Medicaid.js, a FAIR Approach to Real Time Drug Analytics

Kunaal Agarwal,[1*] Hae Rin Kim[1*]

[1]Division of Cancer Epidemiology and Genetics, National Cancer Institute

*Joint first authors

Abstract

- As prescription drug prices have drastically risen over the past decade, so has the need for real-time drug tracking resources.
- Increased transparency for drug pricing and utilization is essential for informing public policy and consumers alike.
- Nevertheless, drug metric resources remain concealed behind intricate nomenclature and complex categorical patterns.
- Some web applications, such as GoodRX, allow patients to compare real-time drug prices across different pharmacies.
- However, scarcity of GoodRx API keys resulted in limited production of open-source tools and minimal increase of metric data transparency.
- Medicaid.js streamlines the process of accessing drug metric data. It combines three of the largest drug data repositories (data.medicaid.gov openData API, the Food and Drug Administration openFDA API, rxNorm API) into a serverless Software Development Kit (SDK) governed by foundational FAIR principles and data provenance.

Introduction

- FAIR principles (Findable, Accessible, Interoperable, and Reusable) were designed to ensure seamless distribution of data between machines (Wilkinson et al., 2019).
- Findability refers to ease of locating a digital resource and its metadata (Jacobsen et al., 2020). Metadata allows for data provenance. Accessibility and interoperability outline requirement for placing strict protocols regarding data access and retrieval (Jacobsen et al., 2020).
- Reusability dictates ability of media to regenerate specific data (Jacobsen et al., 2020).
- Development of FAIR digital media requires usage of data commons or cloud-computing infrastructures built as interoperable resources (Grossman 2016).
    - Data must be accessed through stateless Application Programming Interfaces (API)
- Medicaid.js contributes to the existing Data.Medicaid API and directly abstracts Data.medicaid built-in methods to scalable computational workflows.
    - Automating data processing tasks such as cleansing, normalization, and extraction through standardized computational workflows increases reusability and FAIRness (Gobble et al., 2020).
- Another FAIR aspect of Medicaid.js: its scalability as a client-side software development kit (SDK). The serverless nature of this approach decreases the computational needs.
    - Allows for the creation of serverless Web APIs, the gold standard when interacting with distributed data resources in real-time (Almeida et al., 2019).

Implementation

- Abstraction of data retrieval methods contained within Data.Medicaid API, integrated data from API data sources, hosted complete analytical workflows
- Categories of workflows: National Average Drug Acquisition Cost (NADAC), State Drug Utilization, Healthcare Quality Measures, and miscellaneous datasets
- Data retrieval through HTTP get requests, HTTP post requests, “SQL-like” queries. This was followed by normalization and cleaning of the retrieved data.
- Dynamically imported dependencies: PlotJS, LocalForage API

Results

National Average Drug Acquisition Cost over Years

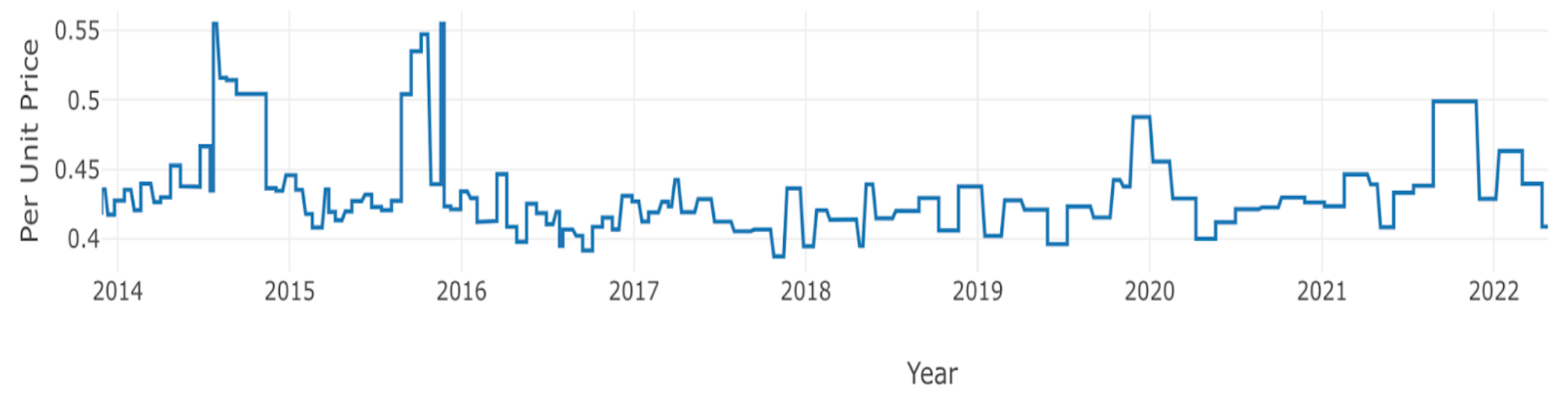

Figure 1. A time series of per unit price of HM LANSOPRAZOLE DR 15 MG CAP across all available NADAC data starting from the year 2014 to 2022.

2020 Emergency Department Visits per 1,000 Beneficiary Months: Ages 0 to 19

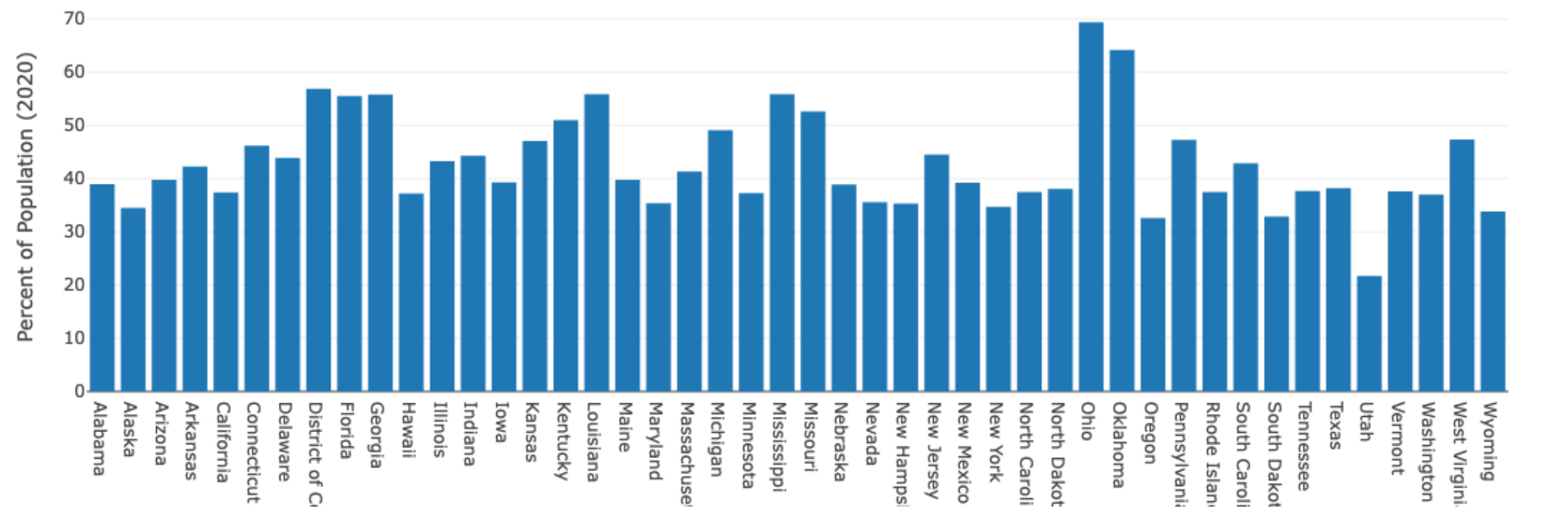

Figure 2. A bar plot for percent of population over each state from a Healthcare Quality Measures dataset named “Emergency Department Visits per 1,000 Beneficiary Months: Ages 0 to 19” in the year 2020.

2022 State Drug Utilization Across States

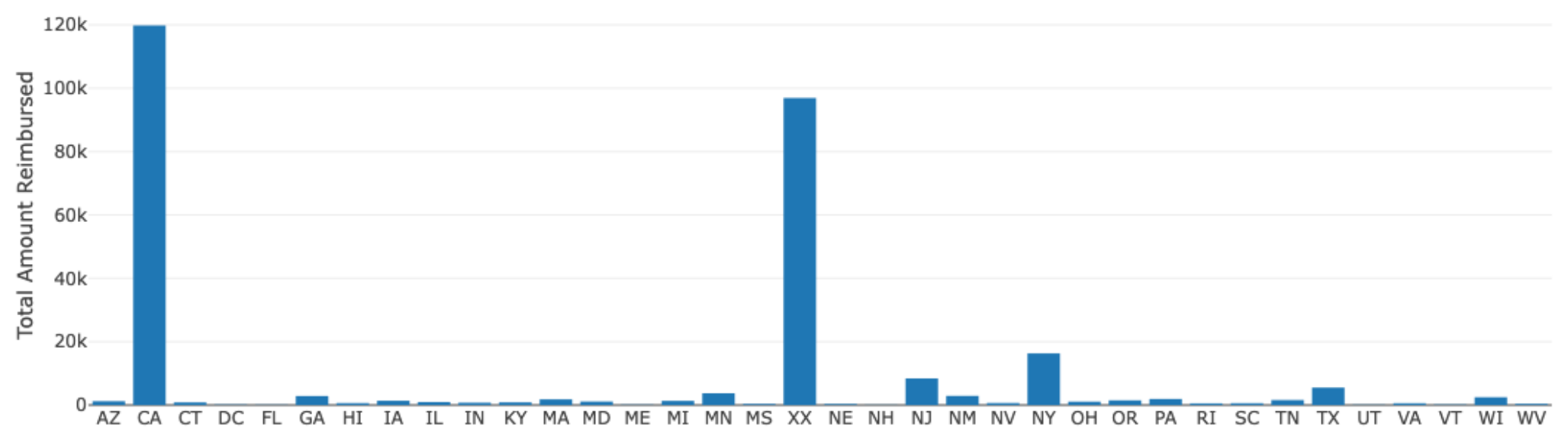

Figure 3. A bar plot of the total amount reimbursed for the national drug code 00536105556 over the states for the year 2022. This national drug code corresponds to the medication ACNE MEDICATION 10% GEL.

2022 US Total Amount Reimbursed by State

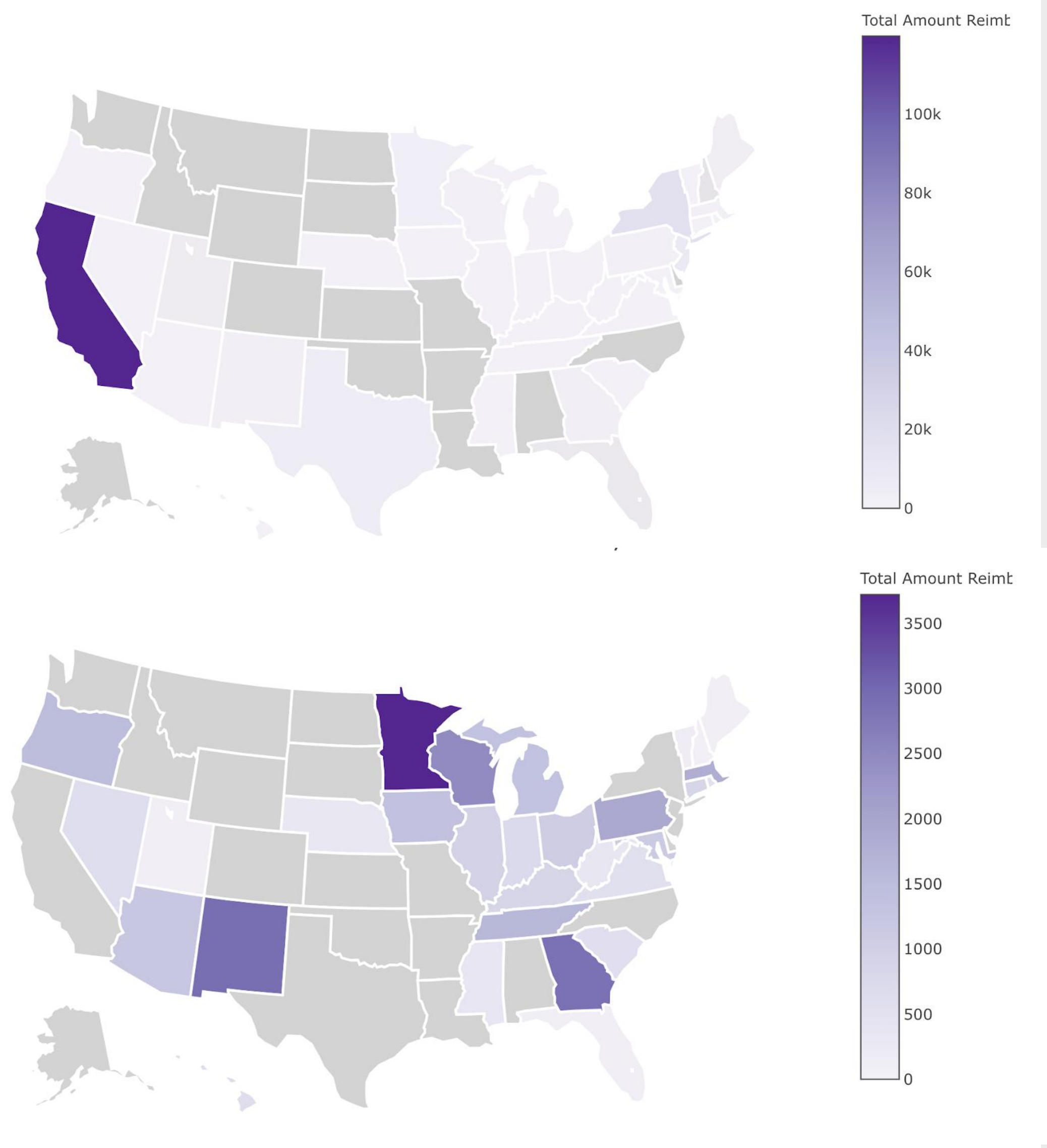

Figure 4. A comparison of the choropleth maps across the USA of the total amount reimbursed for the national drug code 00536105556 in the year 2022. This national drug code corresponds to the medication ACNE MEDICATION 10% GEL. The upper map contains outliers, and the lower map excludes outliers.

2022 Total Amount Reimbursed for Aggregated States

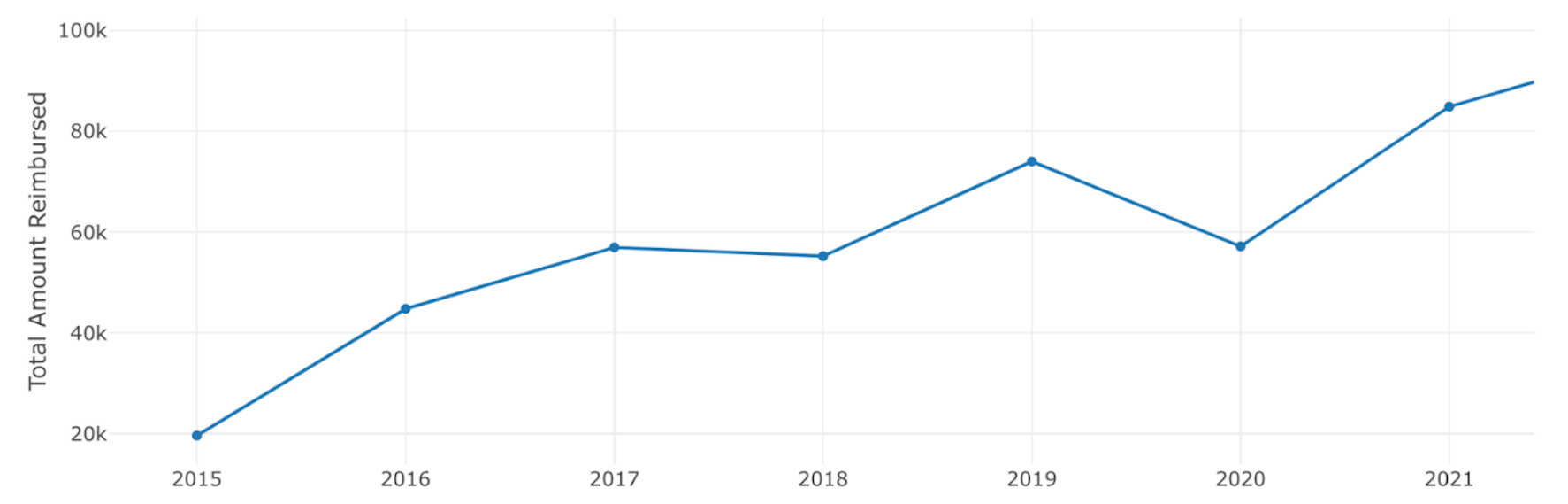

Figure 5. A time series of the total amount reimbursed for the national drug code 00536105556 for the aggregated states (e.g., national total average), represented by XX, for the year 2022. The national drug code corresponds to the medication ACNE MEDICATION 10% GEL.

ADAPALENE-BNZYL PEROX 0.1-2.5% Prices over Years

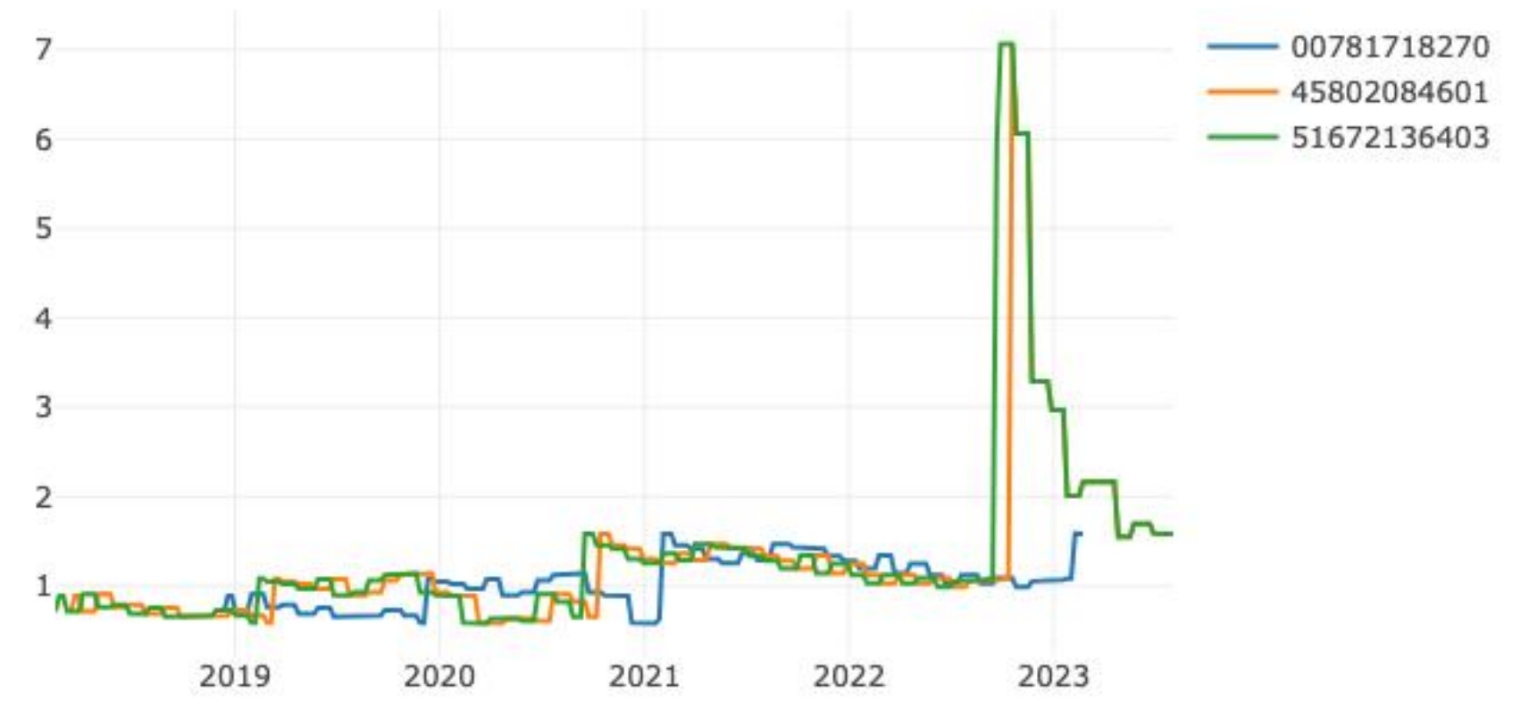

Figure 6. A time series of NADAC prices over years for three different national drug codes of ADAPALENE-BNZYL PEROX 0.1-2.5%, a specific drug associated with the disease Acne Vulgaris in RxNorm.

Discussion

The same source code, versioned by GitHub pages, serves the Medicaid.js library and live application. Plots were independently generated by distinct Medicaid.js methods. Clear trends were observed in the time series plots (Figure 1, 5, 6). State by volume plots were generated (Figure 2, 3). Outliers in the choropleth map were removed (Figure 4). As XX represented the aggregated states, its total amount reimbursed was described by a time series (Figure 5).

Conclusion

- Medicaid.js is an SDK that uses three large repositories for the analysis of drug metric data.
- In accordance with FAIR principles and data provenance, it operates on the client-side, and it allows easy access to functions that handle the data and generate plots.
- As the functions in Medicaid.js retrieve, normalize, and clean the data, the user only needs to provide input to a plotting function in order to generate time series and categorical plots through PlotJS.

References

Almeida JS et al. PeerJ 7:e6230

Annika Jacobsen et al. Data Intelligence 2020; 2 (1-2): 10–29

Grossman RL. Cancer J. 2018 May/Jun;24(3):126-130.

R. L. Grossman et al. Computing in Science & Engineering, vol. 18, no. 5, pp. 10-20, Sept.-Oct. 2016

Wilkinson, M et al. Sci Data 3, 160018 (2016).

Acknowledgments

Special thanks to Lorena Sandoval and Dr. Jonas Almeida of the Division of Cancer Epidemiology and Genetics at the National Cancer Institute.

This project has been funded in whole or in part with federal funds from the National Cancer Institute, National Institutes of Health, under contract HHSN261200800001E. The content of this publication does not necessarily reflect the views or policies of the Department of Health and Human Services, nor does mention of trade names, commercial products, or organizations imply endorsement by the U.S. Government.

The source code for Medicaid.js is available at https://github.com/KunaalAgarwal/medicaid and is available live at https://kunaalagarwal.github.io/medicaid/.

Contract HHSN261200800001E - Funded by the National Cancer Institute