

Big Data Algorithms (Fall 2021) Project Report

Vaccine Adverse Event Reporting System

(VAERS)

December 12, 2021

Doug Jih

Harica Bhogavalli Naga Lakshmi

Table of Contents

[Abstract 2](#_Toc90153734)

[Motivation 3](#_Toc90153735)

[Data Description 4](#_Toc90153736)

[Data Analysis/Algorithms 5](#_Toc90153737)

[Exploratory Data Analysis 5](#_Toc90153738)

[Frequent Itemsets & Association Rules 5](#_Toc90153739)

[Significance of Findings 6](#_Toc90153740)

[Exploratory Data Analysis 6](#_Toc90153741)

[Frequent Itemsets & Association Rules 6](#_Toc90153742)

[Conclusions 7](#_Toc90153743)

[Specific Contributions 9](#_Toc90153744)

[Doug Jih 9](#_Toc90153745)

[Harica Bhogavalli Naga Lakshmi 9](#_Toc90153746)

[Signatures of Acknowledgement 9](#_Toc90153747)

# Abstract

In this project, we applied data analysis and big data algorithms on the Vaccine Adverse Event Reporting System (VAERS) data using Python code. Our work was mainly in two parts. First, we performed exploratory data analysis, in which we explored the structure of the dataset and produced visualization of some statistical information aggregated based on factors such as state, sex, symptoms, and vaccine manufacturer. Second, we implemented extraction of frequent itemsets and association rules and examined the results. In addition, we explored using Google Cloud Platform for execution of the work. As results of the project, we presented statistical information and frequent patterns that we found.

# Motivation

The COVID-19 pandemic, a major health disaster, has impacted the world for about 2 years now. Vaccines triggering immunity against the SARS‑CoV‑2 virus that causes the COVID-19 disease started to become available near the end of 2020 to mitigate the negative health effects the virus could cause. However, no vaccine is free of side effects and although vaccines protect many people from dangerous illnesses, they can cause side effects, some which may be serious.

The Vaccine Adverse Event Reporting System (VAERS) [1] is a United States program by the Centers for Disease Control and Prevention (CDC) and Food and Drug Administration (FDA) to collect reports of adverse events that may be associated with vaccines. Medical researchers use VAERS data to study effects of vaccinations. [2]

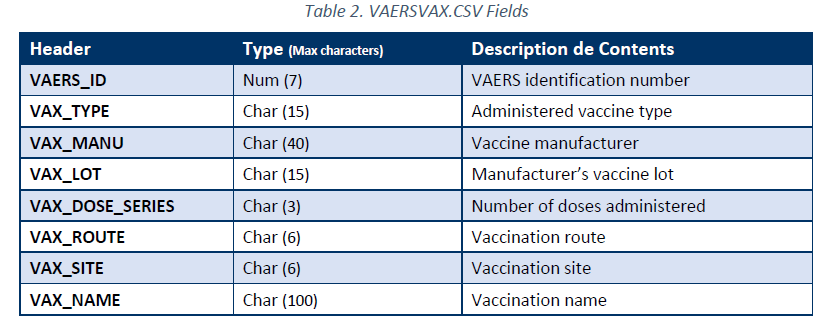
In this project, we were interested in exploration of the VAERS data and application of big data algorithms on the data. The questions we sought to answer were the following:

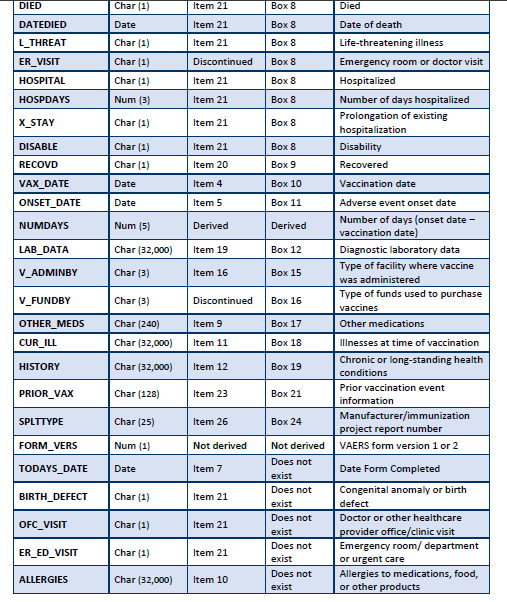
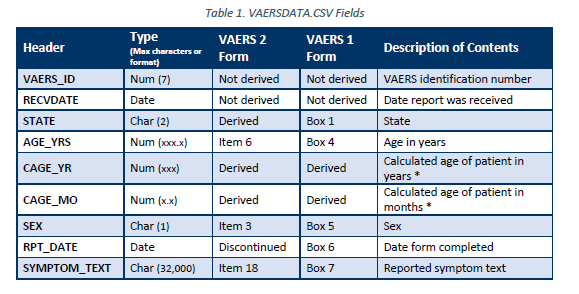
1. How could we transform the raw data and extract statistical information and patterns from the data? In specific, we were looking to aggregate and categorize various summary statistics in the data against various features in the data. We aimed to determine the number of adverse effect reports in each state of the US, for each of the COVID vaccine manufacturer and the top 5 symptoms categorized based on sex and vaccine manufacturers.
2. How could we present the information in a way that was meaningful and clear? In specific, we explored data visualization. Data visualization enables recognition of emerging trends. Such patterns make more sense when graphically represented; because visuals and diagrams make it easier for us to identify strongly correlated parameters. Humans can process visual images 60,000 times faster than text. Therefore, seeing a graph, chart, or other visual representation of data is more comfortable for the brain to process.
3. Which big data algorithm techniques could we apply to extract interesting patterns and information? We ended up experimenting with algorithms involving market basket, frequent itemsets (FP-Growth) and association rules.

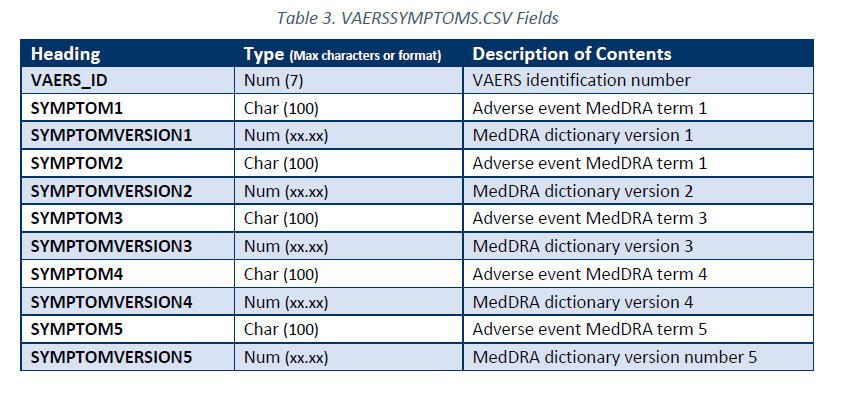
# Data Description

VAERS data is accessible by two mechanisms: by downloading raw data in comma-separated value (CSV) files for import into a database, spreadsheet, or text editing program, or by use of the CDC WONDER online search tool.

VAERS data are separated by calendar years. For this project, we used the year 2021 CSV data files, updated on November 28.

Each year’s data come in 3 CSV files. The main DATA file contains individual reports including patient details such as state, age, sex, report date, previous medical history, allergies, symptoms and so on. The VAX file provides vaccine information such as name, lot number, type. The SYMPTOMS file provides symptoms coded according to the MedDRA (Medical Dictionary for Regulatory Activities) dictionary. These three tables are correlated by the "VAERS\_ID" column as the primary/foreign key. The merged data from contains 993,374 records with 51 columns each.





# Data Analysis/Algorithms

## Exploratory Data Analysis

EDA involves generating summary statistics for numerical data in the dataset and creating various graphical representations to understand the data better. And data preprocessing is the first step of performing EDA, which transforms raw data into useful information.

This involves:

**1. Data Cleaning:**   
The data can have many irrelevant and missing parts. Data cleaning is used to handle missing data, noisy data etc.

**2. Data Transformation:**  
This step is taken in order to transform the data in appropriate forms suitable for mining process.

**3. Data Reduction:**  
While working with huge volume of data, analysis became harder. Data Reduction aims to increase the storage efficiency and reduce data storage and analysis costs.

For performing EDA in our project, we performed data cleaning where we have dropped the columns that do not contribute much to the desired results. After cleaning all the three tables, we joined the data files using an inner join.

We used the Pandas [3] package in Python for general data manipulation and analysis. We used it to loa and merge the raw data from the CSV files. We applied data aggregation, reshaping (“stacking”), filtering, and sorting algorithms to perform exploratory data analysis.

We used the Plotly [4] and Matplotlib [5] packages to produce visualization of Pandas data frames. The visualization included histograms and bar charts.

## Frequent Itemsets & Association Rules

We explored using the FP-Growth algorithms [6] for extraction of frequent itemsets. FP-Growth produces the same result as the classic Apriori [7] that extracts frequent itemsets from “baskets” of items (also referred to as a *transactional database* in the literature), but scales much better as the number of itemsets grows. FP-Growth does this by avoiding the need to generate candidate itemsets through the construction of a “frequent-pattern tree” and a divide-and-conquer approach for extracting frequent patterns from the tree. For the implementation of this algorithm, we used the MLxtend package [8], and wrote the code to prepare the VAERS data into a suitable format for the implementation.

For association rules generation, we also used the MLxtend implementation. We experimented with varying options of metrics and thresholds. The available metrics included confidence, lift, leverage, and conviction.

# Significance of Findings

## Exploratory Data Analysis

The results of our exploratory data analysis showed that the distribution of manufacturers was similar across the states of US. And majority of Adverse Events come from Pfizer. There are very limited number of adverse events reported by unknown manufacturers and it could be due to the fact that the cases were not reported via VAERS.

We found that most common side effects were headache, pyrexia, fatigue, chills, dizziness, and pain. They did not appear to vary much among different vaccines and female versus male patients. We found, however, that the most common side effects accounted for much higher proportions of all reports from female patients than male patients, at about 3-4 times as much. This suggested that for some reason, the side effects reported by male patients were more varied and concentrated less in the most common ones.

## Frequent Itemsets & Association Rules

We explored varying levels of support thresholds. We also compared the performances of the Apriori and FP-Growth implementations and found that the former was faster when the support threshold was sufficient high and the number of itemsets was sufficiently, but the latter greatly outperformed otherwise with increasing advantage as the support threshold decreases.

We explored varying metrics and thresholds in the association rules generation. We found that sorting by the leverage metric tended to produce a different perspective than sorting by the other three metrics (confidence, lift, and conviction).

Conclusions

We learned that iterative cycles of data cleaning and data analysis were necessary to further investigate the more interesting results.

We learned that cloud computing development could be highly time-consuming without pre-existing expertise.

Citations

|  |  |
| --- | --- |
| [1] | “Vaccine Adverse Event Reporting System (VAERS),” [Online]. Available: https://vaers.hhs.gov/index.html. |
| [2] | T. . T. Shimabukuro, M. Nguyen, D. Martin and F. DeStefano, “Safety monitoring in the Vaccine Adverse Event Reporting System (VAERS),” *Vaccine,* pp. 4398-4405, 8 2015. |
| [3] | “Pandas 1.3.4,” The pandas development team, 2021. [Online]. Available: https://doi.org/10.5281/zenodo.5574486. |
| [4] | “Plotly,” Plotly Technologies Inc., [Online]. Available: https://plot.ly. |
| [5] | J. D. Hunter, “Matplotlib: A 2D graphics environment,” *Computing in Science & Engineering,* vol. 9, no. 3, pp. 90--95, 2007. |
| [6] | J. Han, J. Pei, Y. Yin and R. Mao, “Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach,” *Data Mining and Knowledge Discovery,* vol. 8, no. 1, pp. 53-87, 2004. |
| [7] | “Project GitHub Source Code Repository,” [Online]. Available: https://github.com/dougjih/big-data-algorithms-group-project. |
| [8] | S. Raschka, “MLxtend: Providing machine learning and data science utilities and extensions to Python’s scientific computing stack,” *Journal of Open Source Software,* vol. 3, no. 24, p. 638, 22 4 2018. |
| [9] | R. Agrawal and R. Srikant, “Fast algorithms for mining association rules.,” in *Proc. 20th int. conf. very large data bases, VLDB.*, 1994. |

# Specific Contributions

## Doug Jih

* Assisted in preparation for the lightening talk
* Performed frequent itemsets and association rule using computer programming
* Experimented with using PySpark, Google Cloud Storage, DataProc, Dask, Google Cloud Code Repositories, Google Cloud Functions
* Co-authored and co-presented the project presentation
* Co-authored the project report

## Harica Bhogavalli Naga Lakshmi

* Authored and presented the lightening talk
* Experimented with making a web application frontend
* Experimented with using Google Cloud Storage, DataProc
* Performed exploratory data analysis using computer programming
* Co-authored and co-presented the project presentation
* Co-authored the project report

Signatures of Acknowledgement

* Doug Jih:



* Harica Bhogavalli Naga Lakshmi: