

# 卒業論文

## 政策課題解決に向けた効果的な 施策立案のための意思決定支援システム

Development of a Browser-based Automatic Menu Creation  
System Dealing with Restricted Meals and Large Groups of People

放送大学 教養学部情報コース

2010022122 奥原由利恵

指導教員

提出年月: 令和7年(2025年)月



# 目 次

図一覧	ii
表一覧	iii
記号一覧	iv
<b>第1章 はじめに</b>	<b>1</b>
§ 1.1 本研究の背景	1
§ 1.2 本研究の目的	2
§ 1.3 本論文の概要	3
<b>第2章 サイバー空間からのデータ収集と処理</b>	<b>4</b>
§ 2.1 多様な要因を考慮したデータセットの作成	4
§ 2.2 データクリーニングによる前処理	7
§ 2.3 多様な要因と政策課題解決の関係	13
<b>第3章 複数の評価基準からの総合評価</b>	<b>19</b>
§ 3.1 因果探索に基づく入力と出力の分類	19
§ 3.2 データ包絡分析の種類と効率の評価	25
§ 3.3 データ包絡分析による改善策の導出	28
<b>第4章 提案手法</b>	<b>32</b>
§ 4.1 因果探索による入出力の選定とデータの整列	32
§ 4.2 選定されたデータに基づく DEA 分析	35
§ 4.3 提案手法の流れ	37
<b>第5章 数値実験並びに考察</b>	<b>40</b>
§ 5.1 数値実験の概要	40
§ 5.2 実験結果と考察	42
<b>第6章 おわりに</b>	<b>47</b>
<b>謝辞</b>	<b>48</b>
<b>参考文献</b>	<b>49</b>

## 圖一覽

4.1 pymoo における様々な視覚化手法 . . . . . 33

# 表一覧

3.1	6事業体の2入力・1出力の例	29
3.2	DEAの分析結果	30
4.1	pymooの実装アルゴリズムの例	33

# 記号一覧

以下に本論文において用いられる用語と記号の対応表を示す.

用語	記号
$j$ 人目の使用者の名前	$\epsilon_j$
$j$ 人目の身長	$\alpha_j$
$j$ 人目の体重	$\beta_j$
$j$ 人目の基礎代謝量 (下限)	$B_j^L$
$j$ 人目基礎代謝量 (上限)	$B_j^H$
$j$ 人目のアレルギー情報	$x_j$
$j$ 人の有する生活習慣病	$z_j$
対象の日数	$D$
レシピの数	$R$
食材の数	$Q$
栄養素の数	$N$
データベース上の食材数	$S$
データベース上の食材番号	$d : 1, 2, 3, \dots, S$
日の番号	$k : 1, 2, 3, \dots, 3D$
栄養素の番号	$l : 1, 2, 3, \dots, N$
材料の番号	$m : 1, 2, 3, \dots, Q$
レシピの番号	$i : 1, 2, 3, \dots, R$
$i$ 番目のレシピの名前	$y_i$
$i$ 番目のレシピの献立フラグ	$r_{ki}$
$i$ 番目のレシピの主菜フラグ	$\sigma_i$
$i$ 番目のレシピの調理時間	$T_i$
$i$ 番目のレシピの摂取カロリー	$C_i$
$i$ 番目のレシピの調理コスト	$G_i$
$i$ 番目のレシピの $m$ 番目の材料の名前	$q_{im}$
$i$ 番目のレシピの $m$ 番目の材料量	$e_{im}$
$i$ 番目のレシピの $l$ 番目の栄養素の名前	$n_{il}$
$i$ 番目のレシピの $l$ 番目の栄養素の量	$f_{il}$
$d$ 番目の食材名	$Z_d$
$d$ 番目の食材の販売単位	$W_d$
$d$ 番目の食材の値段	$M_d$

## はじめに

### § 1.1 本研究の背景

日本の自治体は、地域社会の基盤として住民の生活を支える重要な役割を担っていますが、近年は少子高齢化や人口減少、財政難、自然災害への対応、さらにはデジタル化の遅れなど、さまざまな課題に直面しています。

特に、地方自治体では限られた人材や予算の中での効率的な業務遂行と住民サービスの向上が求められています。その一方で、地域経済の活性化や防災、環境対策など、個別のニーズにも応える必要があります。こうした複合的な課題に対処するためには、新しい技術の導入や業務プロセスの見直しが急務となっています。

自治体が抱える主要な課題 7つ日本の自治体は地域社会の運営を担う重要な存在ですが、現在その多くが複数の深刻な課題に直面しています。これらの課題は、地域住民の生活に直結し、自治体の持続可能性を左右する要因でもあることから、早急に対応することが求められています。

1. 少子高齢化と人口減少少子高齢化と人口減少は、日本の多くの自治体にとって最も深刻な問題です。特に地方では人口流出が続き、高齢化率が急速に進んでいます。この状況は、自治体の税収減少や公共サービスの維持が難しくなるといった財政的な影響を及ぼし、さらに地域経済の停滞や社会インフラの維持が困難になる悪循環も引き起こしています。

また、人口減少によって学校や病院といった公共施設の統廃合が進められる一方で、高齢化に伴い福祉や医療サービスの需要が増加しています。これにより、住民サービスの質を保ちながら財政を健全に維持することが難しいという課題が浮き彫りになっています。

2. 財政難とリソース不足自治体の財政難も深刻な課題です。特に地方自治体では、インフラの維持や住民サービスの提供にかかるコストが年々増加しています。こうした状況下では、いかに限られた予算内で効率的に業務を遂行し、かつ地域の活性化を図るかが問われます。

この課題に対処するためには、行政の業務フローを見直し、無駄を省くとともに、デジタル化や自動化による効率化を進めることが求められます。しかし、多くの自治体では、これらの施策に対応するための人手不足が顕著で、業務の効率化が思うように進んでいないのが現状です。

人口減少社会の本格的な到来、地方創生を契機とした地域の特性に応じたまちづくり、激甚化する自然災害、DXの推進、感染症対策など、自治体を取り巻く環境は大きく変化しています。そのため、自治体職員には、地域が抱える様々な課題を的確に把握し、先送りすることなく、その課題解決のために前向きに取り組むことが、ますます求められています。

## § 1.2 本研究の目的

地方と都市部のバランスを取るために、多くの自治体が地方創生に向けた取り組みを進めています。

地方では、地域資源を活用した観光業や農業の振興が主要な戦略です。例えば、特産品を活かした「地域おこし協力隊」の派遣や、地方の歴史と文化を生かした観光プログラムの整備が進められています。これにより、地方経済の活性化が図られ、都市からの人口流入も期待されています。

また、近年はテレワークやリモートワークの普及により、地方でも都市部と同様の働き方が可能になってきています。これに応じて、地方に移住しつつ都市部の企業にリモートで勤務する「デュアルライフ」や「ワーケーション」を支援する自治体も増えています。インフラの整備や住居支援など、地方への移住促進策が地域格差是正の鍵となっています。

栄養のバランスが取れた献立を作成するには、膨大なメニューの組み合わせや、複雑な計算や必要な栄養価を考慮する必要があり、献立を考える時間が無かったり、自分で献立を考えることが面倒だと考える人は少なくない [4]。また、短い調理時間でお手軽にかつ食材コストを抑えられ、さらに自身の好みに合った献立を作成することは、時間がない人や空き時間を作りたい人、できるだけ献立作成の費用を節約をして料理を作成したい人にとっては理想的であるといえる。

また、献立作成業務を行っている学校給食や病院食の現場での栄養士は、煩雑な栄養計算や食材にかかる費用の計算などの様々な条件を考慮した上で何度も見直しながら献立を作成している。そのため、献立作成を自動化することによって、献立作成業務の負荷を軽減することが求められている。

他にも病院の現場では、毎日の食事は患者にとって生活リズムの中心であり、日々の楽しみの1つでもある。そのため、食の感動を大切にし、病院では医食同源の精神を基本に患者の好みに合ったメニューを提供することが求められている。食に関する専門性を高めるために、日々食に対して研究や開発、研修を行っていることから、献立作成する業務を行っている人にとって負荷の高い業務を行っていることが分かる。

そこで、本研究では、献立作成を組み合わせ多目的最適化問題として考えることにより、栄養のバランスがとれていて、調理時間、食材コストが少なくなるように、なおかつ、使用者の身体情報にあった栄養量や摂取カロリーについての要望を満たされるように、さらには使用者の好みや病態に最も適した献立を、自動的に作成するシステムを提案する。

また、最適化に用いる料理のデータは、Web上に存在する複数のレシピサイトからPythonのプログラムによるスクレイピングによって蓄積する。具体的な料理データの中身として、必要食材や摂取する栄養量、カロリーなどが挙げられる。また、料理のデータに対するコスト計算を行うため、食品価格の推移を調査しているWebサイトから食材と販売単位あたりの価格をスクレイピングによってデータベースに蓄積する。上記の方法によって蓄積したデータを入力とした、組み合わせ多目的最適化問題を解く手法として、遺伝的アルゴリズムを応用した、非優越ソート遺伝的アルゴリズム (Non-dominated Sorting Genetic Algorithm: NSGA-II) を採用する。

本研究では、NSGA-II や遺伝的アルゴリズムをはじめとした、様々な最適化手法について幅広く対応している、pymoo という Python ライブラリを利用して、最適化プログラムを記述する。さらに、最適化され、出力された料理の中から、ユーザ自身の希望する料理

が選択できるように、分かりやすく感覚的にシステムを利用できるような対話型処理を用いる。具体的には、料理の合計調理時間と合計コストの候補を表示し、ユーザに選択してもらうものである。

また、本プログラムは大量のレシピデータやプログラムを使っているため、使用者が利用するたびにデータをダウンロードしてはプログラムの実行する環境を整えるのに時間がかかってしまう。そのため本研究ではサーバー上にプログラムを置き、利用者にはブラウザでサーバーにアクセスするだけで利用できるようなシステムを実現する。

最後に、本研究によって提案された制限食や大人数料理を考慮した、自動献立作成システムをブラウザ上で動作させ、数値実験を行い、実験結果に基づき考察をする。

### § 1.3 本論文の概要

本論文は次のように構成される。

**第1章** 本研究の背景と目的について説明する。背景では栄養バランスの摂れた献立を作成することの難しさと、自動で献立を作成することの重要性について示す。目的は制限食を考慮した多目的遺伝的アルゴリズムによる最適な自動献立作成について提案することを述べる。

**第2章** 多目的最適化による自動献立作成システムの概要と、Web上のデータを活用した例について説明する。

**第3章** 多目的最適化と、GAを応用した多目的GAの仕組みを説明する。また、本研究で用いる制限食及びブラウザベースのシステムについて説明する。

**第4章** 提案手法の中で利用者が入力する部分と、NSGA-IIによる多目的最適化によって最適な献立を対話型で出力する部分について説明する。

その後、提案手法について説明する。

**第5章** 提案手法に基づいて自動献立作成システムを構築して、実際に献立の作成を行った結果を示す。そして、本研究の提案手法によって得られた結果が有意であることを示す。

**第6章** 本研究で述べている提案手法をまとめて説明する。また、今後の課題について述べる。



# サイバー空間からのデータ収集と処理

## § 2.1 多様な要因を考慮したデータセットの作成

本研究では、対象データの収集を異なるウェブサイトから行った。これらのサイトは、情報の網羅性、更新頻度、データの信頼性を考慮した選定した。以下に各サイトの概要について説明する。

### 地域経済分析システム（RESAS）

RESAS とは、地域創生の実現に向けて、内閣府地方創生推進室ビッグデータチーム、内閣官房まち・ひと・しごと創生本部事務局と経済産業省地域経済産業調査室が提供しているWEB アプリケーションである。日本全国の各自治体区分における人口、産業、観光、まちづくり、医療・福祉、財政など幅広い分野のビッグデータを官民より集約し、表、グラフ、マップなどのフォーマットを用いてデータを可視化するシステムとして提供されている。

また、データを可視化するだけではなく、集積されたデータをもとに各テーマに沿った分析を行うことや、API を利用することによってシステムにデータを直接取り込むことも可能である。これらの機能は地方自治体をはじめ、その他地域の活性化に関心を持つ人々に対して一般に公開されており、地方自治体における地域課題の抽出、地域版総合戦略の立案といった活用法に加えて、地方創生に関心のある民間の団体・個人による活用も期待される。

### e-Stat

日本の政府統計に関する情報のワンストップサービスを実現することを目指した政府統計ポータルサイトです。これまで各府省等が独自に運用するWeb サイトに散在していた統計関係情報を本サイトに集約、社会の情報基盤たる統計結果を誰でも利用しやすいかたちで提供することを目指し、各府省等が登録した統計表ファイル、統計データ、公表予定、新着情報、調査票項目情報、統計分類等の各種統計関係情報を提供していきます。

e-Stat で提供されている統計データは、集計されている区分ごとに、全国ごと、都道府県ごと、市区町村ごと、”…丁目”といった小地域ごと、グリッドセルごとの 5 種類が存在する。本システムでは、予測する空間的な単位をグリッドセルとしているため、グリッドセルごとの統計データを使用するが、より考慮できる要因を増やすため、小地域ごとの統計データも使用できるようにした。小地域ごとのデータはそのまま用いることはできないため、グリッドセル単位に変換する必要がある。そのため、小地域ごとのデータについて、

小地域全体に均等に分布していると仮定し, 対象のグリッドセルに重なっている割合だけを足し合わせる. すなわち, 対象のグリッドセル  $C$  に重なる小地域  $A_1, A_2, \dots, A_n$  について, それぞれの全体の面積を  $S_1, S_2, \dots, S_n$ , 対象のグリッドセルと重なる面積を  $s_1, s_2, \dots, s_n$ , データ値を  $x_1, x_2, \dots, x_n$  とすると, 対象のグリッドセル  $C$  のデータ値  $X$  を次のように算出する.

$$X = \sum_{k=1}^n x_k \frac{s_k}{S_k} \quad (2.1)$$

## 不動産情報ライブラリ

不動産情報ライブラリとは, 不動産の取引価格, 地価公示等の価格情報や防災情報, 都市計画情報, 周辺施設情報等, 不動産に関する情報をご覧になることができる国土交通省の WEB サイトです.

## 犯罪データ

今回の数値実験で使用する犯罪発生データは, 富山県警が公開している「犯罪発生マップ」[?] から取得したものを用いる. 犯罪発生データには, データ項目として「発生時刻」, 「罪種」, 「発生場所 (緯度, 経度)」が含まれており, 「自転車盗難」, 「ひったくり」, 「車上ねらい」, 「部品ねらい」, 「わいせつ」, 「声かけ, つきまとい」, 「タイヤ盗難」の 7 種類の路上犯罪が収録されている. また, 今回使用するレコードは, 2010 年 9 月 1 日から 2020 年 9 月 31 日の約 10 年間とし, 発生時刻や発生場所が不正 (欠損や範囲外など) なものを除外した 25,814 件である. なお, 「犯罪発生マップ」は, このような分析を目的としておらず, 掲載されている情報に誤差や欠損が存在する可能性があることに注意されたい.

## 施設データの取得

施設データを取得できるサービスとして, Google Maps API が存在するが, 無料で取得できる数に制限があるほか, たとえば遊園地や水族館など, レジャー施設としてジャンル分けるものに対して, 「レジャー施設」と検索しても, それらを網羅できるとは限らない点で, 採用しなかった. そこで, ナビゲーションサービスのひとつである「NAVITIME」から施設データをスクレイピングして取得することとした.

スクレイピングとは, データを収集した上で利用しやすいうように加工をすることである. 特に, Web 上から必要なデータを取得することを, Web スクレイピングと呼ばれている. スクレイピングと似ている意味の言葉にクローリングがあり, スクレイピングとは違い, これは, 単に Web 上のデータを収集することを意味する. データを活用するために, 使いやすく抽出や加工をしたりするのがスクレイピングの特徴である.

BeautifulSoup4 とは, Web サイト上の HTML から, 必要なデータを抽出することができるライブラリである. BeautifulSoup4 でスクレイピングする際, 最初に対象の Web ページから HTML を取得する必要がある. HTML を取得する方法として, 同じく Python のライブラリである, Requests の get 関数などがある. 上記の方法によって取得された HTML テ

キストを、BeautifulSoup4 の BeautifulSoup 関数に渡すことで BeautifulSoup オブジェクトを作成ができ、そのオブジェクトから要素検索をすることで必要な情報を抽出する。

NAVITIME は、施設のジャンルごと、さらには都道府県ごとに一覧となって表示される。

施設の最短距離と立地数の算出を行った。さまざまなジャンルの施設について、NAVITIME からスクレイピングを行い、施設名と、その緯度と経度を取得する。本システムでは、それぞれのジャンルごとに、対象のグリッドセルに含まれる数と、最も近くにある施設までの距離を説明変数とする。

なお、地球は楕円体であるため、単純なユークリッド距離では誤差が生じてしまう。そこで、本システムではヒュベニの公式 [?] を用いて、距離を算出している。対象のグリッドセルの中心を  $P_o(x_o, y_o)$ 、注目する施設を  $P_n(x_n, y_n)$  とすると、それら 2 点間の距離  $D$  は以下で求まる。

$$D = \sqrt{(D_y M)^2 + (D_x N \cos P)^2} \quad (2.2)$$

$$M = \frac{R_x(1 - E^2)}{W^3} \quad (2.3)$$

$$N = \frac{R_x}{W} \quad (2.4)$$

$$W = \sqrt{1 - E^2 \sin P^2} \quad (2.5)$$

$$E = \sqrt{\frac{R_x^2 - R_y^2}{R_x^2}} \quad (2.6)$$

## 地図画像にもとづく説明変数の抽出

Mapbox Static Tile API を利用して、それぞれのメッシュに対応する地図画像を取得する。本研究では、建物、道路、水、空き地の 4 つを色分けした地図画像を取得し、それぞれの画像の大きさに対する面積の比率を説明変数としている。

対象の要素の面積比率  $p_a$  は、地図画像の大きさを  $n \times m$ 、その要素と同一の RGB 値をもつピクセル数を  $x$  とすると、以下のように算出する。

$$p_a = \frac{x}{nm} \quad (2.7)$$

なお、Mapbox Static Tile API によって取得する地図画像は、本来は画像処理を目的としていない。そのため、Mapbox Studio 上で指定した RGB 値と誤差があるピクセルがある。そのため、対象のピクセルの RGB 値と、それぞれの要素の RGB 値とのユークリッド距離を算出し、最も小さい要素を指定する。

また、地図画像から道路ネットワークを抽出し、道路に関連する属性を説明変数として抽出する。まず、道路とそれ以外の 2 値画像に変換し、ノイズを削除するためにオープニング処理を行う。その画像に対して、ネットワークを抽出するアルゴリズム [?] を使用し、ネットワークの属性であるノード数  $N$ 、エッジ数  $E$  を取得する。また、それらから密度  $d$ 、平均次数  $k$  を以下のとおり算出する。

$$d = \frac{2E}{N(N - 1)} \quad (2.8)$$

$$k = \frac{2E}{N} \quad (2.9)$$

なお、密度  $d$  と平均次数  $k$  は、道路のネットワークとしてみたとき、それぞれ次のような特徴をもつ [?]. 密度  $d$  が大きい道路ネットワークは、道路が網目状に相互に接続された状態であり、幅員の狭い生活道路であると考えられる。また、平均次数  $k$  が小さい道路ネットワークは、交差点の少ない直線的な道路が多いと考えられる。

### 動的データと静的データの組み合わせ

上で述べたものはすべて、1日ごとに変化しない静的データであった。それらと、1日ごとに変化する動的データから、空間（グリッドセル）軸 × 特徴量 × 時間軸の、3次元のテーブルを作成する。

以上により、機械学習に用いるデータセットの作成が完了する。

### データセット

予測に用いる説明変数は、表??に示す計 65 個とした。

3,126 個のグリッドセルについて、静的データをまとめたテーブルを作成し、それに動的データを追加した 3 次元のテーブルを作成する。予測モデルの精度を検証するため、データセットのうち、2020 年 8 月 31 日までの 10 年間を学習用、それ以降の 1 か月間を検証用とする（図??参照）。

よって、この時点におけるデータセットのレコード数は、 $N = 11419278$  であり、式 2.14 における不均衡度  $r$  は、 $r \approx 0.0023$  である。そこで、学習用データについて、ランダムサンダーサンプリングを行い、 $N = 51065$ 、 $r \approx 0.477$  となった。

### 予測モデルの精度検証

なお、本研究では、データセットとして使用した、または可視化された説明変数を「要因」と仮定し分析を行ったが、あくまでも予測値に対する説明変数の「関係性」を示しており、実際に因果関係を検証するためには、因果推論などの手法を用いる必要があることに注意する必要がある。さらに、解釈された結果は、予測モデルの精度に左右されるため、その精度が大きいことが前提となっていることにも留意すべきである。

## § 2.2 データクリーニングによる前処理

機械学習における分類問題で、学習に用いるデータセットが、そのクラスの比率に極端な偏りが生じているものを、一般に「不均衡データ」という。

### 不均衡なデータが予測精度に及ぼす影響

実世界で観測されるデータは、そのクラス比率が不均衡になっているものが多い。分かりやすい例として、医療データがある。身体検査のデータから特定の疾患を発病しているかどうかを出力するモデルを作成しようとするとき、健常者のデータは比較的容易に収集

できるが、それと比較して、その患者のデータは、健常者よりも母数が小さく、収集できるデータ数が少なくなってしまう。

しかしながら、一般的な機械学習アルゴリズムは、それぞれのクラスのデータ数が同一であることを仮定していることが多く、不均衡な学習データを用いて機械学習を行うと、少数派に対する分類精度が著しく低下する [?]. この理由を、線形判別分析 (Linear Discriminant Analysis: LDA) を例に説明する。LDA は、2つのクラスを、最も識別できる直線 (識別境界) で分離する手法である。多次元であるときを考慮し、サンプル  $\mathbf{x}_n \in \mathbb{R}^M$  を、識別境界に直行している軸  $\mathbf{w}$  で線形変換した

$$y_n = \mathbf{w}^\top \mathbf{x}_n \quad (2.10)$$

を用いて、クラスを分類する。このとき、目的関数  $J(\mathbf{w})$  について、

$$\text{maximize} \quad J(\mathbf{w}) = \frac{\mathbf{w}^\top \mathbf{S}_B \mathbf{w}}{\mathbf{w}^\top \mathbf{S}_W \mathbf{w}} \quad (2.11)$$

となる射影軸  $\mathbf{w}$  を求める。なお、クラス間変動行列  $\mathbf{S}_B$  は、

$$\mathbf{S}_B = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^\top \quad (2.12)$$

であり、クラス内変動行列  $\mathbf{S}_W$  は、

$$\mathbf{S}_W = \sum_{n \in C_1} (\mathbf{x}_n - \mathbf{m}_1)(\mathbf{x}_n - \mathbf{m}_1)^\top + \sum_{n \in C_2} (\mathbf{x}_n - \mathbf{m}_2)(\mathbf{x}_n - \mathbf{m}_2)^\top \quad (2.13)$$

である。いま、クラス  $C_1, C_2$  の平均  $\mathbf{m}_1, \mathbf{m}_2$  は変化せずに、 $C_2$  のサンプル数が十分小さいとき、式 2.13 の第2項は、第1項と比べて非常に小さくなる。したがって、射影軸  $\mathbf{w}$  は、ほとんどクラス  $C_1$  の変動のみを考慮することとなり、クラス分類の精度は小さくなる (図 ?? 参照)。このような現象は、LDA に限ったことではない。

多数クラス  $C^{maj}$  のサンプル数を  $N^{maj}$ 、少数クラス  $C^{min}$  のサンプル数を  $N^{min}$  とすると、データの不均衡度  $r$  は

$$r = \frac{N^{min}}{N^{maj} + N^{min}} \quad (2.14)$$

と定義できる。 $r < 0.2$  であると、そのデータは十分不均衡であり、機械学習に用いるときは、何らかの工夫が必要である [?]. 以降、不均衡データに対するアプローチとして、サンプリングとアンサンブル学習を説明する。

## サンプリング

サンプリングとは、多数クラス  $C^{maj}$  のデータを間引いたり、少数クラス  $C^{min}$  のデータを生成することで、式 2.14 における不均衡度  $r = 0.5$  に近づける手法である。

### 1. アンダーサンプリング

アンダーサンプリングとは、多数クラス  $C^{maj}$  のデータを間引く手法であり、選択型と生成型の2つのアプローチがある。

選択型は、既存のデータから間引くものを選択するアプローチであり、ランダムに選択するランダムアンダーサンプリング (Random Under Sampling: RUS)、クラスをいくつかに分割して、それぞれからランダムに選択するクラスタ基準アンダーサンプリング、トメクリンク (距離が近い  $C^{maj}$  と  $C^{min}$  のデータのペア) の多数クラス  $C^{maj}$  側のデータを間引くアンダーサンプリングがある。

生成型は、多数クラス  $C^{maj}$  の既存のデータをそのまま使用せずに、新たなデータを少数クラス  $C^{min}$  と同数生成するアンダーサンプリングである。新たなデータの生成には、 $k$ -平均法が用いられることが多い。

## 2. オーバーサンプリング

オーバーサンプリングとは、少数クラス  $C^{min}$  のデータを新たに生成する手法であり、そのアルゴリズムは、さまざまなもののが提案されている。

Synthetic Minority Oversampling Technique (SMOTE) は、注目している  $i$  番目の少数クラス  $C^{min}$  のデータを  $\mathbf{x}_i^{min}$  として、 $\mathbf{x}_i^{min}$  と  $k$  番目までに距離が近い  $k$  個の  $\mathbf{x} \in C^{min}$  を取り出す。これを、 $k$ -近傍サンプル  $\mathbf{x}_k^{min}$  と呼ぶ。 $\mathbf{x}_k^{min}$  のなかからランダムにひとつ選択し、これを  $z$  とする。 $\mathbf{x}_i^{min}$  と  $z$  の内挿となる点に、 $\mathbf{x}_i^{min}$  の新たなデータ  $\mathbf{y}_i$

$$\mathbf{y}_i = \mathbf{x}_i^{min} + r(\mathbf{x}_i^{min} - z) \quad (2.15)$$

を生成する。これを、すべての  $\mathbf{x}_i^{min} \in C^{min}$  に対して行い、生成された  $\mathbf{y}_i$  を学習データに追加する。ほかにも、SMOTE から派生した Adaptive Synthetic (ADASYN) や、ボーダーライン SMOTE などの手法が存在する。

## アンサンブル学習

アンサンブル学習とは、複数のモデルを学習し、それらをもとに最終的な出力を決定する手法である。最終的な出力を決定するために学習させる複数のモデルを弱学習器と呼び、それをどのように学習させるかによって、バギングとブースティングの 2 つに分けることができる。

バギングとは、ブートストラップ法 (データからランダムに一部のサンプルを取り出すことを繰り返し、ひとつのデータセットから複数のデータセットを生成する手法) を行い、それぞれのデータセットで弱学習器を作成し、それぞれの出力をもとに、最終的な出力を決定する手法である。

ブースティングとは、 $n$  個の弱学習器をそれぞれ直列に学習させ、それぞれの出力をもとに最終的な出力を決定するが、 $k$  番目 (ただし、 $k = 1, 2, 3, \dots, n-1$ ) に学習させた弱分類器の出力を、 $k+1$  番目に学習させる弱分類器に応用する手法である。

機械学習、たとえば、近年急速に注目されている深層ニューラルネットワークといったアルゴリズムは、複雑かつ非線形な性質であってもモデリングすることができる。すなわち、より予測精度の大きいモデルを作成することができる。しかしながら、一般にモデルの精度が大きくなるほど、その解釈性は小さくなる性質がある。

## 予測モデルを解釈する重要性

近年、深層ニューラルネットワークなどの表現力の高いモデルを作成できるアルゴリズムの登場により、多くの分野で機械学習が活用されるようになってきた。医療分野では、網膜

の画像から、糖尿病網膜症かどうかを診断するシステムが、米国で認可されている [?]. そのような責任が大きい判断の場合は、予測精度が大きいことはもちろん、なぜその予測値を出力したのか、その根拠も人間が知る必要がある。そのため、総務省が示している「AI利活用原則」や、EUが施行している「一般データ保護規則（General Data Protection Regulation: GDPR）」においても、機械学習モデルの説明責任について言及しており、予測モデルを解釈することは、国際的に重要視されているといえるだろう。

## 予測精度と解釈性のトレードオフ

以下のような線形回帰モデルを考える。

$$f(X_1, X_2) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 \quad (2.16)$$

このとき、 $X_1$ が1だけ大きくなると、 $f(X_1, X_2)$ は $\beta_1$ 倍だけ大きくなることが明示的に分かる。このように、線形回帰モデルは、目的変数と説明変数とのあいだに単純な関係を仮定しており、モデルに対する透明性が高いと言える。これを、一般に「解釈性が高い」と言う。

一方で、比較的近年に発表されたアルゴリズム、例えば深層ニューラルネットワークやランダムフォレストなどは、目的変数と説明変数とのあいだに線形性などの仮定を置いていない。よって、より複雑な関係をモデリングできるようになり、一般に線形回帰モデルよりも予測精度は大きくなりやすい。しかしながら、線形回帰モデルと違い、その複雑さから、なぜその予測値を出力するのかを理解することができず、その中身はブラックボックスとなりやすい。これを、一般に「解釈性が低い」と言う。

## 予測モデルを解釈する主な手法

機械学習によって作成されたモデルに対して、何らかの解釈を与える手法はいくつか存在するが、特に有用なものとして、以下の4つが挙げられるだろう。

- Permutation Feature Importance (PFI)
- Partial Dependence (PD)
- Individual Conditional Expectation (ICE)
- Shapley Additive Explanations (SHAP)

それぞれは何を解釈できるのかが異なり、用途によって使い分ける必要がある（図??参照）。例えば、モデル全体の傾向など、マクロな視点から解釈する場合はPFIを、出力されたひとつの予測値に対する根拠など、ミクロな視点を知りたい場合はICEを用いるべきだろう。本研究では、ミクロな視点から解釈できるものの、マクロな視点からの解釈も可能なSHAP [?]を用いることとする。

## SHAP

$\mathbf{X} = (X_1, \dots, X_J)$  を説明変数とする学習済みのモデルを  $\hat{f}(\mathbf{X})$  とする。インスタンス  $i$  の説明変数が  $\mathbf{x}_i = (x_{i,1}, \dots, x_{i,J})$  とすると、インスタンス  $i$  の予測値は  $\hat{f}(\mathbf{x}_i)$  である。ここで、予測の期待値を  $\mathbb{E}[\hat{f}(\mathbf{X})]$ 、インスタンス  $i$  の説明変数  $x_{i,j}$  の貢献度  $\phi_{i,j}$  としたとき、

$$\hat{f}(\mathbf{X}_i) - \mathbb{E}[\hat{f}(\mathbf{X})] = \sum_{j=1}^J \phi_{i,j} \quad (2.17)$$

のように、期待値からの差分を貢献度の総和で表現できるように、貢献度を分解することができる、SHAPの基本的な考え方である。線形モデルであれば、比較的容易に分解することができるが、非線形モデルではこのままでは難しい。そのため、SHAPでは、協力ゲーム理論のShapley値の考え方を用いて、貢献度を分解する。

ここで、協力ゲーム理論のひとつであるアルバイトゲームを説明する。アルバイトの参加者として、A, B, Cの3つのプレイヤーを仮定し、アルバイトの参加者とそのときに得られる報酬には、表??のような関係があるとする。

A・B・Cの3プレイヤーが参加したときの報酬は24である。より貢献度が大きいプレイヤーに、より多くの報酬を配分するとすれば、その貢献度はどのように算出すべきだろうか。ここで、限界貢献度という概念を導入する。限界貢献度とは、あるプレイヤーが参加したときの報酬と、参加する直前の報酬との差を表す。例えば、B・Cがすでに参加しているときにAが参加した場合の限界貢献度は、 $24 - 10 = 14$ である。しかし、各プレイヤーがどのような順序で参加するかにより、限界貢献度は異なる。例えば、Aの限界貢献度について、A, B, Cという順番で参加したときは6であるが、B, C, Aという順序で参加したときは14である。

この影響を解消するため、考えられるすべての順序で限界貢献度を算出し、その平均を求めることがある。例えば、Aの限界貢献度の平均値は、 $(6 + 10 + 16 + 14 + 13 + 14)/6 = 11.5$ である。この限界貢献度の平均値をShapley値といい、これをもとに報酬を分配する。一般に、 $J$ つのプレイヤーが存在するとき、プレイヤー  $j$  のShapley値  $\phi_j$  は以下のように算出される。

$$\phi_j = \frac{1}{|\mathcal{J}|!} \sum_{\mathcal{S} \subseteq \mathcal{J} \setminus \{j\}} (|\mathcal{S}|! (|\mathcal{J}| - |\mathcal{S}| - 1)!) (v(\mathcal{S} \cup \{j\}) - v(\mathcal{S})) \quad (2.18)$$

SHAPは、このShapley値の考え方を機械学習のモデルに適用している。例えば、説明変数が  $X_1, X_2$  であるモデルにおいて、インスタンス  $i$  の予測値  $v(\{1, 2\})$  の、説明変数を  $x_{i,1}, x_{i,2}$  とすると、

$$v(\{1, 2\}) = \hat{f}(x_{i,1}, x_{i,2}) \quad (2.19)$$

である。また、 $x_{i,1}$  と  $x_{i,2}$  のいずれも未知の場合は、予測値の期待値とし、

$$v(\emptyset) = \mathbb{E}[\hat{f}(X_1, X_2)] \quad (2.20)$$

である。では、 $x_{i,1}$  は既知であり、 $x_{i,2}$  は不明であるときの予測値  $v(\{1\})$  は、後者について周辺化を行い、

$$v(\{1\}) = \mathbb{E} \left[ \hat{f}(x_{i,1}, X_2) \right] = \int \hat{f}(x_{i,1}, x_2) p(x_2) dx_2 \quad (2.21)$$

である。よって、 $x_{i,1}, x_{i,2}$  という順序で説明変数が判明したときの、それぞれ時点における限界貢献値  $\Delta_{i,1}, \Delta_{i,2}$  は、

$$\Delta_{i,1} = \mathbb{E} \left[ \hat{f}(x_{i,1}, X_2) \right] - \mathbb{E} \left[ \hat{f}(X_1, X_2) \right] \quad (2.22)$$

$$\Delta_{i,2} = \mathbb{E} \left[ \hat{f}(x_{i,1}, x_{i,2}) \right] - \mathbb{E} \left[ \hat{f}(x_{i,1}, X_2) \right] \quad (2.23)$$

である。Shapley 値と同様に、考え得るすべての順番で算出し、それらを平均する。すなわち、説明変数  $x_{i,1}, x_{i,2}$  について、その平均値  $\phi_{i,1}, \phi_{i,2}$  は、

$$\phi_{i,1} = \frac{1}{2} \left( \left( \mathbb{E} \left[ \hat{f}(x_{i,1}, X_2) \right] - \mathbb{E} \left[ \hat{f}(X_1, X_2) \right] \right) + \left( \hat{f}(x_{i,1}, x_{i,2}) - \mathbb{E} \left[ \hat{f}(X_1, x_{i,2}) \right] \right) \right) \quad (2.24)$$

$$\phi_{i,2} = \frac{1}{2} \left( \left( \hat{f}(x_{i,1}, x_{i,2}) - \mathbb{E} \left[ \hat{f}(x_{i,1}, X_2) \right] \right) + \left( \mathbb{E} \left[ \hat{f}(X_1, x_{i,2}) \right] - \mathbb{E} \left[ \hat{f}(X_2, X_2) \right] \right) \right) \quad (2.25)$$

である。このとき、 $\phi_{i,1}, \phi_{i,2}$  は、協力ゲーム理論においては Shapley 値と呼ぶが、SHAPにおいては SHAP 値と呼ぶ。式 3.17 と式 3.18 より、 $\phi_{i,1}$  と  $\phi_{i,2}$  を足すと、

$$\phi_{i,1} + \phi_{i,2} = \hat{f}(x_{i,1} + x_{i,2}) - \mathbb{E} \left[ \hat{f}(X_1, X_2) \right] \quad (2.26)$$

であり、式 3.10 と同様に、インスタンス  $i$  の予測値と、予測の期待値との差分になっていることが分かる。

## SHAP の有用性

SHAP は、ブラックボックスなモデルであっても、なぜその予測値を出力したのか、説明変数ごとにその貢献度を出力できる。その貢献度がどれほど的確に推定できているかを検証したところ、回帰問題について、既存の解釈手法より正しく貢献度が推定できることが示された [?].

平均 0、標準偏差 30 の正規分布  $N(0, 30^2)$  に従う 5 つの説明変数  $x_1, x_2, x_3, x_4, x_5$  と、平均 0、標準偏差 10 の正規分布  $N(0, 10^2)$  に従うノイズ  $b$  を生成し、式 3.20 および式 3.21 で教師データをそれぞれ 10000 行作成する。式 3.21 で作成した教師データについては、 $x_2$  だけ十の位で四捨五入し、ほかの説明変数とデータの解像度が異なるケースを再現する。

$$y = 15x_1 + 10x_2 + 5x_3 + x_4 + 0.3x_5 + b \quad (2.27)$$

$$y = 10x_1 + 10x_2 + 5x_3 + x_4 + 0.3x_5 + b \quad (2.28)$$

それぞれの教師データを、決定木をベースとした機械学習アルゴリズムである XGBoost でモデルを学習した。そのモデルを SHAP で解釈した結果と、従来手法である Future Importance で解釈した結果を比較する。式 3.20 による教師データの結果を図 3.6、式 3.21 の結果を図 3.7 に示す。なお、Future Importance は、ある説明変数が予測精度をどれだけ向上させたかを、その説明変数の「重要度」として示した値である。図中の赤字で書かれた値は、 $x_4$  の大きさを 1 としたときの各説明変数の比率であり、Future Importance と比較しても、おおむね正確に貢献度を推定できていることが分かる。

#### Boruta による説明変数の選択

本システムでは、ユーザが自由に説明変数を選択することができるが、過度に説明変数の数が大きかったり、目的変数と相関がない説明変数があると、過学習などによって、かえって予測精度が低下してしまう可能性がある。そこで、本システムでは、犯罪発生予測モデルを作成する前に、Boruta [?] と呼ばれるアルゴリズムを用いて、適切な説明変数を選択する。

Boruta のアルゴリズムは、以下のような流れである。

- 1 もともとのテーブルをコピーし、各列をシャッフルする。もとのテーブルの説明変数を Original features、シャッフルした説明変数を Shadow features と呼ぶことにする。このとき、Shadow features は、なんら目的変数に寄与しないはずである。
- 2 Original features と Shadow features を結合し、ランダムフォレストでモデルを作成する。
- 3 そのモデルにおいて、それぞれの説明変数の重要度を算出し、Shadow features における最大値よりも大きい Original features を見つける (hit する)。
- 4 ランダムフォレストの性質により、モデルを作成するごとに重要度は変化するため、1~3 を  $n$  回くり返す。
- 5 各 Original features について、Shadow features の重要度と同じことを帰無仮説、より大きい・より小さいことを対立仮説とし、hit した合計を検定統計量  $T$ 、 $p = 0.5$  としたときの二項分布を用いて検定を行う。

検定の結果、説明変数が、Confirmed, Tentative, Rejected の 3 つに分類される。本システムでは、Confirmed, Tentative の 2 つを、説明変数として用いることとする。

### § 2.3 多様な要因と政策課題解決の関係

先般、全国的に共通して緊急性が高い 3 つの政策課題 (A. 中心市街地の機能回復、B. 産業構造の変化への対応、C. 環境負荷の小さな都市の構築) に対応する都市計画運用指針を策定しましたが、新たに 3 つの政策課題 (D. 職住バランスのとれた大都市の都心構造の構築、E. 高齢者が生き生きと暮らせる環境の実現、F. 防災上危険な市街地の改善) に

対応する都市計画運用指針を策定しました。改めて言うまでもないことですが、今我々は、産業構造の変化、少子・高齢化社会、地球環境問題、さらには人口減少時代の到来と大きな時代の転換期に立っています。このような変化は、おおげさに言えば、これまでの都市計画が拠って立っていた前提が変わろうとしているという意味では、都市計画にも大きな変革を迫るものと考えられます。勿論都市計画は、それぞれの地域の特性に応じて進められるものですが、このような変革期にあっては、一方で国を挙げた取組みも必要ではないかと思っております。昨年制定された都市再生特別措置法はこのような取組みの一環ですが、単に制度の企画・立案だけでなく、その運用についても一定の範囲で問題意識を国全体で共有しながら進めて行くことが大切ではないかと考えます。都市計画は良好なまちづくりを進めるうえで、どちらかといえば守りの視点から都市をとらえがちです。しかしながら近年の経済社会構造の急激な変化に伴う都市構造やまちづくりに関する課題は、従来の受け身的な都市計画の対応では解決できないものも現れてきています。このような状況に鑑みて、様々な都市の課題を解決していくという立場から、これから都市計画は主体的に他の政策手段と連携をとって取組んでいく姿勢（消極的な都市計画から積極的な都市計画、ポジティブ・プランニング）が求められている時期に来ているといえます。このような認識のもとに、今般国全体で共有すべきいくつかの政策テーマに関し、都市計画の運用の指針を策定することにしました。これは、テーマに関する課題の解決に向けて、都市計画が採るべきスタンスや関連する個別都市計画の使われ方などについて、国の考え方だけではなく地方公共団体の皆様のご意見を取り入れつつ策定するものであり、都市計画に関する各主体の意見を集約したものとなっています。選定されるテーマは、そのいずれもがわが国の行く末にも関わるもので都市計画だけで解決できるものではありませんが、都市計画が取り組まなければ解決できないものもあります。先に言いましたように、都市計画は、権限上も実態上も地方公共団体の役割であることは当然のことであり、今回の指針には強制力はありませんし、ここで取り上げたテーマ以外でも、それぞれの都市で取り組むべきテーマはあると思いますが、少なくともこの指針で取り上げるテーマについては国と地方公共団体で気持ちをあわせて取組みたいものだと考えています。指針が、地方公共団体の皆様にとって、その一助となればとの思いで策定するものです。現在、我々は、重層的転換点にある。急速な経済成長を遂げた戦後半世紀から新たな半世紀へ、また、20世紀から21世紀へと時代が転換しつつあるのみならず、内外において次のような大きな潮流の変化が生じつつあり、それに対する対応が求められている。今後21世紀に向けた我が国の発展を考えるに当たっては、我々はこのような転換点にあることをはっきりと認識することから出発しなければならない。

1. グローバリゼーションの進展第1の潮流はグローバリゼーションの進展である。情報通信の高度化、輸送技術の飛躍的発達や自由貿易体制の拡大に伴い、人、モノ、カネ、情報が地球的規模で動くようになっている。また、東アジア等新興国経済の台頭や東欧等の市場経済への参入等を背景として、市場経済の拡大と深化が進行している。この結果、経済活動はボーダーレス化し、メガコンペティションとも呼ばれる大競争時代が到来している。これは、企業の活動の舞台を拡大する一方で、国際競争を一層激化させており、企業は、地球規模の競争を勝ち抜くため、最適な事業環境を求めて国を選び、また、各国は、事業活動の場として優位な環境の整備を巡って競争を展開するという時代となっている。他方、経済活動の地球規模の拡大は、人々の意識のボーダーレス化も促し、経済問題のみ

ならず、人口、環境、食料問題等について、全地球規模で対応すべき課題であるとの認識が広まっており、こうした意味でもグローバリゼーションが進展している。

2. 高次な成熟経済社会への転換第2の潮流は、我が国経済社会が一つの段階を終え、より高次な成熟経済社会へ転換しつつあることである。我が国経済は、主として海外からの技術導入とその応用により、新しい製品・良質な製品を大量に生産し、内外の市場に販売するというパターンにより、急速な発展を遂げてきた。しかしながら、新製品開発をもたらすような独創的な技術を自ら創出していかなければならないことや、単にモノを製造するばかりでなく、製品・サービスの適切な組合せにより消費者ニーズに対応しなければならないことなど、従来型の発展パターンを続けることは困難となっている。特に、かつて日本が競争力を持っていた大量生産型商品分野においては、東アジアを中心とした海外諸国の競争力の向上が著しい。こうした環境変化の下で、日本経済は、その新たな発展をもたらすような、より高次な経済社会への転換を迫られている。さらに、大量生産・大量消費・大量廃棄型の社会経済活動や生活様式のあり方を問い合わせし、生産と消費のパターンを持続可能なものに変革していく必要がある。他方、従来の大量生産指向型経済成長の効率的実現の基礎となっていた企業中心的、集団主義的考え方や行動様式自体、従来型成長パターンを脱却し、新たな経済社会へ転換を図っていく際の阻害要因となりつつあり、また、人々の意識も、こうした考え方や行動様式から脱却し、より個性的で自由な生き方を求めるようになっており、その観点からも、より高次な経済社会への転換が求められている。

3. 少子・高齢社会への移行第3の潮流は少子・高齢社会への移行である。我が国は、21世紀初頭には人口減少型社会に移行することが予想され、労働力人口も21世紀に入ると減少に転じるものと見られている。こうした中で、平均寿命の伸びに伴い、老齢人口が大幅に増加する一方、女性の晩婚化・非婚化などにより、少子化傾向が続いていることも推測される。特に、日本の人口の高齢化は予想を上回るペースで進展しており、先進諸国の中でも例を見ない速さで進むものと予想され、厚生省人口問題研究所「日本の将来推計人口（平成4年（1992年）9月推計）」等によると、65歳以上の人口が総人口に占める割合（老人人口比率）は、平成12年（2000年）には17%、平成32年（2020年）には25.5%と急激な上昇を続けると予想される。この結果、1人の高齢者（65歳以上）に対する現役世代（20歳から64歳）の人数も、2000年で3.6人、2020年には2.1人と予想され、本格的な少子・高齢社会の到来が見込まれる。

4. 情報通信の高度化第4の潮流は情報通信の高度化である。高度の情報通信技術の活用は、時間的・空間的制約を大幅に取り払い、個人と地域、組織、社会との関係や、企業における組織や就業の形態に変化をもたらすことになると予想される。また、情報通信の高度化は、情報やモノの流れを一変させ、産業の生産性の向上をもたらすとともに、新たな関連産業や新規雇用を創出することが見込まれることから、21世紀への明るい展望を開くものとして大きな期待が寄せられている。

第2章 対応すべき構造的諸問題以上のような大きな潮流の変化は、相互に関連しながら我が国に大きな影響を与えつつある。しかし、現在の経済社会構造は、これらの潮流変化にうまく対応できていないため、様々な構造的諸問題が顕在化してきており、これが人々の将来に対する漠然とした不安感をもたらしている。現在我々が直面しているこれらの主要な問題は、次のように整理できる。

1. 新規産業の展開の遅れと産業空洞化グローバリゼーションの進展の中で、企業は最

適な事業環境を求めて積極的な国際展開を進めているが、日本の高コスト構造や過剰規制の存在等により、本来であれば日本国内で国際競争力を持ち得る企業までもが海外へその生産拠点を移転することが懸念されている。また、経済のファンダメンタルズと乖離した円高が進行する場合には一層この傾向を強めることになる。さらに、経済発展の新しい局面に入りつつある今、我が国産業は自ら新しい社会のニーズを的確に把握して潜在的な需要を発見し、次代を担う産業分野を開拓していくことが不可欠であるが、新しい事業展開が遅れており、我が国産業の空洞化が懸念されている。このため、我が国としては、情報通信の高度化等を軸に、新規産業を創出し、新たな経済フロンティアを切り拓いていくよう、経済の活力を高めていくことが重要な課題となっている。

2. 雇用に対する不安グローバリゼーションの進展に伴う厳しい国際競争の中で、企業は一層の経営効率化を目指し、事業再構築を進めており、雇用への影響が生じつつある。また、少子・高齢化の進行に伴う人口構成の変化の中で、年功的待遇制度等の我が国の雇用慣行には、部分的にゆらぎが見られ、この影響を強く受けると見込まれる団塊の世代を始めとした中高年層を中心の雇用への不安が生じている。また、企業の海外展開の進行、新たな雇用を生み出すと期待される新規産業の展開の遅れは、更に雇用に対する不安を高めている。また、日本経済の先行き不透明感の増大は、新規学卒者を中心に若年層の就職をも困難なものにしている。こうした雇用に対する不安に対応するためにも、我が国経済の活力を高めていくことが重要な課題となっている。

3. 少子・高齢社会のくらしへの不安少子・高齢化の進展に伴い、年金・医療・福祉面における給付が増加していくことが見込まれるが、これに伴う負担増に対する社会の対応について、国民の間に漠然とした不安がある。また、後期高齢者（75歳以上の高齢者）比率の増加が見込まれる中で、高齢者介護への家族の私的な負担の高まりへの懸念が生じている。さらに、労働力人口の伸びの鈍化・減少により、我が国経済の活力自体が低下していくのではないかとの懸念も生じている。このため、人口増加型社会、特に現役世代の増加を前提としていた従来の経済社会の見直しを行い、国民の不安や懸念を解消し、豊かで安心できる少子・高齢社会への円滑な移行を可能とする条件整備を進めることが重要な課題となっている。

4. 豊かさの実感の欠如への不満我が国経済社会は新たな段階に入ろうとしているが、社会資本整備や良質な住宅ストック形成の立ち後れ等を背景に、これまでの経済成長の成果が生活の豊かさに必ずしもつながっていないとの意識が高まっている。また、内外価格差への関心の高まりの中で、その背景にある我が国経済の高コスト構造が、生活の豊かさを実感する上での阻害要因となっているとの意識が一層強まっている。さらに、多様で個個的な生き方を求めていた従来の大量生産指向型経済社会を支えてきた企業中心的、集団主義的な経済社会制度や慣行も豊かさの実感の阻害要因として働いている。このため、豊かさが実感できる社会の構築の観点からも経済社会の構造改革を進めていくことが重要な課題となっている。

5. 地球社会における責任と役割の増大グローバリゼーションの進展の下、各国の相互依存関係が深まっていることから、各国は世界経済全体の発展の観点を踏まえた経済運営が求められている。我が国としても、その国際的役割を認識し、内外に開かれた経済社会を築くとともに、国際的な自由競争のルール作りに積極的に参加することにより、世界経済の発展に貢献していくことが重要な課題となっている。また、世界の人口、食料問題等に

ついても地球社会の一員としてその解決に積極的に取り組むことが求められているほか、地球環境問題への対応も迫られている。このように、我が国の地球社会における責任と役割は増大しており、それに十分こたえていくことが重要な課題となっている。

以上の諸問題は、決して目新しいものではなく、従前より我が国が直面しつつあった構造的問題であったが、それに対する構造調整が十分進まない段階で、バブルの発生及びその崩壊、円高の進行がこれらの問題を更に顕在化させた。また、資産価格の下落に伴い、不良資産問題が発生し、金融システムの安定性に対する懸念が生じている。さらに、近年、阪神・淡路大震災や地下鉄サリン事件等が発生し、国民生活の安全に対する不安が生じており、これが、我が国経済社会の先行きに対する不透明感を一層強めるものとなっている。

### 第3章 政策運営の基本方向第1節 構造改革の必要性

我が国の内外に生じている大きな潮流の変化は、相互に複合しながら我が国に変革を迫っている。しかしながら、現在の我が国の経済社会構造は、これらの潮流変化に対応したものになっておらず、むしろ新たな発展の足かせとなっている面もある。このため、潮流と経済社会構造がぶつかり合う中で、構造的な諸問題が顕在化しており、将来に対する不安感をもたらしている。その解決のためには、潮流変化に対応できていない我が国経済社会の構造を抜本的に改革していくことが必要であり、一人一人の意識改革も求められている。構造改革の過程においては、変革に伴うある程度の痛みを伴うことは避けられないが、構造改革なくしては、現在我々が感じている将来に対する不透明感を払拭し、我が国の中長期的発展を切り拓いていくことはできない。このような認識の下、構造改革に直ちに取り組む必要がある。第2節 構造改革の基本方向

こうした認識に立って、構造改革を進めていく際の基本的方向としては、目指すべき経済社会のあり方に即して、次の3つの方向で整理することができる。

1. 自由で活力ある経済社会の創造不確実性の高い環境の下、また、厳しい地球的規模の国際競争にさらされる中で、我が国経済・産業の将来に対する不透明感を払拭し、自ら新たな発展を切り拓いていくためには、改めて自由で活力ある経済社会の創造に大胆に取り組んでいくことが急務である。真の活力をもたらすには、自己責任の下、自由な個人・企業の創造力が十分に発揮できるようになることが重要である。このため、市場メカニズムが十分働くよう、規制緩和や競争阻害的な商慣行の是正を進め、個人、企業の自由な活動を確保する環境整備を図ることが重要である。また、我が国経済の高コスト構造を是正し、新規産業の展開を支援していくとともに、これらを通じ、雇用機会の創出を図り、雇用の安定を図っていくことが必要である。自由な企業や個人のイニシアティブが発揮されることは、活力ある地域経済を創出する上でも有効であり、また、規制緩和等を通じ内外に開かれた経済社会を形成していくことは、我が国経済の活性化のみならず、グローバルな視点での適切な国際分業体制の構築にも資するものと考えられる。

2. 豊かで安心できる経済社会の創造経済や雇用の先行きや、少子・高齢社会への移行に伴う将来のくらしに対する不安、豊かさが実感できないことに対する不満を解消していくためには、豊かで安心できる経済社会の創造を着実に進める必要がある。そのためには、1. で述べた方向に沿って、経済社会の活力を高めるとともに、その成果が生活に反映されることが重要である。また、国民、企業、政府間の適切な役割分担による、自立のための効率的な社会的支援システムを構築し、少子・高齢社会の安心の確保を図るとともに、防災等の国民生活の安全に係る基盤の強化や環境と調和した持続可能な経済社会の構築を

図っていくことが重要である。さらに、人々の意識の変化に対応し、一人一人の個性が尊重され、自立した個人が自己責任の下に多様な選択を行い、多様な役割を持って、参加できるような公正な機会が保障された社会を目指すことが重要である。このような自立した個人や企業の自由な活動は、経済社会の活性化にもつながるものと考えられる。

3. 地球社会への参画グローバリゼーションの進展の中で、我が国が、地球社会における責任と役割の増大にこたえていくためには、地球社会への積極的な参画を目指す必要がある。そのためには、我が国経済が世界の動向と分かれ難く結び付いていることにかんがみ、内外に開かれた経済社会を目指し、制度・仕組みの国際的調和や、国際的なルール作りに積極的に取り組んでいく必要がある。これは、我が国企業がグローバルな市場で自由に活動することを通じ、我が国経済の活性化をもたらすのみならず、適切な国際分業を通じ、調和ある対外経済関係の達成にも資するものと考えられる。さらに、地球社会の一員として、地球的規模での取組が求められている問題についても、その望ましい対処の方針や、その実現のための道筋について自らの考え方を率直に提示した上で、世界の国々と意見交換を行うことにより、地球社会の発展に積極的に参画していくことが重要である。

以上の3つを基本的方向としつつ、これらを共通して支える基本的政策として、以下のものが必要である。

4. 発展基盤の確立新しい経済社会を支える基盤として、1人材の育成、2科学技術の創造、3情報通信の高度化、4社会資本整備の推進を図ることが重要である。(1) 人材の育成

経済社会の発展を支える基礎となるのは人である。異なる価値観に対する受容力を持ち、創造性に富み、様々な経済社会の変化に柔軟に対応し、経済社会を活性化し、安全で豊かなくらしを支え、地球社会に参画できるような人材を育成することが重要な課題である。このため、多様な生涯学習機会の提供と自己啓発への支援等により、個人の能力が発見、開発、発揮、評価される「能力開花型社会」の構築を目指す必要がある。

(2) 科学技術の創造新しい経済社会のフロンティアを生み出す源泉は、科学技術である。従来型の経済成長が終焉した中で、新たな成長を切り拓いていくためには、創造的な研究開発を推進するとともに、科学技術を社会のニーズと結び付け、新規産業を創出していくこと等を通じた、経済フロンティアの拡大や豊かで安心できるくらしの実現を図ることが必要である。このため、研究開発という知的創造活動を効果的・効率的に行うことを可能とする知的資本の整備を進め、「科学技術創造立国」を目指す必要がある。

(3) 情報通信の高度化情報通信の高度化は、現在の世界の大きな潮流の一つであり、その潮流に適切に対応することが21世紀の経済社会の発展を切り拓くために不可欠である。情報通信の高度化は、新たな産業や雇用を創出し、豊かで活力ある経済社会の実現に寄与するものと期待されることから、「高度情報通信社会」の早期構築を目指す必要がある。

(4) 社会資本整備の推進社会資本は、快適な生活環境の形成、安全で安心できる生活の確保、新しい日本経済の発展基盤の構築といった観点から経済社会発展の基礎となるものであり、その整備の推進を図る必要がある。

5. 行財政改革の推進等上で見た潮流変化や課題に適切に対応し、構造改革を推進していくに当たっては、公的部門自らも改革を進めなければならない。このため、変化への対応力に富み、簡素で効率的な行政を確立すべく、行政改革を引き続き推進する必要がある。また、財政の対応力を回復するため、財政改革を一層強力に推進していく必要がある。



## 複数の評価基準からの総合評価

### § 3.1 因果探索に基づく入力と出力の分類

因果探索とは、観測データを用いて、そのデータ群の因果グラフ（複数の観測データにおいて、それぞれの値がお互いに及ぼしあっている影響の度合いを構造的に示したもの）を導出するための教師なし学習のことである。

また、類似する手法として因果推論が挙げられるが、因果推論では因果関係の向きが既知である場合にその因果関係が本当に有意であるのかをデータから分析する手法であるのに対し、因果探索は因果関係が不明かつ因果関係の向きも不明であるデータ群に対して、それらの間に因果関係が成立するかを導く手法である。

例えば、「ある小売店 A でアイスクリームの安売りを行った際にアイスクリームの売り上げが向上した。また、同日の小売店 A の来客数は前日より 100 人多かった」というケースがあったとする。このとき、アイスクリームの安売りを行ったことが売り上げの向上につながったかどうかを調べるのが因果推論である。これに対して、アイスクリームが安かつたから来客数が増加したのか、来客数が多かったためにアイスクリームの売り上げが向上したのかという因果の方向性も含めて分析を行うのが因果探索である。

このような特徴を持つため、因果探索は適用されるデータの分野に対しての制約が少なく、様々な分野のデータに適用することができる。それゆえ、因果探索を用いた応用研究も盛んにおこなわれており、疫学、経済学、神経科学、化学、医学をはじめとした幅広い分野のほか土木計画学 [?] の研究でも用いられている。

#### LiNGAM

近年、因果探索の手法における研究が活発化したことで、因果探索における様々なモデルが提唱されている。代表的なものとしては独立主成分分析の手法を用いたセミパラメトリックなもので、非時系列データに対しても適用可能な LiNGAM が挙げられる。LiNGAM とは、一般的に以下の式 (3.1) のように定式化され、

$$x_i = \sum_{i \neq j} b_{ij} x_j + e_i \quad i, j = 1, \dots, p \quad (3.1)$$

以下のような仮定を用いることで因果グラフを導出する手法である [?].

1. 外生変数と内生変数をつなぐ関数は線形関数とする。（内生変数とは実際に観測されている変数、外生変数とは内生変数以外の変数で内生変数のそれぞれに関する未知の値である。）

2. 外生変数の分布は非ガウス連續分布とする.
3. 因果グラフは非巡回とする.
4. 外生変数は互いに独立とする.

ここで、図??のような内生変数が4つの因果グラフがあると考える。このとき、仮定3があることによって4つの内生変数のうち、どの変数からも因果的影響を受けない変数が少なくとも1つ必ず同定される。

$$\begin{bmatrix} x_1 \\ x_2 \\ x_4 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_4 \\ x_3 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_4 \\ e_3 \end{bmatrix} \quad (3.2)$$

つまり、式(3.2)のように因果的影響を受ける場合にその値、受けない場合には0を入れたパス係数行列を考えると必ず右肩が逆三角形に全て0になる行列となる。そのため、式(3.2)における $x_1$ のように全てのパスに対する係数が0となる内生変数を因果グラフから除外し、再度パス係数行列を求めるという操作を繰り返すことによって未知である因果グラフを同定することが可能になる。

また、上記の同定法を成立させるにあたって、仮定4がなくてはならない。前述のとおり、LiNGAMにおける因果関係の同定では内生変数同士の因果関係のみに着目して因果グラフの最も外側に位置する内生変数を順に除外する方法をとるため、内生変数同士の間に成立する因果関係以外の因果関係が内生変数間に発生してはならない。

ここで、もし外生変数同士が独立ではなければ外生変数同士の間に因果関係が生じてしまい、図??の例に見られるように内生変数同士がそれぞれの内生変数に関わる外生変数同士の因果関係を介在として内生変数間に存在しない新たな因果関係を持つてしまう。このような場合には前述のような因果関係の同定法が成り立たなくなるため、LiNGAMにおける仮説4は必ず必要となる。

## Direct-LiNGAM

前述のようなアルゴリズムによって内生変数間の因果関係を推定するLiNGAMであるが、推定時の計算方法の違いによって現在までにいくつかのアプローチが提唱されている。代表的な例として、独立成分分析によるアプローチであるICA-LiNGAMや回帰分析と独立性評価によるアプローチであるDirect-LiNGAMなどが挙げられる。その中でも、本研究で取り扱うDirect-LiNGAMに関する解説を行う。Direct-LiNGAMによるアプローチの基本的な考え方

- 観測変数群から2変数を取り出しそれらの変数間に成り立つ因果関係を同定することを繰り返して観測変数群全体における因果の始まりとなる変数を探す。
- その変数を観測変数群から除外し、残った変数のみで再度、観測変数群を形成する。

という2つの操作を観測変数群に属する変数が存在しなくなるまで繰り返すことによって元の観測変数群の因果グラフを同定するというものである。例として、観測変数群から2変数 $x_1, x_2$ を取り出し、以下の構造方程式モデルが背後にあるものと想定する。

$$\begin{cases} x_1 = e_1 \\ x_2 = b_{21}x_1 + e_2 \end{cases} \quad (3.3)$$

ここで  $e_1, e_2$  は互いに独立かつ非ガウス分布に従い,  $b_{21} \neq 0$  とする. これについて単回帰分析を行うことによって, 因果順序の同定を行う. まず  $x_2$  を目的変数,  $x_1$  を説明変数として回帰する場合を考える. この場合元の構造方程式モデルの第 2 式がそのまま成り立つことになる. そのため, 回帰残差は  $e_2$  となりこれは  $x_1 = e_1$  と独立となる. 一方,  $x_1$  を目的変数,  $x_2$  を説明変数とした場合, 回帰残差  $r_1$  は

$$r_1 = \left\{ 1 - \frac{b_{21} \text{cov}(x_1, x_2)}{\text{var}(x_2)} \right\} e_1 - \frac{b_{21} \text{var}(x_1)}{\text{var}(x_2)} e_2 \quad (3.4)$$

となり  $e_2$  の項が出てくる. 一方冒頭の構造方程式に戻ると,  $x_2$  は式として  $e_2$  を含むので, この回帰残差と説明変数  $x_2$  とは従属する. この従属性の成立に関しては前述の仮定 2 にて示した「外生変数が非ガウス分布とする」というきまりに基づいており, 以下に示すダルモア・スキットビッチの定理を用いている.

## VAR モデル

VAR モデルは多変量時系列データを分析する統計的手法であり, 柔軟性や汎用性から将来予測の分野などで特に注目を集めている. この手法では複数の変数が同時に相互作用する複雑なシステムを捉えることができる. 経済の分野においては, VAR モデルは異なる経済指標や変数間の相互作用を捉え, 経済の動向や政策の影響を理解するために頻繁に活用されている. 為替変動や売上高成長率の予測 [16] などの研究があり, 経済の分野でも幅広く利用されていることがわかる.

VAR モデルの中でも一般的な VAR モデル (p) の構造は以下のようになる.

$$Y_t = A_1 Y_{t-1} + A_2 Y_{t-2} + \dots + A_p Y_{t-p} + \varepsilon_t \quad (3.5)$$

$$\begin{aligned} \begin{bmatrix} Y_{1,t} \\ Y_{2,t} \\ \vdots \\ Y_{k,t} \end{bmatrix} &= \begin{bmatrix} a_{11,1} & a_{12,1} & \dots & a_{1k,1} \\ a_{21,1} & a_{22,1} & \dots & a_{2k,1} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1,1} & a_{k2,1} & \dots & a_{kk,1} \end{bmatrix} \begin{bmatrix} Y_{1,t-1} \\ Y_{2,t-1} \\ \vdots \\ Y_{k,t-1} \end{bmatrix} \\ &+ \begin{bmatrix} a_{11,2} & a_{12,2} & \dots & a_{1k,2} \\ a_{21,2} & a_{22,2} & \dots & a_{2k,2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1,2} & a_{k2,2} & \dots & a_{kk,2} \end{bmatrix} \begin{bmatrix} Y_{1,t-2} \\ Y_{2,t-2} \\ \vdots \\ Y_{k,t-2} \end{bmatrix} + \dots \\ &+ \begin{bmatrix} a_{11,p} & a_{12,p} & \dots & a_{1k,p} \\ a_{21,p} & a_{22,p} & \dots & a_{2k,p} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k1,p} & a_{k2,p} & \dots & a_{kk,p} \end{bmatrix} \begin{bmatrix} Y_{1,t-p} \\ Y_{2,t-p} \\ \vdots \\ Y_{k,t-p} \end{bmatrix} + \begin{bmatrix} u_{1,t} \\ u_{2,t} \\ \vdots \\ u_{k,t} \end{bmatrix} \end{aligned}$$

近年では VAR モデルは単体として使われることは少なく、構造 VAR モデルなど拡張されて使われることが多いが、未だに最も基本的な多変量時系列データを分析するための手法として注目されている。経済分析に特に頻繁に用いられるのは構造 VAR である。構造 VAR モデルの構造については以下の式にして示す。

$$Y_t = A_1 Y_{t-1} + A_2 Y_{t-2} + \cdots + A_p Y_{t-p} + BX_t + \varepsilon_t \quad (3.6)$$

構造 VAR モデルでは VAR モデルに外部説明変数を取り入れることで経済ショックが内政変数にもたらす動学的な影響をインパルス応答などで数量的に表すことができる。また変数間の因果関係の方向性や即時効果の程度を知ることができる。これにより、金融政策の効果検証や政策分析への研究に適用されることがあり、構造 VAR モデルを用いた日本経済の資産蓄積、所得分配、負債の動態分析などの研究が行われている [17]。

## VAR-LiNGAM

因果探索も計量経済モデル計量経済モデルの一種として LiNGAM によるものがある。VAR-LiNGAM は因果探索手法の一つであり、時系列データに対する因果関係の推定を行うためのモデル化手法である。VAR-LiNGAM は 2010 年に提唱され [18]、様々な分野で活用されている。VAR-LiNGAM は VAR モデルと Linear Non-Gaussian Acyclic Model (LiNGAM) を組み合わせたものであり、図のような形になる。LiNGAM は因果探索のセミパラメトリックなアプローチの手法であり定式化は以下のようになる。

$$x_i = \sum_{i \neq j} b_{ij} x_j + e_i \quad (i = 1, \dots, p) \quad (3.7)$$

そしてこれは行列表現で以下のように表すことができる。

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix}, \quad B = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{p1} & b_{p2} & \cdots & b_{pp} \end{bmatrix}, \quad E = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_p \end{bmatrix}$$

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{p1} & b_{p2} & \cdots & b_{pp} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_p \end{bmatrix}$$

VAR-LiNGAM ではこれに VAR モデルを加える。VAR-LiNGAM のイメージは図に示す。VARLiNGAM における仮定として VAR-LiNGAM の一般的な定式化としては以下のようになる

$$x(t) = \sum_{\tau=0} b_{ij} B_r x(t - \tau) + e(t) \quad (3.8)$$

これは VAR モデルと似た形になっているが, 時間差を表す  $\tau$  が 0 から始まっている.  $B_0$  は LiNGAM における  $B$  のように現時点での多変数からの因果を表す係数であり,  $B_\tau (\tau = 1, \dots, k)$  は過去時点の  $x(t - \tau)$  からの直接的な影響を表す係数であり, これにより現在のベクトル変数の値  $x(t)$  を現時点の他の変数だけでなく過去時点のベクトル変数も用いて推定することができる.

**モデルの推定方法** VAR-LiNGAM の推定方法はいくつかあるがここでは VAR モデルとして推定を行ったあとに通常の LiNGAM モデルの推定を行う 2 段階法について紹介する. まず, 時系列ベクトル  $x(t)$  に対する通常の VAR モデルを以下のように構築する.

$$x(t) = \sum_{\tau=1}^k M_\tau x(t - \tau) + n(t) \quad (3.9)$$

これから最小二乗推定量  $M^\tau$  を計算する. ここで求めた  $M^\tau$  を代入することで搅乱項の推定値  $n^t$  を計算する.

$$\hat{n}(t) = x(t) - \sum_{t-\tau}^k \hat{M}_\tau x(t - \tau) \quad (3.10)$$

$\hat{n}(t)$  に対する LiNGAM 推定を行う. これにより因果係数行列の推定値  $\hat{B}_0$  が求まる.

$$\hat{B}_\tau = (I - \hat{B}_0) \hat{M}_\tau \quad (3.11)$$

上で求めた  $\hat{M}_\tau$  と  $\hat{B}_0$  から  $\hat{B}_\tau$  を求める. サンプル数を大きくすると正しい値に収束することが VAR-LiNGAM の元論文では示されている.

### ダルモア・スキットビッチの定理

2 つの確率変数  $y_1, y_2$  が, 互いに独立な確率変数  $s_i (i = 1, \dots, q)$  の線形和で下記のように表されているとする. この時, もし  $y_1, y_2$  が独立なら,  $\alpha_j \beta_j \neq 0$  となるような変数  $s_j$  はガウス分布に従う

$$y_1 = \sum_{i=1}^q \alpha_i s_i \quad (3.12)$$

$$y_2 = \sum_{i=1}^q \beta_i s_i \quad (3.13)$$

上記の考察から, 両方のパターンで回帰分析を行い残差と説明変数の独立性を判定することで因果の向きを推定することが可能となる. なお独立性の評価には相互情報量という量を用いる. この量が 0 となるときに独立であると判定するが, 実際には推定誤差があり正確には 0 にはならないため, 相互情報量が 0 に近い方を独立とみなして因果の順序を決定する.

計量経済学は経済現象を数理・統計的手法を用いて分析・モデル化する学問分野である. 計量経済モデルとは, この計量経済学の考え方に基づき, 経済のメカニズムや変数間の関係性を数学的な式や方程式系によって定式化したものを指す.

計量経済モデルには大きく分けて、経済構造そのものを表現する構造形モデルと、特定の経済変数間の総合的な相関関係を表現する縮小形モデルがある。前者は経済主体の意思決定や市場の働きなど、経済の根本メカニズムを体系的にモデル化する点に特徴がある。後者は実証分析に基づいて、複数経済変数間の相関関係を簡潔に定式化することに特に重要である。

これらのモデルは消費者や企業の最適化行動の仮定、市場の均衡条件の設定、複数の内生変数や外生変数の導入などの方式で構築される。その際、個々の変数の挙動はパラメータによって定量的に規定される。これらパラメータは実際の時系列データや断面データを用いた回帰分析などの計量的手法に基づいて推定されることが多く、可能な限り実態を反映するよう設定される。

構築されたモデルは政策変更がマクロ経済・ミクロ経済に与える影響の定量的分析や経済変数の将来予測に利用される。特に政策当局にとって異なる政策オプションを比較検討する上での有用性が高いとされている。ここでは現在使われている計量経済モデルの例について紹介する。

### ブートストラップ法による係数の有意性検定

VAR-LiNGAM 含め LiNGAM の手法において推定された因果係数が 0 と有意に異なるかどうかは非常に重要である。そこでブートストラップサンプリングを用いて係数の経験分布を求めることで有意性を確認する必要がある。分散を示すシンプルな統計量を提案し、それを用いて仮説検定を行う方法を紹介する。

### 分散統計量

$\tau = 0$  の場合 (同時因果効果) と  $\tau > 0$  の場合に分けて説明する。 $x_i(t)$  から  $x_j(t)$  への同時因果効果を表す係数の分散統計量を以下の式で定義する。

$$S_0 = [B_0]_{ij}^2 \cdot \frac{\text{var}(x_i(t))}{\text{var}(x_j(t))} \quad (3.14)$$

過去からの因果効果を検証する際は、通常ある 1 つの  $\tau > 0$  のみでの効果には関心はなく、 $\tau > 0$  全体での因果効果を見る。よって  $x_i(t - \tau), \tau > 0$  から  $x_j(t)$  への総合的な因果効果の分散統計量を次の式で定義する。

$$S_{\text{lag}} = \frac{\text{var}(\sum_{\tau > 0} [B_\tau]_{ij} x_i(t - \tau))}{\text{var}(x_j(t))} \quad (3.15)$$

### 検定方法

因果係数が 0 という帰無仮説の本での各統計量の経験的分布を求めるために、サロゲート法によるブートストラップ法を用いる。サロゲート法によるブートストラップ法ではブートストラップの際に元の系列  $x_i(t)$  の時間的順序をランダムに並び変えたものをサンプリングする。これによって得られたサロゲートデータから統計量  $S$  の推定値である  $S^*$  を計算する。そしてブートストラップから得られる  $S^*$  の分布の上位  $\alpha$  分位点  $c \cdot \text{top}\alpha$  を求め、元データから算出した  $S^*$  がこれを超えれば帰無仮説を棄却する。

## § 3.2 データ包絡分析の種類と効率の評価

DEA の分析対象となる事業体は銀行, デパート, 病院, 都道府県, 学校, 等のように多種多様である. 事業体が  $n$  個存在するとし, それらを  $DMU_1, DMU_2, \dots, DMU_n$  と番号づける. 次に, それぞれの生産活動に共通した投入(入力)項目と産出(出力)項目を選ぶ. ごく一般的な選び方として次のような方針をとる

- まず,  $n$  個の事業体はある程度独立して経営を行っていると考える. 数学的にいようと, 全ての事業体が一時独立性を満たしていると考える.
- 投入項目, 産出項目とも数値データを準備する. 一般に, 入出力値は原則としてすべて正とする. また, すべての事業体は同数の入出力を持っていると考える. 入出力が各事業体によって違う場合は双対比較ができないので, DEA では通常取り扱わない.
- 投入項目, 産出項目の選定にあたっては, 企業のトップや政策決定者に直接聞いてみるのが一番よく, 経営者が現実の意思決定に際して, どのような項目に注目してそれぞれの事業体の評価を判断しているのかを確かめる必要性がある. もしそのような機会が得られない場合は, 自分が試してみたいと思う入出力項目を選べばよい
- 原則として入力に関していえば, 値の小さいものほど好ましく, 出力に関しては大きいものほど好ましいものを選ぶ.
- 投入項目, 産出項目の数値の単位は任意に取ってよい.

いま,  $m$  個の投入項目と  $s$  個の産出項目が選定され,  $DMU_j$  の投入データを  $x_{1j}, x_{2j}, \dots, x_{nj}$  産出(出力)データを  $y_{1j}, y_{2j}, \dots, y_{nj}$  とする. 各事業体の生産活動に関するデータを縦に並べて行列を作り, 入力データ行列  $X$  と出力データ行列  $Y$  とする. それらは次のような  $(m \times n)$  型,  $(s \times n)$  型の行列で表される.

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix}, \quad Y = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1n} \\ y_{21} & y_{22} & \cdots & y_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ y_{s1} & y_{s2} & \cdots & y_{sn} \end{bmatrix} \quad (3.16)$$

複数の事業体それぞれについて比率尺度で効率性を測定していくが, 対象になっている事業体を  $k$  番目とし,  $DMU_k$  と書くことにする. 以下, 添字  $k$  は  $1, 2, \dots, n$  のどれかを指すものとする. 入力につける重みを  $v_i$  ( $i = 1, 2, \dots, m$ ) 出力につける重みを  $u_r$  ( $r = 1, 2, \dots, s$ ) として, その値を次の分数計画問題を解くことによって定める.

$$\text{目的関数} \quad \text{Max} \quad \theta = \frac{u_1 y_{1k} + u_2 y_{2k} + \dots + u_s y_{sk}}{v_1 x_{1k} + v_2 x_{2k} + \dots + v_m x_{mk}} \quad (3.17)$$

$$\text{制約式} \quad \frac{u_1 y_{1j} + \dots + u_s y_{sj}}{v_1 x_{1j} + \dots + v_m x_{mj}} \leq 1 \quad (j = 1, 2, \dots, n) \quad (3.18)$$

$$\quad \quad \quad (3.19)$$

$$v_1, v_2, \dots, v_m \geq 0, u_1, u_2, \dots, u_s \geq 0$$

この式は、その制約式で、仮想的に考えられた総入力と総出力の比をすべての事業体の生産活動において、1以下に抑えるようにモデル化されている。その上で、 $k$ 番目の事業体の効率値  $\theta$  を最大化するように重み  $v_i$  と  $u_r$  を決めてある。したがって、最適な  $\theta$  の値 ( $\theta^*$ ) は 100% かそれ以下である。下限は 0% と考えてよい。

上の分数計画問題に対して次の線形計画問題を考える。

<入力指向 >

$$\begin{aligned} \text{目的関数} \quad & \text{Max} \quad \theta = u_1 y_{1k} + \dots + u_s y_{sk} \\ \text{制約式} \quad & v_1 x_{1k} + \dots + v_m x_{mk} = 1 \\ & u_1 y_{1j} + \dots + u_s y_{sj} - (v_1 x_{1j} + \dots + v_m x_{mj}) \leq 0 \quad (j = 1, 2, \dots, n) \\ & v_1, v_2, \dots, v_m \geq 0, u_1, u_2, \dots, u_s \geq 0 \end{aligned} \quad (3.20)$$

<出力指向 >

$$\begin{aligned} \text{目的関数} \quad & \text{Min} \quad \theta = v_1 x_{1k} + v_2 x_{2k} + \dots + v_m x_{mk} \\ \text{制約式} \quad & u_1 y_{1k} + u_2 y_{2k} + \dots + u_s y_{sk} = 1 \\ & u_1 y_{1j} + \dots + u_s y_{sj} - (v_1 x_{1j} + \dots + v_m x_{mj}) \leq 0 \quad (j = 1, 2, \dots, n) \\ & v_1, v_2, \dots, v_m \geq 0, u_1, u_2, \dots, u_s \geq 0 \end{aligned} \quad (3.21)$$

線形計画問題は普通の線形計画法のアルゴリズムによって解くことができる。したがって、実用的な解法としては分数計画問題より線形計画問題を解く方が良い。次に式 (1.3) の性質についてまとめてみる。

- 式 (1.3) の最適解を  $(v^*, u^*)$  とし、目的関数値を  $\theta^*$  とする。そのとき  $\theta^* = 1$  ならば  $DMU_k$  は効率的であるという。逆に  $\theta^* < 1$  の場合でも、スラックと呼ばれる入力の余剰や出力の不足が発生していることがあるので注意を要する。
- $DMU_k$  が  $\theta^* = 1$  (非効率) のとき、式 (1.3) の制約の中には重み付け  $(v^*, u^*)$  に対して、次のような等式が成立する  $j$  が存在する。

$$R_k = \{j : \sum_{r=1}^s u_r^* y_{rj} = \sum_{i=1}^m v_i^* x_{ij}, \quad j = 1, \dots, n\} \quad (3.22)$$

この集合  $(R_k)$  は  $DMU_k$  を非効率と判定させるもとになっている事業体群である。その意味で、 $DMU_k$  に対する参照集合という。この参照集合  $(R_k)$  は効率的フロンティアの一部を形成している。容易にわかるように、この集合に属する事業体はそれ自体が効率的である。

式 (1.3) をより短く書くと

$$\text{目的関数} \quad \text{Max} \quad \sum_{r=1}^s u_r y_{rk}$$

$$\text{制約式} \quad - \sum_{i=1}^m v_i x_{ij} + \sum_{r=1}^s u_r y_{rj} \leq 0 \quad (j = 1, 2, \dots, n) \quad (3.24)$$

$$\sum_{i=1}^m v_i x_{ik} = 1 \quad (3.25)$$

$$v_i \geq 0 \quad (i = 1, 2, \dots, m), \quad u_r \geq 0 \quad (r = 1, 2, \dots, s)$$

となる。その双対形は

$$\text{目的関数} \quad \text{Min} \quad \theta \quad (3.26)$$

$$\text{制約式} \quad - \sum_{j=1}^n x_{ij} \lambda_j + \theta x_{rk} \leq 0 \quad (i = 1, 2, \dots, m) \quad (3.27)$$

$$\sum_{j=1}^n y_{rj} \lambda_j \geq u_{rk} \quad (3.28)$$

$$\lambda_j \geq 0 \quad (j = 1, 2, \dots, n), \quad \theta : \text{制約なし}$$

となる。この式(1.5), (1.6)がCCRモデルと呼ばれるDEAモデルである。式(1.5)と(1.6)の目的関数値は双対定理により一致し、DEA効率値を示している。なお、式(1.6)において $\theta$ を制約なしとしたのは、式(1.5)において $\sum_{n=1}^m v_i x_{ik} = 1$ が等式となっているためであるが、式(1.6)を解く際に $\theta \geq 0$ としても問題ではない。

次に $\theta^* = 1$ の場合でも、入力の余剰 $d_i^k$  ( $i = 1, 2, \dots, m$ )と出力の不足 $d_r^y$  ( $r = 1, 2, \dots, s$ )はそれぞれ

$$d_i^x = \theta x_{ik} - \sum_{j=1}^n x_{ij} \lambda_j \quad (i = 1, 2, \dots, m)$$

$$d_r^y = \sum_{j=1}^n y_{rj} \lambda_j - y_{rk} \quad (r = 1, 2, \dots, s)$$

で表される。 $d_i^k$ と $d_r^y$ は次の式を解くことにより得られる。

$$\text{目的関数} \quad \text{Max} \quad \sum_{i=1}^m d_i^x + \sum_{r=1}^s d_r^y \quad (3.29)$$

$$\text{制約式} \quad \sum_{j=1}^n x_{ij} \lambda_j + d_i^x = \theta^* x_{ik} \quad (i = 1, 2, \dots, m)$$

(3.30)

$$\sum_{j=1}^n y_{rj} \lambda_j - d_i^y = y_{rk} (r = 1, 2, \dots, s) \quad (3.31)$$

$$\begin{aligned} \lambda_j &\geq 0 (j = 1, 2, \dots, n), & d_i^x &\geq 0 (i = 1, 2, \dots, n) \\ d_r^y &\geq 0 (r = 1, 2, \dots, s) \end{aligned} \quad (3.32)$$

ここで  $\theta^*$  は式 (1, 6) の最適目的関数値を示している。よって第 1 段階として式 (1, 6) を解き  $\theta^*$  を求めた後、第 2 段階として式 (1, 7) を解くことになる。式 (1, 6) より得られる最適解  $(\theta^*, \lambda_1^*, \dots, \lambda_n^*)$  と式 (1, 7) から得られるスラックの最適解  $(d_1^{x*}, \dots, d_m^{x*}, d_1^{y*}, \dots, d_s^{y*})$  を合わせて DEA 最適解と呼ぶ。

DEA(CCR) 効率性の定義として、DEA 最適解において  $\theta^* = 1$  かつ全てのスラックが 0 の場合、効率的である。それ以外のときは、非効率的である。

### § 3.3 データ包絡分析による改善策の導出

DEA は事業体の効率性を示すだけでなく、同時に非効率的な事業体の改善案についても示すことができる。 $DMU_k$  が非効率であるとき式 (1.6) に対応する参照集合は

$$R_k = \{j \mid \lambda_j^* > 0, j = 1, \dots, n\} \quad (3.33)$$

として表される。この式○は式○と同等である。

参照集合  $R_k$  に属する事業体は効率的である。 $R_k$  に属する事業体の存在が  $DMU_k$  を非効率とさせる原因であり、この集合を非負結合させたものが  $DMU_k$  の効率値を決定する効率的フロンティアを形成する。

式○において  $DMU_k$  の効率値は DEA 最適解を  $(\theta^*, \lambda_1^*, \dots, \lambda_n^*, d_1^{x*}, \dots, d_m^{x*}, d_1^{y*}, \dots, d_s^{y*})$  とすると、この効率的フロンティアの上のダミー事業体  $(\theta^* x_{ik}, \dots, \theta^* x_{mk}, y_{1k}, \dots, y_{sk})$  と双対比較することで決定される。ここで  $\theta^* x_{ik}$  と  $u_{rk}$  は次のように表される。

$$\begin{aligned} \theta^* x_{ik} &= \sum_{j \in R_k} \lambda_j^* x_{ij} + d_i^{x*} (i = 1, 2, \dots, m) \\ y_{rk} &= \sum_{j \in R_k} \lambda_j^* x_{rj} - d_r^{y*} (r = 1, 2, \dots, s) \end{aligned}$$

すなわち、 $DMU_k$  は入力を  $\theta^*$  倍に縮小し、さらに余剰を除去する。出力は不足を補うことで、効率的な事業体となる。 $DMU_k$  の入力の過剰量 ( $\Delta x_{ik}$ )、出力の不足量 ( $\Delta y_{rk}$ ) は次のように表される。

$$\begin{aligned} \Delta x_{ik} &= x_{ik} - (\theta^* x_{ik} - d_i^{x*}) = (1 - \theta^*) x_{ik} + d_i^{x*} (i = 1, 2, \dots, m) \\ \Delta y_{rk} &= d_r^{y*} \end{aligned}$$

よって,

$$x_{ik} \Rightarrow x_{ik} - \Delta x_{ik} = \theta^* x_{ik} - d_i^{x*} \quad (i = 1, 2, \dots, m)$$

$$y_{rk} \Rightarrow y_{rk} + \Delta y_{rk} = y_{rk} + d_r^{y*}$$

とすれば,  $DMU_k$  は効率的な事業体となる. ただし, 非効率な事業体を効率的に改善する方法は 1 つではないことに留意する.

表 3.1: 6 事業体の 2 入力・1 出力の例

事業体	A	B	C	D	E	F
入力 $x_1$	4	4	4	3	2	6
入力 $x_2$	2	3	1	2	4	1
出力 $y$	1	1	1	1	1	1

表 3.1 に示すように A-F の 6 事業体があり, それぞれが 2 入力・1 出力をもつとする. ただし, 例を簡略化するため, 出力値はいずれも 1 である.

1. 事業体 A の効率性を求める CCR モデルは,  $v_1, v_2, u$  を入力と出力の重み付けとして, 次のように定式化される.

$$\text{目的関数} \quad \text{Max } u$$

$$\text{制約式} \quad -4v_1 - 2v_2 + u \leq 0 \quad (\text{A})$$

$$-4v_1 - 3v_2 + u \leq 0 \quad (\text{B})$$

$$-4v_1 - v_2 + u \leq 0 \quad (\text{C})$$

$$-3v_1 - 2v_2 + u \leq 0 \quad (\text{D})$$

$$-2v_1 - 4v_2 + u \leq 0 \quad (\text{E})$$

$$-6v_1 - v_2 + u \leq 0 \quad (\text{F})$$

$$4v_1 + 2v_2 = 1$$

$$v_1 \geq 0, v_2 \geq 0, u \geq 0 \quad (3.34)$$

$$(3.35)$$

この最適解は,  $v_1^* = 0.167, v_2^* = 0.167, u^* = 0.833$  であり, 目的関数値, すなわち A の DEA 効率値は 0.833 となる. A の参照集合は  $R_A = \{C, D\}$  であり, C, D の存在が A を非効率にしていることがわかる.

次に,  $\theta, \lambda$  を双対変数として双対系をとると,

$$\text{目的関数} \quad \text{Min } \theta$$

$$\text{制約式} \quad -4\lambda_A - 4\lambda_B - 4\lambda_C - 3\lambda_D - 2\lambda_E - 6\lambda_F + 4\theta \geq 0$$

$$-2\lambda_A - 3\lambda_B - \lambda_C - 2\lambda_D - 4\lambda_E - \lambda_F + 2\theta \geq 0$$

$$\begin{aligned} \lambda_A + \lambda_B + \lambda_C + \lambda_D + \lambda_E + \lambda_F &\geq 1 \\ \lambda_j \geq 0 (j = A, B, \dots, F), \quad \theta : \text{制約なし} \end{aligned}$$

のようになる。この最適解は、 $\lambda_A^* = \lambda_B^* = 0, \lambda_C^* = 0.333, \lambda_D^* = 0.677, \lambda_E^* = \lambda_F^* = 0, \theta^* = 0.833$  となる。これらの最適値から次の関係が分かる。

$$\begin{aligned} 0.833 * (\text{A の入力}) &= 0.333 * \text{C の入力} + 0.677 * (\text{D の入力}) \\ (\text{A の入力}) &= 0.333 * (\text{C の入力}) + 0.677 * (\text{D の入力}) \end{aligned}$$

図○からわかるように、事業体 A は OA に直線を引いたとき、効率フロンティアである CD の直線と交わる点  $A_1$  (A の入力を一様に 0.833 倍に縮小した点) において効率的になる。この点は CD 間の比率を表している。すなわち、効率的な事業体 C と D をそれぞれどの程度参照しているかがわかる。次に、スラックの最適解を求めるために、以下の式を解く。

$$\begin{aligned} \text{目的関数} \quad & \text{Max} \quad d_1^x + d_2^x + d_1^y \\ \text{制約式} \quad & 4\lambda_A + 4\lambda_B + 4\lambda_C + 3\lambda_D + 2\lambda_E + 6\lambda_F + d_1^x = 4 * 0.833 \\ & 2\lambda_A + 3\lambda_B + \lambda_C + 2\lambda_D + 4\lambda_E + \lambda_F + d_2^x = 2 * 0.833 \\ & \lambda_A + \lambda_B + \lambda_C + \lambda_D + \lambda_E + \lambda_F - d_1^y = 1 \\ & \lambda_j \geq 0 (j = A, B, \dots, F), \quad d_1^x \geq 0, d_2^x \geq 0, d_1^y \geq 0 \end{aligned}$$

この最適解は、 $d_1^{x*} = d_2^{x*} = d_1^{y*} = 0, \lambda_A^* = \lambda_B^* = 0, \lambda_C^* = 0.333, \lambda_D^* = 0.677, \lambda_E^* = \lambda_F^* = 0$  となる。このように、事業体 A の DEA 最適解は  $\theta^* = 0.833, d_1^{x*} = d_2^{x*} = d_1^{y*} = 0, \lambda_A^* = \lambda_B^* = 0, \lambda_C^* = 0.333, \lambda_D^* = 0.677, \lambda_E^* = \lambda_F^* = 0$  となる。なお、A と同様に、事業体 B も非効率的であることがわかる。

表 3.2: DEA の分析結果

事業体	DEA 効率値	参照集合	$v_1^*$	$v_2^*$	$u^*$	$d_1^{x*}$	$d_2^{x*}$	$d_1^{y*}$
A	0.833	$C(\lambda_C^* = 0.333), D(\lambda_D^* = 0.677)$	0.167	0.167	0.833	0	0	0
B	0.727	$D(\lambda_D^* = 0.909), E(\lambda_E^* = 0.091)$	0.182	0.091	0.727	0	0	0
C	1	$C(\lambda_C^* = 1)$	0.200	0.200	1	0	0	0
D	1	$D(\lambda_D^* = 1)$	0.250	0.125	1	0	0	0
E	1	$E(\lambda_E^* = 1)$	0.500	0	1	0	0	0
F	1	$C(\lambda_C^* = 1)$	0	1	1	2000	0	0

- 事業体 C,D,E は、A や B の参照集合になっているので、効率的である。例えば C の場合、 $\lambda_A^* = \lambda_B^* = 0, \lambda_C^* = 1, \lambda_D^* = \lambda_E^* = \lambda_F^* = 0$  となり、C 自身が参照されている。これは C 自身が効率的であり、参照集合も C となることを意味している。D と E の場合も同様である。

3. F に関する DEA 最適解は  $\theta^* = 1$  であるが, 入力 1 に対するスラック  $d_1^{x*}$  が 2 であるので, DEA 効率性の定義により F は非効率であると判定する. F を効率的にするには, C のレベルまで入力 1 ( $x_1$ ) を減らす必要がある.

最後に, DEA による分析結果を表 3.2 にまとめた.



## 提案手法

### § 4.1 因果探索による入出力の選定とデータの整列

本研究では、献立作成を多目的最適問題としてとらえる。目的関数を献立に含まれる料理の調理時間と調理にかかるコストの最小化とし、制約条件を必要栄養素や摂取カロリーなどとして多目的最適化を行う。そして、二つの目的関数の最小化と、複数の制約条件に基づいた、パレート最適である献立を出力する。なお、献立の日数や料理の準備にかかる時間などのユーザの選好がかかわる制約条件はユーザーが選択できる。多目的最適化問題を解く手段として、NSGA-II という遺伝的アルゴリズムを多目的最適化に応用した手法を用いる。なお、このシステムは pymoo という Python プログラム用いて、NSGA-II によって組み合わせ最適化問題を解かせるようにプログラムの記述を行う。NSGA-II によって出力したパレート解である献立は、摂取栄養量やカロリー、主菜と副菜の数、アレルギー制限などの制約条件をみたし、調理にかかるコストおよび調理にかかる時間が最小化された、パレート最適な集合として複数出力される。この出力された複数の献立のうちユーザに最もあった献立をユーザ自身に選択してもらう。今回用いるライブラリの1つである pymoo について説明する。

#### pymoo

pymoo とは、単目的最適化や多目的最適化などの最適化アルゴリズムを解くために使われる Python ライブラリの一つである。pymoo は GA や、粒子群最適化、NSGA-II、Nelder-Mead 法などの最適化手法を、ライブラリからインポートすることによって簡単に使うことができる。pymoo が扱える最適化手法の一部の例を表 4.1 に示す。また、多目的最適化問題である ZDT2 から DTL6 などの数十個のベンチマーク関数が数多く実装されており、自分で関数を自作し、その問題に制約条件などを設けることによりその問題に対して最適化を行うこともできる。さらに、関数をカスタマイズすることによって、ZDT5 などのバイナリ決定変数の問題や、混合変数問題などの最適化問題も解くことが可能となる。また、多目的最適化による結果は、さまざまな視覚化手法を利用することにより出力できる。pymoo で扱える視覚化の手法の一部を図 4.1 に示す。各手法は、それぞれ異なる目的を持っており、低次元もしくは高次元で表現される目的関数空間にも適応することができる。具体的な手法を挙げると、散布図や平行座標プロット、ヒートマップやレーダープロットなどが実装されている [25]。

散布図は、2つの要素からなる1組のデータが得られたときに、2つの要素の関係を見るために作られたプロットしたグラフで、1つ目の要素が変化したときに、2つ目の要素はど

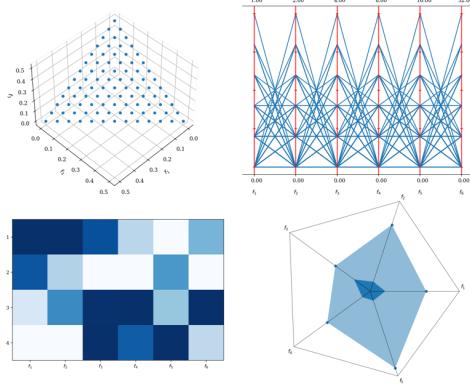


図 4.1: pymoo における様々な視覚化手法

表 4.1: pymoo の実装アルゴリズムの例

アルゴリズム	目的関数
GA	単目的
DE	単目的
BRKGA	単目的
Nelder-Mead	単目的
MOEAD	多目的
NSGA-II	多目的
CTAEA	多目的

のように変化するかを確認することができるものである。2つの要素の間に何らかの関係がある場合、これらのデータ間には「相関関係」があるといえる。

3変数以上の関係を把握することが困難であり、また各散布図の間で、点がどのような対応関係にあるのかは分からことが多い。高次元データを、直交座標に映して把握すれば可能だが人間は3次元以上の空間は認識できないため、不可能である。それを何とかして可能にしようと考えられたものが、平行座標プロットである。平行座標プロットとは、各変数軸(座標軸)を平行に並べたものである。これは、直交に並べることが不可能な場合に用いられる。これによって、ある程度の多変数間の繋がりを視覚化することが可能となる。使われている例として、医学研究者は、異なるそれぞれの薬物が様々な種類の細菌の増殖に与える影響を評価するため、平行座標プロットを作成して評価するということが挙げられる。

さらに、pymoo は並列処理、分散処理にも対応している。pymoo では、1台のPC内で各スレッドに処理をどう割り当てるかなどの操作を行ったり、複数台のPCをクラスタとして用意して、処理の管理をするスケジューラとするPCから実際に処理をするワーカーとするPCに最適化処理に関するタスクを分散し処理することが可能となっている。

## 変数

各変数は、対象の日数を  $D$ 、日の番号を  $k$ 、レシピの数を  $R$ 、料理レシピが献立に含まれている場合に1、含まれていない場合に0の値をとる献立フラグを  $r_{ki}$ 、料理レシピが主菜の場合に1、副菜の場合に0の値をとる主菜フラグを  $\sigma_i$ 、 $i$ 番目の料理レシピの調理時間を  $T_i$ 、 $i$ 番目の料理レシピの食材コストを  $G_i$ 、 $i$ 番目の料理レシピの  $l$ 番目の摂取栄養素を  $f_{il}$ 、 $l$ 番目の栄養素の制約の最大値を  $F_l^H$ 、最小値を  $F_l^L$ 、 $i$ 番目の料理レシピの摂取カロリーを  $C_i$ 、基礎代謝量の制約の最大値を  $B^H$ 、最小値を  $B^L$ 、朝食、昼食、夕食における最大調理時間をそれぞれ  $\tau_1, \tau_2, \tau_3$  とする。

また、入力画面でアレルギーが選択された場合に1、選択されていない場合に0の値をとるアレルギーフラグを  $x_i$ 、各制限食が選択されていた場合に1、選択されていない場合に0をとる制限食フラグを  $y_i$ 、制限食における栄養素の制約の最大値を  $E_l^H$ 、最小値を  $E_l^L$  とする。

本研究で提案する、自動献立作成システムにおける多目的最適化問題の目的関数と制約条件は、上記の変数を用いて下の式によって定式化される。

<定式化>

$$\text{minimize} \quad \sum_{k=1}^{3D} \sum_{i=1}^R r_{ki} T_i \quad (4.1)$$

$$\text{minimize} \quad \sum_{k=1}^{3D} \sum_{i=1}^R r_{ki} G_i \quad (4.2)$$

$$\text{subject to} \quad F_l^L \leq \sum_i^R r_{ki} f_{il} \leq F_l^H \quad (\forall k, \forall l) \quad (4.3)$$

$$B^L \leq \sum_i^R r_{ki} C_i \leq B^H \quad (\forall k) \quad (4.4)$$

$$\sum_i^R r_{ki} T_i \leq \tau_1 \quad (k \% 3 = 1) \quad (4.5)$$

$$\sum_i^R r_{ki} T_i \leq \tau_2 \quad (k \% 3 = 2) \quad (4.6)$$

$$\sum_i^R r_{ki} T_i \leq \tau_3 \quad (k \% 3 = 3) \quad (4.7)$$

$$0 < \sum_i^R r_{ki} \sigma_i \leq 1 \quad (\forall k) \quad (4.8)$$

$$0 \leq \sum_i^R r_{ki} (1 - \sigma_i) \leq 3 \quad (\forall k) \quad (4.9)$$

$$\sum_{k=1}^{3D} r_{ki} \leq 1 \quad (4.10)$$

$$0 \leq \sum_i^R r_{ki} x_i < 1 \quad (\forall k) \quad (4.11)$$

$$E_l^L \leq \sum_i^R y_i r_{ki} f_{il} \leq E_l^H \quad (\forall k, \forall l) \quad (4.12)$$

## 目的関数

本研究の献立作成における多目的最適化問題を構成する目的関数と制約条件式について説明する。まず、目的関数は、式(4.1)と式(4.2)の2つであり、(4.1)は調理時間の最小化であり、(4.2)は食材コストの最小化である。0-1変数である献立フラグを用いて、設定した日数での料理の組み合わせを表現する。

## 制約条件

制約条件は、式(4.3)から式(4.12)の10つである。式(4.3)は摂取栄養量制約、式(4.4)は摂取カロリー量制約、式(4.5)から式(4.7)は朝食、昼食、夕食における最大の調理時間制約、式(4.8)と式(4.9)は主菜は1つ、副菜は3つ以下で献立を構成する制約、式(4.10)は献立の中に、同じ料理が存在しないようにする制約である。また、式(4.11)は、入力画面でアレルギーを選択した時に、そのアレルギーが含まれるレシピが含まれないようにする制約であり、式(4.12)は、入力画面で制限食が選択されたときの摂取栄養素量の制約である。

摂取栄養量制約は、1日あたりに摂取する特定の栄養量に、下限と上限を設定して表現する。摂取カロリー量制約は、1日あたりに摂取するカロリー量に、下限と上限を設定して表現する。朝食、昼食、夕食における最大の調理時間制約は、入力画面で入力した朝、昼、夜の各時間帯における献立にかかる調理時間をそれぞれ上限に設定した。

主菜は1つ、副菜は3つ以下で献立を構成する制約は、その料理が主菜であるか、副菜であるかを表現する0-1変数の主菜フラグを用いて、献立に含まれる主菜と副菜の下限と上限を表現する。

## § 4.2 選定されたデータに基づく DEA 分析

対話型とは、利用者とシステムがディスプレイなどの出力装置、キーボードやマウスなどの入力装置を介して会話をするように互いに指示や応答をしながら作業を行う処理方式のことである。対話型で処理を行うソフトウェアやシステムの具体例としては、ユーザが次に選択したいものをディスプレイ上の音声や画像、動画などの形で提示することや、操作するユーザの意図を汲み取り、それに対して反応を返したりすることなどが挙げられる。

また、他のシステムの利用形態としてバッチ処理、リアルタイム処理がある。バッチ処理とリアルタイム処理の流れを図??に示す。バッチ処理とはプログラム（データ）を処理目的ごとにまとめ、そのデータを順次処理していく一連の流れ、システムを指す。バッチ処理は特定の処理にまとめて実行するので、大量のデータを一括に計算することに向いている。しかし一定のデータがたまつたときにそのデータをまとめて処理をするため、リアルタイム性が求められる機能にはバッチ処理は向いていない。バッチ処理が向いている処理の事例として、事務所における給与計算、振込・請求処理などがある。

リアルタイム処理とは、端末から入力されたデータ、発生したデータを、処理要求が発生した時点から即時にコンピュータで処理する処理方法である。リアルタイム処理はデータの遅延が極めて少なく、処理に使われる情報が最新なものであるため、受け取ったデータの情報をすぐ知ることができ、問題も特定しやすいという利点がある。しかし、受け取ったデータをきわめて短い時間で処理をするため、処理をするハードウェアが高性能になり、単純なシステムで実装することが難しくなる。バッチ処理が向いている処理の事例として、GPS追跡アプリケーションや、株式のリアルタイム売買などがある。

バッチ処理、リアルタイム処理はどちらも大量のデータを形式的な処理方法で処理をする事例には向いているが、ユーザの選好を反映した処理は難しい。対して処理対話型処理は、ユーザにとって最も適した解を出力したい場合によく使われている。そのため、本研究においては対話型処理によるシステムを開発している。

また、対話型の処理のシステムは、機能が増え、複雑化したシステムを専門的な知識を持っていない人でも利用できるようにすることを目的としている。そのため、表示の分か

りやすさ、見やすさ、感覚的な操作が可能であることなどが重要な要素として挙げられる。そのため、対話型処理を行うソフトウェアやシステムには、GUIが用いられることが多い。対話型処理を行うソフトウェアやシステムのイメージを図??に示す。GUIは、Graphical User Interfaceの略であり、コンピュータの画面上に表示されるウィンドウ、アイコンやボタン、ドロップダウンメニューなどを用いることで、マウスやタッチパネルなどのポインティングデバイスで感覚的な操作を可能とするインターフェースである。これと反対に、文字でコマンドを入力して操作するインターフェースは、Character User Interfaceの略でCUIと呼ばれている。また、現在のPCのインターフェースにはほとんど全てにGUIが採用されている。

次に、多目的最適化で解いたパレート解のうち、ユーザにどのようにして最適な献立を出力させるかについて説明する。複数の目的関数の最小化または最大化を考える多目的最適化において、複数の目的関数を同時に満たすような解は存在せず、一方の目的関数が高い評価を得た場合、他方の目的関数は犠牲となってしまうトレードオフの関係になってしまることが普通なため、目的関数が複数にある場合においての解は、意思決定者にとって、最も好ましいものを選択できるようにすることが大事である。

対話型処理を用いたパレート最適解を選好する従来事例として、多目的最適化問題に関して、制約式と目的関数に含まれるパラメータの決定などの問題の設定時に含まれるあいまい性と、意思決定者があいまいな目標を持つことを考慮した、対話型ファジィ満足化手法がある。この手法では、個体の作成から最適化処理の部分はアルゴリズムが担い、その最適化処理の過程における評価の部分行っている。このシステムでは、ランダムで生成された個体をユーザに画像で提示し、提示された画像に対してユーザが5段階評価をし、その評価に従って近似最適解を再度作成している[26]。

また、单一目的の大規模な多目的離散最適化問題を、効率的に解を探索するためのアルゴリズムである、モジュラーアプローチを用いて解き、それによって求められたパレート最適解集合の大きさを表示したのち、パレート最適解集合の大きさが決められた値以内になるまで繰り返しモジュラーアプローチを用いて解き、縮小されたパレート最適解集合の中から各目的関数の重要度などの自分の選好条件に基づいて選好最適解を決定する手法が挙げられる。

他には、対話型GAによる近似最適解の探索を基本としつつも、GAの最適化処理の過程における、個体の適応度を評価をする部分を人間が行うといった手法も提案されている。一般的なGAでの評価の役割は、評価関数が担っているが、対話型GAでは、この評価関数により個体の適応度を決定する部分を、人間が評価を行うようにしている。人間の意思決定を個体の適応度評価の過程に組み込むことにより、人間による主観的な評価が1つのシステムの要素となることから、対話型GAは人の感性をシステムに落とし込むことが可能な手法である。

対話型GAは、感覚や個人の好みなどといった、数値では表すことが困難な個人の感性を、対話型GAによる設計やデザインに取り入れることが可能となっているため、服飾やオフィスデザインや感性による様々な事柄への推薦、補聴器を使用する人の、聞こえに合わせるフィッティングなどへの研究に応用することが可能となっている。

意思決定者の選好解を求めるために、大きく分けて、以下の3つのアプローチがある。

1. 全て、もしくは十分に多くパレート最適解を求め、それを意思決定者に提示し、選好

解を自分で決定してもらう.

2. 意思決定者の選好を表す実数値関数である, 値値関数または効用関数を求め, それを最適化するような数理計画問題を解く.
3. コンピュータによって導出されたパレート最適解と, その解に基づく意思決定者の局所的な選好情報を用いて, ユーザとコンピュータの対話を繰り返すことによって, 選好解を決定する.

最初のアプローチでは, 目的関数の数が少ない場合や, 実行可能解が少数で, 有限個しか存在しない場合に有効であるとされる. この方法で代表的なものとして, 各目的関数に対する重みを用いて, 問題を解く加重和最小化や, 1つの目的関数を残し, 他の目的関数に対する要求水準を制約条件に用いる, 制約変換法などがある.

2番目のアプローチの, 値値関数もしくは効用関数の同定について, 多属性効用理論が知られており, 目的関数間の独立性が十分確保されていることが重要となる [?]. 1番目のアプローチで挙げた, 加重和目的関数を, 値値関数もしくは効用関数として想定して, そのパラメータを同定するといった, このアプローチの簡略版も考えられる.

最後のアプローチは, 対話型解法と呼ばれており, 意思決定者が, システムとの対話をすることによって, 複数ある目的関数をどのように選り好みするかといった, 局所的な選好情報を用いて, パレート最適解から解を自動的に選択する, という方法である.

この方法は, コンピュータとユーザの両者の情報交換の仕方によるので, いくつもの方法が考えられるが, ユーザという人間が関わっているということから, ヒューマンフレンドリーである方法が望まれる. 提案してきた対話型解法として, 意思決定者が目的関数に対する, 望ましいと考える値である希求水準を設定して, それに最も近い解をパレート最適解から得るという, 希求水準法などが挙げられる [?].

本研究では, 対話型処理によって, ユーザに対して分かりやすく献立を, 選択してもらいたいと考えたため, 3番目のアプローチをとる. 多目的最適化によって調理時間と材料コストを最小化するように得られた, 入力した日数分の料理レシピをそれぞれ表すパレート最適集合の中から, ユーザがコンピュータとの対話型処理によって献立を選択する.

### § 4.3 提案手法の流れ

本研究で提案する制限食と多人数考慮した自動献立作成システムの流れを図??に示す. また, 本システムの流れを説明する.

#### Step 1: 料理レシピ, 食材価格のデータベースの作成

はじめに, Python を使って Web サイトからスクレイピングし, 料理レシピと食材価格のデータベースを作成する. 本研究で用いるスクレイピング手法として, Python のライブラリである BeautifulSoup4 を扱う. BeautifulSoup4 は Web サイト上の HTML から, 取得したいデータを HTML 内のクラスや ID などの要素検索して抽出することができる. また, 最初に対象の Web ページから HTML を取得する必要があるため, その際に HTML パーサーである Requests を用いる. これらのスクレイピング手法を用いて, レシピ情報を取得する. 料理レシピサイトとして, 「ボブとアンジー」のみを参考にしている研究も存在して

いるが [?], 1つのレシピサイトでは出力されるレシピに偏りがあるという問題点があったため, 本研究においては「ボブとアンジー」, 「EatSmart」, 「おいしい健康」の3つのサイトからスクレイピングした情報をレシピサイトとしている. また, 料理情報とその食材の価格, 販売価格の情報は. 食材とその価格動向を載せているサイト, 「小売物価統計調査による価格推移」からスクレイピングしている.

それぞれの料理レシピサイトからは, その料理から摂取することができる全栄養素やカロリー, 調理時間, 必要な材料名, 材料量, 料理のイメージ, アレルギー情報, 作り方などのデータがスクレイピング可能である. それらの各料理レシピデータはそれぞれ CSV ファイルに出力され保存されるようになっている. また, それぞれのレシピサイトから取得できる栄養素の単位などが異なる場合があるので, すべてのレシピサイトから取得できる栄養素の単位, 順番を統一している.

また, 食材価格データは全て1つの CSV ファイルに出力されるようになっている. 食材から得られる栄養素は, 食材価格データベースに付随して出力される. その後, 料理レシピにて必要な食材名を, 食材価格データベースから一致するものを検索し, 見つかったときに, その必要な食材量を食材価格データベースの販売単位と価格から計算し, 食材コストを算出する.

その際, 料理レシピにおける食材名と食材価格データベースにおける食材名が微妙に違っていた時には見つからないため, Python のライブラリである `difflib` を用いて, ゲシュタルトパターンマッチングという手法で, 2つの食材名の類似度を計算し, 類似度の近い食材を検索して見つけるようにしている.

この手法を用いても食材価格データベースから見つけることができなかつたときは, ショッピングサイトである楽天市場から, カテゴリを食品に設定してその食材について検索をかけ, 販売単位と値段をスクレイピングし, スクレイピングした食材価格データは, データベースの CSV ファイルに追加している.

## Step 2: ユーザ情報と制約条件の入力

組み合わせ最適化を解くに当たって, 設定する制約条件がユーザーの身体情報によって異なる. そのため, 献立作成をする前にユーザーに, 入力情報を入力しておく必要がある. ユーザ情報を入力する手順としてまず, 献立作成に必要な人数分のユーザ情報を利用者に入力してもらう. その後入力する情報として入力する人数を入力すると人数分のユーザ情報を入力する画面が出てくる. その中でそれぞれのユーザの名前, 身長と体重, 年齢, 性別, アレルギー情報, 予防したいまたはすでに患っている生活習慣病を入力する. 入力された身体情報は身体情報データベースに蓄積されており, 最適化処理を実行するときに使用する.

その次に, 出力する献立の日数及び, 朝, 昼, 夜に調理の準備にかける時間を選択する. 入力した日数は多目的最適化を解く際の選ぶ献立数を設定している. 1日あたりに出力する献立は7品としているため入力した日数によって選択するレシピの数が異なる. また, 朝, 昼, 夜に調理の準備にかける時間は多目的最適化を解く際の朝, 昼, 夜にかける時間の最大値として設定される.

## Step 3: NSGA-II による多目的最適化と最適な献立の出力

次に, 料理レシピデータ群から入力された情報をもとに作成された制約条件と目的関数に沿った料理レシピを選択するという組み合わせ最適化問題と捉え献立作成を行う. 献立作

成に用いるデータまたは変数として、料理レシピサイトと食材とその価格を載せているサイトからスクレイピングしたレシピデータ、食材価格、栄養素データと、ユーザーによって入力された身長と体重、年齢、性別から計算された基礎代謝量、推定エネルギー必要量、アレルギー情報、疾患情報を用いる。これらの情報を多目的最適化問題を NSGA-II によって解き、パレート最適な献立を出力する。

NSGA-II は、NSGA をエリート保存選択、混雑距離の導入、高速ソートの 3 点について変更と改良を施した手法であり、多目的最適化問題を解くアルゴリズムの 1 つである。目的関数には調理時間の最小化、使用する材料のコストの最小化が与えられ、制約条件には、3 大栄養素の摂取量、摂取カロリー量、朝、昼、夕の時間帯別の調理時間合計、献立に含まれる主菜と副菜の数、アレルギーがある場合にそのアレルギーの材料が含まれないようにする、疾患を患っている場合、その疾患にあった栄養素の摂取量の制限、出力される日数のうち、料理が被らないようにする、などの条件が設定されている。

自動献立作成における多目的最適化問題を定式化し、それをプログラム上に記述する際には、Python のライブラリである pymoo を用いた。pymoo は PSO や GA、多目的進化アルゴリズムや NSGA-II などの、単目的最適化問題や多目的最適化問題を解くための様々な手法をサポートしている。目的関数や制約条件が視覚的にわかりやすく記述できることや、自作関数の作成や用意した変数を最適化処理に組み込むことが容易なこと、今までスクレイピングで扱ってきた Python のプログラムで実装ができるところから、本研究のシステムにて組み込むことにした。

#### Step 4: 対話型処理による献立の選択

自動献立作成システムにおける、多目的最適化によって得られたパレート最適な献立から、ユーザ自身の希望に叶った献立を選択できるような対話型処理を行う。具体的には、調理時間と食材コストの 2 つの目的関数が最小化されたパレート解と、最適解におけるそれぞれのコストと調理時間を候補として表示し、ユーザに提示させる。ユーザは候補番号を選択し、最適解を評価する。これらの対話型処理によってパレート最適な献立の中から選好された、入力した日数分の献立の情報は、HTML 上に表示するようになっている。

自動献立作成の最適化プログラムにて出力された、入力した日数分の献立を構成する要素である料理レシピデータの料理名、料理イメージ画像、調理時間、食材コスト、摂取栄養素量や摂取カロリー、調理時間、必要な食材とその量、作り方などのデータを HTML に渡し、献立データを HTML 上に表示する。HTML にはその日に作るべきレシピの名前が表示されており、また、それぞれの日にちのページにジャンプするリンクが設置されている。ジャンプ先では、その日にちについての献立を構成する要素である料理が朝、昼、夕に分けられて表示される。なお、HTML は、本サーバー上に配置されている。これらの方によって、ユーザに自動献立作成の最適化処理によって HTML 上に出力された、入力した日数の献立のデータを提示する。

## 数値実験並びに考察

### § 5.1 数値実験の概要

本研究の流れは料理レシピ、食材価格のデータベースの作成、ユーザ情報と制約条件の入力、NSGA-IIによる多目的最適化と最適な献立の出力、対話型処理による献立の選択となっている。まず、使用するレシピデータ数は3000個とした。Pythonによるスクレイピングを行う際は、PythonのライブラリであるurllibとBeautifulsoup4を使った。

使用したレシピサイトは「ボブとアンジー」、「EatSmart」、「おいしい健康」の3種類からスクレイピングする。urllibにより、目的のレシピサイトと食材価格サイトのWebページのURLを渡し、そのページのHTML情報を取得したのちに、Beautifulsoup4を用いてWebページ上の料理レシピ名や摂取栄養素、食材とその価格などの必要な要素を、class名やid名などで指定し取得する関数を用いてスクレイピングを行う。Webサイトからのスクレイピングによって作成した料理レシピデータベースの例を図??に示す。

3つのレシピサイトからはスクレイピングする情報として、その料理から摂取することができる全栄養素やカロリー、調理時間、必要な材料名、材料量、料理のイメージ、アレルギー情報、作り方などをスクレイピングする。また各料理レシピの食材コストについては、料理に必要な食材と、食材価格データベースの中の食材を照らし合わせ、必要食材量と食材の価格、販売単位を用いて計算する。次に、NSGA-IIによる多目的最適化をしている際の実行画面を図??に示す。これはプログラムの内部で行われている処理を可視化したものであり、本研究はブラウザでシステムを用いている。そのため本研究では表示されない。NSGA-IIを用いた多目的最適化プログラムは、Pythonのライブラリである、pymooを利用して記述した。pymooは、多目的最適化や単目的最適化などの様々な解法をサポートを可能とするライブラリである。

今回の実験で設定した目的関数と制約条件について説明する。目的関数は、調理時間の最小化と、食材コストの最小化を設定する。制約条件は、健常者の場合と制限食が必要な人の場合で異なる。まず、健常者の場合について説明する。摂取栄養素については、3大栄養素である、たんぱく質、脂質、炭水化物のそれぞれに、1日に最低でも摂取すべき量を摂取できるように設定した。設定した値は、それぞれの3大栄養素に対して、たんぱく質は1日に必要な推定エネルギーの13%以上、脂質は15%以上、炭水化物は40%以上である。超えて摂取すると、健康障害のリスクが高まる定義される耐容上限量は、3大栄養素に関しては設定されていないため[27]、制約条件として上限値は設定しないことにした。

摂取カロリーについては、1日に必要なエネルギー量の目安を掲載している農林水産省のサイト[28]を参考にして、基礎代謝量と身体活動レベルの係数をかけ合わせたものを使用

した。そのため、上限値は 2536 キロカロリーに設定した。

次に、制限食が必要な人の制約条件について説明する。本研究で対象となる制限食が必要な人は、アレルギーを持っている人と、生活習慣病を患っている人である。また、対象となる生活習慣病は糖尿病、腎臓病、脂質異常症、高血圧とする。

まず、糖尿病を患っている人についてだが、4.2 章で述べた通り、糖尿病は、内臓脂肪型肥満によってインスリン抵抗性により発症する。そのため糖尿病の予防と改善には脂肪の是正が重要となってくる。また、厚生労働省によると、1 日あたりの炭水化物摂取量を 100 g 以下とする炭水化物制限が、肥満の是正に有効だとし、糖尿病の予防に有効だとしている。また、食物繊維の 1 日の平均摂取量が 20g を超えた時点から糖尿病の発症リスクに有意な低下傾向が見られている [19]。そのため、本研究における糖尿病の患者に対する制約条件として、エネルギー量、タンパク質摂取量は健常者と同じだが、1 日の炭水化物摂取量、脂質の摂取量、食物繊維の摂取量をそれぞれ 100g 以下、必要推定エネルギーの 15~25%，20g 以上とする。

次に、腎臓病を患っている人については、4.2 章で述べた通り腎臓病はたんぱく質制限、塩分制限、カリウム制限などの食事療法を行うことにより、腎機能障害の進行を抑え、慢性腎臓病の合併症を予防することができる。具体的な数値として日本腎臓学会によると 1 日のタンパク質の摂取量を標準体重当たり 0.6~0.7g とし、塩分の 1 日の摂取量は 3g 以上 6g 未満とし、カリウムの 1 日の摂取量が 1500mg 以下に制限することが推奨されている [20]。そのため、本研究における腎臓病の患者に対する制約条件として、エネルギー量、脂質、炭水化物の摂取量は健常者と同じだが、1 日のタンパク質と塩分と、カリウムの摂取量をそれぞれ標準体重当たり 0.6~0.7g、3g 以上 6g 未満、1500mg 以下とする。

脂質異常症を患っている人については、4.2 章で述べた通り脂質異常症は、コレステロール、食物繊維、脂質の摂取量を調整することにより脂質異常症の予防と改善に役に立つとされている [22]。本研究における具体的な制約条件としては、エネルギー量、炭水化物、タンパク質の摂取量は健常者と同じだが、1 日のコレステロール、脂質、食物繊維の摂取量をそれぞれ 200mg 以下、総エネルギーの 15% 未満、20g 以上にすることとする。

高血圧と診断されている人は、4.2 章で述べた通り高血圧の要因として、塩分を過剰に摂取することによる血圧上昇が大きな要因となるため、塩分制限が必要となっている。日本高血圧学会による「高血圧治療ガイドライン 2019」によると、高血圧者の減塩目標を食塩 6 g/日 未満としている。また、カリウム摂取量増加によって高血圧者にとって血圧低下効果を認めた。厚生労働省によると、カリウムの摂取量を 3510mg 以上摂取することが推奨されている。さらに、1 日の食物繊維の摂取量を 20g 以上にすること推奨されている [23]。以上のことから、本研究における高血圧者に対する制約条件として、エネルギー量、脂質、炭水化物、タンパク質の摂取量は健常者と同じだが、1 日の塩分、カリウム、食物繊維の摂取量をそれぞれ 6g 未満、3510mg 以上、20g 以上とする。

次に、NSGA-II による最適化を行っている際の、実行画面について説明すると、*n\_gen* は現在の世代数、*n\_level* はこれまでの個体を評価した数、*cv (min)*、*cv (avg)* はそれぞれ現在の母集団における最小の制約違反、現在の母集団における平均の制約違反、*n\_nds* は多目的最適化問題の場合の非劣解の数、*eps* は過去数世代にわたるインジケーターの変化、*indicator* はパフォーマンスインジケーターを表す。

次に、NSGA-II による最適化処理が終わり、パレート最適解が出力された様子を図??に

示す。縦軸は、指定した日数分の献立の合計調理時間を表しており、横軸は指定したに数の合計の食材コストを表している。

次に、パレート最適解から、対話型処理によって献立を選択する画面を図??に示す。図??は、図??のパレート解を数値として表示している画面である。ユーザは、画面に表示されている選択ボタンで、表示されている候補を選択する。「献立を表示する」というボタンをクリックすると選択した候補に対応した献立が出力される。

本研究においては以下の数値実験を行う。まず、健常者のユーザー像を想定し、システムを動かし、出力された献立が制約条件を満たしているかを考察する。次に、それぞれの生活習慣病を患っているユーザー像を想定し、同様の検証をする。さらに、複数人のユーザー像を想定し入力した全員の制約条件を満たしているか考察する。

## § 5.2 実験結果と考察

今回数値実験をするにあたって、朝、昼、夜の合計の調理時間の合計と主菜、副菜の数を共通にしておく。具体的な数値として、朝、昼、夜の合計の調理時間の合計をそれぞれ15分、45分、60分とし、主菜、副菜の数を1つ、2つとした。また、以下は今回の数値実験に使用する1日に必要なエネルギーである、必要推定エネルギー量の計算式である。

< 基礎代謝基準値 >

$$\text{基礎代謝基準値} = \frac{\text{基準体重での基礎代謝量 (kcal/日)}}{\text{基準体重 (kg)}} \quad (5.1)$$

< 基礎代謝量 >

$$\text{基礎代謝量 (kcal/日)} = \text{基礎代謝基準値} \times \text{体重 (kg)} \quad (5.2)$$

< 必要推定エネルギー量 >

$$\text{必要推定エネルギー量 (kcal/日)} = \text{基礎代謝量} \times \text{身体活動レベル指数} \quad (5.3)$$

また、栄養素の制約条件に関しては、3大栄養素であるたんぱく質、脂質、炭水化物について設定した。具体的に設定した制約値としては、厚生労働省によると健常者におけるたんぱく質、脂質、炭水化物の必要摂取エネルギー量はそれぞれ必要推定エネルギーの40%以上、15%以上、13%以上とされているため[27]、そのように設定した。以下は、最低でも1日に摂取すべき3大栄養素の量を計算する式である。

< 必要たんぱく質 >

$$\text{たんぱく質 (g/日)} = \frac{\text{必要推定エネルギー量 (kcal/日)} \times 0.13}{4(\text{kcal/g})} \quad (5.4)$$

< 必要脂質 >

$$\text{脂質 (g/日)} = \frac{\text{必要推定エネルギー量 (kcal/日)} \times 0.15}{9(\text{kcal/g})} \quad (5.5)$$

< 必要炭水化物 >

$$\text{炭水化物 (g/日)} = \frac{\text{必要推定エネルギー量 (kcal/日)} \times 0.4}{4(\text{kcal/g})} \quad (5.6)$$

今回はこれらの数式をもとに複数のユーザー像を想定して数値実験を行う。

### 1: 健常者に対する出力結果の考察

今回想定したユーザー像は以下のとおりである。まず、健常者のユーザーとして22歳の平均男性の平均値である、身長を172.3cm、体重を65.3kg、活動レベルを普通とした。

式(5.1)～(5.3)から1日に必要推定エネルギー量を計算すると、2695kcalとなるため、1日に摂取するカロリーは2695kcalプラスマイナス100kcalの範囲に入るように制約が設定されるようにした。また、(5.4)～(5.6)から、1日に最低でも摂取すべきたんぱく質は、84.33g以上の値であり、1日に最低でも摂取すべき脂質は、摂取エネルギーの2595kcalの15%以上であるから、1kcalに対して9g摂取できるとしたときに43.25g、炭水化物は摂取エネルギーの2595kcalの40%以上より、1kcalに対して4g摂取できるとしたときに259.5gである。各3大栄養素の上限値に対して、栄養素を摂取するための指標の1つであり、健康障害をもたらすリスクは医学的ないとみなされ、摂取量の上限を与える量と定義される耐容上限量は、それぞれ厚生労働省によって設定がされていないため、最低でも1日に摂取すべきである栄養素量を下限に設定している。今回設定した、摂取カロリーとたんぱく質、脂質、炭水化物の各3大栄養素、朝、昼、夜での各時間帯の調理時間合計、主菜と副菜による制約条件と、実験にて設定した値について、表??に示す。

本研究で提案する自動献立作成システムにおける、献立を作成した出力結果について図献立結果に示す。最適化処理によって出力された献立は、flaskにより作成したWebサーバー上で、HTMLファイルによって表示される。

また、以上の制約条件を入力したときの献立の出力結果およびパラメータはそれぞれ表??、表??に示す。

次に、自動献立作成システムによって実際に出力した献立が設定した制約条件を満たしているか比較を行う。最初に、摂取エネルギーの比較を行うと、出力された献立の1日に摂取エネルギーは2621kcalであり、これは制約条件である、2595kcal以上2795kcal以下を満たしている。

次に、3大栄養素の制約条件の比較を行う。出力された献立から得られる1日のたんぱく質、脂質、炭水化物はそれぞれ92.3g、72.1g、276.3gであり、これは1日に摂取すべきであるたんぱく質、脂質、炭水化物量である84.33g、43.25g、259.5gを満たしていることがわかる。

また、各時間帯別での、1日の献立の調理時間合計についての制約条件について比較を行う。出力された献立の朝、昼、夕の調理時間の合計はそれぞれ15分、40分、45分となった。これは制約した朝、昼、夕の調理時間の合計の15分以下、45分以下、60分以下を満たしている。

### 2: 生活習慣病を患っている人に対する出力結果の考察

まず、糖尿病を患っている人の数値実験について説明する。厚生労働省によると、BMIが23以上になると糖尿病のリスクがなりやすいと報告されている。また2009年の糖尿病の平均

年齢が 71 歳であることから、年齢を 71 歳、身長は 71 歳の男性の平均身長である 163.1cm、体重は BMI が 23 で身長が 163.1cm の場合の 61.18kg、身体活動レベルを低いとする。

また、糖尿病を患っている人に対する必要推定エネルギーは健常者と同じ式を使う。以上の身体情報とし、式 (5.1)～(5.3) から 1 日に必要推定エネルギー量を計算すると 1919kcal となるから 1 日に摂取するカロリーは 1919kcal プラスマイナス 100kcal の範囲に入るよう に制約が設定されるようにした。また、5.1 章よりたんぱく質の式は (5.4) を使い、炭水化物の摂取量を 100g 以下、脂質の摂取量を必要推定エネルギーの 15～25%、食物纖維の摂取量を 20g 以上とする。その結果、タンパク質の摂取量は 62.36g、脂質の摂取量は 31.98g～53.3g となった。

次に、腎臓病を患っている人の数値実験に関して説明する。糖尿病と同様、厚生労働省によると、BMI が 23 以上になると腎臓病のリスクがなりやすいと報告されている。また 2019 年の腎臓病の平均年齢が 70 歳であることから、年齢を 70 歳、身長は 71 歳の男性の平均身長である 163.1cm、体重は BMI が 23 で身長が 163.1cm の場合の 61.18kg、身体活動レベルを低いとする。

腎臓病を患っている人に対する必要推定エネルギーは健常者と同じ式を使う。以上の身体情報とし、式 (5.1)～(5.3) から 1 日に必要推定エネルギー量を計算すると 1919kcal となる。よって 1 日に摂取するカロリーは 1919kcal プラスマイナス 100kcal の範囲に入るよう に制約が設定されるようにした。脂質、炭水化物の制約条件は式 (5.5)、(5.6) から 30.31g、181.9g となった。タンパク質の制約条件は、腎臓病の場合標準体重当たり 0.6～0.7g となっ ているため 61.18kg の場合 36.7～42.82g となる。また、塩分の摂取量の制約は 3～6g、カリウムの摂取量は 1500mg 未満である。

続いて、脂質異常症を患っている人の数値実験に関して説明する。厚生労働省によると、BMI が 35 以上になると脂質異常症を発症するリスクが高まるとしている。また 2019 年の脂質異常症の平均年齢が 45 歳であることから、年齢を 45 歳、身長は 45 歳の男性の平均身長である 171.5cm、体重は BMI が 35 で身長が 171.5cm の場合の 102.94kg、身体活動レベルを低いとする。

また、厚生労働省総エネルギーを減らすことによる脂質異常症の抑制のを示す直接的なエビデンスはないとされている。よって必要推定エネルギーは式 (5.1)～(5.3) を使用して求める。以上の身体情報とし、式 (5.1)～(5.3) から 1 日に必要推定エネルギー量を計算すると 2838kcal となる。よって 1 日に摂取するカロリーは 2838kcal プラスマイナス 100kcal の範囲に入るよう に制約が設定されるようにした。また、たんぱく質、炭水化物の摂取量は式 (5.4)、(5.6) を使用してそれぞれ 92.23g、283.8g なった。また、コレステロールは 200mg 未満、脂質は総エネルギーの 15% 未満であるから 47.3g 未満、食物纖維の摂取量は 20g 以上である。

最後に、高血圧を患っている人の数値実験に関して説明する。厚生労働省によると、肥満の人が高血圧になりやすいとされている。そのため、肥満の平均値である BMI が 30 の人を想定する。また 2019 年の高血圧の平均年齢が 37 歳であることから、年齢を 37 歳、身長は 37 歳の男性の平均身長である 171.5cm、体重は BMI が 30 で身長が 171.5cm の場合の 88.24kg、身体活動レベルを低いとする。

また、高血圧を患っている人に対する必要推定エネルギーは健常者と同じ式を使う。以上の身体情報とし、式 (5.1)～(5.3) から 1 日に必要推定エネルギー量を計算すると 2624kcal

となるから1日に摂取するカロリーは2624kcal プラスマイナス100kcalの範囲に入るよう<sup>1</sup>に制約が設定されたようにした。また、たんぱく質、脂質、炭水化物の摂取量は式(5.4)、(5.5)、(5.6)を使用してそれぞれ82.03g、42.06g、252.4gとなつた。また、塩分の摂取量を6g未満、カリウムの摂取量を3510mg以上、食物纖維の摂取量を20g以上とする。

以上の制約条件から数値実験を行つた時の結果を以下の表??に示す。また、その時の制約条件を以下の表??に示す。

次に、自動献立作成システムによって実際に出力したそれぞれのパラメータが設定した制約条件を満たしているか比較を行う。最初に、摂取エネルギーの比較を行うと、出力された献立の1日に摂取エネルギーは糖尿病、腎臓病、脂質異常症、高血圧の順に行くと1926kcal、1830 k cal、2925kcal、2655kcalであり、これは制約条件の1日に摂取すべきカロリーである1819kcal以上2019kcal以下、1819kcal以上2019kcal以下、2738kcal以上2938kcal以下、2524kcal以上2724kcal以下の範囲内に入っているため制約条件を満たしている。

次に、それぞれの栄養素の制約条件の比較を行う。まず糖尿病患者の比較を行う。出力された献立から得られる1日のたんぱく質、脂質、炭水化物、塩分、食物纖維、カリウム、コレステロールはそれぞれで80.1g、42.3g、96.2g、8.1g、22.5g、3210mg、141mgであり、これは制約条件で設定した1日に摂取すべきたんぱく質62.36g以上、脂質31.98g以上53.3g以下、炭水化物100g以下、食物纖維20g以上を満たしていることがわかる。

次に腎臓病患者の比較を行う。出力された献立から得られる1日のたんぱく質、脂質、炭水化物、塩分、食物纖維、カリウム、コレステロールはそれぞれで35.7g、32.6g、194.5g、3.9g、18.3g、1465mg、72mgであり、これは1日に摂取すべきたんぱく質36.7g以上42.82g以下、脂質30.31g以上、炭水化物181.9g以上、塩分3g以上6g未満、カリウム1500mg未満を満たしていることがわかる。

続いて、脂質異常症患者の比較を行う。出力された献立から得られる1日のたんぱく質、脂質、炭水化物、塩分、食物纖維、カリウム、コレステロールはそれぞれで93.5g、40.5g、294.6g、8.2g、26.7g、2988mg、154mgであり、これは1日に摂取すべきたんぱく質92.23g以上、脂質47.3g未満、炭水化物283.8g以上、食物纖維20g以上、コレステロール200mg未満を満たしていることがわかる。

最後に高血圧患者の比較を行う。出力された献立から得られる1日のたんぱく質、脂質、炭水化物、塩分、食物纖維、カリウム、コレステロールはそれぞれで92.3g、43.5g、265.1g、4.5g、26.4g、3612mg、204mgであり、これは1日に摂取すべきたんぱく質82.03g以上、脂質42.06g以上、炭水化物252.4g以上、塩分6g未満、食物纖維20g以上、カリウム3510mg以上を満たしていることがわかる。

### 3: 大人数を想定した場合に対する出力結果の考察

次に、大人数を想定したときの考察する。今回実験した大人数として4人家族世帯を想定する。1人目のモデルとして年齢は52歳、身長と体重は52歳の平均身長、平均体重である170.8cm、70.4kgとし、性別は男、身体活動レベルは普通とした。2人目のモデルとして年齢は48歳、身長と体重は48歳の平均身長、平均体重である158.3cm、55.2kgとし、性別は女、身体活動レベルは普通とした。3人目のモデルとして年齢は22歳、身長と体重は22歳の平均身長、平均体重である172.6cm、64.0kg、性別は男、身体活動レベルは高いとした。4人目のモデルとして年齢は17歳、身長と体重は17歳の平均身長、平均体重である154.8cm、47.2kgとし、性別は女、身体活動レベルは低いとした。

上記の実験と同様に式(5.1)～(5.6)を用いてそれぞれのモデルの必要エネルギー、必要なたんぱく質、必要脂質、必要炭水化物をまとめたものを表??に示す。また、出力するにあたって出力日数を1日、朝の調理時間の合計を20分、昼の調理時間の合計を45分、夜の調理時間の合計を60分とした。また、本研究で提案する自動献立作成システムにおけるパラメータを表??に示す。

次に、自動献立作成システムによって実際に出力した結果が設定した制約条件を満たしているか比較を行う。最初に、摂取エネルギーの比較を行うと、モデル1、モデル2、モデル3、モデル4の出力された献立の1日に摂取エネルギーはそれぞれ2620kcal、2017kcal、3039kcal、1755kcalであり、これは制約条件の1日に摂取すべきカロリーである2509kcal以上2709kcal以下、1876kcal以上2076kcal以下、2953kcal以上3153kcal以下、1575kcal以上1775kcal以下の範囲内に入っているため制約条件を満たしている。

次に、それぞれの栄養素の制約条件の比較を行う。まずモデル1の比較を行う。出力された献立から得られる1日のたんぱく質、脂質、炭水化物はそれぞれ85.15g、43.67g、262gであり、これは制約条件で設定した1日に摂取すべきたんぱく質81.54g以上、脂質41.82g以上、炭水化物250.9g以上を満たしていることがわかる。

次にモデル2の比較を行う。出力された献立から得られる1日のたんぱく質、脂質、炭水化物はそれぞれ65.57g、33.63g、201.74gであり、これは制約条件で設定した1日に摂取すべきたんぱく質60.97g以上、脂質41.82g以上、炭水化物187.6g以上を満たしていることがわかる。

続いてモデル3の比較を行う。出力された献立から得られる1日のたんぱく質、脂質、炭水化物はそれぞれ98.77g、50.66g、303.92gであり、これは制約条件で設定した1日に摂取すべきたんぱく質95.97g以上、脂質49.22g以上、炭水化物295.3g以上を満たしていることがわかる。

最後にモデル4の比較を行う。出力された献立から得られる1日のたんぱく質、脂質、炭水化物はそれぞれ80.1g、42.3g、96.2gであり、これは制約条件で設定した1日に摂取すべきたんぱく質57.05g以上、脂質29.26g以上、炭水化物175.54g以上を満たしていることがわかる。

#### 4: 出力する日数を変更した場合に対する出力にかかった時間の比較

自動献立作成にあたって、出力する日数を変更した場合に対する出力にかかった時間の結果を表??に示す。表??より、1日分の献立を出力したときの出力時間の平均は946.73秒、2日分の献立を出力したときの出力時間の平均は942.257秒、3日分の献立を出力したときの出力時間の平均は956.817秒、4日分の献立を出力したときの出力時間の平均は1092.31秒、5日分の献立を出力したときの出力時間の平均は904.821秒、6日分の献立を出力したときの出力時間の平均は928.317秒、7日分の献立を出力したときの出力時間の平均は934.55秒であることが分かった。

この結果から、出力する日数を変更しても実行時間はあまり変化しないことがわかる。この結果より、出力する人数を変更は処理時間に大きく影響しないことがわかった。

## おわりに

急激な生活様式の欧米化に伴い、ジャンクフードといった、余分にエネルギーを摂取してしまうような食生活が大きく広まつたことから、現在、生活習慣病を患う人々が増加している。生活習慣病を予防する一つの方法として、栄養バランスのとれた食事をとることが推奨されている。しかし、栄養バランスの取れた献立作成には、その人の身体情報、疾患情報などによってメニューや料理の分量を調整しなければならなく、献立作成業務の負荷は高いことがわかる。

これらの問題を解決するために、本研究では、Web サイトから得られるレシピ情報や食材価格を活用し、制約条件を考慮できる多目的遺伝的アルゴリズムによって自動的に献立を作成をするシステムを考案した。

本研究で用いるレシピデータとして、3つのレシピサイトからスクレイピングを行うことによってレシピデータベースに多様性を持たせることができた。また、この献立作成システムは健常者だけではなく、生活習慣病を患っている人やアレルギーを患っている人でも利用できるようにした。さらに、プログラム実行に必要なすべてのプログラムをサーバーに置き、実行に必要な URL を用意することによって、ユーザはその URL をクリックするだけでプログラムを実行できるようにした。

また、プログラムの実行にはレシピデータなどの大量のデータが必要なため、プログラムの環境を整えるための手間が大変になってしまう問題があった。そのためプログラムをサーバー上に置くことでプログラム実行の環境を整える手間を省くことができた。

本研究で提案した制限食と大人数料理に対応した自動献立作成システムを実際に動作させた実験結果として、多目的最適化によって作成された献立は調理時間、料理コストを最小化しながら、設定した制約条件を満たしながら出力することができた。

本研究の課題として、摂取栄養素や摂取カロリーの上限、下限の設定などの制約条件を、ユーザ自身で決められるようにすることや、並列分散処理などを施すことにより、最適化プログラムの実行処理時間を向上し、よりユーザに快適に利用できるようにプログラムを改良する必要がある。また、ユーザが好みの料理を入力することによって、出力する料理がユーザの好みに近しいもの出るようにすることや、ユーザが現在持っている食材を入力することによって、その食材を含む料理が出力されるようにする必要があると考えられる。



## 謝辞

本研究を遂行するにあたり、多大なご指導と終始懇切丁寧なご鞭撻を賜った富山県立大学工学部電子・情報工学科情報基盤工学講座のAntónio Oliveira Nzinga René講師、奥原浩之教授に深甚な謝意を表します。最後になりましたが、多大な協力をしていただいた研究室の同輩諸氏に感謝致します。

2023年2月

水上和秀



## 参考文献

- [1] 公益社団法人 千葉県栄養士会, “生活習慣病の予防、食生活 生活習慣病の予防と食事”, <https://www.eiyou-chiba.or.jp/commons/shokujikou/preventive/seikatusyukan/>, 閲覧日 2023.1.7.
- [2] 国立研究開発法人 国立循環器病研究センター, “食事療法について”, <https://www.ncvc.go.jp/hospital/pub/knowledge/diet/diet02/>, 閲覧日 2023.1.7
- [3] ソフトム株式会社, “ソフトム通信 第 79 号「給食業界における A I 活用」”, [https://data.nifcloud.com/blog/food-service-provider\\_ai-use-case\\_01/](https://data.nifcloud.com/blog/food-service-provider_ai-use-case_01/), 閲覧日 2022.12.28.
- [4] 貝沼やす子, 江間章子, “日常の献立作りの実態に関する調査研究（第 1 報）”, 日本調理学会誌, Vol.30, No. 4, pp. 364-371, 1997.
- [5] 政府統計の総合窓口, “e-Stat”, <https://www.e-stat.go.jp/>, 閲覧日 2025.3.20.
- [6] 総務省統計局, “RESAS 地域経済分析システム”, <https://resas.go.jp/>, 閲覧日 2025.4.3.
- [7] 富山県, “富山県オープンデータポータルサイト”, <https://opendata.pref.toyama.jp/dataset/https-opendata-pref-toyama-jp-dataset-settou2023>, 閲覧日 2025.4.3.
- [8] 国土交通省, “不動産情報ライブラリ”, <https://www.reinfolib.mlit.go.jp/>, 閲覧日 2025.4.5.
- [9] C. A. Coello Coello and M. S. Lechuga, “MOPSO: a proposal for multiple objective particle swarm optimization, ” *Proceedings of the 2002 Congress on Evolutionary Computation (CEC'02)*, Vol. 2, pp. 1051-1056, 2002.
- [10] Q. Zhang and H. Li, “MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition”, *IEEE Trans. Evolutionary Computation*, Vol. 11, No. 6, pp. 712–731, 2007.
- [11] LeftLetter, “多目的進化型アルゴリズム MOEA/D とその改良手法”, <https://qiita.com/LeftLetter/items/a10d5c7e133cc0a679fa>, 閲覧日 2023.1.6.
- [12] C. A. Coello Coello and M. S. Lechuga, “統計的因果探索による社会基盤整備のストック効果の検証, ” *Proceedings of the 2002 Congress on Evolutionary Computation (CEC'02)*, Vol. 75, No. 6, pp. 583-589, 2020.
- [13] J. H. Holland, “DEA を用いた商圈属性に適合したホームセンターの部門別陳列棚数構成方法”, *Communications of the Operations Research Society of Japan*, Vol. 62, No. 10, pp. 677-684, 2017.

- [14] J. H. Holland, “統計的因果探索アルゴリズム“LiNGAM”を用いた若手研究者支援政策に関する研究”, *Japan Advanced Institute of Science and Technology*, Vol. 36, pp. 758-763, 2021.
- [15] ときわ会栄養指導課, “減塩について”, 栄養指導, <http://www.tokiwa.or.jp/nutrition/diet/low-salt.html>, 閲覧日 2023.01.15
- [16] 全国健康保険協会, “ちょっとした工夫で脂質をコントロール”, <https://www.kyoukaikenpo.or.jp/g4/cat450/sb4501/p004/>, 閲覧日 2023.01.15
- [17] 厚生労働省, “日本人の食事摂取基準(2020年度版)”, <https://www.mhlw.go.jp/content/10904750/000586559.pdf>, 閲覧日 2023.01.15
- [18] 東京医科大学病院, “カリウムは調理のくふうで減らせます”, 内臓内科, <https://articles.oishi-kenko.com/syokujinokihon/dialysis/05/>, 閲覧日 2023.01.15
- [19] 厚生労働省, “糖尿病”, <https://www.mhlw.go.jp/content/10904750/000586592.pdf>, 閲覧日 2023.01.17
- [20] 厚生労働省, “慢性腎臓病”, <https://www.mhlw.go.jp/content/10904750/000586595.pdf>, 閲覧日 2023.01.17
- [21] 腎臓内科, “慢性腎臓病の食事療法”, 東京女子医科大学, <https://www.twmu.ac.jp/NEP/shokujiryouhou.html>, 閲覧日 2023.01.17
- [22] 厚生労働省, “脂質異常症”, <https://www.mhlw.go.jp/content/10904750/000586590.pdf>, 閲覧日 2023.01.17
- [23] 厚生労働省, “高血圧”, <https://www.mhlw.go.jp/content/10904750/000586583.pdf>, 閲覧日 2023.01.17
- [24] 厚生労働省, “食べ物アレルギー”, アレルギーポータル, <https://allergyportal.jp/knowledge/food/>, 閲覧日 2023.01.17
- [25] J. Blank, “pymoo: Multi-objective Optimization in Python”, <https://www.egr.msu.edu/kdeb/papers/c2020001.pdf>, 閲覧日 2023.1.22.
- [26] 和正敏, “多目的線形計画問題に対する対話型ファジィ意思決定手法とその応用”, 電子情報通信学会論文誌 Vol. J 65-A, No. 11, pp. 1182-1189, 1982.
- [27] 厚生労働省, “日本人の食事摂取基準(2020年版)”, <https://www.mhlw.go.jp/content/10904750/000586553.pdf>, 閲覧日 2022.12.26.
- [28] 農林水産省, “一日に必要なエネルギー量と摂取の目安”, [https://www.maff.go.jp/j/syokuiku/zissen\\_navi/balance/required.html](https://www.maff.go.jp/j/syokuiku/zissen_navi/balance/required.html), 閲覧日 2023.1.22.

