

# オープンデータとデータ解析環境の現在

## —RStudioとAPI—

石田 基広

東日本大震災以降、行政府などでデータを公開する試みが進んでいる。こうしたデータはオープンデータとよばれ、民間活用による社会や経済の活性化が期待されている。また現在はデータを解析する環境が簡単に整い、ハードルも非常に低い。本稿ではオープンデータの動向とデータ解析環境について概観する。

キーワード：R, オープンデータ, データサイエンス

## 1. Reproducible Research

### 1.1 再現性の保証

データサイエンスの分野では近年 Reproducible Research (以下 RR と略す) という考えが広く浸透しつつある [1]。これは研究成果について、第三者が追試を行う環境を公開すべきという主張である。また RR は、データの整形から検証、レポートの作成までをコンピュータで自動化する手法を指すこともある。

一般にデータ分析では、あるデータに複数の分析手法を適用したり、多数のデータに同じ (ないし複数の) 分析手法を次々と当てはめるという作業を繰り返すことが多い。ここで複数のデータを切り替える、あるいは一部を取り出す、さらに分析手法を変更するなどの作業を繰り返していると、分析者自身も混乱してくるものである。特に作業段階ごとにソフトウェアを使い分けていると、ミスが起りやすくなる。たとえばデータの整形は表計算ソフトで行い、分析は専用の統計ソフト、そしてレポートやプレゼンテーションの作成には文書作成ソフトウェアを利用していると、ソフトウェア間で出力のコピーペーストなどを繰り返すことになるだろう。また、それぞれのソフトウェアで保存ファイルの形式が異なってくるため、分析を繰り返せば繰り返すほど作成されるファイル数も増え、それぞれの段階での操作と出力を管理するのが面倒になる。

RR では、データの指定や前処理から、分析手法の適用と出力の要約、さらにはグラフィックスやレポートの作成を「シナリオ」として記述しておく。これを

ファイルないしプロジェクトとして保存しておき、対応するソフトウェアで読み込めば、自動的に研究成果が再現される仕組みだ。こうしたシナリオを最初に作成しておけば、後日、同じ手順をいつでも何度でも繰り返すことができる。実行するのはシナリオを作成したユーザーでなくても問題ない。RR を実行できるソフトウェアさえあれば、誰でもいつでも分析を実行でき、結果を共有できる。本稿で紹介するフリーのデータ解析環境 R [2] および RStudio [3] にはこうした機能がある。

### 1.2 データ源

RR は、同じデータと手順を指定すれば、同じ結論が得られることを保証する仕組みであるが、肝心のデータそのものが公開されなければ、第三者が同一の分析結果を得ることはできない。手順さえ確認できればよいという場合もあるだろうが、意思決定などの根拠を提示したい場合、データを含めて公開されるのが理想的である。ただしデータはさまざまな理由から一般に公開することが憚られる、あるいは許されないという場合があるだろう。あるいはデータの性質には関係なく、収集者がデータを公開することに消極的か、適切な手段をもっていない場合もある。個人的な経験だが、国語学分野のある研究者から「自分が収集したデータを第三者と共有すると告知をしているが一度も申し出がないので、データを公開しようという動向には懐疑的だ」という話を聞いた。詳細を尋ねると、自身のホームページに入力フォームを設置し、データを希望するユーザーに名前と所属、そして利用目的を入力するよう指示されていた。

データ収集者が手元のデータを公開するにあたって制約を課すことは当然かもしれないが、その制約は個人的な見解に基づくのではなく、なんらかの統一的な

いしだ もとひろ  
徳島大学大学院総合科学教育部  
〒770-8502 徳島県徳島市南常三島町1-1  
ishida.motohiro@tokushima-u.ac.jp

基準に従うのが望ましいだろう。たとえばソフトウェアのソースを公開する場合、作者の個人的な見解で配布条件を設定するよりは、パブリックなライセンスに基づくことが望ましい。社会的に許容された形態で配布が行われることで、後日、提供者と利用者間で不毛な論争が生じるおそれは少なくなるだろう。さまざまな懸念から公開されないままになっているデータが自由に利用できるよくなれば、予想外の方向で活用され、それが社会に役に立つかもしれない。

一方で、公開されてはいるが利用しにくいデータもある。典型的なのは官公庁のデータである。こうしたサイトではデータを PDF ファイルとしてリンクが張ってある場合が多い。ただ公開されている PDF ファイルの多くが、一度プリンタから印刷された用紙をスキャンで読み取り、画像として保存されたファイルである。もちろん目視でデータを確認できないわけではない。しかし掲載されている数値をグラフで表わしたいという場合、利用者は PDF を眺めながらソフトウェアに数値などを入力し直さなければならない。もともと官公庁では Excel 形式のファイルとして保存していると思われるので、印刷後に PDF 化したりせず、元の xlsx ファイルをそのままリンク先に置いてもらえればと思う。

PDF ではなく元のデータファイルを公開しようとする動きも活発化しつつある。きっかけの一つは、5年前の東日本大震災である。当時、放射性物質の拡散や放射線の強度について政府がデータを公開していないと批判された。実際には経済産業省などのサイトで公開されている場合もあったわけだが、リンク先を探すのが簡単ではなく、またサイトに置かれていたファイルもほとんどが PDF 画像であった。このためファイルから数値の情報などを正確に抽出するのに多大な手間がかかり、中央官庁の外にいる研究者らがデータを分析し、そこから得た知見を公開する妨げになった。そうした研究者の一人である奥村晴彦氏は、当時、次のように述べている [4]。

「データは自動処理可能な形で提供してほしい」

データが表計算ソフトのファイル形式のまま公開されていれば、研究者それぞれが利用するソフトウェアで直ちに分析することができる。さらに言えば、データへのリンクがサイトのどこかに埋もれており、探し出すのに苦労するような構成ではなく、データの取得だけを目的とした URI が用意されていれば利便性が高まる。データの取得と公開にタイムラグが生じないような自動化がサーバーの側で設定されていればより理想的である。

東日本大震災以前、こうしたデータ源は日本においては全くの理想に過ぎなかった。しかし海外では当時すでにその萌芽があり、「オープンデータ」という名前前で知られてはいた。大震災以降、日本でもオープンデータへの関心が高まりつつある。特に第二次安倍内閣で策定された「日本再興戦略」[5]では「世界最高水準の IT 社会の実現」が目標として立てられ、「オープンデータやビッグデータの利活用」の推進が明記されている。

## 2. オープンデータ

### 2.1 政府の推進

日本政府によるオープンデータの取り組みとしては、平成 24 年 7 月に策定された「電子行政オープンデータ戦略」[6]と平成 25 年 6 月の「我が国政府におけるオープンデータの取組」[7]が挙げられる。後者では以下のように説明されている。

オープンデータは、単なる情報公開にとどまるものではなく、公共データを二次利用可能な形（二次利用が可能な利用ルールかつ機械判読に適したデータ形式での公開）で民間へ開放することにより、行政機関自身がサービスを提供しなくても、民間主導でネットワークを通じた多様な公共サービスが創造されることとなる。このように、オープンデータの推進により、行政の透明性・信頼性の向上、国民参加・官民協働の推進、経済の活性化・行政の効率化が三位一体で進むことが期待されている。

これらを受けて、地方自治体においても、公共データをサイトに公開する試みがなされてきている。福井県の鯖江市はもっとも早くからオープンデータ化に取り組んできており、「データシティ鯖江 (XML, RDF によるオープンデータ化の推進)」というページが用意されている [8]。また静岡県では「ふじのくにオープンデータカタログ」として専用サイトを設け、主に Excel 形式のファイルを公開している [9]。いずれもデータへのリンクを貼られたページにはライセンスとして「クリエイティブ・コモンズ」のロゴが表示されている。これは非営利団体であるクリエイティブ・コモンズが策定したライセンスである [10]。このライセンスはデータの著作権者を明示することだけを要求し、データの使用や共有、さらには二次加工も自由だとしている。

### 2.2 XML と RDF

また「データシティ鯖江」で特徴的なことはデータの

提供形式として XML と RDF が挙げられていることである。XML (eXtensible Markup Language) について解説する必要はないかもしれないが、要はデータを構造化するフォーマットのことである。たとえば「学生番号 12345 は山田太郎である」という情報は XML では以下のように表現できる。

```
<?xml version="1.0" encoding="UTF-8"?>
<student>
  <id>12345</id>
  <name>山田太郎</name>
</student>
```

XML はデータの内訳を細分化し構造的に記述したフォーマットである。人間が目視して確認するにはかえって煩雑となるが、コンピューターで処理を自動化するには適しており、この例では「学生番号」と「名前」を機械的に抽出できるようになる。

一方、RDF (Resource Description Framework) は、セマンティック WEB という技術に基づいている。これについても詳細は省くが、簡単に言えば、データ構造をグローバルに標準化しようとする試みである [11]。RDF ではデータの内訳をそれぞれ「主語」「述語」「目的語」として関連づける。この例であれば、学生番号が主語、「という名前である」が述語、「山田太郎」が目的語である。RDF では主語と述語は URI としてリソースがあるものとして表現する。たとえば次のようになる。

```
<http://or.ac.jp/stid/12345>
  http://or.ac.jp/st/name
    "山田太郎".
```

鯖江市がオープンデータとして公開しているのは、RDF に基づき XML で表現したデータである。データは構造化されているのでルールさえわかれば第三者に加工しやすい仕組みになっている。また URI で指定可能なデータ源なので、いつでもどこからでもデータを引き出すことができる。行政に限らず、構造化されたデータが URI を指定すると自由に引き出せる環境というのは、データを利用する側にとっては理想的な環境である。もちろん歓迎されるべき試みだが、一般に公共団体の構築するサイトはページ URL の変更が激しく、オープンデータへのリンクを含むページも同様に変更される可能性が高いのが気になる。

## 2.3 API

より理想的な方法は、通常のサイトとは別にデータを抽出するためのサーバーを用意してもらうことだ。たとえば総務省統計局は国勢調査などのデータを e-Stat という専用サイトで公開している [12]。このサイトでは必要とするデータを検索して Excel ファイルとして取り出すことも可能だが、より魅力的なのは API (Application Programming Interface) 機能が提供されていることである。e-Stat の通常ページの場合、データを指定するにはマウス操作を繰り返して絞り込む必要がある。たとえば「主要統計」である「家計調査」で「二人以上の世帯」の 2015 年 11 月期のデータから、米の支出を抽出しようとする、10 回以上マウスを操作する必要がある。

これに対して API ではデータを抽出する窓口の URI は固定されている。ユーザーはこの URI に必要とするデータを指定してアクセスする（ただし、事前に API 機能を使って目的とする統計表の番号や抽出カテゴリ ID を特定しておく必要はある）。

```
http://api.e-stat.go.jp/rest/1.0/app/
  getStatsData?appId=ユーザー ID&
  statsDataId=データ ID
```

この URI をブラウザに入力すると、データが XML で構造化されて表示されることになる。もっとも、XML によるデータ構造は人間が目視確認するものではない。通常は、ソフトウェアあるいはプログラムのコード内で API にアクセスし、取得された XML の構造を解釈しデータとして利用することになる。

前節の鯖江市などが提供する XML では必要とするデータを登録しているページを探し出し、該当ページ内に貼られたリンク URI をコピーしておかねばならない。これに対して API ではベースとなる URI は固定されており、ここにデータ種別を追記することで、目的とするデータを一意に指定できる。またデータを提供する側では、データベースなどから機械的に XML のフォーマットにデータを変換する作業を行う。この処理を自動化しておけば、XML 構造で取り出されるデータはデータベースの最新の状態を反映していることになる。

データ源が URI で一意に指定抽出できる環境が整ってくれば RR はより柔軟な運用ができるようになる。ただし XML 形式で提供されるデータを利用するには、その構造を解釈する方法をコンピュータに指定しな

ればならない。すなわち、ユーザーの側にプログラミングの技能が要求される。しかし、このハードルは今はさほど高くない。RR をサポートするソフトウェアとして最初の節で言及した R および RStudio を利用すると、プログラミング経験のないユーザーでも、きわめて低い学習コストで API や XML の操作が可能になる。

### 3. R/RStudio によるデータ抽出

#### 3.1 プログラミング言語としての R

R はデータ解析とグラフィックス作成のためのデスクトップアプリケーションであるが、プログラミング言語としても利用できるため、ユーザーの側で自由に拡張することができる。R はここ 10 年ほどの間に、学会や企業などで急速に普及し、統計解析のデファクトスタンダードの地位を築きつつある。また、最近の R には統計やグラフィックス以外の機能が多数加えられ続けている。その一つがインターネット上のリソースにアクセスし、取得した XML (や JSON) を解釈する機能である。

たとえば前節で総務省統計局の API について言及したが、コンピューターでアクセスしてデータを抽出する場合、次のような手順を踏むことになる。

- ・ e-StatURI とユーザ ID、統計表の ID を結合した URI を作成
- ・ その URI にアクセスして XML データを取得
- ・ 取得した XML を解析してデータ表に整理

R ではこの手順をすべて自分でプログラミングする必要はなく、すでに用意されている機能 (コード) を流用すればよい。ユーザーが自身で記述するのは URI とデータの指定だけである。もちろん XML についての知識はある程度必要となるが、基本的には HTML タグと同じ仕組みなので、見た目ほどハードルは高くないだろう。XML の仕組みがおおよそ理解でき、かつ e-Stat がデータを XML に変換するルールまでが把握できるのであれば、R で数行の命令を書けばデータを取り出すことができる。

#### 3.2 XML

実例を示そう。e-Stat の説明書に宇都宮市と浜松市で各家庭の餃子支出額を抽出する例が書かれている [13]。これを R で実行し、時系列グラフにまとめる処理はおおむね以下ようになる。

```
1: urlx <- paste0 (
  "http://api.e-stat.go.jp/rest/1.0/app/
```

```
getStatsData?appId=",
ids, "&statsDataId=0003013276
&cdArea=09003,22004",
"&cdCat01=010920070",
"&cdTimeFrom=2012000101",
"&cdTimeTo=2015001031")
2: gyoza <- xmlParse(urlx)
3: gyoza %>% ggplot(aes(x = time,
  y = value)) +
  geom_line(color = area))
```

左端に 1: とマークした行が URI を生成してアクセスする処理である。2: で XML を解読し、データを gyoza という名前で保存する。最後の 3: はグラフを作成する命令である。実行すると図 1 が生成される。実際にデータを抽出し、グラフを作成するシナリオファイル Gyoza.Rmd を [14] に公開しているので、興味のある読者はぜひ RStudio で試してみてください。

#### 3.3 WEB スクレイピング

XML ほど構造化されていなくとも、必要とするデータのフォーマットが特定できるのであれば、通常の HTML タグで記述されたページからデータを抽出することも可能である。この処理を WEB スクレイピングとよぶ。現実にはデータが XML などで構造化され API によって一意に指定できる仕組みが提供されているケースはまだ稀である。しかし通常の WEB サイトであっても、データ部分が TABLE タグなどで記述されていれば簡単に抽出することができる。

たとえばウィキペディアの「花札」にはカード一覧が掲載されている [15]。R でこれを抽出して表データとして保存するには命令を二つ実行するだけである。この命令を記載したファイル Hana.Rmd を [14] に公開しているので、興味のある方は RStudio で試してみてください。

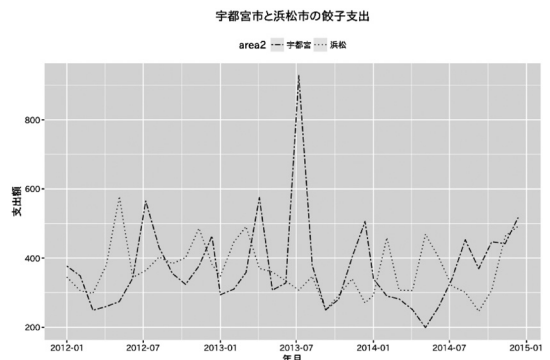


図 1 餃子への月別支出



表 1 Wiki 花札ページから

月	花	20点札(光)	10点札(種)	5点札(短冊)	1点札(カス)
1	1月 松(まつ)	松に鶴(つる) [絵札 1]		松に赤短 [絵札 2]	松のカス (2枚)
2	2月 梅(うめ)		梅に鶯(うぐいす) [絵札 3]	梅に赤短 [絵札 2]	梅のカス (2枚)
3	3月 桜(さくら)	桜に幕(まく)		桜に赤短 [絵札 4]	桜のカス (2枚)
4	4月 藤(ふじ) [絵札 5]		藤に不如帰(ほととぎす)	藤に短冊	藤のカス (2枚)
5	5月 菖蒲(あやめ)		菖蒲に八橋(やつはし) [絵札 6]	菖蒲に短冊	菖蒲のカス (2枚)
6	6月 牡丹(ぼたん)		牡丹に蝶(ちょう)	牡丹に青短	牡丹のカス (2枚)
7	7月 萩(はぎ) [絵札 7]		萩に猪(いのしし)	萩に短冊	萩のカス (2枚)
8	8月 芒(すすき) [絵札 8]	芒に月(つき)	芒に雁(かり)		芒のカス (2枚)
9	9月 菊(きく)		菊に盃(さかずき)	菊に青短	菊のカス (2枚)
10	10月 紅葉(もみじ)		紅葉に鹿(しか) [絵札 9]	紅葉に青短	紅葉のカス (2枚)
11	11月 柳(やなぎ) [絵札 10]	柳に小野道風 [絵札 11]	柳に燕(つばめ)	柳に短冊	柳のカス
12	12月 桐(きり)	桐に鳳凰(ほうおう)			桐のカス (3枚) [絵札 12]

ほしい。

```
1: wiki <- read_html(
  "http://ja.wikipedia.org/wiki/
  %E8%8A%B1%E6%9C%AD")
2: html_table(html_nodes(wiki, "table"))
```

1: でウィキペディアにアクセスし, 2: でページ内の TABLE タグ部分だけを取り出している。この結果, 表 1 のデータがパソコンに取り込まれる。

## 4. 多様な API との連携

### 4.1 地理情報の取得

API として提供されているのは政府統計ばかりではない。画像を含む地図情報を Google Maps API [16] や Open Street Map [17] の提供する API から取り出すことができる。これにより, ほかの情報源から得たデータに地理情報を加えて表示させることができる。Leaflet [18] は JavaScript による地図作成機能であるが, R でも利用できる。たとえば, 四国八十八箇所の寺院の名前と場所をデータとして取得し, これを地図にマッピングした画像を作成したいとしよう。

「四国八十八箇所」をキーワードに Google などで検索すると多数ヒットする。リンク先のページによっては, 寺院の名前だけでなく住所なども説明されていることもある。ただ筆者が閲覧した複数のサイトでは, 寺院の名前や住所は説明の文章の中に散在しており, 表などの形式でまとめられてはいなかった。もちろん読みながら拾い出せばいいのだが, できればコンピューターで自動化したい。WEB ページは HTML タグで記述されているので, 表を指定する TABLE タグでデータがまとめられていれば, プログラムで機械的に抽出できる。また住所の情報は地図 API によって緯度経度に変換することができる。緯度経度を前節で言及し

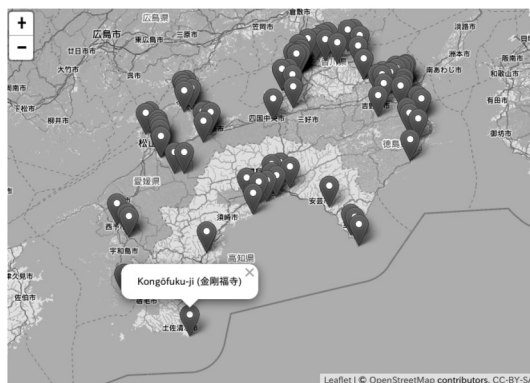


図 2 Leaflet と Open Street Map による四国八十八箇所地図

た Leaflet API へのインプットとすると, 必要な地図画像がアウトプットされる。

そこで寺院名と住所が TABLE タグにまとめられたサイトを探したところ, 海外の Wikipedia に Shikoku-Pilgrimage というページを見つけることができた [19]。

このサイトでは住所がすでに緯度経度に変換されている。そこで, R ではおおむね次の手順で地図を作成することが可能になる。

- ・ Wikipedia にアクセス
- ・ TABLE タグ部分を取り出す
- ・ 寺院の名前と位置情報を抽出
- ・ 地図 API にアクセスして作図

これにより図 2 が作成できる。

この地図では, ポップアップ表記されたバルーンをマウスクリックすると寺院の名称が表示されるようになっている。実行可能なファイル Shikoku.Rmd を [14] に公開しているので興味のある読者は確認されたい。

## 5. RによるRR

### 5.1 RStudio

APIなどを通して、リアルタイムなデータをいつでも誰でも自由に取得できる条件が整っているのであれば、Reproducible Research としては完璧である。RにはRRを簡単に実現できる機能が用意されている。一方、Rを効率的に操作できる統合環境IDEとしてRStudioというソフトウェアがフリーで提供されているが、RStudioではRRがより簡単に実現できる仕組みが整っている。最後にRStudioにおけるRRを紹介しよう [20]。

レポートの作成では、文書を段落に分けたり、引用があればこれをインデントさせて挿入したり、あるいは画像を適切な箇所に挿入するなど、細かい作業が必要となる。これらはワープロソフトではマウスなどでその都度操作して整形を行うことになろう。一方RRでは、文書を入力する段階では目印を残しておくだけで、実際の整形そのものは文書が完成した段階でソフトウェアが変換することに任せる。段落の指定や画像の挿入の目印として使われるのがMarkdown記法である [21]。Markdown記法では段落の深さを#記号の数で、ボールド体として表示する文字はアスタリスク\*で挟んでおくなどの方法で目印を入れておく。以下はRStudioで新規にMarkdownファイルを作成した際に表示される雛形を編集したファイルの一部である。

```
---
title: "Untitled"
author: "Ishida Motohiro"
date: "2016年1月20日"
output: html_document
---
# コードの挿入
チャンクというフォーマット内に命令を書く：
```{r}
summary(cars)
```

## 画像の生成と挿入
画像は作成の**命令**を書いておく：
```{r, echo=FALSE}
plot(cars)
```
```

バックティックを三つ並べた行の中間に挟まれた部

分はRで実行すべきコードであり、ここにデータを整形したり、統計的検定を行うための命令を記述しておく。この部分はチャンクとよばれるが、記述した段階で命令は実行されない。シナリオが完成した段階で、RStudioで一括実行すると、コード部分は出力やグラフに置換される。チャンク以外の部分は地の文でレポートを自由に書くことができる。ボールド体などのフォント周りの指定には\*bold\*などのMarkdown特有の方法で指定しておく。

前述したように、分析を実行するソフトと、レポートを作成するソフトが異なる場合、ユーザーは前者の実行結果をコピーして、後者のソフトにペーストする操作を繰り返すことになるだろう。データ分析作業だけに専念している場合は、実行手順を把握していられるかもしれないが、第三者に手順を渡す際に説明に苦勞するだろう。また自分自身にとっても、数日後にレポートに掲載された出力がどの作業段階での結果なのかを確認するのは簡単ではないだろう。

RRでは、報告と実行命令、画像挿入目印などを同一のテキストファイルに記載しておく。これを最終的にはPDFないしHTMLファイルに変換する作業をソフトウェア上で行う。といっても、たとえばRStudioではメニューバーに表示されたボタンの一つを押すだけである。RStudioは指定されたファイルのMarkdown記法に従って文書を段落分けし、フォントを指定どおりに変え、分析コードが挿入されている部分はこちらを実行結果に変換し、画像を挿入する目印があれば、そこに適切なサイズとフォーマットの画像を挿入したファイルを新たに出力する。

こうしたファイルをRStudioではRmdという拡張子で保存する。このRmdファイルとRStudioさえあれば、いつでも誰でも同じ出力を得ることが可能になる。サンプルとしてWikipediaから四国八十八箇所の寺院データを取得して、地図を描く命令ファイルShikoku.Rmdを [14] に掲載しているので、興味ある方はRStudioで試してみてください。

## 6. まとめ

データに基づく科学、意思決定では、仮説の設定と分析の手順、そして結論を第三者と共有することが求められるだろう。そうであれば、インプットしたデータと分析手順の透明化が望ましい。ただ現代のデータ分析では、データを探索的に分析するのが普通である。探索的分析では、たとえばデータから部分集合を取り出し解析を行うことを繰り返し、その度にグラフを作

成したり統計的検定を行ったりする。結論を導くまでには数十あるいは数百の手順の繰り返しが行われることも稀ではない。こうした作業工程を個人が手作業でメモにまとめるのは困難であり、ましてや記憶しておられるものではない。

RR ではデータから結論を導くのに必要な過程をすべてシナリオとしてファイル 1 枚にまとめることができる。このシナリオと対応するソフトウェアさえあれば、誰でもいつでも分析結果を再現し、これを PDF や HTML のレポートにまとめ直すことができる。オープンデータを RR と組み合わせることができれば、誰でもいつでも最新のデータに基づいた分析結果を自分のパソコンで確認できる。データについては、個人情報が含まれている場合もあり、そのままでは公開できない場合も多いだろう。しかし個人情報を秘匿する自動化技術も進歩しつつある。またデータをリアルタイムに収集してデータベースに保存しつつ更新し、API リクエストに応じてリアルタイムに XML などのフォーマットに変換する処理も、いまでは個人で実現できる。各種クラウドコンピューティングを利用すれば物理的なサーバーを用意する必要もない。さらに手軽に実現したければ Github などのプロジェクト管理 SNS を利用し、自身のパソコンなどから定期的にデータを送信するよう自動化スクリプトを組み込めばよい。

実際、ウェアラブル端末で日々記録した歩数データなどを Github にアップし、定期的に更新しているユーザーも見かける。FitBit であれば、収集したデータは FitBit の専用サイトに記録され続け、これを API を通して取得する仕組みがある [22]。今後は、こうした個人的なデータを個人的な目的のために分析し、個人的に活用するユーザーも増えてくるだろう。オープンデータの原理と RR は、ユーザーが自身のための分析を完全に自動化するための手段としても普及していくものと思われる。データに基づく意思決定は、企業だけでなく、個人にとっても価値あるものだからである。

## 参考文献

- [1] CRAN Task View: Reproducible Research, <https://cran.r-project.org/web/views/ReproducibleResearch.html> (2016 年 1 月 18 日閲覧)
- [2] R: The R Project for Statistical Computing, <https://www.r-project.org/> (2016 年 1 月 18 日閲覧)
- [3] RStudio, <https://www.rstudio.com/> (2016 年 1 月 18 日閲覧)
- [4] Okumura's Blog, <https://oku.edu.mie-u.ac.jp/~okumura/blog/node/2578> (2016 年 1 月 18 日閲覧)
- [5] 首相官邸, 日本再興戦略改訂 2015 の概要, [http://www.kantei.go.jp/jp/topics/2015/seicho\\_senryaku/pdf1\\_new\\_seika\\_torikumi.pdf](http://www.kantei.go.jp/jp/topics/2015/seicho_senryaku/pdf1_new_seika_torikumi.pdf) (2016 年 1 月 18 日閲覧)
- [6] 内閣官房 IT 総合戦略本部, 電子行政オープンデータ戦略, [http://www.kantei.go.jp/jp/singi/it2/pdf/120704\\_siryoku2.pdf](http://www.kantei.go.jp/jp/singi/it2/pdf/120704_siryoku2.pdf) (2016 年 1 月 18 日閲覧)
- [7] 平成 25 年 6 月の総務省情報通信白書, 我が国政府におけるオープンデータの取組, <http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h26/html/nc132110.html> (2016 年 1 月 18 日閲覧)
- [8] データシティ鯖江 (XML, RDF によるオープンデータ化の推進), <http://www.city.sabae.fukui.jp/pageview.html?id=11552> (2016 年 1 月 18 日閲覧)
- [9] ふじのくにオープンデータカタログ, <http://www.pref.shizuoka.jp/kikaku/ki-330/opendata/> (2016 年 1 月 18 日閲覧)
- [10] クリエイティブ・コモンズ, <https://creativecommons.org/> (2016 年 1 月 18 日閲覧)
- [11] 荒木雅弘, 『フリーソフトで学ぶセマンティック WEB とインタラクション』, 森北出版, 2010.
- [12] <https://www.e-stat.go.jp/> (2016 年 1 月 18 日閲覧)
- [13] 政府統計の総合窓口, API 機能の活用事例と登録方法, [http://www.soumu.go.jp/main\\_content/000320558.pdf](http://www.soumu.go.jp/main_content/000320558.pdf) (2016 年 1 月 18 日閲覧)
- [14] Ishida Motohiro's Gist, <https://gist.github.com/IshidaMotohiro/>
- [15] ウィキペディア, 花札, <https://ja.wikipedia.org/wiki/%E8%8A%B1%E6%9C%AD> (2016 年 1 月 18 日閲覧)
- [16] Google Maps API, [www.google.co.jp/work/maps^e2^80^8e](http://www.google.co.jp/work/maps^e2^80^8e) (2016 年 1 月 18 日閲覧)
- [17] Open Street Map, <https://www.openstreetmap.org/> (2016 年 1 月 18 日閲覧)
- [18] Leaflet - a JavaScript library for interactive maps, <http://leafletjs.com/> (2016 年 1 月 18 日閲覧)
- [19] Wikipedia, Shikoku Pilgrimage, [https://en.wikipedia.org/wiki/Shikoku\\_Pilgrimage](https://en.wikipedia.org/wiki/Shikoku_Pilgrimage) (2016 年 1 月 18 日閲覧)
- [20] Making Reproducible Research Enjoyable, <http://yihui.name/en/2012/06/enjoyable-reproducible-research/> (2016 年 1 月 18 日閲覧)
- [21] R Markdown: Dynamic Documents for R, <http://rmarkdown.rstudio.com/> (2016 年 1 月 18 日閲覧)
- [22] fitbitScraper: Scrapes Data from 'www.fitbit.com', <https://cran.r-project.org/web/packages/fitbit-Scraper/index.html> (2016 年 1 月 18 日閲覧)