christine.n.buckler@gmail.com
linkedin.com/in/christinebuckler
github.com/christinebuckler

Provider Prescriber

Christine Buckler
Galvanize Denver Platte

Background

Objective:

Provide the top 10 most similar healthcare providers given a specific National Provider Identifier (NPI).

Use cases:

- Patients that have changed insurance plans
- Pharmaceutical representatives selling specialty products

The data:

- Public NPPES dataset
- 5,315,800 entries
- 328 features

Features used in this study include: entity type, gender, state of business location, specialties, credentials, sole proprietor status, and organizational subpart status.

Method

The brute force method compares each item to every other item which doubles the computation and memory storage with each addition to the input data set.

O(n²)

Instead, I used MinHash LSH (Locality Sensitive Hashing) as an efficient algorithm to find similar items using hashes. This technique allows for an approximate similarity solution.

Model

The MinHash LSH algorithm:

1. Transform data into binary vectors where non-zero values indicate presence of element.
2. Randomly permutate rows with k hash functions

row	S_1	S_2	S_3	S_4	h_1=x+1 mod 5	h_2=3x+1 mod 5
0	1	0	0	1	1	1
1	0	0	1	0	2	4
2	0	1	0	1	3	2
3	1	0	1	1	4	0
4	0	0	1	0	0	3

3. Compute MinHash Signature Matrix (these are the "min hash" values)

	S_1	S_2	S_3	S_4
h_1	∞	∞	∞	∞
h_2	∞	∞	∞	∞

	S_1	S_2	S_3	S_4
h_1	1	∞	∞	1
h_2	1	∞	∞	1

	S_1	S_2	S_3	S_4
h_1	1	∞	2	1
h_2	1	∞	4	1

	S_1	S_2	S_3	S_4
h_1	1	3	2	1
h_2	1	2	4	1

	S_1	S_2	S_3	S_4
h_1	1	3	2	1
h_2	0	2	0	0

	S_1	S_2	S_3	S_4
h_1	1	3	0	1
h_2	0	2	0	0

4. Group items into buckets within a similarity threshold.

5. Calculate estimated distance between items in the same bucket.

6. Tune parameters.

- Increasing the **number of hashes** increases accuracy but also increases computational cost and run time.
- Increasing the **similarity threshold** increases the number of buckets.

Measures

Jaccard distance: explicit relationship between intersection and union:

$$d(A,B) = 1 - \frac{|A \cap B|}{|A \cup B|}$$

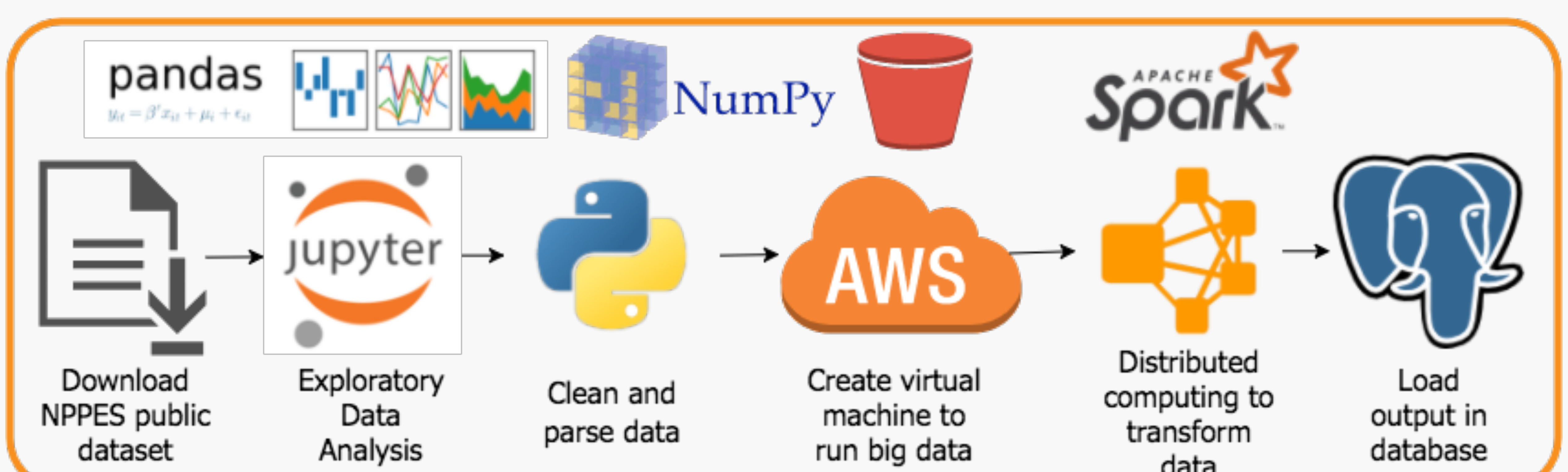
Where max error: $\varepsilon \approx \frac{1}{\sqrt{k}}$

For k=10, max error ~32%

Types of error:

False Positive: pair of dissimilar items grouped in the same bucket

False Negative: pair of similar items *not* grouped in the same bucket



Process Flow

Results

Similarity distances were computed for a subset of the data (10,000 NPIs) and stored inside a database that can be queried for specific NPIs.

Next Steps

With more time, I would like to explore the following areas:

- Improve virtual machine configuration to scale for more items
- Expand input method to allow for updates without re-hashing existing data
- Evaluate other features that add value to similarity measure such as standardized provider ratings
- Integrate query with NPPES API to give context to the results
- Add functionality to search for similar providers based on a list of NPIs
- Cluster or graph items to visualize groupings

References & Credits

1.Stanford's Mining of Massive Datasets Ch3
2.Pyspark Docs http://spark.apache.org/docs/2.2.0
3.https://en.wikipedia.org/wiki/MinHash
4.https://www.cs.utah.edu/~jeffp/teaching/cs5955/L5-Minhash.pdf
5.Getting Started on LSH by Vinicius Vielmo Cogo
6.Near Neighbor Search in High Dimensional Data (2) by Anand Rajaraman
7.Locality Sensitive Hashing at Uber Engineering https://databricks.com/blog/2017/05/09

Special thanks to the Galvanize instructors, DSRs, mentors, web dev, classmates and my family and friends for their support and encouragement during my immersive experience.