning. The proposed method uses label information to design splitting function, and to decide whether additional node is necessary or not. The proposed method can reduce the size of decision tree about 68%, comparing with the conventional method.

**Key words** Random Forests, Mondrian Forests, Machine Learning, Online Learning

### 1. はじめに

機械学習 [1] [2] は、高い汎用性から自然言語処理、音声処理や画像処理など広い分野で用いられている。機械学習では学習データを利用して識別器を構築する。機械学習において、識別

器の構築に利用する学習データの量は重要であり、大規模な学習データを用いて汎化性能の高い識別器を構築する需要が高まっている [3]。近年のインターネットや SNS の普及により学習データの収集が容易になったことで、大規模な学習データを十分に時間をかけ学習し、高い性能を持つ学習器が得られるよ

うになった。しかし、データ量が増加することで、機械学習では難しい問題を解くことを可能としているが、処理時間の増加や、使用メモリの増加などの問題がある。これらの問題を解決するために複数の計算機により処理を行う分散コンピューティングや、複数の CPU コアや GPU を用いて処理を行う並列コンピューティングが用いられている。その他の問題として、学習したモデルを実際に運用する際に、消費者のニーズの変化や、環境変化によるデータ分布の推移より、分類・回帰精度が低下することがある。このような問題を解決するために、逐次的に現在の環境に学習モデルを更新するオンライン学習が用いられている。オンライン学習は、与えられた1つずつ、もしくは全体の一部のデータに対して逐次的にモデルの更新を行うことで、高速かつ省メモリでの学習が可能となる。

Breiman により提案された Random Forests [4] は、複数の決定木構造を持つマルチクラス識別器を構築するアンサンブル学習アルゴリズムである。Random Forests では決定木に Bagging [5] と同様にブートストラップサンプリングを取り入れることで過学習を抑制し、Random Feature Selection [6] を取り入れることで特徴ベクトルの次元数に関わらず高速に学習することが可能である。また、Random Forests は入力と出力を変更するだけで、分類・回帰分析・クラスターリング・密度推定など様々な問題に適用可能である [7]。Random Forests は設計の自由度が高く学習・評価が高速である点から、画像認識のアプリケーションとしても利用されている [8]。また、Random Forests はオンライン学習に適用可能であり、様々な手法 [9] [10] が提案されている。

本研究では、Random Forersts を用いたオンライン学習の中でも、高速な学習が可能である Mondrian Forests [11] のフレームワークを使用する。Mondrian Forrsts は学習の際に、木構造を全て更新せず、更新が必要と判断した場合にのみノードを追加することで、計算コストを抑えた効率の良い学習を実現している。しかし、Mondrian Forests は学習時に教師ラベルを用いていないため、学習時に不要なノードを追加する可能性がある。その場合、最小構成での決定木を構築できないため、決定木群のパラメータを保存するために必要なメモリ量が増加するという問題がある。そこで、本研究では教師なし学習である Mondrian Forests に教師あり学習を導入する。提案手法は、教師ラベルを用いたノードの追加の判定と、分岐ノードにおいて通常の Random Forests と同様の情報利得による分岐候補の選択を導入することで、決定木の構築に必要なメモリ量を削減する。

## 2. Mondrian Forests

本研究では、Random Forersts のフレームワークを用いてオンライン学習を行う。以下に、Mondrian Forests [11] の学習・識別と学習の問題点について述べる。

### 2.1 Mondrian Forests の学習

Mondrian Forests は、Random Forests と同様に決定木を構築することで学習を行う。Random Forests との相違点として、図 1 に示すように、分岐・末端ノードは到達した学習サンプル

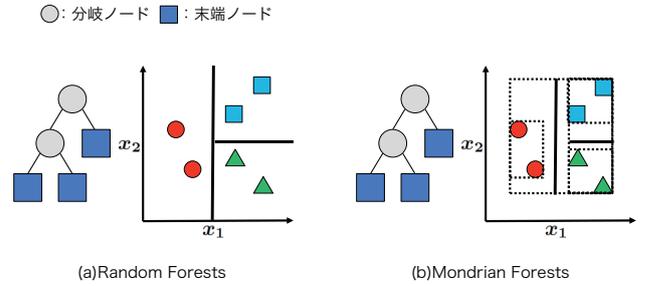


図 1 Mondrian Forests の構造

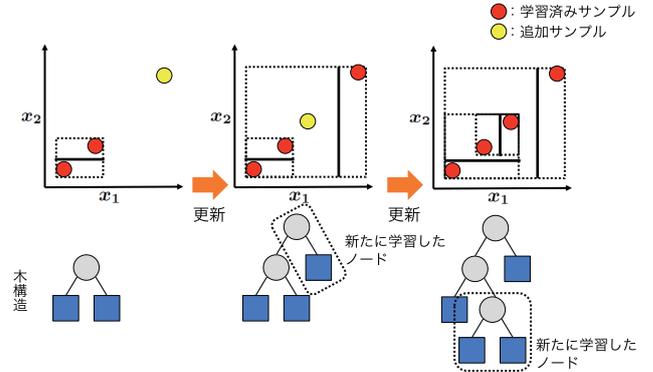


図 2 Mondrian Forests による学習

集合の全特徴次元における最小値と最大値から求めた特徴量の範囲情報を保持している。Mondrian Forests の学習は、1 サンプルを入力する毎に実行され、図 2 に示すように、決定木をトラバースし、トラバース中にノードが持つ特徴量の範囲外れた場合にノードの追加を行う。Mondrian Forests は、決定木の各ノードの分岐関数の決定に教師ラベルを用いず、サンプル分布を考慮し分散が大きな次元を優先的を選択する。このとき、特徴次元と閾値は複数の候補から選ばれるのではなく、1 度しか選択しない。これは、文献 [12] の分岐関数の設計と類似している。Mondrian Forests は、木構造を更新の際に木構造を全て再学習せず、適応的にノードを追加すること、分岐関数の選択を 1 度しか行わないことで高速な学習を実現している。

### 2.2 識別

Mondrian Forests の識別は、Random Forests と同様に、未知サンプルを  $v$  とした時に、学習により作成された全ての決定木の到達した末端ノード  $l$  におけるカテゴリ  $c$  の確率  $P(c|l)$  を出力する。全ての決定木から出力された  $(P_1(c|l), P_2(c|l), \dots, P_t(c|l))$  を用いて、未知サンプル  $v$  の事後確率  $P(c|v)$  を式 (1) で求める。

$$P(c|v) = \frac{1}{T} \sum_{t=1}^T P_t(c|l) \quad (1)$$

ここで、 $T$  は決定木の数である。この式により未知のサンプルを入力した際にカテゴリである事後確率を求めることができる。そのため、未知のサンプルのクラスを推定するには、以下の式の単純ベイズ識別器により未知のサンプルのクラスを推定する。

$$C_i^* = \arg \max_{c_i} P(c_i|v) \quad (2)$$

### 2.3 Mondrian Forests の問題点

Mondrian Forests は、学習サンプルと追加サンプルの差異に

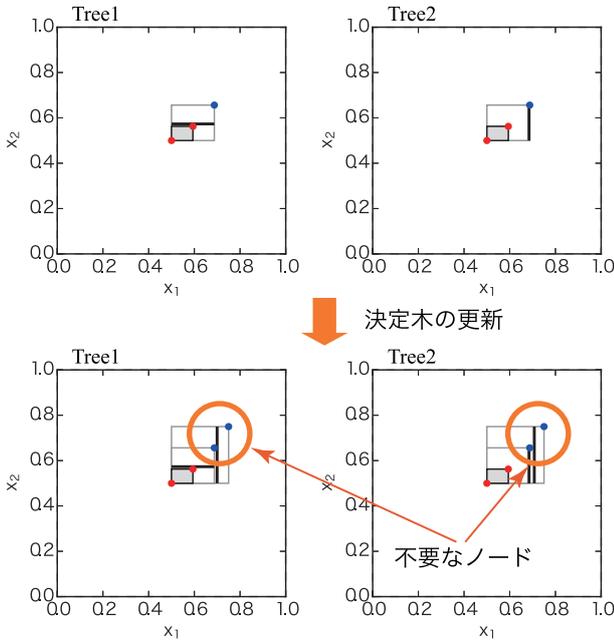


図3 Mondrian Forests の問題点

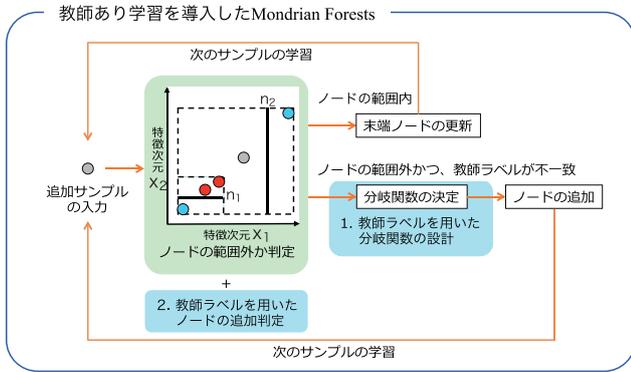


図4 提案手法の流れ

基づいて、更新が必要と判定されたノードのみを更新することで、高速なオンライン学習を実現している。しかし、図3に示すように、Mondrian Forestsは教師ラベルを用いない学習法のため、ノード内のサンプルの教師ラベルが一致し、既に分岐の必要がない場合においてもノードを追加することがあり、不要なノードを追加する。また、これにより、各ノードの分岐関数の決定の際に最適な関数を選択できていないため、分岐ノードの数も増加する傾向がある。これらのことから、木構造が肥大化するという問題がある。そこで、本研究はMondrian Forestsの効率化を目的とし、教師ラベルを用いた分岐関数の設計とノードの追加判定を導入したオンライン学習法を提案する。

### 3. 提案手法

本研究では、教師なし学習であるMondrian Forestsに教師あり学習を導入することで効率化を図る。提案手法は、2つのアプローチにより高精度かつ効率的な学習を行う。提案手法の流れを図4に示す。

#### 3.1 教師ラベルを用いた分岐関数の設計

Mondrian Forestsの分岐関数は、教師ラベルを用いない

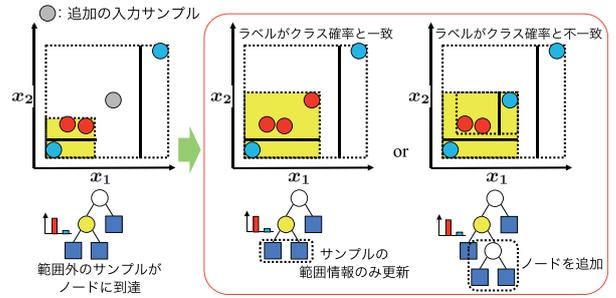


図5 教師ラベルによるノードの追加判定

め、必ずしも最適な分岐関数が選択されているとは限らない。そこで、提案手法ではRandom Forestsと同様に分岐関数の評価を行う。各決定木の構築は、入力データを分岐ノードにより分岐することで行う。各分岐ノードに対し入力データが $S_n$ 、左右に分岐するデータがそれぞれ、 $S_l$ 、 $S_r$ であるとすると、分岐条件は式(3)、(4)で表される。

$$S_l = \{i \in S_n | f(v_i) < t\} \quad (3)$$

$$S_r = S_n \setminus S_l \quad (4)$$

ここで、 $f(v_i)$ はサンプル $v_i$ の特徴量、 $t$ は閾値であり、これらはランダムに選択される。 $f()$ は、任意の分岐関数で、特徴量を入力した際の応答値である。上記のサンプルの分岐を行った後、式(5)で情報利得(Information gain) $\Delta E$ を求め、その分岐を評価する。

$$\Delta E = -\frac{|S_l|}{|S_n|} E(S_l) - \frac{|S_r|}{|S_n|} E(S_r) \quad (5)$$

ここで、関数 $E(S)$ とは、情報エントロピーであり、サンプルに付与されている教師ラベルを用いて以下の式で算出する。

$$E(I) = -\sum_{i=1}^n P_i \log P_i \quad (6)$$

$P_i$ はサンプル集合 $S_n$ に含まれる教師ラベルから得られるクラス分布、 $n$ は学習する教師ラベルの数である。サンプルの分岐と評価を繰り返すことで、様々な特徴量と閾値の組み合わせをランダムで選択し、最も式(5)が大きくなった組み合わせをその分岐ノードのパラメータ(分岐関数)と決定する。

#### 3.2 教師ラベルによるノード追加の判定の導入

より効率よくオンライン学習を行うために、図5に示すように教師ラベルによるノードの追加判定を導入する。あるノードにおいて追加サンプルが範囲外になった場合、ノードに保持したクラス確率と追加サンプルの教師ラベルを比較する。最も高いクラス確率と追加サンプルの教師ラベルが一致している場合にはノードの追加を行わない。逆に、最も高いクラス確率と追加サンプルの教師ラベルが一致していない場合、ノードの追加を行う。

#### 3.3 事前学習の導入

先行研究として、並列分散環境下におけるRandom Forestsの学習法[13]が提案されている。文献[13]の手法では、図6に示すように、データを分割して学習する際に共有データとして事前に学習済みのRandom Forests(事前RF)の情報を利用し、

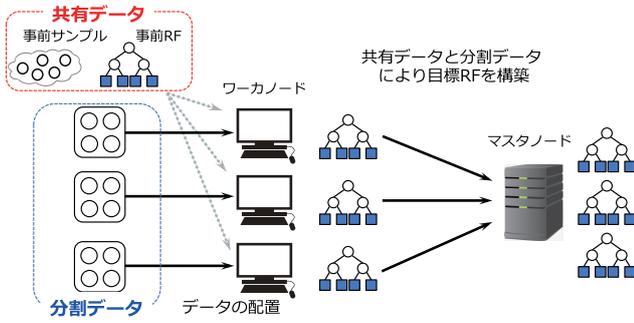


図 6 文献 [13] での学習法

目標となる Random Forests(目標 RF) を構築する。これにより、データの分割による識別精度の低下を抑制し、学習時に事前 RF の情報を用いることの有効性を示している。実運用を想定した場合、並列分散環境を用いてビッグデータを学習した後も、次々と新たなサンプルが収集され続けることが考えられる。収集された新たなサンプルに対応するには、構築した学習器を逐次的・定期的に更新する必要がある。

しかし、先行研究のようにバッチ学習による木構造の再構築を行うと、学習時間が増加するという傾向がある。この時、新たなサンプルへの対応は低コストに効率良く行うことが望まれ、オンライン学習が適している。そこで本研究では、事前 RF を新たに収集されるサンプルに対応させる学習環境を想定し、図 7 に示すように、事前サンプルを用いて学習した事前 RF を用意し、事前 RF に対して木構造の更新を行うことで追加サンプルに対応する。以下に、事前学習の導入したオンライン学習法の流れを述べる。

#### Step1:事前 Random Forests の学習

本手法では、既に得られている事前サンプルを用いて事前 RF を構築する。事前 RF は、前述した並列分散環境等により、オフラインでの学習を行う。この時、事前 RF は各ノードに、到達したサンプルの範囲情報とクラス確率を保持している。

#### Step2:サンプルの範囲情報による判定

Step1 で作成された事前 RF に収集された追加サンプルを入力する。入力された追加サンプルは事前 RF を捜査し、到達した各ノードにおいて、事前 RF が保持するサンプルの範囲外であるか判定を行う。範囲外の場合には、Step3 の教師ラベルを用いたノードの追加判定を行う。範囲内の場合には、到達した各ノードのクラス確率を更新し、学習を終了する。

#### Step3:教師ラベルを用いたノードの追加判定

step2 で範囲外と判定された追加サンプルの教師ラベルと、範囲外となったノードのクラス確率を比較しノードの追加判定を行う。教師ラベルが一致しなかった場合、ノードの追加が必要と判定しノードを追加する。教師ラベルが一致した場合、ノードが保持するクラス範囲のみを更新し、学習を終了する。Step2,Step3 をサンプルが入力される限り繰り返すことで、決定木の更新を行う。

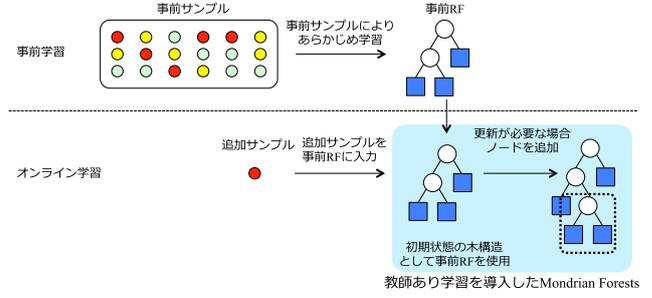


図 7 事前学習の導入

### 3.4 回帰問題への応用

Random Forests は回帰問題に応用可能であり、Regression Forests として提案されている。Mondrian Forests は同様に、回帰問題に応用することが可能である。回帰問題に応用した場合においても、教師あり学習を導入の効果が見られると考えられるため、回帰問題に対する評価を行う。分岐関数の設計には、Regression Forests と同様に、ランダムに選択された複数の分岐関数候補の中から目的関数  $I_j$  を最大とする特徴次元と閾値を選択した分岐関数を選び、サンプルを分岐する。目的関数  $I_j$  は式 (7) として定義される。

$$I_j = \sum_{\mathbf{v} \in S_j} (y - \bar{y}_j)^2 - \sum_{i \in \{L, R\}} \left( \sum_{\mathbf{v} \in S_j^i} (y - \bar{y}_j)^2 \right) \quad (7)$$

ここで  $\mathbf{v}$  は選択された特徴次元を示し、 $y$  は各ノードの出力を  $\bar{y}$  は  $y$  の平均を示している。 $j$  は選択しているノード番号、 $L, R$  はそれぞれ左と右の子ノード番号を示している。これを末端ノードまで繰り返す。各末端ノードは、それぞれ回帰係数を保存する。各末端ノードの回帰係数の設計は問題設定に合わせて様々な設計方法を使用する。最も単純な回帰係数は末端ノードに存在するサンプルのラベルの平均を出力するものである。他には、最小二乗法などの線形回帰がよく利用される。回帰時には、テストサンプルを回帰木へ入力し分岐ノードの分岐関数に従い末端ノードまで分岐させる。末端ノードには学習で得られた回帰係数が保存されており、この回帰係数と入力値を基に連続値の出力  $\hat{Y}_t$  を得る。複数の回帰木の出力  $\hat{Y}_t$  の平均を式 (8) で求め、最終的な出力  $\hat{Y}$  を求める。

$$\hat{Y} = \frac{1}{T} \sum_{t=1}^T \hat{Y}_t \quad (8)$$

ここで、 $T$  は木の本数を表す。

### 4. 評価実験

提案手法の有効性を確認するために評価実験を行う。実験 1 では、教師あり学習の導入による効果の評価のために、学習時間、識別時間、ノード数、識別率の比較を行う。実験 2 では、事前学習の導入による効果の評価のために、事前学習を導入した場合と導入しない場合のノード数と識別率の比較を行う。また、実験 3 では、提案手法を回帰問題に応用した場合の評価として、学習時間、識別時間、ノード数、識別率の比較を行う。

表 1 教師あり学習の導入の評価 (letter recognition)

	学習 [msec]	識別 [msec]	ノード数	識別率 [%]
RF	-	17.6	387	93.71
Mondrian	24.7	16.8	3172.56	89.63
提案手法 A	326.8	18.8	998.33	93.84
提案手法 B	28.1	18.7	1986.58	90.13

表 2 教師あり学習の導入の評価 (MNIST)

	学習 [msec]	識別 [msec]	ノード数	識別率 [%]
RF	-	17.3	455	92.65
Mondrian	26.3	17.5	3237.12	89.62
提案手法 A	184.2	17.6	1183.38	92.11
提案手法 B	28.7	16.9	2104.65	89.98

表 6 回帰問題への応用の評価 (Concrete Compressive Strength)

	学習 [msec]	識別 [msec]	ノード数	回帰誤差
RegF	-	12.7	73	11.64
Mondrian	21.5	14.6	167	19.42
提案手法	123.6	15.2	147	11.91

ここで、提案手法は 2 種類あり、提案手法 A として、教師ラベルを分岐関数とノードの追加判定に導入したもの。提案手法 B として、教師ラベルをノードの判定のみに導入したものがあ。ただし実験 3 では、教師ラベルが連続値であるため、教師ラベルによるノードの追加判定は行っていない。

#### 4.1 データセット

教師あり学習の導入による評価と事前学習の導入による評価には、UCI Machine Learning Repository から letter recognition と、文字認識のベンチマークとして用いられている MNIST を使用する。letter recognition は、A から Z までの手書きの文字認識を行うデータセットであり、20,000 個のサンプルを持ち、クラス数 26、文字画像から抽出した 16 次元の特徴量からなる。MNIST は、0 から 9 までの手書きの文字認識を行うデータセットであり、70,000 個のサンプルをもち、クラス数 10、 $28 \times 28$  の画像からなる。評価サンプルには、学習に使用していないサンプルを用いる。

#### 4.2 実験 1:教師あり学習の導入の評価

実験 1 では、教師あり学習の導入の評価を行う。パラメータは、木の数を 10、特徴次元の選択回数を 5、閾値の選択回数を 10 とする。Random Forests のみ深さを 15 と設定し、その他の手法では木の深さに制限を設定しない。また、サンプルの入力順による影響を考え、計 10 回学習したときの平均値をとる。比較手法として、全サンプルをまとめて学習した Random Forests(RF) と Mondrian Forests(Mondrian)、提案手法 A、提案手法 B を比較する。letter recognition を用いて、学習サンプル数を 6,000 とした場合の結果を表 1 に示す。MNIST を用いて、学習サンプル数を 60,000 とした場合の結果を表 2 に示す。実験結果から、提案手法 A は 1 ノードあたりの学習時間が Mondrian Forests と比較して増加しているが、作成されるモデルのサイズは letter recognition では 70.59%、MNIST では 66.84%削減しており、識別精度はどちらも向上している。

学習時間の増加の要因として、教師ラベルを用いて分岐関数候補の中から最も良かった候補を選択していることが考えられ、分岐関数に教師ラベルを用いない提案手法 B では、Mondrian Forests と同等の学習時間である。

#### 4.3 実験 2:事前学習の導入の評価

実験 2 では、事前 RF を使用した場合の評価を行う。事前 RF のパラメータは、木の数を 10、深さを 10、特徴次元の選択回数を 5、閾値の選択回数を 10 とする。比較手法として、事前学習を導入しない場合と事前学習を導入した場合の Mondrian Forests(Mondrian)、提案手法 A、提案手法 B を比較する。letter recognition を用いて、事前学習なしでサンプル数 12,000 とした場合と事前学習サンプル数を 6,000 とし、追加サンプル数を 6,000 とした場合を比較した結果を表 3 に示し。MNIST を用いて、事前学習なしでサンプル数 60,000 とした場合と事前学習サンプル数を 30,000 とし、追加サンプル数を 30,000 とした場合を比較した結果を表 4 に示す。実験結果から、事前学習を導入することにより、より少ないノード数で木の構築が可能であることが確認できる。この傾向は、事前 RF が事前サンプルをまとめて一括で学習しているため、オンライン学習を繰り返し行った場合よりも小さな木構造で構築可能であること、事前 RF を初期状態の木構造とすることで、ノードの追加の必要性が低下することの 2 点が起因していると考えられる。

#### 4.4 実験 3:回帰問題への応用の評価

実験 3 では、Mondrian Forests を回帰問題へ応用した場合の評価を行う。回帰問題への応用の評価には、UCI Machine Learning Repository から Concrete Compressive Strength と Energy efficiency を使用する。Concrete Compressive Strength は、コンクリートの圧縮強度を回帰するためのデータセットであり、1,030 個のサンプルを持ち、特徴次元数 8 からなる。Energy efficiency は、建物の形状を用いたエネルギー分析を行うためのデータセットであり、768 個のサンプルを持ち、特徴次元数 8、出力として回帰値が 2 種類存在する。評価サンプルには、学習に使用していないサンプルを用いる。決定木のパラメータは、木の数を 10、深さを 10、最小サンプル数を 10、特徴次元の選択回数を 3、閾値の選択回数を 10 とする。比較手法として、Regression Forests(RegF)、回帰問題に応用した Mondrian Forests(Mondrian)、提案手法を比較する。なお、本実験の提案手法ではノードの追加判定を行っていない。Energy efficiency を用いて、学習サンプル数を 800 とした場合の結果を表 5 に示し、Concrete Compressive Strength を用いて、学習サンプル数を 600 とした場合の結果を表 6 に示す。ここで、Energy efficiency の回帰値が 2 種類存在するため、それぞれ回帰誤差 1、回帰誤差 2 とし評価する。実験結果から、提案手法は Mondrian Forests と比較して、回帰精度は向上しつつ、ノード数は削減可能であることがわかる。この傾向は、教師ラベルを用いた分岐関数設計により、各々のノードの分岐精度が向上していることに起因している。このことから、Mondrian Forests を回帰問題に応用した場合においても、教師あり学習の導入の効果が確認できたと考えられる。

表 3 事前学習導入の評価 (letter recognition)

	事前学習なし		事前学習あり		削減率 [%]
	ノード数	識別率 [%]	ノード数	識別率 [%]	
Mondrian	3172.56	89.63	993.81	92.36	68.69
提案手法 A	998.33	93.84	432.13	93.94	56.50
提案手法 B	1986.58	90.13	631.23	92.62	68.22

表 4 事前学習導入の評価 (MNIST)

	事前学習なし		事前学習あり		削減率 [%]
	ノード数	識別率 [%]	ノード数	識別率 [%]	
Mondrian	3237.12	89.62	952.84	92.12	70.59
提案手法 A	1183.38	92.11	406.32	92.58	65.68
提案手法 B	2104.65	89.98	614.79	92.16	70.81

表 5 回帰問題への応用の評価 (Energy efficiency)

	学習 [msec]	識別 [msec]	ノード数	回帰誤差 1	回帰誤差 2
RegF	-	11.8	51	1.94	2.04
Mondrian	19.6	13.2	103	6.39	10.49
提案手法	118.1	12.6	85	2.23	2.43

## 5. ま と め

本研究では、教師あり学習を導入した Mondrian Forests の効率化について提案した。提案手法は、2つのアプローチにより高精度かつ効率的な学習を行っている。1つ目は、教師ラベルを用いた分岐関数の設計の導入である。教師ラベルを用いて分岐関数を設計することで、各ノードでの分類精度を高め、モデルサイズを抑制することができる。2つ目は、教師ラベルを用いたノードの追加判定の導入である。教師ラベルを用いたノードの追加判定を導入することで、学習サンプルと追加サンプル分布に差異が生じる場合においても、ノードに保持しているクラス確率を参照することで、不必要なノードの追加を抑制することが可能である。

また、本研究では、オンライン学習の環境に事前 RF の使用を想定しており、学習済みの事前 RF に追加サンプルを入力した場合においてノードの追加をさらに抑制できる傾向があることを確認した。さらに、提案手法を回帰問題に適応した場合においても、提案手法の有効性を示した。これらのことから、提案手法の Mondrian Forests の効率化を実現したといえる。

今後の課題は、オンライン学習を繰り返し、過成長した決定木構造を修正するための手法を確立することである。

## 文 献

- [1] Y. Freund, and R.E. Schapire, Experiments with a New Boosting Algorithm, International Conference on Machine Learning, Vol. 96, pp. 148–156, 1996.
- [2] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based Learning Applied to Document Recognition, IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.
- [3] J. Deng, W. Dong, R. Socher, L. Li, K. Li, L. and Fei-Fei, Imagenet: A large-scale hierarchical image database., Computer Vision and Pattern Recognition, Computer Vision and Pattern Recognition, pp. 248–255, 2009.
- [4] L. Breiman, Random Forests, Machine Learning, vol. 45, pp. 5–32, 2001.
- [5] L. Breiman, Bagging Predictors, Machine Learning, vol. 24,

- no. 2, pp. 123–140, 1996.
- [6] T.K. Ho, The Random Subspace Method for Constructing Decision Forests, IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 20, no. 8, pp. 832–844, 1998.
- [7] A. Criminisi, J. Shotton and E. Konukoglu, Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning., Foundations and Trends(R) in Computer Graphics and Vision, Vol. 7, no. 2–3, pp. 81–227.
- [8] J. Taylor, J. Shotton, T. Sharp, and A. Fitzgibbon, The vitruvian manifold: Inferring dense correspondences for one-shot human pose estimation, IEEE Computer Vision and Pattern Recognition pp. 103–110, 2012.
- [9] A. Saffari, C. Leistner, J. Santner, M. Godec, and H. Bischof. On-line random forests., IEEE International Conference on Computer Vision Workshops, 2009.
- [10] M. Denil, D. Matheson, and N. de Freitas. Consistency of online random forests., International Conference on Machine Learning, pp. 1256–1264, 2013.
- [11] B. Lakshminarayanan, D. M. Roy, and Y. W. Teh, Mondrian forests: Efficient online random forests., Advances in Neural Information Processing Systems., pp. 3140–3148, 2014.
- [12] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees, Machine learning, vol. 63, no. 1, pp. 3–42, 2006.
- [13] R. Wakayama, R. Murata, A. Kimura, Y. Yamauchi, T. Yamashita and H. Fujiyoshi, Distributed forests for MapReduce-based machine learning, IAPR Asian Conference on Patten Recognition, 2015.