

Pythonで公的統計APIのオープンデータ活用

Ryo YOSHI

- Twitter: Ryo YOSHI@well_living_ry
- GitHub: WeLLLiving@well-living

自己紹介

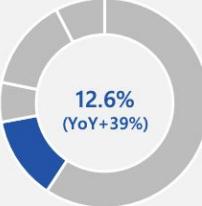
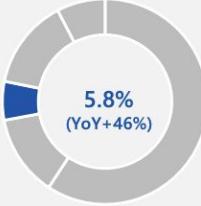
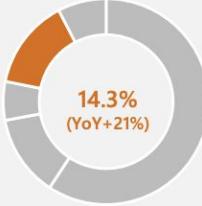
Ryo YOSHI @ well-living

- 所属
 - 株式会社マネーフォワード 
- Pythonでデータ分析とかしてます
 - PyConは2017初参加,2018ポスター,今回初トーク
- コミュニティ
 - fin-py(金融 x Python)によく出没
 - <https://fin-py.connpass.com/>
- 興味
 - ファイナンシャル・プランニング、社会保障制度
 - Pythonで自分の確定拠出年金(DC)の資産配分決めたり、自分の家計分析したりしてます
 - 趣味は、読書、旅行、など
- SNS等
 - note: https://note.com/well_living_ry
 - zenn: Ryo YOSHI@wellliving
 - Qiita: https://qiita.com/well_living
 - Twitter: Ryo YOSHI@well_living_ry
 - GitHub: WeLLiving@well-living



↑作画:@akiraturuさん

【会社紹介】株式会社マネーフォワードの事業

法人向け	個人向け	金融機関向け		
Money Forward Business	Money Forward Finance	Money Forward Home	Money Forward X	
<p>バックオフィス向けSaaS</p> <p>バックオフィス (経理財務・人事労務・法務・情シス) 向けクラウドサービスを提供 21万以上の課金顧客、 約半数が法人</p>  <p>59.6% (YoY+45%)</p> <p>12.6% (YoY+39%)</p> <p>5.8% (YoY+46%)</p> <p>14.3% (YoY+21%)</p> <p>7.8% (YoY△23%)</p> <p>Money Forward クラウド STREAMED Manageboard V-DNE クラウド Money Forward IT管理クラウド HITTO</p>	<p>SaaSマーケティング支援</p> <p>スマートキャンプ社が運営、 国内最大級のSaaSマーケティング プラットフォーム</p>  <p>12.6% (YoY+39%)</p> <p>BOXIL SaaS BALES BOXIL EXPO ADXL ADD SHIRU</p>	<p>ファイナンスサービス</p> <p>企業間後払い決済サービス、売掛金早期 資金化サービスを提供 MUFGと提携し、オンライン売掛金早期資 金化事業の提供</p>  <p>5.8% (YoY+46%)</p> <p>Money Forward Kessai Money Forward Early Payment SHJKJN SEJKYU</p>	<p>PFM（家計簿・資産管理） サービス</p> <p>1,330万以上の利用者、 国内シェアNo.1¹ 38万人以上の課金ユーザー 約18兆円²の連携資産残高</p>  <p>14.3% (YoY+21%)</p> <p>Money Forward ME Money Forward 固定費の見直し Money Forward お金の相談</p>	<p>Fintech推進・DX支援</p> <p>85のサービスを提供（共同開発含む） アカウントアグリゲーション基盤を活用し、 個人/法人、金融機関等、Fintech企業 をつなぐ、Fintechプラットフォームを提供</p>  <p>7.8% (YoY△23%)</p> <p>Money Forward X Money Forward Fintech Platform</p>

* 2022年11月期 第2四半期決算説明資料8ページ

- note「マネーフォワード・データ&AIマガジン」でデータ活用事例等を発信しています。
 - <https://note.com/mfdata/m/m6a5006afd25f>

但し書き

- 本資料に記載された情報は発表者にて判断した情報源を元に個人が作成したものであり、所属組織(マネーフォワード)が作成したものではありません。
- 本資料に記載された内容は、資料作成時点においてのものであり、予告なく変更する場合があります。
- 本資料の内容および情報の正確性、完全性等について、何ら保証を行っておらず、また、いかなる責任を持つものではありません。
- 本資料は、政府統計総合窓口(e-Stat)のAPI機能で取得したデータを使用していますが、資料の内容(数値)は国によって保証されたものではありません。
- 70ページ以上あります。巻きで発表します。気になる箇所は動画でスロー再生ください。
- 発表中、咳をするかもしれません、持病があります。

前提知識

- このトークはライブラリの requestsとpandasを利用したことがある程度の知識で十分な内容
 - requests
 - ドキュメントのリンク→<https://requests.readthedocs.io/en/latest/>
 - pandas
 - ドキュメントのリンク→<https://pandas.pydata.org/>
- グラフはPlotlyを使っています
 - Plotly
 - ドキュメントのリンク→<https://plotly.com/python/>
 - 参考書籍→@driller・小川 英幸・古木 友子,『Python インタラクティブ・データビジュアライゼーション入門—Plotly/Dashによるデータ可視化とWebアプリ構築—』,朝倉書店,2020

発表コンテンツ

- はじめに
- 公的統計とオープンデータ
- 世界と米国の公的データ
- pandas-datareaderでデータ取得
- 日本の公的データ
- 公的統計APIとe-Stat
- Pythonでのe-Statデータの扱い方
- 事例:家計調査のデータ活用
- 公的統計のこれから
- おわりに

日本の統計に関するクイズ

(1)日本の人口は約1億2千6百万人。では、15～64歳の生産年齢人口とそれ以外の人口、どちらが多い？(2021年)

(2)日本の企業数は、「情報通信業」と「医療、福祉業」だとどちらが多い？(2021年)

(3)60～69歳の平均貯蓄額は3,000万円を超えている？(2021年)

(4)平均的な家計で、交通費、光熱・水道費、教養娯楽費のうち、2022年1～6月と2年前の2019年1～6月を比べて、増えた支出は？

日本の統計に関するクイズ

(1)日本の人口は約1億2千6百万人。では、15～64歳の生産年齢人口とそれ以外の人口、どちらが多い？(2021年)

人口推計(国勢調査)を使う

(2)日本の企業数は、「情報通信業」と「医療、福祉業」だとどちらが多い？(2021年)

経済センサスを使う

正解は資料を最後まで
ご覧ください

(3)60～69歳の平均貯蓄額は3,000万円を超えている？(2021年)

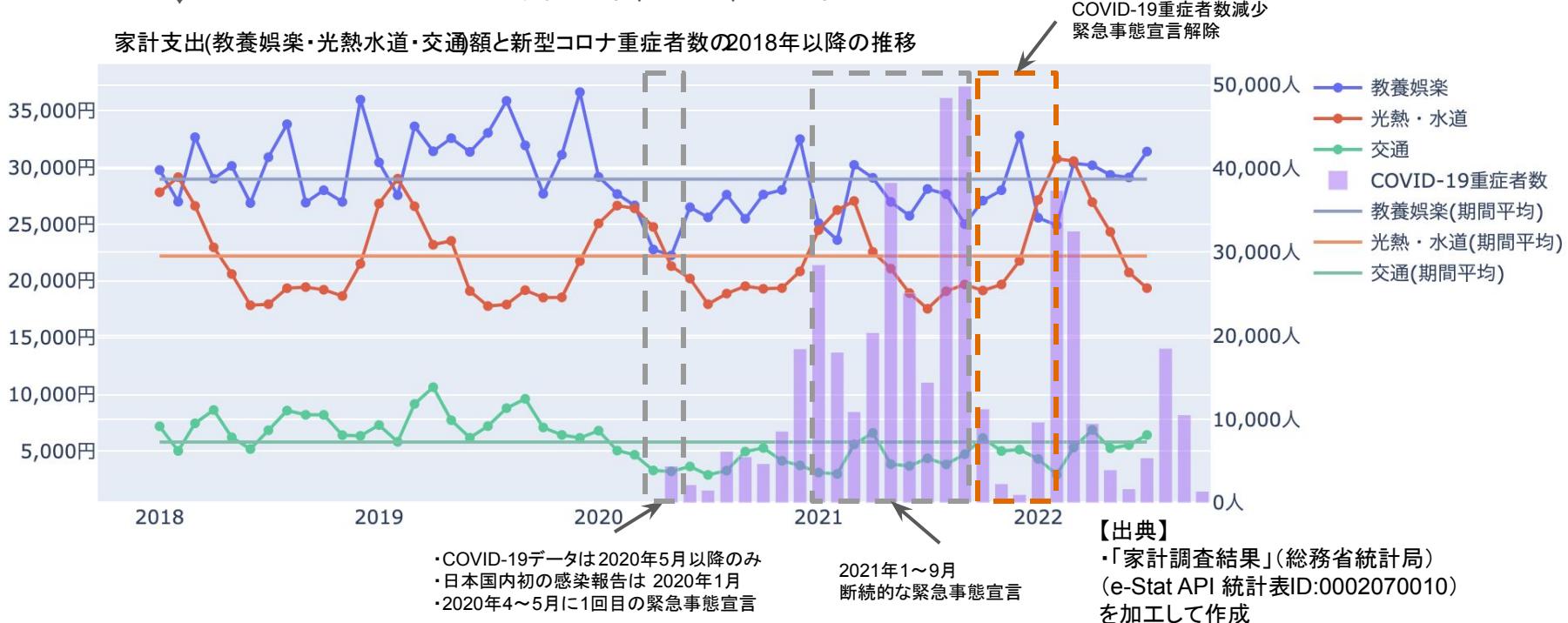
家計調査を使う

(4)平均的な家計で、交通費、光熱・水道費、教養娯楽費のうち、2022年1～6月と2年前の2019年1～6月を比べて、増えた支出は？

交通費：約36千円→ 約23千円
光熱・水道費：約15万円→ 約16万円
教養娯楽費：約17万円→ 約15万円

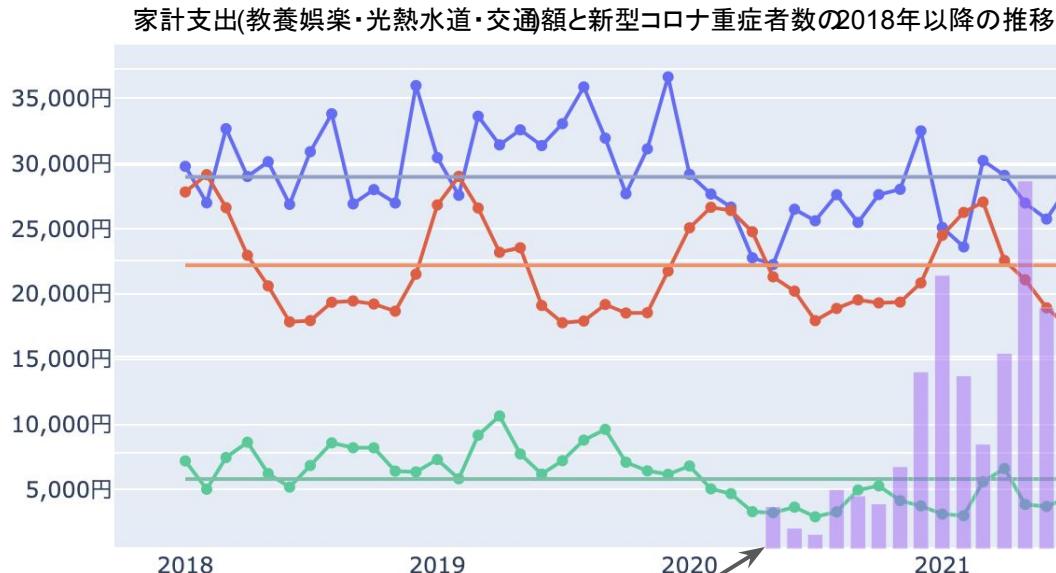
世の中の変化をデータで見てみよう！

- 2020年以降、テレワークやイベント自粛といった生活の変化
- PyCon JP 2020, 2021もオンライン・小規模会場に..(今年は有明で発表！スタッフの皆様に感謝)
- 下図↓、家計の支出において、教養娯楽費(旅行等)、交通費は減少？



世の中の変化をデータで見てみよう！

- 公的オープンデータを使って可視化しています



総務省「家計調査家計収支編二人以上の世帯用
途分類月次」
<https://www.stat.go.jp/data/kakei/>



厚生労働省
「新型コロナウイルス感染症について-
オープンデータ」
<https://www.mhlw.go.jp/stf/covid-19/open-data.html>

【出典】
・「家計調査結果」(総務省統計局)
(e-Stat API 統計表ID:0002070010)
を加工して作成

公的統計やオープンデータを見ると世の中のいろんなことが見えてくる

- そもそも公的統計やオープンデータって何？
- 公的統計やオープンデータでどんなことに使えるの？
- 公的統計やオープンデータって簡単に使えるの？
 - データの扱い、結構大変なことも..
 - 何とかしてよ



このテーマを選んだ理由、持ち帰れるもの、話さないこと

- このテーマを選んだ理由
 - 個人では、家計調査や年金などの公的オープンデータを使った記事等を書いてきた
 - 各会社でも、公的オープンデータを活用するケースが出てきており、公的統計の収集から加工までを効率化・自動化したかった
 - その際Pythonならデータの取得から集計、分析まで一気通貫で扱いやすいメリットがある
 - 公的統計は多種多様なデータがあり、知っていると仕事や趣味に役立つ
- このトークで、持ち帰れる知識やノウハウ
 - 公的統計に関する知識（公的統計の探し方など）
 - 公的統計APIをPythonで扱うノウハウ
 - おまけとして、事例で扱う家計に関する知識
- 話さないこと
 - 統計データの話で統計学的な分析の話ではないです

公的統計とオープンデータ

- はじめに
- 公的統計とオープンデータ
- 世界と米国の公的データ
- pandas-datareaderでデータ取得
- 日本の公的データ
- 公的統計APIとe-Stat
- Pythonでのe-Statデータの扱い方
- 事例:家計調査のデータ活用
- 公的統計のこれから
- おわりに

【簡易】そもそも、公的統計とオープンデータとは

一応、ちゃんと日本国政府が統計法とかで定義してたりします

- 公的統計
 - “国の行政機関・地方公共団体などが作成する統計”
 - 参考URL→https://www.soumu.go.jp/toukei_toukatsu/index/seido/1-1n.htm
 - “合理的な意思決定を行うための基盤となる重要な情報”
 - 参考URL→<https://elaws.e-gov.go.jp/document?lawid=419AC0000000053>
- オープンデータ
 - 二次利用可能で、機械判読に適していて、無償で利用できる形で公開されたデータ
 - 参考URL→https://www.soumu.go.jp/menu_seisaku/ictseisaku/ictriyou/opendata/

【詳細】公的統計とオープンデータとは

日本国政府が統計法とかで定義しています

- 公的統計
 - “国の行政機関・地方公共団体などが作成する統計を言います。統計調査により作成される統計(調査統計)のほか、業務データを集計することにより作成される統計(いわゆる「業務統計」)や他の統計を加工することにより作成される統計(加工統計)についても公的統計に該当します。”(総務省)
 - 参考URL→https://www.soumu.go.jp/toukei_toukatsu/index/seido/1-1n.htm
 - “合理的な意思決定を行うための基盤となる重要な情報”
 - 参考URL→<https://elaws.e-gov.go.jp/document?lawid=419AC0000000053>
- オープンデータ
 - “国、地方公共団体及び事業者が保有する官民データのうち、国民誰もがインターネット等を通じて容易に利用(加工、編集、再配布等)できるよう、次のいずれの項目にも該当する形で公開されたデータ”(総務省)
 - “営利目的、非営利目的を問わず二次利用可能なルールが適用されたもの”
 - “機械判読に適したもの”
 - “無償で利用できるもの”
 - 参考URL→https://www.soumu.go.jp/menu_seisaku/ictseisaku/ictriyou/opendata/
 - “官民データ活用推進基本法(平成28年法律第103号)において、国及び地方公共団体はオープンデータに取り組むことが義務付けられました。オープンデータへの取組により、国民参加・官民協働の推進を通じた諸課題の解決、経済活性化、行政の高度化・効率化等が期待されています。”(デジタル庁)
 - 参考URL→https://www.digital.go.jp/resources/open_data/

国の行政機関・地方公共団体など≡公的機関にはどんなところがある？

国内外政府の省庁や国際機関などが該当します

- 国(日本)の行政機関
 - 省庁。主には、内閣府、総務省、法務省、外務省、財務省、文部科学省、厚生労働省、農林水産省、経済産業省、国土交通省、環境省、防衛省、デジタル庁、復興庁
- 日本の地方公共団体
 - 都道府県、市町村、特別区(東京23区)など
- 外国政府の行政機関等
 - アメリカ合衆国連邦政府、イギリス政府など
- 国際機関
 - 国際連合(UN)、経済協力開発機構(OECD)、欧州連合(EU)など
- その他、「公的」に近そうなもの
 - 認可法人(日本銀行など)、社団法人(投資信託協会など)⋯⋯など

公的機関が作成している公的統計

- 公的統計にはどんなものがあるの？
- 統計データはどこで取得できるの？
- どうやって取得できるの？



世界と米国の公的データ

- はじめに
- 公的統計とオープンデータ
- 世界と米国の公的データ
- pandas-datareaderでデータ取得
- 日本の公的データ
- 公的統計APIとe-Stat
- Pythonでのe-Statデータの扱い方
- 事例:家計調査のデータ活用
- 公的統計のこれから
- おわりに

世界の公的データ～国際連合～

サイト名	政府機関	サイトの説明	CSV等 ダウンロード	API	URL
UNData	United Nations (国際連合)	国連の統計データベースを単一のエントリーポイントを通して簡単に利用できるインターネット上のデータサービス	あり	あり	https://data.un.org/
Data Portal Population Division	United Nations (国際連合)	世界の人口統計指標へのインラクティブなアクセスができるサイト	あり	あり	https://population.un.org/dataportal/about/dataapi
UN Comtrade	United Nations (国際連合)	世界の貿易統計を製品別、貿易相手国別に年間および月次で詳細に集計	あり	あり	https://comtrade.un.org/data/dev/portal/
IMF DATA	IMF (国際通貨基金)	マクロ経済と金融データへのアクセス	あり	あり	https://data.imf.org/?sk=_388dfa60-1d26-4ade-b505-a05a558d9a42
World Bank Open Data	World Bank (世界銀行)	グローバルな開発データへの自由でオープンなアクセス	あり	あり	https://data.worldbank.org/

世界の公的データ～国連以外～

サイト名	政府機関	サイトの説明	CSV等 ダウンロード	API	URL
OECD Data	OECD (経済協力 開発機構)	OECD 加盟国および一部の非加盟国のデータを網羅した最も利用されているデータセットにアクセス	あり	あり	https://data.oecd.org/api/
eurostat	EU (欧州連合) 統計局	EU加盟国の国家統計機関およびその他の国家当局との協力のもと、欧州の統計を作成	あり	あり	https://ec.europa.eu/eurostat/web/main/data/database

アメリカ合衆国の公的データ～統計局～

サイト名	政府機関	サイトの説明	CSV等 ダウンロード	API	URL
Commerce Data Hub	ESA (経済統計局)	商務省およびその局やオフィスが維持する公開データの概要を提供	あり	あり	https://data.commerce.gov/
United States Census Bureau	USCB (国勢調査局)	米国の主要統計への一般市民のアクセス	あり	あり	https://www.census.gov/data/developers.html
Bureau of Economic Analysis	BEA (経済分析局)	業界標準の方法と手順を用いて、BEAの公表済み経済統計へのプログラムによるアクセスを提供	あり	あり	https://apps.bea.gov/API/signup/index.cfm
Bureau of Labor Statistics	BLS (労働統計局)	公的データAPIにより、一般市民はBLSの全プログラムの経済データにアクセス可能	あり	あり	https://www.bls.gov/developers/home.htm

アメリカ合衆国の公的データ～統計局以外～

サイト名	政府機関	サイトの説明	CSV等 ダウンロード	API	URL
Data.gov	GSA (共通役務庁)	米国政府のオープンデータのホームページ	あり	あり	https://data.gov/developers/apis/
FRED (Federal Reserve Economic Data)	FRB of St. Louis (セントルイス連邦準備銀行)	セントルイス連邦準備銀行の経済研究部門が提供するFRED®およびALFRED®ウェブサイトから経済データを取得するアプリケーションを構築できるようにするウェブサービス	あり	あり	https://fred.stlouisfed.org/docs/api/fred/

どうやってデータ取得する？ pandas-datareaderで オープンデータ取得

- はじめに
- 公的統計とオープンデータ
- 世界と米国の公的データ
- pandas-datareaderでデータ取得
- 日本の公的データ
- 公的統計APIとe-Stat
- Pythonでのe-Statデータの扱い方
- 事例:家計調査のデータ活用
- 公的統計のこれから
- おわりに

pandas-datareaderでデータ取得

外国政府や国際機関の一部のデータは pandas-datareaderでさくっと取得できる

- pandas-datareader
 - 様々な公的データや市場データを pandasのデータフレーム形式で取得できるライブラリ
 - ドキュメント
 - <https://pandas-datareader.readthedocs.io/en/latest/>
 - GitHub
 - <https://github.com/pydata/pandas-datareader>
- 取得できるデータの例
 - World Bank(世界銀行)
 - OECD(経済協力開発機構)
 - Eurostat(EU統計局)
 - FRED:セントルイス連邦準備銀行のデータ

pandas-datareaderでデータ取得

```
import pandas_datareader as pdr  
pdr.get_data_fred('GS10').tail(10) # 10年満期米国債市場金利
```

GS10	
DATE	
2021-11-01	1.56
2021-12-01	1.47
2022-01-01	1.76
2022-02-01	1.93
2022-03-01	2.13
2022-04-01	2.75
2022-05-01	2.90
2022-06-01	3.14
2022-07-01	2.90
2022-08-01	2.90



Fredから金利
データを取得

- 使い方は簡単！
- インポートして、1行で取得でき、
- pandasのDataFrame or Series型で扱える

```
import pandas_datareader.data as web  
web.DataReader('TUD', 'oecd') # 労働組合データ
```

Country	Australia	Austria	Belgium	Canada	Czech Republic	Denmark
Frequency	Annual	Annual	Annual	Annual	Annual	Annual
Measure	Percentage of employees					
Time						
2018-01-01	NaN	26.700001	50.700001	26.299999	11.7	66.699997
2019-01-01	NaN	26.299999	49.099998	26.100000	NaN	67.000000
2020-01-01	NaN	NaN	NaN	27.200001	NaN	NaN

3 rows × 39 columns

OECDから各国の
労働組合データを取得

【参考】世界のオープンデータの枠組み

- SDMX(Statistical Data and Metadata eXchange)
 - 國際機関とその加盟国との間で統計データとメタデータを交換するメカニズムとプロセスを標準化および近代化することを目的とした国際的な取り組み
 - 参考URL→<https://sdmx.org/>
 - ライブラリ→<https://pandasdmx.readthedocs.io/en/v1.0/>
- Comprehensive Knowledge Archive Network
 - オープンデータを保存および配布するためのオープンソースのオープンデータポータル
 - 参考URL→<https://ckan.org/>
 - 参考ドキュメント→<http://docs.ckan.org/en/2.9/>
- その他のサイト
 - Google Public Data
 - 参考URL→<https://www.google.com/publicdata/directory>

日本の公的データ

- はじめに
- 公的統計とオープンデータ
- 世界と米国の公的データ
- pandas-datareaderでデータ取得
- 日本の公的データ
- 公的統計APIとe-Stat
- Pythonでのe-Statデータの扱い方
- 事例:家計調査のデータ活用
- 公的統計のこれから
- おわりに

日本の公的データ～統計局等～

サイト名	政府機関	サイトの説明	CSV等 ダウンロード	API	URL
e-Stat	総務省 統計局	政府統計の総合窓口。日本の統計が閲覧できる政府統計ポータルサイト	あり	あり	https://www.e-stat.go.jp/
miripo	総務省 統計局	委託を受けて新たな集計表を作成して提供するサービスや、調査対象の秘密の保護を図った上で、集計していない個票形式のデータを提供するサービス	詳細は サイト参照	なし	https://www.e-stat.go.jp/microdata/
RESAS	内閣府 & 経済産業省	産業構造や人口動態、人の流れなどの官民ビッグデータを集約し、可視化するシステム	あり	あり	https://resas.go.jp/#/13/13101

日本の公的データ～デジタル庁～

サイト名	政府機関	サイトの説明	CSV等 ダウンロード	API	URL
DATA GO JP データカタログ サイト	デジタル庁	デジタル庁が整備、運営するオープンデータに係る情報ポータルサイト	あり	あり	https://www.data.go.jp/
e-Govポータル	デジタル庁	政府サイトのリンク集	なし	なし	https://www.e-gov.go.jp/about-government/statistics.html

日本の公的データ～地理情報～

サイト名	政府機関	サイトの説明	CSV等 ダウンロード	API	URL
GIS ホームページ	国土交通省	地理情報システム	あり	なし	https://www.jma.go.jp/jma/menu/arc_data.html
G空間情報セン ター	社会基盤情報 流通推進協議会 (国土交通省と連 携)	地理空間情報の有効活 用と流通促進を図るデー タ流通支援プラットフォー ム	あり	なし	https://front.geospatial.jp/
土地総合情報 システム	国土交通省	不動産の取引価格、地価 公示・都道府県地価調査 の価格を検索してご覧に なることができる国土交 通省のWEBサイト	あり	なし	https://www.land.mlit.go.jp/webland/
気象庁	気象庁	気象庁ホームページに掲 載している過去の数値 データ	あり	なし	https://www.jma.go.jp/jma/menu/arc_data.html

日本の公的データ～経済情報～

サイト名	政府機関	サイトの説明	CSV等 ダウンロード	API	URL
日本銀行 主要統計 データ閲覧	日本銀行	日本銀行の主要統計データ閲覧サイト	あり	なし	https://www.stat-search.boj.or.jp/index.html
【参考】 J-Quants (ベータ版)	日本取引所 グループ	投資にまつわるデータ・環境を提供し、個人投資家の皆様によるデータ利活用の可能性を検証するプロジェクト	なし	あり (制限付き)	https://jpX-jquants.com/#jquants-api

* 日本取引所グループは株式会社なので、公的機関ではないですが、公共性の高い市場データを提供しており、参考情報として載せています。

日本の公的データ～法人情報～

サイト名	政府機関	サイトの説明	CSV等 ダウンロード	API	URL
gBizINFO	経済産業省 & デジタル庁	政府保有の法人情報を法人番号に紐づけてデータ整理を行い、2次利用可能なオープンデータとして情報提供するサイト	あり	あり	https://info.gbiz.go.jp/
法人番号 公表サイト	国税庁	法人番号の指定を受けた者の1.商号又は名称、2.本店又は主たる事務所の所在地、3.法人番号(基本3情報)を公表	あり	あり	https://www.houjin-bangou.nta.go.jp/
適格請求書 発行事業者 公表サイト	国税庁	適格請求書発行事業者登録を行っている事業者の情報を公表	あり	あり	https://www.invoice-kohyo.nta.go.jp/

* 2022年9月に提供の一時見合わせや再開があったため、最新の情報はサイトをご確認ください。
<https://www.invoice-kohyo.nta.go.jp/news/r04/r04news04.html>

公的統計APIとe-Stat

- はじめに
- 公的統計とオープンデータ
- 世界と米国の公的データ
- pandas-datareaderでデータ取得
- 日本の公的データ
- **公的統計APIとe-Stat**
- Pythonでのe-Statデータの扱い方
- 事例:家計調査のデータ活用
- 公的統計のこれから
- おわりに

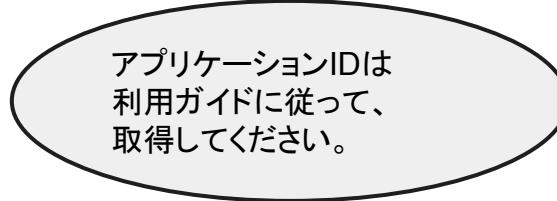
e-Stat 政府統計の総合窓口

- e-Stat
 - 日本の統計が閲覧できる政府統計ポータルサイト (総務省統計局)
 - 参考URL→<https://www.e-stat.go.jp/>
 - CSVなどでデータをダウンロードできる
 - e-Statの利用規約
 - 参考URL→<https://www.stat.go.jp/info/riyou.html>
- e-Stat API機能
 - e-Stat(政府統計の総合窓口)で提供している統計データを機械判読可能な形式で取得できる API機能
 - テーブル数で20万件程度のデータセットを提供
 - e-Stat API機能の利用規約
 - 参考URL→<https://www.e-stat.go.jp/api/agreement/>
 - 利用にはアプリケーションIDの取得が必要
- (参考)統計ダッシュボード
 - 国や民間企業等が提供している主要な統計データをグラフで提供
 - 利用登録不要なAPIを提供
 - 統計Dashboardの利用規約
 - 参考URL→<https://dashboard.e-stat.go.jp/static/terms>

【参考】e-Stat API機能

e-Stat API機能→<https://www.e-stat.go.jp/api/>

1. 利用ガイドに従って、
 - a. <https://www.e-stat.go.jp/api/api-info/api-guide>
2. 利用規約を確認し、
 - a. <https://www.e-stat.go.jp/api/agreement/>
3. ユーザー登録して、
 - a. <https://www.e-stat.go.jp/mypage/user/preregister>
4. アプリケーションIDの取得
 - a. 「マイページ」→「API機能(アプリケーションID発行)」から、必須入力項目を埋めて発行
5. API仕様を見ながら、開発！
 - a. <https://www.e-stat.go.jp/api/api-info/api-spec>
6. クレジット表示を忘れずに
 - a. <https://www.e-stat.go.jp/api/api-info/credit>



アプリケーションIDは
利用ガイドに従って、
取得してください。

クレジット表示を忘れずに

“API機能を使用したサービスを公開される場合には、下記のクレジット表示をお願いします。

「このサービスは、政府統計総合窓口(e-Stat)のAPI機能を使用していますが、サービスの内容は国によって保証されたものではありません。」

利用される方が参照できる場所であれば、表示場所の指定はございません。”

【出典】政府統計の総合窓口(e-Stat)API機能 クレジット表示
<https://www.e-stat.go.jp/api/api-info/credit>

API利用時は、サーバーへの思いやりが大切

“第8条(禁止事項)

1. 利用者は、以下に掲げる行為を行ってはならないものとします。

(1) 本機能の運用を妨害する行為

(2) 短時間における大量のアクセスその他本機能の運用に支障を与える行為

2. 統計センターは、上記に該当する行為を行っている場合又は該当する行為を行うおそれがあると判断した場合、利用者に対して、本機能の利用を停止することができるものとします。

【出典】政府統計の総合窓口(e-Stat)API機能利用規約

<https://www.e-stat.go.jp/api/agreement/>

【補足:本発表の事例でメインで利用する統計】家計調査

- 概要
 - 一定の統計上の抽出方法に基づき選定された全国約 9千世帯の方々を対象
 - 家計の収入・支出、貯蓄・負債などを毎月調査
 - 家計簿、年間収入調査票及び貯蓄等調査票は、調査世帯が記入する自計申告により、世帯票は、調査員の質問調査による
- 参考URL
 - 家計調査の統計データのページ
 - <https://www.stat.go.jp/data/kakei/>
 - 家計調査の利用規約
 - <https://www.stat.go.jp/data/kakei/inyou.html>
- 指標を読むときの注意点など
 - 著名エコノミストによると、
 - 「家計調査はきわめて詳細な統計 (1)」
 - 「エコノミストの間で非常に評判が悪く使いにくい統計として有名 (2)」
 - 「月々の振れが非常に大きい (3)」「サンプル数の少なさ(4)」「回答負担の重さから記入者の『回答者疲れ』が生じ、調査にバイアスがかかるとの指摘 (5)」

【出典】新家義貴著、『経済指標の読み方』日本経済出版社 2017

(1,2,3,4)p.187, (5)p.188

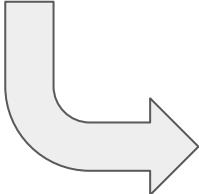
* (4)直接引用していますが、おそらく、サンプル数ではなく、サンプルサイズを意図されていると思います。

いよいよ、APIでデータ取得してみる

```
In [ ]: import requests
```

```
url = "https://api.e-stat.go.jp/rest/3.0/app/json/getStatsData?"  
params = {  
    "appId": "Your_Application_ID",  
    "statsDataId": "0003000795",  
}  
  
res = requests.get(url, params)  
out = res.json()
```

```
out
```

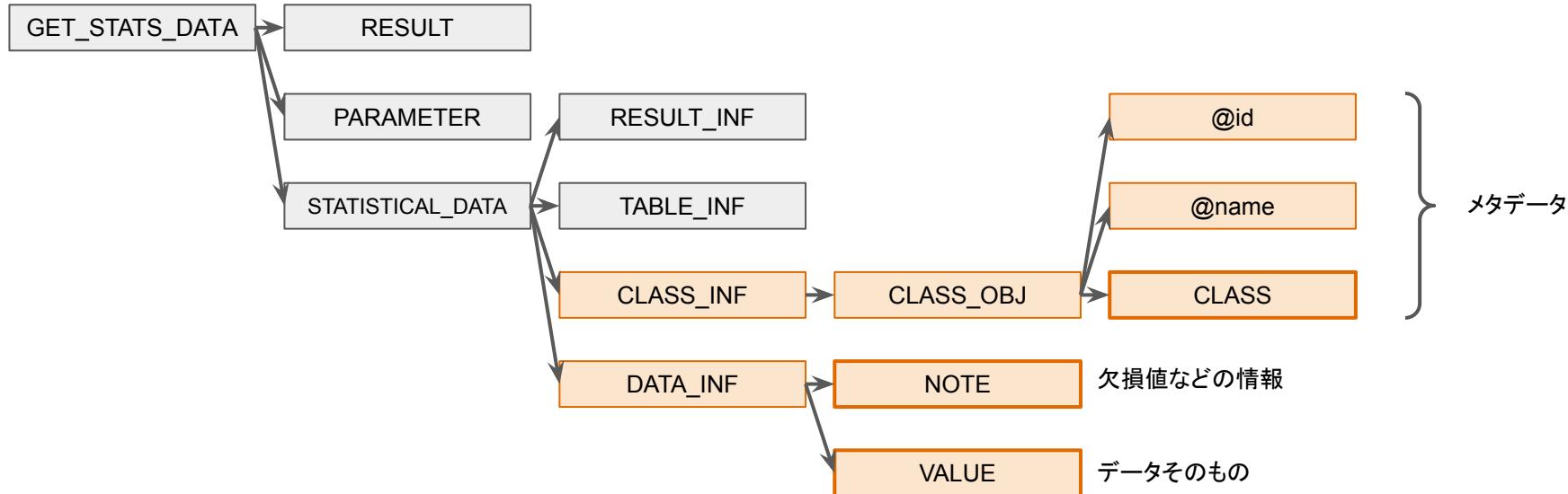


```
{'GET_STATS_DATA': {'RESULT': {'STATUS': 0,  
    'ERROR_MSG': '正常に終了しました。',  
    'DATE': '2022-09-11T17:04:21.088+09:00'},  
    'PARAMETER': {'LANG': 'J',  
        'STATS_DATA_ID': '0003000795',  
        'DATA_FORMAT': 'J',  
        'START_POSITION': 1,  
        'METAGET_FLG': 'Y'},  
    'STATISTICAL_DATA': {'RESULT_INF': {'TOTAL_NUMBER': 343578,  
        'FROM_NUMBER': 1,  
        'TO_NUMBER': 100000,  
        'NEXT_KEY': 100001},  
        'TABLE_INF': {@id: '0003000795',  
            'STAT_NAME': {@code: '00200561', '$': '家計調査'},  
            'GOV_ORG': {@code: '00200', '$': '総務省'},  
            'STATISTICS_NAME': '家計調査 家計収支編 単身世帯',  
            'TITLE': {@no: '002', '$': '用途分類 用途分類(年齢階級別)'},  
            'CYCLE': '四半期',  
            'SURVEY_DATE': 0,  
            'OPEN_DATE': 1626000000000}}
```

- requestsライブラリでデータ取得してみます。
- アプリケーションIDと統計表IDをパラメータに設定します。
- 戻り値のJSON(dict)は深い入れ子構造になっています。

JSONの深い入れ子の中身

- 戻り値の JSON(dict) は深い入れ子構造になっています。
- 入れ子構造の中身を見ていくと…



APIで取得したデータを見てみる

```
In [ ]: import pandas as pd
```

```
df = pd.DataFrame(out['GET_STATS_DATA']['STATISTICAL_DATA'][ 'DATA_INF']['VALUE'])  
df
```

	@tab	@cat01	@cat02	@cat03	@cat04	@area	@time	@unit	\$
0	01	001	21	A00	0	00000	2000000103	一万分比	10000
1	01	001	21	A00	0	00000	2000000406	一万分比	10000
2	01	001	21	A00	0	00000	2000000709	一万分比	10000
3	01	001	21	A00	0	00000	2000001012	一万分比	10000
4	01	001	21	A00	0	00000	2001000103	一万分比	10000
...
99995	01	085	22	435	1	00000	2005001012	円	13
99996	01	085	22	435	1	00000	2006000103	円	82
99997	01	085	22	435	1	00000	2006000406	円	15
99998	01	085	22	435	1	00000	2006000709	円	13
99999	01	085	22	435	1	00000	2006001012	円	10

100000 rows × 9 columns

- VALUEに格納されているメインの統計データをpandasのデータフレーム形式で読み込みます。
- VALUEだけでは、(データフレームに変換した際の)列名も項目値も数字の羅列で、それぞれの列が何を表しているのかわかりません。
- 値の列は'\$'ですが、数値でない特集文字が紛れ込んでいます。
 - '!', '**', 'X'など

悩みどころ

- 戻り値のJSON(dict)は深い入れ子構造になっている
- VALUEだけでは、それぞれの列が何を表しているのかわからず、メタデータが必要
- 値の列「\$」を見ると、数値でない欠損と思われる特殊文字が複数ある
- そもそもstatsDataId(統計表ID)の調べ方がわからない(後述)
- その他、いろいろ
 - 値の列「\$」には、金額も人数も割合もすべての値が入っている(適切に非正規化されていない)
 - 1回あたりの取得データ数上限10万件
 - など

メタデータの紐付け方

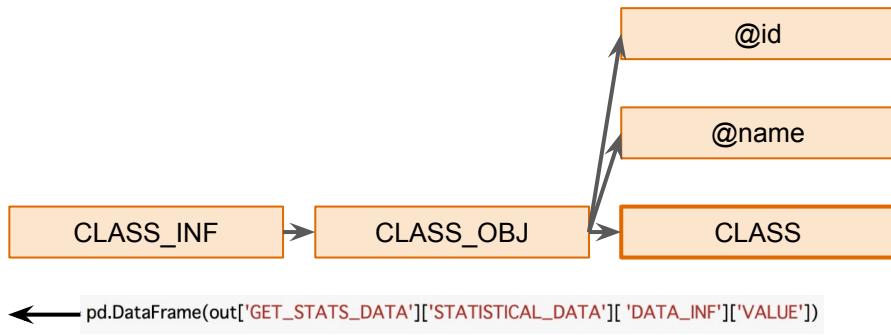
```
pd.DataFrame(out['GET_STATS_DATA']['STATISTICAL_DATA']['CLASS_INF']['CLASS_OBJ'][1]['CLASS'])
```

	@code	@name	@level	@unit	@parentCode
0	001	世帯数分布（抽出率調整）	1	一万分比	NaN
1	002	集計世帯数	1	世帯	NaN
2	009	世帯主の年齢	1	歳	NaN
3	010	有業者比率	1	%	NaN
4	011	持家率	1	%	NaN
...

	0番目	1番目							
	@tab	@cat01	@cat02	@cat03	@cat04	@area	@time	@unit	\$
0	01	001	21	A00	0	00000	2000000103	一万分比	10000
1	01	001	21	A00	0	00000	2000000406	一万分比	10000
2	01	001	21	A00	0	00000	2000000709	一万分比	10000
3	01	001	21	A00	0	00000	2000001012	一万分比	10000
4	01	001	21	A00	0	00000	2001000103	一万分比	10000

0が@tab、1が@cat01、
2が～という順で対応

- VALUEの各属性のメタデータはCLASS_INF下層に保持されており、以下があります。
 - 値のメタデータの”CLASS”
 - VALUEの列名と紐づく”@id”(ID)
 - IDの意味に相当する”@name”(名称)



```
← pd.DataFrame(out['GET_STATS_DATA']['STATISTICAL_DATA']['DATA_INF']['VALUE'])
```

メタデータの紐付け方

```
pd.DataFrame(out['GET_STATS_DATA']['STATISTICAL_DATA']['CLASS_INF']['CLASS_OBJ'][1][ 'CLASS'])
```

	@code	@name	@level	@unit	@parentCode
0	001	世帯数分布（抽出率調整）	1	一万分比	NaN
1	002	集計世帯数	1	世帯	NaN
2	009	世帯主の年齢	1	歳	NaN
3	010	有業者比率	1	%	NaN
4	011	持家率	1	%	NaN
...

	0番目	1番目							
	@tab	@cat01	@cat02	@cat03	@cat04	@area	@time	@unit	\$
0	01	001	21	A00	0	00000	2000000103	一万分比	10000
1	01	001	21	A00	0	00000	2000000406	一万分比	10000
2	01	001	21	A00	0	00000	2000000709	一万分比	10000
3	01	001	21	A00	0	00000	20000001012	一万分比	10000
4	01	001	21	A00	0	00000	20010000103	一万分比	10000

0が@tab、1が@cat01、
2が～という順で対応

- VALUEの各属性のメタデータはCLASS_INF下層に保持されており、以下があります。
 - 値のメタデータの”CLASS”
 - VALUEの列名と紐づく”@id”(ID)
 - IDの意味に相当する”@name”(名称)
- メタデータの”CLASS”は、VALUEの各分類(”@cat”)等の数だけ出力されます。
- 属性(列)としてコード(”@code”)、名称(”@name”)、階層レベル(”@level”)、単位(”@unit”)、親コード(”@parentCode”)等があります。

```
← pd.DataFrame(out['GET_STATS_DATA']['STATISTICAL_DATA'][ 'DATA_INF'][ 'VALUE'])
```

メタデータの紐付け方

```
pd.DataFrame(out['GET_STATS_DATA']['STATISTICAL_DATA']['CLASS_INF']['CLASS_OBJ'][1][ 'CLASS'])
```

	@code		@name	@level	@unit	@parentCode
0	001	世帯数分布（抽出率調整）		1	一万分比	NaN
1	002	集計世帯数		1	世帯	NaN
2	009	世帯主の年齢		1	歳	NaN
3	010	有業者比率		1	%	NaN
4	011	持家率		1	%	NaN
...

	@tab	@cat01	@cat02	@cat03	@cat04	@area	@time	@unit	\$
0	01	001	21	A00	0	00000	2000000103	一万分比	10000
1	01	001	21	A00	0	00000	2000000406	一万分比	10000
2	01	001	21	A00	0	00000	2000000709	一万分比	10000
3	01	001	21	A00	0	00000	2000001012	一万分比	10000
4	01	001	21	A00	0	00000	2001000103	一万分比	10000

VALUE

- VALUEの各属性のメタデータはCLASS_INF下層に保持されており、以下があります。
 - 値のメタデータの”CLASS”
 - VALUEの列名と紐づく”@id”(ID)
 - IDの意味に相当する”@name”(名称)
- メタデータの”CLASS”は、VALUEの各分類(”@cat”)等の数だけ出力されます。
- 属性(列)としてコード(”@code”)、名称(”@name”)、階層レベル(”@level”)、単位(”@unit”)、親コード(”@parentCode”)等があります。
- 「VALUEの”@cat01”列」と「”@id”(ID)がcat01のCLASSの”@code”列」で結合することでメタデータと紐付けることができます。

欠損の特殊文字

```
out['GET_STATS_DATA']['STATISTICAL_DATA'][ 'DATA_INF']['NOTE']
```

```
[{@char: '***', '$': '調査又は集計していないもの'},  
 {@char: '-', '$': '該当数字がないもの'},  
 {@char: 'X', '$': '数値が秘匿されているもの'}]
```

- 特殊文字はNOTEで、どのような種類があるかとその意味を調べることができます。

statsDataId(統計表ID)で欲しいデータが探せる！

- e-Statのデータベースから検索できる
 - 参考URL→<https://www.e-stat.go.jp/stat-search/database?page=1>
- 「政府統計コード」ではなく、「統計表表示ID」←間違えやすい！！

In []: `import requests`

```
url = "https://api.e-stat.go.jp/rest/3.0/app/json/getStatsData?"  
params = {  
    "appId": "Your Application ID"  
    "statsDataId": "0003000795",  
}
```



統計表ID

```
res = requests.get(url, params)  
out = res.json()  
  
out
```

さて、公的統計って簡単に使えるの？

これまで見てきたように
データを加工するのは
結構、大変…



ライブラリ作りました

```
pip install jpy-datareader
```

JPy-DataReaderでデータ取得



pipでインストールできて、

```
pip install jpy-datareader
```

- GitHub
 - <https://github.com/well-living/jpy-datareader>
- サンプルnotebook ダウンロード
 - <https://github.com/well-living/PyConJP2022>

インポートして、1行！！！

```
import jpy_datareader.data as web
df = web.DataReader("0003000795", "estat", api_key=api_key)
df
```

取得したアプリケーションID文字列

用途分類	世帯区分	世帯主の年齢階級	性別	地域区分	時間軸（四半期）	金額
0 クレジット購入借入金	(単身) 勤労者世帯	34歳以下	女	全国	2020年10～12月期	29838.0
1 クレジット購入借入金	(単身) 勤労者世帯	34歳以下	女	全国	2020年1～3月期	25476.0
2 クレジット購入借入金	(単身) 勤労者世帯	34歳以下	女	全国	2020年4～6月期	31154.0
3 クレジット購入借入金	(単身) 勤労者世帯	34歳以下	女	全国	2020年7～9月期	26843.0
4 クレジット購入借入金	(単身) 勤労者世帯	34歳以下	女	全国	2021年10～12月期	60782.0
...

JPy-DataReaderでデータ取得



- 大事なことなのでもう一度
- 一つだけ持ち帰るならこの2行→

これが



@tab	@cat01	@cat02	@cat03	@cat04	@area	@time	@unit	\$
0	01	001	21	A00	0	00000 2000000103	一万分比	10000
1	01	001	21	A00	0	00000 2000000406	一万分比	10000
2	01	001	21	A00	0	00000 2000000709	一万分比	10000
3	01	001	21	A00	0	00000 2000001012	一万分比	10000
4	01	001	21	A00	0	00000 2001000103	一万分比	10000
...
99995	01	085	22	435	1	00000 2005001012	円	13
99996	01	085	22	435	1	00000 2006000103	円	82
99997	01	085	22	435	1	00000 2006000406	円	15
99998	01	085	22	435	1	00000 2006000709	円	13
99999	01	085	22	435	1	00000 2006001012	円	10

100000 rows × 9 columns

import jpy_datareader.data as web

web.DataReader("0003000795", "estat", api_key=api_key)

これに



用途分類	世帯区分	世帯主の年齢階級	性別	地域区分	時間軸 (四半期)	金額
0 クレジット購入借入金	(単身) 勤労者世帯	34歳以下	女	全国	2020年10～12月期	29838.0
1 クレジット購入借入金	(単身) 勤労者世帯	34歳以下	女	全国	2020年1～3月期	25476.0
2 クレジット購入借入金	(単身) 勤労者世帯	34歳以下	女	全国	2020年4～6月期	31154.0
3 クレジット購入借入金	(単身) 勤労者世帯	34歳以下	女	全国	2020年7～9月期	26843.0
4 クレジット購入借入金	(単身) 勤労者世帯	34歳以下	女	全国	2021年10～12月期	60782.0
...

- GitHub

- <https://github.com/well-living/jpy-datareader>

- サンプルnotebook

- <https://github.com/well-living/PyConJP2022>

JPy-DataReaderでデータ取得



- JP(日本:ジェイ・ピー)のデータをPythonで取得するライブラリ
 - 日本円のJPYではない
 - 読み方は、ジェイ・ピー・データ・リーダー
 - (読み方に、そんなにこだわりないけど...)
- pandas-datareaderとほぼ同じ使い方！
- 簡単に使って、「悩みどころ」が全部解決！
 - 戻り値のJSON(dict)は深い入れ子構造になっている→よしなに、データフレーム化
 - VALUEだけでは、それぞれの列が何を表しているのかわからず、メタデータが必要→よしなに結合
 - 値の列「\$」を見ると、数値でない欠損と思われる特殊文字が複数ある→よしなにnan置換
 - そもそもstatsDataId(統計表ID)の調べ方がわからない→StatsListReaderで調べられる
 - その他、いろいろ→非正規化など、いろいろ加工機能

JPy-DataReaderとpandas-datareader



- pandas-datareaderとほぼ同じ使い方
 - pandas-datareaderの3種類のデータ取得の書き方とほぼ同じ

JPy-DataReader

```
import jpy_datareader as jdr  
jdr.get_data_estat_statsdata(api_key, statsDataId="0003000795")
```

```
import jpy_datareader.data as web  
web.DataReader("0003000795", "estat", api_key=api_key)
```

```
from jpy_datareader import estat  
estat.StatsDataReader(api_key, statsDataId="0003000795").read()
```

pandas-datareader

```
import pandas_datareader as pdr  
pdr.get_data_fred('GS10').tail(10) # 10年満期米国債市場金利
```

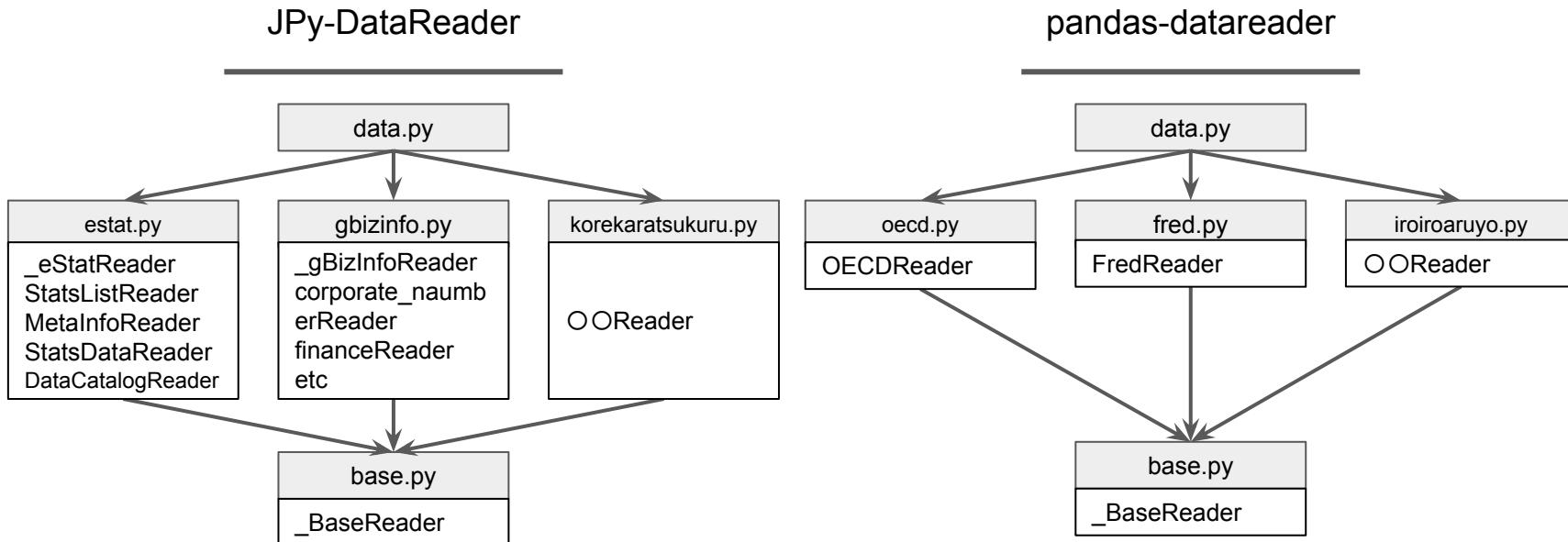
```
import pandas_datareader.data as web  
web.DataReader('TUD', 'oecd') # 労働組合データ
```

```
from pandas_datareader import fred  
fred.FredReader('GS10').read()
```

JPy-DataReaderとpandas-datareader



- pandas-datareaderとほぼ同じライブラリの構造
 - base.pyの_BaseReaderクラスを各データソースごとの API対応クラスが継承
 - data.pyのDataReader関数やget_data_estat_statsdata等がインターフェースとなる

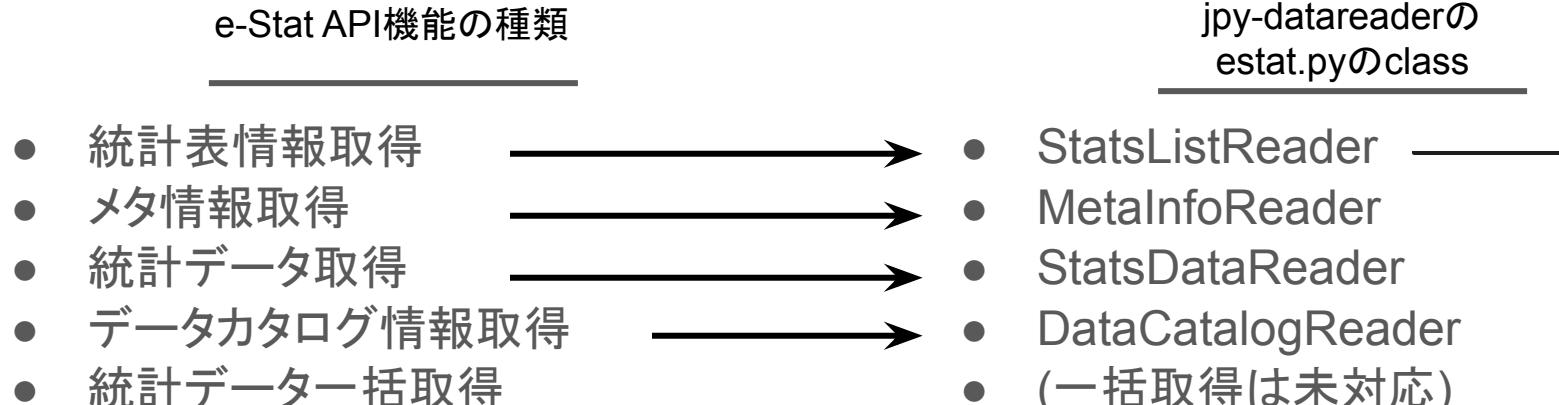


* pandas-datareaderはあまりメンテナンスされていない部分があるため、pandas-datareaderのクラスを継承する方法にはしていない

e-Stat API機能各種への対応



- e-Stat API機能の種類のうち、データ取得 (getメソッド) にほぼ対応



```
import jpy_datareader as jdr
statslist = jdr.get_data_estat_statslist(api_key)
```

* 統計表情報を取得する例
(20万件以上あるので時間かかる)

できていないこと



- e-StatとgBizINFOから作ったので、国税庁など他の公的データは未対応
 - テスト、ドキュメント、不十分な点がまだいろいろ…
 - プロポーザルで、「発表日までにライブラリ化して紹介」と書いてしまったので、採択いただいてから急ピッチで作りました&はじめてライブラリ作りました。至らぬ点が多くあると思いますが、ご了承ください
- ...

公的統計やオープンデータで 何がわかるの？ 家計調査等を見てみる

- はじめに
- 公的統計とオープンデータ
- 世界と米国の公的データ
- pandas-datareaderでデータ取得
- 日本の公的データ
- 公的統計APIとe-Stat
- Pythonでのe-Statデータの扱い方
- 事例:家計調査のデータ活用
- 公的統計のこれから
- おわりに

公的統計で何がわかる？

- 世の中について知りたいとき
- 自社(自分)と世の中との比較をしたいとき



公的統計が役に立つ

【再掲】日本の統計に関するクイズ

(1)日本の人口は約1億2千6百万人。では、15～64歳の生産年齢人口とそれ以外の人口、どちらが多い？(2021年)

人口推計(国勢調査)を使う

(2)日本の企業数は、「情報通信業」と「医療、福祉業」だとどちらが多い？(2021年)

経済センサスを使う

(3)60～69歳の平均貯蓄額は3,000万円を超えている？(2021年)

家計調査 貯蓄・負債編を使う

(4)平均的な家計で、交通費、光熱・水道費、教養娯楽費のうち、2022年1～6月と2年前の2019年1～6月を比べて、増えた支出は？

家計調査 家計収支編を使う

【統計クイズ解答】人口推計と国勢調査

- 日本的人口は約1億2千6百万人。では、15~64歳の生産年齢人口とそれ以外の人口、どちらが多い？(2021年)

```
from jpy_datareader import estat
df = estat.StatsDataReader(api_key, statsDataId="0003448228").read()
cond = df["男女別・性比名"] == "男女計"
cond &= df["人口名"] == "日本人口"
cond &= df["年齢各歳名"] != "総数"
cond &= df["時間軸(年月日現在)名"] == "2021年10月1日現在"
population = df[cond][["分類03_コード", "年齢各歳名", "値"]]
population["年齢"] = population["分類03_コード"].astype(int) - 1001
working_age_population = int(population[(population["年齢"]>=15)&(population["年齢"]<65)]["値"].sum())
print("生産年齢人口:", working_age_population, ", 生産年齢除く人口:", int(population["値"].sum() - working_age_population))
```

【出典】

- 「人口推計」(e-Stat API 統計表ID:0003448228)を加工して作成
- 各年10月1日現在人口 令和2年国勢調査基準年次

【統計クイズ解答】経済センサス

- 日本の企業数は、「情報通信業」と「医療、福祉業」だとどちらが多い？(2021年)

```
metainfo = estat.MetaInfoReader(api_key, statsDataId="0003449721", name_or_id="name")
meta = metainfo.read()
metainfo.OVERALL_TOTAL_NUMBER
```

1805760 ← データ数が多いときはメタ情報を確認して、分割して必要なデータのみ取得するのをおすすめします

```
statsdata = estat.StatsDataReader(api_key, statsDataId="0003449721",
    cdTab="202-2021", cdArea="00000",
    cdCat02="00", cdCat03="00") ← 条件を指定してデータ取得
df = statsdata.read(normal=True)
cond = df["単一・複数名"] == "総数"
cond &= (df["企業産業大分類名"] == "情報通信業") | (df["企業産業大分類名"] == "医療,福祉")
df[cond][["企業産業大分類名", "値"]]
```

【出典】

- 「経済センサス」(e-Stat API 統計表ID:0003449721)を加工して作成
- 令和3年経済センサス-活動調査速報集計 企業等に関する集計

【統計クイズ解答】家計調査 貯蓄・負債

- 60～69歳の平均貯蓄額は3,000万円を超えてる？(2021年)

```
import jpy_datareader.data as web
df = web.DataReader("0002210018", "estat", api_key=api_key)
cond = df["貯蓄・負債"] == "貯蓄"
cond &= df["世帯区分"] == "二人以上の世帯(2000年～)"
cond &= df["世帯主の年齢階級"] == "60歳以上"
cond &= df["世帯の負債有無"] == "全世帯"
cond &= df["時間軸(年次)"] == "2021年"
df[cond]
```

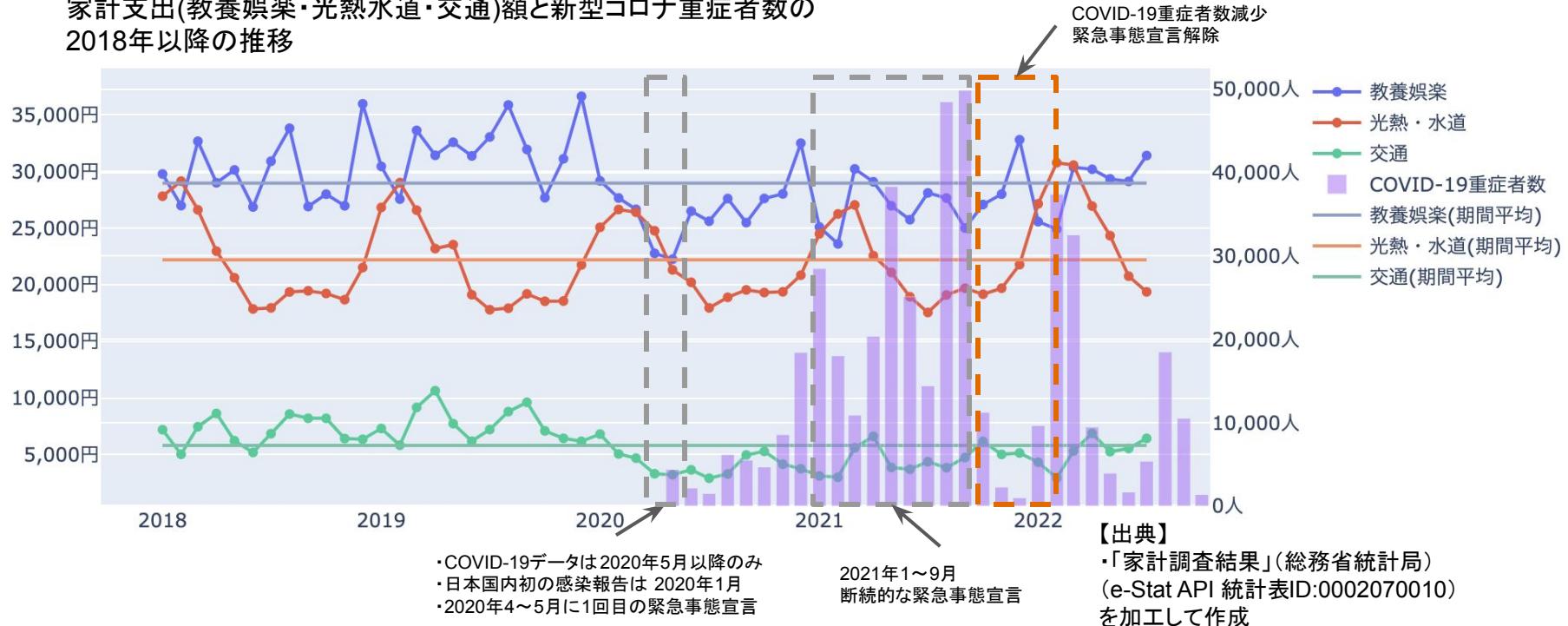
【出典】

- ・「家計調査」(e-Stat API 統計表ID:0002210018)を加工して作成
- ・家計調査 貯蓄・負債編 二人以上の世帯(世帯主の年齢階級別)

【再掲】家計調査支出と新型コロナ重傷者数の推移

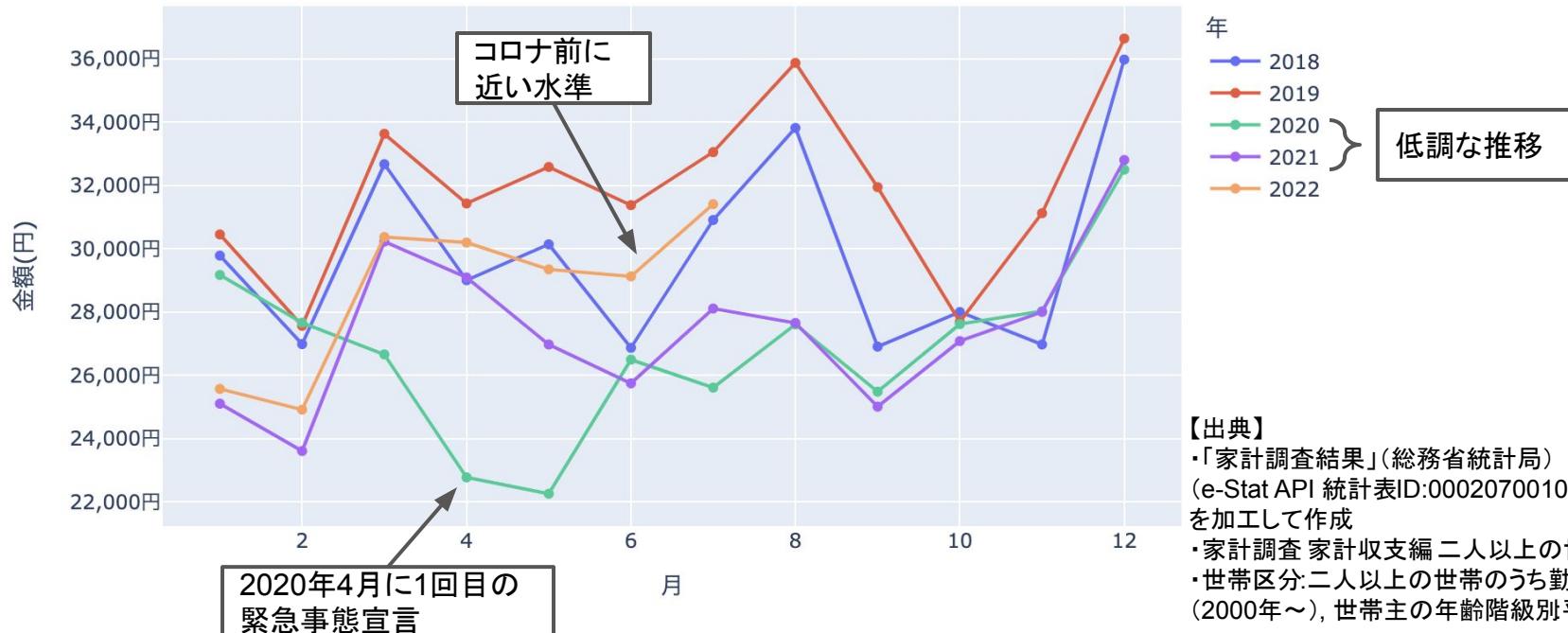
- 冒頭の例では、複数の公的データを組み合わせることで、関係性を可視化
- 時系列推移を見て、前年同月比が見にくい場合は？

家計支出(教養娯楽・光熱水道・交通)額と新型コロナ重症者数
2018年以降の推移



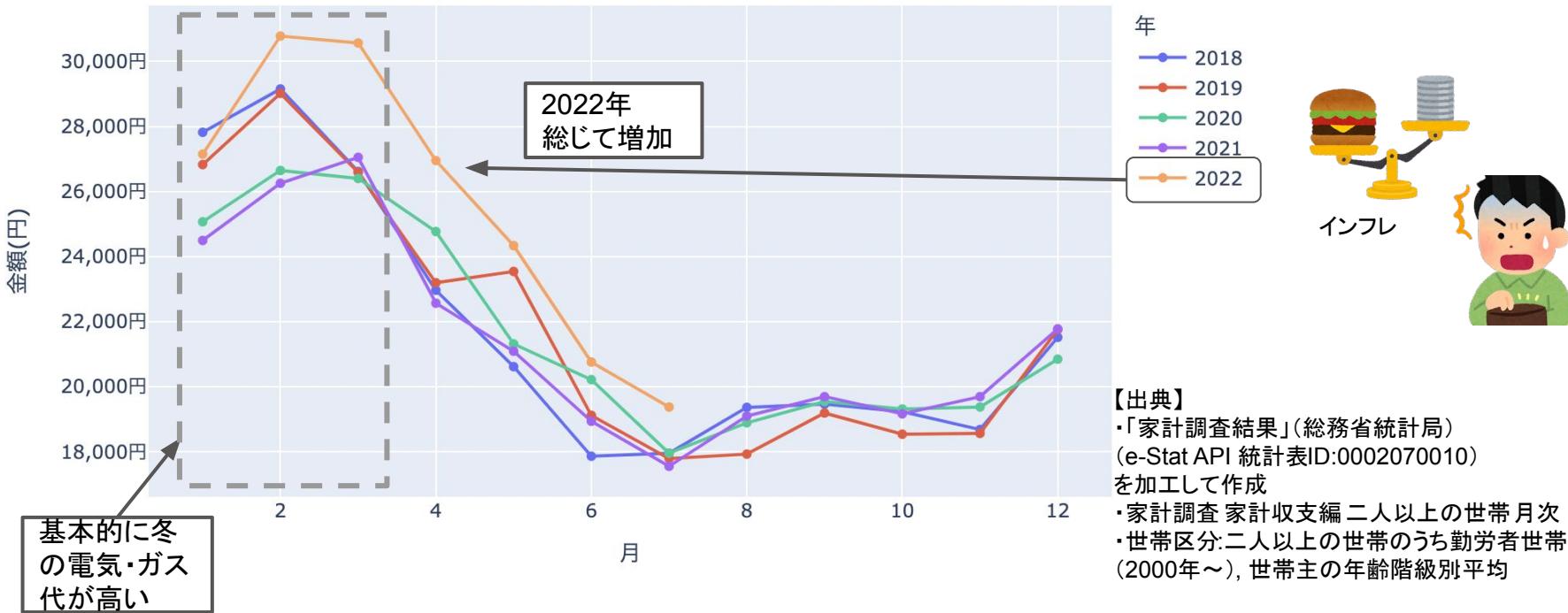
【参考】家計調査 教養娯楽費の各年同月比較

- 教養娯楽費について、各年同月比で見方を変えてみます
- コロナ禍の2020(緑線),2021(紫線)年はほとんどの月で2019年以前より支出が少なくなっています



【参考】家計調査 光熱・水道費の各年同月比較

- 光熱・水道費も見てみます。2022年(黄色線)は光熱・水道費が増加しています。
- インフレ(・原油価格上昇)など家計への影響、世の中の変化が数字で見えてきます



【補足】日本の新型コロナウイルス感染症データ

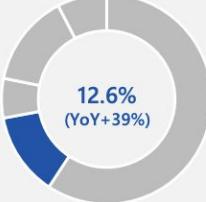
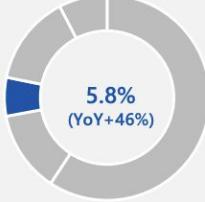
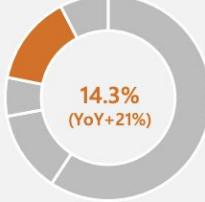
- 厚生労働省「新型コロナウイルス感染症について - オープンデータ」で取得できます
 - 参考URL→ <https://www.mhlw.go.jp/stf/covid-19/open-data.html>
- 以下のデータが取得でき、本発表では、「重症者数の推移」を用いています
 - 新規陽性者数の推移(日別)
 - 入院治療等を要する者等推移
 - 死亡者数(累積)
 - 重症者数の推移
- Pythonのコード

```
covid_severe = pd.read_csv("https://covid19.mhlw.go.jp/public/opendata/severe_cases_daily.csv", parse_dates=[ "Date" ])
covid_severe = covid_severe[["Date", "ALL"]].rename({ "ALL": "重症者数の推移"}, axis=1).set_index("Date").resample('MS').sum().reset_index()
covid_severe
```

公的統計で何がわかる？

- 世の中について知りたいとき
- 自社(自分)と世の中との比較をしたいとき

【再掲】株式会社マネーフォワードの事業

法人向け	個人向け	金融機関向け		
Money Forward Business	Money Forward Finance	Money Forward Home	Money Forward X	
<p>バックオフィス向けSaaS</p> <p>バックオフィス (経理財務・人事労務・法務・情シス) 向けクラウドサービスを提供 21万以上の課金顧客、 約半数が法人</p>  <p>59.6% (YoY+45%)</p> <p>Money Forward クラウド STREAMED Manageboard V-DNE クラウド Money Forward IT管理クラウド HITTO</p>	<p>SaaSマーケティング支援</p> <p>スマートキャンプ社が運営、 国内最大級のSaaSマーケティング プラットフォーム</p>  <p>12.6% (YoY+39%)</p> <p>BOXIL SaaS BALES BOXIL EXPO A DXL ADD SHIRU</p>	<p>ファイナンスサービス</p> <p>企業間後払い決済サービス、売掛金早期 資金化サービスを提供 MUFGと提携し、オンライン売掛金早期資 金化事業の提供</p>  <p>5.8% (YoY+46%)</p> <p>Money Forward Kessai Money Forward Early Payment SHJKJN SEJKYU</p>	<p>PFM（家計簿・資産管理） サービス</p> <p>1,330万以上の利用者、 国内シェアNo.1¹ 38万人以上の課金ユーザー 約18兆円²の連携資産残高</p>  <p>14.3% (YoY+21%)</p> <p>Money Forward ME Money Forward 固定費の見直し Money Forward お金の相談</p>	<p>Fintech推進・DX支援</p> <p>85のサービスを提供（共同開発含む） アカウントアグリゲーション基盤を活用し、 個人/法人、金融機関等、Fintech企業 をつなぐ、Fintechプラットフォームを提供</p>  <p>7.8% (YoY△23%)</p> <p>Money Forward X Money Forward Fintech Platform</p>

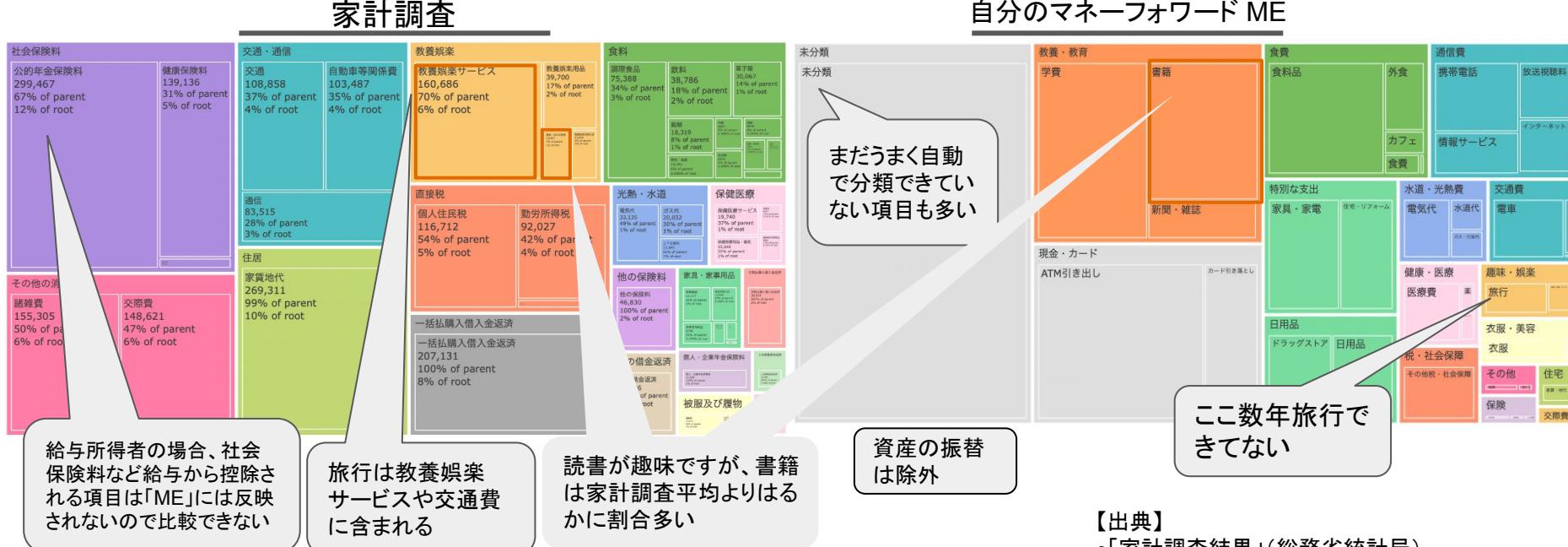
* 2022年11月期 第2四半期決算説明資料8ページ

自社(自分)と世の中との比較

- 会社の場合、
 - 新規事業を検討中だけど、市場規模はどのくらいあるだろうか？
 - 来期の戦略を立てる上で、市場全体に占める自社の法人顧客数や業種の偏りを把握したいけど、そもそも全国で業種別企業数は何社あるのだろうか？
 - →経済センサスで日本の企業全体の産業、都道府県、従業者数規模、売上規模等ごとの企業数と内訳が調べられる
 - 参考URL→<https://www.stat.go.jp/data/e-census/index.html>
- 個人の場合、
 - 自分と同年代・世帯構成の人と比べて、お金の使い方(支出)はどうなんだろう?
 - →家計調査で年齢階級別、世帯区分、勘定科目等ごとの支出金額が調べられる
 - 参考URL→<https://www.stat.go.jp/data/kakei/>

家計調査と自分の家計支出を比べてみる

- 自分のマネーフォワード MEの支出割合と家計調査をツリー・マップで比較してみました
 - さすがに、自分の支出は金額は隠して割合のみにしています



給与所得者の場合、社会保険料など給与から控除される項目は「ME」には反映されないので比較できない

旅行は教養娯楽
サービスや交通費
に含まれる

読書が趣味ですが、書籍は家計調査平均よりはるかに割合多い

資産の振替
は除外

ここ数年旅行で
きてない

【出典】

- ・「家計調査結果」(総務省統計局)
(e-Stat API 統計表ID:0003000795)
を加工して作成

【余談】家計に関する(個人的な)考え方

お金を使っていることには、自分が大切にしていることが表れていると思います。

→お金のデータを見て、比べることで、自分が大切にしていることが見えてくる

そうでないなら、お金の使い方を変えていく必要があります。

→お金のデータを見ることで、何を大切にしたいか問い合わせができる

ちなみに



- マネーフォワード MEのプレミアム会員は家計簿データをCSV形式でダウンロードができます
- 個人向けAPIはありません。家計調査と比較するツリーマップもないです
- ほしい方がいれば、自分用に作ったマネーフォワード MEのCSVを可視化するコードによければ、共有できるように整理します

家計簿データの出力 (Excel、CSV形式対応)

※ご利用のサービスによって「書き出し」「エクスポート」「出力」「ダウンロード」等名称が異なります。

ダウンロード

データ取得は簡単になった！活用レベル上げたい！そのために…

- もっと事例がほしい！
- いろんな統計データや他の社内外データを組み合わせて使いたい！
- 公的統計の今後の取り組みはどこでキャッチアップすればよいの？



総務省統計局の方に
公的統計の
推し(アピールポイント)
を聞いてきました

公的統計のアピールポイントとこれから

- はじめに
- 公的統計とオープンデータ
- 世界と米国の公的データ
- pandas-datareaderでデータ取得
- 日本の公的データ
- 公的統計APIとe-Stat
- Pythonでのe-Statデータの扱い方
- 事例:家計調査のデータ活用
- 公的統計のこれから
- おわりに

アピールポイントとこれからの取り組み

- e-Stat 統計LOD
- 公的統計とビッグデータの補完
- 統計データ・APIの活用事例

【簡易】e-Stat 統計LOD(Linked Open Data)

- Linked Open Data(LOD: リンクト・オープン・データ)
 - 簡潔に言うと、「他へのリンクを入れたデータ」
- e-Stat 統計LOD
 - オープンデータの(5段階のうち)最高ランクであるLODで統計データを(一部)提供
 - 参考URL→ <http://data.e-stat.go.jp/lodw/>

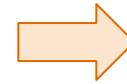
“各省庁がネット上で公開する統計を機械判読可能にするために、データの表記方法を統一させます。「政府統計の総合窓口(e-Stat)」で本日から12月1日までの間、表記方法案に関する意見照会を行います。研究者をはじめ、皆様のご意見をお待ちしています。”

2段階目

コンピュータで
処理可能なデータ

まずはこれが大事

引用参考 URL↓
<https://twitter.com/konotarogomame/status/1331478653977325569>



5段階目

他へのリンクを
入れたデータ

よりオープンに、
より横断的に、
使いやすいデータ
を目指されている！

【詳細】e-Stat 統計LOD(Linked Open Data)

● オープンデータ【再掲】

- “国、地方公共団体及び事業者が保有する官民データのうち、国民誰もがインターネット等を通じて容易に利用（加工、編集、再配布等）できるよう、次のいずれの項目にも該当する形で公開されたデータ（総務省）
 - “営利目的、非営利目的を問わず二次利用可能なルールが適用されたもの
 - “機械判読に適したもの”
 - “無償で利用できるもの”
 - 参考URL→https://www.soumu.go.jp/menu_seisaku/ictseisaku/ictriyu/opendata/
- “官民データ活用推進基本法（平成28年法律第103号）において、国及び地方公共団体はオープンデータに取り組むことが義務付けられました。オープンデータへの取組により、国民参加・官民協働の推進を通じた諸課題の解決、経済活性化、行政の高度化・効率化等が期待されています。”（デジタル庁）
 - 参考URL→https://www.digital.go.jp/resources/open_data/

● Linked Open Data(LOD: リンクト・オープン・データ)

- ティム・バーナーズ=リーによる定義
 - “あらゆるデータの識別子としてURIを使用する。”
 - “識別子には（URNや他のスキームではなく）HTTP URIを使用し、参照やアクセスを可能にする。”
 - “URIにアクセスされた際には有用な情報を標準的なフォーマットRDFなど）で提供する。”
 - “データには他の情報源における関連情報へのリンクを含め、ウェブ上の情報発見を支援する。”
 - 参考URL→<https://5stardata.info/ja/>
- 簡潔に言うと、「他へのリンクを入れたデータ」
- データ操作にはSPARQLを用いる
 - SPARQLはRDF(Resource Description Framework)問い合わせ言語
 - ライブラリ→<https://sparqlwrapper.readthedocs.io/en/latest/index.html>

【詳細】e-Stat 統計LOD(Linked Open Data)

- オープンデータには、5つの段階がある(下表:5つ星オープンデータ)
 - 参考URL→<https://5stardata.info/ja/>
- e-Stat 統計LOD
 - オープンデータの最高ランク(5スター)であるLOD(Linked Open Data)で統計データを提供
 - 参考URL→<http://data.e-stat.go.jp/lodw/>

段階		公開の状態	5つ星オープンデータのサイト例示	データ形式の例
★	1段階	オープンライセンスでデータを公開	どんな形式でも良いのであなたのデータをオープンライセンスでWeb上に公開しましょう	PDF,JPG
★★	2段階	コンピュータで処理可能なデータを公開	(データを構造化データとして公開しましょう(例: 表のスキャン画像よりもExcel))	XLS,DOC
★★★	3段階	オープンに利用できるフォーマットでデータを開	非独占の形式を使いましょう(例: ExcelよりもCSV)	XML,CSV
★★★★	4段階	Web標準(RDF等)のフォーマットでデータを開	物事を示すのにURIを使いましょう、そうすることで他の人々があなたのデータにリンクすることができます	RDF
★★★★★	5段階	他へのリンクを入れたデータ(LOD)を開	あなたのデータのコンテキストを提供するために他のデータへリンクしましょう	Linked RDF

【参考】統計表における機械判読可能なデータ

- 統計局にて、統計表における機械判読可能なデータの表記方法の統一ルールの策定
 - 参考URL→https://www.soumu.go.jp/menu_news/s-news/01toukatsu01_02000186.html
 - 参考PDF→https://www.soumu.go.jp/main_content/000723626.pdf

アピールポイントとこれからの取り組み

- e-Stat 統計LOD
- 公的統計とビッグデータの補完
- 統計データ・APIの活用事例

【簡易】公的統計とビッグデータ

- ビッグデータとは、「**多種多量の民間企業が保有するデータ**」(公的統計の整備に関する基本的な計画)
 - “既存の公的統計に比べはるかに早い時点で入手可能となる”
 - “従来の手法では取得できなかった大量のデータが利用可能となる”
 - “ビッグデータはデータ生成方法や生成プロセス等に起因する偏りや特徴を有している”



公的統計とビッグデータは、互いに補完しあうことにより、
「高め合う」関係を構築することが可能

【出典】ビッグデータ等の利活用推進に関する産官学協議のための連携会議
ビッグデータの更なる活用の方向性～政策の質の向上を目指して～

- https://www.soumu.go.jp/main_sosiki/kenkyu/02toukatsu01_04000422.html
- https://www.soumu.go.jp/main_content/000817238.pdf

【詳細】公的統計とビッグデータ

公的統計とビッグデータの定義

- 公的統計【再掲】
 - “国の行政機関・地方公共団体などが作成する統計を言います。統計調査により作成される統計(調査統計)のほか、業務データを集計することにより作成される統計(いわゆる「業務統計」)や他の統計を加工することにより作成される統計(加工統計)についても公的統計に該当します。”(総務省)
 - 参考URL→https://www.soumu.go.jp/toukei_toukatsu/index/seido/1-1n.htm
- ビッグデータ
 - “ICTの進展により生成・収集・蓄積等が可能・容易になる多種多量の民間企業が保有するデータ”(公的統計の整備に関する基本的な計画<平成30年3月閣議決定>における定義)
 - 参考URL→https://www.soumu.go.jp/main_content/000554053.pdf

公的統計とビッグデータ活用の今後の取り組み

- 公的統計とビッグデータ活用の今後の取り組みは以下でウォッチできます
 - 統計改革推進会議(総務省)
 - 参考URL→<https://www.kantei.go.jp/jp/singi/toukeikaikaku/index.html>
 - ビッグデータ等の利活用推進に関する産官学協議のための連携会議(総務省)
 - 参考URL→https://www.soumu.go.jp/main_sosiki/kenkyu/big_data/index.html

アピールポイントとこれからの取り組み

- e-Stat 統計LOD
- 公的統計とビッグデータの補完
- 統計データ・APIの活用事例

事例・コンテスト

- 統計データ活用事例
 - 統計データ利活用センターのData StaRt(データ・スタート)
 - 参考URL→<https://www.stat.go.jp/dstart/>
 - e-StatのAPI機能の活用事例
 - 参考URL→<https://www.e-stat.go.jp/api/api-dev/case>
 - 過去には、内閣府主催のコンテストも

この発表きっかけに
アイデアソン等やっていきたい！！

おわりに

- はじめに
- 公的統計とオープンデータ
- 世界と米国の公的データ
- pandas-datareaderでデータ取得
- 日本の公的データ
- 公的統計APIとe-Stat
- Pythonでのe-Statデータの扱い方
- 事例:家計調査のデータ活用
- 公的統計のこれから
- おわりに

ご清聴ありがとうございました



- よかつたら、GitHubにスター頂けると嬉しいです m(_ _)m
 - 一緒にアップデートしてくださる方も歓迎！
 - <https://github.com/well-living/jpy-datareader>
- 公的統計を使ったアイデアソンなど面白い事例を作りたい！
 - 統計局や地方公共団体に「こんなことしませんか？」といった提案もしたい
- 家計調査や公的統計の分析例あまり載せれなかつたので、SNSで発信します！

```
pip install jpy-datareader
```

- SNS等
 - note: https://note.com/well_living_ry
 - zenn: Ryo YOSHI@wellliving
 - Qiita: https://qiita.com/well_living
 - Twitter: Ryo YOSHI@well_living_ry
 - GitHub: WellLiving@well-living

參考資料

参考資料

- 政府統計の総合窓口 (e-Stat)API機能 クレジット表示
 - <https://www.e-stat.go.jp/api/api-info/credit>
- 政府統計の総合窓口 (e-Stat)API機能利用規約
 - <https://www.e-stat.go.jp/api/agreement/>
- 「家計調査結果」(総務省統計局)(e-Stat API 統計表ID:0002070010)を加工して作成(家計調査 家計収支編 二人以上の世帯)
- 「家計調査結果」(総務省統計局)(e-Stat API 統計表ID:0003000795)を加工して作成(家計調査 家計収支編 単身世帯)
- 「家計調査結果」(総務省統計局)(e-Stat API 統計表ID:0002210018)を加工して作成(家計調査 貯蓄・負債編 二人以上の世帯)
- 「人口推計」(e-Stat API 統計表ID:0003448228)を加工して作成(各年 10月1日現在人口 令和2年国勢調査基準 年次)
- 「経済センサス」(e-Stat API 統計表ID:0003449721)を加工して作成(令和3年経済センサス-活動調査 速報集計 企業等に関する集計)
- 厚生労働省「新型コロナウイルス感染症について - オープンデータ」
 - <https://www.mhlw.go.jp/stf/covid-19/open-data.html>
- 新家義貴著,『経済指標の読み方』,日本経済出版社,2017
- @driller・小川 英幸・古木 友子,『Python インタラクティブ・データビジュアライゼーション入門 —Plotly/Dashによるデータ可視化とWebアプリ構築—』,朝倉書店,2020