

POSIX 中心主義に基づく大規模データ処理の実践

～2022 年フンガ・トンガ噴火における空振の可視化～

南山 真吾

金沢大学学術メディア創成センター
sminamiyama@staff.kanazawa-u.ac.jp

森 祥寛

金沢大学学術メディア創成センター
mori4416@staff.kanazawa-u.ac.jp

木下 晶文

金沢大学学術メディア創成センター
a-kinoshita@staff.kanazawa-u.ac.jp

大野 浩之

金沢大学学術メディア創成センター
hohno@staff.kanazawa-u.ac.jp

要旨

気象庁が運用するアメダス・地上観測所の 1 分毎観測値は、15 年間で累計 100 億件を超える、公的な大規模データである。このデータをプレーンテキスト形式に変換し、POSIX 中心主義に基づいたシェルコマンド群およびシェルスクリプトを用いることで、市販ノート PC 並の処理能力でも大規模データ処理が充分可能となる。

実例として 2022 年 1 月 15 日発生のトンガ空振 [1] を取り上げ、そのデータを抽出・解析し、GUI システムと連携させ大気圧変化を可視化した。本報告ではそれら手法を比較・評価し、われわれの手法が持つ再利用性と計算結果の可搬性の高さを示す。

ダス・地上観測所データの持つ有用性に着目した [3]。本報告では、2008～2022 年の累計で 100 億レコード超のデータセットを用いた、POSIX 環境下での加工、およびその解析手法を紹介し、気象庁観測記録に対して大規模データ処理を行う。

データサイエンスの分野においては、往々にして解析を行おうとする元データの件数が膨大となる。その際、理想的には、対象となる巨大データセットに相応しい規模を取り扱えるだけの、高速かつ大容量の計算資源を備えた処理系を準備し、リソースが枯渇する心配のない状態で解析に臨める事が望ましい。しかし現実には、必ずしも計算資源が潤沢に提供されるとは限らない上、メモリ、ストレージ、帯域幅、応答性能などといった技術的制約や、経済性に起因するリソースの利用上限に制約される。さらには、データを情報として活用する過程で統計処理を加えようとすればするほど、成果物に至るまでの各段階で出力された、中間生成物を介した処理が行われる。そうなれば、元データ自体の容量のみならず、中間生成物をも取り扱えるだけの総容量と性能を考慮し見積もる必要が発生する。こうした場合、対象の問題に分割統治法が適用可能か否かは非常に重要な観点だが、一度に扱うべきデータセットが、採用したソフトウェアのレコード数上限を超えてしまえば扱えなくなるし、上限付近では GUI や入出力の動作が安定しなくなったり、処理速度の低下といったトラブルを起こす原因となる事がある。これらは作業効率の悪化を招き、調査計画の手戻りにも直結する。

1 はじめに

現代では、さまざまな大規模データが各種団体・企業によって日々蓄積され、多様な目的のため利用されている。しかし、一般にその多くは知的財産の構築を目指し莫大なコストを費やして収集されている。それゆえ、一個人である学生や小規模の組織にとって、それらのデータセットへのアクセスは容易ではなく、利用許諾を得ようとするとは非常に困難を伴うのが現状である。そうした中で、気象庁は観測した気象データを公共事業として広く一般に公開しており、「数値データ、簡単な表・グラフなどは著作権の対象ではありません」[2] としている。われわれは各種ある気象庁観測データの中から、アメ

大規模データが本質的に内包するそうした性質に対し、POSIX 中心主義 [4] に基づいてデータの前処理を適切に行うことで、限られたリソースでありながらも例えば 100 億件といった大量のデータを市販の民生用ノート PC 1 台で処理し、出力を得るといったことも、条件によっては可能である。

本報告では、GUI ベースのシステムも適宜併用しつつ実施した、可視化の実例を示す。

2 大規模データ処理を取り巻く背景

2.1 大規模データの定義

大規模データとは規模による分類であり、その本質は目的をもって収集された記録のうち、件数が莫大となったデータセットの総称である。われわれは、データ量として、一般的な表計算ソフトウェアで扱えない量（約 105 万行¹以上）のデータを大規模データとして定義している。

2.2 主要な蓄積手法

前述のように知的財産の性質を帯びやすい分析手法は、具体的には公開されないことがあるが、蓄積に関してはそれほど多くの種類があるわけではない。一般的に用いられる代表的な手法には以下がある。

2.3 プレーンテキスト形式

可読性が高く、小規模な機器からクラウドまで広く扱える、最も基本的な形式である。

Web サイトのアクセスログのような行動記録の場合、後述する RDBMS へも同時に記録するが、IoT 機器から API 経由で収集したり、オフグリッドの端末から集積したデータ群を定期的にインポートするような用途にも用いられる。なお、テキストが行指向で処理されかつ各行が不定長の場合、データを処理するバッファサイズの事前の見積もりが難しくなるといった点が欠点とされる場合があるが、正しく書式を定めておくことで一般的に後処理は容易になる。

¹改良され上限は日々更新されうる。外部ツール連携を除くと Microsoft Excel で 1,048,576 行、16,384 列 [5] および Google スプレッドシートは 1,000 万セルまたは 18,278 列（列 ZZZ） [6] である。

2.4 バイナリ形式

機械が扱いやすい形式だが、可読性は低く、他の形式への変換が必要となる。ただし、ストレージの節約が期待される場合や、可読性の低さから覗き見を防止したい場合に採用されることがある。集積手法を別途検討する必要があるのは、プレーンテキスト形式同様である。

派生形式として、プロプライエタリなアプリケーションが独自に定めたファイル形式を採用する場合がある。ファイル形式の内部仕様が非公開であったり、ライセンスによる利用制限があると可搬性に影響を与える。アプリケーションの機能拡張やサポート終了・組織の解散などに影響され、後方互換性が保たれず後年再利用するのが困難となる場合がある。

2.5 マークアップ（構造化）形式

プレーンテキストから派生した形式で、文書の構造を明確にし、各レコードや個々のデータ値といった要素に対して、規格で定義したタグ付けを施す。

テキスト形式のもつ互換性の高さを維持したまま、データ型だけでなくレコード内外の要素との意味的関連までもが記述可能であり、それでいて正しく設計することで疎結合に保つことも出来る利点がある。OOP におけるインスタンスのシリアライズ/デシリアライズとも相性が良いことから、近年利用が広がっている。しかし、実データに対してタグの記述量が多くなるため、必要なデータ通信量やストレージが増大する傾向にある。また、記録時・展開時の両方で構文解析処理が必要であるので、件数が増大し、データ構造が複雑であるほど、無視できないコストとなる。

2.6 RDBMS

WWW が一般に普及した 2024 年現在、リレーショナルデータベースマネジメントシステム（RDBMS）が Web サイトのバックエンドに広く採用されている。いまや稼働中のオンラインシステム上の行動履歴は、RDBMS のログテーブルに格納する手法が一般的である。

RDBMS にはさまざまな利点があるが、大規模データ処理の観点に限れば、適切にインデックスが設計されている限り、ランダムアクセスが容易である点は他の蓄積方式に対して優位である。範囲の絞り込みや集計関数など統計処理の適用も比較的容易な部類に入る。また、リ

アルタイムに発行した SQL クエリの結果に、計画的なバッチ処理による集計済の値を組み合わせで統合する事で、運用中でも即時の統計情報を得られる点は大きなメリットである。その上、多数の条件が関連しあう複雑な条件での分析も比較的構築しやすい。

RDBMS のようなバックエンドサーバでデータを永続化する手法は柔軟性が高いが、RDBMS ではテーブル数およびレコード数の累計が大規模になるほど、処理速度が低下することが知られている。定番の対策として、設計段階で垂直・水平分割を行う、頻出する情報はオンメモリとする、定期バッチで古い情報を分離するなどの他、近年では NoSQL のような必ずしもリレーショナルでない手法が採用または併用されることもある。クラウドサービスの進歩により需要に応じてリソースを自動で追加し、スケールする事でも解決を図っている。しかし、過去の履歴の解析をするにあたっては、本質的に静的な記録であることから、必然的にシーケンシャル読み出しが多くなり、トランザクションもランダムアクセスも、大規模データ処理に関しては優位性を発揮しにくい処理が主となる。そのため、運用中のシステムに負荷をかけないよう、DB から必要なテーブルのみ書き出し別の環境で解析する、といった処理がしばしば行われる。

3 気象観測データを用いた大規模データ処理

著者の大野と森は、金沢大学にて POSIX 中心主義とシェルスクリプトを核とした講義を展開し [7] それらの講義の 1 つで、加工済のアメダス気象データを題材として、学生に対し分析テーマを立案させ、実際に分析をさせる演習を毎年行っている。

2023 年度には、第 3 クォーターに共通教育科目の演習講義として開講した「シェルスクリプトを用いた「大規模データ処理」演習 B」や金沢大学大学院講義「スマート創成科学」にて、2022 年の観測データを使って、トンガの空振現象を観測できることを示し、演習課題の 1 つとして提示した。

トンガの空振現象とは、日本時間 2022 年 1 月 15 日 17 時ごろ、オーストラリア大陸北西の南太平洋、トンガ近海の海底火山「フンガ・トンガ＝フンガ・ハアパイ」が噴火し、大規模な爆発的噴火によって衝撃波が発生した事象を指す。その圧力が海面を押し下げることで、津波が誘発された。これら衝撃波と津波が同心円状に広がり、世界中に伝播して 日本にも 15 日夜、1.2m の潮位

変動と共に空振が到達し、気象庁が観測していた潮位と大気圧に記録され、津波とともに日本列島・諸島の全域で観測された。トンガの空振現象を観測するにあたって使用した 2022 年のデータの観測点の数と観測データ総数、観測件数は、以下の通りである。

表 1. アメダス・地上観測所観測データの規模

アメダス		1,287 箇所
件数 (毎分)	2008～2022 年	9,483,485,591 行
	内 2022 年	675,483,046 行
ファイルサイズ	無圧縮テキスト	519GB
	gzip 圧縮後	38GB
地上観測所		154 箇所
件数 (毎分)	2008～2022 年	1,178,902,174 行
	内 2022 年	80,942,400 行
ファイルサイズ	無圧縮テキスト	74GB
	gzip 圧縮後	6.8GB
合計		1,441 箇所
件数 (毎分)	2008～2022 年	10,662,387,765 行
	内 2022 年	756,425,446 行
ファイルサイズ	無圧縮テキスト	593GB
	gzip 圧縮後	44.8GB

※ 2023 年 11 月 22 日時点

これらのデータは、気象庁の外郭団体である気象業務支援センターでバイナリ形式で頒布されているものを購入し、プレーンテキスト形式に変換して使用した [3]。データは、1 年毎に 1 つのファイルにまとめている。1 分毎のデータのためそのままではファイルサイズが大きいため、gzip コマンドによる圧縮を行った。

POSIX 中心主義に基づくアプローチで大規模データ処理を実践するに際し、対象にはさまざまなテーマを選択しうる。今回、処理結果を可視化しやすく規模としても適度な難易度と考え、題材として 2022 年フンガ・トンガ噴火の空振現象を選択した。その狙いとして、発生的事实とそのおおまかな時間帯が、既に明らかである点が挙げられる。発災当時この規模の空振は未知の現象であり、日本への影響範囲および被害程度が不明であったことが注目されたが、本事象が引き起こした変化の調査

に限れば発災前後の区間に絞られ、取り扱うのが容易である。したがって、事前調査の時点で事象発生日付近のデータを抽出することで、取り扱う総量が減らせると考えた。

日本の観測値に残された痕跡を気象庁観測データから検出し、GUI ツールを用いて現象の可視化を行うが、その際に可視化や可読性に寄与する必要最小限の前処理を行う。

4 POSIX 中心主義に基づいたデータ処理手法

4.1 POSIX 中心主義

われわれは、POSIX 環境下でシェルスクリプトを中心に使用し、プレーンテキスト形式のデータ処理を行う。これは松浦が提唱した「POSIX 中心主義」に基づくプログラミングである。POSIX 中心主義は、「POSIX 準拠」「代替可能性の担保」「W3C 勧告準拠」の三要素によって成り立っており、これを踏まえることで、25 年後も使用可能なプログラムを作成できる。ここで POSIX 準拠とは、プログラム作成時の基本指針として POSIX (IEEE Std 1003) 規格に極力準拠することであり、特に問題がない限りはこの文書に記載されている [8]、文法やコマンド、オプション、正規表現、保証される出力結果のみに依存したプログラムを書くものとする。

次に、代替可能性担保とは、POSIX の範囲では実装が難しい処理を実現するための指針で、今利用している依存ソフトウェア（仮に A とする）と同等機能を有する別の実装（仮に B とする）が存在し、何らかの事情により A が使えなくなったときでも、B に交換することで A を利用していたソフトウェアを継続して使える性質として定義し、これを担保しうる可能性のことである。POSIX 中心主義では、代替可能性の担保を条件に POSIX の範囲外のコマンドの使用を認めている。

最後に W3C 勧告準拠とは、クライアントサーバ構成をとるクライアント側アプリケーション開発のための指針である。

4.2 抽出と可視化に用いた機材と妥当性

本手法の実践に用いた PC 機材の一例、表 2 に示す。実行環境として Windows 11 Home 22H2(x64) 上で WSL2

表 2. 使用 PC の仕様と周辺機器

機種名	Microsoft Surface Pro 9
CPU	12th Gen Intel® Core™ i7-1255U 2.60GHz
RAM	16GB LPDDR5-5200
SSD	512GB PCIe 4.0x4
内 C: ドライブ	475GB
モニター	3840x2160 pixel / 60Hz
データ格納元	USB Memory, 256GB, USB3.0(Type-A)

環境を構築し、Ubuntu 22.04LTS をインストール後に USB メモリにアクセスするための設定を行った。

表 2 の PC は、2023 年 10 月より研究など通常業務に必要な処理能力があるとして使用を開始した。この PC は、大規模データ処理向けに特化せず、一般的な環境と変わらないよう配慮²した。起動ドライブは総容量であって、実際には OS 本体と各種業務データで使用している状況で、残容量は半分以下を想定している。

関連して、講義 [7] に際し、学生が持参した PC の諸元が大学側が示した仕様の下限の学生らを確認し、動作可否について毎回点検している。2016 年の開講当初は、メインメモリ 4GB (VRAM と共有) PC の学生若干名が同課題に取り組んだ際、当時のデータ量は 2024 年現在の 20 分の 1 程度であったが、メモリ 8GB 以上の環境と比べレスポンスに難があるとの報告が見られた。現在この状況は改善してきており、メモリ 8GB 環境の学生が若干見られた 2023 年後期においても、メモリ不足に起因する演習の実施不能や支障をきたした事例は皆無であった。

4.3 処理量の見積もり

手当たり次第に取り掛かっても結果を得られることはあるが、処理対象の規模を見積もることは、計算資源の過不足と所要時間に直結するため重要である。手始めに、

²これは、特別な環境を用意しなくとも、大規模データ処理が可能である事を示す狙いがある。高校生や大学生の所有するノート PC と比すればわずかに性能が高いが、同世代のデスクトップ PC やゲーミングノートに比べれば、モバイルプロセッサは非力である。その上、USB メモリの格納データを本体 SSD に移動させるまでは出来たとしても、すべてを展開するだけの残容量は用意できない。後述のように最初に範囲を絞り込むにしても空き容量の圧迫は避けられない。

対象となる総観測地点数を抽出し、傾向の把握を行う方針とした。

空振現象は、空気の振動であるため、アメダス・地上観測所データの中で観測されるのは気圧の変化であると考え、現地気圧のデータを観測している地上観測所データを使用することとした。この現象が発生したのは、日本時間の2022年1月15日17時ごろであることが分かっているため、地上観測所データから同日のレコードを抽出した。

gzip で圧縮し USB メモリ内に格納した気象データから、空振があった日の観測データを抽出し、その件数を数える。抽出の初段には zcat コマンドを用いた。zcat はシェルコマンドであり、以下のようにして gzip 形式の圧縮ファイルから直接解凍された結果を抽出できる。

```
$ zcat < surf_2022.txt.gz
| awk '{if ($2=="20220115"){print $0}}'
| awk '{if ($3=="1200"){print $0}}'
| wc -l
```

ここでは、日本時間で2022年1月15日の正午のデータを抜き出し、その行数を数えることで観測地点数を出力している。その結果、地上観測所データの観測施設は154カ所あった。ただし、元データセットに含まれない南極昭和基地、および現地気圧の観測値がない富士山特別地域気象観測所は除外している。要点として、zcat は gzip -cd と等価であり、解凍結果をファイルではなく標準出力へと出力するため、シェルに対してあたかもプレーンテキストを直接扱っているかのごとく振舞う。オンメモリで処理するため、別ファイルへの展開が必要なく、中間生成物を別途要しない特徴を持つ。そのため、解凍後のサイズを下回るほどストレージ残量が乏しい実行環境でも、展開結果をパイプ処理で後段へ容易に渡せる。解凍してから処理するやり方と比べ、低負荷でデータ全件を走査できる。

4.4 1 分毎の観測データの切り出し

POSIX 中心主義に基づいたシェルコマンド群を使って、空振があった日の観測データを抽出する。

```
$ zcat < surf_2022.txt.gz
| awk '{if ($2=="20220115"){print $0}}'
| less
$ zcat < surf_2022.txt.gz
| awk '{if ($2=="20220115"){print $0}}'
| tee /tmp/surf-20220115.txt
| wc -l
```

実際に抽出されているかどうかの確認は、次のコマンドによって行う。

```
$ cat /tmp/surf-20220115.txt
| awk '{if ($1=="47605"){print $3,$18}}'
| less
```

結果、00:01 から翌日 00:00（データ上は 2400 と表記）を抽出範囲としたため、各地 1 分毎のデータとして 1,440 件/箇所 × 154 箇所、22 万 1760 件となった。

抽出した surf-20220115.txt が使用可能なので、任意の観測地点番号 1 カ所について、抽出し TSV³形式のファイルを作成した。

```
$ zcat < surf_2022.txt.gz
| awk '{if ($2=="20220115"){print $0}}'
| awk '{if ($1=="47605"){print $1,$3,$18}}'
| sed 's/ /\t/g'
| tee /tmp/surf-20220115-Kanazawa.tsv
```

作成した TSV ファイルを Google スプレッドシートに読み込ませ、時間帯を指定して⁴切り出し、全区間を対象として図 1 のピボットテーブルを作成し、気圧を降順で整列したのち、図 2 のように 1 分刻みの折れ線グラフとして描いた。

極めて簡単ではあるが、これらのコマンドの使用によって、必要とされるデータを抽出することができた。この段階では、1 つの観測地点であれば、1,440 件のデータであり、全観測地点のデータであっても 22 万 1760 件のため表計算ソフトウェアなどを使用することも可能となる。

³TSV(Tab-Separated Values) は、複数の値をカンマで区切って列挙する CSV と同種のデータ形式で、区切り文字にタブ文字を用いる。CSV の方が知名度としては知られているが、タブ自身を含まない限り、エスケープ処理を必要としない上、インポートも容易である利点を持つ。

⁴当初、事象が日本近海の大気圧に影響を及ぼしていた時間帯の範囲についてはあらかじめ見当を付けて決定したが、作成後に南鳥島が通過後であったことが判明し、切り出しの手順まで戻って範囲を決め直している。

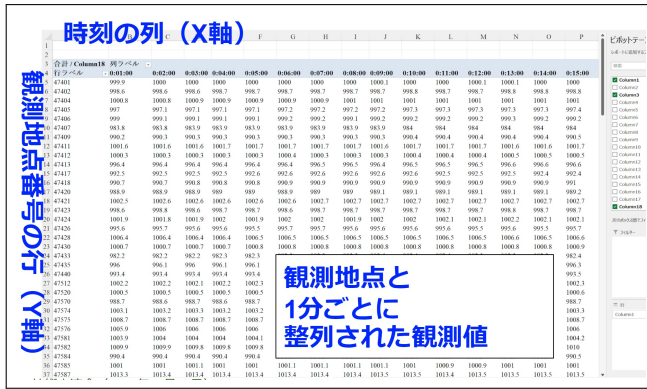


図 1. ピボットテーブルによる時系列の集計

5 データ処理後のデータ表現手法

われわれのデータ処理においては、すべての処理を POSIX 中心主義に基づいたシェルコマンド群によって作業することもできるが、今回は 必要な処理量の見積もりをつけ、データの切り出しなどを行った後であれば、表計算ソフトウェアなどを使用しても構わないことにした。以下では 抽出したデータを元に表計算ソフトウェアや 3D 可視化について示す。なお、用いた可視化手法の特徴・利点・欠点は表 4 にまとめた。

5.1 探索範囲の絞り込み

衝撃波が空气中を伝わる速度と距離を考えると、空振の日本の地上観測所への到達までに少なくとも数時間程度を要する。地上観測所データの観測地点に南鳥島や父島が含まれており、本土と島しょ部の距離の差は、絞り込み区間に影響する。そこで、辺縁に位置する観測所数カ所についてトンガからの直線距離を調査し、到達時刻にどのくらいの時間差が想定されるかを確認した。距離の調査には Google Maps の機能を利用した。

表 3. トンガ火山一観測拠点間の距離

南鳥島気象観測所	5,987.2km	最南最東
父島気象観測所	6,994.6km	2 番目
根室特別地域気象観測所	8,527.1km	本州最東
稚内地方気象台	8,521.1km	最北
与那国島特別地域気象観測所	8,332.4km	最西
(南極昭和基地を除く)		

気象庁の観測網は、当該火山から眺めて西北西から北

北西にかけての範囲約 62.6 度の方角に分布している。当然ながら発災前後の区間を含むが、同時に各観測地点での平常時の気圧変化も、比較のため含まれるように期間を検討する。

5.2 空振到達の傾向の確認 (グラフによる可視化)

観測地点番号の行 (Y 軸) と時刻の列 (X 軸) で、ピボットテーブル (集計した表) を作成する。空振の到達時刻付近を割り出し、その区間のみ抽出した。その際、観測地点ごとに各時刻の測定レコードは 1 件とした。また、南鳥島や父島などの南洋・小笠原諸島へ先に到達するため、その直前を始点とした。

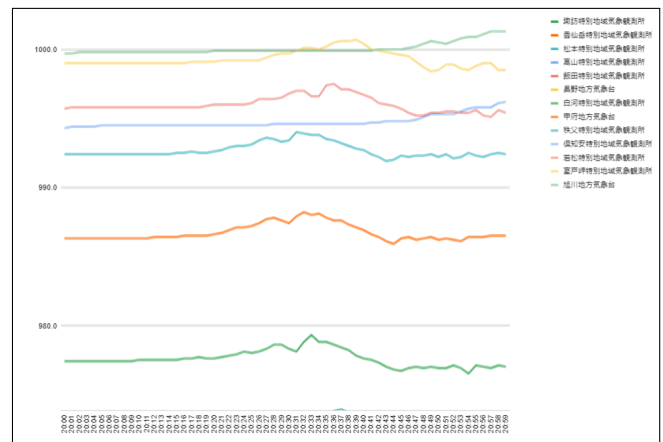


図 2. 可視化 1: グラフ (Google Spreadsheet)

これにより、観測地点単位では空振とみられる気圧の変動を観測することができたが、変動幅が 1~2hPa 程度であったため、大気圧 (およそ地上で 1000hPa 付近) に対して微小で、生の値から生成したグラフでは、変動が捉えにくいことが分かった。

5.3 集計結果に対する補正

より直感的な可視化のために、蒸気圧の各観測値に、地点毎で正規化 (に近い処理) を適用した。空振による変動は 1~2hPa 程度で、大気圧 (およそ地上で 1,000hPa 付近) に対して微小でそのままの値では変動が捉えにくく、観測地点ごとに気圧は時々刻々と変化するうえ、高度も異なっている。そのため、広い範囲の絶対値で扱うと空振以外の条件に影響を受け、波形が揃わなかった。当初、特定の時刻 (20:00) を基準とすることを試みたが、南洋の島が期待通り可視化されなかった。

そこで、後からバイアス値で調整する前提に立ち、観測点毎に期間内での中央値を求め、中央値からどれだけ離れているかを個々に算出したのち、各可視化手法を試みた。各可視化手法では、原則として波形の形状にできるだけ変化が起きないように、位置を調整する程度に留め全体にバイアス値を加える加工とした。

5.4 Microsoft Excel による表の染色

ピークが特徴的であるので、時系列でなるべく到達順となるよう並べ替えた表を作成し、各セルを染色することで更に特徴を強調した。そのため、作成したピボットテーブルに対して、「最大値のセルが最初に出現する位置」を算出した列を作成し、昇順で整列した。当初は恣意的な閾値を決めセルの色を変更していたが、手作業では時間がかかるため、Excel の条件付き書式（カラースケール）機能で3色スケールを適用し、染色する方針に変更した。最低値と中央値、最大値の3色で塗り分けている。

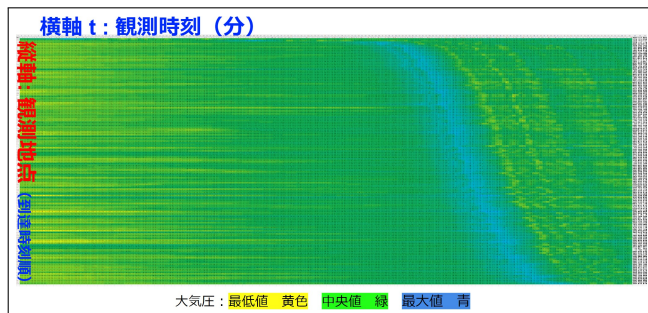


図 3. 可視化 2: セルの染色 (Microsoft Excel)

横軸 t は観測時刻 (分)、縦軸を観測地点 (到達時刻順) とし、大気圧の最低値を黄色、中央値が緑色、そして最大値が青色とした。配色の選定に当たっては、彩度が高い原色かつ色覚多様性に配慮し、ブレンドされた際に見にくくならないよう調整した。調整に際しては、スマートフォン向けに無料で提供されている色覚シミュレーションツール『色のシミュレータ』[9]を確認のため利用している。

5.5 Microsoft Excel の 3D Maps によるプロット

Excel2016 以降では、「3D マップ」機能が標準実装されている。この機能では、世界測地系で緯度・経度の列を与えると、出来合いの地球儀にプロットすることが可能

であり、ツアーという単位で時系列でのスライドショーが作成できる。そこで次の段階では、この機能をフロントエンドとし、国内の観測所の座標を地図上に反映することで位置関係の可視化を試みた。この際、気象庁が公開している地上観測所の位置を示す緯度・経度を使用した。これは、地域気象観測所一覧 [10] に「度分単位 (0.1 分精度)」で格納されており、分の列は 60 進数の表記であったため、10 進数による基数表記に変換した。地球儀上にプロットする条件として使用するため、観測地点マスタに含めてあった、観測地点それぞれの緯度・経度をピボットテーブルに連結した。

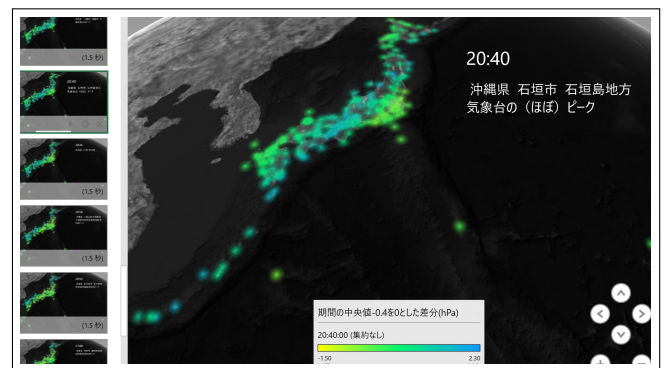


図 4. 可視化 3: 地球儀プロット (Microsoft Excel 3D Maps)

その後、必要な列がどれかを手作業で指定して、1 分毎の画面を作図することで、時系列でのアニメーションを作成した。図 4 では、デフォルトの棒グラフではなく、ヒートマップの機能を流用している。

5.6 Blender を用いた 3D 可視化

分析結果の可搬性の実例を示すため、Blender v3.5.1 および Python 3.10.9 のプラグイン実装機能を用いての 3D アニメーションの作成を試みた。前節にて抽出・処理したデータを TSV ファイルとして出力し、該整形済データソースを基に、観測拠点の気圧変化を示す 3D アニメーションを作成した。Blender は Python スクリプトから Blender オブジェクトを操作する API を有しており、TSV ファイルの読み込みと、各地点の気圧の高低に対応して個別に長さが変形する直方体オブジェクトのアニメーションを組み合わせ、タイムライン再生を実現するスクリプトを記述し実行した。

表 4. 各可視化手法の比較

手法	使用ツール	次元	作業時間	実行負荷	可搬性/流用性	将来性/大規模耐性
グラフ	Google スプレッドシート	2D	0.5h～	軽	低	一覧しにくい 不要にはならない
セル染色	Microsoft Excel	2D	1h～	面積比例	低	画面の面積が上限
3D Maps	Microsoft Excel	2.5D	4h～	重	2D 動画のみ	富豪的に解決か
Python3 プラグイン	Blender	3D	2～3h	軽	多種形式に対応	高め

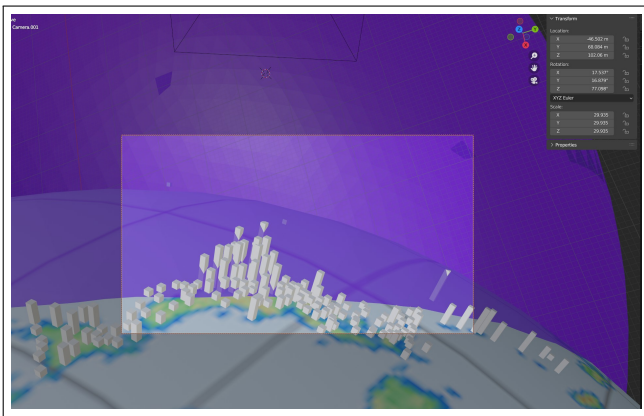


図 5. 可視化 4:3D アニメーション (Blender)

6 評価と考察

6.1 本実践の利点

今回、短期間に発生した災害が及ぼした変動を扱う上で、適切な負荷となるよう調査対象区間から切り出し絞り込む判断をした。もし全区間を対象とする集計・分析を試みるならば、相応の処理時間をかけて行う予想となるが、個人所有の PC であれば、夜寝る前に実行しておき、翌朝に結果を得るという感覚で、比較的容易に立てた仮説を確かめることが出来る。たとえば、観測地点ごとに、15 年間の全観測値を対象とした最高・最低・平均気温を調査したり、特定の地域に対する傾向を調べる、メンテナンスや故障の傾向を抽出することもできるだろう。

PC から見ると、ストリームとして渡された文字列を逐次的に処理するため、プレーンテキスト形式の大規模データは、シーケンシャルに取り扱える。その処理系が取り扱える処理能力に律速され、POSIX コマンドによ

るフィルタ処理の対象にでき、コマンドをパイプで連結することで、中間ファイルを介さずオンメモリのまま処理間の連携が行える。データ全件に対するコマンドを実行しても、時間はかかるが分析が可能であるが、このように gzip 形式から解凍し、ファイル展開する代わりに標準出力へと渡すよう設計された由来を遡ると、磁気テープに保管したデータを取り出して処理していた時代に培われた、限られた計算資源を効率よく利用する創意工夫へとたどり着く。

データセットから特徴を抽出し結果の意味を解析することを主目的とする場合に、処理効率（計算速度やリソース消費）を重視しすぎると、他の重要な要素が軽視される可能性がある。短期間で意味を失う分野では、この処理効率の評価が過大評価されがちである。しかし、長期的に見れば、結果の再現性やデータの可読性も重要である。単純なテキスト形式のデータは、100 年後でも読める可能性が高いため、長期的な保存や再利用が期待できる。計算に直接関わるリソース消費と実行時間の大小に関わる効率は、POSIX 中心主義の観点でも支配的な要素となりやすいが、これらは計算フロー途中の工程の一部に過ぎない。われわれが考えるべきなのは、機器やデータセットの導入コスト、システム構築や実装の容易さ、処理フローの自動化のしやすさ、運用維持コスト、他分野への転用の容易さなど、様々な要素であり、計算環境や使用するプログラミング言語などの変化に対応するための更新や、データの長期的な保管についても考慮するべきである。

本実践手法の学習ならば、大規模データ処理の入門者にとって、表面的なデータ構造や処理手法に留まらず、規模と効率の視点で大規模データを俯瞰する視野を会得しやすい。2024 年現在、20 億件までのデータが処理でき

るとされている PowerQuery は、Excel のバックエンドとして実装されており、前処理を Excel に担当させないという点で当手法と同様のアプローチを取っている。しかし、実行環境で動作が重くならないような操作を注意して習熟する必要がある、解析対象とする大規模データの中身に興味を持つ前に、変換方法の探索に時間を費やす事になる。気象観測は人々の生活に密着した情報を蓄えており、興味の対象としても好適である。大規模データの前処理のために POSIX コマンドを活用し、実際に取り扱いながらデータ処理への理解を深められる。初学者が大規模データ処理を身に着ける第一歩として、気象観測データの採用は相性がいいと言えるだろう。その上で、第5節と表4で示したような4種類の異なる可視化手法を用いると、それぞれの手法による特徴を活かして処理結果を示すことができる。

6.2 POSIX 中心主義への考察

本実践においては、POSIX 中心主義を導入している。POSIX 中心主義の3大特徴の一つである「POSIX 準拠」に沿ったコマンドの利用を推進しようとする、いくつかの有用なコマンドの「方言」が問題となることがある。ここで言う方言は、UNIX 系 OS が BSD 系統と AT & T 系統に分裂していた時代に作られたコマンドに同じ名前でオプションや挙動が異なるコマンドがあることを指している。

この方言問題の解消は各 OS 実装で進められている。このため、この問題に分析に着手する時点でまず利用可能な最新バージョンを用いるべきである。例えば Oracle 社の Solaris において、バージョン 10 の時点ではいわゆる方言問題を抱えるコマンドセットが標準パスにあった。しかしバージョン 11 が登場し、POSIX に準拠したコマンドセットが登場し、POSIX に定められた方法 (getconf PATH) でコマンドパスを設定すると、POSIX 準拠版のコマンドセットが収録されたパス (/usr/xpg6/bin: /usr/xpg4/bin) が優先されるようになった。この他にも、それぞれの OS 製品やコマンドパッケージにおける過去のものとの互換性を理由に、デフォルトでは POSIX 非準拠な（いわゆる方言問題を抱える）動作をするものもあるが、そのような製品であっても、環境変数を設定しておくことで、POSIX 標準な動作をするようになるものもある。例えば、HP-UX では `export UNIX_STD=2003` を設定しておけば、OS に標準添付されているコマンドは

POSIX.1-2001 に準拠した動作になり、GNU coreutils においては `export POSIXLY_CORRECT=1` などと設定しておけば、POSIX に準拠した動作をするとしている。また、POSIX 中心主義による処理には人目を引くような新規性は存在しない。しかし、新規性がないように思えることはとても重要である。本実践では、Python などによる最新のデータ分析手法の「今、ここで分析できる」ことだけでなく、POSIX に準拠した OS 間での互換性という、いわば空間的な広がりに加え、10年20年という長い時間が経過しても実践できるといった時間的な広がりも確保している。この点に目をつけたわれわれの実践は2024年現在主流となっている手法とは異なっているが、「古さの中に新しさをもち時間の流れに対抗できる手法」である。

7 今後の課題とまとめ

今回、われわれは気象庁の提供するデータを用い、大規模データ処理技能の習熟に資する成果を得た。われわれの研究には、要素同士の相互参照が比較的多くランダムアクセスが求められる種類の、例えばクラスター解析のような実践が可能で、公共性が高く、かつプライバシーの問題と無縁なデータセットが必要である。今後それら条件を満たす大規模データが見つかった場合、われわれの POSIX 中心主義に基づくアプローチが有効であるかは検証が必要である。

また、3D アニメーションによる可視化では Blender と Python を使用したが、代替可能性担保の観点からは他手法の検討も必要である。認知しているが今回試みなかった 3D 可視化手法に対しても、本手法が適用可能かどうかは、今後も継続して調査の対象としたい。

本研究では、アメダス大規模観測データから、トンガ空振という気象現象に対して痕跡の抽出を試みた。その際、POSIX 中心主義に基づいて、POSIX 準拠のシェルコマンド群を効果的に組み合わせることにより、単独のノート PC および最小限のストレージのような限られた計算資源でも、億単位レコードのデータセットに対する抽出が十分可能であることを示した。そして、CLI ベースのシェルスクリプトと Microsoft Excel、Blender といった GUI ツールの相互連携、および可搬性の高いデータ形式の採用により、微小な気象現象を効果的に可視化するに至った。

市販のノート PC やこれらのソフトウェア資源は、学

生やアマチュアにも入手可能な機材であり、かつ、比較的入手が容易な公開情報が基となった大規模データを、扱いやすい形式へ変換することを経て、取り扱いやすい題材となることが分かった。このことは、必ずしも高速な計算資源や高価な解析ソフトウェアを用いずとも、大規模なデータ処理の実践を個人的に行うことが可能であることを意味する。また、POSIX 準拠で長い実績のある CLI ツールをワークフローの各段で積極的に採用し、個々の処理を汎用性の低い方法で独自実装することへの依存度を低くすることにより、各々をより疎結合に保てる。実行系におけるバージョン間差異による潜在的な挙動の違いは、往々にして演算結果の信頼性にも影響を及ぼしかねないリスクを内包しているが、出力結果の同一性が長期間保障される「枯れた」ツールを採用することで、後年の追試や他用途への流用がより容易となるメリットが生まれる。このことは、長期間依存することで料金プランや使用制限の変更、サポートの終了などに影響を受けてしまうプロプライエタリなプロダクトと比較してより顕著だが、それに限らず他ソフトウェアに依存せずデータ転用が容易であることは、再現性を保障する観点からも大きな利点となる。

なお、本報告に関する発表資料、追加情報、修正情報などは、以下に掲載する。

<https://bit.ly/3V0jRRn>



謝辞

本研究の一部は JST【ムーンショット型研究開発事業】【JPMJMS226C】の成果を活用している。また、本研究は産学連携共同研究として、ユニバーサル・シェル・プログラミング研究所支援の下、金沢大学学術メディア創成センターにて実施された。関係各位、特に POSIX 中心主義を主導する松浦智之氏のご厚意ご高配に深く感謝する。

参考文献

- [1] 気象庁, Japan Meteorological Agency. 気象庁が噴火警報等で用いる用語集 火山活動全般に関する用語. <https://www.data.jma.go.jp/svd/vois/data/tokyo/STOCK/kaisetsu/kazanyougo/katsudo.html>. Accessed: 2024-05-27.
- [2] 気象庁総務部総務課広報室. 著作権・リンク・個人情報保護について. <https://www.jma.go.jp/jm>
- [3] 森祥寛, 木下晶文, 南山真吾, チャワナットナカサン, 大野浩之. データサイエンス教育における大規模データ処理演習への提案～シェルスクリプトを用いて, 100 億件を越えるアメダス・地上観測所データを処理する方法とその教育～. *JSiSE research report*, Vol. 38, No. 6, pp. 84–91, March 2024.
- [4] 松浦智之, 大野浩之, 當仲寛哲. ソフトウェアの高い互換性と長い持続性を目差す POSIX 中心主義プログラミング. マルチメディア, 分散協調とモバイルシンポジウム 2016 論文集, pp. 1327–1334, July 2016.
- [5] Microsoft. Excel specifications and limits - microsoft support. <https://support.microsoft.com/en-us/office/excel-specifications-and-limits-1672b34d-7043-467e-8e27-269d656771c3>, 2024. Accessed: 2024-05-27.
- [6] Google. Google ドライブに保管可能なファイル - google ドライブ ヘルプ. <https://support.google.com/drive/answer/37603>, 2024. Accessed: 2024-05-27.
- [7] 森祥寛, 松浦智之, ナカサンチャワナット, 大野浩之. シェルスクリプトを中心とすることで高い普遍性を持たせた情報技術教育プログラムの設計と実装. 情報処理学会論文誌デジタルプラクティス (TDP), Vol. 4, No. 2, pp. 67–84, April 2023.
- [8] The Open Group. PASC committee: What is POSIX? the IEEE POSIX standards are all currently entitled. <https://collaboration.opengroup.org/external/pasc.org/plato/>. Accessed: 2024-05-27.
- [9] 浅田一憲. 色のシミュレータ. <https://asada.website/cvsimulator/j/>. Accessed: 2024-05-27.
- [10] 気象庁, Japan Meteorological Agency. 地域気象観測所一覧. https://www.jma.go.jp/jma/kishou/known/amedas/ame_master.pdf. Accessed: 2024-03-18.