

確率的な属性注目と反復消去：選択課題の汎用認知モデリング

Stochastic attention on attributes and iterated elimination: A generic cognitive modelling of selection tasks

犬童 健良[†]

Kenryo Indo

[†] 関東学園大学

Kanto Gakuen University

kindo@kanto-gakuen.ac.jp

概要

本研究では属性への確率的な注目と代替案の反復的消去を用いて意思決定者の認知プロセスをモデル化し、実際にコンピュータプログラムによるシミュレーション実験を行った。この結果、文脈効果やギャンブル比較のアノマリー例題の多くを基本的に一つのモデルで実験的に再現できる可能性があることが示された。

キーワード：確率的な属性への注目、反復消去、文脈効果、意思決定と推論のアノマリー

1. はじめに

選択肢がいくつかあって、その中から一つを選ぶような課題に対して、人々はどのような認知プロセスで候補を絞り込んだり、評価を下したりするのだろうか。意思決定者の認知プロセスの研究は、たんに選択結果を予測するだけではなく、意思決定者が環境のどの側面を注目したり注目しなかったりするか、またどのようにその最終判断に影響するのかを予測できる理論を必要とする。このような選択問題における属性への確率的な注目とそれによって代替案を絞り込む認知プロセスのことを、本論文では情報反応と呼ぶ。

より一般的な文脈においては、情報反応とは生体やソフトウェアロボットが環境の変化をシグナルとして察知し、実行できる行動を選んで適用できるかどうか素早く判断する状況適合を意味する。本研究では選択問題における情報反応を、次のような手続きの反復としてモデル化する。すなわち各時点である属性がランダムに注目され、その属性についての評価が許容されない、すなわちその属性にかんしてその時点で残っている他の代替案に対して何らメリットがないものを消去する。

また異なるジャンルのよく知られた例題に対して提案する手続きを適用する。条件文推論、消費者ブランド選択における文脈効果、ギャンブル選択（リスク選択）、時間選択の4領域にわたって文献で報告されたアノマリー現象が再現されるか実験的に確かめる。

```
Function 反復消去( 代替案リスト, 属性リスト )
  候補リスト=代替案リスト;
  注目履歴=[];
  While 候補リスト=[] is false
    注目属性=類似性に基づく確率選択(属性リスト)
    削除リスト=非許容(注目属性, 候補リスト)
    候補リスト=remove(候補リスト, 削除リスト)
    注目履歴=append(注目履歴, 注目属性:削除リスト)
  If 先頭要素(注目履歴).length=1
    Return 先頭要素(注目履歴)
  Else Return ランダム選択(先頭要素(注目履歴))
```

図1 確率的な反復消去アルゴリズム

本研究では、属性への確率的な注目に基づき選択する候補を絞り込むヒューリスティックな手続きを提案し、異なる複数の領域のサンプル例題に対して適用する。以下ではこれを反復消去アルゴリズムあるいは、たんに反復消去と呼ぶことにする。図1にその疑似コードを示す。サンプル例題は条件文推論[17]、消費者ブランド選択[2,6,9,10,12,14]、ギャンブル比較[1,3,4,7,8,15]、時間選択[11]という行動選択の4領域にわたる。

反復消去アルゴリズムへの入力は、与えられた選択問題の問題文章や図表から属性とその価値の組に変換された問題表現のデータである。提案する手法は多属性意思決定問題として述べられていない問題に対しても適用可能である。これについては、第2節で条件文推論の課題であるカード選択課題をモデル化する例を用いて説明したい。ここでは図1に示された手順を簡潔に説明しておこう。まず「候補リスト」にすべての属性がセットされ、そこから類似性を考慮して、ランダムに注目される属性が選ばれる。選択者が注目した属性は「注目履歴」に登録される。つぎに注目した属性と候補リストから、他の候補に比べて魅力のないものとして、非許容的な代替案を見つけて削除リストに追加する。候補リストから削除リストに挙げた候補を削除し、注目履歴に注目した属性とそれに基づいて

削除された候補を記録する。これを候補リストが空になるまで繰り返す。

上で述べた手続きは、要するに非許容な代替案を見つけて消去できるような局所的な決定力をもった属性および分割の基準点を、その都度ランダムに選択する。注目履歴の先頭要素は、反復消去の最終期手前まで残った候補たちであり、もし単一の候補であればこれを最終選択とし、複数であればその中からランダム選択する。付録 A および付録 B に反復消去アルゴリズムのプログラムを掲載した（開発言語：SWI Prolog for Windows 64 bit version 8.2.1）。これらを実行することによって確率的注目を伴う反復消去の下での選択比率の予測値が実験的に得られる。ただし便宜的に反復の深さを 15 回に制限した。Prolog のバックトラック機能によって反復消去を自然にシミュレーションすることができる。また選択される代替案の比率は、すべてのバックトラック分岐点において等確率を仮定したときの理論分布と一致する。これは確率論理プログラミングないし点過程の一種として扱おうが、本研究の分析結果に直接影響しないためこれ以上触れない。

ただし提案するアルゴリズムでは代替案間の類似性に依存したサンプリングの調整を行う。似ている代替案同士は似ていない代替案よりも頻繁に比較されるといふ経験的なルールである。属性ベクトル間のノルム d の一次変換に反比例して、付録 A の最後方にある `distribution_of_comparison` 述語のバックトラック回数を調整する。その頻度は具体的には次の式(1)にしたがう。

$$[\alpha/(\beta + \gamma d)] \quad (1)$$

ここで 3 つのパラメータは、標準的に $\alpha=2, \beta=0, \gamma=1$ とする。なお式 (1) は学習心理学や行動経済学においてマッチング法則として知られる。

以降の部分では、4 領域の行動選択問題に順次提案するモデルを適用する。第 2 節では推論課題の一種である Wason のカード選択課題におけるマッチングバイアスを再現する。第 3 節では消費者ブランド選択における文脈効果、いわゆるビッグスリー（魅力効果、妥協効果、類似効果）に対して現象の再現にとどまらず、実証的な選択比率を再現する。両節を通じて反復消去の挙動が具体的に理解されるはずである。第 4 節ではとくにアノマリーの種類が多く知られるギャンブル比較の領域でのサンプル例題を集中的に実験する。また時間選択問題についてもとりあげる。また第 5 節では関連研究にふれる。最後に第 6 節で結論を述べる。

2. 認知モデルとしての反復消去の解釈

本節では Wason のカード選択課題として知られる例題をとりあげ、回答者の認知プロセスに対応して、図 1 の反復消去アルゴリズムがどのように適用され、そのパフォーマンスを再現するのか説明してみよう。

Wason のカード選択課題[17]は次のような論理的推論の問題である。机の上に 4 枚のカードが並べられ、それぞれのカードの各面にはアルファベットの文字と数字が記入されている。たとえば見えている文字が「E」「K」「4」「7」であるとする。実験参加者は、カードをめくって「もしカードの片面が母音ならば、もう一方の面は偶数でなければならない」というルールが正しいどうかを確かめる。ただし、調べる必要のないカードを裏返さないよう指示される。論理的には、このルールは前提 A（母音）と結論 C（偶数）とを結ぶ条件命題 $A \rightarrow C$ 、つまり「もし A ならば C である」として理解され、その反例「A でありかつ C でない」を検出するためには、A のカードであるカード「E」および C の否定であるカード「7」を調べる必要がある。ところが、「7」を調べる回答者は極めて少なく、その正答率は 1 割程度とされる。誤答パタンの多くは母音の「E」のみか、偶数の「4」との組み合わせである。前提に一致するカードを選ぶ傾向は確証バイアスと呼ばれた。しかしルールが抽象的であっても、「もし A ならば C ではない」のように結論が否定表現されると正答率は大きく改善されることから、抽象的なルールでは前提と結論に直接対応するカードが選ばれやすいと考えられる。これはマッチングバイアスと呼ばれる。ちなみにマッチングバイアスは、意思決定心理学において選好逆転現象などのアノマリーを説明するために用いられた（尺度）両立可能性と呼ばれる仮説に対応している。

反復消去を適用すると、確証バイアスあるいはマッチングバイアスが再現される。アルゴリズムを適用する前に、カード選択課題の認知プロセスを計算的なメカニズムとして解釈しておこう。回答者は 1（調べる）または 0（調べない）を割り当てる。全部で 2 の 4 乗、8 通りの可能な回答パターンがあり、回答者はそのうちの一つを選ぶことで正解（利得 1）を得る。この問題を解くため、回答者の心の中に、適切なメッセージプロセス、あるいはメカニズムが構成される。すなわち、まず問題文からルールを理解し、カード表面に書かれた文字を判読し、ルールに対してその違反を調べる必

要性を評価するための情報を得る。そしてこの情報に反応して回答パターンが決定される。

ここで Prolog プログラムの使い方も含めて、反復消去がどのようにコンピュータで自動実行されるのかについて簡単に説明しておこう。付録の Prolog プログラムではカード選択問題における選択対象を次の 5 つの事実節として表現する。

```
data( wst( 2 ), [ 0, 0, 1, 1 ], 'K').
data( wst( 2 ), [ 1, 10, 0, 0 ], 'E').
data( wst( 2 ), [ 1, 0, 1, 0 ], '6').
data( wst( 2 ), [ 0, 10, 0, 1 ], '7').
data( wst( 2 ), [ 0, 0, 0, 0 ], '-').
```

上の各行, `data(wst(2), [S, T, U, V], X)`における S~V の 4 変数は属性ベクトル, 各次元は (母音, 偶数), (母音, 奇数), (子音, 偶数), (子音, 奇数) という 4 種類のカードの可能な状態に対応する。属性値はカードを調べたときの期待利得と解釈し, また変数 X は回答者の選択に対応する。最後の '-' はいずれのカードも選択しないケースである。各行はデータの単位としての事実節として Prolog システムに登録される。

ユーザーがプロンプト?- の後に目標節を入力することはクエリー (質問) と呼ばれる。Prolog システムは目標節を証明できる事実を見つけ出す。

```
?- P=wason_selection_task(2), simulate_choices( P, [W], S, K, N, V).
```

```
P = wason_selection_task(2),
W = S, S = '6',
K = 3552,
N = 20496,
V = 0.17330210772833723
```

セミコロン;を入力してバックトラックさせる。

```
P = wason_selection_task(2),
W = S, S = '7',
K = 2600,
N = 20496,
V = 0.12685402029664325 ;
P = wason_selection_task(2),
W = S, S = 'E',
K = 14344,
N = 20496,
V = 0.6998438719750195.
```

変数 W (=S), K, N, V の意味はそれぞれ, カードのオモテ面, 選択頻度, 試行回数, 選択率である。選択率は「E」が 70%, 「6」が 17%, 「7」が 13%,

「K」が 0%である。

同じ例題についての, より率直な問題表現としては, 属性値ベクトルをルール的前提と結論の成否に直接対応させる。

```
data( wst( 1 ), [ 0, 0 ], 'K').
data( wst( 1 ), [ 1, 1 ], 'E').
data( wst( 1 ), [ 1, 0 ], '6').
data( wst( 1 ), [ 0, 0 ], '7').
data( wst( 1 ), [ 0, 0 ], '-').
```

以下のクエリーを実行すると, クエリー末尾の fail でバックトラックが強制されて, k 番目の属性 `attr(k)` に注目したときの各代替案 (カード) ペア間の優劣比較がすべて書き出される。

```
?- P=wason_selection_task(1), compare_attribute( P, A, E, C ), nl, write( wst(1);A;E;C), fail.
```

```
wst(1);attr(1);eval(1,0);item(E>K)
wst(1);attr(1);eval(1,0);item(E>6)
wst(1);attr(1);eval(1,0);item(E>7)
wst(1);attr(1);eval(1,0);item(E>-)
wst(1);attr(2);eval(1,0);item(6>K)
wst(1);attr(2);eval(1,0);item(6>E)
wst(1);attr(2);eval(1,0);item(6>7)
wst(1);attr(2);eval(1,0);item(6>-)
false.
```

以下のクエリーを実行すると, 反復消去の系列が一つ生成される。

```
?- P = wason_selection_task(1), [ _ : [ S ] | _ ] = T,
elimination_wrt_attribute( P, T).
```

```
P = wason_selection_task(1),
S = 'K',
```

```
T = [attr(1):['K'], attr(2):['E'], attr(1):['6'], attr(1):['7'],
attr(1):[-]].
```

候補リスト T には, 反復消去のステップごとに注目された属性とそれによって消去された候補集合が, リストの左端に順次追加されていく。上の実行例では, 3 ステップ目までは 1 番目の属性が注目されて「E」と「K」が候補に残るが, 4 ステップ目で 2 番目の属性が注目されると「E」が消去されて, 「K」が最終的に選択される。すべての注目経路をバックトラックさせている。付録 A の `attr_trace` 述語を用いると全注目経路を収集できる。バックトラックさせると, 最終選択の比率は「E」と「6」が 29%, 「K」と「7」および無選択が 14%ずつとなる。

3. 文脈効果

マーケターは消費者に対して売れるブランド品（ターゲット）を提供する役割を担う。マーケターは、魅力効果を含めたビッグスリーと呼ばれる典型的な3タイプの文脈効果を熟知する必要がある。これらは消費者ブランド選択における伝統的な理論（Luce モデル）と矛盾することから、計量マーケティングや意思決定心理学においてその代替理論が提案された。

消費者ブランド選択は、より一般的に多属性意思決定問題の一種であり、代替案（ブランドアイテム）は複数の次元からなる属性値の組として評価されると仮定され、評価の高いものがより選択されやすいものと予測される。ビッグスリーと呼ばれる現象は、実験的に想定される以下のような状況である。ターゲットと拮抗する市場シェアを持つ競合品に対し、おとり商品（デコイ）を2次元の属性空間のある位置に投入すると、ターゲットの相対的魅力が増す。1) おとりを投入するターゲットから見て劣った位置に追加する、2) ターゲットを中間位置になるように両製品を結ぶ延長線におとりを追加する。3) おとりを競合品の近くに追加してそのシェアを奪わせる。それぞれ1は魅力効果、2は妥協効果（あるいは折衷効果）、3は類似効果と呼ばれる。

魅力効果の例における属性への注目と反復消去の系列を具体的に説明しよう。属性空間は値引の程度と品質の高さという2次元の値の組として表現され、消費者はこの情報に反応して選択を行うと仮定する。例として、例えば X: 値引率, Y: 品質とし、また3ブランドの評価値ペア (X, Y) がそれぞれ A: (2割値引, 中品質), B: (定価, 極上), C: (1割値引, 高品質) とする。マーケターは B をターゲットとして競合品 A よりも多く売りたい。消費者の参照点 a_0 が「0割引」とすると、時点 $t=1$ で「値引」が注目されたとき、おとり商品 C は消去され、A と B が残る。続いて $t=2$ でも再び「値引」が注目されたとすると A が最後に残る。しかし $t=2$ で「品質」が注目され、参照点が「高品質」とすると最後に B が残る。一方もし $t=1$ で「品質」が注目されると、残るのは B と C になり、 $t=2$ で「品質」あるいは「値引」のいずれが注目されても、B が選ばれる（図2も参照）。ちなみに B だけには劣った選択肢 C がある（つまり C は B に対して局所的に非許容的である）とき、B がより選ばれやすくなる。

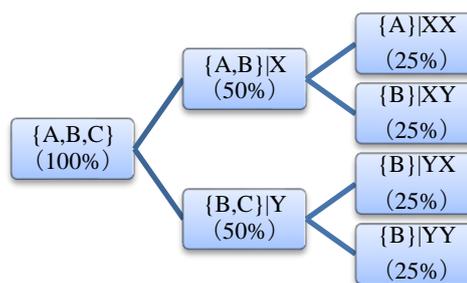


図2 魅力効果を再現する属性注列とその確率

図2は属性の注目順序（縦棒の右側）の下で残った代替案集合（縦棒の左側）を表す。注目される属性が等確率で選ばれると仮定すると、Aの選択確率は25%、Bは75%、Cは0%となり、魅力効果が再現される（図2参照）。類似効果は妥協効果と属性評価順序が同じであるため、順序だけでは再現できない。そこで代替案の類似性を反映してモデルを修正する。付録Dにこのアイデアを実行する Prolog プログラムを示した。反復消去はこのアイデアを精密化し、かつ文脈効果以外にも適用できるように一般化したものである。

表1 文脈効果のサンプルデータ

	ターゲット	競合品	おとり
魅力効果	4, 8	8, 4	2, 6
妥協効果	7, 4	5, 6	3, 8
類似効果	8, 3	3, 8	4, 7

表1に文脈効果の数値例を示す。また表2として付録Aと付録Bに掲載した実験用プログラムを用いて、文脈効果を再現した実験結果をまとめておく。

表2 シミュレーション実験の結果[†]

	ターゲット	競合品	おとり
魅力効果	68.2%	31.8%	0%
妥協効果	36.4%	31.8%	31.8%
類似効果	39.1%	26.1%	34.8%

[†]式(1)のパラメータは $\alpha = 2, \beta = 0, \gamma = 1$ とし、prologのバックトラックによって系統的サンプリングを行った。

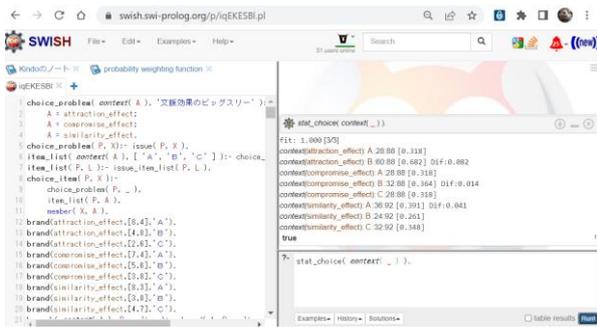


図3 ビッグスリーのシミュレーション実験

図3はWeb上のProlog処理系であるSWISH Prolog上で実際に文脈効果のシミュレーション実験を行った画面である

最後に、両極の代替案AとBを固定し、Cの位置を変化させたときに優勢になる代替案のフェイズダイアグラムを図4に示す(文献[2]の図13と比較されたい)。

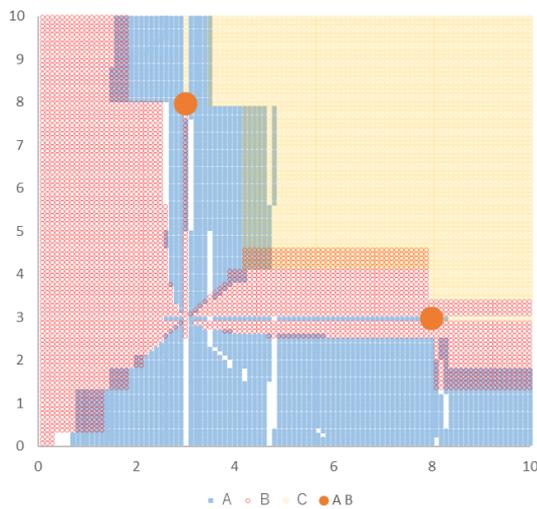


図4 おとりの追加位置と文脈効果

4. ギャンブル比較, 時間選択

本節では4領域からギャンブル比較に焦点を絞る、サンプル例題を集中的に採取する。ギャンブル比較では代替案はすべて可能な賞金金額とその確率を組み合わせたものである。実験では問題ごとに金額、確率、金額と確率の乗算(期待値)、参照点およびこれらの順序といった属性へのマッピングを工夫した。またいずれも二択問題であるので、参照点として非選択に相当する第3の代替案を設定した(詳細は付録Cを参照されたい)。

実験に用いるサンプル例題は以下の通りである。

- イ) common_ratio_effect : アレの背理[1]
- ロ) common_consequence_effect : アレの背理[1]
- ハ) reflection_effect : 鏡映効果[7]
- ニ) possibility_effect : 可能性効果[7]
- ホ) isolation_effect : 孤立効果[7]
- ヘ) ellsberg_paradox : エルズバークの背理[4]
- ト) branch_splitting_effect : 事象分岐効果[3]
- チ) preference_reversals : 選好逆転現象[8]
- リ) framing_effect : フレーミング効果[15]
- ス) time_preference : 現在バイアス[11]

最後の例題を除くといずれも人々の典型的な選択パターンが期待効用最大化の仮定に矛盾する。これらは従来それぞれ機序の異なるアノマリーとして論じられてきた。たとえばアレの背理(確実性効果と共通比による確率縮小あるいは共通結果削除による確実性効果の消失)、鏡映効果、可能性効果などは、独立性公理への違反だが、エルズバークの背理は確率のあいまい性を嫌う傾向、孤立効果は二段階クジにおける疑似的な確実性効果である。現在バイアスは、ギャンブル比較ではなく、報酬に時間遅れがある選択問題におけるアノマリーであり、指数割引と矛盾する(双曲割引を示す)が、その逸脱パターンはギャンブル比較のアノマリーのそれと対応する[11]。表3に反復除去アルゴリズムを用いた実験結果をまとめた。

表3 ギャンブル比較のアノマリーの実験結果

分類	問題番号識別名	代替案1	期待率1	確率1	確率2	代替案2	期待率2	確率2	代替案3	期待率3	A
イ	common_ratio_effect(1)	40/112	35.7%		72/112	64.3%	0.043				1.03
イ	common_ratio_effect(2)	40/80	50.0%	-0.1	40/80	50.0%					0.08
ロ	common_consequence_effect(1)	40/112	35.7%		72/112	64.3%	0.043				1.03
ロ	common_consequence_effect(2)	40/80	50.0%	-0.1	40/80	50.0%					0.08
ハ	reflection_effect(1)	72/112	64.3%	0.043	40/112	35.7%					1.03
ハ	reflection_effect(2)	40/112	35.7%		72/112	64.3%	0.043				1.03
ニ	possibility_effect(1)	40/96	41.7%		56/96	58.3%	-0.217				0.91
ニ	possibility_effect(2)	72/2108	3.4%	-0.566	2036/2108	96.6%					-1.04
ニ	possibility_effect(3)	40/112	35.7%		72/112	64.3%	-0.217				1.00
ニ	possibility_effect(4)	72/16108	0.4%	-0.566	16036/16108	99.6%					-1.15
ホ	isolation_effect(1)	80/120	66.7%	0.067	40/120	33.3%					1.04
ホ	isolation_effect(2)	40/108	37.0%		68/108	63.0%	0.03				1.03
ホ	isolation_effect(3)	40/108	37.0%		68/108	63.0%					1.00
ヘ	ellsberg_paradox(1)	72/156	46.2%	-0.138	84/156	53.8%					-0.53
ヘ	ellsberg_paradox(2)	60/144	41.7%		84/144	58.3%	-0.017				0.94
ヘ	ellsberg_paradox(3)	72/156	46.2%	-0.138	84/156	53.8%					-0.53
ト	preference_reversals(1)	40/144	27.8%		84/144	58.3%	-0.017		24/80	30.0%	0.94
ト	preference_reversals(2)	32/80	40.0%	-0.2	24/80	30.0%					0.73
チ	preference_reversals(3)	56/84	66.7%	0.067	28/84	33.3%					1.00
リ	branch_splitting_effect(1)	40/112	35.7%		72/112	64.3%	-0.087				1.00
リ	branch_splitting_effect(2)	112/152	73.7%	-0.203	40/152	26.3%					1.00
リ	branch_splitting_effect(3)	72/112	64.3%	-0.097	40/112	35.7%					1.00
リ	branch_splitting_effect(4)	8/24	33.3%		16/24	66.7%	0.047				1.01
リ	framing_effect(1)	84/124	67.7%	-0.043	40/124	32.3%					1.00
リ	framing_effect(2)	144/144	100.0%	0.22							1.00
ス	time_preference(1)	56/104	53.8%	0.138	24/104	23.1%			24/104	23.1%	2.61
ス	time_preference(2)	32/80	40.0%		24/80	30.0%	-0.3		24/80	30.0%	-0.63
ス	time_preference(3)	112/264	42.4%	-0.176	40/264	15.2%			112/264	42.4%	1.00
ス	time_preference(4)	40/152	26.3%		112/152	73.7%	-0.99				-0.99

表3の各行はそれぞれの例題のモデル識別名であり、各列に3代替案の相対的選択頻度(分数)とそのパーセントのデータが示される。空白部分はゼロ件であり、その代替案の選択データは生成されなかったことを示す。カッコ中の数値が1と2,または3と4の行ペアで代替案1と代替案2の間の選択率の大小が逆転するパターンがアノマリー,つまり理論モデルへの違反である。ニ以外で識別名のカッコ内が3や4である行は、属性マッピングを変更した同じ例題のモデルである。

また実験では文献を参考に再現目標となる選択率を設定した。典型的とされた回答については表3の乖離の列に実験で得られた相対選択率 δ から目標値 γ を引いた差 $\delta - \gamma$ を示した。表3末尾の列 λ は、選択率50%を帰無仮説としてベルヌーイ試行100回のときの片側累積確率の比、 $P(\delta + \text{非選択率} \times 0.5) / P(\gamma)$ である。 λ の大きさはモデルの適合の直観的な目安となる。 $\lambda = 1$ の近くは実証値がよく近似され、 $\lambda < 1$ のときはより帰無仮説に近く、 $\lambda > 1$ のときはバイアスをより強く予測する。 $\lambda < 0$ のときは選択傾向の予測に失敗している。表3からわかるように、 π (可能性効果) と ν (現在バイアス) の2と4、 η (エルズバーク背理) の1と3で λ が負であるが、選択変化の傾向は再現されている。また ν では多数派は時間整合という実証的事実と符合する。それ以外はおおむね良い。 π については確率が小さく反復回数が突出しており、類似度のカットオフ閾を0から0.01に変更することによって近似が改善する ($\lambda = 0.91, 1.03, 1.00, 1.00$)。

```
?- stat_choice(possibility_effect(_)).
fit: 1.000 [4/4]
possibility_effect(1);A:40/96 [0.417]
possibility_effect(1);B:56/96 [0.583] Dif:-0.217
possibility_effect(2);A:72/112 [0.643] Dif:0.043
possibility_effect(2);B:40/112 [0.357]
possibility_effect(3);A:40/112 [0.357]
possibility_effect(3);B:72/112 [0.643] Dif:-0.217
possibility_effect(4);A:72/112 [0.643] Dif:-0.087
possibility_effect(4);B:40/112 [0.357]
true.
```

類似度のカットオフは、いわば排斥項付の結合エネルギー (LJ ポテンシャル) の簡便な代用である。

5. 関連研究

確率的選択モデルは、意思決定論ではランダム効用とも呼ばれ、Luceによって公理化された。McFaddenによってマーケティング分野や交通工学へ応用されロジットモデルとして計量分析者の間でおなじみである。学習心理学において数学的に同等のモデルは古くはThurstoneによって、後にHerrnsteinらによってマッチング法則として論じられた。後者は時間割引のアノマリーである現在バイアスを再現する(双曲型割引)。それゆえ魅力効果はマッチング法則の違反でもある。これはブランドから感じる魅力は単純に代替案の評価値

に対する反応ではないことを示唆する。

確率的な属性への注目を通じて代替案を消去する多属性意思決定モデルとしては、Tverskyらによって提案された属性による消去 (elimination by aspects; EBA) が知られている[14]。EBAは文脈効果の研究の先駆けとなった。EBAでは情報反応 (属性へのランダム注目) にかんしてはマッチング法則 (Luceの公理) が成り立つ。本論文の提案するモデルは属性に確率的に注目して反復消去を行うことと、類似性に依存して比較頻度が修正されるという点では、EBAと発想を同じくする。しかしEBAでは類似効果を再現できるが魅力効果や妥協効果を再現できない。反復消去はビッグスリーをその実証比率まで正確に予測することができる。

Tversky, Sattath, and Slovic[16]は反復消去に似たヒューリスティック手続きを考察しており、またその心理的要因を検証している。2案の評がバランスするよう未定の属性値を埋める手続き (マッチング) と比べて、代替案を選択する手続きでは優先的に注目された属性が決定を左右する傾向が強まる。その結果として手続き不変性に違反する。これはプロミネンス仮説と呼ばれ、両立可能性の特殊ケースである。

反復消去アルゴリズムは、特定の属性に注目する傾向を持ったエージェントの集合体を考え、エージェントのランダム選択をメッセージとし、選択対象の集合を絞り込むような情報反応あるいはそれを引き起こす認知的なメカニズムとして解釈される。意識的な情報処理が省かれるという観点では、限定された合理性 (bounded rationality) のモデルを提供する。分散化された認識行動サイクルでは、エージェントの行動が、他のエージェントを含む環境の残りの部分へのメッセージとなり、情報反応の連鎖が引き起こされる。このような分権化 (decentralization) されたメカニズムは、ゲーム理論やマルチエージェントシステムのAI研究でよく研究されており、意思決定の認知モデル化にも応用されている。

ちなみに投票理論[13]において反復消去手続きは、ヘア(Hare)方式と呼ばれる。情報反応モデルは、その単純なバージョンとして、意思決定者の認知プロセスを単一のエージェントからなるメカニズムの下で説明するが、これをエージェントの集団による組織的メカニズム、あるいは「心の社会」へと自然に拡張することができる。ちなみにVAMPモデル[2]ではビッグスリーの再現のためSaariの投票方式を用いる。すでに述べたように、類似性を考慮したランダム生成を、推移性を保

つようにするだけでビッグスリーは再現される。つまり Condorcet 方式でも同様に可能である。安定結婚問題を解く遅延受入れ方式 (deferred acceptance), いわゆる DA アルゴリズム[5]はブロックペアを見つけて反復消去する。安定結婚問題における「男性」を「注目属性」, 「女性」を「消去候補」と読み替えることで, 反復消去の分権化されたバージョンが得られる。

6. 結論

本研究では属性への確率的な注目と代替案の反復的消去を用いて意思決定者の認知プロセスのモデル化を提案し, 4領域計 14 のアノマリーに対して実証的な回答傾向を予測できるか検証した。

反復消去アルゴリズムでは注目属性と消去候補がランダム選択される。類似性によるバックトラック頻度調整 (1式) 以外に心理学的なバイアスが入り込む余地はない。このことは次のような率直な疑問を投げかける。心理学者や行動経済学者らがバイアスと呼んでいるものの多くは, 本質的にランダムなノイズと区別できないのではないか。反復消去アルゴリズムは, これらの選択問題の領域ごとに考案されたあらゆる特殊モデルに対し, いわば洗練された帰無仮説を提起するという意味において認知モデルとしての汎用性をもつ。またそこからもう一つの疑問が呈される。すなわち認知プロセスの基本的なアルゴリズムは領域固有的に進化した集合体ではなく, 単一的であり, 問題タイプや行為の状況に応じて問題の表象の仕方が管理されているのではないか。

本研究で提案されたモデルはあくまで萌芽的であり, 予測の当てはまりは限定的である。消去において切断する集合に再注目を許すと若干の改善が得られるが, 計算効率が悪い。また限られた少数の例題に適用したにとどまる。例えばモンティホール問題のような確率判断の領域のアノマリーは本研究で論じられなかった。これについては読者の練習とされたい。またこれは属性値ベクトルのマッピングがアドホックであることと関連する。属性マッピングを系統化してより多くの例題に適用したとき, 精度だけでなく, 多様な表象方法の下でも予測が大きくぶれないことが期待される。この観点での表象安定性は, 規範的な意思決定モデルの手続き不変性と対比されよう。また, 提案されたモデルでは同一意思決定者内での一貫性と他の課題への転移可能性を論じていない。バックトラックの制限

(Prolog のカット演算のようなもの) を個人間に分散させる解釈は可能だが, 被験者間アンサンブルが等確率の事前分布に一致することを説明する必要が生じる。こうした課題を残しつつ, 反復消去による認知的メカニズムを探求する意義はあろうかと思われる。

文献

- [1] Allais, M. (1953) "Le comportement de l'homme rationnel devant le risque: critique des postulats et axiomes de l'école américaine", *Econometrica*, 21, 503-546.
- [2] Bergner, A. S., Oppenheimer, D. M., & Detre, G. (2019). VAMP (Voting Agent Model of Preferences): A computational model of individual multi-attribute choice. *Cognition*, 192, 103971. doi: 10.1016/j.cognition.2019.05.008
- [3] Birnbaum, M. H. (2008). New paradoxes of risky decision making. *Psychological Review*, 115(2), 463-501. doi: 10.1037/0033-295X.115.2.463
- [4] Ellsberg, D. (1961). Risk, ambiguity, and the Savage axioms. *The Quarterly Journal of Economics*, 75(4), 643-669. doi: 10.2307/1884324
- [5] Gale, D. and Shapley, L.S. (1962) . College admissions and stability of marriage, *The American Mathematical Monthly* 69, 9-15.
- [6] Huber, J. W., Payne, and C. Puto (1982). Adding asymmetrically dominated alternatives: violations of regularity and the similarity hypothesis, *Journal of Consumer Research*, 9 (1): 90-98.
- [7] Kahneman, D., and Tversky, A. (1979). Prospect theory: An analysis of decision under risk. *Econometrica*, 47, 263-291. doi: 10.1017/CBO9780511609220.014
- [8] Lichtenstein, S., & Slovic, P. (1971). Reversals of preference between bids and choices in gambling decisions. *Journal of Experimental Psychology*, 89(1), 46-55. doi: 10.1037/h0031207
- [9] Noguchi, T., & Stewart, N. (2018). Multialternative decision by sampling: A model of decision making constrained by process data. *Psychological Review*, 125(4), 512-544. doi:10.1037/rev000010
- [10] Roe, R. M., Busemeyer, J. R., & Townsend, J. T. (2001). Multialternative decision field theory: A dynamic connectionist model of decision making. *Psychological Review*, 108, 370-392.
- [11] Prelec, D., and Loewenstein, G. (1991). Decision making over time and under uncertainty: A common approach. *Management Science*, 37(7), 770-786. doi: 10.1287/mnsc.37.7.770
- [12] Simonson, I. and Tversky, A. (1992). Choice in context: tradeoff contrast and extremeness aversion. *Journal of Marketing Research*, 29: 281-295.
- [13] Taylor, A. D. (2005). *Social Choice and the Mathematics of Manipulation*. Cambridge University Press.
- [14] Tversky, A. (1972). Elimination by aspects: A theory of choice. *Psychological Review*, 79(4): 281-299.
- [15] Tversky, A., and Kahneman, D. (1981). The Framing of decisions and the psychology of choice. *Science*, 211 (4481): 453-58. doi: 10.1126/science.7455683
- [16] Tversky, A., Sattath, S., and Slovic, P. (1988). Contingent weighting in judgment and choice. *Psychological Review*, 95(3), 371-384. doi:10.1037/0033-295X.95.3.371
- [17] Wason, P. C. (1966). "Reasoning". In Foss, B. M. *New Horizons in Psychology*, Vol. 1, Harmondsworth: Penguin, pp.135-151.

付録 C. 選択問題の属性表現データ

```

%data( common_ratio_effect,[qSz, $x,$y,p$X],_).
data( common_ratio_effect( 1 ),[0, 400, 0, 80],'A').
data( common_ratio_effect( 1 ),[20, 300,300,100],'B').
data( common_ratio_effect( 1 ),[0, 0,0,0],'C').
data( common_ratio_effect( 2 ),[75, 400,0,20],'A').
data( common_ratio_effect( 2 ),[80, 300,0,25],'B').
data( common_ratio_effect( 2 ),[0, 0,0,0],'C').
%data( common_consequence_effect,[qSz,$x,$y,$z,p$X],_).
data( common_consequence_effect( 1 ),[0, 500,100, 0, 10],'A').
data( common_consequence_effect( 1 ),[1, 100,100,100,100],'B').
data( common_consequence_effect( 1 ),[0, 0,0,0,0],'C').
data( common_consequence_effect( 2 ),[89, 500,0,0,10],'A').
data( common_consequence_effect( 2 ),[90, 100,0,0,11],'B').
data( common_consequence_effect( 2 ),[0, 0,0,0,0],'C').
%data( reflection_effect,[$x, -p$X, $y, p$Y],_).
data( reflection_effect( 1 ),[-400, -80, 0, 20],'A').
data( reflection_effect( 1 ),[-300, -100, 0, 0],'B').
data( reflection_effect( 1 ),[-600, -100, 0, 0],'C').
data( reflection_effect( 2 ),[-400, -20, 0, 80],'A').
data( reflection_effect( 2 ),[-300, -25, 0, 75],'B').
data( reflection_effect( 2 ),[-400, -100, 0, 0],'C').
%data( possibility_effect,[$x, p$X, $y, p$Y],_).
data( possibility_effect( 1 ),[400, 0.80, 0, -0.20],'A').
data( possibility_effect( 1 ),[300, 1.00, 0, -0.00],'B').
data( possibility_effect( 1 ),[0, 0, 0, 0],'C').
data( possibility_effect( 2 ),[400, 0.008, 0, -0.992],'A').
data( possibility_effect( 2 ),[300, 0.010, 0, -0.990],'B').
data( possibility_effect( 2 ),[300, 0, 0, -1],'C').
%data( possibility_effect,[$x, p$X, $y, p$Y],_).
data( possibility_effect( 3 ),[6000, 0.45, 0, -0.55],'A').
data( possibility_effect( 3 ),[3000, 0.90, 0, -0.10],'B').
data( possibility_effect( 3 ),[0, 0, 0, 0],'C').
data( possibility_effect( 4 ),[6000, 0.001, 0, -0.999],'A').
data( possibility_effect( 4 ),[3000, 0.002, 0, -0.998],'B').
data( possibility_effect( 4 ),[3000, 0, 0, -1],'C').
%data( isolation_effect,[$x, $y1, p$Y1, $y2, q$Y2],_).
data( isolation_effect(1), [1000, 1000, 0.50, 0, 0.00],'A').
data( isolation_effect(1), [1000, 500, 1.00, 0, 0.50],'B').
data( isolation_effect(1), [0, 0, 0, 0, 0],'C').
%data( isolation_effect,[$x+y1, p$Y1, $x+y2, q$Y2],_).
data( isolation_effect(2), [1000+1000, 0.50, 1000+0, 0.00],'A').
data( isolation_effect(2), [1000+500, 1.00, 1000+0, 0.50],'B').
data( isolation_effect(2), [0, 0, 0, 0, 0],'C').
%data( isolation_effect,[$x+y1, p$Y1, $x+y2, q$Y2],_).
data( isolation_effect(3), [1000+1000, 0.50, 1000+0, 0.00],'A').
data( isolation_effect(3), [1000+500, 1.00, 1000+0, 0.50],'B').
data( isolation_effect(3), [1000+0, 0, 0, 0, 0],'C').
%data( ellsberg_paradox,[Pmax$, Pmin$, Sred, Syellow, Sblue],_).
data( ellsberg_paradox(1), [1/3, 1/3, 1000, 0, 0],'A').
data( ellsberg_paradox(1), [2/3, 0, 0, 1000, 0],'B').
data( ellsberg_paradox(1), [0, 0, 0, 0, 0],'C').
data( ellsberg_paradox(2), [1, 1/3, 1000, 0, 1000],'A').
data( ellsberg_paradox(2), [2/3, 2/3, 0, 1000, 1000],'B').
data( ellsberg_paradox(2), [0, 0, 0, 0, 0],'C').
%data( ellsberg_paradox,[P$1max/min, $1, Q$2max/min],_).
data( ellsberg_paradox(3), [1/3, 1/3, 1000, 1/3, 1],'A').
data( ellsberg_paradox(3), [2/3, 0, 1000, 2/3, 2/3],'B').
data( ellsberg_paradox(3), [0, 0, 0, 0, 0],'C').
data( ellsberg_paradox(4), [1/3, 1, 1000, 1/3, 1/3],'A').
data( ellsberg_paradox(4), [2/3, 2/3, 1000, 1/3, 2/3],'B').
data( ellsberg_paradox(4), [0, 0, 0, 0, 0],'C').
%data( framing_effect,[結果1, Pr$結果1, 結果2, Pr$結果2],_).
data( framing_effect( 1 ),[200, 1, 0, 0],'A').
data( framing_effect( 1 ),[600, 1/3, 0, 2/3],'B').
data( framing_effect( 1 ),[0, 0, 0, 0],'C').
data( framing_effect( 2 ),[-400, -1, 0, 0],'A').
data( framing_effect( 2 ),[0, 1/3, -600, -2/3],'B').
data( framing_effect( 2 ),[-400, -1, -600, -1],'C').
%data( branch_splitting_effect( 1 ),[$96,$90,$14,$12],_).
data( branch_splitting_effect( 1 ),[90,0.5,5],'A').
data( branch_splitting_effect( 1 ),[85,5,0,10],'B').
data( branch_splitting_effect( 1 ),[0,0,0,0],'C').
%data( branch_splitting_effect( 2 ),[$96,$90,$14,$12,$12],_).
data( branch_splitting_effect( 2 ),[85,5,0,5,5,0],'A').
data( branch_splitting_effect( 2 ),[85,0,5,0,5,5],'B').
data( branch_splitting_effect( 2 ),[0,0,0,0,0,0],'C').
%data( branch_splitting_effect( 3 ),[$100,$50,$7],_).
data( branch_splitting_effect( 3 ),[80,20,0],'A').
data( branch_splitting_effect( 3 ),[95,0,5],'B').
data( branch_splitting_effect( 3 ),[0,0,0,0],'C').
%data( branch_splitting_effect( 4 ),[$100,$100,$50,$50,%7],G1').
data( branch_splitting_effect( 4 ),[80,0,15,5,0],'A').
data( branch_splitting_effect( 4 ),[80,15,0,0,5],'B').
%data( preference_reversals,[$x, $y, p$X],_).
data( preference_reversals( 1 ),[4 * 99, -4 * 1],'A').
data( preference_reversals( 1 ),[16 * 33, -2 * 67],'B').
data( preference_reversals( 1 ),[0, 0, 0, 0],'C').
data( preference_reversals( 2 ),[4, -4, 99, 67],'A').
data( preference_reversals( 2 ),[16, -2, 33, 1],'B').
data( preference_reversals( 2 ),[0, 0, 0, 0],'C').
%つづ

```

```

%つづき
%data( time_preference,[delay$, $x],_).
data( time_preference( 1 ),[0, 1000],'A').
data( time_preference( 1 ),[-7, 2000],'B').
data( time_preference( 1 ),[0, 0],'C').
data( time_preference( 2 ),[-365, 1000],'A').
data( time_preference( 2 ),[-365-7, 2000],'B').
data( time_preference( 2 ),[0, 0],'C').
%data( time_preference,[reference, delay$, $x],_).
data( time_preference( 3 ),[0, 0, 1000],'A').
data( time_preference( 3 ),[0, -7, 2000],'B').
data( time_preference( 3 ),[0, 0, 0],'C').
data( time_preference( 4 ),[-365, 0, 1000],'A').
data( time_preference( 4 ),[-365, -7, 2000],'B').
data( time_preference( 4 ),[0, 0, 0],'C').

/*デモ実行
?- stat choice( wason_selection_task( _ ).
fit: 0.500 [24]
wason_selection_task(1);-:8192/59280 [0.138]
wason_selection_task(1); 6:17352/59280 [0.293] Dif:0.014
wason_selection_task(1); 7:8192/59280 [0.138]
wason_selection_task(1); E:17352/59280 [0.293] Dif:-0.196*
wason_selection_task(1); K:8192/59280 [0.138]
wason_selection_task(2); 6:3552/20496 [0.173] Dif:-0.106*
wason_selection_task(2); 7:2600/20496 [0.127]
wason_selection_task(2); E:14344/20496 [0.700] Dif:0.211
true.

```

注. 6とEの行尾の Dif 値は予め設定した目標値からの乖離である。この例題では Evans & Lynch(1973)の実験値より「E」は21/43, 「6」は12/43とした。*

付録 D. 文脈効果を再現する代替的方法

```

選択('A',P):- 順位('AC',P), 順位('CB',P), between( 1, 2, _).
選択('A',P):- 順位('AC',P), 順位('BC',P).
選択('B',P):- 順位('BC',P), 順位('CA',P), between( 1, 2, _).
選択('B',P):- 順位('AC',P), 順位('BC',P).
選択('C',P):- 順位('CB',P), 順位('CA',P), between( 1, 2, _).
サンプル発生( 百回中の度数, Id):- between(1, 百回中の度数, Id).
順位('AC',魅力効果):- サンプル発生( 100, _).
順位('CA',魅力効果):- サンプル発生( 0, _).
順位('BC',魅力効果):- サンプル発生( 50, _).
順位('CB',魅力効果):- サンプル発生( 50, _).
順位('BC',妥協効果):- サンプル発生( 50, _).
順位('CB',妥協効果):- サンプル発生( 50, _).
順位('CA',妥協効果):- サンプル発生( 45, _).
順位('AC',妥協効果):- サンプル発生( 55, _).
順位('BC',類似効果):- サンプル発生( 50, _).
順位('CB',類似効果):- サンプル発生( 50, _).
順位('CA',類似効果):- サンプル発生( 55, _).
順位('AC',類似効果):- サンプル発生( 45, _).
問題タイプ( 魅力効果, ターゲット('A')).
問題タイプ( 妥協効果, ターゲット('A')).
問題タイプ( 類似効果, ターゲット('B')).
実験( P, T, [ A, B, C ], N, [ RA, RB, RC ]):-
問題タイプ( P, T ),
findall( 1, 選択('A',P), L1 ),
findall( 1, 選択('B',P), L2 ),
findall( 1, 選択('C',P), L3 ),
length( L1, A ),
length( L2, B ),
length( L3, C ),
N is A + B + C,
RA is A / N,
RB is B / N,
RC is C / N.

/* デモ実行
?- 実験( P, T, D, N, R ), nl, write( P; T; N; D; R ), fail.
魅力効果:ターゲット(A):20000:[15000,5000,0]:[0.75,0.25,0]
妥協効果:ターゲット(A):20000:[8250,7250,4500]:[0.4125,0.3625,0.225]
類似効果:ターゲット(B):20000:[6750,7750,5500]:[0.3375,0.3875,0.275]
false
*/

```