RUTGERS
UNIVERSITY | CAMDEN

Big Data Algorithms (Fall 2021) Project: Vaccine Adverse Event Reporting System (VAERS)

GROUP 7:

- DOUG JIH
- HARICA BHOGAVALLI NAGA LAKSHMI

Introduction

- The Vaccine Adverse Event Reporting System (VAERS) was created by the Food and Drug Administration (FDA) and Centers for Disease Control and Prevention (CDC) to receive reports about adverse events that may be associated with vaccines.
- Vaccines protect many people from dangerous illnesses, but vaccines, like drugs, can cause side effects, a small percentage of which may be serious.
- CDC and FDA analyze VAERS data to monitor potential safety concerns in vaccines. (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4632204/)

Dataset

- The VAERS data are in 3 types of CSV files. We use the year 2021 data last updated on November 28.
  - The main data file that contains individual reports including patient details such as state, age, sex, report date, previous medical history, allergies, symptoms and so on. – 581MB
  - The VAX file provides vaccine information such as name, lot number, type so on. – 52MB
  - The Symptoms file provides symptoms coded according to the MedDRA (Medical Dictionary for Regulatory Activities) dictionary. – 70MB
- These three tables are correlated by the "VAERS_ID" column as the primary/foreign key.
- The merged data from the 3 files has 993,374 entries with 51 columns each.

Goals

- Perform **Exploratory Data Analysis (EDA)**:
  - Uncover underlying structure
  - Extract important variables from the dataset
  - Detect outliers and anomalies (if any)
  - Test underlying assumptions

- Building **visualization** using python libraries to show important patterns depending on geographical regions and other factors.

- Experiment the application of **frequent itemset and association rule** algorithms and look for interesting patterns in the results

Algorithms - EDA

EDA involves generating summary statistics for numerical data in the dataset and creating various graphical representations to understand the data better.

In this project we have made use of the following python libraries to perform exploratory data analysis:

- pandas
- numpy
- matplotlib.pyplot
- plotly.express

Algorithms – Frequent Itemsets

- **FP-Growth**: an algorithm for extracting frequent itemsets that produces the same result as the original Apriori algorithm, but scales much better by using a frequent-pattern tree data structure, which is an "extended prefix-tree structure for storing compressed, crucial information about frequent patterns." [1]
- **Association rules generation**: metrics computed from frequent itemsets results
- For implementation of the above, we use the **MLxtend** package [2]
- We write code to **transform** VAERS data into "**baskets**" of items, one basket per VAERS report; each basket has items that are mapped from data fields of interest

GCP Resources Used

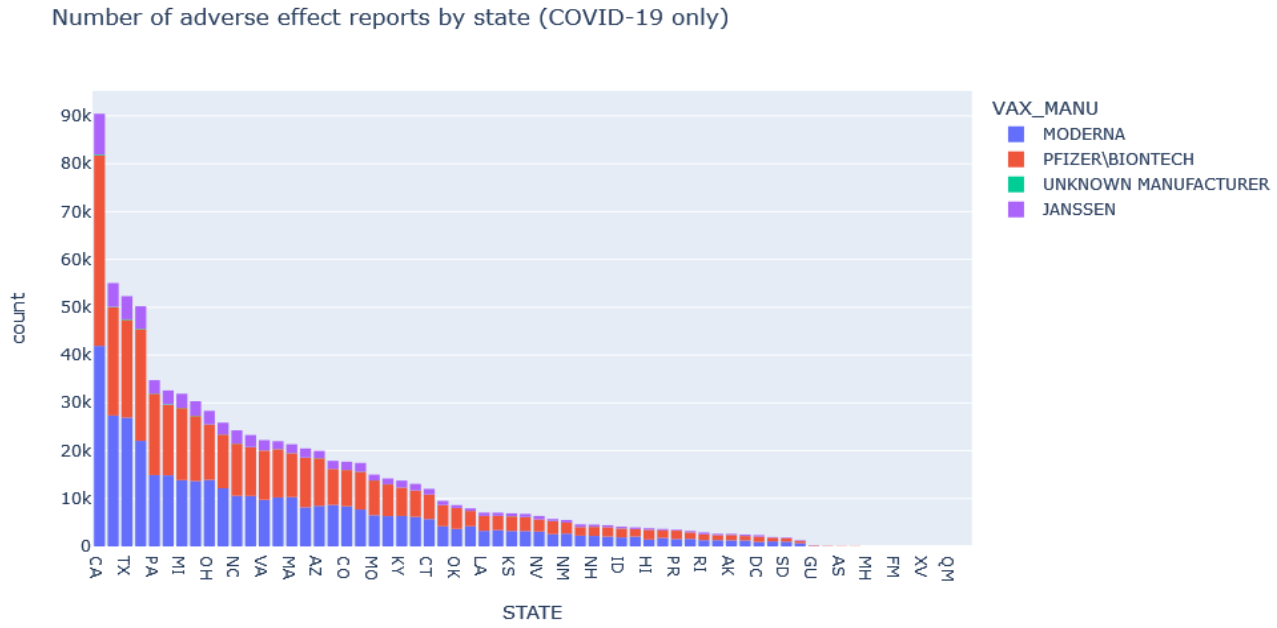
- Google Cloud Storage bucket to hold the data files
  - The code reads data from the bucket
- Google Cloud Functions for executing code

Project Interface

It is a proof of concept, accessible through Google Cloud Console.

Code and artifacts are in a git repository hosted on GitHub.

Results – EDA (1)

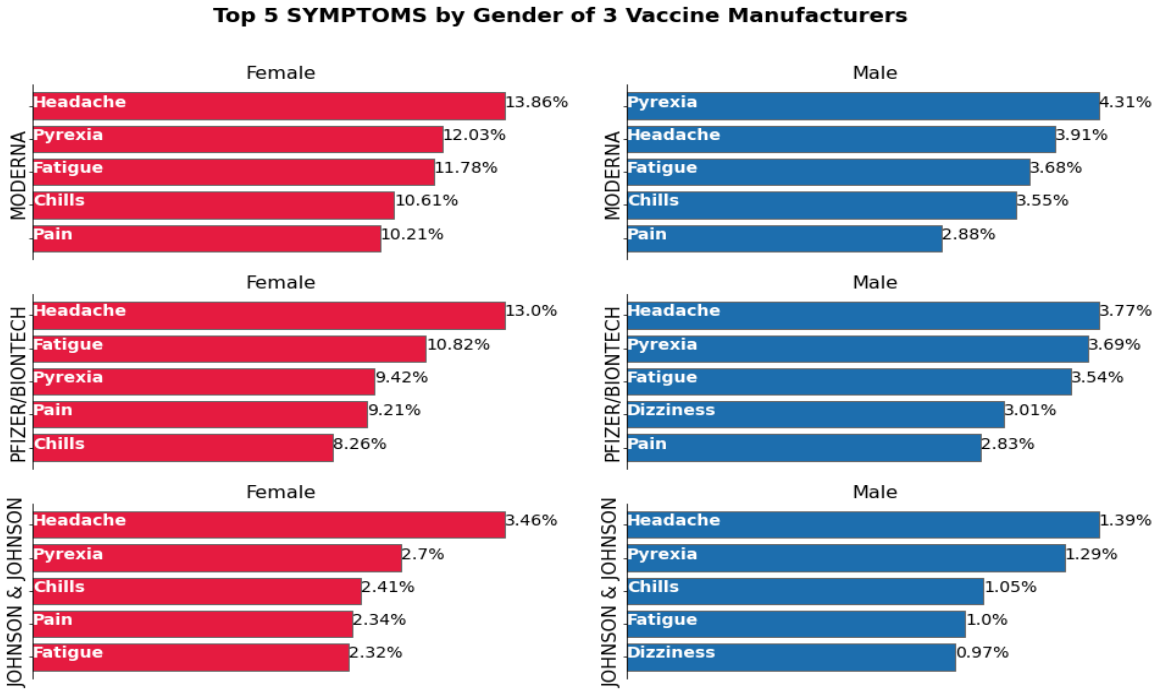


Numbers of adverse events reported in different states.

Distribution seems similar across states

There are a limited number of Adverse Events reported for Unknown Manufacturer

Results – EDA (2)



- The top symptoms for Females and Males are different for the Moderna vaccine, while the top symptoms are same, but differ in volumes, for the Pfizer/BioNTech and Johnson & Johnson Vaccine.
- Headache is the most common reported side effect, followed by pyrexia.
- Much higher percentages of reports for female patients involve headache, pyrexia, fatigue, chills, and pain, than reports for male patients – but only for Moderna and Pfizer/BioNTech, not Johnson & Johnson.

Results – Frequent Itemsets (1)

- Generated 14260 frequent itemsets from 993374 “baskets” (using 0.001 support threshold)
- Examples of baskets:
  - 'TX', 'Age 19-33', 'Female', 'Recovered', 'COVID19 (COVID19 (MODERNA))', 'Dysphagia', 'Epiglottitis'
  - 'CA', 'Age 65-78', 'Female', 'Recovered', 'COVID19 (COVID19 (MODERNA))', 'Anxiety', 'Dyspnoea'

Results – Frequent Itemsets (2)

Top frequent itemsets:

support	itemsets
0.677220	(Female)
0.428552	(COVID19 (COVID19 (MODERNA)))
0.312031	(Recovered)
0.152127	(Age 19-33)
0.056070	(TX)
0.004864	(Dysphagia)
0.173983	(Age 65-78)
0.096221	(CA)
0.040967	(Dyspnoea)
0.008004	(Anxiety)

Results – Frequent Itemsets (3)

- Generated 1,912 association rules (using 0.8 confidence threshold), sorted by leverage

antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
(Unknown Sex)	(Age 79-older)	0.036581	0.145562	0.030099	0.822807	5.652606	0.024775	4.822085
(COVID19 (COVID19 (MODERNA)), Unknown Sex)	(Age 79-older)	0.013102	0.145562	0.010840	0.827353	5.683834	0.008933	4.949045
(Death)	(Died)	0.008738	0.017816	0.008686	0.994009	55.792909	0.008530	163.949167
(Injection site pruritus)	(COVID19 (COVID19 (MODERNA)))	0.019756	0.428552	0.016782	0.849478	1.982206	0.008316	3.796437
(Life-threatening illness, Male)	(Hospitalized)	0.011318	0.101345	0.009365	0.827448	8.164707	0.008218	5.208033

Results – Frequent Itemsets (4)

- **Frequent itemset results show only "correlation" not "causation".**
- **(Female)** has support of 0.68, meaning 68% of adverse effect reports are for female patients. In comparison, females account for only about 52% of COVID-vaccinated people. [3] VAERS data do not tell us why this discrepancy occurs.
- **(Unknown Sex) -> (Age 79-older):** This is a curious pattern. It's suggestive of a data entry quality problem.
- **(Death) -> (Died)** with 99% confidence, not 100%. This looks like a data entry quality issue.
- **(COVID19 (COVID19 (MODERNA)), Product administered to patient of inappropriate age) -> (Age 14-18)** with high lift and conviction
- **(Age 79-older, Death) -> (Died)** with high confidence, lift and conviction
- **(Death, Male) -> (Died)** with support of 0.4678% versus **(Female, Death) -> (Died)** with support of 0.3559%.
- **(Age 79-older, Product storage error) -> (Unknown Sex)** : a curious association rule
- Various injection site symptoms are frequent

Alternatives Considered / Roadblocks

We investigated some alternatives that we ended up not using:

1. Use **DataProc** cluster to run code: The downside of its expense and overhead did not justify when our data is small enough and the code runs quickly enough on a personal computer.
2. Develop a **web application** interface: This was nice to have but not central to big data algorithms, and we did not have time.
3. **PySpark**: This would provide distributed computing but would also require using PySpark in addition to / instead of other Python packages. We did not have time.
4. **Dask**: This would provide distributed computing while using Pandas, Numpy and existing packages. But again, this turned out to be infeasible in the time we had.

References

1. Han, J., Pei, J., Yin, Y. et al. Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach. Data Mining and Knowledge Discovery 8, 53–87 (2004). https://doi.org/10.1023/B:DAMI.0000005258.31418.83
2. Raschka, (2018). MLxtend: Providing machine learning and data science utilities and extensions to Python's scientific computing stack. Journal of Open Source Software, 3(24), 638. https://doi.org/10.21105/joss.00638
3. https://usafacts.org/visualizations/covid-vaccine-tracker-states

Live Demonstration of Project Code

Questions and Answers

Thank You!