

決定木を用いた大貧民プログラムの分析に関する研究

小西 正人¹ 大久 保誠也² 若月 光夫¹ 西野 哲朗¹

概要: コンピュータ大貧民は大貧民をプレイするプレイヤープログラムを対戦させるもので、現在までに数多くのプレイヤープログラムが作成されている。強いプレイヤープログラムは、モンテカルロ法などの機械学習を用いているため、各場面でどの手を選択するかを予測することや、その行動原理を知るとは非常に難しい。そこで、本研究ではプレイヤープログラムの特徴を捉えることを目的として、ログに対して決定木分析によるデータマイニングを行うことで、プレイヤープログラムの特徴を抽出する方法を提案する。提案手法では、試合から得たログから処理時間などの3種類の視点に基づく決定木を生成することで、各プログラムの特徴を抽出する。また、提案手法の妥当性を明らかにするために、計算機実験を行った。まず、挙動が比較的明らかとなっている代表的な3種類のプレイヤープログラムの対戦ログから、提案手法により各プレイヤープログラムの特徴を抽出した。そして、抽出した特徴とプログラムの挙動を比較することで、各プレイヤープログラムの特徴をおおむね正しく抽出できることを確認した。加えて、挙動がまだ解析されていないプレイヤープログラムに対して分析を行い、いくつかの特徴を抽出した。

1. はじめに

ゲーム情報学の研究は経済学や心理学などの分野への応用が期待されており、多くのゲームが研究対象とされている。特に、多人数不完全情報ゲームではトランプゲームの大貧民を題材とした研究が行われている。コンピュータ大貧民は大貧民をプレイするプログラム（プレイヤープログラムと呼ぶ）を計算機上で対戦させるものである。その強さを競う大会として、電気通信大学 (UEC) において UEC コンピュータ大貧民大会 (UECda) が 2006 年度から毎年開催されており、そのプログラムは年々強くなってきている。UECda に出場した、強いプレイヤープログラムは、モンテカルロ法などの機械学習を行っている。一般に、機械学習を用いるプレイヤープログラムが、試合中の各場面でどの手を選択するかを予測することは難しい。そのため、クラスター分析を用いたプレイヤープログラムの分類など、プレイヤープログラムの分析を対象とした研究も行われている [1]。しかし、各プレイヤープログラムの行動原理の特徴を抽出することは非常に難しく、様々な課題が残されている。

本研究の目的は、大貧民をプレイするさまざまなプログラムの行動原理の特徴を捉えることである。そこで本研究では、コンピュータ大貧民の試合のログ（将棋で言う棋譜）を収集し、それらのログに対して決定木分析によるデータマイニングを行うことで、プレイヤープログラムの特徴を抽出する方法を提案する。

提案手法は、1 手分の処理時間、特定の札の出た状況、場に何も出ていない状態で出した役の種類の種類に注目した決定木の生成を行う。そして、提案手法が生成した決定木が正確にプログラムの特徴を示しているかを確認するため、挙動がある程度予測できる代表的なプレイヤープログラム (default, Nakanaka, snowl) に対して分析を行った。その結果、抽出した特徴はプレイヤープログラムの特徴をおおむね正しく表していることがわかった。また、挙動がまだ詳細には解析されていない、UECda-2015 の優勝プログラム (wisteria, kou2) に対しても同様の分析を行うことにより、幾つの特徴を抽出した。

2. コンピュータ大貧民のルール

本研究では、UECda-2015 におけるルールを使用した。これは UECda-2007 以降で使用されているものと同様であり、その詳細は以下のとおりである。

2.1 基本ルール

コンピュータ大貧民では 5 人のプレイヤー（クライアントとも言う）でゲームが行われる。使用するカードは、52 枚の数字のカードにジョーカー 1 枚を加えた計 53 枚のカード

¹ 電気通信大学大学院情報理工学研究所
Graduate School of Informatics and Engineering, The University of Electro-Communications

² 静岡県立大学経営情報学部
School of Management and Information, University of Shizuoka

ドであり、試合の開始時に各プレイヤーになるべく同じ枚数になるようにカードが配られる。カードの強さは弱い数字から順に、3, 4, 5, 6, 7, 8, 9, 10, J, Q, K, A, 2となる。配られたカードを手札とし、各プレイヤーは自分の番になったとき、可能であれば手札を提出する。手札を早く無くしたプレイヤーは勝ち抜けとなり、勝ち抜けたプレイヤーが4人になった時点で、1試合が終了する。

早く勝ち抜けた人から順に高い階級が与えられる。階級は、大富豪、富豪、平民、貧民、大貧民という順番に序列付けされる。階級が決められた後の試合では、試合の最初にカード交換が行われる。カード交換では、大貧民と大富豪との間で2枚、貧民と富豪との間で1枚カードを交換する。この時、大貧民と貧民は必ず強いカードを交換しなくては行けない。

2.2 手札の提出

カード提出時に出せる役は単騎、ペア(グループ)、階段の3種類ある。ここで、単騎はカード1枚のみの役、ペアは同じ数字のカードを複数枚同時に提出する役、階段は同じマークで数字の強さが連続するカードを3枚以上そろえた役である。また、カード提出時に役を出さずに、パスをすることができる。提出されたカードは、場が流れるまで場札として残る。主な場が流れる条件は、8切りの場合か、ゲームに参加しているプレイヤー全員が連続してパスした場合である。

場にカードが無い場合(この状態をleadという)は好きなカードを提出できるが、場にカードがある場合(この状態をfollowという)には、直前に出された場札と同じ役で、かつそれよりも強いものしか提出できない。

場には縛りおよび革命といった特別な状況が存在する。縛りは直前の場札と同じマークのカードが提出されると起こり、場が流れるまで同じマークのカードしか提出できなくなる。革命は、ペアなら4枚、階段なら5枚以上同時に提出されると起こり、カードの強さが逆転する。主な特殊なカードとしては、数字が8のカードとジョーカーがある。数字が8のカードがいずれかの役の一部として提出されると、強制的に場を流せる8切りが起こる。ジョーカーは、単騎で使用する場合、通常時でも革命時でも一番強いカードとして使える。ペアや階段で使用する場合にはジョーカーは、任意の数字のカードとして代用できる。その他の特殊なカードとしては、場札が単騎のジョーカーのときに提出することで場を流せるスペードの3や、手札交換後に手札にあれば最初にそのプレイヤーから提出できるダイヤの3がある。

3. 決定木

決定木とは、ノード(節点)の部分进行分类の説明、リーフ(葉)进行分类結果とする木構造の概念表記、もしくはその

概念表記を生成するデータマイニング手法のことである。決定木生成のアルゴリズムは、分析元のデータをばらつきが無いように分ける。ばらつきの基準となる変数を目的変数、分割の基準となる変数を説明変数と呼ぶ。ばらつきの基準はジニ係数や、エントロピーなどを用いる。生成された木は、説明変数がカテゴリ変数の場合は決定木、連続変数の場合は回帰木と呼ばれる。決定木は予測と分類によく使われるが、得られた決定木はデータの分析元の法則を抽出しているものとみなされるため、この法則を知る目的にも決定木は利用される[4][5]。

また、麻雀や将棋などのコンピュータ大貧民以外のゲームについては、決定木を利用する研究が存在する[6][7]。

4. 提案手法

提案手法では、最初に分析対象となるプログラムの試合のログを収集し、その後、得られたログに対して決定木分析を行う。そして、得られた決定木を比較したり分析することにより、プログラムの特徴を抽出する。本節では、ログの収集方法と決定木の生成について説明する。

4.1 ログの収集

決定木分析に必要なデータを取るため、解析対象とするプログラムのログを、コンピュータ大貧民の試合から収集する。この際、決定木分析の説明変数と目的変数に当たるものを要素としたログを収集する。ログの要素は以下の通りである。

- ゲーム数
通算の試合数。
- ターン
試合が始まってからの手数。
- 流れ数
場が流れてからの手数。
- 全体の札
プレイヤー全員の手札の残り枚数の合計。
- 自分の札
自身の手札の残り枚数。
- leadかfollowか
カードを提出した際の、場の状況。
- 役の種類
提出する役の種類。normal(単騎), group(ペア), sequence(階段), pass(パス), follow(場にカードがある状態で役を出したとき)の5種類と、役に8のカードを含む場合を特別に表記した8+normal, 8+group, 8+sequence, 8+follow。
- ジョーカーを持っている
手札にジョーカーを持っているか否か。データでは、持っていたら1が入り、無ければ0が入る。
- 数字が8のカードを持っている

手札に数字の 8 が含まれるカードを持っているか否か。データでは、持っていたら 1 が入り、無ければ 0 が入る。

- グループを持っている
手札にジョーカーを使わずにペア (グループ) として出せるカードが揃っているか否か。データでは、持っていたら 1 が入り、無ければ 0 が入る。
- 階段を持っている
手札にジョーカーを使わずに階段として出せるカードが揃っているか否か。データでは、持っていたら 1 が入り、無ければ 0 が入る。
- 1 手分の処理時間
1 手分の処理時間。順番が回ってきたことをサーバーがクライアントへ通信した時から、サーバーがクライアントから提出カードを受け取った時までの間の時間。
- クライアント名
その行動をしたクライアントの名前。

4.2 決定木生成

収集したログの要素をすべて利用して決定木を生成するのではなく、どのような分析をするかの視点を定め、その視点にあった要素を用いて決定木の生成を行う。したがって、視点により、明らかとなる特徴や説明変数、目的変数などは異なってくる。また、決定木を生成する際に利用する基準は、エントロピーを用いている。

本研究では、以下の 3 つの視点による決定木を提案する。なお、視点ならびに要素の選択は著者らの経験と試行錯誤により決定した。

分析 1 : 1 手分の処理時間に基づいた分析

プログラムの戦略は処理時間に影響を与える。そこで、1 つのプログラムのログを対象に処理時間を目的変数、処理時間の状況に関わりそうな要素を説明変数として回帰木を生成する。

- 説明変数
ゲーム数, ターン, 流れ数, 全体の札, 自分の札, ジョーカーを持っている, 8 を持っている, lead か follow か
- 目的変数
1 手分の処理時間

分析 2 : 特定の札の出た状況に基づいた分析

特定の札に注目してみると、プログラムによってその札の使い方は異なる。そこで、比較する複数のプログラムのログを対象に、特定の札の使用方法に注目した決定木を生成する。この分析で使う特定の札は、特別な札であるジョーカーと数字が 8 のカードである。プログラムがどのように特定の札を使うのかが重要なため、プログラム名を目的変数に、札の使い方に関わりそうな要素を説明変数と

する。

- 説明変数
全体の札, 自分の札, lead か follow か, 役の種類
- 目的変数
クライアント名

分析 3 : lead 時に提出する役の種類に基づいた分析

プログラムは戦略を考慮した結果として役を出す。そこで、1 つのプログラムのログを対象として役に注目し、決定木を生成する。提出した役の種類を目的変数として、場にも出ていない場合の戦略に関連しそうな要素を説明変数として決定木を生成する。

- 説明変数
ターン数, 全体の札, 自分の札, ジョーカーを持っているか, グループを持っているか, 階段を持っているか
- 目的変数
役の種類 (normal, group, sequence のみ)

5. 計算機実験 1

計算機実験により、提案手法の妥当性の評価を行った。具体的には、その挙動が比較的明らかとなっている 3 つのプレイヤープログラムのログを収集し、そこから各決定木を生成した後、得られた決定木から読み取れる特徴がプログラムの挙動と一致するかを検証した。

本実験でのログの収集と決定木の生成は、次のように行った。コンピューター大貧民のサーバープログラムに、試合の内容を記録するログを出力するプログラムを追加した。そして、同じプログラム同士での試合を行い、そのログを収集した。決定木の生成には、統計解析とその解析結果をグラフィカルに表すためのプログラム言語である R の、rpart を用いた [8][9]。R には決定木を生成する際、ノードの分割方法を指定できるが、本実験では情報量を指定した。決定木の剪定は R の決定木自身の枝刈りの他、同じ条件でとった別のログから新たな決定木を生成し、それらの比較に基づいて行った。また、決定木の作図は R の partykit を利用した [10]。

5.1 分析するプログラム

5.1.1 default

標準的な動作のみを行うプログラムであり、アルゴリズムは単純である。本実験では、UECda-2015 で使用された default プログラム (ver0.21) を使用した。

- 場に何も無い場合
階段, ペア, 単騎の順番でなるべく枚数が多い役を提出する。提出できるカードがある限り必ずカードを提出する。
- 場に提出されたカードがある場合
場が単騎の場合は階段とペアを崩さないように、崩さ

ないカードがある場合は一番弱い役を必ず出す。なお、この崩さないようにしている階段とペアはジョーカーを含まない。場がペア、階段の場合は、ジョーカーなしで役を探し、無ければジョーカーありで役を探し。役が見つかったならば、その中で一番弱い役を出す。

5.1.2 Nakanaka

Nakanaka は藤田悟氏によって作られた、UECda-2011の参加クライアントである。また、UECda-2015のライト級の基準プログラムでもある。人間が大貧民をプレイする際の戦略を参考にしたヒューリスティックなプログラムであり、23個の戦略を組み合わせている。主な特徴としては以下の通りである。

- 終盤は約3手から必勝手を探索する。
- 手札の強さの平均が高くなるような手を優先する。
- 縛りと階段を、積極的に行おうとする。

5.1.3 snowl

snowl は須藤郁弥氏によって作られた、UECda-2010の優勝プログラムである。UECda-2015での無差別級の基準クライアントである。snowlはコンピュータ大貧民のモンテカルロ法を使用している代表的なプログラムである。

探索により必勝手が見つければその手を選び、見つからない場合はモンテカルロ法による行動選択を行う [3]。モンテカルロ法による行動選択とは、とり得る各行動に対して試合終了まで複数回のシミュレーションを行い、得た勝ち点の平均が最も良かったものを選択することで行う。また、snowlは相手の手札を推定するシミュレーション時の手札の割り振りの確率を重みづけして学習している。

このように、シミュレーションで行動を決めているため、プログラムのソースコードから、snowlの試合時の具体的な行動を予測するのは難しい。しかし、多くの研究で使用されている [1] ため、その挙動はある程度明らかになっている。

5.2 実験結果

分析1により生成された回帰木を図1と図2に示す。ここで、図1がNakanaka、図2がsnowlのものである。なお、分析1でのdefaultの回帰木は生成できなかった。

分析2により生成された決定木を図3と図4に示す。ここで、図3がジョーカーの使われ方の、図4が数字が8のカードの決定木である。葉の部分は、左から、default、Nakanaka、snowlとなっている。

分析3により生成された決定木を図5、図6、および図7に示す。ここで、図5がdefault、図6がNakanaka、図7がsnowlである。葉の部分は、左から、グループ、単騎、階段となっている。

これらの決定木から、次のような特徴が読み取れる。

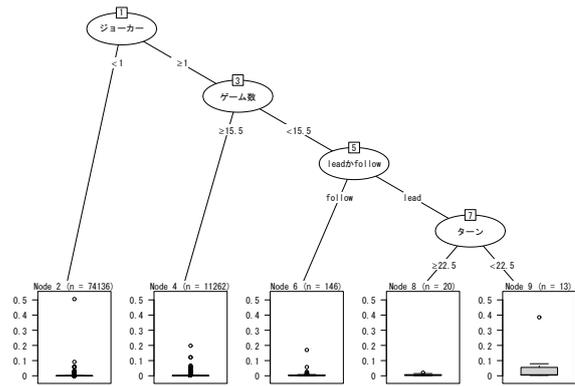


図1 分析1のNakanakaの回帰木

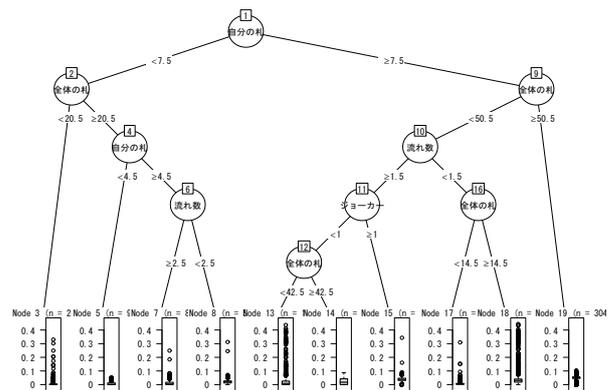


図2 分析1のsnowlの回帰木

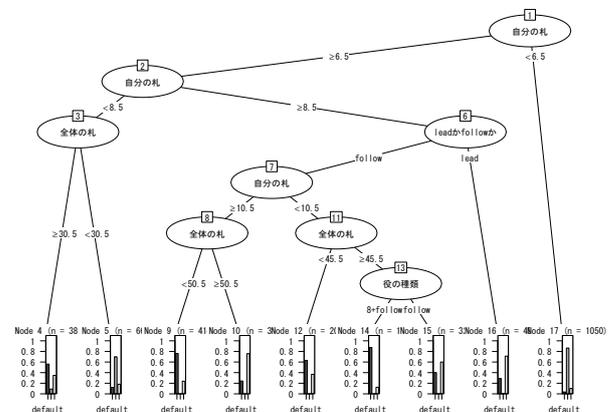


図3 分析2のジョーカーの使われ方に対する決定木

分析1

defaultの決定木が生成されなかったということは、目的変数の値は、各説明変数の値に依存せず、一定ということである。つまり、defaultはどんな状況でも1手分の思考時間は変わらない。図1の最初の分岐より、Nakanakaは主にジョーカーの有無で思考時間が変わる。図2において、自分の札数や全体の札数による分岐が多いことから、snowlは札の数で思考時間が変わる。また、札が少なければ思考時間が短く、札が多ければ思考時間が長い。

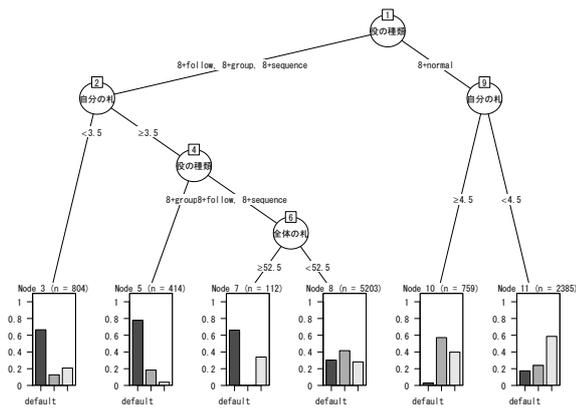


図 4 分析 2 の数字が 8 のカードの使いわれ方に対する決定木

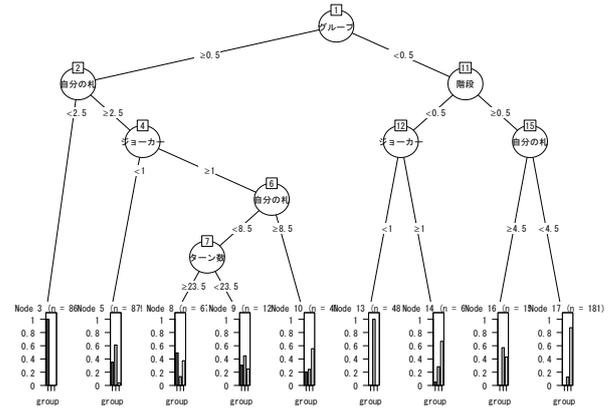


図 7 分析 3 の snowl の決定木

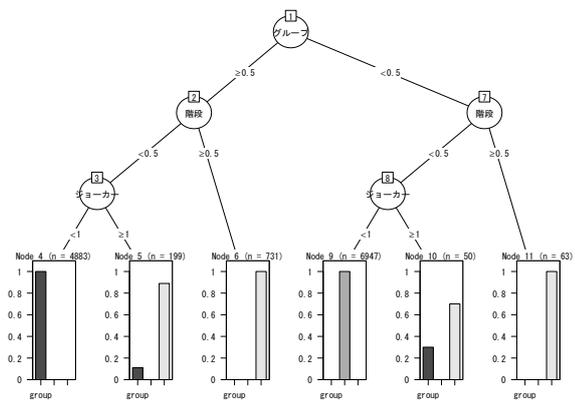


図 5 分析 3 の default の決定木

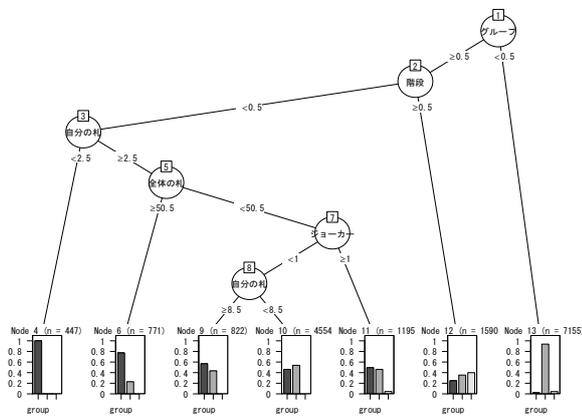


図 6 分析 3 の Nakanaka の決定木

分析 2

図 3 より、自分の札が 9 枚以上の序盤では Nakanaka は絶対にジョーカーを出さない。また、Nakanaka は終盤にジョーカーを提出しやすく、default と snowl は序盤にジョーカーを提出しやすい。snowl と default を比較すると、序盤で場に何も札が出ていない場合は snowl はジョーカーを出しやすい。図 4 より、default は数字が 8 のカードを単騎で提出しにくい。snowl は手札が 3 枚以下のときに単騎で数字が 8 のカードを提出しやすい。

分析 3

図 5 より、default は手札に階段があれば階段を必ず提出する。また、手札に階段の役とジョーカーを持っておらず、グループの役を持っているならばグループを提出する。図 6 より、Nakanaka はグループの役が手札にない場合は、階段の有無で分岐せずに単騎を提出しやすい。グループの役が手札にあり、階段の役がない場合、自分の札が 2 枚以下ならペアを出す。図 7 より、snowl は手札にグループの役があったとしても、ジョーカーを持っていて、かつ自分の札の枚数が 9 枚以上のときは、階段を出しやすい。グループの役が手札にあり、階段の役がない場合、自分の札が 2 枚以下ならペアを出す。手札にグループはないが階段があり、かつ自分の札が 4 枚以下の時にペアを出す。

5.3 考察

提案手法により抽出された特徴と、各プログラムのアルゴリズムを比較することにより、抽出された特徴の妥当性を検討する。

5.3.1 分析 1

default

提案手法により、どんな状況でも 1 手分の試行時間は変わらないという特徴が抽出された。default はプログラム内で、非常に単純な処理しか行っておらず、どんな場面でも、ほとんど処理時間を必要としない。つまり、時間の差がつくほどの処理を行っていない。したがって、抽出された特徴は妥当である。

Nakanaka

提案手法より、主にジョーカーの有無で分岐が行われるという特徴が抽出された。ジョーカーは複数枚の役で提出する場合、任意のカードとして使用できる。Nakanaka はジョーカーを除外して考える戦略が多いが、それでも必勝手を考えるときなどにジョーカーを考慮に入れるので、ジョーカーの影響は大きいと考えられる。したがって、抽出された特徴は妥当である。

snowl

提案手法より、全体の札や自分の札の多さでの分岐が多いという特徴が抽出された。また、札が少ないほど思考時間は短く、多いほど思考時間は長いという特徴も抽出された。snowl は試合中に取り得る行動に対して一定数のシミュレーションを行っており、試合が進むほど1つのシミュレーションの時間は短くなる。ゲームの進み具合は、各プレイヤーの自分の持ち札に関係があるため、このような特徴が出たと考えられる。

5.3.2 分析 2

default

提案手法より、default は序盤にジョーカーを出しやすいという特徴と、数字が8のカードを単騎で提出しにくいという特徴が抽出された。default のアルゴリズムでは、場に何か出ている場合、出せる役が手札にあれば必ず出すアルゴリズムである。そのため、場に提出されているカードが単騎の場合は、最後にジョーカーを出す。場に何も出ている場合は、階段、ペア、単騎の順番で枚数が多い役を提出しようとするため、その時にジョーカーと一緒に提出する。また、単騎の優先度が低いため、数字が8のカードを単騎で提出しにくい。これらのことより、抽出された特徴は妥当である。

Nakanaka

提案手法より、Nakanaka は自分の札が9枚以上の序盤では絶対にジョーカーを出さず、終盤にジョーカーを提出しやすいという特徴が抽出された。そもそもNakanaka のアルゴリズムでは、終盤までジョーカーを出さないようになっているが、革命を起こす際にはジョーカーを出す可能性がある。Nakanaka が革命を起こす場合、残りの手札の強さの平均がその役を出す前の平均より高くなくてはならないという条件がある。そのため、初期手札である自分の札が9枚以上の時には、強いカードも含まれていることも多いため、手札の平均を下げないため革命をしない。よって、抽出された特徴は妥当である。

snowl

提案手法より、序盤にジョーカーを提出しやすいという特徴と、数字の8のカードが自分の札4枚以下で出しやすいという特徴を抽出した。snowl はモンテカルロ法によるプログラムのため、明示的にソースコードに書かれてはいないが、実際に、snowl の挙動を観察すると、確かに序盤にジョーカーを提出していた。また、数字の8のカードについては、必勝手で数字が8のカードを提出していると考えられる。よって、抽出された特徴は妥当である。

5.3.3 分析 3

default

提案手法より、default の決定木は綺麗に目的変数が分割されて作成された。大きな特徴は、場に何も出ている状態では、グループの有無と階段の有無で分岐するという特

徴である。これは default のプログラムの主要な提出手決定プロセスを示しているといえる。また、ジョーカーの有無でも分岐が行われている。これは、ジョーカーがあることによって、階段とペアの選択肢が広がっているためだと考えられる。よって、抽出された特徴は妥当である。

Nakanaka

提案手法より、Nakanaka でもグループの有無と階段で分かれているが、グループが無い場合は階段の有無では分岐せずに単騎を出しやすいという特徴を抽出した。この理由は明確にはわからないが、グループよりも早く階段を出すためと推察される。また、Nakanaka はそもそも場に何も出ている状態では階段を出しにくいクライアントだとわかる。

また、グループがある場合で階段がない時の自分の札が2枚以下の時にペアを出すという特徴が抽出された。これは当たり前のことかもしれないが、この分岐が現れたということは最後に必勝手を探索しているということが言えるのではないかと推察され、正しく特徴を抽出されたかと推察できる。

snowl

提案手法より、snowl でも最初はグループで分岐をしているが、グループが無い場合は階段の有無で分岐をし、ある場合は階段の分岐は出てこないという特徴を抽出した。また、グループがあり、ジョーカーがある場合の自分の札が9枚以上のとき、階段を出しやすいという特徴も抽出した。この条件のときはゲームの序盤であり、序盤でジョーカーがあれば階段を出しやすいということが分かる。

他の特徴としては、Nakanaka と同様なグループがある場合で階段がない時の自分の札が2枚以下の時にペアを出すという特徴、グループがなく階段がある場合の自分の札が4枚以下ならほとんど階段を出すという特徴が抽出された。これらは必勝手のことであると推察され、正しく特徴を抽出されたかと推察できる。

6. 計算機実験 2

この実験の目的は、挙動がまだ詳細には解析されていない2種類のプレイヤープログラムに対して提案手法を適応することによって、そのプログラムの特徴を知ることである。

6.1 分析するプログラム

UECda-2015 の優勝クライアントで分析を行った [11]。UECda-2015 では、クライアントの実行時間から、無差別級とライト級の2つの部門に分かれて対戦が行われた。

wisteria

wisteria は大渡勝己氏によって作られた、UECda-2015 の無差別級の優勝クライアントである。snowl と同じようにモンテカルロ法を使っているほか、様々な工夫が組み

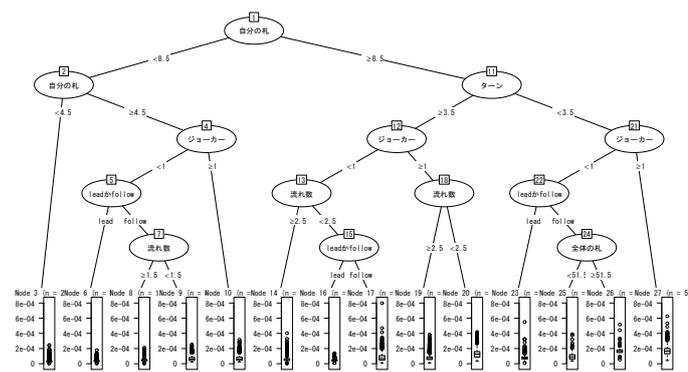


図 8 分析 1 の kou2 の回帰木

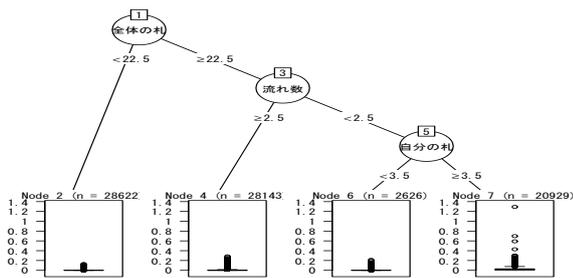


図 9 分析 1 の wisteria の回帰木

込まれている。wisteria についての発表などはあまり行われておらず、また解析もされていない。本実験で使用したバージョンは、「wisteria151101」である。

kou2

kou2 は田頭幸三氏によって作られた、UECda-2015 のライト級の優勝クライアントである。kou2 は、UECda-2014 のライト級の優勝クライアントである kou を発展させたプログラムである [12]。Nakanaka と同様にヒューリスティックなクライアントである。しかし、単純な条件分岐を行っているわけではなく、評価値を割り振るといった方法を行っているため、挙動は分かりづらい。

6.2 実験結果

分析 1 により生成された回帰木を図 8 と図 9 に示す。ここで、図 8 が kou2、図 9 が wisteria のものである。

分析 2 により生成された回帰木を図 10 と図 11 に示す。ここで、図 10 がジョーカーに対する決定木、図 11 が数字が 8 のカードに対する決定木である。なお、目的変数は kou2、wisteria の他に default, Nakanaka, snowl も加えた。それぞれの決定木の葉は、左から default, kou2, Nakanaka, snowl, wisteria である。

分析 3 により生成された回帰木を図 12 と図 13 に示す。ここで、図 12 が kou2、図 13 が wisteria のものである。それぞれの決定木の葉は、左からグループ、単騎、階段である。

これらの決定木から、次のような特徴が読み取れる。

図 8 より、kou2 は、かなり複雑な条件で思考時間が変

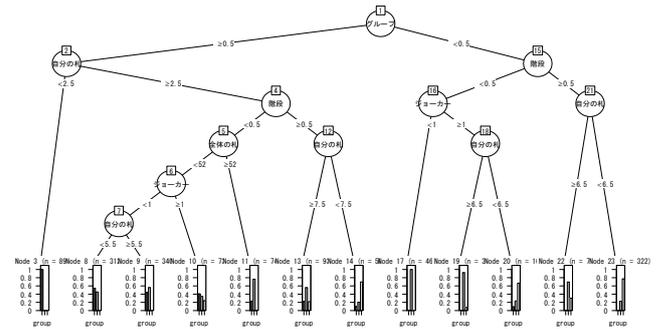


図 12 分析 2 の数字の 8 のカードに対する決定木

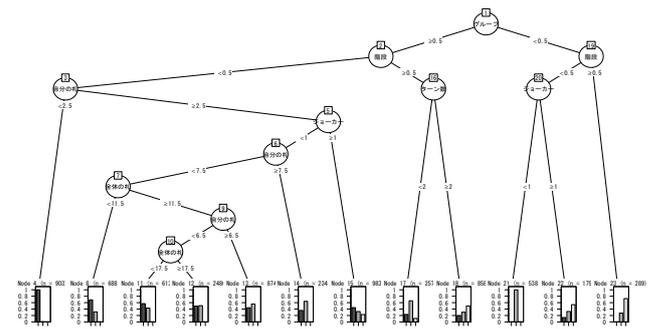


図 13 分析 2 の数字の 8 のカードに対する決定木

わっている。図 9 より、wisteria は、snowl のときに札の数で思考時間が変わっているわけでもなく、むしろどんな場面もほぼ同じである。これは、snowl はモンテカルロ法を回数で打ち切っているのに対し、wisteria は時間で打ち切っているのではないかと推察される。図 10 より、kou2 も wisteria もジョーカーを極端に偏った使い方をしていないことが分かる。また、どちらかという順序盤にはジョーカーを出さないことが分かる。図 11 より、wisteria は、自分の札 2,3 枚の時に 8 をペアで出しやすい。このことから、wisteria は 8 を必勝手として保持しやすいことが分かる。図 12 より、一番左の葉の自分の札が 2 枚以下の分岐より、kou2 も必勝手を探索していると分かる。また、一番右の葉より、kou2 は終盤に階段を出しやすいことが分かる。図 13 より、wisteria は、一番左の葉の自分の札が 2 枚以下の分岐より、必勝手を探索していると分かる。しかし、階段での必勝手の分岐がないため、snowl のように、階段での必勝手は少ないのではないと思われる。また、wisteria は、グループと階段がある時にターン数が 1 で分岐をしている。この時は試合の一番最初のときであり、その時点では wisteria は単騎を出しやすい。

7. おわりに

コンピュータ大貧民のログに対して決定木分析を用いることでプログラムの特徴を捉える手法を提案し、計算機実験ではその妥当性を確認した。加えて、挙動がまだ詳細には解析されていないプログラムに対して提案手法を適応

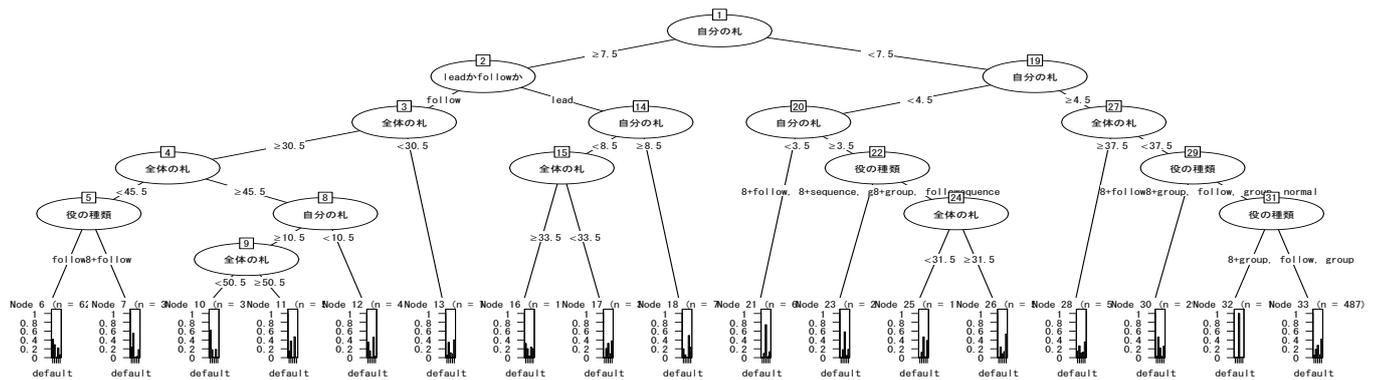


図 10 分析 2 のジョーカーの使われ方に対する決定木

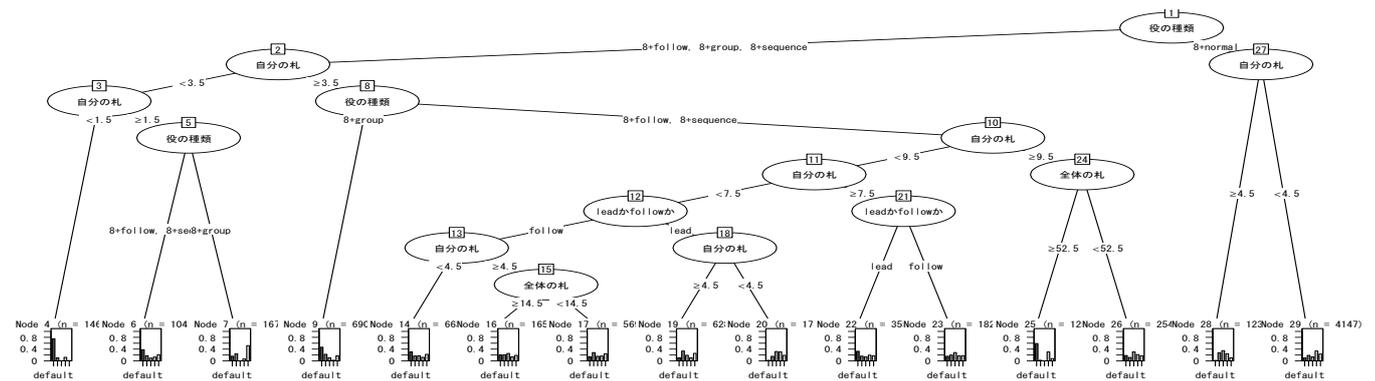


図 11 分析 2 の数字の 8 のカードに対する決定木

し、いくつかの特徴を明らかにした。

この手法によって、snowl などの機械学習を行っているようなクライアントの特徴を知ることが出来るようになった。また、内容が分からないような他の多くのクライアントに適応することで、それぞれの特徴をソースコードを読むことなく明らかにできると期待できる。

提案手法では、3 種類の異なる視点で決定木を生成したが、それぞれの決定木から違った特徴を見つけることができた。このことより、データの作り方や説明変数の決め方は重要であると考えられる。しかし、見つけられた特徴のほとんどは大まかな特徴であり、提案手法だけでは複雑な特徴は見つけづらい。今後の課題として、新しい視点の決定木の生成と、適切な説明変数での分析が挙げられる。特に適切な説明変数に関しては、主成分分析などの分析を行って説明変数を探す必要がある。また、適切な目的変数の選び方を手動で選ぶのではなく、自動的に適切な目的変数を抽出するシステムを構築することも重要であり、これも今後の課題である。

参考文献

[1] 綾部孝樹, 大久保誠也, 西野哲朗. 大貧民プログラムの n-gram 統計による特徴抽出とクラスタ分析. 情報処理学会研究報告. MPS, 数理モデル化と問題解決研究報告, Vol. 2013, No. 2, pp. 1-6, 2013.
[2] 電気通信大学. UEC コンピュータ大貧民大会 (UECdA).

<http://uecda.nishino-lab.jp/>.
[3] 須藤郁弥, 成澤和志, 篠原歩. UEC コンピュータ大貧民大会向けクライアント「snowl」の開発. 第 2 回 UEC コンピュータ大貧民シンポジウム講演予稿集, 2010.
[4] マイケル・J.A. ベリー, ゴードン・S. リノフ. データマイニング手法 - 営業, マーケティング, CRM のための顧客分析. 海文堂出版, 2 訂, 2005.10
[5] 酒巻 隆治, 里 洋平. ビジネス活用事例で学ぶ データサイエンス入門. SB クリエイティブ, 2014.6
[6] 田中悠, 池田心. 麻雀初級者のための状況に応じた着手モデル選択. 研究報告ゲーム情報学 (GI), 2014-GI-31(10), 1-8 (2014-03-10)
[7] 柳圭二郎, 柴原一友, 但馬康宏, 小谷善行. 将棋における決定木を用いた候補手の生成. ゲームプログラミングワークショップ 2006 論文集, 第 2006 巻, pp.163-166, 2006.11
[8] R: The r project for statistical computing. <https://www.r-project.org/>.
[9] rpart: Recursive partitioning and regression trees. <https://cran.r-project.org/web/packages/rpart/index.html>.
[10] partykit: A toolkit for recursive partitioning. <https://cran.r-project.org/web/packages/partykit/index.html>.
[11] 電気通信大学. UEC コンピュータ大貧民大会 (UECdA) 歴代優勝クライアント. <http://uecda.nishino-lab.jp/download/index.php>.
[12] 田頭幸三, 但馬康宏, 菊井玄一郎. 大貧民プログラムにおけるヒューリスティック戦略の評価. 研究報告ゲーム情報学 (GI), 2015-GI-34(9), 1-6 (2015-06-27), 2188-8736