Identification of Latent Networks in Psychiatric Comorbidity Using Graph Neural Networks

Georgia Smith
Faculty Advisor: Dr. James Vance
Department of Mathematics and Computer Science at UVA Wise

Introduction

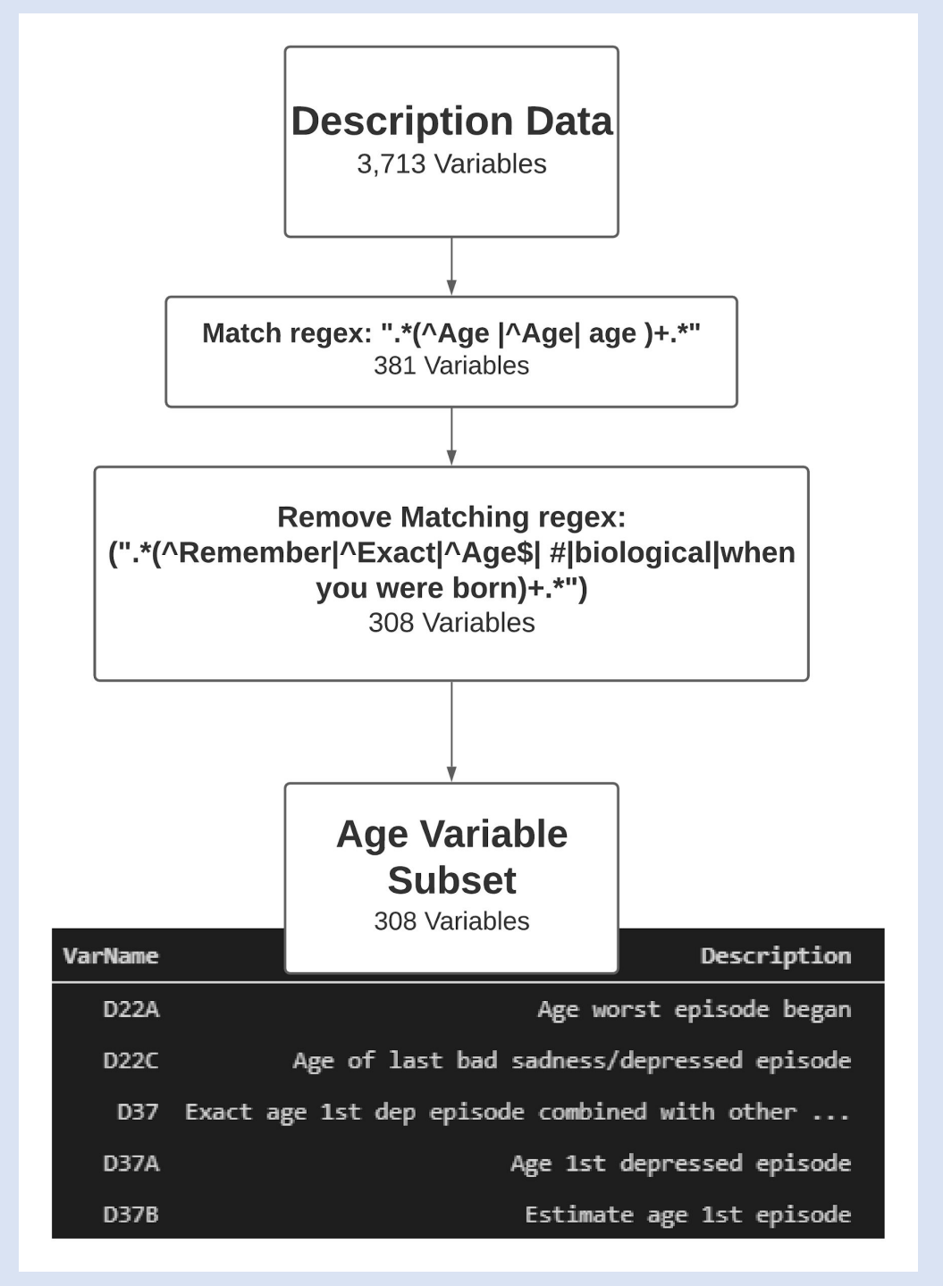
Psychiatric comorbidity (comorbidity) occurs when someone suffers from two or more psychiatric diagnoses at one time. Recently, the field of psychometric analysis, or the analysis of psychological measurement techniques and theories, has had two major ideas of how two analyze comorbidity: a latent variable perspective and a network perspective developed by Cramer et. al. [1]. We propose an extension of Cramer's original 2010 work using Graph Neural Networks (GNNs) to identify latent paths of comorbid networks. We use the Deep Graph Library (DGL) [7] to implement GraphSAGE to identify node connections by using Long Short-Term Memory (LSTM) on the National Comorbidity Survey Replication (NCS-R) dataset.

Data Cleaning

The NCS-R dataset is comprised of 9,282 subjects with 3,174 variables comprised of the subjects' answers to a questionnaire and diagnosis from mental health professionals based on their answers. Our first step was to extract the variables which we could use to initially infer connections - those which had to do with age of a symptom's and/or event's occurrence.

The NCS-R is hosted on the Inter-university Consortium for Political and Social Research (ICPSR) at University of Michigan. ICPSR includes a section using the Survey Documentation and Analysis (SDA) package which allows users greater insight into the data including descriptions for each variable. This information can also be accessed as a JavaScript array containing all variables and their description or queried by variable to access JSON objects.

We utilized the JavaScript array to develop a searchable data tree with all 3,174 variables which we then used to isolate the variables that had to do with age. The Process with which we isolated the age variables is outlined to the right. We used two regular expressions based on study of the variable descriptions to isolate 308 variables that had to do with age.


Description Data
3,713 Variables
Match regex: ".*(^Age |^Age| age)+.*"
381 Variables
Remove Matching regex: (".*(^Remember|^Exact|^Age$| #|biological|when you were born)+.*")
308 Variables
Age Variable Subset
308 Variables

```
VarName                                           Description
  D22A                                    Age worst episode began
  D22C                      Age of last bad sadness/depressed episode
   D37  Exact age 1st dep episode combined with other ...
  D37A                                    Age 1st depressed episode
  D37B                                     Estimate age 1st episode
```

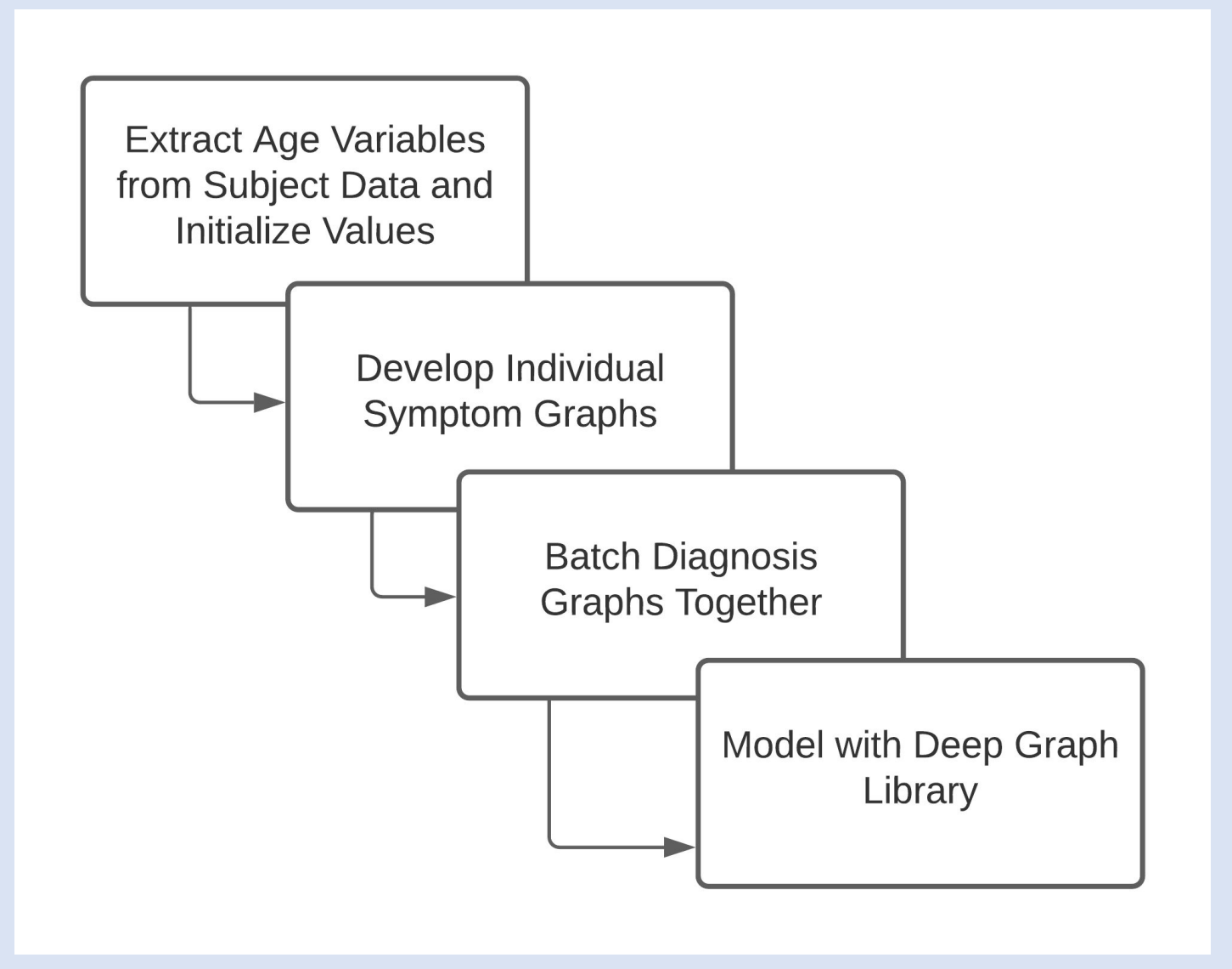
Programming Environment

Python was chosen as the primary programming language of this project due to its versatility, our prior expertise in the language, and because it Python is the language used in multiple prior works on latent networks providing us libraries and modules to utilize.

We first developed code in Jupyter Notebook for testing and debugging. Once the code was working satisfactorily we merged it into a Python file using Visual Studio Code. This code was then either run on a local machine or exported to the Rivanna HPC cluster, depending on computational intensity of the code. We then took code output, in the form of a CSV or serialized with Pickle, and imported it into Jupyter Notebook for further analysis and visualization development. Visualizations were developed using the NetworkX [3] and MatPlotLib [5] Libraries.

Methods

The outline of or methods can be seen to the right. Once we extracted the age variables we continued to develop what we term as 'Individual Symptom Graphs'. These graphs have nodes representing a subject's individual symptoms and directed edges made with the assumption that there is a possible causal influence between previous events and symptoms and current events and symptoms.

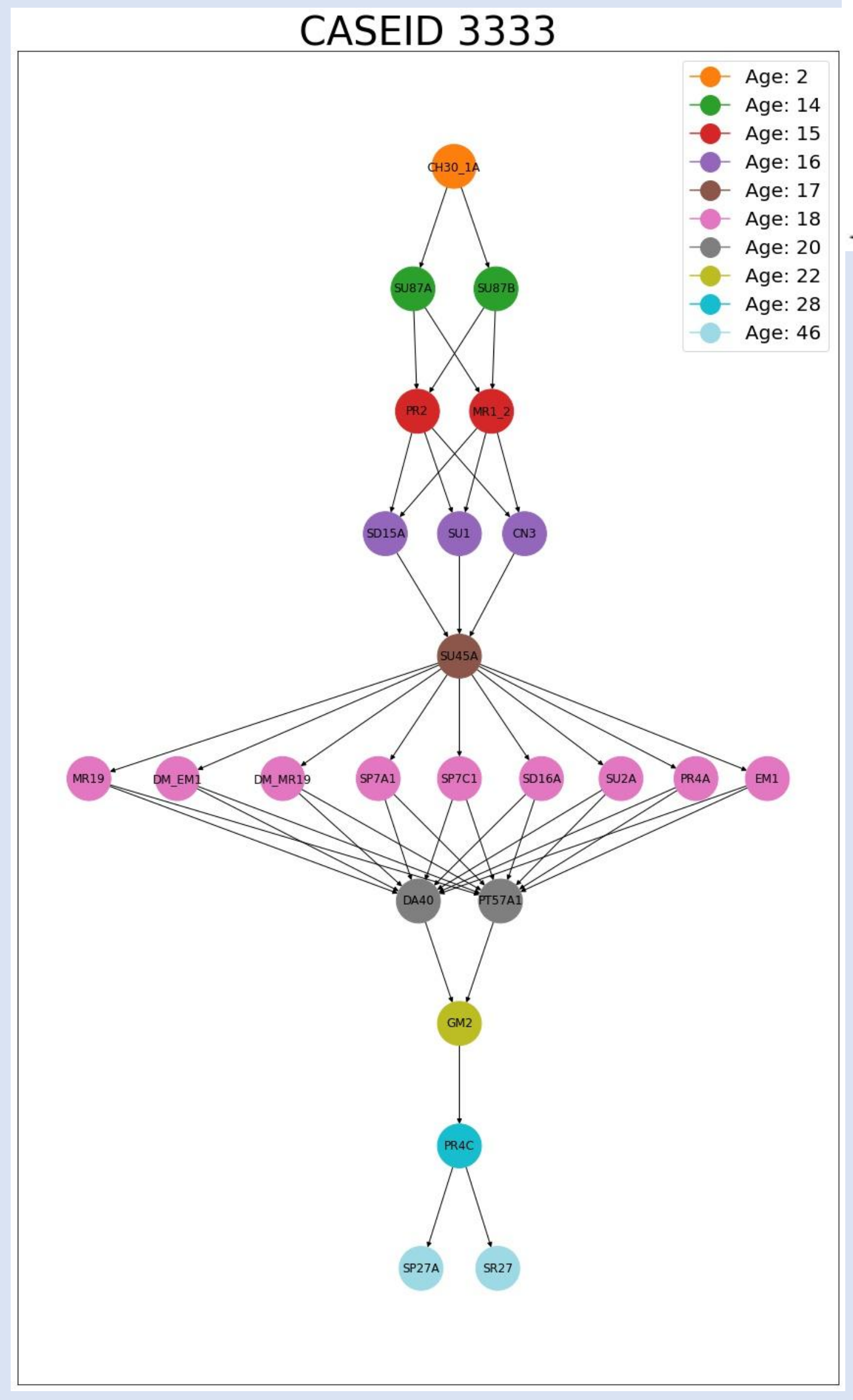

Extract Age Variables from Subject Data and Initialize Values
Develop Individual Symptom Graphs
Batch Diagnosis Graphs Together
Model with Deep Graph Library

Individual Symptom Graphs

To create an individual symptom graph we used Algorithm 1 based on Gomez-Rodriguez's 2012 work [2] on latent networks where events are only influenced by the most recent prior events. We first developed individual graphs for all 9,282 subjects before subsetting for individual and comorbid DSM diagnoses. Below, we have the individual graph for Subject Case ID 3333.

Algorithm 1: Building Individual Graphs

```
for $subject = 1, 2, \ldots$ do
    Sort $subject$ lifetime events by age
    $levels = \{$
    $age_1$: [$event_1, \ldots$],
    $age_2$: [$event_1, \ldots$],
    $\vdots$
    $\}$
    for $n = 1, 2, \ldots$ do
        if $n = 1$ then
            continue
        else
            for $event_n \in age_n$ do
                for $event_{n-1} \in age_{n-1}$ do
                    draw_edge_from_to($event_n$, $event_{n-1}$)
                end
            end
        end
    end
end
```



Batching Graphs Together

With individual graphs made and subset for further analysis, we the used the Deep Graph Library to batch graphs for a single diagnosis together. That is, we merged all individual graphs from a diagnosis into one large graph that treats each node from the individual graph as unique which we will refer to as a diagnosis graph.

Neural Network Architecture

This allowed us to feed the diagnosis graphs into the GraphSAGE algorithm [4] that outputs our data into ANN usable format. We then ran the diagnosis graphs through a Convolutional and Recurrent Neural Network architecture built into the Deep Graph Library to infer edges and node connections.

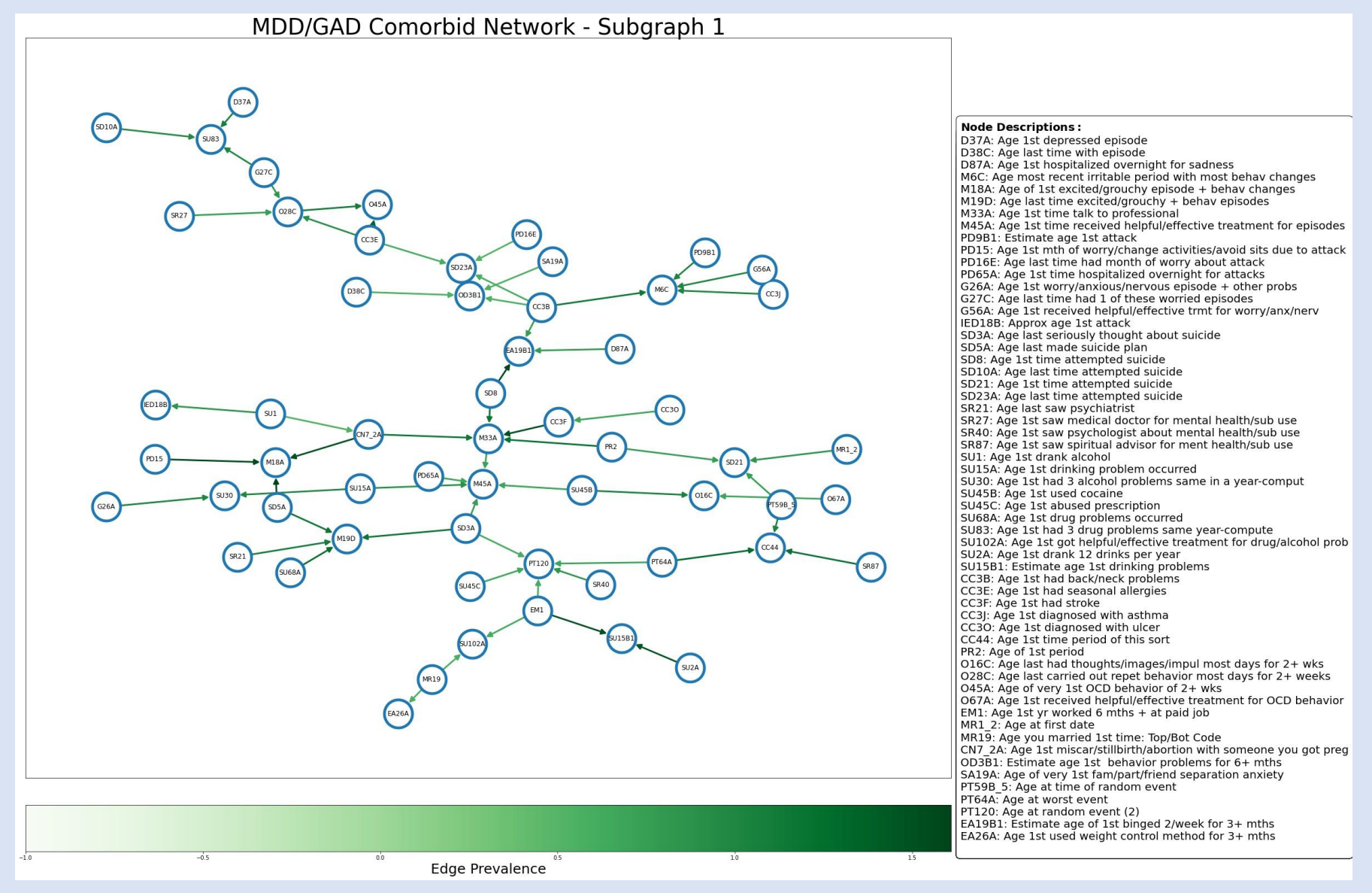
Contact Information

Email: ga.lyn.smi@gmail.com
Link for Code: https://github.com/025georgialynny/seminar
GitHub Link: https://github.com/025georgialynny

Results

With our model architecture defined we continued to test our network on the Major Depressive Disorder (MDDH) and General Anxiety Disorder (GAD) dataset since they have a substantial amount of comorbid cases (over 350). In preliminary testing both the MDDH and GAD graph neural networks predicted edges with between 74-77 area under the curve (AUC). This led us to the conclusion that the model was congruent with our data even if some noise is distorting the output.

GAD-MDDH comorbid cases performed similarly and we were able to get an output graph from those cases. A larger version of the following comorbidity graph can be found at the GitHub link, below.



Conclusions

We were able to use Graph Neural Networks to infer edge connections with moderate success which has potential use for aid in diagnosis and preventative care. Although, there is a large amount of noise when data is batched which may mask some more important causal relations, but it also allows for identifications of rarer pathways.

Overall this approach is a novel way to identify causal features of comorbid mental illness and could be enhanced to increase AUC and accuracy with more robust filtering methods and Individual Graph building.

Acknowledgments

I'd like to thank Dr. Vance for his perseverance through my process of finding a working method. Thanks to Dr. Hezy for pushing me to complete good work. Finally, I'd like to thank the entire MCS Faculty for putting up with my constant questions and encouraging me to learn more every day.

References

[1] Cramer, A. O., Waldorp, L. J., Van Der Maas, H. L., and Borsboom, D.(2010). Comorbidity: A network perspective.*Behavioral and brain sciences*,33(2-3):137. 1
[2] Gomez-Rodriguez, M., Leskovec, J., and Krause, A. (2012). Inferring networks of diffusion and influence.*ACM Transactions on Knowledge Discovery from Data (TKDD)*, 5(4):1–37. 1
[3] Hagberg, A. A., Schult, D. A., and Swart, P. J. (2008). Exploring network structure, dynamics, and function using networkx. In Varoquaux, G., Vaught,T., and Millman, J., editors,*Proceedings of the 7th Python in Science Conference*, pages 11 – 15, Pasadena, CA USA. 1
[4] Hamilton, W. L., Ying, R., and Leskovec, J. (2017). Inductive representation learning on large graphs.CoRR, abs/1706.02216.1
[5] Hunter, J. D. (2007). Matplotlib: A 2d graphics environment.*Computing in Science & Engineering*, 9(3):90–95. 1
[6] Kessler, R. C., Berglund, P., Chiu, W. T., Demler, O., Heeringa, S., Hiripi,E., Jin, R., Pennell, B.-E., Walters, E. E., Zaslavsky, A., et al. (2004). The us national comorbidity survey replication (ncs-r): design and field procedures.*International journal of methods in psychiatric research*, 13(2):69–92. 4, 5, 6,7, 8
[7] Wang, M., Zheng, D., Ye, Z., Gan, Q., Li, M., Song, X., Zhou, J.,Ma, C., Yu, L., Gai, Y., et al. (2019).Deep graph library: A graph-centric, highly-performant package for graph neural networks.*arXiv preprintarXiv*:1909.01315. 1