平成 26 年度

修士学位論文

機械学習におけるランダム性を持つ特徴抽出法のテキストデータへの応用

Feature Extraction with Randomness for an Application to Machine Learning from Text Data

1175087　　藤森　夏輝

指導教員　　吉田　真一

高知工科大学大学院 工学研究科 基盤工学専攻

情報システム工学コース

要 旨

機械学習におけるランダム性を持つ特徴抽出法のテキストデータへの応用

藤森　夏輝

テキストデータが機械学習アルゴリズムで処理できるよう単語出現頻度などをもとに数値化された単語ベクトルは，本来は比例尺度のような定量的指標ではなく，順序尺度のような定性的な指標である．データマイニングに用いられる手法は特徴ベクトルの内積や距離が定義できる量的データの分類に効果のある数値計算で行うものが多いが，決定木を用いる Random Forest は，説明変数をランダムに選ぶことで様々な形の決定木群を構築するため，定性的なテキストデータに適した機械学習手法であると考える．説明変数をランダムに選ぶとき，従来は疑似乱数列を用いたランダムサンプリングが行われる．しかし，構築する決定木の深さの最大数，分類に用いる決定木の数，決定木の説明変数となる特徴の数が小さいと，生成される疑似乱数に偏りが発生してしまうことがある．この問題に対し，本研究では準乱数列生成器を適用する．準乱数列は一様の点列で構成されている．準乱数列を Random Forest へ適用することによって，木の深さの最大数，識別に用いる木の数，木の構築に用いる特徴の最大数がそれぞれ低く設定されても，ランダムサンプリングにおいて生成される数列が偏る可能性はなくなり，疑似乱数列を適用した同アルゴリズムが困難としていた，同条件下における高精度識別が可能となると考える．独自に収集した日本語 SPAM メールデータセット（データ数 : SPAM600，非 SPAM1000）を用いて，疑似乱数列と準乱数列を適用した場合において実験を行う．日本語 SPAM メールは，これまでの研究でサポートベクターマシンやニューラルネットワークに比較して，Random Forest が安定して高い識別精度となるという報告があり，決定木をベースとするためテキストデータとの相性が良いこ

とが考えられるため，識別精度比較の実験に用いる．その結果，木の深さが 2，構築に利用する特徴の最大数と識別に用いる木の数が 10 以下の条件の下で，変更後の識別率が 2〜3% 向上することを示す.

キーワード　Random Forest，疑似乱数列，Mersenne Twister，Low-Discrepancy 列，準乱数列

Abstract

Feature Extraction with Randomness for an Application to Machine Learning from Text Data

Natsuki Fujimori

In text processing, a word vector, which is converted from text document data, is usually used as a feature vector. A word vector is a histogram of word frequency or occurrence of a document. It is a numerical data, however it is not a quantitative data such as ratio scale. It is originally a qualitative data such as ordinal or nominal scale. Some methods for data mining using machine learning employ numeric calculation which can discriminate quantitative data that able to define the distance or inner product of feature vectors, but random forests employing decision trees is machine learning technique which is suitable for qualitative text data because it constructs various shapes of decision trees by choosing predictor values randomly. When choosing predictor values, the random sampling is performed using pseudo-random sequence. However, if the number of tree depth to construct, the number of decision trees to use discriminate, or the number of features predictor values of decision trees are small values, the bias may appear because of non-uniformness of pseudo-random numbers. In order to solve this problem, in this study, we propose an application of quasi-random number generator. quasi-random sequence generates uniform sequence of points. When the number of tree of depth, the number of trees using for discrimination, or the number of features constructing tree are small, the proposed method is able to achieve higher performance than pseudo-random. By using the original Japanese SPAM e-mail dataset (600: SPAM,

1000: non-SPAM), we perform the experiment of conventional and proposed methods. As a result, under the condition that the maximum number of tree is 2, and that the number of feature is under 10, the proposed method improves 2 or 3 percent of the precision.

key words Random Forest, Pseudo-Random Sequence, Mersenne Twister, Low-Discrepancy Sequence, Quasi-Random Sequence

目次

図目次

表目次

第 1 章

序論

テキストデータの分類は，文書分類の研究として 1970 年代より様々な研究が行われている．また，インターネットの普及に伴って電子メールや Web ページをはじめ，個人の利用者が自由に情報発信できるようになり，多くのテキストデータが公開されるようになり，その中には価値の低い情報も多い．このようなテキストデータに対して，テキストデータの分類を自動的に行うことで価値の低い情報を表示しないようにしたり，重要な情報のみを抽出するテキストマイニングの重要性が高まっている．さらに，個人が Web 上で簡単に情報を発信できる Weblog（ブログ）や短文投稿サイト Twitter の流行により，こうしたテキストマイニングの需要は今後ますます増加すると考えられる．

このようなテキストマイニングに機械学習アルゴリズムを用いる研究がある．機械学習を用いてテキストマイニングを行う際，テキストデータは機械学習アルゴリズムが処理できるよう数値データである単語ベクトルに変換される．単語ベクトルはテキストデータ中における単語の出現頻度などの数値データであり，本来は比例尺度のような定量的な量ではなく，順序尺度あるいは名義尺度のような定性的な指標である．しかし，現在よく用いられるサポートベクターマシンは量的データの分類に効果のある数値計算によってデータを分類するものが多い．これに対して，決定木を用いる手法は原理的に定性的なデータに向いており，また集団学習と組み合わせた Random Forest などは精度も高い． Random Forest は，説明変数をランダムに選ぶことで様々な形の決定木群を作成し，決定木は各特徴量がしきい値を超えるか否かで条件分けをする方法のため，定性的な指標にも親和性が高い．

決定木学習はそのモデルが単純であり，現在では古典的手法のため，分類・回帰問題にそのままの形で利用されることはあまりない．しかし，複数の決定木学習の結果を組み合わせ

てより精度の高い学習器を構築する集団学習の内部で用いられている．集団学習において決定木を構築するのに必要な特徴の選択に関し，ブートストラップから得られるすべての特徴を使って木を構成する手法が Bagging，ランダムサンプリングによって訓練に用いる特徴を選択する手法が Random Forest である．Random Forest のようなランダム性を持つ特徴抽出法は，そのランダム性を維持するために計算機により生成されるランダムな数列を利用している．

計算機で用いられるランダムな数列には疑似乱数が使われているが，一部に数の偏りが発生するという問題が挙げられる．この問題を解決する数列に，準乱数（準モンテカルロ数列または低食い違い量列（Low Discrepancy Sequence））がある．これは，数列の点の選び方が，なるべく既に出現した点と異なるように選択されるという特徴を有している．

そこで本研究では，この二つのランダム性を持つ数列に着目し，疑似乱数を用いてランダムサンプリングを行う Random Forest において，特徴の選択の部分を準乱数列へ変更する．生成される弱学習器のモデルが小さいときに，準乱数列がその条件下で識別性能の向上を図ることが可能であることを確認する．疑似乱数を用いる場合，弱学習器の木の深さの最大数や選択する特徴の最大数，識別に用いる木の総数が低い場合，生成される疑似乱数列の一様性が保証されていないため，生成される学習器の説明変数が似ると考える．したがって，様々な弱学習器の多数決により高精度識別を実現する Random Forest の強みが低下する．一様性の高い点列を生成する準乱数列を用いることにより，この問題を解消し，学習器の生成に関するパラメータが少ない場合において，疑似乱数列よりも高精度な識別が可能となると考える．

本研究では，これまでの研究 [1],[2] で識別精度の向上を試みてきた，日本語の電子メールにおける SPAM・非 SPAM メールの識別を Random Forest にて行う．日本語 SPAM メールの識別は，英語に比較して識別精度が低く，文献 [1],[2] などでは，Random Forest が最も精度が高かった．そこで，日本語 SPAM メールの識別を題材に，Random Forest の乱数を準乱数列に置き換える影響を調べる．

本論文の構成として，第 2 章で決定木学習の概要と，それに付随して集団学習の概要およ

び集団学習に属する学習アルゴリズムの説明を行う．第 3 章では，第 2 章で説明した学習アルゴリズムの一つである Random Forest について，ブートストラップから決定木の構成に用いる特徴の選択法で用いられる疑似乱数列と，準乱数列について説明する．第 4 章で Random Forest に利用する乱数関数を準乱数に変更する理由を説明する．第 5 章で，数値実験により特定条件下において準乱数列で弱学習器の説明変数を選択する指標にすることが，疑似乱数列を選択の指標とする場合よりも高精度識別が可能であることを示す．最後に第 6 章では，本研究のまとめと今後の課題について述べる．

第 2 章

決定木に基づく乱数を用いる集団学習

本章では，まずデータマイニング手法のひとつである決定木学習について説明する．次に，決定木を学習器とする手法である集団学習について説明する．そして，集団学習において最初に考案されたアルゴリズムである Bagging と，そこから派生した Random Forest について説明する．

2.1 決定木学習

テキストマイニングで扱うデータは非数値データであり，分類されるクラスのラベルとなる名義尺度や，順序関係を示す順序尺度などの定性的なデータである．一般のパターン認識で用いられる数値データには，一定の単位で測定され等間隔性を有する間隔尺度や，原点に対してどの程度増減したかなどの，量間における比例関係を示す比例尺度など定量的なデータが存在する．文書データを分類器で分類する際，名義尺度や順序尺度は，単語の出現頻度や評価値の出現頻度のような比例尺度に変換されることが多い．

決定木学習における分類器は木構造であり，テキストマイニングに利用される際に多くの場合は 2 値の文書特徴を用いることから，木は図 2.1 のような二分木構造となる [4]．図 2.1 のように，単純な識別規則を組み合わせて複雑な識別境界を得る手法が決定木学習であり，データマイニングにおける決定木は葉ノードがそのデータの分類，枝はその分類に至るまでの特徴集合を表す．決定木における分類は，根を起点に，入力データが満たす条件へ 1 つず

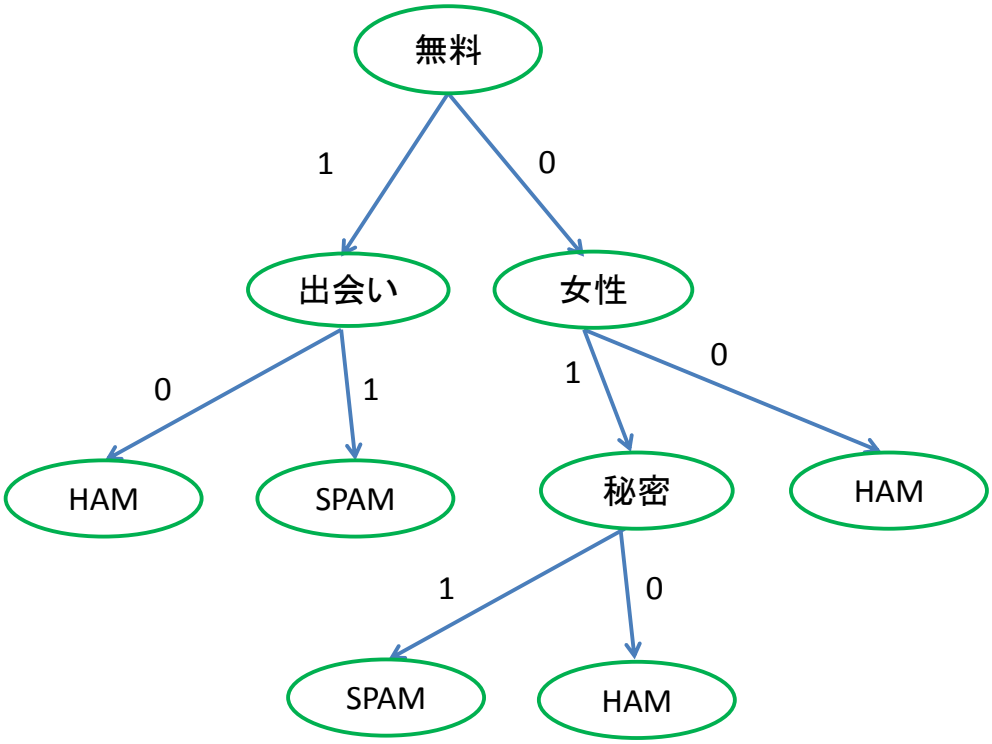

図 2.1　二分木構造をもつ決定木

つ進み，葉に対応するクラスに分類される．一般的に決定木は再帰的に構築される．元となる学習データから特徴 f を一つ選び，その集合を f を含む集合と含まない集合の 2 つに分割する．ここで，選択される特徴は，情報利得やエントロピーなどの指標に基づいて選択される．この特徴選択および学習データ分割を進めることで，決定木が構築される．

決定木学習は設定するパラメータにより様々な性能を有するが，単体の学習アルゴリズムとしては他のアルゴリズムに性能面でしばしば劣る．しかし，決定木学習を他の学習アルゴリズムと比較する際の基準としたり，決定木学習を組み合わせて性能の向上をはかる学習手法も存在する．

2.2 集団学習

集団学習は，精度の高いといえない学習結果を多数組み合わせ，精度の向上を図る．組み合わせの具体的方法は，分類問題には多数決を，回帰問題には平均を取るという手法で精度の向上が図られている．集団学習では，異なる重みやサンプルから，先に述べた決定木のような単純なモデル（弱学習器と呼ばれる）を複数作成し，その結果を組み合わせている．

2.2.1 決定木の生成

集団学習の弱学習器における決定木の生成は，訓練用の学習データに対しブートストラップサンプリングを用いて複数のブートストラップを生成し，生成したブートストラップの数だけ決定木を構築する．ブートストラップサンプリングとは，サンプル集合 $\{x_i \mid 1 \leq i \leq N\}$ から重複を許したサンプリングを行い，新たなサンプル集合 X' を構築する手法である．

2.2.2 特徴選択

ブートストラップサンプリングにより生成したブートストラップを用いて決定木を生成する際，決定木の分岐ノードに用いる特徴を選択する必要がある．集団学習において決定木を構成する特徴選択の方法は 2 種類あり，ブートストラップに含まれるすべての特徴を分岐ノードに用いて決定木を生成するアルゴリズムが Bagging である．また，各ブートストラップに含まれる特徴を乱数列を用いてランダムサンプリングし，決定木を構成するアルゴリズムは Random Forest である．以降の節でそれぞれのアルゴリズムについて説明する．

2.2.3 Bagging

Bagging は，Leo Breiman が 1996 年に提案した集団学習の手法である [6]．Bagging という名称は Bootstrap Aggregating の文字列からの造語である．先に述べたように，Bagging は ブートストラップと呼ばれる複数の学習データを，与えられた元のデータセッ

トからブートストラップサンプリングにより作成し，それぞれのブートストラップに含まれる特徴を全て用いて弱学習器 T_1～T_m を構成し，入力データに対し各学習器で分類を行い，それぞれの出力結果 $f_1(x)$～$f_m(x)$ の多数決で分類を行う．図 2.2 は Bagging におけるアルゴリズムの概念を図示したものである．

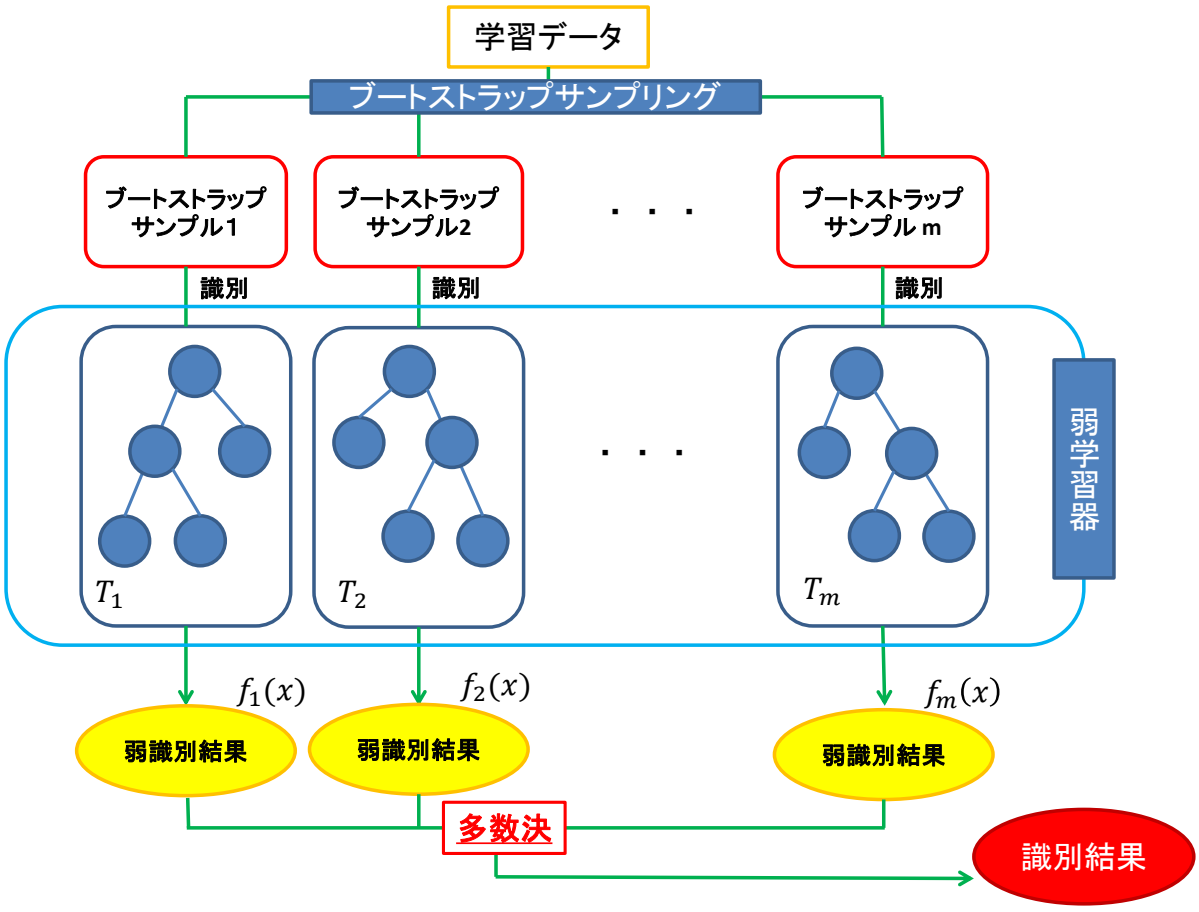

図 2.2　Bagging における処理の流れ

各ブートストラップから生成される弱学習器は独立して学習・識別処理を行えるため，並列処理が可能である．しかし，ブートストラップ内の特徴を全て用いて決定木を構成しているため，ブートストラップにおける特徴のばらつきは各弱学習器の出力結果にも影響を与える．学習データからのブートストラップサンプリングは重複を許した復元抽出法であるため，ブートストラップに含まれる特徴は他のそれに類似することがあり，その場合，各弱学

習器の出力結果の組み合わせによる学習精度の向上は期待できない．この問題を解決したアルゴリズムを実装した手法が次節で述べる Random Forest である．

2.2.4 Random Forest

Random Forest は，バギングを提案した Leo Breiman によって 2001 年に提案された集団学習手法である [7]．Bagging がブートストラップに含まれる特徴を全て用いて弱学習器である決定木を構成していたのに対し，Random Forest ではブートストラップに含まれる特徴を，疑似乱数を用いたランダムサンプリングにより任意の数だけ選択し，それを用いて決定木を構成する．選択する特徴の数を d 個としたとき，開発者である Breiman は利用する学習データにおける総特徴数の正の平方根を取った数にすることを推奨している [5]．特徴の選択にランダムサンプリングを用いることによって，Bagging で問題になっていた各弱学習器の出力結果同士の相関を下げることが可能となり，多様な特徴を持つ決定木の森（＝ランダムなフォレスト）の構築が可能となった．図 2.3 は Random Forest におけるアルゴリズムの概念を図示したものである．

Bagging を改良した Random Forest は，以下の長所を有する [5].

- 分類精度が高い.
- 高次元の入力データに対して頑健性がある.
- 分類に用いられる変数の重要度の推定が可能である.
- 各決定木が選択された特徴数に応じた大きさとなるため，計算コストを抑えられる.

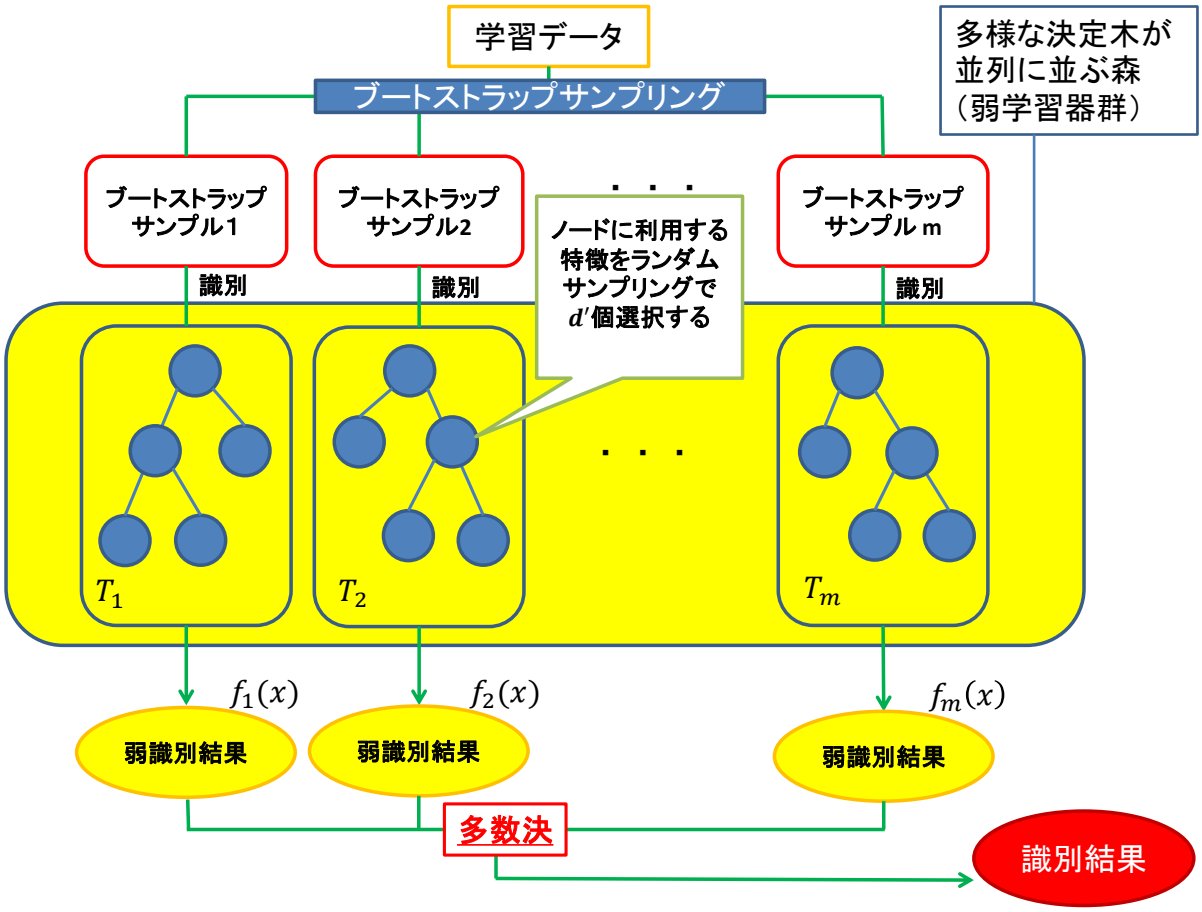

図 2.3　Random Forest における処理の流れ

第 3 章

計算機で生成するランダム性を持つ数列

本章では，計算機のアルゴリズムで用いられる疑似乱数について説明する．従来より．計算機シミュレーションをはじめ様々な計算機アルゴリズムで乱数が用いられてきた．ランダムにサンプリングを行い近似的に数値積分や最適化を行うモンテカルロ法は乱数を用いるアルゴリズムとして代表的なものである．その他，ニューラルネットワークの重みや非階層的クラスタリングの k-means 法の所属クラスタの初期値などに乱数を用いる例や，近年では遺伝的アルゴリズムなどの進化計算法の個体生成，突然変異，交叉選択，選択などにも用いられており，乱数の生成はアルゴリズムの計算の結果にも影響を及ぼす．本論文では，ブートストラップサンプリングや特徴選択に乱数を用いる機械学習アルゴリズムである， Random Forest の乱数について研究を行う．

3.1 疑似乱数列

疑似乱数（pseudorandom sequence）とは，乱数表や再現性のない物理乱数などを用いず，決定的な算法により，計算機で生成する乱数列に見える数列である [9].

計算機での疑似乱数として古くから使われている手法は線形合同法である．これは，漸化式

$$X_{n+1} = (A \times X_n + B) \bmod M \tag{3.1}$$

を計算する方法で，C 言語の標準ライブラリなど広く用いられている．多次元の点のプロッ

トなどでは点の分布が一様でないことや，下位のビットのランダム性が低いなどの問題がある。条件が整えば，周期は最大で M となる．最大周期 M を保障する M 系列を用いる線形帰還レジスタもスペクトル拡散などで用いられる．近年，疑似乱数列生成法として広く一般的に使用されている手法に，Mersenne Twister 法がある．これは，松本，西本らによって 1998 年に発表された疑似乱数生成アルゴリズム [8] であり，それまで疑似乱数列生成法として使われていた線形合同法における周期の短さや下位ビットのランダム性の低さ，M 系列における出力ビットの不十分なランダム性といった短所を持たない．周期は $2^{19937}-1$ で，623 次元で均等に分布する．このことから，Mersenne Twister は疑似乱数列生成法として高い評価を得ている [10]．図 3.1 および 3.2 は，Mersenne Twister 法において，前者が 1000*1000 の疑似乱数を 2 次元に散布したものを表し，後者が 10000*10000 の散布図を表す．その他，2003 年には Marsaglia により排他的論理和を用いた Xorshift 法が提案されており，周期は $2^{128}-1$ で線形合同法より長周期であることが示されている [12]．また，排他的論理和（XOR）とビットシフトのみの演算のため，高速であるという特徴も持つ．scikit-learn では決定木の生成に Xorshift を用いている．

3.2　準乱数列

準乱数（Quasi random numbers）とは，準モンテカルロ法（Quasi-Monte Carlo method）で用いられている数列である．$I := \{0 \leq x < 1\}$ 上において一様な乱数を一様乱数と言い，I^n に一様分布する独立な N 個のベクトル乱数 $x_i\,(1 \leq i \leq N)$ を発生し，$S\,(f)$[*1]を

$$S_N\,(f) = \frac{1}{N} \sum_{1 \leq i \leq N} f\,(x_i) \tag{3.2}$$

で近似するのがモンテカルロ法における数値積分である．この式について，点集合 $\{x_i \mid 1 \leq i \leq N\}$ をうまく選び誤差を減らすのが準モンテカルロ法である．各 n に対し，I^n の無限点列，最初の N 項を P_N としたとき $D_*\,(P_N) = O\left(N^{-1}\,(\log N)^n\right)$[*2]を満たすもの

[*1] 関数 f の定積分値

[*2] D_* は，モンテカルロ積分における近似誤差の上限を表す.

が知られており，n 次元低食い違い量列（Low Discrepancy Sequence）とも呼ばれている [9]．これを疑似乱数の代わりに用いるものが準乱数（quasirandom）である．準乱数の 1 つの SOBOL 法を本研究では用いる．SOBOL 法とは，1967 年に I.M. Sobol によって提案された準乱数列生成アルゴリズムである [11]．図 3.3 は SOBOL 法で生成した 1000*1000 の低食い違い量列を 2 次元に散布したものを表し，図 3.4 は同じく SOBOL 法で生成した 10000*10000 の低食い違い量列の 2 次元散布図を表す.

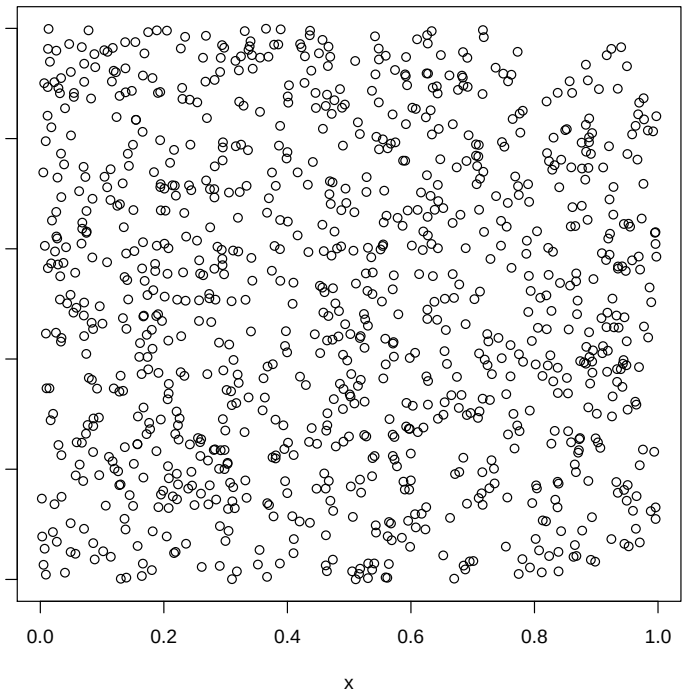

図 3.1　Mersenne Twister 法による 1000*1000 の疑似乱数散布図

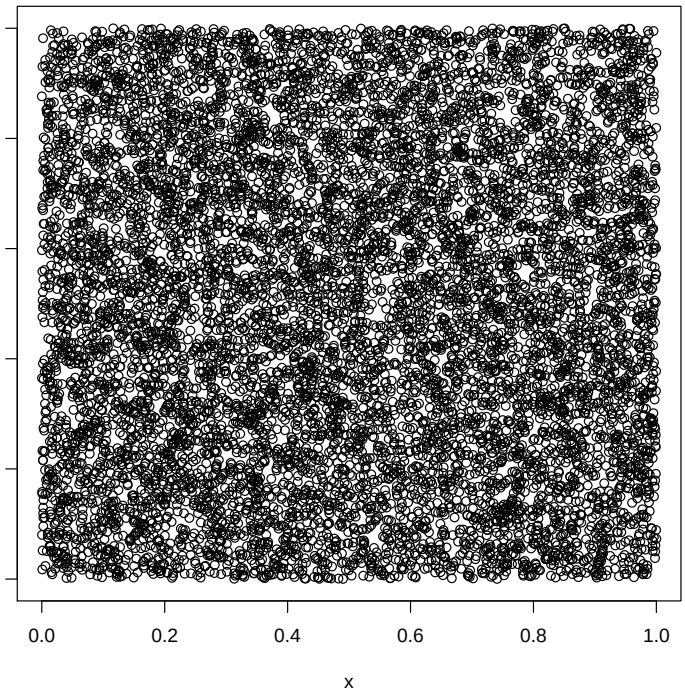

図 3.2　Mersenne Twister 法による 10000*10000 の疑似乱数散布図

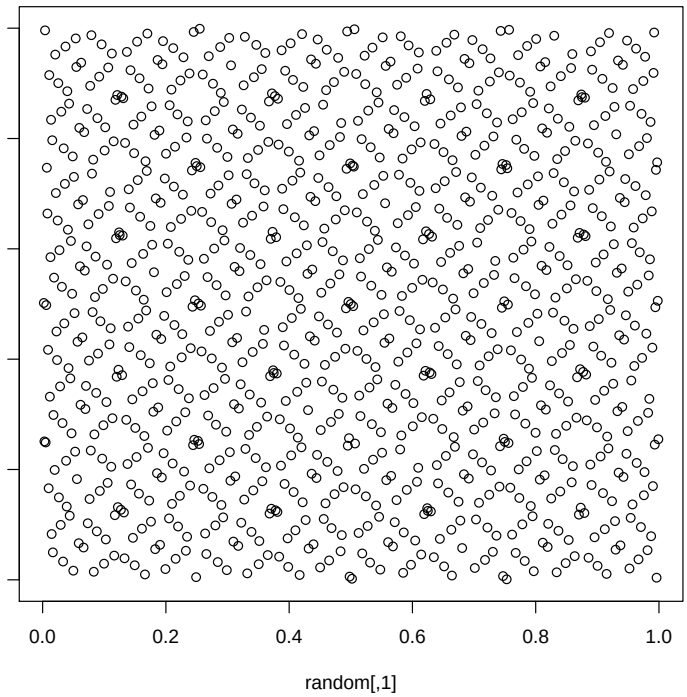

図 3.3　SOBOL 法による 1000*1000 の 2 次元準乱数散布図

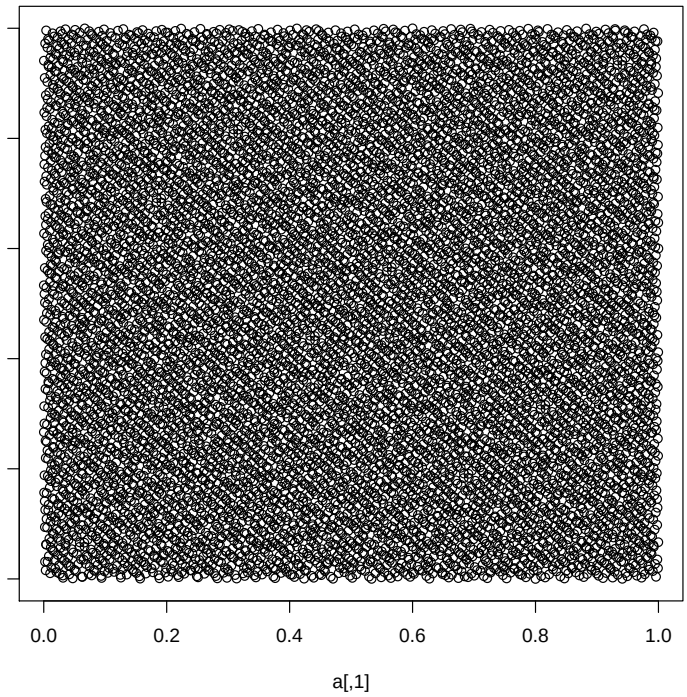

図 3.4　SOBOL 法による 10000*10000 の 2 次元準乱数散布図

第 4 章

Random Forest における乱数関数の準乱数への置き換え

本章では，先に述べた，Random Forest の決定木作成時のサンプリングおよび特徴選択に従来の疑似乱数を用いることの問題点を述べる．次に，Python の機械学習ライブラリ “scikit-learn” に収録されている Random Forest モジュールに，準乱数列を生成する関数を適用する．

4.1 Random Forest における疑似乱数列利用の問題点

Random Forest では，与えられた訓練データから復元抽出であるブートストラップサンプリングを乱数に基づいて行い，サンプリングにより得られたデータから，与えられた特徴に基づいて決定木を生成する．特徴の全てを決定木生成には用いず，乱数に基づいて限られた特徴を選択し，選択されたもののみを決定木生成に用いる．

Random Forest における疑似乱数利用の問題点は，生成する弱学習器である決定木の生成パラメータに関し，ユーザが任意に設定する決定木の数，1 つの決定木における木の深さの最大数，選択する特徴の数が少ない場合，生成する疑似乱数に偏りがあると，Random Forest の識別結果に影響を及ぼすことが考えられる．これは生成される疑似乱数に偏りがあることにより，弱学習木を構成するために選択される特徴にも偏りが生じる．偏りが生じたまま次のブートストラップ標本において特徴の選択を行うことで，直前に同じ処理を行ったブートストラップ標本から構成される決定木の出力結果と，その時点で選択される特徴か

ら構成される決定木の出力結果に相関が生じ，多数決でモデル全体の識別結果を分類する際に悪影響を与える．

4.2 Random Forest でのサンプリングおよび特徴選択における準乱数列の適用

疑似乱数をブートストラップサンプリングの乱数として利用すると，一部に乱数列の偏りが発生し，ブートストラップ間の相関が高まることが考えられる．また，特徴選択に用いる乱数に Xorshift 法で生成された乱数を用いる場合も同様で，同じ特徴を異なるブートストラップサンプルから複数回選択すると，生成されるそれぞれの弱学習器の出力結果間に相関が発生すると考える．Random Forest は弱学習器で出力された識別結果の多数決によって高精度識別を可能とするため，弱学習器の出力結果間に相関が発生することで多数決の結果に悪影響を与えることが考えられる．

一方で，準乱数列は均等で一様な点列を分布するアルゴリズムである．この均等で一様な数列を Random Forest に用いられているブートストラップサンプリングの乱数や決定木を生成する際の特徴選択に用いられる乱数に用いることで，疑似乱数で懸念される数列の偏りを解消すると考える．均等に分布する数列に基づくサンプリングでブートストラップサンプルを生成することで，決定木生成に利用するサンプルの相関を抑えることができると考える．同様に、均等分布な数列に基づいた特徴選択を行うことにより，同じ特徴を異なるブートストラップサンプルから複数回選択することを防ぎ，個々に独立した決定木群を生成することが期待される．

本研究では，一様分布な点列を生成する準乱数列を Random Forest のサンプリング，および特徴選択に用いる乱数関数に適用する．一様分布な点列であれば，疑似乱数列で問題となる数列の偏りが発生せず，各弱学習器の出力結果は決定木構成時の特徴選択に起因する，生成される決定木の間の相関が発生することはなく，Random Forest の長所である高精度の識別が可能であると考える．

4.3 Python の Random Forest モジュールへの準乱数列の適用

4.3.1 サンプリング乱数への準乱数列の変更

開発言語に Python を利用する．多くの機械学習アルゴリズムを含むライブラリ “scikit-learn”に収録されている Random Forest のメインプログラム “forest.py” 中で，Random Forest の決定木群をユーザ設定のパラメータおよびデータから構築する関数 fit() において，Random Forest が利用する疑似乱数列を準乱数列に変更するため，以下のように変更する．

```
...
for i in range(self.n_estimators):
    #追記部分 (1)，準乱数列の seed 決定
    seed = np.random.randint(MAX_INT)
    tree = self._make_estimator(append=False)
    #追記部分 (2)，準乱数列生成
    [j, seed] = i4_uniform(0, MAX_INT, seed)
    #追記部分 (3)，決定木生成のための乱数に準乱数列を設定
    tree.set_params(random_state=j)
    #決定木生成のための乱数に疑似乱数を設定（デフォルトで記述済）
    tree.set_params(random_state=random_state.randint(MAX_INT))
    trees.append(tree)
...
```

また，i4_uniform() 関数を利用するためのライブラリ “sobol_seq” をインポートする記述を以下のように “forest.py” の最上部に追記する．

```
from sobol_seq import *
```

以上の記述を追加したプログラムを用いて，識別実験を行う．なお，疑似乱数適用時の識別では追記部分全てをコメントアウトし，準乱数列適用時の識別では追記部分 (3) 直下の一文をコメントアウトしてプログラムを実行する．以上の一連の流れの概要を表したものを図 4.1 に示す．図中の実行プログラムの 1 行目で，オブジェクト rf に RandomForestClassifier 関

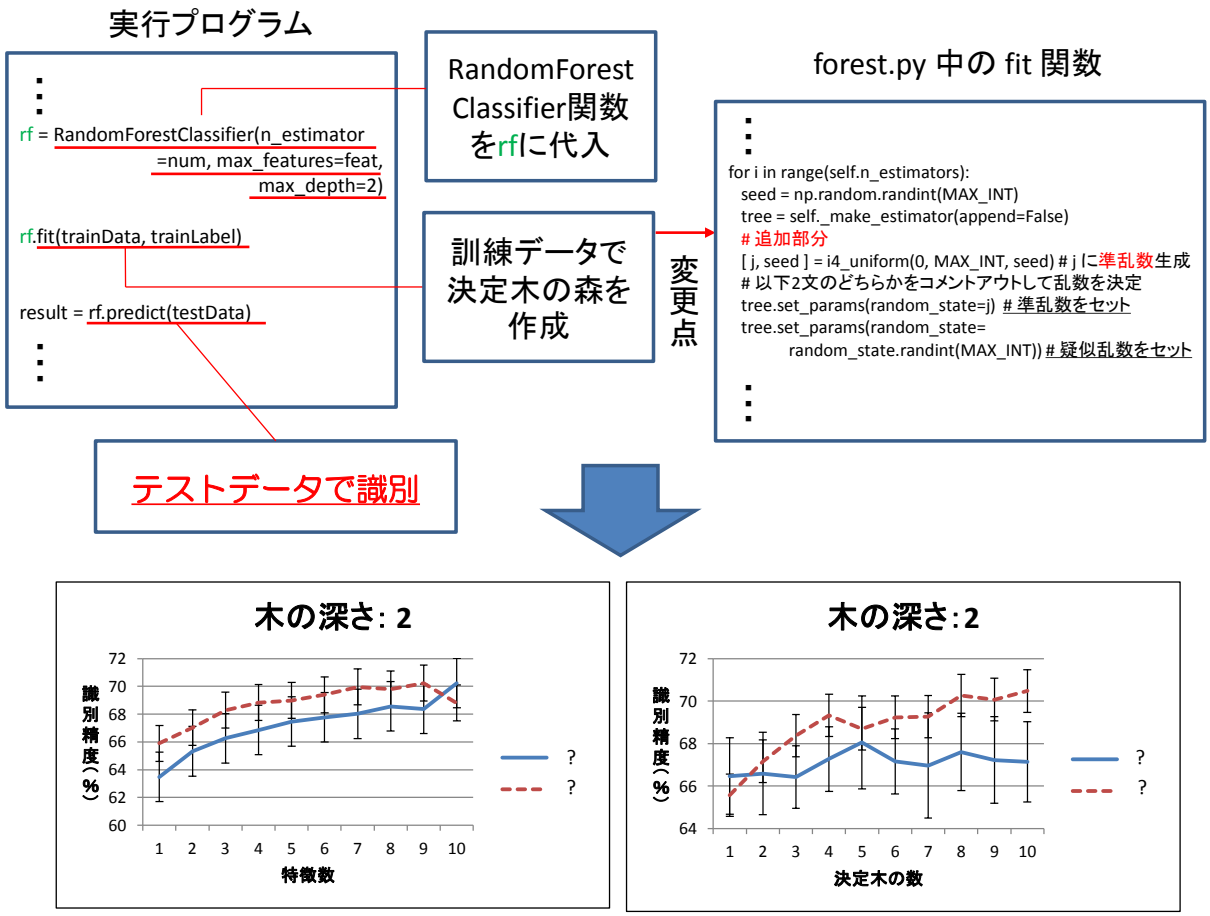

図 4.1　Random Forest においてサンプリングに用いる乱数列の変更点

数を格納する．与えられているパラメータは，“n_estimator”が決定木の数，“max_features”が選択する特徴の最大数，“max_depth”が生成される決定木の最大深度である．次の行で，引数に訓練データとその教師データとなるラベルを指定した fit 関数を呼び出している．こ

れは，訓練データを用いたトレーニングを表しており，ここで Random Forest のアルゴリズムと訓練データを用いて弱学習器となる決定木群を構築する．サンプリングにおける乱数の指定はこの fit 関数内で定義されているため，この乱数指定を Mersenne Twister による乱数から SOBOL による数列へ変更する記述を追加する．なお，識別実験を行う際には利用する乱数列に応じて，利用しない乱数をコメントアウトする方法を取る．図中の実行プログラムの最下段で，オブジェクト result に predict 関数の出力結果を格納している．この関数は，fit 関数で構築された弱学習器群を用いてテストデータの予測・識別を行う関数である．引数にはテストデータを指定している．

4.3.2 特徴選択に用いる乱数の準乱数列の変更

scikit-learn における Random Forest の特徴の選択は，決定木の生成を行う，“sklearn/tree” ディレクトリ内のコードで実行される．このコードは，Python と似た記述を行うことができ，C 言語のコードに変換してコンパイルすることで処理の高速化を実現できる，Cython にて実装されている．実行時には，動的リンクの実行ライブラリの形式である下記ファイルが実行される．

- _tree.so（UNIX 系）
- _tree.pyd（Windows）

このファイルを生成する元であるソースプログラムは，ソースコードが “_tree.pyx”に，宣言が “_tree.pxd”に格納され，Cython によって C 言語にコンパイルし，“_tree.c”を生成したのち，同ディレクトリにて “setup.py install”を実行することで .so または .pyd ファイルを生成する．

“_tree.pyx”内部では，特徴選択のための乱数を出力する rand_int 関数が呼び出されている．この rand_int 関数は，デフォルトでは xorshift 乱数 [12] を用いて計算し，その値を返すようになっているが，本研究ではこの命令を SOBOL 列に書き換えることによって，特徴選択に SOBOL 列を用いる．

第 5 章

前提条件下における準乱数を用いた識別と疑似乱数を用いた識別の比較

本章では Random Forest のサンプリング，および弱学習器における特徴選択に用いる乱数関数を準乱数へ変更することにより，日本語の SPAM メールデータ 1600 通（SPAM : 600，非 SPAM : 1000）に対して，特定条件下での Random Forest の分類精度が向上することを示す．まず実験環境，設定したパラメータ類について説明し，実験の結果と考察を述べる．

5.1 識別実験の内容

5.1.1 実験環境および実験条件

表**??**に実験環境を示す．開発言語に Python 2.7.6 を用い，数値計算ライブラリに NumPy，機械学習ライブラリに scikit-learn，SOBOL 列の生成に sobol_seq.py を用いる．学習・テストに用いるデータには，独自に集めた日本語電子メールデータを用いる．このデータは，2004 年〜2007 年に日本国内の大学で収集された電子メールで，SPAM メールが 600 通，非 SPAM メールが 1000 通含まれており，それらが 52 次元の特徴で表現されている．

表 5.1　実験環境

OS	Windows 7 Enterprise
メモリ	4.00 GB
CPU	Intel(R) Core(TM) i5-2400S CPU @ 2.50GHz
開発言語	Python 2.7.6
利用ライブラリ	scikit-learn，NumPy，sobol_seq
利用データセット	日本語 SPAM メールのデータセット （SPAM：600 通，非 SPAM：1000 通，格納特徴数：52）
訓練データ数	50
予測データ数	1550

5.1.2　実験内容

本研究は，Random Forest において，ユーザが設定できる 3 つのパラメータに関し，いずれの値も低い状態に設定した条件の下で，サンプリングに用いる乱数に準乱数を適用することによって，疑似乱数よりも識別精度が向上することを示すものである．3 つのパラメータの値を表 5.2 に示す.

表 5.2　本研究での Random Forest におけるパラメータ

木の最大の深さ $D_{\max}$	1, 2, 3, 4, 5
特徴の最大数 $F_{\max}$	1,2,3,...,10
認識に用いる木の数 $N_{\max}$	1,2,3,...,10

表中の $D_{\max}$ が生成される木の深さの最大数を，$F_{\max}$ が選択される特徴数の最大数を，$N_{\max}$ が生成される木の総数を示す．また，本稿で述べる「木の深さ」を図 5.1 に示す．木の深さの最大数を 1〜5 にした理由は，二分決定木のノード数があまり大きくならず，使用する特徴の数に制限がある環境下における性能の向上を考えるためである．したがって，木

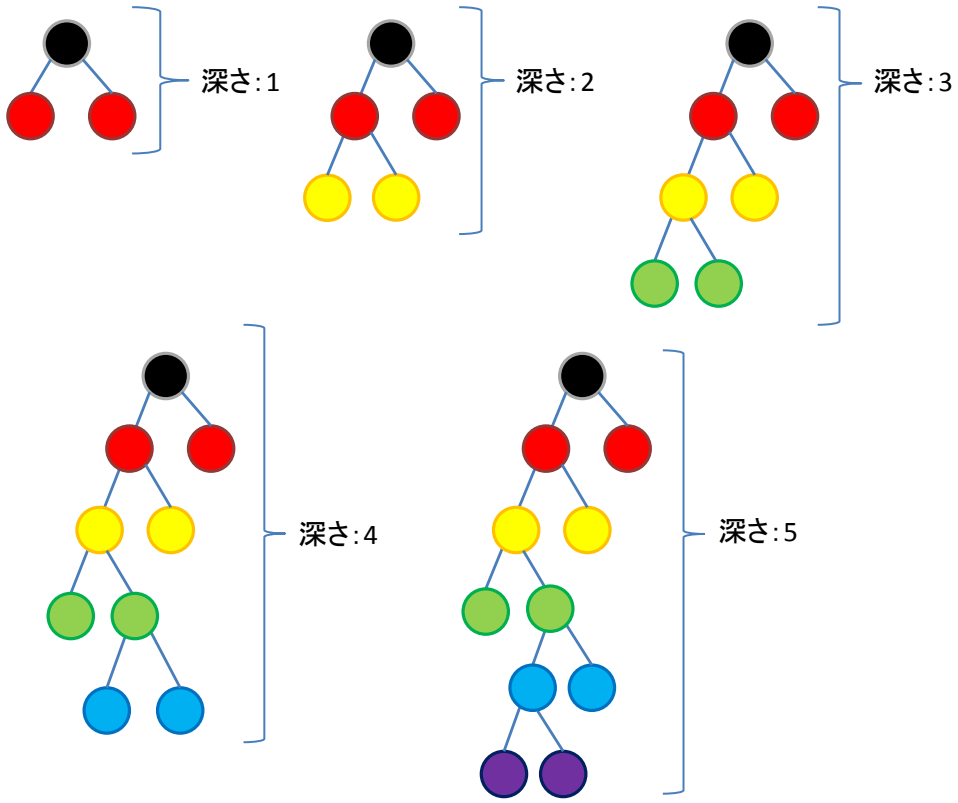

図 5.1　各深度で作成される決定木の例

の深さを 6 以上に設定することはノード数，つまり利用する特徴数が増加するということであり，また本研究は各パラメータを低く設定した時における準乱数適用時と疑似乱数適用時の識別精度の比較を目的とするため，木の深さの最大数は 5 層までとする．選択される特徴の最大数および生成される決定木の総数に関しても大きな学習器を作成しない状況下での比較をするため，各パラメータは最大 10 までとする．また，ランダム性が含まれるアルゴリズムの特性を考慮し，同一条件の試行を 10 試行繰り返し行うものとする．

5.1.3　評価方法

本研究を行うにあたって，生成される木の深さの最大数，選択する特徴の最大数，生成する木の総数の 3 つのパラメータに関し，いずれの値も低い状態で準乱数列と疑似乱数列をそれぞれ適用した場合に，識別精度の高かった乱数列を，その条件下における適した乱数列で

あると評価するものとする．評価基準は，各パラメータが固定された 10 回の試行を疑似乱数を用いた場合と準乱数列を用いた場合の 2 つのケースで比較し，識別率の高い手法にマーキングし，固定されたパラメータの中で 10 回の試行のうち何回識別率が他方より高いのかをカウントする．この評価を，$F_{\max} * N_{\max} * 10$（試行）$* D_{\max} = 5000$ データに関して行い，各パラメータの条件下で，2 種類の乱数を用いた手法それぞれで，識別精度がもう一方の乱数を用いる方法より高かった場合の回数を数え，その評価値が標準偏差，標準誤差を超える高い値である手法に対し，識別性能が高いという評価を行うものとする．

5.2　サンプリングにおける乱数に準乱数を用いた識別の結果と精度比較における考察

5.2.1　準乱数適用時の識別と疑似乱数適用時の識別精度が他方を上回る回数についての考察

表 5.3　木の深さに着目した際の各手法の識別精度におけるが他方を上回る数

	準乱数列	疑似乱数列
深さ：1	260	740
2	695	305
3	535	465
4	546	454
5	190	810
標準偏差	190	190
平均値	445	555

条件を変えながら試行回数を増やし実験を行った結果，5000 個の判別データを取得することができた．この中から，定義した評価基準に従って有利な数をカウントした結果，木の

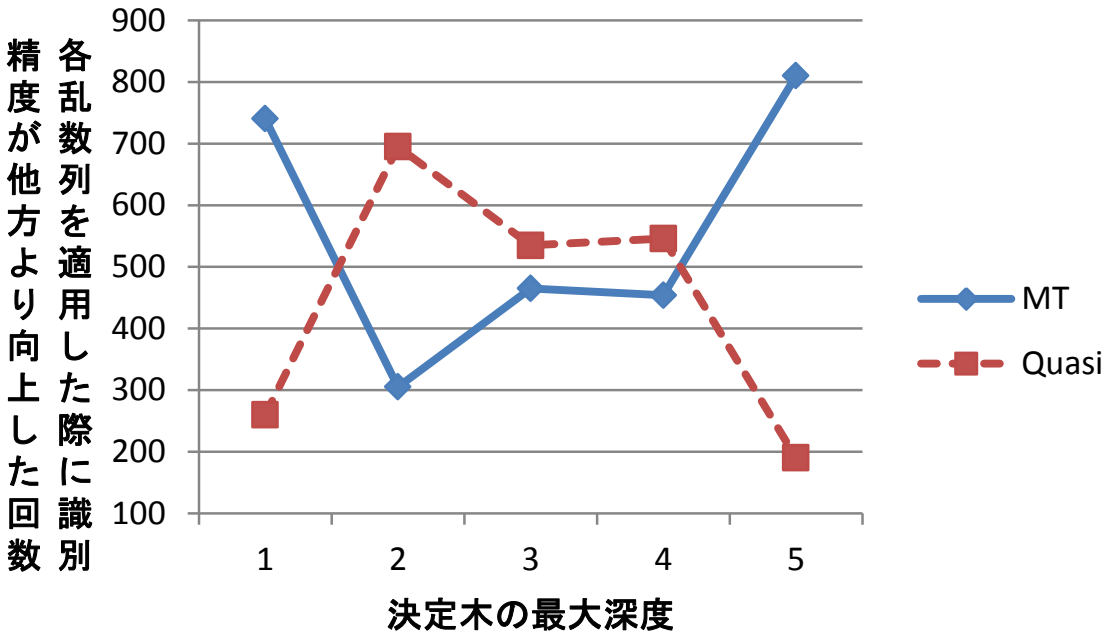

図 5.2　木の深さに着目した時の各手法の識別精度が他方を上回る数の推移

深さが 2（$D_{\max} = 2$）の時に準乱数列を適用した場合が疑似乱数列を適用した場合を上回った．カウントの総和についてグラフにまとめたものが図 5.2 であり，その数値に関して表にまとめたものが表 5.3 である．このグラフから，$D_{\max} = 2$ の時に準乱数列を適用した場合が疑似乱数列を適用した場合を上回る識別能力を見せ，その後低下しており，$D_{\max} = 5$ の時には疑似乱数列を適用する場合の精度が良くなる．この比較指標を用いる理由は，本研究の初期段階において条件となるパラメータを定義する際に，識別精度に差が出にくいと考えていたためである．識別精度に大きな差がなくとも，他方より識別精度が向上している回数を調べ，その総和を比較し，その差が大きければ精度が良いか悪いかを判断することができるのではないかと考える．

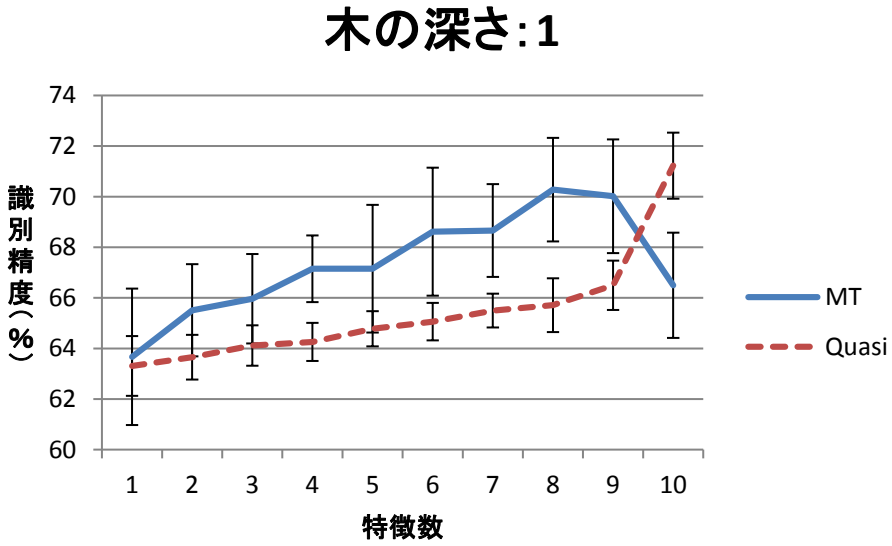

図 5.3　$D_{\max} = 1$ において選択特徴数に着目した際の各手法の識別精度の推移

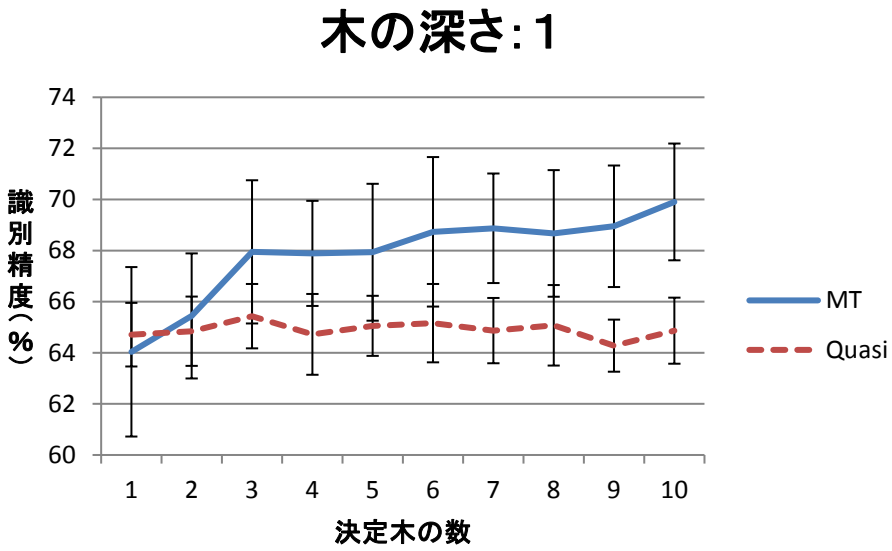

図 5.4　$D_{\max} = 1$ において決定木の数に着目した際の各手法の識別精度の推移

5.2.2　$D_{\max} = 1$ のときの疑似乱数と準乱数の比較

図 5.3 は $D_{\max} = 1$ において最大選択特徴数に着目した識別精度の推移を，図 5.4 は同じく決定木数に着目した識別精度の推移を表す．グラフ中において，特徴数の各値における識別精度には，もう一方のパラメータである $N_{\max}$ が 1〜10 の際の精度を平均したものを記している．この時点での準乱数適用時の精度は，平均的に疑似乱数適用時の精度よりも低い．図 5.3 における精度差の平均は 2.9% , 最も大きい差は $F_{\max} = 8$ の時で 4.6% の差が見られる．図 5.4 における精度差では，平均 3.1% , 最大 4.6% の差が確認できる．しかし，本パラメータにおいて木の深さが 1 であるという決定木は，利用する特徴の数がただ 1 つと非常に少ない特徴数で構成される決定木が生成される．1 つの特徴値を基準に SPAM メールか否かを判別するため，決定木のノードに利用される特徴の値によって，SPAM か否かの 2 値判別に与える影響は少なからず存在する．しかし，2 値判別であることからどちらかに偏って判別されるというような結果を得ることはまれであると考えられるため，その識別精度を他の値の精度と比較することは難しいと考える．

5.2.3　$D_{\max} = 2$ のときの疑似乱数と準乱数の比較

図 5.5 は $D_{\max} = 2$ において最大選択特徴数に着目した識別精度の推移を，図 5.6 は同じく決定木数に着目した識別精度の推移を表す．グラフ中において，特徴数の各値における識別精度には，もう一方のパラメータである $N_{\max}$ が 1〜10 の際の精度を平均したものを記している．$D_{\max} = 1$ に比べ，こちらは平均的に準乱数適用時が疑似乱数適用時の精度を上回る．図 5.5 では $F_{\max} = 9$ までの間，準乱数を用いた識別が疑似乱数を用いるより高い精度で識別しており，平均 1.8% ，最高で $F_{\max} = 3$ のとき 2.0% の精度向上が確認できる．図 5.6 からは，$N_{\max} = 2$ から $N_{\max} = 9$ の間，準乱数を用いた識別が疑似乱数を用いるより高い精度で識別しており，平均 2.0% ，最高で $N_{\max} = 10$ のとき 3.3% の精度向上が確認できる．しかし，どちらのグラフのどのパラメータにおいても標準偏差の重複が見られるため，必ずしも準乱数を用いる識別が疑似乱数を用いる識別を上回るとは断定できない．ただ

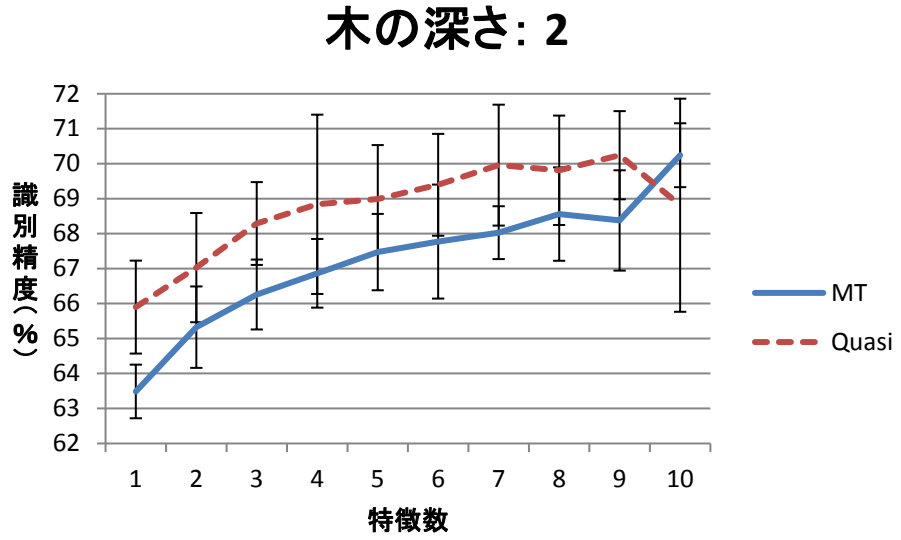

図 5.5　$D_{\max} = 2$ において選択特徴数に着目した際の各手法の識別精度の推移

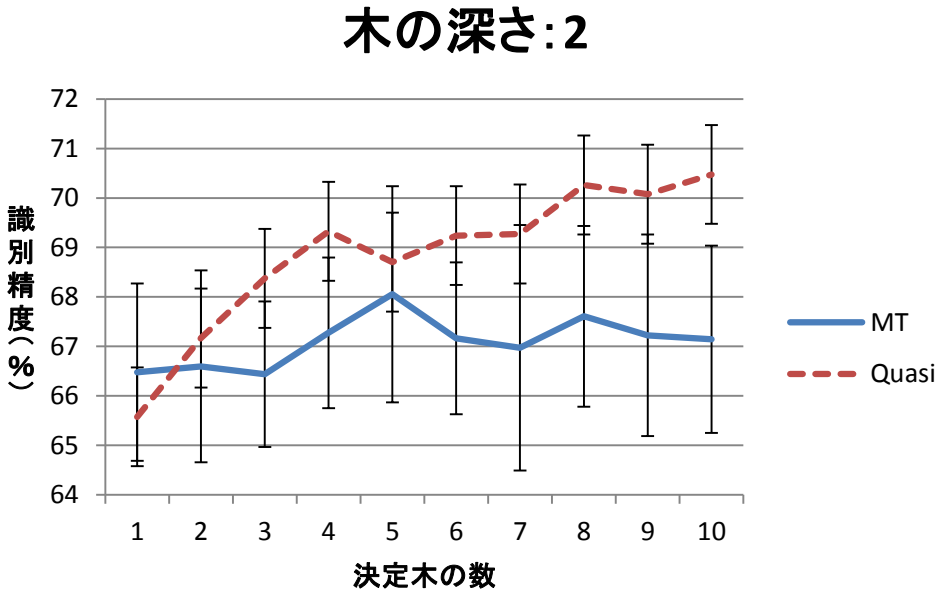

図 5.6　$D_{\max} = 2$ において決定木の数に着目した際の各手法の識別精度の推移

し，$N_{\max} = 10$ においては標準偏差の重複がないため，準乱数を用いる方が疑似乱数を用いるより精度が良くなると考える．

5.2.4　$D_{\max} = 3$ および $D_{\max} = 4$ のときの疑似乱数と準乱数の比較

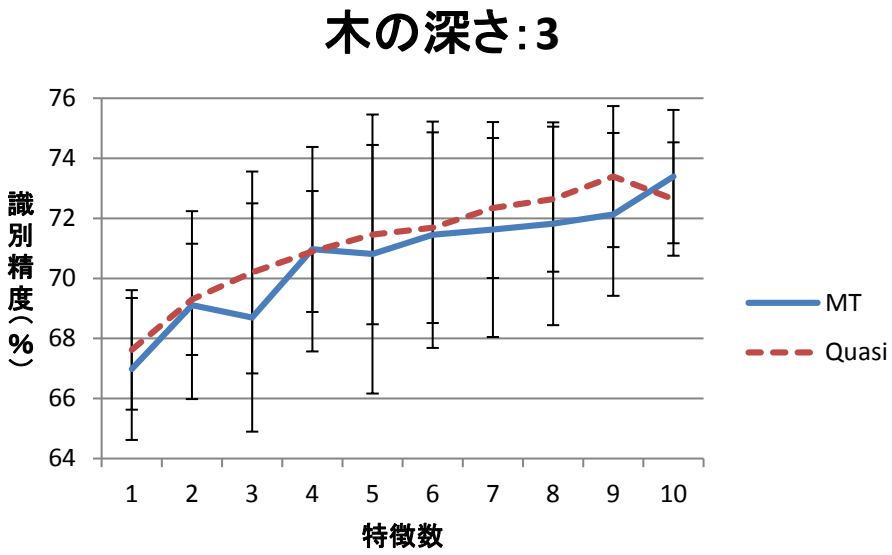

図 5.7　$D_{\max} = 3$ において選択特徴数に着目した際の各手法の識別精度の推移

図 5.7 は $D_{\max} = 3$ において最大選択特徴数に着目した識別精度の推移を，図 5.8 は同じく決定木数に着目した識別精度の推移を表す．同様に，図 5.9 は $D_{\max} = 4$ において最大選択特徴数に着目した識別精度の推移を，図 5.10 は同じく決定木数に着目した識別精度の推移を表す．いずれのグラフにおいても，特徴数の各値における識別精度には，もう一方のパラメータである $N_{\max}$ が 1〜10 の際の精度を平均したものを記している．図 5.7 における精度差は平均 0.7% ，最大 1.5% となり，図 5.8 では平均 0.7% ，最大 3.6% の精度差を確認した．しかし，こちらは $D_{\max} = 2$ とは異なり，途中で識別精度の優劣が逆転した．また，図 5.9 における精度差は平均 0.8% ，最大で 1.2% となり，図 5.10 では平均 0.6% ，最大 1.0% の精度差が確認できる．$D_{\max} = 3$ は着目したパラメータの 1 つの値で優劣の逆転が

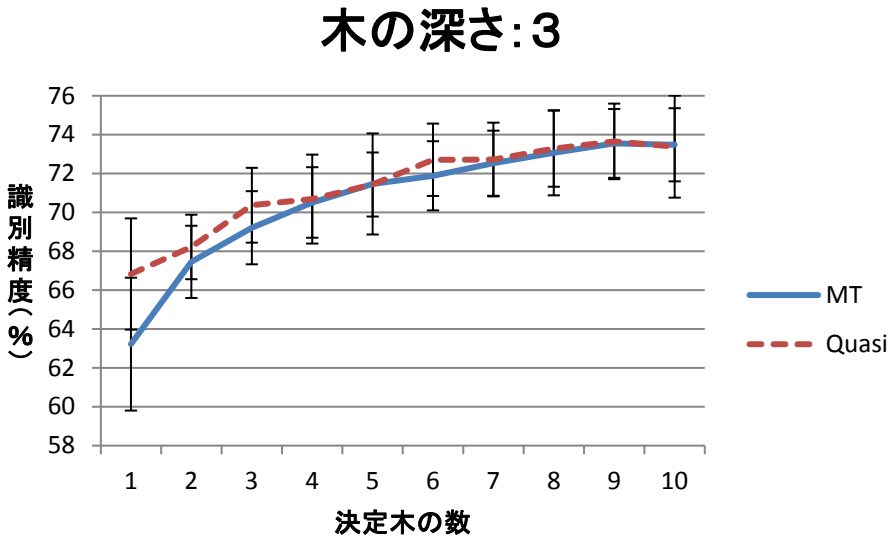

図 5.8　$D_{\max} = 3$ において決定木の数に着目した際の各手法の識別精度の推移

発生し，$D_{\max} = 4$ でも優劣逆転が発生している．$D_{\max} = 4$ においては，準乱数を用いる識別の精度が疑似乱数を用いる識別精度を上回る回数が多い．しかし，どのパラメータにおいてもその標準偏差に重複が見られるため，本パラメータを利用した識別精度の比較においては差が見られない．

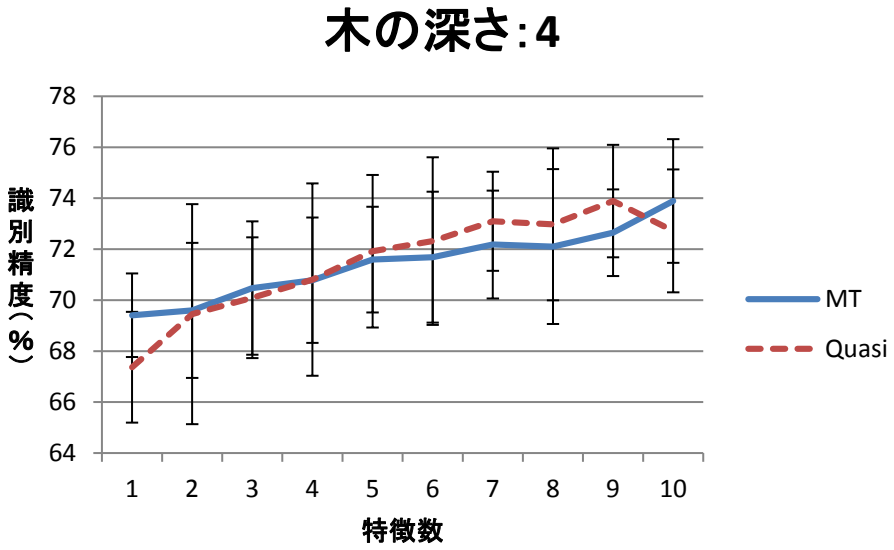

図 5.9　$D_{\max} = 4$ において選択特徴数に着目した際の各手法の識別精度の推移

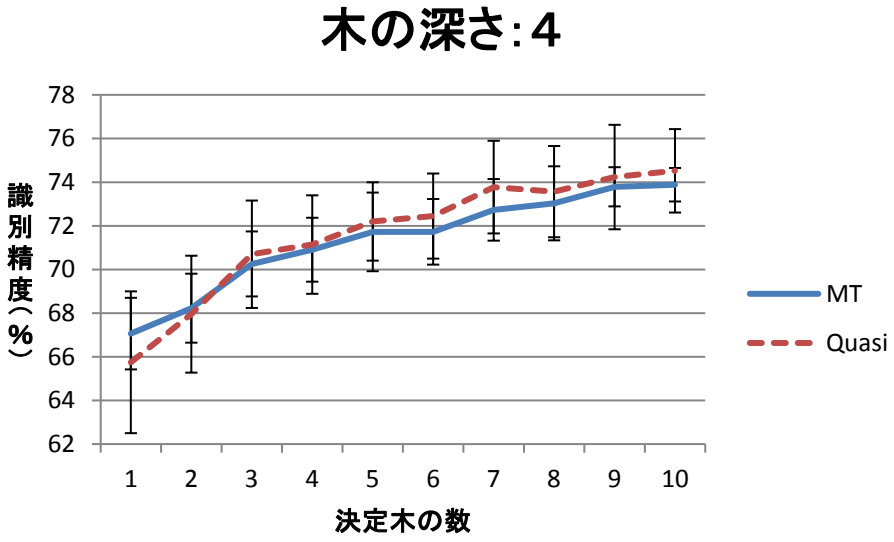

図 5.10　$D_{\max} = 4$ において決定木の数に着目した際の各手法の識別精度の推移

5.2.5　$D_{\max} = 1$ のときの疑似乱数と準乱数の比較

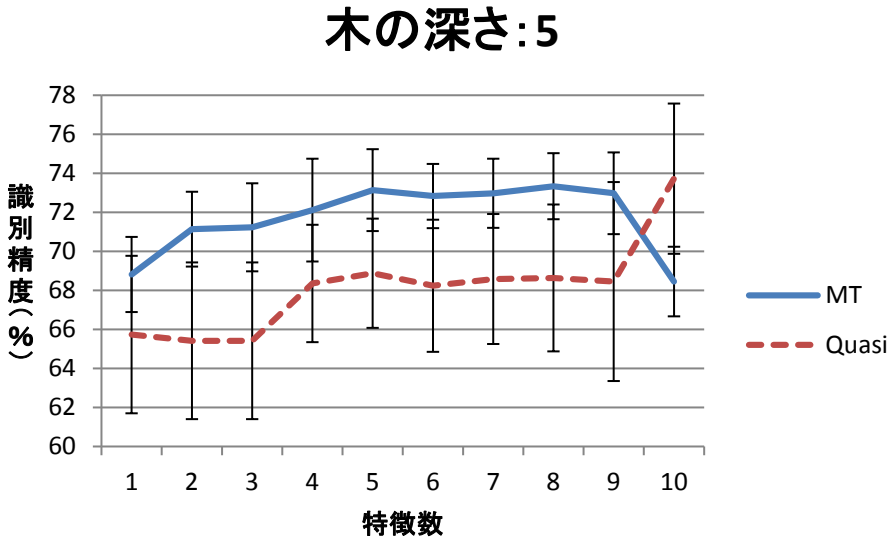

図 5.11　$D_{\max} = 5$ において選択特徴数に着目した際の各手法の識別精度の推移

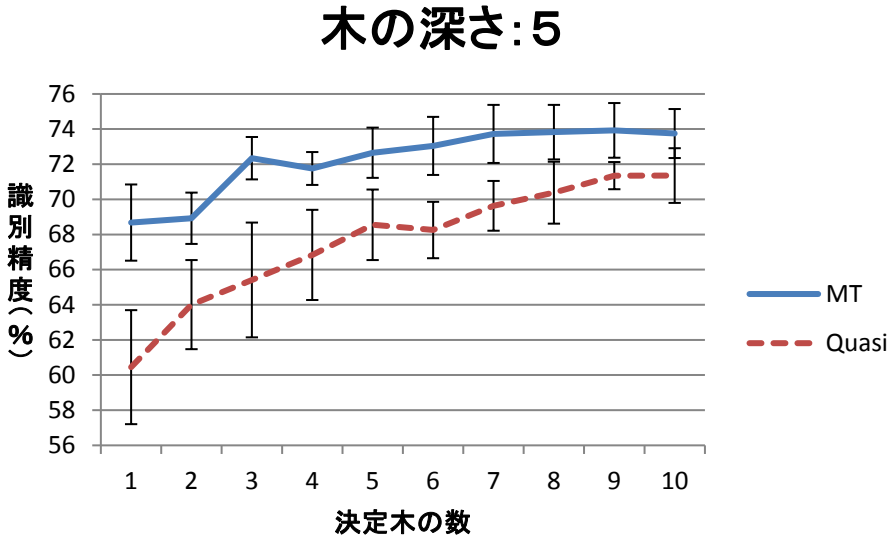

図 5.12　$D_{\max} = 5$ において決定木の数に着目した際の各手法の識別精度の推移

図 5.11 は $D_{\max} = 5$ において最大選択特徴数に着目した識別精度の推移を，図 5.12 は同じく決定木数に着目した識別精度の推移を表す．グラフ中において，特徴数の各値における識別精度には，もう一方のパラメータである $N_{\max}$ が 1〜10 の際の精度を平均したものを記している．図 5.11 における識別精度の差は平均 4.6% ，最大 5.7% となり，図 5.12 では平均 4.6% ，最大 8.2% の精度差が確認できる．$D_{\max} = 5$ では $D_{\max} = 1$ と同様に疑似乱数を用いる識別が準乱数を用いる識別より高い識別率となる．また，図 5.11 が $F_{\max} = 10$ でその優劣順序が逆転するのに対し，図 5.12 では疑似乱数を用いる識別が良くなる．

木の深さが 5 の決定木では，約 1000 個の特徴を利用するが，本研究ではその特徴の数を最大 10 までに限定している．また，疑似乱数が偏りを多少持った数列を生成するのに対し，準乱数は一様な分布を持つ数列を生成する．Random Forest において，弱学習器である決定木それぞれは，他の決定木と異なった多様な決定木として生成されるよう考えられた学習モデルである．最大選択特徴数を 10 以下に制限した場合，特徴選択に疑似乱数を用いる方法では偏りは発生するものの多様な決定木が生成されるが，特徴選択に準乱数を用いる方法では，一様な分布を持つ数列を特徴選択に利用するため，似通った決定木が生成され，そこから出力される結果も似通ってしまい，集団学習の効果が低下してしまうと考えられる．以上の結果および考察から，疑似乱数を用いた識別が本パラメータにおいて SPAM メール判別に有効な手法であると考える．また，それぞれのパラメータで取得した識別精度の表は付録 A を参照されたい．

5.2.6　前提条件を上回るパラメータ値における識別精度の比較

本研究を行うにあたって設定したパラメータ等の前提条件に関し，このしきい値を超えたパラメータを設定した場合，準乱数を用いる識別は疑似乱数を用いる識別を下回ると言えるのかについて，先に行った実験で準乱数を用いる識別に有利な精度を記録した $D_{\max} = 2$ において実験を行った結果，ほぼすべてのパラメータの組み合わせで準乱数列用いる識別が疑似乱数を用いる識別よりも精度が下回る，もしくは標準偏差範囲内におさまる．この実験結果を精度で比較したグラフを，図 5.13 および図 5.14 に示す.

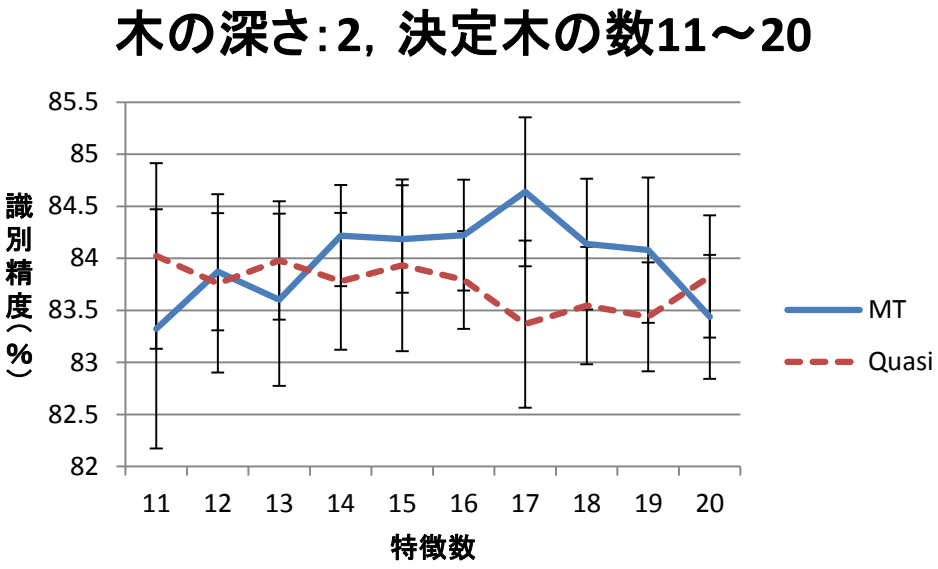

図 5.13　$D_{\max} = 2$，$F_{\max} = 11$～20，$N_{\max} = 11$～20 での実験において特徴数に着目した際の識別精度の比較

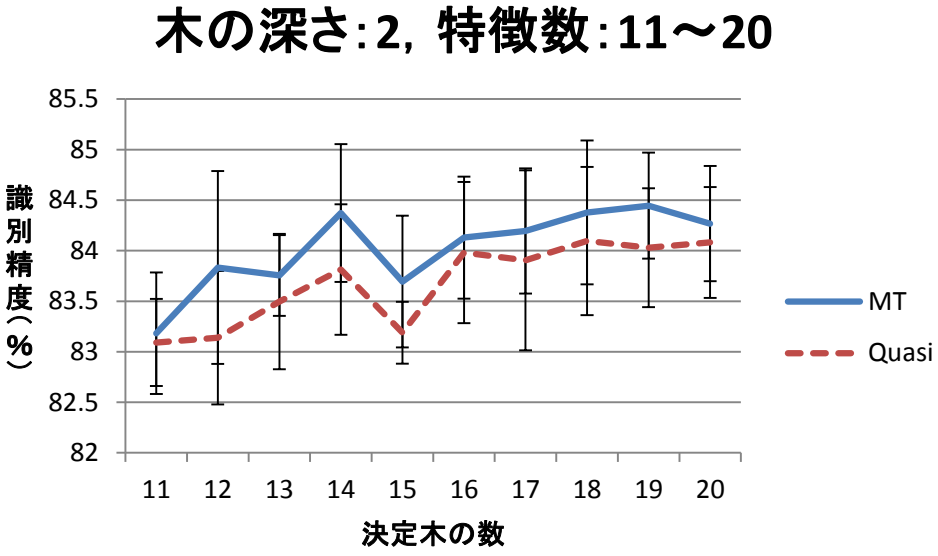

図 5.14　$D_{\max} = 2$，$F_{\max} = 11$～20，$N_{\max} = 11$～20 での実験において決定木数に着目した際の識別精度の比較

5.2.7　設定パラメータとそのしきい値を超える値の組み合わせによる識別の比較

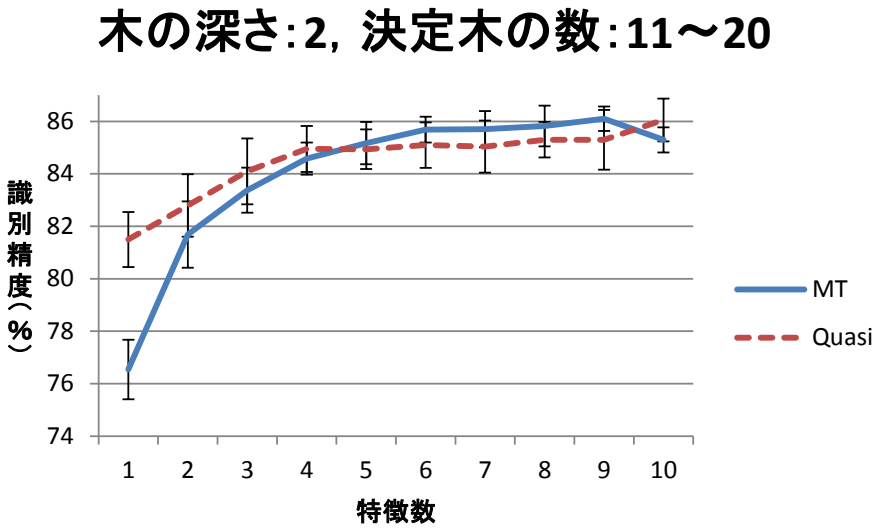

図 5.15　$D_{\max} = 2$，$F_{\max} = 1$～10，$N_{\max} = 11$～20 での実験において特徴数に着目した際の識別精度の比較

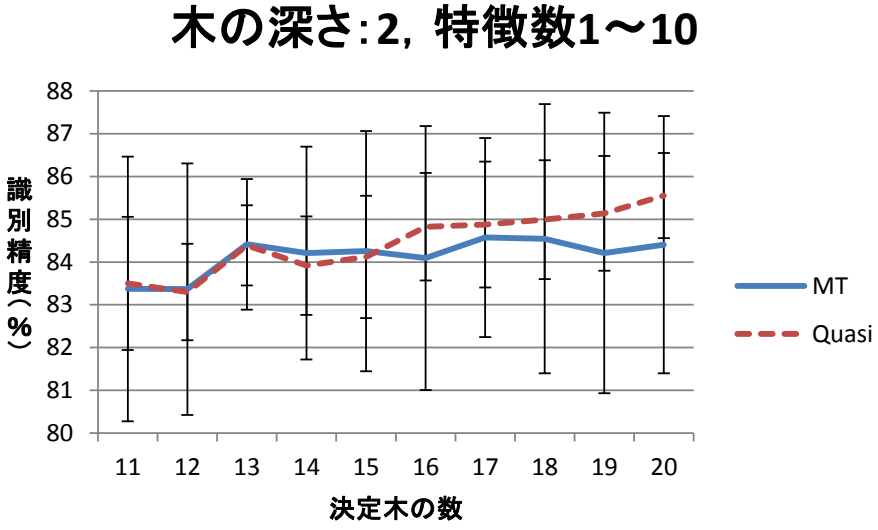

図 5.16　$D_{\max} = 2$，$F_{\max} = 1$～10，$N_{\max} = 11$～20 での実験において決定木数に着目した際の識別精度の比較

また，先に行った実験は $F_{\max}$，$N_{\max}$ ともに 1〜10 のパラメータを用いて識別実験を行った．このパラメータを片方が維持した状態で，もう片方を 11〜20 に変化させることで識別精度はどう変化するのかを調べるため，さらに 2 パターンの実験を行った．この結果の精度を比較したグラフを図 5.15〜図 5.18 に示す．これらの図からわかるように，いずれのパターンにおいても準乱数を用いる識別が疑似乱数を用いる識別の識別精度を下回るか，標準偏差の範囲内で競合することを確認した．この 2 つの追加実験の結果から，準乱数を用いる識別の精度は本研究で前提条件として定義した，パラメータを低くした実験条件の下で，疑似乱数を用いる識別の精度との間に差は見られない．

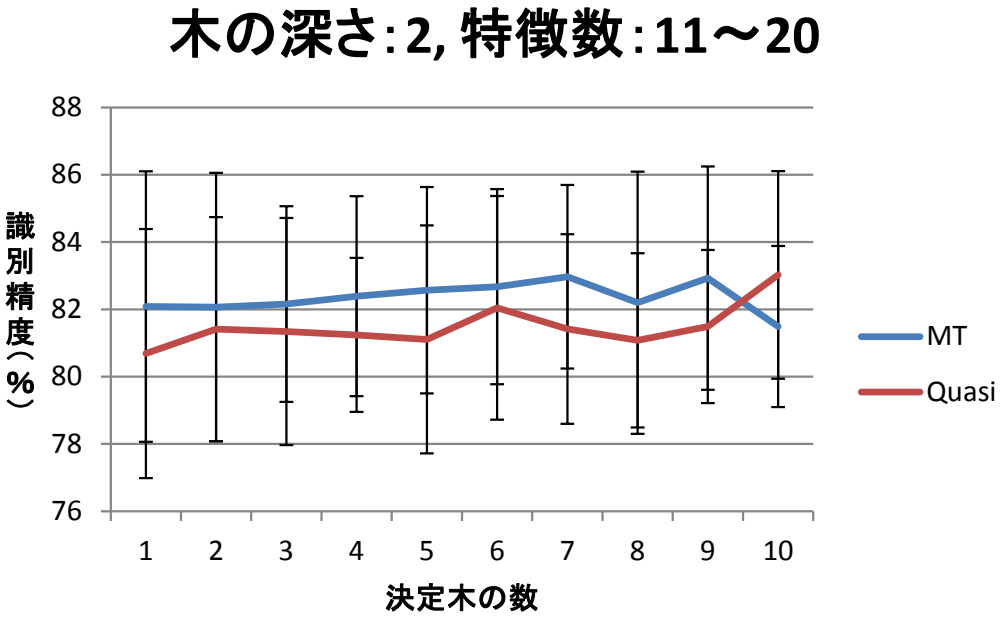

図 5.17　$D_{\max} = 2$，$F_{\max} = 11$～20，$N_{\max} = 1$～10 での実験において特徴数に着目した際の識別精度の比較

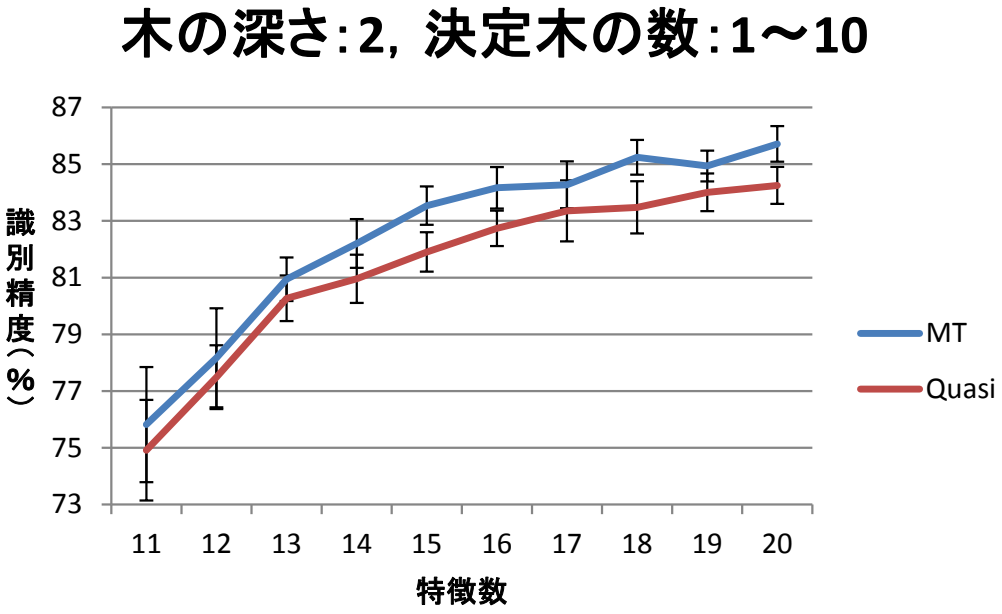

図 5.18　$D_{\max} = 2$，$F_{\max} = 11$～20，$N_{\max} = 1$～10 での実験において決定木数に着目した際の識別精度の比較

5.3　特徴選択の乱数へ準乱数を適用した際の選択結果と考察

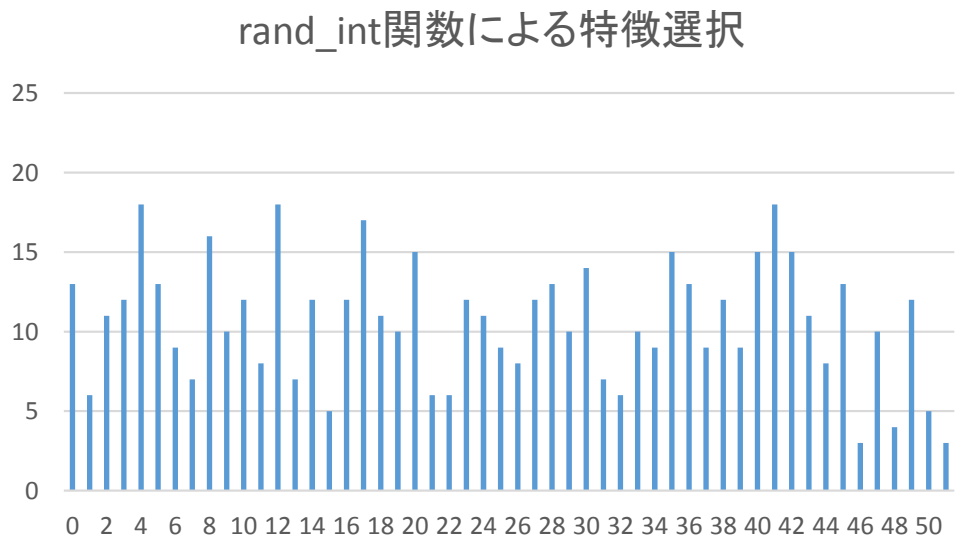

図 5.19　rand_int 関数による特徴選択のヒストグラム

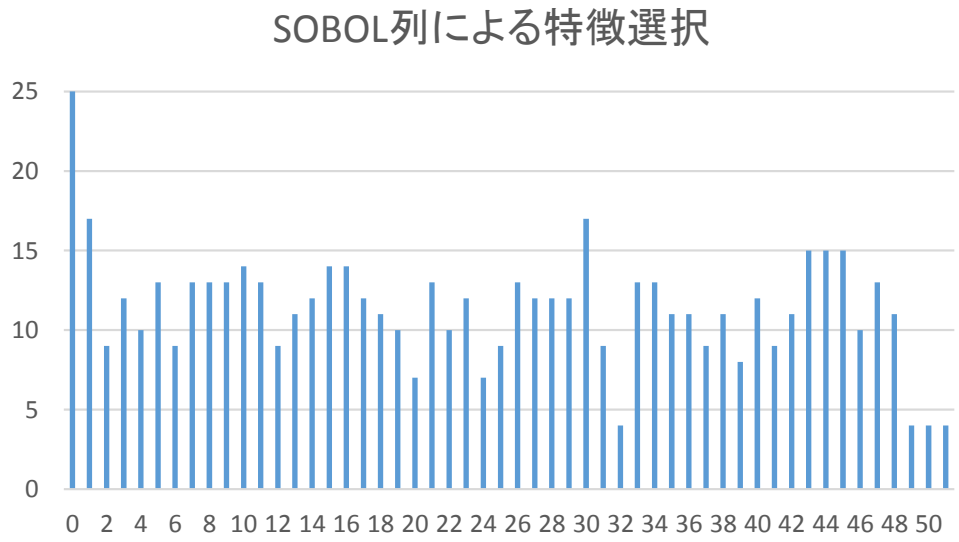

図 5.20　SOBOL 列による特徴選択のヒストグラム

図 5.19 が決定木生成時に rand_int 関数により選択された特徴のヒストグラム，図 5.20

が本研究で提案する SOBOL 列二より選択された特徴のヒストグラムである．縦軸はその特徴が選択された回数，横軸は特徴のラベル番号（0〜51）を示している．実験条件については，サンプリング乱数に準乱数を適用した際に，準乱数を用いた識別の精度が良かった $F_{\max} = 5, N_{\max} = 5, D_{\max} = 2$ とする．SOBOL 列による特徴選択においては，SOBOL 列の特長である一様性が反映された結果であるのに対し，rand_int 関数による特徴選択は選択される特徴の選択回数の差が大きい．決定木間の相関は，この選択される特徴がより均一であるほど，相関が低くなることが期待されると考えるため，rand_int 関数に比べ SOBOL 列を用いた特徴選択は Random Forest の精度向上に適した手法であると考える．

第 6 章

結論

本研究では，機械学習アルゴリズムである Random Forest の特徴選択時に用いられる乱数列を疑似乱数列から準乱数列に置き換えることで，特定条件下における識別精度の向上を目指した．具体的には，条件として，ここでは弱学習器の木の深さの最大数，選択される特徴の数，識別に利用される木の数を小さく設定した場合を考える．これらのパラメータは Random Forest の弱学習器である決定木の構築に関わるパラメータであり，一般的には大きな値が良いとされる．しかし，ユーザが利用する端末の計算リソースが少ない場合，このパラメータを小さく設定する必要があると考える．このような条件下で疑似乱数を用いた特徴選択による決定木の構築を行うと，疑似乱数生成時の数列の偏りにより，弱学習器間の相関が高まる問題が考えられる．そこで本研究では，この条件下において特徴選択に用いる乱数を，値のばらつきの一様性が高い数列である準乱数を用いることによって解決することができると考え，この条件を前提とした識別実験を行った．本研究における特定条件として以下のパラメータを定義した．決定木の最大深度 $D_{\max}$ を 1〜5，選択する特徴の最大数 $F_{\max}$ を 1〜10，識別に利用する決定木の総数 $N_{\max}$ を 1〜10 とし，それぞれの組み合わせにおいて準乱数適用時と疑似乱数適用時のそれぞれで識別実験を行った．識別するデータには，SPAM メール 600 通，非 SPAM メール 1000 通からなる，独自に用意した日本語 SPAM メールのデータセットを用いて，訓練データ数を 50，テストデータ数を 1550 として識別実験を行った．また，選択する特徴に乱数を利用することから識別結果そのランダム性があるため，組み合わせ 1 パターンごとに 10 回識別を行い，その平均値を 1 パターン分の識別率とした．実験の結果，$D_{\max} = 2$ の時に，$F_{\max} = 1$〜9 と $N_{\max} = 2$〜10 の場合に，準乱数列を用いる方が疑似乱数列を用いる方より平均 1.9% の精度向上を確認した．設定パラ

メータにおける他パターンでの識別結果の比較から，$D_{\max}$ が増加するにしたがって準乱数を用いる場合の識別精度は低下し，疑似乱数を用いる場合は上昇する．このことから，決定木の深さの最大数，選択する特徴の最大数，学習に用いられる木の数が少ない，$D_{\max} = 2$, $F_{\max} = 1$〜9, $N_{\max} = 2$〜10 といった条件の下で準乱数を用いる場合は疑似乱数を用いる場合より高精度な識別が可能であり，特に $D_{\max} = 2$ の場合に，疑似乱数を用いるより準乱数を用いる方が精度が向上することを確認した．しかし，この時の各パラメータにおける標準偏差が，両乱数列で重複している箇所が多いため，深さ 2 が準乱数列を適用するのに最適なパラメータであるとは断定できないと考える．また，前提条件として定義した $F_{\max}$ および $N_{\max}$ の範囲が前提条件として有効であるかどうかを調べるため，それぞれの範囲を 1〜20 に拡大して，$D_{\max} = 2$ において再実験を行った．その結果，ほとんどのパラメータ設定パターンにおいても，有意な差は見られなかった．

■追加実験：UCI Spambase dataset での検証　以上の結果が有効であることを確認するため，本研究で用いたデータセットとは異なる，UCI Machine Learning Repository で公開されている “Spambase dataset（4601 件分）” を用いて識別実験を行い，その精度の比較を行った．この識別実験に利用するパラメータは，独自作成のデータセットにおいて識別を行った実験と同じパラメータを利用した．また，訓練データとテストデータの数は，前の実験でデータセットのうち全体の 3% を訓練データ，残りをテストデータとしていたことから，同様に全体の 3% にあたる 138 件を訓練データ，残りである 4463 件をテストデータとして実験を行った．実験の結果，やはり $D_{\max} = 2$ において，疑似乱数を用いる識別の精度より準乱数を用いる識別の精度が，平均 1.9 %，最大 3.2% の精度向上を確認した．このことから，本研究で提案している，$D_{\max} = 2$, $F_{\max} = 1$〜10, $N_{\max} = 1$〜10 の条件下では，SOBOL 列を用いる方が良いと考える．

今後の展望としては，SOBOL 法以外の準乱数列生成アルゴリズムを用いた場合の識別精度の比較を行うことや，疑似乱数の数列の偏りを検知して疑似乱数列を準乱数列に切り替えるアルゴリズムを Random Forest に組み込むことによって，Random Forest の更なる高

精度識別化ができると考える．また，本研究の結果から，一様分布である準乱数を用いることが疑似乱数の数列の偏りへの対策法となり得ると考え，このことは Random Forest だけでなく乱数を用いる機械学習アルゴリズムに対しても，識別精度の低いパラメータセットに対して準乱数を用いることで精度の向上を図ることができると考える．

謝辞

本研究を進めるにあたり，ご指導いただきました高知工科大学　情報学群　吉田 真一 准教授には大変お世話になりました．吉田先生には，3 年生の終わりに，それまで就職一本で就職活動をしていたにも関わらず，急に進学したいとわがままに対し，自分の意思ならば尊重すると歓迎してくださいました．ありがとうございます．その後の 2 年間では，今後決して経験できないような，国際学会での論文発表を 2 回も与えていただきました．中国・韓国と国際学会に進出するにあたってまずはアジアから発表を重ね，最終的に欧米の会議にも参加できるようになる，とのお考えのもとでしたが，それでも 1 年の間に 2 回も発表の機会を頂けるとは思っていなかったので，大変貴重な経験になりました．ありがとうございます．ありがとうございます．その後も，研究や就職活動で行き詰った私を支えていただき，無事就職先の内定を頂くことができ，本稿となる修士論文もまとめることができました．特に，修士論文に関しては，研究修了や卒業が絶望的となった私を何度も励ましてくださいました．あの時，先生のバックアップがなければ私はこうして卒業もできず，退学という選択肢を選んでいたと思います．先生にご指導，ご助言を頂けたこと，深く感謝致します．

本研究の副査を引き受けていただきました，高知工科大学　情報学群　福本 昌弘 教授と高知工科大学　情報学群　高田 喜朗 准教授にはは大変お世話になりました．福本先生には，発表前の審査用論文提出時に，私の論文中のおかしな表現，私の書いた論文の問題点を多数ご指摘いただきました．また，提出期限を過ぎてしまったにもかかわらず，様々なご意見を頂き，受け取っていただきました．お陰様で，私の書いた論文の問題をはっきりと認識することができ，その後の公開版論文の執筆を行うことができました．深く感謝致します．高田先生には，修士学位論文発表会にて，貴重なご意見を頂きました．お陰様で，公開版論文がより良いものとなりました．また，梗概と審査用論文の提出が遅れてしまったにもかかわらず，暖かく受け取っていただきました．深く感謝致します．

同研究室の皆様にも大変お世話になりました．私はとても皆様の手本となれるような人間

ではなく，むしろ反面教師として捉えていただいていた方も少なくなかったと思いますが，それでも面白おかしく交流していただき，私の研究室生活が楽しく充実したものとなりました．ありがとうございました．今後，卒業する同期の小池氏と 4 年生の皆様はそれぞれ新たな地にて社会人として歩み始めることと思いますが，本研究室で得た知識や教養を糧として，共に頑張っていきましょう．また，本研究室で残り 1 年間の研究活動を行われる修士 1 年の松尾氏，3 年生の方々にも，大変お世話になりました．松尾氏には，卒業研究において強力なサポート役となっていただきました．後輩が先輩のサポートをする，不可思議な構図となってしまいましたが，それでも嫌がることなく様々な面でサポートしていただけ，無事修士論文を書き上げることができました．深く感謝致します．3 年生の方々には，イベントの幹事や進行をしていただきました．お陰様で，楽しい研究室生活を満喫することができました．今後，来年度からまた新たな 3 年生が研究室に配属になりますが，その精神をぜひ受け継いでいただきたいと思います．今後の皆様のご活躍をお祈りするとともに，今までの交流に深く感謝致します.

そして，同研究室の諸先輩方，学士で卒業された同期の皆様，高知工科大学でお世話になりました皆様，高知県でお世話になりました皆様に，深く感謝致します.

最後に，学費や生活費など経済面と，精神面の二面から支え続けてくれた家族に心より感謝致します.

参考文献

[1] Natsuki Fujimori and Shinichi Yoshida, “Comparison of Machine Learning Algorithms for English and Japanese SPAM mail discrimination,” The 3rd International Workshop on Advanced Computational Intelligence and Intelligent Informatics, IWACIII 2013, Shanghai, China, October 18-21, ss1-10, 2013.

[2] Natsuki Fujimori and Shinichi Yoshida, “Machine Learning Algorithms applied to SPAM Filtering for English and Japanese E-Mail,” The 14th International Symposium on Advanced Intelligent Systems, ISIS2013, Daejeon, Korea, November 13-16, F3a-1, 2013.

[3] 平井有三，“はじめてのパターン認識，” 森北出版株式会社，2013.

[4] ローネン・フェルドマン， ジェイムズ・サンガー， 辻井潤一，“テキストマイニングハンドブック，” 東京電機大学出版局，2010.

[5] 金 明哲， ”R によるデータサイエンス，” p266，森北出版株式会社，2007.

[6] Leo Breiman, “Bagging Predictors,” Machine Learning, 24, pp.123-140, 1996.

[7] Leo Breiman, “Random Forests,” Machine Learning, 45, pp.5-23, 2001.

[8] Makoto Matsumoto and Takuji Nishimura, “Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator.” ACM Transactions on Modeling and Computer Simulation (TOMACS) 8.1 (1998), pp.3-30.

[9] 日本数学会編，“岩波　数学辞典　第 4 版，” 岩波書店，p.1560，2007.

[10] 乱数ライブラリー， “メルセンヌ・ツイスタ，” 統計数理研究所，
http://random.ism.ac.jp/info01/random_number_generation/node6.html.

[11] IM Sobol, “On the distribution of points in a cube and the approximate evaluation of integrals,” USSR Computational mathematics and mathematical physics 7, pp.86-112, 1967.

[12] George Marsaglia, “Xorshift RNGs,” Journal of Statistical Software 8(14), pp.1-6, 2003.

付録 A

$D_{\max}$ の各値における識別精度

表 A.1　$D_{\max} = 1$ における識別精度

$F_{\max}$	疑似乱数適用時	準乱数適用時	$N_{\max}$	疑似乱数適用時	準乱数適用時
1	63.5	63.3	1	64.0	64.7
2	65.6	63.7	2	65.4	64.8
3	66.1	64.1	3	67.9	65.4
4	67.2	64.3	4	67.9	64.7
5	67.2	64.8	5	67.9	65.1
6	68.6	65.1	6	68.7	65.2
7	68.7	65.5	7	68.9	65.7
8	70.3	65.7	8	68.7	65.1
9	70.0	66.5	9	69.0	64.3
10	66.5	71.2	10	69.9	64.9

表 A.2　$D_{\max} = 2$ における識別精度

$F_{\max}$	疑似乱数適用時	準乱数適用時	$N_{\max}$	疑似乱数適用時	準乱数適用時
1	63.5	65.9	1	66.5	65.6
2	65.3	67.0	2	66.6	67.1
3	66.3	68.3	3	66.4	68.4
4	66.9	68.8	4	67.2	69.3
5	67.5	69.0	5	68.1	68.7
6	67.8	69.4	6	67.2	69.2
7	68.0	70.0	7	67.0	69.3
8	68.6	70.0	8	67.6	70.3
9	68.4	70.2	9	67.2	70.1
10	70.2	68.8	10	67.1	70.5

付録 B

決定木の最大深度を 6 とした場合の優位回数と識別精度の比較

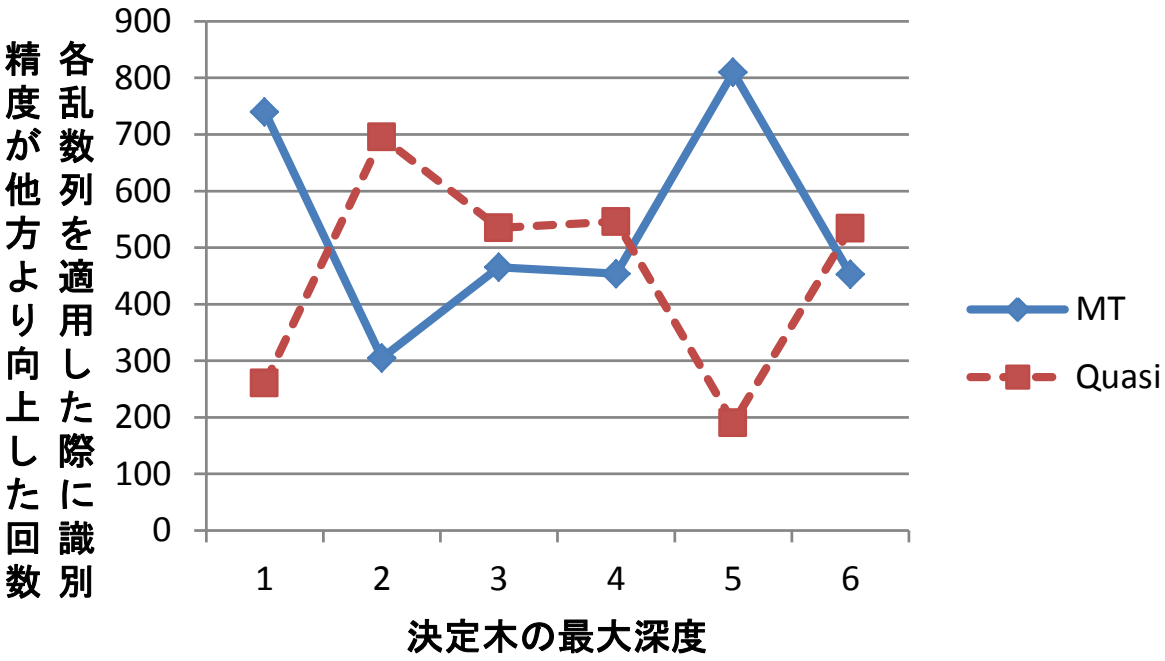

図 B.1　各手法の識別精度における優位回数の総和の推移

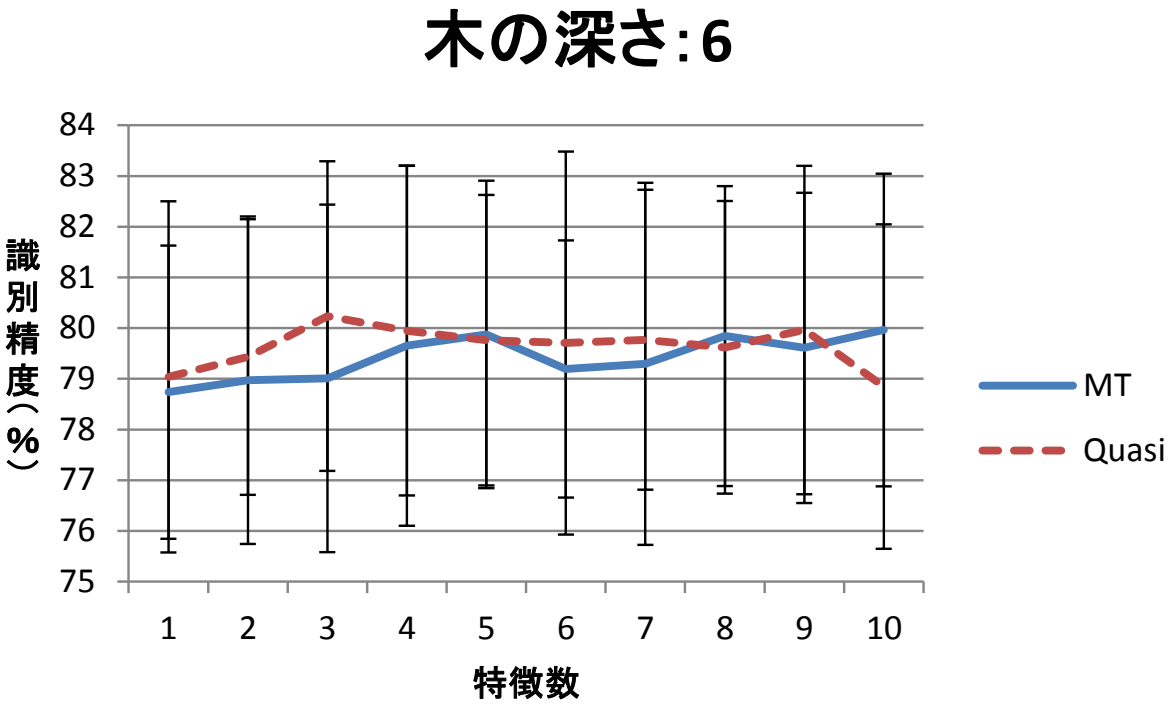

図 B.2　木の深さが 6 の時において選択特徴数に着目した際の各手法の識別精度の推移

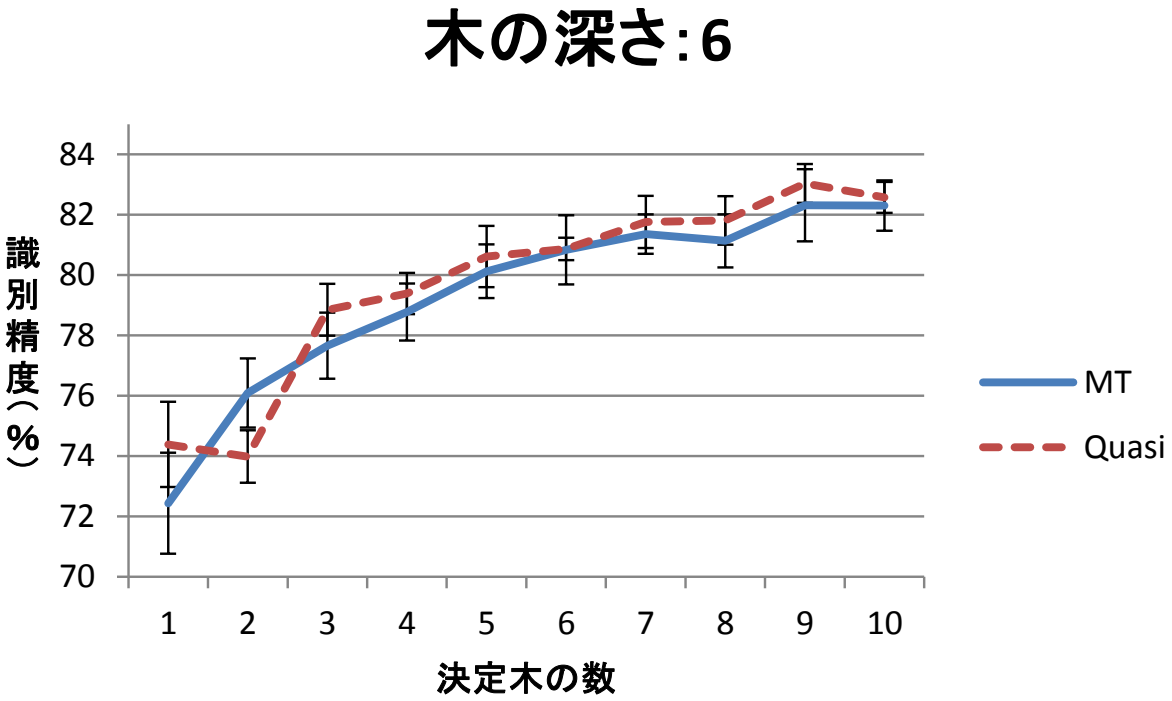

図 B.3　木の深さが 6 の時において決定木の数に着目した際の各手法の識別精度の推移