

# Interactive Visualization of Multivariate Clinical Data in Force Graph

Mingi Ryu\*

University of Illinois at Urbana-Champaign

## ABSTRACT

Visualization of structure and multivariate data are often correlated, yet remains a major challenge. Recent advances in graph visualization have introduced domain-specific visualizations that are mostly tailored for domain experts. This paper introduces a web-based application where a basic set of interaction and visualization techniques is used to perform visualization tasks.

**Index Terms:** Human-centered Computing—Visualization, Force-directed Graph, Multivariate networks

## 1 INTRODUCTION

As an alternative to the hierarchical graph used by Lineage [4], force graph is used to visualize the dataset without additional abstractions. Nevertheless, force-directed graph runs the risk of "hairball-like" visualization due to overdraw and clutter [5], while the hierarchical graph retains a consistent layout and most of the topology via decycling and linearization [4].

To address the possible overdraw and clutter, the application uses an interactive force graph framework created by Vasco Asturiano. force-graph is a web component to represent a graph data structure in a 2-dimensional canvas using HTML5 canvas for rendering and d3-force for the underlying physics engine [1]. It supports a core set of features such as node-link selections, colored nodes, directional links, view manipulation, and more as an interactive framework.

## 2 DATASET AND IMPLEMENTATION

The dataset provide a sample of 10 families in the Utah Population Database that have a high incidence of suicide, as defined by their FSIR (Family Suicide Incidence Ratio) [2]. It provides information on the topology of the families, and each data point has a list of approximately 30 demographic and clinical attributes.

To make full use of the dataset, attributes and structure datasets were combined into a single dataset by joining on the "personid" field. Because the "personid" field contained duplicates with different attribute values, only the first instance of the duplicate was stored in the final dataset and no form of aggregation was made to merge the duplicates.

Since the force-graph framework utilizes d3-force library to visualize the graph data, the dataset was converted from rows and columns to nodes and links format. For this particular dataset, a single row corresponds to a single node and a single directed link represents the relationship between a parent and a child. In addition, not all 30 attributes were included since certain attributes did not provide useful information (race and location) or were essentially redundant (AgeID- and Nr.Diag-).

While the key component of the application is the force-graph, a significant amount of JavaScript, HTML, and CSS was written for the implementation of interactive components. In addition, the Semantic UI framework was used for the modern look and responsive design. For more details, the code can be found on GitHub. <https://github.com/mingir2/ten-families-graph/>

\* e-mail: mingir2@illinois.edu

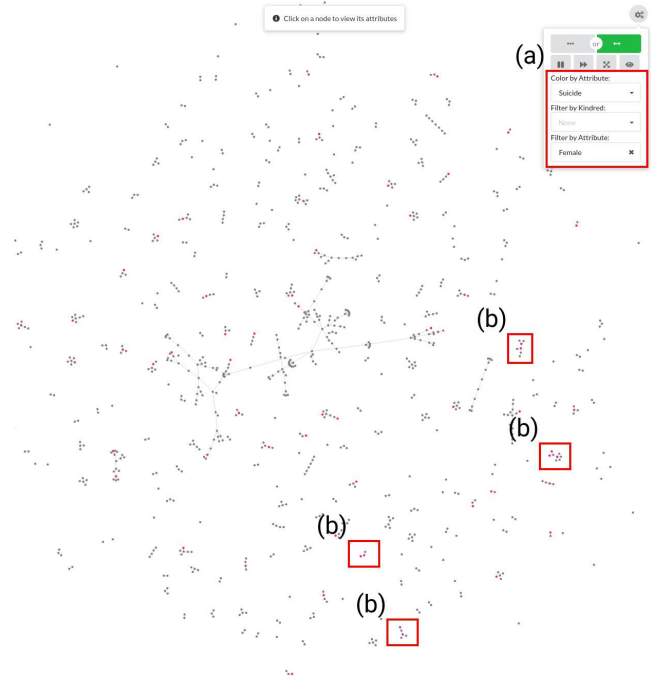


Figure 1: A graph of suicide cases with female attribute. (a) Colored by suicide cases and filtered by female attribute. (b) Sub-graphs with co-occurrence of suicide cases with female attribute.

## 3 TASKS AND RESULTS

To better understand the outlined tasks [2], the Graph Task Taxonomy was used to dissect the four high-level task into a set of broader low-level tasks [3]. Within each paragraph, each high-level task is broken down into a set of tasks, followed by an example figure for further clarification.

### 3.1 Co-occurrence

- For a given target individual, identify similar cases, including how they are related amongst themselves (such as whether they co-occur in a given family).

Given that the individual has an attribute of interest, a topology-based task may be used to identify similar cases and co-occurrence in a family. As such, the task is broken down into the following set of tasks: [Find on Nodes + Filter on Nodes + Retrieve Value on Nodes + Scan on Nodes + Find Adjacent Nodes on Nodes].

In Fig. 1, an individual is chosen based on suicide cases with female attribute. In order to identify similar cases, the graph filtered by female attribute and colored by suicide cases. Then, the graph is scanned to locate any sub-graphs with co-occurrence.

### 3.2 Distribution

- Characterize the distribution of clinical attributes for suicide cases in families with high incidence ratios (high relative number of cases).

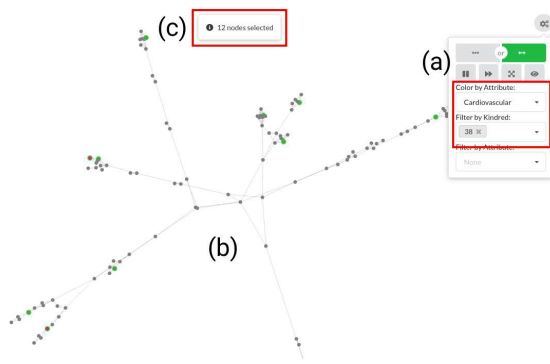


Figure 2: A graph of family 38 with 12 suicide cases selected. (a) Colored by cardiovascular attribute and filtered by family 38. (b) Selected nodes are highlighted in green. (c) Number of nodes selected, which corresponds to the number of suicide cases in family 38.

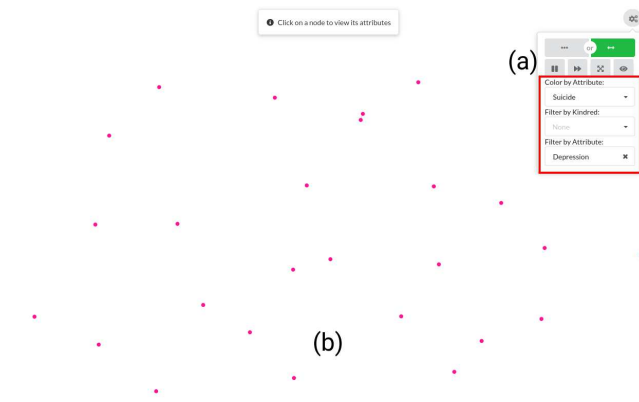


Figure 3: A graph of relationship between depression and suicide. (a) Colored by suicide cases and filtered by depression attribute. (b) While this figure shows that each person with depression has committed suicide, the contrary is not necessarily true. Not every person who has committed suicide was also in depression.

When characterizing a distribution, a much broader range of low-level tasks may be used to represent both the topology and the attributes. Because the graph needs to be filtered based on multiple attributes, the following set of tasks are used: [Find on Nodes + Filter on Nodes + Retrieve Value on Nodes + Count on Nodes + Filter on Nodes + Retrieve Value on Nodes + Scan on Nodes].

In Fig. 2, the number of suicide cases are counted by filtering suicide cases and selecting all the nodes within the graph. Then, the graph is filtered by family group where the nodes are colored by cardiovascular attribute.

### 3.3 Relationship

- Characterize (i.e, the relationship between cases and their attributes) suicide cases in families with high incidences of a given clinical attribute (such as depression).

This task is relatively trivial and may be broken down into the following set of tasks: [Find on Nodes + Filter on Nodes + Retrieve Value on Nodes + Scan on Nodes].

In Fig. 3, the graph is filtered based on depression attribute and colored by suicide cases to illustrate the relationship between the attribute and the cases.

### 3.4 Comparison

- Compare clinical information for suicide cases with their im-

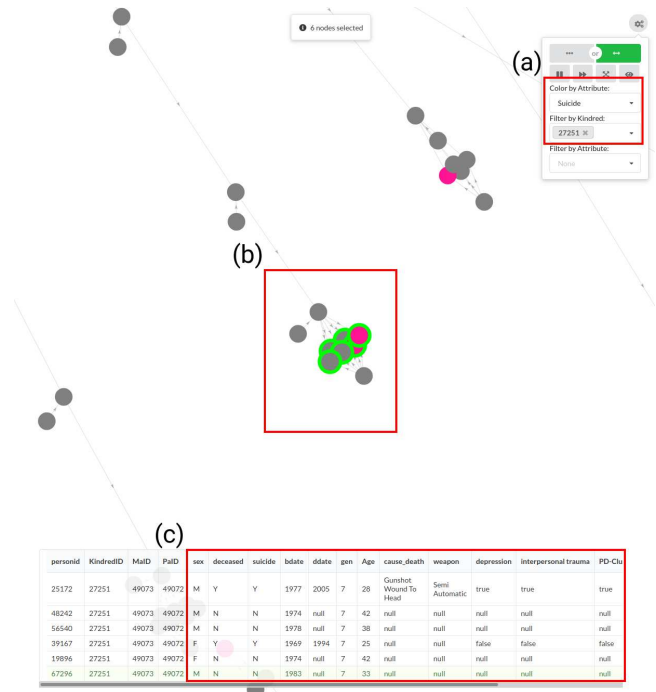


Figure 4: A graph of a family within 2 suicide cases. (a) Colored by suicide cases and filtered by family 27251. (b) 2 suicide cases (colored in pink) with their siblings selected, which are highlighted in green. (c) Table of clinical attributes for the selected nodes.

mediate relatives (siblings, parents, and children).

In order to compare clinical information of immediate relatives, a person of interest must first be identified and the relevant cases may then be selected for a detailed comparison. The task may be defined as follows: [Find on Nodes + Find Adjacent Nodes on Nodes + Retrieve Value on Nodes].

In Fig. 4, the siblings (adjacent nodes) are selected to compare clinical attributes. In this case, there are 2 suicide cases of similar age, but of different genders.

## 4 CONCLUSION

A robust visualization system and domain experts are often needed to perform high-level visualization tasks on multivariate graph data. Nonetheless, a basic set of interactions and visualization techniques is often sufficient to perform several high-level visualization tasks as long as they can be broken down into low-level tasks defined by the Graph Task Taxonomy.

## REFERENCES

- V. Asturiano. Force-directed graph rendered on html5 canvas, 2018.
- A. Crisan and C. Nobre. Biovis challenges workshop, 2020.
- B. Lee, C. Plaisant, C. S. Parr, J.-D. Fekete, and N. Henry. Task taxonomy for graph visualization. In *Proceedings of the 2006 AVI Workshop on BEyond Time and Errors: Novel Evaluation Methods for Information Visualization*, BELIV '06, p. 1–5. Association for Computing Machinery, New York, NY, USA, 2006. doi: 10.1145/1168149.1168168
- C. Nobre, N. Gehlenborg, H. Coon, and A. Lex. Lineage: Visualizing multivariate clinical data in genealogy graphs. *IEEE Transactions on Visualization and Computer Graphics*, 2018. doi: 10.1109/TVCG.2018.2811488
- S. van den Elzen and J. J. van Wijk. Multivariate network exploration and presentation: From detail to overview via selections and aggregations. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2310–2319, 2014. doi: 10.1109/TVCG.2014.2346441