人とコンピュータによる数の整列プロセスの時間的な共起分析による比較
Sort Process: Comparison between Humans and Computers by Temporal Co-occurrence Analysis

山口 琢†, 新美 礼彦‡, 大場 みち子‡
Taku Yamaguchi, Ayahiko Niimi, Michiko Oba
† フリー, ‡ 公立はこだて未来大学システム情報科学部
Independent Researcher, Faculty of Systems Information Science, Future University Hakodate
study.yamahige@gmail.com

概要

人とコンピュータで数の整列プロセスは似てるのか？9 個の数の整列プロセスを、移動対象の時間的な共起関係で分析した。まず 9 行 9 列の共起行列を目視で比較し、次に目視の結果をクラスタリングで機械的に再現できるか試みた。目視比較では、桁数が少ないとき人による整列は選択ソートに似ていた。クラスタリングでは、桁数が少ないとき人による整列はクラスターを作ったが、人による整列が選択ソートに似ているとは言えなかった。

キーワード：数の整列, 整列プロセスのクラスタリング, 時間的な共起分析, 時間的な共起行列, 整列アルゴリズム, 学習分析, 知的プロセスの分析

1. はじめに

論理的思考力への関心の高まりとともに、学習分析 (Learning Analytics) 研究では近年、プログラミングのプロセスを測定・分析する研究が増えている。

本稿では、それよりもシンプルな思考課題として数の整列 (並べ替え、sorting) を題材とし、整列プロセスを分析する。

2. 問い

人による整列のプロセスは、コンピュータによる整列とどこが似ているだろうか？このとき、どのような測定や分析を行うと「似ている」と判断できるだろうか？

コンピュータによる整列は、アルゴリズムが複数考案されてきた。人による整列プロセスが、それら既知のアルゴリズムによる整列プロセスと似ているなら、人の思考過程にはそのアルゴリズムと共通するものがあるかもしれない。そのようなアルゴリズムや比較・分析手法を見つければ、それを名義尺度として人の思考過程を外部から推定／評価する手がかりになると期待できる。

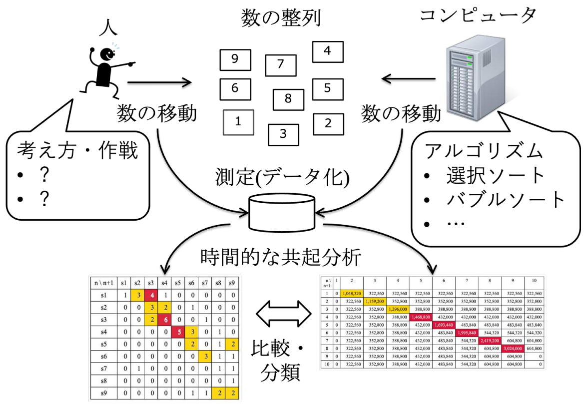

図 1　アプローチの概要

3. アプローチ

本稿では、整列処理を外部から観察して、数を移動する操作を記録することでプロセスの測定とし、整列の手順に着目して比較・分析する (図 1)。

3.1 整列プロセスの測定

整列の手順の測定では、整列プロセスで数の位置を移動するとき、移動対象の数と時刻を記録する。測定データは、移動対象となった数の時系列データである。

3.1.1 人による整列

人による整列を測定するには、並べ替えパズルの Web アプリケーション「ジグソー・テキスト」を使う [1](図 2、図 3)。ジグソー・テキストは、文章の順序を並べ替えるプロセスを測定して、ユーザの考え方や、人にとっての文章の意味を分析するツールとして開発した。文章に書く内容の順序が重要であることを

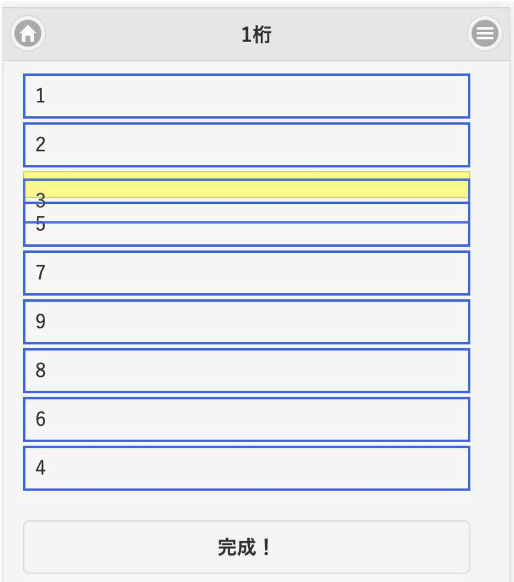

図 2　ジグソー・テキストは文章の並べ替えパズル・アプリケーション。ドラッグ&ドロップで並べ替える。これは数の整列問題。

学ぶためのツールとして、大学の授業や社会人向けセミナーでも使われてきた。

3.1.2　コンピュータによる整列

コンピュータによる整列を測定するには、一般に知られた整列アルゴリズム [2] を Python 言語で実装し、交換 (入れ替え、swap) やマージなど数を移動するタイミングで移動対象の数を出力する。アルゴリズムによって数を移動する方法が異なるので、どの数をどの順序で動かしたと見なすか、アルゴリズムごとに決める [3]。この決め方によって結果が異なることに留意する。

3.2　測定実験の実施

それぞれのデータがどのようにして取られたか説明する。

人による整列もコンピュータによる整列も共に、一部のデータとして既存の研究の測定データを利用する。そのため、共起行列 (次の節で説明する) を目視で比較する分析 (「3.4.1 時間的な共起行列を目視」) では、人による整列は 9 個の数、コンピュータによる整列は 10 個の数で、数の個数が異なる。この違いは影響しないと、現時点では考えている。

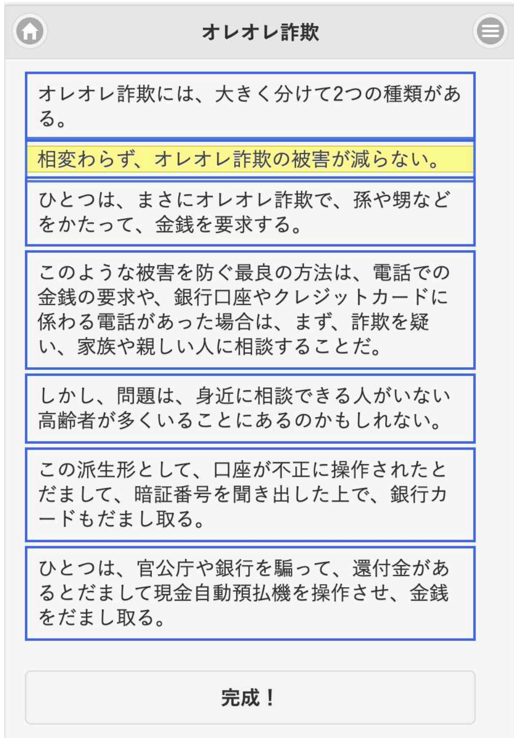

図 3　ジグソー・テキストは文章の並べ替えパズル・アプリケーション。ドラッグ&ドロップで並べ替える。これは文章の並べ替え問題。

3.2.1　人による整列

ジグソー・テキストを使って、1 桁から 5 桁の 9 個の数がシャッフルされたものを人が整列し、並べ替え操作を記録した。

9 個の 1 桁の数とは、例えば { 1, 2, 3, 4, 5, 6, 7, 8, 9 } である。9 個の 5 桁の数とは、例えば { 15743, 21789, 34785, 47932, 51845, 69430, 71463, 83046, 97324 } である。

対象者は、情報系の大学で情報システムや複雑系科学や知能システムを学ぶ、学部 2 年生 10 名であった。実験より前に「アルゴリズムとデータ構造」の講義があり、その中で整列を扱っていて、被験者の半数にとって必須科目であった。

このデータは本研究に先立って、人による数の整列プロセスと、人がプログラム・コードを並べ替えて完成させるプロセスとを比較した文献 [4] の研究で得られたものである。

3.2.2　コンピュータによる整列

コンピュータについては 2 種類の測定データを用意する。1 つは、1 から 10 の 10 個の数の全順列を問題

として整列を実行し、ソートアルゴリズムごとに共起行列 (次の節で説明する) を集計する。このデータは、コンピュータによる整列を人とは比較せずに分析した文献 [3] の研究で得られたものである。

もう 1 つは、1 から 9 の 9 個の数の列をシャッフルして 10 個の順列を問題として整列を実行し、個々の整列処理について共起行列を求める。

前者は共起行列を目視で比較するため (「3.4.1 時間的な共起行列を目視」) であり、後者は共起行列を階層的クラスタリングするため (「3.4.2 共起行列の階層的クラスタリング」) である。

3.3 測定データの分析手法

測定データの可視化および比較にあたって、操作対象の時間的な共起に着目する [5]。

時間的な共起に着目する理由は、テキスト分析における従来の共起分析と同様である。時間的／順序的に近くで頻繁に起きる、すなわち共起するなら、それら出来事の間には何らかの関係があると考えられるからである。

分析では、ある数を n 番目に動かした直後に (n+1 番目に) 別の数 (n 番目に動かした数自身も含む) を動かした回数を、操作対象のすべての数について集計して、数の一覧を昇順で縦横軸に設定した共起行列で表現する。共起頻度の標準偏差を求めて、平均+標準偏差、および平均+標準偏差*2 をしきい値として、セルの背景をそれぞれ黄 (薄い背景色)、赤 (濃い背景色に白抜き文字) で色分けする (図 4、図 5、図 6、図 7、図 8)。

これらの図では、s1〜s9 はアプリケーションが内部的に持っている数の ID で、本稿では s1、s2、…s9 の順に、数が昇順に割り当てられている、すなわち「s1 の数 <s2 の数」である。行列の縦軸が n 番目に動かした数、横軸が n+1 番目に動かした数である。例えば、図 4 では、s3 の数を動かした直後に s4 の数を動かした回数が 9 回あったことが分かる。

3.4 分析手順

次の 2 つの分析を行う。これら 2 つの方法のそれぞれについて、人のみのデータ、コンピュータのみのデータ、人とコンピュータの両方を含むデータの 3 種類でクラスタリングを行う。

n \ n+1	s1	s2	s3	s4	s5	s6	s7	s8	s9
s1	0	9	0	0	0	0	0	0	0
s2	0	1	10	0	0	0	0	0	0
s3	0	0	0	9	1	1	0	0	0
s4	0	0	1	0	5	1	2	0	0
s5	0	0	0	0	0	3	1	0	2
s6	0	0	0	0	0	0	2	2	0
s7	0	0	0	0	0	0	0	2	1
s8	0	1	0	0	0	0	0	0	1
s9	0	0	0	0	0	0	0	0	0

図 4 1 桁の数の人による整列の時間的な共起行列

3.4.1 時間的な共起行列を目視

まず、時間的な共起行列を目視で比較する。時間的な共起行列を使うこれまでの研究、例えばプログラミング・パズル「ジグソー・コード」で問題を解くプロセスを分析した研究 [6][7] などでは、共起頻度の大きい (>=平均+標準偏差*2) 順序対が共起行列中に現れるパターンを見つけるのが有効であった。

3.4.2 共起行列の階層的クラスタリング

次に、共起行列に対し階層的クラスタリングを行う。整列プロセスのログが似ているならば、アルゴリズム／思考過程が似ているのかもしれない。共起行列をクラスタリングの対象とすることで、整列の操作回数／手数の違いによらず時系列データを分析対象とできる。操作回数が多かったプロセスでは、共起行列のセルの値の合計値が大きくなる。

クラスタリングは 2 つの方法で行う。

方法の 1 つは、9 個の数の整列に関する 9 行 9 列の行列を 81 要素のベクトルとみなして、ユークリッド距離を使ってウォード法でデンドログラムを描く。

方法のもう 1 つは、対角線周辺の頻度を重視する。共起行列の各行のうち対角線、対角線の左隣 (あれば)、対角線の右隣 (あれば) を取り出した 25 要素のベクトルを抽出して、同様にデンドログラムを描く。対角線周辺を重視するのは、「3.4.1 時間的な共起行列を目視」の結果「4.1.1 人による整列」と、その考察「5.2 比較結果の考察」を踏まえたものである。

n\n+1	s1	s2	s3	s4	s5	s6	s7	s8	s9
s1	1	7	1	1	0	0	0	0	0
s2	0	0	5	0	0	2	1	1	1
s3	0	0	0	4	4	0	0	0	0
s4	1	1	1	0	2	2	0	1	0
s5	0	0	0	0	0	2	3	0	0
s6	0	0	1	1	0	0	1	0	1
s7	0	0	0	1	0	2	1	1	0
s8	0	1	0	1	0	1	0	1	1
s9	0	0	0	1	1	0	0	0	0

図 5　2 桁の数の人による整列の時間的な共起行列

n\n+1	s1	s2	s3	s4	s5	s6	s7	s8	s9
s1	2	4	2	1	1	1	0	0	0
s2	0	1	3	1	0	0	1	0	0
s3	0	1	0	6	0	0	1	0	0
s4	0	0	3	0	5	0	1	1	0
s5	0	0	0	0	1	5	1	0	0
s6	0	0	1	2	0	0	2	0	1
s7	0	0	0	0	2	2	0	0	1
s8	0	0	0	0	0	0	1	0	0
s9	0	0	0	0	0	0	1	1	1

図 6　3 桁の数の人による整列の時間的な共起行列

n\n+1	s1	s2	s3	s4	s5	s6	s7	s8	s9
s1	0	5	2	1	0	0	0	0	0
s2	0	0	3	2	1	0	0	0	0
s3	0	0	0	4	2	1	1	0	0
s4	0	0	0	0	6	1	0	0	1
s5	1	0	1	0	1	3	1	1	1
s6	0	0	0	0	0	0	2	0	1
s7	0	0	0	1	0	0	0	3	0
s8	0	0	0	0	0	1	1	0	0
s9	0	1	0	0	0	0	0	1	0

図 7　4 桁の数の人による整列の時間的な共起行列

n\n+1	s1	s2	s3	s4	s5	s6	s7	s8	s9
s1	1	3	4	1	0	0	0	0	0
s2	0	0	3	2	0	1	0	0	0
s3	0	0	2	6	0	0	1	0	0
s4	0	0	0	0	5	3	0	1	0
s5	0	0	0	0	0	2	0	1	2
s6	0	0	0	0	0	0	3	1	1
s7	0	1	0	0	0	0	0	1	1
s8	0	0	0	0	0	0	0	0	1
s9	0	0	0	0	0	1	1	2	2

図 8　5 桁の数の人による整列の時間的な共起行列

4.　結果

4.1　時間的な共起行列の目視

4.1.1　人による整列

人による整列では、数の桁数が少ない場合、時間的な共起行列は対角線から 1 つ右隣のセルにまとまる。桁数が多くなると他のセルに散らばる (図 4、図 5、図 6、図 7、図 8)。

4.1.2　コンピュータによる整列

コンピュータによる整列の時間的な共起行列を示す。

選択ソートでは、対角線から 1 つ右隣のセルにまとまる、すなわち次に 1 つ大きな数を動かすことが多い (図 9)。マージソートも選択ソートと同様だが、選択ソートほどにはまとまらない (図 10)。バブルソートは対角線上のセルに集中、すなわち同じ数を続けて動かすことが多い (図 11)。ヒープソートは、対角線から 1 つ左隣のセルに集中、すなわち次に 1 つ小さな数を動かすことが多い (図 12)。他の整列アルゴリズム、特にクイックソートでは、バブルソートのような極端な集中が見られない (図 13、図 14)。

4.2　共起行列の階層的クラスタリング

共起行列を階層的クラスタリングした結果を示す。

4.2.1　81 要素によるクラスタリング

共起行列の全セル 81 個を使ったクラスタリングの結果を示す。

図 15(本稿の最終ページに掲載) は、人とコンピュータとを合わせてクラスタリングした結果である。最後が”-1”になっている赤色のラベルが人による 1 桁の数の整列、”-2”となっている黄色のラベルが人による 2 桁の数の整列、”selection-”で始まる青色のラベルが選択ソートである。人による 1 桁の数の整列 (赤) がクラスタを作っている。このクラスタの近くに選択ソー

n\ n+1	1	2	3	4	5	6	7	8	9	10
1	0	1,048,320	322,560	322,560	322,560	322,560	322,560	322,560	322,560	322,560
2	0	322,560	1,159,200	352,800	352,800	352,800	352,800	352,800	352,800	352,800
3	0	322,560	352,800	1,296,000	388,800	388,800	388,800	388,800	388,800	388,800
4	0	322,560	352,800	388,800	1,468,800	432,000	432,000	432,000	432,000	432,000
5	0	322,560	352,800	388,800	432,000	1,693,440	483,840	483,840	483,840	483,840
6	0	322,560	352,800	388,800	432,000	483,840	1,995,840	544,320	544,320	544,320
7	0	322,560	352,800	388,800	432,000	483,840	544,320	2,419,200	604,800	604,800
8	0	322,560	352,800	388,800	432,000	483,840	544,320	604,800	3,024,000	604,800
9	0	322,560	352,800	388,800	432,000	483,840	544,320	604,800	604,800	0
10	0	322,560	352,800	388,800	432,000	483,840	544,320	604,800	604,800	0

図 9　選択ソートの共起行列: 選択ソートでは、対角線から 1 つ右隣のセルにまとまる、すなわち次に 1 つ大きな数を動かすことが多い

n\ n+1	1	2	3	4	5	6	7	8	9	10
1	952,560	1,337,490	990,045	724,455	525,450	377,820	252,090	176,370	92,505	0
2	544,530	771,120	1,289,610	939,960	670,705	466,536	297,978	178,730	93,925	0
3	556,815	431,130	627,480	1,239,750	886,175	611,404	385,965	209,150	94,975	0
4	562,665	433,560	334,470	502,200	1,185,585	826,260	530,001	281,718	109,899	0
5	556,125	429,930	328,600	251,985	383,400	1,123,136	742,560	409,456	152,408	0
6	529,395	412,548	315,422	237,779	179,456	264,960	1,034,640	604,320	235,680	0
7	472,830	372,246	286,254	214,534	156,896	110,160	155,520	877,200	372,360	0
8	374,940	298,380	231,348	173,460	124,512	80,160	53,520	60,480	574,560	0
9	222,390	178,830	139,890	105,234	74,736	43,440	26,760	10,080	0	0
10	0	0	0	0	0	0	0	0	0	0

図 10　マージ・ソートの共起行列: マージソートも選択ソートと同様だが、選択ソートほどにはまとまらない

ト (青) のクラスタがある。また、バブルソートやヒープソートが明確にクラスタを作っている。

4.2.2　対角線系の 25 要素によるクラスタリング

対角線を重視して共起行列から抜き出した 25 個の要素のクラスタリング結果を示す。

人とコンピュータとを合わせてクラスタリングした結果では、人による 1 桁の数の整列 (赤色) がクラスタを作っている。このクラスタに選択ソート (青色) が一部入っているが多くはない (図 16、本稿の最終ページに掲載)。

n\ n+1	1	2	3	4	5	6	7	8	9	10
1	13,063,680	595,200	670,880	616,896	507,360	382,848	263,520	158,400	70,560	0
2	1,128,960	10,160,640	661,178	709,932	646,340	515,808	365,760	222,720	99,360	0
3	529,200	1,411,200	7,620,480	910,176	737,500	616,288	448,416	276,384	123,600	0
4	272,880	730,800	1,411,200	5,443,200	1,286,880	684,688	505,368	314,664	141,312	0
5	154,920	395,280	806,400	1,209,600	3,628,800	1,645,168	524,316	328,216	148,644	0
6	92,784	224,208	464,352	745,920	887,040	2,177,280	1,791,420	300,068	138,728	0
7	55,758	129,192	267,408	446,400	569,520	524,160	1,088,640	1,574,748	98,378	0
8	31,310	70,758	146,136	249,600	331,920	327,600	201,600	362,880	953,498	0
9	13,699	30,714	63,660	110,400	150,120	152,880	100,800	0	0	0
10	0	0	0	0	0	0	0	0	0	0

図 11　バブル・ソートの共起行列: バブルソートは対角線上のセルに集中、すなわち同じ数を続けて動かすことが多い

n\ n+1	1	2	3	4	5	6	7	8	9	10
1	3,938,756	2,908,320	3,908,456	3,505,804	2,414,256	2,213,716	1,630,812	2,159,600	1,103,232	1,233,976
2	6,360,968	1,728,344	2,661,264	2,016,332	2,219,548	1,787,904	1,656,784	1,537,396	1,412,516	1,386,012
3	6,411,964	6,324,196	695,640	622,256	1,083,552	1,165,272	1,295,364	1,059,988	1,421,492	1,406,268
4	3,982,384	3,593,440	5,625,328	390,768	409,000	645,736	848,116	725,644	1,267,180	1,320,916
5	2,505,680	2,451,816	2,708,272	5,911,952	185,088	283,208	522,440	507,368	1,007,816	1,176,104
6	1,983,464	1,853,640	1,791,968	2,055,544	6,042,056	67,200	250,292	363,092	704,852	1,023,092
7	1,582,248	1,495,800	1,409,064	1,449,136	1,738,564	6,004,612	17,280	267,552	456,192	915,552
8	954,864	1,212,352	1,248,080	1,206,900	1,263,408	1,609,152	5,972,352	0	302,286	907,086
9	532,392	640,392	709,352	749,112	825,678	1,085,046	1,646,046	6,273,006	0	401,056
10	392,112	492,912	589,632	681,352	772,648	876,256	1,005,856	1,199,056	4,513,696	0

図 12　ヒープ・ソートの共起行列: ヒープソートは、対角線から 1 つ左隣のセルに集中、すなわち次に 1 つ小さな数を動かすことが多い

n\ n+1	1	2	3	4	5	6	7	8	9	10
1	533,400	404,520	664,220	709,792	627,770	569,382	521,569	430,344	377,547	334,896
2	55,512	330,792	393,940	591,884	643,412	612,126	572,400	465,422	401,321	347,040
3	191,408	154,180	323,848	428,540	628,932	711,788	687,245	544,890	455,513	368,040
4	581,762	343,798	204,750	253,388	402,562	605,533	713,512	586,496	481,105	358,200
5	420,714	410,502	341,188	212,100	209,086	391,977	626,804	651,199	531,405	359,352
6	470,138	433,990	415,608	336,266	134,819	192,840	498,395	674,131	647,392	414,012
7	483,562	426,628	462,432	439,514	273,830	157,665	234,206	573,350	683,807	466,740
8	419,436	394,330	426,916	452,942	391,489	288,112	128,844	199,219	482,283	478,560
9	410,208	370,108	391,506	425,446	401,518	407,129	257,670	87,482	105,982	217,680
10	429,644	381,696	384,936	402,712	354,246	445,546	396,119	226,613	73,238	0

図 13　クイック・ソートの共起行列: 他の整列アルゴリズム、特にクイックソートでは、バブルソートのような極端な集中が見られない

5.　考察

可視化の手法として時間的な共起行列は整列プロセスの特徴を捉えていると考えられる。比較の手法として時間的な共起行列を使うには、さらなる研究が必要である。

人とコンピュータの比較では、共起行列を目視すると、桁数が少ないときの整列では人は選択ソートと似ている。クラスタリングでは、桁数が少ないとき人による整列はクラスターを作ったが、人による整列が選択ソートに似ているとは言えなかった。

n\ n+1	1	2	3	4	5	6	7	8	9	10
1	0	410,553	405,513	398,313	387,873	372,321	348,075	307,179	226,980	0
2	410,553	0	396,853	390,373	380,773	366,205	343,117	303,591	225,023	0
3	396,853	396,853	0	381,196	372,556	359,116	337,360	299,416	222,740	0
4	381,196	381,196	381,196	0	362,604	350,508	330,348	294,312	219,936	0
5	362,604	362,604	362,604	362,604	0	339,384	321,240	287,640	216,240	0
6	339,384	339,384	339,384	339,384	339,384	0	308,280	278,040	210,840	0
7	308,280	308,280	308,280	308,280	308,280	308,280	0	262,080	201,600	0
8	262,080	262,080	262,080	262,080	262,080	262,080	262,080	0	181,440	0
9	181,440	181,440	181,440	181,440	181,440	181,440	181,440	181,440	0	0
10	0	0	0	0	0	0	0	0	0	0

図 14　挿入ソートの共起行列: 他の整列アルゴリズム、特にクイックソートでは、バブルソートのような極端な集中が見られない

5.1 時間的な共起分析の考察

時間的な共起行列は、それぞれの整列プロセスの、ある特徴を表していると言える。

共起行列の見出しは数の昇順に上から下、左から右に並んでいる。人による整列で桁数が少ない場合には、共起行列の対角線の 1 つ右隣のセルに高い頻度が現れる。ある数の次にそれより 1 つ大きい数を動かすことが多いということで、これは、小さい数から順番に動かしていることに対応していると考えられる。

コンピュータによる整列の共起行列は、それぞれのアルゴリズムから理解しやすいものもある。選択ソートは、まず全体から最小のものを見つけて先頭 (1 番目) と交換し、次に 2 番目以降から最小のものを見つけて 2 番目と交換し、…を繰り返す。すると、最小の数から順に、1 つずつ大きい数を動かすことが多くなる。すなわち、”1”の次に”2”、”2”の次に”3”、…を動かすことが多くなる。”1”の次に”2”を動かさない場合があって、それは、”1”の隣に”2”があった場合である。”1”の次に”2”がある場合の数は、ない場合の数よりも小さいので、選択ソートの共起行列は図 9 のようになる。

5.2 比較結果の考察

人とコンピュータとの共起行列の目視比較では、桁数が少ないときの人による整列は、(こちらは桁数に関係ないが) 選択ソートによる整列に似ている。人による整列では、桁数が少ないと対角線の 1 つ右隣に高い頻度が現れ、小さい数から順番に動かしていることに対応している。コンピュータによる整列では、選択ソートに同様の傾向がある。これは選択ソートのアルゴリズムからも理解できる。これは同時に、整列プロセスの特徴が対角線周辺に現れる可能性を示唆している。

このことは、整列プロセスの特徴が対角線周辺に現れる可能性を示唆している。

一方、クラスタリングの結果は、対角線重視を支持しているとは言い難い。81 要素でのクラスタリングでは、「桁数が少ないときの人による整列は似ている」とは言えそうである。しかし、「桁数が少ないときの人による整列は、選択ソートによる整列に似ている」と言うのは難しい。これは、対角線周辺のセルに絞った 25 要素でのクラスタリングでも同様であった。

「桁数が少ないときの人による整列」は、今回のクラスタリング結果の範囲でも何らかの特徴を持っていると期待できる。

「桁数が少ないときの人による整列は、選択ソートによる整列に似ているか？」については、目視判断とクラスタリング結果が不一致であった。「対角線周辺を重視する」ことが、25 要素というセルの選び方では不十分なのか、距離やクラスタリング方法の選び方の問題なのか、今後の研究が必要である。

5.3 その他

選択ソートと似ているからといって、桁数が少ない場合に人が選択ソートのアルゴリズムで整列しているとは限らない。これは体重が同じでも別人であることと同様である。例えば、頭の中で完成した整列結果をジグソー・テキスト上に出力しているだけなら、外から観察できたのはその「出力」プロセスであり、頭の中で行われた整列プロセスではない。

本稿は、時間的な共起行列を採用し共起頻度の大小に着目した。他の分析手法によって異なる結果が出るならば、それは別の観点による新たな知見として期待する。

謝辞

渡邉雄之介氏には、人による整列プロセスを測定したデータを提供していただいた。本研究は JSPS 科研費 17K01085 の助成を受けたものである。

文献

[1] 山口琢, 小林龍生, 高橋慈子, 大場みち子, パズル操作の測定・分析による思考の推定, 日本認知科学会 第 35 回大会 sP2-17, 2018
[2] 奥村晴彦, ［改訂新版］C 言語による標準アルゴリズム事典, 技術評論社, 2018
[3] 山口琢, 大場みち子, コンピューターの整列処理におけるデータ操作の時間的共起分析, 情報処理学会 研究報告コンピュータと教育 (CE), 2020-CE-156(1), 2020
[4] 渡邉雄之介, 大場みち子, 中村陽太, 山口琢, 異種パズル操作の時系列データ分析, 情報処理学会 第 82 回全国大会講演論文集, 2021
[5] 山口琢, 大場みち子, 編集操作の時間的共起分析の提案, 情報処理学会 研究報告コンピュータと教育 (CE), 2019-CE-151(9), 2019
[6] 中村陽太, 大場みち子, 山口琢, 伊藤恵, 学習進度に対応するパズルを利用したプログラミング思考過程の分析, 情報処理学会 研究報告コンピュータと教育 (CE), 2019-CE-151(1), 2019
[7] 藤井沙苗, 松澤芳昭, パズル型問題を利用したプログラミング初学者の理解度と思考過程の分析, 情報処理学会 第 82 回全国大会講演論文集, 2021

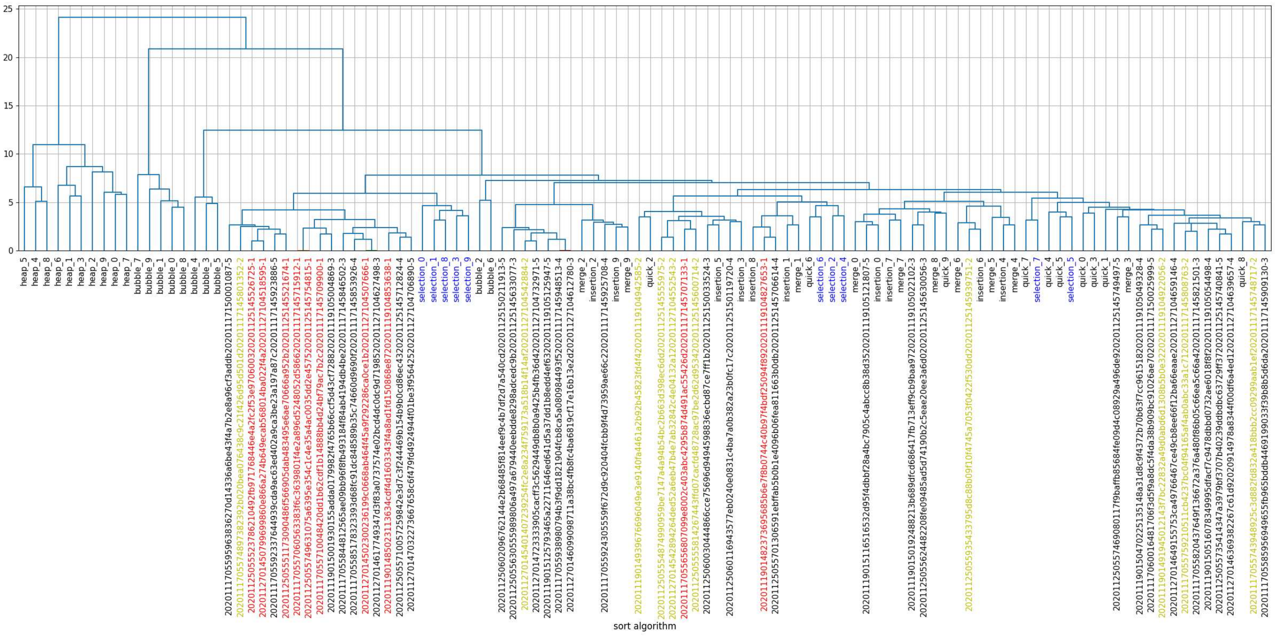

図 15　共起行列の全セル 81 個を使ったクラスタリングの結果。人とコンピュータによる整列を合わせたもののデンドログラム。人による 1 桁の数の整列 (赤色) がクラスタを作っている。このクラスタの近くに選択ソート (青色) のクラスタがある。ここでも、バブルソートやヒープソートが明確にクラスタを作っている。

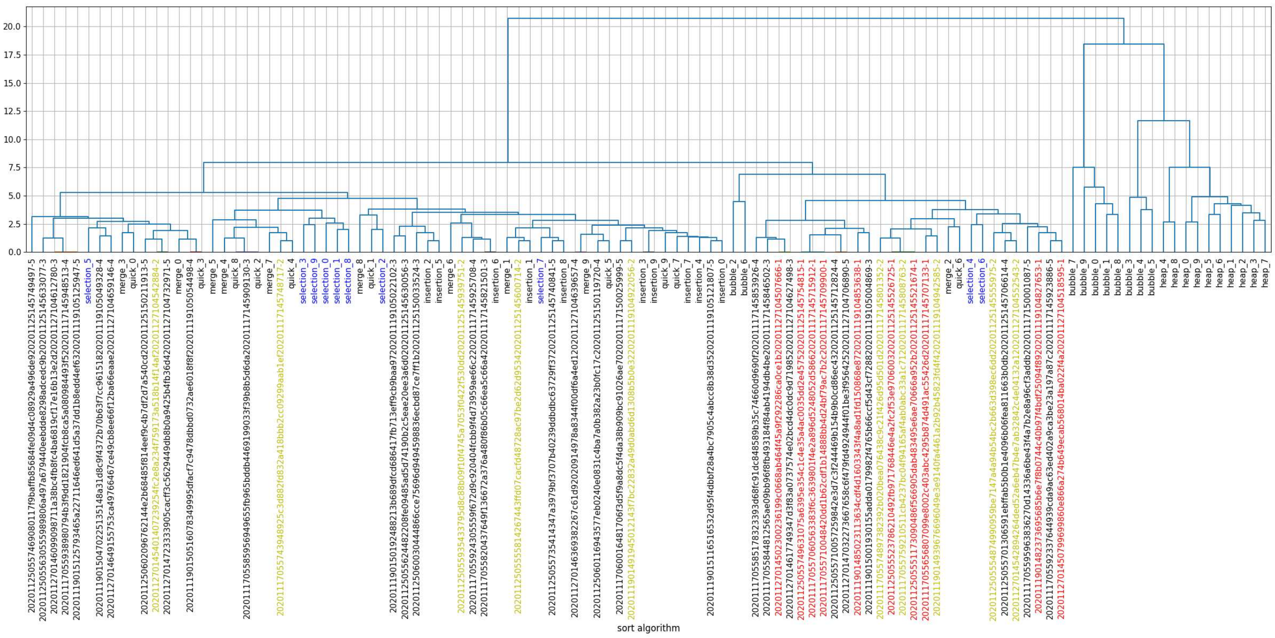

図 16　対角線を重視して共起行列から抜き出した 25 個の要素のクラスタリング結果。人とコンピュータによる整列を合わせたもののデンドログラム。人による 1 桁の数の整列 (赤色) がクラスタを作っている。このクラスタに選択ソート (青色) が一部入っているが多くはない。