MindMatrix: Decoding Mental Health Through Search Data

Authors: Nika Faraji , John Hennigan

*****_Each author contributed equally to the design, coding and development, analysis, and writing of this project_

December 2024

Abstract

The ability to predict the overall utilization rate of mental health services for the upcoming year would provide significant benefit to State governments looking to allocate budgets and determine necessary funding levels. This project aims to provide methods and the models derived from them to identify a States projected mental health utilization rate based on the current and previous years data. This project combined data from the Google Trends API, and N-MHSS(Nation Mental Health Services Survey)[1] datasets. The models produced by this project were able to accurately predict the upcoming years mental health utilization rate and could be incorporated in a planning process for a State government.

Background

This project aims to answer the following question:
Can Google search trends help predict an increase in mental health treatment utilization?
This can help provide useful information to State government officials about the percentage of people who seek mental health treatment once they search about related topics such as diagnosis and treatment centers on Google. According to the CDC[2], the US is currently in a mental health crisis, so being able to predict dips and upticks in mental health treatments could help provide critical insights for allocating resources and implementing timely mental health campaigns. Our initial *hypothesis* is; A rise in mental health search topics on Google correlates with a rise in patients seeking treatment at mental health centers. Our *prediction* is that states can anticipate increased mental health treatment utilization by monitoring Google search trends about mental

[1] Data Files | CBHSQ Data SAMHSA
[2] Center for Disease Control, August 8, 2024 Protecting the Nation's Mental Health | Mental Health | CDC

health. This question is somewhat novel in that there have been other projects that have circled around Google Search's and mental health related topics (ie. Searching for online therapy, general mental health information) but not necessarily focused on mental health awareness. For instance this project had a focus on ADHD but did notice trends that may show a leaning to overall trends, ADHD/Google Research Project.

Data

Data Acquisition

The data used in this project will be a mix of data from datasets from the Substance Abuse and Mental Health Services Administration (SAMHSA), a US government agency, including the MH-CLD (Mental Health Client Level Data) Data Files | CBHSQ Data SAMHSA, as well as data received from the Google trends API for common mental health related search terms, including but not limited to "therapist near me", "Depression", "ADHD", "Anxiety", etc. mental health - Explore - Google Trends, which are all in CSV form. The health data is a good choice for this purpose because it gives an estimate of how many people in each state are receiving some kind of mental health treatment.

Data Cleaning

We curated our own dataset by pulling from these two datasets and combining different variables which were significant to our question. We chose variables from the health data and google keywords data and merged them on state and year so that each row represents a specific state's data values in a specific year (from years 2013-2022). The variables which were aggregated from the health data include diagnosis totals and utilization totals such as;

“anxiety_ct”, “depression_ct”, “ADHD_ct”, “PTSD_ct”, “bipolar_ct”, “tot_service”, which is the total number of people who were checked into a mental health facility by state every year. We then had to include state populations estimates from the U.S. Census for the years 2010-2020 in CSV form and had to pull these estimates separately for years 2021-2022, to get a sense of the percentage of people who received treatment. Thus, our target variable ended up being “utilization_rate” which is the total service value divided by the population value to give us the proportion of utilization in a state. The features which were included in our data from the google trends data consisted of several keywords which we chose to include in our analysis such as “depression”, “anxiety”, “PTSD”, “ADHD”, “bipolar”, “therapist near me”, “psychiatrist near me”, and “mental hospital near me”. These keywords were chosen to mirror the diagnosis totals from the health data. In the google data, these keywords are given a score between 0 -100 depending on its popularity in that year, scores closer to 100 indicates they were highly popular in search trends. We chose to use the measure of tendency values for the keyword score values (median and max) to have a better estimation of the popularity scores. Once all of the variables were chosen and aggregated, the inner merge was done to line up all the features by “state_id” and “year”, meaning that since the health data didn’t include a few States’, the merge was only done on the states which were already in the original data.

Data Exploration

For the EDA, we used visualization techniques such as creating line plots, barplots, pie charts, as well as correlations matrix to indicate collinearity between variables. In most cases, we used average values or total values by either “year” or “state_id” to be able to compare trends and patterns. We also used summary statistics such as the describe() function to summarize the statistics of our variables. Below is the summary statistics for our continuous variables.

The following charts depict the changes over time for each Google Trends search criteria and each state, as well as Washington DC.While difficult to see any individual states trends, these do illustrate common trends which can be expounded upon. It is important to note that just because there is an increase in the mean search interest for any of the keywords it does not necessarily mean it is because of increase in utilization and may need to be considered as outliers.

As seen in figure 1, the increase in searches for ‘ADHD’ in 2022 may have in part caused by the ongoing Adderall Shortage[3].

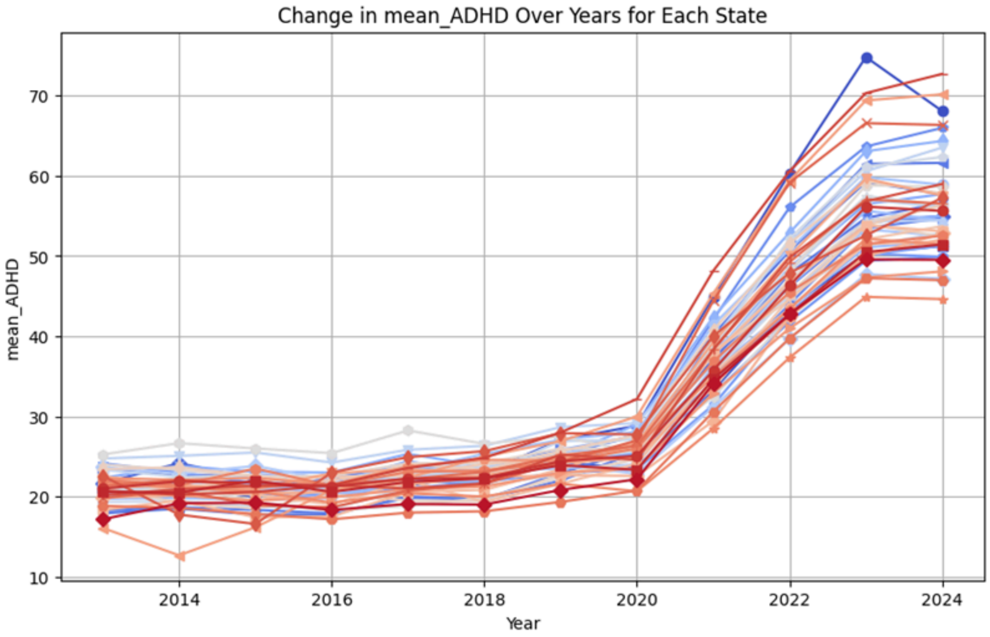


Figure 1

[3] FDA confirms shortage of Adderall, with delays possible until end of year

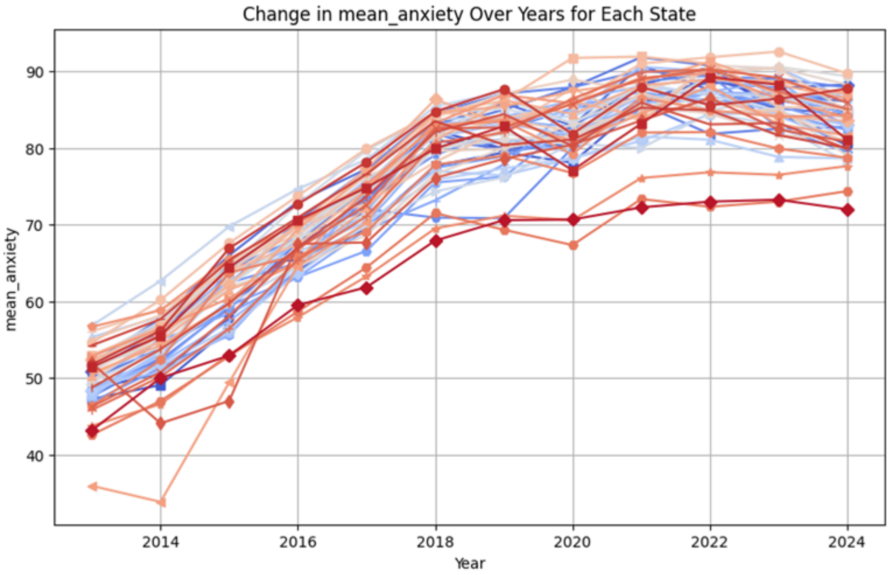


Figure 2

In the plot above, the searches for ‘Anxiety’ increase drastically over the past 5 years and reach their peak around 2020 which could be a direct result of COVID.

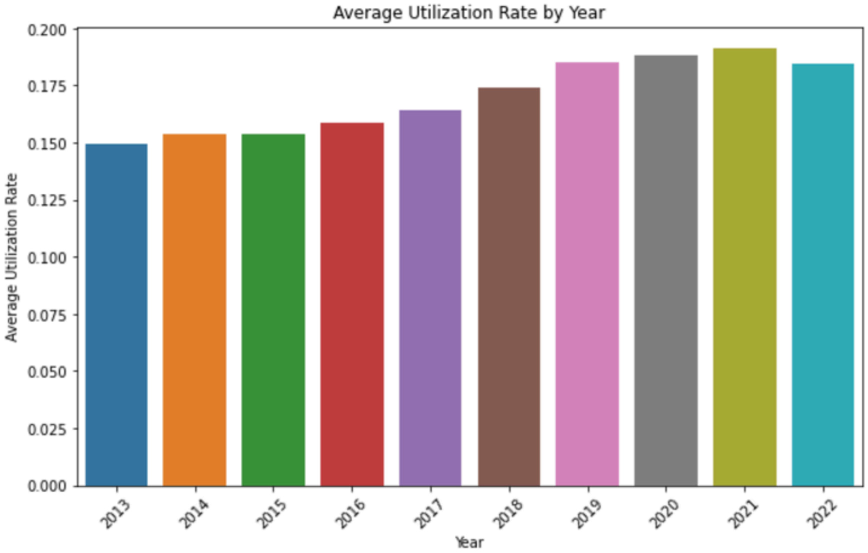


Figure 3

In this bar graph, the average utilization of mental health treatment throughout all US states is compared throughout the years. It is apparent that this average has increased somewhat every year since 2013, specifically around the time of COVID-19. However, we see a slight decrease in 2022 which can be from the gradual relief of COVID related tensions.

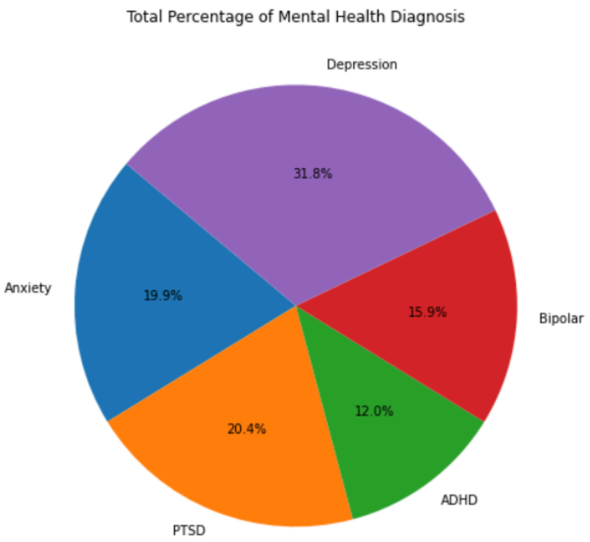


Figure 4

In the pie chart in Figure 4, the percentage of mental health diagnosis in the US between the years of 2013-2022 is shown with Depression being the most diagnosed followed by anxiety.

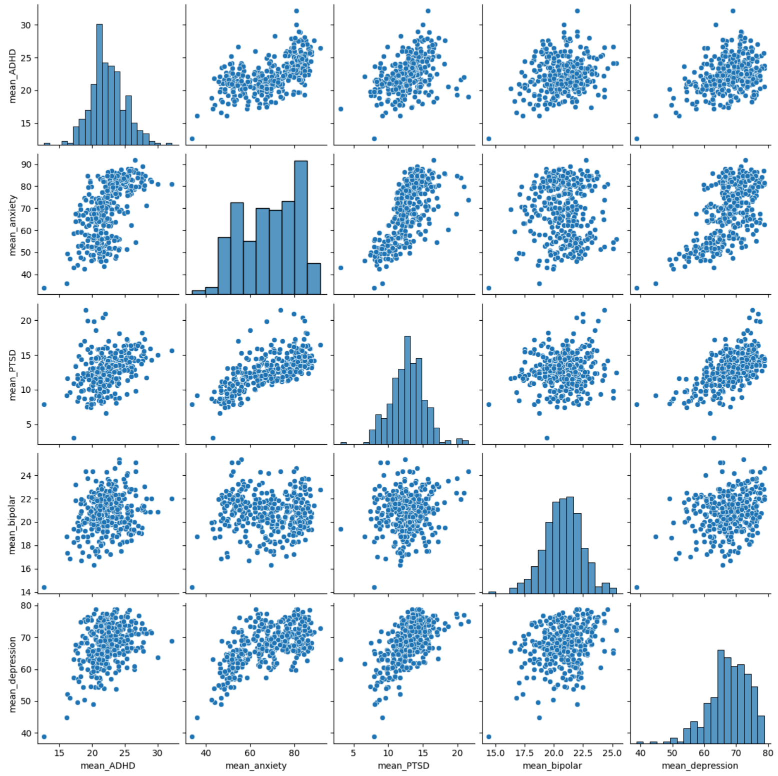


Figure 5

Figure 5 illustrates the collinearity within our diagnostic feature set derived from PyTrends data. The charts reveal a somewhat notable collinearity among anxiety, PTSD, and depression, which aligns with the frequent comorbidity observed in these conditions[4][5]. This collinearity is not unexpected but underscores the need for meticulous data preparation. Ensuring that this pattern does not introduce bias or unfair impact is crucial as we proceed with analysis and modeling.

Models

Preprocessing & Feature Engineering

For our preprocessing and feature engineering step before modeling, the most important thing is to get the data in the format needed for our machine learning models. Fortunately, most of our variables are numerical so we did not need to do any encoding. The only categorical variable is the 'State' variable with the state abbreviation, but we already have another variable 'STATEFIP' which is the state code for each corresponding state, and likely won't be treated as a feature. We had no missing values because when we combined our datasets we used an inner join where all the data points only merged if there was data for both sets being joined. We also don't have a lot of dimensionality so in this case dimensionality reduction is not needed since we built our own dataset.

Using the 'calcsplitratio' method which was provided to us, the ideal split ratio for our dataset is **0.86:0.14.** This means that it is best if we use 86% of our data for training purposes and 14% for testing. This method is efficient in helping us choose a fitting split ratio based on our dataset rather than choosing a random split we think may be good.

[4] The Comorbidity of Anxiety and Depression | NAMI: National Alliance on Mental Illness

[5] Psychiatric Co-Morbidities in Post-Traumatic Stress Disorder: Detailed Findings from the Adult Psychiatric Morbidity Survey in the English Population - PMC

A feature selection method we chose for this step of the project is K-means clustering followed by PCA. This is an unsupervised feature reduction/selection technique which will give us some insight into the feature importance of this dataset. For this portion, we are focusing on the keyword data and features because we are interested in seeing the trend of people who are searching these terms on Google and who may end up receiving some sort of treatment as a result. The reason we chose to do clustering is because we are not necessarily trying to do any dimensionality reduction. Instead, this clustering will help us identify distinct groupings of states into different clusters and help visualize and interpret the different groupings.

The first step is to standardize the data using the features we wanna use in the clustering which in this case is the trend data keyword columns. The next step of the K-means clustering is to use the Elbow Method to choose the optimal *k* value which is the number of clusters. Once the elbow method is plotted, we can see that the optimal *k* value is 3.

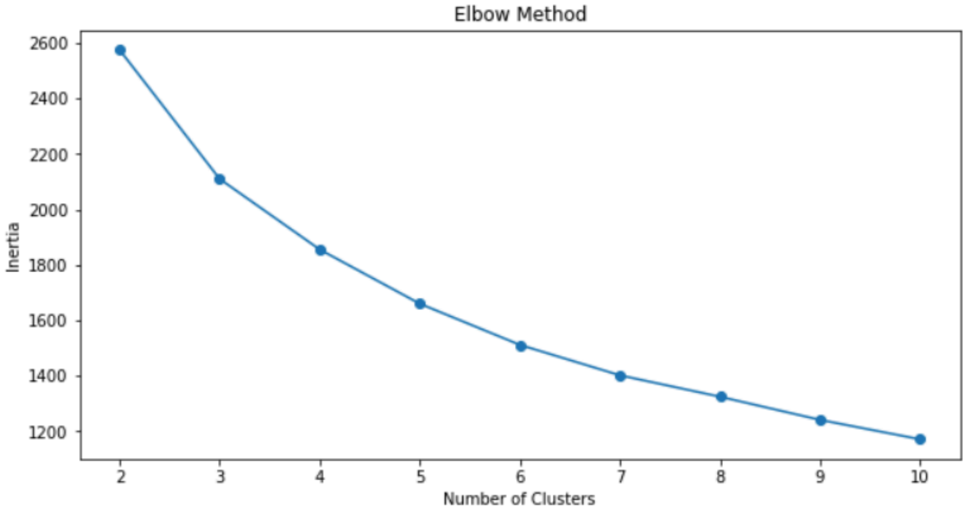


Figure 6

We then fit K-means using this optimal *k* value for the chosen features and add the clusters to the original dataset where every row is given a cluster number. We then can analyze the clusters using the mean values of each feature for each cluster. Looking at the summary of the mean of the features in each cluster group, it is apparent that cluster 0 is grouped by lower values of diagnosis and popularity score of the keywords, followed by cluster 1, and lastly cluster 2 has the highest mean values for each feature including diagnosis, utility rate, and keyword popularity score. Once we have our clusters and optimal *k* value, we use PCA to analyze the contribution of original features to these components. The PCA component loadings indicate which features are most important in defining the clusters in reduced dimensions. With PCA, we can also visualize the clusters better in the figure below. The clusters are clearly segmented with little overlap.

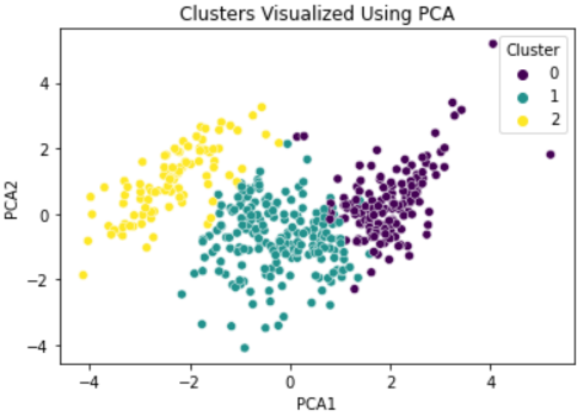


Figure 7

The last step is to look at the feature importance for each cluster and see which features have the most significance in segmenting each cluster. This can be done visually and by printing the feature contributions using the components of the PCA. The PCA component loadings indicate which features are most important in defining the clusters in reduced dimensions. Below is the significance rate of each feature for the 2 components.

Another way to analyze the feature importance is using the K-means cluster centroids, which represents the mean value of each feature in each cluster. Features with larger absolute values for a specific cluster centroid are more significant for that cluster. A heatmap can be used to visualize the importance of features across clusters. In the heatmap below representing the cluster centers, features with higher values in a cluster's centroid are more significant for that cluster. For example, we can see that for **cluster 1,** the keywords 'bipolar' as well as 'psychiatrists near me' have the highest values. For **cluster 2,** 'depression' is the most prominent and for **cluster 3**; 'ADHD' and 'therapist near me'. Features that vary most across clusters in the heatmap are the most significant overall in separating the clusters. Ultimately, clients from different states are segmented into these 3 clusters based on their Google search words which are different mental health diagnoses.

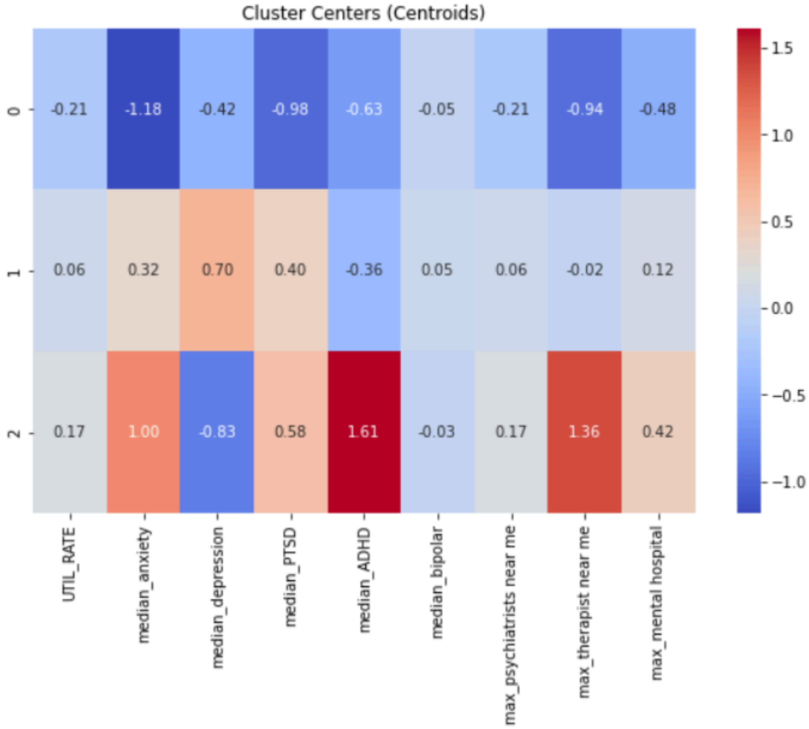


Figure 8

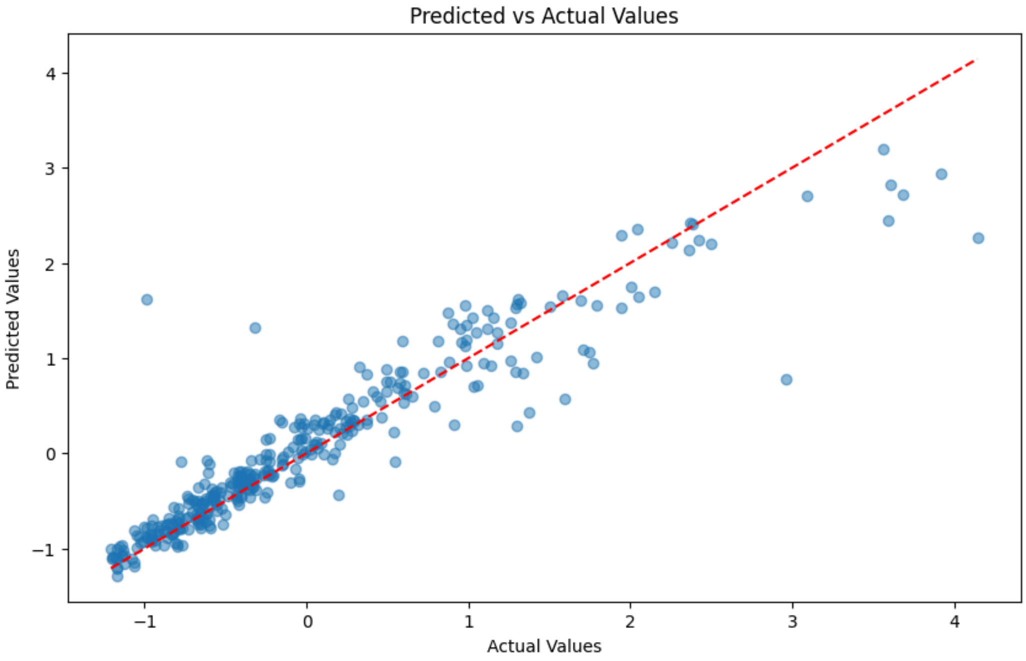
Algorithm Selection

The reasoning behind selecting a neural network as a model to explore was the overall ability of neural networks to adapt to complex data. Our initial concern was that since our dataset was relatively small a neural network may not be able to fit the data well enough. However the model was able to achieve a moderate degree of success early on which strengthened our decision to continue trying to tune the model.

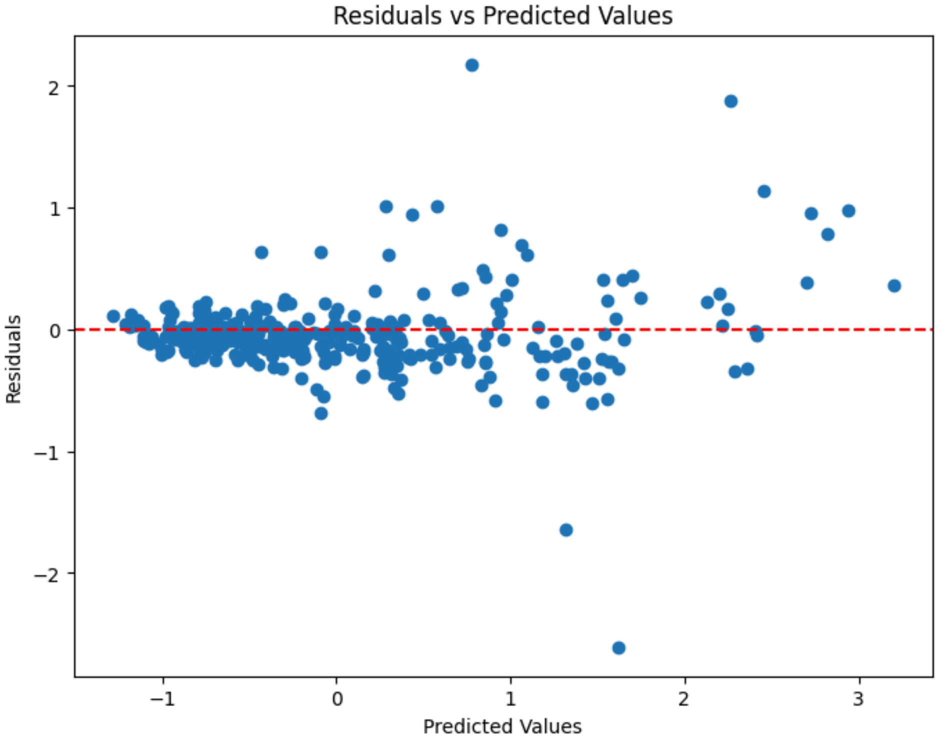
For the neural network we assumed a normal distribution of input features which was achieved by scaling the data prior to training. Other assumptions included that the entries were independent, and the data was collected independently. Given the nature of our datasets and where they were derived from, as well as how we aggregated and joined the data these assumptions were deemed to be sound. While homoscedasticity was not necessarily assumed the model still was able to produce results that show a mostly random distribution of residuals.

The evaluation of the model was done using R-squared and mean average error (MAE). The model performed admirably with an average R-squared value of 0.84, using 5 k-fold validation. The range for each fold was within 0.82 to 0.88. The MAE of the model on the same k folds was 0.079 which gives a strong indicator that the model was fitting well to the data but was not overfitting. As the target variable approached higher values the model did tend to under predict which was likely due to the lack of data with extreme values for the target variable. As more data becomes available for this question it is anticipated that this problem would begin to lessen. The below graphs provide a visual representation of the models overall performance.

These graphs use predictions from the entire dataset including the training data



NN Figure 1

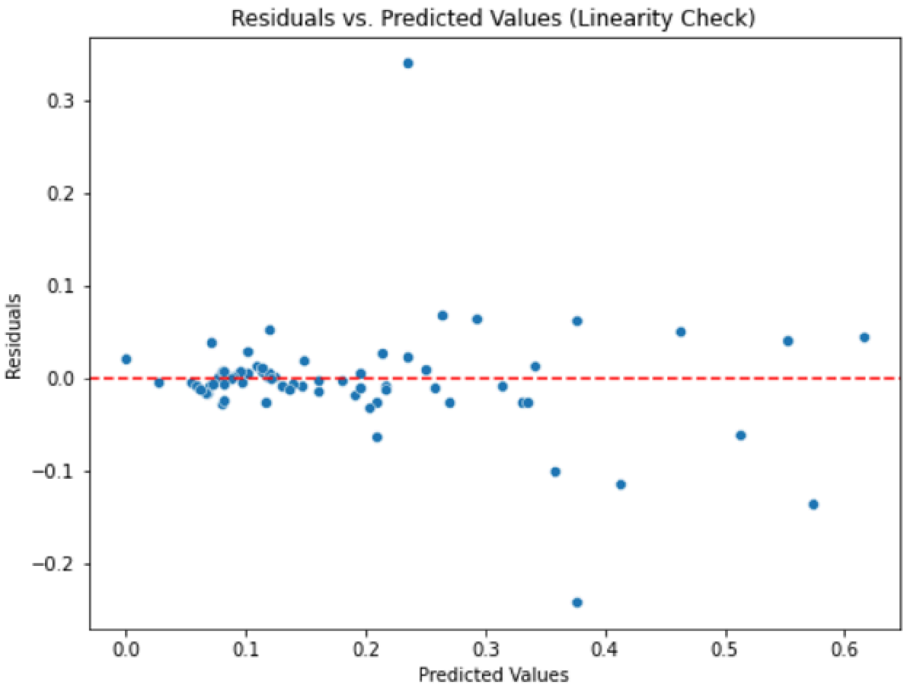


NN Figure 2

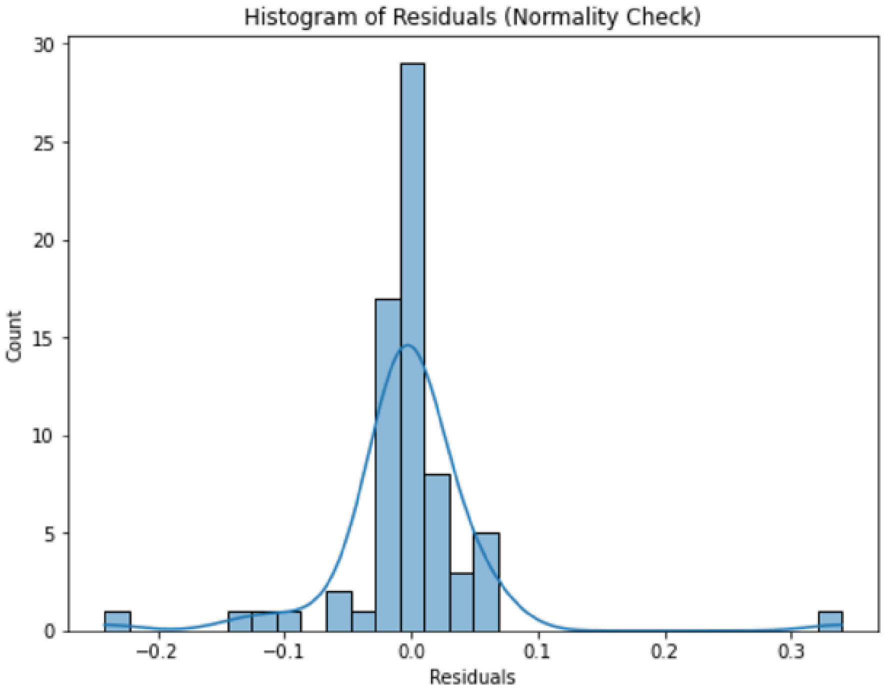
We chose to do an OLS linear regression model because our target variable is continuous rather than a binary categorical variable. Linear regression has straightforward mathematical foundations and is computationally efficient. It also provides coefficients for each predictor, which can be interpreted as the average change in the dependent variable for a unit change in the predictor which also helps identify which predictors are most influential in explaining the variability in the dependent variable. It also is a good baseline for comparison when evaluating more complex models, which in our case is a deep learning Neural Network model.

The assumptions for this model are linearity, normality of residuals, homoscedasticity, no high influence, data collected randomly & independently, and no multicollinearity of predictor variables. In the residuals vs. predicted plot (figure 1) below, linearity is tested and validated as the points mostly line up around the line. The independence of errors is tested using the Durbin-Watson test which gives a Durbin-Watson Statistic of about 1.7387. This value is close to 2 which indicates that there is no significant autocorrelation. Homoscedasticity is checked using the Breusch-Pagan test which gives a p-value of about 0.4145, which is greater than 0.05 thus proving homoscedasticity. However, the model does not satisfy the normality and multicollinearity assumptions. The normality plot in figure 2 shows that it skews away from normality. The multicollinearity test also shows multiple features with very high collinearity.

The evaluation of the model is done using mean squared error and R-squared score. The R-square score for this model is 0.93 which is a very high score and means that it is 93% accurate in predicting the utilization rate. This is alarming because it is almost too accurate and could mean that the model is overfitting. Similarly, the MSE of the model is showing as 0.00. This is also not normal because there often is some error. After further evaluation, this is most likely because the variance of our target variable is almost 0, which makes this an unfit model.



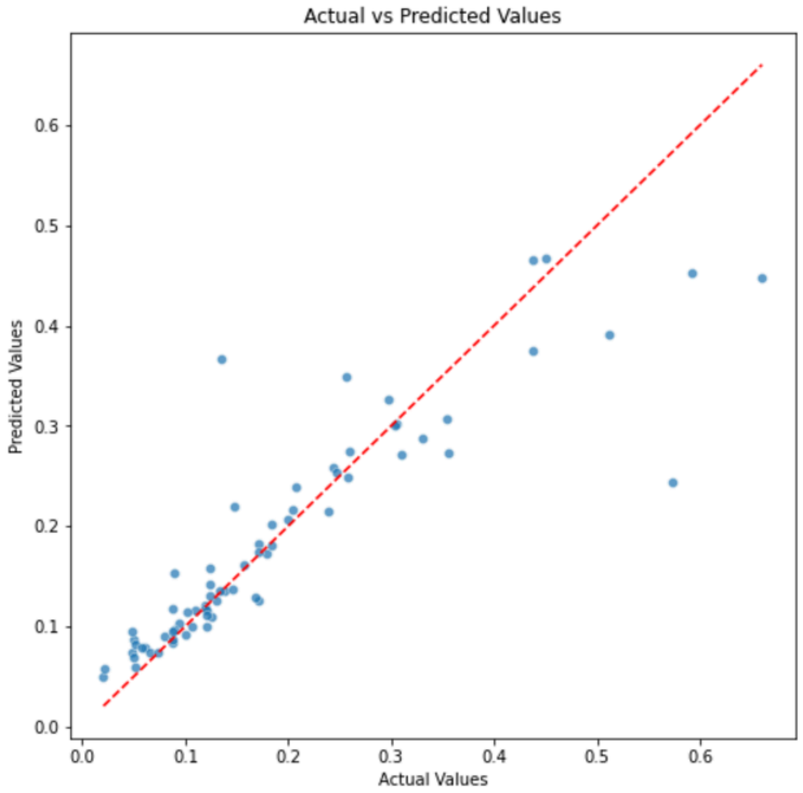
Regression Figure 1



Regression Figure 2

After not getting clear results from the OLS regression model, we decided to try a Random Forest model. At first, we trained the model with standard parameters of n_estimators=100, max_depth=none, min_sample_split=2, min_sample_leaf=1, and max_features=sqrt. This gave us an R-squared score of about 0.83. With further exploration to prevent overfitting, we decided to perform hypertuning and cross-validation to this model using the “GridSearchCV” function from sklearn. GridSearchCV implements a “fit” and a “score” method. The parameters of the estimator used to apply these methods are optimized by cross-validated grid-search over a parameter grid.

We first create this parameter grid with different parameters to find the optimal hyperparameters. Then we perform the grid search using the training set and have it output the best parameters. The results were n_estimators=100, max_depth=15, min_sample_split=2, min_sample_leaf=2, and max_features=sqrt. After hypertuning the model using these values, the model had an R-squared score of 0.78. The plot below shows the actual vs. predicted values which shows it is performing well with some deviations and outliers. The mse still appears to be small (0.004) which may be because these regression models may be too simplistic to capture the complex relationships between the independent and dependent variables. The hypertuning was performed to prevent overfitting which means that the model is still learning the data too well, making it not a feasible model for our purpose.



Random Forest Figure 1

Final Model

The final model chosen for consideration by our team is the Neural Network. There is some concern over the regression model's ability to adapt to new data as well as its low error rate requiring more research before it could be deemed reliable. The Neural Network showed a strong ability to predict outcomes and adapt to new data.

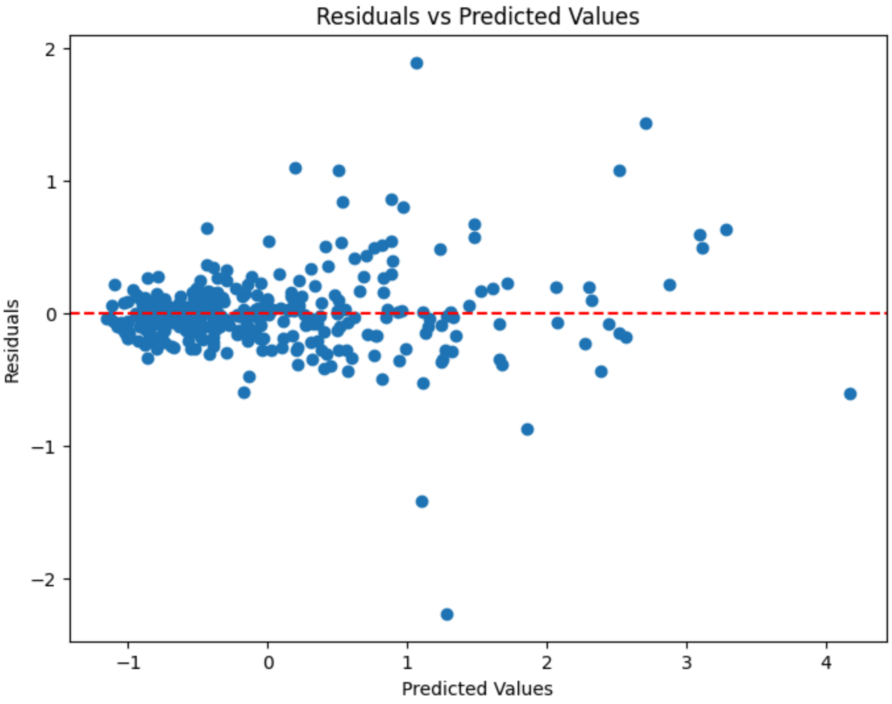
The hyperparameters for the final model were derived using a random search on 1000 models. The evaluation was done on R-squared values and resulted in the following parameters.

{'model__optimizer': 'adam', 'model__learning_rate': 0.001, 'model__layers': [32], 'model__dropout_rates': [0.2], 'model__activations': ['relu'], 'epochs': 70, 'batch_size': 16}

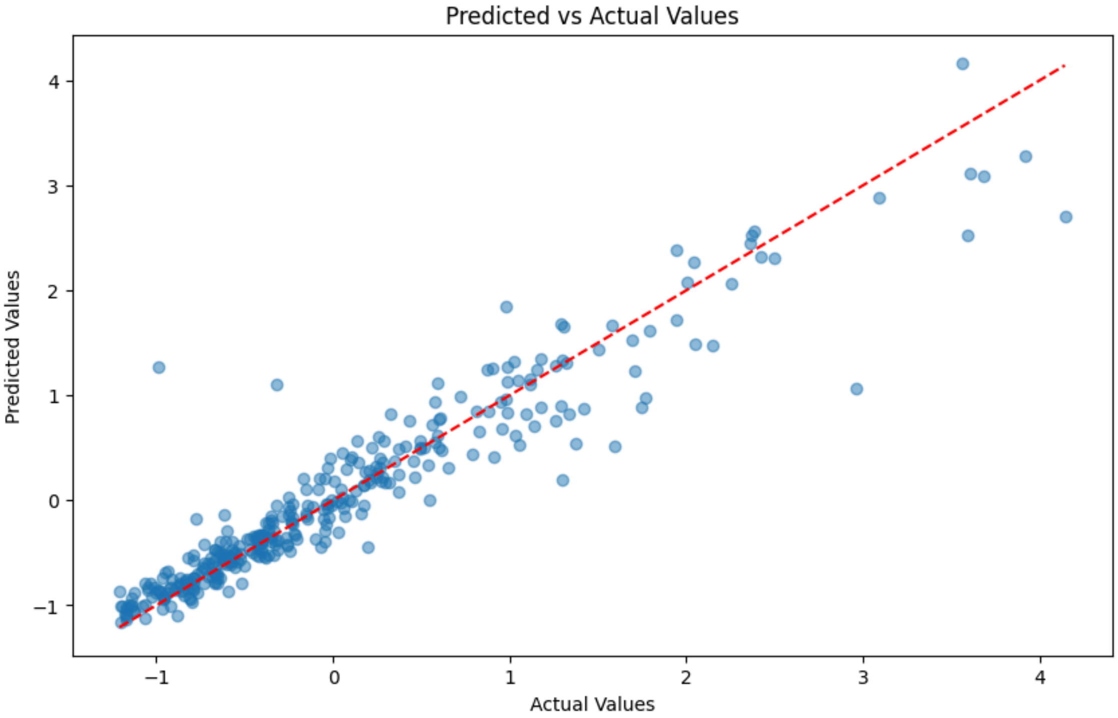
This shows a model with a single hidden layer with 32 nodes using relu activation followed by a dropout layer with rate of 0.2 and a final Dense layer with a size of 1 to provide the prediction. Additional further feature selection was performed pre training to ensure that the model was not fed features with a high level of collinearity or features that were mainly adding noise to the data. This was done using a three step process. First the features are checked for collinearity and if a feature has a correlation of greater than 0.95 with an existing feature the second feature is removed from consideration. Then the remaining features are fed into a Random Forest Regression model and the top 12 features are selected (12 features was determined to be the least amount of features we could feed to the final model before drop off in predictions occurred). The remaining 12 features are then validated with cross validation. The feature set provided to the final model was as follows

> 'prev_util_rate', 'mean_depression', 'prev_max_depression', 'prev_mean_depression', 'prev_mean_PTSD', 'mean_PTSD', 'prev_max_anxiety', 'mean_bipolar', 'max_anxiety', 'prev_max_ADHD', 'prev_mean_bipolar', 'prev_mean_ADHD'

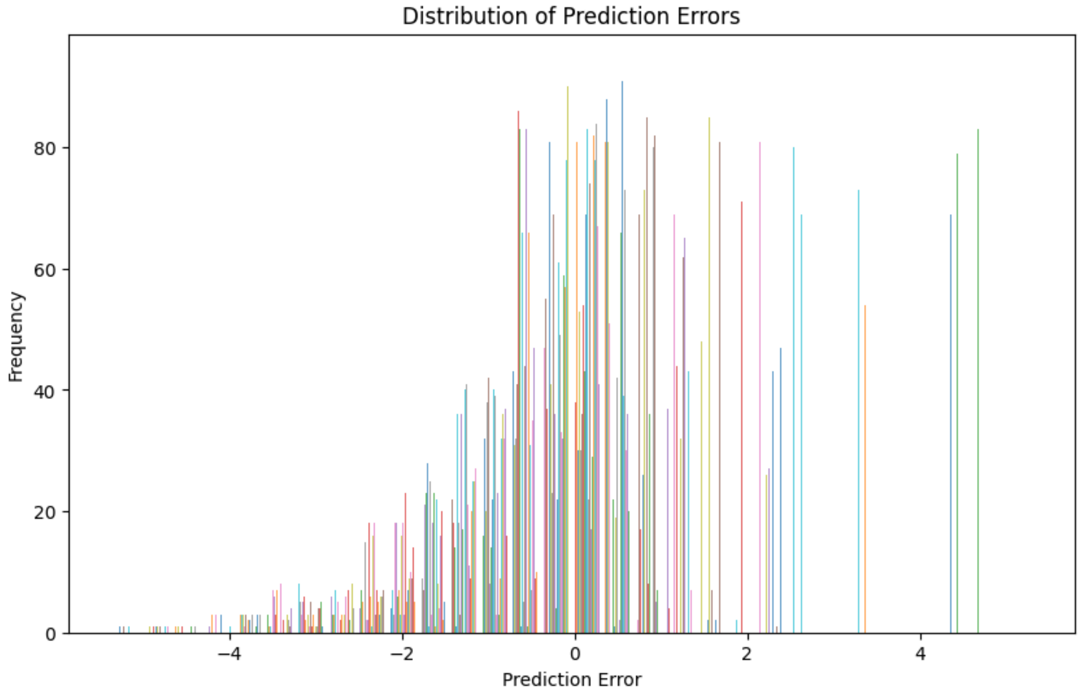
The model has an overall R-squared value of 0.901 and MAE of 0.096 which is a very strong result especially considering the size of the dataset. The model did perform better with lower target values than higher but did predict closer to the actual values for higher targets than previous models. It also did not just under predict high target values which gives a stronger indicator that it will fit to future data more accurately.



Final Model Figure 1



Final Model Figure 2



Final Model Figure 3

Conclusions

Given that the final model for this project was able to predict the utilization rate for the coming year with an R-squared value of over 0.9 it is our contention that the answer to the question proposed by the project is that we are able to predict the following year's utilization rate using Google Trends data. Given that this final model was able to achieve a p-value of less than 0.05 during a paired t-test on each of the folds in our k-fold validation, we feel we can reject the null hypothesis that google search trends do not help predict health care utilization rates. Thus we can provide our stakeholders, in this case state governments, with this tool to help them monitor and budget for the upcoming year’s utilization rates for mental health resources.

Discussion & Next Steps

The next steps for the neural network portion of the code will be to make the majority of the code a “one-click” solution for creating and training the model and producing a result. As training a model is not an exact science and requires a certain level of expertise to perfect this is not intended to completely and unequivocally answer the question posed at the beginning of this project. This is simply a way to monitor and compare the results generated now and as more data becomes available. Since the dataset we have for this question is relatively small, being able to quickly generate similar models as the dataset grows larger will help to further tune predictions and solidify the answers provided by this project. Similarly by making it simple to add to the current dataset as new data becomes available it makes it possible to test the current model on new data and see if the answers remain true.

This also feels like a very practical approach given that new target data would only become available annually. Having a job that also ran annually to create a new model and one to test the existing would be useful to decision makers who were relying on this data. New data from the google Trends API will become available extremely quickly as well so being able to incorporate this and update the models feature set would also be of benefit.

Repository Link

https://github.com/jonjonbinx1/GSMH.git

Appendix

Figures legend:

- Figure 1 - line plot that shows the change in mean ADHD trend score over the years of 2013-2022 for every state. Each line represents each state
- Figure 2 - line plot that shows the change in mean Anxiety trend score over the years of 2013-2022 for every state. Each line represents each state
- Figure 3 - bar plot that shows the average utilization rate for all of US between the years of 2013-2022
- Figure 4 - pie chart that shows the percentages of mental health diagnosis in the US between the years of 2013-2022
- Figure 5 - collinearity scatterplots that show collinearity between trends data features
- Figure 6 - line plot showing the k-means elbow method.
- Figure 7 - visualization of the clusters using PCA for feature selection
- Figure 8 - heatmap of cluster centroids which shows the feature importance for each cluster
- NN figure 1 - plot showing predicted vs actual values of the neural network model. The closer the points are to the diagonal line, the more accurate the model
- NN figure 2 - plot showing residuals vs predicted values of the neural network model. The closer the points are to the red dotted line, the more accurate the model
- Regression figure 1 - plot showing residuals vs predicted values of the OLS regression model. This plot helps validate the linearity assumption of the model
- Regression figure 2 - histogram showing the distribution of residuals. This plot helps validate the normality assumption of the model

- Random forest figure 1 - plot showing predicted vs actual values of the neural network model. The closer the points are to the diagonal line, the more accurate the model
- Final Model figure 1 - plot showing residuals vs predicted values of the neural network model. The closer the points are to the red dotted line, the more accurate the model
- Final Model figure 2 - plot showing predicted vs actual values of the neural network model. The closer the points are to the diagonal line, the more accurate the model
- Final Model figure 3 - plot showing the distribution of prediction errors

Final Data Features:

['prev_util_rate', 'prev_max_ADHD', 'prev_max_PTSD', 'prev_max_anxiety', 'prev_max_bipolar', 'prev_max_depression', 'prev_max_mental hospital', 'prev_max_psychiatrists near me', 'prev_max_psychologist near me', 'prev_max_therapist near me', 'prev_mean_ADHD', 'prev_mean_PTSD', 'prev_mean_anxiety', 'prev_mean_bipolar', 'prev_mean_depression', 'prev_mean_mental hospital', 'prev_mean_psychiatrists near me', 'prev_mean_psychologist near me', 'prev_mean_therapist near me', 'prev_median_ADHD', 'prev_median_PTSD', 'prev_median_anxiety', 'prev_median_bipolar', 'prev_median_depression', 'prev_median_mental hospital', 'prev_median_psychiatrists near me', 'prev_median_psychologist near me', 'prev_median_therapist near me', 'max_ADHD', 'max_PTSD', 'max_anxiety', 'max_bipolar', 'max_depression', 'max_mental hospital', 'max_psychiatrists near me', 'max_psychologist near me', 'max_therapist near me', 'mean_ADHD', 'mean_PTSD', 'mean_anxiety', 'mean_bipolar', 'mean_depression', 'mean_mental hospital', 'mean_psychiatrists near me', 'mean_psychologist near me', 'mean_therapist near me', 'median_ADHD', 'median_PTSD', 'median_anxiety', 'median_bipolar', 'median_depression', 'median_mental hospital', 'median_psychiatrists near me', 'median_psychologist near me', 'median_therapist near me']