Meaning and noise in self-report public health data

CS224W // 9 December 2014 // Toman (005955208)

The Vaccine Adverse Event Reporting System contains self-reported adverse events following vaccination for safety monitoring. In seeking communities within VAERS, we develop a generative model for VAERS data which provides insight into the structure of the adverse event network. We then use the model and experimental techniques to gain insights into the nature of noise in the VAERS data. We identify specific challenges and needs for identifying communities in networks like VAERS, which is weighted and noisy, and we introduce the idea that affiliation graph models may produce “subclusters” that match emergent properties of the underlying clusters in the data rather than the underlying clusters themselves.

Introduction and motivation

The Vaccine Adverse Event Reporting System (VAERS) provides monitoring of the safety of vaccines to uncover safety problems that did not arise during clinical trials, such as events occurring predominately in underrepresented demographic subgroups, under off-label uses, or when clinical trial sample populations did not initially suggest statistically worrisome outcomes. This reporting system is co-sponsored by the CDC and the FDA. It allows any individual to describe adverse events experienced following vaccination.

Although the point of the VAERS data is vaccine monitoring, using it for that purpose is challenging. The safety events of interest are rare, and the data are quite dirty: the reports are a non-random sample of the population that received a vaccine, descriptions may contain errors, and there is no clarity whether the medical events reported were caused by the vaccine. Additionally, under the traditional statistical monitoring approach currently used for evaluation, when multiple reports of the same rare medical event are reported in slightly different terms each time, a potential safety problem can remain undetected because no single term occurs substantially more frequently than expected in the rest of the data.

To better receive vaccine safety information from the VAERS dataset, we consider the VAERS data as a network in which potential safety problems are communities of terms that coalesce through shared relationships to underlying causes. We seek algorithmic methods for finding clusters of coherent symptoms that are robust to the dirtiness of the dataset. We find that noise in the data poses a significant challenge to community detection, and we develop a generative model that enables us to characterize the noise. We identify techniques for noise reduction to find communities in the data, and we suggest that including cluster-based analysis of self-report data in the arsenal of public health officials may include value over pure statistical analysis.

Review of prior work

This work is based on four approaches to clustering and to initial work treating the VAERS reporting system as a network and suggesting that it offers a complementary approach to identifying patterns (Ball and Botsis 2011).

Hierarchical clustering is based on the intuition that the nodes that are most similar should be grouped together. This is a simple intuition that has been in common use since the 1960s (Wasserman and Faust 2007). Although it is conceptually simple, it is time-consuming to calculate – even under implementations involving priorities queues and other optimizations to reduce the processing time required – and its hierarchical philosophy requires that nodes can only participate in a single group at any level. We use hierarchical clustering as a baseline approach.

Non-negative matrix factorization finds two matrices A and B that can be multiplied to approximate the original matrix such that the returned matrices contain only non-negative values; in recent years it has become a popular clustering technique (Zhang 2012). Matrix A transforms the rows of the original data into the lower-dimensional space representing clusters, and matrix B transforms from that lower-dimensional space back to the higher-dimensional space of the input columns. Unlike hierarchical clustering, this approach allows a single row (symptom) to participate in multiple clusters; however, because the clusters are based on decompositions of high dimensional spaces, they may be less interpretable.

Mixed membership stochastic block models use a Bayesian approach to estimate the parameter of a model of *k*-groups where each group has a proportionality in the network and the likelihood of edges from a node to each other node depend on their group membership (discussed by Airoldi et al. 2008, as well as in multiple other papers). As such, it defines the entire network in terms of a smaller number of *k*-groups that interact with each other in characteristic ways. This approach, too, allows symptoms to participate in multiple clusters.

BigClam (Yang and Leskovec 2013) does not operate from the assumption that communities are separated by few edges, but rather that communities are tied together by many edges between nodes in multiple communities simultaneously. This is a scalable methodology similar to community affiliation graph models (AGMs) (Yang and Leskovec, 2012); BigClam's fitting problem has been simplified to enable it to perform well on larger graphs. We use both algorithms in this paper depending on the size of the problem. Their theoretical perspective is aligned to our expectations about the VAERS dataset: nodes that participate in multiple communities (like "pyrexia") will tend to have edges to all of their communities, and to link to other nodes that have similar community profiles. Both algorithms allow symptoms to participate in multiple communities.

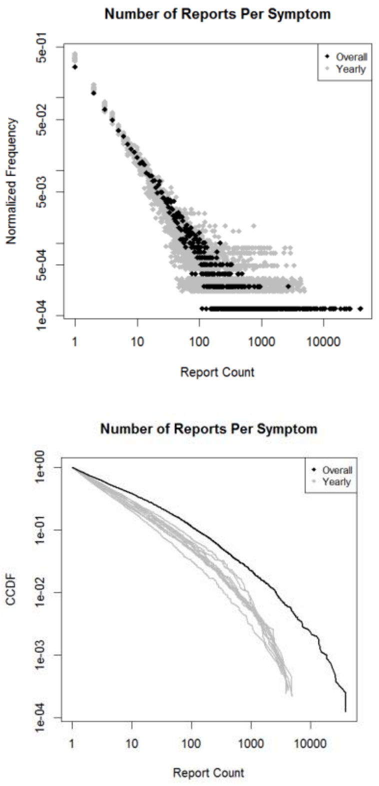
Data

The 25-year-old VAERS system[1] monitors approved vaccines in case safety concerns were not identified during clinical trials. Reports include demographic information, the vaccine(s) received, a textual description of the adverse reaction, and vaccine information such as the date, type, manufacturer, and lot number. The textual description of the adverse event is subsequently coded into a defined vocabulary of adverse event terms, which include primarily symptoms but also a smaller number of medical tests and results. The remainder of this paper uses the phrase "symptom" to refer to any adverse event term.

We work primarily with the symptom-report and symptom-symptom data across the last ten years, including the first nine months of 2014. The average year has about 25,000 reports, with the interquartile range on number of symptoms per report being [2, 4] or [2, 5], and the interquartile range on vaccines being [1, 2] or [1, 3]. The maximum number of symptoms associated with a single report is 153 in 2009; the maximum number of vaccines associated with a single report is 11 in 2008. As suggested by the summary statistics, the distribution of the number of times each symptom is reported follows a heavy-tail distribution (see Figure 1).

To form the symptom-report network, we build an undirected, unweighted bipartite network for each year of data, linking each symptom with its mentioning reports. We also form an undirected weighted symptom-symptom unipartite network for each year of data, linking each symptom to the other symptoms with which it was co-reported through multiplying the two-mode adjacency matrix by its transpose (such that each pair of symptoms is linked by an edge with a weight reflecting the frequency of co-reporting). The average yearly symptom-report network has on the order of 30,000 nodes and 100,000 edges, and the average yearly symptom-symptom network has around 3000 nodes and 180,000 edges.

Figure 1. Heavy tail of symptom reporting.



[1] https://vaers.hhs.gov/

The symptom-symptom network simultaneously exhibits many nodes that link only to other terms in the same report and a small diameter of approximately 2.6 due to the most common symptoms having a large number of edges. For instance, pyrexia, the most common node, has direct connections to 62% of other symptoms in the 10-year network.

Method primitives

We find clusters of coherent symptoms that are robust to the dirtiness of the dataset using variants of four approaches, a self-implemented ensemble hierarchical clustering method, non-negative matrix factorization, mixed membership stochastic block modeling, and the BigClam/affiliation graph model algorithms.

Clustering methods

Under all methods, when a set number of clusters is desired, we consistently use 30 clusters or consistently use the Bayesian Information Criterion (BIC) for the number of clusters in the data.

Ensemble hierarchical clustering

We implement an ensemble variant of hierarchical clustering that merges the results from hierarchical clustering based on cosine dissimilarity, Jaccard dissimilarity, and Euclidean distance once those methods can submit clusters of no more than 10 nodes. The ensemble approach accounts for substantial observed differences in final clustering given multiple distance metrics (on 30 sample nodes there is 0% overlap from method to method when limiting nodes to their 10-closest neighbors, and 3% overlap from method to method when limiting nodes to their 20-closest neighbors). Efficiency improvements (as for instance in Murtagh 1983) were introduced.

Non-negative matrix factorization

We use non-negative matrix factorization to generate two matrices that, when multiplied, approximate the original similarity matrix network representation. We use the row vectors of the first matrix to define each node's degree of membership in each cluster.

Mixed membership stochastic block modeling

We use mixed membership stochastic block modeling as implemented in the R *lda* package.

BigClam

We use bigclam and agmfit as implemented in the C++ SNAP library.

Model

To explore characteristics of the VAERS data, we develop a generative model that allows us to simulate synthetic VAERS-like data. The synthetic data has no noise and also provides "truth" clusters that assist in evaluation.

The model is based on the processes by which we believe VAERS data is generated in the real world. The model has two stages. In the first, we set up relationships between terms. In the second, we sample from those relationships.

Specifically, the model posits that VAERS data is generated by a set of underlying medical problems, some of which are common and many of which are rare. For instance, there is an underlying medical problem of analphylaxis that is relatively common, and there is an underlying medical problem of thrombotic thrombocytopenic purpura (a microscopic blood clotting disease that may be triggered by live vaccines) that is extremely rare. Each underlying problem can be defined exhaustively by a set of relevant terms from the defined vocabulary used in VAERS. For instance, the underlying problem "anaphylaxis" can manifest with words like "urticaria", "flushing", "angioedema", "rhinorrhea", "dyspnea", and many others. Symptoms have relationships to underlying problems, and they also have relationships to other near-synonym terms; "dyspnea" commonly occurs with terms like "chest pain", "cough", "asthenia", "dizziness", and "activities of daily living impaired." Reports then reflect one or more underlying medical problems. The symptoms listed in a report are selected from the possibly relevant terms from the corresponding underlying medical problem. The number of terms included in a report varies, and some terms (like pyrexia) are much more likely than others to be included. Theoretically, selecting which terms are used in reports according to preferential attachment aligns with priming effects toward already used terms in the reporter and the data entry person who selected the defined vocabulary.

Following the above processes, we developed and tested the following generative model:

1. We generate a set of *N* underlying possible reportable events. Each event *i* has a relative likelihood p_i. Each event is potentially characterized by a vector of terms v_i. The terms themselves form clusters of coherent meaning, and whenever a term appears in a reportable event, it increases the likelihood of its near-synonym terms to participate in that same event. This process ensures that synonym clusters appear in the underlying reportable events.
2. Each report reflects *k* reportable events. Each report selects a random subset of terms from the symptoms of its parent underlying event, where the subset could be as large as the entire set of terms available. The terms are selected so that terms that already appear often in the network are more likely to be re-selected, to account for the observation that a few symptoms participate in many reports and many symptoms participate in very few reports.
3. We generate *j* reports using this model.

The basic model used in the paper has the model parameters initialized as follows to reflected observed values and emergent properties for a year of data:

- *N*, the number of underlying events, is variable; we use values between 10 and 500
- *p*, the relative likelihood for each underlying event, reflects a geometric distribution with $p=0.5$, such that a few events are very likely ("pain at injection site" and "fever") and many events are extremely unlikely ("petechiae" and "thrombotic thrombocytopenic purpura")
- *q*, the vocabulary size, is 1200 terms
- *v*, the underlying event vectors of terms, each consist of 150 terms
- *k*, the number of underlying events per report, is 1
- *j*, the number of reports to generate, is 30,000, reflecting the empirical number of reports in a year

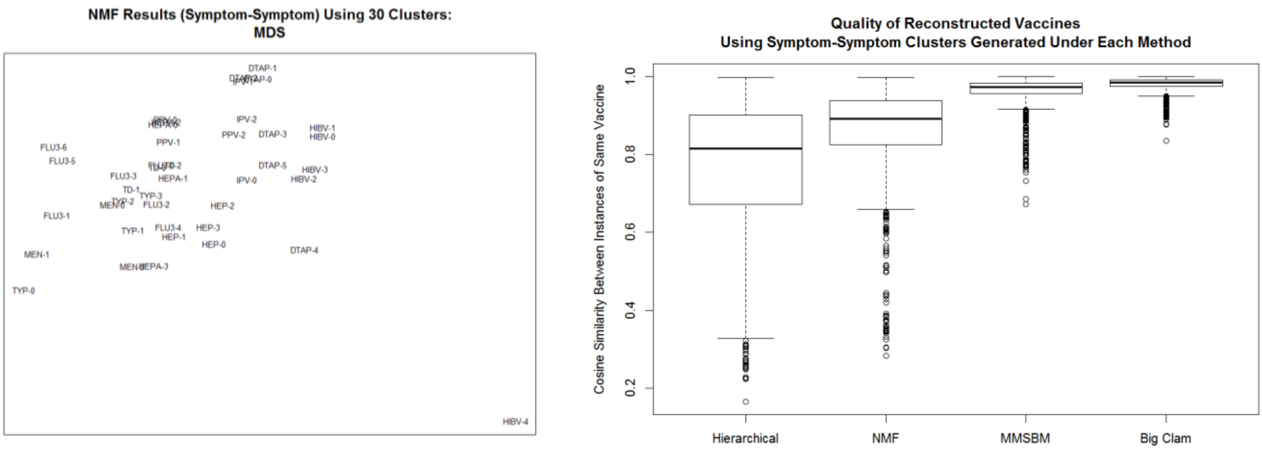
Evaluation methods and results

Evaluation of clustering methods

Ability to regenerate vaccines

Assuming that the types of symptoms that people exhibit following each vaccine are meaningfully different across vaccines but similar among the same vaccine, meaningful clusters should be able to reconstruct which vaccines are equivalent to each other. To test the clusters using a vaccine reconstruction approach, we identify all the vaccines/manufacturer combinations in a year with more than 20 reports, and from that set we select the vaccines with

Figure 2. At left we represent in two dimensions a sample multi-dimensional scaling projection of the similarity between each pair of vaccines in the reconstituted space. At right we display the performance of each completed approach at reconstructing, as per the multi-dimensional scaling result.



at least 3 manufacturers. If a method accurately identifies clusters, then when we represent each vaccine in terms of its clusters, we should find that instances of the vaccine are still near each other (see Figure 2 left). We learn the clusters from 2004 data and apply them to 2005 data to ensure the analysis is out-of-sample, and then we calculate the cosine similarities between pairs of vaccines in the reconstituted space.

All the approaches have outliers indicating that the distance between vaccine variants fluctuates. However, as represented in Figure 2, we find a median cosine similarity above 0.8 for all the approaches, with BigClam best able to recover clusters from a single year that are predictive of the next year's relationships.

Medical coherence

An evaluation of the medical coherence of clusters under each method suggests that although BigClam performs well, its clusters are not clearly meaningful. When using the Bayesian Information Criterion (BIC), BigClam selects 137 clusters that are not clearly coherent; they involve multiple body systems, different age ranges, and do not have clear mappings to common adverse events (see Table 1 for sample results on FLU3). However, we can achieve medical coherence from BigClam using a smaller number of clusters. For instance, if we request four clusters, BigClam finds two generic "flu-like illness" clusters, a variety of symptoms related to a rare but widely known auto-immune response called Guillain-Barre syndrome, and sepsis.

This suggests that although BigClam is able to find structure in the data that persists across time, the structure may not map directly to a natural interpretation of "communities" – in other words, the mathematical structure identified by BigClam may not have obvious meaning. We revisit this idea later.

BigClam Sample BIC-Decided Clusters	**BigClam Sample Few Clusters**
• Nervous system disorder, Pulmonary congestion, Sudden infant death syndrome, Liver disorder, Irritability • Paresis, Lab test abnormal, injection site inflammation, nervousness, hyperreflexia • Erythema, Pyrexia, Petechiae, Purpura, Pharyngolaryngeal pain	• Pyrexia, Erythema, Pain, Pruritus, Chills • Lab test abnormal, Headache, Pyrexia, Dizziness, Vomiting • Asthenia, Gait disturbance, Paraesthesia, Guillain-Barre syndrome, Hypoaesthesia • Lab test abnormal, sepsis, pyrexia, cardiac failure, dyspnoea

Table 1. Sample contents of n clusters, where n is BIC-decided (left) and deliberately limited to 4 (right) using the BigClam algorithm on FLU3 data. BigClam's clusters may have mathematical meaning that allows good performance on re-identifying vaccines given only symptom information, but their medical meaning is not immediately obvious.

Evaluation of model

As an initial evaluation of the model, we verify that when we set the model parameters to match basic properties of observed data, the model produces structures that have complex qualities that continue to match the data. For instance, when we pass it parameters corresponding to the FLU3 vaccine, the model produces networks that exhibit other observed properties, such as the distribution of symptoms in each report (see Figure 3).

As a more complex evaluation of the model, we run the AGM algorithm to generate clusters, and we compare the estimated clusters to the underlying truth sets of symptoms that generated the network. Using the model in this way, we can (1) evaluate how the model parameters affect our ability to detect the true clusters of interest in VAERS data, that is, the underlying symptom-symptom clusters, and (2) begin to contextualize the possible performance of the clustering methodology for some particular observed data. We evaluate performance through calculating the average Jaccard similarity on the best AGM-generated-cluster-to-underlying-cluster pairing set.

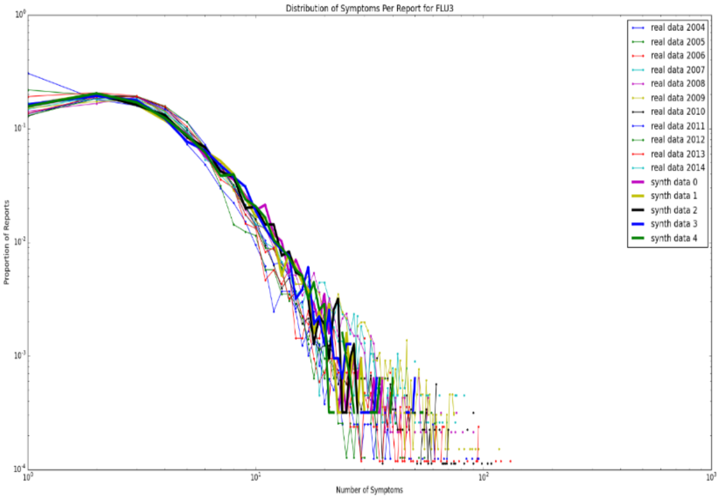


Figure 3. Sample chart for FLU3 data illustrating that the model produces characteristics similar to those of the real-world data we are simulating.

In addition to expected conclusions like the more reports, the better our ability to detect underlying clusters, the model suggests interesting phenomena occurring in VAERS. For instance, we find evidence of a high degree of synonymy and self-redundancy related to clusters emerges from the graph structure itself, rather than from the medical or topical domain: we find that vocabulary size has no effect on performance, rather than evidence that as the number of terms increases, it becomes harder to find clusters. We also find that when the data contains a report with a large number of nodes, we are much less able to reconstruct the underlying problems generating the data. This effect may arise because extremely large clusters in the data tie many unrelated terms together, which substantially alters the overall network structure with what is potentially noisy data. As work in community detection on VAERS continues, researchers may want to look specifically to the effect of extremely large clusters, especially as the probability of finding such clusters increases with graph size. We address the general issue of noise in VAERS in the next section.

On the whole, the synthetic model does not perform especially well. Given a year of data from a common VAERS vaccine like FLU3 (3000 reports, 1000 symptoms, 100 max nodes, a variety of underlying symptoms), we expect overlap of 0.41 with the truth. In general, unless the data is minimal and clear, we should expect that the symptoms clusters recovered by AGM do not reflect the symptom clusters that generated the network (see Table 2). The larger networks in particular have lackluster performance.

The results on synthetic data contrast with BigClam's excellent ability to identify that vaccines are self-similar using clusters learned on previous data (>0.80 cosine similarity), and support the qualitative evaluation that BigClam clusters have a tendency to have less than clear meaning. The disconnect suggests the need for further evaluation of AGM and BigClam's performance.

The next section addresses two questions raised during the model evaluation: the effect of noise in VAERS data and the disconnect between the mathematics and the meaning of BigClam clusters.

Table 2. Characterizing performance of synthetic network.	Average Jaccard Similarity
1st Quartile: Reports: 25 Max Terms: 5 Vocabulary: 10 Syndromes: Variable	1.00
3rd Quartile Reports: 500 Max Terms: 50 Vocabulary: 500 Syndromes: Variable	0.22
Maximum Reports: 4000 Max Terms: 150 Vocabulary: 2000 Syndromes: Variable	0.17

Exploration of lower performance in VAERS data using model

In this section, we test hypotheses about the challenges posed by community detection on this data, which gives insight into community detection algorithms and the VAERS dataset. We are interested especially in the question of dataset noise and its relationship to reconstructing clustering algorithms that account for increased likelihood of a link as shared community memberships increase, as represented in this paper by AGM and BigClam, and in further understanding conflicting evidence about the quality of BigClam clusters.

We observe three properties of the VAERS data that may contribute to difficulty in finding true underlying health events over clusters that are mathematically meaningful: (1) the central regions of the network are quite dense, (2) the network is weighted, and (3) self-reported data is likely to be less than immaculate. We perform three meta-evaluations to better understand the effect of each of these dimension on the VAERS data and on the affiliation graph model family of approaches for community detection.

Dense central clusters

Our first hypothesis is that it is possible for two clusters to overlap "too much" for the original communities to be recoverable. Our intuition is that clusters that are completely independent are trivially identifiable, and clusters that are completely intermingled into a complete graph are impossible to identify. We suspect that there is a point where the AGM method is unable to identify meaningful clusters, and we seek (a) whether that point exists, and (b) what it is.

To approach the problem of dense central clusters, we initialize generic generated networks (preferential attachment, random, and small world) to two components of modeled networks, and then gradually introduce edges as ordered by an approach that makes some nodes more likely and some less likely to receive edges, as we expect is the case in VAERS. Figure 4 illustrates the process.

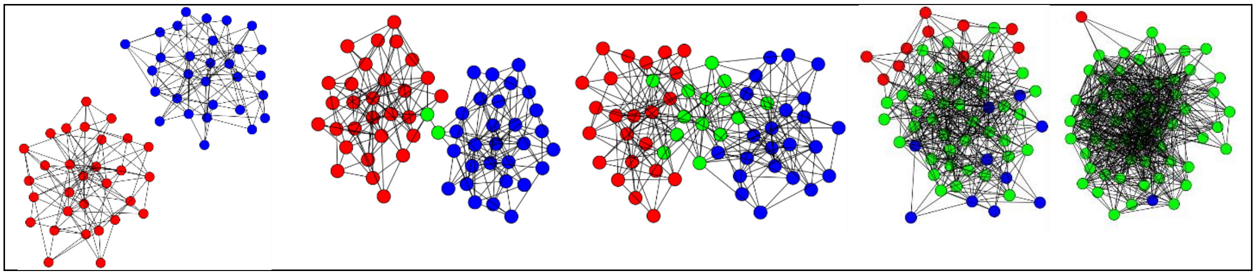


Figure 4. Illustrated progression of nodes being merged (preferential attachment networks), with many intermediary stages not displayed. Blue and red represent initial clusters, and green represents nodes that have connections to both clusters (each green node still has a constant cluster assignment). Note that this approach illustrates a continuum of community detection algorithms: algorithms based in the idea that there are few connections between communities are appropriate for network structures like those at the far left of the figure, whereas AGM and BigClam are appropriate for data that is similar to the center and center-right of the figure.

When fewer than 5% of the possible links between clusters exist, AGM is able to recover the clusters successfully. However, as more links are added, it becomes steadily more challenging for the AGM to recover the same clusters (Figure 5).

This finding suggests that AGM and BigClam do not find medically meaningful clusters in the VAERS dataset, but rather mathematically meaningful “subclusters.” For instance, AGM and BigClam might prefer to find three clusters in networks like those in the later stages of Figure 4, adding a green cluster to reflect the overlapping nodes and achieve the simplest possible cluster representation for the data. These “subclusters” are not synonymous with the clusters that generated the data, and thus they do not perform well when tested against truth, as we found in the model evaluation.

In the VAERS network, the mathematical “subclusters” found by AGM might be comparable to saying that the overlap between “anaphylaxis” and “heart attack” symptoms are their own cluster, full of symptoms like “chest pain”, “dyspnea”, “sweating”. Such a cluster would not include terms like “swelling”, “urticaria”, and “erythema” (anaphylaxis) nor would it include terms like “myocardial infarction”, “nausea”, and “fatigue” (heart attack). One could imagine that the emergence of these subclusters might sometimes reveal a true previously unperceived hierarchical structure in the data, as when two underlying problems both cause a shut-down of the same organ class. However, a subcluster might just as easily be the result of finding signal in noise (as described above in the overlap between anaphylaxis/heart attack).

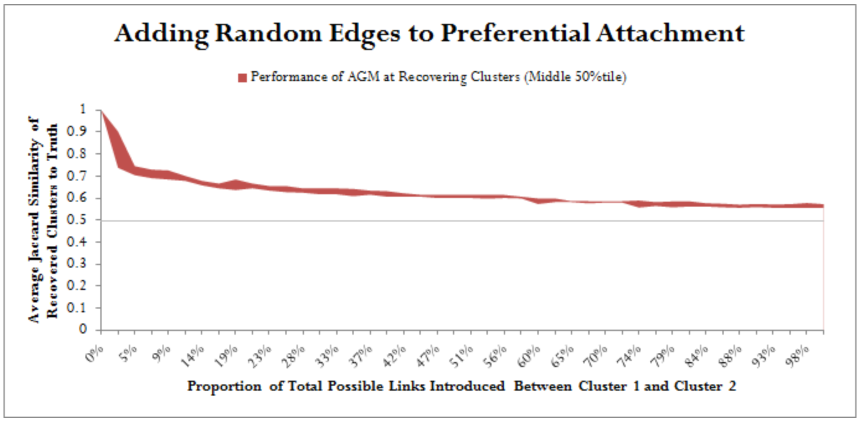


Figure 5. Effect of adding random edges to a preferential attachment network; steep drops or discontinuities indicate amounts of overlap in which AGM does not perform well at recovering the original clusters. Note that the random network and the small world network had similar profiles to this one, though the small world network had a larger interquartile range.

Weighted

We hypothesize that another reason the VAERS data clusters are not medically meaningful is that not all edges in the network are equally meaningful. In particular, because the VAERS data is weighted but AGM/BigClam operate on unweighted data, we hypothesize (a) that the level of dichotomization matters, and (b) that dichotomizing at presence/absence may be inappropriate given the noise in the data.

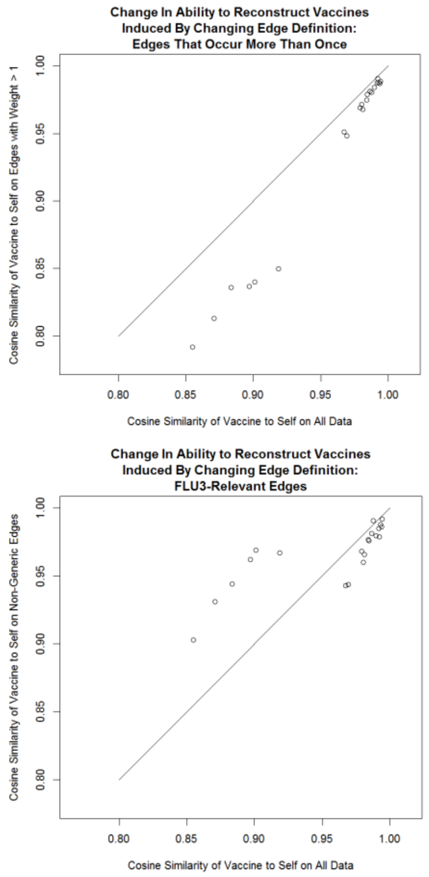


Figure 6. Ignoring edges of weight 1 when forming clusters decreases performance (top), but reweighting edges according to how much more they compare to the expected value in the rest of the network improves low clustering performance (bottom).

More formally, we suggest a conceptual revision to the AGM model approach to better reflect VAERS data, which itself suggests the concern addressed in this section. Namely, whereas links between nodes in the AGM model are formalized as $p(e_{u,v}) \propto f(\{c_u \cap c_v\})$ where c_i is the set of communities that node i belongs to, a formulation that better fits the VAERS data might be $w(e_{u,v}) \propto g(\{c_u \cap c_v\}) + \varepsilon$. The revised formulation moves from considering the probabilities of edges to the weights of each edge (where some weights may be zero), and we incorporate a term ε that accounts for the possibility that the observed weight value be larger or smaller by some amount. Under the new formulation, we see that when $|\varepsilon|$ is large, the error term can have an overpowering effect on the outcome variable w. If we dichotomize w at a low value like 1, then the range of $|\varepsilon|$ does not even need to be similar to the range of g for ε to have an outsized importance on the outcome variable. At low levels of dichotomization, then ε is likely to account for both observed and missing edges that are incorrect, whereas if we dichotomize w at higher values, then ε has a weaker influence on the outcome variable w.

To this end, we build three edgelists for the FLU3 vaccine. The first uses all 2004 data and dichotomizes at "any observations." This is the default dataset. The second dataset account for the effect of ε by dichotomizing at "more than one observation." The third dataset re-weights the edges according to the gap between the bootstrapped interquartile range of relative weights for FLU3 vs. the interquartile range of relative weights for all the data; any edges that are negatively weighted are removed, thereby leaving only edges that are non-generic and likely related to FLU3 in particular. We calculate whether we can reconstruct the FLU3 vaccine using clusters derived by BigClam on each of these networks.

We find that a dichotomization scheme that simply ignores the data of weight 1 performs worse than a dichotomization scheme that uses all the data (Figure 6, top). This suggests that although there is indeed noise in the data, there is also valuable information captured by edges of weight 1, and it is inappropriate to entirely ignore such edges. We also find that a more data-driven schema for dichotomization may be useful: we see some improvement in the performance of BigClam when the dichotomization scheme is changed to account for the effect of ε through interquartile ranges, especially for vaccine instances that previously were the most challenging to correctly identify (Figure 6, bottom). For applications where boosting the performance of the "worst we will do" at finding clusters, interquartile range has promise. Thus we find that there is noise in the data, and that pointed attempts to eliminate that noise may improve clustering performance.

Random noise

We delve deeper into the ε value to identify whether there is noise from random edges in the network and to characterize the extent to which VAERS has random noise. We explore the extent to which random noise is rife in VAERS through a minimum description length approach: we begin with a network, we introduce increasing amounts of random noise to the network as random edges, and we measure the gzip compression size of the resulting network. If the initial network is highly ordered, then the initial compression will be quite good and the relative increase in compression size from introducing random noise will be quite large. However, if the initial network is highly disordered, then the initial compression will be less good and the relative increase in compression size from introducing random noise will be small.

Consider, for instance, that the increase in entropy from the difference between adding k and $k + 1$ edges is larger on a ring network than on a random Erdős-Rényi G_{np} network. We use entropy on the stream of endpoints to each edge rather than minimum description length because its formulation is straightforward.

$$inc(H_{ring}) = -\sum_i^n \frac{2k+3}{2nm}\log\frac{2k+3}{2nm} + \sum_i^n \frac{2k+1}{2nm}\log\frac{2k+1}{2nm}$$
$$= -\frac{1}{2m}[(2k+3)\log(2k+3) + (2k+1)\log(2k+1)]$$

$$inc(H_{rand}) = -\sum_i^n \frac{\binom{n-1}{i}p^i(1-p)^{n-1-i} + \frac{2k}{n}}{pn(n-1)}\log\frac{\binom{n-1}{i}p^i(1-p)^{n-1-i} + \frac{2k}{n}}{pn(n-1)}$$
$$+ \sum_i^n \frac{\binom{n-1}{i}p^i(1-p)^{n-1-i} + \frac{2k+2}{n}}{pn(n-1)}\log\frac{\binom{n-1}{i}p^i(1-p)^{n-1-i} + \frac{2k+2}{n}}{pn(n-1)}$$
$$\le -\frac{1}{pn(n-1)}[(2k+3)\log(0+\frac{2k+2}{n})$$
$$+ p(n-1)\log\left(0+\frac{2k+2}{n}\right) + (2k+1)\log\left(0+\frac{2k}{n}\right) - p(n-1)\log\left(0+\frac{2k}{n}\right) - 2\log(pn(n-1))]$$
$$\le -\frac{1}{pn(n-1)}[(2k+3)\log(2k+2) - (2k+3)\log(n) + p(n-1)\log(2k+2) + (2k+1)\log(2k) - (2k+1)\log(n)$$
$$- p(n-1)\log(2k) - 2\log(pn(n-1))]$$

We thus find that the increase for a ring network is larger than the increase for a random network through pairwise comparison of elements and the recognition that $-\frac{1}{pn(n-1)} \le -\frac{1}{2m}$ because $m \ \le n(n-1)$.

Now we turn to the observed results of how much noise there is in VAERS, contextualized through the model and other real-world networks. We find that as additional edges are added at random into VAERS data, the compression initially gets worse before hitting an inflection point at which it begins to improve again, perhaps because the DEFLATE algorithm is now able to find repeated sections that it can replace with pointers.

Considering the minimum description length results as displayed in Figure 7, we find results consistent with the hypothesis that the VAERS data is noisy. We find that it is slightly less noisy than the configuration model, which is in keeping with the configuration model being generated randomly to match a specified degree distribution. We also find that it is slightly more noisy than the synthetic data, which does not simulate any random noise. In general, once we reach the inflection point, the continued growth in the relative file size is similar to the growth in the $G_{nm.}$

For contextualization, we also consider the compressibility of a variety of large networks. Other non-self-report datasets exhibit a different pattern of compression. Even when the edge counts of these networks are increased by 50%, the ratio of compressed-to-original data continues to grow. This suggests that the inflection point at which compression begins to improve again in the face of randomness occurs much later, if at all, in most comprehensive and clean networks – and thus that VAERS data has significantly more "unexpected" edges between nodes than most peer network data.

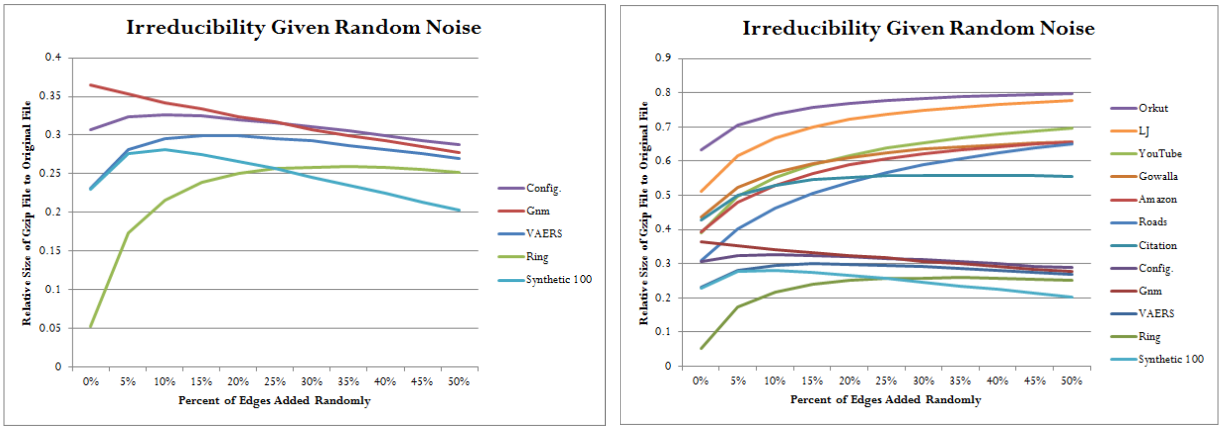


Figure 7. The compressibility of VAERS data in comparison to a variety of peers networks designed to be similar suggests that VAERS data is noisy (left). The compressibility of VAERS data vs. a variety of peer networks contextualizes that noise (right).

Conclusion

We explored a network view of VAERS data, a self-report vaccine safety monitoring dataset whose communities can assist in improving public safety. We develop a generative model for VAERS data that accurately reproduces network characteristics observed in the VAERS data. We then use that model to explore the characteristics of the VAERS data, specifically as relates to two findings: first, that we can find very high quality clusters with mathematical meaning that simultaneously do not reflect the clusters that generated the data, and second, that there is noise in the VAERS data that complicates our attempts to find clusters.

Specifically we find that although AGMs can decompose dense clusters, the result does not reflect the original clusters; users of AGMs should thus be aware of the sorts of clusters that it will produce, and be wary of entirely data-driven clusters produced by that method. We also find that the weighted nature of the VAERS network is challenged by the dichotomized nature of many community detection algorithms, and that we can improve the performance of lower-performing clustering by using a dichotomization scheme that uses weights and accounts for the possibility of error. As part of this work, we propose a more generalized model for understanding overlapping communities where link likelihood increases with shared communities, of which AGM and BigClam are a subset. Finally, we quantify the relative amount of noise in the VAERS data compared to peer synthetic networks, compared to synthesized data using the model developed for this project, and compared to a variety of real-world observed large networks, and find that VAERS has a substantial amount of random noise.

Further work might address community detection in weighted networks, efficient ways of identifying and removing noise from VAERS data so as to support the public health mission, and the development of machine learning methods specifically to find communities in the VAERS case.

Works cited

Airoldi, E. M., Blei, D. M., Fienberg, S. E., and Xing, E. P. 2008. Mixed membership stochastic blockmodels. Journal of Machine Learning Research, 9, p. 1981-2014.

Ball, R. and Botsis, T. 2011. Can network analysis improve pattern recognition among adverse events following immunization reported to VAERS? Clinical Pharmacology and Therapeutics, 90 (2), p. 271-279.[2]

Murtagh, F. 1983. A survey of recent advances in hierarchical clustering algorithms. The Computer Journal, 26 (4), p. 354-359.

Wasserman, S. and Faust, K. 2007. Social Network Analysis. Cambridge University Press, New York.

Yang, J. and Leskovec, L. 2012. Community-affiliation graph model for overlapping community detection. IEEE International Conference on Data Mining.

Yang, J. and Leskovec, J. 2013. Overlapping community detection at scale: A nonnegative matrix factorization approach. ACM International Conference on Web Search and Data Mining.

Zhang, Z. 2012. Non-negative matrix factorization: models, algorithms and applications. Holmes, J. and Jain, L. (Eds). Data Mining: Foundations and Paradigms. Intelligent Systems Reference Library (24). Springer: New York.

[2] The author was involved in building a network analysis software tool in Java as follow-on work to this paper. The author has not previously worked on identifying, evaluating or considering computational algorithms for finding clusters in VAERS or any other data. As such, the content of this paper is entirely new.