Vol.2017-DBS-165 No.20
Vol.2017-IFAT-128 No.20
2017/9/19

Bag of Condition Chains：決定木をベースとした異常検出のための学習法と名刺の電子化ミス検出への応用

糟谷勇児[†1]

概要：本稿では，決定木をベースとした，データからの異常値検出を行うための学習方法と，その応用について述べる．異常検出においては下記の１．２．のような課題を抱える例が少なくない．
１．異常のサンプルが全体の数パーセント程度と少数であり，サンプルデータを集めるのが困難，あるいは集めた異常のサンプルにバイアスがかかってしまう
２．観測しているデータからでは異常と正常を確率的にしか判断できない
このような際に，ある程度量はあるが多少バイアスがかかっている学習用サンプル１と，少数ではあるがバイアスの少ない学習サンプル２を用いて学習を行う方法を提案する．具体的には学習サンプル１を用いて，パラメータをランダムに変化させた決定木を複数生成し，これらの木から有効な枝を学習サンプル２を用いて抽出し，枝の集合を最終的な判別機とする．このようにすることで，バイアスの影響による過学習を避け，シンプルで容易に実装可能な判別機を学習することが可能である．本稿では本手法を名刺の入力ミス半手に利用する実験を実施し，名刺のクリティカルなミスを 1.5 倍から 2 倍多く検出できることが分かった．

キーワード：決定木，異常値検出，不均衡問題

Bag of Condition Chains : Machine learning method for anomaly detection based on decision trees and application to name card entry mistake detection

Yuji Kasuya[†1]

Abstract: This paper describes a new learning method based on a decision tree for anomaly detection, and the application of this to the detection of entry mistakes for name cards. Anomaly detection tasks sometimes have problems, such as:
1. Anomaly samples are very few (e.g., less than 1%). Because of this, it is difficult to collect samples, and the samples collected tend to be biased.
2. It is impossible to perfectly determine if something is an anomaly or not from observed data, even manually.
To overcome these problems, our method uses two types of sample data for learning. These are (1) data with sufficient quantity, but biased, and (2) data having little bias but also small quantity. Our method has two steps. In step one we use the first type of data to create decision trees randomly with variable parameters such as tree depth, class weight, features to use, and parting methods. In step two we divide all trees into branches and select effective combinations of branches using a greedy algorithm with the second type of data. Our method makes a simple result model that is easy to implement and more effective than the decision tree model. Our experiment reveals that our method found 1.5 to 2 times more name card entry mistakes than normal decision trees.

Keywords: decision tree, anomaly detection, unbalanced classes problem

1. はじめに

近年，データサイエンスの領域で，特徴量から機械学習技術を用いて判別することで，多数のデータからごく少数の異常を検出する技術開発が盛んに行われている．例えば[1]では予防医学において，検診データから機械学習の技術を用いて数パーセント程度の陽性者を抽出し，医師が詳細に診察することで病気を発見するというようなことが行われている．別の例として[2]ではプリペイド式携帯電話の解約を予測するシステムの例が記載されている．この例も解約率は月の解約者は 8%程度と少数の異常をとらえる例であるといえる．このような少数の異常をとらえるような異常検出においては下記の１．２．のような課題を抱える例が少なくない．

１．異常のサンプルが全体の数パーセント程度と少数であり，サンプルデータを集めるのが困難，あるいは集めた異常のサンプルにバイアスがかかってしまう
２．データからでは異常と正常を確率的にしか判断できない

例えば，[1]の例において，実際に病気になるのはごく少数であり，サンプルの収集には数年を要する．また，正常と思われている集団の中にも発見されていないが病気の患者が含まれている．逆に，データ上正常値から乖離していたとしても病気ではないという例もある．

Sansan 株式会社の持つ名刺データの入力ミスの検出にお

†1 Sansan 株式会社　Sansan inc.

Vol.2017-DBS-165 No.20
Vol.2017-IFAT-128 No.20
2017/9/19

いても同様の課題がある．名刺の入力ミスを正しく検出できれば，その名刺を打ち直すことで入力精度をさらに高めたり，分析のデータに使用しないことで分析の精度を高めたりすることが可能である．一方で，入力ミスは全体の 1%以下であることと，入力ルールが多岐にわたるため専門のオペレータでなければミスかどうかの判断が難しく，１．２．の課題を抱えていた．

本稿では，これらの課題に対応した決定木をベースとした，データからの異常検出を行うための学習方法と，その応用について述べる．

ある程度，量はあるが多少バイアスがかかっている学習用サンプル（量重視のサンプル）と，少数ではあるがバイアスの少ない学習サンプル（質重視のサンプル）を用いて学習を行う方法を提案する．具体的には量重視のサンプルを用いて，パラメータをランダムに変化させた決定木を複数生成し，これらの木から有効な枝を質重視のサンプルを用いて選別し，枝の集合を最終的な判別機とする．このようにすることで，バイアスの影響による過学習を避け，シンプルで容易に実装可能な判別機を学習することが可能である．

本稿ではこの手法を名刺入力データに対して適用し，入力ミスを検出する場合の応用を紹介する．

2. 関連研究

本節では，提案手法に関連する関連研究について述べる．異常検出に関しては近年出版された[3]において丁寧に解説されており，機械学習を用いた方法も記載されている．機械学習には様々な手法が存在するが，決定木学習（CART のアルゴリズム[4]など）は連続値とカテゴリ値を同時に扱えることや視覚的に妥当性が判断できることから，データサイエンスの文脈で研究が続けられている．また，Random Forest (RF)[5]，Gradient Boost Decision Tree (GBDT)[6]などのより精度が高いとされる手法も決定木をベースとしているため乗り換えが容易である．なお，筆者が事前に分析を行ったところ，数パーセントを陽性として抽出することで数パーセントの異常を発見するという問題では，あまり RF, GBDT は決定木と差異はなかった．これは実際に判定に寄与している決定木の枝は全体の少数であるため，その枝が出現する木のみが判定に大きく寄与し，多数の木を組み合わせる効果が低いためであると考えられる．

[6]では一定の条件下で学習データに対して最適な決定木を構成する方法が提案されている．また，[7]ではそこに条件を加えて高速化する手法が提案されている．ただし，学習サンプルにバイアスがあるという条件下では最適な決定木を学習したとしても過学習を招く恐れがあった．また，指数オーダーの時間がかかるため大規模なデータには適用できなかった．

3. 提案手法

本章では，提案手法の具体的な内容を記載する．

大枠の考え方としては，パラメータを変えた複数の決定木を学習し，その中から有効な枝のみを残すというものである．ここで枝とはトップのノードからある一つのリーフノードまでのたどり方を条件式の AND 条件の組みとして抽出したものである．

異常検出の問題においては，異常と判定するためのごく一部の枝のみしか実際には必要ないが，そのような有効な枝がどのようなパラメータを用いるとどこに出現するのかは経験のある技術者でも事前に判断することは難しく，長時間のチューニングが必要であった．下記に詳細な手法を記載する．

3.1 使用するデータ

本アルゴリズムは学習に 2 種類のラベル付き学習データを使用する．一つは，量があるが，バイアスがかかっている可能性もあるようなデータ（量重視の学習データ）であり，もう一つは少数だがバイアスの少ないデータ（質重視の学習データ）である．ここでバイアスとは例えば，

- 異常と正常のサンプル比率が実際と異なる
- 正常のサンプルの中に異常なものが混ざっている
- ある特徴量の寄与率が実際より高く，その特徴量を用いると学習データの中では簡単に判定ができてしまう

などであり，このまま使用すると過学習を起こしてしまうようなデータを指す．このような学習データのバイアスは異常検出の分野ではしばしば発生する．なぜなら，異常となるデータが非常に少数であるため，

- そのままのサンプル比率では異常率が１％とすると正常を９９倍集めなければならず現実的ではない
- データを全数検査することはコストの面から困難であり，正常と思われるものの中にも異常なものが混じりこんでしまう
- サンプルを集めるために時間がかかるので一部のサンプルは古くなり現在のものと状況が異なる
- これまでに発見した異常のサンプルは，ある特徴量をキーとして発見したものであることがあり，その場合にその特徴量が実際よりも効果的であるように見えてしまう

などのことが起こるからである．

一方，少数であれば，ある程度バイアスの少ないデータを集めることもできるが，このデータだけでは複雑な学習，例えば深さが１０以上ある決定木の学習などは困難である．決定木はデータの分割を繰り返していくものであり，仮に異常値のデータが半数ずつに分割されていったとすれば，３２個のデータからは５回の分割しかできないことになる．

Vol.2017-DBS-165 No.20
Vol.2017-IFAT-128 No.20
2017/9/19

3.2 アルゴリズムのステップ

本アルゴリズムは下記の２ステップからなる

1．量重視の学習データを用いてランダムに複数の決定木を学習する．

2．質重視の学習データを用いて有効な枝を選別する

3.2.1 ステップ１：量重視の学習データを用いて決定木を学習する

ここでランダムに変えるのは

- クラスの重み（正常を異常と判断してしまう誤りと，異常を正常と判断してしまう誤りのコストの比）
- 木の深さ
- 学習から抜く特徴量（全特徴量を使用するか，どれか一つを選んで使用しないようにする）
- 木の分割方法(情報エントロピーを使用するか Gini 係数を使用するか)

などである．

ランダムフォレストでは，使用するデータを 2/3 程度ランダムに選択したり，N 個の特徴量の中から√N 個をランダムに選択したりするが本稿の手法ではこれは行わない．これらを行うと，学習毎に結果が変わりすぎてしまい，本手法には適さないことが理由である．この方法により，ランダムに数十から１００程度の決定木を得る．図 1 にランダムに複数の木を生成した際のイメージを示す．

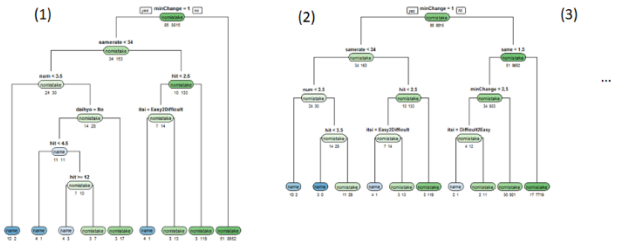

図 1 ランダムなパラメータで学習した決定木の例

3.2.2 ステップ 2：質重視の学習データを用いて有効な枝を選別する

ステップ 1 で得た決定木をばらばらの枝に分解する．枝とは決定木のトップのノードからリーフノードまで一つのたどり方をした条件の組であり，リーフノードの数だけ枝が存在することになる．図 2 に枝への分解のイメージを示す．

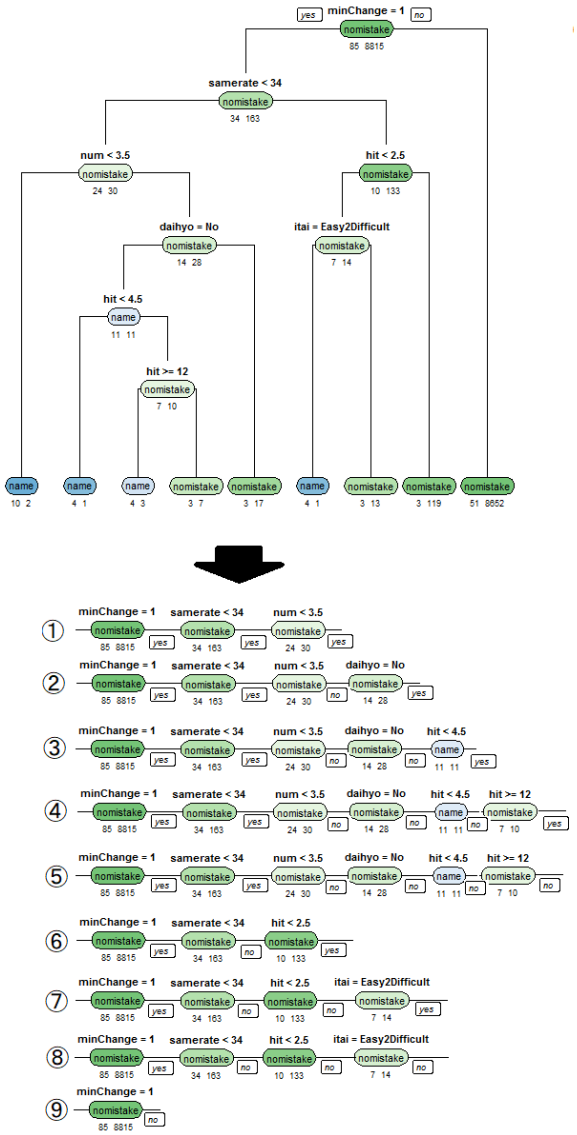

図 2 枝の分解の例

分解された枝から，グリーディーアルゴリズムを用いて質重視の学習データから効果的な枝の組を選定する．すなわち，ある枝を質重視の学習データに適用してみて，下記の式(1)で最も効果的な枝を選ぶ．

$$\underset{k}{\operatorname{argmax}} \frac{true(k)}{positive(k)+1} \qquad (1)$$

ここで k は枝につけられた番号, positive(k)は枝 k によって抽出された陽性となるサンプルの数, true(k)は positive(k)のうち，実際に異常があったサンプルである．ここではスムージングを目的として分母に 1 を加算している．これによってたまたま positive(k)=1, true(k)=1 となるような枝が選ばれてしまいやすくなることを避ける狙いがある．

選んだ枝によって陽性となったサンプルを学習データから取り除き，次の枝を選ぶ，ということを繰り返し，一定の陽性数となるまで枝を選択する．ここではグリーディーアルゴリズムとしたが，実際には 1 万のサンプルから

0.2%にあたる２０サンプルを抽出する問題においては候補となる枝がかなり絞られるので最適解の計算も可能である．図３にグリーディーアルゴリズムで選択された枝の例を示す．

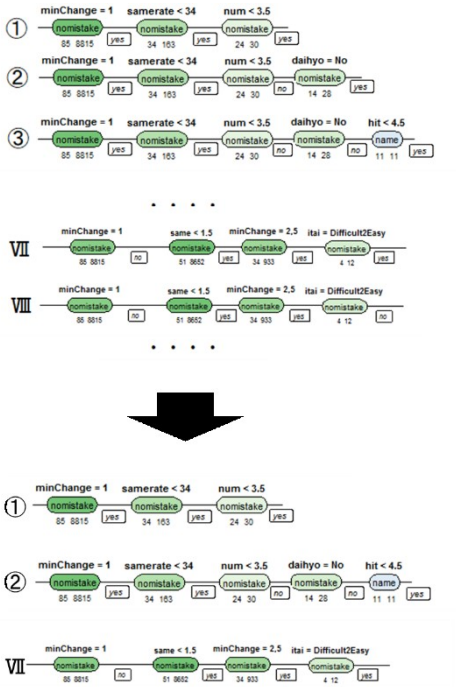

図 3 枝の選別

3.2.3 可視化

本手法では木構造を分解することでかなり単純で可視性が高い表現が可能であり，非技術者がその場で閾値を調整したり、枝を削るなどのことも可能である．一方で，木の表現に慣れていると木の表現でなるべく重複なく見たいという要望もある．そこで得られた枝の各ノードを使用されているデータでソートし（各条件の AND にすぎないため，順序を入れ替えてもよい），同じものを結合していくことである程度は木の形にすることが可能である（図 4）．ソートには最も長い枝に出てくる順（最も長い枝に出てこない特徴量は次に長い枝での出現順）を用いるとよい．ただし，決定木とは異なり，複数の進む先があるような木になる場合もあるし，スタート地点から二つに分かれている可能性もあることは留意されたい．

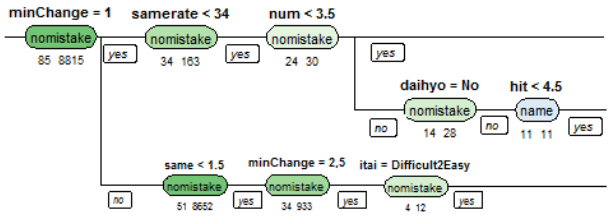

図 4 木構造の例

3.3 過学習の判定

ランダムに生成した決定木から，グリーディーに選択していることで，ここでも過学習が起こりうる．すなわち，量重視の学習データでも質重視の学習データでも効果的であったが，実際には役に立たない枝が選ばれてしまう可能性がある．そこで，ラベルなしのサンプルに対して選ばれた枝の組を適用し，２．の学習時の陽性数とどのくらい差があるかで過学習しているかどうかをある程度判定できる．例えば，３０の決定木から陽性数５０になるまで枝を選び，ラベルなしのサンプルに対して適用したところ，６０個の陽性のサンプルが得られたとする。また１００の決定木から陽性数５０になるまで枝を選び，ラベルなしのサンプルに対して適用したところ，２００個の陽性のサンプルが得られたとする。この場合１００の決定木をもとにした結果は過学習している可能性が高く，ステップ１の決定木の数を減らしたほうが良いと考えられる．

4. 実験と結果

4.1 実験方法

4.1.1 使用するデータ

実験では実際の名刺のミスを検出する問題に対して本手法を適用する．特に今回は住所部分の入力ミスを検出する問題について適用した．量重視のデータとしては，11615 の正常とラベルの付いた名刺のデータ，1338 のミスとラベルの付いたデータを用いる．一方，質重視のデータとしては，4137 の正常とラベルの付いたデータ，27 のミスとラベルの付いたデータを用いた．

量重視のサンプルのうち，正常とラベルがついたものは定期的に抜き取りチェックをしてきた中で正常と判断されたものであり，実際に正常である確率は高いと考えられるが，人がチェックしている以上，一部不正確なデータが混ざっている．

一方，ミスのラベルがついたサンプルは，上記抜き取りチェックで異常と判定されたものと，これまでのミス検出システム（決定木ベース）で見つけられたミスであり，バイアスがかかっていることが推測される．

サンプル比率が正常：異常で 9：1 となっているが実際には 99：1 以下　であり，実際のサンプルとは乖離があるが，正常なサンプルを今の 10 倍集めることは非常に難しい．一方，異常のサンプルは 1024 以上あるため決定木で 2 分割ずつされて行っても，10 段階の木でも異常のサンプルが残る計算になる．

質重視のデータとしては，比率が 995:5 程度と実際にある程度近く，異常のサンプルは抜き取り試験によって得たもののみを使用したため，バイアスも少ない．一方で異常のサンプル数は 32 とデータがないため、これだけで決定木を構成すると高々5 段階でこれ以上分割できなくなってしまう．

Vol.2017-DBS-165 No.20
Vol.2017-IFAT-128 No.20
2017/9/19

量重視のサンプルと質重視のサンプルのほかに 10000 件のラベルなしのデータを用意し評価を行った.

4.1.2 ミス検出手法

Sansan 株式会社では毎月数千万枚の名刺を電子化しており，その中では同じ人の名刺や同じ住所の名刺を複数回電子化している可能性が高い．そこで，ミス検出の方法として，過去に正しく打たれている名刺があると考え，email や氏名と会社名などで同一人物の名刺と考えられる名刺群を抽出し，判定対象の名刺の電子化結果と比較することで判定を行う．具体的には下記(1)-(10)のような特徴量から判定を行う.

(1) 同一人物と思われる名刺の枚数
(2) 検証対象の文字列（今回は住所）の文字数
(3) (1)の中に同一の文字列があった回数
(4) (3)の割合
(5) (1)の中に空白があった割合
(6) (1)の中に何種類の氏名があるか
(7) (1)のなかで最も似た文字列と何文字異なっているか
(8) (7)は文字列の追加/文字列の削除/削除と追加両方のどれに相当するか
(9) OCR をかけたときに(7)の文字列は OCR 中に現れるか
(10) (7)は住所部かビル部かまたはその両方か

ステップ 1 の学習において表 1 のようにパラメータをランダムに変え，学習を行った.

表 1 パラメータとその変更範囲

パラメーター	ランダムに変える範囲
木の深さ	5〜15
抜く特徴量	すべて使用(30%)、どれか一つ抜く
分割指標の指標	gini係数、情報エントロピー
正常クラス重み	1-200
ミスクラスの重み	1-200

ステップ 1 では過学習の可能性を考え，30 の木を作った場合と 100 の木を作った場合をそれぞれ作成した.

ステップ 2 では質重視のサンプルデータを用いて，木から目標とする陽性数を 5, 7, 10, 15, 20, 25 と変えて枝の抽出を行った．時間の測定は ThinkPadT470s Intel Core-i7 2.8GHz, メモリ 24GB OS:Windows10 の環境で行った. プログラムは C#で作成し，特段の高速化は用いていない．表 2,3 に目標陽性数と，未知のサンプルでの陽性数，実行時間の関係をそれぞれ示す.

表 2 30 の決定木のステップ 2 において目標とした陽性数と未知のサンプルでの陽性数（実行時間）

目標陽性数	未知のサンプルでの陽性数	実行時間
5	13	12.9s
7	11	10.7s
10	15	18.1s
15	37	25.6s
20	42	30.5s
25	51	32.9s

表 3 100 の決定木のステップ 2 において目標とした陽性数と未知のサンプルでの陽性数（実行時間）

目標陽性数	未知のサンプルでの陽性数	実行時間
5	20	24.8s
7	31	30.6s
10	39	45.3s
15	87	68.2s
20	94	84.1s
25	96	91.2s

3.3 に示す過学習の判定方法の考え方に従い 100 の木で作った場合は，目標とする陽性数に対して 4 倍程度未知のサンプルで陽性となるサンプルがあり，30 の場合は 2 倍以内に収まっていたため 100 の場合は過学習していると考え以下の実験では 30 の木から抽出した枝を用いた.

表 4 が提案手法において，それぞれの陽性数でどの程度ミスが検出できたかを示す結果である．表 5 は筆者が量重視のサンプルを学習データとし、質重視のサンプルを評価データとして最適になるようにチューニングした決定木の検出結果である．ここで，多数文字ミスとは住所自体が入力されていない，または住所のビル部が抜けてしまうなどの致命的なもので重要なミスである．一文字ミスは文字を似た文字と間違えてしまうようなものである．結果として，抽出したものの中に提案手法のほうがより多く多数文字のミスを発見することができている．一方で，合計検出数では決定木が陽性数 50 で上回った．これは質重視のサンプルにおいてはハイフンや空白などのなくても問題ないような文字を許容としてミス扱いにしていないものが存在するため，それを検出できる枝が優先されなかったために起こったものと考えられる.

表 4 住所ミスに関する提案手法の陽性数と検出数

陽性数	11	21	36	51
多数文字ミス検出数	2	4	4	4
一文字ミス検出数	1	2	3	3

表 5　住所ミスに関する決定木の陽性数と検出数

陽性数	24	35	52
多数文字ミス検出数	2	2	2
一文字ミス検出数	1	2	6

同様に会社名のミスについても決定木を 30 作成する場合で実験を行い，提案手法がより多くクリティカルなミスを検出できていることが分かった．なお，会社名の場合ミスにおいてクリティカルなミスは文字数だけでは判断しにくいため,「もともとの会社名と全く異なる記載になってしまっていて，記載だけからは元の会社名を想起できないようなもの」と定義とした．表 6 に提案手法の陽性数と検出数の関係を，表 7 に決定木による陽性数と検出数の関係を示す。

表 6　会社名ミスに関する提案手法の陽性数と検出数

陽性数	26	57	76
クリティカルなミス	1	2	3
軽微なミス	8	13	17

表 7　会社名ミスに関する決定木の陽性数と検出数

陽性数	24	54	76
クリティカルなミス	1	2	2
軽微なミス	7	10	11

4.2 考察

今回実験したケースではクリティカルなミスを 1.5 倍から 2 倍検出できていた.

なお, 4.1 で比較のために用いた決定木は筆者が時間をかけてチューニングしたものであり，それで抽出できないようなミスを抽出できるモデルが，あまりチューニングが必要ないランダムな手法で得られることは興味深い．また本手法は，新しいサンプルが得られたり，データの組成が変わったりしたときなども再学習や追加学習が容易であるという利点もあり運用コスト面の向上も期待できる. 一方で、2 種類のサンプルを用意するのはコストが高いのではという考えもある。しかし、機械学習においては学習に使うデータセットと評価用のデータセットを分けて用いることも一般的に行われており、評価用のデータセットをうまく活用すれば同様のことが可能であると考える。

5. おわりに

本稿では量重視のサンプルで学習した決定木から，質重視のサンプルを用いて，必要な枝のみを取り出して組み合わせることで，シンプルで効果的なモデルを構成できることを提案した. 本手法を用いることで一般的な決定木の 1.5 倍から 2 倍のクリティカルなミスを検出することができた. 本手法により作成されたモデルは人が見てもわかりやすく，メンテナンス性に富み，if 文だけでどんな言語でも実装可能である.

一方で，まだ評価や実装について粗削りな面も多く，今後，一般的なベンチマークでの他手法との比較などを交えて詳細に評価をするとともに，ヒューリスティックもとに決定している定数などをどのように選ぶかなど検討していきたい.

謝辞　本稿手法作成に当たり，日々目視チェックで名刺電子化ミスを発見し，正解データ作成に協力いただいた，菊田ゆう子さん，森香澄さん，仲西那納さん，大岩智洋さんに敬意を表する.

参考文献

[1] 青木空眞, 佐藤憲一, 星憲司, 川上準子, 鈴木祥子, 森弘毅, 佐藤研, 中川吉則, 志村浩己, 齋藤芳彦, 吉田克己. 複数の基本的検査を組み合わせて甲状腺機能異常を発見する診断支援ツールの改良 ―心拍数と服薬補正を加えた予測モデルおよび時系列変化解析の有用性―. 人間ドック 2012 vol.27 no.1, p.87-96

[2] Huang Y,, Zhu F., Yuan M., Deng K., Li Y., Ni B., Dai W., Yang Q. and Zeng J..Telco Churn Prediction with Big Data. proc. of ACM SIGMOD conf. on Management of Data 2015

[3] 井手剛, 杉山将. 異常検知と変化検知. 講談社 2015

[4] Breiman,L. Friedman,J.H. Olshen,R.A. and Stone,C.J. Classification and Regression Trees, Wadsworth.inc 1984

[5] Jerome, H. F. Greedy function approximation: A gradient boosting machine. Ann. Statist 2001, vol.29, no.5, p.1189-1232

[6] Breiman, L. Random Forests. Machine Learning 2001, vol.45, issue.1, p.5-32

[7] S. Nijssen and E. Fromont. Mining optimal decision trees from itemset lattices. proc. of ACM SIGKDD conf. on KDDM 2007, p. 530–539.

[8] 長部和仁,宇野毅明,有村博紀. 最適な順序付き決定木の高速発見とその文書分類への応用. DEIM 2017 F7-4