教師あり学習を導入した Mondrian Forests の効率化に関する研究

村田 隆英　　指導教授：藤吉 弘亘

1. はじめに

学習したモデルを実際に運用する場合において，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問題が発生する．そのため，新たなサンプルに合わせて高速に学習モデルを更新する必要がある．Random Forests[1] のフレームワークを用いたオンライン学習法として Mondrian Forests[2] が提案されている．しかし Mondrian Forests は教師なし学習のため，不必要なノードを追加することがある．そこで本研究では，Mondrian Forests に教師あり学習を導入することで効率化を図る．

2. Mondrian Forests

本章では，Mondrian Forests[2] の学習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， 分岐・末端ノードは到達した学習サンプル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たびに実行され，図 1 に示すように，決定木をトラーバーサルし，トラバーサル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．

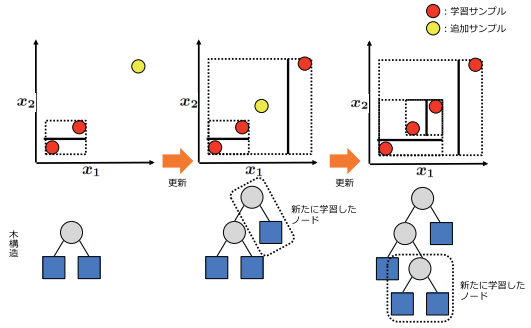

図 1：MondrianForests による学習

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，最初に選択されたものを使用する．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 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 2 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，不要なノードを追加することがあり，木構造が肥大化するという問題がある．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学習法を提案する．

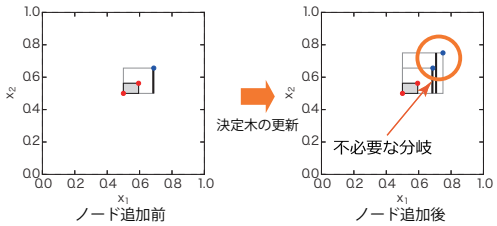

図 2：Mondrian Forests の問題点

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 3 に示す．

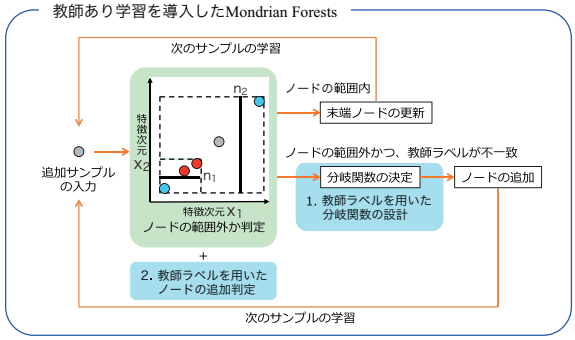

図 3：提案手法の流れ

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ため，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，提案手法では式 (1) によって情報利得 (Information gain)ΔE を求め，分岐関数を評価する．

$$\Delta E = -\frac{|\mathcal{S}_l|}{|\mathcal{S}_n|} E(\mathcal{S}_l) - \frac{|\mathcal{S}_r|}{|\mathcal{S}_n|} E(\mathcal{S}_r) \tag{1}$$

ここで，関数 $E(\mathcal{S})$ 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．

$$E(I) = -\sum_{i=1}^{n} P_i \log P_i \tag{2}$$

P_i はサンプル集合 $\mathcal{S}_n$ に含まれる教師ラベルから得られるカテゴリの分布，n は学習カテゴリの数である．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 (1) 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 2 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．

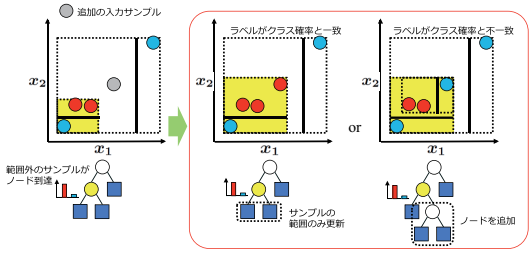

図 4：教師ラベルによるノードの追加判定

3.3 事前学習の導入

並列分散環境下において，共有データとして事前 RF を用いることの有効性が文献 [3] で示されている．本研究は，同様に事前 RF を用いた環境での運用を想定している．そこで，図 5 に示すように，事前学習で学習した事前 RF を用意し，事前 RF に対して木の更新を行う．

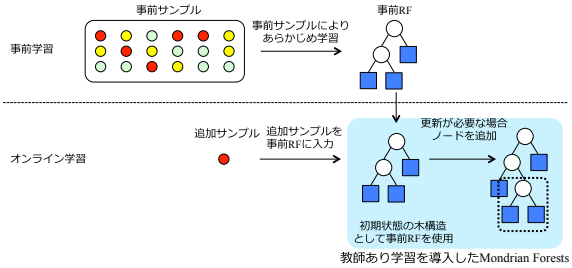

図 5：事前学習の導入

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得られると考えられるため，回帰問題に対する評価を行う．

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．本実験では，全サンプルをまとめて学習した Random Forests (RF) と Mondrian Forests(Mondrian), 提案手法の比較を行う．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る．比較はこれら 4 つの手法について行う．

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, Letter Recognition, MNIST を使用し，事前学習の導入の評価には，Letter Recognition を使用する．回帰問題への応用の評価には，Concrete Compressive Strength を使用する．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．

4.2 実験 1:教師あり学習の導入

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．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．

表 1：Letter Recognition の結果

	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：MNIST の結果

	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	28.1	16.9	2104.65	89.98

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．1 ノードあたりの学習時間は，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ため，約 80%増加している．一方，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

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．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 に示す．

表 3：事前学習導入の評価

	事前学習なし		事前学習あり		
	ノード数	識別率 [%]	ノード数	識別率 [%]	削減率 [%]
Mondrian	3237.12	89.62	993.81	92.12	68.69
提案手法 A	1183.38	92.11	432.13	92.58	57.68
提案手法 B	2104.65	89.98	631.23	92.16	69.08

実験結果から，事前学習を導入することにより，より小さなモデルサイズで木の構築が可能であることが確認できる．この傾向は，事前サンプルにより事前 RF を学習することで，ノードの追加の必要性が低下すること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

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．決定木のパラメータは，木の数を 10，深さを 10，特徴次元の選択回数を 3，閾値の選択回数を 10 とする．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．学習サンプル数を 800 とした場合の結果を表 4 に示す．

表 4：回帰結果

	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

実験結果から，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と考えられる．

5. おわりに

本研究では，教師あり学習を導入した Mondrian Forests について述べた，Mondrian Forests に教師ラベルを用いた分岐関数とノードの追加判定を導入することで，Mondrian Forests と比較してモデルサイズの削減しつつ，より高精度な識別が可能となった．このことから，提案手法の Mondrian Forests の効率化を実現したといえる．今後の課題として，オンライン学習を繰り返し，歪に成長した決定木構造を修正するための手法の確立を予定している．

参考文献

[1] L. Breiman, ”Random Forests. ”, Machine Learning, vol.45, pp.5-32, 2001.

[2] Lakshminarayanan, Balaji, Daniel M. Roy, and Yee Whye Teh. , ”Mondrian forests: Efficient online random forests.” , Advances in Neural Information Processing Systems. , 2014.

[3] Ryoji Wakayama, Ryuei Murata, Akisato Kimura, Yuji Yamauchi, Takayoshi Yamashita and Hironobu Fujiyoshi,“ Distributed forests for MapReduce-based machine learning“ , ACPR, 2015.

研究業績

[1] 村田隆英, 山下隆義, 山内悠嗣, 藤吉弘亘, ”Random Forest を用いた能動学習における有効なサンプル選択“ , CIVM 研究会, 2014.

[2] Ryuei Murata, Yohei Mishina, Yuji Yamauchi, Takayoshi Yamashita and Hironobu Fujiyoshi, “ Efficient Feature Selection Method Using Contribution Ratio by Random Forest ”, FCV, 2015.

（他 学会口頭発表 3 件）

受賞

[1] CVIM2014 卒論セッション 最優秀賞

[2] IEEE ICRA2015 Amazon Picking Challenge Travel Reimbursement Awards

[3] 第 19 回 PRMU アルゴリズムコンテスト 優秀賞